**FINAL SPECIFICATION DOCUMENT**

**Simplilearn-**

**Full stack Java**

**Developer (Assignment-2)**

**Name of Project: Learner’s Academy- Admin Portal**

**Documentation prepared by: Anjali Gopalakrishnan**

**Date submitted:**

**18-11-2022**

**Version 0.1**

| REVISION HISTORY | | | |
| --- | --- | --- | --- |
| DATE | VERSION | DESCRIPTION | AUTHOR |
| 18/11/22 | 0.1 | First version of final specification document | Anjali Gopalakrishnan |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[Name of Project: Learner’s Academy- Admin Portal 1](#_Toc119666234)

[Documentation prepared by: Anjali Gopalakrishnan 1](#_Toc119666235)

[Date submitted: 1](#_Toc119666236)

[18-11-2022 1](#_Toc119666237)

[Version 0.1 1](#_Toc119666238)

[INTRODUCTION 4](#_Toc119666239)

[PURPOSE 4](#_Toc119666240)

[PROJECT DETAILS 4](#_Toc119666241)

[SOFTWARE, PROGRAMMING LANGUAUGE AND OTHER CONCEPTS TO BE USED 4](#_Toc119666242)

[SCRUM PLANNING 5](#_Toc119666243)

[SPRINTS PLANNED 5](#_Toc119666244)

[FLOWCHART 6](#_Toc119666245)

[CORE JAVA CONCEPTS USED 6](#_Toc119666246)

[Steps for set up 7](#_Toc119666247)

[Flow of control 7](#_Toc119666248)

[ Login page 7](#_Toc119666249)

[ Homepage 8](#_Toc119666250)

[Add (Subjects, classes, teachers) 9](#_Toc119666252)

[ Add Student 9](#_Toc119666253)

[ List (Subject, class, teachers) 10](#_Toc119666254)

[ Assign (Subjects to course & Teachers to subjects) 10](#_Toc119666255)

[ Generate class report 11](#_Toc119666256)

[LINKS TO GITHUB REPOSITORY 11](#_Toc119666257)

[Project Folder, Final specification document and screenshots – 11](#_Toc119666258)

[CONCLUSION 11](#_Toc119666259)

# INTRODUCTION

## PURPOSE

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.

## PROJECT DETAILS

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

The administrator can:

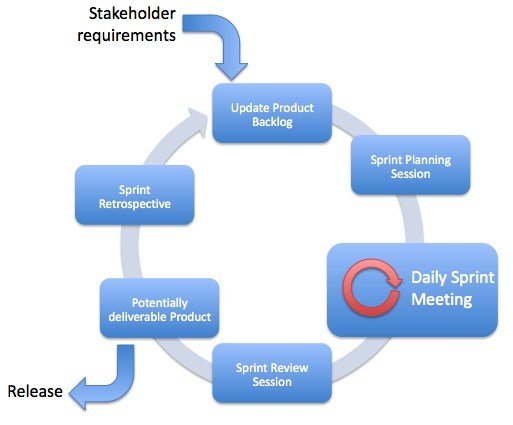
● Set up a master list of all the subjects for all the classes  
● Set up a master list of all the teachers  
● Set up a master list of all the classes  
● Assign classes for subjects from the master list  
● Assign teachers to a class for a subject (A teacher can be assigned to different classes for 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

## SOFTWARE, PROGRAMMING LANGUAUGE AND OTHER CONCEPTS TO BE USED

● Eclipse/IntelliJ: An IDE to code for the application   
● Java: A programming language to develop the web pages, databases, and others  
● SQL: To create tables for admin, classes, students, and other specifics  
● Git: To connect and push files from the local system to GitHub   
● GitHub: To store the application code and track its versions   
● Scrum: An efficient agile framework to deliver the product incrementally   
● Specification document: Any open-source document or Google Docs 

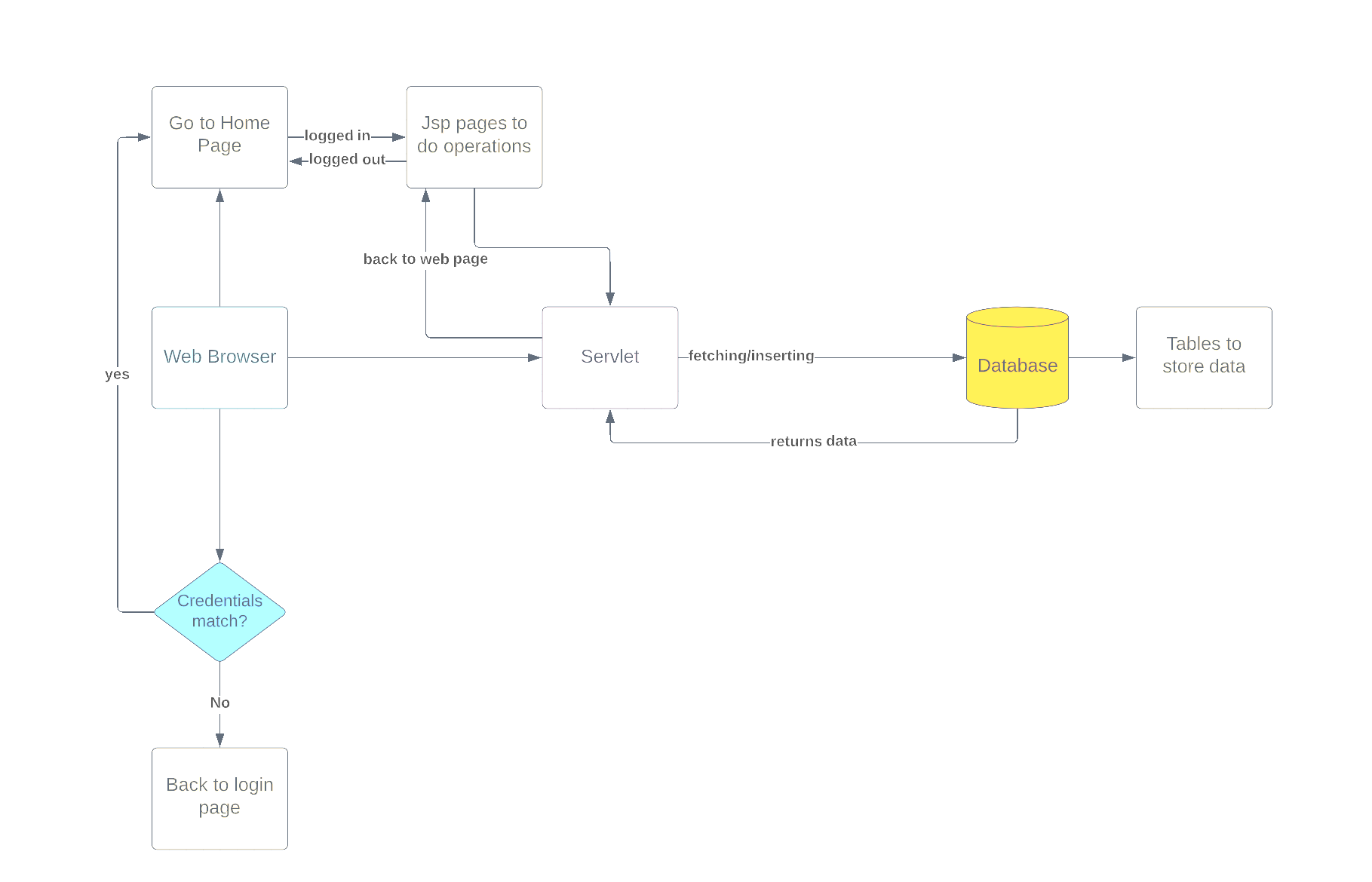
# SCRUM PLANNING

****

## SPRINTS PLANNED

* SPRINT 1
  + Goal to be accomplished – Creation of specification document and uploading to GitHub
  + Software – MS Word
  + Deliverable product – The final product specification document
* SPRINT 2
  + Goal to be accomplished – Coding the JSP pages
  + Software – Eclipse (IDE), Java, GitHub
    - Pages for
      * Add Subjects
      * Add Students
      * Add Courses
      * Add Teachers
      * Assign Subject to Courses
      * Assign Teachers to Subjects
      * Generate report
  + Deliverable product – JSP page code
* SPRINT 3
  + Goal to be accomplished – Coding servlet, models and create database
  + Software – Eclipse (IDE), Java, GitHub
  + Deliverable product – The final product

# FLOWCHART



# CORE JAVA CONCEPTS USED

* Servlet: to do the business logic and works a controller for the project
* JSP: to handle the presentation view
* SQL: to create and manage the database Files
* JDBC: to make operations on the database for the project
* CSS: to format the contents.
* Eclipse: to write and run the code
* Tomcat: to run and deploy servlet application
* Hibernate: To establish one to one mapping

# Steps for set up

1.First set up the database with the query given in the SQL Query file

2.Run application on Server

3.Add Subjects, classes and teachers

4.Assign subjects to classes

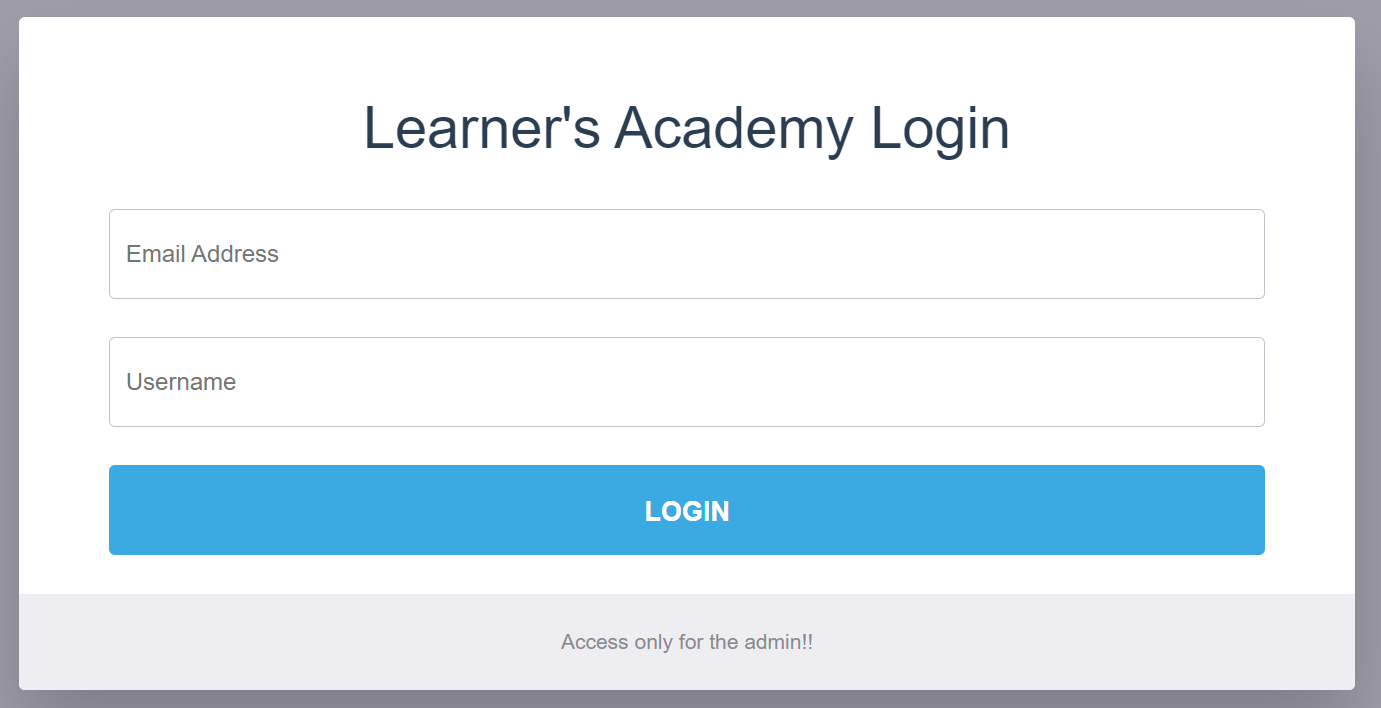
5.Assign teachers to subjects

6.Add Student

7.Generate Report

# Flow of control

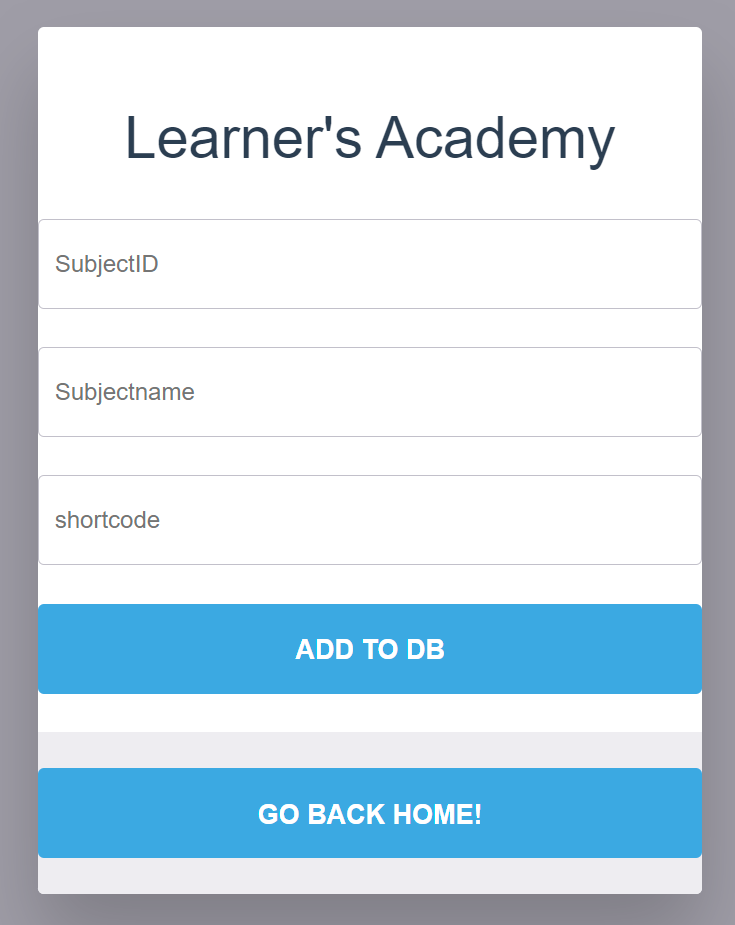
# Login page



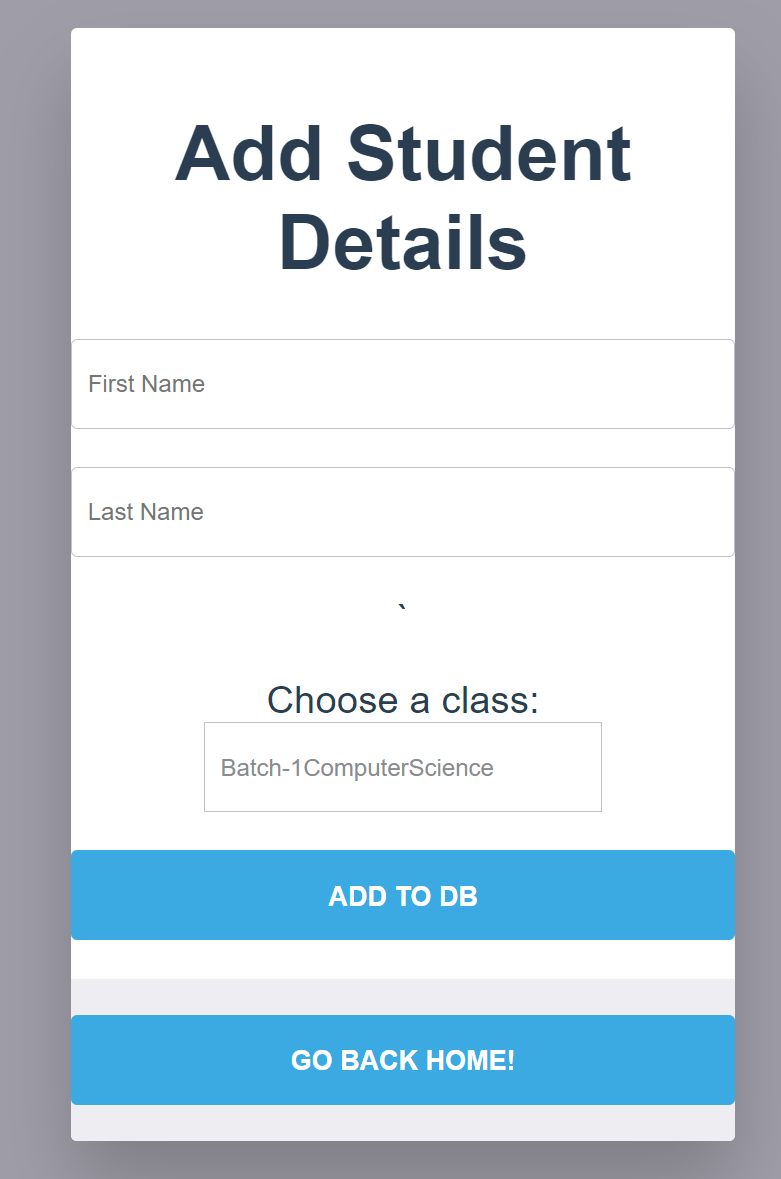
### **Homepage**

# 

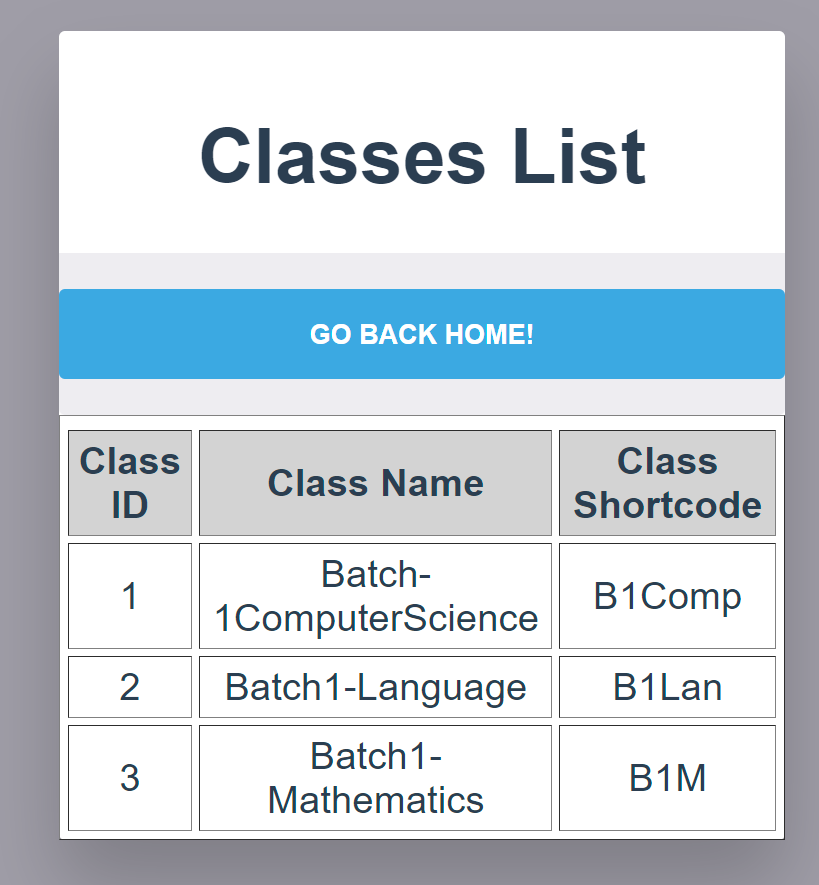
### **Add (Subjects, classes, teachers)**



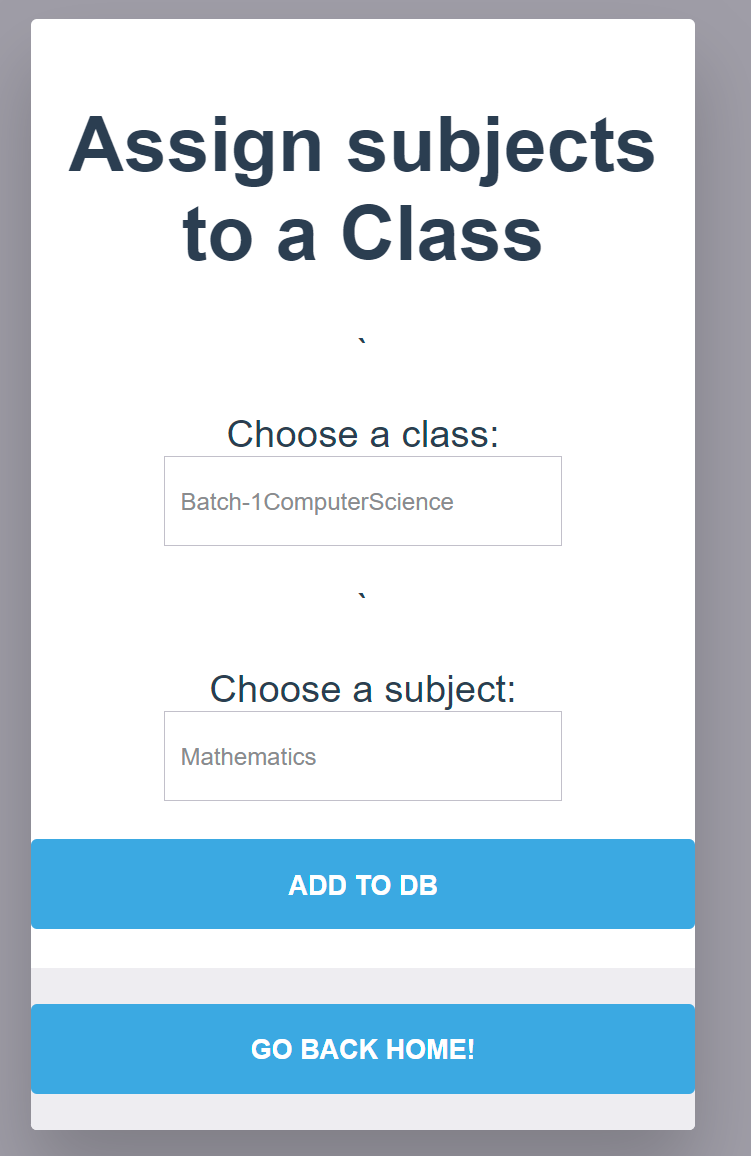
### **Add Student**



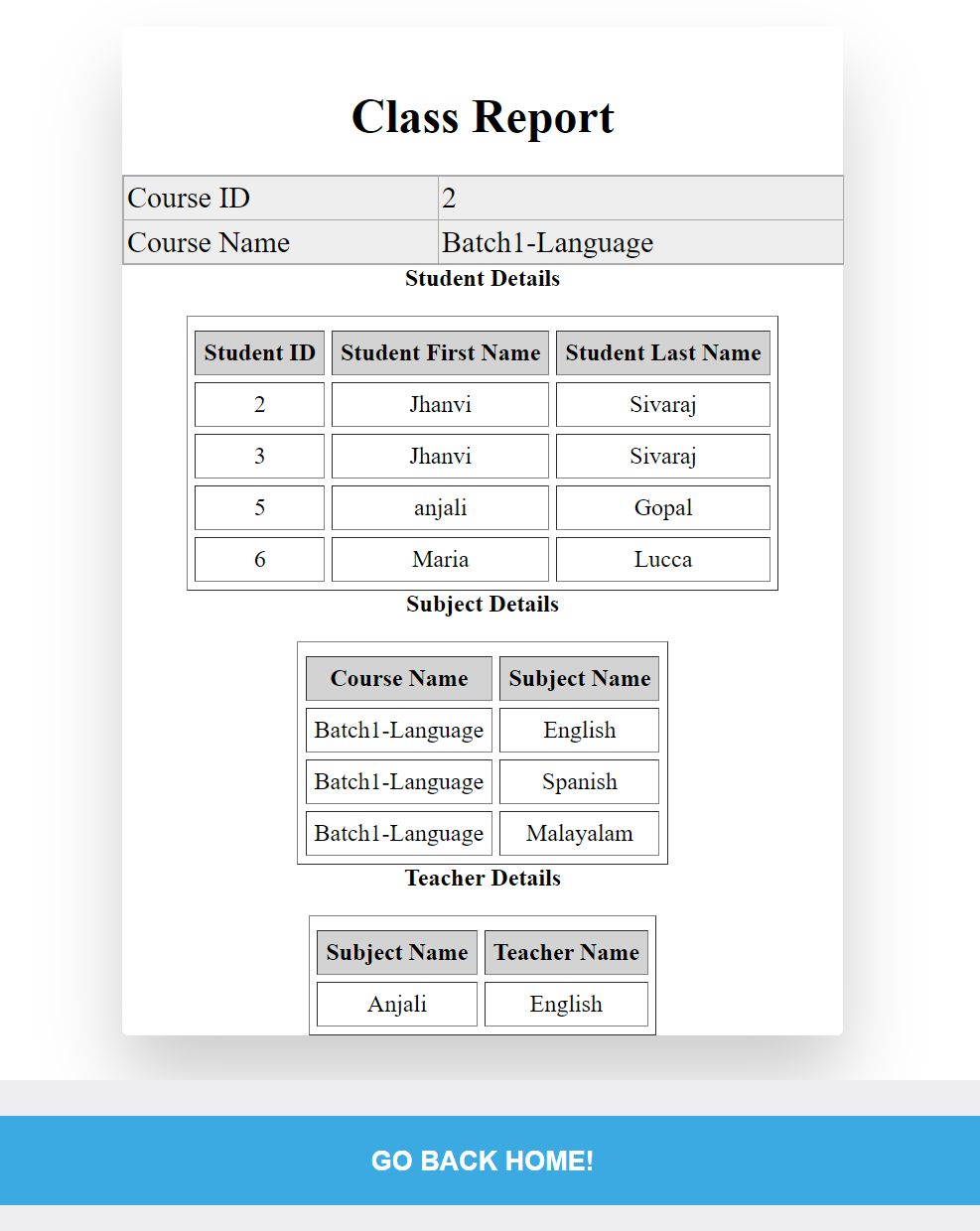
### **List (Subject, class, teachers)**



### **Assign (Subjects to course & Teachers to subjects)**



### **Generate class report**



# LINKS TO GITHUB REPOSITORY

## Project Folder, Final specification document and screenshots –

<https://github.com/AnjaliRadhe/LearnersAcademy.git>

# CONCLUSION

 The project is to design and develop a highly functional backend administrative portal for the Learner’s Academy.