

Università degli studi di Napoli Parthenope



Myst - RDD **Documento di Progettazione**

Progetto di Ingegneria del Software e
Interazione Uomo-Macchina

A.A 2023/2024

Candidati:

Acampora Simone

0124002485

Arcopinto Lorenzo

0124002626

Gaudino Daniele

0124002544

Sommario

1 - Introduzione	2
1.1 - Scopo del sistema	2
1.2 - Obiettivi di progettazione	3
1.3 - Riferimenti	5
1.4 - Panoramica	5
2 - Sistema corrente	6
3 - Sistema proposto	7
3.1 - Decomposizione del sistema	8
3.2 - Hardware/Software mapping	10
3.3 - Gestione dati persistenti	10
3.4 - Controllo accessi e sicurezza	10

1 - Introduzione

1.1 - Scopo del sistema

Myst è una piattaforma online per i videogiocatori per permettere loro l'acquisto di videogiochi digitali.

L'obiettivo principale sul quale ci siamo focalizzati è quello di permettere agli utenti, dopo aver creato un account, di acquistare i giochi pubblicati dai publisher sulla piattaforma sia in modo "nuovo" al prezzo accordato dal publisher, sia in modo "usato", acquistando la licenza del gioco digitale direttamente da un altro utente a un prezzo abbassato.

La scelta di poter permettere all'utente di poter vendere i propri giochi in modo "usato" rappresenta la differenza rispetto alle altre piattaforme online per la vendita di videogiochi, dove ciò non è possibile. Questa scelta è stata fatta pensando anche al problema dell'usato digitale che è praticamente inesistente al giorno d'oggi nel mercato dei videogiochi, dove l'unico modo per poter risparmiare sui giochi oltre agli sconti sono i siti di terze parti che vendono chiavi di attivazione digitali che rappresentano un mercato grigio per l'acquisto di videogiochi (sono stati documentati molto casi di siti di chiavi che vendevano chiavi ottenute con carte di credito rubate, e non si può avere una conferma al momento del pagamento che queste non siano state ottenute in questo modo).

1.2 - Obiettivi di progettazione

I diversi obiettivi di progettazione sono stati divisi in punti in modo da dare chiarezza su ciò che il sistema finale vuole rappresentare.

- **Usabilità**

Per garantire un apprendimento facile ed immediato della piattaforma si è prestata particolare attenzione a mantenere un design semplice ed intuitivo, con l'utilizzo di immagini e simboli che possano risultare familiari alla gran parte del target.

- **Lock and feel**

L'interfaccia è stata realizzata in modo da essere immediata e responsive, così da adattarsi al meglio alle esigenze dell'utente finale.

- **Response time**

Tutte le funzionalità presenti nel sistema sono state schematizzate in poche azioni e veloci in modo tale che l'utente possa rimanere soddisfatto del proprio utilizzo e da evitare casi in cui ci possa essere uno sconforto da parte sua.

- **Estensibilità**

Le diverse componenti del sistema sono state realizzate in modo tale da poter permettere eventuali modifiche e/o ampliamenti senza dover andare a modificare l'intero sistema, ma solo la parte interessata.

- **Piattaforma di destinazione**

Il sistema è stato sviluppato utilizzando Python 3 e Flask per poter essere considerato come una web application ed essere accessibile da qualsiasi browser scelto dall'utente per accedervi.

- **Portabilità**

Trattandosi di una piattaforma accessibile tramite internet, questa è stata resa responsive in modo da poter essere accessibile da qualunque dispositivo.

- **Sicurezza dei dati**

Gli unici dati sensibili inseriti dagli utenti sono le credenziali di accesso, ed eventualmente altri dati che l'utente sceglie di inserire volontariamente. E-mail e password saranno protetti tramite crittografia. Per i pagamenti la sicurezza dei dati è garantita da PayPal.

- **Gestione dei dati persistenti**

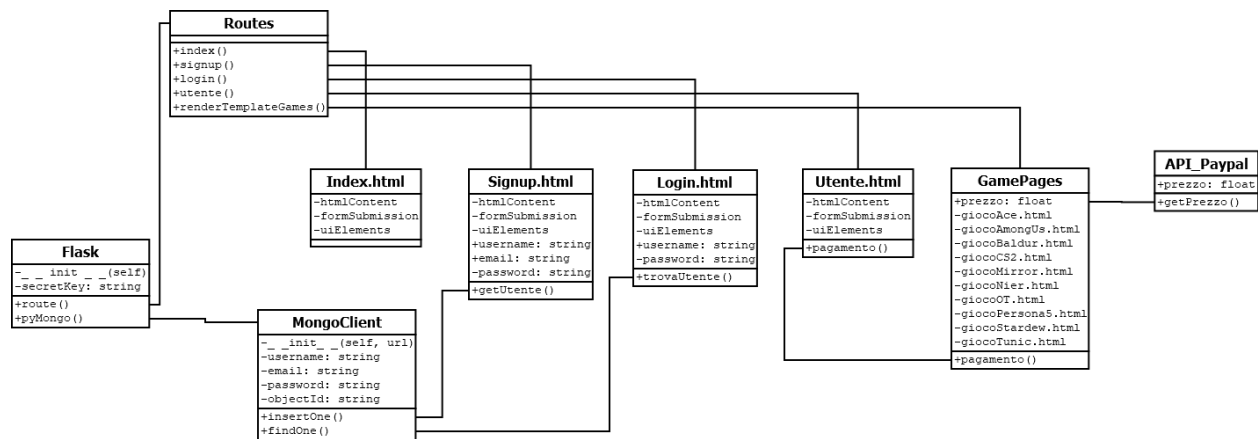
I dati verranno trattati mediante l'utilizzo di un database MongoDB e solo l'amministratore, avrà accesso a quest'ultimo.

1.3 - Riferimenti

Per maggiori chiarimenti su alcuni dei termini utilizzati si prega di consultare il glossario che si trova come ultima sezione del documento RAD di Analisi dei requisiti del sistema, presente insieme alle immagini utilizzare in entrambe le documentazioni.

1.4 - Panoramica

In questa sezione si riporta il diagramma delle classi, anche in questo caso per gli stessi motivi spiegati nel file RAD il diagramma non è sviluppato in maniera dettagliata.



(Per visualizzarlo meglio consigliamo di vederlo nella cartella immagini allegate con il documento)

2 - Sistema corrente

Come spiegato nel documento RAD, il nostro team ha notato che nel mercato PC videoludico sorge un problema a causa della mancanza della possibilità per la vendita dei propri giochi in digitale, al contrario della controparte fisica che è possibile rivendere tranquillamente senza problemi.

Siamo portati a credere che i leader del settore, come Steam o Epic Games, non possano mai fallire e che realizzare una nuova piattaforma sia uno sforzo futile, poiché verrà oscurata dalla concorrenza. Ma non per forza è così.

Nonostante la massiccia presenza di questi leader del settore, vediamo che nessuno si è concentrato sul provare a dare la possibilità agli utenti di poter rivendere i loro giochi, e per una persona che non può permettersi di comprare giochi a prezzo pieno sempre questo può risultare frustrante.

Per questo Myst, a differenza delle altre piattaforme, vuole dare una possibilità a tutti di divertirsi giocando ai loro giochi preferiti, dando loro quindi un motivo per iscriversi ed utilizzare la piattaforma.

Noi crediamo che, in un mercato dove tutti cercano l'attenzione degli utenti per sovrastare la concorrenza, fornire la possibilità di rivendita dei propri giochi e l'acquisto di giochi digitali "usati" possa portare una ventata fresca in un mercato di prodotti sempre più simili.

3 - Sistema proposto

Per la progettazione di Myst, volendo essere un web application, si è scelto di utilizzare un'architettura client server.

Nell'architettura del sistema, come scaturito dall'analisi dei requisiti, distinguiamo in 3 principali categorie di utenti:

- Il superutente, o amministratore, che può usare il sistema sia come utente sia accedendovi tramite il pannello di amministrazione e quindi andare a modificare qualsiasi contenuto al suo interno.
- L'utente, che non possiede i permessi dell'amministratore, ed ha la possibilità di creare un profilo, di comprare giochi e rivendere i propri giochi.
- Il publisher, cioè l'utente che pubblica i giochi sulla piattaforma, creando e modificando le pagine dei giochi.

Oltre all'aspetto legato agli utenti, si è curato anche l'aspetto riguardante la memorizzazione dei dati, tramite un apposito database MongoDB che non può essere accessibile tramite l'interfaccia visibile agli utenti, e l'aspetto della portabilità, resa possibile poiché il sistema è stato implementato come web application.

3.1 - Decomposizione del sistema

Il sistema può essere suddiviso in diverse sottosezioni, in base alla funzione alla sezione a cui sono legate.

La prima divisione possiamo farla tra la parte utente e la parte funzionale. Al loro interno troviamo delle macrosezioni.

Per quanto riguarda la parte dell'utente:

- Signup management, che si occupa di tutto ciò che è compreso nella gestione della registrazioni degli utenti la prima volta che effettuiamo l'accesso al sistema.
- Login management, che si occupa della parte relativa al login degli utenti, quindi all'interazione dell'utente con il sistema in fase di accesso.
- Profile management, per tutto ciò che riguarda la gestione, la visualizzazione e la modifica del profilo degli utenti.

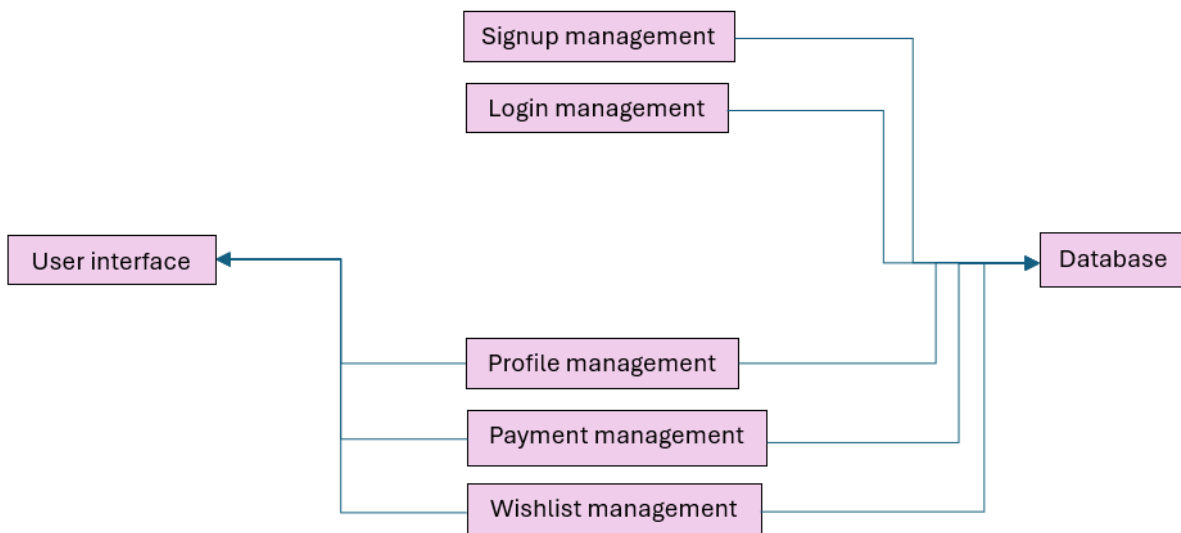
- User interface, tutto ciò che visualizza l'utente e con cui può interagire.

Per quanto riguarda la parte funzionale:

- Payment management, per la gestione dei giochi acquistati dagli utenti.
- Wishlist management, per la gestione delle liste dei desideri dei giochi voluti dagli utenti

Il database invece è una sezione comune a tutto il sistema in quanto è il modo in cui vengono immagazzinati e gestiti tutti i dati.

Se vogliamo visualizzare le sezioni graficamente possiamo farlo così:



3.2 - Hardware/Software mapping

Il sistema è stato progettato avendo in mente la portabilità come uno dei requisiti fondamentali per l'usabilità. Detto ciò, abbiamo implementato il sistema come un client-server, in modo tale da poter essere una web application e accessibile su qualunque dispositivo disponga di un browser di ricerca che si connette ad internet.

3.3 - Gestione dati persistenti

Per quanto riguarda la gestione dei dati, Myst utilizza un database MongoDB per memorizzare tutti i dati relativi agli utenti, oltre che altre informazioni sui log del sistema, migrazione di cambiamenti e dati legati alle funzionalità interne.

I dati visibili sulla piattaforma accessibile dall'utente non sono direttamente richiesti da questi ultimi, e quindi risultano essere protetti in quanto ci sono pochi casi dove è possibile fare richieste che si traducono in query per il database.

3.4 - Controllo accessi e sicurezza

Poiché il sistema si suddivide in diverse sottosezioni, bisogna distinguere quali utenti possono accedervi e quali no.

L'amministratore del sistema ha accesso a tutte le sottosezioni in quanto ha accesso al pannello di controllo, oltre che a tutti i file di implementazione del sistema.

Gli altri utenti, invece, possono accedere solo alle funzioni presenti nelle sezioni strettamente legate alle interfacce del sistema, avendo quindi parzialmente accesso a quelle sezioni.

Ad esempio, tutti gli utenti possono effettuare l'accesso al sistema; posso quindi accedere alle sezioni di gestione del login e della registrazione, ma in modo limitato in quanto possono solo utilizzare le funzioni per il login e per la registrazione, mentre l'amministratore ha accesso anche ai file in cui sono definite tali funzioni e le può modificare.