جامعة الملك سعود
King Saud University
كلية علوم الحاسب والمعلومات
قسم تقنية المعلومات

Wjjhni وجّهني

IT 497: Graduation Project Report
Product Release-2

Prepared by
Shaden Alshabanat, 441201170
Mona Alafari, 438202881
Esraa Alhasan 439201131
Ibtihal Almutairi, 442202323
Lamia Alhelali, 442200388

Supervised by
Dr.Hend Al-Bassam

Second Semester 1445
Spring 2024

Table of Contents

Table of Figures

Table of Tables

كلية علوم الحاسب والمعلومات
قسم تقنية المعلومات

Wjjhni-وجّهني

Shaden Alshabanat [1], Ibtihal Almutairi [2], Lamia Alhelali [3], Esraa Alhasan[4], Mona Alafari [5]

[1]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441201170@student.ksu.edu.sa
[2]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 442202323@student.ksu.edu.sa
[3]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 442200388@student.ksu.edu.sa
[4]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 439201131@student.ksu.edu.sa
[5]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 438202881@student.ksu.edu.sa

Abstract (English):

Academic advising plays a vital role in students' educational journeys. However, traditional advising methods face significant challenges in meeting the diverse needs and expectations of students, leading to frustration and delays in academic progress. To address these issues, we have developed a software system called "Wjjhni." The purpose of this system is to provide timely guidance to students by combining the power of artificial intelligence with user-friendly interfaces that can be accessed anytime and anywhere. The main methodology for the development of “Wjjhni” involved creating an AI chatbot using the IBM Watson Assistant platform, which answers frequent questions about courses, university rules, and guidelines. The system was evaluated through interviews, questionnaires, and comparative analyses with other universities to ensure the chatbot's effectiveness in addressing frequently asked questions. The evaluation revealed that “Wjjhni” significantly improved communication between students and advisors, enhanced access to vital information, and facilitated course planning and scheduling. The main results indicate that the system successfully addresses the challenges faced by students and advisors, providing a valuable tool for streamlining academic advising processes. The conclusions drawn from this work emphasize the importance of integrating AI technology into educational systems to enhance student support and improve their overall educational experience.

Abstract (Arabic):

يلعب الإرشاد الأكاديمي دورًا حيويًا في رحلة التعليم للطلاب. ومع ذلك، تواجه أساليب الارشاد التقليدية تحديات كبيرة في تلبية احتياجات الطلاب المتنوعة وتوقعاتهم المختلفة، مما يؤدي إلى الإحباط والتأخير في التقدم الأكاديمي. لمعالجة هذه المشكلات، قمنا بتطوير نظام برمجي يسمى "وجّهني". يهدف هذا النظام إلى توفير إرشادات في الوقت المناسب للطلاب عن طريق دمج قوة الذكاء الاصطناعي مع واجهات سهلة الاستخدام يمكن الوصول إليها في أي وقت ومن أي مكان. تتضمن المنهجية الرئيسية لتطوير "وجّهني" إنشاء دردشة آلية تعتمد على منصة IBM Watson Assistant، والتي تقدم إجابات على الأسئلة الشائعة حول المقررات وقواعد الجامعة والمبادئ التوجيهية. تم تقييم النظام من خلال إجراء مقابلات واستبيانات وتحليلات مقارنة مع جامعات أخرى لضمان فعالية الدردشة الآلية في التعامل مع الأسئلة الشائعة. أظهر التقييم أن "وجّهني" يحسن بشكل كبير التواصل بين الطلاب والمرشدين، ويعزز الوصول إلى المعلومات الهامة، ويسهل تخطيط وجدولة المقررات. تشير النتائج الرئيسية إلى أن النظام يعالج بنجاح التحديات التي يواجهها الطلاب والمرشدون، ويوفر أداة قيمة لتبسيط عمليات الإرشاد الأكاديمي. تشدد الاستنتاجات المستخلصة من هذا العمل على أهمية دمج تكنولوجيا الذكاء الاصطناعي في الأنظمة التعليمية لتعزيز دعم الطلاب وتحسين تجربتهم التعليمية بشكل عام.

Keywords:

Academic advising, student, advisor, chatbot, education.

1 Introduction

1.1 Problem

At our institution, King Saud University (KSU), students frequently encounter an inefficient course registration process and limited support from academic advisors, as well as not having access to the answers to questions they have about the university rules and guidelines. Moreover, students who have delays in their academic journey need extra help and advice and that's not always available for them. Furthermore, the students sometimes won't know what their courses require as in other courses or knowledge, they also face difficulty finding an appropriate time to discuss and communicate with their academic advisors even when they need them urgently. As with academic advisors, they don't know the history of the student if the student had a different academic advisor for some time beforehand, which could cause communication issues and they might give an unfit advice.

These issues lead to frustration, confusion, and delays in academic progress, ultimately impacting students' overall educational experience since they can't locate important information that might aid them in knowing what they need in several different manners which will make them feel lost and negatively affect their academic goals.

1.2 Objectives as the solution

- Product (customer focus-value):

Our objective is to develop an academic advising mobile application "Wjjhni" that facilitates the academic advising process and provides effective academic advising services. This application will offer several features including:

1- Provide the students with an AI-based chatbot that answers their academic questions.

2- Provide the advisors with the ability to add and edit their availability so the student can book a meeting without delays or asking.

3- Provide the students with the ability to book advising appointments with academic advisors online, without chatting and asking the advisor about which time is suitable.

4- Provide the students with a page containing all the academic advising forms, to save time and effort.

5- Provide the students and the advisors with a page that contains all the department plans for each major.

6- Provide the students and the advisors with a calendar of important dates, such as the last day of the “add-drop course”.

7- Provide an online chat between the student and her advisor, to make it easier for them to contact each other.

8- Provide the students with a Calculator that calculates the absence hours allowed for a course, based on the courses’ hours and other things. So, they don’t exceed the maximum absence limit.

9- Providing the advisor with the ability to record each student visit, so if the advisor was changed, the new advisor can know what the student has been through and what her academic past is.

10- Providing the students and advisors with the ability to update their profiles.

11- Provide the students with the ability to evaluate their academic advisors, so the administration can know if the advisor was efficient or not.

12- Provide The admins a dashboard to manage database operations, such as editing the plans and adding/deleting forms.

13- Provide the admins with the ability to add and delete students and academic advisors to the system.

14- Provide the admin with the ability to assign academic advisors to students

Wjjhni is an academic mobile application that supports Arabic language only and will be limited to the Android System. It aims to help female students and academic advisors of the College of Computer and Information Sciences at King Saud University. Conversely, it will not cover all the colleges of King Saud University.

Wjjhni will assist the academic advising process by providing accessible advising services such as allowing the advisor to set up their available time slots. Hence, the student can check availability and book an advising appointment. In addition, the AI chatbot helps students by

answering their questions. The students can ask the AI chatbot to answer their inquiries about university rules.

It will enable instant messaging between the advisor and student, so they communicate in real time with no delays such as those associated with email communication. Wjjhni will contain all the study plans for each department of CCIS so students and advisors can interactively view them. They can use an academic calendar to know about significant days plus an absence calculator.

Finally, students and advisors can log in after adding them by the admin and view their profiles. The admin will have a web dashboard to manage database operations such as adding, deleting, and updating the content.

Product Vision

For academic advisors and students
Who needs to facilitate and automate the academic advising process.
The Wjjhni is an Android mobile application.
That facilitates the academic advising process and provides effective academic advising services.
Unlike Edugate System
Our product will provide an AI based chatbot that offers guidance for students.

The approach for building the "Wjjhni" academic advising software involves several steps and stages, following a systematic software development process. The methodology used for this project is Agile, specifically Scrum, which emphasizes iterative development and collaboration between the development team, stakeholders, and end-users. steps involved in building “wjjhni”:

1. Requirements Gathering and Analysis:

- Understand user needs and expectations by conducting interviews, surveys, and observations.

- Collect requirements from academic advisors, students, and administrators to identify key features and functionalities of the application.

- Analyze the collected data to prioritize requirements and define the scope of the project.

2. System Design:

- Design the architecture and components of the application based on the gathered requirements.

- Create wireframes and prototypes to visualize the user interface and user experience.

- Define the database structure and relationships to store and retrieve data effectively.

3. Development Iterations:

- Adopt an iterative and incremental approach to development.

- Divide the project into multiple development iterations or sprints, each lasting a fixed duration.

- Prioritize and select a set of requirements to implement in each iteration.

4. Implementation:

- Develop the application using appropriate programming languages, frameworks, and technologies.

- Implement the user interface based on the designed wireframes and prototypes.

- Integrate necessary APIs and services, such as the IBM Watson Assistant API for the AI chatbot functionality.

- Build the backend logic and connect it to the database.

5. Testing and Quality Assurance:

- Conduct comprehensive testing throughout the development process.
- Perform unit testing to ensure individual components and functions work correctly.
- Conduct integration testing to verify the interaction between different modules and components.
- Perform system testing to validate the application.
- Use automated testing tools and manual testing techniques to identify and fix bugs and issues.

6. Deployment and Evaluation:

- Deploy the application to a staging environment for further testing and evaluation.
- Gather feedback from end-users, including students, advisors, and administrators.
- Assess the effectiveness and usability of the application based on user feedback.
- Make necessary improvements and refinements based on the evaluation results.

7. Maintenance and Iterative Enhancements:

- Provide ongoing maintenance and support for the application after deployment.
- Address any issues or bugs reported by users promptly.
- Continuously gather user feedback and iterate on the application to enhance its features and functionalities based on user needs and emerging requirements.

The main contribution of the "Wjjhni" academic advising application lies in its comprehensive and user-friendly solution to the challenges faced by students and academic advisors. The

application is designed specifically for female students and academic advisors of the College of Computer and Information Sciences (CCIS) at King Saud University. It aims to address the limitations of traditional advising methods and provide an efficient and accessible platform for academic support.

The application offers a range of features to enhance the advising process. Advisors can set up their available time slots, allowing students to conveniently check availability and book advising appointments. Additionally, an AI chatbot component that answers students' questions, answering inquiries about university rules. This feature enables students to receive quick responses to their questions.

One of the key advantages of the "Wjjhni" application is its emphasis on real-time communication. By incorporating instant messaging functionality, the application enables advisors and students to communicate without delays typically associated with email exchanges. This feature facilitates efficient and timely interactions, ensuring that students receive the support they need promptly. The application also includes study plans for each department of CCIS, an academic calendar, and an absence calculator. These features assist students and advisors in their academic planning and scheduling, providing them with essential information and tools to make informed decisions. Furthermore, user profiles and an admin dashboard are available, allowing users to manage their information, while the admin can handle database operations.

The impact of the "Wjjhni" solution extends to the local and global community. Locally, the application aims to improve the academic experience, increase student satisfaction, and contribute to better academic performance for CCIS students. By addressing the specific needs of this community, it can have a direct and positive impact on the students and advisors at King Saud University.

On a global scale, the "Wjjhni" application can serve as a model for similar academic advising systems in other universities. Its comprehensive features, user-friendly interface, and integration of AI technology through the chatbot component make it a novel solution. By promoting efficient and personalized advising practices, the application has the potential to inspire similar solutions worldwide, leading to improved academic support and enhanced student experiences in universities globally.

Overall, the "Wjjhni" offers a unique and impactful solution to the challenges faced by female students and academic advisors in the CCIS at King Saud University. Its comprehensive features, emphasis on real-time communication, and potential for global adoption make it a valuable tool for enhancing academic advising practices and improving the overall academic experience for students.

The report is organized into several sections. It begins with an introduction, followed by a background section providing context for the topic. A literature review summarizes relevant research, while the system design and development section delve into the methodology, system requirements, and design specifics like architectural diagrams and data models. The implementation and evaluation of the system are covered, including results and user acceptance testing and discussion. The report concludes by outlining conclusions and potential future work. It also includes acknowledgments, references, and an appendix for supplementary information.

2 Background

In this comprehensive background, we embark on a journey to explore the multifaceted realm of academic advising, with a particular focus on its application in the esteemed King Saud University (KSU) and, more specifically, within the Computer Science and Information College (CCIS). Section 2.1 initiates our exploration by delving into the fundamental aspects of academic advising, elucidating its significance, and outlining the services it provides.

The subsequent sections provide an in-depth examination of academic advising within Saudi Arabia and KSU, highlighting its adaptation to the digital age, its specialized approach in the CCIS, and the timeline followed by advisors. As we transition to Section 2.2, our focus shifts to the innovative use of chatbots in academic advising.

We unravel the intricacies of chatbot classification, emphasizing the hybrid system employed for helping CCIS students at KSU to find a faster way to answer their common questions. Section 2.3 introduces IBM Watson as the chosen platform for implementing the chatbot, offering a comprehensive overview of its functionality, from intent and entity recognition to training with examples, machine learning, integration with external systems, and deployment.

This comprehensive background lays a solid foundation for our upcoming endeavors in developing our system, encompassing both the application and website. By delving into the intricacies of academic advising and the innovative integration of technology, particularly chatbots, at King Saud University's Computer Science and Information College, we gain invaluable insights that will greatly contribute to the development and enhancement of our system.

2.1 Academic Advising

Academic advising is an indispensable facet of higher education across the world, acting as a vital support structure for students in their academic and professional endeavors. It involves a collaborative relationship between an advisor and a student, providing guidance, resources, and support throughout their academic journey. We will delve into the meaning of academic advising, its importance in universities, and its current application in Saudi Arabia, with a specific focus on King Saud University.

2.2 Academic Advising Definition and Significance

Academic advising systems are meticulously crafted to facilitate the academic journey. They provide essential tools and platforms that streamline communication, track academic progress, and grant access to relevant resources. These systems act as cornerstones, enhancing the efficiency and effectiveness of academic advising services. By centralizing information, they allow for seamless communication and provide timely, accurate guidance.

The crux of academic advising lies in its ability to help students navigate the intricate landscape of higher education. It equips students with the means to make informed decisions about course selection, to explore various academic opportunities, and to tackle challenges that may arise. Academic advisors offer a wide array of services, including aiding with course planning, elucidating degree requirements, offering career guidance, supporting personal and academic development, and linking students with valuable campus resources. By fostering an environment of support and inclusivity, academic advising becomes a catalyst for student success, encouraging engagement and holistic growth. This commitment ensures that every student receives the guidance and support required to flourish academically and to realize their full potential.[1]

On the other hand, academic advising is a dynamic and personalized process that aims to guide students in making informed decisions about their academic and career paths. It establishes a collaborative relationship between advisors and students, providing a wealth of guidance,

resources, and support throughout the academic journey. The primary objectives of academic advising encompass an array of essential facets.

Advisors help students choose courses and academic programs that align with their interests, strengths, and goals, ensuring they meet the requirements of their chosen majors or degree programs. Additionally, advisors monitor students' academic progress, helping them uphold necessary academic standards and providing support when challenges arise. The collaborative spirit of academic advising encourages the setting of short-term and long-term academic and career goals, fostering personal and intellectual growth. Academic advisors provide crucial career counseling, assisting students in exploring diverse career paths, industries, job opportunities, and connections. Personal development, time management, study skills, and strategies for balancing academics and life are also areas advisors address, fostering well-rounded development.

The referral to campus resources and problem-solving assistance are further testaments to the comprehensive support advisors offer. The assistance doesn't halt there; it extends to degree and graduation planning to ensure students meet requirements and avoid unnecessary delays. The importance of academic advising cannot be understated, as it provides the means for students to effectively navigate higher education's intricacies. It ensures students have access to the guidance and support required to flourish academically and realize their full potential.

2.3 Academic Advising in Saudi Arabia

In Saudi Arabia, academic advising is a pivotal element of the higher education system, and King Saud University (KSU), one of the leading universities in the region, has a well-established academic advising program. Academic advising in Saudi Arabia aligns with international practices, striving to help students make informed decisions about their academic and career paths while fostering holistic development. KSU's academic advising system is structured to cater to the unique needs of students. Advisors, often faculty members or professionals well-versed in the university's programs and policies, provide personalized academic advising services through various colleges and departments. They assist students in

selecting the appropriate courses, understanding degree requirements, and setting academic and career goals. This support extends to personal development and problem-solving, ensuring students receive the guidance and resources essential for their academic success [[2].

Furthermore, King Saud University has adapted to the digital age, embracing technology and online advising to provide additional support and convenience to students. Digital platforms facilitate communication between advisors and students, making it easier for students to access guidance and academic resources. In conclusion, academic advising plays a pivotal role in the academic journey, ensuring students have access to the guidance and support needed for academic success and personal development [3].

In the Computer Science and Information College at King Saud University, academic advising takes on a tailored approach to address the specific needs and objectives of students pursuing degrees in these fields. Computer science and information technology are rapidly evolving domains, and academic advisors within the college are well-equipped to guide students through the dynamic landscape of technology and innovation.

The academic advising process within the Computer Science and Information College at King Saud University can go beyond the standard academic guidance. In this setting, advisors will be able to work closely with students to help them understand the latest trends and developments in computer science and information technology. They assist students in selecting courses and projects that align with their career aspirations and the ever-changing demands of the tech industry.

Moreover, advisors could emphasize the importance of hands-on experience, encouraging students to participate in internships, research projects, and collaborations with industry partners. This practical exposure ensures that graduates not only have a strong theoretical foundation but are also well-prepared for the challenges of the real-world technology landscape.

The support provided by academic advisors can extend to helping students stay updated with industry certifications and emerging technologies that they keep students informed about relevant workshops, conferences, and networking opportunities. By fostering a culture of continuous learning and adaptability they empower students to become not just graduates but tech-savvy professionals who can make significant contributions to the ever-evolving field of computer science and information technology. Through this specialized approach, academic advising at King Saud University's Computer Science and Information College can play a pivotal role in preparing students to thrive in a tech-driven world.

Academic advising at KSU and CCIS.

Within the Collage of Computer and Information Sciences (CCIS) at King Saud University, academic advising operates on a well-defined timeline, ensuring that students receive tailored support at various stages of their academic journey. The responsibilities of academic advisors are delineated across a semester timeline, providing clarity and structure to the advising process.

At the beginning of each semester, advisors engage in comprehensive orientation sessions to acquaint students with the academic resources available, explain degree requirements, and discuss the importance of academic planning. This initial phase sets the foundation for a successful academic experience [4]

During the course registration period, advisors play a crucial role in helping students select appropriate courses aligned with their academic goals and degree requirements. They guide students through the intricacies of the curriculum, ensuring a cohesive and well-rounded course selection that fosters both specialization and breadth of knowledge.

Midway through the semester, advisors conduct check-in meetings to assess students' academic progress, addressing any challenges they may be facing. These sessions provide an opportunity for proactive intervention, whether it be additional academic support, clarification of course material, or assistance in navigating personal challenges that may impact academic performance.

As the semester approaches its conclusion, advisors shift their focus to degree planning. They work closely with students to review their academic trajectory, verify that degree requirements are being met, and plan for subsequent semesters. This proactive approach minimizes the risk of delays in graduation and ensures that students are on track to fulfill all necessary criteria for their chosen degree.

In addition to the semester-based timeline, academic advising within the CCIS adheres to a broader degree timeline. This encompasses the entirety of a student's academic journey, from orientation to graduation. Throughout this continuum, advisors provide ongoing support and guidance, emphasizing the importance of long-term academic and career planning.

It's essential to note that while the overarching principles of academic advising remain consistent within the CCIS, there may be department-specific nuances. Different departments within the college might have unique requirements, industry partnerships, or internship opportunities. Advisors, therefore, stay informed about department-specific details to offer tailored guidance to students pursuing degrees in computer science and information technology.

2.4 Data Collection

Regarding the data collection used for training the chatbot, the survey functions as a valuable tool to collect input from students regarding their advising experiences. The gathered data encompasses students' satisfaction levels, perceived effectiveness of advising sessions, and suggestions for improvement. This information proves invaluable for the ongoing

enhancement of the academic advising program, facilitating iterative improvements guided by real-time student feedback.

Furthermore, the collected data also aid in identifying trends and areas of improvement on a broader scale. For instance, if a significant number of students express challenges in understanding a particular aspect of the curriculum, advisors can proactively address this issue in future advising sessions or propose curriculum adjustments [5].

In conclusion, academic advising within the CSIC at King Saud University is a dynamic and structured process, with a clear timeline and specific responsibilities at each stage. This approach ensures that students receive comprehensive support throughout their academic journey, promoting success in both their studies and future careers.

2.5 Chatbots

A chatbot is a computer program that simulates human-to-human dialogue using natural language processing techniques. The application of Artificial Intelligence (AI) in chatbots allows these systems to perceive, interpret, and respond to user inputs in a manner that closely resembles human communication. The incorporation of artificial intelligence (AI) in chatbots is used to improve user experiences, expedite communication processes, and give effective, automated solutions to various enquiries and tasks, ultimately contributing to enhanced user engagement and operational efficiency.

Chatbots can be classified using different methods. The first method classifies chatbots based on the knowledge domain or the amount of data trained [6]. We have two categories: closed domain and open domain. The closed domain focuses on answering specific questions on one domain, while the open domain chatbot answers questions in general not concerned with a specific domain like ChatGPT you can ask it about almost everything.

The second method to classify chatbots is based on the service provided [6] and classified into three categories: Interpersonal, Intrapersonal, and Inter-agent. Interpersonal bots are for allowing services such as table booking in restaurants and answering frequently asked questions. Intrapersonal bots exist in the user domain and do tasks assigned to it, like managing the calendar. Intr-agent requires intercommunication between the user, Alexa-Cortana is an example of an Inter-agent bot.

The third method to classify chatbots is based on the response generation method [6]. We have three categories: rule-based system, intelligence system, and hybrid system. The rule-based bots interact with the users based on a conversation flowchart and defined outline trees. On the other hand, intelligence bots use natural language understanding with systems in a narrow domain and sufficient data exists. Hybrid bots are a combination of rules and machine learning.

The suitable chatbot for Wjjhni is considered a Hybrid system since it uses two algorithm rule-based grammar matching and machine learning (ML) algorithms to understand end-user inputs, match them to intents, and extract structured data which is closed domain. The chatbot learns from training phrases that we provide, and the language models are built into IBM Watson Assistant. Based on this data, it builds a model for making decisions about which intent should be matched to an end-user input machine. It is considered a chatbot in the closed domain which is academic advising for CCIS students and academic advisors.

To implement the chatbot in our system, we collected the data using a survey that was sent to the CCIS students and academic advisors. We have collected data from the university website and the Academic Advising Committee website of the CCIS. Then we identified the intents that represent possible user intentions and then to match the intent corresponding to the user input phrase we trained the chatbot with training phrases and then designed the UI for conversation between the user and chatbot after that, we integrated our chatbot with our application. We used IBM Watson API to send user input and retrieve the chatbot response.

IBM Waston was chosen as a platform to implement the chatbot in our system. In the next section, this platform is explained.

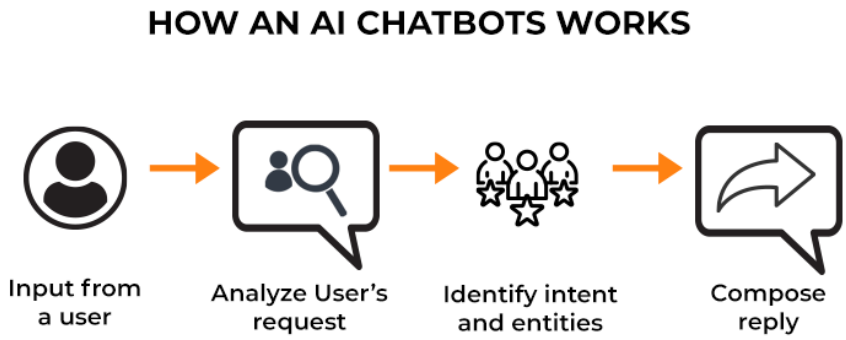


Figure 1: how AI chatbot works

2.6 Chatbot IBM Watson

IBM Watson Assistant is a chatbot platform developed by IBM that leverages natural language processing (NLP) and machine learning capabilities to create conversational interfaces. Here's a general overview of how IBM Watson Assistant works:

1. Intent Recognition:

- Users interact with the chatbot by sending messages or questions.

- Watson Assistant employs intent recognition to understand the purpose or goal behind each user input. It tries to determine what the user is trying to accomplish [7].

2. Entity Recognition:

- In addition to understanding intent, Watson Assistant identifies entities within the user input. Entities are specific pieces of information that are relevant to the user's request.

3. Dialog Flow:

- Watson Assistant uses a dialog flow to manage the conversation. The dialog flow consists of a set of predefined responses and actions based on identified intents and entities.

- It allows the chatbot to respond appropriately to user inputs, guiding the conversation in a meaningful way.

4. Training with Examples:

- Watson Assistant is trained using examples of user inputs and their corresponding intents and entities. This training helps the system improve its understanding of natural language and context over time [8].

5. Machine Learning:

- Watson Assistant leverages machine learning algorithms to continuously improve its performance. As more users interact with the chatbot, the system learns from the data and refines its understanding of language patterns.

6. Integration with External Systems:

- Watson Assistant can be integrated with external systems and databases to retrieve or update information. This allows the chatbot to provide dynamic and personalized responses based on real-time data.

7. User Context:

- The chatbot maintains context during the conversation, allowing it to remember previous interactions and provide more coherent and relevant responses.

8. Deployment:

- Once trained, the Watson Assistant chatbot can be deployed across various platforms, such as websites, mobile apps, or messaging platforms.

9. Analytic Insights:

- IBM Watson Assistant provides analytics and insights into user interactions, helping developers and administrators understand user behavior and improve the chatbot's performance [9].

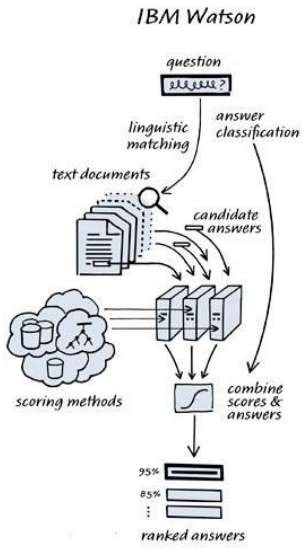


Figure 2: How IBM Watson works

2.7 Why IBM Watson for “Wjjhni”?

The selection of IBM Watson for our academic advising application, "Wjjhni," a deliberate choice driven by several key factors:

1. AI Capabilities: IBM Watson is renowned for its natural language processing and understanding capabilities. This ensures an effective interaction between students and the chatbot, enhancing the overall user experience.

2. Versatility in Deployment: IBM Watson provides flexibility in deployment across various communication channels, including websites and messaging apps. This adaptability aligns with our goal to offer a seamless and accessible user interface that caters to diverse preferences.[8]

3. Development Efficiency: Leveraging IBM Watson accelerates the development process. The platform offers pre-built models and tools, reducing the time and resources required for training the chatbot. This efficiency is crucial for meeting project timelines.

4. Scalability: As "Wjjhni" grows and evolves, scalability is vital. IBM Watson's scalability ensures that our application can handle increasing user interactions and data, accommodating the expanding needs of our user base.[9]

5. Industry Reputation: IBM Watson is a well-established and respected player in the AI industry. Choosing a platform with a solid reputation enhances the credibility and trustworthiness of "Wjjhni," instilling confidence in both users and stakeholders.

6. Integration Capabilities: IBM Watson facilitates seamless integration with external systems. This capability allows us to connect "Wjjhni" with databases and information repositories, ensuring real-time access to accurate and up-to-date information.

7. Community and Support: Being part of the IBM Watson ecosystem provides access to a robust community and support system. This ensures that our development team has the necessary resources and assistance, fostering long-term success and sustainability for "Wjjhni."[7]

In summary, the decision to adopt IBM Watson for "Wjjhni" is grounded in its AI capabilities, deployment versatility, development efficiency, scalability, industry reputation, integration

capabilities, and supportive community. These factors collectively position "Wjjhni" to deliver an innovative and reliable academic advising solution for the students at King Saud University.

3 Literature Review

In order to determine the requirements of Wjjhni, we conducted research on the market to benefit from the strengths and overcome the weaknesses of existing academic advising systems. In the following, we will describe each system and its features that will help us to improve Wjjhni.

1. KSU Students

   The University's Deanship of e-Transaction & Communication and Deanship of Registration offers many valued services for students See Figure 3. The service is offered to fulfill e-services goals. And deliver those services within easy interactive way, and awesome user experience [10].

   Student features:

   - Ability to see the student's academic record and details for each semester.
   - Ability to see the student's rewards.
   - Ability to see the student's loans.
   - Ability to see the student's requests.
   - Ability to see student's schedule.

   Limitations:

   - KSU Student has no features that support academic advising.
   - Support only one type of users which are students.

Figure 3 KSU Student interfaces

2. Nabiha: An Arabic Dialect Chatbot

Al-Ghadhban and Al-Twairesh [11] proposed the "Nabiha" chatbot which is a social chatbot that supports conversations with the students about the information technology (IT) department at King Saud University (KSU). It is available on Android, Twitter, and the Web. The dataset consists of 248 question/answer pairs that were collected from King Saud University IT students' accounts in Askme.com. The chatbot was developed using AMIL (Artificial Intelligence Markup Language) and pattern matching.

3. KSU Help (منصة مرشد)

A student platform that cares about reliable news, guidance, and answering inquiries from students at King Saud University - affiliated with the Student Partnership Program at the Deanship of Student Affairs See Figure 4. It has two channels X (previously Twitter account and Telegram [12].

Figure 4 Murshid account at X (Twitter)

4. My KKU

My KKU application See Figure 5 for all King Khalid University employees including students, faculty members, employees and even visitors can benefit from the application [10].

Student features:

- Viewing academic schedule during the weekday.
- Reviewing absences and the percentage of absences for each course.
- Searching for university employees and faculty members and presenting the means of contacting them.
- Viewing profile.
- The application contains the university community, which is a service that allows you to make friends and communicate instantly from university employees, and it enables you to create groups for communication.
- Calculate GPA to predict next GPA.

- Follow up the news of the university.
- A special section for alerts and notifications.

Limitations:

- There are no features that support academic advising processes.
- It supports three types of users: students, faculty members, and visitors. There are no specific features for the academic advisor.

Figure 5 My KKU Interfaces

5. KFUPM Student

King Fahad University of Petroleum and Minerals has a mobile application called KFUPM Student See Figure 6. Designed to help their students manage their daily academic activities. It is available on Android and IOS systems [13].

KFUPM Student features:

- View the current semester's calendar.
- View transcript and GPA.
- Check class schedule.
- Show academic profile.

- Login to the application.

Limitations:

- KFUPM Student has no communication method with academic advisors. There are only the phone numbers of important contacts at the university.
- Support only one type of users which are students.

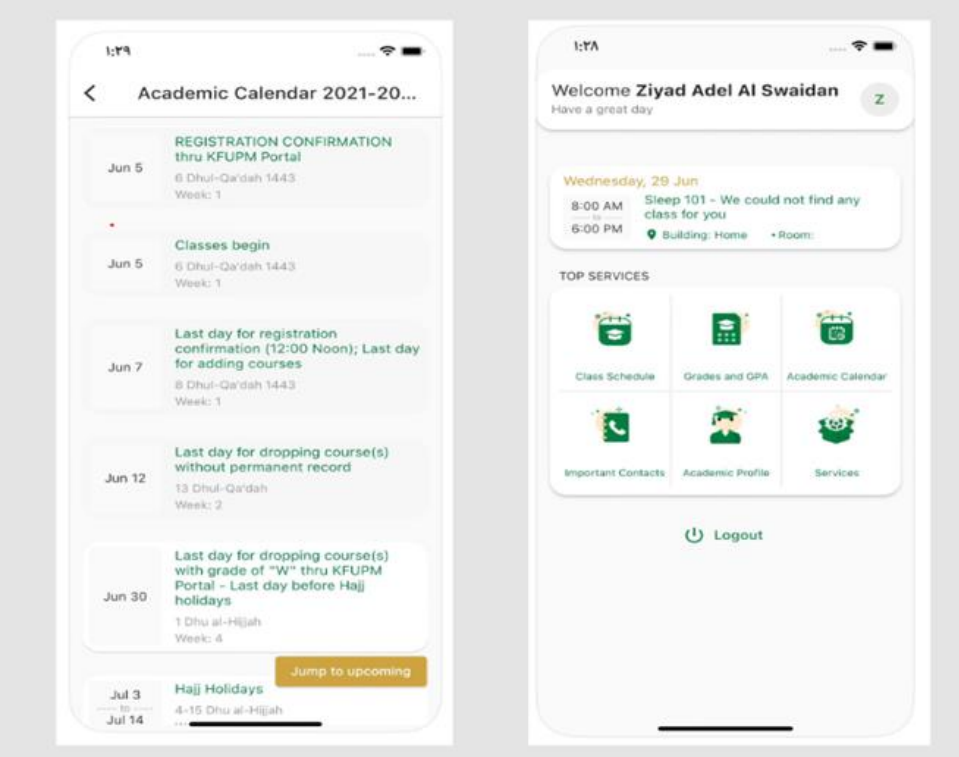


Figure 6 KFUPM Student Interfaces

6. King Abdulaziz University

King Abdulaziz University activates academic advising on their Odus Plus system. See Figure 7. It is a web-based comprehensive academic system for all services related to the educational process for students and faculty members [14].

Academic Advisor features:

- The advisor can communicate with specific students by writing the subject.
- The advisor can select students and make an appointment with them.

- The advisor can rack students' performance by viewing their current semester marks and cumulative grade point average.

Student features:

- The student can reply to the academic advisor's message.
- The student can confirm an appointment.
- The student can schedule appointments.

Limitations:

- Communication between academic advisors and students is like email messages. It is not real-time chatting between them.

Figure 7 King Abdulaziz University Interface

7. Qatar university

Qatar University has an academic advising center See Figure 8 that provides support and academic advising to all students [15].

Student features:

- The student can book an appointment with an academic advisor.
- The student can view/cancel booked appointments.

- The student can view the academic advising session notes.
- The student can view frequently asked questions.
- The student can contact the academic advising center through an online form.
- The student can get electronic forms.
- The student can view required documents uploaded by the advisor.

Academic advisor features:

- The advisor can write academic session notes about the student.
- The advisor can view/cancel booked appointments.
- After an academic session the advisor can upload required documents for the student.

Limitations:

- Booking an appointment feature supports English language only.
- The frequently asked questions are not interactive, just some questions written with their answers.

Figure 8 Qatar University Interface

8. UIS Mobile

The University of Illinois Springfield is a public university in Springfield, Illinois and established in 1969. The University has published a mobile application See Figure 9 that is designed to elevate their students' daily lives at UIS. The UIS Mobile is available on Android and IOS systems. It offers a comprehensive range of features to support academic success [16].

Student Features:

- The student can identify their academic advisor.
- The student schedule appointments with the advisor
- The student can create a tentative drag-and-drop course schedule for an upcoming semester.
- View quick links to campus essentials.
- View events, news, announcements.
- Access academic information such as grades and course schedules.

Limitations:

- It supports three types of users: students, faculty members, and visitors. There are no specific features for the academic advisor.

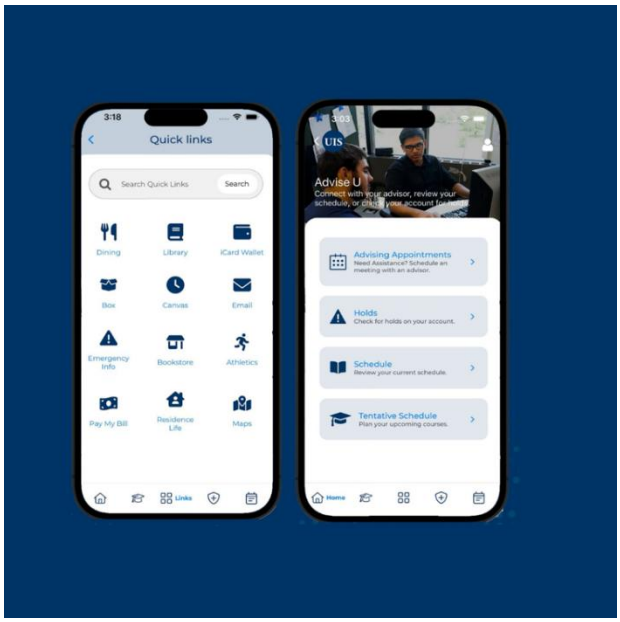


Figure 9 UIS mobile Interfaces.

	Academic Advising systems							
Features	**KSU Student**	**Nabiha**	**My KKU**	**KFUPM**	**King Abdulaziz University**	**Qatar University**	**UIS Mobile**	**Wjjhni**
Type of the system	Mobile application	Mobile application	Mobile application	Mobile application	Website	Website	Mobile application	Mobile application
OS support	Android and IOS	Android	Android and IOS	Android and IOS	Support all OS	Support all OS	Android and IOS.	Android
Book advising appointments					✓	✓	✓	✓
Advisor communication with the students					✓			✓

Academic semester calendar	✓		✓	✓	✓			✓
View the academic advising session notes								✓
AI chatbot to answer students' questions		✓	✓				✓	✓
Forms								✓

Table 1 Comparison table between academic advising systems

After analysis of seven existing systems applications and websites that are used in the academic advising field. There are similarities in the features of the systems, like allowing students to book an appointment and providing communication methods between students and academic advisors. We have found many helpful features that have been implemented in our system.

Our system is composed of a mobile application and a website (admin panel). The admin panel offers several features for the admin such as logging into the dashboard, assigning students to their academic advisors, adding the important dates of the academic year, and removing/adding students and academic advisors.

The mobile application can be divided into two parts the first part concerns students' features such as log-in, asking an AI chatbot, chatting with their advisor, booking an advising appointment, canceling an advising appointment, evaluating advisor, viewing study plans, viewing forms, view important dates of the academic year, use an absence calculator, and update profile. The second part is concerned with the academic advisor features such as logging in, updating profiles, chatting with students, recording student visits, setting availability hours, and canceling a booked appointment.

4 System Design and Development

4.1 Methodology

Wjjhni was developed using the agile software development process. Agile development focuses on quickly creating working software, frequently collaborating with customers, and easily adapting to changes[17]. The development process is divided into small, iterative phases known as sprints, and Wjjhni was developed within five sprints.

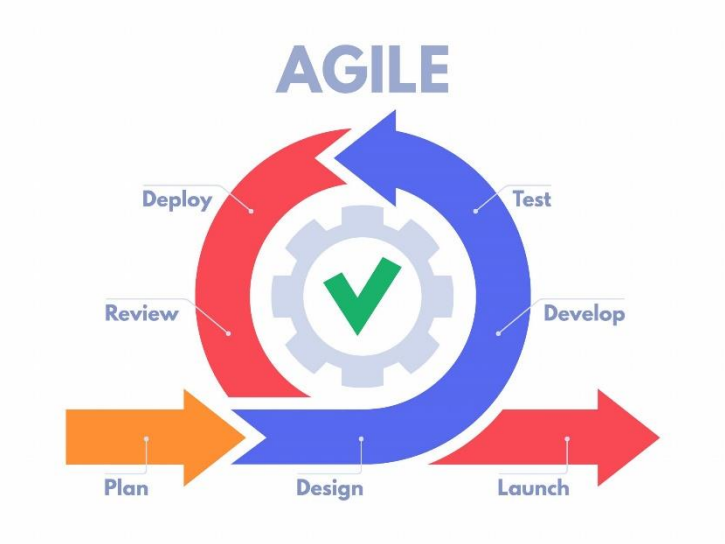


Figure 10 Agile approach [18]

Scrum is a lightweight framework See Figure 11 that helps people, teams and organizations generate value through adaptive solutions for complex problems.

Scrum has three roles: the product owner and scrum master roles are for Dr. Hend Albassam. The development team consists of Ibtihal Almutairi, Lamia Alhelali, Shaden Alshabanat, Mona Alafari, and Esraa Alhasan.

The Events are used in Scrum to create regularity and to minimize the need for meetings and designed to enable the transparency required. Scrum has Five events [19]:

1. The Sprint: They are fixed length events of one month or less to create consistency. And all other events happen in sprint.

2. Sprint Planning: initiates the Sprint and a plan is created by the collaborative work of the entire Scrum Team.
3. Daily Scrum: is a 15-minute event for the Developers of the Scrum Team to inspect progress toward the Sprint Goal and adapt the Sprint Backlog.
4. Sprint Review: to inspect the outcome of the Sprint and determine future adaptations.
5. Sprint Retrospective: to plan ways to increase quality and effectiveness.

Scrum's artifacts are tangible representations of work or value within a project. They serve to increase transparency by providing clear and consistent information to all team members[20].

1. Product backlog: A product backlog is a list of new features, additions, bug repairs, task represented as user stories.
2. The sprint backlog is another list that contains every task that the scrum team has to achieve in every sprint.
3. A product increment is the set of customer deliverables created during a sprint by fulfilling product backlog items.

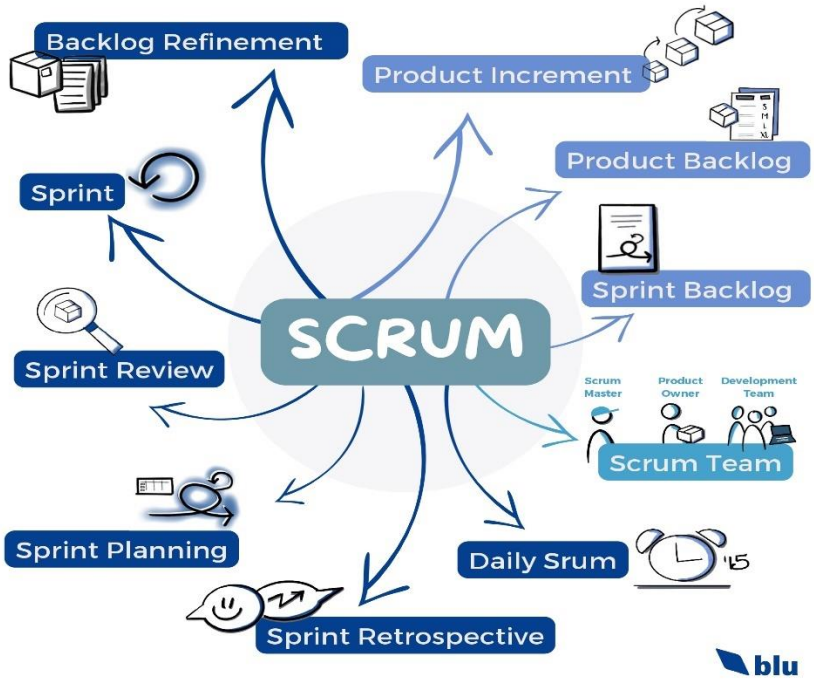


Figure 11 Agile scrum methodology[21]

Throughout the sprint, the team meets for concise weekly stand-up meetings to assess progress, address any difficulties, and determine if any alterations to the plan are required. At the sprint's conclusion, the team did a sprint review meeting to showcase the completed product and gather feedback.

In addition, the team conducts a retrospective meeting to evaluate the sprint and identify enhancement potentials. The Scrum process places a significant emphasis on collaboration and transparent communication. The team frequently collaborates to accomplish tasks and provides mutual feedback, increasing the likelihood of maintaining focus and delivering high-quality work within the sprint.

In conclusion, the Scrum framework and Agile principles foster a flexible, collaborative, and iterative approach to software development. This approach empowered our team to quickly and effectively adapt to evolving requirements.

Jira[1] allowed us to apply Agile methodology as it provided us with several features. We used it to plan our sprints and track progress. It allowed our team to create and prioritize their product backlog. Jira was a handy tool to track the progress of our tasks. We can update the status of each task (e.g., to-do, in progress, done) and log time spent on them. Lastly, we used it to record all meetings that have been held with our supervisor.

Using GitHub[2] in Wjjhni was necessary to work as a team and enhance collaboration. We used it to create, store, change, merge, and collaborate on code. Any member of the Wjjhni development team can access the GitHub repository and see the most recent version in real-time. Managing this manually would be extremely difficult and not good practice.

[1] https://2023-1st-gp11.atlassian.net/jira/software/projects/GP/boards/1/backlog

[2] https://github.com/mona-ahmed2/2023-GP1-11

4.2 System Requirements

4.2.1 System Users

In the system, there are three main user roles: Academic advisors, students, and administrators See Table 2. Academic advisors are responsible for guiding a group of students through any academic challenges they may encounter. They have the capability to add their available hours so that students can schedule appointments with them. Students, on the other hand, are individuals enrolled in the college seeking academic guidance, appointment scheduling with their advisors, and access to forms and college plans. Administrators oversee the database and its operations, including setting up calendars, managing forms and plans, and assigning students to their respective advisors.

	Age	Educational level	Experience	Technical expertise	Platform
Academic advisor	Above 18	Bachelor's degree or equivalent	Should have experience with similar systems	Medium	Mobile application
Student		High school		Medium	Mobile application
Admin		Bachelor's degree or equivalent		High	Website

Table 2 System users characteristics

4.2.2 Requirements Elicitation and Analysis

We gathered as much information as possible from different sources. We used questionnaires and interviews, with students and academic advisors in Kind Saud university, that information elicitation helped us to decide which features are most important and which are less. To make sure we implement the most important ones first.

We received 38 responses from the questionnaire (Appendix A). We did it online using Google Forms. We've sent it to our CCIS students' group through 'WhatsApp'. As for the advisors, we have sent a couple of them through email.

The questionnaire contained multiple different questions. We got 92.1% of the students' responses and the rest were academic advisors. We asked them how long they have been at the college, 52.6% have been in the college for 2-3 years. We asked them if they had used any academic advising app before and 97.4% hadn't used any similar app. We also asked about the preferred language of the app and the chatbot and 65.8% chose Arabic.

We asked about "Wjjhni" features that we planned, and they agreed that it is needed in the app. One of the questions was about the most important features in their opinion and the most chosen ones were the absence hours calculator, chatbot, and calendar. Finally, we asked for their suggestions, and one of the suggestions was to provide the student with the ability to request an Advisor change. Which is a good suggestion.

With regard to the interviews, the total number of interviews is 5 (Appendix B). Two of them were with academic advising expertise. and the other three were with CCIS students. We asked them different questions, we asked them about the academic problems they face, any suggested features they would like to add to "Wjjhni", and if they used academic advising system before or not.

The first one was with Dr. Ameera Almasoud through Zoom. and She suggested setting a reminder for the advisor if she didn't answer the student within a period, to remind her to answer. The second one was with Mrs. Mashael Zamil, and it was in person. She suggested providing the advisor with the ability to add a state for each student that only can be seen by her, to help her give attention to the seniors or others.

The other three were CCIS students. The first one was with Reem Alnasser, she suggested enabling the student to change the advisor, and that suggestion was already suggested at the questionnaires. The second one was with Aldana Albeshr, and one of her suggestions was already planned for and it was a calendar of important dates. The last one was with Alin Altowim. She has suggested many features and the one we liked the most was to track degree progress.

From the information we gathered, we know that the users need almost all of the features we listed. Also, they suggested some features that we didn't think of, like providing the student with the ability to request an advisor change. Another feature that one of the interviewees suggested was to provide the advisor with the ability to set a student state to help her focus more on the students who need more attention.

4.2.3 User Interactions

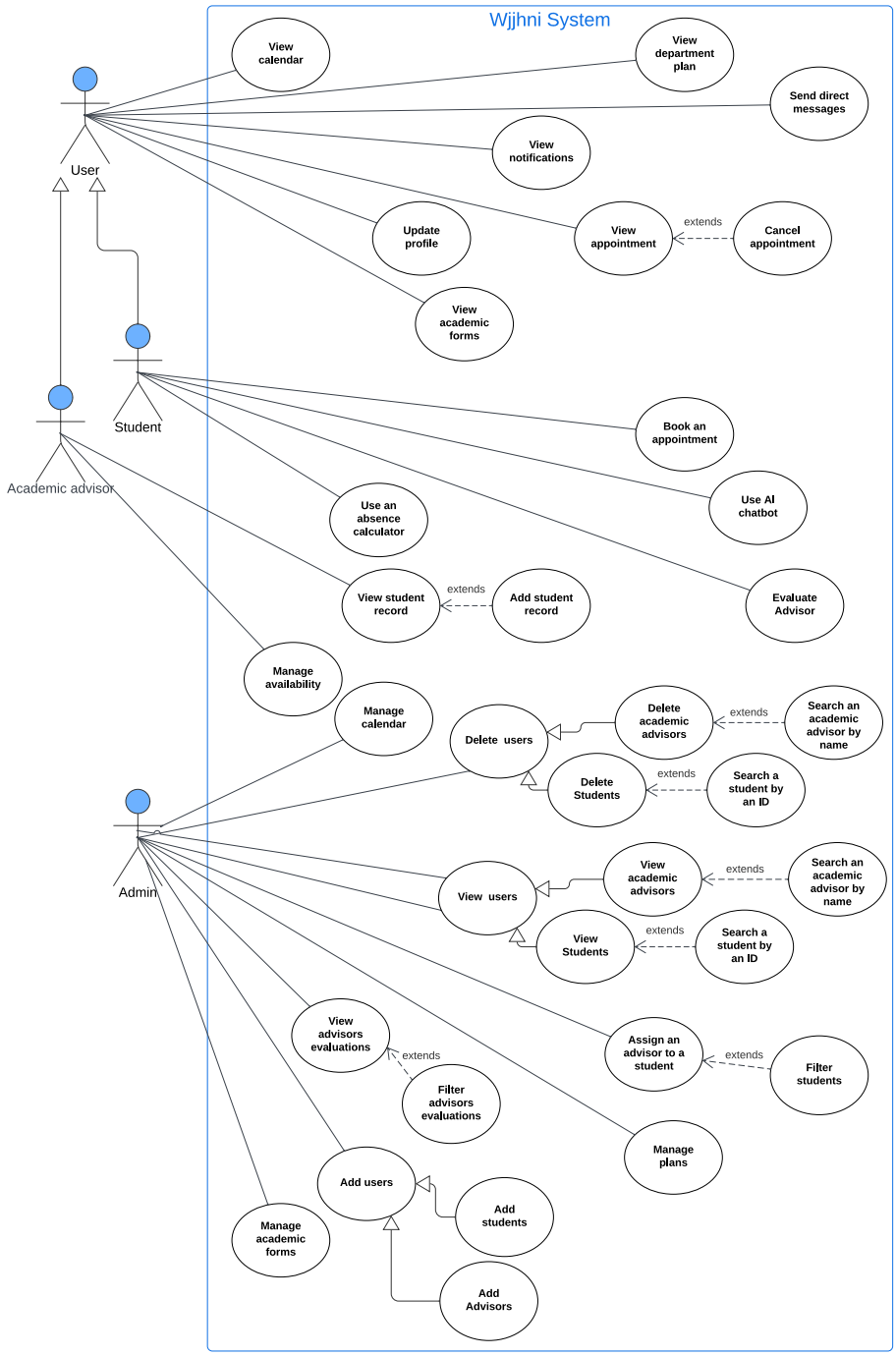


Figure 12 Wjjhni Use Case diagram

4.2.4 Roadmap and Product Backlog

Our product roadmap consists of five sprints where each sprint we have implemented several features. See Figure 13 summarizes the product roadmap. Table 2 shows these features as user stories in the product backlog.

Sprint-0 was to get familiar with environmental learning tools and languages that will be used and data collection. Sprint-1 was the development of an AI chatbot.

Sprint-2 has many features which are adding students and advisors by admin, logging in/out, updating profiles, and allowing the advisor to add availability hours.

Sprint-3 has many features which are booking/canceling appointments, assigning students to advisors, online chatting, recording students' visits by advisors, and managing academic forms.

Sprint-4 has many features which are rating advisors by the student, managing study plans, deleting students and advisors, and adding important dates to the calendar.

The last sprint, sprint-5 has many features such as an absence hours calculator, viewing the academic calendar, and some enhancements to features that were implemented in previous sprints like adding a notifications center when a plan has been updated or academic form booking an appointment, or a message is received.

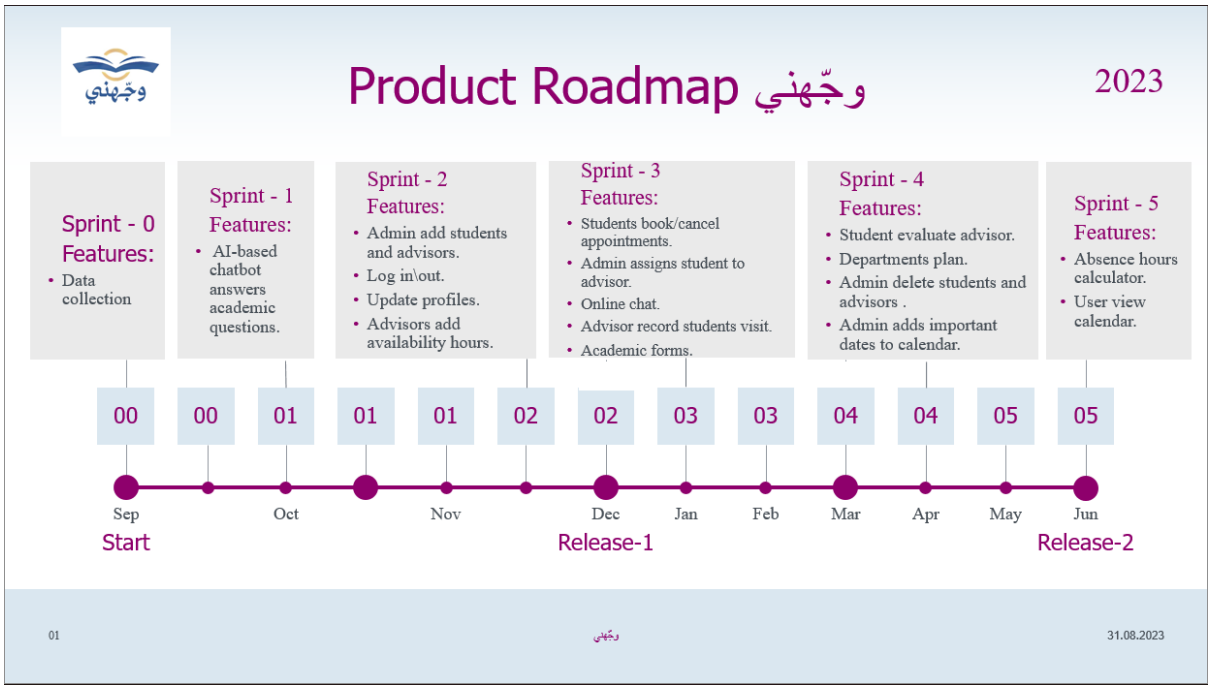


Figure 13 Wjjhni Roadmap

ID	PBIs (User Stories)	Size	Type (Feature, defect, technical work, knowledge acquisition)	Status (To do, in progress, or Done)	Acceptance Criteria The conditions of satisfaction that must be met for that item to be accepted.
1	As a student, I want to ask AI chatbot about academic advising so that I can get the information easily and correctly.	13	Feature	Done	As a student, if I asked the chatbot an academic question then it should answer me. As a student, if I didn't write anything then it should not respond.

					As a student, if I write a multiline message then it should be displayed appropriately.
2	As an admin, I want to add students to the database so that they can log-in to the application.	6	Feature	Done	As an admin, If I go to the add student page, I upload a file that contains the student's information, then a confirmation message will appear that there are new students has been successfully added, and it will be sent to each student through email the username and password. As an admin, If I go to add a student page, I upload a file that contains the student's information, and all students were already added, then an error message will appear saying that all students were already added.
3	As an admin, I want to add academic advisors to the	5	Feature	Done	As an admin, If I go to the add academic advisor

	database so that they can log-in to the application.				page, I upload a file that contains the advisor's information's, then a confirmation message will appear that there are new academic advisors has been successfully added, and it will be sent to each academic advisor through email the username and password. As an admin, If I go to add an academic advisor page, I upload a file that contains the advisor's information, and if all advisors were already added, then an error message will appear saying that all advisors were already added.
4	As an admin, I want to view the students in the database so that I can manage them effectively.	3	Feature	Done	As an admin, If I go to view students' page then I should be able to view their information which includes name, ID, email, phone and major.
5	As an admin, I want to view the academic advisors in the	3	Feature	Done	As an admin, If I go to view

	database so that I can manage them effectively.				academic advisors' page then I should be able to view their information which includes name, email, phone, and department.
6	As a registered user, I want to login so that I can use application features.	2	Feature	Done	As a registered user, if I wanted to use the application features, then I should login using the correct university email and password. As a registered user, if I wanted to use the application features, then if I tried to login with incorrect email and/or password an error message should appear.
7	As an academic advisor, I want to be able to add my availability, so that my students can book an appointment with me.	7	Feature	Done	As an academic advisor if I go to add my availability interface then I select the suitable hours and dates from the Calendar and click the add button then a red color should appear to make it clear that hour was chosen.

					As an academic advisor, if I go to add my availability interface then I try to select the hours and dates from the Calendar that are before today's date, then it will not let me choose it.
8	As a student, I want to update my profile so that I can keep it updated.	4	Feature	Done	As a student, if I go to the account page and click on the update my information button and change phone then my information should be updated.
9	As an academic advisor, I want to update my profile so that I can keep it updated.	4	Feature	Done	As an academic advisor, if I go to the account page and click on the update my information button and change my phone then my information should be updated.
10	As a logged in user, I want to log out of the application so that I can log in with another account.	1	Feature	Done	As a logged-in user, if I click on the logout button, then I will be directed to the login page.

11	As an admin, I want to log in to admin panel so that I can manage students and academic advisors.	2	Feature	Done	As an admin, if I wanted to use the admin panel features, then I should login using the correct email and password. As an admin, If I wanted to use the admin panel features, then If I tried to login with incorrect email and/or password an error message should appear.
12	As an admin, I want to log out of the dashboard so that I cannot access the dashboard.	1	Feature	Done	As a logged-in admin, if I click on the logout button, then I will be directed to the login page.
13	As an admin, I want to reset my password, so that I can log in to the admin panel again.	2	Feature	Done	As an admin, if go to login page click on the forgot password then I enter my email then I should receive a confirmation showing that the reset link was sent via your email. As an admin, if go to login page click on the forgot password then I have entered email

					does not exist then an error message showing that email not found.
14	As a user, I want to reset my password, so that I can log in to the application again.	2	Feature	Done	As a user, if I want to log in to the application again but I forget the password, then I should be able to reset the password.
15	As an admin, I want to assign academic advisors to students so that no student becomes without an advisor.	5	Feature	Done	As an admin, if I wanted to assign academic advisors to students, then I would be able to find their names to assign them to each other with no mistake.
16	As a student, I want to book an appointment with my academic advisor so that I can discuss my academic situation.	5	Feature	Done	As a student, if I go to the book advising appointment page and select my advisor's name then the available dates should appear, and I click to one of them and click the button book then a confirmation message should appear.
17	As a user, I want to cancel an appointment so that I can reschedule it.	3	Feature	Done	As a user, if I go to my appointments

					interface and select an appointment, click on cancel button then a confirmation message should appear, and the appointment will be removed from the interface.
18	As an advisor, I want to chat in real-time with my students, so that I can communicate with them without delays.	8	Feature	Done	As an advisor, if I wanted to communicate with my student, then I should be able to chat online with her without delays.
19	As a student, I want to chat in real-time with my advisor, so that I can communicate with her without delays.	8	Feature	Done	As a student, if I wanted to communicate with my advisor, then I should be able to chat online with her without delay.
20	As an academic advisor, I want to record the students' visit so that I can check it anytime.	6	Feature	Done	As an academic advisor, if I recorded my students' visit, then I should be able to check it anytime. As an academic advisor, If I didn't record my students' visit then I shouldn't be able to check it.

21	As an admin, I want to edit the academic forms, so that it is up to date.	4	Feature	Done	As an admin, if I wanted the academic forms to be up to date, then I should be able to edit them.
22	As a user, I want to find academic forms so that I can use them.	3	Feature	Done	As a user, if I wanted to use the academic forms, then I should be able to find them easily.
23	As a student, I want to evaluate the academic advisor so that the administration can know if the advisor was efficient or not.	6	Feature	Done	As a student, if I wanted to evaluate an advisor, then I would be able to do that quickly.
24	As an admin, I want to edit the department's plan, so it is up to date.	4	Feature	Done	As an admin, if I wanted the department's plan to be up to date, then I should be able to edit them.
25	As a user, I want to view all the department plans for each major so that I will know about all the major plans and courses.	3	Feature	Done	As a user, if I wanted to view the plans, then I would be able to view it in an easy representation.
26	As an admin, I want to remove students and academic advisors from the database so that I can manage it effectively.	4	Feature	Done	As an admin, If I go to users' page click remove button then a confirmation will appear to confirm the deletion.
27	As an admin, I want to add important dates to the academic calendar so that all users can view it updated on Wjjhni mobile application.	5	Feature	Done	As an admin, If I go to the add important dates page, I fill the form with the date and the event's name, and then a confirmation

					message will appear.
28	As a user, I want to view important dates on the academic calendar so that I will not miss important dates.	4	Feature	Done	As a user, if I wanted to view important dates on the academic calendar, then I should be able to view the calendar correctly.
29	As a student, I want to calculate my absence hours allowed for a course so that I will not exceed the maximum absence limit.	4	Feature	Done	As a student, If I wanted to calculate the absence hours allowed for a course, then I should be able to calculate it without any mistakes.
30	As a user, I want the Wjjhni application to load within 2 seconds of lunch so that I will not have to wait long for the application to load.	2	Feature	Done	As a user, if I open the Wjjhni application, then it will load and lunch within 2 seconds
31	As an admin, I want the admin dashboard pages to load in less than 3 seconds so that I do not get frustrated by slow performance.	2	Feature	Done	As an admin, if I use the dashboard, then it would be available 99% of the time I try to access it.
32	As a user, I want the system to be available 99% of the time I try to access it, so that I don't get frustrated.	2	Feature	Done	As a user, if I wanted to use the system, then it should be available 99% of the time I try to access it.
33	As a user, I want to have a clear and simple interface, so that I can learn how to use the application in less than 15 minutes.	2	Feature	Done	As a user, if I open the application for the first time, then I should learn how to use

					it in less than 15 minutes.
34	As a student, I want the AI chatbot to answer within 10 seconds so that I won't have to wait a long time.	2	Feature	Done	As a student, If I asked the chatbot, then it must answer within 10 seconds.

Table 3 Wjjhni Prodcut backlog

4.2.5 Architectural Diagram

We have found that the Client-Server architecture style is the most appropriate for Wjjhni. The ability to distribute servers throughout a network is the primary benefit of client-server architecture. The server will be responsible for processing client's (Student, Academic Advisor, Admin) requests, connecting with the database to store or recall data, and returning the result via the proper protocol. Using the client-server architecture will enable centralized management because all information will be in one place and under the administrator's full authority, making it easier to edit data. Access to the file's records will also be straightforward because they are all kept on a single server. Furthermore, there are several non-functional system criteria that must be addressed. In terms of performance and availability, client-server is the better option for our project. Figure 14 shows the client-server architecture pattern that has been chosen for Wjjhni.

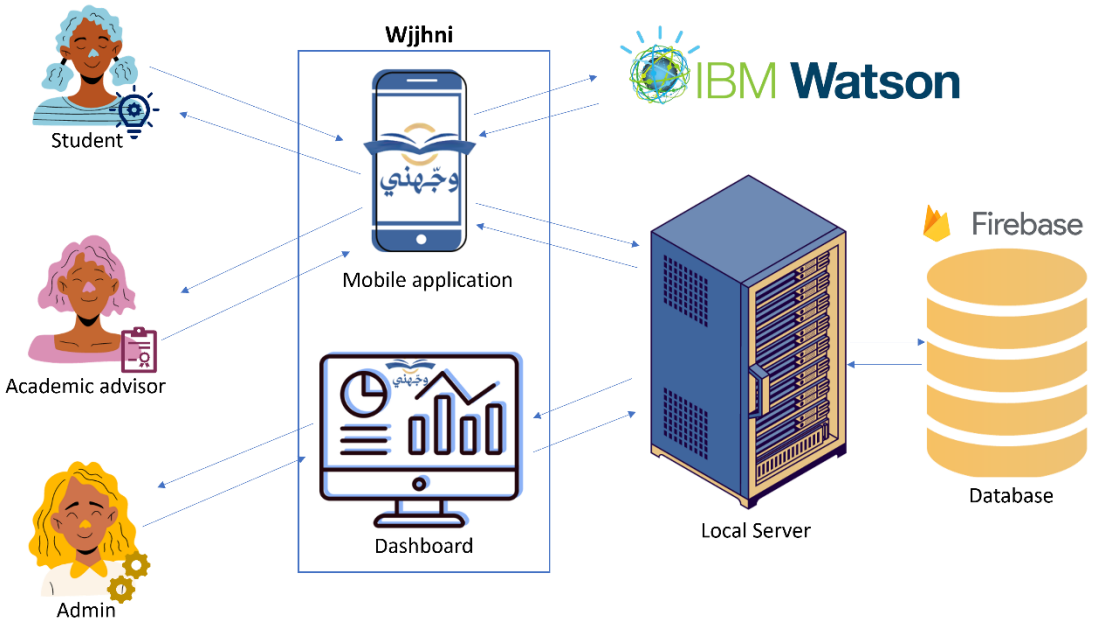


Figure 14 Client-server architecture pattern.

4.2.6 Class Diagram

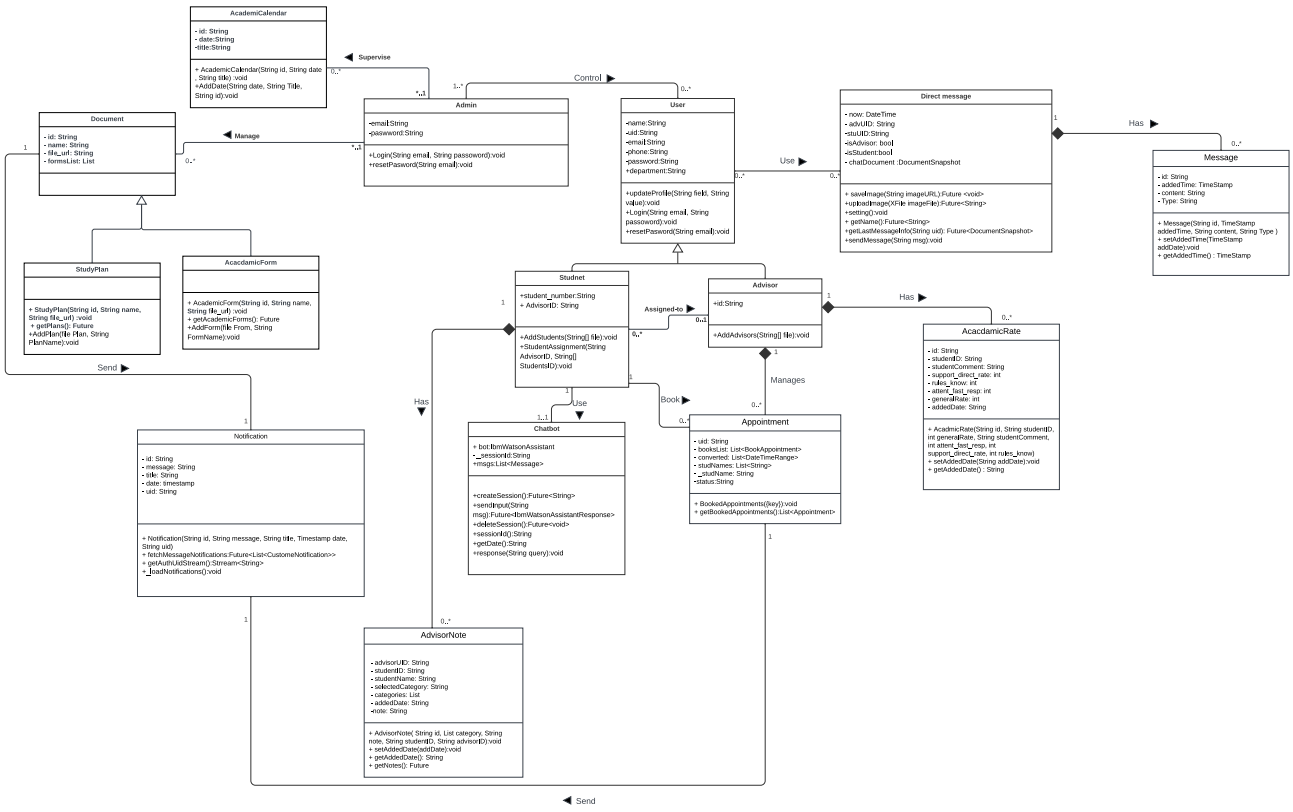


Figure 15 Wjjhni Class Diagram

4.2.7 Component Level Design

- Adding students

As an admin, I want to add students and academic advisors to the database so that they can log in to the application. ID (3). See Figure 16.

We see here the flowchart of adding students using csv file. First, when the file is uploaded successfully a message is shown that it is successfully uploaded. Then, when “add” button is clicked it checks if the file format is correct or not, if not it shows a message, is yes it starts to read the file, if there was an empty id and a student that is already added to the database it skips that student and goes to the next one. At the end it shows the number of students that are added successfully. If all students in the file are already in the database, it will show a message.

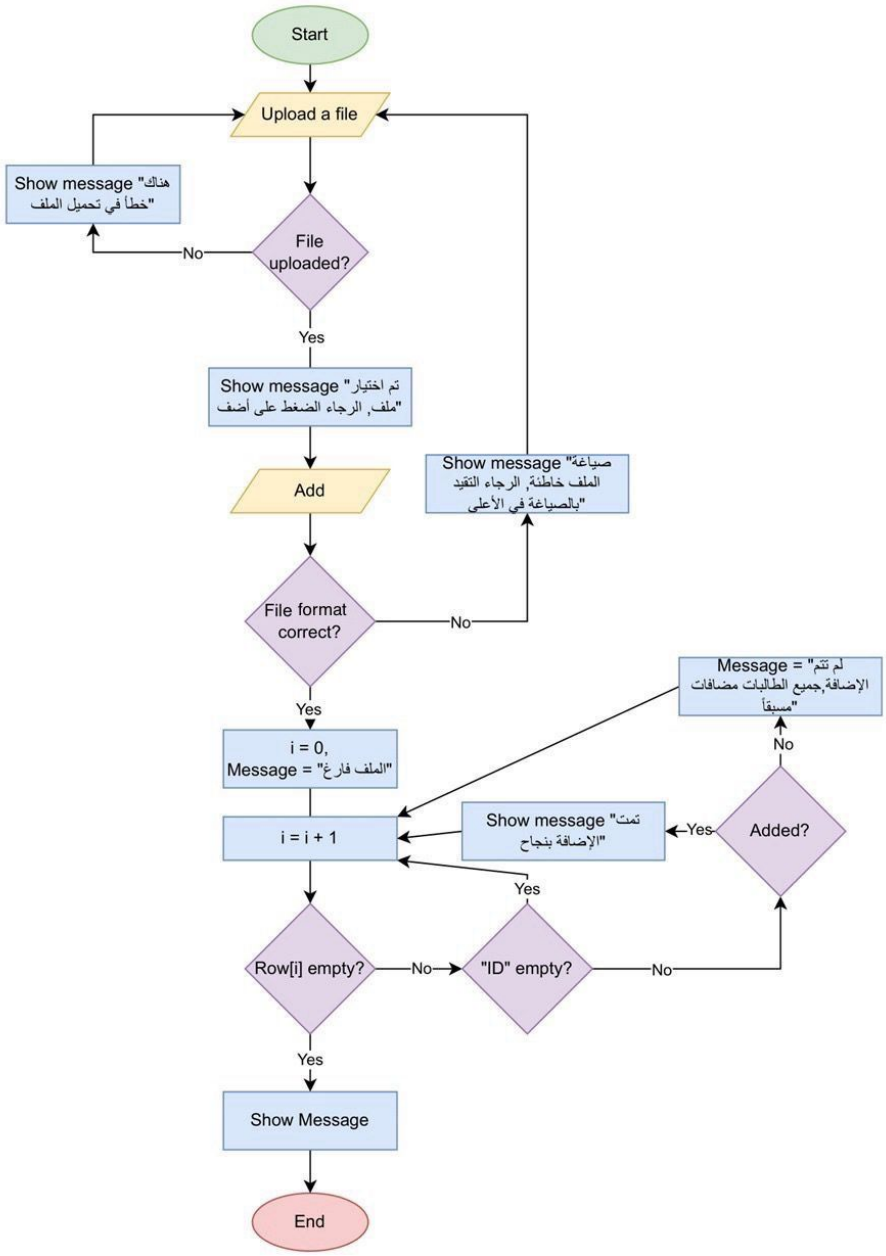


Figure 16: Adding Students flowchart

- Students Assignment

As an admin, I want to assign academic advisors to students so that no student becomes without an advisor. ID (15)

BEGIN

selectedStudents = GetSelectedStudentsFromCheckboxList()

selectedAdvisor = GetSelectedAdvisorFromDropDownList()

```
    FOR EACH student IN selectedStudents
        student.AdvisorUID = selectedAdvisor.UID
    END FOR
END
```

Asking the chatbot

As a student, I want to ask AI chatbot about academic advising so that I can get the information easily and correctly. ID (1)

```
BEGIN
    SHOW chatbot Screen
    List<Message> msgs
    welcoming_msg
    ADD welcoming_msg in msgs
    chatbotService chatbot= chatbotService()
    READ user message
    IF user message=
THEN
 PRINT
ELSE
ADD user message in msgs
 chatbot.createSession()
response=chatbot.sendInput
 ADD response in msgs
        SHOW response
ENDIF
```

```
FUNCTION createSession()
  bot = IbmWatsonAssistant(auth);
TRY
    sessionId = Await bot.createSession()
   Catch (Exception e)
    PRINT 'session error: ' + e.toString()
RETURN e.toString()
PRINT 'created session: ' + sessionId
RETURN  _sessionId
END

FUNCTION sendInput(query)
  PRINT 'Sending chatbot input: ' + input
  IF sessionId == null
    Await createSession()
TRY
   Return bot.sendInput(input, sessionId: sessionId)
CATCH (Exception e)
    Print 'Error sending chatbot input: ' + input + '.\n' + e.toString()
END
END
```

- Advisor Rating

As a student, I want to evaluate the academic advisor so that the administration can know if the advisor was efficient or not. See Figure 17.

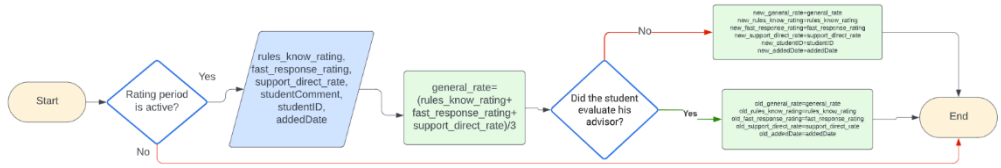


Figure 17: Advisor Rating Flowchart

4.3 Data Design

4.3.1 Data Models

- The ER diagram

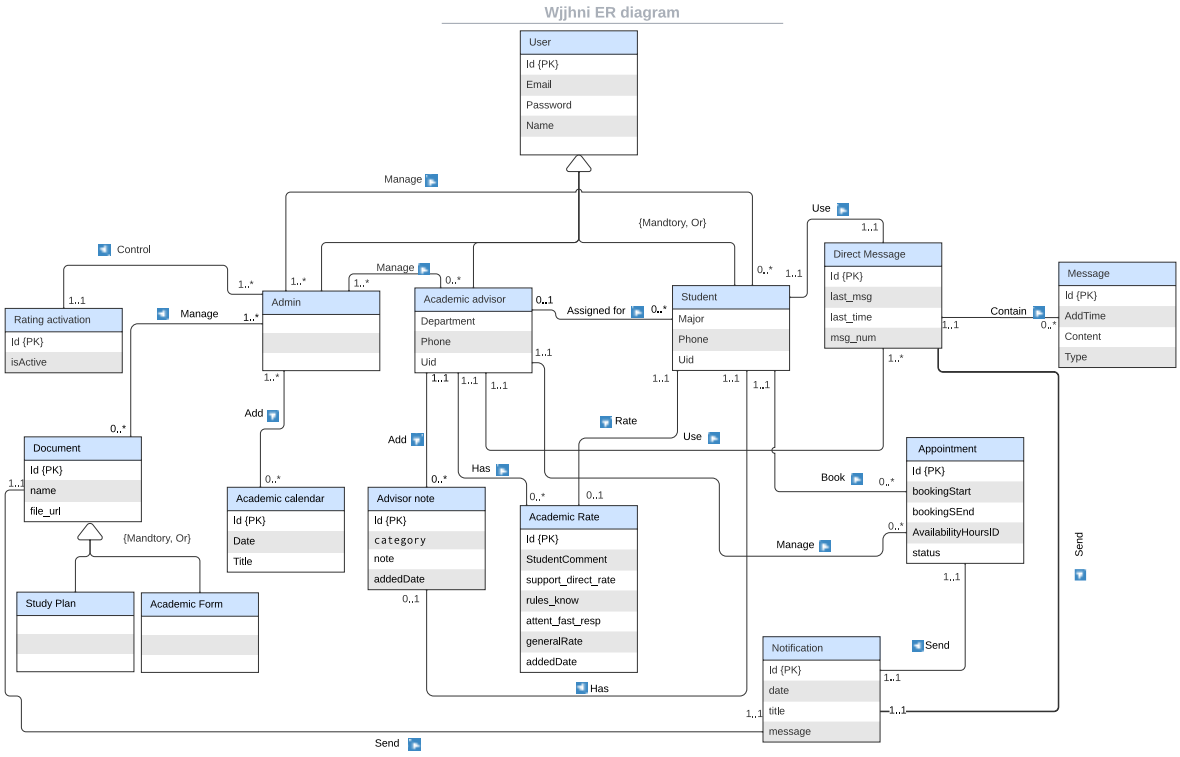


Figure 18 Wjjhni ER diagram

- The non-relational data model

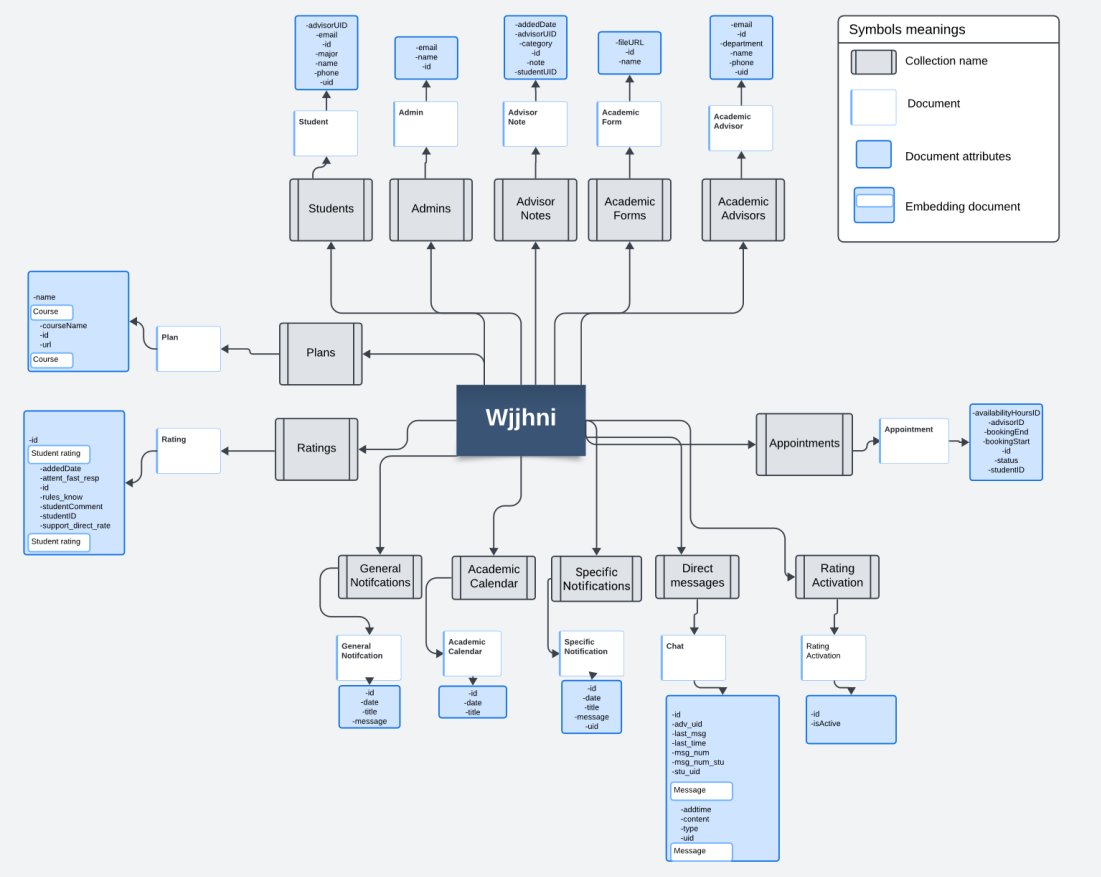


Figure 19 The non-relational data model of Wjjhni

We used Firebase, which is a document-oriented database system. We have the following collections represented in Figure 19, by the main branches of the tree which are Students, Admin, Academic advisors, Ratings, Plans, and Academic forms etc. Each collection has documents, and each document has attributes. In our Firebase database, we have implemented referencing and embedding approaches to efficiently organize and manage collections.

Specifically, we've employed an embedding approach in the Plans, Chats, and Ratings collections. Within the Plans collection, courses are embedded within each plan document. Similarly, in the Chats collection, messages are embedded within each chat document. Additionally, within the Ratings collection, students' ratings are embedded within each advisor document.

For other data collections, we've opted for a referencing approach to establish relationships between documents. For example, to connect advisors with their availability (appointments), we've included references or keys in the "Appointments" documents that correspond to the unique identifier (advisorID) in the Advisors collection.

This combination of embedding and referencing methodologies ensures an efficient and well-organized database structure, facilitating accessibility and scalability while upholding data integrity.

4.3.2 Data Collection and Preparation

In the AI Chatbot feature, we needed to collect data about academic advising in the form of questions and answers. First, we collected data from the Deanship of Admission and Registration Affairs- KSU University [22], the data was about student stipends, continuing students, graduate students, etc. Second, we distributed a questionnaire (Appendix C) with one question to our college students, we asked them to write any question they may ask their advisors about academic advising. The responses were 35, each response with at least one question. Then we answered these questions and arranged them in an Excel sheet with fixing any spelling and grammatical errors.

Since the academic advising of our college and university lack some common questions and concepts related to academic advising, we expanded our collection method to include other universities like King Faisal University – Abqaiq branch[23], King Faisal University – college of Arts[24] and Deanship of Admission and Registration Affairs- Taiba university [25]. We collected the common questions that applied to all Saudi universities.

4.4 Interface Design

- Admin Website:

- Site Map:

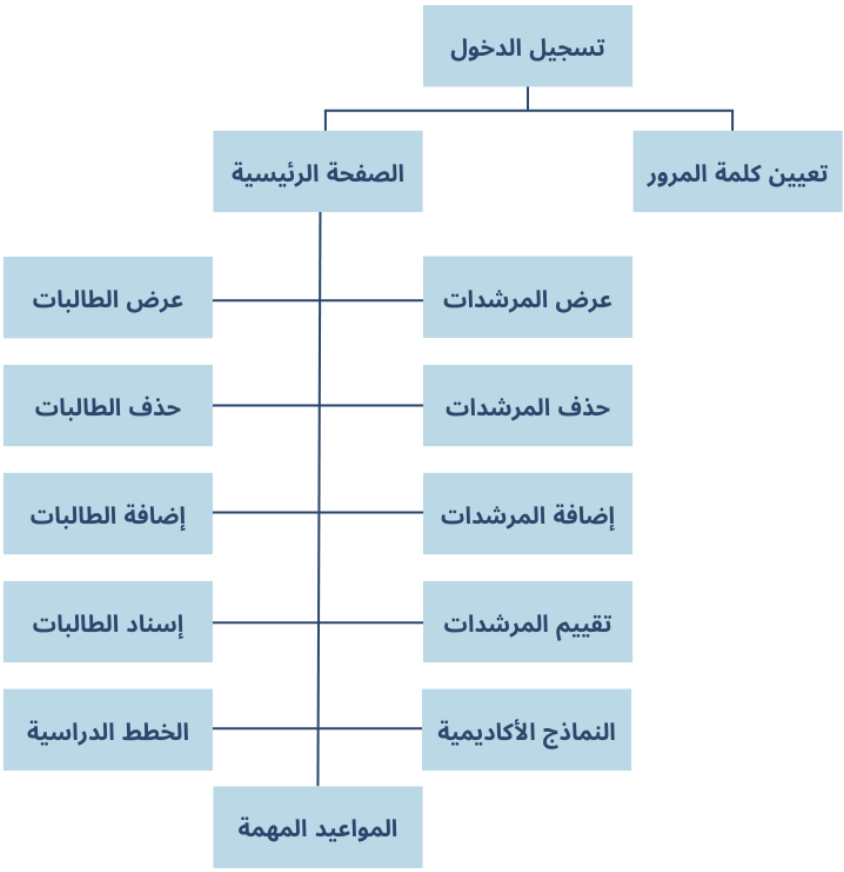


Figure 20: Website Sitemap

- Application:

- Site Map:

Student:

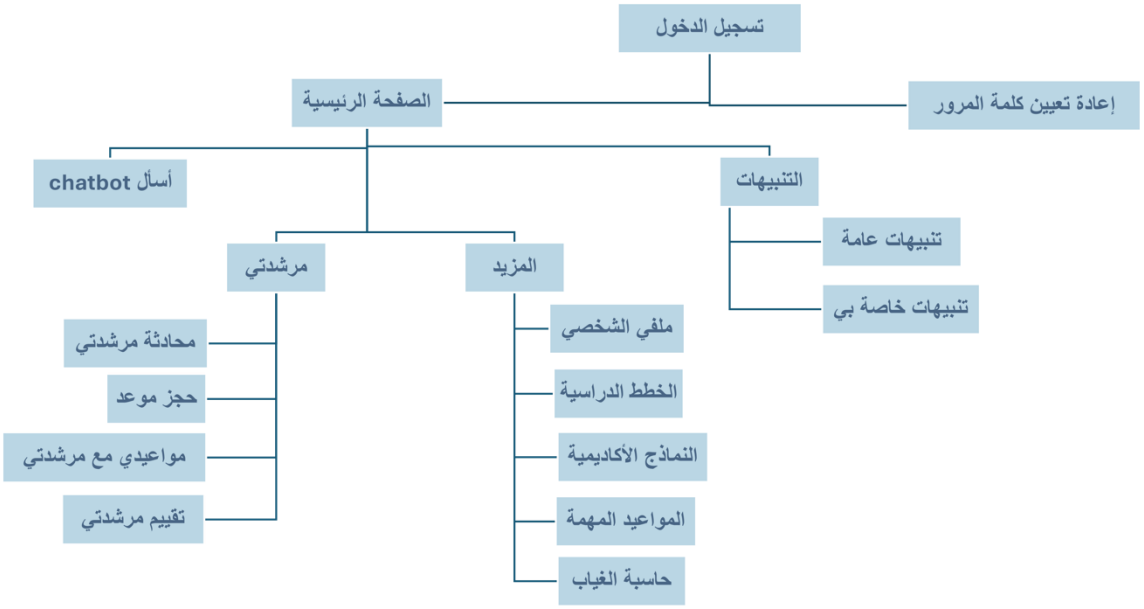


Figure 21: Student Application Sitemap

Academic advisor:

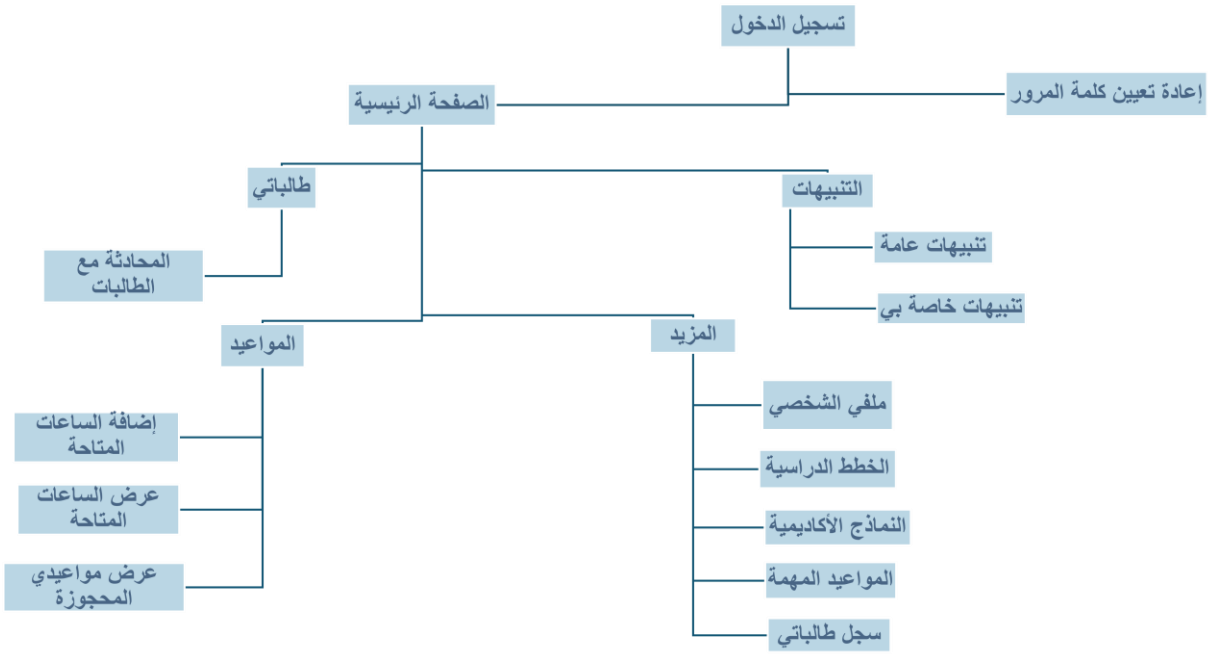


Figure 22: Advisor application sitemap

We applied these rules and guidelines based on the user experience design course:

- We used universally recognized symbols for deletion, See Figures 23 and 27 such as a trash can or trash bin, this helps users quickly identify the purpose of the icon.
- We used a clear and prominent "Login" button See Figures 24 and 28. This makes sure it stands out visually, guiding users to the next step.
- We used colors that are familiar to the average user and attract his attention See Figure 24, such as red color for "unavailable" or incorrect credentials and green color for "available".
- We used the sidebar menu in the admin website See Figure 26, to make the navigation easier..
- We provided visual feedback See Figure 25, when users hover over a menu item indicates that the item is clickable, and that the cursor is on that item.
- We used high contrast colors between background and font color See Figure 26, such as if the background is dark the font is light and vice versa.

Figure 23: Plans Page

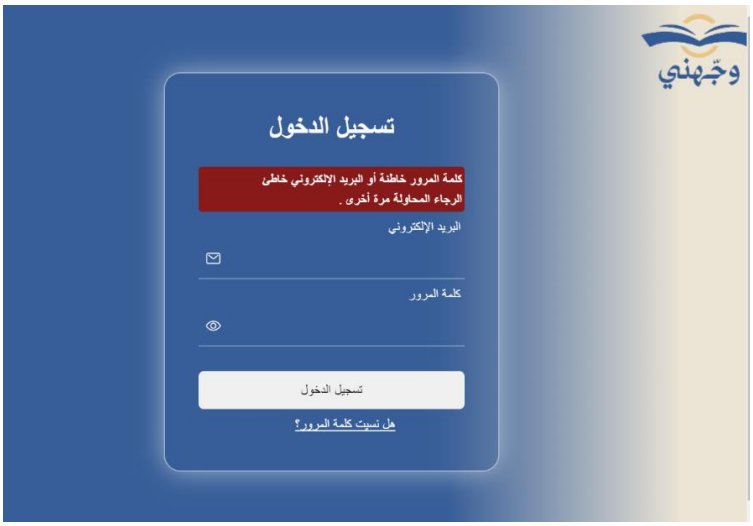


Figure 24: Website Log-in Page

Figure 25: Website Forms Page

Figure 26: Website Main Page

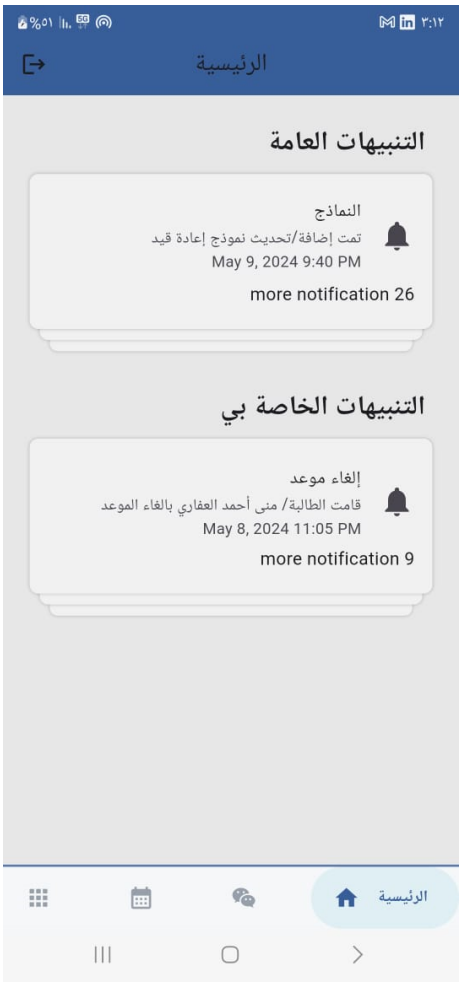


Figure 27: Application Notification Page

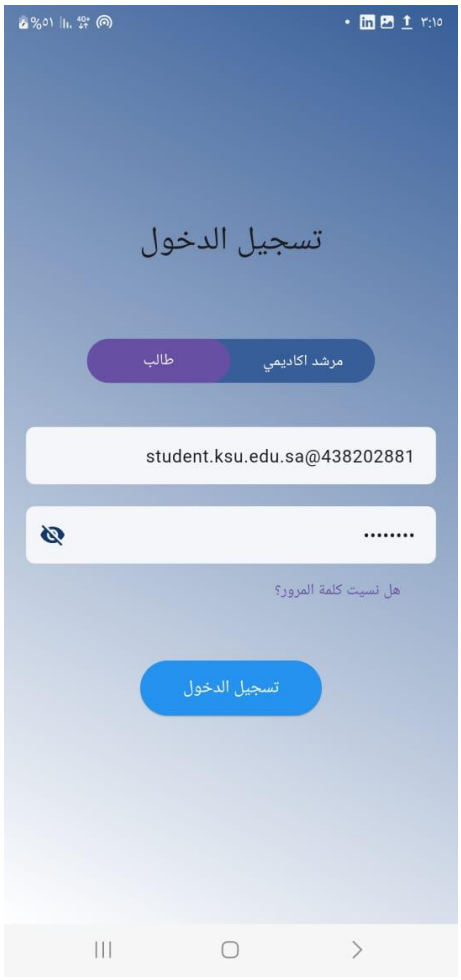


Figure 28: Application Log-in Page

4.5 Implementation

In this part, we have discussed the challenges we faced. We have mentioned the steps we followed in implementing the system and how to configure the software. We have explained in detail the reason for the shift in choosing the chatbot development platform, the challenge we have faced in linking the chatbot API with the application, and some of the problems we faced in building the application and website.

4.5.1 Chatbot Development

- Changing Chatbot Development Platform

The decision to transition from Dialog flow to IBM Watson Assistant in chatbot development can be attributed to the following compelling reasons. When evaluating crucial

aspects such as intent classification, confidence scores, and entity extraction, IBM Watson emerges as the top-performing Natural Language Understanding (NLU) platform with an F1-measure>84% compared to Dialogue Flow F1-measure>78%. In the realm of chatbot development, NLUs play a central role, serving as the foundation for the chatbot's ability to comprehend and respond effectively to user inputs[26].

- IBM Watson Assistant API

After collecting data, we classified each question into an intent and these intents were used to train the NLU of IBM Watson Assistant, see Figure 29. We then designed the chatbot interface and then started integrating our chatbot with the Wjjhni mobile app. We used the ibm_watson_assistant package [27], but it was poorly documented. In the beginning, we created a service class see Figures 30 and 31 that contains functions based on the IBM Watson API such as the createSession() function which creates a session for the user to enable him to send his response to the chatbot. SendInput() function will send the user's question and return the response in a JSON format see Figure 31. While working with the ibm_watson_assistant package [27] we encountered a problem, the session could not be created to solve this problem we followed the following steps:

1-Read the documentation of the ibm_watson_assistant [27] but it was not helpful.

2- Refer to the documentation of ibm cloud [28] there was not documentation for dart language, but we used python documentation it turns out the assistantId is the environment id value not assistant id value and the problem was solved.

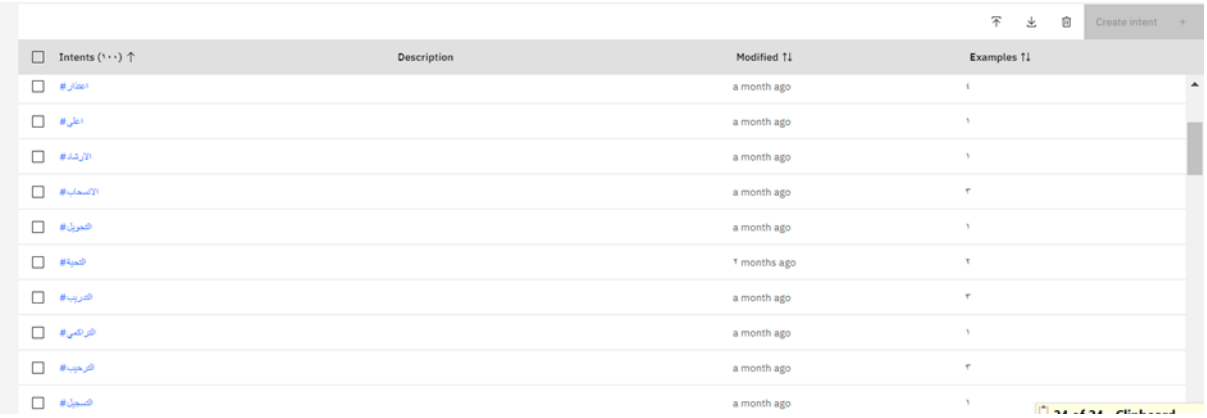


Figure 29: A list of intents with their examples

```dart
import 'package:ibm_watson_assistant/ibm_watson_assistant.dart';
import 'package:ibm_watson_assistant/models.dart';


class ChatbotService {
  late IbmWatsonAssistant bot;
  late String _sessionId;
  String get sessionId => _sessionId;

  ChatbotService() {
    final auth = IbmWatsonAssistantAuth(
      version: '2021-06-14', //4006e7f3-9ac5-4a92-ae6b-9622bb48532e
      assistantId: "b84cf70e-4299-4055-820a-33e7e6498809",

      url:
          "https://api.au-syd.assistant.watson.cloud.ibm.com/instances/339ffd31-1a97-4bed-b529-f24e4ba8ac3
      apikey: "jCg-PB4qj0un1OBCfv477FDywn-FvFKagTo6bMaiw6sp",
    );

    bot = IbmWatsonAssistant(auth);

    print('Initialized Chatbot Service');
  }
```

Figure 30: Snapshot of chatbot service class

```dart
  Future<String> createSession() async {
    print('creating session');
    try {
      _sessionId = (await bot.createSession())!;
    } catch (e) {
      print('session error: $e');
      return e.toString();
    }
    print('created session: $_sessionId');
    return _sessionId;
  }

  Future<IbmWatsonAssistantResponse?> sendInput(String input) async {
    print('Sending chatbot input: $input');
    if (_sessionId == null) await createSession();
    try {
      return bot.sendInput(input, sessionId: _sessionId);
    } catch (e) {
      print('Error sending chatbot input: $input.\n$e');
      // return e;
    }
  }

  Future<void> deleteSession() async {
    await bot.deleteSession(_sessionId);
    _sessionId = "";
  }
}
```

Figure 31: Snapshot of chatbot service class

The response(query) function is primarily used to communicate with the chatbot. First, it utilizes the chatbot Service object by creating a session and sending user questions and when the response is received it will show it to the user see Figure 32.

```dart
class _ChatbotScreenState extends State<ChatbotScreen> {
  final messageControler = TextEditingController();
  ChatbotService chat = ChatbotService();

  ScrollController scrollController = ScrollController();
  List<Message> msgs = [];
  bool isTyping = false;

/* ------------------------------------------------------------

this method for sending question and receiving chatbot answer @ibtihalx
------------------------------------------------------------------------*/
  void response(query) async {
    chat.createSession().then((value) => {
          chat.sendInput(query).then((response) => {
                setState(() {
                  msgs.add(Message(
                      false, response?.output?.generic?[0].toJson()["text"]));
                })
              })
        });
  }
```

Figure 32: The response(query) function

4.5.2 Application Development

- Availability Hours

In the development of the "Wjjhni" application, one of the significant challenges we encountered was implementing availability hours for each advisor. This task required meticulous planning, attention to detail, and a thorough understanding of the user experience to ensure a seamless interaction with the application. Initially, we explored various options for presenting and managing availability, considering the nuances of advisor schedules and the need for a user-friendly interface.

We opted for a calendar-like booking appointment style see figures 33 and 34 to represent availability, as it provided a clear visual representation of time slots. However, we quickly realized that a simple table format was insufficient to incorporate the complexity of features we envisioned, such as the ability to delete or edit booked times. To address this, we devised a different interface that allowed advisors to easily modify the dates and times marked as booked or available see figures 35 and 36.

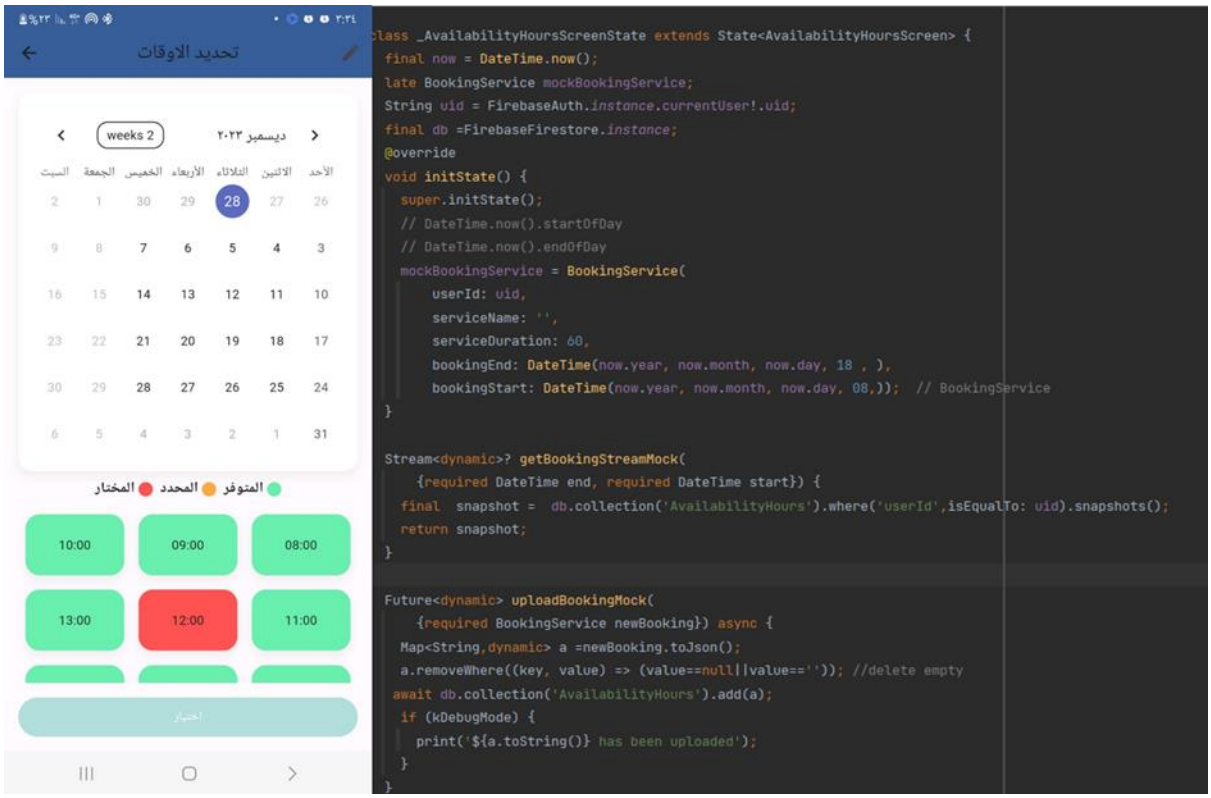


Figure 33: Snapshot of calendar like availability hours feature and its implementation class

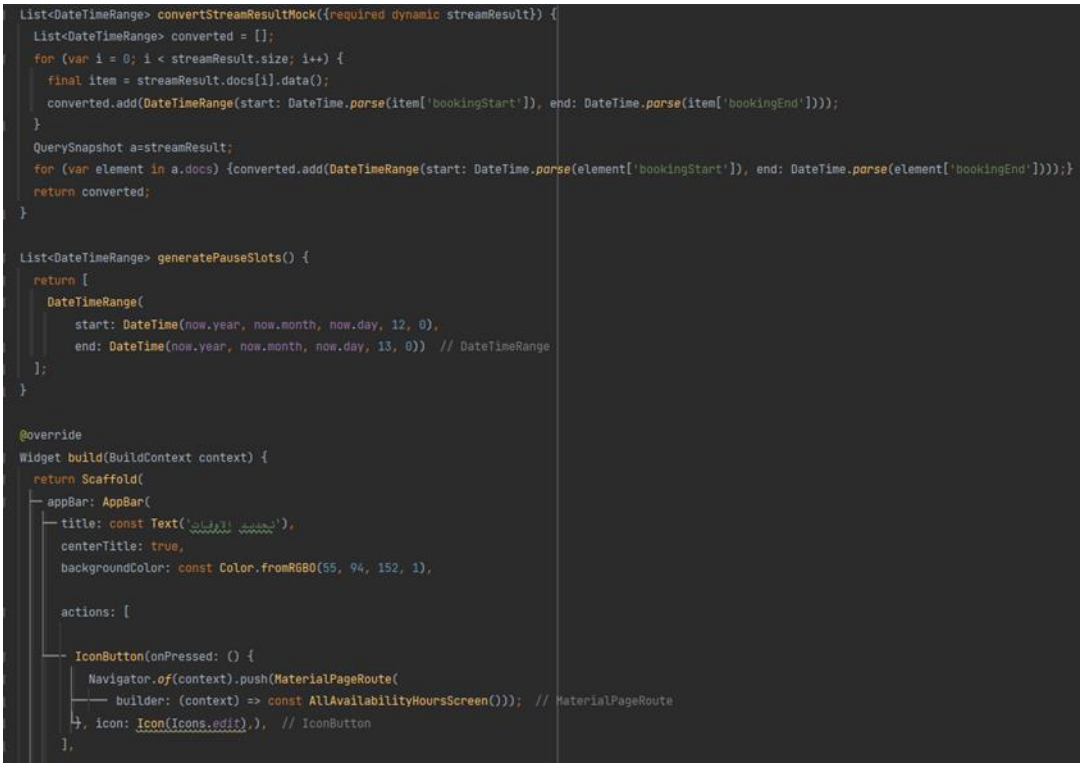


Figure 34: Cont. Snapshot of availability hours feature implementation class

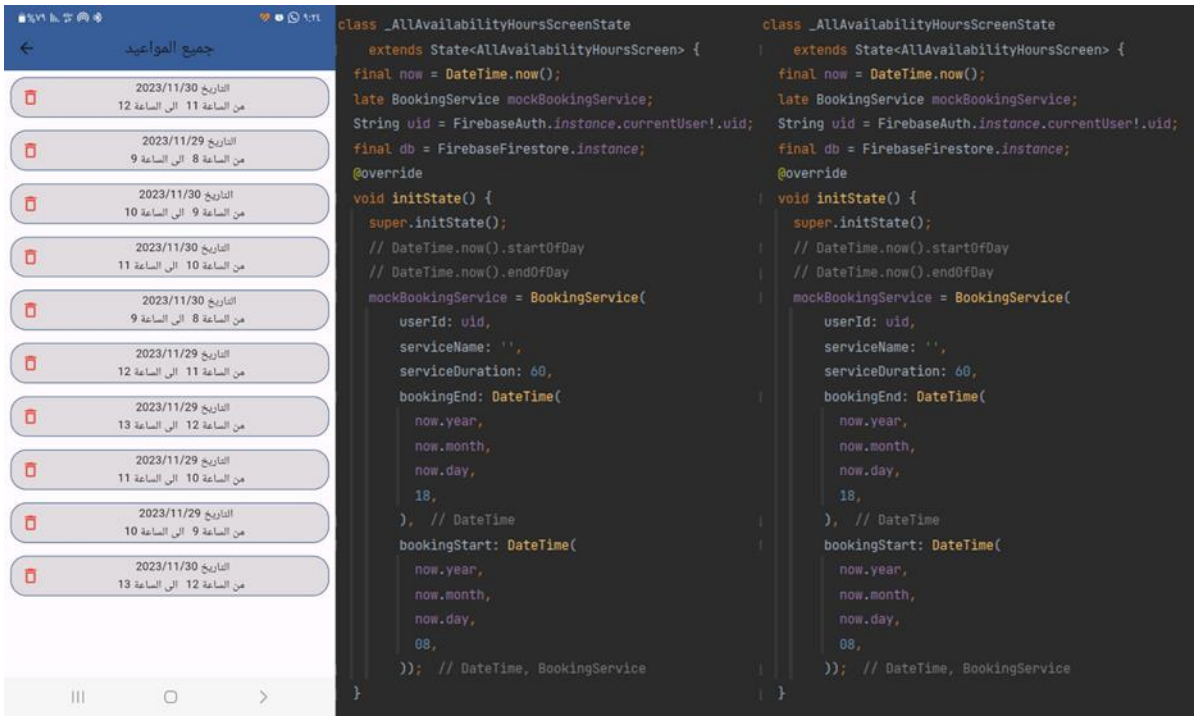


Figure 35: Snapshot of all availability hours feature and its implementation class

```
List<DateTimeRange> generatePauseSlots() {
  return [
    DateTimeRange(
        start: DateTime(now.year, now.month, now.day, 12, 0),
        end: DateTime(now.year, now.month, now.day, 13, 0))  // DateTimeRange
  ];
}
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('جميع المواعيد'),
      centerTitle: true,
      backgroundColor: const Color.fromRGBO(55, 94, 152, 1),
    ),  // AppBar
    body: Directionality(
      textDirection: TextDirection.rtl,
      child: StreamBuilder<dynamic>(
          stream: getBookingStreamMock(),
          builder: (context, snapshot) {
            if (snapshot.hasData && snapshot.data != null) {...} else if (snapshot.connectionState == ConnectionState.waiting) {
              return const Center(
                child: CircularProgressIndicator(),
              );  // Center
            }
            return Center(
              child: Text("لا يوجد مواعيد"),
            );  // Center
          }),  // StreamBuilder
    ),  // Directionality
  );  // Scaffold
}
```

Figure 36: Cont. Snapshot of all availability hours feature implementation class

The implementation also involved creating features such as a sign-in/sign-out functionality for users, ensuring a personalized experience. Each advisor could have a dedicated profile with the ability to update essential information. Implementing these features required careful consideration of user flows, security measures, and an intuitive design to enhance overall usability.

Throughout the development process, we embraced an iterative approach, adapting our solutions based on user feedback and testing results. Additionally, we made use of modern technologies and best practices to create a robust and user-friendly application. The integration of features like availability management, profile customization, and seamless deleting of booked slots proved to be essential components in delivering a comprehensive and user-centric experience within the "Wjjhni" application.

- Android Studio, Flutter, And Dart

In developing the "Wjjhni" app, we utilized Android Studio, Flutter, and Dart, creating a cross-platform mobile application with a responsive UI. Flutter's hot reload feature expedited development, allowing real-time adjustments and immediate visualization on an Android Galaxy phone. This dynamic environment facilitated UI refinement, ensuring a polished design. An Android Galaxy phone served as our primary testing device, simulating real-world interactions. Extensive testing validated features like availability management and profile customization. Leveraging these technologies, we delivered a robust and visually appealing "Wjjhni" app, tailored to users, combining functionality and user-friendliness.

- Firebase

In developing the "Wjjhni" application, we streamlined data management by integrating Firebase Google as our backend database. Firebase's cloud-based solution facilitated secure storage and retrieval of key information, such as advisor availability, user profiles, and booking details. This integration, coupled with Flutter and Dart, enabled real-time communication between the application and Firebase, ensuring dynamic updates. Firebase Authentication played a pivotal role in implementing a secure sign-in/sign-out mechanism for users. Additionally, Firebase Cloud Firestore served as our database, accommodating complex data structures and enhancing scalability.

4.5.3 Website Development

Wjjhni admin website implemented using HTML, CSS, and JS for the front end, and backend using PHP and Firebase. We followed the following steps to implement the admin's features.

- Design UI which is the front end of the pages and use dummy data.
- Download required libraires to connect with firebase.
- Start building backend for the features and retrieve data from database.

- Server Configuration

Setting up our local host server with the required configuration was a challenging but a crucial step in the development process. We used MAMP, which is the local server environment with version 8.1 of PHP. We have made modifications to the php.ini file which is a special file for PHP. It is where we declare changes to our PHP settings. We have enabled the extensions for grpc and sodium see Figure 37.

```
extension=php_grpc

; The MIBS data available in the PHP distribution must be installed.
; See https://www.php.net/manual/en/snmp.installation.php
;extension=snmp

;extension=soap
;extension=sockets
extension=sodium
;extension=sqlite3
;extension=tidy
;extension=xsl
```

Figure 37: Server extensions configuration

gRPC is a modern, open-source, high-performance remote procedure call framework we used to communicate with Google Cloud Firestore service [29] we download this library and store it in "ext" folder in our MAMP.

Sodium is a modern, easy-to-use software library for encryption, decryption, signatures, password hashing, and more. [30] it was required extension to work with firebase authentication.

PHP uses libraries like cURL or Guzzle for making HTTP requests. When making secure requests (HTTPS), these libraries need to validate the SSL/TLS certificates of the servers they communicate with. The cacert.pem file can be used to provide the necessary certificates for this validation [31] we download cacert.pem and set it is configuration see Figure 38.

```
[curl]
; A default value for the CURLOPT_CAINFO option. This is required to be an
; absolute path.
curl.cainfo ="C:/MAMP/bin/apache/bin/cacert.pem"
```

Figure 38: Setting cacert.pem path

Then we hosted our website on Hostinger, we followed a series of steps. Firstly, we accessed the Hostinger control panel and opened the file manager. Within the file manager, we uploaded all the necessary project files and ensured they were placed in the appropriate directories.

Next, we proceeded to update the composer. Composer is a dependency management tool for PHP that installs the required libraries and dependencies for our project. Through SSH (Secure Shell).

Finally, with the project files uploaded and the composer updated, we were ready to host our website on a server. Hostinger provided the necessary infrastructure to make our website accessible to the public. By hosting it on a server, we were able to ensure that our website could be accessed and utilized by visitors.

- Problem While Working On Github

We encountered a problem while working on GitHub. The problem was caused by uploading all the libraries for the project to GitHub. This problem caused the project to increase in size and made it difficult to share it with other teammates. To solve this problem, we followed these steps:

1- We opened the project folder using the Terminal.

2- We created a ".gitignore" file using the following command "touch .gitignore". This file ignores unnecessary files that can be downloaded using Dependency Managers (which in the

case of our project represents a composer). The ".gitignore" file did not exist when the project was uploaded for the first time.

3- Then we opened this file using an editor and put in it the name of the folder (Vendor) that contains the libraries that increased the size of the project.

4- We made a commit and pushed it to the GitHub repository.

5- We execute the following command "git rm -r –cached" to untrack all files

6- We execute the following command "git add." which returns all files except the ignored files (which we added in the ".gitignore" file).

7- Then we execute the following command "git commit -m gitignore fixed" then pushed this commit and it worked and thus the problem was solved.

4.5.4 Consultation

During our consultation process, we explored the "Stack Overflow" website [32] and engaged with the community on the "Fiverr" application [33] to seek expertise in chatbot development. After explaining our system, the experts recommended utilizing either Dialogflow or IBM Watson for Sprint-1. Furthermore, in Sprint 2, we sought consultation from Dr. Abeer Al-Drees regarding the ER Diagram.

5 System Evaluation

5.1 User Acceptance Testing

We conduct User Acceptance Testing to check if the system fulfills requirements and can be used by the end users. To do this testing, we created a testing team comprised of 21 end users from each student, academic advisors, and admins. We gave them the system and a list of tasks:

For the website, the admin asked to do the following:

1. Log in.
2. Add academic advisors to the system.
3. View all academic advisors in the system.
4. Add students to the system.
5. View all students in the system.
6. Log out.
7. Reset a password.
8. Assign academic advisors to students.
9. Edit the academic forms.
10. Edit the department's plan.
11. Remove students and academic advisors from the system.
12. Add important dates to the academic calendar.

For the mobile application, the student asked to do the following:

1. Log in.
2. Update the profile.
3. Ask the AI chatbot.
4. Log out.
5. Reset the password.

6. Book an appointment with the academic advisor.
7. Cancel an appointment with the academic advisor.
8. Chat in real time with the academic advisor.
9. View academic forms.
10. Evaluate the academic advisor.
11. View all the department plans for each major.
12. View important dates on the academic calendar.
13. Calculate the absence hours allowed for a course.

For the mobile application, the academic advisor asked to do the following:

1. Log in.
2. Update the profile.
3. Set the availability hours.
4. Log out.
5. Reset the password.
6. Cancel an appointment with a student.
7. Chat in real time with the students.
8. Record the students' visit.
9. View academic forms.
10. View all the department plans for each major.
11. View important dates on the academic calendar.

5.1.1 Demographics of Participants

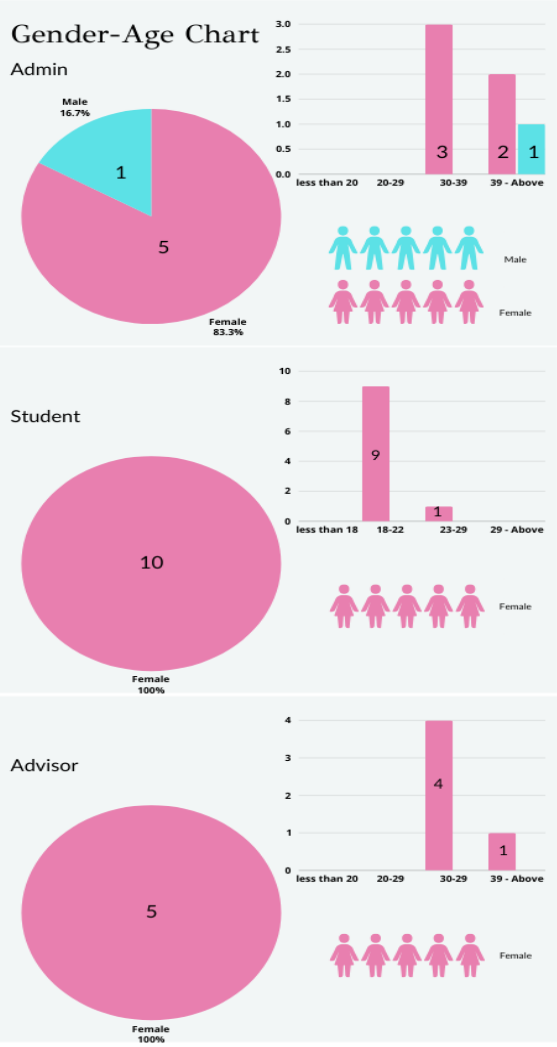


Figure 39 Demographics of Participants

5.1.2 Questionnaire Results

We gave the users after the testing a questionnaire, see Tables (4, 5, 6) to evaluate the system and obtain their feedback.

- For the website (Admin)

In the testing phase, we presented a series of questions to the admins to evaluate the website of the system. The following table illustrates the number of 6 individuals tested and

their corresponding responses for each given answer in each column, where the numbers represent the count of individuals who chose that particular answer.

Question	Very satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
How attractive is the overall design of the website?	3	3			
How do you rate the ease of browsing on the website?	5	1			
How easy is it to navigate through the website's pages?	4	2			
How satisfied are you with the website pages' loading speed?	3	2	1		
How good is the website's performance across different web browsers?	4	1	1		
How satisfied are you with the feature for adding students/advisors to the database?	4	2			
How satisfied are you with the feature of assigning students to academic advisors?	4	2			
How satisfied are you with the feature of attaching a form/plan to the database?	4	2			
Question	**Yes**		**No**		
Have you encountered any errors or problems while using the website?			6		

Table 4 Questionnaire Results for Admins (Website)

- For the application (Student)

In the testing phase, we presented a series of questions to the students to evaluate the application of the system. The following table illustrates the number of 10 individuals tested and their corresponding responses for each given answer in each column, where the numbers represent the count of individuals who chose that particular answer.

Question	Very satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
How would you rate the overall ease of use of the application?	8	2			
How do you rate the ease of accessing the features you want?	7	3			
How satisfied are you with the app's speed and responsiveness?	7	3			
During your experience with the app, how satisfied are you with the feature of AI chatbot'?	3	5	2		
How confident are you in the security of your data while using the app?	7	2	1		
How satisfied are you with the app's user interface?	5	4	1		
How easy is it to navigate through the different sections of the app?	8	2			
Did the app meet your expectations regarding academic advising?	6	4			

Do you think 'Wjjhni' provides real added value to the academic advising process?	7	3			
Based on your experience using the app, how likely would you be to recommend it to other individuals or academic institutions based on its usability and functionality?	7	3			
Question	**Yes**		**No**		
Have you encountered any errors or problems while using the app?			10		

Table 5 Questionnaire Results for Students (mobile app)

- For the application (Academic Advisor)

In the testing phase, we presented a series of questions to the Academic Advisors to evaluate the application of the system. The following table illustrates the number of 5 individuals tested and their corresponding responses for each given answer in each column, where the numbers represent the count of individuals who chose that particular answer.

Question	Very satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
How would you rate the overall ease of use of the application?	3	2			
How do you rate the ease of accessing the features you want?	3	2			
How satisfied are you with the app's speed and responsiveness?	3	2			
How satisfied are you with the feature of adding	4	1			

office hours in 'Available Hours'?					
How confident are you in the security of your data while using the app?	3	1	1		
How satisfied are you with the app's user interface?	3	2			
How easy is it to navigate through the different sections of the app?	4	1			
Did the app meet your expectations regarding academic advising?	3	2			
Do you think 'Wjjihni' provides real added value to the academic advising process?	4	1			
Based on your experience using the app, how likely would you be to recommend it to other individuals or academic institutions based on its usability and functionality?	2	3			
Question	**Yes**		**No**		
Have you encountered any errors or problems while using the app?	2		3		

Table 6 Questionnaire Results for Academic advisors (mobile app)

The users provided us with valuable feedback and suggestions about the system through the questionnaire. We find that their feedback very helpful and we will consider it in the future while improving and implementing.

In the initial questionnaire it involved five users (admins), as indicated in Table (4), it was observed that they found the website to be highly clear and easily navigable. Additionally, they provided constructive suggestions, such as developing the feature which was an addition to the ability of admin to upload academic forms and only to be viewed by users, but to make them sign the academic form and upload it through our system.

The second questionnaire involved ten users (students), as depicted in Table (5). The feedback indicated that the application is highly clear and easy to navigate, boasting a commendable response time. Users reported a positive experience with the chatbot. Suggestions for improvement included enhancing the chatbot to answer more questions efficiently and improving the application to work with IOS system.

The final questionnaire involved five users (advisors), as indicated in Table (6). The feedback revealed that the application was clear, highly navigable, boasted a commendable response time, and was considered visually attractive. Suggestions for enhancement included adding more or less than one hour as an available hour of the advisor, and making sure the advisor's notes she wrote about the student are saved when the advisor get changed from one to another. However, one user mentioned that there was a yellow line that appears when the advisor add notes about students, the presence of a yellow line when adding notes about students is an issue observed in the emulator only. The functionality works well on actual phones, indicating that the problem is specific to the emulator only.

5.2 Quality Attributes (NFR testing)

User story	Quality Attribute	Measure	Results

As a user, I want the Wjjhni application to load within 2 seconds of lunch so that I will not have to wait long for the application to load.	Performance: Ability to meet timing requirements. When clicking on the icon, the system must load quickly.	Latency: The time between the arrival of the stimulus and the system's response to it [34]	The Wjjhni application was tested for its loading time across 5 loading requests. We can calculate the latency using the formula: Latency = total time for loading requests. The maximum latency recorded was 7.79 seconds, and the minimum latency recorded was 7.54 seconds. The average latency can be calculated as follows: Average Latency = (total time for 5 load requests) / (number of load requests) Average Latency = (7.61+7.64+7.79+7.55+7.54) / 5 requests = 7.628 seconds. This means our initial estimate was underestimated, as beginners, and the ideal loading time is between 7-8 seconds.
As an admin, I want the admin dashboard pages to load in less than 3 seconds so that I do not get frustrated by slow performance.	Performance: Ability to meet timing requirements. When clicking on a particular page, the system must load quickly	Latency: The time between the arrival of the stimulus and the system's response to it [34]	Wjjhni website was tested for its page loading time to 10-page loading requests. We can calculate the latency: Latency = The time the page takes to load. The maximum latency is 4 seconds. The minimum latency is 2 seconds. The average latency can be calculated by using the following formula:

			Latency = (Total time for 10 pages load requests/ Number of page load requests) Latency= (2.5+2.8+3+3.1+3.2+3.4+ 3.5+3.6+3.7+4)/10=3.28 seconds. This result is close to our assumption that was less than 3 seconds.
As a user, I want the system to be available 99% of the time I try to access it, so that I don't get frustrated.	Availability: Can the software carry out a task when needed?	Availability: Measured as (uptime) / (total time observed) [34]	We observed the running of our application for a period of 3 days. The uptime was calculated as follows: Uptime=72 hours−1 hour of shutdown from Firebase=71 hours. Total observation time =3 days×24 hours per day =72 hours. The availability of the application was calculated using the formula: Availability= (Uptime/Total observation time) ×%100. Availability= (71/72) ×100%=98.6% The period over which the application is observed can significantly affect the availability metric. The longer the observation period, the more opportunities there are for the application to experience downtime, which in turn affects the

			overall availability metric.
As a user, I want to have a clear and simple interface, so that I can learn how to use the application in less than 15 minutes.	Usability: user's experience when interacting with products or systems.	Using the following metrics: task time: (Average Time for n users) / (total number of users [34]	During our observation of 4 users testing our application, we monitored their learning time to understand their engagement with the features. Among them, 3 were students, and one was an advisor. The maximum time spent by any user to learn about the application features was 10.5 minutes. The minimum time was 7 minutes. On average, these 4 users took 8.75 minutes to familiarize themselves with all the app features. Notably, two of the students and advisor out of the 4 users successfully learned all the app features within 10 minutes.
As a student, I want the AI chatbot to answer within 10 seconds so that I won't have to wait a long time.	Performance: Ability to meet timing requirements. When asking the chatbot, it should answer within 10 seconds.	Latency: The time between the arrival of the stimulus and the system's response to it [34]	We asked the chatbot 5 questions and recorded the time it takes to answer each one. Therefore, latency is defined as the time for answering questions. The maximum latency was 4.03 seconds. And the minimum latency was 2.53 seconds. We can calculate the average latency as follows: Average Latency = (total time for answering 5 questions) / (number of questions)

			Latency = (4.03+3.44+2.58+2.83+3.53)/ 5 = 3.282 seconds.

Table 7 Quality Attributes (NFR testing)

5.3 Discussion

We employed different testing approaches for each user group. We met with administrators in person at the college to gather their feedback. For advisors, we successfully coordinated testing sessions by reaching out to them via email and scheduling meetings at the college. We also conducted in-person meetings with students to gather their feedback. These testing methods allowed us to engage administrators, advisors, and students in the evaluation process, ensuring comprehensive feedback from multiple user perspectives.

The current version of Wjjhni system has met the user's requirements and specifications. Based on the results shown in Section 5.1.2, results indicated that the purpose of the system was clear to the users, and their feedback provided valuable insights that helped to improve the system. By incorporating the feedback that we received from testers during the UAT phase, we were able to make valuable changes in our system that helped us gain our users satisfaction. The feedback leads us into fixing some errors that happened to some users like the yellow line in the add note screen.

After we tested the non-functional requirements, we concluded that our performance was good, but our assumptions was wrong due to the lack of our knowledge in the performance quality, we had a 7.628 second average latency across 5 loading requests, which suggests that the performance of the application is good but also has the potential for enhanced speed. As for usability, the fact that 3 out of 4 users were able to learn all the app features in less than 10 minutes shows that the current version of Wjjhni application is relatively easy to learn and pick up, and the availability was good.

The positive outcomes of our testing affirm that our system demonstrates good performance, usability, and availability. We are grateful for the valuable feedback received from our users, including suggestions such as the incorporation of a feature enabling the signing and uploading of academic forms within the system. We genuinely appreciate these suggestions and assure our users that we will thoroughly evaluate and consider their

recommendations for future updates. By actively incorporating user feedback, we are dedicated to continuously improving Wjjhni, making it a more comprehensive and user-friendly platform that meets the evolving needs of our users.

6 Conclusions and Future Work

In conclusion, we recognize that numerous students struggle in obtaining prompt answers to their questions from academic advisors, while academic advisors face challenges in communicating with their students. "Wjjhni" is a comprehensive application designed to address the challenges faced by students and academic advisors. Its main feature is an AI chatbot that answers frequent questions about courses, university rules, and guidelines, utilizing the IBM Watson Assistant which is an AI-powered chatbot and virtual assistant platform developed by IBM.

To complete this release, we followed the Agile methodology, which allowed us to divide the requirements into three sprints. Each sprint took four weeks and went through a five-step process: planning, designing, building, testing, and reviewing. We found that having daily meetings was crucial to staying on track and making sure everyone was on the same page. The scrum team was an essential part of the process, providing valuable input and contributions. By breaking down the tasks into smaller pieces, we were able to assemble everything together to create the release of one product. Overall, using the agile methodology gave us a chance to gain experience with this approach and work more efficiently as a team.

The previous version of "Wjjhni" has met some requirements and specifications of our users. In sprint zero, we have completed some of the documentation for the project. In sprint one, we implemented the main feature "AI Chatbot". In release one which included sprint two, we implemented more features related to all users in the application and the website such as log-in and log-out, view and update profiles, add users, and add availability hours.

This current version, which includes sprint 3-5 we implemented the rest of the features which are: students can book and cancel appointments, admin assigns student to advisor, online chat, advisor records student visit, academic forms, student evaluates advisor, department plans, admin deletes students and advisors, admin adds important dates to calendar, absence hours calculator, use can view calendar.

6.1 Global and local impact

We believe that our system has implications locally and globally. Locally in our college, when many students need services related to academic advising, whether simple or important instructions, they find it difficult to obtain these services in one system linked to academic advisors. "Wjjhni" will combine all academic advising services in one system that facilitates the process to students and advisors. In a global aspect, it will be easy for KSU university to adopt our system for all colleges, creating one academic advising system as well as becoming an international system to be used all over the world for students and academics advisors to use.

6.2 Problems and challenges encountered during software development.

No project proceeds without difficulties of some kind, expected or otherwise. A diverse range of obstacles were faced in bringing "Wjjhni" to life, including both programming and soft skills challenges. We first encountered some difficulties in understanding the requirements and which information was necessary to gather from both the student and advisor, but after conducting many interviews and questionnaires, we had a better understanding of the requirements, also developing "Wjjhni" while balancing coursework and learning a new framework added significant time pressures and management challenges.

Hardware challenges added to our journey, with some laptops and devices experiencing breakdowns and malfunctions. These unexpected hurdles prompted us to navigate through unforeseen obstacles, demonstrating resilience and adaptability in overcoming technical difficulties.

But Through wrestling with time constraints and unanticipated time sinks, our team developed stronger time management skills. Developing the AI Chatbot system for "Wjjhni" introduced

several challenges that complicated our progress, starting with a selecting platform step to the implementation step. In selecting the platform, first, we selected DialogFlow as our platform to build the AI chatbot, but after more searching, we chose IBM Watson as a platform. Also with IBM Watson, we faced an issue with the budget which resulted in a change in the backlog to manage the budget well, we also faced issues with hosting were in Saudi Arabia, IBM Watson does not provide hosting for individuals and only for companies. that resulted in us contacting IBM Watson customer service to find a suitable solution.

In our implementation of the IBM Watson chatbot, we encountered a limitation with the number of intents available. Unfortunately, we were unable to exceed the limit of 100 intents imposed by IBM Watson's free tier. To overcome this constraint and unlock unlimited intents, we explored the option of subscribing to a paid plan. However, we faced a hurdle as the subscription process in Saudi Arabia required a tax number, which we, as students, did not possess. Consequently, we were unable to avail the subscription and expand the intent capacity of our chatbot within the given constraints.

In addition, we took the initiative to reach out to IBM Watson support, explaining our situation and requesting an exception to the intent limit. Unfortunately, despite our efforts to seek assistance, we did not receive a response from IBM Watson regarding our request. This lack of response further hindered our ability to expand the intent capacity of our chatbot beyond the imposed limit.

6.3 Limitations of the system

"Wjjhni" is an Arabic mobile application that supports an Android platform designed to manage the academic advising for students and advisors in the College of Computer and Information Sciences only. "Wjjhni" at this stage, doesn’t support the English language or IOS platform.

6.4 The main contribution of the project

"Wjjhni" aims to make the academic advising process easier between students and advisors. it will support many features that other systems lack, like an AI chatbot, an advisor communicating with the students via chat, and others. "Wjjhni" combines all academic advising services in one system, unlike many systems.

6.5 Future work

Now that the GP2 is complete, our focus shifts to ensuring widespread adoption of our system in colleges and universities in Saudi Arabia as well as the rest of the world. We aim to have many educational institutions embrace our solution, benefiting students with personalized academic advice. We will promote our system, establish partnerships, gather user feedback, and make improvements accordingly. Our goal is to make a positive impact in education by providing accessible and tailored guidance to students.

7 Acknowledgements

We would like to express our sincere gratitude to Dr.Hend Albassam, our supervisor, for her invaluable guidance, encouragement, and support throughout this journey. her expertise and insights have been instrumental in shaping the direction of wjjhni.

We are deeply thankful to the information technology department, king Saud university, for providing us with the necessary resources and facilities to carry out this research. The academic environment fostered by the institution has been conducive to our intellectual growth and development.

We also owe the biggest thank you to our family. for their endless love, support, encouragement, and understanding. Their unwavering belief in us has been a constant source of motivation for us throughout the making of this project, this would have been a very different experience without their encouraging words and support, everything we do is for them, and this is no exception.

Lastly, we would like to express our appreciation to all those who have contributed in any way, no matter how small, to the completion of wjjhni.

8 References

[1] J. W. Adeel Khali1, “Role of Academic Advisors in the Success of Engineering Students,” *Universal Journal of Educational Research*, vol. 2, no. n1, pp. 73–79, 2014, Accessed: May 04, 2024. [Online]. Available: https://eric.ed.gov/?id=EJ1053985

[2] M. Awadh, “The Relationship Between the Quality of Academic Advising and the Perception of Academic Outcomes of Saudi College Students,” *International Research in Higher Education*, vol. 3, no. 4, p. 22, Nov. 2018, doi: 10.5430/irhe.v3n4p22.

[3] “Academic Advising Timeline | Walker College of Business Academic Advising Center.” Accessed: May 14, 2024. [Online]. Available: https://businessadvising.appstate.edu/about/academic-advising-timeline

[4] “كلية علوم الحاسب والمعلومات.” Accessed: May 04, 2024. [Online]. Available: https://ccis.ksu.edu.sa/ar

[5] “Academic Advising And Registration Unit.” Accessed: May 04, 2024. [Online]. Available: https://ccis.ksu.edu.sa/ar/node/2858

[6] R. Tamrakar and N. Wani, “Design and development of CHATBOT: A review,” *ResearchGate, Apr*, 2021.

[7] “Watson.” Accessed: May 04, 2024. [Online]. Available: https://www.ibm.com/blog/tag/watson/

[8] “IBM Developer.” Accessed: May 04, 2024. [Online]. Available: https://developer.ibm.com/

[9] “IBM Watson documentation.” Accessed: May 04, 2024. [Online]. Available: https://cloud.ibm.com/developer/watson/documentation

[10] “KSU Student.” Accessed: May 06, 2024. [Online]. Available: https://apps.apple.com/sa/app/ksu-student/id545166496

[11] D. Al-Ghadhban and N. Al-Twairesh, “Nabiha: An Arabic Dialect Chatbot,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 3, 2020, doi: 10.14569/IJACSA.2020.0110357.

[12] “منصة مُرشد.” Accessed: May 04, 2024. [Online]. Available: https://twitter.com/KSU_HELP?ref_src=twsrc%5Egoogle%7Ctwcamp%5Eserp%7Ctwgr%5Eauthor

[13] “KFUPM Student King Fahd University of Petroleum and Minerals.” Accessed: May 04, 2024. [Online]. Available: https://apps.apple.com/sa/app/kfupm-student/id1630637797

[14] “عمادة القبول والتسجيل - الارشاد الاكاديمي - الصفحة الرئيسية.” Accessed: May 04, 2024. [Online]. Available: https://academicadvising.kau.edu.sa/Default-210999-ar

[15] “Academic Advising Center.” Accessed: May 04, 2024. [Online]. Available: https://www.qu.edu.qa/sites/en_US/students/success-and-development/academic-advising

[16] “Welcome to the Future of Campus Life. Introducing the UIS Mobile App!” Accessed: May 04, 2024. [Online]. Available: https://www.uis.edu/uis-mobile-app

[17] “What is Agile Development?” Accessed: May 11, 2024. [Online]. Available: https://www.opentext.com/what-is/agile-development

[18] "What is Agile Methodology?" Accessed: May 11, 2024. [Online]. Available: https://www.edubridgeindia.com/blog/understanding-what-is-agile-methodology/
[19] S. J. Schwaber Ken, *The Scrum Guide*. 2020. Accessed: May 11, 2024. [Online]. Available: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100
[20] "What Are Scrum Artifacts and How Do I Use Them?" Accessed: May 11, 2024. [Online]. Available: https://niftypm.com/blog/scrum-artifacts/
[21] "Scrum blu." Accessed: May 11, 2024. [Online]. Available: https://www.bluprofessionals.com/wp-content/uploads/2023/03/SCRUM-LinkedIn.pdf
[22] "Deanship of Admission and Registration Affairs KSU University." Accessed: May 04, 2024. [Online]. Available: https://dar.ksu.edu.sa/en
[23] "King Faisal University | Abqaiq branch." Accessed: May 04, 2024. [Online]. Available: https://kfu.edu.sa/ar/Colleges/community/SiteAssets/Pages/acadmicadvisor/%D8%AF%D9%84%D9%8A%D9%84%20%D8%A7%D9%84%D8%A7%D8%B1%D8%B4%D8%A7%D8%AF%20%D8%A7%D9%84%D8%A7%D9%83%D8%A7%D8%AF%D9%8A%D9%85%D9%8A.pdf
[24] "King Faisal University | college of Arts." Accessed: May 04, 2024. [Online]. Available: https://www.kfu.edu.sa/ar/Colleges/arts/Documents/rfq143.pdf
[25] " Deanship of Admission and Registration Affairs | Taiba university." Accessed: May 04, 2024. [Online]. Available: https://www.taibahu.edu.sa/Pages/En/Sector/SectorPage.aspx?ID=40
[26] A. , et al. Abdellatif, "A comparison of natural language understanding platforms for Chatbots in Software Engineering," *IEEE Transactions on Software Engineering*, vol. 48, no. 8, pp. 3087–3102, 2022.
[27] "Ibm_watson_assistant: Dart Package," https://pub.dev/packages/ibm_watson_assistant.
[28] "Watsonx Assistant V2 | IBM Cloud API Docs," https://cloud.ibm.com/apidocs/assistant-v2?code=python.
[29] "Install gRPC for PHP." Accessed: May 04, 2024. [Online]. Available: https://cloud.google.com/php/grpc#:~:text=gRPC%20is%20a%20modern%2C%20open,must%20install%20gRPC%20for%20PHP
[30] " Introduction php," https://www.php.net/manual/en/intro.sodium.php#:~:text=Sodium%20is%20a%20modern%2C%20easy,build%20higher%2Dlevel%20cryptographic%20.
[31] "Savani, H. (2023) PHP curl request with certificate (CERT PEM file option) example, ItSolutionStuff.com," https://www.itsolutionstuff.com/post/php-curl-request-with-certificate-cert-pem-file-option-exampleexample.html#google_vignette.
[32] "Stack Overflow," https://stackoverflow.com/.
[33] "fiverr," https://play.google.com/store/apps/details?id=com.fiverr.fiverr.

[34] G. Gay, “Lecture 1: Software Quality, Verification, and Validation.” Jan. 20, 2021. Accessed: May 18, 2024. [Online]. Available: https://greg4cr.github.io/courses/spring21dit635/Lectures/Spring21-Lecture1-Intro.pdf

9 Appendix

9.1 APPENDIX A: Questionnaire for requirements elicitation

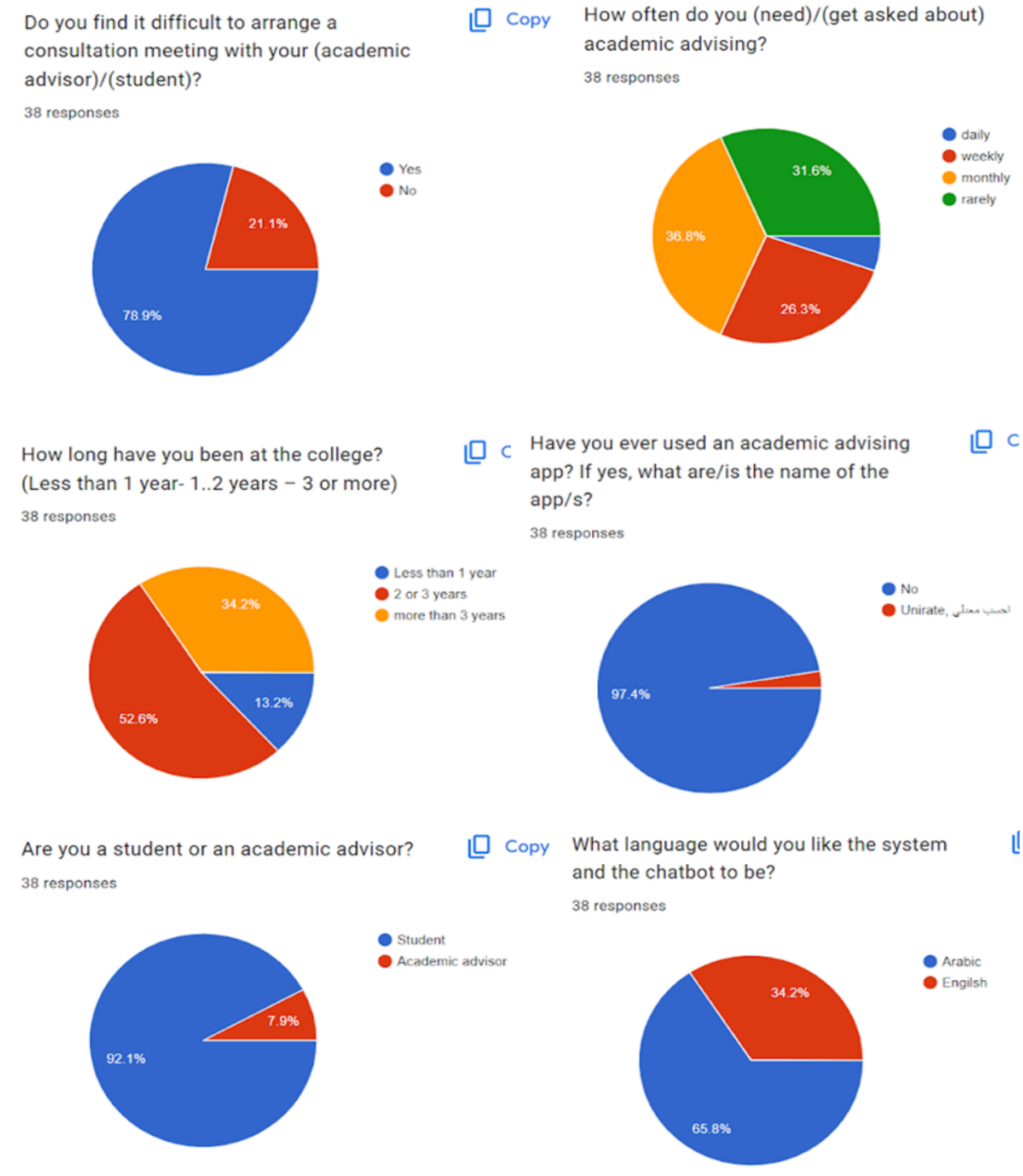
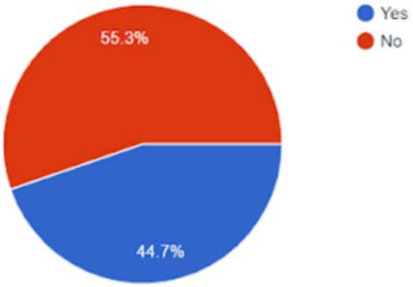


Figure 40 Questionnaire requirements elicitation

Is it easy to find important dates like the last day of drop-add course etc...

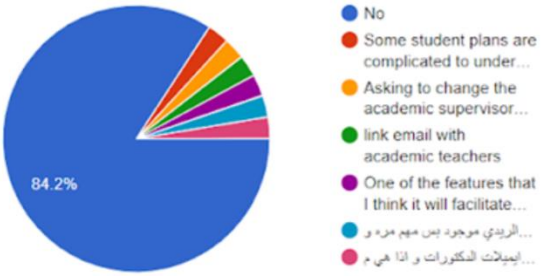
38 responses



Do you have any features that you would like us to provide at our app?

Copy

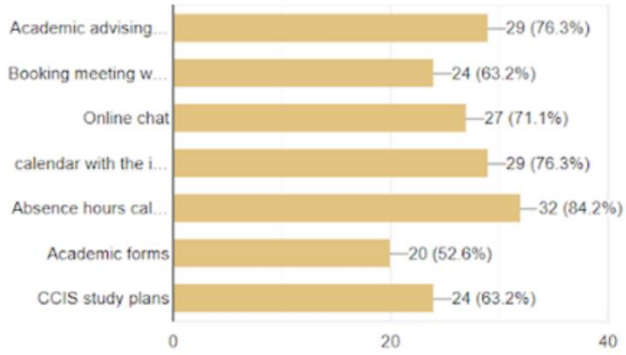
38 responses



Which feature will make you use the system?

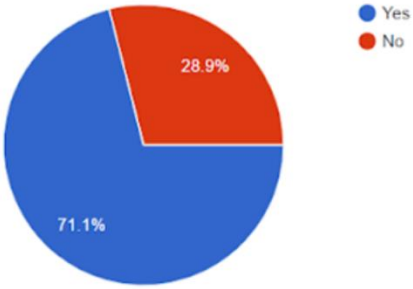
Copy

38 responses



Do you have difficulty finding specific academic forms?

38 responses



Which of the features listed below, you don't think you will need to use?

Copy

38 responses

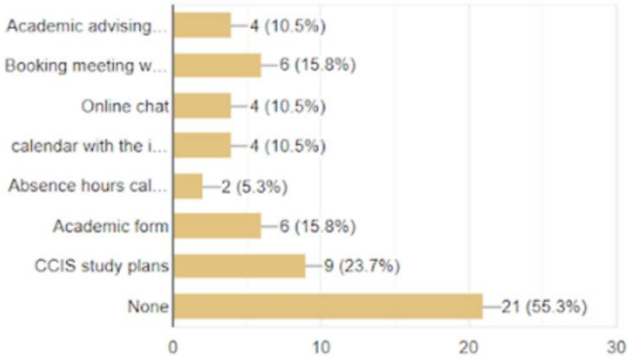


Figure 41 Questionnaire requirements elicitation

9.2 APPENDIX B: Interviews for requirements elicitation

Interview 1 outline	
Interviewee name: Dr. Ameera Almasoud	Interviewers: Lamia Alhelali and Ibtihal Almutairi
Location: ZOOM	Date: 17/9/2023
Questions	**Answers**
Q1\ As a student or an academic advisor, what are the academic problems that you face, other than we mentioned?	I think that your system will solve most of the problems, so no I don't have anything else in mind.
Q2\ Have you ever tried an academic advising app? If yes, how was your experience? If not, have you ever wanted to try one?	Yes, I used Microsoft teams. I loved it because I was able to chat to the students easily for any reason such as arranging a meeting.
Q3\ As a student or an academic advisor, what features would you like to be in our system?	I think you should set a reminder for the advisor if she doesn't answer the students' questions for a period. You should also provide a text space to write the reason for the meeting.
Q4\ Have you ever chatted with an AI chatbot, what did you like and what didn't?	Yes, but it wasn't academic advising.
Q5\ In your opinion, what makes an app, regardless of its purpose, easy to use?	The interface should be simple and clear.
Q6\ What are the main disadvantages of an app you used before, regardless of its purpose?	The buttons are small.
Interview 2 outline	
Interviewee name: Dr. Mashael Zamil	Interviewers: Lamia Alhelali and Ibtihal Almutairi
Location: At her office	Date: 17/9/2023
Questions	**Answers**
Q1\ As a student or an academic advisor, what are the academic problems that you face, other than we mentioned?	I think your system solved enough problems and it would be enough as a scope.
Q2\ Have you ever tried an academic advising app? If yes, how was your experience? If not, have you ever wanted to try one?	Yes, I did but it was very simple. The academic advisor wasn't involved in that system, so it wasn't the best experience.
Q3\ As a student or an academic advisor, what features would you like to be in our system?	I think the advisors should be able to set a student state for each student if needed, to get more attention, like a senior student.
Q4\ Have you ever chatted with an AI chatbot, what did you like and what didn't?	Yes, I tried ChatGPT. It had the biggest scope. The answers weren't that accurate, but it was okay.
Q5\ In your opinion, what makes an app, regardless of its purpose, easy to use?	The interface should be clear and not complicated.
Q6\ What are the main disadvantages of an app you used before, regardless of its purpose?	Colors too bright or too dark, and buttons size and place.
Interview 3 outline	
Interviewee name: Reem Alnasser	Interviewer: Lamia Alhelali

Location: ZOOM	Date: 17/9/2023
Questions	**Answers**
Q1\ As a student or an academic advisor, what are the academic problems that you face, other than we mentioned?	I do have problems with the academic advisor because she wasn't responding to my emails.
Q2\ Have you ever tried an academic advising app? If yes, how was your experience? If not, have you ever wanted to try one?	No, I would really love to try one.
Q3\ As a student or an academic advisor, what features would you like to be in our system?	I think it would be great if we were able to request a change of advisor.
Q4\ Have you ever chatted with an AI chatbot, what did you like and what didn't?	Yes, I loved that it would send the resource to anything he answers.
Q5\ In your opinion, what makes an app, regardless of its purpose, easy to use?	The language and interface.
Q6\ What are the main disadvantages of an app you used before, regardless of its purpose?	When there is only one language.

Interview 4 outline	
Interviewee name: Aldanah Albeshr	Interviewer: Lamia Alhelali
Location: ZOOM	Date: 17/9/2023
Questions	**Answers**
Q1\ As a student or an academic advisor, what are the academic problems that you face, other than we mentioned?	As an IT student I face several academic problems that can impact my studies. One of them is managing multiple assignments and deadlines, and I have faced a situation before where I forgot to submit an assignment before the deadline because I was busy doing another one. With numerous projects and coursework to juggle, it can be overwhelming to stay organized and ensure timely submission. Other problems I have faced are finding relevant resources, balancing coursework with other activities, understanding complex concepts, and tracking academic progress.
Q2\ Have you ever tried an academic advising app? If yes, how was your experience? If not, have you ever wanted to try one?	No, I have never but I would really like to try one.
Q3\ As a student or an academic advisor, what features would you like to be in our system?	- User-friendly interface with intuitive navigation. - Personalized academic planning and goal-setting tools. - Notifications for upcoming deadlines, class changes, or important announcements. - Access to academic support services, such as tutoring or writing assistance. - Integration with academic calendars and reminders for exams, assignments, and project due dates.
Q4\ Have you ever chatted with an AI chatbot, what did you like and what didn't?	Yes, I have chatted with an AI chatbot before. I liked its quick responses and availability but found limitations in handling complex queries and lacking personalized guidance.

Q5\ In your opinion, what makes an app, regardless of its purpose, easy to use?	5- In my opinion, an app is easy to use when it has an intuitive interface, clear navigation, minimal learning curve, organized information, fast performance, clear instructions, visual cues, customization options, accessibility features, and adequate feedback/error handling.
Q6\ What are the main disadvantages of an app you used before, regardless of its purpose?	The main disadvantages I have encountered in some apps include unintuitive interfaces, slow performance, limited integration, inadequate support/documentation, excessive notifications, lack of customization, no offline functionality, inconsistent updates, privacy concerns, and difficulty in exporting/sharing data.

Interview 5 outline	
Interviewee name: Alin Altowim	Interviewer: Lamia Alhelali
Location: ZOOM	Date: 17/9/2023
Questions	**Answers**
Q1\ As a student or an academic advisor, what are the academic problems that you face, other than we mentioned?	I faced common academic problems like difficulties in course selection, understanding degree requirements, scheduling conflicts, lack of guidance on career paths, and challenges in time management and study skills.
Q2\ Have you ever tried an academic advising app? If yes, how was your experience? If not, have you ever wanted to try one?	No I haven't and yes I would love to try that .
Q3\ As a student or an academic advisor, what features would you like to be in our system?	- Clear and comprehensive course catalog with detailed descriptions - Academic planning and degree progress tracking - Personalized recommendations for course selection based on academic goals and requirements - Integrated scheduling tools with the ability to check course availability and avoid conflicts - Access to academic resources such as study materials, research databases, and library services - Collaboration features to connect with advisors and peers for guidance and support - Notifications and reminders for important academic deadlines and events - Career guidance and internship opportunities - Feedback and rating system for courses and instructors
Q4\ Have you ever chatted with an AI chatbot, what did you like and what didn't?	Yes , I like the quick and efficient responses, availability 24/7, and the ability to provide instant answers to common questions. However, and what I don't like, I find limitations in the chatbot's ability to understand complex inquiries or provide personalized assistance. Also The lack of human interaction and empathy.
Q5\ In your opinion, what makes an app, regardless of its purpose, easy to use?	An app is considered easy when it has a user-friendly interface, intuitive navigation, and clear instructions. Consistency in design elements, logical organization of features, and responsive performance contribute to a positive user experience. Additionally, providing adequate onboarding tutorials or tooltips can help users understand the app's functionality quickly.
Q6\ What are the main disadvantages of an app you used before, regardless of its purpose?	Some common disadvantages of apps, regardless of their purpose, can include: - Poor user interface design or confusing navigation - Slow performance or frequent crashes - Insufficient or outdated content

	- Lack of customization options - Limited or non-existent customer support - Excessive ads or intrusive notifications - Privacy and security concerns

Table 8 Interviews outlines

9.3 APPENDIX C: Questionnaires for data collection

Figure 42 Data collection questionnaire

ما هي اسئلتك لمرشدتك الأكاديمية سواء كانت تخص المواد أو التخصص أو أي شي يتعلق بكلية علوم الحاسب والمعلومات والدراسة فيها ؟ (الرجاء كتابة أكبر قدر ممكن من الأسئلة)

35 responses

هل اذا واجهتني مشكله تخص الدرجات ارجع للأرشاد الاكاديمي؟

متطلبات المواد، امكانية الحذف والمستوى الصفري، تعادل المواد، التحويل ومتطلباته

كيف استفيد من التدريب التعاوني؟

ما هي متطلبات المواد التي أريد إضافتها أو حذفها
ما هو عدد الساعات المسموح به في الفصل
ما هو عدد الاعتذارات المسموح به خلال سنوات الدراسه
كيف اقوم بحذف ماده
كيف اقوم بإضافة ماده
كم عدد المرات التي استطيع ان أطلب مراجعه ورقة الإختبار النهائي
ما هو السجل المهاري
كيف أضيف ساعاتي التطوعيه في السجل المهاري
اريد ان اعرف نبذه عن مستوى صعوبه المواد
هل يسمح التحويل لتخصص آخر داخل الكليه أو خارجها
متى يمنع التحويل الى تخصص آخر

كيف أطور من نفسي ؟

ما هي اسئلتك لمرشدتك الأكاديمية سواء كانت تخص المواد أو التخصص أو أي شي يتعلق بكلية علوم الحاسب والمعلومات والدراسة فيها ؟ (الرجاء كتابة أكبر قدر ممكن من الأسئلة)

35 responses

هل هناك أشياء تساعدني على التميز الأكاديمي

كيف اختار مشرفة مشروع التخرج؟
كيف ممكن اقدم على جهة للتدريب؟ هل فيه معايير معينة؟ هل احتاج ختم من الكلية؟

كيف اقدم على تعديل الجداول + كيف أقدم على بعض الفرص المفيده داخل الكلية +عذر الغياب + اختبار بديل (موقفين بارب)

ما هي افضل المواد الاختياريه لي في هذا السنه ؟

هل يمكن اتدرب اذا التزم ؟

هل هذا الماده متطلب لماده أخرى؟

ما افضل اماكن تدريب للطالبات الحاسب ؟

كيف يمكن اني اطور تخصصي ؟

ما هي اسئلتك لمرشدتك الأكاديمية سواء كانت تخص المواد أو التخصص أو أي شي يتعلق بكلية علوم الحاسب والمعلومات والدراسة فيها ؟ (الرجاء كتابة أكبر قدر ممكن من الأسئلة)

35 responses

١- كيف اقدر ارفع طلب اضافه ماده؟ ٢- كم ماده أقدر أحذف؟ ٣- كيف استطيع الحصول على خطاب التدريب؟ ٤- كم هي المده المطلوبه للتدريب؟ ٥-كم ماده اقدر أخذها في التزم؟ ٦- ماهو السجل المهاري؟ ٧- كيف أحصل على السجل الأكاديمي ؟ ٨- ماهي المده النظامية لتخصصات كلية الحاسب؟ ٩- ماهي شروط مرتبة الشرف؟ ١٠- هل استطيع اضافه ماده لشعبة مغلقة؟ ١١- هل استطيع اضافه ماده لها متطلب؟

نموذج تبديل الشعبة هل يحتاج توقيع قبل رفعه على البوابة؟

ماهي اجراءات حذف ماده؟

هل اداوم بالشعبة الجديده اذا لم يتم الرد على طلبي بعد في البوابة؟

ما قد كلمتها الا بتسجيل المواد، بنفع اسوي كذا؟ لو سويت كذا بتخرب خطتي؟ اسألها عن اقتراحات لو إني احتاج استشاره

ما هي طريقه تبديل مجموعه؟
هل استطيع اضافه هذه الشعبه لهذه الماده؟

غالب اسئلتي لمرشدتي تتمحور حول المواد والجدول
مثل طلب تغيير مجموعه، او التسجيل في شعبة مغلقة وغيرها.

Figure 43 Data collection questionnaire

9.4 APPENDIX D: Questionnaire used for admin feedback in testing.(figure 44)

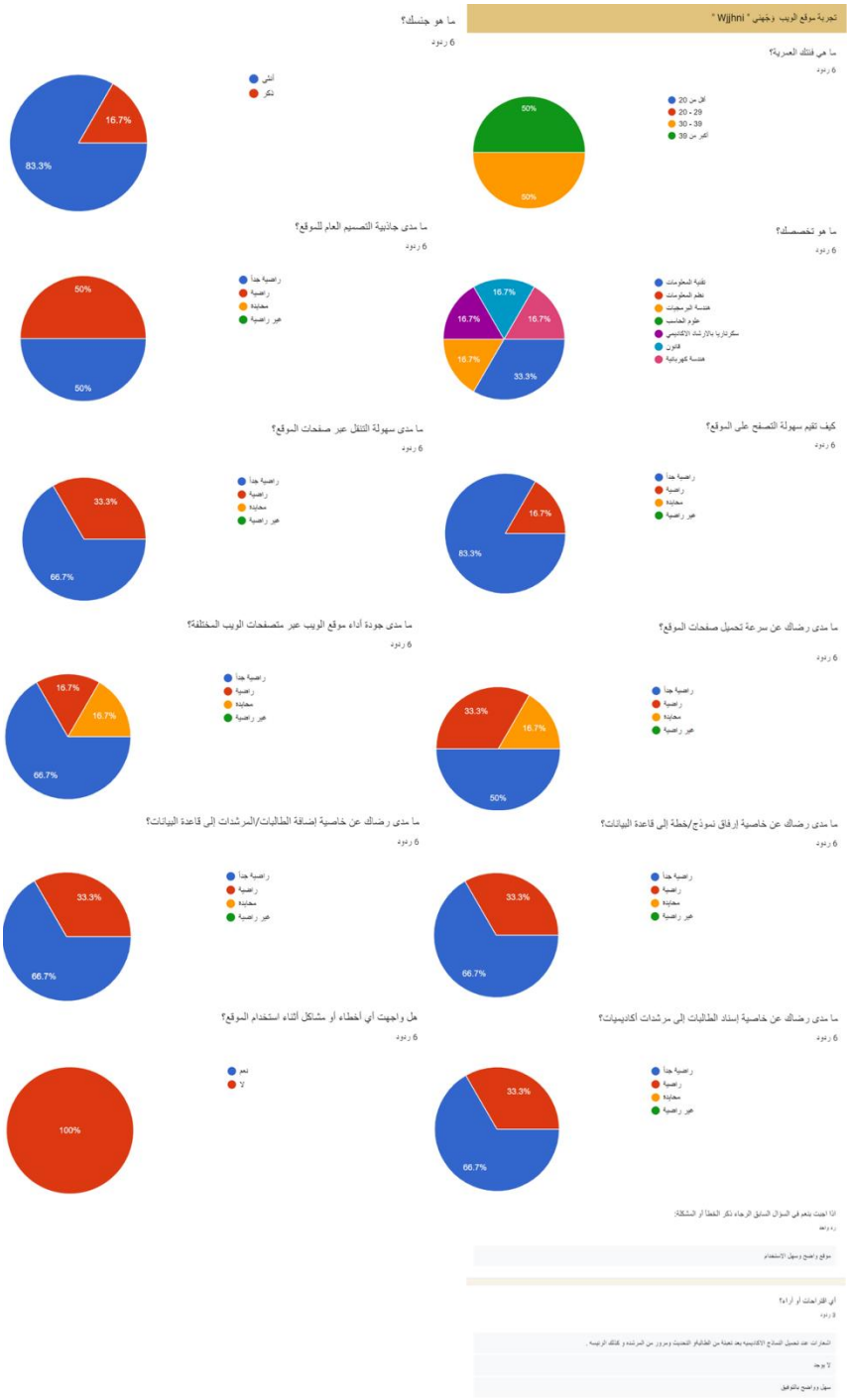


Figure 44 Admin questionnaire

9.5 APPENDIX E: Questionnaire used for academic advisors feedback in testing. (figure 45)

Figure 45 Academic advisor questionnaire

9.6 APPEBDIX F: Questionnaire used for students feedback in testing. (figure 46)

Figure 46 Student questionnaire