LONDON METROPOLITAN UNIVERSITY

Final Year Project (Proposal)

Project Title: EasyLearn

CS6P05NP

Student Name: Kshitij Magar

Group: C3

London Met ID:21049983

Assignment Due Date:2023/09/01

Assignment Submission Date: 2023/09/01

Table of Contents

List of Figures

1. Introduction

1.1. Current Scenario

E-learning has completely changed how people learn. A digital environment that makes it easier to distribute educational materials and courses online is known as an e-learning platform. Since peoples always have mobile devices with them, accessing these platforms is incredibly simple for them. It enables learners to select topics and resources from a wide range. In addition to learning from their preferred tutor who is not located nearby, students can also learn from remote locations at their convenience. These platforms frequently have tools like forums and progress monitoring. They provide chances for guided instruction or self-directed learning. These platforms promote constant learning and skill development in today's digital world in schools, universities, businesses, and individuals.

With a strong dedication to enabling learners of all ages and backgrounds to set off on a voyage of exploration and personal development, I created my application, Easy learn. Students can select from a wide variety of programs and resources to suit their individual needs and interests. Additionally, they can talk to the tutors about their issues and creative concepts.

1.2. Problem Statement and Project as a Solution

Concepts in online courses can occasionally be challenging to comprehend. They have to spend a lot of time looking for videos that are appropriate for them to understand.

Finding the greatest instructor nearby will be appropriate for students who want to learn or need a tutor physically. In that scenario, it would be preferable if they were given a list of tutors nearby who had the appropriate rates and schedules.

It would enable learners to search for physical classes or discussion sessions, providing them with the option to engage in face-to-face interactions with tutors or peers. This innovative approach recognizes that some learners may benefit from real-world interactions to clarify concepts or ask questions.

1.3.Related projects

There are projects that are in the market which is similar to mine. Some of them are listed below.

a. Coursera

With the help of more than 275 top institutions and businesses, Coursera is a global platform for online learning and career development that offers flexible, inexpensive, and job-relevant online learning opportunities to people and organizations all over the world. The platform provides a variety of learning options, such as practical projects, classes, credentials that are employable, and degree programs. Daphne Koller and Andrew Ng started Coursera in 2012 with the intention of giving students all over the world learning experiences that would change their lives. Coursera obtained B Corp designation in February 2021, which implies that the platform has a legal obligation to lower obstacles to world-class education for everyone in addition to its stockholders. (coursera , 2022)

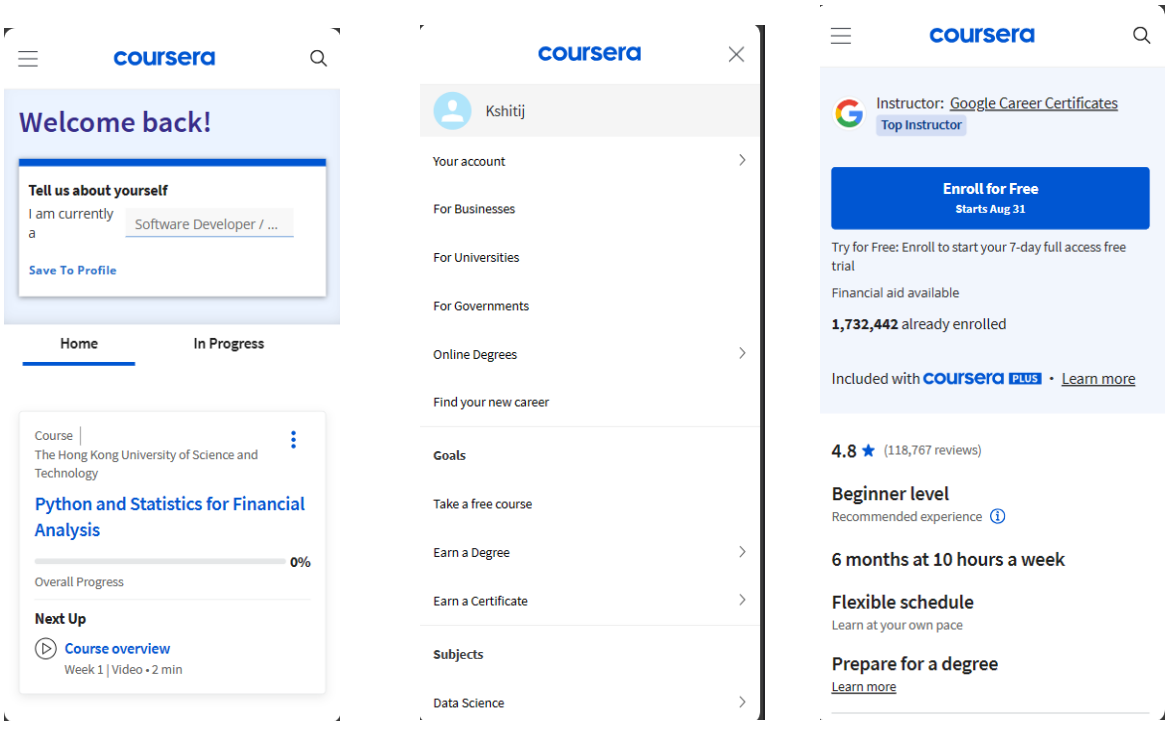

Figure 1 Courseera

b. My Second Teacher

MySecondTeacher is a portal for education that was created in Nepal and has won numerous honors. All throughout the world, parents, instructors, and students use it. The platform administers assignments, offers online lectures and resources, and gives students immediate access to information about their learning habits. This enables educators and parents to spot potential improvement areas and act quickly to solve them. The platform's Chatroom feature makes it possible for students, teachers, parents, and school administrators to work together effectively so that everyone is aware of the student's development and can offer support as needed. This enables individualized attention and prompt interventions, which improve academic results. (mysecondteacher, 2020)

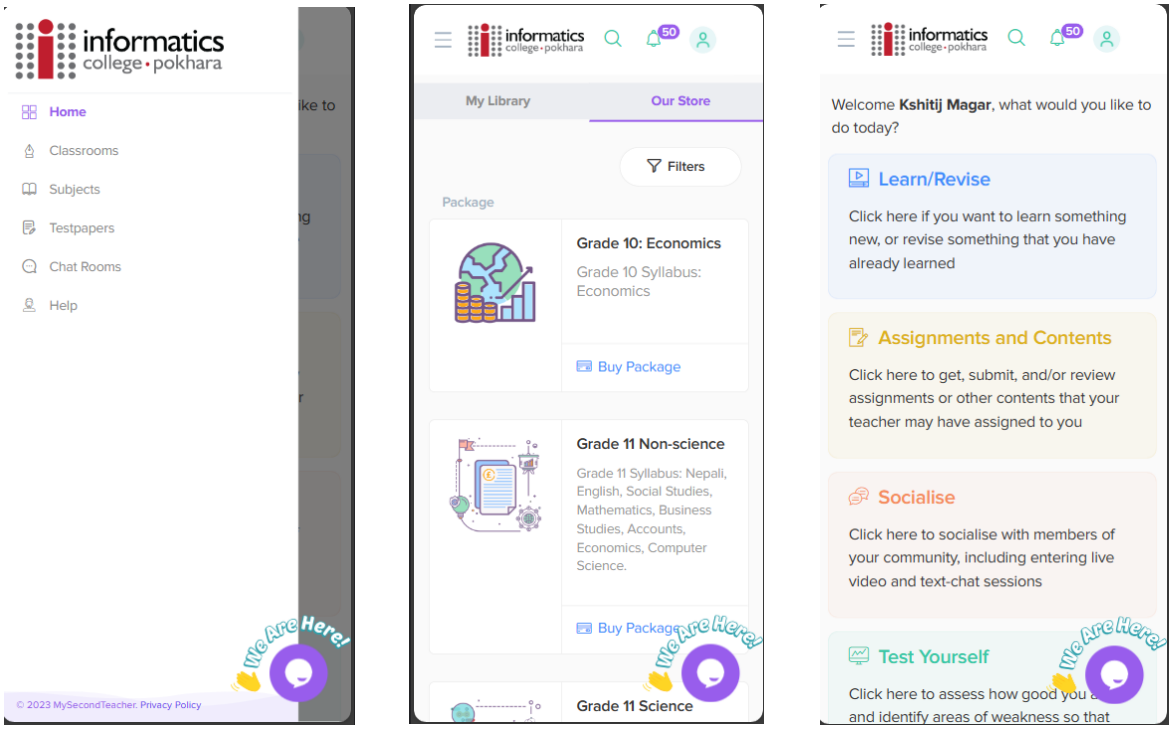

Figure 2My Second Teacher

c. Ottish

OTTISH is an online learning platform with a huge selection of professionally created courses. The platform offers 24/7 live discussion groups with classmates and faculty as well as access to competent teachers with a thorough understanding of the subject. To learn and review ideas, you can practice chapter-based quizzes and complete tasks. Additionally, the platform provides live seminars where you may study ideas, practice problems, and receive prompt clarification of your doubts. High-quality, simple-to-follow video lectures are provided in the app, along with offline video lectures for learning. For last-minute revision, the portal offers key concepts and formulas in PDF format. With a reasonable payment schedule, you can study under the top professionals in the field. You will be informed each time there is an update to the course material because it is updated frequently. You can be sure you're learning from reliable sources because the content is extensive and interactive. (OTTISH, 2022)

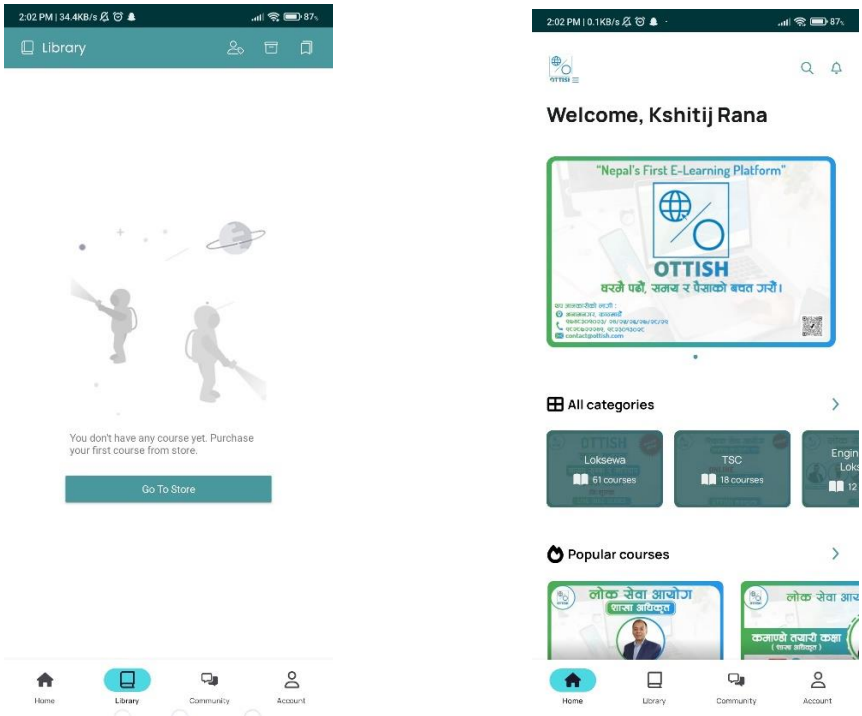

Figure 3 OTTISH

1.4.Comparison

Although this apps have provided e-learning apps in Nepal, they are not implemented properly.

Easy learn will allow users to learn skills of their own liking in their own pace. Learners can buy a course by paying with e-pay.

Occasionally, learners encounter difficulties with online learning and prefer a physical learning environment. To overcome this issue, Easy Learn introduces a distinctive feature that sets it apart from other e-learning platforms. This feature addresses a common challenge faced by online learners: the ability to seek additional support when they encounter difficulties in understanding the course material. They can search for an appointment with the tutors who are in the nearby location and also can join physical classes.

2. Aims and Objectives

The main aims of the project are given below:

a. Provide online learning platform which is flexible
b. Provide users with suitable course of their choice
c. To save time and efficiency
d. Provide details on physical classes
e. Allow users to book an appointment with the tutors

The main objectives of the project are given below:

a. Providing enhanced remote education facility
b. Offer a wide range of subjects and materials to cater to various interests and learning needs.
c. Provide learners the freedom to study at their own pace and convenience.
d. Selecting and applying methodologies that best suit the project's needs.
e. Providing education at a lower cost compared to alternative platforms.
f. Offering timely feedback to track progress and guide improvement.

3. Expected Outcomes and Deliverables

Project deliverables are the products you anticipate having at the end of your project when developing an application. In addition to reports, test results, budgets, technical documentation, builds, sprints, and other things, they can be internal or external. The advantages and outcomes you anticipate your project will bring about are known as expected outcomes. They might include more profits, better user experiences, higher levels of customer happiness, or higher levels of productivity. Expected outcomes cover more ground than your project deliverables and include the advantages and outcomes you anticipate from them. You must specify your project objectives in order to determine your projected results.

3.1. Expected Outcomes

The List of expected outcome at the end of the project are:

a. Mobile application:

   In the end while the project is completed, we will have a mobile application which have all the features which were mentioned above. It will be a E-Learning platform as well as physical classes finder. This app will be available on all android devices and ios devices.

   Features:

   - ❖ Users can choose and enrol in their desired courses.
   - ❖ Users can also appoint the best tutor available around them.
   - ❖ Users can also book physical courses around them.
   - ❖ Allow users to pay through e-pay.
   - ❖ User friendly UI

3.2. Deliverables

- ❖ Functioning Android e-learning app developed using android studio
- ❖ A PDF file including comprehensive research and testing reports

4. Methodology

Since methodology is one of the key success determinants and the management team's core skill, it is essential to avoid failure and minimize risks. When businesses began looking for more effective ways to streamline their operations and arrange their work into an organized, distinctive entity in the 1960s, methodology was first introduced. A project's successful planning, execution, control, and completion are guided by a set of organized tools and steps known as a methodology. It's a proven and structured way to manage projects from start to finish. (MyManagementGuide, 2020)

4.1.Waterfall Methodology

Software and system development is done sequentially using the waterfall model. It consists of a series of phases that are structured one after another and are dependent. The strategy places a focus on high level design or freezing requirements early in the development process. When project needs are well-defined and unlikely to change over time, this strategy is appropriate. This methodology has multiple phases like system analysis, software requirements, program design, coding, testing and operation. These phases advance in a linear fashion, and once they are finished, they cannot be changed. Although the model has features like precise documentation and architecture, it can also have drawbacks like high costs, challenges with adjustments, and potential misalignment with changing client expectations. (Casteren, 2017)

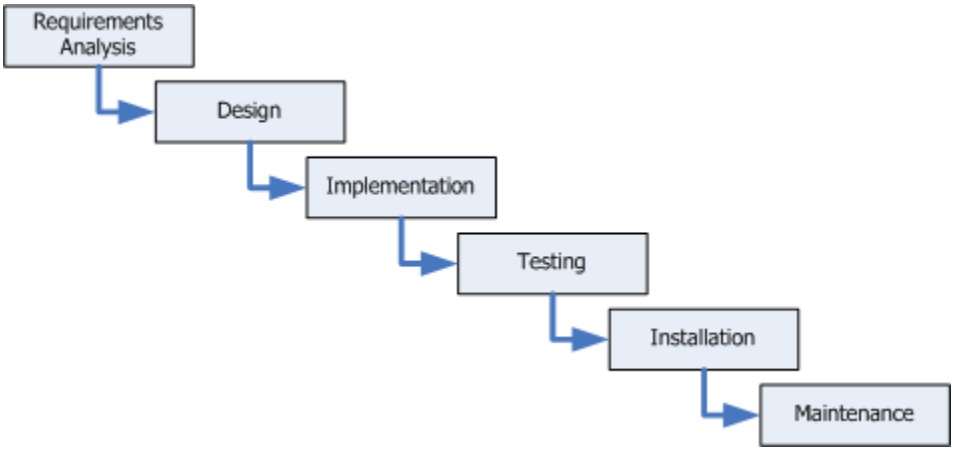

Figure 4 WaterFall Methodology

Advantages of Waterfall methodology:

1. Clear planning reduces issues and delays.
2. Documentation parallels coding for efficient development.
3. Thorough documentation aids communication and knowledge transfer.
4. Structured process provides tangible milestones for tracking.
5. Limited client involvement minimizes communication challenges.
6. Early troubleshooting improves user interface quality. (Half, 2014)

Disadvantages of waterfall methodology:

1. This model lacks flexibility for unexpected changes.
2. The waterfall's inflexibility can lead to wasted time and invalidated progress.
3. It postpones testing until project's late stages, risking extensive revisions and user problems.
4. Waterfall's strict phase completion can lead to longer project delivery times.
5. Waterfall suits small projects, struggles with larger ones due to rigid stages. (Gaille, 2020)

4.2. Agile Methodology

Agile methodology refers to a software development approach characterized by its flexibility and iterative nature. It stems from the "Agile Software Development manifesto," which prioritizes values like communication, adaptable processes, working software, customer involvement, and responsiveness to change. Agile methods encompass a flexible software development framework, involving iterative interactions from planning to deployment. It aims to minimize process overhead, embrace changes, and prioritize customer collaboration. These values underlie twelve guiding principles, including customer satisfaction, adaptive requirements, continuous delivery, collaboration, technical excellence, and regular reflection. Agile methods like TDD, FDD, and DSDM offer benefits such as early defect identification, quick releases, flexibility, and quality improvement, but also pose challenges like skill demands and potential time consumption. (Samar Al-Saqqa, 2020)

Figure 5 Agile Methodology

Advantages of Agile methodology:

1. Customer satisfaction through continuous valuable software delivery.
2. Embracing changing requirements for competitive advantage.
3. Frequent delivery of working software enhances project visibility.

4. Collaboration between business and developers drives success.
5. Adaptability and self-improvement foster efficiency and effective results.

Disadvantages of Agile methodology:

1. Agile's flexibility makes predictability challenging, leading to uncertainty in time and resource estimates.
2. Constant communication demands more time and energy from all parties involved.
3. Agile's success requires full commitment from developers and clients, impacting project quality.
4. Agile's just-in-time approach often results in inadequate documentation, causing misunderstandings later.
5. Agile's loose structure increases the risk of projects going off track or expanding beyond the original scope. (Minnesota, 2022)

4.3.Rapid Application Development (RAD)

Rapid Application Development is an agile framework that is primarily concerned with rapid software product prototyping, iterative product upgrades based on user feedback, and ongoing product deliveries to the market. This model is comprised of four phases: Requirement planning, user design , Rapid construction and Cutover. RAD emerged as a response to the waterfall method and was introduced in 1991 by James Martin. RAD's strengths encompass leveraging software's adaptable nature for continuous improvement, breaking down projects into actionable tasks, and delivering functional products promptly. (ProductPlan, 2022)

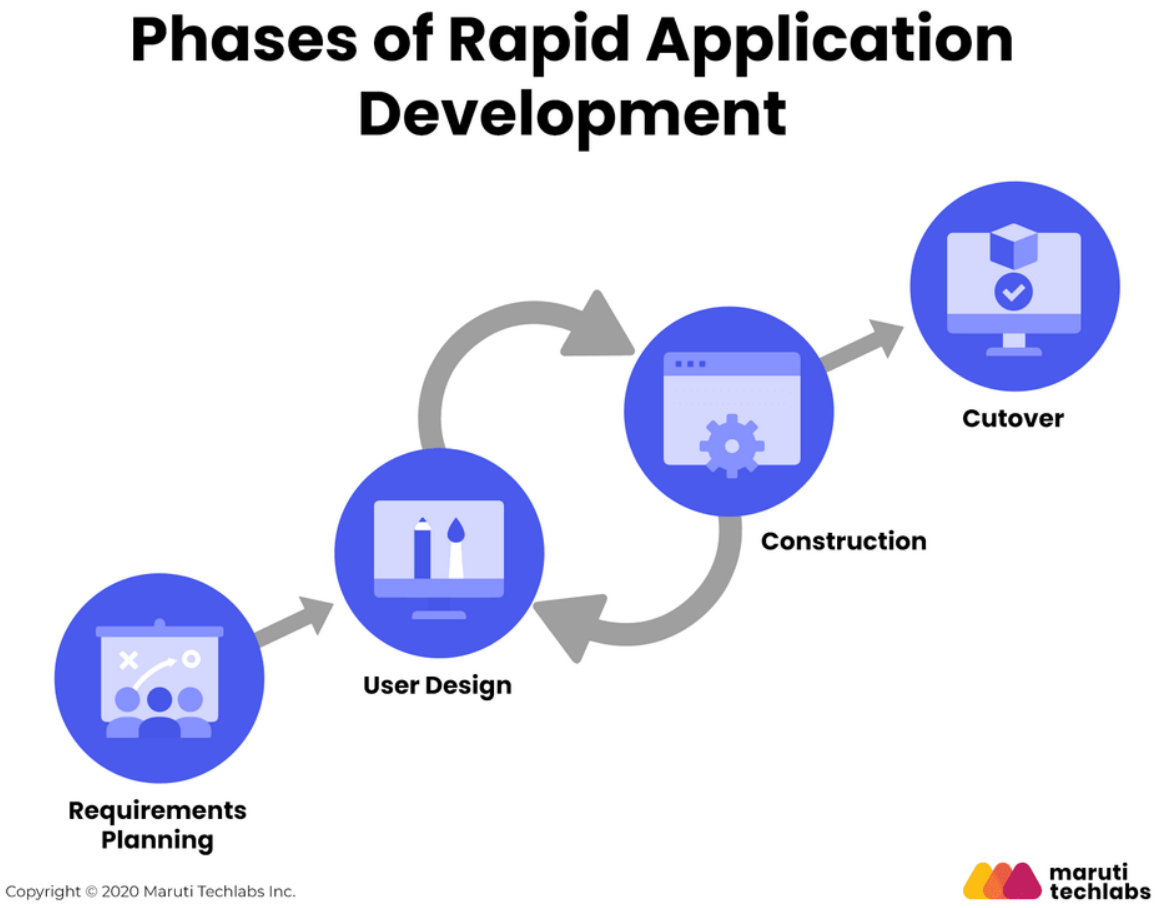

Figure 6 Rapid Application Development

Advantages of RAD:

1. Rapid prototyping shortens software development compared to traditional methods.
2. Product is used efficiently making RAD cost-effective.
3. Continuous feedback leads to software meeting client expectations.
4. Client appreciation motivates developers to work harder.
5. Flexible requirements in RAD ease risk management during development.

Disadvantages of RAD:

1. Highly skilled developers needed, foreseeing requirements and strong modeling skills.
2. Communication complexities arise with bigger teams in RAD.
3. Only modularizable projects suitable; not all software supports this.
4. Progress and problem are difficult to monitor as there is no record of what have been done.
5. Success depends on active client testing and feedback. (Makadia, 2023)

4.4.Iterative methodology

An important software method called iterative development, which is frequently combined with incremental development, splits complicated projects into manageable cycles. Through numerous repetitions of conceiving, implementing, and testing feature code, program functionality is gradually improved. A software application that is ready for deployment is produced via these cycles. Every iteration result in measurable outputs when it is aligned with agile approaches, such as time-boxed sprints. It thrives at complicated tasks and changes to meet changing needs. This method emphasizes the flexibility and reactivity of contemporary software development and exemplifies the industry's drive for excellence through planned cycles of design, development, and testing.

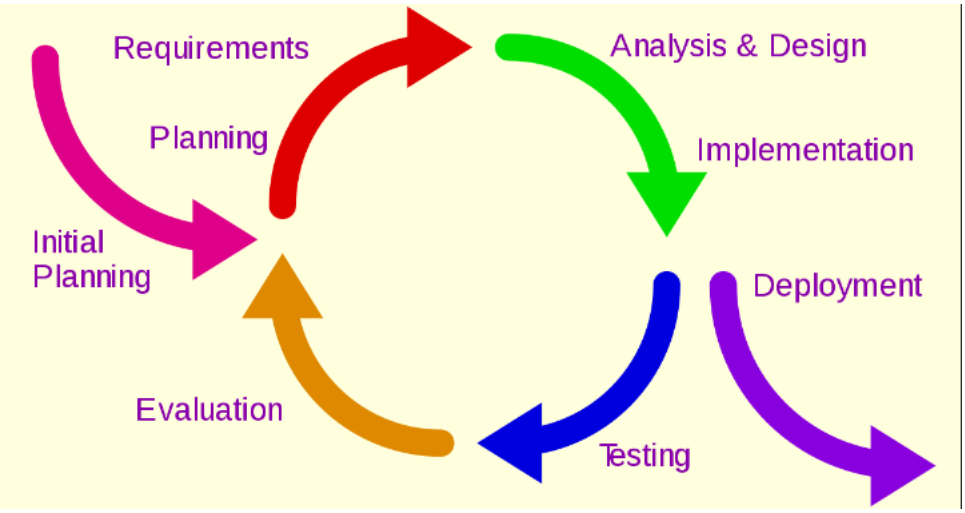

Figure 7 Iterative Methodology

Advantages of Iterative methodology:

1. Improves adaptation and flexibility by breaking down the SDLC into manageable cycles.
2. Takes consumer feedback into account, steadily enhancing products with each iteration.
3. Makes sure that progress is evaluated and keeps the quality and design of the product in mind.
4. Allows for the preservation of software iterations for versioning, enabling accountability and risk reduction.

5. Improves early issue detection, minimizing issues and delays during development. (Awati, 2023)

Disadvantage of iterative methodology:

1. Not appropriate for smaller tasks.
2. Needs more managerial participation.
3. Dependent on risk analysis for advancement.
4. Increment definition can include a description of the entire system.
5. Increasing demand for resources. (professionalqa, 2019)

4.5.Spiral methodology

Within Software Development Life Cycle (SDLC) techniques, the Spiral Model is an important strategy known for emphasizing effective risk management. With its numerous loops, it has a spiral-like appearance and can have any number of loops, depending on the project. A different stage of the software development process is covered by each loop. The project manager, who takes project risks into account, is responsible for making the key decision regarding the number of stages. The SDLC framework presented by Spiral Model is characterized by systematic iterations and covers the whole software development cycle. This iterative process covers everything from obtaining requirements, to analyzing them, to designing and implementing them, to testing and maintaining them. It encourages a thorough and dynamic approach to development.

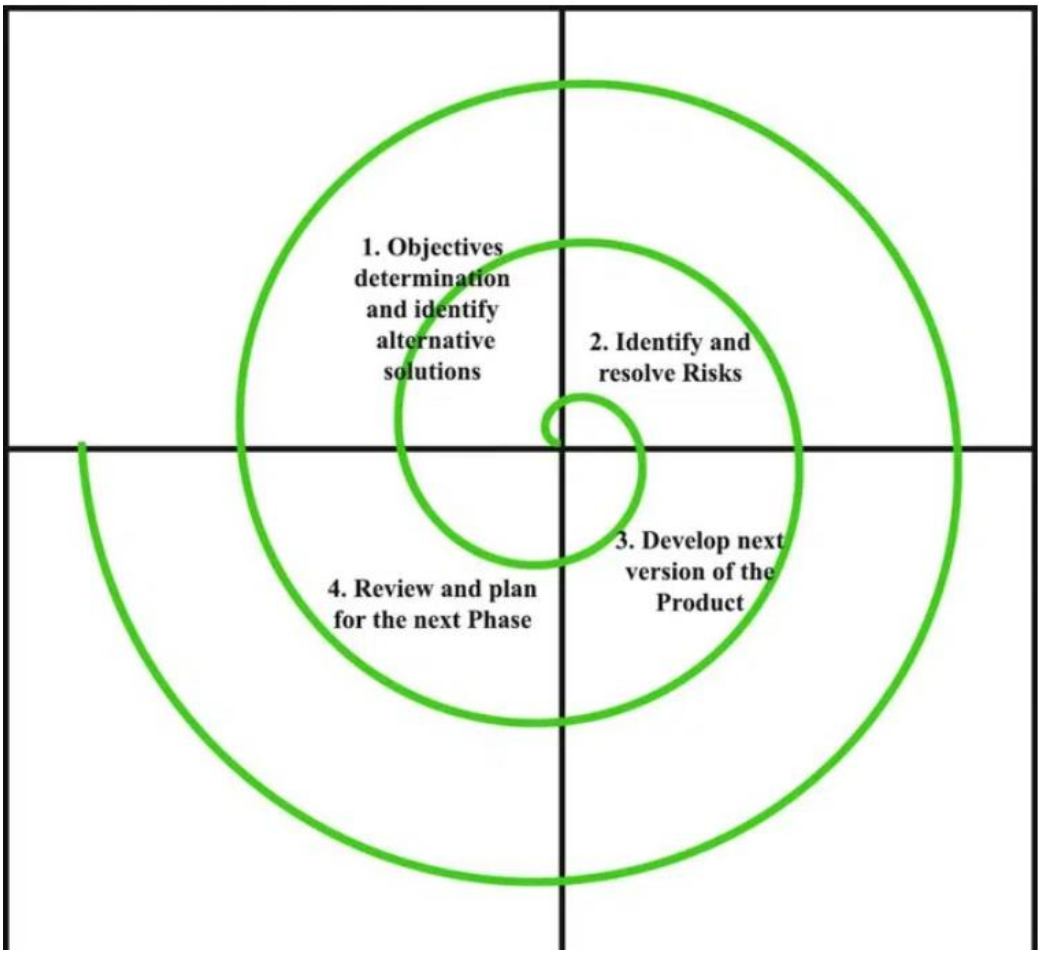

Figure 8 Spiral Methodology

Advantage of spiral methodology:

1. Because of the systematic approach, recommended for large and complex projects.
2. Flexibility to incorporate evolving requirements in later stages.
3. A focus on risk management reduces the impact of uncertainty.
4. Continual reviews improve team-customer connection.
5. Repeated iterations improve the dependability and quality of software.

Disadvantages of Spiral methodology:

1. Expensive and complicated, less suited for small projects.
2. Heavy reliance on expert risk analysis, challenging without experienced professionals.
3. Time estimation and project management are made more difficult by an ambiguous phase count.
4. The iterative nature can result in lengthy rounds of reviews and evaluations.
5. Requires a lot of resources because there has been a lot of preparation, risk analysis, and evaluation. (PAL, 2022)

4.6.Selected methodology – Iterative methodology

Iterative methodology in software development is a specific method within the SDLC. It starts with a basic version and gradually adds more complexity and features until achieving the complete final system. When talking about the iterative technique, the notion of incremental development is also frequently employed interchangeably. This refers to the gradual modifications made during the design and implementation of each new phase. It was used in projects as early as the 1950s.

In iterative methodology, after initial phase, a small handful of stages are repeated over and over, with each completion of the cycle incrementally improving and iterating on the software. Processes involved in iterative methodology are as follow:

1. Planning and Requirements: The initial part of any development project includes planning, creating specifications, determining the hardware and software needs, and getting ready for later stages.
2. Analysis and Design: After planning, analysis solidifies business logic, database models, and technical requirements for this project phase. Design work also takes place, specifying the technical elements like languages, data structures, and services needed to fulfill the analysis stage's demands.
3. Implementation: Once planning and analysis are complete, coding and actual implementation is now started. All preceding planning, specification, and design documents are translated into code and integrated into the initial project iteration.
4. Testing: After coding and implementation of the current build iteration, a series of tests is conducted to identify and rectify any potential bugs or issues that may have arisen.
5. Evaluation: Upon completion of preceding stages, a comprehensive assessment of the development process occurs. This enables the entire team, including clients and external parties, to review the project's status, required improvements, potential changes, and more. (Airbrake, 2016)

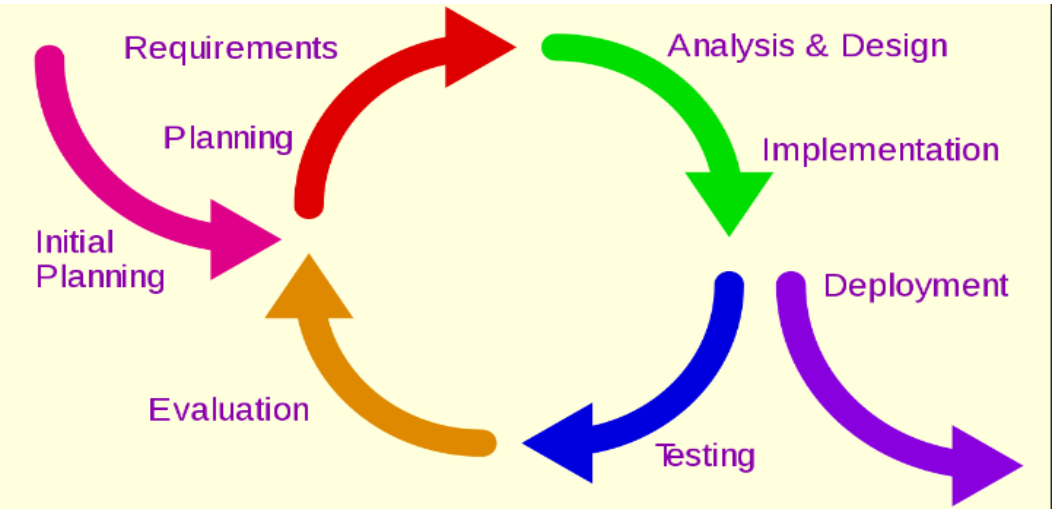

Figure 9 Iterative Methodology

4.6.1. Reasons for selecting Iterative methodology

I selected iterative methodology for developing project because of the following reasons:

1. It enables us to efficiently respond to changing demand and market conditions.
2. Consistent feedback loops guarantee that during development, client needs are accurately satisfied.
3. By breaking up large pieces of software into smaller, more manageable pieces, we may focus on particular features throughout each iteration.
4. Despite not being agile, this methodology can be helpful because I need to submit my project only once.
5. It emphasizes enough time for concept implementation and design improvement by eliminating superfluous documentation.
6. Its adaptability is ideal for projects that frequently need to be changed and improved.

5. Resource Requirements

For the development of our project, certain criteria should be met and external resources are required too. Resources required are divided into two parts:

Hardware Requirements:

a. Device with RAM of 8 GB or more.
b. Device with at least 20 GB free storage.
c. Device with processor of 8-12th generation (i3, i5, i7) or other alternatives with similar specs.
d. Internet Connection.

Software requirements:

a. VS Code

Visual Studio Code (VS Code), a renowned Microsoft product, has gained popularity as a widely used text editor among professionals and newcomers alike. Being open-source, cost-free, and interoperable with Windows, Linux, and macOS contribute to its appeal. Its efficiency is aided by the simplicity of installation and small weight, and it is a flexible tool thanks to the wide range of languages it supports, including C++, Java, and Python. Notably, the ability to integrate crucial functionality like debuggers and cloud/web development tools is made possible by VS Code's extension capabilities. Additionally, its intuitive layout, divided into discrete areas, improves the coding experience and makes it easier to write understandable code and perform efficient error debugging. (Tuama, n.d.)

In my project, I employed Visual Studio Code (VS Code) as my text editor, utilizing it for both frontend and backend development. For the frontend, I adopted Flutter, a framework running on the Dart language, while for the backend, I utilized PHP. To enhance the testing of my APIs, I incorporated the Thunder Client extension. This combination allowed me to efficiently manage and execute both frontend and backend tasks, leveraging the capabilities of VS Code and the specific tools tailored to each aspect of the development process.

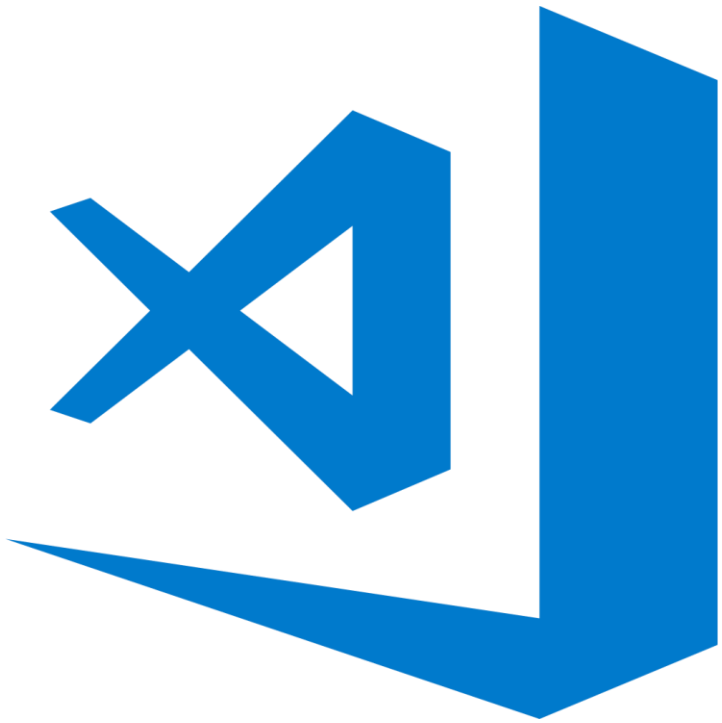

Figure 10 VS Code

b. XAMPP

A popular cross-platform web server that enables developers to locally write and test programs is called XAMPP, created by Apache Friends. It combines MariaDB, the Apache HTTP Server, and PHP and Perl interpreters. The open-source nature of XAMPP enables code change. With Apache, Perl, MySQL, and PHP, it provides a testing environment for websites prior to deployment, assisting with local testing and verification. Cross-Platform, Apache, MySQL, PHP, and Perl are all abbreviated as XAMPP to emphasize its all-inclusive online solution bundle. On desktops and laptops, it makes hosting and client testing easier, speeding up project review before launch. (javatpoint, 2021)

I used XAMPP in my project to create database which was done in MYSQL. I also used it to debug my backend APIs.

Figure 11 XAMPP

c. Figma

A flexible UI and UX design tool, Figma is used to create websites, apps, and UI elements that can be used into a variety of projects. Figma offers a range of design features, including innovative tools like the Arc tool and Vector Networks, to streamline design processes from conception to implementation. Its vector-oriented, cloud-based methodology enables work from any browser and is focused on organizational design systems as well as design, prototyping, and collaboration. Figma is used to create UI designs and prototypes for my project. (COUSINS, 2019)

Figure 12 FIgma

d. MSWord

Microsoft Word is a world-renowned word-processing tool that has had numerous updates with bettering features and technology since it was first released in 1983. With its built-in styles, Word makes formatting simple and is perfect for lengthy texts. It makes it easy to include pictures, movies, shapes, and charts and provides tools for setting margins, making columns, and controlling spacing. With features like margins, page breaks, columns, and table of contents, Word excels at complex jobs like writing books or creating brochures. Its facilities for footnotes, headers, cross-references, and bibliographies improve the construction of expert documents. I wrote every section of my project's documentation using MS Word. (Ballew, 2021)

Figure 13 MS Word

e. Android Studio

The official Integrated Development Environment (IDE) for creating Android apps is Android Studio, which builds on IntelliJ IDEA's powerful code editor and developer tools while also including additional capabilities to increase productivity. These include an adaptable Gradle-based build system, a quick testing emulator, a centralized workspace for all Android device development, real-time Live Edit capabilities for composable on emulators and actual devices, code templates and GitHub integration for streamlined app feature development, extensive testing tools and frameworks, lint tools for identifying various issues, and built-in integration with Google Cloud Platform

for seamless testing. For my project, I created an Android app using Android Studio. (Studio, n.d.)

Figure 14 Android Studio

6. Work Breakdown Structure

In a work breakdown structure (WBS), project components are visibly broken down and linked to the overall scope. It hierarchically arranges deliverables using workflow tools and project frameworks like timelines, Kanban boards, and calendars. A project baseline, stakeholder identification, a planned schedule, and deliverables with subtasks are among the essential elements of the WBS. WBS is used by project managers to clarify complex scopes, show dependencies, and provide teams a bigger picture than just task lists. It facilitates teamwork and increases comprehension of project complexity. The hierarchy of the structure streamlines project management and promotes clarity. Phases depending on work and timetable may also be included. (Raeburn, 2022)

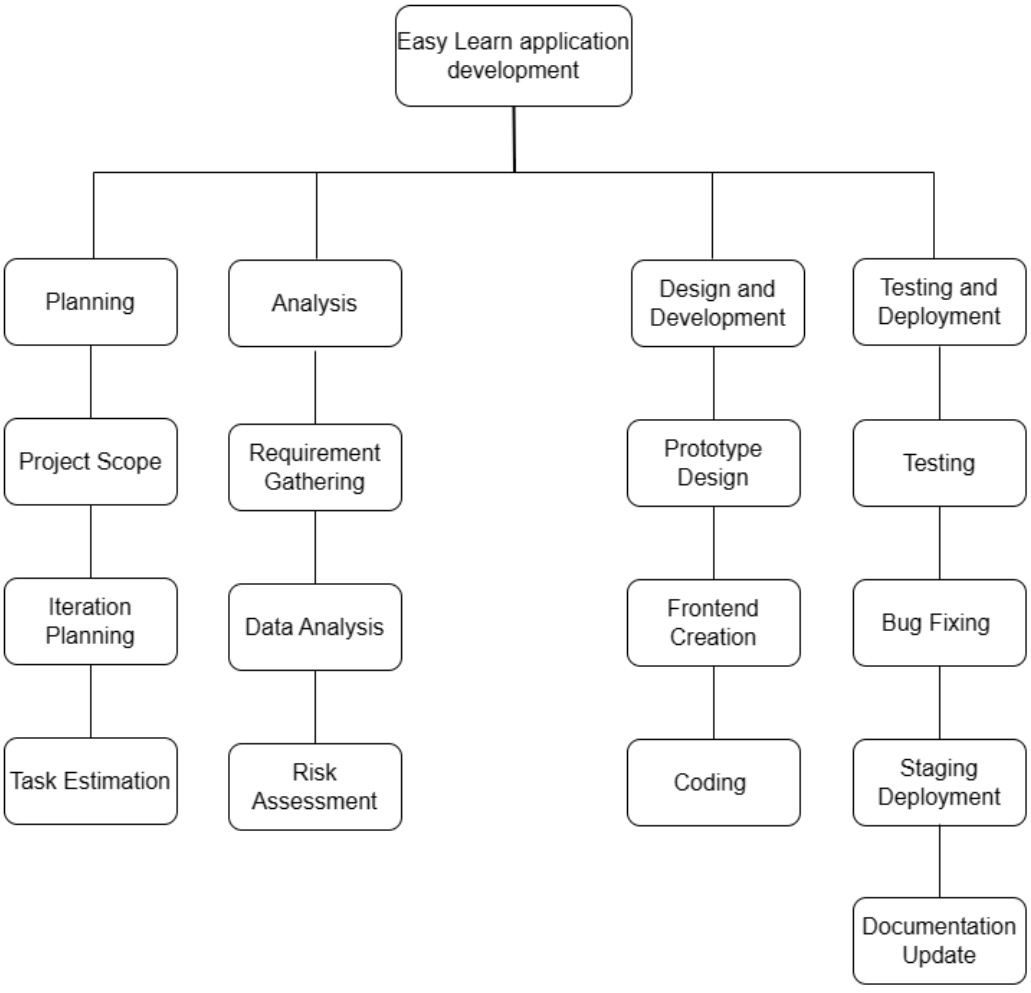

Figure 15 Work Breakdown Structure of Easy Learn

7. Milestone Listing

Milestone listing is a process of making a schedule for developing a project. Milestone for my project is given below:

- Project Start: 11th September
- Research: 26th September
- Requirement Gathering: 12th October
- Prototype design:30th October
- Development: 21st November
- Initial Testing: 23rd February
- Final testing:10th March
- Implementation: 7th March
- Documentation:20th March
- Complete project:8th April

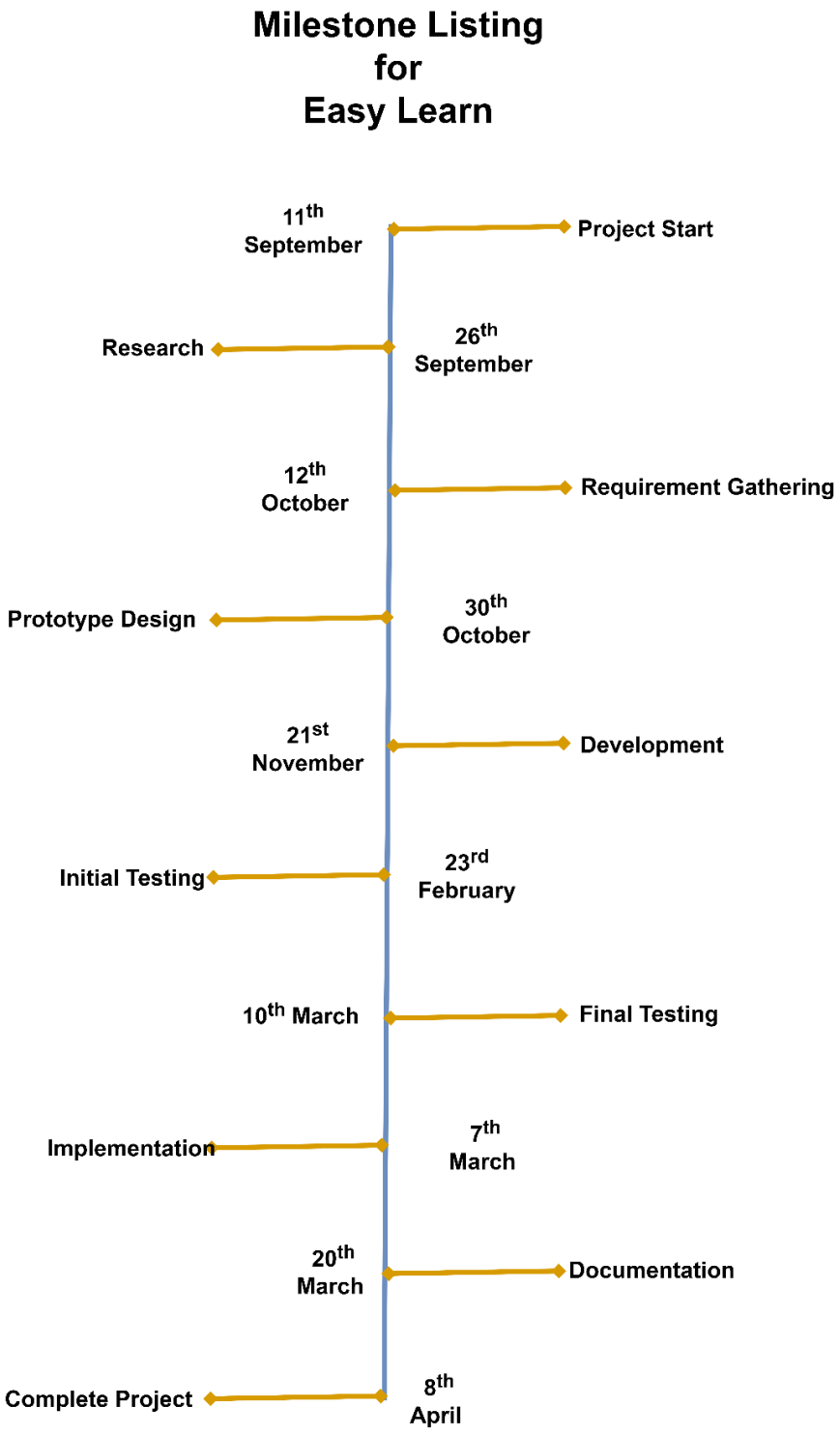

Figure 16 Milestone Listing

8. Project Gantt Chart

A Gantt chart, a common project management tool, plots activities or events against time visually. Tasks are shown on the y-axis of the graph, and progress is shown through horizontal bars on the time scale on the x-axis. The positions and lengths of these bars represent task progress. The chart includes all project tasks and their relationships, frequently using color to aid with comprehension. It makes the start and conclusion dates of the project clear, ensuring a thorough comprehension of the timeline. The status bar, which shows task progression, is an essential component. Gantt charts can also offer additional context, such as task priority, accountable parties, and contact information. Gantt charts effectively communicate project structure, timing, progress, and other supplemental data to support efficient project management. (Thomas, 2023)

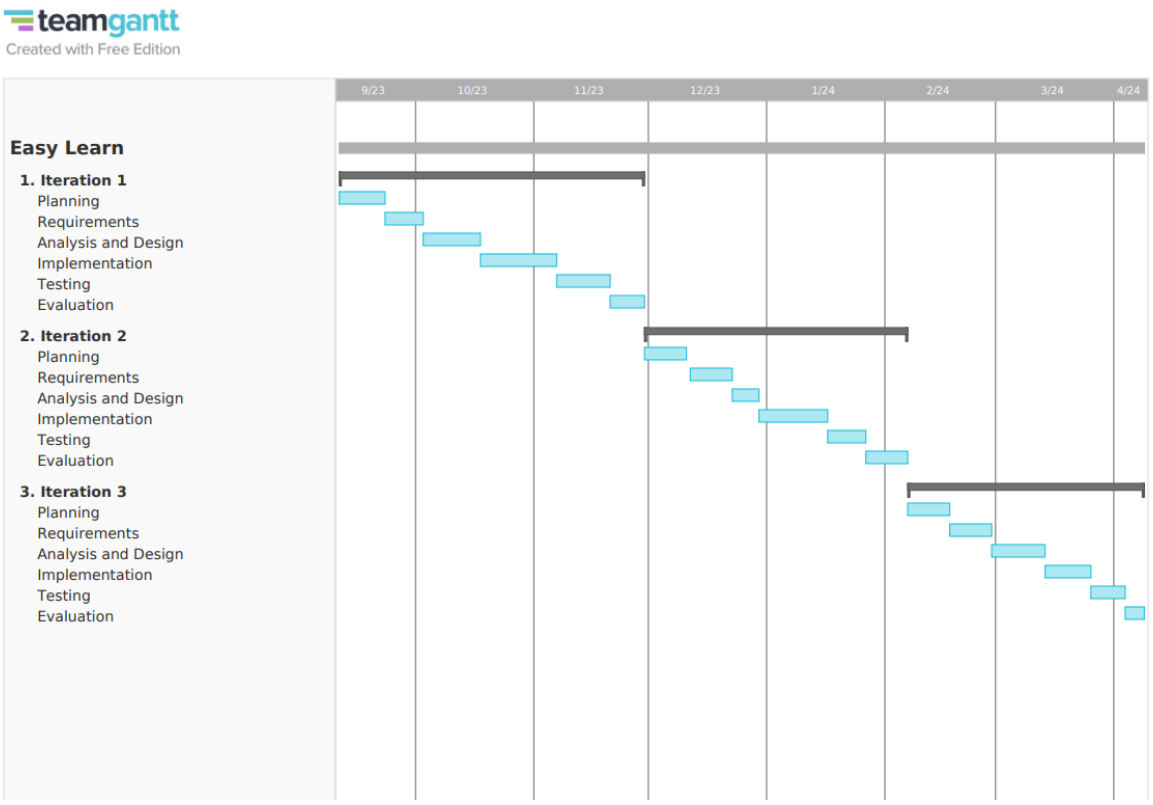

Figure 17 Gantt Chart

Bibliography

Airbrake. (2016, December 15). *Iterative Model: What Is It And When Should You Use It?* Retrieved from Airbrake: https://blog.airbrake.io/blog/sdlc/iterative-model

Awati, R. (2023). *Agile, DevOps and software development methodologies*. Retrieved from TechTarget.

Ballew, J. (2021, June 12). *What Is Microsoft Word?* Retrieved from Lifewire: https://www.lifewire.com/microsoft-word-4159373

Casteren, W. V. (2017). The Waterfall Model and Agile Methodologies :A comparison by project characteristics. 1 - 3.

coursera . (2022). *Our Vision*. Retrieved from coursera : https://about.coursera.org/

COUSINS, C. (2019, November 20). *What Is Figma? a 101 Intro* . Retrieved from Design Shack: https://designshack.net/articles/software/what-is-figma-intro/

Gaille, L. (2020, March 19). *15 Advantages and Disadvantages of a Waterfall Model*. Retrieved from vittana: https://vittana.org/15-advantages-and-disadvantages-of-a-waterfall-model

Half, R. (2014, February 11). *Waterfall Methodology 101: the Pros and Cons*. Retrieved from Robert Half: https://www.roberthalf.com/us/en/insights/career-development/waterfall-methodology-101-the-pros-and-cons

javatpoint. (2021). *XAMPP TUTORIAL*. Retrieved from javatpoint: https://www.javatpoint.com/xampp

Makadia, H. (2023, August 16). *Your Complete Guide To Rapid Application Development (RAD)*. Retrieved from marutitechlabs: https://marutitech.com/rapid-application-development/

Minnesota, U. o. (2022, February 11). *Agile Methodology: Advantages and Disadvantages*. Retrieved from University of Minnesota: https://ccaps.umn.edu/story/agile-methodology-advantages-and-disadvantages

MyManagementGuide. (2020). *PROJECT MANAGEMENT METHODOLOGY: DEFINITION, TYPES, EXAMPLES*. Retrieved from mymanagementguide.com: https://mymanagementguide.com/basics/project-methodology-definition/

mysecondteacher. (2020). *Your second teacher anytime, anywhere!* Retrieved from mysecondteacher: https://mysecondteacher.com.np/about

OTTISH. (2022). *OTTISHOTTISH*. Retrieved from OTTISH: https://www.ottish.com/

PAL, S. K. (2022). *Software Engineering | Spiral Model*. Retrieved from geeksforgeeks.: https://www.geeksforgeeks.org/software-engineering-spiral-model/

ProductPlan. (2022). *Rapid Application Development (RAD)*. Retrieved from ProductPlan: https://www.productplan.com/glossary/rapid-application-development/

professionalqa. (2019, April 30). *What is Iterative Model?* Retrieved from professionalqa: https://www.professionalqa.com/iterative-model

Raeburn, A. (2022, December 9). *The work breakdown structure (WBS) for project management: What it is and how to use it*. Retrieved from asana: https://asana.com/resources/work-breakdown-structure

Samar Al-Saqqa, S. S.-N. (2020). Agile Software Development: Methodologies and Trends. 248-250.

Studio, A. (n.d.). *Meet Android Studio*. Retrieved from developers: https://developer.android.com/studio/intro

Thomas, A. (2023, Jan 24). *What Is a Gantt Chart?* Retrieved from builtin: https://builtin.com/operations/gantt-chart

Tuama, D. Ó. (n.d.). *What is Visual Studio Code?* Retrieved from codeinstitute: https://codeinstitute.net/global/blog/what-is-vs-code/