

Faculty of Computer Science and Engineering
Computer Science Department



[Shipment booking Software]

Introduction to Software Engineering Course
CS 281
Fall 2021

Contents

Abstract	4
Chapter One: Introduction	5
1.1 Overview	5
1.2 Problem Definition	5
1.3 Description of Proposed System	5
1.4 Process Model	5
Chapter Two: System Analysis	6
2.1 Domain Analysis	6
2.2 The Environment	7
2.3 Customers and Users	8
2.4 Existing Systems	8
2.5 Use Case Model	9
2.5.1 Actors of the system	9
2.5.2 Use Case Diagram	10
2.5.3 Use Case Descriptions	11
2.5.4 Functional Requirements	13
2.5.5 Non-Functional Requirements	13
Chapter Three: System Design	14
3.1 System Architecture	14
3.2 Sequence Diagrams	15
3.3 Activity Diagram	17
Chapter Four: Implementation (Optional)	19
4.1 User Interface Prototype	19
Chapter Five: Testing (Optional)	21
5.1 Initial Test Plan	21
5.2 Test Cases	21
Chapter Six: Conclusion	22
6.1 Summary	22
6.2 Lessons Learnt	22
6.3 Challenges and Limitations	23

6.4 Future Work	23
6.5 References	23

Abstract

The shipment booking software is a web-based tool for suppliers to manage their shipments. It enables suppliers to register with different warehouses and monitor their goods stored there. The software allows warehouse staff to accept shipments, check their condition, and create a gate pass for the cargo. It calculates weight and other shipment details and records them. The admin has full control over the system and can moderate the process. Customers can place orders, track their shipments, and choose to have their delivery picked up from any warehouse

Chapter One: Introduction

1.1 Overview

It is enterprise software project related to shipment sector. There are two targeted users, suppliers can check goods and warehouses and customers also check their shipments and track them to picking them up

1.2 Problem Definition

is shipment booking software designed for suppliers to manage their goods in a warehouse. It provides features such as registering to different warehouses, monitoring goods, checking the condition of goods, generating a gate pass, calculating the shipment weight, and making entries. The software also allows customers to place orders, track shipments, and request delivery from any warehouse. The admin has overall control over the system and can moderate the process.

1.3 Description of Proposed System

This system invented to facilitate communication between customers and supplies. One of the main goals of the system is safety and performance. We will achieve that by using less communication between components and backup systems. The deliverables methods are software prototypes until the stakeholders are completely approved and satisfied then launching a new website. The estimated cost of the project is 12k\$ negotiable.

1.4 Process Model

incremental development the system depends on customer requirement and it changes, involves breaking a project down into smaller, more manageable chunks or iterations, with each iteration delivering a discrete or self-contained piece of functionality. it becomes easier to manage risks, adjust to changing requirements, and get feedback from stakeholders. Additionally, the ability to deliver discrete pieces of functionality in each iteration can help to reduce the overall time to market and provide value to the end-users more quickly.

Chapter Two: System Analysis

2.1 Domain Analysis

Time domain analysis of control systems involves studying the response of a system to various inputs over time. This can include analyzing the transient response, steady-state response, and the stability of the system. Techniques used in time domain analysis include step response analysis, impulse response analysis, and frequency response analysis. These techniques can be used to determine the stability and performance of a control system, and to design control systems that meet specific performance requirements.

Domain analysis for shipment booking software web-based application requires an understanding of the shipping industry, the shipping processes, and the challenges that arise when booking shipments. The following are some key aspects of domain analysis for shipment booking software:

1. **Industry Overview:** It's important to have a basic understanding of the shipping industry, including the various modes of transportation (e.g. air, sea, land), the different types of cargo (e.g. containerized, bulk, refrigerated), and the regulatory requirements (e.g. customs clearance, hazardous materials handling).
2. **Shipment Booking Process:** Understanding the end-to-end shipment booking process is essential, which includes the steps such as identifying shipment requirements, obtaining rates and transit times, selecting the carrier, generating shipping documents, tracking the shipment, and receiving delivery confirmation.
3. **Customer Needs:** Identifying the needs of the customers and their expectations from the software. It is important to know the users and their technical proficiency.
4. **Features and Functionality:** Identifying the essential features and functionality for the software, such as shipment tracking, rate management, document generation, carrier selection, etc.
5. **Integration with External Systems:** Analyzing the external systems, which are part of the shipping industry, such as carriers, shippers, consignees, customs agencies, and other relevant parties. Understanding their technological capabilities and adapting the software to seamlessly integrate with their systems to ensure smooth communication and process.

6. **Security and Compliance:** Understanding the data security and privacy compliance measures that are required for handling sensitive data like shipment details and financial transactions.
7. **Reporting and Analytics:** The software should provide reports and analytics to monitor shipment history, status, transit times, and other relevant data points.
8. **User Interface:** The software should have an intuitive user interface with clear navigations and easy to use functionalities to ensure a smooth experience for the user.

By analyzing these aspects, the domain of shipment booking software can be understood thoroughly to build a software solution that best fits the business requirements.

2.2 The Environment

To develop a shipment booking software web-based application, you'll need to consider various factors related to the environment in which the application will be used. Here are some considerations:

1. **Web Hosting:** You will need a web hosting service that can support your application, including the necessary storage, bandwidth, and processing power to handle the expected traffic and user load.
2. **Programming Languages:** You'll need to choose the appropriate programming language(s) and web framework(s) for developing the application. Some popular choices include PHP, Python, Ruby on Rails, and Node.js.
3. **Database:** You'll need to choose a database to store the application's data. Some popular choices include MySQL, PostgreSQL, and MongoDB.
4. **Security:** Security is a crucial consideration for any web-based application, particularly one that involves sensitive information like shipment details. You'll need to implement security measures such as SSL/TLS encryption, secure login systems, and firewalls to prevent unauthorized access.
5. **User Interface:** The user interface of the application should be user-friendly and intuitive, with clear and concise instructions and easy navigation.
6. **Compatibility:** The application should be compatible with different web browsers, operating systems, and mobile devices.

7. **Integration:** You'll need to consider how the application will integrate with other systems and software that the users may be using.
8. **Performance:** The application should be fast and responsive, with minimal downtime and fast loading times.
9. **Scalability:** The application should be designed with scalability in mind, so it can handle increased traffic and user load as the business grows.
10. **Maintenance and Support:** You'll need to have a plan in place for maintaining and supporting the application, including regular updates, bug fixes, and user support.

2.3 Customers and Users

- Supplier
- Warehouse
- Customer
- Admin

2.4 Existing Systems

There are many existing systems for a shipment software web-based application, and the specific features and functionalities of such a system will vary depending on the needs of the users and the nature of the shipments being managed. However, some common features of a shipment software web-based application may include:

1. **Order Management:** This includes features to manage, process, and track shipments, including creating and managing orders, generating shipping labels, tracking shipments, and providing real-time visibility into shipment status.
2. **Carrier Integration:** This feature enables users to connect with multiple carriers and compare shipping rates, choose the best carrier for their shipment, and print shipping labels directly from the software.
3. **Inventory Management:** This feature allows users to manage their inventory, track stock levels, and receive notifications when inventory levels reach certain thresholds.
4. **Accounting and Invoicing:** This feature includes the ability to create and manage invoices, generate financial reports, and integrate with accounting software.
5. **Customer Support:** This feature includes tools for managing customer inquiries, resolving customer issues, and providing customers with real-time shipment updates.

6. **Analytics and Reporting:** This feature allows users to access and analyze data on their shipping activities, such as order volume, carrier performance, shipping costs, and delivery times, in order to optimize their operations.

Overall, the existing system for a shipment software web-based application will depend on the specific needs of the users but will likely include a combination of the features listed above, as well as other capabilities to support the efficient and effective management of shipments.



There are many existing web-based shipment software applications that are available. Here are some examples:

1. **ShipStation:** This web-based application offers shipping automation, label printing, and order tracking from multiple sales channels, including e-commerce platforms, marketplaces, and more.
2. **Freightos:** This software enables users to instantly compare and book international shipping services from a network of over 50 carriers.
3. **uShip:** uShip is a web-based shipping platform that connects shippers with carriers to get the best prices for shipping their goods.
4. **ShipBob:** This is a fulfillment platform that offers order management, inventory tracking, and shipping services for e-commerce businesses.
5. **Parcelhub:** This web-based software allows users to compare shipping rates from multiple carriers, create and print shipping labels, and track packages.
6. **ShipRocket:** ShipRocket is an e-commerce shipping solution that provides features such as order management, courier management, and shipping rate comparison.

These are just a few examples of the many web-based shipment software applications available. Each one has its own set of features and benefits, so it's important to research and compare them to find the best fit for your business.

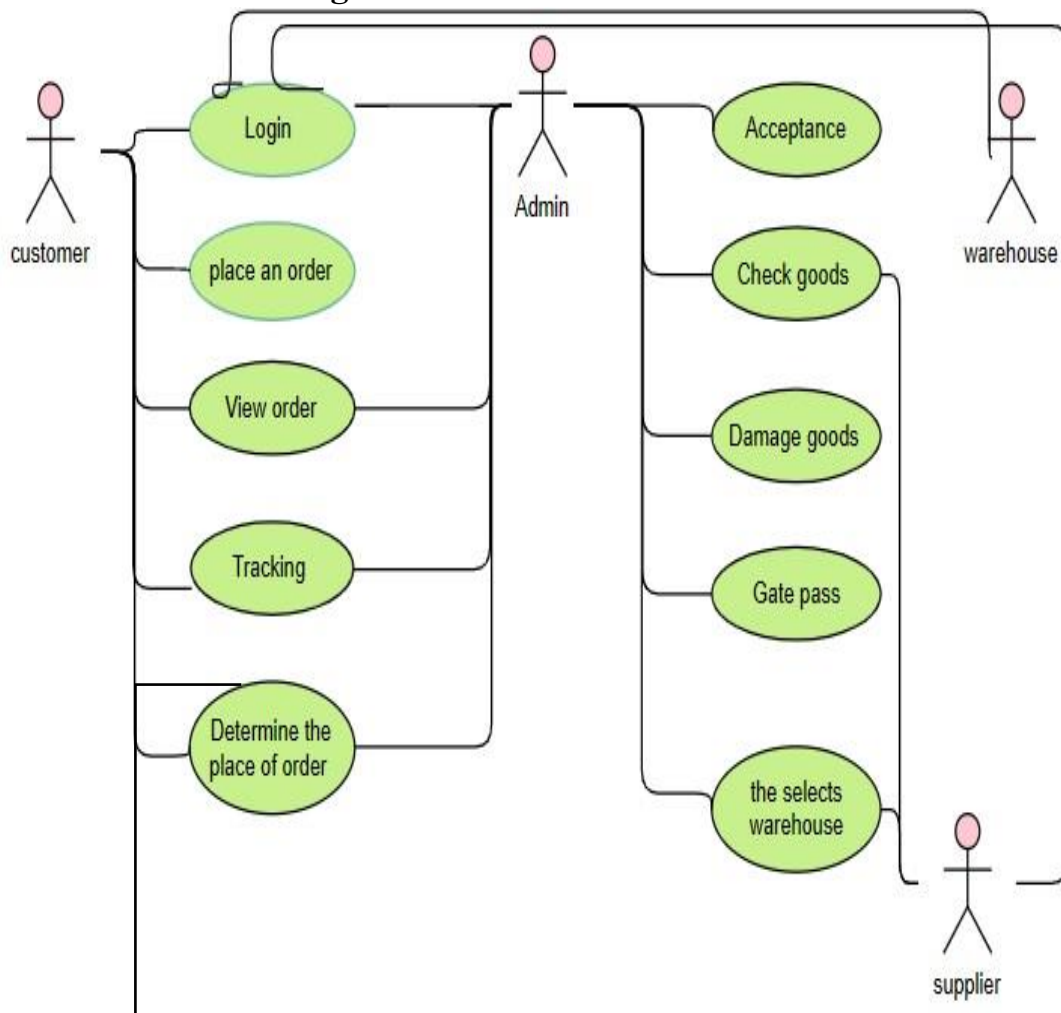
2.5 Use Case Model

2.5.1 Actors of the system

1.  **Supplier :**Use cases of supplier are ,check goods and select warehouse
2.  **Warehouse:** Use cases of Warehouse are goods acceptance ,check goods , and Returning goods when damaged and requesting them again

- Customer : 3. he can order the goods ,choose the place of delivery and track the order
4. Admin : Access to all processes

2.5.2 Use Case Diagram



2.5.3 Use Case Descriptions

USE CASE 1
Name – keep track of their goods(check goods)
Brief Description – suppliers need to make sure that their goods are save and in good state they can achieves that by this feature
Actors – Suppliers , and Admin are the only one can access the tracking goods
Preconditions – suppliers and Admin must to be register in the system and pass authentication process
Basic Flow – the suppliers or Admin should first login form their account by using correct user name and password then they can access tracking good form the list of services after the they must enter the goods ID number the system will show the state and location of goods
Alternate Flows – the suppliers or Admin after they pass authentication process the can track goods by enter ID number in search bar
Exception Flows –incorrect username, password, and ID number
Post Conditions – state and location of goods

USE CASE 2
Name – Determine place the delivery
Brief Description – The customer(user) has the option to pick up the order where he wants it or pick it up from the nearest warehouse to him
Actors – Admin ,and customer

Preconditions Customers and the administrator must register in the system and fill the basket with the goods they want and confirm them
Basic Flow – customers or the administrator must first log in from their account using the correct username and password, then put the goods they want in the basket and then check them and then click on the location icon to choose the place from which to receive the order or to receive it from the nearest warehouse
Exception Flows – incorrect username and incorrect password, The basket is empty
Post Conditions – Location successfully selected

USE CASE 3
Name –Damage or error in the product.
Brief Description – When the order is processed, and the item appears to be damaged, the storehouse replaces it with another and compensates the customer.
Actors – Admin and the storehouse.
Preconditions – Checks the item
Basic Flow – The supervisor checks the commodity through a barcode When there is an error or damage, the system sends the product back to the factory.
Alternate Flows – –The supervisor can track from the serial number
Post Conditions – Ensure the integrity of the commodity

2.5.4 Functional Requirements

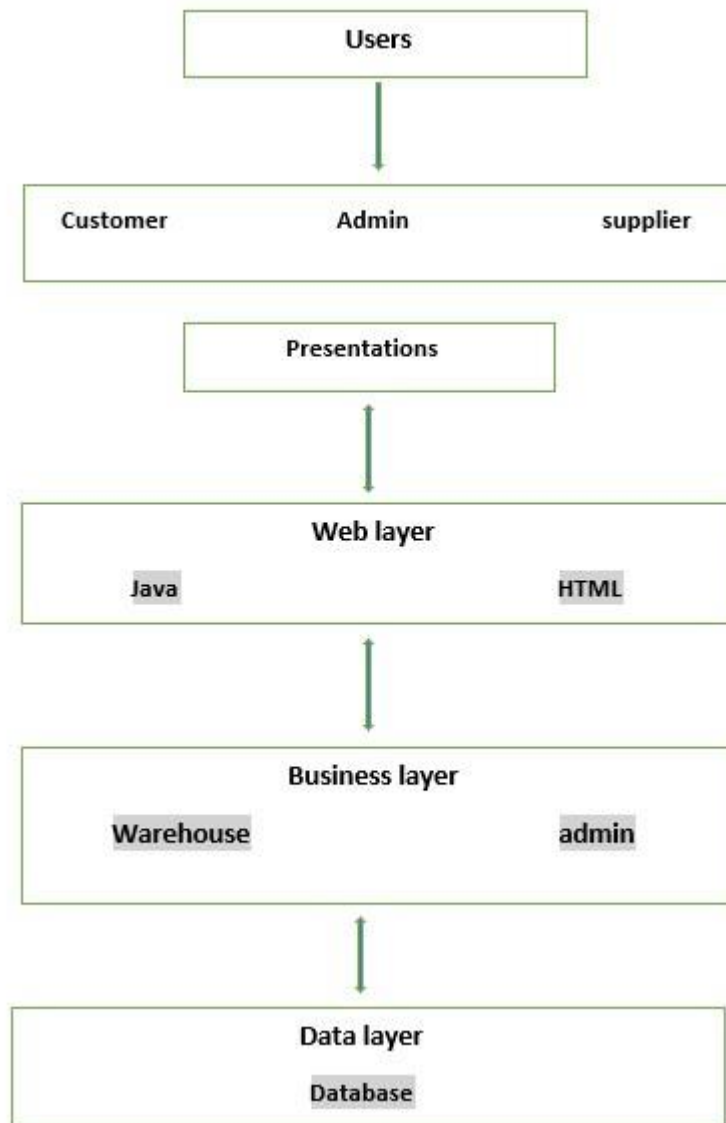
- The must be login process foe administrator and authorized users to avoid unauthorized access to the software
- The system shall allow to choose warehouse
- The system shall admin to update the current progress of the shipment
- The system shall verify the information entered by the customer during login
- The system shall allow Track goods (Goods is in which Warehouse)
- The system shall allow admin to display the current tracking information about the order
- The system shall allow user to enter order information for tracking .
- It shall ask the user to enter the goods details like shipment number or reference Number
The system shall allow view order
- the system shall to check the goods if damages the system shall allow reorder if the status is Damages
- the system shall the dispatch goods and create gate pass

2.5.5 Non-Functional Requirements

- **Accuracy:** All the processes provided by the system are accurate thus increase the reliability
- **Security** System is secure in itself by checking the illegal access of invalid users. System is also capable to input some security checks at certain points.
- **Portability:** The system should run in any environments
- **Mobility** - By this we can remove the problem of specific place and time. This system is able to survive in the network environment.
- **Multitasking** - By this character we can perform more than one task at a time.
- **Quick response** :- As the processing time of any task is minimized therefore the user gets the quick response of his/her queries even though there is a huge amount of data.
- **Robust Error handling:** The errors and undesirable situation, generated through the user intervention, are handled successfully to ensure that the system operates without halting

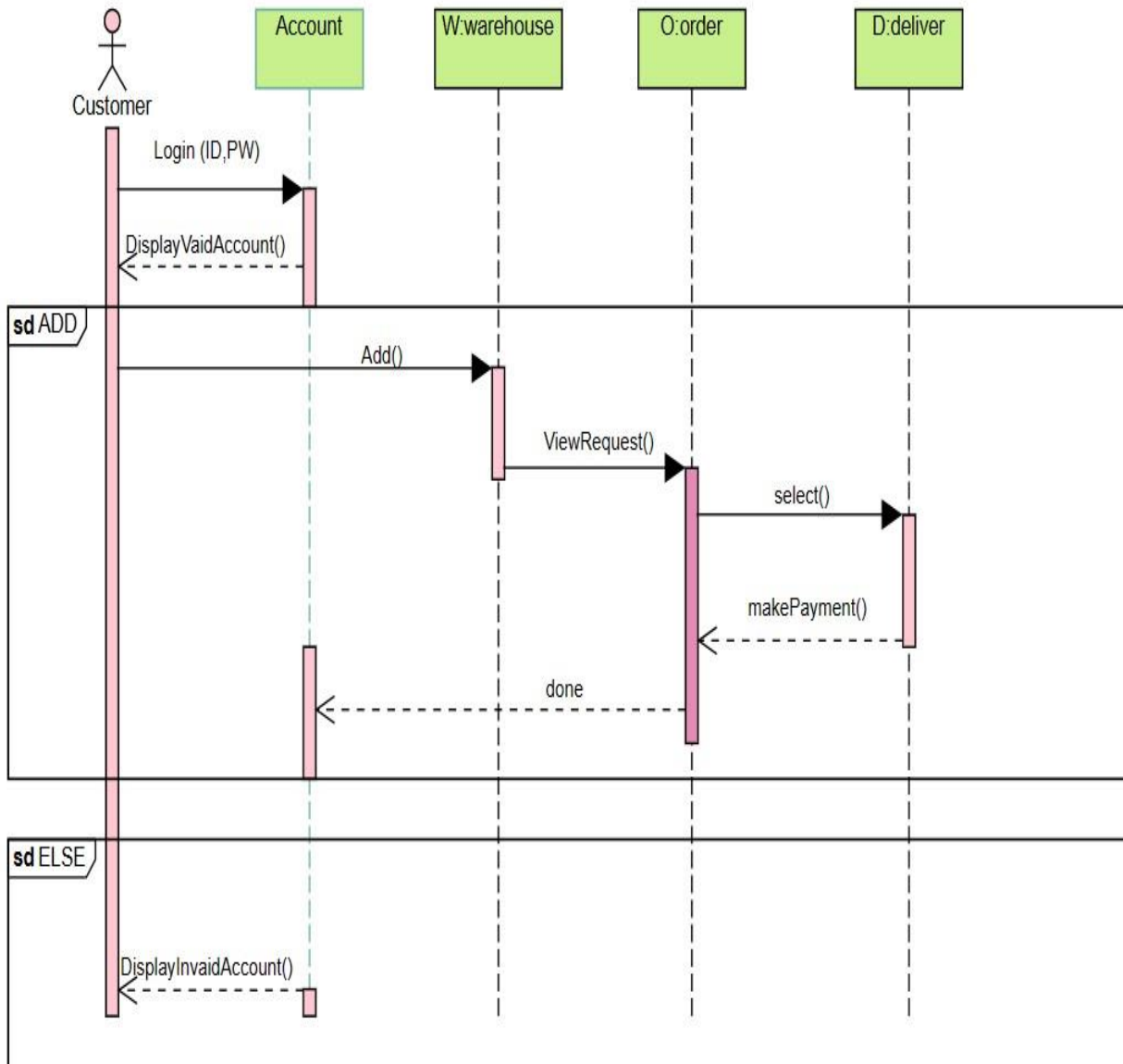
Chapter Three: System Design

3.1 System Architecture

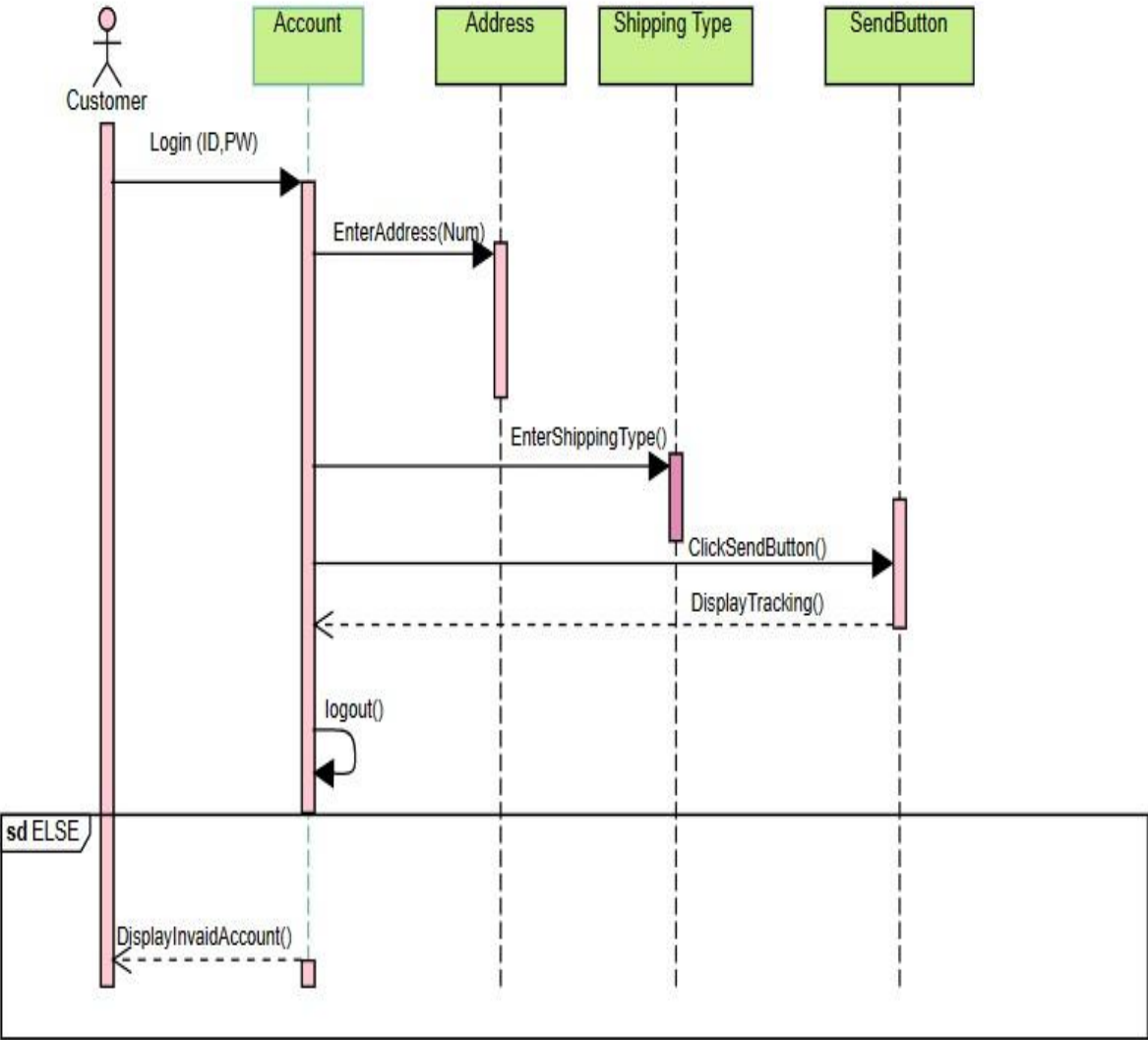


3.2 Sequence Diagrams

(make order)

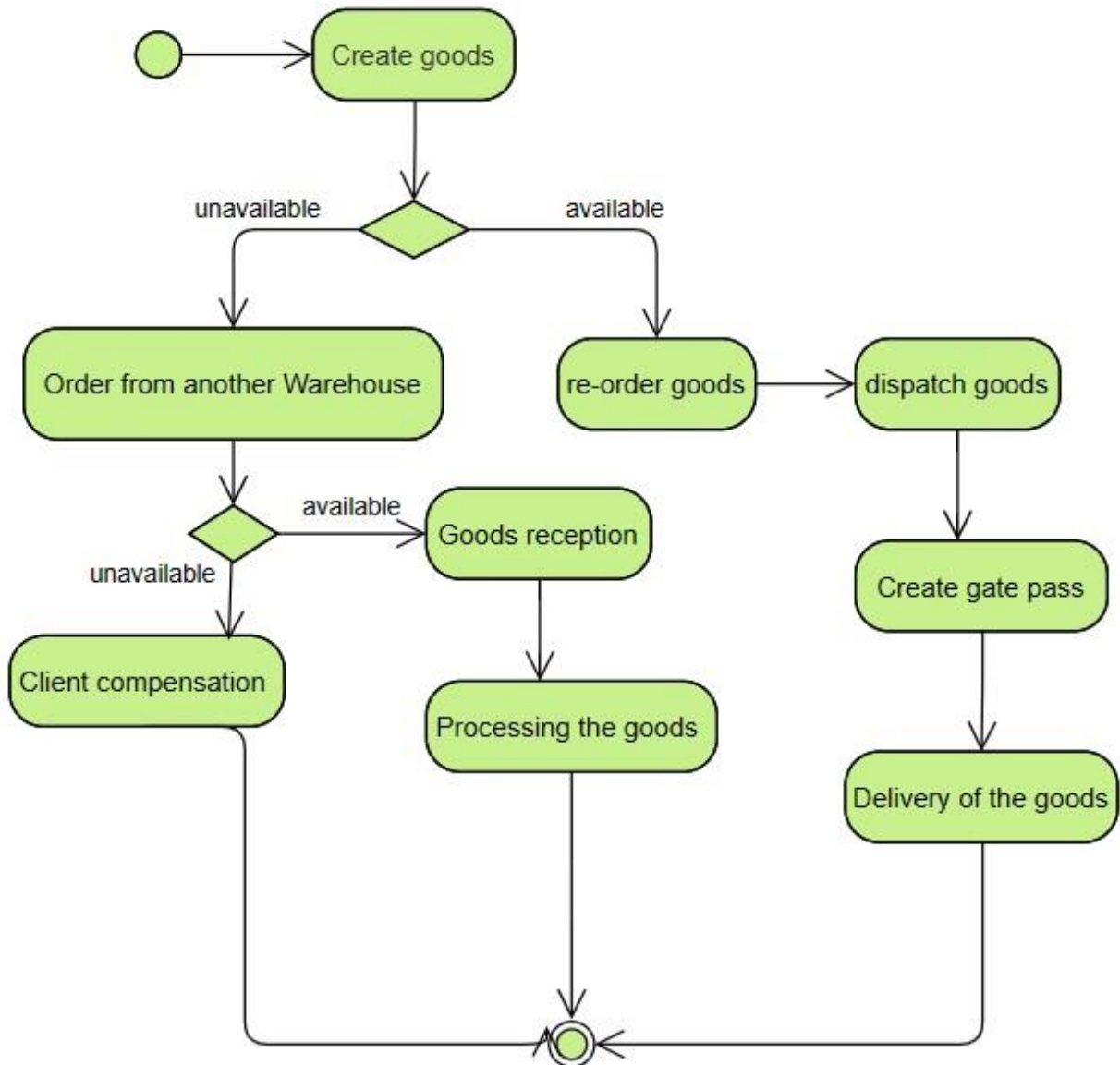


(order tracking)

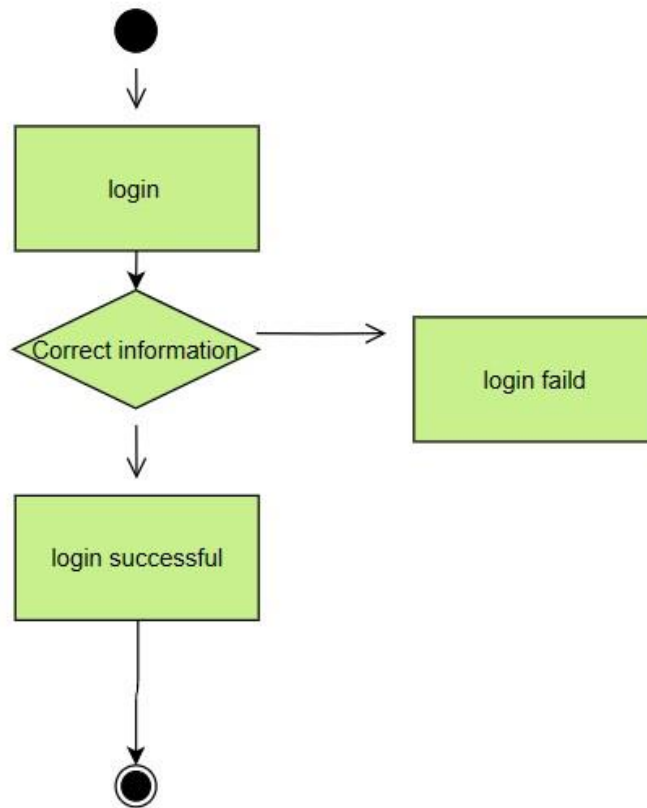


3.3 Activity Diagram

(When the goods are damaged)



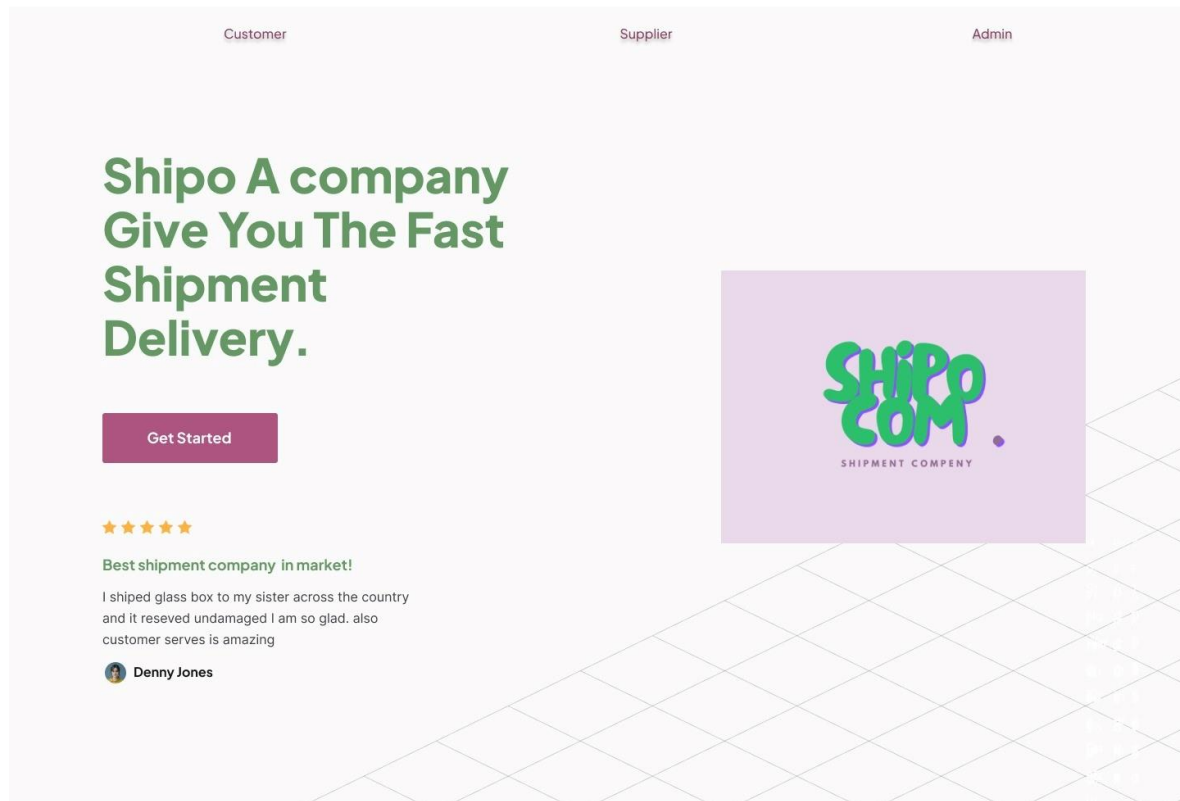
3. 4 State Diagram




Chapter Four: Implementation

(Optional)

4.1 User Interface Prototype





Start Your Journey

First name

Last name

Email

you@company.com

Phone number

SA (+966) 000 0000

☐ Customer

☐ Spplier

☐ Admin

LOGIN

Bills

Manage your billing and payment details

Overview

Segments

Dashboard

Search by invoice number, name, amount...

Filter

NUMBER	NAME	BILLING DATE	STATUS	AMOUNT	
<input type="checkbox"/> 5146846548465	Jane Cooper	2/19/21	Paid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 5467319467348	Wade Warren	5/7/16	Paid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 1345705945446	Esther Howard	9/18/16	Unpaid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 5440754979...	Cameron Williamson	2/11/12	Paid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 1243467984543	Brooklyn Simmons	9/18/16	Unpaid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 8454134649707	Leslie Alexander	1/28/17	Unpaid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 2130164040451	Jenny Wilson	5/27/15	Paid	\$500.00	<div>Pay</div>
<input type="checkbox"/> 0439104645404	Guy Hawkins	8/2/19	Paid	\$500.00	<div>Pay</div>

Chapter Five: Testing

(Optional)

5.1 Initial Test Plan

Project's team members must ensure that the quality is applicable on the project. Therefore, there should be criteria for measuring the quality of the project. Firstly, the most important quality measurement is delivering the project on time. Secondly, completing the project within the scope, plan, and all its objectives. Thirdly, designing the final system in such a way to be user friendly, and easy to be used by novice and expert people with no errors. Finally, using the resources which are needed in a proper way to accomplish the project's aims and objectives.

5.2 Test Cases

The test cases which must be applied on the project are as follows.

- All fields are validated and do not accept invalid values, such as Goods, Gate Pass, and Entries.
- Forms must not be left blank.
- Preventing data mistakes.
- Managing the amount of input.
- The system's modules/forms are all integrated.
- Setting up the test scenarios.
- Preparation of potential test data, including necessary validation checks.
- The actual testing is carried out by hand.
- Keeping track of all errors that have been repeated.
- Modifications are made in response to errors discovered during testing.
- After the errors have been corrected, preparing the test result scripts.

- The module's/forms' overall functionality.
- User input validations.
- Ensuring that the coding standards are followed during the coding process.
- Using all possible test data to test the module.
- Functionality testing, including various types of computations, etc.
- Standard commenting in source files.

Chapter Six: Conclusion

6.1 Summary

his website will benefit the public by fulfilling their desires in any possible way. The customer will benefit greatly from this service because it saves convenience, time and effort. The application will be very useful in the coming years, as we are working on many initiatives and partnerships that will help us provide a more sustainable service.

6.2 Lessons Learnt

- **Communicate Clearly:** Check in often with the team and keep communication lines open. Be clear with your expectations.
- **Give Praise Often:** Let your team know when they have done something well.
- **Ask for Advice:** Ask for the input you need from leaders and teammates.
- **Check In Often:** Communicate effectively with all levels of project involvement and update your manager(s) at regular intervals.
- **Improve Your Work:** Take feedback and use it to improve. This will help you learn from your missteps and grow your successes.

6.3 Challenges and Limitations

- Team skills are very important regarding to web development.
- Human resource management was one of the biggest challenges because some countries prevent entry or exit or prohibit or restrict goods according to customs regulations and procedures.
- Web cannot be operate without internet.
- Coding skills must be up to the marks.

6.4 Future Work

My future plan is to turn this site into an app and have it do multiple partnerships and take 10% of the profits and donate it to charity.

6.5 References

- https://tympanus.net/codrops/css_reference/
- Hricko, Mary (Editor). Design and Implementation of Web-Enabled Teaching Tools, Idea Group Publishing, 2002
- Kalbag, Laura. Accessibility for Everyone, Morgan Kaufmann, 2017
- Barrell, Dylan. Agile Accessibility Explained: A practical guide to sustainable accessible software
- Blanck, Peter. eQuality: The Struggle for Web Accessibility by Persons with Cognitive Disabilities, Cambridge Disability Law and Policy Series, 2015.
- Chisholm, and May. Universal Design for Web Applications: Web Applications That Reach Everyone, O'Reilly Media, 2008.
- Duckett, Jon. Accessible XHTML and CSS Web Sites Problem Design Solution, Wrox, 2005.
- Gay, Greg et al. Digital Accessibility as a Business Practice, Ryerson University Pressbooks, 2017.