COMSATS University Islamabad (CUI)

Software Requirement Specification

(SRS DOCUMENT)

for

Vendor Management System

Version 1.0

By

Sarmad Tuseef	**CIIT/FA20-BSE-073/ISB**
Zain Mushtaq	**CIIT/FA20-BSE-084/ISB**
Ahmad Mahmood	**CIIT/FA20-BSE-096/ISB**
Fatima Riaz	**CIIT/FA20-BSE-098/ISB**

Supervisor

Mr. Atique Ahmad Zafar

Bachelor of Science in Computer Science (2020-2024)

Table of Contents

1. Introduction

This is the introduction of the **Software Requirements Specification (SRS)** for the Vendor Management System providing an overview of the entire SRS. It includes the purpose, scope, modules and overview of the **SRS**.

1.1 Purpose

This SRS document contains functional and non-functional requirements of Vendor Management System, an iOS and android based application designed especially for vendors and store owners. The purpose of the application is to provide a platform where store owners can order items in bulk from the store owners, by removing the middle party to reduce the extra tie and cost.

1.2 Scope

The Vendor Management System is a system in which store owners have the opportunity to order their products online and manage their records feasibly. This system is valuable for the emerging businesses, they can find the vendors online and can demand any type of product they need. The system will facilitate to accept orders without any time restrictions. The store owners will create their own accounts to use the system. Vendors will manage inventory and sales record. Vendors will access the system using login ID. Vendors will accept the orders based on the items available in stock. Vendor management will store order bills and details. The delivery management will make sure to deliver the product and generate order tracking ID for store owner. The store owner will be able to view all records and details. The payment will be managed through online transactions as well as cash on Delivery. The store owner will rate the services of vendor after the delivery process. The system will be accessed using any mobile with internet connection.

1.3 Modules

Our application will have the following necessary modules:

- Account Management
- Records Management
- Delivery Management
- Order Management

1.3.1 Module 1: Account Management

FE-1: Log in to the application using id.
FE-2: Change passwords and reset phone number and other details.
FE-3: Edit the profile.
FE-4: Log out of the application.

1.3.2 Module 2: Record Management

FE-3: Manage the inventory.
FE-4: Add and manage the sales record.

FE-5: View all the records and reports.
FE-6: Pitch the price for the particular order.

1.3.3 Module 3: Delivery Management

FE-1: Deliver the product selected by the customer from the available items.
FE-2: Generates the product tracking id to track the order by the customer.

1.3.4 Module 4: Order Management

FE-1: Place the order.
FE-2: It will store the customer's order and pay for the items to be selected from the menu.
FE-3: It will manage the order id and the product details (i.e., what is quantity of the product, customer's address etc.)
FE-4: View all the orders that has been placed.
FE-5: Track the order that has been places.
FE-6: Make online payment. There will be no calculation errors.
FE-7: Can accept or decline the vendor's offer.

1.4 Overview

The rest of the document provides the **Overall Description** which includes **Product Perspective**, **User Classes**, **Operating Environment** and **Design and Implementing Constraints**. It then describes the **Techniques** we used to identify the functional requirements and then explains the **Functional Requirements**. After that we describe the **Non-Functional Requirements** and any **Quality Attributes** of the system. At Last, we describe the **External Interface Requirements** which on further includes **User**, **Software** , **Hardware,** and **Communication Interfaces**.

2. Overall Description

This section presents a high-level overview of the product and the environment in which it will be used, the anticipated users, and known constraints, assumptions, and dependencies.

2.1 Product Perspective

The Vendor Management System is a new system that aims at solving problems for store owners and vendors in particular and public in general. In the past people had hard time ordering different items because they had to personally visit the shops. It replaces the current manual and telephone processes for ordering items from a vendor. With Vendor Management System you can use all these services on one platform. The context diagram illustrates the external entities and system interfaces for release 1.0. The system is expected to evolve over several releases.

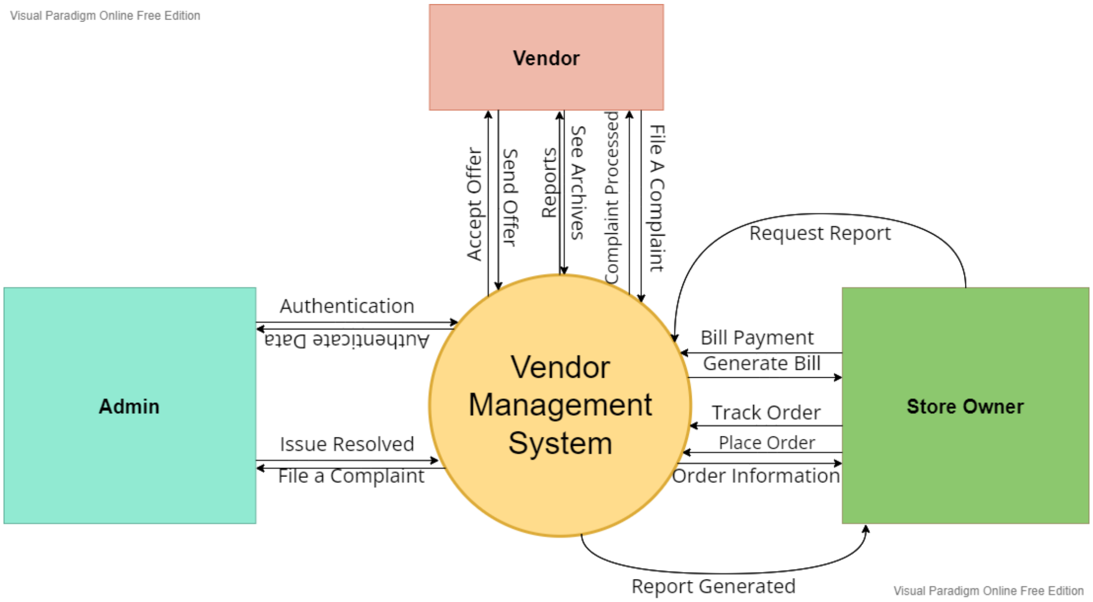


Figure 1: Vendor Management System Context Diagram

2.2 User classes and characteristics

User Class	Description
Store Owner	The store owner is the process influencing user class that orders various items through our application. There are more than 700 prospective store owners who place 30-35 orders every day. Shop owners use the app on average of 4 times a week and place multiple orders. The store owner also used the app to track orders. They also use the app to file a complaint against the order. Around 60 to 70% of orders are placed from home using smartphones. If the store owner wants to see the previous order record, he can do it through the app. The app generates the report for the store owner. Even the store owner can choose different payment methods to pay for the order. The store owner also provides order and app feedback, which keeps our app growing.
Vendor	The vendor is the potential and the application influencing user class that manages multiple store owners. There are several vendors who use the app on a daily basis. The vendor sends various offers to the store owner. If some items are missing and are not available, it checks the inventory and keep it updated. It also checks the reports and keeps track of the store owner's previous orders. If some business owners haven't paid for the order yet, the vendor can send them the payment request. Several vendors manage store owners from their homes using the app. The vendor can also provide feedback about the order and the store owner so that the user experience can be improved.
Admin	The Administrator is the important user class that authenticates vendors and store owners. If there are certain complaints from the vendor or the store owner, the admin will handle and respond to the complaints. The admin will check if some users are using the app for deceptive purposes and block them.

2.3 Operating Environment

OE-1	The application can only be used on mobile devices.
OE-2	The application shall run on Android operating system versions 8.0 and higher.
OE-3	The application shall run on IOS versions 10.0 and higher.

2.4 Design and Implementation Constraints

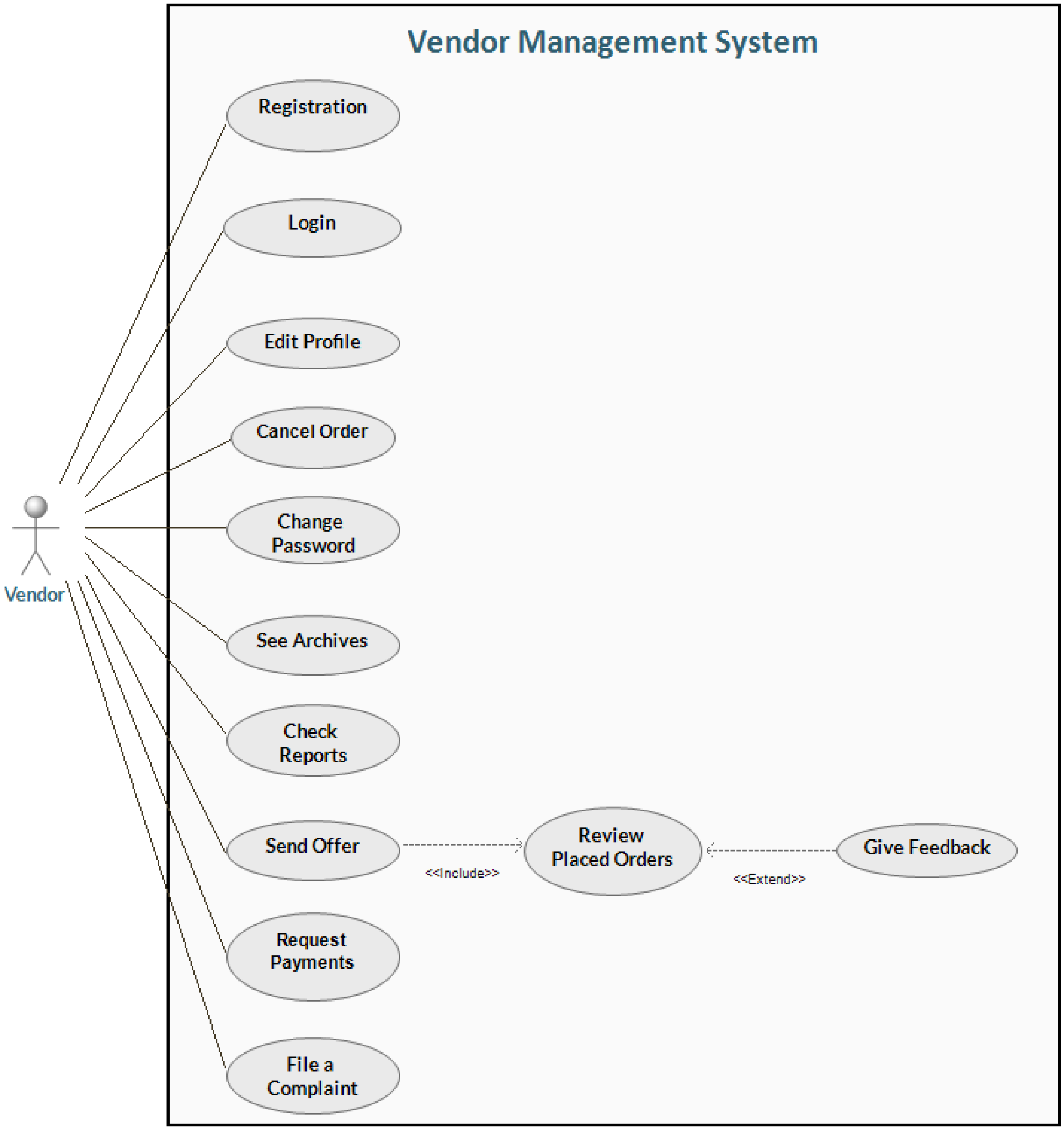
CON-1	The system shall use Mongo DB as database to store data.
CON-2	The coding shall be done using React, Node and Express JS as languages.
CON-3	Android X shall be used as an android library.
CON-4	The system must use an internet connection, otherwise the user will not be able to communicate with the database.

3. Requirement Identifying Technique

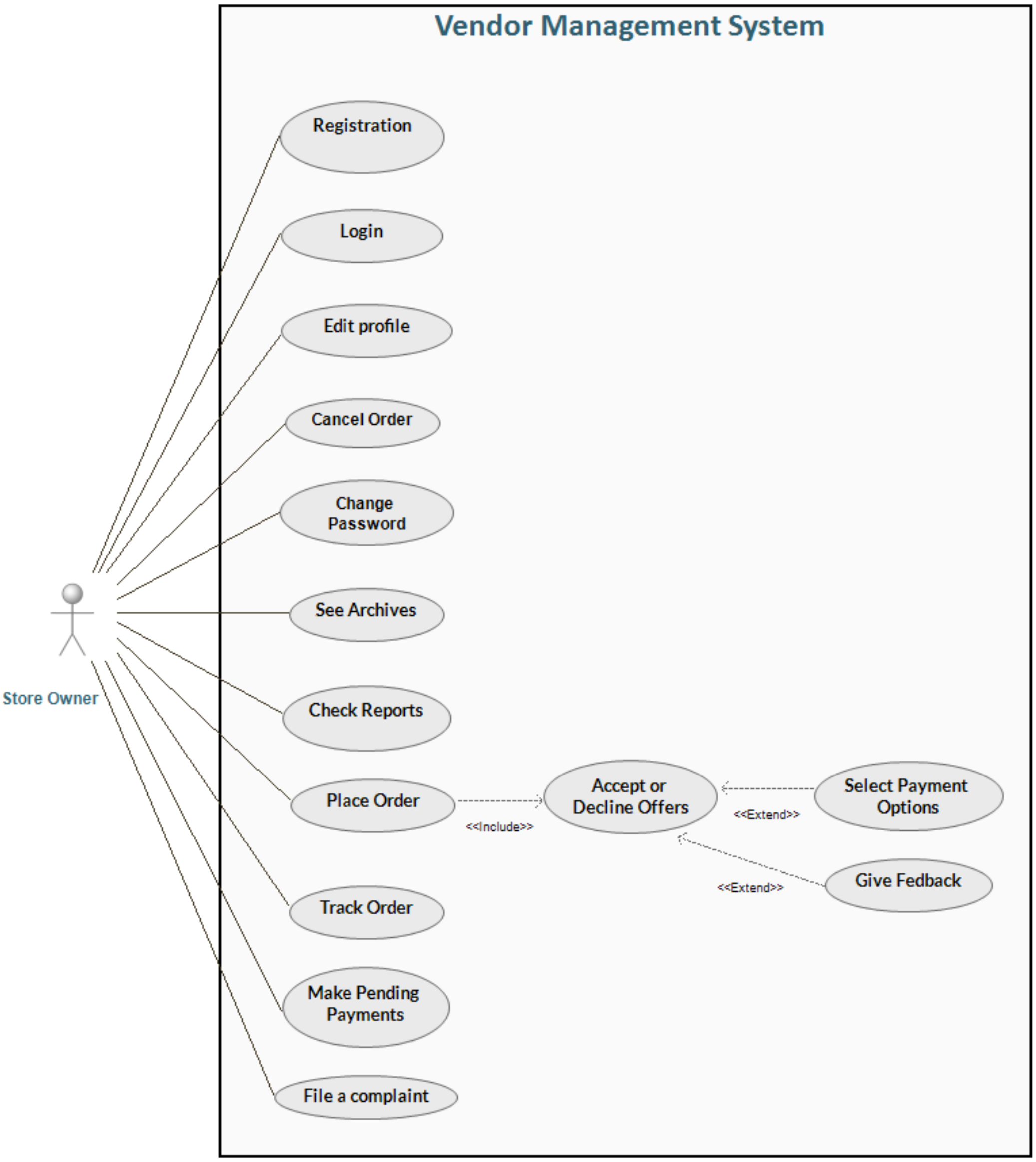
This section of SRS describes the requirements identifying technique which further help to derive functional requirements specification. The selection of the technique will depend on the type of project. For our project, we use the (Use case Diagram + Detail Use case).

3.1 Use Case Diagram

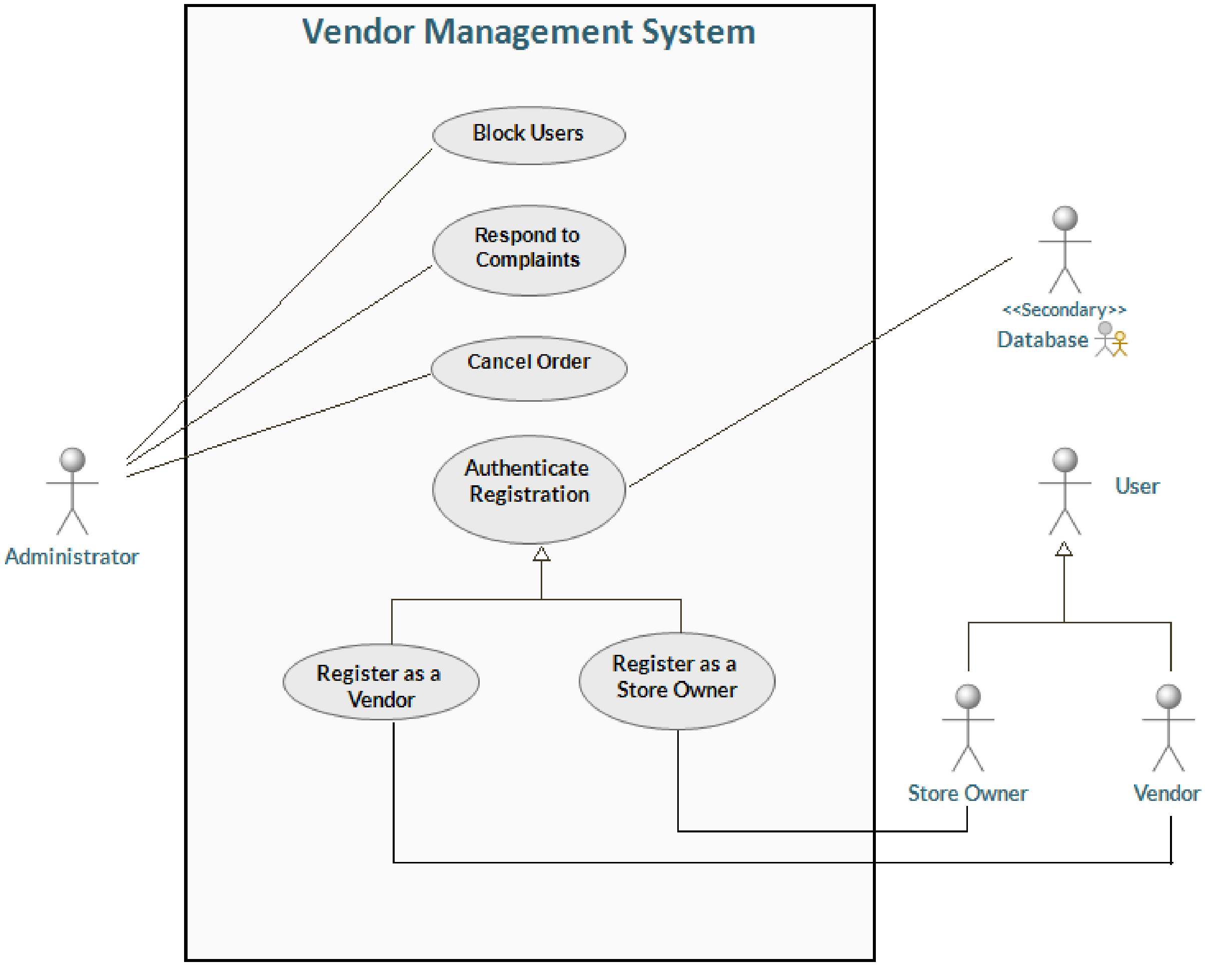
3.1.1 Actor Name: Vendor



3.1.2 Actor Name: Store Owner



3.1.3 Actor Name: Administrator



3.2 Event response Table

3.2.1 Event Response Table: Order Status

ID	Event	Data Element	System State	Exception Condition	System Response		System State
					In case of Yes	In case of No	
1.	Store owner requests for order tracking	Order Status	Shows in progress	If order is not placed	System displays the order status.	System displays an error message that order is not placed.	System displays the current order status.

3.2.2 Event Response Table: Confirmation of User

ID	Event	Data Element	System State	Exception Condition	System Response		System State
					In case of Yes	In case of No	
1	Store Owner and Vendor request for log in to the system.	Store Owner and Vendor email address and password.	System is on the login page.	If the Store Owner or the Vendor is not in the data base.	The system will provide access to the store owner and the vendor.	The system will show a message that store owner or the vendor that your account is not registered.	System will display the home page to the store owner and the vendor.

3.3 Detail Use Case Description

Following are the detail use case descriptions of different use cases of associated actors.

3.3.1 Admin Detail Use Case

UC-1: Block Users

Description: The admin will block the users who misuse the application or the intruders trying to sabotage the application.

Actors:	Administrator (Primary)
Trigger:	To trigger the event, the actor has to click the “Block” button.
Preconditions:	No preconditions.
Postconditions:	The unauthorized users or intruders will be blocked.
Normal Flow:	1. The system will notify the admin to block the user. 2. The admin review the request. 3. The admin block the user. 4. The system will be updated.

Alternative Flows : [Alternative Flow 1 – Not in System] **Alternative Flow 2 – Item not available]**	2a. In step 2 of the normal of the view the admin review the request and if the user is unauthorized 1. The admin will block the user. 2. The use case will resume from 4 step. 2b. In step 2 of the normal of the view the admin review the request and if the user is authentic. 1. The admin will notify the system that user is authentic. 2. The use will resume from step 4.
Exceptions:	No exceptions.
Includes:	None
Business Rules:	Unauthorized users should be blocked immediately with minutes due to security purposes.

UC-2: Respond to complaints

Description: The admin will respond to the complaints made by the customers.

Actors:	Administrator (Primary)
Trigger:	This event will be triggered when a customer post a complaint then admin will be notify and will process the complaint by this use case.
Preconditions:	The complaint should be available to respond.
Postconditions:	The customer issue will be resolved.
Normal Flow:	1. The customer will post the complaint 2. The admin will receive the notification of the complaint. 3. The admin will process the complaint. 4. The issue will be solved.
Alternative Flows:	No alternative flow.
Exceptions:	No exceptions.
Includes:	None
Business Rules:	The complaint should be processed within 3 to 4 working days.
Special Requirements/ Other Information:	The admin should be available to process the request during the working days. His availability is necessary.

UC-3: Authenticate Registration

Description: The database will authenticate the registration of store owner and vendor. It will verify their identity and email address. The admin will notify the store owner and vendor about their authentication.

Actors:	User
Trigger:	This event will be triggered when a vendor or a store owner will register themselves then system will be notified and verify them with the help of this use case.
Preconditions:	No pre conditions.
Postconditions:	The store owner and vendor will be register as authentic users.
Normal Flow:	1. The store owner and vendor will start the registration process. 2. The system will be notified. 3. It will verify their details and information. 4. The admin will notify the store owner and vendor. 5. Store owner and vendor will be verified. 6. The system will be updated.
Alternative Flows:	3a. In step 3 of the normal flow if the details of the user are not valid or email address is not authentic. 1. The system will send him a message to provide valid details and the email that exists. 2. The use case will resume from step 1.
	3b. In step 3 of the normal flow if the details provided by the users are authentic. 1. The admin and system will mark them as verified users and send them a verification by email address. 2. The use case will resume from step 5.
Exceptions:	No exceptions.
Includes:	None
Business Rules:	The verification and authentication process shall not take more than 3 mins.
Special Requirements/ Other Information:	The system and admin should be available to authenticate the users during the working days. Their availability is necessary. The system should be secure enough, so details of user don’t go into wrong hands.

3.2.2 Store Owner Detail Use Case

UC-1: Make Pending Payments

Description: If there are any outstanding payments for the orders he placed, the store owner will look into them. He will be able to go over them and complete the payments by the due date.

Actors:	Store Owner (Primary)
Trigger:	1. To start this use case, the actor will click the "Make Pending Payments" button.
Preconditions:	1. The Store Owner must login before accessing the use case. 2. The store owner must have any outstanding payments
Postconditions:	1. The Store Owner pending payments will be fulfilled 2. The Store Owner's data will be updated
Normal Flow:	1. Store Owner logs in. 2. Store Owner's validation by the system 3. Store Owner selects "Make Pending Payments" option from the home page. 4. System will generate the list of outstanding payments of the store owner 5. Store Owner selects an outstanding payment from list 6. Store Owner reviews the outstanding payment 7. Store Owner selects the payment option 8. System Generates the list of Payment Options
	9. Store Owner completes the payment 10. System updates the Store Owner's data 11. Store Owner will go back to home page. 12. Store Owner logs out

Aternative Flows: **[Alternative Flow 1 – Not in the System]** **[Alternative Flow 2 – Not in the System]** **[Alternative Flow 3 – Didn’t have any pending payments]** **[Alternative Flow 4 – Multiple Payment Options]** **[Alternative Flow 5 – Multiple Payment Options]** **[Alternative Flow 6 – If the user wants to complete multiple pending payments]**	2a. In step 2 of the normal flow, if the store owner is not in the system 1. The system will prompt the user to register as a Store Owner. 2. The user registers as the Store Owner. 3. Restart the Use Case from step 1. 2b. In step 2 of the normal flow, if the store owner is not in the system 3. The system will prompt the user to close the application. 4. The user exits 4. In step 4 of the normal flow, if the store owner doesn’t have any outstanding payment 1. The system will generate a return to home page option. 2. Store owner will return back to home page 8a. In step 8 of the normal flow, if the store owner wants to pay with card 1. Store Owner enter card details 2. System will save card details for future use 3. Use Case resume step 9 8b. In step 8 of the normal flow, if the store owner wants to pay by cash 1. System will verify from the vendor if the payment has been made 2. Use case resume from step 9 10. In step 10 of the normal flow, if the store owner wants to complete another pending payment 1. Use Case will resume from the step 4
Exceptions:	8a. In the step 9 of the normal flow, if the store owner enters invalid card details 1. Payment fails 2. Message to re-enter card details 3. Store Owner enters card details

	4. Use case resumes from step 9 8b. in the step 9 of the normal flow, if the store owner has paid the vendor by cash 1. Verification fails 2. System waits until verification is done 3. Use Case resumes from step 9
Includes:	Nothing is included.
Business Rules:	1. If the pending payment is not completed by the due date, an additional fee will be assessed. 2. If the payment is made with a credit card, the amount should be within the card's limit.
Special Requirements/ Other Information:	1. System should update the pending payment list in less than 5 seconds 2. System should process the card payment within 5 minutes 3. Store Owner card details should be secured

UC-2: Login

Description: The user will be able to login to the system by providing his login details and use the system.

Actors:	User
Trigger:	To start this use case, the user will write his login details on the screen.
Preconditions:	1. The user must be registered to login
Postconditions:	1. The user will be logged in.
Normal Flow:	1. The user will login 2. User Validation by the system

Alternative Flows: **[Alternative Flow 1 – Not in the System]** **[Alternative Flow 2 – Not in the System]** **[Alternative Flow 3 – Not in the System]**	2a. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to register as a vendor. 2. The user registers as the vendor. 3. Restart the Use Case from step 1. 2b. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to sign up as a store owner. 2. The user registers as the store's owner. 3. Restart the Use Case from step 1. 2c. In step 2 of the normal flow, if the user is not in the system 5. The system will prompt the user to close the application. 6. The user exits
Exceptions:	2. In step 1 of the normal flow, if the user enters invalid email or password 1. Login fails 2. Message to re-enter email or password 3. The user enters details 4. Use case resumes from step 1 of normal flow
Includes:	Nothing is included.
Business Rules:	There are no business rules that are restricting the functionality of this use case.
Special Requirements:	1. The email should be verified

UC-3: Logout

Description: When the user has finished his work on the app or is no longer using it, he can log out.

Actors:	User
Trigger:	To start this use case, the user will click “Log Out” option.
Preconditions:	1. The user must be logged in.
Postconditions:	1. The user will be logged out of the system.
Normal Flow:	1. The user will login 2. User Validation by the system 3. The user selects "Log Out" option. 4. System log outs.

Alternative Flows: [Alternative Flow 1 – Not in the System]	2a. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to register as a vendor. 2. The user registers as the vendor. 3. Restart the Use Case from step 1.
[Alternative Flow 2 – Not in the System]	2b. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to sign up as a store owner. 2. The user registers as the store's owner. 3. Restart the Use Case from step 1.
[Alternative Flow 3 – Not in the System]	2c. In step 2 of the normal flow, if the user is not in the system 7. The system will prompt the user to close the application. 8. The user exits
Exceptions:	No Exceptions
Includes:	Nothing is included.
Business Rules:	There are no business rules that are restricting the functionality of this use case.
Special Requirements:	1. If the user logs in again, the system should save his information.

UC-4: Select Payment Options

Description: When the store owner checks out or has to pay pending payments, he must choose a payment method from which to pay.

Actors:	Store Owner (Primary)
Trigger:	When the store owner is about to pay, this use case will be triggered. To begin the use case, the system will display the "Select Payment Option" button.
Preconditions:	No pre-condition for this case.
Postconditions:	1. The store owner’s data will be updated.
Normal Flow:	1. The user will login 2. User Validation by the system 3. System generates a “Select Payment Option”. 4. Store Owner selects a payment option 5. Store Owner completes the payment 6. System updates the data 7. Store Owner logs out

Alternative Flows: [Alternative Flow 1 – Not in the System] **Alternative Flow 2 – Not in the system]** **Alternative Flow 3 – Multiple Payment Option]**	2a. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to register as a vendor. 2. The user registers as the vendor. 3. Restart the Use Case from step 1. 2b. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to sign up as a store owner. 2. The user registers as the store's owner. 3. Restart the Use Case from step 1. 4a. In step 4 of the normal flow, if the store owner wants to pay with card 1. Store Owner enter card details 2. System will save card details for future use 3. Use Case resume step 5 4b. In step 4 of the normal flow, if the store owner wants to pay by cash 1. System will verify from the vendor if the payment has been made 2. Use case resume from step 5 4c. In step 4 of the normal flow, if the store owner wants to pay with card and the required amount is not available
	1. System will notify the user and ask him to enter the valid account details. 2. Use case will resume from step 3
Exceptions:	4a. In the step 4 of the normal flow, if the store owner enters invalid card details 1. Payment fails 2. Message to re-enter card details 3. Store Owner enters card details 4. Use case resumes from step 5 4b. in the step 9 of the normal flow, if the store owner has paid the vendor by cash 1. Verification fails 2. System waits until verification is done 3. Use Case resumes from step 9
Includes:	Nothing is included
Business Rules:	1. If the payment is made with a credit card, the amount should be within the card's limit.
Special Requirements/ Other Information:	1. The card details should be secured 2. Online Payment will be completed in less than 5 minutes

UC-5: Place Order

Description: The actor will be able to add items to the cart and place the order.

Actors:	Store Owner (Primary)
Trigger:	To trigger the event, the actor has to click on the "Check Out" button.
Preconditions:	1. The user must login before placing the order. 2. The store owner must provide the valid delivery address and information.
Postconditions:	1. The order of the store owner will be placed. 2. The message regarding order will be sent to store owner.
Normal Flow:	1. The store owner logs in to the system. 2. The store owner writes a description of the item(s) to be ordered and make an offer.
	3. He/she returns to the home page.
Alternative Flows: [Alternative Flow 1 – Not in System] **Alternative Flow 2 – Item not available]**	2a. In step 2 of the normal flow if store owner is not registered with the system 1. The system will prompt the store owner to register with the system. 2. The store owner will register with the system. 3. The use case will restart from Step 1. 3a. In step 3 of the normal flow if the item to be ordered is not available. 1. The system will sent " Item not Found Message". 2. The use case will resume from step 3. 3b. In step 3 of the normal flow if the item to be ordered is available. 1. The item will be displayed. 2. The use case will continue from step 4.
Exceptions:	In step 2 of the normal flow if user enters invalid description or places an order with incomplete information. 1. The store owner will re-enter details and description. 2. Use case will resume from step 2 of the normal flow.
Includes:	1. Accept and decline offers
Business Rules:	The maximum order to be placed should not be more than the batch size.
Special Requirements:	The order details should be saved. The order should be processed in 2 to 3 mins. The details of the order must be secure.

UC-6: Accept/Decline Offers

Description: The actor will accept the offers of his interest otherwise decline the offers.

Actors:	Store Owner (Primary)
Trigger:	To trigger the event, the actor has to "accept" or "decline" the offer.

Preconditions:	The user must login before accepting or declining the offer.
Postconditions:	1. After accepting the offer the item will be added to the cart. 2. After declining the offer the system will not place the order.
Normal Flow:	The store owner logs in to the system. 1. The store owner is verified by the system. 2. The store owner accepts or decline the offer. 3. The system will update the system. 4. He logs out of the system.
Alternative Flows: [Alternative Flow 1 – Not in System] **Alternative Flow 2 – Item not available]**	2a. In step 2 of the normal flow if store owner is not registered with the system 1. The system will prompt the store owner to register with the system. 2. The store owner will register with the system. 3. The use case will restart from Step 1. 3a. In step 3 of the normal flow if the store owner accept the offers. 1. The system will ask him about the Payment options. 2. The order will be placed. 3. The store owner can provide the feedback about the offer and the product. 4. The use case will resume from step 4. 3b. In step 3 of the normal flow if the store owner declines the offer. 1. The item will not be placed for order. 2. The use case will continue from step 4.
Exceptions:	In step 1 of the normal flow if user enters invalid email or password. 3. Login fails. 4. The store owner will reenter email and password. 5. The store owner will insert details. 6. Use case will resume from step 1 of the normal flow.
Includes:	Nothing is included
Business Rules:	The offers should be provided for limited time period such as for one week or two days only.

UC-7: Give Feedback

Description: Vendor will be able to give feedback on orders placed and store owner will be able to give feedback on offers.

Actors:	User
Trigger:	User will click on Give Feedback button.

Preconditions:	1.User must login 2.There should exist orders and offers.
Postconditions:	1.Users will be able to give comments.
Normal Flow:	1.User logs in. 2.User validation by the system. 3.User will select Give Feedback option. 4.User writes a feedback. 5.User sends feedback. 6.User moves back to home page. 7.User logs out.
Alternative Flows: [Alternative Flow 1 – Not in System]	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1.
Alternative Flows: [Alternative Flow 2 – Not in System]	2b. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as store owner. 2. User registers as store owner. 3. Restart Use Case from step 1.
Alternative Flows: [Alternative Flow 3 – Not in System]	2c. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to exit the system. 2. The user exits.
Alternative Flows:	4a. In step 4 of normal flow, if user does not want to give feedback 1. Skip step 4 and step 5. 2. Continue from step 6.

Exceptions:	2a. In step 2 of normal flow, if user enters invalid data 1. Login fails 2. User needs to re-enter login details 3. Again proceed to Step 1 of normal flow
Includes:	It does not include any use case.
Business Rules:	There is no business rule for this use case.
Special Requirements/ Other Information:	The system should not allow to give stars until user have written a review comment.

3.2.3 Vendor Detail Use Case

UC-1: Check Reports

Description: The actors will be able to review his previous performances and make any necessary edits or modifications

Actors:	User
Trigger:	To start this use case, the actor will click the "Check Reports" button.
Preconditions:	1. The user must login before accessing the use case. 2. The user must have any records
Postconditions:	1. The user's data will be updated with the latest information
Normal Flow:	1. The user logs in. 2. User validation by the system 3. The user selects "Check Reports" from the home page. 4. The system creates user records. 5. The user updates any information or reads it. 6. The user saves the report's modifications. 7. The user will go back to home page. 8. The user logs out

Alternative Flows: **[Alternative Flow 1 – Not in the System]** **[Alternative Flow 2 – Not in the System]** **[Alternative Flow 3 – Not in the System]** **[Alternative Flow 4 – Didn’t**	2a. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to register as a vendor. 2. The user registers as the vendor. 3. Restart the Use Case from step 1. 2b. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to sign up as a store owner. 2. The user registers as the store's owner. 3. Restart the Use Case from step 1. 2c. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to close the application. 2. The user exits

perform any changes]	6. In step 6 of the normal flow, if the user didn't perform any changes 1. The user will skip the step 6 2. Use Case resumes on step 7
Exceptions:	1. The user is new to the system and there are no reports to be checked. In this case the user will be notified about it
Includes:	Nothing is included.
Business Rules:	There are no business rules that are restricting the functionality of this use case.
Special Requirements/ Other Information:	1. The system shall generate report in less than 1 second 2. The report should be in a proper format

UC-2: Change Password

Description: If the user wants to change the password for his login in, he can do it.

Actors:	User
Trigger:	The user will click on "Change Password" button.
Preconditions:	This use case requires no preconditions. The user can change his password whenever he wants.
Postconditions:	1. The user's password will be changed.
Normal Flow:	1. The user logs in. 2. User validation by the system 3. The user selects "Change Password" option. 4. System shows the change password page 5. The user enters his current password 6. System verifies the current password 7. The user enters and confirm new password

	8. System saves the data and updates the user’s data 9. The user log outs
Alternative Flows: [Alternative Flow 1 – Not in the System] **[Alternative Flow 2 – Not in the System]** **[Alternative Flow 3 – In Valid Current password]** **[Alternative Flow 4 – Not fulfilling new password requirement]**	2a. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to register as a vendor. 2. The user registers as the vendor. 3. Restart the Use Case from step 1. 2b. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to sign up as a store owner. 2. The user registers as the store's owner. 3. Restart the Use Case from step 1. 6. In step 6 of the normal flow, if the user enters invalid current password 1. System will generate a message “In-valid Current Password” 2. The user will re-enter the current password 3. Use case resumes at step 7 7. In step 7 of the normal flow, if the user does not fulfill the requirements of the new password 1. System will generate a message “Please fulfill all password requirements” 2. The User enters the password 3. Use case resumes at step 8
Exceptions:	1. In step 7 of the normal flow, if the user enters invalid email or password 1. Message to re-enter email or password 2. The user enters details 3. Use case resumes from step 7 of normal flow
Includes:	Nothing is included.
Business Rules:	1. The password should be more than 8 characters and it should include an “Upper Case”, “Lower Case” and “a special character”.
Special Requirements	1. The password should be secure

UC-3: Registration

Description: When the user is using the app for the first time, he will have to register himself.

Actors:	User
Trigger:	The user will click on “Register” button.
Preconditions:	No pre-condition for this case.
Postconditions:	1. The user will be registered.
Normal Flow:	It is the first step. 1. The user will provide his details. 2. System verifies the details through email. 3. The user gets registered
Alternative Flows: [Alternative Flow 1 – in valid email]	2a. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to register as a vendor. 2. The user registers as the vendor. 3. Restart the Use Case from step 1. 2b. In step 2 of the normal flow, if the user is not in the system 1. The system will prompt the user to sign up as a store owner. 2. The user registers as the store's owner. 3. Restart the Use Case from step 1.
Exceptions:	The registration is failed because of wrong information input. It will start again from the step 1 of normal flow.
Includes:	Nothing is included
Business Rules:	1. One person cannot have two login details
Special Requirements:	1. The login details of the user must be secure 2. The account should be registered in less than 10 minutes.

UC-4: Send Offer

Description: Vendor will be able to send offer if store owner has placed an order.

Actors:	Vendor (primary)
Trigger:	User will click on Send Offer button.
Preconditions:	1. Vendor must login. 2. Store owner must have placed an order.
Postconditions:	1. Store owner will receive an offer from vendor.
Normal Flow:	1. Vendor logs in 2. User validation by the system. 3. User selects Review Placed Orders button. 4. Vendor selects Send Offer button. 5. Vendor will go back to home page. 6. Vendor logs out.
Alternative Flows: [Alternative Flow 1 – Not in System]	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1. 2b. In step 2 of normal flow, if the user does not exist in system
Alternative Flows: [Alternative Flow 2 – Not in System]	1. The system will ask the user to exit the system. 2. The user exits.
Exceptions:	2a. In step 2 of normal flow, if user enters invalid data 1. Login fails 2. User needs to re-enter login details 3. Again proceed to Step 1 of normal flow
Includes:	1.Review Placed Orders
Business Rules:	1. The offer will be valid for 2 days only.
Special Requirements	Offers should be shown in 10s.

UC-5: See Archives

Description: Vendor and store owner will be able to view their previous records.

Actors:	User
Trigger:	User will click on Archives button.

Preconditions:	1. The user should login 2. The user should have archives.
Postconditions:	1. User will be able to view the archives.
Normal Flow:	1. User logs in. 2. User validation by the system. 3. User selects See Archives button. 4. System will load and update archives. 5. User will view archives. 6. User will go back to home page. 7. User logs out.
Alternative Flows: [Alternative Flow 1 – Not in System]	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1.
Alternative Flows: [Alternative Flow 2 – Not in System] **Alternative Flows: [Alternative Flow 3 – Not in System]**	2b. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as store owner. 2. User registers as store owner. 3. Restart Use Case from step 1. 2c. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to exit the system. 2. The user exits.
Exceptions:	2a. In step 2 of normal flow, if user enters invalid data 1. Login fails 2. User needs to re-enter login details 3. Again proceed to Step 1 of normal flow
Includes:	It does not include ant use case.
Business Rules:	There is no business rule for this use case.
Special Requirements/ Other Information:	1. The system must save the old records. 2. System must load archives in 10s.

UC-6: Edit Profile

Description: If user wants to edit profile, he will be able to do it.

Actors:	User

Trigger:	User will click on Edit Profile button.
Preconditions:	1.User must login
Postconditions:	1.User’s profile will be edited.
Normal Flow:	1.User logs in. 2.User validation by the system. 3.User selects Edit Profile button. 4.User will modify his profile. 5.User will save the profile with modifications. 6.User will go back to home page. 7.User logs out.
Alternative Flows: [Alternative Flow 1 – Not in System] **Alternative Flows: [Alternative Flow 2 – Not in System]** **Alternative Flows: [Alternative Flow 3 – Not in System]** **Alternative Flows: [Alternative Flow 2 – If User Do Not Want to Modify]**	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1. 2b. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as store owner. 2. User registers as store owner. 3. Restart Use Case from step 1. 2c. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to exit the system. 2. The user exits. 5a. In step 5 of normal flow, if user does not modify his profile 1. Skip step 5 of normal flow
Exceptions:	No exceptions
Includes:	It does not include ant use case.
Business Rules:	1. Modifications should be done on clicking save button. 2. User ID cannot be modified.
Special Requirements:	The user cannot enter number in name section.

UC-7: Request Payments

Description: Vendor can ask for payment from store owner.

Actors:	Vendor (primary)
Trigger:	User will click on Request Payment button.
Preconditions:	1. Store Owner should have pending payments.
Postconditions:	1. Vendor will be able to request payment.
Normal Flow:	1.Vendor logs in. 3.Vendor selects Request Payment button. 4.Vendor requests payment from store owner. 5.Vendor will go back to home page. 6.Vendor logs out.
Alternative Flows: [Alternative Flow 1 – Not in System] **Alternative Flows: [Alternative Flow 2 – Not in System]** **Alternative Flows: [Alternative Flow 3 – Not in System]**	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1. 2b. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as store owner. 2. User registers as store owner. 3. Restart Use Case from step 1. 2c. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to exit the system. 2. The user exits.
Exceptions:	4a. In step 4 of normal flow, if vendor requests payment for order which was already completed 1. An error message will be displayed 2.Vendor will return to home page
Includes:	It does not include ant use case.
Business Rules:	1. If store owner have not paid in due time vendor can ask for extra charge. 2. Vendor can only request payments if store owner have not paid it.
Special Requirements:	System should be able to request within a second.

UC-8: File a complaint

Description: Vendor can file a complaint against a store owner.

Actors:	Vendor (primary)
Trigger:	User will click on File a Complaint button.
Preconditions:	1.Vendor must login
Postconditions:	1.Vendor will be able to file a complaint.
Normal Flow:	1. Vendor logs in. 2.User validation by the system. 3.Vendor will select File a Complaint button. 4.Vendor will write a complaint. 5.Vendor will report this complaint. 6.Vendor will go back to home page. 7.Vendor logs out.
Alternative Flows: [Alternative Flow 1 – Not in System] **Alternative Flows: [Alternative Flow 2 – Not in System]** **Alternative Flows: [Alternative Flow 3 – User do not want to File a Complaint]**	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1. 2b. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to exit the system. 2. The user exits. 4a. In step 4 of normal flow if user does not want to file a compliant 1. Skip step 4 and step five of normal flow. 2. Continue from step 6.
Exceptions:	No exception
Includes:	It does not include ant use case.
Business Rules:	There is no business rule for this use case.
Special Requirements:	The system should respond/take action on the complaint.

UC-9: Review Placed Order

Description: Vendor will be able to review the orders placed by different store owners.

Actors:	Vendor (primary)
Trigger:	User will click on Review Placed Orders button.
Preconditions:	1. Vendor must login. 2. Store owners must have placed some orders.
Postconditions:	1.Vendor will be able to review orders.
Normal Flow:	1. Vendor logs in. 2. User validation by the system. 3. User will select Review Placed Orders button. 4.User review orders. 5. User will go back to home page. 6. User logs out.
Alternative Flows: [Alternative Flow 1 – Not in System] **Alternative Flows: [Alternative Flow 2 – Not in System]**	2a. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to register as a vendor. 2. User registers as vendor. 3. Restart Use Case from step 1. 2b. In step 2 of normal flow, if the user does not exist in system 1. The system will ask the user to exit the system. 2. The user exits.
Exceptions:	No exceptions
Includes:	It does not include ant use case.
Business Rules:	There is no business rule for this use case.
Special Requirements:	The system should create a list of orders that can be viewed.

4. Functional Requirements

Following are the functional requirements of Vendor Management System expressed in the natural language style. Functional requirements are organized by feature as a system feature name and specific functional requirements associated with this feature.

4.1 Use case <Registration>

4.1.1 <Registration.Mobile>

Identifier	1.1
Title	Registration.Mobile
Requirement	The user shall have an app of Vendor Management System named as VMS in their smart phone to register.
Source	Fatima Riaz
Rationale	The user wants to be registered on the app.
Business Rule (if required)	In order to register, user must have the app on their smart phone. User can only register via Smart Phones.
Dependencies	NA
Priority	High

4.1.2 <Registration.Internet>

Identifier	1.2
Title	Registration.Internet
Requirement	The user shall have an internet connection to register.
Source	Fatima Riaz
Rationale	The user wants to be registered on the app.
Business Rule (if required)	In order to register, user must have internet on their smart phone.
Dependencies	1.1
Priority	High

4.1.3 <Registration.Internet.No>

Identifier	1.2.1
Title	Registration.Internet.No
Requirement	If the user doesn’t have internet connection the system shall prompt the user to turn on their internet in order to register.
Source	Fatima Riaz

Rationale	The user wants to be registered on the app.
Business Rule (if required)	In order to register, user must have active internet on their smart phone.
Dependencies	1.1
Priority	High

4.1.3 <Registration.Details>

Identifier	1.3
Title	Registration.details
Requirement	The user shall enter his name, mobile number, CNIC, age, email, and address for registration.
Source	Fatima Riaz
Rationale	The system wants the credentials of the system.
Business Rule (if required)	The user must be 18 years old to register.
Dependencies	NA
Priority	High

4.1.4 <Registration.details.missing>

Identifier	1.3.1
Title	Registration.details.miss
Requirement	If user missed any information, System would prompt user to fill all the important blanks.
Source	Fatima Riaz
Rationale	The system wants consistent data of the user so it can verify the user.
Business Rule (if required)	The user must be fulfilling all requirements.
Dependencies	1.3
Priority	High

4.1.5 <Registration as Store Owner/Vendor>

Identifier	1.4
Title	Registration.storeOwner/vendor
Requirement	System shall prompt user to join as a store owner or vendor.
Source	Fatima Riaz
Rationale	The system wants to register the user either as a store owner or a vendor.
Business Rule (if required)	Person can register as a store owner or vendor but not both.
Dependencies	1.3
Priority	High

4.1.6 <Registration.Confirmation>

Identifier	1.5
Title	Registration.confirmation
Requirement	System shall be able to validates the provided information of user via text or email.
Source	Fatima Riaz
Rationale	The system wants to validate the credentials of the user through email or text message.
Business Rule (if required)	In order to register information must be valid.
Dependencies	1.3
Priority	High

4.1.7 <Registration.re-confirmation>

Identifier	1.5.1
Title	Registration.re-confirmation
Requirement	If user didn’t receive the msg or mail than the system shall be able to let the user try again.
Source	Fatima Riaz
Rationale	The system wants to validate the credentials of the user through email or text message.
Business Rule (if required)	In order to register information must be valid.

Dependencies	1.5
Priority	High

4.1.8 <Registration.complete>

Identifier	1.6
Title	Registration.complete
Requirement	After filling all the related information, the system shall register the user.
Source	Fatima Riaz
Rationale	The system wants to the user to be registered.
Business Rule (if required)	The system shall register user within 1 business day of user submission.
Dependencies	1.5
Priority	High

4.2 Use case <Login>

4.2.1 <login>

Identifier	2.1
Title	Login
Requirement	The system shall allow the User to login to the app.
Source	Zain Mushtaq
Rationale	The user wants to login on the app.
Business Rule (if required)	The user must login to the application in order to use it.
Dependencies	1.6
Priority	High

4.2.2 <login.use.email.password>

Identifier	2.2
Title	login.use.email,password

Requirement	The system shall allow the User to use email address and password to login to the app.
Source	Zain Mushtaq
Rationale	The user wants to login on the app.
Business Rule (if required)	The user needs email and password to login to the application
Dependencies	1.3
Priority	High

4.2.3 <login.authenticate>

Identifier	2.3
Title	login.authenticate
Requirement	The system shall authenticate the user credentials before login to the app.
Source	Zain Mushtaq
Rationale	The user wants to login on the app and his credentials will be validated.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.2.4 < login.authenticate.invalid>

Identifier	2.3.1
Title	login.authenticate.invalid
Requirement	The system shall prompt the user to use valid email address and password if the user provides invalid data.
Source	Zain Mushtaq
Rationale	The system wants the user to enter valid email and password.
Business Rule (if required)	NA
Dependencies	2.3

Priority	High

4.2.5 <login.resetPassword>

Identifier	2.4
Title	login.resetPassword
Requirement	If the user forgets the password, the system shall prompt the user to reset the password via email address or the phone number.
Source	Zain Mushtaq
Rationale	The system wants the user to maintain his credentials.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.2.6 < login.resetPassword.usingEmail>

Identifier	2.4.1
Title	login.resetPassword.usingEmail
Requirement	If the user chooses the email address, the system shall send an email to the user to reset the password.
Source	Zain Mushtaq
Rationale	The system wants the user to reset his password through email.
Business Rule (if required)	The mail should be accessed within three days.
Dependencies	2.4
Priority	High

4.2.7 < login.resetPassword.usingMobileNumber >

Identifier	2.4.2
Title	login.resetPassword.usingMobileNumber

Requirement	If the user chooses the phone number, the system shall send the message to the user’s phone to reset the password.
Source	Zain Mushtaq
Rationale	The system wants the user to reset his password through phone number.
Business Rule (if required)	The messasge should be accessed within three days.
Dependencies	2.4
Priority	High

4.3 Use case <Logout>

4.3.1 <logout>

Identifier	3.1
Title	Logout
Requirement	The system shall allow the user to logout of the application.
Source	Zain Mushtaq
Rationale	The user wants to logout from the app.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.4 Use case <Edit Profile>

4.4.1 <edit.profile>

Identifier	4.1
Title	edit.profile
Requirement	The system shall allow the user to edit his profile.
Source	Ahmad Mahmood
Rationale	The user wants to update his data such as mobile number, email address etc.

Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.4.2 <edit.profile.change>

Identifier	4.1.1
Title	edit.profile.change
Requirement	The system shall prompt the user if user made changes to his profile.
Source	Ahmad Mahmood
Rationale	The user wants to update his profile.
Business Rule (if required)	NA
Dependencies	4.1
Priority	Medium

4.5 Use case <Change Password>

4.5.1 <change.password>

Identifier	5.1
Title	change.password
Requirement	The system shall allow the user to change his password.
Source	Zain Mushtaq
Rationale	The user wants to update his password.
Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.5.2 <change.password.characters>

Identifier	5.1.1

Title	change.password.characters
Requirement	If the user inserted less than 8 characters the password, the system shall have prompt the user.
Source	Zain Mushtaq
Rationale	The user wants to update his password.
Business Rule (if required)	NA
Dependencies	5.1
Priority	Medium

4.5.3 < change.password.specialCharacters >

Identifier	5.1.2
Title	change.password.specialCharacters
Requirement	If the user doesn’t use any special character, the system shall have prompt the user.
Source	Zain Mushtaq
Rationale	The user wants to update his password.
Business Rule (if required)	NA
Dependencies	5.1
Priority	Low

4.5.4 < change.password.digits>

Identifier	5.1.3
Title	change.password.digits
Requirement	If the user doesn’t use any digits or numbers, the system shall have prompt the user.
Source	Zain Mushtaq
Rationale	The user wants to update his password.
Business Rule (if required)	NA
Dependencies	5.1

Priority	Low

4.5.5 < change.password.success>

Identifier	5.2
Title	change.password.success
Requirement	The system shall prompt the user about the password that it has been updated.
Source	Zain Mushtaq
Rationale	The user wants to update his password.
Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.6 Use case <See Archives>

4.6.1 < see.archives >

Identifier	6.1
Title	see.archives
Requirement	The system shall display all archives if the user wants to see them.
Source	Zain Mushtaq
Rationale	The user wants to see his data.
Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.6.2 < see.archives.empty>

Identifier	6.1.1
Title	see.archives.empty
Requirement	If no previous data is available, the system shall prompt to the user.
Source	Zain Mushtaq

Rationale	The user wants to see his data.
Business Rule (if required)	NA
Dependencies	6.1
Priority	Medium

4.7 Use case <Check Reports>

4.7.1 <display.reports>

Identifier	7.1
Title	display.reports
Requirement	The system shall display all the reports if user wants to check them.
Source	Ahmad Mahmood
Rationale	The user wants to see all the reports.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.7.2 < display.reports.notAvailable >

Identifier	7.1.1
Title	display.reports.notAvailable
Requirement	If no reports are available, the system shall prompt to the user.
Source	Ahmad Mahmood
Rationale	The user wants to see all the reports.
Business Rule (if required)	NA
Dependencies	7.1
Priority	Medium

4.8 Use case <Send Offer>

4.8.1 < send.offer >

Identifier	8.1
Title	send.offer
Requirement	The system shall have sent the notification to the user about the offer, if the user has placed an order.
Source	Ahmad Mahmood
Rationale	The system wants to notify the user about the offers.
Business Rule (if required)	The user cannot send more than five offers.
Dependencies	NA
Priority	High

4.8.2 <accept.offer>

Identifier	8.1.1
Title	accept.offer
Requirement	The system shall prompt a message to the user if the offer has been accepted by the store owner.
Source	Ahmad Mahmood
Rationale	The system wants to notify the user if his offer has been accepted.
Business Rule (if required)	NA
Dependencies	8.1
Priority	High

4.8.3 < review.order >

Identifier	8.2
Title	review.order
Requirement	The system shall allow the user to review his order.
Source	Ahmad Mahmood
Rationale	The user wants to review the cart.

Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.8.4 <delete.order>

Identifier	8.2.1
Title	delete.order
Requirement	If the user deletes the order, the system shall prompt to the user that order has been deleted and empty the cart.
Source	Ahmad Mahmood
Rationale	The user wants to delete the order he placed.
Business Rule (if required)	NA
Dependencies	8.2
Priority	High

4.8.5 <give.feedback>

Identifier	8.3
Title	give.feedback
Requirement	The system shall display the box for writing the feedbacks if user wants to provide the reviews against the offers and orders.
Source	Ahmad Mahmood
Rationale	The user wants to give feedback about the order.
Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.9 Use case <Request Payments>

4.9.1 <request.payment.notification>

Identifier	9.1

Title	request.payment.notification
Requirement	The system shall send the notification to the store owner if the vendor request for the payment.
Source	Sarmad Tuseef
Rationale	The system wants to notify the user about the payment.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.9.2 < request.payment.orderID>

Identifier	9.1.1
Title	request.payment.orderID
Requirement	The vendor shall provide the order id to request the payment.
Source	Sarmad Tuseef
Rationale	The vendor wants to request the payment.
Business Rule (if required)	The order id must be integers.
Dependencies	9.1
Priority	High

4.9.3 < request.payment.orderID.notValid >

Identifier	9.1.2
Title	request.payment.orderID.notValid
Requirement	The system shall prompt that the order id is not valid if the vendor provides invalid id.
Source	Sarmad Tuseef
Rationale	The vendor wants to request for the payment.
Business Rule (if required)	The order id must be integers.
Dependencies	9.1
Priority	High

4.10 Use case <File a Complaint>

4.10.1 <file.complaint>

Identifier	10.1
Title	file.complaint
Requirement	The system shall prompt to the user to provide the order id and phone number of the store owner if he wants to file a complaint.
Source	Sarmad Tuseef
Rationale	The user wants to file a complaint.
Business Rule (if required)	The order id must be integers.
Dependencies	NA
Priority	Medium

4.10.2 <file.complaint.orderID.notValid>

Identifier	10.1.1
Title	file.complaint.orderID.notValid
Requirement	If the order id is not correct, the system shall prompt the user to provide valid id to file a complaint.
Source	Sarmad Tuseef
Rationale	The user wants to file a complaint.
Business Rule (if required)	The order id must be integers.
Dependencies	10.1
Priority	Medium

4.10.3 < file.complaint.placed >

Identifier	10.2
Title	file.complaint.placed
Requirement	The system shall send the notification to the user that the complaint has been placed and is in processing phase.

Source	Sarmad Tuseef
Rationale	The user wants to file a complaint.
Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.11 Use case <Payment>

4.11.1 < payment.method >

Identifier	11.1
Title	payment.method
Requirement	From the provided payment options, the store owner shall be able to select an option.
Source	Sarmad Tuseef
Rationale	The user can have multiple payment option and can chose which will benefit him the most.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.11.2 < **payment.method.card** >

Identifier	11.2
Title	payment.method.card
Requirement	If the store owner chooses card payment, the system shall request the user to enter the card information.
Source	Sarmad Tuseef
Rationale	The user wants to pay by card.
Business Rule (if required)	The user must enter valid card details and online payment should be enabled on his/her card. If the payment is made with a card, the amount should be within the card's limit.

Dependencies	NA
Priority	High

4.11.3 < payment.method,card,invalidDetails >

Identifier	11.2.1
Title	payment.method,card,invalidDetails
Requirement	If the card details are invalid, the system shall display prompt the user to submit proper information.
Source	Sarmad Tuseef
Rationale	The system wants to help the user.
Business Rule (if required)	The user must enter valid card details and online payment should be enabled on his/her card. If the payment is made with a card, the amount should be within the card's limit.
Dependencies	11.2
Priority	High

4.11.4 < payment.method.card.validDetails >

Identifier	11.2.2
Title	Payment.method.card.validDetails
Requirement	If the card details are valid, the system shall prompt the user that the payment is successful.
Source	Sarmad Tuseef
Rationale	The user wants his payment to be made.
Business Rule (if required)	The user must enter valid card details and online payment should be enabled on his/her card. If the payment is made with a card, the amount should be within the card's limit.
Dependencies	11.2
Priority	High

4.11.5 < payment.method,payByCash >

Identifier	11.3
Title	payment.method,payByCash
Requirement	If the store owner chooses cash payment, the system shall inform the vendor to receive payment when the order is finished.
Source	Sarmad Tuseef
Rationale	The user wants tp pay for his order.
Business Rule (if required)	The payment should be made as soon as the order is finished.
Dependencies	NA
Priority	High

4.12 Use case <Make Pending Payment>

4.12.1 < make.payment.pending>

Identifier	12.1
Title	make.payment.pending
Requirement	If the store owner has outstanding payments, the system shall notify the user twice a week to complete the payment.
Source	Fatima Riaz
Rationale	The system wants the user to complete his payment.
Business Rule (if required)	If the pending payment is not completed by the due date, an additional fee will be assessed. If the payment is made with a credit card, the amount should be within the card's limit.
Dependencies	NA
Priority	High

4.12.2 < make.pending.payment.list>

Identifier	12.2
Title	make.pending.payment.list
Requirement	If the store owner wishes to complete outstanding payments, the system shall display the user the pending payment list.

Source	Fatima Riaz
Rationale	The system wants the user to complete his payment.
Business Rule (if required)	NA
Dependencies	NA
Priority	High

4.12.3 < make.pending.payment.list.selectPayment >

Identifier	12.2.1
Title	make.pending.payment.list.selectPayment
Requirement	The store owner shall be able to select an outstanding payment from the pending payment list.
Source	Fatima Riaz
Rationale	The system wants the user to complete his payment.
Business Rule (if required)	NA
Dependencies	12.2
Priority	High

4.12.4 < make.pending.payment.list.selectPaymentOption >

Identifier	12.2.2
Title	pending.payment.list.selectPaymentOption
Requirement	The system shall provide payment options to complete the outstanding payment.
Source	Fatima Riaz
Rationale	The system wants the user to complete his payment.
Business Rule (if required)	The user must enter valid card details and online payment should be enabled on his/her card. If the payment is made with a credit card, the amount should be within the card's limit.
Dependencies	12.2
Priority	High

4.12.5 < **make.pending.payment.no** >

Identifier	12.2.3
Title	make.pending.payment.no
Requirement	If there are no outstanding payments, the system shall notify the user.
Source	Fatima Riaz
Rationale	The system wants the user to complete his payment.
Business Rule (if required)	NA
Dependencies	12.2
Priority	High

4.13 Use case < Update Store Owner Information >

4.13.1 < update.storeOwner.infromation >

Identifier	12.3
Title	update.storeOwner.infromation
Requirement	If the store owner has completed any outstanding payment, the system shall update the store owner's information.
Source	Zain Mushtaq
Rationale	The system wants the data to be maintained.
Business Rule (if required)	NA
Dependencies	NA
Priority	Medium

4.14 Use case < Place Order >

4.14.1 < place.order >

Identifier	13.1
Title	place.order
Requirement	If the store owner wants to place an order, the system shall provide the user with the list of vendors from which he/she can select any vendor.
Source	Ahmad Mahmood

Rationale	The store owner wants to place the order.
Business Rule (if required)	The maximum order to be placed should not be more than the batch size.
Dependencies	13.1
Priority	High

4.14.2 < place.order.selectVendor >

Identifier	13.1.1
Title	place.order.selectVendor
Requirement	If the store owner selects a vendor, the system shall provide a list of items that the vendor can provide.
Source	Ahmad Mahmood
Rationale	The store owner wants to place the order.
Business Rule (if required)	The store owner cannot select more than 3 vendors at a single time,
Dependencies	13.1.1
Priority	High

4.14.3 < place.order.itemAvailable >

Identifier	13.2
Title	place.order.itemAvailable
Requirement	If the store owner selects item that are available, the system shall prompt the user to enter the order details.
Source	Ahmad Mahmood
Rationale	The store owner wants to place the order.
Business Rule (if required)	The maximum order to be placed should not be more than the batch size.
Dependencies	NA
Priority	High

4.14.4 < **place.order.itemAvailable.enterDetails** >

Identifier	13.2.1
Title	Place.order.itemAvailable.enterDetails
Requirement	The store owner shall enter valid order details.
Source	Ahmad Mahmood
Rationale	The store owner wants to place the order.
Business Rule (if required)	The maximum order to be placed should not be more than the batch size.
Dependencies	13.2
Priority	High

4.14.5 < **place.order.itemAvailable.invalidDetails** >

Identifier	13.2.2
Title	place.order.itemAvailable.invalidDetails
Requirement	If the store owner enters invalid order details, the system shall prompt the user to enter valid details.
Source	Ahmad Mahmood
Rationale	The store owner wants to place the order.
Business Rule (if required)	The maximum order to be placed should not be more than the batch size.
Dependencies	13.2
Priority	Medium

4.14.6 < **place.order.itemNotAvailable** >

Identifier	13.3
Title	place.order.itemNotAvailable
Requirement	If the store owner selects an item that is not available, the system shall display a message that “item not available”.
Source	Ahmad Mahmood
Rationale	The store owner wants to place the order.

Business Rule (if required)	The maximum order to be placed should not be more than the batch size.
Dependencies	NA
Priority	Medium

4.15 Use case < Track Order >

4.15.1 < track.order >

Identifier	14.1
Title	track.order
Requirement	If the store owner has placed an order, the system shall allow the store owner to track the order.
Source	Sarmad Tuseef
Rationale	The store owner wants to track the order he has placed.
Business Rule (if required)	The store owner must have an internet connection and google maps to track the order.
Dependencies	NA
Priority	High

4.16 Use case < Order Cancel >

4.16.1 <order.cancel>

Identifier	15.1
Title	order.cancel
Requirement	If the store owner wishes to order the cancel, the system shall prompt the user to do it.
Source	Sarmad Tuseef
Rationale	The store owner wants to cancel the order he has placed.
Business Rule (if required)	The store owner must cancel the order within a week after the order has been placed.
Dependencies	NA
Priority	High

4.16.2 <order.cancel.cashback>

Identifier	15.2
Title	order.cancel.cashback
Requirement	If the store owner has cancelled the order, the system shall be able to transfer his money back into his account.
Source	Sarmad Tuseef
Rationale	The store owner wants to cancel the order he has placed and wants the money back that he has paid.
Business Rule (if required)	The store owner must cancel the order within a week after the order has been placed.
Dependencies	NA
Priority	High

4.17 Use case < Respond to Complaints >

4.17.1 <respond.to.comaplaints.reply>

Identifier	16.1
Title	respond.to.comaplaints.reply
Requirement	if there are any complaints of store owner or vendor, the administrator shall reply to them
Source	Sarmad Tuseef
Rationale	The administrator wants to enhance the user experience of the app.
Business Rule (if required)	If the complaint is software related, then the administrator will inform the developer team to improve the bugs and errors.
Dependencies	NA
Priority	Low

4.18 Use case < Block Users >

4.18.1 <block.users >

Identifier	17.1
Title	block users

Requirement	The administrator shall be able to block registered users.
Source	Sarmad Tuseef
Rationale	The administrator wants to block.
Business Rule (if required)	If the user has rating of 2.5 or below, and if any user is doing scam. It will get blocked.
Dependencies	NA
Priority	High

4.19 Use case < Cancel Order >

4.19.1 <order.cancel>

Identifier	18.1
Title	order.cancel
Requirement	The administrator shall be able to cancel order of vendor and store owner
Source	Sarmad Tuseef
Rationale	The user wants to cancel the order.
Business Rule (if required)	If any exception occurs administrator must be able to cancel any order
Dependencies	NA
Priority	Medium

5. Non-Functional Requirements

Following are the non-functional requirements of Vendor Management System

5.1 Performance

PE-1	The system shall be able to authenticate the user details to register the user within 10 seconds 90% of the time.
PE-2	The system shall be able to authenticate the user details within 10 seconds 90% of the time.
PE-3	The system shall be able to process the payment within 5 seconds 95% of the time.
PE-4	The system shall send email to user within 5 seconds after a user request for password change 90% of the time.
PE-5	The system shall forward the complaint to the admin within 5 seconds 95% of the time.

PE-6	The system shall update the user’s data within 2 seconds after it is changed 95% of the time.
PE-7	The system shall display the confirmation message within 1 second 98% of the time.

5.2 Security

SE-1	The system shall lock a user’s account after five constant unsuccessful login attempts within a period of five minutes.
SE-2	A user shall have to change the temporary once time password given by the security administrator to a previously unused password instantly following the first successful logon with the temporary password.
SE-3	All network transactions involving personal or financial information shall be encrypted.
SE-4	The system shall permit store owners to view their own past orders but not other store owners orders.
SE-5	Only the authorized vendors shall be able to send an request on an order.
SE-6	The system shall ensure integrity of customer account information.

5.3 Reliability

REL-1	The system shall stay offline for at least 15 minutes before it considerably affect the operations performed if system crashes.
REL-2	The system must perform without failure in 95 percent of use cases during a month.

5.4 Usability

USE-1	Store owner shall be able to place an order in an average of three minutes and in a minimum of five minutes, 95 percent of the times.
USE-2	9 out of 10 newly registered store owner who have never used Vendor Management System before shall be able to place an order in no more than ten minutes.
USE-3	User shall be able to get to its desired page in no more than three mouse clicks.
USE-4	User shall be able to complete task correctly without needing help, 98 percent of the times.
USE-5	User shall be able to learn and navigate through interface easily.
USE-6	User shall be able to find a specific button buried somewhere in menus in no more than three tries.

5.5 Software Quality Attributes

Availability-1	The Vendor Management System shall be available to the users 99.9 percent of the time in weekdays during 10:00 AM to 4:00 PM and 95 percent available in weekdays from 8:00 AM to 12:00AM.
Installability-1	The system shall retain all data of the user and convert it into new format when installing a new version of software.
Robustness-1	The system shall save the order details provided by the store owner if the system goes down.
Integrity-1	The system shall validate the backup copy with the original to prevent any conflicts.
Integrity-2	The system shall protect against unauthorized deleting or updating of data.

6. External Interface Requirements

6.1 User Interfaces Requirements

UI-1	The buttons/icons should be interactive in the sense they should provide guidance as to what each of them is for.
UI-2	The color scheme should be aesthetic and pleasing to the user's eye.
UI-3	The size of the pop-ups should be as such as to not make the screen look congested.
UI-4	Topics should be displayed in proper format and arrangement.
UI-5	The interface shall be simple in a way that user should be able to perform the main tasks in less than 4 clicks.

6.2 Software interfaces

SI-1	The system shall resolve common problems of the users through an AI which will resolve them automatically
SI-2	The system shall verify the payments made by the user with the help of Payment system.
SI-3	The system shall return the payments made by the user in case of any inconvenience made by anyone.
SI-4	The user shall be able to track the shipment with the help of tracking system.

6.3 Communications interfaces

CI-1	If the user wants to reset his password, the system shall send message or email for validation of the user.
CI-2	The system shall verify the user credentials through email.

7. References

This SRS refers to the following documents and resources:

Articles from Conference Proceedings (published)
B. Angrian and T.R. Saharoni. "Development of vendor management and e-Procurement systems using android platform" in IOP Conference Series: Materials Science and Engineering, 2019

World Wide Web
https://kissflow.com/procurement/vendor-management/guide-to-vendor-management-system/
https://www.investopedia.com/best-vendor-management-systems-5115463