Università degli Studi di Napoli Federico II

Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione

Classe di Laurea di Scienze e tecnologie Informatiche, Classe n. L-31

Corso di Laurea Triennale in Informatica

Progetto

Cerca Viaggi 2019

Docente:
Prof. Di Martino Sergio

Candidati:
Turco Mario
Matr. N86002503
Longobardi Francesco
Matr. N86002468
Rauso Giuseppe
Matr. N86002481

Academic Year
2019/2020

Indice

Revisioni **1**

I Documento dei Requisiti Software **2**

1 Introduzione **3**
1.1 Requisiti Funzionali 3
1.2 Requisiti non funzionali 4

2 Modello funzionale **5**
2.1 Modellazione dei casi d'uso 5
2.2 Tabelle di Cockburn 6
2.2.1 Amministratore 6
2.2.2 Utente 14
2.3 Mockup 23
2.3.1 Mobile 23
2.3.2 Dekstop 30

3 Modello di Dominio **38**
3.1 Classi, oggetti e relazioni di analisi 38
3.2 Diagrammi di sequenza di analisi 45
3.3 Diagrammi di stato di analisi 52
3.4 Diagrammi di attività di analisi 55

4 Impegno risorse e pianificazione dell'attività **59**
4.1 Diagramma di Gantt 59

II Documento di Design del sistema **62**

5 Design **63**
5.1 Descrizione del dominio 63

5.2 Analisi dell'architettura 63
5.2.1 Model-View-Presenter 64
5.3 Diagramma delle classi di design 65
Applicativo Desktop 65
Applicativo Mobile 68
5.4 CRC Cards 68
5.4.1 Entità 68
5.4.2 Applicativo Desktop 69
5.4.3 Applicativo Mobile 72
5.5 Diagramma di stato di design 75
5.6 Diagramma di sequenza di design 75

III Documento di Testing del sistema 89

6 Testing 90
6.1 Test Plan per il System Testing 90
6.1.1 Applicativo Desktop 90
6.1.2 Applicativo Mobile 95
6.2 Codice jUnit per unit testing 100
6.2.1 Testing Black Box 100
6.2.2 Testing White box 104

IV Glossario 107

Revisioni

Data	Versione	Autore	Descrizione
23-03-2020	0.1	Turco Mario	Prima versione del documento
24-03-2020	0.2	Turco Mario	Aggiunte tabelle di Cockburn degli usecase

Parte I

Documento dei Requisiti Software

Capitolo 1

Introduzione

Lo scopo di questo documento è quello di specificare i requisiti del sistema software "Cerca Viaggi" per facilitarne la realizzazione e la validazione ed, in particolare, si vogliono formalizzare i requisiti funzionali e non funzionali del sistema.
Il documento prevede vari livelli di raffinamento partendo dal linguaggio naturale ed arrivando ad un linguaggio strutturato ed a modelli UML.

1.1 Requisiti Funzionali

1. **Accesso amministratore al sistema**: un amministratore dovrà poter accedere al sistema tramite email e password (usando l'applicativo dekstop).

2. **Amministratore valuta una recensione**: un amministatore dovrà poter accettare o rifiutare una recensione scritta da un utente decidendo quindi se questa verrà pubblicata o eliminata.

3. **Accesso utente al sistema**: un utente, dopo aver effettuato la registrazione, dovrà poter accedere al suo account tramite email e password (usando l'applicativo mobile).

4. **Registrazione di un nuovo utente**: l'applicativo mobile dovrà permettere ad un utente, non precedentemente registrato, di creare un account specificando i propri dati anagrafici (nome, cognome e data di nascita), i dati per effettuare l'accesso (email e passowrd) ed un nickname da utilizzare, secondo volonta dell'utente, per firmare le proprie recensioni

5. **Amministratore modifica dati dei visitatori**: un amministratore dovrà poter modificare i dati di un utente in particolare, email e password

6. **Amministratore elimina account utente**: un amministratore dovrà poter eliminare l'account di un utente

7. **Amministratore visualizza dati di un visitatore**: un amministratore dovrà poter visualizzare i dati anagrafici di un visitatore (nome, cognome e data di nascita), il nickname ed il numero di recensioni approvate e rifiutate.

8. **Utente visualizza struttra**: un utente dovrà poter visualizzare i dettagli di una struttura dopo averla selezionata con la possibiltà di vederla sulla mappa.

9. **Utente scrive una recensione**: un utente, una volta aver effettuato l'accesso, dovrà poter scrivere una recensione ad una struttura.

10. **Utente visualizaza dettagli recensione**: un utente dovrà poter leggere i dettagli di una recensione, in particolare leggendone l'autore, la data di inserimento, la valutazione ed il testo integrale.

11. **Utente ricerca una struttura**: un utente dovrà poter ricercare una struttura in vari modi: per nome, tramite filtri, per prossimità o tramite l'ausilio della mappa.

1.2 Requisiti non funzionali

1. Il sistema deve garantire una buona usabilità per tutti coloro che ne usufruiscono.

Capitolo 2

Modello funzionale

2.1 Modellazione dei casi d'uso

2.2 Tabelle di Cockburn

2.2.1 Amministratore

Tabella 2.1: L'amministratore effettua il login

Use Case #1	Effettua Login
Goal in Context	L'amministratore vuole accedere all'applicativo di Back-Office
Preconditions	L'amministratore non ha effettuto lo use case "Effettua Login".
Success End Conditions	Il login dell'amministratore va a buon fine.
Failed End Conditions	I dati di login sono errati o il server non è raggiungibile.
Primary Actor	Amministratore
Trigger	L'amministratore preme il pulsante "Login" nella schermata Login visibile all'avvio del software.

Main Scenario		
Step#	Amministratore	Sistema
1	Compila correttamente i text-Field Username e Password	
2	Preme il pulsante "Login" dalla schermata Login	
3		Mostra schermata HomePage

Tabella 2.2: Effettua Login - Estensione 1

Extension 1: l'amministatore inserisce dati errati		
Step#	Amministratore	Sistema
1 a	Non complia o compila erroneamente i textField Username e Password	
2 a	Preme il pulsante Login	
3 a		Mostra 'Credenziali Errate Dialog' e termina caso d'uso

Tabella 2.3: Effettua Login - Estensione 2

Extension 2: il server non è raggiungibile		
Step#	Amministratore	Sistema
1 b	Compila i campi username e password	
2 b	Preme il pulsante Login	
3 a		Mostra 'Connessione assente dialog' e termina caso d'uso

Tabella 2.4: L'amministratore valuta una recensione

Use Case #2	Valuta Recensione
Goal in Context	L'amministratore valuta una recensione.
Preconditions	L'amministratore ha effettuto lo use case "Effettua Login".
Success End Conditions	L'amministratore valuta una recensione. Il sistema tiene traccia di tale operazione.
Failed End Conditions	L'amministatore preme annulla. L'amministrazione valuta una recensione che è già stata valutata. Il server non è raggiungibile.
Primary Actor	Amministratore
Trigger	L'amministratore preme il pulsante "Recensioni" nella Homepage.

Main Scenario

Step#	Amministratore	Sistema
1	Preme il pulsante "Recensioni" nella schermata principale	
2		Mostra schermata "Recensioni"
3	Clicca sulla card di una recensione	
4		Mostra schermata "Visualizza Recensione"
5	Clicca sul pulsante Accetta	
6		Mostra 'Recensione Approvata Dialog' e termina lo use case

Tabella 2.5: Valuta una recensione - Subvariation 1

Subvariation 1: l'amministatore rifiuta una recensione		
Step#	Amministratore	Sistema
5 i	Preme il pulsante "Rifiuta".	
6 i		Mostra 'Recensione Rifiutata Dialog' e termina lo use case.

Tabella 2.6: Valuta una recensione - Estensione 1

Extension 1: l'amministatore preme annulla		
Step#	Amministratore	Sistema
3/5 a	Preme il pulsante "Recensioni" dal menu laterale sinistro	
4/6 a		Ritorna alla schermata principale e termina il caso d'uso.

Tabella 2.7: Valuta una recensione - Estensione 2

Extension 2: la recensione è già stata valutata		
Step#	Amministratore	Sistema
4 b		Mostra Fallimento Dialog e termina caso d'uso

Tabella 2.8: Valuta una recensione - Estensione 3

Extension 3: il server non è raggiungibile		
Step#	Amministratore	Sistema
4/6 c		Mostra Connessione Assente Dialog e termina caso d'uso

Tabella 2.9: L'amministratore visualizza dati di un visitatore

Use Case #3	Visualizza dati visitatore
Goal in Context	L'amministratore visualizza i dati di un visitatore.
Preconditions	L'amministratore ha effettuto lo use case "Effettua Login".
Success End Conditions	L'amministratore visualizza con successo i dati di un visitatore
Failed End Conditions	Il server non è raggiungibile
Primary Actor	Amministratore
Trigger	L'amministratore preme il pulsante "Visitatori" a partire dalla Schermata "Recensioni" .
Main Scenario	

Step#	Amministratore	Sistema
1	Preme sul rigo corrispondente all'utente da visualizzare	
2		Mostra schermata "Visualizza Visitatore" e termina caso d'uso

Tabella 2.10: Visualizza dati visitatore - Estensione 1

Extension 1: il server non è raggiungibile		
Step#	Amministratore	Sistema
1	Preme sul rigo corrispondente all'utente da visualizzare	
2		Mostra schermata "Connessione assente" e termina caso d'uso

Tabella 2.11: L'amministratore modifica i dati di un visitatore

Use Case #4	Modifica i dati visitatore
Goal in Context	L'amministratore vuole modificare lo username e la password di un visitatore
Preconditions	L'amministratore ha effettuto lo use case "Effettua Login".
Success End Conditions	L'amministratore modifica con successo i dati di un visitatore
Failed End Conditions	Il server non è raggiungibile o l'amministratore non compila correttamente tutti i campi
Primary Actor	Amministratore
Trigger	L'amministratore preme il pulsante "Modifica" dalla Schermata "Visualizza Visitatore" .
Main Scenario	

Step#	Amministratore	Sistema
1	Preme sul tasto "Modifica" dalla schermata "Visualizza Visitatore"	
2		Mostra schermata "Modifica Visitatore"
3	Compila correttamente tutti i campi di testo	
4	Preme il tasto "Conferma"	
5		Mostra schermata "Modifica Success Dialog" e termina caso d'uso

Tabella 2.12: Modifica dati visitatore - Estensione 1

Extension 1: il server non è raggiungibile		
Step#	Amministratore	Sistema
1a	Preme sul tasto "Modifica" dalla schermata "Visualizza Visitatore"	
2a		Mostra schermata "Connessione Assente Dialog"

Tabella 2.13: Modifica dati visitatore - Estensione 2

Extension 2: l'amministratore compila erroneamente i campi di testo		
Step#	Amministratore	Sistema
3b	Non compila, compila parzialmente o compila in modo errato i campi di testo	
2a		Mostra schermata "Password o username invalidi dialog"

Tabella 2.14: L'amministratore elimina un visitatore

Use Case #5	Elimina un visitatore
Goal in Context	L'amministratore vuole eliminare un visitatore
Preconditions	L'amministratore ha effettuto lo use case "Effettua Login".
Success End Conditions	L'amministratore elimina un visitatore ed il sistema tiene traccia di questa azione
Failed End Conditions	Il server non è raggiungibile
Primary Actor	Amministratore
Trigger	L'amministratore preme il pulsante "Elimina" dalla Schermata "Visualizza Visitatore" .
Main Scenario	

Step#	Amministratore	Sistema
1	Preme sul tasto "Elimna" dalla schermata "Visualizza Visitatore"	
2		Mostra schermata "Utente Eliminato Dialog" e termina caso d'uso

Tabella 2.15: Elimina visitatore - Estensione 1

Extension 1: il server non è raggiungibile		
Step#	Amministratore	Sistema
1a	Preme sul tasto "Elimina" dalla schermata "Visualizza Visitatore"	
2a		Mostra schermata "Connessione Assente Dialog"

2.2.2 Utente

Tabella 2.16: L'utente effettua il Login - Main Scenario

Use Case #1	Utente effettua Login
Goal in Context	L'utente vuole effettuare l'accesso al suo account
Preconditions	L'utente non ha effettuato lo use case Effettua Login
Success End Conditions	Il login dell'utente va a buon fine.
Failed End Conditions	I dati di login sono errati oppure il server non è raggiungibile.
Primary Actor	Utente non loggato
Trigger	L'utente preme il pulsante 'Login' nel Navigation Drawer laterale dalla schermata 'HomePage utente non loggato'.

Main Scenario		
Step#	Utente	Sistema
1	Compila correttamente i textField Username e Password	
2	Preme il pulsante "Login" dalla schermata LoginForm	
3		Mostra schermata HomePage

Tabella 2.17: Effettua Login - Estensione 1

Extension 1: l'utente inserisce dati errati		
Step#	Utente	Sistema
1 a	Compila erroneamente i textField Username e Password	
2 a	Preme il pulsante Login	
3 a		Mostra schermata Dati errati Dialog e termina caso d'uso

Tabella 2.18: Effettua Login - Estensione 2

Extension 2: l'utente non compila correttamente i campi di testo		
Step#	Utente	Sistema
1 b	Non compila o compila solo un tra i textField Username e Password	
2 b	Preme il pulsante Login	
3 b		Mostra schermata 'Campi vuoti Dialog' e termina caso d'uso

Tabella 2.19: Effettua Login - Estensione 3

Extension 3: il server risulta non raggiungibile		
Step#	Utente	Sistema
1 c	Compila correttamente tutti i campi	
2 c	Preme il tasto "Login"	
3 c		Mostra schermata "server irraggiungibile dialog" e termina caso d'uso

Tabella 2.20: L'utente non loggato effettua la registazione

Use Case #2	L'utente non loggato si registra alla piattaforma
Goal in Context	L'utente vuole registrare un nuovo account
Preconditions	-
Success End Conditions	La registrazione dell'utente va a buon fine.
Failed End Conditions	Il server non è raggiungibile o l'utente immette dati non validi.
Primary Actor	Utente non loggato
Trigger	L'utente preme il pulsante 'Registrati' nel Navigation Drawer laterale dalla schermata 'HomePage utente non loggato'.

Main Scenario

Step#	Utente	Sistema
1	Compila correttamente tutti i textField ed il date picker	
2	Preme il pulsante "Fine" dalla schermata Registrazione	
3		Mostra schermata "Registrazione success dialog" e termina il caso d'uso

Tabella 2.21: Effettua registrazione - Estensione 1

Extension 1: l'utente inserisce dati di un utente già registrato		
Step#	Utente	Sistema
1 a	Compila i textField inserendo i dati di un account già registrato	
2 a	Preme il pulsante "Fine"	
3 a		Mostra schermata "Utente esistente fail dialog" e termina il caso d'uso

Tabella 2.22: Effettura registrazione - Estensione 2

Extension 2: l'utente non compila tutti i campi o li compila in modo errato		
Step#	Utente	Sistema
1 b	Non compila o compila erroneamente i textfiel ed il datepicker	
2 b	Preme il pulsante "Fine"	
3 b		Mostra a schermo il messaggio "Compilare tutti i campi" e termina caso d'uso

Tabella 2.23: Effettura registrazione - Estensione 3

Extension 3: il server risulta non raggiungibile		
Step#	Utente	Sistema
1 c	Compila correttamente tutti i campi della registrazione	
2 c	Preme il tasto "Fine"	
3 c		Mostra schermata "Registrazione server irraggiungibile dialog" e termina caso d'uso

Tabella 2.24: L'utente visulizza una struttura

Use Case #3	L'utente visulizza una struttura
Goal in Context	L'utente vuole visualizzare i dettagli di una struttura
Preconditions	-
Success End Conditions	L'utente visualizza con successo i dettagli di una struttura
Failed End Conditions	Il server non è raggiungibile
Primary Actor	Utente
Trigger	L'utente clicca sulla card di una struttura dalla schermata "Lista Strutture"

Main Scenario		
Step#	Utente	Sistema
1	Clicca la Card di una struttura dalla schermata "Lista Strutture"'.	
2		Mostra la schermata "Pagina Struttura" e termina caso d'uso

Tabella 2.25: Visualizza struttura - Estensione 1

Extension 1: il server non è raggiungibile		
Step#	Utente	Sistema
2 a		Mostra schermata "Connessione assente" e termina caso d'uso

Tabella 2.26: L'utente visulizza i dettagli di una recensione

Use Case #4	L'utente visulizza i dettagli di una recensione
Goal in Context	L'utente vuole visualizzare il testo completo di una recensione
Preconditions	
Success End Conditions	L'utente visualizza i dettagli di una recensione
Failed End Conditions	Il server non è raggiungibile
Primary Actor	Utente
Trigger	L'utente clicca sulla card di una recensione dalla schermata "Pagina Struttura"

Main Scenario		
Step#	Utente	Sistema
1	Clicca la Card di una recensione dalla schermata "Pagina struttura"'.	
2		Mostra la schermata "Pagina Recensione" e termina caso d'uso

Tabella 2.27: Visualizza struttura - Estensione 1

Extension 1: il server non è raggiungibile		
Step#	Utente	Sistema
2 a		Mostra schermata "Connessione assente" e termina caso d'uso

Tabella 2.28: L'utente aggiunge una recensione

Use Case #5	L'utente aggiunge una recensione
Goal in Context	L'utente vuole aggiungere una recensione
Preconditions	
Success End Conditions	L'utente aggiunge con successo una recensione ed il sistema tiene traccia di tale operazione
Failed End Conditions	Il server non è raggiungibile, l'utente non compila tutti i campi
Primary Actor	Utente Autenticato
Trigger	L'utente clicca sul Floating Action Button e preme su "Aggiungi recensione" dalla schermata "Pagina Struttura"

Main Scenario

Step#	Utente	Sistema
1		Mostra schermata Aggiungi Recensione
2	Compila correttamente tutti i campi della schermata	
3	Preme il tasto "Aggiungi"	
4		Mostra la schermata "Aggiungi recensione success Dialog" e termina il caso d'uso

Tabella 2.29: Aggiungi recensione- Estensione 1

Extension 1: il server non è raggiungibile

Step#	Utente	Sistema
4 a		Mostra schermata "Aggiungi recensione errore dialog" e termina caso d'uso

Tabella 2.30: Aggiungi recensione- Estensione 2

Extension 1: l'utente non compila uno o più campi		
Step#	Utente	Sistema
1 b	Compila il form tralasciando uno o più campi	
2 b	Preme il tasto "Aggiungi"	
3 b		Mostra schermata "Aggiungi recensione empty dialog" e termina caso d'uso

Tabella 2.31: L'utente visulizza la posizione della struttura sulla mappa

Use Case #6	L'utente visulizza la posizione della struttura sulla mappa	
Goal in Context	L'utente vuole visualizzare tramite la mappa la posizione della mappa	
Preconditions	-	
Success End Conditions	L'utente visualizza con successo la struttura sulla mappa	
Failed End Conditions	L'utente non ha connessione internet	
Primary Actor	Utente	
Trigger	L'utente clicca sull'opzione "Visualizza su mappa" dal Floating Action Button presente in "Pagina Struttura"	
Main Scenario		
Step#	Utente	Sistema
1	Clicca sull'opzione "Visualizza su mappa" dal Floating Action Button	
2		Mostra la schermata "Visualizza su mappa" e termina caso d'uso

Tabella 2.32: Visualizza struttura - Estensione 1

Extension 1: l'utente non ha connessione internet		
Step#	Utente	Sistema
2 a		Mostra schermata "Connessione assente" e termina caso d'uso

Tabella 2.33: L'utente cerca una struttura

Use Case #7	Utente cerca una struttura
Goal in Context	L'utente vuole cercare una struttura.
Preconditions	-
Success End Conditions	L'utente effettua con successo la ricerca
Failed End Conditions	Il server non è raggiungibile
Primary Actor	utente
Trigger	
Main Scenario	

Step#	Utente	Sistema
1	Clicca sull'iconda dei filtri sulla schermata "Homepage"	
2		Mostra schermata "Ricerca Filtri"
3	Compila i textfield ed i comboBox in base alle necessità della sua ricerca	
4	Preme il pulsante "Cerca"	
5		Mostra schermata "Lista Strutture" e termina caso d'uso

2.3 Mockup

Presentiamo di seguito i principali mockup dell'applicativo Desktop e Mobile.

2.3.1 Mobile

Login

Dati errati Dialog

Server irraggiungibile Dialog

Campi vuoti Dialog

Registrazione

Registrazione success Dialog

Utente esistente fail dialog

Server irraggiungibile dialog

Homepage

Ricerca Filtri

Pagina Struttura

Lista Strutture

Floating Action Button Struttura

Dialog Filtri Recensione

Aggiungi recensione

Visualizza su mappa

Aggiungi recensione errore dialog

Aggiungi recensione success dialog

Aggiungi recensione empty dialog

Navigation Drawer utente loggato

Navigation Drawer utente non loggato

Pagina recensione

Connessione Assente

2.3.2 Dekstop

Login

Credenziali errate Dialog

Connessione assente dialog

Recensioni

Visualizza Recensione

Recensione approvata Dialog

Recensione rifiutata dialog

Fallimento Dialog

Visitatori

Visualizza Visistatore

Visitatori

Visualizza Visistatore

Password o username errati Dialog

Utente eliminato dialog

Tabella 2.34: Ricerca Struttura - Subvariation 1

Subvariation 1: l'utente cerca una struttura tramite filtri		
Step#	Utente	Sistema
4 ii	Clicca sullo switch "prossimità", abilitando la ricerca per prossimità	
5 ii	Preme il pulsante "Cerca"	
6 ii		Mostra schermata "Ricerca Filtri"

Tabella 2.35: Ricerca Struttura - Subvariation 2

Subvariation 2: l'utente cerca una struttura per prossimità		
Step#	Utente	Sistema
1i	Scorre sulla mappa presente nella schermata "Homepage" fino a trovare la struttura desiderata e termina caso d'uso	

Tabella 2.36: Ricerca Struttura - Estensione 1

Extension 1: il server non è raggiungibile		
Step#	Utente	Sistema
5a		Mostra schermata "Connessione assente" e termina caso d'uso

Capitolo 3

Modello di Dominio

3.1 Classi, oggetti e relazioni di analisi

Si riportano di seguito i Class Diagram di analisi dei casi d'uso.

Admin visualizza dati Utente

Admin elimina utente

Admin effettua Login

Admin modifica Utente

<<boundary>>
GestioneRecensioniView
+seleziona(Recensione)
+mostra(List<Recensione>)
+confermaClick()
+annullaClick()
<<boundary>>
ValutaRecensioneView
+ approvaClick(Recensione)
+ rifiutaClick(Recensione)
+ annullaClick()
+ mostra(Recensione)
<<enumeration>>
Stato
+ Approvata
+ Rifiutata
+ In moderazione
<<enumeration>>
Prezzo
economico
medio
alto
<<boundary>>
RifiutaButton
+ click()
<<boundary>>
SuccessDialog
+ clickOk()
+ mostra()
<<controller>>
GestioneRecensioneController
+ mostaVautaRecensione(Recensione)
+ mostraGestioneRecensioni(List<Recensione>)
+ approvaRecensione(Recensione)
+ rifiutaRecensione(Recensione)
<<boundary>>
ApprovaButton
+ click()
<<boundary>>
FailDialog
+ clickOk()
+ mostra()
<<boundary>>
ConfermaButton
+ click()
<<entity>>
Amministratore
- UserName: String
- Password: String
+ controllaCredenziali(String, String): boolean
1
*
Viene valutata
Valuta
<<entity>>
Recensione
- valutazione: int
- titolo: String
- data: Date
- approvata: Stato
- testo: String
+ getStruttura(): Struttura
*
1
Possiede
si riferisce
<<entity>>
Struttura
- nome: String
- indirizzo: String
- valutazioneMedia: float
- categoria: String
- descrizione: String
- fasciaDiPrezzo: Prezzo
- città: String
- longitudine: double
- latitudine: double
+ calcolaValutazioneMedia(List<Recensione>): void
+ getRecensioni(): List<Recensione>
+ addRecensione(Recensione): void
+ getStruttureFiltri(filtri: Map): List<Struttura>
+ getStruttureProssimità(filtri: Map): List<Struttura>
*
Scrive
È scritta
1
<<entity>>
Utente
- userName: String
- password: String
- recensioniApprovate: int
- recensioniRifiutate: int
- nome: String
- cognome: String
- email: String
- dataDiNascita: Date
- mostraUsername: boolean
+ calcolaRecensioniApprovate(List<Recensione): void
+ calcolaRecensioniRifiutate(List<Recensione>): void
+ usernameEsistente(username: String): boolean
+ validaCredenzialiUtente(nickname: String, password: String): boolean

Registrazione utente

Utente visualizza dettagli struttura

Utente visualizza Struttura

Utente ricerca Strutture

Utente effettua Login

Utente visualizza dettagli recensione

3.2 Diagrammi di sequenza di analisi

Si riportano di seguito i diagrammi di sequenza di analisi dei casi d'uso.

Admin elimina utente

Admin modifica utente

Admin visualizza dati utente

Admin gestisce una recensione

Admin esegue login

Registrazione utente

Utente esegue login

Utente ricerca con filtri

Utente visualizza dettagli struttura

Utente visualizza struttura su mappa

Utente scrive una recensione

Utente visualizza dettagli recensione

3.3 Diagrammi di stato di analisi

Sono riportati di seguito i diagrammi di stato di alcuni casi d'uso.

Utente ricerca Strutture

Utente aggiunge recensione

Utente esegue login

3.4 Diagrammi di attività di analisi

Sono riportati di seguito i diagrammi di attività di alcuni casi d'uso.

Admin modifica Utente

Utente scrive una recensione

Utente visualizza recensioni

Capitolo 4

Impegno risorse e pianificazione dell'attività

Si riporta di seguito il diagramma di Gantt con una proposta di impiego di risorse e di pianificazione dell'attività.

4.1 Diagramma di Gantt

Pianificazione CercaViaggi 19

Durata del piano | Inizio effettivo | % completamento | Effettiva (oltre piano) | % completamento (oltre piano)

ATTIVITÀ	INIZIO DEL PIANO	DURATA DEL PIANO	INIZIO EFFETTIVO	DURATA EFFETTIVA	PERCENTUALE DI COMPLETAMENTO
Raccolta dei requisiti	1	3	1	2	100%
Analisi dei requisiti	3	5	2	5	100%
Stesura del modello funzionale	8	7	5	6	100%
Specifica dei requisiti	15	7	11	7	100%
Stesura del modello di dominio	22	4	18	7	100%
System Design	26	7	25	7	100%
Object Design	33	7	30	7	100%
Stesura del documento di Design	40	2	38	3	100%
Implementazione	42	15	41	15	100%
Stesura Test Plan	22	15	18	19	100%
Unit testing	57	7	56	7	100%

DATA DI INIZIO EFFETTIVA: 12/12/2019 (Giorno 1)

DATA DI FINE EFFETTIVA: 13/03/2020 (Dopo 62 giorni *lavorativi**)

***Sono esclusi Sabato, Domenica e le feste nazionali**

ATTIVITÀ	INIZIO DEL PIANO	DURATA DEL PIANO	INIZIO EFFETTIVO	DURATA EFFETTIVA	PERCENTUALE DI COMPLETAMENTO
Raccolta dei requisiti	1	3	1	2	100%
Analisi dei requisiti	3	5	2	5	100%
Stesura del modello funzionale	8	7	5	6	100%
Specifica dei requisiti	15	7	11	7	100%
Stesura del modello di dominio	22	4	18	7	100%
System Design	26	7	25	7	100%
Object Design	33	7	30	7	100%
Stesura del documento di Design	40	2	38	3	100%
Implementazione	42	15	41	15	100%
Stesura Test Plan	22	15	18	19	100%
Unit testing	57	7	56	7	100%

DATA DI INIZIO EFFETTIVA: **12/12/2019** **(Giorno 1)**

DATA DI FINE EFFETTIVA: **13/03/2020** **(Dopo 62 giorni *lavorativi**)**

***Sono esclusi Sabato, Domenica e le feste nazionali**

Parte II

Documento di Design del sistema

Capitolo 5

Design

5.1 Descrizione del dominio

In seguito ad un'analisi dei requisiti del sistema si è resa evidente l'esistenza di alcune entità principali:

- gli *Utenti* che possono registrarsi, visualizzare delle *Strutture* e scrivere delle recensioni
- degli *Amministratori* i quali possono gestire i dati dei visitatori e le loro recensioni

5.2 Analisi dell'architettura

Il sistema si basa principalmente su due pattern architetturali: Client-Server e Model-View-Controller.
Il sistema infatti è composto da diversi client che tramite internet interagiscono con un Server remoto.

Figura 5.1: Architettura client server

5.2.1 Model-View-Presenter

Per quanto riguarda l'architettura interna dei client si è deciso di utilizzare il modello Model-View-Presenter (variante Passive View) favorendo la separazione dei componenti software del sistema e separandone le responsabilità. In questo modo si riesce ad ottenere alta coesione e basso accoppiamento, ottenendo una buona manutenibilà.
A favore di ciò il progetto è diviso in tre sottocartelle principali: Model, View e Presenter.

Il modello MVP è un'evoluzione del modello Model View Controller ed è usato principalmente per creare user interfaces ed in particolare nella varian-

te Passive View: la componente **View** è completamente passiva e statica deputata soltanto alla visualizzazione dei dati.
la componente **Model** ha la responsabilità di fornire i metodi per accedere ai dati utili all'applicazione, ha conoscenza del dominio applicativo ed è indipendente dagli altri sottosistemi.
la componente **Presenter** fa da "middle man" tra la View ed il Model e si occupa di prendere i dati dal model, fornirli alla view e gestire gli input dell'utente

Figura 5.2: Architettura Model-View-Controller

5.3 Diagramma delle classi di design

Applicativo Desktop - Diagramma delle classi di design

Applicativo Mobile - Diagramma delle Classi di Design

com.google.android.gms
com.mapbox.mapboxsdk
com.android.volley
com.squareup.picasso.Picasso
com.mapbox.mapboxsdk.maps.MapboxMap
com.google.android.gms.maps.SupportMapFragment
com.google.android.gms.maps.OnMapReadyCallback
com.google.android.gms.maps.GoogleMap
com.google.android.gms.maps.model.LatLng
com.google.android.gms.maps.model.MarkerOptions
com.google.android.gms.maps.model.BitmapDescriptor
com.google.android.gms.maps.model.BitmapDescriptorFactory
com.google.android.gms.maps.model.CameraUpdate
com.google.android.gms.maps.CameraUpdateFactory
com.google.android.gms.maps.model.LatLngBounds
com.google.android.gms.maps.model.LatLngBounds.Builder
com.google.android.gms.maps.model.Polyline
com.google.android.gms.maps.model.PolylineOptions
com.google.android.gms.maps.model.Polygon
com.google.android.gms.maps.model.PolygonOptions
com.google.android.gms.maps.model.Marker
com.google.android.gms.maps.model.MarkerOptions
com.google.android.gms.maps.model.Circle
com.google.android.gms.maps.model.CircleOptions
com.google.android.gms.maps.model.BitmapDescriptorFactory
com.google.android.gms.maps.model.MapStyleOptions
com.google.android.gms.maps.model.PointOfInterest
com.google.android.gms.maps.model.IndoorBuilding
com.google.android.gms.maps.model.IndoorLevel
com.google.android.gms.maps.model.TileProvider
com.google.android.gms.maps.model.UrlTileProvider
com.google.android.gms.maps.model.TileOverlayOptions
com.google.android.gms.maps.model.TileOverlay
com.google.android.gms.maps.model.GroundOverlay
com.google.android.gms.maps.model.GroundOverlayOptions
com.google.android.gms.maps.model.VisibleRegion
com.google.android.gms.maps.model.LatLngBounds.Builder
com.google.android.gms.maps.model.Projection
com.google.android.gms.maps.model.MapBuilderOptions
com.google.android.gms.maps.model.MapBuilder
com.google.android.gms.maps.model.MapBuilder.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.Builder
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions
com.google.android.gms.maps.model.MapBuilder.BuilderOptions

5.4 CRC Cards

Di seguitop sono riportate tutte le Class Responsability Collaboration(CRC) del prodotto.

5.4.1 Entità

Nome Classe	Recensione
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Rappresentare l'entità Recensione	Utente, Struttura

Nome Classe	Struttura
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Rappresentare l'entità Struttura	-

Nome Classe	Utente
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Rappresentare l'entità Utente	-

Nome Classe	Admin
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Rappresentare l'entità Amministratore	-

Nome Classe	Filtri
Superclassi	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Rappresenta l'entità Filtro applicabile in una ricerca	-

5.4.2 Applicativo Desktop

Nome Classe	AdminDAO
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Effettuare le operazioni CRUD per l'entità Admin su database Postgres	-

Nome Classe	UtenteDAO
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Effettuare le operazioni CRUD per l'entità Utente su database Postgres	-

Nome Classe	RecensioneDAO
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Effettuare le operazioni CRUD per l'entità Recensione su database Postgres	-

Nome Classe	LoginPresenter
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Gestire l'interazione per il login	AdminDAO, Admin

Nome Classe	SideMenuPresenter
Superclasse	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Gestire l'interazione con la SideBar	Admin

Nome Classe	ModificaUtentePresenter
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Gestire l'interazione per la modifica delle credenziali di accesso di un utente	Utente, UtenteDAO

Nome Classe	RecensioniPresenter
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostra le recensioni in attesa di approvazione sotto forma tabellare	RecensioneDAO

Nome Classe	UtentiPresenter
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Riempire la tabella con gli utenti registrati alla piattaforma	UtenteDAO

Nome Classe	VisualizzaRecensionePresenter
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Gestire l'interazione per la moderazione di una recensione da parte di un admin	RecensioneDAO, Recensione

Nome Classe	VisualizzaUtentePresenter
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostrare i dati completi di un utente	UtenteDAO, Utente
Permette l'eliminazione dell'utente mostrato	UtenteDAO, Utente

Nome Classe	PasswordUtils
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Fornire funzionalità per il sistema di cifratura delle password tramite Sale	-

Nome Classe	MainApp
Superclasse	Application
Sottoclassi	-

Responsabilità	**Collaboratore**
Avviare con i giusti parametri la finestra con la prima schermata	-

5.4.3 Applicativo Mobile

Nome Classe	UtenteDAO
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Effettuare le operazioni CRUD per l'entità Utente tramite richiesta GET HTTP al web service	-

Nome Classe	RecensioneDAO
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Effettuare le operazioni CRUD per l'entità Recensionetramite richiesta GET HTTP al web service	-

Nome Classe	StrutturaDAO
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Effettuare le operazioni CRUD per l'entità Struttura tramite richiesta GET HTTP al web service	-

Nome Classe	RegistrazioneFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostra il form di registrazione di un utente	-
Gestisce l'operazione di registrazione di un utente	UtenteDAO, Utente

Nome Classe	LoginFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Gestisce l'accesso, tramite credenziali, di un utente	UtenteDAO

Nome Classe	AggiungiRecensioneFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostra il form per l'aggiunta di una recensione	
Gestisce l'aggiunta di una recensione	RecensioneDAO

Nome Classe	ConnessioneAssenteFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostra una schermata che avvisa l'utente dell'impossibilità di connettersi al server.	-

Nome Classe	ListaRecensioniRecyclerViewAdapter
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Gestisce la lista delle strutture mostrate	Struttura
Gestisce il click sulle strutture della lista	Struttura

Nome Classe	ListaStruttureFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostra la lista delle strutture risultanti da una ricerca	Strutture

Nome Classe	DettagliStrutturaFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Permette la visualizzazione dei dati di una struttura	Struttura, StrutturaDAO
Permette la visualizzazione delle Recensioni di una struttura	Recensione, RecensioneDAO

Nome Classe	PasswordUtils
Superclasse	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Fornire funzionalità per il sistema di cifratura delle password tramite Sale	-

Nome Classe	NessunaStrutturaTrovata
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Mostra una schermata apposita nel caso in cui una ricerca non abbia nessun risultato da visualizzare	-

Nome Classe	VisualizzaRecensioneFragment
Superclassi	-
Sottoclassi	-

Responsabilità	**Collaboratore**
Permette la visualizzazione (con testo integrale) di una Recensione	Recensione

Nome Classe	FiltriFragment
Superclassi	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Gestisce l'applicazione dei flitri alla ricerca delle strutture	StrutturaDAO

Nome Classe	FiltriRecensioniDialog
Superclassi	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Gestisce l'applicazione di filtri alla lista delle recensioni di una Struttura	-

Nome Classe	VisualizzaSuMappaDialog
Superclassi	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Permette di visualizzare la posizione di una Struttura sulla mappa, all'interno di un dialog	Struttura

Nome Classe	MainActivity
Superclassi	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Inizializza l'app quando viene avviata	-
Richiede i permessi per l'uso del gps	-
Visualizza la mappa sulla homepage	-
Visualizza la schermata di una struttura dopo averci cliccato sopra dalla mappa	StrutturaDAO

Nome Classe	CustomSupportMapFragment
Superclassi	-
Sottoclassi	-
Responsabilità	**Collaboratore**
Gestisce la mappa visualizzata sull'homepage	-

5.5 Diagramma di stato di design

5.6 Diagramma di sequenza di design

Admin elimina Utente

Admin effettua login

Admin gestisce recensione

Admin modifica Utente

Admin visualizza Utente

Utente effettua Registrazione

Utente effettua Login

Utente visualizza dettagli Struttura

Utente visualizza Struttura su mappa

Utente visualizza dettagli recensione

Utente ricerca con filtri 1/2

Utente ricerca con filtri 2/2

Utente scrive recensione

Parte III

Documento di Testing del sistema

Capitolo 6

Testing

Il documento di testing è mirato a verificare e validare il sistema in modo che sia conforme alle specifiche ed ai requisiti evidenziati dai precedenti documenti.

6.1 Test Plan per il System Testing

6.1.1 Applicativo Desktop

Tabella 6.1: Test Case #1

Applicativo	Desktop
Nome	Amministratore effettua login
Descrizione	Test che assicura il corretto funzionamento del login dell'amministratore
Note	Esiste un amministratore con le seguenti credenziali: "admin", "admin"
Stato	Passato/Fallito

Continua alla pagina seguente

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene aperto l'applicativo Desktop	Viene visualizzata la schermata "Login" contententue due Textbox (Username e Password) ed un tasto	Come ci si aspettava
2	Vengono inserite delle credenziali **errate** "0000", "0000" e viene cliccato il tasto "Login"	Il login fallisce e viene visualizzata il dialog "Credenziali errate"	Come ci si aspettava
3	Viene premuto il tasto "OK"	Viene mostrata nuovamente la schermata di login	Come ci si aspettava
4	Vengono inserite le credenziali "admin", "admin" e viene cliccato il tasto "Login"	Il login ha successo e viene visualizzata la schermata " Revisioni"	Come ci si aspettava

Tabella 6.2: Test Case #2

Applicativo	Desktop
Nome	Amministratore visualizza dati di un utente
Descrizione	Test che assicura il corretto funzionamento della visualizzazione dei dati dell'utente
Note	L'amministratore deve essere loggato, deve esistere l'utente con nickname "utente"
Stato	Passato/Fallito

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene premuto il tasto "Visitatori" dalla schermata "Recensioni"	Viene visualizzata la schermata "Visitatori" contententue una lista di visitatori	Come ci si aspettava
2	Viene cliccata la riga contenente l'utente con nickname "utente"	Viene visualizzata la schermata utente contenete i tasti "Modifica", "Elimina" ed i textfild nome, nickname, data di nascita, email, recensioni rifiutate ed approvate contenenti i relativi dati.	Come ci si aspettava

Tabella 6.3: Test Case #3

Applicativo	Desktop
Nome	Amministratore valuta una recensione
Descrizione	Test che assicura il corretto funzionamento della valutazione delle recensioni
Note	L'amministatore deve essere autenticato e deve esistere una recensione scritta da "utente" per la struttura "Struttura" in data "GEN 1 2020"
Stato	Passato/Fallito

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene premuto il tasto "Recensioni" dalla schermata "Visitatori"	Viene visualizzata la schermata "Recensioni" contententue una lista di recensioni	Come ci si aspettava
2	Viene cliccata la riga contenente la recensione scritta in data "GEN 1 2020" dall'utente "utente" sulla struttura "Struttura"	Viene visualizzata la schermata della recensione contenete i tasti "Acceta" e "Rifiuta", i textFiel contenenti il titolo, il testo e l'autore della descrizione. Viene visualizzato, in oltre, il numero di stelline della recensione.	Come ci si aspettava
3	Viene cliccato il bottone "Accetta" o "Rifiuta"	Viene visualizzato il dialog che ci informa che l'azione è stata svolta con successo	Come ci si aspettava
4	Viene premuto il tasto "Ok"	Viene visualizzata la schermata Recensioni e nella lista delle recensioni non compare più la recensione scritta dall'utente "Utente" sulla struttura "Struttura" in data "GEN 1 2020"	Come ci si aspettava

Tabella 6.4: Test Case #4

Applicativo	Desktop
Nome	Amministratore elimina visitatore
Descrizione	Test che assicura il corretto funzionamento dell'eliminazione di un visitatore'
Note	L'amministatore deve essere autenticato e deve esistere l'utente "utente"
Stato	Passato/Fallito

Continua alla pagina seguente

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene premuto il tasto "Visitatori" dalla schermata "Recensioni"	Viene visualizzata la schermata "Visitatori" contententue una lista di visitatori	Come ci si aspettava
2	Viene cliccata la riga contenente l'utente "utente"	Viene visualizzata la schermata di visualizzazione dell'utente contenente i bottoni "Elimina" e "Modifica" Sulla schermata sono presenti anche i seguenti textfield: nome, email, nickname, data di nascita, recensioni approvate e rifiutate .	Come ci si aspettava
3	Viene cliccato il bottone "Elimina"	Viene visualizzato il dialog che ci informa che l'utente è stato eliminato con successo	Come ci si aspettava
4	Viene premuto il tasto "Ok"	Viene visualizzata visualizzata nuovamente la schermata con la lista dei visitatori	Come ci si aspettava

Tabella 6.5: Test Case #5

Applicativo	Desktop
Nome	Amministratore modifica dati visitatore
Descrizione	Test che assicura il corretto funzionamento della modifica dei dati di un visitatore'
Note	L'amministatore deve essere autenticato e deve esistere l'utente "utente" con nickname "user"
Stato	Passato/Fallito

Continua alla pagina seguente

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene premuto il tasto "Visitatori" dalla schermata "Recensioni"	Viene visualizzata la schermata "Visitatori" contententue una lista di visitatori	Come ci si aspettava
2	Viene cliccata la riga contenente l'utente "utente"	Viene visualizzata la schermata di visualizzazione dell'utente contenente i bottoni "Elimina" e "Modifica" Sulla schermata sono presenti anche i seguenti textfield: nome, email, nickname, data di nascita, recensioni approvate e rifiutate .	Come ci si aspettava
3	Viene cliccato il bottone "Modifica"	Viene visualizzata la schemata di modifica dei tasti utente contenente due editText ("Nickname" e "Password") e due bottoni ("Annulla" e "Conferma")	Come ci si aspettava
4	Viene inserito il Nickname "user" e la nuova password: "pass"; viene premuto il tasto "Ok"	Viene visualizzata un dialog che avvisa che l'azione non ha avuto successo	Come ci si aspettava
5	Viene premuto il tasto "OK"	Viene visualizzata nuovamente la schermata contenente le informazioni di "utente"	Come ci si aspettava
6	Viene inserito il Nickname "nuovoNickname" e la nuova password: "pass"; viene premuto il tasto "Ok"	Viene visualizzata un dialog che avvisa che l'azione ha avuto successo	Come ci si aspettava
7	Viene premuto il tasto "Ok"	Viene visualizzata nuovamente la schermata con le informazioni dell'utente e viene visualizzato il nickname ("nuovoNickname")	Come ci si aspettava

6.1.2 Applicativo Mobile

Tabella 6.6: Test Case #1

Applicativo	Mobile
Nome	Utente si registra alla piattaforma
Descrizione	Test che assicura il corretto funzionamento della funzione di registrazione
Note	L'utente non deve aver effettuato il login e non deve esistere nessun utente registrato con i seguenti dati: nickname = "nick" email = "email.email@gmail.com".
Stato	Passato/Fallito

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Dall'homepage viene cliccata l'iconda del menù in alto a sinistra	Viene visualizzato il menù laterale sinsitra	Come ci si aspettava
2	Viene cliccata la riga contenente "Registrati"	Viene visualizzata la schermata di registrazione con il bottone "Fine", 5 TextField, 1 datePicker ed una checkbox	Come ci si aspettava
3	Non viene compilato nessun campo e viene premuto il tasto "Fine"	Viene mostrato un dialog di errore	Come ci si aspettava
4	Viene premuto il tasto "Ok"	Viene visualizzata nuovamente la schermata di registrazione	Come ci si aspettava
5	Vengono inseriti i campi: "email.email@gmail.com", "password","Mario", "Rossi", "17/11/1998", "nick" e viene premuto il tasto "Fine".	Viene visualizzata mostrato il dialog di registrazione effettuata	Come ci si aspettava

Tabella 6.7: Test Case #2

Applicativo	Mobile
Nome	Utente accede alla piattaforma
Descrizione	Test che assicura il corretto funzionamento della funzione di Login
Note	L'utente non deve aver effettuato il login e deve esistere l'utente con nickname="nickname" e password "password"
Stato	Passato/Fallito

Continua alla pagina seguente

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Dall'homepage viene cliccata l'iconda del menù in alto a sinistra	Viene visualizzato il menù laterale sinsitra	Come ci si aspettava
2	Viene cliccata la riga contenente "Login"	Viene visualizzata la schermata di Login con il bottone "Login" 2 TextField	Come ci si aspettava
3	Non viene compilato nessun campo e viene premuto il tasto "Fine"	Viene mostrato un dialog di errore	Come ci si aspettava
4	Viene premuto il tasto "Ok"	Viene visualizzata nuovamente la schermata di Login	Come ci si aspettava
5	Vengono inseriti i campi: "nickname" e "password" e viene premuto il tasto "Login"	Viene visualizzata mostrata la homepage dell'applicativo	Come ci si aspettava

Tabella 6.8: Test Case #3

Applicativo	Mobile
Nome	Utente aggiunge una recensione
Descrizione	Test che assicura il corretto funzionamento dell'aggiunta di recensioni
Note	L'utente deve aver effettuato il login
Stato	Passato/Fallito

Continua alla pagina seguente

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene aperta la schermata di visualizzazione di una struttura	Viene visualizzato una schermata contenente tutti i dettagli della struttura	Come ci si aspettava
2	Viene cliccato il pulsante "Aggiungi Recensione"	Viene visualizzata la schermata di aggiunta recensione	Come ci si aspettava
3	Non viene compilato nessun campo e viene premuto il tasto "Fine"	Viene mostrato un dialog di errore	Come ci si aspettava
4	Viene premuto il tasto "Ok"	Viene visualizzata nuovamente la schermata di Login	Come ci si aspettava
5	Vengono inseriti i campi: "titolo", "testo", viene cliccata la prima stellina a sinistra e viene premuto il tasto "Aggiungi"	Viene visualizzato dialog che notifica l'avvenuta operazione.	Come ci si aspettava
6	Preme "Ok"	Viene visualizzato nuovamente la schermata della struttura	Come ci si aspettava

Tabella 6.9: Test Case #4

Applicativo	Mobile
Nome	Utente ricerca una struttura
Descrizione	Test che assicura il corretto funzionamento della ricerca di una struttura
Note	
Stato	Passato/Fallito

Continua alla pagina seguente

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene aperta l'homepage dell'applicazione '	Viene visualizzato una schermata contenente una mappa con tutte le strutture esistenti in database	Come ci si aspettava
2	Ci si sposta sulla mappa fino a visualizzare la struttura desiderata e ci si clicca sopra	Viene visualizzata la schermata della struttura	Come ci si aspettava
3	Viene cliccato il tasto back in alto a sinistra	Viene mostrata l'homepage	Come ci si aspettava
4	Viene cliccato il tasto filtri in alto a destra	Viene visualizzata la schermata filtri	Come ci si aspettava
5	Vengono inseriti i filtri: "Struttura", "Pompei", "Pub" e viene premuto il tasto "Cerca"	Viene visualizzata la lista delle strutture corrispondenti ai criteri di ricerca	Come ci si aspettava
6	Viene cliccato il tasto back in alto a sinistra	Viene mostrata l'homepage	Come ci si aspettava
7	Viene cliccato il tasto filtri in alto a destra	Viene visualizzata la schermata filtri	Come ci si aspettava
8	Viene cliccato il tasto prossimità, viene inserito "100" nel textfield "Distanza massima" viene cliccato il tasto "Cerca"	Viene visualizzata la schermata contenete la lista delle strutture entro 100km di prossimità	Come ci si aspettava

Tabella 6.10: Test Case #5

Applicativo	Mobile
Nome	Utente visualizza i dettagli di una struttura
Descrizione	Test che assicura il corretto funzionamento della visualizzazione di una struttura
Note	Esiste la struttura "struttura"
Stato	Passato/Fallito

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene ricercata la struttura "Struttura"	Viene visualizzato una lista di strutture con il nome "Struttura"	Come ci si aspettava
2	Viene cliccata sulla card della prima struttura	Viene visualizzata la schermata della struttura	Come ci si aspettava

Tabella 6.11: Test Case #6

Applicativo	Mobile
Nome	Utente visualizza i dettagli di una recensione
Descrizione	Test che assicura il corretto funzionamento della visualizzazione di una recensione
Note	Esiste una recensione alla struttura "struttura"
Stato	Passato/Fallito

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene aperta la pagina struttura "Struttura"	Vengono visualizzati i dati della struttura ed una lista di recensioni	Come ci si aspettava
2	Viene cliccata sulla card della prima recensione	Viene visualizzata la schermata contenente i dettagli della recensione	Come ci si aspettava

Tabella 6.12: Test Case #7

Applicativo	Mobile
Nome	Utente visualizza una struttura su mappa
Descrizione	Test che assicura il corretto funzionamento della visualizzazione su mappa di una struttura
Note	
Stato	Passato/Fallito

Step#	Input	Risultato Atteso	Risultato Ottenuto
1	Viene aperta la pagina struttura "Struttura"	Vengono visualizzati i dati della struttura ed una lista di recensioni	Come ci si aspettava
2	Viene cliccato sul bottone in basso a destra	Viene visualizzata l'icona della visualizzazione su mappa	Come ci si aspettava
3	Viene cliccata l'icona della visualizzazione su mappa	Viene aperta la schemata di visualizzazione su mappa che mostra la posizione della struttura sulla mappa.	Come ci si aspettava

6.2 Codice jUnit per unit testing

6.2.1 Testing Black Box

Effettueremo una serie di test, tramite strategia Strong Equivalence Class Testing sul seguente metodo:

```
public boolean tryLogin(String username, String password)
```

Definiamo le classi di equivalenza dei valori assumibili dai parametri in input:

Tabella 6.13: Classi di equivalenza di 'username'

Classe di Equivalenza	Descrizione
EC1	username è null
EC2	username è una stringa vuota
EC3	username è sbagliato
EC4	username è corretto

Tabella 6.14: Classi di equivalenza di 'password'

Classe di Equivalenza	Descrizione
EC5	password è null
EC6	password è una stringa vuota
EC7	password è sbagliata
EC8	password è corretta

Effettuiamo il prodotto cartesiano tra le classi di equivalenza del primo parametro e quelle del secondo parametro per ottenere tutti i possibili testcase.

Tabella 6.15: Lista dei TestCase

Test Case	Classe di Equivalenza	Descrizione
TC1	EC1	EC5
TC2	EC1	EC6
TC3	EC1	EC7
TC4	EC1	EC8
TC5	EC2	EC5
TC6	EC2	EC6
TC7	EC2	EC7
TC8	EC2	EC8
TC9	EC3	EC5
TC10	EC3	EC6
TC11	EC3	EC7
TC12	EC3	EC8
TC13	EC4	EC5
TC14	EC4	EC6
TC15	EC4	EC7
TC16	EC4	EC8

Tabella 6.16: Precondizioni

Precondizione	Input	Output Atteso
'admin' ha diritto d'accesso	"admin", "admin"	true
'sbagliato' non ha diritto d'accesso	"sbagliato" , "sbagliato"	false

Listing 6.1: Casi di test

```
public class AdminDAO_TestSuite {
        private static AdminDAO dao;

        @BeforeClass
        public static void setUpClass() {
             dao = new AdminDAO();
        }

        //Black-box SECT
        //TestCase 1 (EC1 - EC5)
        @Test
        public void tryLogin_EntrambeCredenzialiNull_test(){
            assertEquals( dao.tryLogin(null, null), false);
        }

        //TestCase 2 (EC1 - EC6)
        @Test
        public void tryLogin_usernameNull_passwordVuota_test(){
            assertEquals( dao.tryLogin(null, ""), false);
        }

        //TestCase 3 (EC1 - EC7)
        @Test
        public void tryLogin_usernameNull_passwordErrata_test(){
```

```java
            assertEquals( dao.tryLogin(null, "sbagliata"), false);
        }
        //TestCase 4 (EC1 - EC8)
        @Test
        public void tryLogin_usernameNull_passwordCorretta_test(){
            assertEquals( dao.tryLogin(null, "admin"), false);
        }

        //TestCase 5 (EC2 - EC5)
        @Test
        public void tryLogin_usernameVuoto_passwordNull_test(){
            assertEquals( dao.tryLogin("", null), false);
        }

        //TestCase 6 (EC2 - EC6)
        @Test
        public void tryLogin_usernameVuoto_passwordVuota_test(){
            assertEquals( dao.tryLogin("", ""), false);
        }

        //TestCase 7 (EC2 - EC7)
        @Test
        public void tryLogin_usernameVuoto_passwordErrata_test(){
            assertEquals( dao.tryLogin("admin", "sbagliata"), false);
        }

        //TestCase 8 (EC2 - EC8)
        @Test
        public void tryLogin_usernameVuoto_passwordCorretta_test(){
            assertEquals( dao.tryLogin(null, "admin"), false);
        }

        //TestCase 9 (EC3 - EC5)
        @Test
        public void tryLogin_usernameSbagliato_passwordNull_test(){
            assertEquals( dao.tryLogin("sbagliato!!!", null), false);
        }

        //TestCase 10 (EC3 - EC6)
        @Test
        public void tryLogin_usernameSbagliato_passwordVuota_test(){
            assertEquals( dao.tryLogin("sbagliato!!!", ""), false);
        }

        //TestCase 11 (EC3 - EC7)
        @Test
        public void tryLogin_usernameSbagliato_passwordErrata_test(){
            assertEquals( dao.tryLogin("sbagliato!!!", "sbagliata!"), false);
        }
        //TestCase 12 (EC3 - EC8)
        @Test
        public void tryLogin_usernameSbagliato_passwordCorretta_test(){
            assertEquals( dao.tryLogin("sbagliato!!!", "admin"), false);
        }
        //TestCase 13 (EC4 - EC5)
        @Test
        public void tryLogin_usernameCorretto_passwordNull_test(){
            assertEquals( dao.tryLogin("admin", null), false);
        }

        //TestCase 14 (EC4 - EC6)
        @Test
```

```java
        public void tryLogin_usernameCorretto_passwordVuota_test(){
            assertEquals( dao.tryLogin("admin", ""), false);
        }

        //TestCase 15 (EC4 - EC7)
        @Test
        public void tryLogin_usernameCorretto_passwordErrata_test(){
            assertEquals( dao.tryLogin("admin", "sbagliata!"), false);
        }

        //TestCase 16 (EC4 - EC8)
        @Test
        public void tryLogin_usernameCorretto_passwordCorretta_test(){
            assertEquals( dao.tryLogin("admin", "admin"), true);
        }
    }
```

6.2.2 Testing White box

Effueremo testing white box tramite Branch coverage della seguente funzione:

Listing 6.2: Funzione da testare

```
public String getFiltriNonVuoti(double longitudine, double latitudine) {
    String returnString = "";
    if (!nome.equals(""))
        returnString += "&nome=%25" + nome + "%25";
    if (!citta.equals(""))
        returnString += "&citta=%25" + citta + "%25";
    if (!categoria.equals("Nessuno"))
        returnString += "&categoria=" + categoria;
    if (!prezzo.equals("Nessuno"))
        returnString += "&prezzo=" + prezzo;
    if (!valutazione.equals("Nessuno"))
        returnString += "&valutazione_media=" + nome;
    if (prossimita) {
        if (!distanzaMassima.equals(""))
            returnString += "&radius=" + distanzaMassima;
        else
            returnString += "&radius=0";
        returnString += "&x=" + longitudine;
        returnString += "&y=" + latitudine;
    }
    return returnString;
}
```

2
3.if
True
False
4
5.if
True
False
6
7.if
True
False
8
9.if
True
False
10
11.if
True
False
12
13.if
True
False
14.if
True
False
15
17
18
19
21
F

Listing 6.3: Casi di test

```java
public class FiltriUnitTest {
    private static Filtri filtri;

    @Test
    public void getFiltriNonVuoti_BranchCoverage_2_3_5_7_9_11_13_21(){
        filtri = new Filtri("", "", "Nessuno", "Nessuno", "Nessuno","",
            false);
        assertEquals("", filtri.getFiltriNonVuoti(0,0));
    }
    @Test
    public void getFiltriNonVuoti_BranchCoverage_2_a_15_18_19_21(){
        filtri = new Filtri("a", "a", "a", "a", "1","1", true);
        String returnString = "&nome=%25a%25&citta=%25a%25&categoria=a&
            prezzo=a&valutazione_media=a&radius=1&x=0.0&y=0.0";
        assertEquals(returnString, filtri.getFiltriNonVuoti(0.0,0.0));
    }
    @Test
    public void getFiltriNonVuoti_BranchCoverage_2_a_14_17_18_19_21(){
        filtri = new Filtri("a", "a", "a", "a", "1","", true);
        String returnString = "&nome=%25a%25&citta=%25a%25&categoria=a&
            prezzo=a&valutazione_media=a&radius=0&x=0.0&y=0.0";
        assertEquals(returnString, filtri.getFiltriNonVuoti(0.0,0.0));
    }
}
```

Ramo 2-3-5-7-9-11-13-21

Il primo caso di test rappresenta una ricerca in cui l'utente non ha inserito nessun filtro e non ha attivato la ricerca per prossimità

Ramo 2-...-15-18-19-21

Il secondo caso di test rappresenta una ricerca in cui l'utente inserisce tutti i filtri ed ha attivato la ricerca per prossità

Ramo 2-...-14-17-18-19-21

Il terzo caso di test rappresenta una ricerca in cui l'utente inserisce tutti i filtri, ha attivato la ricerca per prossità e non ha inserito la distanza massima delle strutture rispetto alla posizione dell'utente.

In questo modo abbiamo effettuato sia un branch coverage completo che un path coverage dei cammini linearmente indipendenti.
Tutti gli altri cammini risultano essere combinazioni lineari dei cammini indipendenti già testati ed un loro test esaustivo richiederebbe 32*3 casi di test

Parte IV

Glossario