

Nome: 8-bit	Versione: 1.0
Documento: Problem Statement	Data: 10/10/2017

Università degli Studi di Salerno
Corso di Ingegneria del Software

8-bit
Problem Statement
Versione 1.0



Data: 10/10/2017

Coordinatore del progetto:

Nome	Matricola
Aniello Giugliano	0512103007

Partecipanti:

Nome	Matricola
Aniello Giugliano	0512103007
Giacomo Coccoziello	0512103097

Scritto da:	Aniello Giugliano
	Giacomo Coccoziello

INDICE:

- 1. Problem Domain**
- 2. Scenarios**
- 3. Functional Requirements**
- 4. NonFunctional Requirements**

PROBLEM DOMAIN

8-bit è un sito di e-commerce che permette agli appassionati di retrogaming la possibilità di acquistare on-line giochi che hanno segnato un'epoca. Il sito vuole comunque rivolgersi ad un pubblico che desidera avvicinarsi a questo nuovo (vecchio) mondo per la prima volta. I clienti infatti dovranno trovare nel sito un punto di riferimento per quanto riguarda l'acquisto di copie originali e certificate dalle maggiori software house degli albori del "videogaming".

SCENARIOS

SCENARIO CARICA PRODOTTO

Scenario Nome: caricaProdotto

Attore: Carlo: Gestore

Flusso di Eventi:

1. Carlo si collega al software mediante un qualsiasi browser.
2. Il software visualizza la homepage.
3. Carlo clicca sul tasto di login.
4. Il software risponde mostrando la pagina di login.
5. Carlo, per loggarsi, inserisce e-mail e password e clicca sulla voce Accedi.
6. Il software risponde mostrando l'area riservata all'amministratore.
7. Carlo entra nella pagina "inserimento di un gioco".
8. Il software mostra un form di inserimento del prodotto.
9. Carlo compila il form con le seguenti descrizioni:
Titolo, Prezzo, Anno di Uscita, Console, SoftwareHouse, Categoria, Descrizione e preme sul pulsante inserisci.
10. Il software elabora la richiesta.
11. Se l'elaborazione va a buon fine sarà mostrato il prodotto appena caricato, altrimenti sarà mostrato un messaggio di errore.

SCENARIO CARRELLO CLIENTE

Scenario Nome: carrelloCliente

Attore: Andrea: Cliente

Flusso degli Eventi:

1. Andrea si collega al software mediante un qualsiasi browser.
2. Il software visualizza la homepage.

3. Andrea clicca sul tasto di login.
4. Il software risponde mostrando la pagina di login.
5. Andrea, per loggarsi, inserisce e-mail e password e clicca sulla voce Accedi.
6. Il software restituisce l'area riservata all'utente.
7. Andrea accede allo store e prende visione dei vari prodotti e per accedere ai dettagli del prodotto clicca sul tasto dettagli.
8. Il software restituisce la pagina di descrizione del prodotto che contiene:
 Titolo, Descrizione, Prezzo, PrezzoScontato, Anno di Uscita, Genere, Console, SoftwareHouse, Lingua, Quantità consentendo di aggiungere al carrello il prodotto che si sta visionando.
9. Andrea procede all'aggiunta del prodotto nel carrello tramite il pulsante aggiungi al carrello.
10. Il software elabora la richiesta e permette ad Andrea di continuare la ricerca di altri prodotti. Inserendo una quantità maggiore rispetto alla quantità disponibile del prodotto e nel caso in cui la quantità non fosse specificata o si inserisce una quantità negativa, il software avviserà, tramite messaggio di errore.
11. Una volta selezionati i vari prodotti ne prende visione nella sezione carrello.
12. Il software per ogni prodotto nel carrello mostra i seguenti campi: descrizione, prezzo, quantità, sconto e totale.
9. Andrea può rimuovere un prodotto dal carrello premendo sul pulsante "elimina".
10. Il software risponderà ricaricando la pagina senza mostrare più il prodotto eliminato.

SCENARIO ACQUISTO PRODOTTI CLIENTE

Scenario Nome: acquistoProdottiCliente

Attore: Andrea: Cliente

Flusso di Eventi:

1. Andrea attraverso la propria sezione carrello visualizza i vari prodotti precedentemente selezionati.
2. Il software mostra la lista dei prodotti selezionati precedentemente.
3. Andrea ricontrolla i prodotti, verificando il

nome, descrizione e le quantità selezionate ed effettua l'acquisto premendo il pulsante acquista.

4. Il software elabora la richiesta del cliente. Se l'elaborazione va a buon fine il software visualizza nuovamente il carrello con le varie informazioni di tutti gli acquisti effettuati da Andrea, altrimenti verrà mostrato un messaggio di errore e non si potrà procedere all'acquisto.

SCENARIO ACQUISTO PRODOTTI UTENTI

Scenario Nome: acquistoProdottiUtenti

Attori: Michele: Utente(non registrato)

Flusso di eventi:

1. Michele accede al software mediante qualsiasi browser.
2. Il software visualizza la Homepage.
3. Michele preme sul pulsante Store.
4. Il software mostra la lista di tutti i prodotti disponibili sullo Store.
5. Michele prende visione dei vari prodotti e per accedere ai dettagli del prodotto clicca sul tasto dettagli.
6. Il software restituisce la pagina di descrizione del prodotto che contiene:
Titolo, Descrizione, Prezzo,
PrezzoScontato, Anno di Uscita, Genere,
Console, SoftwareHouse, Lingua, Quantità
consentendo di aggiungere al carrello il prodotto che si sta visionando.
7. Michele procede all'aggiunta del prodotto nel carrello tramite il pulsante aggiungi al carrello.
8. Il software elabora la richiesta dell'utente ma gli impedisce l'acquisto del prodotto perché non è loggato e gli mostra un form per la registrazione compilando i seguenti campi:
E-mail, Nickname, Password e la ripetizione della Password.
9. Michele compila il form, preme il pulsante Iscriviti.
10. Il software mostra l'area dell'utente appena registrato.
11. Michele può accedere all'acquisto del prodotto.

SCENARIO ISCRIZIONE ALLE NEWSLETTER

Scenario Nome:	iscrizioneNewsletter
Attore:	Andrea: Cliente
Flusso di Eventi:	<ol style="list-style-type: none">1. Andrea si collega al software mediante un qualsiasi browser.2. Il sistema visualizza la homepage.3. Andrea clicca sul pulsante “newsletter.”4. il sistema mostra un form contenente il campo e-mail.5. Andrea compila il form inserendo la sua e-mail. Se il procedimento è andato a buon fine ad Andrea arriverà una e-mail di avvenuta iscrizione, altrimenti comparirà un messaggio di errore sullo schermo.

Functional Requirements

RF1: Il software dovrà permettere agli utenti la visualizzazione dei prodotti commercializzati.

RF2: Il software dovrà permettere agli utenti di autenticarsi nel momento in cui vorranno selezionare potenziali prodotti da acquistare .

RF3: Il software dovrà permettere la registrazione dei vari clienti compilando un apposito modulo.

RF4: Il software dovrà permettere ai clienti di poter acquistare in diverse quantità uno o più prodotti.

RF5: Il software dovrà permettere ai clienti di poter modificare le quantità dei prodotti da acquistare.

RF6: Il software dovrà permettere ai clienti di effettuare visualizzazione di news e contatti.

RF7: Il software dovrà permettere ai clienti di iscriversi alle newsletter per essere aggiornati sui vari prodotti.

RF8: Il software dovrà permettere all’ amministratore l’inserimento di un gioco.

RF9: Il software dovrà permettere all’ amministratore la cancellazione di un gioco.

RF10: Il software dovrà permettere l’autenticazione tramite login e password, dei clienti e

dell'amministratore in maniera tale da dare la possibilità di accedere allo loro area utente.

NonFunctional Requirements

Usability

Il sistema sarà sviluppato per la visualizzazione web, è responsive per favorire un maggior utilizzo da parte degli utenti, è dotata di interfaccia semplice, immediata ed intuitiva. Per la grafica si utilizzerà uno stile vicino agli standard del HTML5, potendo appoggiarsi ad un framework come bootstrap.

Reliability

Il software dovrà essere attivo 24 ore su 24. Inoltre dovrà garantire la sicurezza su tutte le operazioni effettuate sia dall'amministratore che dagli utenti. Nel caso in cui si verificano dei comportamenti anomali, da parte dell'amministratore o degli utenti, verranno notificati tramite avvisi.

Performance

Il software dovrà rispondere velocemente alle richieste del sistema. Il numero degli utenti che potranno collegarsi e acquistare prodotti contemporaneamente sarà dato dalla disponibilità del Web Server utilizzato.

Supportability

Il software consisterà in un sistema client/server in cui il server sarà disponibile su ogni tipo di piattaforma e il client potrà collegarsi a tale server mediante un qualsiasi browser. Il software dovrà essere suddiviso in vari sotto moduli per permettere una facile modifica e aggiornabilità in futuro.