ALBUKHARY INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTING AND INFORMATICS
SEMESTER 1 2023/2024

CCC2143
SOFTWARE ENGINEERING
GROUP PROJECT ASSIGNMENT (20%)

PROJECT TITLE	E-commerce platform for social business products and services	
GROUP LEADER	Subhija Golic	AIU21102146
GROUP MEMBERS	Raghad Rami MUqbel Ahmed Raghad Al-yatim Rehima Sheikhmohammed Kemal Thinley Yeshey Choden	AIU22102189 AIU22102082 AIU21102138 AIU22102188

For Examiner's use only

ITEMS	MARKS
REPORT - 40	
PRESENTATION - 20	
TOTAL (60)	

Table of contents

Introduction

The purpose of this report is to gather small social businesses operating in Universities by using an all-in-one freelance application and grow the social business into an international economy easily. The software user requirements are talking about our ways to make the platform existent, system requirements to help with understanding the business, system models through the diagrams we presented, how we plan on testing our application and more details about said things. We are planning to make our application as user friendly as possible making it easier for our customers to understand and use platform.

Objectives

What our project is focusing on is making a website that will help social businesses in universities to bloom and operate all in one place.

The objectives and parts we were focusing on in this project are:

- The user and system requirements are helping to make a difference between functional and nonfunctional parts where rules users and vendors have to follow are thoroughly described explaining functional and nonfunctional parts.
- Other types of requirements, such as performance, organizational and external requirements and provided input on them.
- The system models we used in this project are activity diagram, use-case diagram, sequence, class and state diagram and provided input on the same ones.
- To test our product we fully opted for a few testing strategies which are: unit testing, system testing, scenario testing, component testing, release testing and requirement based testing explaining what and how actual testing in our project will happen.

Requirement Analysis

User requirements

Functional:

1. Users will be able to create accounts. There will be two different types of accounts. Customer’s account and vendor/seller’s account.
2. Social business vendors should be able to showcase their products and services on the e-commerce platform.
3. Customers should be able to view and purchase products and services from the website.
4. Customers should be able to make payments through the website.
5. Business vendors should be able to generate a sales report through the website that lists the sales of the shop for the past 30 days.

Non-functional:

6. When creating an account, the seller should upload identification documents to the website for verification.

7. The admin should run a functionality and security checkup every 3 months and should be available to be contacted in case of unexpected errors or malfunctions in the website.

8. After placing an order, a customer can track it via a third-party website, not the original platform.

9. Customer service should be accessible to all customers.

System requirements

Functional

1. Users will be able to create accounts. There will be two different types of accounts. Customer's account and vendor/seller's account.	**1.1** To open the customer's account, the user is only required to provide a valid email address, first and last name, mailing address, and an 8-character password. **1.2** To open a vendor's account a user is required to provide a valid email, the vendor's name, a brief description of their business, their warehouse or office address, and an 8-character password. Then for verification, the owner should also upload a photocopy of their identification documents. **1.3** Then both types of users can sign-in to their respective account using their email and password.

	1.4 Only the admin can view the information of the sellers' and customers' accounts. **1.5** Both types of users should be able to sign-out of their accounts by pressing the sign-out button on the user interface.
2. The e-commerce platform should have a page for vendors to showcase their products.	**2.1** Each business vendor can list their products in the platform in a specific products page. **2.2** This product page appears after registering as a vendor on the website. **2.3** This page only appears to business vendors and can only be accessed and edited by them. **2.4** The page should have a separate field for each product, with 2 mandatory subfields one that holds the price, and another one that holds product description and specification.
3. Customers should be able to view and purchase products and services from the website.	**3.1** Any customer can view the product on the website. **3.2** A customer can add items to their cart only after creating an account and signing in to it.

	3.3 A customer can check-out the items or purchase a service only being signed in to their account. **3.4** A customer can display the items in their cart whenever needed.
4. Customers should be able to make payments through the website.	**4.1** A customer can choose between different payments through the website (e.x. e-wallet, credit/debit card, PayPal). **4.2** After making payment a customer can track their order through a third party website.
5. Business vendors should be able to generate a sales report through the website that lists the sales of the shop for the past 30 days.	**5.1** A sales report is automatically generated by the website at 11.59 PM of the 30th day of each month. **5.2** Those reports will contain the total number of purchases of the month, with details of each individual item or service that has been purchased from the vendor. The details will contain the item's date of purchase, price, the details of the purchaser, and the payment method **5.3** additional reports can be generated by the seller at any time. Those reports will summarize the sales of thirty days before the day they were generated.

	5.4 sales reports can only be generated and viewed by the seller.

Non-functional

3.3 Order tracking is done by a third-party website for all customers to avoid server overload.

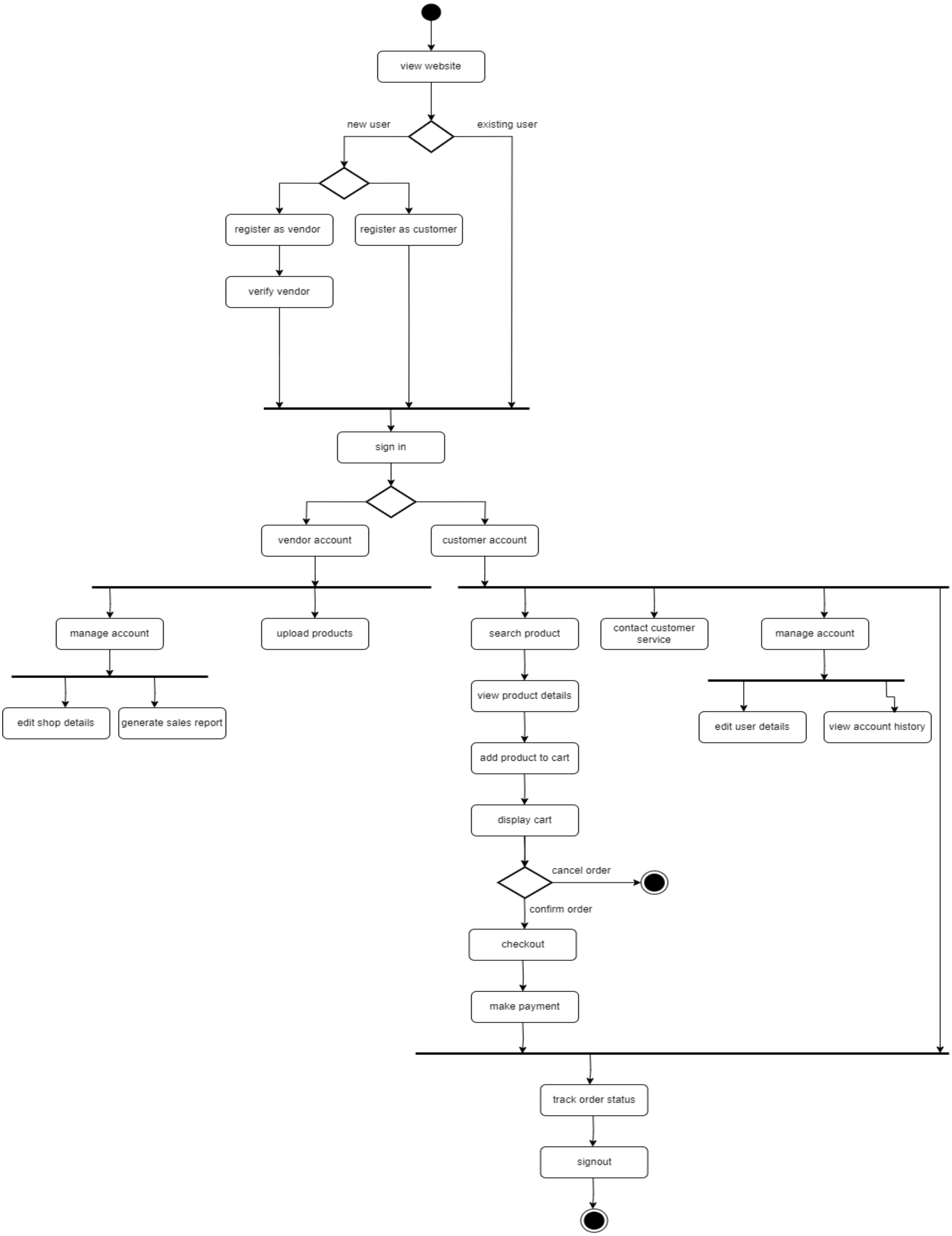
1. **Performance requirements:**

4.1 The platform shall not take more than 3 seconds to respond to clients requests during peak time.

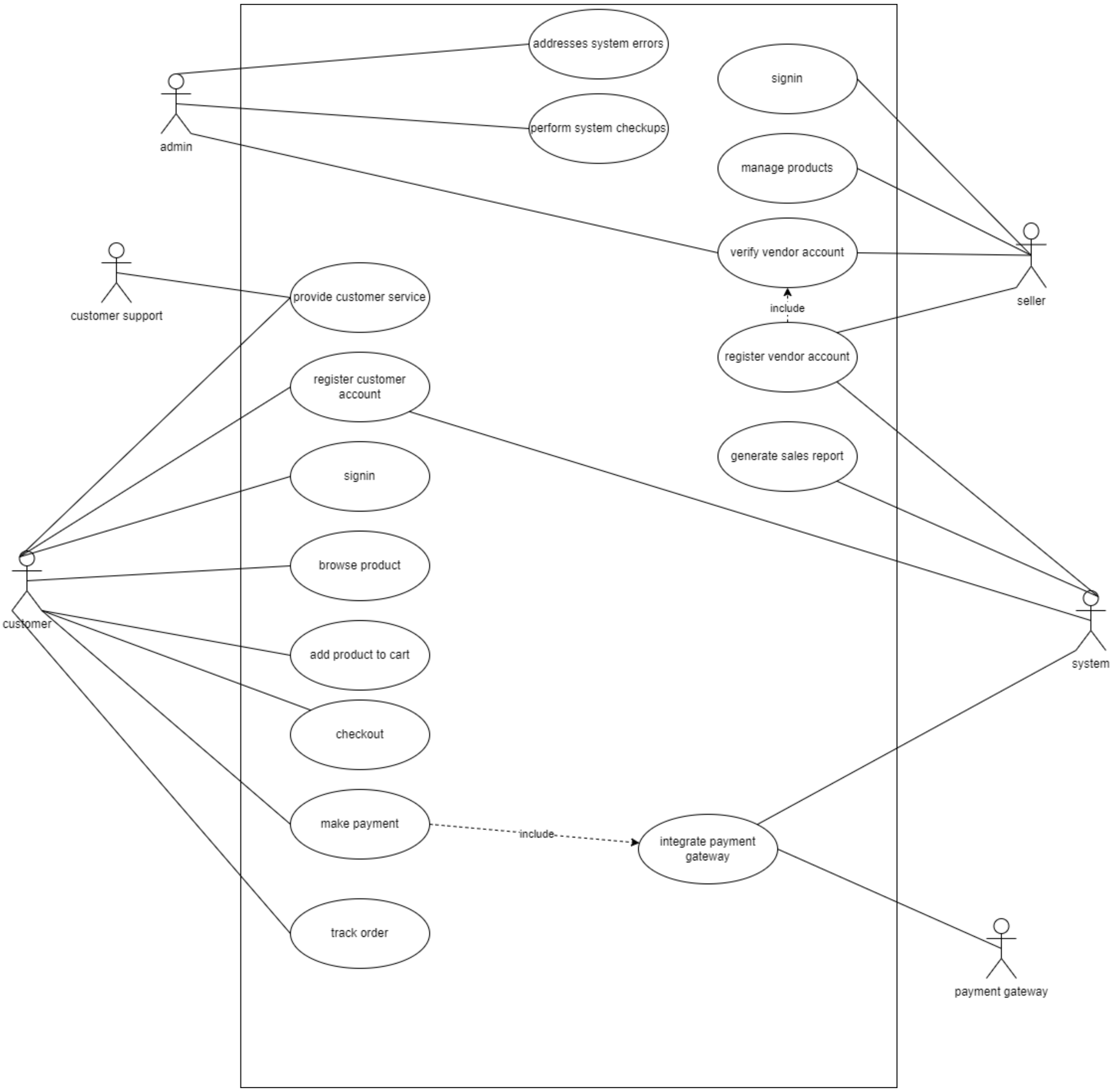
4.2 Users should be able to sign in and out of their accounts in no more than 1 second.

System Models

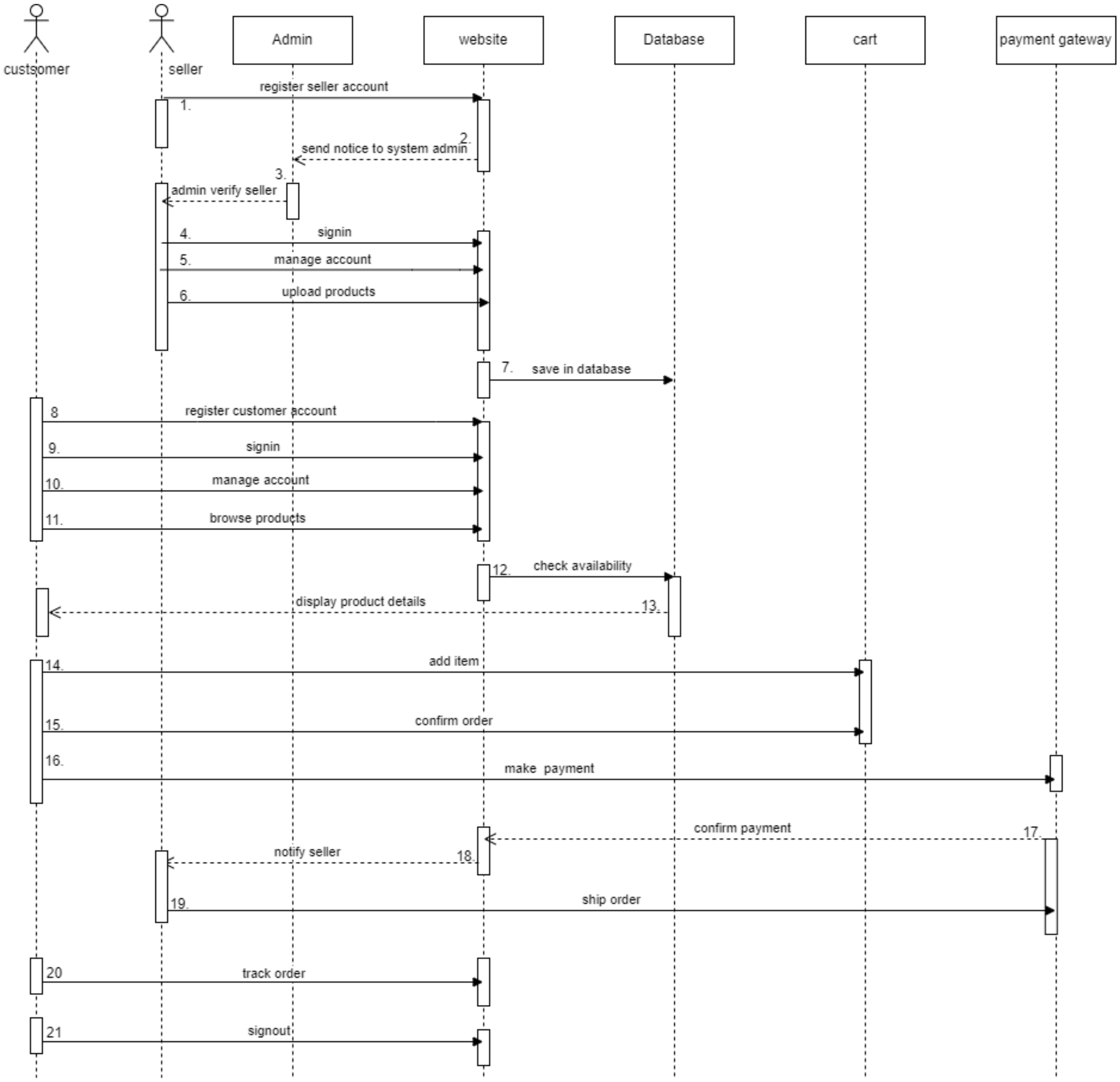
1. Activity diagram



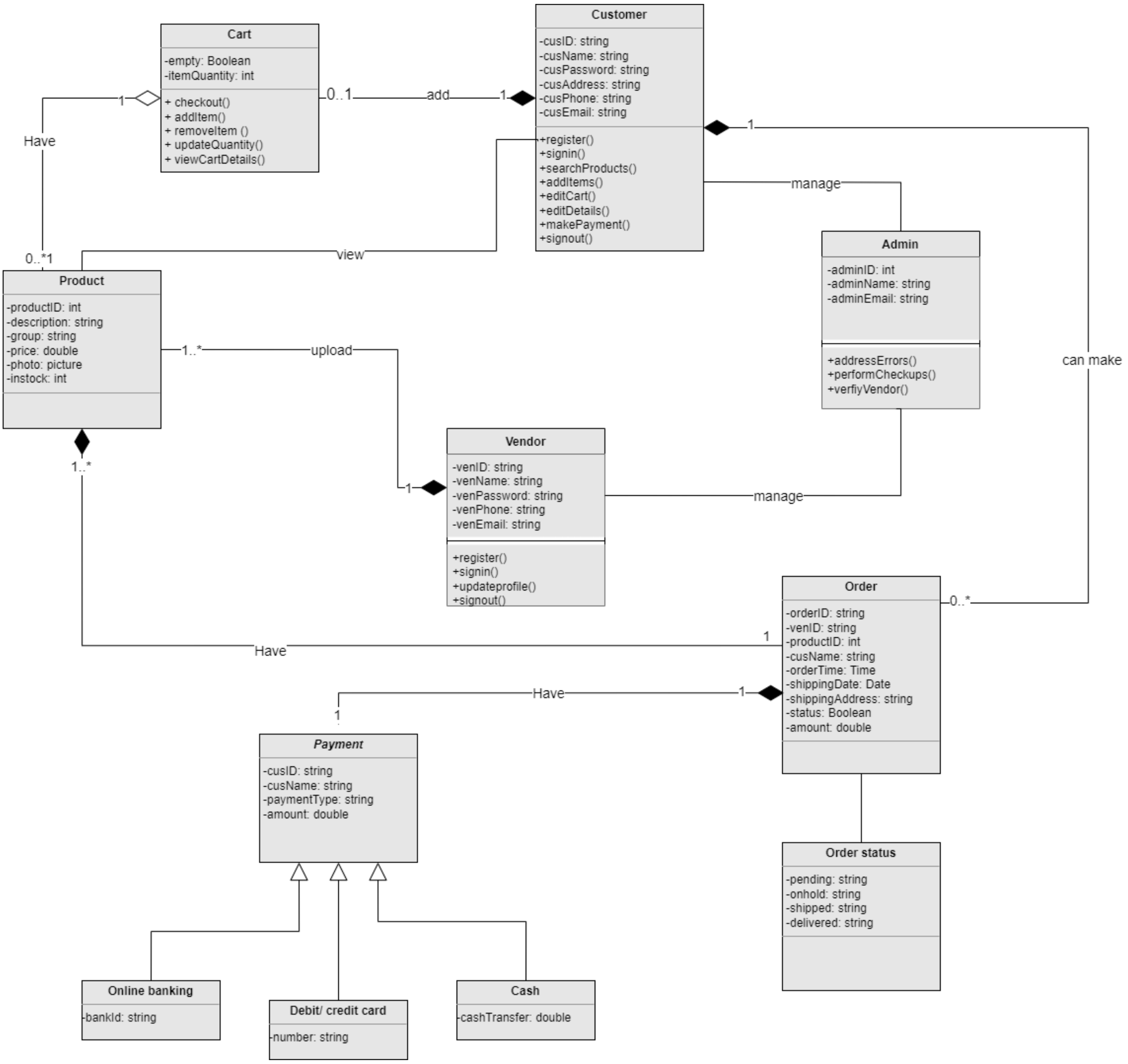
2. Use case diagram



3. Sequence diagram

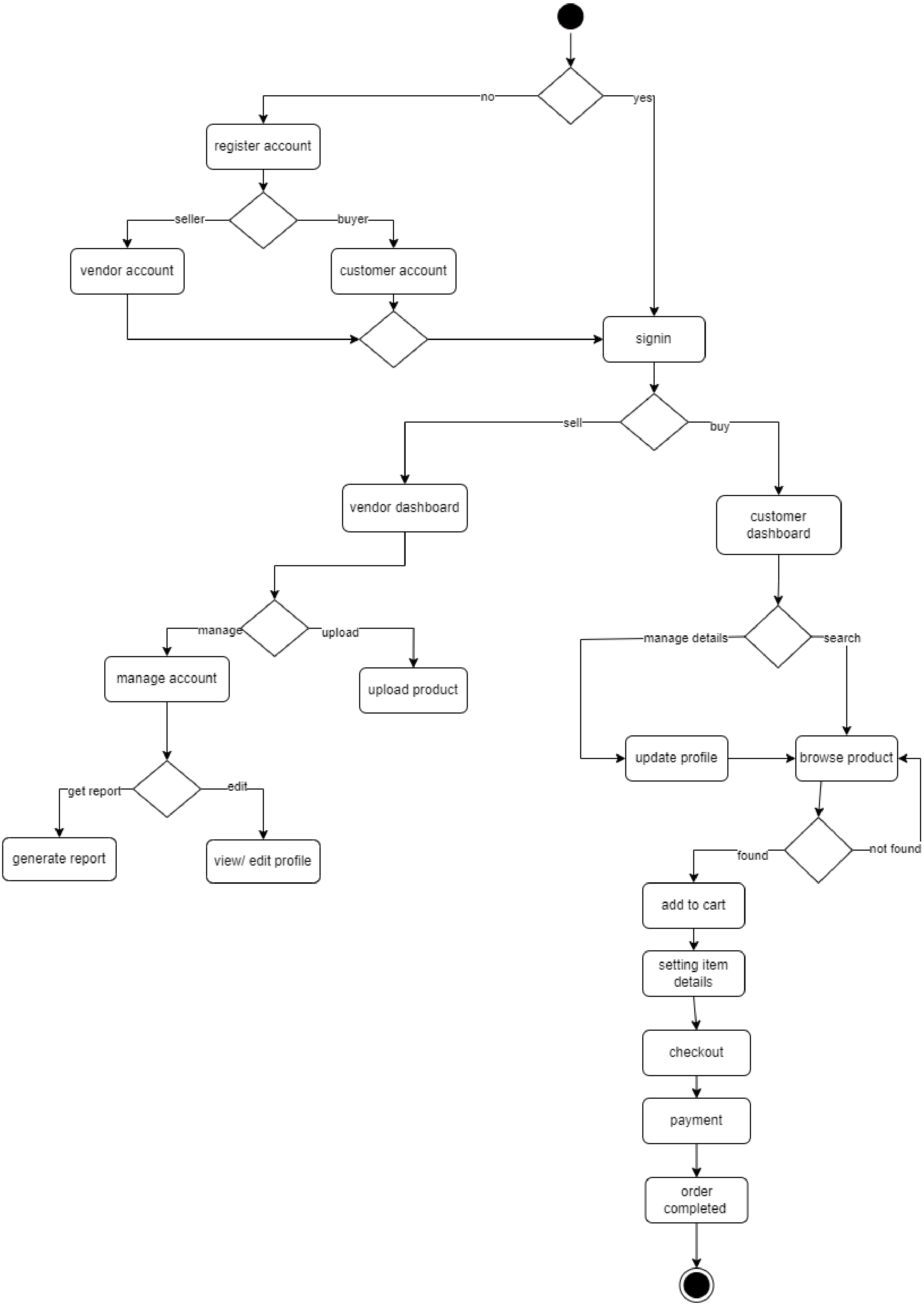


4. Class diagram

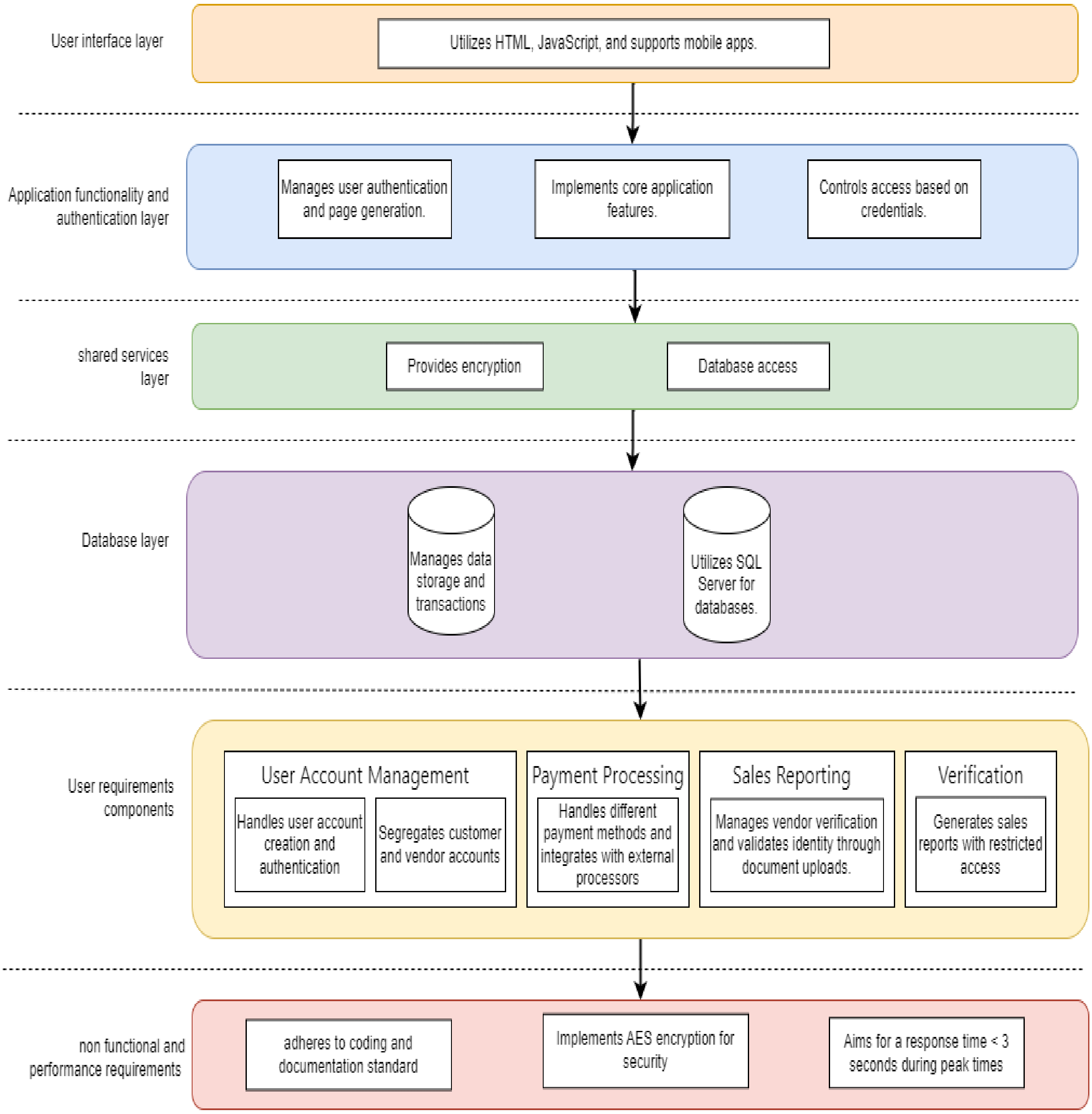


1. Admin has an association with customers and vendors. Likewise, the customer has an association with the product as it can view the products on the website.
2. Customer has a composition relationship with the cart and the order class as both are dependent on the customer. It cannot exist outside of the customer.
3. Vendor also has a composition relationship with the product as the product class cannot exist without the vendor.
4. Product has an aggregation relationship with the cart.
5. The order has a composition relationship with the product.
6. The payment has a composition relationship with the order.
7. Online banking, card and cash inherits the payment class.
8. Order and order status have association relationships.

5. State diagram



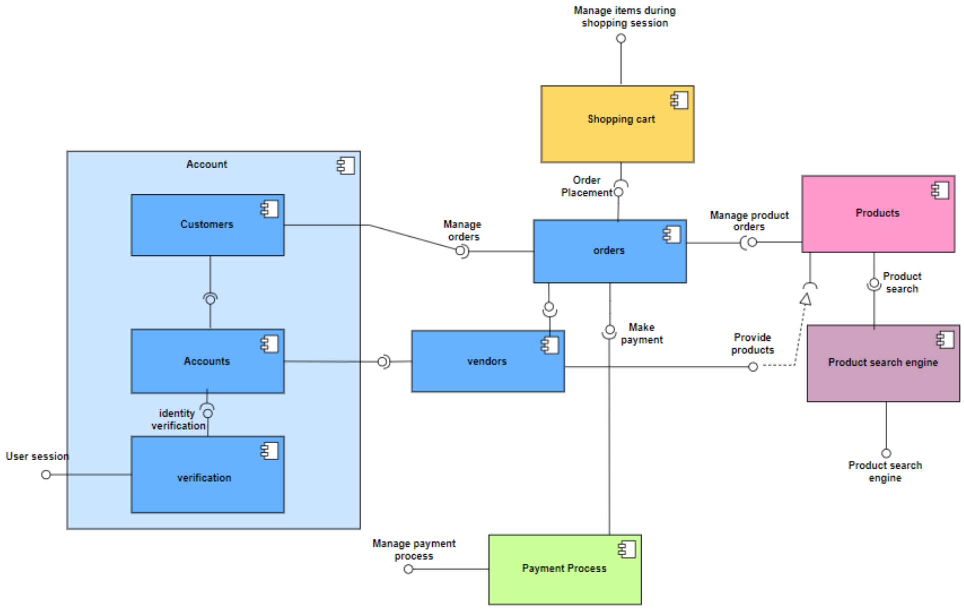
System Architecture


User interface layer
Utilizes HTML, JavaScript, and supports mobile apps.
Application functionality and authentication layer
Manages user authentication and page generation.
Implements core application features.
Controls access based on credentials.
shared services layer
Provides encryption
Database access
Database layer
Manages data storage and transactions
Utilizes SQL Server for databases.
User requirements components
User Account Management
Handles user account creation and authentication
Segregates customer and vendor accounts
Payment Processing
Handles different payment methods and integrates with external processors
Sales Reporting
Manages vendor verification and validates identity through document uploads.
Verification
Generates sales reports with restricted access
non functional and performance requirements
adheres to coding and documentation standard
Implements AES encryption for security
Aims for a response time < 3 seconds during peak times

Our e-commerce system architecture is designed with a layered approach to ensure modularity and clarity. The architecture comprises key layers responsible for distinct functionalities, ensuring scalability, maintainability, and performance.

This architecture facilitates a clear separation of concerns, making the system robust, scalable, and aligned with user and non-functional requirements. Further details can be found in the diagram above.

Component-based model



The above components diagram shows the key components and processes of our e-commerce platform as well as the connections between them. Our reusable components are the following: payment processing, product search engine, shopping cart, and verification.

The relationship between the verification and account is that the user would only be allowed to enter the account only after verifying its access, and then the system will distinguish the user to vendors or customers, each account type has specific access levels and functionalities. Vendors can list their products on the platform. As well as they can access order details related to their specific products. Customers will be able to place orders by adding their desired items to the shopping cart, and also search for specific products using the search engine which aids users in finding desired products efficiently. Once orders are confirmed, users proceed to the payment stage. This involves managing payment details and completing the transaction.

Software testing strategy

The goal of testing software is to ensure that it performs as intended and to identify any bugs before it is put into use. Testing will primarily examine our platform’s performance, security, compatibility, and functionality (Taley & Pathak, 2020). In addition, testing our e-commerce platform aids in determining whether the application complies with all guidelines, protocols, and standards necessary for safe transactions done by our users. Our E-commerce platform will be evaluated, verified, validated, and tested with the aid of software testing stages which include development, release, and user testing

Our platform objectives of testing are to prove to the customer and the developer that the software satisfies their needs and is ready for release and to identify inputs or input sequences where the software behaves in an incorrect, undesirable, or non-specified manner.

Development testing

A crucial step in the software development process is development testing which makes sure the product is accurate, dependable, and of high quality. Unit testing, component testing, and system testing are some of the phases that development testing entails. Every step concentrates on a distinct component of the program, and when combined, they enhance the overall effectiveness of the testing procedure. (Taley & Pathak, 2020)

Therefore, it is crucial for our E-commerce platform for social business products and services to go through the development testing strategy as a part of the overall testing procedure. Going through development testing procedures will allow us to see the errors and how costumers and vendors experience the platform.

Unit testing

Unit testing is a process that concentrates on testing individual units separately to make sure they perform as intended. Testing every operation connected to an object, setting and examining every attribute of an object, and exercising the object in all conceivable states comprise comprehensive test coverage of an object class. (Sawant & Chawan, 2012)

Below will be appended with an overview of a possible methodology for unit testing on our e-commerce platform for social business products and services;

E-commerce Platform
Vendor/seller

register()
login()
addProduct()
updateProfile()

The table above contains the “E-commerce platform” as the object indicating the attribute “vendor/seller” and all the possible methods that can be done by the seller such as the “addProduct()” method that can be performed only after the “login()” method is executed first.

With testing that method we can use the partition unit testing strategy to figure the product details into sections for testing and verification, such as those that are available, those that are out of stock, and those that contain special features.

Component Testing

Component testing is the second development testing stage that evaluates the usability and behavior of each component of the E-commerce platform such as the user authentication and payment process. It requires each component to be independent, controllable, and user-friendly.

For instance; the main goal of testing the user verification component is to ensure proper and validated authorization for both the clients and sellers and that is done through verifying their logins and logouts through the user interface successfully by checking the user credentials stored in the user database system. If verification is successful access is allowed otherwise it is denied and error messages will be displayed such as incorrect credentials and ask them to try again.

System Testing

System testing assesses the functionality and performance of the fully integrated software solution. It checks if the e-commerce platform system meets the specified requirements and if it is suitable for delivery to end-users (Sawant & Chawan, 2012). It is done by our testing team after the process of integrating reusable parts that have been independently developed and ready-made systems with newly created parts.

Use case testing is an effective strategy used to test our e-commerce platform system through its various system components and their seamless interactions.

For example, the sequence diagram related to the use case shows the importance of testing if “displaying the products” and “adding items to the shopping cart” are interacting and performing seamlessly together.

Release testing

Release testing is the second strategy of software testing which verifies that a particular software product release satisfies the requirements needed. It helps us to guarantee that our software is ready to release to end users at the time of release. It is done to find and fix any flaws, problems, or defects that might not be noticed during development.(Kapur et al., 2016)

Our E-commerce platform will have to go through release testing to help us guarantee that platform can be released to public and to ensure all bugs are fixed completely that we might’ve missed.

Requirements Based Testing

A testing process based on requirements is considered a systematic method for designing test cases that take into account every requirement and generate a set of tests for it.

For instance, considering a successful checkout process and confirming compliance with the stated requirements in the E-commerce platform for social business products and services;

The checkout process must satisfy the following user requirements of our platform;

- User verification
- Browsing products
- Adding products to the cart
- Checkout completion
- Order confirmation
- Payment process
- Tracking product delivery status

When the user logs in and adds items to the shopping cart after searching and browsing products, the user proceeds through a simple process. The "Checkout" button opens a page with sections for order summary, shipping, billing, and making payments. Order processing begins when you click "Place Order" and provide accurate information. Then an email with a confirmation message will be sent as confirmation of the order and completed checkout thus the user will be able to track their order delivery. Otherwise, if any step is missed the process fails and no confirmation message is sent. This test guarantees a smooth checkout process that complies with the requirements.

Scenario Testing

We can use scenario testing as an approach to release testing by creating realistic potential scenarios that could occur while using the e-commerce platform to apply test cases on the system by going through a sequence of actions that symbolize user interactions with the system from beginning to end.

An example of a possible scenario from the e-commerce platform for social business products and services:

Nora is an environmentally friendly candle maker who registered and logged in for the social business e-commerce site. She was required to provide a valid email, her vendor's name, a brief description of her business, her warehouse or office address, and an 8-character password. Then for verification purposes, she was asked to upload a photocopy of her identification documents. After she finished all the registration requirements she received an email containing successful account registration. Then a product page appeared that has a separate field for each product, with 2 mandatory subfields one that holds pricing, and another one that holds product description and specification. Then she decided to upload her first product post by clicking the button "Add New Product" on her seller dashboard. She filled in the title, description, and category of her candles on the form that was displayed. Nora specified the available stock, set her reasonable pricing, and uploaded the photographs of her products to the desired platform specifications. Nora was able to emphasize the social impact of this platform, which is intended for social enterprises. She gave it a brief look and then hit "Submit." A message verified that her environmentally friendly candles were successfully posted and can be seen by customers to make purchases through the e-commerce platform for social businesses.

The scenario tests several features of the e-commerce platform;

- Verification by logging into the system
- Sellers uploading and posting their products
- Products details configuration
- Social effectiveness of seller vendors
- Product review and confirmation
- Sellers being notified for successful product posts

Performance Testing

Performance Testing makes sure there is no decline in the system's performance and ability to manage more than the suspected user loads, transactions, and features.

It involves stress testing which examines the stability of the system in challenging conditions by applying loads higher than what the system can handle. Then preserving updates on how the system handles high-stress circumstances to determine any potential problems or system failures after the system is tested. (Kapur et al., 2016)

User Testing

User testing is a stage in the testing process in which users or customers provide input and advice on our platform testing. That helps us observe the influences from the user's working environment that have a major effect on the reliability, performance, usability, and robustness of the e-commerce platform.

User testing involves various types including;

Alpha testing

Allowing our e-commerce platform users to assist our development team in checking the system at the developer site before proceeding with beta testing.

Beta testing

It is the last testing phase before a product is made available to the general public. A system version is released to some users to try things out and report any issues they find to the system developers.

Acceptance testing

Allowing customers to test our system to determine whether or not the system is working properly and check if it meets their expectations and can be accepted by the system developers and finally launched to the marketplace.

(Mohd & Shahbodin, 2015)

Conclusion

To finish up our essay we did, we hope the website will first and foremost help in smaller social business to boost up their sales and sellers and costumers both to be satisfied with the platform. Things presented in our report gave us the insight of the whole software process and how will everything work enough for both web developers and others involved into the project to understand each and every step. We hope the explanation of our project satisfied both parties in the way of simple way how would everything work from registering our account to actual payment and purchase to a feedbacks costumers can give.

References

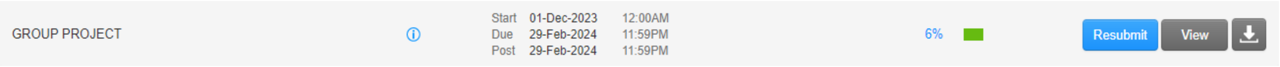
Taley, D. S., & Pathak, B. (2020). Comprehensive Study of Software Testing Techniques and Strategies: a review. *International Journal of Engineering Research and Technology*, *V9*(08). https://doi.org/10.17577/ijertv9is080373

Sawant, A. A., Bari, P. H., & Chawan, P. M. (2012). Software testing techniques and strategies. *International Journal of Engineering Research and Applications (IJERA)*, *2*(3), 980-986. https://www.academia.edu/download/28319593/FQ23980986.pdf

Kapur, P. K., Shrivastava, A. K., & Singh, O. (2016). When to release and stop testing of a software. *Journal of the Indian Society for Probability and Statistics*, *18*(1), 19–37. https://doi.org/10.1007/s41096-016-0012-6

Mohd, C. K. N. C. K., & Shahbodin, F. (2015). Personalized Learning Environment: Alpha Testing, Beta Testing & User Acceptance Test. *Procedia - Social and Behavioral Sciences*, *195*, 837–843. https://doi.org/10.1016/j.sbspro.2015.06.319

Turnitin Group Project:



GROUP PROJECT

Start	01-Dec-2023	12:00AM
Due	29-Feb-2024	11:59PM
Post	29-Feb-2024	11:59PM

6% 🟩