Esempio di documentazione

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 3

2 Analisi 4

2.1 Analisi del dominio 4

2.2 Analisi e specifica dei requisiti 4

2.3 Use case 7

2.4 Pianificazione 7

2.5 Analisi dei mezzi 7

2.5.1 Software 7

2.5.2 Hardware 7

3 Progettazione 8

3.1 Design dell’architettura del sistema 8

3.2 Design dei dati e database 8

3.3 Design delle interfacce 8

3.4 Design procedurale 8

4 Implementazione 9

5 Test 9

5.1 Protocollo di test 9

5.2 Risultati test 10

5.3 Mancanze/limitazioni conosciute 10

6 Consuntivo 10

7 Conclusioni 10

7.1 Sviluppi futuri 10

7.2 Considerazioni personali 10

8 Bibliografia 10

8.1 Bibliografia per articoli di riviste: 10

8.2 Bibliografia per libri 10

8.3 Sitografia 11

9 Allegati 11

# Introduzione

## Informazioni sul progetto

Allievi: Andrea Frati, Luca Fumasoli, Manuel Grosso

Classi: I3AC / I3AA

Docente responsabile: Guido Montalbetti

Data inizio: -

Data fine: -

## Abstract

## Situazione iniziale

*Lo scopo di questo progetto è di creare un’applicazione web che possa aiutare le persone a trovare in modo semplice un viaggio che vogliono percorrere ad una data ed orario voluto. Una volta trovato il viaggio ricercato dovrà metterli a disposizione un metodo per pagare all’interno del sito con la scelta di un posto libero desiderato.*

*Quindi il risultato atteso è un sito per facilitare la ricerca e la riservazione dei viaggi alle persone.*

## Attuazione

Per questo progetto sarà necessario avere un account amministratore per creare viaggi ed i mezzi per i viaggi, e ci sarà bisogno di account per gli studenti che possono visualizzare tutti i viaggi e riservare posti sui viaggi desiderati. Per realizzare il sito abbiamo usato i principali linguaggi web, ossia HTML, CSS, PHP e Javascript.

## Risultati

*Breve analisi dei risultati ottenuti (obiettivi raggiunti, …), …*

## Scopo

Lo scopo del progetto è di creare un sito facile da navigare su cui riservare biglietti per viaggi che si è interessati a fare.

# Analisi

## Analisi del dominio

I viaggi all’interno del sito potranno essere visualizzati da chiunque e se sono interessati potranno creare un account per riservare un biglietto. Ci saranno anche degli account di amministratore che potranno creare viaggi ed i mezzi su cui percorrere i viaggi.

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **ID: REQ-01** | |
| **Nome** | Creazione di un database |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |

|  |  |
| --- | --- |
| **ID: REQ-02** | |
| **Nome** | Gestione di differenti livelli di utente |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | 3 livelli, amministratore, utente registrato e utente non registrato con annesso il salvataggio nel database  Ogni tipo di utente ha interfacce grafiche diverse. |
| **Sotto requisiti** | |
| **021** | Possibilità di registrarsi |
| **022** | Possibilità di fare login |
| **023** | Possibilità di distinguere i vari tipi di utente |

|  |  |
| --- | --- |
| **ID: REQ-03** | |
| **Nome** | Interfacce grafiche differenti per i 3 tipi di utente |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Amministratore:   * Tutto quello che può fare un utente loggato * Gestire i partecipanti al tragitto * Stampare la lista dei partecipanti e filtrarla * Possibilità di stampare una fattura da inviare ai partecipanti o inviare un’email * Creare nuovi tragitti * Creare nuovi autobus * Inviare email agli utenti registrati con filtri   Utente registrato:   * Vedere i tragitti * Filtrare i tragitti e ordinarli * Vedere disponibilità di posti in un viaggio * Selezionare e prenotare uno o più posti in un viaggio   Utente non registrato:   * Vedere i tragitti * Filtrare i tragitti e ordinarli * Vedere disponibilità di posti in un viaggio * Registrarsi al sito * Fare login |
| **Sotto requisiti** | |
| **031** | Poter differenziare i 3 differenti utenti. |

|  |  |
| --- | --- |
| **ID: REQ-04** | |
| **Nome** | Inserire nel database un tragitto |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Tragitto composto da:   * Località di partenza * Località di arrivo * Data partenza * Data arrivo * Fermate e partenze intermedie con gli orari * Costo del tragitto da partenza a arrivo * Costo per le differenti tappe |
| **Sotto requisiti** | |
| **041** | Possibilità di salvare e leggere le tratte su un database |

|  |  |
| --- | --- |
| **ID: REQ-05** | |
| **Nome** | Controllare e gestire i partecipanti dei viaggi |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | L’amministratore può vedere tutti i partecipanti e cambiare i dati se non sono corretti. |
| **Sotto requisiti** | |
| **051** | Creazione dei viaggi |
| **052** | Utenti devono poter partecipare ai viaggi |
| **053** | Poter leggere dal database chi partecipa ai viaggi |

|  |  |
| --- | --- |
| **ID: REQ-06** | |
| **Nome** | Stampare la lista di partecipanti |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | L’amministratore può stampare la lista dei partecipanti che può essere divisa per località  La lista indica per uno specifico luogo, dando orario e partenza, chi sale o scende e quale posto ha o aveva |
| **Sotto requisiti** | |
| **061** | Sistema di filtraggio per località |
| **062** | Stampare i dati dei partecipanti |

|  |  |
| --- | --- |
| **ID: REQ-07** | |
| **Nome** | Sistema filtrabile di mass mail |
| **Priorità** | 3 |
| **Versione** | 1.0 |
| **Note** | Mail da mandare in contemporanea a tutti i partecipanti selezionati. |
| **Sotto requisiti** | |
| **072** | Poter selezionare gli utenti |

|  |  |
| --- | --- |
| **ID: REQ-8** | |
| **Nome** | Gestione dei pagamenti |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | L’amministratore può vedere tutte le fatture degli utenti |
| **Sotto requisiti** | |
| **001** | Fattura del viaggio |
| **002** | Opzione di stampa fattura da inviare ai partecipanti |
| **003** | Opzione di invio email di fattura ai partecipanti |

|  |  |
| --- | --- |
| **ID: REQ-09** | |
| **Nome** | Ricerca tragitto |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Ci deve essere la possibilità di cercare i tragitti esistenti tramite inserimento di filtri: - orario di partenza - luogo di partenza e luogo di destinazione |
| **Sotto requisiti** | |
| **091** | Filtro sulla ricerca |

|  |  |
| --- | --- |
| **ID: REQ-10** | |
| **Nome** | Registrazione al sito |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Ci deve essere la possibilità di registrarsi al sito inserendo un’email e una password che poi verranno salvate nel database |
| **Sotto requisiti** | |
| **101** | Controllo della mail |
| **102** | Criptaggio delle password |

|  |  |
| --- | --- |
| **ID: REQ-11** | |
| **Nome** | Log-in al sito |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Ci deve essere la possibilità di fare il log-in al sito inserendo il proprio email e la propria password |
| **Sotto requisiti** | |
| **111** | Bisogna essere già registrati nel sito, quindi avere email e password salvati nel database |

|  |  |
| --- | --- |
| **ID: REQ-12** | |
| **Nome** | Prenotazione biglietto |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Se si è loggati al sito, dopo aver selezionato una tratta si può prenotare uno o più posti |
| **Sotto requisiti** | |
|  | Bisogna essere già registrati nel sito, quindi avere email e password salvati nel database |
|  | Bisogna avere la possibilità di selezionare una tratta |

**Spiegazione elementi tabella dei requisiti:**

**ID**: identificativo univoco del requisito

**Nome**: breve descrizione del requisito

**Priorità**: indica l’importanza di un requisito nell’insieme del progetto, definita assieme al committente. Ad esempio, poter disporre di report con colonne di colori diversi ha priorità minore rispetto al fatto di avere un database con gli elementi al suo interno. Solitamente si definiscono al massimo di 2-3 livelli di priorità.

**Versione**: indica la versione del requisito. Ogni modifica del requisito avrà una versione aggiornata.

Sulla documentazione apparirà solamente l’ultima versione, mentre le vecchie dovranno essere inserite nei diari.

**Note**: eventuali osservazioni importanti o riferimenti ad altri requisiti.

**Sotto requisiti**: elementi che compongono il requisito.

## Use case

I casi d’uso rappresentano l’interazione tra i vari attori e le funzionalità del prodotto.

## Pianificazione

Prima di stabilire una pianificazione bisogna avere almeno una vaga idea del modello di sviluppo che si intende adottare. In questa sezione bisognerà inserire il modello concettuale di sviluppo che si seguirà durante il progetto. Gli elementi di riferimento per una buona pianificazione derivano da una scomposizione top-down della problematica del progetto.

La pianificazione può essere rappresentata mediante un diagramma di Gantt:

|  |
| --- |
| Figura 2: Esempio di diagramma di Gantt. |

Se si usano altri metodi di pianificazione (p.es. scrum), dovranno apparire in questo capitolo.

## Analisi dei mezzi

### Software

I software che abbiamo usato sono:

* Microsoft office professional plus 2019:
  + Microsoft word: per fare la documentazione
  + Microsoft visio: per fare lo schema del sito e dei database
  + Microsoft Project: per fare il diagramma di Gantt
* Visual Studio Code versione 1.58.2
* XAMPP versione 3.3.0
* Google Chrome versione 96.0.4664.110
* La libreria …

### Hardware

Come hardware abbiamo usato solo i computer scolastici.

# Progettazione

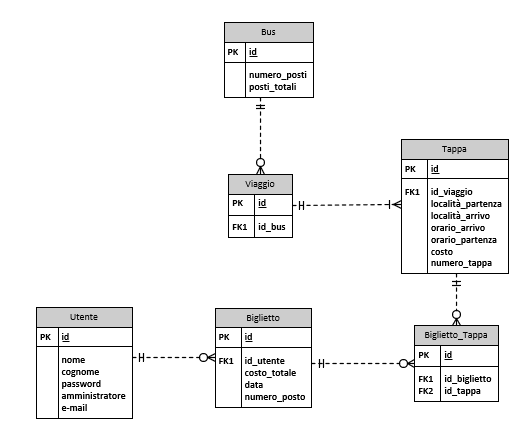
## Design dell’architettura del sistema

Descrive:

* La struttura del programma/sistema lo schema di rete...
* Gli oggetti/moduli/componenti che lo compongono.
* I flussi di informazione in ingresso ed in uscita e le relative elaborazioni. Può utilizzare *diagrammi di flusso dei dati* (DFD).
* Eventuale sitemap

## Design dei dati e database

### Diagramma E/R



Questo diagramma E/R mostra tutte le tabelle del database del nostro progetto con le primary e foreign key.

Tutti gli attributi in ogni tabella del database sono obbligatori alla creazione di un nuovo record ed ogni tabella ha un id come primary key.

#### Utente

I campi contenuti nell’entità utente sono il nome ed il cognome dell’utente, la password dell’account, l’email ed un booleano amministratore che con un valore di 1 rende l’account un amministratore e con il valore di 0 un utente normale.

#### Biglietto

L‘entità biglietto contiene una foreign key per sapere che utente ha acquistato il biglietto, un attributo per il costo totale del biglietto, uno per la data del viaggio ed un ultimo per sapere che posto è stato prenotato dall’utente.

#### Bus

L’entità bus contiene solo 2 campi oltre la primary key, il numero di posti totali del bus ed il numero della tratta che percorre il bus.

#### Viaggio

L‘entità viaggio ha solo la propria primary key ed una foreign key per sapere che bus percorre questo viaggio.

#### Tappa

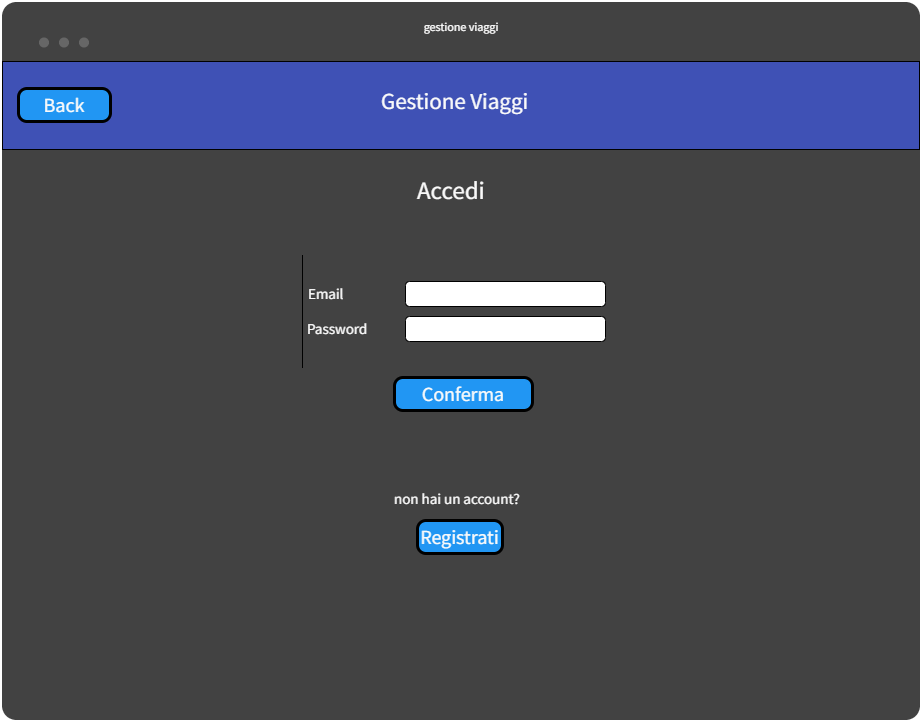
L‘entità tappa contiene tutte le fermate di un viaggio. Ha un attributo per la località di partenta ed un altro per quella di arrivo, e ne ha altri due per sapere a che ora parte dalla località di partenza ed a che ora arriva alla destinazione. Ha anche un campo per il costo della tappa, uno per il numero della tappa ed una foreign key per sapere di che viaggio è la tappa,

#### Biglietto\_Tappa

La tabella biglietto\_tappa è una relazione tra biglietto e tappa e contiene solo due foreign key, ossia l’id del biglietto riservato per la tappa ed un id per sapere quale è la tappa. Questa tabella sarà popolata da vari record per ogni biglietto per sapere su quali tappe vale il biglietto.

## Design delle interfacce

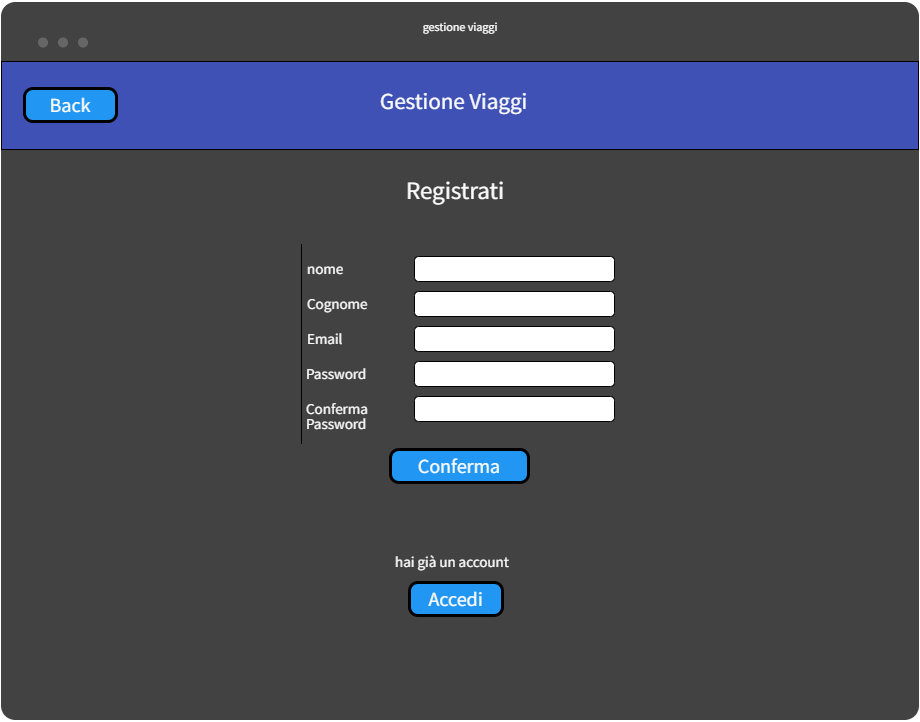
### Pagina di login



Pagina per accedere al proprio account immettendo la propria e-mail e password. Se non si ha creato un account si può premere su registrati per creare un account.

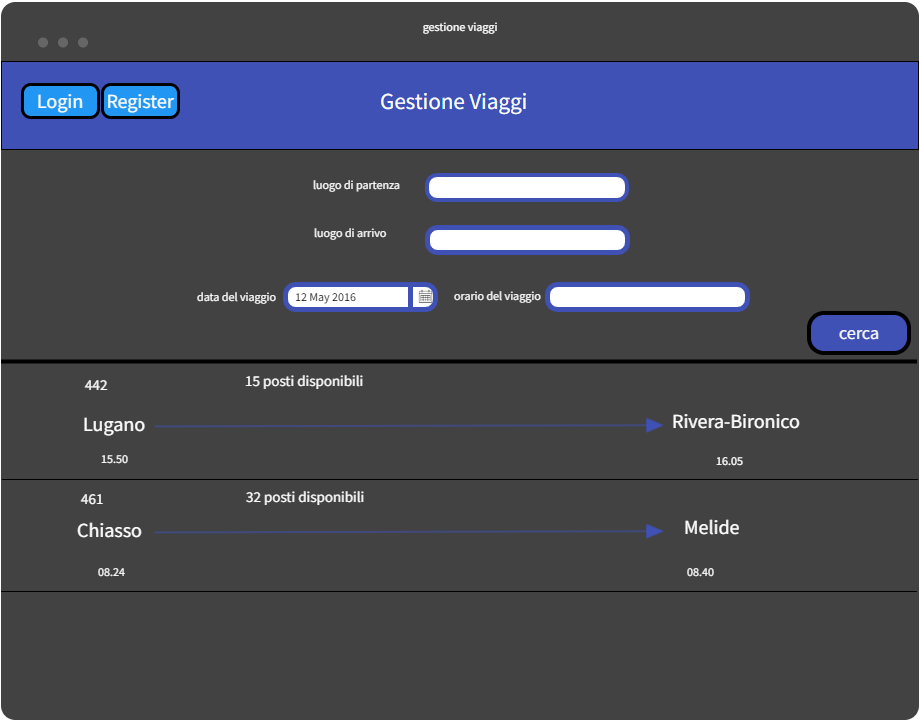
login.

### Pagina per registrarsi



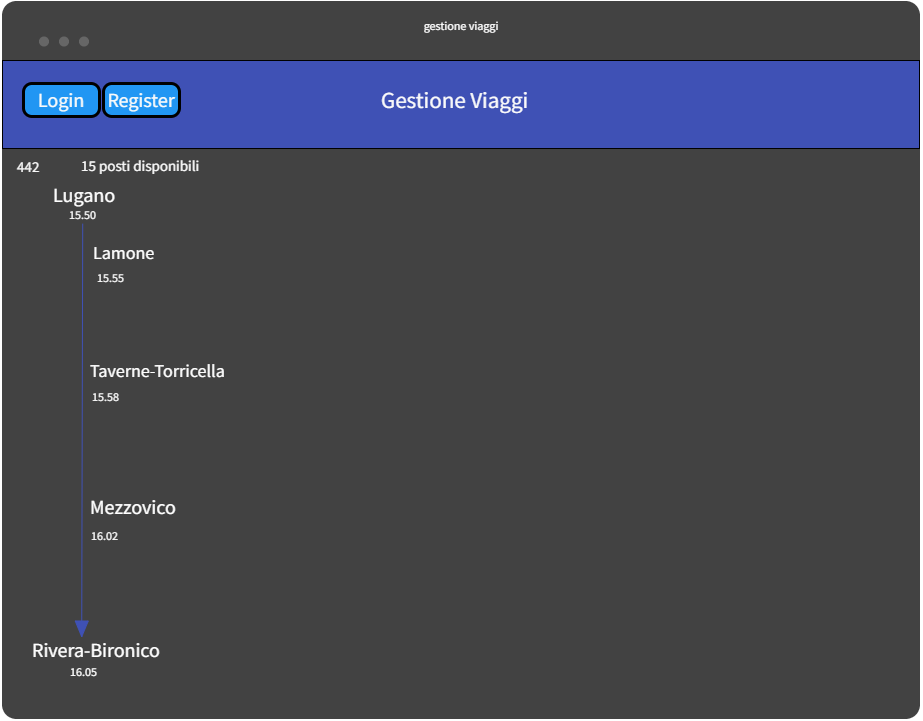
Pagina per creare il proprio account immettendo nome, cognome, E-mail e 2 volte la password per essere sicuri che non si fanno errori e poi premendo su conferma viene creato l’account. Se si ha già un account si può premere su accedi e ti porta alla pagina di login.

### Pagina ricerca viaggi



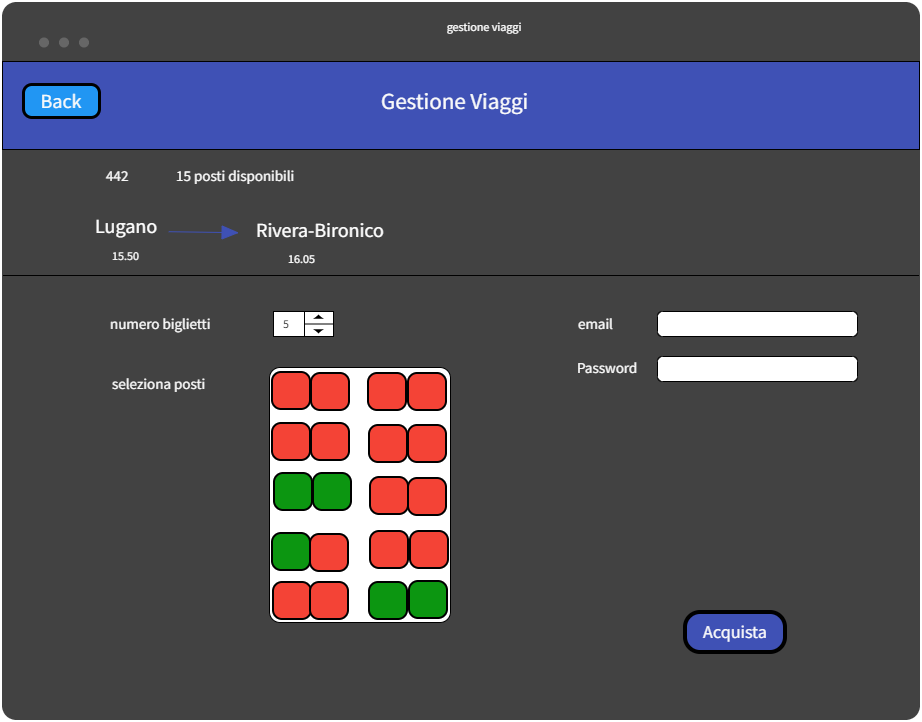
Pagina per cercare viaggi immettendo il luogo di partenza, il luogo di arrivo e la data ed orario in cui si vuole fare il viaggio. Poi premendo su cerca mostrerà tutti i viaggi previsti lungo quelle due località con il numero della linea, il numero di posti disponibili e l’orario in cui parte ed arriva a destinazione.  
Premendo sopra le tratte porterà ad una pagina con sui tutte le informazioni della tratta.

### Pagina mostra tratta



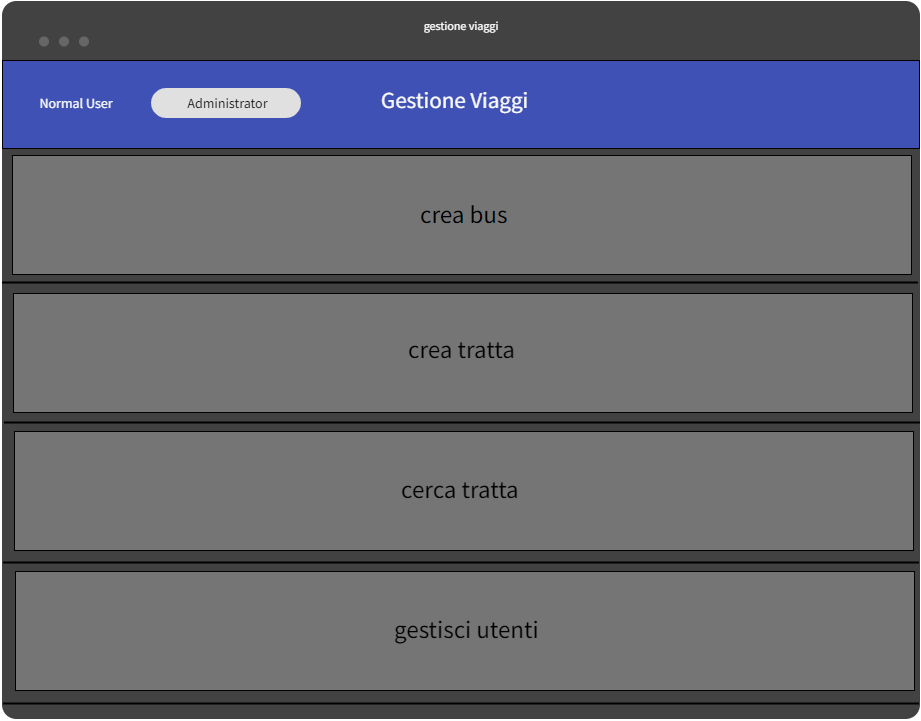
Pagina per mostra tutte le tappe in cui passa la tratta con l’orario in cui ci arriva e su in alto il numero della tratta ed il numero di posti disponibili. Se si è loggati nel proprio account allora apparirà anche un bottone acquista a destra della tratta che premendoci sopra porta alla pagina per acquistare biglietti per la tratta.

### Pagina acquista biglietti



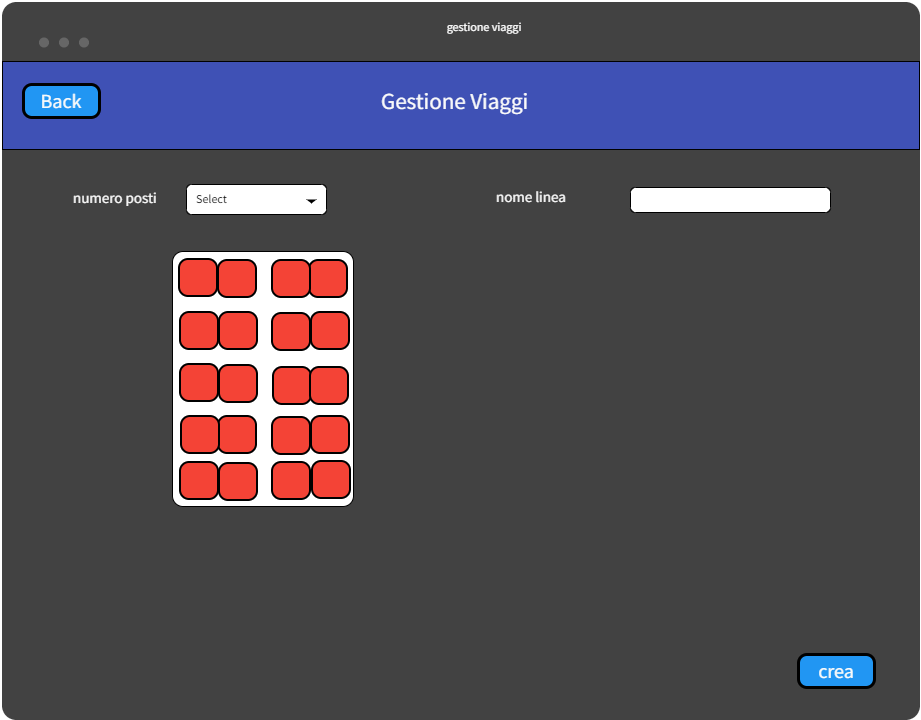
Pagina per acquistare biglietti accessibile solo se si è loggati nel sito. In alto viene mostrato il numero della tratta con i posti disponibili di fianco con sotto la località e l’orario di partenza con la località e l’orario di arrivo.  
Sotto ci sarà una tabella con su tutti i posti del bus in cui quelli colorati di rosso sono già stati riservati da altri utenti e quelli verdi sono riservabili premendo sopra a quelli che si vogliono riservare e poi premendo sopra acquista vengono acquistati.

### Pagina home amministratore



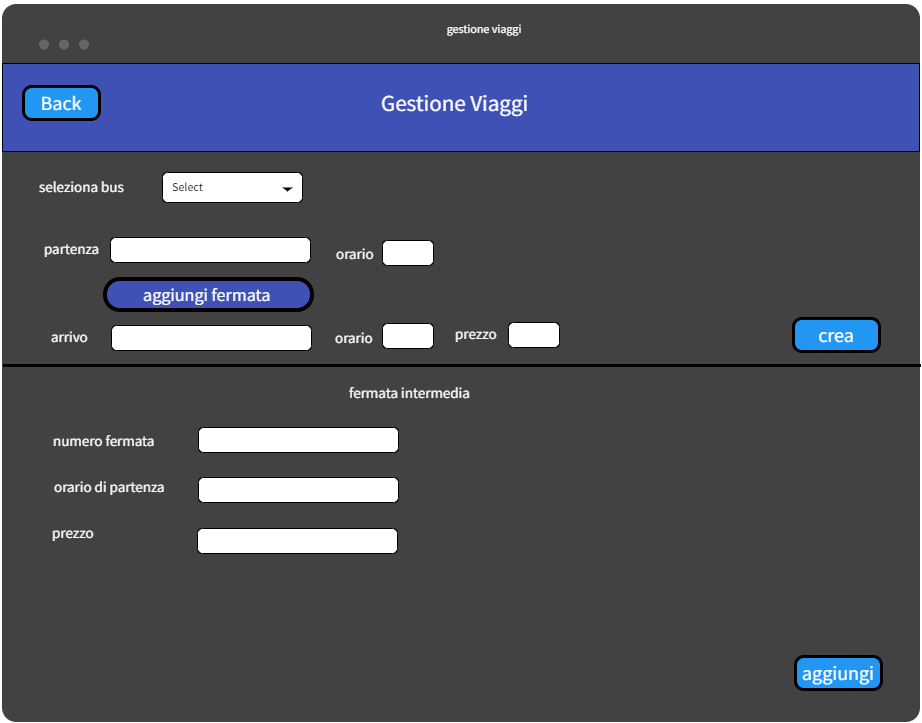
Pagina di home dopo un login di un account amministratore in cui ci sono 4 bottoni per portare alle pagine di creazione di bus e tratte, per cercare le tratte come per un utente normale, e la pagina per gestire gli utenti.

### Pagina crea bus



Pagina per creare un bus scegliendo tra 3 grandezze ossia, grande: 60 posti, medio: 40 posti e piccolo: 20 posti e dando un nome alla linea. Premendo su crea il bus verrà aggiunto al database.

### Pagina crea tratta



Pagina per creare una tratta

## Design procedurale

Descrive i concetti dettagliati dell’architettura/sviluppo utilizzando ad esempio:

* Diagrammi di flusso e Nassi.
* Tabelle.
* Classi e metodi.
* Tabelle di routing
* Diritti di accesso a condivisioni …

Questi documenti permetteranno di rappresentare i dettagli procedurali per la realizzazione del prodotto.

# Implementazione

In questo capitolo dovrà essere mostrato come è stato realizzato il lavoro. Questa parte può differenziarsi dalla progettazione in quanto il risultato ottenuto non per forza può essere come era stato progettato.

Sulla base di queste informazioni il lavoro svolto dovrà essere riproducibile.

In questa parte è richiesto l’inserimento di codice sorgente - Print Screen - di maschere solamente per quei passaggi particolarmente significativi e/o critici.

Inoltre, dovranno essere descritte eventuali varianti di soluzione o scelte di prodotti con motivazione delle scelte.

Non deve apparire nessuna forma di guida d’uso di librerie o di componenti utilizzati. Eventualmente questa va allegata.

Per eventuali dettagli si possono inserire riferimenti ai diari.

# Test

## Protocollo di test

Definire in modo accurato tutti i test che devono essere realizzati per garantire l’adempimento delle richieste formulate nei requisiti. I test fungono da garanzia di qualità del prodotto. Ogni test deve essere ripetibile alle stesse condizioni.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-10 | **Nome:** | Registrazione di un utente non registrato |
| **Descrizione:** | Un utente non registrato ha la possibilità di registrarsi cliccando su “Register”.  Una volta cliccato si viene portati nella pagina di registrazione dove si inserisce nome, cognome, email, password e conferma password.  Se tutti I valori verranno messi correttamente si passerà alla pagina da utente loggato e si potranno acquistare biglietti. | | |
| **Prerequisiti:** | Avere un database dove salvare gli utenti gia registrati con la possibilità di aggiungere nuovi utenti. | | |
| **Procedura:** | 1. Cliccare sul bottone “Register”. 2. Inserire le proprie credenziali 3. Cliccare su “Register” 4. Controllare se c’è la possibilità di rieffettuare il login e di comprare dei biglietti | | |
| **Risultati attesi:** | Possibilità di rifare il login se ci si disconnette e di acquistare I biglietti. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-002  REQ-03 | **Nome:** | Interfaccie diverse tra i livelli di utenti |
| **Descrizione:** | A dipendenza del tipo di utente con la quale si ha fatto il login verranno visualizzate delle interfaccie differenti. | | |
| **Prerequisiti:** | Poter registrare un utente di vario tipo nel database e avere tutte e tre le interfaccie per i vari utenti. | | |
| **Procedura:** | 1. Aprire la pagina da utente non registrato e controllare se viene mostrata la view corretta. 2. Loggarsi con un normale utente e controllare se viene mostrata la view corretta. 3. Effetturare il logout, fare login con un utente amministratore e controllare che viene mostrata la view corretta. | | |
| **Risultati attesi:** | Ad ogni tipo di utente verrà mostrata una view differente. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-003  REQ-04 | **Nome:** | Inserire nel database un tragitto |
| **Descrizione:** | Un utente di tipo amministratore può creare nuovi tragitti. | | |
| **Prerequisiti:** | Avere il database funzionante con la possibilità di aggiungere tragitti e capire se l’utente loggato è un amministratore. | | |
| **Procedura:** | 1. Fare il login con un utente amministatore 2. Cliccare sul bottone aggiungi tratte 3. Mettere tutti i parametri richiesti per creare una tratta 4. Cliccare su aggiungi tratta 5. Controllare che la tratta è stata aggiunta correttamente | | |
| **Risultati attesi:** | Se cerchiamo la tratta appena aggiunta sarà presente. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-004  REQ-05 | **Nome:** | Controllare e gestire i partecipanti dei viaggi |
| **Descrizione:** | Un utente di tipo amministratore può visualizzare tutti gli utenti che hanno acquistato un biglietto nella tratta selezionata e può vedere e modificare I dati da loro inseriti.. | | |
| **Prerequisiti:** | Avere il database funzionante, capire se l’utente loggato è un amministratore e se lo è ha la possibilità di aggiungere tragitti.  Poter creare utenti che possono acquistare I biglietti | | |
| **Procedura:** | 1. Fare il login con un utente amministatore 2. Ricercare una tratta 3. Cliccare sulla tratta desiderata 4. Verrà mostrata una lista dei partecipanti, selezionarne uno 5. Modificare I suoi dati. | | |
| **Risultati attesi:** | Controllare che una volt ache I ati sono stati modificati si salvano corretamente sul database e si salvano. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-005  REQ-06 | **Nome:** | Stampare lista dei partecipanti di un viaggio |
| **Descrizione:** | Un utente di tipo amministratore può visualizzare tutti gli utenti che hanno acquistato un biglietto nella tratta selezionata e può stamparne la lista. | | |
| **Prerequisiti:** | Avere il database funzionante, capire se l’utente loggato è un amministratore e se lo è ha la possibilità di aggiungere tragitti.  Poter creare utenti che possono acquistare i biglietti | | |
| **Procedura:** | 1. Fare il login con un utente amministatore 2. Ricercare una tratta 3. Cliccare sulla tratta desiderata 4. Verrà mostrata una lista dei partecipanti con un tasto stampa 5. Selezionare la stampante e stampare | | |
| **Risultati attesi:** | Una volta mandata la stampa controllare che venga fisicamente stampata la lista di partecianti. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-006  REQ-07 | **Nome:** | Sistema filtrabile di mass mail |
| **Descrizione:** | Un utente di tipo amministratore può visualizzare tutti gli utenti dalla gestione utenti filtrarli e mandare una mail a tutti quelli selezionati.. | | |
| **Prerequisiti:** | Avere il database funzionante, capire se l’utente loggato è un amministratore.  Poter creare utenti normali che si salvano sul database. | | |
| **Procedura:** | 1. Fare il login con un utente amministatore 2. Cliccare su gestione utenti 3. Filtrare a piacimento gli utenti. 4. Cliccare su “Invia mail” | | |
| **Risultati attesi:** | Una volta mandata la mail controllare che sia arrivata a tutti gli utenti selezionati | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-007  REQ-08 | **Nome:** | Gestione dei pagamenti |
| **Descrizione:** | Un utente di tipo amministratore può visualizzare tutti gli utenti dalla gestione utenti filtrarli e selezionare un utente per vedere nel dettaglio i dati e i pagamenti | | |
| **Prerequisiti:** | Avere il database funzionante, capire se l’utente loggato è un amministratore.  Poter creare utenti normali che si salvano sul database e possono aquistare viaggi. | | |
| **Procedura:** | 1. Fare il login con un utente amministatore 2. Cliccare su gestione utenti 3. Filtrare a piacimento gli utenti. 4. Selezionare l’utente desiderato 5. Possibilità di vedere e modificare i pagamenti | | |
| **Risultati attesi:** | Una volta modificati controllare che vengono salvati correttamente. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-008  REQ-09 | **Nome:** | Ricerca tragitto con qualsiasi tipo di utente |
| **Descrizione:** | Qualisiasi tipo di utente può ricercare e filtrare un viaggio. | | |
| **Prerequisiti:** | Avere il database funzionante, aver la possibilità di creare dei viaggi da un utente amministatore.  Avere tutti e tre i tipi di account | | |
| **Procedura:** | 1. Ricercare un viaggio 2. Filtrare i viaggi 3. Cliccare su “Cerca” 4. Ripetere con tutti i tipi di utente | | |
| **Risultati attesi:** | Con ogni tipo di utente una volta schiacciato cerca devono apparire I viaggi correti. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-00?  REQ-011 | **Nome:** | Log-in al sito |
| **Descrizione:** | Un utente dovrà potersi loggare al sito. | | |
| **Prerequisiti:** | Avere il database funzionante, con un utente creato. | | |
| **Procedura:** | 1. Andare sulla pagina di login del sito 2. Immettere l’email e la password 3. Premere il bottone login | | |
| **Risultati attesi:** | Una volta premuto login dovrà portarti alla pagina di home del sito e si dovrà vedere che si è loggati sull’icona del profilo | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-00?  REQ-012 | **Nome:** | Prenotazione biglietto |
| **Descrizione:** | Un utente dovrà potersi loggare al sito. | | |
| **Prerequisiti:** | Avere il database funzionante, con un utente creato. | | |
| **Procedura:** | 1. Fare login sul sito 2. Premere su una tratta 3. Premere sul bottone acquista 4. Selezionare quanti biglietti si vogliono riservare 5. Premere sopra lo schema quali posti liberi si vogliono riservare 6. Premere effettua pagamento | | |
| **Risultati attesi:** | Si dovrà ricevere una mail di conferma dell’acquisto del biglietto con le informazioni del biglietto. | | |

## Risultati test

Tabella riassuntiva in cui si inseriscono i test riusciti e non del prodotto finale. Se un test non riesce e viene corretto l’errore, questo dovrà risultare nel documento finale come riuscito (la procedura della correzione apparirà nel diario), altrimenti dovrà essere descritto l’errore con eventuali ipotesi di correzione.

## Mancanze/limitazioni conosciute

Descrizione con motivazione di eventuali elementi mancanti o non completamente implementati, al di fuori dei test case. Non devono essere riportati gli errori e i problemi riscontrati e poi risolti durante il progetto.

# Consuntivo

Consuntivo del tempo di lavoro effettivo e considerazioni riguardo le differenze rispetto alla pianificazione (cap. 1.7) (ad esempio Gantt consuntivo).

# Conclusioni

Quali sono le implicazioni della mia soluzione? Che impatto avrà? Cambierà il mondo? È un successo importante? È solo un’aggiunta marginale o è semplicemente servita per scoprire che questo percorso è stato una perdita di tempo? I risultati ottenuti sono generali, facilmente generalizzabili o sono specifici di un caso particolare? ecc.

## Sviluppi futuri

Migliorie o estensioni che possono essere sviluppate sul prodotto.

## Considerazioni personali

Cosa ho imparato in questo progetto? ecc.

# Bibliografia

## Bibliografia per articoli di riviste:

1. Cognome e nome (o iniziali) dell’autore o degli autori, o nome dell’organizzazione,
2. Titolo dell’articolo (tra virgolette),
3. Titolo della rivista (in italico),
4. Anno e numero
5. Pagina iniziale dell’articolo.

## Bibliografia per libri

1. Cognome e nome (o iniziali) dell’autore o degli autori, o nome dell’organizzazione,
2. Titolo del libro (in italico),
3. ev. Numero di edizione,
4. Nome dell’editore,
5. Anno di pubblicazione,
6. ISBN.

## Sitografia

1. URL del sito (se troppo lungo solo dominio, evt completo nel diario),
2. Eventuale titolo della pagina (in italico),
3. Data di consultazione (GG-MM-AAAA).

**Esempio:**

* http://standards.ieee.org/guides/style/section7.html, *IEEE Standards Style Manual*, 07-06-2008.

# Allegati

Elenco degli allegati, esempio:

* Diari di lavoro
* Codici sorgente/documentazione macchine virtuali
* Istruzioni di installazione del prodotto (con credenziali di accesso) e/o di eventuali prodotti terzi
* Documentazione di prodotti di terzi
* Eventuali guide utente / Manuali di utilizzo
* Mandato e/o QdC
* Prodotto
* …