SOFT-FORCE TEAM

SDU Dormitory Portal System Documentation

Instructor: Zhasdauren Duisebekov

Made by: Khamit Berik, Abai Zhardem, Askar Baibossyn, Abilkaiyr Doszhanov

27 February, 2016 year.

CONTENT

1 Introduction

1.1 PURPOSE

The purpose of this document is to provide detailed requirements information about the SDU Dormitory Portal System.

1.2 SCOPE

This document will describe the use cases and features of the SDU Dormitory Portal System.

1.3 DEFINITIONS, ACRONYMNS, ABBREVIATIONS

TERM	DESCRIPTION
SERVER	A part of application which runs on the physical server system and stores information about users, events, etc.
CLIENT	A part of application which runs on the user's computer
ADMIN	An admin is head of the dormitory, up bringers. They always can access to any table and any data in the application
CORPORATIVE MAIL	As there is 'Send Message' functionality, this system must have its own mail. It is going to be something like <sdu.dormitory@gmail.com>
SDU Dormitory Portal	Name of the application described in this documentation
STUDENT	Students are second kind of users. They will have their own ID's and passwords to login into the system, but will not have any ability to change and commit datas except own profile
LIST OF STUDENTS	Special kind of functionality to access the student in the system, change his datas and commit the changes.
DOCUMENTS	Means students and applicants private documents to submit when they register in the dormitory (ID, Photo, X-ray card, 084 Form etc.)

1.4 REFERENCES

// list of references for the reader of this document (if any)

1.5 OVERVIEW

2 Overall description

SDU Dormitory Portal System is an application which will supplement dormitory working system and handle a lot of hand works such as fill the blanks, glue the photos to the students documents and etc. Later this is going to be one of the most important part of not only dormitory but the university else. Because if you are an advisor of some student, you can easily login to the system and get full address information about your pupil.

2.1 PRODUCT PERSPECTIVE

This application is similar to SDU Educational Portal. Because there is also same students their ID's, and information, But one of the big differences there is no courses and marks. And as the same adviser exist and control of the administration staff.

2.1.1 Concept of Operations

Dormitory System application's server part will run no the university's server and client part will be as WEB Application. It means that user can access their data any time via any device. All the datas will be stored on the server database.

2.1.2 Major User Interfaces

See Appendix.

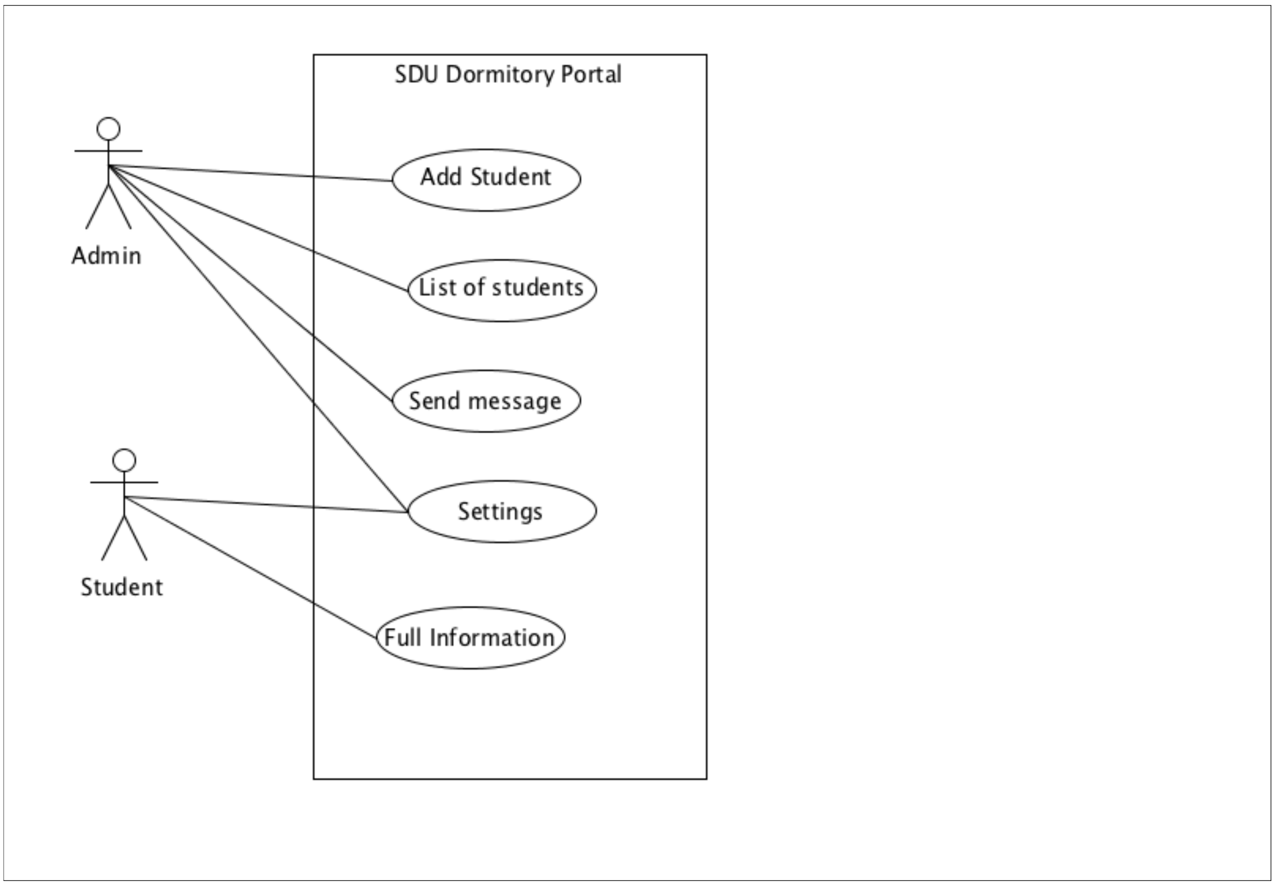
2.1.2.1 Example Screenshot and description

See Appendix.

2.1.3 Hardware Interfaces

This software requires no more than standard personal computer or tablet and even the mobile phone.

2.1 PRODUCT FUNCTIONS



2.2.1 Add student

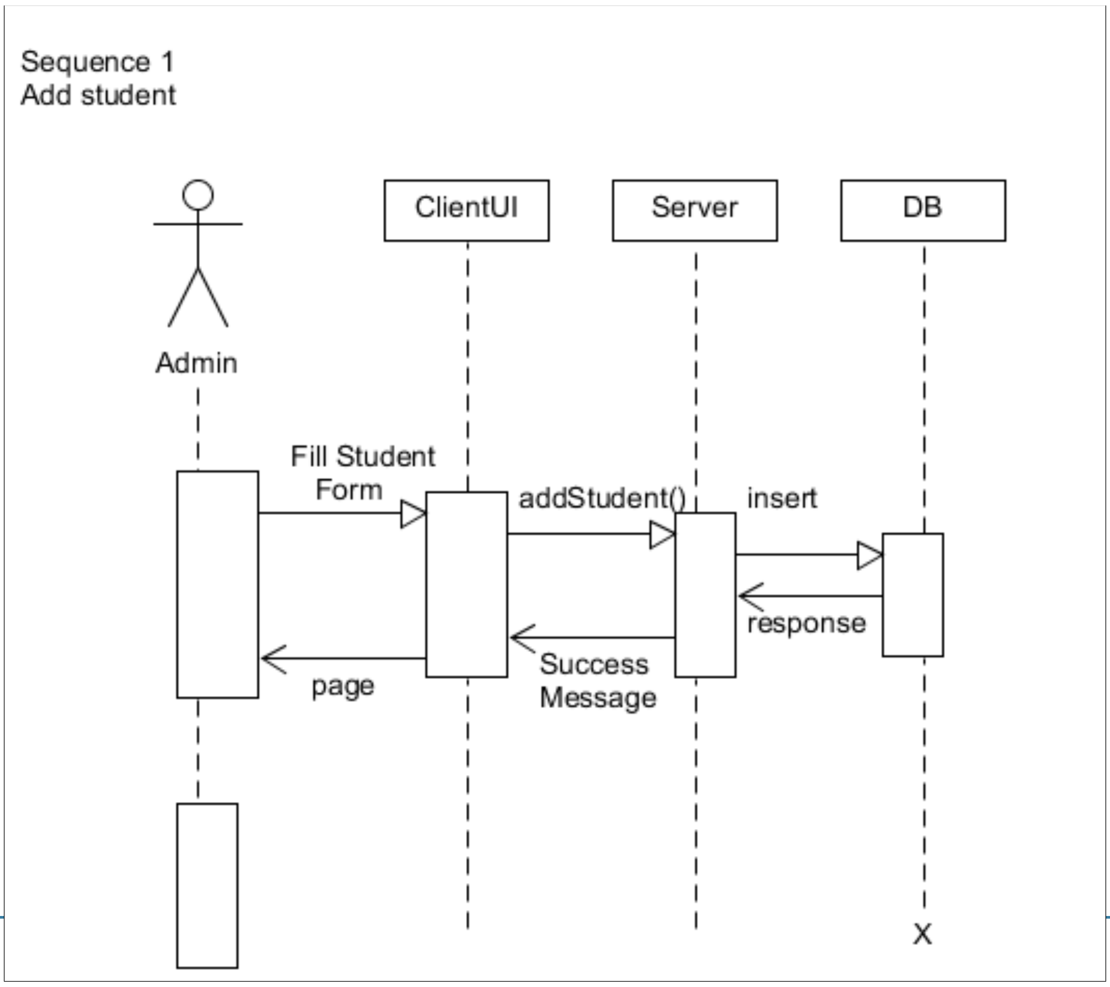
Only admin has possibility to this stuff.

Step 1: Admin fill the student form and presses the 'submit' button

Step 2: 'Submit' button binds with the server, where server sends all the datas to the database.

Step 3: Server uses commands such like 'insert into students values(id, name, surname …)'

Step 4: After all database responses that datas added successfully, that responses comes to the ClientUI via server again in the alert form.



2.2.2 Get list of the Students

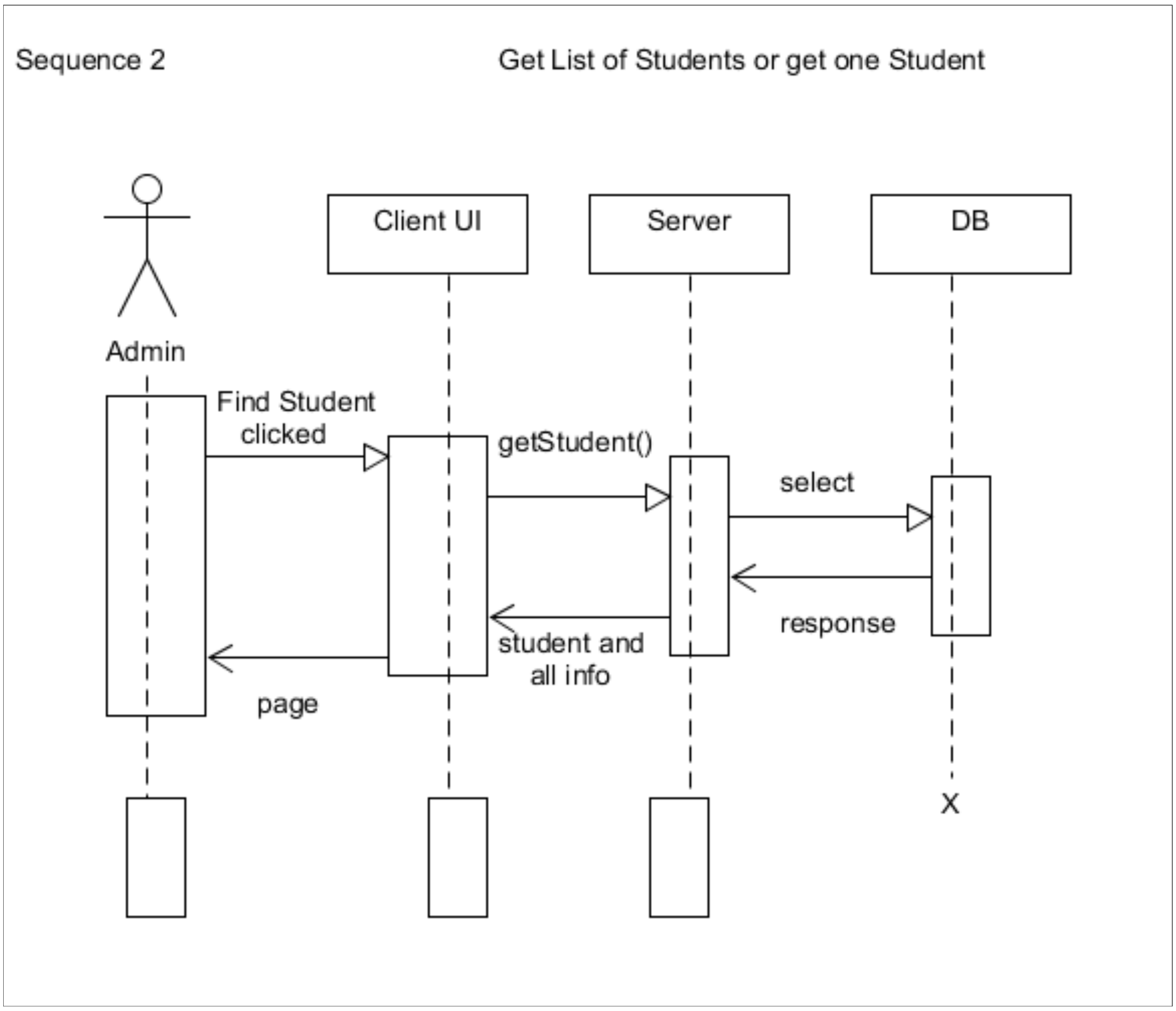
Only Admin has privileges to execute this functions.

Step 1: Admin types name or surname of the student on the Search Bar and presses ‘Find’ button.

Step 2: After what, ‘Find’ button calls a function in there server ‘getStudent()’.

Step 3: ‘getStudent()’ function’s executes sql request such and references to the database with some think like ‘select * from Student where name = %search_bar.value% ’. And it gives success response.

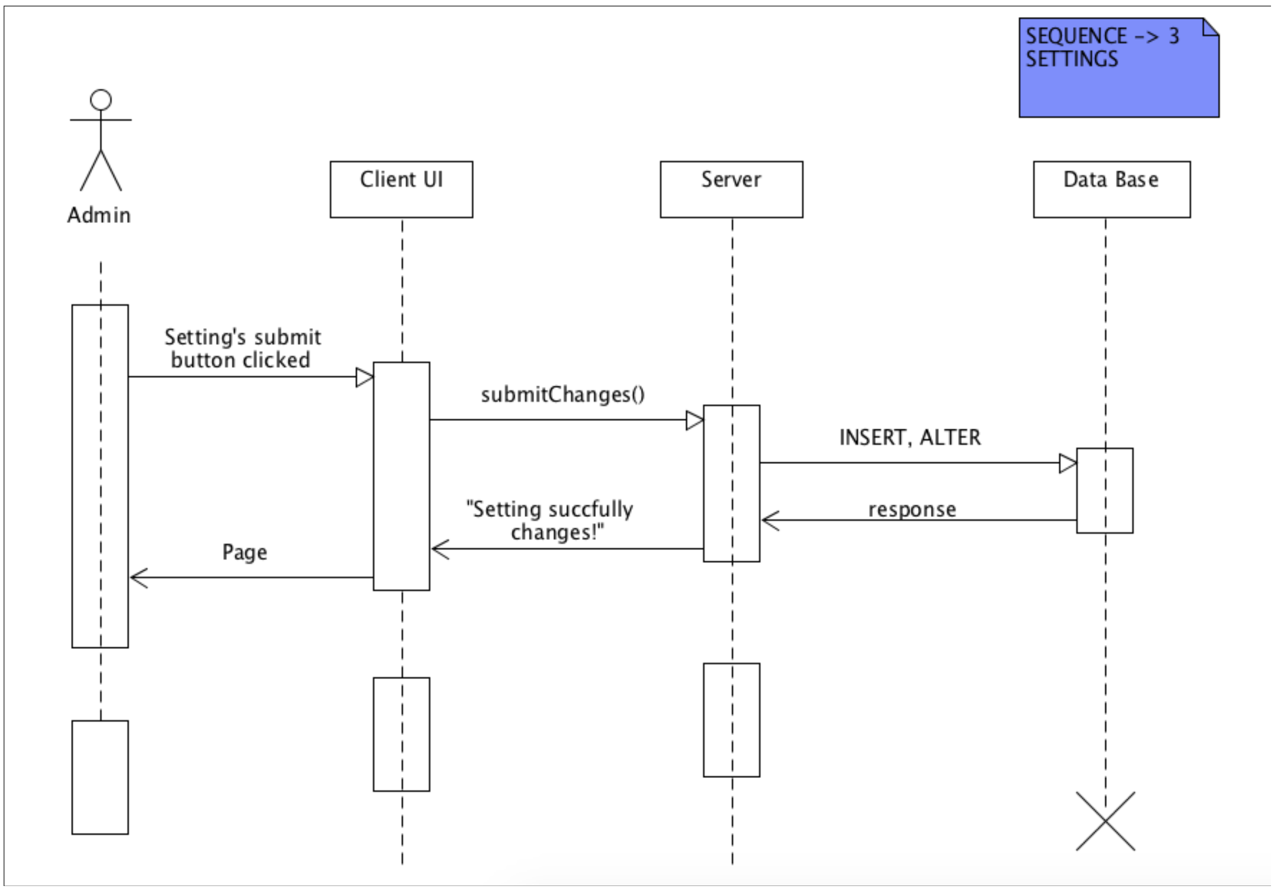
Step4: Response with result appears on the client UI.



2.2.3 Setting Profile

This function is available to both of the users [Admin and Student]

Step 1: User click to the object which he/she wants to edit and after doing press save changes.
Step 2: Save changes button calls method submitChanges() from the server
Step 3: Submit changes execution sql request such as 'insert, alter table' with following values and so on.
Step 4: After what, it gives response that tables successfully changed, and it appears on the client UI



2.2.4 Send Message

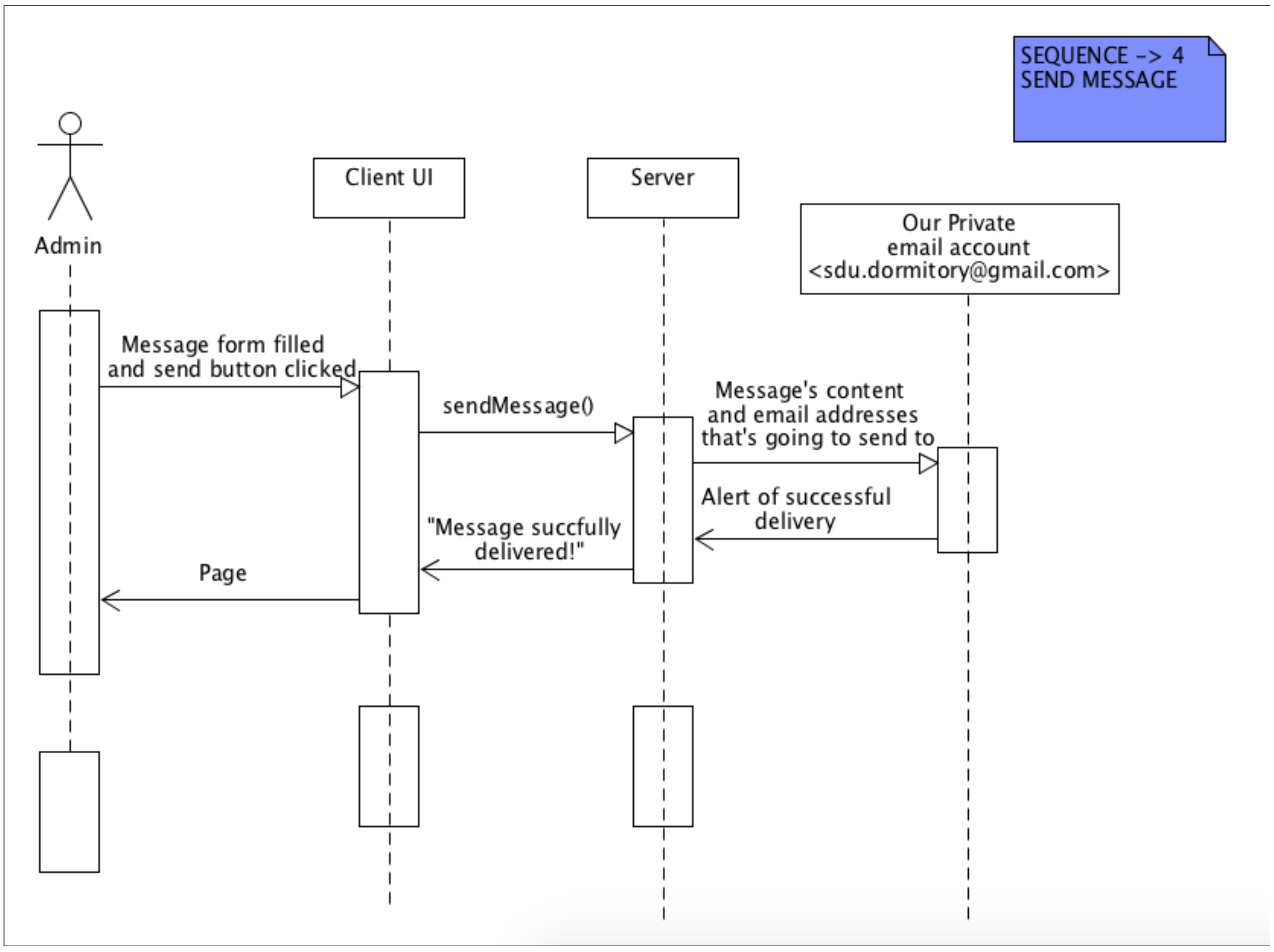
This can do only admin.

Step 1: Admin fills the message form, types message text, and chooses via checkbox to which course students send this message.

Step 2: After what, admin presses send button and it calls sendMessage() functions from the server.

Step 3: Server executes special header by the certain plugin focused on sending stuffs to the emails. And sends it to or Corporative mail account.

Step 4: Our account automatically send message text to the given addresses and responses about success. That success appears on the client UI via server part.


SEQUENCE -> 4
SEND MESSAGE
Admin
Client UI
Server
Our Private email account <sdu.dormitory@gmail.com>
Message form filled and send button clicked
sendMessage()
Message's content and email addresses that's going to send to
Alert of successful delivery
"Message succfully delivered!"
Page

2.2 USER CHARACTERISTICS

TYPES	CHARACTERISTICS	FREQUENCY OF USAGE
ADMIN	As we said before, admin has access to all the tables and datas and has absolutely full privileges according to student's datas manipulation.	HIGH
STUDENT	Can login, only see private information. And set profile photo, add mail and change password.	LOW

3 Specific Requirements

3.1 EXTERNAL INTERFACE REQUIREMENTS

3.1.1 User Interfaces
3.1.2 Hardware Interfaces
3.1.3 Software Interfaces
3.1.4 Communications Interfaces

3.2 FEATURES

3.2.1 FEATURE-1 .... 3.2.1.1 ...

3.2.1.2 .....

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 OTHER REQUIREMENTS

// ADD Appendices (if any)
// Regenerate Table of Contents