

Quizzy - Project plan

Università degli studi di Bergamo
Ingegneria Informatica

Arnoldi Elisa Matr. 1080572
Colpani Filippo Matr. 1078874
Foglieni Luca Matr. 1081399

Dicembre 2023



Indice

1	Introduzione	3
1.1	Contesto	3
1.2	Obiettivi	3
1.2.1	Creare e modificare quiz	3
1.2.2	Domande “vero o falso” ed a “scelta multipla”	3
1.2.3	Partecipare e giocare ad un quiz	3
1.2.4	Personalizzazione profilo utente	3
1.2.5	Clonare quiz esistenti	3
1.2.6	Aggiungere immagini alle domande del quiz	3
1.2.7	Numero di risposte variabili	3
1.2.8	Modalità di presentazione offline tramite app	3
2	Metodo di processo	4
3	Organizzazione del progetto	4
3.1	Ruolo e tipo di interazione con le Entità	4
3.2	Struttura del Team di Progetto (basato su XP e RUP)	4
4	Standard, linee guida, procedura	4
5	Attività di gestione	5
5.1	Incontri Settimanali per Aggiornamenti	5
5.2	Utilizzo di Git e Hosting su GitHub	5
6	Rischi	5
6.1	Problemi di compatibilità browser	5
6.2	Problemi di connessione	5
7	Personale	5
8	Metodi e tecniche	6
8.1	Controllo della Versione e Configurazione:	6
8.2	Documentazione Tecnica:	6
8.3	Ordine e Priorità dell’Implementazione:	6
8.4	Scrittura dei Test e Determinazione delle Milestone:	6
8.5	Ambiente di Prova e Apparecchiature:	6
8.6	Pianificazione dei Test e Procedure di Accettazione:	6
9	Garanzia di qualità	6
9.1	Correttezza	6
9.2	Affidabilità	7
9.3	Efficienza	7
9.4	Integrità	7
9.5	Usabilità	7
10	Risorse	7
10.1	Risorse Hardware e Strumenti:	7
10.2	Risorse Umane:	7
11	Cambiamenti	8
12	Consegna	8

1 Introduzione

1.1 Contesto

Quizzzy è una web app dedicata all'apprendimento tramite gioco, utilizzata a scopo didattico in un ambiente scolastico ed educativo. La scelta di tale tecnologia è dovuta alla sua portabilità così da risultare uno strumento efficace e in grado di soddisfare le necessità di studenti e docenti. Il suo funzionamento si basa sui quiz, ovvero una serie di domande decise dal docente che mirano a valutare le conoscenze degli studenti e allo stesso tempo svolgono il ruolo di un importante ripasso interattivo.

1.2 Obiettivi

1.2.1 Creare e modificare quiz

Questa funzionalità è esclusivamente riservata al docente, e consiste nell'inserire un numero a piacere di domande a cui gli studenti saranno poi sottoposti.

1.2.2 Domande “vero o falso” ed a “scelta multipla”

Le domande dovranno essere di tipo “vero o falso” o di tipo “scelta multipla”. Nel caso il professore inserisca delle domande a “scelta multipla”, dovrà anche inserire 4 risposte alternative, tra le quali è presente una o più risposte corrette, che gli studenti dovranno poi scegliere durante il quiz.

1.2.3 Partecipare e giocare ad un quiz

Partecipare e giocare ai quiz è una funzione esclusivamente riservata agli utenti, ovvero gli studenti. Gli utenti potranno, tramite un loro dispositivo personale, inserire il codice fornito dal professore per partecipare al quiz. I quiz consisteranno in una serie di domande impostate precedentemente dal professore. I giocatori (utenti) potranno rispondere alle domande del professore tramite il loro dispositivo, scegliendo una delle risposte precedentemente fornite dal professore.

1.2.4 Personalizzazione profilo utente

La personalizzazione del profilo utente consente agli studenti di selezionare un nome ed una immagine di profilo appena partecipano ad un nuovo quiz.

1.2.5 Clonare quiz esistenti

Clonare quiz esistenti consente ai docenti di clonare un quiz già esistente, precedentemente creato da loro o da un altro professore, per poi modificarlo nel caso lo si desideri.

1.2.6 Aggiungere immagini alle domande del quiz

Per arricchire l'esperienza è possibile aggiungere un'immagine per domanda. Questo permette al professore di fare una domanda specifica sulla domanda a cui gli studenti dovranno rispondere.

1.2.7 Numero di risposte variabili

A discrezione del docente, si possono inserire dalle 2 alle 4 risposte per ogni domanda.

1.2.8 Modalità di presentazione offline tramite app

I quiz sono scaricabili localmente dal professore tramite l'app Quizzzy per Windows, Linux MacOS, Android ed iOS. Nel caso in cui ci si trovi in una situazione in cui la connessione non è sufficiente per un'esperienza ottimale, il professore potrà presentare i quiz scaricati localmente nell'app in modo da poter comunque utilizzare le domande per una lezione interattiva con gli studenti.

2 Metodo di processo

Lo sviluppo del progetto seguirà una combinazione di Extreme Programming (XP) e Rational Unified Process (RUP). L'approccio XP garantirà una flessibilità e una risposta rapida ai cambiamenti nei requisiti del cliente attraverso pratiche come lo sviluppo iterativo, la programmazione a coppie e il test-driven development (TDD). Allo stesso tempo, l'implementazione in stile RUP fornirà una struttura ben definita, con fasi chiaramente delineate come avvio, elaborazione, costruzione e transizione. Questa combinazione mira a sfruttare la rapidità e l'adattabilità di XP durante lo sviluppo, integrandole con l'approccio strutturato e guidato dai casi d'uso di RUP nella fase di implementazione. Ciò consentirà di massimizzare la gestione dei rischi, la chiarezza dei requisiti e la qualità complessiva del prodotto, mantenendo al contempo un ritmo di lavoro sostenibile e rispondendo prontamente ai feedback del cliente.

3 Organizzazione del progetto

Il progetto coinvolge un team di sviluppo composto da tre membri che adotteranno una combinazione di Extreme Programming (XP) e Rational Unified Process (RUP) per realizzare un'applicazione di quiz mirata a facilitare l'insegnamento dei docenti e migliorare l'apprendimento degli studenti. L'organizzazione del progetto sarà strutturata per massimizzare l'efficienza, mantenendo al contempo una collaborazione stretta con i docenti e garantendo una user experience ottimale per gli studenti.

3.1 Ruolo e tipo di interazione con le Entità

Docenti: Interazione costante per ottenere informazioni dettagliate su come progettare i quiz, i metodi di valutazione dei quiz e le risorse educative.

Studenti: Coinvolgimento regolare per ottenere valutazioni sull'accessibilità e l'intuitività nell'utilizzo dell'applicazione e feedback sulle prestazioni.

3.2 Struttura del Team di Progetto (basato su XP e RUP)

Analisti (RUP): Analisi dei requisiti iniziali, interazione con docenti e studenti per comprendere le esigenze e definire la visione del prodotto.

Sviluppatori (XP): Implementazione iterativa delle funzionalità, programmazione a coppie e adozione del test-driven development (TDD).

Testers (XP): Esecuzione di test continui per garantire la qualità del software durante tutte le fasi del ciclo di sviluppo.

Questa organizzazione è progettata per sfruttare i benefici di XP e RUP, fornendo un'applicazione che si adatta rapidamente alle esigenze dei docenti, mantiene standard qualitativi elevati e coinvolge attivamente gli studenti nel loro percorso di apprendimento.

4 Standard, linee guida, procedura

Gli standard Java Oracle definiranno le regole e le pratiche per la scrittura del codice, assicurando che il nostro software sia conforme alle migliori pratiche dell'industria, così che il nostro approccio allo sviluppo possa garantire uniformità e qualità. La documentazione seguirà gli standard UML (Unified Modeling Language) per garantire una rappresentazione chiara e standardizzata del progetto. Utilizzeremo diagrammi UML per modellare requisiti, progettazione e strutture di dati, consentendo una comprensione accurata e coesa del sistema. Quando il software raggiungerà la prima release, avverrà anche la pubblicazione della documentazione. Per garantire il mantenimento di una documentazione sempre aggiornata, per ogni nuova versione del software verrà fornita una nuova documentazione che contenga eventuali nuove specifiche. All'interno delle linee guida per l'organizzazione del progetto, si stabilisce che l'implementazione completa del sistema dovrà essere realizzata esclusivamente all'interno

del IDE di Eclipse, senza fare ricorso a servizi esterni. Questa direttiva è stata introdotta con l'obiettivo di garantire uniformità nell'ambiente di sviluppo e di semplificare l'integrazione del codice all'interno del contesto dell'IDE di riferimento. Limitare l'utilizzo di servizi esterni contribuirà a semplificare la gestione del progetto e a mantenere il pieno controllo sull'intero processo di sviluppo.

5 Attività di gestione

La pianificazione e la supervisione delle attività sono strettamente guidate da obiettivi e priorità stabilite per garantire il successo complessivo del progetto. Le seguenti attività di gestione sono progettate per mantenere il controllo sullo sviluppo e garantire il rispetto degli obiettivi prefissati nei tempi limite.

5.1 Incontri Settimanali per Aggiornamenti

Gli incontri periodici settimanali fungono da piattaforma chiave per presentare relazioni periodiche sullo stato del progetto. Durante questi incontri, verranno discussi gli avanzamenti, i risultati raggiunti e le sfide attuali. La direzione utilizzerà questi aggiornamenti per valutare il progresso complessivo e prendere decisioni informate in merito alle priorità e agli obiettivi successivi.

5.2 Utilizzo di Git e Hosting su GitHub

L'adozione di sistemi di controllo di versione come Git e l'hosting delle repository su GitHub rientrano nella strategia di gestione per garantire tracciabilità e controllo sullo sviluppo del software. La direzione può monitorare le modifiche al codice, garantendo l'integrità del progetto e facilitando la revisione delle attività svolte dai membri del team. Queste attività di gestione sono integrate con la presentazione di relazioni periodiche, fornendo un quadro completo dello stato del progetto. La direzione, guidata da obiettivi ben definiti, utilizzerà queste informazioni per bilanciare requisiti, tempi e costi, assicurando un allineamento costante con gli obiettivi del progetto e una risposta tempestiva alle sfide emergenti.

6 Rischi

6.1 Problemi di compatibilità browser

Potrebbero verificarsi problemi di compatibilità del codice con determinati browser, generando dei problemi alla partecipazione e all'esperienza degli utenti. Per evitarlo si devono condurre dei test di compatibilità con i browser più noti per garantire la loro funzione. Notare che futuri aggiornamenti dei vari browser potrebbero creare problemi di compatibilità, quindi sarà necessario un costante testing per mantenere il corretto funzionamento.

6.2 Problemi di connessione

L'esperienza utente potrebbe essere compromessa dalla qualità della connessione, in quanto l'essenza di Quizzy è la velocità di risposta, le cause principali sarebbero dovute a problemi con l'access point locale. Per un'esperienza migliore verificare la connessione prima dell'effettivo utilizzo, in una futura release si implementerà una versione offline, dove il docente potrà inserire le risposte offline dei vari utenti.

7 Personale

Non è richiesto l'intervento di personale esterno al team di sviluppo.

8 Metodi e tecniche

In questa sezione, vengono delineati i metodi e le tecniche che saranno impiegati durante le diverse fasi del ciclo di sviluppo del progetto, compresi l'ingegneria dei requisiti, la progettazione, l'implementazione e le prove.

8.1 Controllo della Versione e Configurazione:

Per gestire il controllo della versione del software, adotteremo il sistema di gestione delle versioni Git, con le repository ospitate su Github. Questa scelta ci consentirà di mantenere una tracciabilità chiara delle modifiche e di facilitare la collaborazione tra i membri del team.

8.2 Documentazione Tecnica:

La documentazione tecnica, essenziale per ciascuna fase del progetto, sarà prodotta secondo gli standard di JavaDoc. Questo garantirà coerenza e chiarezza nella documentazione generata, consentendo una comprensione approfondita del codice sorgente e delle funzionalità implementate.

8.3 Ordine e Priorità dell'Implementazione:

Dopo la definizione dell'architettura, stabiliremo un chiaro ordine e priorità per l'implementazione dei componenti. Questa sequenza sarà indicata esplicitamente, permettendo una progressione strutturata e coerente nello sviluppo dell'applicazione.

8.4 Scrittura dei Test e Determinazione delle Milestone:

Saranno scritti test di unità utilizzando JUnit per determinare le milestone del progetto. Questa pratica ci consentirà di valutare il corretto funzionamento dei singoli componenti e di raggiungere obiettivi chiave durante lo sviluppo.

8.5 Ambiente di Prova e Apparecchiature:

Gli ambienti di prova saranno configurati su una rete locale. Si presta particolare attenzione alla definizione dell'ambiente di prova e delle apparecchiature necessarie, poiché la corretta esecuzione dei test influirà notevolmente sulla qualità del prodotto finale.

8.6 Pianificazione dei Test e Procedure di Accettazione:

L'ordine di integrazione e test sarà chiaramente indicato, con una pianificazione attenta per garantire che ogni componente sia adeguatamente valutato. Le procedure di test di accettazione, supervisionate dall'utente, saranno dettagliatamente descritte per assicurare una valutazione completa e soddisfacente delle funzionalità implementate.

9 Garanzia di qualità

Al fine di assicurare la qualità durante il processo di sviluppo e la realizzazione del software richiesto, saranno presi in considerazione i seguenti 5 criteri chiave:

9.1 Correttezza

La correttezza del software sarà garantita attraverso un approccio di sviluppo basato su continui test di utilizzo. Questa metodologia assicurerà un prodotto accurato e in grado di soddisfare gli obiettivi definiti nelle diverse milestones del progetto.

9.2 Affidabilità

Il sistema progettato sarà orientato a garantire lunghi periodi di utilizzo e una reattività eccezionale. Questi requisiti guideranno sia lo sviluppo che la progettazione, ponendo particolare attenzione alla stabilità e alla robustezza del sistema.

9.3 Efficienza

La massimizzazione dell'efficienza sarà una priorità, anche se la strategia specifica sarà definita in fasi successive del progetto. L'obiettivo sarà ottimizzare le risorse per garantire un funzionamento fluido e performante dell'applicazione.

9.4 Integrità

Per garantire un elevato livello di protezione e integrità dei dati, l'architettura del sistema sarà progettata con layer isolati, consentendo scambi di messaggi strutturati. Questo approccio contribuirà a preservare l'integrità dei dati e a garantire una maggiore sicurezza e privacy.

9.5 Usabilità

Le interfacce utente saranno progettate focalizzandosi sull'usabilità. Questo obiettivo sarà raggiunto attraverso un design minimal e responsive, facilitando l'interazione degli utenti con l'applicazione di quiz e migliorando complessivamente l'esperienza d'uso.

10 Risorse

All'interno del nostro progetto la realizzazione dell'applicazione di quiz per docenti richiede un'attenta considerazione di diverse risorse suddivise in due categorie principali: risorse hardware e strumenti, e risorse umane.

10.1 Risorse Hardware e Strumenti:

Server Web: Un server web dedicato sarà fondamentale per gestire l'interfaccia dell'applicazione, garantendo un'esperienza utente intuitiva, responsiva e cross-platform.

Server per la Gestione in Tempo Reale: Un secondo server sarà allocato per la gestione in tempo reale delle partite, assicurando un'esperienza di gioco fluida.

Database Embedded (SQLite): Sarà necessario integrare un database embedded, come SQLite, per gestire in modo efficiente e locale i dati dell'applicazione, migliorando la reattività del sistema.

Hardware: Dispositivi per lo sviluppo saranno essenziali per supportare le attività di programmazione e di test.

Strumenti: L'utilizzo di strumenti come IDE (Integrated Development Environment), sistemi di controllo di versione come Git, e piattaforme di hosting come GitHub contribuiranno alla collaborazione e alla gestione efficace del codice sorgente.

10.2 Risorse Umane:

Il team di tre persone, seguendo gli approcci XP e RUP, si dividerà tra sviluppatori, analisti e tester, contribuendo in modo sinergico all'intero processo di sviluppo. L'aggiunta di un database embedded come SQLite si integra perfettamente con l'architettura, contribuendo a garantire un'efficiente gestione dei dati. La gestione agilità del team, unita a risorse hardware e strumenti adeguati, consentirà al progetto di progredire in modo dinamico e coerente secondo gli standard definiti.

11 Cambiamenti

La gestione dei cambiamenti terrà conto dell'approccio di sviluppo basato sui principi di Extreme Programming (XP) e Rational Unified Process (RUP).

Ad ogni fase del processo di sviluppo, ci aspettiamo un'evoluzione del software integrando nuove funzionalità e correggendo eventuali errori progettuali rilevati. Questo approccio mira a mantenere un alto grado di flessibilità, consentendo uno sviluppo rapido e la pronta risposta ai feedback degli utenti. Considerando l'inevitabilità dei cambiamenti, implementeremo procedure chiare per gestire le modifiche proposte.

Nell'ambito di una metodologia agile, dove ogni iterazione introduce cambiamenti gestiti in modo leggero, queste modifiche sono considerate come parte integrante del processo e non come astratti cambiamenti.

In risposta a cambiamenti strutturali, ogni proposta di modifica sarà attentamente registrata e sottoposta a revisione. L'approvazione di una richiesta di modifica comporterà la valutazione dell'impatto, inclusi costi e tempi associati, seguita dall'introduzione della modifica nel progetto secondo procedure ben definite.

Considerando il contesto specifico dell'applicazione di quiz, la gestione dei cambiamenti sarà guidata dalla necessità di migliorare l'esperienza degli utenti, integrando nuove funzionalità e garantendo la stabilità del sistema durante il processo evolutivo.

12 Consegna

Una volta raggiunta la release, il codice sorgente e la documentazione completi saranno forniti al committente.