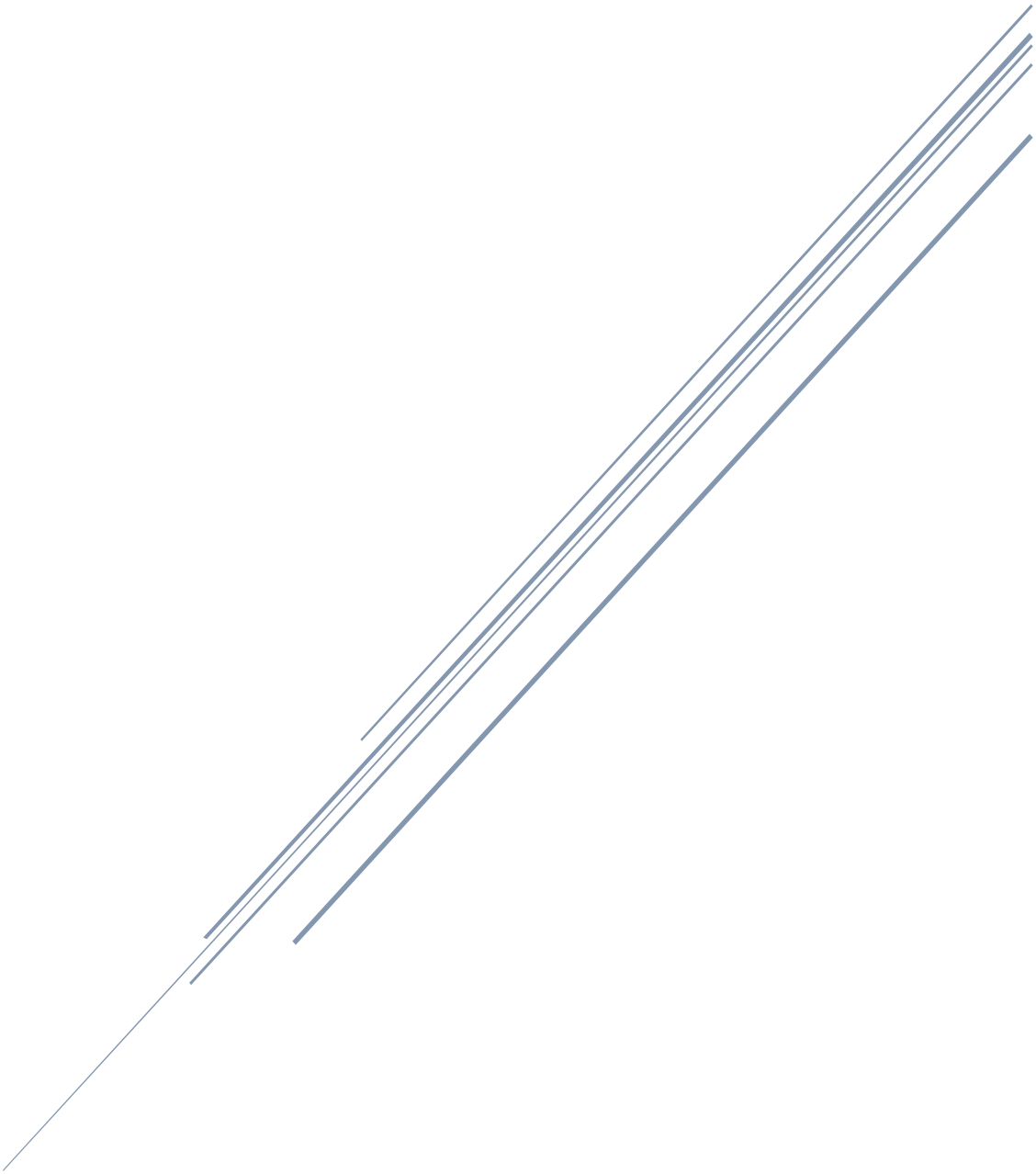
DISP PORTFOLIO

Sudhanshu Malla 17034820



UWE

Development of Information Systems Projects

Table of Contents

1. Task 1:

1.1. Description for the enterprise:

Marks and Spencer (M & S) are a high street retailer specialising in food. It sells fresh food imported from various parts of the world under the common brand M & S. Being a retailer business it uses a range of information system to conduct its day-to-day operations. Firstly, it receives the fresh products (cold chain) from M & S warehouse and staff process them using **Deliver Processing System (DPS)**. Then the staff manually display the products in the store. Customers purchase this product, and the store uses a **Transaction Processing System (TPS)** to track the progress of daily revenue through sales of food. The system supports barcode scanning of product, payment system with cash or cards and prints the receipts through a printer. Likewise, the system is capable of handling various loyalty cards schemes, store gift cards, voiding the transaction and training staff in the till.

Besides, the store uses a **Decision Support System (DSS)**, which aids the manager decision on future orders on food delivery based on current purchase models of the store. This operates by providing a statistical projection of future stock and the current sale model. For the current stock level and operation of the store, the system uses **Central Store Stock Management (CSSM)** system.

Likewise, it uses **Management Information System (MIS)** to create a report of a staff member about their transaction behaviour. This provides the manager with the performance data of a particular staff member. The manager could use this to generate a report and send it to higher management when needed.

Finally, the business is using **Customer Relationship Management Systems**, which requests customers to give feedback on their visit to the store. Besides this system allows the store to communicate with its customer through email, phones.

1.2. Zachman Framework:

Zachman Framework	**Why (Motivation)**	**When (Time)**	**Who (People)**	**What (Content)**	**How (Function)**	**Where (Network)**
	Vision (Guidelines)					
Scope (Contextual)	1. Convenient food hub in a busy hospital like Royal Berkshire Hospital	2. Identify the deadline/ completion date	3. Identify the customer base, (who will the store be serving? like mostly hospital staff and patients) Stakeholders, Roles in the organisation (Organisation chart)	4. List of products, customer information, transaction details, warehouse info	5. List of products, transaction detail.	6. Identification and description of organisation & warehouse location
	Design (Standards)					
Enterprise and Environment (Conceptual)	7. Identify the business goal, steps and store daily/ monthly targets	8. Identify the required steps to complete the process on time.	9. Organisation chart, service that allows a customer to pay in terminal	10. identification of the nature of store products (e.g., quantity and food department)	11. Conceptual activity model of the retail working process (used to describe the processing, receiving and selling stages)	12. Structure and interrelationship of store facilities with a web-based desktop application (Store launcher) that is used to run the store services.
Store Information System (Logical)/ system engineer	13. Identify the functional requirement of the business	14. Action plan covering the process and timeline of steps required E.g., charts, timeline	15. Creating store information system human-system interface architecture	16. Characteristics of entities like warehouse order details (food department, quantity, variant, the delivery time of lorry)	17. Application architecture/ software model. used to identify and describe a software system and process	18. Connectivity and distributed system architecture (Logical section of running services like delivery management system, gap scanning etc.)

	Implementation (Standards)					
Store Information Technology (Physical design)	19. Identify the technical requirement of IS of the business	20. Developing an information system and conforming requirements are met.	21. Creating store information system human-system interface description	22. Physical data model, database languages, web programming language statements	23. Requires system architecture design. describe techniques of information system and data exchange between various system	24. Store information system network detailed architecture like a server
Store Information Components (Modules and subsystem)	25. Responsible for making system works when the store is open and every part of services are working properly.	26. Setting hard deadline fairly to test and release the system before making it live	27. System security & its architecture	28. releasing the data into the database using various database languages	29. Develop a system, Code statements, control blocks, DBMS stored procedures, etc	30. Physical data network components, communication protocols like releasing the system in the store and train staff to uses system in computer and handheld terminal (HHT)
	Operation (Standards)					
Functioning Store Information System	31. Provide quality fresh food to the customers by bringing it from the warehouse on time.	32. Deploy the function system by pre-set date.	33. Store staff running daily shifts serving customer and managing store, deliveries, payment using provided desktop application and terminals.	34. Functioning database, knowledgebase, contacts	35. Releasing the software system in the store	36. Day-to-day operation in-store using the delivery system, CSSM food system, etc.

1.3. Description of framework cells:

Cell	Row	Column	This cell is appropriate for standards, models and descriptions which:	Examples of standards that may fit this cell:
1.	Scope	Why	Address customer service and food stock across store boundaries.	A standard method to quantify customer service in the store
2.		When	Identify and describe the fundamental store events	A standard characterization of the essential store and food delivery events useable by store staff and warehouse workers
3.		Who	Identify the essential components of the store	A standard method for identifying the key organizational components of the store information system and the standard description developed by this method.
4.		What	Identify and describe the important store services and warehouse (delivery) information.	A designation of the principal information components in the food store.
5.		How	Identify, describe, and regulate important store product information and delivery process.	A standardized designation of the fundamental processes shared by the food store
6.		Where	Identify and describe the global entities involved in delivering service by the store	A standard identification and description of individual and organizational participants in the store operation
7.	Enterprise & Environment	Why	Identify the business goal and objectives like daily/ monthly sales target.	A standard method for quantifying the value of individual satisfaction and its contribution to organizations.
8.		When	Determine the order and timing for the processes of fundamental store information services.	A standardized process modelling methodology or a conceptual process model which could be a standard for a group of similarly functioning food store
9.		Who	Identify and define the roles of individuals Related to the store.	A standardized workflow modelling method or specification could be a standard for similarly operating food delivery organizations.
10.		What	Define and describe the essential types of information required for the operation of a food store	A standard method for semantic description, narrative or conceptual data model useable for food delivery for the store.
11.		How	Identify and describe the fundamental retail working, management, and support activities in a food store.	A standardized activity modelling methodology or a conceptual activity model standardized for organizations that operate in an essentially identical manner.

12.		Where	Specify and describe the layout of store facilities and their interconnection with the information system.	A standard functional schema for the organization and linkage of facilities within an organization.
13.	Store Information System	Why	Relate to the functional requirements andthe test and acceptance criteria for a store information system.	Standards for information system project life cycle management, testing, and documentation, along with standard functional requirements shared by similarly functioning organizations.
14.		When	Detail the methods used to describe or descriptions of processes and event sequences within the enterprise.	Standard methods for specifying events and timing at the logical level and a specification for event sequence which could be standardized for similarly functioning food store.
15.		Who	Detail the methods used to describe, or the description of the functioning architecture for the interaction of individuals with the store information system.	Standard methods for specifying the architecture of the human-computer interface, and descriptions of such interfaces used by similarly functioning food store.
16.		What	Detail the methods used to prepare a logical data model or the non-technological description of the data used for store operation in an enterprise.	Standard methods for preparing logical data models, and logical data models useable by similarly functioning food store
17.		How	Describe the structure of software to support the store and food delivery processes.	Standard methods, techniques and software components for the food store
18.		Where	Describes the communication architecture supporting the store systems	Standard methods and techniques for representing information system linkages within the enterprise
19.	Technology Model	Why	converts store functional requirements into system operational requirements.	Standard high-level technical specifications of system operational requirements.
20.		When	Technical design of store information system control and timing structures.	Standard technical specifications for store information system control mechanisms.
21.		Who	Technical descriptions of the interaction of individuals with the store information system.	Standard practices for the human-system interaction in-store information systems.
22.		What	Detail the methods used to prepare a physical data model or the technological description of the data used to store information system in an enterprise.	Standard methods for preparing physical data models, and physical data models useable by similarly functioning food store.
23.		How	Specifies the technical design of a store information system, including structure, language, database and communication components.	Standard specifications for program languages and communication protocols.
24.		Where	Details the technical network architecture of a store information system	Standard practices for representing network architecture along with standard architectures for networks supporting similarly functioning food store

25.	Modules and subsystem	Why	Description of technical requirements for store care information system function.	Standard rules and specifications of the end conditions and means to obtain results.
26.		When	Timing descriptions of the components of store information systems.	Standard timing and machine cycle descriptions and definitions.
27.		Who	Identification of individuals and their access to specific components of the store information system.	Standard definitions and descriptions of individual roles, data access and system operation permissions.
28.		What	Physical data definitions, fields and addresses for store information and operation.	Standard metadata for technology-specific implementations of store information.
29.		How	Descriptions or scripts for component level applications in-store information systems.	Standard "programs" and similar structures such as a relational database stored procedures for information systems that support similarly functioning food store.
30.		Where	Description of the physical network components as nodes and linkages.	Standards for specification of node addresses and the protocols for communicating among nodes.
31.	Functioning Enterprise	Why	Definition of the business and operational strategy of the store.	Standards for store information system outcomes and quality assurance of food delivery processes.
32.		When	Schedule of releasing tasks and operations.	Standard timelines and cycles for store and delivery tasks such as approving delivery, verifying stock using the information system by the deadline.
33.		Who	The store staff/ team running the daily shifts	A standard list of providers and practitioners performing their responsibilities.
34.		What	Description of store information.	Standard specification of store information
35.		How	Release and actual store and food delivery activities.	Practice guidelines, accepted practices, and activities and outcomes required by a regulatory authority.
36.		Where	The store and provider network.	Standard procedure in information system like delivery processing and CSSM food system.

2. Task 2: Basic Business Process of the enterprise

This business process model uses a single pool with several human-based tasks. This BPM aims to demonstrate the business process of the enterprise in a very basic way with the use of BPMN gateways, events and tasks.

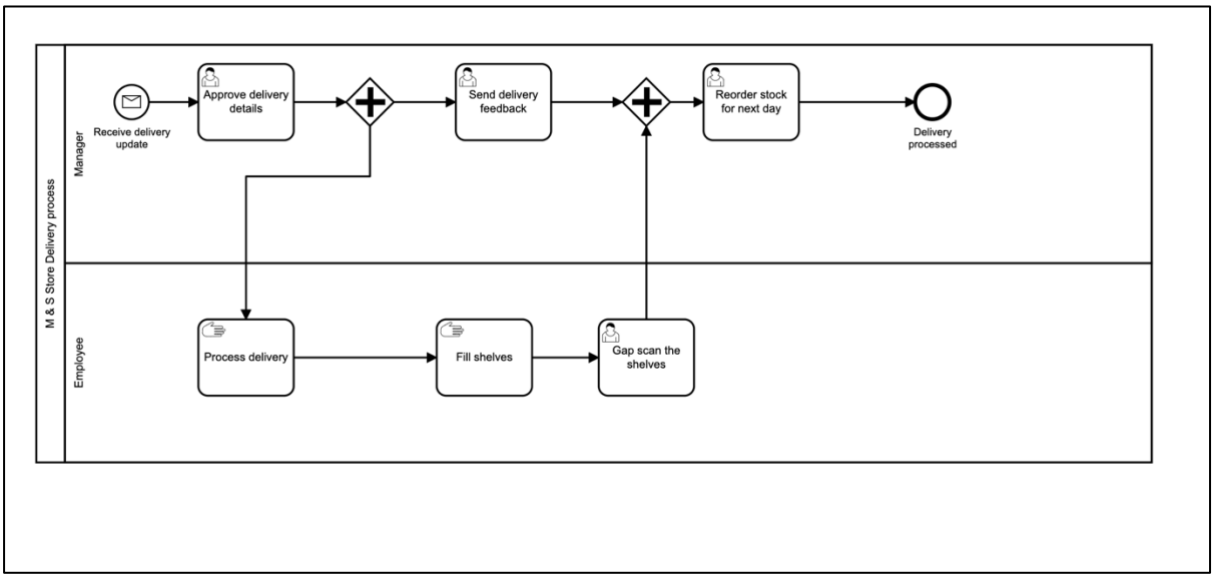


Figure 2.1: Basic Business Process Model (finalised)

The above business process model (figure 2.1) was finalised as task 2. However, before that, another basic business process model was created which was later scoped down.

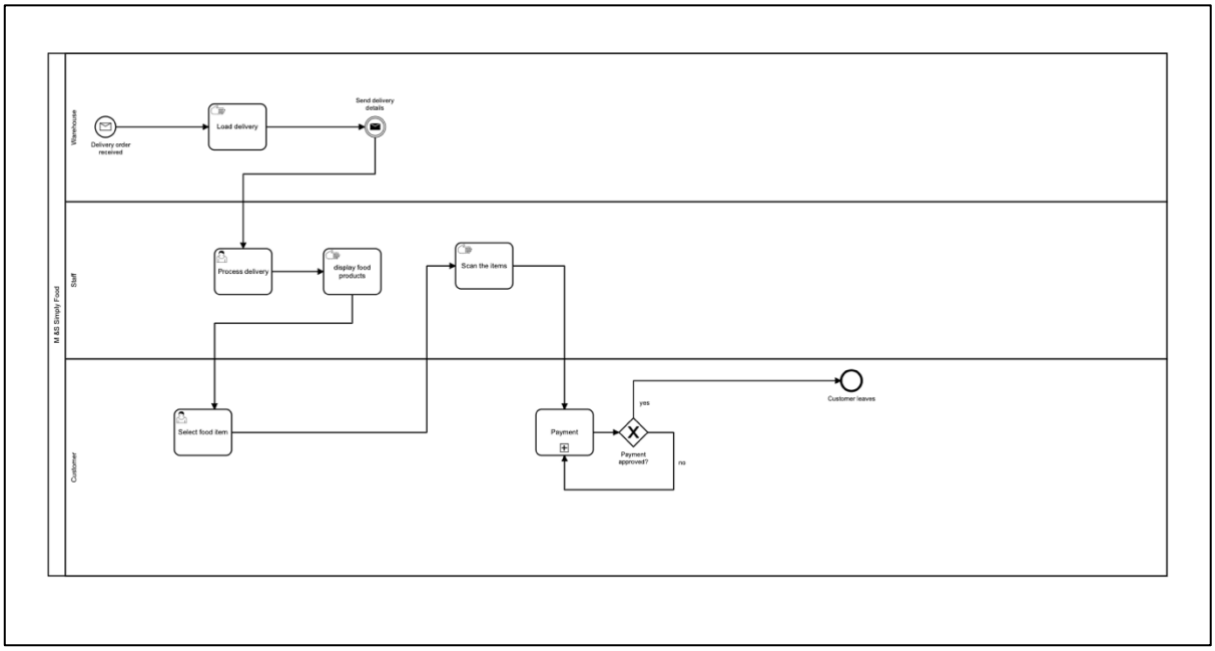


Figure 2.2: Initial Basic Business Process Model

3. Task 3: Elaborate business process

This business process model is based on the basic one however it explains the business process in more detail. It makes use of two pools involving both human and service-based tasks.

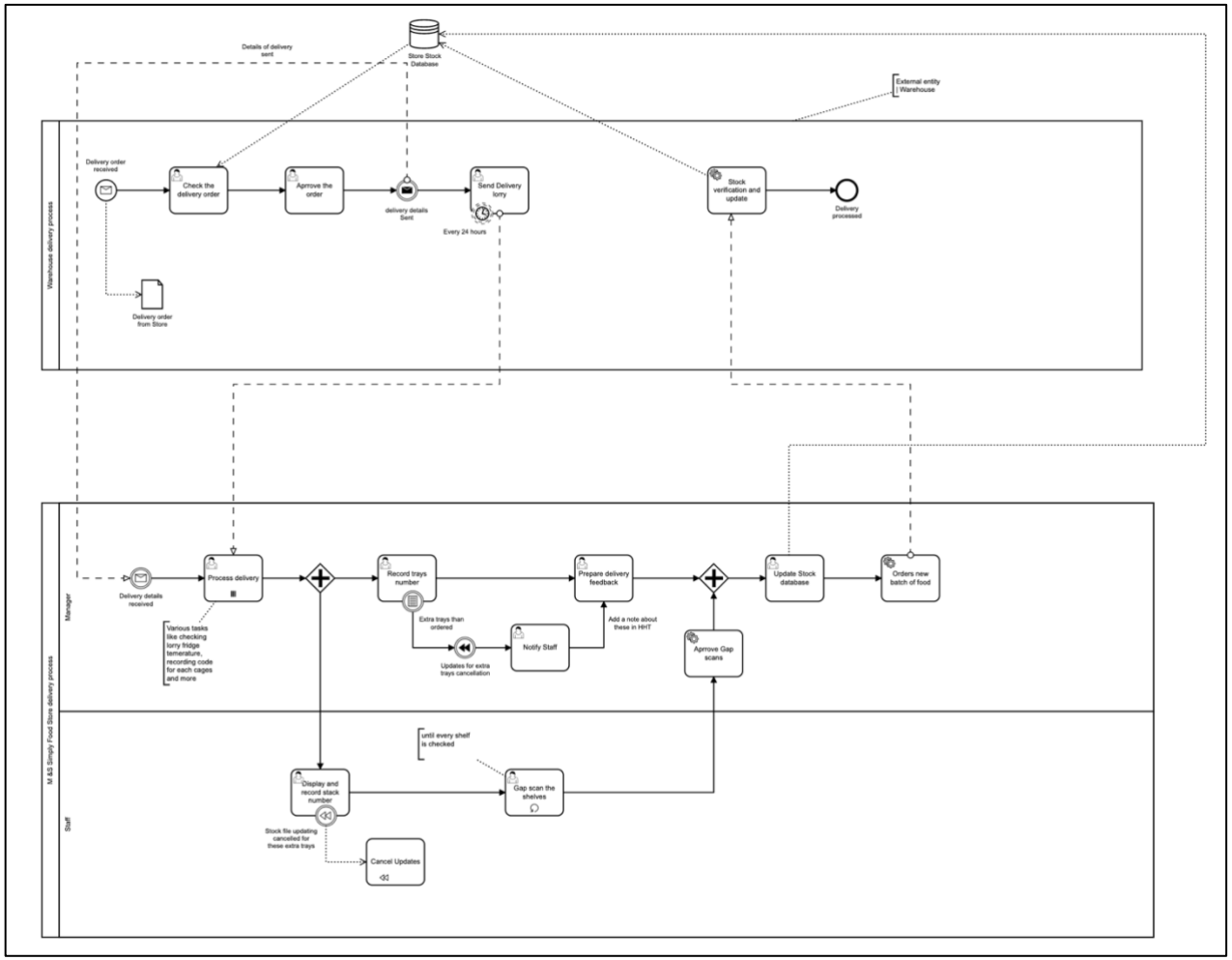


Figure 3.1: Elaborate Business Process Model (Final)

In the elaborate BPMN (Figure 3.1), the process starts with receiving an order request in a warehouse. The warehouse staff checks the delivery request of the store and approves it. The details of it can be seen from the centralised database. Then after confirmation, a message is sent to the store about the delivery details containing the items and quantity the store can expect. Then, the warehouse staff dispatches a lorry. This is a non-interrupting timer boundary event set every 24 hours. So, each day the store received a fresh batch of foods.

In the store delivery, the process starts with the delivery details of the day sent by the warehouse. Upon receiving the lorry, the manager checks the lock code for the lorry, the temperature of the fridge in a lorry, items received. After that, the tasks are divided between the manager and staff. While the manager starts counting the trays, the staff will record the stack number and displays the item. Based on the quantity of the item received and how closely it matched with the store order, the manager prepares delivery feedback. Similarly, at the same time, the staff carries on with gap scanning of the shelves. As the name suggests, the staff simply records the gaps in the store shelves. The gap scanning result is made available to the manager which the manager should approve. Ideally, the gap scanning result and delivery feedback result should match closely.
However, in some cases, the warehouse might send extra products. So, a compensation event is in place to cancel such quantity to be updated in the system. Likewise, the manager informs the staff to avoid updating the stack number as well. These trays number should be reported back to the warehouse in delivery feedback.

Then concluding from both the tasks, the manager updates the stock file which is linked to the same centralised database. Upon doing so, the system creates a new order which is received by the warehouse staff.

Here, the system validates and updates the order in the warehouse system. And the process ends. Then this final order will then be received.

As with the basic version, the above elaborate business process model (figure 3.1) was also scoped down from the following elaborated business process model. This was done to focus on one section of the enterprise.

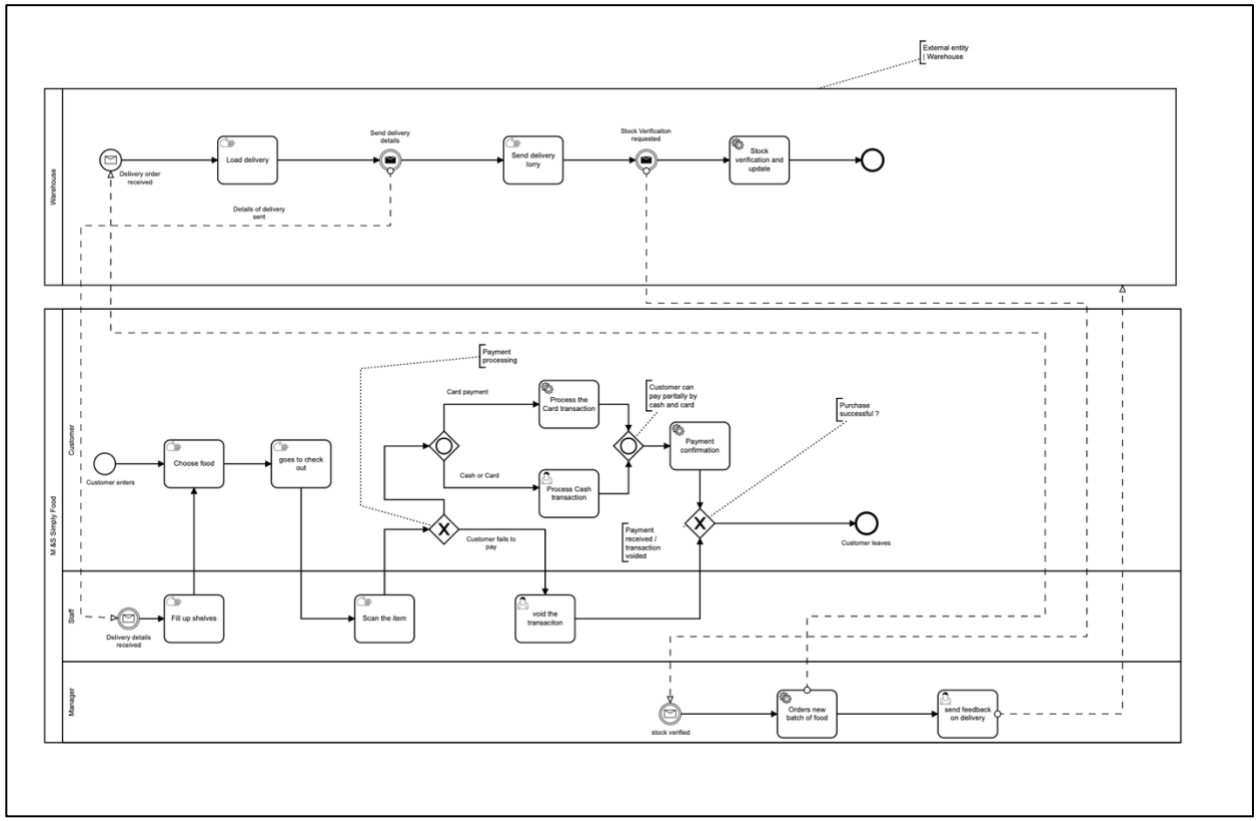


Figure 3.2: Initial Elaborated Business Process Model

4. Task 4: Deployed business process

The elaborate business process model was again further scoped down to deploy the model. It was a model to include just the basic human and service task of the enterprise. In this model, the event is triggered in the store with the store manager as an actor. A system is in place to check the delivery. After this, the token passes through an exclusive OR gateway. Based on the result, the delivery is processed, or the missing trays report is printed by the system. Both of the human tasks were there merely to check that the random Boolean is generated correctly, and the token can pass through both ways.

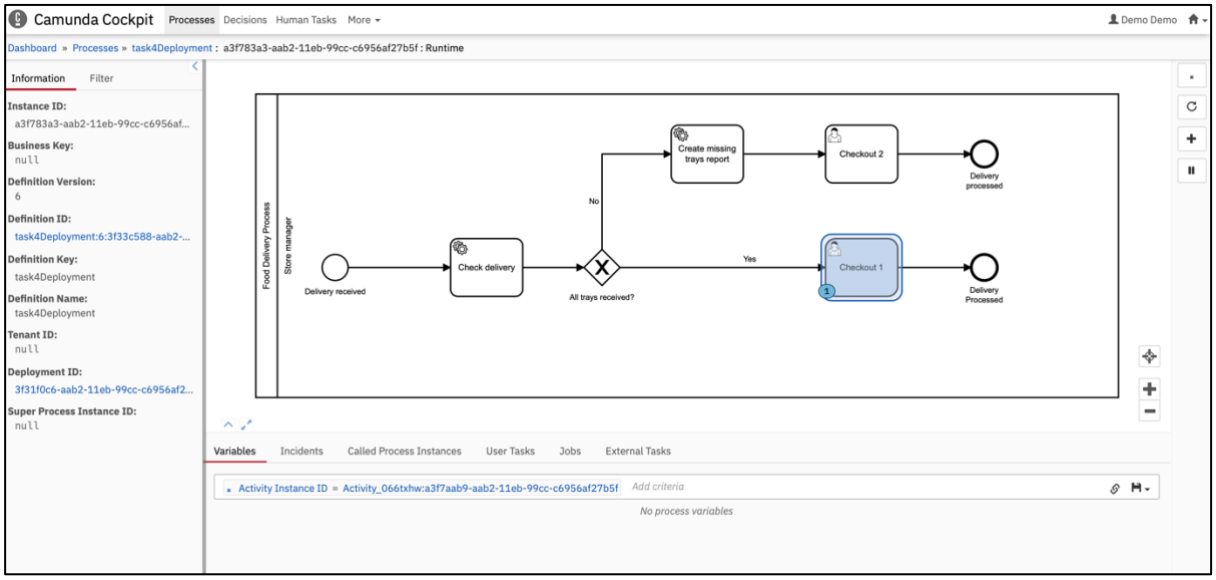


Figure 4.1: Deployed Business Process Model in Camunda Workflow engine (final).

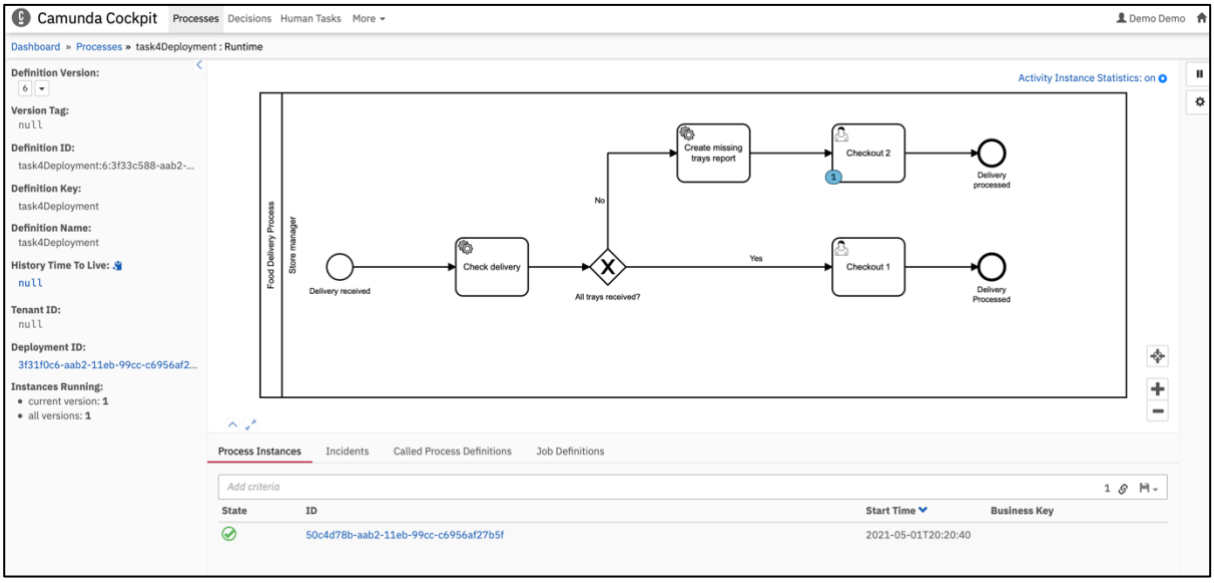


Figure 4.2: Deployed Business Process Model with a token in checkout 2

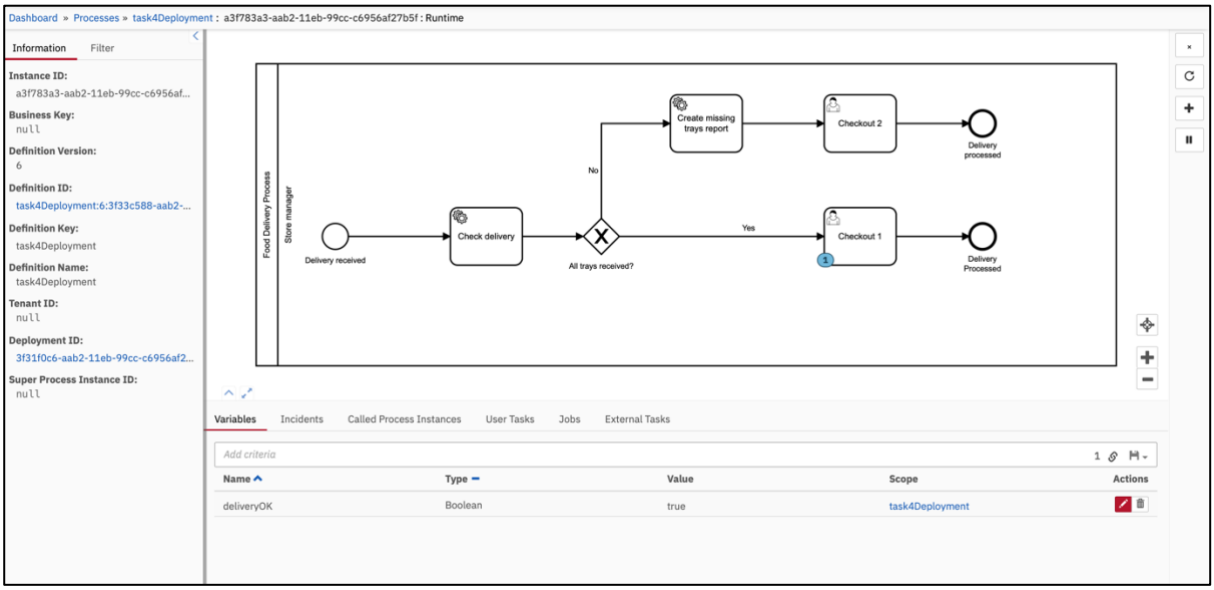


Figure 4.3: *Deployed Business Process Model with a token in checkout 1*

As seen in the figure above (see figure 4.2 & 4.3), the token goes through a different route based on the Boolean value. The service tasks linked with a Java Class automatically generates a random Boolean value. In a real-world scenario, the value could differ based on the delivery details but for modelling, a random value generator was used.

A Maven project was created with a Java class to implement the service tasks in the above BPMN. While the real scenario could be different due to various factor, for the modelling, a random value is generated to execute the service tasks.

```java
package com.camunda.myBPM.task4Deployment;

import java.util.Random;

//Utilises Java Delegate interface
//this class is implemented by Check delivery service task in BPM
public class CheckDeliveryDelegate implements JavaDelegate {

    @Override
    public void execute(DelegateExecution execution) throws Exception {
        // generates a random value
        Random rando = new Random();
        // sets a random variable
        execution.setVariable("deliveryOK", rando.nextBoolean());

    }

}
```

Figure 4.4: Java implementation for BPM service task

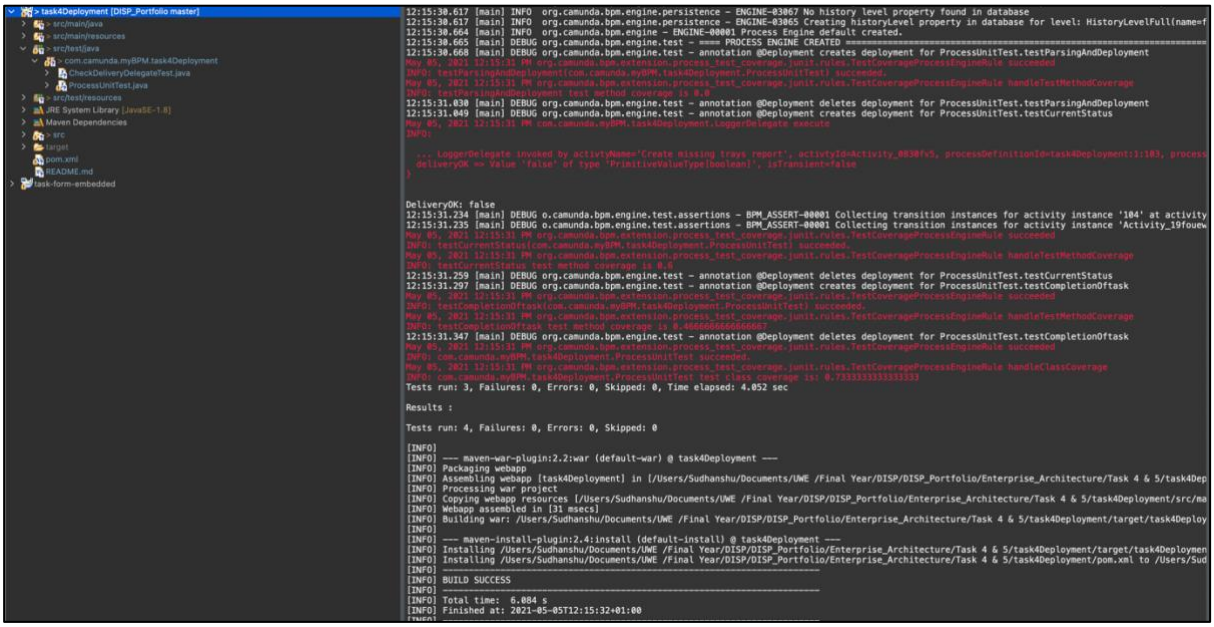


Figure 4.5: Build success of Maven project

The project was finally run aa s Maven install of which the build was a success. The project was then deployed by copying the .war file in the directory of the Camunda workflow engine.

Moreover, initially, a more elaborate business process was deployed using Camunda. It was scoped down to just basic for the final version (see figure 4.1)

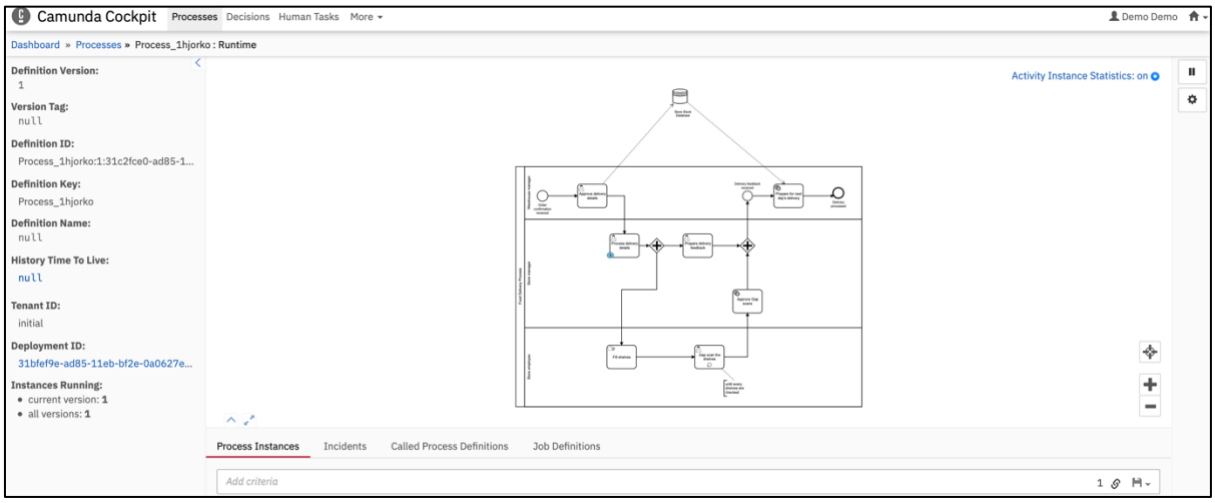


Figure 4.6: Initial Deployed BPM

5. Task 5: JUnit test

To test the deployed model, Junit tests were done. The tests were coped to be a behaviour test or interaction test. The unit test file had different methods and they were called to see if the test passed or failed. While the behaviour test did not validate the result, it was able to check if the input parameters were correct and the implementation was done properly.

```
/**
 * Just tests if the process definition can be deployed.
 */
@Test
@Deployment(resources = "process.bpmn")
public void testParsingAndDeployment() {
    // nothing is done here, as we just want to check for exceptions during
    // deployment
}

// Tests the checkout tasks are linked correctly.
// This test should fail if wrong activity name is used or processInstance variable is changed.
```

Figure 5.1: Parsing and Deployment test

In the above test, the method testParsingAndDeployment was used just to check the links. Upon entering the resources name incorrectly, the test will fail. This test was important to do before carrying on to other tests.

```
@Test
@Deployment(resources = "process.bpmn")
public void testCurrentStatus() {

    // Obtain test run of BPMN
    ProcessInstanceWithVariables processInstance = (ProcessInstanceWithVariables)processEngine().getRuntimeService().startProcessInstanceByKey(PROCESS_DEFINITION_KEY);

    // Obtain the value of the deliveryOK variable
    boolean deliveryOK = (boolean)processInstance.getVariables().get("deliveryOK");
    System.out.println("DeliveryOK: " + deliveryOK);


    // Obtain a reference to the current task
    TaskAssert task = assertThat(processInstance).task();

    if (deliveryOK) {
            assertThat(processInstance).isWaitingAt("Activity_066txhw");
            task.hasName("Checkout 1");
            task.isNotAssigned();
        } else {
            assertThat(processInstance).isWaitingAt("Activity_19fouew");
            task.hasName("Checkout 2");
            task.isNotAssigned();
        }

    }
```

Figure 5.2: Current Status test

In this test (see figure 5.2), a method was created to test the status of the token. Here a scenario of **deliveryOK** was passed as the Boolean variable. The test should pass as long as the activityid in BPM is linked correctly. The test should fail if the value of Boolean Variable deliveryOK is set to **not deliveryOK**.

```
// tests the completion of task.
@Test
@Deployment(resources="process.bpmn")
public void testCompletionOftask() {
    ProcessInstanceWithVariables processInstance = (ProcessInstanceWithVariables) processEngine().getRuntimeService().startProcessInstanceByKey(PROCESS_DEFINITION_KEY);

    //Obtain a reference to the current task
    TaskAssert taskAssert = assertThat(processInstance).task();
    TaskEntity task = (TaskEntity) taskAssert.getActual();
    task.delegate("user");
    task.resolve();


}
```

Figure 5.3: Task Completion Task

This test will attempt to delegate the task to a user and resolve it. It will confirm that the task has been completed.

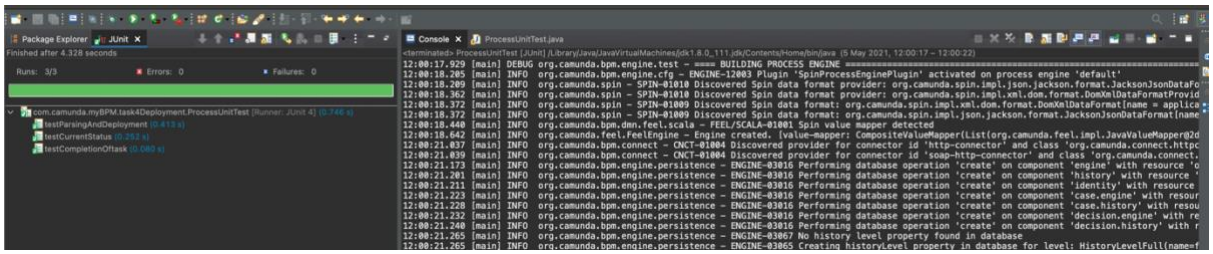


Figure 5.4: Junit Test cases passed

As seen above, all of the test cases passed as expected.

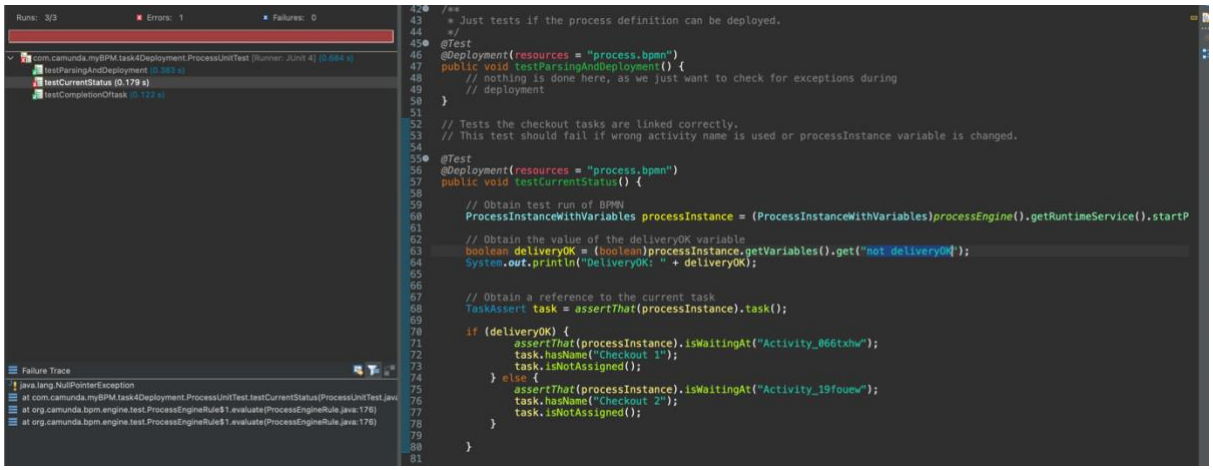


Figure 5.5: Test fails upon changing parameter (intentional)

However, if the parameter is changed the test fails as seen in figure 5.5. This was done intentionally to test the behaviour of the tasks.

Thus, the project passed all three of the JUnit tests, which was used to test the behaviour of the project.

6. Versioning process

For versioning initially, Gitlab was used. However, upon recommendation from the module leader, GitHub was used to commit the changes and kept track of the development of the portfolio components across both semesters. A screenshot of the activities can be seen from the screenshot below.

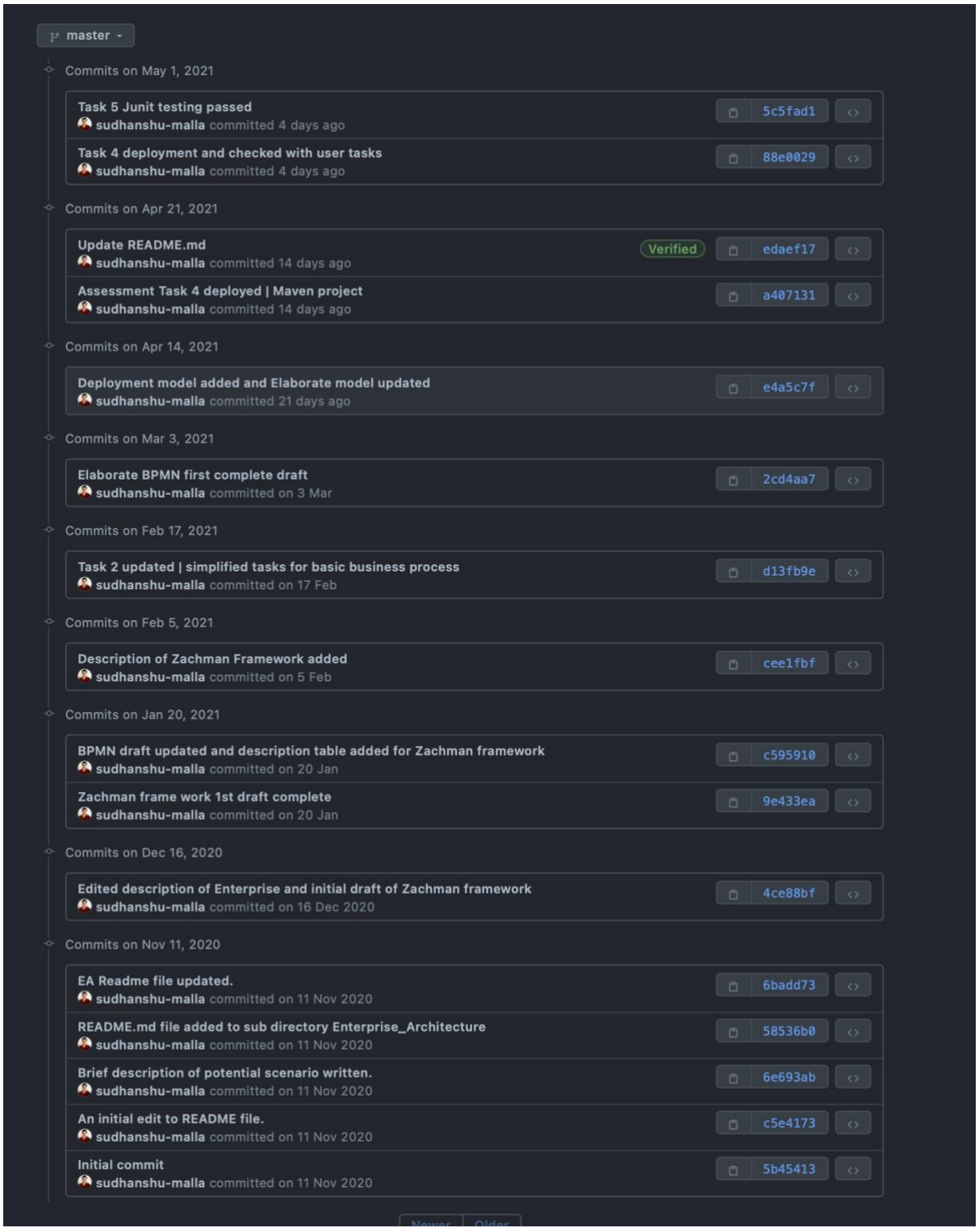


Figure 6.1: Commit history in GitHub

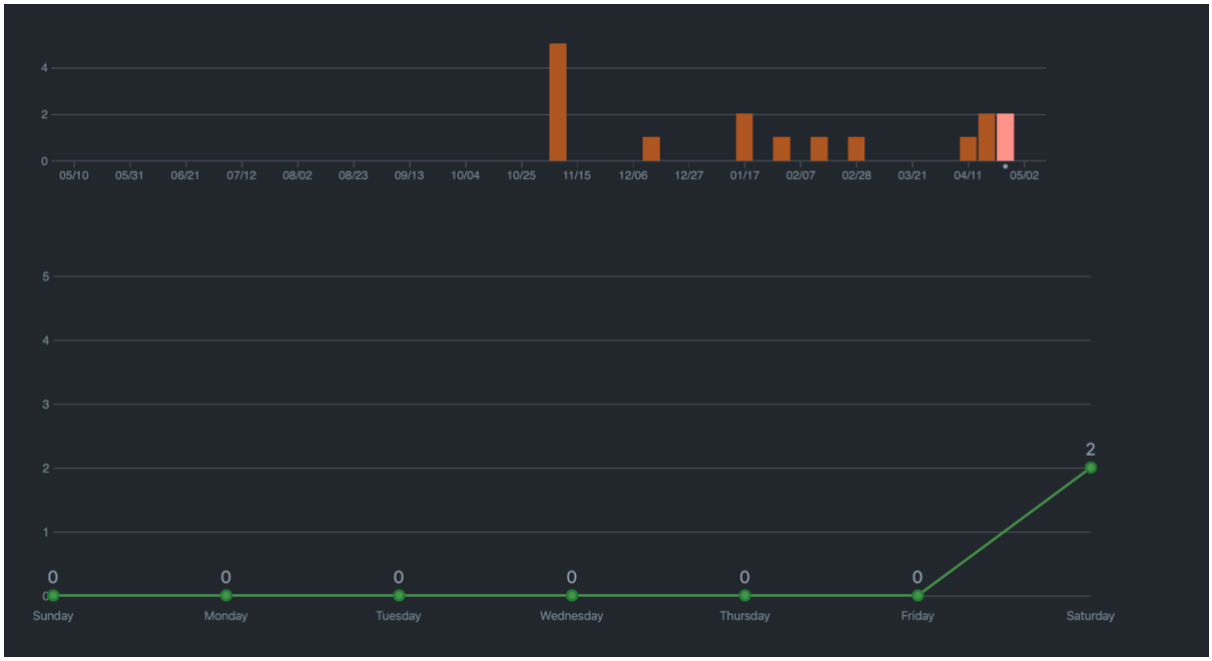


Figure 6.2: Commit insight of the project