****

***Arab American University/Jenin***

***Software Engineering   
Project***

***Smart  Presence and  Absence System  (SPA)***

****

**Students Team “Group one “ :**

***Noor Khzaimiah  211511023***

***Wala’ Abo Alrub  21411060***

***Dr. Rami Hodrob***

**Table of Contents :**

- **list of Contents                                               2**

**- list of tables                                                    3**

**- list of figures**                                                  3

**chapter 1: Introduction**

       1.1Introduction                      5

         1.2Objectives.                      5

**chapter 2: Requirements**

    2.1 Introduction                                                6

2.2  Scenarios                                                   6

2.3 Functional Requirements                               7

2.4 NonFunctional Requirements                         7

2.5  User Requirements                                      8

2.6  System Requirements                                   9

**chapter 3:** **Modelling**                                      10

    3.1 Introduction                                                10

3.2  Use Case of the system                                 10

3.3 Use Case description                                     11

3.4 Sequence Diagram                                         14

3.5 Context Diagram                                          16

3.6  Activity Diagram                                          16

3.7 Class Diagram                                              19

3.8 Detail Architectural Design                             20

3.9  Design                                                        21

3.10 MVC\Codes                                                23

**chapter 4**:**Conclusion and Recommendation**                  26

    3.1 Conclusion                                                26

3.2  Recommendation for future work                26

**List of Tables :**

Table 1 : description of Login By doctor use Case                                   11

Table 2 : description of Manage Time use Case                                       12

        Table 3 : description of opening the device use Case                                12

        Table 4 : description of Calculate percentage of absenteeism use Case         12

        Table 5 : description of Changing the system user use Case                       13

        Table 6 : description of Managing Rules of the system use Case                 13

        Table 7 : description of View Attendance use Case                                   14

        Table 8 : description of View Report  use Case          14

**List of Figures :**

Figure 1 : Scenario                       6

       Figure 2  : Use Case Diagram                                                      11

Figure 3 : Sequence Diagram (1)                                                 15

       Figure 4 : Sequence Diagram (2)                                                 15

       Figure 5 : Sequence Diagram (3)                                                 16

       Figure 6  : Context Diagram                                                        16

Figure 7 : Activity Diagram (1)                                                    17

       Figure 8 : Activity Diagram (2)                                                    18

       Figure 9 : Class Diagram                                                             19

       Figure 10 : Architectural Design                                                   20

Figure 11 : Login Page                                                                21

       Figure 12  : Registration Page                                                       22

Figure 13 : Logout Page                                                               22

       Figure 14  : ASP.NET with C# login Code                                      23

       Figure 15  : ASP.NET with C# login Code                                      24

       Figure 16  : Prove Attendance (Student) Structure Code Java            25

       Figure 17  : System Action (Administration) Structure Code Java       25

**Chapter one: INTRODUCTION**

**1.1 Introduction :**

At the university, the University's laws strongly emphasize the registration of the student's attendance, in order to motivate and force students to attend their lectures to make greater use of the course material, depending largely on their presence.

And to do so they put these laws and regulations to run the educational process in an orderly manner, it gives the student a specified percentage of lectures for absence from the course, if the student exceeded this limit and this ratio, will be deprived of this .

In order to register the student's attendance, the lecturer takes the student's names, checks the number of attendance, and monitors the names of the absent students. This requires time. If the student delays a few minutes from the lecture.

This process is always done during the lecture, and it wastes a lot of time.

In our system (SPA) , our goal is to help the lecture on take students presence in the  lecture in a technical and fast way and do not need the effort or the lost time of the lecture , and the system associated with student in absence way can present an excuse.

**1.2 Objectives :**

* To minimize lost time.
* To help the lecturer monitor the student's attendance.
* Eliminate the possibility of error monitoring attendance.

**\_**     Ensure the right of student

**Chapter Two: REQUIREMENT**

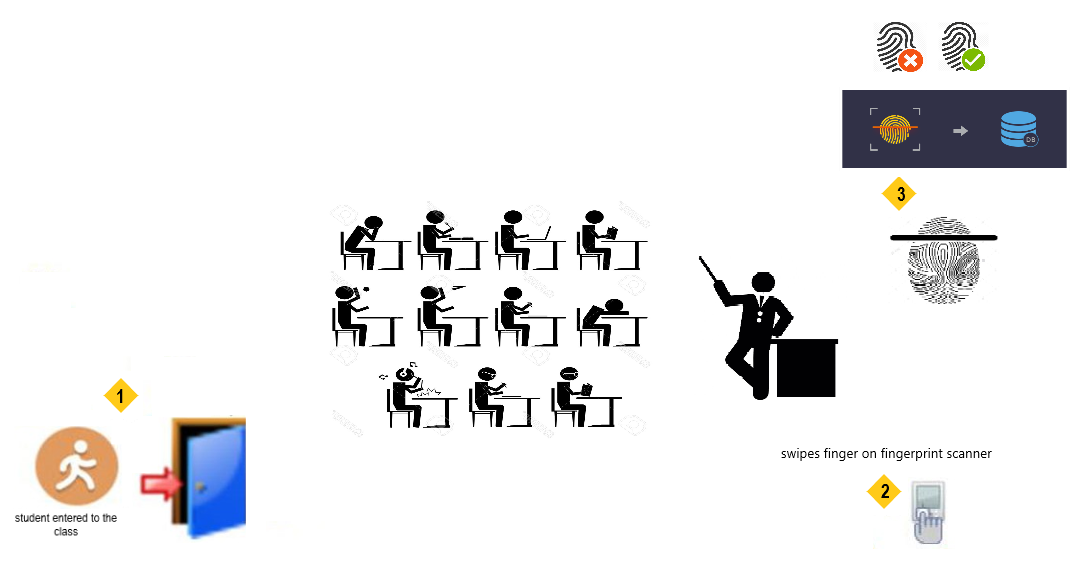
**2.1 Introduction:**

**In this section we describe the requirements of the system .**

**Functional requirements that include user requirements and system            requirements , from functional requirements we define the services, features and requirements of users.**

**Also, we define non functional requirements , that any system without this will be useless system,also non functional requirements affect the functional requirements, so it is important to define it.**

**2.2 Scenarios:**

****

**Figure 1 : Scenario .**

**2.3 Functional Requirement:**

**Functional requirements describe the services that our project provide also the needs that users want from this project.**

**For example , this project shall allow user to prove his presence in the lecture**

**In addition , the system shall allow administrator to manage and control the time of proof student presence .**

**2.4 Non-functional Requirement:**

**Non-functional requirements define the needs in terms of performance , logical database**

**requirements , design constraints , standards compliance , reliability , availability ,**

**security , maintainability and portability.**

**1. Reliability Requirements:**

**Specify the factors required to establish the required reliability of   the**

**software system at time.**

**2. Availability Requirements:**

**The system shall be available during normal working hours**

**(SAT\_WEN,8\_4).**

**Report should be generated automatically every week for doctor and**

**any time upon request.**

**3. Security Requirements :**

**-Doctors will be able to log into the Presence Management**

**System.**

**-Students have access to prove presence .**

**-Doctor have access to subsystem for the define the lecture and time and**

**his name, and access to the various subsystems will be protected by a**

**user login screen that requires a user name and password .**

**-All data must be stored , protected or protectively marked.**

**4. Maintainability Requirements:**

**The Presence and Absence  management system is being developed in**

**java. Java is an  object oriented programming language and shall be**

**easy to maintain.**

**5. Portability Requirements:**

**The Presence and Absence  management system shall run in any**

**Microsoft windows environment that contain java runtime and the**

**Microsoft access database .**

**6. Performance Requirements:**

**-Prove Presence validation should be done within 3 seconds.**

**-Load time of UI should not take more than 2 seconds .**

**-Data in database should be update within 2 seconds .**

**7. Safety Requirements:**

**-Database should backed up every hour .**

**-Under failure , system should be able to come back at normal operation**

**under an hour.**

**8. Capacity requirements:**

**-not more than 50 members to be prove presence .**

**9. Flexibility Requirements :**

**System should be flexible enough to provide space to add new features and**

**to handle them  conveniently .**

**10. Integrity Requirements :**

**System should focus on securing the customer information and avoid data**

**losses as much as possible.**

**11. Efficiency Requirements :**

**Enough resources to implemented to achieve the particular task efficiently**

**without any problem.**

**2.5 User Requirement:**

**1.The user must be able to prove the presence in the lecture.**

**2.The user must be able to sure from that prove his presence in the lecture.**

**3. The administrator must be able to define the time which allow for the**

**student prove his presence in the lecture  .**

**3.1 The administrator must be able to add additional time when the need .**

**3.2 The administrator must be able to add any lecture.**

**3.3 The administrator must be able to add all name student in they lecture.**

**4. The administrator must be able to manage users .**

**4.1 The administrator must be able to frees a user(for ex,withdraw any**

**course or pass the number allowed for the absence)**

**5.The administrator must be able to manage staff.**

**5.1 The administrator must be able to add a staff(doctor) .**

**5.2 The administrator must be able to remove a staff.**

**6. The administrator must be able to log in for start a his lectures .**

**2.6 System Requirement:**

1. **The system shall allow the users to prove presence in the lecture just once time.**
2. **Making Error Request.( request in the event of any error with any user).**
3. **The system shall record the expected checkout date and time.**
4. **The system shall display the rate of his absence from the lecture and the remaining rate of absence.**
5. **The system shall record the expected check-in date and time.**
6. **The system shall allow the customers to confirm of presence.**
7. **The system shall record presence details into database.**
8. **The system shall allow input  to be modified without reenter**

**all the customer information.**

**9. The system shall send a true signal to the portal that meet each user**

**prove the presence or maybe shall send a rate of absence for each user**

**at the end.**

**10.The system shall allow the administrators to enter necessary**

**fields (in the future).**

**11. Managing staff**

**11.1 Add staff**

**11.1.1 The system shall allow administrator to add new user.**

**11.1.2 The system shall allow administrator to open page for new**

**user added.**

**11.1.3 The system shall allow administrator to enter necessary fields**

**and submit .**

**12. Logging in by the administrators.**

**12.1 The system should enable administrators to access system only with**

**username and password.**

**13. Logging in by the user.**

**13.1 The system should enable user to access system only with user**

**fingerprint.**

***Chapter Three: Modeling***

**3.1 Introduction**

**In this chapter we use modeling to describe the system and its**

**services and features .**

**We define the use case to describe the requirements of the**

**actors .**

**We also use activity diagram to describe the activities of actors**

**,we use also the sequence diagram.**

**In addition we define the context diagram to define our project**

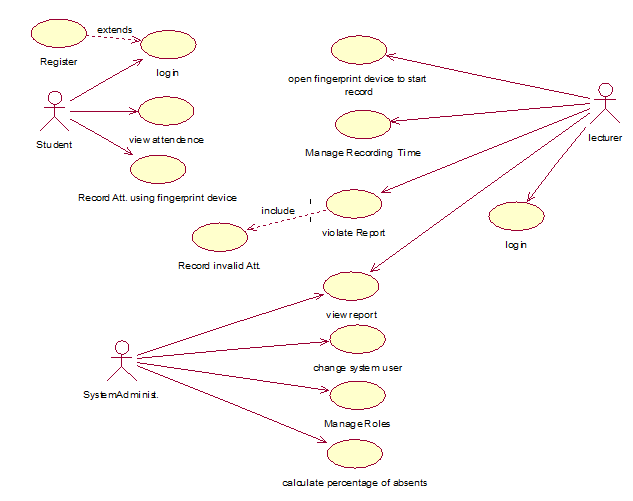
**and its boundaries and the external systems that our system**

**interact with it.**

**Through modeling we define exactly what the system should do**

**and the requirements of stakeholders.**

**3.2 Use Case of system :**

****

**\* Figure 2 : Use Case diagram**

**3.3 Use Case description :**

|  |  |
| --- | --- |
| **SPA: Login by doctor** | |
| **Actors:** | **Lecturer, SPA system** |
| **Type:** | **Primary and essential.** |
| **Description:** | **The lecturer login to system by the correct name and password, then the system show a main menu and lecturer select any choice he need it.** |

**\*Table 1: description of Login By doctor use Case.**

|  |  |
| --- | --- |
| **SPA: Manage recording time** | |
| **Actors:** | **Lecturer, SPA system.** |
| **Type:** | **Primary and essential.** |
| **Description:** | **The lecturer activates the SPA system by logging into the system by the correct name and password, specifying the course that will begin and determining the time period for proving the student's presence. The lecturer can determine the start and end time of each lecture.** |

**\*Table 2: description of Manage Time use Case.**

|  |  |
| --- | --- |
| **SPA: Open fingerprint device** | |
| **Actors:** | **SPA hardware, lecturer** |
| **Type:** | **Primary and essential.** |
| **Description:** | **When the lecturer selects his or her lecture and sets the time to start the student fingerprint, he activates the hardware by pressing the power on button.** |

**\*Table 3: description of opening the device use Case.**

|  |  |
| --- | --- |
| **SPA: Calculate percentage of absenteeism** | |
| **Actors:** | **SPA system, SPA hardware** |
| **Type:** | **Secondary and not essential.** |
| **Description:** | **When the student inserts his fingerprint to prove his presence, the system prove his presence, the system calculates the percentage of absenteeism that this student has missed and the remaining percentage of his absence. The system shows this percentage on the screen hardware for student.** |

**\*Table 4: description of Calculate percentage of absenteeism use Case.**

|  |  |
| --- | --- |
| **SPA: Change system user** | |
| **Actors:** | **Administration and SPA system.** |
| **Type:** | **Primary and essential.** |
| **Description:** | **Administration enter the lectures which will be in this room, time, Dr.  for every lecture and enter the names of students for each lecture and has the authority to modify, add and delete** |

**\*Table 5: description of changing the system user use Case.**

|  |  |
| --- | --- |
| **SPA: Manage rules** | |
| **Actors:** | **Administration and SPA system.** |
| **Type:** | **Primary and essential.** |
| **Description:** | **The Administration shall determine all the laws imposed by the University in respect of attendance and absence, such as the number of absences allowed on Saturdays, Mondays and Wednesdays, and the rate of absence on Sunday and Tuesday, as well as the percentage of absences allowed for laboratories and days when attendance and absence are not taken.** |

**Table 6: description of Managing Rules of the system use Case.**

|  |  |
| --- | --- |
| **SPA: View attendance** | |
| **Actors:** | **SPA system, student** |
| **Type:** | **Secondary and not essential.** |
| **Description:** | **Student enter into system by correct name and password,SPA system show main menu and student select view attendance then SPA system presents the rate of absence , days of absence and time for each lecture.** |

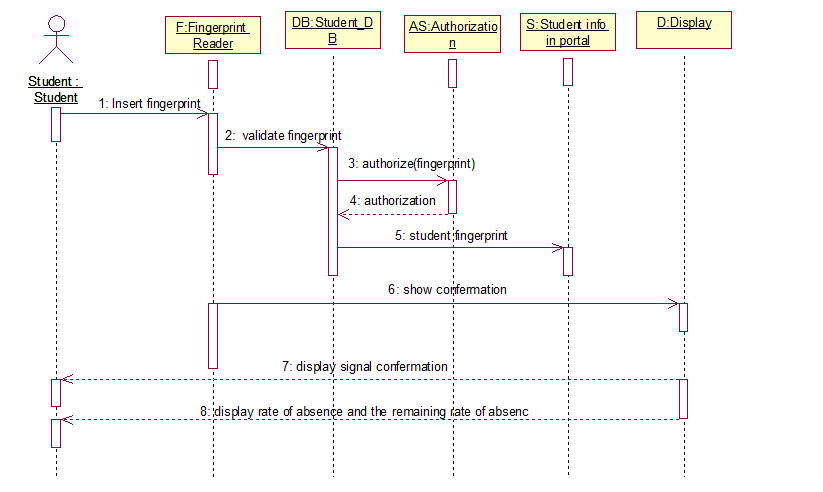
**Table 7 : description of View Attendance use Case .**

|  |  |
| --- | --- |
| **SPA : View report** | |
| **Actors:** | **SPA system, lecture** |
| **Type:** | **Primary and essential.** |
| **Description:** | **The system provides a report to Dr. and administration once a week and at any time the report is requested. Where it contains a full description of the process of proof of attendance and every time the student entered his presence, name, university number, lecture, date and lecturer of each lecture.** |

**Table 8 : description of View Report  use Case .**

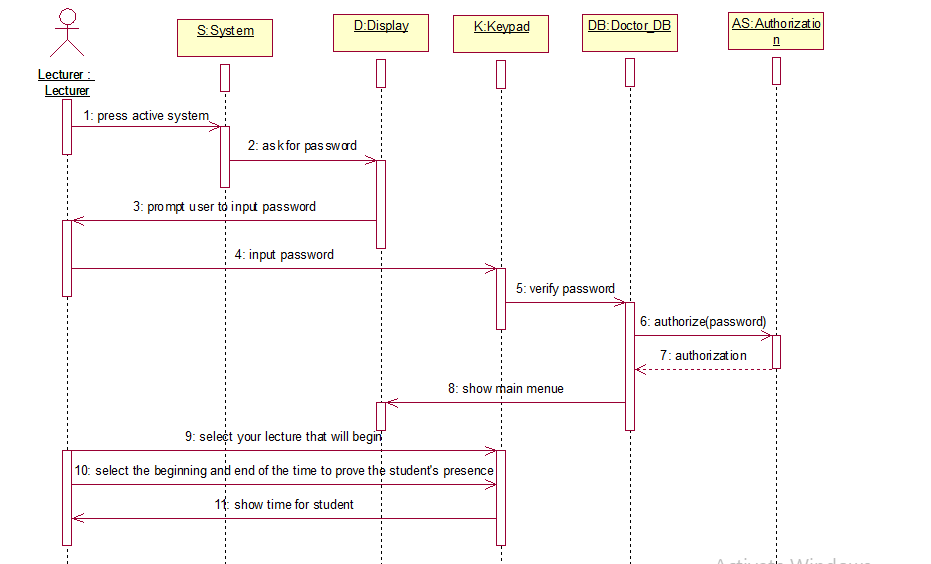
**3.4 Sequence Diagram:**

**Sequence Diagram for student proves presence:**

****

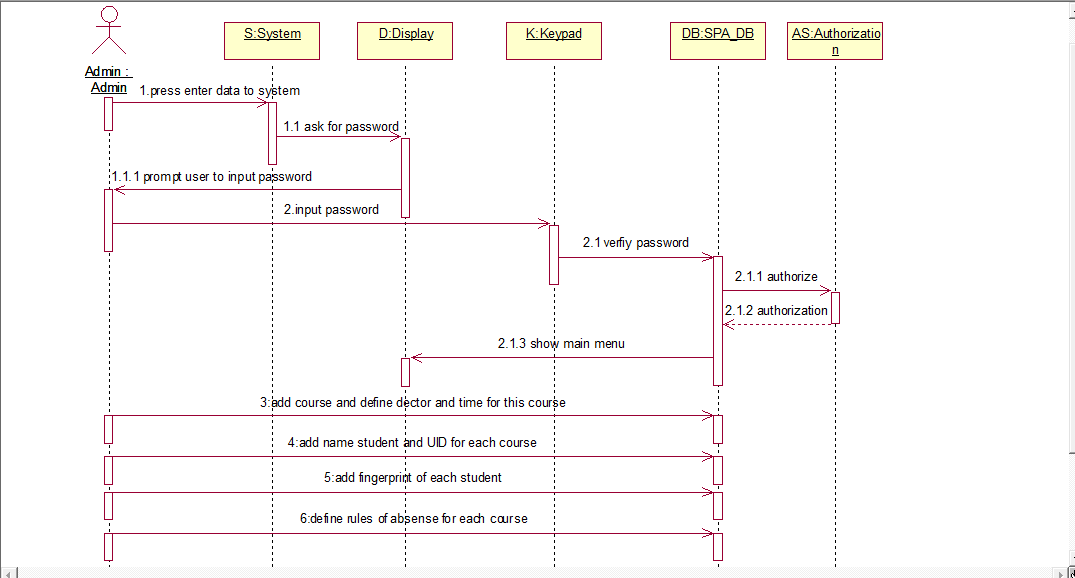
**\* Figure 3: sequence diagram (1)**

**Sequence diagram to activate the system of proof of student attendance by the doctor:**

****

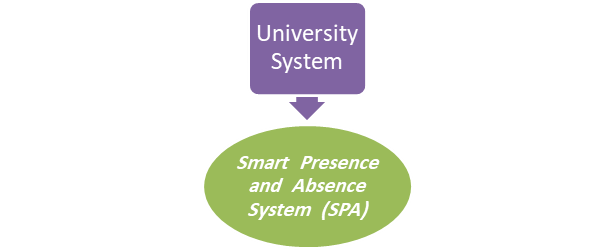
**\* Figure 4: sequence diagram (2)**

**Sequence Diagram for add data into system by administration**

****

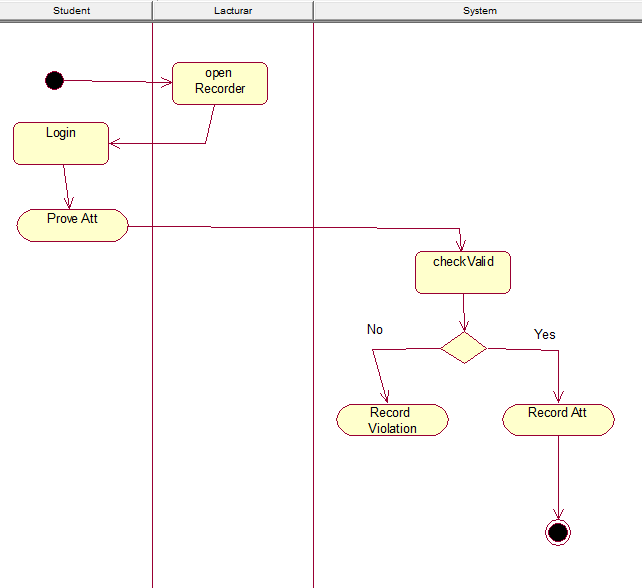
**\* Figure 5: sequence diagram (3)**

**3.5 Context Diagram:**

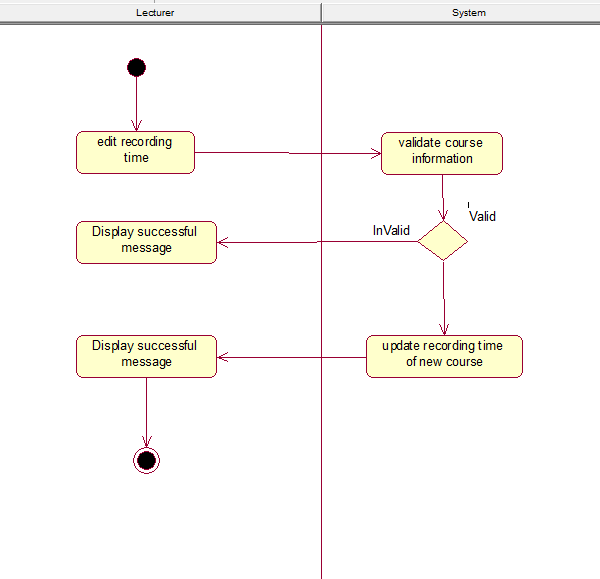
****

**\* Figure 6: Context diagram**

**3.6 Activity Diagram:**

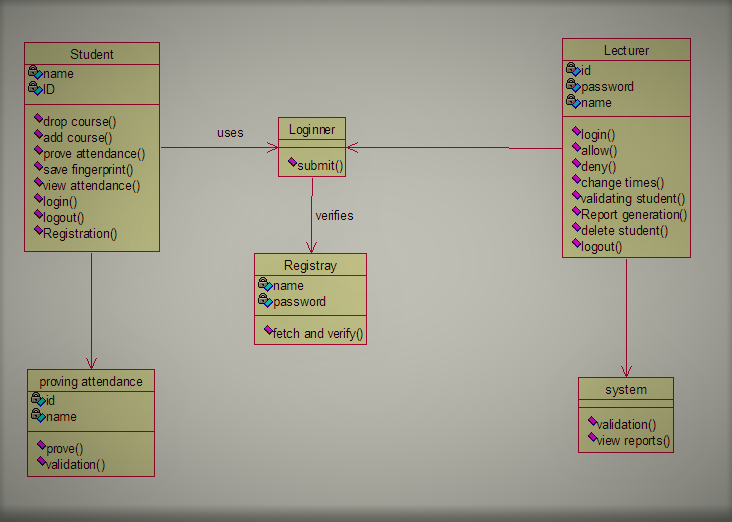


**\* Figure 7: Activity diagram (1)**

****

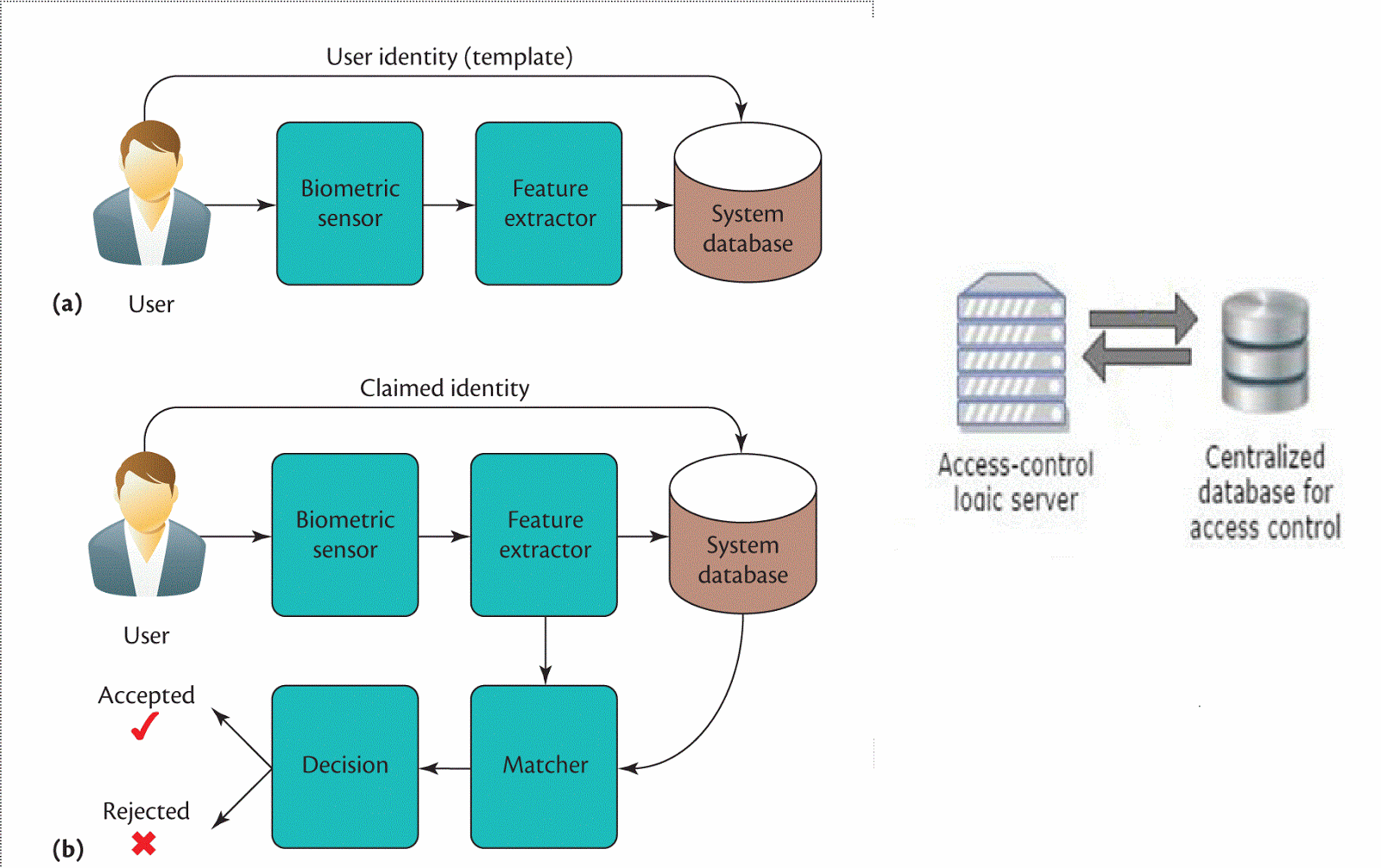
**\* Figure 8: Activity diagram (2)**

**3.7 Class Diagram:**

****

**\* Figure 9: class diagram**

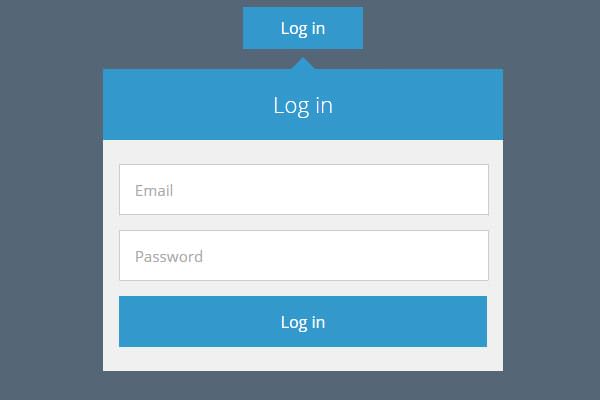
**3.8 Detailed architectural design**



**Figure 10: architectural design**

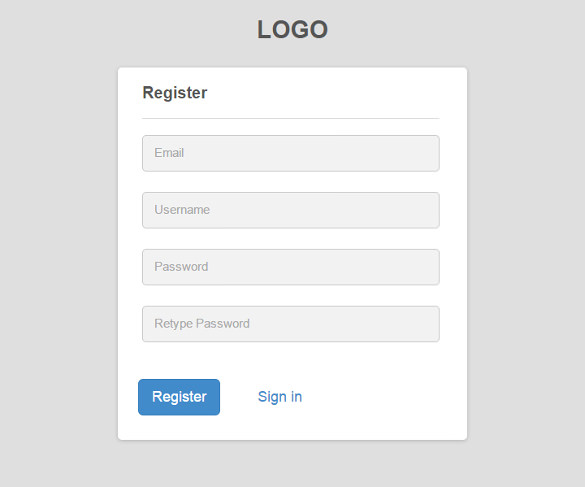
**3.9 Design**

**Login Page**

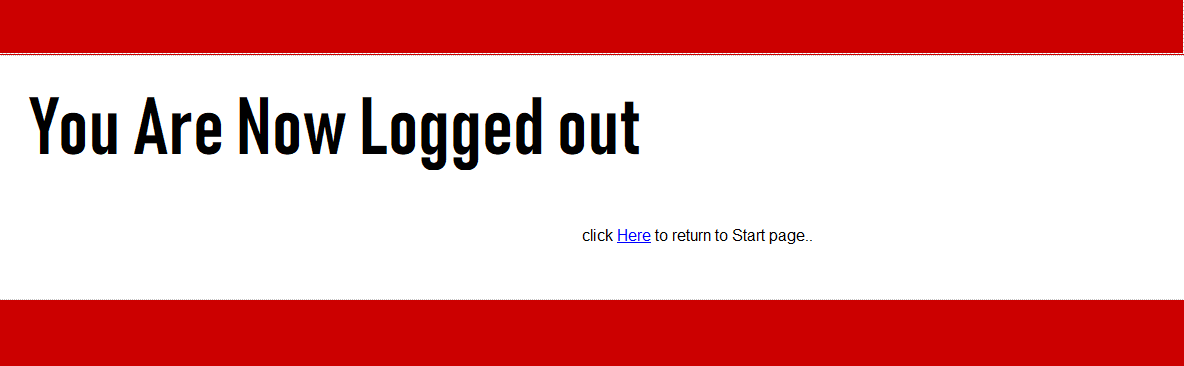
****

**Figure 11: login page**

**Registration Page**



**Figure 12: Registration Page**

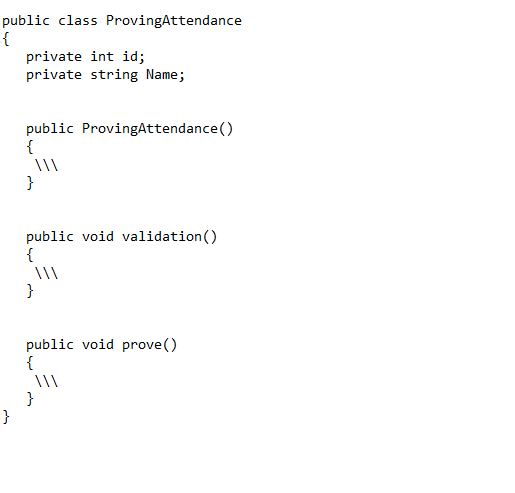
****

**Figure 13: logout Page**

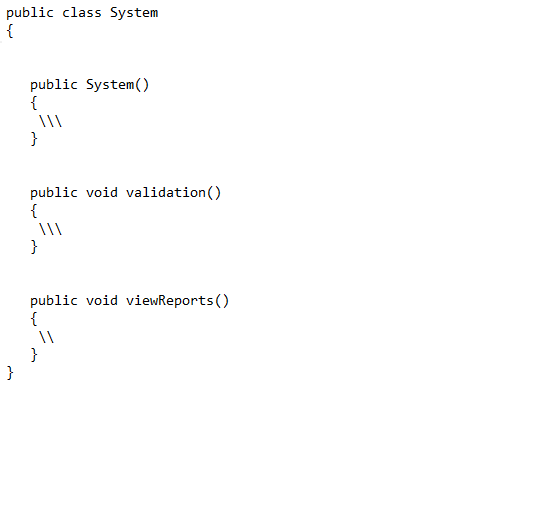
**3.10 MVC \ Codes:**



**Figure 14: ASP.NET with C# login Code.**

****

**Figure 15: Prove Attendance (Student) Structure Code (JAVA).**

****

**Figure 16: System Action (Administration) Structure Code (JAVA).**

***Chapter Four: Conclusion and Recommendations:***

**4.1 Conclusion:**

The Smart  Presence and  Absence System is a solution to help the lecturer on take students presence in the  lecture in a technical and fast way and do not need the effort or the lost time of    the lecture , and the system associated with student in absence way can present an excuse.

**4.2 Recommendations for Future Works:**

We will keep trying to develop the system and add new features and services based on user requests.

In addition to working on trying to expand the scope of his work. And to reduce the possible errors in the system.