SOFTWARE REQUIREMENTS SPECIFICATION

Version 3.1

April 3, 2013

mini

Schedule Auto-Generator system

Fady Aziz
Nesma Talaat Abbas
Kiro Magdy
Maram El-Sayed
Yaser Abdelaziz

Submitted in partial fulfillment Of the requirements of CS 251 Software Engineering -1

Contents

List of Figures

1 Introduction

1.1 Purpose

Making a class schedule is one of those NP hard problems. The problem can be solved using a heuristic search algorithm to find the optimal solution, but it only works for simple cases. For more complex inputs and requirements, finding a considerably good solution can take a while, or it may be impossible. This is where genetic algorithms come in to the game. This software web system implements genetic algorithms to achieve the optimal scheduale for college semsters.

1.2 Scope of Project

This software system should generate all the optimal solutions for a college depending on it's resources, The system designed to minimize conflicts between rooms, professor place and student group place, also it makes sure that number of seats at the room sufficient for number of students at each group. The system provides a control panel for the users with different tools that meet the users needs while remaining easy to understand and use.

More specifically, this system is designed to allow an admin to generate the scheduale or configure it manually, and he is the only person can full control the web site, add courses, add professors, add rooms, add students and delet them all. other users can view scheduale and interact with it simply.

Also the system fully intergrated with a forum for techincal and non-technical support and professor announcements. And also integrated with vote system to allow admin to check users opnion before puplushing new scheduale.

1.3 Glossary

Term	**Definition**
SAG	Scheduale Auto-Generator.
CMS	Content management system.
Database	Collection of all the information monitored by this system.
Genetic Algorithm	Check Appendix B.
Stakeholder	Any person with an interest in the project who is not a developer.

1.4 References

- IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
- Webopedia: Database. Retrieved 2 April, 2013, Available from the community site, `http://www.webopedia.com/TERM/D/database.html`
- Wikipedia: Entity relationship model, Retrieved 2 April, 2013, Available from the community site, `http://en.wikipedia.org/wiki/Entity-relationship_model`
- Wikipedia: Database, Retrieved 2 April, 2013, Available from the community site, `http://en.wikipedia.org/wiki/Database`

1.5 Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2 Overall Description

2.1 System Environment

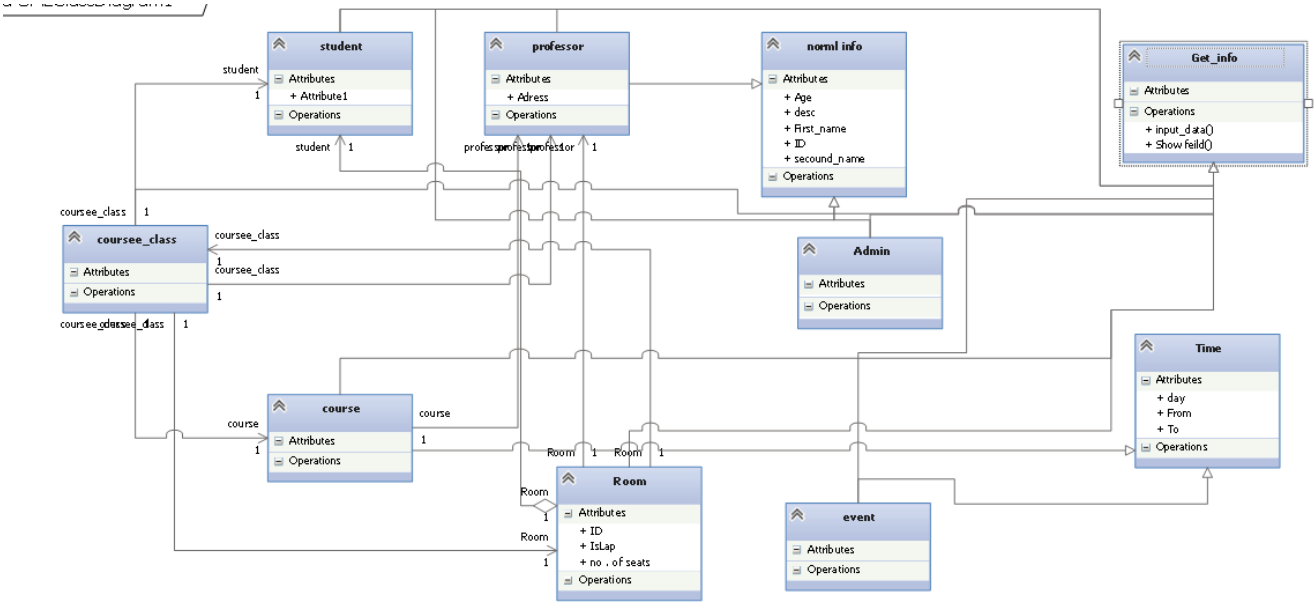


Figure 2.1: Class Diagram of System Environment.

The Scheduale Auto-Generator system has four active actors and one cooperating system. The Admin, Professor, Student or Guest user access mini through the Internet. Admin, Professor and Student accesses the entire system directly after passing the login phase successfully.

2.2 Functional Requirements Specification

This section outlines the use cases for each of the active actors separately. The Admin and the Student have three use cases apiece while the Professor has two use cases, All are main actors in this system.

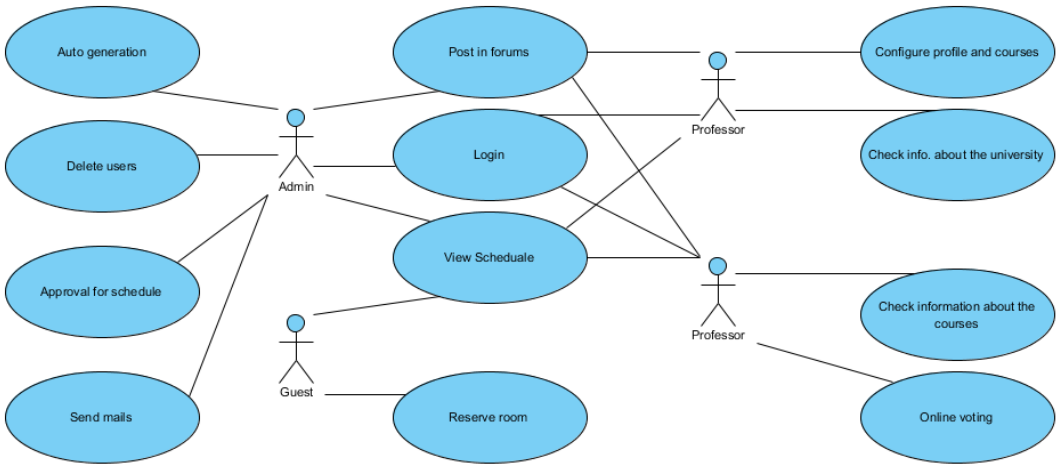
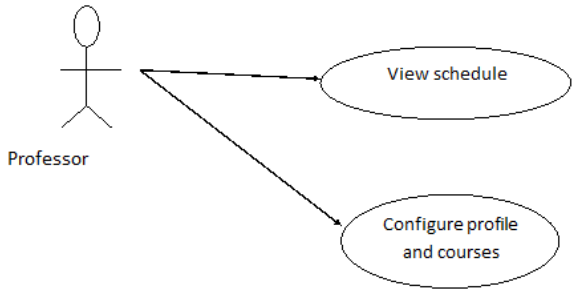


Figure 2.2: Use Cases Diagram.

2.2.1 Professor Use Cases

Use Case: View schedule and configure profile and courses

Diagram:



Brief Description:

The professor can see his schedule and also can configure his profile and change courses or delete as he want.

Initial Step-By-Step Description:

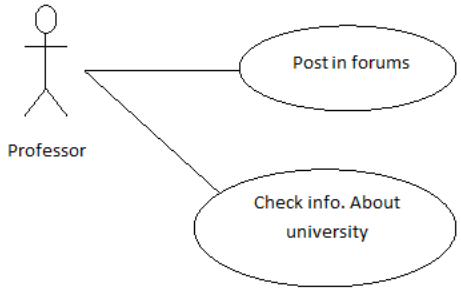
1. The doctor login to the website.
2. Doctor press the button to go to the schedule.
3. The doctor also configure his profile whether information or pictures etc...

4. the doctor configure his courses by changing them or add the description of the syllabus.

Xref: Section 3.2.2, View schedule and configure profile, courses

Use Case: Access and post in forums and check information about the university

Diagram:



Brief Description:

The professor can access the forum and see posts he also can know more information about the university.

Initial Step-By-Step Description:

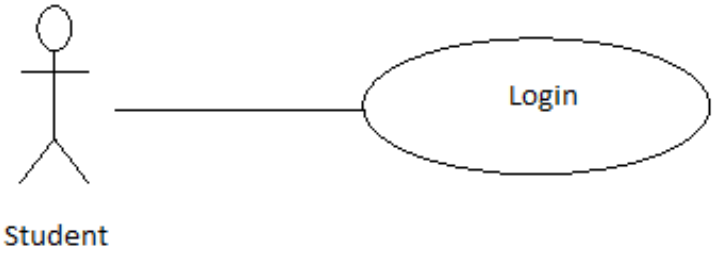
1. The professor can access the forum.
2. The professor can also post in forums.
3. The professor can check information about the university.

Xref: Section 3.2.4, Access information about the university

2.2.2 Student Use Cases

Use Case: Login

Diagram:



Brief Description:
The student login the website.

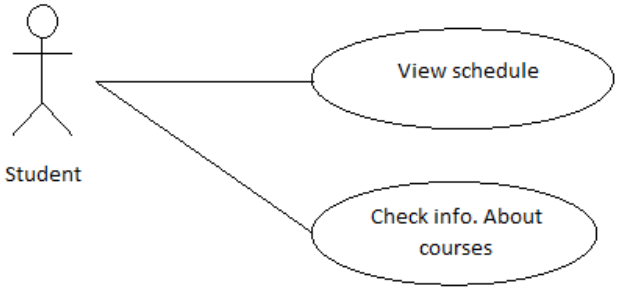
Initial Step-By-Step Description:

1. The student login with his ID and password.
2. The student also can edit on his profile.
3. The student configure his courses.
4. The student also request removal from the system.

Xref: Section 3.2.1, Login and configuration

Use Case: View schedule and check information about the courses

Diagram:



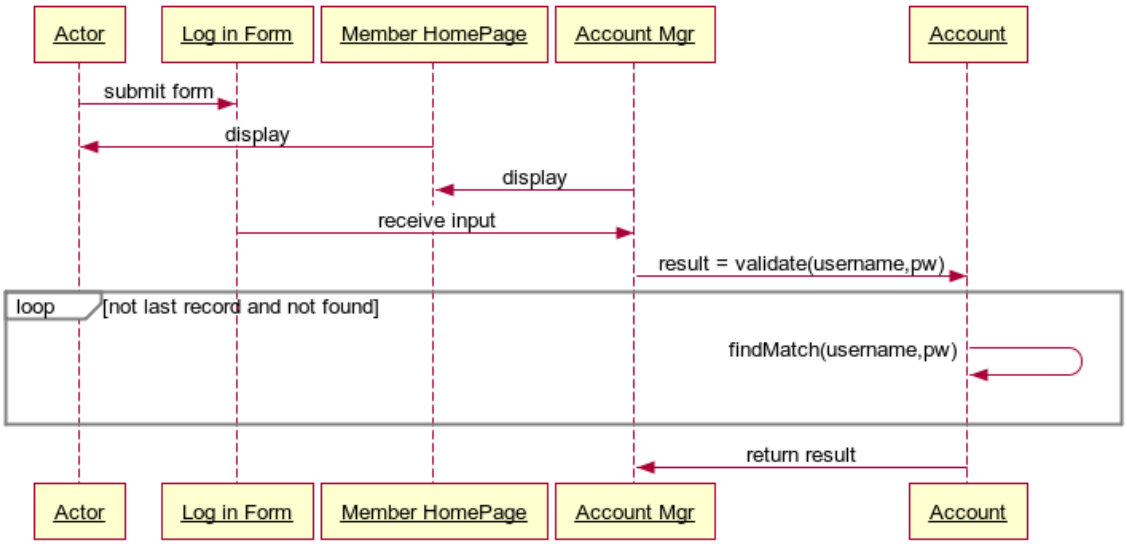


Figure 2.3: Sequence Diagram of login process.

Brief Description:

The student accesses the Online Website, to know the schedule.

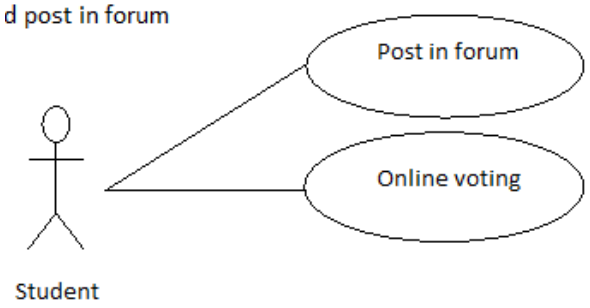
Initial Step-By-Step Description:

1. The student login with his ID and password.
2. The student press view schedule button.
3. The student chooses to see what schedule level.
4. The website offers The student with the schedule he wants to see.
5. The student can check information about university.
6. The student also can know more about the syllabus.

Xref: Section 3.2.3, View schedule and check information about the courses

Use Case: Online voting and post in forum

Diagram:



Brief Description:

The student has the privilege to vote for the schedules to help system choose the most effective schedule.

Initial Step-By-Step Description:

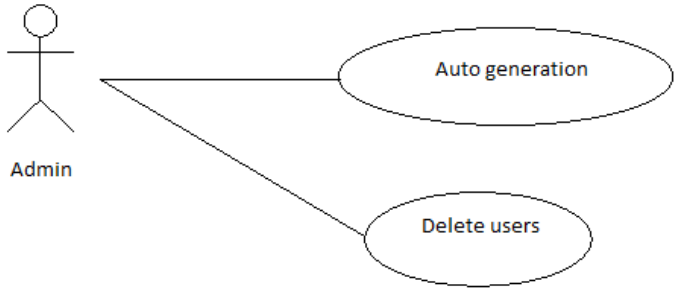
1. There is an online voting for the schedule version to choose the best one.
2. There is a forum where the student can see it.
3. The student has the privilege to post in the forum.
4. The online voting process helps the student to more comfortable with the schedule.

Xref: Section 3.2.5, Vote and post in forum

2.2.3 Admin Use Cases

Use Case: Auto generation and delete users

Diagram:



Brief Description:
The admin make auto generation for the schedule and he can also delete users from the data base.

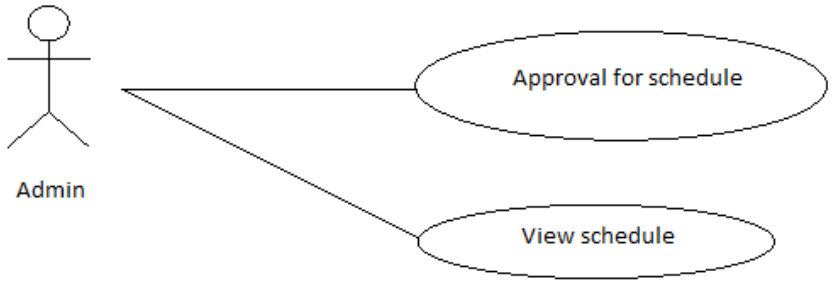
Initial Step-By-Step Description:

1. When a user request removal the admin make sure first the user identity.
2. the admin waits a week if the request don't deleted he will remove the user from the database.
3. the admin main role is to generate the schedule automatic.
4. the admin can also make manual generation for the table.

Xref: Section 3.2.6, Auto generation and delete users

Use Case: View schedule and approval for schedule

Diagram:



Brief Description:
The admin can view schedule and he must approve schedule before publishing.

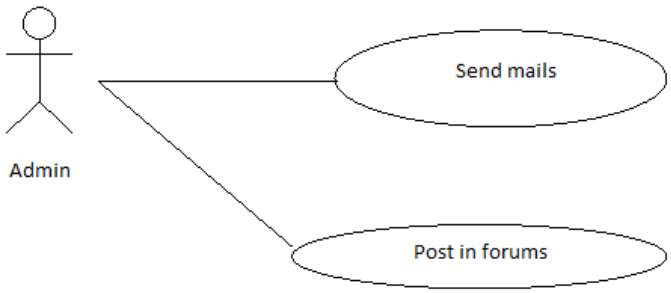
Initial Step-By-Step Description:

1. The admin can see the schedule.
2. The admin confirm the schedule before voting.
3. The admin approve the schedule also after voting to check every thing.

Xref: Section 3.2.7, View schedule and approval for schedule

Use Case: Send schedule to mail and post in forum

Diagram:



Brief Description:
The admin can send mails including new schedule in mails.

Initial Step-By-Step Description:

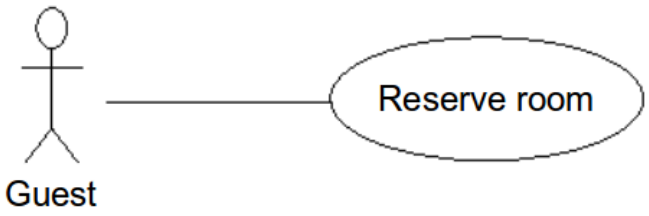
1. the admin have the privilege to send mails to students including schedule.
2. the admin also post in forums.
3. the admin delete posts that aren't suitable.
4. the admin also send the schedules on mobiles.

Xref: Section 3.2.8, Send schedule to mail and post in forum

2.2.4 Guest Use Cases

Use Case: Reserve room

Diagram:



Brief Description:
The guest can reserve free room.

Initial Step-By-Step Description:

1. The guest view scheduale page.
2. The guest choose room.
3. The guest choose time slot.
4. After reserving, admin should accept the request.

Xref: Section 3.2.9, Reserve room

2.3 User Characteristics

The student is expected to be Internet literate and able to fill simple forms and use a search engine.

The professor is expected to be Internet literate and familiar with sending and receiving emails with attachments to interact with students. He should also be able to write news posts in case of cancelling a lecture or at least stay in contact with the admin to do this for him whether by news posts or mass emails.

The Admin is expected to be highly internet literate with a preferable background on some coding techniques in case any error occurs, however such knowledge is not obligatory as he will have an admin panel.

2.4 Non-Functional Requirements

The Schedule auto-generator shall be available for use 99 percent of the time between 6 am and midnight. Generation of the schedule shall be no sooner than 10 seconds after pressing Generate button. The site server shall cater for 300 simultaneous users within the period from 6:00 PM to 12:00 PM. Maximum loading at other periods will be 200 simultaneous users. The SAG system shall be scalable to accommodate unrestricted growth in the number of users. SAG system database could be backed up and restored as soon as needed. Installation shall be able to done by an untrained user without recourse to separately printed instructions. The SAG system website shall run on PC, Ipad, Iphone browsers with different operation systems.

3 Requirements Specification

3.1 Functional Requirements

The Logical Structure of the Data is contained in Section 3.3.1.

3.1.1 Login

Use Case Name	**Login and configuration**
XRef	Section 2.2.2, Login
Trigger	The student login the website
Precondition	The Web is displayed with grids for login in
Basic Path	1. The student login with his ID and password. 2. The student also can edit on his profile. 3. The student configure his courses. 4. The student also request removal from the system.
Alternative Paths	If student want to login with his Facebook he can do it by enter mail face book and password
Postcondition	The user have an account in the website
Exception Paths	The user must student or professor
Other	

3.1.2 View schedule and configure profile, courses

Use Case Name	**View schedule and configure profile, courses**
XRef	Section 2.2.1, View schedule and configure profile and courses
Trigger	The professor can see his schedule and also can configure his profile and change courses o delete as he want
Precondition	The doctor login to the website and see his profile and schedule
Basic Path)	1. The doctor login to the website 2. The doctor press the button to go to the schedule 3. The doctor also configure his profile whether information or pictures etc... 4. The doctor configure his courses by changing them or add the description of the syllabus
Alternative Paths	
Postcondition	The professor seen schedule and configured profile, courses
Exception Paths	
Other	

3.1.3 View schedule and check information about the courses

Use Case Name	**View schedule and check information about the courses**
XRef	Section 2.2.2, View schedule and check information about the courses
Trigger	The student accesses the Online Website, to know the schedule
Precondition	The student has accessed the schedule and information main
Basic Path)	1. The student login with his ID and password 2. The student press view schedule button 3. The student chooses to see what schedule level 4. The website offers The student with the schedule he wants to seen 5. The student can check information about university 6. The student also can know more about the syllabus
Alternative Paths	Student can login with Facebook mail
Postcondition	The student had seen the schedule
Exception Paths	A disconnected number indicating the type of 'other constructed object'.
Other	Student cannot edit the schedule

3.1.4 Access information about the university

Use Case Name	**Access and post in forums and check information about the university**
XRef	Section 2.2.1, Access and post in forums and check information about the university
Trigger	The professor can access the forum and see posts he also can know more information about the university
Precondition	The PROFESSOR has accessed the INFORMATION Manager main screen
Basic Path)	1. The professor can access the forum 2. The professor can also post in forums 3. The professor can check information about the university
Alternative Paths	Professor cannot edit in student profile
Postcondition	Professor accessed information about the university
Exception Paths	Professor can log in with mail Facebook
Other	Professor cannot edit in student profile

3.1.5 Vote and post in forum

Use Case Name	**post in forum**
XRef	Section 2.2.2, Online voting and post in forum
Trigger	The student has the privilege to vote for the schedules to help system choose the most effective schedule
Precondition	The student has accessed the schedule Manager main screen
Basic Path)	1. There is an online voting for the schedule version to choose the best one 2. There is a forum where the student can see it 3. The student has the privilege to post in the forum 4. The online voting process helps the student to more comfortable with the schedule
Alternative Paths	Student cannot vote twice
Postcondition	The voting had been done
Exception Paths	
Other	

3.1.6 Auto generation and delete users

Use Case Name	**Auto generation and delete users**
XRef	Section 2.2.3, Auto generation and delete users
Trigger	The admin make auto generation for the schedule and he can also delete users from the data base
Precondition	The admin can generate the schedule automatic
Basic Path)	1. When a user request removal the admin make sure first the user identity 2. The admin waits a week if the request don't deleted he will remove the user from the database 3. The admin main role is to generate the schedule automatic 4. The admin can also make manual generation for the table
Alternative Paths	If the request don't deleted the admin will remove the user from the database
Postcondition	The database has been updated. and schedule generated
Exception Paths	
Other	

3.1.7 View schedule and approval for schedule

Use Case Name	View schedule and approval for schedule
XRef	Section 2.2.3, View schedule and approval for schedule
Trigger	The admin can view schedule and he must approve schedule before publishing
Precondition	The Admin has accessed the manager main screen
Basic Path)	1. The admin can see the schedule 2. The admin confirm the schedule before voting 3. The admin approve the schedule also after voting to check every thing
Alternative Paths	
Postcondition	The admin confirm the schedule after voting
Exception Paths	
Other	

3.1.8 Send schedule to mail and post in forum

Use Case Name	Send schedule to mail and post in forum
XRef	Section 2.2.3, Send schedule to mail and post in forum
Trigger	The admin can send mails including new schedule in mails
Precondition	The admin have the privilege to send mails to students including schedule
Basic Path)	1. The admin have the privilege to send mails to students including schedule 2. The admin also post in forums 3. The admin delete posts that aren't suitable 4. The admin also send the schedules on mobiles
Alternative Paths	
Postcondition	Mails sent
Exception Paths	The admin may abandon the operation at any time
Other	

3.1.9 Reserve room

Use Case Name	Reserve room
XRef	Section 2.2.4, Reserve room
Trigger	The Guest selects a free slot to reserve
Precondition	
Basic Path)	1. The guest view scheduale page 2. The guest choose room 3. The guest choose time slot 4. After reserving, admin should accept the request
Alternative Paths	
Postcondition	A request for approval of the slot sent to the admin
Exception Paths	The Guest may abandon the operation at any time
Other	

3.2 Detailed Non-Functional Requirements

3.2.1 Logical Structure of the Data

The logical structure of the data to be stored in the internal Univsersity database is given below.

The data descriptions of each of these data entities is as follows:

Student Entity

Data Item	Type	Description	Comment
ID	Int	ID given by the college(Primary key)	
Name	Text	Name of Student	

Student Group Entity

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Name	Text	Name of student group	
Size	Int	Number of students	

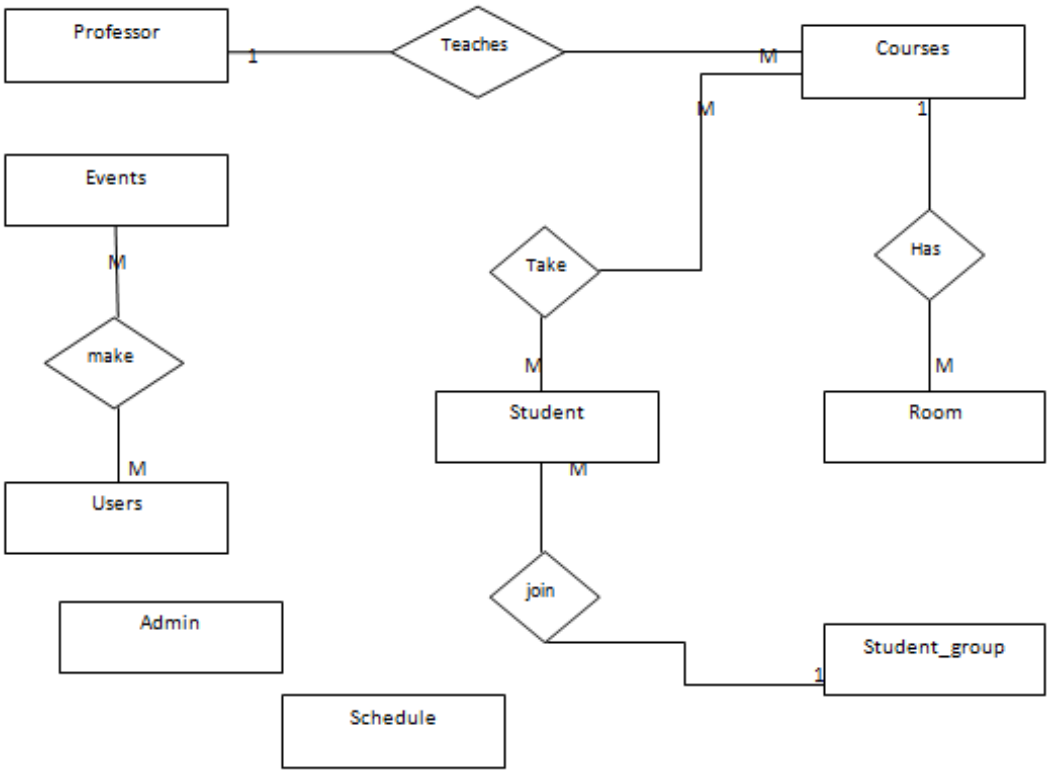


Figure 3.1: Logical Structure of the University Data.

Student Group Entity

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Size	Int	The number of the students in the group	
Level	Int	The level of the students	

Schedule

Data Item	Type	Description	Comment
ID	Int	The ID of the schedule	
Course_ID	Int	Foreign key	
From	Int	The begining time of the courses	
To	Int	The ending time of the course	
Day	Text	The day	

Room

Data Item	Type	Description	Comment
ID	Int	The ID of the room	
Name	Text	The name of the room	
Is_lab	Int	It checks if the lab is lab or room	
No_seats	Int	The number of seats In the room	

Student Group Entity

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Size	Int	The number of the students in the group	
Level	Int	The level of the students	

Courses Entity

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Name	Text	Course names	
Description	Text	Details about courses	

Rooms Entity

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Name	Text	Name of the Room	
Number of seats	Int	Size of room	

Professor Entity

Data Item	Type	Description	Comment
ID	Int	Primary Key	
First_name	Text	The name of professor	
Second _name	Text	The second name of professor	
Username	Text	The name will use to log in	
Hashed_password	Text	Encrypted password	
Description	Text	The details	
Age	Text	The age of professor	
Phone number	Text	The number of professor	

Events

Data Item	Type	Description	Comment
ID	Int	Primary Key	
User_name	Text	Name of user	
Hashed_password	Int	Password of the user	

Events

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Name	Text	Name of event	
From	Int	The start	
To	Int	The finish	
Day	Text	The day of the event	
Content	Text	The details of the event	
desc	Text	The details of description	

News

Data Item	Type	Description	Comment
ID	Int	Primary Key	
Author	Text	The person who publish the new	
Title	Text	The title of the new post	
Content	Text	The details of news	

Course Class Entity

Data Item	Type	Description	Comment
ID	Primary key	Primary key	
Professor_ID	Pointer	Pointer to professor entities	
Course_ID	Pointer	Pointer to course entities	
Room_ID	Pointer	Pointer to room entities	
Student_group_ID	Pointer	Pointer to student group entities	
Duration	Int	Duration of a course	

4 Prototypes

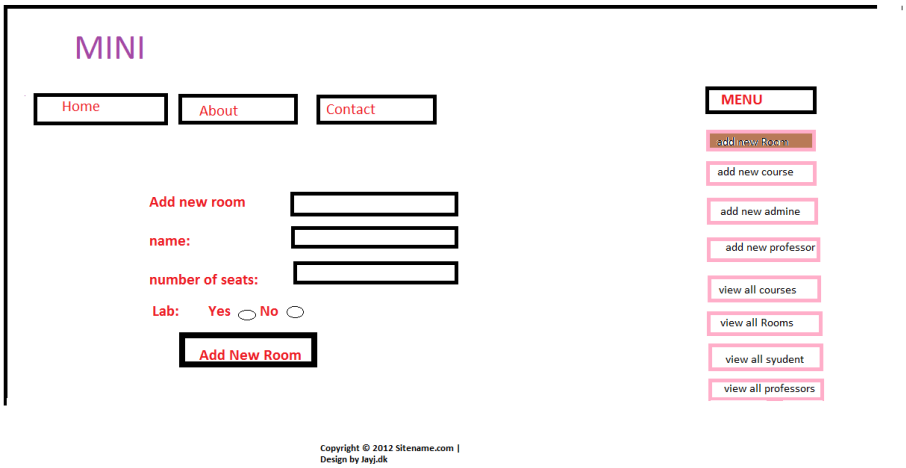


Figure 4.1: Prototype of add room page.

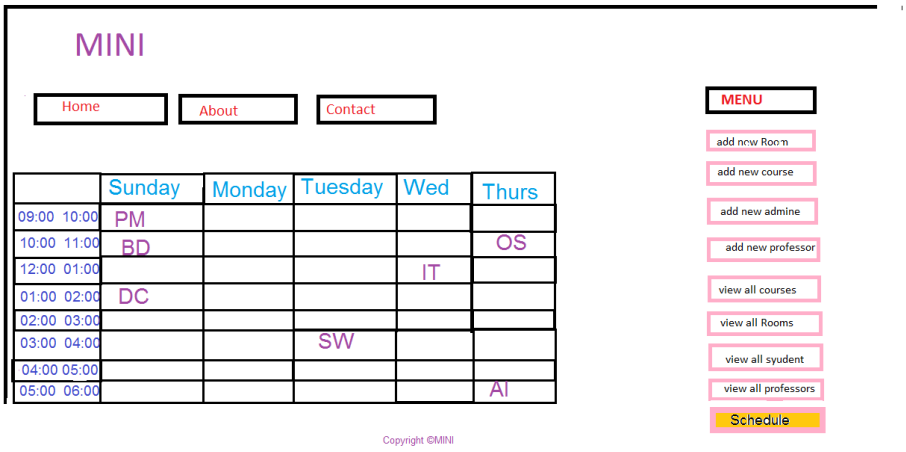


Figure 4.2: Prototype of a schedule.

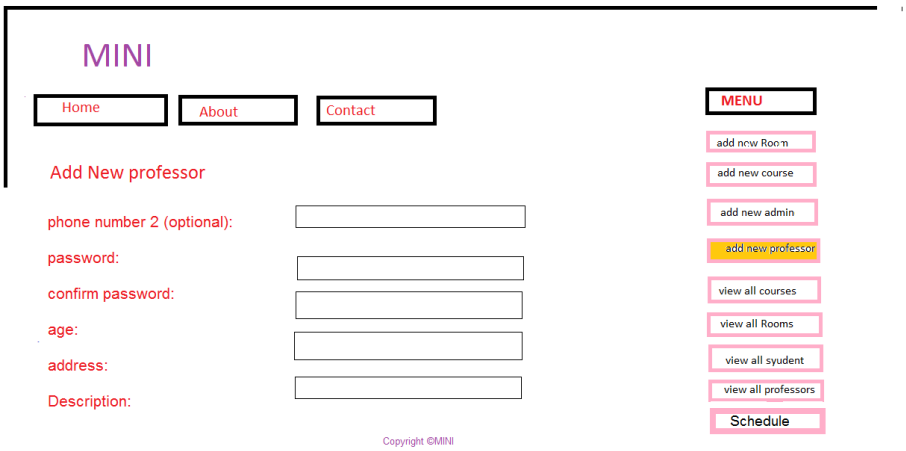


Figure 4.3: Prototype of add new schedule.