CSC 536
Due 3/13/19

Project Status Report 1

Robe Rental

- 3: Robe Rental
- Prathyusha Butti, Priyamvadha Gopalakrishnan, Ripan Chowdhury and Royan Tuscano

1. Introduction

- We have developed a minimal viable system implementing the following features:
  - A home page with images and details of available garments.
  - A sign in button that takes users to a sign in page.
  - A sign up button that takes users to a sign up page.
  - A team page with profiles of our team members.
  - A how it works page to describe about the website and the services that are offered.
- The user stories we have implemented for this iteration are as follows:
  - User Story 5 : As a seller, I want to display the size and material of the garment so that the buyer can choose a garment based on the size and material.
  - User Story 7 : As a user, I want to quickly learn about the website's rental policies so that I can comply accordingly.
  - User Story 11 : As a buyer/seller, I want to login to my profile anytime so that I can update my personal details.
  - User Story 13 : As a seller, I want to display the price of a garment clearly so that the buyer can select garments based on price.
- We have prioritized our user stories based on feedback obtained from talking to friends and fellow students.
- Problem Definition : **"No changes"**
- Basic Website Design :
  - Front End technologies use HTML, CSS, Angular 7, Angular Material, JavaScript and TypeScript. Operates on port 4200 requests are sent to port 3000 on backend.
    - **Home** contains the list of garments available to rent.
    - **Sign in** to sign in to the website.
    - **How it works** to describe about the website and the services that are offered.
    - **MyAccount** it will show the details about the owner of the account. Information like address credit card/Debit card details.
    - **Cart** it is used to store the item that is selected by the customer.

        - **Sign Up** this part is used to sign up for the website.

    - Backend technologies used MongoDB, express and bcrypt. The server operates on port 3000.
        - MongoDB NoSQL database to store user information that is email id and password.
        - The **api/signup** post request is used to insert the login details like email and password into the MongoDB database. The password is encrypted using bcrypt module.
        - The **api/login** post request is used to verify the user who is trying to login . We check if user exist if he does than we check his password. If it's correct than we will log in him else he is rejected.

- Technology :
    - We have included TypeScript, ExpressJS and MongoDB as we have decided to implement MEAN stack in addition to AWS, HTML, CSS, Angular, JavaScript in the Technology stack.

- Team Roles : **“No changes”**
    - Prathyusha : Front-End Developer and Product Owner
    - Priyamvadha : Tester and Scrum Master
    - Ripan : Back-End Developer
    - Royan : Back-End Developer

2. Customer Value and Problem Definition

- *Overview of Customer Value:* **“No changes”**

  The primary customers fall under two types:
    - Buyers who need a garment for one time use.
    - Sellers who will rent their clothes.

Unmet Customer Needs

- Customer needs:
    - A user/buyer wants to rent a garment for a low cost or for one-time use without paying full price for it.
    - A seller wants to rent or sell dresses they no longer want to use and make a business out of it.

- Desired overall experience:
    - To provide various categories for the buyer to select a garment of their choice.
    - To build a flexible business model that will have fair pricing depending on the number of days of rent for both the buyer and seller.
    - To provide profiles for both the buyer and seller to do the rental activity.
    - To integrate a payment system for purchases.
    - To address following type of internet customers
        - Product focused shoppers: Search bar or navigation that quickly shows the product customer is interested in buying.

        - Browsers and Researchers: Show customized products based on users' browsing history and past searches.
        - Bargain hunters: Give exclusive deals on products to this type of customers.
        - One time shopper: Offer incentive to visit again by offering rewards based on history.

- Unmet needs and user stories based on the unmet needs:
    - There are people who do not want to buy a dress for just one time use as they cannot afford to buy it, for example a wedding gown or prom dress. In such cases the person can rent the dress for a week for fraction of price of original dress .This will save the purchase cost for the buyer.
    - There are people who have too many designer clothes that they don't use, they can rent those clothes and generate revenue.
- _Overview of Problem Definition_ : **"No changes"**
    - Two major customer opportunities we found and plan to address are:
        - Excessive items in a wardrobe.
        - Budget constraints holding back garment purchase.
    - Options for addressing these needs:
        - Website for connecting a potential garment buyer with a seller.
        - Mobile app to connect a buyer to a seller.

  We chose to build a website first and then at a later point, make the website adaptive to screen size.

- Any changes from the Project Proposal? **"No changes"**

3. Use Cases

For each use case, show the basic flow and give titles and one-line descriptions of the alternative flows separately from the basic flow.

Goal 1 Basic Flow: *Rent garment to a guest user*

1. Guest opens Robe Rental website on a desktop computer

   {Display Garment}
2. Website displays garment images that are available for rent
3. Guest goes through the description, size, material for selecting a garment.
4. Guest selects garments to add to cart

   {Save Cart}
5. Systems saves cart information for Guest
6. Guest checks out cart without creating a profile
7. System requests payment information
8. Guest enters payment information

   {Authenticate Payment}
9. System authenticates payment

   {Generate Confirmation}

10. Guest receives confirmation number for their rental
11. Guest closes website

Goal 1 Alternate Flows

Alternate Flow 1 - Authentication failure

At **{Authenticate Payment}** if authentication has failed
The system displays "Authentication Failed" message
Resumes basic flow at **{Save Cart}**

Alternate Flow 2 - Network disconnects during a transaction

At any point between **{Display Garment}** and **{Generate Confirmation}**
if network connectivity is lost
The system displays "Unable to connect" message
Resumes basic flow at **{Display Garment}**

Alternate Flow 3 - Garment unavailable

At **{Save Cart}** if selected garment becomes unavailable
The system displays "Sorry, an item in your cart has been claimed" message
Resumes basic flow at **{Display Garment}**

Goal 2 Basic Flow: *Rent garment to a registered user*

1. Registered user opens Robe Rental website on a desktop computer

   {Display Garment}
2. Website displays garment images that are available for rent
3. User goes through the description, size, material for selecting a garment
4. User selects garments to add to cart

   {Save Cart}
5. Systems saves cart information for user
6. User checks out cart

   {User Payment Services}
7. System displays saved payment information
8. User accepts payment information

   {Authenticate Payment}
9. System authenticates payment

   {Generate Confirmation}
10. User receives confirmation number for their rental in their email

    {User Profile services}
11. System saves rented garments in user's profile history
12. User closes website

Goal 2 Alternate Flows

Alternate Flow 1 - Network disconnects during a transaction

At any point between **{Display Garment}** and **{User Profile Services}**
If network connectivity is lost
The system displays “Unable to connect” message
Resumes basic flow at **{Display Garment}**

Alternate Flow 2 - Garment unavailable

At **{Save Cart}** if selected garment becomes unavailable
The system displays “Sorry, an item in your cart has been claimed” message
Resumes basic flow at **{Display Garment}**

Alternate Flow 3 - Authentication failure

At **{Authenticate Payment}** if authentication has failed
The system displays “Authentication Failed” message
Resumes basic flow at **{Save Cart}**

Goal 3 Basic Flow: *Advertise robe rental policies to all users*

1. User opens Robe Rental website on a desktop computer
   {Display Garment}
2. System displays landing page that has a link to the “How it works” page
3. User clicks on “How it works” link
   {Display ‘How it works’ page}
4. Website displays FAQs on how Robe Rental works
5. User closes website

Goal 3 Alternate Flows

Alternate Flow - Network disconnects during access

At any point between **{Display Garment}** and **{Display ‘How it works’ page}**
If network connectivity is lost
The system displays “Unable to connect” message
Resumes basic flow at **{Display Garment}**

Goal 4 Basic Flow: *User login*

1. User opens Robe Rental website on a desktop computer
   {Display Garment}
2. System displays landing page with option to Sign In
3. User clicks on Sign In link
   {Display Sign In page}

4. System displays Sign In page
5. User enters credentials

   {Authentication Services}

6. System authenticates login information
7. User updates profile information

   {User Profile Services}

8. System saves updated profile
9. User logs out

Goal 5 Alternate Flows

Alternate Flow 1 - Network disconnects

At any point between **{Display Garment}** and **{Update Profile Services}**

If network connectivity is lost

The system displays “Unable to connect” message

Resumes basic flow at **{Display Garment}**

Alternate Flow 2 - Authentication failure

At **{Authentication Services}** if authentication has failed

The system displays “Authentication Failed” message

Resumes basic flow at **{Display Garment}**

Goal 6 Basic Flow: *Create a profile*

1. User opens Robe Rental website on a desktop computer

   {Display Garment}

2. System displays landing page with option to Sign In
3. User clicks on Sign In link

   {Sign In Services}

4. System displays Sign In page with an option to Sign up
5. User clicks Signup button
6. User enters account information

   {User Account Services}

7. System encrypts and saves account information
8. User logs out

Goal 6 Alternate Flows

Alternate Flow 1 - Network disconnects

At any point between **{Display Garment}** and **{User Account Services}**

If network connectivity is lost

The system displays “Unable to connect” message

Resumes basic flow at **{Display Garment}**

Alternate Flow 2 - Incomplete information

At **{User Account Services}** if missing any user account sign up information

The system displays "Please fill out missing information" message

Resumes basic flow at **{Sign In Services}**

Goal 7 Basic Flow: *Upload garments as a user to rent out*

1. User opens Robe Rental website on a desktop computer

   {Display Garment}
2. System displays landing page with option to Sign In
3. User clicks on Sign In link

   {Sign In Services}
4. System displays Sign In page
5. User enters email ID and password

   {Authentication Services}
6. System authenticates profile
7. User uploads images of garment along with price, description, material, care to be taken and size of the garment

   {Garment Services}
8. System saves the garment information
9. User logs out

Goal 7 Alternate Flows

Alternate Flow 1 - Network disconnects during a transaction

At any point between **{Display Garment}** and **{Garment Services}**

If network connectivity is lost

The system displays "Unable to connect" message

Resumes basic flow at **{Display Garment}**

Alternate Flow 2 - Incomplete garment details

At **{Garment Services}** if missing any garment details

The system displays "Please fill out missing information" message

Resumes basic flow at **{Authentication Services}**

Alternate Flow 3 - Incompatible garment images

At **{Garment Services}** if garment image incompatible with database

The system displays "Please upload only JPEG images" message

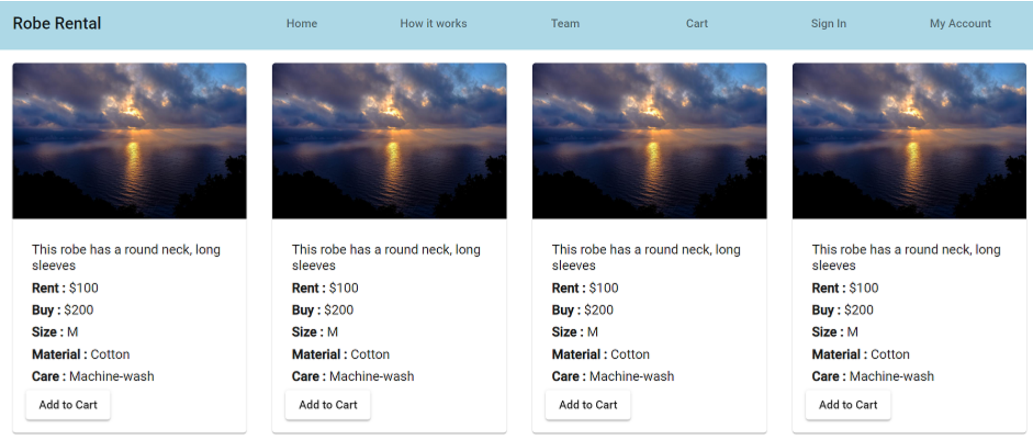
Resumes basic flow at **{Authentication Services}**

4. Technology

System at the end of this iteration

- Goals for this iteration :
  - Working frontend and backend using MVC architecture.
  - Implementing sign in and sign up functions that binds the backend with frontend.
  - Store the password and email address in database.
  - Do the proper encryption for the passwords that are going to be stored.

- **Home** Page for the website :



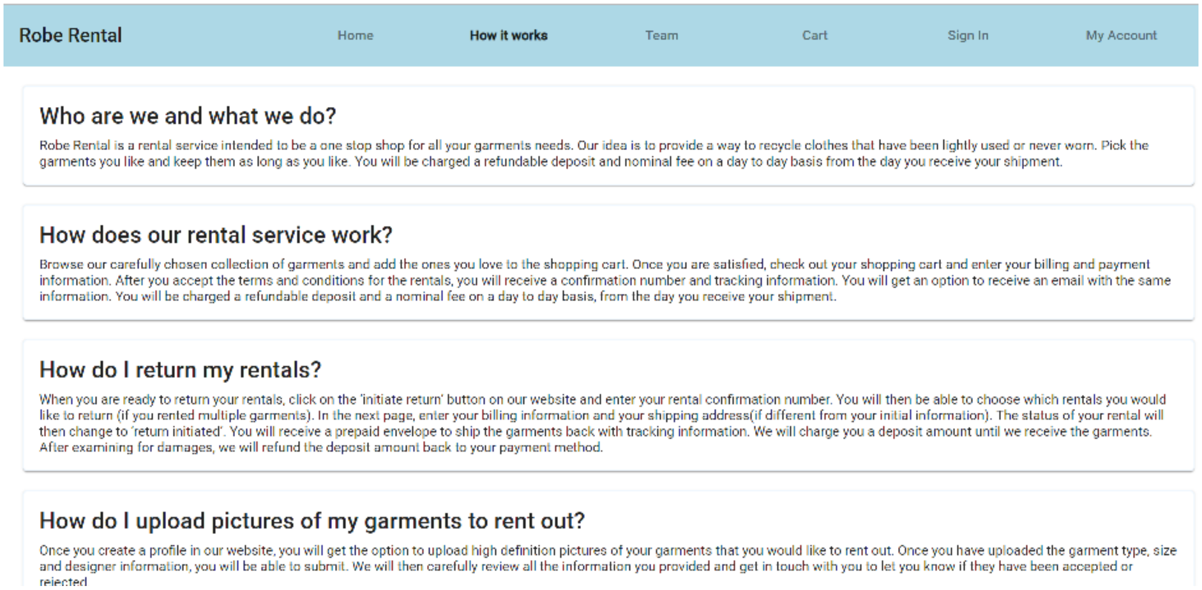
This robe has a round neck, long sleeves
Rent : $100
Buy : $200
Size : M
Material : Cotton
Care : Machine-wash

This robe has a round neck, long sleeves
Rent : $100
Buy : $200
Size : M
Material : Cotton
Care : Machine-wash

This robe has a round neck, long sleeves
Rent : $100
Buy : $200
Size : M
Material : Cotton
Care : Machine-wash

This robe has a round neck, long sleeves
Rent : $100
Buy : $200
Size : M
Material : Cotton
Care : Machine-wash

- **How it works** Page for the website :



Who are we and what we do?

Robe Rental is a rental service intended to be a one stop shop for all your garments needs. Our idea is to provide a way to recycle clothes that have been lightly used or never worn. Pick the garments you like and keep them as long as you like. You will be charged a refundable deposit and nominal fee on a day to day basis from the day you receive your shipment.

How does our rental service work?

Browse our carefully chosen collection of garments and add the ones you love to the shopping cart. Once you are satisfied, check out your shopping cart and enter your billing and payment information. After you accept the terms and conditions for the rentals, you will receive a confirmation number and tracking information. You will get an option to receive an email with the same information. You will be charged a refundable deposit and a nominal fee on a day to day basis, from the day you receive your shipment.

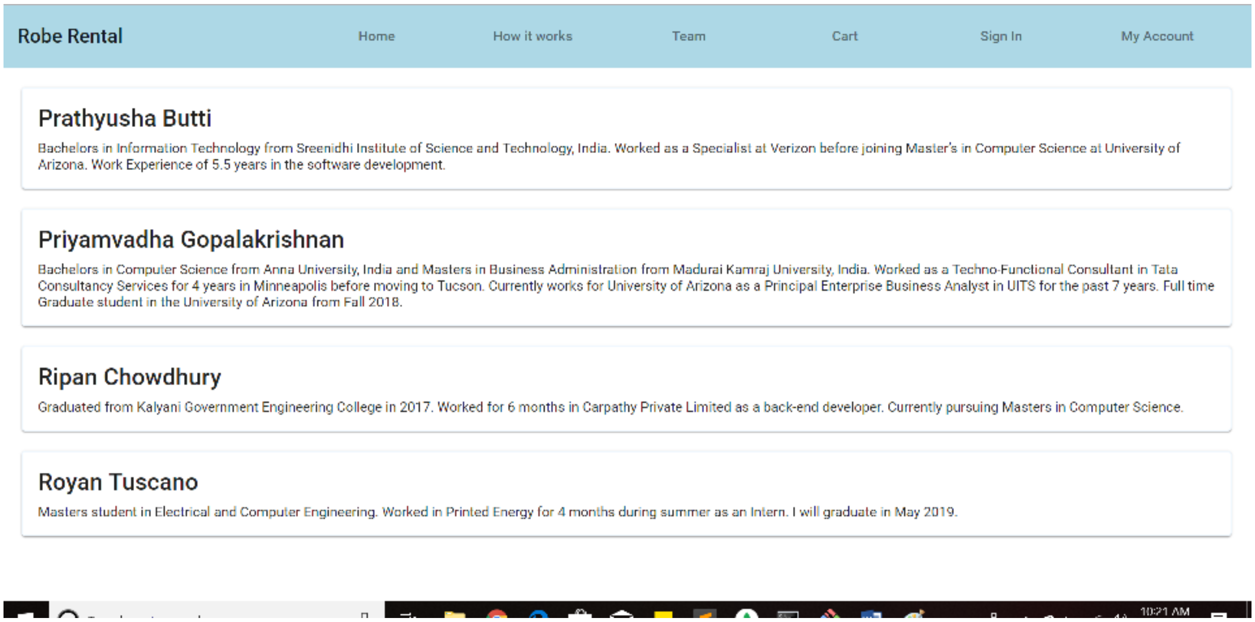
How do I return my rentals?

When you are ready to return your rentals, click on the 'initiate return' button on our website and enter your rental confirmation number. You will then be able to choose which rentals you would like to return (if you rented multiple garments). In the next page, enter your billing information and your shipping address(if different from your initial information). The status of your rental will then change to 'return initiated'. You will receive a prepaid envelope to ship the garments back with tracking information. We will charge you a deposit amount until we receive the garments. After examining for damages, we will refund the deposit amount back to your payment method.

How do I upload pictures of my garments to rent out?

Once you create a profile in our website, you will get the option to upload high definition pictures of your garments that you would like to rent out. Once you have uploaded the garment type, size and designer information, you will be able to submit. We will then carefully review all the information you provided and get in touch with you to let you know if they have been accepted or rejected

- **Team** Page for the website:



Prathyusha Butti

Bachelors in Information Technology from Sreenidhi Institute of Science and Technology, India. Worked as a Specialist at Verizon before joining Master's in Computer Science at University of Arizona. Work Experience of 5.5 years in the software development.

Priyamvadha Gopalakrishnan

Bachelors in Computer Science from Anna University, India and Masters in Business Administration from Madurai Kamraj University, India. Worked as a Techno-Functional Consultant in Tata Consultancy Services for 4 years in Minneapolis before moving to Tucson. Currently works for University of Arizona as a Principal Enterprise Business Analyst in UITS for the past 7 years. Full time Graduate student in the University of Arizona from Fall 2018.

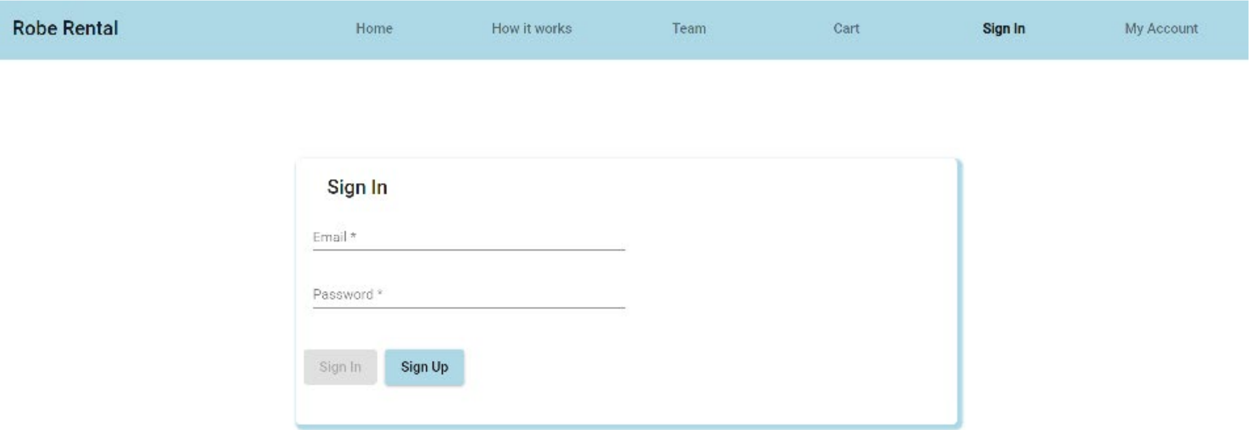
Ripan Chowdhury

Graduated from Kalyani Government Engineering College in 2017. Worked for 6 months in Carpathy Private Limited as a back-end developer. Currently pursuing Masters in Computer Science.

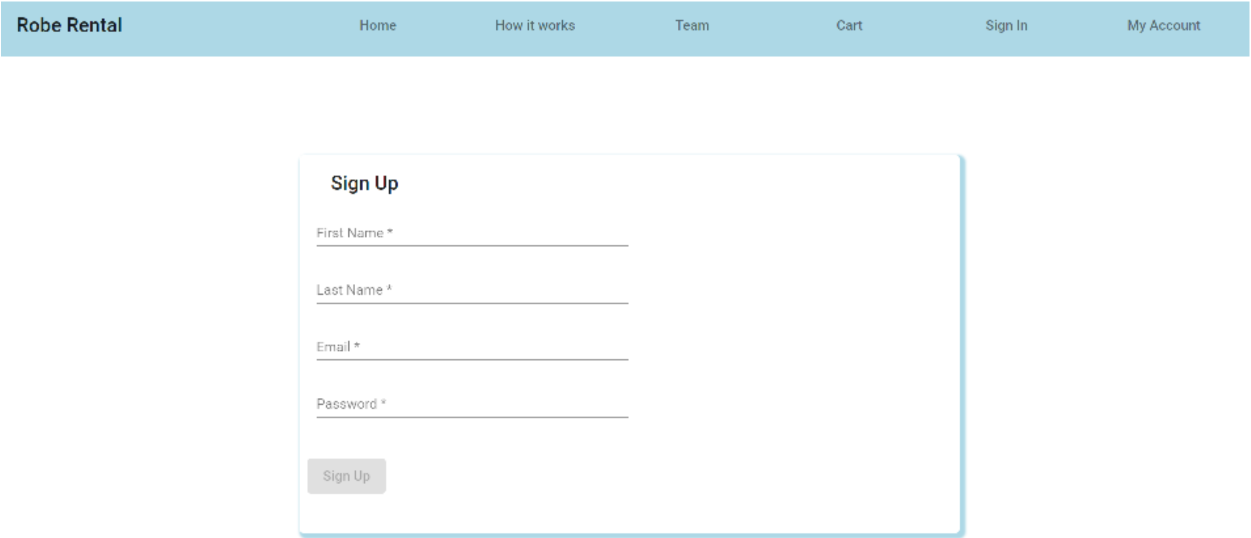
Royan Tuscano

Masters student in Electrical and Computer Engineering. Worked in Printed Energy for 4 months during summer as an Intern. I will graduate in May 2019.

- **Sign In** Page for the website:



- **Sign Up** Page for the website:



- What tests have you run?
  - We have run manual test cases for the system as mentioned in the google sheets.
    - https://docs.google.com/spreadsheets/d/1WHFJXMZ_ISyXC3QH-vWQMHL-1JhGgMryJ12aWaAxIz0/edit?usp=sharing

  - We have covered the following scenarios
    - User signs up in the website
    - User signs in to the website
    - Team page access
    - How it works page access
    - Home page display

- How many lines of code have you written as a team?
  - The database server has around 200 lines of code.
  - The front-end has around 1000 lines of code.

5. Team

- Team member roles and contributions for this iteration :
  - **Prathyusha** - As a Product Owner, she has prioritized which user stories needs to be implemented so that a minimum viable system is up and running. As Front-end developer she learnt new technologies like Angular 7, Angular Material and TypeScript. She has developed the entire front-end of the minimum viable system. She has contributed around 45% for this iteration.
  - **Priyamvadha** - As a Scrum Master, she has conducted daily scrums. As a Manual Tester, she has written and executed the test cases for the first iteration. She has contributed around 20% for this iteration.
  - **Ripan** - As a Back-end developer he needs to develop the GET API to display the list of garments in the home page. But he could not succeed in doing the task. His contribution for this milestone is 0%.

- **Royan** - As a Back-end developer he learnt new technologies like ExpressJS, NodeJS, MongoDB and successfully developed the sign in and sign up GET and POST API's for the first iteration. He even encrypted the passwords to secure them. He has contributed around 35% for this iteration.

6. Project Management

- We plan to implement the following for the next iterations :
  - Second Iteration (4/01) : Login Module, GET API for the list of garments for the home page, Create a profile for the user, categorize the garments along with respective images, Upload the image of the garment and it's specifications.
  - Final Iteration (4/22) : Payment module.

- Track changes to design, test cases and code :
  - We are using Google Docs to maintain the API contract.
    - The link to the API contract is as follows: https://docs.google.com/document/d/1hRxrKzGn-Iy87juDzL9V_BBJbUYksuX6qmn9i8vHpVQ/edit?usp=sharing
  - We are using Google Sheets to maintain the test cases and its results.
    - The link to the test cases sheets is as follows: https://docs.google.com/spreadsheets/d/1WHFJXMZ_ISyXC3QH-vWQMHL-1JhGgMryJ12aWaAxIz0/edit?usp=sharing
  - We are using GitHub to track the changes to the code
    - The link to the GitHub for both front-end and back-end code is as follows: https://github.com/buttiprathyu/roberental

7. Reflection

- What went well?

  - We were able to implement the Sign in and Sign up functionality as we had imagined with the setup.
  - We implemented the Team and How it works page as expected.
  - We implemented the Home page layout as expected.

- What didn't go well?

  - We weren't able to implement the GET API for list of garments to be displayed for the home page. So we hard-coded the list of garments for this iteration.
  - Communication with some of the team members was not upto the mark.

- For the features that were planned but not implemented, what were the issues?

  - The GET API for list of garments was not implemented. One of the developers faced some difficulties in learning the new technology in short span of time.

- How do you plan to overcome the issues you encountered?

  - We need to learn and imbibe the new technologies faster and improve our skills further in the new technologies to progress with our project.
  - Communication between the team members needs to be more streamlined. We plan to conduct a google hangouts voice conference call every day for the daily scrum instead of group chat to know their status.

- Redefining the problem - **"No changes"**

- What will you do differently for the next iteration?

  - We will have to coordinate and communicate the tasks amongst us better than the first iteration by letting the team know well in advance whether they can achieve the task.
  - We will have to make it a practice to check in the code regularly to coordinate the deployment of the builds.
  - We even plan to deploy our website to the AWS in the next iteration.