Documentazione ImageDot

# Indice

1 Indice 2

2 Introduzione 3

2.1 Informazioni sul progetto 3

2.2 Abstract 3

2.3 Scopo 3

3 Analisi 4

3.1 Analisi del dominio 4

3.2 Analisi e specifica dei requisiti 4

3.3 Use case 7

3.4 Pianificazione 7

3.5 Analisi dei mezzi 9

3.5.1 Software 9

3.5.2 Hardware 9

4 Progettazione 9

4.1 Design dell’architettura del sistema 9

4.2 Design dei dati e database 9

4.3 Design delle interfacce 12

4.4 Design procedurale 12

5 Implementazione 13

6 Test 13

6.1 Protocollo di test 13

6.2 Risultati test 15

6.3 Mancanze/limitazioni conosciute 15

7 Consuntivo 15

8 Conclusioni 17

8.1 Sviluppi futuri 17

8.2 Considerazioni personali 17

9 Bibliografia 18

9.1 Bibliografia per articoli di riviste: 18

9.2 Bibliografia per libri 18

9.3 Sitografia 18

10 Glossario 19

11 Indice delle figure 19

12 Allegati 19

# Introduzione

## Informazioni sul progetto

Programma dove un utente può caricare un’immagine e su due diversi layer può scegliere quali tratti tenere e nel secondo invece dove mettere i puntini. Così facendo si crea un’immagine per il gioco Unisci i puntini che successivamente si può esportare come immagine, inoltre viene creata anche la soluzione se fosse necessaria.

* Allievo: Luis Dhungana Docente: Geo Petrini
* CPT Trevano, I3BB, Modulo 306
* 01.09.23-01.12.23

## Abstract

È 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 del progetto (scopi didattici/scopi operativi). Dovrebbe descrivere il mandato, ma non vanno ricopiate le informazioni del quaderno dei compiti (che va invece allegato).

# Analisi

## Analisi del dominio

L’applicativo verrà utilizzato maggiormente dai genitori per i loro figli. Principalmente si vuole aiutare l’utente evitando ricerche troppo lunghe e a volte anche non produttive. Grazie a questo programma l’utente potrà trasformare la propria immagine nel gioco dell’Unisci i puntini

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **Req-01** | |
| **Nome** | Grafica programma/sito |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Grafica funzionale e gradevole da vedere |
| **Sotto requisiti** | |
| **Req-01\_1** |  |

|  |  |
| --- | --- |
| **Req-02** | |
| **Nome** | Input Image |
| **Priorità** | 3 |
| **Versione** | 1.0 |
| **Note** | Accetta Immagini in webp, jpeg, png |
| **Sotto requisiti** | |
| **Req-02\_1** | Controllare se l’immagine passata è accettata |
| **Req-02\_2** | Ridimensionare l’immagine se è troppo grande |

|  |  |
| --- | --- |
| **Req-03** | |
| **Nome** | Strumenti di lavoro |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | Penna, disegno, cerchio, ovale, righe, mano libera, forme, quadrati, rettangolo |
| **Sotto requisiti** | |
| **Req-03\_1** | Creare menu bar per gli strumenti |

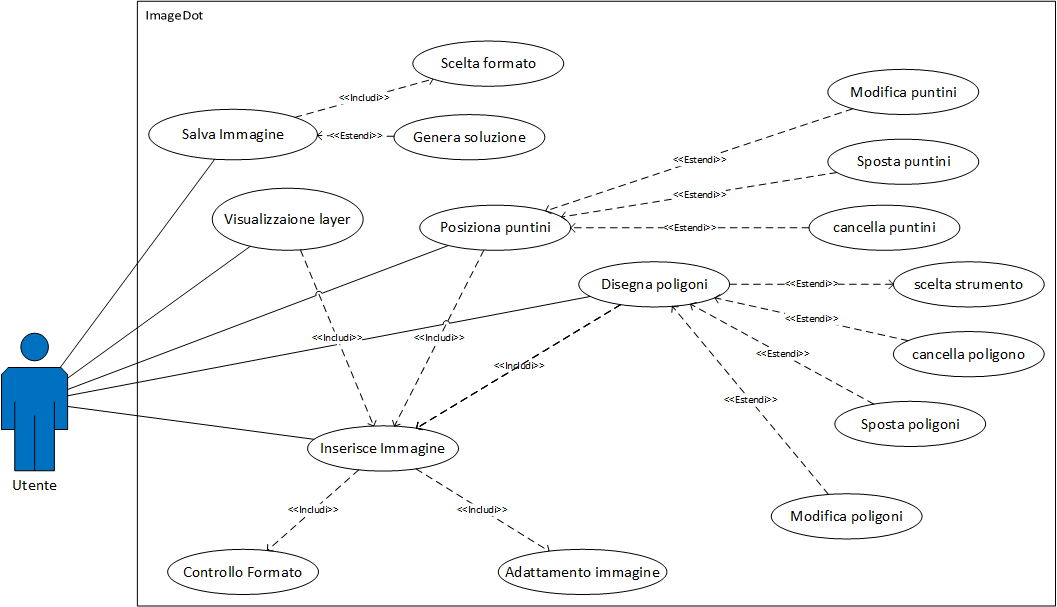
|  |  |
| --- | --- |
| **Req-04** | |
| **Nome** | Posizionamento/modifica puntini |
| **Priorità** | 2 |
| **Versione** | 1.0 |
| **Note** | Poter scegliere dove posizionare i puntini e in che ordine |
| **Sotto requisiti** | |
| **Req-04\_1** | Poter cambiare il numero dei puntini |
| **Req-04\_2** | Poter cancellare i puntini |
| **Req-04\_3** | Poter spostare puntini |
| **Req-04\_4** | Numerati automaticamente |

|  |  |
| --- | --- |
| **Req-05** | |
| **Nome** | Layer |
| **Priorità** | 3 |
| **Versione** | 1.0 |
| **Note** | L’immagini, i puntini e le righe visibili sono su layer differenti |
| **Sotto requisiti** | |
| **Req-05\_1** | L’utente può scegliere quale layer visualizzare |

|  |  |
| --- | --- |
| **Req-06** | |
| **Nome** | Soluzione |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Il programma deve fornire la soluzione dell’immagine con tutti i puntini uniti |
| **Sotto requisiti** | |
| **Req-06\_1** | Richiesta all’utente se si vuole la soluzione |
| **Req-06\_2** | Esportazione di una seconda immagine con la soluzione |

|  |  |
| --- | --- |
| **Req-07** | |
| **Nome** | Output |
| **Priorità** | 3 |
| **Versione** | 1.0 |
| **Note** | Immagine di output può essere in png o svg |
| **Sotto requisiti** | |
| **Req-07\_1** | Poter scegliere in che formato esportare |

## Use case



Questo è lo use case sulla quale mi sono basato per la progettazione della mia applicazione.

## Pianificazione

La durata del progetto, come si può anche verde nell’immagine alla pagina successiva, è di 3 mesi. Di questi avevo 8 ore il venerdì per lavorare al progetto per un totale di 94 ore di lavoro a disposizione, di cui 8 tenute libere in caso di imprevisti durante lo svolgimento del progetto. Per questo lavoro ho sviluppato uno schema di Gantt (visibile nella pagina successiva) con un totale di 20 attività da svolgere per arrivare alla conclusione del progetto.

Queste attività le ho raggruppate nelle seguenti macrocategorie:

1. **Analisi:** Contiene tutte le attività che riguardano l’analisi, lo studio e la pianificazione del progetto. Ho stimato che per questa parte avrei impiegato circa 14 ore.
2. **Progettazione:** Contiene tutte le attività che riguardano il design del progetto. Per questa fase del progetto ho stimato di investirci circa 8 ore.

* **Design:** sotto categoria che contiene la progettazione del design delle classi e della GUI

1. **Implementazione:** Contiene tutta la parte di sviluppo di tutte le funzionalità dell’applicazione WEB e l’implementazione di quest’ultime con la grafica fatta in precedenza. Per questa attività ho stimato di impiegarci 64 ore.
2. **Documentazione e Diario**: Questa attività dura dall’inizio del progetto fino alla fine poiché sono delle cose che bisogna portare avanti man mano che il progetto avanza.

Stile orizzontale



Figura 1 Gantt progetto

## Analisi dei mezzi

Elencare e descrivere i mezzi disponibili per la realizzazione del progetto. Ricordarsi di sempre descrivere nel dettaglio le versioni e il modello di riferimento.

### Software

* Javascript
* Visual Studio Code

### Hardware

Su quale piattaforma dovrà essere eseguito il prodotto? Che hardware particolare è coinvolto nel progetto? Che particolarità e limitazioni presenta? Che HW sarà disponibile durante lo sviluppo?

# 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.

Diagramma Chen

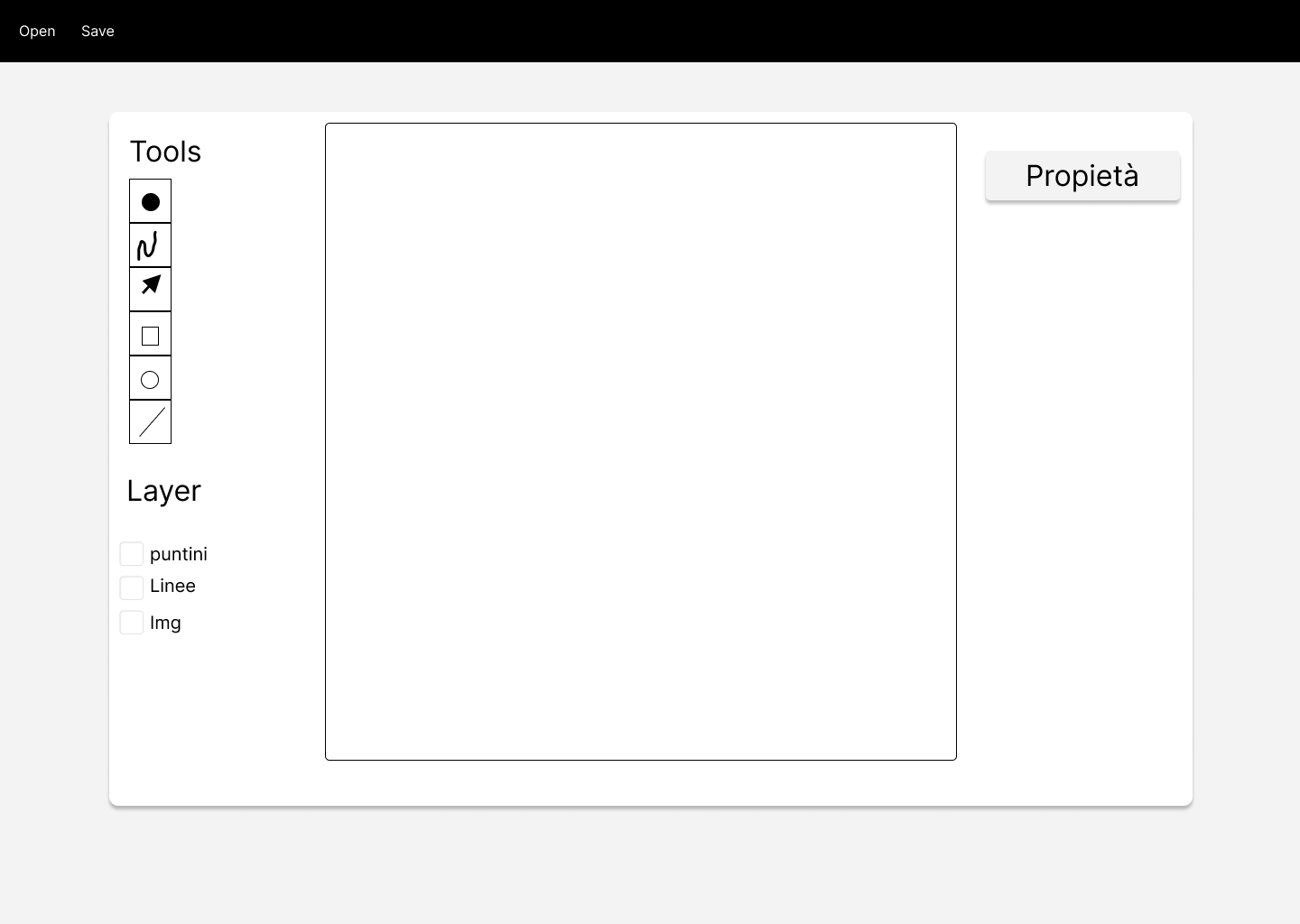
Figura Diagramma ER Chen

Diagramma Barker/Bachmann



Figura Diagramma ER Barker

## Design delle interfacce



Questa è l’interfaccia che troverà l’utente una volta aperto il sito, questa interfaccia è composta da 5 funzioni principali:

* Open: serve per far scegliere la foto che vuole inserire l’utente e fare l’upload di quest’ultima nel riquadro
* Riquadro: dove verrà visualizzata l’immagine scelta dall’utente
* Tools: è una barra degli strumenti, dove l’utente potrà scegliere lo strumento che gli serve ed in seguito utilizzarlo. La barra contiene i seguenti strumenti:
  + Puntino: per creare i puntini numerati sull’immagine
  + Penna: per creare delle line a mano libera
  + Mouse: per spostare gli elementi creati dove si vuole all’interno del canvas
  + Quadrato: per Creare dei quadrati
  + Ellissi: per creare delle ellissi
  + Retta: per creare delle line dritte
* Proprietà: mostrerà tutte le proprietà e funzionalità dello strumento scelto dall’utente, come: color, grandezza, elimina poligono, …
* Layer: Tramite questi checkbox l’utente potrà scegliere in tempo reale cosa vedere nel riquadro nascondendo il resto.

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.

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Risultato ottenuto** | **Stato** |
| TC-001 | Testo o immagine a conferma del test | Passato |
| TC-002 | Testo o immagine a conferma del test | Fallito |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 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).

Stile orizzontale



CONSUNTIVO

Figura Esempio di diagramma di 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:

## Bibliografia per libri

## Sitografia

* <https://www.geeksforgeeks.org/how-to-get-the-coordinates-of-a-mouse-click-on-a-canvas-element/>, 15-09-2023
* <https://dirask.com/posts/JavaScript-draw-point-on-canvas-element-PpOBLD>, 15-09-2023
* <https://medium.com/front-end-weekly/draw-an-image-in-canvas-using-javascript-%EF%B8%8F-2f75b7232c63>, 22-09-2023
* <https://medium.com/@zxlee618/drawing-on-a-html-canvas-b7566624b17f>, 06-10-2023
* <https://codepen.io/seangeleno/pen/BjGexE>, 06-10-2023

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.

# Glossario

|  |  |
| --- | --- |
| **Termine** | **Significato** |
| ASCII | American Standard Code for Information Interchange, codice per la codifica di caratteri. |
|  |  |
|  |  |
|  |  |
|  |  |
| ZFS | Zettabyte File System, filesystem opernsource a 128 bit con funzionalità avanzate. |

# Indice delle figure

[Figura 1 Esempio di diagramma di Gantt 7](#_Toc124775984)

[Figura 2 Esempio di diagramma di Gantt 8](#_Toc124775985)

[Figura 3 Diagramma ER Chen 10](#_Toc124775986)

[Figura 4 Diagramma ER Barker 11](#_Toc124775987)

[Figura 5 Esempio di diagramma di Gantt consuntivo 16](#_Toc124775988)

# 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
* …