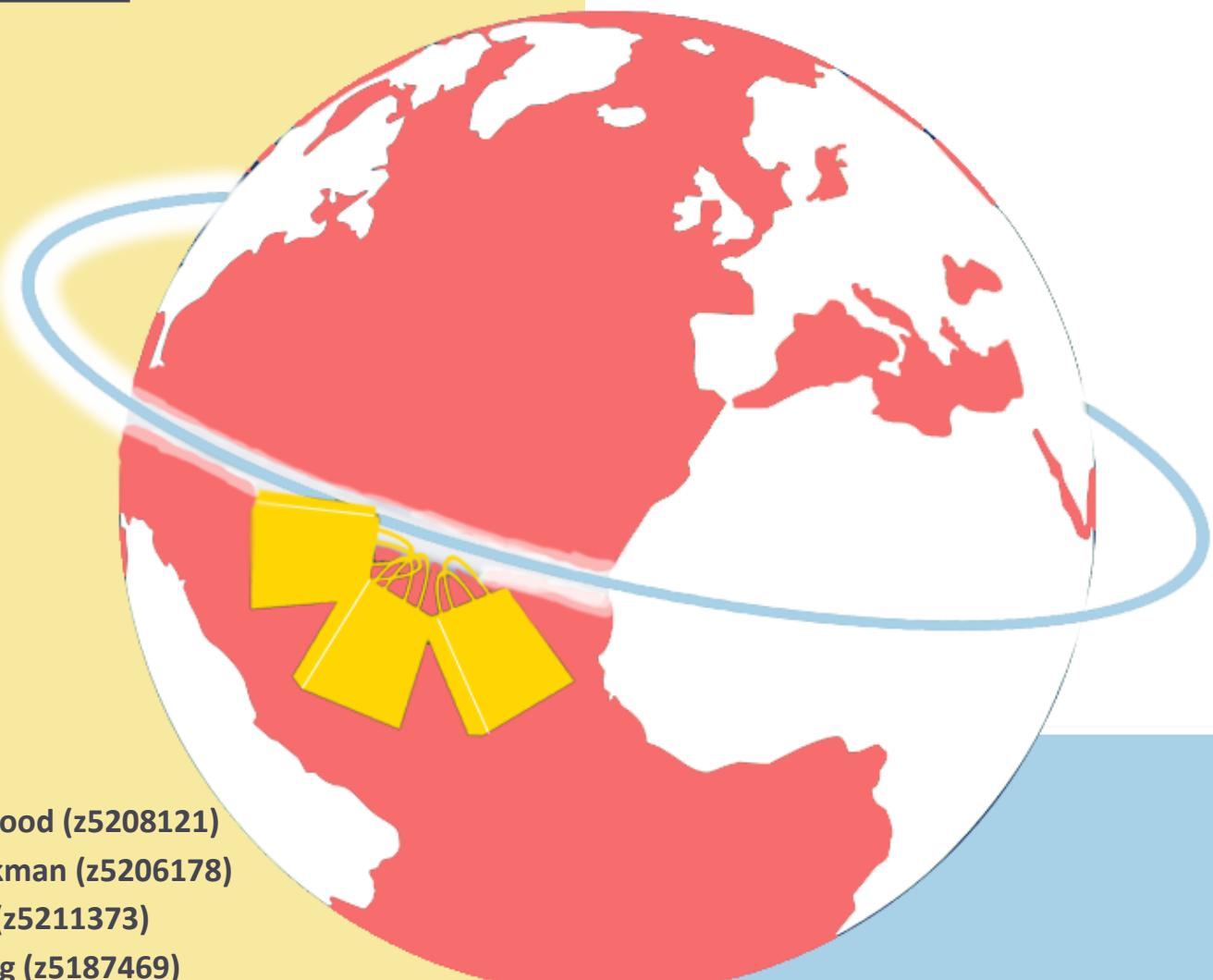


FINAL REPORT

Globeshoppers Website



Lavanya Sood (z5208121)
Sarah Oakman (z5206178)
Yin Huey (z5211373)
Yiyun Yang (z5187469)
Xiaorui Li (z5206613)

Introduction

PURPOSE OF GLOBESHOPPERS

In today's context where getting items overseas might seem easier than before due to logistical advancements, one could not escape from paying expensive international shipping fees just to obtain items abroad. Some items are also exclusive overseas hence it is hard to acquire unless done by travelling. While travelling, traveller's extra luggage space often gets wasted too. There is also no website that shows the most sought-after items in different countries.

Thus, we decided to create a peer to peer marketplace that matches locals who want to buy things from overseas, with travelers who have extra luggage space and are keen to earn some money.

Globeshoppers relies on its users for its functioning as Australian consumers post products they want and people travelling overseas can visit the website and offer to buy these products. Similar to gumtree, in order for the consumer to receive their goods, a messaging system is in place for the consumer and the traveller to organise a mechanism to exchange payments and products.

PROBLEMS GLOBESHOPPERS ADDRESSES

1. It is hard to get items from overseas without travelling
2. International shipping fees for items overseas are expensive
3. The extra luggage space of travellers often gets wasted
4. There is no website that shows the most sought-after items in different countries

List of Globeshoppers features

- **Consistent UI Design**
 - Base.html is used for headers and footers
 - Navigation bar is present at the top of each page with the logo of the website. This navigation bar allows users to access the other pages on our website
 - Footer is presented at the bottom of each page to navigate the users to the web pages that can provide them with information about how our website works
 - Consistent colour scheme
 - Readable Font
- **Clear instructions for the users**
 - These are present on the welcome page as well as the footer in order make the implementation of the website easier for the users
- **Login/ Logout**
 - The users have the ability to login into the website in order to request products and offer help to other users
 - Users are allowed to view products on the main page however in order to post a request or add products the users will be redirect to the login page in order to access these features (Refer to **User Story 1c**)
 - Refer to **User Story 1a**
- **Register a new account**
 - Gives users the ability to create an account in order to purchase products from our website
 - Refer to **User Story 1b**
- **Add a product**
 - Users have the ability to add their own product which they want to get from overseas
 - Refer to **User Story 2a**
- **Request an existing product**
 - The users have the ability to add their own requests to an existing product so that other users can get it for them
 - Refer to **User Story 2b**
- **See own purchase history**
 - Users have the ability to see their own purchases and the products that they have requested currently as well as previously
 - Refer to **User Story 3a**

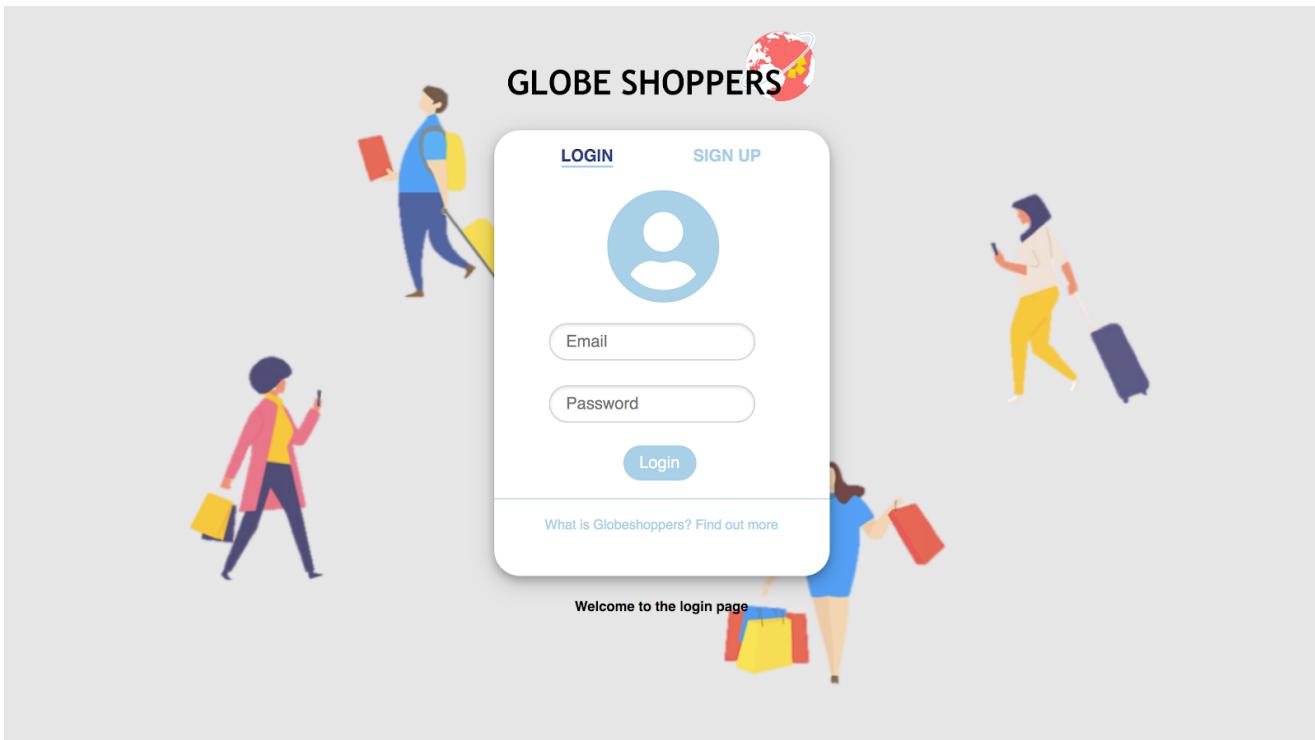
- Users have the ability to delete product requests if they change their mind about products with no consequences
 - Refer to **User Story 4a**
 - However, if a user deletes an offer that they have already accepted they are given a penalty for backing out of their deal
 - Refer to **User Story 4b**
- **Search function**
 - Users can search through the website for their desired product
 - Refer to **User Story 5a**
- **Look for items through category**
 - Users can browse the categories given on the website in order to find similar products to the one they are interested in
 - Refer to **User Story 6a**
 - The users also have the ability to sort the categories according to price as well as the country they will be available in
 - Refer to **User Story 6b**
- **View a product**
 - Ability to view the details of the products which have been uploaded by other users
 - Refer to **User Story 7a**
- **Offer help to other users**
 - Users can provide help to other users in order to bring an item from overseas
 - Refer to **User Story 8a**
- **Receive help from other users**
 - Users have the ability to accept or reject the offer given to them by another user
 - Refer to **User Story 9a**
- **Provide rating to other users**
 - Users are able to provide feedback and give rating to people who they have been in contact with
 - Refer to **User Story 10a**
- **View other profiles**
 - Users have the ability to view other people's profiles in order to learn if the other user is reliable
 - Refer to **User Story 11a**
- **Access own profile**
 - Users can view their own profile on which various details such as purchase history, trip history and reviews from other users are given

- Refer to **User Story 11b**
- Users also have the ability to edit their own profile details
 - Refer to **User Story 11c**
- Users can add upcoming trips on their profile page for other users to see.
 - Refer to **User Story 14a**
- Users can also delete the trip information from their profile page
 - Refer to **User Story 14b**
- **Send messages**
 - Users are able to contact other users by sending them messages in order to clarify details about the purchases
 - Refer to **User Story 13a**
- **Inbox**
 - The inbox provides the user with all the previous message history between them and another user
 - Refer to **User Story 12a**
 - Users are also able to reply to the other user through this inbox
 - Refer to **User Story 12b**

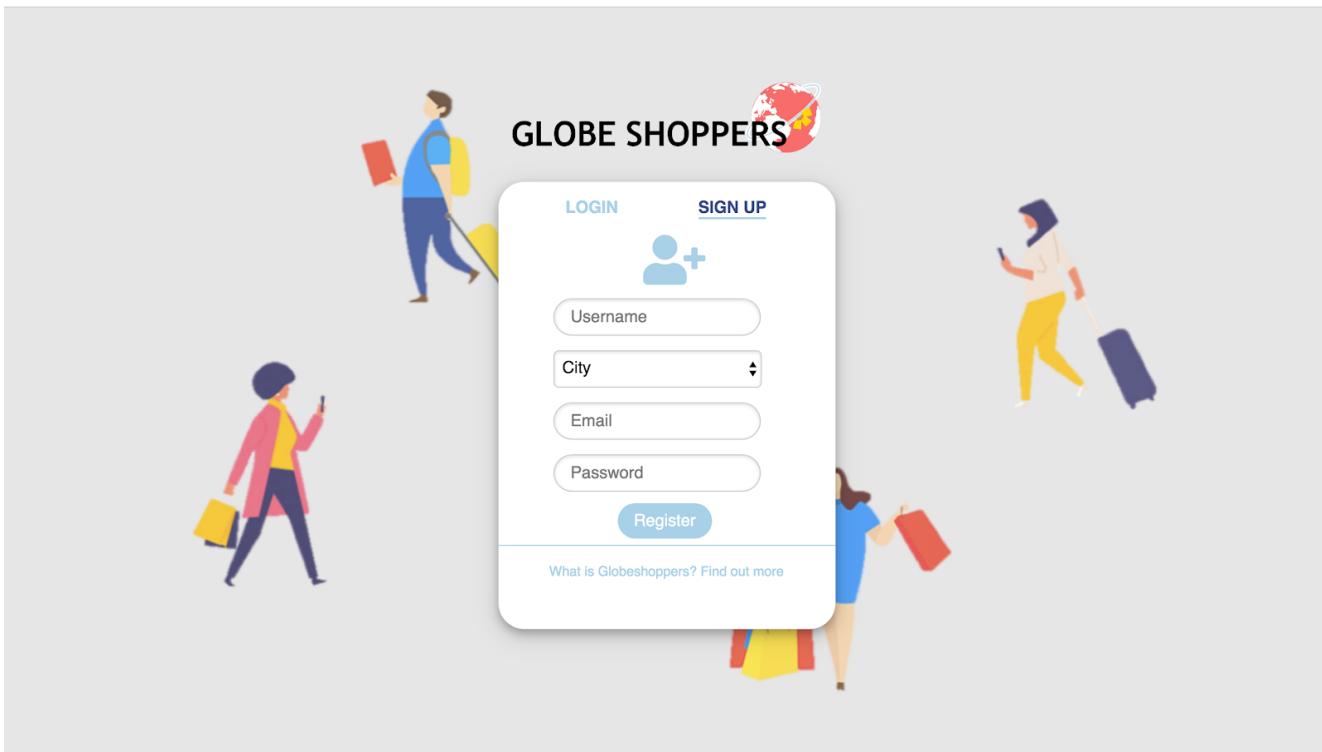
Updated User Stories

Requirement 1: Users are able to login or register a new account

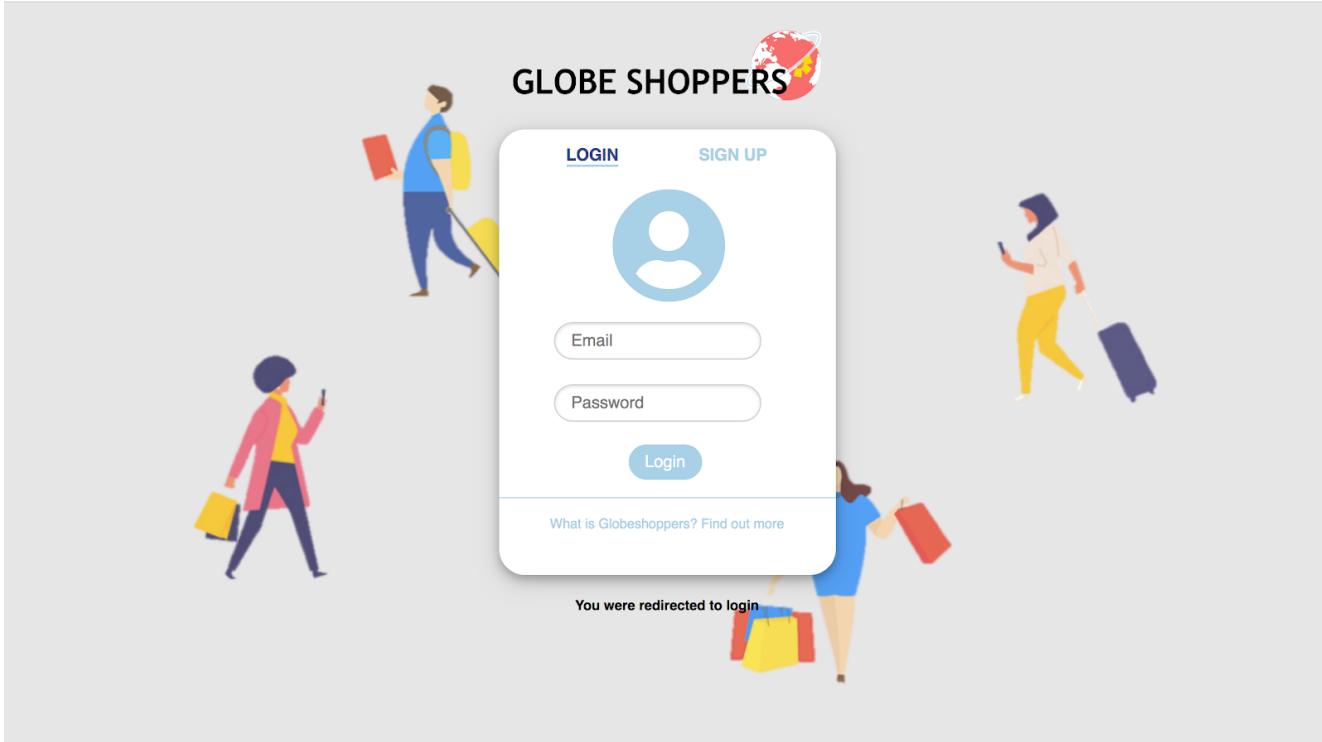
ID	US1a
Feature	Login to my existing account
Description	<p>As a user, I want to login to my existing account So that I can access my previous activity on the website while keeping my information secure</p>
Given	I am on any page under the website
When	I select the user icon at the top bar
Then	I am redirected to a login page displaying two textboxes for email and password and button for 'login' and a button labelled 'Sign Up'
When	I type in my email and password and click 'Login'
Then	I am redirected back to the page I started on before login
Estimate	3 (1SP = 2.5 hrs)
Priority	Moderate



ID	US1b
Feature	Register a new account
Description	<p>As a new user, I want to register a new account So that I can access all of the website's features and create a profile which stores my information</p>
<p>Given I am on the login page When I click "Sign Up" button Then I am redirected to a sign-up page displaying text-boxes for username, city (dropdown), email and password (which is hidden) with a "register" button below When I type in my new account details and click 'Register' button Then I am redirected back to the page I started on before sign-up </p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Moderate

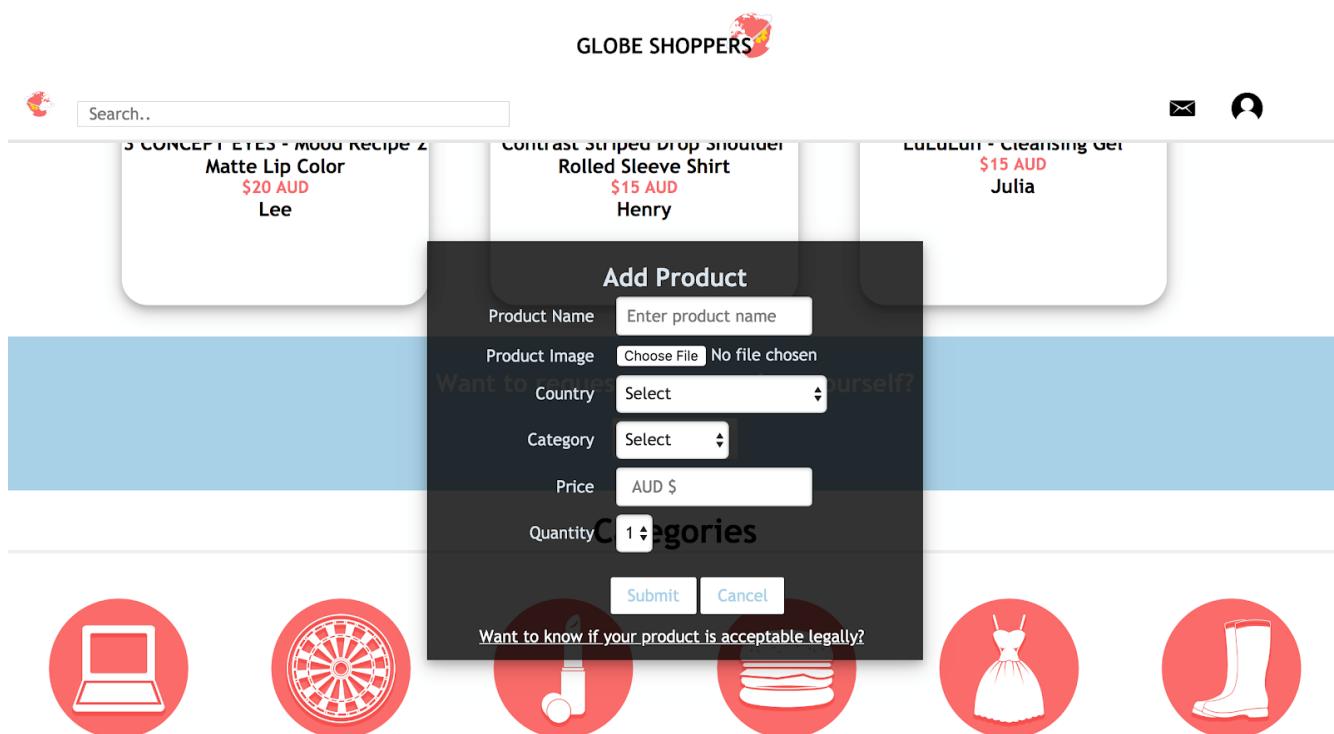


ID	US1c
Feature	Redirect to login page
Description	<p>As a user, I want to login first So that I can access features/pages that is only for users that have logged in</p>
	<p>Given I want to request item/post item/message users (pages that can only be accessed after logging in)</p> <p>When I click those action buttons</p> <p>Then I am redirected to the login page with email and password textboxes, 'Login' and 'Sign up' selection</p> <p>When I type in my email and password and click 'Login'</p> <p>Then I am redirected back to the page where I left</p>
Estimate	2 (1SP = 2.5 hrs)
Priority	Low

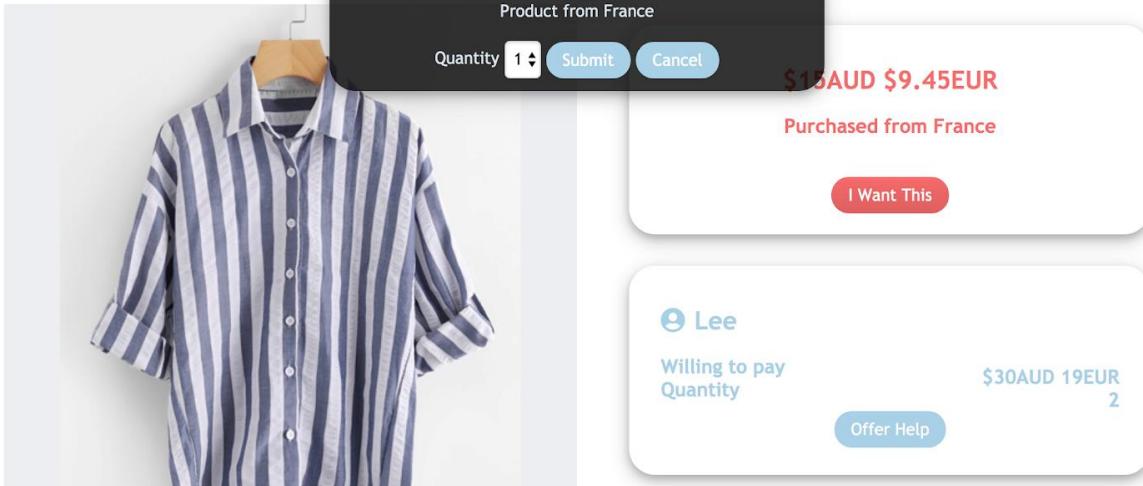


Requirement 2: User wants to request items from overseas

ID	US2a
Feature	Post a request for an item (login required)
Description	<p>As a user, I want to post a request for an item So that other users can view and accept my request if they are able to help</p>
<p>Given I want to post a request for an item from overseas When I fill up the details of my item (name, country, price, category, quantity) at the “Add Product” pop-up form, upload a picture and click the ‘Submit’ button to post my request Then pop up confirmation will appear</p>	
Estimate	4 (1SP = 2.5 hrs)
Priority	High



ID	US2b
Feature	Add a new request onto existing item requests (login required)
Description	<p>As a user, I want to add a new request onto existing item requested by another user, So that I can obtain the same item requested by others</p>
<p>Given I am on the item's description page When I am also interested with the item requested by another user and click the "I want this" button Then a pop-up appears where I can select quantity from a drop-down When I click 'submit' button Then my request will appear on the item's description page and my profile page When I click 'cancel' Then the pop-up will disappear and I will be back on the item's description page </p>	
Estimate	4 (1SP = 2.5 hrs)
Priority	Moderate



The screenshot shows a product page for a "Contrast Striped Drop Shoulder Rolled Sleeve Shirt". The shirt is displayed on a hanger. A black pop-up window is overlaid on the page, containing the following information:

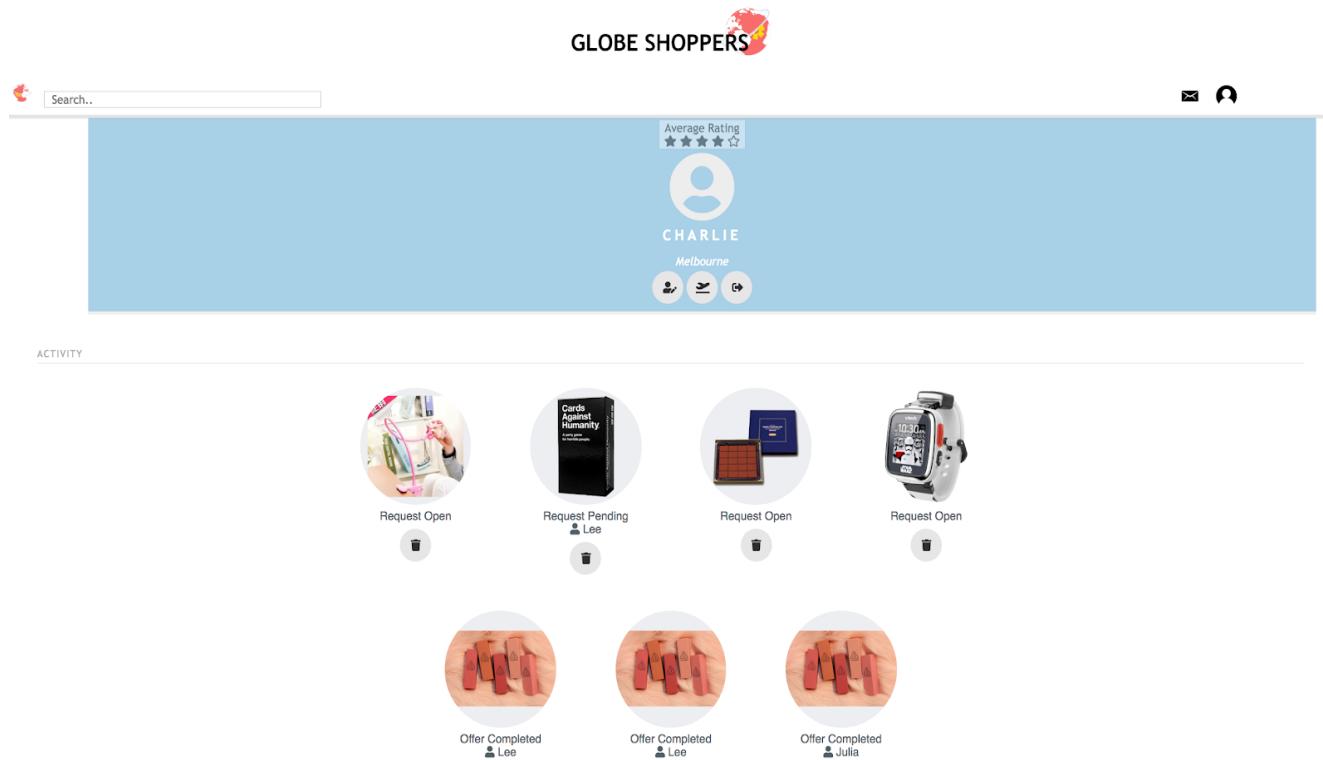
- Product name: Contrast Striped Drop Shoulder Rolled Sleeve Shirt
- Price: \$15 AUD each
- Origin: Product from France
- Quantity selector: Set to 1
- Buttons: Submit and Cancel
- Price again: \$15AUD \$9.45EUR
- Text: Purchased from France
- Red button: I Want This

Below the pop-up, there is a user interaction section:

- User: Lee
- Willing to pay: \$30AUD 19EUR
- Quantity: 2
- Offer Help button

Requirement 3: User wants to view requested items

ID	US3a
Feature	View list of own requested items (login required)
Description	<p>As a user, I want to view a list of my requested items, So that I can know the status of the items requested</p>
Estimate	3 (1SP = 2.5 hrs)
Priority	Moderate



The screenshot shows a user profile for 'CHARLIE' from Melbourne. The profile includes a placeholder image, an average rating of 3 stars, and three activity items:

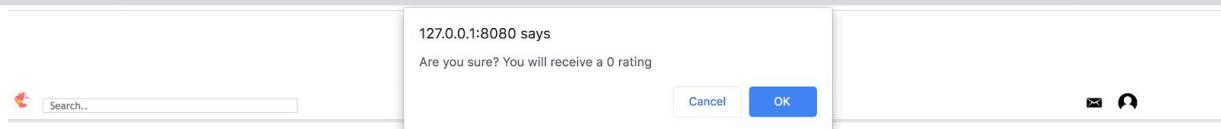
- Request Open:** A circular icon showing a person holding a pink item, labeled 'Request Open'.
- Request Pending:** A circular icon showing a black box labeled 'Cards Against Humanity', labeled 'Request Pending'.
- Request Open:** A circular icon showing a blue and orange box, labeled 'Request Open'.
- Offer Completed:** Two circular icons showing red lipsticks, both labeled 'Offer Completed'.

Requirement 4: User wants to delete requested items

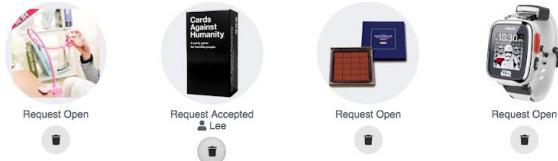
ID	US4a
Feature	Delete pending requests (login required)
Description	As a user, I want to delete my request for items, So that I don't have to spend money on something I no longer desired
Given	I am on any page in the website
When	I click the user icon at the top bar of the website
Then	I am redirected to my profile with a list of my requests and offers with their status
When	I delete my request with "Open" or "Pending" status by clicking the bin button below
Then	a pop up will appear with "Ok" and "Cancel" button requesting deletion confirmation by the user
Estimate	3 (1SP = 2.5 hrs)
Priority	Moderate

The screenshot shows a user profile page for 'CHARLIE' from 'Melbourne'. The profile picture is a placeholder. Below the name, there are three circular icons: a person, a gift box, and a gear. The page displays a grid of items categorized into 'REQUESTS' and 'OFFERS'. In the REQUESTS section, there are four items: 'Request Open' (image of a hand holding a pink object), 'Request Pending' (image of a 'Cards Against Humanity' box), 'Request Open' (image of a small blue box), and 'Request Open' (image of a smartwatch). Each request has a trash can icon below it. In the OFFERS section, there are three items: 'Offer Completed' (image of several red lipsticks), 'Offer Completed' (image of several red lipsticks), and 'Offer Completed' (image of several red lipsticks). Each offer also has a trash can icon below it. At the top of the page, there is a search bar and a user icon. A modal dialog box is centered over the page, containing the text '127.0.0.1:8080 says' and 'Are you sure?'. It has 'Cancel' and 'OK' buttons.

ID	US4b
Feature	Delete accepted requests (login required)
Description	<p>As a user, I want to delete my item request accepted by travellers, So that I don't have to spend money on something I no longer desired</p>
<p>Given I am on any page in the website When I click the user icon at the top bar of the website Then I am redirected to my profile with a list of my requests and offers with their status When I delete my request with "Accepted" status by clicking the bin button below Then a pop up will appear with a penalty warning, "Ok" and "Cancel" button requesting deletion confirmation by the user And a penalty of 0-star rating and automated review statement will appear at my profile regarding my deletion </p>	
Estimate	3 (1SP = 2.5 hrs)
Priority	Low



ACTIVITY



REVIEWS

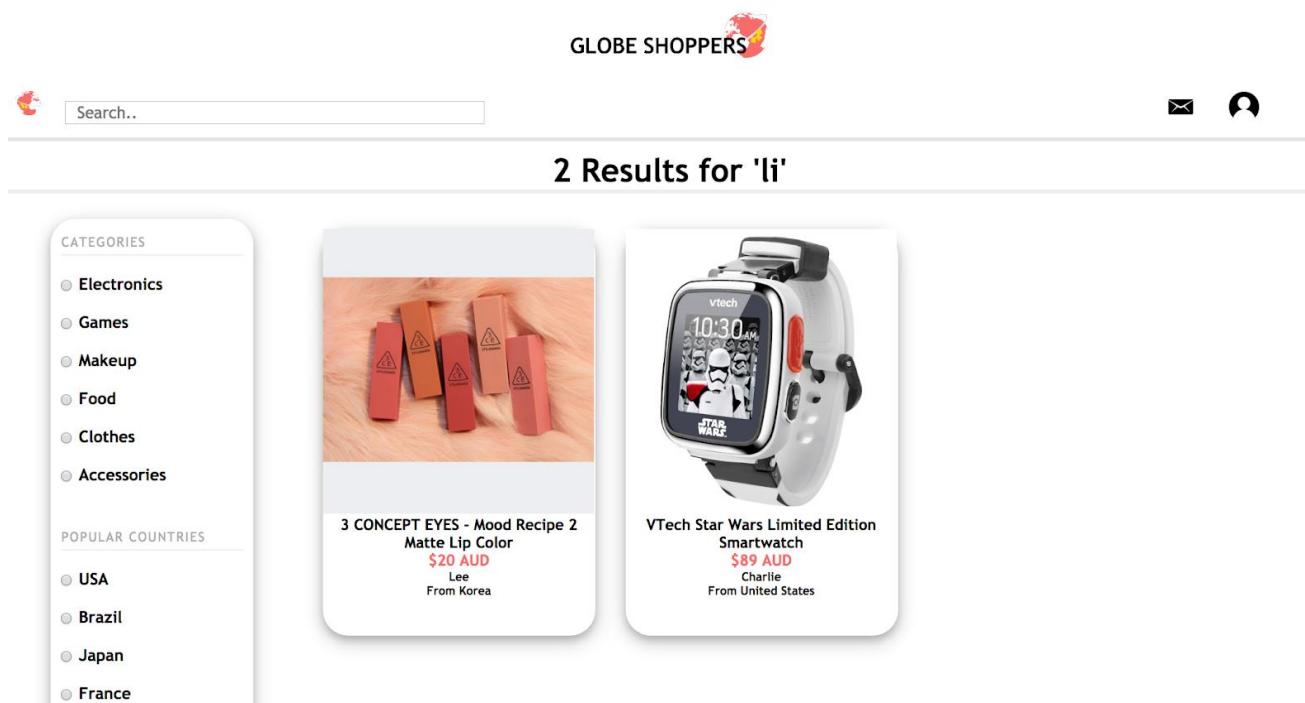
Tom
Excellent service, has good recommendations

Lee

GlobeShoppers
This user deleted their accepted request and a 0 rating was given.

Requirement 5: Users want to find specific items that they want

ID	US5a
Feature	Search for the items that have been requested
Description	<p>As a user, I want to search for the items that have been requested, So that I can add a request on the existing item or offer help</p>
Given	I am on any page in the website
When	I fill in the search bar with the item that I am interested and press “Enter”
Then	I should see all the item requests which include the word or words I entered
Estimate	5 (1SP = 2.5 hrs)
Priority	High

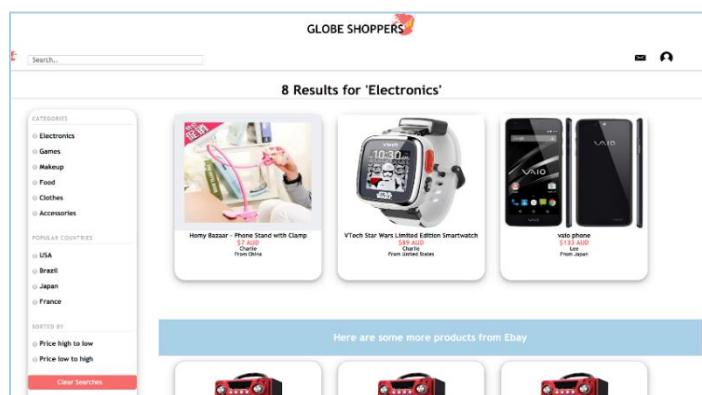
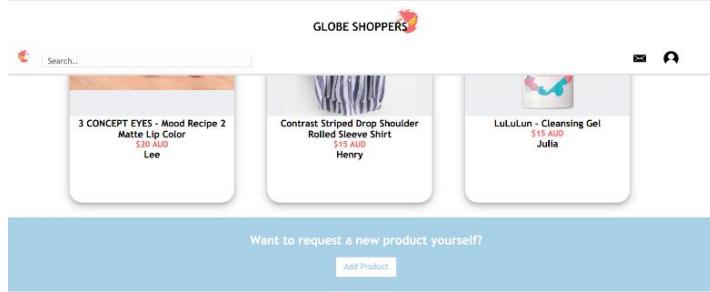


The screenshot shows a search interface for 'GLOBE SHOPPERS'. On the left, there's a sidebar with categories like Electronics, Games, Makeup, Food, Clothes, and Accessories. Below that is a list of popular countries: USA, Brazil, Japan, and France. The main area displays search results for the query 'li'. It shows two items: '3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color' by Lee from Korea, and 'VTECH Star Wars Limited Edition Smartwatch' by Charlie from United States.

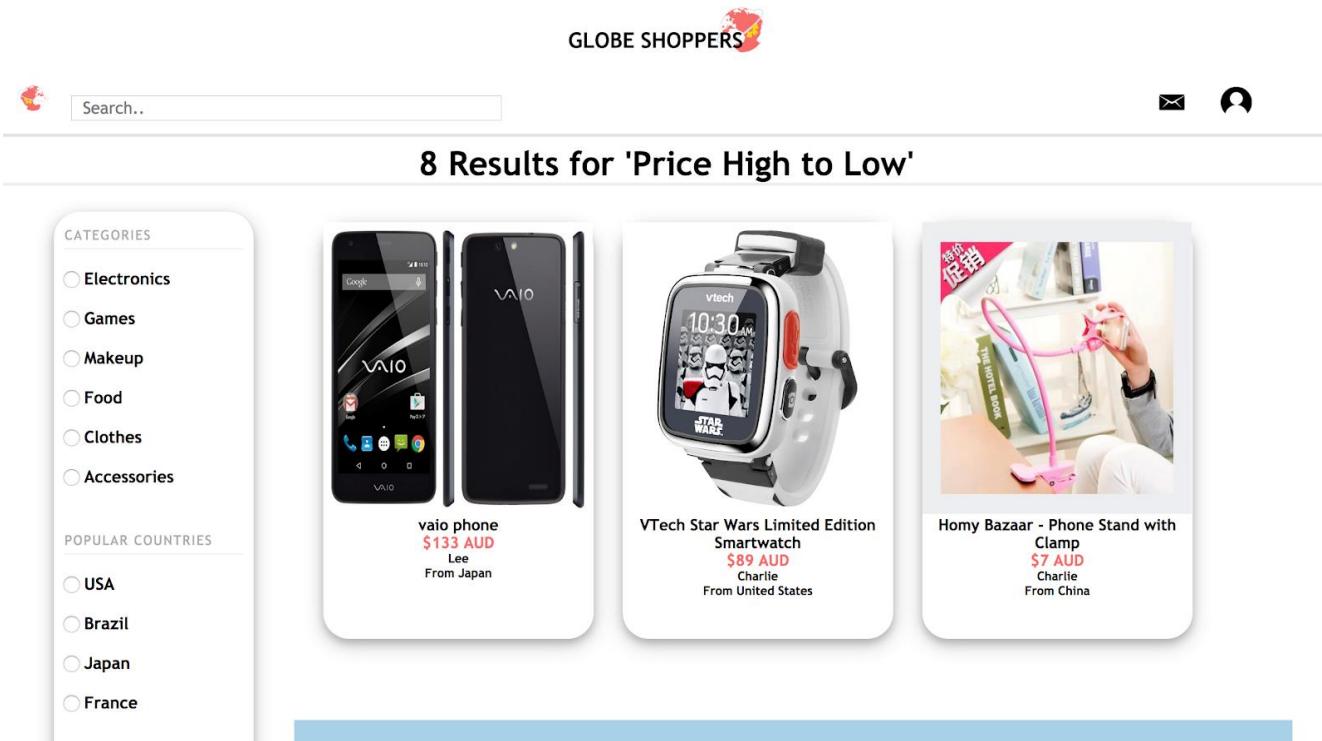
Category	Product Name	Brand	Price	Origin
3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color	Lee	\$20 AUD	From Korea	
VTECH Star Wars Limited Edition Smartwatch	Charlie	\$89 AUD	From United States	

Requirement 6: Users want to view specific items by category or sorted according to price

ID	US6a
Feature	View requested items by other users according to category
Description	<p>As a user, I want to view items according to their category So that I can easily view all products I am interested in and view all similar products on one page</p>
<p>Given I am on the “Home” page When I press a category icon (“games”, “electronics”, “accessories”, etc) Then I will be redirected to the category page which shows all of the items within the matching category </p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Low



ID	US6b
Feature	View items sorted according to price
Description	<p>As a user, I want to view items sorted according to price So that I can easily look for cheapest or most expensive item</p>
<p>Given I am on the “Search Results” page When I press “Price High to Low” or “Price Low to High” button at the side panel Then I will see a list of items sorted according to price on the same page</p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Low

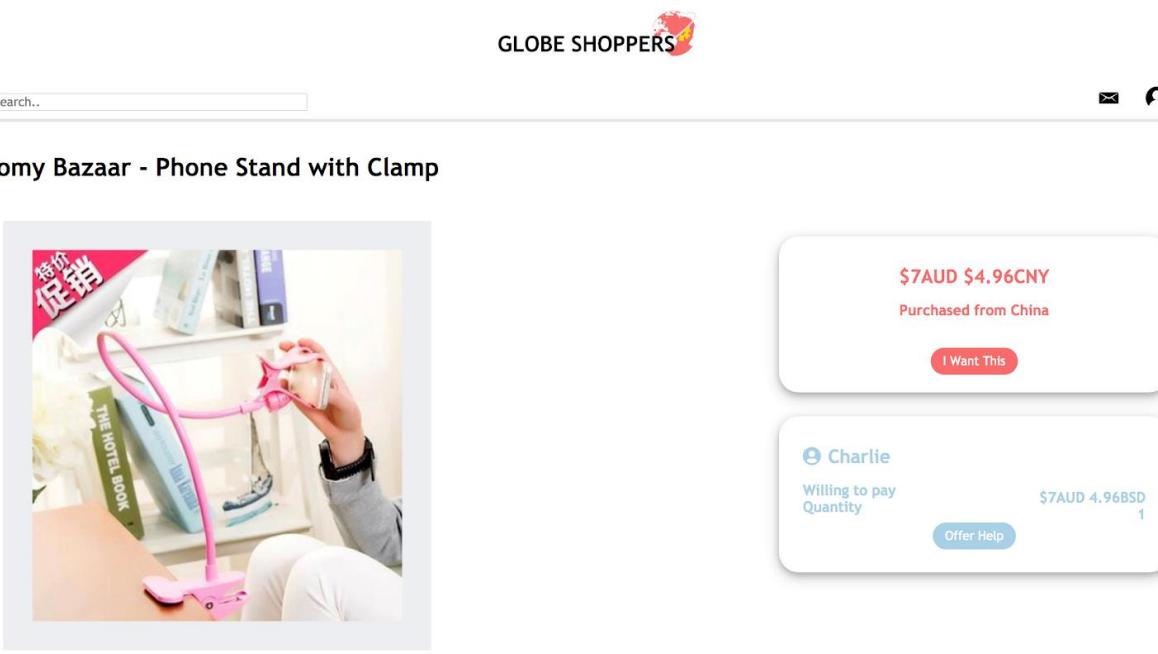


The screenshot shows a search results page for 'Price High to Low'. The interface includes a sidebar with categories like Electronics, Games, Makeup, Food, Clothes, and Accessories, and popular countries like USA, Brazil, Japan, and France. The main content area displays three products: a VAIO phone, a VTech Star Wars Limited Edition Smartwatch, and a Homy Bazaar Phone Stand with Clamp.

- VAIO phone**: \$133 AUD, Lee, From Japan
- VTech Star Wars Limited Edition Smartwatch**: \$89 AUD, Charlie, From United States
- Homy Bazaar - Phone Stand with Clamp**: \$7 AUD, Charlie, From China

Requirement 7: User wants to view items requested by other users

ID	US7a
Feature	View item requests made by another user
Description	<p>As a user, I want to view an item request made by another user, So that I can add my request if I'm interested or want to offer help to other users</p>
<p>Given I am on the “Home” or “Search Results” page with a list of items requested by other users</p> <p>When I click on the item to see more of a particular item request</p> <p>Then I should be on the item’s description page with the item’s details such as name, item image, country, quantity, price offered and user who requested it</p>	
Estimate	3 (1SP = 2.5 hrs)
Priority	High



The screenshot shows a product listing for a "Hommy Bazaar - Phone Stand with Clamp". The product image is a pink flexible phone stand being held over a desk. To the right of the image, there is a summary box containing the price (\$7AUD \$4.96CNY) and the fact that it was purchased from China. Below this is a button labeled "I Want This". Further down, a user named Charlie is shown with their profile picture, indicating they are willing to pay and have offered help. The overall interface is clean and modern.

Requirement 8: User wants to offer help by accepting purchase requests by other users

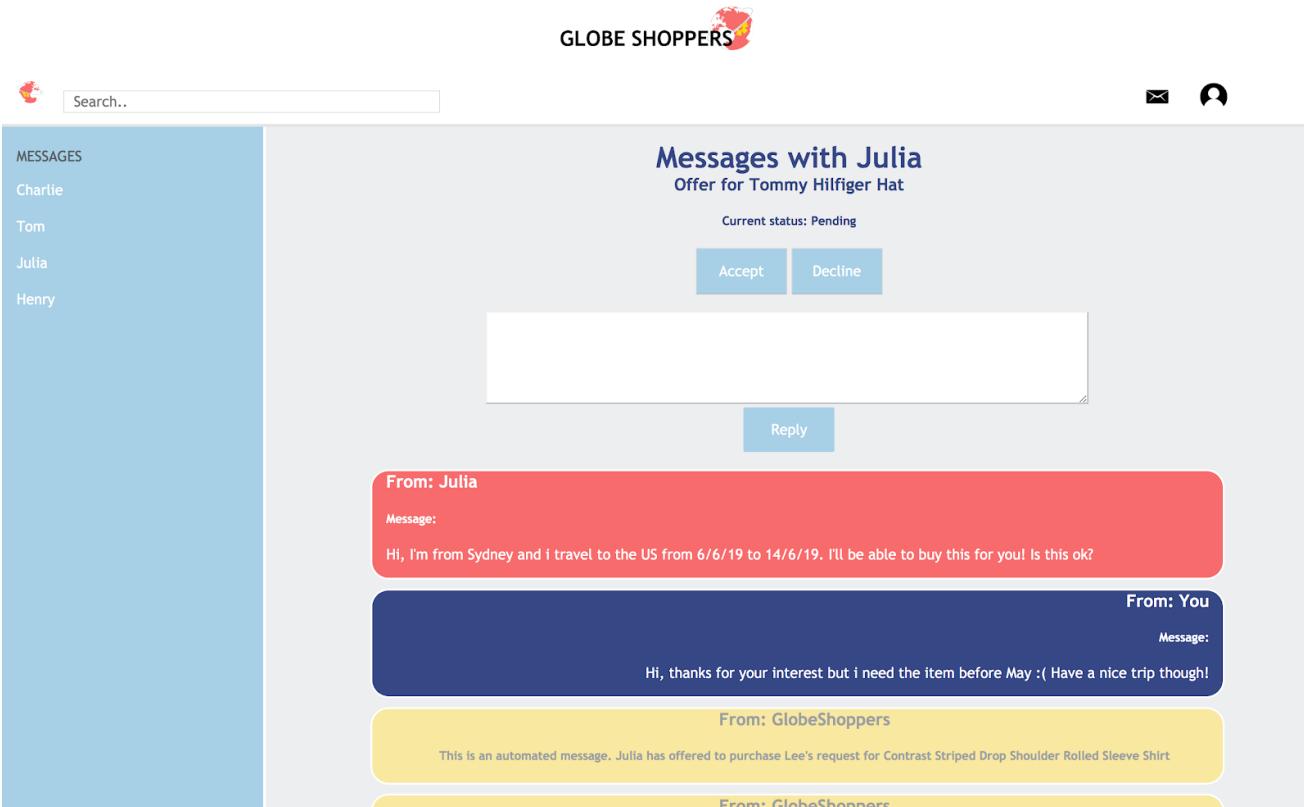
ID	US8a
Feature	Accept item requests (login required)
Description	As a user, I want to accept an item request from another user, So that I can make a profit with my extra luggage space
Given I am on the requested item's description page When I click the 'offer help' button at the request column Then a verification message will appear with the item's details reiterated with a message column, submit and cancel button When I click the submit button to confirm offer Then I am redirected back to the item's description page and an offer message is sent to the customer's inbox	
Estimate	4 (1SP = 2.5 hrs)
Priority	High

The screenshot shows a product page for a "Contrast Striped Drop Shoulder Rolled Sleeve Shirt". The shirt is displayed on a hanger. A modal window is overlaid on the page, containing the following information:

- Product Details:** Contrast Striped Drop Shoulder Rolled Sleeve Shirt, \$15 AUD each, Quantity: 3, Product from France.
- Message Input:** Send Message to Henry (button) and Enter message... (text input field).
- Buttons:** Submit and Cancel.
- Offer Help Modal:** A separate modal window for Henry, showing he is willing to pay \$45AUD 28.35EUR for 3 units. It includes an Offer Help button.

Requirement 9: User wants to accept offers by other users

ID	US9a
Feature	Accept offers by other users (login required)
Description	<p>As a user, I want to be able to accept offers by the other users, So that I can have the choice to decide which traveller I would like to accept for help</p>
<p>Given I am at any page on the website When click the mail icon at the top bar of the website Then I will be redirected to the “Inbox” page with messages and offers by users(travellers) When I click the “Accept” at the top of the inbox message page to accept the traveller’s offer Then the status of my item in my profile page will change from “Pending” to “Accepted”</p>	
Estimate	4 (1SP = 2.5 hrs)
Priority	High



The screenshot shows the 'GLOBE SHOPPERS' app interface. On the left is a sidebar with a globe icon, a search bar, and a list of messages from 'Charlie', 'Tom', 'Julia', and 'Henry'. The main screen displays a message thread titled 'Messages with Julia' about an 'Offer for Tommy Hilfiger Hat'. The status is 'Pending'. There are 'Accept' and 'Decline' buttons. Below the buttons is a large empty message input field with a 'Reply' button. The message history shows a message from 'Julia' asking if the offer is okay, a response from 'You' saying they need it by May, and automated messages from 'GlobeShoppers' confirming the offer and providing item details.

GLOBE SHOPPERS

MESSAGES

- Charlie
- Tom
- Julia
- Henry

Messages with Julia
Offer for Tommy Hilfiger Hat

Current status: Pending

Accept Decline

Reply

From: Julia
Message:
Hi, I'm from Sydney and i travel to the US from 6/6/19 to 14/6/19. I'll be able to buy this for you! Is this ok?

From: You
Message:
Hi, thanks for your interest but i need the item before May :(Have a nice trip though!

From: GlobeShoppers
This is an automated message. Julia has offered to purchase Lee's request for Contrast Striped Drop Shoulder Rolled Sleeve Shirt

From: GlobeShoppers

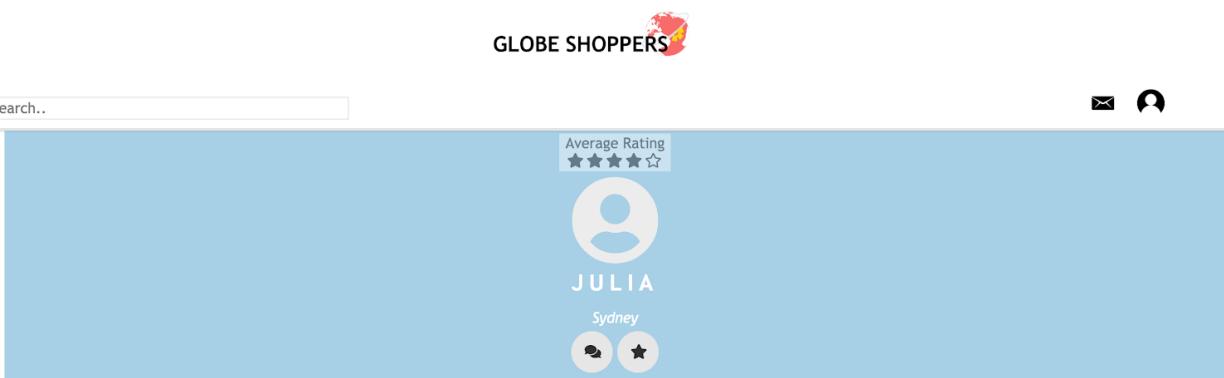
Requirement 10: Users need to be able to review each other to provide feedback

ID	US10a
Feature	Write reviews for other users (login required)
Description	<p>As a user, I want to write a review for another user after completing a purchase with them So that other users can know their reliability</p>
<p>Given I am on another user's profile page When I click the button with the star icon on their profile to give them a review Then I see the popup box with ratings and comment section When I select the star ratings, type in my feedback And I press star icon Then I should see my review and updated ratings under the user's profile page </p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Low

The screenshot shows a user profile for "CHARLIE" from Melbourne. The profile includes a placeholder for an average rating and a large circular user icon. Below the name, there is a button labeled "ADD REVIEW" with a star rating scale and a text input field for "Review". A blue airplane icon with the word "KOREA" is positioned below the profile. At the bottom of the screen, there is a "REVIEWS" section and two user profile cards for "Tom" and "Lee".

Requirement 11: User need to be able to view, edit their own profile and view other users' profiles

ID	US11a
Feature	View another users' profile
Description	<p>As a user, I want to view other users' profiles So that I can learn more about the user I am selling to/buying from.</p>
<p>Given I am on any page that displays another user's name such as an item's information page or the activity section in my profile page When I select the other user's name Then I am directed to their profile page with their bio, city location, future trip information, ratings and reviews from other users, buttons for messaging and review </p>	
Estimate	3 (1SP = 2.5 hrs)
Priority	Moderate



The screenshot shows a user profile for 'JULIA'. At the top, there is a red globe icon and the text 'GLOBE SHOPPERS'. Below the globe is a search bar with the placeholder 'Search..'. On the right side of the header are a mail icon and a user icon. The main profile area has a light blue background. It features a circular user icon, the name 'JULIA' in bold capital letters, and the city 'Sydney' below it. To the right of the name are two small circular icons: one with a speech bubble and another with a star. Above the profile picture, there is a 'Average Rating' section with five yellow stars and a small grey star. At the bottom of the profile area, there is a large blue airplane icon with the word 'USA' underneath it. Below the profile area, there is a horizontal line with the word 'TRIPS' above it. Further down, there is a section titled 'REVIEWS' with two entries. The first review is by 'Tom' with a rating of five yellow stars, followed by the text 'Excellent service, has good recommendations'. The second review is by 'Henry' with a rating of four yellow stars and one grey star, followed by the text 'Didn't get the item i wanted but offered an alternative which i liked'.

ID	US11b
Feature	View my own profile (login required)
Description	<p>As a user, I want to view my own profile So that I can update my information, check my information is correct and implement the feedback I receive from reviews.</p>
<p>Given I am on any page on the website When I select my profile icon in the bar at the top of the page Then I am directed to my own profile page And I am able to see reviews others have written about me, my bio, my activity with my requests and offers, future trip information and buttons for editing profile and adding trip details</p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Moderate

The screenshot shows a user profile page for 'LEE' on the GLOBE SHOPPERS platform. At the top, there's a search bar and a navigation bar with icons for envelope and profile. The profile section features a placeholder image, an average rating of 3 stars, and the name 'LEE' with 'Sydney' listed below it. The bio says 'Hi, I like food :)' and includes edit and delete buttons. Below this is the 'ACTIVITY' section, which displays four items: 'Request Accepted' from 'Charlie' (image of red lipsticks), 'Request Accepted' from 'Julia' (image of a baseball cap), 'Request Completed' from 'Charlie' (image of red lipsticks), and 'Request Completed' from 'Julia' (image of a striped shirt). Each item has a trash icon below it. The 'TRIPS' section shows three travel records: a flight from '2019-03-11 to 2019-03-17' to 'JAPAN', a flight from '2019-09-23 to 2019-07-10' to 'ITALY', and a flight from '2019-04-09 to 2019-04-25' to 'ALGERIA'. Finally, the 'REVIEWS' section shows two reviews: one from 'Charlie' giving 5 stars and saying 'Delivery was made quickly. Very responsive. Would Recommend!', and one from 'Henry' giving 4 stars and saying 'A nice customer who was polite and not pushy.' Below each review is a trash icon.

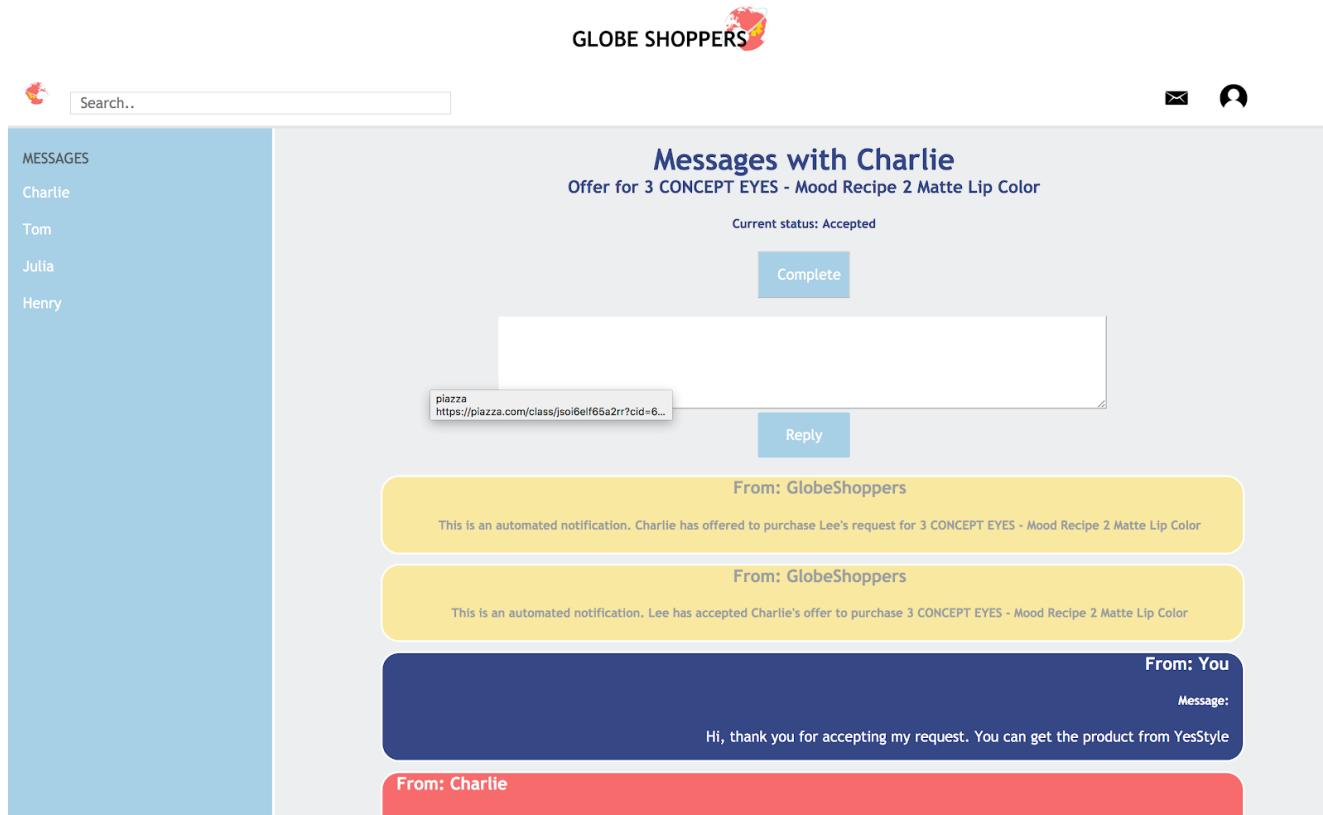
ID	US11c
Feature	Update my personal information (login required)
Description	<p>As a user, I want to update my personal information So that other users can receive the correct updated information and to help them find an appropriate user to suit their needs.</p>
<p>Given I am on my profile page When I select the edit user icon Then I can replace my profile picture And add or remove information in my 'about me' bio , And change my username, email or contact number When I click the submit icon to save these changes Then I will be redirected back to my own profile with information updated </p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Low

The screenshot shows a mobile application interface for "GLOBE SHOPPERS". At the top, there is a navigation bar with a globe icon, a search bar containing "Search..", and icons for message and user profile. Below the navigation is a user profile card. The profile card includes a placeholder for an average rating (5 stars) and a large circular placeholder for a profile picture. Below the profile picture is a dark rectangular "EDIT PROFILE" button. This button contains input fields for "Username" (Lee), "Email" (lee@gmail.com), "Password" (>Password), "City" (Sydney), and "About Me" (Hi, I am ...). There are also "Edit" and "X" buttons at the bottom of the modal. Below the profile card, there is a section titled "ACTIVITY" which displays four circular cards, each representing a user interaction:

- A card with a red lip product image labeled "Request Accepted" by "Charlie" with a trash bin icon below it.
- A card with a blue baseball cap image labeled "Request Accepted" by "Julia" with a trash bin icon below it.
- A card with a red lip product image labeled "Request Completed" by "Charlie" with a trash bin icon below it.
- A card with a striped shirt image labeled "Request Completed" by "Julia" with a trash bin icon below it.

Requirement 12: Users are able to view and reply to messages in inbox

ID	US12a
Feature	View the messages in the inbox (login required)
Description	<p>As a user, I want to view a list of messages in my inbox So that I can know which user accepted my request or users that are interested to connect with me</p>
<p>Given I am on any page under the website and have logged in When I select the mail icon in a bar on the top of all the pages Then I am redirected to the Inbox page And I am able to see a list of all users who contact me in the column on the left of the page and the detail conversations on the right of the page including their name, requested item, a message box to message back and all the past messages between that user and I </p>	
Estimate	4 (1SP = 2.5 hrs)
Priority	High



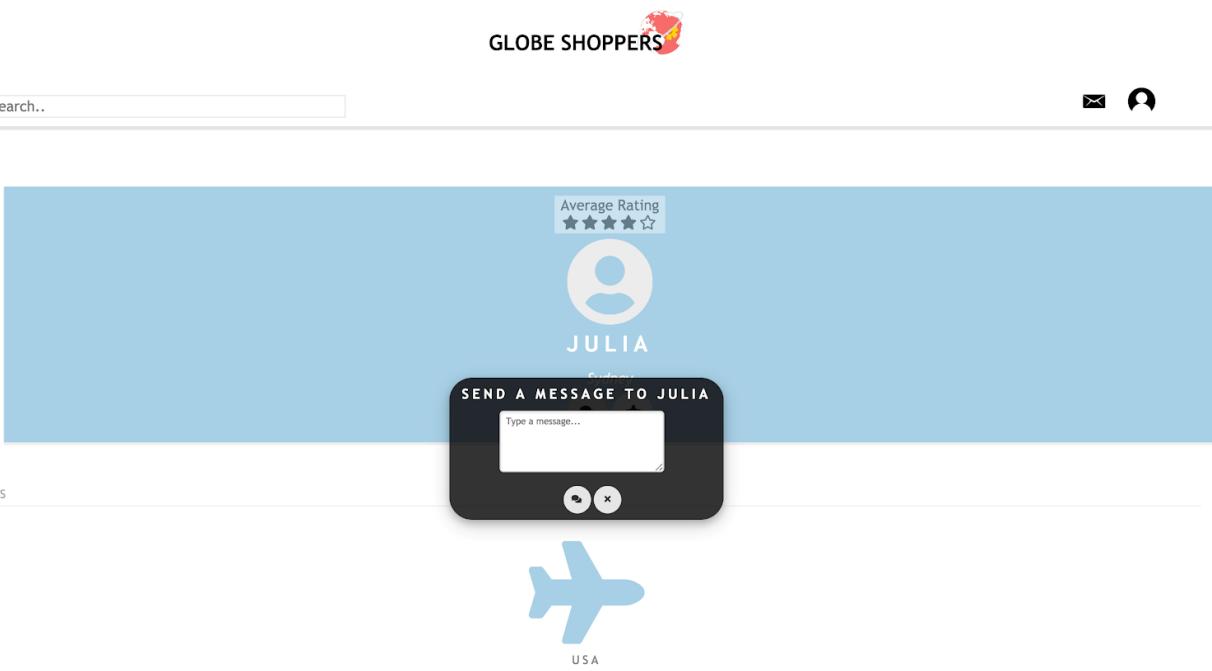
The screenshot shows the 'Messages with Charlie' section of the Globe Shoppers inbox. On the left, a sidebar lists 'MESSAGES' with names: Charlie, Tom, Julia, and Henry. The main area displays a message from 'GlobeShoppers' offering 3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color, with a status of 'Accepted'. A 'Complete' button is visible. Below this, a message from 'GlobeShoppers' states: 'This is an automated notification. Charlie has offered to purchase Lee's request for 3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color'. Another message from 'GlobeShoppers' says: 'This is an automated notification. Lee has accepted Charlie's offer to purchase 3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color'. A message from 'You' (Lee) is partially visible, starting with 'Hi, thank you for accepting my request. You can get the product from YesStyle'. A message from 'Charlie' is at the bottom, starting with 'From: Charlie'.

ID	US12b
Feature	Reply to messages in the inbox (login required)
Description	<p>As a user, I want to reply the message in my inbox So that I can tell the other user my purchase intention or some detail information of the meet-up place/time</p>
<p>Given I am on the inbox page and I click the user in the left bar who I want to send message to</p> <p>When I write a reply in the message box and click 'Reply'</p> <p>Then I will be redirected back to my inbox with my new reply updated at the button of the page</p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Low

The screenshot shows the 'Messages with Charlie' interface on the GLOBE SHOPPERS platform. On the left, there's a sidebar with 'MESSAGES' and a list of contacts: Charlie, Tom, Julia, and Henry. The main area is titled 'Messages with Charlie' and shows an offer for 3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color. The current status is 'Accepted'. A blue 'Complete' button is visible. In the message input field, the text 'hi👋' is typed. Below it is a 'Reply' button. Three notifications are displayed in colored boxes: a yellow box from 'GlobeShoppers' stating 'This is an automated notification. Charlie has offered to purchase Lee's request for 3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color'; another yellow box from 'GlobeShoppers' stating 'This is an automated notification. Lee has accepted Charlie's offer to purchase 3 CONCEPT EYES - Mood Recipe 2 Matte Lip Color'; and a dark blue box from 'You' with a message placeholder 'Message:' and the text 'Hi, thank you for accepting my request. You can get the product from YesStyle'.

Requirement 13: Users are able to contact each other through messages

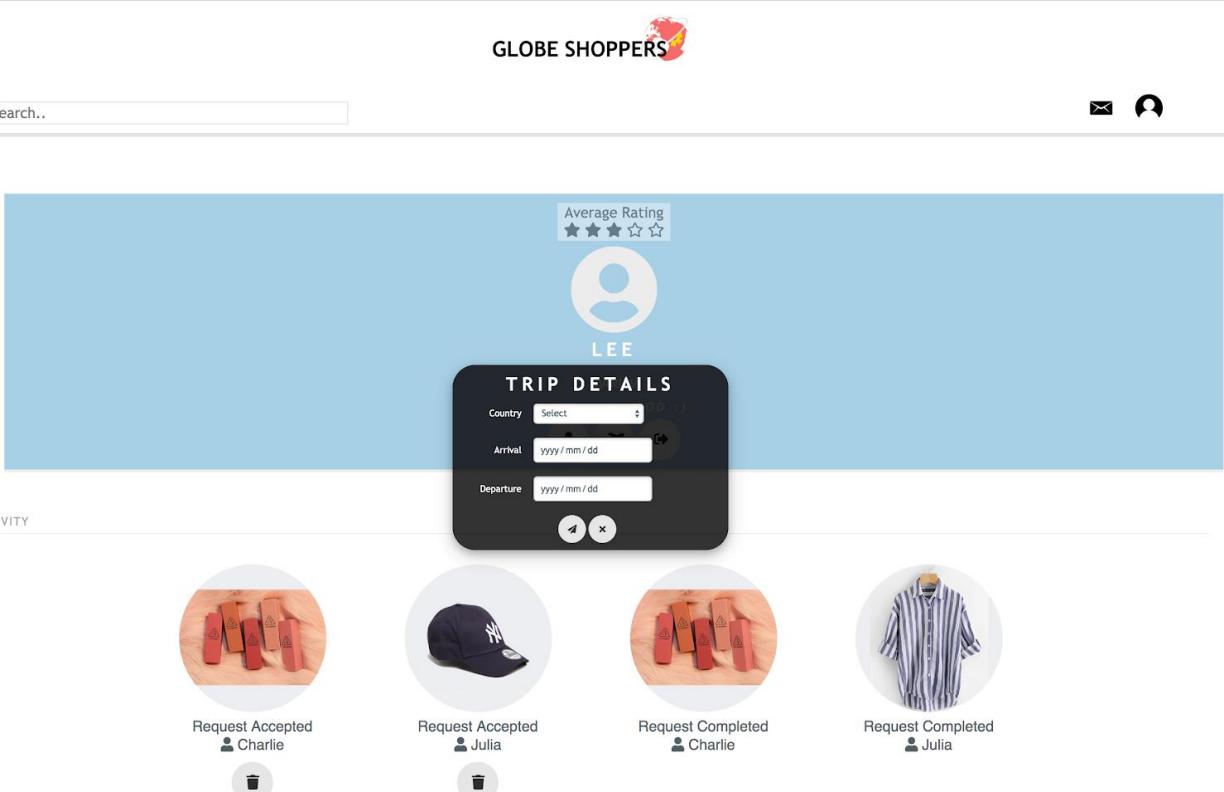
ID	US13a
Feature	Contact other users (login required)
Description	<p>As a user, I want to contact another user So that I can clarify purchase details with them</p>
<p>Given I am on that user's profile page When click the chat icon on their profile Then a pop-up message box will appear When I type in my question/message to the user and click the chat icon in the popup Then I can read my messages with this user in my inbox page</p>	
Estimate	3 (1SP = 2.5 hrs)
Priority	Moderate



The screenshot shows a user profile for 'JULIA'. At the top, there is a placeholder for a profile picture and an 'Average Rating' of 4 stars. Below this is a large blue button labeled 'SEND A MESSAGE TO JULIA' with a text input field that says 'Type a message...'. At the bottom of the profile section is a flight icon with the text 'USA' underneath it. The rest of the page is mostly blank, with some navigation elements like a search bar and a menu at the very top.

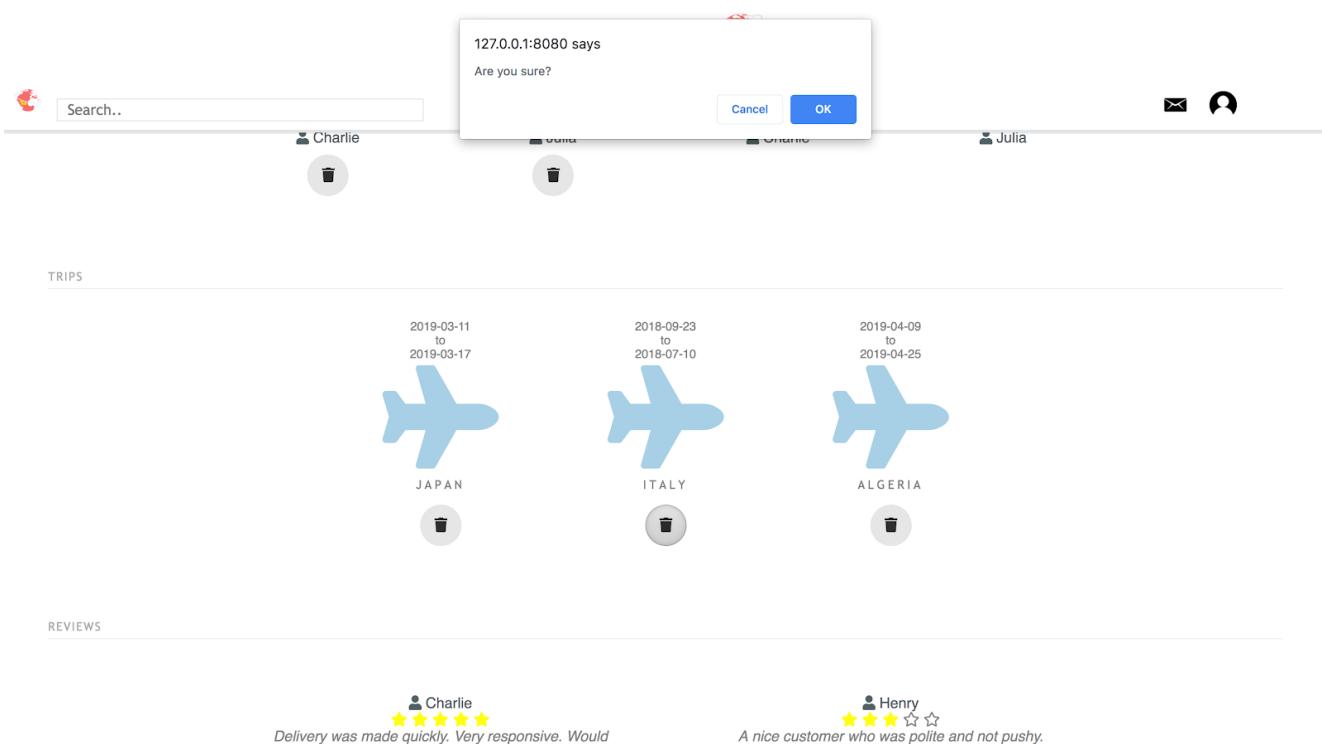
Requirement 14: Users are able to add or delete their upcoming trip information

ID	US14a
Feature	Add upcoming trips information onto my profile (login required)
Description	<p>As a user, I want to add my upcoming trip information on my profile So that other users can contact me if they need my help to purchase their request from my destination country</p>
<p>Given I am on my profile page When I click the flight icon Then a pop-up message box will appear with the box to select my destination country, trip start and return date When I have filled in my trip details When I click the submit icon to save these changes Then I will be redirected back to my own profile with information updated </p>	
Estimate	3 (1SP = 2.5 hrs)
Priority	Low



The screenshot shows a user profile for "LEE". At the top, there is a placeholder for "AVERAGE RATING" with five stars. Below the name is a circular profile picture placeholder. A modal window titled "TRIP DETAILS" is open, containing fields for "Country" (with a dropdown menu showing "Select" and "INDIA"), "Arrival" (text input "yyyy / mm / dd"), and "Departure" (text input "yyyy / mm / dd"). Below the modal, the word "ACTIVITY" is visible. There are four circular icons representing past requests: one for "Request Accepted" by "Charlie" (with a small trash icon below it), one for "Request Accepted" by "Julia" (with a small trash icon below it), one for "Request Completed" by "Charlie" (with a small trash icon below it), and one for "Request Completed" by "Julia" (with a small trash icon below it).

ID	US14b
Feature	Delete trip information on profile (login required)
Description	<p>As a user, I want to delete my upcoming trip information on my profile So that I can notify other users that I am no longer travelling to a country and cannot buy products there</p>
<p>Given I am on my profile page and have logged in as a user When I click the bin icon below the trip details Then a pop-up confirmation will appear with an “OK” button to confirm and a “Cancel” button to cancel the deletion</p>	
Estimate	2 (1SP = 2.5 hrs)
Priority	Low



The screenshot shows a user profile page with a "TRIPS" section. Three trips are listed, each represented by a blue airplane icon and the travel dates:

- 2019-03-11 to 2019-03-17 (Japan)
- 2018-09-23 to 2018-07-10 (Italy)
- 2019-04-09 to 2019-04-25 (Algeria)

Below each trip icon is a small circular trash bin icon. A modal dialog box is centered over the page, containing the text:

127.0.0.1:8080 says
Are you sure?

With "Cancel" and "OK" buttons.

Final Software Architecture

THE SYSTEMS EXTERNAL DATA SOURCES

Globeshoppers uses external data sources to provide information in order to make the website functional for its users. These functions include converting currencies and creating a product inventory to allow browsing of products and viewing products by their category. In order for our application to access external data sources, several Application Programming Interfaces (API) have been used.

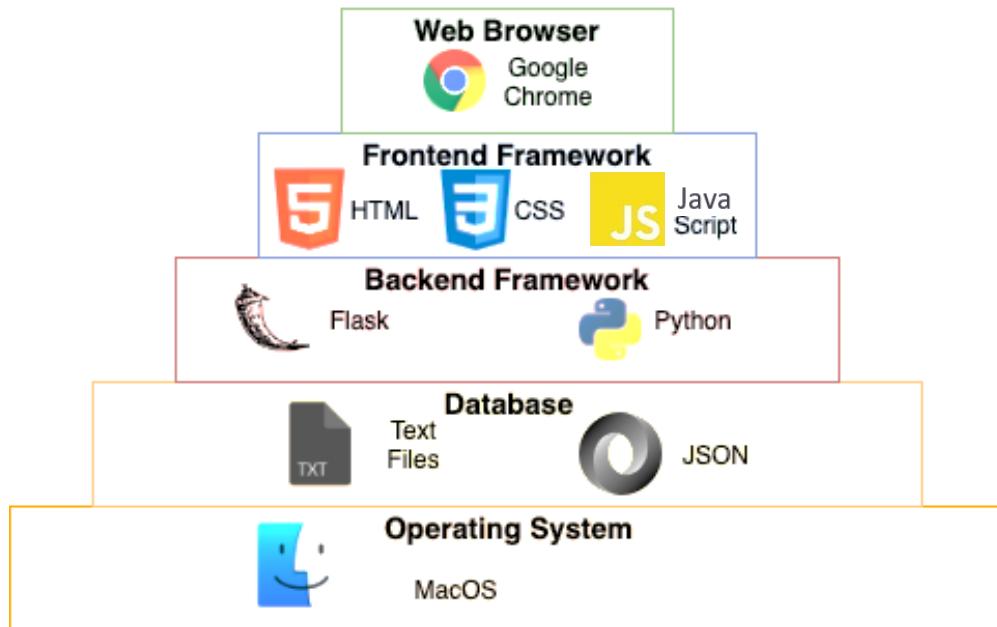
Foreign exchange rates API by European Central Bank:

This API is a free service that provides past and present exchange rates. The data it uses allows for currency conversion. In the website, when Australian consumers post a product they want from overseas, they also provide a price they are willing to pay. This price allocated is entered by the user in Australian dollars, however, if a user decides to provide help whilst travelling, it would be useful for the user to also see the retail and offered price in the country's currency where the product is located. Since the exchange rates are fluctuating constantly and the vast amount of countries where products may be from, it would not be logical or manageable to program a currency conversion. Hence, an external data source has been used to provide an easier way to incorporate this feature in our website.

Browse API by eBay:

The Browse API provides data on eBay's items including its description, category, reviews and images. Using the metadata provided by eBay, the API allows searching and gives information to sort products into categories. In relation to the Globeshoppers website, the API provides a wider set of products, i.e. products that have been posted by users for others to buy overseas, as well as allowing keyword searches. Also, viewing products by category is another feature incorporated into this API which has been used and extended to better suit the website. Thus, this API allows browsing features and displays a wide range of products.

SOFTWARE COMPONENTS – THE WEBSTACK



CHOICE OF LANGUAGE COMPONENTS

Backend Language

For the backend software development multiple languages were considered. Each of these languages had their advantages and disadvantages which allowed the decision to be made on what language to use.

The following were the languages that were considered by the team during the decision process

C

Advantages:

- Every team member was familiar with the language and thus the learning curve needed for was smaller
- Has a good performance when used on multiple devices

Disadvantages:

- Not frequently used for backend programming thus less libraries are available
- Time consuming
- Harder to maintain software in comparison to other languages such as python

JavaScript

Advantages:

- JavaScript is relatively simple to learn and implement.

Disadvantages:

- Team members are unfamiliar with the language and thus it could be harder to develop the backend
- No Object-Oriented Programming available
- Because the code executes on the users' computer, in some cases it can be exploited for malicious purposes. This is one reason some people choose to disable JavaScript.
- May be exploited for malicious purposes since the code executes on the user's computers. (This is one reason why some people choose to disable JavaScript)
- Client-side servers can be unpredictable causing different interpretations across different browsers (whereas server-side scripts always produce the same output)

Python

Advantages:

- Team members past experience working with the language limits the learning curve in programming with this language
- High level language with multiple resources available online to help solve any problems
- It is an Object-Oriented language.
- Multiple libraries for web development
- Compatible with multiple APIs which are needed
- Good at error handling
- Very compatible with the web server Flask which is going to be used
- Good performance across devices

Disadvantages:

- Slower to run than C
- High memory consumption due to flexible data types limiting its use with memory intensive tasks
- Requires more testing and more errors at runtime because its dynamically typed

Therefore, for this website Python was chosen mostly due to the team members familiarity with the program. To prevent losing time learning a new language, Python is the more desirable choice. Also, its simplistic nature, easy implementation and compatibility with the backend server, Flask, makes it the obvious choice for developing Globeshoppers.

Frontend Language

For the frontend development of our program we had to consider a language which is compatible while working with HTML and CSS. The language that we found which was most efficient for this was JavaScript.

JavaScript

Advantages:

- More dynamic, aesthetically pleasing webpages due to popups and page slides
- Good compatibility across devices and thus can work well on different operating systems with various web browsers
- Easy to learn as compared to other languages
- Object oriented programming language
- Quickly interprets and runs code
- Abundant resources to learn JavaScript
- Plugins can be used

Disadvantages:

- Some team members unfamiliarity with the language and thus it could be harder to develop
- Client-side servers can be unpredictable causing different interpretations across different browsers (whereas server-side scripts always produce the same output)

We also found the language TypeScript which is a superset of JavaScript which had some well-defined features and thus was also considered as a potential frontend language.

TypeScript

Advantages:

- Typescript is a superset of JavaScript and is designed for development of a large program that compiles to JavaScript.
- Can write only one type of code which can be implemented on various different browsers without the use of building tools like Webpack
- Due to the lack of high-level Object-Oriented Programming in JavaScript, TypeScript can be used to create larger projects and structure the system thus making the project easier to maintain

Disadvantages:

- Some team members were familiar with JavaScript due to previous experience and thus prefer to use that in order to create the frontend framework.
- TypeScript is harder to learn and required installation on the devices in order to be used

Thus, JavaScript being an easier to learn object-oriented programming language allows us to create well-structured code in order to increase the maintainability of our code and thus was used for the frontend framework for our software.

CHOICE OF IMPLEMENTATION/TECHNOLOGY OR FRAMEWORKS

Frameworks and technologies chosen or considered

The frameworks/libraries that were used includes Flask and Pure CSS for design and feature styling.

Client-side frameworks make it easier to produce interactive and dynamic pages compatible with multiple browsers/devices. They also provide pre-configured libraries for common functions. Low coupling was achieved by separating the contents of the page to HTML, styling and layout to CSS and the pages' behaviour through JavaScript.

Other client-side options included React and Angular. Angular uses Typescript and plain HTML templates to set up routing and http request handling. It is highly testable and uses languages that are already being used making it more familiar which would have allowed for faster prototyping; however, it still had a steep learning curve especially with the inexperienced knowledge of Model-View-Controller architectural patterns. React is a scalable, re-usable, flexible JavaScript library which has a big community and ecosystem and it constructs efficient virtual Document Object Models to allow faster rendering. However, it's View-oriented and required prior knowledge on JSX. There was also no defined methodology on how to implement React, and it isn't as straightforward as JavaScript and HTML.

Server-side frameworks make it easier to write, maintain and scale web applications. They consist of libraries that simplify databases, authorization and rerouting URLs. JSON was used, which is a templating system which takes stored data and formats it into text files. Flask (@app.route) helped us to route requests to the appropriate handler. This was useful because it allowed URLs to change without having to change the code. Flask also made it easier to access data in a request through working with HTML forms in get/post requests.

Alternative server-side options included Rails, Node.js/ Express.js, Django, Laravel and Spring. Rails uses Ruby, Laravel uses PHP and Spring uses Java, which eliminated them. This leaves Django and Express/Node.js. Django is similar to flask because they both are based off Python; however, Flask offers simplicity and flexibility while Django offered admin panels, database interfaces and ORM (Object-relational Mapping). The URL-router in Django was also more complicated because it uses regular expressions which was harder to use for beginners. (No members know Django). Flask was chosen over Django mainly because all team members have worked with Flask before. Express was another option since JavaScript was used in front end, however Python syntax was easier and more readable, and also had more extensive documentation, libraries and object-orientation.

Hence, the main reason these technologies were chosen, is that we were familiar with how they work and they served their purpose well, as well as a few new technologies (JSON and JavaScript) because they were quick to learn and there was plenty of documentation out there for us to learn from.

Table of Technologies with their uses and benefits

TECHNOLOGY	USE	BENEFITS
Python classes and libraries	Managing information about users and products. This includes creating new users when they register, storing updated information, adding and deleting products.	Making classes is efficient (we've all used it before), as well as it allowed us to create attributes, methods(functions) and sub-classes that can inherit/override from their parent class.
JSON	Making stored class instances persist in a database (e.g. store available products so that when we add new products, they do not disappear from the website when we refresh).	We chose JSON over SQL or other databases, because it worked cohesively with python classes (python classes can easily be translated into JSON and saved in text files and can also be used in front-end JavaScript). JSON is also more compact and readable and allowed use of arrays compared to XML. XML parsing process is also slower. JSON was better for us than SQL since it is more flexible than a relational database with set schemas, and has a better system for variable data structures.
HTML/CSS	Displays information on websites and manages layout.	We've all coded in this language and there were lots of source codes online for layout and styling. It also worked cohesively with Flask in terms of forms, submit buttons etc.
Flask and JavaScript including their libraries (redirect, url_for, request, render_template)	Accept requests from client/user. Flask was used mainly for redirection of pages, and JavaScript for pop-ups.	We used a combination of both so that not every request requires refreshing and keeps the website more efficient and professional. Flask handles requests which require authorization or going through many conditional statements, while JavaScript handles confirmation or error messages. Additionally, Flask/Python have large libraries of functions which we used for conditional statements.

Pure CSS	Provides HTML/CSS design templates for buttons, navigation, tables and also supports JavaScript plugins	It has thorough documentation online and is open source. Pure CSS also is compatible across most modern browsers. This provided our website with a professional and consistent look.
----------	---	--

CHOICE OF PLATFORM

Since all members have MacOS all code was written and tested on MacOS and Chrome. Our customers should be able to access our website and it should look the same on recently updated (current update or one previous) MacOS, and on Chrome, Firefox and Safari.

Machine Requirements:

- Website displays all necessary features in a useable and easy to navigate manner on multiple machines using the same browser especially Windows, MacOS and Linux.
- Website displays all necessary features in a useable and easy to navigate manner on multiple browsers such as Chrome, Safari, Firefox etc.
- Styling of buttons, tables, texts and fonts remains consistent across the website.
- Spacing and layout of information should remain consistent. When the browser display resolution is changed, text/information should not be cut off or hidden and must adjust.

Websites do not look the same across different platforms because different browsers interpret code and render pages differently. Issues also occur when an old browser version doesn't support the latest standards e.g. HTML5 forms. Different operating systems also visualise buttons, check boxes and fields differently based on their own style. For example, Mac default submit buttons are ovals with shaded gradient but look square and flat on PC. The user's display resolution (whether it takes up the whole screen or smaller) also affects how the website looks.

Solutions

- We used and extended Pure CSS custom buttons to bypass the issue with different operating systems using different default styles.
- We used relative dimensions instead of static. For example, width = 100% rather than width = 1024px so that the size of the image/ feature adjusts depending on resolution and page size.
- We provided fallbacks or avoided features that do not display on all browsers. For example, images have an alt attribute describing the picture and flash was avoided.
- We did cross-browser and cross-machine testing on Chrome and Firefox and mac OS and Linux OS to check if the website still maintained its functionality.

SUMMARY OF KEY BENEFITS OF THE DESIGN

There are many benefits to our website's design including its user interface, functionality and architectural choices. However, our website's unique idea is the key benefit to our website as it truly possesses a potential market in Australia. Since Australia is diverse but geographically isolated, many Australians are dissatisfied with expensive or a lack of shipping, so our website would be incredibly useful.

Regarding Globeshoppers' user interface, it has several benefits due to the effective frontend design. One benefit is it's easy to understand nature as we have included extensive information on how to use our website and the idea behind it on our welcome page. Also, it's minimal and consistent design makes it visually appealing to its users and makes easy navigation through the site.

Furthermore, we considered ethical issues whilst designing our website. We have prevented users from viewing other users' private information such as their purchase history and the dates of their trips. Additionally, we created a terms and conditions sheet located at the websites bottom banner and we also displayed forbidden imported items whilst adding products where we have provided the link to the government page for more information. This benefits our website as it takes into consideration user's privacy and legalities.

Also, from a technical aspect, we picked beneficial external sources, languages and frameworks to allow for an effective website design. This effective design is exemplified through the absence of errors whilst running our website.

The external sources, the foreign exchange rates API and eBay API, allows for an enhanced user experience. The foreign API allowed for currency conversions to occur allowing travellers to see what price an item is in the currency where the item is located whilst the eBay API provides users with a wider library of products to potentially buy and receive. Also, the use of these external sources saved us time whilst developing the website since the eBay API possessed a searching capability we could easily integrate.

Pure CSS and JavaScript enhanced our frontend design since it created consistent designs and allowed for more complex functions such as popups which also helped to limit excessive amounts of html pages.

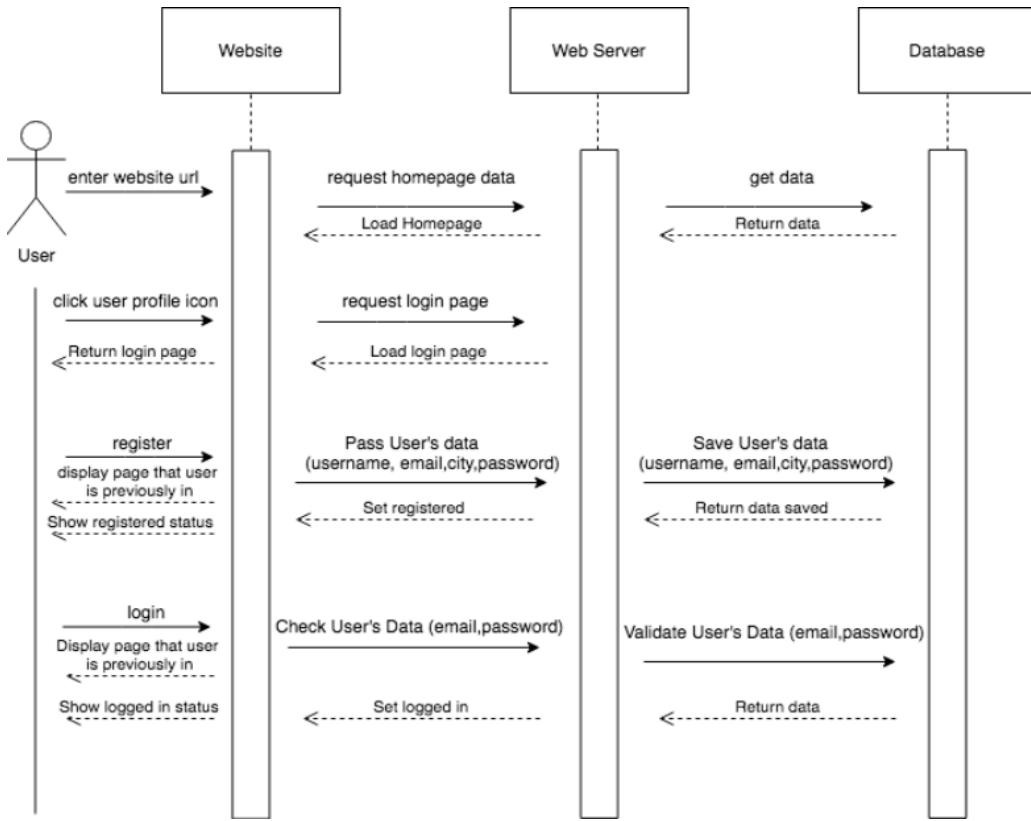
The combination of Flask, Python and JSON was extremely advantageous as it allowed us to create a backend framework that maintains all of our data. Also, we were able to use the extensive libraries flask and python contain to load and dump into JSON and redirect URLs.

Our familiarity with the languages chosen allowed for quicker development as we didn't have to learn the language from scratch. This allowed us to develop a highly functional website. Moreover, the compatibility of the languages we picked allowed for the website to develop without severe complications.

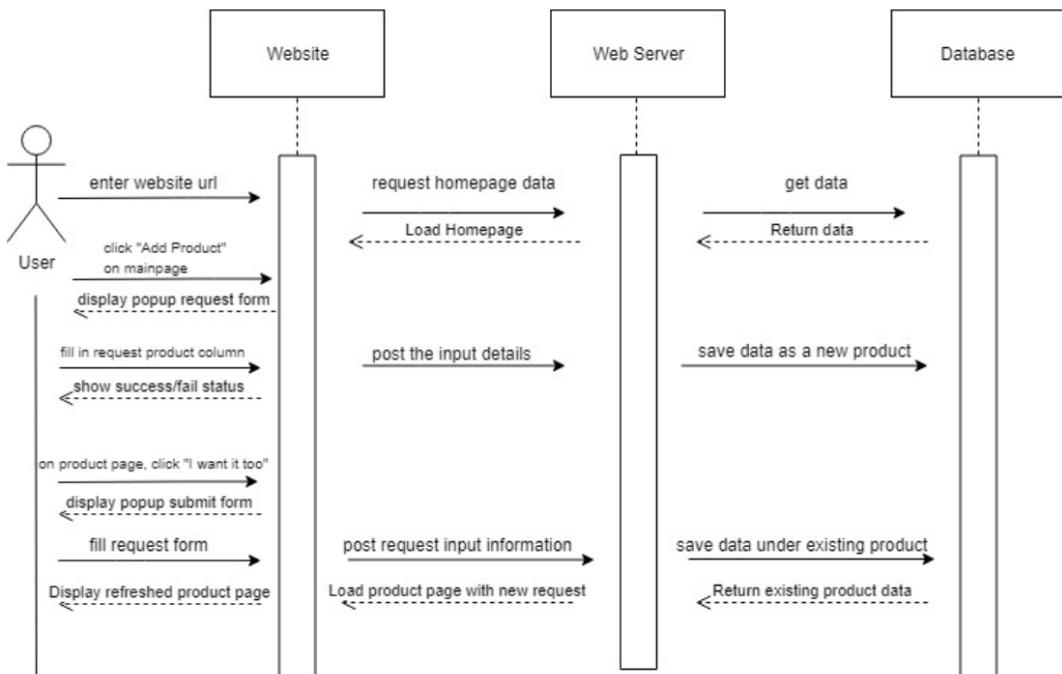
For developing the website, Mac OS was used since all team members use this operating system. Due to past development of websites on this operating system, incompatibility and inconsistency issues were reduced by using one operating system. Similarly, by using chrome for development and testing we were able to avoid disparate website designs.

SEQUENCE DIAGRAMS FOR EACH USE CASE

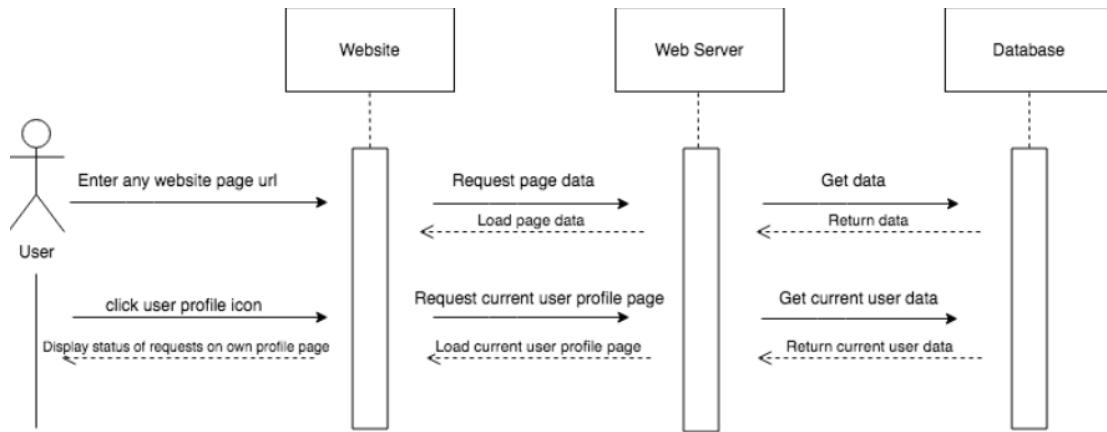
Requirement 1



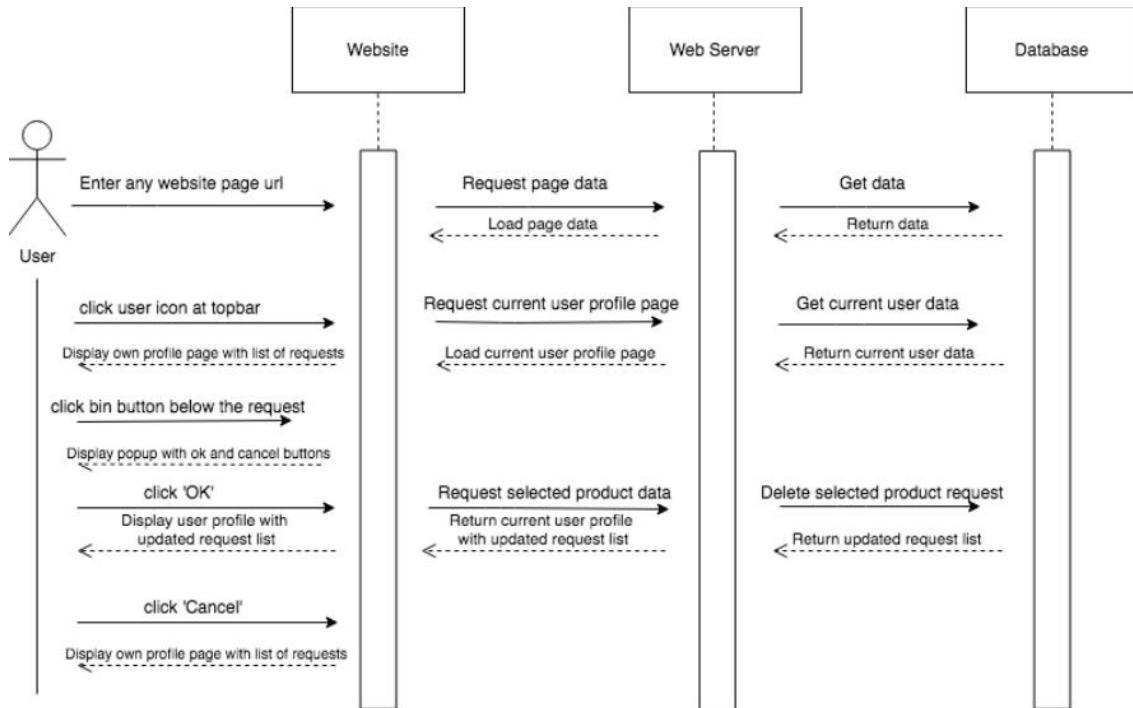
Requirement 2



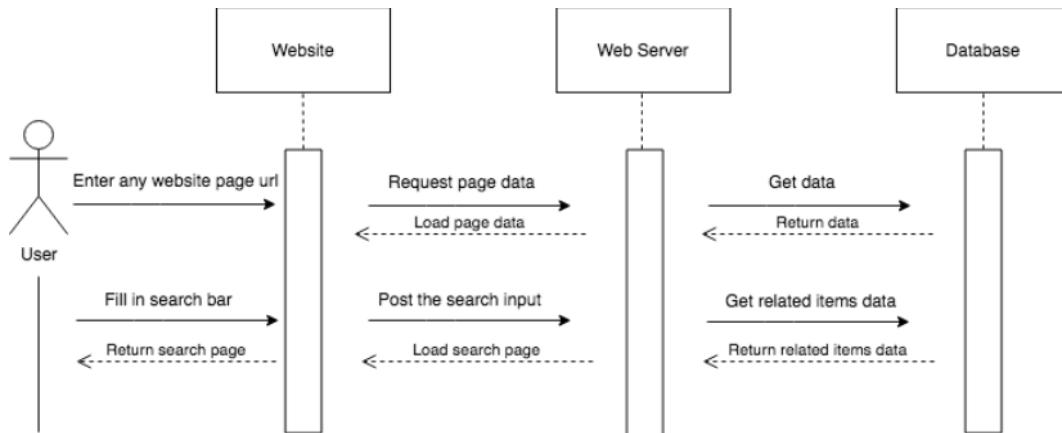
Requirement 3



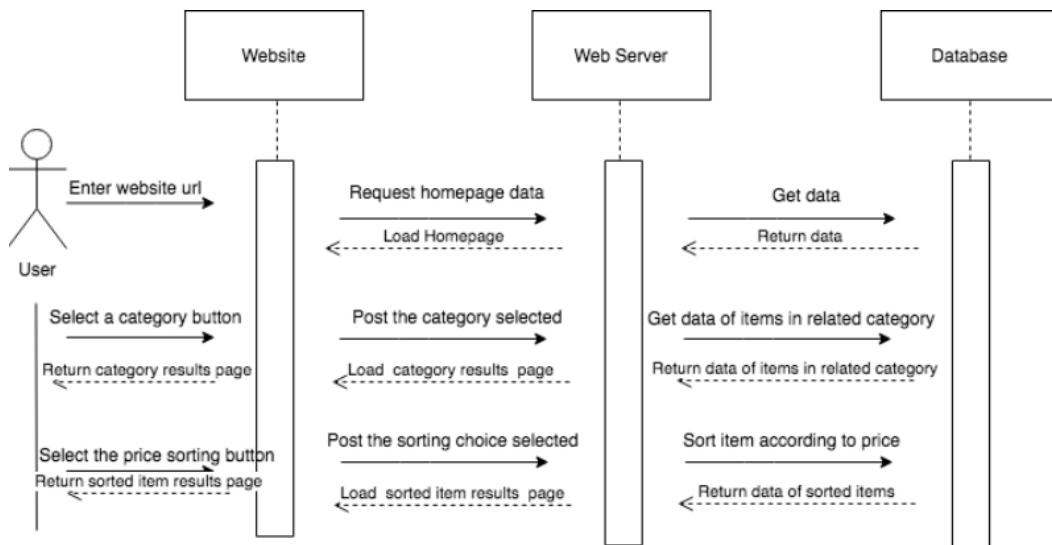
Requirement 4



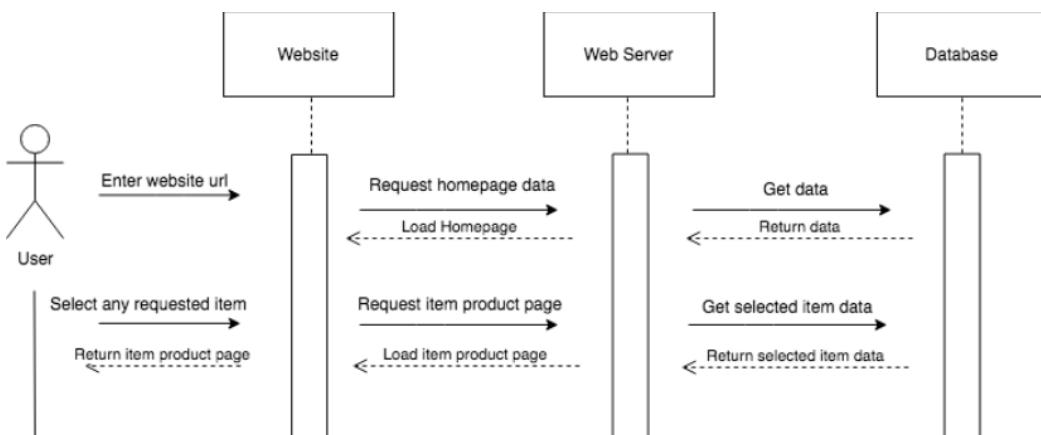
Requirement 5



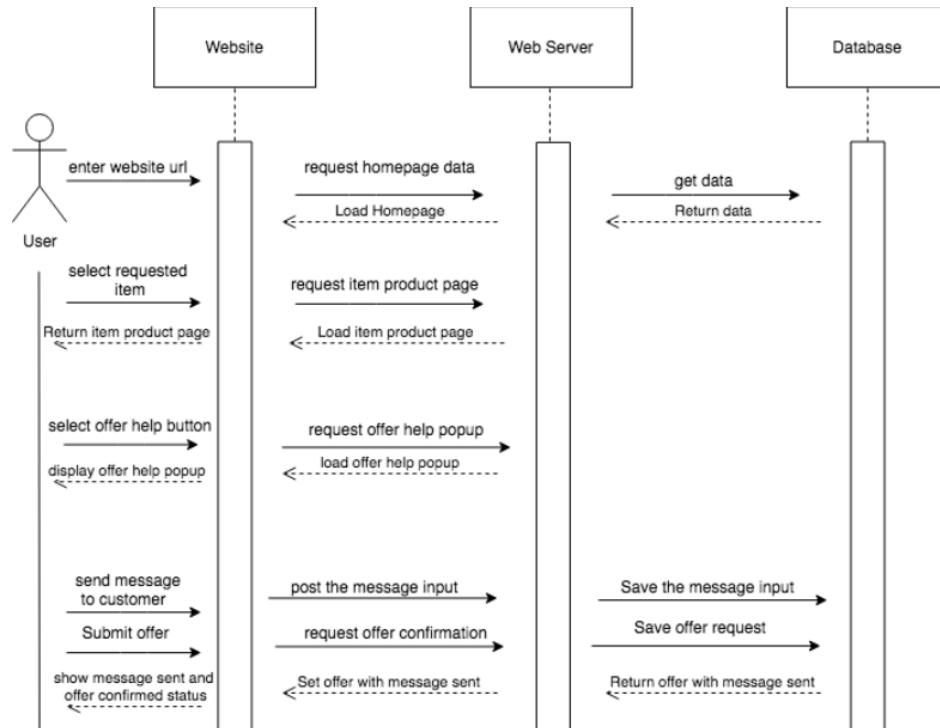
Requirement 6



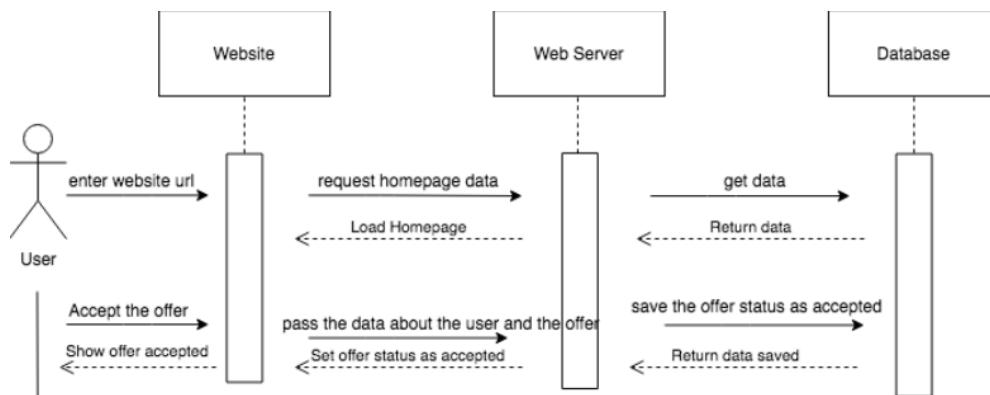
Requirement 7



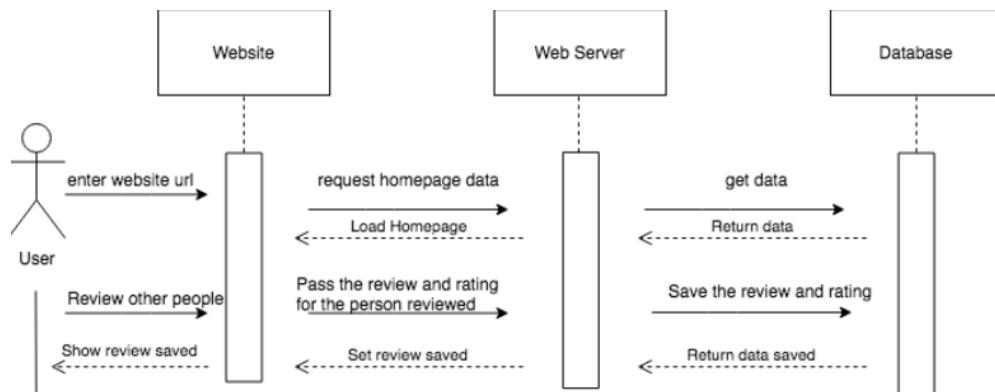
Requirement 8



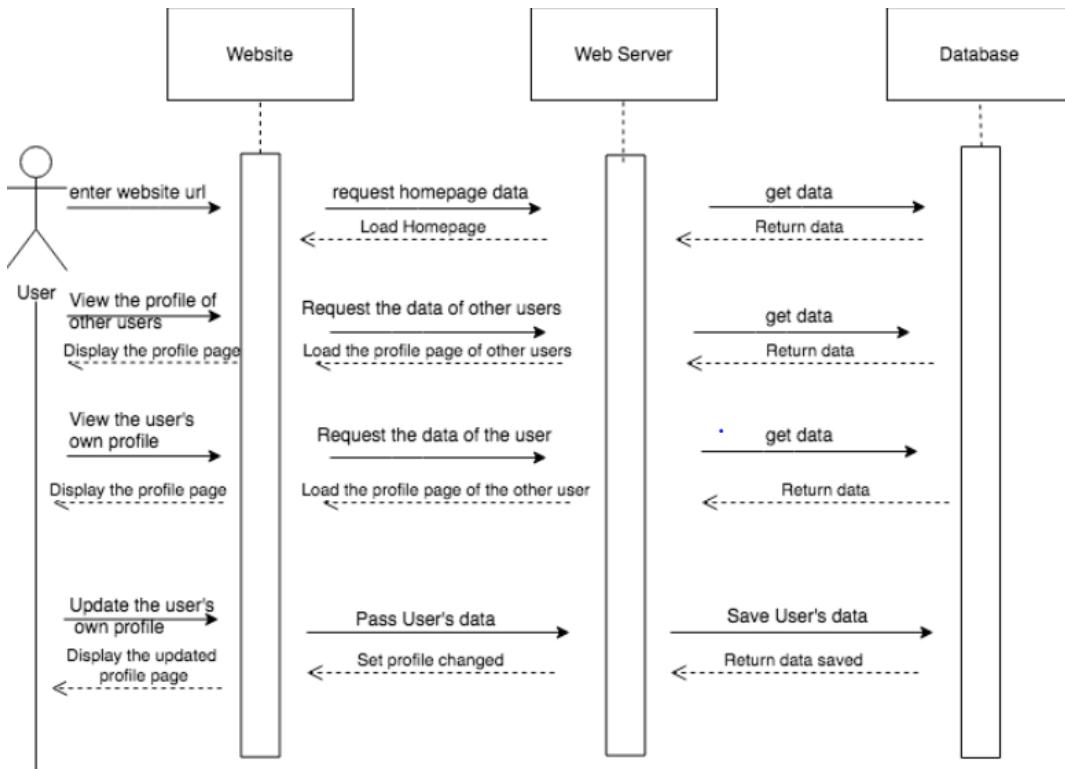
Requirement 9



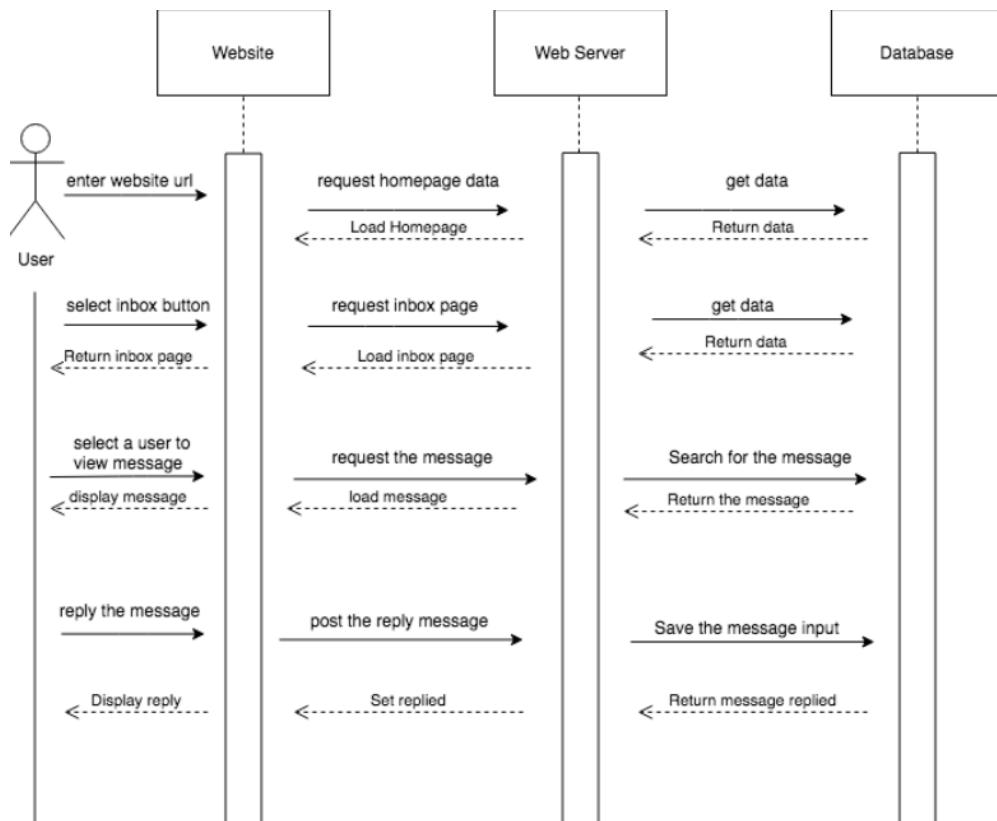
Requirement 10



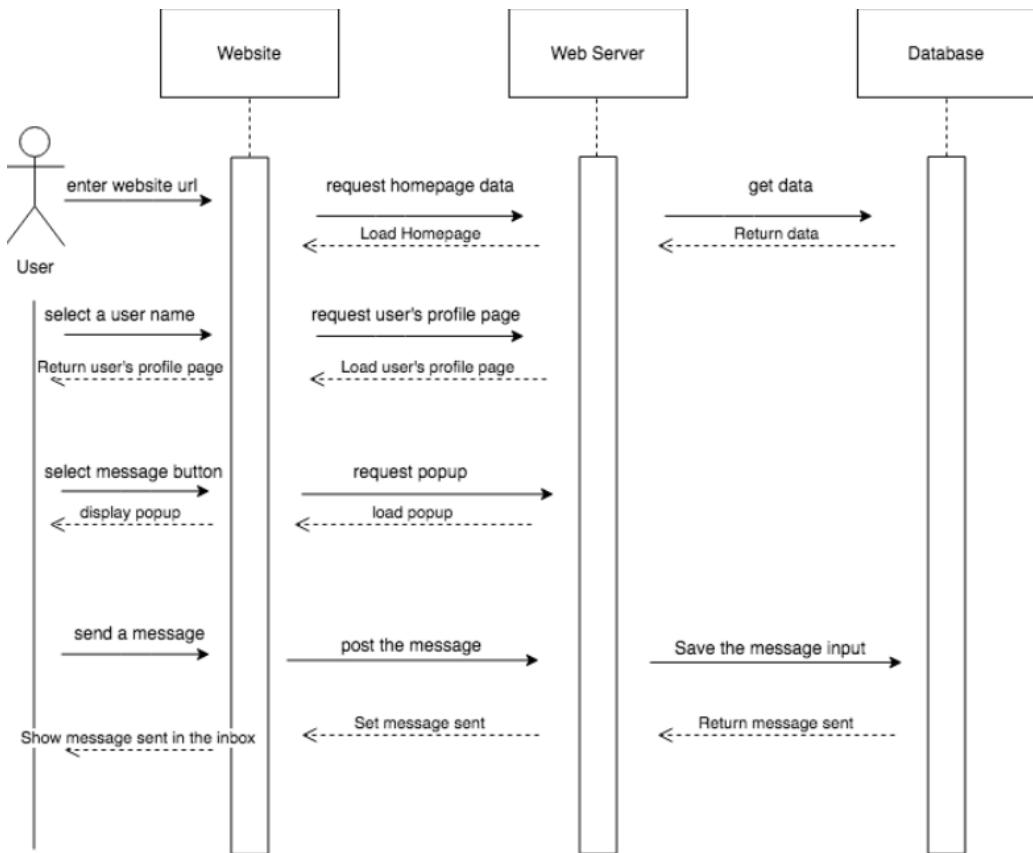
Requirement 11



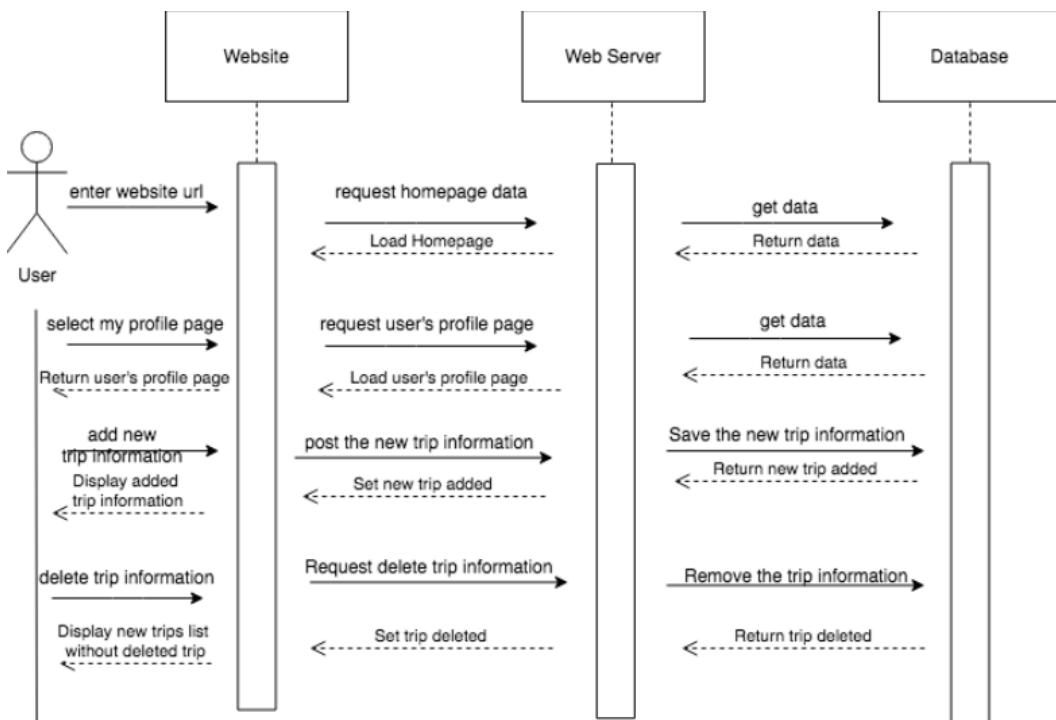
Requirement 12



Requirement 13



Requirement 14



Team Organisation and Conclusion

ORGANISATION OF THE TEAM

We used 'Asana' to delegate tasks and notify everyone when tasks are completed / started. This helped us make sure all assessed parts were covered and there were no overlaps. Facebook Chat was used for urgent questions and discussions of problems encountered. We held regular meetings on Tuesdays with our mentor and after, as well as additional meetings when needed. On GitHub, we used branches, named after our tasks, to simultaneously edit files, and merged them at weekly meetings. We tried to delegate tasks so that people were not depending on others to finish their tasks first. For deliverables, we would discuss drafts of the information needed e.g. problem statements together during meetings, and then refined each section individually. Additionally, we split tasks by frontend and backend so people could focus on one language.

RESPONSIBILITIES OF THE TEAM

	DEL 1	DEL 2	DEL 3	DEL 4	DEL 5
Amanda	<ul style="list-style-type: none">• User Stories 3 & 4• High Fidelity Prototype (main page, search, categories)	<ul style="list-style-type: none">• Software Architecture Q4,5• Sequence diagrams 2-4	<ul style="list-style-type: none">• Frontend search• Frontend categories• Frontend eBay API• Backend search• Backend categories• Backend eBay API	<ul style="list-style-type: none">• Backend review & ratings• Backend inbox including automated messages• Backend deleting offers• Backend eBay API connect to product page• Backend redirection saving previous page• Sign up/login background design	<ul style="list-style-type: none">• Team organisation
Carol	<ul style="list-style-type: none">• User Stories 1c, d, e	<ul style="list-style-type: none">• Fixing User Stories	<ul style="list-style-type: none">• Frontend inbox• Frontend product page including 'offer'	<ul style="list-style-type: none">• Frontend search including search filter• Frontend categories	<ul style="list-style-type: none">• User Stories update

	<ul style="list-style-type: none"> • High Fidelity Prototype (profile, review, edit profile, inbox) 	<ul style="list-style-type: none"> • Sequence diagrams 9-11 	<p>help' & ' I want this' functions</p> <ul style="list-style-type: none"> • Backend inbox • Backend product page including 'offer help' & ' I want this' functions 	<ul style="list-style-type: none"> • Frontend country categories • Frontend product page • Frontend eBay API photos 	<ul style="list-style-type: none"> • Sequence diagram update
Lavanya	- Low Fidelity Prototype	<ul style="list-style-type: none"> • Software Architecture Q2,3 	<ul style="list-style-type: none"> • Frontend main page • Frontend login & signup • Frontend product page • Frontend profile page • Frontend base including navigation bar & add trip • Frontend search • Frontend categories 	<ul style="list-style-type: none"> • Frontend main page • Frontend welcome page • Frontend base including navigation bar and outro bar • Frontend about us page • Frontend inbox 	<ul style="list-style-type: none"> • Fix DEL 2 • Purpose of system • List of features
Nicole	<ul style="list-style-type: none"> • Problem Statement • User Stories 1a, b • Final User Stories fixing 	<ul style="list-style-type: none"> • Fixing User Stories • Sequence diagrams 1,8,12-14 	<ul style="list-style-type: none"> • Backend login • Backend logout • Presentation slides • Presentation infographics 	<ul style="list-style-type: none"> • Backend edit profile • Backend login including adding city • Research legal (terms of use) • Backend redirection when not logged in • Collect icons for design • Backend search filter & sorting • Backend countries category 	<ul style="list-style-type: none"> • User Stories update • Sequence diagram update

Sarah	<ul style="list-style-type: none"> User Stories 2 High Fidelity Prototype (base, product page, add trip, login, signup) 	<ul style="list-style-type: none"> Software Architecture Q1,6 Sequence diagrams 5-7 	<ul style="list-style-type: none"> Backend main page including add products and requests Backend profile pages Backend add trip Backend foreign currency API 	<ul style="list-style-type: none"> Frontend login & signup Frontend own profile (including deletion and reviews) Frontend other profiles Frontend edit profile Frontend add trip Frontend add review Frontend viewing other profiles Backend other profiles Backend trending items Fix backend editing username 	<ul style="list-style-type: none"> Organise GitHub Finalise report
-------	---	---	--	---	--

HOW THE PROJECT WENT IN OUR OPINION

Overall, our time management was done fairly well since we spread our workload throughout most of the weeks (there were some exceptions when a deadline was far away and we had other assignment priorities). We set deadlines before meetings to make sure tasks were done on time. Tasks were also completed efficiently. Asana helped us remain organised and aware of other people's roles so we know who to work with specifically e.g. frontend and backend for profiles can discuss together to make sure both parts fit cohesively. We also communicated well with each other, notifying others when problems occurred or when parts were finished, as well as for feedback. Everyone contributed and was present in discussions. In the end, we were all satisfied and proud of our final product. We achieved more than we expected and since this wasn't our first time building a functioning website, we knew the basics and could focus on more specific details and intricate features.

ANY ISSUES/PROBLEMS ENCOUNTERED

For the technical side, we were able to solve most of the problems we encountered. For example, a main issue that was brought up during feedback was the legality aspect of our website, including designing a trust system to monitor or protect people from not being scammed. To overcome this problem, we included a terms of use detailing illegal items and a penalty as well as a rating and review system so that good behaviour would be encouraged, and users would be able to spot suspicious behaviour. We also considered monitoring specific words used when products are uploaded (e.g. banning illegal words), or allowing users to report and blacklist others. An initial problem we had was inconsistent design, partly due to the fact that everyone had different visions about the features and layout of the page and also because we weren't using any frontend framework for deliverable 3. To resolve this, we implemented Pure CSS and reused button/banner/font designs throughout the website. We also had issues with aligning divs side by side and not being dynamic when the page size changed, because we were initially spacing them out by pixels or percentages. We fixed this by using Flex CSS. Our eBay API products also weren't showing images, and if there were images, they would be blurry and unclear. Their object format was also different so it was difficult to combine with our own products, and this affected our search filter because we weren't able to control which eBay products would be shown. We partially fixed this just by manually finding better quality photos for a selection of test products, and creating a new separate dictionary to store eBay item's information so that it was the same as our own products.

There were technical problems we weren't able to implement or fix. For example, we considered payment options (some sort of API/PayPal) but we weren't able to implement this due to time and cost. We also had some frontend bugs such as base interfering with other pages (blocking popups), icons not displaying on Macs (but showing on Linux) and how sending a post request through inbox always refreshed the page to show the first message thread (not necessarily the user we sent the message to.) We also had to slowly adapt to learning how to use Json and in the end abandoned typescript altogether because we were more comfortable with JavaScript.

For non-technical issues, our main problem was to do with GitHub, branches and merge conflicts. Splitting workload into frontend and backend helped because there weren't multiple people editing one html file for example, but it made it harder to combine these two parts. This meant when both sections were completed, we would spend a lot of time just making it work cohesively together. Since different features were being implemented at the same time, there were a lot of merge conflicts in routes.py and system.py. Because someone had to manually fix all these conflicts, changes might be accidentally removed or replaced.

WOULD WE HAVE DONE IT DIFFERENTLY

Overall, we worked really well together, we were organised, active, and completed tasks on time. The technical changes we would make would be:

We would try to finish the base.html first. This is because merging base into the already existing other html files meant class names or IDs that were the same had to be changed. This is very tedious and could have been avoided if base was our first priority.

- We would discuss a colour theme, or use Pure CSS earlier. This would allow us to have consistent design and layout. We started using a frontend framework too late, because we had to redesign and re-program most of our html pages again within two weeks between deliverable 3 and 4, as well as implement new features.

- If we could, we would use a different API or an API with more documentation and tutorials e.g. Google. eBay has a lot of notes on the classes used, but not much on how to access that data. Tutorials were often based of PHP instead of Flask/Python and took a lot of tweaking to make it work with our website.
- If we had more time, we would try and use new and different libraries and languages e.g. React or Typescript. Because most of the languages we used we were somewhat familiar with, we felt more comfortable working with them especially under deadlines. However, it would benefit us if we broaden our language choices and work with new ones.
- To solve the GitHub merge conflict issue, we should split system.py into multiple files, separated by their feature. Right now, system.py holds all the functions routes.py calls, and could be broken into different files. This means people working on different features won't have to edit the same system.py file.
- Finally, we only fully rehearsed our demo prototype for deliverable 4 a day before it was due. This meant we weren't able to fix all the bugs we found on that day. We should have rehearsed earlier, so we would have more time to fix any issues.