

SEPT Project Report

Group: 5.WED-16.30-7

Group members:

Sean Tan (s3806690)

Isuru Wijesinghe (s3719421)

Gabriel Lat (s3723189)

Table of Contents

Vision Statement.....	3
System architecture	4
Gitflow Organization.....	5
Scrum Process	6
Enhancements	7
Acceptance Tests Documentation	9

Vision Statement

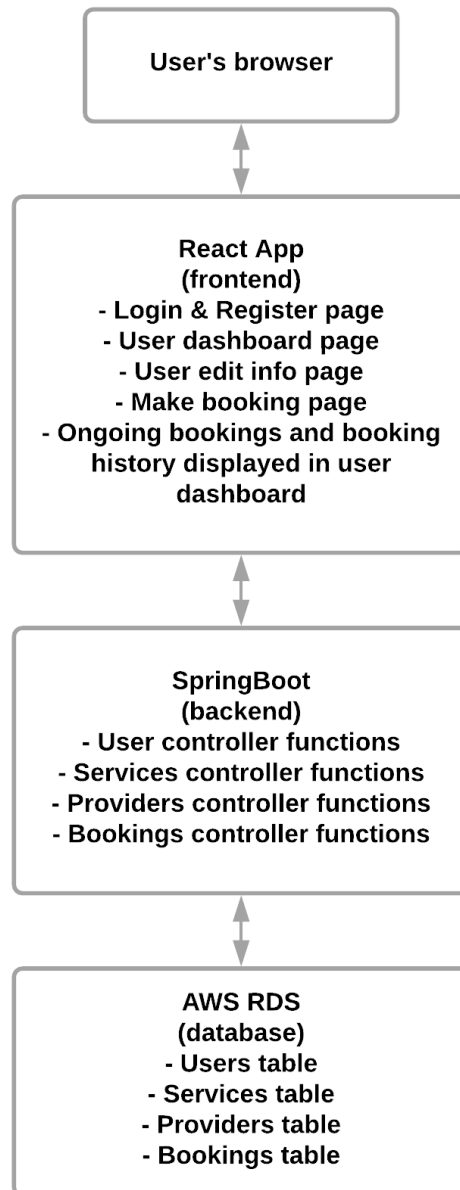
For AGME with the product owner being our tutor Mohamad Ali, who requires a platform that allow service providers to offer their services and users to book and use those services, the product they desire is a website that allows users and services providers to login and use those features.

In the duration of 3 milestones on building our AGME project, we have managed to deliver the following features:

1. Users are able to register and login into their accounts
2. Users are able to add and edit their account information which includes address, contact, etc.
3. Users each have a custom dashboard which will display their booking information
4. Users are able to make a booking for services and choose the specific providers of their choice
5. Users are able to see their ongoing bookings and its status
6. Users are able to see their booking history

With the features listed above, we believe that our features provided for AGME's website certainly matches their requirements and thus is valuable as a product for AGME.

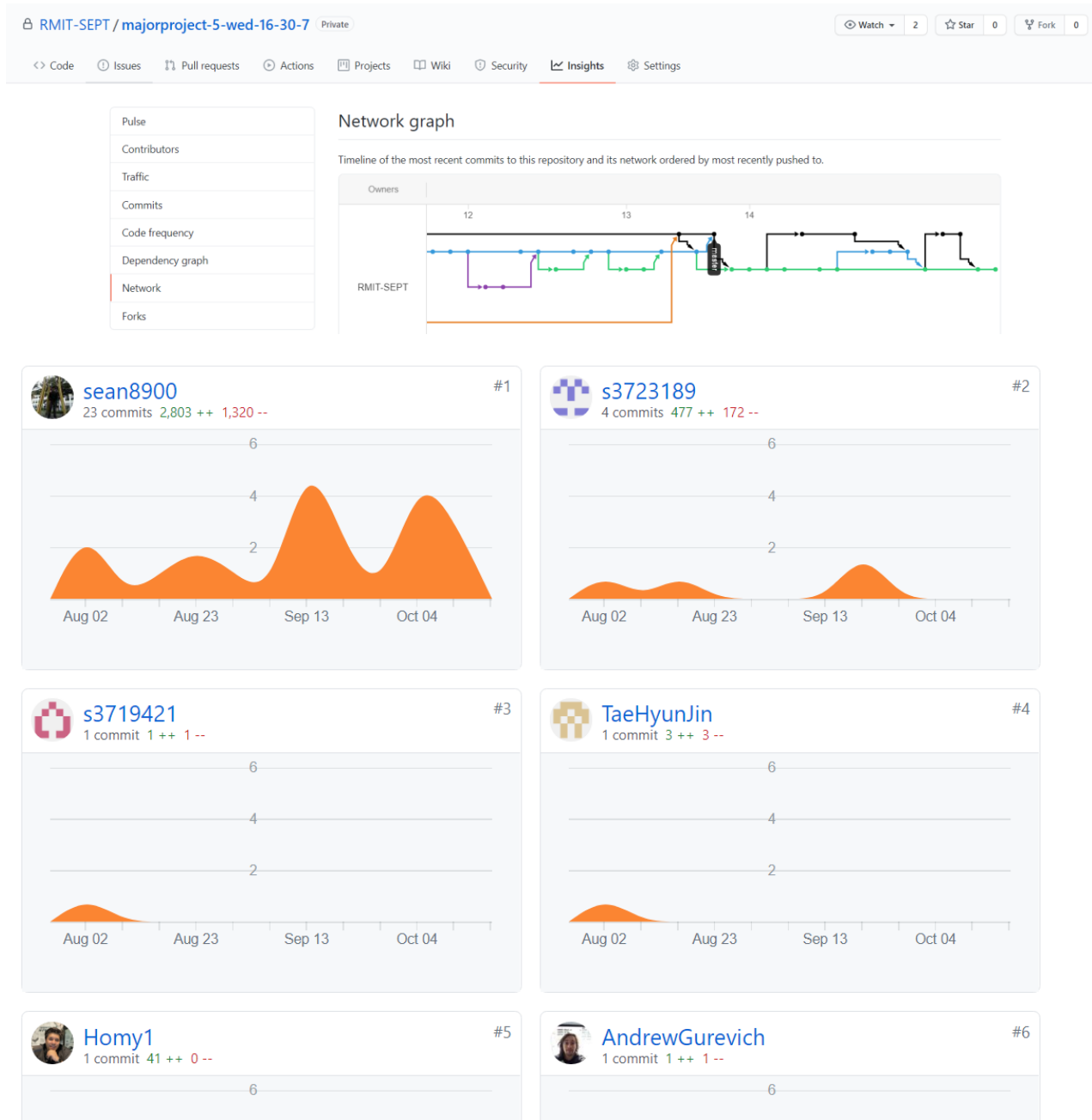
System architecture



In our system, the user interacts with our frontend React App which contains features for the user such as login, register, making a booking. The React App which acts as a consumer to SpringBoot would then use POST and GET methods available in the controller functions in SpringBoot to get, create or alter information in the database which is hosted on AWS with their RDS (relational database service).

Gitflow Organization

On GitHub, we would organize features into different segments, creating a new feature branch using Gitflow for each feature that needs implementation, and then committing it back to our “develop” branch for testing. We would only merge “develop” into “master” for deployment during our Milestone submissions. We commit into our repository very often, including when working on the feature branches to improve our teamwork. Unfortunately, not all team members are contributing equally (this issue is already raised previously) thus I would hope we are not penalized for the inactive group members.



Scrum Process

The Scrum Master for our group is Sean Tan who would be organizing meeting times and well as facilitating the discussion topics. The scrum members are Isuru Wijesinghe and Gabriel Lat with Mohamad Ali being our product owner. We averagely meet 4 times per sprint, 1 for sprint planning, 1 for sprint retro, 2 for stand-up meetings.

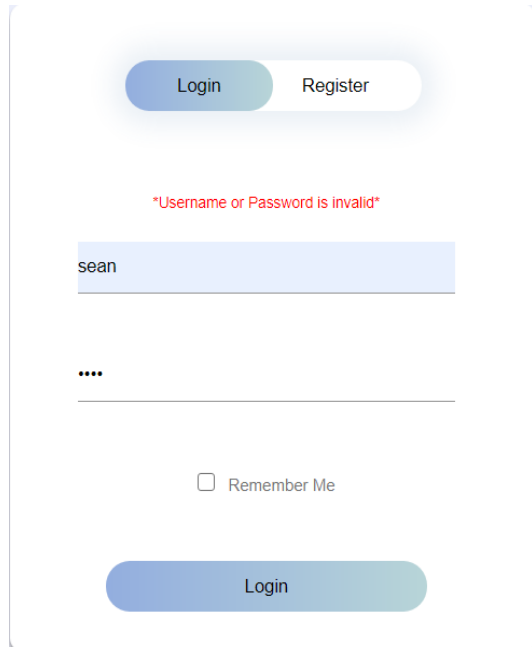
Recordings of our stand-up meetings are as below:

1. <https://web.microsoftstream.com/video/47eb24fe-e98d-4fd4-a7aa-fa89aa64b0f8>
2. <https://web.microsoftstream.com/video/3e8385b3-3f1e-48b2-9cb7-9db9f7d289cf>
3. <https://web.microsoftstream.com/video/f594d372-be82-4307-8147-3f6689182331>
4. <https://web.microsoftstream.com/video/c0604cab-8170-4c4d-8531-2e688716aada>
5. <https://web.microsoftstream.com/video/c0604cab-8170-4c4d-8531-2e688716aada>
6. <https://web.microsoftstream.com/video/0f7d84f5-062a-412b-af03-490a7b416520>
7. <https://web.microsoftstream.com/video/206ab515-9919-4d42-9e7a-c3597cc9cc20>
8. <https://web.microsoftstream.com/video/ead9625e-e2ae-44ef-ac8b-ab604c78a28b>

Enhancements

We have implemented the below enhancements:

- Login verification
- Messages for different invalid inputs during register (register verification)



The image shows a login form with the following elements:

- A toggle switch with 'Login' (active, blue) and 'Register' (inactive, white) buttons.
- An error message: ***Username or Password is invalid*** in red text.
- A username input field containing the text 'sean'.
- A password input field with masked characters '....'.
- A checkbox labeled 'Remember Me'.
- A 'Login' button at the bottom.

For our login feature, we will verify the login inputs to make sure the user had valid credentials.

<div>LoginRegister</div>	<div>LoginRegister</div>
Full Name	Sean
Address	melbourne
Phone Number	0123456789
User Name	sean
Password	..
Confirm Password	..
Register	Register

For our register feature, we will verify the username, password and the confirm password field, and display an error message accordingly.

Acceptance Tests Documentation

We have not implemented all of our user stories as we have created more user stories than our team could actually handle. We started out having 5 team members in total when creating the user stories, but now we only have 2 members with active contribution. Thus, we would only be discussing the acceptance tests of the user stories we have successfully implemented.

1. User Story: "As a customer I want to be able to log in so that I can access my dashboard"

Test: for Success

ID	1
Purpose	Test login customer with valid username and password
Set Up	Customer enters valid username and password
Steps	1. Navigate to login page
	2. Enter correct username and password
	3. Click login button
Expected Result	System successfully logs the customer in and redirects to their dashboard.

Our user database contains the following information:

id	name	username	password	account_type	address	phone	created_At	updated_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26	NULL

We enter the correct username and password in the login page and the user is redirected to their dashboard.

The image shows two screenshots of a web application. The left screenshot is the login page, featuring a 'Login' button, a 'Register' button, a text input field containing 'sean1234', a password input field with masked characters, a 'Remember Me' checkbox, and a 'Login' button at the bottom. The right screenshot is the 'MY DASHBOARD' page. It includes a sidebar with navigation icons, a main content area with a 'TODAY'S SCHEDULE' section showing a service for 'Crazy Curles' at 1:16:19 PM, an 'UPCOMING BOOKINGS' section with a list of appointments (Hairdresser, Gym, Hairdresser, Dentist) and 'Cancel' buttons, and a 'PAST BOOKINGS' section with a table of completed services.

Service Provider	Service	Status	Date
Crazy Curles	Hairdresser	completed	October 17th 2020, 12:05:50 pm
Crazy Curles	Hairdresser	completed	October 18th 2020, 12:05:50 pm
Crazy Curles	Hairdresser	completed	October 19th 2020, 12:05:50 pm

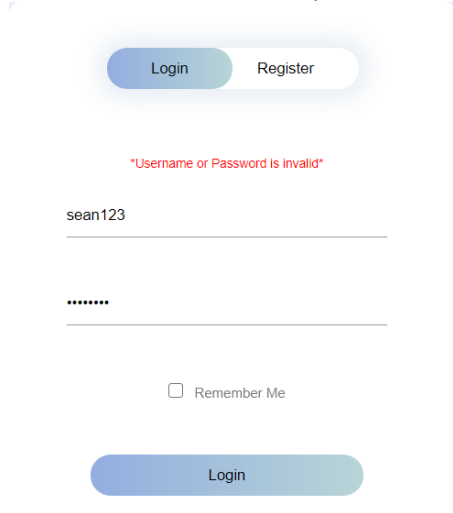
2. **User Story:** “As a customer I want to be able to log in so that I can access my dashboard”

Test: for Failure

ID	1
Purpose	Test login customer with invalid username or password
Set Up	Customer enters an invalid username or password
Steps	1. Navigate to login page
	2. Enter invalid login details
	3. Click login button
Expected Result	System displays the message that either username or password entered is invalid and stays on the login page.

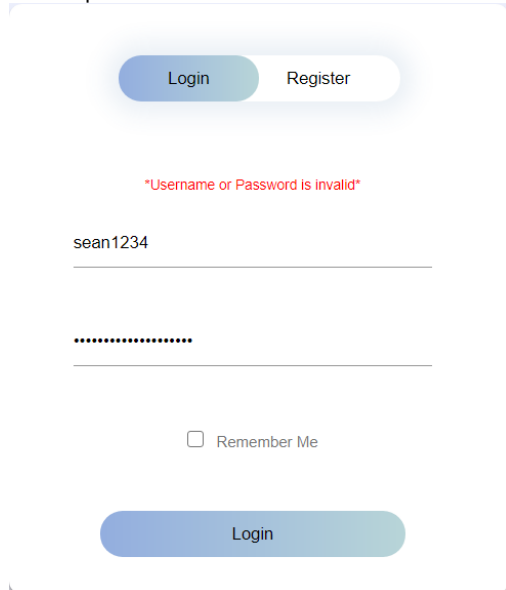
With the same information in our user database, we enter an invalid username or password, and the system displays “Username or Password invalid”

Invalid username correct password:



The screenshot shows a login interface with a light blue header containing 'Login' and 'Register' buttons. Below the header, a red error message reads '*Username or Password is invalid*'. The username field contains the text 'sean123' and the password field is masked with dots. Below the password field is a 'Remember Me' checkbox. At the bottom is a large blue 'Login' button.

Invalid password correct username:



The screenshot shows the same login interface as the previous one. The red error message '*Username or Password is invalid*' is displayed. The username field contains the text 'sean1234' and the password field is masked with dots. Below the password field is a 'Remember Me' checkbox. At the bottom is a large blue 'Login' button.

3. **User story:** “As a customer, I want to register an account, so that I can use the services provided”

Test: for Success

ID	3
Purpose	Test the register customer account with non-existing username
Set Up	A user called “sean12345” does not exist in the system
Steps	1. Select register new customer account
	2. Enter “sean12345” as username, other values for other fields
Expected Result	System reports that the new customer account is successfully registered, the new user is then added into the system

Our user database contains the following information:

id	name	username	password	account_type	address	phone	created_At	updated_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26	NULL

When we register with username “sean12345” and other values for other fields, the user is successfully created in our database:

LoginRegister

Sean

melbourne

0123456789

sean12345

.....

.....

Register

id	name	username	password	account_type	address	phone	created_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26
3	Sean	sean12345	11111111	c	melbourne	0123456789	2020-10-18 02:32:04

4. **User story:** “As a customer, I want to register an account, so that I can use the services provided”

Test: for Failure

ID	3
Purpose	Test the register customer account with non-existing username
Set Up	A user called “sean1234” already exist in the system
Steps	1. Select register new customer account
	2. Enter “sean1234” as username, other values for other fields
Expected Result	System reports that the username already exist, the new user is not added into the system

Our user database contains the following information:

id	name	username	password	account_type	address	phone	created_At	updated_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26	NULL

When we register with username “sean1234” and other values for other fields, the user is not created in the database:

LoginRegister

Sean

melbourne

0123456789

sean1234

Register

id	name	username	password	account_type	address	phone	created_At	updated_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26	NULL

5. User story: “As a customer, I want to update my personal information, so that my account has my updated and accurate personal information”

Acceptance tests:

ID	4
Purpose	Test customer update personal information feature with valid inputs
Set Up	A customer account is set up and logged in
Steps	1. Select update personal information in customer’s dashboard
	2. Enter valid name, username and address
Expected Result	System reports that the information is successfully updated and the new information will be updated in the system

Our user database contains the following information:

id	name	username	password	account_type	address	phone	created_At	updated_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26	NULL
2	Isuru	isuru123	12341234	c	melbourne	0123456789	2020-10-14 06:53:26	NULL

Since sessions was not successfully implemented in our project, we have manually pick the user id at the top right corner. Thus, we pick user id 2 which is the user “Isuru”, we then modify his name, username and address.

Before edit:

localhost:3000/edit

2

Isuru

melbourne

0123456789

isuru123

.....

Edit

After edit:

2 ▾

Isuru 2

sydney

0123456789

isuru222

Edit

After pressing “Edit” button, we can see in our database now that the data of user id 2 had been changed successfully, it also shows the timestamp it was edited at:






id	name	username	password	account_type	address	phone	created_At	updated_At
1	Sean Tan	sean1234	12345678	c	melbourne	0123456789	2020-10-14 06:53:26	NULL
2	Isuru 2	isuru222	12341234	c	sydney	0123456789	NULL	2020-10-18 0


6. User story: As a customer, I want to be able to easily make reservations so that I can make appointments

ID	5 (Customer make a booking)
Purpose	Test that the customer can select a timeslot of the selected provider to make a reservation
Set up	A provider already exists in the system, there are available time slots of the chosen provider
Steps 1	Select a timeslot
2	Fill in appropriate details if necessary
3	Click confirm
Expected Result	Reservation/booking is accepted and system reports success; reservation is added

The user “Sean” now has 2 upcoming bookings:

MY DASHBOARD



TODAY'S SCHEDULE

Service Name
Employee Name
Time of Appointment

CURRENT TIME
2:06:41 PM

UPCOMING BOOKINGS

Hairdresser Crazy Curls	October 15th 2020, 12:05:50 pm	Cancel
Gym Fitness First	October 16th 2020, 12:05:50 pm	Cancel
Dentist Dental Group	October 20th 2020, 12:00:00 am	Cancel

The user then proceeds to make a booking with the service being “Dentist” with the provider “Dental Group” and date being 20th October.

MAKE YOUR RESERVATION

We provide the best services available by liscened experts in their field.

SERVICE:

Dentist | v

PROVIDER:

Dental Group | v

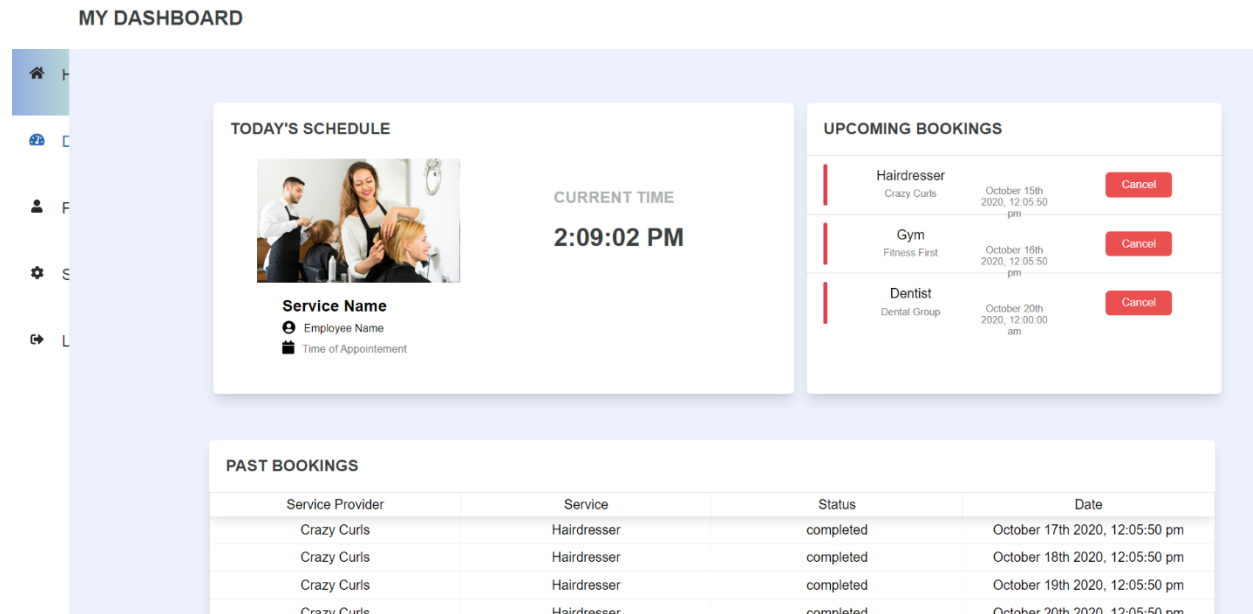
DATE/TIME:

October 20, 2020 12:00 AM

[Make Booking](#)

[Cancel](#)

After the user clicks on “Making Booking”, the user is directed back to his dashboard and it shows the new booking in his upcoming booking list:



7. **User story** : “As a Customer, I want to be able to view my dashboard after login, so that I can see my booking history, any upcoming bookings.”

Test: Success

ID	16
Purpose	Test that the dashboard is being rendered to the customer without any components missing
Set Up	Customer has to be logged in
Steps	1. Customer logs in into their account
Expected Result:	The customer should be redirected to their dashboard displaying all the components (booking history, upcoming bookings)

Our booking table in the database contains the following information, which are bookings under the user_id 1:

booking_id	booking_date	provider_id	service_id	status	user_id
1	2020-10-15 01:05:50	1	1	ongoing	1
2	2020-10-16 01:05:50	5	5	ongoing	1
3	2020-10-17 01:05:50	1	1	completed	1
6	2020-10-20 01:05:50	1	2	completed	1
8	2020-10-22 01:05:50	1	2	cancelled	1

In our services table shown below, we can see that service id 1 and 2 is Hairdresser and id 5 is Gym:






service_id	provider_id	service_name
1	1	Hairdresser
2	2	Hairdresser
3	3	Hairdresser
4	4	Dentist
5	5	Gym


In our provider table shown below, we can see that provider id 1 belongs to Crazy Curls and id 5 is Fitness First:

provider_id	provider_name
1	Crazy Curls
2	Bed Head
3	Good Cuts
4	Dental Group
5	Fitness First



In user_id 1's account dashboard, we can see that the information displayed is all accurate and correct:

MY DASHBOARD



TODAY'S SCHEDULE

CURRENT TIME
2:18:20 PM

Service Name
 Employee Name
 Time of Appointment

UPCOMING BOOKINGS

Hairdresser Crazy Curls	October 16th 2020, 12:05:50 pm	Cancel
Gym Fitness First	October 16th 2020, 12:05:50 pm	Cancel

PAST BOOKINGS

Service Provider	Service	Status	Date
Crazy Curls	Hairdresser	completed	October 17th 2020, 12:05:50 pm
Crazy Curls	Hairdresser	completed	October 20th 2020, 12:05:50 pm
Crazy Curls	Hairdresser	cancelled	October 22nd 2020, 12:05:50 pm