Lavoro temporaneo

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 4

2 Analisi 5

2.1 Scopo 5

2.2 Requisiti 6

2.3 Verificare i requisiti 9

2.4 Risultati 10

2.5 Use case 11

2.6 Pianificazione 12

2.7 Analisi dei mezzi 12

2.7.1 Software 12

2.7.2 Hardware 12

3 Progettazione 12

3.1 Design dell’architettura del sistema 13

3.2 Design dei dati e database 13

3.3 Design delle interfacce 13

3.4 Design procedurale 13

4 Implementazione 14

5 Test 14

5.1 Protocollo di test 14

5.2 Risultati test 15

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 16

9 Allegati 16

# Introduzione

## Informazioni sul progetto

* Allievo:
  + Samuele Abbà
* Docente:
  + Guido Montalbetti
* Scuola:
  + SAMT Trevano
* Sezione:
  + Informatica
* Materia:
  + M306
* Inizio:
  + 16.09.2021
* Fine:
  + 23.12.2021

## Abstract

* **Situazione iniziale**
  + All’inizio di questo progetto, mi è stato chiesto di realizzare un sistema che possa gestire la domanda e l’offerta di posti di lavoro temporanei. Questo progetto nasce da un bisogno ben specifico. Fino ad ora, non esiste un prodotto o servizio che offre la possibilità di mettere in contatto due o più persone che hanno bisogno di un aiuto o che possono dare un aiuto. In breve: questo progetto è incentrato sull’elaborazione di un portale web che possa gestire comodamente le domande e le offerte per un lavoro temporaneo.
* **Attuazione**
  + Per questo progetto, saranno necessarie buone conoscenze nella programmazione web, poiché complesso nella sua struttura e nelle sue funzionalità. Seguendo i requisiti dati, per questo progetto è stato quindi necessario adoperare una banca dati per immagazzinare delle informazioni, compresa l’implementazione di tutta la sua sicurezza. Sono stati implementati una serie di sistemi che permettano l’aggiunta, la modifica e la rimozione delle varie offerte di lavoro all’interno della banca dati. Questo permettendo agli utenti presenti nel sistema di essere il più comodi, veloci e sicuri nell’utilizzo del prodotto. Vi sarà inoltre la possibilità di caricare dei documenti, quali p.es. Curriculum Vitae, Diplomi, ad ogni richiesta e una volta completata la candidatura, i clienti riceveranno tramite eMail la conferma della loro iscrizione al posto in offerta. L’offerta scadendo permette di fa calcolare al sistema i soldi che il datore di lavoro deve pagare al lavoratore e all’agenzia, inoltre ci saranno degli amministratori che potranno eseguire delle stampe, con inserimento del periodo desiderato, per gli stipendi da consegnare ai collaboratori e la fatturazione al datore di lavoro da parte dell’Agenzia.
* **Risultati**
  + Questo sito è realizzato da zero con i principali linguaggi web (HTML, CSS, PHP). Ho quindi avuto possibilità di lavorare su argomenti già consolidati e argomenti non ancora consolidati, questo mi ha permesso di ampliare il mio bagaglio delle conoscenze durante il progetto. Uno dei punti fondamentali tenuti in considerazione durante il progetto riguarda lo studio per raggiungere la facilità di utilizzo per un utente medio. Questo portale web è difatti utilizzabile e configurabile anche da un’utenza che non possiede conoscenze informatiche. In conclusione, questo progetto è ora l’unico sito funzionante che permetta di dare o prendere lavori temporanei.

## Scopo

* Vedi QdC in allegato

# Analisi

## Scopo

Spesso per poter trovare un lavoro 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 trovare un lavoro in pochi minuti.

Per poter accedere al mio sito sarà necessario come minimo un dispositivo con accesso ad Internet e un’email e password nel caso in qui si voglia creare o accettare lavori, una conoscenza basilare per poter usare un sito Web. In questo capitolo sono andato a suddividere i vari requisiti che ho trovato nel QdC, con i rispettivi sotto requisiti.

Come primo requisito ho messo la verifica del sito web, che funzioni correttamente e senza essere loggati, questo ti permette di vedere le varie offerte di lavoro.

Come secondo requisito ho messo la verifica del form di registrazione e anche la verifica del form di accesso.

Dopodiché come terzo requisito ho messo la verifica del sito web per i clienti/datori di lavoro, nell’ aggiungere, rimuovere o modificare, e accettare offerte di lavoro.

Poi come quarto requisito ho messo la fatturazione per i datori di lavoro, e i loro lavoratori e per i collaboratori.

Successivamente i requisiti per le stampe; per gli stipendi dei collaboratori e per le fatture dei datori di lavoro

da parte degli amministratori (con diritti di amministratore).

I prossimi due requisiti concernono la gestione delle richieste/offerte dei clienti registrati, è importante definire gli attributi delle offerte e delle richieste di lavoro, come anche la conferma tramite e-mail (senza diritti di amministratore).

Come ottavo requisito vi sarà inoltre la possibilità di caricare dei documenti, quali p.es. Curriculum Vitae, Diplomi, ad ogni offerta e una volta completata la candidatura, i clienti riceveranno tramite eMail la conferma della loro iscrizione al posto in offerta.

10 req

## Requisiti

|  |  |
| --- | --- |
|  | **ID: REQ-001** |
| **Nome** | Verifica del sito web senza login |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare che il sito web funzioni correttamente e senza essere loggati visualizzare i lavori disponibili. |

|  |  |  |
| --- | --- | --- |
|  | | **ID: REQ-002** |
| **Nome** | | Verifica del sito web con login |
| **Priorità** | | 1 |
| **Versione** | | 1.0 |
| **Note** | | È necessario verificare che il sito web funzioni correttamente. |
| **Sotto Requisiti** | | |
| **0021** | Verifica del funzionamento di: form di registrazione. | |
| **0022** | Verifica del funzionamento di: form di accesso, conferma tramite email. | |

|  |  |  |
| --- | --- | --- |
|  | | **ID: REQ-003** |
| **Nome** | | Verifica login |
| **Priorità** | | 1 |
| **Versione** | | 1.0 |
| **Note** | | È necessario verificare il corretto funzionamento del login |
| **Sotto Requisiti** | | |
| **0031** | Si possano creare richieste. | |
| **0032** | Si possano creare offerte. | |

|  |  |  |
| --- | --- | --- |
|  | | **ID: REQ-004** |
| **Nome** | | Verifica fatturazione |
| **Priorità** | | 1 |
| **Versione** | | 1.0 |
| **Note** | | È necessario verificare che le fatture funzionino correttamente. |
| **Sotto Requisiti** | | |
| **0041** | Verifica fine del lavoro. | |
| **0042** | Calcolo spese del datore di lavoro. | |
| **0043** | Calcolo stipendio collaboratori. | |

|  |  |  |
| --- | --- | --- |
|  | | **ID: REQ-005** |
| **Nome** | | Verifica per gli amministratori |
| **Priorità** | | 1 |
| **Versione** | | 1.0 |
| **Note** | | È necessario verificare gli amministratori possano gestire e controllare il sistema. |
| **Sotto Requisiti** | | |
| **0051** | Verifica login amministratori. | |
| **0052** | Verifica pagina amministratori. | |
| **0053** | Verifica fatturazione. | |
| **0054** | Verifica gestione. | |

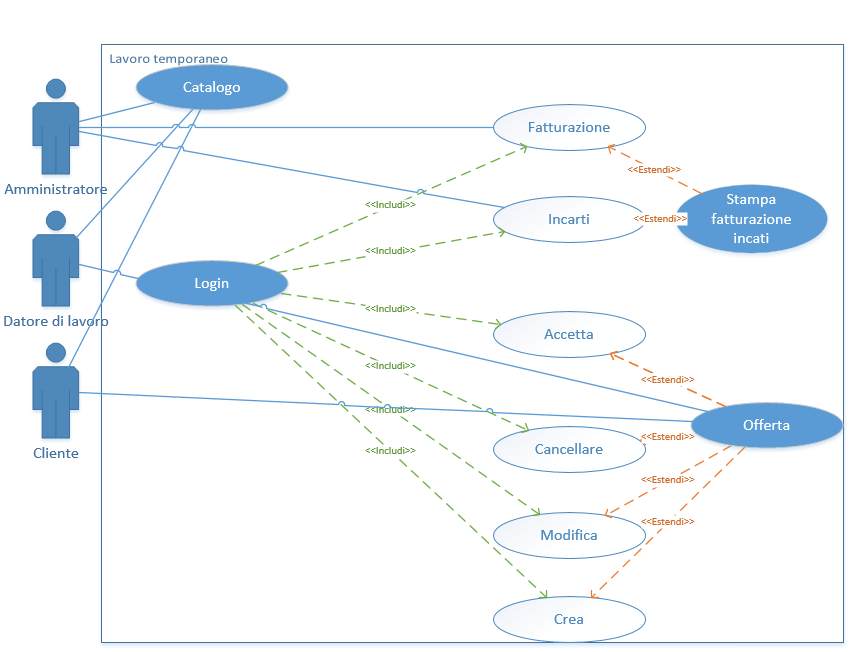
## Verificare i requisiti

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-001 | **Nome:** | - |
| **Descrizione:** | - | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | - | | |
| **Risultati attesi:** | - | | |

## Risultati

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Funzionamento | Commento | Data |
| TC-XXX | OK | - | - |
| TC-XXX | NOK | - | - |

## Use case



All’interno dell’applicativo ho inserito 3 tipi diversi di utente, più specificatamente:

* Amministratore: Gli utenti appartenenti a questo gruppo sono presenti nel database l’unico compito che avranno sarà quello di preparare le fatture, gli incarti e gestire il sistema. Per fare ciò gli amministratori dovranno loggarsi e la loro aggiunta avverrà nel database alla sua creazione.
* Datore di lavoro e cliente: essi dovranno obbligatoriamente eseguire l’accesso tramite un account per interagire con il sito. Essi potranno se datori di lavoro creare, modificare e cancellare le offerte oppure, se clienti accettare le varie richieste.
* In ogni caso se non si è loggati si può comunque guardare il catalogo.

## 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

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

SDK, librerie, tools utilizzati per la realizzazione del progetto e eventuali dipendenze.

### 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.

## 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 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
* …