

**Documentazione System Design**

**“GLITCH”**

**Progetto presentato da:**

Annunziata Elefante

Ferdinando Napolitano

Santolo Mutone

**INDICE**

1. **Introduzione . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .** 
   1. Scopo del sistema . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   2. Obiettivi di progettazione . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   3. Definizioni, acronimi e abbreviazioni . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   4. Panoramica . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
2. **Architettura software corrente . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .**
3. **Architettura software proposta . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .** 
   1. Panoramica . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   2. Decomposizione del sottosistema . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   3. Mappatura hardware/software . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   4. Gestione dei dati persistenti . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   5. Controllo degli accessi e sicurezza . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   6. Controllo globale del software . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   7. Condizioni di confine . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
4. **Servizi del sottosistema** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
5. **Glossario** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
6. **Introduzione**
   1. **Scopo del sistema**

Il progetto Glitch è stato commissionato con le specifiche di permettere a tutti gli utenti registrati di poter acquistare, comodamente da casa, tutti i videogiochi e le console che desiderano.

Il sistema proposto andrà a velocizzare la vendita dei prodotti sopra citati, permettendo il passaggio da un negozio fisico, in cui il cliente deve recarsi senza avere la certezza di trovare ciò di cui necessita, a un e-commerce ricco di prodotti e di facile utilizzo. Ne segue che il software sarà rivolto a tutti gli appassionati di “gaming”, ai quali sarà consentito di inserire i prodotti scelti in un apposito carrello personale e acquistarli comodamente dal proprio divano.

Tenendo presente le specifiche, Glitch dovrà permettere di filtrare i prodotti mediante appositi filtri, consentirne l’acquisto o la rimozione dal carrello dei prodotti, permettere un servizio di assistenza, gestire il catalogo.

* 1. **Obiettivi di progettazione**

Glitch è stato sviluppato per semplificare le procedure di visualizzazione e vendita di console e videogiochi. Ciò grazie ad interfacce intuitive che consentono di interagire e gestire il sistema.

Il sistema si appoggia su un DBMS esterno, sfruttandone al massimo le potenzialità di ottimizzazione per il recupero dei dati. La documentazione gestita da Glitch è a carattere riservato e ad uso esclusivo dell’utente, pertanto si è scelto di adottare un sistema di autenticazione che controlla l’accesso al sistema tramite l’inserimento di un username e una password assegnati agli utenti al momento della registrazione.

Il linguaggio di programmazione scelto per il suo sviluppo è Java, con paradigma di programmazione ad oggetti.

* 1. **Definizioni, acronimi e abbreviazioni**
* **SDD**: System Design Document
* **DBMS**: DataBase Management System
* **DB**: DataBase
  1. **Panoramica**

Il seguente documento SDD è diviso in sezioni ed ha la seguente composizione:

* Sezione di *INTRODUZIONE*: vi è presente una descrizione dell’esigenza di progettare il sistema nel miglior modo per poter facilitare le operazioni degli utenti, seguita dal contesto di utilizzo del sistema per poi presentarne gli obiettivi principali. Importante è anche la presenza di un elenco di definizioni, acronimi e abbreviazioni usato per facilitare la comprensione dei concetti citati al lettore.
* Sezione sull’ *ARCHITETTURA SOFTWARE CORRENTE*: mostra le tecnologie e le idee di progettazione prese in considerazione per lo sviluppo di Glitch.
* Sezione sull’ *ARCHITETTURA SOFTWARE PROPOSTA*: in primo luogo si mostra una panoramica sull’idea di base di come il sistema dovrebbe essere progettato. Ne segue la presentazione dei vari sottosistemi in cui il sistema è stato decomposto. Si continua con la descrizione del mapping effettuato sia sulle componenti hardware sia sulle componenti software, seguito dal modo in cui sono stati gestiti i dati persistenti. Le altre due sessioni successive spiegano il controllo degli accessi e della sicurezza, nonché quello globale del software. Infine, vi è una presentazione delle condizioni di confine esterne al sistema.
* Sezione sui *SERVIZI DEL SOTTOSISTEMA*: descrive i servizi forniti da ciascun sistema in termini di operazioni.
* Sezione di *GLOSSARIO*: in conclusione del SDD per specificare i termini utilizzati nel documento così da evitare ambiguità.

1. **Architettura software corrente**

Nella progettazione del sito i progettisti non si nono potuti avvalere di un sistema preesistente in quanto il cliente non disponeva di alcun sistema di gestione e vendita prodotti on-line.

Quindi si è proceduto con una implementazione di tipo “Greenfield Engineering”, ovvero un’implementazione da zero cercando di soddisfare le esigenze del cliente e, prendendo spunto da siti simili alcune meccaniche da poter implementare e migliorare secondo i requisiti.

Esempi di siti considerati si possono individuare ai seguenti link: [www.amazon.it](http://www.amazon.it), esempio di sito di e-commerce generico, e [www.gamestop.it](http://www.gamestop.it) , come sito di vendita console e videogiochi.

1. **Architettura software proposta**
   1. **.** **Panoramica**

Il Sistema commissionato è un software distribuito costituito da un nucleo centrale, contenente tutti i dati relativi agli utenti registrati, prodotti che il sito dispone, tiene traccia delle mail di assistenza, delle offerte e del carrello utente con i relativi ordini effettuati.

Gli utenti interagiscono con il sistema centrale da diverse postazioni client.

I client non hanno alcuna possibilità di interagire tra di loro pertanto, tutte le richieste dovranno sempre essere fatte al server centrale. Ne deriva che l’architettura più idonea al progetto è di tipo **Client-Server**.

Le tipologie di utenti si dividono in: utente, gestore account, gestore catalogo e gestore assistenza.

* Il primo è l’utente classico, che si registra al sito e può effettuare vari acquisti;
* Il secondo è il gestore account che ha il compito di definire i ruoli di alcuni utenti rendendoli gestori del sito (del catalogo, account o assistenza);
* Il terzo è il gestore catalogo che è colui che si occupa di inserire/rimuovere prodotti e offerte in modo da tenere il catalogo sempre aggiornato;
* Il quarto è il gestore assistenza che si occupa di rispondere alle e-mail di assistenza degli utenti in modo tale da poter aiutare e risolvere i vari problemi che gli utenti possono riscontrare.

* 1. **. Decomposizione del sottosistema**