

Calendar Application That Manages Events and Appointments

Siddharth Bhorge^{1,*}, Tanvee Jungade², Snehal Wabale², Akanksha Kamble²,
Payal Chavan²

Abstract

This article introduces an innovative approach to enhancing the overall user experience of digital calendars by addressing key challenges related to scheduling and conflict management. The proposed solution leverages the capabilities of Flutter and Firebase technologies to streamline event planning and organization using intelligent, adaptive algorithms. These algorithms are specifically designed to resolve scheduling conflicts efficiently while tailoring the experience to individual user preferences. The system supports smooth operation across personal, public, and professional settings, offering users a high degree of flexibility and control. A clean, intuitive user interface ensures ease of use and caters to the diverse requirements of various user groups. The article provides a comprehensive walkthrough, covering every stage from project setup and user interface design to Firebase integration, testing, and debugging. Furthermore, it incorporates Zeller's Congruence formula to explain the mathematical foundation behind date calculations. Overall, this paper highlights how integrating Flutter and Firebase can result in a dynamic, reliable, and user-friendly calendar management application.

Keywords: CalDAV, firebase, Flutter, SELFPLANNER, SyD

INTRODUCTION

The calendar application is designed with an emphasis on efficiency and user-friendly scheduling. It uses advanced technologies including Flutter software, to streamline the management of events and appointments. Employing a client-server architecture, the application consists of a user interface (UI), a robust calendar engine handling the business logic, and efficient data storage. Flutter's cross-platform capabilities ensure a consistent and engaging experience across web and mobile platforms. Comprehensive error handling and logging mechanisms enhance the application's reliability, while optimizations in calendar rendering prioritizing a responsive user interface. The integration of the CalDAV (Calendar based Distributed Authority and Versioning) protocol facilitates seamless online

*Author for Correspondence

Siddharth Bhorge

E-mail: siddharth.bhorge@vit.edu

¹Professor, Department of Electronics and Telecommunication Engineering, Vishwakarma Institute of Technology, Pune, Maharashtra, India

²Student, Department of Electronics and Telecommunication Engineering, Vishwakarma Institute of Technology, Pune, Maharashtra, India

Received Date: September 26, 2024

Accepted Date: March 08, 2025

Published Date: April 19, 2025

Citation: Siddharth Bhorge, Tanvee Jungade, Snehal Wabale, Akanksha Kamble, Payal Chavan. Calendar Application That Manages Events and Appointments. Journal of Web Engineering & Technology. 2025; 12(2): 6–10p.

sharing of calendar data, introducing virtual calendars for collaborative access. Speech software integration, using Server Application Programming Interface (SAPI) text-to-speech, enhances efficiency by allowing event entry through speech. The application also introduces a novel calendar queue that excels in scenarios with uneven event distribution. Notably, it addresses the gap between messaging and calendar apps on smartphones, providing a comprehensive solution for users. Overall, the incorporation of Flutter and innovative features positions the calendar application as a modern, efficient, and user-centric solution for event management and scheduling. At NLC, managing each employee's training, schedule, event participation, and related activities was a time-

consuming and hectic manual task. This calendar application has reduced the traditional method to a new technical method. Admin follows web application for arranging the schedule for training and employees follow mobile application for registering the training [1].

A proper fine-grained calendar is needed where data can be shared easily. So, for this a Calendaring based Distributed Authority and Versioning (CalDAV) protocol is used which is based on online sharing of the calendar data. Calendaring based Distributed Authority and Versioning (CalDAV) is an extension of WebDAV that enables editing websites. Also, the concept of virtual calendars is introduced where many users can access at same time [2].

A traditional method of entering important events by typing is replaced by speech software namely SAPI (Server Application Programming Interface) text to speech to realize speaking. This method helps in reducing time required and effectively uses it. At low level or personal use it may not be that effective but on an industrial level it pays a lot [3]. A brand-new calendar queue that can outperform the traditional calendar queue in situations where event distribution is unequal. But calendar queues struggle with skewed event distributions [4]. Even though all smartphones come with calendar and messaging apps, there is typically little overlap between the two widely used app categories. There are currently few messaging apps with calendar and events features, or calendar apps with messaging features [5].

LITERATURE REVIEW

In this literature review, various scholarly works are analyzed investigating the use of data structures like binary search trees and hash tables in calendar applications, aiming to uncover insights into their effectiveness in efficiently managing events and appointments. System on Device (SyD) is a middleware enabling collaboration among diverse devices and databases. The prototype calendar app showcases its benefits, including automatic updates, real-time constraints enforcement, and coordination between heterogeneous elements. It supports dynamic group formation, mobility through proxies, and efficient group transactions across independent data stores. Overall the System on Device (SyD) enhances collaborative systems by enabling seamless interactions and coordination among various components [6].

By requesting that people place mild limitations on their availability for certain time windows, it is possible to automate the process of scheduling events into time slots. Users provide values to timeslots in a soft constraint system to indicate the relative importance of each occurrence. The issue can then be formalized as an optimization problem with the goal of improving the effectiveness of event scheduling relative to the participant availability [7]. The calendar's function is to indicate when a period has passed, typically the passage of time is marked by the completion of a particular task. Each timer that is entered into a calendar has a unique ID since calendars can manage a huge number of timers [8]. To enable access to a resource in Google Calendar, the account holder who owns it can create a secondary calendar specifically for scheduling events related to it. Furthermore, the owner has the option to grant access to other interested users, allowing them to view and even take control of the events on this shared calendar [9]. A project aimed to create an Android application that addresses the news propagation issues and streamlines event management. The app provides interesting news and allows users to manage event participation, including seat reservations, registration, etc. It uses QR codes for easy participant identity verification during events, eliminating the need for paperwork and long queues [10]. Recent study aims to the manual memory work by automatic software planning system namely SELFPLANNER. In contrast to prior intelligent calendar application initiatives that some tasks were ignored and storing tasks lists without attempting to add them to the user's calendar. This application adds tasks to the user's calendar while considering several limits along with some preferences [11]. When using conventional time tabling systems, users often must actively contribute towards a problem-solving procedure. Hence, we can say that calendar scheduling becomes a continuous process, nevertheless makes it plausible that over time, by seeing instances of meeting scheduling, this knowledge might be automatically picked up [12].

Calendar capability has been added to an increasing number of personal mobile devices in recent years, including all the mobile models, etc. However, due to the complexity of human time management behavior, individual time management and the calendar tools remain understudied [13]. It mainly focuses on the investigation of calendar usage in personal ecologies. It is the first study exploring diverse calendar artefacts and their adaptation. Findings show that changing demands, new tools and user knowledge influence the adoption of calendars. Both paper and digital calendars coexist, emphasizing a precise “appointment culture” and maintaining relationships. The work provides valuable insights into individuals’ interactions with calendars in various aspects of life [14]. It is an extension of decentralized calendar systems which focus on privacy preservation for sensitive data access and management. In these systems, users delegate their timetables to autonomous agents, which process data automatically without user intervention. However, this results in users losing control over their data. To address this issue, a specific protocol is implemented for specific data transactions in a multi-agent system [15]. This protocol aims to enhance privacy preservation and give users more control over their sensitive data in the decentralization in calendar application (Figure 1).

METHODOLOGY

We create a new Firebase project in the Firebase Console and registered our app for the Android platform in the Firebase Console. Then we downloaded configuration files, google-services.json for Android.

- *FlutterFlow project setup:* Install the FlutterFlow CLI and create a new project using Flutterflow create. Connect Firebase to FlutterFlow using Flutterflow firebase and provide the path to the Firebase configuration files.
- *Design UI in FlutterFlow:* Open the FlutterFlow Editor using Flutterflow open. Design the UI using the visual editor by dragging and dropping widgets, customizing styles, and setting up navigation.
- *Calendar management system:* Utilize the Month class to handle input for year and month and calculate the start day and number of days in the calendar. Implement Zeller's congruence to calculate the days of the week. Display the month's calendar, highlighting personal and professional events, as well as appointments entered by the user.

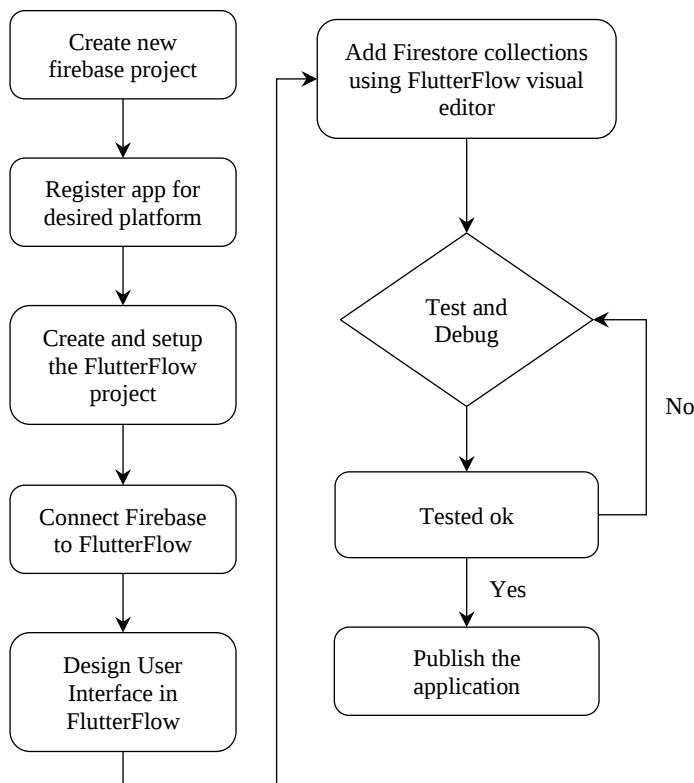


Figure 1. Flowchart of development of calendar application.

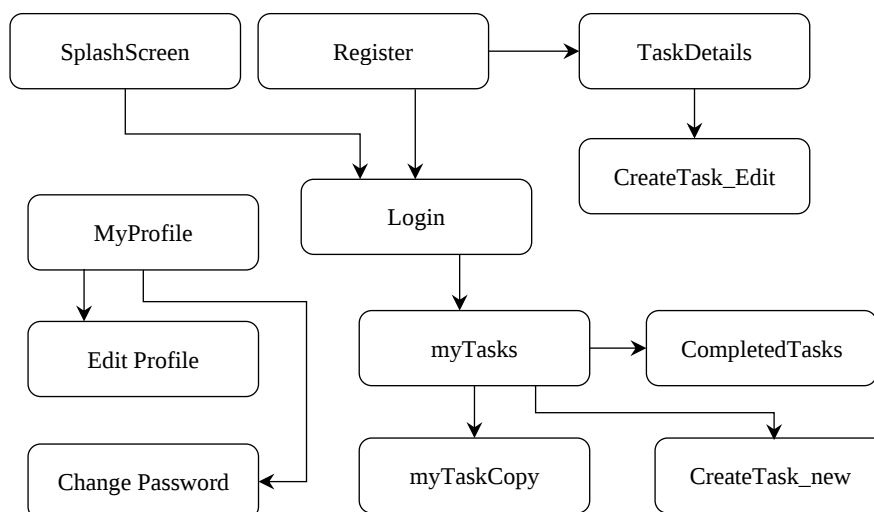


Figure 2. Structure/organization of calendar application.

- *Implement firebase functionality:* Use FlutterFlow's visual editor to add Firestore collections, define data structures, and bind UI elements to Firestore collections and fields. Optionally, set up Firebase Authentication if your app requires user authentication (Figure 2).
- *Test and debug:* Run the app using Flutterflow run and debug as needed. Identify and fix any bugs and refine the UI/UX based on testing.

So, this outlines the steps to set up a FlutterFlow project with Firebase integration, design the UI, implement Firebase functionality, test the app, and publish it. Additionally, it briefly describes the implementation of a Calendar Management System using the Month class and Zeller's congruence for day calculation.

RESULT AND DISCUSSION

The Task Manager excels in efficient event management, providing users with a seamless daily schedule experience through a user-friendly interface. Its key strength lies in privacy preservation, cleverly segregating personal and professional events to ensure sensitive information remains secure. Time optimization is a focal point, allowing users to create action plans in advance, optimizing productivity by prioritizing tasks. User input validation is integrated, reducing errors and upholding data integrity. The clear interface enhances user experience, promoting accessibility.

The dynamic calendar adapts to changing schedules, ensuring up-to-date event tracking. Notably, the system persists data between executions, allowing users to access and modify historical events, promoting continuity. In essence, the Task Manager is a holistic solution, incorporating diverse features to streamline daily planning, safeguard privacy, optimize time, and offer a pleasant user experience with its intuitive design and dynamic functionalities.

CONCLUSION

Calendar application efficiently manages the personal, professional and public events and appointments using data structure like binary search tree (BST) and the HashMap, this organization allows for quick retrieval and display of events for each day. It offers users the chance to add, view and display specific events and appointments of a month. The solution allows users to add and categorize an unlimited number of events for any day, accommodating a wide range of personal and professional appointments. The calendar dynamically determines the number of days in February, considering whether it's a leap year, ensuring an accurate calendar that efficiently matches day and date. While the advantages of the application include improved time management and the ability to schedule meetings and appointments from anywhere, challenges such as difficulty in sharing data with others or migrating

to another system are acknowledged. The limitations of physical space for scheduling far in advance are also recognized. To address these limitations and enhance the user experience, the project would integrate Flutter into the development stack. Flutter framework with its cross-platform compatibility, provides a solution to the challenges such as sharing and accessibility. The utilization of Flutter enables the creation of a mobile app that can be easily shared across devices, overcoming the constraints of physical space and facilitating seamless collaboration. Integration of AI to suggest optimal meeting times based on participants availability and preferences. Enabling real time collaboration on event planning and management, like google docs for documents.

REFERENCES

1. Rajasundari K, Mathi VK. Learning and development center training calendar application. in 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN) 2019 Mar 29 (pp. 1–5). IEEE.
2. Nomura Y, Murata Y, Taniguchi H, Urata M, Muto S. Bring your own calendar: A caldav-based virtual calendar system. In 2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications 2013 Oct 28 (pp. 551–556). IEEE.
3. Yuanlou G, Lianxia L, Jianli Z. SAPI Text-to-speech Technology Application in Industry Calendar System. In 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control 2012 Dec 8 (pp. 355–358). IEEE.
4. Oh S, Ahn J. Dynamic lazy calendar queue: An event list for network simulation. In Proceedings High Performance Computing on the Information Superhighway. HPC Asia'97 1997 Apr 28 (pp. 254–259). IEEE.
5. Yoon I, Ng G, Dong X, Duan H, Aggarwal A, Shah R, Ku G. SuperCaly: Smart life assistance by integrating a calendar app with location tracking, media organization and social connection. In 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC) 2017 Jan 9 (pp. 1–7). IEEE.
6. Prasad SK, Bourgeois AG, Dogdu E, Sunderraman R, Pan Y, Navathe S, et al. Implementation of a calendar application based on SyD coordination links. Proceedings International Parallel and Distributed Processing Symposium, Nice, France, 2003, pp. 8. doi:10.1109/IPDPS.2003.1213438.
7. Boodhoo S, Hosein P. On the distributed optimization of calendar events. In 2017 IEEE 10th International Workshop on Computational Intelligence and Applications (IWCIA) 2017 Nov 11 (pp. 79–84). IEEE.
8. Pontarelli S, Bianchi G, Welzl M. A programmable hardware calendar for high resolution pacing. In 2018 IEEE 19th International Conference on High Performance Switching and Routing (HPSR) 2018 Jun 18 (pp. 1–6). IEEE.
9. Wannous M, Nakano H, Nagai T. Google Calendar™ for managing and monitoring the utilization of a web-based laboratory's resources. In 2011 IEEE Global Engineering Education Conference (EDUCON) 2011 Apr 4 (pp. 210–213). IEEE.
10. Hathaiwichian P, Siriwhattayachoen L, Wongwachirawanich A, Ragkhitwetsagul C. Android application for event management and information propagation. In 2014 Third ICT International Student Project Conference (ICT-ISPC) 2014 Mar 26 (pp. 139–142). IEEE.
11. Alexiadis A, Refanidis I. Defining a task's temporal domain for intelligent calendar applications. In Artificial Intelligence Applications and Innovations III 5 2009 (pp. 399–406). Springer US.
12. Oh J, Smith SF. Learning user preferences in distributed calendar scheduling. In International Conference on the Practice and Theory of Automated Timetabling 2004 Aug 18 (pp. 3–16). Berlin, Heidelberg: Springer Berlin Heidelberg.
13. Wu D. Identifying usability issues in personal calendar tools. In International Conference on Networked Digital Technologies 2011 Jul 11 (pp. 136–146). Berlin, Heidelberg: Springer Berlin Heidelberg.
14. Dittmar A, Dardar L. Personal ecologies of calendar artifacts. J Interact Sci. 2015 Dec;3:1–6.
15. Ludivine C, Yves D, Olivier B, François J. Privacy Preservation in a Decentralized Calendar System. 7th International Conference on Practical Applications of Agents and Multi-Agent Systems, PAAMS 2009, Salamanca, Spain, 25–27 March 2009