
Software Requirements Document for Attendance by Face Recognition System.

Author: TEAM # 7

Inabat Muratova IM

Mukhammedali Shakhan MS

Meiirzhan Aitenova MA

Khanzada Myrzabekova KM

Yersaiyn Kereybayev YK

Version	Date	Author	Change
0.1	28.02.14	Team #7	Initial Document

Table of Contents

1	Introduction.....	3
1.1	Purpose	3
1.2	Scope	3
1.3	Definitions, acronymns, abbreviations	3
1.5	Overview	4
2	Overall Description	5
2.1	Product Perspective	7
2.2	Product functions	7
2.3	User characteristics	14
2.4	Constraints	114
2.5	Assumptions and Dependencies	114
3	Specific Requirements	15
3.1	External Interface Requirements	15
3.2	FEATURES	115
3.3	Performance requirements	118
3.4	Design Constraints.....	119
3.5	Software System Attributes	119

1 Introduction

1.1 PURPOSE

The purpose of this document is to provide detailed requirements information about Face Recognition System

1.2 SCOPE

This document will describe the use cases and features of the Face Recognition System.

1.3 DEFINITIONS, ACRONYMS, ABBREVIATIONS

Term	Description
Admin	Administrates all the content and users of the system, can do everything
Instructor	Manages the system. Modifies, adds/deletes information about student, can view student's information.
Student	User who can register to the system. Student can mark him/herself by taking photo.
List	List of the students
Attendance	Attendance table where list of the students with their attendance marks are filled.
Login	Process by which instructor can access to his profile
Face Recognition	The system which recognizes face
Modify	The process which allows instructor to change, edit data
Registration	The procedure of registering
Group	Collective of students defined by university
Subject	Current course of that period

1.4 REFERENCES

Google API: <https://code.google.com/p/attendanceby-face-recognition/>
<http://docs.opencv.org>

1.5 OVERVIEW

[OMIT]

2 Overall Description

Attendance by Face Recognition System. The main function of this system is to automatically identify or verify individuals in institutions (or universities, colleges, offices and other places) where the presence or attendance is to be controlled. This will help to increase the productivity and reduce extra paper work. Places where our system will be used will save their time.

2.1 PRODUCT PERSPECTIVE

There are several face recognition applications, but for now we don't know such application which is for attendance checking system. As we mentioned previously this app helps to increase the productivity of attendance checking system in the university. Target user is university students and instructors

Concept of Operations

The Face Recognition System Application will run on any computer device which has web-Camera. The recognizing of taken photo is done with Eigenfaces algorithm which is taken from OpenCV library. In order to use this library in java we will use connector library JavaCV.

Major User Interfaces

See appendix A (Screenflow)

See appendix B (screenshots)

Hardware Interfaces

This application requires standard personal computer peripherals with web camera, and also it must be connected to the Internet. The web-site requires standard personal computer peripherals. MySQL database is used for storing data on the server.

Software Interfaces

// example: CGI-URL or function signatures etc (OMIT for now).

Communication Interfaces

// example: modem etc (OMIT for now)

Memory Constraints

// RAM, and other storage constraints (OMIT for now)

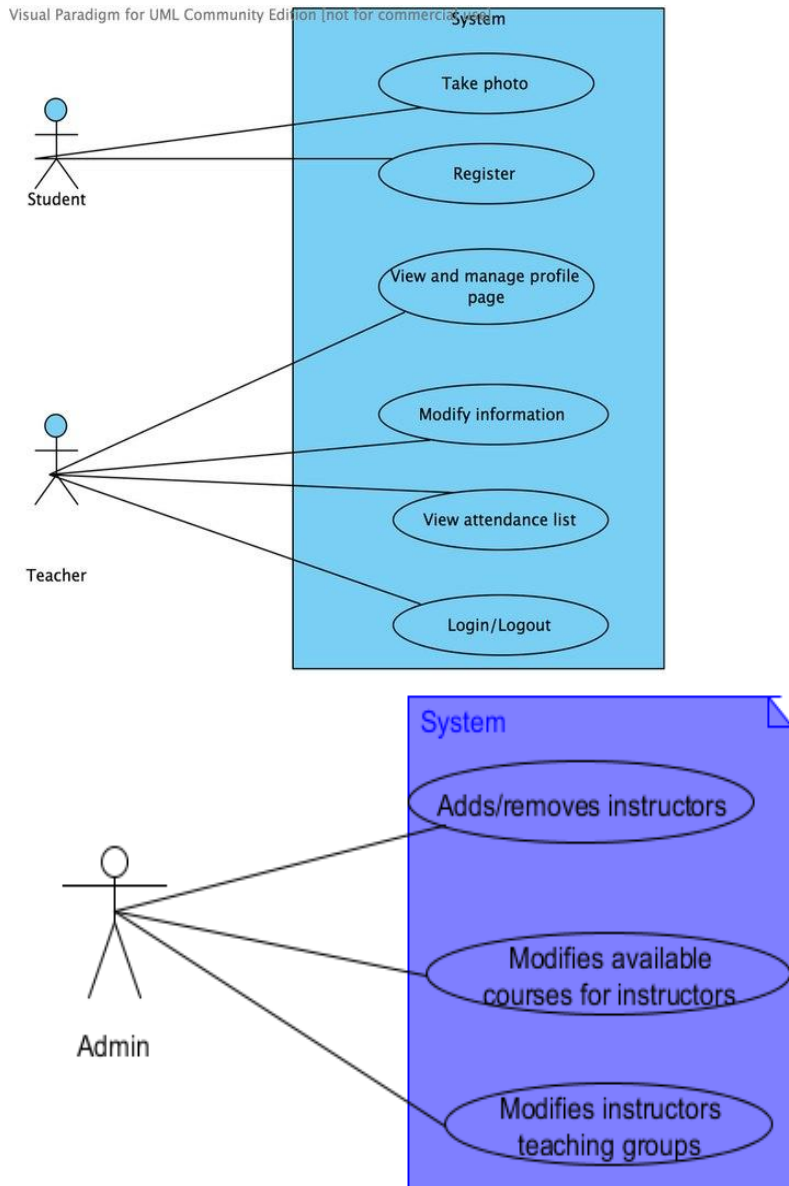
Operations

// special operations (if any) (OMIT for now)

Site Adaptation Requirements

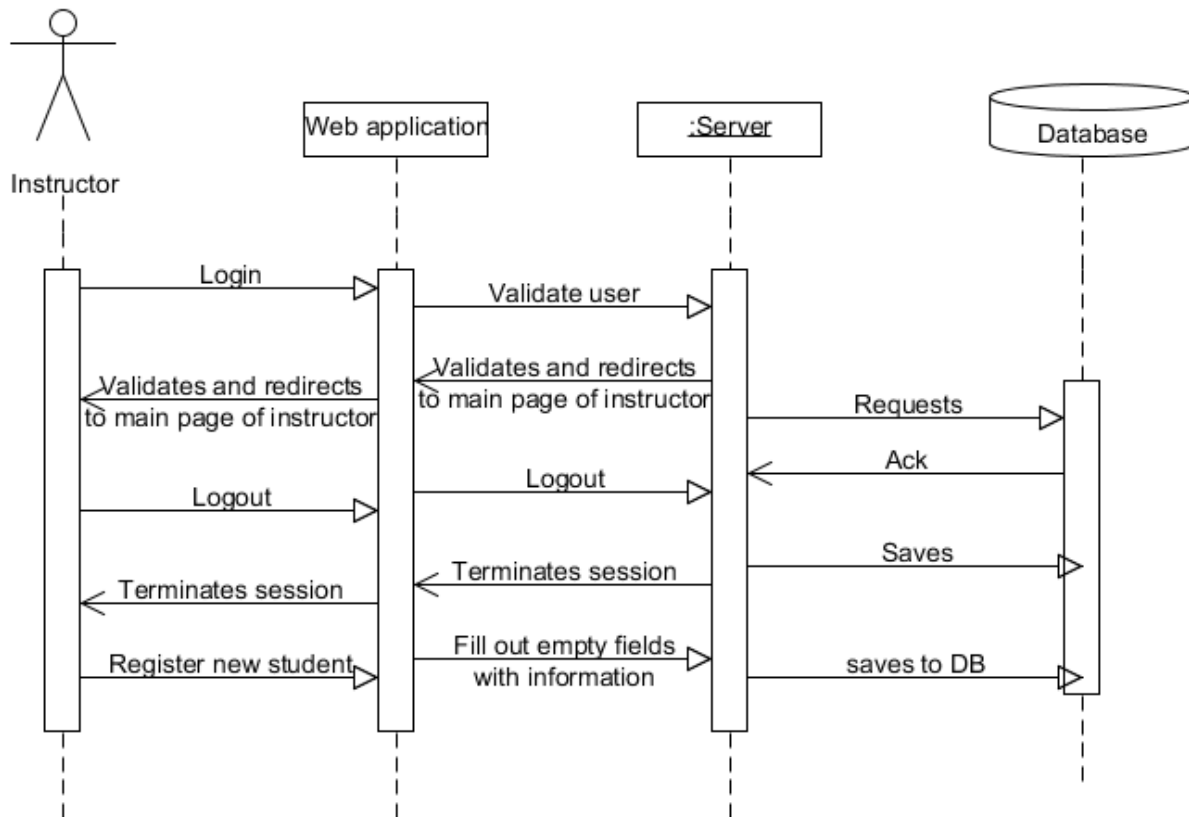
//ex: Japanese language etc (OMIT for now)

2.2 PRODUCT FUNCTIONS



- View and manage profile page
 - View all information about student (id, name, surname, faculty, specialty, group, photo)
 - View presence and absence list of students
 - Can modify information fields (absence and presence list)
-
- **Student:**
 - Registration
 - Take photo for the attendance

Login Page (IM)



2.2.1 Login page

Description: For instructors to login and for students for register themselves to the system.

Actors: Instructor, student

Main flow:

2.2.1.1 This page is opened in the web application.

2.2.1.2 Instructor fills Login/password fields. If authentication is successful, instructors profile page is opened.

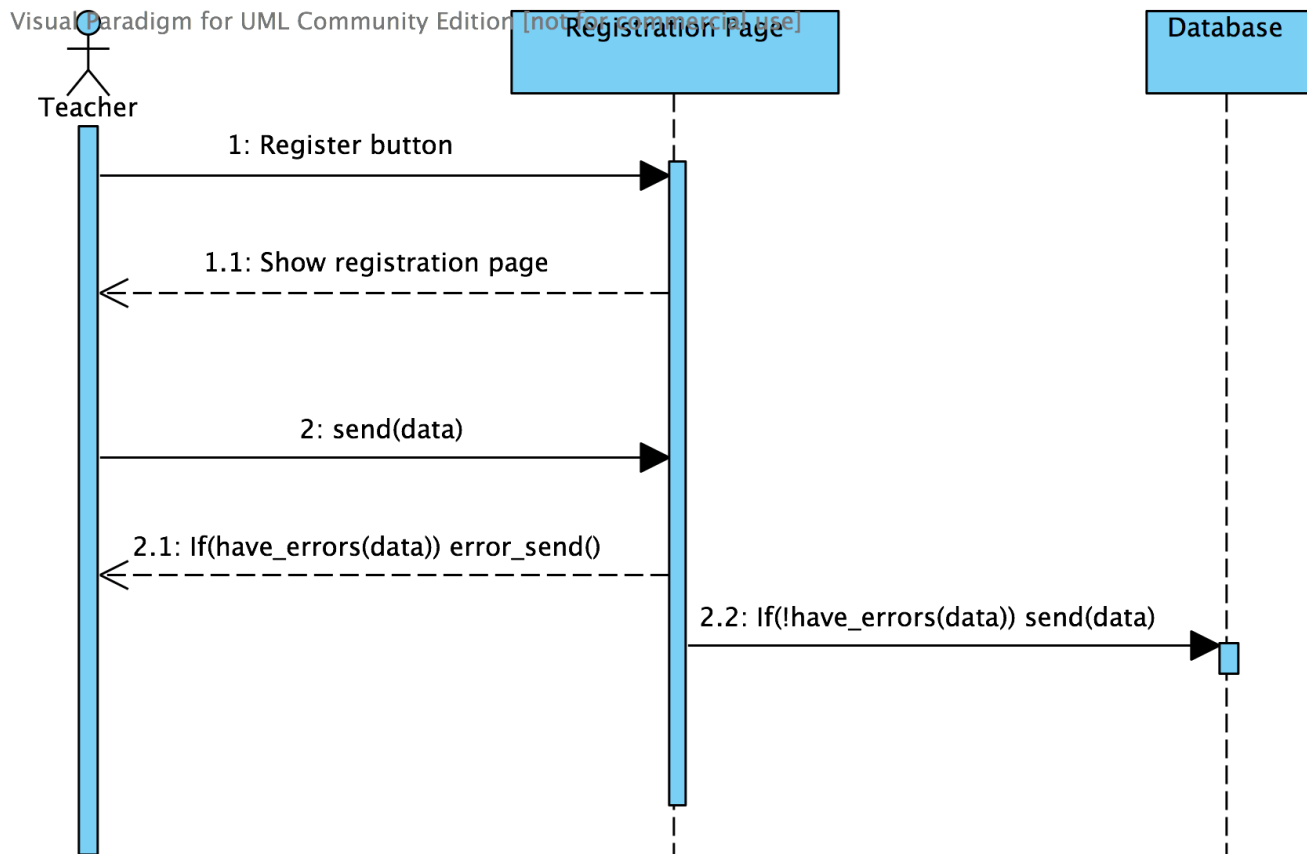
2.2.1.3 Students clicks applies empty fields with needed information (faculty, specialty, id, group, name, surname, photo) and clicks save button. The information about the student are sent to database

Alternate flow:

2.2.1.4 Error can occur if login/password are incorrect, the system will request again to fill login/password correctly.

Preconditions: The browser application is opened with an internet connection

2.2.2 Student registration page (YK)



Description: For students to register to the system so that they could use the face recognition system. General information (id, name, surname, faculty, specialty, group, photo) is added, so that system could recognize the following students by their photo and sign them like “attended” in student attendance list.

Actors: student

Main flow:

2.2.2.1 This page is opened in the web application after student clicks on a register button in the login page.

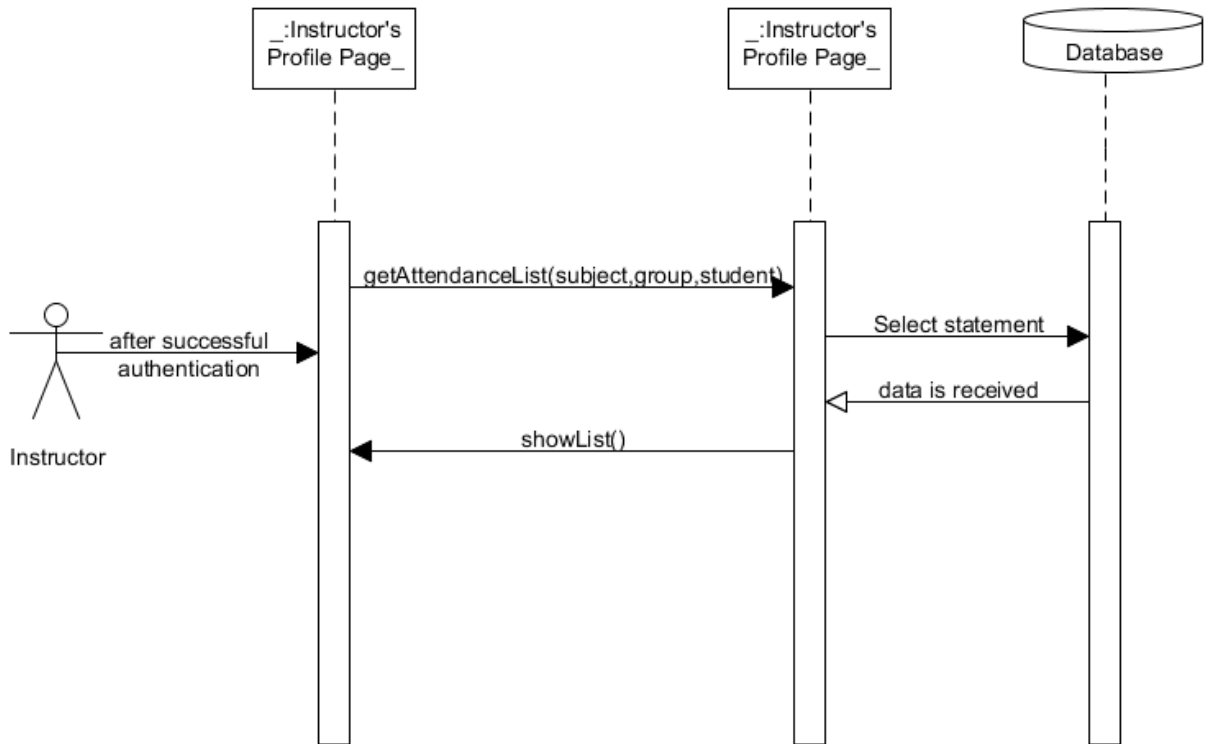
2.2.2.2 Student fills all required fields with his information and clicks “register” button

Alternate flow:

2.2.2.3 Error can occur if id of the following student is same with previously registered students’ id.

Preconditions: “Register ” button must be clicked in the login page

2.2.3 Instructor's profile page(MA)



Description: For instructors to view and get information about students

Actors: Instructor

Main flow:

2.2.3.1 This page is opened after instructors authentication in the login page is successful.

2.2.3.2 After that the instructor has option to select group he can view (for each instructor related groups are viewable. This means the following instructor can view only those groups whom he teaches)

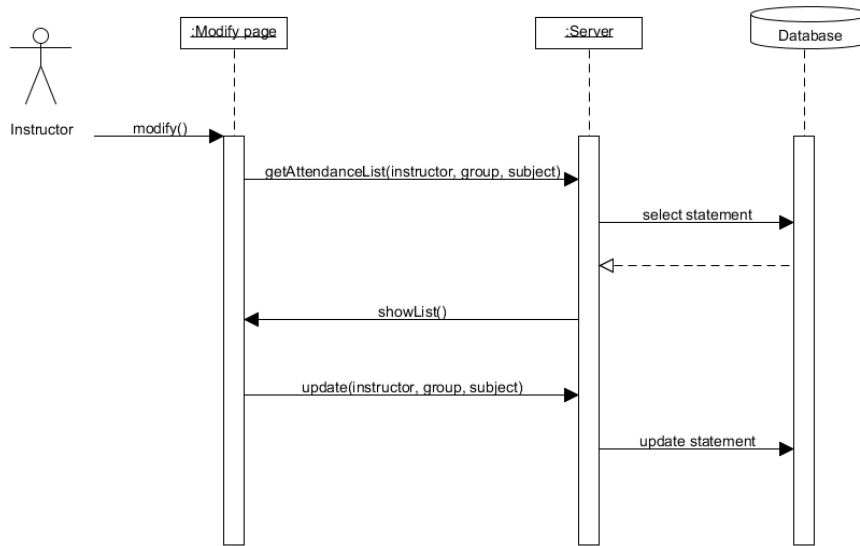
2.2.3.3 After selecting one group, the attendance list table of that group is viewed with date/time.

Alternate flow:

2.2.3.4 Error can occur if login/password are incorrect, the system will request again to fill login/password correctly.

Preconditions: The browser application is opened with an internet connection

2.2.4 Instructor's Modify Page (KM)



Description: Teacher will check attendance list, if student was absent but has permission teacher can edit attendance student list by signing them as attendant.

Actors: Teacher

Main flow:

2.2.4.1. This page is opened after teacher clicks the “modify” button which is in the instructor’s profile page.

2.2.4.2. After that the instructor has options to select groups, students which he wants to edit.

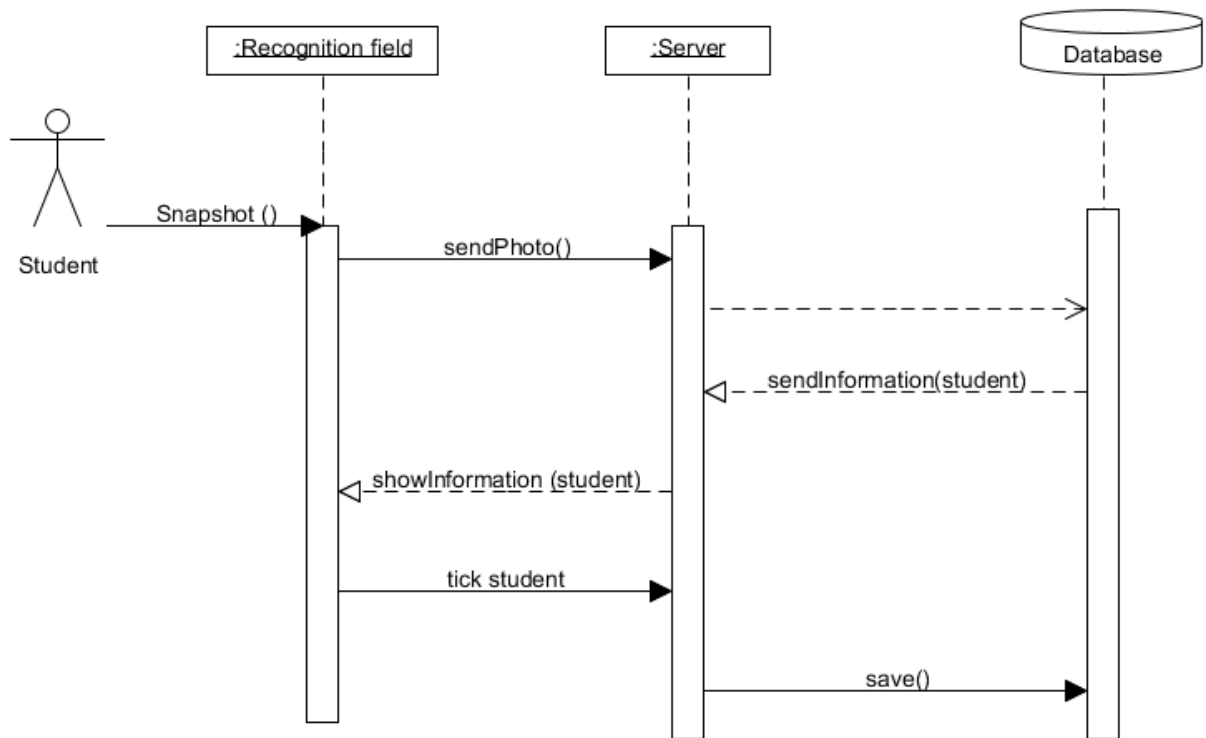
2.2.4.3. After selecting the group attendance list table, following table becomes editable/modifiable. He can add/remove attendance marks and saves.

Alternate flow: If instructor clicked “modify” button after selecting the group he wanted to edit, needed group list will be viewed in the modify page automatically.

Preconditions: Instructor must click on the “modify” button in the instructor’s profile page

Post conditions:

2.2.5 Recognition Page(desktop application) (MS)



Description: Application page where student can take himself to photo for attendance mark.

Actors: Student

Main flow:

2.2.4.1. Camera display and “shot” button is displayed in this application page.

2.2.4.2. Student will take himself to photo. After this process, taken photo is sent to database, if it matches with existing student’s photo in db, attendance mark is added to corresponding student.

Alternate flow: If there’s no matching photo in db error message occurs and system prompts student to try again or to call administrator.

2.3 USER CHARACTERISTICS

Types	Characteristics	Usage frequency
Student	Someone who attends university	High
Instructor	User who checks, modifies the presence/absence of student	High
Admin	Adds/removes users,	Low

2.4 CONSTRAINTS

- The system is used mainly by university students and instructors/teachers.
- This is an application for university.
- This application is used for attendance checking system of the university.

2.5 ASSUMPTIONS AND DEPENDENCIES

- 2.5.1 It is assumed that our desktop application will already be installed on computer in lecture/class/lab rooms.

3 Specific Requirements

3.1 EXTERNAL INTERFACE REQUIREMENTS

User Interfaces

Hardware Interfaces

Software Interfaces

None

Communications Interfaces

3.2 FEATURES

3.2.1 LOGIN PAGE

3.2.1.1 Login success/fail

3.2.1.1.1 The user types his unique username into a username field.

3.2.1.1.2 The user types his password into a password field.

3.2.1.1.3 The user presses “LOGIN” button to enter into system.

3.2.1.1.4 [Inverse Requirement] The system do not allow the user to enter to system, if user types wrong username or password, or both, or leaves it empty.

3.2.1.1.5 [Error Handling] The system displays an error event, if user types wrong username or password, or both after pressing “LOGIN” button.

3.2.1.2 Forgot PASSWORD/LOGIN

3.2.1.2.1 The system provides “Forget password/login” link in login page.

3.2.1.2.2 System redirects user to change password/login page, if user pressed “Forget password” link.

3.2.1.2.2.1 User types his username into username field.

3.2.1.2.2.2 User types his email into email field.

3.2.1.2.2.3 The user presses “Remind” button, to send the new password to his/her email

3.2.1.2.3 The system checks for matching and existence of entered username and email in database.

3.2.1.2.4 [Inverse Requirement] The system do not redirect user to “Forget password” page, if user did not pressed “Forger password” link.

3.2.1.2.5 [Inverse Requirement] The system do not send message with user's password, if his/her entered username and email does not exists in database.

3.2.1.2.6 [Error Handling] The system displays an error message, if username or email does not exist in database.

3.2.2 Instructor's view page

3.2.2.1 This page shall display all information about student's attendance

3.2.2.2 The system shall allow to user to select needed faculty/group/student

3.2.2.3 The system should display all available faculties for the current user

3.2.2.4The system shall display all available groups after one faculty is selected by user

3.2.2.5 The system shall display all students' attendance list after one group is selected by user

3.2.2.6 The "modify button" shall be displayed in the page where required group's attendance list is displayed

3.2.2.7 [Inverse requirements] System should not allow the user to select the faculty which is not related to him(system will provide this function with user's profile information)

3.2.3 Modify Page

3.2.3.1 The Instructor shall press "Modify" button from "View attendance" page to open "Modify page".

3.2.3.2 The system shall show attendance table of group that selected in "View attendance" page.

3.2.3.3 Instructor can change "-" to "+" or vice versa ("- means absent, "+" means present) in shown table.

3.2.3.4 The Instructor shall press "Save" button to save changes.

3.2.3.5 The system shall check changes and update database.

3.2.3.6 The system shall return to "View attendance" page.

3.2.4 Student Registration Page

3.2.4.1.1 Only Students can register to the system.

3.2.4.1.2 The System should allow to Student to register when register item is pressed.

3.2.4.1.3 The System shall return the registration form.

3.2.4.1.4 The Student must fill all the fields.

3.2.4.1.5 The Student must select courses to study from existing courses.

3.2.4.1.6 The Student shall insert his/her photo when "Browse" button is pressed.

3.2.4.1.7 The ID of the Student has to be same as in www.portal.sdu.

3.2.4.1.8 The System shall automatically check if the “ID” of Student is already registered using AJAX (Message “This ID is already exists” is showed).

3.2.4.1.9 The System shall save all data when “Submit” button is pressed.

3.2.4.2 Add New Instructor

3.2.4.2.1 Only Administrator can add new Instructor.

3.2.4.2.2 The System should allow to Administrator to add the Instructor when “Add New Instructor” item is pressed.

3.2.4.2.3 The System shall return registration form.

3.2.4.2.4 The Administrator must fill all the fields.

3.2.4.2.5 The System shall check if the Instructor is already registered.

3.2.4.2.6 The Administrator shall select courses for Instructor from existing courses.

3.2.4.2.7 The Administrator shall give the password for Instructor (The System shall ask to change password when Instructor logs first time).

3.2.4.2.8 The System shall save all data when “Save” button is pressed.

3.2.4.2.9 The System shall return to main page of Administrator when “Cancel” button is pressed.

3.2.4.3 Delete Instructor

3.2.4.3.1 Only Administrator can delete the Instructor.

3.2.4.3.2 The System shall display list of all Instructors when “Delete” button is pressed.

3.2.4.3.3 The Administrator shall select Instructors that he/she wants to delete.

3.2.4.3.4 The System shall send message box to Administrator that he/she is going to delete.

3.2.4.3.5 The System shall allow deleting Instructors when “Ok” button is pressed.

3.2.4.3.6 The System shall return to list of Instructors when “Cancel” button is pressed.

3.2.4.4 UPDATE Instructor

3.2.4.4.1 Administrator can update Instructor’s information.

3.2.4.4.2 The System shall display list of all Instructors when “Update Instructor” item is pressed.

3.2.4.4.3 The Administrator shall select an Instructor that needs to be updated.

3.2.4.4.4 The System shall display profile of Instructor.

3.2.4.4.5 The Administrator shall edit fields that need to be updated (For example change courses of the Instructor).

- 3.2.4.4.6 The System shall save updated info when “Update” button is pressed.
- 3.2.4.4.7 The System shall send message box that profile has successfully updated.
- 3.2.4.4.8 The System shall return list of all Instructors when “Cancel” button is pressed.

3.2.5 Face Recognition Page

- 3.2.5.1 This application page takes photo for attendance
- 3.2.5.2 Camera display and “shot” button shall be displayed in this page after running the program
- 3.2.5.3 The process of taking a photo of student shall be done by student manually
- 3.2.5.4 Only correctly taken photo shall be sent to database
- 3.2.5.5 Correctly taken photo shall be sent to database in order to recognize the student's face
- 3.2.5.6 System shall display message like "attendance mark is added " after all of the process above are done.
- 3.2.5.7 [Error Handling]If the taken photo **has bad quality/taken incorrectly(face in the photo is too big/too small/half face,etc)** system shall display error message like "Try again"
- 3.2.5.8 [Inverse requirement] System should not send the taken photo to db if it **has bad quality/taken incorrectly(face in the photo is too big/too small/half face,etc)**

3.3 PERFORMANCE REQUIREMENTS

3.4 DESIGN CONSTRAINTS

3.5 SOFTWARE SYSTEM ATTRIBUTES

- 3.5.1 Reliability**
- 3.5.2 Availability**
- 3.5.3 Security**

3.5.4 Maintainability

3.5.5 Portability

3.6 DESIGN CONSTRAINTS

Java, PHP, MySQL, OpenCV

3.7 SOFTWARE SYSTEM ATTRIBUTES

Reliability

Availability

Security

Maintainability

Portability