

Phase-2 Project

(Developing a Backend Admin Portal for Learner's Academy.)

Project Name: Learner's Academy Admin Portal.

Project Details:

Learner's Academy is a school that has an online management system. The system keeps track of its classes, subjects, students, and teachers. It has a back-office application with a single administrator login. The admin can create master list of classes, subjects, teachers, students, assign class to subjects and assign a teacher to a class with a subject. There is an option to view Class Report which will show all the information about the class, such as the list of students, subjects, and teachers.

Developer Details:

Name: Twarit Soni.

Simplilearn UserId: twarit.s88@gmail.com

Sprint Planning and task achieved:

- Number of Sprints Planned: 3

Sprint 1:

- The Objective of this sprint was to develop a admin login page with a servlet, Data access object classes, Data transfer object classes.
- Admin login page and dashboard successfully developed that takes username and password as input which is then validated by Login Servlet.
- The Data transfer and access object classes successfully developed.

- A new Database LearnersAcademy created using MySQL successfully.

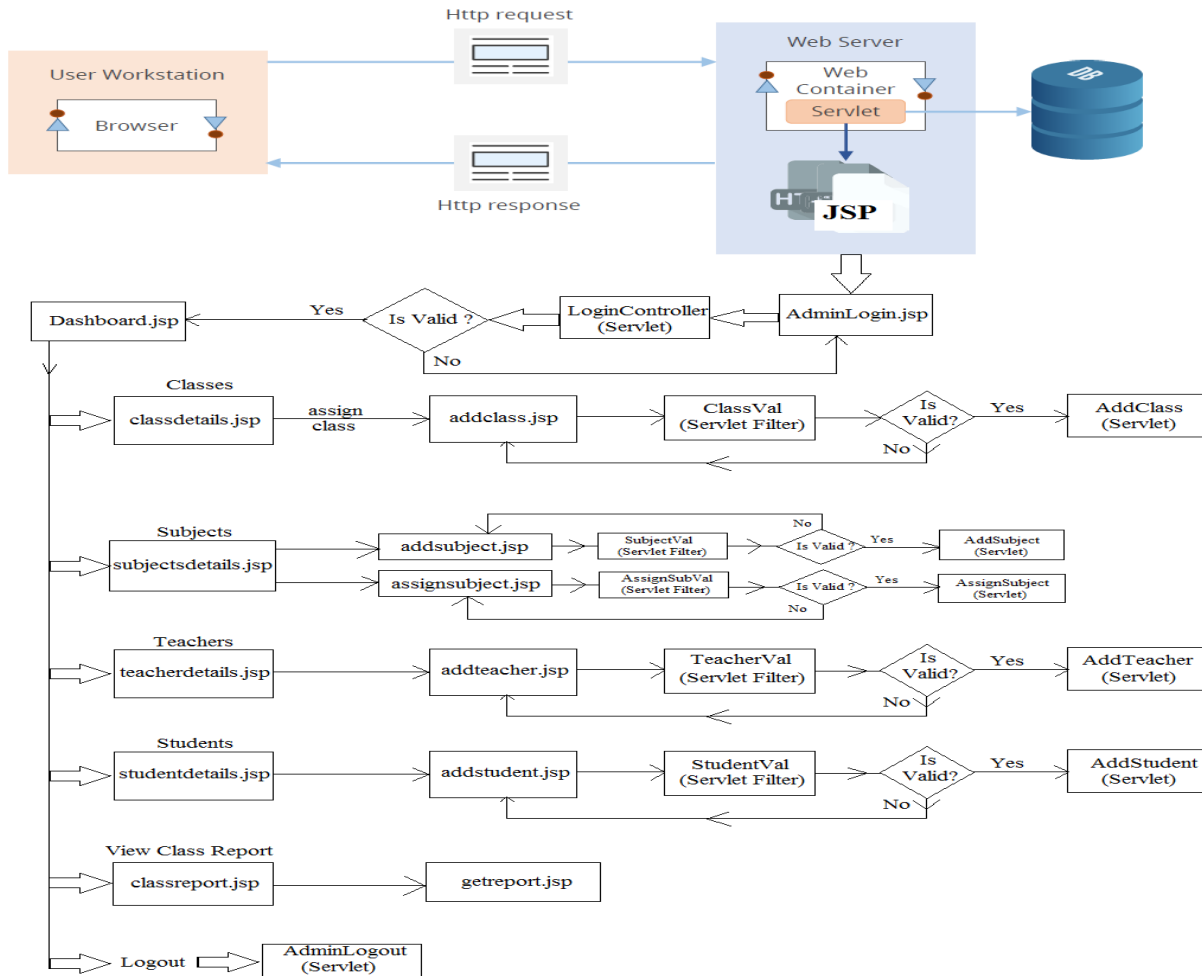
Sprint 2:

- The Objective of this sprint was to develop web pages, Servlets, Filters for the main operations like to setup master list of classes, teachers, students, subjects, assign classes and subjects.
- Web Pages, Servlets, Filters were successfully developed.
- Successfully configured web.xml and hibernate.cfg.xml files.

Sprint 3:

- The Objective of this sprint was to test the application by entering the Data and checking data from the database.
- The application is working as per the requirement given and ready for Release.

Flowchart and Algorithms:



- The application is launched by adding the dynamic web project: Learner's Academy in the web server. The 'adminlogin.jsp' page is welcome page where user have to enter the username and password.
- The username and password is validated by the 'LoginController.java' servlet. If credentials are valid then the Admin dashboard is displayed and if credentials are incorrect then 'Invalid Credentials' message is displayed included in the 'adminlogin.jsp' page.
- After successful validation of credentials the session is created and the 'dashboard.jsp' page is displayed. The Admin dashboard displays five options:

1. Classes
2. Subjects
3. Teachers
4. Students
5. View Class Report
6. Logout

- When Classes is clicked, 'classdetails.jsp' page is displayed where the class details are shown. The classes details is fetched from the table 'classes' in the database. Within the page there is an option to assign a teacher where admin can assign teacher to a class with subject. When Assign Teacher option is clicked the 'addclass.jsp' page is displayed where admin can fill class details in the form.
- The details entered in the form is validated by the 'ClassVal.java' filter. If invalid details are filled, a message 'Kindly fill correct details' will be displayed by including addclass.jsp page. After successful validation, the details are added in database in classes table through 'AddClass.java' servlet by calling the register method from 'ClassDao.java'(data access object class) and using 'Classes.java'(data transfer object class).
- When Subjects is clicked, 'subjectsdetails.jsp' page is displayed where the subjects details are shown. The subjects details are fetched from the table 'subjects' in the database. Within the page there are two options. First one is Add Subject. By clicking Add Subject 'addsubject.jsp' page is displayed. By filling the details and click on add, the details filled are validated by 'SubjectVal.java' filter. If wrong details are filled, a message 'Kindly fill correct details' will be displayed by including addsubject.jsp page.
- After successful validation, the details are added in database in subjects table through 'AddSubject.java' servlet by calling addsub method from 'SubjectsDao.java'(data access object class) and using 'Subjects.java'(data transfer object class).

- Second option is to Assign a class for subject. By clicking Assign class for subject 'assignsubject.jsp' page is displayed. By filling details and click on assign, the details are validated by 'AssignSubVal.java' filter. On filling invalid details, a message 'Kindly fill correct details' will be displayed by including assignsubject.jsp page.
- After successful validation, the details are added in database in subclass table through 'AssignSubject.java' servlet by calling assign method from 'SubjClassDao.java' (data access object class) and using 'SubjectClass.java' (data transfer object class). The Subject-Class option displays the details of assigned subject to class from subclass table in database.
- When Teachers is clicked, 'teacherdetails.jsp' is displayed where teachers details are shown. The details are fetched from teachers table in database.
- Within the page an option Add Teacher is given. By clicking the option 'addteacher.jsp' is displayed from which teachers details can be filled and added to the list. By filling details and click on Add, the details first goes to 'TeacherVal.java' filter. If invalid details are filled a message 'Kindly fill correct details' will be displayed by including addteacher.jsp page.
- After successful validation, the details are added in database in teachers table through 'AddTeacher.java' servlet by calling addTeacher method from 'TeachersDao.java' (data access object class) and using 'Teachers.java' (data transfer object class).
- When Students is clicked, 'studentsdetails.jsp' is displayed which shows student details which are fetched from table students in database. An option Add Student is their within the page. By clicking it 'addstudent.jsp' page is displayed. Student can be added by giving correct details in the form and clicking Add. The details filled is first validated in 'StudentVal.java'. If details are incorrect a message 'Kindly fill correct details' will be displayed by including addstudent.jsp page.

- After successful validation , the details are added in database in students table through 'AddStudent.java' servlet by calling enroll method from 'StudentsDao.java'(data access object class) and using 'Students.java'(data transfer object class).
- When View Class Report is clicked, 'classreport.jsp' page is displayed in which the class report can be generated by selecting the respective class and section. 'getreport.jsp' page is displayed that consist of the class report of respective class and section. The details of class report are fetched from the database by using hibernate query language (hql).
- When Logout is clicked, the session invalidate through 'AdminLogout.java' servlet which redirect to 'adminlogin.jsp' page and displays a message 'Successfully logged out'.

Core Concepts Used:

- Java Servlets.
- Java Server Pages(JSP).
- Java Classes, Objects, Methods.
- Hibernate ORM.
- MySQL RDBMS.
- HTML, CSS.
- List.
- XML.
- Filters.
- Apache Tomcat to implement JSP and servlets.
- Session Management (HTTP session).
- Annotations.
- Servlet API Interface.
- Exception handling.

Github Link:

<https://github.com/twarit08/Phase-2-Project--Learner-sAcademy.git>

Conclusion:

The Development of the backend admin portal for Learner's Academy applies many core concepts of Java servlets, JSP, Hibernate, HTML which provides a clear understanding of client-server architecture and also to perform dynamic operations and send to web server, the web server then sends this response to web client. It also clears the concepts of Hibernate object-relational mapping (ORM) tool which provides a framework for mapping an object-oriented domain model to a relational database.