**Project Specification Document**

**Learner’s Academy**

*End of Phase 2-Become a back-end expert Project*

**Student: Nomula Devendhar**

devendharreddy.n@gmail.com

*Full Stack Java Developer*

*Master’s Program*

**Table of Contents**

1.Project Details and Objective

2.Sprint Planning

3.Project Overview

4.Implemented Concepts

5.Flow Chart

6.GitHub Link

7.Web Output Screenshots

# Project Objective

As a Full Stack Developer, design and develop an airline booking portal named as Learner’s Academy**.**

# 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. As a Full Stack Developer, design and develop a backend administrative portal for the Learner’s Academy. Use the GitHub repository to manage the project artefacts.

**The administrator can:**

* Set up a master list of all the subjects for all the classes
* Set up a master list of all the teachers
* Set up a master list of all the classes
* Assign classes for subjects from the master list
* Assign teachers to a class for a subject (A teacher can be assigned to different classes for different subjects)
* Get a master list of students (Each student must be assigned to a single class)

There will be an option to view a Class Report which will show all the information about the class, such as the list of students, subjects, and teachers

The goal of the company is to deliver a high-end quality product as early as possible.

# Sprint Planning and Tasks Achieved

The project is planned to be completed in 1 sprint. Tasks assumed to be completed in the sprint are:

* Creating the flow of the application
* Initializing git repository to track changes as development progresses.
* Writing the Java program to fulfill the requirements of the project.  Testing the Java program with different kinds of User input  Pushing code to GitHub.
* Creating this specification document highlighting application capabilities, appearance, and user interactions.

# Project Overview

The main objectives of this Projects are

* To gain an understanding of core concepts of the Java Programming

Language (abstraction, polymorphism, inheritance, and encapsulation),

* Embrace the Eclipse Integrated Development Environment (IDE),
* Understand the Agile software development life cycle
* Gain familiarity with Java data structures for object-oriented applications.
* Familiarize with concepts of Servlets , Servlet Filter, HTTP Methods
* Familiarize with concepts of Web Application Development using Servlets
* Learn Maven, A build Automation tool and handle dependencies
* To learn SQL and do CRUD Operation(Create, Read, Update, Delete)
* To learn Hibernate and design a Web Application with Servlets and perform CRUD Operation (Create, Read, Update, Delete).
* To learn Hibernate’s Association and perform @OnetoOne , @OnetoMany

@ManytoMany Associations with different tables of a database

* Learn Cookie and HTTP Session Management and Perform User’s Session

Management in a Web Application

* Learn MVC Architecture and build a Web Application using Servlets, JSP, Hibernate and perform CRUD Operations

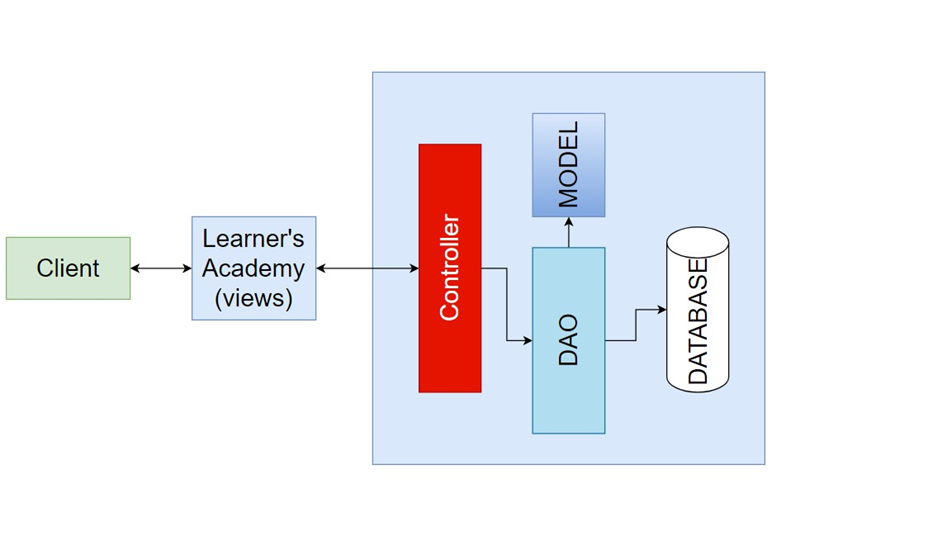
# Implemented Concepts

This section will highlight the Java concepts used to create the Learner’s

Academy Airline website. HTML, CSS, JSP, JSTL, Servlets, Servlet Filters, HTTP

Methods, SQL, JDBC Connection for SQL, Java Persistence API, Hibernate, HTTP Sessions and Cookie, Servlet MVC Architecture, Collections framework, Flow Control, Exception Handling, are the core concepts used in this program.

# Flow Chart



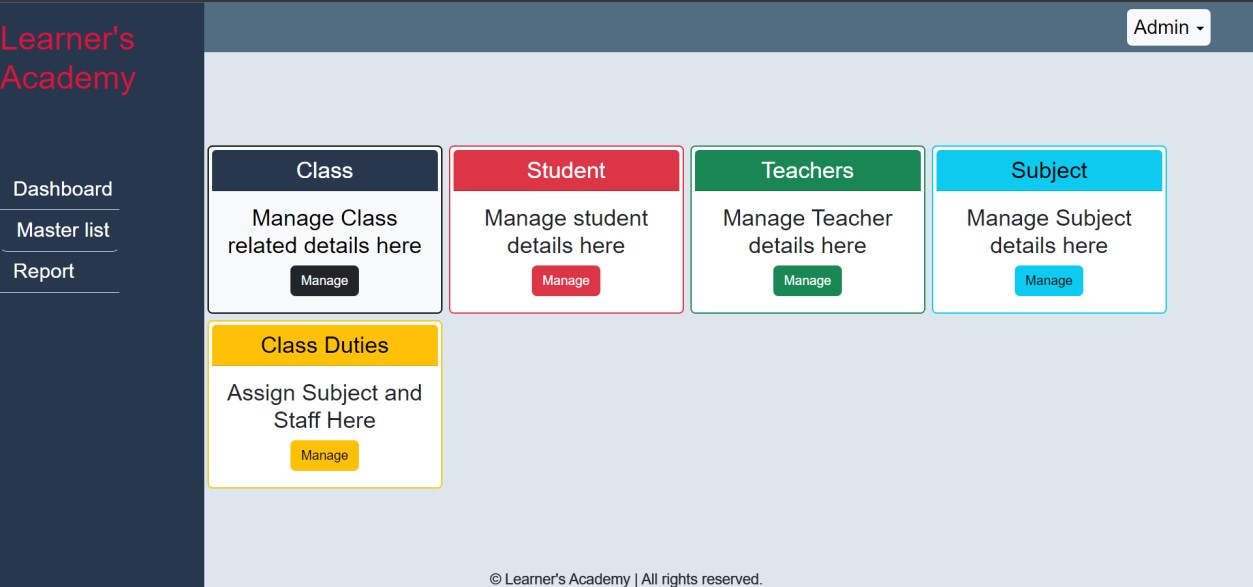
# GitHub link

I have pushed my code and associated documentation to the following GitHub link:

https://github.com/NomulaDevendhar/Simplilearn-LearnerAcademy.git

**Web Output Screenshots**

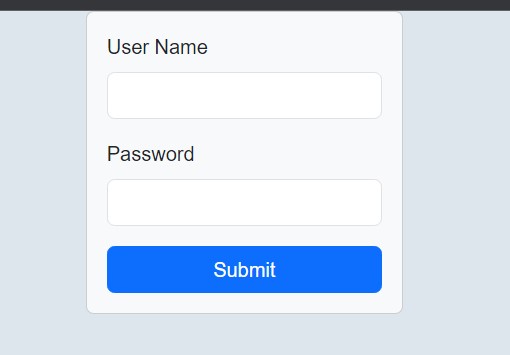
# Home Page



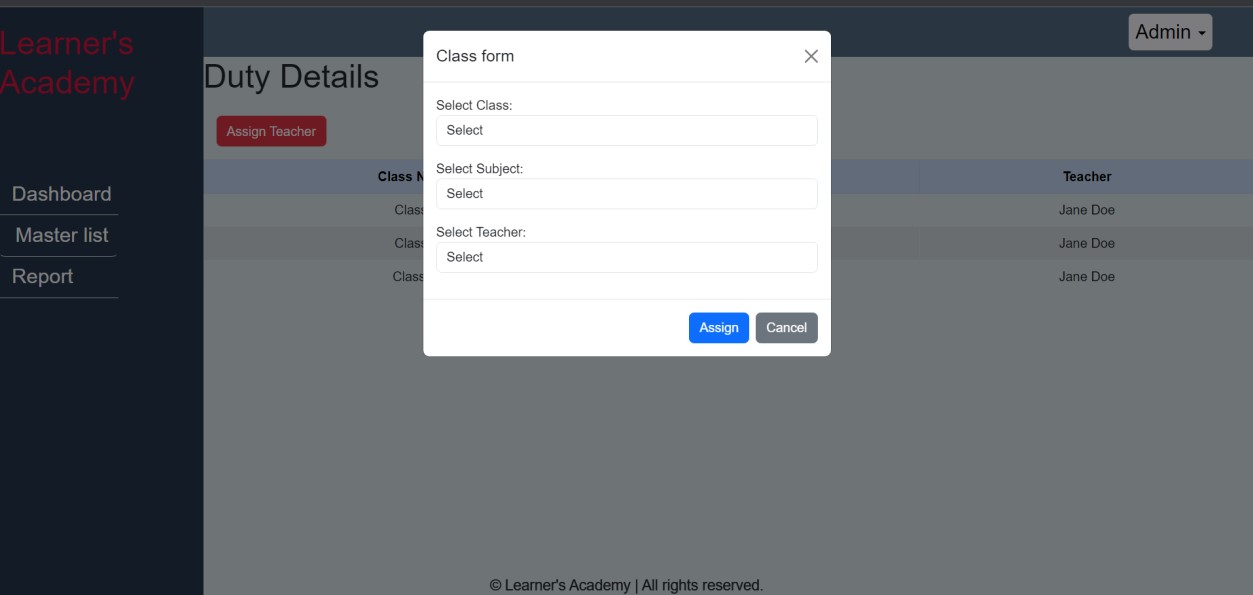
**Default Email Id: admin@gmail.com**

**Default Password: admin**

# However admin should change his password after login



# Add Duty



# Master-list of all the subjects



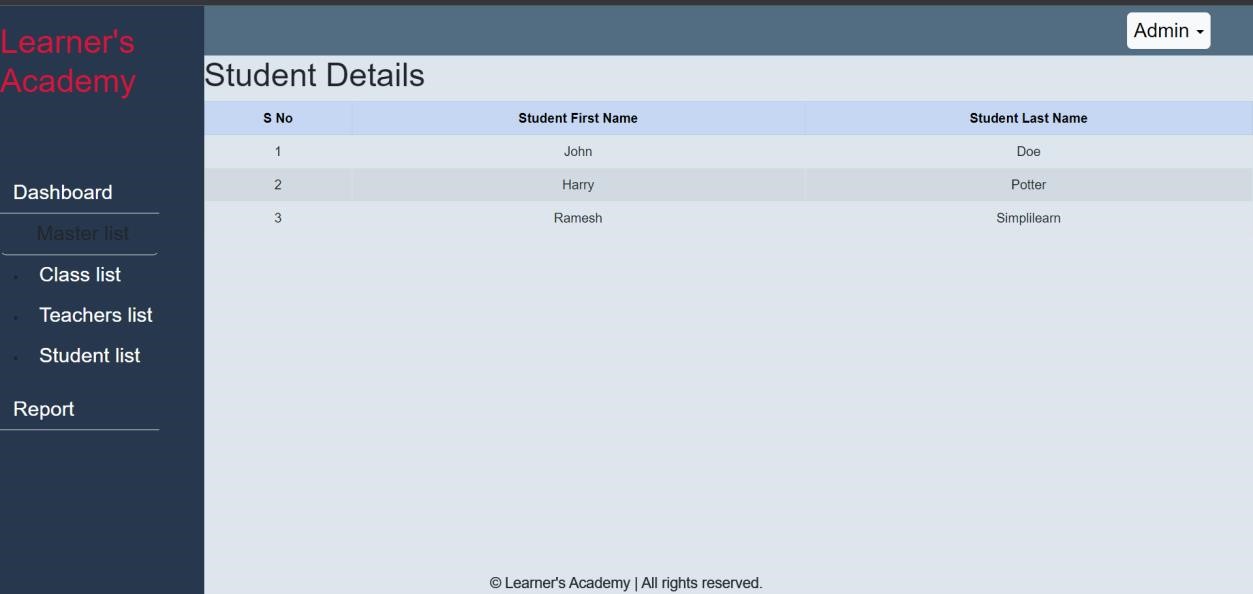
# Master-list of all the teachers



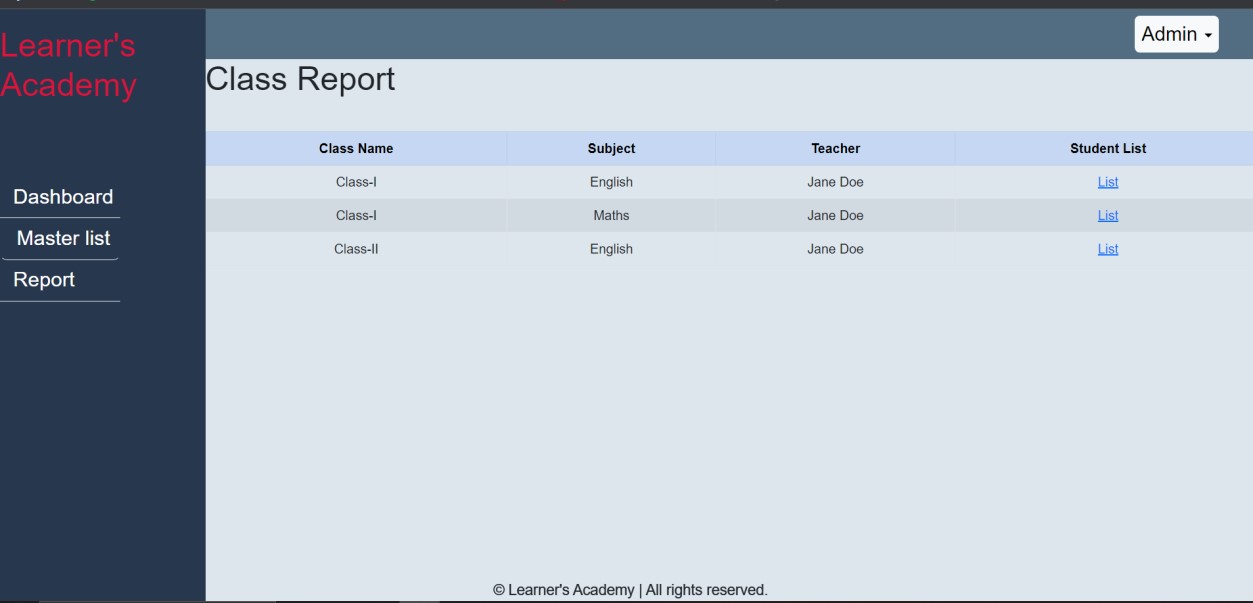
# Master-list of all the classes



# Master-list of all the students



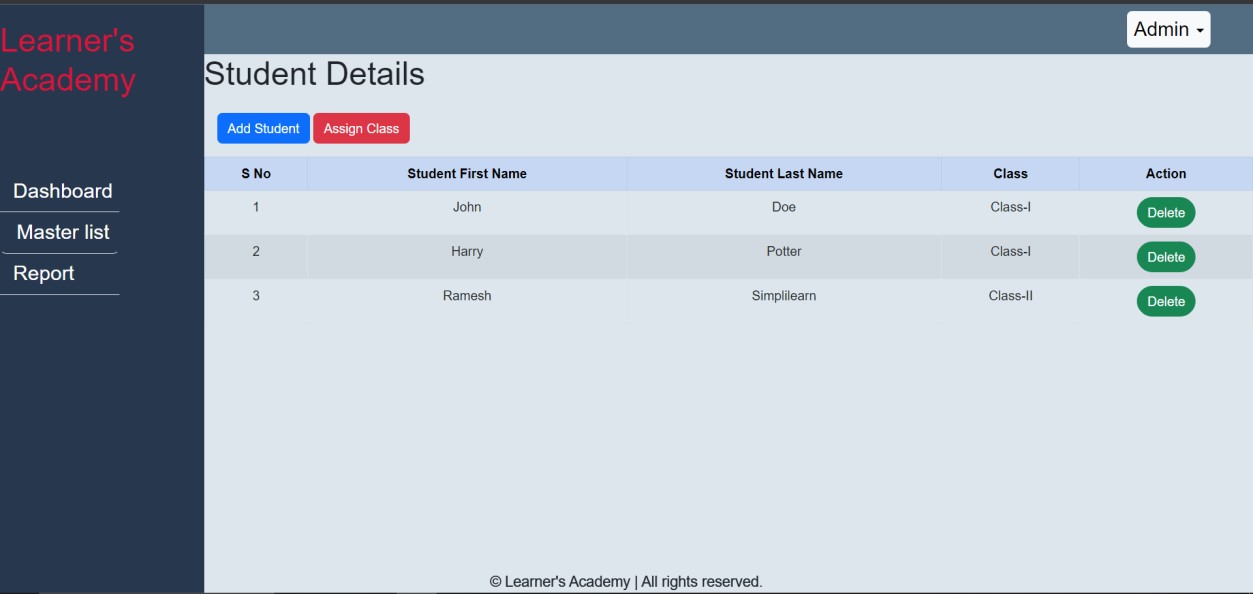
# Class Report



# Add/delete Class



# Add/delete Student



# Add/delete Teacher



# Add/delete Subject

# 