Gestione Appartamenti

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 8

2.4.1 Analisi 9

2.4.2 Progettazione 10

2.4.3 Implementazione 11

2.5 Analisi dei mezzi 12

2.5.1 Software 12

2.5.2 Hardware 12

3 Progettazione 12

3.1 Design dell’architettura del sistema 12

3.2 Design dei dati e database 12

3.3 Design delle interfacce 12

3.4 Design procedurale 12

4 Implementazione 14

5 Test 14

5.1 Protocollo di test 14

5.2 Risultati test 14

5.3 Mancanze/limitazioni conosciute 15

6 Consuntivo 15

7 Conclusioni 15

7.1 Sviluppi futuri 15

7.2 Considerazioni personali 15

8 Bibliografia 15

8.1 Bibliografia per articoli di riviste: 15

8.2 Bibliografia per libri 15

8.3 Sitografia 15

9 Allegati 15

# Introduzione

## Informazioni sul progetto

**Titolo:** Gestione Appartamenti

**Allievi**: Mattia Toscanelli, impiegato nello svolgimento del progetto.

**Classe**: I4AC

**Docenti**: Massimo Sartori

**Sezione scuola**: Scuola Arti e Mestieri Trevano, Informatica

**Data inizio**: 23.01.2020

**Fine**: 06.04.2020

## Abstract

E’ una breve e accurata rappresentazione dei contenuti di un documento, senza notazioni critiche o valutazioni. Lo scopo di un abstract efficace dovrebbe essere quello di far conoscere all’utente il contenuto di base di un documento e metterlo nella condizione di decidere se risponde ai suoi interessi e se è opportuno il ricorso al documento originale.

Può contenere alcuni o tutti gli elementi seguenti:

* **Background/Situazione iniziale**
* **Descrizione del problema e motivazione**: Che problema ho cercato di risolvere? Questa sezione dovrebbe includere l'importanza del vostro lavoro, la difficoltà dell'area e l'effetto che potrebbe avere se portato a termine con successo.
* **Approccio/Metodi**: Come ho ottenuto dei progressi? Come ho risolto il problema (tecniche…)? Quale è stata l’entità del mio lavoro? Che fattori importanti controllo, ignoro o misuro?
* **Risultati**: Quale è la risposta? Quali sono i risultati? Quanto è più veloce, più sicuro, più economico o in qualche altro aspetto migliore di altri prodotti/soluzioni?

Esempio di abstract:

*As the size and complexity of today’s most modern computer chips increase, new techniques must be developed to effectively design and create Very Large Scale Integration chips quickly. For this project, a new type of hardware compiler is created. This hardware compiler will read a C++ program, and physically design a suitable microprocessor intended for running that specific program. With this new and powerful compiler, it is possible to design anything from a small adder, to a microprocessor with millions of transistors. Designing new computer chips, such as the Pentium 4, can require dozens of engineers and months of time. With the help of this compiler, a single person could design such a large-scale microprocessor in just weeks.*

## Scopo

Lo scopo di questo progetto è quello di realizzare un sito web che permetta di gestire l’affitto di appartamenti in determinati luoghi di villeggiatura. Gli utenti registrati a questo sito avranno la possibilità di affittare giornalmente o definitivamente un appartamento oppure ne potranno registrare di nuovi. Gli appartamenti potranno essere riservati sia da parte degli utenti registrati al sito oppure dal proprietario. Ogni riservazione dovrà essere confermata da un responsabile. Per ogni appartamento messo a disposizione dai proprietari dell’immobile si dovranno avere dei resoconti settimanali, mensili e annuali. In seguito sarà presente una pagina principale dove l’amministratore potrà inserire foto, inserire notizie e modificare il testo. Infine il sito web dovrà essere visualizzabile da tutti i principali browser e cellulari.

# Analisi

## Analisi del dominio

Bisogna trovare un metodo che permetta di gestire l’affitto di appartamenti in determinati luoghi di villeggiatura. L’obiettivo di questo progetto è dunque quello di creare un applicativo web che concede alle persone registrate al sito di poter cercare appartamenti da affittare oppure aggiungerne di nuovi. Esistono sul mercato già prodotti simili che svolgono queste funzioni ma che si differenziano per dei piccoli particolari. Infatti, grazie a questo progetto, i proprietari di appartamenti potranno visualizzare un riepilogo settimanale, mensile o annuale dei propri appartamenti.

Quando un utente di registra al sito dovrà verificare la propria e-mail per poter incominciare ad utilizzare l’applicativo. A questo punto l’utente avrà la possibilità di incominciare ad affittare appartamenti. Se invece l’utente vuole inserire dei propri annunci, bisognerà aspettare finché l’amministratore del sito abbia verificato quest’ultimo.

I compiti dell’amministratore non finiscono qui, infatti esso dovrà poter gestire una pagina principale dove vengono mostrare delle foto e delle notizie riguardanti il sito.

L’interfaccia dell’applicazione deve essere molto semplice e intuitiva, in poche parole deve essere comprensibile anche al meno esperto di informatica. Inoltre l’applicazione dovrà essere visualizzabile da tutti i principali browser e dispositivi mobili.

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Sito web per la gestione degli appartamenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Ci deve essere una pagina principale accessibile a tutti gli utenti, anche a coloro che non si sono registrati. |
| **002** | Ogni funzione del sito non deve presentare imprevisti. |
| **003** | Il sito deve essere responsive, cioè deve essere adattato a qualsiasi tipo di schermo. |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Pagina di registrazione |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Tutti possono accedere alla pagina di registrazione. |
| **002** | La e-mail dell’utente che si vuole registrare al sito deve essere verificata e non dove appartenere a un dominio di e-mail temporanee. |
| **003** | I dati obbligatori da inserire nella registrazione sono 6:  Email, password, nome, cognome, indirizzo e numero di telefono mobile. |
| **004** | I dati opzionali nella registrazione sono 2:  Numero di telefono di ufficio e numero di telefono privato. |
| **ID: REQ-003** | |
| **Nome** | Pagina di login |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Tutti gli utenti che vogliono accedere devono dapprima effettuare la verifica della e-mail. |
| **002** | Ci deve essere la possibilità di cambiare la password in caso di smarrimento. |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Pagina principale |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Si devono mostrare le notizie del sito. |
| **002** | Vengono mostrate del foto. |
| **003** | Solo gli amministratori della pagina possono aggiungere notizie, aggiungere foto e modificare il testo della pagina principale. |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | Pagina di ricerca di appartamenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Ci deve essere una pagina principale dove vengono mostrati tutti gli appartamenti. |
| **002** | La pagina deve avere una barra di ricerca per permettere di filtrare tutti gli appartamenti. |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Pagina singola di un determinato appartamento. |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Ogni appartamento deve presentare le seguenti informazioni:  Titolo, città, via, CAP, numero di locali, se offre posteggi o no, prezzo (mensile e/o giornaliero), spese, , ammobiliato, se si può fumare o no, se si possono tenere animali, se si possono tenere bambini nell’appartamento o no e infine se è disponibile o meno. |
| **002** | Ogni appartamento può presentare le seguenti informazioni:  Foto, pianta dell’appartamento, accessori, l’ubicazione e la carta e infine i commenti. |
| **003** | I proprietari dell’appartamento possono visualizzare un resoconto settimanale, mensile o annuale. |
| **004** | Ogni appartamento può essere riservato sia dal proprietario sia da un utente registrato. |
| **005** | Per ogni riservazione va scelto un periodo e quest’ultima deve essere accettata dal proprietario. |
| **006** | I proprietari devono avere la possibilità di aggiungere una riservazione per un cliente che magari lo chiama per telefono. |

|  |  |
| --- | --- |
| **ID: REQ-007** | |
| **Nome** | Pagina di gestione dei propri appartamenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Ci deve essere una pagina principale dove vengono mostrati tutti gli appartamenti. |
| **002** | La pagina deve avere una barra di ricerca per permettere di filtrare i propri appartamenti. |
| **003** | Si deve poter inserire nuovi appartamenti |

|  |  |
| --- | --- |
| **ID: REQ-008** | |
| **Nome** | Pagina di amministrazione |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | - |
| **Sotto requisiti** | |
| **001** | Questa pagina è accessibile solamente dall’amministratore. |
| **002** | C’è una tabella dove vengono mostrati tutti gli utenti registrati. |
| **003** | Ci deve essere la possibilità di modificare o eliminare un utente. |
| **004** | Ci deve essere una sezione per la modifica della pagina principale. |

## Use case

Questo è lo Use case (casi di utilizzo) per l’applicazione Gestione Appartamenti:

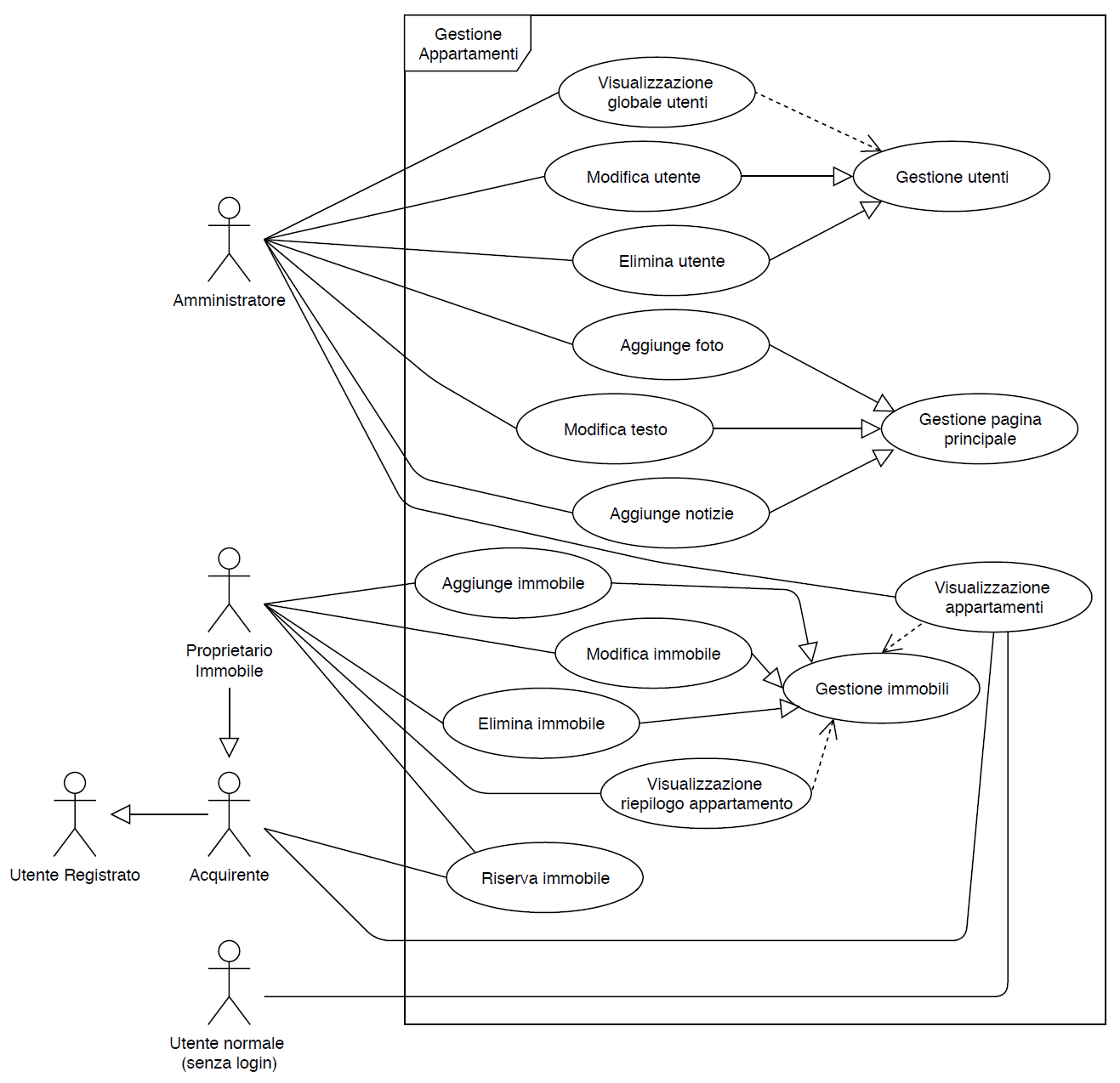


Figura 1 Use Case

Lo schema inizia con quattro tipologie di scenari: Amministratore (cioè colui che gestisce tutta l’applicazione), Proprietario immobile, Acquirente e Utente normale.

L’amministratore ha la possibilità di visualizzare una situazione globale di tutti gli utenti registrati al sito, mostrando tutti i dati non sensibili. Grazie a questa visualizzazione esso può modificare o eliminare un utente.

Inoltre l’amministratore può aggiungere foto, aggiungere notizie e modificare il testo nella pagina principale dell’applicazione.

Quando un utente si registra al sito, bisognerà verificare la propria e-mail. Una volta verificata questo utente diventa automaticamente un acquirente dell’applicazione. Esso può visualizzare tutti gli appartamenti messi a disposizione da altri e effettuare delle riservazioni.

L’utente proprietario immobile assume tutte le funzioni di un acquirente ma ha anche la possibilità di aggiungere, modificare o eliminare un immobile. Inoltre potrà visualizzare un riepilogo settimanale, mensile o annuale dei propri appartamenti. Per essere un proprietario immobile viene richiesta una verifica da parte dell’amministratore.

Infine esiste l’utente normale che non ha bisogno di una registrazione. Esso ha solamente la possibilità di visualizzare tutti gli appartamenti del sito non riservarlo.

## Pianificazione

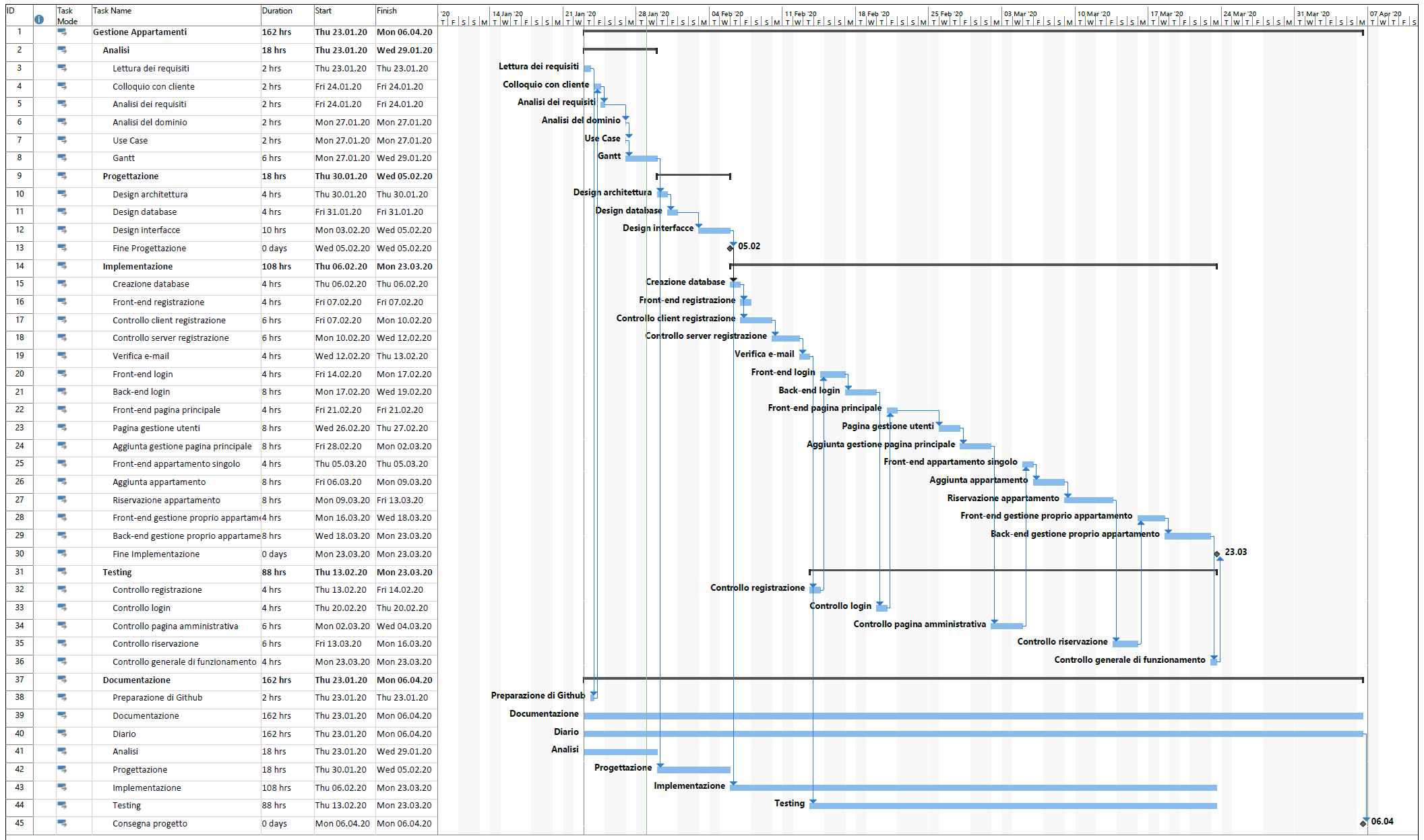


Figura 2 Gantt Preventivo Intero

Il Gantt si suddivide in 4 fasi di lavoro (Analisi, Progettazione, Implementazione e Testing) e 1 fase di ricapitolazione.

### Analisi

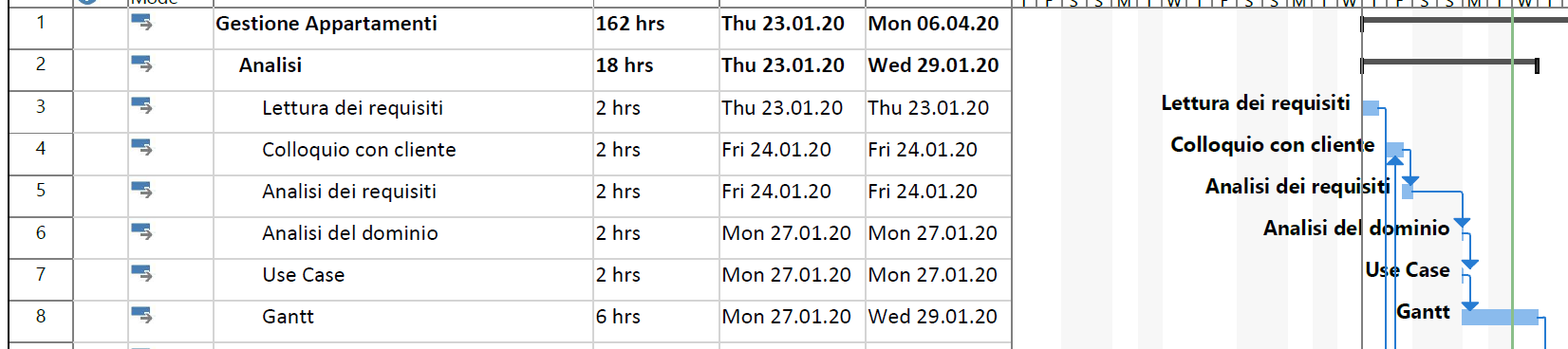


Figura 3 Gantt Preventivo – Analisi

L’analisi si suddivide in 6 lavori:

* Lettura dei requisiti 🡪 la lettura del diario dei compiti consegnato dal responsabile.
* Colloquio con il cliente 🡪 la risoluzione dei dubbi con il responsabile per quanto riguarda il diario dei compiti.
* Analisi dei requisiti 🡪 i requisiti che deve avere necessariamente applicazione.
* Analisi del dominio 🡪 riflessione se ci sono dei prodotti simili sul mercato e se c’è seriamente bisogno di intraprendere questo progetto.
* Use Case 🡪 la progettazione dello schema Use Case.
* Stesura Gantt pianificazione 🡪 la progettazione e la descrizione dello schema Gantt.

Per le attività di analisi ho previsto che sono necessari circa 4 giorni di lavoro.

### Progettazione

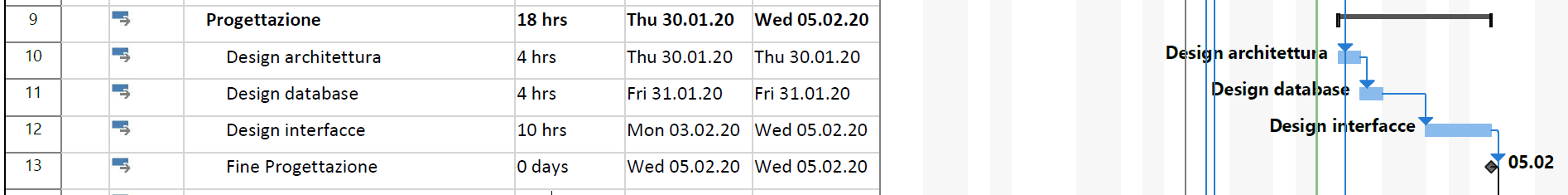


Figura 4 Gantt Preventivo – Progettazione

La progettazione si suddivide in 3 lavori e un punto cardine:

* Design architettura 🡪 come funziona l’applicazione, come sono collegate le varie operazioni l’una fra l’altra.
* Design database 🡪 come sono composto il database.
* Design interfacce 🡪 come si presentano all’utente finale le varie interfacce grafiche.
* Fine progettazione 🡪 punto conclusivo della progettazione per dare inizio all’implementazione.

Ho previsto che per svolgere queste 4 attività mi sono necessari circa 4 giorni di lavoro.

### Implementazione

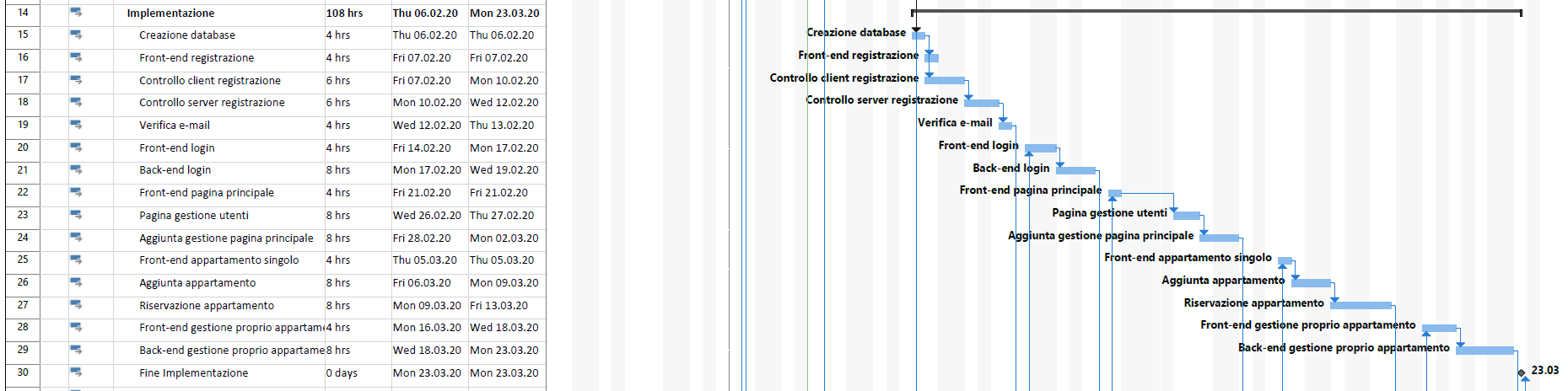


Figura 5 Gantt Preventivo – Implementazione

L’implementazione è la fase del progetto che richiede più tempo per essere svolta e si suddivide in 10 attività e un punto cardine:

* Creazione Database 🡪 scrittura e utilizzo dei vari database.
* Front-end registrazione 🡪 creazione interfaccia grafica per la registrazione di un utente.
* Controllo client registrazione 🡪 controllo lato client dei dati inseriti nella pagina di registrazione.
* Controllo server registrazione 🡪 controllo lato server dei dati inseriti nella pagina di registrazione.
* Veridica e-mail 🡪 controllo se la e-mail inserita nella pagina di registrazione è veritiera.
* Front-end pagina login 🡪 interfaccia grafica per il login che consente l’accesso alle funzioni aggiuntive.
* Back-end login 🡪 controllo e accesso dei dati inseriti in modo protetto.
* Front-end pagina principale 🡪 interfaccia per la visualizzazione della pagina principale.
* Pagina gestione utenti 🡪 aggiunta della pagina riservata all’amministratore per gestire gli utenti della pagina.
* Aggiunta gestione pagina principale 🡪 aggiunta alla pagina riservata all’amministratore la possibilità di modificare la pagina principale
* Front-end appartamento singolo 🡪 interfaccia per la visualizzazione della pagina che descrive un determinato appartamento.
* Aggiunta appartamento 🡪 operazione per aggiungere un appartamento da parte di un proprietario immobile.
* Riservazione appartamento 🡪 operazione di riservazione di un appartamento da un utente registrato.
* Front-end gestione proprio appartamento 🡪 interfaccia per la visualizzazione del riepilogo del proprio appartamento.
* Back-end gestione proprio appartamento 🡪 inserimento dei dati di riepilogo nell’interfaccia.
* Fine implementazione 🡪 punto conclusivo della implementazione.

Per completare la fase di implementazione ho calcolato che ci vogliono circa 24 giorni lavorativi.

### Testing

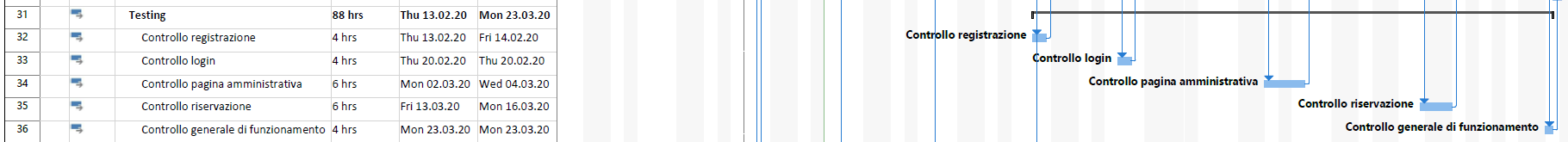
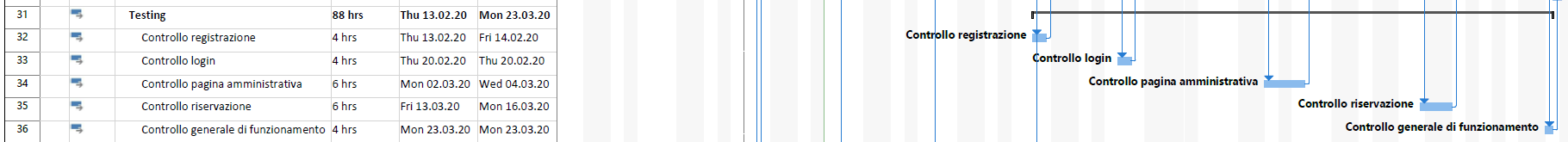


Figura 6 Gantt Preventivo - Testing

La fase di testing si suddivide in 5 lavori:

* Controllo registrazione 🡪 controllo per verificare che i campi della registrazione vangano controllati a dovere, compreso l’invio della e-mail di verifica.
* Controllo login 🡪 controllo per verificare che il login funzioni.
* Controllo pagina amministrativa 🡪 controllo di funzionamento della pagina amministrativa.
* Controllo riservazione 🡪 controllo di funzionamento della riservazione di un appartamento.
* Controllo generale di funzionamento 🡪 controllo globale di tutte le funzionalità dell’applicazione.

Ho previsto che per svolgere queste 5 attività mi sono necessari circa 4 giorni di lavoro suddivisi in tutto l’arco del progetto.

### Documentazione

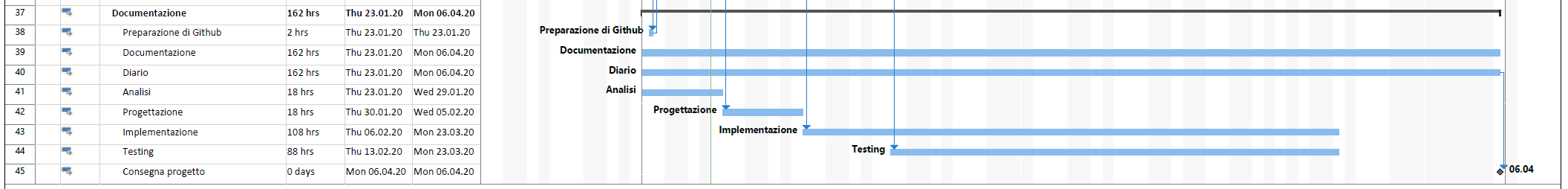
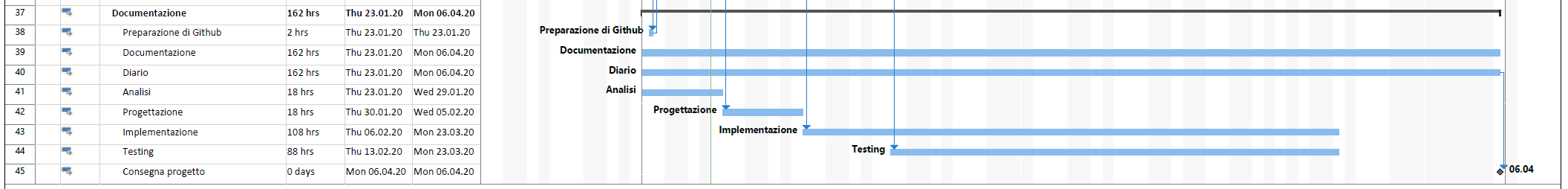


Figura 7 Gantt Preventivo - Documentazione

La fase riassuntiva si suddivide in 7 lavori e un punto cardine:

* Preparazione di GitHub 🡪 preparazione dell’ambiente su GitHub per mostrare l’andamento del progetto.
* Documentazione 🡪 scrittura della documentazione del progetto.
* Diario 🡪 scrittura dei lavori effettuati giornalmente.
* Analisi 🡪 analisi sulle funzioni del progetto.
* Progettazione 🡪 progettazione delle funzioni del progetto.
* Implementazione 🡪 implementazione dei lavori che deve svolgere il progetto.
* Testing 🡪 verifica delle funzioni implementate.
* Consegnare progetto 🡪 momento di consegna del progetto.

Questa fase di documentazione dura per tutto lo svolgimento del progetto.

## Analisi dei mezzi

Per la realizzazione di questo progetto si ha bisogno di:

* Un PC con performance in grado di girare Windows 10 e un WebServer Apache.

### Software

I programmi utilizzati per svolgere questo progetto sono:

* Word 2016 🡪 per la documentazioni
* PowerPoint 2016 🡪 per la presentazione
* Draw.io 🡪 per i disegni dei vari schemi e progettazioni
* PhpStorm 2019.2.1 🡪 per lo sviluppo web
* Apache 2.4 🡪 per il Web Server
* PHP 7.1.9 🡪 per l’utilizzo del linguaggio PHP

### Hardware

Il pc su cui è stato sviluppato il progetto ha le seguenti caratteristiche:

* OS: Windows 10 Pro
* CPU: i7-8550U
* RAM: 8,00GB
* Scheda video: Nvidia MX150

### Librerie

Inserire le librerie utilizzate.

# Progettazione

## Design dell’architettura del sistema

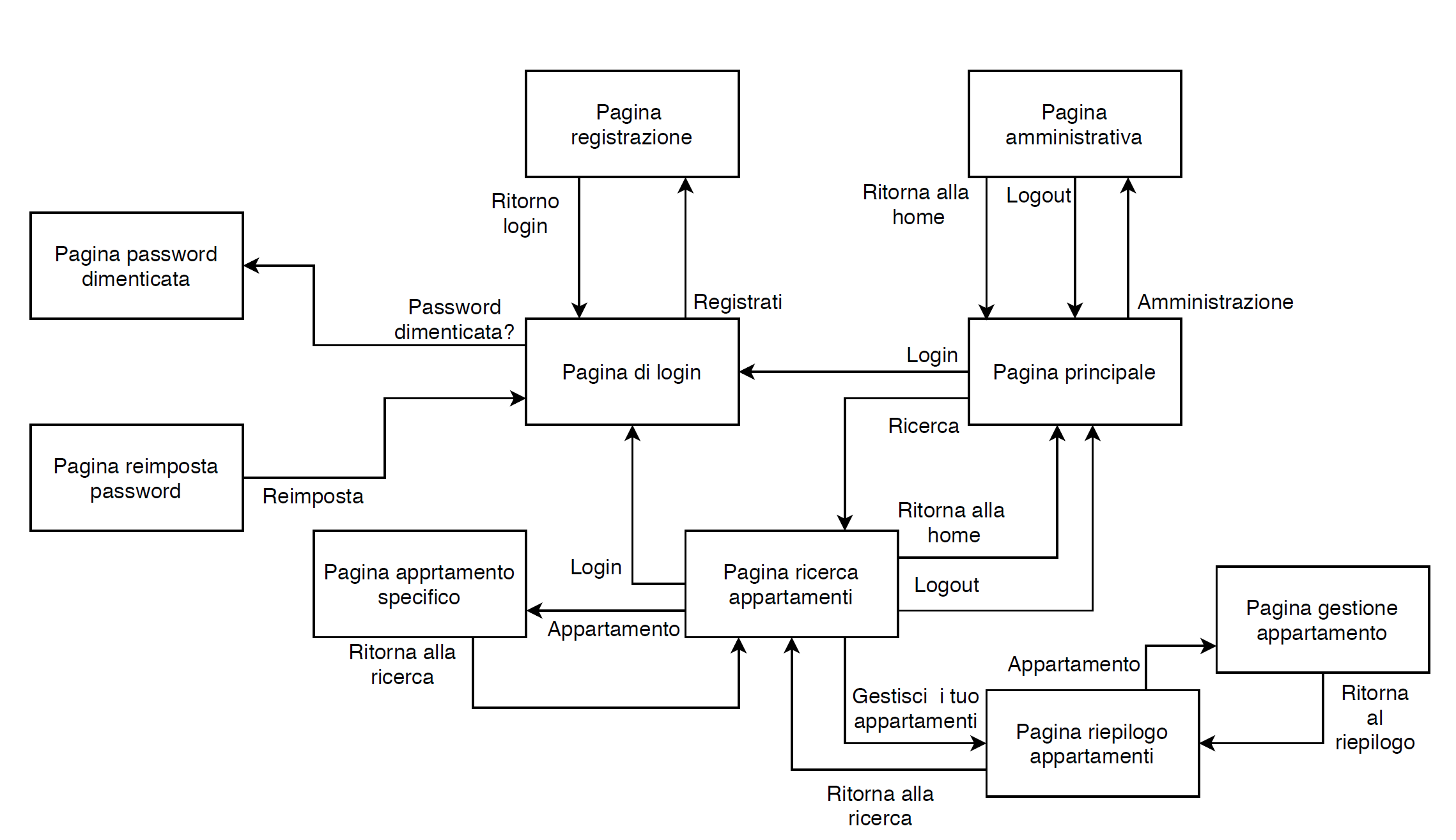


Figura 8 Design architettura del sistema

La pagina di partenza della mia applicazione, cioè la pagina iniziale a cui si collega l’utente finale per utilizzare il mio sito, è la pagina principale. In questa pagina l’utente può farsi un’idea di quello che offre tutto il sito, infatti può visualizzare delle immagini di appartamenti e leggere delle notizie riguardanti novità dell’applicazione. In questa pagina, come nella maggior parte delle altre, sarà possibile effettuare un accesso cliccando il bottone “Login” in una navbar in alto. Nella pagina di login si può naturalmente dare il login inserendo le credenziali di accesso (e-mail e password), ci si può registrare cliccando sul pulsante “Registrati” oppure infine si può modificare la propria password cliccando il bottone “Hai dimenticato la password?”. Quando viene richiesta la procedura di modifica password tramite la pagina di password dimenticata viene inviata una e-mail, all’utente diretto interessato, la quale contiene un link di recupero. Cliccando questo link l’utente verrà trasportato sulla pagina reimposta password dove esso potrà inserire una nuova password.

Ritornando alla pagina principale si può accedere alla pagina dove si possono ricercare appartamenti. In questa pagina sarà presente una barra di ricerca e alcuni filtri per semplificare la ricerca. Quando l’utente ha trovato l’appartamento, lo clicca, e viene “portato” nella pagina specifica di un appartamento. In questa pagina se l’utente ha effettuato l’accesso può effettuare la riservazione altrimenti no.

Ritornando alla pagina di ricerca appartamenti, se l’utente ha effettuato l’accesso ed è stato verificato dall’amministratore per essere un proprietario immobile, può accedere ad una pagina riservata dove gli vengono mostrati tutti i gli appartamenti aggiunti da lui. Cliccando su un proprio appartamento, l’utente accede ad un pagina di gestione dove può visualizzare un riepilogo generale dell’appartamento (settimanale, mensile e annuale) e un calendario che gli mostra le future prenotazioni. Inoltre, in questa pagina, il proprietario può modificare le informazioni dell’appartamento oppure aggiungere una riservazione da parte di un cliente che magari lo ha chiamato per telefono.

Infine in quasi ogni pagina, dopo aver effettuato il login, si può effettuare il logout tramite le navbar.

## Design dei dati e database

### Schema E-R

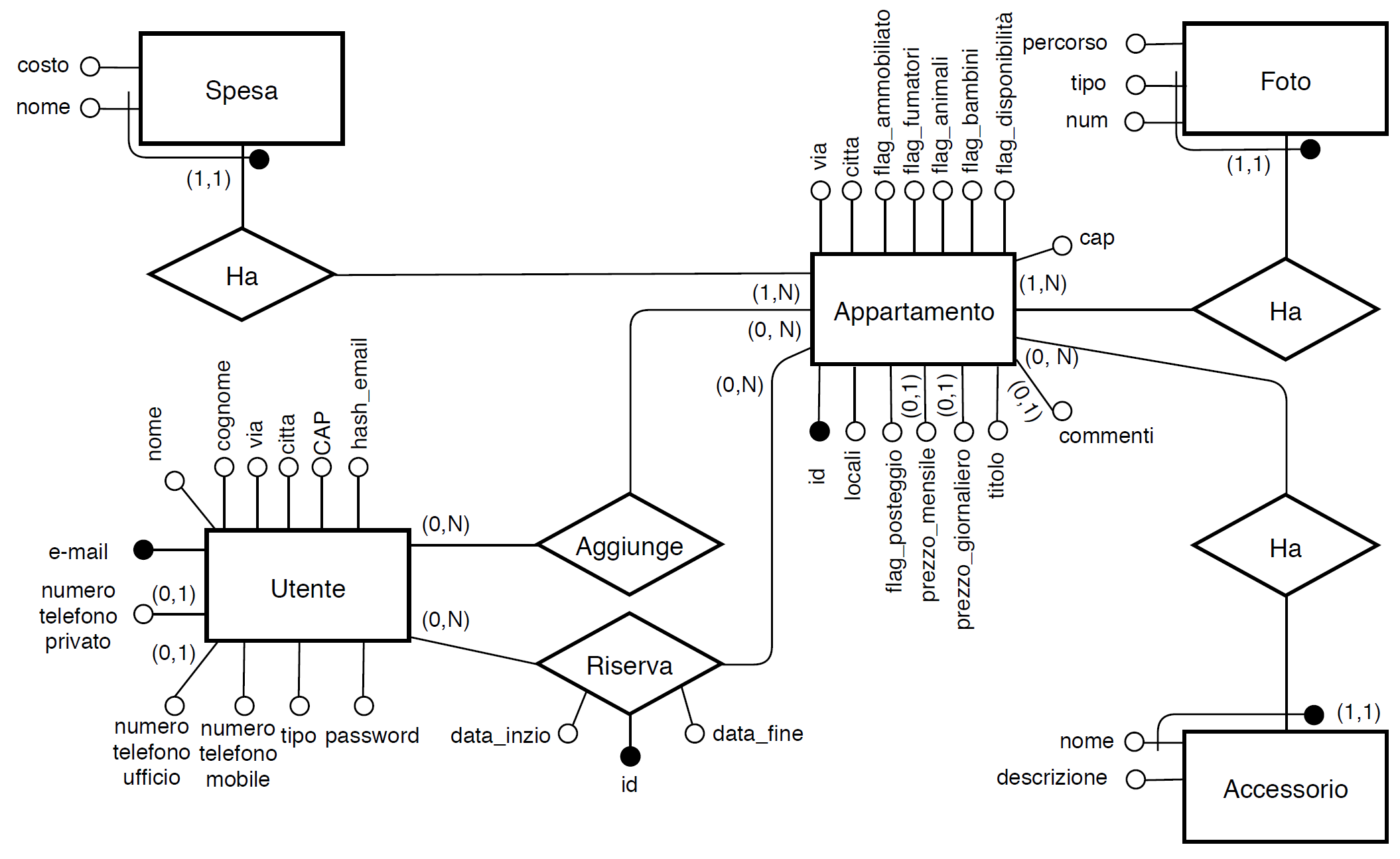


Figura 9 Schema E-R

### Schema logico:

Utente(e-mail, nome, cognome, via, citta, cap, numero\_telefono\_privato\*, numero\_telefono\_ufficio\*, numero\_telefono\_mobile, tipo, password, hash\_email)

Appartamento(id, locali, prezzo\_mensile\*, prezzo\_giornaliero\*, titolo, commenti, via, citta, cap, flag\_posteggio, flag\_ammobiliato, flag\_fumatori, flag\_animali, flag\_bambini, flag\_disponibilita)

Accessorio(nome, id\_appartamento (FK), descrizione)

Spesa(nome, id\_appartamento (FK), costo)

Foto(num, id\_appartamento (FK), tipo, percorso)

Riserva(id, e-mail (FK), id\_appartamento (FK), data\_inzio, data\_fine)

Aggiunge(e-mail(FK), id\_appartamento(FK))

### Vincoli

Nella tabella Utente l’attributo tipo specifica il livello di gerarchia dell’utente. I livello sono tre e sono i seguenti:

* 2: utente registrato al sito ma senza essere stato verificato dall’admin, cioè un utente che può solo effettuare riservazioni di appartamenti.
* 3: utente registrato ma con verifica da parte dell’admin, cioè l’utente con possibilità di inserire i propri appartamenti nel sito.
* 5: admin del sito. Cioè colui che può effettuare tutte le azioni di un utente verificato e con l’accesso alla pagina di amministrazione.

Il vicolo da rispettare per la tabella Utente è dunque:

* Il campo tipo può contenere i valori 2, 3 e 5.
  + Se tipo = 2, non può partecipare alla relazione Aggiunge ma può partecipare alla Riserva.
  + Se tipo = 3 oppure tipo = 5, può partecipare sia alla relazione Aggiunge sia alla relazione Riserva.

Ci sono anche due piccoli vincoli per quanto riguarda la tabella Riserva ed è il seguente:

* Non ci possono essere due riservazioni dello stesso appartamento con intervalli di tempo che collidono.
* L’intervallo di tempo fra data\_inzio e data\_fine deve essere maggiore o uguale ad un giorno.

Per quanto riguarda i dati ci sono delle piccole regole da gestire:

* Tutti i campi di testo devono avere almeno 3 lettere.
* Numeri di telefono devono avere almeno 10 numeri.
* Password deve avere almeno 8 caratteri tra cui deve essere presente un numero o carattere speciale. (Naturalmente nel database verranno inserite le password cifrate quindi sarà da gestire lato server).

### Foreign key e utenti

L’utente utilizzato per le query sul database ha le seguente credenziali:

* Nome: gestione\_appartamenti\_admin
* La password si deve richiedere al proprietario del progetto per questioni di sicurezza.

Questo utente ha i permessi di scrittura e lettura su tutte le tabelle del database gestione\_appartamenti.

Nella tabella Aggiunge ci sono due foreign key. Se si elimina un utente oppure un appartamento vengono eliminati tutti i rifermenti nella tabella Aggiunge.

Nella tabella Riserva ci sono due foreign key. Se si elimina un utente oppure un appartamento vengono eliminati tutti i rifermenti nella tabella Riserva.

Nelle tabelle Spesa, Foto e Accessorio c’è una foreign key di Appartamento. Se si elimina un appartamento vengono eliminati tutti i rifermenti nella rispettive tabelle.

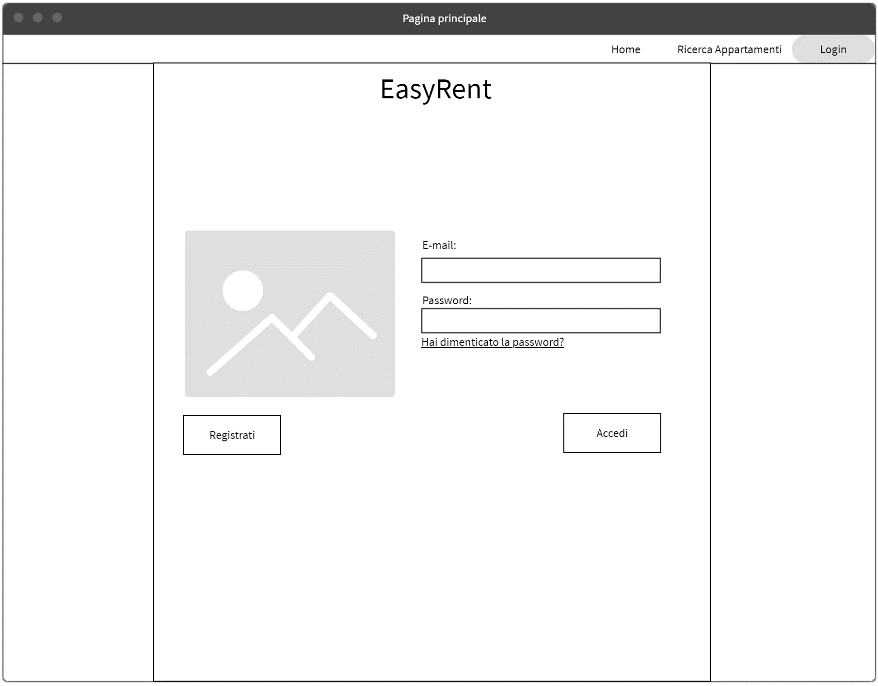
## Design delle interfacce

In questo capitoletto mostrerò delle anteprime delle varie interfacce delle pagine con una breve descrizione.

### Pagina principale

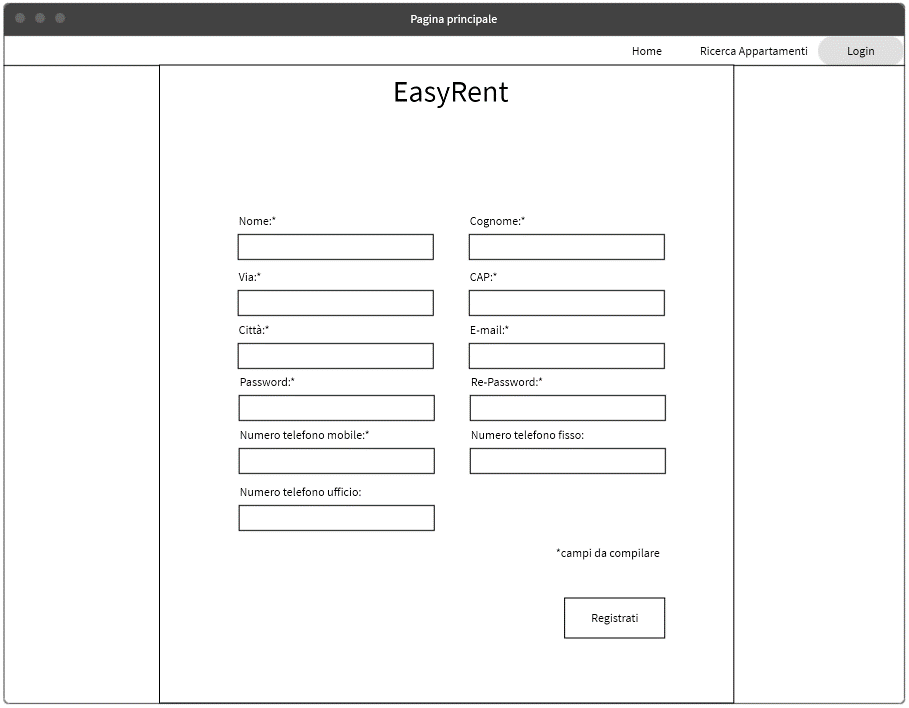
Questa è la pagina iniziale a cui l’utente finale si collega. In questa pagina si ha la possibilità di visualizzare delle immagini di appartamenti e delle varie notizie riguardanti il sito. Al di sopra della pagina si può trova una navbar. Attraverso questa navbar si potrà accedere alla pagina di ricerca appartamenti, alla pagina di login oppure alla home (cioè la pagina principale).

### Pagine di login



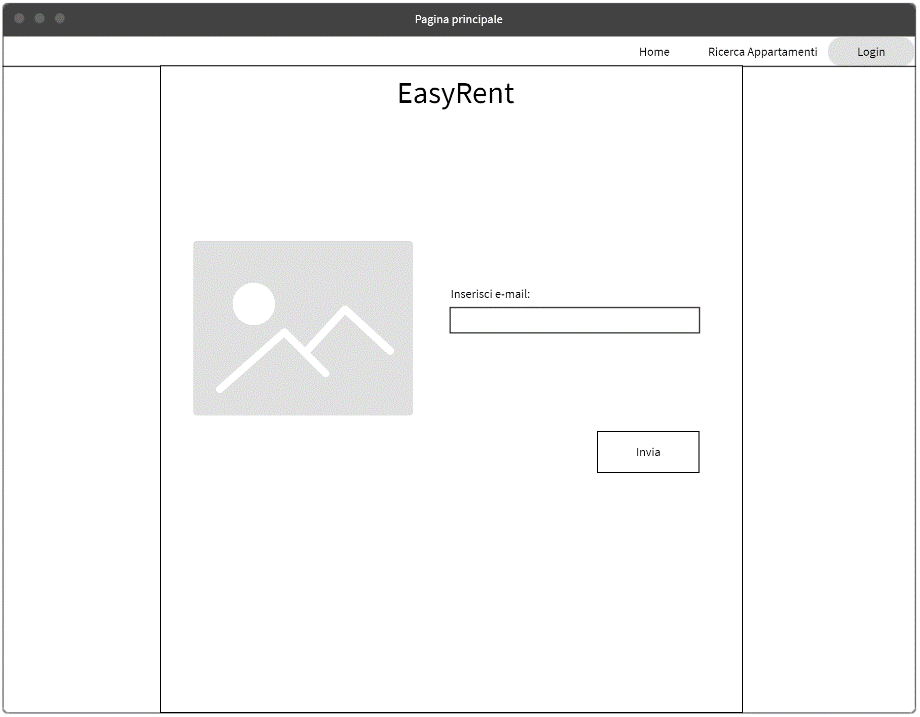
Questa è la pagina di login in cui l’utente ha la possibilità di effettuare l’accesso, inserendo le proprie credenziali oppure ci si può registrare al sito premendo il bottone “Registrati”. Inoltre in caso che l’utente avesse dimenticato la password si può cliccare il link “Hai dimenticato la password?” e si avvia una procedura di ripristino password.

### Pagina di registrazione



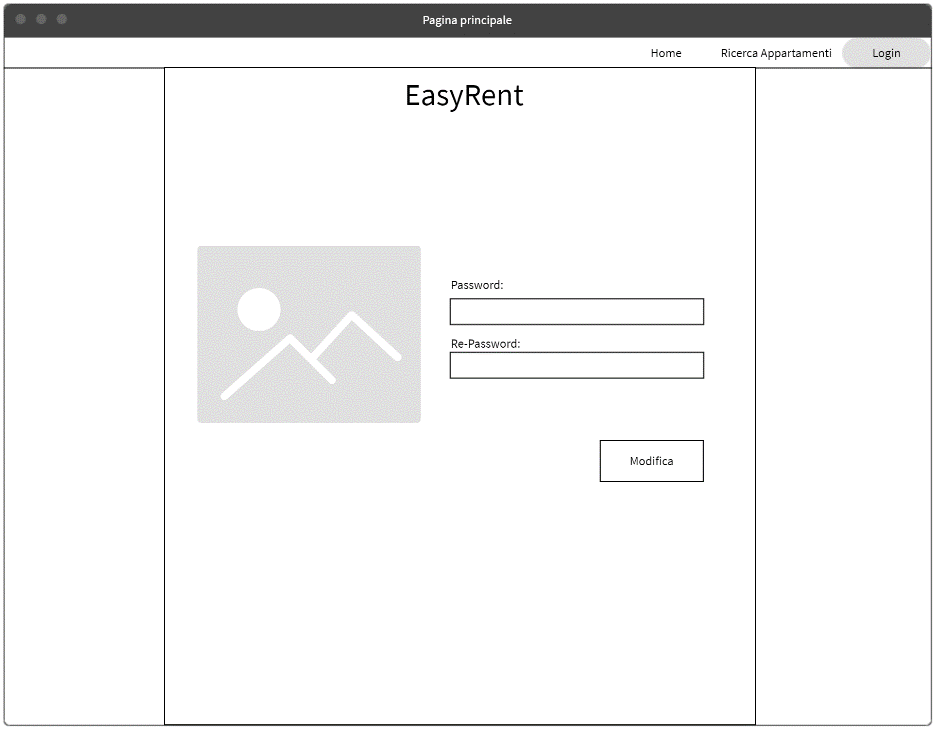
Questa è la pagina di registrazione. La pagina è composta da un form con 10 campi. L’utente finale dovrà compilare tutti i campi con un “\*” di fianco con i propri dati personali e in modo corretto. I campi senza “\*” potranno essere lasciati in bianco. Una volta completata la compilazione gli basterà cliccare il bottone “Registrati” per inviare la richiesta di registrazione al sito.

### Pagina richiesta nuova password



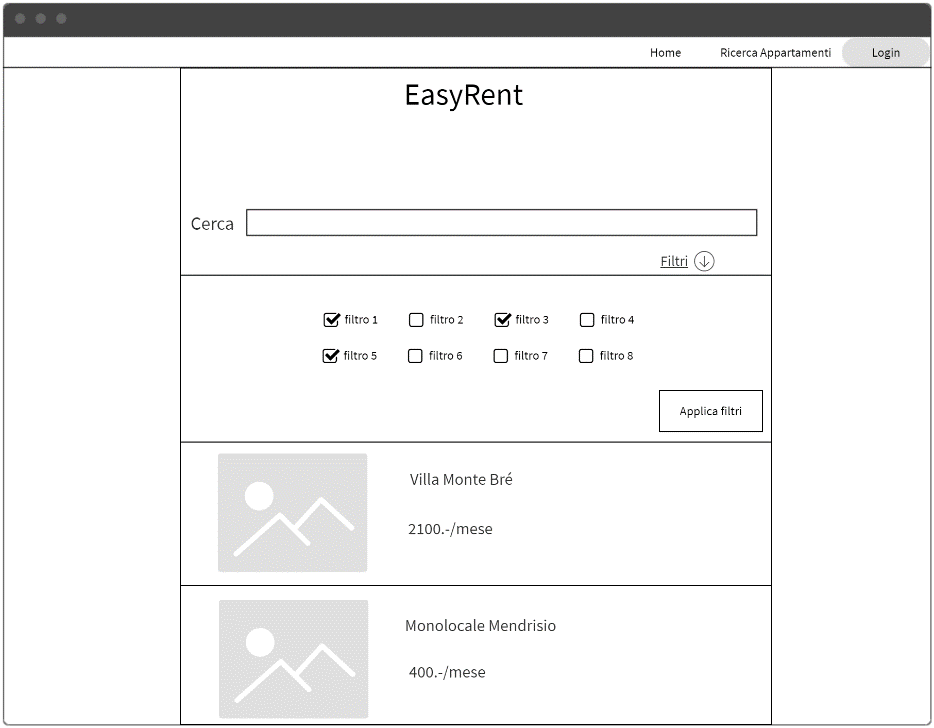
Questa pagina permette di richiedere una procedura per il ripristino della password. L’utente finale raggiunge questa pagina cliccando il link “Hai dimenticato la password?” dalla pagina di Login. Inserendo la propria mail e cliccando il bottone “Invia” si potrà avviare la procedura.

### Pagina reimposta password



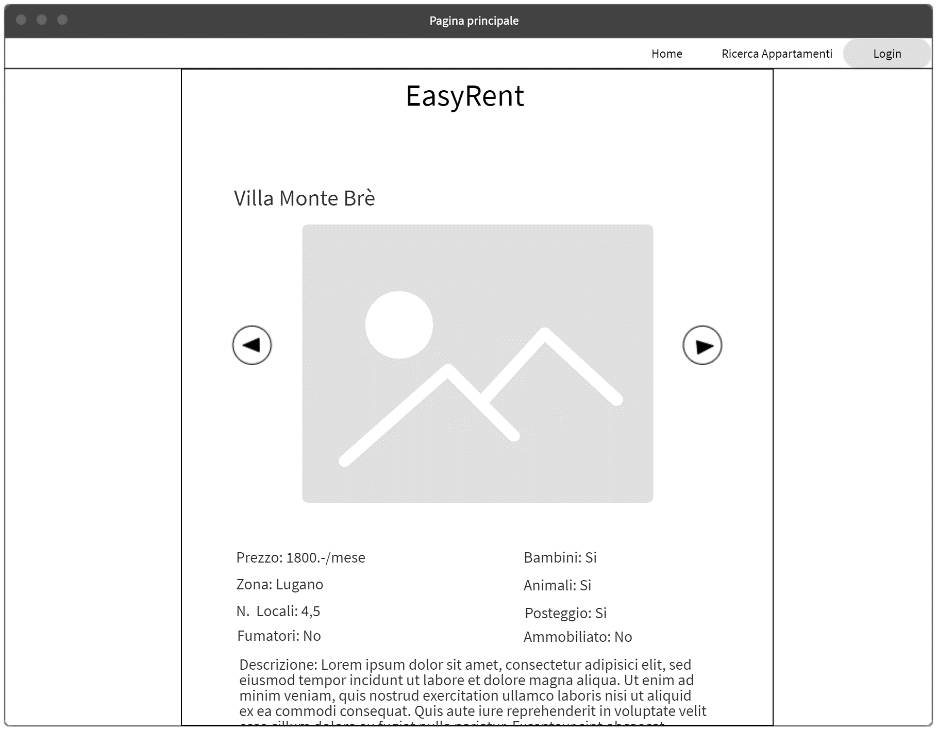
Una volta inviata la richiesta di recupero password, arriverà una e-mail con un link che riporterà a questa pagina. In questa pagina si dovrà inserire due volte la nuova password di accesso. Una volta riempiti i due campi basterà cliccare il bottone “Modifica” e il cambiamento verrà applicato.

### Pagina ricerca appartamenti

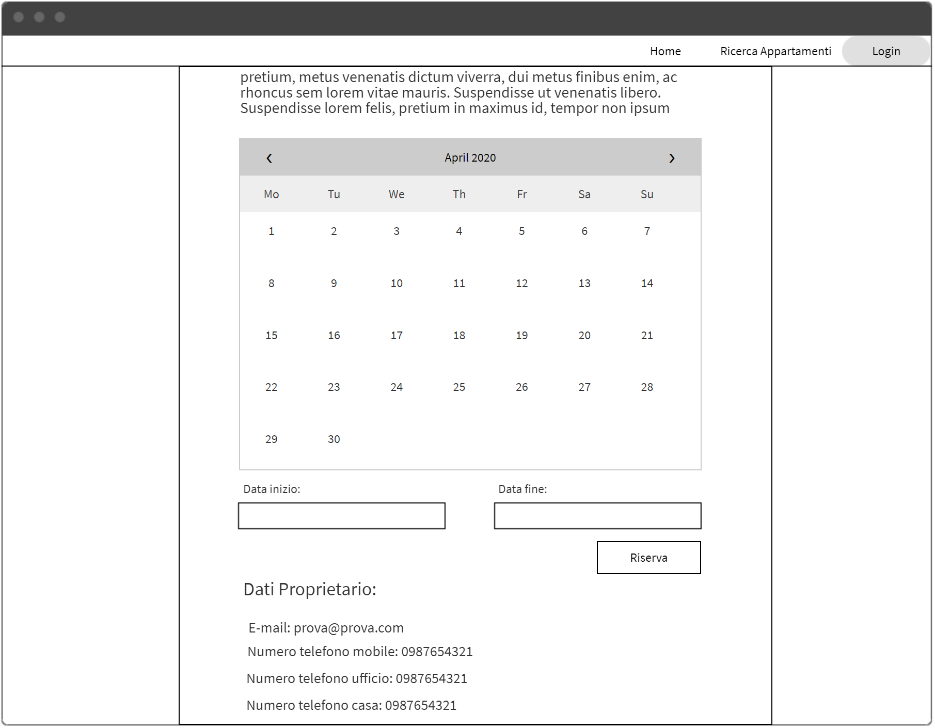


Questa è la pagina dove l’utente finale può ricercare un appartamento. Sopra alla lista degli appartamenti è presente una barra di ricerca che permette di filtrare gli appartamenti in base al loro titolo. Inoltre è presente una spunta che permette di applicare dei filtri per fare delle ricerca più mirate. Infine cliccando su un determinato appartamento l’utente verrà mandato alla pagina dell’immobile specifico.

### Pagina appartamento specifico

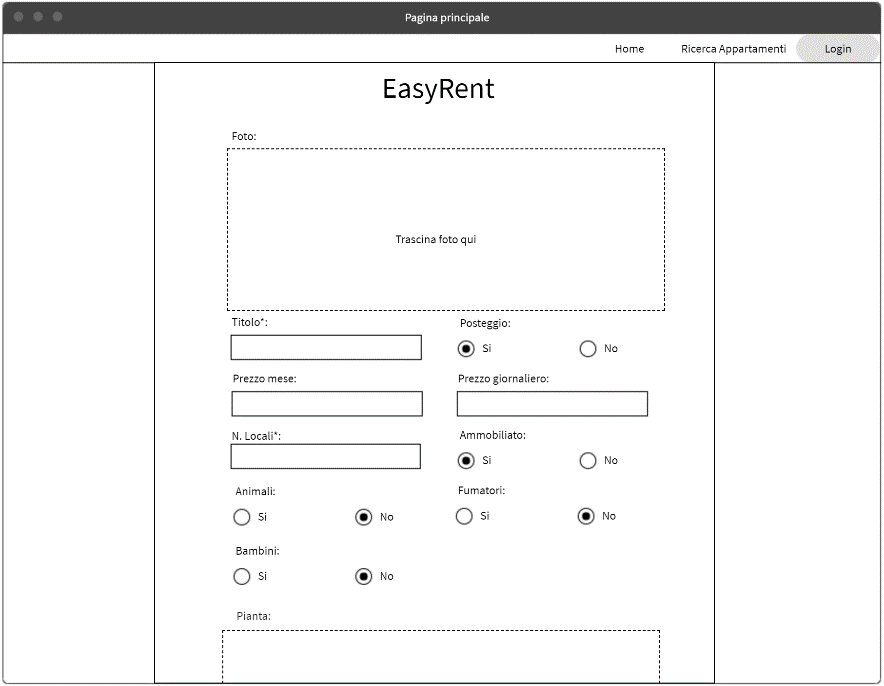


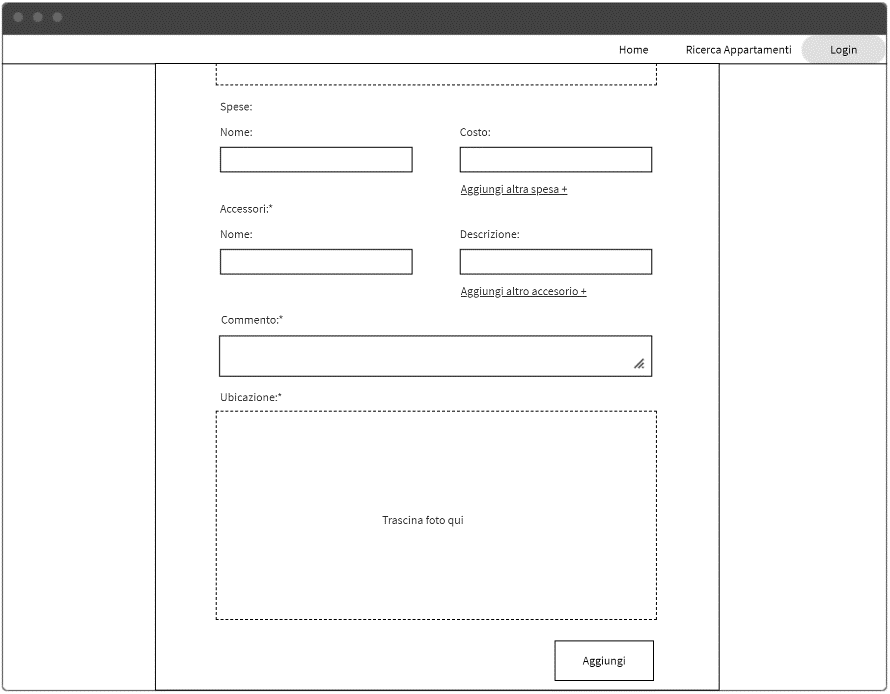
Questa è la pagina per la visualizzazione di un appartamento specifico. In questa pagina si potranno vedere le foto dell’appartamento e varie caratteristiche che offre quest’ultimo.



Scorrendo la pagina di un appartamento specifico si può trovare un calendario che mostra la disponibilità dell’immobile in base alla data. Infine sotto a questo calendario, se si è registrati al sito si può effettuare una riservazione. Infine, sempre se si è registrati, si possono vedere i dati per contattare il proprietario.

### Pagina aggiunta appartamento



Questa è la pagina per l’aggiunta di un appartamento. Essa è raggiungibile solo dagli utenti che sono stati verificati da un amministratore. Come per la pagina di registrazione, i campi obbligatori sono contrassegnati con un “\*”.

Questa è la continuazione della pagina di sopra. Una volta compilati tutti i dati obbligati sarà sufficiente cliccare il bottone “Aggiungi” e l’appartamento sarà messo a disposizione sul sito.

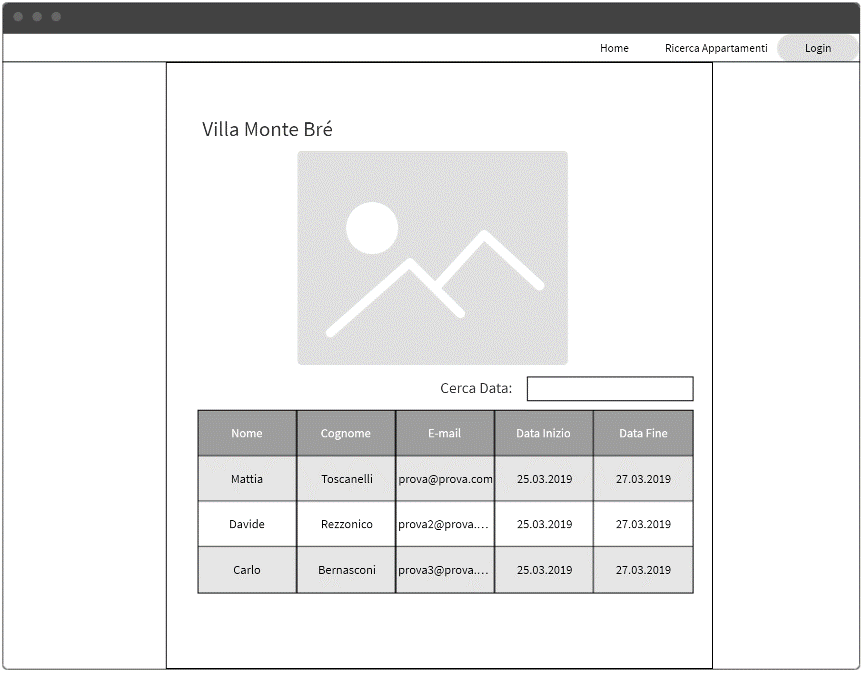
### Pagina gestione appartamento

Questa è la pagina riservata ai proprietari di appartamenti. Da qua si possono vedere tutti gli appartamenti che sono stati aggiunti al sito. Accanto ad ogni appartamento sono presenti tre bottoni. Il bottone “Riserva” permette di aggiungere una riservazione per un cliente che ha chiamato per telefono. Il bottone “Modifica” apre la pagina di aggiunta appartamento (spiegata nel punto 3.3.8) ma con i campi già riempiti con i dati dell’appartamento. Infine c’è il bottone riepilogo che permette di vedere una situazione globale sulle riservazioni dell’appartamento.

### Pagina riservazione appartamento da parte del proprietario

Questa è la pagina per l’aggiunta di una riservazione da parte del proprietario dell’immobile.

### Pagina di riepilogo di un appartamento per i proprietari



Questa è la pagina per visualizzare una situazione globale sulle riservazioni fatte per un appartamento specifico.

### Pagina di amministrazione

Questa è la pagina di amministrazione del sito. Questa non può essere visualizzata se non dagli amministratori dell’applicazione.

Nella prima parte della pagina è possibile effettuare la modifica degli utenti (per esempio verificarli). Scendendo per la pagina si può trovare la sezione di modifica della pagina principale.

## 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-012 | **Nome:** | Import a card with KIC, KID and KIK keys, but not shown with the GUI |
| **Descrizione:** | Import a card with KIC, KID and KIK keys with no obfuscation, but not shown with the GUI | | |
| **Prerequisiti:** | Store on local PC: Profile\_1.2.001.xml (appendix n\_n) and Cards\_1.2.001.txt (appendix n\_n).  PIN (OTA\_VIEW\_PIN\_PUK\_KEY) and ADM (OTA\_VIEW\_ADM\_KEY) user right not set. | | |
| **Procedura:** | 1. Go to “Cards manager” menu,  in main page click “Import Profiles” link, Select the “1.2.001.xml” file, Import the Profile 2. Go to “Cards manager” menu,  in main page click “Import Cards” link, Select the “1.2.001.txt” file, Delete the cards,  Select the “1.2.001.txt” file, Import the cards 3. Research the “41795924770” Card, Click the imsi card link Check the card details 4. Execute the SQL: SELECT imsi, dir, keyset, cntr, rawtohex(kickey), rawtohex(kidkey), rawtohex(kikkey), rawtohex(chv), rawtohex(dap)FROM otacardkey a where imsi='340041795924770' ORDER BY keyset; | | |
| **Risultati attesi:** | Keys visible in the DB (OtaCardKey) but not visible in the GUI (Card details) | | |

## 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 Gannt 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

## Sitografia

<https://www.draw.io/>, Schemi/Diagrammi Progetto, 31.01.2020

<https://www.mockflow.com/>, Mockup delle interfacce, 03.02.2020

# 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
* …