

Dipartimento di Ingegneria e Scienze dell’Informazione e Matematica

CORSO DI LAUREA IN INFORMATICA

Web Engineering

NOME PROGETTO

|  |  |  |  |
| --- | --- | --- | --- |
| **Membri del team** | | | |
| **Cognome e nome** | **Matricola** | | **Indirizzo e-mail** |
| Emanuele Pizzoli | 272424 | | emanuele.pizzoli@student.univaq.it |
| Falleroni Francesco | 261164 | | francesco.falleroni@student.univaq.it |
| Paolo Celani | 272163 | | paolo.celani@student.univaq.it |
|  | |  | | |
|  | | | | |
|  | |  | | |

A.A. 2021/2022

Sommario

[Capitolo 1 – Dipendenze Software 3](#_Toc156840447)

[1.1 – Librerie usate 3](#_Toc156840448)

[1.1.1– Freemarker 3](#_Toc156840449)

[1.1.2– MySQL-Connector 3](#_Toc156840450)

[1.1.3– Javaee-Web 4](#_Toc156840451)

[1.2 – Server 4](#_Toc156840452)

[1.3 – Client 4](#_Toc156840453)

[Capitolo 2 – Funzionalità 4](#_Toc156840454)

[2.1 – Realizzate 4](#_Toc156840455)

[2.2 – Non Realizzate 4](#_Toc156840456)

[Capitolo 3 – Diagrammi 5](#_Toc156840457)

[3.1 – Struttura sito 5](#_Toc156840458)

[3.2 – Diagramma Entità-Relazione 6](#_Toc156840459)

[Capitolo 4 – Tecnologie 7](#_Toc156840460)

[4.1 – Linguaggi 7](#_Toc156840461)

[4.2– Librerie 7](#_Toc156840462)

[Capitolo 6 – Contributo 8](#_Toc156840463)

[6.1 – Tabella 8](#_Toc156840464)

# Capitolo 1 – Dipendenze Software

## – Librerie usate

Le librerie usate per la realizzazione di questa web application sono:

1. Freemarker (2.3.30)
2. MySQL-connector-j (8.0.32)
3. Javaee-web (7.0)

La loro implementazione è effettuata all’interno del file di configurazione [pom.xml](Web-Engineering/AuleWeb_Back-UP/pom.xml).

## 1.1.1– Freemarker

* Template engine per la generazione il template delle pagine Web
* Utilizzo di variabili per generare dinamicamente l’output delle pagine
* Sintassi in grado di poter effettuare flussi di controllo (cicli for, if, …), assegnazioni, iterazioni, ecc.…

|  |
| --- |
| <#listListaDipartimenti as dipartimento>  [${dipartimento.nome}](about:blankhome?k=$%7bdipartimento.key%7d)  </#list> |
|  |

Il codice usto nella pagine [home.ftl](Web-Engineering/AuleWeb_Back-UP/src/main/webapp/templates/home.ftl.html) permette una visualizzazione di tutti i dipartimenti e poli contenuti nel nostro database MySQL.

## 1.1.2– MySQL-Connector

* Connessione al database MySQL
* Accesso ai dati dei dati del database MySQL

La connessione viene effettuata nel file [context.xml](Web-Engineering/AuleWeb_Back-UP/src/main/webapp/META-INF/context.xml) in cui sono specificati:

* Name => nome del db;
* Username => username dell’utente MySQL che può accedere al db;
* Password => password dell’utente MySQL che può accedere al db;
* url => link url per la connessione al db in cui viene specificato indirizzo IP e porta di connessione;

## 1.1.3– Javaee-Web

* Uso di Servlet per le richieste HTTP dinamiche (navigazione dinamica per il contenuto delle pagine)

## 1.2 – Server

Come server abbiamo usato un database MySQL generato per la gestione e visualizzazione di eventi all’interno dell’università.

## 1.3 – Client

La nostra applicazione è stata testa sui vari browser

|  |  |
| --- | --- |
| Browser | Versioni |
| FireFox | 121.0.1 – 120.0 |
| Opera | 106.0.4998.49 - 90.0.4480.84 |
| Google Chrome | 120.0.6099.225-103.0.5060.53 |
| Internet Explorer/ Microsoft Edge | 120.0.2210.144 |

(versioni più vecchi di queste testa potrebbero presentare errori di formattazione html).

# Capitolo 2 – Funzionalità

## 2.1 – Realizzate

La nostra applicazione permette di:

1. Effettuare un login se si hanno le credenziali
2. Visualizzare le aule di un **Dipartimento** e/o **Polo**
3. Generare e modificare un **Aula** associate ad un **Evento**
4. Generare e create un **Evento**
5. Generare e modificare un **Calendario**
6. Visualizzare le informazioni relative ad un **Evento**
7. Fare un “filtraggio” sulle aule in modo da poter visualizzare gli eventi:
   1. Di un’aula specifica durante la settimana
   2. Di un’aula in giorno specifico
   3. Nelle prossime 3 ore
8. Possibilità di esportare la configurazione degli eventi correnti in formato CSV

## 2.2 – Non Realizzate

* Non è stato implementato la possibilità di importare il contenuto di una configurazione CSV;
* Non è stato realizzato in maniera ottimale la selezione dell’attrezzatura presente in un aula e la selezione del dipartimento associato ad un aula;
* Evitare le collisioni per la selezione delle aule gli eventi già assegnati, per evitare le collisioni si intende selezionare un aula/eventi già assegnati (sono presenti i metodi per gestire le aule già assegnate ma per aggiungere la logica per gli eventi associati all’aula era più complesso a livello di implementazione);

# Capitolo 3 – Diagrammi

## 3.1 – Struttura sito

Descrizione dettagliata ed illustrata di cosa fa l’applicazione, ovvero diagramma sulle pagine dell’applicazione (navigazioni model)

Pagine:

1. Avvio 🡺l’utente può selezionare un dipartimento di cui vedere gli eventi presenti nelle aule assegnate al dipartimento
2. Home 🡺dopo aver selezionato il dipartimento l’utente trova nella sua home l’insieme di tutti i corsi assegnati al dipartimento selezionato
3. Aule 🡺 dopo aver selezionato il corso si accederà a una “seconda home” in cui sono presenti le aule associate al corso del dipartimento selezionato e che vengono usate durante la settimana. In questa pagina l’utente può effettuare dei **filtraggi** per sapere quali eventi:
   1. Settimanali (può visualizzare tutte le attività assegnate anche dai diversi dipartimenti che usano quell’aula)
   2. Nelle prossime 3 ore (sono gli eventi del dipartimento che si possono svolgere in qualsiasi aula a sua disposizione)
   3. giornaliero (è come il filtraggio settimanale ma solo nell’arco di una giornata universitaria)

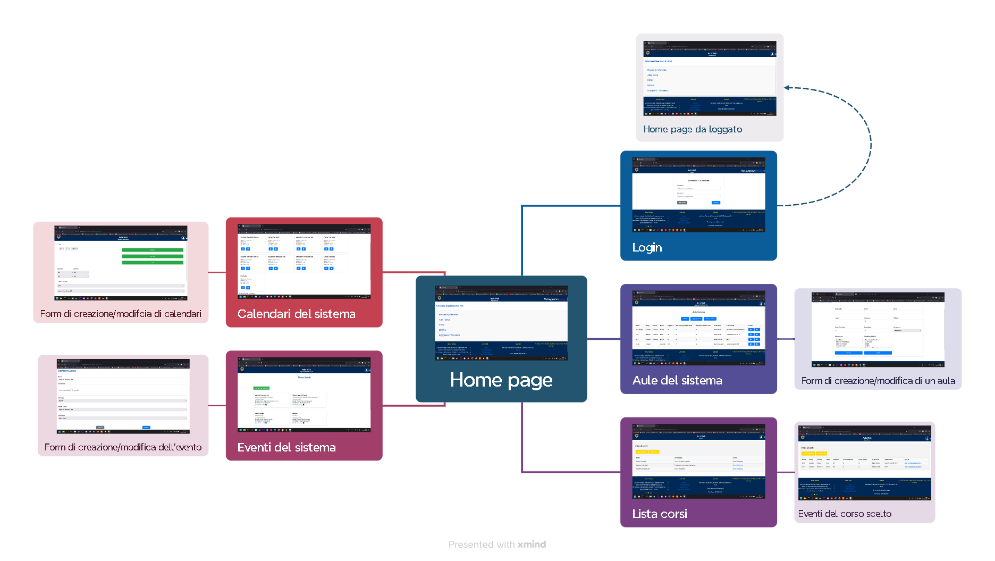
(nei casi di settimanale e giornaliero è possibile selezionare il giorno/la settimana su cui si vuole avere le informazioni)

1. login 🡺 a questa pagina è possibile accedere in qualsiasi momento tramite il link nell’header che porterà a un form in cui, se si inseriscono i dati personali di un **Professore**, si accederà alle pagine successive
2. Aula 🡺 vista di tutte le aule presenti nel sistema
3. Eventi 🡺 vista di tutte gli eventi associate alle aule presenti
4. Calendario 🡺 le informazioni degli eventi (giorno e ora)

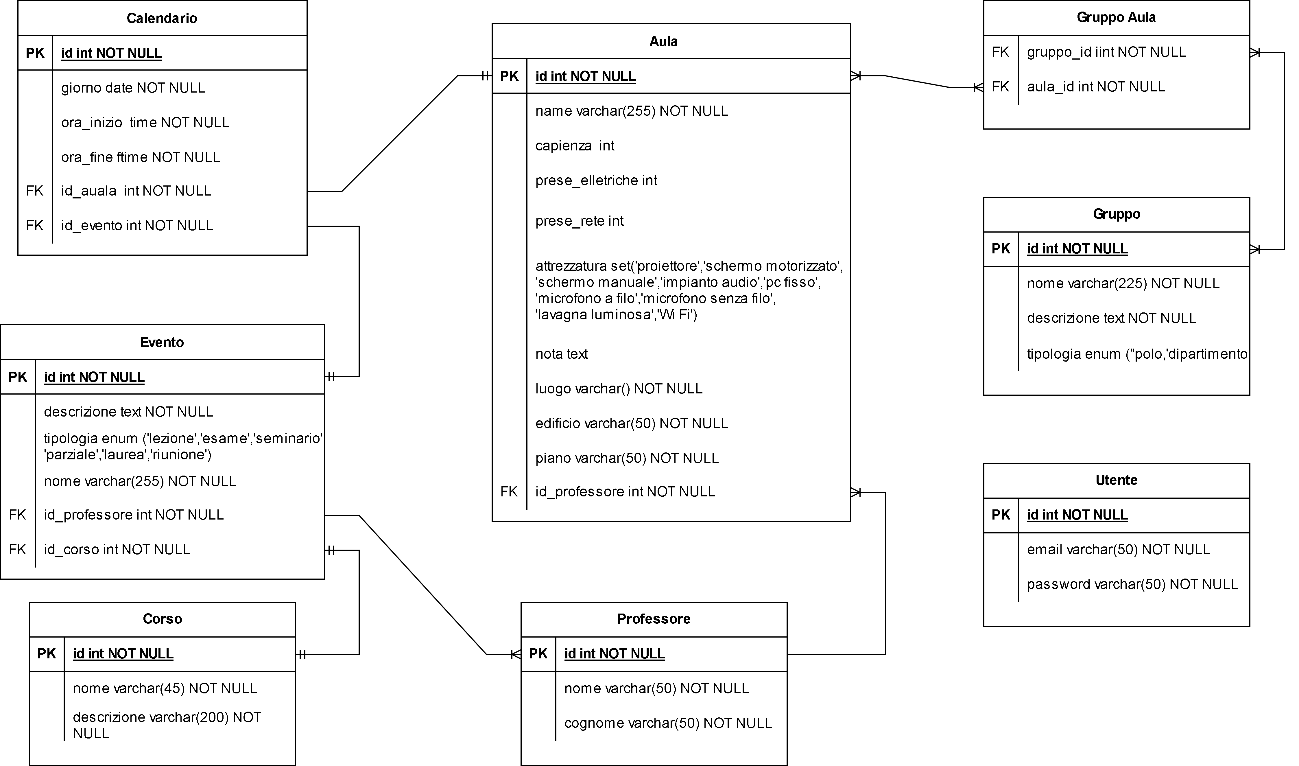
Nelle ultime 3 pagine il **Professore** ha anche la possibilità di:

* Creare
* Modificare
* Cancellare

i relativi oggetti di tipo **Aule**, **Evento** e **Calendario.**

****

## 3.2 – Diagramma Entità-Relazione



* Aula 🡺 luogo in cui si svolgerà un determinato **Evento**, specificandone:
  + Nome dell’aula
  + Numero di prese elettriche e di rete
  + Luogo (zona in cui è situata l’aula)
  + Edificio (quale edificio contiene l’**Aula**)
  + Piano (piano dell’edificio)
  + Attrezzatura (oggetti presenti all’interno dell’**Aula)**
  + Nota (problemi/avvertenze)
  + **Professore** (docente/assistente che presiederà l’**Evento**)
* Evento 🡺attività generica che si svolge giornalmente composta da:
  + Descrizione
  + Tipo (Lezione, Riunione, Parziale, Esame, Seminario, Laurea)
  + **Corso** (riferimento al nome del **Corso** che verrà si svolgerà in un’**Aula**)
  + **Professore** (docente/assistente che presiederà l’**Evento**)
* Calendario 🡺indica la ricettività di un **Evento** in una **Aula**, specificandone giorno ed ora
* Gruppo 🡺 Dipartimenti e Poli che compongo l’Università
* Gruppo Aula 🡺 collegamento tra **Gruppo** ed **Aula** dato che un’aula può essere usata da più dipartimenti
* Corso 🡺 Corso di studi di un **Gruppo**
* Professore 🡺 utente che può effettuare il login ed è in grado di creare e/o modificare il contenuto di un **Evento**, un’**Aula** e un **Calendario**
* Utente 🡺 utente generico che può solo visualizzare le informazioni relative alle **Aule** ed **Eventi** che si svolgono in esse (non può effettuare il login)

# Capitolo 4 – Tecnologie

## 4.1 – Linguaggi

I linguaggi principalmente usati per realizzare il progetto sono:

* Java:  
  usato per la realizzazione della struttura dati (rappresentazione ad oggetti delle varie tabelle in MySQL), scrittura delle operazioni di create, update, remove e filtering (si tratta della “traduzione” in java delle query applicata alle tabelle) e creazione delle pagine html tramite le java Servlet
* Java-script:  
  usato per la generazione dinamica di liste di oggetti come:
  + Fasce orarie dei menù per la selezione dell’ora e dei minuti riferite ad un calendario
  + Scrittura di un file Excel CSV per la composizione delle aule

## 4.2– Librerie

Uso delle librerie java:

* Calendar 🡺 implementazione di un calendario con la possibilità di selezionare giorno, mese e anno;
* LocalTime 🡺 riferimenti temporali per la fascia oraria di un **Calendario**, ovvero per indicare l’ora di inizio e fino e il relativo minutaggio;
* Date 🡺 usata per creare il riferimento giornaliero di **Calendario**, ovvero per creare e gestire giorno, mese ed anno riferiti ad un **Calendario**;
* Commons CSV 🡺 libreria usata per l’esportazione del contenuto di una tabella HTML per creare un file CSV, ovvero per compilare un file Excel (nella nostra applicazione è stato fatto per scaricare le disposizioni delle aule presenti nel db);

# Capitolo 6 – Contributo

Plugin usati per la realizzazione/implementazione di funzionalità

## 6.1 – Tabella

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Data Base | Data Structure | Java Servlet | HTML & CSS |
| Francesco Falleroni |  | x | x |  |
| Emanuele Pizzoli | x |  | x |  |
| Paolo Celani |  |  | x | x |