Project Report On

"To-Do Application"

Course Code	: CSE 420
Course Title	: System Analysis and Design Sessional
Level	: 4
Semester	: I

Submitted by

Name	Student Id
Mst. Amena Akter Asha	1902012
Aktara Yasmin Bithi	1902016
Noman Zuhaer Antor	1902035

Submitted to

Hasi Saha

Associate Professor

Department of CSE

HAJEE MOHAMMAD DANESH SCIENCE AND TECHNOLOGY UNIVERSITY, DINAJPUR-5200.

ABSTRACT

A To-Do application is a software application designed for creating, organizing, and storing digital notes. These apps are often used to capture ideas, thoughts, and information, as well as to keep track of tasks, lists, and reminders. This application offers a variety of features, such as the ability to create and organize notes by topic or category, add tags, search for specific notes, collaborate with others, and sync across devices. This application can be used on a variety of devices, including desktop computers, laptops, tablets, and smartphones. To-Do application is popular among students, professionals, and anyone who wants to keep their thoughts and information organized in a digital format.

INDEX

TABLE OF FIGURES

CHAPTER 1

INTRODUCTION

A mobile application, often referred to as a mobile app, is a software application developed specifically for use on small, wireless computing devices, such as smart phones and tablets, rather than desktop or laptop computers.

Mobile applications can be downloaded and installed directly onto a mobile device from an app store or marketplace, such as the Apple App Store for ios devices or Google Play Store for Android devices. They are usually optimized for the specific operating system of the device they are intended to run on.

Mobile apps can offer a wide range of features and capabilities, including access to online services, offline functionality, push notifications, location-based services, in-app purchases, and integration with other device features like cameras and sensors. Developing a mobile app typically involves designing the user interface, writing the code in a programming language such as Java or Swift, testing and debugging, and finally deploying it to the app store for distribution to users.

1.1 Motivation

Some motivations for using To-Do application are given here:

Everyone needs to write things down. A todo application makes it easy to take notes anytime, add tasks with time and date on any device, and then sync them across all the other devices. The real trick, though, is for those notes to be useful.

The best todo application can give us several options for organizing our notes. That could be folders or notebooks, but it can also be tagging and competent search. When done well, a todo app lets us jot down our thoughts anytime and easily reference them later.

Having a good todo app is like having our own personal internet where we can look up anything we might ever need to know about our work or personal life, no matter where we are. For example; What movies and books did my friends recommend? What are the time for prayers? When's the time to take medicine?

Overall, a todo app helps to be more organized and more productive—even with things that wouldn't expect.

1.2 Objectives

- The primary objective of a todo app is to help us become more aware of how we spend time in the process of doing a specific tasks and how productive that time is.
- Developing a todo app provides facility to add tasks to the list simply and quickly.
- Todo app offers a way to increase productivity, stopping us from forgetting things, helps prioritize tasks, manage tasks effectively, use time wisely and improve time management as well as workflow.
- A good digital tasks list makes it easier to get work done and makes it harder to miss deadlines.

1.3 Organization of the Project

The project work consists of five chapters.

Chapter 1 is Introduction. The purpose of this chapter is to introduce about the system that will be developed. Chapter 2 explains Literature Review, this chapter gives the review of all related software that are currently exists. Chapter 3 discuss about Design and Development of the Proposed software, this chapter gives the working principles of this project. This includes Software life cycle model, DFD, ER Diagram, provides the information about the project in a graphical way. Beside this, the usage of this project is also described here. For example, how the admin and user interact with the system is briefly discussed in this chapter. Chapter 4 explains about Conclusion. The advantage and disadvantages of the project is discussed here. Beside this, the goal for updating this project is also be mentioned.

CHAPTER 2

LITERATURE REVIEW

The To-Do app contains the list of the most important tasks at the top of the list, and the least important tasks at the bottom. By keeping such a list, one can make sure that his tasks are written down all in one place so he doesn't forget anything important. There are various types of todo application available that is close enough to this app.

2.1 Related Work

There are some related work that is similar to this project. Here, we describe some of them -

TickTick is a todo app that embedded calendars and timers. In this app, tasks can be organized using lists, tags, due dates, and priorities, and there's also the ability to add subtasks to any task.

Microsoft todo app is also one of the prettiest to-do list apps on the market. One can set custom background images for every one of his lists, allowing to tell at a glance which list he is looking at.

Any.do offers a really slick mobile app that makes it quick to add tasks, organize them into lists, and add due dates also. But where it really shines is with its daily "Plan my Day" feature, which forces you to schedule when you'll accomplish your various tasks.

If there are hundreds of tasks to keep track of, one of the other great Mac to-do apps might be a better fit, but for most Apple users, Reminders now ticks a lot of boxes.

CHAPTER 3
DESIGN AND DEVELOPMENT OF THE PROPOSED SOFTWARE

Software design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation. Design enables us to achieve the highest level of abstraction, allowing to comprehend better and satisfy objectives. Duplication is avoided, and reusability is increased through design. It is also the most effective method for mitigating risks that we are unaware of.

Software design is the first step in SDLC (Software Design Life Cycle), which moves the concentration from problem domain to solution domain. It tries to specify how to fulfill the requirements mentioned in SRS (Software Requirement Specification).

3.1 Software Development Life Cycle Model

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

SDLC is the acronym of Software Development Life Cycle. It is also called as Software Development Process. SDLC is a framework defining tasks performed at each step in the software development process. ISO/IEC 12207 is an international standard for software lifecycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software. The SDLC composes of four phases: Planning, Analysis, Design and Implementation. Iterative waterfall model is applied to design the project. Because, this SDLC contains phase containment of errors. When an error is occurred to any phase, we can resolve the errors easily to avoid difficulties.

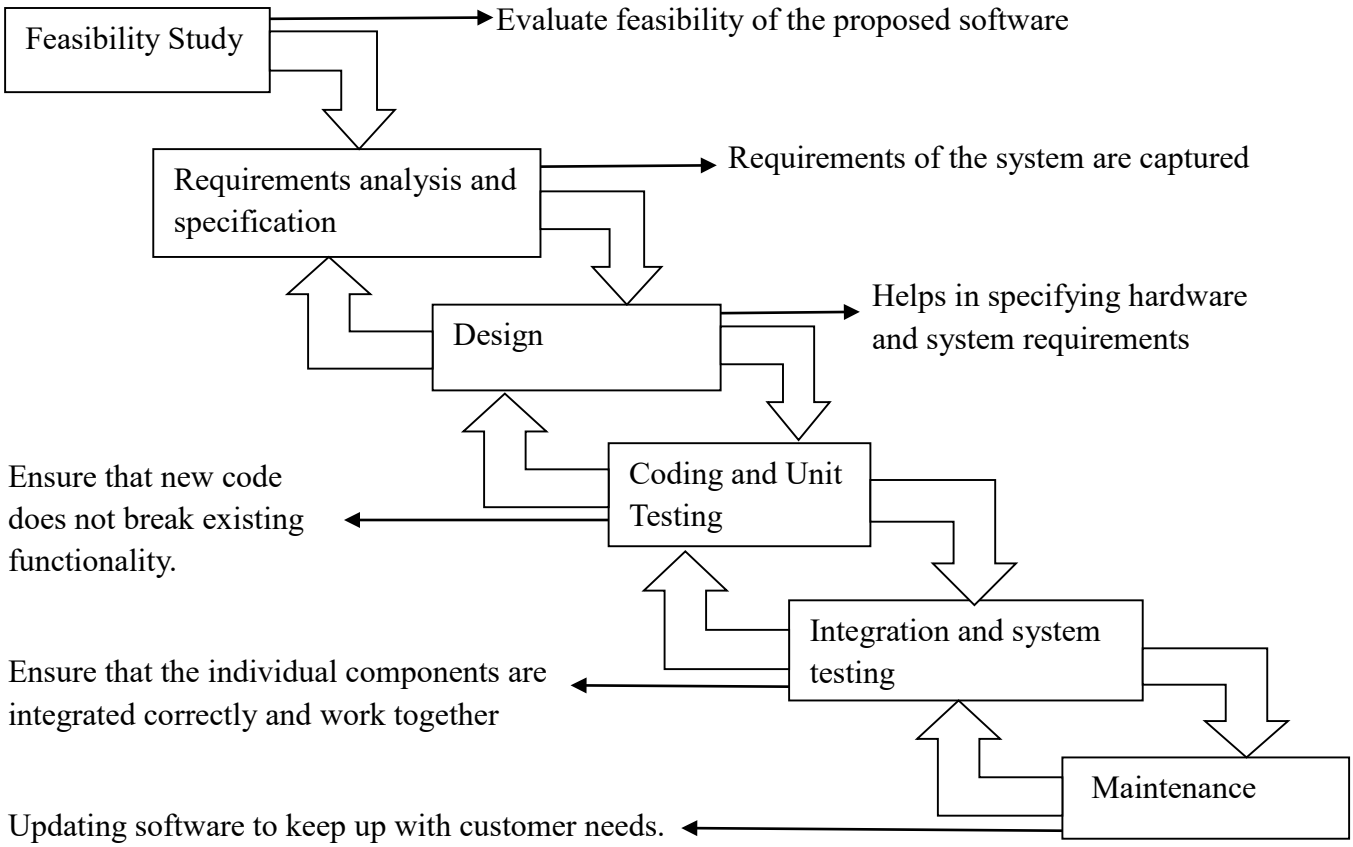

Figure 3.1: Software Life Cycle Model.

3.2 Introduction to Framework

The note application is developed with a framework named Flutter.

Flutter is an open-source UI software development kit (SDK) developed by Google. It allows developers to build native applications for mobile, web, and desktop platforms from a single codebase. Flutter focuses on fast development, expressive and flexible UI, and native performance. Flutter uses the Dart programming language and provides its own widget system called Flutter Widgets. These widgets are the building blocks of a Flutter application's user interface. Flutter also comes with a rich set of pre-designed widgets for common UI components like buttons, text fields, images, and more. Flutter also comes with a range of tools and libraries to aid in the development process, such as the Flutter DevTools for debugging and performance analysis, and the Flutter Firebase library for integrating with various Firebase services.

3.3 Introduction to To-Do App

A "To-Do" app, short for a task management or to-do list application, is a software tool designed to help individuals and teams organize, prioritize, and manage their tasks and activities. These apps are widely used to enhance productivity and ensure that important tasks are not overlooked in the midst of a busy schedule.

3.4 Software Requirements

Name of the components	Specification
Software	Android Studio
Plugin	Flutter
Language	Dart
Database	SQFlite

3.5 System Requirements

Name of the components	Specification
OS	Android 11 (API level 30) or higher
Ram	2 GB
ROM	4 GB
Display	Touch
Resolution	480x880 or higher
Network Capability	Wi-Fi or Cellular data

3.6 Project Analysis

The name of the proposed software is To-Do application which is a mobile platform based app. This app consists of a list of tasks or notes, time for tasks remainder. This application is developed under Flutter framework which is used in Android Studio software. It is used to develop cross platform applications from a single codebase for any web browser, Fuchsia, Android, iOS, Linux, macOS, and Windows.

Dart programming language is required to develop the application under flutter framework.

3.6.1 Database

Database, that is used to store user information named SQFlite.SQFlite is a Flutter plugin that provides a simple way to work with SQLite databases in Flutter applications. It is a lightweight and easy-to-use library that allows developers to store and retrieve data in a structured manner using SQL queries. With SQFlite, developers can create, open, and manage SQFLite databases in their Flutter applications. It provides methods to perform common database operations such as inserting, updating and deleting records as well as executing raw SQL queries. The plugin offers seamless integration with Flutter and provides support for asynchronous operations, allowing developers to perform database operations without blocking the user interface. It also offers transaction support, which helps ensure data consistency and atomicity.

At first, a static method is defined for creating tables and then it is called from another Future type method which initializes the database. These methods contain asynchronous programming for faster execution. After that, Create, Update, Read operations are defined.

3.6.2 Description of the Proposed Project

The source code of To-Do application is divided into 12 dart files (including levels and database). The execution starts from a page that contains the name of the app and a background theme. This page is visible to the user only for 5 seconds. After 5 seconds, user navigates to the page where they can create notes or schedule activities .

Adding tasks: Adding tasks is the main page of the system. Each time user enters into the software, he/she has to go through this page. This page is linked to the database class for checking validation. If any mistake occurs, then it will show a snackbar message to the user. After adding tasks, about to the tasks and fixing time and date, there will be two icons for the operation named exit and save button. Here is a look of this page:

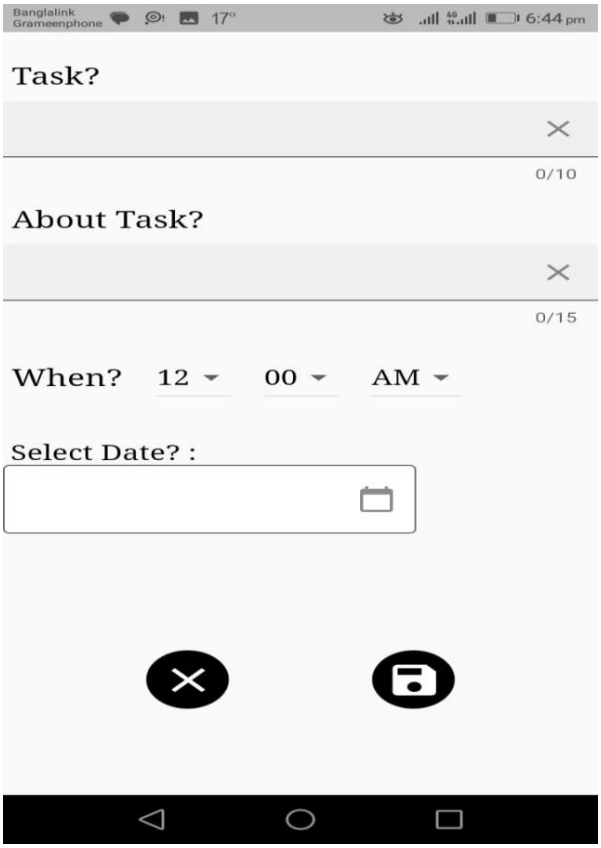

Figure 3.2: Adding tasks page

Tasks list: After adding tasks , all information are saved to the database and user navigates to his tasks list. Read & Insert operation of database is used here. Here is a look of this page:

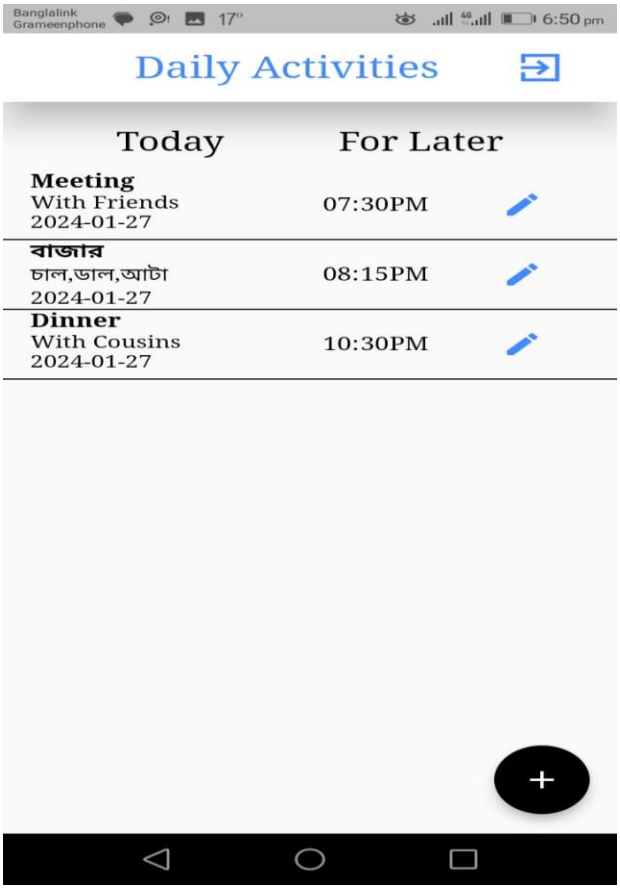

Figure 3.3: Tasks list page

Future tasks list: One can also save tasks for tomorrow or for future. Here is a look of future tasks list:

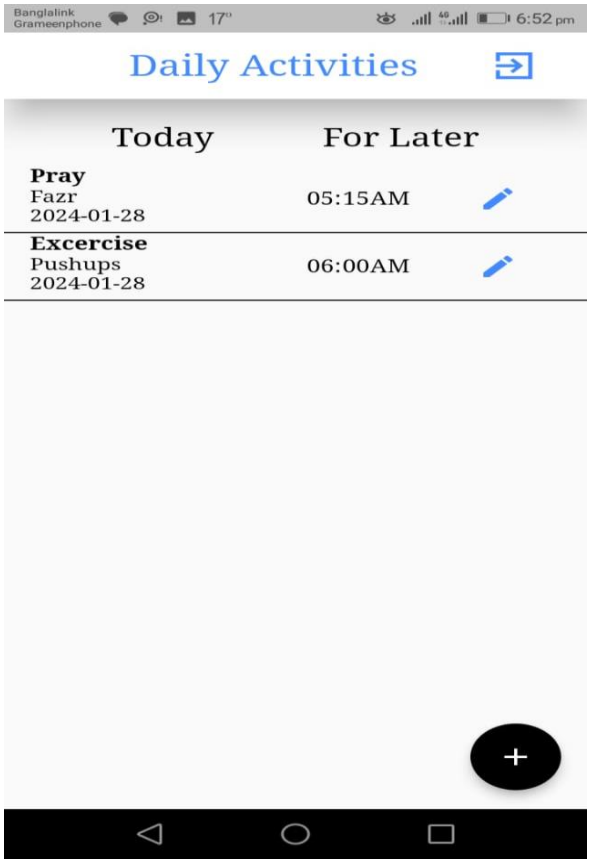

Figure 3.4: Future tasks list

Edit or delete page: One can edit his/her tasks by pressing the edit button and can delete tasks by swiping right to the screen . Here is a look of Edit page:

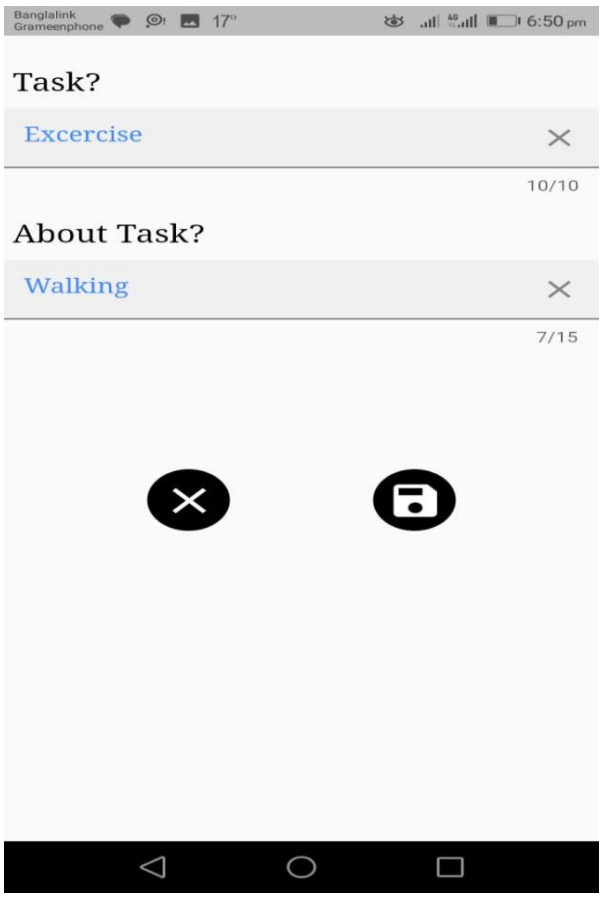

Figure 3.5: Edit Page

3.6.3 DFD diagram of the project

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. The development of DFD'S is done in several levels. Each process in lower level diagrams can be broken down into a more detailed DFD in the next level. The Top-level diagram is often called context diagram. It consists of a single process bit, which plays vital role in studying the current system.

LEVEL 0 DFD

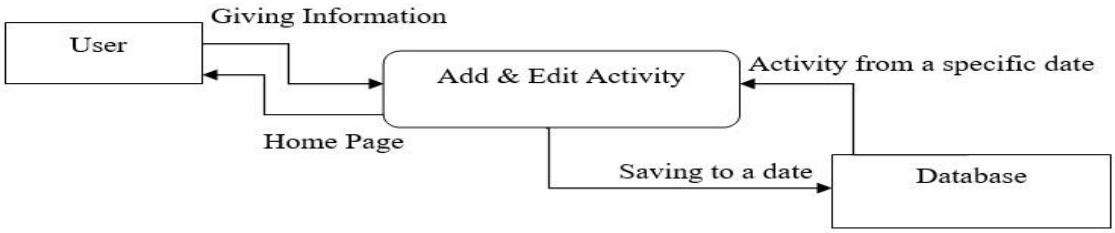

Figure 3.6: Level 0 DFD.

LEVEL 1 DFD

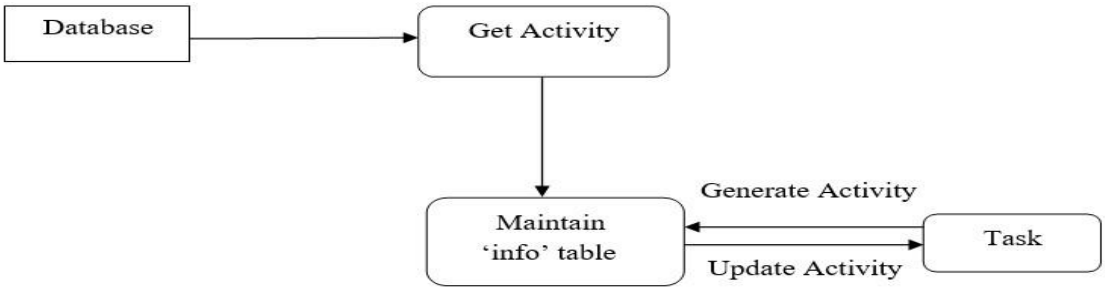

Figure 3.7: Level 1 DFD.

LEVEL 2 DFD

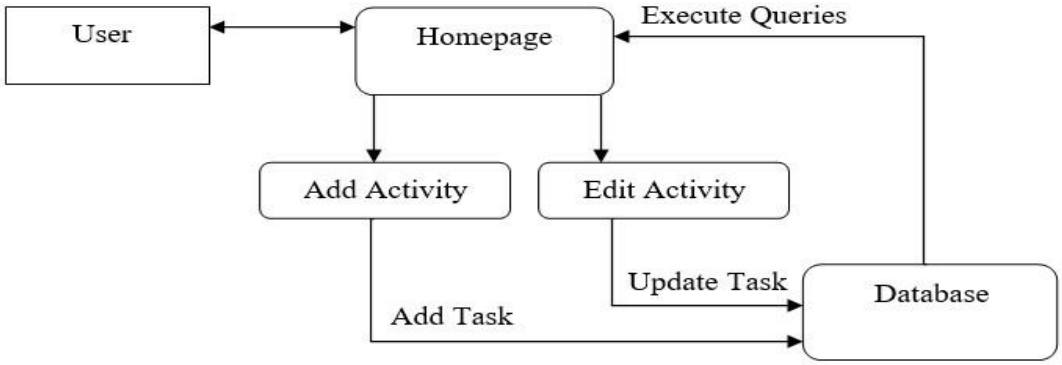

Figure 3.8: Level 2 DFD.

CHAPTER 4

CONCLUSION

To-Do app performs very efficient comparing to competitor. There is a space for further developments with regards of keeping app small and quick. This app can be developed as a sole application as well as a very efficient module to be combined in a larger project. One of the key challenges is to choose appropriate storage solution, that will allow to maintain its biggest advantages.

4.1 Limitation

The following limitations have arisen while developing this project:

- There is no individual delete button.
- It can not give alarm to the user for nearly executing task.
- It can not handle multiple user from a single device.

4.2 Future Enhancement

We will try to remove all the limitations of this software. The code of this application can be expanded in future by including the following features:

- This application will be enhanced by adding user account.
- User interface will be more attractive that is more friendly to user.