Inventario HW

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

Situazione iniziale 3

Attuazione 3

Risultati 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 8

2.4 Pianificazione 9

2.5 Analisi dei mezzi 10

2.5.1 Software 10

2.5.2 Hardware 10

3 Progettazione 10

3.1 Design dell’architettura del sistema 10

3.2 Design dei dati e database 10

3.3 Design delle interfacce 10

3.4 Design procedurale 10

4 Implementazione 12

5 Test 12

5.1 Protocollo di test 12

5.2 Risultati test 13

5.3 Mancanze/limitazioni conosciute 13

6 Consuntivo 13

7 Conclusioni 13

7.1 Sviluppi futuri 13

7.2 Considerazioni personali 13

8 Bibliografia 13

8.1 Bibliografia per articoli di riviste: 13

8.2 Bibliografia per libri 13

8.3 Sitografia 14

9 Allegati 14

# Introduzione

## Informazioni sul progetto

* Allievo: Dennis Donofrio
* Docente: Guido Montalbetti
* Azienda: SAM Trevano
* Classe: I3BC
* Materia: M306 / Laboratorio progetti
* Periodo: 16.09.2021 – 23.12.2021

## Abstract

## Situazione iniziale

All’inizio di questo progetto mi è stato chiesto di realizzare un sistema che possa gestire i vari computer e periferiche. Al momento tutti questi oggetti sono catalogati in un file Excel. Ogni volta per prendere o portare un oggetto bisogna chiedere all’amministratore di aggiungere o togliere questo componente. Questo progetto permette di sapere quanti oggetti ci sono in totale, quanti oggetti sono liberi, quanti oggetti sono occupati e in quale aula sono usati attualmente. Un utente con i permessi di livello amministratore può creare degli oggetti, rimuoverli, riservarli, creare nuovi utenti e rimuovere degli utenti.

L’utente base può semplicemente riservare degli oggetti e liberarli.

## Attuazione

Soluzioni e processi applicati nella risoluzione del problema, …

## Risultati

Breve analisi dei risultati ottenuti (obiettivi raggiunti, …), …

## Scopo

Lo scopo di questo progetto è quello di semplificare la riservazione di materiale informatico. In questo modo si possono riservare degli oggetti e vedere quali oggetti sono ancora disponibili, quanti e altre informazioni.

# Analisi

## Analisi del dominio

All’inizio di questo progetto mi è stato chiesto di realizzare un sistema che possa gestire i vari computer e periferiche. Al momento tutti questi oggetti sono catalogati in un file Excel. Ogni volta per prendere o portare un oggetto bisogna chiedere all’amministratore di aggiungere o togliere questo componente. Questo progetto permette di sapere quanti oggetti ci sono in totale, quanti oggetti sono liberi, quanti oggetti sono occupati e in quale aula sono usati attualmente. Un utente con i permessi di livello amministratore può creare degli oggetti, rimuoverli, riservarli, creare nuovi utenti e rimuovere degli utenti.

L’utente base può semplicemente riservare degli oggetti e liberarli.

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Ambiente funzionante |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Apache, PHP, MySQL funzionanti |
| **Sotto requisiti** | |
| **001** | Apache con la porta 80 |
| **002** | MySql con la porta 3306 |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Struttura database |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessita di tabelle riservate agli utenti e tabelle riservate ai componenti HW |
| **Sotto requisiti** | |
| **001** | Tabella utente |
| **002** | Tabella oggetto |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| **Nome** | Creazione login |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Una pagina unica che faccia accedere gli utenti. |
| **Sotto requisiti** | |
| **001** | Si necessita una maschera di login |
| **002** | Si necessita di un tasto per cambiare password in caso di smarrimento |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Distinzione utenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Amministratori e utenti base avranno permessi diversi. |
| **Sotto requisiti** | |
| **001** | Si necessita di utenti base ed amministratori |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | Creazione varie pagine web |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessitano 4 tipi di pagine. |
| **Sotto requisiti** | |
| **001** | Si necessita una pagina per la home degli utenti |
| **002** | Si necessita una pagina per la home degli amministratori |
| **003** | Si necessita una pagina per vedere tutto l’inventario |
| **004** | Si necessita una pagina per ogni componente HW |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Creazione di filtri |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Si necessitano di filtri per migliorare l’efficacia della ricerca |
| **Sotto requisiti** | |
| **001** | Filtri sulla descrizione e sulla tipologia |

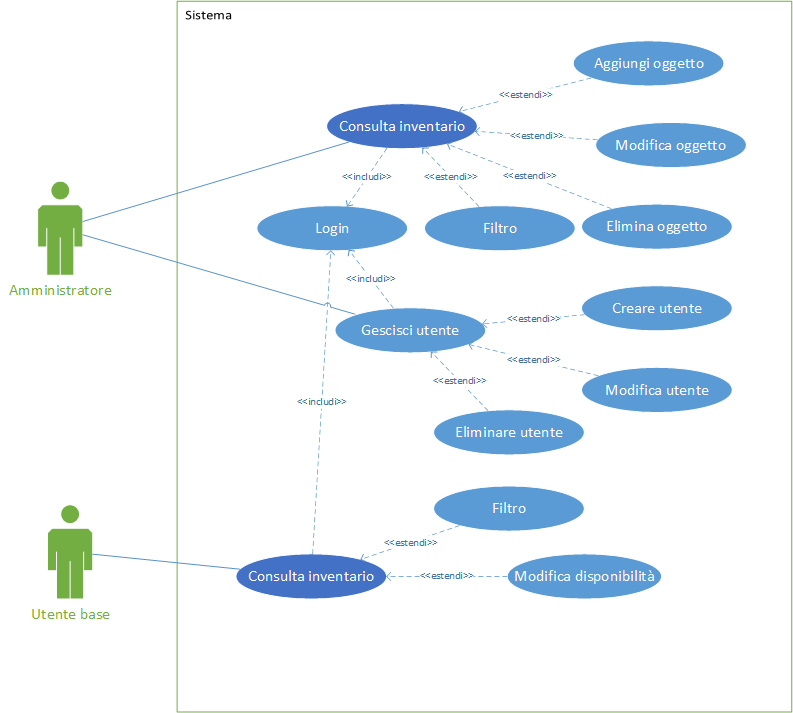
|  |  |
| --- | --- |
| **ID: REQ-007** | |
| **Nome** | Percentuale di soglia |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Percentuale sotto la quale verrà inviata una email per comunicare che la disponibilità quella tipologia è sceso sotto la percentuale |
| **Sotto requisiti** | |
| **001** | Si necessita la possibilità di parametrizzare la percentuale di attenzione |
| **002** | Se la percentuale è stata superata, viene inviata una mail notificando che una tipologia sta finendo |

|  |  |
| --- | --- |
| **ID: REQ-008** | |
| **Nome** | Funzioni amministratori |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Gli amministratori hanno il controllo completo dell’applicativo. |
| **Sotto requisiti** | |
| **001** | Si necessita la modifica da parte degli amministratori |
| **002** | Gli amministratori potranno aggiungere o rimuovere i componenti HW |

|  |  |
| --- | --- |
| **ID: REQ-009** | |
| **Nome** | Gestione utenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Gli amministratori hanno la possibilità di aggiungere, modificare o rimuovere un utente. |
| **Sotto requisiti** | |
| **001** | Si necessita un collegamento alla pagina di gestione degli utenti |
| **002** | Si necessitano 3 pagine per le varie operazioni sugli utenti |

|  |  |
| --- | --- |
| **ID: REQ-010** | |
| **Nome** | Modifica inventario |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | Gli amministratori hanno la possibilità di aggiungere, modificare o rimuovere un oggetto dall’inventario. |
| **Sotto requisiti** | |
| **001** | Si necessita un collegamento alla pagina di gestione dell’inventario |
| **002** | Si necessitano due pagine per le varie operazioni sull’inventario |

## Use case



## Pianificazione

|  |
| --- |
|  |

## 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

* Sublime text
* Apache2.4
* Mysql
* PHP7
* VirtualBox
* HTML5
* CSS3

### Hardware

Il server è installato su una macchina virtuale virtualbox con un processore con 4 core, 8gb di ram e un hdd da 50gb.

# 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.

|  |
| --- |
|  |

La struttura del database è composta da 4 tabelle. Queste tabelle sono necessarie per il corretto funzionamento dell’applicativo. Di seguito ci sono le descrizioni complete di tutte le tabelle.

* **user:** Questa tabella contiene molti attributi. L’id permette di distinguere gli utenti e controllare se un utente ha effettuato il login nel sito. Is\_admin permette di sapere se un utente ha privilegi di livello amministratore o utente base. Name e surname sono essenziali per l’autenticazione perché quando si registra un utente non si chiede di inserire il nome utente, ma viene creato automaticamente. Il nome utente viene creato concatenando il nome ed il cognome, separati dal carattere ‘.’. L’attributo email potrebbe essere utile per un futuro sviluppo. Può essere utilizzata per mandare delle mail di conferma nella registrazione, per la modifica, per la password persa o per altre comunicazioni importanti. L’ultimo attributo contiene l’hash della password. Questo è utile per la sicurezza degli utenti e per la sicurezza dell’accesso.
* **classroom:** Questa tabella contiene solo due attributi. L’attributo number definisce il numero dell’aula. L’attributo description contiene una descrizione dell’aula, per esempio “Aula disegnatori”.
* **type:** Questa tabella contiene solo 3 attributi. L’id permette di distinguere i tipi a livello di database, in modo da non dover lavorare con le stringhe. L’attributo id è univoco. L’attributo name contiene il nome del tipo, per esempio “proiettore”. L’attributo description contiene una breve descrizione del tipo, per esempio “Proiettore e retroproiettore”.
* **object:** Questa tabella è la più delicata perché contiene 3 chiavi esterne. In totale ha 7 attributi. L’id permette di distinguere i vari oggetti a livello di database. L’id è univoco. L’attributo description contiene una descrizione dell’oggetto nell’inventario. L’attributo serial\_number contiene il numero seriale dell’oggetto. L’attributo riservation\_date contiene la data di riservazione di un oggetto, vuol dire la data in cui un oggetto è stato spostato dal magazzino ed inserito in un’aula.

## 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.

|  |
| --- |
|  |

Pagina di login

Questa pagina può essere visualizzata da tutti. È presente un textbox per l’inserimento dello username e un campo password per l’inserimento della password.

## 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.

**Per adesso sono nell’altro file**

|  |  |  |  |
| --- | --- | --- | --- |
| **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
* …