

ANTONIO VIGILANTE

TE

Manuale  
pratico

di didattica  
digitale

XT

US

della  
filosofia

ZENODO

Antonio Vigilante

# **TEXTUS**

Manuale pratico di didattica  
digitale della filosofia

Zenodo

DOI: <https://doi.org/10.5281/zenodo.19956640>

Licenza Creative Commons\* BY-SA

Prima versione: maggio 2026

Sito web: <https://textusweb.netlify.app>

Visita il sito per verificare la presenza di una versione aggiornata.

Impaginato con LaTeX.

In copertina: Gardens by the Bay, Singapore. Foto di di Andrew Kow su Unsplash (particolare), url: <https://unsplash.com/it/foto/Ndxgo2Bn3WY>. Licenza Unsplash, url: <https://unsplash.com/it/licenza>.

# Indice

Presentazione. Per una didattica digitale conviviale	7
Uno strumento di base: il Markdown	11
I Argomentare	13
Dare forma al pensiero	14
Argomentare al tempo dei social network . . . . .	14
Persuadere e cercare la verità . . . . .	15
Oltre la logica formale . . . . .	17
La nuova retorica . . . . .	17
Il modello di Toulmin . . . . .	18
La logica informale . . . . .	20
Jürgen Habermas e l'etica del discorso . . . . .	21
L'ipotesi della cognizione estesa . . . . .	22
Ambienti di pensiero . . . . .	23
Mappare il pensiero . . . . .	24
Come si costruisce una mappa argomentativa . . . . .	25
Un esempio . . . . .	26
Osservazioni didattiche . . . . .	29
Usi delle mappe argomentative . . . . .	29
Kialo-Edu	31
Un social network per il pensiero critico . . . . .	31
Guida essenziale . . . . .	31
Kialo-Edu e il modello dell'argomentazione . . . . .	36
Kial-Edu in classe . . . . .	37

Mindmup	39
Cos'è Mindmup . . . . .	39
Creazione di una mappa . . . . .	39
Altre funzionalità . . . . .	43
Esportare, pubblicare e condividere le mappe . . . . .	45
Come usare Mindmup . . . . .	45
Argdown	46
Il Markdown al servizio dell'argomentazione . . . . .	46
Cosa ci serve per cominciare . . . . .	47
Le dichiarazioni . . . . .	47
Gli argomenti . . . . .	49
Le mappe grafiche . . . . .	51
Raggruppare gli elementi . . . . .	53
Come usare Argdown in classe . . . . .	55
Lexis	56
Uno strumento per l'analisi del testo . . . . .	56
Guida operativa . . . . .	57
Lexis in classe . . . . .	59
II Coltivare un giardino di pensieri	60
Connettere le idee	61
Imparare <i>davvero</i> . . . . .	62
Prendere appunti . . . . .	64
Il ruolo della filosofia . . . . .	64
Il metodo Zettelkasten . . . . .	65
Dallo Zettelkasten al Digital Garden . . . . .	67
Il Digital Garden in classe . . . . .	69
Vicus lurosum	74
Giocare con le perle . . . . .	74
Come funziona . . . . .	75
Usi didattici . . . . .	77
Supporto alla lezione . . . . .	77
Lavoro di gruppo . . . . .	78

Digital Garden personale . . . . .	78
TiddlyWiki . . . . .	79
Tutto in una pagina . . . . .	79
L'architettura di base . . . . .	79
Tre modi di usare TiddlyWiki . . . . .	80
TiddlyHost . . . . .	82
Prima configurazione . . . . .	83
Lavorare con i tiddler . . . . .	84
Navigare in TiddlyWiki . . . . .	86
Salvare il lavoro . . . . .	87
Organizzare il Digital Garden . . . . .	88
Usare TiddlyWiki in classe . . . . .	89
Logseq . . . . .	90
Obsidian e Logseq . . . . .	90
Caratteristiche di base di Logseq . . . . .	91
Esploriamo la versione demo . . . . .	92
Il menu . . . . .	95
Il grafo . . . . .	100
L'uso didattico di Logseq . . . . .	101
III Raccontare storie . . . . .	103
Dare forma al pensiero . . . . .	104
Il valore epistemico della narrazione . . . . .	104
Bruner e il pensiero narrativo . . . . .	105
Storie che pensano . . . . .	106
Una breve storia delle storie interattive . . . . .	107
Leggere e creare storie interattive . . . . .	109
Twine . . . . .	112
Salvare un bambino? . . . . .	112
Guida essenziale a Twine . . . . .	113
Ink . . . . .	121
Ink e Inky . . . . .	121

Guida essenziale a Inky . . . . .	121
Un esempio . . . . .	124
IV L'intelligenza artificiale	126
Il farmaco dell'IA	127
Perché questa sezione . . . . .	127
Le IA non sono conviviali . . . . .	127
Possibilità e rischi . . . . .	128
Simulazione di dialoghi filosofici . . . . .	130
L'IA per la creazione di immagini . . . . .	133
L'analisi del testo . . . . .	135
Individuare le fallace logiche . . . . .	139
Considerazioni finali (e provvisorie) . . . . .	139
Appendice	142
Guida alla sintassi Markdown	143
Glossario	148

# Presentazione

## Per una didattica digitale conviviale

Questo libro è nato nell'ambito dei laboratori legati all'insegnamento di *Esercitazioni di didattica della filosofia* presso l'Università di Siena. Presenta alcuni strumenti e pratiche digitali che è possibile utilizzare nell'insegnamento della filosofia; ha dunque un carattere eminentemente pratico.

Non sono tra quelli che ritengono che gli strumenti informatici siano la via maestra per fare scuola oggi, né tanto meno che siano indispensabili per la didattica della filosofia. Sono convinto che al centro del fare scuola – quale che sia la disciplina, ma soprattutto quando si fa filosofia – debbano esserci il confronto dialogico, l'uso vivo della parola, la discussione approfondita in un contesto relazionale quanto più possibile aperto<sup>1</sup>. E tuttavia, centrata sul dialogo, la scuola ha bisogno anche di esperienze, e alcune di queste oggi sono inevitabilmente legate all'informatica.

Qui presento, dunque, alcuni strumenti che consentono di fare cose interessanti in campo filosofico utilizzando l'informatica. A guidarmi nella scelta e conseguente proposta degli strumenti è un'idea precisa dell'informatica e del rapporto tra informatica e scuola, che vorrei illustrare brevemente.

Il mondo della scuola è destinatario di una certa quantità di prodotti e di strumenti digitali: i libri di testo, che nel loro formato elettronico sono fruibili solo attraverso la piattaforme proprietarie delle case editrici, e che consentono esperienze più o meno interattive; software e piattaforme con contenuti didattici o per l'elaborazione di testi, mappe concettuali, linee del tempo, esercizi interattivi eccetera; piattaforme per la gestione dell'apprendimento (Learning Management System) come Google Classroom o Microsoft Teams, che hanno ottenuto gran-

---

<sup>1</sup>Per la mia idea di scuola, rimando a A. Vigilante, *Senza cattedra. La scuola possibile*, Loescher, Torino 2026.

de successo durante la pandemia, quando la chiusura delle scuole ha costretto a ricorrere con urgenza a strumenti per la didattica online.

Questi prodotti e servizi hanno diversi aspetti in comune. Anche quando sono gratuiti per l'utente finale, sono finalizzati a generare profitto e abitano gli studenti fin da piccoli a muoversi in spazi digitali chiusi e proprietari. Anche quando si presentano come sistemi interattivi, che consentono agli utenti di creare alcuni prodotti propri (si pensi alla creazione di una presentazione con PowerPoint), sono di fatto scatole chiuse: l'utente che fosse in possesso delle competenze tecniche necessarie non potrebbe intervenire sul codice per modificarlo e creare una versione personalizzata, adatta alle proprie esigenze, del software.

Esiste un'informatica diversa, che potremmo definire *conviviale*, per usare una categoria di Ivan Illich. In *Tools for Conviviality* (1973) il filosofo distingueva in questo modo gli strumenti conviviali da quelli industriali:

Lo strumento conviviale è quello che mi lascia il più ampio spazio e il maggior potere di modificare il mondo secondo le mie intenzioni. Lo strumento industriale mi nega questo potere; di più: attraverso di esso, è un altro diverso da me che determina la mia domanda, restringe il mio margine di controllo e governa il mio senso della vita.<sup>2</sup>

Gli strumenti informatici conviviali sono, in primo luogo, open source<sup>\*</sup>: il codice è disponibile e consultabile da chiunque, ma soprattutto chiunque ne abbia le competenze può modificarlo, adattarlo alle proprie esigenze e ridistribuirne versioni migliorate. Uno strumento informatico conviviale stimola poi l'autonomia, la sperimentazione e la ricerca. Non richiede, per poterlo usare, corsi di formazione tenuti da esperti certificati. Questo non vuol dire che sia sempre semplice da usare; si tratta però di uno strumento dietro al quale c'è una comunità di sviluppatori che è anche una comunità di apprendimento: chi usa impara da chi ha programmato, chi impara contribuisce a migliorare, e il sapere tecnico si diffonde invece di concentrarsi nelle mani di pochi specialisti. È precisamente questo modello di trasmissione del sapere che rende

---

<sup>2</sup>I. Illich, *La convivialità*, Boroli, Milano 2005, p. 43.

gli strumenti open source\* interessanti non solo sul piano tecnico, ma anche su quello pedagogico.

Uno strumento conviviale, inoltre, non persegue scopi stabiliti da altri. Non raccoglie i dati dell'utente, non ha lo scopo di modificare i suoi comportamenti, né di tenerlo connesso il più a lungo possibile. È uno strumento che si fa semplicemente usare per lo scopo per il quale è stato progettato.

C'è infine una dimensione politico-economica che non si può trascurare. Gli strumenti digitali dominanti sono prodotti da grandi multinazionali tecnologiche che per operare richiedono infrastrutture globali, investimenti enormi e competenze altamente specializzate. Questo modello concentra il potere in poche mani, genera dipendenza sistemica e – come si vede con crescente chiarezza nel caso dell'intelligenza artificiale – produce ricadute sempre più decisive anche sul piano militare e geopolitico. La scelta di strumenti leggeri e con codice aperto ha dunque implicazioni etiche e politiche, e che come tale merita di essere non solo discussa e argomentata, ma anche preferita in un contesto didattico.

La pubblicazione di questo stesso manuale segue questi criteri. Il formato digitale e la pubblicazione su Zenodo, repository di risorse aperte del CERN di Ginevra, consentono di realizzare nuove edizioni aggiornate senza dipendere dalle esigenze anche di mercato di una casa editrice. La licenza Creative Commons\* BY-SA consente a chiunque di creare versioni derivate di quest'opera, con la sola condizione di dover inserire un riferimento all'opera originale e di distribuire la nuova opera con la medesima licenza.

Questo libro esce a ridosso della pubblicazione della bozza delle nuove Indicazioni Nazionali per i Licei<sup>3</sup>, che propone un cambiamento piuttosto significativo della didattica della filosofia, con la distinzione tra due modalità: la trattazione storica, tipica della didattica della filosofia nella scuola italiana dalla riforma Gentile in poi, e un approccio tematico, con "l'analisi di problematiche fondamentali della tradizione filosofica". Non è qui possibile approfondire; osservo solo che questa seconda modalità apre uno spazio per vivere la filosofia come una *pratica* per la quali gli strumenti qui proposti possono risultare utili. Nella bozza si

---

<sup>3</sup>Disponibile all'url: <https://www.mim.gov.it/-/pubblicato-il-testo-delle-nuove-indicazioni-nazionali-per-i-licei->

legge ancora che

.

Sia lo studio della filosofia, sia l'esercizio delle pratiche filosofiche permetteranno inoltre agli studenti di acquisire una consapevolezza critica nei confronti degli sviluppi tecnologici, in particolare dell'intelligenza artificiale, comprendendone, anche attraverso il confronto con le materie scientifico-tecnologiche, le potenzialità e i limiti, e promuovendone un uso responsabile ed eticamente orientato.

Il tema dell'intelligenza artificiale è ineludibile. Ad esso è dedicata l'ultima sezione di questo libro. Benché non soddisfi i criteri etici che considero irrinunciabili per la scelta degli strumenti da usare a scuola, non si può fare a meno di confrontarsi con essa, perché è ormai parte della vita quotidiana di docenti e studentesse<sup>4</sup>, e con ogni evidenza lo sarà sempre più.

Siena, 1 maggio 2026

---

<sup>4</sup>Nel libro alternerò il maschile e il femminile sovraestesi.

## Uno strumento di base: il Markdown

I nostri studenti praticano la scrittura soprattutto attraverso la tastiera degli smartphone, che sono di fatto piccoli computer dalle potenzialità enormi, sfruttate però solo in minima parte. La scrittura passa per lo più attraverso le app di messaggistica, che hanno un ruolo centrale nella vita degli adolescenti e tutt'altro che periferico in quella degli adulti. I più smaliziati sanno già che aggiungendo su WhatsApp un asterisco prima e dopo una parola la si ottiene in grassetto, che un trattino basso la mette in corsivo, che una tilde la barra. In sostanza, trasformano le parole attraverso semplici segni di testo, e senza saperlo stanno già usando il principio del Markdown.

Il Markdown è un linguaggio di marcatura leggero che fa esattamente questo: ottenere certi risultati concentrandosi solo sul testo. Tecnicamente si tratta di un insieme coerente di segni che servono a dire a una macchina che cos'è quella parola o quella frase (un titolo, un corsivo, un elenco, una citazione) ma in fase di scrittura ci si concentra solo sul contenuto. La forma finale arriva in seguito, in fase di conversione: lo stesso file Markdown può diventare un PDF, una pagina web o un documento Word, a seconda delle esigenze. È il contrario di quello che accade con un word processor\*, dove si crea direttamente la pagina finale e il contenuto è inseparabile dalla sua veste grafica.

Questo ha conseguenze concrete sulla qualità della scrittura. Quando non si deve pensare ai margini, al carattere o all'interlinea, il testo scorre più aderente al pensiero. Ma c'è un vantaggio meno ovvio, e forse più importante in un contesto didattico: scrivere in Markdown obbliga a pensare la struttura del testo – titoli, paragrafi, sottoparagrafi, citazioni – prima ancora di preoccuparsi di come apparirà. È un'abitudine mentale preziosa, che insegna a distinguere l'architettura del ragionamento dalla sua presentazione.

Un file Markdown è, nella sostanza, un semplice file di testo, anche se l'estensione del file è .md e non .txt. Può essere aperto con qualsiasi editor, su qualsiasi sistema operativo, oggi come tra trent'anni, senza

dipendere da nessuna azienda né da nessuna licenza. Chi ha ritrovato dopo anni un documento scritto con un vecchio word processor\*, illeggibile perché il software non esiste più o la versione è incompatibile, comprende immediatamente il valore di questa caratteristica. Il Markdown è, per riprendere la categoria con cui abbiamo aperto questo capitolo, uno strumento conviviale: trasparente, portatile e controllabile interamente da chi lo usa.

Per chi non lo conosce, imparare Markdown richiede al massimo una mezz'ora. In Appendice si trova una guida essenziale alla sintassi e ai principali software per scrivere in Markdown. L'invito è a provarlo direttamente, perché è uno di quegli strumenti che si capiscono meglio usandoli che leggendone la descrizione.

# **I Argomentare**

# Dare forma al pensiero

## Argomentare al tempo dei social network

Le persone argomentano di continuo. Lo fanno al bar, in famiglia, nelle assemblee condominiali e sempre più spesso sui social network. Raramente, tuttavia, lo fanno in modo rigoroso, esaminando gli argomenti e cercando onestamente la verità o qualcosa che non si allontani troppo da essa. La psicologia cognitiva degli ultimi trent'anni non è rassicurante da questo punto di vista, mostrando come il ragionamento comune sia raramente rigoroso. Procediamo per intuizioni, per analogia, per coerenza narrativa, per identità di gruppo (le nostre posizioni di base: essere di destra o di sinistra, credenti o non credenti, e così via). Jonathan Haidt<sup>1</sup> ha mostrato il carattere *tribale* del modo di funzionare comune della nostra mente. Le nostre conclusioni, condizionate dalle emozioni e dal senso di appartenenza, arrivano prima del ragionamento, che subentra in seguito, chiamato a difendere una tesi già scelta. Nella vita quotidiana il ragionamento non serve dunque tanto a cercare la verità – o a cercare una posizione onesta riguardo a un tema rilevante, come ad esempio l'eutanasia o la condanna o meno di un genocidio in atto – quanto a giustificare una posizione adottata in modo emotivo e irriflesso.

L'avvento dei social network ha peggiorato la situazione in modo strutturale, per il modo stesso in cui le piattaforme sono progettate. I social network sono ottimizzati per il coinvolgimento emotivo, gli algoritmi premiano i contenuti che suscitano reazioni forti – in particolare indignazione e paura – perché queste emozioni tengono gli utenti incollati allo schermo più a lungo. Il risultato è una sistematica amplificazione dei contenuti polarizzanti: le posizioni estreme vengono rilanciate più delle posizioni moderate, le affermazioni semplici e nette

---

<sup>1</sup>J. Haidt, *Menti tribali: perché le brave persone si dividono su politica e religione*, Codice, Torino 2013.

più delle analisi sfumate, gli attacchi personali più degli argomenti seri. Un'argomentazione lunga e rigorosa può essere liquidata, su un social network, con faccine di scherno, insulti o attacchi personali; non v'è un obbligo condiviso, una sorta di etica di base della comunicazione, che implichi il ricorso all'argomentazione.

Il discorso pubblico finisce così per essere ostaggio di un modello di business che cattura l'attenzione attraverso il conflitto, raccogliendo intanto e monetizzando i dati degli utenti, che sono la principale fonte di ricchezza nell'ambito del *capitalismo della sorveglianza*<sup>2</sup>. Mentre il confronto diventa sempre più uno scontro tra posizioni rigide, sui social network si diffondono anche notizie infondate e visioni complottistiche o allarmistiche, che inquinano gravemente qualsiasi dibattito.

Le ricadute sulla democrazia sono concrete e ben evidenti. Una democrazia funziona quando i cittadini sono in grado di deliberare: di valutare argomenti, di cambiare idea in presenza di buone ragioni, di riconoscere la legittimità del disaccordo. Queste capacità richiedono una cultura dell'argomentazione che i social network erodono sistematicamente. In un ambiente in cui vincere un dibattito equivale a ottenere più like e in cui l'identità di gruppo si consolida attraverso la demonizzazione dell'altro, il confronto razionale diventa difficile, e la democrazia deliberativa che pensatori come John Rawls e Jürgen Habermas avevano immaginato come ideale regolativo si fa sempre più lontana.

Insegnare ad argomentare in modo rigoroso o almeno intellettualmente onesto non è, in questo contesto e in questo passaggio storico, una competenza scolastica astratta, ma un atto politico e di educazione civica. Vuol dire contribuire a formare cittadini capaci di partecipare al discorso pubblico con maturità, senso critico e rispetto di chi ha una visione diversa.

## **Persuadere e cercare la verità**

Il momento decisivo per la formazione dell'identità della filosofia occidentale è segnato dalla tensione tra argomentazione come persuasione

---

<sup>2</sup>S. Zuboff, *Il capitalismo della sorveglianza. Il futuro dell'umanità nell'era dei nuovi poteri*, Luiss University Press, Roma 2019

e argomentazione come ricerca della verità. Da un lato i Sofisti – Protagora, Gorgia, Trasimaco – che sono maestri del discorso che persuade, che porta il destinatario dove vuole grazie alla forza della parola (si pensi all'*Encomio di Elena* di Gorgia), dall'altra la ricerca socratica della verità, il discorso maieutico che distrugge le certezze dell'altro non per condurlo dove si desidera, ma per consentirgli di far emergere la verità che egli stesso ha in sé.

La filosofia occidentale si è costruita, da Socrate in poi, come alternativa anche etica alla retorica sofistica. Il filosofo cerca il vero e il bene, non il vantaggio immediato che viene dall'uso abile della parola. Una medesima tensione si trova nel pensiero indiano. Nella scuola Nyaya la conoscenza vera, ottenuta attraverso un uso rigoroso della logica, è l'unica che può condurre alla liberazione; e a livello di confronto dialettico – nell'India classica le dispute tra scuole di pensiero diverse erano occasioni pubbliche importanti – distingue la discussione autentica (*vada*), che fa ricorso ad argomenti per sostenere una tesi, da *jalpa*, che si potrebbe tradurre con *sofisma*, la pratica di chi cerca di assicurarsi la vittoria in una discussione ricorrendo a fallacie e stragemmi argomentativi, e da *vitanda*, il cavillare che cerca di squalificare le tesi dell'avversario senza nemmeno discuterle<sup>3</sup>

La logica aristotelica, centrata sul sillogismo e rielaborata e codificata in modo raffinatissimo dalla filosofia medioevale, è stata per secoli la base del pensiero occidentale. La filosofia moderna, da Leibniz in poi, ha cominciato a sognare una *mathesis universalis*, un linguaggio formale capace di risolvere qualsiasi questione attraverso il calcolo. Con Frege, Russell, e Whitehead, all'inizio del Novecento questo sogno ha trovato la sua realizzazione più ambiziosa: la logica matematica, o logica formale, un sistema simbolico di straordinaria potenza e precisione.

La logica formale contemporanea è alla base dell'informatica, della linguistica formale, della filosofia analitica del linguaggio, della matematica dei fondamenti. Ma è anche, considerata dal punto di vista didattico, uno strumento che presenta un problema serio: la sua astrazione la rende quasi inaccessibile per chiunque non abbia una formazione specifica.

Chi insegna filosofia si trova di fronte a due possibilità. La prima

---

<sup>3</sup>L.V. Arena, *Il Nyaya Sutra di Gautama*, Edizioni Asram Vidya, s.l. 1994, Libro I, cap. II, pp. 97 segg.

è di formare alla logica così come è stata elaborata dal pensiero occidentale, lavorando sul sillogismo o spingendosi fino alla logica formale contemporanea. Il rischio di questa impostazione è che la logica appaia come uno strumento astratto, formale, distante dalla concretezza spesso violenta delle discussioni quotidiane. L'alternativa può essere quella di muovere da un modello meno rigoroso, ma più aderente alla realtà dell'argomentazione comune.

## Oltre la logica formale

Nel corso del Novecento si sono sviluppati diversi approcci che concordano nel ritenere che l'argomentazione reale, sia nella vita quotidiana che in campi come la filosofia, il diritto, la scienza e la politica, non si lasci ridurre alle strutture della logica formale, né possa essere semplicemente considerata una forma degenerata di argomentazione rigorosamente logica<sup>4</sup>.

### *La nuova retorica*

Uno dei testi fondamentali in questo ambito è il *Traité de l'argumentation. La nouvelle rhétorique* del filosofo Chaïm Perelman e della sociologa Lucie Olbrechts-Tyteca, pubblicato nel 1958<sup>5</sup>. Perelman e Olbrechts-Tyteca partono dalla constatazione che nei campi che abbiamo citato si usano per lo più argomenti che non sono né dimostrativi (come quelli della logica formale) né puramente retorici nel senso manipolativo del termine; sono argomenti che mirano all'adesione razionale di un uditorio, che non è né coercizione logica – essere costretti ad accettare una conclusione perché la logica la impone – né alla manipolazione emotiva, ma alla persuasione ragionevole.

Perelman e Olbrechts-Tyteca recuperano la categoria aristotelica di dialettica – l'argomentazione probabile, rivolta a un uditorio – trasformandola in una teoria sistematica. Il loro concetto chiave è quello di *uditorio universale*: un argomento è veramente valido quando potrebbe

---

<sup>4</sup>Per una panoramica si veda P. Cantù, I. Testa, *Teorie dell'argomentazione. Un'introduzione alle logiche del dialogo*, Bruno Mondadori, Milano 2006.

<sup>5</sup>Edizione italiana: Ch. Perelman, L. Olbrechts-Tyteca, *Trattato dell'argomentazione. La nuova retorica*, Einaudi, Torino 1966.

convincere qualsiasi persona ragionevole, non solo l'interlocutore contingente. Questa nozione permette di distinguere la propaganda (rivolta a un uditorio particolare e ai suoi pregiudizi) dalla vera argomentazione (rivolta all'uditorio universale della ragione).

Perelman e Olbrechts-Tyteca non forniscono una lista chiusa di criteri, ma dal *Trattato* si ricavano alcune condizioni che un'argomentazione deve soddisfare per aspirare all'uditorio universale. Le premesse non devono fare affidamento su ciò che solo un gruppo particolare condivide (pregiudizi culturali, interessi di parte, credenze locali) ma devono essere riconoscibili come ragionevoli da chiunque ragioni in buona fede. L'argomento deve essere internamente coerente: ciò che si afferma in un caso deve valere in tutti i casi analoghi e i principi cui si ricorre non possono essere applicati selettivamente. Deve inoltre rivolgersi all'interlocutore come essere razionale, senza fare leva sulle sue debolezze psicologiche, sulle emozioni contingenti o sulle pressioni sociali. La conclusione, infine, deve poter essere accettata non solo da chi ascolta in quel momento, ma da qualsiasi persona ragionevole in qualsiasi contesto.

Non sfuggirà il carattere problematico di questo uditorio universale, che non è un dato empirico ma una costruzione dell'oratore: chi argomenta decide implicitamente chi conta come "persona ragionevole", e questa decisione non è neutrale. La sfera del pubblico ragionevole può essere costruita a partire dal proprio orizzonte sociale, culturale, etnico e di genere, finendo per escludere chiunque abbia criteri di lettura della realtà non conformistici o estranei alla propria visione del mondo.

### *Il modello di Toulmin*

Nello stesso anno dell'opera di Perelman e Olbrechts-Tyteca, il filosofo britannico Stephen Toulmin pubblica *The Uses of Argument*<sup>6</sup>, un'opera destinata a diventare uno dei testi più citati nella storia dell'argomentazione. Anche Toulmin muove dalla preoccupazione relativa al modo in cui funzionano realmente gli argomenti in campi come il diritto, l'etica e la critica letteraria. Nella sua analisi un argomento reale ha almeno sei componenti: *claim* (la tesi che si vuole sostenere), *data* (le prove

---

<sup>6</sup>Edizione italiana: S. Toulmin, *Gli usi dell'argomentazione*, Rosenberg & Sellier, Torino 1975.

o i fatti su cui ci si basa), *warrant* (la regola o il principio che autorizza il passaggio dai dati alla tesi), *backing* (il fondamento che sostiene il *warrant*) e *qualifier* (il grado di certezza con cui si sostiene la tesi), e un *rebuttal* (le condizioni di eccezione o le obiezioni che potrebbero indebolire l'argomento).

Vediamo questi componenti all'azione in un argomento reale. Consideriamo una discussione sui limiti di velocità in autostrade e analizziamo questa affermazione: "Il limite di velocità sulle autostrade italiane dovrebbe essere abbassato a 110 km/h, perché ridurrebbe i morti sulle strade".

*Claim*: Il limite di velocità sulle autostrade italiane dovrebbe essere abbassato da 130 a 110 km/h.

*Data*: Gli studi dell'Organizzazione Mondiale della Sanità mostrano una correlazione significativa tra velocità media e mortalità negli incidenti stradali; i Paesi europei con limiti più bassi registrano tassi di mortalità autostradale inferiori alla media.

*Warrant*: Se una misura legislativa è causalmente efficace nel ridurre la mortalità e non produce danni comparabili, lo Stato ha l'obbligo di adottarla.

*Backing*: Il principio che la tutela della vita umana costituisce un obiettivo prioritario della legislazione, che giustifica limitazioni delle libertà individuali quando i benefici collettivi sono dimostrabili (argomento di tipo consequenzialista, implicito nella maggior parte delle politiche sanitarie e di sicurezza).

*Qualifier*: Probabilmente / stando ai dati disponibili (la correlazione è robusta ma non deterministica: altri fattori incidono sulla mortalità autostradale).

*Rebuttal*: A meno che i costi economici e sociali della misura — tempi di percorrenza, impatto sul trasporto merci, effetti sull'industria automobilistica — non superino i benefici in termini di vite salvate; oppure a meno che misure alternative (controlli sull'alcol, manutenzione stradale, formazione dei conducenti) non siano ugualmente efficaci a costi inferiori.

Tra questi, il *warrant* è il componente teoricamente più interessante: è la regola implicita che permette il salto dai dati alla conclusione, e viene raramente enunciata esplicitamente nell'argomentazione ordinaria. Renderla esplicita è uno degli esercizi analitici più importanti,

anche sul piano didattico, che il modello suggerisce: molto spesso il disaccordo tra due persone non riguarda i dati né la conclusione, ma proprio il *warrant* implicito che ciascuno usa senza saperlo. Due persone che discutono di pena di morte possono condividere gli stessi dati statistici sulla recidiva e arrivare a conclusioni opposte perché adottano *warrant* diversi: uno consequenzialista, l'altro deontologico.

Toulmin inoltre insiste sul fatto che un argomento non è una struttura statica da valutare in astratto ma un processo che si svolge di fronte a un interlocutore reale che può sollevare obiezioni. Il *rebuttal* non è pertanto una componente accessoria, ma il segno che chi argomenta ha preso sul serio la possibilità di essere confutato ed è interessato a una discussione intellettualmente onesta.

In campo didattico il modello ha vantaggi evidenti, perché si può insegnare in poche ore e applicare subito all'analisi di testi reali. Il suo limite principale è che descrive bene la struttura di un argomento ma dice poco sulla valutazione della sua forza (sapere che un argomento ha un *warrant* non dice ancora se quel *warrant* è buono). È su questo punto che la logica informale di Walton e Johnson può essere vista come un completamento.

## *La logica informale*

A partire dagli anni Settanta, un gruppo di logici e filosofi nordamericani, tra cui Ralph Johnson e John Anthony Blair dell'Università di Windsor, e Douglas Walton dell'Università di Winnipeg, sviluppano quello che è oggi noto come movimento della *logica informale* (*informal logic*).

La premessa del movimento è, anche in questo caso, l'osservazione che la logica formale non appare in grado di analizzare adeguatamente le fallacie argomentative che si incontrano nel ragionamento ordinario. Un argomento può essere formalmente valido ed essere comunque fallace in senso pratico, perché sfrutta pregiudizi cognitivi, o perché ignora prove rilevanti, o perché ancora confonde il peso dell'evidenza con la sua mera presenza. La logica informale studia dunque gli argomenti nel loro contesto naturale (il dibattito politico, il giornalismo, la conversazione quotidiana, ma anche il testo filosofico) e si chiede non solo se la conclusione segue formalmente dalle premesse, ma se le premesse sono accettabili, se le prove sono sufficienti, se il ragio-

namento è onesto. Introduce così criteri di valutazione che la logica formale non contempla: la rilevanza (le prove portate sono pertinenti alla tesi?), la sufficienza (sono abbastanza forti da sostenerla?), l'accettabilità (le premesse sono ragionevoli per l'interlocutore?). Questi tre criteri – rilevanza, sufficienza, accettabilità – sono il nucleo normativo che Johnson e Blair elaborano in *Logical Self-Defense*<sup>7</sup>, e che diventa il riferimento del movimento.

Douglas Walton<sup>8</sup> elabora una teoria sofisticata dei tipi di dialogo: il dialogo persuasivo (in cui si difende una tesi), la negoziazione, l'inchiesta, il debate, l'eristica. Ogni tipo di dialogo ha le proprie regole implicite, e una mossa che è legittima in un contesto può essere fallace in un altro. Questa attenzione ai contesti dialogici rende la logica informale particolarmente utile per la didattica: insegnare ad argomentare significa anche insegnare a riconoscere il tipo di dialogo in cui ci si trova.

### *Jürgen Habermas e l'etica del discorso*

Una prospettiva radicalmente diversa, ma complementare, viene dalla filosofia sociale di Jürgen Habermas. Nella sua *Theorie des kommunikativen Handelns*<sup>9</sup> del 1981 e nei lavori successivi sull'*etica del discorso*, Habermas ha sviluppato una teoria normativa dell'argomentazione che considera non solo come funziona l'argomentazione reale, ma come dovrebbe funzionare in una comunità di parlanti razionali. La nozione centrale è quella di azione comunicativa: a differenza dell'azione strategica (che mira al successo personale, eventualmente a spese degli altri), l'azione comunicativa mira all'intesa razionale tra i partecipanti. Una vera argomentazione, nel senso di Habermas, è quella in cui i partecipanti si impegnano a usare solo la forza non coercitiva dell'argomento migliore, in condizioni di simmetria e di rispetto reciproco.

Da questa premessa Habermas deriva il principio del discorso: una norma è valida solo se potrebbe essere accettata da tutti i potenziali

---

<sup>7</sup>R.H. Johnson, J.A. Blair, *Logical Self-defense*, McGraw-Hill, Toronto, Canada 1977.

<sup>8</sup>D. Walton, *The New Dialectic: Conversational Contexts of Argument*, University of Toronto Press, Toronto, Canada 1998.

<sup>9</sup>Edizione italiana: J. Habermas, *Teoria dell'agire comunicativo*, Il Mulino, Bologna 1986.

interessati in un discorso pratico razionale. L'argomentazione, in questa visione, non è solo uno strumento cognitivo per trovare la verità, ma la struttura fondamentale della vita democratica. Una democrazia sana è una democrazia deliberativa, in cui le decisioni collettive emergono da processi di argomentazione pubblica libera e razionale.

Alla didattica della filosofia Habermas non offre particolari strumenti analitici, ma una sorta di cornice etica per l'insegnamento dell'argomentazione. Il suo pensiero consente di riflettere sull'importanza etica e politica del confronto dialogico e di formare una disposizione – quella di chi partecipa al discorso pubblico con l'intenzione genuina di cercare la verità insieme agli altri, non di vincere una competizione – che, come accennato, è fondamentale per la sopravvivenza della nostra democrazia.

## **L'ipotesi della cognizione estesa**

L'ultimo contributo teorico che merita attenzione in questa introduzione, pur non rientrando nelle teorie dell'argomentazione, viene dalla filosofia della mente. Nel 1998 Andy Clark e David Chalmers pubblicarono sulla rivista *Analysis* un articolo destinato ad un grande successo: "The Extended Mind"<sup>10</sup>. La loro tesi era che la mente non si limita al cervello, ma si estende agli strumenti e agli artefatti cognitivi con cui interagiamo.

L'esempio canonico è quello di un paziente con deficit di memoria che usa un taccuino per annotare le informazioni di cui ha bisogno. Clark e Chalmers sostengono che il taccuino di questo paziente svolge esattamente la stessa funzione della memoria biologica di un individuo senza deficit: in entrambi i casi, l'informazione è disponibile, può essere recuperata all'occorrenza, e guida il comportamento. Se siamo disposti a dire che la memoria nel secondo caso fa parte della mente, dovremmo essere disposti a dire lo stesso del taccuino nel caso del paziente.

Le implicazioni per l'argomentazione e per la didattica sono significative. Se la mente si estende agli strumenti, allora un diagramma argomentativo, come quelli che studieremo, non è solo una rappresentazione esterna di un pensiero che avviene nella testa, ma fa parte in modo integrante del processo cognitivo stesso. Costruire una mappa

---

<sup>10</sup>A. Clark, D. Chalmers, *The Extended Mind*, in *Analysis*, 58(1), 1998, pp. 7-19.

argomentativa non vuol dire tradurre un pensiero già formato in una forma visiva, ma farlo nascere *attraverso* la forma visiva. Lo strumento non è neutro rispetto al pensiero, ma lo modifica, lo struttura e lo rende possibile in modi che la sola elaborazione mentale interna non permetterebbe.

## Ambienti di pensiero

Questa costellazione di teorie ha alcune implicazioni didattiche convergenti che vale la pena enunciare esplicitamente, perché orientano le scelte presentate in questa sezione.

La prima implicazione è una presa di distanza dal formalismo logico astratto come punto di partenza dell'educazione argomentativa. Questo non significa che la logica formale non abbia valore, ma che usarla come porta d'ingresso all'argomentazione rischia di produrre l'effetto opposto a quello desiderato: studenti che memorizzano la forma dei sillogismi e non sanno costruire un argomento in una discussione reale.

Il punto di partenza adeguato è invece l'argomentazione concreta: testi filosofici, casi giuridici, dibattiti politici, questioni etiche reali, articoli di giornale su temi sensibili. Da questi si può risalire alle strutture, identificare le premesse implicite, riconoscere le fallacie, valutare la forza delle prove. All'astrazione si può giungere dopo, quale strumento di analisi.

La seconda implicazione riguarda l'obiettivo. Insegnare ad argomentare non significa insegnare a vincere i dibattiti (come avviene attraverso la pratica *analogica* del Debate), ma formare studenti capaci di costruire argomenti che rispettino tre qualità fondamentali: la chiarezza (sapere esattamente cosa si sostiene), l'onestà (non nascondere le obiezioni, non esagerare la forza delle prove), e la disponibilità a cambiare posizione in presenza di argomenti migliori.

La terza implicazione riguarda il ruolo degli strumenti digitali. Alla luce dell'ipotesi della cognizione estesa, gli strumenti presentati in questa sezione vanno intesi come ambienti di pensiero, più che come semplici ausili. Kialo-Edu struttura il dibattito come un albero di pro e contro, rendendo impossibile ignorare le obiezioni e favorendo una cultura della confutazione onesta. È lo strumento più adatto alla dimensione dialogica e pubblica dell'argomentazione, vicino allo spirito

della logica informale di Walton. Mindmap, flessibile e non vincolato a una logica argomentativa specifica, si presta alla fase esplorativa del pensiero: alla costruzione di mappe concettuali aperte. Meno facile da usare (e per questo didatticamente più significativo), Argdown permette di scrivere argomenti in un linguaggio semplice e di vederne automaticamente la struttura logica, ed è particolarmente utile per l'analisi di testi filosofici complessi: aiuta a ricostruire la mappa inferenziale di un argomento, a identificare i warrant impliciti nel senso di Toulmin, a distinguere le premesse dalla conclusione.

## Mappare il pensiero

Tutti questi strumenti hanno in comune l'aspetto visivo: costruiscono strutture grafiche che consentono una maggiore consapevolezza dello sviluppo del ragionamento e della connessione tra le idee. La *mappa argomentativa* è una rappresentazione diagrammatica della struttura logica di un testo o di un discorso, in cui le tesi sono nodi e le relazioni di supporto o attacco sono archi.

La genealogia delle mappe argomentative si snoda lungo tre filoni distinti che si intrecciano tra gli anni Settanta e gli anni Novanta. Il punto di partenza è la tradizione, che già conosciamo, della logica informale. Nel 1950 Monroe Beardsley aveva già proposto, nel suo *Practical Logic*,<sup>11</sup> un sistema di diagrammi per visualizzare la struttura degli argomenti. Il suo modello viene arricchito e perfezionato da altri autori, come James Freeman, che aggiunge cinque modelli di argomentazione di base<sup>12</sup>, ma è nel 1978 che Michael Scriven e Richard Paul, con il movimento del *critical thinking*, rendono il diagramma argomentativo uno strumento didattico sistematico.

Il secondo filone è informatico. Nel 1970 Horst Rittel e Werner Kunz introducono il modello *IBIS (Issue-Based Information System)*, concepito per gestire problemi complessi nei processi di design e pianificazione. La struttura IBIS si articola su tre tipi di nodo: *issue* (la domanda o il problema), *position* (una risposta possibile) e *argument* (un pro o un contro rispetto a quella posizione). È uno schema ternario,

---

<sup>11</sup>M. Beardsley, *Practical Logic*, Prentice-Hall, New York 1950.

<sup>12</sup>Cfr. P. Alotto, *Le mappe argomentative*, Dino Audino, Roma 2022, p. 116.

non binario, in cui la domanda precede logicamente la risposta e gli argomenti valutano le risposte.

Nel 1988 Jeff Conklin e Michael Begeman implementano IBIS in un software, *gIBIS*, il primo strumento informatico per la mappatura argomentativa. La struttura IBIS diventerà poi la base di software come Compendium e, più recentemente, di alcune funzionalità di Kialo-Edu.

Il terzo filone è esplicitamente pedagogico. Negli anni Novanta Tim van Gelder, filosofo australiano, sviluppa una proposta didattica fondata sull'idea che il modo più efficace per migliorare il pensiero critico degli studenti sia la pratica esplicita di *argument mapping*, la costruzione sistematica di diagrammi argomentativi su testi e problemi reali. Il suo software Reason!Able e la successiva piattaforma Rationale<sup>13</sup> sono il primo strumento pensato specificamente per l'insegnamento. Le ricerche successive di van Gelder e collaboratori mostrano miglioramenti significativi nelle capacità di pensiero critico degli studenti che usano regolarmente mappe argomentative rispetto a quelli che seguono un insegnamento tradizionale.

## Come si costruisce una mappa argomentativa

Prima di passare agli strumenti digitali è utile stabilire i principi metodologici della costruzione di una mappa argomentativa. Questi principi valgono indipendentemente dallo strumento usato e sono validi anche per una mappa disegnata con carta e matita. Anzi, è consigliabile che la prima esperienza degli studenti con le mappe avvenga in forma analogica, per separare l'apprendimento del metodo dall'apprendimento dello strumento.

1. Identificare il *claim*, la tesi principale che si vuole valutare o difendere. Identificarla con precisione è il passaggio più difficile e più importante. Una tesi vaga produce una mappa vaga. La tesi deve essere una proposizione assertiva, valutabile come vera o falsa, non una domanda e non un fatto indiscutibile.

---

<sup>13</sup>Url: <https://rationaleonline.com>. La piattaforma è un ottimo strumento, molto apprezzato ad esempio da Piero Alotto, autore del libro citato nella nota precedente. Non è trattata in questo libro perché richiede una licenza a pagamento e non è open source.

2. Distinguere tesi, argomenti e prove. Una tesi afferma qualcosa; un argomento dà una ragione per accettarla o rifiutarla; una prova è un dato di fatto, un esempio, una citazione che supporta l'argomento. Confondere i tre livelli è l'errore più frequente nei principianti: inserire prove dove dovrebbero stare argomenti, o argomenti al posto della tesi.
3. Individuare i *warrant*. Nella gran parte delle argomentazioni la connessione tra le premesse e la conclusione non è dichiarata esplicitamente ma viene data per scontata. Renderla esplicita è uno degli esercizi filosoficamente più formativi della mappa argomentativa. *Perché questo dato supporta quella conclusione? Quale principio generale sta operando in sottofondo?*
4. Distinguere argomenti indipendenti da argomenti convergenti. Due argomenti sono *indipendenti* quando ciascuno, da solo, sarebbe sufficiente a supportare la tesi (se uno cade, l'altro regge ancora). Sono *convergenti* quando funzionano insieme, come co-premesse di una conclusione (se uno cade, l'altro perde forza). La distinzione è importante perché cambia la struttura della mappa e la valutazione della solidità dell'argomento complessivo.
5. Cercare le obiezioni. Una mappa argomentativa non è completa se non include almeno un argomento contrario alla tesi principale. L'abitudine di cercare attivamente le obiezioni alla propria posizione è importante nell'educazione all'argomentazione filosofica, sia per l'importanza che in essa ha l'onestà intellettuale (non conta avere ragione con ogni mezzo, ma giungere a una conclusione corretta), sia perché la considerazione delle obiezioni possibili rende la propria argomentazione più solida.

## Un esempio

Per rendere concreto il metodo, applichiamolo a un brano del *Critone* di Platone: il passo in cui Socrate espone le ragioni per cui rifiuta di fuggire dalla prigione, nonostante Critone gli offra la possibilità di farlo. È uno

dei testi più accessibili e più densi dal punto di vista argomentativo del corpus platonico, perfetto per un primo esercizio in classe terza.

Il testo di riferimento è *Critone*, 49a-50d. Ecco una sintesi del ragionamento di Socrate<sup>14</sup>:

Non si deve mai commettere ingiustizia, nemmeno in risposta a un'ingiustizia subita. Andarmene senza il permesso della città significherebbe commettere ingiustizia nei confronti delle leggi. Le leggi mi hanno dato tutto: mi hanno fatto nascere, crescere, educare. Ho vissuto ad Atene per settant'anni senza mai andarmene, il che equivale a un tacito accordo di obbedirle. Se ora le tradisco quando mi tornano scomode, dimostro che il mio rispetto per la giustizia era solo convenienza. E un uomo che ha passato la vita a discutere di giustizia non può morire fuggendo dalla giustizia.

La tesi di Socrate è formulabile così: *Socrate non deve fuggire dalla prigione, anche se potrebbe farlo senza conseguenze pratiche*. È una tesi normativa (riguarda ciò che si deve fare), non descrittiva.

Dall'analisi del testo emergono quattro argomenti distinti, tutti convergenti verso la tesi principale:

Argomento 1: Il principio di non-reciprocità del male. Non si deve mai fare ingiustizia, nemmeno in risposta a un'ingiustizia subita. Fuggire sarebbe commettere ingiustizia verso le leggi atenesi (danneggiarle). Quindi non si deve fuggire.

Argomento 2: Il contratto implicito. Socrate ha vissuto settant'anni ad Atene, ha usufruito di tutti i benefici che le leggi offrono, non è mai emigrato. Con questo comportamento ha implicitamente accettato un patto di obbedienza. Violare il patto ora sarebbe disonesto.

Argomento 3: Il debito verso le leggi. Le leggi hanno reso possibile la sua nascita, educazione, vita intera. Non si ripaga il benefattore con il tradimento.

Argomento 4: La coerenza filosofica. Socrate ha insegnato per tutta la vita che la giustizia viene prima della vita. Fuggire per sopravvivere significherebbe smentire tutto ciò che ha insegnato.

---

<sup>14</sup>Per il passaggio completo si veda Platone, *Tutti gli scritti*, a cura di G. Reale, Bompiani, Milano 2000, pp. 59-60.

I warrant impliciti Gli argomenti 2 e 3 condividono un *warrant* implicito che vale la pena esplicitare: *chi accetta deliberatamente i benefici di un sistema ha l'obbligo morale di rispettarne le regole, anche quando non gli conviene*. Questo principio non è dichiarato esplicitamente nel testo: viene dato per scontato. Renderlo esplicito è già un esercizio filosofico.

L'argomento 4 si regge su un altro warrant: *la coerenza tra ciò che si insegna e ciò che si fa è una condizione necessaria dell'autorità morale*. Anche questo è un presupposto contestabile, e la sua contestazione produce immediatamente un argomento contro la tesi.

Struttura della mappa (rappresentazione testuale):

TESI: Socrate non deve fuggire dalla prigione

[PRO] A1: Non si deve mai rispondere all'ingiustizia con l'ingiustizia

[DATO] Fuggire vuol dire commettere ingiustizia verso le leggi

[PRO] A2: Socrate ha un contratto implicito con le leggi di Atene

[DATO] 70 anni vissuti ad Atene senza emigrare

[WARRANT implicito] Chi accetta i benefici di un sistema deve rispettarne le regole

[PRO] A3: Le leggi hanno un titolo di credito sulla vita di Socrate

[PRO] A4: La coerenza tra vita e morte è necessaria per chi ha insegnato la giustizia

[CONTRO] C1: L'obbedienza è dovuta alle leggi giuste, non a qualsiasi legge

[DATO] La condanna di Socrate era ingiusta (per ammissione dello stesso Socrate)

[CONTRO] C2: La coerenza con sé stessi non giustifica la morte se produce solo danno

## Osservazioni didattiche

Questo esempio mostra alcune caratteristiche pedagogicamente rilevanti della mappa argomentativa applicata a un testo filosofico.

Prima di tutto, la costruzione della mappa costringe a distinguere quello che il testo dice esplicitamente da quello che presuppone. Il *warrant* implicito dell'argomento 2, per esempio, non è presente nel testo: è il lettore che deve ricostruirlo.

In secondo luogo, la struttura pro/contro emerge naturalmente dall'analisi: le due obiezioni (C1 e C2) nascono da domande che la mappa stessa suscita. C1 nasce da una tensione interna all'argomento 1: *non si deve commettere ingiustizia, ma cosa succede se le leggi stesse sono ingiuste?*

Terzo: la differenza tra argomenti *indipendenti* e argomenti *convergenti* si vede qui con chiarezza. A1, A2, A3 e A4 sono tutti argomenti a favore della stessa tesi, ma funzionano in modo parzialmente indipendente: se A2 (il contratto implicito) fosse smontato, rimarrebbero ancora A1, A3 e A4. Questo rende la posizione di Socrate robusta, perché non dipende da un singolo argomento.

Dal punto di vista della docente la mappa offre l'opportunità di stimolare non solo la capacità di analisi, ma anche il pensiero critico della studentessa. Anziché chiedere genericamente perché Socrate non vuole fuggire, si può chiedere: "Quale argomento ti convince di più? Quale ti convince meno? Riesci a costruire un'obiezione a C1?".

## Usi delle mappe argomentative

L'esempio del *Critone* illustra uno dei tre modi in cui una mappa argomentativa può essere impiegata: l'analisi di un testo dato. Ma non è l'unico uso possibile, né necessariamente il più frequente nella pratica didattica. In questo caso si fa un uso *ermeneutico* della mappa, quale strumento per leggere e interpretare un testo filosofico, giuridico o argomentativo. Si parte da un testo scritto e si ricostruisce la struttura logica sottostante: si rendono espliciti i *warrant* impliciti, si distinguono premesse da conclusioni, si identificano le obiezioni che l'autore anticipa o ignora.

Il secondo uso è quello *dialogico*: la mappa come strumento per strutturare una discussione in corso, in tempo reale o in differita. Invece di ricostruire un testo già scritto, si mappa una disputa tra posizioni diverse (in una classe, in un consiglio comunale, in un dibattito pubblico). La mappa non descrive un pensiero già formato ma lo costruisce collettivamente, rendendolo visibile e costringendo i partecipanti ad argomentare invece di limitarsi ad affermare.

Il terzo uso è quello *euristico*: la mappa come strumento per mettere in ordine le proprie idee, prima ancora di doverle comunicare o difendere. Si tratta di un uso esplorativo: la mappa serve a capire cosa si pensa davvero su un problema, a identificare le proprie premesse implicite, a scoprire dove il ragionamento è solido e dove è lacunoso.

# Kialo-Edu

## Un social network per il pensiero critico

Kialo (<https://www.kialo.com>) è una piattaforma online creata nel 2017 de Errikos Pitsos, laureato alla London School of Economics and Political Science, che permette agli utenti di analizzare un'affermazione centrale (la tesi) e di sviluppare una rete di ragioni pro e contro, disposte in modo gerarchico e facilmente navigabile. Ogni ragione può essere supportata o confutata da altre ragioni, creando una struttura ad albero che aiuta a visualizzare la logica e le connessioni tra le idee. La piattaforma è pensata per mantenere il confronto ordinato, collaborativo e focalizzato sui contenuti, evitando le derive polemiche tipiche dei social network.

Kialo-Edu (<https://www.kialo-edu.com>) è la versione educativa di Kialo, sviluppata appositamente per l'uso in contesti scolastici e universitari. Offre agli insegnanti un ambiente sicuro e controllato dove gli studenti possono esercitarsi nel costruire e valutare ragioni, partecipare a discussioni strutturate, analizzare testi o prepararsi a dibattiti.

Kialo-Edu è una piattaforma proprietaria (non quindi open source\*) sviluppata da un'organizzazione non profit con una specifica missione educativa e un modello fortemente incentrato sulla privacy e sull'etica dei dati. L'organizzazione ha sede in Germania e rispetta rigorosamente il GDPR\*, oltre a essere conforme ad altre normative internazionali sulla privacy degli studenti.

## Guida essenziale

Dopo essersi registrati è possibile esplorare una discussione di esempio in inglese. Il tema è: "I giovani dovrebbero essere obbligati a svolgere

un anno di servizio nazionale." La discussione si presenta come nella figura 1.

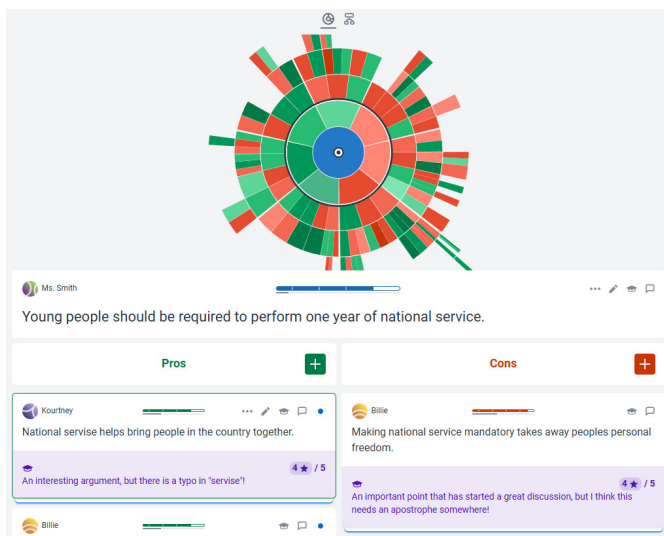


Figura 1

I cerchi concentrici raffigurano in modo sintetico ed efficace lo sviluppo della discussione: nel primo cerchio sono incluse le prime ragioni a favore (verdi) e contrarie (rosse), nel secondo cerchio le prime obiezioni e così via.

È possibile impostare una diversa visualizzazione abilitando la navigazione ramificata. Si ottiene la schermata della figura 2. Gli studenti, individuati con il loro nome (Kourtney, Billie, Hayes ecc.) hanno inserito le loro ragioni a favore o contro. Il docente è intervenuto, nell'area in basso. Clicchiamo sul primo commento a sinistra, quello di Kourtney. Si apre la finestra della figura 3. Il docente può commentare la ragione dello studente (in questo caso osservando che la ragione è interessante, ma notando un errore di battitura) e dare ad essa un voto (in questo caso 4 su 5).

Per restituire un feedback e assegnare un voto occorre cliccare sull'icona del cappello accademico, in alto a destra, ed entrare nella finestra della figura 4.

Ms. Smith

Young people should be required to perform one year of national service.

**Pros** **Cons**

**Kourtney**  
National service helps bring people in the country together.  
An interesting argument, but there is a typo in "service!" 4★ / 5

**Billie**  
Many countries around the world already require at least one year of national service.  
This is certainly true, so you should be able to find a reliable source for this claim. 4★ / 5

**Hayes**  
Mandatory national service makes a country safer.

**Billie**  
Making national service mandatory takes away peoples personal freedom.  
An important point that has started a great discussion, but I think this needs an apostrophe somewhere! 4★ / 5

**Alex**  
Mandatory national service is bad for people's mental and physical health. 4★ / 5

**Obl**  
Mandatory national service destroys jobs.

Figura 2

**Kourtney**

National service helps bring people in the country together.

4★ / 5

For **Kourtney** ([grading details](#)). Only visible to them ([change](#)).

**Grade claim:**

0 1 2 3 4 5

**Feedback for claim:**

An interesting argument, but there is a typo in "service!"

2942 Save

Figura 3

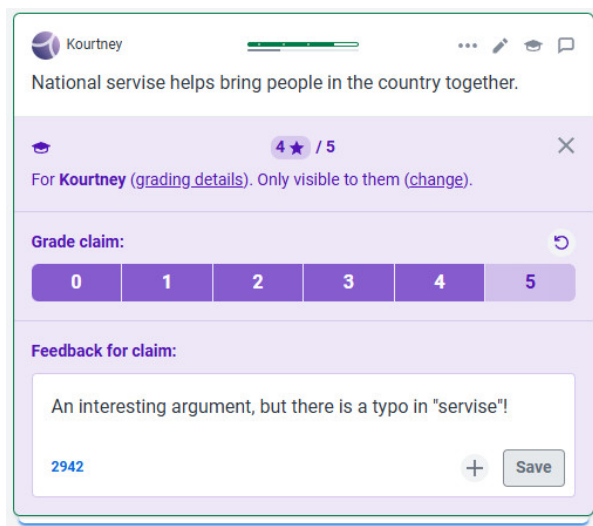


Figura 4

Cliccando sull'icona con la matita, accanto a quella con il cappello accademico, abbiamo la possibilità di editare il *claim*, in questo caso l'argomento dello studente. La piattaforma mostra anche le possibili ragioni duplicate (*duplicate claims*). Questa lista compare anche in fase di scrittura della ragione, e serve ad avvisare lo studente che quello che sta scrivendo è già stato detto da altri.

Se clicchiamo ancora sull'icona con il callout, a destra dell'icona con il cappello accademico, si apre la finestra della figura 5.

Abbiamo qui quattro icone. La prima a sinistra (ancora un callout) mostra i commenti e le diverse modifiche fatte nel tempo, mentre l'icona accanto consente nuovamente di visualizzare le ragioni duplicate. L'icona con la bandierina permette di contrassegnare la ragione come necessaria di revisione da parte dello studente, per una delle seguenti ragioni:

*Unsupported*: La ragione non è supportata (senza spiegazione, senza dati, senza esempi).

*Not a claim*: Quella dello studente non è una vera ragione.

*Unclear*: La ragione è poco chiara.

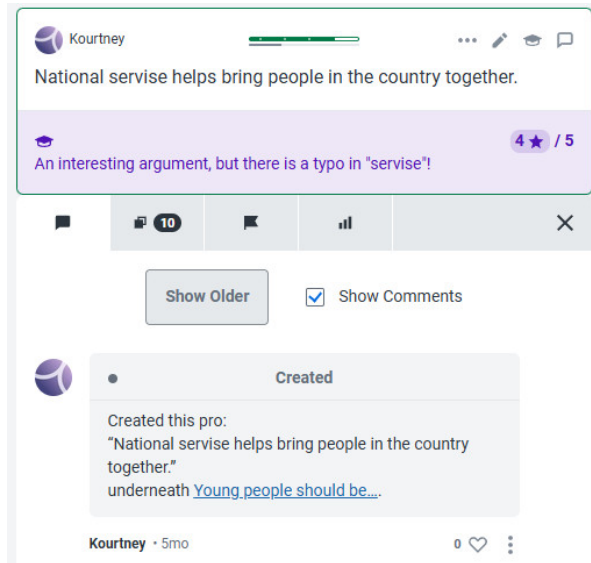


Figura 5

*Vulgar/Abusive*: La ragione è espressa in modo inaccettabile.

*Duplicate claim*: La ragione è uguale a una espressa già da qualche altro studente.

*Unrelates*: La ragione non è legata a ciò di cui si sta discutendo.

*Move elsewhere*: La ragione va spostata in un'altra parte della discussione. Le ragioni possono essere spostate semplicemente trascinandole nella nuova destinazione.

*More than one claim*: In ciò che ha scritto lo studente è presente più di una ragione.

L'ultima icona, con le tre linee graduate, ci consente di visualizzare l'esito del voto dei compagni, nel caso in cui questa funzione sia stata attivata.

## Kialo-Edu e il modello dell'argomentazione

Kialo-Edu segue una struttura binaria: ogni nodo è un pro o un contro rispetto alla tesi o all'argomento che lo precede. Questa semplicità è una scelta progettuale precisa, che riflette una specifica concezione dell'argomentazione: quella del confronto dialogico tra posizioni, in cui ogni mossa consiste nell'appoggiare o nell'attaccare ciò che l'interlocutore ha detto.

Chi ha letto l'introduzione a questa sezione riconoscerà che questa struttura è più vicina alla logica informale di Walton, con la sua attenzione ai tipi di dialogo e alle mosse legittime nel confronto tra posizioni, che al modello analitico di Toulmin. Il modello di Toulmin descrive la struttura interna di un singolo argomento: *claim, data, warrant, backing, qualifier, rebuttal*. Kialo-Edu non chiede allo studente di esplicitare il warrant, né di indicare il grado di certezza della tesi, né di distinguere tra argomenti indipendenti e convergenti, ma di prendere posizione e di motivarla, in modo che altri possano rispondere. Questa distinzione ha conseguenze didattiche dirette. Kialo è lo strumento giusto se l'obiettivo è allenare la dimensione pubblica e dialogica dell'argomentazione: imparare a sostenere una posizione di fronte a obiezioni reali, a riconoscere la forza degli argomenti avversari, a non ignorare le controobiezioni. È uno strumento meno adatto quando l'obiettivo è l'analisi strutturale di un testo filosofico complesso, dove conta rendere espliciti i presupposti impliciti e distinguere con precisione i livelli dell'argomento. Per quell'uso, come vedremo, è più indicato Argdown.

Un'ultima osservazione sul rapporto tra Kialo-Edu e il pensiero critico. La struttura ad albero dello strumento rende visibile qualcosa che nell'argomentazione orale o nei social rimane invisibile: la gerarchia delle ragioni. Ogni argomento su Kialo-Edu ha un padre – la tesi o l'argomento che intende supportare o attaccare – e può avere figli, cioè argomenti che lo supportano o attaccano a loro volta. Questa struttura costringe lo studente a chiedersi continuamente: sto attaccando la tesi principale o un argomento specifico? Sto aggiungendo una ragione nuova o sto ripetendo qualcosa che è già stato detto? Sono domande che nella discussione orale in classe raramente vengono poste in modo esplicito, e che Kialo-Edu rende invece strutturalmente necessarie.

Vediamo ora come usarlo in concreto, con qualche esempio.

## Kialo-Edu in classe

La prima volta che una classe usa Kialo-Edu è consigliabile che sia il docente a creare la discussione e a scegliere la tesi. Quest'ultima deve essere filosoficamente significativa ma immediatamente comprensibile, formulata in modo assertivo e genuinamente controverso, cioè tale che posizioni ragionevoli esistano su entrambi i lati. Il tema può essere ripreso dai filosofi studiati o essere legati all'attualità e alle questioni etiche e filosofiche che essa pone. In questo caso la scelta del tema richiede attenzione. Un tema troppo vicino all'esperienza personale degli studenti rischia di attivare dinamiche tribali: posizioni già formate, identità di gruppo in gioco, difficoltà a cambiare idea. Un tema troppo astratto non produce coinvolgimento genuino. La zona di equilibrio è quella delle questioni etiche reali ma non immediatamente identitarie: l'obbligo vaccinale, la pena di morte, il reddito universale di base, la leva militare obbligatoria, l'eutanasia, la responsabilità morale dell'intelligenza artificiale.

Prima di lavorare su Kialo-Edu, può essere utile dedicare del tempo in classe a costruire una mappa argomentativa analogica sulla lavagna: tesi al centro con alcuni pro e contro identificati collettivamente. Questo separa l'apprendimento del metodo dall'apprendimento dello strumento e fornisce agli studenti un punto di partenza quando accedono alla piattaforma.

La consegna individuale può essere semplice: ciascuno inserisce almeno due argomenti (uno pro e uno contro, oppure due dallo stesso lato con motivazioni diverse) e risponde ad almeno un argomento già presente. Il docente segue la discussione in tempo reale o in differita, usando la funzione di feedback per segnalare argomenti non supportati, poco chiari o duplicati, senza correggere il contenuto, ma orientando la qualità argomentativa.

La valutazione non riguarderà la posizione assunta, ma la qualità della motivazione. L'argomento è pertinente alla tesi? È supportato da qualcosa (un esempio, un riferimento al testo, un principio esplicitato)? Risponde davvero all'argomento che intende attaccare?

L'analisi di un testo filosofico è un uso di Kialo-Edu più avanzato e presuppone che la classe abbia già lavorato sull'analisi argomentativa di testi, ad esempio con l'esercizio sul *Critone* proposto nell'introduzione.

L'obiettivo non è discutere liberamente una tesi, ma ricostruire collettivamente la struttura argomentativa di un testo già letto. In questo caso il docente crea una discussione su Kialo-Edu la cui tesi coincide con la tesi principale del testo analizzato. Gli studenti, lavorando sul testo, inseriscono come argomenti pro le ragioni che l'autore porta a sostegno della sua tesi, e come argomenti contro le obiezioni che l'autore anticipa o che emergono dall'analisi critica. Le prove testuali, come citazioni e riferimenti a passi specifici, vanno inserite come motivazione degli argomenti, non come argomenti autonomi.

Questo uso ha un effetto che la lettura individuale non produce: rendere visibile il disaccordo interpretativo. Due studenti che hanno letto lo stesso testo possono formulare lo stesso argomento in modo diverso, o collocarlo in punti diversi dell'albero. Ne emerge una discussione – perché questo è un pro e non un contro? questo argomento attacca la tesi o un argomento specifico? – che ha un autentico carattere filosofico.

# Mindmup

## Cos'è Mindmup

Mindmup è uno strumento online per la creazione di mappe mentali, sviluppato dal programmatore serbo Gojko Adzic. Sebbene in passato (nella versione 1) il codice fosse disponibile con una licenza di tipo MIT, il progetto è stato successivamente cambiato. Attualmente, il software di Mindmup non è open source\*; la piattaforma offre la possibilità di creare fino a cinque mappe nella versione gratuita. Mindmup dichiara esplicitamente di applicare standard di protezione dei dati allineati ai principi del GDPR\* dell'Unione Europea. Gli abbonamenti consentono all'azienda di non basarsi sulla pubblicità o sul tracciamento degli utenti per sostenere il servizio e di non condividere informazioni di identificazione personale con terze parti (se non per finalità strettamente necessarie come l'autenticazione tramite Google o Microsoft, se preferita dall'utente).

La piattaforma dunque soddisfa solo parzialmente i criteri di questo manuale; ne parlo comunque perché si tratta di uno strumento che ha una curva di apprendimento molto bassa – è in lingua inglese, ma risulta essere intuitiva anche per chi non conosca la lingua – e può per così dire fare da ponte verso uno strumento più complesso come Argdown.

## Creazione di una mappa

Entrati nel sito (<https://www.mindmup.com>), ci viene offerta la possibilità di esplorare la piattaforma senza registrazione, selezionando *No sign-up. Go*. Accediamo ad una schermata con al centro un box nero con la scritta “Hit Space or double-click”. Seguendo l'invito clicchiamo due volte e il testo diventa editabile. Qui possiamo inserire la tesi, affermazione o posizione che vogliamo discutere; clicchiamo fuori dal

box per terminare la modifica. Se selezioniamo il box cliccandovi su, si apre un menu che ci consente di editarlo: possiamo modificare il colore del box, il tipo di carattere, il colore e lo stile del testo (bold, italic ecc.); possiamo anche inserire liste, link e note, mentre la possibilità di allegare file o immagini è riservata agli utenti premium. Interessante è la possibilità di aggiungere una *sticky note*, che compare come una sorta di post-it che possiamo usare per aggiungere osservazioni al nodo che abbiamo inserito.

Ora spostiamoci in alto a sinistra. Troviamo un menu con tre voci. Da sinistra: *Main menu*, sul quale torneremo; il nome della mappa, che attualmente è *Tesi*, perché è il testo che abbiamo inserito nel box, ma che possiamo editare cliccandolo; *Customize layout and style*. Cliccando su quest'ultima voce si apre il sottomenu della figura 1.

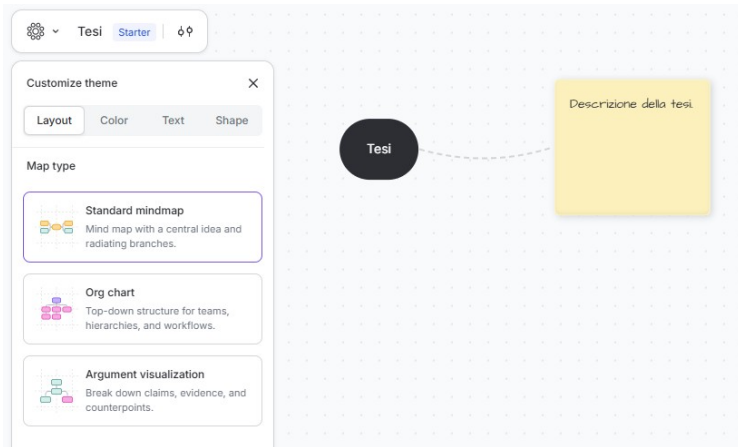


Figura 1

Abbiamo qui diverse possibilità di personalizzare la mappa: possiamo cambiare colori (*Color*), testo (*Text*) e forme (*Shape*), modificando lo stile dei box e delle linee. Ma a noi interessa la voce *Layout*, che ci consente di entrare nella modalità *Argument visualization*.

Subito dopo averla selezionata, notiamo che compare un in alto a destra del box: 1.1. Di default, Mindmap assegna dei numeri ai box per ordinarli gerarchicamente: 1.1 è la tesi, 2.1, 2.2 eccetera sono gli argomenti, 3.1, 3.2 i sottoargomenti e così via. Per avere

una mappa più pulita, si può eliminare questa numerazione da Main menu>View>Labels>None.

Se ora clicchiamo sul box con la tesi vediamo che è disponibile a destra il menu *Actions*, che ci consente di aggiungere argomenti a favore (*Add reason*) e obiezioni (*Add objection*), oltre a poter aggiungere una tesi subordinata (*Add child claim*) o sullo stesso livello (*Add sibling claim*) (figura 2).

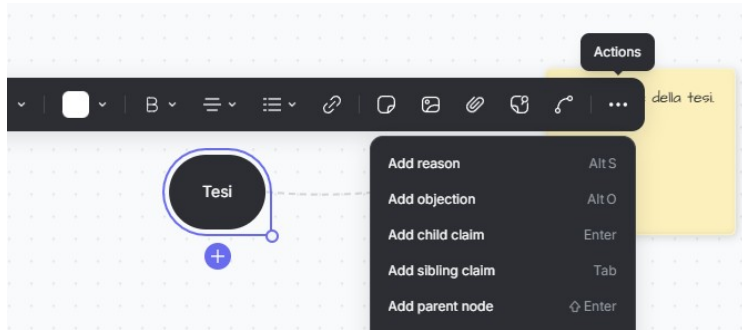


Figura 2

Aggiungiamo quindi il primo argomento a favore della tesi e la prima obiezione. Di default le *reasons* appaiono di colore verde e le *objections* di colore rosso (figura 3).

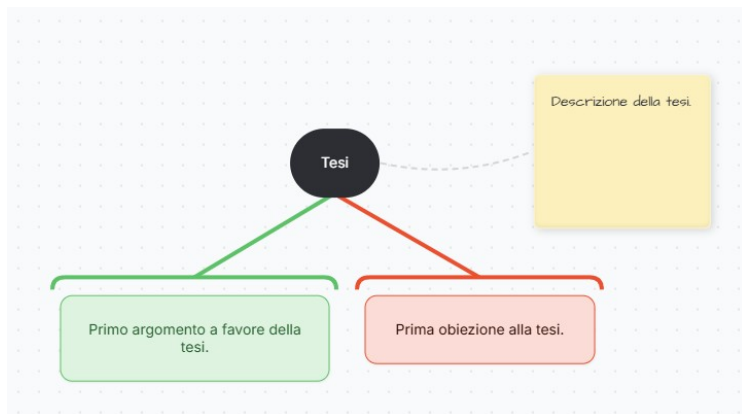


Figura 3

Selezionando i due box possiamo usare la stessa voce di menu *Actions* per aggiungere ancora argomenti a favore e obiezioni. Vediamo però dalla figura 4 che c'è un problema: sia l'obiezione alla tesi che l'obiezione all'obiezione sono di colore rosso; sembra dunque, visivamente, che tutto il ramo destro della mappa sia un'obiezione alla tesi.

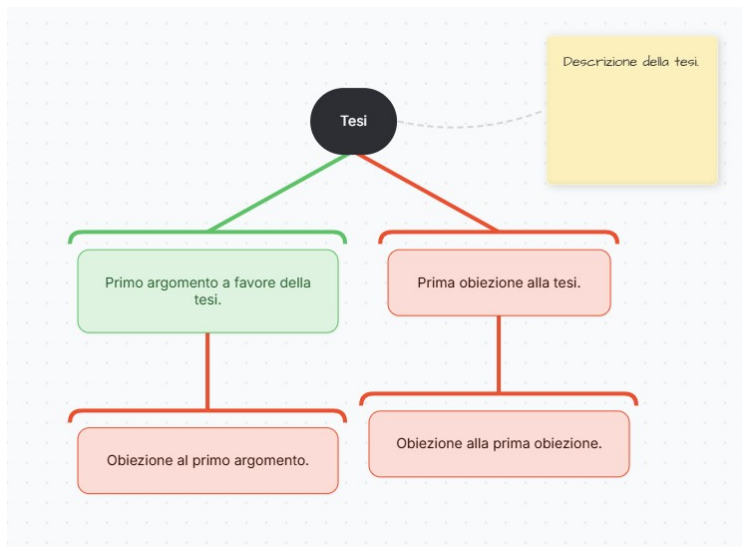


Figura 4

Per rendere la mappa più leggibile possiamo stabilire di considerare *reason* tutti gli argomenti di chi è favore della tesi centrale (anche quando si tratta di risposte ad obiezioni) e *objections* tutti gli argomenti contrari, anche quando si tratta di ragioni a sostegno di una obiezione. Una mappa così impostata si presenta come nella figura 5.

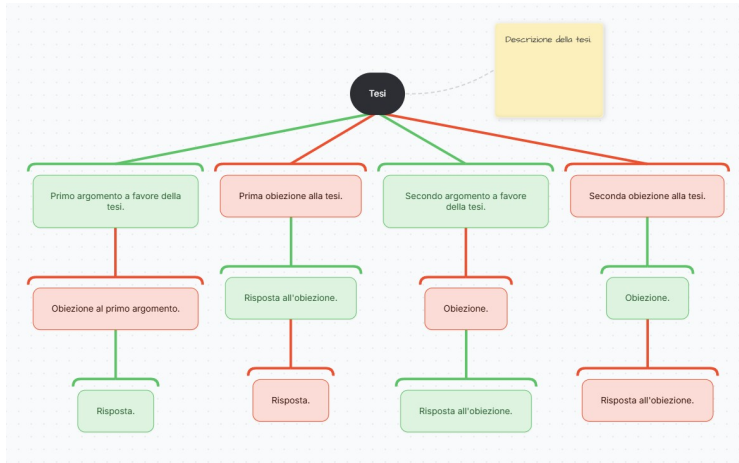


Figura 5

## Altre funzionalità

Selezioniamo un box e torniamo a dare uno sguardo al menu. Due voci sono particolarmente interessanti. Cliccando sulla voce *Notes* si apre un menu a destra della mappa che ci consente di inserire un'annotazione relativa alla ragione a favore o all'obiezione, ad esempio considerazioni che sono state fatte in classe discutendo l'argomento. Nel box relativo compare una piccola icona cliccabile.

Ancora nel menu troviamo la voce *Add connection*. Selezionandola compare una freccia curva tratteggiata che possiamo dirigere verso qualsiasi altro box per stabilire una connessione, una funzionalità utile se notiamo collegamenti tra momenti argomentativi diversi della nostra mappa.

Nella voce *Actions* troviamo altre due funzionalità interessanti. *Convert to implicit claim* converte l'argomento del box, appunto, in un *implicit claim*: il box assume un bordo tratteggiato. *Detach node* stacca il nodo selezionato e gli fa assumere la stessa forma della tesi centrale. *Make a linked submap* invece crea una sottomappa con il box selezionato e tutti i box che ne derivano. Nel box compare dunque l'icona di un link, cliccando sulla quale si apre un'altra mappa con il ramo che è stato separato.

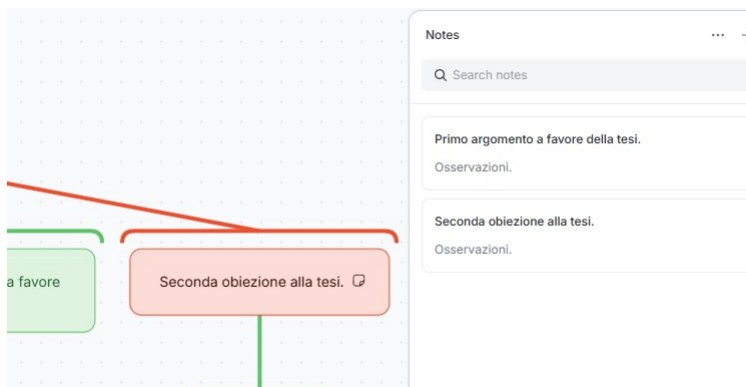


Figura 6

La versione precedente di Mindmup offriva una funzionalità particolarmente interessante: la possibilità di trasformare la mappa argomentativa in uno *storyboard* o *presentazione*, con l'aggiunta di *scene* o *diapositive*. Nella versione attuale lo storyboard non compare. Esso, si legge nel sito di Mindmup, sarà sostituito da una modalità di presentazione della mappa:

Poco dopo il lancio introdurremo la Modalità Presentazione (*Present Mode*), che permette di presentare le mappe in modo più naturale, come una sequenza di idee. Invece di gestire manualmente la struttura degli storyboard, la Modalità Presentazione consente di esplorare una mappa come se fosse una storia guidata.<sup>1</sup>

Al momento la voce *Present* è visibile nel menu in alto a destra, ma cliccando compare un callout che avvisa che la modalità presentazione arriverà presto.

---

<sup>1</sup>*What to Expect as Mindmup 3 Becomes the New Standard*, 25 marzo 2026, url: <https://www.mindmup.com/help/news/Mindmup-3-migration>. Traduzione mia.

## Esportare, pubblicare e condividere le mappe

Fino ad ora abbiamo creato la mappa senza alcuna registrazione. Finita la costruzione della mappa, possiamo esportarla dal *Main menu* e sinistra, selezionando File>Export. Abbiamo diverse possibilità: possiamo esportare la mappa come immagine (nei formati PNG e JPG, mentre il formato SVG è riservato agli utenti premium), come PDF, come *outline*, ossia come testo, mentre la possibilità di esportare nel formato proprietario di Mindmup o di salvare in locale, in modo da poter riprendere a lavorare alla mappa in un secondo momento, è riservata agli utenti premium. Possiamo esportare anche le note (*Notes*), in formato Word, Markdown o HTML\*. C'è ancora, da File>Share/Publish, la possibilità di condividere la mappa con altri, via e-mail, o di pubblicarla online, scegliendo anche un URL personalizzato. La mappa sarà pubblicato con un url così impostato: <https://atlas.mindmup.com/titolo-della-mappa>.

## Come usare Mindmup

Per la sua facilità di uso, Mindmup è uno strumento utile per mappare per così dire in presa diretta le discussioni filosofiche in classe. Il docente può aggiungere gli argomenti e le obiezioni mano a mano che vengono presentati dagli studenti. A mappa conclusa, è possibile fare una ricognizione critica di tutta la discussione. La mappa può essere salvata nel sito didattico o nella piattaforma di social classroom della classe (auspicabilmente Moodle, anche se non è la scelta più frequente delle scuole).

Il docente può costruire una mappa argomentativa anche per rendere più chiari i passaggi logici di un pensatore, illustrandone il pensiero, o come supporto all'analisi del testo filosofico. Non è possibile invece usare Mindmup come strumento per la discussione in tempo reale; in questo caso lo strumento più adatto è Kialo-Edu.

# Argdown

## Il Markdown al servizio dell'argomentazione

Argdown è uno strumento per la rappresentazione e l'analisi di argomentazioni in forma testuale e grafica. Nasce attorno al 2017 come progetto open source\*, sviluppato da Christian Voigt e altri ricercatori interessati all'argomentazione strutturata e alla visualizzazione delle relazioni tra proposizioni. L'obiettivo principale era creare un linguaggio che combinasse la semplicità dei linguaggi di markup\* come Markdown con la potenza analitica di strumenti per la mappatura argomentativa.

La semplice sintassi di Argdown permette di rappresentare premesse, conclusioni e relazioni tra affermazioni in un formato leggibile e facilmente modificabile. Una delle sue caratteristiche più apprezzate è la possibilità di convertire automaticamente il testo in mappe argomentative visive grazie a strumenti di rendering\*, rendendolo utile per chi si occupa di analisi del discorso e ragionamento strutturato. Inoltre, può essere integrato con diverse librerie di analisi dell'argomentazione, ampliando le sue possibilità di utilizzo.

Dopo la sua creazione, Argdown ha trovato applicazione in vari ambiti. In filosofia e logica è stato utilizzato per la didattica e l'analisi dei ragionamenti filosofici, mentre nel contesto del Debate e della retorica è servito per la costruzione e la valutazione delle argomentazioni nei dibattiti. Anche nel campo dell'intelligenza artificiale e del diritto è stato impiegato per modellare argomentazioni giuridiche e analizzare il ragionamento legale.

Per supportare il suo utilizzo, la comunità ha sviluppato vari strumenti, tra cui un editor online che permette di scrivere e visualizzare mappe argomentative in tempo reale, un plugin\* per Visual Studio Code e librerie JavaScript\* per l'integrazione con siti web e applicazioni.

# Cosa ci serve per cominciare

Per iniziare a usare Argdown non serve avere competenze informatiche avanzate. Il modo più semplice per provarlo è utilizzare la *sandbox* (all'url: <https://argdown.org/sandbox/html>) di Argdown, uno spazio online che permette di scrivere e visualizzare mappe argomentative senza dover installare nulla. Basta aprire il link, scrivere un testo seguendo una sintassi molto intuitiva e vedere immediatamente il risultato in forma grafica.

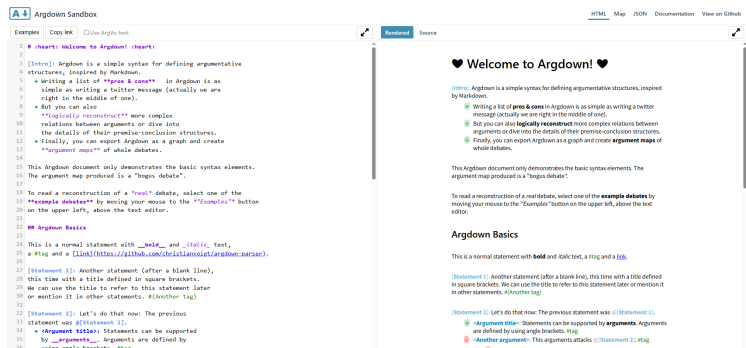


Figura 1

Per un uso avanzato si raccomanda tuttavia di scaricare e installare Visual Studio Code (url: <https://code.visualstudio.com/>), un programma gratuito che, con il plugin\* di Argdown, offre un ambiente più completo per scrivere e analizzare argomentazioni.

## Le dichiarazioni

Entriamo quindi nella sandbox. Troviamo nella finestra a sinistra del codice, che viene elaborato (*rendering*) nella finestra a destra.

Dopo una breve presentazione, ci troviamo di fronte al seguente testo che fornisce alcune basi di Argdown:

```
## Argdown Basics This is a normal statement with __bold__ and __italic__ text, a #tag and a [link](https://github.com/
```

christianvoigt/argdown-parser).

[Statement 1]: Another statement (after a blank line), this time with a title defined in square brackets. We can use the title to refer to this statement later or mention it in other statements. #(Another tag)

[Statement 2]: Let's do that now: The previous statement was @[Statement 1].

- + <Argument title>: Statements can be supported by `__arguments__`. Arguments are defined by using angle brackets. #tag
- <Another argument>: This arguments attacks @[Statement 2]. #tag
- <Yet another argument>: Arguments can also be supported or attacked. #yet-another-tag

Sappiamo, da queste prime righe, che per creare uno statement, ossia una dichiarazione, possiamo semplicemente scrivere del testo, formattandolo se occorre nel modo indicato. Possiamo però anche dare un titolo o nome alla nostra tesi usando le parentesi quadre seguite da due punti. Possiamo usare corsivo o grassetto o inserire link, secondo le regole del Markdown (per le quali si veda l'Appendice). Se lo riteniamo utile, possiamo anche aggiungere un tag.

[Democrazia imperfetta]: La democrazia rappresentativa è una forma di governo imperfetta perché, pur garantendo una partecipazione politica più ampia rispetto ad altre forme di governo, tende a consolidare **\*\*élite politiche ed economiche\*\*** che limitano l'effettivo esercizio del potere da parte dei cittadini. #elitismo

Il rendering\* HTML\* di questo codice è il seguente:

[**Democrazia imperfetta**]: La democrazia rappresentativa è una forma di governo imperfetta perché, pur garantendo una partecipazione politica più ampia rispetto ad altre forme di governo, tende a consolidare **élite politiche ed economiche** che limitano l'effettivo esercizio del potere da parte dei cittadini. #elitismo

Figura 2

## Gli argomenti

Ogni dichiarazione può essere sostenuta da una serie di argomenti a favore o contrari. Per aggiungere argomenti si procede come segue:

- Si va a capo dopo aver scritto la dichiarazione.
- Si lascia uno spazio per indentare il testo.
- Si inserisce quindi un + per un argomento a favore e un - per un argomento contrario.
- Si inserisce un testo di breve descrizione dell'argomento tra parentesi uncinate (<>), seguito da due punti e il testo dell'argomento.

Si veda questo esempio:

[La pena di morte è necessaria]: Bisogna introdurre la pena di morte nel sistema penale italiano.

+ <Deterrenza>: La pena di morte ha un innegabile valore di deterrenza.

+ <Giustizia>: La pena di morte risponde a un semplice bisogno di giustizia; per chi ha tolto la vita il carcere non è una pena sufficiente.

-<Errore giudiziario>: Con la pena di morte c'è il rischio di condannare innocenti.

-<Discriminazione>: La pena di morte può essere usata soprattutto verso gli appartenenti a una minoranza, come accade negli Stati Uniti, dove ad essere condannati a morte sono per lo più persone dalla pelle nera.

Il rendering\* HTML\* di questa mappa è il seguente:

[La pena di morte è necessaria]: Bisogna introdurre la pena di morte nel sistema penale italiano.

- + <Deterrenza>: La pena di morte ha un innegabile valore di deterrenza.
- + <Giustizia>: La pena di morte risponde a un semplice bisogno di giustizia; per chi ha tolto la vita il carcere non è una pena sufficiente.
- <Errore giudiziario>: Con la pena di morte c'è il rischio di condannare innocenti.
- <Discriminazione>: La pena di morte può essere usata soprattutto verso gli appartenenti a una minoranza, come accade negli Stati Uniti, dove ad essere condannati a morte sono per lo più persone dalla pelle nera.

Figura 3

Questo è sufficiente per inserire un argomento in una mappa. Tuttavia ogni argomento è la sintesi di un ragionamento più ampio, che può essere ricostruito nel dettaglio. Analizziamo il primo argomento contrario: “Con la pena di morte c'è il rischio di condannare innocenti”. Esso si basa su un principio di precauzione e giustizia. Vediamo come ricostruirlo logicamente.

Premessa 1: Il sistema giudiziario non è infallibile e possono verificarsi errori giudiziari.

Premessa 2: La pena di morte è irreversibile: una volta eseguita, non è possibile correggere un errore.

Premessa 3: Ci sono casi storici in cui innocenti sono stati giustiziati (es. Sacco e Vanzetti).

Conclusione: Pertanto, la pena di morte comporta il rischio inaccettabile di giustiziare innocenti.

In Argdown possiamo ricostruire questo ragionamento numerando tra parentesi tonda ogni argomento e inserendo quattro trattini brevi prima della conclusione.

(1) Il sistema giudiziario non è infallibile e possono verificarsi errori giudiziari.

(2) La pena di morte è irreversibile: una volta eseguita,

non è possibile correggere un errore.

(3) Ci sono casi storici in cui innocenti sono stati giustiziati (es. Sacco e Vanzetti).

----

(4) Pertanto, la pena di morte comporta il rischio inaccettabile di giustiziare innocenti.

Il rendering\* HTML\* è il seguente:

- (1) Il sistema giudiziario non è infallibile e possono verificarsi errori giudiziari.
  - (2) La pena di morte è irreversibile: una volta eseguita, non è possibile correggere un errore.
  - (3) Ci sono casi storici in cui innocenti sono stati giustiziati (es. Sacco e Vanzetti).
- 
- (4) Pertanto, la pena di morte comporta il rischio inaccettabile di giustiziare innocenti.

Figura 4

È possibile anche inserire annotazioni relative al tipo di inferenza inserendole tra due linee prima della conclusione:

(3)

----

Modus ponens

----

(4)

## Le mappe grafiche

Fino ad ora abbiamo considerato il rendering\* HTML\* del codice. Cliccando sulla voce di menu *Map*, in alto a destra, accediamo al rendering\* grafico del codice. È possibile visualizzare la mappa in due formati grafici, Viz Js e Dagre D3, due librerie JavaScript\* che danno risultati grafici un po' diversi. La nostra mappa sulla pena di morte si presenta in questo modo nel formato Viz Js:

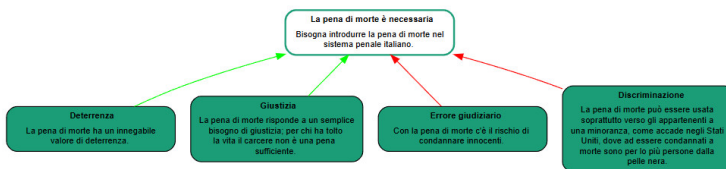


Figura 5

Questo invece è il rendering\* in Dagre D3:

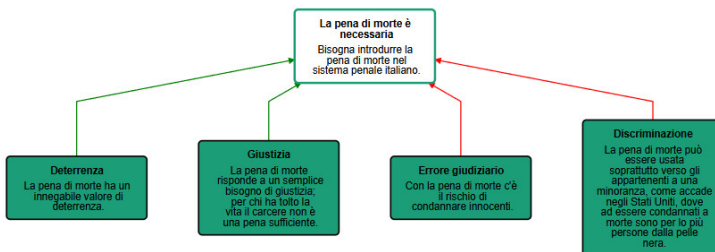


Figura 6

Come si vede, gli argomenti a favore della dichiarazione sono collegati ad essa da linee verdi che terminano con una freccia, mentre quelli contrari sono collegati da linee rosse.

Se si desidera invertire la direzione della freccia, occorre modificare il codice come segue:

[A]: a.  
 +> <a1>: a1  
 + <a2>: a2  
 -><b1>: b1  
 -<b2>: b2

Il risultato è questo:

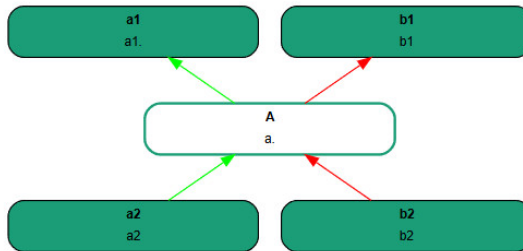


Figura 7

Gli argomenti contrassegnati con +> e -> risultano in uscita dalla dichiarazione, con una freccia che punta verso di essi.

## Raggruppare gli elementi

Osserviamo questo esempio:

# Esempio

## Gruppo

```

[T1]: Tesi 1
+ <Arg 1>: Primo argomento
+ <Arg 2>: Secondo argomento
-> [T2]
  
```

## Gruppo

```

[T2]: Tesi 2
- <Arg 3>: Primo argomento
+ <Arg 4>: Secondo argomento
  
```

### Sottogruppo

```

[T3]: Tesi 3
  
```

- <Arg 5>: Primo argomento
- + <Arg 6>: Secondo argomento
- > [T2]

Il rendering\* HTML\* è il seguente:

## Esempio

### Gruppo

[T1]: Tesi 1

- + <Arg 1>: Primo argomento
- + <Arg 2>: Secondo argomento
- > [T2]

### Gruppo

[T2]: Tesi 2

- <Arg 3>: Primo argomento
- + <Arg 4>: Secondo argomento

### Sottogruppo

[T3]: Tesi 3

- <Arg 5>: Primo argomento
- + <Arg 6>: Secondo argomento
- > [T2]

Figura 8

La mappa grafica invece si presenta come segue:

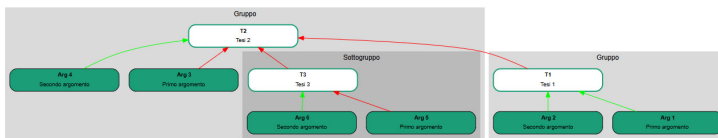


Figura 9

Come si vede, gli elementi ora sono raggruppati. Ci sono due gruppi, uno dei quali include un sottogruppo. Per ottenere questo risultato grafico Argdown fa uso della sintassi usata per formattare i titoli in Markdown. Per un titolo generale si usa # Titolo, mentre per un titolo di paragrafo si ricorre a un doppio cancelletto, e a tre cancelletti per il titolo di un sottoparagrafo (si veda l'Appendice). In Argdown questi titoli strutturano la mappa in gruppi e sottogruppi.

Si noti la freccia rossa che va da T1 e T3 a T2. È stata ottenuta semplicemente scrivendo -> [T2] nelle tesi 1 e 3.

## **Come usare Argdown in classe**

Argdown è lo strumento più esigente tra quelli presentati in questa sezione ed è esattamente in questo che risiede il suo valore didattico. La sintassi richiede di esplicitare ciò che nell'argomentazione ordinaria rimane implicito: la distinzione tra tesi, argomenti e premesse; il tipo di relazione tra ciascun elemento; la struttura inferenziale che porta dalle premesse alla conclusione. Scrivere un argomento in Argdown è diverso dal trascrivere un ragionamento già formato; il ragionamento in questo caso viene costruito passo dopo passo e spesso, nel processo di costruzione, emergono incoerenze o lacune che nella forma discorsiva sarebbero rimaste invisibili.

Argdown richiede un tempo di apprendimento superiore rispetto a Kialo-Edu o Mindmup. Si consiglia di introdurlo in un momento dell'anno in cui la classe ha già acquisito familiarità con il metodo dell'analisi argomentativa – tipicamente dopo avere lavorato con uno degli altri strumenti – e di dedicare almeno una lezione alla sandbox, lavorando su un esempio semplice prima di affrontare un testo filosofico complesso. L'investimento iniziale è ripagato dalla qualità dell'analisi che lo strumento rende possibile.

# Lexis

## Uno strumento per l'analisi del testo

L'ultimo strumento presentato in questa sezione è *Lexis*, una web app creata appositamente per questo manuale, disponibile online all'url <https://lexisweb.netlify.app> e rilasciata con licenza libera (la GNU General Public License\* v3.0). È progettata per supportare l'analisi argomentativa del testo – filosofico, ma anche scientifico, politico, giornalistico – identificando i ruoli funzionali che le sue parti svolgono nell'argomentazione complessiva. Consente di caricare un testo, selezionare i passaggi rilevanti con il mouse e assegnare loro un tag che ne identifica la funzione argomentativa. Il risultato è un testo annotato cromaticamente, in cui la struttura logica diventa visibile a colpo d'occhio.

In quanto web app, non richiede installazione e funziona su qualsiasi dispositivo. Il file è un singolo HTML\*, distribuibile liberamente, che non dipende da alcun server o account esterno; per usarlo in locale, senza connessione Internet, è sufficiente salvare la pagina web.

Lo strumento offre due livelli di analisi, cui si aggiunge il modello di Toulmin che conosciamo:

- La struttura logica: un vocabolario elementare – Premessa, Conclusione, Obiezione, Confutazione – che permette di identificare le componenti di base di qualsiasi ragionamento argomentativo.
- La struttura concettuale: un secondo livello di analisi – Definizione, Esempio, Analogia, Assunzione – che permette di identificare i dispositivi retorici e concettuali con cui l'autore costruisce e sostiene il proprio argomento.

## Guida operativa

L'interfaccia è essenziale (figura 1): una colonna laterale con i tag disponibili, un'area centrale che mostra il testo, una barra di stato in basso con il conteggio delle annotazioni. Dopo aver caricato il testo si selezionano i passaggi con il mouse e si assegna un tag. I passaggi annotati appaiono evidenziati con colori distinti, ciascuno con un'etichetta che ne indica la funzione. Il lavoro di annotazione si salva esportando un file JSON\*, che può essere ricaricato in qualsiasi momento per riprendere o completare l'analisi. Il lavoro può essere condiviso con il docente semplicemente inviando il JSON\* via email o caricandolo su un drive condiviso.

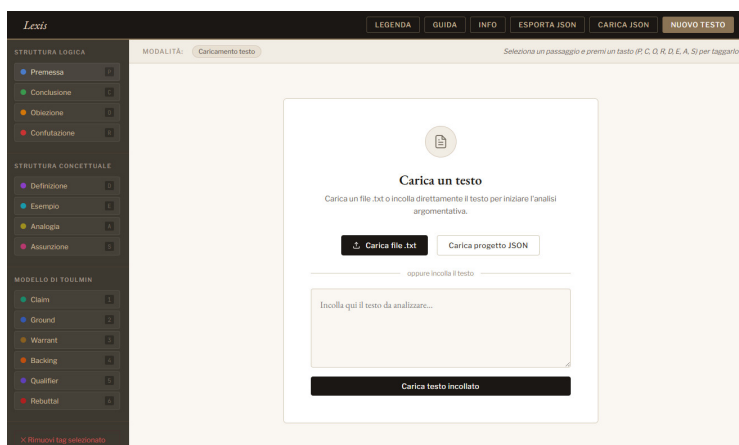


Figura 1

Dalla schermata iniziale ci sono due modalità di caricamento del testo: incollare direttamente il testo nell'area di testo oppure importare un file .txt. Prima di avviare l'analisi, conviene compilare i tre campi in alto: titolo del testo, autore e riferimento bibliografico, nome dello studente. Queste informazioni vengono salvate nell'esportazione JSON\* e permettono di identificare il lavoro in modo univoco.

La procedura di annotazione è identica per tutti i tag: si seleziona il passaggio con il mouse, appare un menu contestuale con tutti i tag disponibili, si sceglie quello pertinente. In alternativa si può usare la

scorciatoia da tastiera: dopo aver selezionato il passaggio, premere il tasto corrispondente al tag desiderato (P per Premessa, C per Conclusione, O per Obiezione, e così via). Il passaggio appare immediatamente evidenziato con il colore del tag assegnato (figura 2).

MODALITÀ: Analisi Seleziona un passaggio per taggato

Testo di esempio

Autore, data — aggiungi riferimento bibliografico

STUDENTE

Nome dello studente

Chiamiamo «credenza giustificata» una credenza che abbiamo buone ragioni per ritenere vera.

**DEFINIZIONE** Ora, supponiamo che una persona competente e razionale, dopo aver esaminato le stesse prove che abbiamo esaminato noi, arrivi a una conclusione diversa dalla nostra. Questo fatto — il disaccordo tra pari epistemici — ha conseguenze importanti per la nostra credenza.

Se qualcuno che ha le stesse informazioni e le stesse capacità di ragionamento giunge a una conclusione opposta, allora la nostra certezza iniziale era probabilmente eccessiva. **PREMESSA** Pertanto, di fronte a un disaccordo genuino tra pari, dovremmo ridurre il grado di confidenza nella nostra posizione. **CONCLUSIONE**

Si potrebbe obiettare che cedere alle opinioni altrui significa relativizzare la verità: se ogni disaccordo obbliga a rivedere le proprie credenze, nessuno potrà mai credere in nulla con fermezza. **OBJEZIONI** Ma questa obiezione confonde due cose distinte. Non si chiede di abbandonare la propria posizione, ma solo di calibrarne la certezza — come fa un buon meteorologo che, di fronte a dati contrastanti, abbassa la probabilità assegnata a una previsione senza smettere di farla. **CONFUTAZIONE**

In realtà, la capacità di aggiornare le proprie credenze alla luce del disaccordo è un segno di razionalità, non di debolezza. Chi non rivede mai le proprie opinioni non sta difendendo la verità: sta difendendo se stesso.

Figura 2

Per aggiungere note di analisi a un passaggio già taggato, si clicca sul testo evidenziato: si apre il pannello laterale destro, dove è possibile scrivere osservazioni libere. Le note vengono salvate insieme all'annotazione nel file JSON\*. Per rimuovere un tag, si clicca sul passaggio evidenziato e si preme il pulsante "Rimuovi tag selezionato" nella barra laterale.

Come anticipato, il lavoro si esporta usando il pulsante *Esporta JSON*. Il file JSON\* contiene il testo integrale, tutte le annotazioni con la loro posizione, le note di analisi e i metadati del documento (titolo, autore, studente). Per riprendere il lavoro in una sessione successiva, si ricarica il JSON\* dalla schermata iniziale o dal pulsante *Carica JSON* nell'interfaccia.

## Lexis in classe

Lexis può essere usato in modi diversi a seconda del livello della classe, degli obiettivi didattici e del tempo disponibile. Propongo tre modalità principali.

### 1. *Analisi del testo guidata dal docente*

Il docente accede a Lexis alla lavagna elettronica, carica il testo da analizzare e conduce l'annotazione collettivamente, chiamando gli studenti a identificare i passaggi e a motivare la scelta del tag. È la modalità più adatta all'introduzione dello strumento e delle categorie. La scelta del tag diventa oggetto di discussione: perché, ad esempio, questa frase è una premessa e non un esempio?

### 2. *Analisi individuale come attività a casa*

Il docente assegna un brano breve (15-30 righe) e chiede agli studenti di annotarlo individualmente con Lexis, esportare il file JSON\* e consegnarlo. Il docente riapre i file degli studenti e confronta le annotazioni. Questa modalità produce una ricchezza di dati sulla comprensione della classe: vedere dove diversi studenti assegnano tag diversi allo stesso passaggio è utile per valutare lo sviluppo delle competenze della classe. Un passaggio che alcuni taggano come premessa e altri come conclusione segnala una difficoltà di comprensione che può diventare oggetto di discussione collettiva nella lezione successiva.

### 3. *Confronto tra annotazioni diverse*

Due studenti o due gruppi annotano lo stesso testo indipendentemente. Si confrontano le annotazioni: dove concordano? Dove divergono? Le divergenze è il punto più filosoficamente interessante: segnalano luoghi del testo in cui la struttura argomentativa è ambigua o contestabile.

Questa modalità introduce naturalmente la dimensione ermeneutica dell'analisi argomentativa: non esiste una sola annotazione giusta per ogni testo. L'interpretazione dell'argomento è essa stessa un atto interpretativo, soggetto a discussione razionale.

## **II Coltivare un giardino di pensieri**

# Connettere le idee

## Nozioni e conoscenza

Insegnare è spesso disperante. Sembra che non vi sia alcun rapporto, o quasi, tra lo sforzo e i risultati. E non solo perché qualcuno dei nostri studenti rifiuta i nostri sforzi generosi per trasmettergli il sapere. C'è chi si mostra volenteroso, si impegna, fa tutto ciò che gli viene richiesto, ma il risultato è modesto. La studentessa si limita a ripetere le cose che ha letto sul manuale o sentito a lezione, senza alcuna originalità e creatività. E dopo qualche mese quelle nozioni così fragili scompariranno del tutto.

C'è, è evidente, non solo un problema di motivazione, ma anche di *metodo*. Lo si ripete spesso: la scuola non deve offrire nozioni, ma un metodo di studio solido, che consenta allo studente di cercare da solo la conoscenza quando sarà uscito da scuola. Ma in cosa consiste questo metodo? Chi invoca questo ruolo della scuola spesso ritiene che si tratti di tenere alto l'impegno delle studentesse con richieste pressanti riguardanti i compiti e lo studio a casa, con verifiche frequenti e rigorose. Ma nulla assicura che uno studente sia in grado, pur adempiendo in modo ineccepibile a tutti gli impegni scolastici, di avvicinarsi realmente alla conoscenza, balzando fuori dal terreno fragile del nozionismo. Sarà magari disciplinato, ma non è detto che questa disciplina gli consenta di acquisire anche l'elemento di creatività e originalità che è essenziale per diventare soggetti attivi di conoscenza.

Cos'altro, allora? Cosa vuol dire lavorare al metodo, a scuola? Vuol dire insegnare tre cose. La prima è la ricerca delle informazioni da fonti affidabili. La seconda è la profondità: non fermarsi alla prima risposta, ma cercare ancora, e ancora, e ancora. La terza, che è quella di cui ci occuperemo in questa sezione, è mettere in relazione, in dialogo, in contrapposizione se necessario le informazioni, i dati, le idee.

## Imparare davvero

La ricerca cognitiva e psicopedagogica degli ultimi cinquant'anni ha prodotto un quadro coerente su come funziona l'apprendimento significativo.

Grazie alle ricerche di Donald O. Hebb<sup>1</sup> sappiamo che i neuroni si attivano insieme e tendono a rinforzare le proprie connessioni sinaptiche. Il *principio di Hebb* stabilisce che "Neurons that fire together, wire together" (neuroni che si attivano insieme, si collegano insieme). Se due neuroni si attivano nello stesso momento, ciò rende più probabile che in futuro si attivino insieme. Ciò ha conseguenze decisive sulla nostra concezione dell'apprendimento. Imparare non vuol dire depositare informazioni in una sorta di archivio cerebrale. Se queste informazioni restano isolate, il decadimento è rapido. La vera conoscenza nasce dalla creazione di connessioni tra le reti neurali. Ogni volta che un concetto nuovo viene collegato a qualcosa di già noto, si modifica fisicamente la struttura del cervello; se l'informazione viene presentata senza ancoraggi al già noto, il segnale prodotto è debole e il cervello non ha ragioni per conservarlo.

Su questa base si costruisce la teoria dell'apprendimento significativo di Ausubel<sup>2</sup>. La tesi centrale è che l'apprendimento avviene in modo duraturo e trasferibile solo quando le nuove informazioni vengono ancorate a strutture cognitive preesistenti. Qualsiasi modello didattico che presenti contenuti nuovi senza costruire ponti con il già noto produce, per Ausubel, un apprendimento meccanico: fragile, non trasferibile e rapidamente dimenticato.

Craik e Lockhart precisano il meccanismo con la teoria dei livelli di elaborazione<sup>3</sup>. La memoria non dipende da quanto a lungo si studia qualcosa, ma da quanto profondamente lo si elabora. L'elaborazione superficiale (trascrivere, copiare, ripetere) produce tracce mnemoniche deboli. L'elaborazione semantica profonda – riformulare con parole pro-

---

<sup>1</sup>D.O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*, Wiley, New York 1949.

<sup>2</sup>D. P. Ausubel, *Educational Psychology: A Cognitive View*, Holt, Rinehart and Winston, New York 1968.

<sup>3</sup>F. I. Craik, R.S. Lockhart, *Levels of Processing: A Framework for Memory Research*, in "Journal of Verbal Learning & Verbal Behavior", 11(6), 1972, pp. 671–684

prie, costruire relazioni di significato, chiedersi perché un'idea è importante – produce tracce robuste e durature. La differenza sperimentale è consistente e replicata.

Roediger e Karpicke<sup>4</sup> aggiungono un dato che può sorprendere: il modo migliore di studiare non è rileggere il materiale, ma recuperarlo attivamente senza guardarlo. Il loro esperimento mostrò che il gruppo che praticava il recupero attivo (*retrieval practice*) ricordava il 50% in più, dopo una settimana, rispetto al gruppo che aveva solo riletto. Il meccanismo, chiamato *testing effect*, funziona perché recuperare un'informazione dalla memoria è un'operazione cognitivamente molto più impegnativa del passivo re-esporsi ad essa, e produce tracce proporzionalmente più solide.

A questo si aggiunge il principio della ripetizione dilazionata (*spaced repetition*), già intuito da Ebbinghaus e confermato da Cepeda e colleghi in una meta-analisi del 2006<sup>5</sup>: tornare sullo stesso materiale a distanza crescente di tempo produce una consolidazione molto più efficace della ripetizione concentrata. Infine, Novak e Gowin<sup>6</sup>, sviluppando il lavoro di Ausubel, mostrarono che gli studenti che organizzavano le proprie conoscenze in forma di rete ottenevano risultati significativamente migliori nelle prove di comprensione e di trasferimento rispetto a quelli che usavano appunti lineari.

Dunque: un apprendimento reale, solido, duraturo richiede operazioni come l'ancoraggio a ciò che è noto, l'elaborazione semantica, il recupero attivo, la costruzione esplicita di connessioni. Qualsiasi pratica didattica che ignori queste evidenze sperimentali produce un apprendimento fragile, indipendentemente da quanto le lezioni dell'insegnante siano brillanti, colte e vivaci.

---

<sup>4</sup>H.L. Roediger III, J.D. Karpicke, *Test-Enhanced Learning: Taking Memory Tests Improves Long-Term Retention*, in "Psychological Science", 17(3), 2006, pp. 249-255.

<sup>5</sup>N. J. Cepeda, H. Pashler, E. Vul, J. T. Wixted, D. Rohrer, *Distributed Practice in Verbal Recall Tasks: A Review and Quantitative Synthesis*, in *Psychological Bulletin*, 132(3), 2006, pp. 354-380.

<sup>6</sup>J. D. Novak, D. B. Gowin, *Learning How to Learn*, Cambridge University Press, Cambridge 1984.

## Prendere appunti

Prendere appunti è la pratica più significativa del *lavoro* dello studente in classe. Anche se non esiste un obbligo reale, può succedere che una studentessa venga ripresa se si limita ad ascoltare l'insegnante, senza prendere appunti; e gli appunti sono una delle cose da cui possono essere dispensati gli studenti con disturbi specifici dell'apprendimento.

Il lavoro sugli appunti per lo più consiste in questo: lo studente cerca di trascrivere quello che dice l'insegnante, il più velocemente possibile; rilegge a casa e risistema (cosa che non sempre accade); qualche giorno prima dell'interrogazione studia sul manuale, usando gli appunti come promemoria di quello che è stato detto in classe.

Questo modello, che è talmente diffuso da sembrare naturale, è invece il risultato di una lacuna formativa sistematica: nessuno insegna alle studentesse come prendere appunti. Si dà per scontato che sappiano farlo, come si dà per scontato che sappiano leggere.

Gli appunti lineari hanno alcune caratteristiche strutturali che li rendono particolarmente inadeguati, anche e soprattutto per una disciplina come la filosofia. Sono sequenziali: rispecchiano l'ordine cronologico delle lezioni, a sua volta legato allo sviluppo storico della disciplina, ma non seguono la struttura logica delle idee. Sono inoltre passivi: la trascrizione rapida è un'operazione meccanica che attiva pochissima elaborazione semantica. Si può copiare perfettamente dalla lavagna senza capire nulla di quello che si copia. E sono, infine, isolati: il quaderno di filosofia non dialoga con quello di storia, quello di storia non dialoga con quello di letteratura. Le connessioni interdisciplinari, che sono le più feconde, rimangono implicite o non vengono mai costruite e pertanto svaniscono: la curva di Ebbinghaus si applica inesorabilmente a qualsiasi informazione registrata in modo lineare e senza rielaborazione nel tempo.

## Il ruolo della filosofia

Nella didattica della filosofia il nozionismo ha conseguenze anche peggiori che in altri campi disciplinari, poiché è possibile *simulare* la conoscenza. La filosofia dispone di un ampio repertorio di "contenuti" che possono essere presentati, memorizzati e verificati in forma no-

zionistica: le quattro cause aristoteliche, le dodici categorie kantiane dell'intelletto, le cinque vie di Tommaso e così via. Uno studente che non sa nulla di matematica difficilmente può simulare. Uno studente che sa elencare le quattro cause aristoteliche senza capire perché Aristotele le introduce può illudersi di sapere, e può essere che se ne convinca anche il suo insegnante.

E così la scuola finisce per rendere un pessimo servizio alla filosofia, che è la disciplina che, per definizione, si occupa delle relazioni tra le idee. Un sistema filosofico è una rete di problemi, risposte, obiezioni, risposte alle obiezioni. Insegnare filosofia significa insegnare a muoversi in queste reti concettuali, a entrarci da un punto e a capire dove porta ogni filo che si segue; e magari a costruire reti proprie.

La filosofia ha una responsabilità particolare rispetto al problema che stiamo affrontando. Essa insegna, si dice, a pensare. E se questo è il suo compito, non può non chiedere di più, esigere che si vada oltre la nozione per costruire una rete di conoscenza che sarà anche, necessariamente, interdisciplinare. La filosofia dialoga con l'arte, con la storia, con la scienza, con la religione; è il sapere che problematizza ogni campo dell'esperienza e che è insofferente verso ogni chiusura disciplinare. Insegnare filosofia vuol dire insegnare non cosa ha detto un filosofo, ma perché lo ha detto, rispondendo a quali problemi, in quale contesto sociale ed economico; e quanto, di quello che ha detto, può essere uno strumento ancora utile e urgente per noi. Vuol dire *andare al di là*, spingersi sempre oltre i dati, alla ricerca del significato.

## **Il metodo Zettelkasten**

Niklas Luhmann (1927-1998) è stato uno dei sociologi più influenti del Novecento ed anche uno dei più prolifici. Dietro la sua ampia produzione saggistica c'era, naturalmente, un ingegno di prim'ordine, ma anche un sistema di gestione della conoscenza particolarmente efficace: lo Zettelkasten (in tedesco "scatola dei foglietti"). Il sistema di Luhmann era fisico, analogico ed apparentemente rudimentale: decine di migliaia di cartoncini contenuti in cassette di legno, ciascuno con una singola idea scritta a mano, un codice identificativo e rimandi ad altri cartoncini. Ma il sistema seguiva un principio organizzativo radicalmente diverso da qualsiasi archivio tradizionale. Le note di Luhmann non

erano archiviate per argomento, per autore, per data, o per qualsiasi altra categoria esterna, ma per connessione. L'intuizione geniale e precorritrice di Luhmann è stata di non mettere le note in una cartella tematica, ma di associarle in base al fatto che avessero qualcosa da comunicarsi e potessero far nascere nuove connessioni. Una nota su Parsons poteva stare vicino a una nota su Darwin, se le due idee si illuminavano a vicenda, così come una nota su un problema giuridico poteva stare vicino a una nota su un problema estetico. Nel tempo, la rete di connessioni diventava così densa da generare associazioni che Luhmann non aveva pianificato; sfogliare lo Zettelkasten produceva sorprese. Lo Zettelkasten diventava una sorta di interlocutore. Scrive Luhmann:

Come risultato di un lavoro prolungato con questa tecnica, emerge una sorta di seconda memoria, un alter ego con cui si può comunicare in modo continuo. Similmente alla propria memoria, manca di un ordine generale meticolosamente costruito, di una gerarchia e certamente di una struttura lineare come un libro. È proprio questo che le permette di acquisire una vita propria, indipendente dal suo autore.<sup>7</sup>

Il metodo Zettelkasten si articola in tre operazioni fondamentali. La prima è la rielaborazione: quando si legge o si ascolta qualcosa di interessante, non si copia il testo, ma si riformula. L'idea viene tradotta in parole proprie, operazione che costringe a capirla davvero prima di scriverla. La seconda è l'atomizzazione: si isola una singola idea per nota, formulata in modo chiaro e autosufficiente, capace di stare da sola senza il contesto da cui è emersa. La terza e più importante è il collegamento: prima di archiviare una nota ci si chiede a quali altre note si connette e si aggiungono i riferimenti espliciti.

---

<sup>7</sup>"Als Ergebnis längerer Arbeit mit dieser Technik entsteht eine Art Zweitgedächtnis, ein alter Ego, mit dem man laufend kommunizieren kann. Es weist, darin dem eigenen Gedächtnis ähnlich, keine durchkonstruierte Gesamtordnung auf, auch keine Hierarchie und erst recht keine lineare Struktur wie ein Buch. Eben dadurch gewinnt es ein von seinem Autor unabhängiges Eigenleben". N. Luhmann, *Kommunikation mit Zettelkasten. Ein Erfahrungsbericht*, in H. Baier et al. (eds.), *Öffentliche Meinung und sozialer Wandel / Public Opinion and Social Change*, Westdeutscher Verlag GmbH, Opladen 1981, p. 225. Traduzione mia.

A queste tre operazioni corrispondono due tipologie di note. La *nota di letteratura* è il prodotto della prima operazione. È una sintesi personale di ciò che si è letto o ascoltato, ancora legata alla fonte. La nota permanente — lo *Zettel* propriamente detto — è il risultato della rielaborazione e del collegamento delle prime note (che vengono poi scartate, perché non più utili): la singola idea viene fissata e inserita nella rete.

Il metodo Zettelkasten rimase a lungo uno strumento di ricerca per accademici, noto ai sociologi che studiavano Luhmann ma non particolarmente diffuso al di fuori di quel contesto. La sua riscoperta è relativamente recente. Sönke Ahrens, nel suo *How to Take Smart Notes* (2017)<sup>8</sup>, ha reso il metodo accessibile a un pubblico molto più ampio, e da quel momento la comunità delle persone interessate al *personal knowledge management* è esplosa. Oggi esistono decine di strumenti digitali progettati esplicitamente per implementare i principi dello Zettelkasten, e una letteratura pratica in rapida crescita.

## Dallo Zettelkasten al Digital Garden

Il passaggio dallo Zettelkasten analogico di Luhmann al Digital Garden è, sul piano concettuale, quasi naturale. I principi sono gli stessi: creazione di una nota atomica, collegamento con altre note, costruzione progressiva di una rete concettuale. A cambiare è la tecnologia, che fa tuttavia una differenza significativa. Quella più ovvia è operativa: i link digitali sono cliccabili, è possibile raggiungere istantaneamente la nota collegata ed è possibile molto facilmente visualizzare un grafico di tutte le note collegate. Ciò che chiedeva a Luhmann un lavoro di codifica elaborato e anni di pratica per ottenere un archivio *navigabile*, è possibile oggi in poche ore.

La prima attestazione registrata del termine Digital Garden risale al saggio del 1998 *Hypertext Gardens: Delightful Vistas* di Mark Bernstein.<sup>9</sup> Nel testo, organizzato anch'esso come ipertesto, Bernstein descriveva i

---

<sup>8</sup>S. Ahrens, *How to Take Smart Notes. One Simple Technique to Boost Writing, Learning and Thinking*, self-published, 2017.

<sup>9</sup>M. Bernstein, *Hypertext Gardens: Delightful Vistas*, Eastgate Systems, 1998, url: <https://www.eastgate.com/garden/Enter.html>

giardini come spazi organizzati ma non rigidamente strutturati: spazi in cui gli utenti possono esplorare, addentrarsi nell'ignoto. Scriveva:

Giardini e parchi si trovano tra i terreni agricoli e la natura selvaggia. Il giardino è un terreno agricolo che delizia i sensi, progettato per il piacere piuttosto che per la produzione di beni. Il parco è natura selvaggia, addomesticata per il nostro godimento. Poiché la maggior parte degli ipertesti non mira né alla selvatichezza dei contenuti non pianificati, né alle file ordinate dell'organizzazione formale, giardini e parchi possono ispirare un nuovo approccio alla progettazione degli ipertesti e possono aiutarci a comprendere gli schemi che osserviamo nella scrittura ipertestuale di qualità.<sup>10</sup>

La vera codificazione filosofica del concetto arriva con Mike Caulfield, il cui intervento del 2015 *The Garden and the Stream*<sup>11</sup> è considerato il testo fondativo del movimento del *digital gardening*. Caulfield, che stava lavorando con Ward Cunningham – l'inventore del wiki\* – sul progetto del *federated wiki*, denunciava la riduzione della rete Internet allo Stream, un flusso di contenuti nel quale il nuovo sostituisce il vecchio, suscitando reazioni immediate a scapito della conoscenza a lungo termine, contrapponendo ad esso l'idea del Web come Garden, appunto, ossia uno spazio in cui le informazioni sono associate non secondo un criterio meramente cronologico, ma secondo percorsi che l'utente stesso può muoversi liberamente creando nuovi percorsi; uno strumento al servizio della complessità e della creatività.

Questi principi hanno trovato espressione tecnologica in una serie di strumenti che privilegiano i collegamenti bidirezionali e le note in

---

<sup>10</sup>"Gardens and parks lie between farmland and wilderness. The garden is farmland that delights the senses, designed for delight rather than commodity. The park is wilderness, tamed for our enjoyment. Since most hypertext aims neither for the wilderness of unplanned content, nor for the straight rows of formal organization, gardens and parks can inspire a new approach to hypertext design and can help us understand the patterns we observe in fine hypertext writing." Ivi, url: <https://www.eastgate.com/garden/Gardens.html>. Traduzione mia.

<sup>11</sup>M. Caulfield, *The Garden and the Stream*, 2015, url: <https://hapgood.us/2015/10/17/the-garden-and-the-stream-a-technopastoral/>

continua evoluzione, la vista d'insieme sotto forma di grafo, l'uso del Markdown e l'indipendenza dalle grandi piattaforme proprietarie.

## **Il Digital Garden in classe**

Il Digital Garden entra in classe inizialmente affiancandosi al quaderno degli appunti lineari, con la finalità di trasformare il modo in cui gli studenti prendono appunti. Può essere previsto un momento, alla fine dell'ora di lavoro, per la rielaborazione di ciò che è stato imparato, che può avvenire anche sul quaderno degli appunti, per essere poi trasferito in digitale.

È prevedibile qualche resistenza. La convinzione che i più giovani siano spontaneamente portati all'uso degli strumenti informatici, in quanto *nativi digitali*, è infondata, come può constatare chiunque insegni. Operazioni come la gestione di una casella di posta elettronica o il caricamento dei file su un cloud possono essere tutt'altro che semplici per degli adolescenti; e molti hanno non poche difficoltà a passare dal metodo analogico della scrittura su quaderno a quello digitale della scrittura in Markdown in un sistema complesso di gestione delle note. Questo attaccamento all'analogico potrebbe risultare perfino commovente, se non vi fosse anche, come componente forse non secondaria, quella certa pigrizia che ostacola l'esperienza del nuovo, e che la scuola non dovrebbe assecondare.

Non c'è da aspettarsi troppo da una forzatura, in questo ambito come in tanti altri che hanno a che fare con la scuola. La creazione di un Digital Garden complesso è un progetto ambizioso e impegnativo, ma anche bello e forse perfino divertente. Se non c'è consapevolezza di queste potenzialità della pratica, conviene non forzare. È importante prevedere una fase iniziale di meta-riflessione sulle proprie pratiche di studio, sui metodi che si usano e i loro limiti e sulle potenzialità che vengono dalla tecnologia. Solo dopo una percezione condivisa del valore della pratica è possibile avviare un percorso che può essere pieno di sfide, ma anche di opportunità. Il consiglio è di partire con uno strumento più semplice come Vicus Iurosum, passando poi, quando la logica del Digital Garden sarà stata compresa, a programmi tecnicamente più impegnativi come TiddlyWiki o Obsidian.

Vediamo intanto come costruire in concreto una nota.

TITOLO: L'essere e il non-essere in Parmenide

DATA:

AGGIORNAMENTO:

TAG: #ontologia #Presocratici #Parmenide

LINK: [[...]]

FONTE:

CONTENUTO: "L'essere è, il non-essere non è." Il non-essere è impensabile: pensare il nulla significa comunque pensare qualcosa. Da questo principio segue che l'essere è ingenerato, eterno, immobile. Il mutamento presuppone che qualcosa passi dall'essere al non-essere: ma questo è impossibile. I sensi ci ingannano; solo la ragione coglie il vero.

OSSERVAZIONI: Parmenide dice che i sensi ingannano, ma sono i sensi l'unica cosa di cui sono sicuro.

In questa nota abbiamo isolato un concetto del pensiero di Parmenide. I *Tag* consentono di collegare questa nota ad altre e recuperarla rapidamente: attraverso il tag #ontologia, ad esempio, potremo collegare tutte le note che hanno a che fare con il tema, anche in autori molto lontani, cogliendo l'evoluzione del concetto di essere ma anche il ripresentarsi di certe posizioni. La *Data* e l'*Aggiornamento* ci consentono di seguire l'evoluzione temporale del giardino e il lavoro di *cura* che gli è stato dedicato. La sezione *Link* è al momento vuota. La studentessa potrà tornare sulla nota man mano che approfondirà lo studio di altri autori e inserirà i collegamenti che riterrà significativi (e in questo caso segnerà la data di aggiornamento). Collegamenti che possono riguardare altri autori – potrà collegare ad esempio Parmenide con Gorgia – ma anche una nota-domanda personale, che sarà lo strumento per la ricerca autonoma della studentessa. Importante è la *Fonte*. Il più delle volte si tratterà del manuale o della lezione, ma questa voce ci consente anche di far emergere le letture autonome, gli approfondimenti su Internet o le conversazioni degli studenti. Il *Contenuto* è una breve descrizione del concetto. La nota naturalmente non esaurisce il concetto, ma lo ancora e lo rende collegabile. Il resto viene dal lavoro in classe e dalla lettura dei testi.

Un secondo esempio:

TITOLO: L'enkrateia in Socrate

DATA:

AGGIORNAMENTO:

TAG: #etica #Socrate #libertà #autodominio

LINK: [[...]]

FONTE:

CONTENUTO: L'enkrateia è il dominio di sé, la capacità di governare i propri desideri e impulsi attraverso la ragione. Per Socrate è la condizione per essere liberi.

OSSERVAZIONI: La ragione può davvero governare i desideri?

Abbiamo isolato qui l'idea di base dell'etica di Socrate secondo Senofonte. Il tag libertà ci permetterà di recuperarla tutte le volte che indagheremo sulla libertà, anche a distanza di un anno, magari studiando Spinoza. In questo caso potremo aggiungere un link ad una nota intitolata *La libertà per Spinoza*.

A queste note, che possiamo definire concettuali, occorrerà aggiungere almeno altre due tipologie. La prima è quella delle *note personali*, auspicabili anche se non prescrivibili, nelle quali la studentessa segnerà le proprie riflessioni personali, utilizzando anche in questo caso tag e link. Una riflessione sulla libertà, ad esempio, può avere il tag #libertà e un link alla nota su un filosofo che l'ha stimolata. La seconda è quella delle *domande*, che dovrebbero essere anch'esse personali, segnando il percorso personale dello studente, e che possono nascere spontaneamente dalle *Osservazioni* delle note-concetto. Una nota-domanda sarà all'inizio vuota. Si riempirà man mano con i link alle note nelle quali si sviluppa la riflessione. Ad esempio una nota con la domanda *Siamo davvero liberi?* può rimandare alla nota *Le cose che ci condizionano* e alla nota *La libertà in Spinoza*.

Man mano che si procede, si crea una struttura complessa, che con alcuni programmi è possibile visualizzare graficamente. Nella figura 1, ad esempio, la vista grafica di un Digital Garden con il software Obsidian.

Come si vede, si tratta di una struttura che ha alcuni punti forti di aggregazione, qualche altro più debole e non poche note isolate. Lo sguardo d'insieme consente di farsi un'idea immediata dell'efficienza della struttura. Più le note sono integrate e dialogano tra di loro, più il Digital Garden è fecondo. La struttura richiama visivamente quella

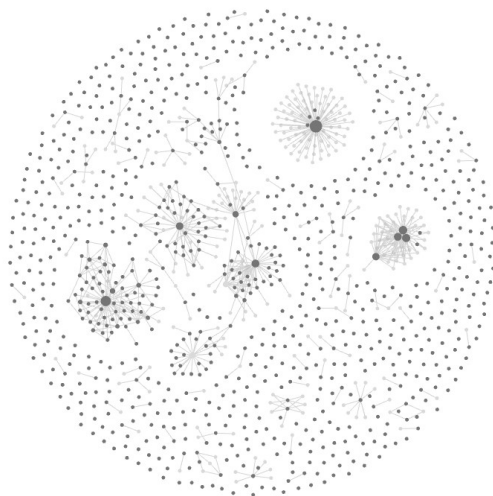


Figura 1

di un cervello, con le note-neuroni collegate da sinapsi. Proprio come per un cervello, per un Digital Garden sono importanti le connessioni, più ancora che la quantità di note. Nel campo del Personal Knowledge Management\* (PKM) si parla di Digital Brain per indicare qualcosa di non troppo diverso dal Digital Garden: informazioni che vengono raccolte, rielaborate e collegate in un ambiente digitale.<sup>12</sup> La differenza è che nel caso del Digital Brain l'enfasi è sull'organizzazione delle informazioni, mentre un Digital Garden ha un approccio più esplorativo e creativo.

## Valutare il Digital Garden

Il Digital Garden è uno strumento per combattere la *simulazione dell'apprendimento* di cui s'è detto. Intende spezzare la routine nozionistica, consolidata e perciò rassicurante, alla ricerca di un sapere che resti nel

---

<sup>12</sup>Il testo di riferimento sul Digital Brain è T. Forte, *Building a Second Brain: A Proven Method to Organise Your Digital Life and Unlock Your Creative Potential*, Profile Books, London 2023.

tempo. Un Digital Garden curato, ricco, aggiornato può essere indice di un approccio non superficiale al sapere e dello sviluppo progressivo di competenze raffinate, come la capacità di porsi domande e di costruire percorsi culturali autonomi.

I criteri di valutazione dovrebbero vertere su tre aspetti principali:

1. L'atomicità e la rielaborazione: si valuta se le note inserite nel giardino sono scritte in parole proprie, evitando il copia-incolla e dimostrando uno sforzo di sintesi personale e comprensione.
2. La ricchezza delle connessioni: il valore del giardino digitale risiede nei collegamenti tra le note. Un indicatore di eccellenza è la capacità dello studente di creare collegamenti trasversali e interdisciplinari, trasformando le informazioni in una rete di senso. Un Digital Garden nella sua forma più completa coinvolge tutto il sapere con cui lo studente entra in contatto a scuola e fuori (e sarebbe una gran cosa se diventasse oggetto di valutazione da parte di tutto il Consiglio di Classe).
3. La crescita incrementale: a differenza di una verifica tradizionale, il Digital Garden è un organismo vivo. La valutazione può quindi premiare la costanza nel tempo, osservando come una nota iniziale venga ripresa, corretta e arricchita man mano che il percorso didattico procede.

La valutazione, in quest'ottica, è un dialogo sul processo cognitivo, ossia una verifica *formativa* nel suo senso più autentico: l'insegnante non giudica solo il punto di arrivo, ma la qualità della *cura* che lo studente dedica al proprio ambiente di pensiero digitale.

# Vicus lusorum

## Giocare con le perle

Nel 1943, mentre l'Europa era in guerra, Hermann Hesse pubblicava *Il gioco delle perle di vetro* (*Das Glasperlenspiel*)<sup>1</sup>, il suo capolavoro, decisivo per il conferimento del premio Nobel tre anni dopo. Hesse immaginava una società in piena decadenza nella quale la cultura è custodita nella Castalia, una austera provincia pedagogica dedita a una disciplina raffinatissima: il Gioco delle perle di vetro, appunto. Esso consiste nel costruire connessioni tra elementi di tutte le discipline e di tutte le epoche – una fuga di Bach e un teorema matematico, un concetto della filosofia buddhista e una forma poetica medievale – intrecciandoli in strutture sempre più complesse che rivelano l'unità profonda del sapere.

*Vicus lusorum* – nel romanzo di Hesse il nome del quartiere dei giocatori – è una web app costruita per questo manuale, disponibile all'indirizzo <https://vicuslusorum.netlify.app>, che si ispira al gioco di Hesse pensando le singole note come *perle* da collegare ad altre in trame sempre più ricche. Mi piace ricordare che uno strumento per la realizzazione di ipertesti inteso come legame tra perle fu realizzato con il programma Textis, allegato al *Nuovo manuale di didattica multimediale* di Roberto Maragliano<sup>2</sup>. Nel disegnare questo semplice programma mi sono ricordato, con piacere, delle prime lontane esperienze con quel software.

---

<sup>1</sup>H. Hesse, *Il giuoco delle perle di vetro*, introduzione e trad. di Ervino Pocar, Mondadori, Milano 1955 e successive riedizioni.

<sup>2</sup>R. Maragliano, *Nuovo manuale di didattica multimediale*, Laterza, Roma-Bari 2002, settima edizione. Il programma è nel CD allegato al libro.

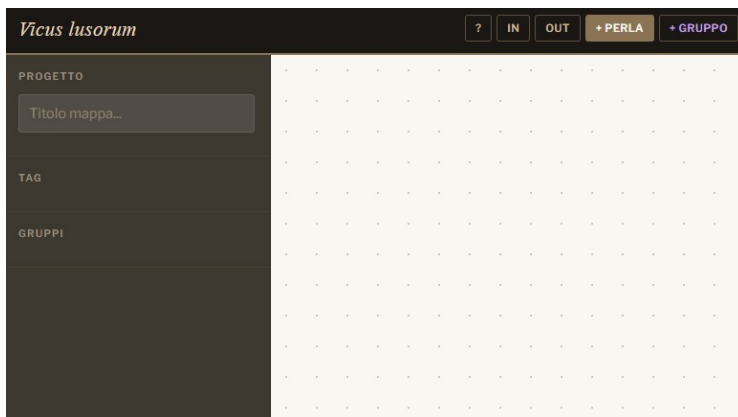


Figura 1

## Come funziona

Lo scopo di Vicus lusorum è di avvicinare gli studenti alla logica del Digital Garden con uno strumento facile da usare. Come Lexis, è un'applicazione che funziona interamente nel browser senza necessità di installazione né di account. L'interfaccia, volutamente sobria, mette al centro la tela di lavoro, uno spazio su cui le perle si dispongono e si collegano liberamente.

Ogni perla, che rappresenta un concetto filosofico, è un nodo della rete. Per crearla si clicca sul tasto *+ Perla* nella barra superiore. Si apre un pannello in cui inserire il titolo della perla, una nota testuale, uno o più tag separati da virgola, un colore scelto dalla tavolozza e un gruppo di appartenenza (creato in precedenza). Infine si seleziona *Aggiungi* per posizionare la perla al centro della tela.

Per selezionarla basta cliccarla: il pannello laterale entra in modalità modifica, mostrando titolo, nota, tag e gruppo. Una volta apportate le modifiche si clicca *Salva*. Cliccando su *Leggi* si apre una finestra di lettura a schermo pieno, comoda per note più estese.

Per creare un gruppo si clicca su *+Gruppo* nella barra superiore. Si sceglie il nome del gruppo e quindi si chiude la finestra selezionando *Aggiungi*. Il gruppo risulterà disponibile per la scelta quando verrà creata una nota. Le perle associate a un gruppo vengono inserite in un'area

grigia che le distingue dal resto della mappa ed è utile per individuare aree tematiche o disciplinari diverse all'interno della mappa.

Le perle si collegano tramite fili. Per creare un nuovo filo è sufficiente cliccare sulla prima perla e poi sulla seconda. Si apre una finestra nella quale è possibile specificare la natura del legame. Nella figura 2 abbiamo creato due perle. La perla rossa riguarda il concetto di *acatalessia* in Pirrone di Elide, mentre quella arancione il concetto di *avyākata* nel Buddhismo. Il legame indicato è di *#Possibile influenza*<sup>3</sup>. Dopo aver creato il filo, la mappa si presenta come nella figura 3. Nel menu laterale abbiamo l'elenco dei tag creati; cliccando su ogni tag le perle collegate ad esso saranno evidenziate.

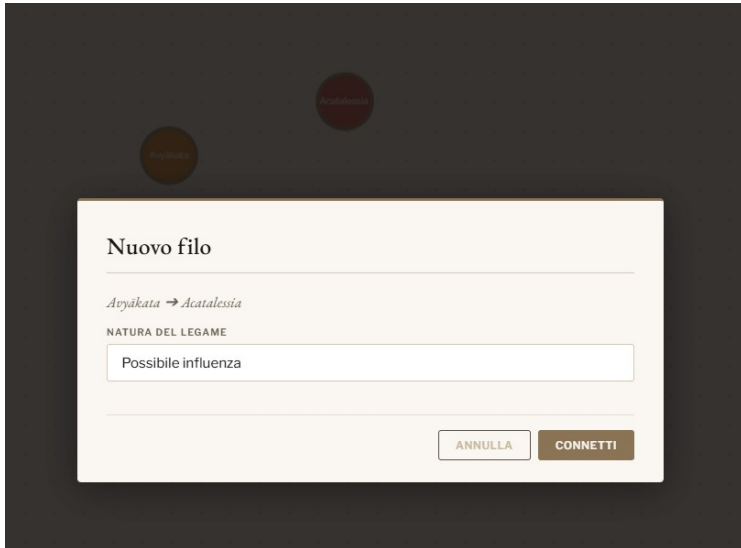


Figura 2

I progetti si salvano e si riaprono in formato JSON\* tramite i tasti *Out* e *In* nel menu in alto. Una volta caricato il file JSON\*, l'intera

---

<sup>3</sup>Secondo diversi studiosi lo Scetticismo di Pirrone di Elide – che come è noto è stato in India al seguito di Alessandro Magno – può essere stato influenzato dal movimento degli Śramaṇa e in particolare dal Buddhismo. Si veda A. Kuzminski, *Pyrrhonism: How the Ancient Greeks Reinvented Buddhism*, Lexington Books, Lanham 2008.



Figura 3

mappa viene ripristinata esattamente com'era. È importante ricordare di salvare prima di chiudere la pagina: il browser non conserva i dati tra una sessione e l'altra.

## Usi didattici

Vicus lusorum è pensato come strumento per una prima introduzione alla logica e alla pratica del Digital Garden. Si è scelto di privilegiare la facilità d'uso, mirando a una piattaforma con una curva d'apprendimento minima e con qualche inevitabile limite su progetti più complessi e a lungo termine.

### *Supporto alla lezione*

Un piccolo Digital Garden può essere un ottimo supporto all'attività didattica in classe. Un docente può lavorare su Platone con una mappa preparata in precedenza, nella quale i concetti fondamentali del filosofo sono isolati in perle, collegati ad altri e organizzati in aree (metafisica, politica, etica ecc.). Questo strumento consente di passare da una logica sequenziale ad una più complessa visione reticolare del sistema di idee platonico. Il docente può anche costruire la mappa dal

vivo, creando una nuova perla ogni volta che introduce un concetto e ragionando con gli studenti sui collegamenti tra quella perla e le altre. Il lavoro fatto in classe può poi essere consegnato agli studenti come file JSON\*, che potranno ricaricare a casa sulla piattaforma.

### *Lavoro di gruppo*

Vicus lurosum può essere usato in classe per le attività di gruppo, per aprire una pausa di riflessione dopo un segmento didattico. Si può assegnare ai gruppi di lavoro il compito di costruire piccoli Digital Garden dei temi studiati, identificando i concetti-chiave e collegandoli tra di loro. Un lavoro più complesso può svolgersi in direzione interdisciplinare. In che modo le perle filosofiche si incontrano con le perle storiche, letterarie, scientifiche, artistiche? È possibile costruire trame complesse? È bene lasciare la massima libertà agli studenti, senza costringerli nella gabbia ferrea della programmazione disciplinare; può essere che una perla di filosofia si colleghi con una perla artistica o scientifica di cui gli studenti sono venuti a conoscenza autonomamente e che non appartiene al periodo storico studiato. Queste connessioni sono assolutamente feconde e nel pieno spirito del *gioco* di Hesse.

### *Il Digital Garden personale*

Nulla impedisce di usare Vicus lusorum anche per un progetto a lungo termine come il Digital Garden personale, elaborato dallo studente nel corso dell'anno e consegnato a fine anno quale sintesi del suo lavoro di rielaborazione, ma per questo scopo sono più adatti gli strumenti che analizzeremo di seguito. Vicus lusorum ha la sola ambizione di essere uno strumento introduttivo, agile e di facile uso, che faccia da ponte verso strumenti più complessi.

# TiddlyWiki

## Tutto in una pagina

La prima versione di TiddlyWiki è stata pubblicata da Jeremy Ruston, sviluppatore britannico, nel 2004. L'idea era quella di uno strumento che consentisse di realizzare un wiki\* contenuto in un singolo file HTML\*, senza alcun bisogno di database o di installazione, una semplice pagina Internet che si apre nel browser, si modifica e si salva nel proprio computer. Un'idea che era nata da un'esigenza personale: Ruston cercava uno strumento per tenere traccia delle proprie note di lavoro in modo non lineare, senza dipendere da software proprietari o connessioni di rete. *Tiddly* è termine dello slang britannico che indica qualcosa di molto piccolo, quasi infinitesimale. TiddlyWiki intende essere un wiki\* per così dire tascabile; ma la leggerezza non deve ingannare: si tratta anche di un sistema potente per organizzare la conoscenza. Non sorprende che, più di venti anni dopo la sua creazione, risulti essere uno degli strumenti open source\* più longevi e attivi nel campo del Personal Knowledge Management\*, ancora in sviluppo attivo e con una comunità internazionale vivace.

La versione 5, riscritta da zero nel 2013, è quella oggi in uso: più veloce, più flessibile, con un sistema di plugin\* che ne estendono le funzionalità in quasi ogni direzione. Il codice sorgente è pubblico e può essere modificato e adattato da chiunque; esistono diversi progetti derivati, come TiddlyRoam (url: <https://tiddlyroam.org/>) o versioni con configurazioni particolari, come Stroll (url: <https://giffmex.org/stroll/stroll.html>).

## L'architettura di base

Prima di entrare nella guida operativa è utile capire la struttura concettuale dello strumento. L'unità di base di TiddlyWiki è il *tiddler*. Ogni

tiddler ha un titolo, che è anche il suo identificatore univoco, un contenuto testuale e un insieme di metadati: tag, data di creazione, data di modifica, e campi personalizzabili. A differenza di altri sistemi di Personal Knowledge Management\*, i tiddler non sono file separati, ma sono tutti contenuti nello stesso file HTML\* e vengono visualizzati e modificati nell'interfaccia del browser.

I link si creano scrivendo il titolo di un altro tiddler tra doppie parentesi quadre: `[[Il metodo maieutico di Socrate]]`. Se il tiddler esiste, il link è immediatamente cliccabile. Se non esiste, il link appare evidenziato in un colore diverso e cliccandoci si apre automaticamente un nuovo tiddler vuoto con quel titolo, pronto da compilare. È lo stesso meccanismo di Wikipedia, applicato alle proprie note personali.

I tag creano relazioni strutturali: un tiddler taggato con `Platone` diventa automaticamente parte di un elenco navigabile di tutti i tiddler su Platone. Possono essere usati per classificare il tipo di nota (concetto, domanda, connessione), l'autore, il periodo, il campo disciplinare.

*Story river* è il nome dato alla colonna centrale dell'interfaccia: i tiddler si aprono in sequenza in questa colonna, uno sopra l'altro, e possono essere tenuti aperti contemporaneamente, riordinati o chiusi individualmente. È un'interfaccia diversa da quella a schede di un normale browser, e richiede qualche ora di pratica perché diventi naturale.

La sidebar è il pannello laterale che dà accesso alla ricerca, all'elenco di tutti i tiddler, ai tag, alla cronologia di navigazione e alle impostazioni dello strumento.

## Diversi modi di usare TiddlyWiki

TiddlyWiki può essere usato in varie modalità, con caratteristiche e requisiti tecnici diversi. Capire la differenza è importante prima di introdurlo in classe.

- La versione web. TiddlyWiki offre la possibilità di usare lo strumento direttamente nel browser, senza scaricare nulla, salvando il garden su un account online attraverso la piattaforma TiddlyHost (url: <https://tiddlyhost.com/>).

- Il file locale. È la modalità originale e più fedele alla filosofia dello strumento. Si scarica un file HTML\* vuoto da <https://tiddlywiki.com/>, lo si apre nel browser e si comincia a lavorare.
- TiddlyDesktop. È una piccola applicazione gratuita, disponibile per Windows, Mac OS X e Linux, che fa da contenitore per TiddlyWiki sul desktop. Aprendo il file HTML\* con TiddlyDesktop invece che con il browser, il salvataggio avviene normalmente con Ctrl+S, senza scaricare copie. L'applicazione è disponibile per Windows, macOS e Linux, ed è scaricabile gratuitamente dall'url <https://github.com/TiddlyWiki/TiddlyDesktop>. Non richiede installazione complessa.
- App mobile. TiddlyWiki può essere gestito anche attraverso app per dispositivi mobili. Per dispositivi Android è disponibile Tiddloid (url: <https://f-droid.org/packages/top.donm.or.tiddloid/>), che può essere scaricato dalla piattaforma (alternativa a Google Play) F-Droid, mentre Quine, l'app per dispositivi Apple, non risulta al momento più disponibile.

Valutiamo la soluzione migliore per la scuola. Il problema principale di TiddlyWiki riguarda il salvataggio dei dati. I browser non consentono di salvare in automatico i dati, per cui utilizzando semplicemente il file HTML\* bisogna ogni volta salvare l'intero file. TiddlyDesktop è stato creato proprio per evitare questo inconveniente: aprendo il lavoro attraverso l'applicazione, i salvataggi avvengono semplicemente cliccando sul tasto apposito di TiddlyWiki. Questa è senz'altro la soluzione ideale. Bisogna tuttavia tener conto del fatto che gli studenti non sempre usano un computer e spesso nemmeno ne hanno uno in casa, mentre lo smartphone è posseduto da tutti, oltre ad essere un dispositivo a loro più familiare.

L'uso di TiddlyHost ha diversi vantaggi: gli studenti possono lavorarci sia da smartphone o da computer, i dati vengono salvati in remoto e sono sempre accessibili ovunque vi sia un computer collegato a Internet. Una studentessa, ad esempio, può accedere con le sue credenziali e mostrare il suo TiddlyWiki al docente in occasione di una verifica; ma il docente può anche monitorare la crescita del lavoro collegandosi all'indirizzo dei loro TiddlyWiki. La piattaforma però richiede, come è normale, un indirizzo e-mail e raccoglie altri dati, anche se è naturalmente

escluso qualsiasi uso commerciale dei dati e la Privacy Policy<sup>1</sup> specifica che i dati saranno protetti da "accesso non autorizzato, alterazione o distruzione", anche se "nessun metodo di archiviazione elettronica è sicuro al 100%"; inoltre i dati potrebbero essere "trasferiti o conservati su computer situati al di fuori del tuo stato, provincia, paese o altra giurisdizione governativa". Si tratta di un compromesso. La mancanza di qualsiasi profilazione, la possibilità di proteggere con una password i contenuti e il fatto che gli stessi restino proprietà degli utenti lo rende un compromesso accettabile. Occorrerà raccomandare agli studenti di evitare di inserire dati personali; non è necessario nemmeno usare il nome reale, potendo usare uno pseudonimo.

## TiddlyHost

Nel caso in cui si scelga questa soluzione, la studentessa potrà creare il suo TiddlyWiki collegandosi al sito TiddlyHost (url: <https://tiddlyhost.com/> e cliccando sul tasto verde *Sign up*. Nella schermata seguente sono richiesti email, nome (come detto è consigliabile non inserire il nome reale), username e password e quindi confermare con il tasto *Create account*. Si riceverà una e-mail con la richiesta di conferma della registrazione, dopo la quale sarà possibile fare il login con la username e la password scelti.

Per creare un nuovo TiddlyWiki si seleziona *+ Create*. Nella pagina che si apre inseriamo il nome del progetto, che può coincidere con l'username scelto dallo studente, selezioniamo *TiddlyWiki (self-contained)* e scegliamo le modalità di accesso. Le opzioni disponibili sono tre: *Private*, *Public* e *Public & searchable*. Nel primo caso il sito resta visibile solo alla studentessa, che potrà mostrarlo al docente solo su un suo dispositivo o da un computer scolastico (o LIM) dopo aver fatto il login con i suoi dati. Nel secondo caso il sito può essere visualizzato da chiunque abbia il link. In teoria è accessibile da chiunque, ma di fatto è altamente improbabile che qualcuno vi acceda se non ha ricevuto il link. Nel terzo caso invece il sito è a tutti gli effetti presente in rete e chiunque può rintracciarlo, anche se non è facile che un sito abbia una certa visibilità sui motori di ricerca senza determinate strate-

---

<sup>1</sup>Disponibile all'url: <https://tiddlyhost.com/privacy-policy>.

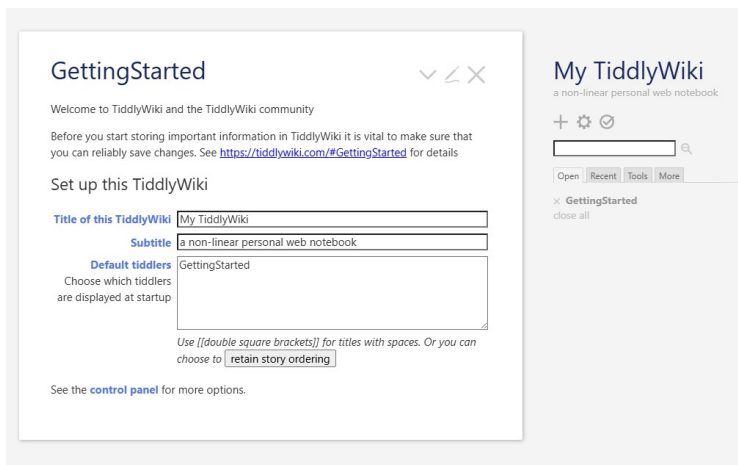


Figura 1

gie di SEO\*. La seconda opzione sembra preferibile perché consente al docente di seguire nel modo migliore il lavoro delle studentesse. È meglio escludere, invece, la terza possibilità, a meno che lo studente non sia maggiorenne e non desideri pubblicare il suo lavoro.

La descrizione del sito e i tag sono opzionali. Possiamo ometterli. Dopo aver cliccato su *Create* il sito sarà disponibile nella pagina <https://tiddlyhost.com/sites>. TiddlyHost consente di realizzare più di un sito. La piattaforma ha diversi piani: quello gratuito consente l'uso di tutte le funzionalità e la creazione di diversi siti; i piani a pagamento aggiungono altre funzionalità, come la possibilità di recuperare versioni precedenti del sito o di personalizzare l'url.

Cliccando sul nome del sito si entra nel TiddlyWiki, pronto per la personalizzazione.

## Prima configurazione

Quale che sia la soluzione scelta, al primo avvio su desktop ci si trova di fronte alla pagina della figura 1.

Possiamo subito impostare il titolo, sostituendo *My TiddlyWiki*, e il sottotitolo. Come si vede, il programma è in lingua inglese. Se

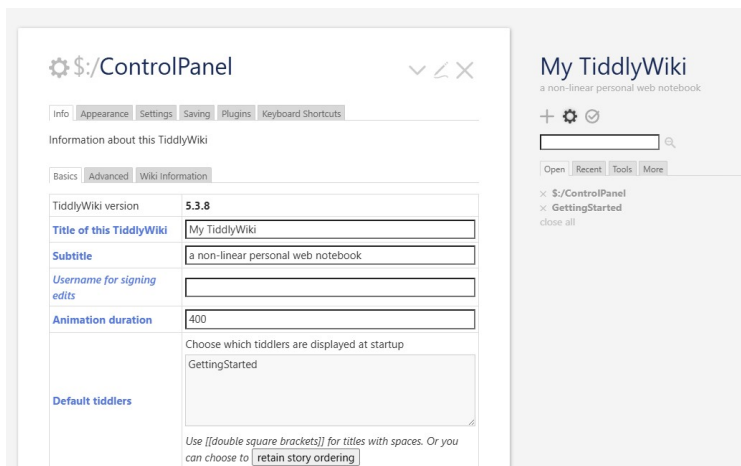


Figura 2

preferiamo, possiamo impostare la lingua italiana con un plugin\*. Come accennato, TiddlyWiki ha una grande quantità di plugin\*, che lo rendono altamente personalizzabile. Clicchiamo sull'icona dell'ingranaggio (una delle tre icone sotto al titolo) ed apriamo il *ControlPanel* (figura 2).

Qui selezioniamo *Plugins* (quinta scheda da sinistra) e poi il tasto blu *Get more plugins*. Si apre la finestra della figura 3.

Scorrendo troviamo la lingua italiana (it-IT). Clicchiamo su *Install* e chiudiamo la finestra. Per impostare la lingua italiana andiamo ancora nel *ControlPanel*. Scorrendo in basso troviamo la voce *Hello! Current language*. Dal menu a tendina possiamo selezionare la lingua italiana.

## Lavorare con i tiddler

Come anticipato, i tiddler<sup>2</sup> sono l'unità di base di TiddlyWiki. Nel sito del progetto sono presentati come segue:

I tiddler sono le unità fondamentali di informazione in TiddlyWiki. Funzionano meglio quando sono il più piccoli

<sup>2</sup>Il termine viene dall'inglese informale e indica un pesciolino.

Get more plugins

Official TiddlyWiki Plugin Library

## Official TiddlyWiki Plugin Library

<https://tiddlywiki.com/library/v5.3.8/index.html>

The official TiddlyWiki plugin library at tiddlywiki.com. Plugins, themes and language packs are maintained by the core team.

« Close plugin library

Updates (0) | Plugins (68) | Themes (9) | Languages (34)

Search:

>	<b>ar-PS:</b> العربية (فلسطين) \$:/languages/ar-PS 5.3.8	install
>	<b>ca-ES:</b> Català (Espanya) \$:/languages/ca-ES 5.3.8	install
>	<b>cs-CZ:</b> Čeština (Česká republika) \$:/languages/cs-CZ 5.3.8	install

Figura 3

possibile, in modo da poter essere riutilizzati intrecciandoli tra loro in modi diversi.<sup>3</sup>

Per creare un nuovo tiddler occorre cliccare sul pulsante *+* nella barra degli strumenti in alto. Si apre un nuovo tiddler (che in italiano è tradotto con *frammento*) in modalità modifica. Sostituiamo la scritta "Nuovo frammento" con il titolo della nostra nota, che dovrà essere descrittivo ed essenziale. Ad esempio: "Protagora e l'homo mensura". Nel box centrale inseriamo una descrizione essenziale del principio dell'*homo mensura*. Importanti sono i tag (etichette nella versione italiana), che vanno usati con intelligenza per recuperare le informazioni. In questo caso possiamo usare i tag *#Protagora*, *#relativismo* e *#verità*. La voce *Tipo di contenuto* ci consente di selezionare Markdown se abbiamo installato il plugin\*. La voce *Aggiungi un nuovo campo* ci consente di aggiungere al tiddler dei campi personalizzati, che possono essere concordati con il docente. Si può stabilire, ad esempio, di aggiungere un campo relativo alla fonte dell'informazione.

Una volta terminata la modifica si può salvare il tiddler cliccando sul segno di spunta in alto a destra.

<sup>3</sup>Url: <https://tiddlywiki.com/static/Tiddlers.html>.



Figura 4

Dopo aver creato il tiddler compariranno in alto a destra tre icone. Cliccando sulla seconda icona entriamo in modalità modifica; possiamo salvare le modifiche apportate ancora una volta con il segno di spunta in alto a destra. TiddlyWiki aggiorna automaticamente la data di modifica.

Per creare un collegamento tra tiddler occorre scrivere all'interno del box centrale del tiddler il titolo del tiddler di destinazione tra doppie parentesi quadre: ad esempio `[[Platone contro il relativismo]]` può essere un buon collegamento nel tiddler su Protagora. Il link diventa immediatamente cliccabile se il tiddler esiste, evidenziato in modo diverso se non esiste ancora. Nel secondo caso, cliccandoci si apre un tiddler vuoto con quel titolo, che invita a essere compilato.

## Navigare in TiddlyWiki

Man mano che il lavoro cresce, può essere difficile orientarsi. Gli strumenti per navigare il proprio giardino digitale sono diversi. Intanto la barra di ricerca nella sidebar, che esegue una ricerca in tempo reale su tutti i titoli e i contenuti. Inserendo il termine da ricercare, si apre un riquadro con due sezioni: *Corrispondenze titolo*, con l'elenco dei tiddler

che hanno il termine nel titolo, e *Tutte le corrispondenze*, con l'elenco dei tiddler che contengono il termine.

Un secondo strumento sono i tag-etichette. È possibile accedere a tutte le etichette attraverso la sidebar, selezionando *Altro* e poi *Etichette*.

Un terzo modo, più interessante, è *TiddlyMap*, un plugin\* per la visualizzazione grafica della rete di note che trasforma automaticamente i link [[. . .]] tra tiddler in archi visualizzabili su una mappa interattiva: ogni tiddler diventa un nodo ed ogni link una linea. La mappa è interattiva, è possibile zoomare, spostare i nodi, cliccare su un nodo per aprire il tiddler corrispondente. È scaricabile dall'url <https://tiddlymap.org/> e si installa trascinando il file del plugin\* nella finestra di TiddlyWiki.

## Salvare il lavoro

Come anticipato, con TiddlyWiki è importante fare attenzione al salvataggio del lavoro. Il pulsante per il salvataggio si trova nella sidebar: è un pulsante di colore rosso quando sono presenti modifiche da salvare, che diventa grigio e con un segno di spunta quando le modifiche sono state salvate (figura 5).



Figura 5

Con TiddlyDesktop ciò è sufficiente per salvare tutte le modifiche. Se invece non si usa TiddlyDesktop, ma il solo file .html, la situazione è più complessa e scomoda: in questo caso cliccando sul pulsante rosso verrà scaricata una nuova copia aggiornata del file, con un nome diverso. Se il file fosse nominato, ad esempio, `garden.html`, il nuovo file sarebbe `garden(1).html`.

## Organizzare il Digital Garden

L'unità minima del Digital Garden con TiddlyWiki è il tiddler-concetto. Un tiddler ad esempio può riguardare la concezione della *insocievole socievolezza* umana secondo Kant. Potremo associare a questo tiddler tre tag: #Kant, #natura umana, #società. Cliccando su questi tag li trasformiamo in nuovi tiddler: avremo così un tiddler-autore per Kant e due tiddler tematici, uno sulla natura umana e uno sulla società. Cliccando su *Altre azioni* (primo tasto da sinistra) si apre un menu verticale (figura 6).

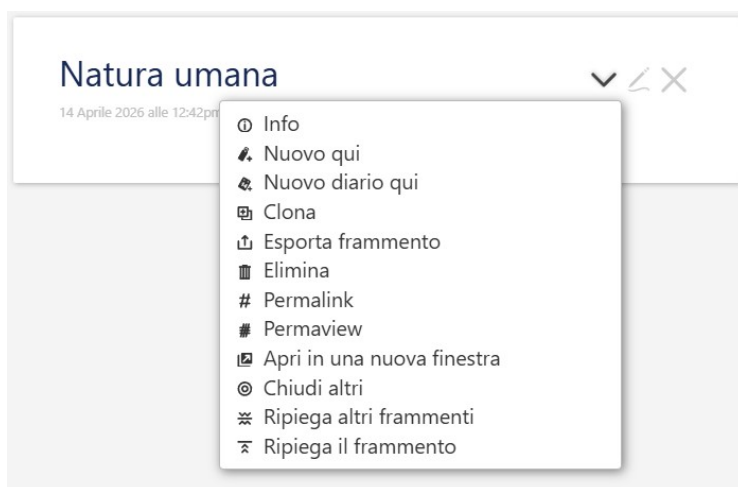


Figura 6

Selezionando *Info* si accede alla informazioni sul tiddler. Selezionando la terza voce, *Etichetta*, è possibile ottenere la lista di tutti i tiddler che hanno quel tag. È opportuno anche creare tag e tiddler corrispondenti generici, legati alle diverse branche della filosofia: etica, metafisica, estetica e così via. In ognuno di questi tiddler generici si possono inserire i link ai singoli tiddler. Il tiddler *Etica*, ad esempio, può avere link come

```
[[Fare e subire il male per Socrate]],  
[[L'importanza dell'amicizia per Aristotele]],  
[[Bene e piacere in Epicuro]].
```

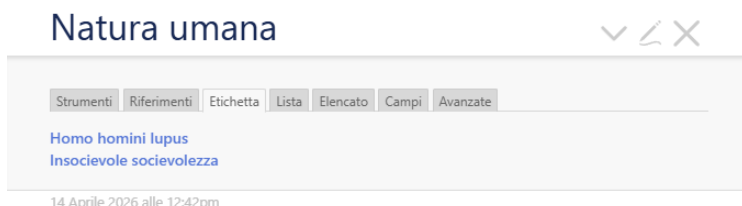


Figura 7

## Usare TiddlyWiki in classe

A differenza di Vicus lusorum, Tiddly Wiki è adatto più alla costruzione lenta, protratta nel tempo, di un Digital Garden personale che al lavoro in classe, personale o di gruppo. L'ideale sarebbe coltivarlo nel corso di più anni, usandolo come uno strumento per tornare spesso indietro, constatando come le idee di Spinoza, ad esempio, riprendano temi presenti nello Stoicismo, o il modo in cui un filosofo studiato al quinto anno risponde a domande che sono già state incontrate al terzo anno.

Introdurre TiddlyWiki in classe richiede una fase preparatoria, che richiederà più di qualche lezione, per imparare le funzioni di base. È bene resistere alla tentazione di mostrare subito tutte le funzionalità: bisognerà piuttosto spiegare alle studentesse che si tratta di uno strumento che ha molte potenzialità, che sarà possibile esplorare insieme con il tempo. Si raccomanda di passare a TiddlyWiki solo dopo che gli studenti hanno già fatto qualche pratica con Vicus lusorum e soprattutto dopo aver dedicato del tempo alla visione sottostante: il passaggio dagli appunti lineari alla creazione di percorsi concettuali non lineari, che si arricchiscono dialogando nel tempo.

# Logseq

## Obsidian e Logseq

Per comprendere Logseq è utile partire da Obsidian (url: <https://obsidian.md>), che negli ultimi anni è diventato uno degli strumenti più diffusi per la gestione della conoscenza personale e per la costruzione di archivi di note interconnesse. Il software, che può essere adoperato su ogni piattaforma, è stato sviluppato dalla società Dynalist Inc., fondata da Shida Li e Erica Xu, ed è stato rilasciato pubblicamente nel 2020. La sua particolarità, che è alla radice del suo crescente successo, è che non utilizza un database proprietario e non impone una struttura chiusa. È basato su note in formato Markdown salvate localmente sul proprio dispositivo; ogni nota resta perfettamente leggibile anche al di fuori di Obsidian. Questo garantisce un elevato grado di controllo sui propri dati e una notevole portabilità dei contenuti.

Come TiddlyWiki, Obsidian consente di creare collegamenti bidirezionali tra le note. Scrivendo il titolo di una nota tra doppie parentesi quadre, si genera un link che connette automaticamente i contenuti. Nel tempo, queste connessioni danno origine a una rete visualizzabile anche graficamente, che rappresenta il modo in cui i concetti si articolano e si richiamano reciprocamente.

La potenza di Obsidian è legata anche alla grande quantità di plugin\*, che ne ampliano le possibilità grazie al contributo di una comunità di sviluppatori. Anche se il software è gratuito (sono previsti piani a pagamento solo per la sincronizzazione, che è comunque possibile anche con strumenti alternativi, e la pubblicazione delle note in un sito Internet), tuttavia non è un software open source\*. Il codice dell'applicazione è proprietario e controllato dall'azienda che lo sviluppa. Questo significa che, pur mantenendo il controllo sui propri dati grazie ai file locali, l'utente non ha accesso al funzionamento interno del programma né la possibilità di modificarlo. Questa è la ragione per la quale in

questo manuale, pur consigliando l'*esplorazione* di Obsidian, preferisco approfondire Logseq. Come vedremo, non è in realtà l'unica.

## Caratteristiche di base di Logseq

Nel febbraio 2020 il programmatore Tienson Qin ha avviato un progetto personale per connettere le proprie note di lavoro in modo non lineare che ha molti aspetti in comune con Obsidian, pur essendo da esso indipendente. Da questo lavoro è nato Logseq (url: <https://logseq.com/>). Nel 2022, dopo che una comunità internazionale di utenti e sviluppatori si era raccolta attorno al progetto, Tienson Qin fonda Logseq Inc. insieme a tre collaboratori incontrati proprio nella comunità open source\*. L'azienda raccoglie più di quattro milioni di dollari.

Alla base di Logseq, che come detto è un progetto open source\* rilasciato con licenza AGPL-3.0\*, sono tre caratteristiche: i blocchi, il grafo e le note giornalieri. Ogni nota è composta da blocchi, ossia unità di testo che possono essere annidate, referenziate, spostate, collegate ad altri blocchi indipendentemente dalla pagina in cui si trovano. Il grafo rende visivamente il rapporto tra le note: ogni cerchio nel grafo rappresenta una pagina; più grande è il cerchio, più collegamenti riceve o invia quella pagina (è un "hub" di informazioni). I collegamenti si creano ogni volta che si usano le doppie parentesi quadre [[Titolo Pagina]] o i tag. Il grafo si aggiorna costantemente, se si aggiunge un link in una nota si vede fisicamente apparire una linea che collega i due punti in tempo reale.

Appena si apre Logseq compare una nota con la data del giorno corrente. È la nota giornaliera, o Journal, che naturalmente può essere adatta a tenere un vero e proprio diario o a pianificare il proprio lavoro o a tenere traccia delle proprie letture o dei propri pensiero. La particolarità delle note giornalieri è che è possibile generare facilmente link ad altre note, che vengono create appositamente. Ad esempio uno studente può annotare i propri appunti della lezione sul [[sublime]] in [[Kant]] e i collegamenti con l'[[arte romantica]]. Ognuna di queste voci diventa una nuova nota, aprendo la quale si avrà la lista delle note che fanno riferimento ad essa.

## Esploriamo la versione demo

Logseq può essere scaricato gratis dal sito web del progetto<sup>1</sup>, nel formato adatto al proprio sistema operativo. Sono disponibili anche app per i dispositivi mobili, sia Android che Apple.

Entrando nel sito si è accolti da una schermata con una riproduzione del *Pensatore* di Rodin, il motto "Connect your notes, increase understanding" e due pulsanti: uno per il *Download* nel proprio sistema operativo e uno per la *Live Demo* (figura 1).

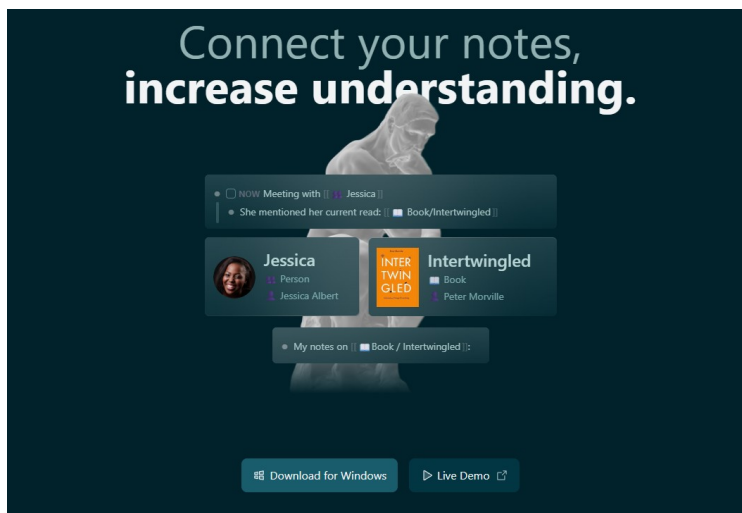


Figura 1

Quest'ultima possibilità per noi è preziosa: possiamo esplorare liberamente lo strumento, passando all'installazione quando abbiamo già una certa pratica (o rinunciandovi se ci sembra che non faccia per noi). Entriamo dunque nella versione demo (url: <https://demo.logseq.com/>). Ci troviamo di fronte la schermata dalla figura 2.

---

<sup>1</sup>Url: <https://logseq.com/downloads>.



Questo è un grafo dimostrativo, le modifiche non saranno salvate finché non aprirai una cartella locale.

Apr 15th, 2026

### Ciao, benvenuto su Logseq

- Logseq è una piattaforma con la *privacy come priorità*, [open-source](#) per la gestione della conoscenza e la collaborazione.
- Questo è un tutorial di 3 minuti su come utilizzare Logseq. Iniziamol
- Ecco alcuni suggerimenti che potrebbero essere utili.



Fare clic per modificare qualsiasi blocco.  
Digita "Invio" per creare un nuovo blocco.  
Digita `Maiusc+Invio` per creare una nuova riga.  
Digita `/` per mostrare tutti i comandi.

- Creiamo una pagina chiamata `| Come prendere appunti fittizi? |`. Puoi fare clic su di esso per andare a quella pagina, oppure puoi "Maiusc + clic" per aprirlo nella barra laterale destra! Ora dovresti vedere sia *Riferimenti collegati* che *Riferimenti non collegati*.
- Facciamo riferimento ad alcuni blocchi su `| Come prendere appunti fittizi? |`, puoi fare `Maiusc+Clic` su qualsiasi riferimento di blocco per aprirlo nella barra laterale destra. Prova a fare alcune modifiche sulla barra laterale destra, verranno modificati anche quei blocchi di riferimento!
  - Ciao, io sono un blocco! : Questo è un riferimento ad un blocco.
  - Hey, io sono un altro blocco! : Questo è un altro riferimento ad un blocco.

Figura 2

La pagina ha creato una nota giornaliera, in questo caso del 15 aprile 2026 (la data in cui sto scrivendo questo paragrafo), con un semplice tutorial. Come si vede, la pagina è organizzata in punti-elenco: sono i blocchi.

Se clicchiamo su "Fare clic per modificare qualsiasi blocco", compare il seguente testo:

`#+BEGIN_TIP`

Fare clic per modificare qualsiasi blocco.

Digita "Invio" per creare un nuovo blocco.

Digita ``Maiusc+Invio`` per creare una nuova riga.

Digita ``/`` per mostrare tutti i comandi.

`#+END_TIP`

Il testo è entrato in modalità modifica ed è comparso il codice. Normalmente Logseq funziona in modalità WYSIWYG\*: il testo appare immediatamente formattato (corsivo, grassetto eccetera), ma cliccando sul testo si accede al codice. Qui ci troviamo di fronte a un blocco particolare, che utilizza la sintassi `org-mode*` per mostrare un riquadro

per dei suggerimenti. Possiamo ignorare questo tipo di sintassi: Logseq utilizza normalmente il Markdown. Se passiamo il mouse sulla scritta `[[Come prendere appunti fittizi?]]` compare un'anteprima della nota collegata; possiamo aprirla cliccando sul testo oppure, se preferiamo, aprirla a lato, senza lasciare la pagina, usando il tasto sinistro del mouse insieme al tasto Maiusc sulla tastiera. Se clicchiamo su *Ciao, io sono un blocco! Questo è un riferimento a un blocco*, compare il seguente codice:

```
((5f713e91-8a3c-4b04-a33a-c39482428e2d)) : Questo è un riferimento ad un blocco.
```

Il lungo codice alfanumerico è l'identificativo del singolo blocco, che consente di riutilizzarlo in altre note. Il resto del testo ci informa che possiamo utilizzare dei tag (e se clicchiamo sul tag di esempio si apre una pagina con i riferimenti collegati) e che possiamo creare facilmente delle liste di cose da fare (todo) digitando uno slash (/) e scegliendo come parola chiave *todo*. In realtà provando a digitare lo slash scopriamo una cosa più interessante: si apre un menu che ci offre diverse opzioni (figura 3).

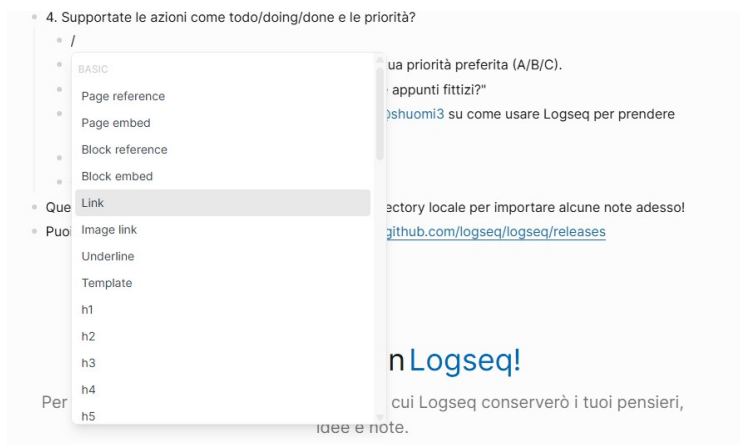


Figura 3

Le voci di base ci consentono di creare rapidamente collegamenti ad altre pagine o a singoli blocchi oppure a incorporarli nella pagina

attuale, di inserire immagini o intestazioni e così via. Sono presenti anche funzioni particolarmente avanzate, come l'integrazione con Zotero, che riguardano chi fa un uso particolarmente avanzato del programma.

Fino ad ora abbiamo visto come lavorare sul testo; scorrendo in basso la pagina vediamo come entrare nel programma vero e proprio, sempre senza installare nulla.

## Il menu

Dopo averci dato il benvenuto, Logseq ci invita a scegliere una cartella sul nostro computer. Possiamo dunque creare una cartella, che chiameremo *Digital Garden*, e la selezioniamo cliccando su *Scegli una cartella*. Da questo momento in poi tutto il lavoro sarà salvato nella cartella nel nostro computer. Questo vuol dire che, se preferiamo non installare nulla, possiamo lavorare dal browser, così come possiamo salvare la cartella su una chiavetta USB e utilizzarla da un altro computer, come quello scolastico.

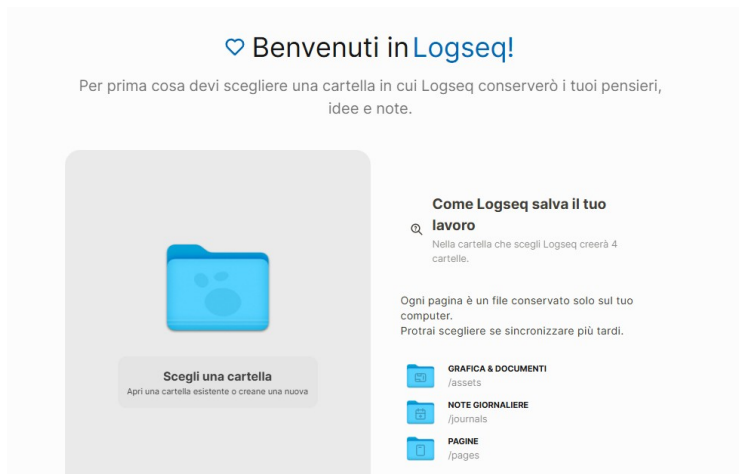


Figura 4

Ci troviamo di fronte a una pagina bianca, con al centro la data odierna, che può risultare spiazzante. È una scelta minimalista per non distrarre dalla scrittura. Guardando meglio, vediamo che sia a destra

che a sinistra sono presenti delle icone. Cliccando sull'icona con le tre linee verticali compare il menu della figura 5.

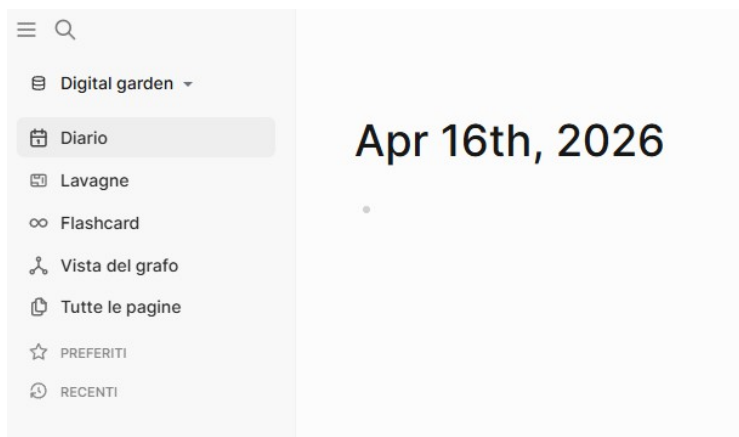


Figura 5

La voce *Diario* ci dà accesso alle note del Journal, in ordine cronologico. Le *Lavagne* (o *Whiteboards*) sono uno degli strumenti più interessanti di Logseq. Sono uno spazio di lavoro visivo e infinito che permette di organizzare le note in modo libero, uscendo dalla struttura rigida dell'elenco puntato; una via di mezzo tra una mappa concettuale e una lavagna infinita, ma con il vantaggio di essere integrata direttamente nel database di note locali. Non meno utili sono le *Flashcards*, sulle quali è bene soffermarsi.

Esistono diversi programmi o app, come Anki (url: <https://apps.ankiweb.net/>), per favorire l'apprendimento basato sulla tecnica della ripetizione spaziata (*Spaced Repetition*) e del richiamo attivo (*Active Recall*). È una tecnica molto usata nel campo dell'apprendimento delle lingue, ma molto utile anche in altri campi disciplinari. In sostanza si creano dei veri e propri mazzi di carte, ognuna delle quali presenta un *fronte*, con il termine o il concetto, e un *retro*, con la traduzione o spiegazione. Le carte si ripropongono a intervalli regolari, con variazioni legate al processo di apprendimento: i termini o concetti che ormai conosciamo si presenteranno meno di frequente, quelli più ostici saranno più frequenti.

In Logseq le flashcards non sono file separati, ma vengono create direttamente all'interno delle note quotidiane o delle pagine. La creazione di una flashcard è semplice. In un blocco, scriviamo la definizione o domande, seguita da #card. Ad esempio: - Estetica #card

Dobbiamo creare un blocco-figlio, che sarà il retro della carta. Andiamo a capo dopo il tag e indentiamo il testo premendo il tasto Tab. Scriviamo quindi la definizione o risposta:

- Estetica #card
  - La branca della filosofia che si occupa della bellezza e dell'arte.

In questo modo abbiamo creato una carta che mostra inizialmente la definizione e poi, dopo aver cliccato *Mostra risposta*, la schermata della figura 6, nella quale possiamo dire al programma se abbiamo ricordato facilmente la risposta o è stato più difficile. Tutte le carte prodotte sono disponibili nel menu *Flashcard*. Lo studente potrà usarle per crearsi un dizionario personale della filosofia, ma anche per ricordare i concetti chiave in vista di una verifica.

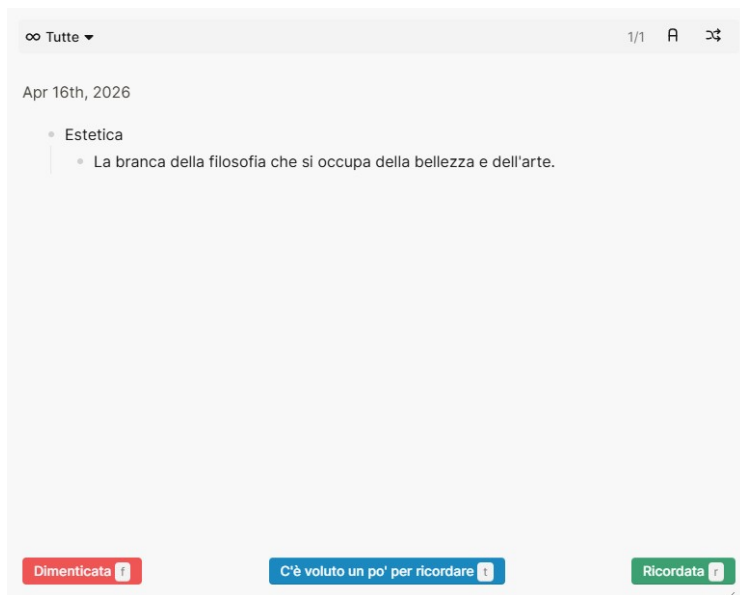


Figura 6

Scorrendo il menu troviamo ancora la *Vista del grafo* e *Tutte le pagine*. Quest'ultima voce ci restituisce l'elenco di tutte le pagine create, ordinate in base al nome del file, con l'indicazione dei *Collegamenti a ritroso* (*Backlinks* in inglese), ossia delle altre pagine che rimandano a quella pagina, le date di creazione e di modifica di ogni pagina e uno strumento di ricerca nelle pagine che non si limita al titolo, ma cerca il testo anche nel contenuto. Si tratta di uno strumento fondamentale per muoversi nel Digital Garden, soprattutto quando diventa complesso, che si affianca alla *Vista del grafo*.

Guardiamo ora in alto a destra. Troviamo altre due piccole icone: una con tre puntini orizzontali e una con un layout stilizzato (figura 7).

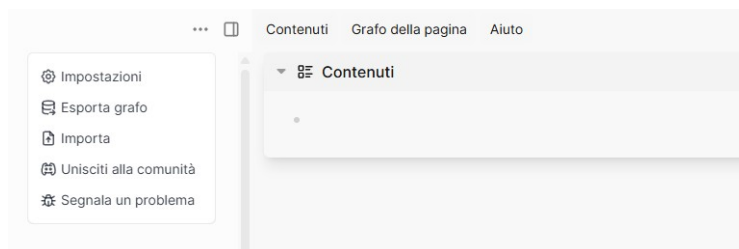


Figura 7

Cliccando sui tre puntini orizzontali si apre un menu verticale con diverse voci. La prima, *Impostazioni*, sarà fondamentale man mano che prenderemo familiarità con il programma e sentiremo il bisogno di personalizzarlo, anche dal punto di vista dell'aspetto (Figura 8). Intanto possiamo settare la lingua, che di base è l'inglese, scegliendo l'italiano (nella versione demo il menu è comunque già in italiano). Sotto la voce *Funzionalità* possiamo anche disattivare le voci che non ci interessano, eliminando ad esempio il *Diario*, le *Lavagne* o le *Flashcard*.

Le altre voci del menu della figura 7 ci consentono di esportare il grafo in diversi formati e di importarlo da diversi formati. Osservando la figura, notiamo anche la colonna che si è aperta cliccando sul layout stilizzato. C'è una prima voce di *Contenuti*. È una pagina normale del graph che ha la proprietà speciale di essere sempre visibile nella sidebar destra, una sorta di bacheca globale in cui l'utente mette quello che vuole: link rapidi a pagine importanti, note fisse o un indice manuale

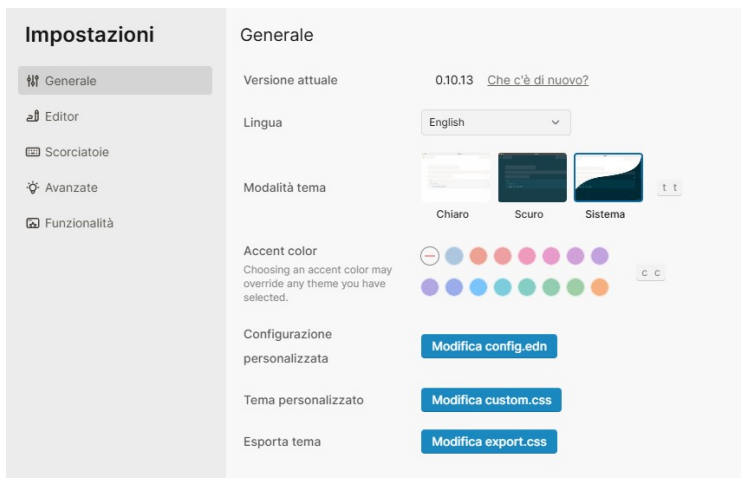


Figura 8

del Garden. Dalla voce accanto si accede al *Grafo della pagina*, di cui ci occuperemo tra un attimo.

In basso a sinistra notiamo ancora il pulsante *+ Crea*. Cliccando abbiamo la possibilità di creare una nuova pagina o una nuova lavafna. Cliccando su *Nuova pagina* ci troviamo di fronte alla finestra della Figura 9. Il programma ci chiede in inglese (non tutte le voci sono state ancora tradotte) cosa cerchiamo. Possiamo digitare semplicemente il titolo della pagina che vogliamo creare e selezionare *Create page*.

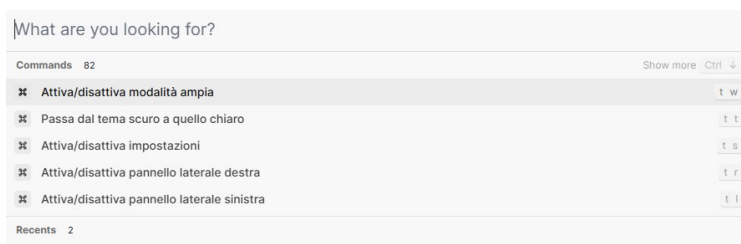


Figura 9

## Il grafo

Logseq dispone di due viste grafo: una globale e una locale. Alla prima si accede attraverso la voce *Vista del grafo* del menu di sinistra e ci offre una panoramica generale del Digital Garden.

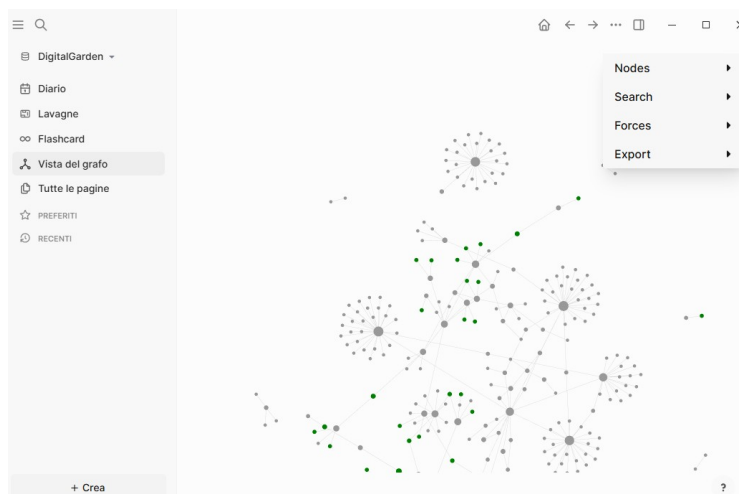


Figura 10

La visualizzazione, che può essere modificata attraverso il menu laterale (ad esempio ampliando la distanza tra i nodi), è *force-directed*: i nodi (le pagine) si posizionano nello spazio in base alle loro connessioni reciproche. Più due pagine sono collegate tra loro – direttamente o tramite pagine intermedie – più tendono a gravitare vicine, senza che l'utente possa imporre una sua.

Nella figura 10 vediamo il grafo globale di un giardino ancora in fase embrionale, con poco più di trecento link. I singoli nodi sono dei semplici puntini. È possibile ingrandire la mappa con lo scroll del mouse: compariranno così nomi delle singole pagine, che è possibile cliccare per accedervi. Notiamo alcuni *cluster*, pagine che si linkano reciprocamente formano gruppi visivamente compatti. Se il grafo è una nuvola uniforme senza nodi dominanti, significa che non ci sono ancora concetti integrativi; se è un arcipelago di isole separate, come nella

figura, significa che il lavoro è avvenuto per compartimenti stagni senza connessioni trasversali.

Come già accennato, dalla sidebar di destra della pagina si accede al *Grafo della pagina*, una visualizzazione che mostra solo quel nodo e i nodi direttamente collegati. Quando il Digital Garden supera una certa grandezza, può diventare poco leggibile, mostrandosi come una massa quasi indistinta. Occorre inoltre considerare che la vista grafo mostra la presenza di collegamenti tra le note, ma non la qualità del contenuto; in teoria è possibile creare un grafo estremamente complesso ma perfettamente vuoto. Per questo è importante, ai fini della valutazione, navigare con attenzione il grafo, tenendo conto anche del grafo locale, che mostra le connessioni delle singole pagine e che mostra dunque le scelte fatte di volta in volta dalla studentessa, e che possono essere oggetti di un confronto in classe.

## **L'uso didattico di Logseq**

Logseq è uno strumento che richiede una curva di apprendimento significativa, e questo può scoraggiare. Tuttavia i molteplici vantaggi che offre dovrebbero spingere a insistere. Vediamo come introdurlo in classe.

In una prima fase Logseq può essere usato dal docente, anche senza installare nulla. Si può usare la versione demo che conosciamo, collegarla a una cartella locale nel computer di classe e cominciare a creare un Digital Garden di classe. Ogni giorno il docente aprirà la pagina di Diario, nella quale il docente metterà una sintesi degli argomenti trattati. Una parte della lezione può essere dedicata a creare pagine con i concetti essenziali che sono stati trattati. La pagina di Diario dunque funziona un po' come il registro del docente, ma non si limita a indicare gli argomenti trattati, bensì presenta dei link che rimandano a pagine in cui sono fissati i concetti-chiave. E ognuna di queste pagine ha dei collegamenti ad altre pagine, che creano giorno dopo giorno un grafo complesso, soprattutto se il docente ha cura di mostrare i collegamenti anche con altre discipline (o, magari, se il Digital Garden è aperto al contributo di tutti i docenti).

In questo caso Logseq è un supporto piuttosto raffinato all'attività didattica in classe; il momento decisivo è quello della creazione delle

pagine *insieme agli studenti*, che diventa una fase di rielaborazione e di fissazione della conoscenze. Un limite è legato alla impossibilità, da parte degli studenti, di accedere al Digital Garden da casa.<sup>2</sup>

Se le circostanze lo consentono – se cioè la classe ha a disposizione tablet o computer portatili, essendo vietato l'uso degli smartphone<sup>3</sup> – è preferibile che Logseq sia usato dalle studentesse per la creazione del proprio Digital Garden. Lo strumento risulta prezioso soprattutto in fase di valutazione, perché consente di tenere traccia dello sviluppo del pensiero, considerando la capacità dello studente di collegare, rivedere e approfondire.

---

<sup>2</sup>Logseq consente di esportare un grafo come sito statico, attraverso un apposito plugin\*; il sito può essere poi pubblicato su servizi di hosting gratuito, come Netlify. Ma non si chiederà realisticamente a una docente di fare questo lavoro in classe.

<sup>3</sup>La circolare del MIM *Disposizioni in merito all'uso degli smartphone nel secondo ciclo di istruzione* del 16 giugno 2026 vieta l'uso degli smartphone anche nella scuola secondaria di secondo grado; resta però consentito l'uso dei tablet.

### **III Raccontare storie**

# Pensare, narrare, decidere

## Il valore epistemico della narrazione

Siamo a bordo di un tram impazzito e abbiamo il controllo di un interruttore che può deviarne il percorso. Sul binario sono legate cinque persone; possiamo deviare il tram su un secondo binario, sul quale è presente una persona legata e ugualmente impossibilitata a muoversi. Ci troviamo di fronte a due sole possibilità: lasciare che il tram faccia il suo corso o deviarne il tragitto e ucciderne una sola.

Si tratta del noto *trolley problem*, formulato originariamente da Philippa Foot nel 1967<sup>1</sup> e che per l'efficacia come attivatore della riflessione etica è molto usato nella didattica della filosofia. Perché funziona? Perché è una storia: ha dei personaggi, una situazione, una tensione. Ma è un tipo particolare di storia, di fronte al quale non siamo semplici ascoltatori, ma siamo chiamati a immedesimarci e a prendere una decisione. La situazione può essere resa più drammatica cambiando le variabili: può essere, ad esempio, che la persona sola legata sul binario alternativo sia una persona che amiamo. In questo caso che faremmo? Sacrificherebbero la vita di una persona che amiamo per salvare la vita a cinque persone sconosciute? Come motiveremo la scelta? Quali valori sono in gioco? Quali criteri di giudizio?

È chiaro, da questo esempio, che in filosofia il racconto, anche quando è mitico (come in Platone)<sup>2</sup>, è uno strumento epistemico potente, un modo di accedere a questioni che la formulazione astratta fatica a catturare e a collegarli alla nostra vita concreta.

---

<sup>1</sup>Ph. Foot, *The Problem of Abortion and the Doctrine of the Double Effect*, in *Virtues and Vices*, Basil Blackwell, Oxford 1978.

<sup>2</sup>Si pensi, per richiamare un altro esperimento mentale che funziona molto nella didattica dell'etica, al mito dell'anello di Gige, raccontato da Platone nel secondo libro della *Repubblica*. Che faremmo noi se venissimo in possesso di un anello in grado di renderci invisibili?

Questa parte del manuale è dedicata agli strumenti digitali che permettono di creare e usare storie interattive in classe. Prima di presentare gli strumenti – Twine e Ink – ci soffermeremo sul fondamento teorico e pedagogico dell'uso del racconto, e in particolare del racconto interattivo, nell'insegnamento della filosofia.

## Bruner e il pensiero narrativo

Nel 1986 lo psicologo Jerome Bruner pubblica *Actual Minds, Possible Worlds*<sup>3</sup>, un testo che ha avuto un'influenza profonda sia sulla psicologia cognitiva sia sulla pedagogia. La tesi centrale del libro è che la mente umana dispone di due modi fundamentalmente diversi di organizzare l'esperienza e costruire la realtà: il pensiero logico-scientifico o pradigmatico e il pensiero narrativo.

Il pensiero paradigmatico aspira all'universalità e si esprime attraverso concetti, categorie, leggi, dimostrazioni. Il suo criterio di validità è la verità: un'affermazione è accettabile se è verificabile, coerente con le premesse, derivabile secondo regole condivise. È il pensiero della matematica, della logica, delle scienze naturali ed ha una importanza fondamentale anche in filosofia.

Il pensiero narrativo funziona in modo radicalmente diverso. Non cerca la verità universale ma il significato particolare. Si esprime attraverso storie: sequenze di eventi che coinvolgono agenti dotati di intenzioni, che si trovano in situazioni specifiche, che agiscono e subiscono conseguenze. Il suo criterio non è la verità logica ma la verosimiglianza, ossia la capacità della storia di rendere intelligibile un'esperienza umana possibile.

Per Bruner questi due modi non sono gerarchicamente ordinati: il pensiero narrativo non è una versione degradata o infantile del pensiero logico. Sono due modi di pensare complementari e irriducibili l'uno all'altro. Non si può *tradurre* la storia di Antigone in un sillogismo senza perdere proprio quello che la storia dice, né si può ridurre la crisi di Amleto a una questione logica. Le questioni che riguardano il senso dell'esistenza, la responsabilità morale, la giustizia nelle relazioni umane richiedono il pensiero narrativo, e non perché siano meno serie, ma

---

<sup>3</sup>Edizione italiana: *La mente a più dimensioni*, Laterza, Roma-Bari 1993.

perché hanno una struttura diversa. E per questa ragione il pensiero narrativo ha a che fare con la filosofia, nella misura in cui essa ha a che fare con le questioni centrali della vita umana.

## Storie che pensano

La storia della filosofia mondiale è anche una storia letteraria, e non solo perché molti testi filosofici hanno un grande valore letterario (mentre molti altri risultano quasi illeggibili a chi non sia in qualche modo *iniziato*), ma anche perché il pensiero sembra non riuscire a fare a meno delle storie, dai celebri ed efficacissimi miti platonici alle storie che costellano i sutra buddhisti o i koan dello Zen, che impongono il superamento della logica corrente, alle storie dei maestri sufi, che procedono anch'esse sfidando la ragione per aprirla all'impensato.

Un caso filosoficamente molto ricco è quello delle storie che costruiscono un mondo possibile per testare un'idea. Ibn Tufayl, nel romanzo filosofico *Hayy ibn Yaqzan* (XII secolo), immagina un bambino cresciuto senza linguaggio né società su un'isola deserta, per verificare narrativamente se la ragione umana possa raggiungere da sola la conoscenza e la verità; un esperimento mentale che ha influenzato non poco il pensiero europeo. La modernità europea è segnata da narrazioni filosofiche che si muovono tra l'utopia e l'esperimento mentale, dall'isola di Utopia in Moro (*Utopia*, 1516) e la Città del Sole di Tommaso Campanella (1623) al contratto sociale di Rousseau con i suoi uomini allo stato di natura (*Discorso sull'origine della disuguaglianza*, 1755) fino al *Voyage en Icarie* di Etienne Cabet (1842) e la posizione originaria di Rawls con il velo di ignoranza (*A Theory of Justice*, 1971). Sono storie che funzionano perché portano il lettore in una situazione ipotetica, chiedendogli di ragionare dall'interno di un mondo costruito apposta.

Lo stesso vale per i casi limite della filosofia analitica – il violinista di Judith Jarvis Thomson (*A Defense of Abortion*, 1971)<sup>4</sup>, il già citato *trolley problem* di Philippa Foot o la stanza cinese di Searle<sup>5</sup> – che

---

<sup>4</sup>J. Thomson, *A Defense of Abortion*, in *Philosophy and Public Affairs*, 1:1 (Autumn 1971), pp. 47–66.

<sup>5</sup>J. R. Searle, *Minds, Brains, and Programs*, in *Behavioral and Brain Sciences*, vol. 3, 1980.

nascono come racconti brevi e costruiscono un'intuizione prima che il lettore abbia il tempo di mobilitare le proprie difese teoriche.

A partire dagli anni Ottanta diversi filoni del pensiero contemporaneo hanno messo la storia al centro della riflessione sul soggetto, sull'etica e sulla conoscenza. Ricoeur ha mostrato che l'identità personale è una trama piuttosto che una sostanza<sup>6</sup>; MacIntyre ha sostenuto che le scelte morali sono intelligibili solo all'interno di narrazioni condivise<sup>7</sup>; Nussbaum ha difeso la letteratura come strumento di educazione morale irriducibile alla teoria<sup>8</sup>. Questa svolta narrativa attraversa la psicologia, l'antropologia, la storiografia e perfino la medicina, e sul versante pedagogico ha trovato in Bruner, come abbiamo visto, la sua formulazione più influente.

È in riferimento a questo quadro che nella didattica della filosofia risultano preziose le storie interattive. Se il pensiero filosofico ha sempre usato le storie come laboratorio del possibile, una storia in cui il lettore compie scelte reali e l'esito dipende dal suo ragionamento aggiunge una dimensione che la narrazione tradizionale non possiede: la responsabilità. Chi legge la parabola del buon samaritano è spinto a valutare in astratto il comportamento dei personaggi, mentre chi naviga una storia interattiva su un dilemma etico sceglie, e deve fare i conti con le conseguenze di quella scelta.

## Una breve storia delle storie interattive

Le storie interattive, ossia racconti in cui il lettore o il giocatore compie scelte che determinano l'andamento della narrazione, non sono un'invenzione digitale, anche se la forma digitale consente una interazione difficile da ottenere con la tradizionale forma del libro. I libri-gioco (in inglese gamebook o choose-your-own-adventure) appaiono nella forma riconoscibile negli anni Sessanta e Settanta. La serie per bambini

---

<sup>6</sup>P. Ricoeur, *Soi-même comme un autre*, du Seuil, Paris 1990; edizione italiana: *Sé come un altro*, Jaca Book, Milano 1993.

<sup>7</sup>A. MacIntyre, *After Virtue: A Study in Moral Theory*, University of Notre Dame Press, Notre Dame, Indiana 1981; edizione italiana: *Dopo la virtù. Saggio di teoria morale* Armando, Roma 2007

<sup>8</sup>M. Nussbaum, *Love's Knowledge: Essays on Philosophy and Literature*, Oxford University Press, Oxford 1990.

*Choose Your Own Adventure*, pubblicata dalla Bantam Books a partire dal 1979, vende più di 250 milioni di copie in tutto il mondo. Il piccolo lettore è invitato a fare delle scelte, ognuna delle quali conduce a una pagina diversa e determina un certo percorso della storia.

Con l'avvento del computer le storie interattive trovano un ambiente naturale. La fiction interattiva (IF) nasce negli anni Settanta con *Colossal Cave Adventure* (1976), creato da Will Crowther e poi esteso da Don Woods: un gioco testuale in cui si esplora un sistema di caverne attraverso comandi scritti in linguaggio naturale. Da lì si sviluppa un filone commerciale che, con la casa produttrice Infocom e titoli come *Zork* (1980) e *The Hitchhiker's Guide to the Galaxy* (1984, scritto insieme a Douglas Adams), raggiunge un pubblico di massa e inizia ad attrarre autori provenienti dalla letteratura. Quando nei primi anni Novanta il mercato commerciale si sposta verso i giochi grafici, la IF non scompare: una comunità indipendente ne eredita l'ambizione narrativa, sviluppa strumenti aperti come Inform (1993), un linguaggio di programmazione e sistema di sviluppo creato da Graham Nelson, e istituisce competizioni annuali come IFComp (1995), producendo nel tempo migliaia di testi.

Negli anni Ottanta e Novanta la fiction interattiva si divide progressivamente in due filoni. Il filone ludico porta ai videogiochi d'avventura e ai giochi di ruolo digitali. Il filone letterario resta fedele all'ambizione narrativa: storie in cui le scelte del lettore hanno un peso drammatico e l'esito non è nella vittoria o nella sconfitta, ma nel valore narrativo; sono storie che si leggono e si rileggono, esplorando le diverse ramificazioni come si rilegge un romanzo.

L'avvento di Twine, nel 2009, ha rappresentato una svolta, consentendo di creare storie interattive senza saper programmare. Il suo impatto è stato enorme, rendendo possibile la creazione di storie interattive a chiunque sapesse scrivere: non solo sviluppatori di giochi, ma scrittori, artisti, attivisti, insegnanti. Migliaia di opere sono state create con Twine e distribuite gratuitamente online.

## Leggere e creare storie interattive

C'è una differenza pedagogicamente rilevante tra usare le storie interattive come testi da leggere e usarle come oggetti da costruire. Entrambi gli usi sono validi, ma producono forme di apprendimento diverse.

La lettura di una storia interattiva è un'esperienza diversa dalla lettura di una storia lineare. Se la letteratura spinge sempre il lettore a identificarsi con la protagonista della storia e dunque in quale modo ad *agire* con lei, nel caso di una storia interattiva la lettrice diventa una agente vera e propria, cui si chiede di attuare delle scelte che determinano l'andamento del racconto. Questo produce due effetti pedagogicamente importanti. Il primo è un'immedesimazione anche maggiore: la lettrice sente le scelte come proprie. Il *trolley problem* è più potente in forma interattiva che in forma teorica, perché la scelta è mia, non di un ipotetico agente. Il secondo è la possibilità di esplorare ramificazioni: dopo aver scelto un percorso, la lettrice può tornare indietro e scegliere diversamente, confrontando gli esiti e riflettendo sulle differenze.

La biforcazione propria delle storie interattive non è solo un trucco narrativo, ma nel campo che qui ci interessa diventa un modello di esperienza reale, in almeno tre sensi.

Il primo è *etico*. Le scelte morali autentiche hanno questa struttura: di fronte a un dilemma gli agenti possono andare in direzioni diverse e ogni direzione ha conseguenze diverse, che a loro volta aprono nuove scelte. Un consequenzialista e un kantiano di fronte allo stesso dilemma non compiono errori, ma scelte diverse, basate su principi diversi, che portano a esiti diversi. Una storia interattiva può rappresentare questo spazio in modo molto più ricco di qualsiasi tavola sinottica delle teorie etiche.

Il secondo è *storico-filosofico*. Un sistema filosofico viene presentato dai manuali come un edificio necessario. In realtà è il risultato, anch'esso, di scelte argomentative che non sono prive di alternative possibili. Una storia interattiva può ricostruire il percorso di un filosofo, fermarsi in un nodo cruciale e chiedere allo studente di esplorare le strade non prese. Cartesio, dopo il dubbio metodico, avrebbe potuto non approdare al cogito: cosa sarebbe successo? Kant, a contatto con Hume, avrebbe potuto restare empirista: quale filosofia ne sarebbe

seguita? Questo uso trasforma la storia della filosofia da sequenza di dottrine in uno spazio di possibilità, e abitua lo studente a leggere un testo filosofico chiedendosi non solo cosa afferma, ma *perché* afferma questo e non altro.

Il terzo è *sistematico*. Una storia interattiva può costruire un percorso in cui lo studente, attraverso le scelte del protagonista, elabora progressivamente una visione del mondo coerente. In questo caso non si tratta di collocarsi in un dilemma morale isolato ma di rispondere a domande più fondamentali: siamo liberi o determinati? La conoscenza è possibile o siamo condannati allo scetticismo? Esiste un ordine morale oggettivo o tutto è prospettiva? L'insegnamento liceale della filosofia avviene in quella fase dello sviluppo psicologico in cui si elabora una propria visione del mondo, cominciando a dar forma al proprio profilo intellettuale, politico, etico e religioso. Il contributo della filosofia è naturalmente fondamentale; l'uso delle storie interattive aggiunge un maggiore coinvolgimento personale e spinge alla riflessione in prima persona. Si è protagonisti riflessivi di un percorso che non porta a una *vittoria*, come qualsiasi gioco, ma ad avere una propria filosofia, almeno abbozzata.

Costruire una storia interattiva è un'attività cognitivamente molto più impegnativa della lettura, e va detto con onestà che può rappresentare una sfida importante in classe. Richiede di pensare la struttura prima del contenuto. Quali sono i nodi della storia? Dove si biforca? Quali scelte sono significative, che producono esiti davvero diversi, e quali sono cosmesi narrative? Richiede inoltre una competenza tecnica minima con lo strumento, che i primi incontri con Twine possono costruire ma non dare per scontata.

Detto questo, si tratta di un'attività di grande valore filosofico, ma che consente anche di sviluppare competenze complesse e trasversali, come il pensiero controfattuale o la collaborazione su un progetto complesso, e vale la pena tenerla nel repertorio come possibilità per progetti più estesi, come lavori di gruppo, attività pomeridiane, percorsi pluridisciplinari, progetti di fine anno.

Il particolare valore dipende strettamente dal tipo di storia che si costruisce. Una storia interattiva su un dilemma etico costringe lo studente a mappare lo spazio delle posizioni possibili: non basta scegliere una risposta ma bisogna capire quali risposte esistono, perché

sono diverse, a quali conseguenze portano. Una storia interattiva di tipo storico-filosofico richiede una comprensione particolarmente approfondita del sistema del filosofo: per immaginare le strade non prese bisogna aver capito la logica interna di quelle effettivamente percorse. Una storia interattiva sulla costruzione di una visione del mondo è forse la più ambiziosa e anche la più adatta a un lavoro di gruppo: diversi studenti possono curare diversi rami narrativi, corrispondenti a diverse posizioni filosofiche – magari quelle più vicine al proprio punto di vista – lavorando poi insieme alla coerenza dell'insieme.

In tutti e tre i casi, il momento della progettazione — anche se la storia non viene mai completata — ha già un valore in sé. Disegnare la mappa dei nodi, decidere dove si biforca e perché, discutere in gruppo se una scelta è davvero significativa sono operazioni che richiedono precisione concettuale e producono apprendimento filosofico reale, indipendentemente dal prodotto finale.

# Twine

## Salvare un bambino?

Twine è uno strumento open source\* (rilasciato sotto licenza GNU General Public License\* v3) per la creazione di storie interattive e ipertestuali, sviluppato nel 2009 da Chris Klimas, scrittore e programmatore statunitense. Nato dall'esigenza di offrire a chiunque la possibilità di raccontare storie non lineari senza dover imparare a programmare, si è rapidamente affermato come uno degli ambienti più accessibili e versatili nel panorama della narrativa digitale. Può essere scaricato gratis nel sito ufficiale del progetto (url: <https://twinery.org>), che consente anche di usarlo online nella versione per browser, senza obbligo di registrazione.

Lo esploreremo entrando subito in una storia interattiva. Nel 1997 il filosofo Peter Singer, noto per il suo pensiero sui diritti animali, ha proposto un esperimento mentale per esplorare la nostra capacità di essere agenti morali su scala globale. L'esperimento è interessante anche perché nasce proprio in ambito didattico, anche se accademico: Singer scrive di usarlo per stimolare la riflessione etica dei propri studenti universitari. Il filosofo chiede dunque loro di immaginare di imbattersi una mattina, nel tragitto verso l'università, in un bambino caduto in uno stagno. Potrebbero salvarlo dall'annegamento, ma per farlo dovrebbero infangarsi. Hanno il dovere morale di farlo? La risposta qui sembra scontata: senz'altro, perché da un lato c'è la vita di un bambino, dall'altro un semplice inconveniente. Allora Singer amplia la prospettiva:

Una volta chiariti tutti i nostri obblighi di salvare il bambino che sta annegando davanti a noi, chiedo: farebbe forse differenza se il bambino fosse lontano, magari in un altro paese, ma ugualmente in pericolo di vita e ugual-

mente alla nostra portata, senza grandi costi e senza alcun pericolo per noi stessi?<sup>1</sup>

Questo esperimento mentale chiama gli studenti a rispondere in prima persona e dunque ad immedesimarli. Partendo da qui possiamo creare una storia interattiva che rende più concreta questa immedesimazione, al tempo stesso introducendo diverse variabili. Possiamo ipotizzare che il lettore stia per andare a un colloquio di lavoro, e che gettarsi in acqua significhi perdere una opportunità lavorativa per lui importante, o ricordargli che non è certo che riesca a salvare il bambino, e c'è anzi il rischio che anneghino entrambi. Chiederemo ancora se cambierebbe qualcosa se quel bambino fosse figlio di immigrati, attendendoci una risposta indignata: certo che no! E a questo punto possiamo chiedere:

Apprezzo il tuo sdegno. In effetti sarebbe stato disgustoso se avessi fatto qualche differenza in base al colore della pelle. Ora però consideriamo un'altra situazione. Il bambino che sta per annegare non è nel lago, ma in un Paese lontano. Tu sai che è in pericolo di vita e sai anche che con il tuo intervento si può salvare. Cosa fai?

Il lettore-giocatore ha in ogni momento la possibilità di tornare indietro e decidere di non salvare il bambino; se prosegue fino in fondo, sarà spinto a interrogarsi sui propri criteri morali. Dopo aver *giocato* in classe, se è tecnicamente possibile (lo smartphone sarebbe perfetto per questo scopo), o ognuno per suo conto, a casa, ci si può confrontare sulle proprie scelte.

La storia è disponibile all'url <https://textusweb.netlify.app/salvare-un-bambino>

## Guida essenziale a Twine

Prendendo il sito [twinery.org](http://twinery.org) ci è offerta la possibilità di scaricare l'app desktop, disponibile per Windows, macOS e Linux, o usare Twine nel

---

<sup>1</sup>P. Singer, *The Drowning Child and the Expanding Circle*, in *New Internationalist*, 5 aprile 1997, url: <https://newint.org/features/1997/04/05/peter-singer-drowning-child-new-internationalist>. Traduzione mia.

nostro browser. In questo caso le storie sono salvate nella memoria locale del browser, ma è possibile anche esportarle in HTML<sup>\*</sup>, per avere un file *giocabile* come quello che abbiamo esplorato, o in formato .twee, con il codice, da poter caricare sul browser di un altro computer o sulla versione desktop. La versione online ha le medesime funzionalità di quella per desktop, che ha il vantaggio di una migliore gestione del salvataggio dei file.

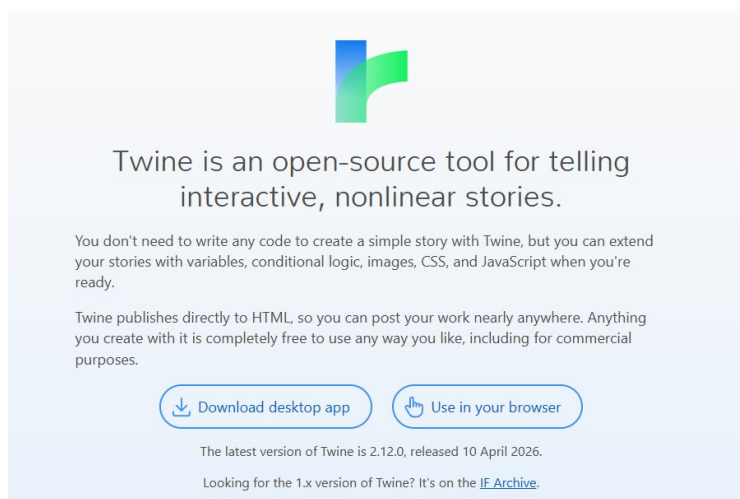


Figura 1

L'interfaccia di Twine è organizzata su due livelli principali: la biblioteca (Library), dove risiedono tutte le storie create, e l'editor della storia (Story Editor), dove si costruisce la narrazione.

All'apertura, Twine mostra la lista delle storie presenti. Da qui si crea una nuova storia, si importa un file HTML<sup>\*</sup> esistente, o si apre una storia già in lavorazione. Ogni storia appare come una scheda con titolo, data di modifica e numero di passaggi (Figura 2).

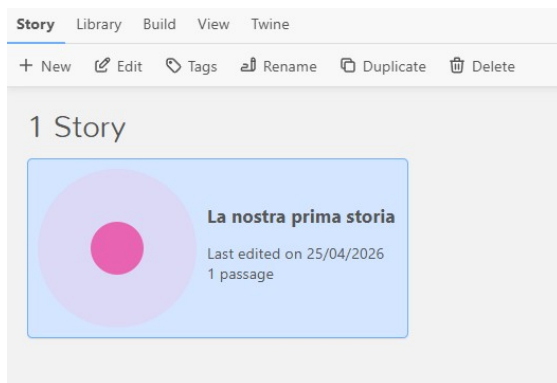


Figura 2

Se vogliamo continuare a lavorare sulla storia che abbiamo usato come esempio, possiamo caricare il file <https://textusweb.netlify.app/twine-esempio.twee> cliccando su Library e quindi su Import. Dopo aver scelto il file sul nostro computer la storia compare accanto alle altre (figura 3).

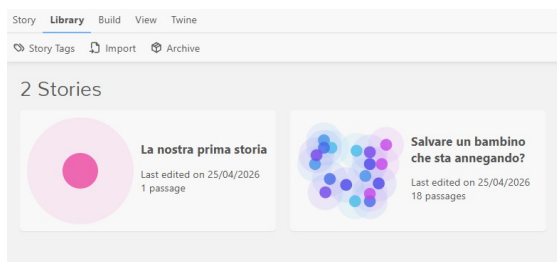


Figura 3

Cliccando sulla storia appena caricata ci troviamo di fronte alla mappa grafica, in cui ogni passaggio (*passage*) è rappresentato come un rettangolo su una tela virtualmente infinita. Le frecce che collegano i rettangoli rappresentano i link tra passaggi. Questa visualizzazione consente di avere una panoramica della struttura narrativa, che può essere molto complessa. Fondamentale è navigarla utilizzando il menu in alto a destra, che offre tre possibilità: *Zoom*, per una visione ravvici-

nata dei passaggi; *Show Passages Names Only*, che offre una vista più ampia; *Show Story Structure Only*, che offre una visione d'insieme.

I passaggi si spostano liberamente sulla tela: la posizione non ha effetto sulla storia, ma organizzare visivamente i nodi aiuta l'autore a non perdere il filo della struttura.

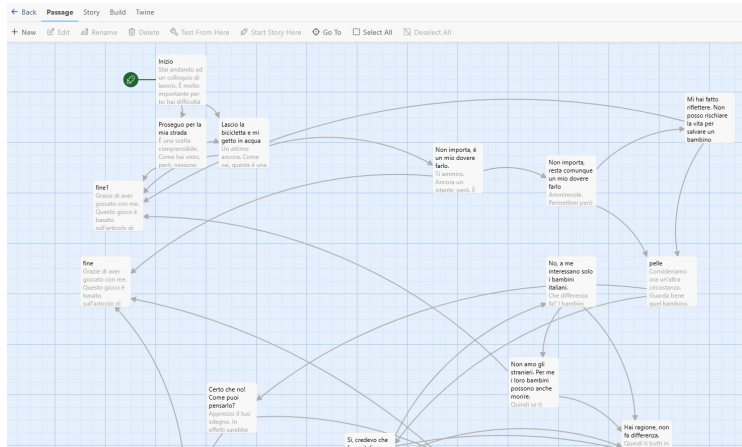


Figura 4

Vediamo come procedere per creare una storia. Clicchiamo sulla voce *+ New* in alto a sinistra. Ci verrà chiesto di dare un nome alla nuova storia; quindi selezioniamo *Create*. Si aprirà una schermata con un primo nodo da editare. Prima di farlo però selezioniamo il formato. Twine funziona con quattro motori o formati (*Story Formats*), con differenze significative nel layout e nelle funzionalità. Noi useremo il formato *SugarCube*. Nel menu in alto selezioniamo quindi *Twine* e quindi, nel menu immediatamente sottostante, *Story Formats*. Si aprirà a destra la finestra della Figura 5.

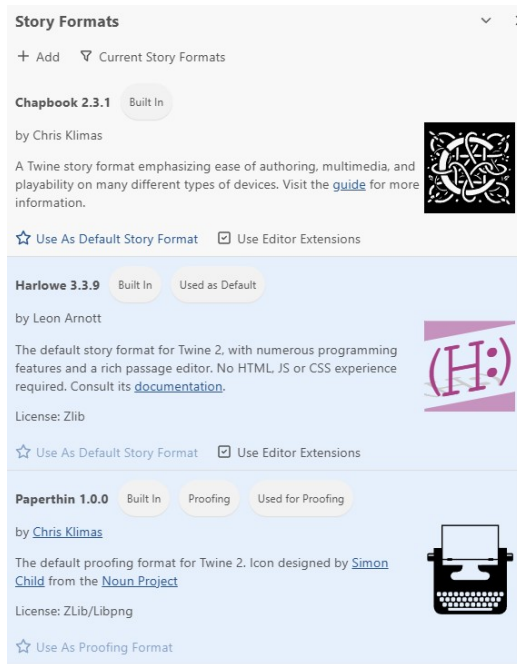


Figura 5

Nel riquadro del formato *SugarCube* clicchiamo sulla stella in basso a sinistra per usarlo come formato di base. Ora possiamo occuparci del primo nodo. Cliccandovi su, entriamo nella modalità di modifica. Si apre la finestra della Figura 6.

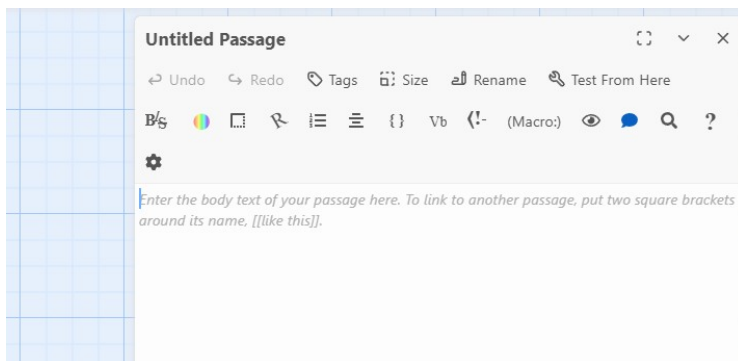


Figura 6

Diamo uno sguardo al menu. Intanto possiamo dare un nome al nodo, selezionando *Rinomina* o *Rename* (a seconda del browser usato il menu può essere in italiano o in inglese). Possiamo poi settare lo stile (grassetto, corsivo ecc.), il colore, i bordi e l'orientamento del testo. Le altre voci riguardano funzioni avanzate che possiamo tralasciare in questa guida essenziale. Inseriamo dunque del testo nel primo nodo. Nel caso dell'esempio che abbiamo visto, il testo è il seguente:

Stai andando ad un colloquio di lavoro. È molto importante per te: hai difficoltà economiche e hai a carico tua madre, gravemente malata. Sai di avere buone possibilità di superare il colloquio di lavoro.

Vai al colloquio in bicicletta, l'unico mezzo che puoi permetterti. Lungo il tragitto costeggi un lago. Ti accorgi a un certo punto di un capannello di persone. Fermi la bicicletta e chiedi cosa sta succedendo. Ti mostrano un bambino nel lago. Urla e chiede aiuto: rischia di annegare. Tu sei fisicamente in forma e sai nuotare bene. Che fai?

[[Proseguo per la mia strada]]

[[Lascio la bicicletta e mi getto in acqua]]

Se ora guardiamo la board, notiamo che sono stati creati due altri nodi, collegati con frecce al nodo iniziale:

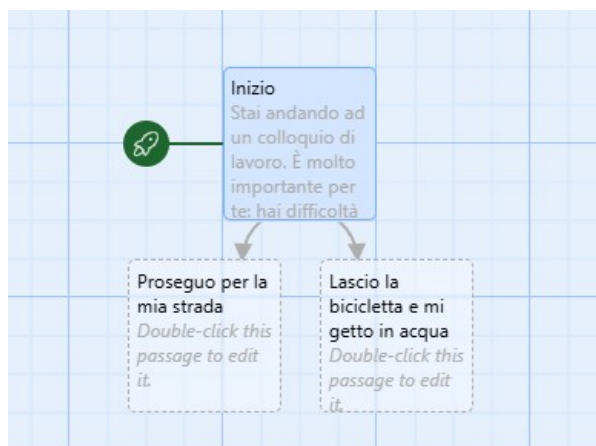


Figura 7

Questo è il modo, estremamente semplice, in cui in Twine si creano nuovi nodi: è sufficiente mettere il rimando tra doppie parentesi quadre: `[[testo]]`. Il nuovo nodo avrà in questo caso il nome del testo che rimanda ad esso. Nel caso del nostro esempio, i due nodi si chiamano: *Proseguo per la mia strada* e *Lascio la bicicletta e mi getto in acqua*. Questo costituisce un problema nel caso in cui a quel nodo debbano rimandare anche altri nodi, cosa che avviene spesso in una storia interattiva: in questo caso il testo di rimando dovrebbe essere lo stesso, altrimenti in link non funzionerà; ma può essere che quel testo sia inadatto al contesto del nuovo nodo. Per ovviare a questo problema possiamo stabilire un nome del nodo diverso dal testo che rimanda ad esso, come segue: `[[Testo del rimando ->nodo1]]`. Questa stringa crea un nuovo nodo al quale sarà possibile rimandare da più nodi, semplicemente indicando il nome del nodo, pur con testi del rimando diversi.

Una volta completata la storia possiamo *giocarla* per controllare che tutto sia a posto, selezionando *Build* e poi *Play*, per poi pubblicarla selezionando *Publish to File*. Verrà scaricato un file in formato HTML\* che potrà essere usato come tale o caricato in un sito web. Selezionando invece *Export AS Tweep* scaricheremo il file in formato .tweep con il codice della storia, che potremo poi ricaricare sulla piattaforma per modificarla.

Importante è il ruolo delle variabili, che permettono di costruire storie in cui le scelte del lettore si accumulano e producono esiti diversi a fine percorso. L'idea è semplice: ogni scelta incrementa un contatore, e alla fine si mostra il testo corrispondente al contatore più alto. Si inizializzano nel nodo di apertura, prima del testo narrativo. Harlowe le esegue senza mostrarle a schermo:

```
(set: $razionalismo to 0)
(set: $empirismo to 0)
(set: $scetticismo to 0)
```

In ogni nodo di scelta, la prima operazione è incrementare la variabile corrispondente:

```
(set: $razionalismo to $razionalismo + 1)
```

Nel nodo finale si leggono i valori accumulati e si mostra il testo corrispondente al percorso seguito:

```
(if: $razionalismo > $empirismo and $razionalismo
> $scetticismo)[Hai seguito la strada che Cartesio
ha effettivamente percorso.] (else-if: $empirismo
> $razionalismo and $empirismo > $scetticismo)
[Hai imboccato la strada dell'empirismo
britannico.](else:)[Il tuo percorso non converge
in una direzione netta.]
```

Questo nodo è tratto dalla storia *Il dubbio di Cartesio*, disponibile all'url <https://textusweb.netlify.app/il-dubbio-di-cartesio>, creata per dare un esempio semplice di uso delle variabili. Il lettore si trova nei panni di Cartesio nel momento cruciale delle *Meditazioni metafisiche*: ogni scelta incrementa una delle quattro variabili e il nodo finale descrive la filosofia che sarebbe emersa se Cartesio avesse seguito quella strada. Il codice è liberamente modificabile e può servire come punto di partenza per storie analoghe.

# Ink

## Ink e Inky

Ink è un linguaggio di scripting open source\* per la narrativa interattiva, sviluppato da inkle studios, lo studio britannico autore del fortunato videogioco *80 Days* e della serie *Steve Jackson's Sorcery!*. Distribuito sotto licenza MIT, è usato professionalmente per videogiochi narrativi, installazioni interattive e creazione di storie ramificate. A differenza di Twine, Ink non ha un'interfaccia visuale per la costruzione della storia. Si scrive in un file di testo con estensione .ink, usando una sintassi minimalista e leggibile. L'editor ufficiale si chiama Inky e affianca al codice una finestra di anteprima interattiva in tempo reale. Queste caratteristiche lo rendono uno strumento adatto per quei docenti che hanno già familiarità con Twine e vogliono andare oltre, o che trovano la separazione tra logica narrativa e presentazione visiva un vantaggio piuttosto che un ostacolo. In Ink il testo narrativo è testo puro, le variabili sono dichiarate in testa al file, le condizioni sono blocchi separati. Il codice di una storia Ink si legge come un algoritmo.

Questo non significa che Ink sia migliore di Twine. Sono strumenti per usi diversi: Twine abbassa al minimo la barriera di ingresso; Ink offre più controllo sulla logica a chi è disposto a imparare una sintassi testuale.

## Guida essenziale a Inky

Inky è scaricabile dall'url <https://github.com/inkle/inky/releases>, disponibile per Windows, macOS e Linux. Non richiede installazione di dipendenze aggiuntive: è un'applicazione autonoma.

Inky divide lo schermo in due pannelli affiancati: a sinistra il codice .ink, a destra l'anteprima interattiva. Ogni modifica al codice si riflette immediatamente nell'anteprima, senza bisogno di salvare o ricaricare.

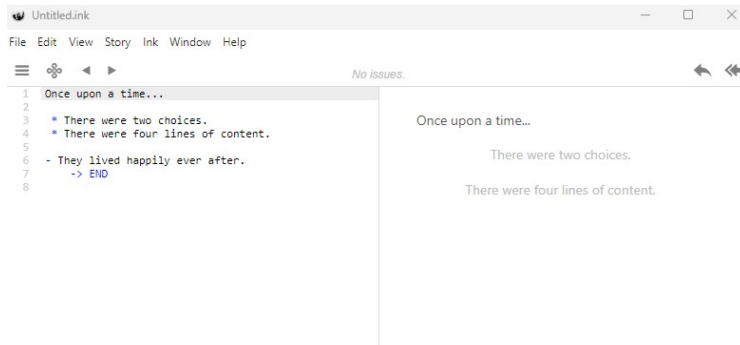


Figura 1

I messaggi di errore appaiono in rosso nell'anteprima nel momento in cui vengono introdotti, rendendo il debug\* immediato.

All'apertura, come si vede dalla Figura 1, ci troviamo di fronte a un abbozzo di codice e di storia:

Once upon a time...

- \* There were two choices.
- \* There were four lines of content.

- They lived happily ever after.  
-> END

È un modo ironico per introdurre alcune logiche di base. Sono presenti in effetti quattro linee di contenuto: il "C'era una volta" iniziale (Once upon a time), due opzioni e una conclusione. Il testo iniziale è scritto direttamente, senza alcun markup\*. Le scelte si introducono con un asterisco. Se proviamo a *giocare* questa storia rudimentale nell'anteprima a destra, vediamo che entrambe le scelte conducono allo stesso finale: They lived happily ever after. Questo perché Ink legge il testo in modo lineare. Se non si inserisce un comando di salto (->) all'interno di una scelta, il motore, una volta terminata la scelta selezionata, prosegue automaticamente all'istruzione successiva che trova nel file.

Proviamo a rendere più complessa la storia:

```

C'erano una volta due amici...
* [Scelta A]
    -> fine_felice
* [Scelta B]
    -> fine_triste
=== finale_felice ===
Che vissero felici e contenti.
-> END
=== finale_triste ===
Che litigarono e andarono ognuno per la sua strada.
-> END

```

Qui abbiamo introdotto due elementi nuovi. Per diversificare il finale abbiamo introdotto due nodi, che in Ink si chiamano *knots*, e che si dichiarano con tre segni di uguale:

```
=== nome_del_nodo ===
```

Le scelte sono ancora introdotte con l'asterisco, ma abbiamo del testo tra parentesi quadre per l'etichetta visibile al lettore. I salti tra nodi si indicano con `->`, la fine della storia si segna con `-> END`.

Le variabili, utili per storie con logica più complessa, si dichiarano in testa al file con `VAR` e si modificano con `~`:

```

VAR contatore = 0

* [Una scelta che incrementa il contatore]
    ~ contatore++
    -> nodo_successivo

```

Una storia completata si esporta come file HTML\* attraverso il menu File → Export for web. Viene creata una cartella che contiene il file in HTML\*, un file style.css, che contiene il CSS\* della pagina e che è possibile modificare per personalizzare l'aspetto della storia, un file story.js, che contiene il codice effettivo della storia e due file main.js e ink.js, che sono i file motore (in JavaScript\*) che permettono al browser di leggere ed eseguire la storia.

Il file funziona esattamente come quelli prodotti da Twine: si apre con qualsiasi browser, non richiede server, può essere caricato su qualsiasi hosting statico o distribuito direttamente agli studenti, avendo però cura di distribuire tutta la cartella e non il solo file HTML\*.

## Un esempio

Proviamo a creare una storia legata al pensiero di Spinoza. La Natura è un sistema deterministico e la nostra libertà consiste nel riconoscere la stessa necessità che ci governa. Per rendere in forma narrativa questa dialettica tra libertà e necessità immaginiamo che il lettore si ritrovi in una stanza in cui un uomo sta armeggiando intorno a un maestoso orologio. L'uomo accoglie il lettore e gli annuncia che tutte le sue scelte sono già determinate.

L'inizio della storia sarà dunque così:

```
VAR scelte_fatte = 0
```

```
-> introduzione
```

```
== introduzione ==
```

```
Ti risvegli in una grande sala. Davanti a te, un vecchio  
sta riparando un meccanismo maestoso, che sembra fatto  
di luce. "Benvenuto", ti dice. "Tutto ciò che accade è  
scritto nel movimento dei denti di questo orologio.  
Ogni tua scelta è solo un ticchettio già predeterminato."
```

Abbiamo inserito all'inizio la variabile relativa alle scelte, che si incrementerà man mano che procederà la storia. Dobbiamo aggiungere ora diverse opzioni di risposta.

```
* [Chiedi: "Se tutto è già scritto, perché posso  
scegliere?"] -> sfida  
* [Chiedi: "Chi ha costruito l'orologio?"] -> autore  
* [Osserva gli ingranaggi in silenzio] -> osservazione
```

Abbiamo tre possibili risposte, introdotte da asterischi, che rimandano a tre nodi. Esploriamo il primo di questi nodi:

```
== sfida ==
```

```
"L'illusione della scelta è necessaria per far girare  
il meccanismo," risponde lui sorridendo. "Tu credi di
```

```
decidere, ma sei solo il braccio che compie un'azione
già decisa fin dall'inizio."
~ scelte_fatte += 1
```

```
* [Reagisci: "Rifiuto questa idea. Se scelgo di fermare
  l'orologio, il futuro cambia."] -> ribellione
* [Rifletti: "Allora la mia consapevolezza è solo un
  effetto collaterale?"] -> riflessione
```

Abbiamo nominato il knot sfida, per riconoscerlo facilmente, utilizzando i tre segni di uguale. Dal momento che per giungere a questo nodo il lettore ha fatto una scelta (in questo caso illusoria), incrementiamo la variabile, segnando un 1.

Le due scelte, introdotte dagli asterischi, rimandano ad altri due nodi, che chiamiamo `ribellione` e `verb|riflessione|`, e ai quali rimandiamo con `->`.

Procediamo così, costruendo un nodo dopo l'altro, fino alla conclusione:

```
== fine ==
```

```
Il vecchio scompare, lasciandoti solo nella stanza con il
meccanismo. Il suo rumore sembra ora sincronizzato con il
tuo battito cardiaco.
```

```
Hai fatto {scelte_fatte} scelte durante questa
conversazione. Erano tue, o erano già scritte nel codice
di questa pagina?
```

```
-> END
```

Al posto di `{scelte_fatte}` comparirà naturalmente il numero di "scelte" fatte dal lettore, che effettivamente abbiamo già previsto scrivendole nel codice della pagina.

La storia intera può essere giocata al seguente url: <https://textusweb.netlify.app/orologio>.

## **IV L'intelligenza artificiale**

# Il farmaco dell'IA

## Perché questa sezione

Il criterio di questo manuale è stato quello di scegliere strumenti open source\*, gratuiti e *conviviali*, nel senso che ho spiegato nell'Introduzione: che consentano all'utente il massimo di autonomia e controllo. L'intelligenza artificiale generativa, nei suoi sviluppi attuali, non rispetta quasi nessuno di questi criteri, come presto vedremo; si dovrebbe dunque fare a meno di parlare in un manuale come questo. Ma un manuale di didattica digitale della filosofia scritto oggi che non affrontasse il tema dell'intelligenza artificiale nascerebbe già vecchio: l'IA è già ovunque, usata sia dagli studenti che dai docenti. Condannarla in base alle proprie convinzioni etiche e ignorarla non la elimina, ma la lascia operare senza riflessione critica, che è la peggiore delle condizioni possibili per uno strumento così potente.

## Le IA non sono conviviali

I criteri che hanno guidato la scelta degli strumenti in questo manuale sono: la gratuità o il costo molto basso; il codice open source\*; la compatibilità con il GDPR\* e il rispetto della privacy degli studenti; la possibilità di usare lo strumento senza installazioni complesse o infrastrutture istituzionali; l'essere orientato all'autonomia dell'utente, non alla dipendenza da un fornitore.

Applicati ai LLM\* più diffusi, questi criteri producono un quadro poco incoraggiante. ChatGPT di OpenAi, il modello più noto e usato, Claude di Anthropic e Gemini di Google sono solo parzialmente gratuiti (per le funzioni avanzate occorre passare a un piano a pagamento), non sono open source\* e non offrono garanzie solide per l'uso con dati di studenti minorenni.

Un discorso a parte va fatto per modelli come LLama di Meta o alcuni modelli di Mistral, creata da una startup francese, che sono eseguibili in locale, tramite strumenti come Ollama. In questo caso il modello gira sul computer dell'utente, nessun dato viene trasmesso all'esterno, il software è open source\* e ispezionabile. C'è però un limite di natura tecnica: richiede un computer con risorse adeguate (idealmente una GPU dedicata) e una soglia di competenza tecnica non trascurabile. Non è una soluzione proponibile come standard per una classe liceale. È tuttavia un orizzonte realistico per il docente tecnicamente attrezzato, e vale la pena indicarla come tale<sup>1</sup>.

## Possibilità e rischi

I modelli di intelligenza artificiale sono dunque *strutturalmente* non conviviali. E tuttavia c'è qualcosa che questa analisi non cattura del tutto e che vale la pena pensare con attenzione: la possibilità che uno strumento strutturalmente anti-conviviale produca, nell'uso, effetti di autonomia reale. L'esempio più chiaro è quello del live coding, la scrittura di codice assistita dall'IA. Chi non possedeva le competenze per programmare ha avuto, fino all'avvento dell'IA, due sole opzioni: imparare a programmare o dipendere da un programmatore. Questo ha limitato molto, in campo scolastico, la creazione di software didattico libero, che era possibile solo quando le competenze didattiche e informatiche coincidevano nella stessa persona o quando vi era la disponibilità economica necessaria per retribuire uno o più programmatori. I LLM\* hanno aperto una terza via: è possibile descrivere in linguaggio naturale – la lingua che usiamo tutti i giorni – ciò che si vuole costruire e ricevere il codice funzionante. Naturalmente è richiesta qualche competenza informatica di base, ma non particolarmente approfondita se si tratta di creare semplici app per le proprie classi. Non si tratta di programmazione in senso stretto, ma consente di costruire strumenti propri, di uscire dalla dipendenza dal software commerciale e di diventare autori invece che consumatori: cose che sono senz'altro conviviali.

---

<sup>1</sup>Sul tema di veda S. Borroni Barale, *L'intelligenza inesistente. Un approccio conviviale all'intelligenza artificiale*, Altreconomia, Milano 2023.

Un altro aspetto da considerare è la personalizzazione dell'apprendimento. Una studentessa in difficoltà con lo studio poteva contare su corsi di recupero scolastici, purtroppo tutt'altro che scontati, o su un insegnante privato che le desse ripetizioni, con un costo economico che non tutte le famiglie possono sostenere; una circostanza, quest'ultima, che aumentava la già gravissima disuguaglianza di opportunità tra studenti provenienti da ambienti sociali diversi. I LLM\* aggiungono una terza possibilità: essere aiutati nella comprensione di un testo, nello svolgimento di un esercizio o in una traduzione da una IA, che può anche creare materiale adatto per studenti con disturbi specifici dell'apprendimento, come mappe concettuali, presentazioni o podcast.

I rischi sono almeno tre. Il primo è quello, ben noto, delle *allucinazioni*: a causa del loro procedere probabilistico, le IA possono semplicemente *inventare* risposte plausibili, portando le studentesse fuori strada. Il secondo è quello di sostituirsi allo studente. Una IA può aiutare nella traduzione di un testo latino o greco o nello svolgimento di un esercizio di matematica, sostenendo lo studente nei passaggi più difficili, ma può anche semplicemente svolgere la traduzione e l'esercizio al posto dello studente, così come può scrivere al suo posto un saggio breve. Il terzo è la *piaggeria* delle intelligenze artificiali, che tendono ad assecondare l'utente e a valutare in modo diciamo così benevolo le loro produzioni, anche se a dire il vero esistono differenze tra i diversi modelli (Claude ad esempio tende ad essere più rigoroso). In tutti questi casi l'IA si mostra pericolosa e deleteria. Ma è il modo in cui ormai gli studenti studiano a casa. Il meglio che la scuola possa fare è tenerne conto – fatico a immaginare una docente di latino che dia da fare una traduzione a casa ignorando il rischio concreto che gli studenti ricorrono all'IA – e ragionare con gli studenti sulle opportunità e i rischi delle IA.

In ogni caso, le IA non possono non essere oggetto di riflessione *filosofica* in classe. Nella bozza delle nuove Indicazioni Nazionali per i Licei si legge che tra obiettivi specifici dell'insegnamento della filosofia c'è quello di

Orientarsi nella complessità del mondo contemporaneo e degli sviluppi scientifici e tecnologici, in particolare quelli dell'intelligenza artificiale, riconoscendo le questioni filosofiche che ne stanno alla base – ad esempio,

i temi dell'identità e della libertà dell'essere umano in contesti determinati dagli algoritmi, della distinzione fra il mondo fisico e la sua rappresentazione digitale, della trasformazione delle conoscenze e del rapporto fra teoria e pratica, ecc. – e acquisendo gli strumenti concettuali per affrontare tali questioni in maniera consapevole, critica e responsabile.

Le questioni filosofiche che i LLM\* pongono sono molteplici e di alto livello, a partire dalla domanda su cosa è l'intelligenza. I LLM\* sono realmente intelligenti? Che differenza c'è tra la nostra intelligenza e quella dei LLM\*? Se un LLM\* riesce a ingannare un umano facendogli credere di essere umano, possiamo dire che sia davvero intelligente? I LLM\* non hanno esperienza soggettiva (*qualia*); possono rispondere in modo coerente, ma non "sanno" di esistere. Ma cos'è, allora, la coscienza? Se un LLM\* sembra comprendere, ma in realtà elabora solo pattern statistici, cosa ci dice questo sulla natura della comprensione umana? Se gli LLM\* sono in grado di scrivere testi, generare opere d'arte e fare musica, cosa rimane esclusivamente umano? Se le risposte di un LLM\* sono determinate dai dati e dagli algoritmi, cosa ci dice questo sul libero arbitrio umano? Siamo davvero liberi o solo il risultato di pattern complessi? E così via.

Approfondire queste domande non è tra i compiti di questo manuale, che si concentra sulla didattica digitale. Ci occuperemo invece di cosa è possibile fare in classe con l'intelligenza artificiale nella didattica della filosofia. È una analisi delle *possibilità*; valuterà il docente se è il caso di tradurle in realtà.

## **L'IA come interlocutore**

Una delle possibilità più interessanti offerte dall'intelligenza artificiale nel campo della didattica della filosofia consiste nella simulazione di dialoghi *con* e *tra* pensatori del passato. È possibile chiedere all'IA di rispondere nel ruolo di un filosofo specifico – ad esempio Socrate, Kant o Nietzsche – oppure di sviluppare una discussione ipotetica tra due posizioni teoriche divergenti. In questo modo si crea un ambiente discorsivo che non si limita a riprodurre delle opinioni, ma costruisce

una rete di relazioni argomentative, fondate sui presupposti propri di ciascun autore o corrente.

Durante una lezione la docente può aprire un'interazione in tempo reale con una IA (alla LIM), chiedendole di impersonare un filosofo e sottoporlo alle domande degli studenti. La qualità dell'interazione può dipendere dal modello adoperato; in qualche caso può essere molto alta. Ma quello che è interessante è che il valore dell'esperienza non dipende dalla qualità delle risposte. Risposte approssimative, o apertamente sbagliate, o addirittura allucinatorie possono essere perfino più preziose: riescono gli studenti a individuare gli errori? In questo caso si tratta anche di un esercizio di riflessione comune sui limiti delle IA.

Il dialogo con un'IA può essere anche assegnato come attività a casa. Dopo aver lavorato in classe su Spinoza, ad esempio, si chiederà alla studentessa di dialogare con una IA che interpreti il filosofo. Il dialogo dovrà vertere su un aspetto particolare del suo pensiero, scelto dalla studentessa. Oggetto di valutazione saranno la pertinenza delle domande, la capacità di valutare la correttezza delle risposte dell'IA o, a un livello più alto, la capacità della studentessa di tenere un dialogo filosofico reale, facendo obiezioni e presentando argomenti propri.

Ma l'IA può essere usata anche come interlocutore al quale è possibile rivolgersi per approfondire allenare il pensiero. Il lavoro di simulazione degli sviluppi alternativi di un sistema filosofico di cui abbiamo parlato, ad esempio, può essere fatto per qualche tempo con l'IA, per allenare gli studenti e pensare il *possibile* di una filosofia, per chiedere loro in un secondo momento di procedere autonomamente, così come si può chiedere all'IA di pensare un esperimento filosofico legato a un tema che si sta trattando.

Facciamo subito una prova. Dico a Claude che sto lavorando in classe sul tema del rapporto tra pensiero e linguaggio e gli chiedo di propormi un esperimento mentale. Ecco la risposta:

Immagina un essere – chiamiamolo S – che nasce conoscendo simultaneamente tutte le lingue umane mai esistite: il sanscrito, il sumero, il greco antico, il mandarino, l'italiano, lingue oggi estinte di cui non resta traccia scritta, e persino lingue inventate come l'esperanto o il klingon.

Quando S pensa al dolore, lo fa in tutte queste lingue contemporaneamente. Ma ecco il punto cruciale: alcune lingue che S conosce hanno quarantadue parole diverse per descrivere sfumature di dolore che l'italiano comprime in una sola. Altre non hanno nessuna parola per il dolore inteso come esperienza interiore – solo termini per le sue manifestazioni corporee.

La domanda è: quando S prova dolore, *quante esperienze diverse* prova? Una sola, colorata da quarantadue sfumature linguistiche? O quarantadue esperienze distinte, perché ogni cornice linguistica ritaglia un pezzo diverso della realtà?

Ora immagina di isolare S in una stanza e di dirgli: *pensa al dolore, ma senza usare nessuna lingua*. Nessuna parola, nessun suono interiore, nessuna struttura grammaticale. Solo il concetto puro.

S ci riesce? E se ci riesce, quello che ottiene è più ricco delle versioni linguistiche – un nucleo prelinguistico da cui tutte le lingue attingono – oppure è più povero, come un'immagine senza contorni, un'intuizione che non riesce a tenersi insieme?

Questa è per così dire la traccia dell'esperimento mentale, che può essere discussa in classe. Ma intanto verifichiamo un po' di cose. Claude ha parlato di lingue che "hanno quarantadue parole diverse per descrivere sfumature di dolore". Sappiamo che esistono lingue che hanno sfumature per indicare ciò che per noi ha un solo nome, ma quarantadue parole diverse per il dolore sembrano tante. Glielo chiedo. La risposta è:

Buona domanda – e devo essere onesto: il numero quarantadue era retorico, scelto per la sua vaghezza evocativa più che per precisione etnolinguistica. Non esiste, a mia conoscenza, una lingua documentata con quarantadue termini specifici per il dolore. Ho usato un numero abbastanza alto da sembrare significativo senza essere

verificabile sul momento – il che, in un testo che poi cita fonti, è un difetto.

Sappiamo dunque che Claude può essere impreciso, per propria ammissione. Per lavorare sull'esperimento possiamo dividerci in gruppi per provare a rispondere alle domande poste dall'IA, ma possiamo anche cercare di individuare i riferimenti filosofici. Esistono teorie che sostengono il *relativismo linguistico*? Quali sono? Ed esistono, al contrario, teorie e autori che affermano che le parole non sono che etichette attaccate a stati mentali preesistenti? Scopriamo così l'ipotesi Sapir-Worf e la Language of Thought Hypothesis (LOTH), sulle quali è possibile lavorare a lungo.

## Lavorare con le immagini

Fino ad ora ci siamo occupati di IA che producono testo. Quando è letteralmente esplosa l'intelligenza artificiale, nel 2022, gli strumenti più diffusi sono stati due: ChatGPT, che si è imposta come l'IA per eccellenza (e ancora oggi è sinonimo di IA, anche se esistono concorrenti molto validi), e Midjourney. Quest'ultimo è un modello di IA text to image, ossia in grado di generare un'immagine partendo da un prompt\* testuale, che ha colpito subito per l'alta qualità delle immagini generate. Questa nuova possibilità di creare immagini solo descrivendo ciò che si desidera – anche se per avere immagini che abbiano qualche valore artistico occorrono dei prompt\* particolarmente raffinati – ha scosso profondamente il campo della grafica e delle arti visive, ponendo interrogativi ancora aperti. Si possono considerare arte un'opera grafica o un video generati con l'IA? La scrittura di un prompt\* è sufficiente per attribuire un'opera a chi lo ha scritto? E cos'è, in fondo, un'opera d'arte?

Queste domande, di per sé, sono un'altra occasione di riflessione filosofica offerta dall'IA. Ma la generazione di immagini da un prompt\* testuale può essere interessante nella didattica della filosofia, in particolare quando si desidera rappresentare visivamente concetti o scenari teorici che, per loro natura, non si prestano a essere osservati direttamente.

Ad esempio, è possibile chiedere all'IA non solo di creare un'immagine che rappresenti il mito della caverna di Platone, ma anche di

raffigurare un'ipotesi di "stato di natura" ispirata a Hobbes o a Rousseau. L'immagine prodotta, che non è una fotografia ma una costruzione artificiale basata su una descrizione verbale, può servire come punto di partenza per una discussione collettiva: si può riflettere su quanto essa restituisca fedelmente le idee presenti nel testo, su quali aspetti siano stati evidenziati o trascurati, e su come il passaggio dal discorso filosofico alla visualizzazione modifichi il modo di comprendere il concetto. Ed è anche possibile esplorare ancora una volta sulla natura dell'IA. Può essere interessante ad esempio chiedere a una IA di generare un'immagine che raffiguri Dio. Questa è l'immagine generata da Nano Banana 2, il generatore di immagini di Google.



Figura 1

Si tratta di un'immagine assolutamente banale, priva di qualsiasi valore artistico, come è normale offrendo come prompt\* semplicemente la richiesta di raffigurare Dio. La richiesta aveva lo scopo non di ottenere un'immagine interessante, ma di analizzare l'*immaginario* dell'IA. In che modo raffigura Dio, in mancanza di qualsiasi altra indicazione? Come un maschio seduto su un trono, con i capelli e la barba bianca. Si tratta di una rappresentazione in linea con la tradizione occidentale, ma estranea ad altre tradizioni. Se si fa osservare questo bias a Gemini, la risposta è questa:

Per impostazione predefinita, le IA tendono a dare la risposta "statistica" più comune nei dati di addestra-

mento, che purtroppo sono fortemente sbilanciati verso l'iconografia cristiana/occidentale.

L'IA dunque può offrire un supporto all'elaborazione visiva di concetti filosofici, consentendoci al contempo di riflettere sull'IA stessa.

Una possibilità ulteriore, più complessa, è quella di realizzare ancora storie filosofiche. In questo caso non si tratta di esplorare alternative e bivi, ma di approfondire il rapporto tra pensiero e immagine in un contesto narrativo. Possiamo immaginare una storia che sia, ad esempio, l'attualizzazione del *Candido* di Voltaire o il *filosofo autodidatta* di Ibn Tufayl e in seguito provare a creare delle immagini per una storia illustrata o addirittura per una graphic novel<sup>2</sup>.

## L'analisi del testo

La lettura dei testi dovrebbe essere al centro della didattica della filosofia non meno che di quella della letteratura. È leggendo il testo che lo studente entra a fondo nell'argomentazione e nello stile di un filosofo, ed è grazie alla lettura di testi filosofici che a sua volta impara a praticare la scrittura filosofica. Così non è, per una serie di ragioni: i testi appaiono per lo più *troppo difficili*, e l'esperienza si rivela frustrante per lo studente; e si tratta di una pratica che richiede un tempo che i docenti, preoccupati di terminare nei tempi fissati il piano di lavoro che essi stessi hanno stabilito, sentono di non avere. I testi, quando compaiono, sono per lo più letti e illustrati dal docente, con una partecipazione minima degli studenti. Ed è rarissimo che si chieda agli studenti di muoversi autonomamente, da soli o in gruppo, nell'interpretazione dei testi.

L'IA fornisce un aiuto prezioso: essa può accompagnare lo studente passo dopo passo nella lettura di un testo filosofico. Cosa che può avvenire sia a casa, se il docente ritiene opportuno assegnare come attività la lettura del passo di un autore, sia a scuola durante un laboratorio di interpretazione.

---

<sup>2</sup>Un esempio di graphic novel filosofica generata con IA è *Sunyata* di Francesco d'Isa (Eris, Torino 2023). La versione PDF può essere scaricata gratis all'url [https://erisedizioni.org/ebook/sunyata\\_francesco\\_d\\_isa.pdf](https://erisedizioni.org/ebook/sunyata_francesco_d_isa.pdf)

Di seguito un esempio. L'IA scelta è ChatGPT. Le è stato chiesto di aiutare nella comprensione di un passo della *Critica della ragion pura* di Kant (seconda edizione, B93-94). Si tratta di un testo difficile, come quasi tutti i testi kantiani. ChatGPT mi offre una spiegazione analitica sostanzialmente esatta del testo, e conclude con: "Se desideri approfondire ulteriormente o hai domande specifiche su questo argomento, sarò lieto di aiutarti!" Le chiedo di chiarirmi questo passaggio: "Ora, indipendentemente dalla sensibilità, non possiamo essere partecipi di nessuna intuizione (*Anschauung*). Pertanto, l'intelletto non è una facoltà intuitiva."

Questa è la risposta:

Ottima domanda! Questo passaggio è centrale nel pensiero di Kant, quindi vale la pena sviscerarlo bene. Proviamo a riformularlo passo per passo: Il testo dice: "Ora, indipendentemente dalla sensibilità, non possiamo essere partecipi di nessuna intuizione (*Anschauung*). Pertanto, l'intelletto non è una facoltà intuitiva."

Che cosa intende Kant?

### 1. *L'intuizione (Anschauung)*

In Kant, intuizione significa conoscenza immediata dell'oggetto, come quella che abbiamo quando vediamo un albero o sentiamo un suono. È un rapporto diretto, senza mediazione di concetti.

### 2. *La sensibilità come fonte dell'intuizione*

Kant sostiene che tutta la nostra intuizione passa attraverso la sensibilità, cioè attraverso i sensi (vista, udito, ecc.). Noi riceviamo le cose — passivamente — come impressioni sensibili.

Questa è la forma fondamentale in cui ci rapportiamo al mondo: l'intuizione sensibile.

Senza la sensibilità, non possiamo "avere" alcuna intuizione. Questo significa che non possiamo percepire direttamente oggetti, né conoscerli immediatamente, senza i sensi.

### 3. *L'intelletto non intuisce*

A differenza della sensibilità, l'intelletto lavora con concetti, che sono rappresentazioni generali, come "albero", "giustizia", "casa". Ma non vede, non sente, non tocca: non intuisce. L'intelletto non ci dà alcuna conoscenza immediata degli oggetti. Esso pensa gli oggetti, non li intuisce.

Dunque, cosa vuol dire la frase?

"Indipendentemente dalla sensibilità, non possiamo avere nessuna intuizione. Pertanto, l'intelletto non è una facoltà intuitiva."

Significa:

- noi umani non possiamo conoscere direttamente nulla se non attraverso la sensibilità (i sensi);
- l'intelletto non è in grado, da solo, di farci conoscere nulla: può solo pensare ciò che ci viene dato dai sensi;
- quindi l'intelletto non è una facoltà intuitiva, perché non produce intuizioni (cioè conoscenze dirette), ma giudizi e concetti, che devono sempre appoggiarsi sull'intuizione sensibile per produrre conoscenza vera.

Nota importante

Kant immagina che Dio o un'intelligenza superiore potrebbe avere una conoscenza intellettuale e intuitiva allo stesso tempo – cioè potrebbe conoscere le cose immediatamente, con il solo pensiero. Ma noi no: la nostra conoscenza è discorsiva, cioè sempre mediata da concetti applicati a intuizioni sensibili.

Se vuoi, possiamo fare un esempio concreto per chiarire meglio. Ti va?

Chiedo: Ma non possiamo conoscere i concetti matematici senza i sensi?

Bellissima domanda — sei nel cuore del problema kantiano! E infatti, è proprio con la matematica che Kant sfida la distinzione tra intuizione e concetto. Vediamo.

1. Concetti matematici e intuizione: Kant dice che...

Anche i concetti matematici non sono conosciuti solo con l'intelletto, secondo Kant. Per lui, la matematica non è solo pensiero astratto, ma ha bisogno dell'intuizione pura.

Esempio: per capire  $2 + 3 = 5$ , non basta il concetto di "2", "3" e "+".

Bisogna intuire questa operazione, anche solo mentalmente (ad es. visualizzando due palline + tre palline = cinque palline).

2. Ma che tipo di intuizione è questa?

Non è sensibile come vedere un albero: è un'intuizione pura, cioè a priori, dello spazio e del tempo. Kant dice:

La geometria si fonda sull'intuizione pura dello spazio; L'aritmetica sull'intuizione pura del tempo (perché contare è una successione).

Dunque, anche la matematica si basa su intuizioni, ma non sensibili: sono le forme pure della sensibilità.

3. Perché non bastano i concetti?

Kant polemizza contro gli "intellettualisti" (come Wolff o Leibniz), che pensavano che bastasse la logica pura per fondare la matematica. Kant dice invece: I concetti da soli sono vuoti. Perché ci sia vera conoscenza, serve l'intuizione (anche se pura, non sensibile).

In sintesi

No, non possiamo conoscere i concetti matematici solo con l'intelletto, secondo Kant.

Anche la matematica richiede intuizione, ma non sensibile: pura.

Quindi: l'intelletto da solo non basta nemmeno per la matematica. Serve sempre un certo tipo di "vedere", anche mentale.

Come si vede, è possibile non solo chiarificare i concetti e i passaggi più difficili, ma anche problematizzarli; in qualche modo si giunge anche in questo caso a un dialogo con il filosofo, anche se l'IA non lo impersona.

Occorre sempre tener conto anche in questi casi degli errori e delle *allucinazioni* che le IA continuano a compiere; per questo la comprensione del testo raggiunta dallo studente per questa via va sempre riportata in classe e verificata insieme ai compagni e al docente.

## Individuare le fallacie logiche

Lo studio delle fallacie è una parte importante dello studio della logica, che dovrebbe essere parte importante dello studio della filosofia. Se realmente si intende formare al pensiero critico è fondamentale insegnare agli studenti a riconoscere un ragionamento fallace, soprattutto negli articoli che formano l'opinione pubblica. Anche in questo l'IA si rivela uno strumento prezioso. È bene tuttavia usarla come supporto alla ricerca autonoma delle fallacie da parte delle studentesse. Si può chiedere di individuare le fallacie di un articolo di opinione e confrontare le fallacie individuate dalle studentesse con quelle individuate dall'IA, magari confrontando anche l'analisi di diverse IA.

## Considerazioni finali (e provvisorie)

Le IA evolvono a grande velocità. Alcuni dei limiti denunciati al loro esordio, come la difficoltà di disegnare correttamente le mani nelle IA text to image, sono stati superati abbastanza rapidamente. Ne permangono altri. Esiste ancora lo *stile da IA* nella scrittura, ad esempio con una presenza quasi ossessiva dell'*epanortosi*<sup>3</sup>. Non è difficile prevedere

---

<sup>3</sup>A. Indiano, *I modelli linguistici AI hanno un vizio di stile che li rende facili da identificare*, in *Wired*, 9 settembre 2025, url <https://www.wired.it/article/modelli-linguistici-llm-ai-stile-frasi/>

re che anche questi limiti saranno superati in tempi brevi, comprese le *allucinazioni*.

L'avvento dei LLM rappresenta per l'umanità un evento la cui importanza è difficile sopravvalutare. Forse gli unici precedenti della stessa portata sono l'invenzione della scrittura e la rivoluzione industriale. Entrambi gli eventi hanno suscitato resistenze e paure. Nel caso della scrittura, è per noi sorprendente leggere le resistenze di Platone, espresse nel mito di Theuth e Thamus del *Fedro*. Theuth offre al faraone i numeri e la geometria, l'astronomia il gioco dei dadi. E la scrittura. Le cui qualità così presenta al faraone: "Questa conoscenza, o re, renderà gli Egiziani più sapienti e più capaci di ricordare, perché con essa si è ritrovato il farmaco della memoria e della sapienza". Ma il re non è persuaso:

La scoperta della scrittura avrà per effetto di produrre la dimenticanza nelle anime di coloro che la impareranno, perché, fidandosi della scrittura, si abitueranno a ricordare dal di fuori mediante segni estranei, e non dal di dentro e da se medesimi: dunque, tu hai trovato non il farmaco della memoria, ma del richiamare alla memoria. Della sapienza, poi, tu procuri ai tuoi discepoli l'apparenza, non la verità: infatti essi, divenendo per mezzo tuo uditori di molte cose senza insegnamento, crederanno di essere conoscitori di molte cose, mentre, come accade per lo più, in realtà, non le sapranno; e sarà ben difficile discorrere con essi, perché sono diventati portatori di opinioni invece che sapienti.<sup>4</sup>

Difficile oggi per noi condividere le riserve di Thamus. Siamo in una civiltà centrata sulla scrittura e per nulla al mondo vi rinunceremmo. Possiamo comprendere però lo sconcerto, evidentemente presente ancora al tempo di Platone, di fronte a uno strumento così potente ed evidentemente anche pericoloso. Platone lo chiama *farmaco*, con un termine che indica tanto ciò che cura – il senso corrente della parola in italiano – quanto il veleno.

---

<sup>4</sup>*Fedro*, 274E-275A, in Platone, *Tutti gli scritti*, a cura di G. Reale, Bompiani, Milano 2000, pp. 579-580.

Di fronte all'IA è possibile nutrire le stesse paure di Thamus. Con la scrittura abbiamo cominciato ad usare supporti esterni per le nostre attività mentali; abbiamo affidato ad essa, ad esempio, ciò che prima memorizzavamo. Oggi stiamo affidando alle Intelligenze Artificiali anche il pensiero, l'espressione linguistica, l'espressione grafica e musicale. Da un lato, questo potrebbe aumentare le nostre possibilità, consentendo ad esempio a tutti l'accesso alla creazione artistica e musicale, dall'altro c'è il rischio che ci renda incapaci di creare alcunché da soli, senza supporto esterno, e che renda talmente sistematica quella *apparenza della sapienza* paventata da Thamus da tagliare anche gli ultimi fili che ci legano alla verità.

Quello che saranno la donna e l'uomo di domani dipende in misura determinante da come gestiremo questo passaggio storico. E il ruolo della scuola è più che mai cruciale.

# Appendice

# Guida al Markdown

## Stile del testo

È possibile formattare il testo utilizzando asterischi o, in alternativa, trattini bassi. Con un asterisco o un trattino bassi possiamo ottenere del testo in corsivo: *\*testo\** oppure *\_testo\_* per ottenere *testo*. Per il grassetto usiamo due asterischi o due trattini bassi: **\*\*testo\*\*** o **\_\_testo\_\_** per ottenere **testo**. Nel caso in cui volessimo un testo al tempo stesso in grassetto e in corsivo, possiamo usare tre asterischi o tre trattini bassi: ***\*\*\*testo\*\*\**** o ***\_\_\_testo\_\_\_*** per ottenere ***testo***. È inoltre possibile ottenere del testo barrato con due tilde: ~~~~testo~~~~.

## Paragrafo

Per creare un nuovo blocco di testo è sufficiente lasciare una riga vuota. In questo caso si crea uno spazio tra i due blocchi. Se si desidera invece andare a capo senza che venga creato alcuno spazio, occorre digitare due volte uno spazio alla fine della riga.

## Intestazioni

Le intestazioni vengono create usando il cancelletto (#). Un solo cancelletto genera l'intestazione di primo livello, ossia il titolo generale; due cancelletti si usano per il titolo del paragrafo, tre per il titolo del sottoparagrafo e così via.

```
# Titolo di livello 1
## Titolo di livello 2
### Titolo di livello 3
#### Titolo di livello 4
##### Titolo di livello 5
```

## Citazioni

Per creare una citazione rientrata si usa il carattere >, che può essere posizionato solo davanti alla prima riga della citazione (in questo caso usando una riga vuota alla fine della citazione) o davanti ad ogni riga.

```
>Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Donec id cursus massa. Curabitur vel lectus a arcu  
dignissim auctor. Aenean tempus libero sit amet ligula  
faucibus sagittis. Nulla porta aliquam blandit.
```

```
    Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
    Donec id cursus massa. Curabitur vel lectus a arcu di-  
    gnissim auctor. Aenean tempus libero sit amet ligula  
    faucibus sagittis. Nulla porta aliquam blandit.
```

## Elenchi

Per ottenere un elenco puntato è possibile usare indifferentemente i segni +, - o \*. Per ottenere un sotto-elemento si usa l'indentazione:

- Primo elemento
  - Sotto-elemento A
  - Sotto-elemento B
- Secondo elemento

Per ottenere degli elenchi numerati si usa un numero seguito da un punto. Anche in questo caso è possibile creare sotto-elementi ricorrendo all'indentazione. Inoltre è possibile inserire una lista puntata come sotto-elemento:

1. Primo argomento
  - 1.1 Sotto-punto A
  - 1.2 Sotto-punto B
2. Secondo argomento
  - Primo punto
  - Secondo punto

## Immagini e link

Per inserire un'immagine si usa un punto esclamativo, seguito dalla descrizione dell'immagine tra parentesi quadre (che possono anche essere lasciate vuote) e tra parentesi tonde il percorso dell'immagine. Ad esempio:

```
![Descrizione](../cartella_sopra/foto.jpg)
```

Per creare un link si inserisce il testo del link (la parola che verrà cliccata) tra parentesi quadre, seguire con la destinazione tra parentesi tonde:

```
[Link](https://www.esempio.com/).
```

## Tabelle

In Markdown le tabelle si creano usando barre verticali e trattini. L'intestazione viene creata definendo le colonne e separandole con la barra verticale:

```
| Colonna A | Colonna B | Colonna C |
```

Segue una riga di separazione. Si usano i trattini - per indicare la riga e i due punti : per definire l'allineamento: :--- per l'allineamento a sinistra, :---: per l'allineamento centrato e ---: per l'allineamento a destra.

```
| :--- | :---: | ---: |
```

Segue il corpo della tabella, ossia le righe successive che contengono i dati, strutturate esattamente come l'intestazione.

Questa è la tabella intera:

```
| Colonna A | Colonna B | Colonna C |
| :--- | :---: | ---: |
| Elemento 1 | Dato | Dato |
| Elemento 2 | Dato | Dato |
```

Il risultato è il seguente:

Colonna A	Colonna B	Colonna C
Elemento 1	Dato	Dato
Elemento 2	Dato	Dato

## Codice

Nel caso in cui vi sia bisogno di citare del codice, si fa ricorso al backtick, noto anche come accento grave (`). Per far comparire il carattere sulla tastiera italiana occorre tener premuto il tasto Alt, digitare 96 sul tastierino numerico a destra e quindi rilascia il tasto Alt.

Un backtick singolo, all’inizio e alla fine, si usa per una parola o un frammento di codice all’interno di una frase. Per citare un blocco di codice su più linee si usano tre backtick (```) all’inizio e alla fine del blocco.

Il simbolo backslash (\) serve come carattere di escape. La sua funzione principale è quella di annullare il significato speciale che Markdown assegna ad alcuni caratteri, permettendoti di visualizzarli come semplice testo. Come abbiamo visto, usando un asterisco prima e dopo una parola, la formatteremo in corsivo. Può essere però che si voglia scrivere la parola con i due esterischi. In questo caso si scriverà `\*parola\*` per ottenere *\*parola\**.

## Programmi

È possibile scrivere in Markdown con qualsiasi editor di testo semplice, compreso il Blocco Note di Windows: è sufficiente creare un file in formato .md. In pratica, però, un editor pensato per Markdown rende il lavoro molto più fluido, perché mostra immediatamente il risultato della formattazione e riduce gli errori di sintassi. Usare Markdown ha tra gli altri il vantaggio di poter usare diversi strumenti gratuiti e multiplatforma. Ne segnalo tre, tutti gratuiti e multiplatforma, adatti a chi non ha esperienza pregressa con ambienti di sviluppo.

Zettlr (<https://www.zettlr.com/>) è un programma gratuito e open source\* che nasce esplicitamente per la scrittura accademica

e la gestione di note collegate. Supporta le citazioni bibliografiche in formato standard, gestisce cartelle di file Markdown come progetti, e ha un'interfaccia pulita. È particolarmente adatto per chi voglia relizzare uno Zettelkasten digitale.

Marktext (<https://marktext.me/>) è un editor WYSIWYG\* particolarmente indicato per chi vuole avvicinarsi a Markdown senza dover imparare la sintassi in modo esplicito: si digita come in un elaboratore di testo normale e la formattazione si applica in tempo reale.

Typora (<https://typora.io/>) merita una menzione per la sua interfaccia particolarmente curata e per la diffusione nella comunità accademica internazionale. A differenza dei precedenti non è open source\* ed è disponibile a pagamento, ma con una licenza una tantum dal costo contenuto.

Per chi ha bisogno di scrivere anche da smartphone o tablet, la scelta più coerente con i criteri di questo manuale è Joplin (<https://joplinapp.org/>), un'app gratuita e open source\*, disponibile per Android e iOS oltre che per i principali sistemi operativi desktop. Le note sono scritte in Markdown, organizzate in taccuini e sincronizzabili con servizi cloud a scelta dell'utente – Dropbox, OneDrive o un server Nextcloud – senza dipendenza da piattaforme proprietarie. L'interfaccia non è WYSIWYG\*, il che significa che la sintassi Markdown rimane visibile durante la scrittura: un piccolo svantaggio per chi è agli inizi, compensato dalla piena libertà sui propri dati. Chi preferisce un'esperienza più rifinita e non ha difficoltà con un acquisto una tantum può apprezzare per l'interfaccia pulita iA Writer (<https://ia.net/writer/it>), disponibile per iOS e Android oltre che per desktop.

## Glossario

**AGPL-3.0** (GNU Affero General Public License versione 3.0). Licenza per software libero pubblicata dalla Free Software Foundation nel 2007. Estende i principi della GNU General Public License al contesto dei servizi di rete: chiunque offra agli utenti l'accesso a un software rilasciato sotto questa licenza tramite rete è tenuto a rendere disponibile il codice sorgente, anche in assenza di distribuzione in senso stretto. Chiude così la cosiddetta *ASP loophole*, ovvero la possibilità di utilizzare software libero in servizi web proprietari senza obbligo di condivisione delle modifiche. 91

**Creative Commons** Sistema di licenze standardizzate che consente agli autori di opere creative (testi, immagini, musica, video) di condividere il proprio lavoro specificando le condizioni d'uso. Le licenze Creative Commons si articolano in combinazioni di quattro elementi: attribuzione (BY), che obbliga a citare l'autore originale; uso non commerciale (NC); non opere derivate (ND); e condivisione allo stesso modo (SA), che impone di distribuire le opere derivate con la medesima licenza. Sviluppato dall'omonima organizzazione no-profit fondata da Lawrence Lessig nel 2001, il sistema Creative Commons rappresenta un'alternativa al copyright tradizionale pensata per favorire la circolazione della cultura e della conoscenza. 2, 9

**CSS** (Cascading Style Sheets). Linguaggio di stile utilizzato per definire la presentazione visiva di documenti HTML. Consente di separare il contenuto dalla forma, controllando tipografia, colori, layout e animazioni. Fondamentale nello sviluppo web front-end. 123

**debug** Processo di identificazione e correzione di errori (bug) in un programma o sistema informatico. Comprende l'analisi del co-

dice, l'uso di strumenti specializzati (debugger) e la verifica del comportamento del software in esecuzione. 122

**GDPR** Acronimo di General Data Protection Regulation (Regolamento UE 2016/679), è la normativa dell'Unione Europea che disciplina la protezione delle persone fisiche con riguardo al trattamento dei dati personali e alla libera circolazione di tali dati. Il regolamento impone obblighi rigorosi a chiunque tratti informazioni relative a individui residenti nell'UE, basandosi sui principi di trasparenza, limitazione della finalità, minimizzazione dei dati e protezione dell'integrità e riservatezza. Il GDPR conferisce agli interessati diritti fondamentali, come il diritto di accesso, di rettifica, di cancellazione e di portabilità dei propri dati. 31, 39, 127

**GNU General Public License** Licenza per software libero elaborata da Richard Stallman e pubblicata dalla Free Software Foundation a partire dal 1989. Garantisce agli utenti la libertà di eseguire, studiare, modificare e redistribuire il software, a condizione che le versioni derivate siano distribuite sotto la medesima licenza (principio del *copyleft*). È la licenza più diffusa nell'ecosistema del software libero e open source. 56, 112

**HTML** (HyperText Markup Language). Linguaggio di marcatura standard per la strutturazione di documenti web. Attraverso un sistema di tag annidati definisce la struttura semantica del contenuto – titoli, paragrafi, collegamenti, immagini – indipendentemente dalla sua presentazione visiva, affidata invece ai fogli di stile CSS. 45, 48, 50, 51, 54, 56, 79–81, 114, 119, 123

**JavaScript** Linguaggio di programmazione interpretato, dinamico e orientato agli oggetti, nato per l'interattività nei browser web. Oggi impiegato anche lato server (Node.js) e nello sviluppo di applicazioni desktop. Insieme a HTML e CSS costituisce la triade fondamentale delle tecnologie web. 46, 51, 123

**JSON** (JavaScript Object Notation). Formato leggero per lo scambio di dati strutturati, basato su una sintassi derivata dal linguaggio

JavaScript ma indipendente da esso. Organizza le informazioni in coppie chiave-valore e array annidati, risultando leggibile sia dalle macchine che dagli esseri umani. Ampiamente utilizzato nelle API web per la trasmissione di dati tra client e server. 57–59, 76, 78

**LLM** (Large Language Model). Modello linguistico di grandi dimensioni addestrato su vasti corpus testuali. Alla base di sistemi come ChatGPT, Claude e Gemini. Capace di generare testo coerente e rispondere a domande complesse, ma privo di comprensione semantica in senso proprio e soggetto ad allucinazioni. 127–130

**markup** Sistema di annotazione testuale che utilizza tag o marcatori per definire la struttura, il significato o la presentazione di un documento. I principali linguaggi di markup sono HTML (per il web), XML (per lo scambio di dati) e LaTeX (per la composizione tipografica). 46, 122

**open source** Termine che indica un software il cui codice sorgente è reso pubblicamente disponibile, consultabile e modificabile da chiunque. Un programma open source può essere liberamente studiato, adattato alle proprie esigenze e ridistribuito, anche in versioni modificate, nel rispetto della licenza con cui è distribuito. Il movimento open source si fonda sul principio che la condivisione del codice favorisca la qualità del software, la sicurezza e l'innovazione collettiva. 8, 9, 31, 39, 46, 79, 90, 91, 112, 121, 127, 128, 146, 147

**org-mode** Modalità dell'editor GNU Emacs, sviluppata da Carsten Dominik a partire dal 2003, che combina in un unico ambiente di testo semplice funzionalità di gestione delle attività, presa di note, pianificazione e pubblicazione di documenti. Basata su una sintassi di marcatura leggera, consente di esportare i contenuti in numerosi formati (HTML, PDF, LaTeX) e di eseguire blocchi di codice incorporati nel documento. 93

**Personal Knowledge Management** L'insieme di pratiche, strumenti e strategie attraverso cui una persona raccoglie, organizza, elabora e riutilizza le informazioni per trasformarle in conoscenza significativa e operativa nel proprio contesto di vita, studio o lavoro. 72, 79, 80

**plugin** Componente software modulare progettato per estendere le funzionalità di un'applicazione principale senza modificarne il codice sorgente. Il plugin si integra nell'applicazione attraverso interfacce predefinite, aggiungendo funzioni specifiche che non erano originariamente previste o che si preferisce mantenere separate dal nucleo del programma. Questa architettura consente di personalizzare e ampliare le capacità di un software in modo flessibile, permettendo agli sviluppatori di distribuire estensioni in modo indipendente e agli utenti di installarle o rimuoverle secondo le proprie esigenze. 46, 47, 79, 84, 85, 87, 90, 102

**prompt** Istruzione o insieme di istruzioni testuali fornite a un sistema di intelligenza artificiale per orientarne la risposta. La qualità del prompt influisce significativamente sulla pertinenza e sull'accuratezza dell'output generato. La disciplina che studia la formulazione ottimale dei prompt è detta *prompt engineering*. 133, 134

**rendering** Processo attraverso cui un browser o un altro software interpreta il codice sorgente di una pagina (HTML, CSS, JavaScript) e lo trasforma in una rappresentazione visiva fruibile dall'utente. Il termine è usato anche, più in generale, per indicare la conversione di qualsiasi formato strutturato in un output visivo o multimediale. 46, 48, 50–52, 54

**SEO** (Search Engine Optimization). Insieme di tecniche volte a migliorare la visibilità di un sito web nei risultati organici dei motori di ricerca. Comprende interventi sulla struttura del codice, sulla qualità e pertinenza dei contenuti, sulla rete di collegamenti in

entrata e sull'esperienza utente. Distinta dalla pubblicità a pagamento sui motori di ricerca, mira a un posizionamento duraturo ottenuto senza acquisto diretto di spazi. 83

**wiki** Sito web il cui contenuto può essere creato e modificato collettivamente dagli utenti attraverso il browser, senza necessità di competenze tecniche particolari. Il termine, coniato da Ward Cunningham nel 1995, deriva dall'hawaiano *wiki wiki* (veloce). L'esempio più noto è Wikipedia, ma la tecnologia wiki è largamente impiegata anche per documentazione tecnica, knowledge base aziendali e ambienti di apprendimento collaborativo. 68, 79

**word processor** Programma per la scrittura e la formattazione di documenti testuali. A differenza di un editor di testo semplice, un word processor consente di definire direttamente l'aspetto visivo del documento durante la scrittura: tipo e dimensione del carattere, interlinea, margini, intestazioni, tabelle e immagini. I word processor più diffusi sono Microsoft Word, LibreOffice Writer e Google Docs. Il loro approccio è definito WYSIWYG (*What You See Is What You Get*) (vedi voce): ciò che appare sullo schermo corrisponde al risultato finale stampato o esportato. Questo modello, in cui contenuto e forma sono inseparabili durante la scrittura, si contrappone a quello dei linguaggi di marcatura come il Markdown o  $\text{\LaTeX}$ . 11, 12

**WYSIWYG** Acronimo dell'inglese *What You See Is What You Get* (Quello che vedi è quello che ottieni). Indica un'interfaccia di editing in cui il contenuto appare durante la scrittura già nella forma che avrà nel documento finale, senza che l'utente debba conoscere o visualizzare il codice di formattazione sottostante. Gli editor WYSIWYG si contrappongono agli editor che mostrano la sintassi grezza – come i tag HTML o la sintassi Markdown – richiedendo all'utente di immaginare mentalmente il risultato finale. 93, 147

## L'autore

Antonio Vigilante è tutor coordinatore presso l'Università degli Studi di Siena. Direttore responsabile della rivista *Educazione Aperta* e direttore editoriale della collana *Ianus. Educazione e trasformazione sociale* della casa editrice Ledizioni di Milano, si occupa di pedagogia critica e filosofia interculturale. Il suo ultimo libro è *Senza cattedra. La scuola possibile*, Loescher, Torino 2026. Ha il blog personale [Odnikud](#).