TEST CASE : FRD-7

ABSTRACT

* This usecase is to develop a system to manage the information when a client pucrchase products. This usecase involves the administrator of the system and the client who bought the product. Furthermore, this usecase requires the accountant to update the progress or the status of the payments. Therefore, our team make this usecase for make sure that happen.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Introduction of the entire plan

According to the administrator, this usecase is planned to help the accountant be able to update the status of the delivery. Furthermore, the usecase requires investigating the accountant’ insights to perform suitable functions on the system.

Purpose and scope of the plan

This plan aims to develop the system to ensure the trust of the client and the trust of the distributor for our company. The accountant is required to back up every single piece of information of the product that being purchased by the client.The scope of this is data entry and security.

REQUIREMENTS/SPECIFICATIONS-BASED SYSTEM LEVEL TEST CASES

1. State Transition

a. When an agent order product, the accountant will have a function to update the status of payments for that order. If the wrong are wrong the company will be punish and lose trust.

b.

State 1: Agent order products

This state will manage by the system to checked the quantity and quality of the product. There will be a delivery status being create. That status will be update by the accountant

State 2: Status of payments being created

This state also manage by the accountant to check the status of the payments is it pay or not. If not they will send you a notification for it.

c. The accountant will write every information that relate to the agent and input them correctly on the system.Then they will initialize the state where they want to observe and then trigger that event

2. Decision Table

1. The accountant will decide and identify the quantity and quality of the new product that are being purchase by the agents.

2. Then they will create a new delivery note and update the status of payments for the product that the client have pruchased.

3. Lastly, the staff will checked again if the product is good then they input it to the system

3. All pair Testing

* All of the new product shall be checked every times before they been package. The staff and the accountant will always double checks.

4. Use Case Testing

1. The accountant will check the quality first if it not good they will be send back to the distributor and fix

2. Then if the quality is good, the next step will be the staff checking the quantify for ensure that they are enough.

3. Lastly, the staff check your payments if you have pay or not and then they will send it to you.

* Because the payments will relative to banks and many things so there for there will be some mistake:

1. The payment not working

2. The bad connection

TRACEABILITY OF TEST CASES TO USE CASES

1. The payment not working

* The traceability is followed by bi-directional traceability. The system will announce to the client that the payment is not working. And the page will notice the client to choose another payment method.

2. The bad connection

* The traceability is followed by backward traceability. If you have a bad connection the page will not load and you can not pay your order.

TECHNIQUES FOR TEST GENERATION

Techniques used:

* Manual test genenaraion

We use this technique because it will scan all the cases that has no database

* White box-based testing:

We use this because white box-based testing is on code statements, branches, paths, or conditions. It’s low-level testing but it still can use for this use case

EVIDENCE THE TEST CASES, DOCUMENT HAVE BEEN PLACED UNDER

CONFIGURATION MANAGEMENT

REFERENCES