Prepard By

Fatoumata, Ceesay
Habiba Karim, Rinky
Linsong, Li
Project Team

B-BAY ECOMMERCE SYSTEM

DESIGN DOCUMENTATION

Delivery Date

January 3rd, 2023

PREPARED FOR:

Sung, Shin

Owner of
Sung's & Co Limited
550 Main Avenue
Brookings SD, 57007
605-456-7890
Sung.Shin@Outlook.com

October 24, 2022

Table of Contents

Introduction

Purpose of documentation

This document is important as it entails all the design specifications which covers all aspects of our product design. By documenting it we shown stakeholders, cooperators and design team details that they can agree on.

Major Problems and Project Goals

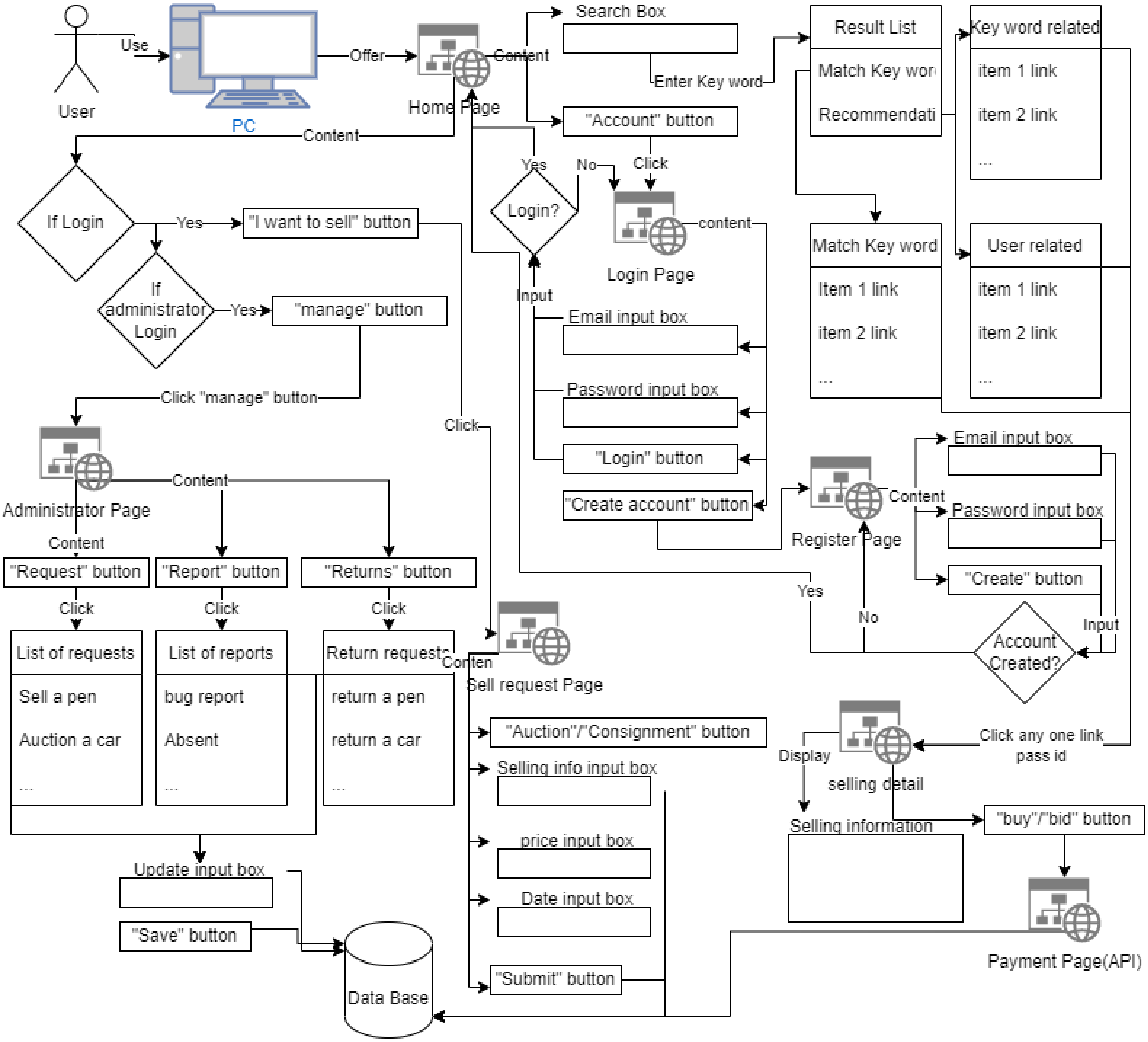
The goal of this project is to build an intelligent marketplace for buyers and sellers in Brookings. This task, however, comes with its challenges which are going to be discussed in this section.

Inadequate documentation on existing processes

There is very poor documentation available about the existing process. In this situation, requirements gathering became a two-step process. Firstly, we back engineered the existing process, and then identified areas for improvement and optimization.

We also drew business process maps and visualized workflows for the client to better understand.

Proposed system overview



Intelligent Recommendation System

While user input key words in search box from Home page, and press Enter, our system will offer a list of links that point to each merchandise’s detail page. This list is generated by results of four different searches. Direct key word search, related words search, what other people searched, and user information based search.

Descriptions and methods to get key word of each search

The direct key word search is the simplest one, it will find the top 10 matches from data base by using the whole key word, and then split the key word into words, do search of each, and find top 3 for each word.

User information based search is similar to direct key word search, but the key words this search used come from user profile.

What other people searched is a search that tell user what’s the top 10 popular key words used by people who did the same search as user. Each user’s search history is stored at data base related to their id. When a user did a search, the What other people searched search will select a table of all searched items from search history of all users who searched the same thing. And count the numbers of appear of each merchandise. The top 10 popular merchandises will be in the result of this search.

Related words search is a search that will find related words and do search for each related tags that’s been found. We will use CNN techniques to train a module that do text classification to each merchandise’s name, classification will relate the name with a tag exist or create a tag, then relate the name with the tag. While user input key word, we will use the same module to classify which tag is related, and use the merchandises in the tag as the result.

Text classification Module

Tools: tensorflow, jupytor notebook, python

Training and testing data collection: category names with merchandises' names, obtain by using web crawler to Best Buy, Amazon and so on… using 80% of data as training data and 20% of data as testing data

Model: by using tensorflow, write a model with more than 6 layers, using ReLU as default activation, set default learning rate to 0.02 for the first 30% of training and 0.005 for the rest of the training.

Hardware/Software

Related to hardware, our program will run on computer systems, so the user is expected to have a computer with dual-core,2.66-GHZ or faster processor and minimum of 2 GB of RAM. Moreover, the graphic card support in Windows for Direct X 9.0c is also expected.

APIs allow for the creation of a minimal interface that is relatively stable that can be used by other software systems to access or manipulate the underlying systems or data. This allows for enhancements to the underlying systems or data without disturbing the software systems that use the API Usually implemented using REST, SOAP, or JSON. Third party application and database integration is simplified as long as all parties support the published API.

Design Priority

Alternatives

B Bay e-commerce system is an online based website. Our initial plan is to run the B Bay e-commerce system with the five main functionalities including intelligent system what we have gotten from the client through frequent meeting and interviewing with them. But if for some reason, we cannot run the B Bay system, then we have plan B for our system.

Plan B is that we will set a meeting with the client within two weeks to submit new request and ask for new requirements. After adjusting the requirements, we will get back to the client again for their review. Then we will run the B Bay e-commerce system with the basic functionalities. There will be no complex functionalities so that I can give a user friendly interface. If our plan B fails, we have plan C as well.

Then we will recommend our client to NibrasInc company. First we will take an appointment on behalf of our client stating our problem and this company is also a software development company. After getting the appointment we will tell them the requirements of our client. And then we will set another appointment for our client. Then they will tell them their functional and non-functional requirements elaborately. So that they can help our client to get their desired B Bay e-commerce system.

Design Priority table

Item number	Item	Weight	Design1		Design2	
DP1	Cost	2	3	6	8	16
DP2	Speed of Execution	10	6	60	4	40

DP3	User FWBSiendliness	9	7	63	2	18
DP4	Complexity of the Codes	4	2	8	9	36
DP5	Customer Satisfaction	7	9	63	4	28
DP6	Level of Integration	5	4	20	2	10
DP7	Security Management	9	9	81	10	90
DP8	Easily Understandable	8	6	48	8	64
				343		302

Development and Execution Environment

Processor	Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz 2.40 GHz
Installed RAM	8.00 GB
Device ID	85D98DFF-81C7-4307-818E-75AD2CCA2833
Product ID	00325-95891-09475-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	Touch support with 10 touch points
Edition	Windows 10 Home
Version	21H2
Installed on	4/22/2021
OS build	19044.2130
Experience	Windows Feature Experience Pack 120.2212.4180.0

Programming tools

A programming tool or software development tool is a computer program that software developers use to create, debug, maintain, or otherwise support other programs and applications. The term usually refers to relatively simple programs, that can be combined to accomplish a task, much as one might use multiple hands to fix a physical object. The most basic tools are a source code editor and a compiler or interpreter, which are used ubiquitously and continuously. Other tools are used more or less depending on the language, development methodology, and individual engineer, often used for a discrete task, like a debugger or profiler. Tools may be discrete programs, executed separately – often from the command line – or may be parts of a single large program, called an integrated development environment (IDE). In many cases, particularly for simpler use, simple ad hoc techniques are used instead of a tool, such as print debugging instead of using a debugger, manual timing (of overall program or section of code) instead of a profiler, or tracking bugs in a text file or spreadsheet instead of a bug tracking system.

The distinction between tools and applications is murky. For example, developers use simple databases (such as a file containing a list of important values) all the time as tools. However, a full-blown database is usually thought of as an application or software in its own right. For many years, computer-assisted software engineering (CASE) tools were sought after. Successful tools have proven elusive. In one sense, CASE tools emphasized design and architecture support, such as for UML. But the most successful of these tools are IDEs.

Naming and Coding Standards

A consistent naming convention for files and for directories shall be developed and used on a perproject basis. A file naming convention makes project files easily distinguishable FWBSom other projects, and it helps associate different file types within the same project. Directories and subtrees can be used to link portions of a project together.

Each project shall adopt a set of coding standards consisting of three parts:

General Coding Standard, described in this document Language specific coding standards for each language used, described in separate appendices to this document. These language standards shall supplement, rather than override, the General Coding standards as much as possible. Project Coding Standards. These standards shall be based on the coding standards in this document and on the coding standards for the given language(s). The project coding standards should supplement, rather than override, the General Coding standards and the language coding standards. Where conflicts between documents exist, the project standard shall be considered correct. Sweeping per-project customizations of the standards are discouraged, so that code can be reused FWBSom one project to another with minimal change.

Error Handling

Functions that can fail (i.e. file I/O) should always return a success or error as a return code parameter. Any time a subroutine calls a function that returns an error condition, the error condition should be tested for and acted on in accordance with the error handling conventions specified in the projects SDDD. Error recovery should be handled in the routine that is responsible for the domain in which the error occurs (e.g. A file error should not be passed up from file_IO() to Main for handling).

Fault Tolerance

Fault tolerance refers to the ability of a system (computer, network, cloud cluster, etc.) to continue operating without interruption when one or more of its components fail.

The objective of creating a fault-tolerant system is to prevent disruptions arising from a single point of failure, ensuring the high availability and business continuity of mission-critical applications or systems.

Fault-tolerant systems use backup components that automatically take the place of failed components, ensuring no loss of service. These include:

- **Hardware systems** that are backed up by identical or equivalent systems. For example, a server can be made fault tolerant by using an identical server running in parallel, with all operations mirrored to the backup server.
- **Software systems** that are backed up by other software instances. For example, a database with customer information can be continuously replicated to another machine. If the primary database goes down, operations can be automatically redirected to the second database.
- **Power sources** that are made fault tolerant using alternative sources. For example, many organizations have power generators that can take over in case main line electricity fails.

In similar fashion, any system or component which is a single point of failure can be made fault tolerant using redundancy.

Fault tolerance can play a role in a disaster recovery strategy. For example, fault-tolerant systems with backup components in the cloud can restore mission-critical systems quickly, even if a natural or human-induced disaster destroys on-premise IT infrastructure.

Design Constraints

Design constraints are limitations that force a more methodical analysis of people and their problems. Because resources aren't inexhaustible and criteria must be met, they push designers to be strategic about the processes they use and energies they expend. Design constraints are limits placed on design. An obvious example is budget. Another is screen size. A less obvious example is a designer's skill level. Skill constraints are especially pressing when a single designer is responsible for deliverables in multiple disciplines (UI, UX, illustration, brand, etc.). There are many types of design constraints. For instance, commercial constraints include considerations like time, budget, and manpower. Stylistic constraints, often shared in brand guides, limit designers' aesthetic choices. In some cases, compliance constraints ensure designs align with laws and regulations.

Hardware and software constraints

Hardware constrains:

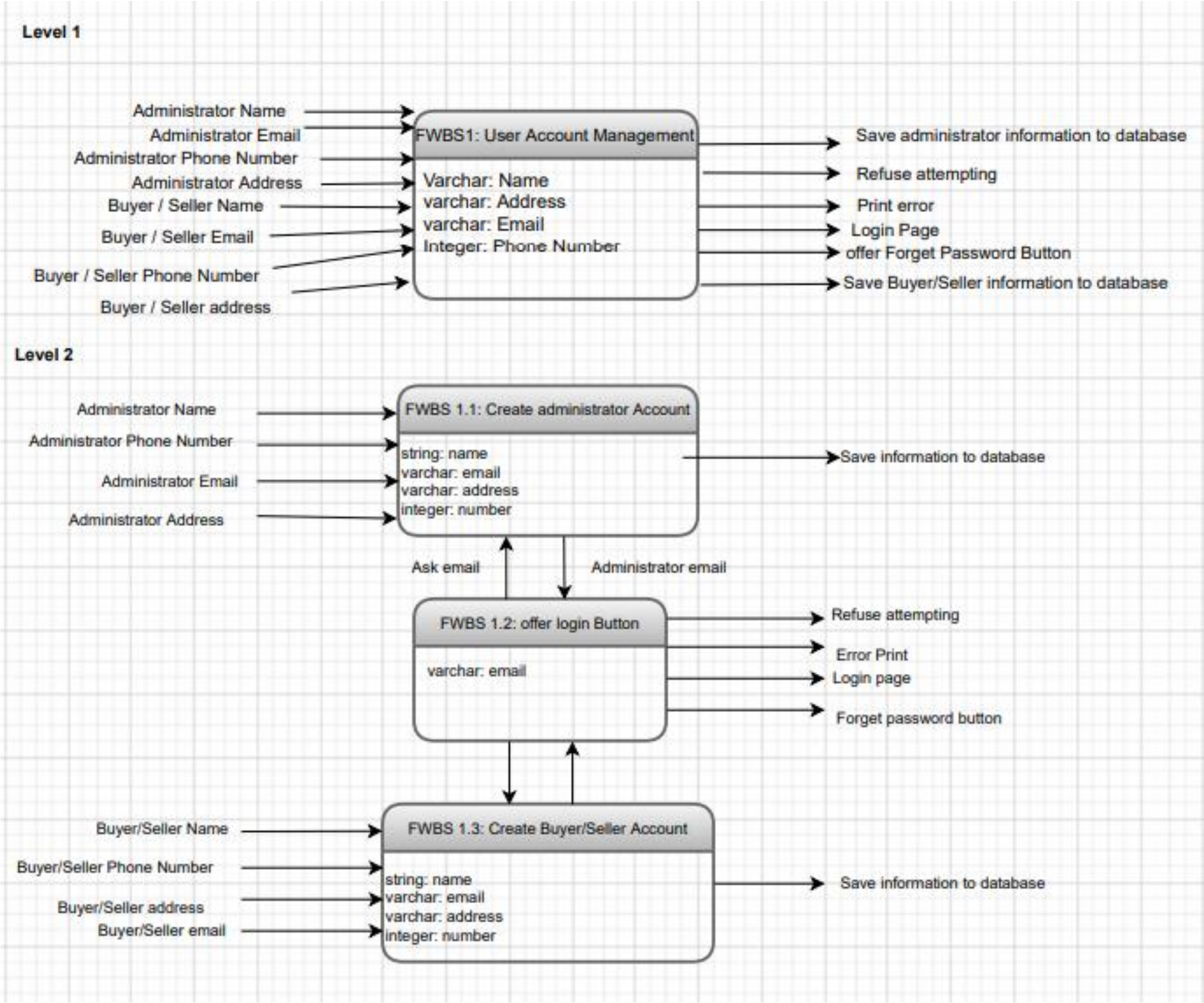
- outdated hardware specifications
- incompatibility between hardware and systems software
- proprietary hardware
- hardware restrictions
- accessibility (e.g. screen size)
- interoperability between hardware components

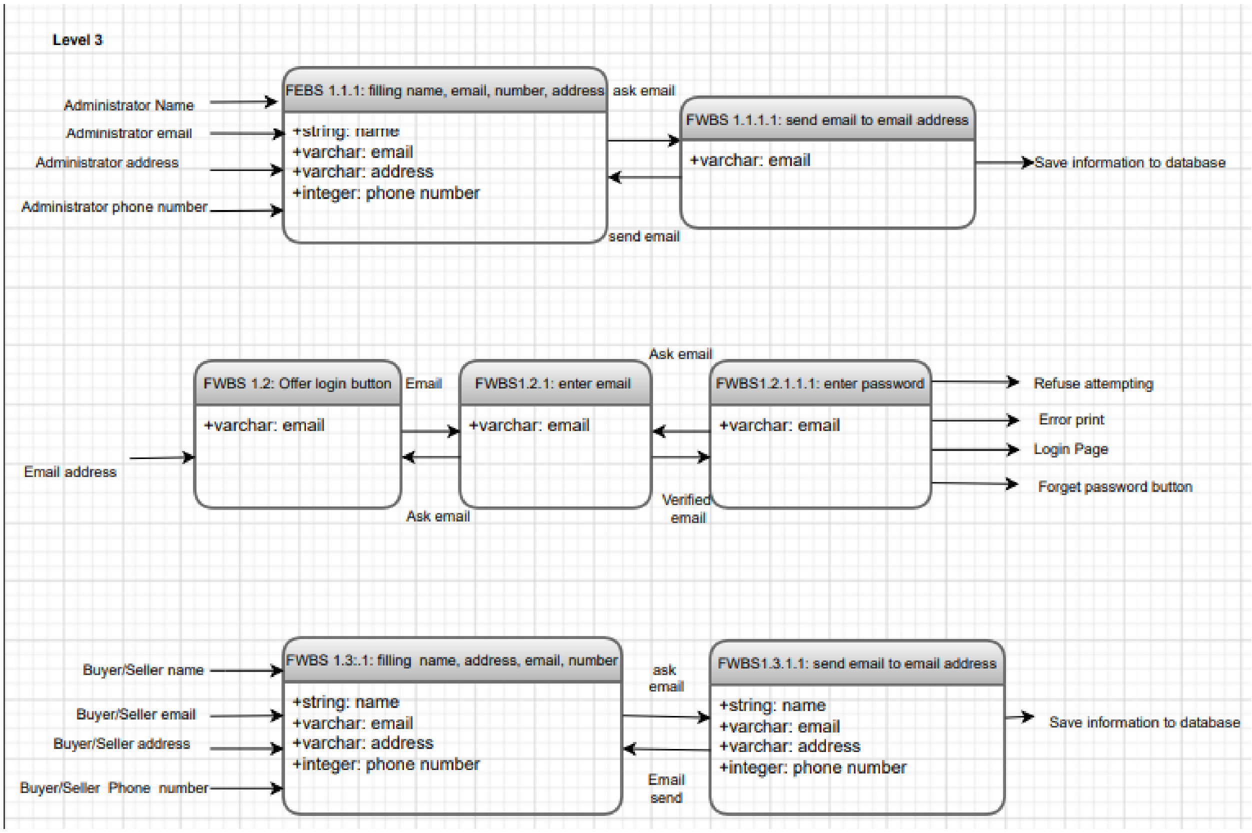
- lack of technical skills
- cost (e.g. financial, time)

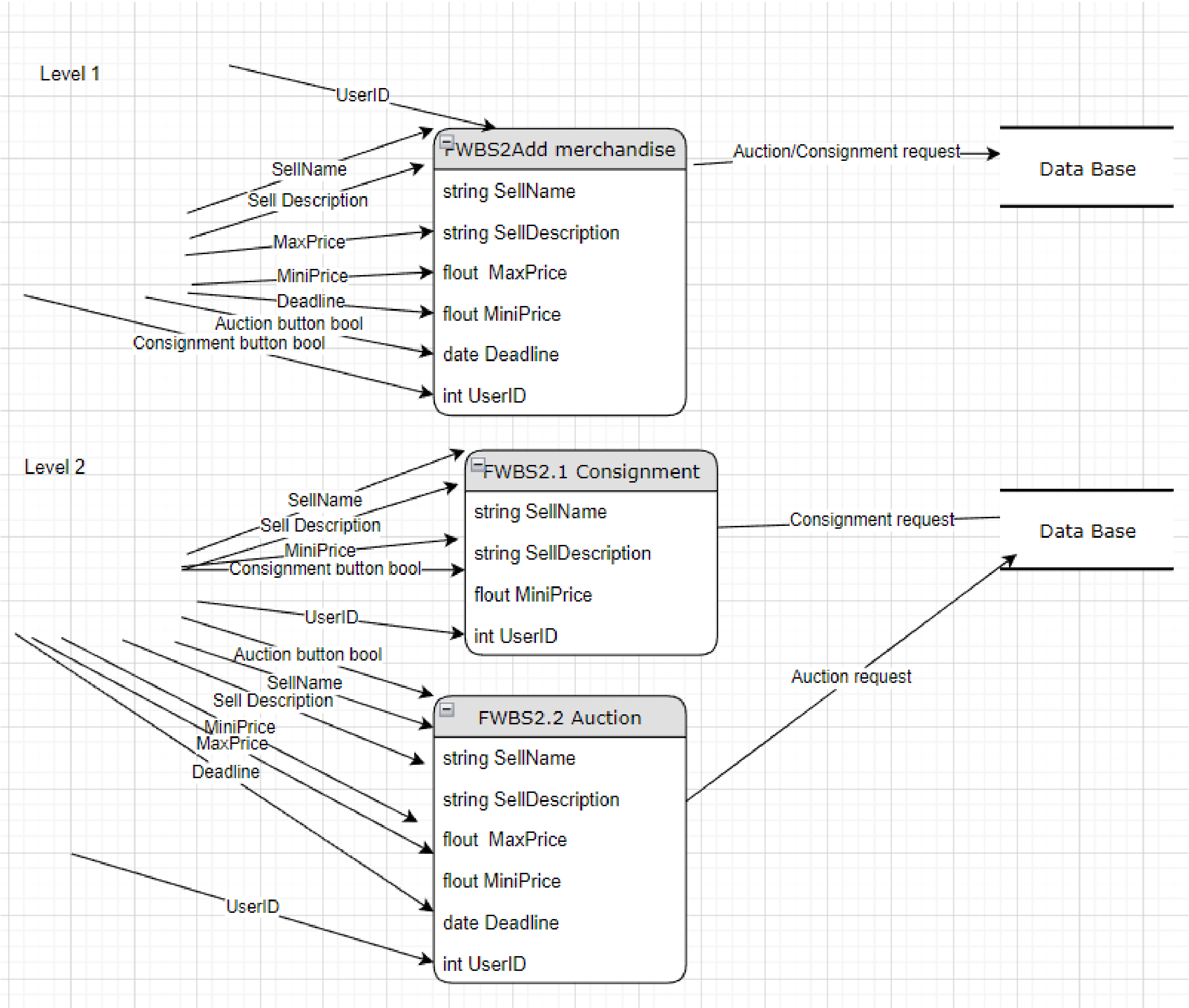
Software constraints:

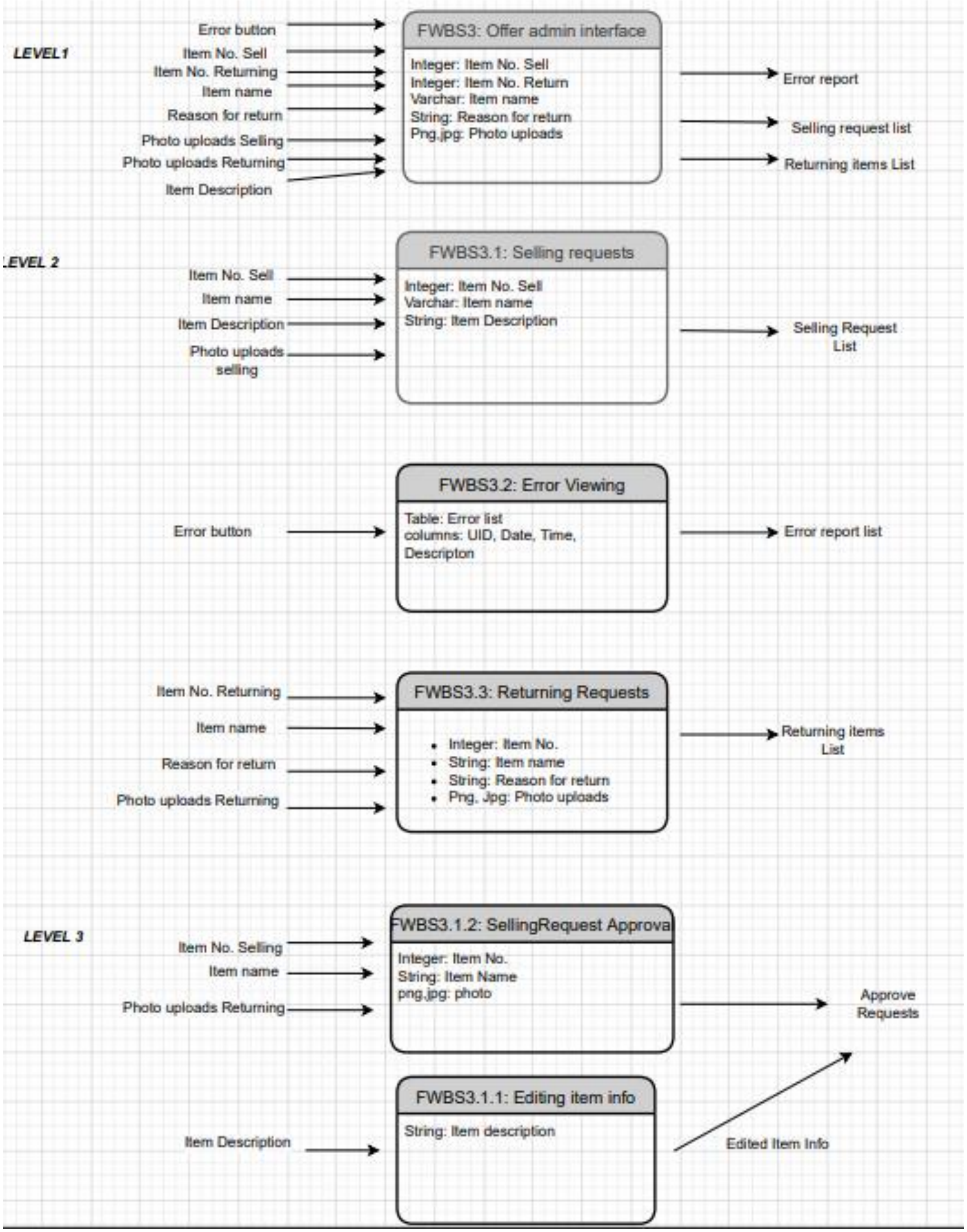
- outdated systems software versions
- incompatibility between software (e.g. operating system and programming language)
- limited memory
- limited storage
- registry issues
- issues with device drivers
- accessibility (e.g. screen resolution)
- lack of technical skills
- cost (e.g. financial, time)

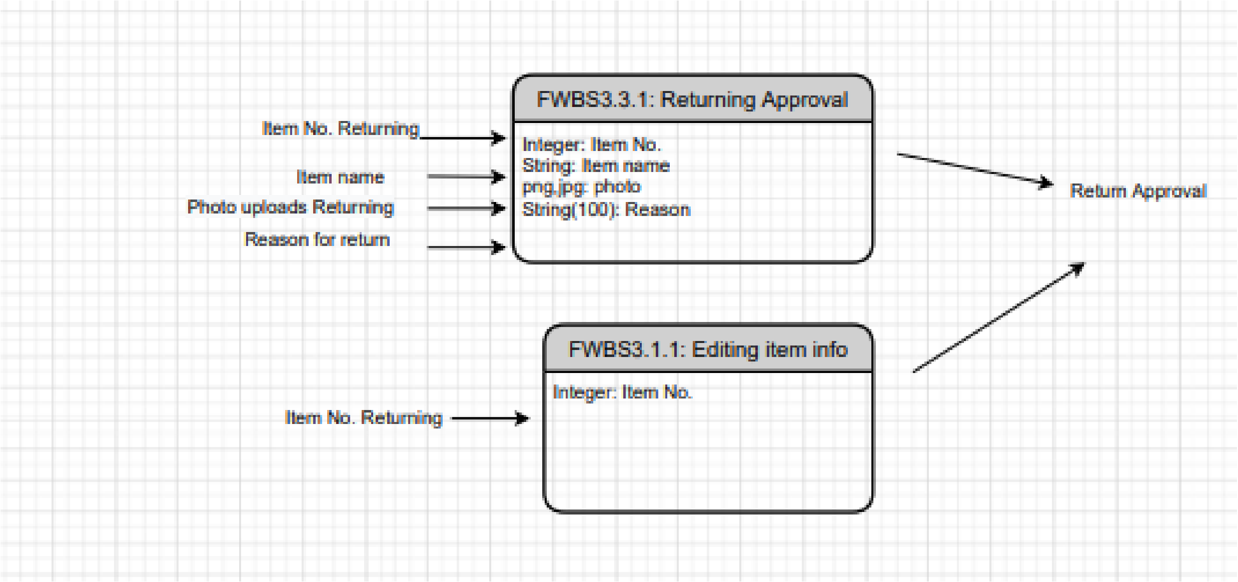
Data Flow Diagram



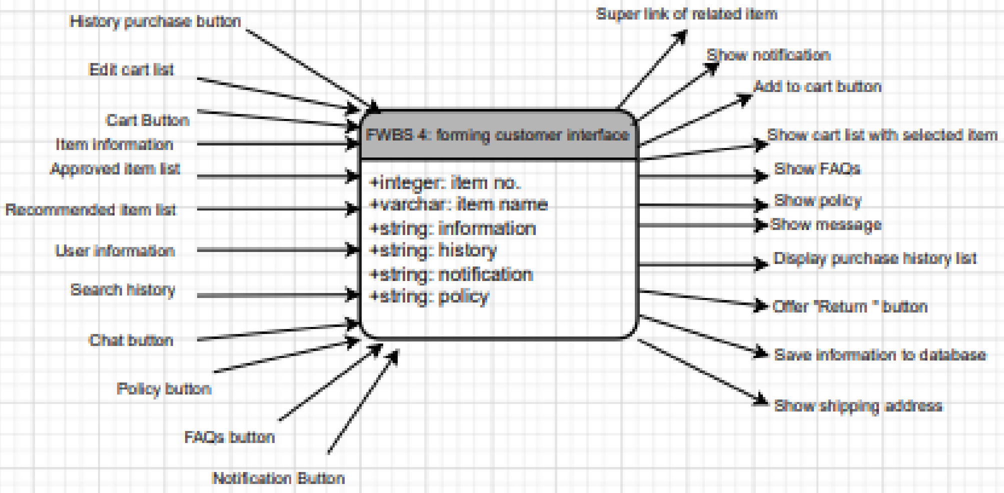

Level 3
Administrator Name
Administrator email
Administrator address
Administrator phone number
FEBS 1.1.1: filling name, email, number, address
+string: name
+varchar: email
+varchar: address
+integer: phone number
ask email
send email
FWBS 1.1.1.1: send email to email address
+varchar: email
Save information to database
Email address
FWBS 1.2: Offer login button
+varchar: email
Email
Ask email
FWBS1.2.1: enter email
+varchar: email
Ask email
Verified email
FWBS1.2.1.1.1: enter password
+varchar: email
Refuse attempting
Error print
Login Page
Forget password button
Buyer/Seller name
Buyer/Seller email
Buyer/Seller address
Buyer/Seller Phone number
FWBS 1.3:.1: filling name, address, email, number
+string: name
+varchar: email
+varchar: address
+integer: phone number
ask email
Email send
FWBS1.3.1.1: send email to email address
+string: name
+varchar: email
+varchar: address
+integer: phone number
Save information to database


Level 1
UserID
SellName
Sell Description
MaxPrice
MiniPrice
Deadline
Auction button bool
Consignment button bool
FWBS2Add merchandise
string SellName
string SellDescription
flout MaxPrice
flout MiniPrice
date Deadline
int UserID
Auction/Consignment request
Data Base
Level 2
SellName
Sell Description
MiniPrice
Consignment button bool
UserID
FWBS2.1 Consignment
string SellName
string SellDescription
flout MiniPrice
int UserID
Consignment request
Data Base
Auction button bool
SellName
Sell Description
MiniPrice
MaxPrice
Deadline
UserID
FWBS2.2 Auction
string SellName
string SellDescription
flout MaxPrice
flout MiniPrice
date Deadline
int UserID
Auction request

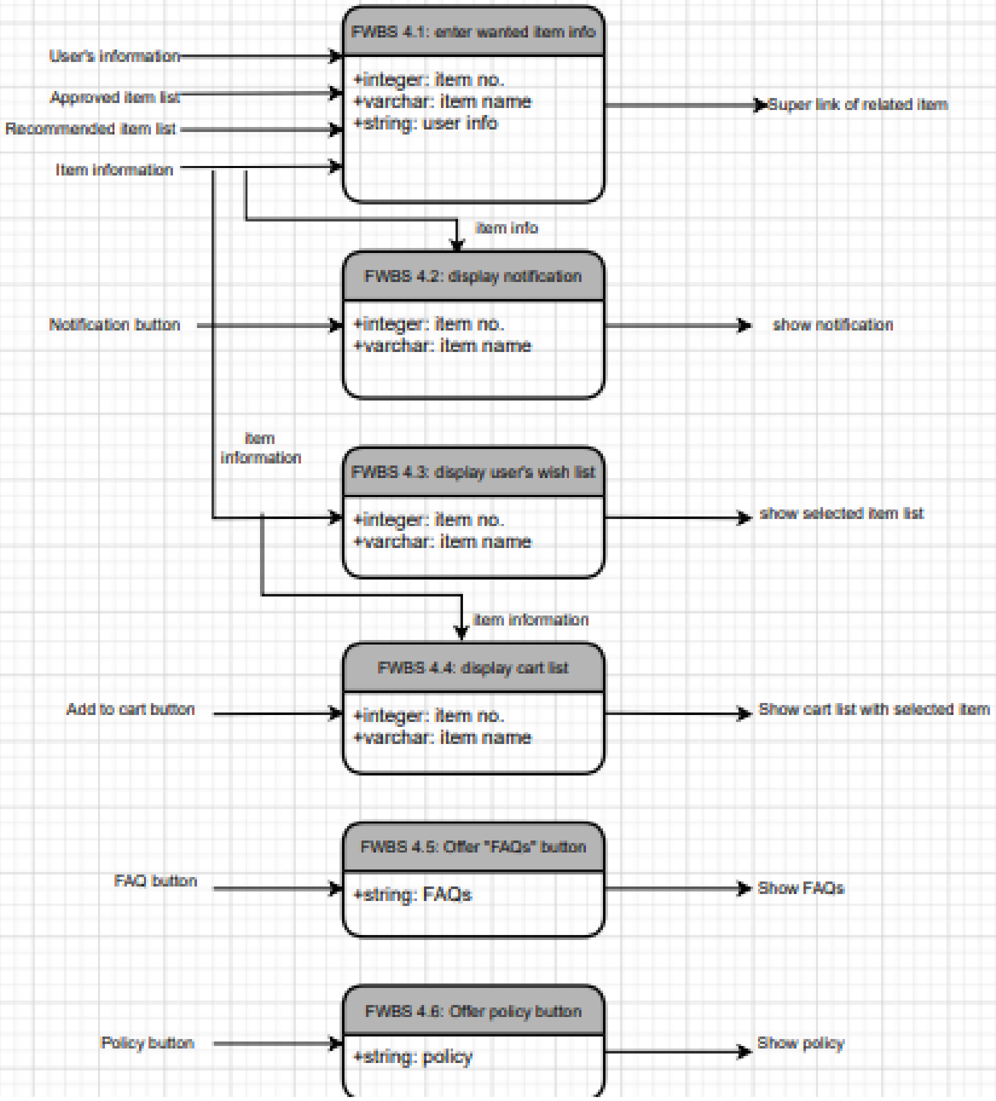

LEVEL1
Error button
Item No. Sell
Item No. Returning
Item name
Reason for return
Photo uploads Selling
Photo uploads Returning
Item Description
FWBS3: Offer admin interface
Integer: Item No. Sell
Integer: Item No. Return
Varchar: Item name
String: Reason for return
Png,jpg: Photo uploads
Error report
Selling request list
Returning items List
LEVEL 2
Item No. Sell
Item name
Item Description
Photo uploads selling
FWBS3.1: Selling requests
Integer: Item No. Sell
Varchar: Item name
String: Item Description
Selling Request List
Error button
FWBS3.2: Error Viewing
Table: Error list
columns: UID, Date, Time, Descripton
Error report list
Item No. Returning
Item name
Reason for return
Photo uploads Returning
FWBS3.3: Returning Requests
Integer: Item No.
String: Item name
String: Reason for return
Png, Jpg: Photo uploads
Returning items List
LEVEL 3
Item No. Selling
Item name
Photo uploads Returning
FWBS3.1.2: SellingRequest Approva
Integer: Item No.
String: Item Name
png,jpg: photo
Approve Requests
FWBS3.1.1: Editing item info
String: Item description
Item Description
Edited Item Info

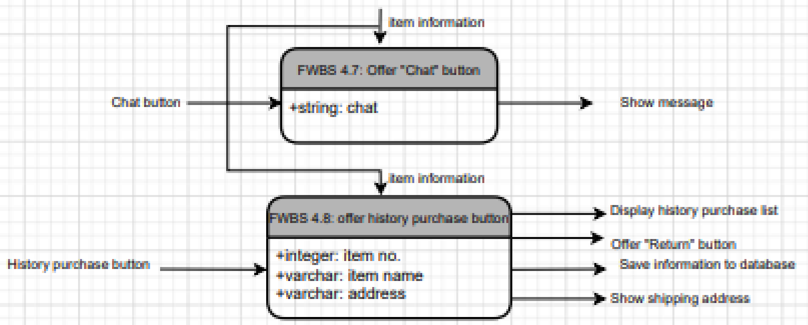

FWBS3.3.1: Returning Approval
Integer: Item No.
String: Item name
png.jpg: photo
String(100): Reason
Item No. Returning
Item name
Photo uploads Returning
Reason for return
Return Approval
FWBS3.1.1: Editing item info
Integer: Item No.
Item No. Returning

Level 1:

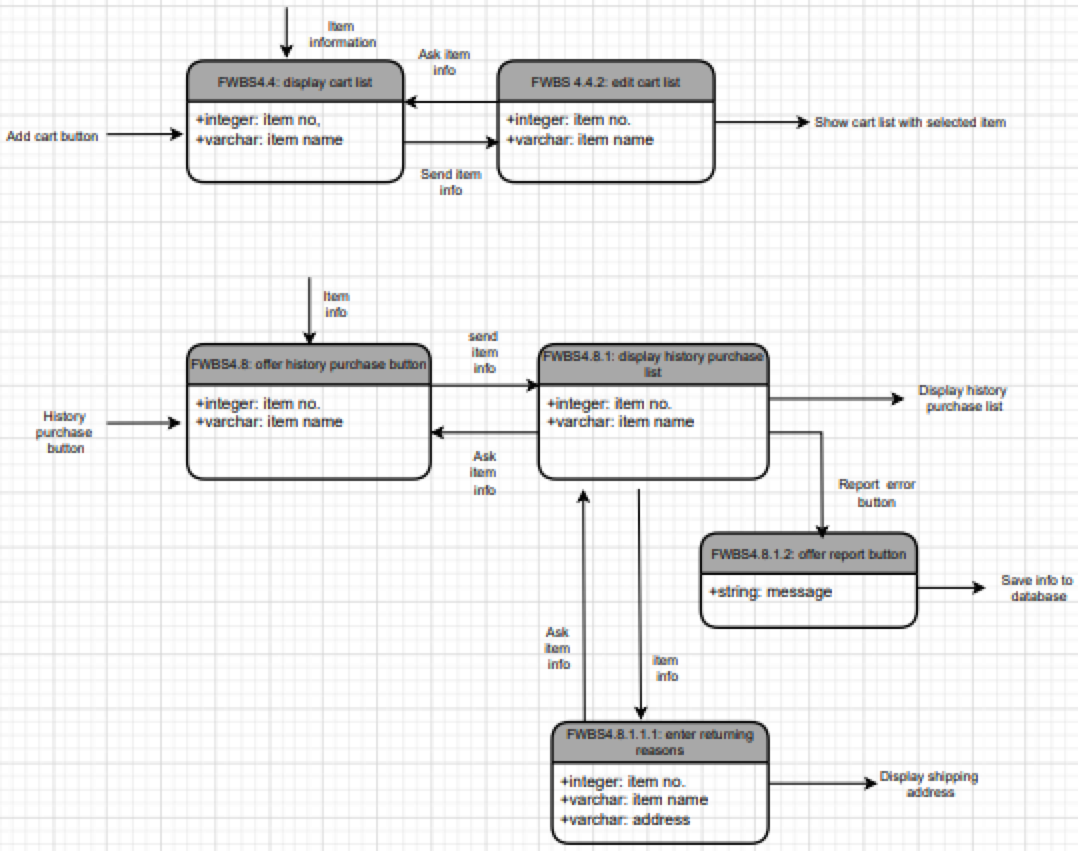


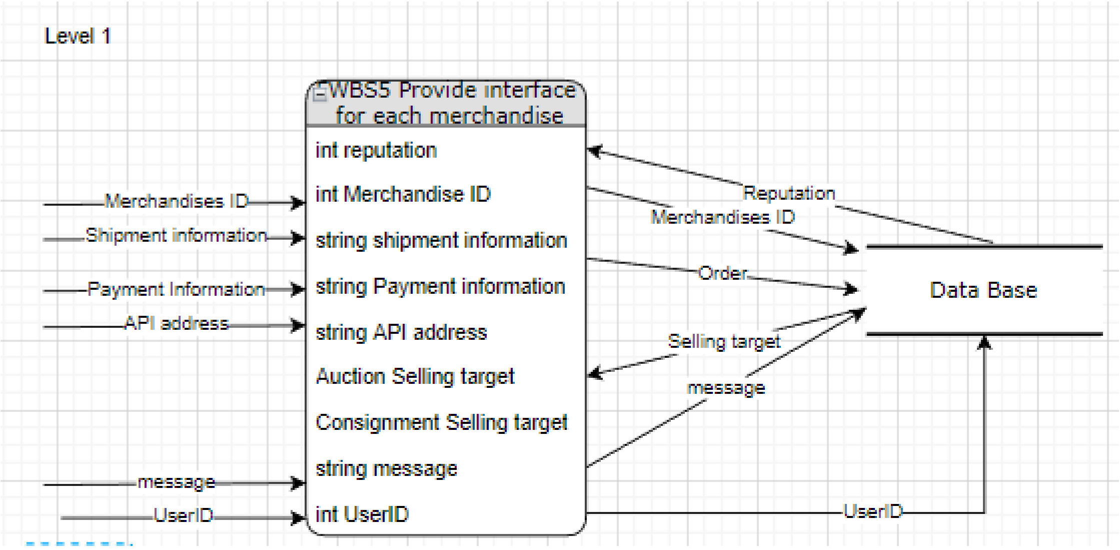
Level 2:

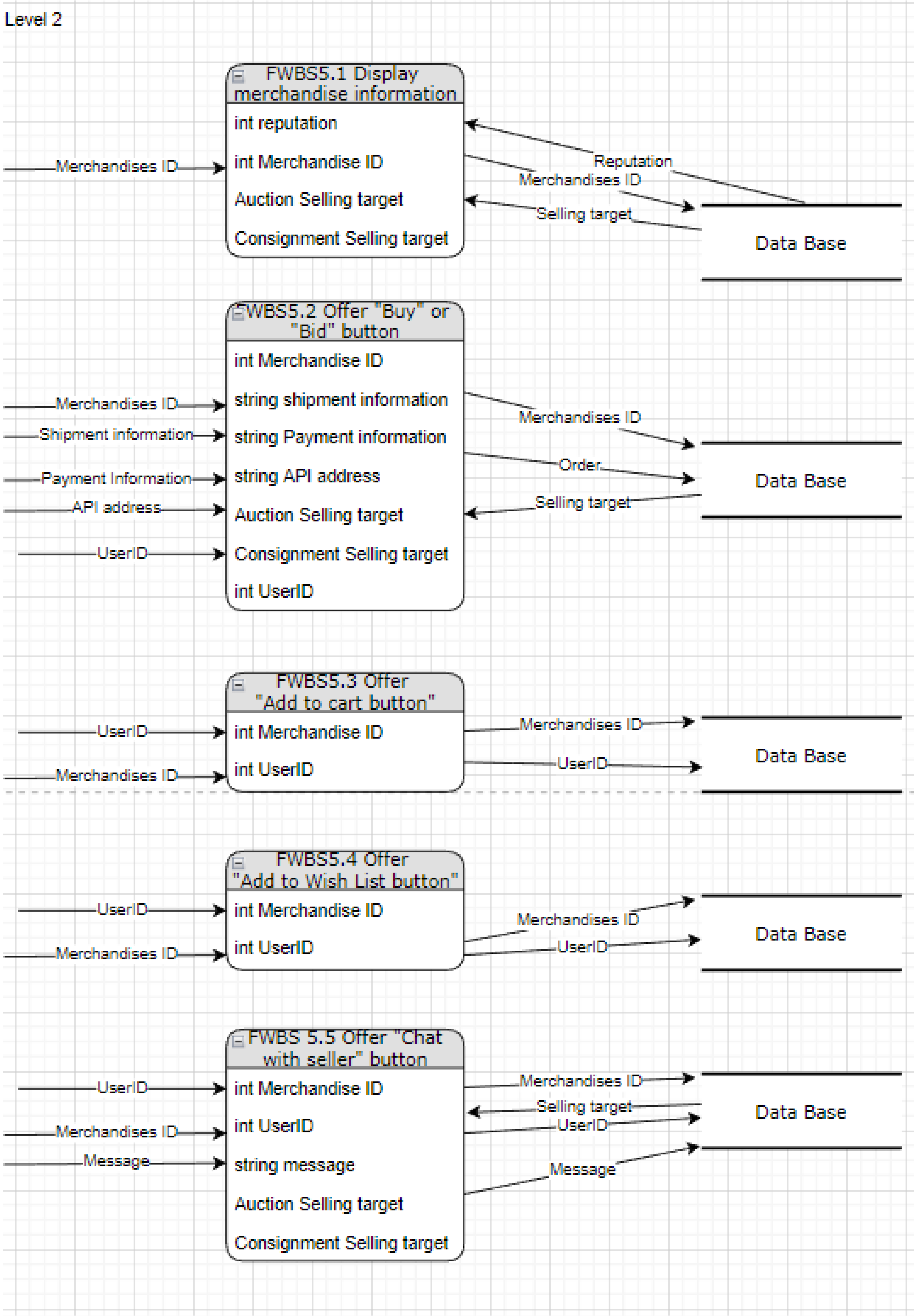



item information
FWBS 4.7: Offer "Chat" button
+string: chat
Chat button
Show message
item information
FWBS 4.8: offer history purchase button
+integer: item no.
+varchar: item name
+varchar: address
History purchase button
Display history purchase list
Offer "Return" button
Save information to database
Show shipping address

Level 3


Item information
Ask item info
FWBS4.4: display cart list
+integer: item no,
+varchar: item name
FWBS 4.4.2: edit cart list
+integer: item no.
+varchar: item name
Add cart button
Show cart list with selected item
Send item info
Item info
send item info
FWBS4.8: offer history purchase button
+integer: item no.
+varchar: item name
FWBS4.8.1: display history purchase list
+integer: item no.
+varchar: item name
History purchase button
Display history purchase list
Ask item info
Report error button
FWBS4.8.1.2: offer report button
+string: message
Save info to database
Ask item info
item info
FWBS4.8.1.1.1: enter returning reasons
+integer: item no.
+varchar: item name
+varchar: address
Display shipping address


Level 1
WBS5 Provide interface for each merchandise
int reputation
int Merchandise ID
string shipment information
string Payment information
string API address
Auction Selling target
Consignment Selling target
string message
int UserID
Merchandises ID
Shipment information
Payment Information
API address
message
UserID
Reputation
Merchandises ID
Order
Selling target
message
UserID
Data Base


Level 2
FWBS5.1 Display merchandise information
int reputation
int Merchandise ID
Auction Selling target
Consignment Selling target
Merchandises ID
Reputation
Merchandises ID
Selling target
Data Base
FWBS5.2 Offer "Buy" or "Bid" button
int Merchandise ID
string shipment information
string Payment information
string API address
Auction Selling target
Consignment Selling target
int UserID
Merchandises ID
Shipment information
Payment Information
API address
UserID
Merchandises ID
Order
Selling target
Data Base
FWBS5.3 Offer "Add to cart button"
int Merchandise ID
int UserID
UserID
Merchandises ID
Merchandises ID
UserID
Data Base
FWBS5.4 Offer "Add to Wish List button"
int Merchandise ID
int UserID
UserID
Merchandises ID
Merchandises ID
UserID
Data Base
FWBS 5.5 Offer "Chat with seller" button
int Merchandise ID
int UserID
string message
Auction Selling target
Consignment Selling target
UserID
Merchandises ID
Message
Merchandises ID
Selling target
UserID
Message
Data Base

Data Dictionary

Related FWBS	Name	Type	Range	Description
1;2;3;4;5	UserID	int	Natural No	Unique number that represent each user
1, 1.1, 1.3, 1.1.1, 1.3.1, 1.3.1.1	Name	string	Within 100 chars	Name of the Administrator, Buyer/Seller
1, 1.1, 1.3, 1.1.1, 1.3.1, 1.3.1.1, 4.8, 4.8.1.1.1	Address	varchar	Within 500 chars	Address of administrator, buyer/seller
1, 1.1, 1.2, 1.3, 1.1.1, 1.1.1.1, 1.2.1, 1.2.1.1.1, 1.3.1, 1.3.1.1	Email	varchar	Within 50 chars	Email address of administrator, buyer/Seller
1, 1.1, 1.3, 1.1.1, 1.3.1, 1.3.1.1	Phone number	Integer	10 integer number	Phone number of administrator, buyer/seller

2	SellName	String	within 100 chars	Name of proposed selling item.
2	Sell Description	String	within 1000 chars	Description of proposed selling item, including as much specifications
2	MiniPrice	int	1-99999	minimum Price requested to sell item.
2.2	MaxPrice	int	1-99999	maximum Price requested to sell item.
2.2	Deadline	Date	>2022/10/22/13:00	The ending time of proposed auction.
3.3	Returns	String in table	5 columns	Displays items to be returned, buyer info and reason
4, 4.1, 4.2, 4.3, 4.4, 4.8, 4.4.2, 4.8.1, 4.8.1.1.1	Item No.	Integer	5 integer number	Unique item number for each item
4, 4.1, 4.2, 4.3, 4.4, 4.8, 4.4.2,	Item name	varchar	With 50 chars	Name of the item

4.8.1, 4.8.1.1.1				
4, 4.1	Information	string	Within 1000 chars	User’s information
4	History	string	Within 1000 chars	History purchase list of user
4	Notification	string	Within 1000 chars	Notification about item, message to user
4, 4.6	Policy	string	Within 10000 chars	Show the policy of B-Bay company
4.5	FAQs	string	Within 10000 chars	The frequently asked questions about B-Bay
4.7, 4.8.1.2	chat	string	Within 5000 chars	Buyer/Seller message with each
5.2	Shipment information	String	within 100 chars	Information of shipment address
5.2	Payment information	String	Within 100 chars	Information of payment

Design Legend

I/O

There are 3 types of users, different option will be provided depending on the user:

Guests: 1,2,3.4

Administrator: 1,2,3.4,4,4.1,4.2,4.3

User: 1,2,3.1,3.2,3.3,3.4,5

1. Home Page
   Home Page is available to registered users, admin and guest users.

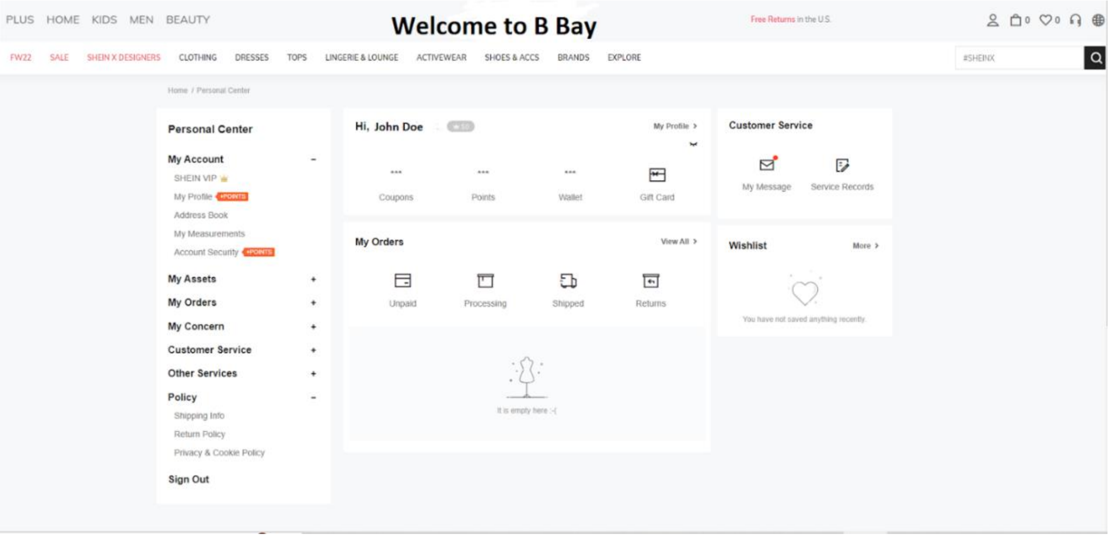
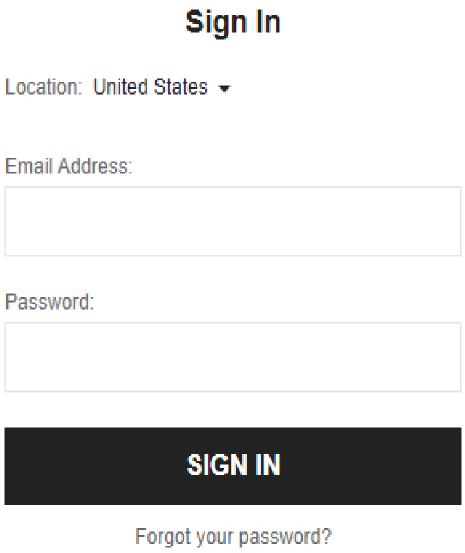


Fig1. Home Page

It contains the basic interface that the guest can use to navigate the website, hence this can be good for guests. Guests however only view but cannot perform any other actions like buying and selling.

2. Login

Upon arriving at the homepage (Fig1), the user will be asked to Login or Create a new account to be able to perform all functionalities.

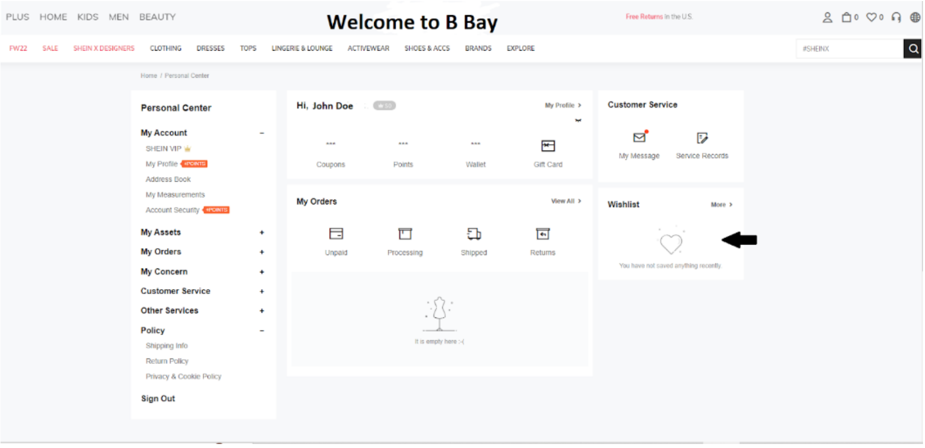


3. User Interface

The user Interface (same as the home page) will be visible after the user logs in or creates a new account. If the user is a returning user, his/her recently viewed items will be displayed. If they are new, the mostly viewed items based on their age, gender and location will be displayed.

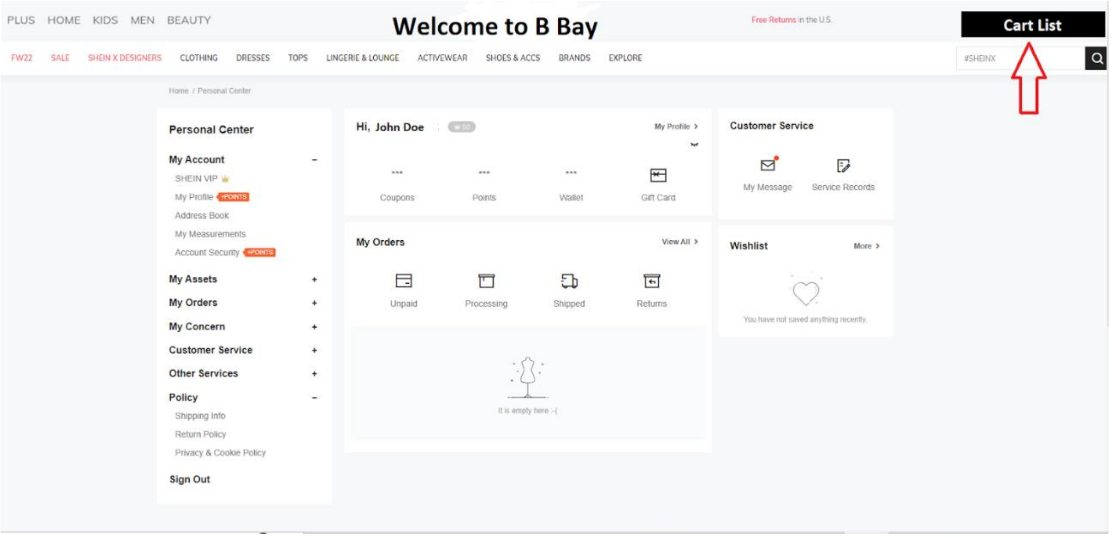
3.1Wish List

On right side of the user interface, there is a wish list button, where the user can add their desired items.



3.2 Cart List

Beside the wish list is a cart button where the user can add items they want to buy to the cart.



3.3 Chat button

Next to the wish list is a chat button where user can send and receive messages from the sellers.

3.4 Language

There is a globe icon on the far end corner of the page that enables the user to change the language

4. Administrator interface

This will allow the admin to interact with the website.

4.1 View Selling requests

The admin will see a table with sellers' information and their item description. He will either approve or reject the requests. Once approved, the item will be added to the database.

4.2 View error list

The admin will see a table with details of errors and attempts that have taken place on the website. If the error is still waiting to be fixed, he fixes it.

4.3 View returns list

The admin will see a table with users' return requests and information on the product and reason for return. The admin approves or rejects it.

5. Payment/ Auction

This can be found just before checkout. The user has the option to buy immediately or go into the auction. Payment is enabled by paypal's API.

Human interfaces:

Aside from an internet connection, A PC with a keyboard and a mouse, or a smartphone with touch screen can be used to visit the webpage.

Report format:

The report is a table with columns that include; item ID, name, name of customer and seller, date and amount spent.

It must be downloadable in pdf and default browser is chrome.

Input

First, The Data comes from user when they put in their bio upon creating an account.

Second, user submits their information and this gets stored in our database. As they navigate the website and search items, we keep track of their search history on the website using our intelligent AI tool and store it.

Thirdly, input can come from the seller by submitting a selling request form. We get the information from this form also to add it to our merchandise list once it is approved.

For example:

Seller Name	Email	Item name	Item description	specifications	Cost($)
John Doe	John.doe@outlook.com	Towels	High quality cotton Towels available for adults	28” by 54” 40” by 72”	14 20

Output

First, the data in the server from user interface of user will be sent into the database.

Secondly, the admin will receive notifications accordingly when a seller or user submits a form.

For example:

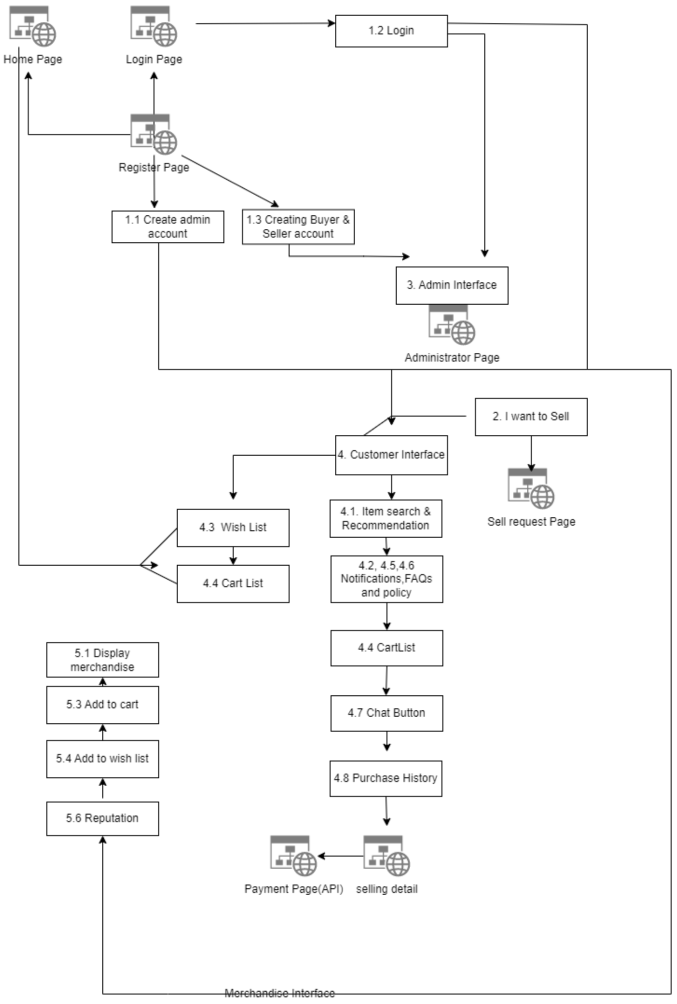
Request ID	User	Description	Cost($)	Status
2001	John Doe	High quality cotton Towels available for adults	14 20	Approve/ Deny

Thirdly, after admin approves the request, it will be stored in file. The approved items will then go to the items folder and sorted accordingly.

For returns, it is almost similar except that after approval, the admin will send a refund which will usually take 3 business days.

Format is in tabular form and is downloadable as pdf. It is stored in our cloud for future references.

Top level Design



Top level design

Performance constraints

Our system can accommodate a maximum of 500 users at once.

If a user unnecessarily and continuously clicks on a page at once, they might have some glitches.

Slow internet connection. Make sure the internet connection is stable.

Archival procedures

Archiving is mostly done once every 2 days to clean the site. Archives are saved in a folder and uploaded to cloud.

Fault handling approach

One of the most important aspects of fault handling is detecting a fault immediately and isolating it to the appropriate unit as quickly as possible. Here are some of the fault detection mechanisms we commonly use.

Sanity Monitoring: A unit monitors the health of another unit by expecting periodic health messages. The unit that is being monitored should check its sanity and send the periodic health update to the monitoring unit. The monitoring unit will report a fault if more than a specified number of successive health messages are lost.

Watchdog Monitoring: This is the hardware based monitoring technique to detect hanging hardware or software modules. The system is configured with a hardware timer that should be never allowed to timeout. The software periodically restarts the timer under normal conditions. If the software goes in an infinite loop or a hardware module gets stuck, the watchdog timer would go off. This typically leads to a hardware reset of the unit and a hardware signal to the mate unit.

Protocol Faults: If a unit fails, all the units that are in communication with this unit will encounter protocol faults. The protocol faults are inherently fuzzy in nature as they may be due to a failure of any unit in the path from the source to destination. Thus further isolation is required to identify the faulty unit.

Transient Leaky Bucket Counters: When the hardware is in operation, many transient faults may be detected by the system. Transient faults are typically handled by incrementing a leaky bucket counter. If the leaky bucket counter overflows, a fault trigger is raised. The following are few examples of transient faults.

Fault Isolation

If a unit is actually faulty, many fault triggers will be generated for that unit. The main objective of fault isolation is to correlate the fault triggers and identify the faulty unit. If fault triggers are fuzzy in nature, the isolation procedure involves interrogating the health of several units. For example, if protocol fault is the only fault reported, all the units in the path from source to destination are probed for health.

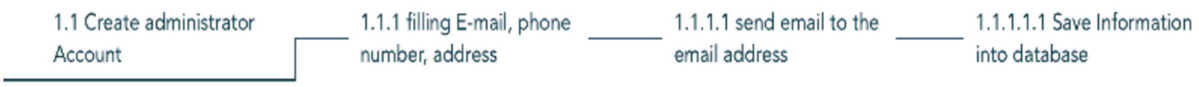
Medium Level Design –general functional characteristics

FWBS1 User Account Management

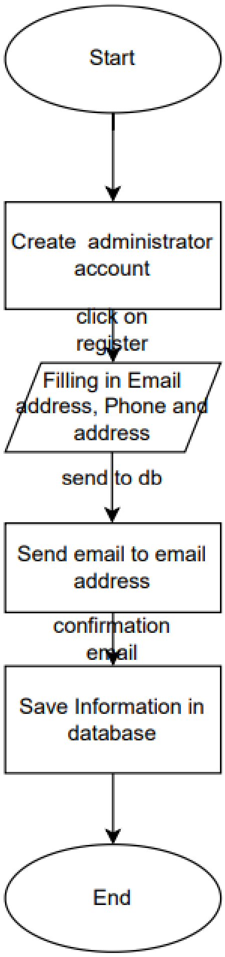
This is going to allow administrators and users (First time or returning) to create/log into their accounts. This unit consists of 3 different functionalities; creating administrator account, login and creating buyer and seller accounts.

FWBS1.1 Creating administrator account

This is going to allow administrators to create their accounts.

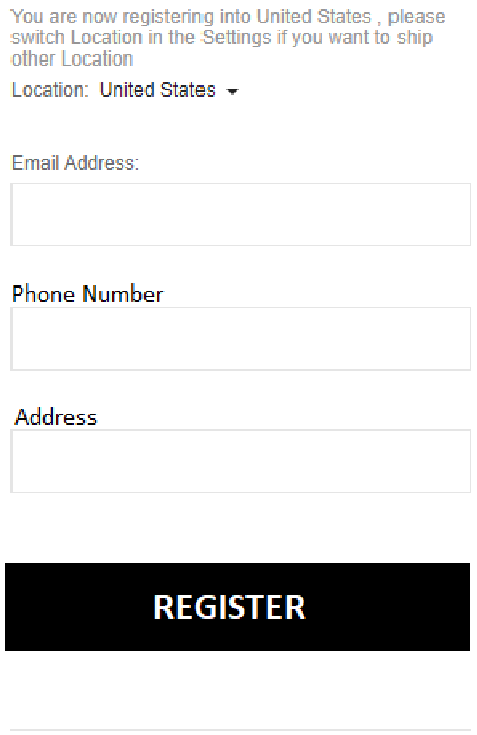

1.1 Create administrator Account
1.1.1 filling E-mail, phone number, address
1.1.1.1 send email to the email address
1.1.1.1.1 Save Information into database

Related FWBS


Start
Create administrator account
click on register
Filling in Email address, Phone and address
send to db
Send email to email address
confirmation email
Save Information in database
End

Data flow chart for FWBS1.1 creating administrator account

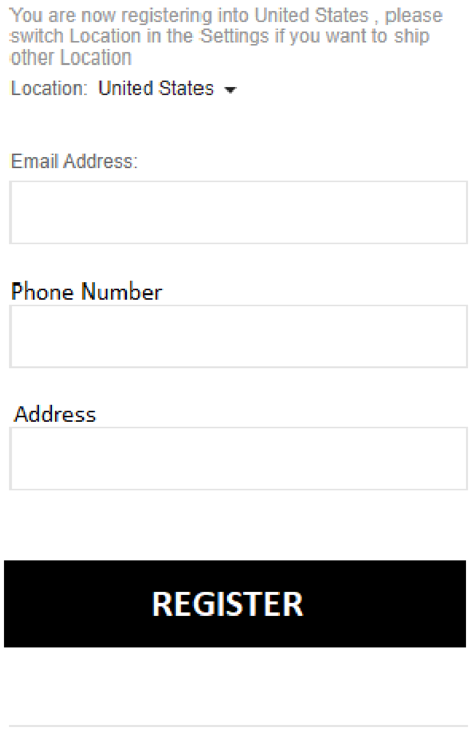
No	Name	Description	Input	Input Type	Expected Outputs
FWBS1.1.1	Filling email address, name, phone number, address	This allows user to create an account	Email, phone, address	Button	See SS2
FWBS1.1.1.1	Send email to that email address.	Email will be send to the user email address	Send email to email address	Link	See SS3



You are now registering into United States , please switch Location in the Settings if you want to ship other Location

Location: United States

Pic: SS1



You are now registering into United States , please switch Location in the Settings if you want to ship other Location

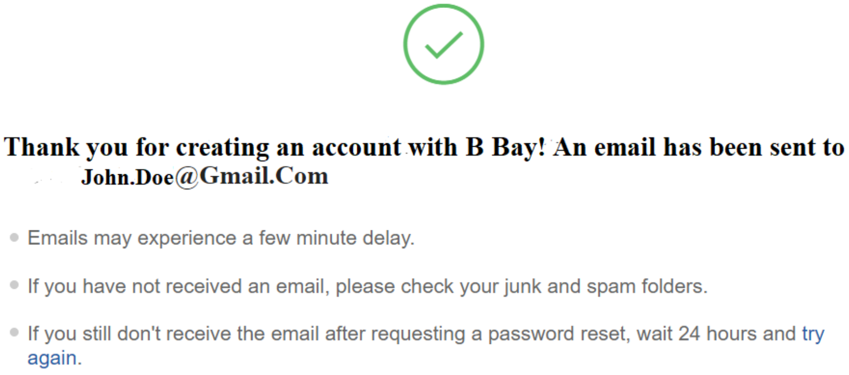
Location: United States

Email Address:

Phone Number

Address

Pic: SS2



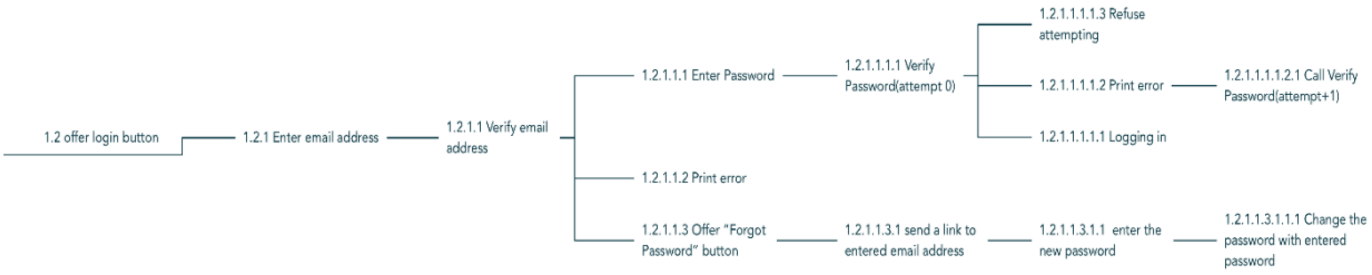
Thank you for creating an account with B Bay! An email has been sent to John.Doe@Gmail.Com

- Emails may experience a few minute delay.
- If you have not received an email, please check your junk and spam folders.
- If you still don't receive the email after requesting a password reset, wait 24 hours and try again.

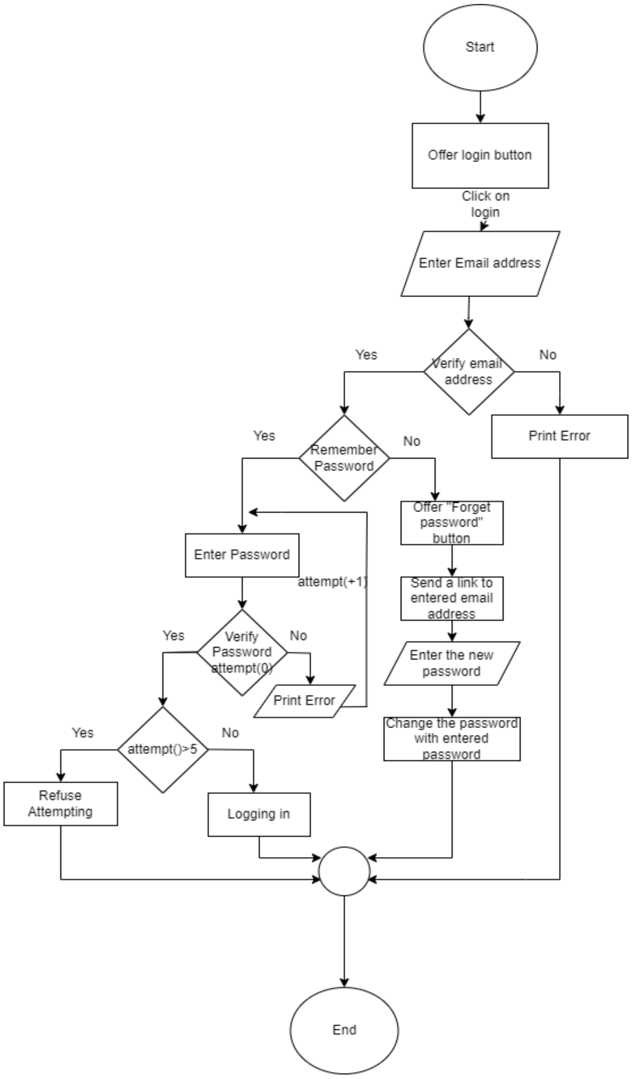
Pic: SS3

FWBS1.2 Offer login button

This is going to allow returning users to log into their accounts.


1.2 offer login button
1.2.1 Enter email address
1.2.1.1 Verify email address
1.2.1.1.1 Enter Password
1.2.1.1.1.1 Verify Password(attempt 0)
1.2.1.1.1.1.3 Refuse attempting
1.2.1.1.1.1.2 Print error
1.2.1.1.1.1.2.1 Call Verify Password(attempt+1)
1.2.1.1.1.1.1 Logging in
1.2.1.1.2 Print error
1.2.1.1.3 Offer "Forgot Password" button
1.2.1.1.3.1 send a link to entered email address
1.2.1.1.3.1.1 enter the new password
1.2.1.1.3.1.1.1 Change the password with entered password

Related FWBS

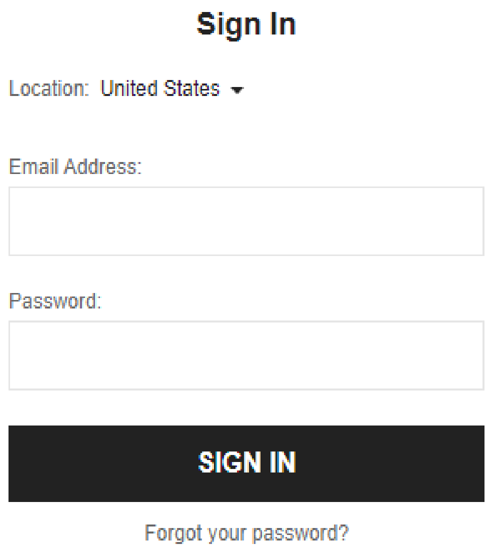


Data Flow chart for FWBS1.2 offer login button

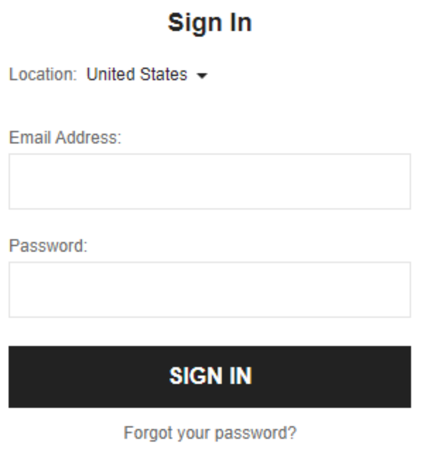
No	Name	Description	Input	Input Type	Expected Outputs
FWBS1.2	Offer login button	By clicking this button, user can login	User will click on	Button	See SS4

			“Login” button		
FWBS1.2.1.1	Verify email address	Verified email will be sent to in the user email address	User will click “Verify Email Address”	Link	See SS5
FWBS1.2.1.1.1.1	Verify Password	User will put their password and get verified	Password will be verified	Link	See SS6
FWBS1.2.1.1.3	Offer “Forget Password” button	Provide a button that will take user to a new page for new password	User will click “Forget Password Button”	Button	See SS7
FWBS1.2.1.1.3.1	Send a link to entered email address	Email will be sent to that particular email address	Email will be sent to the email address.	Link	See SS8

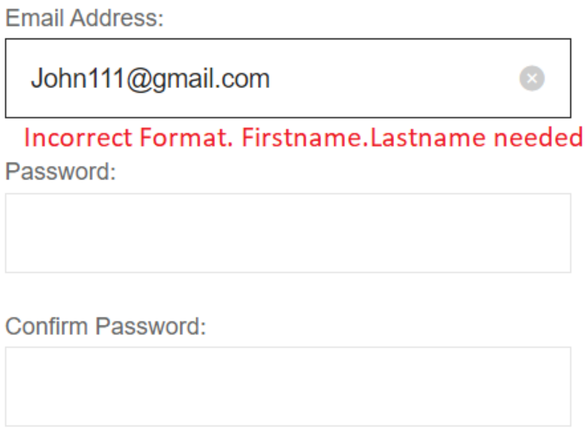
FWBS1.2 .1.1.1.1.2	Print Error	Error will be shown	Users will see error.	Link	See SS9
FWBS1.2 .1.1.1.1.1	Logging in	Welcome page will be shown	Logging in	Link	See SS10



Pic: SS4



Pic: SS4.1


Email Address:

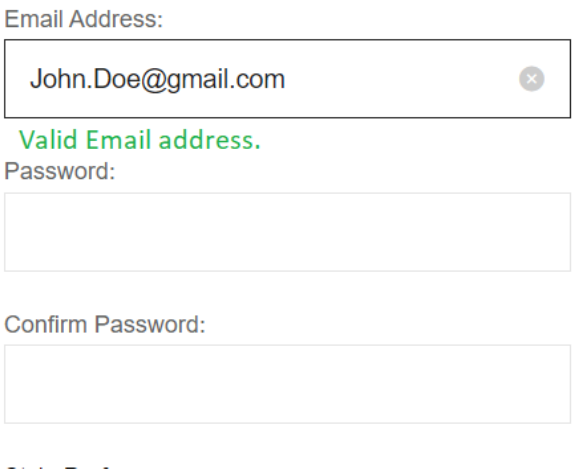
John111@gmail.com

Incorrect Format. Firstname.Lastname needed

Password:

Confirm Password:

Pic: SS5


Email Address:

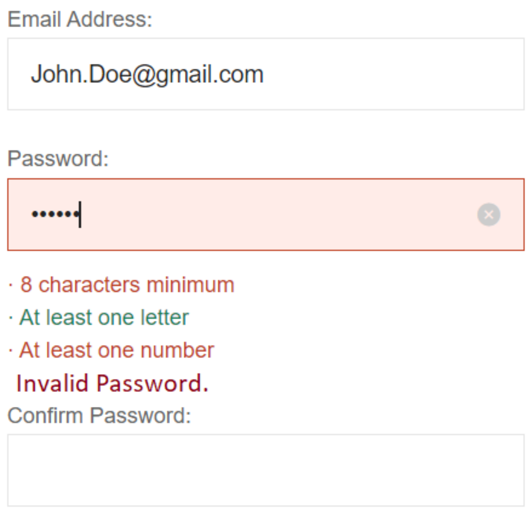
John.Doe@gmail.com

Valid Email address.

Password:

Confirm Password:

Pic: SS5.1



Email Address:

John.Doe@gmail.com

Password:

••••••

· 8 characters minimum

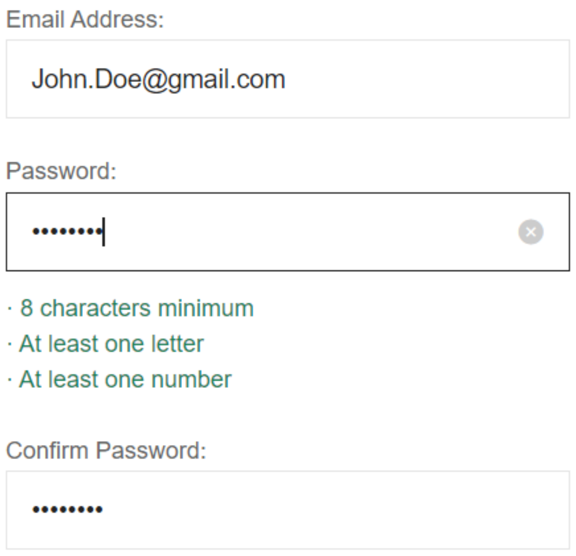
· At least one letter

· At least one number

Invalid Password.

Confirm Password:

Pic: SS6



Email Address:

John.Doe@gmail.com

Password:

••••••••

· 8 characters minimum

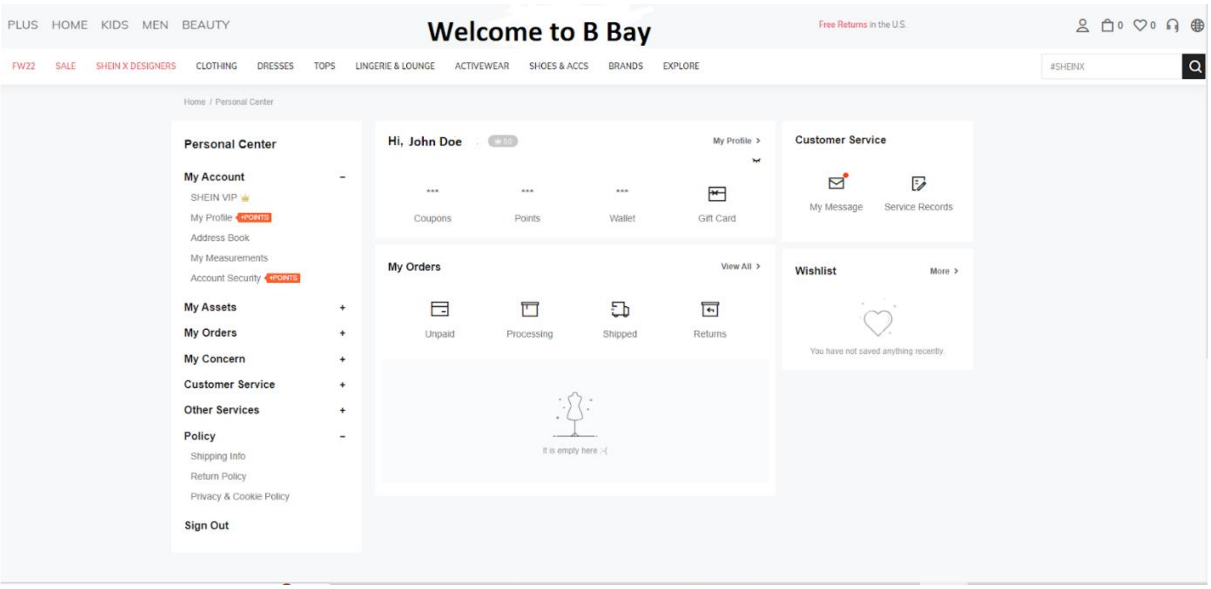
· At least one letter

· At least one number

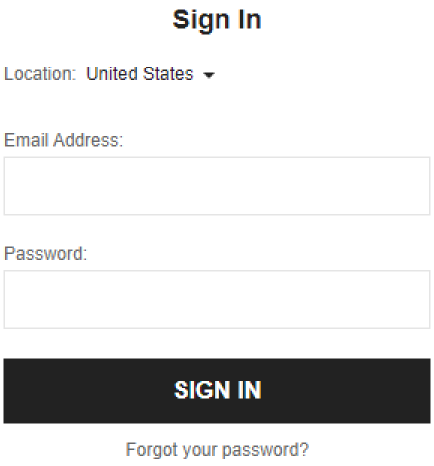
Confirm Password:

••••••••

Pic: SS6.1



Pic: SS10


Sign In

Location: United States

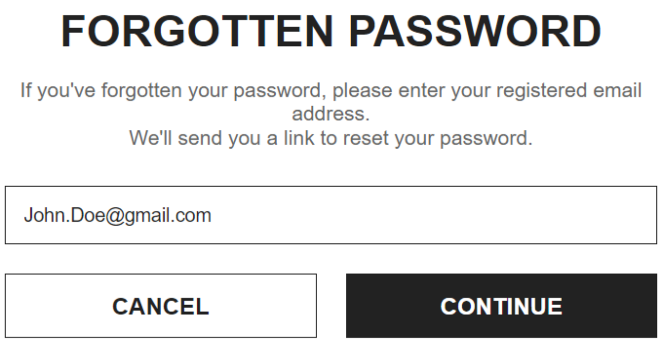
Email Address:

Password:

SIGN IN

Forgot your password?

Pic: SS7

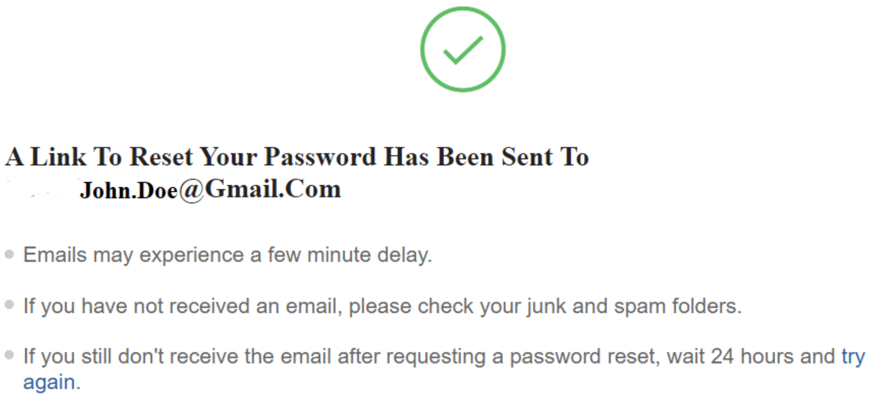


FORGOTTEN PASSWORD

If you've forgotten your password, please enter your registered email address.
We'll send you a link to reset your password.

John.Doe@gmail.com

Pic: SS7.1



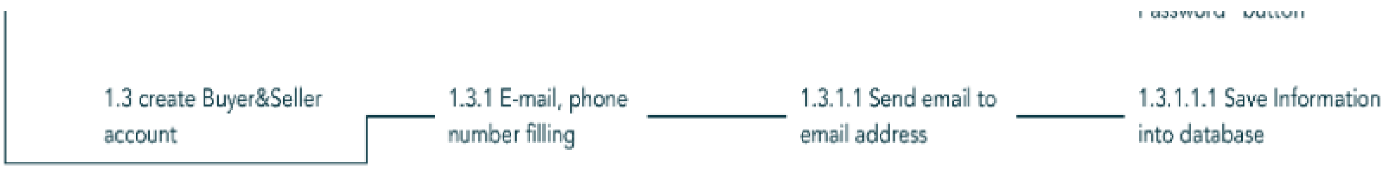
A Link To Reset Your Password Has Been Sent To John.Doe@Gmail.Com

- Emails may experience a few minute delay.
- If you have not received an email, please check your junk and spam folders.
- If you still don't receive the email after requesting a password reset, wait 24 hours and try again.

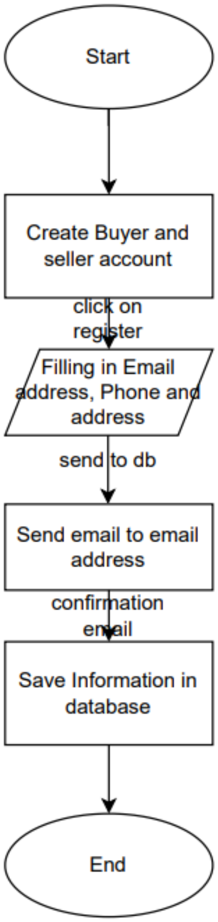
Pic: SS8

FWBS1.3 Creating Buyer and Seller account

This is going to allow buyers and sellers to create their accounts.


Password button
1.3 create Buyer&Seller account
1.3.1 E-mail, phone number filling
1.3.1.1 Send email to email address
1.3.1.1.1 Save Information into database

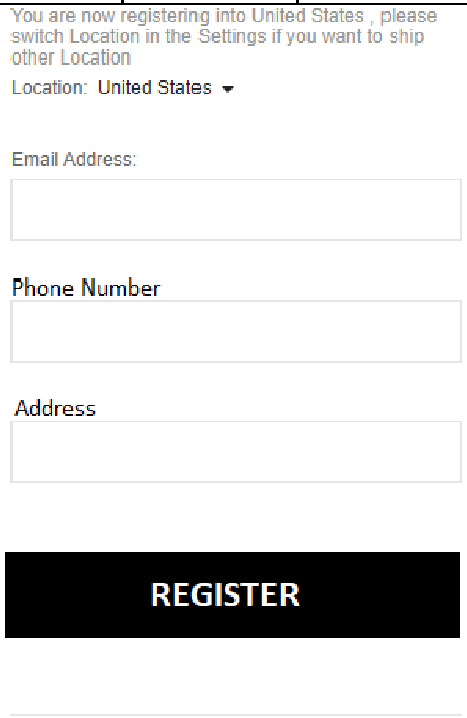
Related FWBS


Start
Create Buyer and seller account
click on register
Filling in Email address, Phone and address
send to db
Send email to email address
confirmation email
Save Information in database
End

Data flow chart for FWBS1.3 creating buyer and seller accoun

No	Name	Description	Input	Input Type	Expected Outputs

FWBS1.3.1	E-mail, phone number, name, address filling	This allows user to create an account	Register	Button	See SS11
FWBS1.3.1.1	Send email to email address	Email will be send to the user email address	Email, phone, address	Button	See SS12



You are now registering into United States , please switch Location in the Settings if you want to ship other Location

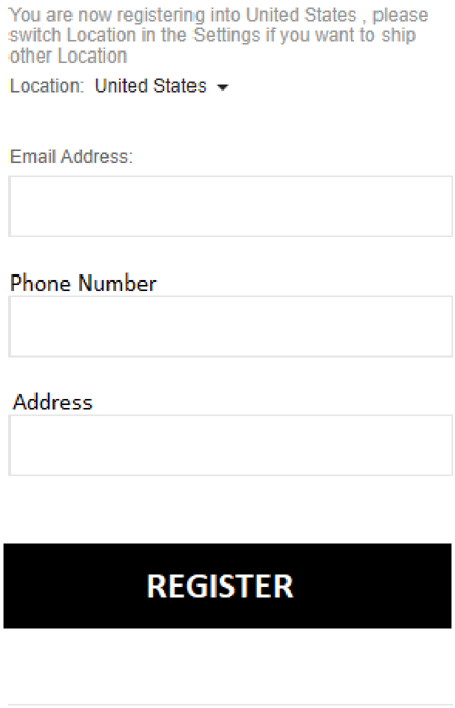
Location: United States

Email Address:

Phone Number

Address

Pic: SS11



You are now registering into United States , please switch Location in the Settings if you want to ship other Location

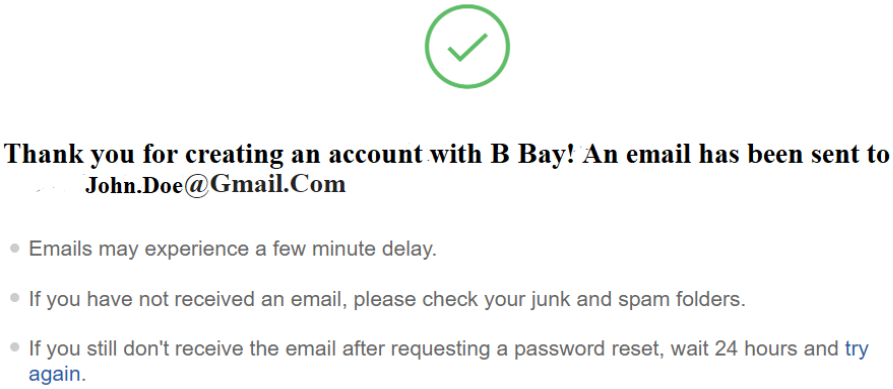
Location: United States

Email Address:

Phone Number

Address

Pic: SS12



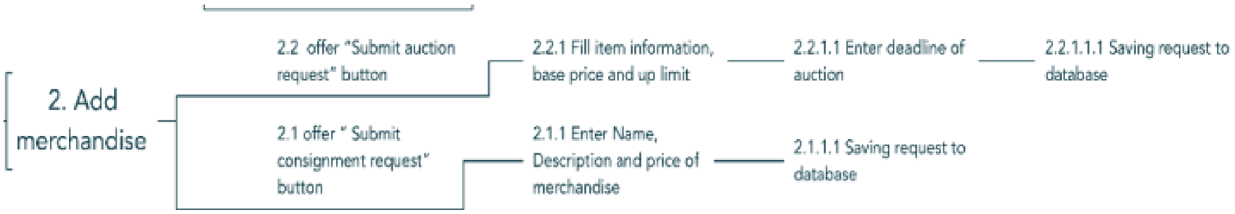
Thank you for creating an account with B Bay! An email has been sent to John.Doe@Gmail.Com

- Emails may experience a few minute delay.
- If you have not received an email, please check your junk and spam folders.
- If you still don't receive the email after requesting a password reset, wait 24 hours and try again.

Pic: SS12.1

FWBS2 I want to sell

This cluster contains a request specification of selling. When users try to sell something with B-bay system, they should be able to submit a request of selling by functions in this cluster to the administrator.


2. Add merchandise
2.2 offer "Submit auction request" button
2.2.1 Fill item information, base price and up limit
2.2.1.1 Enter deadline of auction
2.2.1.1.1 Saving request to database
2.1 offer " Submit consignment request" button
2.1.1 Enter Name, Description and price of merchandise
2.1.1.1 Saving request to database

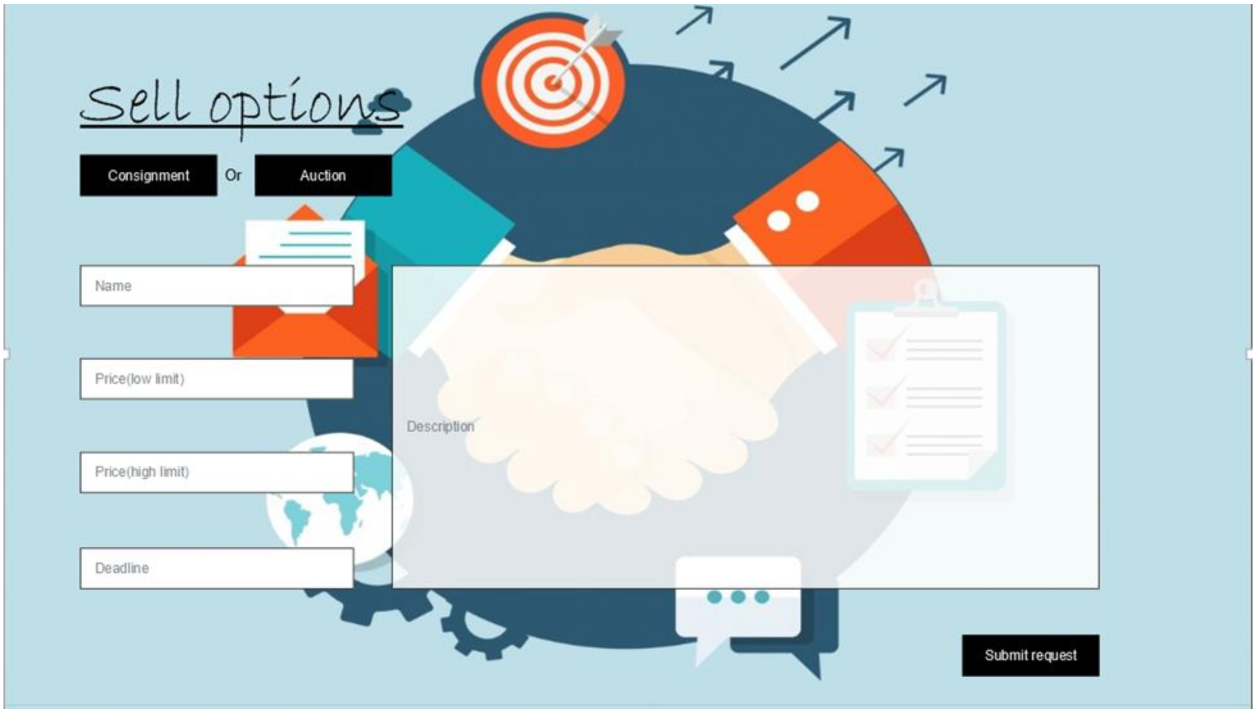
Related FWBS

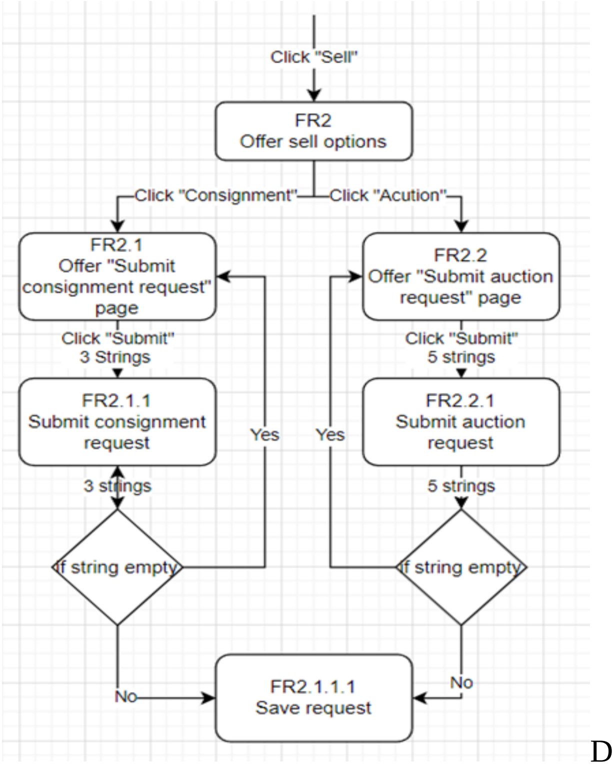
FWBS No.	Name	Description	Input	Input Type	Expected Output
FWBS2	Offer sell options	This function shall navigate to selling page after user click “sell” button	User click "sell" button in SS10		Navigate to sell request page as I8. Including two buttons: "Consignment" and "Auction" also including filling boxes of Name,

					description, price, deadline and a "submit" button
FWBS2.1	offer "Submit consignment request" page	This function shall offer a button “consignment”, After user input, offer boxes for user to enter consignment specifications.	user click "Consignment" button in SS13		AS I8 but disable input box of deadline and Price(low limit), highlight “consignment” button
FWBS2.1.1	Submit consignment request	This function shall offer a button “Submit”, by click it, function will take inputs from FWBS2.1 offered	user click "submit" button, 3 strings	strings	If string not empty, call FWBS2.1.1.1, or back to FWBS2.1

		and call FWBS2.1.1.1			
FWBS2.1.1.1	Save request	save all request information to structure consignment. And save it to data base.	Strings of selling detail	Strings	save information to database
FWBS2.2	offer "Submit auction request" page	This function shall offer a button “Auction”, After user input, offer boxes for user to enter auction specifications.	user click "Auction" button in SS13		As I8 that every input box enabled. Highlight “Auction” button
FWBS2.2.1	Submit auction request	This function shall offer a button “Submit”, by click it,	user click "submit" button, 5 strings	strings	If string not empty, call FWBS2.1.1.1,

		function will save all request to structure auction. And save it to data base.			or back to FWBS2.2



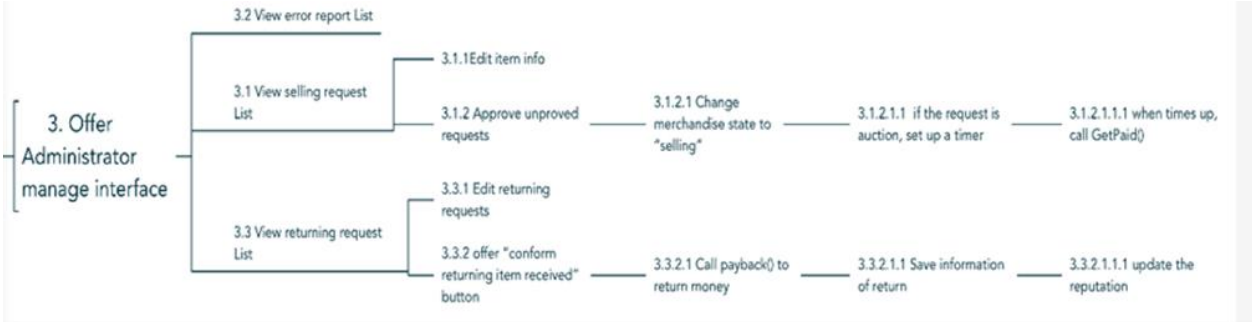


Data Flow Chart for FWBS2 Offer sell options

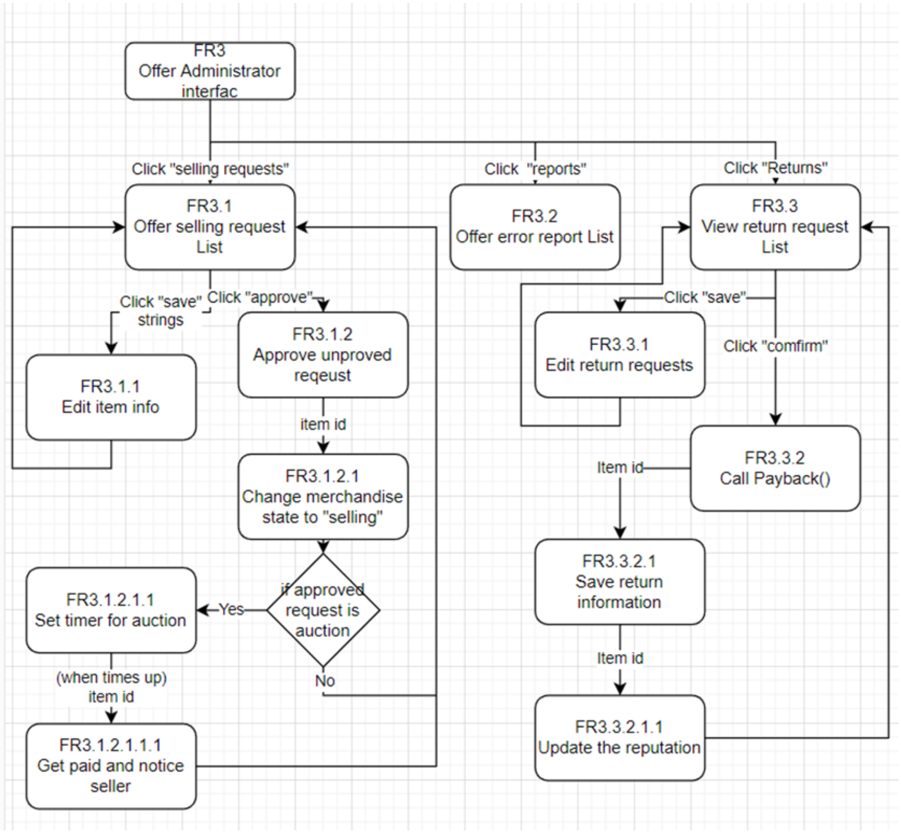
Pic : SS 13

FWBS3 Administrator interface

This cluster contains a request specification of Administrator interface, which allow administrators to manage requests of selling or return, viewing reports and so on…


3. Offer Administrator manage interface
3.2 View error report List
3.1 View selling request List
3.1.1Edit item info
3.1.2 Approve unproved requests
3.1.2.1 Change merchandise state to "selling"
3.1.2.1.1 if the request is auction, set up a timer
3.1.2.1.1.1 when times up, call GetPaid()
3.3 View returning request List
3.3.1 Edit returning requests
3.3.2 offer "conform returning item received" button
3.3.2.1 Call payback() to return money
3.3.2.1.1 Save information of return
3.3.2.1.1.1 update the reputation

Related FWBS



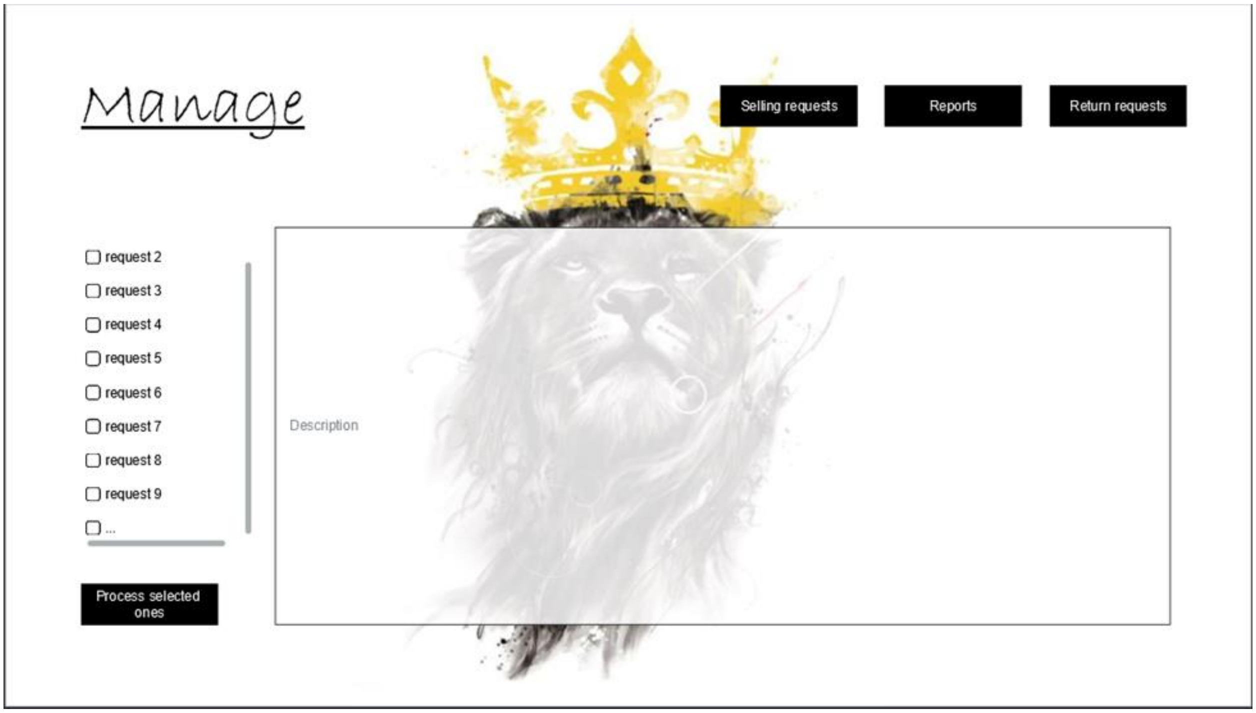
Data Flow Chart for FWBS Offer Administrator interface

No.	Name	Description	Input	Input Type	Expected Output
FWBS3	Offer Administrator interface	This function shall navigate to Administrator page after user click “manage” button	User click "manage" button in I7		Navigate to Administrator page as SS14. Including three buttons: "Selling requests", "Returns" and "reports", also show requests or reports in database, each one in list have a check list beside.
FWBS3.1	Offer selling request list	This function shall offer a list of selling requests in boxes and offers select	User click "selling requests" button in SS14		The List in SS14will show selling requests from database

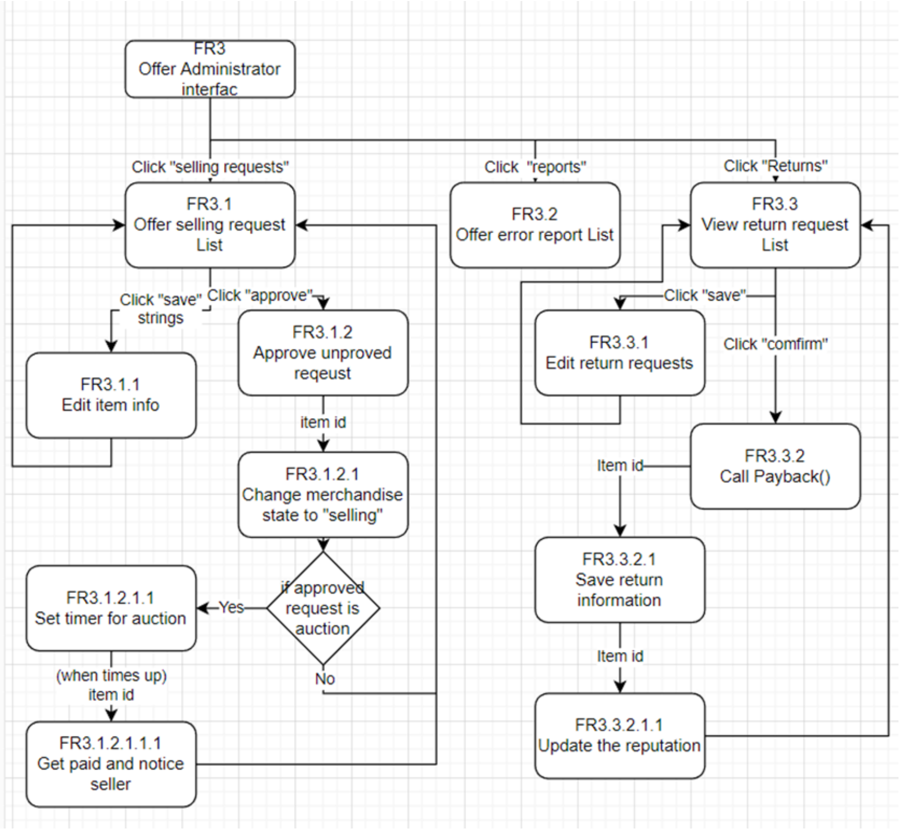
		boxes for each.			
FWBS3.1.1	Edit item info	This function shall update after information FWBS3.1 boxes has been changed	User click the any one in list, and change whatever displayed in “Description” area, in SS14	strings	Save all strings, overwrite the original strings
FWBS3.1.2	Approve unproved request	This function shall be called after click “Process selected ones” button, it will call FWBS3.1.2.1 for each selected ones.	User click check box and click “Process selected ones”		call FWBS3.1.2.1 with clicked item id, and then refresh the list SS14
FWBS3.1.2.1	Change merchandise state to "selling"	it shall update “selling” bool to true.	item id	integer	overwrite original string, if request type is "auction" call FWBS3.1.2.1.1
FWBS3.1.2.1.1	Set timer for auction	This function shall search item id in DB, and set a timer based on deadline found.	item id	integer	Set timer based on data, call FWBS3.1.2.1.1.1 when times up
FWBS3.1.2.1.1.1	Get paid and notice seller	This function shall Navigate to APIs to get paid, and then send a notice to seller.	item id	integer	if buyer exist, call GetPaid(), then write notice with item seller id to database

FWBS3.2	Offer error reports list	This function shall offer a list of error reports .	User click "reports" in SS14		navigate to Error report list, display reports in List and description area in SS14
FWBS3.3	View returning request List	This function shall offer a list of return requests in boxes and offers select boxes for each .	User click "Returns" in SS14		As SS14but switch the requests list to return requests list
FWBS3.3.1	Edit return requests	This function shall update after information FWBS3.3 boxes has been changed	User click "save" in SS14, all string in list	strings	Save all strings, overwrite the original strings
FWBS3.3.2	Call payback()	This function shall Navigate to APIs to pay back.	User click any "confirm" button in SS14		Call FWBS3.3.2.1 with item id
FWBS3.3.2.1	Save information of return	This function shall save return information base on input item id	Item id	integer	Save information to database, call FWBS3.3.2.1.1
FWBS3.3.2.1.1	Update the reputation	This function shall get user id by search item id in DB. update reputation of user base on current value	item id	integer	save new calculated reputation with seller id

		this user buy/sell.			



Pic: SS14



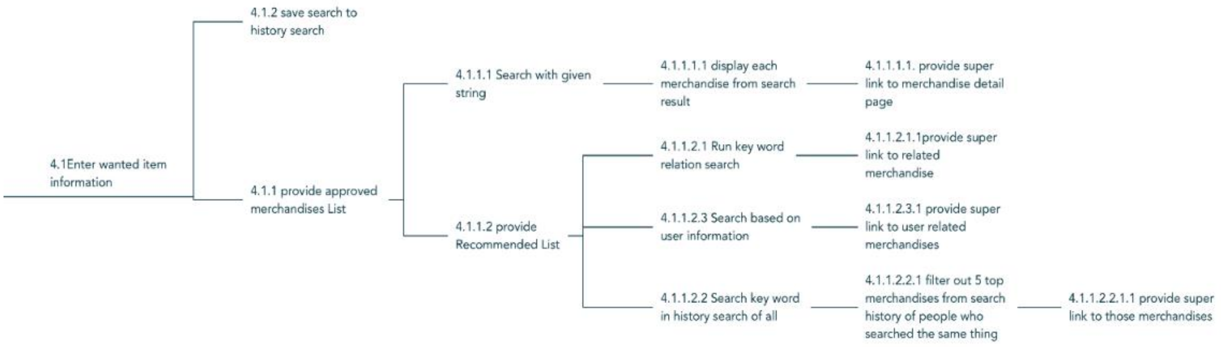
Data Flow Chart for FR Offer Administrator interfac

FWBS4 Forming Customer Interface

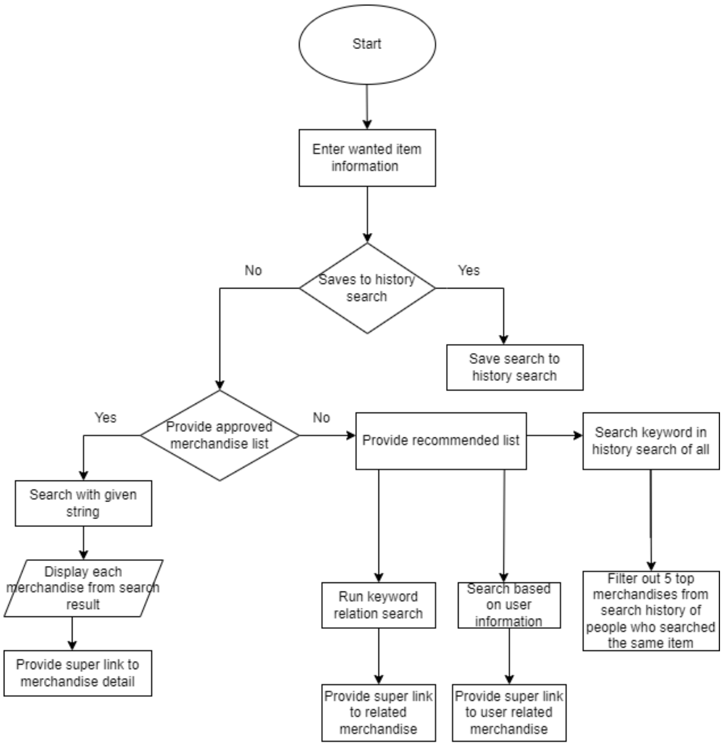
This cluster shows how the user interacts with the website. It will explain what the user sees and how he manages his purchase.

FWBS4.1 Item search and Recommendation

This allows users to search for what they are looking for. We use the data to give recommendations in the future.


4.1Enter wanted item information
4.1.2 save search to history search
4.1.1 provide approved merchandises List
4.1.1.1 Search with given string
4.1.1.1.1 display each merchandise from search result
4.1.1.1.1. provide super link to merchandise detail page
4.1.1.2 provide Recommended List
4.1.1.2.1 Run key word relation search
4.1.1.2.1.1provide super link to related merchandise
4.1.1.2.3 Search based on user information
4.1.1.2.3.1 provide super link to user related merchandises
4.1.1.2.2 Search key word in history search of all
4.1.1.2.2.1 filter out 5 top merchandises from search history of people who searched the same thing
4.1.1.2.2.1.1 provide super link to those merchandises

Related FWBS


Start
Enter wanted item information
Saves to history search
No
Yes
Save search to history search
Provide approved merchandise list
Yes
No
Search with given string
Display each merchandise from search result
Provide super link to merchandise detail
Provide recommended list
Search keyword in history search of all
Run keyword relation search
Search based on user information
Filter out 5 top merchandises from search history of people who searched the same item
Provide super link to related merchandise
Provide super link to user related merchandise

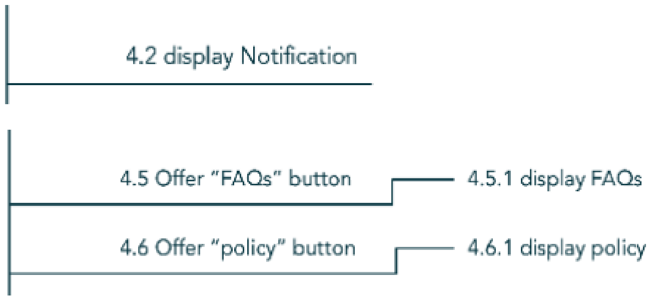
Data flow chart for FWBS4.1 Item search and recommendation

No	Name	Description	Input	Input Type	Expected Outputs
FWBS4.1.1	Provide approved	This is what we have after the seller sent a request and	Search item	Search button	User can see their desired merchandise

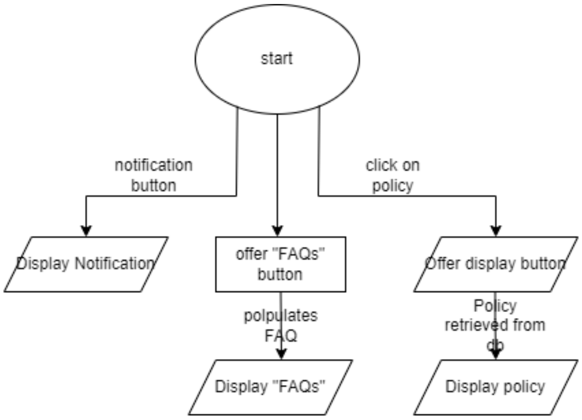
	merchandise list	admin accepted.			
FWBS4.1.2	Save search to history search	Saves search history for future use	Save button	Button	Saves history in database
FWBS4.1.1.2	Provide recommended list	Helps the user to shop by displaying the items they like.	Do not save search history	Button	Leads user to the items list

FWBS4.2,5,6 Notifications, FAQs and Policy

This cluster merges the notification functionality, FAQs functionality and policy functionality.



Related FWBS

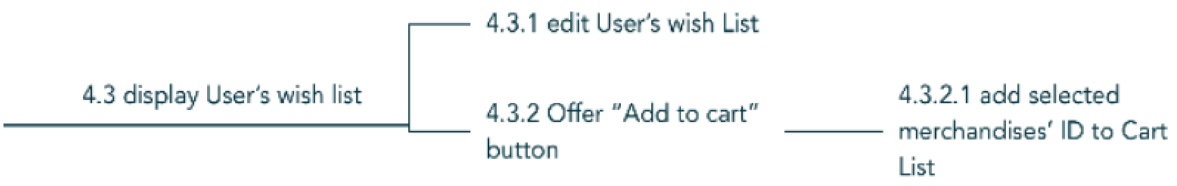


Data flow Chart for FWBS4.2,4.5 and 4.6

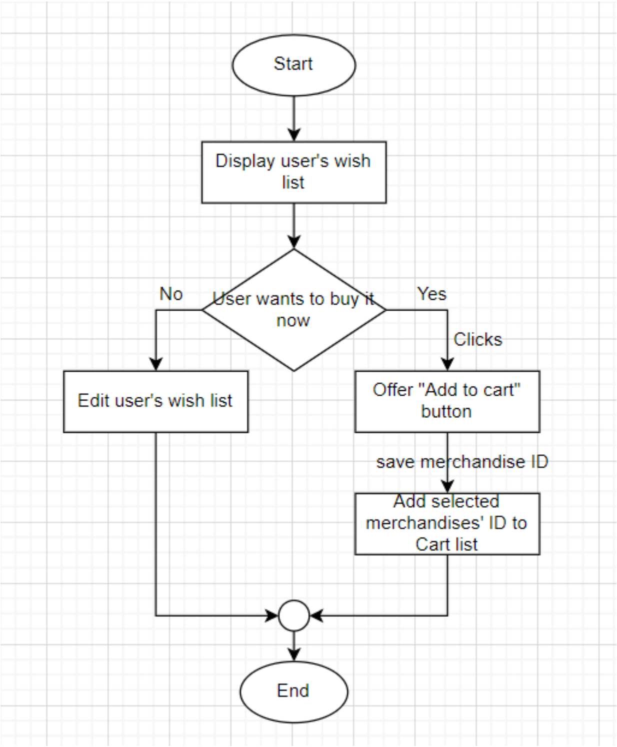
No	Name	Description	Input	Input Type	Expected Outputs
FWBS4.2	Display notification	Helps users receive notifications	User clicks on notification	Link	Notifications will be showed
FWBS4.5	Offer “FAQs” button	Helps display frequently asked questions to help users with quick information.	User clicks on FAQs button	Button	FAQs are displayed
FWBS4.6	Offer “policy” button	Displays the B Bay’s policies	User clicks on policy button	Button	Policy information is displayed

FWBS4.3 Display User’s Wish List

This allows users to show the wish list and also allows them to edit the wish list. And this will offer adding to cart options.

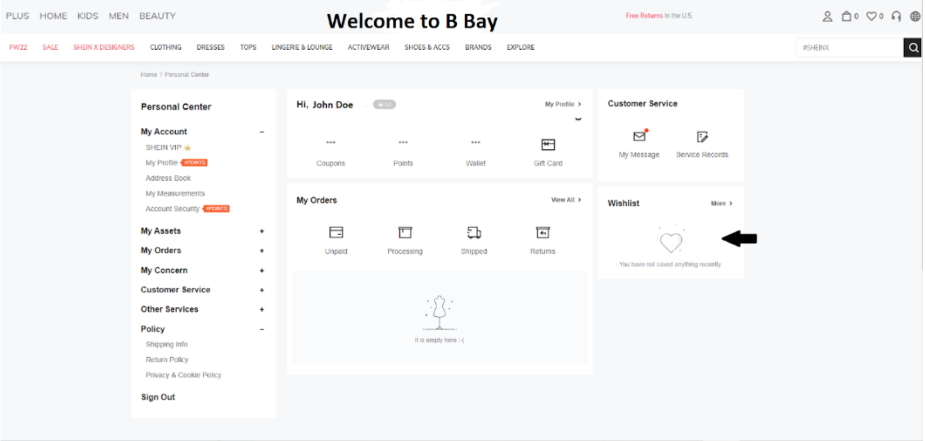

4.3 display User’s wish list
4.3.1 edit User’s wish List
4.3.2 Offer “Add to cart” button
4.3.2.1 add selected merchandises’ ID to Cart List

Related FWBS



Data Flow Chart for FWBS4.3 display user’s wish list

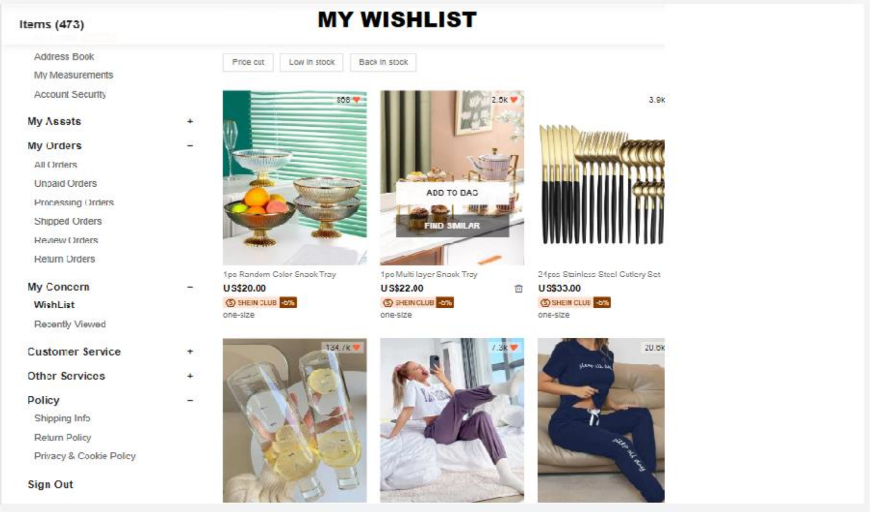
No	Name	Description	Input	Input Type	Expected Output
FWBS4.3	Display user’s wish list	Enables user to view a summary of items they like.	User clicks wish list	Button	All items that were liked by the user will be populated.
FWBS4.3.2	Offer “add to cart” button	Helps buyers to transfer items from their wish list to cart list	User clicks add to cart button	Button	Items go to cart list from wish list.



It is empty here :-(

You have not saved anything recently

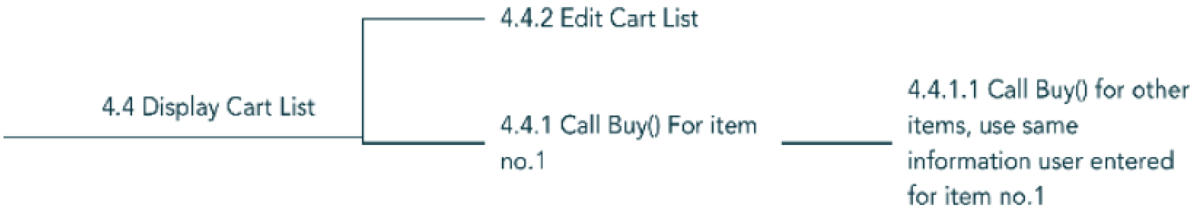
Pic: SS15



Pic: SS 15.1

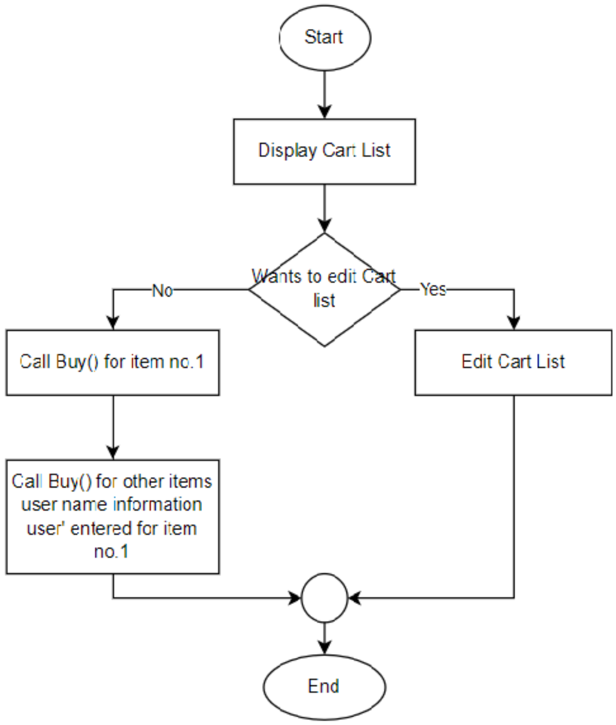
FWBS4.4 Display Cart List

This will show the user list of products they bought and it will also allow the user to edit the cart list.



4.4.2 Edit Cart List
4.4 Display Cart List
4.4.1 Call Buy() For item no.1
4.4.1.1 Call Buy() for other items, use same information user entered for item no.1

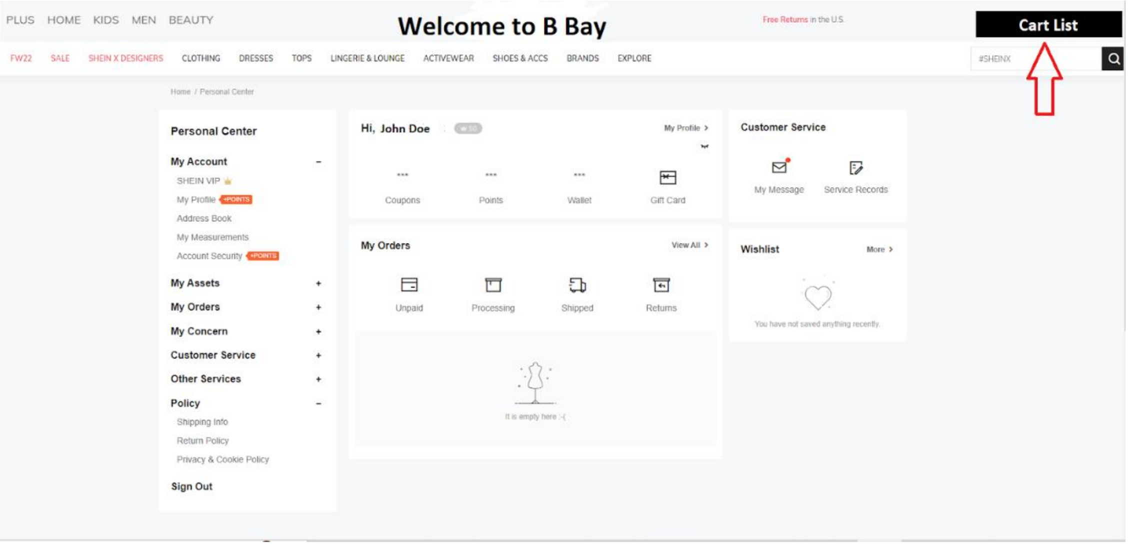
Related FWBS



Data Flow Chart for FWBS4.4 display cart list

No	Name	Description	Input	Input Type	Expected Output
FWBS4.4	Display cart List	Enables user to see what is in their cart	User will click to cart list	Button	The items and their prices are populated.
FWBS4.4.1	Buy functionality	Enables user to pay for item in cart	User will call Buy() to buy items	Button	User will be taken to payment page where they will be

					asked card info.



PLUS HOME KIDS MEN BEAUTY

Welcome to B Bay

Free Returns in the U.S.

Cart List

FW22 SALE SHEIN X DESIGNERS CLOTHING DRESSES TOPS LINGERIE & LOUNGE ACTIVEWEAR SHOES & ACCS BRANDS EXPLORE

#SHEINX

Home / Personal Center

Personal Center

My Account -
SHEIN VIP
My Profile +POINTS
Address Book
My Measurements
Account Security +POINTS

My Assets +
My Orders +
My Concern +
Customer Service +
Other Services +
Policy -
Shipping Info
Return Policy
Privacy & Cookie Policy

Sign Out

Hi, John Doe

My Profile >

Coupons Points Wallet Gift Card

My Orders View All >

Unpaid Processing Shipped Returns

It is empty here :-(

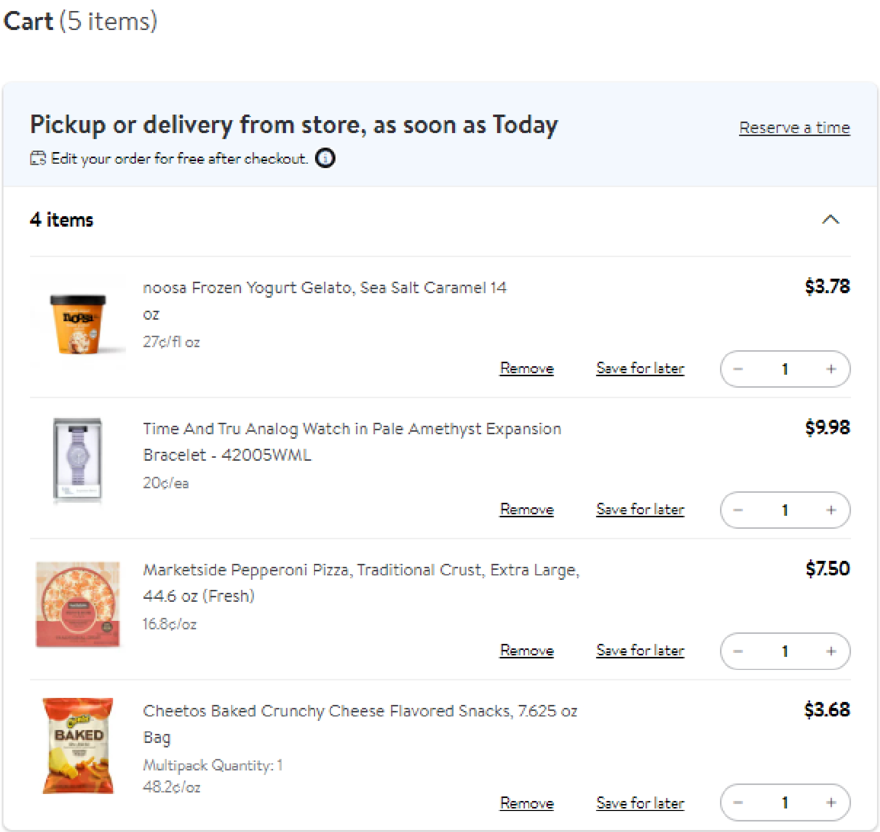
Customer Service

My Message Service Records

Wishlist More >

You have not saved anything recently.

Pic: SS16



Cart (5 items)

Pickup or delivery from store, as soon as Today

Reserve a time

Edit your order for free after checkout.

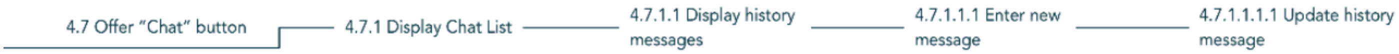
4 items

Item	Price
noosa Frozen Yogurt Gelato, Sea Salt Caramel 14 oz 27¢/fl oz Remove Save for later − 1 +	$3.78
Time And Tru Analog Watch in Pale Amethyst Expansion Bracelet - 42005WML 20¢/ea Remove Save for later − 1 +	$9.98
Marketside Pepperoni Pizza, Traditional Crust, Extra Large, 44.6 oz (Fresh) 16.8¢/oz Remove Save for later − 1 +	$7.50
Cheetos Baked Crunchy Cheese Flavored Snacks, 7.625 oz Bag Multipack Quantity: 1 48.2¢/oz Remove Save for later − 1 +	$3.68

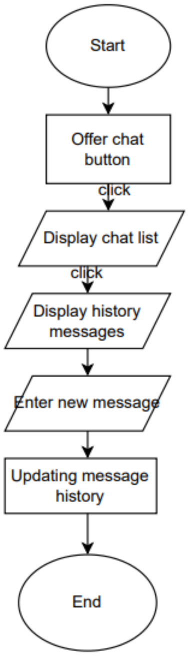
Pic: SS17

FWBS4.7 Offer “Chat” button

This enables the user to send and receive messages. It also shows message history.


4.7 Offer “Chat” button
4.7.1 Display Chat List
4.7.1.1 Display history messages
4.7.1.1.1 Enter new message
4.7.1.1.1.1 Update history message

Related FWBS

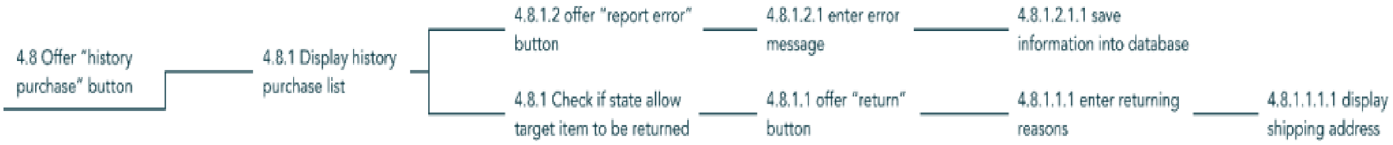


Data Flow Chart for FWBS4.7 Offer “Chat” button

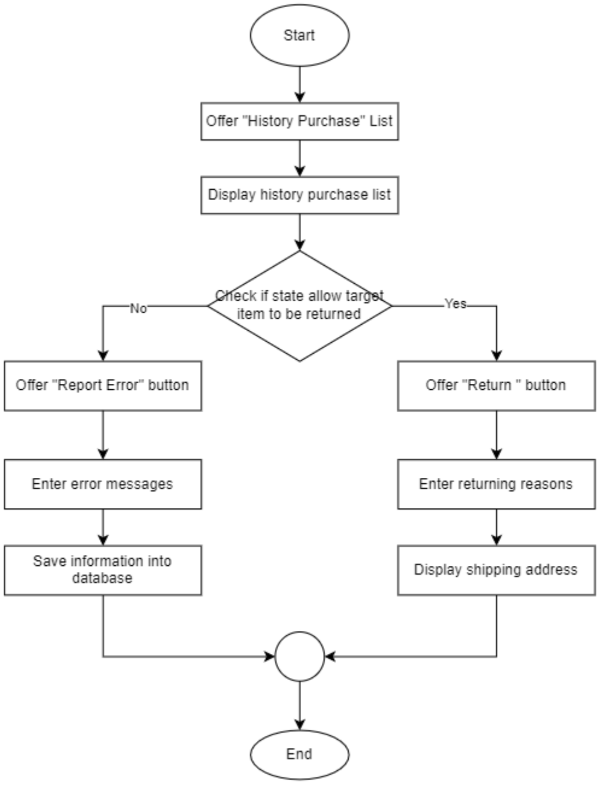
No	Name	Description	Input	Input Type	Expected Output
FWBS4.7.1	Display chat list	Enables user to see other people he/she can chat with	User clicks on chat list	link	List of people with contacts populates
FWBS4.7.1.1	Display message history	Enables users to see the old messages	User clicks on display message history	link	Old messages populate in a list
FWBS4.7.1.1.1	Enter new message	Provides a text box for user to type in message	User types message	Text box	Send option is displayed

FWBS4.8 Offer "History Purchase" button

This will show the user's purchase history and return policy.


4.8 Offer "history purchase" button
4.8.1 Display history purchase list
4.8.1.2 offer "report error" button
4.8.1.2.1 enter error message
4.8.1.2.1.1 save information into database
4.8.1 Check if state allow target item to be returned
4.8.1.1 offer "return" button
4.8.1.1.1 enter returning reasons
4.8.1.1.1.1 display shipping address

Related FWBS


Start
Offer "History Purchase" List
Display history purchase list
Check if state allow target item to be returned
No
Yes
Offer "Report Error" button
Enter error messages
Save information into database
Offer "Return " button
Enter returning reasons
Display shipping address
End

Data Flow Chart for FWBS4.8 Offer “History Purchase” button

No	Name	Description	Input	Input Type	Expected Output
FWBS4.8	Offer history purchase button	This enables users to see the items they have purchased with just a click.	User will click “History purchase” button	Button	The user’s purchase history will be populated including the dates.
FWBS4.8.1.2	Offer “report error” button	Helps admin to see the number of errors and attempts that took place.	User Will click “report Error” button	Button	A table showing the errors will be displayed including device

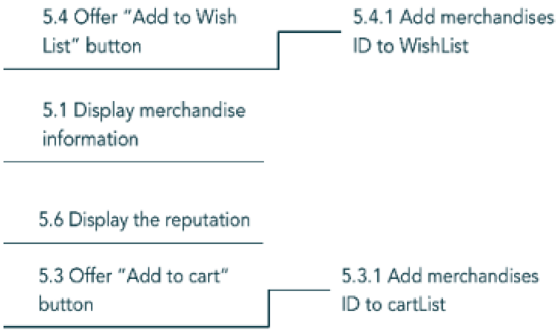
					information and location.
FWBS4.8.1.1	Offer “return” button	This gives buyers the opportunity to return unwanted items.	User will click “Return” button	Button	A form will be displayed for buyer to fill in their information and reason for the return.

FWBS5 Provide Interface for each merchandise

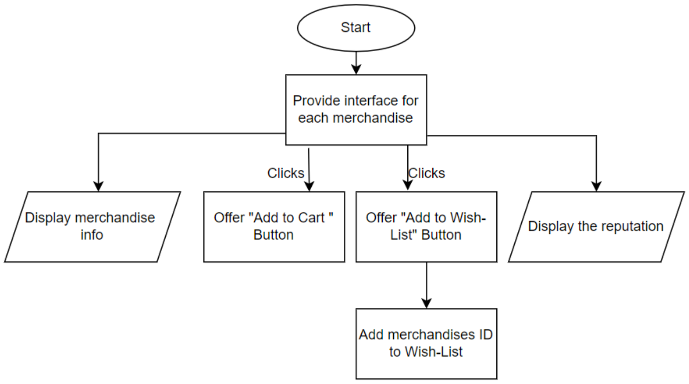
This cluster shows how merchandise is organized within the website. It also explains how bid and payment systems operate.

FWBS5.1,3,4,6 Offer “History Purchase” button

This is a merged cluster consisting of the user's purchase history, user’s reputation and functionalities that will allow users to view items information and add to cart.


5.4 Offer “Add to Wish List” button
5.4.1 Add merchandises ID to WishList
5.1 Display merchandise information
5.6 Display the reputation
5.3 Offer “Add to cart” button
5.3.1 Add merchandises ID to cartList

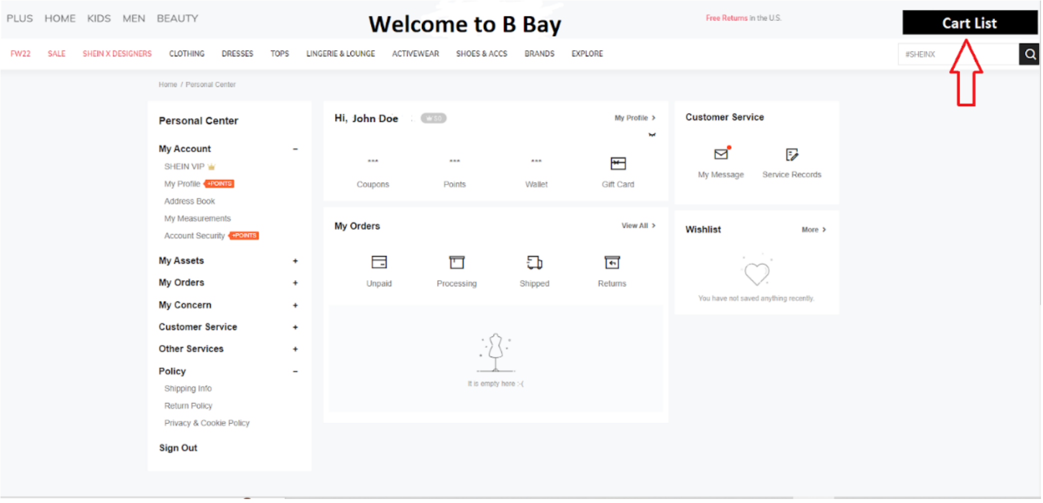
Related FWBS


Start
Provide interface for each merchandise
Clicks
Clicks
Display merchandise info
Offer "Add to Cart " Button
Offer "Add to Wish-List" Button
Display the reputation
Add merchandises ID to Wish-List

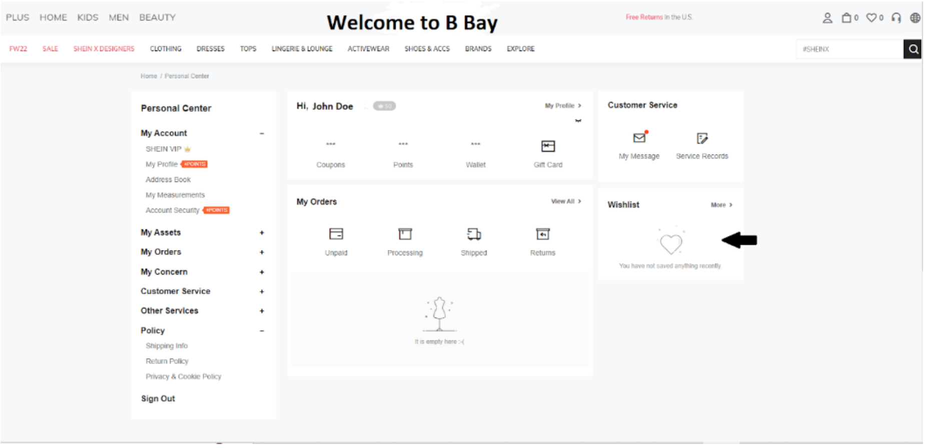
Data Flow Chart for FWBS 5.1, 5.3, 5.4 and 5.6

No	Name	Description	Input	Input Type	Expected Output
FWBS5.1	Display merchandise information	By clicking user will be taken to link from where item information can be seen	User will click merchandise info	Link	Merchandise's information will be seen
FWBS5.3	Offer "Add to Cart" button	This button will help user to add item to user cart list	User will click "Add to Cart" button	Button	See SS18
FWBS5.4	Offer "Add to Wish List" button	This button will help user to add item to user wish list	User will click "Add to Wish-List" button	Button	See SS19
FWBS5.6	Display the reputation	By clicking this button, user will go to a link which shows the	User will click display reputation	Link	Reputation will be seen

		reputation of that user			



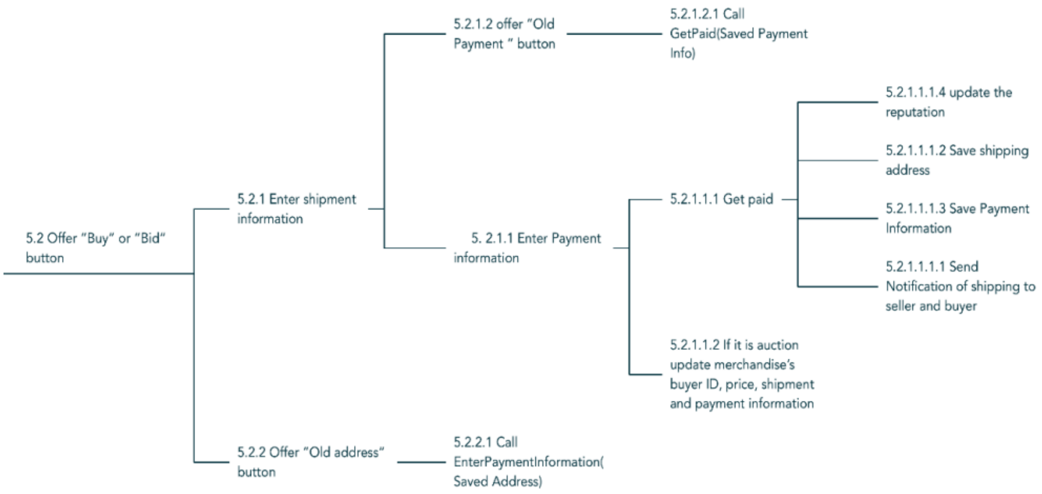
Pic: SS 18



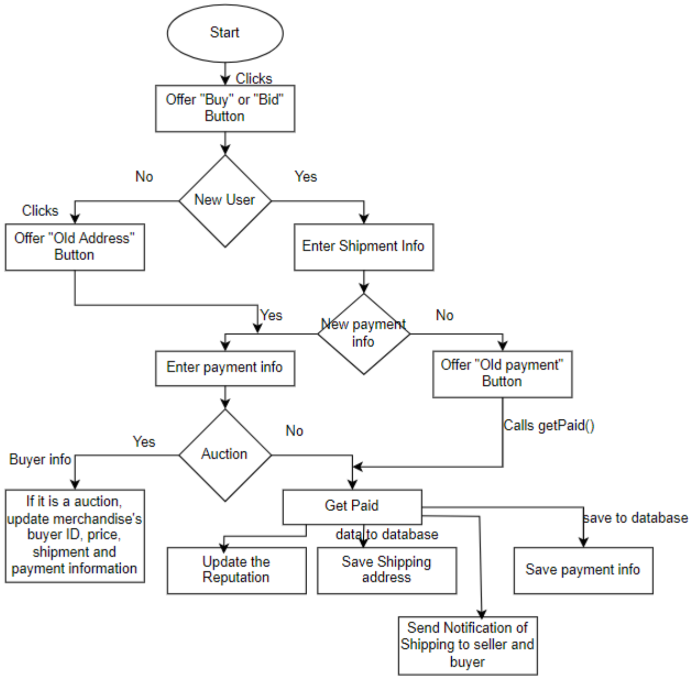
Pic: SS 19

FWBS5.2 Offer “Buy” or “Bid” button

This will show user buy and auction options and show payment interface.


5.2 Offer "Buy" or "Bid" button
5.2.1 Enter shipment information
5.2.1.2 offer "Old Payment " button
5.2.1.2.1 Call GetPaid(Saved Payment Info)
5. 2.1.1 Enter Payment information
5.2.1.1.1 Get paid
5.2.1.1.1.4 update the reputation
5.2.1.1.1.2 Save shipping address
5.2.1.1.1.3 Save Payment Information
5.2.1.1.1.1 Send Notification of shipping to seller and buyer
5.2.1.1.2 If it is auction update merchandise's buyer ID, price, shipment and payment information
5.2.2 Offer "Old address" button
5.2.2.1 Call EnterPaymentInformation(Saved Address)

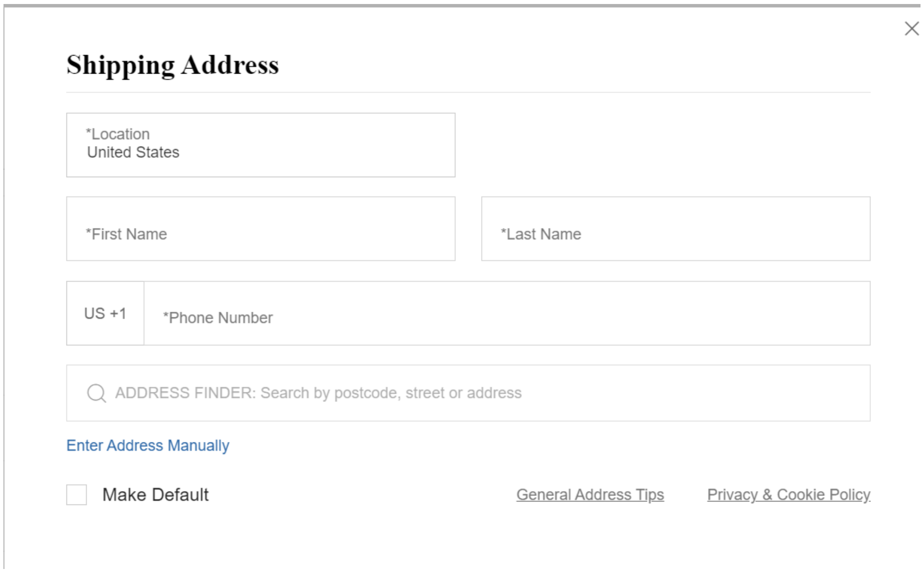
Related FWBS



Data Flow Chart for FWBS5.2 Offer “Buy” or “Bid” button

No	Name	Description	Input	Input Type	Expected Output
FWBS5.2	Offer “Buy” or “Bid” button	This button will help user to go to auction interface	User will click “Buy” or “Bid” button	Button	User gets asked “new user or returning”
FWBS5.2.1	Enter shipment information	This will help user to choose old or new shipping information	User clicks yes or no	Button	If “no” see input for FWBS5.2.2
FWBS5.2.2	Offer “Old Address” button	With this button, user can choose their old shipping address	User will click enter shipment info	Button	See SS20
FWBS5.2.1.1	Enter payment information	By clicking this button, an interface for shipping information will be shown	User will click payment info	Button	See SS21
FWBS5.2.1.2	Offer “Old Payment” button	By clicking this button, an interface for shipping information will be shown	User will click old payment button	Button	User will see “pay” button

FWBS5.2.1.1.1	Get paid	This link will help user to pay for purchase	User will click on pay	Link	Payment will be done



Shipping Address

*Location
United States

*First Name

*Last Name

US +1 | *Phone Number

ADDRESS FINDER: Search by postcode, street or address

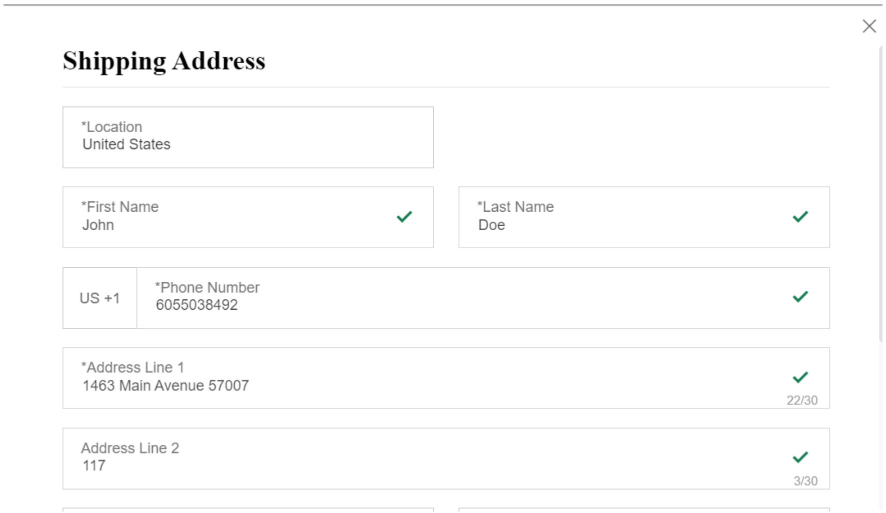
Enter Address Manually

□ Make Default

General Address Tips

Privacy & Cookie Policy

Pic: SS20



Shipping Address

*Location
United States

*First Name
John ✓

*Last Name
Doe ✓

US +1 | *Phone Number
6055038492 ✓

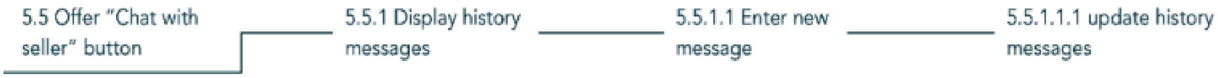
*Address Line 1
1463 Main Avenue 57007 ✓
22/30

Address Line 2
117 ✓
3/30

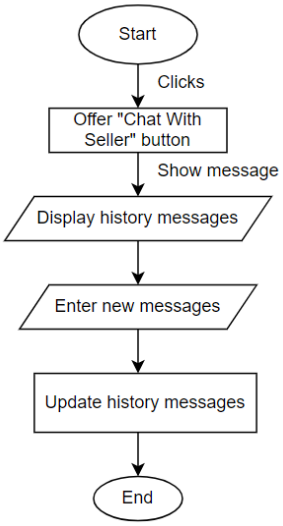
Pic: SS21

FWBS5.5 Offer “Chat with Seller” button

This will show the user messaging interface.


5.5 Offer “Chat with seller” button
5.5.1 Display history messages
5.5.1.1 Enter new message
5.5.1.1.1 update history messages

Related FWBS



Data Flow Chart for FWBS5.5 Offer “Chat with Seller” button

No	Name	Description	Input	Input Type	Output

FWBS5.5	Offer “Chat with Seller” button	By clicking this button, user will be able to chat with the seller	User clicks on “Chat with Seller” button	Button	Message history will be displayed.
FWBS5.5.1.1	Enter new message	This will allow user to enter new message	Enter new message button	Button	User will be able to type new message.

Meeting log

Meeting 9/27 1:00PM

Goal: template checking with client

Member: Linsong, Professor Shin

Result: No check box for every request but one page of acceptance, need to add Data flowing chart.

Meeting 10/1 4:00PM

Goal: RD work split detail specification

Member: ALL

Result: Fatoumata: Intro and FWBS 5 and Gantt chart from appendix

Linsong: Meeting Log, General, FWBS 2,3, and appendix, system overview from intro

Habiba: Non-Functional Requirement, FWBS 1

Fatoumata and Habiba Work together: FWBS 4.

Due 10/2 3:00PM

Change of Due date since deadline extended. New due date is 10/6 3:00PM

Meeting 10/11 1:30PM

Goal: DD discuss

Member: ALL

Result: DFD:

Linsong: 2, 3

Fatou: 4,5

Habiba: 1,4

Meeting 10/18 1:30PM

Goal: DD discuss

Member: ALL

Result: Linsong: intro, Appendix, combine

Fatou: Design Legend, Top Level Design

Habiba: Hardware/ Software, Design Priority

Meeting 10/21 10:50 AM

Goal: template check and discuss with client.

Member: ALL, prof.Shin

Result: For Medium level design, Copy Functional requests from RD and modify a little

For Top Level design, is to show connection between all clusters

For DPT, replace rank with weight, D1/D2 with Example of D1/D2 and a description of each

for template of DFD, if output do not go to Data base, point to withe space

Meeting 10/20 1:30 PM

Goal: DD discuss

Member: ALL

Result: Linsong: intelligent explain, DFD 5

Fatou: DFD 3

Project Acceptance Signatures for Client and Developer

Company ______________________________ **City** **____**

State __________________________________ **Postal Code** **_________________________________**

Phone _________________________________

Acceptance of Agreement:

I certify that I have received all project deliverables from Qnet Company and that all deliverables have been demonstrated, delivered or otherwise completed to my satisfaction. I further certify and release Qnet Company, from any further obligation, support or duty concerning this project, including any and all continued support services. I understand that by signing this agreement, I release Qnet Company from any and all liability, tort or claim concerning this project, its files, source code, object code, programming or other materials, whether printed or digital, provided to the client. I further understand that all files provided to me under this or any other agreement are to be used at my sole risk and responsibility and all files should be tested prior to publication or any mass distribution in any format.

Qnet Company is not responsible for any results obtained from the use of any software, materials, presentation or products provided to the client under this agreement.

Client's Signature ____

Developer Signature ____

References

Minaee, Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2022). Deep Learning--based Text Classification: A Comprehensive Review. ACM Computing Surveys, 54(3), 1–40. https://doi.org/10.1145/3439726

Wang, Xu, B., Xu, J., Tian, G., Liu, C.-L., & Hao, H. (2016). Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. Neurocomputing (Amsterdam), 174, 806–814. https://doi.org/10.1016/j.neucom.2015.09.096

https://www.eventhelix.com/fault-handling/techniques/

Bowers, M. (2020, July 28). Design constraints are not restraints – they stoke creativity. Toptal Design Blog. Retrieved October 24, 2022, from https://www.toptal.com/designers/ui/design-constraints#:~:text=Design%20constraints%20are%20limits%20placed,is%20a%20designer's%20skill%20level.

Certificate/Diploma in IT - oxford, Cambridge and RSA Examinations. (n.d.). Retrieved October 24, 2022, from https://www.ocr.org.uk/Images/560498-system-development-requirements-and-constraints.pdf

Editor. (2019, December 9). *Technical documentation in software development: Types, best practices, and Tools*. AltexSoft. Retrieved October 24, 2022, from https://www.altexsoft.com/blog/business/technical-documentation-in-software-development-types-best-practices-and-tools/