Bilkent University

Department of Computer Engineering

SENIOR DESIGN PROJECT

Project Name: UniStud

Analysis Report

Supervisor:

Prof. H. Altay Güvenir

Jury Members:

Prof. Özgür Ulusoy

Prof. Uğur Güdükbay

Innovation Expert:

Melih Gezer

Presented by:

Aurel Hoxha

Albjon Gjuzi

Arba Hoxha

Eniselda Tusku

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project, course CS491/2.

Table of Contents

1. Introduction

In the recent years, with the wide popularization of the web, the number of people who actively use different online platforms and applications has increased to an incredible rate, reaching 3.58 billion in 2017 [1]. Among these users' statistics show that the majority of them belong to a group age between 17-28 years old, making these people a very ideal target group for application developers. Students seem to use these platforms excessively but even though here are different applications that help them with several individual needs, there is a lack in the market for application that allow students to access several services in the same time, saving their time, money and energy.

If you are a student enrolled in a university or even an aspiring student, you have to navigate between several applications in order to access services related to the search of the opportunities, tutorials and online courses, finding books or even exploring nearby events. What we aim is to introduce an alternative that allows them to access these services in a single application in a simpler and more effective way.

UniStud will be offered as a web service and as an Android application, in this way reaching a higher number of users and operating better according to the demand of the users. What makes UniStud innovative, is the fact that a student is fed with relevant information without passing from link to another. UniStud will keep users updated with everything that is happening around them. The users can not miss the new opportunities and will easily engage into their educational path. In this report, detailed information about current platforms, proposed system and how it will be implemented will be provided.

2. Current System

The purpose of this section is to give an outline of the information regarding the existing application and platforms that resemble the proposed system. Towards the end of the section the features that make the difference between proposed system and existing platforms will be explained.

2.1 University Search Engine apps and platforms

2.1.1 Study in Europe

Study in Europe is a popular University Search Engine that will find you university or college based on the information you provide.

- The platform is focused only on universities located in Europe.
- Study in Europe is offered only as a web platform.
- The information the portal provides on Free Package is not sufficient for student to get a full background on the university that they are searching. Other packages are extremely expensive and they just redirect students to official website of the university that they are searching.
- Users can search for Universities by providing a keyword such as program, country, language etc.
- User can further filter the result by only one keyword.
- User can get information about the Schengen Visa and residence permit according to each country.
- User can get information about The European education system.
- User can get information about The European Grading system and compare the scores between different countries.
- Premium Users can get information about important application deadlines on the country that they are interested.
- User can get different facts about different countries, such as: academic year, categorization of study programs.
- User can compare tuition fees between different countries in Europe, which are categorized as EU students, NON-EU students.

2.1.2 Top Universities

Top Universities help students find their perfect university, course or study destination using university rankings, country and course guides, events and international student forums [2].

- Top Universities is offered only as a web service.
- The platform provides information for worldwide university rankings, courses, guides and events.
- User can find the perfect university program by completing a short survey.
- User can search for universities by providing study level, subject of interest or study destination.
- User can find overviews according each university which is composed of about, facilities, student life, admissions and career.
- User can see the location of the university and other related content such as: latest tweets, news, videos, images.
- User can compare universities with each other.
- User visit the official website of the university.

2.1.3 Big Future - College Board

Big Future is the part of the College Board platform and facilitate student to quickly perform a university or college search by major, location, type of university, financial aid, test and scores. It contains direct connect the students with pre-registered institutions to send several official scores such as SAT, ACT etc. [3].

- Big Future is offered only as a web service.
- User can filter universities by providing type of campus & housing, sports and activities, diversity, addition support programs.
- User can create a list of preferred universities or colleges.
- User can find overviews according each university which is composed of about, facilities, student life, admissions and career.
- User can calculate the net price for a particular university or college.
- User can search for grants and scholarships.
- User can compare universities or colleges with each other.
- User can view suggestions what other people view in order to have multiple options.
- User visit the official website of the university.

2.2 Career Search Engine apps and platforms

2.2.1 LinkedIn

Founded on December, 2002, and launched on May 5, 2003, it is mainly used for professional networking, including employers posting jobs and job seekers posting their CVs. [4].

- LinkedIn operates via websites and mobile applications.
- User can get contact with recruiters for particular jobs or internships.
- While applying for a new job/internship user have the opportunity to view if anyone of his network has previously worked at that position.
- User can comment on different posts.
- User can search jobs by providing keywords.
- User can save jobs opportunities for consuming them later.
- User can search job posted based on the experience level (Internships, Entry Level, Associate, Director, Executive).

2.2.2 Glassdoor

Founded in 2007 and based in Sausalito, CA, Glassdoor gets its internship listings from several sources, including company websites, partnerships with job boards and directly from employers [4].

- Glassdoor is offered both as mobile application and as a web platform.
- Glassdoor collects jobs from over 10000 companies and utilizes a ranking algorithm to show the users only the relevant news.
- Glassdoor offers an instant way to search for salaries, company reviews and descriptions of job interviews.
- User can view jobs/internships nearby from the “Local Tab”. User can also change the location to get other jobs from different parts of the world.
- User can save jobs opportunities for consuming them later.
- User can search jobs by providing keywords.

2.2.3 Internships.com

Founded in 2010, Internships.com is now owned by textbook rental and online tutoring company Chegg in Santa Clara, CA. It offers 100,000 listings from 60,000 employers [4].

- Internships.com is offered only as a web service.
- User can browse different jobs and internships by categories, locations and companies.
- User can use survey in order to evaluate personal preferences and find out which is the right internship or job.
- User can learn the basics of writing a cover letter and interviewing tips.
- User can search jobs by providing keywords.
- User can save jobs opportunities for consuming them later.

2.3 Virtual Learning Environments apps and platforms

2.3.1 Udacity

Founded in June 2011 and launched in February 2012, Udacity is a for-profit educational organization founded by Sebastian Thrun, David Stavens, and Mike Sokolsky offering massive open online courses. Main features are:

- Udacity is offered both as mobile application and as a web platform.
- The platform helps student to be proficient in a particular course.
- User can search tutorials by providing keywords or by navigating through different available categories.
- The basic courses are free while other courses need to be bought.
- User can join Nanodegree program and Learn in-demand skills, build incredible projects, and gain an industry-valued credential.
- Based on the dream job of the user the platform suggests a path the user should follow in order to achieve it.
- User can get a detailed resume review from an industry professional to ensure that resume effectively conveys user abilities and gets attention from employers.
- Users can get a customized review of LinkedIn profile from an industry professional.

2.3.2 Udemy

Udemy.com is an online learning platform. It is aimed at professional adults. Unlike academic MOOC programs which are driven by traditional collegiate coursework, Udemy uses content from online content creators to sell for profit. Main features are:

- Udemy courses can be accessed from several different devices and platforms, including a desktop / laptop, Android app, iOS app, and Apple TV app.
- Udemy features an extensive, multi-language library, which includes over 80,000 courses taught by expert instructors.
- User can take courses across a wide range of categories.
- Each Udemy course is created, owned and managed by the instructor(s).
- The foundation of each Udemy course are its lectures, which can include videos, slides, text and additional resources the instructor has uploaded.
- Instructors can add quizzes, practice tests, assignments and coding exercises, as a way to enhance the learning experience of students.
- If user is not sure if a course is right for him/her, the user can start a free preview and watch a handful of lectures the instructor has selected.
- If user is not happy with a course, user can even request a full refund within 30 days of purchasing a course.

2.3.3 Coursera

Coursera was founded in 2012 by Stanford University computer science professors Andrew Ng and Daphne Koller. Ng and Koller were inspired by their experiences offering their Stanford courses online in fall 2011, and soon after left Stanford to launch Coursera.

- Course courses can be accessed from several different devices and platforms, including a desktop / laptop, Android app, iOS app.
- Currently, Coursera boasts an active catalog of 2,700 online courses created by partner institutions.
- Coursera courses consists of pre-recorded video lectures. The lectures have student discussion forums, homework/assignments, and online quizzes or exams.
- Coursera courses are free to audit (i.e. watch videos) but if user wants to access graded assignments or earn a course Certificate, user will need to pay.

2.4 Broadcasting apps and platforms

2.4.1 Facebook Live

Facebook Live was launched to some high-profile users beginning in August 2015. By April 2016, everyone had the ability to go Live. Some of its features are:

- Facebook Live is accessible from Facebook mobile application and web service.
- User can open live video streaming to engage with followers and grow audience.
- User can choose the group of people which will be able to watch the stream.
- User can enhance the stream by applying camera AR Filters.
- User can comment, react, share a live stream.
- Maximum time for a stream is four hours.
- User can get notified when a person they follow starts streaming.
- Facebook Live Map allow user to find live streams from all over the world.
- User is able to find live streams by providing keywords.

2.4.2 Instagram Live

Instagram started insta-stories in August 2016, with Live Stories later added in November 2016.

- Instagram Live is accessible only from mobile devices and tablets.
- User can comment, react, share a live stream.
- User can see who is viewing the live stream and user is able to turn off commenting.
- Users can get notified when a person they follow starts streaming, granted they have set their notification settings accordingly.
- User has the option to save the video as an Instagram story for 24 hours.
- User can enhance the stream by applying camera AR Filters.
- You can spice up your Instagram Live broadcast with Face filters. This feature lets you add accessories, hats, stars, rainbow light and many other special effects at the top of your face. This is thanks to the advancements in facial recognition and AI.
- With a current functionality, rolled out in October 2017, two users can broadcast using the same stream.

2.4.3 YouTube Live Streaming

Launched in November 2008 YouTube Live was only available to selected partner organizations and later YouTube has quietly rolled the feature out to all accounts [5].

- YouTube Live is accessible from YouTube mobile application and web service.
- YouTube allows you to add a highlights reel/promotional video to your live stream, for display when the stream is offline.
- For event organizers who want to strip out all of the distractions of a standard YouTube page, the live event player can be embedded in any webpage.
- Each YouTube live streaming event is limited to 8 hours.
- User can comment, react, share a live stream.

2.5 UniStud

UniStud will be offering a collection of the best services mentioned above. The main difference of UniStud with other platforms is that it is not based only in the information taken from the different databases but also real-time contacts, such as students studying in different universities. In contrast with other platforms that are pre-recorded, UniStud will create a more enhanced way to learn by following different live streaming tutorials, so the users can actively participate. Our platform will facilitate the internship opportunities for students’ eager to work by proving an easy communication between parties.

3. Proposed System

In this section, the overview of the UniStud application will be presented. On top of that, the necessary requirements (functional & non-functional) for the application will be specified.

3.1 Overview

UniStud will be focused on the students, the target users of our project all over the world and will offer them a platform where they can access different services simultaneously. Accordingly, the users will be able to access information in four main areas.

The first category will be regarding the search for different academic opportunities such as finding universities, scholarships and internships. The students can search based on their preferences on universities and they will be provided with the university information that fits their needs and in addition with the opportunity to connect with students that are currently studying in these universities. The main reason for this is to feed the users with real data and get a more realistic perspective on the requirements, challenges and procedures to be part of that university. In this category users will also be able to find different internships opportunities from different companies depending on their departments.

In the second category, the platform will provide an online tutorial service where the students can create and share these tutorials with others that are interested in studying a new course or improving an old one. The tutorials will be both online which will be live streamed and then stored once they are finished and private which will be based on user agreement.

The third category will be related to study materials where the users will have the opportunity to buy/sell/exchange/loan any important study materials such as books, notes, devices etc.

Finally, the last category will be regarding student events where the user will be able to keep track of all the events happening around.

3.2 Functional Requirements

3.2.1 User Profile

- Everyone with a valid email can sign in/sign up for the platform.
- For the accounts, there will be a two-step verification.
- The platform is not intended to operate as a social network even though it will include communication between students for a variety of purposes.
- The user's profile will have the student personal information such as university, department, country and area of interest.
- The user can access the internship's list of the companies provided in the platform.
- The user can search the information in the platform, in order to find their wanted information related to their searched universities.
- The user will be able to offer online tutorials for specific courses.
- The users can attend online tutorials and make questions related to the topic.
- The users will get notified for the events and tutorials they have joined.
- The users can buy/sell/exchange/loan study materials from each other.

3.2.2 Platform Data

- Only students will be able to offer online tutorials.
- Companies can add internships opportunities in the portal without having an account.
- Different organization can add events in the portal without having an account.
- There will be a map showing demographic data of the students using the portal.
- In different scenarios (such as searching for a university), sensitive data of the students studying in the corresponding university will be filtered out.

3.2.3 Storage

- As the platform grows some of the online tutorials will be gathered together into collection after live stream has ended and will be stored to our servers indefinitely. However, retrieving these tutorials would require to describe clearly the appropriate information with respect to tutorial. The students might contribute to "resolve" difficulties accessing these tutorials by adding other relevant information.

3.3 Non-Functional Requirements

3.3.1 Usability

- The system should be user-friendly and it should be very easy for the users to interact with it.
- The content in the system should follow the same standard everywhere in it.

3.3.2 Reliability

3.3.2.1 Robustness

- The system should be able to withstand a high number of uses at the same time.
- The system should be able to operate without a failure for a long period of time.
- Probability of failure should be less than 0.03.
- MTBF should be larger than 3000h.
- A system crash should not result in data loss.

3.3.2.2 Security

- Third parties should be prevented from breaking into our system and accessing any kind of data from it.

3.3.3 Performance

3.3.3.1 Efficiency

- The response time of the system should be less than 1 second.
- All user inputs should be acknowledged within 1 second.

3.3.3.2 Scalability

- System should be able to accommodate increased volumes, workloads and users.
- The minimum number of users the program should have before starting to slow down is 5000.

3.3.3.3 Scalability

- The system should be able to handle at least 500 transactions within a minute.

3.3.3.4 Availability

- The system should be available for more than 98.3% of the time and for every student around the world.
- Elapsed time for system to recover from incident and failure should be less than one day.

3.3.4 Extensibility

- Design and implementation of the system should allow it to be able to cater future changes.
- System should be able to manage new entities or provide new feeds.

3.3.5 Modifiability

- Dependencies in the system will be minimized in order to allow quick updates in the future.

3.3.6 Maintainability

- The system should be designed and implemented in such a way that it will optimize the ability of a maintenance personnel to revise or enhance it.
- The degree of flexibility should not affect any of the components, services and functionalities of the server.
- Elapsed time for system to recover and come back to its normal state following an incident or failure should be less than one day.

3.4 Pseudo Requirements

Tentative pseudo requirements are provided as follows:

- The application will have a server and mobile and web applications.
- Implementation language of Android application will be Java.

3.5 System Models

In this section 17 scenario that a user may encounter are included. To explain each scenario in detail, they are divided into subsections as: Scenario Name, Participating actors' instances and Main flow of events. Secondly, the system model will include a use-case diagram. The use-case diagram will be used to describe the events and actions between the user and the system. Furthermore, each use-case is explained in detail to for further explanation.

3.5.1 Scenarios

This section, gives an overview on how the app will function during different user program interactions. Importance of this section is demonstrating the storyline of the application, as well as detecting possible alternative solutions to problems we are trying to solve.

Scenario 1
Scenario Name: CreateNewAccount
Participating actor instances: Jim: NormalUser
Main flow of events:

1. Jim has heard of UniStud but has never used it.
2. Jim download the application from Google Play Store and open it.
3. He chooses to sign up using email account.
4. Jim enters his username as "Jim".
5. He enters his email address as: jim@mymail.com.
6. Jim enters his password.
7. Jim taps the "SignUp" button.
8. UniStud checks if the username is available.
9. UniStud checks if the email already exists in the database.
10. UniStud confirms that the credentials Jim inserted meet the requirements.
11. UniStud creates a new account for Jim.
12. UniStud redirects user Jim to his Home page.

Scenario 2

Scenario Name: LoginAtUniStud

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim opens UniStud application and chooses to sign in using email account.
2. He enters his email address as: jim@mymail.com
3. Jim enters his password.
4. Jim taps the “Login” button
5. UniStud confirms that the credentials Jim inserted are correct.
6. UniStud redirect user Jim to his Home page.

Scenario 3

Scenario Name: CreateNewAccountUsingFacebookCredentials

Participating actor instances: Jim: NormalUser

Main flow of events:

1. User Jim taps on the “Sign in With Facebook” button on the screen.
2. UniStud navigates to Facebook Login website.
3. User Jim enters his email address as "jim@facebookmail.com".
4. User Jim enters his password.
5. User Jim taps the “Sign in Using Facebook” button.
6. UniStud checks if the mail address "jim@facebookmail.com exists in the database.
7. UniStud creates a new account for Jim.
8. UniStud redirect user Jim to his Home page.

Scenario 4

Scenario Name: CreateNewAccountUsingGoogleCredentials

Participating actor instances: Jim: NormalUser

Main flow of events:

1. User Jim taps on the “Sign in With Google” button on the screen.
2. UniStud navigates to Google Login website.
3. User Jim enters his email address as "jim@googlemail.com".
4. User Jim enters his password.
5. User Jim taps the “Sign in Using Google” button.
6. UniStud checks if the mail address "jim@googlemail.com exists in the database.
7. UniStud creates a new account for Jim.
8. UniStud redirect user Jim to his Home page.

Scenario 5

Scenario Name: ChangeSettings

Participating actor instances: Jim: NormalUser

Main flow of events:

1. After Jim has logged into the application he enters his Preferences Page.
2. Jim views all his preference for UniStud account.
3. User Jim navigates to the “Change Password” page of Preferences.
4. User Jim changes his password.
5. User Jim taps the “Save Preferences” button.
    a. UniStud checks if all the given information is accurate and valid.
    b. UniStud saves the changes of user Jim’s account.
    c. UniStud navigates to the Homepage of the user Jim.

Scenario 6

Scenario Name: ViewSavedContent

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is on the HomePage of the application.
2. Jim has a list of different contents such as: saved universities, internships, tutorials, books to be bought and upcoming events with different colors and he clicked the nearest event.
3. UniStud extend the event button to offer further information to Jim.
4. Jim decides that he is not interested anymore in this event and remove it.
5. UniStud does not show the event anymore on the Jim's HomePage.
6. Jim explore other contents to find something that is on his interest.

Scenario 7

Scenario Name: FindTutorialsOfferedOnUniStud

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is on the HomePage of the application.
2. Jim decides to see a tutorial for Java, so he open the navigation drawer and selects Tutorials.
3. UniStud provides Jim all the courses offered in the platform with some suggested tutorials on top.
4. User Jim search for Java tutorials and filter the tutorials accordingly.
5. Jim decides to watch a tutorial which will be live after 6 hours and the uploader has great rating.
6. Jim adds it into the saved tutorials.

Scenario 8

Scenario Name: FindEventOfferedOnUniStud

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is on the HomePage of the application.
2. Jim decides to find an upcoming event happening nearby in campus and opens navigation drawer and selects Events.
3. UniStud provides Jim information about different events happening nearby.
4. User Jim search for Games events and filter event accordingly.
5. Jim decides to go to an event where people are going to play Counter Strike.
6. UniStud save this information for Jim and notifies him at a reasonable time before the event is about to start.
7. Jim gets notification and prepares to go to the event.
8. Event is added into automatically into events lists of Jim.
9. Event is deleted from saved list of Jim after the deadline has passed. It will also be deleted from the list of events.

Scenario 9

Scenario Name: SellAnAcademicResourceOnUniStud

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is on the HomePage of the application.
2. Jim decides to sell a book since he does not want it anymore and needs some money. He opens navigation drawer and selects UniTrade.
3. User Jim select the Sell category.
4. User Jim enters the information of the book including the location, by using the current location or choosing on map.
5. UniStud processes all the information added by Jim and adds the book with a specific color into the list of items to be sold.

Scenario 10

Scenario Name: SearchAnAcademicResourceOnUniStud

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is on the HomePage of the application.
2. Jim decides to buy a book since it is very important for him but he does not want to spent too much money. He opens navigation drawer and selects UniTrade.
3. User Jim select the Buy category.
4. UniStud provides information about all the books that are being sold into a reasonable distance range based from Jim current position.
5. Jim refines the specification by increasing the distance and adding some features for the book.
6. UniStud updates the list of the books.
7. Jim finds the book he was looking for and get the seller contact details.
8. Jim add the book into his saved list in order to remember it.

Scenario 11

Scenario Name: CreateTutorial

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is on the HomePage of the application.
2. Jim feels very confident for a course and thinks that he can help other students by offering them online or private tutorials from which he can also gain some money. Jim opens navigation drawer and selects Tutorials.
3. Jim selects the option to insert new tutorial.
4. Jim fills the necessary information for the tutorial such as the course, topic and date of display if he will use livestream, otherwise he also adds the price per hour, contact information and the available schedule.
5. UniStud adds the tutorial in the list of all tutorials and also in the profile of the user.

Scenario 12

Scenario Name: GiveOnlineTutorial

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim has already completed *OfferTutorial* scenario.
2. Jim gets a notification from UniStud 10 minutes prior the livestream starts.
3. Jim opens the notification and is redirected to the tutorial section where he will have an option to start the livestream.
4. Jim gives the lecture at the scheduled time.
5. Jim closes the livestream when the class is over.
6. UniStud saves the lecture in the list of tutorials and also in the profile of the user who gave the class.

Scenario 13

Scenario Name: WatchOnlineTutorial

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim needs to improve his skills and knowledge in a certain course. Jim completes *FindTutorialsOfferedOnUniStud* scenario.
2. Jim is notified 10 minutes before the livestream.
3. Jim opens UniStud and in his homepage, he has a button which he presses in order to start following the stream.
4. Jim can take notes and can ask questions in real time in order to understand a certain part of the lecture.
5. When stream ends the video is added in the tutorials list of Jim.

Scenario 14

Scenario Name: SearchUniversity

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is interested in searching for a university in Europe. He opens navigation drawer and selects Opportunities.
2. Search for universities will be automatically selected over internships.
3. Most searched universities will be listed. Jim can check any of these universities or he can use the search bar and filter according to his preferences.
4. Jim will be provided a list of universities that fulfill his preferences. He can click on any of them and see more detailed information plus an option to see a list of students in the UniStud platform that are currently studying there.

Scenario 15

Scenario Name: ContactUniversityStudent

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim want to be provided with real information about the requirements and life costs for a university in Turkey. He completes *SearchUniversity* scenario.
2. Jim selects the option to see the list of students currently studying in Turkey.
3. Jim selects to contact one student from the list who has been studying in Turkey for three years.
4. Jim writes his message and sends it.
5. A new message is added in the list of messages of Jim and of the person who received it.

Scenario 16

Scenario Name: SeeMessages

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim is contacted from a student who wants to ask him how is it to study in Portugal.
2. Jim gets a notification about the message, but does not open it right away.
3. He opens UniStud later and sees a badge in the messages button reminding him he has an unread message.
4. Jim selects the messages button and is redirected to the messages page.
5. Jim selects the top message in the list, because that one is the unread one.
6. Jim reads the message and replies to the students' questions.

Scenario 17

Scenario Name: SearchInternships

Participating actor instances: Jim: NormalUser

Main flow of events:

1. Jim has to complete an internship during the upcoming summer. He opens navigation drawer and selects Opportunities.
2. Search for universities will be automatically selected over internships so Jim has to select Internships.
3. Jim is provided with most searched Internships and some Internships that might interest him according to his department.
4. Jim can also use search bar and filter to search for internships according to his preferences such as country and department.
5. Jim seems to be interested in one of them so he clicks it and more detailed information is provided.

Scenario 18

Scenario Name: CreateInternship

Participating actor instances: Ali: GuestUser

Main flow of events:

1. Ali is in charge of managing the account for an organization or company. He wants to offer the internship opportunities to the students of UniStud.
2. Ali opens navigation drawer and selects opportunities. Since he is a guest user he is redirected to Internships.
3. Ali selects the button *add internship.*
4. Ali completes the necessary information about the internship such as company, location, department, salary (if it has), position, time period.
5. Ali presses add button and Internship is added in the list of Internships.

Scenario 19

Scenario Name: CreateEvent

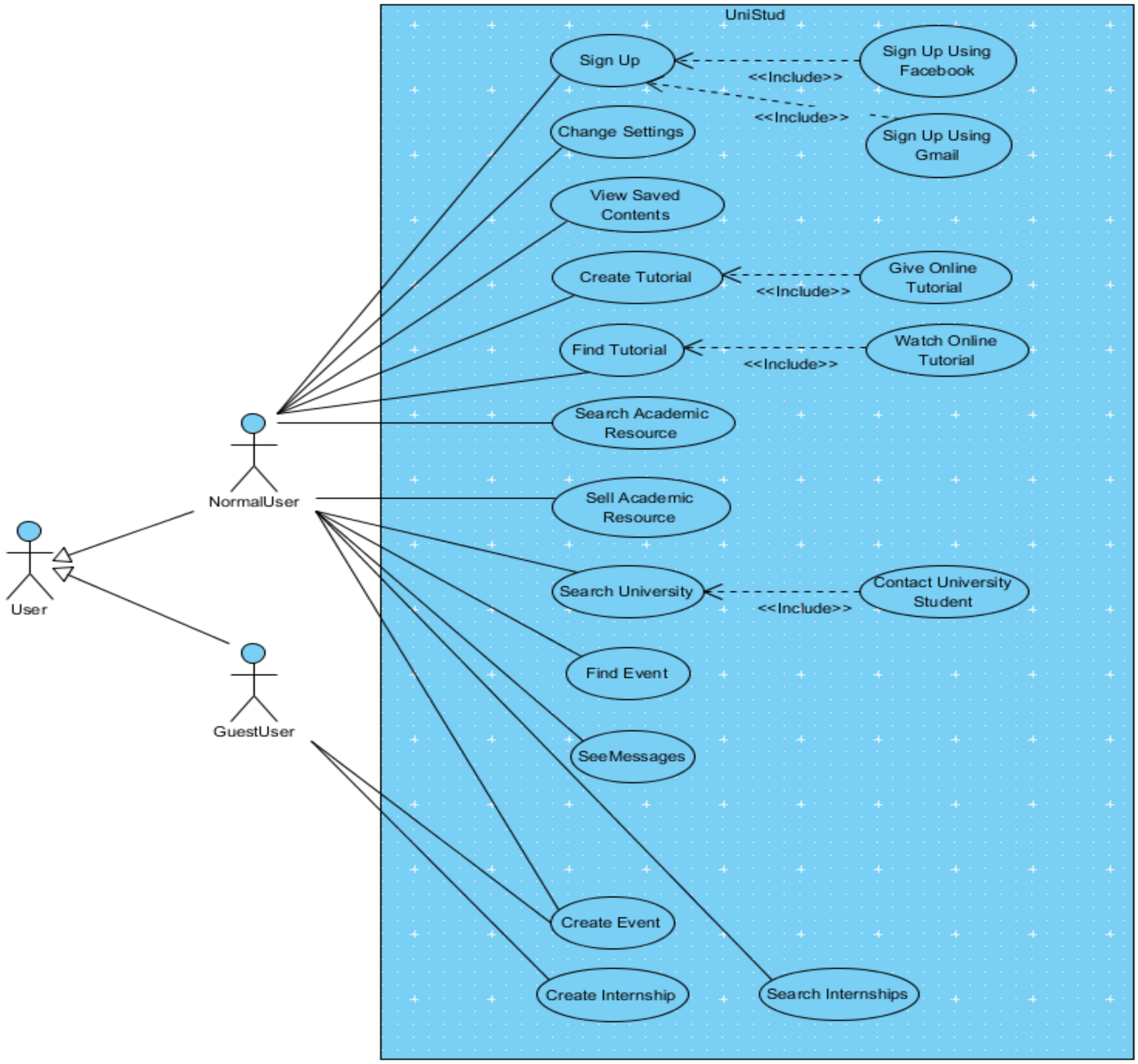
Participating actor instances: Ali: GuestUser

Main flow of events:

1. Ali opens navigation drawer and selects events.
2. Ali selects the option to create a new event.
3. Ali fills the necessary information about the event such as date, place, topic, duration and a short description.
4. Ali selects *Add Event* button and event is added to the list of events.

3.5.2 Use Case Model

In the following section, we will introduce use-case models of our system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal [6]. This is important for showing our user-system interactions. Log In is required for all use cases except Sign Up, but is omitted from the diagram for simplicity.



3.5.2.1 Login Use Case

Use Case Name: Login

Actor: Jim

Entry Condition:

- User is on Login Screen.

Exit Condition:

- User is on HomePage screen.
- Authentication Failed and user is on Login screen.
- User change to SignUp screen.

Main flow of events:

1. User enters his email.
2. User enter his password.
3. UniStud checks if the email and password are correct.
4. UniStud confirms the credentials of the user.
5. UniStud redirect user to HomePage.

Alternative flow of events:

1. User enters wrong email and/or password.
    a. UniStud detects that the credentials are incorrect.
    b. UniStud redirects user to Login page
    c. UniStud notifies user about the wrong entries.
2. User taps “Sign In With Facebook” button
    a. UniStud checks for the user’s Facebook credentials.
    b. UniStud redirects user to HomePage
3. User taps “Sign In With Google” button
    a. UniStud checks for the user’s Google credentials.
    b. UniStud redirects user to HomePage.

3.5.2.2 Create Account Use Case

Use Case Name: Create Account

Actor: Jim

Entry Condition:

- User is on Login Screen.

Exit Condition:

- User is on HomePage screen.
- User change to Login screen.

Main flow of events:

1. User clicks “Sign Up” button
2. UniStud redirects user to Sign Up page.
3. User enters preferred username
4. User enters email address.
5. User enters password twice.
6. User clicks “Register” button.
7. UniStud checks if the username is available.
8. UniStud checks if there is a user with the given email address.
9. UniStud creates a new account for the user.
10. UniStud redirects user to HomePage.

Alternative flow of events:

1. User inputs are inaccurate or missing.
    a. UniStud empties the inaccurate forms and redirect the SignUp page.
    b. UniStud notifies the user about the inaccurate forms properly.
    c. User enters the inaccurate information again and step 8-11 are repeated.

3.5.2.3 Create Account Using Facebook Use Case

Use Case Name: Create Account

Actor: Jim

Entry Condition:

- User is on Login Screen

Exit Condition:

- User is on HomePage screen.
- User change to Sign Up screen.

Main flow of events:

1. User clicks “Sign In with Facebook” button.
2. User enters email address.
3. User enters password.
4. UniStud checks if there is a user with the given email address.
5. UniStud creates a new account for the user.
6. UniStud redirects user to HomePage.

Alternative flow of events:

1. User has already logged into Facebook and does not own any account.
    a. User clicks “Sign In With Facebook” button.
    b. UniStud checks if there is a user with the given email address.
    c. UniStud creates a new account for the user.
    d. UniStud redirects user to HomePage.
2. User has already logged in using Facebook and owns an account.
    a. User clicks “Sign In With Facebook” button.
    b. UniStud checks if the credentials are correct.
    c. UniStud redirects user to HomePage.

3.5.2.4 Create Account Using Google Use Case

Use Case Name: Create Account

Actor: Jim

Entry Condition:

- User is on Login Screen.

Exit Condition:

- User is on HomePage screen.
- User change to Sign Up screen.

Main flow of events:

1. User clicks “Sign In With Google” button.
2. User enters email address.
3. User enters password.
4. UniStud checks if there is a user with the given email address.
5. UniStud creates a new account for the user.
6. UniStud redirects user to HomePage.

Alternative flow of events:

1. User has already logged into Google and does not own any account.
    a. User clicks “Sign In With Google” button.
    b. UniStud checks if there is a user with the given email address.
    c. UniStud creates a new account for the user.
    d. UniStud redirects user to HomePage.
2. User has already logged in using Google and owns an account.
    a. User clicks “Sign In With Google” button.
    b. UniStud checks if the credentials are correct.
    c. UniStud redirects user to HomePage.

3.5.2.5 Change Settings Use Case

Use Case Name: Change Settings

Actor: Jim

Entry Condition:

- *"Change Settings"* Use Case includes *"Login"* Use Case.
- User is on Profile Page.

Exit Condition:

- User is on HomePage screen.
- User Logout from the application.

Main flow of events:

1. User is on Profile Page.
2. User clicks on the gear icon.
3. User navigates thought all preferences.
4. User change the preferences accordingly.
5. User clicks save preferences.
6. UniStud saves the new preferences of the user.
7. UniStud redirects user to Profile Page.

Alternative flow of events:

1. User do not change any preference.
    a. UniStud skip step 6 and executes step 7.
2. User enters a username that currently is taken.
    a. UniStud checks that username is taken.
    b. UniStud notifies the username that the username is currently taken.

3.5.2.6 View Saved Content Use Case

Use Case Name: View Saved Content

Actor: Jim

Entry Condition:

- User is on HomePage.

Exit Condition:

- User change open navigation drawer and go to another page.
- User selects one of saved items and go to the corresponding page.
- User Logout from the application.

Main flow of events:

1. User scrolls through the saved list.
2. User clicks one of the items saved.
3. UniStud extends the area of the item and give general information about the item
4. User wants to get further information and go the item corresponding page.

Alternative flow of events:

1. User has a message from University section.
    a. UniStud shows the message at top of saved list.
    b. User clicks the notification for the message.
    c. UniStud redirect user to the message page of the University section.
2. User is notified for a starting tutorial.
    a. UniStud shows the tutorial notification at top of saved list.
    b. User clicks the notification for the starting tutorial.
    c. UniStud redirect user to the live stream of the Tutorial.

3.5.2.7 Find Tutorials Offered On UniStud Use Case

Use Case Name: Find Tutorials Offered On UniStud

Actor: Jim

Entry Condition:

- User is on Tutorial Page.

Exit Condition:

- User change open navigation drawer and go to another page.
- User Logout from the application.

Main flow of events:

1. User inserts a keyword in the search box.
2. User presses search button.
3. User presses *filters* button.
4. User changes filters according to his preferences.
5. User presses submit filters.
6. User is displayed a list of tutorials according to the filters.

Alternative flow of events:

1. User enters non-existing keyword.
    a. No list is provided.
    b. User is asked to add another keyword.

3.5.2.8 Find Events Offered On UniStud Use Case

Use Case Name: Find Events Offered On UniStud

Actor: Jim

Entry Condition:

- User is on Events Page.

Exit Condition:

- User change open navigation drawer and go to another page.
- User Logout from the application.

Main flow of events:

1. User inserts a keyword in the search box.
2. User presses search button.
3. User presses *filters* button.
4. User changes filters according to his preferences.
5. User presses submit filters.
6. User is displayed a list of events according to the filters.

Alternative flow of events:

1. User enters non-existing keyword.
    a. No list is provided.
    b. User is asked to add another keyword.

3.5.2.9 Sell Academic Resource Use Case

Use Case Name: Sell Academic Resource

Actor: User

Entry Condition:

- User is on UniTrade screen.

Exit Condition:

- User closes the program.
- User adds the item to the list.

Main flow of events:

1. User selects the + button to add a new item which can be book, magazine or other study material.
2. User fills the necessary information about the item.
3. User presses *Add* button and the item is added to the list.

Alternative flow of events:

1. User has to exit from application before adding the item.
    a. User has to fill the information from beginning again next time.

3.5.2.10 Search Academic Resource Use Case

Use Case Name: Search Academic Resource

Actor: User

Entry Condition:

- User is on UniTrade screen.

Exit Condition:

- User closes the program.
- User save the item to the list.

Main flow of events:

1. User enters the keyword and presses *Search* button.
2. User presses *Filters* button.
3. User completes filters according to his needs.
4. User presses *Submit Filters*
5. User is displayed a list of items according to his preferences.
6. User presses *Save* button and the item is added to his list.

Alternative flow of events:

1. User has to exit from application before adding the item.
    a. User has to fill the information in filters from beginning again next time to find the same item.

3.5.2.11 Create Online Tutorial Use Case

Use Case Name: Create Online Tutorial

Actor: Jim

Entry Condition:

- User is on Tutorials screen.

Exit Condition:

- User closes the program.
- User adds the tutorial to the list by pressing *Add* button.

Main flow of events:

1. User selects the + button to add a new tutorial.
2. User fills the necessary information about the tutorial such as course, topic and date.
3. User selects *online* as option for the tutorial.
4. User presses *add* button and the tutorial is added to the list.

Alternative flow of events:

1. User has to exit from application before adding the tutorial.
    a. User has to fill the information from beginning again next time.

3.5.2.12 Give Online Tutorial Use Case

Use Case Name: Give Online Tutorial

Actor: Jim

Entry Condition:

- User has already completer *Offer Tutorial* Use Case.
- User selects the notification that arrives before the livestream starts.
- User selects the option to start the livestream.

Exit Condition:

- User closes the program.
- The lecture and livestream end.

Main flow of events:

1. User gets notified about the lecture 10 minutes prior.
2. User opens UniStud.
3. User selects the livestream event from saved list in his homepage.
4. User selects the option to open the livestream for the lecture.
5. User gives the lecture and answers the questions.

Alternative flow of events:

1. User has to exit from the lecture at half-way.
    a. User can schedule another time and give the same lecture.
2. User misses the time to give the lecture.
    a. User can create another online tutorial event. (Rating will be deduced accordingly)

3.5.2.13 Watch Online Tutorial Use Case

Use Case Name: Watch Online Tutorial

Actor: Jim

Entry Condition:

- User is on Tutorial page.
- User selects the notification that arrives when the livestream starts.
- User selects the tutorial from saved list in Home screen.

Exit Condition:

- User closes the program.
- The livestream ends.

Main flow of events:

1. User gets notified about the lecture.
2. User opens UniStud.
3. User opens the livestream for the lecture.
4. User follows the lecture and asks questions to the lecturer in the form of livestream comments.

Alternative flow of events:

1. User exits from the lecture at half-way.
    a. User can still see the remaining part later.
2. User misses the lecture.
    a. User can watch the lecture any time.

3.5.2.14 Search University Use Case

Use Case Name: Search University

Actor: Jim

Entry Condition:

- User is on Opportunities page and University is selected.

Exit Condition:

- User goes to another screen from the navigation drawer.
- User logs out from the application.
- User closes the application.

Main flow of events:

1. User inserts a keyword in the search box.
2. User presses search button.
3. User presses *filters* button.
4. User changes filters according to his preferences.
5. User presses submit filters.
6. User is displayed a list of universities according to the filters.

Alternative flow of events:

1. User enters non-existing keyword.
    a. No list is provided.
    b. User is asked to add another keyword.

3.5.2.15 Contact University Students Use Case

Use Case Name: Contact University Students

Actor: Jim

Entry Condition:

- User is on Opportunities page and University is selected.

Exit Condition:

- User sends message to one of the students and returns to homepage.
- User logs out from the application.
- User closes the application.

Main flow of events:

1. User inserts a keyword in the search box.
2. User presses search button.
3. User presses *filters* button.
4. User changes filters according to his preferences.
5. User presses submit filters.
6. User is displayed a list of universities according to the filters.
7. User selects a university.
8. User presses *list students* and a list of students currently studying there is displayed.
9. User presses the message icon for one of the students.
10. User sends his message.

Alternative flow of events:

1. User enters non-existing keyword.
    a. No list is provided.
    b. User is asked to add another keyword.
2. No student from UniStud currently enrolled in the selected university.
    a. User will have to refer only to online information.

3.5.2.16 See Messages Case

Use Case Name: See Messages

Actor: Jim

Entry Condition:

- User is on Home screen.

Exit Condition:

- User selects another screen from the navigation drawer.
- User closes the application.

Main flow of events:

1. User selects messages button.
2. User is displayed the list of users he is currently contacting.
3. User selects one of them.
4. User responds to message.

Alternative flow of events:

1. User decides to go back without replying.
    a. Message is marked on read.

3.5.2.17 Search Internship Case

Use Case Name: Search Internship

Actor: Jim

Entry Condition:

- User is on Opportunity Screen.

Exit Condition:

- User selects another screen from the navigation drawer.
- User logs out or closes the application.

Main flow of events:

1. User selects Internships.
2. User inserts a keyword in the search box.
3. User presses search button.
4. User presses *filters* button.
5. User changes filters according to his preferences.
6. User presses submit filters.
7. User is displayed a list of internships according to the filters.

Alternative flow of events:

1. User enters non-existing keyword.
    a. No list is provided.
    b. User is asked to add another keyword.

3.5.2.18 Create Internship Case

Use Case Name: Create Internship

Actor: User

Entry Condition:

- User is on Opportunity Screen.

Exit Condition:

- User selects another screen from the navigation drawer.
- User logs out or closes the application.

Main flow of events:

1. User selects Internships.
2. User inserts a keyword in the search box.
3. User presses search button.
4. User presses *filters* button.
5. User changes filters according to his preferences.
6. User presses submit filters.
7. User is displayed a list of internships according to the filters.

Alternative flow of events:

1. User enters non-existing keyword.
    a. No list is provided.
    b. User is asked to add another keyword.

3.5.2.19 Create Event Case

Use Case Name: Create Event

Actor: User

Entry Condition:

- User is on Events Screen.

Exit Condition:

- User selects another screen from the navigation drawer.
- User logs out or closes the application.

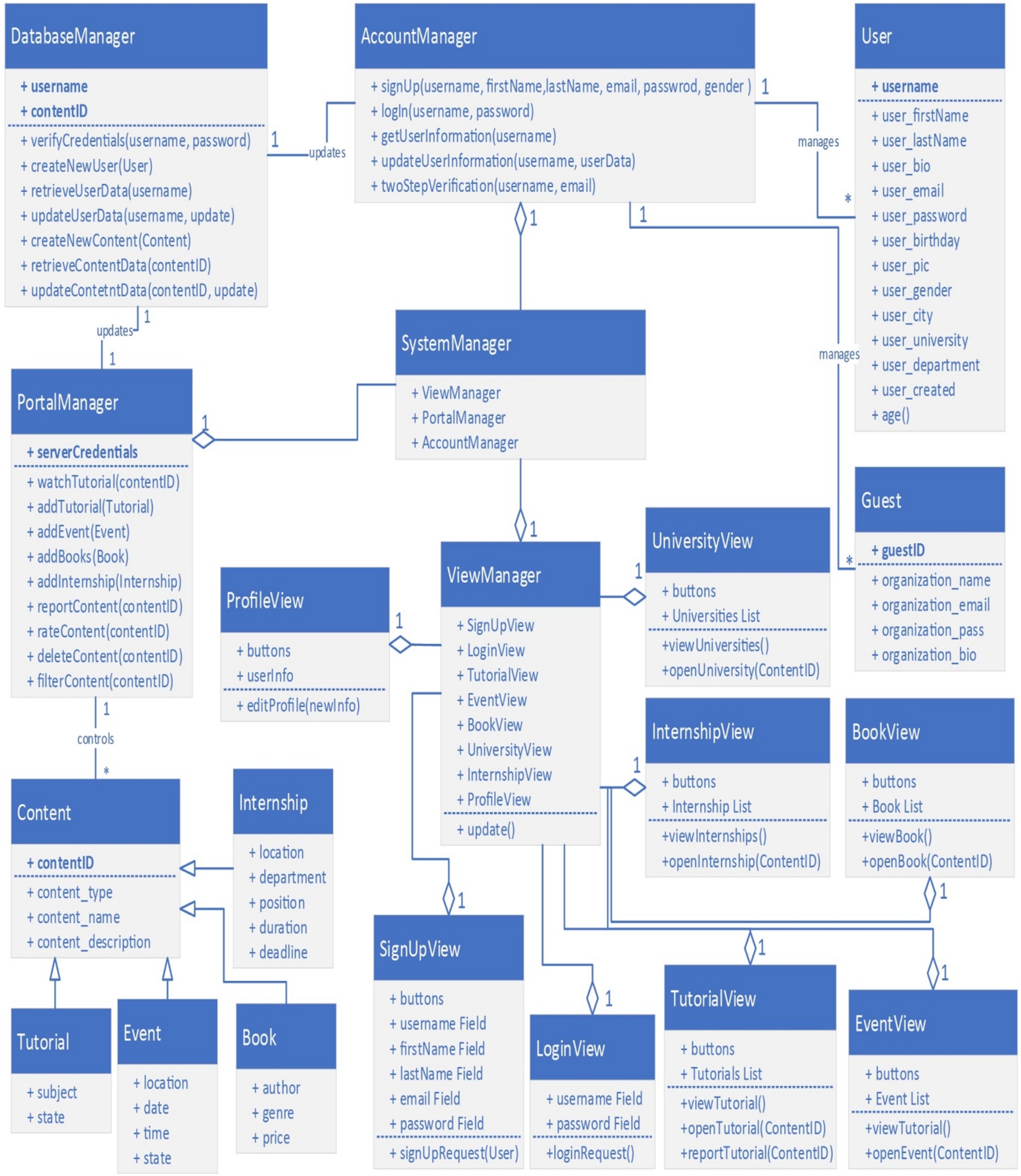
Main flow of events:

1. User presses *Create Events.*
2. User fills the necessary information about the event.
3. User presses *Add Event.*
4. Event is added to the list of events.

Alternative flow of events:

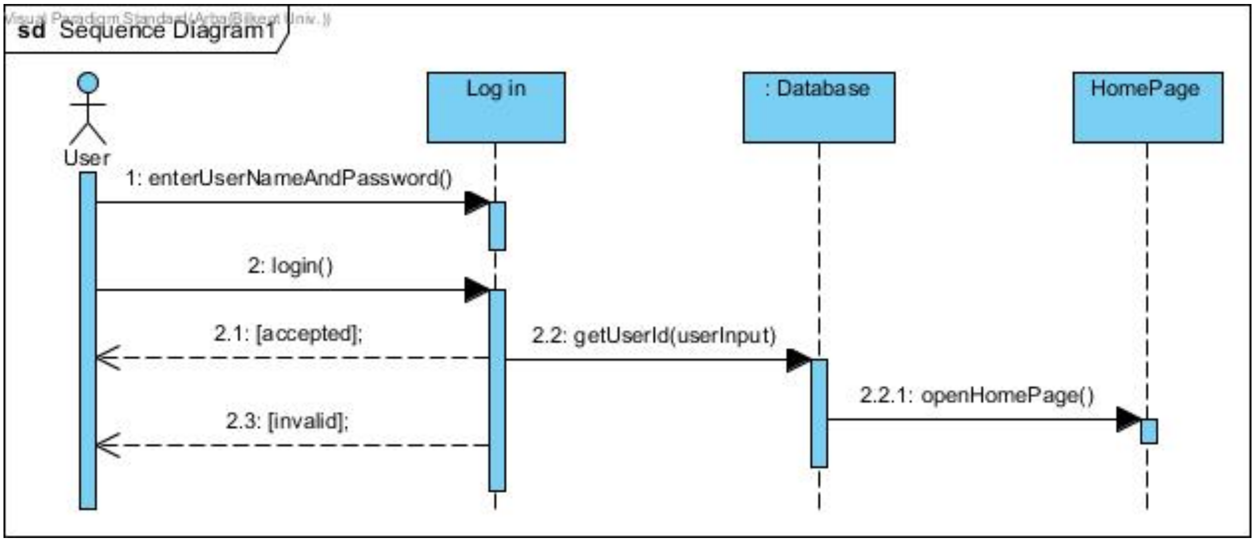
1. User enters bad information.
    a. No event is created, user is asked to insert correct information.

3.5.3 Object and Class Model

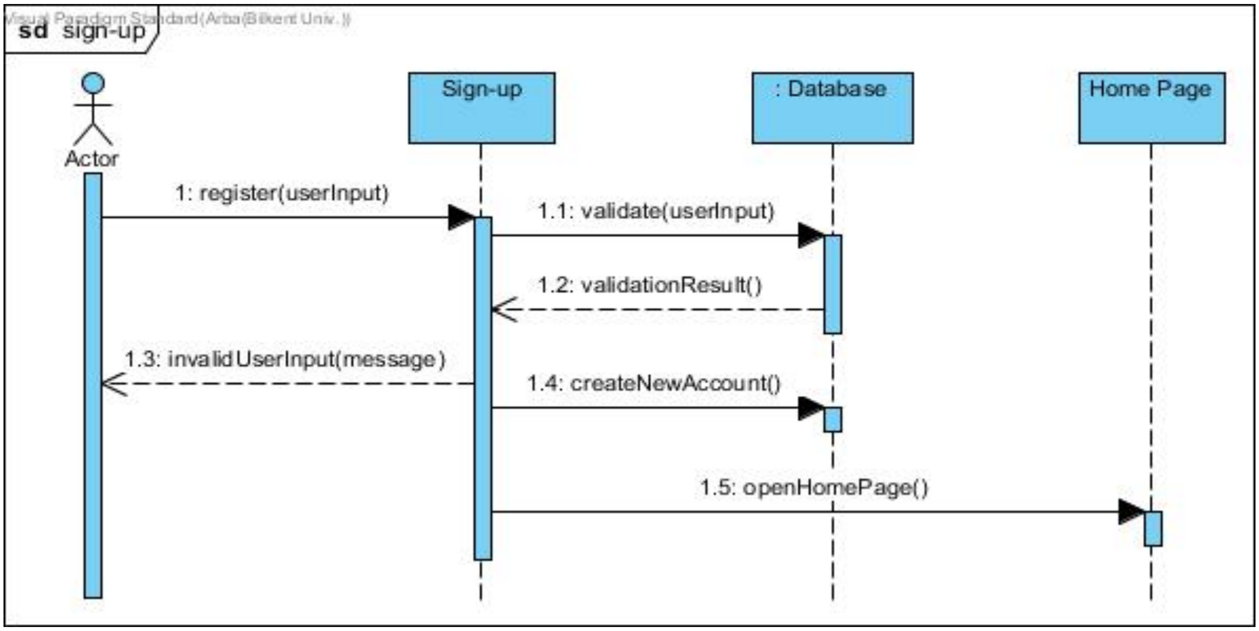


3.5.4 Dynamic Model

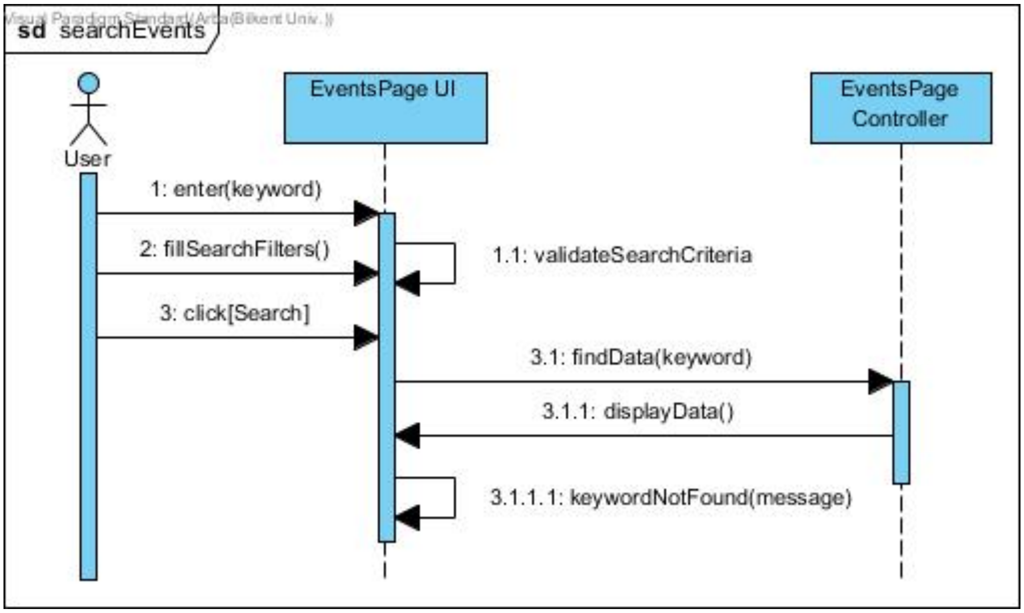
3.5.4.1 Login Sequence Diagram



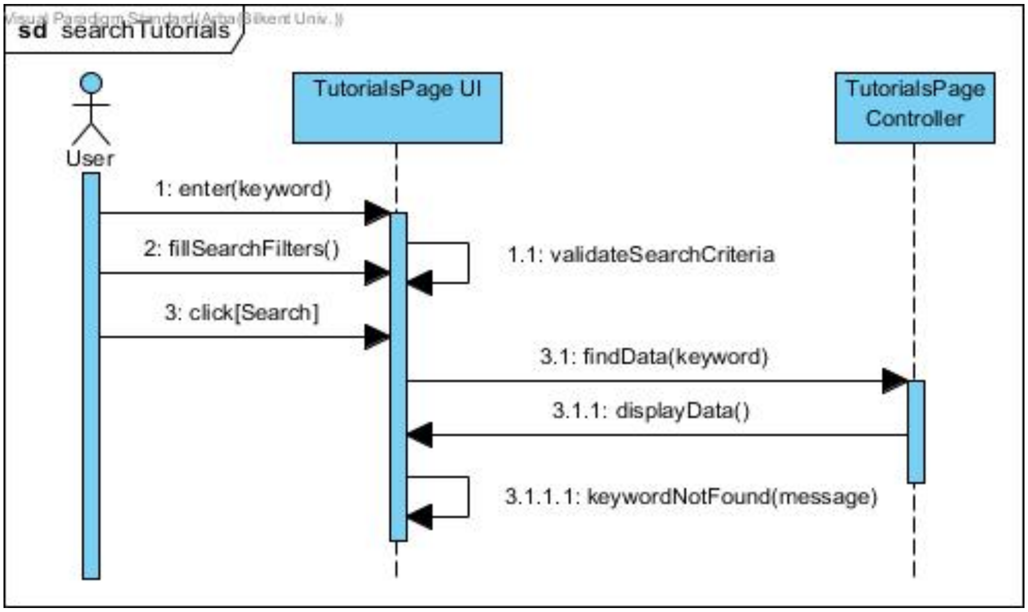
3.5.4.2 Sign-up Sequence Diagram



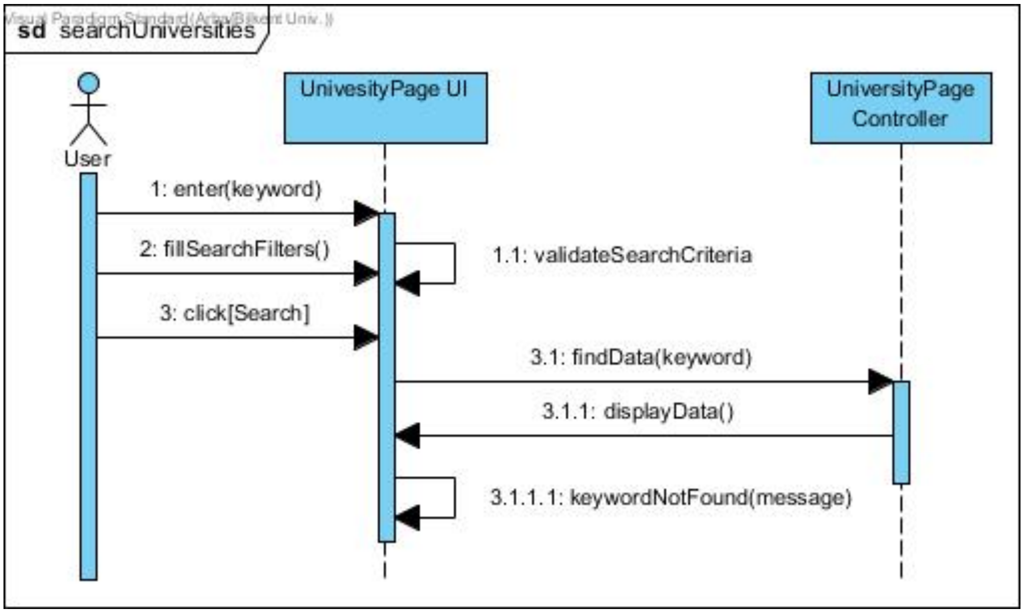
3.5.4.3 Search Events Sequence Diagram



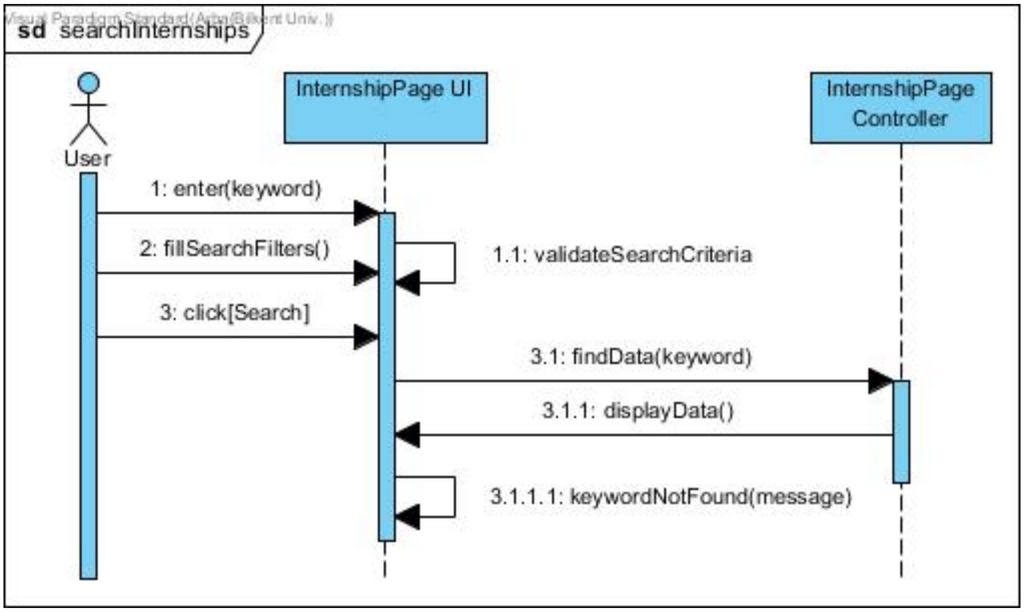
3.5.4.4 Search Tutorials Sequence Diagram



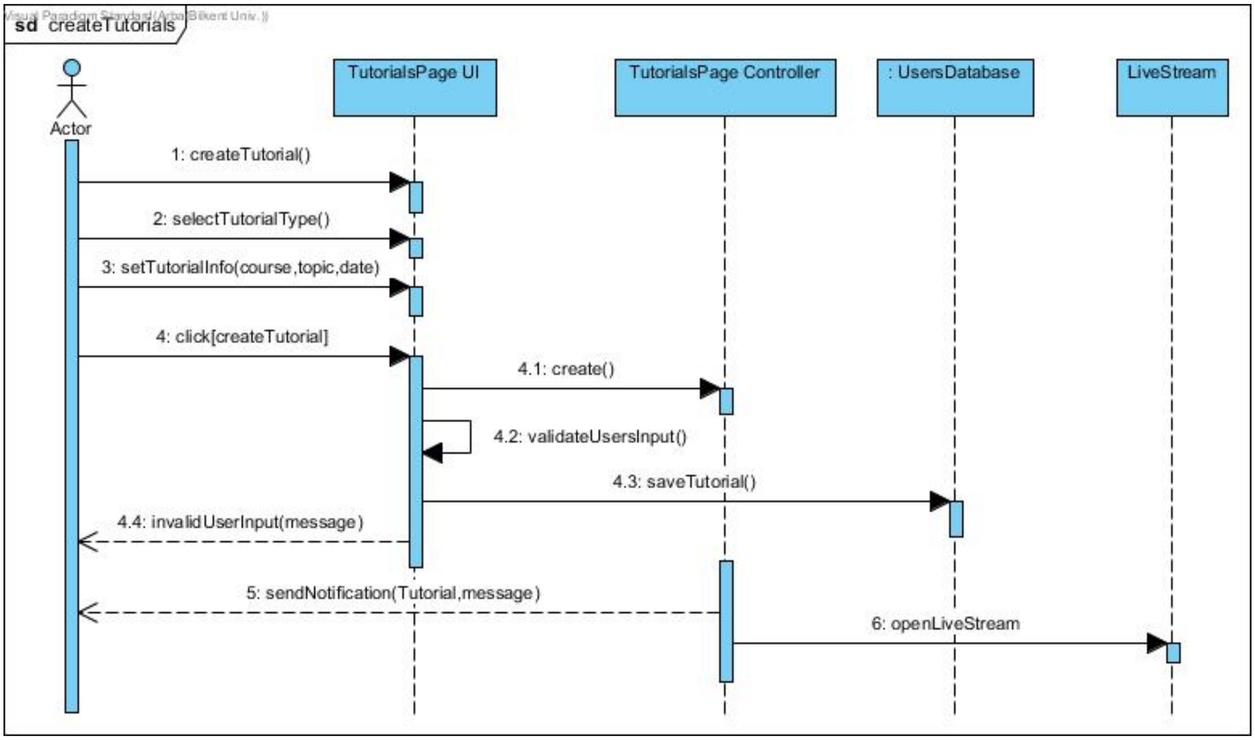
3.5.4.5 Search Universities Sequence Diagram



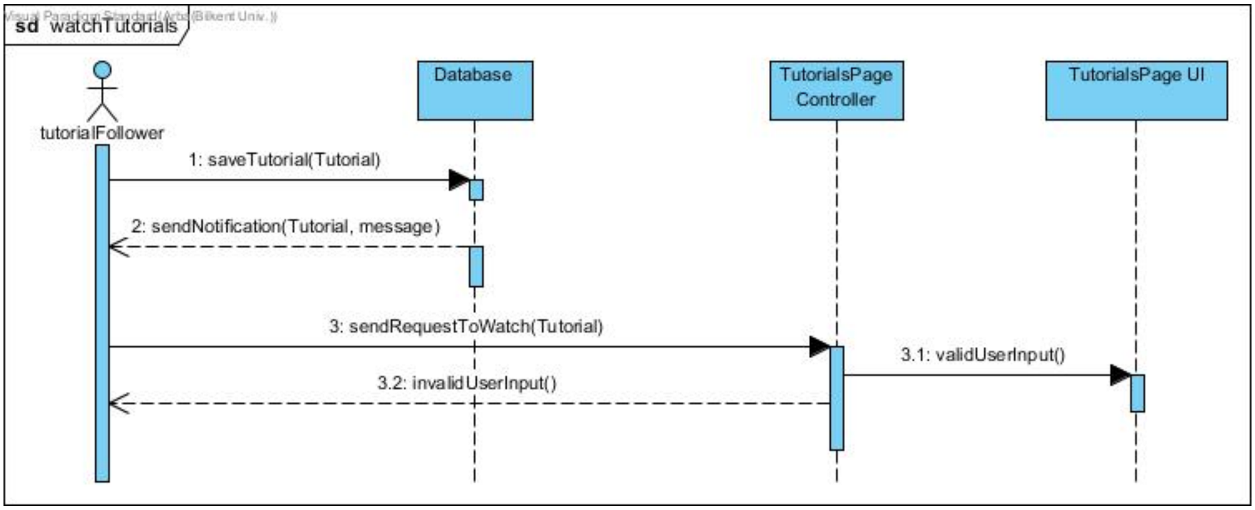
3.5.4.6 Search Internship Sequence Diagram



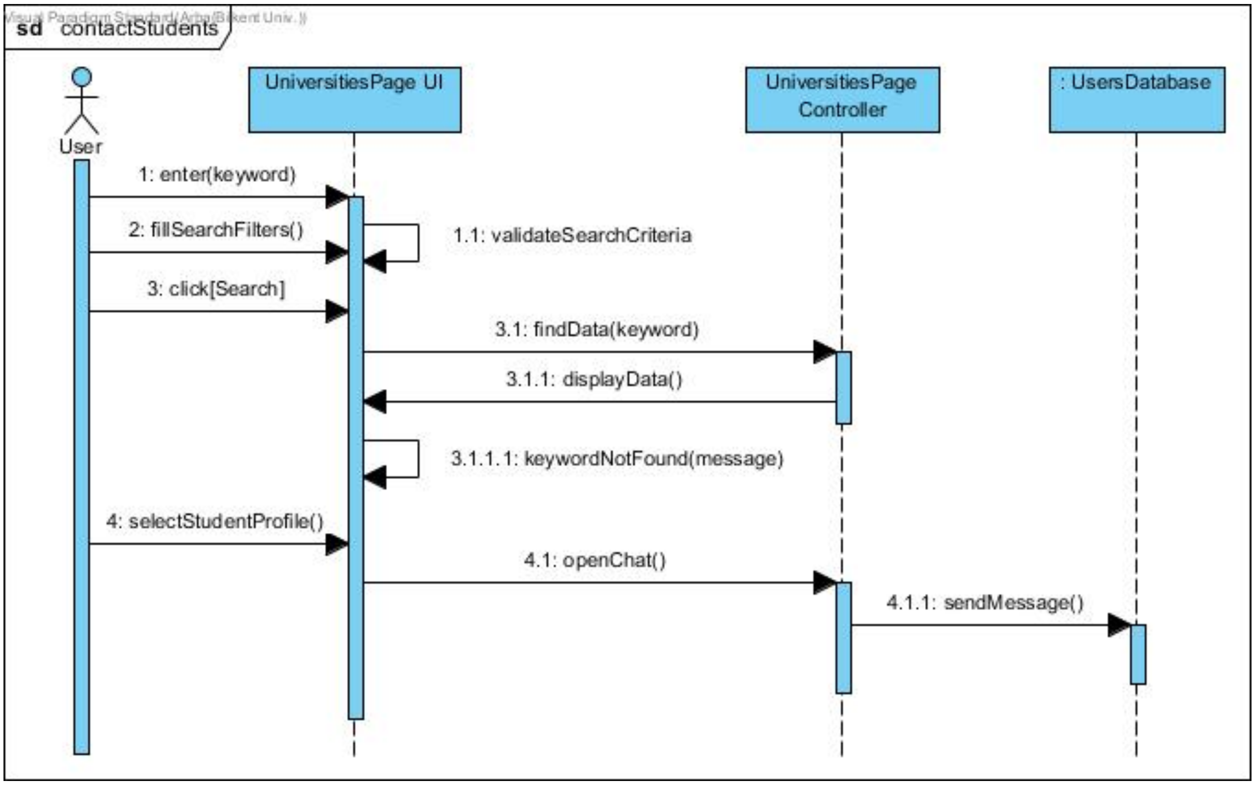
3.5.4.7 Create Tutorials Sequence Diagram



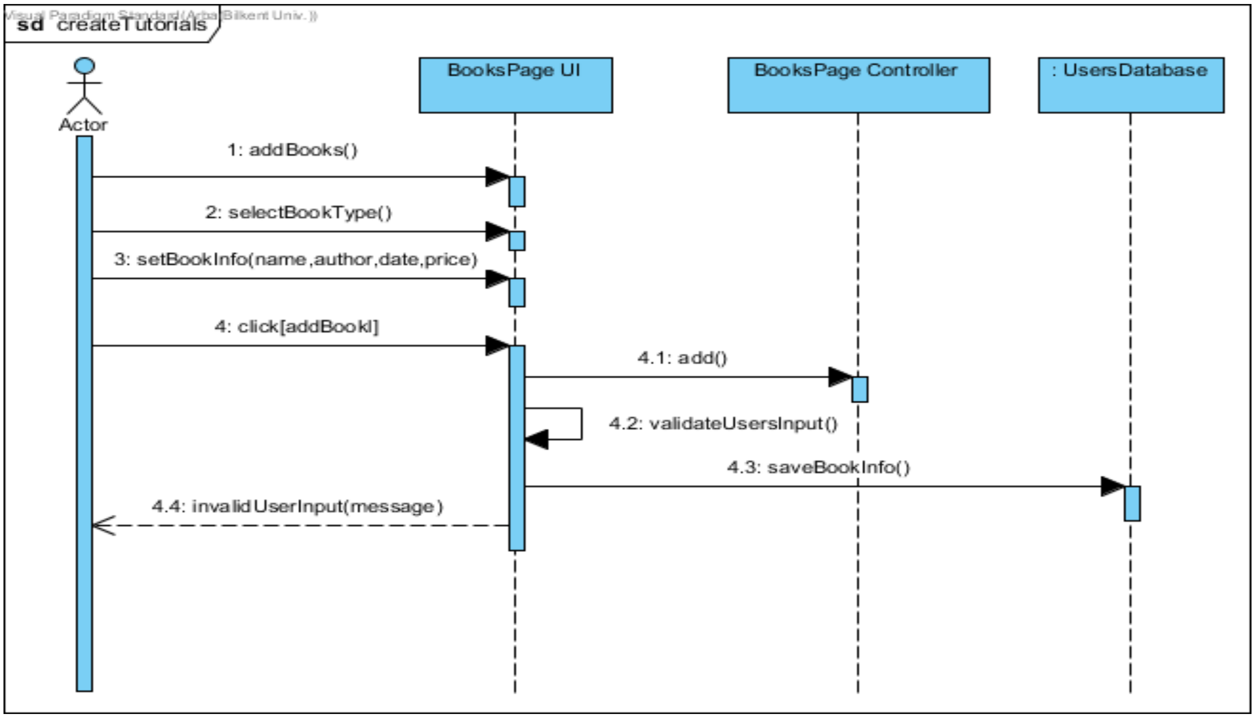
3.5.4.8 Watch Tutorials Sequence Diagram



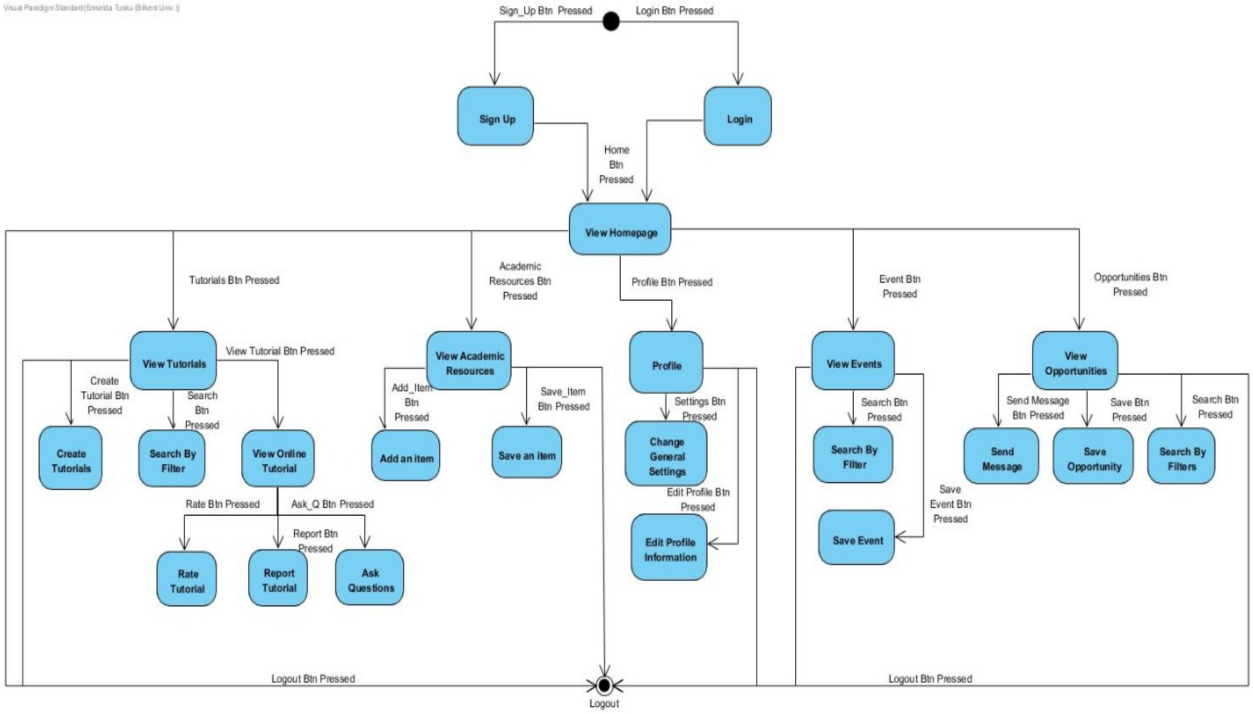
3.5.4.9 Contact Students Sequence Diagram



3.5.4.10 Sell Books Sequence Diagram



3.5.4.11 Application Flow Activity Diagram

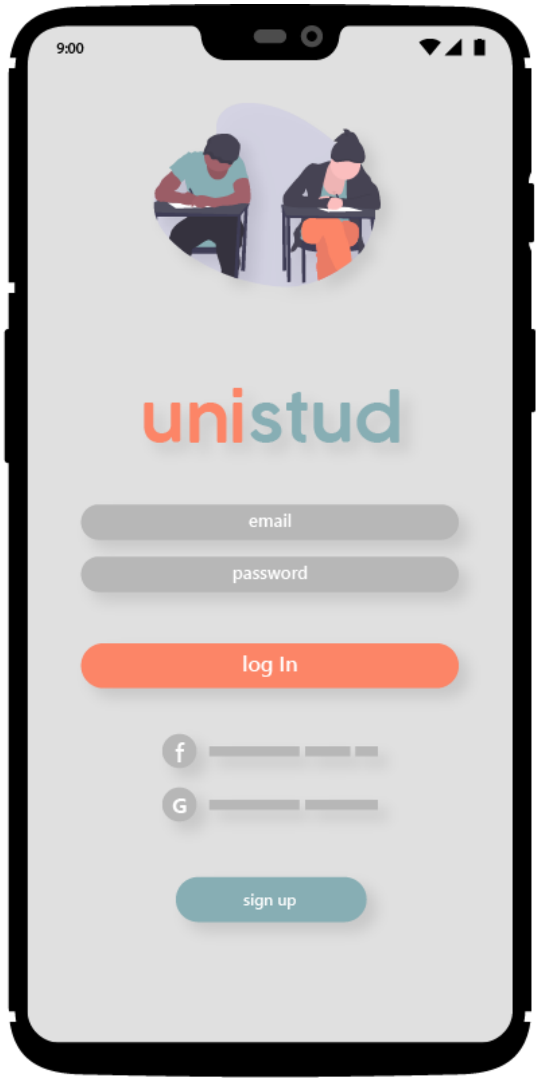


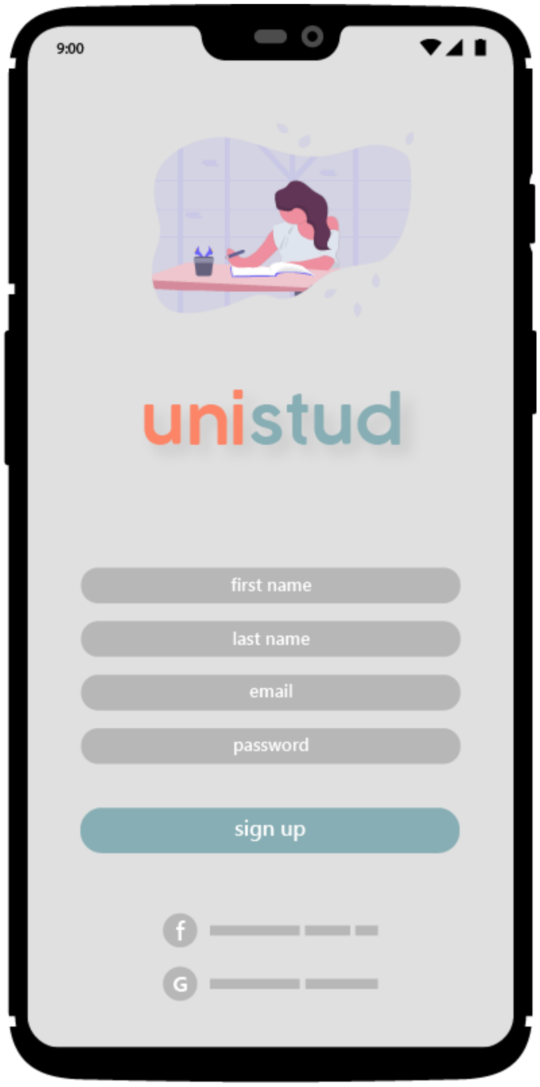
3.5.5 User Interface

3.5.5.1 Login and Sign Up Screen

The first contact with the user will be the Login page in which the user, if he/she is already registered in the platform, can insert information such as email and password and enter into his/her account.

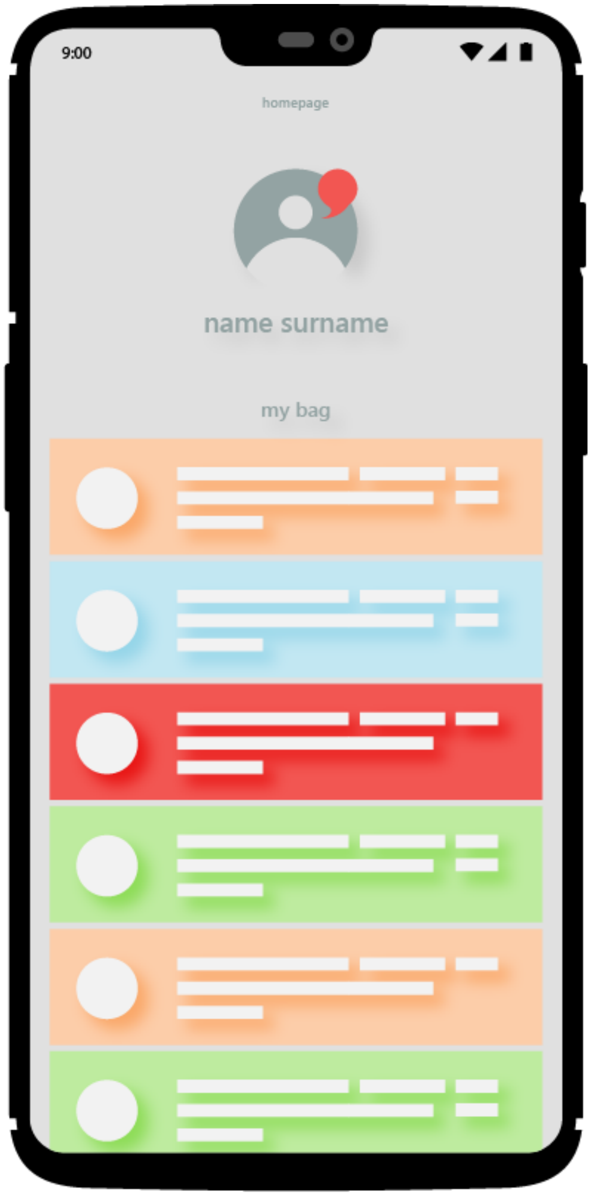
However, if the user has not registered yet, he will deal with the Sign-Up page in which he must enter all the required information in order to open an account in the platform. The user can register directly also by using his Gmail or Facebook account





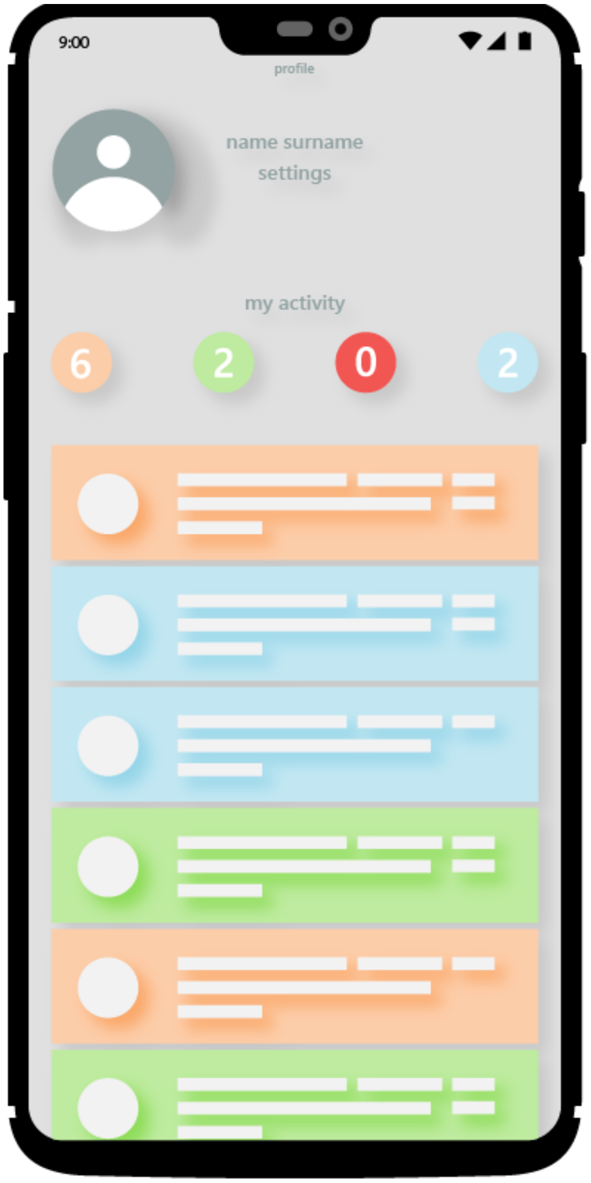
3.5.5.2 HomePage Screen

In the Homepage, the user will be able to see a list of different item from different categories that he has considered of interest and has saved them. To facilitate the distinction each category is represented from a unique color making it easier for the user to keep track.



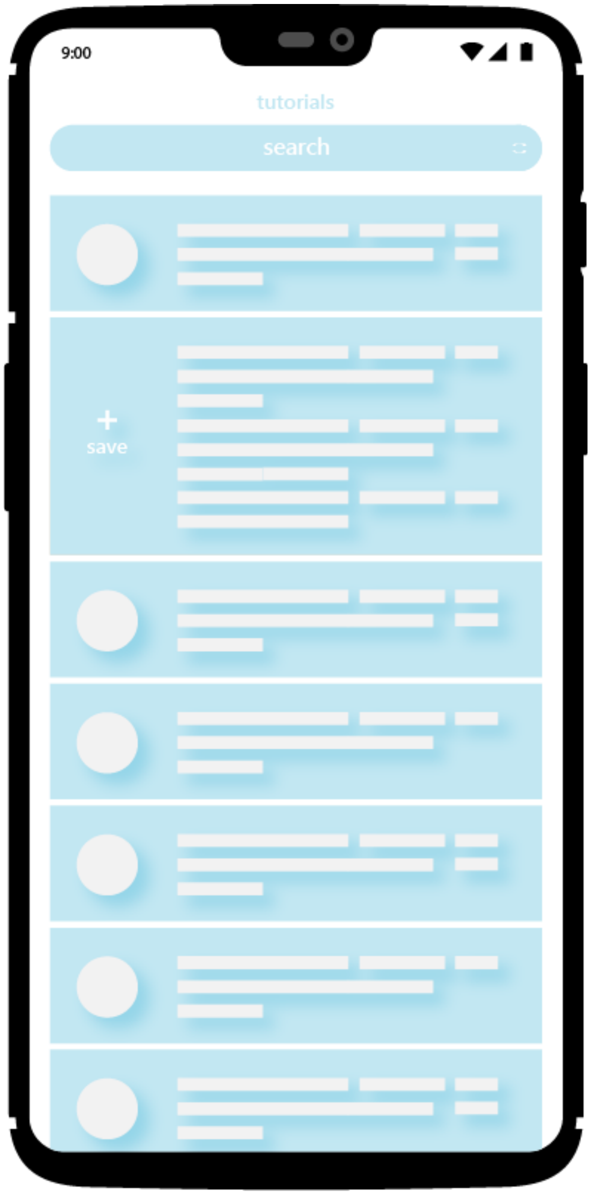
3.5.5.3 Profile Screen

In the profile page the user can see a part of his personal information displayed together with a picture. By pressing “Edit Profile” he is able to see his full information and also edit it. In the profile page a record of the user’s activity from each category will also be displayed showing the academic resources he purchased, the tutorials he followed etc. In order to see the activity belonging to only a specific category the user can go to one of the buttons that will be displayed on the top of the page that will show the number of items his activity involved in each category. By clicking it the items from only that category will be displayed



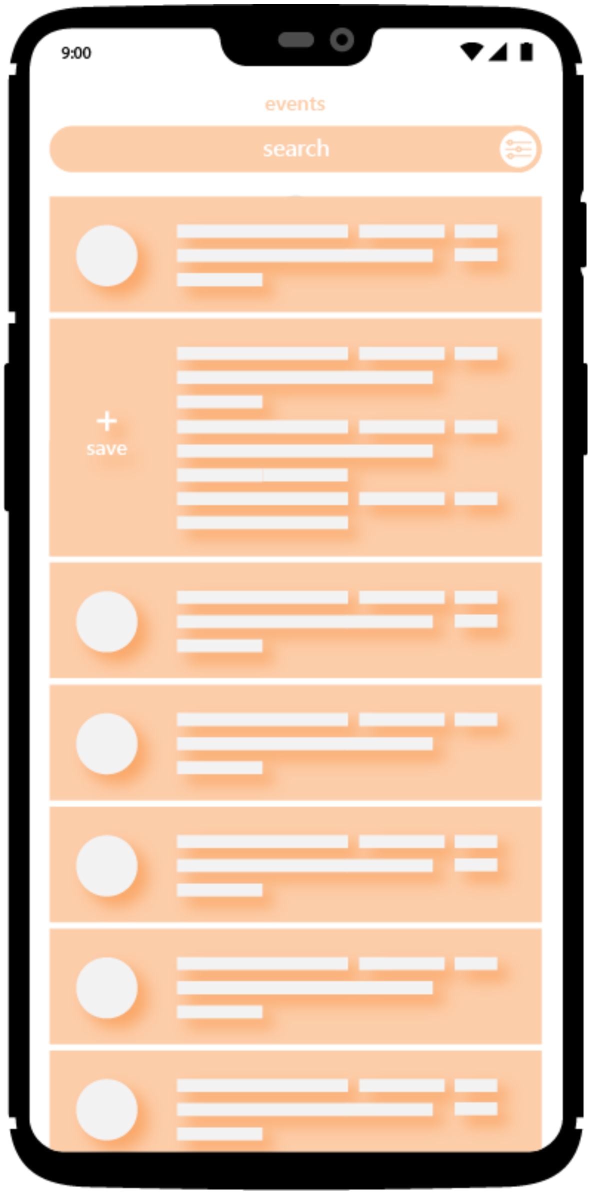
3.5.5.4 Tutorial Screen

The tutorials page will contain a list of all the tutorials available and the user may choose among them the tutorials he would like to see using the filters to narrow down the search.



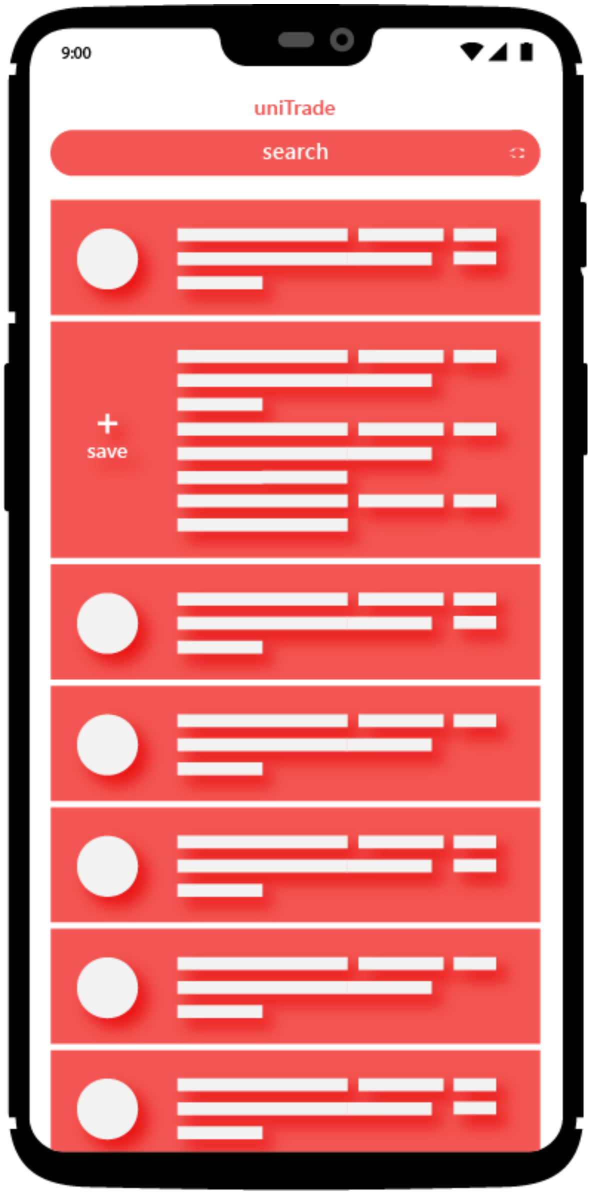
3.5.5.5 Events Screen

The events page will contain a list of the events happening nearby. The user will be able to search for specific events using filters, save an event using the save button and also by clicking on a specific event he can be a more detailed information regarding it. For the events the user saves, notifications will be sent as a reminder.



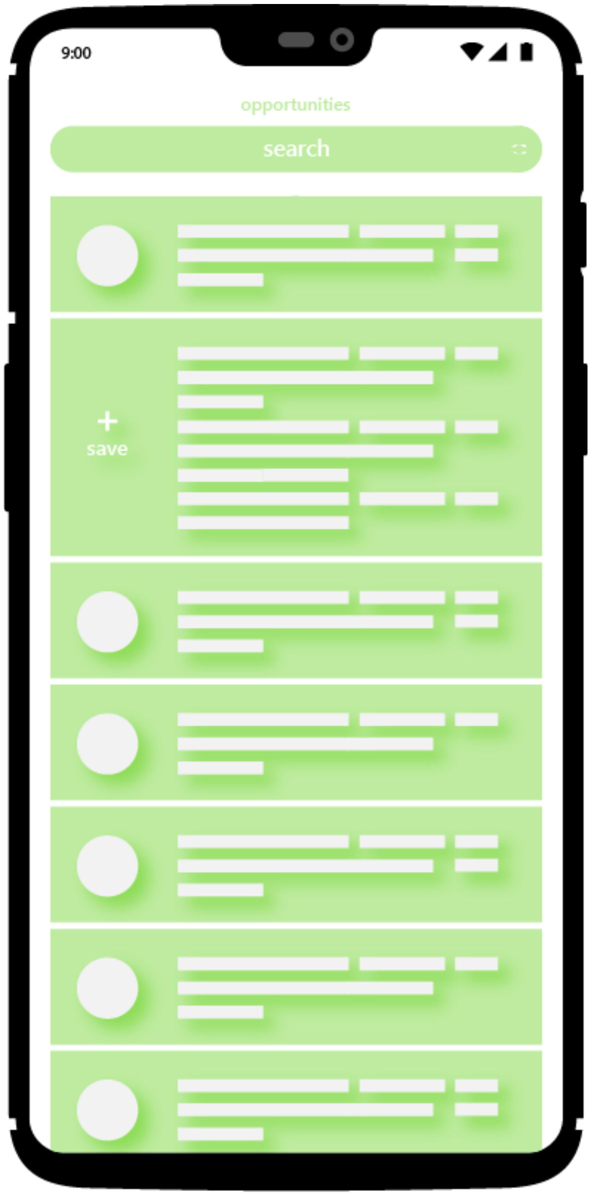
3.5.5.6 UniTrade Screen

The UniTrade will contain a list of all the items that are available to trade including selling, trading etc. Also in this page, the user will be able to search for specific items using filters, save an item using the save button and also by clicking on a specific item he can be a more detailed information regarding it. For the items the user saves, notifications will be sent to him as a reminder.



3.5.5.7 Opportunities Screen

The Opportunities Page will display a list of the opportunities regarding universities and internships. The user can search for a specific opportunity using the filters available, can save the opportunities that he/she prefers and get more detailed information on them. On the side of each opportunity a list of people that were part of that will be displayed and user can send a message to them in order to get a more realistic insight. For the opportunities the user saves, notifications will be sent as a reminder.



4. References

[1] INTERNET USAGE STATISTICS The Internet Big Picture. (n.d.). 2018 https://www.internetworldstats.com/stats.htm. Accessed: 2018-10-14.

[2] “Top Universities.” *Top Universities*, www.topuniversities.com/. Accessed: 2018-11-04.

[3] “BigFuture - Get Ready for College - College Planning, Financial Aid, Educator Resources.” *Pay for College - Where to Find College Scholarships*, bigfuture.collegeboard.org/. Accessed: 2018-11-04.

[4] Adams, Susan. “The 10 Best Websites For Finding An Internship.” *Forbes*, Forbes Magazine, 18 Feb. 2015, www.forbes.com/sites/susanadams/2015/01/30/the-10-best-websites-for-finding-an-internship/#61ef59b11b44. Accessed: 2018-11-04.

[5] “YouTube Live Streaming: Myths, Perks and Perils.” *Event Amplifier*, 6 Nov. 2017, eventamplifier.com/2015/02/11/youtube-live-streaming-myths-perks-and-perils/. Accessed: 2018-11-04.

[6] *Guideline: Communication Diagram*, www.michael-richardson.com/processes/rup_for_sqa/core.base_rup/guidances/guidelines/use-case_model_CC121CF4.html. Accessed: 2018-11-04.