Lerner’s Academy

Administration Project

Developed By: Yousif Al-Dhfeery

GitHub: <https://github.com/xxmelar-ksaxx>

Project Repository: [Lerner’s Academy](https://github.com/xxmelar-ksaxx/java_EE_STP/tree/main/Learners_Academy/src/main)

**Important Note:** To run the website properly, you have to initialize the database tables. To do that, you should navigate to **Starter** package, then open the **Database\_Initialiser** Class and Run it, by doing this, the database tables will be created automatically in MySql.

One more thing, in MySql, you should create a schema before running the Database\_Initialiser class, by the name ( hibernate ), and create a new user with privilege access by the name(jroot) and password (1234).

If you don’t want to do all of this, you can go the **Hibernate\_2** class in the **Control** package and modify the schema and user info to your preference.

Contents

[1- Sprint planning 3](#_Toc77342593)

[1.0 - Sprint goal 3](#_Toc77342594)

[1.1 - Planning meeting 3](#_Toc77342595)

[1.2 - Planning meeting II 3](#_Toc77342596)

[2- Packages, Classes and Algorithms 4](#_Toc77342597)

[2.0 Control Package 4](#_Toc77342598)

[2.1 DAL Package 4](#_Toc77342599)

[2.2 Entities Package 4](#_Toc77342600)

[2.3 Servlets Package 4](#_Toc77342601)

[2.4 Starter Package 5](#_Toc77342602)

[3- Flow chart 5](#_Toc77342603)

[4- Web Pages 6](#_Toc77342604)

[5- Core Concepts Used 6](#_Toc77342605)

[Conclusion 7](#_Toc77342606)

# Sprint planning

## 1.0 - Sprint goal

The goal is to create a website that manages school system. The main functionality is to manage the subjects, classes and teachers. An Administrator manages those functionalities, he is the only user to this website. The Administrator can do several things:

- Add new Subjects

- Add new classes

- Add new teachers

- Assign subject to a class

- Assign Classes to the teachers

- Show the list of Students

- Show class information

## 1.1 - Planning meeting

The meting will discuss the following points:

* Define the project goal
* List project main requirements
* List project features
* Discover initial sprint backlog

## 1.2 - Planning meeting II

* Discover detailed sprint backlog
* Estimating the stores
* List task to do
* Agree of the sprint plan

# Packages, Classes and Algorithms

There are five packages that provide specific services. The functionality, purpose and contents for each of them is discussed in the following sub-sections.

## 2.0 Control Package

The **Control package** is concerned with securing connection with the database. This package has one class called **Hibernate\_2**.

## 2.1 DAL Package

The **DAL** (Data Access Layer) **package**, this is the database manipulation place where all of the CURD operations happening here. There are six Classes in it, all of them have the same algorithms, the only differences between this is the query retrieving statement, otherwise they have the same structure.

The List of classes is:

* Classes\_DAL
* login\_DAL
* Student\_DAL
* Subject\_DAL
* TC\_DAL
* Teachers\_DAL

## 2.2 Entities Package

The **Entities Packages** is obvious, it stores the hibernate entities classes. There are six classes, one for each entity.

The list of entities is:

* Adminstorator
* Classes
* Students
* Subjects
* Teachers\_Classes
* Teachers

## 2.3 Servlets Package

The **Servlets package** is concerned of handling the jsp web pages requests. All of the servlets have the same algorithm’s structure. There are six classes, one for each entity.

The list of servlets is:

* ClassesServlet
* createSubjectListner
* loginServlet
* StudentServlet
* TCServlet
* TeachersServlet

## 2.4 Starter Package

The **Starter package** has one main function, to create and initialize the database tables. There is one class, **Database\_Initialiser**. This class has to start before the tomcat server starts for the first time. You only start it once.

# Flow chart

The website flow:



# Web Pages

In this project I have used JSP HTML pages. And Web.xml file to map the JSP actions to the servlets classes.

The list of files:

* Index.jsp
  + The login page
* MainPage.jsp
  + Main Menu page
* Subjects.jsp
  + Subjects page
* Classes.jsp
  + Classes page
* Teachers
  + Teaachers page
* TC.jsp
  + Teachers classes page
* Students
  + Students page
* Class\_info
  + Classes information page

# Core Concepts Used

- Hibernate. The hibernate concept is used here the create the database entities, and make the relations between the entities. Also is the gate to connect with the database.

- Java Server Pages (JSP). The JSP is used to integrate Java code into the HTML page. There are two main things used here, the for each concept and the if else statement. The for each statement is used to list items into the dropdown option tag. The if statement is used to determine wither to display the incorrect credentials warning in the Login page, or not.

- Java servlets. Used to handle the HTML/JSP actions, and provide the desired functionality.

- Tomcat Server. Used to run the Java Dynamic project website pages.

- Polymorphism, used in the list selectors. The same code structure is repeated in both of them (Main and File Manipulation menus), but slightly different results are returned.

# Conclusion

A school administration website that provides management features, is a grate way to test your abilities on handling data from the database, in a secure way by adding different layers of login credentials validation prosses. And dealing with HTML action requests by mapping the servlets classes through Web.xml file. In general, it’s a good project for learning hibernate, JSP and Servlet principals.