



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Scheduler Android Application

¹Senthil Kumar M, ²Yuvankumar R, ³Suryakumar S, ⁴Varun D

¹Associate Professor, ^{2,3,4}UG Students

^{1,2,3,4} Department of Computer Science and Engineering,

^{1,2,3,4} SRM Valliammai Engineering College, Kattankulathur, Chengalpattu District, Tamil Nadu, India

Abstract: With the modernization of civilization, the uprise on the number of tasks to be done have been increasing steadily. As we go about our lives, we face a lot of tasks which need to be done at specific periods of time. This makes one suffer from managing their tasks effectively. For example, if there are a myriad of people who rely on a particular person to remind them all to do something, it can turn out to be a big and tedious task for that person. This is an android application which makes that task easier, it's a personal scheduler which will manage all your tasks with ease and can also automatically send SMS with the intended message effectively. It can remind a specific number of people simultaneously via SMS and can remind the user with a notification at the time of the event. Thus, it's an all-in-one scheduling app that can take care of all your scheduling needs

Keywords - Task Analysis, Event Analysis, Schedules, Mobile Applications, SMS, Mobile Notification, Offline reminder.

I. INTRODUCTION

In today's world, time management plays a very important role in a person's day-to-day life. From attending important job meetings and conferences in your office to remembering to do smaller tasks such as going to a certain place at a certain time, wanting to buy something or having to do a specific task.

One must plan the events and tasks beforehand or set up a reminder in order to manage the task. Students and workers of today have to keep up with so many different tasks such as assignments, projects, meetings, etc and more often than not they forget to finish these tasks on time or fail to attend certain events simply because they didn't have the memory to do them. The result of this led to the failure of the task which might lead to problems to the specific user.

This is the reason people gravitate towards their friends whom they consider reliable to remind them of important tasks to complete or events to attend. While it usually isn't a big deal for that 'reliable' friend, overtime when more and more of his friends start to rely on him for information, it can become tedious for him to remind everyone and have a memory of what to remind to whom.

The paramount goal of this project is to build a scheduler android application which the user can use to schedule his own events but also remind any of his friends of anything that they wish to be reminded of so that the user and his friends won't miss on any events or any incompletion on any tasks.

II. RELATED WORKS

Ashwini Dalvi made schedulers mostly based on reminding the users only by setting alarm. The set-up alarm helped only the user to get the reminder. This had the limitation of not notifying multiple people at the same time[1]. The one that can remind others required the called person to be online in order for them to get the notification. Jerin joseph created the online application to do scheduling. Since most people don't always have their Wi-Fi or mobile data turned on, it may lead to the failure of the scheduler's entire purpose which is to send reminders[2]. If a person exhausts his daily internet data package, then he can't get reminded by the other person's schedule and might miss out on the event. Sureka lanka created scheduler mainly for academic purposes. There are schedulers which are used only for the sake of a specific event. They couldn't be used for any other purposes[4]. Here, we use the scheduler to send SMS to the specified contacts and inform about the events or tasks by giving a short description about them. Mr. M. Senthil Kumar, in his paper said that current scheduler apps don't support giving reminders to multiple people. So, that encouraged us to build this app [3].

III. MATERIALS AND METHODS

Android Studio is a tool used for developing android applications. It is predicated on JetBrains’s IntelliJ IDEA software. It’s a free software that anyone can use to build apps. It also provides subscription services with extra features. It’s GitHub integration makes it a viable tool for app development as it can enable users to quickly share their code online. It surpassed Eclipse’s ADT tool with its extensive capabilities and went on to become the native platform for Android app development

- The current stable version includes the following features:
- Support for Gradle-based builds
- Refactoring for Android and fast fixes
- Lint tools to detect issues with performance, usability, version compatibility, and other factors.
- Integration and app-signing capabilities with ProGuard
- Wizards based on templates for creating standard Android designs and components
- A drag-and-drop UI component editor, as well as the ability to examine layouts on numerous screen configurations
- Google Cloud Platform compatibility is built-in, allowing connection with Firebase Cloud Messaging and Google App Engine.
- In Android Studio, use the Android Virtual Device (Emulator) to execute and debug apps.

We also have used mobile phones and an emulator to test and use the app. We used XML files to create the frontend display of the application and used Java for the dynamic changes on the application. We get access from the SQLite Database using Room persistence library as an abstraction layer for the backend service.

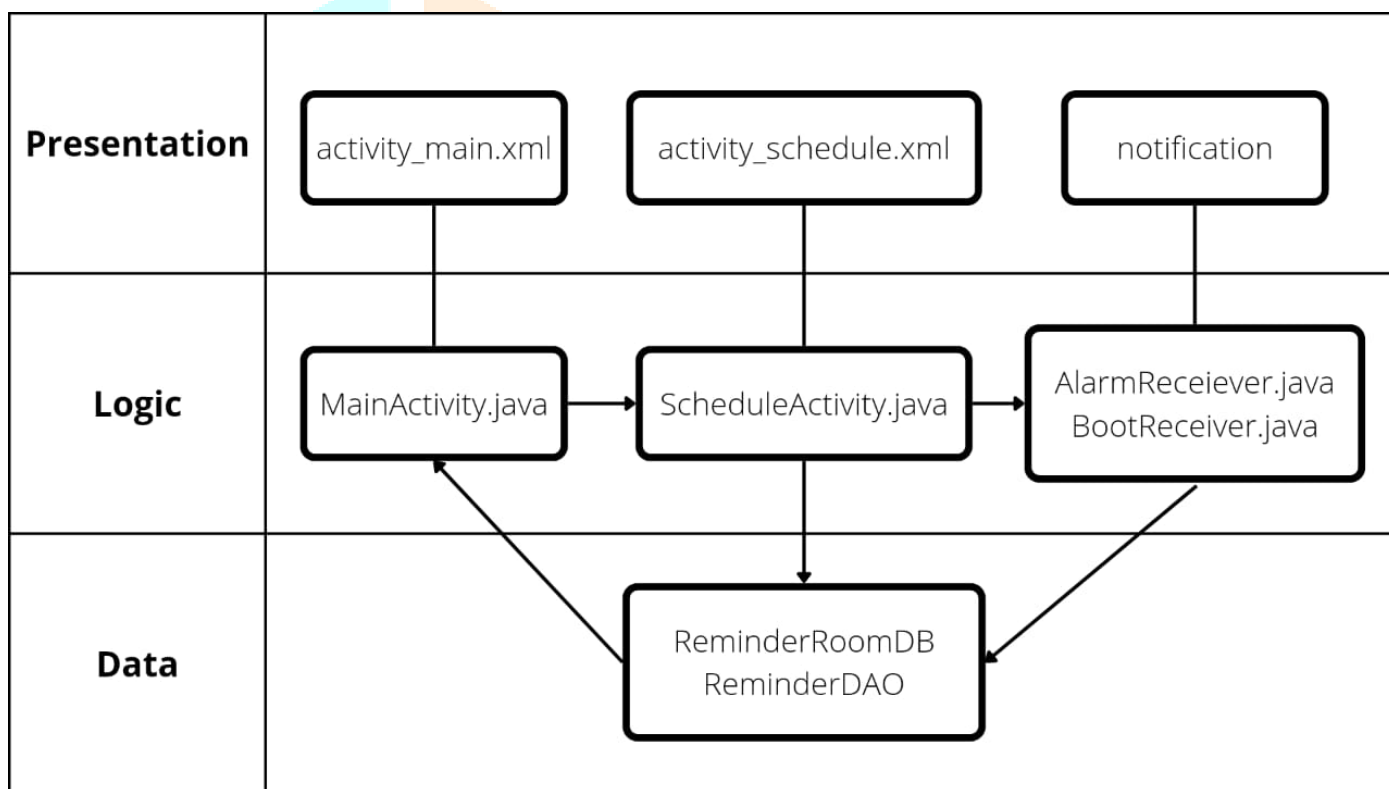


Figure 1 Architecture Diagram of the Scheduler Application

IV. IMPLEMENTATION

First, we created a skeleton structure of the required android application using XML. While creating XML we used customized layout so that we can have full control over the layout. We also used a `CalendarView` class which will help us in creating a visual calendar which can be used to select the date. Then we used a `recycler view` for the calendar which can be used to display all the events or tasks scheduled on a particular day which is selected on the calendar. A floating button is created at the bottom right of the screen which can be used to create a new event on the date selected on the calendar.

Now after we select the date and click on that floating button we get transferred to a new page. Here, we have placed a text box for title where we would mention the title for the event or the task which can be used to easily identify the event or task. Then a checkbox is added to below the title text box with a label of `All-Day Event?` Which is used to check if the event is going to last for many days. Here when we select this option, we don’t need to mention the time since it’s going to be a whole day of event. So, after the textbox we create four buttons which indicate start date, start time, end date, end time. When the checkbox is clicked the times disappear and only the start date and end date will appear. The date will get changed based on what you selected in the calendar. Then we create a description which is optional since it only gives a brief explanation about the event. After all this we create an `add people` button which we can use to insert numbers to the app. These numbers are the people to which the SMS for reminding will be sent to. And finally,

we have created a save button at the bottom. After clicking this button, we can get the created event display on the Recycler View of the calendar on the first page.

At the first page, we wrote the java code to get access to date selection on the calendar and connect it to the recycler view adapter. A Recycler View adapter is used to set datasets to the recycler view. Recycler View makes it easy to efficiently display large sets of data. You can supply the data and define how each items looks and the Recycler View library dynamically creates the elements when they're needed.

The PeopleAdapter class is a RecyclerView Adapter that is used to display the contacts list. The layout for the contacts list is set. The name and number of each person is obtained and their position is set inside the array. The count for the number of people and the list of selected people is returned. ReminderAdapter class is a RecyclerView Adapter that is used to display the scheduled tasks. The layout for the reminders is set. The title and start time of the reminder is obtained. onCreateViewHolder() creates and initializes the ViewHolder and its associated View. onBindViewHolder() is used to fetch the appropriate data and uses the data to fill in the holder's layout. getItemCount() is used to get the size of the dataset.

For the database, we use ReminderDAO class. Reminder DAO class is a database file which is used to insert, delete and select specific reminders from the database. New reminders can be inserted and old ones can be deleted using this function. All or specific reminders can be selected using their start date and id. ReminderRoomDB is a database file which is an extension of RoomDatabase class and it is used to call the ReminderDAO class using a constructor. It holds the database and servers as the main access point for the underlying connection to the app's persisted data.

The Reminders we set using AlarmManager will send the data as an indent to alarm receiver which will alert the user and the people if there is any by creating a notification and sending SMS respectively. When Android system is rebooted, all alarms set are killed, then the system will broadcast a message which will be received by the boot receiver. It'll reset all the needed alarms.

The AlarmReciever class is an extension of BroadcastReciever class. It is used to create a notification channel in which the user can send and receive notifications. It also allows the user to send SMS to the selected people as well. The BootReceiver class is an extension of the BroadcastReciever class. It is used to set the date and time conditions of the reminder. It also allows to create an Alarm Service.

Thus, we can create a SMS and send it to all the required phone numbers and notify them easily. As we are using alarms we can also get notified easily and also schedule tasks beforehand and can execute the schedule without any rush.

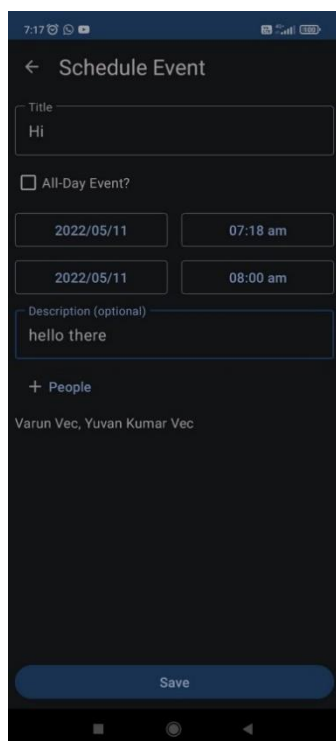


Figure 2 Event Scheduling page

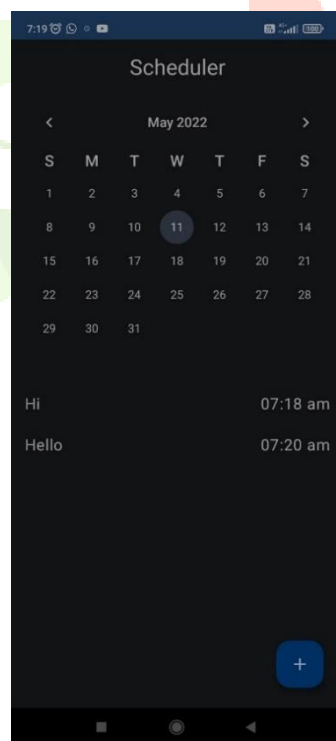


Figure 3 Default page with calendar

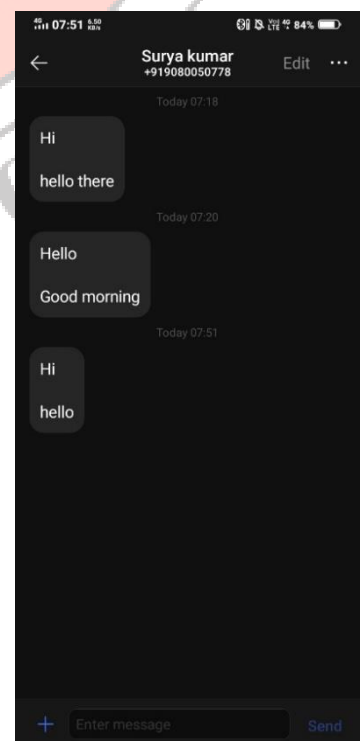


Figure 4 SMS reminders

V. DISCUSSION

Although our app works as intended, sends SMS and notifications correctly and provides all the capabilities that we initially planned for it, over the course of the project we came up with new ideas and discovered new features that could upgrade the app's functionality. Unfortunately, we couldn't find time to add these features as the submission time was inching closer and closer. Nevertheless, we are planning to add these features in the future and those include,

Email support – Though on the surface a SMS seems to be the most accessible way of reminding someone, it won't be a good fit for people working in professional environments. Emails are commonly used for conveying official information to employees in companies around the globe. So, adding the feature to remind someone through email can greatly improve the growth of our app and it may even take it to new heights.

Repeating alarms support - Our app currently sends a reminder once and after that the reminder gets deleted from the database. The issue behind this is that some people may not notice the SMS and thus may not get reminded but if the reminder were to be sent more than once then that person is sure to notice it at some point when they use their phone.

Search for contact support – At the time being, with the add people button in the Schedule Event tab, the user can select the people he wants to remind by scrolling and finding their names in the contacts list. A new problem arises here as the user has to physically scroll and handpick the people he wants to remind. This could be a time-consuming process for the user, especially if some of the people's names start with letters at the end of the alphabet group (Yokesh, Zeeshanth, etc). So to eliminate this issue, we are planning to add the ability to search for contacts in the contact list. This could save so much time for the user and could also significantly improve the event scheduling experience

VI. CONCLUSION

The main objective of our project is to produce an android application which can be used in offline mode. It also provides the feature of sending SMS to other people whose contact we have stored in our Android phone's contact list and notify them at the time of events. The app also provides different types of events so it can help the user to choose the specific type of reminder that he wants to send. The user can also be reminded of the event himself with the notification that our app sends to him at the time of event, thus enabling our app to function as a self-scheduler as well. To conclude, our work will be really helpful for the people who do the arduous task of giving daily reminders to their fellow friends one after the other as it can make that task easier since the app can notify all their friends at the same time. Our further works may provide enhancement to these modules and make scheduling more effective and efficient.

VII. REFERENCES

- [1] A.Dalvi ,I.Siddavatam , “SnoopMe Interactive Task Scheduler Application for Students”, International Conference on nascent Technologies in Engineering, 2019, 23-43
- [2] V.Rohit Kumar, S.Ashwini, T.Jerin Joseph “Schedule Management System using Android application”, International Journal on Recent Technology and Engineering, April 2019, Vol 7, 71-98
- [3] Dr. M. Senthil Kumar “Enhancement of Security Parameters in Mobile Replication Attacks using Wireless Sensor Networks Environment”, International Journal of Advanced Research in Computer Science Engineering, 2014, 1272-1275
- [4] Surekha Lanka, Sidra Ehsam and Aisha Eshan, “Realtime Scheduler System for Academic Institutions on Android Mobile App”, International Engineering Trends and Technology Conference, July 2015, 10-21
- [5] Sohn Timothy, Gunny Lee, I.Smith, ”Place-its: A Study of location-based reminders on mobile phones”, International Conference on Ubiquitous Computing, 2005, 38-47
- [6] S.Usharani, S.Sharmila, P.ManuBala, t.Ananth Kumar, “Mobile Application for Doctor appointment scheduling”, International Conference on System, Computation, Automation and Networking, 29-34
- [7] S.R.Pamulapati, L.Longzhuang, “iDoRemind: A location based Reminder application for Android”, IEEE 5th International Conference, 2017, 21-29
- [8] V.Sarasuk, P.musikasintorn, P.Sinnjakorth, “Alert Me Please: The implementation of an intelligent Time-Management social Application”, Third ICT International study Project Conference, 2014, 39-46
- [9] Z.Fan, Takauki.F, “Scheduling App Utilizing Time-Perspection-Reality in Analog Clocks”, 7th International conference on Advanced Applied Informatics, 19-30