Kingston University, BSc (Hons) (top-up)

Coursework Coversheet

Part 1 - To Remain with the Assignment after Marking

Student ID:	**Student Name:**
Module Code: EM6125	**Module Name: Software Development Practice**
Assignment number: 2	**ESoft Module Leader: Dr. Lochandaka Ranatunga**
Date set:	**Date due: 10th February 2022**

Guidelines for the Submission of Coursework

1. Print this coversheet and securely attach both pages to your assignment. You can help us ensure work is marked more quickly by submitting at the specified location for your module. You are advised to keep a copy of every assignment.

2. Coursework deadlines are strictly enforced by the University.

3. You should not leave the handing in of work until the last minute. Once an assignment has been submitted it cannot be submitted again.

Academic Misconduct: **Plagiarism** and/or **collusion** constitute **academic misconduct** under the University's Academic Regulations. Examples of academic misconduct in coursework: making available your work to other students; presenting work produced in collaboration with other students as your own (unless an explicit assessment requirement); submitting work, taken from sources that are not properly referenced, as your own. By printing and submitting this coversheet with your coursework you are confirming that the work is your own.

ESoft Office Use Only:

Date stamp: work received

Kingston University, BSc (Hons) (top-up)

Coursework Coversheet

Part 2 – Student Feedback

Student ID: K2170663	Student Name: Don. Chathuna Samodya Abeysekara
Module Code: EM6125	Module Name: Software Development Practice
Assignment number: 2	ESoft Module Leader: Dr. Lochandaka Ranatunga
Date set:	Date due: 10th February 2022

Strengths (areas with well-developed answers)

Weaknesses (areas with room for improvement)

Additional Comments

ESoft Module Lecturer:

ESoft Module Marker:

Provisional mark as %:

Date marked:

Software Development Practice Coursework 2 (Individual)

1. Provide an e-portfolio describing your individual contribution to the group artifact which you submitted under coursework1.
2. Prepare presentation slides to present your individual contribution.

	Criteria	**Marks**
1	Reflect on various computational and project management paradigms that are linked to software engineering practices. Prepare and estimated time plan for the project if 4 resource persons are allocated.	10
2	Compare and contrast different architectures and architectural patterns for software design. Justify the selection of the approaches you applied.	10
3	Consider software quality approaches including testing strategies, validation and verification approaches and evaluate their effectiveness in producing quality software. Describe about development and testing tools used and their advantages as good practices.	20
4	Reflection of individual contribution (criteria 2&3) with the lessons learned in the coursework. The use and experience gained through tools applied will be given pripority.	30
5	Successful justification of individual contribution (presentation)	20
6	Presentation materials (slides)	10
	Total	100

Submission method;

1. Report (pdf)
    - Word limit - 1000

Table of Contents

Table of Figures and Tables

1. Computational and project management

1.1 Software Development Life Cycle

The application of standard business procedures to the development of software applications is known as the Software Development Life Cycle. Planning, Requirements, Design, Build, Document, Test, Deploy, and Maintenance are the six to eight stages that are usually followed. The SDLC is a method for evaluating and improving the development process. It enables for a complete study of each process phase. As a result, businesses are able to maximize efficiency at each level. As processing capacity grows, the demand for software and developers grows as well. Companies must cut expenses, deploy software more quickly, and fulfil and exceed the expectations of their consumers. SDLC supports in the achievement of these objectives by detecting inefficiencies and rising costs and resolving them so that everything runs smoothly (www.tutorialspoint.com, n.d.)

During the software development process, various software development life cycle models have been established and designed. To assure success in the software development process, each process model follows a set of stages specific to its type.

The most significant and widely used SDLC models in the industry are listed below:

- Waterfall Model
- Iterative Model
- Spiral Model
- V-Model
- Big Bang Model

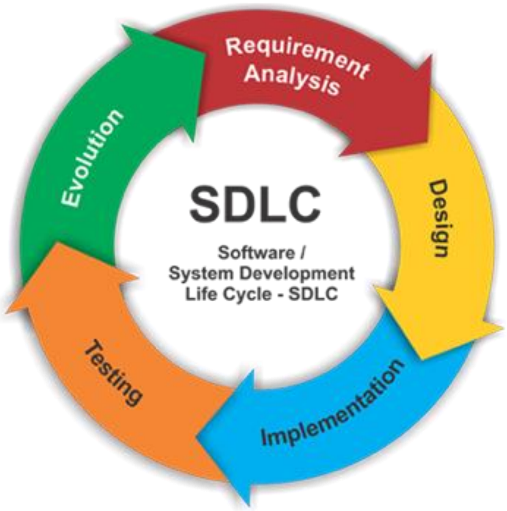


Figure 1 Software Development Life Cycle

A typical Software Development Life Cycle consists of the following stages −

Stage 1: Planning and Requirement Analysis

- It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry.

Stage 2: Defining Requirements

- This is done to clearly define and document the product requirements and get them approved from the customer or the market analysts

Stage 3: Designing the Product Architecture

- Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product.

Stage 4: Building or Developing the Product

- In this stage of SDLC the actual development starts and the product is built. And developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code

Stage 5: Testing the Product

- This stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS

Stage 6: Deployment in the Market and Maintenance

- Once the product is tested and ready to be deployed it is released formally in the appropriate market

1.2 Project Management Paradigms

What is a paradigm, exactly? It's a set of principles that reveals all of a subject's basic characteristics. A paradigm of Project Management tools, from this perspective, would be an excellent framework for selecting a Project Management tool. A paradigm, to put it another way, is something that allows a comprehensive 360° view of any subject or matter. Project management paradigms, such as waterfall and agile, are being researched.

1.2.1 Plan Driven Process

Plan-driven or plan-based development is an approach to software engineering where the development process is planned in detail. A project plan with the records of work to be done, the responsible individual, the schedule or the time plan, and the resources. Managers also use the plan to support project decision making and this can be considered as a tool for measuring progress. Plan-driven development is based on the “traditional” way of managing large software development projects.

The problem with plan-driven development is that early decisions have to be revised because of changes to the environments in which the software is developed and used but the inflexibility of this approach doesn’t give the liberty for any changes. However, planning decisions ahead would avoid unnecessary rework. Several companies are involved in a development project, a plan-driven approach is normally used to coordinate the work across each development site due to the preplanning availability and the possibility (The Paradigm of Project Management Tools - Apptio (2013).

Traditional Methodology

Software development processes are largely aimed at creating a flawless product that conforms to software quality standards and CRM development. Processes included a structured layout of a step-by-step approach from requirements assembly to final testing and release of the product, with a focus on design, development and testing.

Project management is a linear technique that occurs in a pre-planned series of stages. The budget and project time can be modified in this technique, but the criteria stay fixed.

Waterfall methodology

A sequential or linear approach to software development is used in the waterfall methodology. The project is divided into a series of tasks, with phases being the highest-level grouping. A proper waterfall strategy necessitates phases that are completed in order and project stakeholders' approval (Sherman, R. (2015). Typical list of waterfall tasks would include:

Scope and plan project

Gather and document requirements

Design application

Develop application and perform unit tests

Conduct system testing

Fix application as appropriate

Deploy application

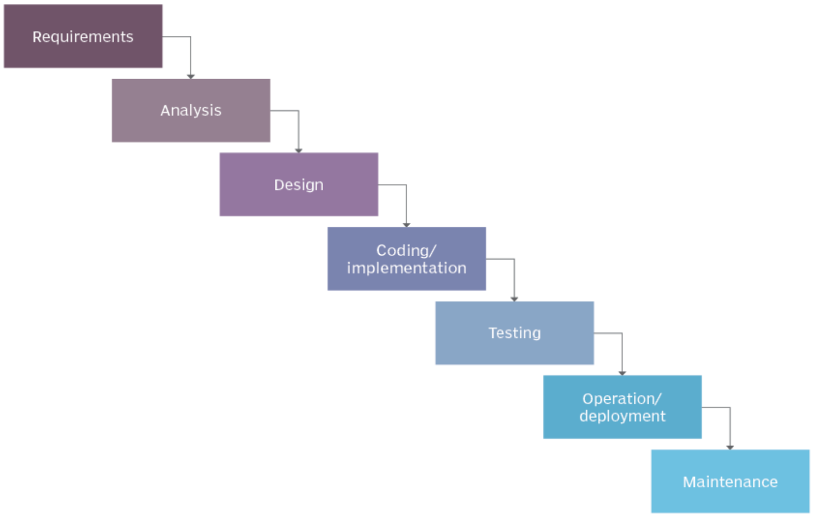


Figure 2- Waterfall Methodology

1.2.2 Agile Planning

Agile design is a project management style with an enhanced, repetitive approach. Instead of using an in-depth plan from the beginning of the project - usually product related - Agile allows for customization changes and relies on constant feedback from end users. Over time, cross-functional teams work toward product repetition and achieving OKR (goals and key outcomes), organizing their work into a recession focused on value addition. The ultimate goal of each iteration is to produce working projects.

The main difference between Scrum and Agile is that Agile is a project management style and Scrum is just one of many different approaches to follow that framework. Like Agile design, its goal is to create a functional product that gives value to the user. Rely on Scrum Sprint (more info) to work on product fixes, updates, new features, requirements and more. Like agile design, this is called production delay. (Anon., n.d.)

Agile Methodology

Every client and driver is given access to the system based on their role, so Agile has the appropriate algorithm. Everything is very carefully pre-mapped and, as a result, the manufacturer can easily manage their supply chain from start to finish without any human intervention.

- Conduct a comprehensive risk assessment.
- Create a quantifiable strategy and set explicit goals to track success.
- Remove any uncertainty about the project's schedule and overall scope.

Project management is an approach to working through the stages of a project - from start to finish - in a way that maximizes and encourages personal performance (my part) - when it happens and when everyone does what I have to do (and at the right time), the plans work and projects succeed.

Agile approaches, unlike SDLC's traditional methods, are precise and server-friendly. During the project development phase, clients / consumers have the chance to make changes. I have to use agile technique since that portion is generated and added independently. For my part, here are five reasons why I adopt agile technique.

- Reduces technical.
- Adapt to change easily and quickly.
- Using Agile for web application development and testing planning.

- Full alignment and transparency.
- Agile Software Development and Test Risk Reduction.

We can see that agile technique is better appropriate for my specific assignment based on both traditional and agile methodologies.

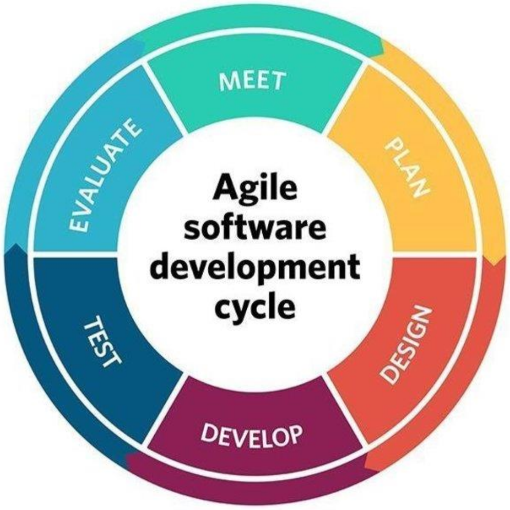


Figure 3- Agile Methodology

SCRUM Methodology

Scrum is a rapid development program that is used to develop software in the face of long and laborious cycles. Scrum's main purpose is to solve customer problems via clear communication, shared accountability, and continuous improvement. The process of improvement begins with the development of a list of desired features organized by necessity (item delay) and a general idea of what the item owner needs to build

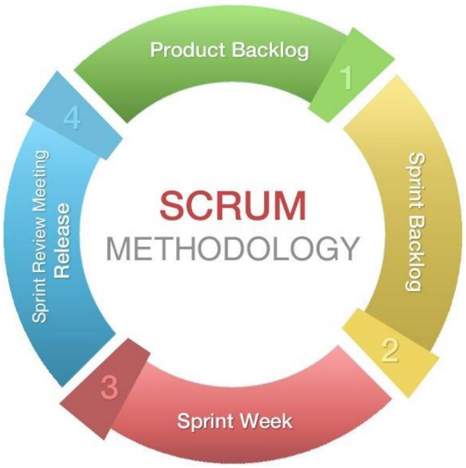


Figure 4- SCRUM Waterfall Methodology

Kanban Method

Kanban Method is a medium for designing, managing and improving flow systems for work. This system allows organizations to start with their existing workflow and make evolutionary changes. They can do this by visualizing their workflow, limiting existing work (WIP) and starting and stopping startup. The Kanban system derives its name from the Kanban-optical signaling mechanism used to control the ongoing work on intangible work products. (Anon., 2021)

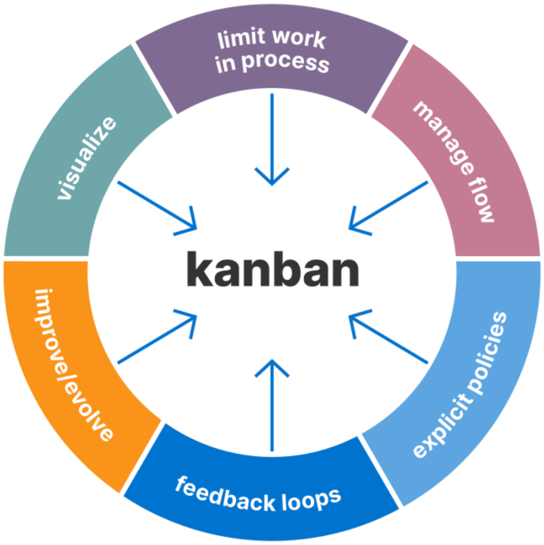
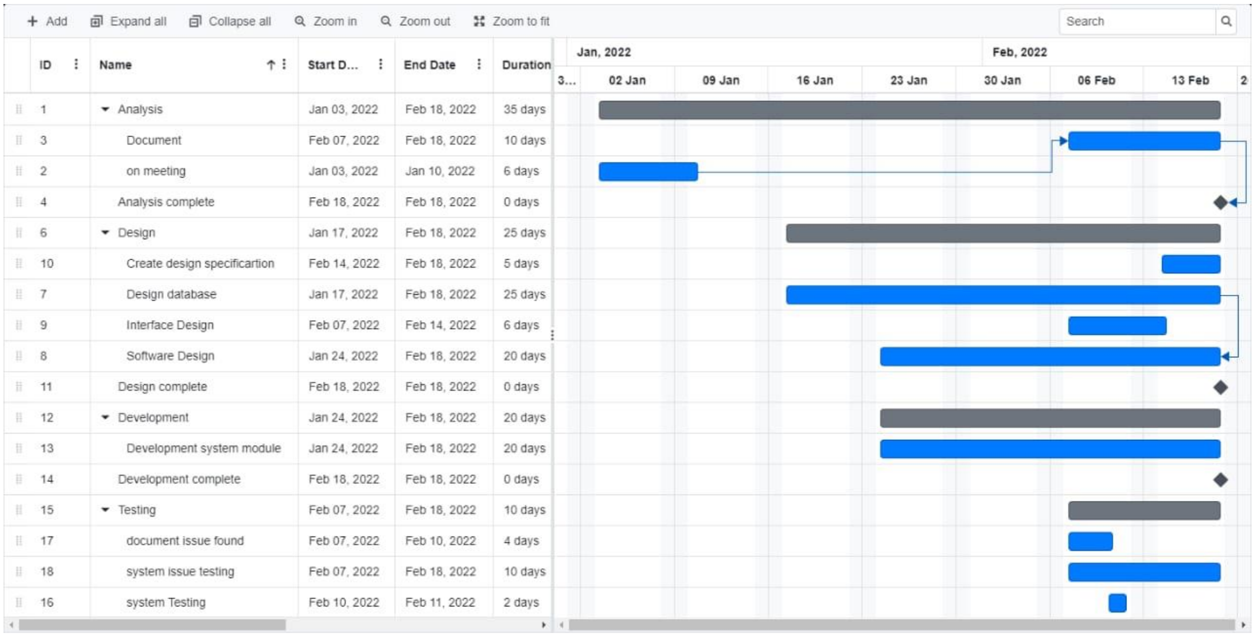


Figure 5- Kanban Waterfall Methodology

1.3 Time Plane

Gantt chart is a project management tool that illustrates a project plan. It usually consists of two parts: the left side outlines a task list, and the right side has a timeline with schedule schedules. The Gantt chart can include tasks start and end dates, milestones, task dependencies, and assignments. These roadmap and planning tools help teams maintain a cohesive project strategy regardless of the repetitive nature of the software development process.



ID	Name	Start D...	End Date	Duration
1	Analysis	Jan 03, 2022	Feb 18, 2022	35 days
3	Document	Feb 07, 2022	Feb 18, 2022	10 days
2	on meeting	Jan 03, 2022	Jan 10, 2022	6 days
4	Analysis complete	Feb 18, 2022	Feb 18, 2022	0 days
6	Design	Jan 17, 2022	Feb 18, 2022	25 days
10	Create design specificartion	Feb 14, 2022	Feb 18, 2022	5 days
7	Design database	Jan 17, 2022	Feb 18, 2022	25 days
9	Interface Design	Feb 07, 2022	Feb 14, 2022	6 days
8	Software Design	Jan 24, 2022	Feb 18, 2022	20 days
11	Design complete	Feb 18, 2022	Feb 18, 2022	0 days
12	Development	Jan 24, 2022	Feb 18, 2022	20 days
13	Development system module	Jan 24, 2022	Feb 18, 2022	20 days
14	Development complete	Feb 18, 2022	Feb 18, 2022	0 days
15	Testing	Feb 07, 2022	Feb 18, 2022	10 days
17	document issue found	Feb 07, 2022	Feb 10, 2022	4 days
18	system issue testing	Feb 07, 2022	Feb 18, 2022	10 days
16	system Testing	Feb 10, 2022	Feb 11, 2022	2 days

Figure 6- GANTT chart

There are two main reasons why people around the world of project management love Gantt charts. They make it easy to create complex designs, especially involving several teams and varying deadlines. Gantt helps chart teams plan work around timelines and allocate resources properly. Project designers also use Gantt charts to keep a bird's eye view of projects. They represent, among other things, the relationship between the start and end dates of tasks, milestones, and dependent tasks. Modern Gantt chart programs and software with advanced maps and advanced roadmaps synthesize information and represent how choices affect timelines. (What is a Gantt chart? | Atlassian (2022).

Agile methodology too chan have an impact on the time plan , the agile model of project design values flexibility and adaptability. Instead of creating a full timeline with deadlines, agile teams break down projects into smaller iterations (also known as sprints). At the start of Sprint, a team plans their work against the project goals over the next two weeks. Once that sprint is over, the achievements and developments that come with it will help you create a plan for the next sprint. Changing one function to a Gantt chart can show how it can affect the layout or product route map. For agile groups, this is essential as stakeholder feedback is a large part of the methodology.

2. What Are Architectural Patterns?

- In a particular context, an architectural pattern is a comprehensive, reusable solution to a common problem in software design. Architectural patterns are comparable to software development patterns; However, they cover a wide range of applications.

Following we have the types of architectural designs;

1. Layered pattern
2. Peer – to – peer pattern
3. Client sever pattern
4. Interpreter pattern
5. Master slave pattern

1. Layered pattern

- This pattern can be used to organize programs that can be subdivided into subgroups, each with its own abstract level. The next top layer is serviced by each layer below it.

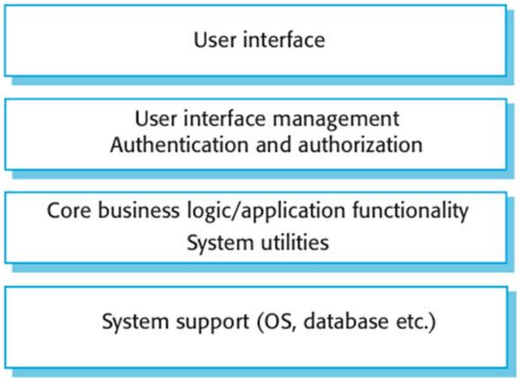

User interface
User interface management
Authentication and authorization
Core business logic/application functionality
System utilities
System support (OS, database etc.)

Figure 7 Layered pattern

Layers found in layered pattern architectures

i. Presentation layers
ii. Application layers
iii. Business logic layers
iv. Data access layers

2. Peer – to – peer pattern

- Individual components are referred to as peers in this structure. Peers can act as both a client and a server, seeking services from other peers and offering services to them. A peer can function as a client, a server, or both, and its role can change dynamically over time.

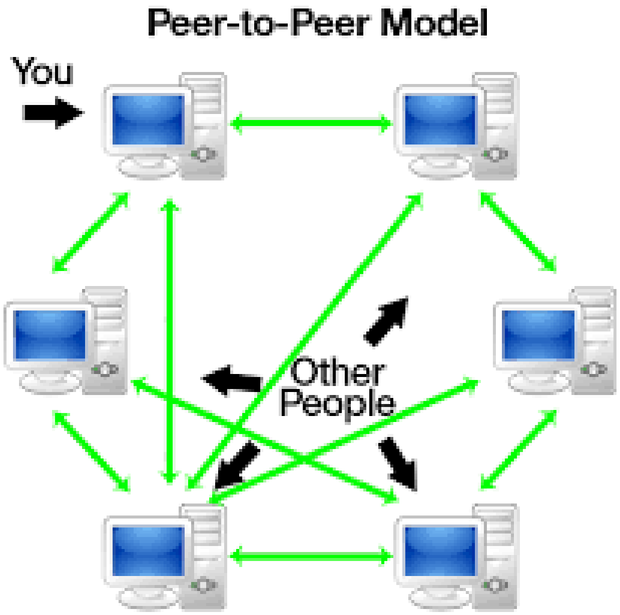


Figure 8 Peer – to – peer pattern

3. Client – severs pattern

- There are two parties in this pattern: a server and many clients. Multiple client components will be served by the server component. Clients request services from the server, and the server responds by providing those services. The server also continues to listen for client requests.

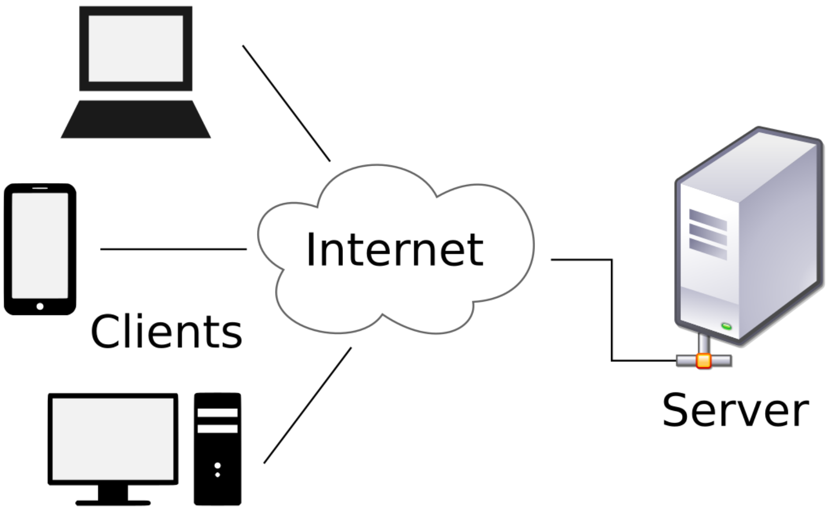


Figure 9 Client – severs pattern

4. Interpreter pattern

- This pattern is used to create a component that reads and interprets programs written in a certain language. It primarily describes how to analyze program lines, often known as phrases or expressions produced in a certain language. The main concept is to create a class for each language symbol.

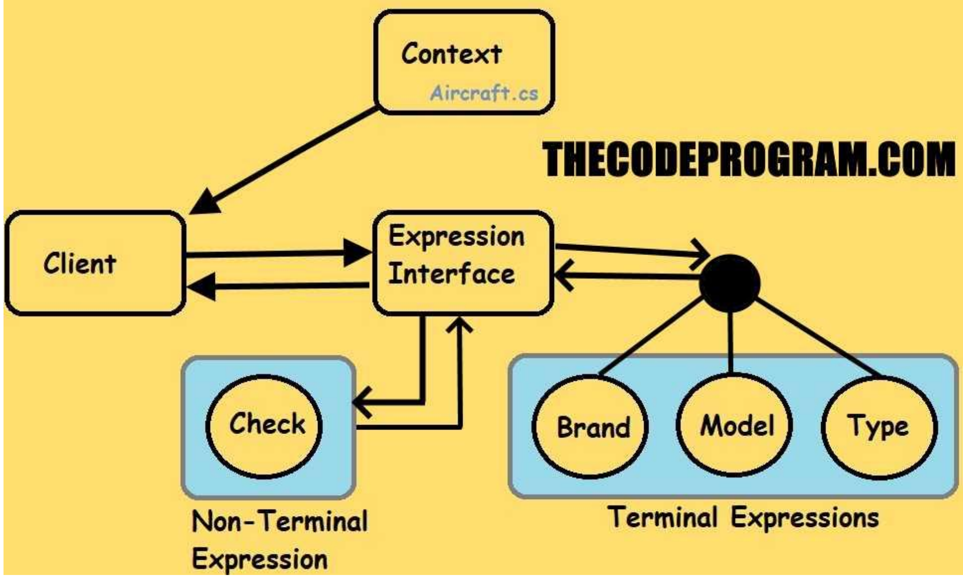


Figure 10 Interpreter pattern

5. Master slave pattern

- There are two parties in this pattern: the master and the slaves. The master component divides the work across identical slave components and computes a final result from the slaves' output.

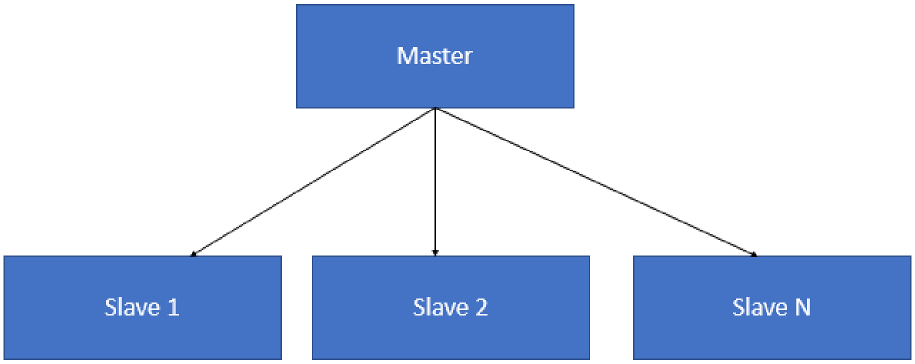


Figure 11 Master slave pattern

Comparison between client server architecture with other architectures

Architecture Name	Advantages	Disadvantages
Layered Pattern	i. Simplicity ii. Consistency iii. Browsability	i. The more layers have, the more risks may occur ii. No dependency inversion iii. Data may get messed up due to duplication of functionality
Peer – to – Peer Pattern	i. Supports spread out computing ii. Does not need specialist knowledge iii. Does not need an expensive	i. Cannot guaranteed about quality of service ii. Performance based number of nodes iii. Folders cannot be locally backed up
Interpreter Pattern	i. High dynamic behavior ii. Enhances flexibility iii. Ensure user programmability	i. Slower than complied programs ii. Less performance
Master slave Pattern	i. Accuracy ii. Efficiency iii. Exchangeability	i. Machine reliance ii. Feasibility
Client sever Pattern	i. Centralized ii. Scalability iii. Security	i. Maintenance ii. Staff should be specialized iii. Severs are expensive

Table 1 Comparison between client server architecture with other architectures

2.1 Justification of the architecture

- Since my part is mobile application, we have chosen to go with client – server architecture because when considering this is real-time system.

  This type of architecture requires simulation of the information server to minimize delays in remote operations and to balance the role of the data server in a distributed and multi-network environment. In this architecture the client does not need to connect to any remote server but only communicates synchronously with the local primary server. All updates are stored in the server cache. The primary server periodically picks it up from the client it currently serves and promotes asynchronous updates to the secondary server. This pickup strategy allows clients to return to document functionality as soon as new values are cached. Primary servers are used as intermediaries between the client and the secondary server. Because the system supports image additions / deletions and incomplete replication, a primary server should learn about mapping from file system to secondary services. Mapping is provided by calling a client-specific mapping server function. Upon delivery the primary server copies its contents to a new primary cache in the transferred cell to provide client-like information. (Sanjay Kumar, et al., 2014)

  Furthermore, this involves security mechanism because we are using google firebase as our real-time time database which has inbuilt verification and validation security. Moreover, if we develop this app in future when large number of users this system, we can monitor the app’s performance and get an idea on what areas we should improve. And also, since this is the free and trusted real time database, we can find in the market thus many people are using it so automatically this has larger community which helps to sort out any issue we face.

3. Software Quality Testing Approaches

Testing Strategies

- This is an approach that includes planning the software testing steps along with planning the time, skills and resources required to test the software. Software testing strategies are designed to test any software that starts with the smallest component of the software and integrate the test into the entire system.

  It is important to develop a testing strategy for the software that if the test isn't performed properly, it'll be a waste of your time and labor and a few bugs or errors might not even be detected.

  These can be some characteristics for good strategies.

  1. you must conduct a technical evaluation, which reveals many errors before starting the test. this may assist you to correct the technical error before starting the test and save time while testing the software.
  2. The test should start at the core of the software design and proceed outward to include the test into the whole software.
  3. You do not need to follow an equivalent strategy to check all the software. Appropriate testing strategies should be developed for various software engineering approaches.
  4. The software developer tests the software, and if the project is large, a separate team should be assigned to test the software.
  5. Debugging should be included in every test strategy.

3.1 Levels of Software Testing Strategies

- Each component of the software is tested separately as a unit to make sure that this component is functioning properly. The test then flows out of the spiral. The rotation of the spiral during a clockwise direction gradually expands the scope of the test, and in each case, the test integrates the components and ensures that each one components of the software are properly executed and therefore the desired performance is achieved.

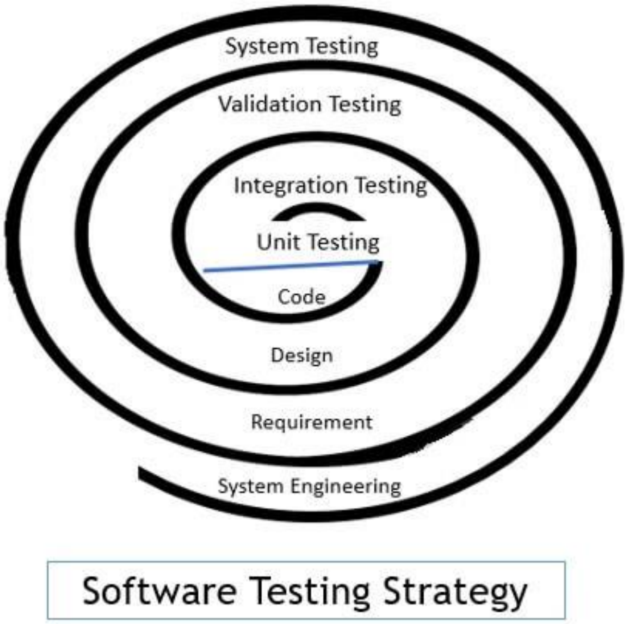


Figure 12 Software Testing Strategies

Unit testing

- Unit testing focuses on individual testing of the lowermost component of the software, also known as the unit. Unit testing involves testing each code segment to make sure it is working properly.

Integration Testing

- Each component of the software is tested separately as a unit to make sure that this component is functioning properly. The test then flows out of the spiral. The rotation of the spiral during a clockwise direction gradually expands the scope of the test, and in each case, the test integrates the components and ensures that each one components of the software are properly executed and therefore the desired performance is achieved.

Validation Testing

- Validation testing focuses on testing software against customer-specified requirements.

System Testing

- System Testing Focuses on testing the entire system and its other system elements and it checks the performance of the system.

3.2 Verification and Validation

- Software verification and validation involves variety of software quality assurance activities. By verifying the software, we make sure that every function of the software is correctly executed and executed. By validating the software, we guarantee that the developed software meets the wants set by the customer.

Verification testing

- Software verification and validation involves variety of software quality assurance activities. By verifying the software, we make sure that every function of the software is correctly executed and executed. By validating the software, we guarantee that the developed software meets the wants set by the customer.

Validation testing

- Ensuring that a product meets its intended use and therefore the needs of its users. After successful verification, development teams must perform validation tests with the essential product and within the authentic (or emulated) use environment. (T, 2020)

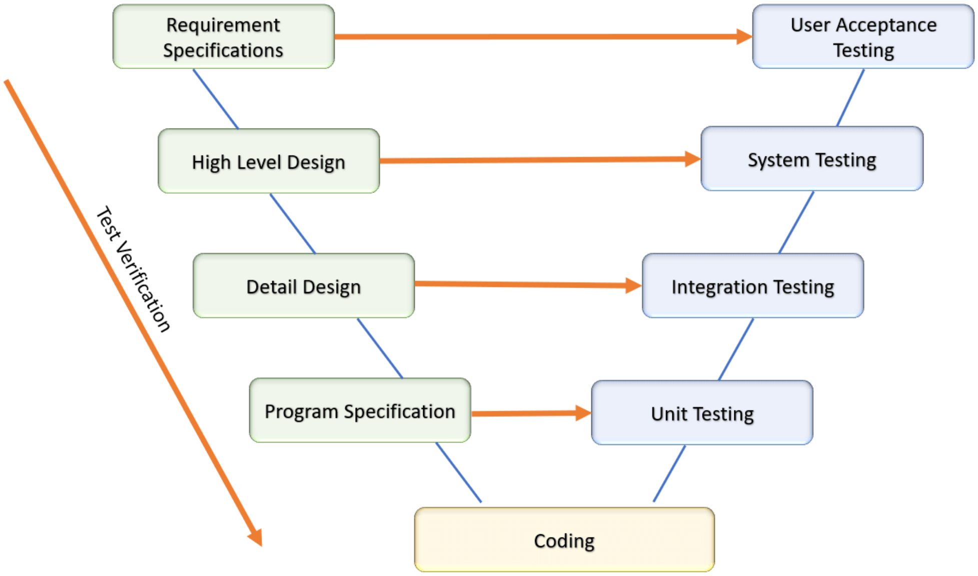


Figure 13 Verification and Validation

Comparison between Verification and Validation

Basis of Differentiation	Verification	Validation
Type of process	Verification is in the form of static steps useful for verifying designs, documents, codes, programs, etc.	Validation, as a process, is dynamic in nature. It serves as a mechanism for validating as well as testing actual products.
Execution of codes	Verification does not involve any form of code execution.	Validation always involves the execution of codes.
Human/ computer based	Verification is completely human based and involves the checking of documents and files.	Validation is computer based and involves the execution of programs.
Methods used	Verification involves methods including reviews, inspections, walkthroughs, desk-checking, and so forth.	Validation involves methods including black box (functional) tests, white box (structural) tests, gray box tests, and so forth.
Area of application	Verification checks whether a specific software conforms to the given specifications or not.	Validation checks whether a specific software adheres to customer requirements and expectations or not.
Level of exercise	It is equipped to catch errors that are not capable of being caught through validation. It serves to be a low-level exercise.	Validation catches errors that are not capable of being caught through the verification process. It serves to be a high-level exercise.

3.3 Testing Tools and Development Tools Used in The System

When developing the Pick & Go Delivery Management System, we divided our group tasks among the members of my team. Three people contributed to the website development and my other partner Parinda and I contributed to the mobile application development. There are two mobile applications in our project. So, I developed a mobile application for the Customer App. Furthermore, various third-party libraries were used there. Therefore, it has a very high tendency to cause errors. But it is my responsibility to ensure that the final product I have to offer is a High-Quality Product. So, I used an Automation Tool to test this developed mobile application. There are various Automation Testing Tools for Mobile Application Development. I used the Appium Testing Tool, one of the best performing testing tools.

Appium Testing Tool

Appium is a free and open-source framework for testing mobile user interfaces. Appium lets you test native, hybrid, and web applications, and support automated testing of both physical devices and emulators or simulators. It provides cross-platform application testing. That is, a single API works with Android and iOS platform testing opportunities.

Figure 14 Appium Testing Tool

It does not depend on the OS of the mobile device. Appium has a framework or package that converts Selenium Webdriver commands into UIAutomation (iOS) or UIAutomator (Android) commands, depending on the device type and not the OS type. (Hamilton, 2022)

Android Studio Development

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development

Figure 15 Android Studio Development Tool

Android Studio provides a unified environment where you can build apps for Android phones, tablets, Android Wear, Android TV, and Android Auto. Structured code modules allow you to divide your project into units of functionality that you can independently build, test, and debug. (Anon., n.d.)

Features of Android Studio

i. Instant App Run

- It is an advanced technology that understands the conversions that take place in applications and provides it instantly without taking the time to rebuild and install the apk. So, as soon as you can see the changes in the app immediately.

ii. Visual Layout Editor

- The layout editor helps to quickly build the layout by adding various attributes such as hard-code or dragging. The preview of the code is easily visible on the visual editor screen and can be modified accordingly by dynamically resizing it.

iii. Fast Emulator

- It's exactly the same as android phones to test how the app looks on physical devices. It provides a real-time experience for Android apps. It lets you speed up your apps and test them on a variety of configuration devices, such as tablets and Android phones. It helps make your application development life cycle shorter and more efficient.

iv. Intelligence Code Editor

- Android Studio provides you with an intelligent and quick code editor. This will help you and guide you with the correct code. It allows you to complete the code beforehand and analyze your code in advance before building. Android Studio has the special feature of developing code from the drop-down list suggesting code that you can integrate.

v. Help to Build Up App for All Devices

- Android Studio builds apps for all screen sizes, drawings and gear devices. Different types of features in hardware such as GPS location tracking and multi-touch can also be stimulated.

vi. Help to Connect with Firebase

- Android Studio helps to provide a real-time experience with IoT-based project development with dynamic enhancements to the app.
  The Firebase connection helps to create direct updates and provide a database connection. We need to use the Firebase Connection to build high quality applications that help build scalable infrastructure for application building. You can create chat apps using the Firebase connection, which will help you to have a pleasant chat experience.

vii. Color Previews

- The Android Studio helps us to see the XML code in the preview to see how perfectly we design the app as required before launching the app. It provides powerful functionality and enhanced features of dragging or resizing the application. It has drag and drop features but does not support all functions, so be careful when doing so.

Android Studio may be a platform that helps build advanced and fully developed applications with the newest features. Android is consistently updated with the newest versions of the newest technologies. Learning the Professional Android App Development Course Series will enhance your career during this vast developing region. Android is one among the fastest growing and developing languages today.

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

Figure 16 GitHub Collaborative Tool

There are numerous benefits that developers have, using GitHub as a version controller or repository. quite that other advantage of using GitHub as follows;

It makes it easy to contribute to open-source projects

- Documentation
- Showcase project works
- Markdown
- GitHub as Repository
- Track changes and Version Control across the code
- Integration Options

GitHub is divided into two parts: Git and Hub. The service includes access restrictions, as well as tools for basic task management and various collaborative capabilities, such as all the projects you manage. Our source code projects are hosted on GitHub's programming languages, and it monitors many changes made to each iteration.

As a result, "Git" features an editing control system, a tool that enables developers to monitor continuous changes in their code. "Hub" refers to a group of people who share similar interests. It's all about community collaborative work to test, improve and absorb new ideas from the published code. It is worth fully investigating (Novoseltseva, 2020)

Firebase

Google Firebase is a Google-backed application development software that enables developers to develop iOS, Android and Web apps. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.

The Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync data between your users in real-time

Figure 17 Firebase Real - time Database

Features and benefits of Firebase

- **Authentication**

The firebase Auth product offers a range of authentication methods, including email and password. Additionally, there are other ways to verify third-party providers, like Google. you'll create your own interface, or use an open-source customizable interface. SDK provides easy-to-use and ready-made UI libraries for authenticating Firebase applications. This allows our system to verify user information and helps secure our system from flooding with fake user info.

- **Real time database**

This feature ensures real-time storage and synchronization of data between users with the assistance of NoSQL Database. once you want the application data to remain offline, firebase is the right choice. It updates data sync in milliseconds. Firebase are often scaled up for mobile application development and accessed from server devices, as a result we can get real-time order updates when parcels are in delivering process, when order places and when pickup request is accepted or decline it will instantly update on the firestore database

- **Cloud Messaging**

Firebase cloud messaging (FCM) allows you to send push messages to point out something that interests users of the system. This helps to focus on users with information that interests them rather than blasting everyone with messages that aren't relevant to you.

4. Individual Contribution

As my part of the development, my task was build the customer side of the mobile application. Which has order placement, tracking, check order status functionalities. So, along with my partners we have concluded to use MVP (Model View Presenter) architecture pattern which leads us to focus on core functionalities of the system. As a result, it allows to test the concept of the scenario given without wasting more time. This architecture prevent us from brainstorming unnecessary functionalities. As a result we have able to clearly identify the user interfaces and back end systems to go with. Moreover, MVP shows that there are three directories: Model, Views and Presenter. For example, when a request is added, the display directory returns the request data to be added to the presenter. After that, the presenter then submits the data to the database and the model directory maps it to the database.

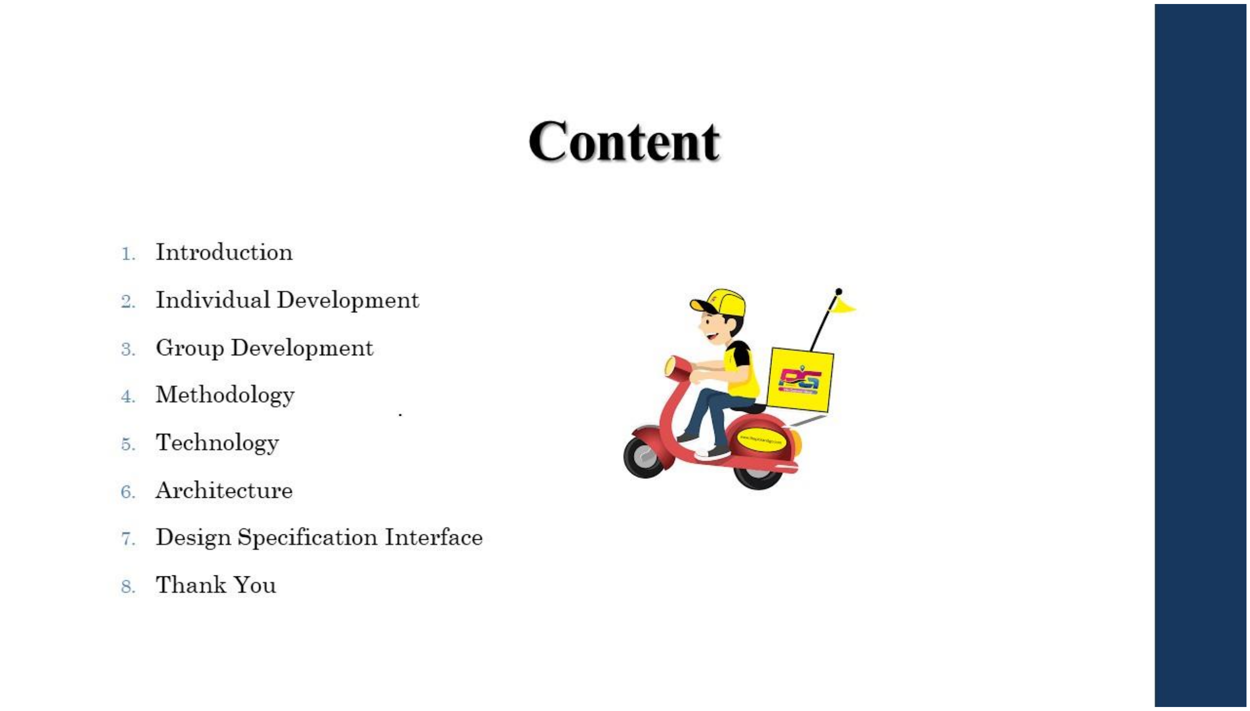
Furthermore, I used android studio as my development IDE. Because It's integrated with development environment for Android Operating System. Furthermore, It allows structured code modules to allow to divide our project into segments of functionalities which helps to debug test & build independently. Since, Android Studio supports Hot Reload function which helps to see the code changes instantly when we making changes. Moreover, with intelligent code editor helps to edit & structure the codes and identify errors in indentations. As a result Its saves us more time on development. Since, google and android are under the same development environment, Android studio helps to connect to the firebase directly. As a result I was able to sync the data fetches from Flutter UI to the same Firebase database which is connected with a pick & go web based system. As a result our team got benefited with the Realtime crud operations done by Mobile Application which are instantly updated on the same data base which is connected with the web based system.

Finally, I have contributed my maximum extent to my team to make this project a success. Meanwhile, I was able to get a knowledge on how web based system works with Architectures and how does a team collaboratively work to make the project success at a given deadlines.

5. Presentation Slides

Software Development Practice Coursework - 2

Name: Chathuna Samodya
Batch 05 – Kingston University



Content

1. Introduction
2. Individual Development
3. Group Development
4. Methodology
5. Technology
6. Architecture
7. Design Specification Interface
8. Thank You

Introduction

Abstract The technology plays a vital role in the today's world, specially it has a major impact on the business. Today's organizations use technology to enhance its competitive edge and the effectiveness. While Pick&GO is a Sri Lankan package delivery service provider who delivers goods island wide. However, their business procedures are facing issues due to the paper-based system which has taken control over the organization, therefore they have decided to automate their entire business procedure with a development of a web application and a mobile application. And this document will emphasis on how the developers contributed toward this development.

Individual development

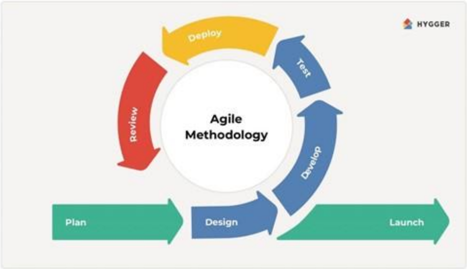
My Individual part is Customer Mobile app. It is,

- Customer login
- Customers sign up
- Customers request
- Delivery
- Order tracking

Group development

Sanju Nimesha	Prinda Sathsara	Methmi Prabavi	Chathuna Samodaya
Web application	**Mobile application**	**Web application**	**Mobile application**
• Add New User • Register User • Remover User • Update User • Landing Page	• Rider login • Riders accept request • Customer delivery • request list	• Login page • Sidebar • TopNav • Dashboard • Parcel sender Results • Parcel Sender Receiver • Master Page	• Customer login • Customers sign up • Customers request • Delivery • Order tracking

Methodology



- Every client and driver is given access to the system based on their role, so Agile has the appropriate algorithm. Everything is very carefully pre-mapped and, as a result, the manufacturer can easily manage their supply chain from start to finish without any human intervention.

- Conduct a comprehensive risk assessment.
- Create a quantifiable strategy and set explicit goals to track success.
- Remove any uncertainty about the project's schedule and overall scope.

- Project management is an approach to working through the stages of a project - from start to finish - in a way that maximizes and encourages personal performance (my part) - when it happens and when everyone does what I have to do (and at the right time), the plans work and projects succeed.

- Agile approaches, unlike SDLC's traditional methods, are precise and server-friendly. During the project development phase, clients / consumers have the chance to make changes. I have to use agile technique since that portion is generated and added independently. For my part, here are five reasons why I adopt agile technique.

Technology

Flutter for the Front-end

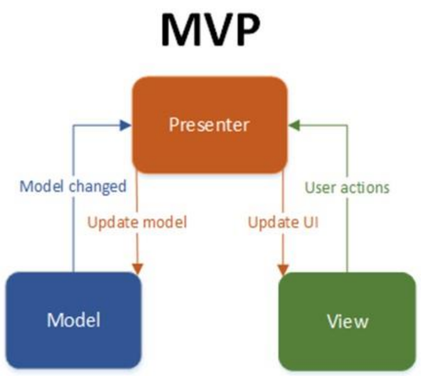
Flutter is defined as the Google's UI toolkit for building beautiful, natively compiled applications for mobile (Android, iOS) desktop (Linux, Mac, Windows, Google Fuchsia) and the web from a single codebase.

Firebase for the back-end

Firebase is basically a Google-backed app development platform providing features such a Realtime database which support quick updates , cloud storage, authentication, crash reporting, machine learning, remote setup, and static file hosting.

Flutter

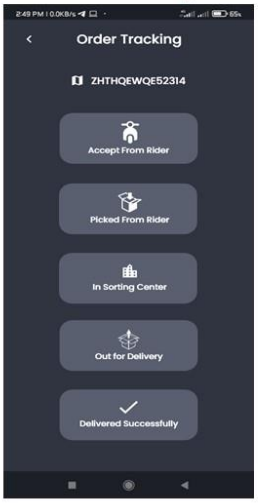
Firebase



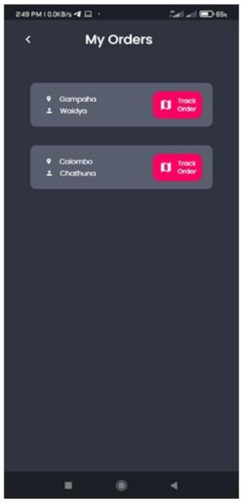
Architecture

- It is composed of the following three components: Model: Layer for storing data. It is responsible for handling the domain logic(real-world business rules) and communication with the database and network layers. View: UI(User Interface) layer.

Design specification Interfaces

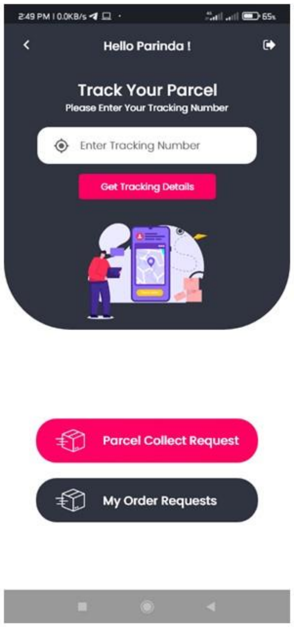


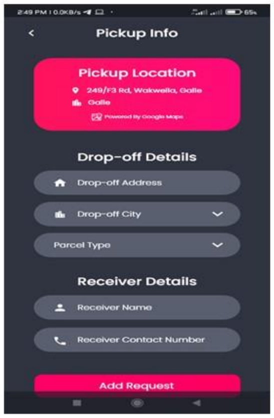
ZHTHQEWQE52314



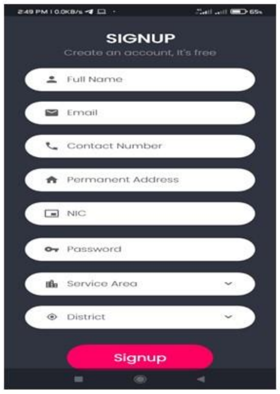
Gampaha
Waidya

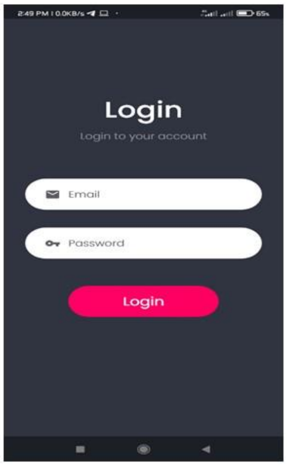
Colombo
Chathuna





249/F3 Rd, Wakwella, Galle
Galle





References

Anon., 2021. *agilealliance.* [Online]
Available at:
https://www.agilealliance.org/glossary/kanban/#q=~(infinite~false~filters~(postType~(~'page~'p

ost~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'kanban))~searchTerm~'~sort~false~sortDirection~'asc~page~

Anon., n.d. [Online]
Available at:
https://developer.android.com/studio/features#:~:text=Android%20Studio%20provides%20a%20unified,build%2C%20test%2C%20and%20debug.
[Accessed 28 02 2022].

Anon., n.d. *monday.*. [Online]
Available at: https://monday.com/blog/project-management/agile-planning/

Hamilton, T., 2022. *https://www.guru99.com.* [Online]
Available at: https://www.guru99.com/introduction-to-appium.html
[Accessed 28 02 2022].

Novoseltseva, E., 2020. *https://apiumhub.com.* [Online]
Available at: https://apiumhub.com/tech-blog-barcelona/using-github/
[Accessed 28 02 2022].

S. K., S. N. & S. R., 2014. *https://www.researchgate.net.* [Online]
Available at:
https://www.researchgate.net/publication/305763900_AN_EXTENDED_CLIENT_SERVER_ARCHITECTURE_IN_MOBILE_ENVIRONMENT
[Accessed 27 02 2022].

T, N., 2020. *https://binaryterms.com.* [Online]
Available at: https://binaryterms.com/software-testing-strategies.html
[Accessed 28 02 2022].

www.tutorialspoint.com, n.d. *https://www.tutorialspoint.com.* [Online]
Available at: https://www.tutorialspoint.com/sdlc/sdlc_overview.htm
[Accessed 27 2 2022].