**NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY**



**DEPARTMENT OF INFORMATION**

**SCIENCE AND ENGINEERING**

# TIMETABLE GENERATOR

**INDEX:**

1. **Introduction**
2. **Project Goal**
3. **Project Description**
4. **System Requirements**
5. **ER Diagram**
6. **Schema Diagram**
7. **Normalization**
8. **Prototype**
9. **Results**

## 10. Conclusion

## INTRODUCTION

Most college administrative work has been computerized, the

lecture timetable scheduling is still mostly done manually due to its inherent difficulties. It is a Constraint satisfaction problem in which we find a solution that satisfies the given set of constraints. A college timetable is a temporal arrangement of a set of lectures and classrooms in which all given constraints are satisfied. Creating such timetables manually is a complex and time-consuming process. By automating this process with a computer assisted timetable generator can save a lot of precious time.

Hence, we intend to develop a practical approach for building

lecture course timetabling systems, which can be customized to fit any college's timetabling problem. We aim to develop an ***Timetable Generator*** using PHP, CSS, Bootstrap, and JavaScript that contains an admin side from where a user can manage all the timetables and records easily. The admin plays an important role in the management of this system. In this project, the user must perform all the main functions from the admin side. It contains the teacher side and student side from where they can login and easily access the timetable.

We have verified the attributes and entities and discovered relationships among entities. Then we have constructed an entity-relationship diagram, converted it to a schema and designed an example prototype on how our project will look.

### Existing System

* The current system is totally manual.
* Due to the manual process, it requires more time for completion ofany work.
* Lots of time consumed is difficult and time consuming.
* Manual system takes more time to complete any task

. • This system requires more human resources to run the system.

* Require infrastructure to operate the system

Need for New System

* To reduce the time required for generating timetables than the existingsystem.
* To increase efficiency and accuracy of the proposed system.
* To reduce paper and labor work.
* The System requires less human resources.

### PROJECT GOAL

● Timetable generator focus on automating the manually generated timetable process which saves time for an admin. The project focuses on a web-based application where the end user- Staff and Student can access their respective timetables and a admin side who is responsible for giving inputs and constraints for timetable generation. The timetable includes details like the department, semester, subject and the allotted time.

### PROJECT DESCRIPTION

Timetable generator is a web-based application that guides us about timetable management system. The project is a practical approach for building lecture course timetabling system. It finds suitable time slots and classrooms which satisfies a given set of constraints.

The project contains admin (who manages all the tasks), staff and student (end users) side

**Admin**: Responsible for generating time table by taking consideration of all teachers, courses, departments and semester

EID - email ID of the user

USER\_ID - ID of the user logged in

USER\_NAME - name of the user logged in PASSWORD - login password

**Teacher**: employee of the college who handles a particular subject for a particular department

TEACHER\_ID - id of the teacher

TEACHER\_NAME- name of the teacher

ADDRESS - address of the teacher

DOB - date of birth of the teacher

GENDER - gender of the teacher

MOBILE NUM - mobile number of the teacher

DEPARTMENT\_ID - ID of the department in which the teacher works

SEMESTER\_ID – ID of the semester handled by the teacher

**Student**: details of candidates enrolled for a department

STUDENT\_ID - id of the student

STUDENT\_NAME - name of the student

ADDRESS - address of the student

DOB - date of birth of the student

GENDER - gender of the student

DATE - date on which the student joined

MOBILE NUM - mobile number of the student

**Department**: specifies the number and name of departments in a college

DEPARTMENT\_ID - ID of the department

DEPARTMENT\_NAME - name of the department

**Semester**: specifies name of the semester SEMESTER\_ID - semester id

SEMESTER\_NAME - name of the semester the user belongs to

**Subject**: specifies subjects of a particular course in a particular semester

SUBJECT\_ID - ID of the subject SUBJECT\_NAME - name of the subject

**7. TIME SCHEDULE:**

TIMESCHEDULE\_ID- time slot for a particular subject

#### FEATURES OF PROJECT

* Admin Panel
* Student Panel
* Staff Panel
* User management system
* Adds teachers,students,subjects,semester,departments and subjects

•Updates and deletes all the fields when necessary

•Time schedule of teachers and students can be viewed

### SYSTEM REQUIREMENTS

Front end: HTML, CSS,Bootstrap,JavaScript

Back end: PHP, MySQL

1. PHP: Hypertext Preprocessor (PHP) is a technology that allows softwaredevelopers to create dynamically generated data.
2. web pages, in HTML, XML, or other document types, as per clientrequest. PHP is open source software.
3. MySQL: MySql is a database, widely used for accessing, querying,updating, and managing data in the database.
4. PhpMyAdmin: phpMyAdmin is a free software tool written in PHP,intended to handle the administration of MySQL over the Web.
5. XAMPP Server / WAMP Server: XAMPP helps a local host or server totest its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself

#### FUNCTIONAL REQUIREMENTS

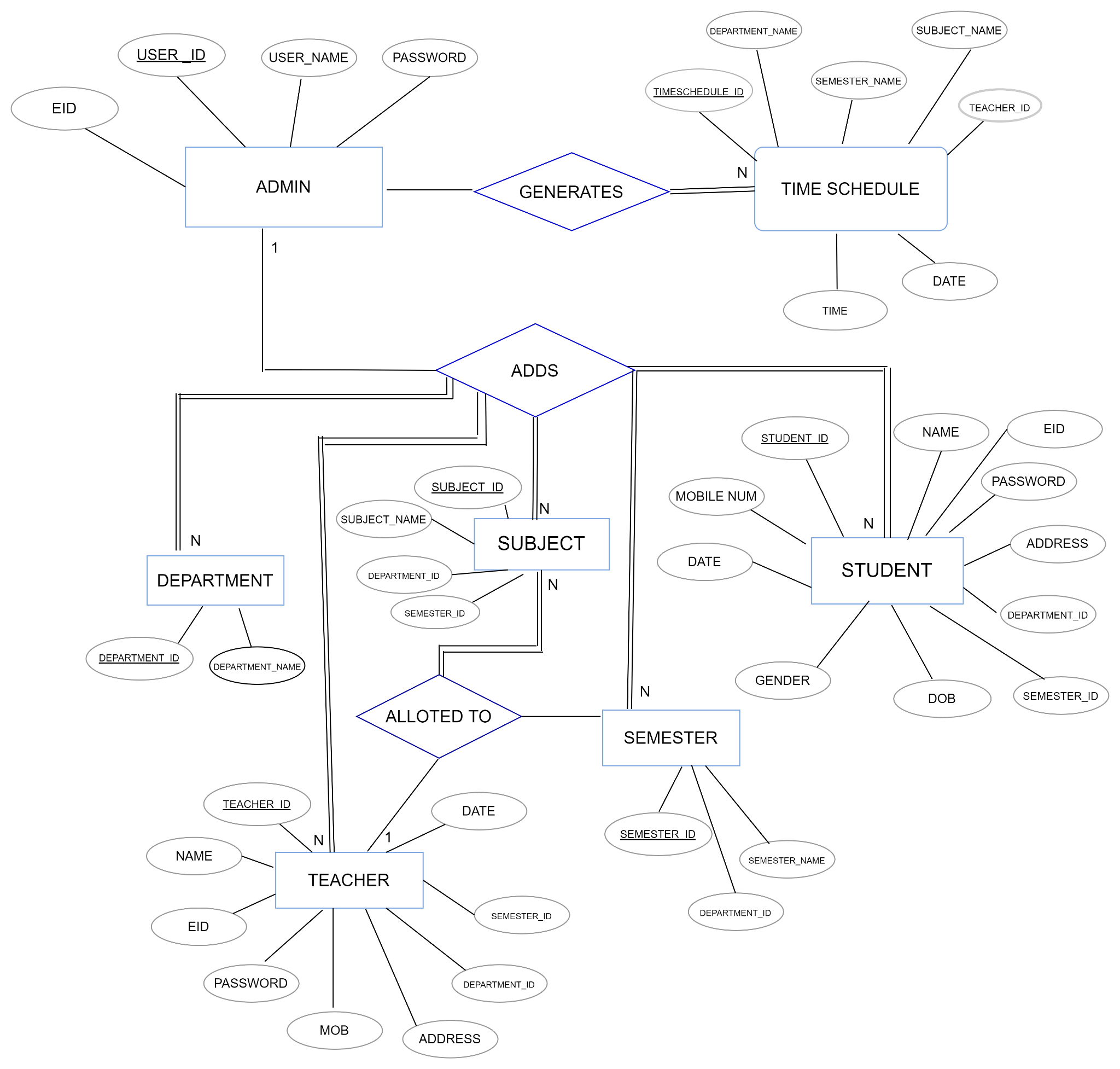
•To implement a User interface on the system

•User-friendly front-end design using CSS.

•Strong authentication while performing various operations.

•JavaScript validations and alerts wherever needed.

## ER DIAGRAM



## RELATIONS

An admin can add many teachers resulting in 1:N relationship

An admin can add many students resulting in 1:N relationship

An admin can add many departments resulting in 1:N relationship

An admin can add many subjects resulting in 1:N relationship

An admin can add many semesters resulting in 1:N relationship

A teacher can be allotted one or more subjects resulting in

1:N relationship

## SCHEMA DIAGRAM

### NORMALIZATION

**Attributes used :**

EID **-** email ID of the user

USER\_ID - ID of the user logged in

USER\_NAME - name of the user logged in

PASSWORD - login password

DEPARTMENT\_ID - ID of the department

DEPARTMENT\_NAME - name of the department

SUBJECT\_ID - ID of the subject

SUBJECT\_NAME - name of the subject

SEMESTER\_ID - semester id

SEMESTER\_NAME - name of the semester the user belongs to

STUDENT\_ID - id of the student

NAME - name of the student

ADDRESS - address of the user

DOB - date of birth of the student

GENDER - gender of the student

DATE - date on which the user logged in

MOBILE NUM - mobile number of the user

TEACHER\_ID - id of the teacher

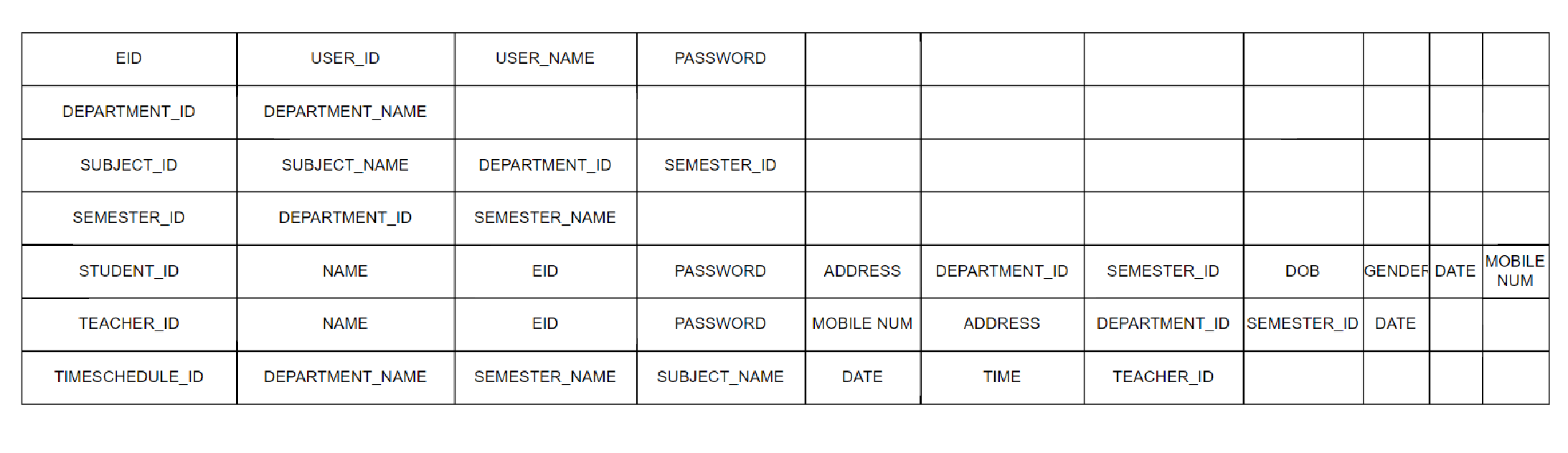
TIMESCHEDULE\_ID -id of the schedule time

DATE- scheduled time date

TIME - time at which the subject is taught

FIRST NORMAL FORM:

Rule 1: atomic values(no composite values) Simple valued attributes



SECOND NORMAL FORM:

Rule 1: be in 1NF

Rule 2: if partial functional dependency exists then divide the table(relation)

**FD1:** USER\_ID → USER\_NAME,EID,PASSWORD

**FD2:** DEPARTMENT\_ID → DEPARTMENT\_NAME

**FD3:** SUBJECT\_ID → SUBJECT\_NAME,DEPARTMENT\_ID,SEMESTER\_ID

**FD4:** SEMESTER\_ID → DEPARTMENT\_ID,SEMESTER\_NAME

**FD5:** STUDENT\_ID →

NAME,EID,PASSWORD,ADDRESS,DEPARTMENT\_ID,SEMESTER\_ID,DOB,GEND

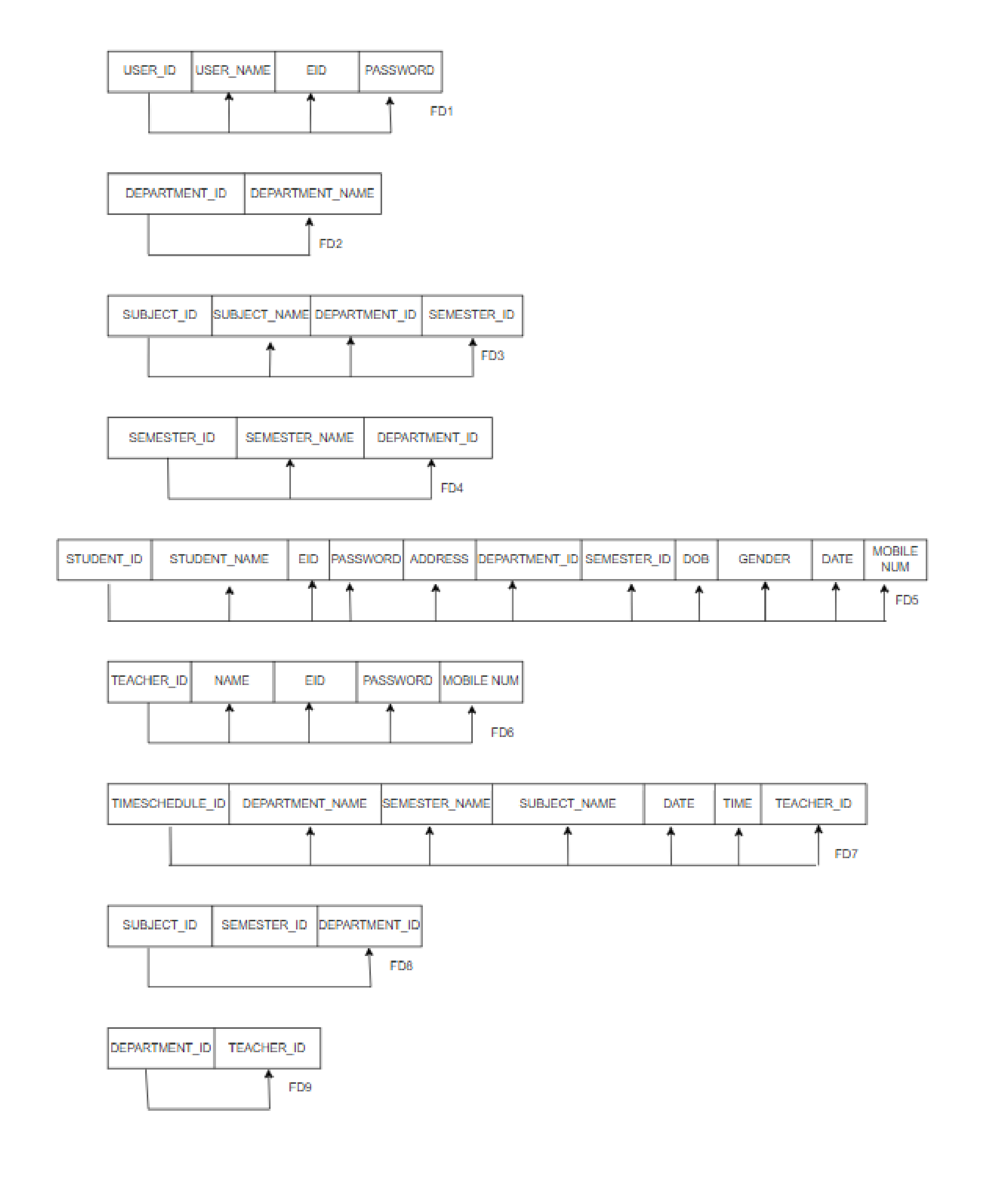
ER,DATE,MOBILE NUM

**FD6:** TEACHER\_ID → NAME,EID,PASSWORD,MOBILE NUM

**FD7:** TIMESCHEDULE\_ID → DEPARTMENT\_NAME,SEMESTER\_NAME

**FD8:** SUBJECT\_ID → SEMESTER\_ID,DEPARTMENT\_ID;

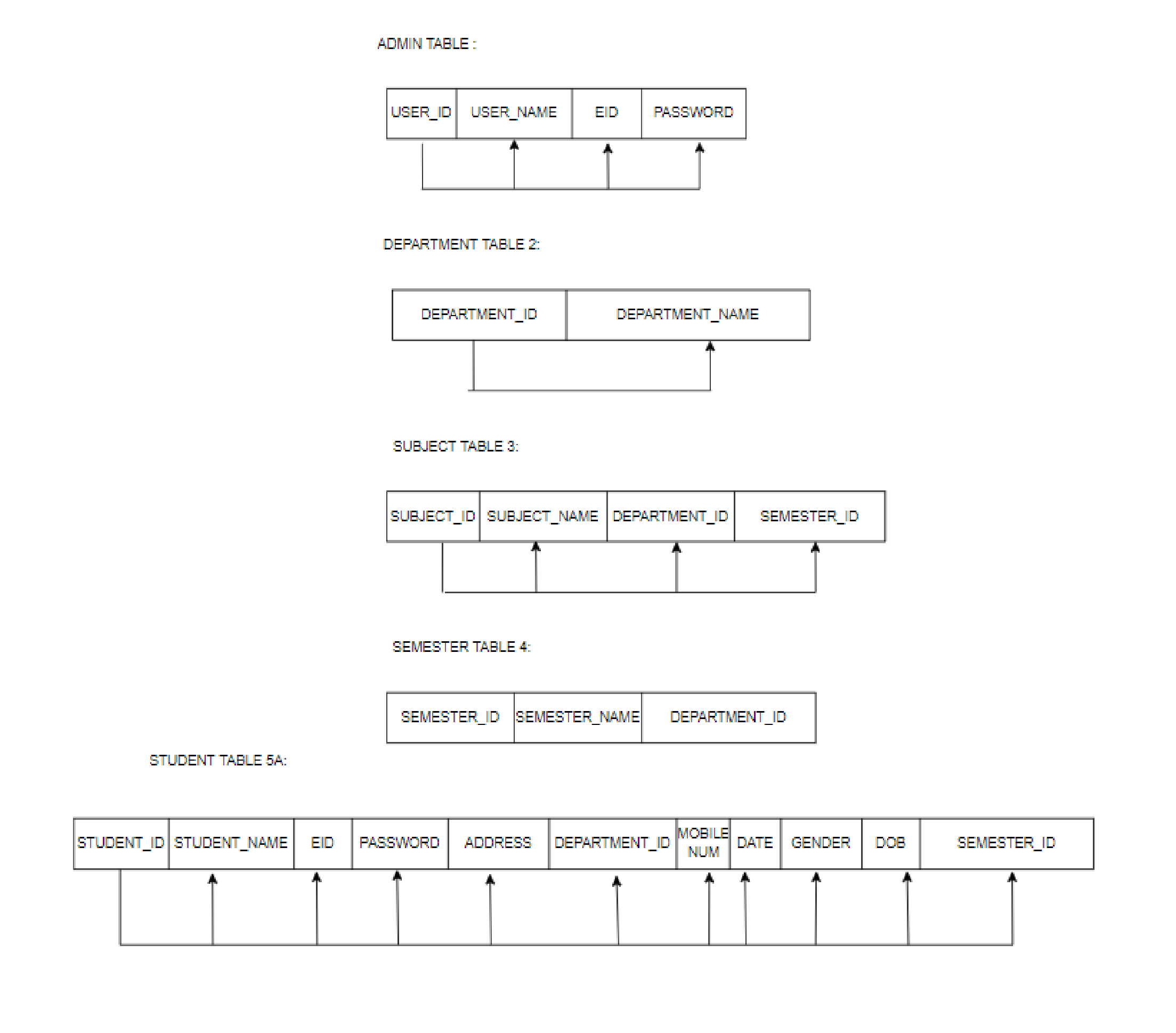
**FD9:** DEPARTMENT\_ID → TEACHER\_ID

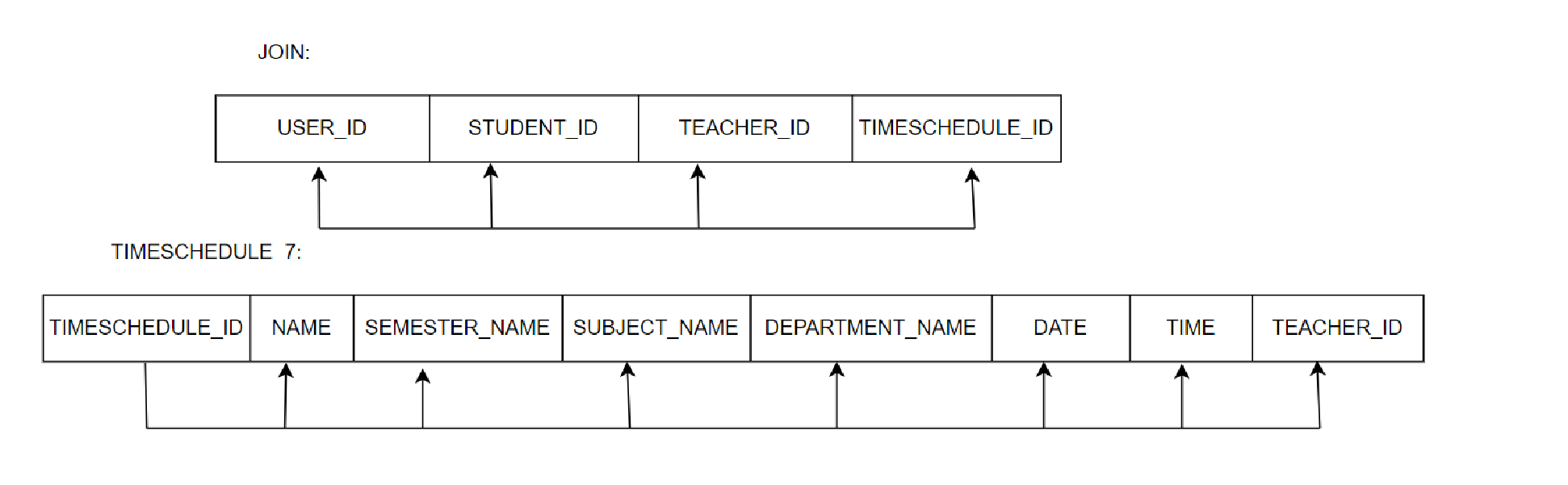
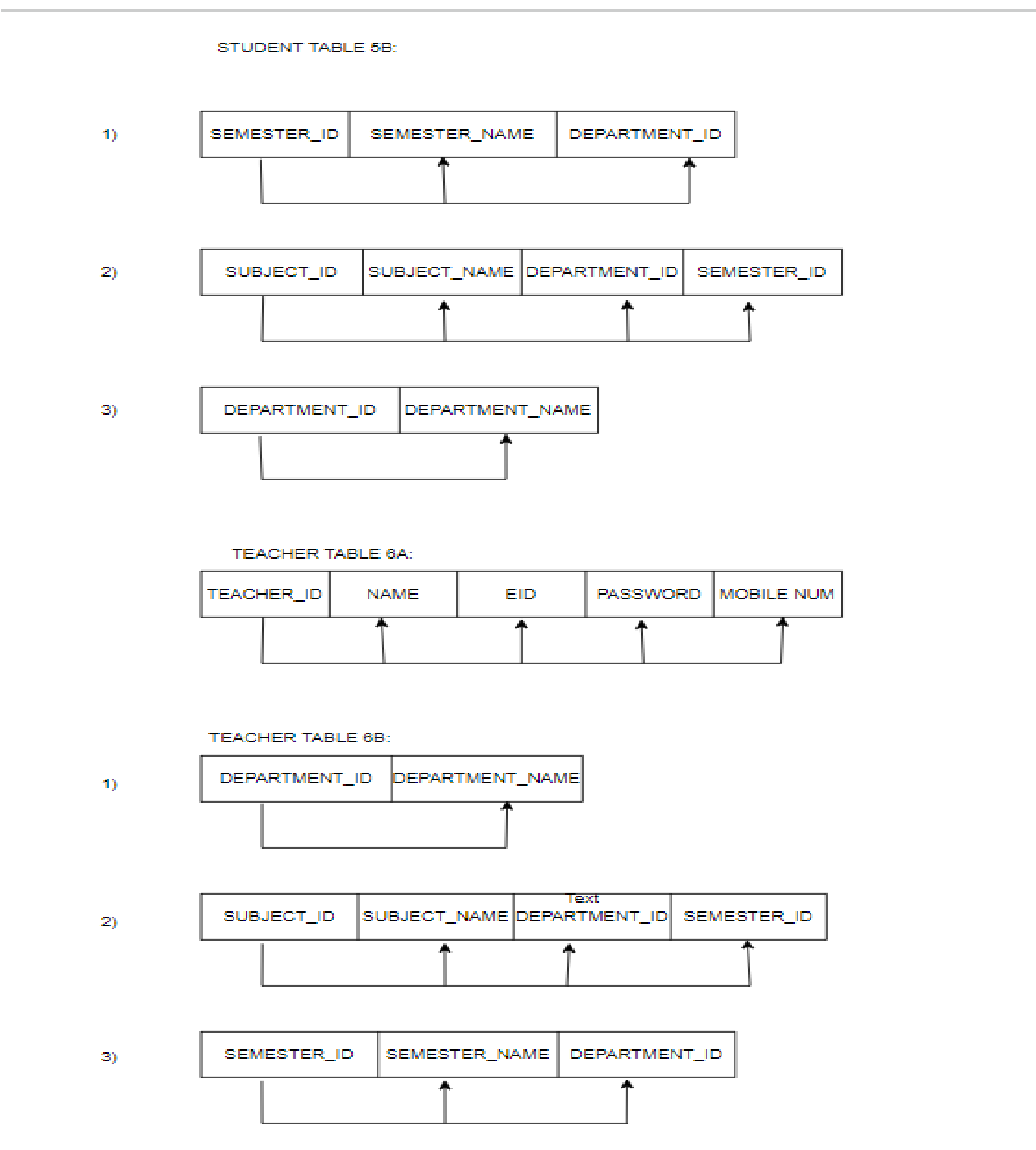


THIRD NORMAL FORM:

Rule 1: be in 3NF

Rule2 : there shouldn’t be any transitive functional dependencies, if exists divide the table(relation)

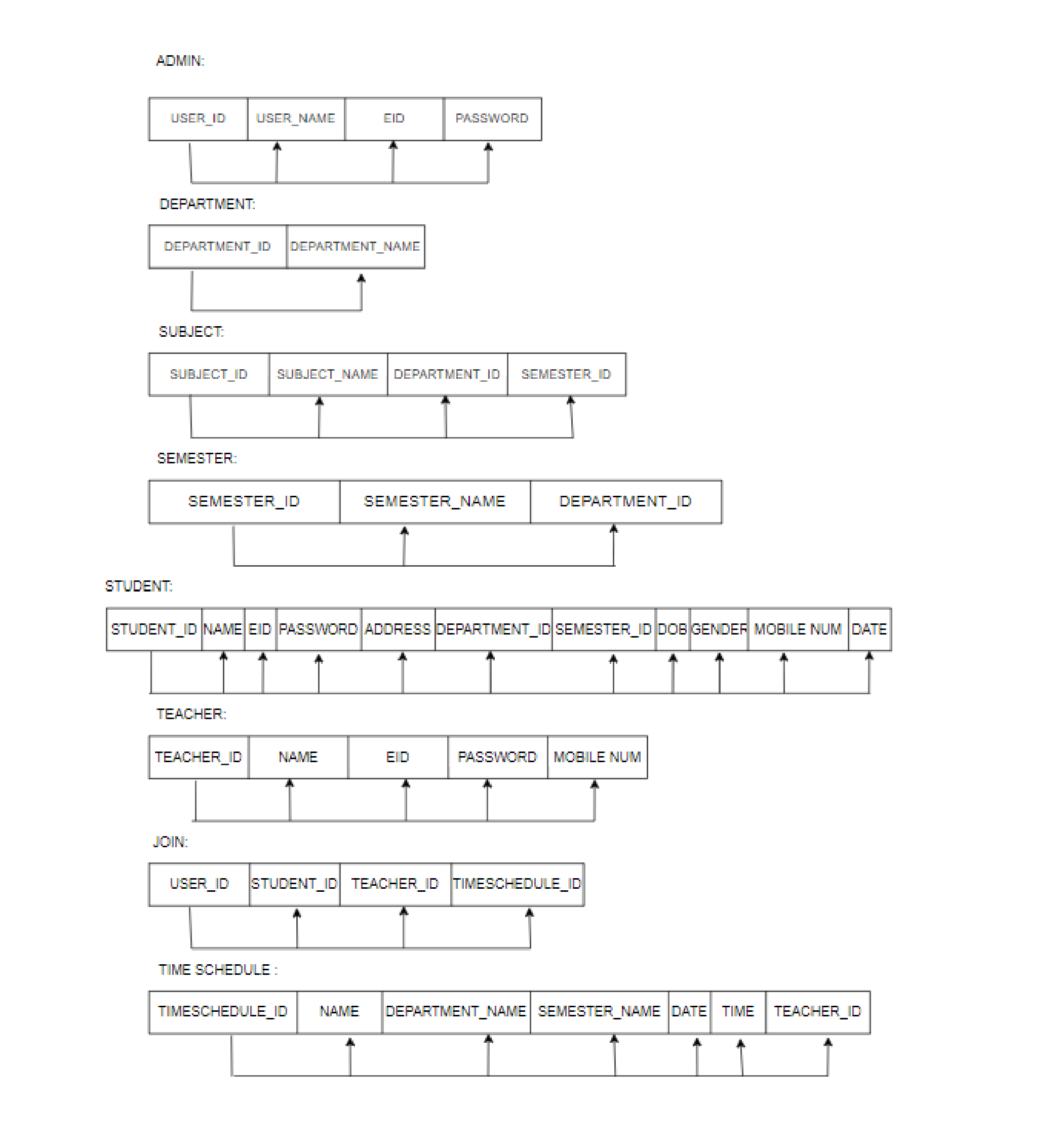




Boyce Codd NORMAL FORM:

Rule 1: should be in 3NF

Rule 2: no multivalued dependency



#### RESULT

The resulting system is able to:

* To add departments, semester ,subjects, students, teachers and time table
* To update and delete all the feilds successfully
* To generate the timetable
* To make a time schedule for all the teachers and students successfully.

##### CONCLUSION

Generally, this system can be considered a useful system since it helps the lecturer to improve their process of preparing the timetable. By providing support through the Timetable Management system, the usage can be increased to any faculties. If the system is successfully uploaded to host, to assist administrator, lecturer and student on how to use the system.

Our approach of developing an automated timetable system is successful in solving college lecture-course timetabling problems. We have also shown how we can fit our timetabling system as a Rich Desktop Application. The graphical user interface used in this application provides an easy way in understanding how the application works and also makes ease in providing the input. This application is provided with necessary details of faculty and subjects which are stored in a database and then by making use of the available data it generates the lecture-course timetable with minimum time when compared to manual generation.