

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
3 Modello di Dominio	36
3.1 Classi, oggetti e relazioni di analisi	36
3.2 Diagrammi di sequenza di analisi	37
3.3 Diagrammi di stato di attività	37
3.4 Diagrammi di attività	38
II Documento di Design del sistema	39
4 Desgin	40
4.1 Descrizione del dominio	40
4.2 Analisi dell'architettura	40
4.2.1 Model-View-Presenter	41
4.3 Diagramma delle classi di design	42
4.4 CRC Cards	42
4.4.1 Applicativo Desktop	42

4.4.2	Applicativo Mobile	43
4.5	Diagramma di stato di design	43
4.6	Diagramma di sequenza di design	43
III	Documento di Testing del sistema	44
5	Testing	45
5.1	Test Plan per il System Testing	45
5.1.1	Applicativo Desktop	45
5.1.2	Applicativo Mobile	50
5.2	Codice jUnit per unit testing	55
5.2.1	Testing Black Box	55
5.2.2	Testing White box	59
	Conclusions	62

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 desktop).
2. **Amministratore valuta una recensione:** un amministratore 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 password) 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 struttura:** un utente dovrà poter visualizzare i dettagli di una struttura dopo averla selezionata con la possibilità 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 visualizza dettagli recensione:** un utente dovrà poter leggere i dettagli di una recensione, in particolare leggendo 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 effettuato 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'amministratore inserisce dati errati		
Step#	Amministratore	Sistema
1 a	Non compila 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 effettuato lo use case "Effettua Login".
Success End Conditions		L'amministratore valuta una recensione. Il sistema tiene traccia di tale operazione.
Failed End Conditions		L'amministratore 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'amministratore 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'amministratore 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 effettuato 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 effettuato 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 effettuato 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 text-Field 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 text-Field 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 registrazione

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 textfield 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 visualizza una struttura

Use Case #3		L'utente visualizza 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 visualizza i dettagli di una recensione

Use Case #4	L'utente visualizza 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 visualizza la posizione della struttura sulla mappa

Use Case #6	L'utente visualizza 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
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'icona dei filtri sulla schermata "Homepage"	
2		Mostra schermata "Ricerca Filtri"
3	Compila i textfield ed i combo-Box in base alle necessità della sua ricerca	
4	Preme il pulsante "Cerca"	
5		Mostra schermata "Lista Strutture" e termina caso d'uso

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

2.3 Mockup

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

Login



Dati errati Dialog



Server irraggiungibile Dialog



Campi vuoti Dialog



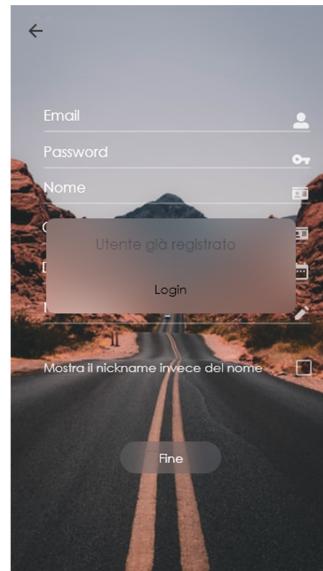
Registrazione



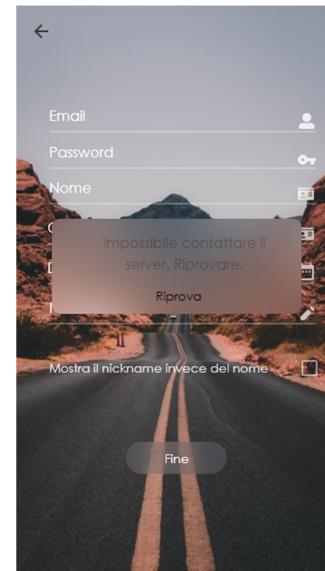
Registrazione success Dialog



Utente esistente fail dialog



Server irraggiungibile dialog



Homepage



Ricerca Filtri

← Ricerca per filtri

Nome	T
Città	filter icon
Categoria	Nessuno
Prezzo	Nessuno
Valutazione	Nessuno
Prossimità	<input checked="" type="checkbox"/>

Cerca

Pagina Struttura

← Informazioni

Nome struttura

Valutazione: ★★★★☆

Descrizione.
Lorem ipsum dolor sit amet,
soluta regione urbanitas vis
in, qui elit populo ut.

Città: Milano, Italia
Indirizzo: Via Nazionale, 23
Prezzo: medio
Categoria: Ristorante



"Titolo recensione"

Valutazione: ★★★★☆ Data: 2/10/2020

Testo recensione.
Lorem ipsum dolor sit amet, soluta
Da: nomeutente

"Titolo recensione"

Valutazione: ★★★★☆ Data: 2/10/2020

Testo recensione.
Lorem ipsum dolor sit amet, soluta
Da: nomeutente2



Lista Strutture

← Lista Strutture

Nome struttura

Città: Milano, Italia

Descrizione.
Lorem ipsum dolor sit amet,
soluta regione urbanitas vis



Categoria: Ristorante Valutazione: ★★★★☆

Nome struttura

Città: Sardegna, Italia

Descrizione.
Lorem ipsum dolor sit amet,
soluta regione urbanitas vis



Categoria: Hotel Valutazione: ★★★★☆

Nome struttura

Città: Napoli, Italia

Descrizione.
Lorem ipsum dolor sit amet,
soluta regione urbanitas vis



Categoria: Ristorante Valutazione: ★★★★☆

Floating Action Button Struttura



Dialog Filtri Recensione



Aggiungi recensione



Visualizza su mappa



Aggiungi recensione errore dialog

← Aggiungi Recensione

Titolo recensione

Testo recensione

Lorem ipsum dolor sit amet, consetetur
sadipscing elitr, sed diam nonumy eirmod
tempor invidunt ut labore et dolore magna
aliquyam erat, sed diam voluptua.

mag Impossibile aggiungere la
volu recensione.

OK

Valutazione ★★☆☆☆

Aggiungi recensione success dialog

← Aggiungi Recensione

Titolo recensione

Testo recensione

Lorem ipsum dolor sit amet, consetetur
sadipscing elitr, sed diam nonumy eirmod
tempor invidunt ut labore et dolore magna
aliquyam erat, sed diam voluptua.

mag Recensione aggiunta con
volu successo.

OK

Valutazione ★★☆☆☆

Aggiungi

Aggiungi

Aggiungi recensione empty dialog

← Aggiungi Recensione

Titolo recensione

Testo recensione

Valu Devi completare tutti i campi
per aggiungere la

OK

Aggiungi

Navigation Drawer utente loggato



CercaViaggi19

Login

Registrati



Navigation Drawer utente non loggato



Pagina recensione



Connessione Assente

← Informazioni

Impossibile caricare la pagina.



Verificare la connessione ad
internet

Login



Credenziali errate Dialog



Connessione assente dialog



Recensioni



Funzionalità

-  **Recensioni**
-  Visitatori
-  Logout

Recensioni

Utente	Struttura	Data
John Lemon	Nome struttura	JAN 8
Indiana Bond	Nome struttura	JAN 8
Angela Mer...	Nome struttura	JAN 8
Buris Johnson	Nome struttura	JAN 8
David Becham	Nome struttura	JAN 8
Angelina Jolie	Nome struttura	JAN 8

Visualizza Recensione



Funzionalità

-  **Recensioni**
-  Visitatori
-  Logout

John Lemon 

Nome struttura	Data
Nome struttura	JAN 8

Lore ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lore ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lore ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et

Rifiuta **Accetta**

Recensione approvata Dialog



Recensione rifiutata dialog



Fallimento Dialog



Visitatori



Funzionalità

 Recensioni

 Visitatori

 Logout

Visitatori

Nome	Nickname
John Lemon	HeyDude
Indiana Bond	L'ultimaBraciata
Angela Mer...	Wretzel
Buris Johnson	HerdImmunity
David Bacon	FryingPan98
Angelina Jolie	Beneficent

Visualizza Visitatore



Funzionalità

 Recensioni

 Visitatori

 Logout

John Lemon

Nome	John Lemon	Email	jhon.lemon@fruit.com
Nickname	HeyDude	Recensioni rifiutate	2
Data di nascita	17-11-1960	Recensioni approvate	20

[Elimina](#) [Modifica](#)

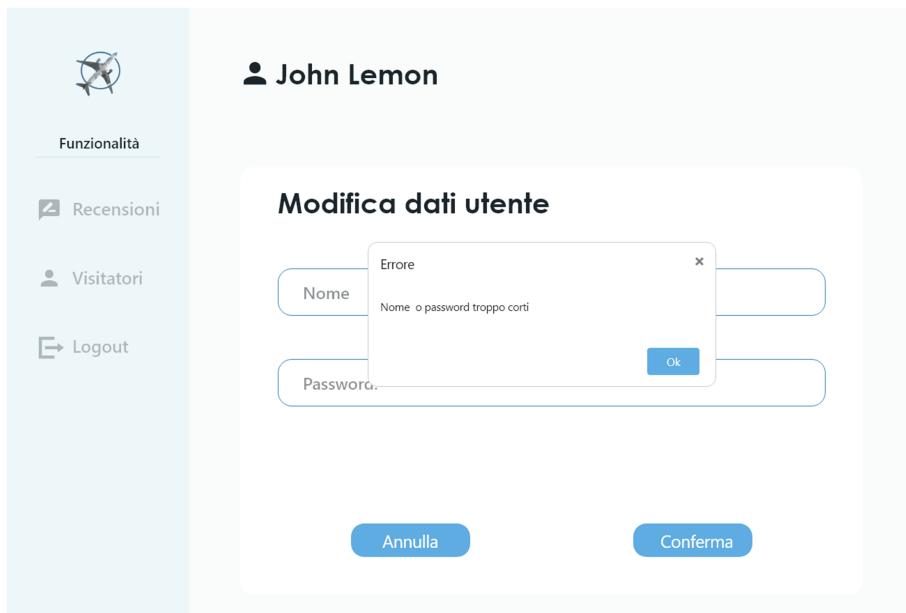
Modifica Visitatore



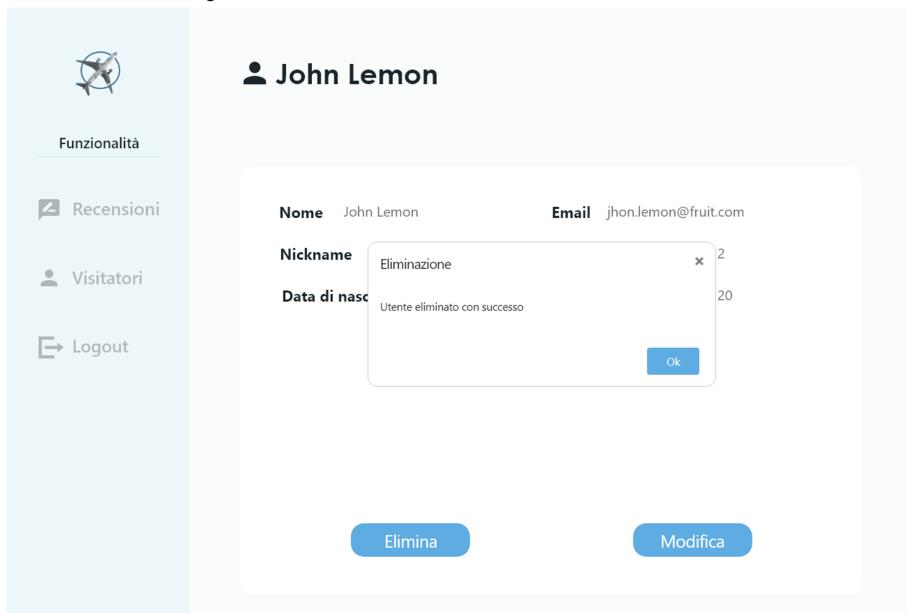
Modifica Visitatore success Dialog



Password o username errati Dialog



Utente eliminato dialog



Capitolo 3

Modello di Dominio

This chapter contains useful information for the preparation and the presentation of the master degree thesis for students of Electronic Engineering (M61), at the University of Study of Naples Federico II.

The final test for the Master Degree course in Electronic Engineering consists in the preparation and discussion of a thesis, written with the help of a supervisor (eventually with one or two co-supervisors). This work is the final result of the student career and it testifies his/her ability in exploring in deep the topics encountered during the degree course.

3.1 Classi, oggetti e relazioni di analisi

The supervisor is one of the professors that the candidate encountered during the degree course. Usually, the student finds its supervisor through informal talks, once provided that the professor is available and the student is interested in the professor's topics of interest. The degree course, on its website www.ingegneria-elettronica.unina.it, has defined a page with a non-exhaustive list of available theses topics, in order to facilitate the information exchange between students and professors.

In case the thesis is developed after an intra-moenia internship, among one of the laboratories of the department, the tutor that has already followed the student during the internship becomes the supervisor.

The supervisor defines the thesis topic. As already mentioned, the supervisor can be helped by one (or two maximum) co-supervisor. Supervisor and co-supervisor must guide and assist the student during the thesis development and also provide him all the needed methodological and practical instruments. During the thesis are usually foreseen periodical meetings of

the candidate with the supervisor, during which the ongoing work and the obtained results are discussed, also to define the future steps of the work.

3.2 Diagrammi di sequenza di analisi

Some theses can require part of the work to be done in a company. The degree course in Electronic Engineering promotes this kind of thesis, usually carried on as the culmination of an extra-moenia internship; in order to facilitate theses in a company, on the degree course website is also present a list of available intership available among companies in the electronic field.

For theses in a company, the supervisor from the University is necessarily complemented by a co-supervisor from the company.

The topic of the thesis is to identify accordingly to both the supervisor from the University and from the company, and also according to the goals of the degree course training. The co-supervisor from the company, in addition to the duties mentioned before, must also follow the activities of the student during the stay in the company, giving him/her the needed assistance. The company can ask to the student and the University supervisor to declare that some of the information and the material concerning the work is not to publish during the working period.

For the theses in company too, during the activities, are generally foreseen meetings between the candidate, the supervisor from the Univeristy and the co-supervisor from the company, during which the results and future steps are discussed.

3.3 Diagrammi di stato di attività

The redaction of the thesis has to be carried on by the candidate independently. A dissertation type thesis has the structure of a scientific article where it is required to derive, from the international literature, the most recent developments on the topic of interest, it is required to synthesize them, present them in an omogenous way, and finally compare the different approaches highlighting pros and cons of each of them. A experimental type thesis has the structure of a scientific report, it faces a specific problem, typically within a more wide project of interest for the supervisor, proposing a solution that is innovative if compared to the state of the art. A experimental thesis also includes a validation of the proposed solution, made by means of experimental measurements and/or numerical simulations.

3.4 Diagrammi di attività

During the thesis discussion, the candidate has at his/her disposal **12 minutes** for the final presentation. The 12 minutes limit is imperative and the committee chairman will take care of the observance of this limit. Thus, the candidate must pay attention in synthesizing in a proper way the done work.

For the final presentation, the candidate has to use a *PowerPoint presentation*. For the time limit, the presentation must include a limited number of slides (more than 15 are not suggested!) and focus the attention on the main aspects of the thesis:

- the faced problem
- the state of the art
- the adopted methodologies
- the obtained results
- other...

highlighting, if it is the case, the personal contribution to the innovation. All the details are not essential and digressions are to avoid.

It is important that the presentation is accurately organized and proved, and that the student expose its work in a clear way to the committee. At the end of the presentation, the committee could also ask clarifications or curiosities to the candidate.

Parte II

Documento di Design del sistema

Capitolo 4

Desgin

4.1 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

4.2 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.

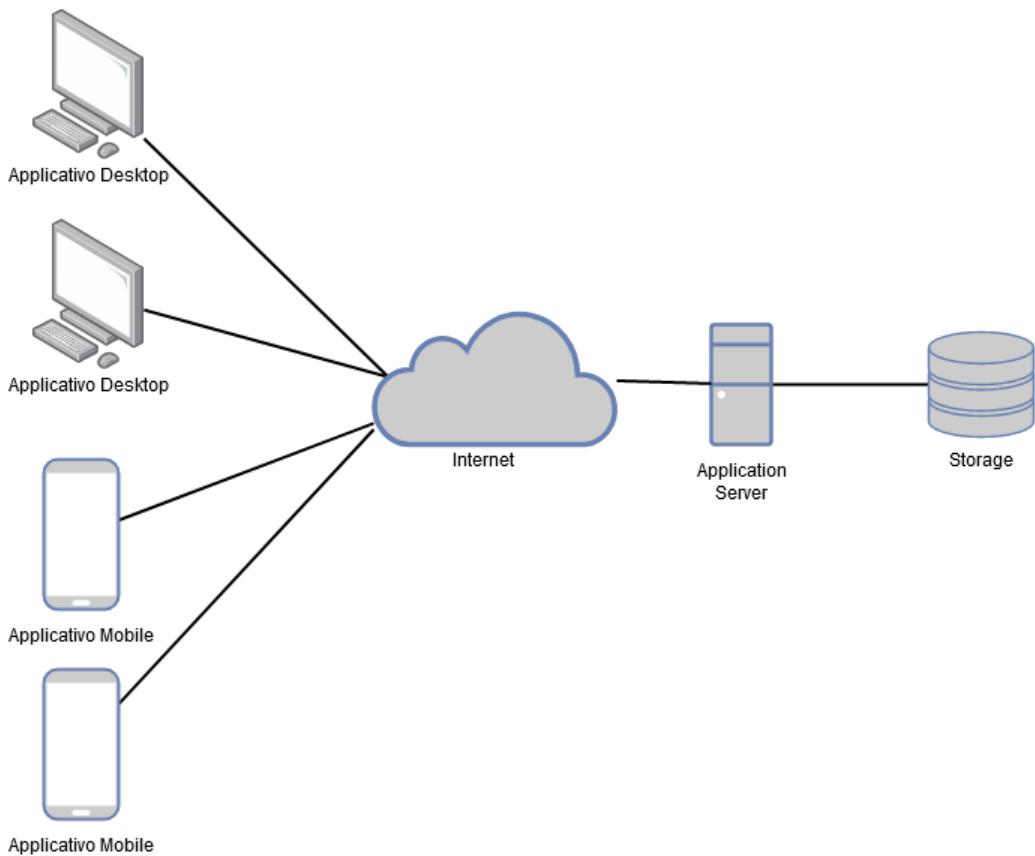


Figura 4.1: Architettura client server

4.2.1 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 manutenibilità.

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

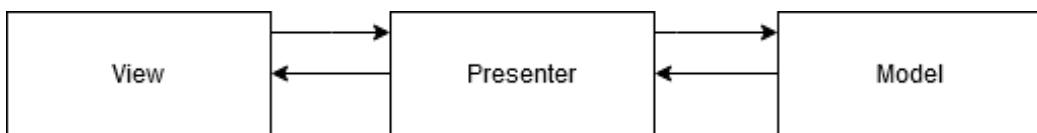


Figura 4.2: Architettura Model-View-Controller

4.3 Diagramma delle classi di design

4.4 CRC Cards

4.4.1 Applicativo Desktop

Nome Classe	AdminDAO
Superclassi	-
Sottoclassi	-
Responsabilità	Collaboratore
Effettuare le operazioni CRUD per l'entità Admin	-

Nome Classe	AdminDAO
Superclassi	-
Sottoclassi	-
Responsabilità	Collaboratore
Effettuare le operazioni CRUD per l'entità Admin	-

4.4.2 Applicativo Mobile

Nome Classe	RecensioneDAO
Superclassi	-
Sottoclassi	-
Responsabilità	Collaboratore
Effettuare le operazioni CRUD per l'entità Recensione	-

4.5 Diagramma di stato di design

4.6 Diagramma di sequenza di design

Parte III

Documento di Testing del sistema

Capitolo 5

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.

5.1 Test Plan per il System Testing

5.1.1 Applicativo Desktop

Tabella 5.1: 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" contenente 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 5.2: 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" contenente una lista di visitatori	Come ci si aspettava
2	Viene cliccata la riga contenente l'utente con nickname "utente"	Viene visualizzata la schermata utente contiene i tasti "Modifica", "Elimina" ed i textfield nome, nickname, data di nascita, email, recensioni rifiutate ed approvate contenenti i relativi dati.	Come ci si aspettava

Tabella 5.3: Test Case #3

Applicativo	Desktop
Nome	Amministratore valuta una recensione
Descrizione	Test che assicura il corretto funzionamento della valutazione delle recensioni
Note	L'amministratore 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" contenente 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 contenente i tasti "Accetta" e "Rifiuta", i textFiel contenenti il titolo, il testo e l'autore della descrizione. Viene visualizzato, inoltre, 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 5.4: Test Case #4

Applicativo	Desktop
Nome	Amministratore elimina visitatore
Descrizione	Test che assicura il corretto funzionamento dell'eliminazione di un visitatore'
Note	L'amministratore 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" contenente 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 text-field: 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 nuovamente la schermata con la lista dei visitatori	Come ci si aspettava

Tabella 5.5: 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'amministratore 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" contenente 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 schermata di modifica dei dati utente contenente due editText ("Nickname" e "Password") e due buttoni ("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

5.1.2 Applicativo Mobile

Tabella 5.6: 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'icona del menù in alto a sinistra	Viene visualizzato il menu 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 5.7: 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'icona del menù in alto a sinistra	Viene visualizzato il menu 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 5.8: 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 5.9: 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'home-page 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'home-page	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'home-page	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 text-field "Distanza massima" viene cliccato il tasto "Cerca"	Viene visualizzata la schermata contenente la lista delle strutture entro 100km di prossimità	Come ci si aspettava

Tabella 5.10: 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 5.11: 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 5.12: 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 schematica di visualizzazione su mappa che mostra la posizione della struttura sulla mappa.	Come ci si aspettava

5.2 Codice jUnit per unit testing

5.2.1 Testing Black Box

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

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

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

Tabella 5.13: 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 5.14: 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 5.15: 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 5.16: Precondizioni

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

Listing 5.1: Casi di test

```

1  public class AdminDAO_TestSuite {
2      private static AdminDAO dao;
3
4      @BeforeClass
5      public static void setUpClass() {
6          dao = new AdminDAO();
7      }
8
9      //Black-box SECT
10     //TestCase 1 (EC1 - EC5)
11     @Test
12     public void tryLogin_EntrambeCredenzialiNull_test(){
13         assertEquals( dao.tryLogin(null, null), false);
14     }
15
16     //TestCase 2 (EC1 - EC6)
17     @Test
18     public void tryLogin_usernameNull_passwordVuota_test(){
19         assertEquals( dao.tryLogin(null, ""), false);
20     }
21
22     //TestCase 3 (EC1 - EC7)
23     @Test
24     public void tryLogin_usernameNull_passwordErrata_test(){

```

```

25         assertEquals( dao.tryLogin(null, "sbagliata"), false);
26     }
27     //TestCase 4 (EC1 - EC8)
28     @Test
29     public void tryLogin_usernameNull_passwordCorretta_test(){
30         assertEquals( dao.tryLogin(null, "admin"), false);
31     }
32
33     //TestCase 5 (EC2 - EC5)
34     @Test
35     public void tryLogin_usernameVuoto_passwordNull_test(){
36         assertEquals( dao.tryLogin("", null), false);
37     }
38
39     //TestCase 6 (EC2 - EC6)
40     @Test
41     public void tryLogin_usernameVuoto_passwordVuota_test(){
42         assertEquals( dao.tryLogin("", ""), false);
43     }
44
45     //TestCase 7 (EC2 - EC7)
46     @Test
47     public void tryLogin_usernameVuoto_passwordErrata_test(){
48         assertEquals( dao.tryLogin("admin", "sbagliata"), false);
49     }
50
51     //TestCase 8 (EC2 - EC8)
52     @Test
53     public void tryLogin_usernameVuoto_passwordCorretta_test(){
54         assertEquals( dao.tryLogin(null, "admin"), false);
55     }
56
57     //TestCase 9 (EC3 - EC5)
58     @Test
59     public void tryLogin_usernameSbagliato_passwordNull_test(){
60         assertEquals( dao.tryLogin("sbagliato!!!", null), false);
61     }
62
63     //TestCase 10 (EC3 - EC6)
64     @Test
65     public void tryLogin_usernameSbagliato_passwordVuota_test(){
66         assertEquals( dao.tryLogin("sbagliato!!!", ""), false);
67     }
68
69     //TestCase 11 (EC3 - EC7)
70     @Test
71     public void tryLogin_usernameSbagliato_passwordErrata_test(){
72         assertEquals( dao.tryLogin("sbagliato!!!", "sbagliata!"), false);
73     }
74     //TestCase 12 (EC3 - EC8)
75     @Test
76     public void tryLogin_usernameSbagliato_passwordCorretta_test(){
77         assertEquals( dao.tryLogin("sbagliato!!!", "admin"), false);
78     }
79     //TestCase 13 (EC4 - EC5)
80     @Test
81     public void tryLogin_usernameCorretto_passwordNull_test(){
82         assertEquals( dao.tryLogin("admin", null), false);
83     }
84
85     //TestCase 14 (EC4 - EC6)
86     @Test

```

```
87     public void tryLogin_usernameCorretto_passwordVuota_test(){
88         assertEquals( dao.tryLogin("admin", ""), false);
89     }
90
91     //TestCase 15 (EC4 - EC7)
92     @Test
93     public void tryLogin_usernameCorretto_passwordErrata_test(){
94         assertEquals( dao.tryLogin("admin", "sbagliata!"), false);
95     }
96
97     //TestCase 16 (EC4 - EC8)
98     @Test
99     public void tryLogin_usernameCorretto_passwordCorretta_test(){
100        assertEquals( dao.tryLogin("admin", "admin"), true);
101    }
102 }
```

5.2.2 Testing White box

Effueremo testing white box tramite Branch coverage della seguente funzione:

Listing 5.2: Funzione da testare

```
1 public String getFiltriniVuoti(double longitudine, double latitudine) {
2     String returnString = "";
3     if (!nome.equals(""))
4         returnString += "&nome=%25" + nome + "%25";
5     if (!citta.equals(""))
6         returnString += "&citta=%25" + citta + "%25";
7     if (!categoria.equals("Nessuno"))
8         returnString += "&categoria=" + categoria;
9     if (!prezzo.equals("Nessuno"))
10        returnString += "&prezzo=" + prezzo;
11    if (!valutazione.equals("Nessuno"))
12        returnString += "&valutazione_media=" + nome;
13    if (prossimita) {
14        if (!distanzaMassima.equals(""))
15            returnString += "&radius=" + distanzaMassima;
16        else
17            returnString += "&radius=0";
18        returnString += "&x=" + longitudine;
19        returnString += "&y=" + latitudine;
20    }
21    return returnString;
22 }
```

Listing 5.3: Casi di test

```
1 public class FiltriUnitTests {
2     private static Filtri filtri;
3
4     @Test
5     public void getFiltriniVuoti_BranchCoverage_2_3_5_7_9_11_13_21(){
6         filtri = new Filtri("", "", "Nessuno", "Nessuno", "Nessuno", "", false);
7         assertEquals("", filtri.getFiltriniVuoti(0,0));
8     }
9     @Test
10    public void getFiltriniVuoti_BranchCoverage_2_a_15_18_19_21(){
11        filtri = new Filtri("a", "a", "a", "a", "1", "1", true);
12        String returnString = "&nome=%25a%25&citta=%25a%25&categoria=a&prezzo=a&valutazione_media=a&radius=1&x=0.0&y=0.0";
13        assertEquals(returnString, filtri.getFiltriniVuoti(0.0,0.0));
14    }
15    @Test
16    public void getFiltriniVuoti_BranchCoverage_2_a_14_17_18_19_21(){
17        filtri = new Filtri("a", "a", "a", "a", "1", "", true);
18        String returnString = "&nome=%25a%25&citta=%25a%25&categoria=a&prezzo=a&valutazione_media=a&radius=0&x=0.0&y=0.0";
19        assertEquals(returnString, filtri.getFiltriniVuoti(0.0,0.0));
20    }
21 }
```

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

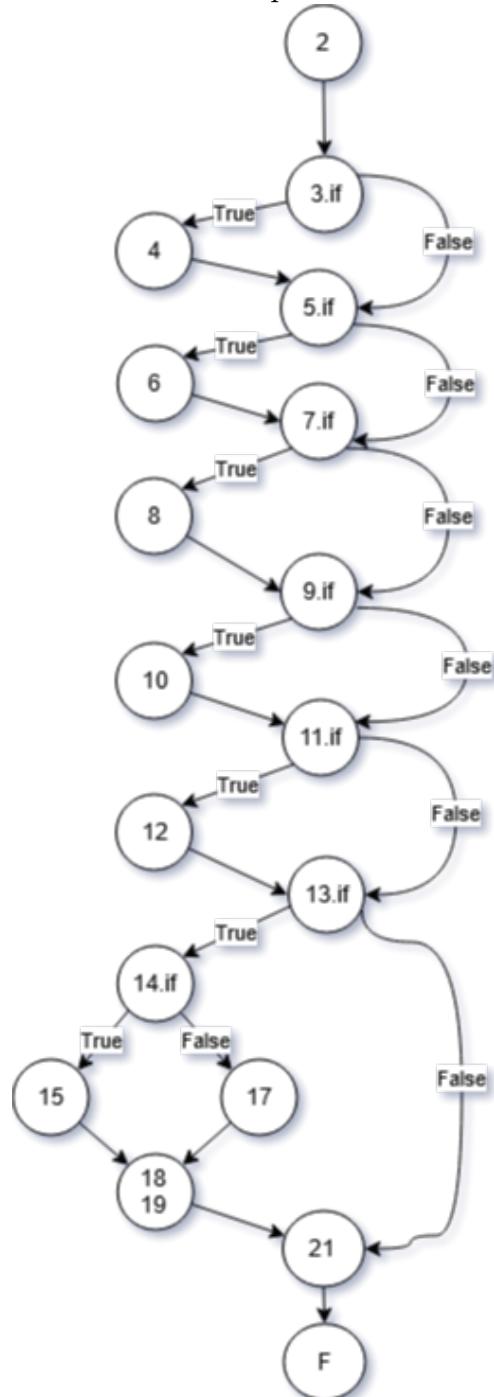
Ramo 2-...-14-17-18-19-21

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

Figura 5.1: Grafo di copertura della funzione



Conclusions

The conclusion of the thesis has to sum up the main considerations and results of the whole work, eventually addressing future steps to continue the work on the discussed topic.