Università degli Studi di Salerno  
 Corso di Ingegneria del Software  
A.A. 2018/2019



Problem statement  
Versione 1.0.0

|  |
| --- |
| COORDINATORE DEL PROGETTO: |
| Prof.re Andrea De Lucia |

|  |  |
| --- | --- |
| PARTECIPANTI AL PROGETTO: | |
| Anagrafica | Matricola |
| Cuomo Enrico | 0512103590 |

|  |
| --- |
| SCRITTO DA: |
| Cuomo Enrico |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
| REVISION HISTORY: | |  |  |
| Data | Versione | Descrizione | Autore |
| 20/11/18 1.0.0 | | Problem statement | Enrico Cuomo |
| 0510200130 | |  |  |

Sommario

[1. Introduzione 3](#_Toc379469311)

[1.1. Descrizione breve 3](#_Toc379469312)

[1.2. Obiettivi 3](#_Toc379469313)

[2. Current Enviroment 4](#_Toc379469316)

[2.1. Scenario 4](#_Toc379469312)

[3. Proposed Enviroment 5](#_Toc379469317)

[3.1. Scenari principali 5](#_Toc379469312)

[3.2. Functional requir 5](#_Toc379469318)

[3.2.1. Registrazione 5](#_Toc379469319)

[3.2.2. Login 5](#_Toc379469320)

[3.2.3. Logout 5](#_Toc379469321)

[3.2.4. acquisto gioco 5](#_Toc379469323)

[3.2.5. Inserimento gioco 6](#_Toc379469323)

[3.3. Not functional requirements 6](#_Toc379469327)

[3.4.1. Usability 6](#_Toc379469328)

[3.4.2. Reliability 6](#_Toc379469329)

[3.4.3. Performance 6](#_Toc379469330)

[3.4.4. Supportability 6](#_Toc379469331)

[4. Deliverables 42](#_Toc379469343)

# Introduzione

## Descrizione breve

Lo scopo del Sistema ideato è quello di realizzare un applicativo web che permetta la vendita o la prenotazione di video-giochi on-line multipiattaforma .  
Tale sistema permetterà ad un utente generico di accedervi come “amministratore” o come “utente registrato”.   
Un utente registrato, accedendo alla propria area utente, potrà gestire i pagamenti dei giochi o disdire una prenotazione; potrà inoltre visualizzare le informazioni dei giochi che sono presenti nel suo carrello o fare il log out.  
Un amministratore, invece, entrando nella propria area amministrativa potrà inserire un’offerta o news (un nuovo gioco in uscita a breve e già prenotabile sul web (poiché è stata fatta una prenotazione dal manager del negozio e si attendono le scorte).

## Obiettivi

Il sistema in sviluppo proposto si pone come obbiettivi l’efficienza e un facile utilizzo da parte dell’utente finale. Di fatti si vuole generare un sistema che sia il più user-friendly possibile. Il nostro ipotetico cliente finale (colui che utilizza il sito) dovrà arrivare all’acquisto di un gioco desiderato in pochi click.

# Current Enviroment

## Scenario

Di seguito è riportato uno scenario di un sito e-commerce già esistente: Amazon

|  |  |
| --- | --- |
| **Nome Scenario** | *Utente si registra al sito* |
| **Attori partecipanti** | *Tommaso: utente che gira il web in cerca di giochi* |
| **Flusso degli eventi** | 1. Tommaso sta ricercando nuovi giochi per la sua ps4 ed entra su Amazon 2. Tommaso non ha idea su cosa fare, siccome tra i risultati sulla pagina corrente (di immediato accesso) non ci sono giochi. Allora nota la barra di ricerca in alto e scrive il nome di un gioco che potrebbe interessarlo: Destiny 2 3. Tommaso preme invio e nota che il gioco è scontato ed è possibile acquistarlo a soli 9,99 euro 4. Tommaso aggiunge il prodotto al carrello 5. Tommaso clicca su “procedi all’ordine” e si rende conto di non essere ancora registrato al sito e che deve obbligatoriamente svolgere questa pratica per acquistare il prodotto da lui desiderato, poiché il sistema apre una pagina dedicata allo svolgimento di una tale pratica 6. Tommaso allora si registra inserendo: e-mail, codice fiscale, nome, cognome, città, via/ quartiere e password. 7. L’utente clicca allora sul tasto prosegui, il quale gli consentirà di completare l’ordine 8. Tommaso può decidere di pagare immettendo i dati relativi ad una carta di credito o pagare alla consegna. 9. Tommaso decide di pagare alla consegna e conclude la pratica |

## Proposed Enviroment

## Scenari principali

|  |  |
| --- | --- |
| **Nome Scenario** | *Utente effettua acquisti* |
| **Attori partecipanti** | *Tommaso Da Lampedusa: utente registrato* |
| **Flusso degli eventi** | * Tommaso da Lampedusa trova un gioco di cui non aveva mai sentito parlare prima: Binary Domain e decide di acquistarlo * Tommaso aggiunge al carrello il gioco che desidera * Il nostro utente decide di ultimare gli acquisti. Entra nella pagina di “Conferma degli acquisti” in cui deve scegliere il metodo di pagamento con cui ultimare gli acquisti (Carta di credito o Paypal) * Dopo aver inserito il codice della carta di credito o della paypal, Tommaso visualizzerà una scritta “Ordine effettuato con successo” |

## Functional requir

### Registrazione

Il sistema deve prevedere una form per la registrazione dell’utente in modo da potergli concedere tutti i privilegi di un utente registrato.

Quest’operazione deve essere svolta in modo da evitare che due utenti si registrino con le stesse credenziali o che un utente si registri non inserendo tutti i campi obbligatori (nome, cognome, password, username, anni)

### Login

Il sistema garantirà che un utente non registrato non acceda al sito web per questioni di sicurezza. Nessun utente può eseguire un login se prima non è stato registrato

### Logout

Il sistema sarà efficiente nel senso che una volta registrato l’utente può eseguire tranquillamente il logout, senza preoccuparsi dei suoi dati i quali non verranno persi ma conservati in una struttura apposita in modo da permettere un eventuale altro accesso

### Acquisto gioco

Il sistema web in sviluppo deve impedire che due utenti acquistino o prenotino lo stesso gioco se le scorte sono esaurite in magazzino. (ad esempio se rimane un solo prodotto)

### Inserimento gioco

Il sistema deve impedire che venga inserito due volte lo stesso gioco da parte dell amministratore.

### Not Functional Requirements

### Usability

I requisiti di “Usabilità” devono garantire la semplice interazione tra utente e sistema; colui che usufruisce dell’applicativo, che sia amministratore o semplice utente, deve poter interagire in modo chiaro ed intuitivo l’applicativo; dunque il suddetto elaborato sarà “interface user-friendly”. Inoltre tale software dovrà essere adatto ad utenti aventi conoscenze informatiche di base.

### Reliability

Il sistema garantirà sempre la disponibilità agli utenti, garantendo a tali utenti la corretta esecuzione di quanto richiesto ad una data azione, evitando, dunque, situazioni indesiderate (evitare che il sistema vada in “crash”, segnali eventuali errori e possibilmente risolverli guidando l’utente alla risoluzione di questi).

### Performance

Il sistema sarà performante garantendo agli utenti l’esecuzione delle loro richieste in un tempo massimo di pochi secondi. Tale sistema inoltre garantirà l’utilizzo contemporaneo dei suoi servizi a più utenti, senza che ciò influisca sulla velocità di elaborazione.

### Supportability

Il sistema sarà supportato da tutte le piattaforme che sono in grado di poter usufruire del servizio web; sarà, dunque, compatibile con tutti i tipi di web-browser.

### **Deliverable & deadlines**

Il sistema software dovrà avere una architettura three-tier con un client che implementa il livello di presentazione, un server che implementa la logica applicativa e un DBMS per la gestione dei dati. I tre livelli dovranno essere potenzialmente installabili su tre macchine diverse. I documenti da produrre ai fini del progetto sono i seguenti:

• Problem Statement

• Requirements Analysis Document

• System Design Document

• Object Design Document

• Test Plan

• Test Case Specification

• Test Execution Report