Contents

Developer Information

Name: Benyam Hadgu Gebru

Email: benyamhadgu@gmail.com

Git location: https://github.com/benyamhadgug/PhaseTwoProject.git

Project Name: LearnerAdmin

Requirement

The application has the administrator as sole user. Through the application, the administrator shall:

- 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)

Work Plan

Three sprints shall constitute the project lifespan from planning to delivery:

Sprint	Stories to Complete	Deliverable
1	- Requirement's analysis - Database design - CRUD Student	- Specification Document - Executable source code
2	- CRUD Teacher - CRUD Subject	Executable source code
3	- CRUD Class - Class-Subject assignment - Report Preparation	Executable source code

Application Flow

Deploying the application

The following environment shall be setup for using the developed application

- MySQL 5.*
- Apache Tomcat 9
- Java JDK 1.8
- Eclipse EE

Other Configurations and setup

The MySQL database account for use in the application is **root**. A password **root** shall be used to connect the application to persistence layer MySQL database. The database name is **academy**.
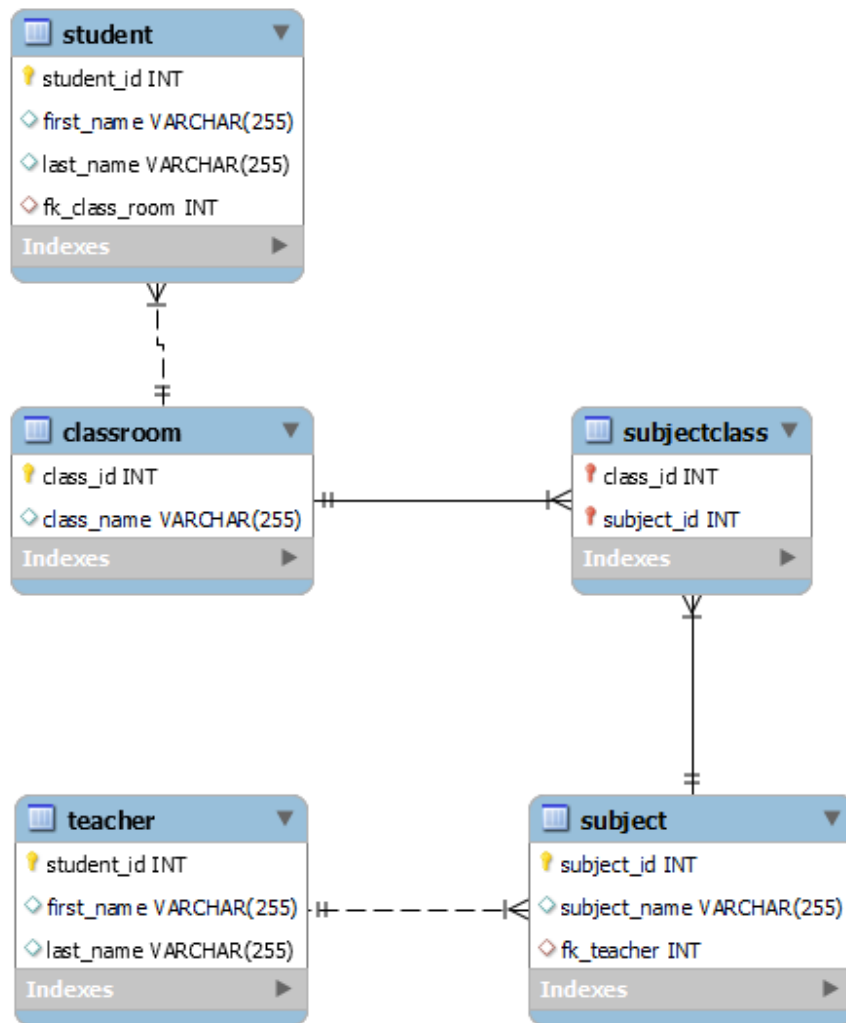
For convenience, the schema tables and their content is included in the project delivery. The folder containing those artifacts is name **Schema_Export**.

Application snapshots

The application's UI snapshots are organized according the subject_action images. For instance, student adding UI is named **students_add.** The snapshot collection for the application's UI is included in the folder **Snapshots**. Moreover, **compiled_snapshots.pdf** contains all snapshots in one file.

Entity Relationship Model

The application's database layer, reflective of the domain model, includes the following concepts.

student
- 🔑 student_id INT
- ◇ first_name VARCHAR(255)
- ◇ last_name VARCHAR(255)
- ◇ fk_class_room INT
- Indexes ▶

classroom
- 🔑 class_id INT
- ◇ class_name VARCHAR(255)
- Indexes ▶

subjectclass
- 🔑 class_id INT
- 🔑 subject_id INT
- Indexes ▶

teacher
- 🔑 student_id INT
- ◇ first_name VARCHAR(255)
- ◇ last_name VARCHAR(255)
- Indexes ▶

subject
- 🔑 subject_id INT
- ◇ subject_name VARCHAR(255)
- ◇ fk_teacher INT
- Indexes ▶

Core concepts and Algorithms used

The following concepts and algorithms were used in the application development process.

Java Collection and operations

- Set
- List

Query Language

- SQL
- HQL

Persistence Layer Patterns

- MVC pattern
- DAO pattern

Entity Relationships/Associations

- OneToMany association
- ManyToMany association

Web Technologies

- JSTL
- JSP
- Servlet
- CSS

Other concepts

- Database Transaction (transation begin and commit)
- Hiberate Annotations
- Fetch Types (EAGER)