



**CSCI 5308 – Advanced Topics in Software Development**

**Group 18 – Budget Surfing**

**Project Report**

**Work done by,**

- 1. Parth Dhirubhai Gondaliya, B00893694**
- 2. GRM Vishnu, B00871849**
- 3. Hardee Rameshchandra Garala, B00869195**
- 4. Tuan Hamid, B00877065**
- 5. Neelansh Gulati, B00892946**

## **Table of Contents**

1. Overview	-----	03
2. Flowchart of the project	-----	04
3. Website Page	-----	05
4. Sign Up Page	-----	07
5. Login Page	-----	10
6. User Profile Page	-----	11
7. Traveler Home Page	-----	14
8. Add Trip Page	-----	18
9. Add Budget Page	-----	20
10. Add Category Page	-----	22
11. Add Expense Page	-----	24
12. All Trips Page	-----	26
13. Single Trip Info Page	-----	28
14. Payment Page	-----	39
15. Place Detail Page	-----	42
16. Host Detail Page	-----	44
17. Host Home Page	-----	45
18. Host Profile Creation Page	-----	46
19. Available Trips Page	-----	49
20. Info about Traveler Page	-----	51
21. Trips Accepted by Host Page	-----	53
22. Feedback Page	-----	55

## **Overview:**

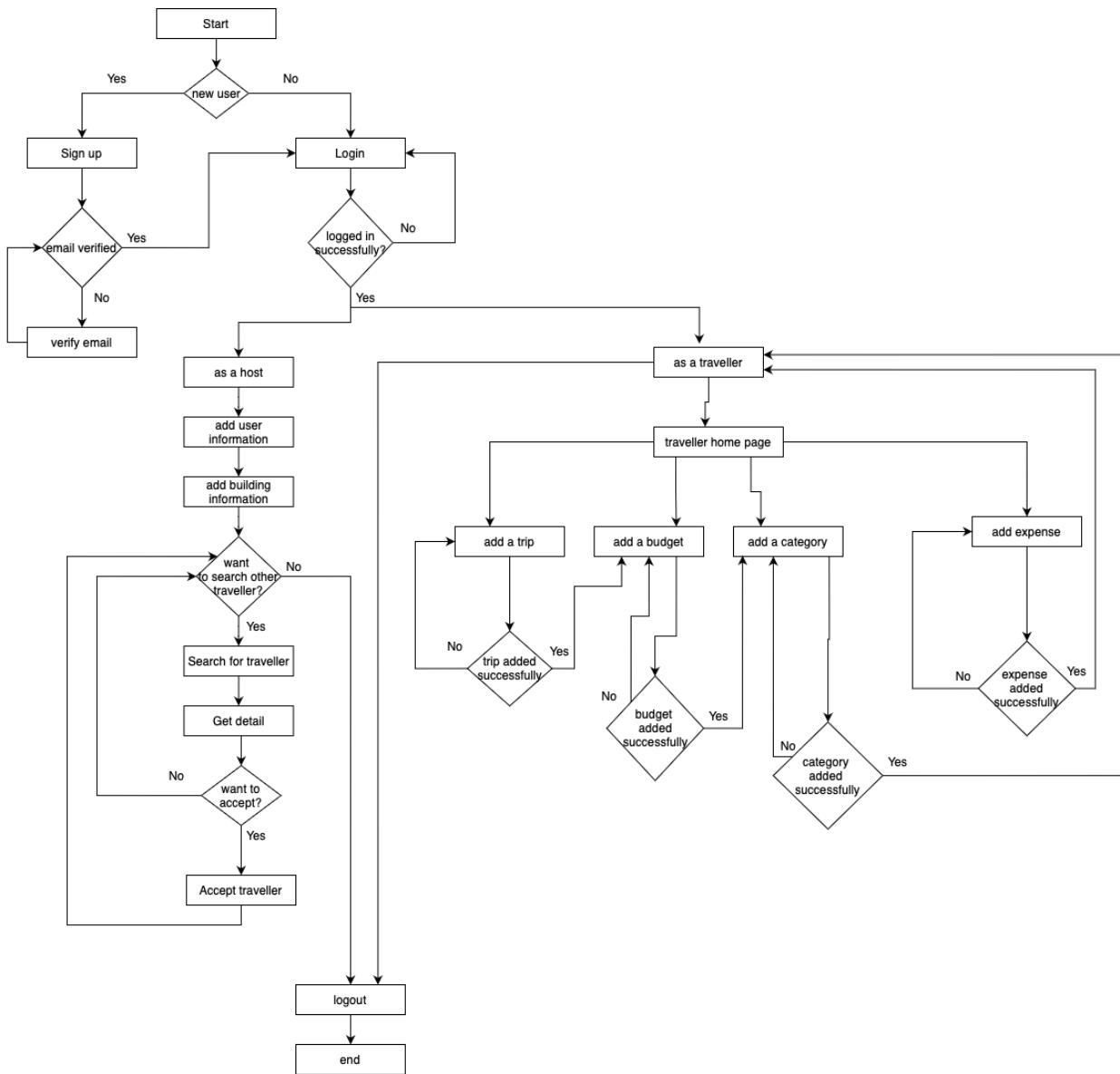
The project is all about the finding the trip as traveler and accepting the trip as host in all over the world. Any user can be a traveler and Host for renting their place and services. The core idea of project is maintaining the trip and managing the budget and expense. If anyone wants to be a traveler, then he/she needs to sign up and login into the system. If user wants to be a traveler, they can add the trips (city, country, date of trip etc.) and can make the request for the host.

As a traveler, I can add number of trips along with single budget associated with it. Moreover, to keep track of all the expense of trips user also can add the expenses with category wise. The interesting part here is user can select their own category while creating the trips. User can edit and delete the expense and as there is chance that they have added by mistake.

As host, the user can see the whole request from others traveler in all over the world and can select the traveler for their trip. User also can search the traveler based on city or country. Every host has the power to rate the traveler for once and based on that rating other host can get the idea of selecting the traveler. The Rating for the user has been calculating using total rating till now divided by total number of rated trips and based on that part each user will be rated.

We are sending the mail on exceeding budget of traveler, signup confirmation and acceptance of request by host to notify the user with updated information in their account.

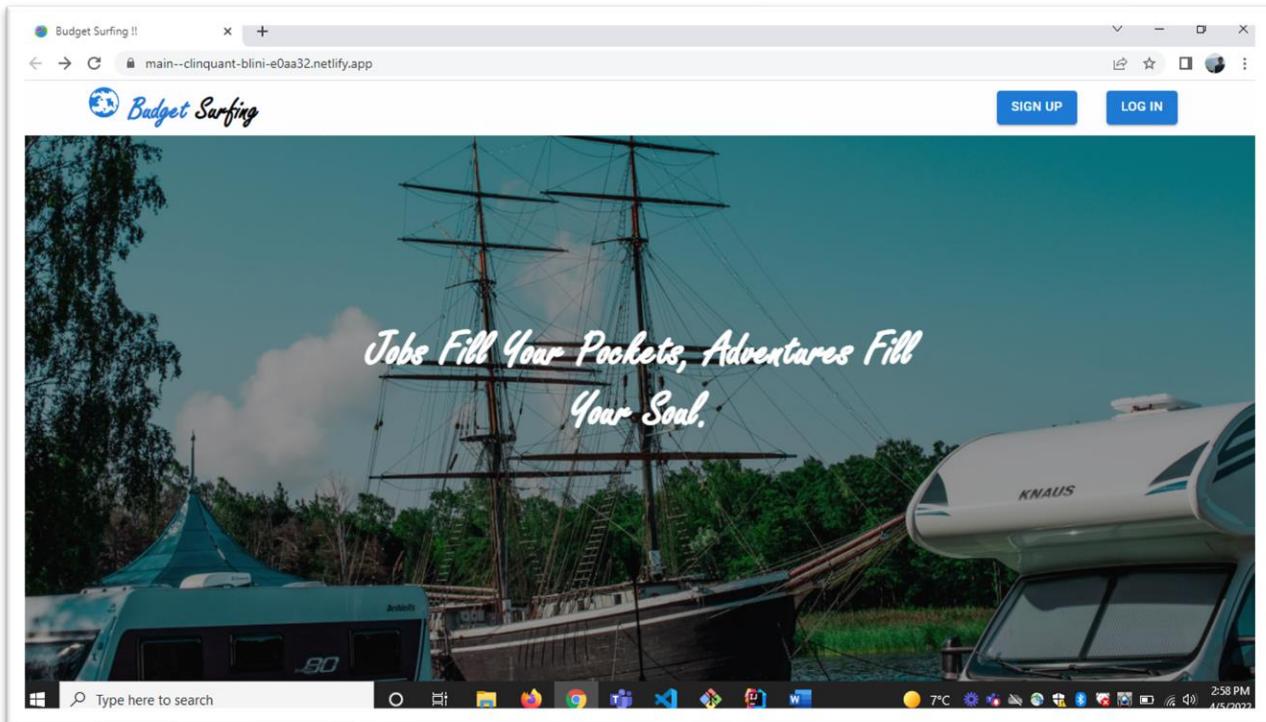
## Flowchart:



The above flowchart has been described and the flow of the process has been discussed below in the respective pages in detail.

## Website Page: (/)

This is our website page for our application. Any user who wants to use our application should signup or login through this page. A glimpse of this page is provided in the diagram below.



This page contains a header and a body with an image and text including in it. The logo of our application and the name of our application is present on the top left side of the header, while there are two buttons on the top right side of the header through which users are navigated into respective pages.

The buttons and their purposes are listed below as,

- Signup

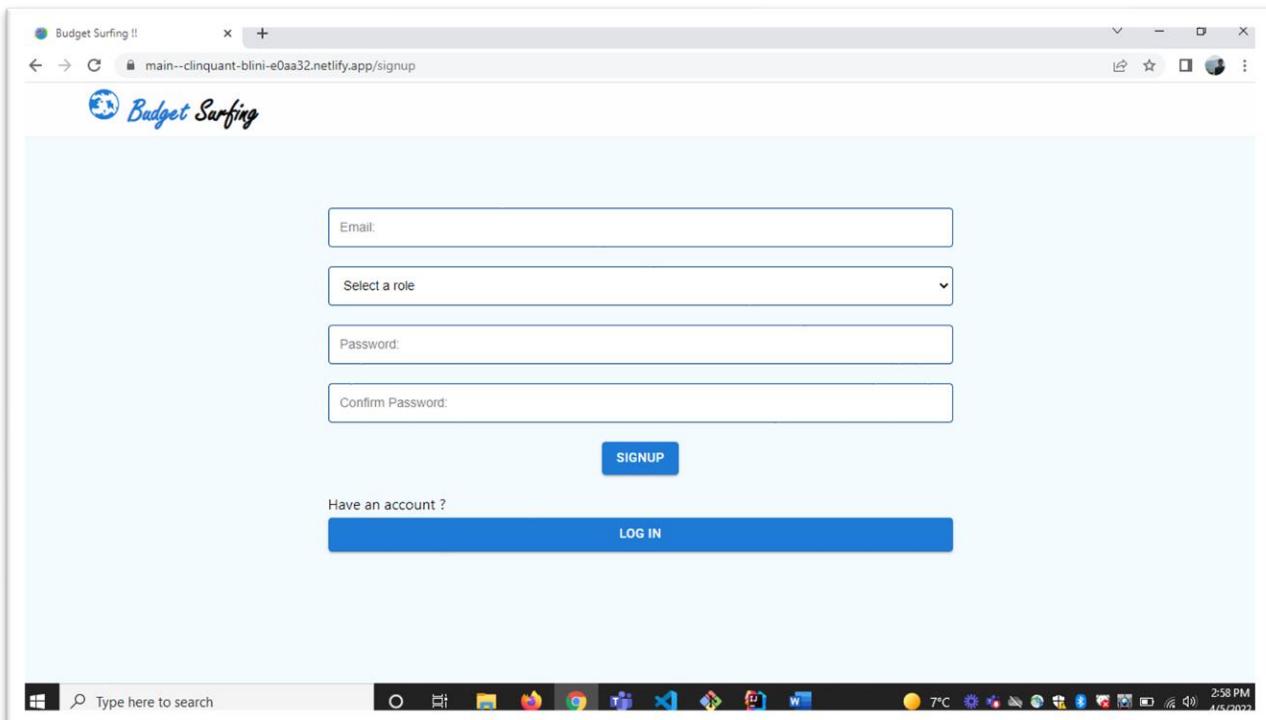
If a new user who is using our application for the first time wants to get in, then they must click the "Signup" button and they will be redirected to the page which allows them to sign up into our database.

- Login

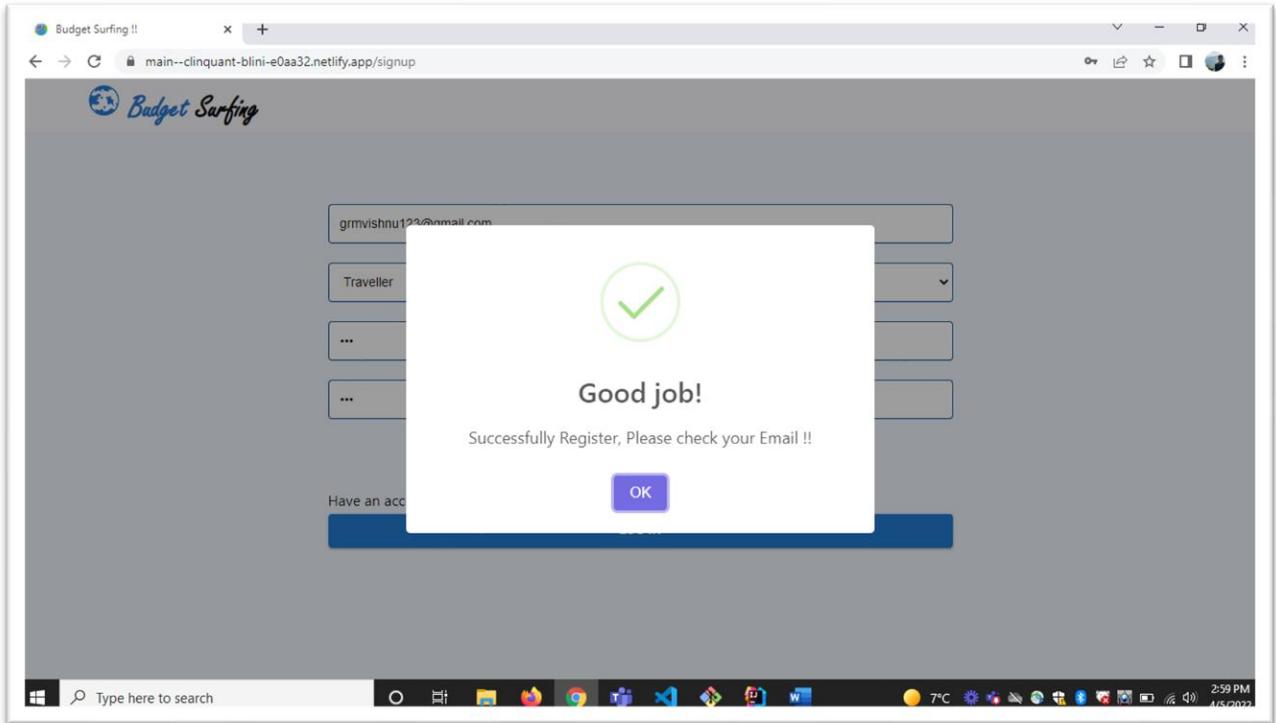
If an existing user wants to login into our application, then they have to click “Login” button and they will be prompted into the respective login page.

## Sign Up: (/signup)

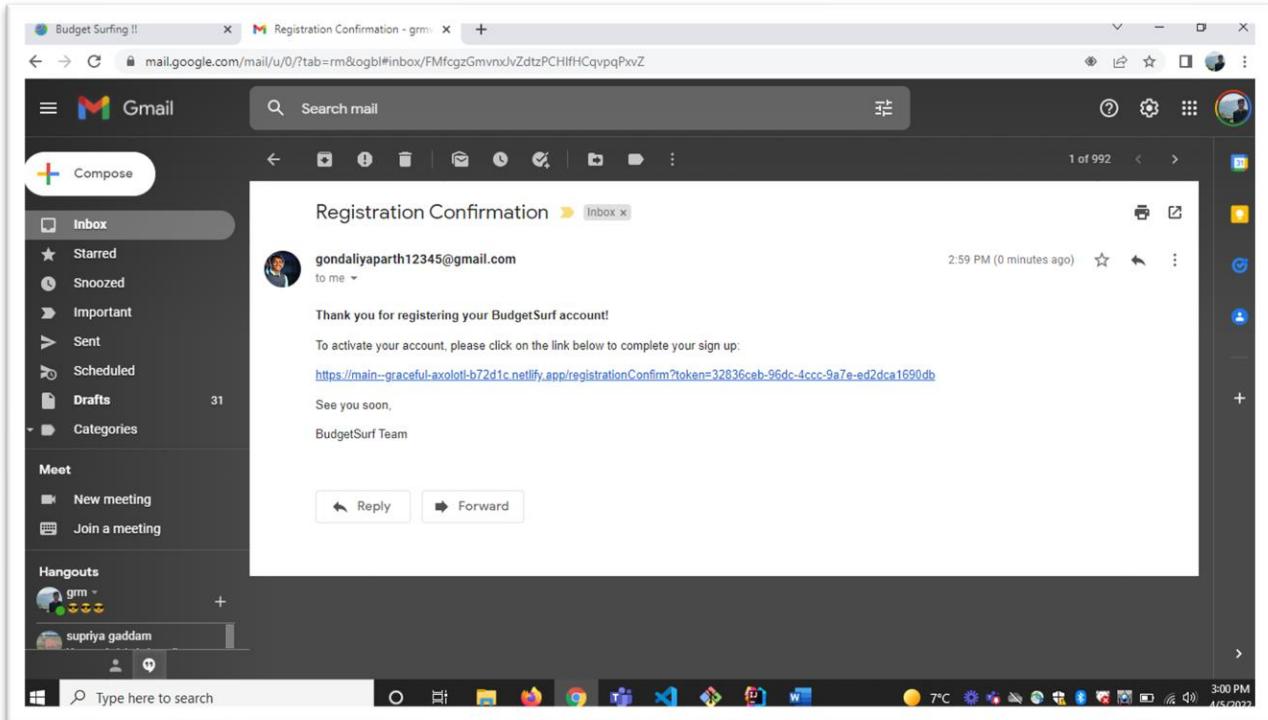
When a new user wants to use our application either as a traveler or a host, the first thing they have to do is to sign up, in other words, register in our database. This page lets the customers do that! Screenshot of that page is given below.



The sign-up form consists of a few fields out of which all of them are required necessarily. A user cannot sign up without filling all the fields. If any of the fields are not filled, then the application will give an error that the field should be filled before submitting the form. After filling the fields with the correct email address, selecting the role as they want to register, and setting up a password, the user has to click the "SignUp" button.



When the user successfully registers, the application will pop a message as shown to the user if the details are as required as by the application. Since it gave a successful pop-up, you might be tempted to think that the sign-up process might have been completed. It's not!!!

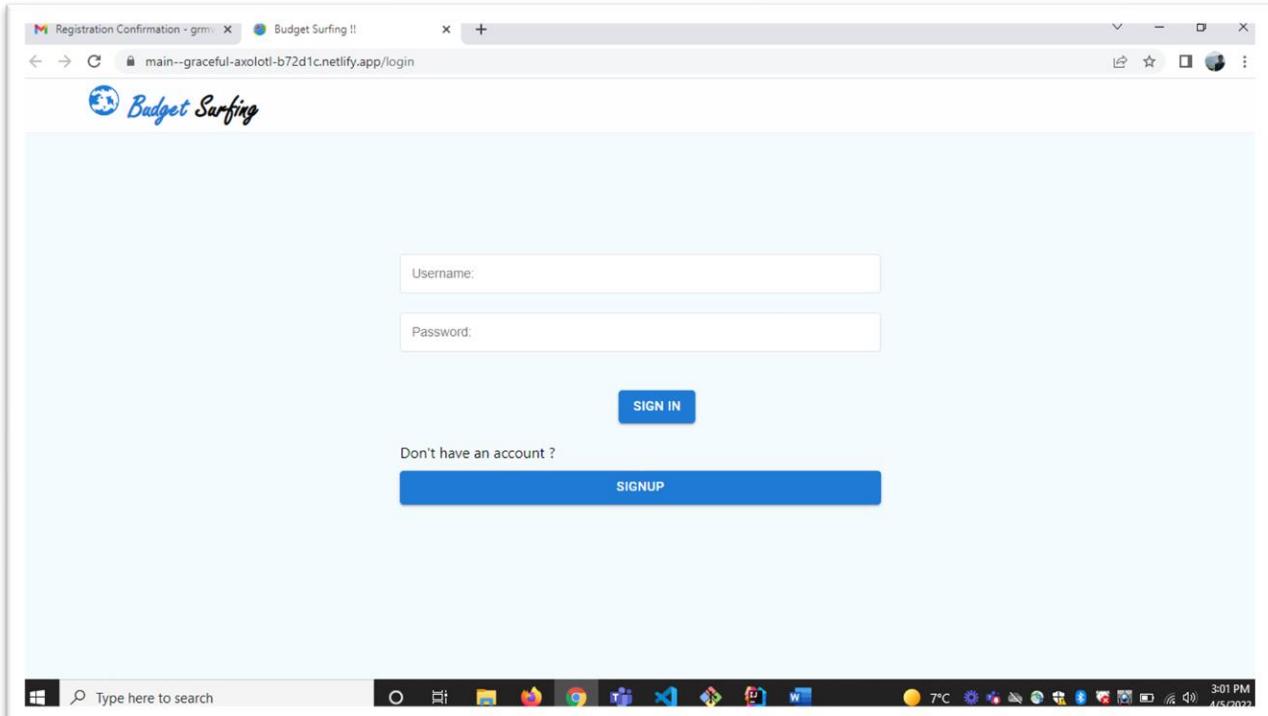


To protect our application from fraudsters and to check that the email address given by any user is valid or not, we do the email verification, and we will add the user into our database only when the user has completed the email verification.

Our main aim is to maintain the privacy of our users as much as possible. And in order to achieve that, we store the details of the user id and password very securely in our database. The password entered by the user will be hashed in the backend of our application and then the hashed password will be stored in the database. This way, even if we lose the details of our users, we can secure those details of the user.

## Login: (/login)

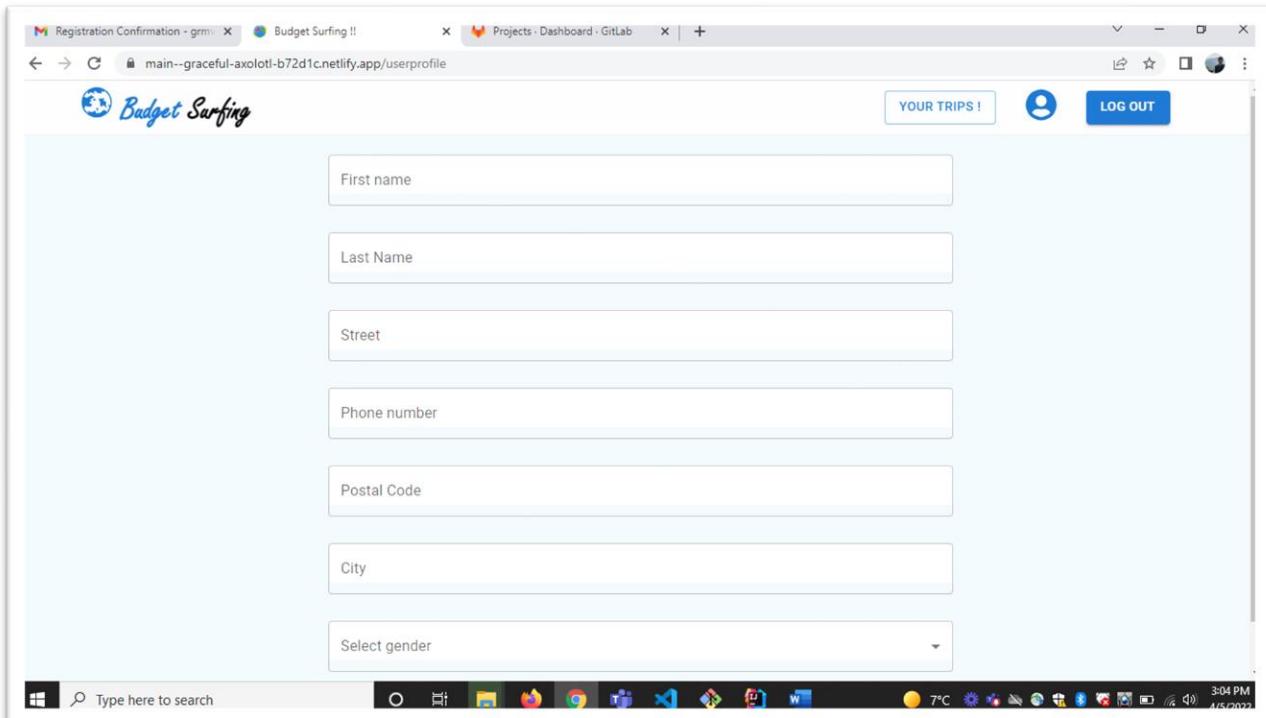
When an existing user wants to use our application either as a traveler or a host, after registering into our database, they would have to login to use the application. This page lets the customers do that! Screenshot of that page is given below.



As you might have guessed, the login page has a form which has only two fields: the username and password. If the username and password are matched with each other and if they are matched with the data we have in our database, it means that the details are correct, and the user is a valid user. After the user is successfully authenticated, the application then navigates the user to the respective page.

## User Profile: (/userprofile)

When a new user logs in after registration, they will be automatically redirected to the user profile page where they would have to fill a form about themselves. The form can be visible down here.



This form is nothing but to take all the required details of the user in order to proceed further. The details we are taking here are solely for the purpose of smooth operations in our application, and nothing else. As we can see the fields in this form, all the fields are a necessity to be filled and after meeting all of them, the user can proceed by clicking the “Save” button.

Registration Confirmation - grm... X Budget Surfing !! X +

main--graceful-axolotl-b72d1c.netlify.app/userprofile

**Budget Surfing**

YOUR TRIPS ! LOG OUT

First name: GRM

Last Name: Vishnu

Street: 2309 Brunswick Street

Phone number: 9024128921

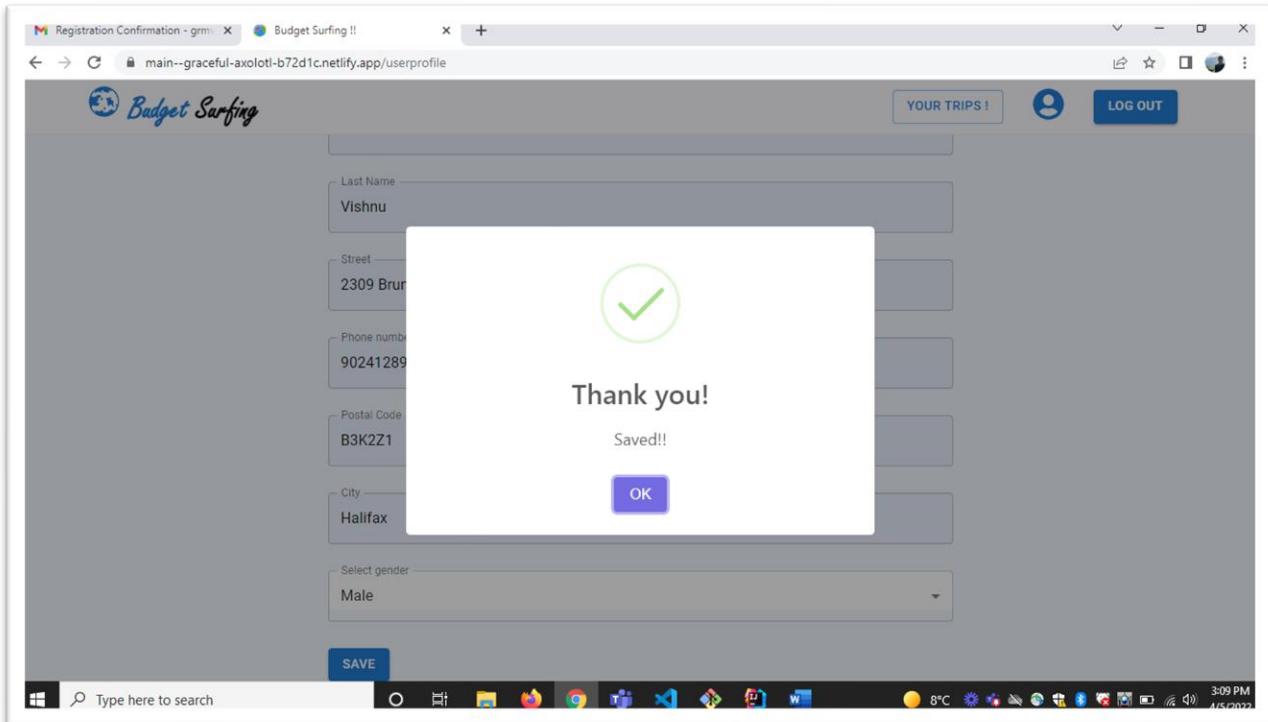
Postal Code: B3K2Z1

City: Halifax

Select gender: Male

Type here to search

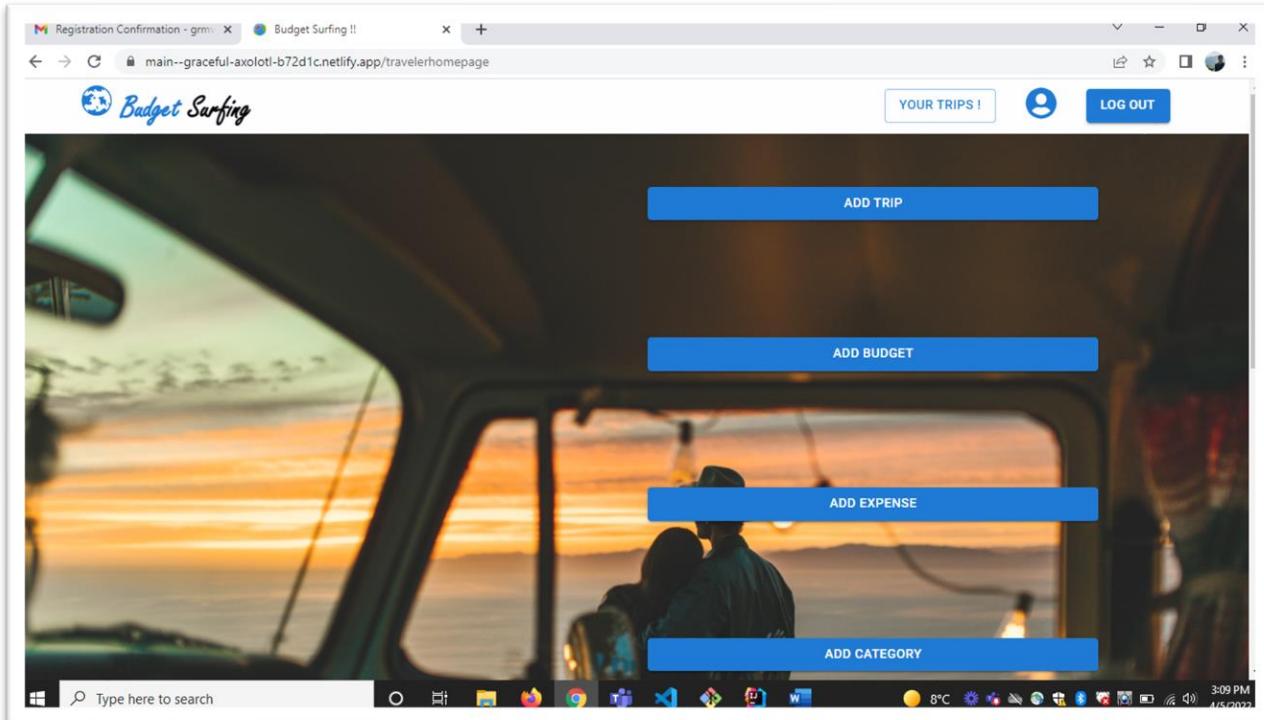
3:08 PM 4/5/2022

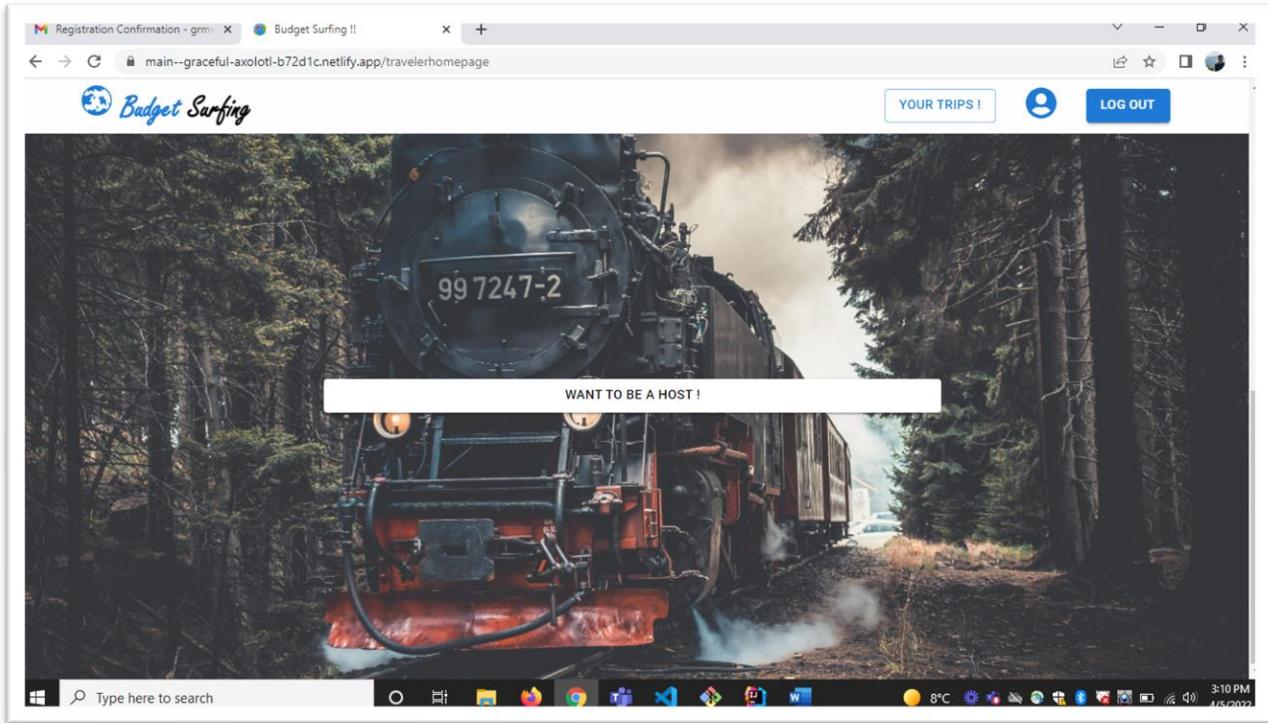


If the details are saved successfully in the database, the application will pop a message like this to let the user know that they can move on now into the real application.

## Traveler Home Page: (/travelerhomepage)

When an existing user successfully fills the user data or when a new user successfully completes their registration and logs in, by default, they are navigated to the traveler home page. This is the home page which any user (if they are a traveler) mostly gets around with. A glimpse of this page is provided below.





This page consists of a header and a body. The header has a logo with the application name on the top left-hand side, whereas the top right-hand side of the header contains a couple of buttons and an icon. The header is going to be common for all the pages in our website from this point. The body is divided into two parts where one part of the body contains an image, and the other side of the body contains few buttons as a list parallel to the image. And if we scroll down a little bit, we will find another image which has a button over it with a completely different purpose. The list of the buttons and their use will be discussed below.

- Your trips

This button is on the top right part of the header in this page, and this allows you to look at all the trips a traveler has had and all the required details of every particular trip.

- User icon

This is nothing but the user profile navigation icon. The icon here navigates you to the page where you can view and edit your user profile and save the changes.

- Add trip

When a user clicks this button, it allows the user to add a new trip to post into the searching platform where hosts can search the trip and will be able to accept the trip.

- Add budget

When a trip is added by a user, it is mandatory for the user to add the budget for that trip. That can be done using the add budget page and this button in the home page will redirect the user to that page.

- Add expense

If a user starts their trip and wants to add any expense which they feel should be counted in, to keep track of all the expenses, then this button navigates them to the page which lets them do the exact thing.

- Add category

After adding a budget to a trip, before adding the expenses, the user should add categories in which the budget is going to be divided into. Clicking onto this button allows you to a form which adds the category into a particular trip and particular budget.

- Want to be a host!

Our application can be divided into two halves': traveler and host. If the page we are in completely is dedicated for a traveler, then the other half of our application lies behind this button. Any person who wants to be a host and wants to start

looking for people and accept trips, they can do all those and become a host by going through this channel.

- Logout

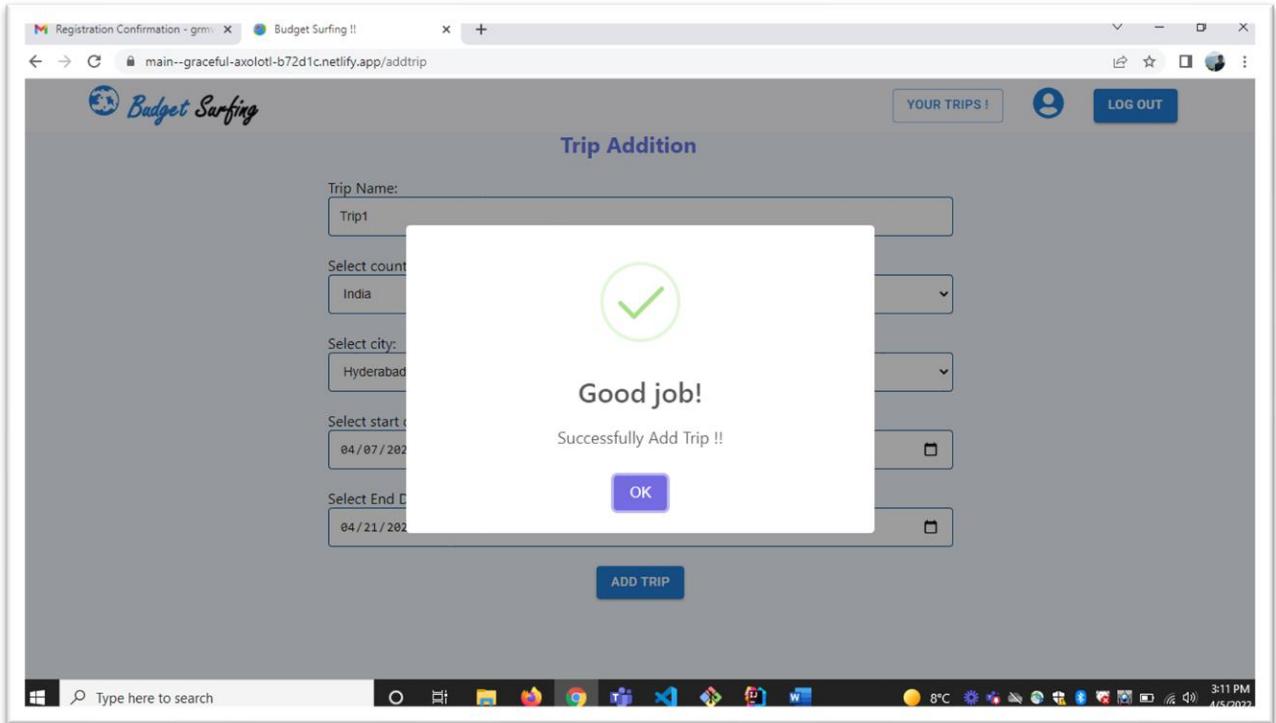
As we all know that in general, logout button lets you to exit the application, this logout button in the header also lets you do the same thing as expected.

## Add Trip Page: (/addtrip)

This is the page every traveler would be in the most need of. How the page looks will be shown as below.

The screenshot shows a web browser window with the title 'Budget Surfing !!'. The main content area is titled 'Trip Addition'. It contains five input fields: 'Trip Name:' (with placeholder 'Trip Name:'), 'Select country:' (with placeholder 'India' in a dropdown menu), 'Select city:' (with placeholder 'Ahmedabad' in a dropdown menu), 'Select start date:' (with placeholder 'mm/dd/yyyy' in a text input field), and 'Select End Date:' (with placeholder 'mm/dd/yyyy' in a text input field). Below these fields is a blue 'ADD TRIP' button. At the top right of the page are links for 'YOUR TRIPS!', a user profile icon, and 'LOG OUT'. The bottom of the screen shows a Windows taskbar with various pinned icons and system status information.

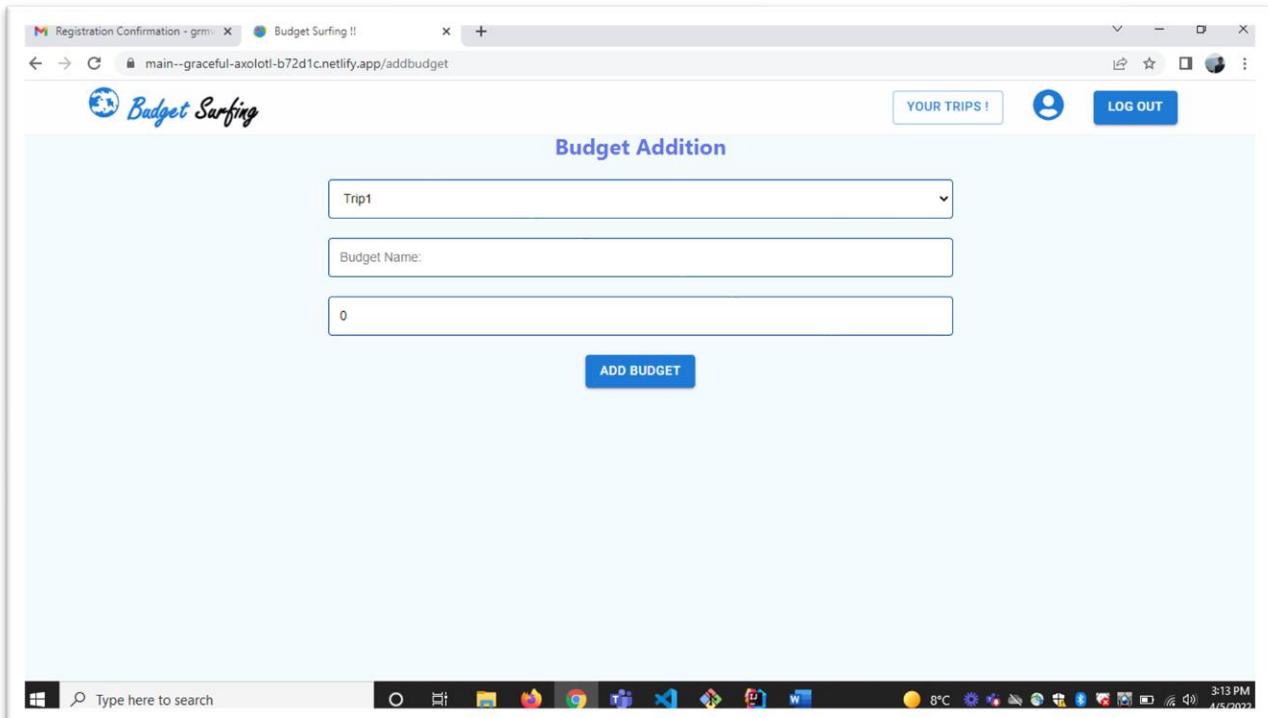
If the traveler wants to travel to a particular city or country and is looking for any host to accept their request, then they have to add the trip through this form so that it can show in the hosts timeline. The user has to enter all the details like the trip name they want to give to the trip, destination country, destination city, start date of the trip and end date of the trip. The user can only select the start date and end date of the trip ahead of the present date. The user can either type the start and end dates with keyboard or can select them from the calendar which pops up if the calendar icon is pressed.



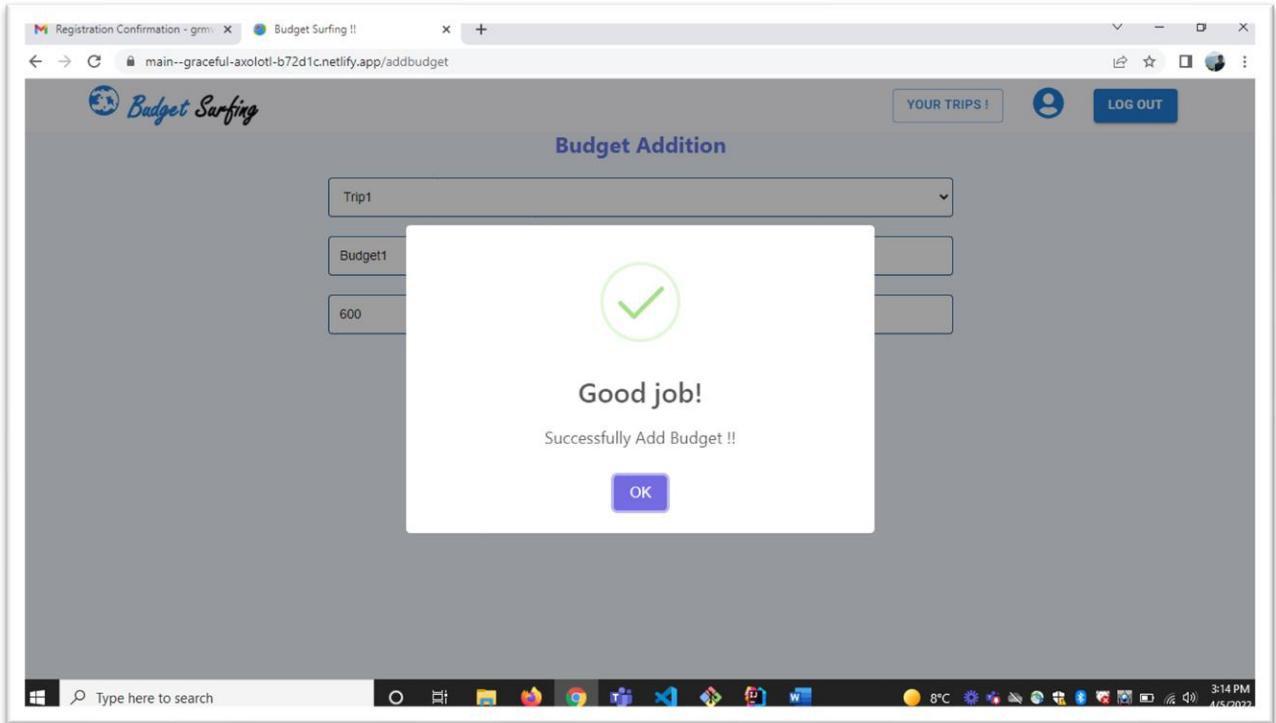
After all the fields are filled, the user clicks the “Add Trip” button which adds the trip details into our database to show to the hosts. To let the traveler know that it has been successfully saved, we give a pop-up like above.

## Add Budget Page: (/addbudget)

When the user successfully adds the trip, our application will redirect them to the add budget page where they have to set a budget for their trip. It can be seen as here.



Budget is nothing but the maximum limit of amount they are planning to spend on their trip. The add budget page has a small form which consists of the few fields. User has to select the trip in which they want to add their budget in, and after selecting the trip they have to give the budget a name and put the maximum amount they want to spend. After entering these details, the user will add the budget into the database by clicking the “Add Budget” button.



To let the traveler know that it has been successfully saved, we give a pop-up like above.

If the user doesn't want to add the budget right now for that trip, they can get back from this page and later when they decide to add the budget, they can do that by going into traveler home page and with the help of "Add Budget" button.

## Add Category Page: (/addcategory)

After adding the budget in the trip, traveler will be redirected to the page which looks like the below one.

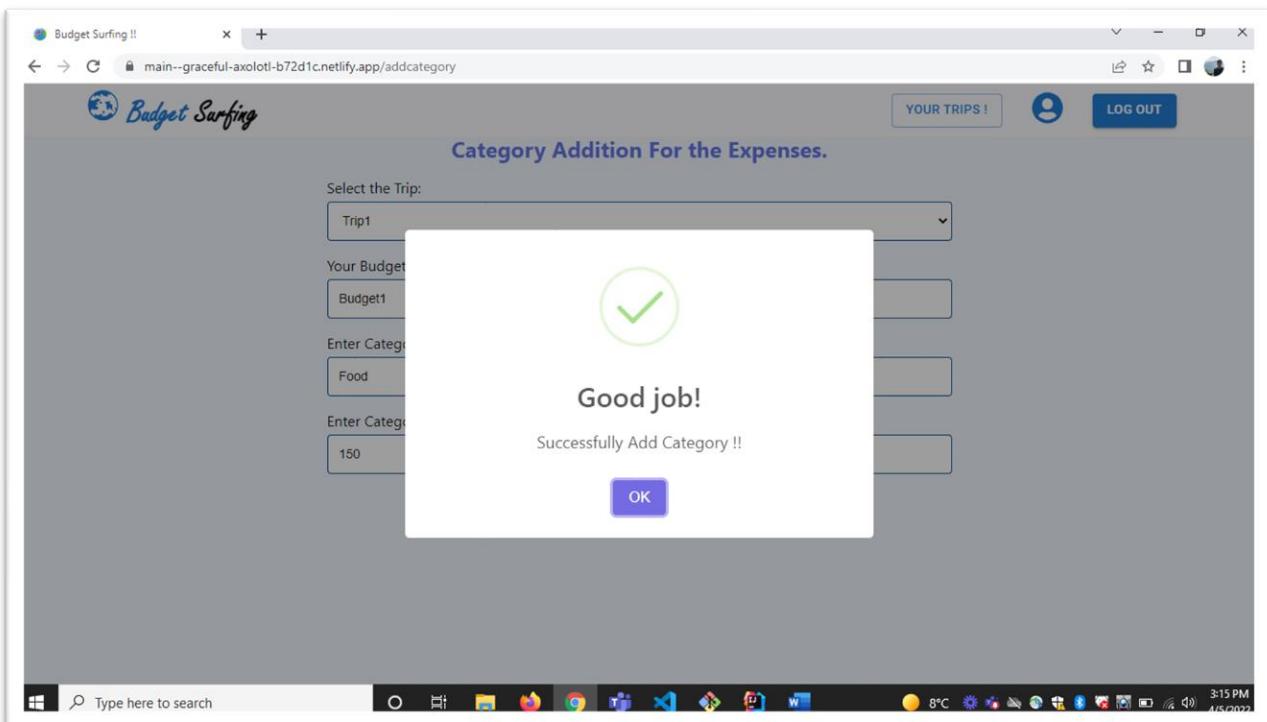
The screenshot shows a web browser window titled "Budget Surfing !!". The URL in the address bar is "main--graceful-axolotl-b72d1c.netlify.app/addcategory". The page has a header with the "Budget Surfing" logo, a "YOUR TRIPS!" button, and a "LOG OUT" button. The main content area is titled "Category Addition For the Expenses.". It contains four input fields: "Select the Trip:" (dropdown menu showing "Trip1"), "Your Budget:" (text input showing "Budget1"), "Enter Category name:" (text input showing "Category Name:"), and "Enter Category Amount:" (text input showing "Category Amount:"). Below these fields is a blue "ADD CATEGORY" button. The browser's taskbar at the bottom shows various pinned icons and the system tray indicates it's 3:14 PM on April 5, 2022, with a battery level of 8%.

This page is for the purpose of adding the categories in the budget. Category is nothing but a part of the budget. For easy understanding, it can be brought down to this.

- Every trip has a budget and only one budget.
- A budget has a category and there can be many categories.
- Categories are just the breakdown of the budget.

For example, if the budget for a trip is 600, this budget can be divided as 200 for Travel, 150 for Hotel, 100 for Food and the rest for Entertainment. So, in this case, “Travel”, “Hotel”, “Food” and “Entertainment” are the categories for this budget for this particular trip.

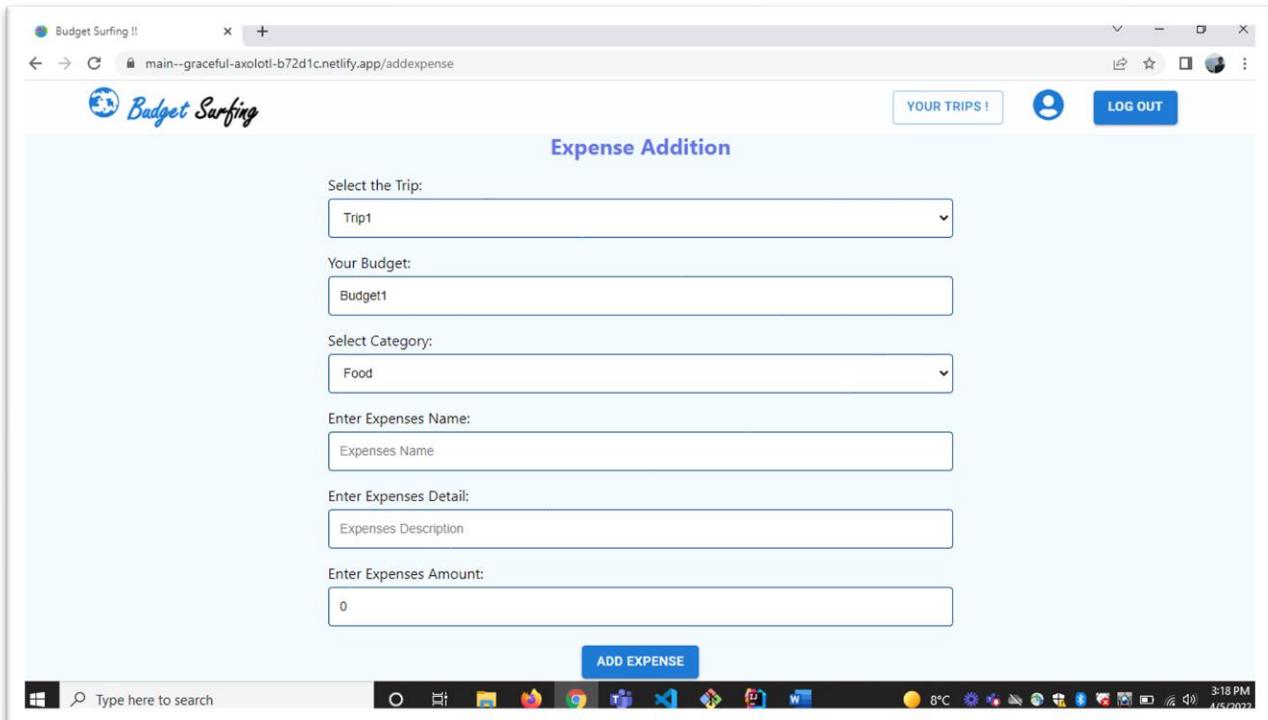
To successfully complete the form, the user has to select the trip from the list of trips if there are more than one trips. When the user selects the trip, our application automatically gives them the Bugdet name for that trip in the budget field and this is non-editable. Now the user has to enter the category name and the amount they decided to allot to that particular category. After giving the details, user has to press the “Add Category” button in order for this category to be updated in their trip details.



After successful addition of the category, the user can see something like this. Since our application allows multiple categories for a single budget and for a single trip, after adding one category, the application redirects the user to the traveler home page. There, the user will have an option to add category again in a simple way and that way, they can add as many categories as they want. There is no restriction on the number of categories for a particular budget, but the only restriction is that the category name should be unique among the respective trips.

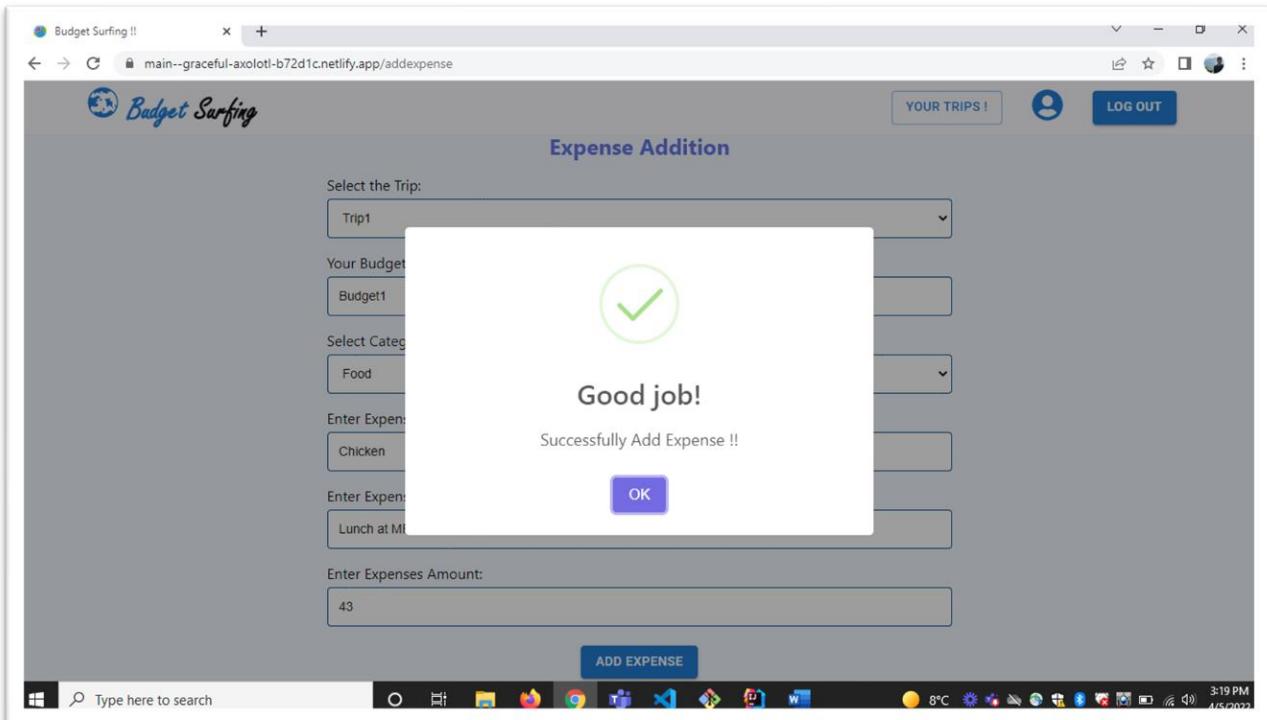
## Add Expense Page: (/addexpense)

This is the page where the user can add any expenses that have been happened during the trip or regarding to the trip. A glimpse of how the page looks is provided below in the picture.



This page consists of a form in which the trip has been selected by default by our application (this happens if there is only trip). If there are more than one trips, the user can select their trip from the list of trips. After the user has selected the trip, the application automatically fills the budget name field with the name of the budget given at the start of the trip. The user has nothing to do with this and won't be able to change it as well. Next, there are the list of categories added for this trip and after selecting the respective category type, the user must enter the details of the expense such as expense name, expense detail and the expense amount. After entering all the details as required, the form should be submitted by the button at

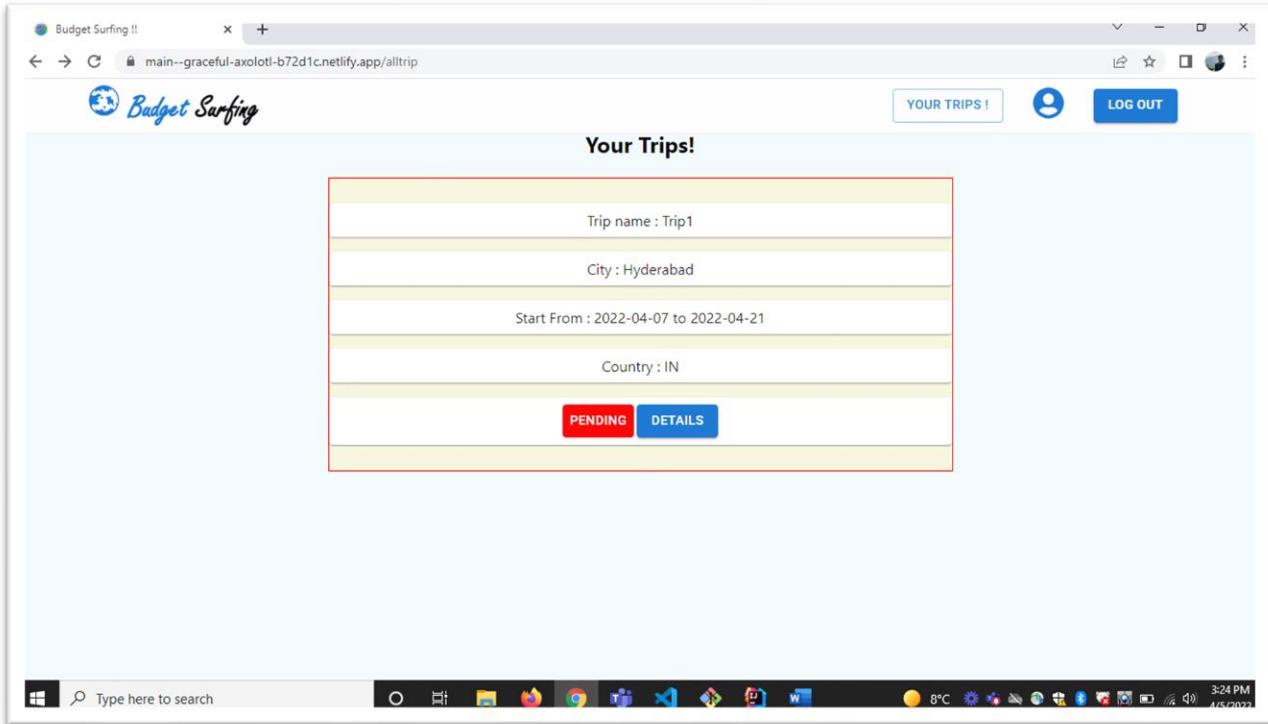
the bottom of the form “Add Expense”. This is a submit-type button, and it will add the details of the expense into the database.



After submitting the form, the user will be prompted a pop-up like shown here, just to know that the details are stored successfully. User can add as many expenses as they would want to, into any particular category. But the only restriction is that, if they want to add an expense into a category, the category should be added first; only then, the application will allow the user to add that expense. Also, within the same trip, the category name should be unique. The application won't allow the same names for multiple categories in the same trip.

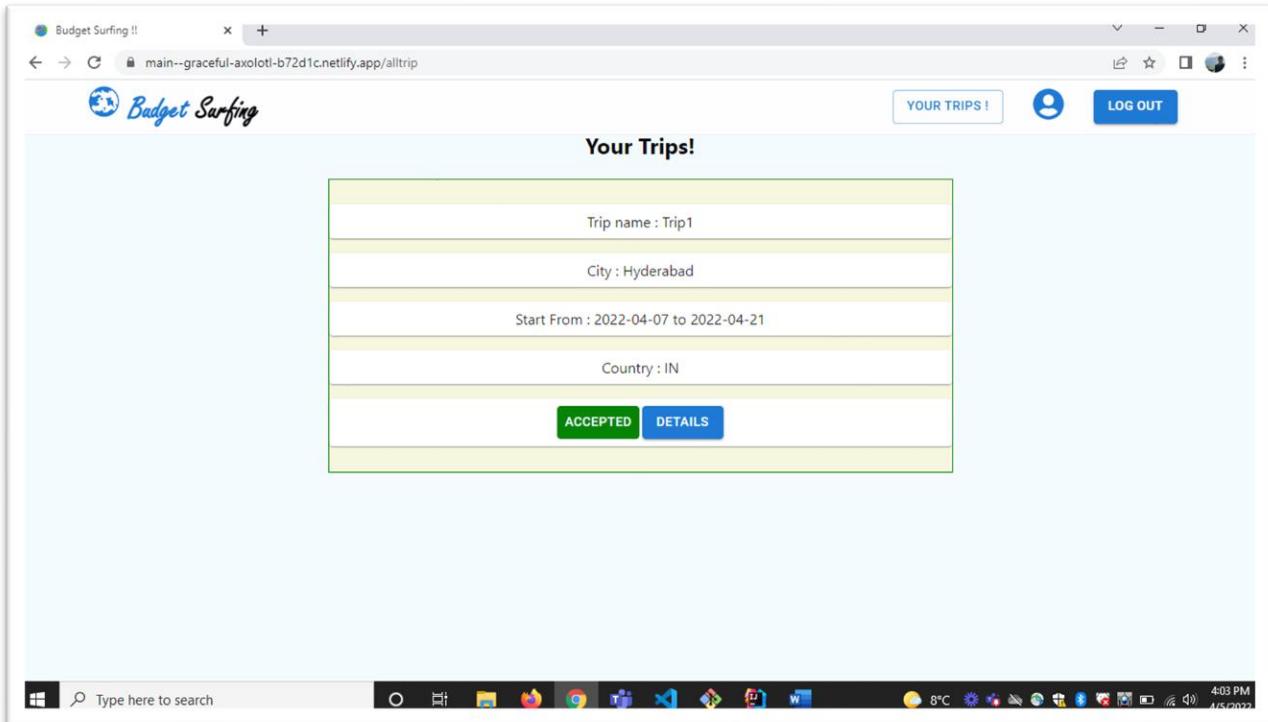
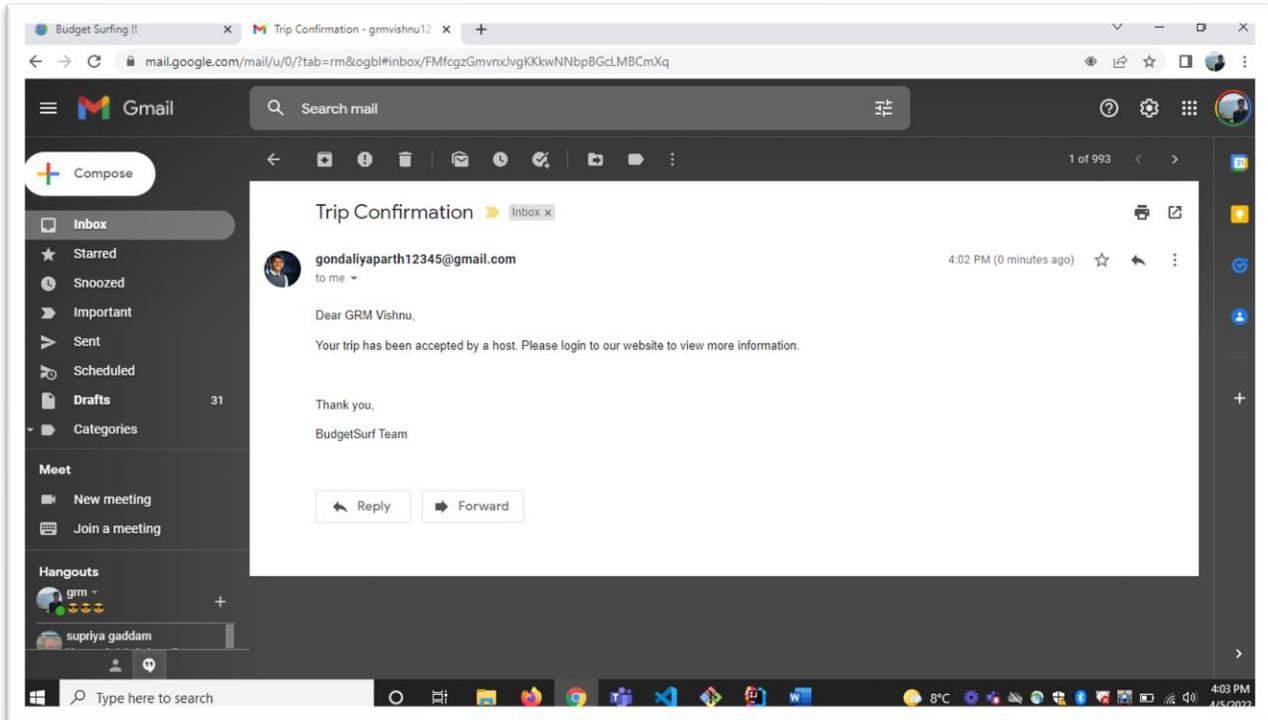
## All Trips Page: (/alltrip)

This page will show all the trips which the user has. It also shows the status of our trip. A glimpse of it has been attached down below.



If the trip of the traveler has been accepted by any user, it shows the status as “Accepted” and in green color. If it is not accepted by any user yet, it will show the status as “Pending” in red color. It shows some basic details of the trip such as trip name, trip city, start date and end date and country.

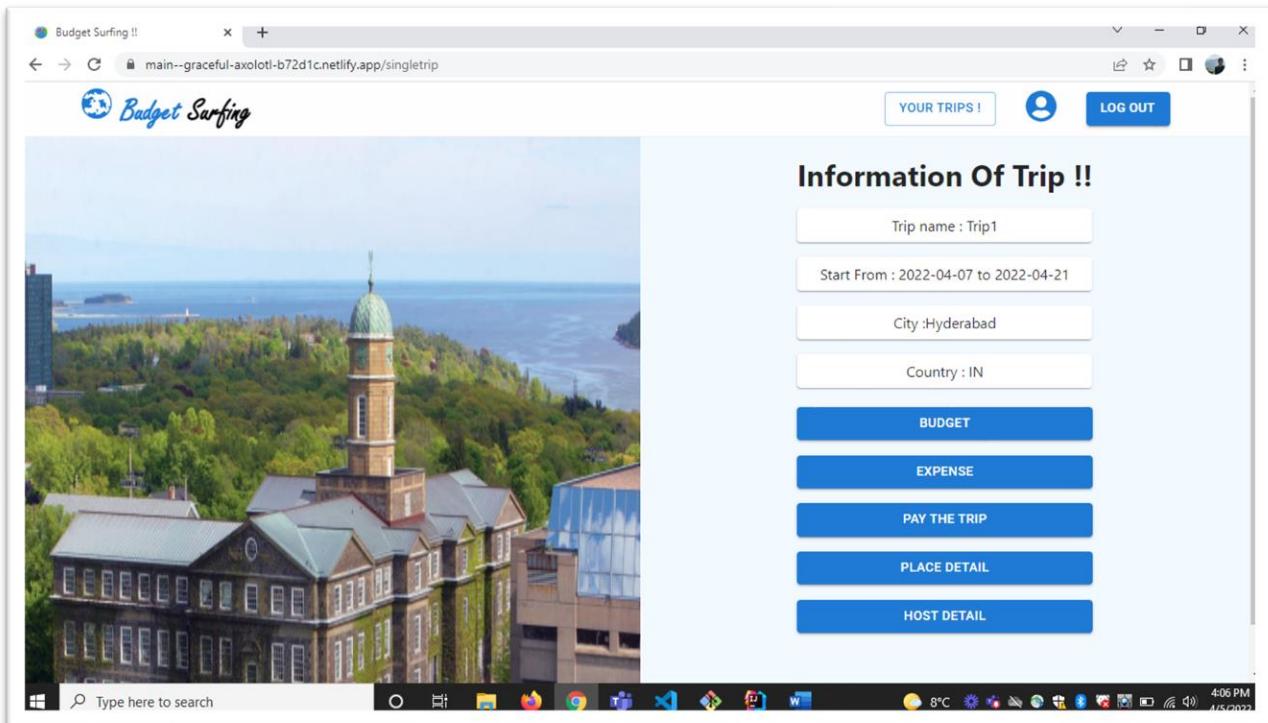
If the traveler wants to know more details about a particular trip, then they have to click the “DETAILS” button which will redirect them to a particular page where they can have a look about all the details clearly.



If the host accepts the traveler trip, then the traveler gets a notification on their email regarding the same like above.

## Single Trip Information Page: (/singletrip)

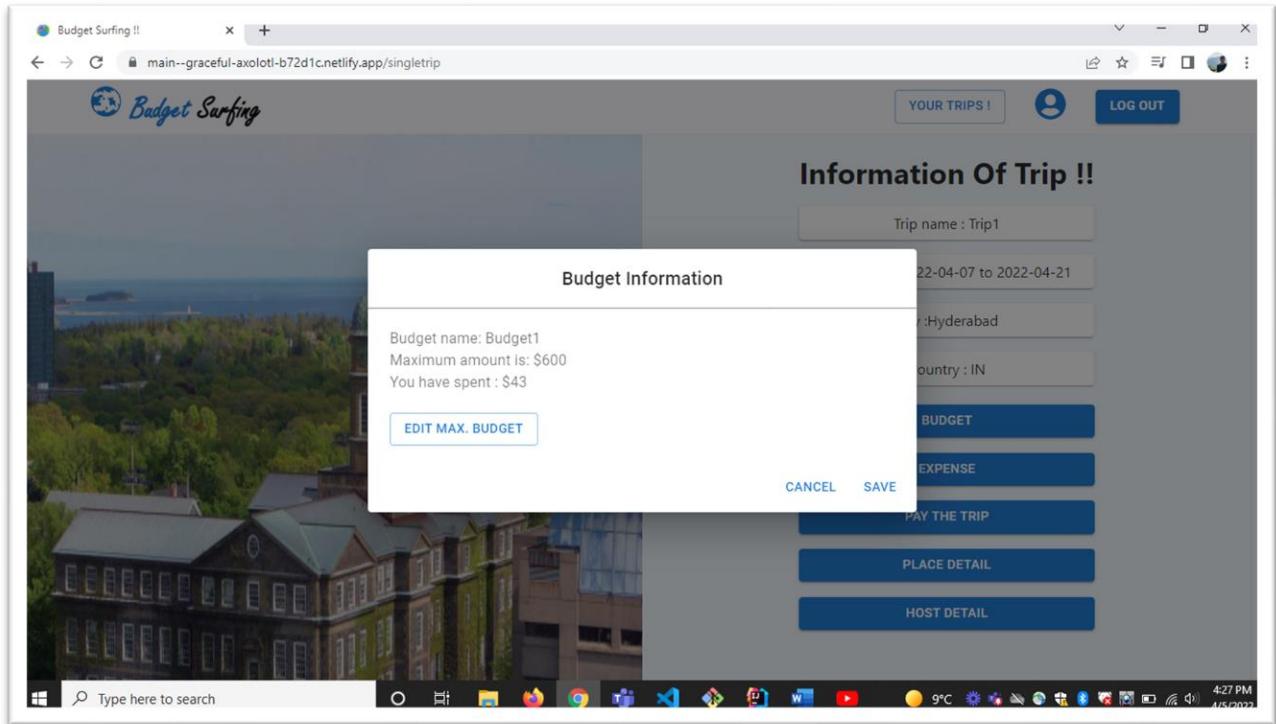
This is the page where the user gets all the information about a particular trip. This page consists of a header as discussed earlier and a body showing the details of the trip and having few buttons to perform operations. Screenshot of this page is given below.



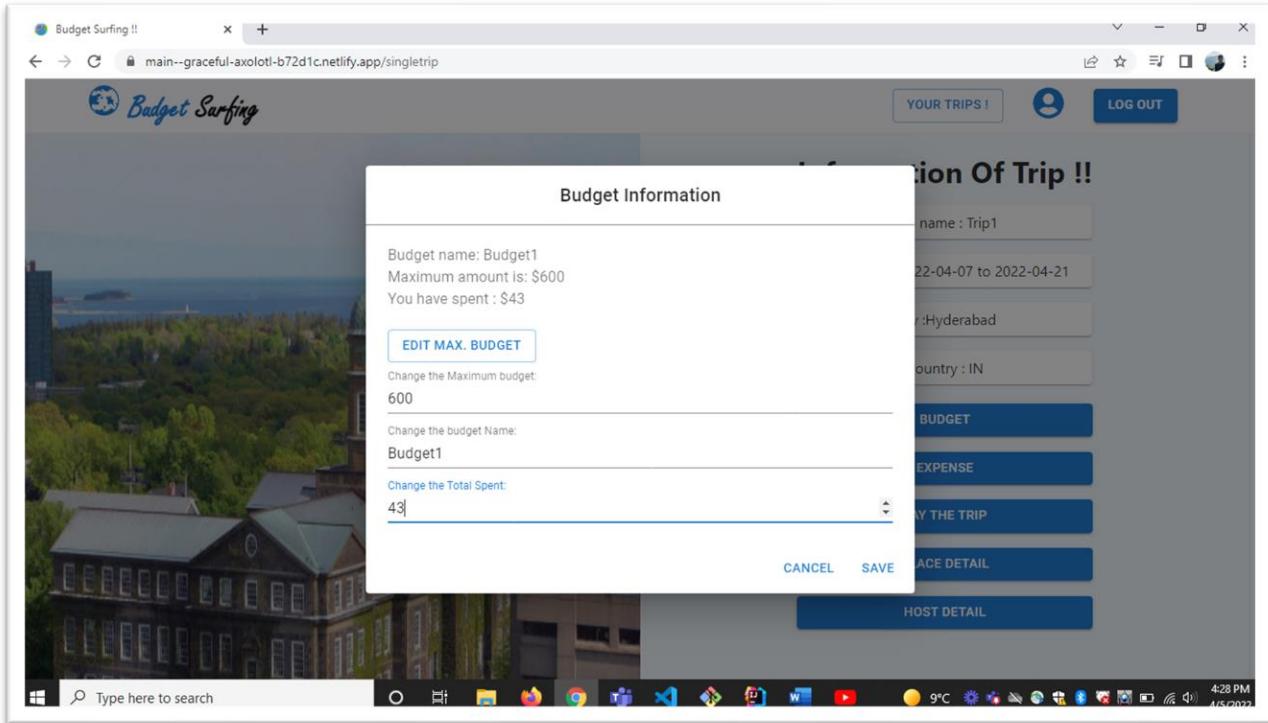
This page has the details of the trip such as the trip name, trip start date, trip end date, location of the trip like city and country etc. Adding to those, there are few buttons in this page of which the purpose of the buttons is discussed in the coming part of the report.

The buttons in this page are BUDGET, EXPENSE, PAY THE TRIP, PLACE DETAIL AND HOST DETAIL.

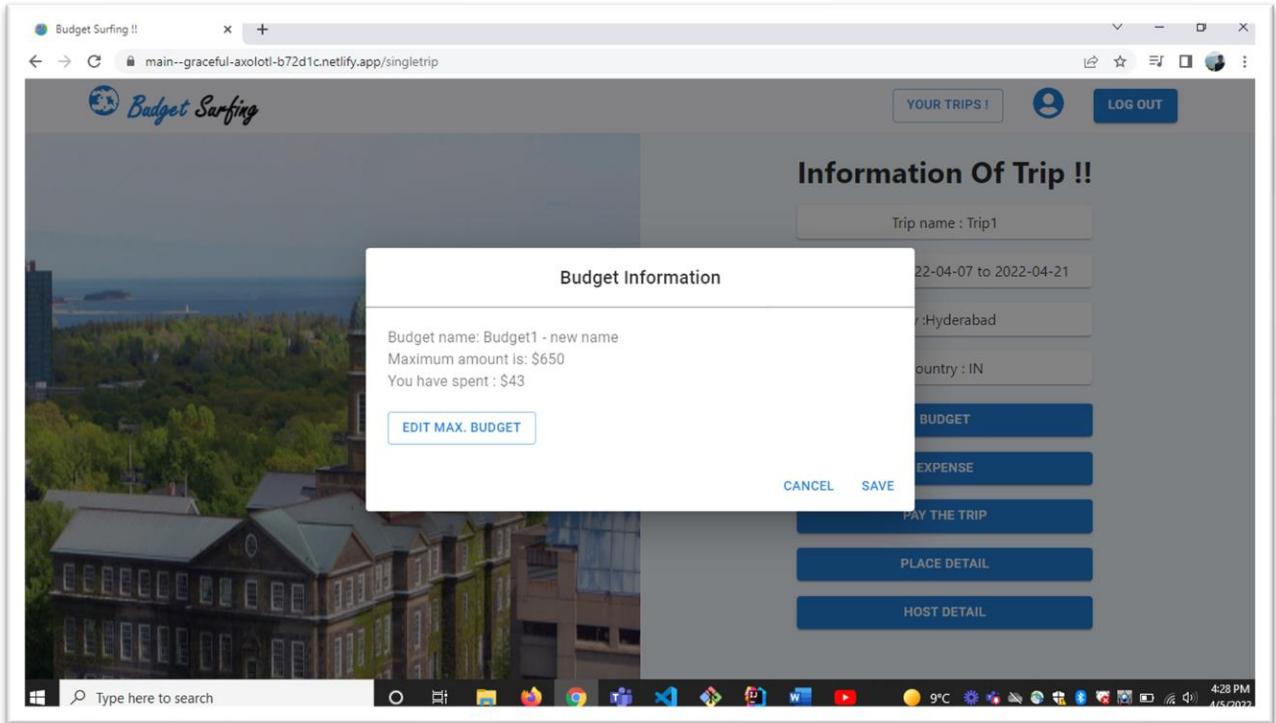
- Budget



On click of this button named “Budget”, the user will be able to have a look at the budget they had set when they were creating the trip. It displays the budget name, the budget maximum amount set and the amount which has been already spent from the maximum limit.

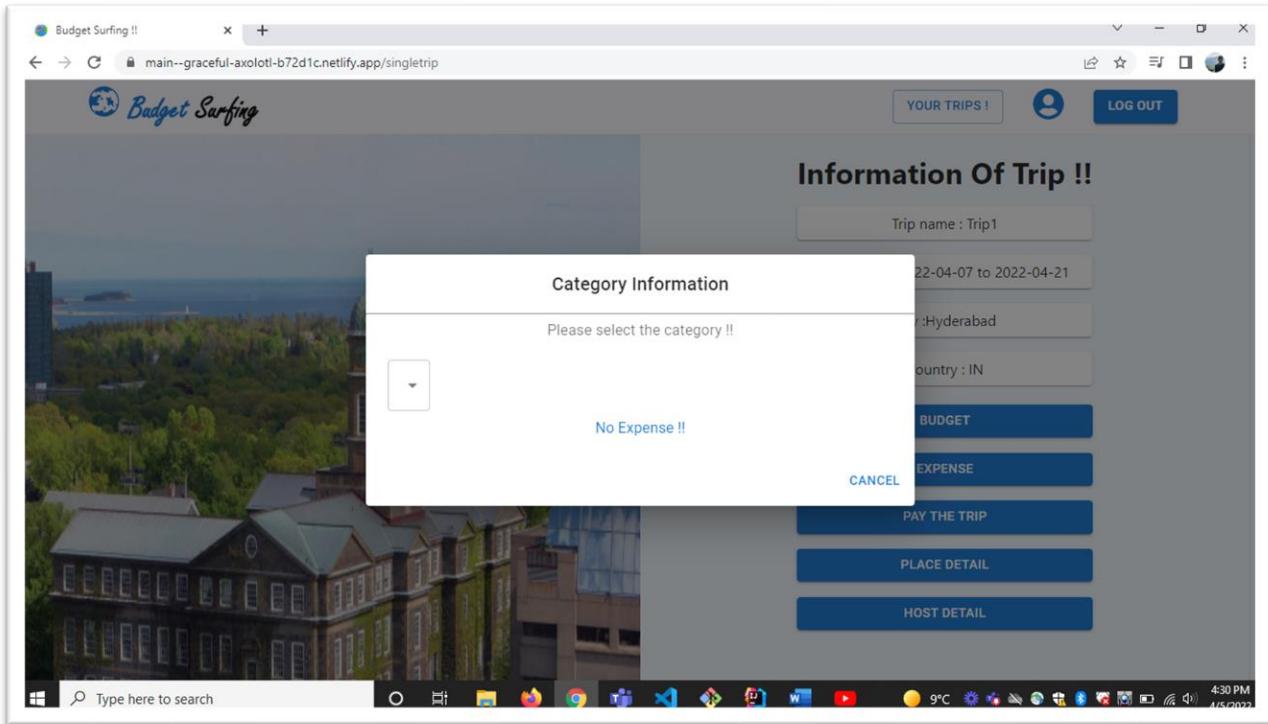


If the user wants to edit the budget or change any information in this budget related details, they can simply do that by clicking “EDIT MAX BUDGET” button and change the details as needed and then save it.



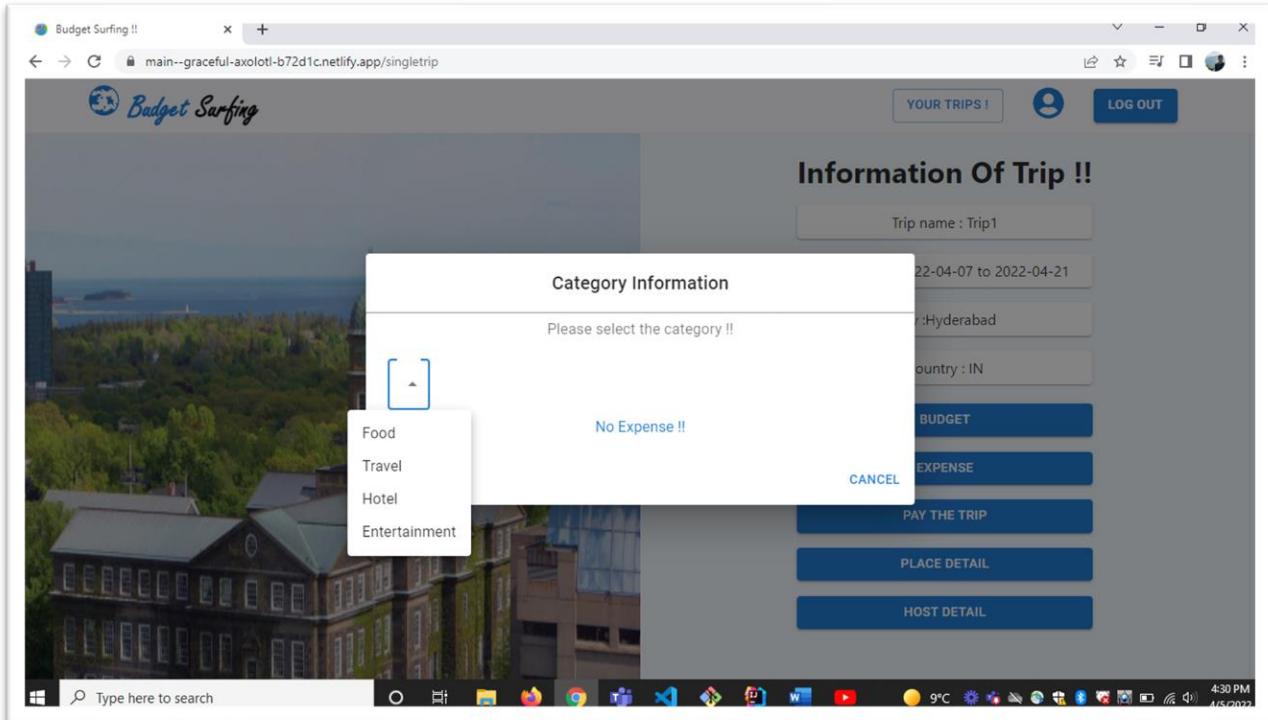
The above image shows the information of the budget after updating the details of the budget.

- Expense

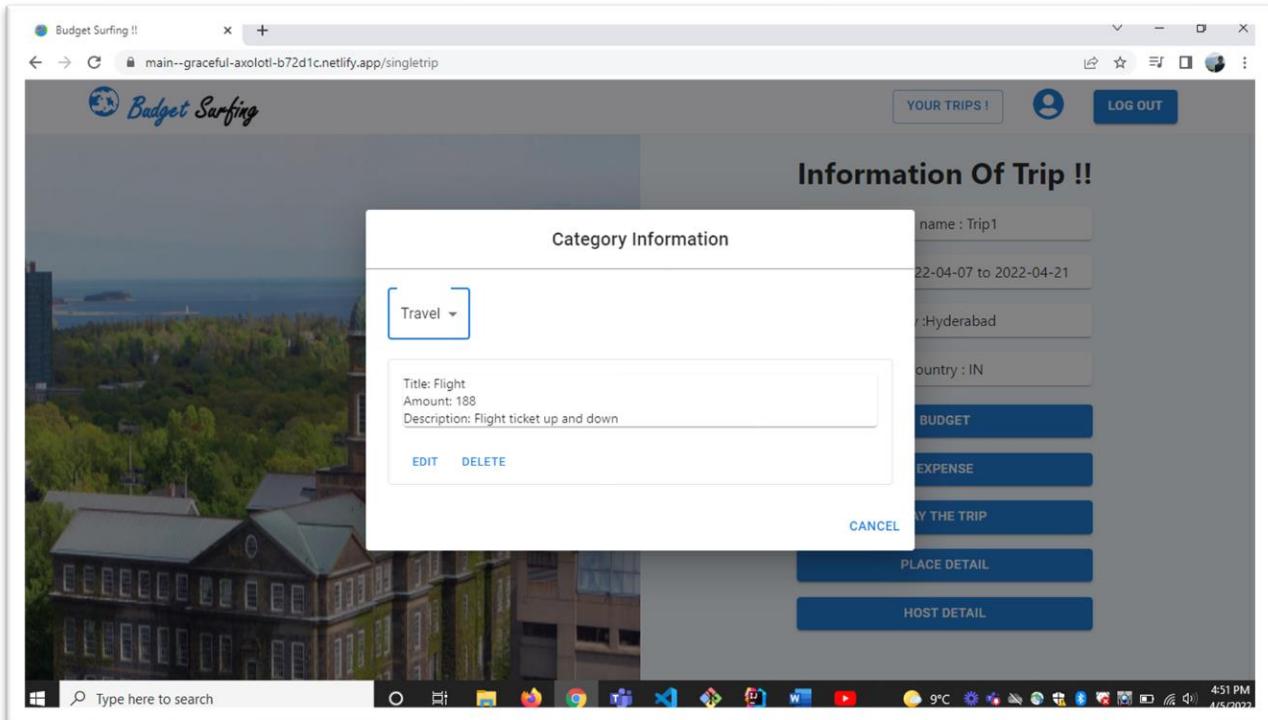
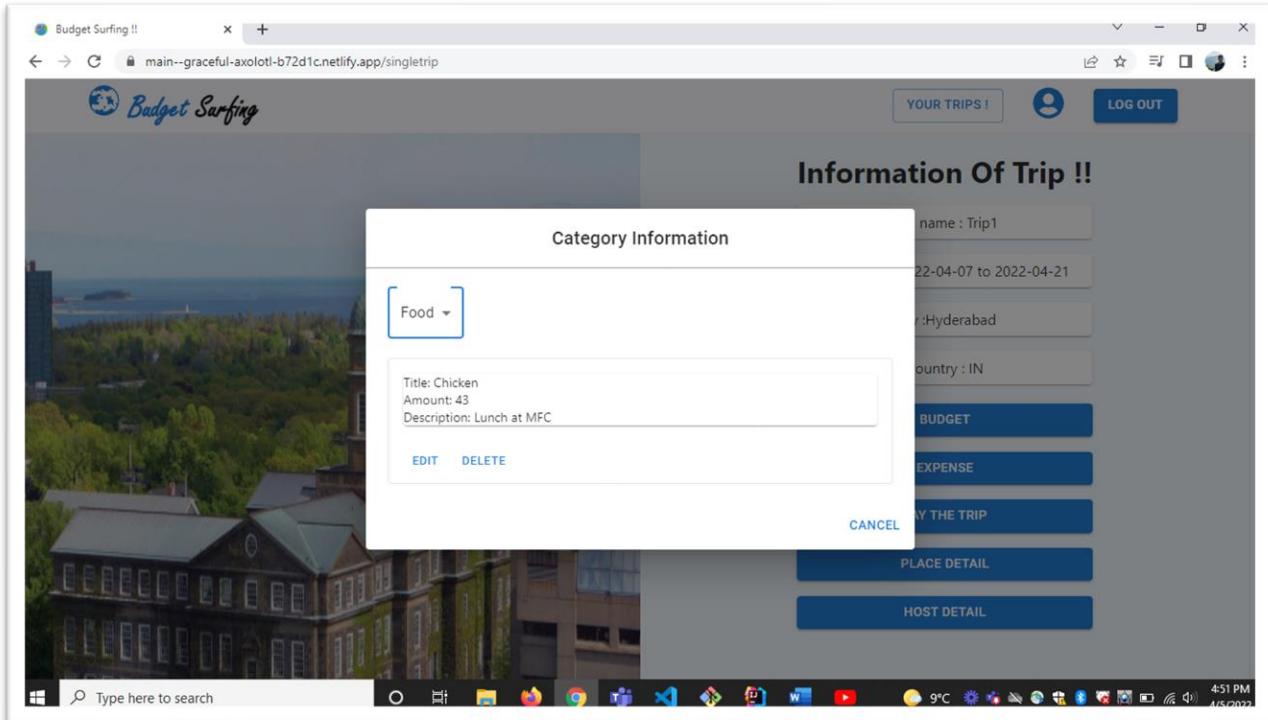


As we discussed earlier, the expenses spent by the user as entered into the application using “ADD EXPENSE” page earlier, can be viewed through this page. This button named “Expense” allows you to view the details of your expenses in all the categories and perform operations on them.

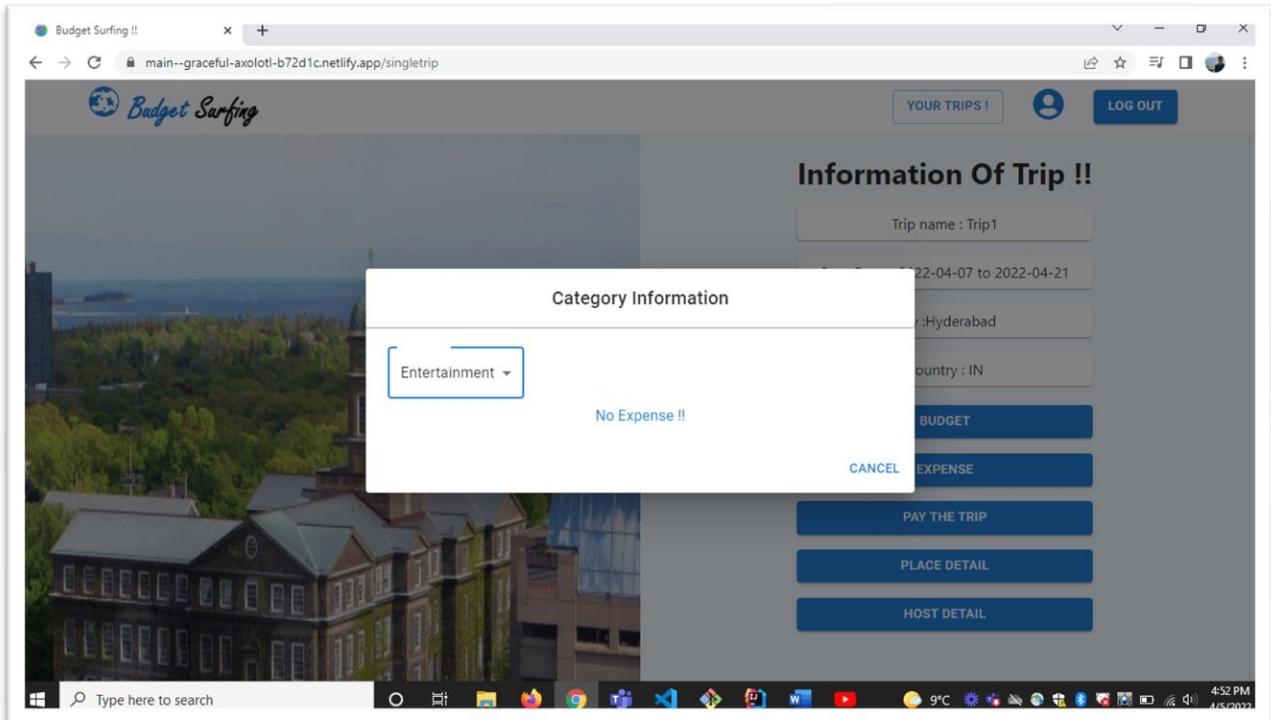
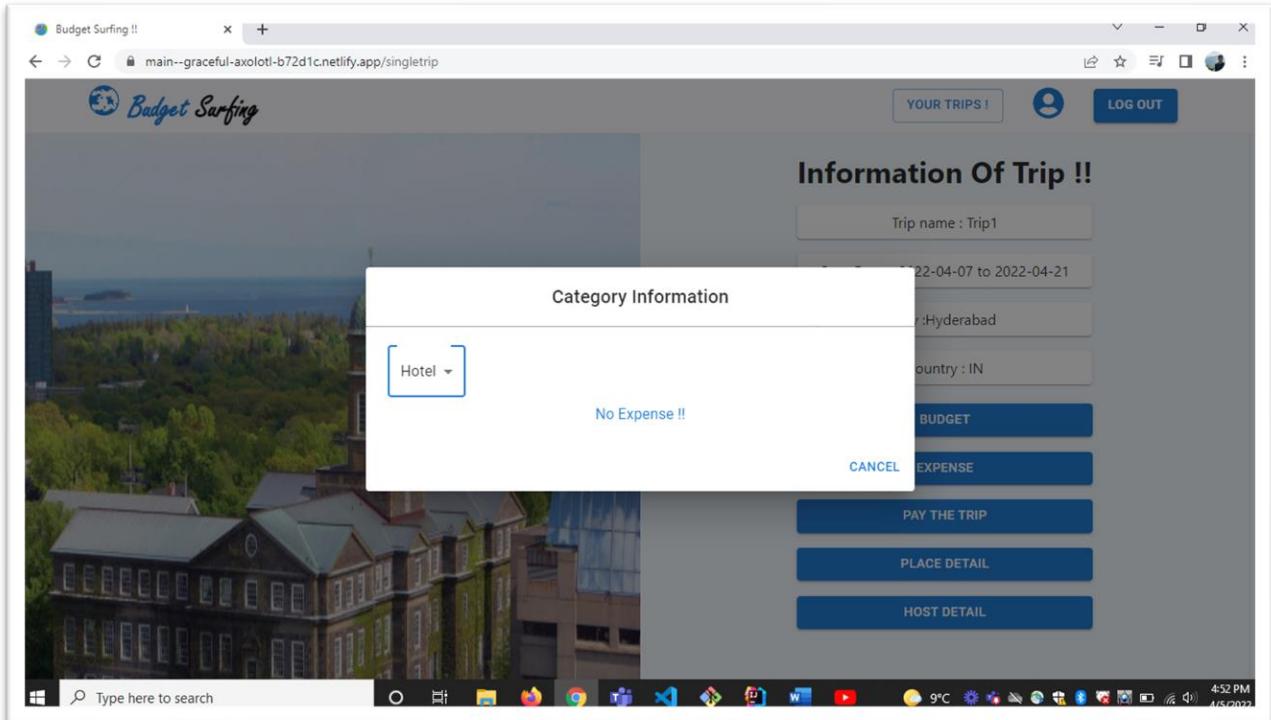
As soon as the pop up comes after clicking the expense button, we will be instructed to select the category first from the dropdown list shown, in order to view the expenses in that particular category.



If the user did not add any categories while adding the trip, the dropdown list will be empty and will show nothing. If the user did add any of the categories, then the dropdown will list all its categories.



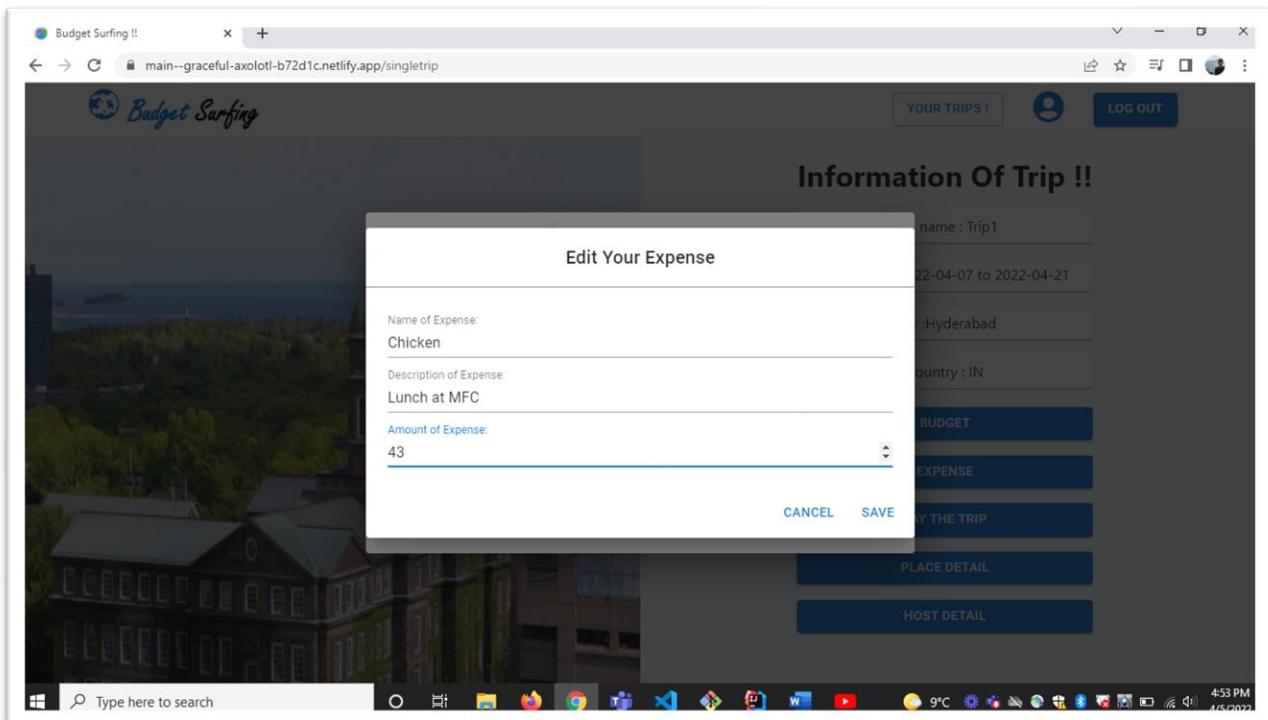
After selecting a particular category from the list, the user will be able to see all the expenses added in that category.



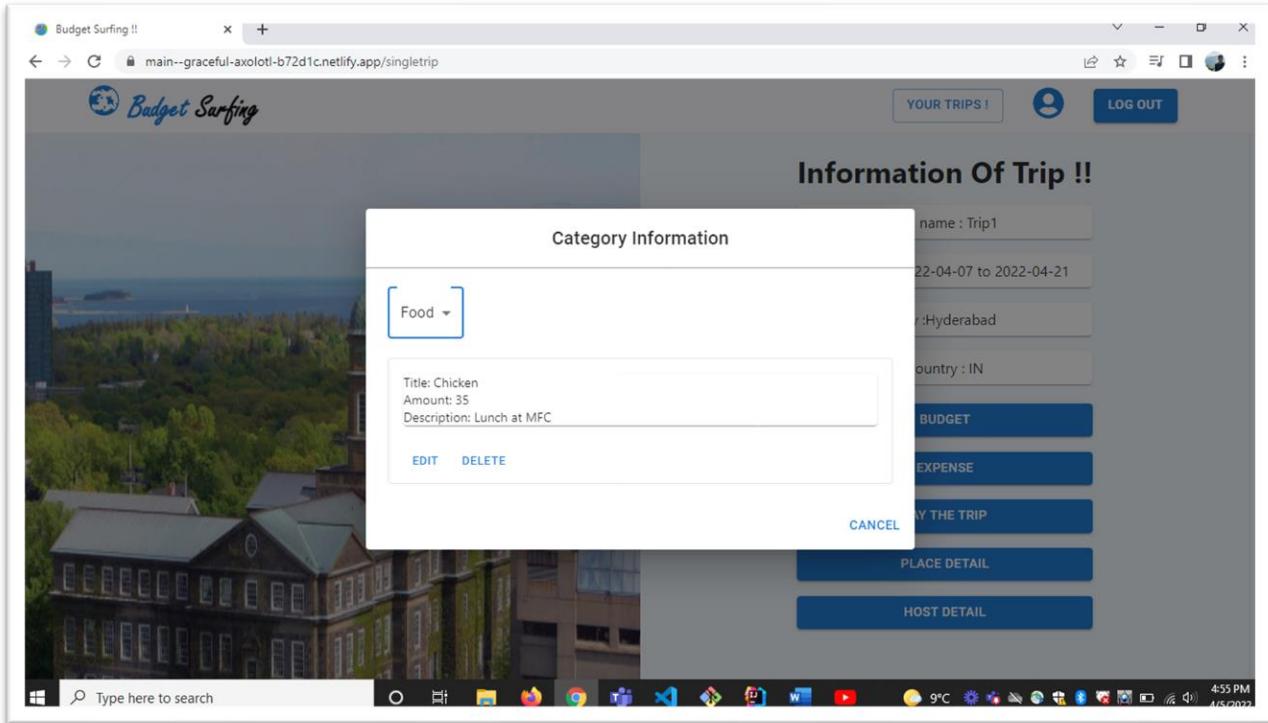
If the user has added only expenses in few categories and few categories are empty, then it will show as “No Expense !!” in those particular categories.

Now the user will be allowed to do a couple of operations if they want, on the expense.

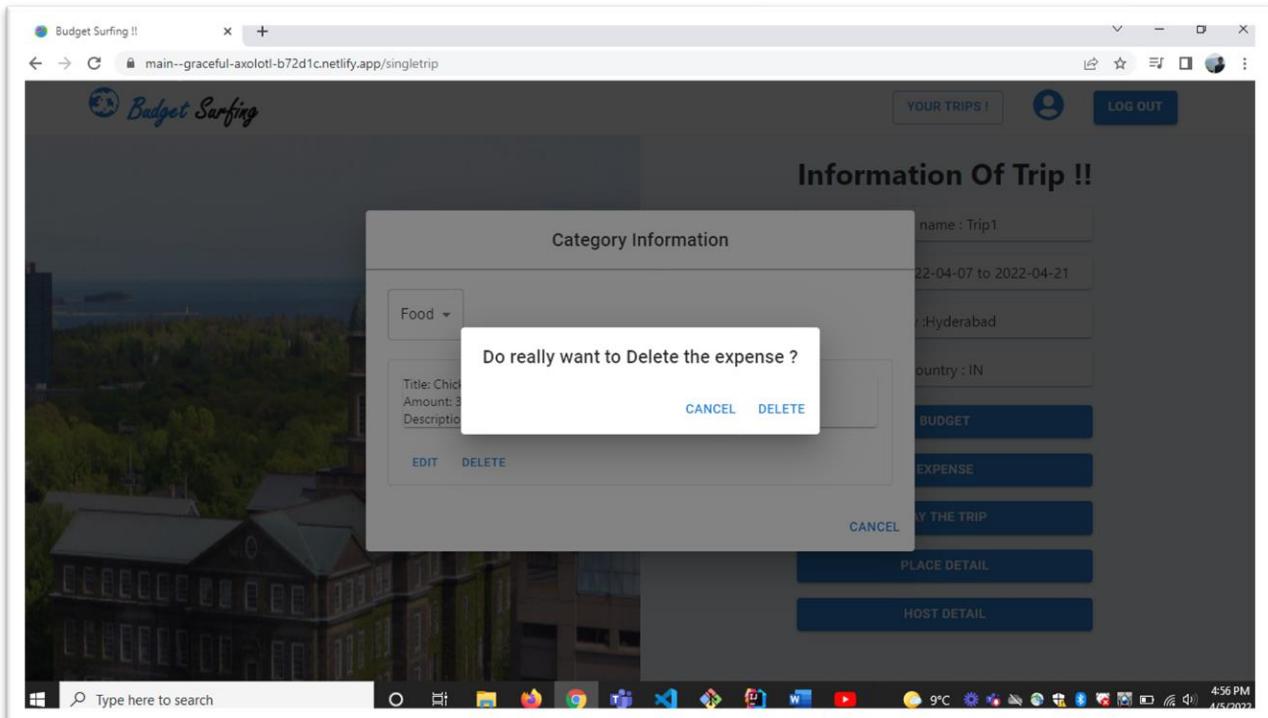
The two operations which they are allowed to perform are “EDIT” and “DELETE”.



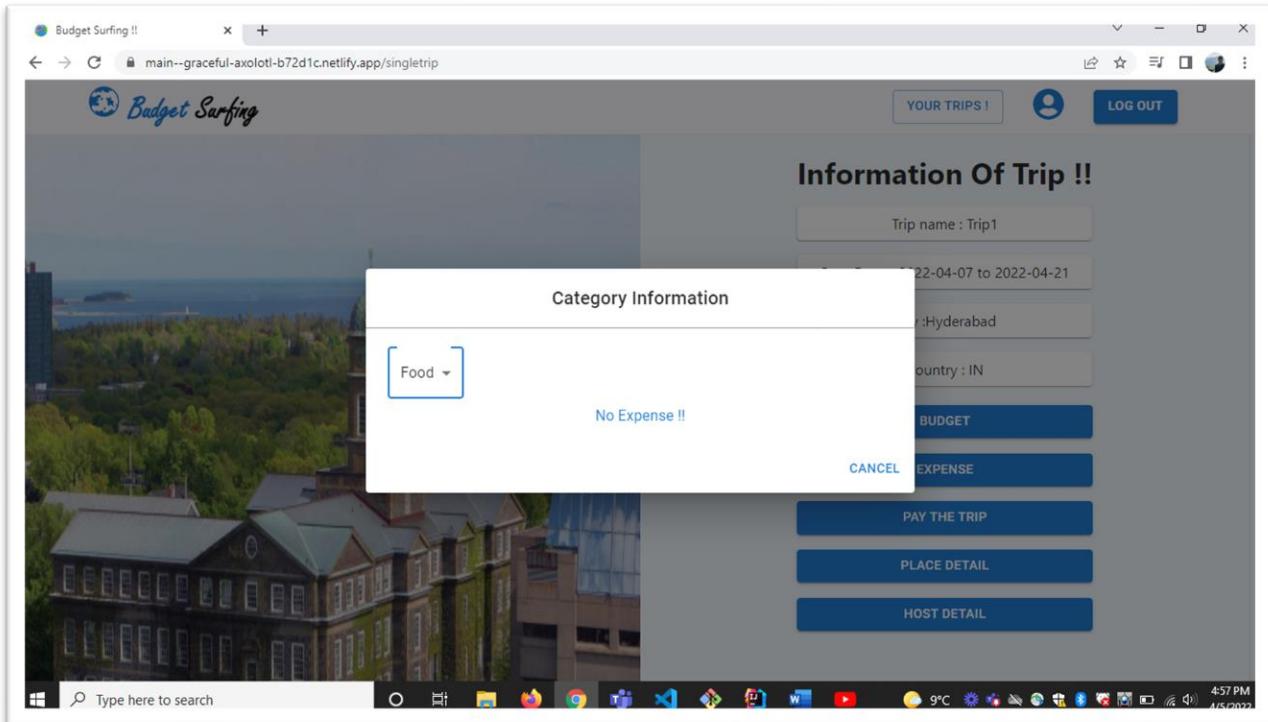
If the user wants to edit the information of a particular expense in the particular category of their trip, then they can do that with this option. The user has to select this option and will be redirected to a form to edit their expenses for the selected expense. After making the respective changes, the user has to save the details of the expense and this way, their expense detail will be updated.



The above image shows the expense after editing in the same category as shown before.



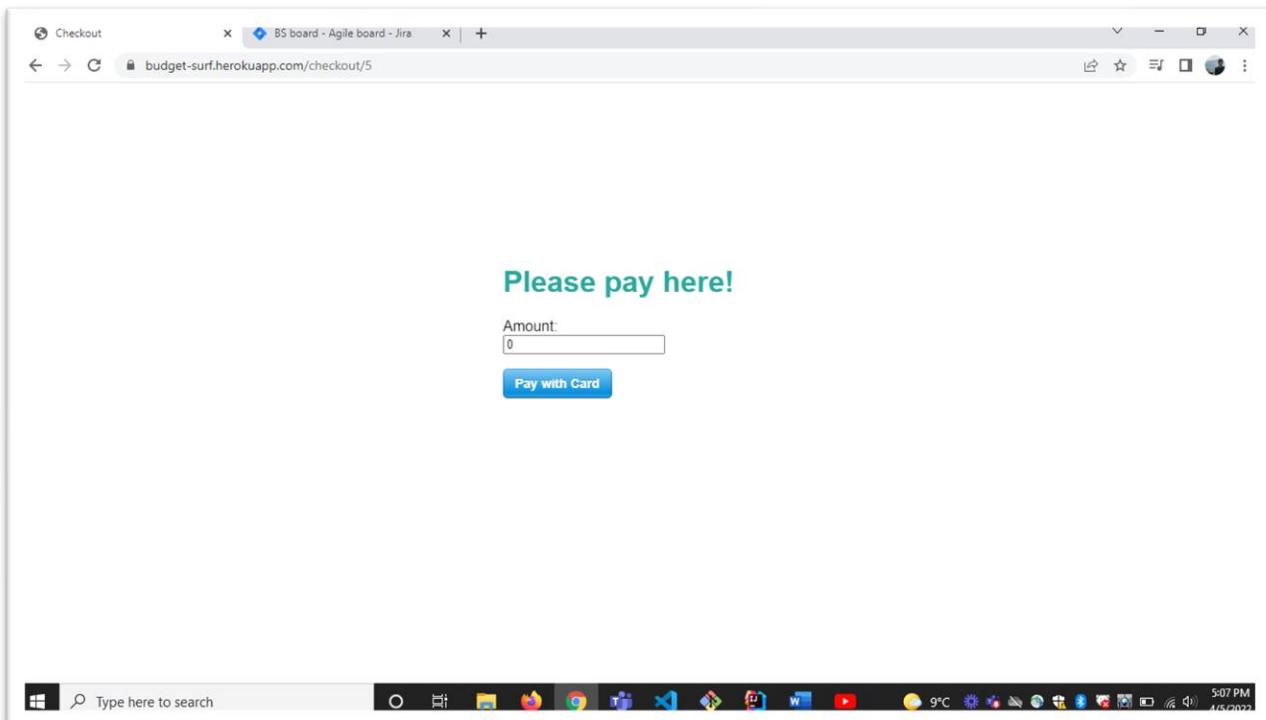
If the user wants to delete an expense which they added for some reason, then that operation can be fulfilled through this way. The user should select the delete option and upon clicking the option, the user will be asked once again to confirm if they really want to delete the expense that was added or was it just a mistake. If the user changes their mind and want to keep the expense, they can cancel the operation. If they want to proceed further with deleting it, they can delete the expense from the list of expenses and after deleting the expense, the total amount spent from the maximum budget limit will be updated.



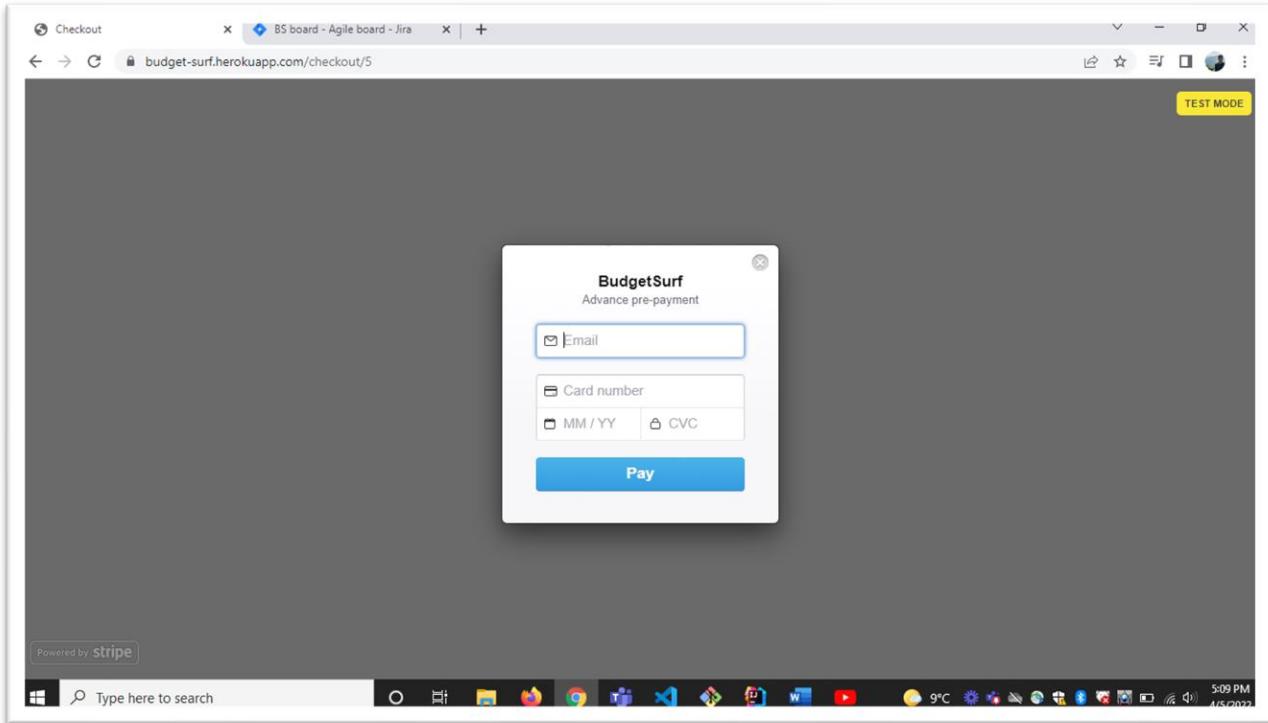
The expenses in the same category as before is shown in the figure after deleting an expense.

## **Payment: (/checkout)**

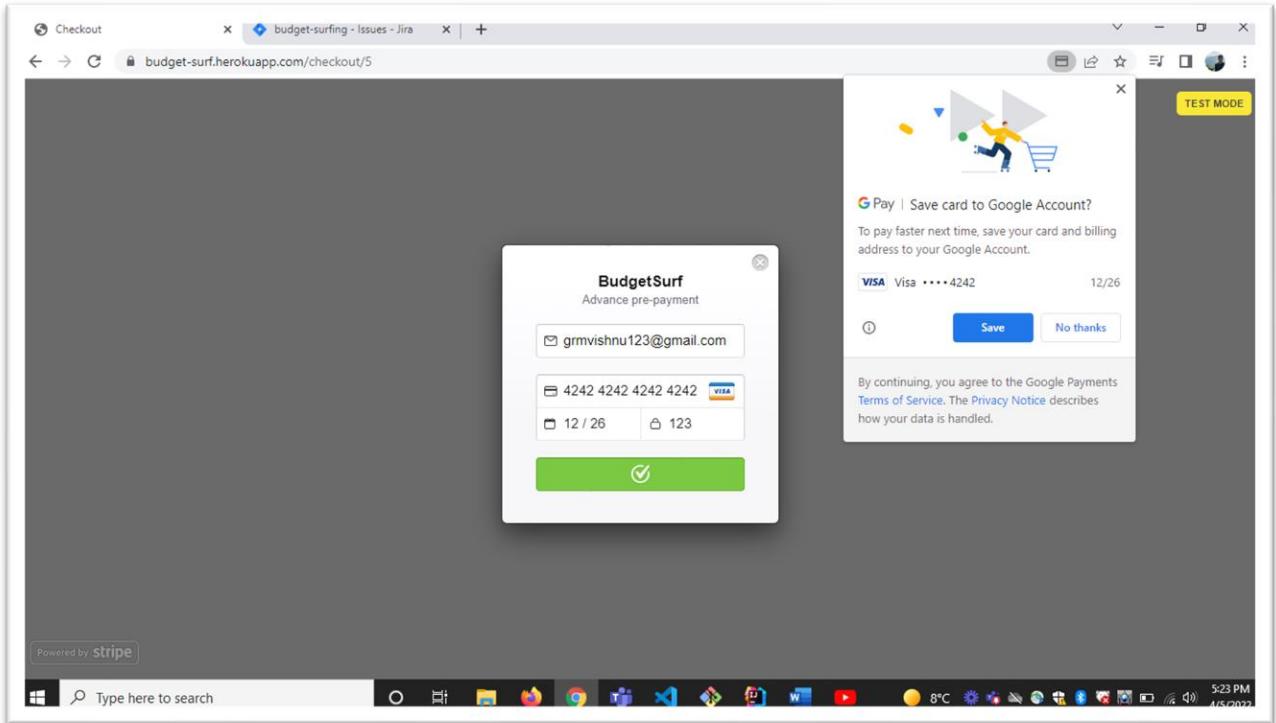
The third button in the single trip information page is the “PAY THE TRIP” button. If the user selects this button, they will be redirected to this page. The payment page can be seen below.



In this page, the traveler has to enter the amount which they are willing to pay to the host. After entering the amount in the field, the user has to click “Pay with Card” button which then directs a pop-up.



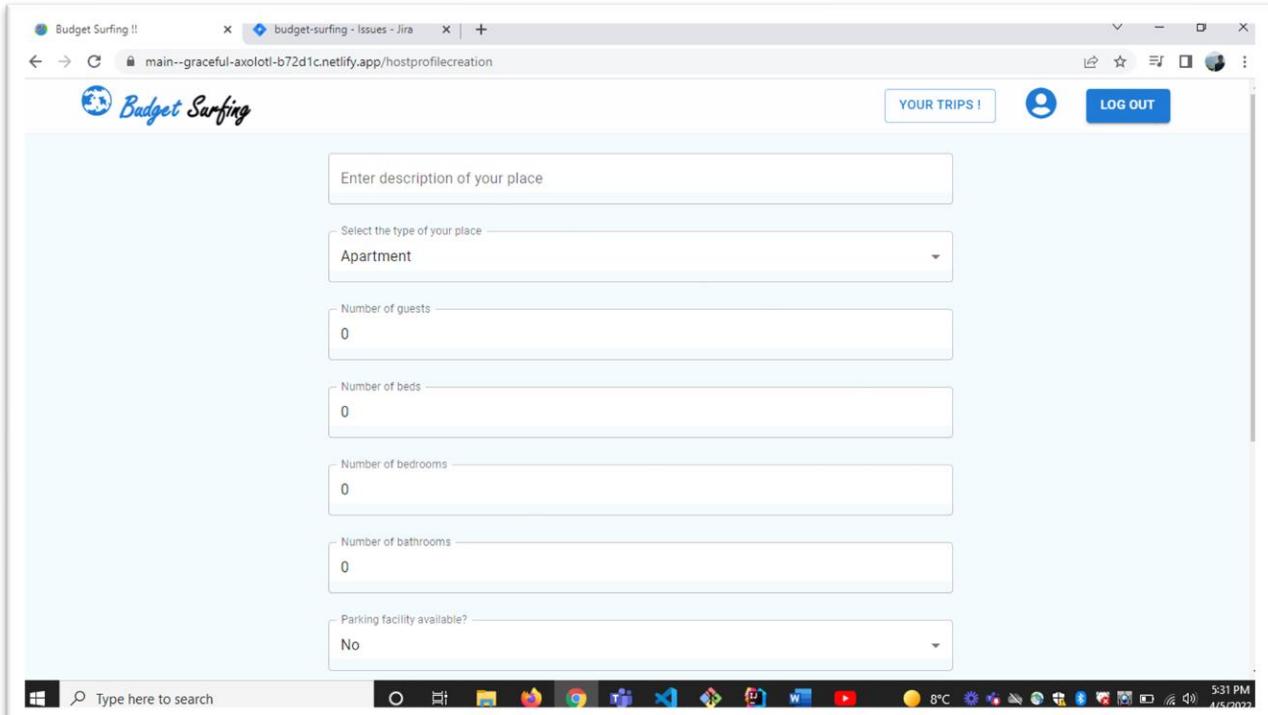
This pop-up is for the card details of the user. The card details like the card number, expiry date, email and cvv should be valid and the card should be linked to the email address provided with the user account. After all the details are entered, the user should pay by clicking the “Pay” button.



When the payment is done, the application shows it as success and the money will be sent to the host account which the host updated in their profile.

## Place Detail: (/hostprofilecreation)

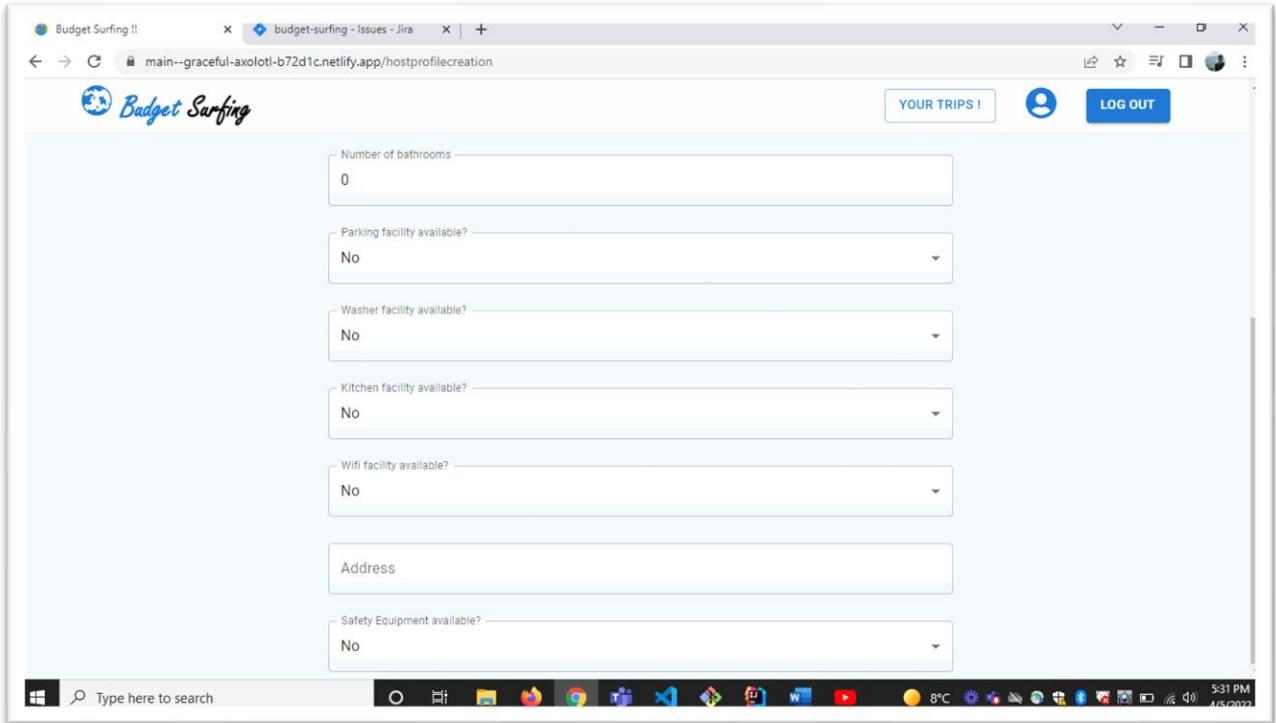
The fourth button in the single trip information page is the “PLACE DETAIL” button. This button will redirect to a page which shows the information of the host building, i.e., the building which the traveler is going to live in or already lives in. A glimpse of it can be shown as this.



A screenshot of a web browser window showing a form for creating a host profile. The title bar reads "Budget Surfing !! budget-surfing - Issues - Jira". The main content area has a "Budget Surfing" logo at the top left. On the right side, there are "YOUR TRIPS!" and "LOG OUT" buttons. The form itself consists of several input fields:

- "Enter description of your place": An empty text input field.
- "Select the type of your place": A dropdown menu set to "Apartment".
- "Number of guests": An input field containing "0".
- "Number of beds": An input field containing "0".
- "Number of bedrooms": An input field containing "0".
- "Number of bathrooms": An input field containing "0".
- "Parking facility available?": A dropdown menu set to "No".

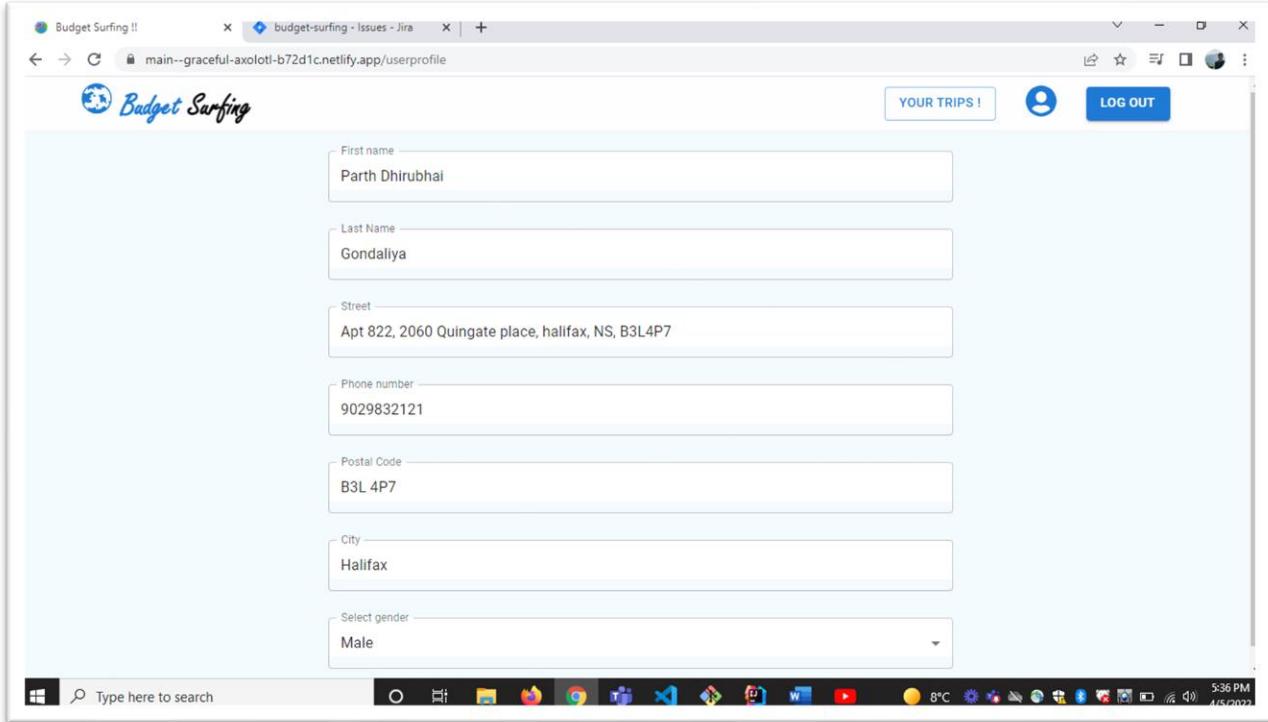
The browser's taskbar at the bottom shows various pinned icons, and the system tray indicates the date as 4/5/2022 and the time as 5:31 PM.



This page shows the basic details of the building like the description of the place, place type, number of bedrooms, no of washrooms, facilities available, number of guests etc. Whatever the user has filled in his profile, the traveler would be able to see that here. And also, these fields are unchangeable for a traveler. It means even if the traveler wants to change any details in this form, he won't be able to do it. Only the host of this building has the authority to access this information and modify it.

## Host Detail: (/userprofile)

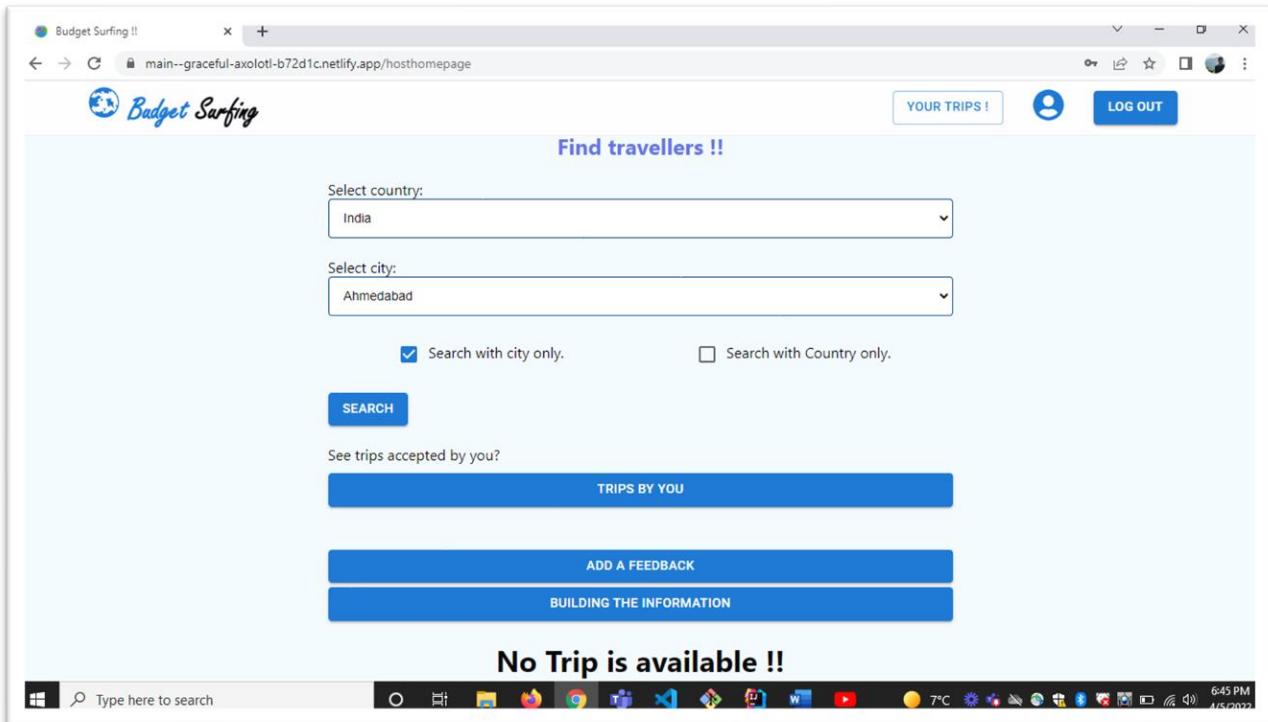
As the name itself talks, this page shows the user profile of the host. This is the last button in the single trip information page. The page will look like this.



These are the details of the host who accepted the traveler to stay at their place. It's basically the same details as the traveler, since the application takes same details of any user irrespective of them being a traveler or a host. The traveler can contact the host if they want, after accessing these details. The address of the host will be present, the phone number, city, name will all be seen in this page.

## Host Home Page: (/hosthomepage)

Till now, all the pages which we have discussed are regarding the traveler. Now let's discuss the pages which are designed for the host. In the traveler home page, the second half of the screen has a button named "WANT TO BE A HOST!". If the user is a host, then they can go to all the host pages through this button. Clicking this button will lead the user to the page shown below.



This is the home page for the host. As you can see, there are many things in this single page. Let's talk about each button and form in detail.

## Host Profile Creation: (/hostprofilecreation)

The first thing any host should do after becoming a host is to update the host building information. The user can get to this page by clicking the “BUILDING INFORMATION” button on the host home page. The page we are talking about looks something like this.

The screenshot shows a web browser window for 'Budget Surfing !!' with the URL 'main--graceful-axolotl-b72d1c.netlify.app/hostprofilecreation'. The page has a light gray background and features a logo for 'Budget Surfing' with a blue globe icon. At the top right are 'YOUR TRIPS!' and 'LOG OUT' buttons. Below the header is a large input field labeled 'Enter description of your place'. Underneath it is a dropdown menu labeled 'Select the type of your place' with 'Apartment' selected. Following are five input fields for guest information: 'Number of guests' (0), 'Number of beds' (0), 'Number of bedrooms' (0), 'Number of bathrooms' (0), and a dropdown for 'Parking facility available?' with 'No' selected. The bottom of the screen shows a Windows taskbar with various pinned icons and a search bar.

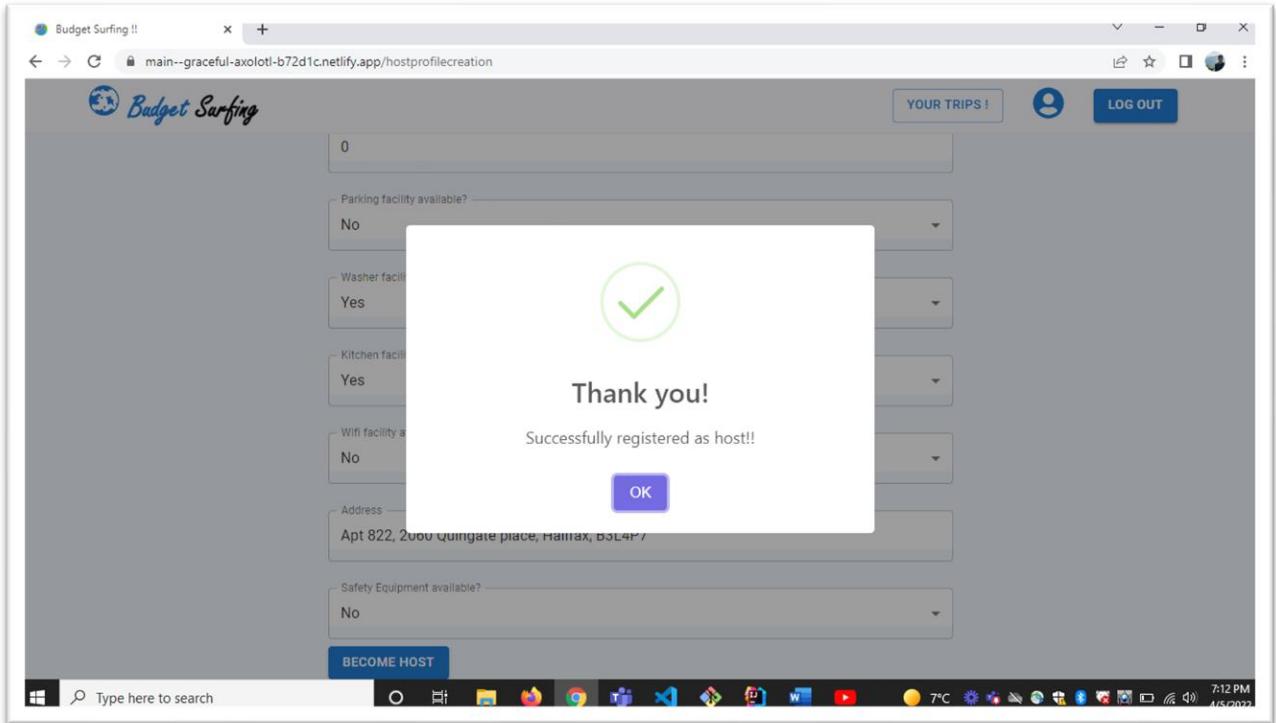
The screenshot shows a web browser window titled "Budget Surfing !!" with the URL "main--graceful-axolotl-b72d1c.netlify.app/hostprofilecreation". The page features a "Budget Surfing" logo at the top left and navigation links for "YOUR TRIPS!", a user icon, and "LOG OUT" at the top right. Below the header, there are several dropdown menus and input fields for host information:

- "Parking facility available?" dropdown set to "No".
- "Washer facility available?" dropdown set to "No".
- "Kitchen facility available?" dropdown set to "No".
- "Wifi facility available?" dropdown set to "No".
- "Address" input field.
- "Safety Equipment available?" dropdown set to "No".

A large blue "BECOME HOST" button is centered at the bottom of the form. The browser's taskbar at the bottom shows various pinned icons and the system clock indicating "7:06 PM" on "4/5/2022".

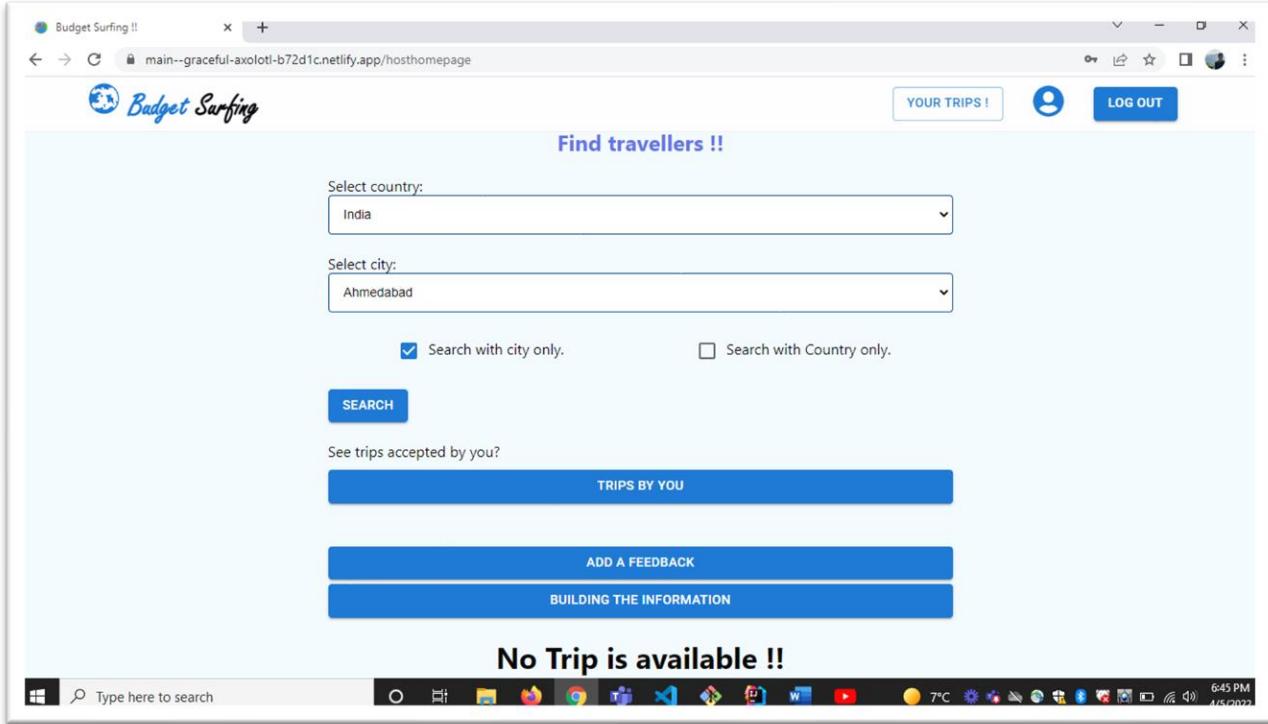
This information is required to be filled by the host in order for the travelers to know about the place if the host accepts their requests.

This form includes details such as description of the place, selecting the type of the place, number of guests the host want to allow, number of beds available, number of bedrooms available, number of bathrooms available, parking facility if available or not, washer facility if available or not, kitchen facility if available or not, wifi facility if available or not, address of the place where the travelers are going to stay and finally, if the safety equipment is available or not. After filling all the details, the host has to submit the form by clicking the button “BECOME HOST”.



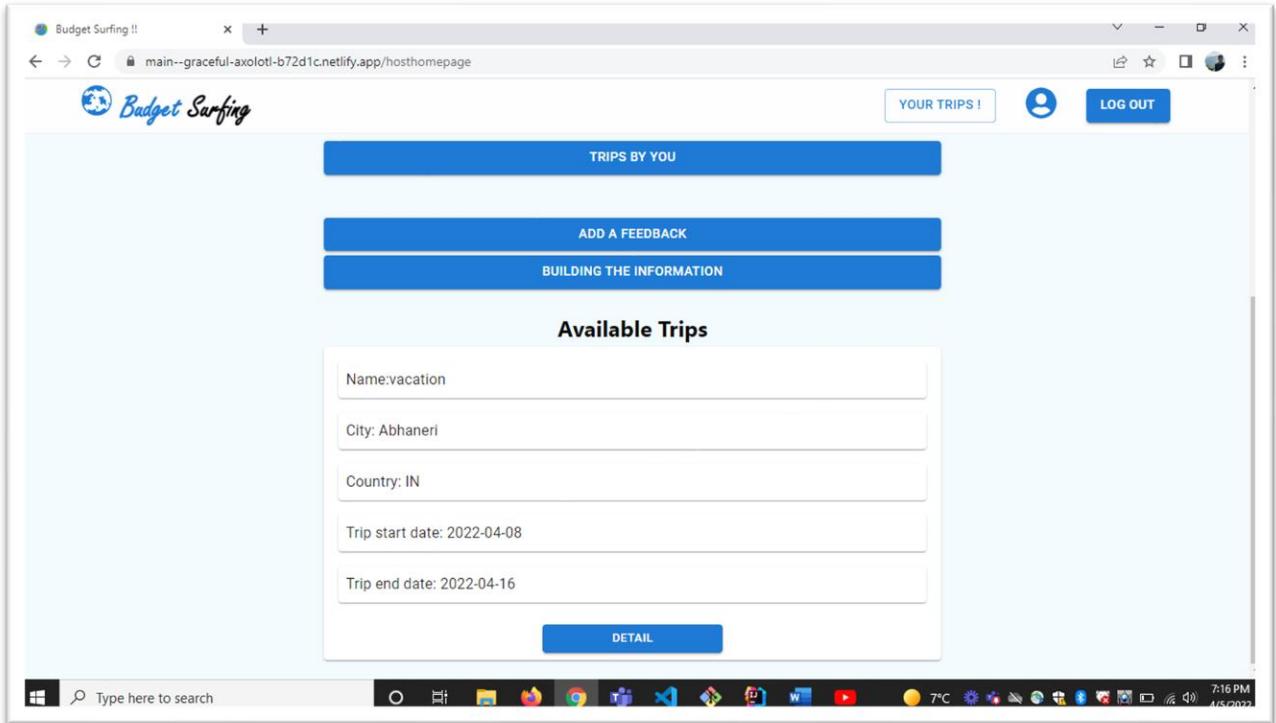
After submitting the form, the application gives a pop-up to let the host know that the details have been saved. After clicking the “OK” button, the application will take the user back to host home page where he can perform more operations.

## Available Trips: (/hosthomepage)



In the host home page, the first form which has a heading of “Find travelers!!” is for the host if they want to search the travelers. They have to select the country first and then the city of where their place is, in order to see the travelers who are searching for a place in that particular city. There are two options to search.

- Search with city only
- Search with country only



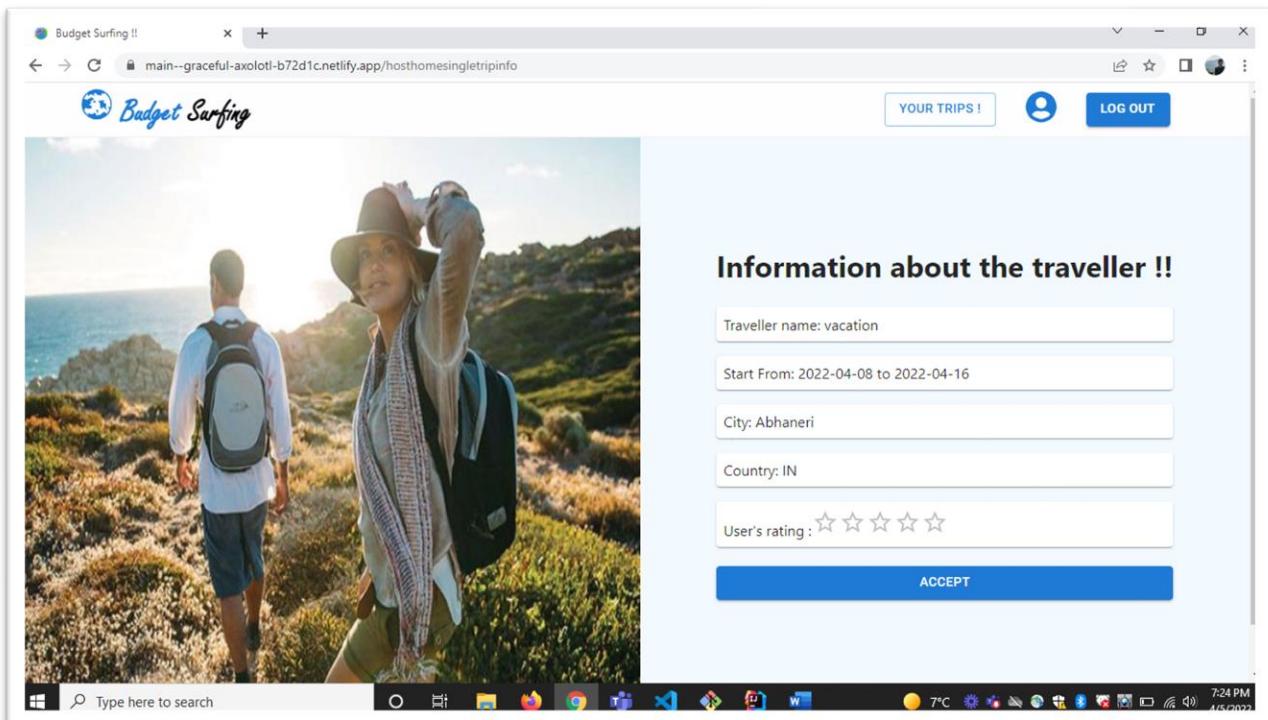
Either way, the host can see the travelers based on the search and when they search with the help of “Search” button, the page will look like this.

These are the details of a traveler who wants to come to the country and city which the host selected. The host can also see the details of the trip such as trip name, trip start date and trip end date in order for the host to check if their place is available in the given period of time or not.

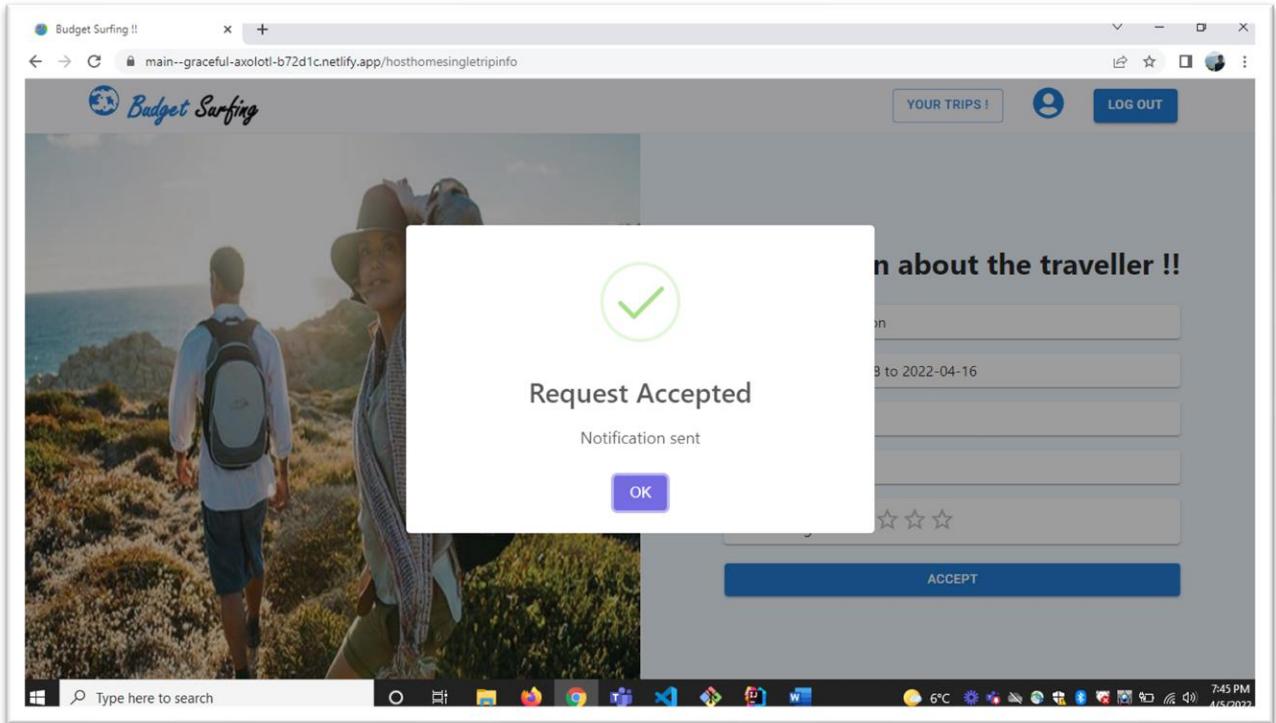
In order for the host to see more details about the traveler, below gives the documentation of how to do that.

## Information about the Traveler: (/hosthomesingletripinfo)

As told in the previous page, if the host wants to know more about the traveler before accepting the traveler, they can do that through this page. The page which gives information about the traveler looks like the below.



This has the details of the trip name, trip start date, trip end date, city name, country name and the user rating. The user rating which we are able to see here is the user rating which will be given by hosts to the traveler. If there are multiple ratings for a user, like if two hosts have given a user of rating 4 and 5 respectively, then the user rating would be shown as 4.5 which is the average of both the ratings. If the host is good with the traveler and is okay to accept the traveler, then the host can do that by clicking the “ACCEPT” button.



After accepting the traveler, a pop-up will be displayed such as this. After clicking the “OK” button, the host will be redirected to the same host home page.

## Trips Accepted by Host Page: (/hostacceptedtrip)

Whenever a host wants to look at all the trips they have accepted, they can do that using this page. What the page shows is as below.

The screenshot shows a web browser window titled "Budget Surfing !!". The URL in the address bar is "main--graceful-axolotl-b72d1c.netlify.app/hostacceptedtrip". The page itself has a header with the "Budget Surfing" logo, a "YOUR TRIPS!" button, and a "LOG OUT" button. The main content area is titled "Trips Accepted by you!". It displays two separate trip entries, each in a green-bordered box:

Trip Details
Trip name : Trip1
City : Hyderabad
Start From : 2022-04-07 to 2022-04-21
Country : IN
<b>TRAVELLER'S DETAILS</b>
Trip name : vacation
City: Abhaneri
Start From : 2022-04-08 to 2022-04-16
Country : IN

At the bottom of the browser window, there is a taskbar with various icons and a system tray showing the date and time as "4/5/2022 8:06 PM".

If there are multiple trips, it will show all the details of all the trips like the above. If the user wants to know more about a particular trip, then they have to select the “TRAVELLERS DETAILS” button which will redirect them to the page (/userprofile).

Budget Surfing

First name: GRM

Last Name: Vishnu

Street: 2309 Brunswick Street

Phone number: 9024128921

Postal Code: B3K2Z1

City: Halifax

Select gender: Male

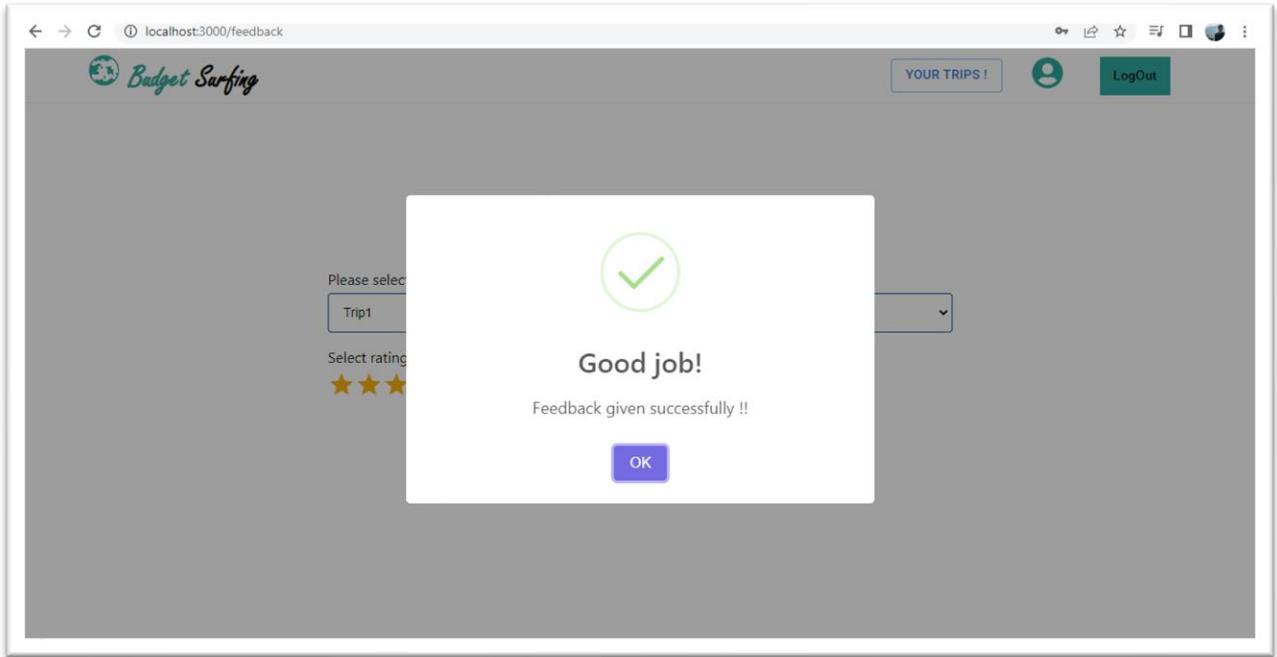
This page shows all the details of the traveller. The host cannot edit any details of the traveler in this form but only can view those details.

## Feedback Page: (/feedback)

This page is used when the host wants to give the feedback about the traveler about their trip. We can get to this page by clicking the “ADD A FEEDBACK” button on the host home page. The form can be seen in the image attached below.

The screenshot shows a web browser window with the URL "localhost:3000/feedback". At the top, there is a logo for "Budget Surfing" and navigation links for "YOUR TRIPS!", a user profile icon, and "LogOut". The main content area is titled "Feedback Form!". It contains two fields: "Please select your trip:" followed by a dropdown menu, and "Select rating for your trip:" followed by a row of five stars. A "Submit" button is located at the bottom right of the form.

This form consists of only two fields. The first one is to select the trip that the host wants to give the feedback of. After selecting the trip, the host can select the number of stars he wants to provide for the traveler and should submit the form using the “Submit” button at the end of the form.



After submitting the form, there will be a pop-up showing that the feedback was given successfully and was added to the user profile of the traveler.