



**Politecnico
di Torino**

Politecnico di Torino

Magistrale in Ingegneria Informatica

A.a. 2024/2025

Sessione di laurea Ottobre 2025

CAT-AI: un sistema basato su LLM per creare esercizi didattici personalizzati

Relatori:

Luigi De Russis

Tommaso Calò

Candidato:

Lorenzo Cuccu

Ringraziamenti



Indice

Elenco delle tabelle	VII
Elenco delle figure	VIII
1 Introduzione	1
1.1 Motivazioni	2
1.2 Obiettivi	3
1.3 Struttura della Tesi	3
2 Analisi e Definizione dei Requisiti	5
2.1 Metodi	5
2.1.1 Selezione dei Partecipanti	6
2.1.2 Contenuti e Struttura dell'Intervista	7
2.2 Bisogni Ricorrenti	8
2.2.1 Obiettivi di Apprendimento, Chiarezza e Coerenza	8
2.2.2 Livello di Difficoltà	8
2.2.3 Varietà degli Esercizi	9
2.2.4 Coinvolgimento degli Studenti	10
2.3 Raggruppamento dei Requisiti	11
2.3.1 User Personas: Cosa e Perché	11
2.3.2 Definizione delle Personas	12
2.3.3 Considerazioni su Personas e Requisiti	13
2.4 Definizione dei Requisiti	15
2.4.1 Requisiti Funzionali	15
2.4.2 Requisiti Non-Funzionali	16
3 Prototipo a Media Fedeltà	17
3.1 Scelte di Progettazione: Esperienza Utente	17
3.1.1 Principi di Esperienza Utente	18
3.1.2 Processo a 3 Fasi	18
3.1.3 Editor User-Friendly	20

3.2	Scelte di Progettazione: Interfaccia Utente	21
3.2.1	Principi di Interfaccia Utente	21
3.2.2	Struttura dell'interfaccia	22
3.3	Architettura di Sistema	24
3.4	Realizzazione Prototipo Figma	25
3.4.1	Pagina Iniziale	25
3.4.2	Generazione	26
3.4.3	Personalizzazione	28
3.4.4	Salvataggio	29
3.4.5	Impostazioni	30
4	Implementazione	31
4.1	Tecnologie	32
4.1.1	Frontend	32
4.1.2	Backend	33
4.2	Implementazione Frontend	35
4.2.1	Pagina Iniziale	36
4.2.2	Inserimento Dati e Generazione	37
4.2.3	Personalizzazione Esercizio	42
4.2.4	Esportazione Esercizio	48
4.2.5	Barra di Caricamento	49
4.2.6	Impostazioni	50
4.3	Implementazione Backend	52
4.3.1	Gestione File	53
4.3.2	Generazione Contenuti	56
5	Valutazione	61
5.1	Metodi	61
5.1.1	Selezione dei Partecipanti	62
5.1.2	Prevenzione dei Bias	62
5.1.3	Struttura del Test	63
5.1.4	Casi d'Uso Testati	64
5.2	Risultati Quantitativi	65
5.2.1	Confronto dei Tempi	65
5.2.2	Confronto Aspettative e Realtà	69
5.3	Analisi Qualitativa	71
5.3.1	CAT-AI: Vantaggi e Svantaggi	71
5.3.2	Baseline: Vantaggi e Svantaggi	72
5.3.3	Analisi delle Modifiche	74

6 Conclusioni	78
6.1 Sviluppi Futuri	79
Bibliografia	80

Elenco delle tabelle

2.1	Distribuzione Partecipanti Interviste	6
2.2	Funzionalità del sistema	15
2.3	Requisiti non-funzionali	16
3.1	Principi di Esperienza Utente	18
3.2	Principi di Interfaccia Utente	21
4.1	Funzionalità della barra orizzontale e verticale	43
4.2	Dati Elementi Esercizio	45
4.3	Dati e Impostazioni di Generazione	51
4.4	Routes di Gestione File	55
4.5	Routes di Generazione Contenuti	60
5.1	Tempi Medi Test Italiano e Matematica	65
5.2	Risultato Tempi Test Italiano 1 e 2	66
5.3	Risultato Tempi Test Italiano 3 e 4	67
5.4	Risultato Tempi Test Matematica 1 e 2	68
5.5	Risultato Tempi Test Matematica 3 e 4	68
5.6	Confronto Aspettative e Realta Test	70
5.7	Numero di Rigenerazione per Test	75

Elenco delle figure

3.1	Processo Interazione Utente	19
3.2	Gerarchia Visiva Interfaccia Utente	23
3.3	Architettura di Sistema	24
3.4	Prototipo: Pagina Iniziale	25
3.5	Prototipo: Scelta Modalità	26
3.6	Prototipo: Uploader	26
3.7	Prototipo: Form Modalità 1	27
3.8	Prototipo: Form Modalità 2	27
3.9	Prototipo: Personalizzazione Esercizio	28
3.10	Prototipo: Rigenerazione con IA	29
3.11	Prototipo: Salvataggio Esercizio	29
3.12	Prototipo: Impostazioni di Generazione	30
4.1	Processo Interazione Tecnologie	34
4.2	Architettura Route Frontend	35
4.3	Pagina Iniziale	36
4.4	Scelta Modalità	37
4.5	Inserimento File	38
4.6	File: Cosa Cambiare	39
4.7	File: Cosa Tenere	40
4.8	Inserimento Manuale	41
4.9	Personalizzazione Esercizio	42
4.10	Aggiungi Casella di Testo	44
4.11	Aggiungi Immagine	44
4.12	Rigenerazione con IA	46
4.13	Risoluzione Problemi	47
4.14	Scelta Formato	48
4.15	Esportazione	49
4.16	Impostazioni	50
4.17	Architettura Route Backend	52

5.1	Esempio Esercizi Creati Test Italiano 4	76
5.2	Esempio Esercizi Creati Test Matematica 4	77

Capitolo 1

Introduzione

Al giorno d'oggi, l'innovazione tecnologica si sta diffondendo ampiamente in una grande varietà di ambiti, e l'istruzione non fa eccezione. Anche se l'istruzione, sotto molti aspetti, appare ancora strettamente legata ai metodi di insegnamento tradizionali, il continuo ricambio generazionale porta con sé nuove esigenze e sfide sia per gli studenti che per gli insegnanti.

Da un lato, abbiamo la generazione dei nuovi studenti abituati a interfacciarsi con le tecnologie moderne fin dalla giovane età, assuefatti a input sempre più rapidi. Dall'altro lato, abbiamo gli insegnanti che devono affrontare questa trasformazione, cercando di trovare nuovi modi e strategie per mantenere l'attenzione e la partecipazione degli studenti durante la lezione. Allo stesso tempo, gli insegnanti devono anche confrontarsi quotidianamente con l'elevata eterogeneità delle peculiarità e delle abilità degli studenti, il che rende difficile per loro gestire tutte le possibili necessità.

Per queste ragioni, le soluzioni tradizionali si rivelano spesso troppo rigide, non facilmente personalizzabili o troppo complesse da applicare in breve tempo, rendendole non adatte all'uso quotidiano. Per esempio, per la creazione di esercizi didattici, gli insegnanti di scuole primarie e secondarie di primo grado si devono affidare ancora in gran parte a libri di testo scelti dalla scuola. Questo processo richiede un lungo tempo di ricerca, oltre a limitare fortemente le possibilità creative e di personalizzazione degli esercizi.

Fortunatamente, al crescente numero di sfide nel campo dell'istruzione corrisponde un numero crescente di nuove tecnologie e applicazioni educative a supporto. Ricerche hanno dimostrato che sia insegnanti che studenti comprendono l'opportunità offerta dalle nuove tecnologie basate su IA per arricchire l'insegnamento e favorire la creazione di materiale personalizzato per studenti con diversi livelli di competenza [1].

1.1 Motivazioni

Tuttavia, questi metodi innovativi portano anche a una crescente diffidenza nei confronti della loro reale efficacia e adeguatezza nel contesto educativo. Infatti, queste tecnologie richiedono spesso uno studio approfondito o una conoscenza preesistente dello strumento al fine di poterlo utilizzare correttamente e ottenere il risultato atteso. Inoltre, gli insegnanti spesso faticano a fidarsi dell'output dell'IA, che può essere soggetto a errore, costringendoli a revisionare sistematicamente il processo logico e la sintassi del materiale generato.

I nuovi strumenti generativi, come ChatGPT, pongono l'utente nella condizione di decidere interamente cosa chiedere, ma soprattutto come chiederlo, il che può rivelarsi un'arma a doppio taglio nel contesto educativo. Infatti, anche se i modelli generativi aprono le porte a un numero esteso di possibilità teoriche, non è raro imbattersi in problemi pratici derivanti dalla conoscenza limitata del funzionamento dello strumento. Come prima cosa, la scelta del prompt giusto è spesso la principale difficoltà da superare per ottenere il risultato atteso [2]. A questo proposito, gli insegnanti dovrebbero considerare nel loro prompt troppi fattori per riuscire a creare materiali didattici soddisfacenti e conformi alle proprie aspettative.

Questi problemi tecnici portano spesso gli insegnanti a preferire modalità più tradizionali, come nel caso della creazione degli esercizi, in cui gli insegnanti privilegiano siti online fidati o libri di testo. Inoltre, come anticipato, la scuola si affida in gran parte a materiale didattico cartaceo, mettendo in secondo piano le possibilità offerte dagli strumenti digitali.

Per questa ragione, anche laddove un insegnante decidesse di creare materiale personalizzato tramite intelligenza artificiale generativa, dovrebbe comunque utilizzare un editor di testo come Word o Canva per convertire il materiale creato in formato cartaceo. In aggiunta, i suddetti editor di testo, introducendo un gran numero di funzionalità e variabili per essere adattati a molti campi, risultano spesso difficili e dispendiosi in termini di tempo per gli insegnanti. Per esempio, Canva è uno strumento di progettazione grafica creato per designer e professionisti, e richiede perciò una formazione o esperienza pregressa per un utilizzo agevole. Allo stesso modo, Word, seppur più comune e utilizzato, è uno strumento di formattazione del testo ideato principalmente per professionisti e aziende. Questi fattori influenzano in modo particolare l'usabilità stessa di queste tecnologie, senza considerare il fatto che entrambe richiedono comunque una ricerca o creazione antecedente dei contenuti.

Da qui nasce la necessità di uno strumento "all-in-one" che consenta agli insegnanti di creare facilmente il materiale per la loro classe. Infatti, attraverso il crescente sviluppo di strumenti basati sull'IA generativa e di Intelligent Tutoring Systems per l'istruzione, vi è la possibilità di superare la diffidenza degli insegnanti, oltre che le difficoltà poste dalle tecnologie più comuni sopra citate. Gli ITS

(Intelligent Tutoring Systems) sono ambienti di apprendimento computerizzati che incorporano modelli computazionali provenienti da diversi campi, tra cui sociologia e intelligenza artificiale [3]. Questi ultimi possono offrire un'esperienza di apprendimento specifica per lo studente al fine di promuovere ciascuna capacità di apprendimento e migliorare le prestazioni complessive degli studenti [4]. Allo stesso modo, questi strumenti permettono agli insegnanti di concentrarsi su metodi e approcci didattici personalizzati, consentendo loro di risparmiare tempo nella creazione del materiale educativo.

1.2 Obiettivi

Per questo motivo, l'obiettivo di questa tesi è progettare e sviluppare un sistema per la creazione di esercizi facile e intuitivo basato sull'IA, denominato "CAT-AI" (Classroom Assistant Tutor), in grado di intercettare le esigenze degli insegnanti e convertirle in nuovo materiale didattico pronto all'uso.

Per fare ciò, saranno studiate le abitudini degli insegnanti nei loro modi comuni di creare esercizi, in modo da poter delineare e soddisfare quelli che sono i requisiti e necessità richiesti. Lo strumento oggetto di sviluppo dovrà inoltre consentire agli insegnanti di modificare e personalizzare il materiale generato tramite un editor di testo intuitivo ed essenziale.

L'applicazione sarà principalmente rivolta alla scuola primaria e secondaria di primo grado, con la possibilità di essere utilizzata in un ambito multidisciplinare. Per questo motivo, essa dovrà mantenere un tono chiaro e semplice, in modo da poter incontrare maggiormente il contesto a cui verrà rivolto. L'applicazione dovrà inoltre offrire un'interfaccia user-friendly per permettere anche a insegnanti con poca confidenza con le nuove tecnologie di riuscire a creare il proprio materiale didattico.

Infine, questa tesi si occuperà anche di analizzare la reale possibilità di adattare l'uso di CAT-AI in un contesto concreto, attraverso test di qualità e usabilità sottoposti direttamente agli insegnanti. Verranno inoltre discusse possibili espansioni future per renderlo adattivo agli studenti, affrontare nuove sfide educative e renderlo ancora più intuitivo per gli insegnanti.

1.3 Struttura della Tesi

Fino a ora, il Capitolo 1 ha introdotto il macro tema che andrà ad affrontare la tesi, ossia la creazione di un sistema di tutor intelligente basato su IA per la creazione di materiale didattico. Inoltre, è stata fatta una panoramica su motivazioni e obiettivi della tesi. I seguenti capitoli della tesi possono invece essere sommariamente riassunti come segue.

- Capitolo 2 :** Indagherà i metodi di raccolta e analisi dei dati iniziali a partire da interviste sottoposte direttamente a insegnanti di scuole elementari e medie. Arriverà poi a un'analisi approfondita dei requisiti e bisogni emersi dalle interviste fino alla redazione di una lista di requisiti funzionali e non-funzionali per l'applicazione finale.
- Capitolo 3 :** Mostrerà la progettazione del prototipo a media fedeltà dell'applicazione a partire dai requisiti creati, spiegando le motivazioni delle scelte prese. In seguito, mostrerà invece l'effettiva architettura e visualizzazione del prototipo creato tramite Figma.
- Capitolo 4 :** Approfondirà le tecnologie, tra cui linguaggi di programmazione e framework, utilizzati per lo sviluppo dell'applicazione, sia per la parte frontend che quella backend.
- Capitolo 5 :** Mostrerà il processo vero e proprio di implementazione, con le strategie attuate e il risultato finale ottenuto. Anche in questo caso l'analisi sarà divisa tra sviluppo frontend e backend.
- Capitolo 6 :** Esporrà i metodi e risultati dei test di qualità e usabilità effettuati per l'applicazione. Verranno messi in evidenza dati quantitativi raccolti, oltre che vantaggi e svantaggi emersi durante i test.
- Capitolo 7 :** Conclusioni della tesi, con riassunto di ciò che è stato fatto e considerazioni finali. Saranno inoltre esposti possibili sviluppi futuri per migliorare l'applicazione e le possibilità offerte.

Capitolo 2

Analisi e Definizione dei Requisiti

Per sviluppare un sistema per la creazione di esercizi efficiente che sia realmente in grado di aiutare gli insegnanti in un contesto concreto per la generazione e personalizzazione di materiale didattico, è necessaria una ricerca qualitativa delle loro metodologie e routine.

Per questa ragione, questo capitolo tratterà delle analisi e dei metodi applicati per indagare le necessità degli insegnanti. Verrà mostrato come abbiamo esplorato i problemi comuni e le preoccupazioni tra gli insegnanti e, allo stesso tempo, le loro metodologie standard per la creazione degli esercizi.

Successivamente, verrà presentato come i bisogni comuni degli insegnanti siano stati analizzati e raccolti per creare delle User Personas. Più avanti, invece, verrà mostrato come l'attività di individuazione delle necessità sia stata utilizzata per trasporre le informazioni raccolte in una definizione dei requisiti per lo strumento oggetto di studio. Infine, verrà riportato l'elenco completo dei requisiti funzionali e non funzionali per lo strumento.

2.1 Metodi

Per una comprensione più approfondita delle necessità degli insegnanti, è stata condotta una ricerca qualitativa. In particolare sono stati intervistati insegnanti provenienti da scuole primarie e secondarie di primo grado.

L'obiettivo principale dello studio era comprendere come gli insegnanti operano nella loro routine per creare materiale didattico. Abbiamo ricercato le metodologie standard applicate dagli insegnanti per creare esercizi, ma anche le loro principali fonti e i loro interi processi di pensiero. Volevamo avere una comprensione completa non solo delle loro preoccupazioni, ma anche delle loro difficoltà e sfide quotidiane.

2.1.1 Selezione dei Partecipanti

I possibili candidati dello studio erano perciò insegnanti ed educatori in servizio nelle scuole primarie e secondarie di primo grado. La selezione preliminare è stata effettuata tramite un sondaggio online, condiviso attraverso canali personali.

Il nostro obiettivo era trovare un equilibrio tra gli insegnanti in servizio nella scuola primaria e quelli nella scuola secondaria di primo grado. Allo stesso tempo, tra questi, volevamo sia partecipanti con una certa esperienza nell'uso dell'Intelligenza Artificiale sia altri senza. In questo modo, abbiamo raggiunto un campione esemplificativo di dieci insegnanti, suddiviso come mostrato in tabella.

Parametro	Distribuzione
Per scuola	
Scuola primaria	50%
Scuola secondaria di primo grado	50%
Per genere	
Femmina	90%
Maschio	10%
Per età	
55–64 anni	40%
35–44 anni	40%
25–34 anni	30%
Per esperienza con IA (Likert 1–5)	
3/5	40%
2/5	40%
1/5	30%

Tabella 2.1: Distribuzione Partecipanti Interviste

Perciò, è stato raccolto un campione rappresentativo di insegnanti per indagare adeguatamente metodi e tecniche degli educatori attraverso diverse età, competenze e settori. Le materie di riferimento per ciascun insegnante sono varie, infatti, specialmente nelle scuole primarie è tipico avere uno o due insegnanti che insegnano tutte le materie. A partire da matematica, scienze e italiano, fino a storia, geografia e arte.

Inoltre, analizzando la distribuzione percentuale per genere, che può sembrare sbilanciata, è possibile affermare che essa rispecchia la reale suddivisione nelle scuole italiane; infatti, la scuola primaria conta una percentuale di insegnanti donne di circa il 96%, mentre nella scuola secondaria di primo grado è di circa l'80%. I partecipanti selezionati sono stati poi sottoposti a un'intervista per la ricerca vera e propria.

2.1.2 Contenuti e Struttura dell'Intervista

Per una migliore comprensione delle necessità degli insegnanti, l'uso di un semplice modulo online da compilare può risultare limitante. Infatti, i moduli online sono per lo più utili per ricerche quantitative, offrendo la possibilità di raggiungere un ampio numero di persone, ma d'altro canto limitano la possibilità per i partecipanti di fornire risposte strutturate e motivazioni alle domande poste.

Per questa ragione, è stata utilizzata un'intervista semi-strutturata per avere una piena comprensione delle esigenze degli insegnanti. Allo stesso tempo, analizzare i fattori che gli insegnanti ritengono utili per l'apprendimento degli studenti e necessari nel contesto scolastico. Nello specifico, le interviste sono state condotte tramite una chiamata telefonica, in modo da incontrare più facilmente la disponibilità degli insegnanti, e ciascuna intervista è durata in media 15-20 minuti. L'incipit e le prime domande sono state predisposte per introdurre gli insegnanti agli argomenti trattati e per farli sentire a proprio agio con l'intervista.

Infatti, l'intervista è iniziata con domande semplici, facendo diretto riferimento all'esperienza quotidiana degli insegnanti, agli argomenti recentemente affrontati in classe e alle principali tipologie di esercizi creati nei giorni precedenti. Successivamente, le domande hanno mirato a comprendere più nello specifico gli strumenti comunemente utilizzati, i processi applicati e le caratteristiche degli esercizi richieste per la creazione di materiale didattico. Nelle domande seguenti, abbiamo chiesto un confronto tra strumenti digitali e materiali cartacei, evidenziando vantaggi e svantaggi di ciascun metodo nel contesto scolastico reale. L'intervista ha poi indagato come gli insegnanti valutavano e adattavano il materiale didattico in base alle necessità di una classe eterogenea.

Infine, agli insegnanti è stato chiesto di fornire, ove possibile, file di esempio o fotografie del loro materiale didattico recentemente creato. Grazie a ciò è stato possibile raccogliere un buon numero di file e foto di esercizi creati e proposti direttamente dagli insegnanti ai propri alunni. Il materiale raccolto è stato essenziale anche per comprendere come apparissero visivamente gli esercizi discussi durante l'intervista.

2.2 Bisogni Ricorrenti

Le interviste sono state registrate, previo consenso da parte degli insegnanti, al fine di analizzare meglio e raccogliere tutte le informazioni utili e rilevanti. Anche i file di esempio forniti sono stati analizzati, per avere una piena comprensione delle tipologie comuni di esercizi. Tutti i principali risultati e i bisogni ricorrenti emersi possono essere presentati e suddivisi come segue.

2.2.1 Obiettivi di Apprendimento, Chiarezza e Coerenza

In tutte le interviste, quando è stato chiesto agli insegnanti quali fossero le caratteristiche più importanti richieste per creare un esercizio, tutti hanno concordato su 3 punti chiave: obiettivi di apprendimento, chiarezza e coerenza.

In primo luogo, l'insegnante deve avere un obiettivo di apprendimento ben definito per gli studenti, ad esempio "risolvere equazioni di primo grado" o "riconoscere le parti del discorso". Conoscere quale sia l'obiettivo di apprendimento dello studente è essenziale sia per ricercare materiale didattico appropriato sia per adattarlo alle necessità degli studenti.

Altri due concetti collegati, emersi in tutte le interviste in modi differenti, sono chiarezza e coerenza. Un esercizio deve essere chiaro in tutte le sue parti, a partire dalle istruzioni dell'esercizio, fino al vocabolario utilizzato. Le istruzioni devono essere al tempo stesso complete e concise, e devono ridurre al minimo le possibili incomprensioni. Direttamente connessa alla chiarezza vi è la coerenza, che gioca un ruolo importante nel raggiungimento della chiarezza stessa. La coerenza non riguarda solo gli argomenti, ma si riferisce anche alla tipologia di esercizi, al tono e al vocabolario utilizzato.

Gli esercizi di verifica devono in qualche modo rispecchiare tutti i modelli e i metodi appresi, cercando quindi di incoraggiare il ragionamento. Specialmente nella scuola primaria, la coerenza con quanto è stato insegnato in classe è essenziale per favorire l'apprendimento dei bambini, ad esempio utilizzando la stessa terminologia, le stesse immagini di riferimento o gli stessi colori. Anche per questa ragione, nella scuola primaria gli insegnanti fanno grande affidamento sui libri di testo, che sono considerati il principale punto di riferimento per gli alunni.

2.2.2 Livello di Difficoltà

Un'altra caratteristica ricorrente sottolineata dagli insegnanti quando si parla delle caratteristiche degli esercizi è il livello di difficoltà dell'esercizio. In questo caso le risposte differiscono leggermente tra scuola primaria e scuola secondaria di primo grado.

Per quanto riguarda la scuola primaria, gli insegnanti affermano sempre che “ogni studente della classe deve essere in grado di comprendere e svolgere gli esercizi proposti”. Per questa ragione, il livello di difficoltà deve essere mediamente basso, per permettere a tutti gli studenti di portare a termine gli esercizi, con particolare attenzione da parte dell’insegnante agli studenti con bisogni educativi speciali (BES).

Invece, nella scuola secondaria di primo grado, con il cambiamento degli obiettivi di apprendimento, cambia anche la considerazione sul livello di difficoltà. Infatti, mentre la scuola primaria è principalmente rivolta a far apprendere agli studenti le basi e gli schemi di apprendimento, la scuola secondaria di primo grado mira a mettere alla prova le capacità degli studenti, mantenendo comunque schemi già conosciuti, ma aggiungendo anche nuove possibilità di ragionamento.

Pertanto, gli insegnanti della scuola secondaria di primo grado hanno affermato che esercizi e verifiche devono raggiungere un buon compromesso tra schemi già visti e schemi non visti, utilizzando spesso strategie incrementalì, partendo da un esercizio facile fino ad arrivare a uno più impegnativo.

Tuttavia, in entrambi i casi scolastici la reale misura della difficoltà è impossibile da stabilire se non da parte dell’insegnante, che attraverso la conoscenza individuale degli studenti arriva a un metro di giudizio comune. Anche in questo caso però, gli insegnanti si trovano spesso in difficoltà nel trovare un livello di difficoltà appropriato per tutti gli studenti, costringendoli a trovare compromessi o a sottovalutare le capacità degli studenti.

2.2.3 Varietà degli Esercizi

Continuando a parlare dei bisogni emersi dalle interviste agli insegnanti, un punto importante riguarda la varietà degli esercizi. Infatti, gli insegnanti hanno concordato sul fatto che, per favorire e promuovere l’apprendimento degli studenti nel loro percorso educativo, essi debbano affrontare problemi diversi e applicare metodologie differenti per comprendere meglio un argomento. Ogni tipologia di esercizio può aiutare gli studenti a sviluppare diverse capacità di ragionamento e problem solving.

Specialmente nella scuola secondaria di primo grado, gli insegnanti lamentano il fatto che i libri di testo siano spesso limitati in termini di varietà di esercizi. Per questo motivo, tutti gli insegnanti si sono dichiarati favorevoli all’utilizzo di strumenti digitali o materiali online per ampliare le possibilità di apprendimento. Nonostante ciò, la maggior parte di loro si è mostrata diffidente nei confronti dei materiali online. Verificare le fonti è infatti essenziale per gli insegnanti per garantire l’affidabilità di un determinato materiale didattico.

Inoltre, anche le fonti online considerate affidabili necessitano spesso di un doppio controllo, rendendo la ricerca e la creazione del materiale più lunga e complessa. La maggior parte degli insegnanti intervistati si affida soltanto a uno o due siti o

strumenti online, limitando fortemente l'enorme quantità di possibilità offerte dalle risorse tecnologiche. Tra i siti più utilizzati dagli insegnanti intervistati possiamo trovare quelli istituzionali, ad esempio la versione online estesa del libro di testo cartaceo o il sito Zanichelli. Un altro sito ricorrente utilizzato dagli insegnanti è Wordwall, che oltre a fornire materiali didattici già preparati dagli utenti, consente anche agli insegnanti di creare il proprio materiale partendo da semplici modelli di riferimento e inserendo i contenuti desiderati.

Tuttavia, anche se tutti questi siti menzionati forniscono modi alternativi per ampliare la varietà di esercizi, presentano tutti alcune limitazioni. Per quanto riguarda i materiali online istituzionali, essi hanno quasi gli stessi problemi dei libri di testo cartacei: il processo di ricerca e creazione degli esercizi è dispendioso in termini di tempo e le tipologie di esercizi sono limitate. Allo stesso modo, Wordwall, pur offrendo un buon strumento per la creazione degli esercizi, è comunque necessaria una previa ricerca di contenuti per creare un esercizio e gli esercizi già pronti non corrispondono spesso esattamente a ciò che gli insegnanti vorrebbero.

2.2.4 Coinvolgimento degli Studenti

Infine, un'ultima caratteristica chiave che gli insegnanti tengono in considerazione nella creazione dei loro esercizi è il coinvolgimento degli studenti. Per coinvolgimento si intende che l'esercizio deve in qualche modo favorire l'apprendimento degli studenti stimolandoli da più punti di vista.

In particolare, gli insegnanti della scuola primaria, durante l'intervista, si sono concentrati sulla necessità di mantenere l'attenzione dei bambini in classe. Infatti, nella nuova era tecnologica, i bambini sono continuamente esposti a input rapidi, richiedendo allo stesso modo un costante input anche a scuola. Gli insegnanti della scuola primaria, che devono fornire l'educazione iniziale ai bambini, cercano di superare questo problema introducendo nuove metodologie di insegnamento.

Questo coinvolgimento è spesso raggiunto attraverso l'organizzazione di attività di gruppo personalizzate, l'uso delle tecnologie digitali o semplicemente tramite la creazione di esercizi interattivi. Gli insegnanti hanno affermato che è importante che i bambini apprendano anche tramite strumenti tecnologici, strettamente collegati alla loro vita quotidiana. Nonostante ciò, essi sostengono ancora che il materiale cartaceo possa essere sostituito solo parzialmente da quello digitale, perché i bambini esprimono anche il bisogno di avere qualcosa di fisico tra le mani per migliorare la loro manualità nella scrittura o nel disegno, incoraggiando la creatività.

Direttamente collegato al coinvolgimento degli studenti, quando si parla di creazione di esercizi, vi è il bisogno di creare esercizi giocosi e colorati, sia nel tono sia nella loro struttura. Anche il layout dell'esercizio è importante per catturare l'attenzione degli studenti, e la presenza di un'immagine illustrativa è spesso richiesta dagli insegnanti per una migliore comprensione.

2.3 Raggruppamento dei Requisiti

Una volta analizzate le risposte degli insegnanti e le necessità emerse durante le interviste, il passo successivo per la definizione dei requisiti è stato il loro raggruppamento. Infatti, per poter avere delle informazioni chiare e solide su cui basare i requisiti dell'applicazione finale, è necessario prima creare una sintesi delle informazioni più rilevanti.

Inoltre, direttamente collegata alla sintesi delle informazioni, c'è la comprensione di chi sia l'utente finale a cui si rivolgerà l'applicazione. Comprendere i vari tipi di possibili utenti finali è necessario al fine di una progettazione mirata e efficace. Per adempiere tale fine, è stata attuata la tecnica degli User Personas, che consiste nella definizione di modelli di persona fittizi che saranno idealmente i fruitori principali del nostro prodotto finale.

2.3.1 User Personas: Cosa e Perché

Durante la progettazione di un prodotto è spesso difficile riuscire a incontrare i bisogni specifici di ogni utente. Per questa ragione, la soluzione migliore è spesso quella di comprendere i bisogni comuni e unirli per avere una migliore visione dell'insieme. Proprio a questo scopo, vengono usate le User Personas, nel libro "About Face: The Essential of Interaction Design" esse vengono definite come "archetipi composti basati sui modelli di comportamento scoperti durante il corso della nostra ricerca" [5]. In altre parole, ogni Personas deve rappresentare un modello ben definito di utente, con bisogni e necessità specifiche.

Queste necessità specifiche devono essere però specchio di uno schema ricorrente ritrovato negli utenti durante le interviste. Infatti, per una creazione efficace delle Personas è necessario attuarne una personificazione, ossia ognuna di esse deve avere un ruolo ben preciso, degli obiettivi, ma soprattutto delle problematiche da risolvere.

Spesso è utile anche dare dei nomi alle Personas, oltre che creare degli scenari verosimili in cui la Personas è portata a utilizzare la nostra applicazione. Questo processo di personificazione è utile al fine di far immedesimare al meglio il designer negli utenti a cui si rivolgerà l'applicazione, per arrivare in seguito a una definizione dei requisiti mirata e efficace. Nel nostro caso specifico, i principali fruitori saranno gli insegnanti di scuole primarie e secondarie di primo grado, perciò a partire da questa considerazione le Personas dovranno andare a modellare i loro bisogni utilizzando i dati raccolti durante le interviste.

2.3.2 Definizione delle Personas

Durante il processo di definizione delle Personas, sono stati creati 4 archetipi di utente a partire dalle 10 persone intervistate. Questi modelli sono stati creati in modo da rispecchiare al meglio le esigenze emerse sia da parte degli insegnanti di scuole primarie che di secondarie di primo grado. Le 4 User Personas sono state definite come segue, divise in base al loro ruolo.

Tradizionalista Moderata : Maestra Maria, 58 anni di cui 33 di insegnamento, insegna le materie scientifiche e letterarie nella scuola primaria. Per lei il libro di testo e gli strumenti cartacei sono le fonti principali per la creazione di esercizi, ma è sempre alla ricerca di nuovi esercizi creativi che possano attirare l'attenzione e stimolare l'apprendimento dei bambini e mantenere vivo l'interesse e l'interattività. Per lei ogni esercizio necessita di soddisfare ed essere funzionale a un determinato obiettivo di apprendimento (es. imparare la tabellina del 5, imparare quando una parola deriva da un'altra). Inoltre, per lei avere consegne brevi e chiare è fondamentale per avere un compito adatto ai suoi alunni, e ci devono sempre essere degli esempi per far comprendere meglio l'esercizio. Anche calibrare la difficoltà in base alla classe è fondamentale, tutti gli alunni devono essere in grado di fare l'esercizio, perciò bisogna partire da un livello alla portata di tutti.

Tutor Multidisciplinare : Maestro Giulio, 31 anni di cui 6 di insegnamento ha insegnato in scuole primarie e secondarie di secondo grado come insegnante di sostegno. Trattando diametralmente tutte le materie è sempre alla ricerca di nuovi esercizi da più fonti fidate possibili. Inoltre, aiutando alunni con bisogni educativi speciali (BES) ha necessità di creare esercizi semplici, ben strutturati, e visivamente stimolanti. Per lui infatti, gli elementi visivi quali immagini, tabelle o grafici sono fondamentali, così che possano aiutare a comprendere e svolgere l'esercizio stimolando più sensi. Anche l'impaginazione dell'esercizio è importante per non perdere la concentrazione degli alunni. Per non far sentire esclusi gli alunni con BES o ADHD, spesso utilizza gli stessi esercizi della classe ma leggermente semplificati (es. numeri più piccoli, o esercizi meno lunghi). La sua difficoltà principale è trovare materiale online personalizzato adatto e affidabile per i suoi alunni. La maggior parte delle volte crea materiale manualmente su editor di testo come Word o Canva, il che risulta molto dispendioso in termini di tempo.

Progressista Coscienziosa : Professoressa Carla, 41 anni e 13 di insegnamento, insegna italiano, storia e geografia nella scuola secondaria di primo grado. Nella creazione di esercizi calibra l'utilizzo di strumenti cartacei e digitali per favorire un apprendimento multilaterale. Per lei il libro di testo è essenziale, ma essendo quest'ultimo spesso carente di materiale ricerca spesso online

del nuovo materiale personalizzato per creare temi e domande che possano stimolare il ragionamento degli alunni. La caratteristica principale che deve avere qualsiasi esercizio è la chiarezza della consegna e delle domande. Per fare ciò, utilizza il linguaggio del libro di testo, che spesso aiuta gli alunni nella comprensione. Il tempo necessario per creare il materiale didattico è spesso un problema, infatti il materiale online seppur vario richiede una verifica minuziosa delle fonti per accertarsi dell'affidabilità. Inoltre, applicazioni e siti per supportare la creazione di esercizi richiedono spesso abbonamenti per una maggiore funzionalità.

Innovativa Sperimentale : Professoressa Lucia, 30 anni, 4 anni di insegnamento, insegna matematica e scienze alle scuole medie. Per il suo materiale didattico, utilizza ed è sempre alla ricerca di siti o applicazioni per supportare la creazione di esercizi. Nel suo quotidiano, essa necessita una buona varietà di esercizi che possano favorire modi diversi di imparare la matematica e le scienze. Allo stesso tempo, vuole ridurre il carico di lavoro e il tempo impiegato per crearlo. Anche per lei la consegna dell'esercizio deve essere chiara e senza margini di interpretazione, la matematica è una scienza esatta, e il linguaggio del libro è spesso quello meglio compreso dagli alunni poiché quotidiano. L'argomento dell'esercizio è il punto focale che indirizza la loro creazione, ed è sempre anche vincolato alle competenze da raggiungere o verificare. Per la sua materia ha spesso necessità di creare esercizi con difficoltà graduale, da facili, a intermedi a più complicati.

2.3.3 Considerazioni su Personas e Requisiti

Dalla definizione delle User Personas sono perciò emerse 4 figure di riferimento con necessità simili ma per contesti diversi. Infatti, per tutte il punto di partenza è il voler creare l'esercizio, ma ciascuno con bisogni e contesti paralleli.

Le figure create rispecchiano tutte le necessità emerse durante le interviste e contemporaneamente le sintetizzano. Da queste possiamo derivare una serie di parametri che dovrà rispettare l'applicazione. Le caratteristiche sono parallele alle necessità, e possono essere elencate come segue.

Obiettivi di apprendimento : Ogni esercizio generato deve sempre essere vincolato all'obiettivo di apprendimento scelto dall'insegnante. Questo non solo esprime la materia di riferimento, ma esprime anche cosa gli alunni devono imparare attraverso l'esercizio che andrà creato. Esso è anche la cosa più familiare e facile da trovare per gli insegnanti, che quotidianamente partono dai loro obiettivi per gli alunni per trovare e creare materiale didattico. Parallelamente ad essi, anche i prerequisiti sono utili per dare un'idea di cosa ci si aspetta che invece l'alunno sappia già.

Chiarezza e Coerenza : Queste due caratteristiche devono essere soddisfatte sotto più punti di vista per rendere un esercizio adatto per gli insegnanti. Per quanto riguarda l'esercizio in se, esso deve avere sempre un titolo e una consegna chiara, e se diviso in più punti, ogni punto deve spiegare all'alunno cosa fare, anche avendo la possibilità di inserire esempi già fatti. Per quanto riguarda la coerenza, essa non è legata solo alla coerenza dell'esercizio rispetto agli obiettivi di apprendimento, ma anche alla coerenza nel linguaggio desiderato e nella tipologia e struttura degli esercizi. Per tal fine, è utile che l'utente possa inserire un riferimento fidato nell'applicazione per la creazione di un nuovo esercizio.

Livello di Difficoltà : L'insegnante deve essere in grado non solo di specificare la difficoltà generale dell'esercizio, per esempio facile, medio o difficile, ma anche di poter fare riferimento alla scuola e classe di insegnamento. Infatti, un insegnante che vuole creare un esercizio per la terza media avrà necessità diverse da un insegnante delle elementari e viceversa.

Varietà degli esercizi : L'insegnante, oltre ad avere la possibilità di inserire un riferimento per avere tipologie di esercizi simili a quelle già utilizzate, dovrà avere la possibilità di creare esercizi completamente nuovi senza alcun tipo di riferimento. In questo ultimo caso l'applicazione dovrà basarsi unicamente su obiettivi di apprendimento, difficoltà e altre informazioni generali desiderate. In questo modo, l'applicazione dovrà permettere due metodologie parallele per la creazione di esercizi, la prima con un riferimento in favore di una creazione più sicura e attinente a materiale già esistente, nel secondo caso senza riferimento in favore di una maggiore variabilità.

Coinvolgimento visivo : Mentre il coinvolgimento emotivo è un fattore strettamente legato alla figura umana dell'insegnante e al suo metodo quotidiano di insegnamento, un fattore che può essere affrontato dall'applicazione è quello di coinvolgimento visivo. L'insegnante deve poter creare esercizi con stili differenti, salvare i proprio stili preferiti, ma anche inserire immagini o crearle direttamente tramite l'applicazione. L'applicazione deve fornire gli strumenti per avere una struttura e impaginazione dell'esercizio semplice e efficace.

Tempistiche : Infine, oltre a una creazione semplice e efficace, un obiettivo che dovrà andare a raggiungere l'applicazione è il far risparmiare tempo agli insegnanti per creare materiale didattico. Infatti, l'applicazione risparmierà agli insegnanti il tempo necessario a ricercare materiale online o tramite libri di testo, perché esso verrà direttamente creato dall'applicazione. Allo stesso modo, risparmierà tempo nelle fasi di modifica, tramite un semplice editor di testo, e di esportazione e salvataggio in PDF.

2.4 Definizione dei Requisiti

2.4.1 Requisiti Funzionali

ID	Nome
FR1	Inserimento Specifiche Esercizio
FR1.1	Inserimento requisiti dell'esercizio
FR1.2	Inserimento prerequisiti dell'esercizio e opzioni aggiuntive
FR1.3	Allegato file di riferimento (opzionale)
FR1.4	Estrazione testo da file allegato
FR2	Impostazioni di Generazione
FR2.1	Definizione difficoltà generale
FR2.2	Definizione numero di esercizi da generare
FR2.3	Scelta lingua di generazione
FR2.4	Definizione tono del vocabolario
FR2.5	Scelta stile/colori dell'esercizio
FR3	Generazione con IA
FR3.1	Estrazione informazioni da file di riferimento
FR3.2	Generazione esercizi e soluzioni
FR3.3	Rigenerazione esercizi e/o soluzioni
FR3.4	Rigenerazione di elementi singoli
FR3.5	Generazione di immagini a supporto
FR4	Editing Manuale Esercizi
FR4.1	Modifica contenuto del testo
FR4.2	Modifica stile e formattazione del testo
FR4.3	Gestione posizione e dimensione degli elementi
FR4.4	Cancellazione o aggiunta di elementi
FR5	Salvataggio ed Export
FR5.1	Salvataggio esercizio e soluzioni nel sistema
FR5.2	Scelta layout di esportazione

Tabella 2.2: Funzionalità del sistema

Le interviste e le analisi qualitative portate avanti in questo capitolo hanno permesso la redazione dei requisiti funzionali mostrati nella tabella precedente. Le funzionalità seguono in modo parallelo le necessità emerse e le considerazioni fatte a seguito della creazione delle Personas. I requisiti funzionali sono divisi per macro-aree in modo da avere una comprensione migliore delle possibilità che offrirà l'applicazione oggetto di studio.

Possiamo notare come, per esempio, la necessità di avere sia degli obiettivi di apprendimento, che una difficoltà ben precisa sia stata tradotta nei requisiti tramite l'inserimento delle specifiche (FR1), che prevederà verosimilmente l'inserimento di parole chiave ben chiare all'insegnante. Invece, la generazione tramite AI (FR3), insieme alle impostazioni di generazione (FR2) andranno a favorire una maggiore varietà di esercizi, guidati dai requisiti inseriti. Allo stesso modo, questi ultimi, insieme alla possibilità di modifica manuale (FR4), permetteranno di creare esercizi chiari e coerenti sia visivamente e dal punto di vista dei contenuti. Infine, il salvataggio immediato tramite l'applicazione (FR5) permetterà di ridurre ulteriormente i tempi di creazione, aiutando gli insegnanti a preparare delle schede pronte da stampare.

2.4.2 Requisiti Non-Funzionali

Allo stesso modo, per affrontare le difficoltà e le necessità emerse dall'analisi portata avanti, sono stati scelti dei requisiti non funzionali che l'applicazione dovrà tentare di rispettare. Questi requisiti fissano uno standard qualitativo da raggiungere per rendere l'applicazione effettivamente utilizzabile anche in contesti educativi reali.

ID	Tipologia	Descrizione
NFR1	Usabilità	L'interfaccia deve garantire un tempo medio di creazione di un esercizio di 9-10 minuti per un nuovo utente
NFR2	Efficienza	L'ITS deve garantire un tempo di generazione inferiore a 30 sec nel 90% dei casi
NFR3	Accuratezza	L'ITS deve garantire che il 90% degli esercizi siano valutati come chiari e coerenti dagli insegnanti
NFR4	Affidabilità	Qualora ci fossero errori di contenuto nell'esercizio l'ITS deve segnalarlo per la revisione

Tabella 2.3: Requisiti non-funzionali

Capitolo 3

Prototipo a Media Fedeltà

Una volta definiti i requisiti dell'applicazione, il passo successivo, nel processo di creazione dell'applicazione, è la progettazione del modello a media fedeltà. Per modello a media fedeltà si intende la prototipazione di un prodotto o di un servizio digitale che non rappresenta però ancora una versione finale del prodotto stesso. Esso offre però maggiori dettagli rispetto a un modello a bassa fedeltà, che invece è un prototipo spesso rappresentato da un semplice disegno su carta dell'interfaccia, con le sue generalità. Infatti, un modello a bassa fedeltà viene utilizzato maggiormente per avere un'idea generale dell'architettura del sistema più che per avere una visione delle sue funzionalità. Al contrario, il prototipo a media fedeltà include elementi visivi, struttura di navigazione e simulazioni di interazione più complesse per esplorare il flusso e le funzionalità del prodotto senza dare troppa attenzione ai dettagli grafici finali.

Per fare ciò, è stato utilizzato Figma, una piattaforma web sviluppata direttamente per progettare e prototipare interfacce utente, siti web e applicazioni. In questa fase, è stato inoltre scelto come nome per l'applicazione oggetto di sviluppo "CAT-AI" (Classroom Assistant Tutor): l'assistente tutor della classe basato su AI. Perciò, a seguire nel capitolo verranno espone le scelte e le motivazioni di progettazione per CAT-AI, fino ad arrivare al prototipo a media fedeltà terminato.

3.1 Scelte di Progettazione: Esperienza Utente

Prima di passare alla progettazione vera e propria dell'applicazione è necessario effettuare alcune considerazioni. In particolare, per raggiungere la creazione di un prototipo efficace che possa rendere l'applicazione facile e intuitiva, sono stati seguiti i principi e le regole fondamentali della progettazione UX, ossia relativa all'esperienza utente (User Experience). Verranno perciò mostrati, oltre ai principi per l'esperienza utente seguiti, anche le scelte progettuali mirate a soddisfarli.

3.1.1 Principi di Esperienza Utente

Con progettazione basata su esperienza utente si fa riferimento a tutta una serie di accorgimenti da fare al momento della progettazione, mirati a rendere l'applicazione intuitiva, accessibile, efficiente e piacevole per l'utente. Il punto chiave del UX design, sta proprio nella centralità dell'utente [6].

Per questo motivo, la scelta delle caratteristiche e funzionalità dell'applicazione finale deve essere fatta sulla base dell'utente stesso. Nel nostro caso, la progettazione dell'esperienza utente ha già fondato le sue basi nel capitolo precedente, in cui sono stati indagati problemi e necessità degli insegnanti. Di seguito, sono perciò riassunti e elencati i principi fondamentali seguiti per la progettazione di esperienza utente.

Principio	Descrizione
Utente al Centro	La progettazione deve mettere al centro le esigenze e gli obiettivi dell'utente.
Facilità d'uso	L'interazione e la navigazione all'interno dell'applicazione deve essere logica e prevedibile.
Chiarezza	Le istruzioni, le descrizioni e gli elementi interattivi dell'applicazione devono avere diciture chiare e comprensibili.
Coerenza	Le interazioni e le funzionalità devono essere coerenti con ciò che l'utente conosce e a cui è abituato.
Controllo Utente	L'utente deve avere la capacità di annullare, ripetere o guidare le azioni all'interno dell'applicazione.

Tabella 3.1: Principi di Esperienza Utente

3.1.2 Processo a 3 Fasi

La prima scelta progettuale riguarda l'interazione generale tra utente e applicazione. Per fare in modo che l'applicazione sia facile e intuibile per la creazione di esercizi, è stata scelta una struttura di interazione guidata a fasi. Infatti, è stato dimostrato che, in sistemi assistiti da IA, è utile per l'utente avere una decomposizione dei compiti da fare per avere una maggiore usabilità oltre che una continua verifica della risposta da parte del sistema [7]. In ogni fase l'utente sarà perciò portato a svolgere un obiettivo, fino ad avere come risultato l'esercizio finale scaricato in formato standard (PDF). Perciò, l'interazione utente è stata divisa in 3 fasi che possono essere descritte come segue.

1. **Imposta:** L'utente in questa prima fase dovrà scegliere la modalità con cui vuole generare l'esercizio. Le modalità di generazione previste, come anticipato nell'analisi dei requisiti, sono due: la prima prevede il caricamento di un file con un esercizio di riferimento, la seconda è invece prevede l'inserimento manuale delle caratteristiche. In entrambi i casi l'utente potrà inserire le caratteristiche desiderate, quali obiettivi di apprendimento, prerequisiti, scuola, classe e altre informazioni opzionali. Questa fase termina con la generazione dell'esercizio da parte dell'IA a partire dalle informazioni inserite.
2. **Personalizza:** La seconda fase inizia una volta che l'esercizio e la corrispettiva soluzione sono stati generati. A questo punto l'utente avrà la possibilità di verificare, modificare o rigenerare l'intera esercitazione o parti di essa in base alle necessità. Questa fase dà più libertà all'utente, oltre che una maggiore possibilità di pensiero critico e azione diretta su ciò che è stato generato dall'IA. La modifica degli elementi potrà anche essere supportata dall'IA stessa, tramite prompt personalizzati. La seconda fase termina quando l'utente è soddisfatto del risultato raggiunto e decide di esportare l'esercizio.
3. **Esporta:** Nell'ultima fase l'utente potrà avere un'anteprima dell'esercizio finale in formato foglio A4 pronto per essere salvato nel sistema. Ma prima di fare ciò avrà la possibilità di scegliere tra diversi formati utili, tra questi il formato con l'intestazione da verifica (nome, cognome, classe) o il formato che comprende solo la soluzione. L'ultima fase termina con il salvataggio del formato desiderato.



Figura 3.1: Processo Interazione Utente

3.1.3 Editor User-Friendly

Un altro punto fondamentale nella progettazione dell'applicazione sta nel creare un editor per gli esercizi facile e intuitivo. Per adempiere a tale risultato, è stato adottato un approccio denominato “quello che vedi è quello che ottieni” (dall'inglese “what you see is what you get”). Questo approccio consiste nel fornire all'utente un'interfaccia che permetta di avere una visualizzazione grafica e visiva equivalente al prodotto finito che otterrà.

Nel caso degli editor digitali, gli utenti hanno la possibilità di creare esercizi con impaginazioni personalizzate senza la preoccupazione che esse vengano modificate o alterate in fase di esportazione. In questo modo, viene rafforzata non solo l'usabilità del tool ma anche la soddisfazione nell'output da parte degli utenti [8]. Infatti, questa stessa scelta di progettazione è anche adottata da tutti gli editor grafici e di testo più conosciuti, quali Word e Canva. Questi due editor in particolare, sono stati nominati direttamente dagli insegnanti intervistati, tra gli strumenti utilizzati per la creazione degli esercizi. A tal proposito, nelle interviste gli insegnanti hanno fatto spesso notare limiti e difficoltà nel loro utilizzo.

L'editor del nostro tool, per permettere all'utente di ottenere un'impaginazione personalizzata in modo più semplice, dovrà adottare un'interfaccia di tipo Drag-and-Drop, similmente a Canva. Entrando nel particolare, ogni esercizio creato dovrà essere diviso in sotto-elementi (es. titolo, consegna, domande, etc.) che possano essere trascinati e rilasciati all'interno del foglio di lavoro. Però, a differenza di Canva, che è un tool grafico di progettazione avanzata, la nostra applicazione avrà l'obiettivo di essere facilmente utilizzabile da chiunque, e soprattutto specializzata unicamente nella creazione di esercizi didattici.

Infatti, nonostante Word e Canva siano spesso molto utilizzati in una grande varietà di ambiti lavorativi, essi hanno un limite dato proprio dal fatto che contengono un'ingente quantità di funzioni. Di conseguenza, per un loro utilizzo ottimale è necessaria un'esperienza e una buona conoscenza delle loro funzionalità. Infatti, nonostante anch'essi stiano implementando nuove possibilità tramite intelligenza artificiale, mantengono in gran parte i limiti di usabilità citati.

Per questi motivi, il nostro tool avrà l'obiettivo di esemplificare le funzionalità principali e unire i vantaggi dati da entrambi gli editor menzionati, per ottenere un editor grafico e di testo intuitivo e facilmente utilizzabile per la creazione di esercizi didattici. La diretta conseguenza di questa scelta progettuale è che il tool avrà un editor con funzionalità ridotte rispetto agli editor sopracitati, in favore di una migliore usabilità e chiarezza.

3.2 Scelte di Progettazione: Interfaccia Utente

Una volta effettuate le considerazioni riguardanti il flusso di interazione base del sistema e l'esperienza utente, il prossimo passo riguarda la progettazione dell'interfaccia utente. Tramite uno studio analogo a quello effettuato per l'esperienza utente, anche per le scelte sull'architettura dell'interfaccia è stato fatto riferimento ai principi base del UI (User Interface) design. In un'applicazione web, l'impatto visivo è fondamentale per il raggiungimento di un sistema digitale efficiente. L'effetto estetico e di usabilità (aesthetic-usability effect) mostra che le interfacce visivamente accattivanti sono percepite come più usabili dagli utenti, con un'alta correlazione tra i due fattori [9]. Insieme a questo, tanti altri fattori architetturali e implementativi possono influenzare l'usabilità del tool.

3.2.1 Principi di Interfaccia Utente

Di seguito possiamo trovare un elenco dei principi chiave di progettazione dell'interfaccia seguiti in tabella 3.2 [6].

Principio	Descrizione
Semplicità	L'applicazione deve essere semplice nel design sia nella sua architettura che nelle sue funzionalità. Questo fa in modo che l'utente possa raggiungere facilmente i suoi obiettivi tramite il tool.
Consistenza	Tutti gli elementi dell'interfaccia devono essere coerenti nel loro stile alla loro funzionalità oltre che richiamare schemi conosciuti dagli utenti. L'applicazione per essere facilmente navigabile deve essere familiare all'utente.
Feedback	Il sistema deve sempre dare una risposta visiva all'utente a ogni azione fatta. Tra i feedback possono essere compresi messaggi informativi o di avvertenza, animazioni o cambi di colore.

Tabella 3.2: Principi di Interfaccia Utente

Dal punto di vista pratico, per riuscire a raggiungere un'interfaccia utente semplice, consistente e con un feedback visivo immediato da parte del sistema, occorre fare particolare attenzione ai seguenti 3 elementi di design.

Il primo elemento preso in considerazione è la tipografia. Per tipografia si intende l'insieme di grandezza del testo, stile del carattere e contenuto testuale. In particolare il testo all'interno dell'applicazione deve creare uno schema gerarchico visuale, guidando l'attenzione dell'utente sulle scritte più rilevanti (es. il titolo). Il font, allo stesso modo, può influire sul tono dell'applicazione stessa, dando uno

stile giocoso o formale a seconda dello stile del carattere scelto. Infine, il contenuto testuale di tutti gli elementi dell'interfaccia come bottoni o messaggi deve essere chiaro e non fraintendibile. La scelta delle parole deve essere semplice e mirata a ridurre il carico cognitivo dell'utente, evitando possibili fraintendimenti.

Il secondo elemento essenziale nel UI design è il colore. Quest'ultimo può influenzare emozioni, umore e comportamenti all'interno del sistema. Il colore può essere usato per evidenziare alcuni elementi rispetto ad altri, oppure per indicare la funzionalità di un elemento, richiamando ad associazioni tipiche conosciute dall'utente. Anche il colore come la tipografia influisce nel tono finale che assumerà l'applicazione.

Infine l'ultimo elemento da prendere in considerazione sono le illustrazioni scelte e inserite nell'applicazione. Infatti, le immagini e icone possono essere usate per aggiungere interesse e stimolo visuale nei confronti del sistema. Esse possono aiutare a guidare l'attenzione dell'utente, o indicare in modo più immediato, insieme al testo, le funzioni degli elementi. Le illustrazioni insieme ai due fattori menzionati precedentemente delinearanno l'aspetto finale e il tono assunto dall'applicazione.

3.2.2 Struttura dell'interfaccia

La struttura dell'interfaccia è stata creata in modo da avere un'organizzazione visivamente gerarchica. Per raggiungere questo obiettivo è stata attuata una disposizione centrale degli elementi. In questo modo, lo sguardo dell'utente può concentrarsi sul processo di creazione, senza distrazioni. Sono state perciò evitate barre laterali, posizionando gli elementi al centro dello schermo verticalmente. La gerarchia all'interno dell'interfaccia è creata anche tramite un insieme di altri fattori tra cui: lo spazio occupato, il colore e il carattere testuale. Di seguito, sono elencati gli elementi strutturali in ordine visivo dall'alto al basso. La struttura si riferisce alle tre fasi del processo, per cui sono escluse l'interfaccia iniziale e le impostazioni.

- **Breadcrumb:** Per permettere all'utente una navigazione semplice all'interno dell'applicazione è stato introdotto un breadcrumb. Essendo un elemento secondario esso avrà un font e un colore meno acceso. Il breadcrumb è utile anche per evidenziare la pagina corrente oltre che poter facilmente tornare indietro nella navigazione.
- **Titolo:** Ogni sezione del processo ha un titolo ben visibile, breve ed esplicativo della pagina attualmente visualizzata. Gerarchicamente il titolo si trova al primo posto tra gli elementi dell'interfaccia, grazie alla scelta di un carattere del testo grande, oltre che il suo posizionamento in alto nella pagina.
- **Stepper:** Per rendere l'utente ancora più consapevole del punto a cui si trova nel processo di creazione dell'esercizio, è stato introdotto un tracciatore delle fasi. In questo modo l'utente può vedere subito che il processo è composto da 3

fasi e seguire il progresso man mano che naviga l'applicazione. In questo modo viene prevenuta anche la possibile frustrazione dell'utente data da processi lunghi o di cui non si conosce la durata.

- **Contenuto Interattivo:** Al secondo posto della gerarchia, subito dopo il titolo, c'è il contenuto della pagina, questo macro-elemento conterrà il processo di creazione dell'esercizio con cui l'utente dovrà interagire. Per questo motivo esso è quello che occupa più spazio, in modo da permettere all'utente un'interazione agevole e intuitiva.
- **Call-to-action:** Infine, l'ultimo elemento molto importante in un'interfaccia utente è il "call to action" (chiama all'azione). Questo elemento consiste in un bottone che permette all'utente di navigare da una pagina all'altra dell'applicazione. Per garantire un processo di creazione fluido e intuitivo questo bottone deve essere non solo colorato ma sempre ben visibile.

Per quanto riguarda invece la struttura della pagina iniziale, essa contiene un titolo e una breve descrizione per attirare l'utente e spiegare in breve cosa può fare l'applicazione e un call-to-action colorato e ben evidente per portare l'utente a iniziare il processo di creazione. Infine, la pagina di impostazioni è invece creata tramite un modale (finestra che appare sopra il contenuto principale), ed è accessibile da qualsiasi pagina tramite la barra di intestazione dell'applicazione. Essa è inoltre divisa in sezioni per essere facilmente visualizzabile e utilizzabile.

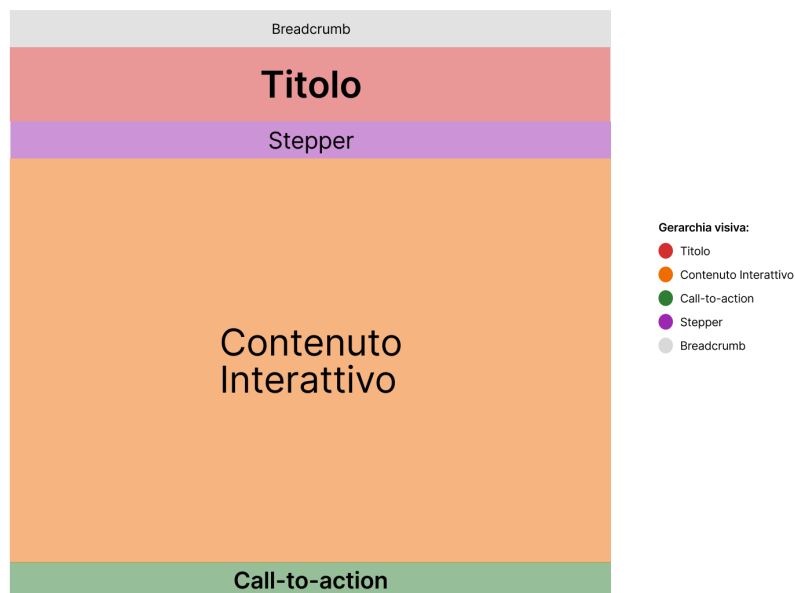


Figura 3.2: Gerarchia Visiva Interfaccia Utente

3.3 Architettura di Sistema

In fase di progettazione sono state anche poste le basi per l'implementazione stessa dell'intera applicazione, ideando la logica di interazione tra i diversi moduli del sistema. In particolare, l'architettura del sistema sarà basata su una comunicazione client-server standard, con l'aggiunta di una terza risorsa esterna rappresentata da OpenAI.

Il lato client sarà la parte dell'architettura a diretto contatto con l'utente, il quale potrà comunicarvi tramite l'interfaccia. Quest'ultima, è l'elemento che permetterà all'utente di interagire con la logica interna fornita dal client stesso. La logica comprende tutta la parte di codice dedita a mantenimento e gestione di variabili e comunicazione con il server. Per quanto riguarda le variabili principali, ossia i requisiti inseriti dall'utente, essi saranno gestiti tramite context e salvati in memoria locale. In questo modo non sarà necessario l'utilizzo di un database esterno, semplificando la logica di mantenimento dei dati temporanei.

La logica del lato client si occuperà di comunicare con il server per implementare la logica funzionale del sistema. Quest'ultima, si occuperà di due mansioni principali: la prima è quella di gestire i file, e la seconda è quella di comunicare con il server di OpenAI per la generazione di contenuti. La gestione dei file si occuperà di importazione e esportazione di tutti i file tramite una cartella dedita denominata "uploads". Parallelamente l'API di OpenAI permetterà di fornire la logica intelligente al sistema.

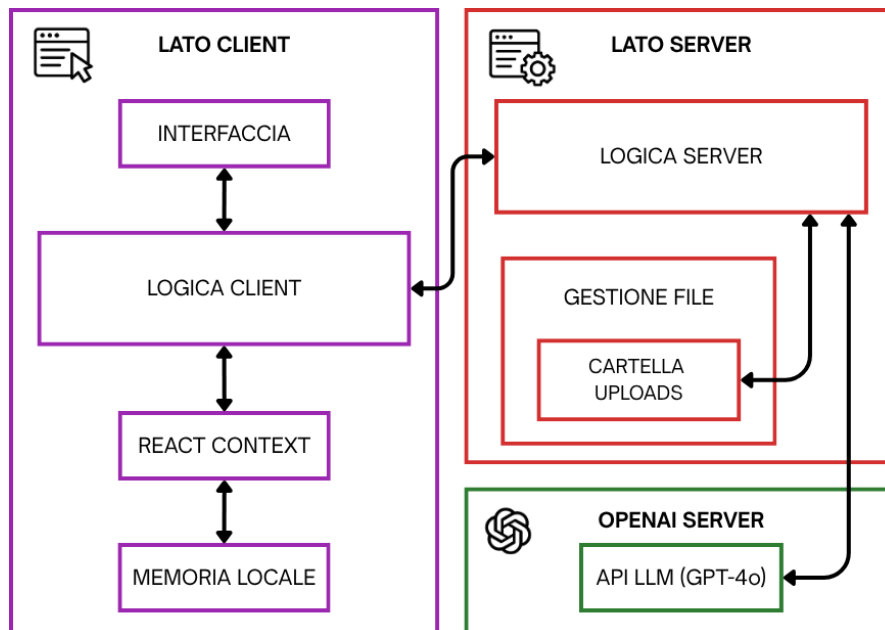


Figura 3.3: Architettura di Sistema

3.4 Realizzazione Prototipo con Figma

Come anticipato nell'introduzione al capitolo, per sviluppare il prototipo a media fedeltà è stato usato Figma, un potente tool per designer che permette di progettare interfacce digitali online. Nel particolare, per evitare di dover creare da zero tutti i componenti dell'interfaccia e assicurarsi una coerenza tra colori, componenti e interazioni, si è fatto affidamento alla libreria Material UI. Quest'ultima è una libreria che offre componenti per interfacce web già pronti e distribuiti gratuitamente per aiutare designer e sviluppatori in uno sviluppo più veloce ed efficace. Material UI si basa sullo stile grafico creato da Google per le sue applicazioni e servizi, creando dei componenti con un aspetto grafico coerente e intuitivo. Inoltre, questi componenti possono essere facilmente trasportati nel codice di sviluppo tramite l'omonima libreria di componenti React. Di seguito vengono mostrati i prototipi per ogni parte dell'interfaccia ideata. Tutti i prototipi mostrati, essendo a media fedeltà, potrebbero cambiare leggermente nella fase di implementazione.

3.4.1 Pagina Iniziale

Come mostrato in Figura 3.3, la pagina iniziale ha una struttura essenziale, composta da un titolo accattivante, una breve descrizione e un call-to-action evidenziato da un blu acceso. Inoltre, è stato creato un logo provvisorio, mantenendo la palette di colori data da Material UI.

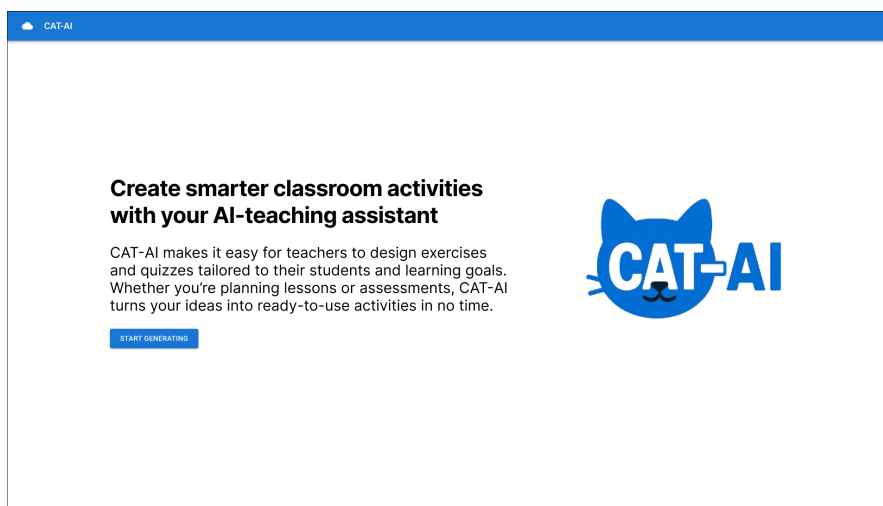


Figura 3.4: Prototipo: Pagina Iniziale

3.4.2 Generazione

Nella figura 3.4, possiamo vedere la schermata di scelta modalità che inizia la fase di generazione. La prima modalità permette di creare esercizi a partire da un file di riferimento, la seconda permette la creazione tramite inserimento manuale dei requisiti.

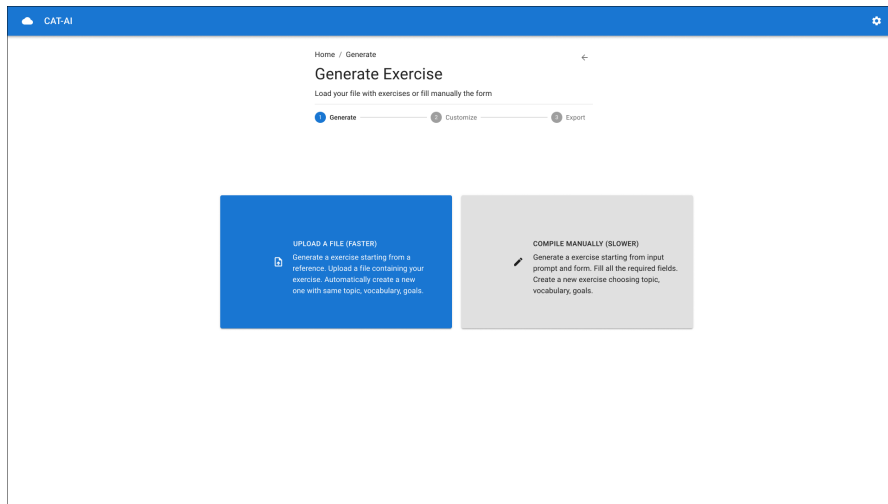


Figura 3.5: Prototipo: Scelta Modalità

La prima modalità permette l'inserimento di un file tramite un uploader (FR1.3) come mostrato in figura 3.6.

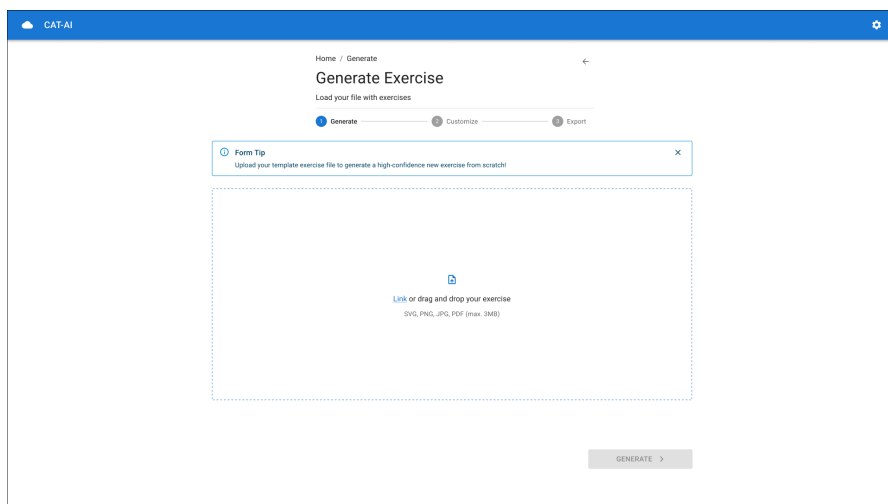


Figura 3.6: Prototipo: Uploader

Una volta caricato il file, le informazioni quali testo del file caricato (FR1.4), obiettivi di apprendimento e prerequisiti saranno estratte automaticamente (FR3.1). Essi potranno essere modificati in base all'obiettivo desiderato (Figura 3.7). Nel

Figura 3.7: Prototipo: Form Modalità 1

caso di scelta della seconda modalità, l'utente dovrà inserire manualmente tutti i requisiti (FR1.1) e prerequisiti (FR1.2) richiesti (Figura 3.8).

Figura 3.8: Prototipo: Form Modalità 2

3.4.3 Personalizzazione

Proseguendo nel processo, una volta generato il contenuto dell'esercizio con soluzione dall'IA (FR3.2) è possibile visualizzare l'interfaccia di personalizzazione dell'esercizio con editor WYSIWYG (FR4) (Figura3.9). In questa sezione è possibile effettuare modifiche e verifiche all'esercizio creato tramite la barra degli strumenti verticale sulla destra.

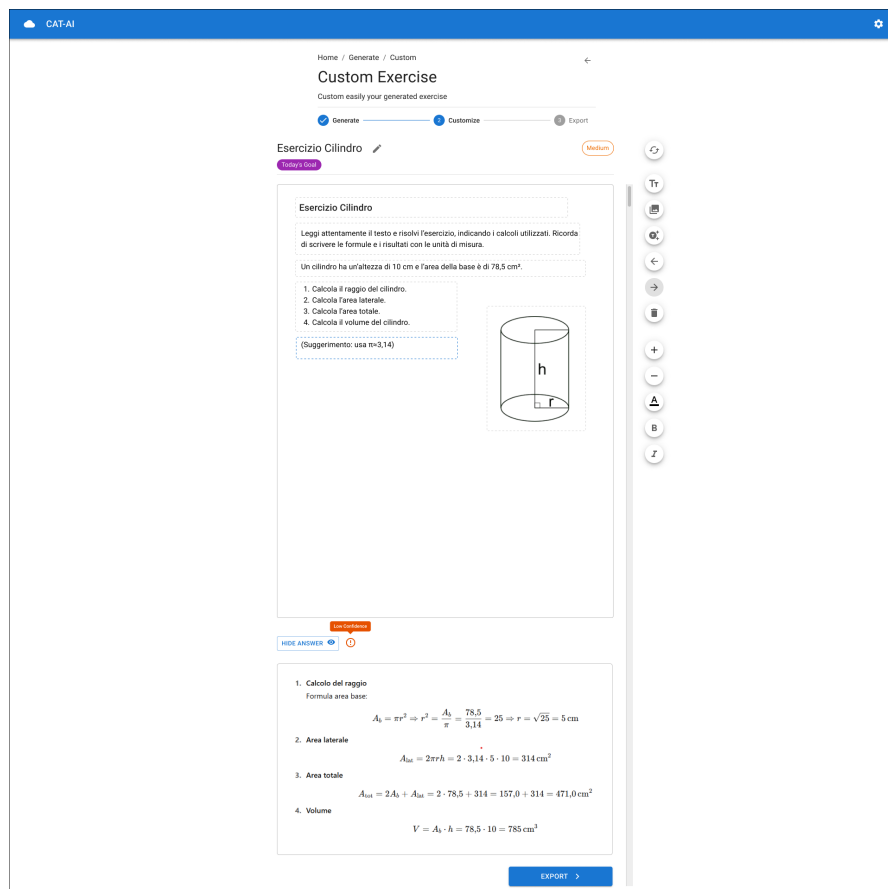


Figura 3.9: Prototipo: Personalizzazione Esercizio

Oltre alle normali funzioni di editing, nella barra degli strumenti laterali possiamo trovare 2 nuove funzionalità supportate da IA, la rigenerazione dell'esercizio e la rigenerazione di un elemento specifico all'interno dell'esercizio tramite prompt in input (FR3.3,FR3.4) (Figura 3.10).

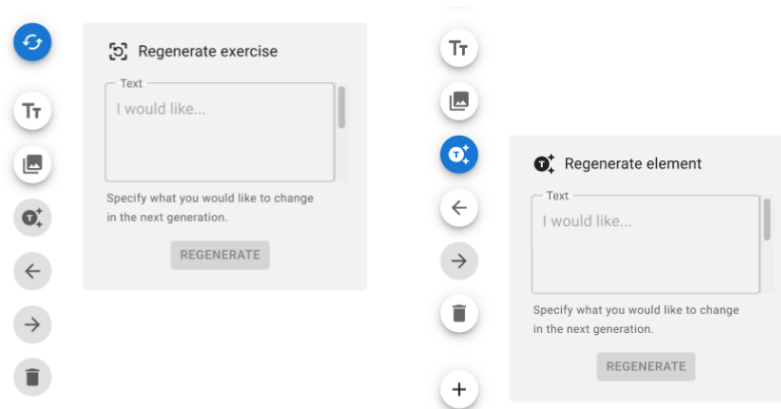


Figura 3.10: Prototipo: Rigenerazione con IA

3.4.4 Salvataggio

Una volta effettuate le modifiche desiderate, l'ultima fase del processo permette di vedere l'anteprima dell'esercizio creato e permette il salvataggio in PDF (FR5). L'utente potrà scegliere il formato dell'esercizio desiderato. Tra i formati disponibili ci sarà la possibilità di salvare l'esercizio con l'intestazione da compito in classe (nome, cognome, data) già pronta.

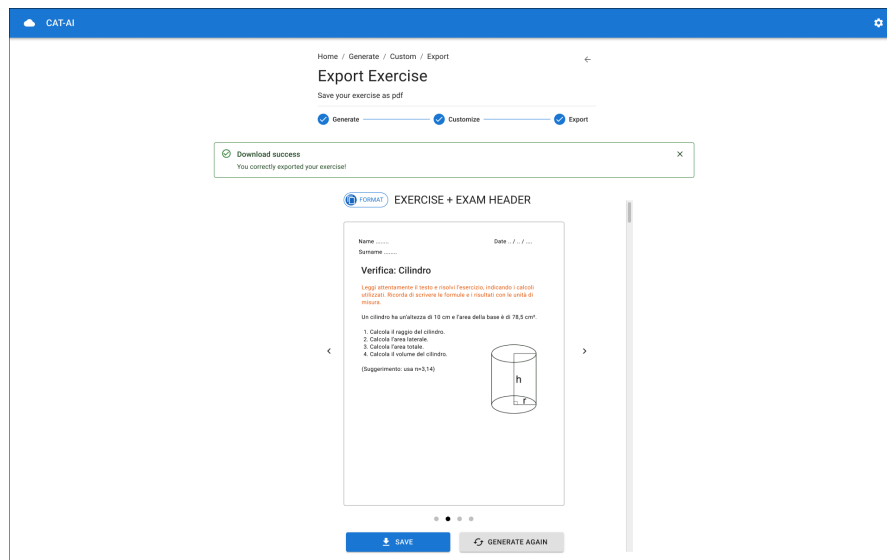


Figura 3.11: Prototipo: Salvataggio Esercizio

3.4.5 Impostazioni

L'ultimo prototipo creato è quello relativo alle impostazioni di generazione dell'esercizio (FR2). Esse sono applicate a tutti gli esercizi generati. Si dividono in impostazioni generali, impostazioni dello stile e impostazioni avanzate. La prima permette di selezionare la difficoltà generale tra facile, medio e difficile; la seconda permette di selezionare font e colori personalizzati per l'esercizio; l'ultima permette di regolare impostazioni di accessibilità e di confidenza.

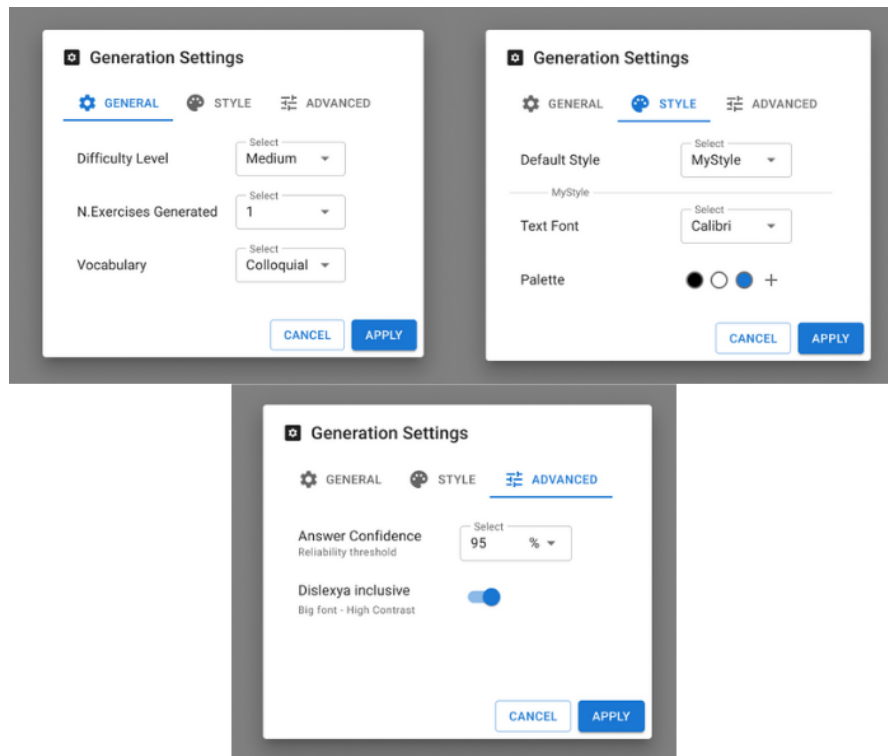


Figura 3.12: Prototipo: Impostazioni di Generazione

Capitolo 4

Implementazione

Una volta definito il prototipo a media fedeltà di CAT-AI, le caratteristiche visive e funzionali dell'applicazione sono state delineate. Perciò, la fase successiva dello sviluppo del tool consiste nell'implementazione effettiva tramite codice dell'interfaccia e delle funzionalità. In questo modo, sarà possibile passare dallo schema grafico creato con Figma a un'applicazione web utilizzabile e funzionante.

In questo capitolo viene descritto il risultato di implementazione del tool CAT-AI, con l'obiettivo di mostrare come, grazie alle tecnologie utilizzate, si sia riusciti a creare un'applicazione web finita e funzionante.

Verranno perciò in primo luogo analizzate le tecnologie utilizzate per lo sviluppo di CAT-AI, dando particolare attenzione a linguaggi di programmazione e librerie di supporto sfruttate per l'implementazione. L'analisi delle tecnologie sarà divisa parallelamente tra la parte di interfaccia utente (frontend) e la parte di logica interna del programma (backend).

Verrà poi analizzato lo sviluppo effettivo del frontend dell'applicazione, evidenziando in particolare l'architettura di navigazione tra le diverse pagine dell'applicazione, i cambiamenti dell'interfaccia utente rispetto al prototipo a media fedeltà e la gestione dei dati lato client. L'analisi di queste caratteristiche sarà fatta seguendo l'architettura di navigazione dell'interfaccia per ognuna delle pagine dell'applicazione implementate. In questo modo, sarà possibile comprendere al meglio non solo le funzionalità del lato client ma anche le motivazioni che hanno portato a determinati cambiamenti rispetto al prototipo a media fedeltà.

In seguito verrà analizzato lo sviluppo del backend, con particolare attenzione all'architettura e gestione della comunicazione client-server, alle funzioni implementate per la gestione dei file all'interno dell'applicazione e alle funzioni implementate per la generazione di contenuti tramite OpenAI API. Verranno anche analizzate le strategie di prompting adottate per l'ottimizzazione dell'output dell'LLM. Verrà delineata perciò una visione completa sull'implementazione di tutti i requisiti funzionali e non funzionali.

4.1 Tecnologie

Prima di mostrare le fasi di sviluppo del codice e i risultati implementativi, questa sezione riassumerà le tecnologie utilizzate per l'implementazione. Come anticipato in fase di progettazione, l'applicazione sviluppata è basata su un'architettura client-server. Per cui viene separata la gestione dell'interfaccia utente (Frontend) dalla logica applicativa e dalle funzionalità esterne implementate (Backend).

La comunicazione tra client e server avviene tramite REST API (Representational State Transfer Application Programming Interface). Quest'ultima, è un'interfaccia di programmazione che consente a diversi sistemi di comunicare tramite internet. Nello specifico, i sistemi comunicano utilizzando i metodi standard HTTP, scambiandosi dati in formato JSON.

Per quanto riguarda invece la scelta del linguaggio di programmazione, per garantire una buona flessibilità, modularità e rapidità nello sviluppo è stato utilizzato un approccio full-stack Javascript. Questa scelta ha permesso di utilizzare Javascript come linguaggio di programmazione sia per il lato client che per il lato server.

4.1.1 Frontend

Il frontend rappresenta la parte dell'applicazione che è direttamente visibile dall'utente. Essa comprenderà le caratteristiche visive che sono state mostrate durante la progettazione dell'interfaccia utente tramite Figma nel capitolo precedente. Di seguito possiamo trovare elencate le principali tecnologie utilizzate per il lato client, tra cui il linguaggio di programmazione, librerie e framework.

Javascript : Come anticipato nell'introduzione, è stato utilizzato *Javascript* come linguaggio di programmazione principale. Quest'ultimo è il linguaggio nativo dei browser e fornisce un ambiente di sviluppo vasto e versatile, grazie alla grande disponibilità di librerie e framework forniti e supportati.

ReactJS : *React* è una libreria Javascript open-source creata da Meta per lo sviluppo di interfacce utente. Essa permette di dividere il codice e i vari elementi dell'applicazione in Componenti riutilizzabili. Questo permette di avere un'organizzazione chiara e organizzata del codice, oltre che modulare e semplificare l'implementazione.

Material UI : *MUI* è una libreria che offre componenti React per l'interfaccia predefiniti (pulsanti, schede, menu, etc.). I componenti forniti seguono le linee guida del Material Design di Google, permettendo a sviluppatori di creare interfacce coerenti e ad alta qualità facilmente e in modo efficace.

Tailwind : Per un maggiore controllo dello stile dell'interfaccia e del layout è stato utilizzato *Tailwind*, quest'ultimo è un framework CSS che permette di personalizzare componenti e struttura dell'interfaccia fornendo delle classi CSS predefinite, e utilizzabili direttamente sul componente HTML o React. Le classi fornite sono anche facilmente modificabili e customizzabili per una gestione facile e immediata dello stile (colori, effetti grafici, posizionamento componenti, etc.)

React Router : Per la gestione della navigazione tra le diverse pagine all'interno dell'applicazione è stato utilizzato *React-Router*. Questa, è una libreria esterna a React che permette una navigazione dinamica tra i diversi componenti tramite URL, senza la necessità di ricaricare l'intera pagina.

Altro : Tra le altre librerie utilizzate possiamo trovare *React-Draggable*, per permettere all'editor manuale di trascinare e rilasciare i componenti dell'esercizio. Le librerie *html2canvas* e *jspdf* sono state utilizzate invece per trasformare il componente che rappresenta il foglio dell'esercizio in una rappresentazione di dati convertibile in un PDF A4. Infine, la libreria *pdfjs-dist* per la visualizzazione dell'anteprima del pdf creato con l'esercizio.

4.1.2 Backend

Il backend ha invece il compito di gestire la logica applicativa e le funzionalità del tool. Esso riceve le richieste HTTP dal frontend tramite le interazioni dell'utente con l'interfaccia e si occupa di elaborarle per restituire correttamente i dati richiesti. Inoltre, nel nostro caso esso funge anche da interfaccia tra frontend e l'API di OpenAI per la generazione automatica di contenuti tramite LLM. Di seguito sono elencate le librerie e i framework principali utilizzati per lo sviluppo del lato server.

Javascript : Allo stesso modo che per il lato client, *Javascript* ha permesso di avere un ambiente di sviluppo agevole anche per il lato server. Esso ha permesso una gestione delle richieste in arrivo dal frontend chiara ed efficace.

NodeJS : *NodeJS* è un ambiente runtime Javascript open-source che permette l'esecuzione di codice JS lato server. Esso permette di creare applicazioni web scalabili e veloci con un unico linguaggio. Inoltre, grazie alla sua architettura è particolarmente adatto per applicazioni real-time e di rete.

Express : *ExpressJS* è un framework per NodeJS per la gestione delle API lato server. Esso semplifica la gestione delle richieste e risposte HTTP e permette l'uso di numerosi middleware per una implementazione più veloce, flessibile e scalabile.

OpenAI : La libreria di *OpenAI* fornisce la possibilità al lato server di poter comunicare con l'API di OpenAI che fornisce numerosi modelli di LLM (Large Language Model) avanzati per la generazione di contenuti a partire da un prompt in input.

Multer : Per una gestione efficiente dei file è stato utilizzato *Multer*, una libreria middleware per NodeJS e ExpressJS che gestisce i dati in input dal frontend e permette il caricamento e scaricamento di file.

Altro : Tra le altre librerie o framework utilizzati troviamo *cors*, che permette una gestione efficiente della comunicazione tra client e server su domini diversi; *dotenv* per la gestione di variabili d'ambiente (OpenAI Key); *PdfParse* per convertire i dati dell'esercizio in arrivo dal frontend in un formato PDF scaricabile.

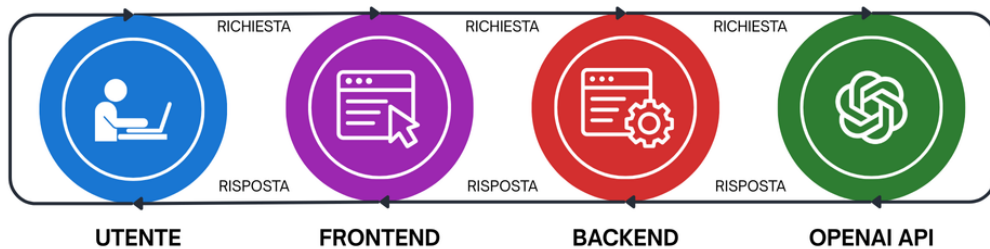


Figura 4.1: Processo Interazione Tecnologie

4.2 Implementazione Frontend

Come anticipato, il frontend è stato sviluppato utilizzando ReactJS per la gestione dei componenti e della logica di visualizzazione di essi, insieme a Tailwind CSS per la definizione dello stile e della visualizzazione dell'interfaccia.

L'obiettivo principale è stato quello di realizzare l'interfaccia a partire dal prototipo creato su Figma, ma dando maggiore attenzione alla sua reale usabilità e intuitività durante la sua implementazione, migliorando di conseguenza gli elementi meno coerenti o che limitavano l'esperienza utente.

Perciò, in questa sezione vengono descritte le diverse parti che compongono il frontend: la pagina iniziale, il modulo che permette la generazione del contenuto, l'editor grafico per la personalizzazione dell'esercizio, l'esportazione e il salvataggio dell'esercizio terminato e infine la pagina di impostazioni. Per ciascuna area verranno illustrate sia le scelte di design più rilevanti che la logica implementativa che connette il frontend alle funzionalità del backend. In Figura 4.2 possiamo vedere uno schema dell'architettura delle route implementate tramite ReactRouter.

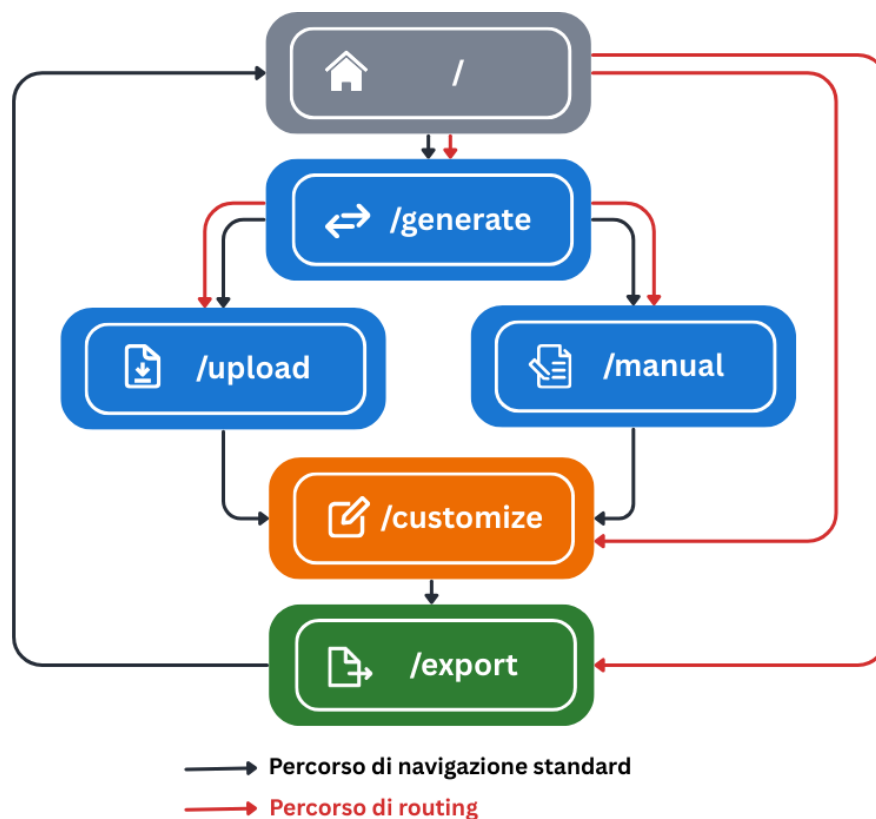


Figura 4.2: Architettura Route Frontend

4.2.1 Pagina Iniziale

La pagina iniziale è alla radice dell'architettura di routing (percorso “/”), ed essa è rimasta invariata rispetto al prototipo Figma per quanto riguarda le sue funzioni. Infatti, essa ha l'unico compito di introdurre l'utente alle funzionalità e possibilità offerte dal tool, e di conseguenza iniziare la navigazione.

A tale fine, sono state aggiunte, insieme alla descrizione scritta, delle illustrazioni visive delle fasi da percorrere all'interno dell'applicazione per creare un esercizio. Tali illustrazioni sono state generate tramite ChatGPT mantenendo un carattere semplice, colorato e giocoso, in linea con gli utenti a cui l'applicazione sarà indirizzata (Figura 4.3). Tramite il bottone “Start Generating” (Call-to-action) l'utente potrà iniziare la navigazione e perciò il processo di creazione.

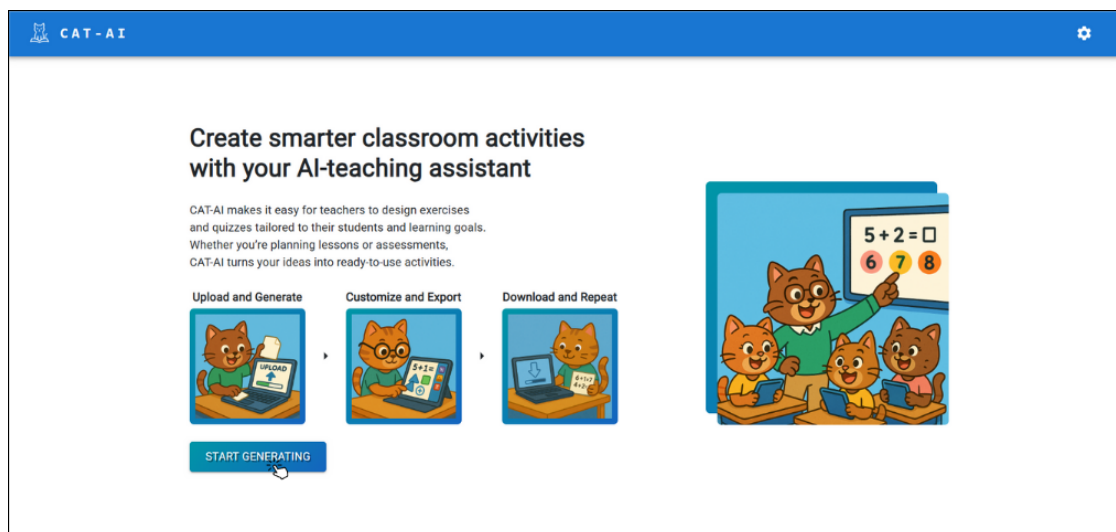


Figura 4.3: Pagina Iniziale

4.2.2 Inserimento Dati e Generazione

Scelta Modalità La prima fase del processo di generazione inizia con la scelta della modalità desiderata. Per una maggiore coerenza nella terminologia, il nome di questa prima fase è stato cambiato in “Set Up” (Imposta), rispetto al nome “Generate” (Genera) che era stato scelto in fase di progettazione. In questo modo è più chiaro quello che l’utente andrà a fare in questa prima fase.

Infatti, come da progettazione, utilizzando una prospettiva lato utente, quest’ultimo sarà portato a scegliere e inserire le informazioni desiderate per la generazione. Rispetto al prototipo, il cambiamento principale è stato il cambio delle semplici icone nei bottoni di scelta con delle illustrazioni, più chiare e d’impatto. In questo modo, anche solo guardando l’illustrazione, un utente può capire la differenza tra le due fasi. Anche in questo caso, le illustrazioni sono state generate con ChatGPT, mantenendo colori, stile e tono.

La funzione dell’interfaccia è rimasta invece invariata rispetto al prototipo: permettere all’utente di scegliere se generare l’esercizio a partire da un file di riferimento oppure procedere con un inserimento totalmente manuale.

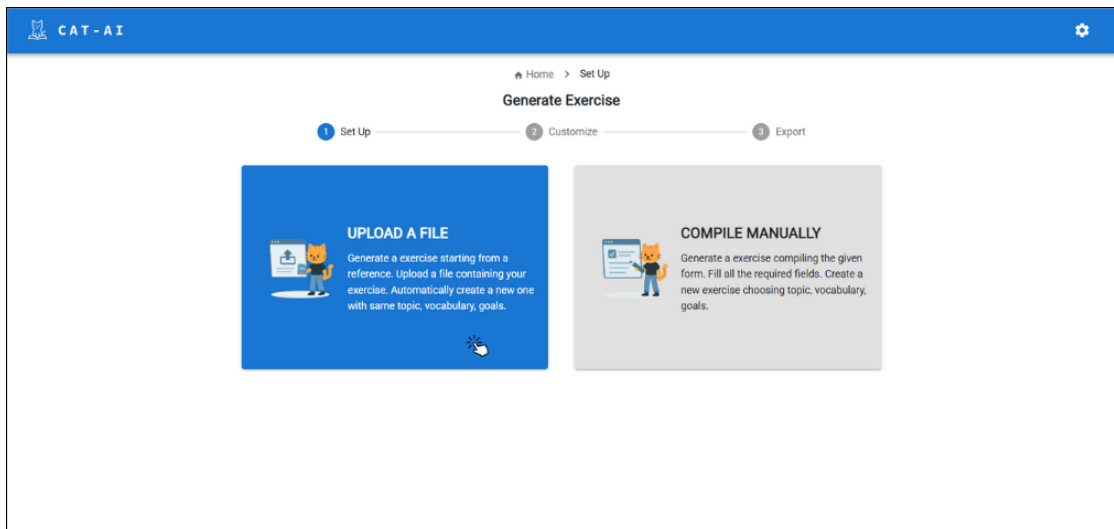


Figura 4.4: Scelta Modalità

Inserimento File L'interfaccia di inserimento file è rimasta essenzialmente invariata. Quest'ultima permette di caricare un file di un esercizio che può essere sia un PDF che un'immagine (es. screenshot o foto) di un esercizio di riferimento.

Il file può essere caricato sia tramite click che tramite drag&drop nell'area evidenziata. L'interfaccia ha inoltre il compito di restituire all'utente errore e specificarne il motivo nei seguenti casi:

- File inserito troppo grande in termini di spazio occupato.
- File inserito non compatibile con i formati supportati.
- Contenuto del file non riconducibile a un esercizio scolastico o non leggibile.

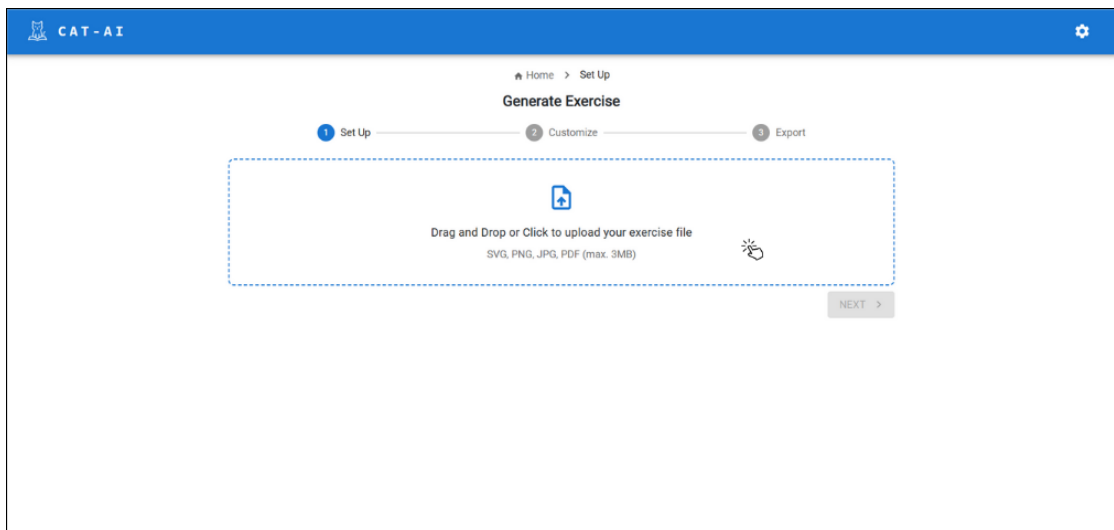


Figura 4.5: Inserimento File

Cosa Tenere Una volta inserito il file, l'utente potrà visualizzare una conferma dell'inserimento corretto del file. Inizia perciò la fase di inserimento dei dati desiderati da parte dell'utente. Oppure, in caso l'utente abbia sbagliato nell'inserimento del file, quest'ultimo può essere eliminato con l'apposito bottone con l'icona del cestino, e ripetere l'operazione di inserimento.

L'interfaccia di inserimento dati, per la modalità 1 (inserimento file) è stata modificata in fase di sviluppo e divisa in due parti. La prima parte è quella dove l'utente può visualizzare l'esercizio di riferimento inserito e decidere quali caratteristiche tenere in quello che verrà generato tra: tipologia di esercizi, struttura o vocabolario utilizzato.

In questo modo il processo di inserimento dati diventa molto più chiaro e guidato rispetto al prototipo iniziale. Il lato client dovrà avvisare l'utente con un errore e impedirgli di proseguire alla seconda parte nei casi in cui:

- Il campo del testo è vuoto.
- Non è stata selezionata nessuna caratteristica da tenere.

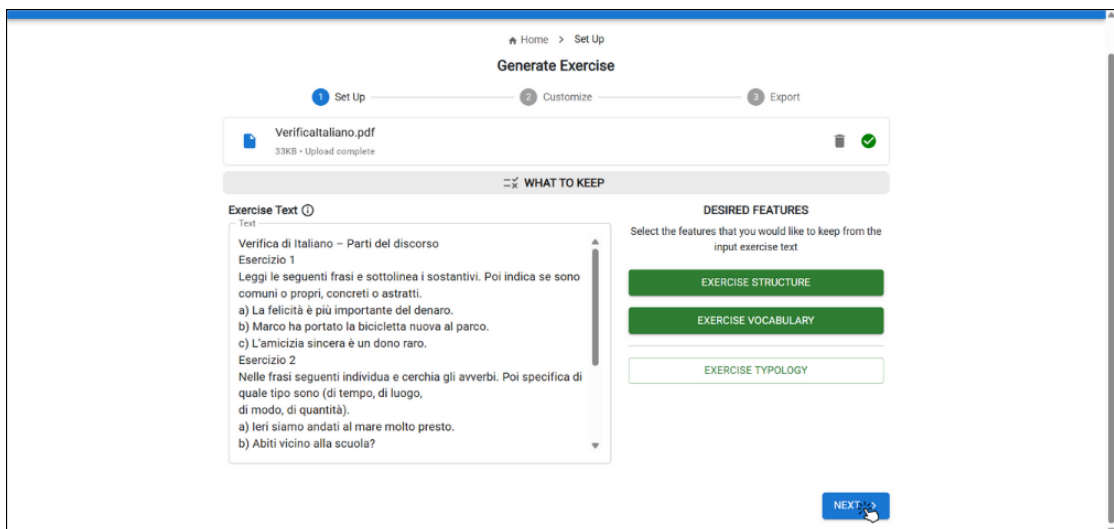


Figura 4.6: File: Cosa Cambiare

Cosa Cambiare La seconda parte dell’inserimento dati permette invece di scegliere cosa cambiare delle caratteristiche del file inserito. Tutti i campi obbligatori in questa modalità saranno automaticamente estratti dal riferimento e poi cambiabili manualmente dall’utente. Anche questa interfaccia è stata cambiata in fase di sviluppo.

Nel particolare, oltre a un infobox con una breve descrizione di cosa fare, è stata aggiunta la possibilità di specificare anche scuola e classe insieme alle altre caratteristiche già inserite nel prototipo. In questo modo sarà ancora più facile per l’utente direzionare la generazione verso un risultato desiderato. Cliccando il tasto “Next” si potrà passare alla seconda parte di inserimento dati. Il lato client avrà anche la funzione di validare i parametri e avvisare l’utente, non permettendogli di generare il contenuto, nel caso in cui uno dei seguenti campi obbligatori sia vuoto:

- Scuola
- Classe
- Obiettivi di apprendimento
- Prerequisiti

The screenshot shows a web interface for generating an exercise. At the top, there's a breadcrumb 'Home > Set Up' and a title 'Generate Exercise'. Below this is a progress bar with three steps: '1 Set Up' (active), '2 Customize', and '3 Export'. The main content area is titled 'WHAT TO CHANGE' and contains several sections: 'School' with a dropdown set to 'elementary', 'Grade' with a dropdown set to '3', 'Learning-Goals' with a text input containing 'passive verbs', 'active verbs', and 'switch from active to passive form', 'Pre-Requisites' with a text input containing 'basic grammar', 'identify verbs', and 'pre-requisite', and 'Features' with two checked checkboxes: 'Reminder' and 'Worked Example'. At the bottom right, there are two buttons: 'BACK' and 'GENERATE'.

Figura 4.7: File: Cosa Tenere

Inserimento Manuale Nel caso di scelta della seconda modalità (Inserimento Manuale), l'utente potrà visualizzare una schermata del tutto analoga alla schermata in figura 4.7 (File:Cosa Tenere). La differenza principale sta nel fatto che in questo caso, non essendoci un file di riferimento, le caratteristiche dovranno essere tutte inserite manualmente. Non essendoci un riferimento, anche il controllo su struttura, tipologia o vocabolario dell'esercizio viene meno.

Nonostante questo, si può fare utilizzo delle impostazioni di generazione per regolare le caratteristiche di creazione dell'esercizio in modo più mirato anche tramite questa modalità. Come per la prima modalità, il lato client validerà e avviserà l'utente, non permettendogli di generare l'esercizio, se uno dei campi obbligatori è vuoto.

The screenshot shows the 'CAT-AI' interface for the 'Generate Exercise' process. The top navigation bar is blue with the 'CAT-AI' logo and a settings icon. Below the bar, the breadcrumb 'Home > Set Up' is visible. The main heading is 'Generate Exercise', followed by a progress indicator with three steps: '1 Set Up' (active), '2 Customize', and '3 Export'. The 'Set Up' step contains a form with the instruction: 'Fill the form with your desired features to create a brand new exercise. The generated exercise TOPICS and COMPLEXITY will depend on the following features'. The form includes several fields: 'School' (a dropdown menu with 'middle' selected), 'Grade' (a dropdown menu with 'grade' selected and a list of options 1, 2, 3 shown), 'Learning-Goals' (a text input field with 'Insert' placeholder), 'Pre-Requisites' (a text input field with 'Insert' placeholder), and 'Features' (checkboxes for 'Reminder' and 'Worked Example'). A blue 'GENERATE >' button is located at the bottom right of the form.

Figura 4.8: Inserimento Manuale

4.2.3 Personalizzazione Esercizio

Una volta generato il contenuto con le informazioni inserite, si passa alla seconda fase di verifica e personalizzazione dell'esercizio generato. L'esercizio e la sua soluzione appariranno su due pagine distinte e già formattate all'interno del foglio. Le parti dell'esercizio e della soluzione saranno divise in una serie di caselle di testo che potranno essere modificate singolarmente. Questa schermata è quella che ha subito più modifiche rispetto al prototipo sia per la struttura dell'interfaccia che per le sue funzionalità. Le modifiche hanno reso l'interfaccia più intuitiva e funzionale per un editing facile e veloce.

Nel particolare, è stata aggiunta una barra degli strumenti orizzontale sempre visibile per raggruppare tutti gli strumenti che non servono a modificare direttamente i contenuti dell'esercizio. Un elenco delle funzionalità completo delle due barre degli strumenti è mostrato nella Tabella 4.1.

Oltre alle funzionalità offerte dalle barre degli strumenti, l'editor grafico permette sia di trascinare gli elementi all'interno della pagina che di ridimensionare la grandezza e di modificare manualmente il contenuto delle caselle di testo (doppio clic sulla casella di testo). Gli elementi possono essere spostati o ridimensionati limitatamente alla pagina in cui sono inseriti. Inoltre, sono state inserite delle linee di riferimento, visibili mentre un elemento viene trascinato, per allineare gli elementi tra loro o al centro del foglio sia orizzontalmente che verticalmente.

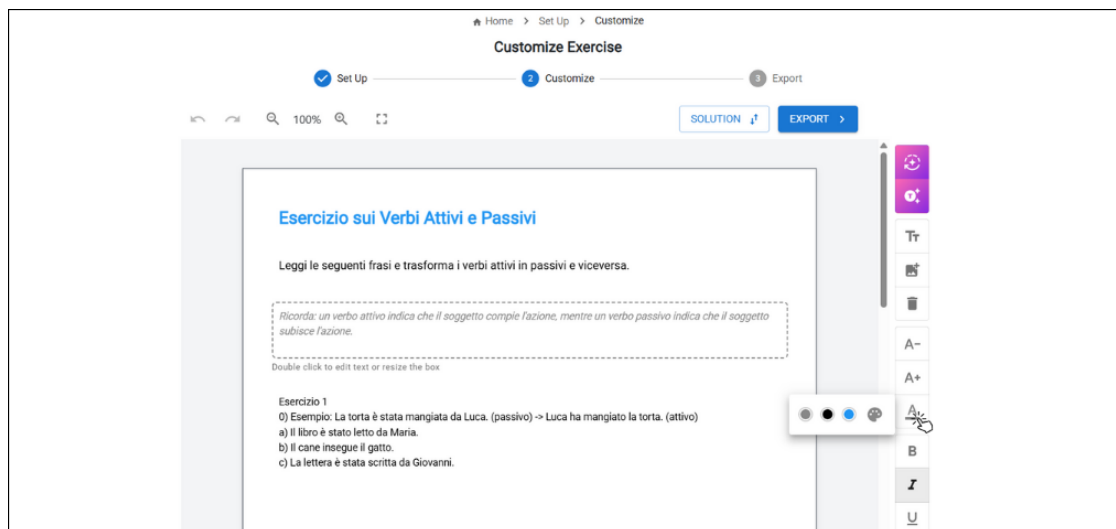


Figura 4.9: Personalizzazione Esercizio

Barra Orizzontale	
Nome	Funzionalità
Undo / Redo	Navigare avanti e indietro la storia delle azioni fatte durante la modifica.
Zoom In / Zoom Out	Ingrandire o diminuire la grandezza del foglio di lavoro.
Expand / Reduce	Passare dalla visualizzazione schermo standard a quella schermo intero e viceversa.
Exercise / Solution	Scrollare facilmente dalla pagina esercizio alla pagina soluzione e viceversa.
Export	Esportare l'esercizio creato e passare all'ultima fase di salvataggio.
Barra Verticale	
Nome	Funzionalità
Rigenerate exercise	Rigenerare tutti gli elementi di testo dell'esercizio o una parte di essi tramite prompt.
Regenerate element	Rigenerare l'elemento di testo selezionato tramite prompt.
Delete Element	Eliminare l'elemento attualmente selezionato.
Add text	Aggiungere una casella di testo vuota nella pagina selezionata.
Add image	Aggiungere un immagine nella pagina selezionata tramite upload o generazione.
Increase/Decrease Text	Ingrandire o rimpicciolire il carattere dell'elemento di testo selezionato.
Change Text Color	Cambiare il colore del contenuto della casella di testo selezionata.
Bold	Mettere in grassetto il contenuto della casella di testo selezionata.
Italic	Mettere in corsivo il contenuto della casella di testo selezionata.
Underline	Sottolineare il contenuto della casella di testo selezionata.

Tabella 4.1: Funzionalità della barra orizzontale e verticale

Inserimento Elementi In Figura 4.10 possiamo vedere l'interazione di inserimento di una nuova casella di testo. Tramite il clic dell'apposito bottone nella barra verticale sarà automaticamente aggiunta al foglio selezionato (esercizio o soluzione) una nuova casella di testo vuota, pronta per essere modificata. Il contenuto della casella sarà modificabile, come specificato sotto la casella, tramite doppio clic.



Figura 4.10: Aggiungi Casella di Testo

Nella Figura 4.11 possiamo invece vedere l'iterazione per aggiungere un'immagine. Per aggiungere un'immagine è possibile sia caricarne una dal dispositivo (Upload) che farne generare una dall'IA tramite prompt in input come mostrato in figura.

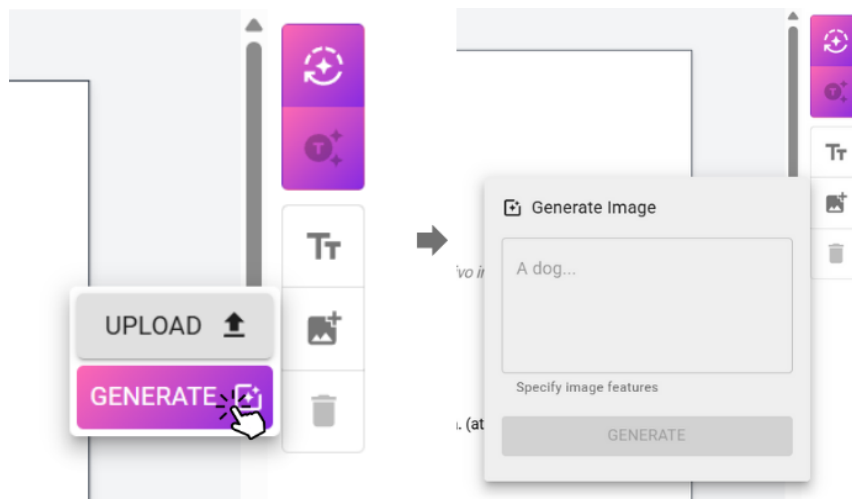


Figura 4.11: Aggiungi Immagine

Parametri Elementi La gestione degli elementi dell'esercizio e di tutte le funzioni associate è stata fatta tramite un Context React. La scelta del Context lato client è data dal fatto che i dati dell'esercizio sono validi unicamente per la generazione corrente, per cui utilizzare un database per immagazzinare tali dati avrebbe rallentato il processo di creazione e complicato la comunicazione client-server, effettuando numerose richieste al server e al database.

Al contrario, tramite il context è possibile non solo avere una condivisione semplice dei dati tra i vari componenti, ma è anche possibile salvare direttamente gli oggetti in locale. Di seguito possiamo vedere la struttura degli elementi di testo e immagini e i dati salvati per ciascuno.

Elemento Esercizio		
Nome	Tipo	Valori
ID	Numerico	Timestamp
Pagina	Numerico	1 (Esercizio) 2 (Soluzione)
Posizione	(X:Numerico,Y:Numerico)	$m < X < 794 - m$ $m < Y < 1123 - m$ dove "m" è il margine
Dimensione	(H:Numerico,W:Numerico)	$50 < H < 1123 - m - Y$, $200 < W < 794 - m - X$ dove "m" è il margine
Casella di Testo		
Contenuto	Stringa	Contenuto della casella
Dimensione Testo	Numerico	tra 12 e 24
Colore Testo	Stringa	Colore Esadecimale
Grassetto	Booleano	True/False
Corsivo	Booleano	True/False
Sottolineato	Booleano	True/False
Confidenza Contenuto	Numerico	1/7
Motivazione	String	Motivazione Confidenza
Immagine		
URL Immagine	Stringa	Percorso relativo

Tabella 4.2: Dati Elementi Esercizio

Rigenerazione con IA All'interno dell'applicazione, tutte le funzionalità che supportano IA sono ben evidenziate da un gradiente rosa/viola. Questo colore è frequente nei tool che implementano funzionalità AI e quindi familiari a chi ne fa già utilizzo. L'editor dell'esercizio comprende, oltre alla generazione di immagini con IA, 3 principali metodologie di rigenerazione guidata dei contenuti (Figura 4.12).

- **Rigenera Tutto:** Questa funzionalità permette di rigenerare tutto o parte dell'esercizio/soluzione tramite un prompt in input. Questa funzione permette di aggiungere, modificare o togliere dinamicamente uno o più elementi di testo all'interno dell'esercizio/soluzione.
- **Rigenera Elemento:** Questa funzionalità permette di rigenerare l'elemento di testo attualmente selezionato, specificando tramite prompt cosa deve essere cambiato. Può essere utile nel caso si voglia fare una modifica mirata e essere sicuri che non venga modificato altro se non l'elemento selezionato.
- **Rigenera Soluzione:** Questa funzionalità non richiede un prompt in input, ma un semplice clic. L'IA a questo punto ricalcolerà la soluzione sulla base dell'esercizio attualmente presente. Può essere utile nel caso in cui siano stati modificati uno o più elementi dell'esercizio e si voglia ricalcolare automaticamente la nuova soluzione.

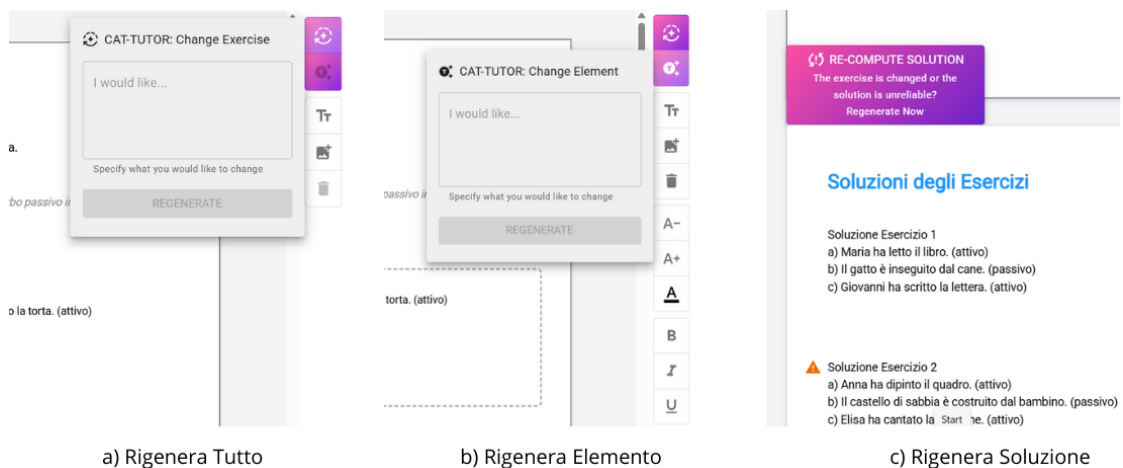


Figura 4.12: Rigenerazione con IA

Risoluzione Problemi Durante la generazione dell'esercizio, l'IA potrebbe generare delle incoerenze o errori logici. Infatti, i modelli linguistici avanzati (LLM), essendo basati su una combinazione probabilistica e matematica di parole e non su un vero ragionamento, sono spesso inclini a errori.

Per questo motivo, in fase di generazione l'IA ha oltre il compito di generazione, anche quello di successiva verifica del contenuto generato per far emergere possibili incoerenze ed errori.

In questo modo, nel foglio di lavoro potrebbero essere presenti degli avvisi, legati alle caselle di testo, per richiedere una revisione del blocco in questione all'utente. Questa funzionalità ha l'obiettivo di aumentare la fiducia dell'utente nell'output dato dall'intelligenza artificiale, mostrando che essa si rende conto di eventuali errori. L'utente potrà perciò verificare la coerenza logica o matematica del blocco di testo interessato e, se necessario, risolvere il problema direttamente tramite IA con prompt in input con il bottone "Fix" (Figura 4.13).

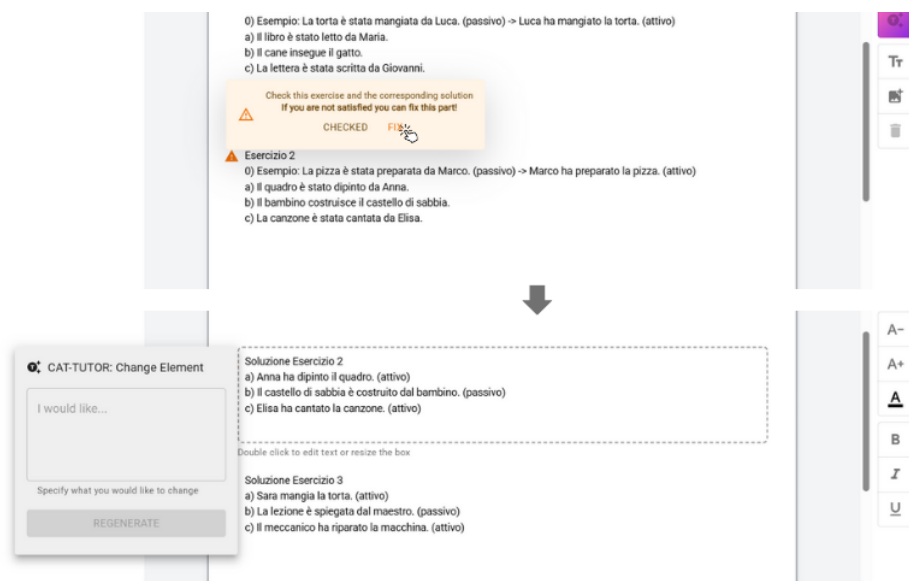


Figura 4.13: Risoluzione Problemi

4.2.4 Esportazione Esercizio

Una volta terminata la verifica ed eventuale modifica dell'esercizio, tramite il pulsante "Export" si passa all'ultima fase. L'utente in questa pagina può visualizzare un'anteprima di ciò che ha creato e può scegliere il formato desiderato da scaricare tra solo esercizio, esercizio con intestazione da verifica e solo soluzione.

Per la scelta del formato, lo Slider del prototipo è stato cambiato con un più semplice Combo Box per permettere all'utente di scegliere e vedere più semplicemente i formati disponibili.

Il processo si conclude con il salvataggio di uno o più formati desiderati, come mostrato in figura 4.13. I dati di esportazione, quali URL dei file da scaricare e dati generali di logging dell'intero processo di creazione, sono salvati localmente lato client tramite un Context React, mentre i file sono tenuti nel lato server.

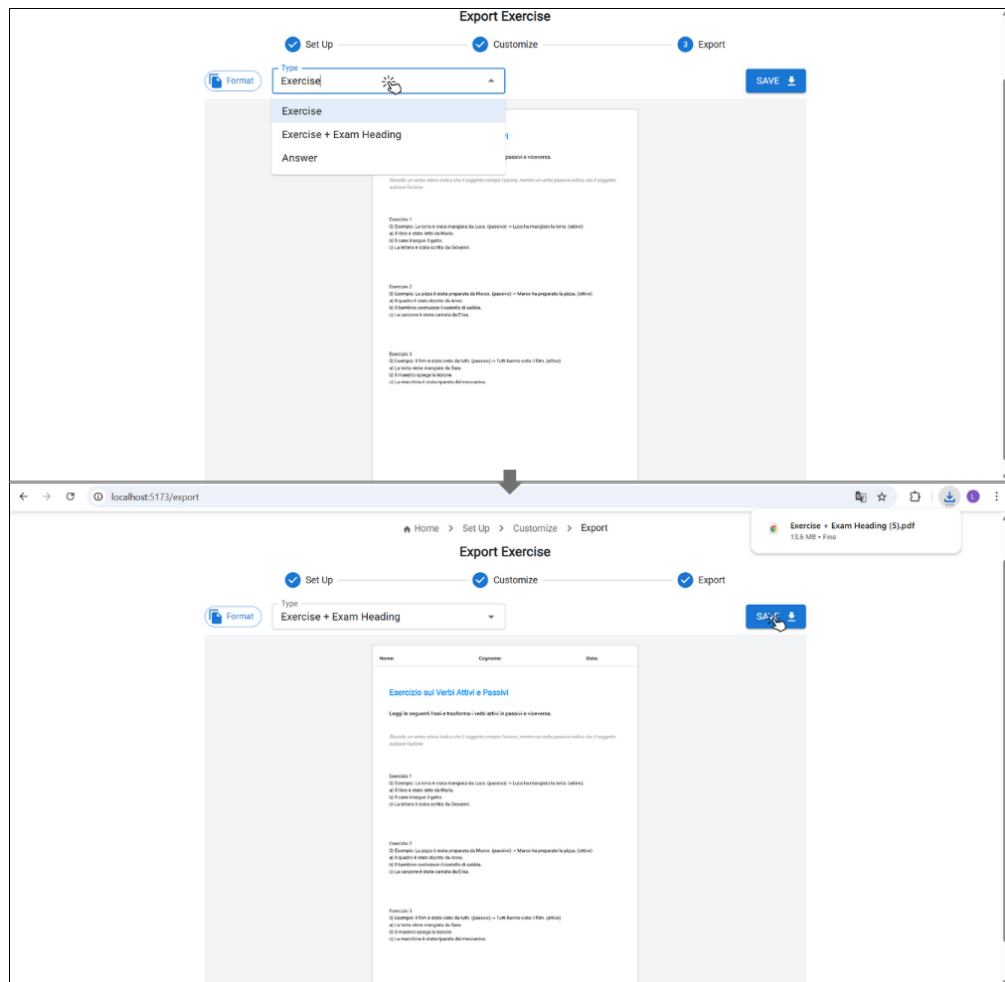


Figura 4.14: Scelta Formato

4.2.5 Barra di Caricamento

Tra una fase e l'altra all'interno del processo di creazione dell'esercizio, le attese di estrazione dei dati, generazione dell'esercizio o esportazione possono essere frustranti per un utente. Infatti, particolarmente durante la generazione dei contenuti, in cui l'IA ha il doppio compito di generazione e verifica, l'attesa può superare i 30 secondi.

Questo può essere un problema per l'esperienza utente e l'usabilità del tool. Per questo motivo, per rendere l'attesa meno frustrante lato utente è stata inserita un'animazione di caricamento in linea con lo stile giocoso dell'applicazione, e assieme ad essa un breve messaggio di avviso.

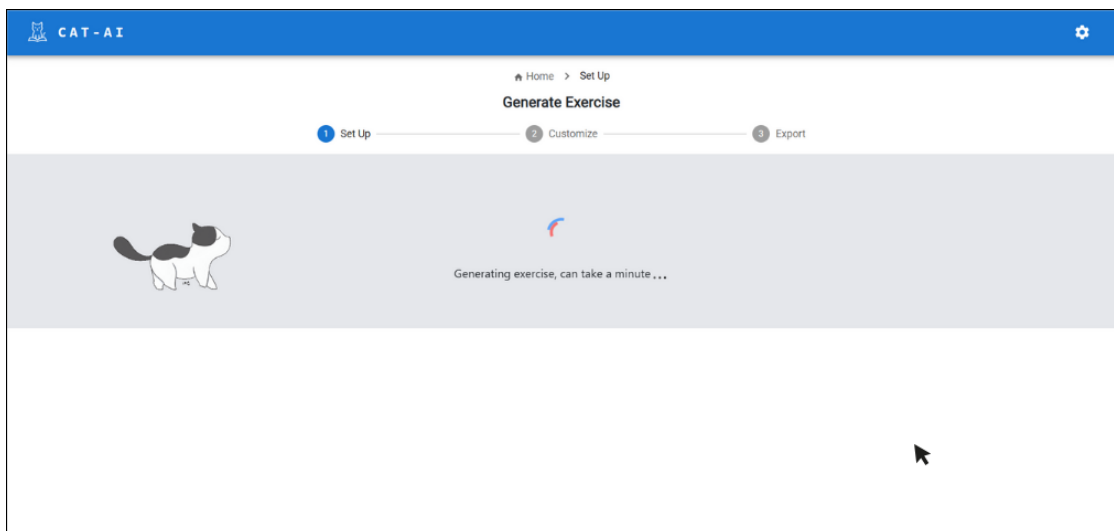


Figura 4.15: Esportazione

4.2.6 Impostazioni

Le impostazioni di generazione hanno ricevuto solo qualche piccolo cambiamento in fase di implementazione. In particolare, non è più presente l'impostazione sulla confidenza di generazione accettata dall'utente, poiché poco intuitiva nel suo funzionamento. Al suo posto è possibile scegliere la lingua dell'esercizio generato (Italiano o Inglese).

Le impostazioni sono sempre visualizzate sotto forma di modale e sono visualizzabili in tutte le pagine dell'applicazione. Una volta modificate le impostazioni di generazione, esse saranno valide a partire dalla generazione dell'esercizio successivo. Le impostazioni sono divise in 3 sezioni come segue.

I dati di generazione, sia quelli delle impostazioni che quelli inseriti dall'utente in fase di generazione, sono raggruppati e salvati localmente lato client tramite un Context React. Un riassunto dei dati di generazione utilizzati è visibile alla Tabella 4.3.

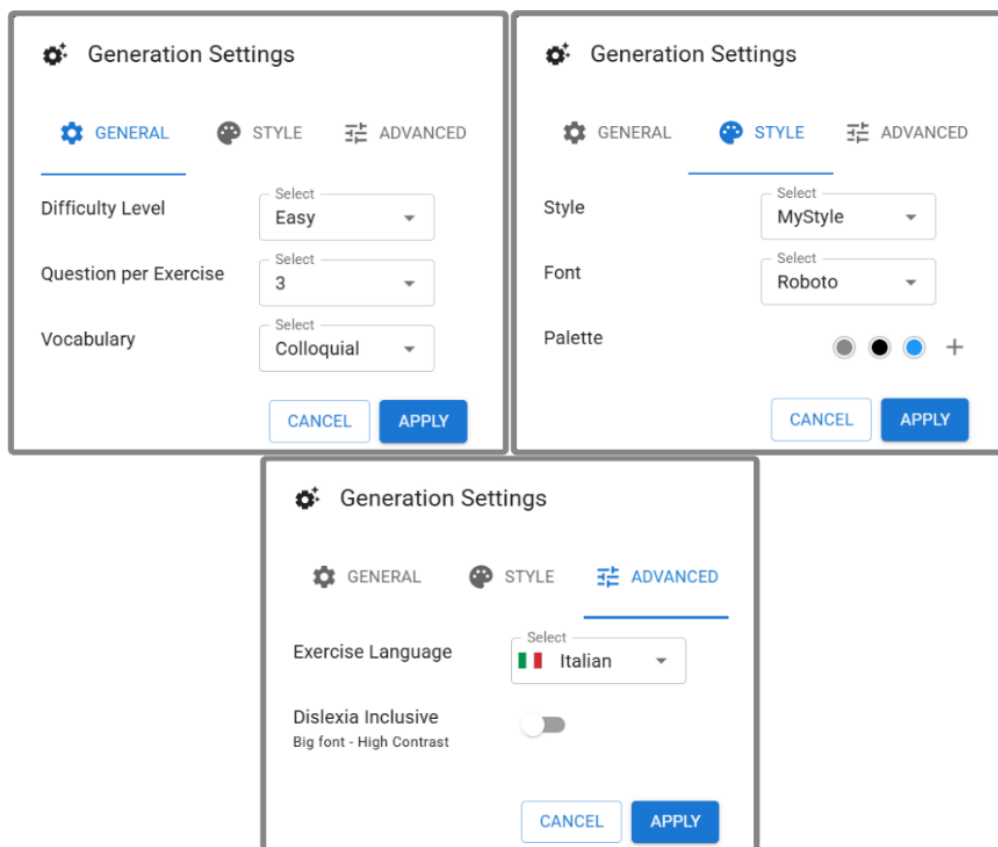


Figura 4.16: Impostazioni

Dati di Inserimento		
Nome	Tipo	Valori
Scuola	Stringa	Elementare, Media
Classe	Numero	If (Scuola == Media) 1, 2, 3 Else 1, 2, 3, 4, 5
Obiettivi di Apprendimento	Array[Stringa]	N obiettivi a scelta
Prerequisiti	Array[Stringa]	N prerequisiti a scelta
Promemoria per Studente	Booleano	True/False
Esercizio esempio	Booleano	True/False
Testo Riferimento (Opzionale)	Stringa	Contenuto esercizio di riferimento
Caratteristiche (Opzionale)	Array[Stringa]	Tipologia, Struttura, Vocabolario
Impostazioni Generali		
Nome	Tipo	Valori
Livello di Difficoltà	Stringa	Facile, Medio, Difficile
N. Domande per Esercizio	Numero	1, 2, 3, 4, 5
Vocabolario	Stringa	Colloquiale, Formale, Giocoso
Impostazioni Stile		
Nome	Tipo	Valori
Stile	Stringa	MioStile, Formale, Giocoso
Per ogni stile		
Font	Stringa	Inter, Roboto, Open Sans, Lato, Montserrat, Poppins, Raleway, Merriweather, Playfair Display, Fira Code
Colori	Array[Stringa]	Fino a 5 colori a scelta
Impostazioni Avanzate		
Nome	Tipo	Valori
Lingua	Stringa	Italiano, Inglese
Dislexia Inclusive	Booleano	True/False

Tabella 4.3: Dati e Impostazioni di Generazione

4.3 Implementazione Backend

Il backend rappresenta il fulcro della logica dell'applicazione e svolge il ruolo di intermediario tra frontend e l'API di OpenAI. Come anticipato nel capitolo precedente, è stato sviluppato con NodeJS e ExpressJS adottando un'architettura modulare e facilmente scalabile.

Il lato server si occupa della gestione delle richieste che arrivano dal lato client, elaborando e inviando le risposte. Il client, per inviare le richieste, effettua delle richieste HTTP tramite le fetch API. Le fetch API sono delle interfacce di comunicazione che fanno uso della funzione `fetch()`.

Questa funzione permette, specificando il percorso (route) di destinazione, di richiedere risorse e dati al server per poi restituirne la risposta in modo asincrono. Nella Figura 4.17 possiamo osservare l'architettura delle route gestite dal backend, ciascuna di esse implementa una funzionalità diversa, ed ognuna è chiamata specificatamente in una delle 3 fasi del processo di creazione. Esse sono divise in 2 macroaree: route di generazione e route di gestione file.

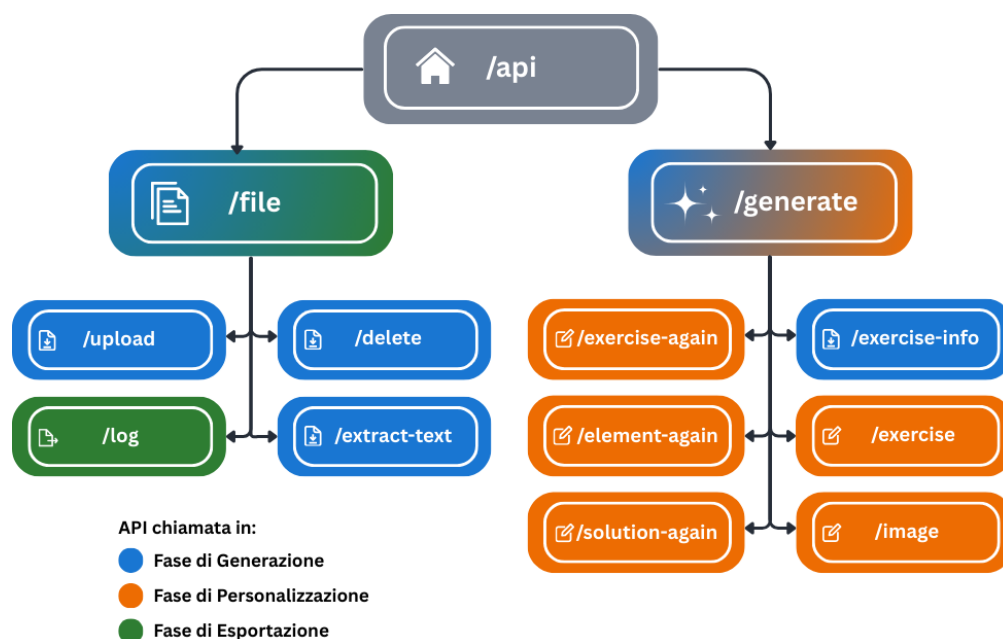


Figura 4.17: Architettura Route Backend

4.3.1 Gestione File

Per la gestione e la conservazione dei file è stato scelto di implementare la logica lato server. Infatti, il backend ha permesso di avere una validazione efficace dei parametri (formato, dimensione, tipo, etc.) oltre che un'organizzazione centralizzata dei file. Infatti, i file sono stati immagazzinati tutti in una cartella “/uploads” all'interno del backend. Per consentire al frontend di accedere ai file caricati e alle immagini generate dall'applicazione, è stato configurato un middleware di Express che espone una cartella locale come cartella di file statici.

In questo modo, i contenuti salvati sul server risultano raggiungibili tramite un endpoint dedicato. Tale configurazione permette di gestire in maniera semplice e sicura l'accesso ai file, dando la possibilità al lato client di accedere alle risorse del backend tramite `fetch()` al percorso della risorsa desiderata. Il middleware per esporre la cartella /uploads è stato integrato nel lato server come segue.

```
1 // server
2 import express from 'express';
3 import path from 'path';
4 import routes from './src/routes.mjs'
5
6 const app = express();
7 const port = 3001;
8 ...
9 app.use('/uploads', express.static(path.join(process.cwd(), '
    uploads'))); //cartella esposta
10
11 app.use("/api", routes);
12
13 app.listen(port, () => {
14   console.log('Server listening at http://localhost:${port}');
15 });
```

Di seguito possiamo vedere un esempio di una possibile chiamata lato client.

```
1 // Esempio di fetch dal client
2 const response = fetch('http://localhost:3001/uploads/Image.png',{
3   method: "GET",
4 })
5
6 if (!response.ok) {
7   throw new Error("Risorsa non trovata");
8 }
```

Tutti i percorsi delle routes di gestione file iniziano con il prefisso `/api/file` e sono di seguito elencate le loro funzionalità.

POST /upload : La funzione di upload (caricamento dei file) permette, prendendo in input il buffer del file caricato dal client, di depositarlo nella cartella esposta `/uploads`. Essa effettua la validazione del file in input verificando la sua dimensione (massimo 3 MB) e il tipo di file (supporta solo png, jpg, svg e pdf). In caso il file non rispetti uno dei parametri citati verrà ritornato un errore al lato client. Se l'upload avviene con successo ritorna l'URL di accesso al file al client, in modo che possa accederci liberamente. Questa route è chiamata in 3 occasioni diverse dal client.

La prima è nella fase di inserimento dei dati. Infatti se si sceglie la modalità di generazione a partire da un riferimento verrà chiesto di caricare un file (Figura 4.5).

La seconda occasione in cui viene chiamata è durante la fase di personalizzazione, nel momento in cui un utente voglia caricare un'immagine all'interno del foglio di lavoro (Figura 4.11).

L'ultimo caso in cui viene chiamata è al momento del clic del bottone "Export" per passare alla fase di esportazione e salvataggio. Infatti il componente React contenente l'esercizio/soluzione creato, una volta convertito in un formato dati compatibile, sarà inviato al backend per essere tenuto fino al momento del salvataggio in locale dell'utente.

POST /extract-text : Questa route permette di estrarre il testo dal file di riferimento nella fase di inserimento dei dati (Modalità "Upload a File"). Essa prende in input il percorso del file da cui estrarre il testo e ne effettua la validazione per verificare la sua reale presenza nella cartella `/uploads`. Se il file non è stato caricato la route ritornerà un errore al client. Se il file esiste, la route implementa due diverse modalità di estrazione del testo a seconda del formato di file.

In caso di PDF, tramite la libreria *pdf-parse* il buffer di informazioni del file viene convertito in un oggetto accessibile nei suoi attributi e tra questi anche il testo. Questa estrazione del testo è quasi sempre molto accurata e priva di errori di lettura.

In caso di immagine (es. foto o screenshot di un esercizio) viene utilizzata la libreria *tesseract.js*, che tramite l'utilizzo di OCR (Riconoscimento Ottico di Caratteri) riesce a riconoscere e estrarre il testo dall'immagine. In questo caso l'accuratezza dell'estrazione dipende dalla qualità dell'immagine.

In caso di successo ritorna il testo del file al client che ne potrà effettuare il display per la verifica da parte dell'utente (Figura 4.6).

POST /log : Questa route permette il salvataggio delle informazioni di creazione dell'esercizio una volta terminato il processo. Infatti durante il processo di creazione il lato client si occupa di raccogliere non solo i dati di generazione inseriti dall'utente ma anche i tempi assoluti e relativi per ogni fase di creazione. Questi dati vengono inviati a questa route, la quale si occupa di formattarli in un'unica stringa con una funzione apposita, e in seguito salvarli in un file .txt nella cartella esposta del server. Questa funzionalità viene chiamata al momento del salvataggio finale dell'esercizio (Fine del processo di creazione). Questa route è stata implementata per la fase di testing e analisi dei dati del capitolo successivo.

DELETE /delete : Questa funzione permette, a partire dal percorso di un file in input, di eliminarlo dalla cartella esposta. La route effettua la validazione del path e se non esiste o l'eliminazione fallisce ritorna errore al client. In caso di successo il file verrà eliminato dalla cartella. Questa route è chiamata durante la fase di inserimento dati, se l'utente decide di eliminare il file di riferimento caricato (Figura 4.6).

Possiamo notare dall'elenco delle route, che non è presente alcuna route dedicata al download delle risorse. Il motivo è direttamente collegato al modo in cui è stato deciso di depositare i file e immagini caricate. Infatti, essendo depositate in una cartella accessibile direttamente dal client, quest'ultimo ha la possibilità di effettuare un'operazione GET direttamente al percorso della risorsa (come mostrato nel codice esempio). Di conseguenza, potendo accedere alla risorsa il download effettivo in locale è gestito dal lato client.

Routes di Gestione File - /api/file			
Percorso	Input	Output	Funzione
POST /uploads	Buffer dati file	Percorso file	Caricare un file nella cartella /uploads
POST /extract-text	Percorso file	Testo file	Estrarre testo dal file indicato in input
POST /log	Dati di generazione	-	Salvare file con informazioni e tempi di creazione
DELETE /delete	Percorso file	-	Cancellare file indicato in input

Tabella 4.4: Routes di Gestione File

4.3.2 Generazione Contenuti

Per la generazione dei contenuti all'interno dell'applicazione si è fatto completo affidamento all'API di OpenAI. Quest'ultima ha permesso di introdurre la logica intelligente e automatizzata nel processo di creazione del tool. Nel particolare, per la creazione dei contenuti testuali si è fatto affidamento al modello GPT-4o offerto da OpenAI, che offre una buona efficienza computazionale, e allo stesso tempo permette una generazione più veloce e un utilizzo delle risorse più economico.

Infatti, un obiettivo importante da raggiungere è un buon compromesso tra accuratezza del contenuto generato e velocità di generazione. Invece, per la generazione di immagini tramite IA si è fatto affidamento al modello Dall-e-2 di OpenAI. Quest'ultimo permette, a partire da un prompt in input, di combinare concetti, attributi e stili per la generazione di immagini.

Per l'utilizzo dei modelli è stata utilizzata una chiave privata di OpenAI che è stata protetta in un file `.env` all'interno del backend e non accessibile dal frontend e configurata come segue.

```
1 // .env
2 OPENAI_API_KEY = "la_mia_chiave_openai"
```

La chiave di OpenAI impostata come variabile d'ambiente è stata poi raggiunta dal file con le route generative tramite l'utilizzo della libreria *dotenv*. In questo modo è stato possibile utilizzare in modo sicuro l'API di OpenAI.

```
1 // generateRoutes.mjs
2 import { OpenAI } from "openai"
3 import dotenv from "dotenv"
4 ...
5 const generateRoutes = express.Router();
6
7 dotenv.config();
8
9 const openai = new OpenAI({
10   apiKey: process.env.OPENAI_API_KEY,
11 });
```

Come anticipato in fase di analisi dei requisiti, il problema che spesso emerge quando un utente si interfaccia con un LLM, è la scelta del prompt. Infatti, non è raro che l'utente non ottenga subito il risultato desiderato, o che fatichi a spiegare ciò che vuole. Questo problema può derivare spesso da una conoscenza limitata del funzionamento degli strumenti basati su LLM, e di conseguenza da

una formulazione errata delle richieste. Per questo motivo, CAT-AI ha il compito di esimere l'utente quanto più possibile dalla scelta dei prompt, permettendogli di guidare la generazione senza dover scrivere tutto il comando, come sarebbe invece necessario con strumenti generativi come ChatGPT. Di conseguenza, per rendere la generazione dei contenuti quanto più accurata possibile sono state applicate le regole principali alla base del prompt engineering (Ingegneria del prompt) di seguito elencate.

Chiarezza e Specificità : I prompt devono essere formulati in modo da non far emergere ambiguità. Nessun elemento deve essere omesso o non specificato. Soprattutto nel caso della creazione dell'esercizio informazioni su contenuti, struttura e attributi devono essere deterministiche.

Dare Contesto : Bisogna fornire tutti gli elementi e le informazioni utili alla generazione, offrendo un contesto generale.

Obiettivi Chiari : L'obiettivo di generazione deve essere esplicito per far comprendere cosa e come l'LLM deve rispondere. Spesso, è utile non solo specificare cosa si vuole ma anche ciò che non si vuole, per evitare del tutto possibili fraintendimenti.

Tono : Adattare il linguaggio e il tono della richiesta può essere utile per evidenziare determinati punti e anche per guidare il tono della risposta stessa.

Formattazione : La formattazione della domanda, come caratteri maiuscoli, o indentazione, o la separazione dei paragrafi è importante per separare semanticamente la richiesta.

Iterazione e Perfezionamento : Per applicare al meglio tutti i principi sopracitati è necessario testare e valutare la risposta dell'IA e migliorare il prompt sulla base di errori e incoerenze riscontrate.

Al livello pratico sono state seguite una serie di tecniche di prompting specifiche per rendere la risposta quanto più coerente e accurata possibile all'interno di CAT-AI.

Dare Ruoli : Dare un ruolo all'LLM è utile non solo per dare un contesto sull'argomento per cui andrà a generare dei contenuti, ma anche per aiutare a rispondere da una prospettiva particolare, migliorando qualità e pertinenza della risposta (es. "Sei un esperto di pedagogia e progettazione di esercizi per l'educazione").

Guida Passo dopo Passo : Per problemi complessi, come la creazione di esercizi didattici, che tengono conto di tanti fattori di generazione, una tecnica molto

utilizzata è quella di guidare l'LLM attraverso più compiti intermedi. Esattamente come un cervello umano, anche per l'LLM è più semplice risolvere più compiti semplici che uno complesso. L'utilizzo di elenchi puntati è molto utile per ottenere risposte più accurate.

Fornire Esempi : Per rendere ancora più specifiche le istruzioni a ogni step, è utile dare all'IA degli esempi di output atteso. In questo modo la risposta sarà ancora più deterministica e coerente con il risultato atteso.

Fornire Struttura : Nel caso specifico dell'implementazione, la struttura della risposta dell'IA dovrà corrispondere a un oggetto convertibile in formato JSON contenente tutte le informazioni richieste. Per questo motivo, per evitare errori di formattazione, è essenziale dare sempre un esempio di struttura completa dell'output atteso, specificando per ogni campo non solo il tipo di dato ma anche il range di valori che può assumere o avvertenze specifiche.

Obiettivo Finale : Una volta scritti tutti i comandi passo dopo passo all'interno del prompt, l'ultimo comando deve dire all'IA esattamente come processare e formattare tutti i dati raccolti per la risposta. Infatti, una caratteristica degli LLM è che rispondono come se stessero conversando, dando oltre alla risposta anche introduzione, spiegazioni e formattazione da testo (es. formattazione markdown). Per evitare ciò è stato necessario specificare all'IA di rispondere unicamente con l'oggetto JSON desiderato.

Tutti i percorsi delle routes per la generazione di contenuti iniziano con il prefisso `/api/generate` e sono di seguito elencate le loro funzionalità.

POST /exercise-info : Questa route serve per estrarre le informazioni a partire dal testo del file di riferimento fornito. Per cui, una volta che il testo è stato estratto dal file (PDF o immagine) tramite la route apposita, il testo viene dato in input a questa route per analizzarlo. L'IA avrà il compito di estrarne informazioni quali: scuola, classe, obiettivi di apprendimento e prerequisiti per restituirli al client. Le informazioni saranno utilizzate per il completamento automatico dei campi del form di generazione. Nel caso in cui il testo estratto dal file non corrisponda a un esercizio o sia incomprensibile, l'IA risponderà al client invece con un messaggio di errore, e quest'ultimo si occuperà di eliminare il file inserito e dare messaggio di errore all'utente, permettendogli di inserire un nuovo file.

POST /exercise : Questa route è quella addetta alla generazione iniziale dell'esercizio. Questa prende in input tutte le impostazioni di generazione elencate in Tabella 4.3, derivanti dal form di generazione e dalle impostazioni. Nel particolare questa route ha due funzioni gestite tramite due chiamate successive

all'API di OpenAI.

La prima chiamata ha il compito di creare l'esercizio e la soluzione corrispondente a partire dalle informazioni in input. Dovrà poi restituire una struttura JSON contenente un array di oggetti, in cui ogni oggetto corrisponde a una casella di testo, ciascuna con le rispettive caratteristiche elencate in tabella 4.2. Questa prima chiamata è perciò addetta non solo alla generazione dei contenuti delle caselle di testo di esercizio e soluzione, ma anche della formattazione e scelta di stile per ciascuna di esse.

La seconda chiamata prende in input il contenuto generato dalla prima chiamata, ossia l'oggetto contenente l'esercizio e la soluzione, e avrà il compito di analizzare tutte le caselle di testo dal punto di vista logico e/o matematico. Infatti è comune che l'IA faccia errori e non sia in grado di accorgersene in fase di generazione. Questa seconda chiamata imposta per ogni casella di testo la confidenza e la motivazione, avvisando il client di eventuali errori o incoerenze.

La scelta di un'unica route anziché due separate deriva dalla necessità di non voler allungare maggiormente il tempo di generazione della risposta finale.

POST /exercise-again : Questa route, accessibile tramite l'opzione di "Rigenerazione Esercizio" nella fase di personalizzazione serve per correggere, modificare o aggiungere contenuti testuali tramite un prompt in input. Nel caso della rigenerazione, non vengono più considerate tutte le impostazioni di generazione, ma il contesto è dato direttamente dalle caselle di testo presenti. Perciò viene dato in input l'elenco di tutte le caselle di testo e il comando, e viene generato un nuovo esercizio seguendo le istruzioni date.

POST /element-again : Parallelamente alla route precedente, questa è accessibile tramite la funzionalità di "Rigenerazione Elemento" in fase di personalizzazione, ma anche tramite la modalità "Fix" per la risoluzione problemi (Figura 4.13). Essa, dà la possibilità di rigenerare un'unica casella di testo selezionata all'interno dell'esercizio partendo da un prompt in input. Anche in questo caso per la rigenerazione il contesto è dato da tutte le caselle di testo e nello specifico da quella indicata (tramite id univoco in input). L'IA avrà il compito di restituire unicamente l'oggetto corrispondente alla casella di testo richiesta di modificare seguendo le istruzioni.

POST /solution-again : In questo caso viene offerta la possibilità di rigenerare la soluzione senza alcun prompt in input, semplicemente partendo dalle caselle di testo che contengono l'esercizio. Questa funzionalità è utile nel caso in cui è stato modificato l'esercizio manualmente e si vuole ricalcolare con un solo clic la nuova soluzione. L'IA avrà il compito di analizzare l'esercizio con la sua logica e generare la soluzione ben formattata e chiara in tutti i suoi passaggi.

POST /image : L'ultima route permette la generazione di immagini tramite prompt in input. Può essere chiamata tramite l'apposita funzione nella fase di personalizzazione mostrata in Figura 4.11. Questa route ha un duplice compito.

Il primo è di generare l'immagine seguendo il prompt in input e il secondo è di salvare la risorsa nella cartella esposta /uploads. Infatti al momento della generazione, l'immagine generata viene salvata temporaneamente nel server di OpenAI, ma per poter raggiungere la risorsa dal frontend e aggiungerla all'esercizio è necessario effettuare il fetch dalla risorsa dal backend al server di OpenAI per ottenere il buffer di dati dell'immagine. Solo una volta ottenuto potremo salvarlo nella cartella accessibile al frontend e visualizzarla nell'interfaccia.

Routes di Generazione Contenuti - /api/generate			
Percorso	Input	Output (JSON)	Funzione
POST /exercise-info	Testo Riferimento	Scuola, Classe, Obiettivi di Apprendimento, Prerequisiti	Estrae caratteristiche dal file di riferimento
POST /exercise	Impostazioni di Generazione	Caselle di Testo	Genera l'esercizio completo e lo verifica
POST /exercise-again	Caselle di Testo, prompt	Caselle di Testo	Rigenera l'intero esercizio seguendo l'istruzione in input
POST /element-again	Caselle di Testo, ID Elemento, Prompt	Casella di Testo Elemento	Modifica elemento seguendo istruzione in input
POST /solution-again	Caselle di Testo	Caselle di Testo Soluzione	Ricalcola soluzione in base all'esercizio
POST /image	Prompt	URL Immagine	Genera un'immagine seguendo istruzione in input

Tabella 4.5: Routes di Generazione Contenuti

Capitolo 5

Valutazione

Dopo aver raggiunto una versione finale e funzionante di CAT-AI, l'ultima fase di questa tesi esporrà lo svolgimento e l'analisi dei test di efficienza e usabilità dello strumento sviluppato. Infatti, testare l'applicazione è un passo essenziale per capire se l'applicazione sviluppata è realmente user-friendly sia per utenti inesperti, sia per utenti che utilizzano l'applicazione per la prima volta.

Allo stesso modo, i test sono importanti per capire se i risultati prodotti dal tool sono accurati e soddisfacenti per gli insegnanti, e rispettano quindi i loro standard. In particolare, per poter valutare con maggiore cognizione l'applicazione, e per poter trarre delle conclusioni veritiere dai test, CAT-AI è stato messo a confronto con una baseline. Nel nostro caso specifico, è stata scelta come baseline per la creazione di esercizi l'unione di ChatGPT e Word.

Questa scelta è stata fatta sulla base degli strumenti che sono emersi tra i più frequentemente utilizzati dagli insegnanti durante la prima fase di analisi. In questo capitolo, verranno in primo luogo esposti i metodi e i casi di test scelti e applicati per valutare e confrontare l'applicazione con la baseline. In secondo luogo, verranno esposti e confrontati i risultati ottenuti, mettendo a raffronto le aspettative e la realtà percepite dagli insegnanti. Infine, verranno evidenziati vantaggi e svantaggi emersi per entrambe le metodologie, analizzando limiti e punti di forza.

5.1 Metodi

Per lo svolgimento dei test, è stata in primo luogo svolta una ricerca dei possibili insegnanti candidati, basandosi su fattori quali esperienze di insegnamento, conoscenza delle tecnologie utilizzate come baseline, propensione all'utilizzo di nuove tecnologie per la creazione di esercizi didattici. In questa sezione verrà quindi esposta la metodologia di selezione dei partecipanti e in seguito esposta la struttura del test ideata per una valutazione corretta e concreta di CAT-AI.

5.1.1 Selezione dei Partecipanti

Come anticipato, per valutare l'efficienza di CAT-AI è stato deciso di metterlo a confronto con una baseline che è ChatGPT + Word, per questo motivo la ricerca dei partecipanti è stata limitata a insegnanti con una conoscenza almeno basilare di questi due strumenti. Per quanto riguarda la materia di riferimento, invece, è stato deciso di testare l'applicazione sulla creazione di esercizi di italiano e matematica parallelamente, poiché esse sono due materie caratterizzanti fin dalle elementari. Per cui, gli insegnanti ricercati dovevano avere esperienze di insegnamento in una di queste due materie sopra citate.

Inoltre, sono stati esclusi dai test di usabilità gli insegnanti che avevano partecipato all'intervista iniziale per la raccolta dei requisiti. Questa scelta è stata fatta in modo che i nuovi partecipanti avessero le minori informazioni possibili sull'applicazione, e di conseguenza non fossero influenzati positivamente o negativamente nel suo uso. Infine, veniva richiesta per lo svolgimento del test una conoscenza base della lingua inglese, dato che CAT-AI è stato creato in inglese.

Per cui sono stati trovati e scelti 8 insegnanti (4 di matematica e 4 di italiano) per partecipare al test di efficienza e usabilità. Per lo svolgimento del test, è stato fatto compilare ai partecipanti un modulo di consenso per la raccolta dei dati, e sono state richieste informazioni generali su strumenti utilizzati abitualmente per la creazione di esercizi. La metà dei partecipanti ha affermato di utilizzare ChatGPT circa una volta a settimana per la creazione di esercizi; solo una persona ha invece affermato di non utilizzarlo quasi mai per la creazione di esercizi; e solo una quasi giornalmente. In generale, i partecipanti hanno affermato di essere mediamente favorevoli all'integrazione di ChatGPT per la creazione di esercizi. Nel particolare, il livello di favorevolezza, su scala Likert a 7 punti, è risultato con un punteggio medio di 5,625 su 7.

5.1.2 Prevenzione dei Bias

Per l'impostazione della struttura del test è stato creato un copione a punti da seguire durante ogni test per dare uguale spiegazione e informazione a tutti i partecipanti, e poter quindi valutare tutti i test con uno stesso metro di paragone. Per lo svolgimento dei test, è stato inoltre preso in considerazione un recente studio che analizzava i possibili preconcetti e interpretazioni creati dagli utenti nei confronti dell'IA durante l'interazione con nuove tecnologie [10].

Questo studio evidenzia come i pregiudizi dell'utente possano influenzare l'usabilità stessa di nuove tecnologie. Per questo motivo, per mitigare questo effetto, è stato scelto di evitare di nominare l'IA all'interno del copione, applicando una "censura" su ciò che potesse creare aspettative o imparzialità all'utente nei confronti di uno strumento intelligente, CAT-AI. Per lo stesso motivo anche l'applicazione stessa è stata "censurata" ai fini del test, modificando il nome CAT-AI in CAT-TUTOR.

5.1.3 Struttura del Test

La struttura del Test si divide essenzialmente in 5 punti di seguito elencati.

1. **Introduzione e Panoramica:** La prima fase introduttiva è stata creata in modo da spiegare all'utente gli obiettivi e lo svolgimento del test. Con una breve introduzione iniziale viene perciò enunciato il proposito di testare due metodologie parallele per la creazione di esercizi. Viene inoltre richiesto all'utente di svolgere i test dando giusta attenzione a possibili errori e bisogni personali, come se dovesse creare un esercizio per la propria classe. In seguito, vengono esposte all'utente le due metodologie che andrà a testare, spiegando in modo generico cosa gli verrà chiesto di fare.
2. **Questionari Aspettative:** Prima di poter testare le due metodologie, sono stati creati due questionari di usabilità personalizzati con 6 domande con scala Likert da 1 a 7 sulle aspettative per le due metodologie. I due questionari sono esattamente paralleli nelle domande, per poter permettere un corretto confronto. Le aspettative riguardano sia l'efficienza che l'usabilità. Ovviamente nel caso della baseline (ChatGPT + Word) l'aspettativa sarà anche influenzata dall'esperienza pregressa positiva o negativa dell'utente.
3. **Test Metodologie:** Dopo aver compilato i questionari sulle aspettative inizia il test effettivo delle due metodologie. Per l'ordine di test è stato deciso di applicare un metodo bilanciato, perciò la metà dei partecipanti (2 matematica e 2 italiano) testeranno prima la baseline e in seguito CAT-AI, i restanti invece inizieranno con il test di CAT-AI e proseguiranno con la baseline. Questa scelta permette infatti di avere una visione coerente e equilibrata delle due metodologie, evitando possibili vantaggi o svantaggi dati dal loro ordine di utilizzo.
4. **Questionari Post-Utilizzo:** Una volta svolti entrambi i test, al partecipante verrà chiesto di compilare due questionari post-utilizzo del tutto analoghi a quelli sulla aspettative. In questo modo sarà possibile confrontare come cambiano le aspettative che l'utente può essersi creato prima dell'utilizzo con la realtà emersa dopo il test effettivo.
5. **Intervista:** Infine, l'ultima fase del test consiste in un'intervista semi-strutturata per far emergere in modo più chiaro vantaggi e svantaggi delle due metodologie. L'intervista rende inoltre possibile capire eventuali difficoltà emerse durante il test e le rispettive motivazioni, o allo stesso modo permette di verificare le modifiche che il partecipante ha effettuato durante la creazione per rendere l'esercizio adeguato. L'intervista semi-strutturata permette perciò di avere una visione completa e esaustiva dell'opinione del partecipante.

5.1.4 Casi d'Uso Testati

Entrando nello specifico nella fase di test delle due metodologie, è stato scelto un caso d'uso di partenza da cui sono stati declinati vari compiti (task) da distribuire in modo bilanciato tra le due metodologie. Nel particolare è stato scelto come caso d'uso di partenza, la creazione di esercizi a partire da un esercizio preesistente di riferimento. Infatti, come emerso dai requisiti, la possibilità di poter creare un esercizio a partire da uno già verificato e fidato dall'insegnante, ha la possibilità di migliorare la soddisfazione oltre che la fiducia dell'insegnante nei confronti dell'esercizio generato, oltre che nei confronti dell'applicazione stessa.

Come anticipato, le materie di riferimento sono matematica e italiano, per cui per lo svolgimento dei test sono stati creati 2 seed (semi) di riferimento con una struttura fissa e semplice, uno per ciascuna delle due materie. Entrambi i riferimenti contengono un'esercitazione con un titolo e 3 esercizi diversi, ciascuno con una consegna. Nel caso di CAT-AI verrà perciò richiesto all'utente di seguire la navigazione dell'applicazione per poter scegliere di inserire un file di riferimento (Modalità "Upload a File"), mentre per la baseline l'utente potrà alternativamente inserire il file come allegato o copiare e incollare il testo dell'esercizio di riferimento nella conversazione in input, insieme al comando di generazione. Entrando nel merito dei compiti assegnati agli insegnanti, ne sono stati creati 2 per ciascuna materia enunciati come segue.

- **Matematica 1:** Creare un'esercitazione sulle addizioni e sottrazioni per la classe 2 elementare.
- **Matematica 2:** Creare un'esercitazione sulle equazioni di primo grado per la classe 1 media.
- **Italiano 1:** Creare un'esercitazione sui verbi passivi e attivi per la classe 2 media.
- **Italiano 2:** Creare un'esercitazione su articoli e le preposizioni per la classe 4 elementare.

Per cui, ogni partecipante dovrà svolgere entrambe le task della propria materia di riferimento (es. insegnante di italiano svolge prima italiano 1 e poi italiano 2) ma utilizzando diverse metodologie. In questo modo, seguendo un ordine bilanciato di utilizzo per le due metodologie come spiegato precedentemente, risulteranno infine quattro esercitazioni create per ciascuna task (2 create con la baseline e 2 create con CAT-AI). Questo metodo incrociato permetterà perciò di avere un confronto tra approcci e metodi di creazione per lo stesso esercizio, ma tra tecnologie e insegnanti diversi.

5.2 Risultati Quantitativi

Per quanto riguarda lo svolgimento dei test, una parte di essi è stata svolta direttamente in presenza con i partecipanti, mentre un'altra parte è stata svolta tramite la piattaforma Teams, utilizzando la funzionalità di controllo remoto. Nella prima parte di questa sezione si metteranno a confronto i tempi di creazione degli esercizi con le due metodologie.

Durante lo svolgimento, i tempi di creazione sono stati presi in modo simultaneo e preciso da CAT-AI, che creava un riassunto finale con i tempi totali e relativi per ciascuna fase del processo (generazione, personalizzazione, esportazione). Al contrario, per tenere i tempi della baseline è stata effettuata la registrazione dello schermo, e in un secondo momento è stato calcolato il tempo per le 3 fasi. Per la divisione precisa e più semplice in 3 fasi della baseline, si è considerata una fase iniziata solo nel momento in cui la fase precedente è stata completamente terminata. Per cui, prevedendo la baseline l'utilizzo di due strumenti separati, è stata considerata iniziata la fase di personalizzazione con Word, solo una volta che l'utente ha terminato completamente la generazione con ChatGPT. In seguito ai tempi, verranno analizzati i risultati quantitativi ottenuti tramite i questionari, confrontando punteggi ottenuti su aspettative e realtà per le due metodologie, e misurando il livello di soddisfazione o insoddisfazione degli utenti.

5.2.1 Confronto dei Tempi

In media, i test, comprendendo anche le parti introduttive di spiegazione, la compilazione dei sondaggi e l'intervista finale, hanno avuto una durata media di circa 30-40 minuti. Come possiamo vedere dalla tabella 5.1, è risultato che in media gli utenti hanno impiegato meno di 10 minuti per creare gli esercizi con CAT.AI e meno di 9 minuti per creare gli esercizi con ChatGPT + Word, con circa un minuto di differenza. Nello specifico gli utenti hanno impiegato, utilizzando la baseline, circa 30 secondi in meno rispetto a CAT-AI per la creazione di esercizi di italiano, e circa 1 minuto e 30 secondi in meno per la creazione di esercizi di matematica.

TEST	CAT-AI	CHATGPT+WORD
Italiano	9m 28s	8m 49s
Matematica	10m 00s	8m 19s
TOT	9m 44s	8m 34s

Tabella 5.1: Tempi Medi Test Italiano e Matematica

Per avere un'ulteriore analisi sui tempi impiegati dagli utenti nello svolgimento delle diverse task assegnate, è possibile effettuare un confronto incrociato tra le due metodologie. In particolare, per quanto riguarda i tempi di creazione di Italiano (Tabelle 5.2 e 5.3), in riferimento alla task 1, gli insegnanti hanno impiegato in media circa un minuto in meno per creare gli esercizi con la baseline, e solo 20 secondi in meno in riferimento alla seconda task. Invece, per quanto riguarda matematica (Tabelle 5.4 e 5.5), le distanze sui tempi incrementano, registrando un tempo di creazione di circa 2 minuti in meno utilizzando la baseline per la prima task, e di circa 1 minuto e 20 secondi in meno per la seconda task.

Queste differenze, in termini di tempo relativo, potrebbero sembrare a primo impatto elevate, ma analizzando i tempi assoluti registrati, possiamo notare che entrambe le metodologie hanno permesso di creare degli esercizi didattici in un tempo minore o uguale a 10 minuti. Inoltre, per avere una visione più specifica su come l'utente abbia realmente impiegato il tempo durante la creazione, è necessario effettuare ulteriori considerazioni. Infatti, il primo fattore da tenere in considerazione nel confronto dei tempi tra le due metodologie è il tempo di generazione dei contenuti. La baseline, grazie alla potenza di ChatGPT, ha avuto la possibilità di generare i contenuti in tempo trascurabile (2-3 secondi), mentre CAT-AI, dovendo comunicare con l'API di OpenAI per mezzo del backend, e

<i>ITALIANO 1</i>		
FASE	CHATGPT+WORD	CAT-AI
Generation	2m 53s	2m 54s
Customization	3m 40s	8m 11s
Export	0m 12s	0m 14s
TOT	6m 45s	11m 19s
<i>ITALIANO 2</i>		
FASE	CHATGPT+WORD	CAT-AI
Generation	2m 46s	3m 01s
Customization	4m 15s	6m 33s
Export	0m 49s	0m 21s
TOT	7m 50s	9m 55s

Tabella 5.2: Risultato Tempi Test Italiano 1 e 2

implementando quindi una logica di comunicazione più complessa, ha introdotto inevitabilmente una latenza molto maggiore. Nello specifico, è stato calcolato che il tempo medio di attesa per la prima generazione dell'esercizio tramite CAT-AI, è stato di circa 50 secondi. Questo tempo prolungato è stato dato dal fatto che la prima generazione dell'esercizio, come spiegato nello scorso capitolo, comprende una doppia richiesta. La prima chiamata si occupa della creazione dei contenuti e la seconda della loro verifica, dilungando inevitabilmente i tempi di attesa in favore di una maggiore affidabilità.

Inoltre, a questo tempo di latenza, si aggiungono ulteriori tempi di attesa minori dati dall'estrazione delle caratteristiche dal file di riferimento e da eventuali tempi di rigenerazione. Per cui, possiamo constatare che, mentre il tempo di creazione registrato per la baseline equivale circa anche al tempo di domanda cognitiva richiesta all'utente per la creazione dell'esercizio; al contrario, il tempo registrato per CAT-AI conta circa 1 minuto in più in cui l'utente non ha dovuto lavorare attivamente per la creazione dell'esercizio. Di conseguenza, possiamo concludere che il tempo di domanda cognitiva e di tempo effettivamente impiegato dagli insegnanti per la creazione degli esercizi è approssimativamente equivalente per i due metodi.

<i>ITALIANO 3</i>		
FASE	CAT-AI	CHATGPT+WORD
Generation	4m 23s	3m 35s
Customization	1m 21s	2m 31s
Export	0m 40s	0m 54s
TOT	6m 24s	7m 00s
<i>ITALIANO 4</i>		
FASE	CAT-AI	CHATGPT+WORD
Generation	4m 11s	6m 09s
Customization	5m 31s	7m 05s
Export	0m 30s	0m 28s
TOT	10m 12s	13m 42s

Tabella 5.3: Risultato Tempi Test Italiano 3 e 4

<i>MATEMATICA 1</i>		
FASE	CHATGPT+WORD	CAT-AI
Generation	5m 58s	3m 29s
Customization	0m 55s	5m 07s
Export	0m 44s	0m 26s
TOT	7m 37s	9m 02s

<i>MATEMATICA 2</i>		
FASE	CHATGPT+WORD	CAT-AI
Generation	8m 30s	3m 08s
Customization	0m 18s	6m 26s
Export	0m 45s	0m 12s
TOT	9m 33s	9m 46s

Tabella 5.4: Risultato Tempi Test Matematica 1 e 2

<i>MATEMATICA 3</i>		
FASE	CAT-AI	CHATGPT+WORD
Generation	4m 46s	5m 51s
Customization	8m 26s	1m 39s
Export	0m 17s	0m 33s
TOT	13m 29s	8m 03s

<i>MATEMATICA 4</i>		
FASE	CAT-AI	CHATGPT+WORD
Generation	3m 18s	1m 53s
Customization	4m 10s	5m 32s
Export	0m 17s	0m 37s
TOT	7m 45s	8m 02s

Tabella 5.5: Risultato Tempi Test Matematica 3 e 4

5.2.2 Confronto Aspettative e Realtà

La sola analisi dei tempi di creazione e di domanda cognitiva non è sufficiente per una comprensione effettiva del livello di soddisfazione finale recepito dagli insegnanti. Le risposte ai questionari di pre-utilizzo delle metodologie e post-utilizzo hanno permesso di comprendere in che misura gli insegnanti sono stati influenzati dai possibili preconcezioni e pregiudizi creati per la baseline e per CAT-AI. Allo stesso modo, hanno permesso poi di verificare quanto le aspettative create siano state soddisfatte o deluse.

Per quanto riguarda la baseline, deve essere considerato il fatto che, essendo uno strumento già conosciuto e utilizzato, l'aspettativa è stata vincolata dall'esperienza stessa. Al contrario, nel caso di CAT-AI l'aspettativa è stata interamente creata durante la spiegazione introduttiva sul suo funzionamento generale. Inoltre, sebbene siano state adottate delle tecniche di riduzione dei bias cognitivi derivanti dall'utilizzo di intelligenza artificiale, non è stato sicuramente possibile eliminarli del tutto. Per la misurazione del livello di soddisfazione attesa e risultante sono state utilizzate 6 domande a risposta chiusa con una scala Likert a 7 punti, dove 1 è equivalente a "Molto in disaccordo" e 7 equivale a "Molto d'accordo".

Per l'analisi complessiva è stata effettuata una media tra i punteggi dati dagli 8 partecipanti per ogni domanda. Come possiamo notare dalla tabella 5.6, CAT-AI ha ricevuto aspettative maggiori della baseline in 4 punti su 5. Il dubbio principale sull'applicazione creata (punteggio <5) ha riguardato la reale possibilità di proporre gli esercizi creati ai propri alunni, e quindi la sua adattabilità a contesti reali. Al contrario, la baseline ha creato maggiori dubbi, in particolare sul numero di modifiche da effettuare per ottenere un risultato soddisfacente, sull'affidabilità delle risposte, e infine sull'allineamento tra domanda e risposta ricevuta.

Perciò, come previsto, l'esperienza di utilizzo pregressa della baseline ha influenzato le aspettative, grazie alla conoscenza di vantaggi e svantaggi offerti da essa. Al contrario, CAT-AI ha creato aspettative medio alte (5.75/7) solo per quanto riguarda la velocità di creazione degli esercizi. Questo fattore è probabilmente dato dal fatto che è stato spiegato agli utenti che l'applicazione avrebbe permesso di creare, modificare e esportare l'esercizio senza l'utilizzo di strumenti ulteriori, al contrario della baseline. Analizzando i risultati dati dopo l'utilizzo delle due metodologie è possibile notare come CAT-AI abbia aumentato tutti i punteggi superando le aspettative degli utenti su usabilità, qualità e affidabilità.

Al contrario, la baseline ha in alcuni casi perfino in parte deluso le aspettative, ottenendo risultati minori in 2 domande su 5. Il dato che più spicca alla luce delle analisi fatte, è quello riguardante la velocità di creazione degli esercizi. In particolare, CAT-AI ha raggiunto un punteggio finale di 6,125, contro i 5,125 della baseline, con un punto di distacco nella scala. Ma come analizzato nello scorso paragrafo, i tempi assoluti di creazione degli esercizi sono stati mediamente più

lunghe con l'utilizzo di CAT-AI di circa un minuto. Questo risultato mette in evidenza come gli utenti abbiano ricevuto una percezione di maggiore scorrevolezza, e minore frustrazione tramite l'utilizzo di CAT-AI, rispetto all'utilizzo di due strumenti separati.

A supporto di ciò, è possibile notare anche il punteggio risultato nella domanda sull'usabilità, con 6,125 di CAT-AI contro i 5,25 della baseline. Questo fattore evidenzia come l'applicazione abbia permesso una creazione più agevole dell'esercizio nonostante gli utenti avessero esperienza di utilizzo pregresse con la baseline. Tra gli elementi rilevanti, possiamo notare anche come l'inserimento degli obiettivi di apprendimento per la creazione dell'esercizio con CAT-AI abbia favorito l'allineamento tra richieste e risultati (5.5/7), e la conseguente affidabilità del sistema (6,625). Al contrario la baseline, affidandosi totalmente alle abilità di prompting dell'utente, ha prodotto risposte meno affidabili (5,25) e meno allineate agli obiettivi (5,625), a paragone con CAT-AI. Nonostante ciò, per entrambe le metodologie sono stati ottenuti punteggi superiori al 5 per tutte le domande su qualità e usabilità poste, indicando una buona soddisfazione generale nella creazione degli esercizi per entrambi i casi di studio.

Domanda	CAT-AI		ChatGPT+Word	
	Prima	Dopo	Prima	Dopo
Aiuta a creare più velocemente gli esercizi	5,75	6,125	5,625	5,125
Riduce numero di modifiche da fare per avere una scheda completa	5	5,875	4,125	5,125
È facile e intuitivo da utilizzare	5,25	6,125	5,5	5,25
Produce risposte corrette e affidabili	5	6,625	4,375	5,25
Gli esercizi prodotti sono allineati agli obiettivi richiesti	5,125	6,5	4,875	5,625
Gli esercizi prodotti sono adatti per essere proposti ai miei alunni	4,75	5,875	5	5,25

Tabella 5.6: Confronto Aspettative e Realta Test

5.3 Analisi Qualitativa

I dati quantitativi raccolti durante i test hanno permesso di delineare sommariamente il livello di soddisfazione degli utenti, nonché i livelli generali di usabilità e accuratezza forniti dalle due metodologie. Invece, l'analisi qualitativa sui risultati dei test è stata possibile grazie all'intervista semi-struttura finale. Quest'ultima, ha permesso di mettere in luce vantaggi e svantaggi delle due metodologie con maggiore accuratezza, e anche di capire possibili frustrazioni o difficoltà emerse durante la creazione.

5.3.1 CAT-AI: Vantaggi e Svantaggi

La prima parte dell'intervista ha mirato a comprendere i limiti e i punti di forza di entrambe le metodologie, mettendole a confronto tra loro. Da ciò è stato possibile stilare una lista di vantaggi e svantaggi per ciascuna metodologia.

Vantaggi I principali vantaggi evidenziati dagli utenti per CAT-AI sono di seguito elencati in ordine di rilevanza.

6 insegnanti su 8 hanno evidenziato tra i punti di forza l'intuitività dell'interfaccia. Gli utenti hanno apprezzato particolarmente il processo di creazione guidata nelle varie fasi. In particolare, la prima fase di inserimento dei dati e generazione è emersa come particolarmente utile e facilmente utilizzabile per generare l'esercizio, grazie anche al fatto che non richiedeva la scrittura di un prompt.

5 insegnanti su 8 hanno invece evidenziato il vantaggio di avere un'unica applicazione che permetta di generare, personalizzare e esportare l'esercizio pronto per essere stampato. Come analizzato precedentemente, questo fattore ha anche influito sulla percezione di un tempo di creazione minore, oltre che di una usabilità maggiore.

La metà degli insegnanti hanno evidenziato invece la facilità con cui è possibile personalizzare l'esercizio sia manualmente che tramite le funzionalità di IA offerte. Tra i partecipanti ai test, solo 2 non hanno usufruito della personalizzazione tramite IA, preferendo una modifica manuale.

Un numero minore di partecipanti ha evidenziato la precisione di CAT-AI nella creazione dell'esercizio. Quest'ultimo ha permesso di creare esercizi ben mirati e specifici grazie ai requisiti inseriti. Allo stesso modo gli utenti hanno sottolineato l'allineamento corretto dei contenuti agli obiettivi richiesti.

Tra gli altri vantaggi emersi singolarmente tra i partecipanti troviamo l'adattabilità dell'applicazione a diversi contesti e tipologie di esercizi e la sua specificità nell'ambito scolastico per la creazione di esercizi. Inoltre, nell'unico test in cui è stato generato un errore logico, l'utente ha elencato tra i vantaggi il fatto che il

sistema si fosse accorto del suo stesso errore (tramite la confidence flag) e avesse dato la possibilità di correggerlo facilmente (tramite funzione “Fix”).

Svantaggi Di seguito sono invece menzionati i principali svantaggi o limiti emersi e sottolineati dagli utenti durante le interviste. Non sono stati considerati gli svantaggi derivanti dal fatto che gli utenti non hanno potuto testare le piene funzionalità e impostazioni dell’applicazione, come per esempio “è necessario utilizzare un file per creare l’esercizio” oppure “l’esercizio creato era molto semplice”. Infatti, le impostazioni generali di generazione e la modalità di creazione sono state decise in modo univoco per tutti i test per avere un metro di paragone coerente.

Lo svantaggio che è stato fatto notare maggiormente durante le interviste è stato il fatto che le caselle di testo non permettessero di modificare lo stile di parole o gruppi di parole singoli, ma solo dell’intera casella di testo. Per gli insegnanti è infatti utile sottolineare o evidenziare parole singole per metterle in risalto e dare una comprensione maggiore della consegna.

Un altro punto a sfavore di CAT-AI è la sua poca varietà nella generazione degli esercizi, nel particolare nelle tipologie scelte. Infatti, quest’ultimo crea tipologie di esercizi molto simili tra loro, o molto simili al riferimento, dando poco spazio alla creatività.

In due casi distinti, gli utenti hanno fatto notare che l’applicazione avesse sovrapposto alcune caselle di testo generate, che sono dovute essere sistemate manualmente. Per cui l’impaginazione generata non è stata del tutto corretta.

Un’utente ha invece evidenziato il limite dato dal fatto che l’applicazione offre la possibilità di creare esercizi di una sola pagina. In alcuni casi sarebbe infatti utile poter strutturare esercitazioni o verifiche più lunghe.

Solo un utente ha invece evidenziato i tempi di attesa prolungati di generazione rispetto alla baseline. Un altro ha invece nominato tra gli svantaggi il fatto di essersi dovuto abituare a una nuova interfaccia. Infine, un utente ha detto di aver avuto difficoltà a comprendere che per inserire gli obiettivi di apprendimento e i prerequisiti doveva premere il tasto “Invio”.

5.3.2 Baseline: Vantaggi e Svantaggi

Come per CAT-AI, anche per la baseline è stato possibile delineare una serie di vantaggi e svantaggi introdotti durante la creazione degli esercizi. In questo caso le caratteristiche citate sono riferite alternativamente a limiti e punti di forza di entrambi gli strumenti utilizzati singolarmente, e in particolare alla loro unione per la creazione di esercizi.

Vantaggi Alcuni vantaggi derivano in modo diretto dall’esperienza e capacità di utilizzo delle tecnologie da parte degli utenti. Altri derivano invece dalle maggiori

capacità funzionali e computazionali degli strumenti della baseline. Di seguito sono elencati i vantaggi esposti dagli insegnanti in ordine di rilevanza.

La metà dei partecipanti ha evidenziato come word, offrisse molte più possibilità e strumenti di modifica per personalizzare l'esercizio. Tra questi, ovviamente, la possibilità di modificare stile e colore di parole o gruppi di parole singoli.

Sempre 4 partecipanti su 8, hanno espresso, tra i vantaggi, il fatto che ChatGPT ha creato tipologie di esercizi più varie, sottolineando la maggiore intelligenza e creatività in questo ambito. Spesso ChatGPT creava anche esercizi più lunghi o più articolati, dando più possibilità di scelta, nonostante prendesse anch'esso come riferimento iniziale il testo dell'esercizio già pronto.

Alcuni partecipanti hanno elencato tra i vantaggi il fatto che ChatGPT offra una generazione dei contenuti molto veloce. Infatti, per utenti con maggiore esperienza che non hanno troppe difficoltà nella scrittura del prompt, la generazione dei contenuti iniziale può essere molto veloce.

2 partecipanti hanno elencato tra i vantaggi il fatto che ChatGPT abbia creato esercizi ben impostati dal punto di vista testuale. In uno di questi casi, l'utente è anche riuscito a farsi dare da ChatGPT direttamente il file docx di Word con l'esercizio pronto da poter modificare senza bisogno di copia e incolla.

Tra gli altri vantaggi elencati singolarmente possiamo trovare l'affidabilità delle risposte e la flessibilità di utilizzo offerta da ChatGPT. Allo stesso modo, la possibilità di eliminare quasi totalmente il lavoro manuale continuando a fare richieste a ChatGPT. Infine, legato a ciò, la possibilità di avere una maggiore interazione e di poter spiegare meglio ciò che si vuole tramite il prompt. Tutti questi ulteriori fattori di vantaggio si sono verificati in concomitanza con utenti abili o con maggiore esperienza nell'utilizzo di ChatGPT e in particolare nella scrittura di prompt di richiesta.

Svantaggi Di seguito sono allo stesso modo analizzati gli svantaggi emersi relativamente alla baseline durante le interviste.

Parallelamente ai vantaggi di CAT-AI, è stato evidenziato dalla metà dei partecipanti lo svantaggio di dover utilizzare due strumenti per creare un esercizio completo nel caso della baseline. Gli utenti hanno trovato una perdita di tempo il dover copiare e incollare i contenuti tra uno strumento e l'altro, ma soprattutto dover fare avanti e indietro in caso di necessità di aggiungere o modificare elementi tramite ChatGPT. Alcuni utenti, per superare parzialmente questo limite, hanno effettuato numerose richieste a ChatGPT per avere un esercizio completo, riducendo al minimo la modifica manuale tramite Word.

3 utenti hanno elencato tra gli svantaggi principali la difficoltà nel farsi comprendere da ChatGPT tramite il prompt. Infatti, gli utenti con minore esperienza non sono riusciti a ottenere alla prima generazione quello che volevano, dovendo specificare in seguito cosa correggere.

2 utenti hanno fatto invece notare lo svantaggio dato dal fatto che ChatGPT inserisca insieme all'esercizio anche spiegazioni o contenuti testuali di interazione con l'utente non richiesti. In questi casi, utilizzando la funzione "copia e incolla", sarà poi necessario eliminare manualmente la formattazione e il contenuto non richiesto.

Mentre tra i vantaggi è stata citata la capacità di modifica testuale con word, tra gli svantaggi è stato invece fatto notare da due utenti che è complicato creare impaginazioni personalizzate con Word, non supportando un editor grafico come CAT-AI.

Tra gli altri svantaggi citati è possibile citare il fatto che ChatGPT spesso utilizza un vocabolario complesso o non sempre adatto a alunni più piccoli, per esempio delle elementari. In un caso singolo, l'utente ha invece fatto notare come la funzione "copia e incolla" di ChatGPT abbia creato un errore di formattazione in tutte le funzioni di primo grado generate, costringendolo a eliminare e correggere la formattazione sbagliata.

5.3.3 Analisi delle Modifiche

L'ultimo studio effettuato a partire dalle interviste agli insegnanti ha riguardato l'analisi delle modifiche apportate durante i test per raggiungere un risultato soddisfacente e corretto. In particolare, sono state analizzate le rigenerazioni con IA effettuate nei due metodi, dando particolare attenzione alle motivazioni che hanno portato a tali modifiche. Alcune di queste motivazioni sono state parzialmente introdotte tra gli svantaggi elencati per entrambe le metodologie. Nella tabella 5.7 è possibile analizzare il numero di rigenerazioni con IA effettuate per ciascun test con le due metodologie.

Nel caso di ChatGPT viene considerata come "rigenerazione" qualsiasi richiesta successiva alla generazione iniziale dell'esercizio. Il numero totale di rigenerazioni è pressoché lo stesso tra le due tecnologie: 13 rigenerazioni totali con CAT-AI e 14 con la baseline. Ciò che cambia sono i motivi principali per cui l'utente ha effettuato tali cambiamenti.

Infatti, durante l'utilizzo di CAT-AI, nella maggioranza dei casi (5) la funzionalità di rigenerazione è stata utilizzata per cambiare o aggiungere un esercizio specificando la tipologia desiderata. Questo è riferimento dello svantaggio relativo alla poca creatività risultata da CAT-AI. Solo in un caso l'applicazione ha generato un errore logico, che è stato però segnalato e corretto tramite la funzione di IA apposita. E solo in un caso CAT-AI ha generato un esercizio che è stato modificato per le consegne poco chiare.

Invece, per quanto riguarda ChatGPT, la maggior parte delle rigenerazioni sono avvenute per errori di generazione o fraintendimenti. In 2 casi ChatGPT ha generato esercizi con le risposte già evidenziate, e in altri 2 casi ha invece generato

un esercizio troppo articolato, ed è stato richiesto di modificarlo. Nella tabella, possiamo analizzare come due utenti (Matematica 1 e 3), abbiano dovuto effettuare numerose rigenerazioni (5) per raggiungere il risultato desiderato. Concludendo, le rigenerazioni effettuate con la baseline sono state per lo più causate da errori o difficoltà nello spiegare esattamente a ChatGPT cosa si voleva.

Al contrario, con CAT-AI la maggior parte delle modifiche sono state apportate per personalizzare le tipologie degli esercizi creati in modo più mirato. Questo fattore ha sicuramente influito sui risultati in tabella 5.6, in cui veniva evidenziato come la baseline avesse dato un senso di minore affidabilità e minore allineamento delle risposte rispetto a quanto richiesto.

TEST	CAT-AI	CHATGPT+WORD
Italiano 1	1	0
Italiano 2	0	1
Italiano 3	0	1
Italiano 4	2	1
Matematica 1	3	5
Matematica 2	3	1
Matematica 3	3	5
Matematica 4	1	0

Tabella 5.7: Numero di Rigenerazione per Test

Sommario Per cui, le analisi effettuate hanno messo in luce limiti e punti di forza di entrambe le metodologie. Seppur i tempi di creazione siano stati leggermente maggiori con l'utilizzo dell'applicazione creata, entrambi i metodi hanno dato la possibilità di generare esercizi pronti in meno di 10 minuti in media, in linea con i requisiti non funzionali in tabella 2.2. Mentre c'è ancora un margine di miglioramento sui tempi di generazione, che superano quanto era stato prefissato nei requisiti.

Nonostante questo, CAT-AI ha dato agli insegnanti una percezione di maggiore usabilità e fluidità durante la creazione dell'esercizio, ricevendo un riscontro migliore di quello ricevuto dalla baseline anche sui tempi di creazione. Per quanto riguarda la qualità e l'affidabilità degli esercizi creati, tutti gli insegnanti hanno apprezzato la specificità e coerenza dei contenuti generati da CAT-AI, oltre che la facilità con cui è possibile personalizzare e guidare la creazione. Infatti, quasi tutti i partecipanti hanno fatto utilizzo e apprezzato le funzionalità basate su IA offerte per la personalizzazione intelligente degli esercizi.

Nome:	Cognome:	Data:
Esercizio sui Verbi Attivi e Passivi Leggi attentamente le frasi e indica se i verbi sono attivi o passivi. <i>Nota: Ricorda che un verbo è attivo quando il soggetto compie l'azione, mentre è passivo quando il soggetto subisce l'azione.</i> Esercizio 1: Leggi le seguenti frasi e individua il verbo e specifica se il verbo è attivo o passivo. a) La lettera è stata scritta da Marco. b) Il cane insegue il gatto. c) La torta è stata mangiata dai bambini. Esercizio 2: Leggi le seguenti frasi e indica se l'affermazione sul verbo è vera o falsa. a) Il libro è stato letto da Anna. (Il verbo è attivo) b) Maria prepara la cena. (Il verbo è passivo) c) La macchina è stata lavata da Luca. (Il verbo è passivo) Esercizio 3: Indica se i seguenti verbi sono attivi o passivi e riscrivi la frase nel modo opposto. a) La pizza è stata cucinata da Paolo. b) Il gatto cattura il topo. c) La canzone è stata cantata da Elisa.		
Nome e cognome VERIFICA ARTICOLI E PREPOSIZIONI 1) Leggi le frasi e sottolinea in blu gli articoli determinativi e in rosso gli articoli indeterminativi. a) Ho visto un cane nel giardino. b) La mamma ha preparato una torta buonissima. c) Vorrei uno zaino nuovo per la scuola. d) I bambini giocano felici al parco. 2) Nelle frasi seguenti cerchia le preposizioni semplici. a) Vado a scuola con il mio amico. b) Ho messo il libro su quel tavolo. c) Il gatto si è nascosto sotto il letto. d) Rimaniamo per tutta la giornata. 3) Unisci la preposizione con l'articolo e scrivi la giusta preposizione articolata come nell'esempio. Esempio: di + la = della a) di + i = b) a + le = c) con + il = d) su + gli = e) da + lo =		

a) CAT-AI

b) CHATGPT+WORD

Figura 5.1: Esempio Esercizi Creati Test Italiano 4

Parallelamente, sono anche stati messi in luce limiti e possibili svantaggi introdotti dal tool, aprendo nuove porte per sviluppi futuri. Complessivamente, la maggior parte dei partecipanti ha affermato di essere stata soddisfatta dei risultati ottenuti con entrambe le metodologie, e che le utilizzerebbero entrambe per la creazione degli esercizi per i propri alunni. Da un lato, CAT-AI per la sua semplicità e affidabilità nella creazione e personalizzazione degli esercizi, dall'altro lato, ChatGPT e Word per avere una maggiore varietà di contenuti e più libertà di modifica del testo.

Nome:	Cognome:	Data:
Esercizi di Matematica - Addizioni e Sottrazioni <i>Istruzioni: Risolvi i seguenti esercizi di addizione e sottrazione. Segui l'esempio per capire come fare.</i> Esercizio 1: Risolvi le seguenti Addizioni come mostrato nell'esempio. 0) Esempio: $12 + 7 =$ Risolvi: $12 + 7 = 19$ 1) $23 + 15 =$ 2) $34 + 22 =$ 3) $45 + 18 =$ Esercizio 2: Risolvi le seguenti Sottrazioni come mostrato nell'esempio. 0) Esempio: $20 - 5 =$ Risolvi: $20 - 5 = 15$ 1) $30 - 12 =$ 2) $50 - 25 =$ 3) $40 - 18 =$ Esercizio 3: Risolvi le seguenti Addizioni e Sottrazioni Insieme come mostrato nell'esempio. 0) Esempio: $15 + 5 - 3 =$ Risolvi: $15 + 5 = 20, 20 - 3 = 17$ 1) $25 + 10 - 5 =$ 2) $40 - 20 + 15 =$ 3) $50 + 5 - 10 =$		

Nome.....	Cognome.....	Classe.....	Data.....
Verifica di Matematica – Equazioni di primo grado Esercizio 1 Risolvi le seguenti equazioni e mostra i passaggi. a) $2x+5=11$ b) $7x-3=18$ c) $x/4+2=5$ Esercizio 2 Verifica se i seguenti valori sono soluzioni delle equazioni date. a) $x=3$ per l'equazione $4x-2=10$ b) $x=-2$ per l'equazione $3x+5=-1$ c) $x=0$ per l'equazione $2x+7=7$ Esercizio 3 Problemi con equazioni: scrivi e risolvi l'equazione corrispondente. a) Ho pensato a un numero, l'ho moltiplicato per 5 e poi ho aggiunto 7. Ho ottenuto 22. Qual è il numero? b) La somma tra il doppio di un numero e 8 è uguale a 20. Qual è il numero? c) Trovare il numero tale che, aumentato di 12, dia come risultato 25.			

a) CAT-AI

b) CHATGPT+WORD

Figura 5.2: Esempio Esercizi Creati Test Matematica 4

Capitolo 6

Conclusioni

Nel corso di questa tesi, è stato in primo luogo esposto l'obiettivo di creare un sistema per la creazione di esercizi intelligente che potesse aiutare gli insegnanti: CAT-AI. Infatti, a fronte di un crescente sviluppo tecnologico e dell'intelligenza artificiale, si è sviluppata la nuova possibilità di poter creare materiale didattico personalizzato per gli alunni, aiutando allo stesso tempo gli insegnanti. Sono state perciò indagate le necessità e le difficoltà degli insegnanti delle scuole elementari e medie nella creazione del loro materiale didattico.

Grazie a questa indagine, sono stati messi in luce fattori chiave nella creazione di esercizi, quali obiettivi di apprendimento, livello di difficoltà, varietà e affidabilità degli esercizi. A partire da ciò, sono stati redatti i requisiti funzionali e non funzionali dell'applicazione oggetto di sviluppo.

In seguito, è stata affrontata la fase di progettazione e creazione del prototipo tramite Figma, che ha permesso di delineare, a partire dai requisiti, le funzionalità principali e l'aspetto visivo e architeturale dell'applicazione. La fase di sviluppo ha invece visto lo sviluppo dell'applicazione in modo parallelo per l'interfaccia (Frontend) e per la logica e parte funzionale legata ad essa (Backend). L'intero sistema è stato sviluppato in un ambiente Javascript moderno, con l'utilizzo di React e Tailwind per il lato client, e di NodeJS e ExpressJS per lo sviluppo del lato server. Il lato intelligente dell'applicazione è stato introdotto tramite LLM, con l'API di OpenAI, che ha offerto la logica di generazione e personalizzazione tramite intelligenza artificiale. Una volta terminato lo sviluppo, l'ultima fase di analisi ha riguardato i test di usabilità.

In particolare, per verificare l'usabilità, l'efficienza e l'affidabilità effettiva dell'applicazione, è stato deciso di confrontare l'applicazione sviluppata con una baseline per la generazione e personalizzazione degli esercizi: ChatGPT e Word. Entrambe le metodologie hanno permesso agli insegnanti che hanno partecipato ai test di creare esercizi in meno di 10 minuti, allineandosi ai requisiti. Un fattore da evidenziare è che gli insegnanti hanno percepito e espresso una maggiore soddisfazione

generale con l'utilizzo di CAT-AI, grazie alla sua intuitività e semplicità offerte per la creazione di materiale didattico.

Infatti, l'applicazione sviluppata ha raggiunto un punteggio medio finale di 6,2 su 7, rispetto ai 5,3 su 7 della baseline nei questionari di misura dell'usabilità. Inoltre, CAT-AI ha rispettato tutti i requisiti funzionali prefissati in fase di progettazione. Parallelamente, per quanto riguarda i requisiti non funzionali e di qualità solo uno non è stato raggiunto, richiedendo un margine di miglioramento per quanto riguarda i tempi di generazione. Infine, le interviste hanno permesso di comprendere a fondo le motivazioni dei punteggi, oltre che i vantaggi e svantaggi introdotti dai due metodi. In conclusione, i test hanno permesso di delineare l'effettiva usabilità e efficienza dell'applicazione sviluppata, fornendo agli insegnanti una piattaforma unica e agevole per la creazione dei propri materiali didattici personalizzati.

6.1 Sviluppi Futuri

La fase di test ha anche permesso di analizzare le limitazioni dell'applicazione sviluppata, oltre a dare nuove idee per introdurre maggiore capacità di personalizzare e adattare i contenuti creati ai bisogni di ciascun insegnante. Tra gli sviluppi futuri devono sicuramente essere presi in considerazione gli svantaggi fatti emergere dagli insegnanti. Tra questi, le migliorie da apportare all'editor grafico, infatti, seppur esso offra un'impaginazione più semplice tramite la funzionalità drag&drop, è carente di funzionalità per personalizzare in modo più mirato il testo.

In seguito, sempre in relazione all'editor, sarebbe utile dare la possibilità di creare esercizi su più pagine, dando un controllo e libertà maggiore sulla creazione dell'esercizio. Un altro problema importante da risolvere, è il tempo di generazione, infatti, come analizzato, per la sola generazione e verifica iniziale, CAT-AI ha impiegato una media di 50 secondi. Questa attesa può creare frustrazione nell'utente e portarlo, nei casi peggiori, ad abbandonare l'applicazione. Per quanto riguarda invece la creatività che è mancata durante i test da parte del sistema, potrebbe essere risolto provando a migliorare ulteriormente le strategie di prompt engineering o semplicemente utilizzando modelli più intelligenti e creativi di GPT-4o, il quale ha una potenza di calcolo ridotta.

Infine, l'applicazione potrebbe anche aprire le porte a nuove strategie di adattamento del sistema intelligente all'utente, mantenendone, per esempio, una storia delle generazioni, oltre alle informazioni didattiche personali, in modo da creare esercizi sempre più accurati e specifici sia per l'insegnante che per gli alunni.

Bibliografia

- [1] Auste Simkute, Viktor Kewenig, Abigail Sellen, Sean Rintel e Lev Tankelevitch. *The New Calculator? Practices, Norms, and Implications of Generative AI in Higher Education*. 2025. arXiv: 2501.08864 [cs.HC]. URL: <https://arxiv.org/abs/2501.08864> (cit. a p. 1).
- [2] Michael Xieyang Liu, Advait Sarkar, Carina Negreanu, Benjamin Zorn, Jack Williams, Neil Toronto e Andrew D. Gordon. «“What It Wants Me To Say”: Bridging the Abstraction Gap Between End-User Programmers and Code-Generating Large Language Models». In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. CHI '23. Hamburg, Germany: Association for Computing Machinery, 2023. ISBN: 9781450394215. DOI: 10.1145/3544548.3580817. URL: <https://doi.org/10.1145/3544548.3580817> (cit. a p. 2).
- [3] Arthur C Graesser, Mark W Conley e Andrew Olney. «Intelligent tutoring systems.» In: (2012) (cit. a p. 3).
- [4] Chien-Chang Lin, Anna YQ Huang e Owen HT Lu. «Artificial intelligence in intelligent tutoring systems toward sustainable education: a systematic review». In: *Smart learning environments* 10.1 (2023), p. 41 (cit. a p. 3).
- [5] Alan Cooper, Robert Reimann, David Cronin e Chris Noessel. *About Face: The Essentials of Interaction Design*. John Wiley & Sons, Inc., 2014 (cit. a p. 11).
- [6] Nasrullah Hamidli. «Introduction to UI/UX design: key concepts and principles». In: *Preuzeto* 28.3 (2023), p. 204 (cit. alle pp. 18, 21).
- [7] Majeed Kazemitabaar, Jack Williams, Ian Drosos, Tovi Grossman, Austin Zachary Henley, Carina Negreanu e Advait Sarkar. «Improving Steering and Verification in AI-Assisted Data Analysis with Interactive Task Decomposition». In: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. UIST '24. ACM, ott. 2024, pp. 1–19. DOI: 10.1145/3654777.3676345. URL: <http://dx.doi.org/10.1145/3654777.3676345> (cit. a p. 18).

- [8] David Wolber, Yingfeng Su e Yih Tsung Chiang. «Designing dynamic web pages and persistence in the WYSIWYG interface». In: *Proceedings of the 7th international conference on Intelligent user interfaces*. 2002, pp. 228–229 (cit. a p. 20).
- [9] Jan Preßler, Lukas Schmid e Jörn Hurtienne. «Statistically Controlling for Processing Fluency Reduces the Aesthetic-Usability Effect». In: *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*. CHI EA '23. Hamburg, Germany: Association for Computing Machinery, 2023. ISBN: 9781450394222. DOI: 10.1145/3544549.3585739. URL: <https://doi.org/10.1145/3544549.3585739> (cit. a p. 21).
- [10] Mahsan Nourani, Amal Hashky e Eric D. Ragan. «User Profiling in Human-AI Design: An Empirical Case Study of Anchoring Bias, Individual Differences, and AI Attitudes». In: *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing* 12.1 (ott. 2024), pp. 137–146. DOI: 10.1609/hcomp.v12i1.31608. URL: <https://ojs.aaai.org/index.php/HCOMP/article/view/31608> (cit. a p. 62).