**Software Design Specification**

**Table of Contents**

1. Introduction ..................................................................................…..............

1.1 Purpose of this document ...........................................................................

1.2 Scope of the development project ..............................................................

1.3 Definitions, acronyms, and abbreviations ...................................................

1.4 References ...................................................................................................

1.5 Overview of document ................................................................................

2. Conceptual Architecture/Architecture Diagram ............................................

2.2 Structure and relationships .........................................................................

2.3 User interface issues ...................................................................................

3. Logical Architecture (Class Diagram, Sequence Diagram, State Diagram) …..

3.1 Logical Architecture Description .......................................................................

4.0 Execution Architecture ....................................................................................

4.1 Reuse and relationships to other products .........................................................

5.0 Design decisions and tradeoffs ...................................................................

**1. Introduction**

The Software Design Document is a document to provide documentation which will be used

to aid in software development by providing the details for how the software should be built.

Within the Software Design Document are narrative and graphical documentation of the

software design for the project including use case models, sequence diagrams, collaboration

models, object behaviour models, and other supporting requirement information.

**1.1 Purpose of this document**

This document will define the design of the one runway simulator. It contains specific

information about the expected input, output, classes, and functions. The interaction between

the classes to meet the desired requirements are outlined in detailed figures at the end of the

document.

**1.2 Scope of the development project**

We describe what features are in the scope of the software and what are not in the

scope of the software to be developed.

In Scope:

a. Application for the job/internship by the students of the NIIT University.

b. Students can retrieve the information about the internship/jobs.

c. Preparation of the aptitude test.

Out of Scope:

a. Selection process for any internship

b. There is no communication between the NIIT University and the Industry via

this application.

**1.3 Definitions, acronyms, and abbreviations**

IEEE: Institute of Electrical and Electronics Engineers

SDS: Software Design Specification

**1.4 References**

1.4.1 R. S. Pressman, Software Engineering: A Practioner’s Approach, 5th

Ed, McGraw-Hill, 2001.

1.4.2

IEEE SDS template

**1.5 Overview of document**

This SDS is divided into seven sections with various sub-sections. The sections of the

Software Design Document are:

1. Introduction: describes about the document, purpose, scope of development

project definitions and abbreviations used in the document.

2. Conceptual Architecture/Architecture Diagram: describes the overview of

components, modules, structure and relationships and user interface issues.

3. Logical Architecture: describes Logical Architecture Description and

Components.

4. Execution Architecture: defines the runtime environment, processes,

deployment view.

5. Design Decisions and Trade-offs: describes the decisions taken along with

the reason as to why they were chosen over other alternatives.

6. Pseudocode for components: describes pseudocode, as the name indicates.

7. Appendices: describes subsidiary matter if any.

**2. System architecture description**

2.1 Overview of modules / components

**Components Overview :**

**Sequence Diagram**: Sequence diagram will depict the usability of the software sequentially which makes the customer completely understand the procedural behavior of the software.

**Class Diagram:** Class diagram depicts the relation between different classes present in the software scenario. It depicts the methodology used by a particular class and how the variables of one class affects the methodology of another class. This will help us to calculate the level of cohesion and coupling in the software.

**State Diagram**: State diagram depicts the behavior of the state when a particular input arrives. When an input goes in a particular state, it traverses to another state based on the given parameters. It gives a detailed information about the usability of the software from user’s perspective.

**Pseudo code**: Pseudo code gives the rough idea of working of the software in technical terms. It gives the details of the different classes and how it is inherited or is a parent of another classes and how they are affected by the variables in all the classes.

**2.2 Structure and Relationships**

* Login/Sign-up
  + Login Authentication (Must be a registered user)
  + Sign Up and create a new account if a new user (student)
  + For Normal Users - Using features like edit profile, view events and register for an event, view and download certificates.
  + For Admin User – Using features like edit profile, add events, delete & edit events, add user, delete user, view users, edit user, add email, delete email, edit email.

* Registration for games
  + users logs in using a registered account
  + user chooses a games according to their interest.
  + Can cancel an existing registration.

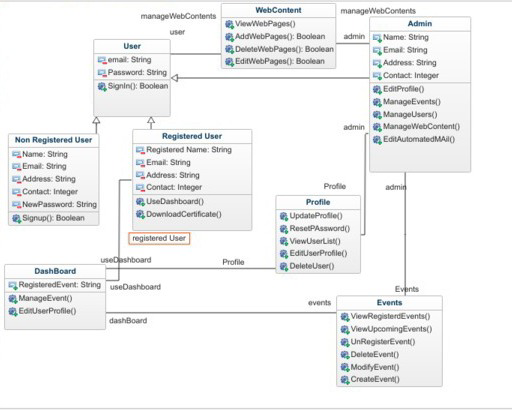
* Creating and editing profile
  + user logs in using a registered account.
  + Can view profile, details etc.
  + edit existing ones.

**2.3 User Interface Issues**

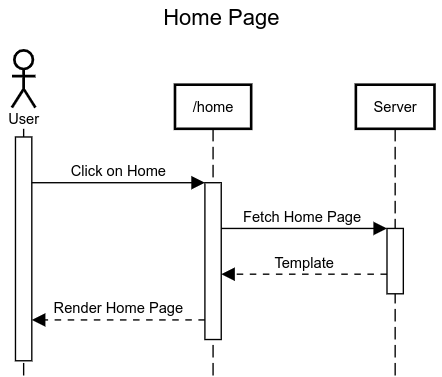
* We had maintained the formality of the design since its being used for a university purpose
* The design would not be very fancy.
* We had to keep in mind all the states the website would achieve while creating.

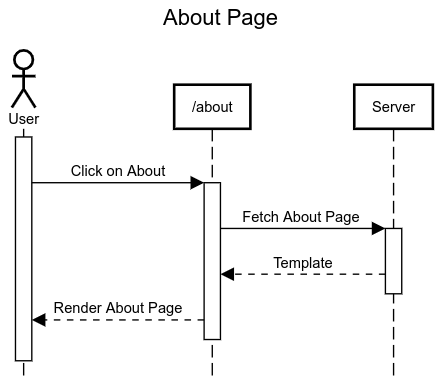
**3. Logical Architecture (Class Diagram, Sequence Diagram, State Diagram)**

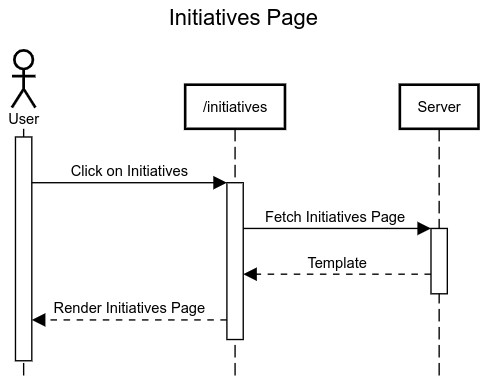
**Class Diagram:**

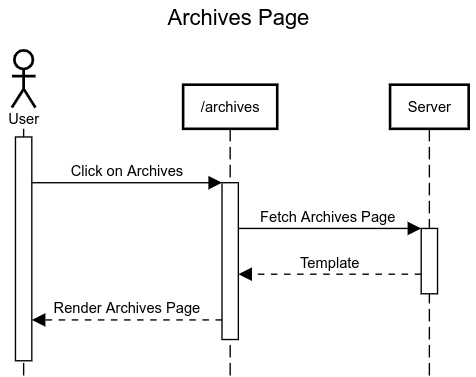


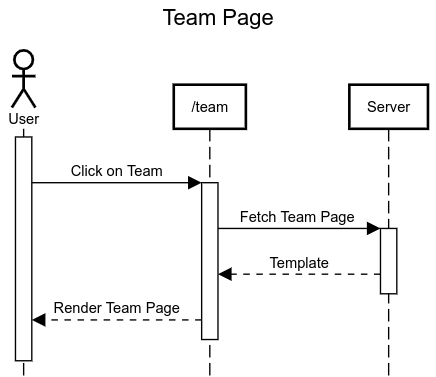
**Sequence Diagrams:**

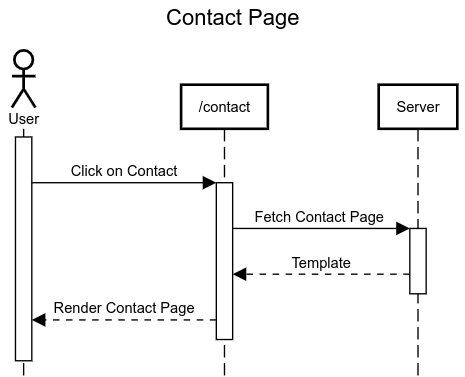


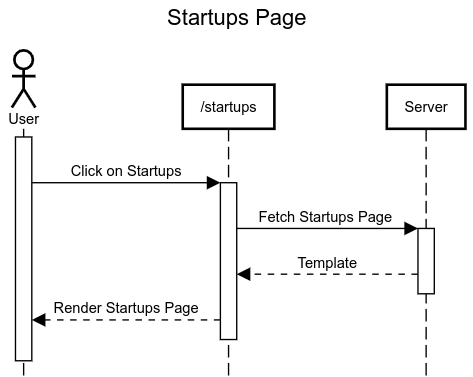


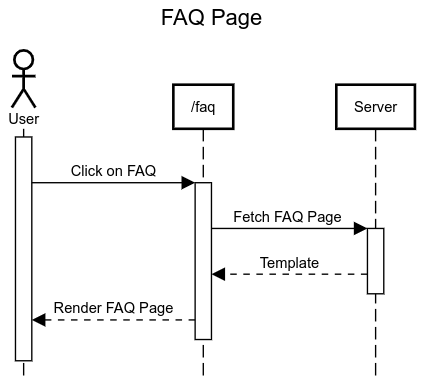


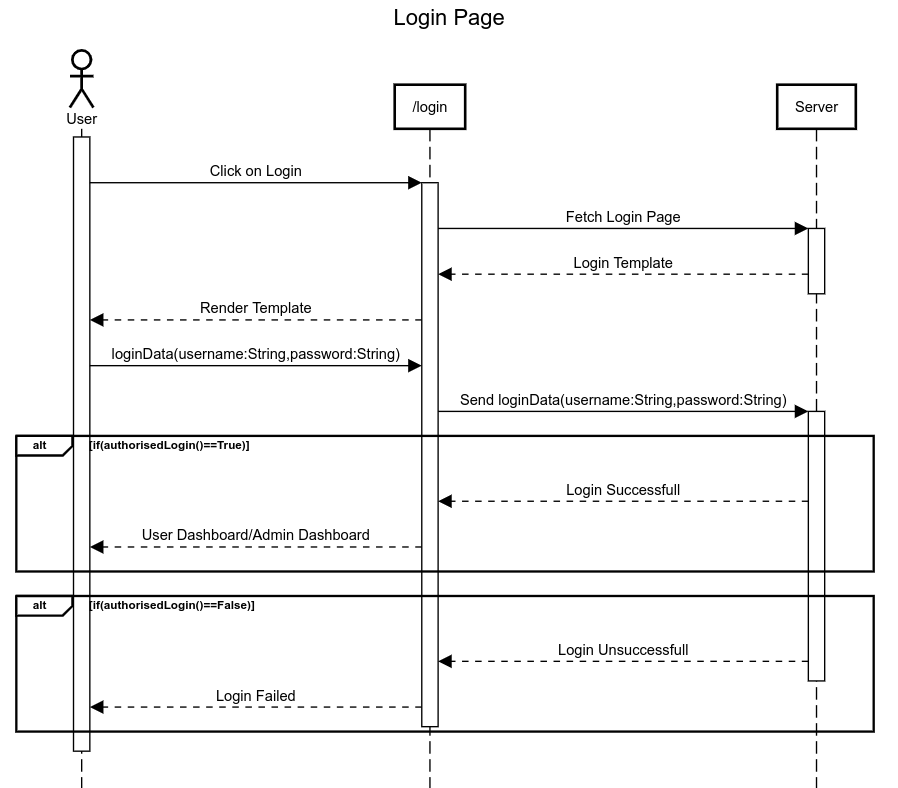


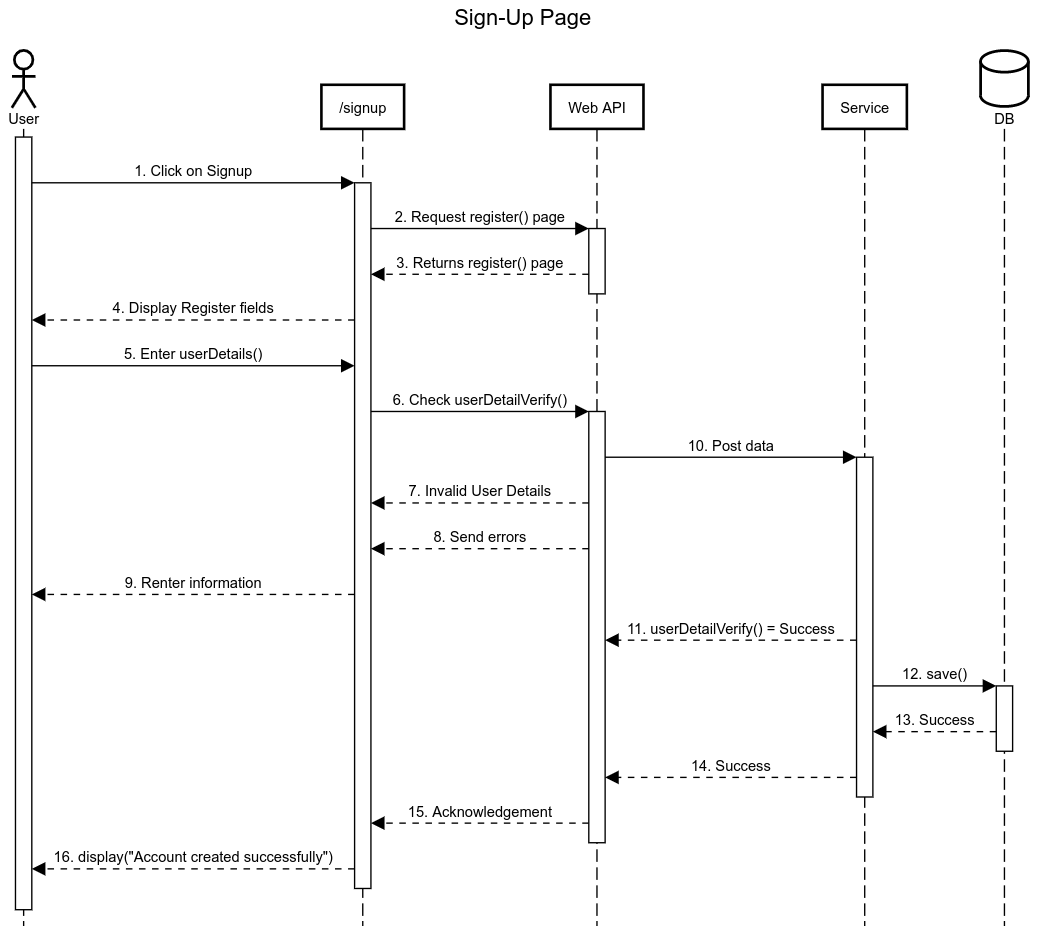


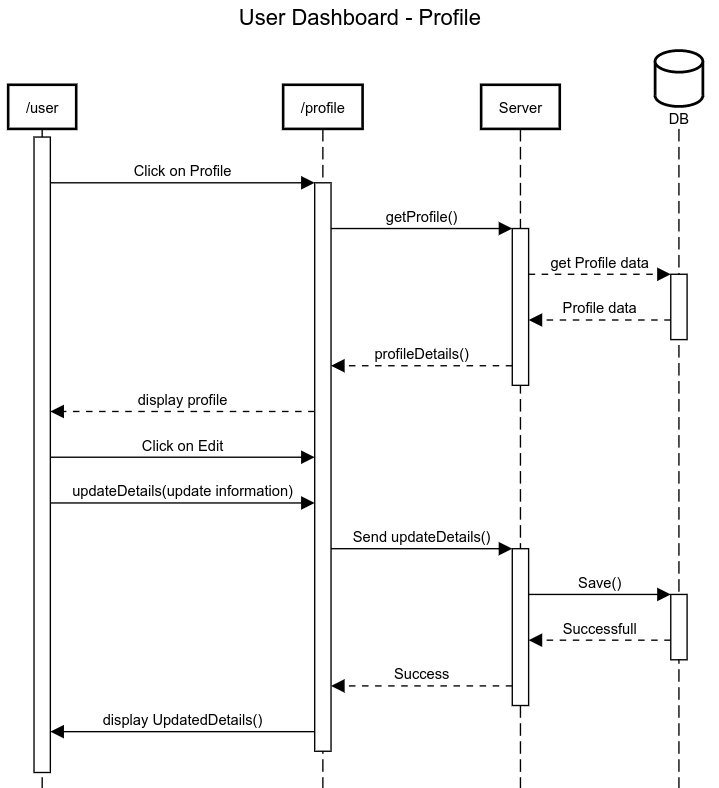


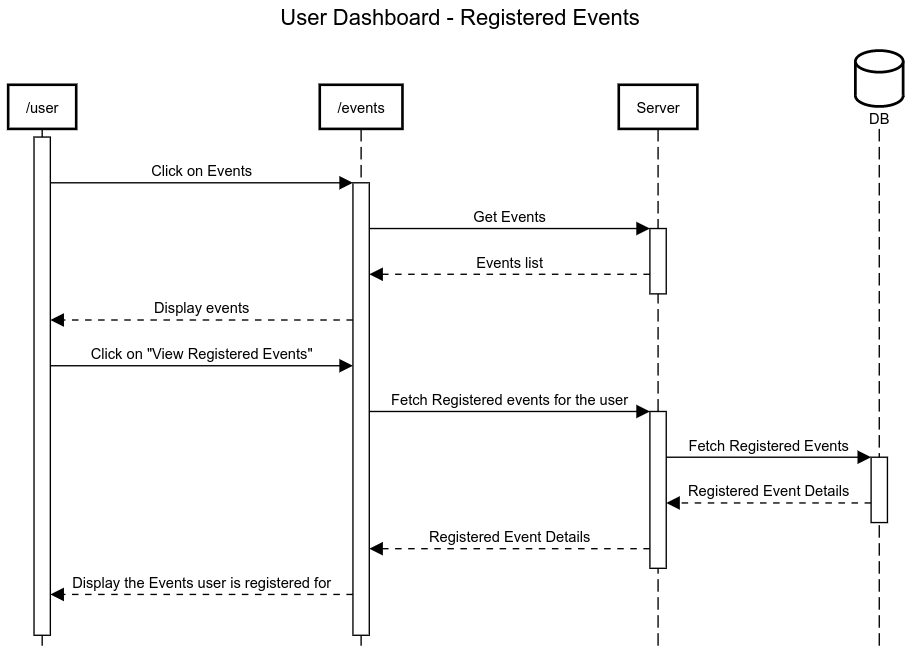


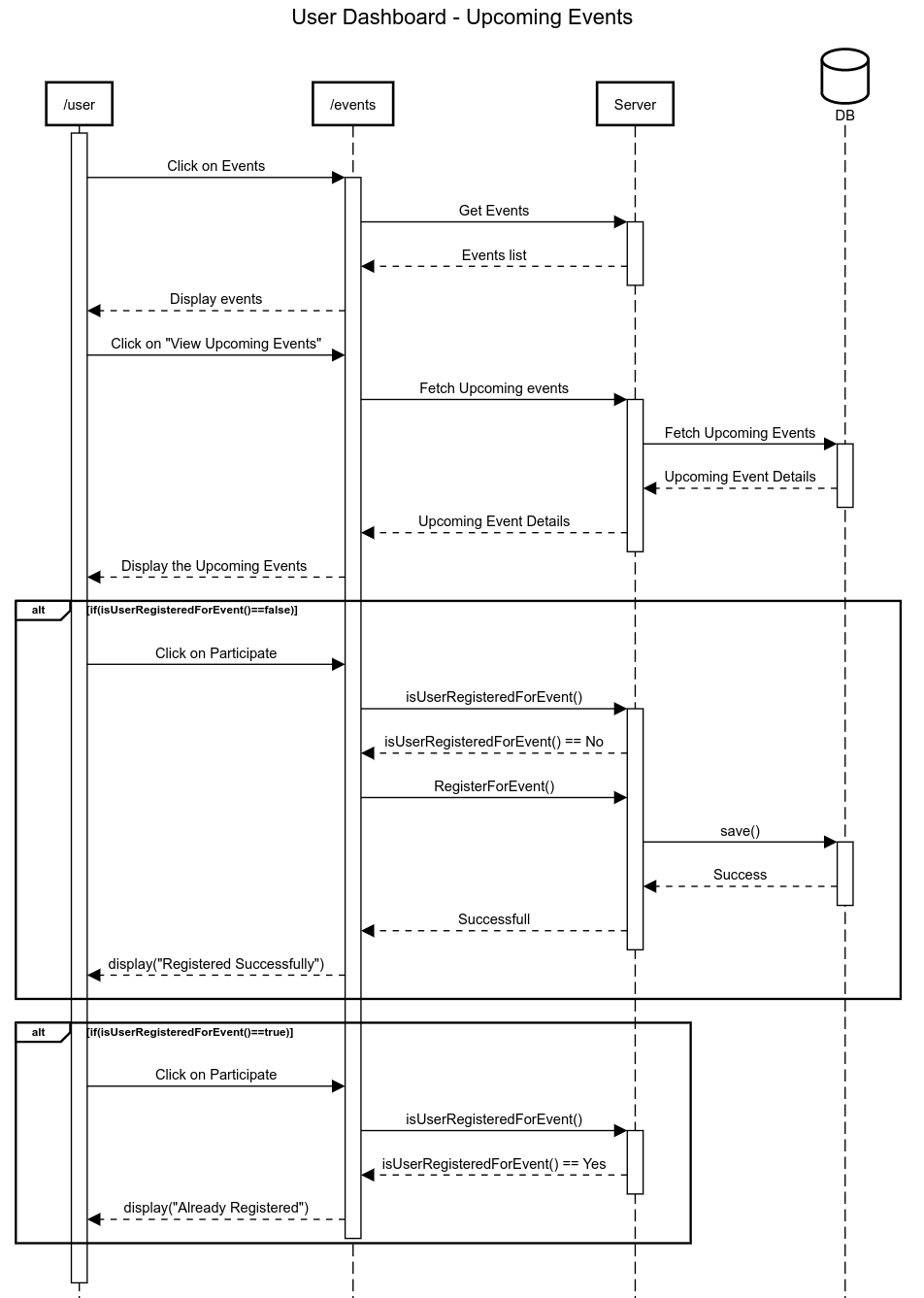


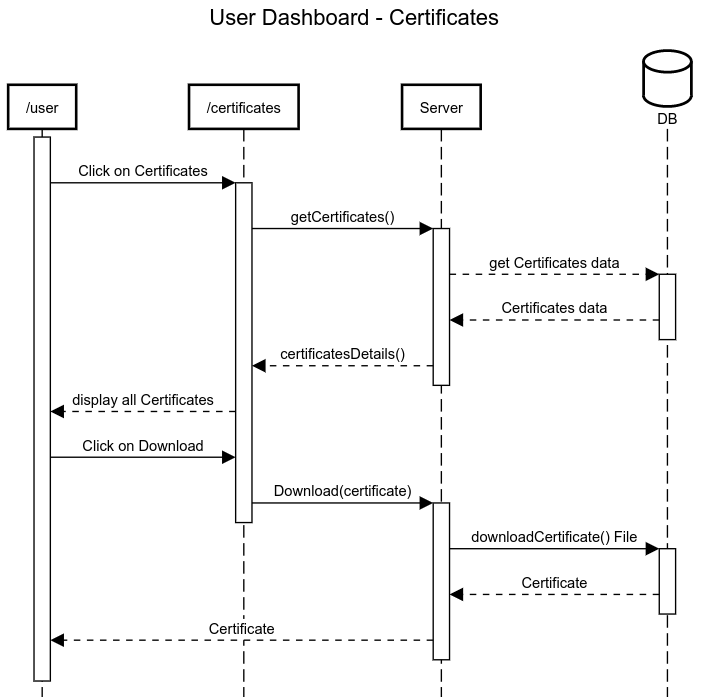


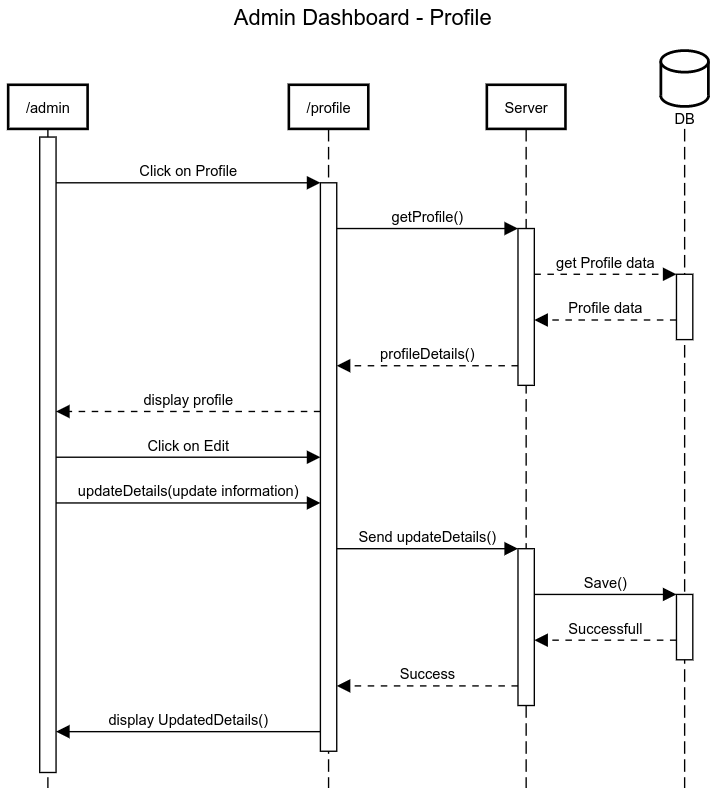


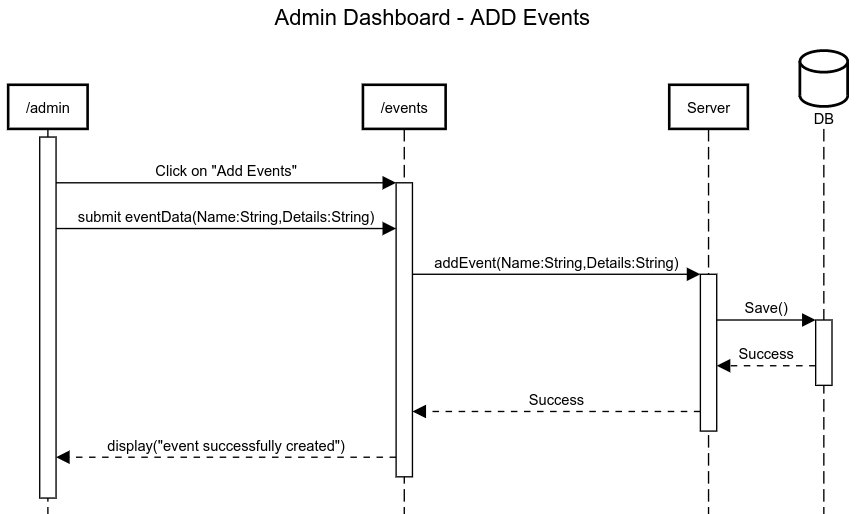


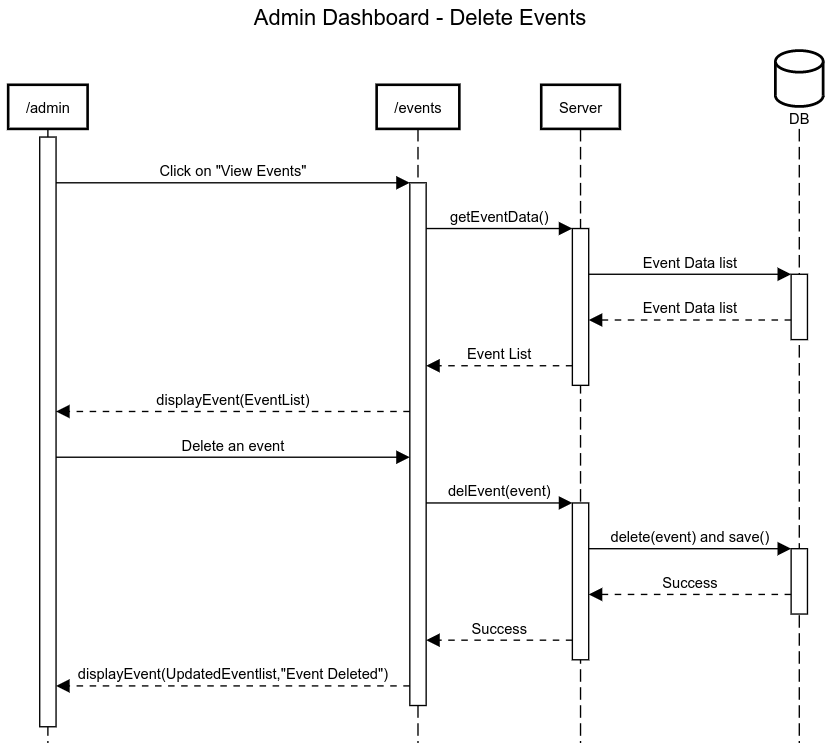


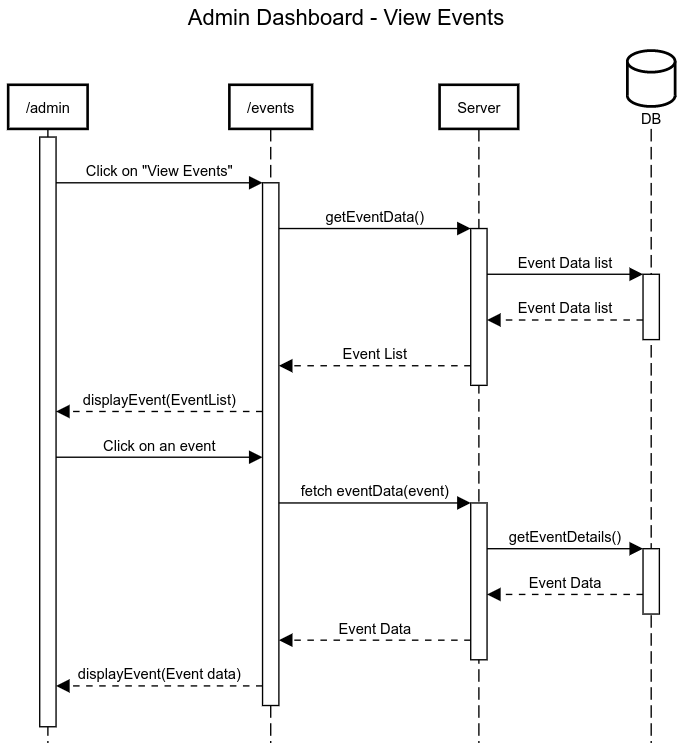


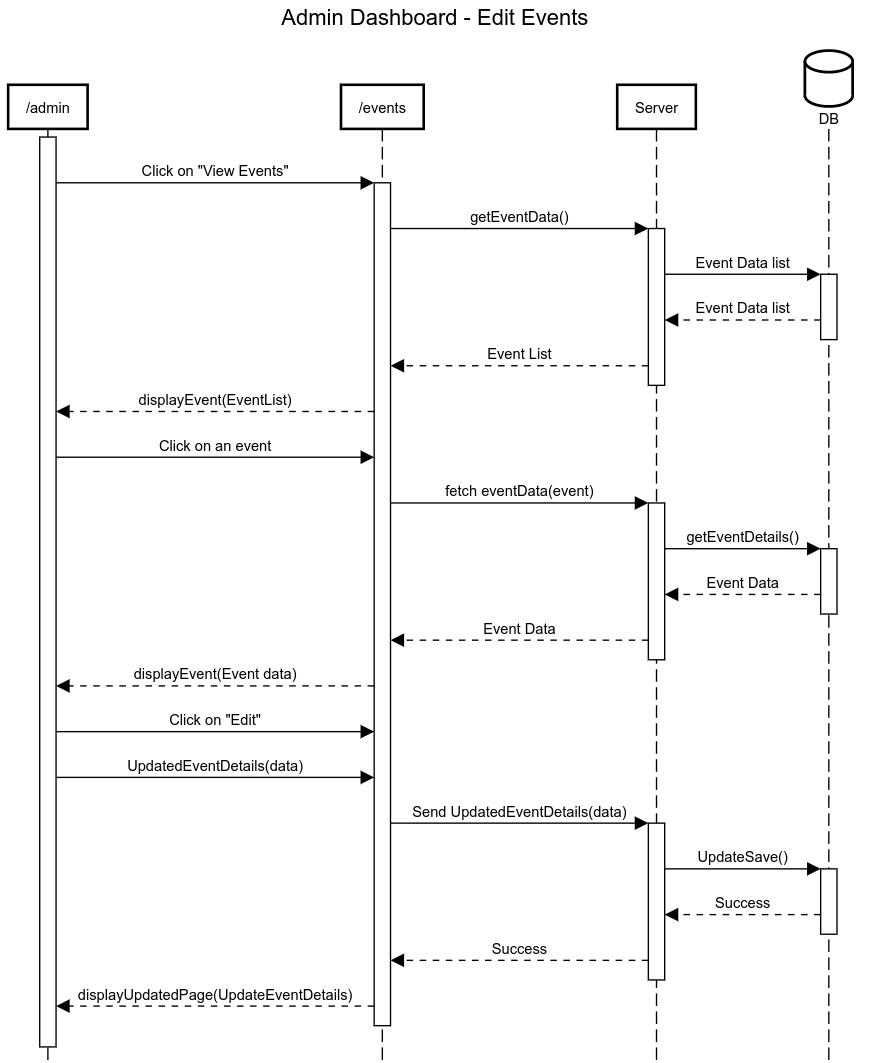


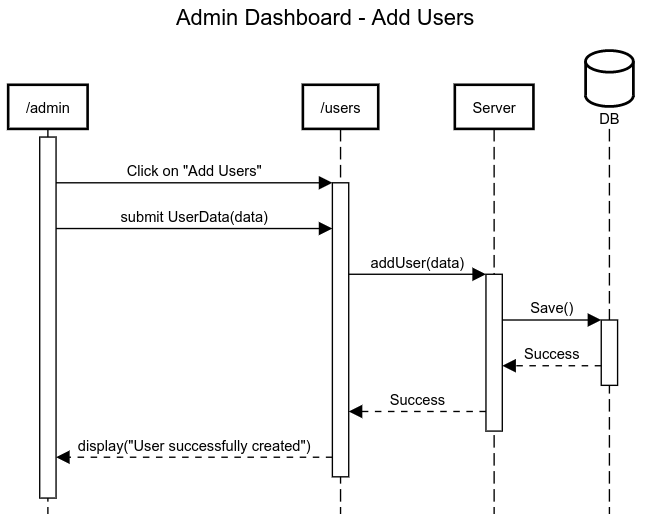


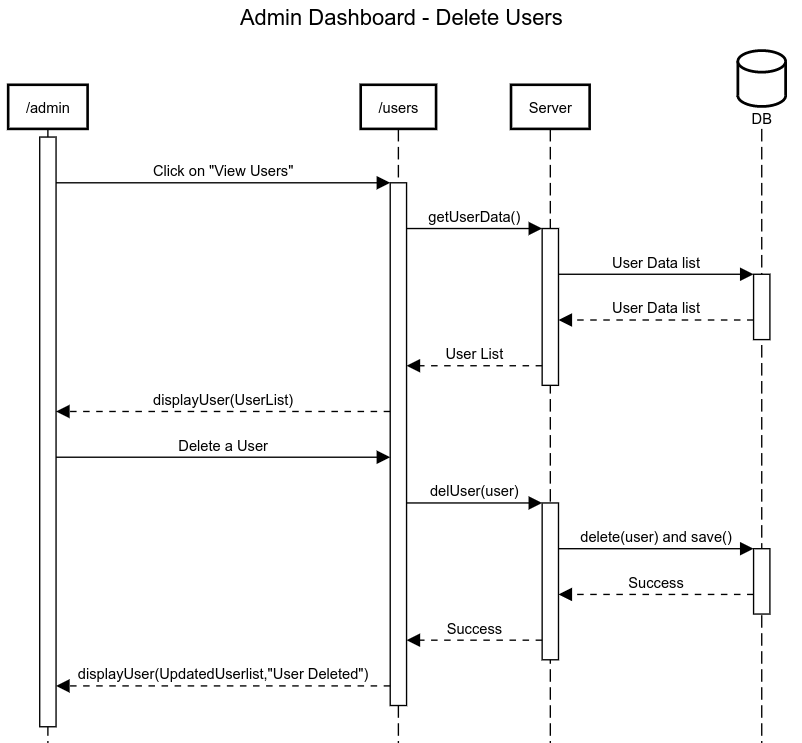


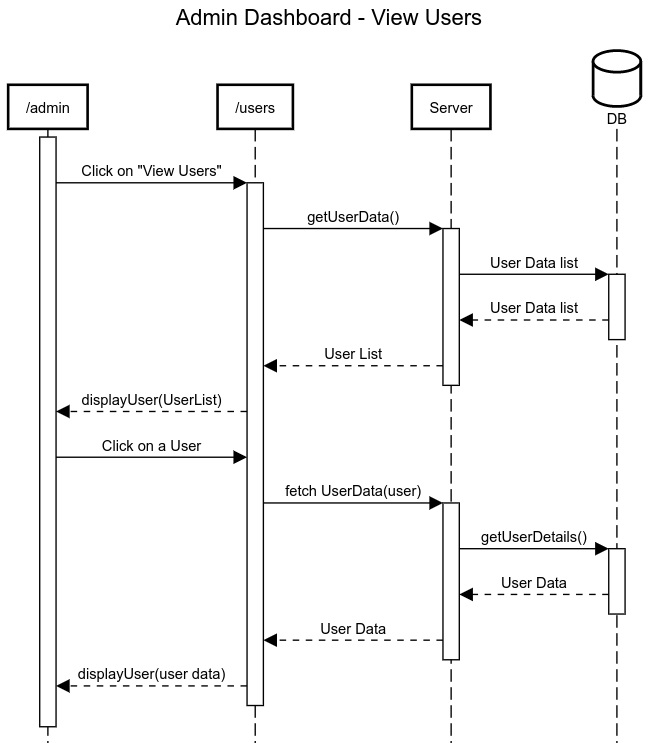


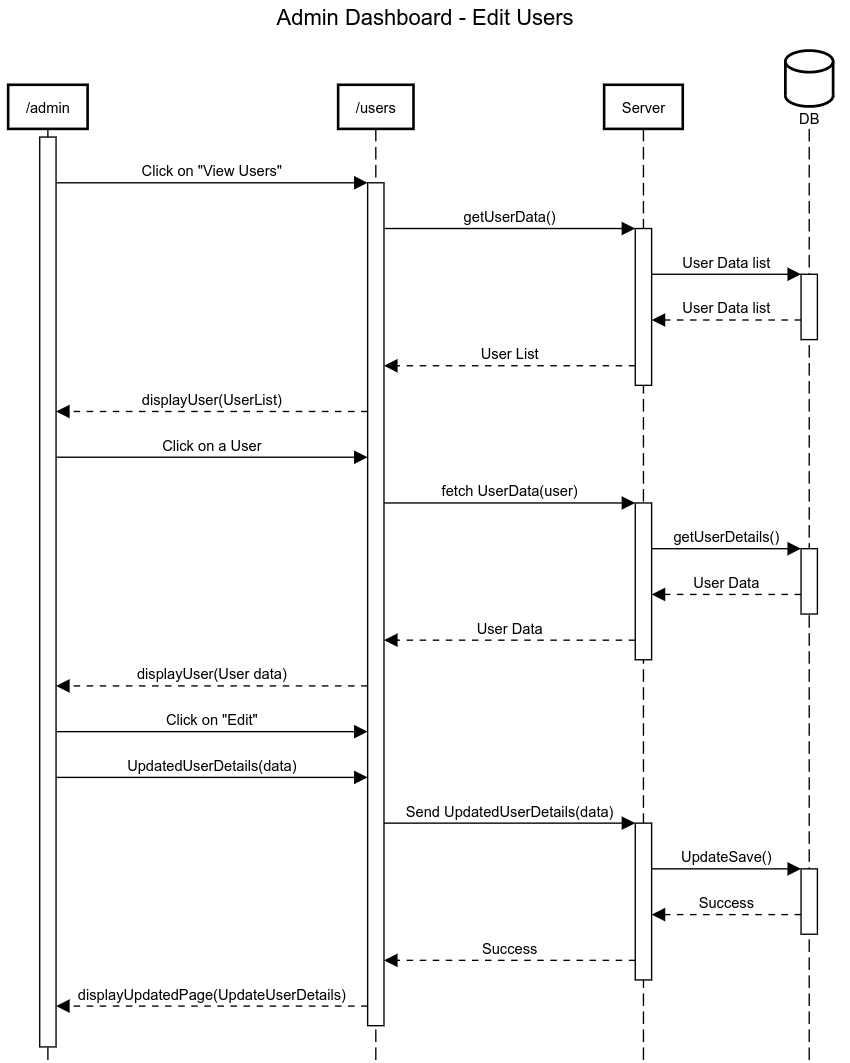


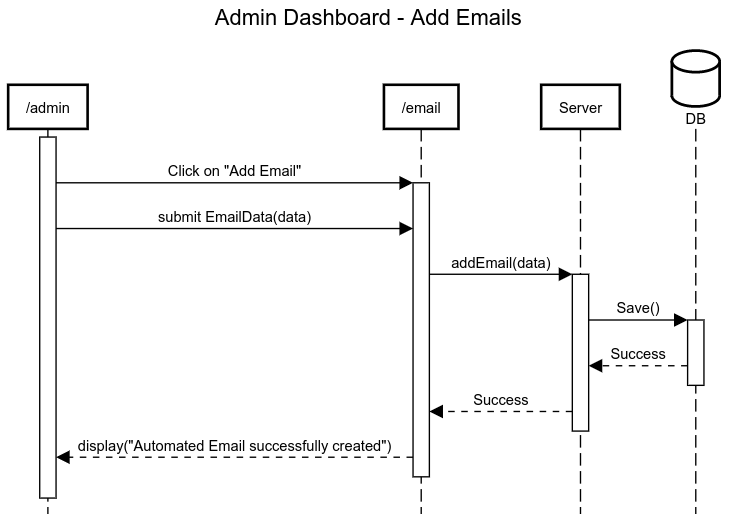


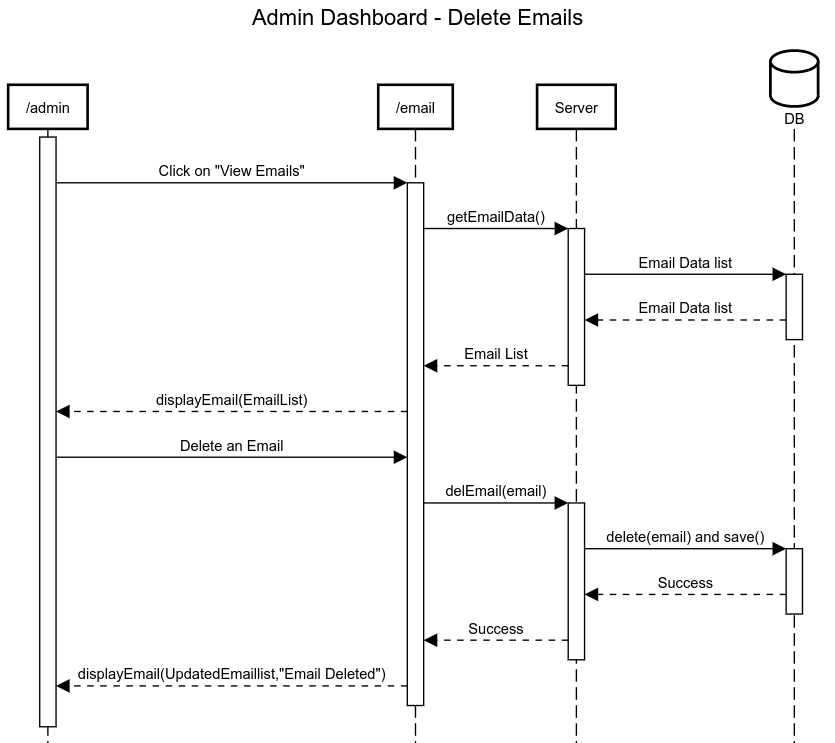


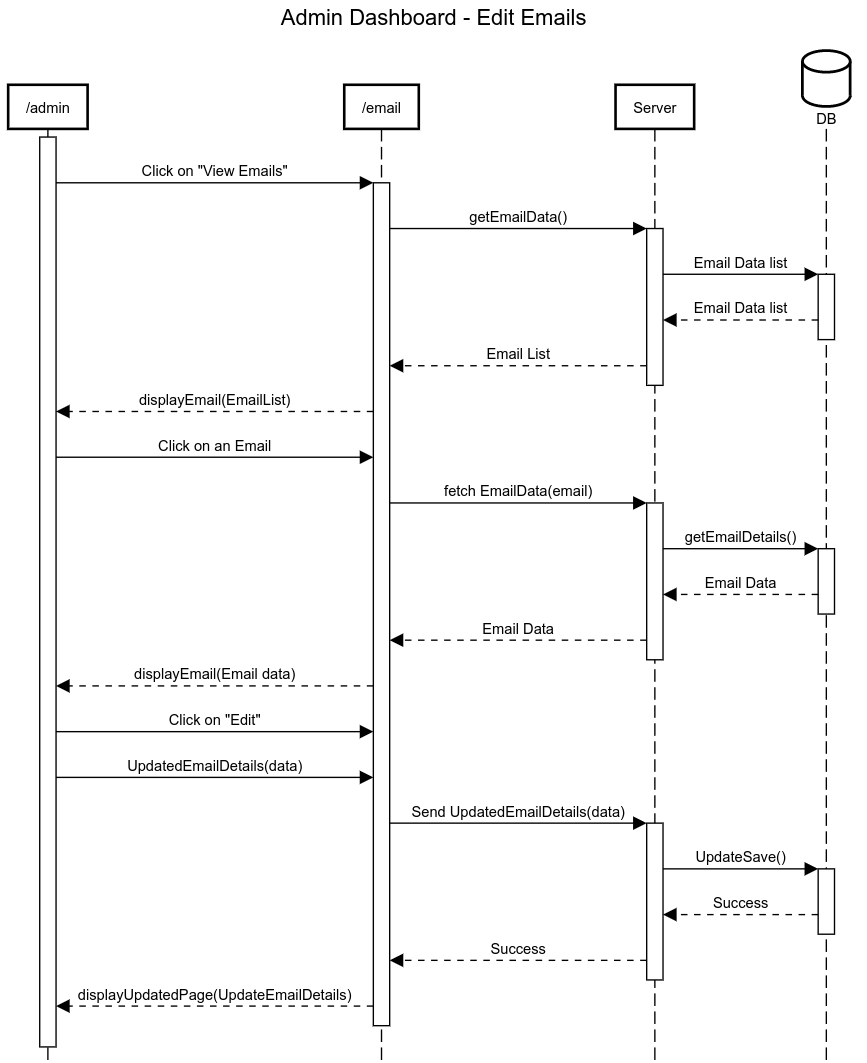




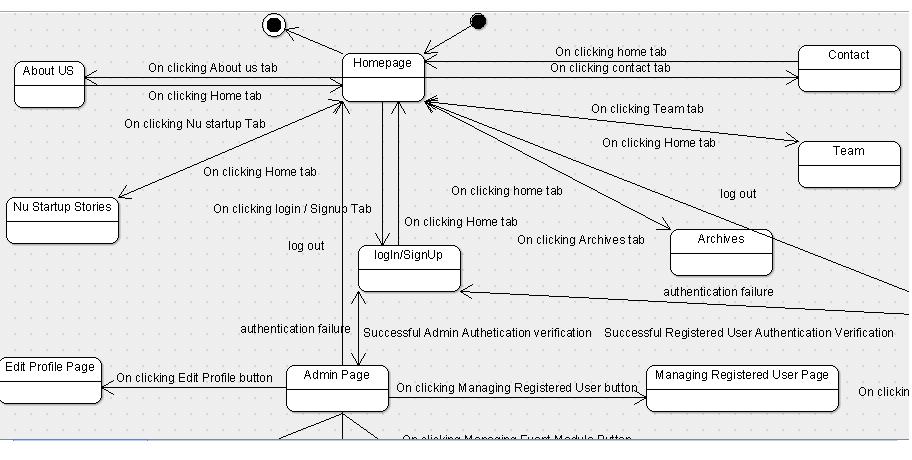


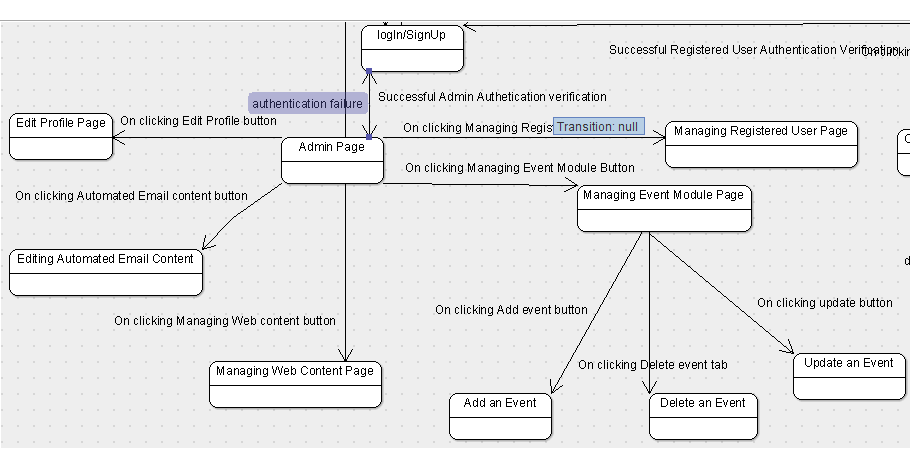


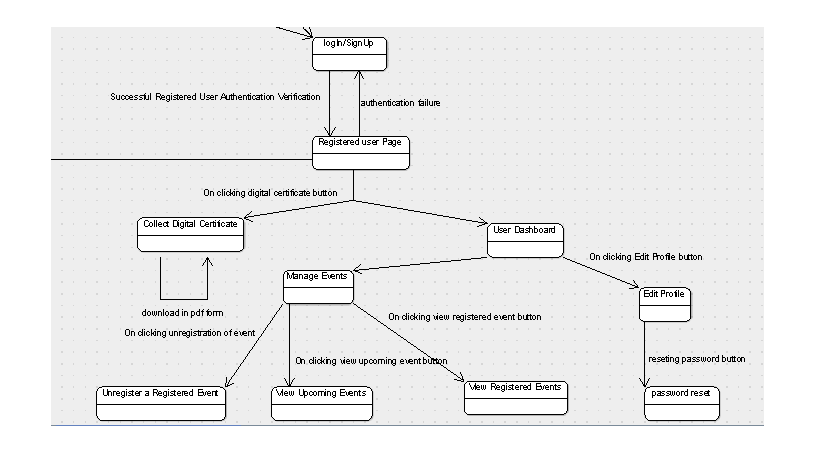




**State Diagrams:**







**3.1 Logical Architecture Description**

**3.1.1 Class Diagram explanation:**

There are 8 classes in class diagram named User , NonRegisteredUser , RegisteredUser , DashBoard , Events , Admin , Profile , WebContent .

RegisterdUser , NonRegisteredUser and Admin extends User class which is shown by generalization linkage . All the member functions and variables of User class can be accessed by RegisteredUser class , NonRegisteredUser class and Admin class. Registerd User has functions UseDashboard and DownloasCertificate an this class is linked with Dashboard class with association linkage .An association indicates that the system you are developing stores links of some kind between the instances of the associated types . Dashboard is linked with RegisteredUser class and along with it is linked with Profile and Events class with association linkage.

NonRegisteredUser class has a single function i.e sign up and is linked only with User class extending it. Admin class is linked with 3 other classes with association linkage. Classes are Webcontent , Profile and Events and is extended to user class . WebContent is linked with user and Admin class with association , profile class is linked with admin class and dashboard

class , Events class is linked with admin and dashboard class .

The Class diagram mainly 2 types of linkage Generalisation and association linkage , first one shows inheritance and second one links between the instances of associated types.

**3.1.2 Sequence Diagram:**

Arrow line signifies there is a send message taken place. Response is being shown by

dotted arrows.

**3.1.2.1 About Page:** Admin puts data on about page regarding company statistics,

which later can be viewed by student on their StudentAbout page.

**3.1.2.2 FAQ:** Student posts question on this page, which later can be viewed by all

other students and can be answered by the Admin. On being answered by Admin, the

student also gets notification for the same.

**3.1.2.3 Home:** Admin posts company data through their InsertCompany Page, which

later is visible on the Home Page of the students. It also allows students to apply for

any job after their profile being approved.

**3.1.2.4 Login Page:** It allows students to login with their niitUniversity mail domain

and Admin to login with the username and password that are being registered in the

database already. It loops being on same page until the correct information is not

given.

**3.1.2.6 UserUpdate Profile:** Student can update their preference choices.

**3.1.2.7 View Profile:** Admin can search for the name of the student, by the data being

searched on the database and shown in response. Then the Admin can also download

the student’s CV and either approves that profile or sends notification to student for

requirement of any document. Approval is given only when no extra requirement of

document is there.

**3.1.2.9 View/Update Contact:** Admin puts up the contact information on the

Admin\_Contact page, which later can be viewed by students on their Student\_Contact page

**3.1.3 State Diagrams:**

**3.1.3.1 Homepage:**

It is the web page that we see when we open the web site.

**3.1.3.2 About:**

On clicking about us tab ton tab bar, we land up on a website description page.

**3.1.3.3 Startups:**

On clicking this tab, we land up on a web page where all the home grown startup stories are laid out briefly.

**3.1.3.4 Archives:**

On clicking this tab, we get a option to choose between further more options like events, media coverage.

**3.1.3.5 Team:**

On clicking this tab menu, we can find out different team members of the CIIE organization.

**3.1.3.6 Contact:**

On clicking this tab menu, we can find out the contact numbers and email id of the corresponding CIIE team members.

**3.1.3.7 Login/Signup:**

On clicking this tab menu, we land up on a user authentication rendering page, where we can sign up for admin or registered user with the valid details.

**3.1.3.7.1 Admin desk:**

On entering valid admin authentication details in the login page, we land up on a page where we can manipulate web pages contents to user data.

**3.1.3.7.1.1 Managing event module page:** In this we can add, delete or update an event in the website.

**3.1.3.7.1.2 Managing registered user page:** In this web page, we can retrieve user data or delete or even change registered user data.

**3.1.3.7.1.3 Managing web content:** in this web page, we get option to manipulate data of different web pages of the website.

**3.1.3.7.1.4 Editing email content:** in this web page, we get a option to change the content of the automated email which goes to the newly registered user.

**3.1.3.7.1.5 Edit profile:** in this web page, we get a option to edit our profile , such as resetting our password.

**3.1.3.7.2. Registered user desk:**

In this web page, we get a option to use various facilities provided to a registered user.

**3.1.3.7.2.1 Collect digital certificate:**

By clicking this button, we get a option to download the digital certificate for the events we have participated in NU CIIE events.

**3.1.3.7.2.2 User dashboard:**

In this web page, we get personalized space to set our profile and manage our events.

**3.2 Class Name : User**

**Description : This class classifies all type of users and enables the user to view webpage and login or sign in**

**3.2.1 Method 1: Sign-In**

**Input : valid Email Id and Password**

**Output : Enters into your account**

**Method Description :**

**After submitting valid credentials and clicking on signIn button , server checks entered details and matches with saved database and when confirmed allows user to use RegisteredUser methods.**

**3.3 Class : NonRegisteredUser**

**Description : This class is for new user and enables a user to sign up.**

**3.3.1 Method 1 : SignUp**

**Input: User needs to fill up all attributes such as Valid Name , Email , Address , Contact details and a new password.**

**Output: Server creates a account for user and gives a account confirmation to user.**

**Method Description :**

**After receiving inputs from user system confirms it and creates a new username and password by which user can signIn into website accessing all functionalities of website.**

**3.4 Class RegisteredUser:**

**Description : This class is used after user is signed in with correct credentials , this opens user account with all the user details and methods accessible by user.**

**3.4.1 Method 1 : UseDashboard**

**Input : Click on Dashboard Icon**

**Output : Dashboard opens for users enabling further functions.**

**Method Description :**

**After signing into account every user can access their dashboard and can see their current activities . Whn clicking on dashboard system provides a dashboard tab with all details and functions.**

**3.4.2 Method 2 : DownloadCertificate**

**Input : Click on certificate you want to download.**

**Output : User certificated will be downloaded on user’s system.**

**Method Description : When user clicks on download button it sends a request to server and server sends the requested certificate in jpg or document format.**

**3.5 Class : Dashboard**

**Description : It has mainly 2 functions which can be used by user by their dashboard for editing their profile and managing their registered events and view Upcoming Events.**

**3.5.1 Method 1 : ManageEvents**

**Input : click on Manage Events button**

**output : It gives option to view your events and register for an event.**

**MethodDescription :**

**When clicked on Manage event button this method is called and it displays upcoming and registered events user can manage their events by this option.**

**3.5.2 Method 2 : EditUserProfile**

**Input : click on edit profile button**

**Output : Option to edit profile and reset password.**

**Method Description :**

**This method has two functions UpdateProfile and Resetpassword which takes input as updating details from user and stores back into databases.**

**3.6 Class : WebContent**

**Description :**

**This class consist all methods related to webpages and is linked with user and admin class. User class only has ViewWebPages method and Admin class accesses all functions associated with it.**

**3.6.1 Method 1 : ViewWebPages**

**Input : No input , Just open website from any browser.**

**Output : Displays starting Webpage.**

**Method Description :**

**As website is accessed from any browser this method displays webpages and content to user.**

**3.6.2 Method 2 : AddWebPages**

**Input : This method is called by admin class , Input is Webpages which are to be added.**

**Output : It gives a boolean value , for success for adding pages.**

**Method Descriptipn :**

**When Admin selects to add page , system requests for pages to be added and return boolean value for the action performed.**

**3.6.3 Method 3 :DeleteWebPages**

**Input : select Web pages to be deleted**

**Output : Deletes WebPAges from server and returns boolean value.**

**Method Description :**

**After se;ecting web pages to be deleted by admin this function deletes the web content associated with it and returns a boolean value .**

**3.6.4 Method 4 : EditWebPages**

**Input: New Content**

**Output : Boolean value for content edited or not**

**Method Description :**

**This takes input from admin and edits the webpages on server returning boolean value for the same.**

**3.7 Class Profile :**

**Description :**

**This class has all the attributes of user and admin and has function update their details and reset or delete their account.**

**3.7.1 Method 1 : UpdateProfile**

**Input : attributes to be updated.**

**Output : Updated profile.**

**Method Description :**

**This function takes all the attributes to be updated by user or admin and sends new data to the database.**

**3.7.2 Method 2 : ResetPassword**

**Input: String ,Old Password , New Password**

**Output: boolean value for password update or not**

**Method description:**

**This method takes old password as input and checks with database , if correct then updates password.**

**3.7.3 Method 3 : ViewUserList**

**Input: Click on View user button**

**Output : Text file consisting details of all user**

**Method Description :**

**This method provides all user details from databse in a text file on admin’s system.**

**3.7.4 Method 4 : Delete User**

**Input : Select User to be deleted.**  
**Output : All data related to selected user and account are deleted.**

**Method Description :**

**Admin selects the User and this method deletes all the details and account from database.**

**3.8 Class Events**

**Description :**

**This class consist all methods related to manage events and is linked to Dashboard class and Admin class.**

**3.8.1 Method 1 : ViewRegisteredEvents**

**Input: Select button**

**Output: displays all events which are registered by user**

**Method Description :**

**On clicking this button this method displays all the events registered by the user.**

**3.8.2 Method 2 : ViewUpcomingEvents**

**Description : This method displays all the events which are upcoming from the database entered by admin.**

**3.8.3 Method 3: UnRegisterEvent**

**Input : Select events to be unregistered**

**Output: Confirmation of unregisteration**

**Method description:**

**Selected events get unregistered and removes from registered events table form database and returns a confirmation message for the same.**

**3.8.4 Method 4 : DeleteEvent**

**Input : Select Events**

**Output : Boolean value for the operation**

**Method Description:**

**Selected Events gets deleted on database by administrator**

**3.9 Class Admin**

**Description - This class has all functions allowed to admin and is linked with WebContent class and Event Class.**

**3.9.1 Method 1 : ManageEvents**

**escription : This method can call all the functions from Events class as per requirement , all functions are described above.**

**3.9.2 Method 2: ManageUser**

**Description : It is linked with Profile class and can call all the functions from Profile class which are described above .**

**3.9.3 Method 3: ManageWebContent**

**Description : It is linked with WebContent class and can call all functions and are described above.**

**3.9.4 Method 4 : EditAutomatedMAIL**

**Input : Select Content to Edit and New Content**

**Output: Boolean value for the action performed.**

**Description : This method allows admin to edit automated email content and save it to database.**

**4.0 Execution Architecture**

Runtime environment required is any device which has internet enabled web browser.

**4.1 Reuse and relationships to other products**

NIL

**5.0 Design decisions and tradeoffs**

The design decision to use two screens separately for admin and student is to provide encapsulation. It may have been possible to get all the information on one screen.

However, using two screens will keep the data of admin separate from the data being accessed by students.

A possible tradeoff when considering links is to use buttons instead of items in the menu. This design decision - to use buttons for navigating between screens - is to enhance visibility. The tradeoff for buttons with descriptive labels rather than text links in the menu bar will be that navigation from screen to screen will be easier.

Descriptive labels will let the user know where he is navigating. Buttons are larger than the text links. Therefore, it is easier for the user to locate the mechanisms needed to navigate from screen to screen.