



Group 3 | Section: 66332

#	Student Name	ID	contribution
1	Nouv B. Al-Qahtani	442201905	MainActivity, AddNewDutyActivity, ViewDutyDetail, utils folder
2	Weam Alahmadi	442200412	DutyDao,DutyDbHelper,DutyRepository,DutiesAdapter class, utils folder
3	Sarah Alkanhal	442201437	NotificationActivity, SeeAllDuties class, utils folder
4	Nada Alkubra	442202368	Duty, DutyEvent ,CalendarActivity,EventDetailAdapter,utils folder

Instructor Name: Dr. M.Kefi

3/12/2023

Table of contents

Introduction:.....	3
Challenges:	3
Lessons learned and issues solving tricks:	4
Screen shots:.....	6

Introduction:

The academic journey can be daunting, particularly when there are a lot of assignments, deadlines, and activities to keep track of. Our suggestion to tackle this issue is to create an Android application tailored only for KSU (King Saud University) students. With the help of this program, which offers a unified platform for tracking assignments, due dates, and activities, academic and scientific experiences will be more efficiently and effectively organized.

This application's main feature is its integration with the user's smartphone calendar, which makes it simple to access and keep track of academic obligations. Students can effectively manage their semester's tasks, including quizzes, midterms, projects, assignments, and exams, by utilizing the capabilities of a calendar app. They can also keep up with pertinent activities such as hackathons, scientific workshops, and camping excursions.

To ensure students never miss a deadline, the application will provide customizable notifications and reminders. Users can set their preferred notification preferences, including the type and timing of alerts. This feature aims to keep students informed and proactive in managing their academic responsibilities.

Challenges:

Here are the challenges we encountered during the development of our app:

- **Time Management:** Managing the available time to finish the project was a significant challenge. Balancing the demands of coursework, other responsibilities, and the app development required effective time management skills and prioritization.
- **Work Division:** As our app's functionalities depended on each other, dividing the work among team members was challenging. It was crucial to identify interdependencies and assign tasks accordingly to ensure smooth collaboration and progress.
- **Understanding Code Interleaving:** When different team members worked on interrelated functionalities, understanding, and integrating each other's code became a challenge. It required clear communication, code documentation, and regular code reviews to ensure seamless integration and avoid conflicts.

- **Implementing New Functions:** Some functionalities we wanted to include in the app were not covered in our course lectures. This required additional research, learning, and self-directed study to acquire the knowledge and skills needed to implement those functions effectively.
- **Debugging and Error Fixing:** Finding and fixing errors in the code consumed a significant amount of time. Identifying the root cause of errors, troubleshooting, and implementing appropriate solutions required patience, attention to detail, and a systematic approach.
- **Code Version Management:** Managing different versions of the code for each team member posed a challenge. Ensuring that everyone was working on the latest version, resolving conflicts during code merging, and maintaining a coherent codebase required effective utilization of version control tools like GitHub.

Lessons learned and issues solving tricks:

We encountered several valuable lessons and experiences during the development of our app:

- **Learning from External Resources:** We used webpages and YouTube videos as well as other external resources to construct a more complex application than what we had learned. We learned sophisticated ideas and methods that weren't addressed in our course lectures by utilizing the expertise and experience of others. Through the use of self-directed learning, we were able to increase our knowledge and proficiency.
- **Effective Error Handling:** By using the error codes that Android Studio provides, faults can be handled more quickly. We discovered how to use these error codes to focus our search for fixes, which helped us troubleshoot and solve problems more quickly and efficiently. By using this method, we were able to fix bugs more quickly and increase the app's overall stability.
- **Efficient Communication and Documentation:** We discovered how crucial it is for team members to communicate and record things properly. We were able to share knowledge, clear up any misunderstandings, and align our aims through regular

meetings and conversations. Keeping records of significant talks, decisions, and code explanations has also proven helpful for future reference and orienting newly hired team members.

- **Problem-Solving and Adaptability:** There were several difficulties and roadblocks we had to overcome during the development process. These encounters made us realize how crucial flexibility and problem-solving abilities are. We discovered how to handle them methodically by dissecting them into more manageable tasks. This helped the team develop a growth attitude by enabling us to overcome obstacles and come up with workable solutions.

Screen shots:

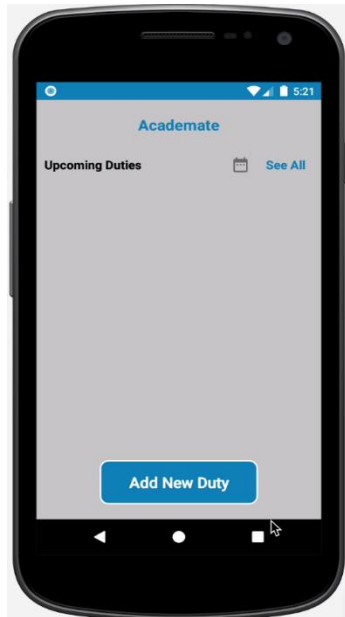


Figure 1 - Upcoming Duties Activity Before Adding Duties.

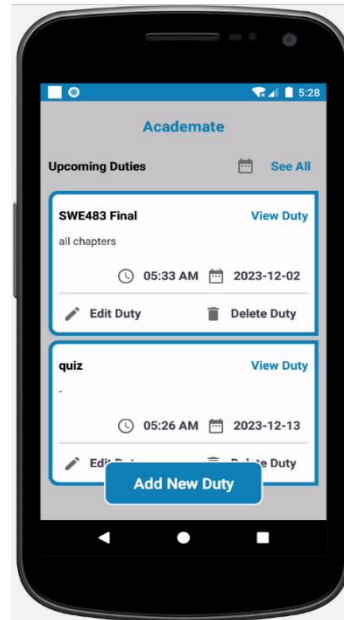


Figure 2 - Upcoming Duties Activity After Adding Duties.

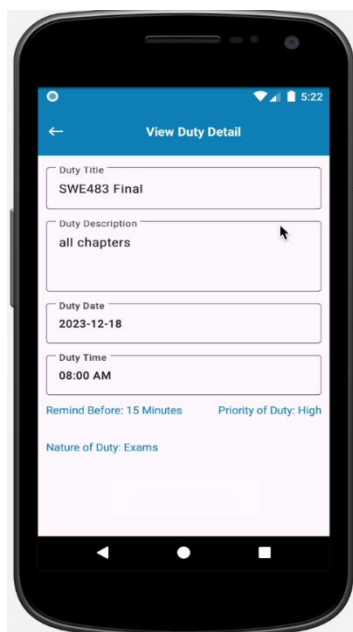


Figure 3 - View Details of Duty.

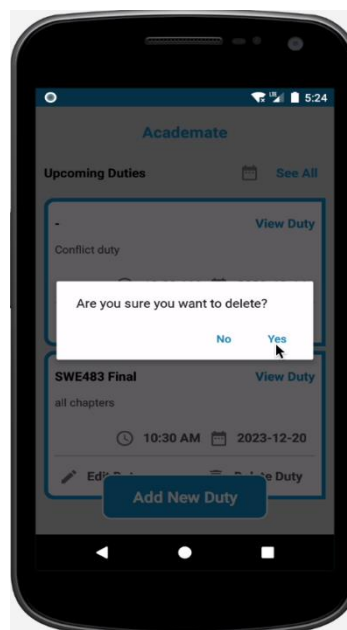


Figure 4 – delete Alert Dialog.

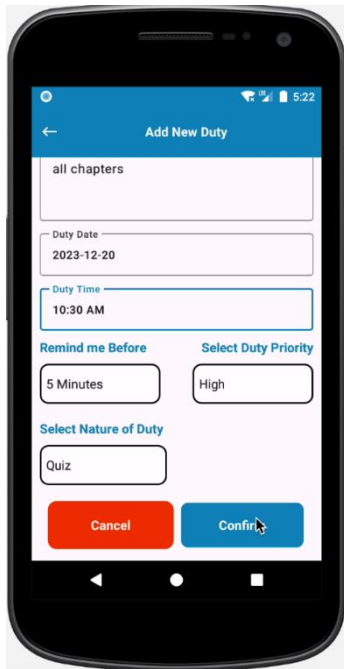


Figure 5 – Edit Duty Activity.

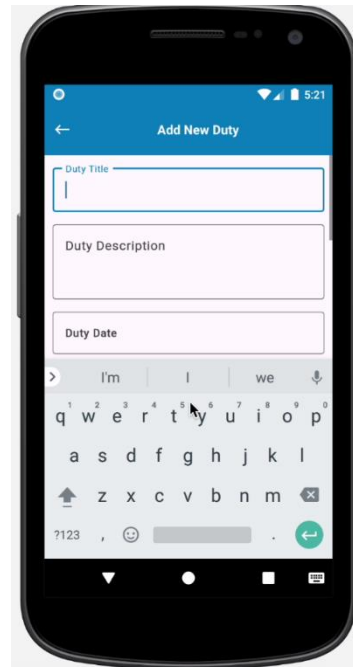


Figure 6– Add Duty Activity.

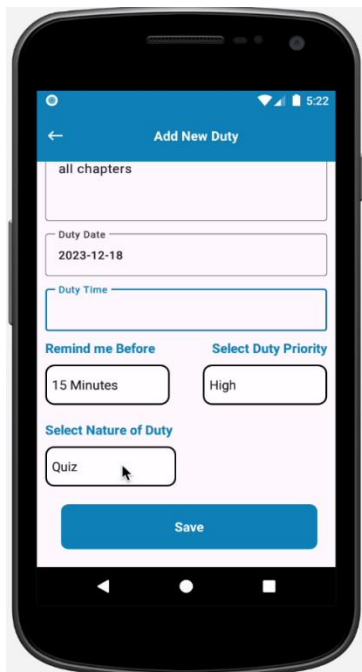


Figure 6 – Add Duty Activity.

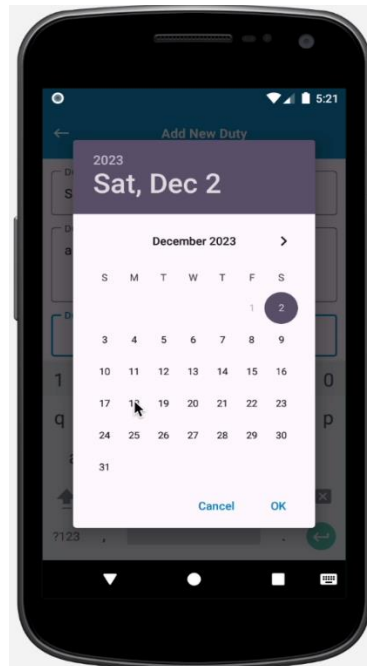


Figure 7 – Date Picker Dialog.



Figure 8 – Time Picker Dialog

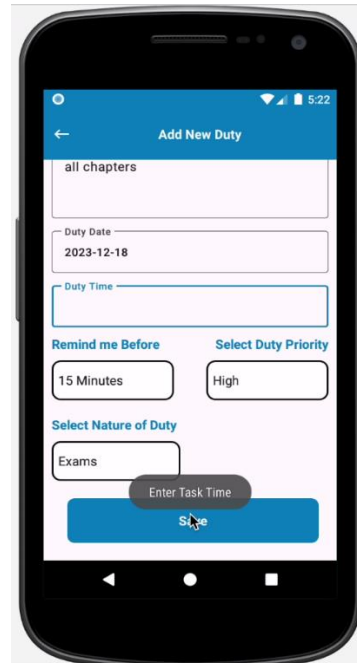


Figure 9 – Missing Field "Time" error Toast

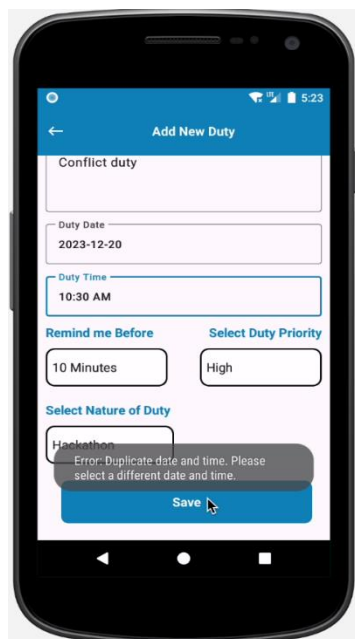


Figure 10 – Duplicate date and Time error Toast (conflict case)

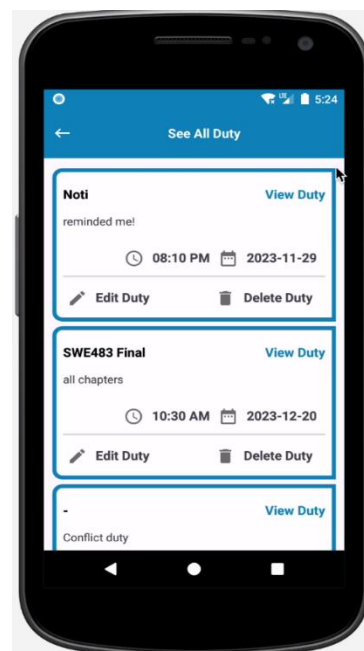


Figure 11 – see All Duty Activity

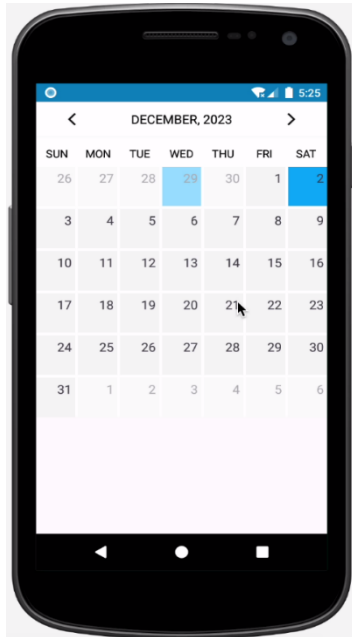


Figure 12 – Calendar Activity After Adding Duties.

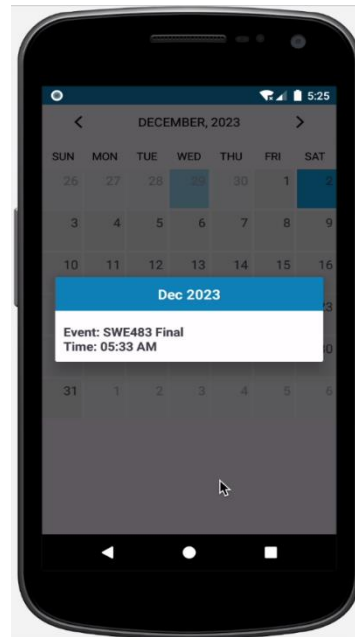


Figure 13 –Duty details after clicking on date in the Calendar.

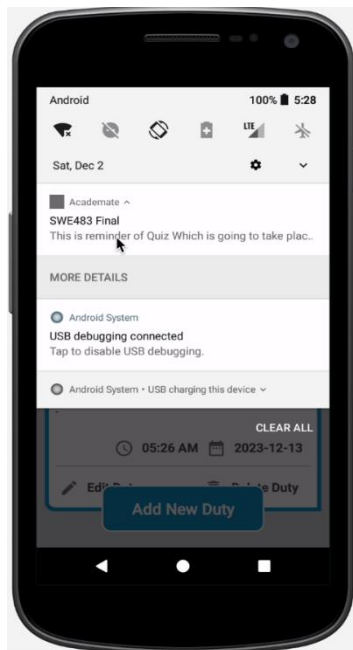


Figure 14 – Duty notification.

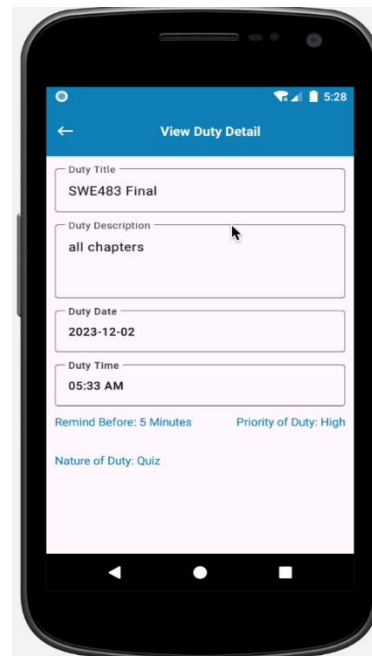


Figure 15 – Duty Details Activity that appears after clicking on the Duty notification