STUDENT-TEACHER APPOINTMENT SYSTEM

Project by

1. Induraj Pudhupattu Ramamurthy
2. Murali Krishna Mopi Devi
3. Kofi Drammane
4. Sachin Jangoli

Contents

List of Figures:

List of Tables

Acknowledgment

First and foremost, our biggest thanks is to Almighty for us having completed this project. Secondly, our sincere Gratitude and appraisal go to our Teacher Mr. Tahar Gherbi for the valuable lessons taught in the course work. We are delighted in pronouncing that the skillset earned in the course work was worthy enough to develop such project by ourselves!

Abstract

Student-Teacher appointment system is the system that can facilitate students and teachers to make appointments with each other. In the analysis phase, we found that the EISTI doesn't have such appointment system. So we believe that this project developed as prototype will be a crucial asset to the university for realizing the appointment system in real time.

By this student teacher appointment system, if a student wants an appointment with a teacher, he first checks the free time listed by the particular teacher and if the mentioned time interests the student then student request teacher for an appointment. Similarly, the teacher can also request a student to meet him at a particular time! The project mainly focuses on providing a platform between student and teacher to schedule their appointments with each other. In this Project, we are using Java-Fx to create the graphical user interface and the core java to implement the projects functionalities.

1. INTRODUCTION

EISTI - As an engineering school officially recognized by the State and made competent by the CTI (French Engineering accreditation institution) the EISTI's vocation trains future engineers in Mathematics and Computing. Having this in mind our team choose to develop an appointment system for the college to streamline the process of appointments requested and given vice versa by students as well as teacher.

1.1 Background of the project

The project done here is intended to effectively create a medium between teachers and students, by which each one of them can fix appointments, store and manage the appointments. This proposed system intends to create a graphical user interface that will act as a communication platform for the task of appointment scheduling in a user friendly way.

1.2 Team Composition

Title: Student Teacher Appointment System			
S.No	***Name***	***Email***	***Responsibility***
1	Induraj Pudhupattu Ramamuthy	pudhupattu@eisti.fr	Developer & GUI designer
2	Murali Krishna MopiDevi	mopidevimu@eisti.fr	Developer & GUI designer
3	Koffii Dramane fotie	fotiekoffi@eisti.eu	Analyzer & Tester
4	Sachin Jangoli	jangonisac@eisti.eu	Analyzer & Tester

Table 1. Team Composition

1.3 Statement of Problems in the existing system

The proposed system overcomes the problems in the existing system, few of such problems are listed here,

- The existing system is not a computerized system, it's completely manual system that causes time wastage and frustration given that a student or teacher is not able to meet for appointments

- Student if he needs to meet a teacher, he has no idea of when teacher will be available in his office. It's tiresome process for the student to go every time to teacher office to check if the teacher is available!
- Teacher if he needs to meet a random student in the class he teaches, he has no mechanism to communicate with that student of his choice,
- Record of appointment made either by student or teacher is not well kept
- No report generation mechanism
- Difficult to track and manage the appointments
- Difficult to have statistical report of consultation hours made by teacher.

1.4 Objectives of the Project

1.4.1 General Objective

The general objective of the proposed system is to transcend from using manual system to complete computerized system

1.4.2 Specific Objective of the Project

- Study and analyze the existing system
- Identify the problems of the existing system
- Identifying the basic functionalities and non-functionalities of the system
- Design the front end system and the back end system
- Incorporating the front end and back end with the database
- Implementing the designed system

1.4.3 Scope of the Project

The scope of our project is to design user friendly quick and effective appointment system.The boundary of the project is restricted to theses following functionalities,

- To register teachers and student information's
- To update or delete teacher and student information's
- To assign teacher to a particular course and department
- To enroll a student to a particular course
- Teacher entering his consultation hours
- Teacher requesting a student to meet him during consultation hours
- Student requesting a teacher for appointment
- Teacher accepting or deleting appointments made or given to him

1.5 Feasibility Analysis

1.5.1 Operational Feasibility

The System developed will provide accurate, active and secured service and decreases the time wastage and also it is not limited to a particular group or body. The system is platform independent and it runs in all operating system and types of computer systems. Since its operations are not just confined to EISTI alone, After the testing and addition of future enhancements mentioned in the later part of documentation, this system can be implemented by any educational institutions. The system does have an interactive user friendly GUI developed by java FX, hence all the actors of the system can effectively use the system without much of technical knowledge, thus making the system much more operationally feasible.

1.5.2 Technical Feasibility

Although The system developed uses core Java, MySQL and Java Fx. The final system can be run on any system having JVM installed on their core. And the system usage itself requires internet connectivity to access the 24x7 active server. So with technical point of view, the system is considered technically feasible.

1.6 Methodology

Unlike using waterfall development methodology, Because of the limited time availability, we used parallel processing methodology to develop and implement the prototype before the given deadline.

1.7 Developmental Tools

S.No	Activities	Tools
1	Database Design & Implementation	Wamp Server
2	System analysis Designing	https://www.draw.io/
3	Back end Programming	Eclipse IDE
4	Gui Design	Scene Builder, Icons 8

Table 2 Software Requirement

1.8 Testing

We used unit testing, integration testing and system testing. By individual testing, we tested the individual components of the system. All components have been tested to maximum satisfaction. By integration testing, we checked if the integrated components integrate well with each other. The last testing namely the system testing helped us realize the operational and technical feasibility of the overall system by running the whole system on different platforms;

2 SYSTEM ANALYSIS

2.1 Proposed system Overview

Our goal is to propose a computerized system that is reliable and operational 24x7. This system is believed to solve the problems in the manual system. Our new system will store the files in more secured and reliable way, we will use SQL database to store details of students, teachers and their appointments. The teacher in need to meet a particular student will basically access this system with his own login credentials, he will enter his consultation hours and will request the student to meet him at his consultation hour! Similarly, the student in need of teacher appointment, will first look at the teacher's consultation hours and if he can visit the teacher at the respective consultation hour then he will request the respective teacher for an appointment!

2.2 Business Rule

The business rule in this system for each actor of the system namely Admin, Teacher, and student are listed below,

1. Admin
   - 1.1. Admin adds teacher and students to the system, they will be given default password which they can change on their next login
   - 1.2. Admin can modify or delete any of the teacher and student in the system
   - 1.3. Admin adds list of course and departments
   - 1.4. Admin assigns teacher a department and courses, similarly he assigns students to concerned department of their choice
2. Teacher
   - 2.1. Teacher first has to add a consultation hour, consultation date must not be lesser than systems date, the consultation starting and ending hours must be between 9.00 A.M and 17.00 P.M, and the consultation ending hour must be greater than consultation starting hour.
   - 2.2. Teacher sees list of students enrolled in his course by each section
   - 2.3. Teacher in order to requests a student to meet him, he first adds consultation hour for that day and request the student with the purpose of meeting, Teacher can also delete this request
   - 2.4. Teacher sees the list of student who had requested for an appointment

2.5. Teacher can accept a student’s request for appointment

3. Student

3.1. Student on his login, can view, edit his profile, change the default password

3.2. Student can add courses or delete courses

3.3. Student views the list of teachers with the respective courses they handle

3.4. Student can request a teacher for appointment provided that teacher has listed the consultation hours

3.5. Student can see the request given by teacher and also the status of the appointment request he has given to a particular teacher

2.3 Functional Requirements

2.3.1 Performance Requirements

In order to achieve the best performance requirement which includes reducing the time and accessibility time of the system to process an appointment, the tasks and memory required for achieving such performance must be as low as possible! The performance of the system can be increased by the way of: Avoiding redundancy of codes which in-turn increases the time of operation and memory space of the system. We believe that the codes are as simple as possible thereby reducing the time and space complexity.

2.3.2 Process Requirements

These are the functional requirements that the system must include to satisfy the system need and to be acceptable by the user. In this prototype, we have added functionalities that we believe will satisfy the actors of the system

- The system allows admin to add, modify/delete and assign departments and courses to teacher
- The system allows admin to add, modify/delete and assign department to teacher
- The system allows teacher to set his consultation hours
- The system allows teacher to change his consultation hours
- The system allows teacher to view the appointments requested to him by the student and also lets him view the list of students he has requested to meet during consultation hours
- The system allows teacher to accept the request given by student
- The system allows respective teacher to delete any particular appointment
- The system allows student to edit his profile, enroll in a particular course

- The system allows student to view the schedule i.e. consultation hours of teachers
- The system allows student to request a particular teacher to give appointment

2.4 Non Functional requirements

Other than functional requirements, the non-functional requirement also plays a very important role in satisfying the actors and the systems full-fledged implementation in real-time, we believe our system is reliable, secure, easy to use, available 24x7, supports data recoverability as it will use centralized data base system if implemented in real time.

2.5 System Requirement specifications

2.5.1 Use case Diagrams:

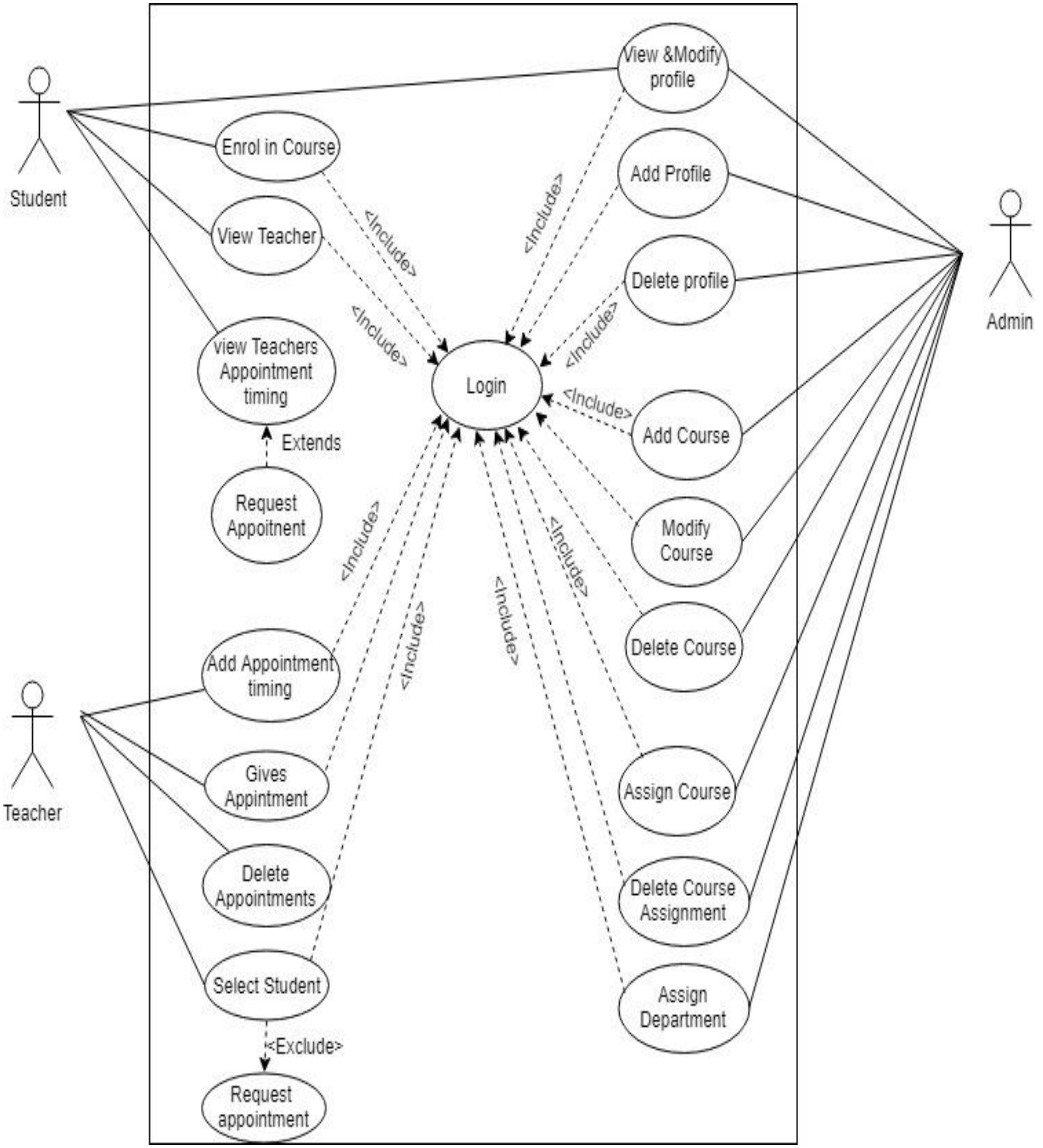

Figure 1 use case diagram

3 SYSTEM DESIGN

3.1 Class Diagrams

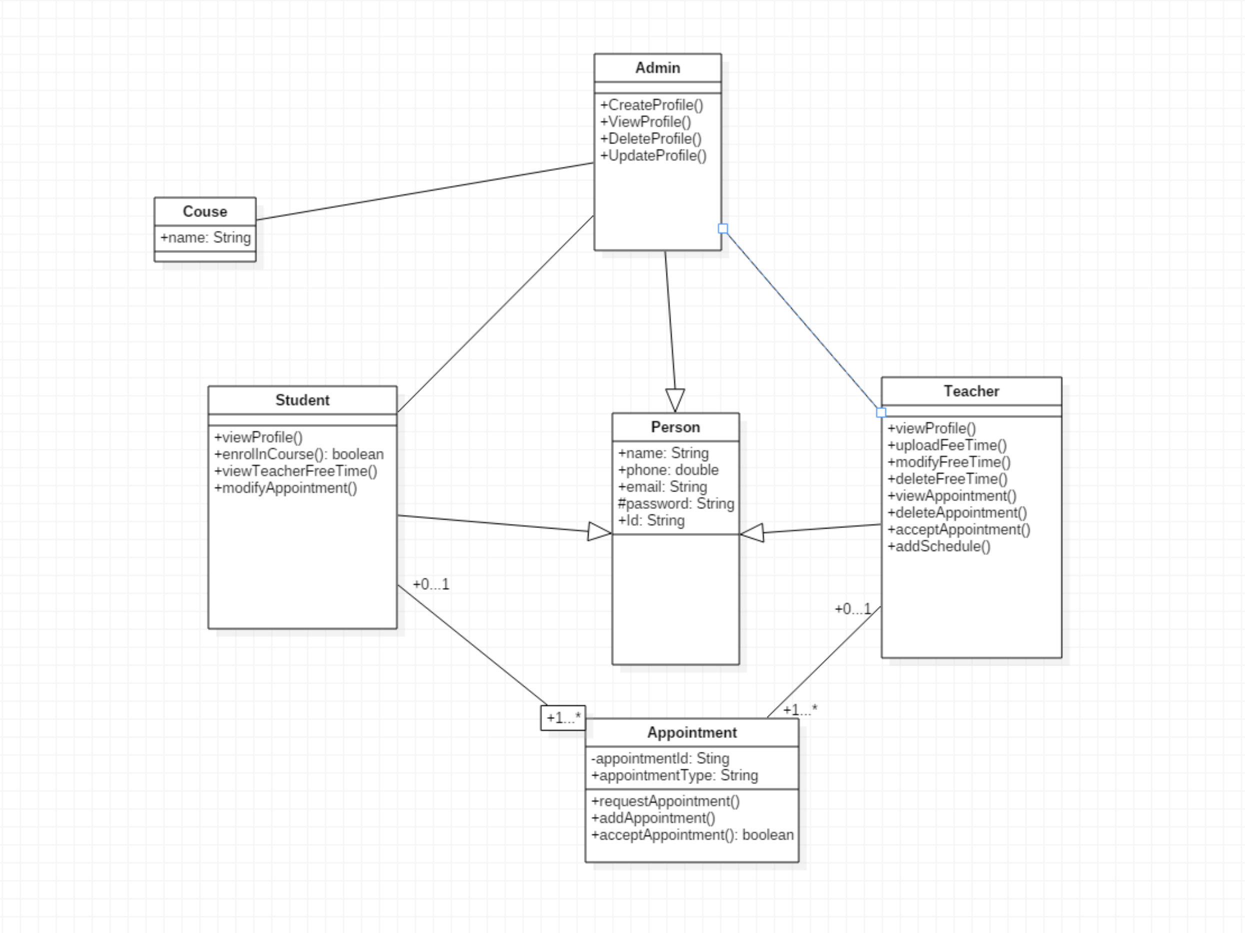

Figure 2 Class Diagram

4. CONCLUSION

4.1 Limitation of the proposed project

The project is limited by many functionalities which we weren’t able to add due to time limitations, few of the limitation of the proposed prototype are

- We have developed a system where teacher is supposed to enter only one free schedule (consultation hour)! Teacher is not provided the facility of entering many different consultation hours in bulk for a couple of days or for whole week or for whole month of his choice
- There is no provision to see the statistical report as how many hours of consultation did a teacher have and how many students he has consulted
- The teacher as an actor is not provided the functionality to view or edit his profile (due to time limitations)
- Student is not provided the ability to modify or delete his appointments
- The system is not generating random password for teachers and students while admin is creating accounts for them and together
- Forgot password implementation not implemented

4.2 Future enhancement

- All the limitations mentioned can be overthrown in the future enhancement
- Further, The Teacher student appointment system can be extended by incorporating appointment between teachers of different department.
- Giving appointment for a particular time slot can be added (Example scenario, if a teacher has consultation hours of 2hrs (between 9.00A.M and 11.00AM), then student can request for appointment for particular time within the consultation hour (9.30A.M to 9.40A.M), similarly teacher can also request a particular student to come at a particular time slot within the consultation hour)
- Modifying the appointment and managing time slots can be further added.