Realizzazione di un plotter verticale da parete basato su Arduino

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 5

1.6 Use case 7

1.7 Pianificazione 7

1.8 Analisi dei mezzi 7

1.8.1 Software 7

1.8.2 Hardware 7

1.9 Analisi dei costi e dei benefici 8

2 Progettazione 9

2.1 Design dell’architettura del sistema 9

2.2 Design dei dati e database 9

2.3 Design delle interfacce 9

2.4 Design procedurale 9

3 Implementazione 10

3.1 Software 10

3.2 Hardware 11

4 Test 12

4.1 Protocollo di test 12

4.2 Risultati test 12

4.3 Mancanze/limitazioni conosciute 12

5 Consuntivo 13

6 Conclusioni 13

6.1 Sviluppi futuri 13

6.2 Considerazioni personali 13

7 Bibliografia 13

7.1 Bibliografia per articoli di riviste: 13

7.2 Bibliografia per libri 13

7.3 Sitografia 13

8 Allegati 13

# Introduzione

## Informazioni sul progetto

In questo capitolo raccogliere le informazioni relative al progetto, ad esempio:

* Allievi: Gabriel Mendonça, Alessandro Colugnat, Jonas Bertossa, Elia Manassero
* Docenti: Francesco Mussi, Massimo Sartori
* Scuola: SAMT
* Sezione: Informatica
* Materia: 306
* Data inizio: 10.11.2.17
* Data consegna: 26.01.2018
* Data fine:
* Nome Progetto: Realizzazione di un plotter verticale da parete basato su Arduino

## Abstract

Every 2 years in Espoprofessioni the stand of SAMT introduce a project who Is made by a student in SAMT.

This year at the stand of Espoprofessioni want to introduce the project named “vertical wall plotter”.

For create this project we use Arduino and PHP the hardware we use two stepper motor and Arduino Uno.

The result of this project is a wall vertical plotter with a Wi-Fi connection for take the picture and print it with the plotter.

## Scopo

Bisogna creare un plotter verticale da parete con Arduino, c’è una penna agganciata a un supporto, questo supporto è attaccato a due fili di nylon e si muoverà grazie a due motori che gli permettono un movimento su tutto il piano di lavoro. Il tutto viene fatto funzionare da Arduino.

Il plotter disegna un immagine vettorizzata che gli viene passata tramite collegamento Wi-Fi

E il i tre motori lavoreranno in modo sincrono per spostare il pennarello e attaccarlo o staccarlo dal muro.

## Analisi

## Analisi del dominio

Ci è stato richiesto di presentare un Wall Plotter verticale a Espoprofessioni 2018.

Ci sono già vari modelli di Wall Plotter.

Per utilizzare il nostro Plotter verticale bisogna usare immagini vettorizzate, gli utenti potranno usare il wall plotter per fare disegni, il nostro progetto si può adattare a molte superfici, e questo lo rende molto utile in molte situazioni.

Il sito in cui si carica l’immagine è molto semplice, quindi non richiede nessuna competenza tecnica nell’ambito informatico. Esistono altri tipi di Wall plotter e hanno tutti una base diversa c’è chi usa 4 motori su tutti i lati oppure si usa una calamita e viene tutto disegnato su una lavagna magnetica.

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **ID: RQ-01** | |
| **Nome** | Arduino |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Deve comunicare con il wifi |
| **002** | Deve gestire 2 motori per il filo e 1 per la penna |

Questo requisito è stato sostituito qui sottostante perché il membro del gruppo che doveva fare questa operazione non è più venuto a lavorare.

|  |  |
| --- | --- |
| **ID: RQ-01** | |
| **Nome** | Arduino |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Deve venire passato un file tramite scheda sd |
| **002** | Deve gestire 2 motori per il filo e 1 per la penna |

|  |  |
| --- | --- |
| **ID: RQ-02** | |
| **Nome** | Interfaccia web |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Deve essere online |
| **002** | Deve passare l'immagine vettorizzata con l'arduino tramite wifi |
| **003** | Deve ricevere un immagine e vettorizzarla |
| **004** | Deve mostrare l'immagine vettorizzata |

|  |  |
| --- | --- |
| **ID: RQ-03** | |
| **Nome** | Struttura |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Deve essere trasportabile |
| **002** | Si deve adattare |

**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.

## Pianificazione

Gannt:

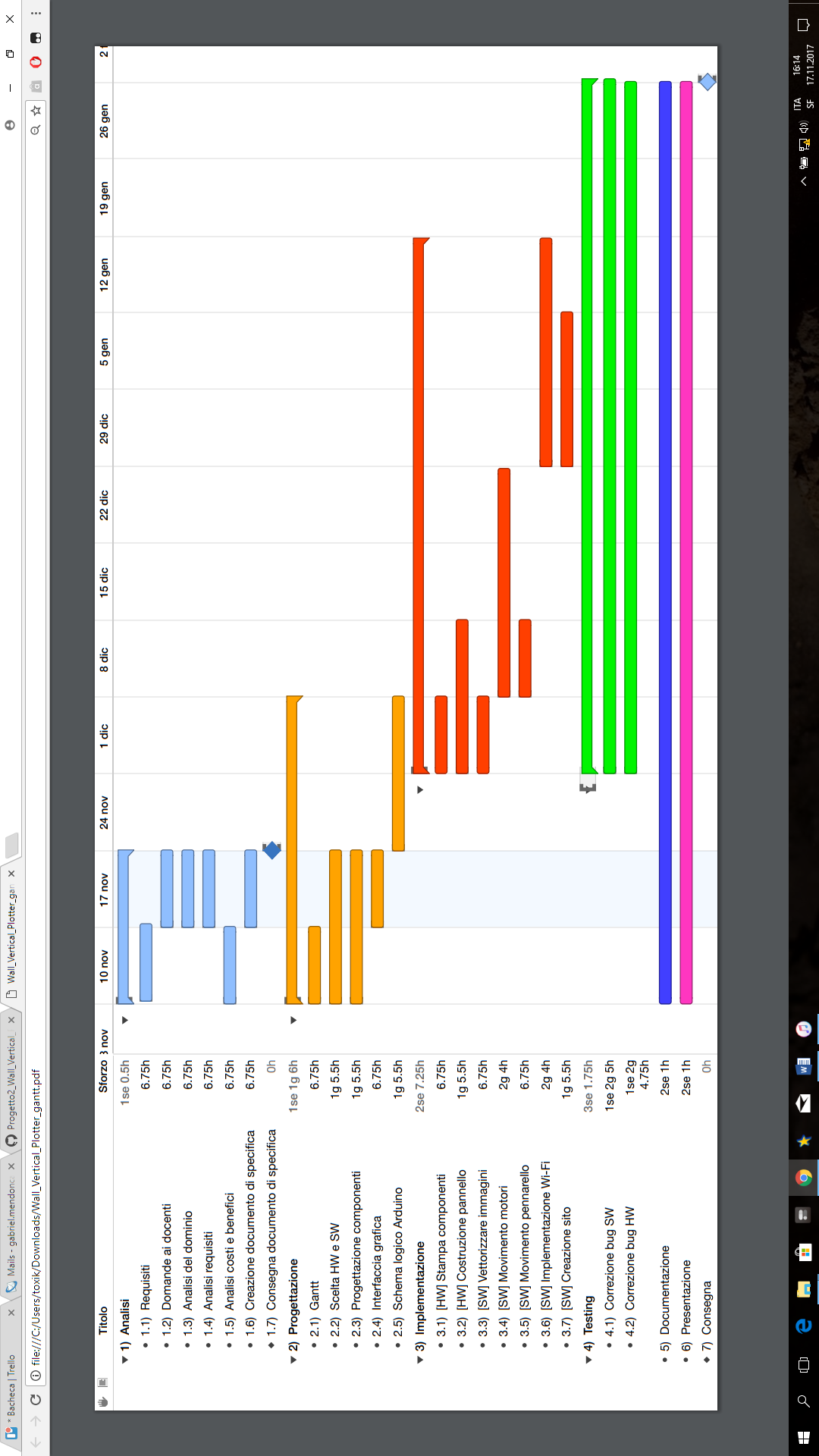


Figura 1: Gannt Preventivo

## Analisi dei mezzi

### Software

Applicazione per vettorizzare un immagine.

Arduino.

Apache

Windows 7

### Hardware

Arduino e RaspBerry

Una scheda di rete Wi-Fi per Arduino

Motori per muovere il filo e il sostegno e un motore per allontanare o avvicinare il pennarello al pannello.

Filo di nylon.

Sostegno per il pennarello fatto con la stampante 3D.

Dei pesi da appendere ai lati.

Fogli e pennarelli.

Una base di legno.

Nastro adesivo per appendere i fogli.

Gancio per appoggiare l’Arduino in modo comodo.

2 ganci per appendere i motori ai lati.

Una batteria per Arduino.

Cavi per vari collegamenti.

Computer per far funzionare Arduino

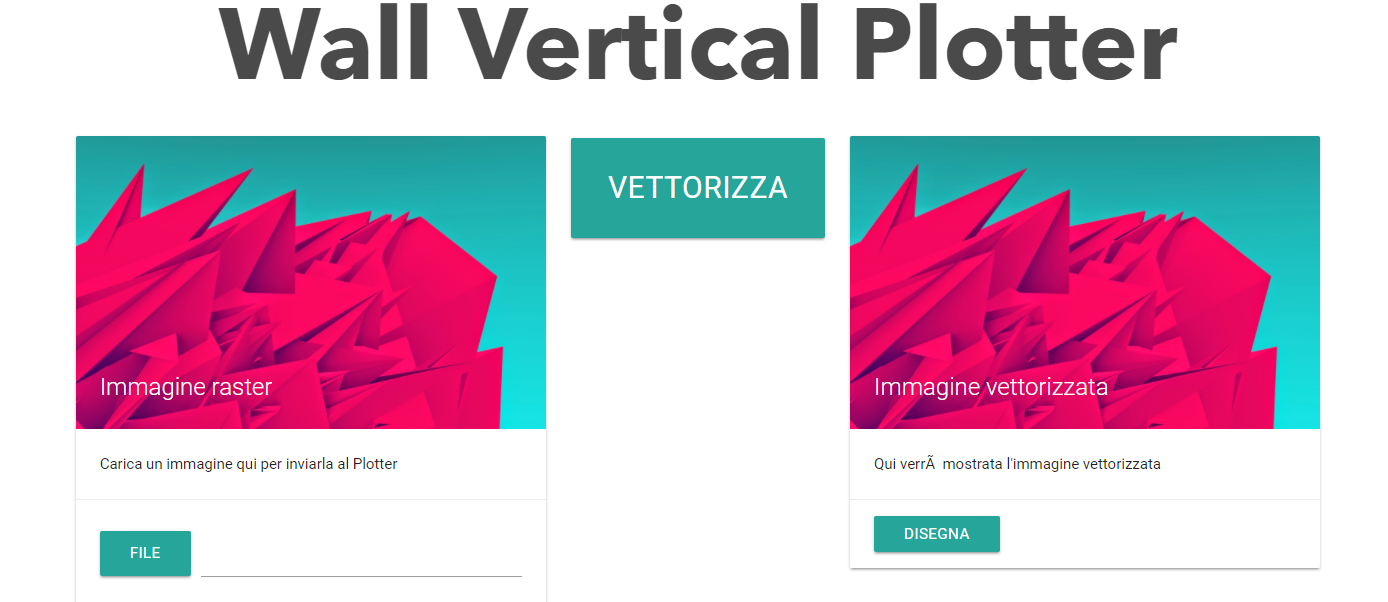
Server apache per trasferire l’immagine

## Analisi dei costi e dei benefici

|  |  |  |
| --- | --- | --- |
| Componenti | Prezzo | Ore |
| 4 persone | 62 fr.-/ora | 54 |
| 2 Stepper motor – 68 oz-in (400 steps/rev) | 53,80 fr.- | - |
| 2 A4998 Stepper Motor Driver Carrier - Stepstick | 13,00 fr.- | - |
| Adafruit motor/stepper/servo shield for Arduino v2 kit | 22,00 fr.- | - |
| Arduino stackable Header Kit(4 pcs) - stepstick | 2,65 fr.- | - |
| **Totale** | **13483,45 fr.** |  |

# Progettazione

## Design delle interfacce



Premendo il tasto “file” sulla sinistra si può scegliere un’immagine dal proprio PC, oppure nella casella di testo a lato si può inserire il percorso dell’immagine sulla sinistra. Premendo vettorizza, un programma preesistente vettorizza l’immagine selezionata e fa vedere l’anteprima dell’immagine vettorizzata al posto dell’immagine sulla destra. Premendo disegna parte il disegno sulla lavagna

# Implementazione

## Software

Per leggere far leggere la microSD, comprendere il g-code e disegnare all’fishino è stato scritto un programma con Arduino IDE.

La lettura della scheda microSD avviene nel setup() del codice, dove viene letto il file con le coordinate in

G-code con il metodo File.read() che ritorna int e legge un carattere per volta, per questo il metodo si trova dentro un ciclo e il dato letto viene convertito in char.

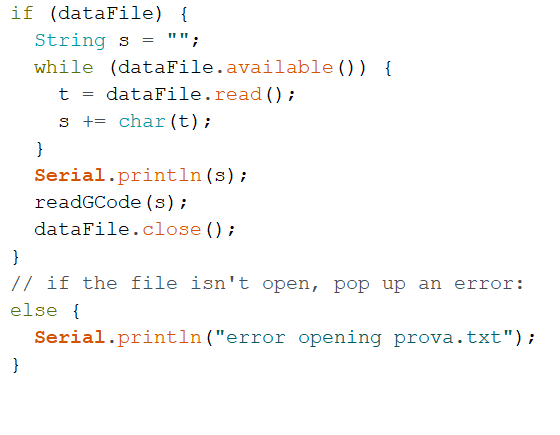


Figura 2: Codice di lettura

Alla fine della lettura viene invocato il metodo readGcode(String) che si occupa di comprendere il codice e far muovere i motori nelle posizioni richieste.

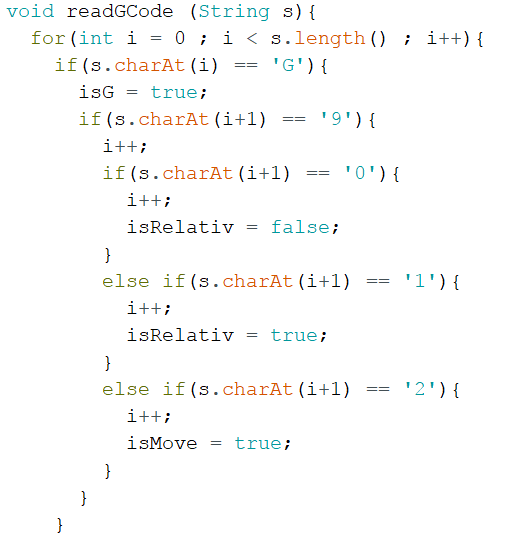


Figura 3: Metodo readGCode

Prima di tutto cerca di capire se i primi caratteri sono G90, che imposta le coordinate in relative, G91, che imposta le coordinate in assolute, e G92, che definisce una nuova posizione.

Dopo comincerà a leggere le coordinate X, Y e Z che in G-Code serve a definire quanto filo bisogna usare nelle stampanti 3D ma il codice lo userà per capire quanto deve disegnare mentre si muove. Quando il programma trova un ‘;’ chiame il metodo paint(int, int, int) e dopo ripete il ciclo.

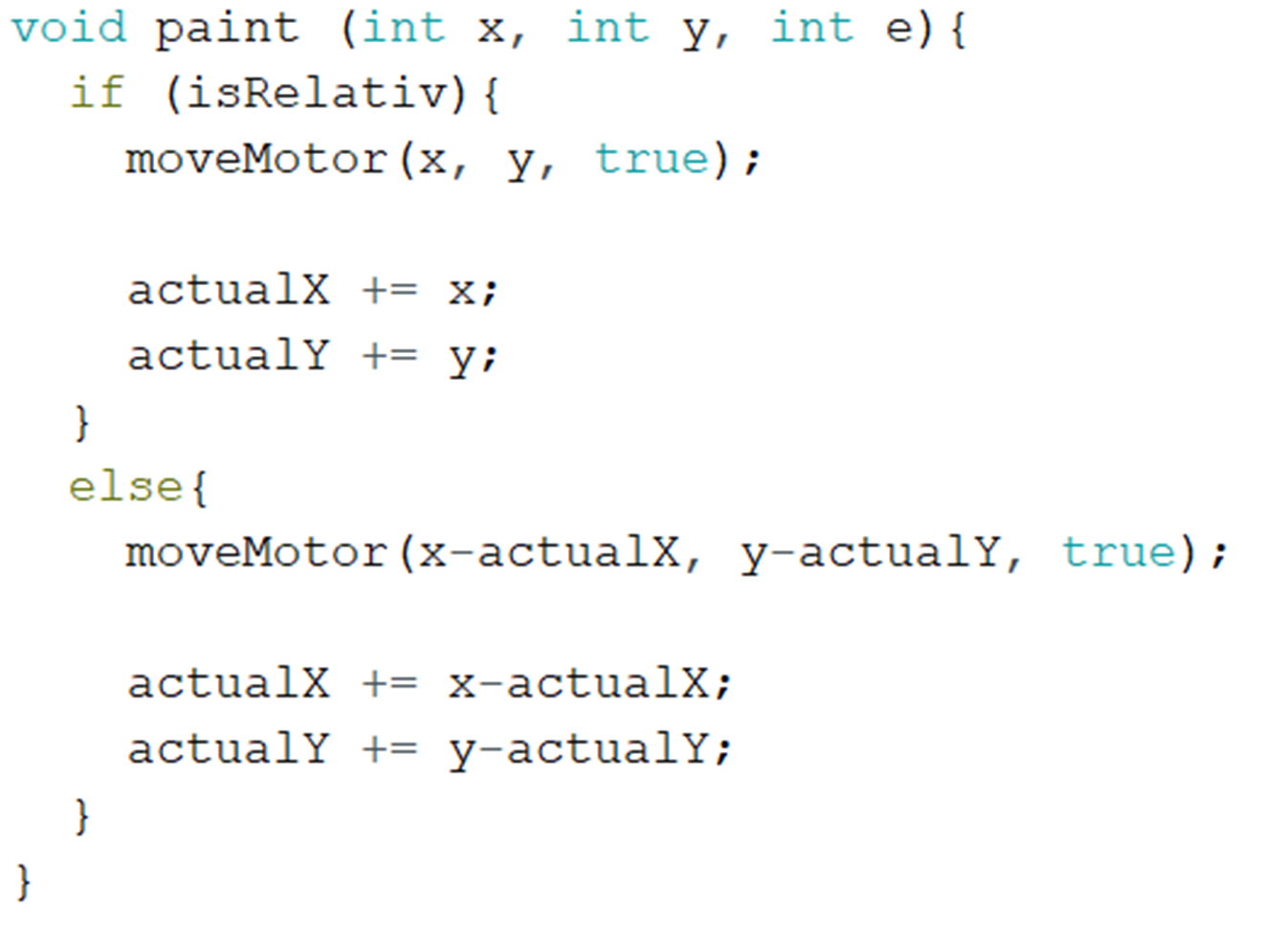


Figura 4: metodo paint

Paint richiama moveMotor(int, int, bool) passandoli le coordinate tenendo conto se sono relative o meno e cambia la posizione attuale. Il metodo moveMotor ha il compito di muove i motori secondo le coordinate.

## Hardware

# 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

Simplify3D (https://www.simplify3d.com/support/articles/3d-printing-gcode-tutorial/)

**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
* …