MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Proactive Application

Submitted By:

Yash Gupta

0901CS191139

Faculty Mentor:

Mr. Mir Shahnawaz Ahmad

Assistant Professor, Computer Science And Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

MAY-JUNE 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Proactive Application

A project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Yash Gupta

0901CS191139

Faculty Mentor:

Mr. Mir Shahnawaz Ahmad

Assistant Professor, Computer Science and Engineering

Submitted to:

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

MAY-JUNE 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Yash Gupta** (0901CS191139) has submitted the project report titled **Proactive App** under the mentorship of **Mr. Mir Shahnawaz Ahmad**, in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

09/05/22

Mr. Mir Shahnawaz Ahmad
Faculty Mentor
Assistant Professor
Computer Science and Engineering

09/05/2022

Dr. Manish Dixit
Professor and Head,
Computer Science and Engineering

Dr. Manish Dixit
Professor & HOD
Department of CSE
M.I.T.S. Gwalior

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfillment of requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Mr. Mir Shahnawaz Ahmad**, **Assistant Professor**, Computer Science and Engineering

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Yash Gupta
0901CS191139
3rd Year
Computer Science and Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Computer Science and Engineering, for allowing** me to explore this project. I humbly thank **Dr. Manish Dixit**, Professor and Head, Department of Computer Science and Engineering, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Mr. Mir Shahnawaz Ahmad,** Assistant Professor, Computer Science and Engineering , for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Yash Gupta
0901CS191139
3rd Year
Computer Science and Engineering

ABSTRACT

Proactive is a simple-cross platform-task management-streak creation-file sharing app that helps users to achieve their difficult goals quickly.When we set a difficult goal, it requires time, practice, and consistency. Most of us when start a new goal to learn a new thing, we were energetic but after a few days or weeks, we start losing this energy and drop the plan to achieve that goal and switch to something else. Now the question is why did we drop the plan? When the burden of doing it outweighs the joy, we start forgetting our initial goal. Proactive is a one-stop solution to fix these problems.Made using Flutter and Firebase at its core Proactive has a simple, clean, and fast UI and unique features which makes it miles ahead of its competitor.

Keywords: Cross-Platform, Task Management, File sharing, Flutter, Firebase.

सार:

प्रोएक्टिव एक सिंपल-क्रॉस प्लेटफॉर्म-टास्क मैनेजमेंट-स्ट्रीक क्रिएशन-फाइल शेयरिंग ऐप है जो यूजर्स को मुश्किल लक्ष्यों को जल्दी हासिल करने में मदद करता है।जब हम एक कठिन लक्ष्य निर्धारित करते हैं तो उसे प्राप्त करने के लिए समय, अभ्यास और निरंतरता की आवश्यकता होती है। हम में से अधिकांश जब नई चीज सीखने के लिए एक नया लक्ष्य शुरू करते हैं, तो हम ऊर्जावान थे लेकिन कुछ दिनों या सप्ताह के बाद हम इस ऊर्जा को खो देते हैं और उस लक्ष्य को प्राप्त करने की योजना को छोड़ देते हैं और किसी और चीज पर स्विच कर देते हैं। अब सवाल यह है कि हम योजना क्यों छोड़ते हैं? जब इसे करने का बोझ आनंद से अधिक हो जाता है, तो हम अपने प्रारंभिक लक्ष्य को भूलने लगते हैं। इन समस्याओं को ठीक करने के लिए प्रोएक्टिव वन स्टॉप सॉल्यूशन है।इसके मूल में Flutter और Firebase का उपयोग करके बनाया गया Proactive में एक सरल, स्वच्छ और तेज़ UI और अनूठी विशेषताएं हैं जो इसे अपने प्रतिस्पर्धियों से मीलों आगे बनाती हैं।

Table

List of Figures

Chapter 1: PROJECT OVERVIEW

1.1 Introduction:

Achieving any goal requires self-discipline. It involves a conscious awareness of our actions and the ability to overcome some of the bad habits that might be holding us. Instilling self-discipline into our lives is no easy task. It requires steady attention to our actions and determination for wanting to achieve something big. In order to achieve your goal in life, you need to deeply desire the goal that you want. Weak desires bring weak results. You need to have a strong desire to achieve the goal. You have to decide what you want. Start to think about what these goals mean to you. Take time to think about why you are setting the goal you have chosen.

1.2 Objective and Scope

The objective of the proactive app is to track the progress of the goal. Specifically, this project will be targeted at stakeholders interested in learning new skills in a time-bound manner.

1.3 Project Feature

1.3.1 Task/Notes

When you open your app there, the First thing you will look is the task section, which are the tasks that you have to do that day. (Helps in reducing time to decide what to do next, and have a clear goal for a day).

1.3.2 Streak

Adding daily streaks to your tasks, helps you be consistent and at the end of the day gives satisfaction. (if you break a streak, the counter again goes to 0).

1.3.3 Image/File Transfer

Send and Recieve files from one device to another device.

1.3.4 Group Project

Chat and discuss the project with the team working on the same project, Assign the task to a team

1.3.5 Donate

Proactive is an open-source application, we don't use Ads which makes the experience bad, and doesn't charge any subscription fees, we depend on users' donations to make the app run for a long time. I have used razor pay API for the donation.

1.4 System Requirements

Windows Based Requirements:

Computers running Microsoft Windows, IOS, and Android, must meet the following minimum hardware and software requirements.

Microsoft Windows: 7/8/10/11 4 GB RAM minimum

,8 GB RAM recommended

1GB of available disk space minimum 1280 * 800 minimum screen resolution

Software Requirement: Google play store, Apple app store, Google chrome

Chapter 2: Implementation Details

2.1 Frontend

The part of a website that the user interacts with directly is termed the front end. It is also referred to as the 'client side' of the application. It includes everything that users experience directly: text colors and styles, images, graphs and tables, buttons, colors, and a navigation menu.

2.1.1 Flutter

Flutter is an open-source UI software development kit created by Google. It is used to develop cross-platform applications for Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web from a single codebase.

2.1.2 Dart

Flutter apps are written in the Dart language and make use of many of the language's more advanced features.

While writing and debugging an application, Flutter runs in the Dart virtual machine, which features a just-in-time execution engine. This allows for fast compilation times as well as "hot reload", with which modifications to source files can be injected into a running application. Flutter extends this further with support for stateful hot reload, where in most cases changes to source code are reflected immediately in the running app without requiring a restart or any loss of state.

For better performance, release versions of Flutter apps on all platforms use ahead-of-time (AOT) compilation.

2.2 Backend

2.2.1 Firebase

Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.

Firebase is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a real-time database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files.

2.3 Packages

2.3.1 Provider

It is basically a wrapper around the InheritedWidgets that makes it easier to use and manage. It provides a state management technique that is used for managing a piece of data around the app.

2.3.2 Shared Preferences

With SharedPreferences, you can configure your Flutter app to remember the data even after the user terminates their activity. SharedPreferences can be used to store critical data such as passwords, tokens, and complex relational data.

2.3.3 Firebase auth

Flutter plugin for Firebase Auth, enabling Android and iOS authentication using passwords, phone numbers, and identity providers like Google, Facebook, and Twitter.

2.3.4 File Picker

A package that allows you to use a native file explorer to pick single or multiple absolute file paths, with extension filtering support.

2.3.5 Share Plus

Flutter plugin for sharing content via the platform share UI, using the ACTION_SEND intent on Android and UIActivityViewController on iOS.

2.3.6 Firebase Analytics

Flutter plugin for Google Analytics for Firebase, an app measurement solution that provides insight on app usage and user engagement on Android, IOS, and the web.

Chapter 3: TESTING

3.1 Introduction

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure. In object-oriented programming, a unit is often an entire interface, such as a class, but could be an individual method. Unit tests are short code fragments created by programmers or occasionally by white box testers during the development process. It forms the basis for component testing. Ideally, each test case is independent of the others. Substitutes such as method stubs, mock objects, fakes, and test harnesses can be used to assist in testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.

3.2 Benefits

The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. A unit test provides a strict, written contract that the piece of code must satisfy. As a result, it affords several benefits.

1)Find problems early

Unit testing finds problems early in the development cycle. In test-driven development (TDD), which is frequently used in both extreme programming and scrum, unit tests are created before the code itself is written. When the tests pass, that code is considered complete. The same unit tests are run against that function frequently as the larger code base is developed either as the code is changed or via an automated process with the build. If the unit tests fail, it is considered to be a bug either in the changed code or the tests themselves. The unit tests then allow the location of the fault or failure to be easily traced. Since the unit tests alert the development team of the problem before handing the code off to testers or clients, it is still early in the development process.

2) Facilitates Change

Unit testing allows the programmer to refactor code or upgrade system libraries at a later date,

and make sure the module still works correctly (e.g., in regression testing). The procedure is to write test cases for all functions and methods so that whenever a change causes a fault, it can be quickly identified. Unit tests detect changes that may break a design contract.

3) Simplifies Integration:

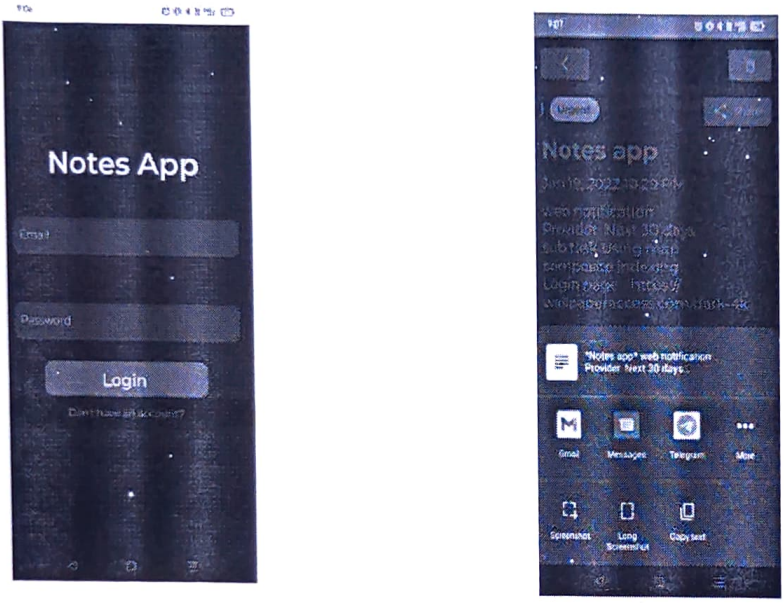
Unit testing may reduce uncertainty in the units themselves and can be used in a bottom-up testing style approach. By testing the parts of a program first and then testing the sum of its parts, integration testing becomes much easier.

4) Documentation:

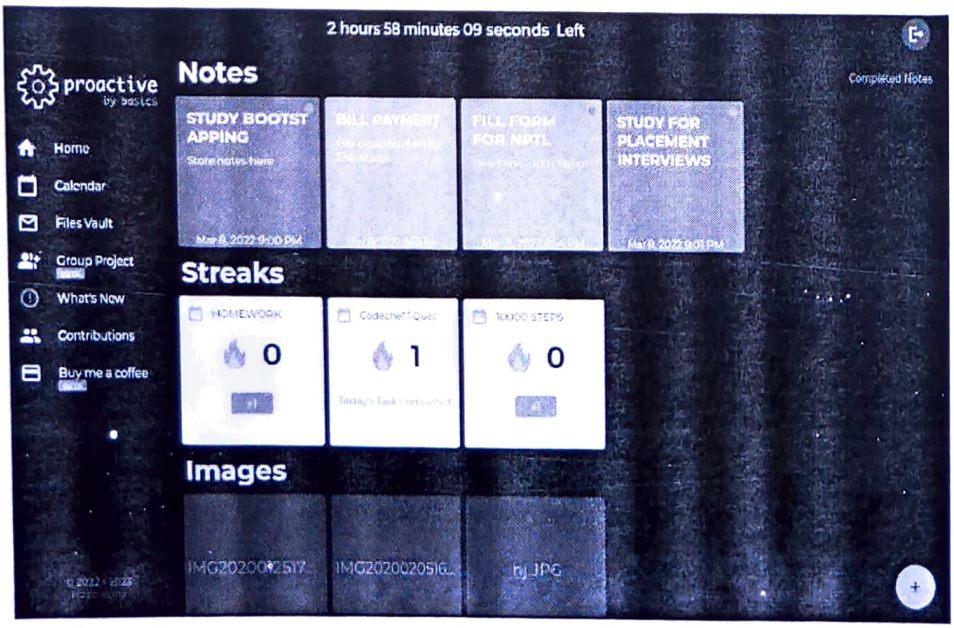
Unit testing provides a sort of living documentation of the system. Developers looking to learn what functionality is provided by a unit, and how to use it, can look at the unit tests to gain a basic understanding of the unit's interface (API). Unit test cases embody characteristics that are critical to the success of the unit. These characteristics can indicate appropriate/inappropriate use of a unit as well as negative behaviors that are to be trapped by the unit.

Chapter 4: FINAL ANALYSIS AND DESIGN

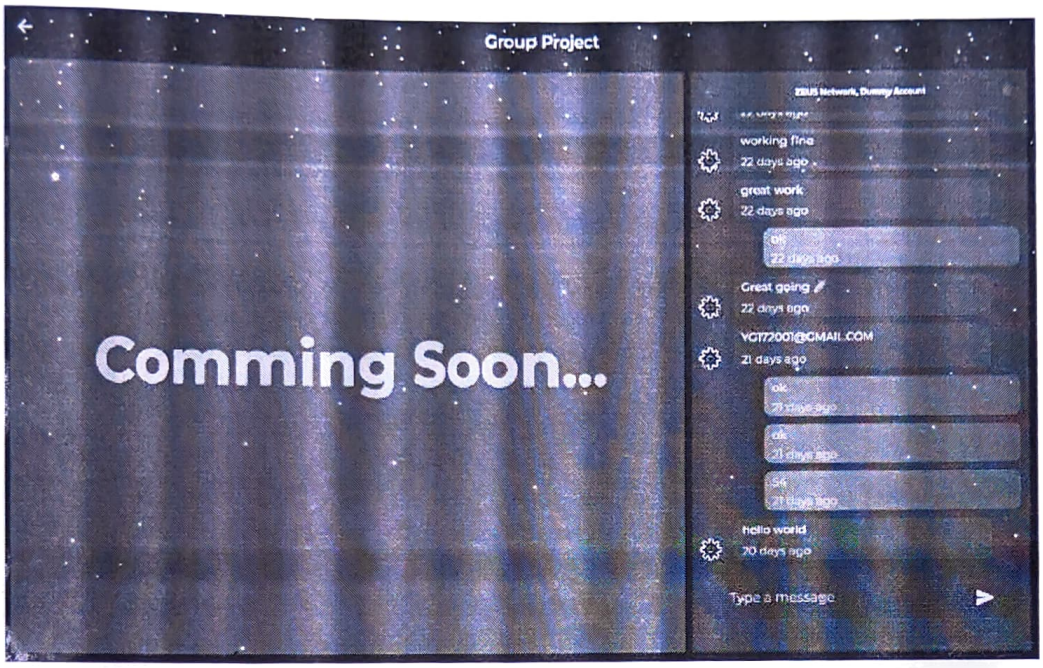
4.1 Screenshots/Snippet



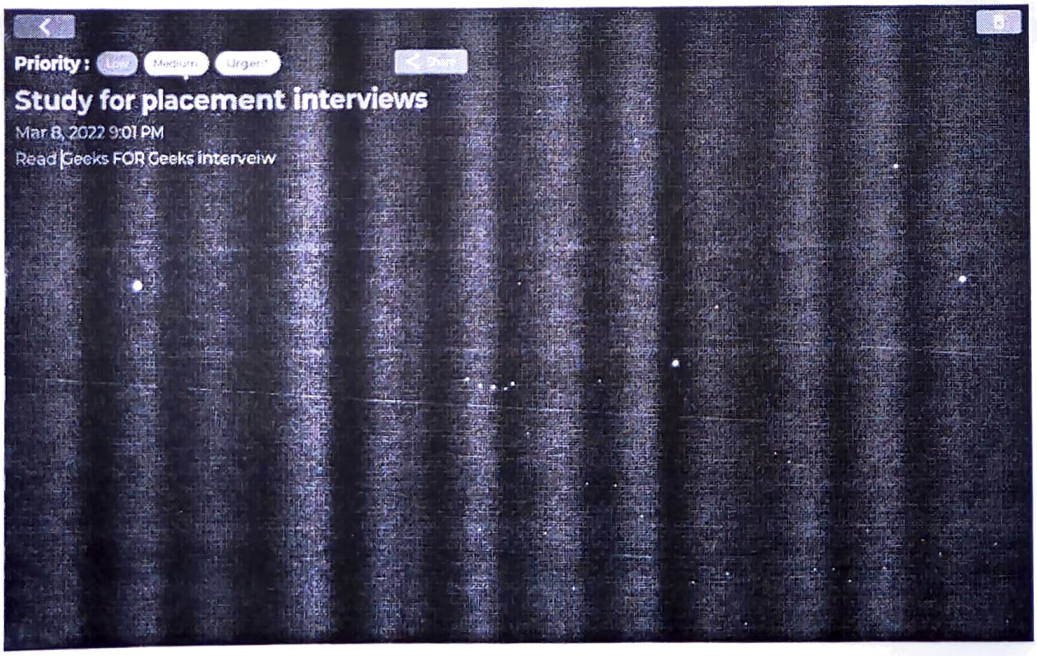
4.1.1 Login Screen (Left) and Share Notes Screen (Right)



4.1.2 Home screen demonstration of Notes, Streaks, Images and many more.



4.1.3 Demonstration of Group project and chat feature



4.1.4 Demonstration of Notes page

4.2 Problem Faced

While developing the project we faced various problems some of them are:

- As Firebase working on Freemium Model the cloud storage is only up to 5 GB for free.
- Not able to use Google Calendar API as it's a paid service.

4.3 Limitations

- Server sometimes takes more time than usual.
- Users can't save notes in proactive shared via another app
- Users can't share streak points with other users.
- Users don't receive notifications when someone sent a message in chat.

4.4 CONCLUSION AND FUTURE SCOPE

The purpose of this project was to provide users ability to track their performance and create future road maps to achieve their goals faster while doing this also make features that make day to day tasks easy like sharing files from one device to another, Priorities the notes, chat with user and more.

In the future, I will fix current limitations and add more features like Google Calendar Integration, Phone Authentification, notifications, and more .

References

1) Flutter Complete Reference: Create beautiful, fast, and native apps for any device Kindle Edition

2) Data Structures & Algorithms in Dart By Vincent Ngo, Jonathan Sande, and Kelvin Lau

3) https://docs.flutter.dev/development/data-and-backend/firebase

4) Living by the Code By Enrique López-Mañas