The Computer Science

DepartmentCommunity College

King Saud University

Academic Office Hours Mobile Application

Prepared By

Nawaf a Soaib | 438190130

Sultan Al-Anzi | 438190559

Graduation project submitted in partial fulfillment of the requirements for the award of associate degree in Computer Science

Supervised By

Dr.Khaled Al-Rajeh

first Semester 1441/2019

1. Declaration

We hereby declare that this submission is my own work and to the best of my knowledge it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the ward of any other degree or diploma at King SaudUniversity or any other educational institution, except where due acknowledgement is made inthe project. Any contribution made to the research by others, with whom we have worked at King Saud University – Community College – Computer Science Department or elsewhere, isexplicitly acknowledged in the project. We also declare that the intellectual content of this project is the product of our own work.

2. Acknowledgments

In performing our graduation project, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this project gives us much pleasure. We would like to show our gratitude **Dr. Khaled AL-Rajeh**, for giving us a good guideline for project throughout numerous consultations. We would also like to expand our deepest gratitude to all those who have directly and indirectly guided us in writing this report. In addition, we also thank the King Saud University (KSU) for consent to include copyrighted pictures as a part of our work. Many people, especially our classmates and team members itself, have made valuable comment suggestions on this project which gave us an inspiration to improve our work.We thank all the people for their help directly and indirectly to complete our project.

3. Abstract

The main objectives of our graduation project are about serves both students and instructors inthe college. Where the Academic office hours project will coordinate appointment between instructors and students using an easy method for both. Where the instructor can manage the appointments easily. The students can identify their problem when they make the request and they can determine both time and date. Based on the available times for the instructor. Our mostimportant goal is to make the meeting process easily which will help the student to find the answers to their questions.

4. Table of Contents (TOC)

CHAPTER 1

Introduction

In this chapter we will describe our project and the tools that we used it to buildthe project.

1.1 Overview

Every student in our community college can go the instructor in the office hours but sometimes some students does not know where the offices and offices hours'time so this system will help Both students and instructors to manage the officehours easily. In the development process we analyzed and designed the system on paper and then created the database and started working on it from what we reached in the analysis phase through the "DFD". Our system has been developed as a Android-application using the tools and resources mentioned above in the introduction software part. After completing the project now every student and instructor can manage their office hours Easily, online and anytime. This will save time both and help them to manage their own time.

1.2 Software Development

In this part we will discuss the tools that help us to build our project.

1.2.1 Android Studio

Figure.1.2.1: Android Studio

Android Studio (Figure.1.2.5) is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA.On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:
A flexible Gradle-based build system
A fast and feature-rich emulator
A unified environment where you can develop for all Android devices
Apply Changes to push code and resource changes to your running app

1.2.2 Java

Figure.1.2.2: Java

Java (Figure.1.2.2) is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It is intended to let application developers writeonce, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.

1.2.3 XML

Figure.1.2.3: XML

Extensible Markup Language (XML) (Figure.1.2.3) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The World Wide Web Consortium's XML 1.0 Specification of 1998 and several other related specifications.

1.2.4 Firebase

Figure.1.2.4: Firebase

Firebase (Figure.1.2.4) is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014. As of October 2018, the Firebase platform has 18 products, which are used by 1.5 million apps.

1.2.5 ERD Plus

Figure.1.2.5: ERD+

(Figure.1.2.5) A database modeling tool for creating Entity Relationship Diagrams, Relational Schemas, Star Schemas, and SQL DDL statements.

ERDPlus enables drawing standard ERD components

- Entities
- Attributes
- Relationships
- The notation supports drawing regular and weak entities, various types of attributes (regular, unique, multi-valued, derived, composite, and optional), and all possible cardinality constraints of relationships (mandatory-many, optional-many, mandatory-one and optional-one). (9)

We have used ERD+ As well to create the initial ERD And Database

1.2.6 Lucid Chart

Figure.1.2.6: Lucid Chart

Lucid chart (Figure.1.2.6) is a web-based proprietary platform that is used to allow usersto collaborate on drawing, revising and sharing charts and diagrams.

1.2.7 Git Hub

Figure.1.2.7: GitHub

Projects on GitHub (Figure.1.2.7) can be accessed and manipulated using the standard Git command-line interface and all of the standard Git commands work with it. GitHub also allows registered and unregistered users to browse public repositories on the site. Multiple desktop clients and Git plugins have also been created by GitHub and other third parties that integrate with the platform.

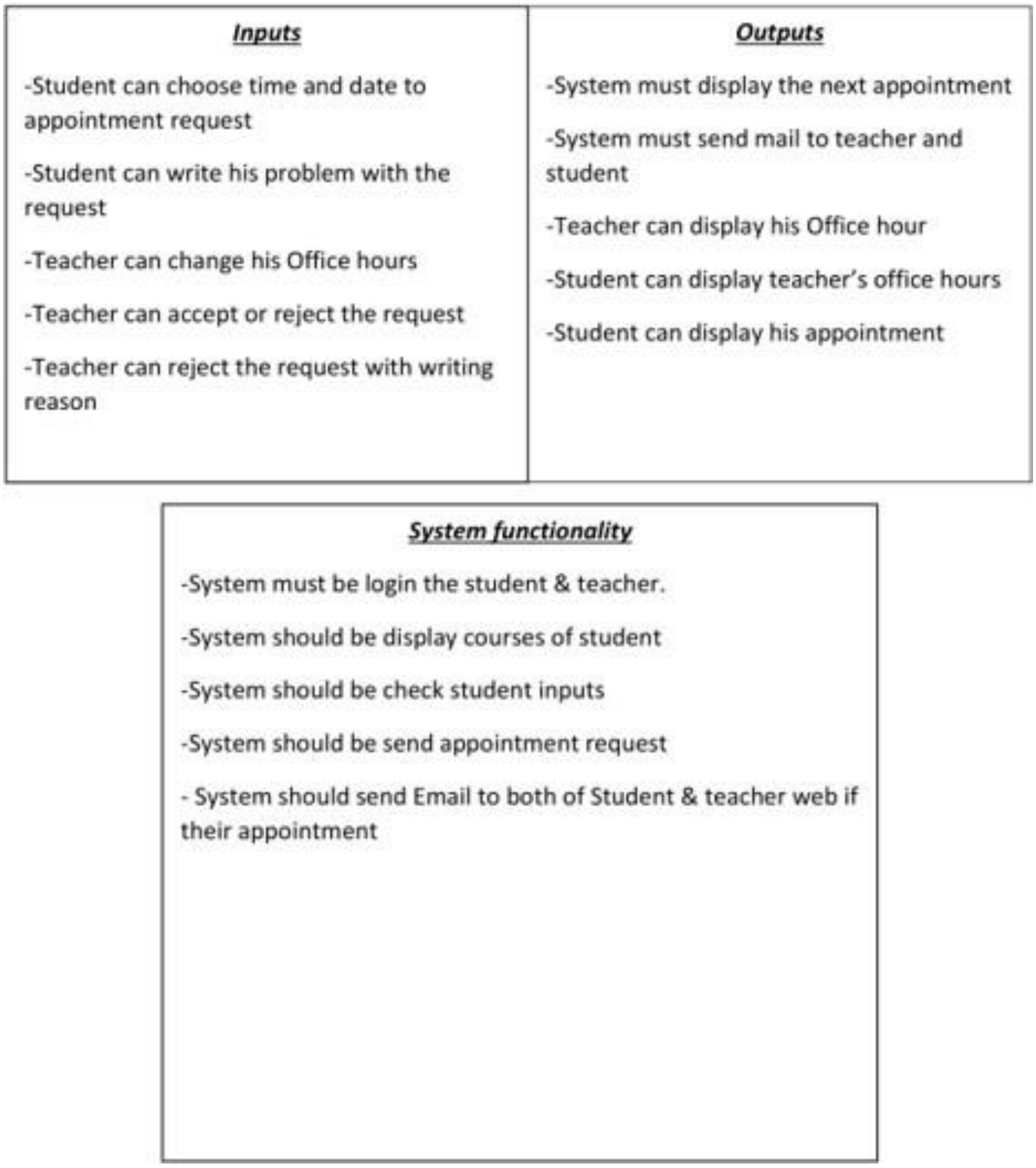
The site provides social networking-like functions such as feeds, followers, wikis (using wiki software called Gollum) and a social network graph to display how developers work on their versions ("forks") of a repository and what fork (and branch within that fork) is newest.

CHAPTER 2

Analysis and System Requirements

In this chapter we will describe the UML diagrams which can help us to describe main our system.

- **System requirements**



Inputs

-Student can choose time and date to appointment request

-Student can write his problem with the request

-Teacher can change his Office hours

-Teacher can accept or reject the request

-Teacher can reject the request with writing reason

Outputs

-System must display the next appointment

-System must send mail to teacher and student

-Teacher can display his Office hour

-Student can display teacher's office hours

-Student can display his appointment

System functionality

-System must be login the student & teacher.

-System should be display courses of student

-System should be check student inputs

-System should be send appointment request

- System should send Email to both of Student & teacher web if their appointment

Figure.2.1: System Requirements

Entity Relationship diagram

Presented below is the Entity Relationship Diagram (ERD) detailing the entities (tables) and their attributes across the entire system. The system encompasses entities such as student, teacher, request, department, section, and course, providing a comprehensive overview of the entire system structure.

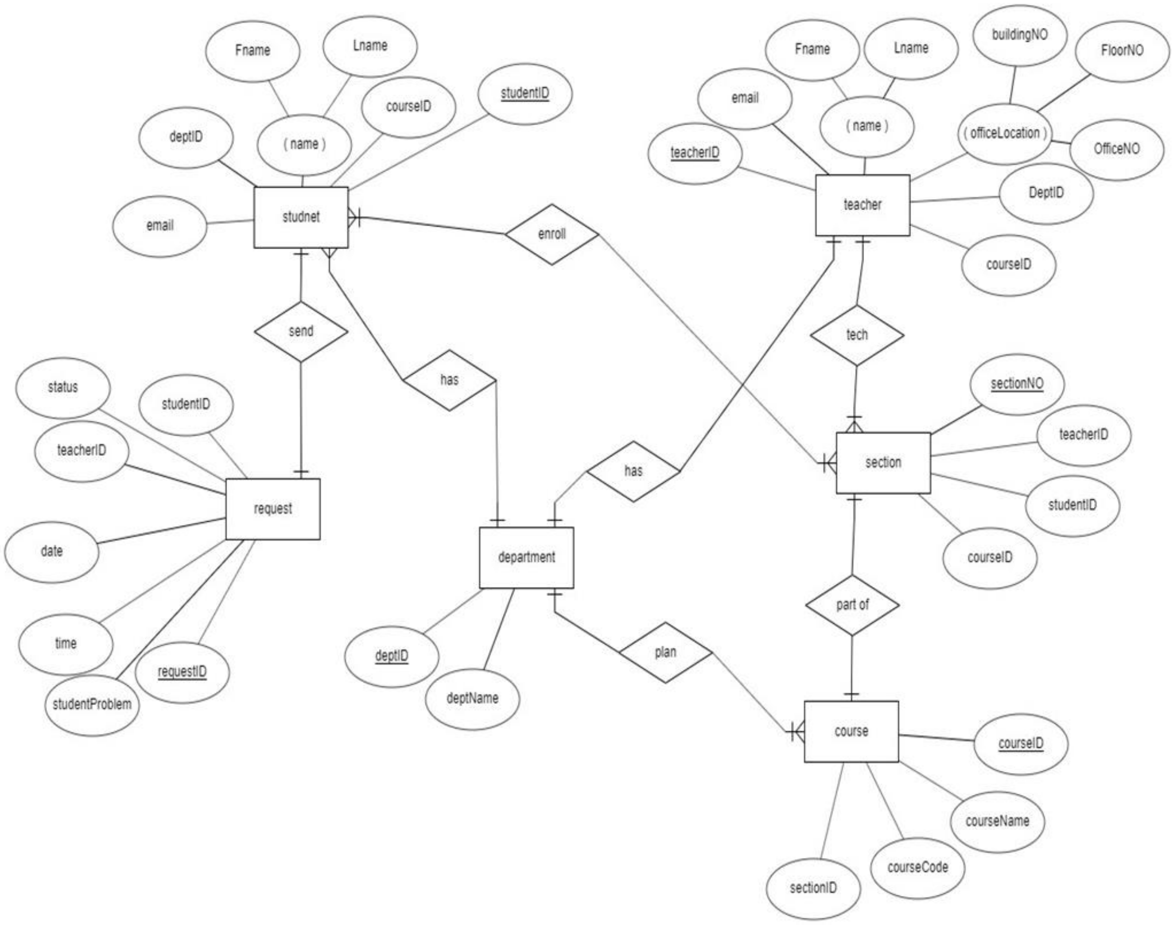


Figure 2.2 Entity Relationship diagram

Relational Schema

The schema, derived from the Entity Relationship Diagram (ERD), encapsulates the organized structure of the entire system. This model meticulously represents entities, including student, teacher, request, department, section, and course, along with their respective attributes. The schema provides a clear and detailed description of the data relationships and design implemented within the system.

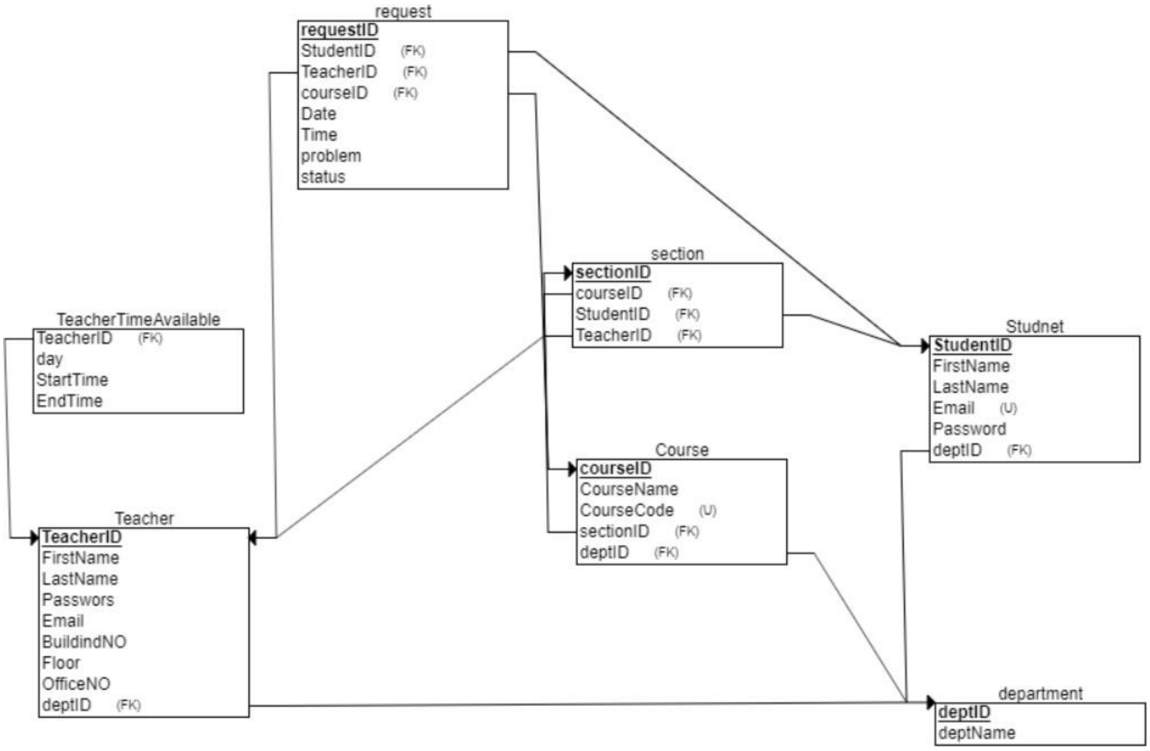


Figure 2.3 Relational schema

⍰ Process Modeling

- **Abstraction level data Flow Diagram:**

In the Context Level Data Flow Diagram, the symbolic '0' serves as a more abstract representation, encompassing both teachers and students and illustrating the fluidity of data between them. The process initiates with a student logging into the system, followed by actions such as checking courses, reviewing appointment information, and sending appointment requests to teachers. From the teacher's perspective, they receive notifications indicating incoming requests from students. Teachers have the capability to confirm these requests, initiating a feedback loop back to the respective student.

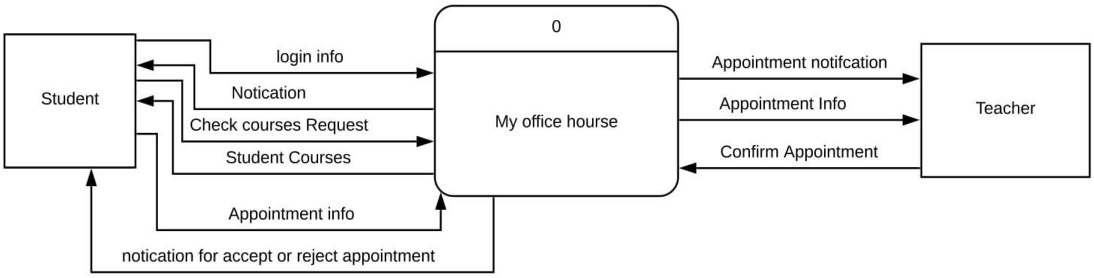


Figure 2.3 Context level 0 diagram

- **Detailed level 0 data flow diagram:**

  In this context, we saw the functionality with more details by describing the flow of the data in the system by executing different functions.

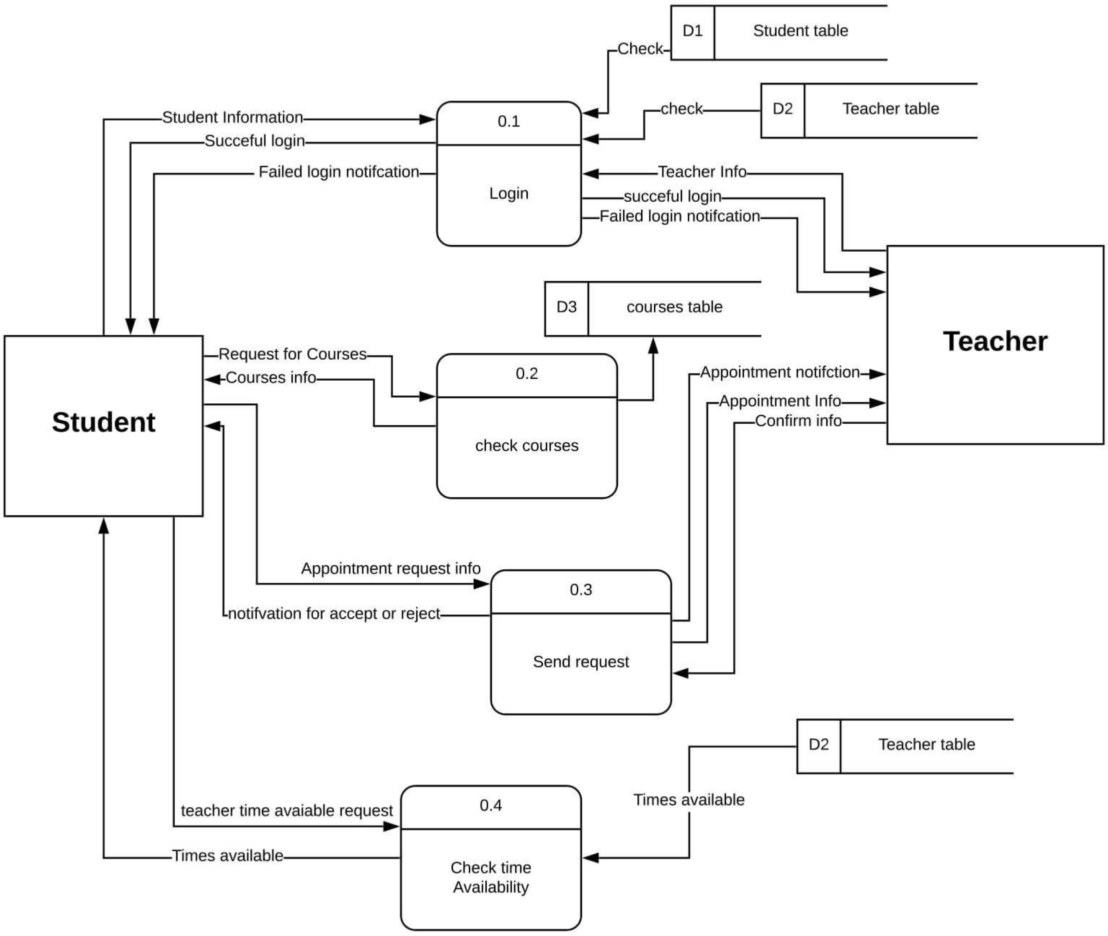


Figure.2.5: Level Zero Diagram

Level N Data Flow Diagram

The 'Send Request' functionality begins with a student initiating the process. Upon logging into the system, the student navigates to the appropriate interface and executes the 'Send Request' action. This action triggers the transmission of a notification to the designated teacher, indicating the incoming request.

From the teacher's perspective, upon receiving the notification, they are prompted to review and respond to the request. The teacher has the ability to confirm or reject the request, and upon confirmation, a corresponding notification is relayed back to the initiating student.

This seamless flow ensures effective communication and coordination between students and teachers, streamlining the request process within the system.

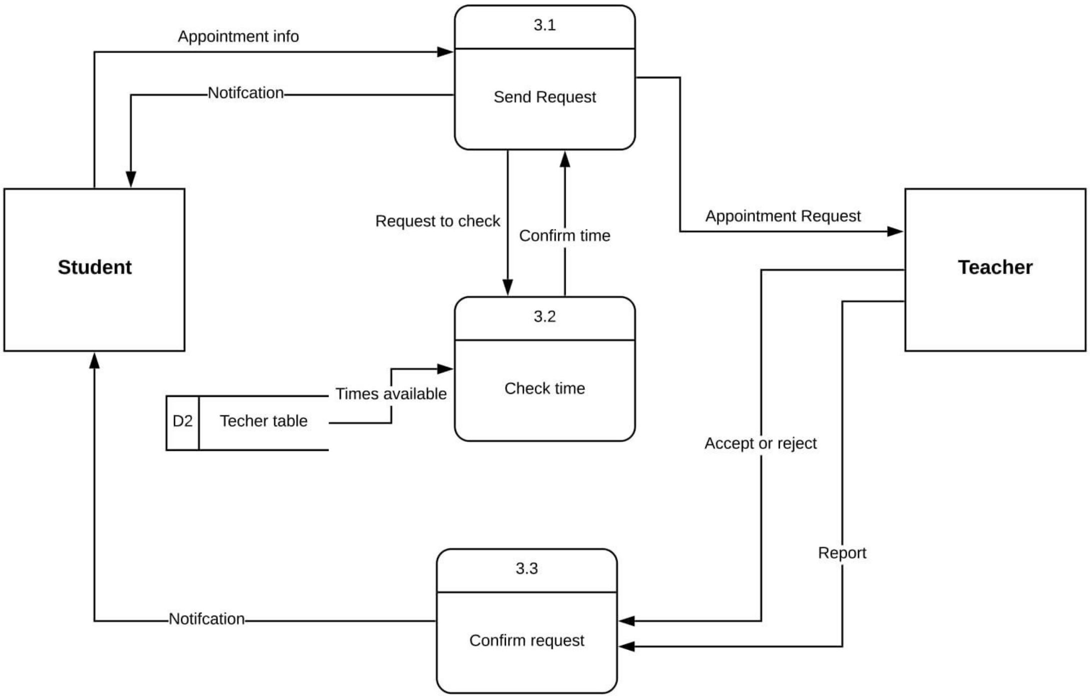


Figure 2.5 Level 3 Diagram

CHAPTER 3

Design

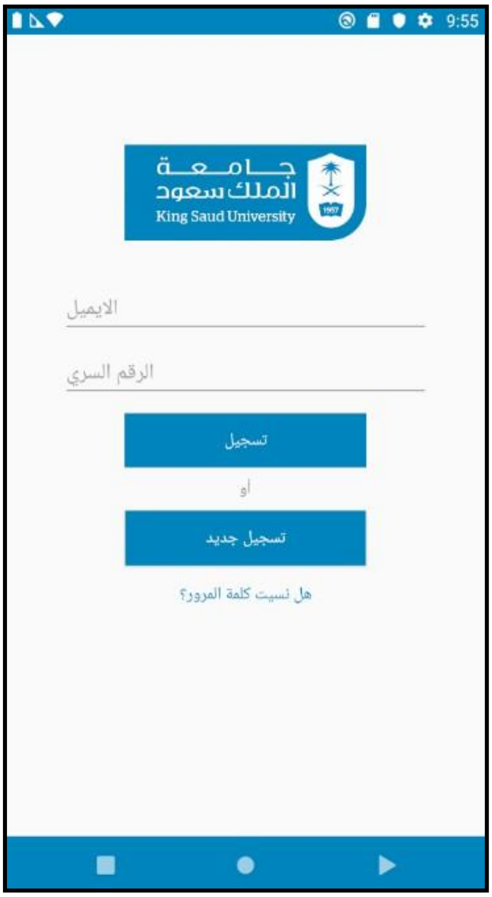
This chapter is bifurcated into two integral sections, each pivotal to the holistic understanding and development of our software system. The initial segment will spotlight the various screens within the application, accompanied by concise yet insightful descriptions. This visual exploration aims to familiarize the reader with the user interface, shedding light on the key components such as the login screen, course checking interface, appointment information display, and more.

Subsequently, the second part of this chapter is dedicated to delving into the intricate realm of database design utilizing Firebase. We will explore the intricacies of structuring collections, establishing relationships, and harnessing Firebase's real-time features. This section is an indispensable guide for creating a robust, scalable, and real-time synchronized database, ensuring the seamless flow of data within the application.

3.1 Forms Design for student

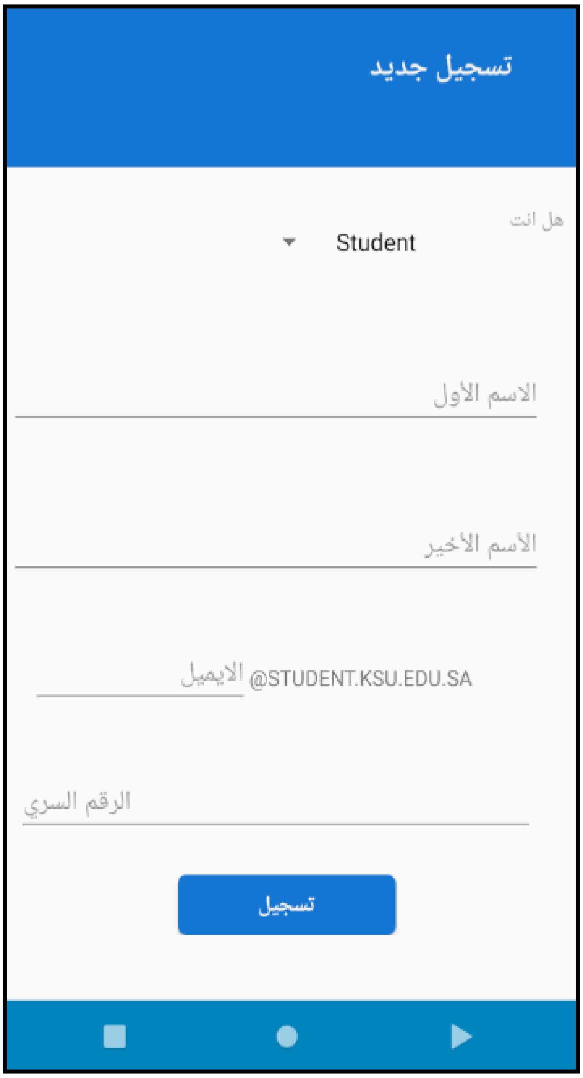
3.1.1 Login page

This is login screen user can register or login to the system.


9:55
جامعة
الملك سعود
King Saud University
الايميل
الرقم السري
تسجيل
او
تسجيل جديد
هل نسيت كلمة المرور؟

3.1.2 Register Page

When a user intends to register on the system, they are prompted to specify their role by choosing between two distinct options: 'Student' or 'Instructor.' This initial step allows for a tailored onboarding experience, ensuring that each user receives access and functionalities relevant to their role within the system. The chosen role not only defines the user's interface but also shapes their interactions and permissions, contributing to a more personalized and efficient user experience.



تسجيل جديد

هل انت: Student

3.1.3 Student’s home page

After successfully logging into the system, students are directed to a dynamic and comprehensive home page. This central hub provides access to a multitude of essential functionalities tailored to enhance the student experience:

Send Request:

A dedicated feature enabling students to initiate appointment requests with instructors. This functionality streamlines communication and scheduling within the system.

View Courses:

A section offering a detailed overview of available courses. Students can explore course details, schedules, and any additional information related to their academic pursuits.

Calendar:

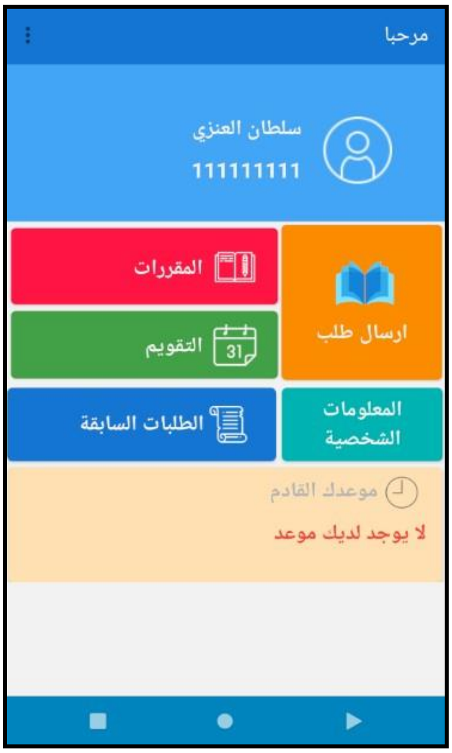
An interactive calendar interface allowing students to manage and visualize their schedules. This feature aids in planning, organizing, and staying on top of academic and appointment commitments.

Recent Requests:

A dedicated space showcasing the most recent appointment requests made by the student. This provides a quick reference to the status and details of ongoing communication.

Status for the Next Appointment:

Real-time updates on the status of the upcoming appointment. This feature keeps students informed about confirmed appointments and any changes in schedule.



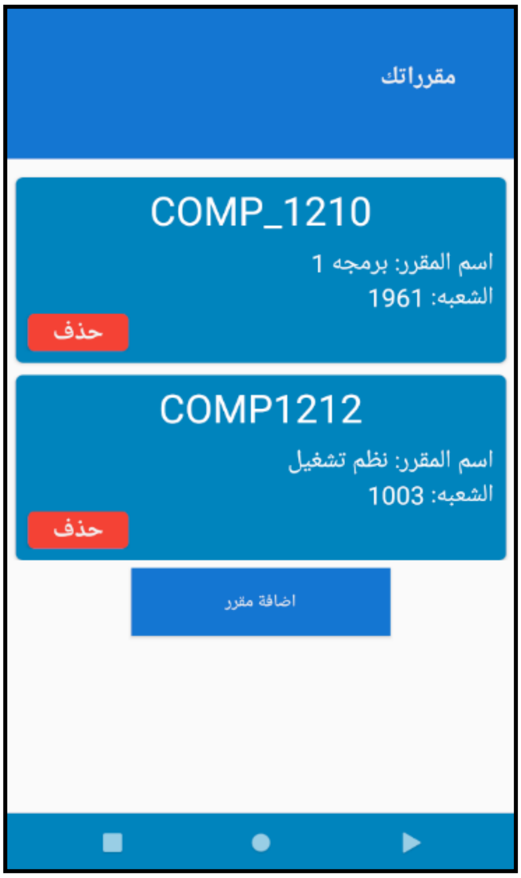
سلطان العنزي
111111111

موعدك القادم
لا يوجد لديك موعد

Figure 1 Student home

3.1.4 Courses page

When a student clicks on the 'View Courses' functionality, the system presents a curated display showcasing the courses the student is currently registered for. This feature not only provides a snapshot of the student's academic commitments but also introduces a seamless integration with the ability to send requests for office hours.



مقرراتك

COMP_1210

اسم المقرر: برمجه 1

الشعبه: 1961

COMP1212

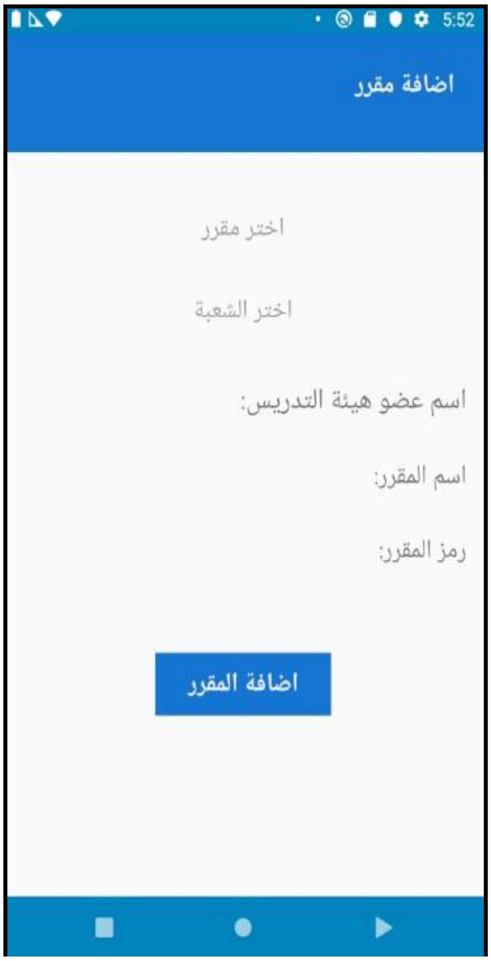
اسم المقرر: نظم تشغيل

الشعبه: 1003

Figure 2 Student courses

3.1.5 Add course page

Empowering students with flexibility in their academic journey, the 'Add Course' feature allows them to proactively enroll in courses of their choice. By navigating to the dedicated 'Add Course' page, students are presented with a straightforward interface where they can enter the specific course code and section number for the desired course.



5:52

اضافة مقرر

اختر مقرر

اختر الشعبة

اسم عضو هيئة التدريس:

اسم المقرر:

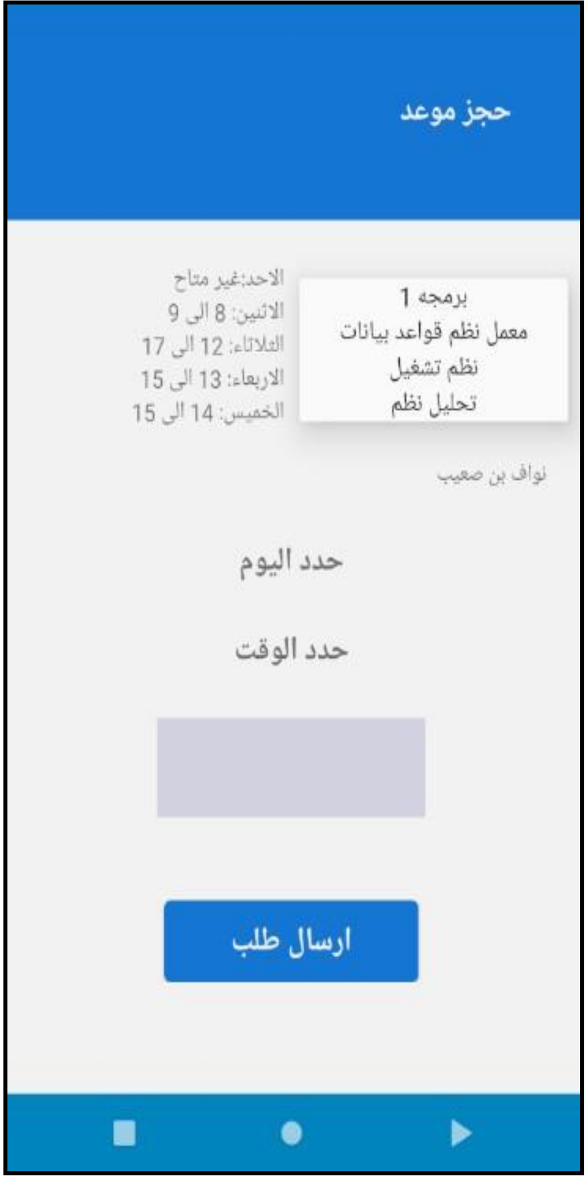
رمز المقرر:

اضافة المقرر

Figure 3 Add course for student

3.1.6 Send Request Page

Students can easily request to meet with their instructors during office hours using the system. By selecting the instructor and specifying their preferred time, students can quickly send a request. Instructors receive these requests, review the details, and respond accordingly. This straightforward process facilitates clear communication and helps students get the support they need.



حجز موعد

برمجه 1
معمل نظم قواعد بيانات
نظم تشغيل
تحليل نظم

الاحد:غير متاح
الاثنين: 8 الى 9
الثلاثاء: 12 الى 17
الاربعاء: 13 الى 15
الخميس: 14 الى 15

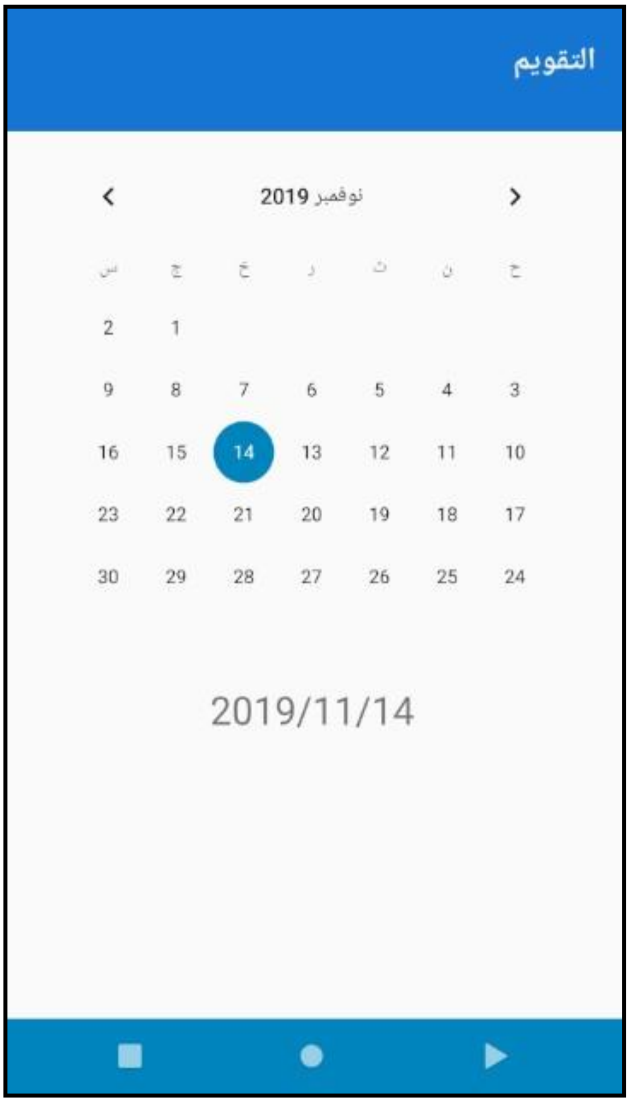
نواف بن صعيب

حدد اليوم

حدد الوقت

ارسال طلب

3.1.7 Calendar Page



التقويم

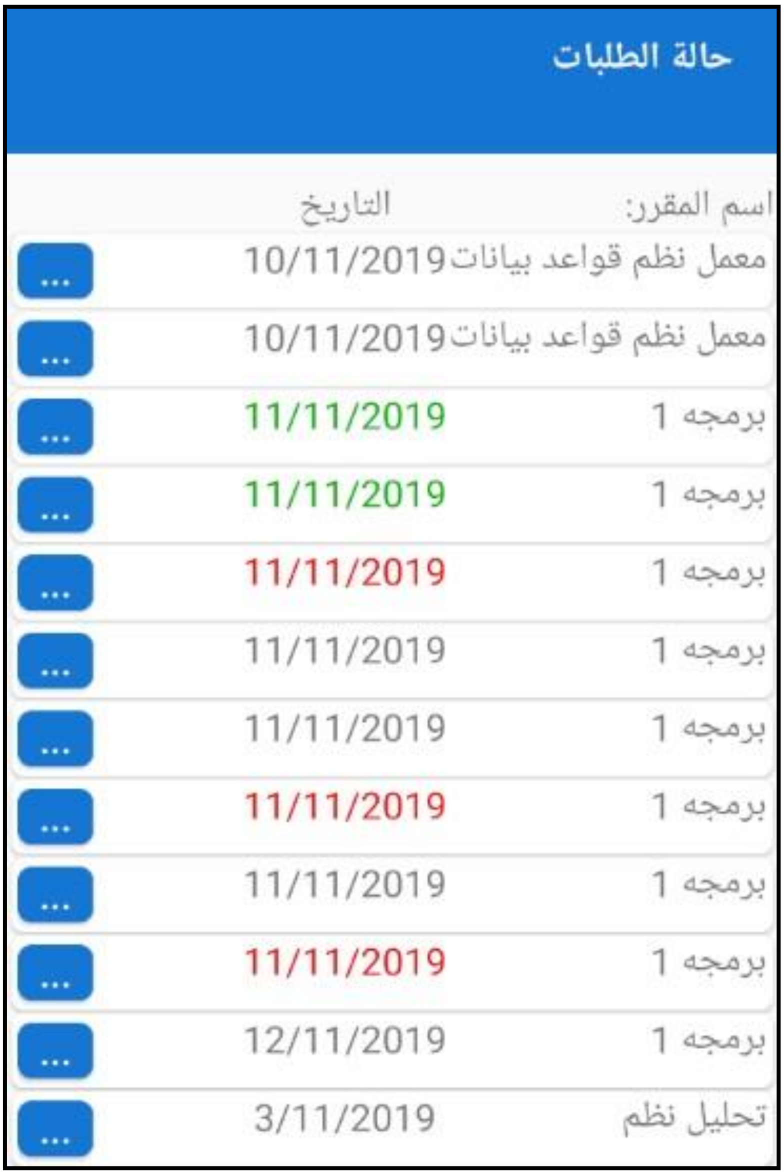
نوفمبر 2019

س	ج	خ	ر	ث	ن	ح
2	1					
9	8	7	6	5	4	3
16	15	14	13	12	11	10
23	22	21	20	19	18	17
30	29	28	27	26	25	24

2019/11/14

3.1.8 Student’s requests history Page

Students can easily view their past appointment requests using the 'Request History' feature. It shows a simple list with dates and statuses, giving students a quick overview of their past interactions with instructors.

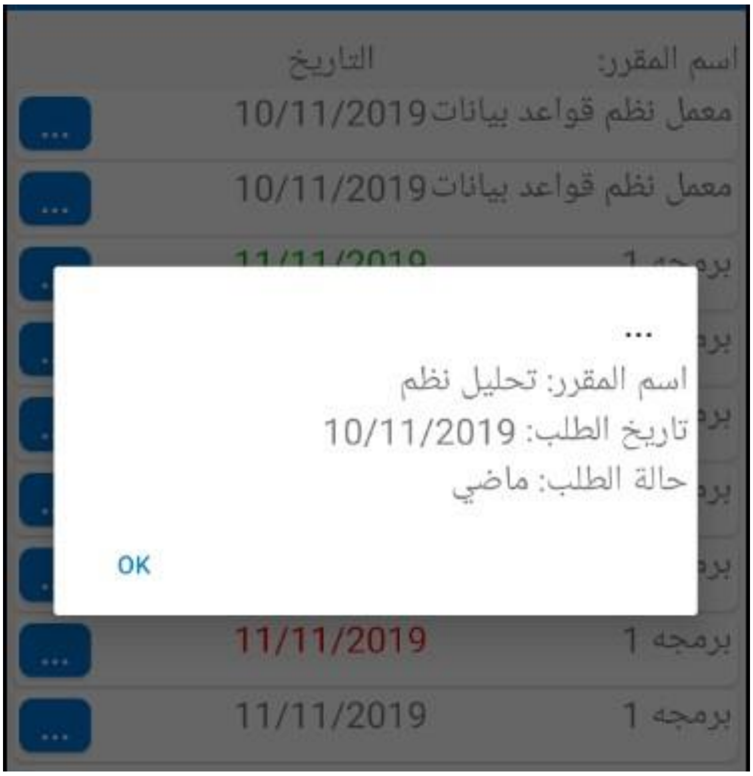


حالة الطلبات

اسم المقرر:	التاريخ	
معمل نظم قواعد بيانات	10/11/2019	...
معمل نظم قواعد بيانات	10/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	11/11/2019	...
برمجه 1	12/11/2019	...
تحليل نظم	3/11/2019	...

3.1.9 More info about request history

When student clicks to "…" this Pop - up window will display.



اسم المقرر: التاريخ

معمل نظم قواعد بيانات 10/11/2019

معمل نظم قواعد بيانات 10/11/2019

برمجه 1 11/11/2019

...

اسم المقرر: تحليل نظم

تاريخ الطلب: 10/11/2019

حالة الطلب: ماضي

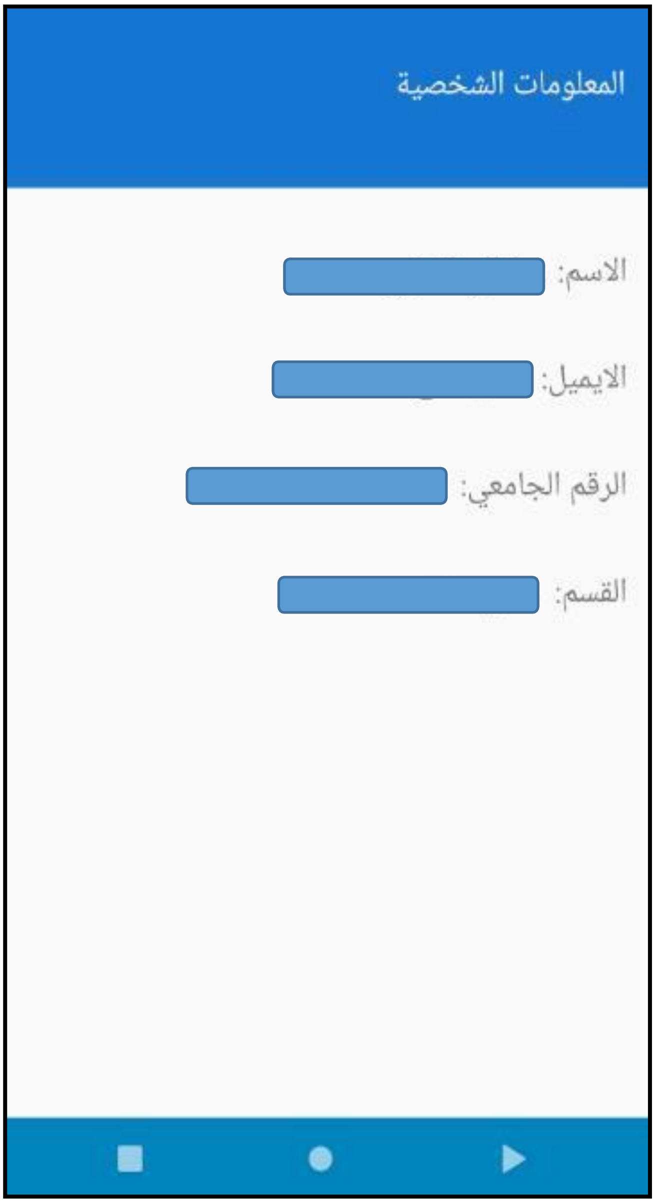
OK

برمجه 1 11/11/2019

برمجه 1 11/11/2019

3.1.10 Profile

The profile page shows important details about the student, like their full name, student ID, and contact information. It's a quick and easy way to check key information



المعلومات الشخصية

الاسم: [redacted]

الايميل: [redacted]

الرقم الجامعي: [redacted]

القسم: [redacted]

3.2 instructor section

3.2.1 Instructor’s home page

The instructor's home page provides a range of functionalities tailored to streamline their interactions within the system:

Received Requests:

A dedicated section displaying incoming appointment requests from students. Instructors can efficiently manage and respond to these requests.

Office Hours:

An interface allowing instructors to set and communicate their office hours. This feature ensures clarity and accessibility for students seeking additional support.

Requests Status:

A concise overview of the status of appointment requests initiated by instructors. This section aids in tracking and managing ongoing interactions.

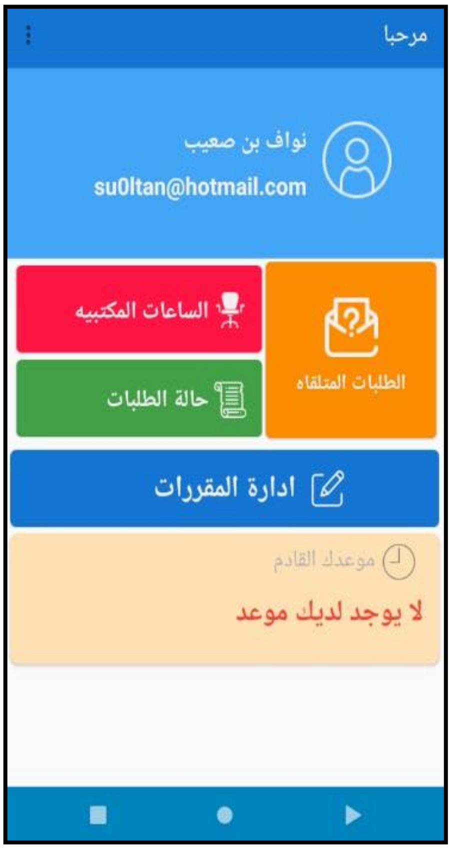
Manage Courses:

A functionality enabling instructors to oversee and manage the courses they are responsible for. This includes access to course details, schedules, and related information.

Next Appointments:

Real-time updates on the upcoming scheduled appointments, ensuring instructors are well-prepared and organized.

This home page serves as a central hub for instructors, offering quick access to essential tools and insights to facilitate effective communication and course management within the system.



مرحبا

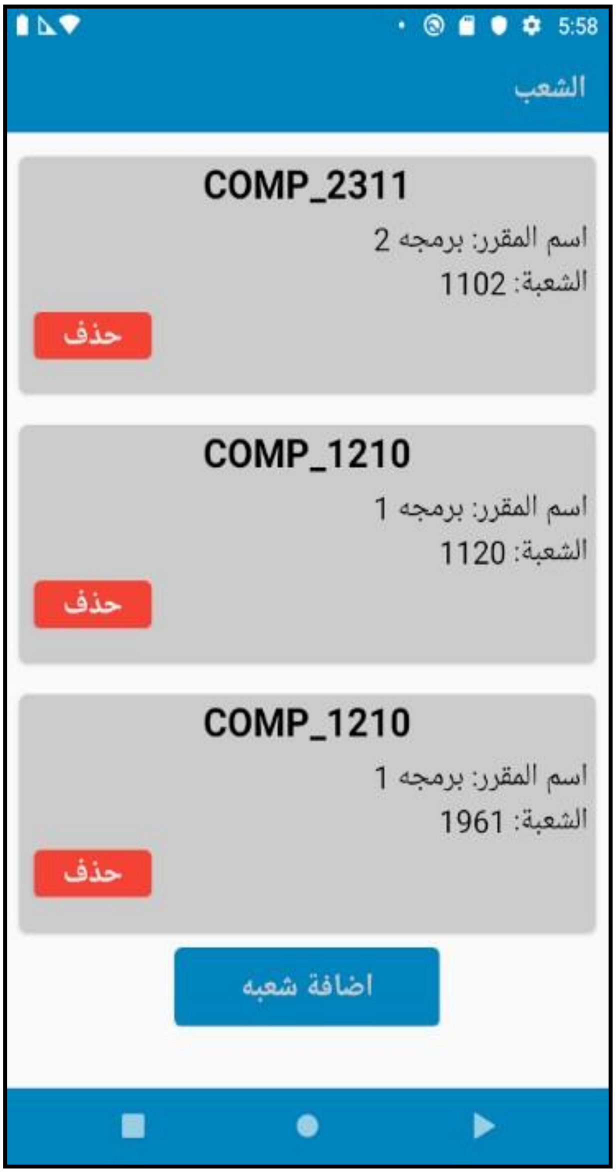
نواف بن صعيب

su0ltan@hotmail.com

لا يوجد لديك موعد

3.2.2 Instructor sections

Instructors can easily handle their course sections using the 'Manage Sections' feature. It lets them view, update, and organize section details for smoother course administration. Also it shows add section button which able the instructor to add the sections easly.



COMP_2311

اسم المقرر: برمجه 2

الشعبة: 1102

COMP_1210

اسم المقرر: برمجه 1

الشعبة: 1120

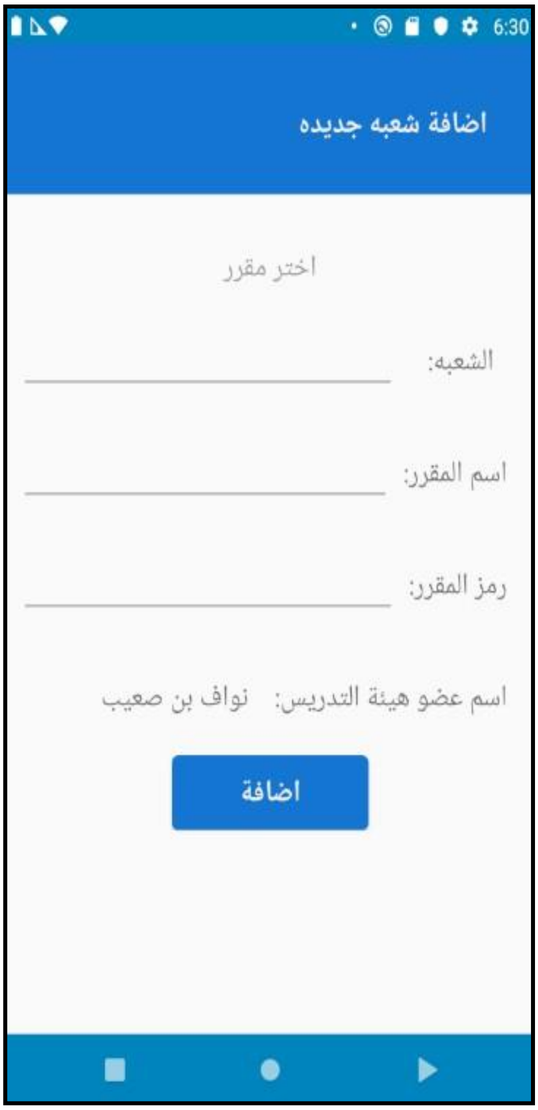
COMP_1210

اسم المقرر: برمجه 1

الشعبة: 1961

3.2.3 Add section page.

Instructors have the ability to add their course sections using the 'Add Section' feature. This allows them to easily include new sections, ensuring flexibility and efficient management of their courses



6:30

اضافة شعبه جديده

اختر مقرر

الشعبه:

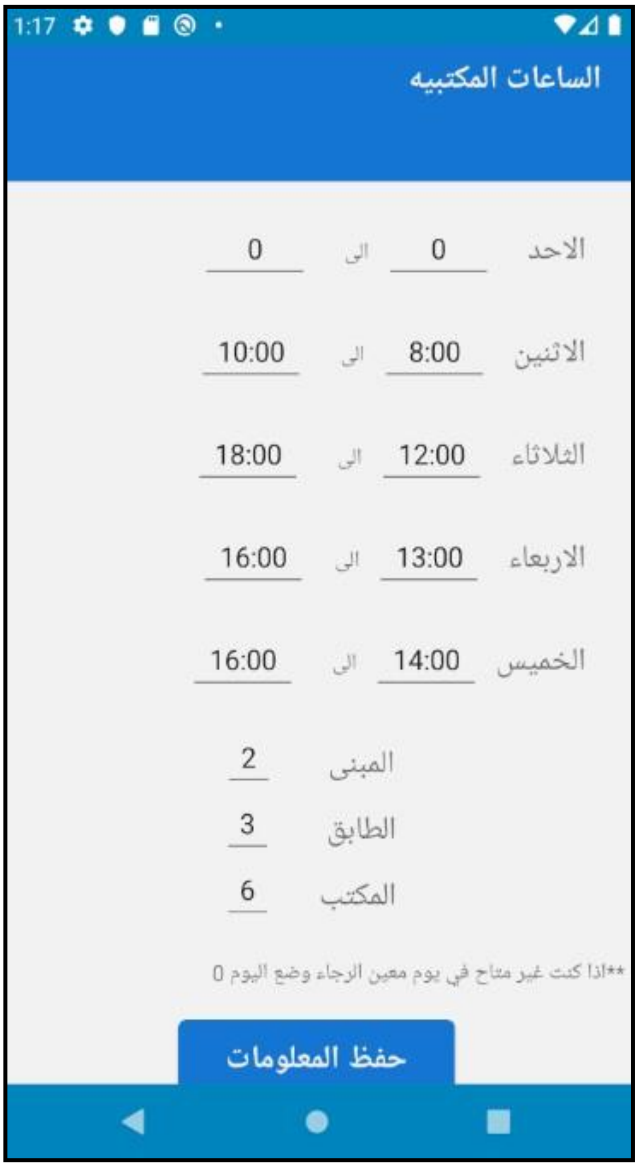
اسم المقرر:

رمز المقرر:

اسم عضو هيئة التدريس: نواف بن صعيب

3.2.4 Office hours page

Instructors can effortlessly manage their office hours and location through the system. Using the 'Manage Office Hours' feature, instructors can specify their available times and the location of their office, ensuring clarity for students seeking assistance or appointments.



الاحد 0 الى 0

الاثنين 8:00 الى 10:00

الثلاثاء 12:00 الى 18:00

الاربعاء 13:00 الى 16:00

الخميس 14:00 الى 16:00

المبنى 2

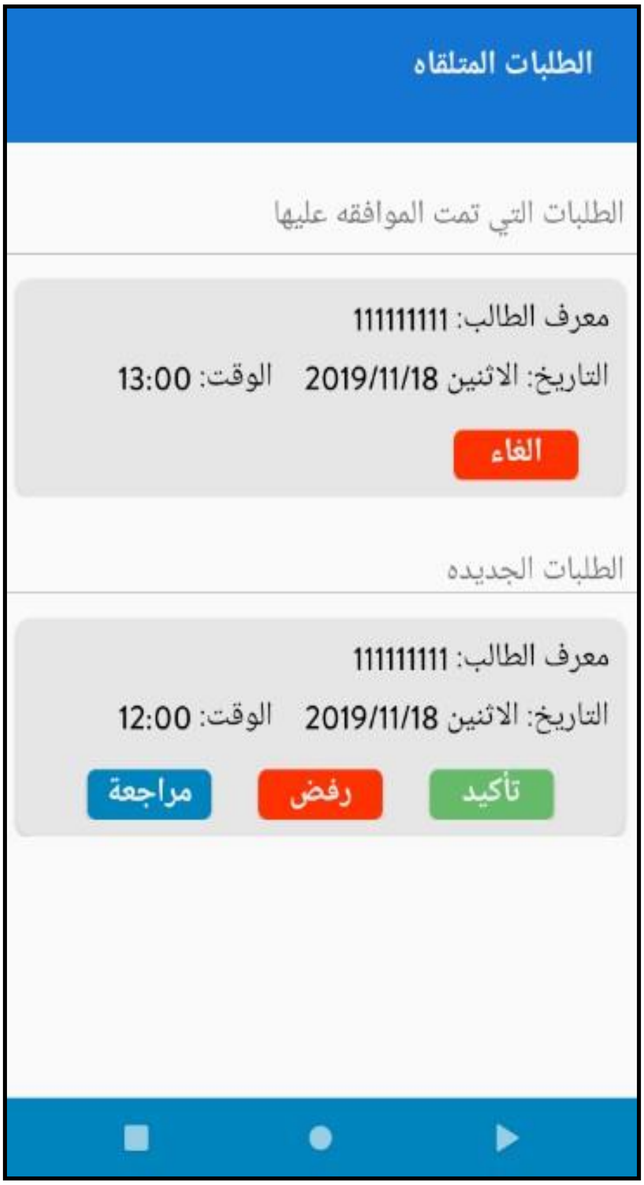
الطابق 3

المكتب 6

**اذا كنت غير متاح في يوم معين الرجاء وضع اليوم 0

3.2.5 Managing requests Page.

Instructors can efficiently handle student requests by using the 'Manage Requests' feature. This allows instructors to review student concerns and choose to accept, reject, or cancel the requests, providing a streamlined process for addressing and managing student inquiries.



الطلبات المتلقاه

الطلبات التي تمت الموافقه عليها

معرف الطالب: 111111111

التاريخ: الاثنين 2019/11/18   الوقت: 13:00

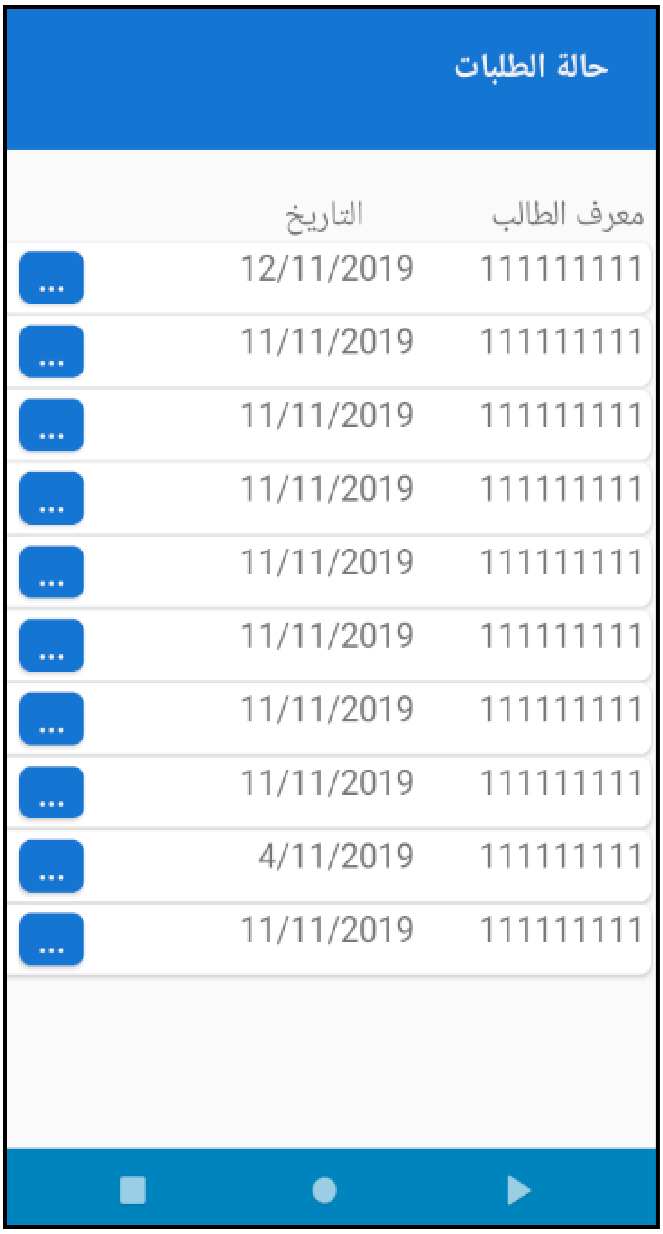
الطلبات الجديده

معرف الطالب: 111111111

التاريخ: الاثنين 2019/11/18   الوقت: 12:00

3.2.6 Instructor requests history Page

Instructors can easily review the history of student requests using the 'Request History' feature. This provides a quick overview of past student interactions, helping instructors stay informed and organized



حالة الطلبات

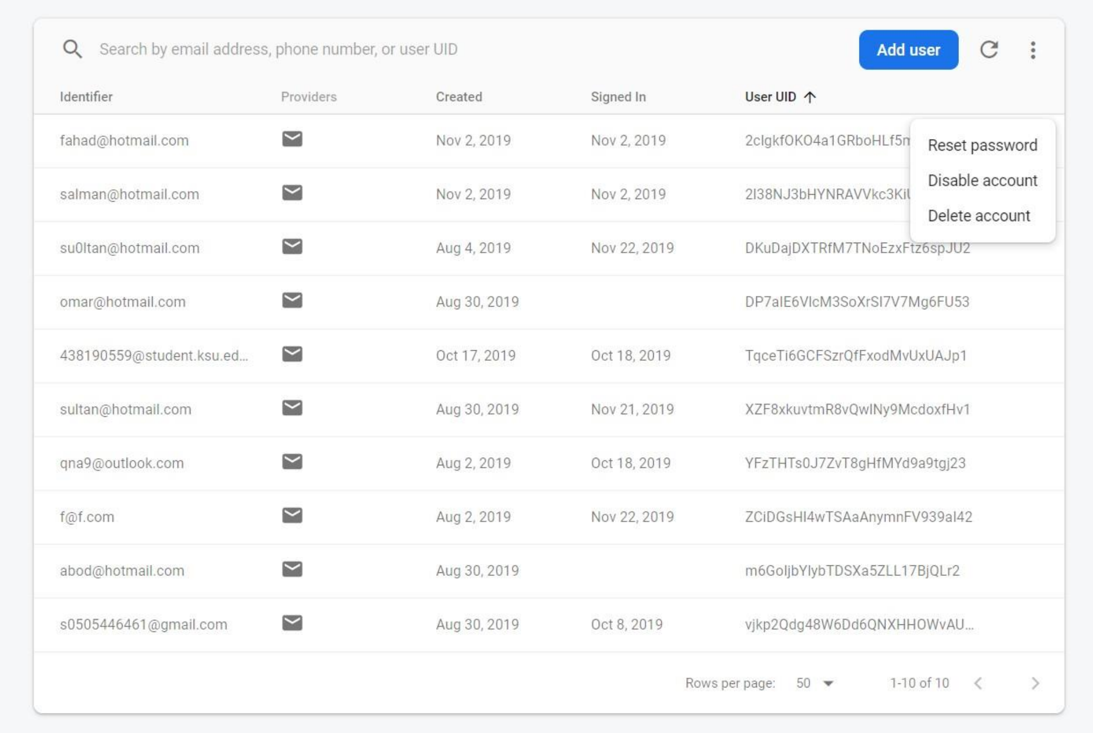
	التاريخ	معرف الطالب
...	12/11/2019	111111111
...	11/11/2019	111111111
...	11/11/2019	111111111
...	11/11/2019	111111111
...	11/11/2019	111111111
...	11/11/2019	111111111
...	11/11/2019	111111111
...	11/11/2019	111111111
...	4/11/2019	111111111
...	11/11/2019	111111111

3.3 Database Design

In this part, we'll talk about the database we used for our project. The database is like the brain of our system, storing and organizing all the important information. We'll discuss how we structured it, the connections between different pieces of data, and why we chose this particular database. Understanding our database is key to how our system works, and this section will give you a clear picture of its role in supporting the features of our project.

3.3.1 Authentication users page

The 'Authentication' table in Firebase is a central repository housing essential user data, including email addresses, unique user identifiers (UIDs), and encrypted passwords. This table is pivotal for verifying user identities and managing secure access, forming a key component of our project's authentication system within the Firebase database.



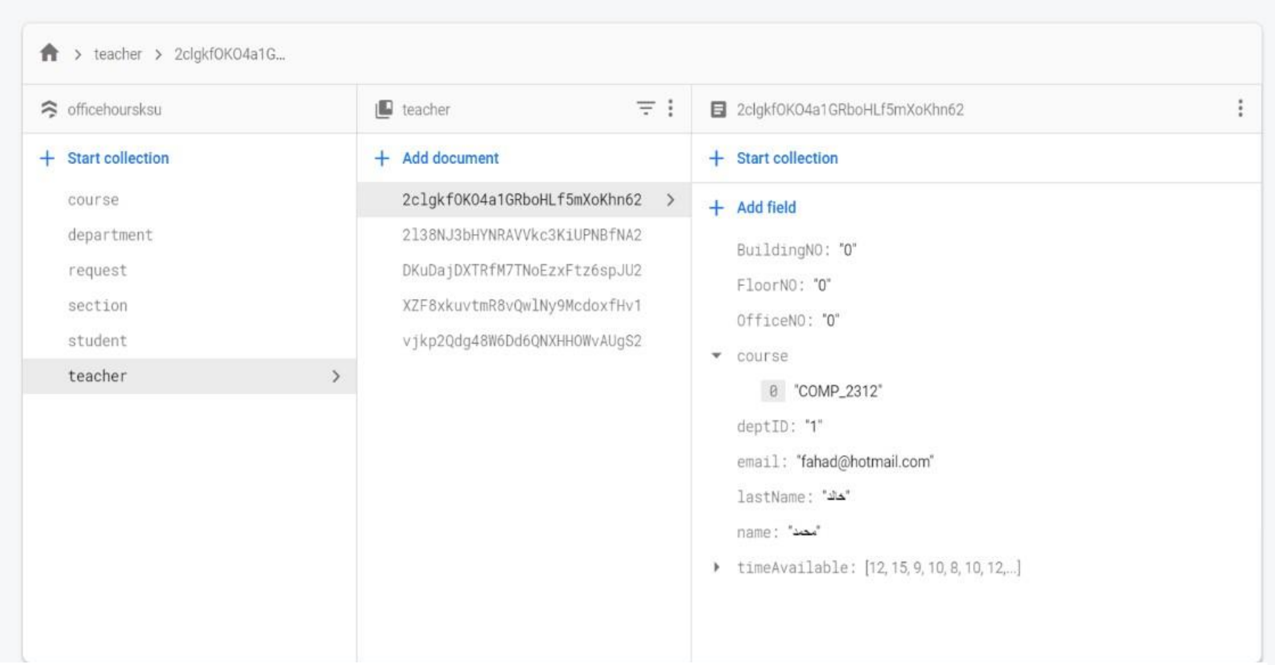
Identifier	Providers	Created	Signed In	User UID ↑
fahad@hotmail.com	✉	Nov 2, 2019	Nov 2, 2019	2clgkfOKO4a1GRboHLf5n
salman@hotmail.com	✉	Nov 2, 2019	Nov 2, 2019	2I38NJ3bHYNRAVVkc3Ki
su0ltan@hotmail.com	✉	Aug 4, 2019	Nov 22, 2019	DKuDajDXTRfM7TNoEzxFtz6spJU2
omar@hotmail.com	✉	Aug 30, 2019		DP7alE6VlcM3SoXrSI7V7Mg6FU53
438190559@student.ksu.ed...	✉	Oct 17, 2019	Oct 18, 2019	TqceTi6GCFSzrQfFxodMvUxUAJp1
sultan@hotmail.com	✉	Aug 30, 2019	Nov 21, 2019	XZF8xkuvtmR8vQwINy9McdoxfHv1
qna9@outlook.com	✉	Aug 2, 2019	Oct 18, 2019	YFzTHTs0J7ZvT8gHfMYd9a9tgj23
f@f.com	✉	Aug 2, 2019	Nov 22, 2019	ZCiDGsHI4wTSAaAnymnFV939al42
abod@hotmail.com	✉	Aug 30, 2019		m6GoljbYlybTDSXa5ZLL17BjQLr2
s0505446461@gmail.com	✉	Aug 30, 2019	Oct 8, 2019	vjkp2Qdg48W6Dd6QNXHHOWvAU...

Rows per page: 50 1-10 of 10

Figure 4 Authentication users page

3.3.2 Teacher table (collection)

The 'Teacher's Collection' is a section in our database that stores comprehensive information about each instructor. This collection serves as a centralized repository, encompassing details such as their name, contact information, and any other pertinent data related to their role within the system. It provides a convenient and organized way to manage and access all relevant information about our instructors. It’s containing teacher’s office info, courses IDs, department, email, name, and office hours.



course
department
request
section
student
teacher

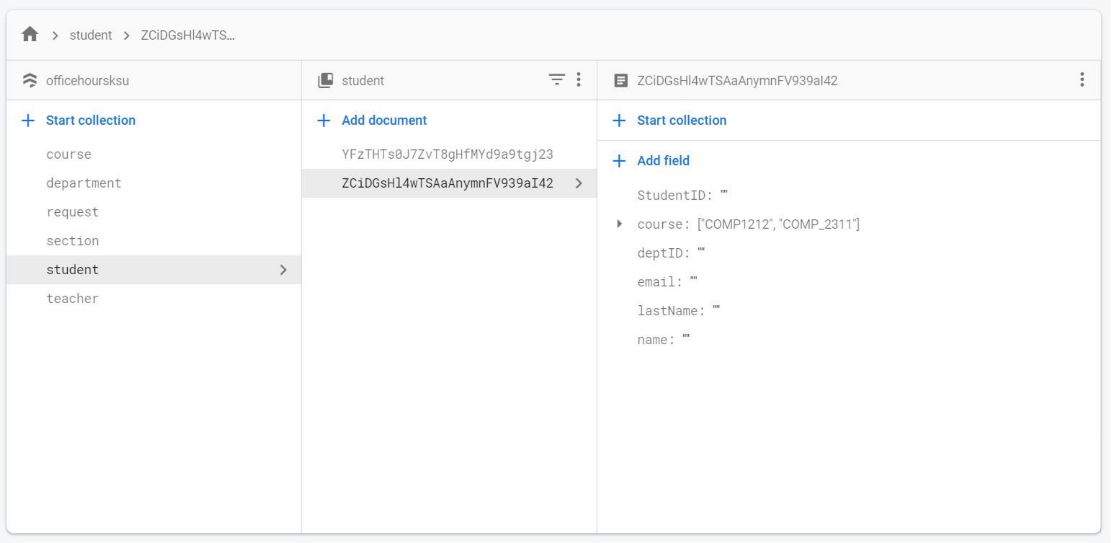
2clgkfOKO4a1GRboHLf5mXoKhn62
2138NJ3bHYNRAVVkc3KiUPNBfNA2
DKuDajDXTRfM7TNoEzxFtz6spJU2
XZF8xkuvtmR8vQwlNy9McdoxfHv1
vjkp2Qdg48W6Dd6QNXHHOWvAUgS2

```
BuildingNO: "0"
FloorNO: "0"
OfficeNO: "0"
▾ course
    0  "COMP_2312"
deptID: "1"
email: "fahad@hotmail.com"
lastName: "خالد"
name: "محمد"
▸ timeAvailable: [12, 15, 9, 10, 8, 10, 12,...]
```

Figure 5teacher collection(table) from firebase database

3.3.3 Student table(collection)

The 'Student's Collection' in our database holds a wealth of information about each student. This collection includes crucial details such as student ID, course IDs, department, email, and name. It serves as a comprehensive repository, organizing all essential student information, facilitating efficient management, and supporting various functionalities within our system.



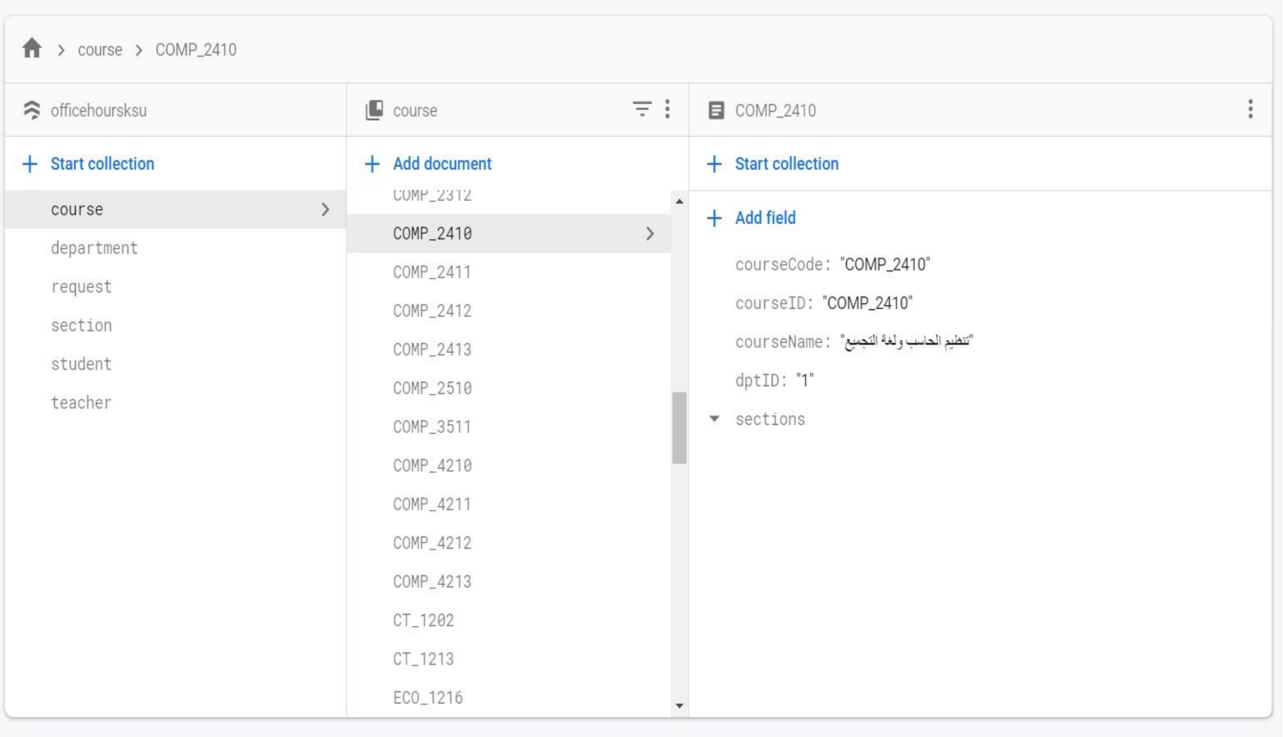
```
YFzTHTs0J7ZvT8gHfMYd9a9tgj23
ZCiDGsHl4wTSAaAnymnFV939aI42
```

```
StudentID: ""
course: ["COMP1212", "COMP_2311"]
deptID: ""
email: ""
lastName: ""
name: ""
```

Figure 6 student collection(table)from firebase database

3.3.4 Course table (collection)

The 'Course's Collection' is a repository in our database encompassing information about all courses in the college. This collection includes key details such as course code, course ID, course name, and arrays of sections. It serves as a central hub for managing and accessing comprehensive information about each course, supporting various functionalities within our system.



course
department
request
section
student
teacher

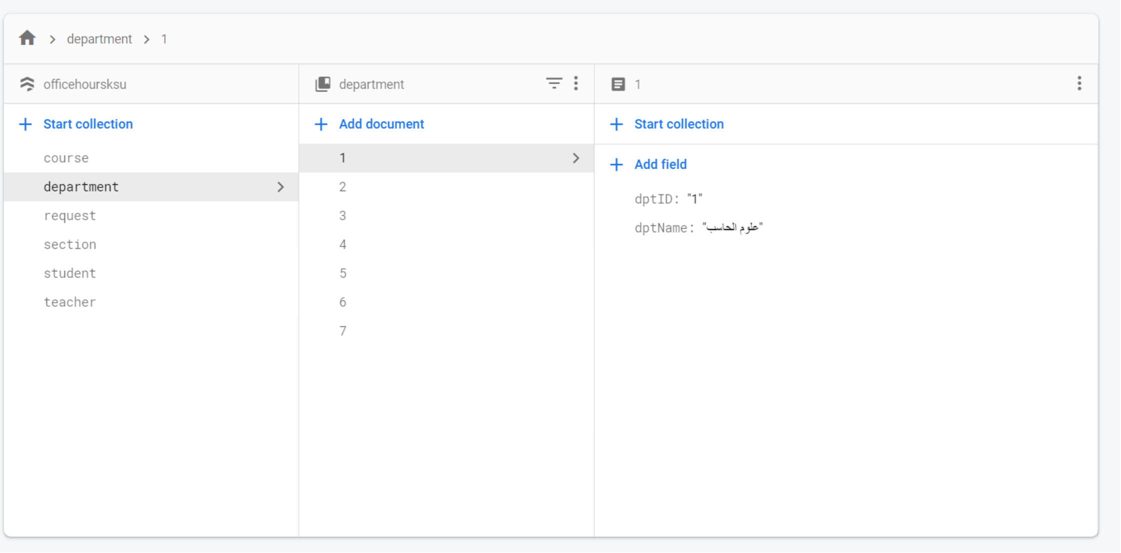
COMP_2312
COMP_2410
COMP_2411
COMP_2412
COMP_2413
COMP_2510
COMP_3511
COMP_4210
COMP_4211
COMP_4212
COMP_4213
CT_1202
CT_1213
ECO_1216

```
courseCode: "COMP_2410"
courseID: "COMP_2410"
courseName: "تنظيم الحاسب ولغة التجميع"
dptID: "1"
▾ sections
```

Figure 7 course collection(table)from firebase database

3.3.5 Department table (collection)

The 'Department's Collection' in our database is a hub that holds information about all departments in the college. It includes essential details such as department ID and name. This collection is a central resource for managing and organizing department-specific information, contributing to the overall structure and functionality of our system.



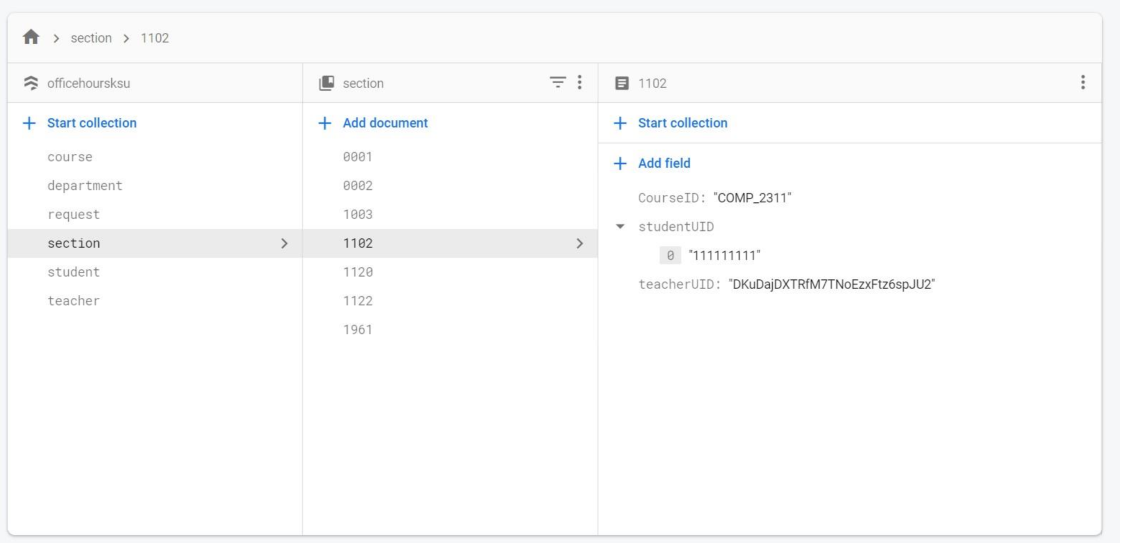
dptID: "1"

dptName: "علوم الحاسب"

Figure 8 department collection(table)from firebase database

3.3.6 Section table (collection)

The 'Section's Collection' in our database captures information about the sections that teachers add. It includes details such as course ID, an array of registered students, and the teacher ID. This collection serves as a dynamic record of course sections, facilitating effective management and tracking of student enrollment under each section, along with the corresponding teacher.



CourseID: "COMP_2311"

studentUID

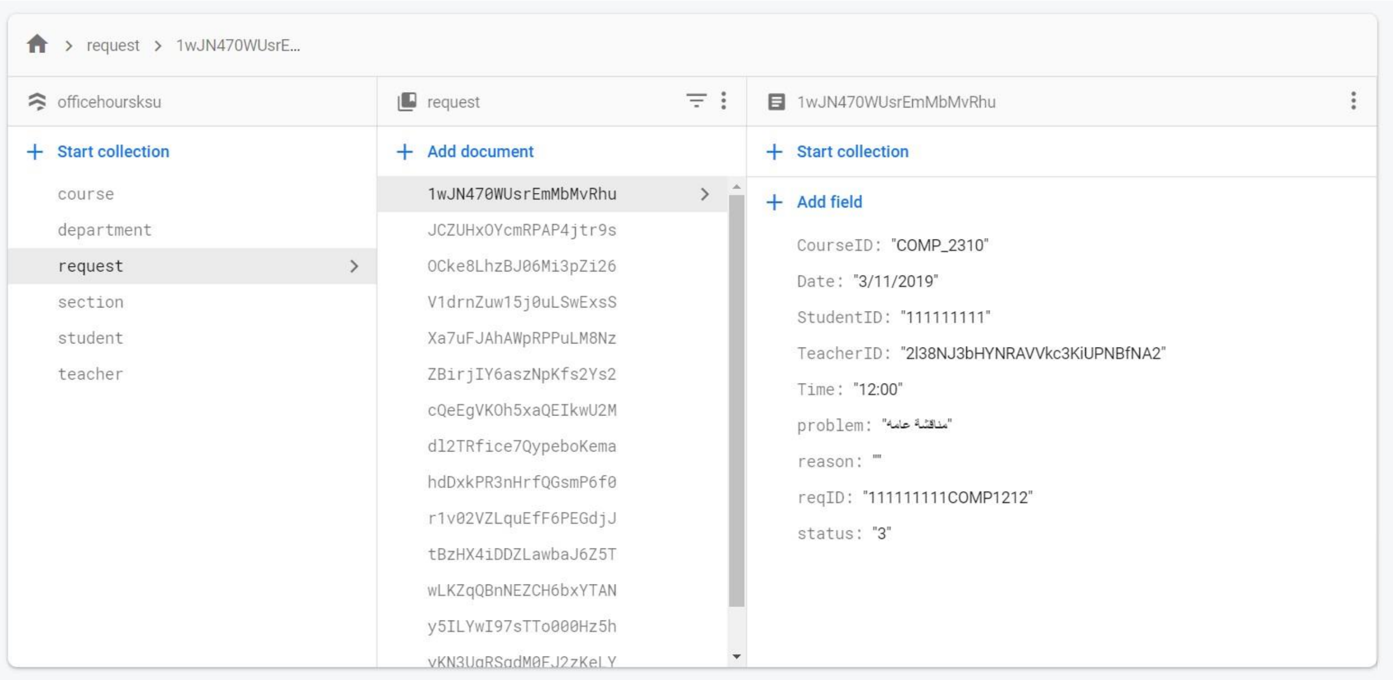
0 "111111111"

teacherUID: "DKuDajDXTRfM7TNoEzxFtz6spJU2"

Figure 9 section collection(table)from firebase database

3.3.7 request table (collection)

The 'Request's Collection' in our database holds information about requests initiated by students. It encompasses details such as course ID, date, student ID, teacher ID, time, problem reason, request ID, and request status. This collection acts as a comprehensive record, capturing essential data related to student inquiries and interactions within the system.



course
department
request
section
student
teacher

1wJN470WUsrEmMbMvRhu
JCZUHxOYcmRPAP4jtr9s
OCke8LhzBJ06Mi3pZi26
V1drnZuw15j0uLSwExsS
Xa7uFJAhAWpRPPuLM8Nz
ZBirjIY6aszNpKfs2Ys2
cQeEgVKOh5xaQEIkwU2M
dl2TRfice7QypeboKema
hdDxkPR3nHrfQGsmP6f0
r1v02VZLquEfF6PEGdjJ
tBzHX4iDDZLawbaJ6Z5T
wLKZqQBnNEZCH6bxYTAN
y5ILYwI97sTTo000Hz5h
vKN3UqRSqdM0FJ2zKeLY

CourseID: "COMP_2310"

Date: "3/11/2019"

StudentID: "111111111"

TeacherID: "2l38NJ3bHYNRAVVkc3KiUPNBfNA2"

Time: "12:00"

problem: "مناقشة عامة"

reason: ""

reqID: "111111111COMP1212"

status: "3"

Figure 10 request collection(table)from firebase database

CHAPTER 4

Implementation

In this chapter, we will delve into the core source codes that power our project, implemented using Android Studio with Java as the main programming language. The primary functions we focus on are sign-in, sign-up, update, and input validation. These functions are fundamental to the seamless operation of our application, ensuring secure user authentication, effective user registration, and the ability to update information.

For a more in-depth understanding and access to the actual source code, you can explore the project repository on our GitHub site. The following sections will highlight key snippets and discuss the rationale behind our code implementation, providing insights into the technical aspects of our project.

- Source code

CHAPTER 5
Testing

In this chapter, we embark on a comprehensive testing phase to assess the functionality and user experience across all roles in our system. From students to instructors, each user role undergoes rigorous testing to ensure a seamless and user-friendly experience. We evaluate critical features such as sign-in, sign-up, request management, and other functionalities unique to each user category.
Our testing procedures aim to identify and address any potential issues, ensuring that the system operates smoothly and efficiently for all users. We will provide insights into the testing scenarios, methodologies, and outcomes, offering a transparent view of the robustness and reliability of our project across diverse user roles.

5.1 (User Role: Instructor)

Features:

5.1.1 – Managing student's appointments
5.1.2 – Managing office hours

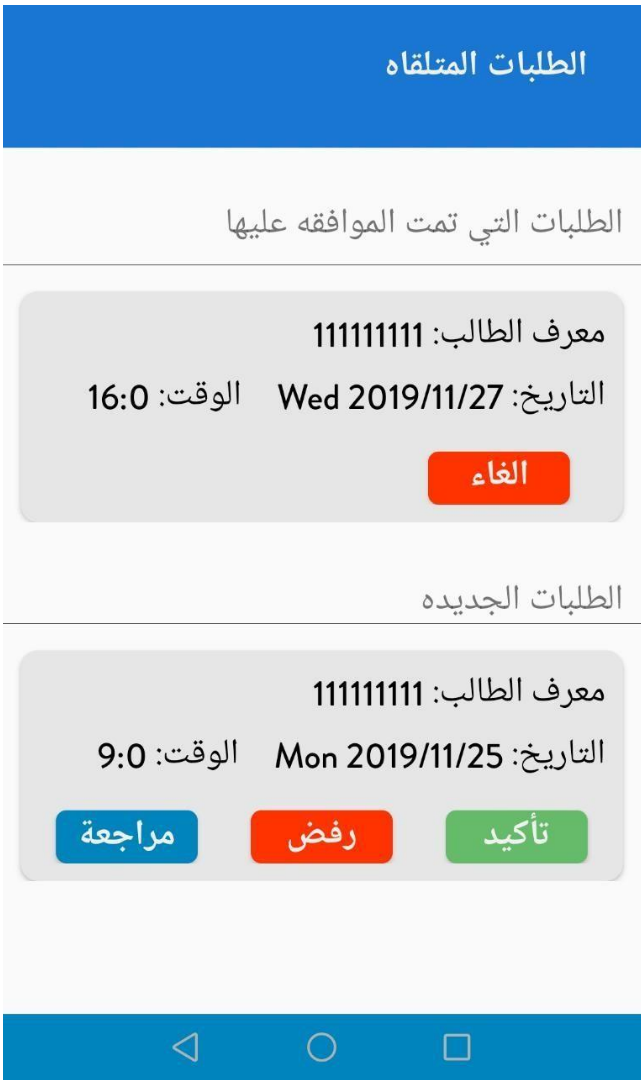
5.2 (User Role: Student)

Features:

5.2.1 – Check appointment availability
5.2.2 – Request to book an appointment

5.1.1 Managing students’ appointments.

To verify the functionality of managing student appointments, including accepting requests, rejecting requests, and viewing appointments.



الطلبات المتلقاه

الطلبات التي تمت الموافقه عليها

معرف الطالب: 111111111

التاريخ: Wed 2019/11/27 الوقت: 16:0

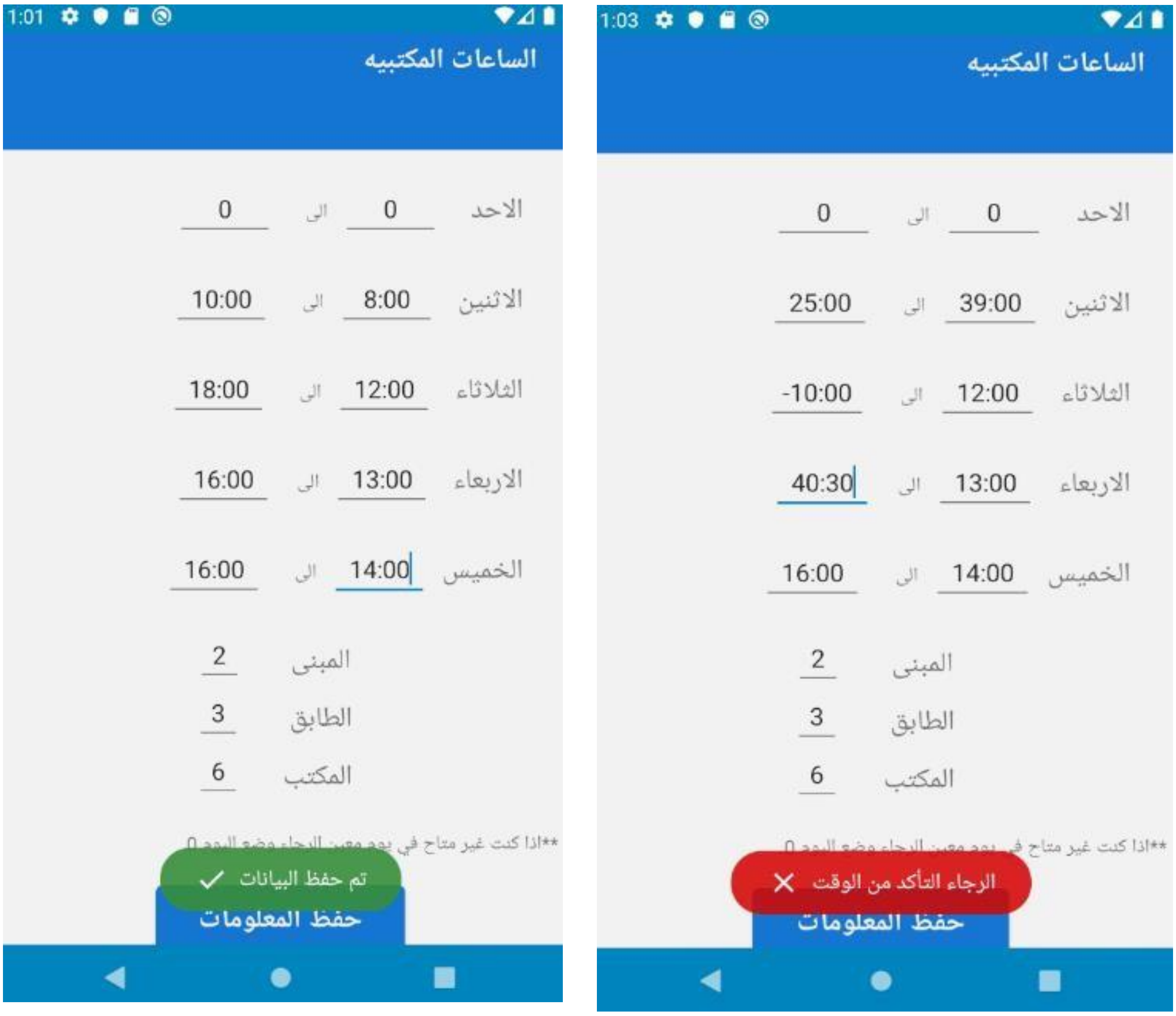
الطلبات الجديده

معرف الطالب: 111111111

التاريخ: Mon 2019/11/25 الوقت: 9:0

5.1.2 Managing office hours.

To verify that the system correctly handles incorrect time input when managing office hours and provides appropriate error messaging.



1:01

الساعات المكتبيه

الاحد 0 الى 0

الاثنين 8:00 الى 10:00

الثلاثاء 12:00 الى 18:00

الاربعاء 13:00 الى 16:00

الخميس 14:00 الى 16:00

المبنى 2

الطابق 3

المكتب 6

**اذا كنت غير متاح في يوم معين الرجاء وضع اليوم 0

تم حفظ البيانات ✓

حفظ المعلومات

1:03

الساعات المكتبيه

الاحد 0 الى 0

الاثنين 39:00 الى 25:00

الثلاثاء 12:00 الى -10:00

الاربعاء 13:00 الى 40:30

الخميس 14:00 الى 16:00

المبنى 2

الطابق 3

المكتب 6

**اذا كنت غير متاح في يوم معين الرجاء وضع اليوم 0

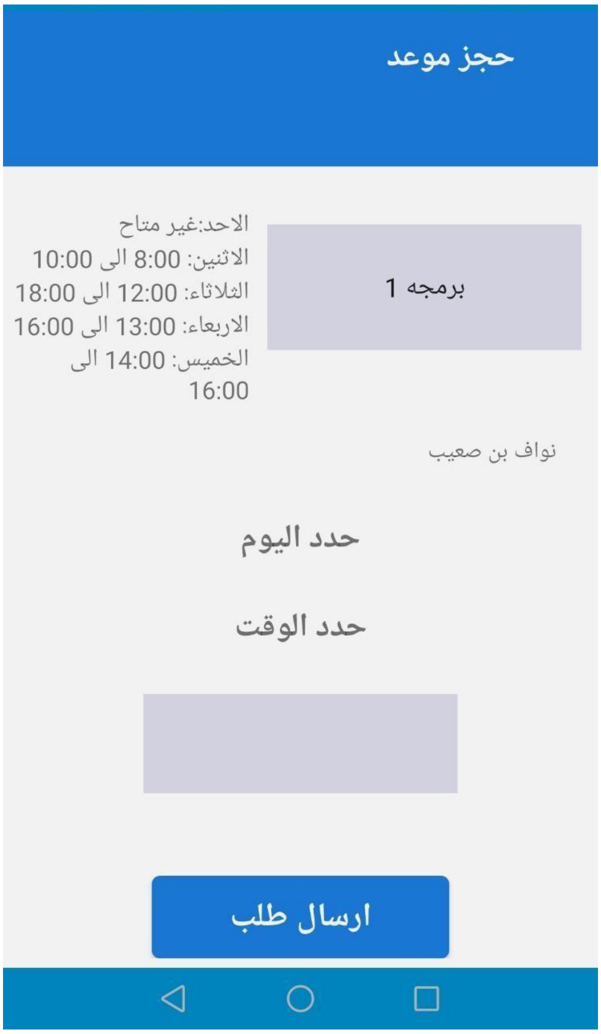
الرجاء التأكد من الوقت ✕

حفظ المعلومات

Figure.5.1.2: Managing office hours

5.2.1 Check time availability

To verify that the system accurately checks and displays the availability of a specified time slot.



حجز موعد

الاحد:غير متاح
الاثنين: 8:00 الى 10:00
الثلاثاء: 12:00 الى 18:00
الاربعاء: 13:00 الى 16:00
الخميس: 14:00 الى
16:00

برمجه 1

نواف بن صعيب

حدد اليوم

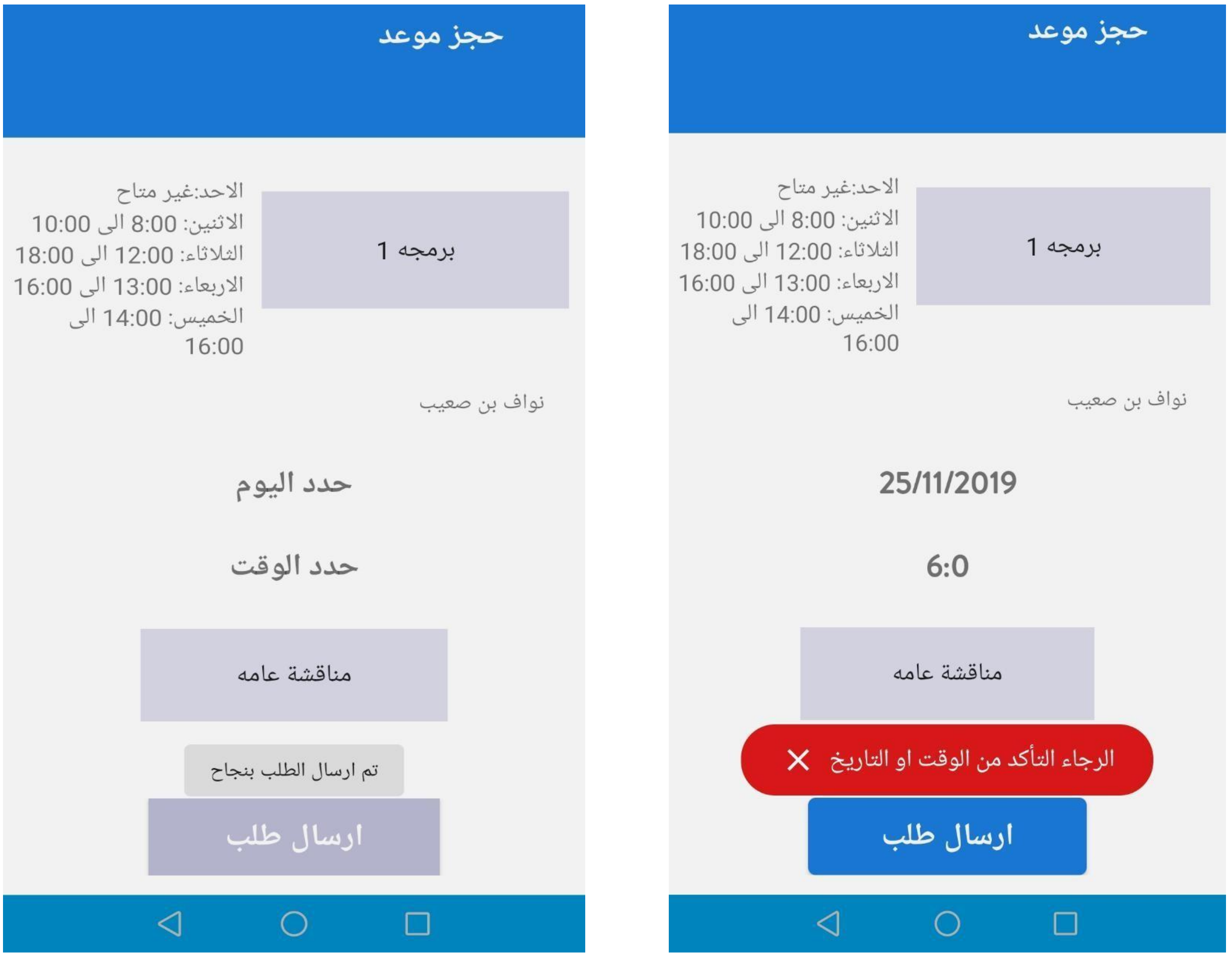
حدد الوقت

ارسال طلب

Figure.5.2.1: Check time availability

5.2.2 Request to book an appointment.

To verify that users can successfully request to book an appointment with an instructor. To confirm that the system correctly rejects appointment requests when the proposed time by the student does not align with the instructor's availability.



حجز موعد

برمجه 1

الاحد:غير متاح
الاثنين: 8:00 الى 10:00
الثلاثاء: 12:00 الى 18:00
الاربعاء: 13:00 الى 16:00
الخميس: 14:00 الى 16:00

نواف بن صعيب

حدد اليوم

حدد الوقت

مناقشة عامه

تم ارسال الطلب بنجاح

ارسال طلب

حجز موعد

برمجه 1

الاحد:غير متاح
الاثنين: 8:00 الى 10:00
الثلاثاء: 12:00 الى 18:00
الاربعاء: 13:00 الى 16:00
الخميس: 14:00 الى 16:00

نواف بن صعيب

25/11/2019

6:0

مناقشة عامه

الرجاء التأكد من الوقت او التاريخ ✕

ارسال طلب

Figure.5.2.2: Request to book an appointment.

CHAPTER 6

Conclusions and Future work

After the successful completion of this project, we believe it will greatly benefit both students and instructors in efficiently managing their time and enhancing overall student knowledge. While some students may already be familiar with the concept of office hours, our system presents a user-friendly and encouraging method to motivate students to actively engage in seeking support, ultimately leading to improved academic performance, higher grades, and elevated GPAs.

Looking ahead, we envision potential enhancements to further optimize the system's functionality. One prospective feature involves empowering instructors with the ability to manage the entire section, facilitating tasks like student additions and removals. Additionally, we are considering the implementation of live conversations between students and instructors, fostering real-time communication and support. Introducing a new page for requesting office hours during alternative time slots addresses the flexibility needed by students with conflicting schedules.

A notable improvement in the future would be the adoption of the 12-hour (AM, PM) time system for a more user-friendly experience. Further, multilingual support could be integrated to cater to a diverse user base, and the development of versions compatible with different operating systems, including IOS, would broaden accessibility.

In the long term, we aim to explore opportunities for integration with the university's infrastructure, creating a seamless connection between our application and university systems. This integration could enhance data accuracy, streamline administrative processes, and further align our project with the overarching goals of the academic institution.

As we conclude this project, we remain committed to its ongoing evolution, welcoming advancements that will continue to positively impact the educational experience of students and instructors alike.

Appendix A: Project Source Code

The Application implementation source code is available [Online] on GitHub:

Source code

References:

1) Java tutorial. W3schools

2) XML tutorial. W3Schools

3) Firebase Firestore Documentation. Firebase documentation

4) XML Overview and Usage. Here

5) Android Studio Tutorial. Here

6) Android Development Basics. Here

7) Firebase Platform Documentation. Here

8) Entity-Relationship Diagram (ERD) Tool. Here

9) Diagramming Tool for Flowcharts and ERDs. Here