CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION TO THE PROJECT

The world has grown to be fast-paced due to developing technology and faster communication. Ideas like Work from Home have become a common trend, especially after the pandemic. These factors had an enormous impact on the work culture that it has bloomed to an extent where if not properly maintained the work would invade even the personal lives of the individuals. The deadlines are being pushed onto the people whether it be in academics, peer group communities or the workplace. Organizations and companies break down projects into smaller modules and delegate those smaller tasks to several employees for them to work on. But as the workloads get hectic people lose track of what to prioritize and in the end, what remains is a cluttered list of tasks left unattended. The crammed-up pending tasks not only affect productivity and attendance in different aspects of life, it even affects the mental health of individuals.

The schools and colleges are accustomed to tailoring the students to master the skill of efficiently organizing their time blocks as it is significant in the coming life ahead but fails to deliver it as something of significance. The idea remains vague and the methods of building a timetable are never handed over to the students as they are made to stick with the schedules built by the schools or colleges. Fig 1.1 sums up the benefits of tracking and organizing time.

Fig 1.1 The Importance of Time Management

1.1.1 Strategies involved in Time Management

a) Pomodoro

The Pomodoro Technique is a time management system that encourages people to work *with* the time they have—rather than against it. Using this method, you break your workday into 25-minute chunks separated by five-minute breaks. These intervals are referred to as Pomodoro. After about four Pomodoro, you take a longer break of about 15 to 20 minutes. The basic structure of a Pomodoro Cycle is given in fig 1.2.

The idea behind the technique is that the timer instils a sense of urgency. Rather than feeling like you have endless time in the workday to get things done and then ultimately squandering those precious work hours on distractions, you know you only have 25 minutes to make as much progress on a task as possible.

Additionally, the forced breaks help to cure that frazzled, burnt-out feeling most of us experience toward the end of the day. It's impossible to spend hours in front of your computer without even realizing it, as that ticking timer reminds you to get up and take a breather.

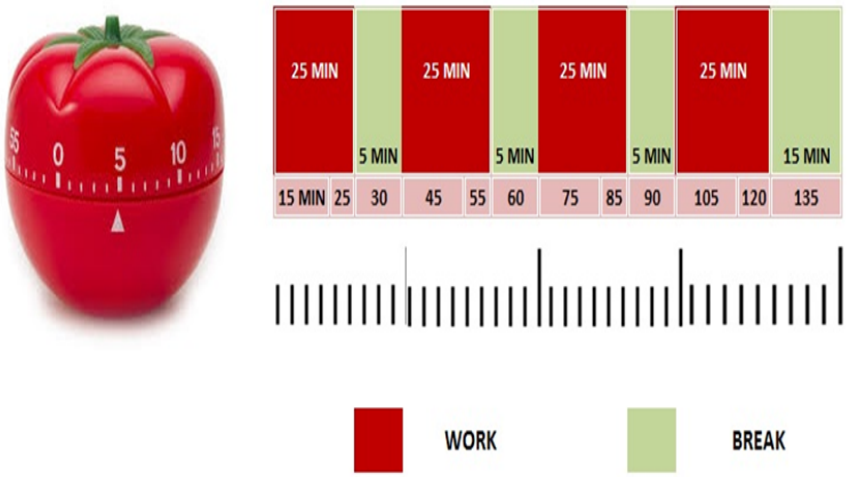


Fig 1.2 The basic structure of a long Pomodoro cycle

b) Eisenhower Matrix

In a 1954 speech, Eisenhower quoted an unnamed university president who said, "I have two kinds of problems, the urgent and the important. The urgent are not important, and the important are never urgent."

Over 3 decades later in his best-selling book The 7 Habits of Highly Effective People, Stephen Covey repackaged Eisenhower's insights into a simple tool to prioritize tasks, now known as the

Eisenhower Matrix (also known as The Time Management Matrix, The Eisenhower Box, The Eisenhower Method, and The Urgent-Important Matrix). This framework for prioritization helps you combat the "mere-urgency" effect (more on that later), eliminate time-wasters in your life, and create more mental space to make progress on your goals.

Try the Eisenhower Matrix when

i. Sliding into deadlines after the other
ii. busy but don't feel like the work has a high impact
iii. There are long-term goals to be pursued but no time or energy to make progress on them
iv. Having a hard time delegating and/or saying no

A recent study in the Journal of Consumer Research set out to examine how people decide what to work on when faced with tasks of mixed urgency and importance. Across five separate experiments, researchers observed a curious pattern: people's attention is drawn to time-sensitive tasks over tasks that are less urgent even when the less urgent task offers

greater rewards. This psychological quirk — dubbed the "Mere-Urgency Effect" — explains why people are so bad at task and time management. Individuals are likely to prioritize tasks with a deadline over tasks without one regardless of their long-term payoffs.

And the effect is even more prominent in people who describe themselves as “busy”. The same researchers found that self-described busy people were more likely to select urgent tasks with lower payouts because they were already fixated on task duration. If someone is experiencing a time crunch, they are most likely to continue prioritizing tasks that keep them focused on the clock.

But there's good news too — the mere-urgency effect can be reversed. When participants were prompted to consider the consequences of their choices at the time of selection, they were significantly more likely to choose the important task over the urgent one. The findings suggest that if the long-term importance of non-urgent tasks is kept in view, the pull toward urgent distractions could be overcome and focus on what matters. That’s where the Eisenhower Matrix comes in.

The Eisenhower Matrix is a simple tool for considering the long-term outcomes of daily tasks and focusing on what would be most effective, not just most productive. It helps to visualise all the tasks in a matrix of urgent/important. All of the day-to-day tasks and bigger projects will fall into one of these four quadrants:

1. Urgent & Important tasks/projects to be completed immediately
2. Not Urgent & Important tasks/projects to be scheduled on your calendar
3. Urgent & Unimportant tasks/projects to be delegated to someone else
4. Not Urgent & Unimportant tasks/projects to be deleted

In the real world, the distinction between urgent/non-urgent, and important/not important is much murkier than under experimental conditions. Here's how Steven Covey breaks it down:

Urgent matters are those that require immediate action. These are the visible issues that pop up and demand your attention now. Often, urgent matters come with clear consequences for not completing these tasks. Urgent tasks are unavoidable, but spending too much time putting out fires can produce a great deal of stress and could result in burnout.

Important matters, on the other hand, are those that contribute to long-term goals and life values. These items require planning and thoughtful action. When important matters are focused, time, energy, and attention are managed rather than mindlessly expanding these

resources. What is important is subjective and depends on the values and personal goals. No one else can define what is important for individuals.

Below in Fig 1.3 is an example of an Eisenhower Matrix and an in-depth look at each of the four quadrants of the Eisenhower Matrix to help you identify which tasks go in each and how to handle them accordingly.

Fig 1.3 Eisenhower Matrix

Quadrant 1: Urgent & Important

Urgent and Important tasks demand actions to be taken quickly. These items typically have visible deadlines and consequences for stalling on taking action. Most often, these are either things that were sprung from an external source or things that were put off until faced with a looming deadline. Either way, they require a crisis mode response.

For example:

Covering a project for a colleague out sick

Car stalls on the highway

Sink springs leak and flood the kitchen with two feet of water

Clients come with a pressing problem

Getting assigned a last-minute deadline

Quadrant 1 tasks are inevitable. There will always be something beyond a person's control. However, the problem comes when focused on these unexpected or deadline-driven tasks to the exclusion of important long-term goals.

Covey cautions that spending too much time on Quadrant 1 tasks can lead to increased stress, burnout, and the sense that the days are out of control. Spending all day putting out fires will quickly rob energy and passion for the work, and may make it easier to settle into mindless escapism found in Quadrant 4.

Quadrant 2: Not Urgent & Important

Not urgent, but important tasks are the activities that help achieve long-term goals. These may not have a deadline (or even an end date) so it is easy to put them off in favour of more urgent tasks. However, these tasks have a much greater effect on long-term effectiveness in completing goals.

Examples:

Planning for long and short-term projects

Regular chores or maintenance projects

Professional networking and personal relationship building

Learning a new skill, keeping up with current research in your field, attending educational events

Exercise and routine healthcare

Covey says that Quadrant 2 is the sweet spot of personal time management. This is the spot where the focus is on problems (as with Q1) but on opportunities and growth. Living from this quadrant of the matrix leads to proactive and prioritised activities that grow the skills and energy of the individuals, and contribute to accomplishing meaningful goals. Quadrant 2 is where "deep work" happens because pressing distractions are freed.

By attending to Q2 consistently, the number of pressing problems that pop up in Q1 is decreased. Living in Q2 means creating a plan to complete projects and avoid possible

problems. For example, if completing routine car maintenance is put off, when the car stalls out it must be paid off.

Quadrant 3: Urgent & Not Important

Urgent but Not Important tasks are best described as busy work. These tasks are often based on expectations set by others and do not provide progress to long-term goals.

Examples:
Unnecessary interruptions from coworkers
Checking phones or emails whenever it goes off
Responding to certain texts, emails, or social media messages
Acting on coupons or limited-time offers
Some meetings

Quadrant 3 is where the mere urgency affects lives. The drive to complete tasks because of real or assumed deadlines means taking on tasks that aren't meaningful. Given that Q3 tasks are urgent but typically related to someone else's priorities, spending too much time in this square can feel like attending to tasks that are of no personal interest. Focus on Q3 tasks may feel like not living up to the larger life goals or not having control over day-to-day life.

Covey suggests delegating as many Q3 tasks as possible. Can the meeting notes be taken by someone else? Getting the groceries delivered instead of going to the store? Letting the children do the dishes after they are done with the late. Can a digital assistant be hired to schedule family doctor visits? Is there anything in life that can be automated?
If these tasks can't be delegated, try to keep them from taking over the day:
Turn off notifications on the phone and computer while working
Be clear with others about the time required to be spent on a particular task.
Save Q3 tasks for times low on energy rather than putting them first thing in the morning
Negotiate the workload with the boss
Practice saying no

Quadrant 4: Not Urgent & Not important

Not urgent and not important tasks are time-wasting activities that should be ruthlessly cut out. These activities don't contribute to progress on the goals but can end up taking over large chunks of time.

Examples:
Watching TV for hours
Mindlessly refreshing social media and scrolling
Avoidance activities such as sorting and organizing email rather than answering it
Excessive shopping or online browsing

Quadrant 4 is the quadrant of excess and immediate gratification that ultimately leaves feeling unfulfilled. That doesn't mean leisure time is blocked off completely. Eisenhower himself was a well-known bridge player — even playing nightly up to D-Day — and was famously criticized for his many golf trips while in office. The key is that these activities balance the many stressful aspects of being a political leader. However, if not intentional about it, the way of spending downtime can drain energy, passion, and creativity.

A recent study published in the Journal of Applied Psychology found that how employees spend their off-job leisure time is a strong predictor of how much energy and positivity they expressed the following workday. Employees who engaged in self-mastery activities such as exercise or volunteering were more motivated the following day. Employees who relaxed with yoga, meditation, or listening to music approached the workday more calmly.

Employees who engaged in distraction activities to avoid or ignore problems, like watching excessive TV, did show a renewed positivity the following workday. However, with continued use of distraction, their moods and motivation worsened as the week continued. In other words, distraction in moderation was ok, but habitual distraction resulted in less work satisfaction overall.

1.2 OBJECTIVES

The main objective of the project is to encapsulate the above-mentioned time managing strategies into a single application.

- Implement a smaller version of Eisenhower Matrix where the tasks/projects can be assigned a priority level out of the three: High(Important & Urgent), Medium(Important), Low(Urgent)
- Implement a Pomodoro timer where the tasks/projects can be assigned a Pomodoro cycle

1.3 ORGANIZATION OF THE REPORT

The remainder of this report is laid out as follows. In Chapter 2, an in-depth examination of the many literature works on this subject is discussed. The technology used in the project is discussed in Chapter 3. The design methodology is discussed in Chapter 4 and the result, i.e, the final product is shown in Chapter 5. Chapter 6 concludes with some thoughts on the work's future scope.

CHAPTER 2
LITERATURE SURVEY

Students complain of not having enough time in the day, and that their academic life falling sideways as a result. They have shared some of the serious consequences they have encountered due to poor time management.

Procrastination is the most obvious result of unorganised time management. Students who don't have control over the time end up letting tasks sit until the last minute – and then feel a lot of stress when trying to play catch up. Letting too many tasks sit, might lead to missing the deadlines entirely. Students having trouble balancing the demands of the course load and test prep with internships or clubs, likely finds both academics and test performance taking a hit. Not keeping up with the studies will lead to lesser engagement as the new material is not understood creating a bad cycle of academic performance.

Also constantly staying up late to finish assignments that should've been done days ago will cut into the sleeping time. Sleep is significant for the health of college students, but poor time management can move sleep to the bottom of the list of priorities. Many college students skip their breakfast as they overslept trying to meet an assignment deadline. Skipping the meal would typically lead them to reach for snacks and most of them end up reaching for the unhealthy junk food option.

Poor time management also leads to getting late for classes, professor's office hours and scheduled appointments. Thereby making the students look unprofessional due to a lack of punctuality. People's impression matters and consistently being late would easily be noticed.

Not only the students but also the professionals also struggle with time management. Problems like having to work extra hours to hit the deadlines and cancelling meetings with clients as it is down to the wire for one project or another are a few among them. Inefficient time management can cause serious burnouts that are tough to get out from.

People confuse having a jam-packed schedule as good time management. It just means they are extremely busy. Doing only the work or career aspect of life doesn't necessarily mean that whole other aspects of life are being efficiently addressed.

But professionals do not retrospect on the effects of poor management of time while struggling to meet deadlines and get everything done. There are not only personal costs but larger overall costs to the entire organization. When staff miss deadlines or are not very

productive, this costs the organization money. It can also lead to losing clients and entire departments becoming less productive.

The biggest single time-management problem for most people is the lack of priorities and focus. It's even worse when there aren't clear goals to follow. If an employee allows themselves to be distracted by numerous unimportant tasks, they end up wasting a lot of time. For example, getting distracted by every new email or starting one task after another without finishing anything.

Most workers have more work to do than could be completed in one day or even one week so they structure their tasks in resource management software tools. The feeling of overwhelming arises due to inefficient time management arising out of vague priorities. To counter it would be to work on the most important task at your hand. The sense of achievement feels as if the deadline has been met. But it will only mean leaping straight into the next crisis to meet another deadline. Eventually, there won't be enough time to get ahead of the curve. Getting proactive about the work can greatly help meet all the deadlines early. Sometimes the issue is that an employee won't delegate the least urgent and significant tasks. Managers, in particular, can be guilty of this. Managers must delegate when appropriate. Then can focus on the tasks of most value. This results in higher quality work from everyone involved.

Lack of motivation is another adverse effect of poor time management as people tend to be aware that they are not completing work but the cluster of piled-up unattended tasks urges them to slide into the comfort of bed sheets or the instant dopamine by attaining virtual milestones in video games. This can have a huge effect on the emotional and mental state as well. Low motivation and morale can become a negative cycle at work. They often result in a bad performance, and therefore low motivation, and so on. One way of altering this cycle is through the use of time management resources or training.

Professionals often complain of having to spend less personal and family time leading to more stress, more tiredness, burnout and even affecting their health. Not only the bond with family but also the team morale could also be affected due to a cluttered to-do list. It can impact the team dynamics and affect how the team runs. This is never truer than in the case of the manager or team leader. Some of the negative effects can be:

- Missed deadlines show that there is a serious lack of respect and professionalism. This will surely impact the career of the person in question.

- It can also act as a demotivator for the team because no one wants to put the work in only to fail because of someone else. This effect is worsened if the person with poor time management is the manager.
- More widely, unprofessionalism can lower the good reputation of the organization. Any client would be upset if you missed a deadline related to their project.
- When others feel like they can't achieve anything because of one team member, motivation is lowered. So are satisfaction and completion rates. This all contributes to lower energy levels.
- When people have little energy, they often start performing worse. The risk of this is that a manager responds by piling on the pressure to perform. However, instead of rectifying the problem, this can make it worse.
- Deadlines being missed can come at a huge cost for the company.
- Without proper time management training, deadlines may be missed time and time again. As an example, if you create products for a shipment leaving the next day, the shipment goes whether or not you've filled it up. The cost of this can be massive. These conflicts can make your client relationships suffer as well as you having to lose out on profit.
- Even just one employee with bad time management can cause client relationships to break down.
- When you don't have good time management, there is a chance that someone in another department is encountering a huge backlog due to your actions (or inaction). For example, one manager giving a payroll administrator the payment information at the last minute can cause the administrator a ton of stress. All because one manager failed to manage their time well.
- Over time, this sort of behaviour is sure to cause resentment. This can result in repeated conflicts between employees. Poor time management and stress are strongly linked. That's why time management challenges can affect the health and emotions of employees. This could lead to further conflicts as well as low mood and lack of motivation.
- One of the biggest time management problems is that when the team produce low-quality work at the last minute, the clients are sure to notice. They will no longer be happy with the service and may choose to go with another provider. This loses the company money but also tarnishes its reputation. It's even worse when there are

contracts with the client, but it includes a service level agreement. If the agreement is not met, a fine must be paid.

CHAPTER 3
TECHNOLOGY

3.1 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.

3.1.1 Why Flutter?

- provides much faster development across different operating systems.
- provides an interactive UI, which allows app developers to design the applications as per user requirements.
- used to build apps for iOS, Android, and Windows.
- has everything necessary for seamless cross-platform app development.
- allows you to design robust, high-performing, and scalable mobile apps for all operating systems using a single codebase.
- Provides the best icons, typography, and scrolling.
- Based on Dart, the in-house language of Google, it is easier for developers to learn this language if they are familiar with Java and JavaScript.
- receives brilliant community support with tutorial guidelines. Regardless of experience, mobile application developers can start using this platform easily.

3.1.2 Implementing the Front End

Splash Screen

The Animated Splash screen is used as a startup screen in a Flutter application. More or less all applications use them generally to show the logo of the institution and its creators' awareness. This although it holds no functionality but can be great to increase product awareness and promotion.

The dependency for the Animated Splash Screen is added to the *pubspec.yaml* file. The dependency is then imported to the *main.dart* file. The custom GIF is added to the asset folder as well as pubspec.yaml, for use in the application. The page is to be loaded just after the splash screen is provided to the nextScreen attribute.

Bottom Navigation Bar

A material widget that's displayed at the bottom of an app for selecting among a small number of views, typically between three and five.

The bottom navigation bar consists of multiple items in the form of text labels, icons, or both, laid out on top of a piece of material. It provides quick navigation between the top-level views of an app. For larger screens, side navigation may be a better fit.

The bottom Navigation bar is used to seamlessly traverse across various pages like the Timer page, Project page, Statistics Page and the Settings page.

Stream Builder

A controller with the stream it controls (refer to *Fig 3.1* for the Stream Builder Architecture). This controller allows sending data, errors and done events on its stream. This class can be used to create a simple stream that others can listen to and push events to that stream. It's possible to check whether the stream is paused or not and whether it has subscribers or not, as well as get a callback when either of these changes.

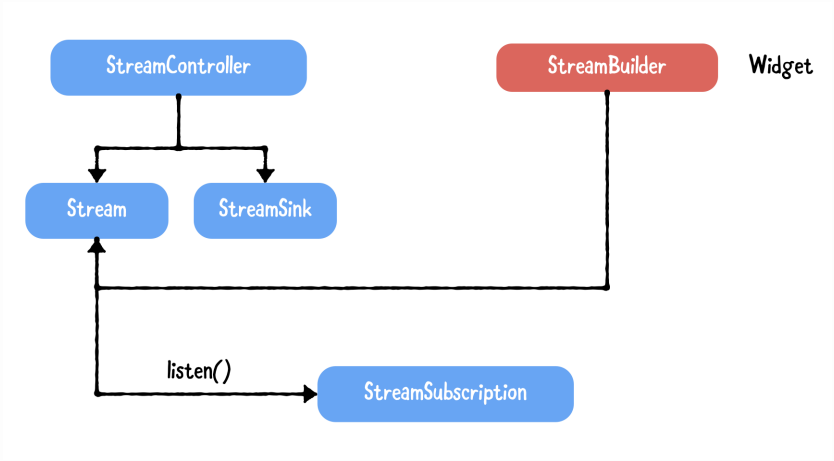


Fig 3.1 Stream Builder Architecture

BlocProvider

Have you ever started an application that crashed for no reason? Challenges can occur in many ways; for example, when you click a button, it freezes for five to ten seconds without displaying a message in the user interface. Also, perhaps an application that may only work with an internet connection doesn't work correctly when an internet connection isn't available.

Then, suppose your answer to the above is yes. In that case, you will know how frustrating it is for a user not to receive a visual representation or any interaction within the application.

The primary problem is that the code is not clear, organized, maintainable, or testable. The app developers generally concentrate on building it for the short term rather than maintaining it for a long time.

Programming is all about how you and your team need to write code using an architecture that features the best, accommodating every feature and scale over time.

The best way to keep your code organized, clean, and maintainable in Flutter is to have a component that can mediate what the user sees and its logic. This component I'm referring to is Bloc (Business Logic Component.)

BlocProvider is a Flutter widget that uses BlocProvider.of <T> to supply a bloc to its children. It’s a dependency injection (DI) widget that lets you deliver a single instance of a bloc to many widgets within a subtree. BlocProvider should be used to create new blocs that will be available to the remainder of the subtree in most cases as in Fig 3.2. Because BlocProvider is in charge of generating the bloc in this situation, it will also be in charge of closing it.

You will almost always want to put the BlocProvider above the MaterialApp so that it will be available everywhere in your application.

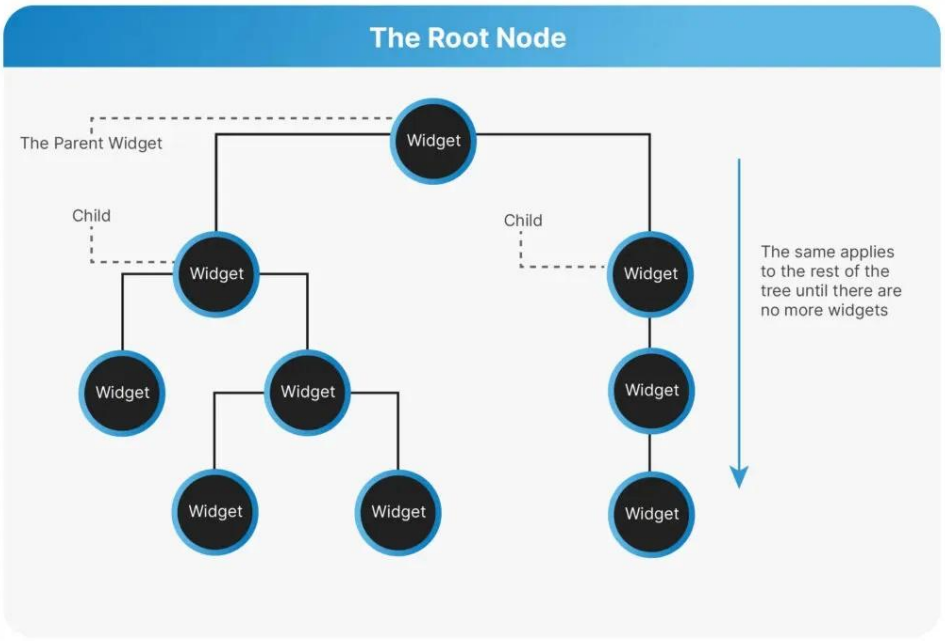


Fig 3.2 The Widget Injection in BlocProvider

Bloc Builder

BlocBuilder is a Flutter widget which requires a bloc and a builder function. BlocBuilder handles building the widget in response to new states. BlocBuilder is very similar to StreamBuilder but has a more simple API to reduce the amount of boilerplate code needed. The builder function will potentially be called many times and should be a pure function that returns a widget in response to the state.

Multi Bloc Provider

Merges multiple BlocProvider widgets into one widget tree. MultiBlocProvider improves the readability and eliminates the need to nest multiple BlocProviders. MultiBlocProvider converts the BlocProvider list into a tree of nested BlocProvider widgets. As a result, the only advantage of using MultiBlocProvider has improved readability due to the reduction in nesting and boilerplate.

3.2 HIVE

Hive is a quick, lightweight, NoSQL database for flutter and dart applications. Hive has a straightforward key-value database without numerous relations. It is an offline database. Hive bolsters all stages upheld by Flutter.

3.2.1 Why Hive?

- Easy to use, just like Dart maps. You don't have to write long and complicated queries to query data.
- High flexibility. You can change your data structure easily because Hive doesn't use tables like SQLite.
- No native dependencies
- Fast and can handle a large amount of data.
- Supports primitives (string, number, list, map, etc) and Dart objects (with the help of adapters)

3.2.2 Implementing the Back End

For using Hive, add the following dependencies to the pubspec.yaml file.

dependencies:

hive: ^[version]

hive_flutter: ^[version]

dev_dependencies:

hive_generator: ^[version]

build_runner: ^[version]

await Hive.initFlutter()

Initializes Hive with a valid directory in your app files. You can also provide a subdirectory

Box Collections

BoxCollections are a set of boxes which can be similarly used as normal boxes, except that they dramatically improve speed on the web. They support opening and closing all boxes of a collection at once and more efficiently store data in indexed DB on the web.

Aside, they also expose Transactions which can be used to speed up tremendous numbers of database transactions on the web.

On dart.io platforms, there is no performance gain by BoxCollections or Transactions. Only BoxCollections might be useful for some box hierarchy and development experience.

Store Objects

Hive not only supports primitives, lists and maps but also any Dart object you like. You need to generate a type adapter before you can store objects.

3.3 INVISION

InVision is the online whiteboard and productivity platform. A real-time workspace for teams to collaborate better, align faster and move forward together.

3.3.1 Why InVision?

- Good quality presentation
- Control of design
- Real-time to-do lists
- Mobile prototyping
- Version History and syncing
- Hover States
- Live Share

3.4 BIT.AI

Bit.ai is a powerful document collaboration platform for teams regardless of where they are located. Collaborate together from anywhere in the world. Bit documents are collaborative, integrated, smart and one of the most powerful ways to communicate internally and externally.

3.4.1 Why Bit.ai?

- Organize Work with Fluid Workspaces
- Communicate More Intelligently with Rich Integrations
- Study User Engagement with Built-in Document Tracking
- Save Valuable Time Searching Across All Your Content
- Share in Multiple Ways

3.5 GITHUB

GitHub, Inc. is a provider of Internet hosting for software development and version control using Git. It offers the distributed version control and source code management functionality of Git, plus its own features.

3.5.1 Why Github?

- Easy to contribute to open source projects
- Documentation
- Showcase of the work
- Track changes in the code across versions
- Integration options

CHAPTER 4
METHODOLOGY

4.1 AN OVERVIEW

While building a Time Management application that is built on the concepts of Pomodoro Timer and Eisenhower Matrix, the following problems arose:

- Uniting the concepts with a good blend avoids any conflict
- Delivering a seamless and efficient User Interface
- Employing a database that ensured secure, intact and sorted storage of data

The approach for exploring the solutions to the given problem was both quantitative and qualitative. Qualitatively numerous documents, articles, concept videos and tutorials were referred to. The quantitative approach was via experiment. The best approach was selected by the method of trial and error where the assimilated concepts were implemented and what seemed best was embroidered onto the project.

In the earlier version of the application, Flutter was used to implement the front end and Django was used for the backend. Flutter seemed the best choice as it continued to be used in the final version. But the case was not the same with the backend as various other tools were explored in choosing what fitted the project's need.

The tools explored for the application:

- Flutter (Front End)
- Django (Back End)
- Firebase (Back End)
- Hive (Back End)

The methodology of trial and error helped in exploring not only the technical aspects of developing software but also helped in delving into and experiencing the various facets of project management.

4.2 SOFTWARE DEVELOPMENT LIFE CYCLE

The Software Development Life cycle or SDLC is a thorough process to manufacture software that systematically ensures that the quality and the correctness of the software are according to the standards set by the company and the industry. Fig 4.1 gives an overview of the entire SDLC.

This life cycle aims to ensure that the software that is being manufactured in the industry is properly falling under the expectation bracket of the customer. One thing it ensures is that the development process of the software stays inside the already agreed-upon cost and time frame. The software development life cycle is a compilation of ideas and plans that explain the whole process of planning, building, and maintaining software from initiation to completion. And it's not a stagnant process either.

At every step of the way, various processes and deliverables need to be worked on, to make sure that the next phase of the lifecycle goes smoothly.

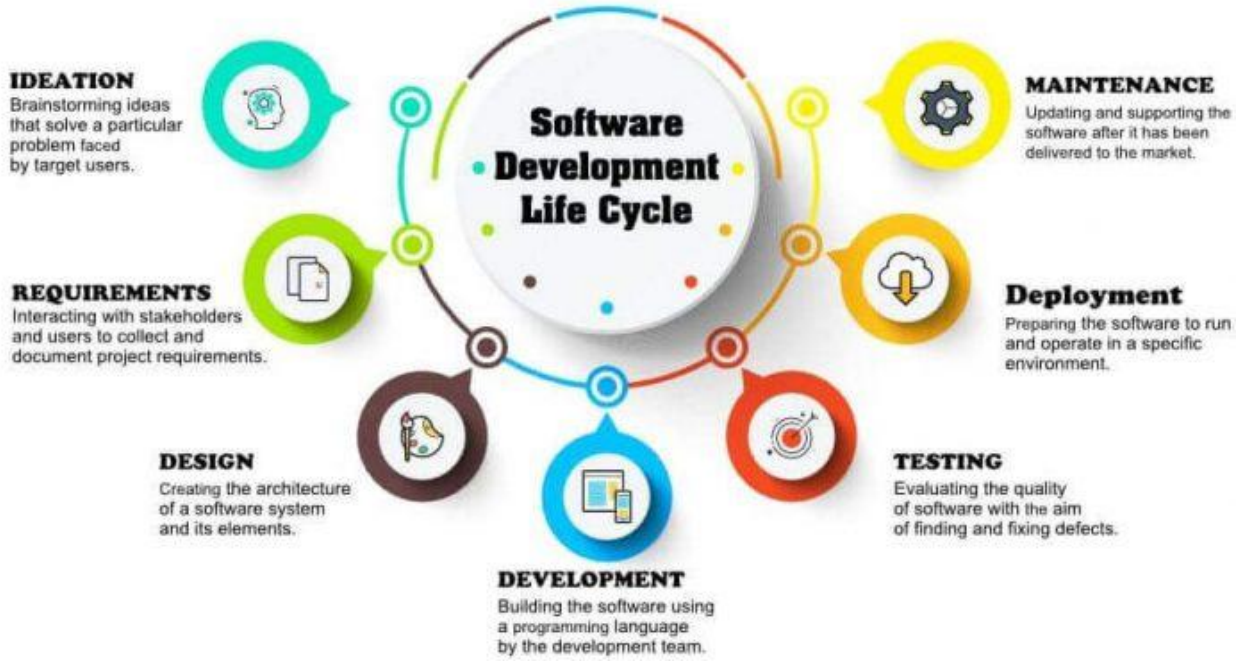


Fig 4.1 The Software Development Life Cycle

4.2.1 Why SDLC?

- A common vocabulary for each step
- Defined communication channels between development teams and stakeholders
- Clear roles and responsibilities among developers, designers, business analysts, and project managers
- Clearly-defined inputs and outputs from one step to the next
- A deterministic “definition of done” that can be used to confirm whether a step is truly complete

4.3 AGILE MODEL

The agile SDLC model is a combination of iterative and incremental process models with a focus on process adaptability and customer satisfaction by rapid delivery of working software products. Agile Methods break the product into small incremental builds. These builds are provided in iterations as shown in fig 4.2. Each iteration typically lasts from about one to three weeks. Every iteration involves cross-functional teams working simultaneously on various areas like −

- Planning
- Requirements Analysis
- Design
- Implementation
- Unit Testing
- Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

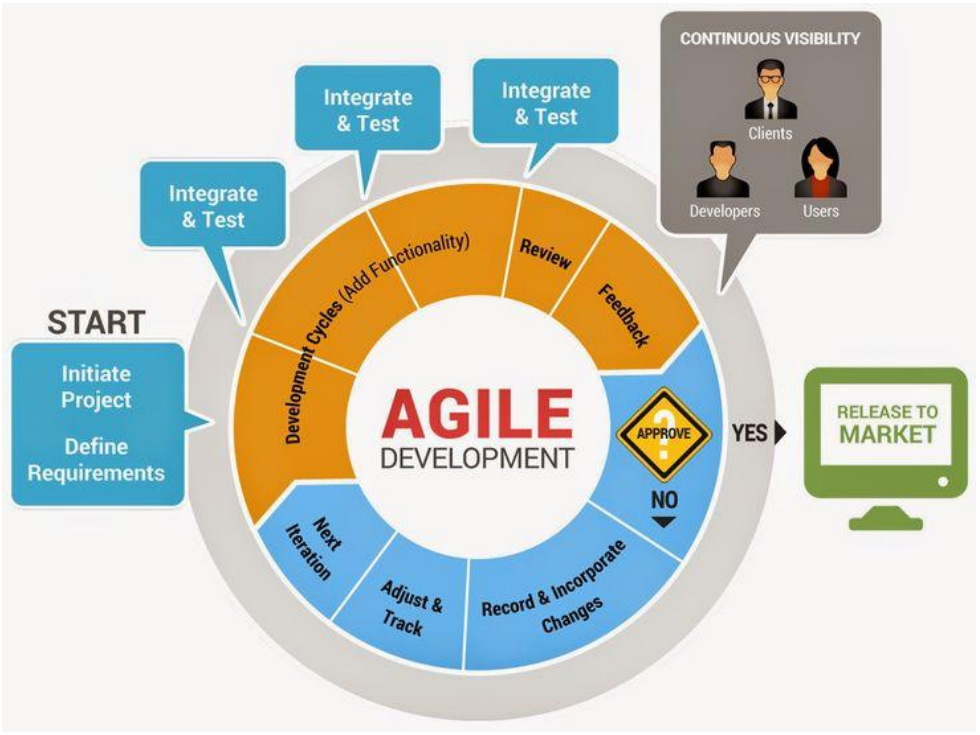


Fig 4.2 Agile SDLC model

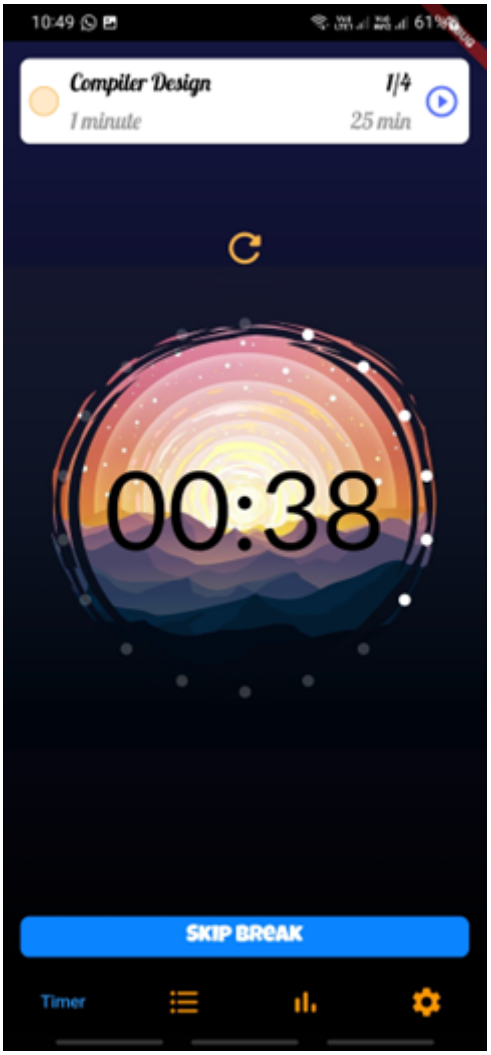
4.3.1 Why Agile Model?

- Waste fewer resources because of work on up-to-date tasks
- Better adaptability to change and respond faster
- Faster turnaround times
- Detect and fix issues and defects faster
- Immediate feedback (which also improves team morale)
- Developers can improve their skills based on QA feedback
- No need of worrying about premature optimization
- Experiment and test ideas because their costs are low

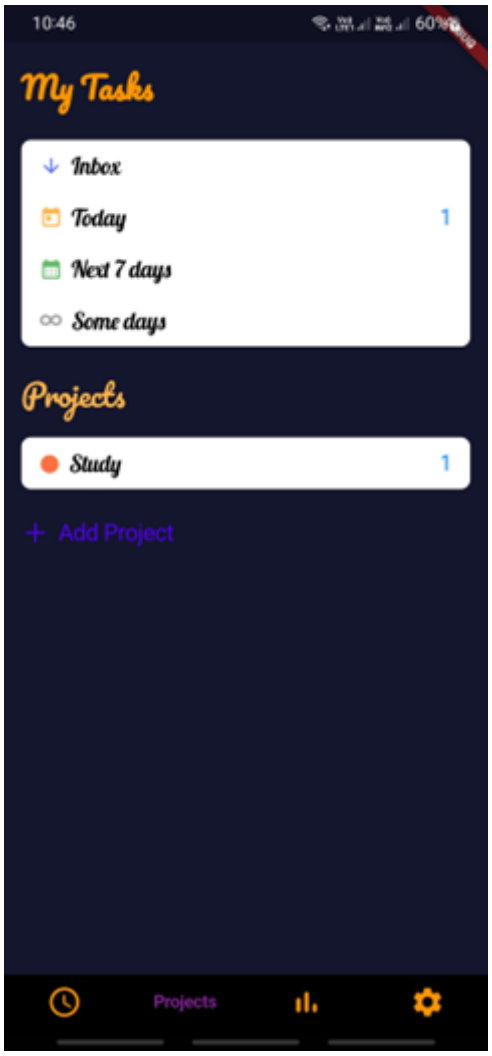
CHAPTER 5

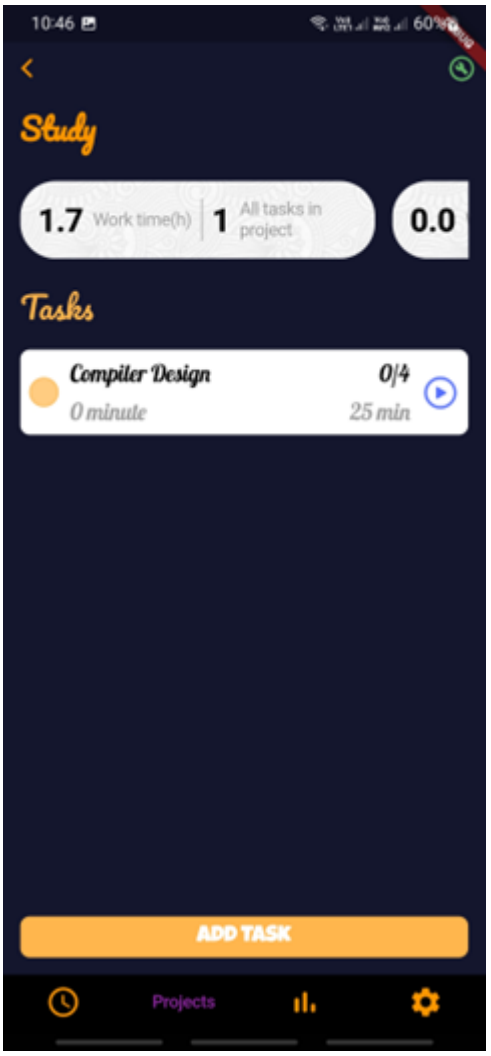
FINAL PRODUCT

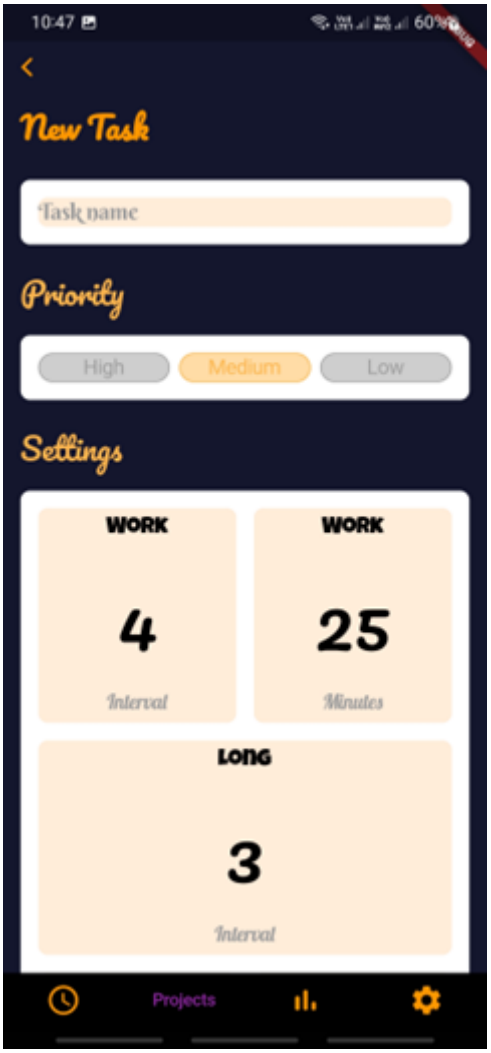
Timer


10:49
Compiler Design
1/4
1 minute
25 min
00:38
SKIP BREAK
Timer

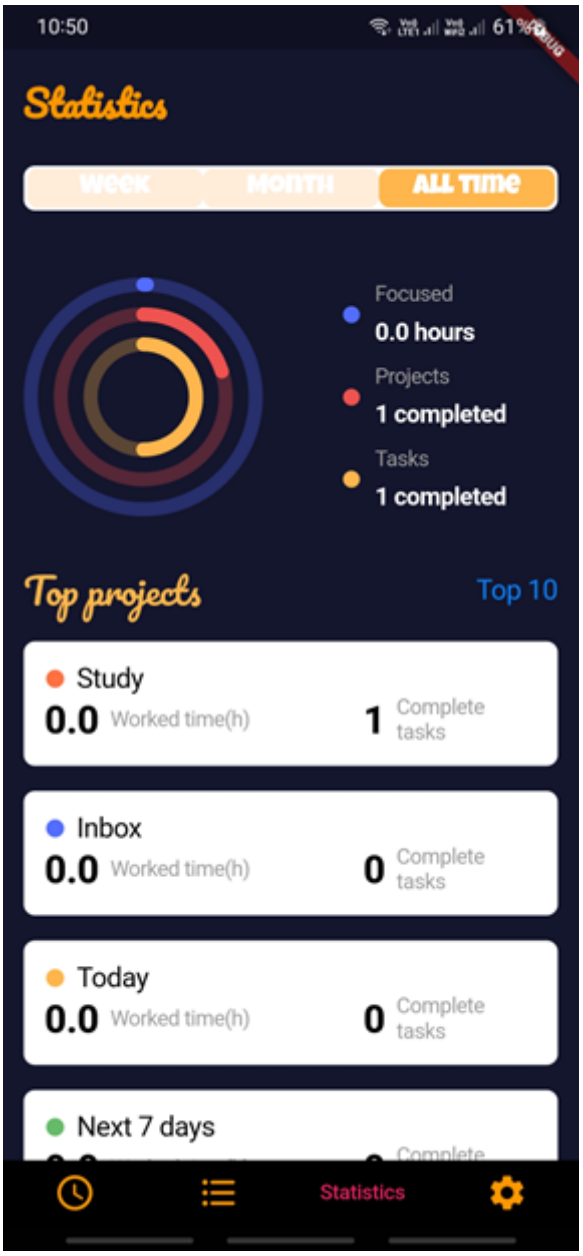
Projects


10:46
My Tasks
Inbox
Today
1
Next 7 days
Some days
Projects
Study
1
Add Project
Projects

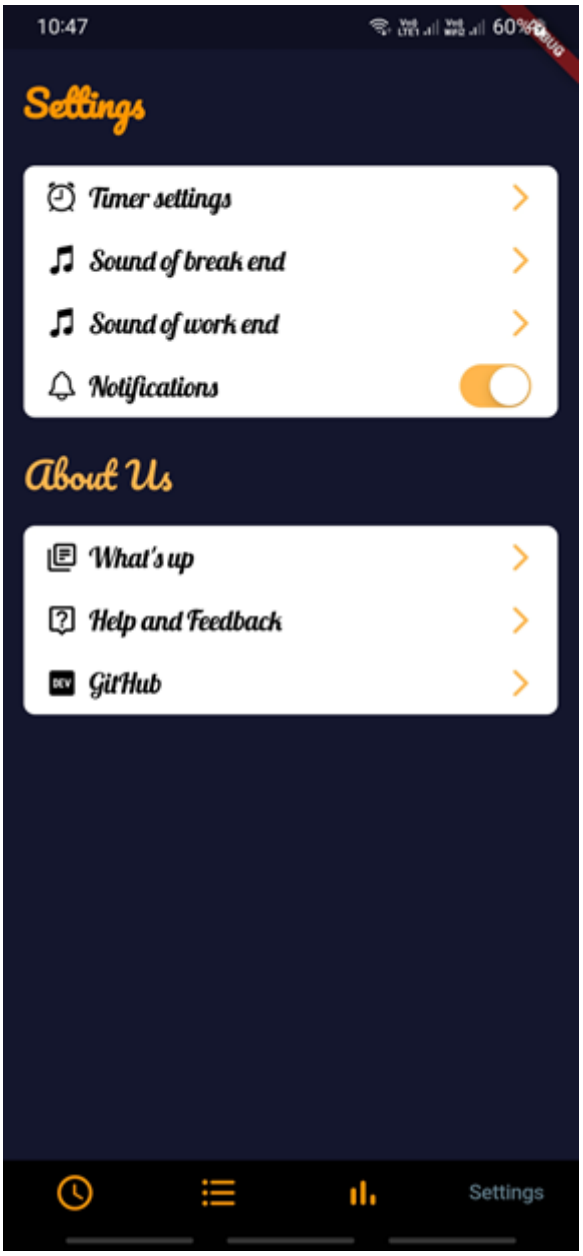

10:46
Study
1.7
Work time(h)
1
All tasks in project
0.0
Tasks
Compiler Design
0/4
0 minute
25 min
ADD TASK
Projects


10:47
New Task
Task name
Priority
High
Medium
Low
Settings
WORK
4
Interval
WORK
25
Minutes
LONG
3
Interval
Projects

Statistics


10:50
Statistics
Week
Month
All time
Focused
0.0 hours
Projects
1 completed
Tasks
1 completed
Top projects
Top 10
Study
0.0 Worked time(h)
1 Complete tasks
Inbox
0.0 Worked time(h)
0 Complete tasks
Today
0.0 Worked time(h)
0 Complete tasks
Next 7 days
Statistics

Settings


10:47
60%
Settings
Timer settings
Sound of break end
Sound of work end
Notifications
About Us
What's up
Help and Feedback
GitHub
Settings

CHAPTER 6

PROGRAMME OUTCOMES

Sl No.	Programme Outcome	Explain how attained through mini project
1	Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialisation to the solution of complex engineering problems.	Applied knowledge of software engineering is required to plan and build functioning software and databases for the project.
2	Problem analysis: Identify, form, date, review research literature, and analyse complex engineering problems reaching substantiated conclusions using fest principles of mathematics, natural sciences, and engineering sciences.	Problems from the daily lives of people have been identified, analysed and an approach to solve the problems has been formulated using the principles of computer science.
3	Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.	The needs for the solution have been specified in the SRS document and the appropriate design details have been documented.
4	Conduct investigations of complex problems: Use research- based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.	Several sources have been referred on the efficiency of time management, in accordance with the problem at hand.
5	Modern tool usage: Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.	Modern software tools such as IDE, Git and Github have been used during the course of the project.
6	The engineer and the society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.	The application we developed is for everyone who is struggling to manage time and maintain a balance between personal and professional time.

7	Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental context and demonstrate the knowledge of, and need for sustainable development.	The project was tested to be efficient such that it will not consume too much power. It has support for older devices, so that it does not encourage people to buy new devices.
8	Ethics: Apply ethical principles and commit professional ethics and responsibilities and norms of the engineering practice.	During the project we have cited all the references from where we have acquired materials and guidance.
9	Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.	All the members of the team have contributed effectively to the project.
10	Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large,such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.	Effective group communication and communication with the supervisors have been maintained throughout the project. Regular project presentations were held. Design and requirements documents were scrutinised regularly by the supervisors and improvements were constantly made to the same.
11	Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	Members in the project choose the infrastructure in accordance with the budget of the project.
12	Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.	Members were able to refer various resources from the internet and research new technologies to solve problems.

CHAPTER 7
CONCLUSION AND FUTURE SCOPE

The application “Trail” was developed using Flutter and Hive. Users can view and control the work/break timer based on Pomodoro cycle, manage projects and tasks, view their progress through statistics. The user can manage notification and other preferences through app settings.

The future scope for this project is providing user sign in (using Firebase instead of Hive) and cloud backup, so that users can retrieve their projects and progress in case of data failure in local device. It can also be improved to sync the data between multiple devices. Other possible improvements include a more detailed statistics report, project tag, etc.

REFERENCES

[1] https://www.themuse.com/advice/take-it-from-someone-who-hates-productivity-hacksthe-pomodoro-technique-actually-works

[2] https://todoist.com/productivity-methods/eisenhower-matrix

[3] https://docs.flutter.dev/

[4] https://docs.hivedb.dev/

[5] https://bit.ai/about

[6] https://www.invisionapp.com/company

[7] https://monday.com/p/about/

[8] https://docs.github.com/en