Gestione disponibilità alloggi

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 3

Analisi 4

1.4 Analisi del dominio 4

1.5 Analisi e specifica dei requisiti 4

1.6 Pianificazione 5

1.7 Analisi dei mezzi 6

1.7.1 Software 6

1.7.2 Hardware 6

2 Progettazione 7

2.1 Installazione ambiente di sviluppo 7

2.1.1 Librerie 7

2.1.2 Scatola del word clock 8

2.2 Analisi e verifica funzionamento componenti 10

2.2.1 Striscia di led 10

2.2.2 RTC 11

2.3 Design dell’architettura del sistema 12

3 Implementazione 13

3.1 Definizione dei parametri 13

3.2 Setup() 15

3.3 getPacket() 17

3.4 pixelOn() 18

3.5 generateWord() 18

3.6 Generazione parole 19

3.7 printBreak() 19

3.8 generateSeconds() 20

3.9 printSecond() 20

3.10 printTime() 21

4 Test 24

4.1 Protocollo di test 24

4.2 Risultati test 25

4.3 Mancanze/limitazioni conosciute 25

5 Consuntivo 26

6 Conclusioni 27

6.1 Sviluppi futuri 27

6.2 Considerazioni personali 27

7 Bibliografia 28

7.1 Bibliografia per libri 28

7.2 Sitografia 28

8 Allegati 28

# Introduzione

## Informazioni sul progetto

* Allievo: Mattia Lazzaroni.   
  Superiore professionale: Guido Montalbetti.
* Scuola d’Arti e Mestieri di Trevano, sezione Informatica, classe 5, progetti individuali.
* Data inizio: 03.09.2019  
  Data fine: 20.10.2019

## Abstract

*A word clock is a clock that shows the time by using spoken language. In this project a box is used where below there is a strip of led and above there is a sheet with the words that together form the time. This strip is programmed in Arduino so that the led that form the words of the current time are turned on. The time is obtained from a time server and set up by a RTC, otherwise it can be configured directly from the box using physical button.   
The final product can be considered as a prototype, since the main idea is to expose a large word clock on the roof of the study center.*

## Scopo

Lo scopo di questo progetto è quello di creare un sito Web che permetta agli utenti di visualizzare e riservare degli alloggi. Innanzitutto, l’utente deve poter trovare degli alloggi disponibili in base ai filtri che andrà ad inserire. Una volta scelto il proprio alloggio, se volesse riservarlo, l’utente dovrà registrarsi andando a riempire dei campi minimi. Per confermare la registrazione, verrà inviato un link di conferma per e-mail. Inoltre, l’utente dovrà confermare la riservazione tramite e-mail. Ci dovrà anche essere una sezione riservata unicamente agli amministratori e una riservata agli amministratori gerenti. Dopo aver riservato un alloggio l’utente potrà stampare la propria riservazione con tutti i dettagli. Gli amministratori gerenti avranno la possibilità di stampare le riservazioni di una certa persona o di un certo periodo. Ci sarà inoltre una parte per la gestione delle fatturazioni.

# Analisi

## Analisi del dominio

Spesso per poter trovare un alloggio in cui passare la notte occorre girare molti siti perché quelli presenti sono incompleti e non forniscono tutte le informazioni oppure sono troppo complicati e risultano difficili da usare. Il mio prodotto servirà a quegli utenti che cercano un sito semplice ed intuitivo con cui riservare un alloggio in pochi minuti. Per poter accedere al mio sito sarà necessario come minimo un dispositivo con accesso ad Internet, un’e-mail, un recapito telefonico e una conoscenza basilare per poter usare un sito Web munito di registrazione e conferma tramite e-mail.

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Verifica dell’hosting |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare che l’hosting fornito dalla scuola funzioni correttamente. |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Creazione menu amministratori |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario creare una sezione riservata agli amministratori del sito |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| **Nome** | Creazione menu amministratori gerenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario creare una sezione riservata agli amministratori gerenti. |

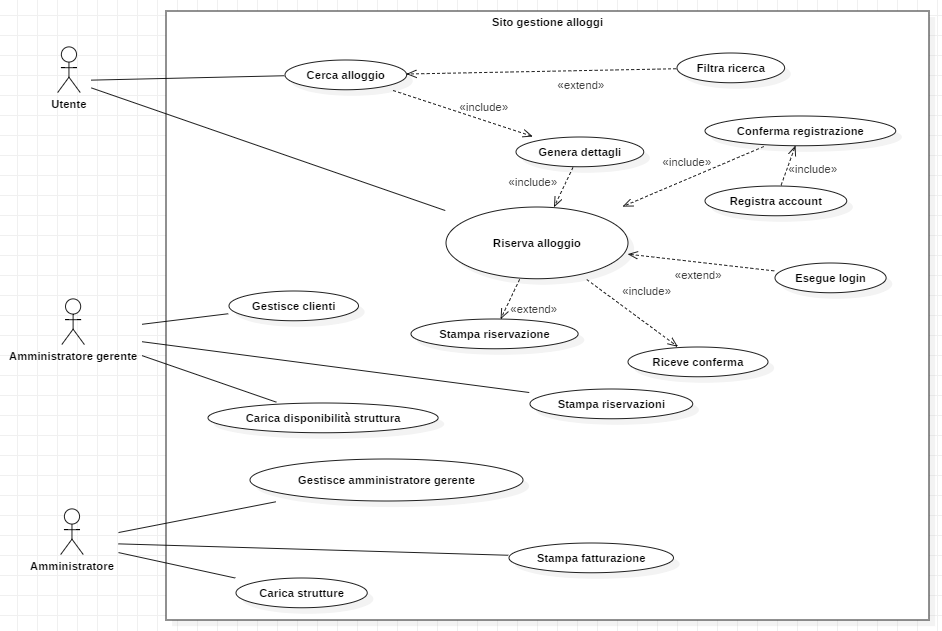
|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Stampa degli alloggi |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario creare una sezione riservata agli amministratori del sito |
| **Sotto requisiti** | |
| **001** | Sarà possibile filtrare i risultati |
| **002** | Sarà possibile visualizzare i dettagli degli alloggi |
| **003** | Sarà possibile fare una riservazione di un alloggio |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | Fornire la possibilità di registrarsi |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Se si vuole svolgere una registrazione è necessario registrarsi al sito |
| **Sotto requisiti** | |
| **001** | Sarà necessario anche un login |
| **002** | Bisognerà mandare per mail un link per confermare la registrazione |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Gestire il sistema di stampa |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Coloro che riservano un alloggio possono stampare la riservazione con tutti i dettagli. |
| **Sotto requisiti** | |
| **001** | Gli amministratori del sito potranno stampare le fatturazioni di un certo periodo e per una struttura. |
| **002** | Gli amministratori gerenti potranno stampare le riservazioni di un certo periodo o di una certa persona. |

|  |  |
| --- | --- |
| **ID: REQ-007** | |
| **Nome** | Gestire le fatturazioni |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Il sito deve prevedere di poter gestire la fatturazione delle strutture. |
| **Sotto requisiti** | |
| **001** | Il sito guadagna tramite le riservazioni |
| **002** | Vi deve essere la possibilità di modificare le percentuali percepite |

## Use case



Use Case del sito web

Nel seguente Use Case Diagram troviamo tre attori: utente, amministratore gerente e amministratore. Per quanto riguarda l’amministratore e l’amministratore gerente non mi soffermerò a spiegare, in quanto hanno entrambi solo tre Use Case facilmente comprensibili tramite l’immagine qua sopra. L’utente invece è l’attore che richiede spiegazioni in quanto complicato. Egli può riservare un alloggio, tuttavia per compiere questa azione deve prima cercarlo obbligatoriamente. Durante la ricerca, volendo, può filtrare i risultati. Questa ricerca produrrà degli alloggi che se cliccati mostreranno una pagina con i dettagli. È proprio da questa pagina dei dettagli che è possibile svolgere la riservazione. Per poter riservare, l’utente deve registrarsi al sito web e deve confermare la registrazione tramite e-mail. In seguito, se l’utente dovesse tornare al sito non dovrà più registrarsi, bensì loggarsi.

## Pianificazione

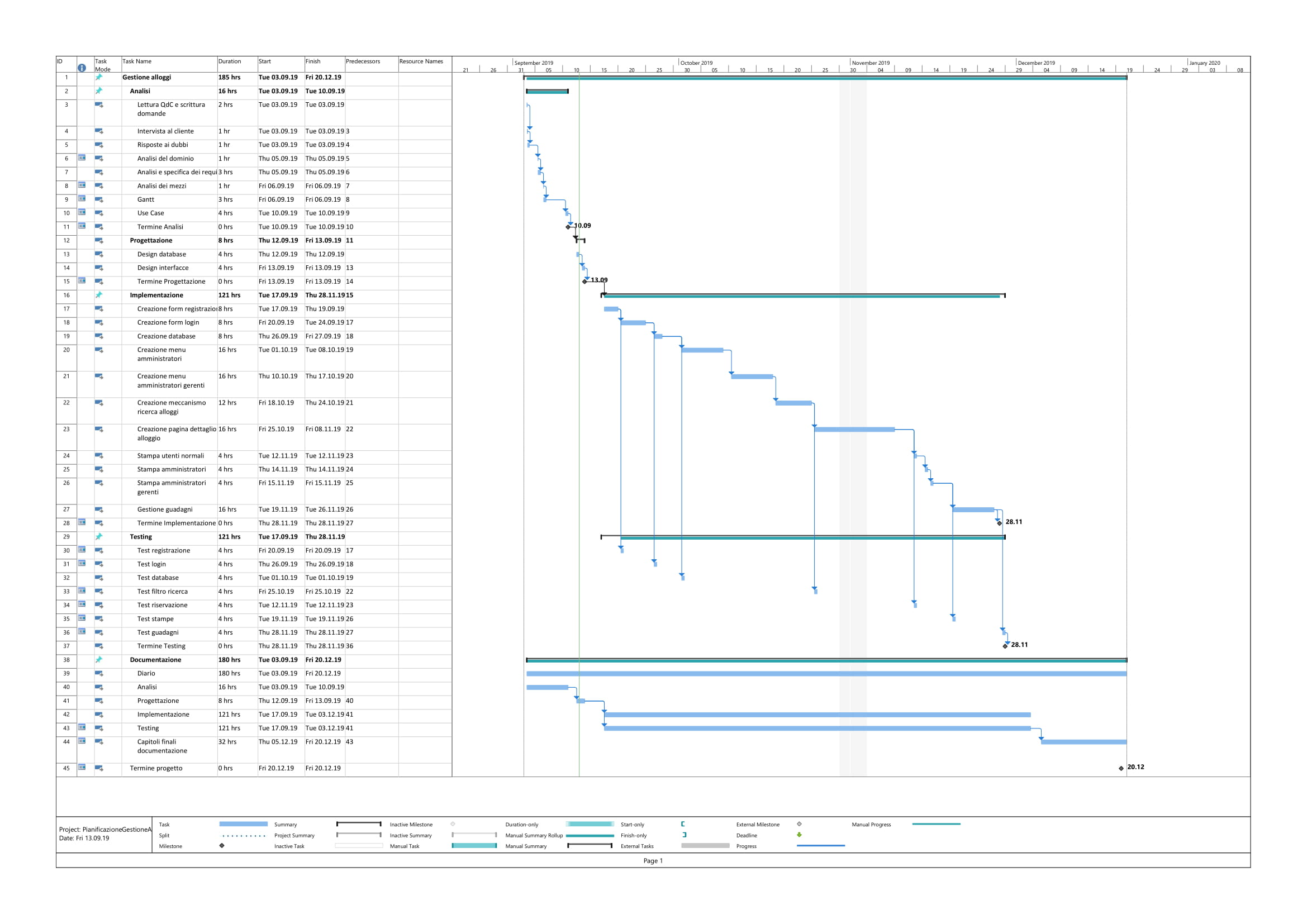
La pianificazione del progetto è stata effettuata mediante la realizzazione di un diagramma di Gantt.

Diagramma di Gantt rappresentante la pianificazione del progetto

Inizialmente ho pianificato un’analisi di 12 ore, in cui vado a leggere il QdC e a porre le domande al mio responsabile. Successivamente vado ad analizzare il dominio, i requisiti e i mezzi a disposizione. Infine, realizzo il Gantt e l’analisi termina qui, con una milestone. Dopo di questo, inizio la progettazione, che dura 12 ore come l’analisi. Durante questa fase vado a progettare l’architettura, il database e le interfacce. Finita la progettazione ho inserito un’altra milestone. Successivamente inizia l’implementazione, che è la fase che dura di più in cui ho diviso tutte le varie attività per poter realizzare il vero e proprio sito web. Anche la fine di questa fase è segnata da una milestone. Più sotto è possibile notare tutti i vari test che vengono svolti sull’applicazione e la milestone di fine fase. Infine, come ultima attività possiamo vedere la documentazione, che dura tutto il progetto e contiene i diari giornalieri e le 4 fasi principali. All’ultima riga è presente la milestone che indica la fine del progetto.

## Analisi dei mezzi

### Software

I software utilizzati per la realizzazione di questo progetto sono:

* Microsoft Word 2016
* Microsoft Power Point 2016
* Microsoft Project Professional 2016
* Google Chrome 71.0.3578.98
* GitHubDesktop 2.1.3
* Visual Studio Code 1.34.0

### Hardware

* Computer Mattia Lazzaroni:
  + Modello: Acer Aspire E 15
  + Processore: Intel Core i7 7500u
  + RAM: 16GB LPDDR4 2133 MHz
  + GPU: NVIDIA 940MX
  + SSD: 256 GB

# Progettazione

Questo capitolo descrive esaustivamente come deve essere realizzato il prodotto fin nei suoi dettagli. Una buona progettazione permette all’esecutore di evitare fraintendimenti e imprecisioni nell’implementazione del prodotto.

## 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

Descrizione delle strutture di dati utilizzate dal programma in base agli attributi e le relazioni degli oggetti in uso.

Schema E-R, schema logico e descrizione.

Se il diagramma E-R viene modificato, sulla doc dovrà apparire l’ultima versione, mentre le vecchie saranno sui diari.

## Design delle interfacce

Descrizione delle interfacce interne ed esterne del sistema e dell’interfaccia utente. La progettazione delle interfacce è basata sulle informazioni ricavate durante la fase di analisi e realizzata tramite mockups.

## 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

## 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
* …