WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Task Scheduling Using Mail Notification

Siddharth Padwal[1], Vivek Sonone[2], Dr. Monika Rokade[3]

Student[1,2], Assistance Professor[3]

(TE) Department of Computer Engineering: Shree Gajanan Maharaj ShikshanPrasarak Mandal Dumbarwadi's Sharadchandra Pawar College of Engineering. Junnar Tal, Pune District, Maharashtra, India.

Sononevivek2020@gmail.com, Spadwal909@gmail.com, and monikarokade4@gmail.com

ABSTRACT:

The primary purpose of the application is to enable the sending of emails to the designated user, which operate as invitations to any business meeting. To sum up, the program helps the user remember the tasks he creates and their schedule. The program can send invites or email notifications to an arbitrary number of people. The creation of an intuitive, feature-rich, and accessible interface was the aim of this application's development. The function helps users manage their time more efficiently and improves the app over its predecessor, increasing the dependability of using our app by sending out email alerts when tasks are due..

Keywords: Task, Scheduler, Email,Notification, Client server Architecture,Android.

Introduction:

Smartphones have become an indispensable aspect of daily life in the modern world. They are frequently used and carried with individuals everywhere they go in their daily lives. Hence, using mobile devices to receive reminders for the crucial chores you want to finish is a productive and successful method. Today's smartphones, with their abundance of features and technological advancements, can offer a wide range of user interfaces. Students and businesspeople can use the Android-based task scheduler application created for this thesis to plan and complete their daily, prioritised work. Users can also receive email notifications for timely alerts.The application allows us to examine a list of chores, create and schedule tasks automatically depending on time and date, and update and delete tasks as needed. Scheduled tasks will be saved in the application calendar, and a fast view of tasks is provided by the program.In addition to providing email reminders for Gmail, Yahoo, and business email accounts that can be set up on a phone, the task scheduler software also keeps track of all the scheduled tasks in its calendar. It also sends out alarm reminders.This program will send email notifications to the specified recipient email address when the user chooses to use their Gmail account as the sender and enters any recipient email address as the recipient. Later, the user will receive an alarm notification on their phone. The preferred user will receive email notifications from the host user using this application, which will function as an invitation to any business meeting. In conclusion, the program assists in reminding the user of the chores he prioritises and the timetable he creates.The program has the ability to send invites or email notifications to random individuals. The goal of this application's development was to create a user-friendly, accessible, and feature-rich interface. The tool aids in improved time management by alerting the user via email and alarm to finish chores in order of priority. The program offers versatility and user-friendly features and was created on the Android operating system. The Android platform powers the devices that are used to build and test the project.

Problem Statement

The existing system of Task Scheduler notify through only the specific application on the device not by email notification.

1. The design of an existing system which is not that comfortable to the user
2. The user can’t Schedule the task as per the upcoming dates.
3. Can’t getting the information of previous scheduled task like task history.
4. Can’t getting the mail notification to the user with task details scheduled by the.

Proposed System.

1. Create a New Account on Application.

2. Login Application. Initialize the application and display the user interface.
3. Add New Task (if task is already scheduled with same title and time then Notify that task is already exist). Allow the user to create tasks and events with their names, start time, end time, and other necessary details.
4. Delete or Edit Task. Allow the user to modify, delete tasks and events with their names, start time, end time, and other necessary details.
5. Trigger Task Notification Mail (If the current time matches the start time of a task, trigger a notification to alert the user that the task has started.) If the current time matches the end time of a task, trigger a notification to alert the user that the task has ended.
6. Allow User to View all Task list. To display the task and sort the tasks and events based on their start time, to ensure that they are displayed in chronological order. Allow the user to view a list of all tasks and events, sorted by start time.
7. Save any changes made by the user to the database. Store the user's input and changes into a data structure such as an array or a database.
8. Close the application when the user exits or logs out. Close the application when the user exits or logs out.

The Task Scheduler comprises the following components:

• Case Actions

Determine which actions to initiate that affect cases, as well as the frequency of these actions. By default, the Timer event is activated in the actions list. The available actions are as follows:

• Unpause cases: The default interval is set to every hour. When hovering over the Service label, a tooltip indicates that it unpauses any case whose designated pause duration has lapsed.

• Calculate elapsed time: The default interval is every hour in UTC. Hovering over the Service label reveals a tooltip that states it calculates the elapsed time based on the configured calendar for all open tasks within active cases.

• Unassigned case: The default interval is every hour. Hovering over the Service label provides a tooltip that explains it triggers actions for self-service cases that have a specified timeout configuration.

• Clean self-service tables: The default interval is every hour in UTC. When hovering over the Service label, a tooltip clarifies that it cleans up unused records associated with the Self-Service Value-Based feature, serving as a maintenance command.

Timer Event: The standard interval is set to one minute. When hovering over the Service label, a tooltip appears indicating the execution of timer start and intermediate timer events. This task is enabled by default.

Clean Web Entries: This feature became available starting with Process Maker version 3.6.0. The default schedule is set for once daily at 8:00 PM on Fridays. Hovering over the Clean Web Entries label reveals a tooltip that states "clean web-entries." By default, this task is disabled.

Emails and Notifications:

Users can select which emails and case notifications are sent by Process Maker and the frequency of these notifications. All actions in the list are enabled by default, including:

- The Service label, which displays a tooltip for Intermediate and end email events.

- Action by Email Response: The default interval is every five minutes. Hovering over the Service label shows a tooltip for Action by email response account email revision.

- Send Notifications: The default interval is also every five minutes. Hovering over the Service label reveals a tooltip for Process Maker mobile notifications.

- Case Emails: The default interval is every five minutes as well. Hovering over the Service label displays a tooltip for Task, triggers, and actions by email notifications.

- Message events: The default time is every five minutes. Hovering on.

SYSTEM DESIGN

- Sequencing diagram:

Figure 1: The user logs in to the portal first. User accounts are established with email address confirmation if they do not expire in the scheduler. Because of the scheduler's interaction, all of the user's data is kept in the database. A user can use a scheduler to assist them plan their duties and add, edit, and remove items as needed. It is necessary to schedule the task and validate it in the database using the user's ID. The email message with the task information is supposed to be sent to the user's email following the task meeting.

Conclusion

Therefore, we draw the conclusion that task schedulers are essential for organising and streamlining our everyday activities so that we may maximise our time and resources. Given the speed at which technology is developing, task scheduling seems to have a bright future. Potential growth and development areas include personalisation, collaborative scheduling, artificial intelligence, integration with smart devices, and environmental considerations. Task schedulers will continue to be a crucial tool for boosting sustainability, efficiency, and productivity as we depend more and more on technology to manage our hectic lives.

REFERENCES

1. Swapnil Kumbhalkar, Shubham Meshram, Grishma Hedaoo, Raksha Tabhane, Priyanka Thoke, Prof. Mukesh Barapatre, "Online Task Management System" November- 2018.
2. B. Persis Urbana Ivy, G Uma Maheswari, Kumar P, JSuganya Pandi."Android based task scheduler and indicator", January 2014."
3. J. A. Larco, J. C. Fransoo, V. C. S.Wiers. Scheduling the scheduling task: a time-management perspective on scheduling", October 2017.
4. Dr. MonikaDhananjay Rokade."Android battery saver system: an alarm telling the user to force stop or close the apps that are drawing power using ai-based programming" Year 02021.
5. Dr. Monika Dhananjay Rokade"Data Science: The emerging domain in IT" Year 2023.
6. Dr. Monika Dhananjay Rokade "Improvement of an Android based Real Time Bus Tracking System" Year 2021.