University of Jordan

King Abdullah College of Information Technology

Supervisor: Dr. Mohammad Abushariah

Project subject:

Chabot for KASIT Graduation Project.

Name	Id
Basel Abu Alaish	0187604
Ahmad Abdallah	0187126

ABSTRACT

Chabot for KASIT Graduation Project is a system that helps students of King Abdullah II College of Information Technology in knowing the details of the graduation project and answering common questions at any time and from anywhere the student needs.

Acknowledgment

We would like to express our gratitude to all people who did provide us with the support on this road of achieving our bachelor's degree.

Also, we would like to thank our project supervisors Dr. Mohammad Abushariah for their help and support.

Table of Contents

1. CHAPTER ONE: INTRODUCTION 10

1.1 Preamble: 10

1.2 Project Motivation: 10

1.3 Problem Statement**:** 10

1.4 Project Aim and Objectives: 10

1.5 Project Scope 10

1.6 Project Software and Hardware Requirements 10

1.6.1 Hardware: 10

1.6.2 Software: 10

1.7 Project Limitations 10

1.8 Project Expected Output: 11

1.9 Project Schedule: 11

2. CHAPTER TWO: RELATED EXISTING SYSTEMS 13

2.1 Introduction 13

2.2 Existing Systems 13

2.3 Overall Problems of Existing System 13

2.4 Overall Solution Approach 13

2.5 Summary 14

3.0 CHAPTER THREE: SYSTEM REQUIREMENTS ENGINEERING AND ANALYSIS……..……14

3.1 Introduction…….……...… 14

3.2 Feasibility Study ……...……..…...14

3.3 Requirements Elicitation Techniques……..……16

3.4 Targeted Users…….…….…….……...17

3.5 Functional Requirements Definition ……..…..…….….…17

3.6 Functional Requirements Specification…….…….….…18

3.7 Non-Functional Requirements……..…….…….…..…19
3.8 Summary……...……..….20

4.0 CHAPTER FOUR: SYSTEM DESIGN……...……....20

4.1 Introduction..... 20

4.2 Context diagram..... 21

4.3 Data flow Diagram (DFD) 22

4.4 Entity Relationship Diagram (ERD) 23

4.5 UML Use Case Diagram..... 24

4.6 UML Sequence Diagram 25

4.7 UML Class Diagram 27

4.8 Graphical user interface design..... 28

4.9 Summary 34

5.0 CHAPTER FIVE: SYSTEM IMPLEMENTATION…….…….…34

5.1 Introduction…….……...34

5.2 Database Implementation……...34

5.3 Graphical User Interface Implementation…….……..…35

5.4 Backend…….……..37

5.5 Summary…….……....43

6.0 CHAPTER SIX: SYSTEM TESTING AND INSTALLATION……....44

6.1 Introduction…….…...44

6.2 Heuristic Evaluation……...…….……...44

6.3 Cooperative Evaluation……...……47

6.4 Requirements Validation and Completeness ……..……....49

6.5 Summary……..50

7.0 CHAPTER SEVEN: PROJECT CONCLUSION AND FUTURE WORK..……50

7.1 Introduction…..……50

7.2 Overall Weaknesses……50

7.3 Overall Strengths……..…….…50

7.4 Future Work……..……50
7.5 Summary……....……51
REFERENCES.....51
APPENDICES.....51
Appendix A: Questionnaire 51
Appendix B: User Manual…….52

List of Tables

Table 1.1: Project Schedule 10

Table 2: Personal Costs 14

Table 3: Hardware and Software Costs 15

Table 4: Functional Requirements Definition 17

Table 5: Functional Requirements Specification ..... **Error! Bookmark not defined.**

Table 6.2.1: Heuristic Evaluation 45

Table 6.2.3: Summary of Validation heuristic 46

Table 6.2.4: Summary of violations by severity ratings for participant 1 46

Table 6.2.5: Summary of violations by severity ratings for participant 2 46

Table 6.2.5: Summary of violations by severity ratings for participant 3 47

Table 6.3.1: Cooperative testing participant's data 47

Table 6.3.2 Cooperative evaluation for participant 1 47

Table 6.3.3 Cooperative evaluation for participant 2 47

Table 6.3.4 Cooperative evaluation for participant 3 47

Table 6.3.5 Cooperative evaluation for participant 4 48

Table 6.3.6 Cooperative testing participant's data 48

List of Figures

Figure 1: Context Diagram 21

Figure 2: Data Flow Diagram 22

Figure 3: ERD 23

Figure 4: Class Diagram 23

Figure 6: User Use Case 24

Figure 7: Admin Use Case 24

Figure 9:admin Sequence Diagram2 ..... **Error! Bookmark not defined.**

Figure 10: user Sequence Diagram ..... **Error! Bookmark not defined.**

Figure 11: Admin Sequence Diagram ..... **Error! Bookmark not defined.**

Figure 12: user Sequence Diagram ..... **Error! Bookmark not defined.**

Figure 17: GUI - Home page 32

Figure 19: GUI - chatbot 32

Figure 20: GUI – frequency questions 33

Figure 21: GUI – filter questions 33

Figure 22: GUI imp - Home page 35

Figure 23: GUI imp - chatbot 35

Figure 24: GUI imp – frequency questions 36

1. CHAPTER ONE: INTRODUCTION

1.1 Preamble:

The graduation project for students of King Abdullah II College of Information Technology is taken after the student passes 90 hours of college courses. Our chatbot aims to answer students' questions about the graduation project and help them understand the requirements of the project.

1.2 Project Motivation:

The graduate students have difficulties in knowing the basic characteristics of the graduation project, so we decided to make this bot to help them know all the important details and announcements.

1.3 Problem Statement:

The students have many questions about the project and they need someone to answer on their questions, and how to deal with the parts of the project.

1.4 Project Aim and Objectives:

The project aims to help students by:

1- Reducing the student's time in searching for different sources.

2- Gathering all project information in one place, which reduces effort and time for the student.

3- The ability to search and ask about any information related to the project at any time.

1.5 Project Scope

The system allows students to ask about any information related to the project at any time and any place. The system provides many features such as the possibility of direct questioning or access to frequently asked questions.

1.6 Project Software and Hardware Requirements

1.6.1 Hardware:

- Any laptop that can connect to a network
- Any smartphone that can connect to a network
- Tablet, IPAD

1.6.2 Software:

- Operating System.
- Browser
- Atom
- Visual Studio Code (VSC)

1.7 Project Limitations

1. That the project budget does not exceed 10000 $

2. The project completion time shall not exceed one year

3. High accuracy in answering questions

1.8 Project Expected Output:

The bot will be able to answer students' questions with high accuracy and efficiency, and it will also enable individuals to view frequently asked questions and open the site from more than one device.

1.9 Project Schedule:

Table 1.1: Project Schedule

Task Number	Task definition	Duration (days)	Dependency	Done By
T1	Introduction	11	-	Basel
T1.1	Preamble	1	-	Basel
T1.2	Project motivation, Problem statement	2	-	Basel
T1.3	Project aim, Project scope	1	-	Ahmad
T1.4	Software and hardware requirements	2	T1.2	Ahmad
T1.5	Project limitation, project expected output	2	T1.3 T1.4	Basel
T1.6	Project schedule, report outline	3	-	Basel
T2	Related existing Systems	4	-	Ahmad
T2.1	Introduction, existing systems	1	-	Ahmad
T2.2	Overall problems of existing systems	2	T2	Ahmad
T2.3	overall solution approach, Summary	1	T2	Basel
T3	System Requirements engineering and analysis	18	-	All team members
T3.1	Introduction	1	-	Basel
T3.2	Feasibility study	4	T1.4 T1.5	Basel
T3.3	Targeted users	2	T1.3	Ahmad

T3.4	**Functional Requirements**	**7**	**T1.2 T1.3**	**All team members**
T3.5	**Non-Functional requirements, Summary**	**4**	**T3.4 T1.3**	**All team members**
T4	**System design**	**20**	**T3**	**Basel**
T4.1	**Introduction**	**1**	**-**	**Ahmad**
T4.2	**Context diagram, Introduction**	**3**	**T3.4 T3.5**	**Basel**
T4.3	**Data flow diagram**	**3**	**T4.2**	**Ahmad**
T 4.4	**ERD**	**2**	**T4.3**	**Ahmad**
T4.5	**Use case diagram**	**3**	**T3.4**	**All team members**
T4.6	**Sequence diagram**	**3**	**T4.5**	**Basel**
T4.7	**Class diagram**	**2**	**T4.5 T4.6**	**Ahmad**
T4.8	**Graphical user interface, summary**	**3**	**T3.4**	**All team members**

2. CHAPTER TWO: RELATED EXISTING SYSTEMS

2.1 Introduction

There are a lot of Chabot's in different fields and websites, in our college, we noticed that the students have many questions that they need to answer about Graduation Projects. But we did not find a Chabot that answers the inquiries of university students, especially to help them with their graduation projects. In this chapter we will show the related, we will present some of the existing systems that provide Chabot's.

2.2 Existing Systems

- https://www.uaeu.ac.ae/en/
- https://www.mongodb.com/
- https://www.onepeloton.com/
- https://outgrow.co/blog/
- https://seattleballooning.com/
- https://www.massageenvy.com/

2.3 Overall Problems of Existing System

- No suggestions for the most frequently asked questions.

  We plan to direct highlight the most popular questions when launching the Chabot to get the most benefit out of the Chabot.

- Design of Chabot.

  some websites include Chabot's but it's not clear logo and without simple message or sound notification.

2.4 Overall Solution Approach

1. ·Add most frequently asked questions as suggestions on page

2. ·Design of Chabot:

We plan to create a suitable design that fits logo in the angle or full screen

2.5 Summary

In this chapter, we presented related existing systems with our project, and we are going to improve its advantages and ignore its disadvantages and we will propose our solutions for creating a new Chabot for graduation project.

3.0 CHAPTER THREE: SYSTEM REQUIREMENTS ENGINEERING AND ANALYSIS

3.1 Introduction

To make the system good and make the most of it as much as possible, we must do a general feasibility study and collect the functional and non-functional requirements. Upon completion of all required procedures, we can describe the system to be operated, its characteristics, and know the total cost.

3.2 Feasibility Study

3.2.1 Operational Study:

Study to determine the acceptable solutions for problems in two cases internal issues or external issues and if system still effective when upgraded in adequate throughput, response time, provide end user and managers with timely and accurate useful formatted information and reliable service and flexible.

3.1.2 Technical Feasibility

The web service will be mainly developed using web development and NLP programing, it will target all kinds of computer devices such as desktops, laptops, tablets, and mobiles, in other words, the system will be responsive. With all that has been said, this can be achieved by learning some web development and NLP programming might require technical support. The Information technology center that will use the system may need someone to support them if any bugs or errors appear.

Table : Personal costs

Name	Quantity	Cost	Description
Developer	2	200	Web development and very good NLP programming
Designer or designing software	1	50	Design widget and logo for the chatbot
data analyst	1	50	Collect data and question to get the maximum benefit possible

Table : Hardware and Software Costs

Name	Quantity	Cost per unit	Description
Laptops	2	500	Laptops will be used to develop and test the application.
Mobile	1	100	Laptops will be used to test the application.

Initial cost :1600 JD

Cost per month :500 JD

3.1.4 Benefits:

Tangible & Intangible benefits:

1. Reducing students' problems and questions.

2. Reducing the questions that burden the heads of departments and doctors.

3. Improve the level of graduation projects

4. Raising the level of awareness of the skills of college graduates

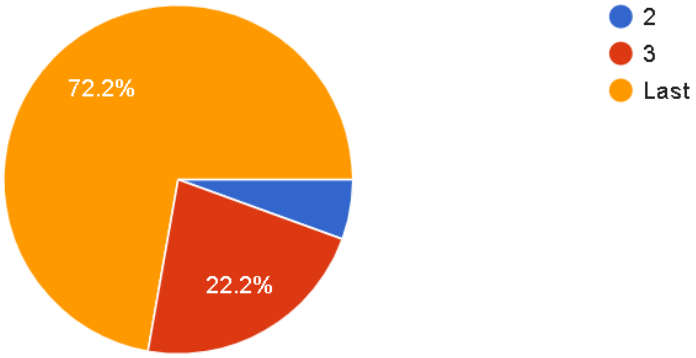
5.The possibility of expanding the chatbot to more colleges in the university to help students.

3.3 Requirements Elicitation Techniques

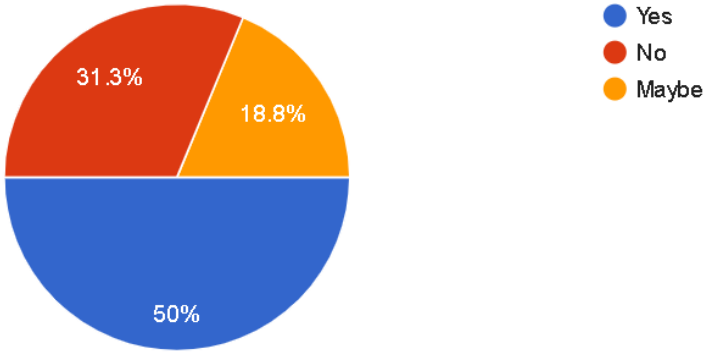
Questionnaire:

A form was created and posted to public groups to gather information about the requirements and to know their opinion if they will use the system or not and if is useful to them and the results are as follows:

? Which year are you in university



? Do you have difficulties understanding the documentation



3.4 Targeted users

All college students who are supervised to graduate after completing 90 hours of study.

These users are from different departments, races, and personalities.

Primary stakeholders:

1. Students(S1)

Secondary stakeholders:

1.Admin(S2)

3.5 Functional Requirements Definition

Table4: Functional Requirements Definition

Requirement No.	Requirement name	Stakeholders	Priority Range (1-10)
R1	Click and display for the system	S1	10
R2	Ask a question	S1	9
R3	Download full answer	S1	8
R4	Rate benefit and feedback	S1	5
R5	Send new question	S1	9
R6	See Frequently Asked Questions	S1	7
R7	Login	S1, S2	7
R8	Add new questions and answers	S2	8
R9	Delete question	S2	7
R10	Edit answer	S2	7
R11	Disable or enable system	S4	7

3.6 Functional Requirements Specification

Table 5 : Functional Requirements Specification

Requirement No.	Requirement name	Description
R1	Click and display the system	The ability of the student to click and turn on the system
R2	Ask a question	The student's ability to ask any question that comes to his mind and the system is analysing and trying to answer it.
R3	Download full answer	If the answer is long and you need a long explanation or clarification, the conversion to a link with all the details.
R4	Rate benefit and feedback	The student's evaluation of the extent to which the answer was useful and whether it covered all his questions.
R5	Send new question	The student's ability to submit a new question that has not been answered before
R6	See Frequently Asked Questions	Ability to view frequently asked questions
R7	Login	The ability of the admin/student to log into the site
R8	Add new questions and answers	The ability of the admin to add new questions and answer previously unanswered questions.
R9	Delete question	If the admin wants to delete the question, this is within his ability.
R10	Edit answer	The ability of the admin to modify the answer in proportion to the changes and developments.
R11	Disable or enable	The ability of the admin to deactivate and reactivate the system

3.7 Non-Functional Requirements

R#	Requirements	Description
1	Reliability	The system should provide the services as specified; when a user/admin selects the service he/she wants, then the system will response and offer it as required.
2	Usability	All the users can use the services offered by the system through an easy to use and simple interface.
3	Availability	The system is available to be used at any time; when a admin/user requests a service, then the system should be ready to provide it at the time of doing such request.
4	Security	The system should be able to prevent any unauthorized access to its components, and prevent users from make any violations, this achieved system during checks processes.
5	Maintainability	This system is coded in an organized maintainable way, to make it easier to add new features that users need and be able to understand by the maintenance team.
6	Efficiency	The system will save time and money, also facilitate the operations by the information provided by the system.
7	Accuracy	The information accuracy must be to accurate to provide information in real time

3.8 Summary:

The results of the feasibility study showed that this project is beneficial in all aspects, and therefore the writing of functional and non-functional requirements has been started. The number of functional requirements was 12 and the number of non-functional requirements was 7. Also, each functional and non-functional requirement was described step by step. In addition, this section determined the targeted users which are students they have done 90 hours and about to graduate from college

4. CHAPTER FOUR: SYSTEM DESIGN

4.1 Introduction

This section will provide the basic concepts of our system including all important diagrams (Context Diagram, Data Flow Diagram (DFD), ERD, Use Case, Sequence Diagram, Class Diagram). Additionally, we will provide a set of interfaces that reflect our system (GUI).

4.2 Context diagram

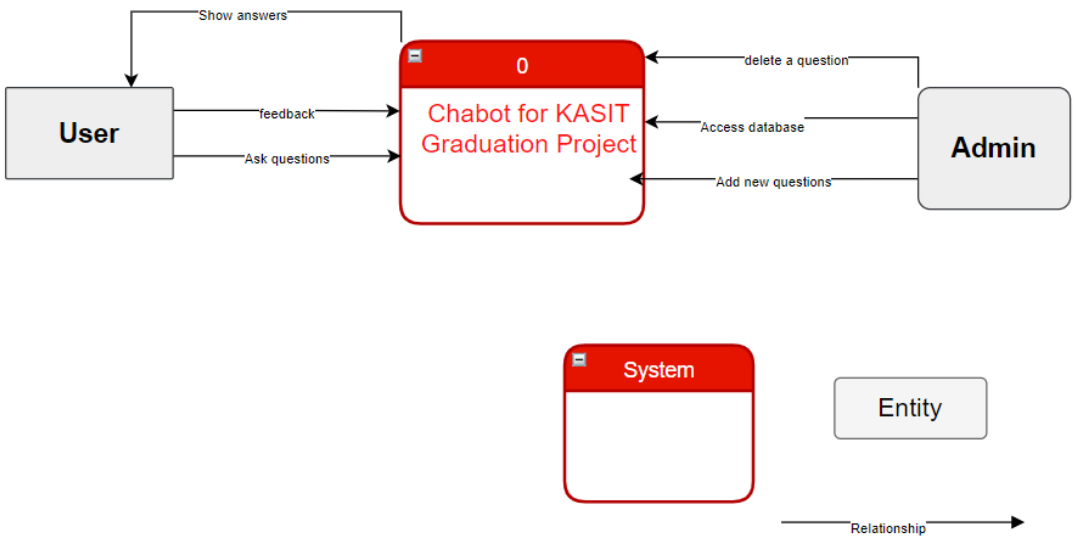


Figure 1: Context Diagram

4.3 Data flow Diagram (DFD)

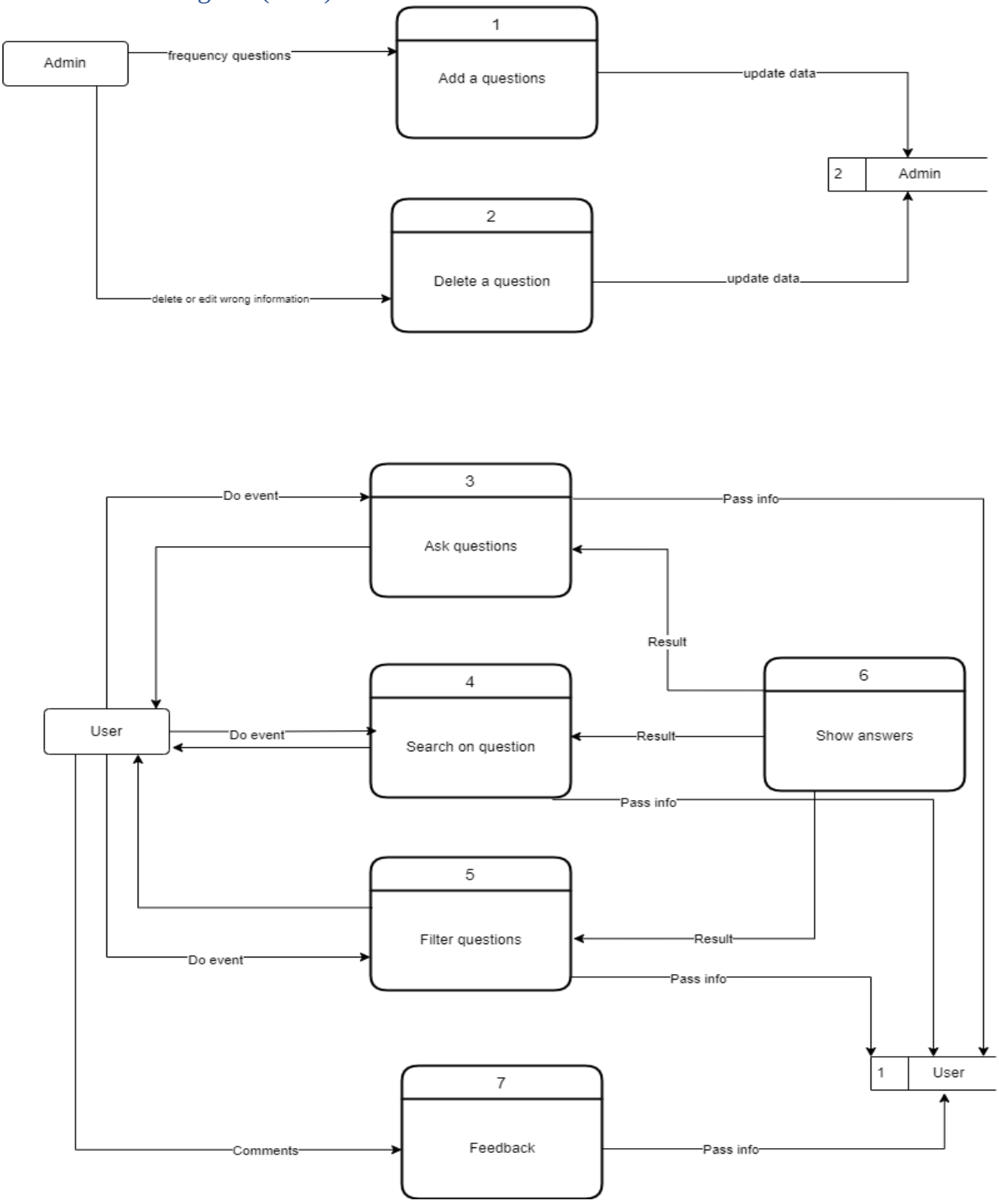


Figure 2: Data Flow Diagram

4.4 Entity Relationship Diagram (ERD)

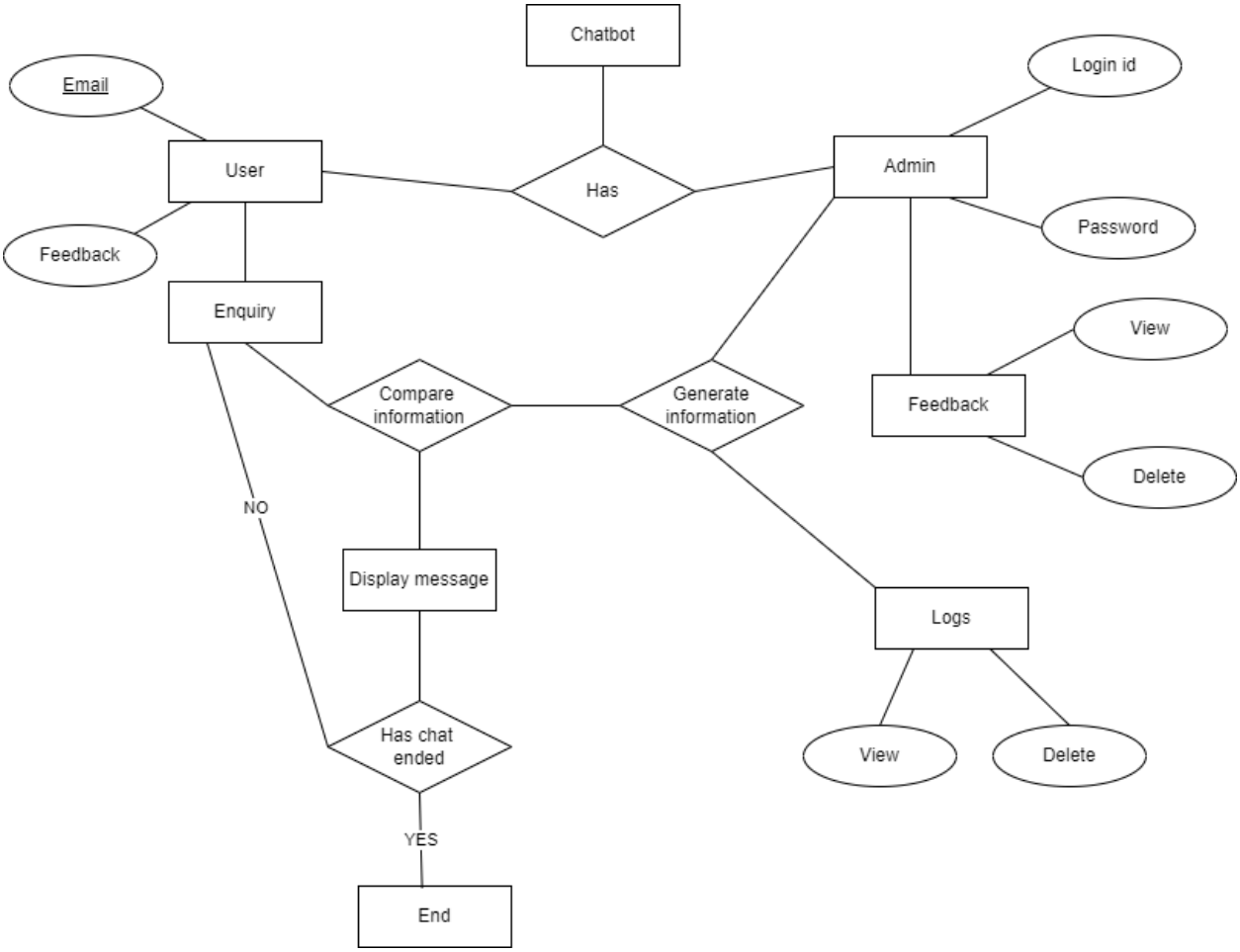


Figure 3: ERD

4.5 UML Use Case Diagram

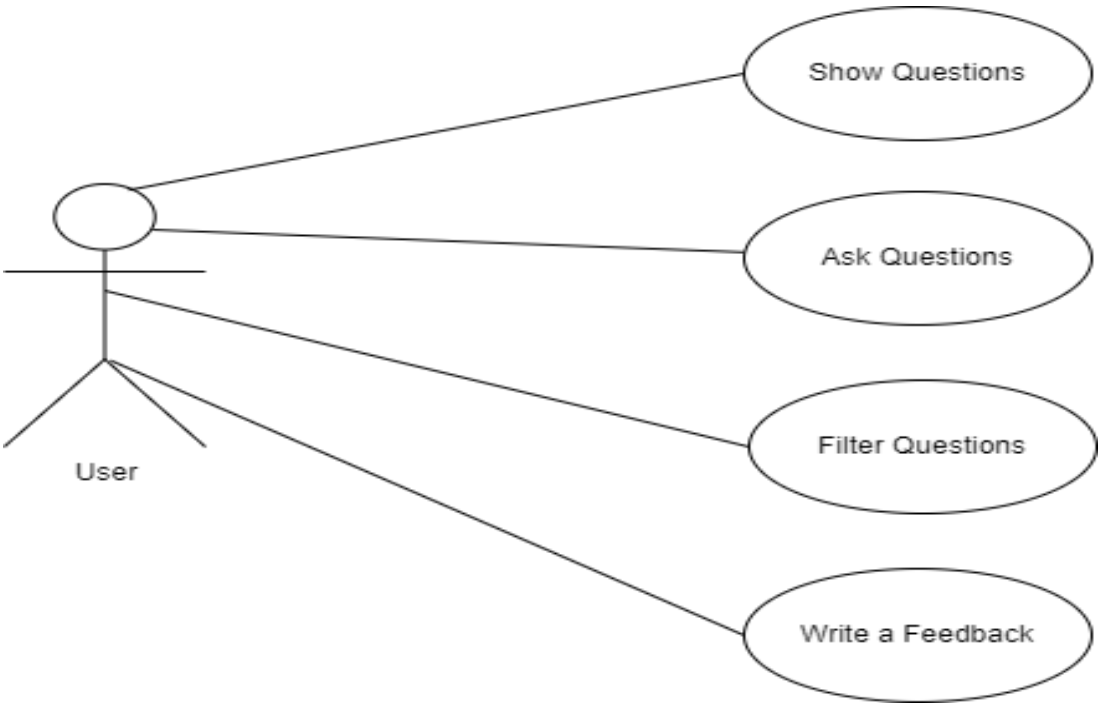


Figure 4: User Use Case

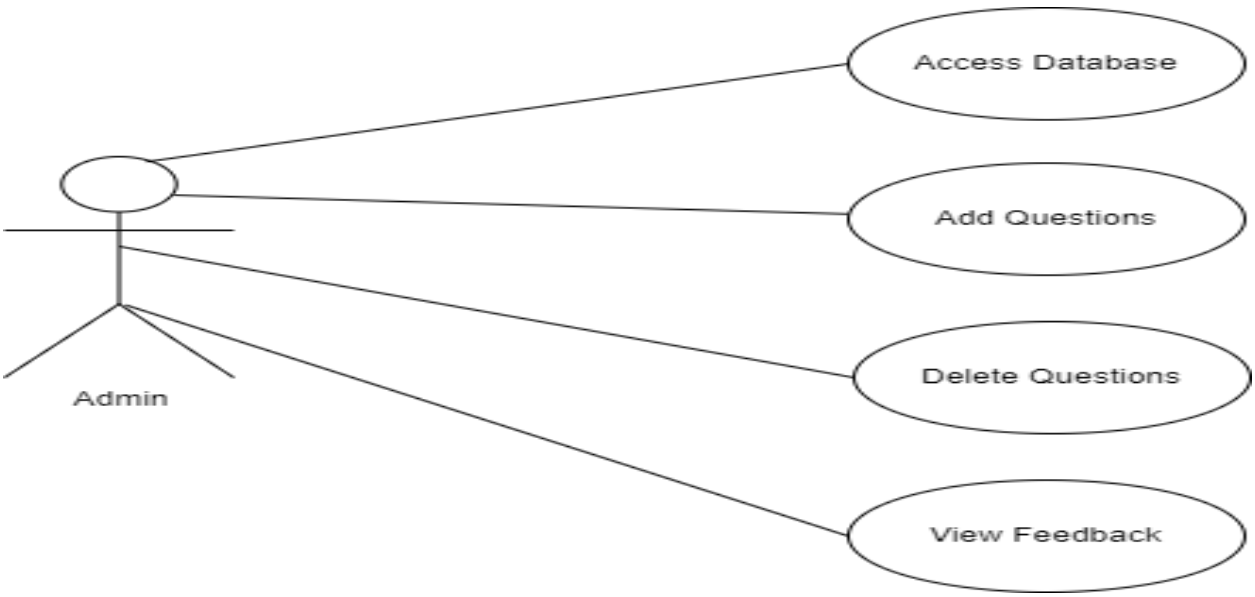


Figure 5: Admin Use Case

4.6 UML Sequence Diagram

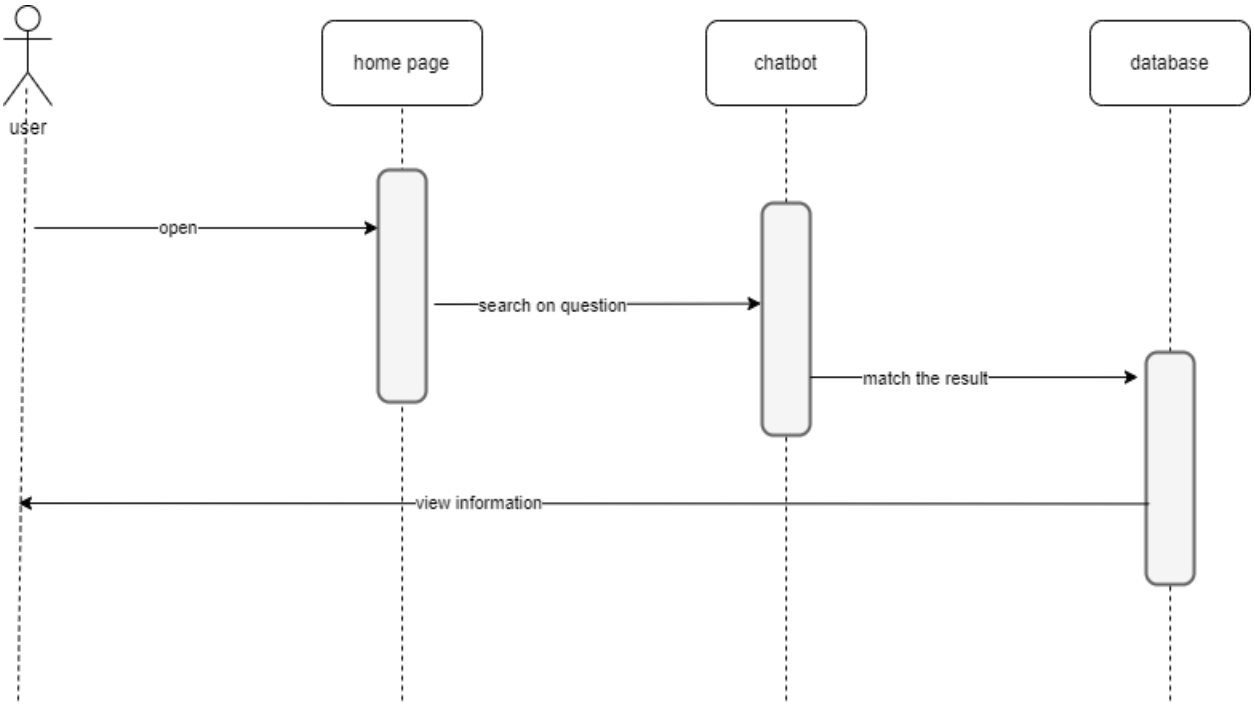


Figure 4: user search Sequence Diagram

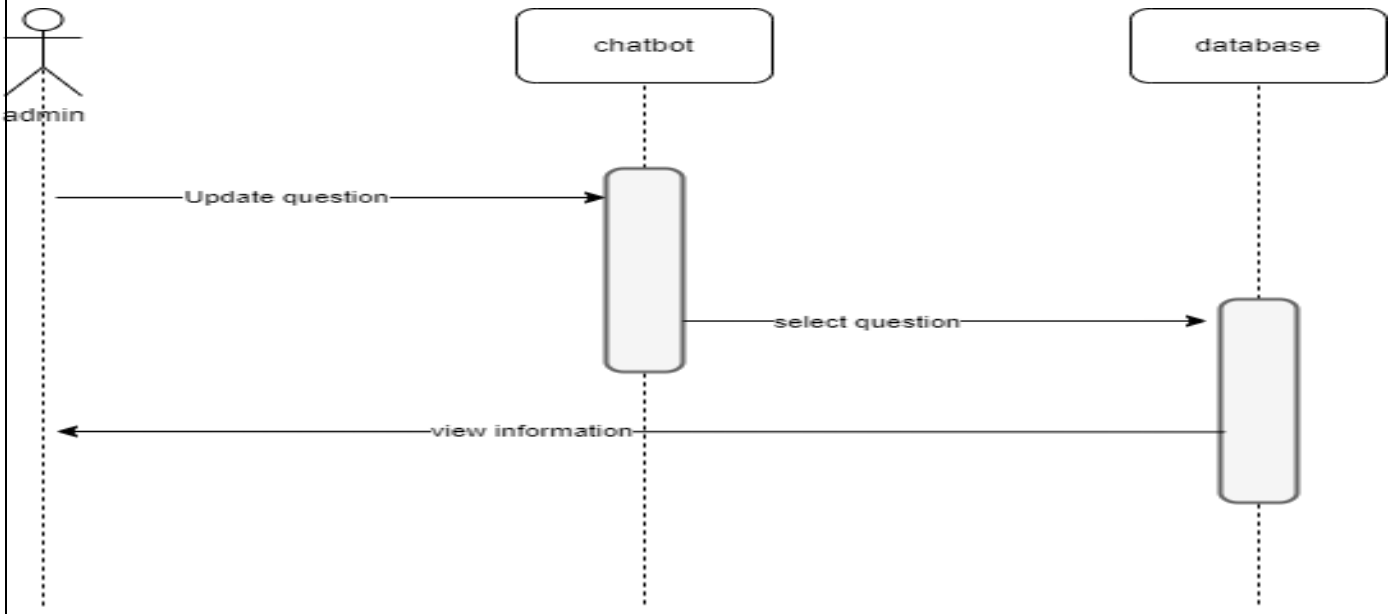


Figure 7: Admin Sequence Diagram

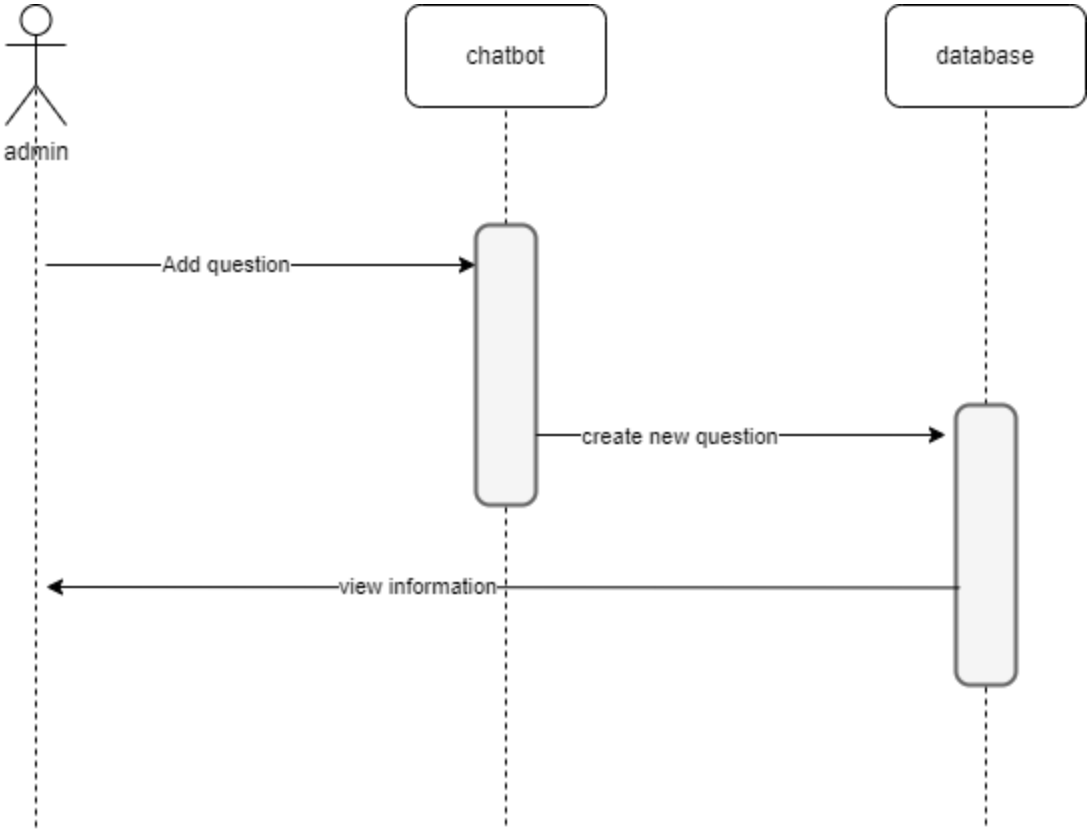


Figure 8: Admin Sequence Diagram2

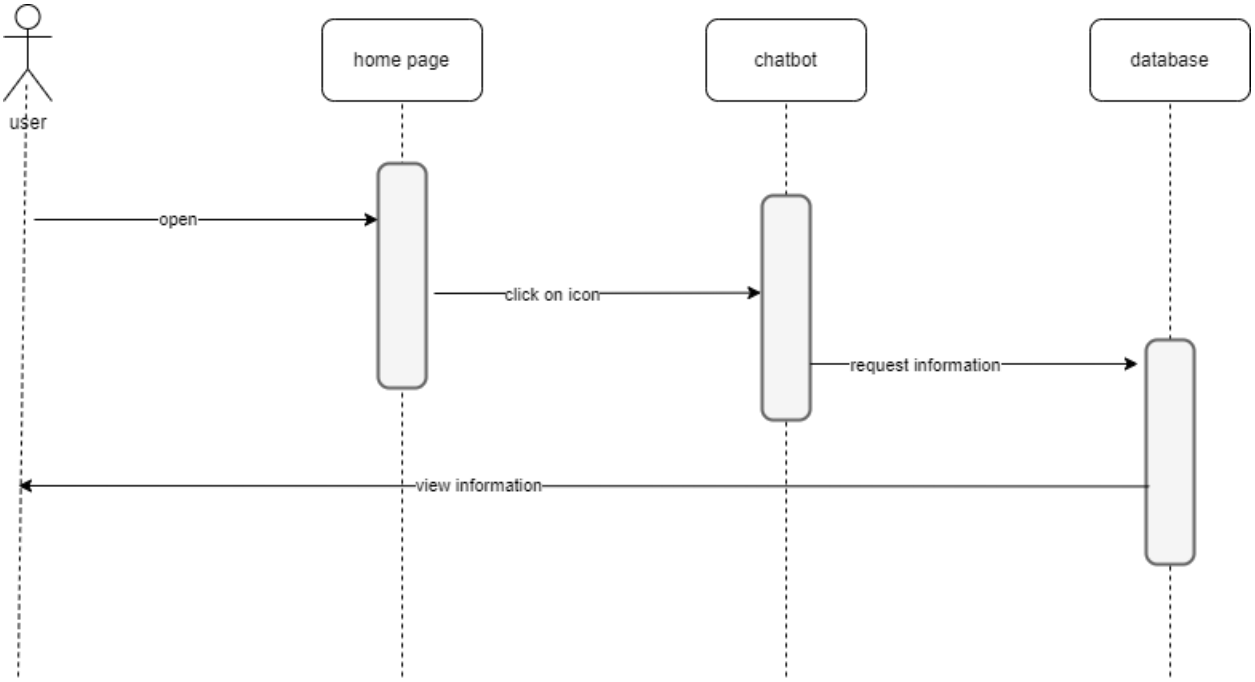


Figure 9: user Sequence Diagram

4.7 UML Class Diagram

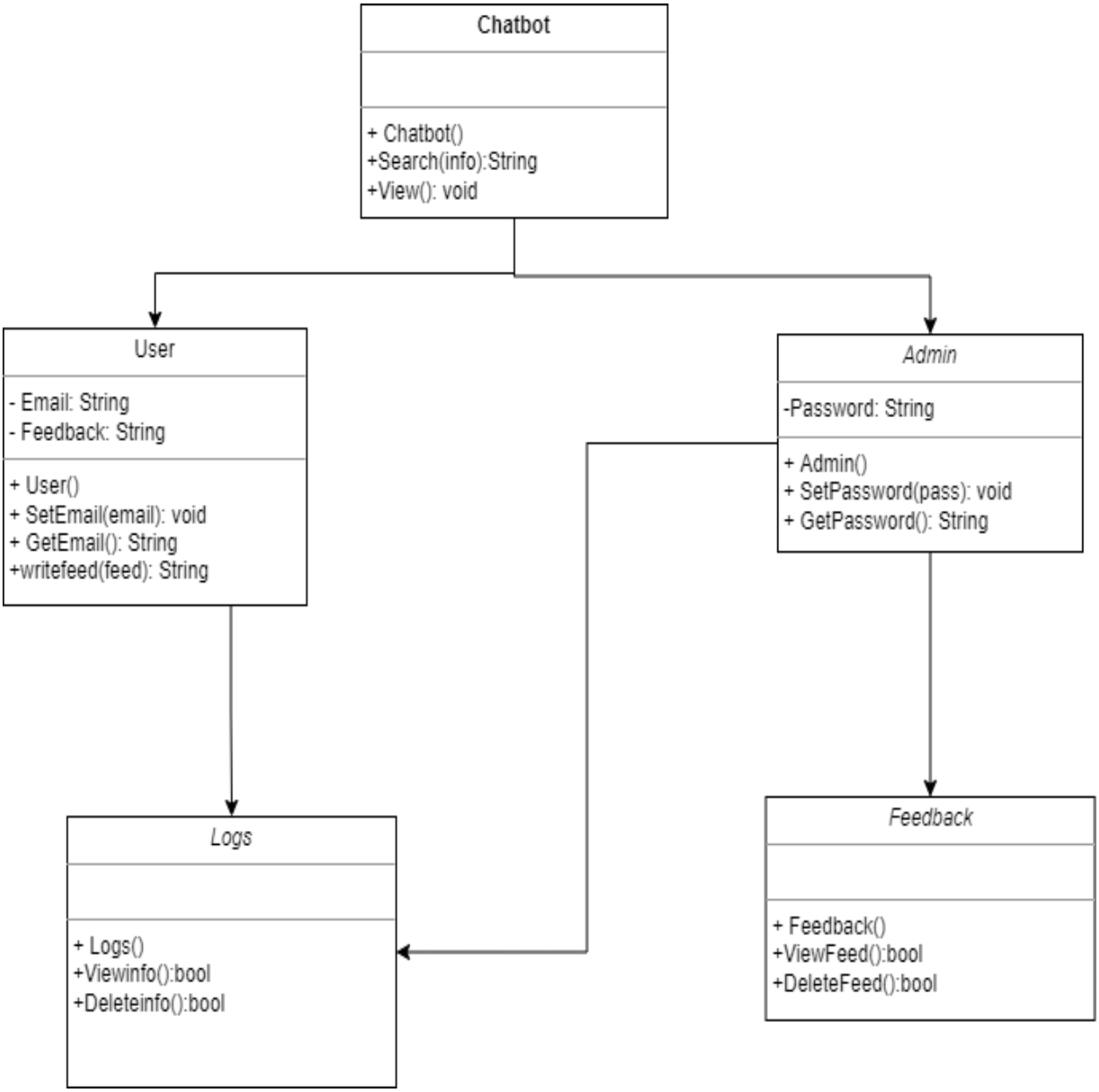


Figure: Class Diagram

4.8 Graphical user interface design

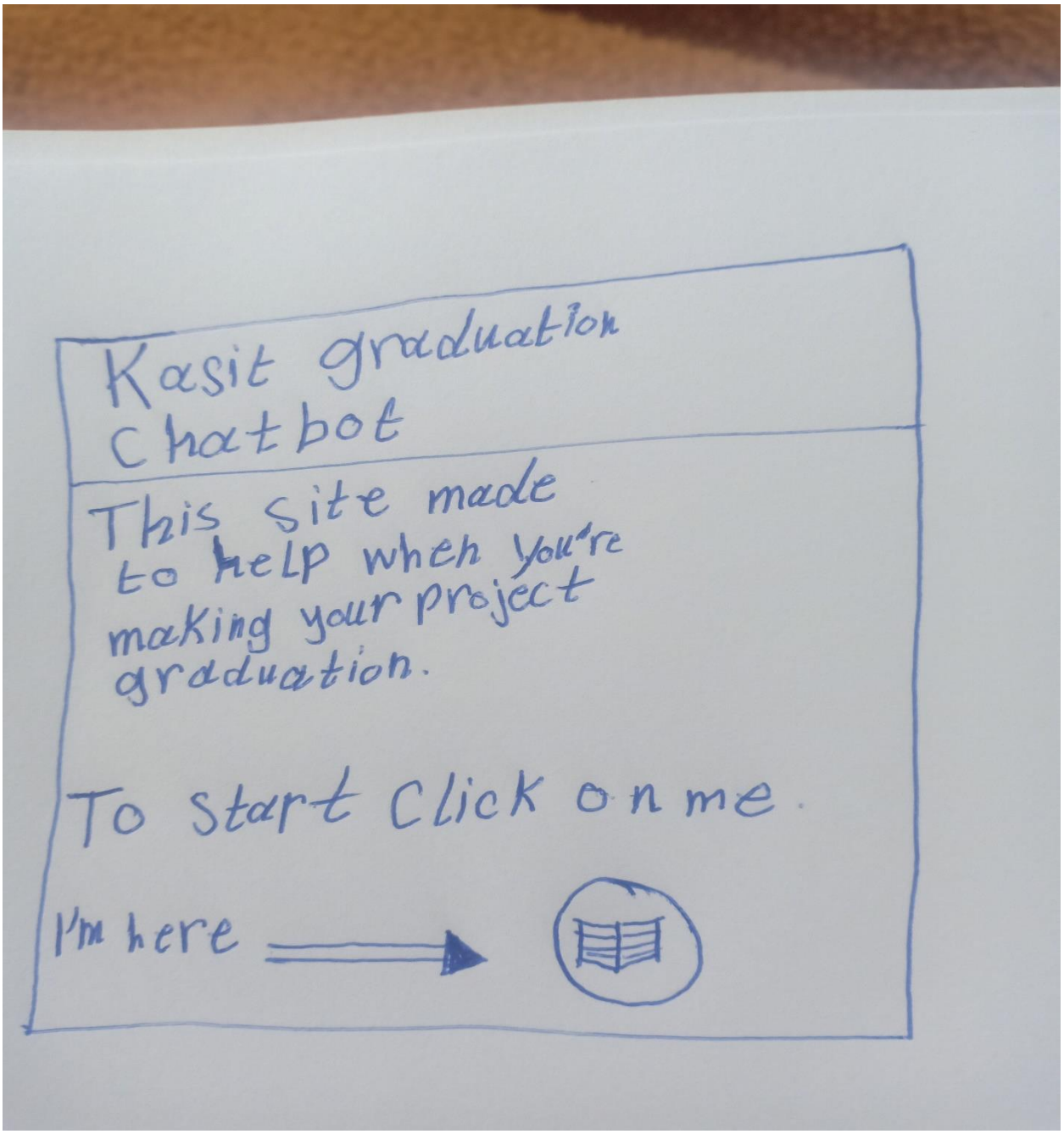


Figure 11: GUI - Home page

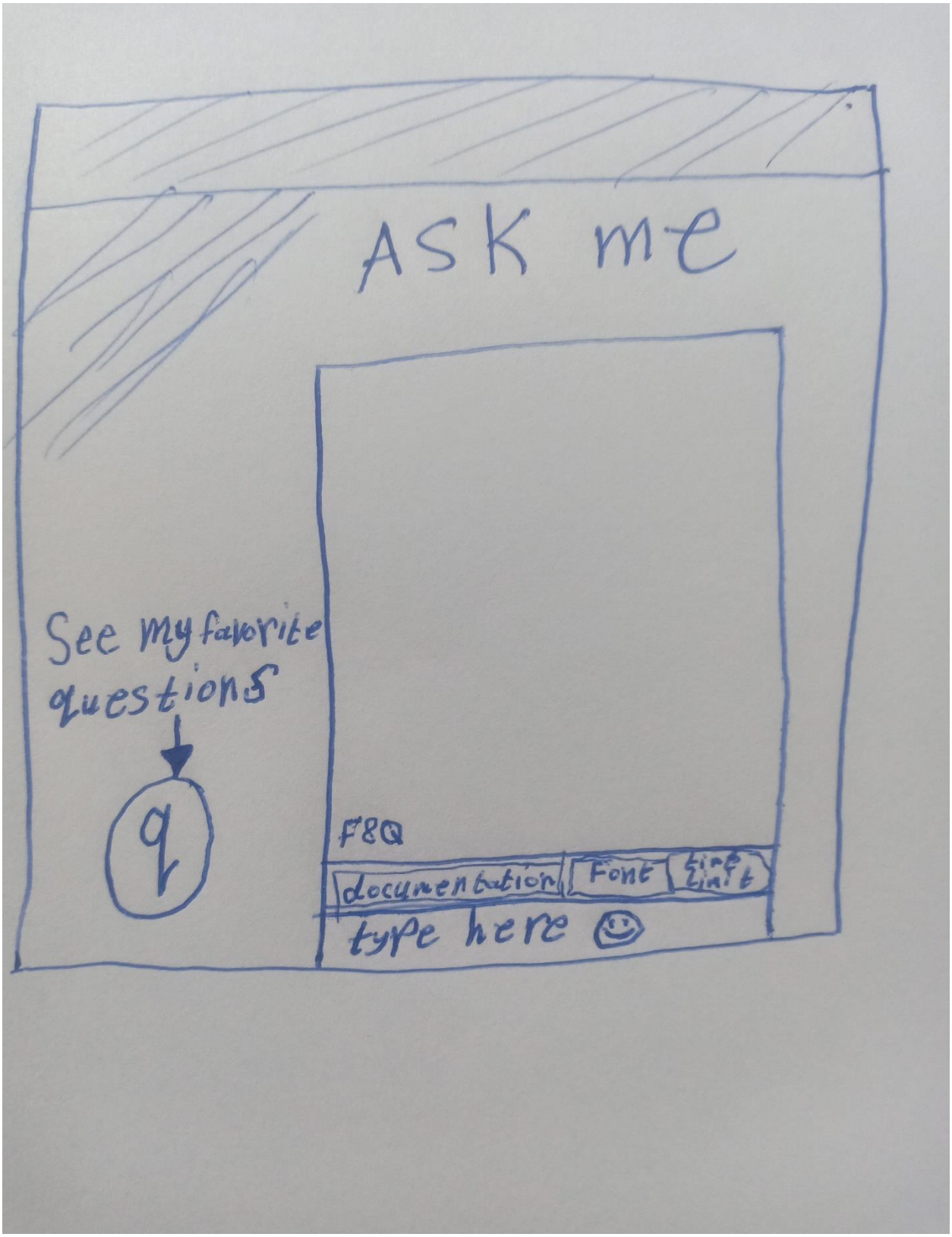


Figure 12: GUI - Chabot

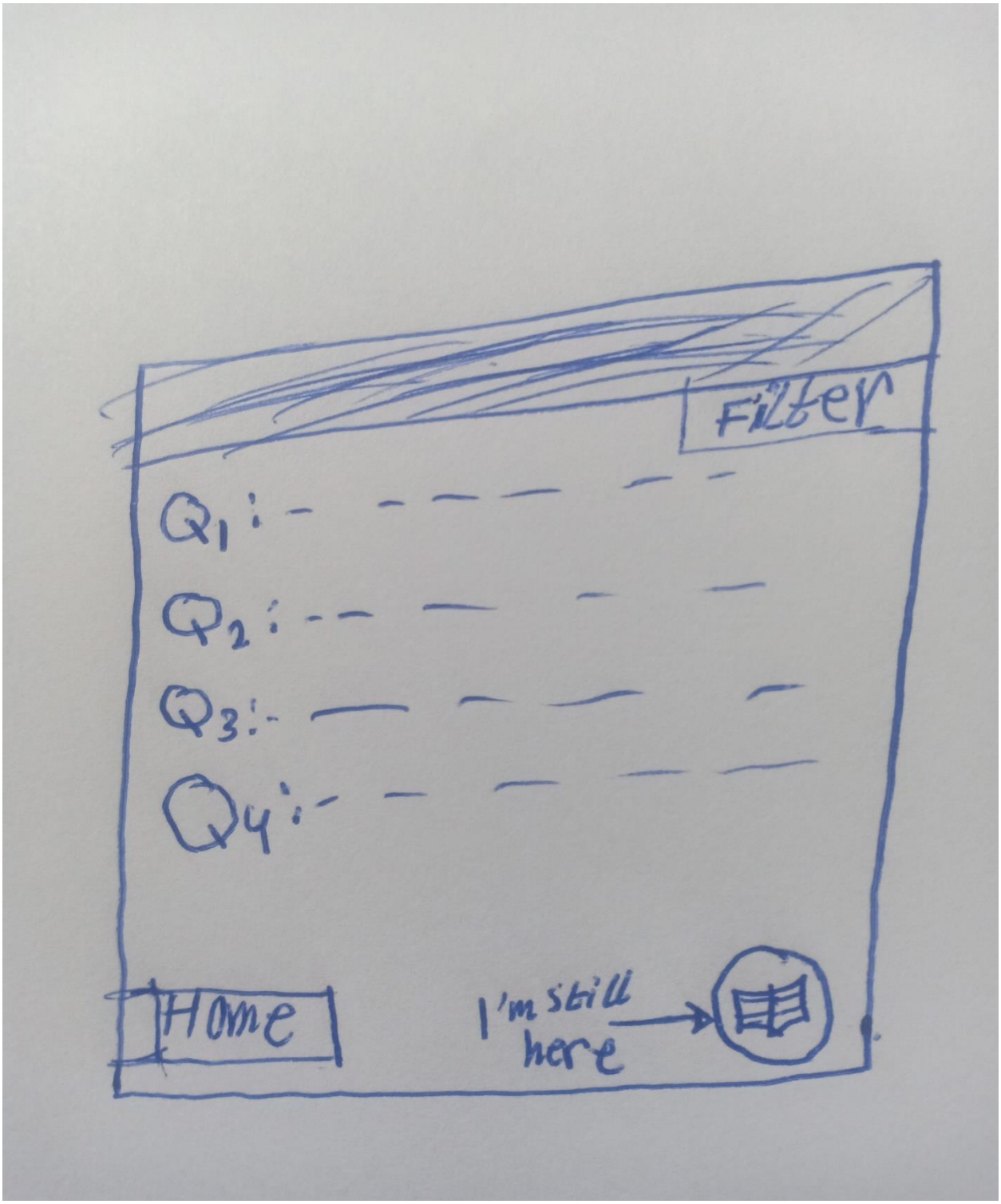


Figure 13: GUI – Frequency questions

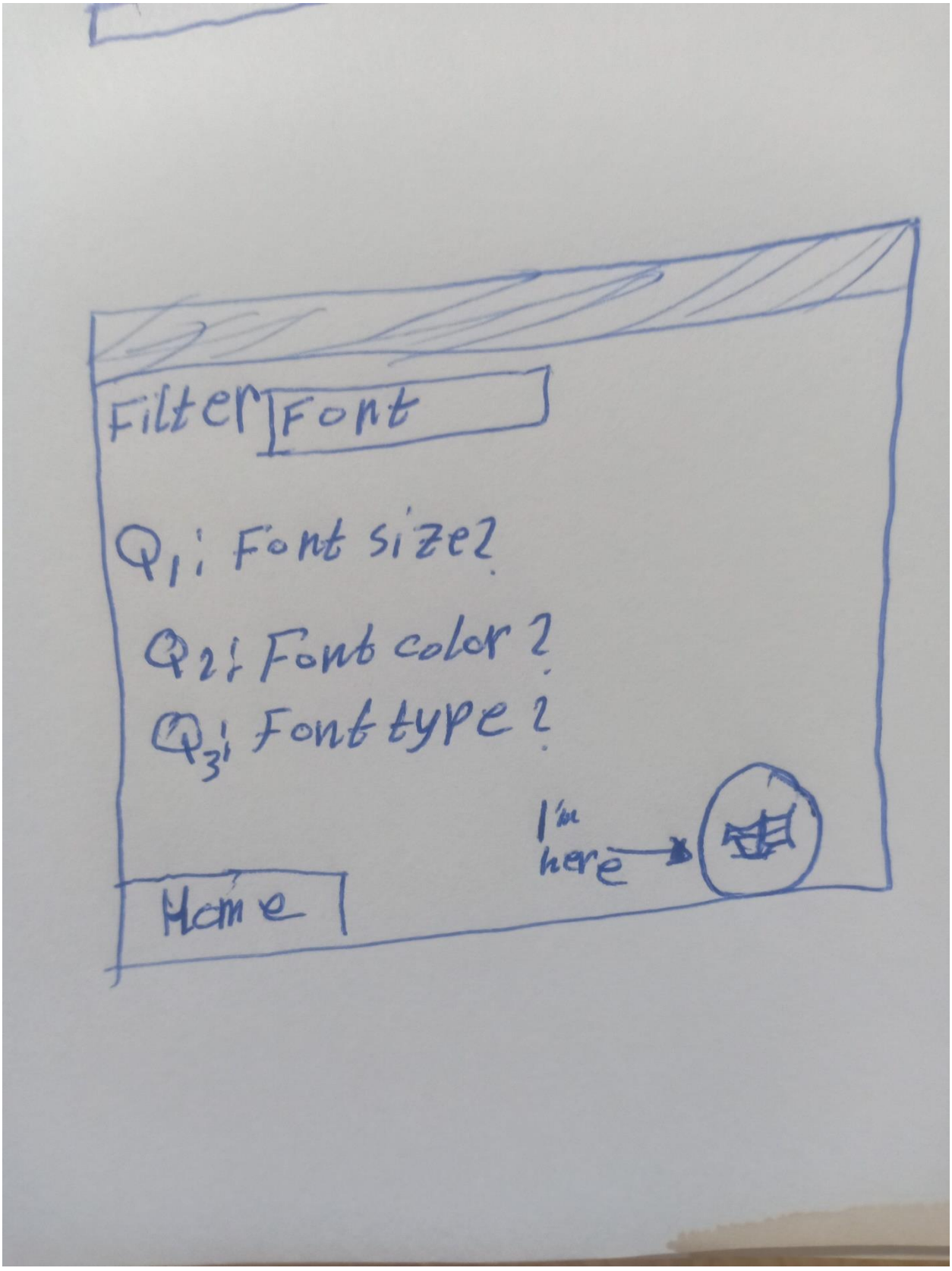


Figure 14: GUI –Filter questions

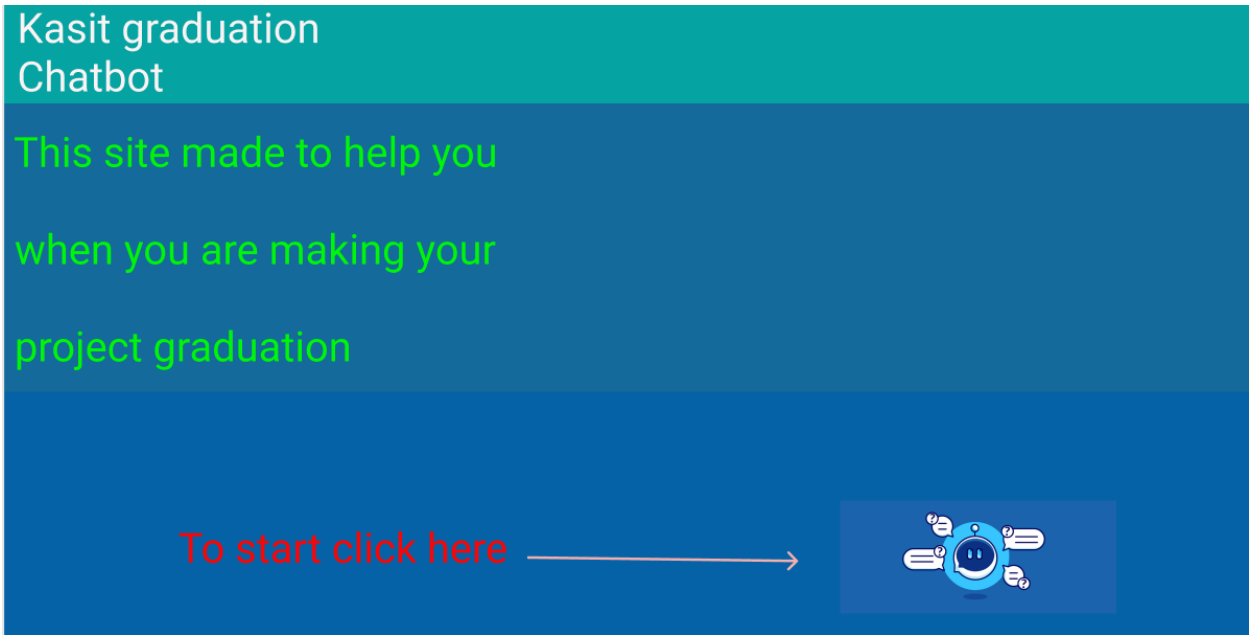


Figure 15: GUI - Home page

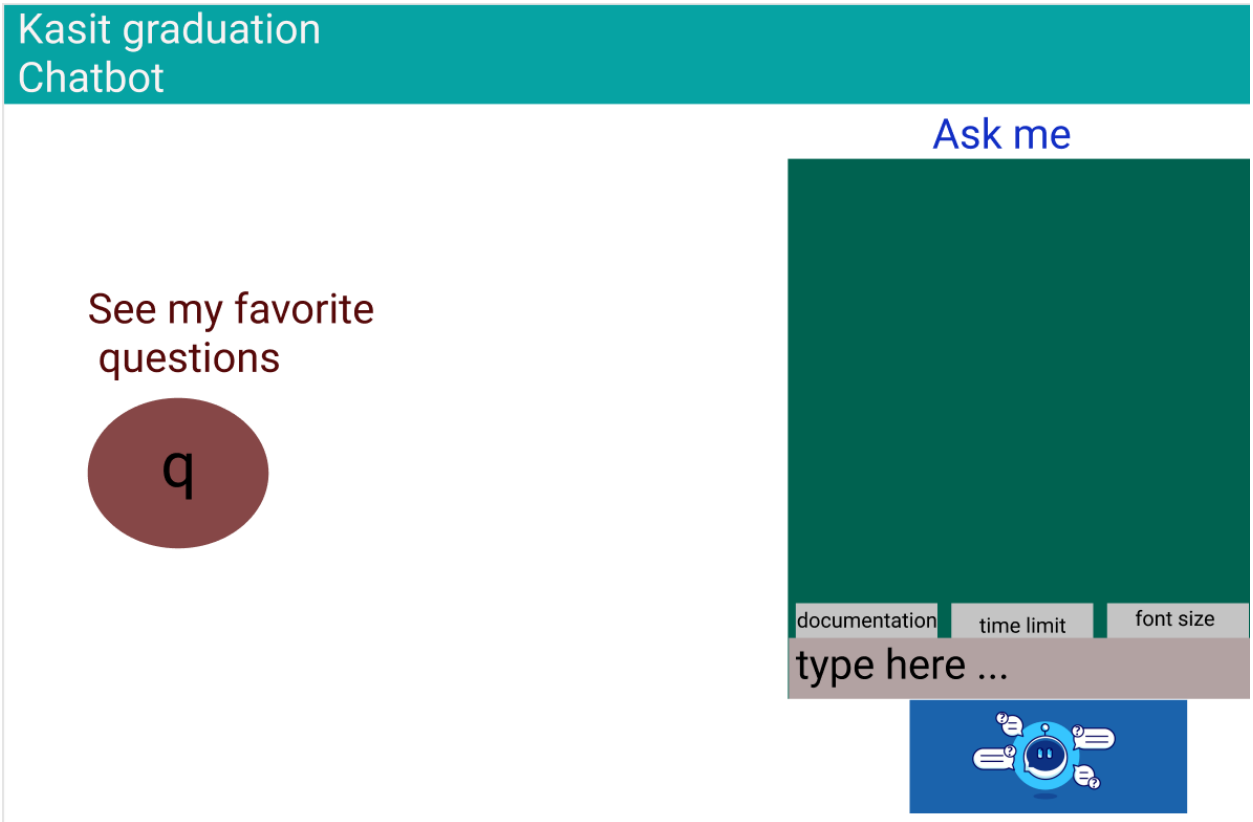


Figure 16: GUI - Chabot

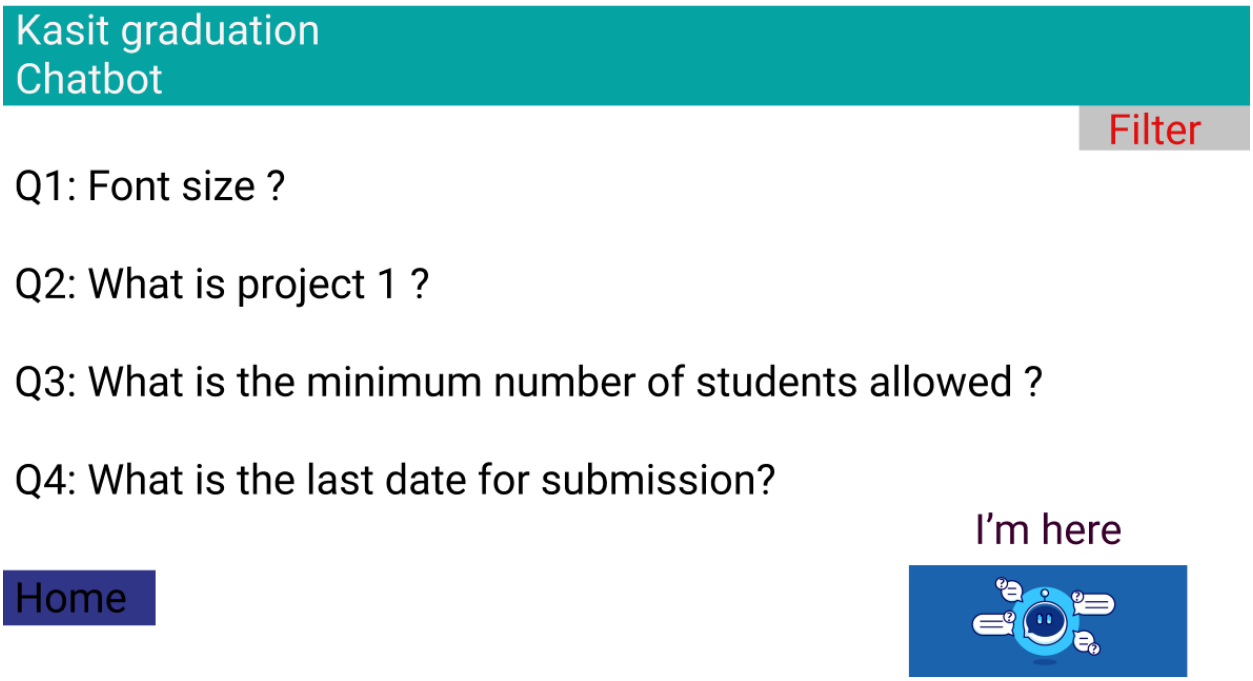


Figure 17: GUI – Frequency questions

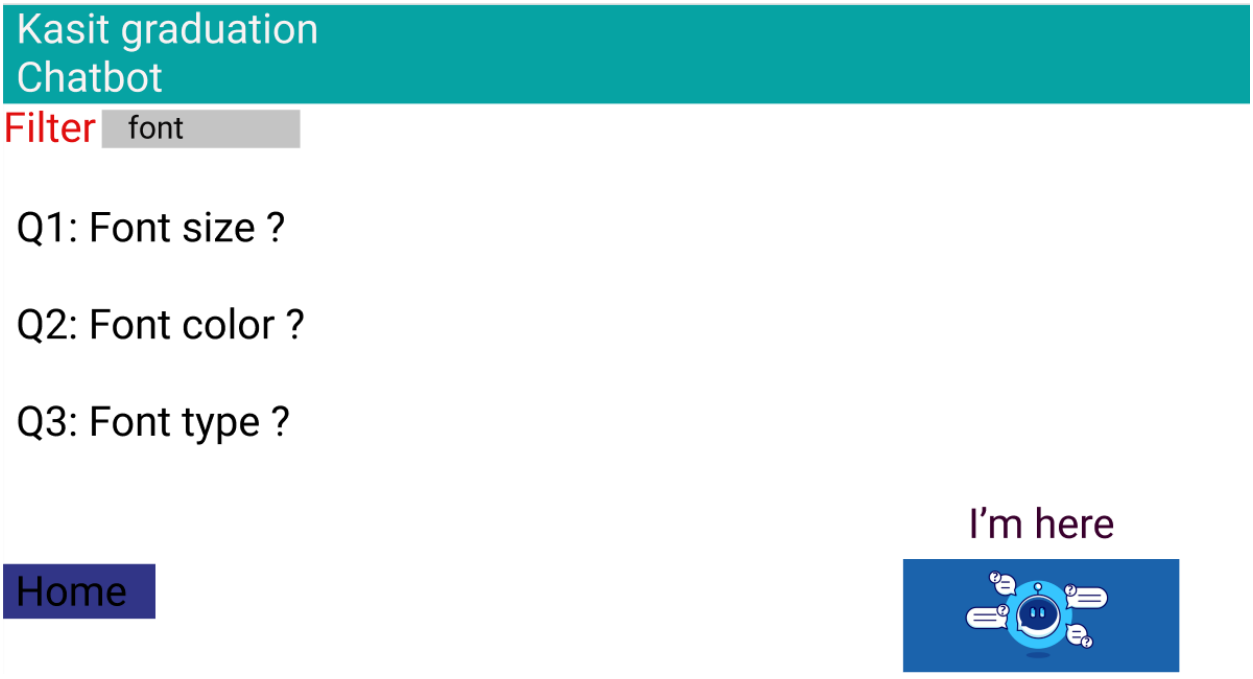


Figure 18: GUI –Filter questions

4.9 Summary

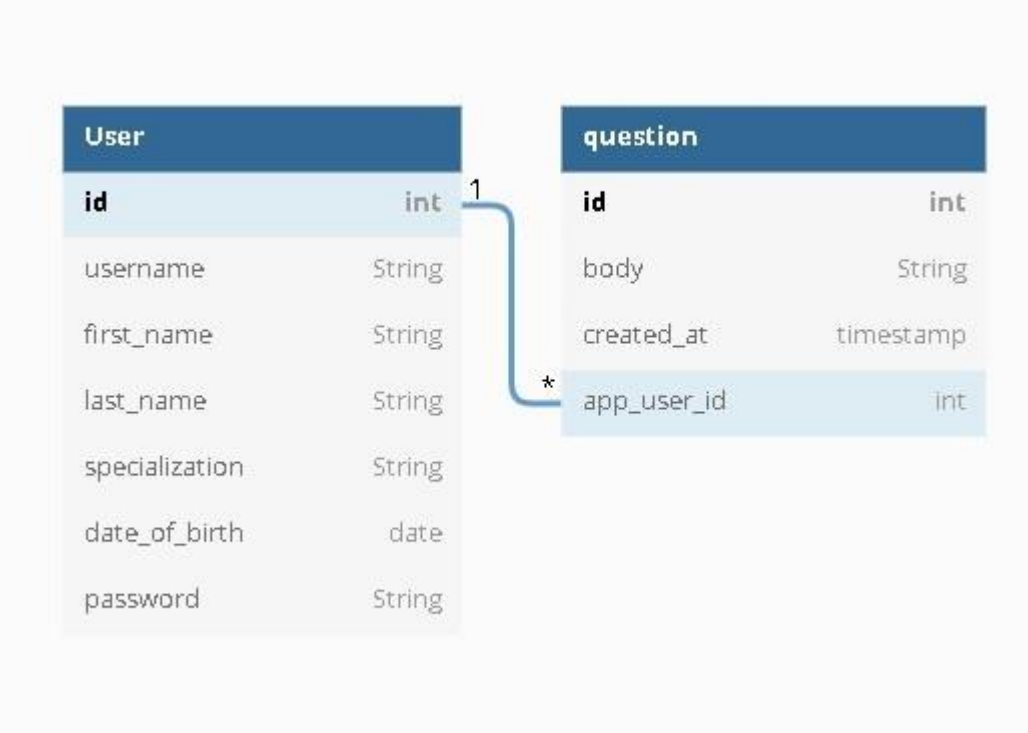
This chapter included all the required graphics for the system from the graphical user interfaces, Context Diagram, DFD, ERD and UML cases.

5.0 CHAPTER FIVE: SYSTEM IMPLEMENTATION

5.1 Introduction

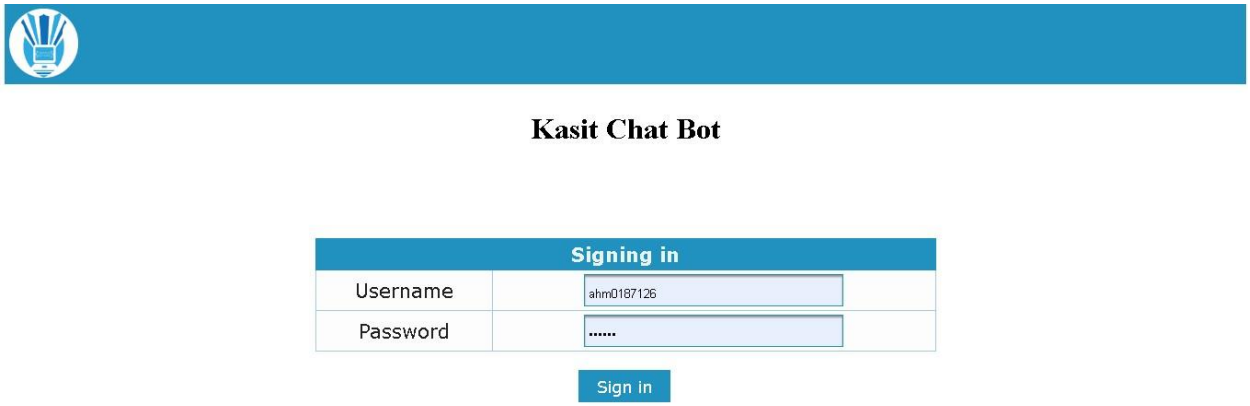
In this part we will look at the system implementation or in other words the database and backend (functions).

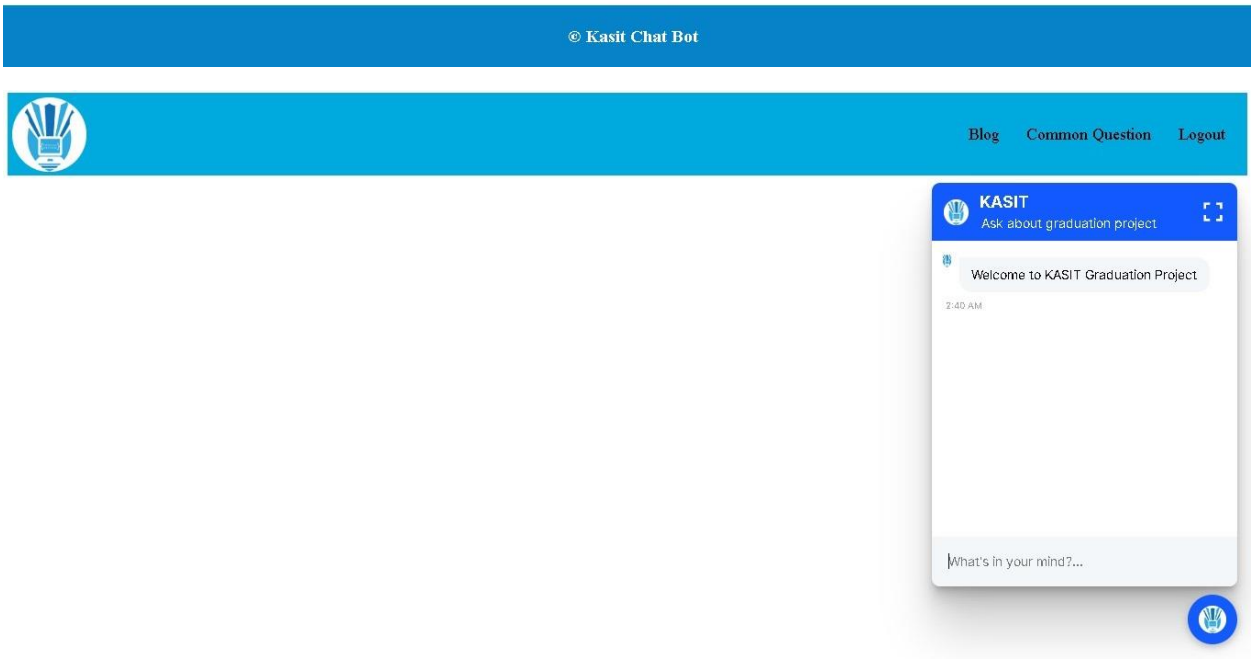
5.2 Database Implementation

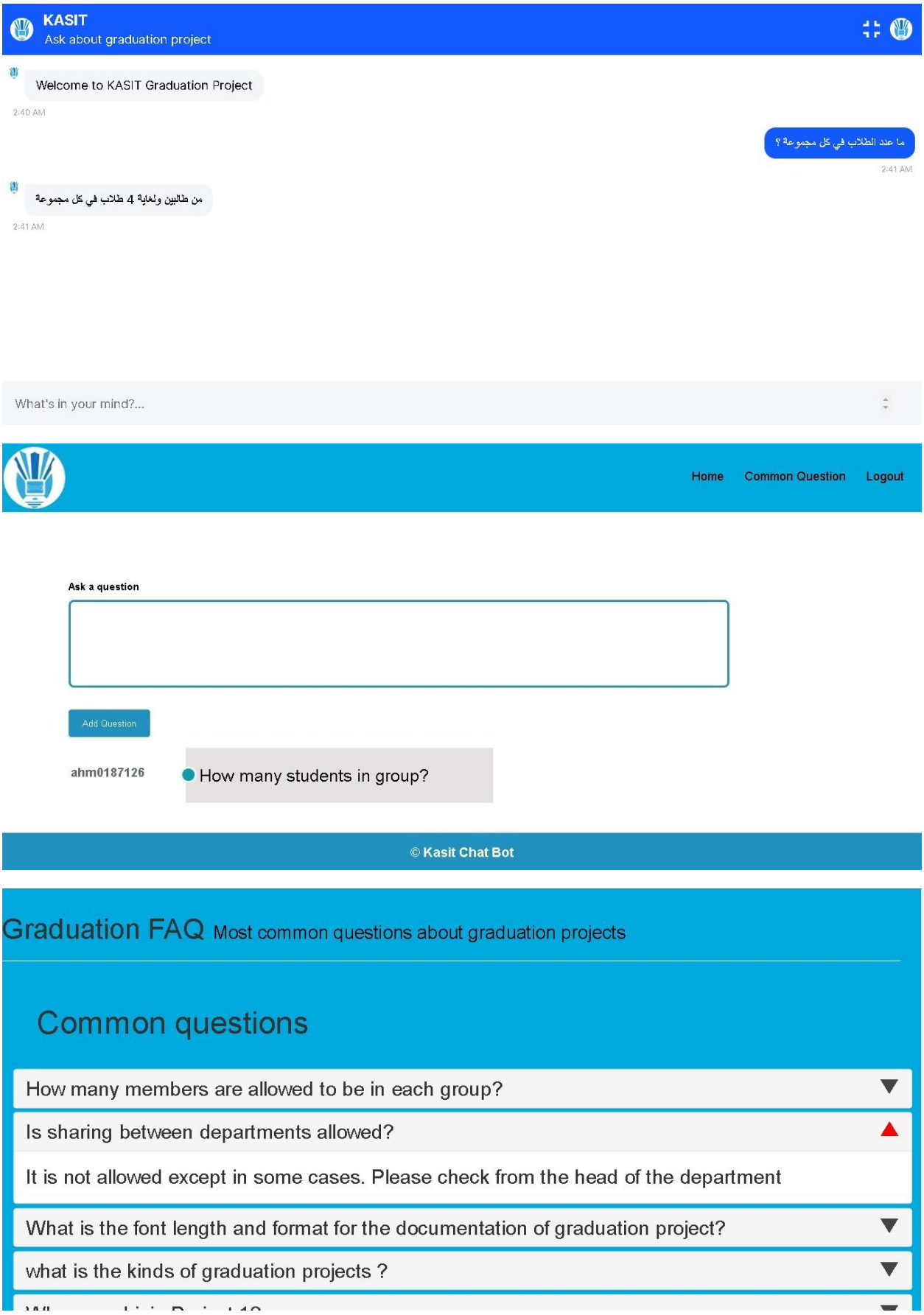


```
server.port=8080
spring.datasource.platform=postgres
spring.datasource.url=jdbc:postgresql://localhost:5432/graduate
spring.datasource.username=ahmad_
spring.datasource.password=0000
spring.jpa.hibernate.ddl-auto=update
spring.datasource.initialization-mode=always
```

5.3 Graphical User Interface Implementation






KASIT
Ask about graduation project
Welcome to KASIT Graduation Project
2:40 AM
ما عدد الطلاب في كل مجموعة ؟
2:41 AM
من طالبين ولغاية 4 طلاب في كل مجموعة
2:41 AM
What's in your mind?...
Home
Common Question
Logout
Ask a question
Add Question
ahm0187126
How many students in group?
© Kasit Chat Bot
Graduation FAQ Most common questions about graduation projects
Common questions
How many members are allowed to be in each group?
Is sharing between departments allowed?
It is not allowed except in some cases. Please check from the head of the department
What is the font length and format for the documentation of graduation project?
what is the kinds of graduation projects ?

5.4 Backend

```
    @GetMapping("/")                                                                    ⚠ 7
    public String HomePage() {
      return "login";
    }
    A7mad3bd
    @GetMapping("/chatbot")
    public String bot() { return "index"; }
    A7mad3bd
    @GetMapping("/signup")
    public String getSignupPage() { return "signup"; }
    A7mad3bd *
    @PostMapping("/signup")
    public String signupUser(@RequestParam String username,
                             @RequestParam String password,
                             @RequestParam String firstName,
                             @RequestParam String lastName,
                             @RequestParam String dateOfBirth,
                             @RequestParam String specialization
//                             @RequestParam Set<Role> role

                             ) {
        AppUser appuser = new AppUser(username, encoder.encode(password), firstName, lastName, dateOfBirth, specialization );
        appUseraRepository.save(appuser);
        return "login";
    }
    A7mad3bd *
    @GetMapping("/blog")
    public String comments(Model m, Authentication auth) {
        m.addAttribute( attributeName: "allquestions", reviewRepo.findAll());
        return "blog";
    }
    A7mad3bd
    @GetMapping("/common")
    public String common(Model m, Authentication auth) {
        return "Common";
    }
    A7mad3bd
    @PostMapping("/blog")
    public RedirectView addcomment(Principal p, String body, Model model) {
        AppUser user = (AppUser) appUseraRepository.findByUsername(p.getName());
        LocalDate createdAt = LocalDate.now();
```

```java
5 usages  A7mad3bd
@Entity
public class Question {
    2 usages
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    3 usages
    @CreationTimestamp
    private LocalDate createdAt;
    3 usages
    private String body;
    3 usages
    @ManyToOne
    AppUser appUser;

    4 usages  A7mad3bd
    public Question(String body, AppUser user, LocalDate createdAt) {
        this.body = body;
        this.appUser = user;
        this.createdAt=createdAt;
    }
```

```
protected void configure(HttpSecurity http) throws Exception {
    http.cors().disable().csrf().disable() HttpSecurity
            .authorizeRequests() ExpressionUrlAuthorizationConfigurer<...>.ExpressionInterceptUrlRegistry
            .antMatchers( ...antPatterns: "/").permitAll()
            .antMatchers( ...antPatterns: "/login*").permitAll()
            .antMatchers( ...antPatterns: "/signup*").permitAll()
//              .antMatchers("/chatbot*").permitAll()
            .antMatchers( ...antPatterns: "/css/styles.css").permitAll()
            .antMatchers( ...antPatterns: "/css/styleserror.css").permitAll()
            .antMatchers( ...antPatterns: "/css/blog.css").permitAll()
            .antMatchers( ...antPatterns: "/css/chat.css").permitAll()
            .antMatchers( ...antPatterns: "/logo.jpg").permitAll()
            .antMatchers( ...antPatterns: "/faculty.jpg").permitAll()
            .anyRequest().authenticated()
            .and() HttpSecurity
            .formLogin() FormLoginConfigurer<HttpSecurity>
            .loginPage("/login")
            .loginProcessingUrl("/perform_login")
            .defaultSuccessUrl("/chatbot")
            .failureUrl( authenticationFailureUrl: "/login")
            .and() HttpSecurity
            .logout() LogoutConfigurer<HttpSecurity>
            .logoutUrl("/preform_logout")
            .logoutSuccessUrl("/")
            .deleteCookies( ...cookieNamesToClear: "JSESSIONID");
```

```yaml
version: "3.1"

intents:
  - greet
  - goodbye
  - affirm
  - deny
  - mood_great
  - mood_unhappy
  - bot_challenge
  - new
  - research
  - applied
  - submit
  - upload
  - Doctors
  - publish
  - topics
  - members
  - sharing
  - content
  - font
  - join
  - kinds
  - grading
  - deadline
  - research1
  - applied1
  - submit1
  - upload1
  - Doctors1
  - publish1
  - topics1
  - members1
  - sharing1
  - content1
  - font1
  - join1
  - kinds1
  - grading1
  - deadline1
```

```
utter_applied:
- text: "Product-based planning means that you put the focus on tangible deliverables and outcomes, and plan the project from the perspectiv
        After choosing an idea and then implementing it by turning it into a software using code."

utter_submit:
- text: "Each group should submit two copies of the project's software (including source code files) and the documentation to the advisor. T

utter_upload:
- text: "Only one student uploads the final copy of the project to the e-learning system and sends another copy to the supervisor."

utter_Doctors:
- text: "All the graduation projects' defenses will be scheduled in the last day of classes. A complete schedule for all the defenses will b
In the project's defense, the advisor and the examiners are going to investigate the project's deliverables with the group. Then, they are goi
"
utter_publish:
- text: "On the eLearning website or college announcements"
- text: "http://computer.ju.edu.jo/ar/arabic/Home.aspx"

utter_website:
- text: "http://computer.ju.edu.jo/ar/arabic/Home.aspx"


utter_upload1:
- text: "يرفع فقط طالب واحد النسخة النهائية للمشروع
  على المكان المخصص للتسليم وارسال نسخة اخرى للمشرف"

utter_Doctors1:
- text:  "يحدد القسم مشرفي مشروع التخرج تلقائيا لكل فريق او يعرض الفريق فكرته
  على دكتور محدد و بعد موافقته يصبح هو مشرف الفريق ،
  حيث يتراوح عدد المشرفين من واحد الى اثنين لكل فريق "

utter_publish1:
- text: "على موقع ال elearning , اعلانات موقع الكلية"

utter_sharing1:
- text: "غير مسموح إلا ببعض الحالات يرجى التأكد من الشروط من رئيس القسم"

utter_content1:
- text: "6 نشابترات"
```

```
story: story16
steps:
- intent: greet
- action: utter_greet
- intent: sharing1
- action: utter_sharing1

story: story17
steps:
- intent: greet
- action: utter_greet
- intent: font1
- action: utter_font1

story: story18
steps:
- intent: greet
- action: utter_greet
- intent: members1
- action: utter_members1
```

```
<header>
 <script>!(function () {
    let e = document.createElement("script"),
      t = document.head || document.getElementsByTagName("head")[0];
    (e.src =
      "https://cdn.jsdelivr.net/npm/rasa-webchat@1.x.x/lib/index.js"),
      // Replace 1.x.x with the version that you want
      (e.async = !0),
      (e.onload = () => {
        window.WebChat.default(
          {
            selector: "#webchat",
            initPayload: "/greet",
            customData: { "language": "en" }, // arbitrary custom data. Stay minimal as this will be added to the socket
            socketUrl: "http://localhost:5005/",
            showFullScreenButton: true,
            title: 'KASIT',
            subtitle: 'Ask about graduation project',
            profileAvatar: "faculty.jpg",
            //embedded: true,
            showMessageDate: true,
            openLauncherImage: "faculty.jpg",
            closeImage: "faculty.jpg",
            inputTextFieldHint: "What's in your mind?...",
            customMessageDelay: (message) => {
              let delay = message.length * 40;
              if (delay > 3 * 1000) delay = 3 * 1000;
              if (delay < 800) delay = 800;
              return delay;
            },
            params: { "storage": "session" }
          },
          null
        );
      }),
      t.insertBefore(e, t.firstChild);
  })();
 </script>
```

```
    <title>Sign Up Page</title>
    <link th:href="@{/css/styles.css}" rel="stylesheet"/>
<!--    <script th:src="@{/js/myjs.js}"></script>-->
</head>
<body>
<form action="/signup" method="post">
    <label for="username">Username</label>
    <input type="text" name="username" id="username">
    <label for="password">Password</label>
    <input type="password" name="password" id="password">
    <label for="firstName">FirstName</label>
    <input type="firstName" name="firstName" id="firstName">
    <label for="lastName">LastName</label>
    <input type="lastName" name="lastName" id="lastName">
    <label for="dateOfBirth">DateOfBirth</label>
    <input type="dateOfBirth" name="dateOfBirth" id="dateOfBirth">
    <label for="specialization">Specialization</label>
    <input type="specialization" name="specialization" id="specialization">
    <input type="submit">
</form>
</body>
<footer>
```

5.5 Summary

In this chapter we showed the database implementation and graphical user interface implementation.

6.0 CHAPTER SIX: SYSTEM TESTING AND INSTALLATION

6.1 Introduction

System testing is performed on the entire system in the context of the user's or system's requirement specifications. System testing doesn't only test the design, but also the actual behavior and the believed expectations of the customer. This chapter presents the introduction in Section 6.1, Heuristic evaluation is elaborated in section 6.2, Section 6.3 Cooperative Evaluation, and Section 6.4 provides Test Cases. Furthermore, this chapter is summarized in Section 6.5.

6.2 Heuristic Evaluation

Heuristic Evaluation contributes to finding problems easily in user interface design and determines the level of acceptance in the design of user interface, this assessment was conducted by a small set of expert evaluators, who have examined the user interface. Such processes help prevent product failure. In an expert review, the reviewers already know and understand the heuristics. As a result, it is an informal usability inspection technique, and they are not required to use a specific heuristic to each problem that may be faced. Below is Table 6.1 that shows the list of heuristics of usability evaluation and their descriptions.

Num	Heuristic	Description
H1	**Visibility of system status**	The system should always keep users informed about what is going on, through appropriate feedback within a reasonable time.
H2	**Match between system and the real world**	The system should speak the user's language, with words, phrases, and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order
H3	**User control and freedom**	Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
H4	**Consistency and standards**	Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform Conventions

H5	**Recognition rather than recall**	Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
H6	**Flexibility and efficiency of use**	Accelerators – unseen by the novice user – may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
H7	**Aesthetic and minimalist design**	Dialogues should not contain information that is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
H8	**Aesthetic and minimalist design**	Dialogues should not contain information that is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
H9	**Help users recognize, diagnose, and recover from errors**	Error messages should be expressed in plain language (no codes), precisely indicating the problem, and constructively suggesting a solution.
H10	**Help and documentation**	Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user task, list concrete steps to be carried out, and not be too large.

Table 6.2.1: Heuristic Evaluation

Severity Rating	Description
0	I don't agree that this is a usability problem at all.
1	Cosmetic problem only: need not be fixed unless extra time is available on the project.
2	Minor usability problem: fixing this should be given low priority.
3	Major usability problem: important to fix, so should be given high priority.
4	Usability catastrophe: imperative to fix this before the product can be released.

Table 6.2.2: Severity Ratings and their Descriptions

NUM	Frequency	Ratio%
H1	5	7.57
H2	5	7.57
H3	5	7.57
H4	10	15.15
H5	7	10.60
H6	6	9.09
H7	8	12.12
H8	5	7.57
H9	5	7.57
H10	10	15.15

Table 6.2.3: Summary of Validation heuristic

Severity Rating	FREQUENCY	RATIO%
0	37	0.56
1	21	0.32
2	7	0.11
3	1	0.01
4	0	0.00
TOTAL	66	100%

Table 6.2.4: Summary of violations by severity ratings for participant 1

Severity Rating	FREQUENCY	RATIO%
0	24	0.364
1	24	0.364
2	18	0.272
3	0	0.00
4	0	0.00
TOTAL	66	100%

Table 6.2.5: Summary of violations by severity ratings for participant 2

Severity Rating	FREQUENCY	RATIO%
0	27	0.41
1	39	0.59
2	0	0.00
3	0	0.00
4	0	0.00
TOTAL	66	100%

Table 6.2.5: Summary of violations by severity ratings for participant 3

6.3 Cooperative Evaluation

After using the system and answering KASIT Chatbot System, please indicate the extent to which you agree or disagree with each of the following statements regarding to your experience with the system.

NUM	Criteria	Participant 1	Participant 2	Participant 3	Participant 4
1.	Gender	Male	Male	Male	Male
2.	Age	22	22	22	21
3.	Educational Level	Bachelor's degree	Bachelor's degree	Bachelor's degree	Bachelor's degree
4.	Programmer	Husam	Ahmed	Mohammed	Samer
5.	Institution	Kasit	Kasit	Kasit	Kasit

Table 6.3.1: Cooperative testing participant's data

A.	Participant (1)		
1.	Sign-in	20 Seconds	-
2.	View F&Q	50 Seconds	-
3.	Ask question	60 Seconds	-
4.	Add a post	120 Seconds	-
5.	Logout	4 Seconds	-

Table 6.3.2 Cooperative evaluation for participant 1

A.	Participant (2)		
1.	Sign-in	40 Seconds	-
2.	View F&Q	40 Seconds	-
3.	Ask question	50 Seconds	-
4.	Add a post	80 Seconds	-
5.	Logout	3 Seconds	-

Table 6.3.3 Cooperative evaluation for participant 2

A.	Participant (3)		
1.	Sign-in	16 Seconds	-
2.	View F&Q	30 Seconds	-
3.	Ask question	32 Seconds	-
4.	Add a post	60 Seconds	-
5.	Logout	1 Seconds	-

Table 6.3.4 Cooperative evaluation for participant 3

A.	Participant (4)		
1.	Sign-in	25 Seconds	-
2.	View F&Q	17 Seconds	-
3.	Ask question	30 Seconds	-
4.	Add a post	70 Seconds	-
5.	Logout	5 Seconds	-

Table 6.3.5 Cooperative evaluation for participant 4

After our four participants finished testing our application, each one due their rule, we asked them to answer a follow-up questionnaire about their opinion and experience in general with the system.

No	Statement	P 1	P 2	P 3	P 4
1	Chatbot is easy to use.	4	5	4	4
2	Chatbot has accomplished its goals.	4	4	5	5
3	Chatbot Interface is Interactive.	5	5	4	4
4	It is easy to understand the Functionality of the website without prior experience	4	3	4	5
5	Chatbot is an enjoyable website.	5	5	4	4
6	The Concept of Chatbot was difficult to understand.	2	2	1	2
7	Chatbot complete.	3	3	2	2
8	I felt that the tasks were difficult to complete using the website.	2	2	1	2
	Average	3.63	3.63	3.13	3.5

Table 6.3.6 Cooperative testing participant's data

6.4 Requirements Validation and Completeness

Upon completion of the design and implementation of the code complete, here comes the software product evaluation process to ensure that the software meets the specified and predetermined requirements as well as the requirements and expectations of theend users. After completing the evaluation process, we came to these cases.

Experimental cases:

Scenario 1 -: Log in to the system as a user.

- When the user enters a valid username in the username text box, the user enters a valid password in the password text box and clicks the login button. Then the portal page opens

Passed

successfully.

What could go

wrong?

User entered invalid username.

Expected result: An error message is displayed.

Scenario 2 -: Ask Chatbot.

Go back to chapter 5 then go to 5.3

Scenario 3 -: Add questions.

Go back to chapter 5 then go to 5.3

6.5 Summary

This chapter showed a testing and evaluation for KASIT Chatbot System. The heuristic evaluation was conducted with 3 expert users. The Heuristic and cooperative evaluation have also shown competitive and acceptable performance for the system indicating that the system is easy to use and has less usability problems.

7.0 CHAPTER SEVEN: PROJECT CONCLUSION AND FUTURE WORK

7.1 Introduction

We found that hard to meet all of user needs or requirements, but we aim to update our project and functionality of it and fix the weaknesses in our website. In this chapter we will present introduction, overall weaknesses and strength points, future work and finally summarize the chapter.

7.2 Overall Weaknesses

The main weakness points in our project are:

1-Sometimes the Chatbot does not understand the question.

2-In blog page everyone can ask questions and answer on it even if the answers are wrong.

7.3 Overall Strengths

1-The Chatbot supports multiple languages.

2-The Chatbot can identify the question even if it has spelling errors.

7.4 Future Work

We would like to add more features to this project that will make the user experience even better and easier such as: Making the Chatbot available to all faculties, adding the question and answer feature using voice and making a specialized person to answer questions for students in the blog.

7.5 Summary

This chapter summarize the points of weakness and strength in our Chatbot and future work and some of updates which we planned for them.

REFERENCES

1.0 Figma design website

2.0 https://app.diagrams.net/

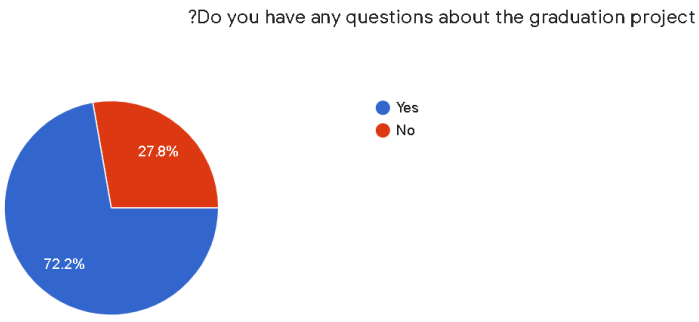
3.0 Creately Design

4.0 Spring Boot

5.0 Open source conversational AI | Rasa

APPENDICES

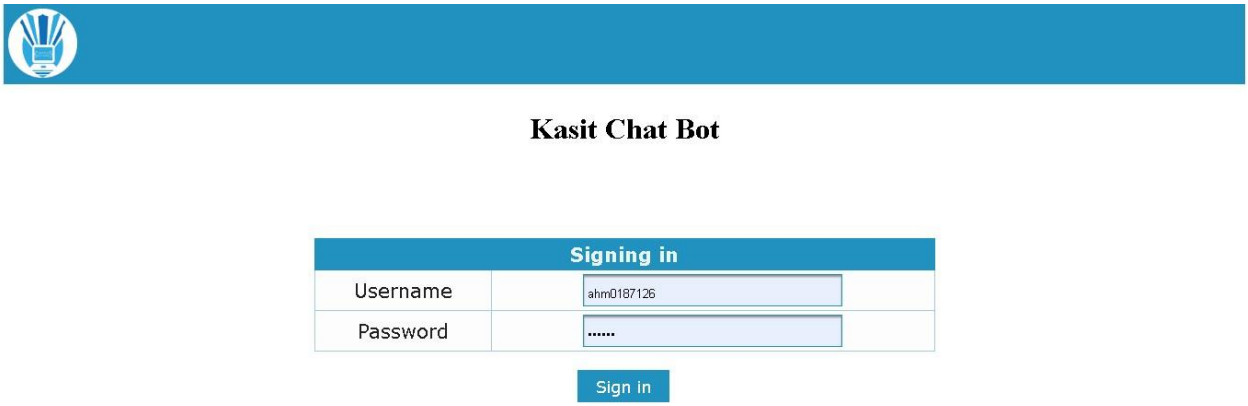
Appendix A: Questionnaire



Appendix B: User Manual

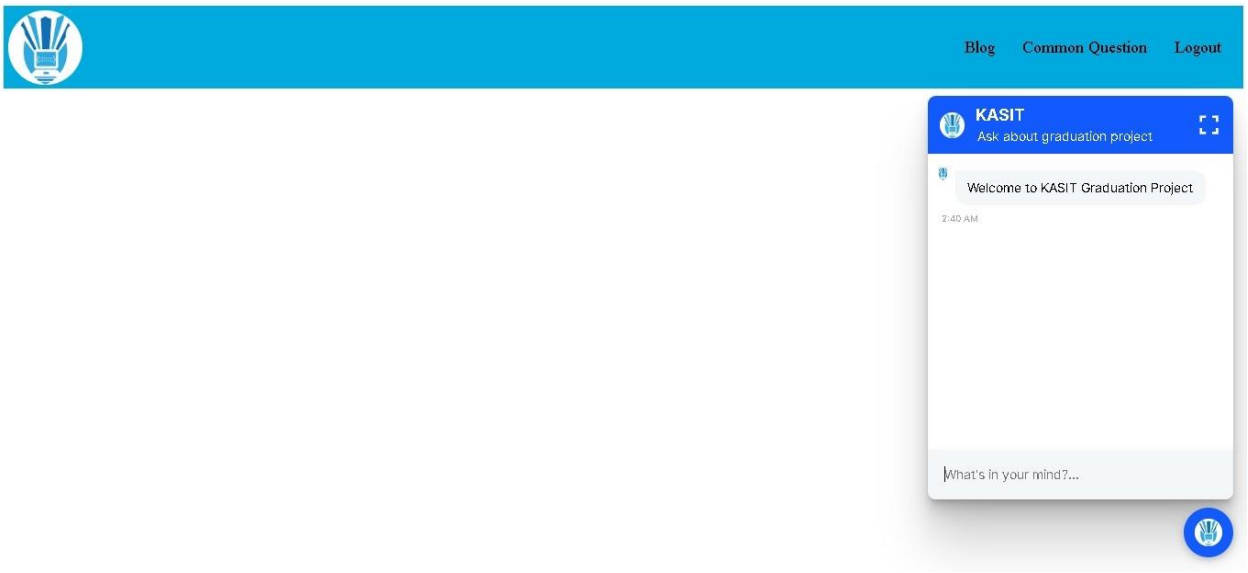
1.Sign in using your user name and password that you used to enter on e-earning platform.

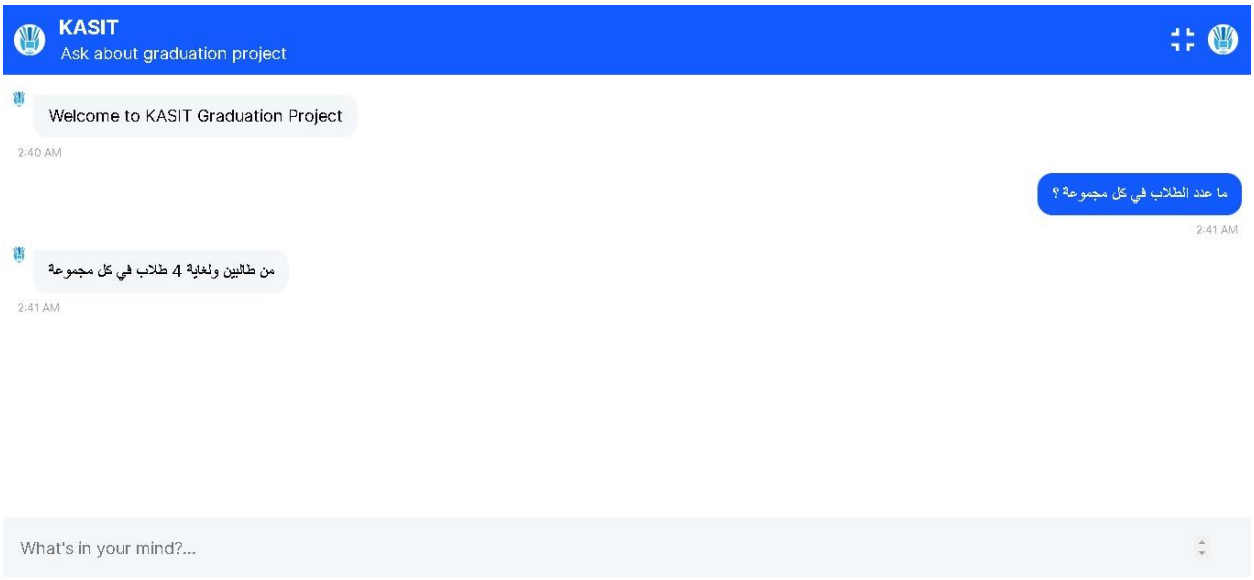
See the next figure:



2.Click on Chatbot icon to ask the question you want about graduation project.

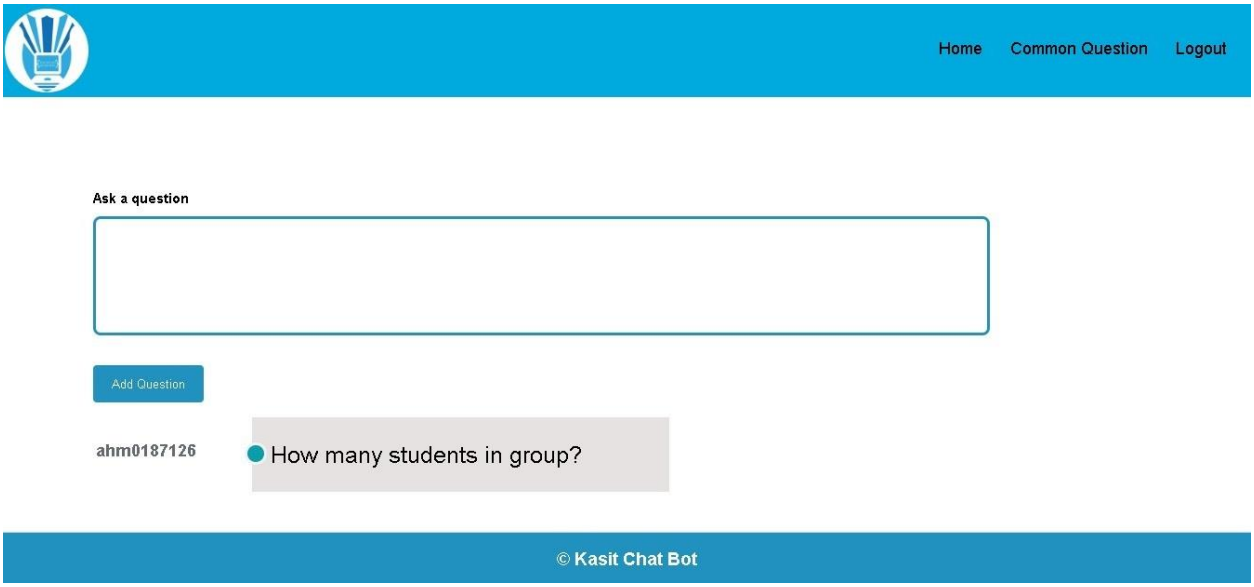
See the next figures:

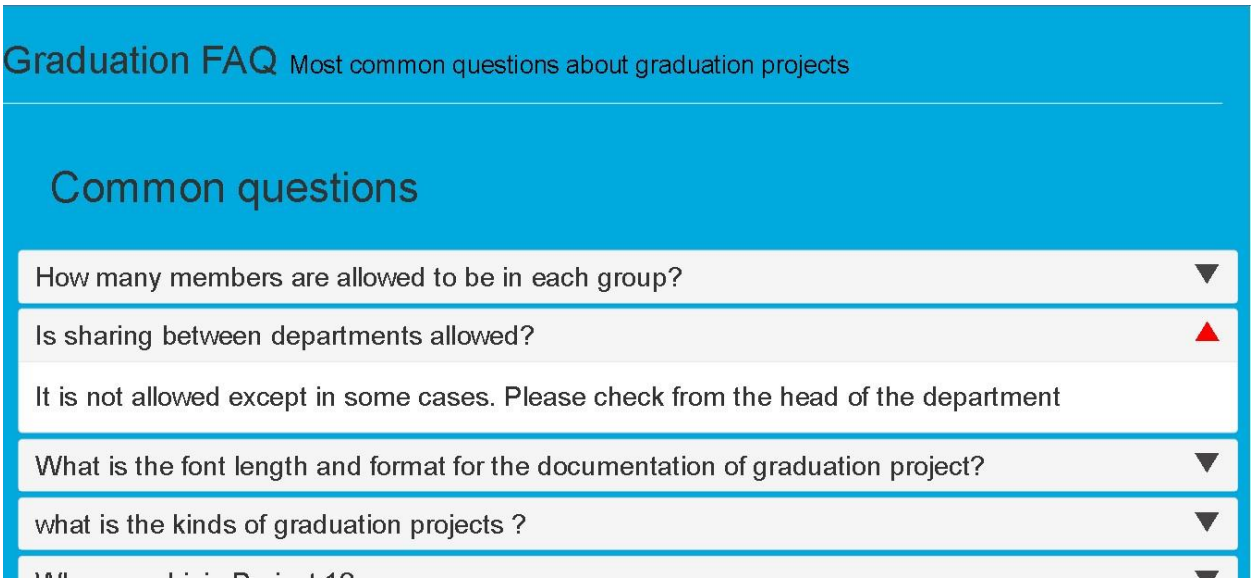




3.Click on Blog button to ask any question you want if you don’t find your question inside Chatbot.

See the next figure:



4.Click on Common Question to see the most trending questions.

See the next figure: