

# Documentation for Transport AI

Dan Coleman,

Jamie Cotter.

Andy Jeffers,

and Avinash Nagarajan.

## Table of Contents

	Title	Page Number
1	Transport AI Vision Statement	3
2	Functional Requirements	4
3	Sprint Backlog	8
4	Use Case Diagrams	10
5	Prototype/Wireframe	12
6	User Stories	19
7	Sprint Retrospectives	21
8	Testing	29

# Vision Statement

For people who want a quick and environmentally friendly way to travel through Ireland TransportAI is an on-demand transportation service that will allow users to order an autonomous electric car to deliver them to a destination of their choice,  
Unlike Taxi services  
our product will allow you to arrive at your destination following an optimal route, without damaging the environment with a fair payment scheme.

## Project Management Tool: Wrike

### Roles:

Dan Coleman: Scrum Master and Documentation

Jamie Leon Cotter: Mobile App Developer

Andy Jeffers: Hardware and Documentation

Avinash Nagarajan: Team Leader and Website Developer

# Functional Requirements

## Application Requirements

*Note: This were written at the beginning and thus a few things changed between this and the final product.*

### *Login page*

- **Must** allow the user to register an account by email by entering users name, email, phone number, password and re-entering the password into each unique text field then hitting the submit button.
- **If the namespace is empty** the user is prompted to place their name into the provided space. The name doesn't need to be unique to each user (they can use their real names) as the email/phone number will be unique, the name is for billing purposes. Highlights the areas that are problematic to login.
- **If the email is empty** the user is prompted to place their email into the provided space. Highlights the areas that are problematic to login.
- **If the email is invalid** the user is prompted to place a valid email into the provided space. Highlights the areas that are problematic to login. This will be caught anyway when the user is prompted on the next screen to input the code they receive through email.
- **If the email has already been used** user is prompted to use a new email or log in to an existing account
- **If phone number is empty** the user is prompted to place their number into the provided space. Highlights the areas that are problematic to login.
- **If the passwords are not the same between the last two fields** the password boxes are reset and the user is prompted to input both again. This will minimise any mistakes.
- **Must** also allow the user to b in through **social media**. This/These will be boxes with the company logo on them. Clicking one will create a pop-up that allows you to sign in to the relevant account (e.g facebook button redirects to facebook, twitter button redirects to twitter).
- **If social media login is successful** user is prompted by site to allow app access to their info as is standard for connecting any application to your account. Skips account validation code page (the four digits sent by email) and brings you directly to the booking.
- **If social media login is unsuccessful** user is brought back to the sign up page. Allowing them to attempt again or sign up by email.
- **Must** allow existing users to log-in using their email and password to get back into an account. As user may have logged out or be reinstalling the app.

### *Sign up validation page*

- **Must** allow user to input the 4 digit code that they are emailed. This makes sure the email the user inputted is their own and not an email they can't access. The 4 digits are separate text fields. The submit button checks to see if the code is the same as what was sent out, a second button "Resend Code" re-sends the email.
- **If the user submits the correct code** the account information is updated to the database. The user can now make bookings.
- **If the user submits an incorrect code** the fields empty and another code is sent out.

### *Settings/Homepage*

- **Shows user** a map near their current location. Ideally integrates with Google Maps.
- Bar on the top right "slide" open on clicking allowing user to choose to Order Vehicle, Add Funds, Change Settings, or call in support. **This is open by default.**
- **If there is a current order**
  - **Must allow user to** Edit their current location
  - **Must allow user to** edit their destination
  - **Must allow user to** quick access the admin support page.
  - **Must allow user to** see current location
  - **Must allow user to** see incoming vehicles current location
  - **Must allow user to** edit number of passengers

### *Ordering Lift*

- **On clicking the order lift button from the homepage must** bring user to a screen where the user is prompted either enter their location.

### *Location Screen*

- **User** is prompted to input current location
- **Must allow user to answer** by entering text into a textfield.
- **Must allow user to answer** by tapping a button to input their current location. This allows the phone to use the users current GPS location from their phone.
- **After selecting a location** the user can hit the next button and is brought to the destination screen.

### *Destination Screen*

- The user is now prompted to enter their destination.
- **Must allow the user to answer** entering text into a textfield
- **Must allow user to answer** by dragging a pinpoint icon onto the map at chosen destination.
- **Must allow** user to specify the number of passengers. The default is one, but clicking on the box a drop down allows you to specify from 1 to 8.
- **After selecting a destination and passenger number** the user can hit the next button to be brought to booking confirmation screen.

### *Booking confirmation screen*

- **Must show** user is chosen Pickup Location, Destination, Number of Passengers to confirm.
- **Must allow user to** edit the above 3 answers.
- **Must allow user to** specify special requests. This is a textarea. This is so that a user can alert an admin if a special vehicle will be needed (e.g the user is in a wheelchair or elderly and needs help to get into the vehicle)

- **Must** show the user the calculated price from point A to point B even before ordering. The price updates in real time on this screen. So for example if a destination is changed so does the price.
- **Must** be *cashless and in-app*. Fare is pre-added to the app before the purchase and is invoiced to the user by phone, email or both (as specified in the settings). Money is taken from the user once a destination is reached.
- **When the destination is reached and user is to leave car** funds are deducted from account.
- **If there is not sufficient funds on the account** the user is given 30 days to add the necessary funds (notified by push notification and email).
- **If the funds are not paid in 30 days** the local authorities are called.

#### *Add Funds Page*

- **Must** show the users current funds.
- **Must allow user** to add funds by credit card.
- **Must allow the user** to set the amount of funds they would like to add.
- **If the card info is incorrect** prompts the user to try again.

#### *Notifications*

- **Must** send a *Push notifications* to users upon order confirmation that informs user on if order goes through successfully.
- **If successful** user is sent the *estimated time of vehicle arrival*, the vehicles registration number, color and make. (To allow user to recognise car)
- **If order fails** a notification is sent apologising that no vehicles are available.
- **Must** allow user to turn off notifications if requested.

#### *Rate us Screen*

- **Must** allow user to *Rate and Review* experience.
- After a ride the user may choose to rates the trip. Ratings are assigned to each car, so Admin can find problematic cars.

#### *Settings Page*

- **Must allow user to** log out
- **Must allow user to** turn off push notifications

#### *Support Screen*

- **Must allow user to contact admin** for assistance if they are having trouble/feel they were unfairly treated.

#### *Additional Feature*

- **If** a nearby vehicle is to pass the destination of another nearby customer **then** both customers are offered to use the same vehicle and share the payment. **Must** allow the user to *contact admin for support* if wanted.

## Hardware Requirements

- Mount 1Sheeld device on arduino board
- Install 1Sheeld software on a Android device
- Set up 1Sheeld for Android on a Phone to interact with the board
- Configure app and board with necessary sheelds
- **Must** take GPS data from 1Sheeld and store in local variables
- **Must** pull user name from Facebook and store as username locally
- **Must** use local username to update database
- **Must** update database every 10 seconds with new location data

## Website Requirements

### Login Page

- **Must** allow Administrator to securely *login to a admin account* that allows access to vehicle schedule and status
- **Must** allow Admin to reset password if he has forgotten theirs.
- **Must show side panel** with information on the company.
- **If password is incorrect** the password box is reset and the user is prompted to input it again.

### Homepage

- **Must show** information for *Incoming Orders, Current Trip, Available Vehicles and Support*
- **Must** show map with real time locations for all currently in use vehicles.
- **Must** show next to Support button, number of unchecked messages.

### Incoming Orders

- **Must** allow *push notifications* for *order alerts* and *vehicle booking data* (such as location, route, payment) for *order updates*.
- **Must** integrate with a service such as Google Maps to get *optimal directions* to customer location and destination.
- **Must** show user table with Vehicle ID, User Name, User ID, Rating, Location Coordinates, and Destination Coordinates
- **Must** allow Admin to *deny or accept incoming orders* with red X and green tick buttons next to each request. Upon accepting it is moved to the current trip table. (If viable user should need to click twice)
- **If user clicks on a request on the table** the map should update to show user's current location, destination, and the location of the nearest vehicle available.
- **Must allow user to change** Vehicle to send to requested lift.

### Current Trips

- **Must show a table with** *Trip ID, Vehicle ID, Customer ID, Location Coordinates and Destination Coordinates*
- **Must show all vehicles in use** on map with real time tracking.
- **If a trip is clicked on on the table** the *Vehicle info* (ID, reg no, color, make, maintenance info, Rating) and *Customer Info* (ID, Name, Rating) should be available

in formatted text [note: think web4 student grades] and the **Map should show the pickup location, vehicle location, destination, and route.**

#### **Available Vehicles**

- **Must show a table with** *Vehicle ID, Reg, Color, Make, Maintenance Info and Rating* for cars currently with no passengers.
- The Map **must** show the current location of cars currently with no passengers.

#### **Support**

- **Must** have a *support function page* to facilitate interaction between admin and customers if there are any issues with Cost estimation, reports, etc.
- Site **Must** update from any page notification of new support request [think facebook messenger]
- Page **must** show different user names for support request with a chat log on the right. (see prototype for more info)

#### **Additional Feature**

- **Must** track order and **If** a nearby vehicle is to pass the destination of another nearby customer **then** a push notification should be sent out to offer users to share the vehicle at a reduced rate.

## **Sprint Backlog**

### **Application Backlog**

1. Create Login Gui
2. Sign in via Email and Password
3. Sign in users using Firebase Authentication
4. Connect Application to Firebase
5. Create Gui for Rate Us screen
6. Develop Database Connection for App
7. Create Sign Up Gui
8. Register Account through us
9. Create Homepage GUI
10. Get real-time database info from phone GPS
11. Clean up Previous GUI'S
12. Get Google Maps functionality displaying in fragment view.
13. Logging into Firestone using Social Media
14. Fix the Gui alignment issue.
15. Logging into Firestore using Social Media.
16. Implement Directions API.
17. Fix GUI alignment issue.
18. Create Payment GUI.
19. Build payment fragment.
20. Test Application.
21. Implement Stripes payment feature.
22. Implement trip completion feature.



23. Implement confirmation before ride.
24. Implement funds deduction on completion of ride.
25. Add functionality to query if a vehicle has been assigned to user.
26. Create Vehicle Arriving Fragment.
27. Create Vehicle Arrived Fragment.
28. Add functionality for monitoring the users location in respect to the destination.
29. Create Trip Finished Fragment.
30. Create Rate Us Fragment.
31. Rework of Sign Up Activity.
32. Implement Log out.
33. Create on Route Fragment.
34. Increase accuracy of distance calculator.

## **Database Backlog**

1. Set up Firebase and Firestore.
2. Make Admin Collections.
3. Make User Collections.
4. Make Vehicle Collections.
5. Make Trip Collections.
6. Add dummy Collections.

## **Hardware Backlog**

1. Get GPS coordinates using Arduino and 1Sheeld
2. Update Firebase with GPS Coordinates to correct location

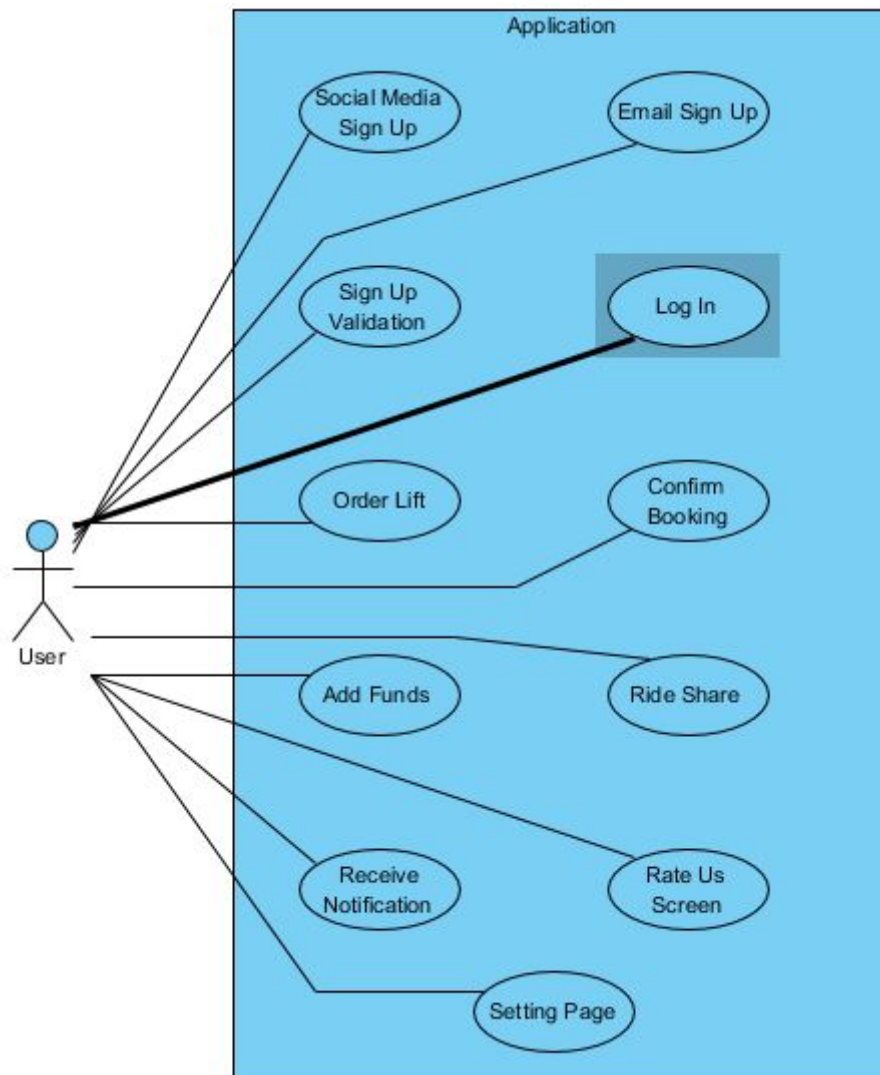
## **Website Backlog**

1. Securely log in by giving my admin ID and password.
2. Implement landing page with info on the trips.
3. Implement Google Maps API.
4. Display Customer location on Google Maps.
5. Display Customer details.
6. Implement directions in relation to the position of the car.
7. Display the vehicle location on the map.
8. Edit Vehicle details.
9. Enter Dummy data into Firebase.
10. Edit Customer details.
11. Update Google Maps based on Firestore Coordinates.
12. Display Vehicle Info.
13. Implement Secure login with route protection.
14. Implement Logout.
15. Add logo to the website.
16. Allow Admin to change the cost per kilometer.
17. Display highest rated vehicle.

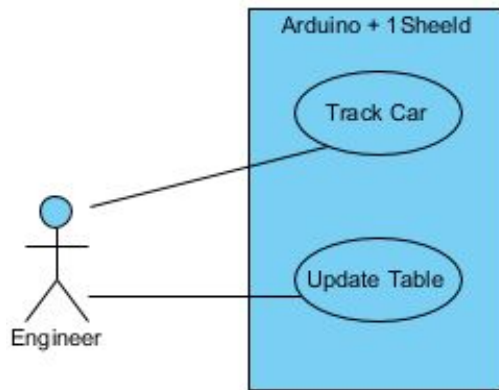
18. Display most valuable customer.
19. Added logo to the website tab.
20. Testing the Website.
21. Dynamically allocated nearest car to a trip order.
22. Add Contact Page.
23. Implement Stripe on the server end.
24. Fine Tune the Rating.

## Use Cases

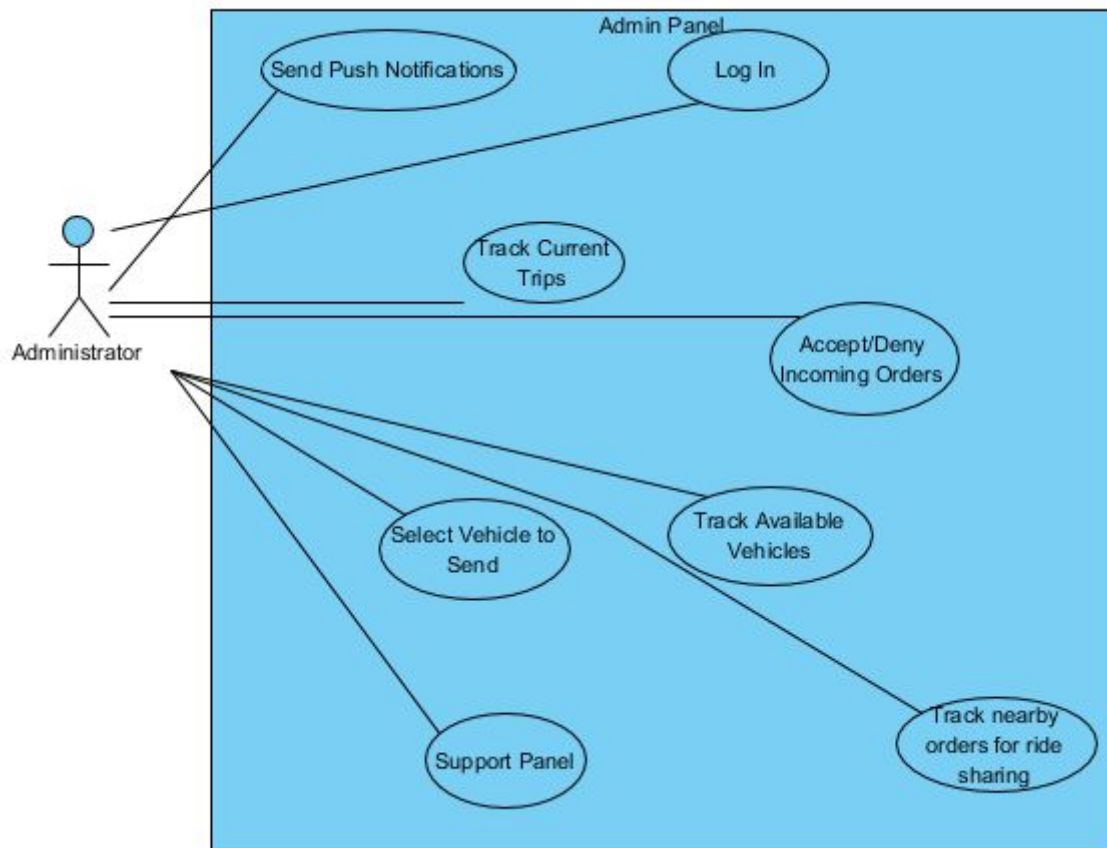
### Application



## Hardware



## Website



# Prototype/Wireframe

## Application Prototype

### Login Screen



Log In

Email:

Password:

Don't have an account? [Sign up](#)

OR SIGN IN USING SOCIAL MEDIA



### Sign up screen

SIGN UP

Enter Name:

Enter Email:

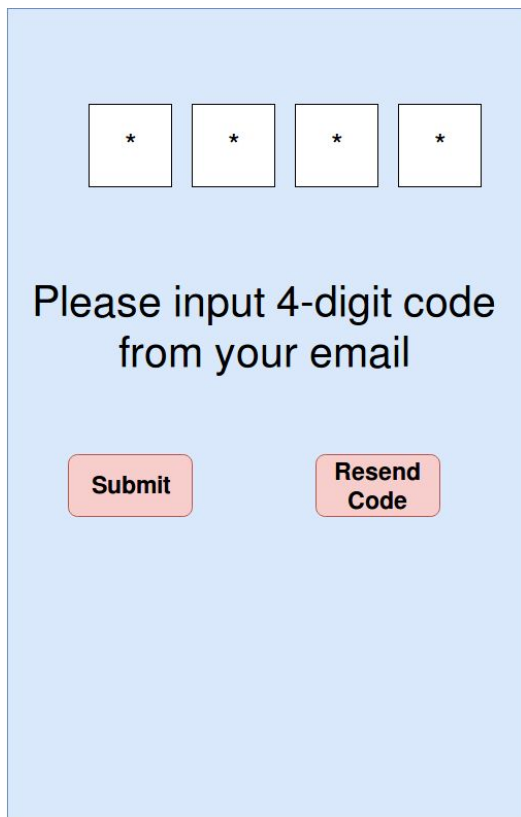
Enter Phone Number:

Enter Password:

Re-enter Password:

Already have an account? [Log in](#)

## Sign in Validation

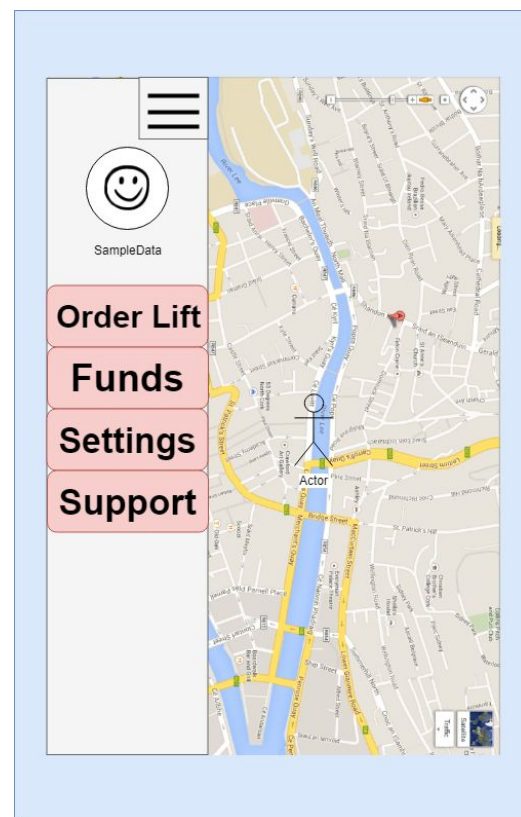


Four input fields, each containing an asterisk (\*), for a 4-digit code.

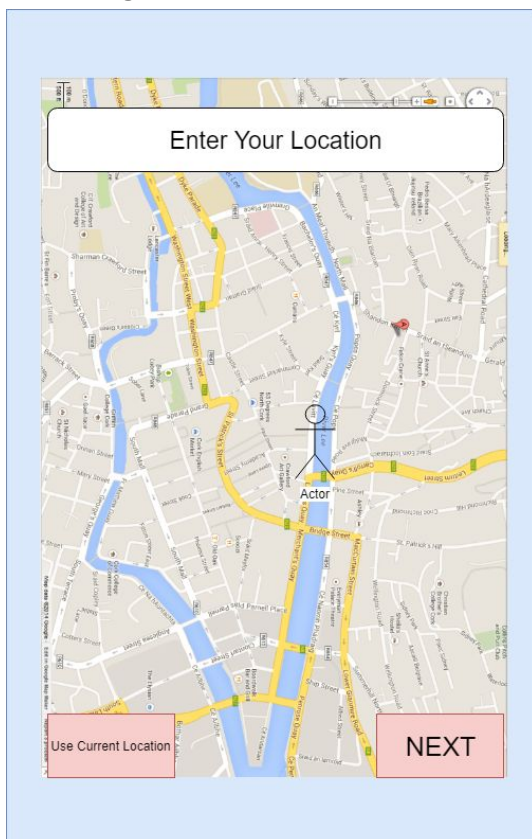
Please input 4-digit code  
from your email

Submit Resend Code

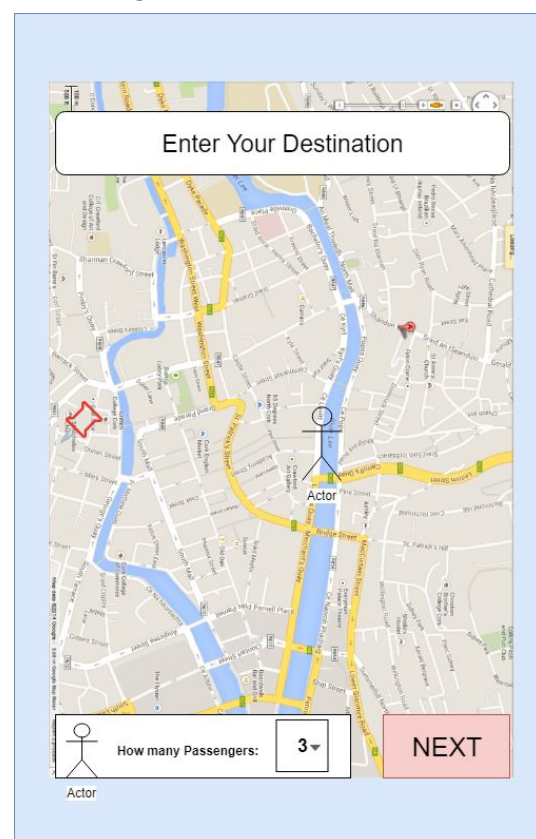
## Homepage/Navigation Bar



## Choosing User Location



## Choosing User Destination



## Confirmation Screen

**CONFIRMATION**

Pickup Location

1 Sample Street, Road Lane, Cork

Destination

Cork Airport, Cork

No. of Passengers:

3

Special Requests:

Wheelchair, Elderly ,etc.

Fare Total:

€8.00

**CHECKOUT**

## Funds screen

**FUNDS**

**Current Funds: €25.00**

Please set the amount of funds you would like to add to this account :

€0.00

**ADD FUNDS**

**BACK TO HOMEPAGE**

## Payment Screen

**THANK YOU!**

**You have reached your destination!**  
**The fare (€8.00) has been deducted from your funds.**  
**Thank you for riding with TransportAI**

**Enjoy your trip?** [Rate Here](#)


**HOMEPAGE**


## Rating Page

**RATE THE TRIP**

You were travelling with Vehicle **A1**

Were you satisfied with your service today?





Leave a comment (optional)...

**BACK TO HOMEPAGE**



## Tracking vehicle and Push Notification

placeholder logo

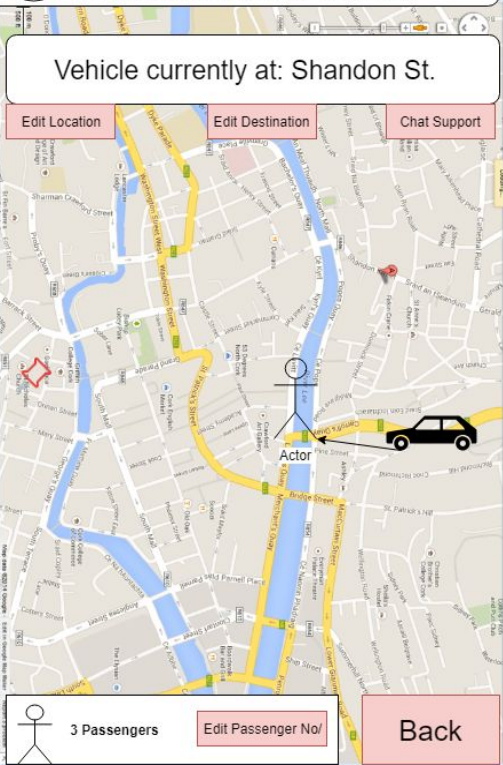
Vehicle A1 is now on its way!  
Estimated time of arrival: 5 minutes  
Reg:182-C-1234 Color: Grey Make: Tesla

Vehicle currently at: Shandon St.

Edit Location

Edit Destination

Chat Support



3 Passengers

Edit Passenger No/

Back

Actor

## Settings Page

SETTINGS

Push Notifications:

☐ Send me all notifications

☐ Only send me notifications with info on incoming pickup vehicles

☐ Do not send me notifications

BACK TO HOMEPAGE

LOG OUT

## Website Prototype

### Login

Welcome to the Admin Panel.

Admin ID:

Password:

Log In

[Forgot my password](#)

Who are we?

LOGO HERE

TransportAI is Irelands easiest and most environmentally friendly transportation service

...

Homepage

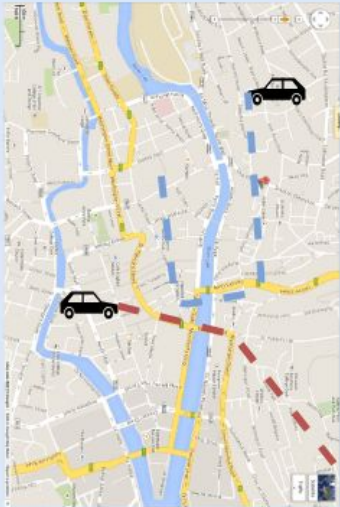
Welcome to the Admin Panel.

Incoming Orders

Current Trips

Available Vehicles

Support



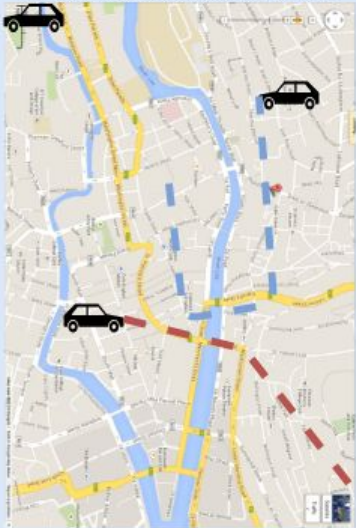
Incoming Orders Main Page

Welcome to the Incoming Orders.

Customer ID	Name	Vehicle ID	Location	Destination
ZZ0001	Mikey Purcell	A1	Cork Airport Business Park	The Linen Weaver, Paul St.
ZZ2946	Steve Benoit	W12	Main Campus, UCC	The Linen Weaver, Paul St.
SO1337	James Potter	W1	12 Killeagh Orchard	CIT Maritime Campus

Back

Closest Vehicle assigned by default





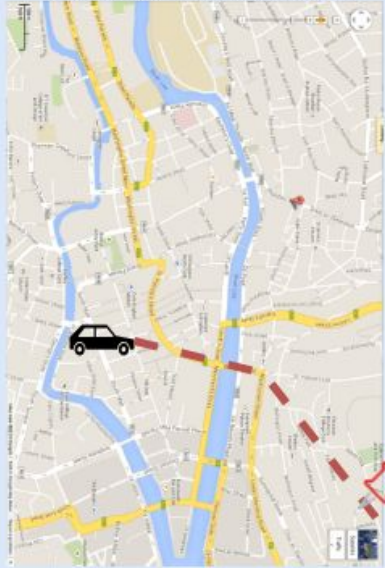
## Incoming Orders Specific Vehicle

### Welcome to the Incoming Orders.

Customer ID	Name	Vehicle ID	Location	Destination		
ZZ0001	Mickey Purcell	A1	Cork Airport Business Park	The Linen Weaver, Paul St.	✓	✗
ZZ2346	Steve Benoit	W12	Main Campus, UCC	The Linen Weaver, Paul St.	✓	✗
SO3337	James Potter	W1	12 Killalea Orchard	CIT Maritime Campus	✓	✗

Back

### Vehicle Currently assigned to order




## Current Trips Main Page

### Welcome to the Current Trips.

Trip ID	Vehicle ID	Customer ID	Location	Destination
A001	A2	AA0001	24 Meadow Park Ave.	Apache Pizza City Centre
A002	A13	MY1101	Main Campus, UCC	Leisure World, Bishopstown
A003	W1	WW3333	Admin Building, CIT	CIT Maritime Campus

Back



Clicking on a specific trip in Current Trips

### Welcome to the Current Trips.

**Trip: A001**

*Vehicle Information*

ID: A2

Registration Number: 182 - C- 1111

Color: Grey    Make: Tesla

Requires Maintenance? No

Rating: 100% Positive

*Customer Information*

Name: Joe Bloggs      Customer ID: A0001

Back

Available Vehicle Main Page

### Welcome to the Available Vehicles.


	Vehicle ID	Registration No.	Color	Make	Needs Maintenance?	Rating - % Positive
1	A1	181-C-10	Grey	Tesla	FALSE	100
2	A3	181-C-11	Grey	Tesla	FALSE	100
3	A4	181-C-12	Grey	Tesla	FALSE	98
4	W2	182-C-1	Red	Tesla - Wheelchair Access	FALSE	100
5	W3	182-C-2	Red	Tesla - Wheelchair Access	FALSE	100
6						
7						
8						



Back


Vehicles not with passengers

## Support Page

### Welcome to the Support Panel.

 **Dan Coleman**

 **Jamie Cotter** 

 **Andy Jeffers**


Back

### You are currently chatting to Dan

Avinash: Hello, this is Avinash from TransportAI. How may I help you today?

Dan: I have an issue with adding funds.

Enter your text here ...



# User Stories

## Application:

1. As an unregistered user, I want to be able to register an account by email by entering users name, email, phone number, password and re-entering the password into each unique text field then hitting the submit button.
2. As an unregistered user, I want to be able to sign in through social media.
3. As an existing user, I want to be able to log-in using my email and password
4. As an unregistered user I want to be able to confirm my account by inputting the 4 digit code I am emailed.
5. As a user I want to be able to see my location on a map.
6. As a user I want to be able to easily access features of the app from the navigation bar.
7. As a user, I want to be able to update my request in real time so that I can change my destination if needed.
8. As a user, I want to be able to order a vehicle to my location by inputting text.
9. As a user I want to be able to use my phones gps to input as my location.
10. As a user, I want to be able to order a vehicle to my destination by inputting text.
11. As a user, I want to be able to order a vehicle to my destination by dragging the icon to a given spot.
12. As a user, I want to be able to choose the size of my vehicle so that I can choose a suitable vehicle for my number of passengers.

13. As a user, I want to be able to give any special specifications needed (e.g Wheelchair).
14. As a user, I want to be able to confirm my information before ordering.
15. As a user, I want to be able to edit my information before ordering.
16. As a user, I want to be able to see the price of my lift before ordering.
17. As a user, I want to be able to add funds to the application using my credit card.
18. As a user, I want to be able to check my current funds.
19. As a user, I want to be able to rate my lift and leave comments for the admin.
20. As a user, I want to be able to get push notifications to get information on the car, so that I will recognise the car when it comes to collect me.
21. As a user, I want to be able to be able to track the car so that I know how long I will be waiting.
22. As a user, I want to change my notification settings.
23. As a user, I want to be able contact the admin for support.
24. As a user I want to be able to log out of the application.
25. As a user, I want to be offered the chance share a car with other users going the same route as me at a reduced rate.

### **Admin Panel:**

1. As an admin I want to be able to securely log in by giving my admin ID and password.
2. As an admin, I want the panel to integrate with a service like Google Maps allowing for the vehicles to take the optimal routes to their destination.
3. As an admin, I want the panel to integrate with a service like Google Maps allowing me to see each of the vehicles at once, and update their location in real time (every 10 seconds).
4. As an admin, I want to be able to accept/deny incoming orders.
5. As an admin, I want to be able to change the vehicle to send to a user. [auto picks nearest]
6. As an admin I want a homepage with quick access to all important areas of the site.
7. As an admin, I want to be able to access the info of the Vehicles and user by clicking on a current trip and view their path on the map (Vehicle, Customer, Destination, Location, No.of Passengers).
8. As an admin, I want to be able to access specific user info (ID, name, history, special request) by clicking on a user from the trip table.
9. As an admin, I want to be able to see specific vehicle info (ID, reg no, color, make, maintenance info, Rating) by clicking on a vehicle from the trip table.
10. As an admin, I want to be able to see available vehicle info (ID, reg no, color, make, maintenance info, Rating) on a database table.
11. As an admin, I want to be able to communicate with users that require assistance with our service [support].
12. As an admin I want to be able to send push notifications about our vehicles status and destination [auto sends on accept].
13. As an admin I want to be to able to offer multiple users the ability to share a vehicle if they desire at a reduced rate.

## Hardware:

1. As an engineer I want to be able to provide real time tracking of the car through the Arduino.
2. As an engineer I want to be able to access the GPS Coordinates of the user through the Arduino.
3. As an engineer I want the details each party needs be accessible from the Arduino.

# Sprint Retrospectives

## Application Retrospectives

### Sprint 1

Jamie worked alone on this Sprint as I was setting up our documentation, but we were happy how it went. Nobody in our group had touched App Development before, so we were happy that he got the Login GUI and a start into the functionality done all in one week along with having to read up on how to do it.

**Sprints Completed:** Create Login Gui

**Sprints Incomplete:** Sign in via Email and Password

### Sprint 2

This week Dan was learning how to build GUI's as Jamie wanted him to build the GUI's during the early week that Jamie would expand upon during the later week. Jamie also finished up the Sprint on the Log In using existing account. We cancelled our Sprint for Developing Database connection for the Application after we decided it was easier to use firebase (our original plan was PostgreSQL and Amazon Web Service). He also set up the firebase and allowed users on the firebase to log in.

Overall, we were very happy with this Sprint.

**Sprints Completed:** Sign in via Email and Password

Sign in users using Firebase Authentication

Connect Application to Firebase

Create Gui for Rate Us screen

**Sprints Cancelled:** Develop Database Connection for App

### Sprint 3

This week Dan created the GUI for the Signing up feature and wrote a skeleton controller which Jamie used to implement the Signing up feature for the application later in the week. We finished earlier than usual this week as we are getting more used to Android Studio.

**Sprints Completed:** Create Sign Up Gui

Register Account through us

## Sprint 4

This week Dan built the GUI for the Homepage and made the skeleton for the Controller.

There was an issue with the Map not updating but this was fixed.

Jamie implemented the feature of accessing the homepage by signing in. The app now gets the user location from the phones GPS, allows inputting destination and opening/closing the navigation bar.

We just made the Sprint deadline due to the issue of the Map not loading, and are happy with our progress this Sprint.

**Sprints Completed:** Create Homepage GUI

## Sprint 5

This week Jamie implemented the feature that updates the user's current location on the map in Real Time (the planned implementation for the final product will be every ten seconds). Dan updated the older GUIs as they were looking plain, after finishing the feature Jamie also helped in fixing GUI's.

We didn't get too much work done this week as we had assignments due, but that was by recommendation of our Supervisor.

**Sprints Completed:** Get real-time database info from phone GPS

Clean up Previous GUI'S

## Sprint 6

Jamie had run into a pretty big error where he couldn't get the Map to load into fragment view which is the recommended way of going about loading the maps, so he had to take the week to look into this error. As it impeded any further development of the main function of our app (ordering vehicles to pick up/drop a user).

Fortunately it was eventually solved.

Dan started researching and implementing the Social media login as it wasn't a major priority and would train him up in Kotlin, this took longer than expected as his laptop broke and he was forced to buy a new one. This meant he missed his deadline and so the task carried over to his next Sprint.

We originally put Andy on fixing our issue where our GUI didn't line up correctly on certain phones, but our supervisor had suggested we add dummy data to the database as it made our website look better, so we put the issue on hold and moved him to creating dummy data.

We weren't very happy with the work done in this Sprint, but it is incredibly unlikely we will have another computer break on us.

**Sprints Completed:** Get Google Maps functionality displaying in fragment view.

**Sprints Incomplete:** Logging into Firestone using Social Media

**Sprints Cancelled:** Fix the Gui alignment issue.

## Sprint 7

This week Dan spent the week working on the Log in through Social media feature that was carried over from last week. His new laptop showed up which massively sped up unnecessary wait times like boot times, compile times, and loading up Android Studio. He ran into an issue later in the week where there was an API exception that wasn't fixing no matter what but eventually we realized it was to do with the Android Emulator. Dan Finished his Sprint just in time.

Jamie implemented Directions API much faster than we had expected him to and took the extra time to solve a known bug we had where the GUI's were not scaling to different phone sizes. He also helped Dan catch the error with the Android Emulator allowing him to finish his Sprint in time.

Originally Andy was to create the GUI for the Payment screen however as Jamie had been moving things to fragments he decided it wasn't a good idea to make the gui until we were certain what we needed, so he spent the week fixing up the Wrike

**Sprints Completed:** Logging into Firestore using Social Media.

Implement Directions API.

Fix GUI alignment issue.

**Sprints Cancelled:** Create Payment GUI.

## Sprint 8

This week Dan went through the app and did the testing criteria that was decided on by the team and supervisor (screenshots of each feature working) he did this for all completed features of the app (up to Sprint 7) and added the completed pictures to the Wrike to prove each feature worked.

Jamie decided since he was the most comfortable with fragments that it would be faster and easier if he built it, after completing that he got a good part of the way into implementing Stripe but we eventually needed a payment server which was on Nash's end so we put it on hold for the time being.

Overall we are happy with our progress.

**Sprints Completed:** Build payment fragment.

Test Application.

**Sprints On Hold:** Implement Stripes payment feature.

## Sprint 9

This week was our final push after Christmas to finish the app. This week Jamie got a massive amount of work done. He finished up the section in relation to going on trips. This included making the admin approve a ride before it starts, along with a way for the ride to be considered complete (it now is complete once the vehicle is within 10m of the destination). A vehicle is now assigned to the user upon the ride being approved. Once you arrive at the destination the funds are deducted from your account and you are given the option to Rate your vehicle, this allows the admin to know which cars are performing well.

He also implemented a Log out feature as was required in the spec that allows the user to change accounts. He also increased the accuracy of the distance guessing calculator.

Fragments were also created by Jamie for tracking the incoming vehicle, travelling in the vehicle, the finishing ride, the ride ended screen and optional rate us screen.

He also rewrote the Sign Up Activity, he replaced it with a better looking fragment that is overlaid onto the log in GUI.

Overall we are very impressed with the work he managed to get done this week.

**Sprints Completed:** Implement trip completion feature.

Implement confirmation before ride.

Implement funds deduction on completion of ride.

Add functionality to query if a vehicle has been assigned to user.

Create Vehicle Arriving Fragment.

Create Vehicle Arrived Fragment.

Add functionality for monitoring the users location in respect to the destination.

Create Trip Finished Fragment.

Create Rate Us Fragment.

Rework of Sign Up Activity.

Implement Log out.

Create on Route Fragment.

Increase accuracy of distance calculator.

## Database Retrospectives

### Sprint 1

This week Nash set up the database for our project (using Firebase) and added 4 collections Admin (which is a list of administrators), User, Vehicle (which store info on the User and Vehicle respectively) and Trip (which store info on the trip such as destination).

We are happy with the progress of this sprint.

**Sprints Completed:** Set up Firebase.

Make Admin Collections.



Make User Collections.  
Make Vehicle Collections.  
Make Trip Collections.

## Sprint 2

This week Andy was told to add dummy data to the Firebase as Seamus thought it was be a nicer display for the website. He added 15 new vehicles and 15 new customers.

**Sprints Completed:** Enter Dummy data into Firebase

# Hardware Retrospectives

## Sprint 1

This week Andy setup the Arduino to access the GPS Coordinates of the car from an Android phone, via the 1 Sheeld board and app.

Andy finished this Sprint early. Overall we are happy with the progress on the Sprint.

**Sprints Completed:** Get GPS coordinates using Arduino and 1Sheeld

## Sprint 2

This week Andy started updating the firebase with the information from the Arduino GPS. He ran into an issue with Authentication tokens for the database. He worked on fixing this issue through the week. This was mandatory to fix for progress to continue.

Overall we are not so happy with how this Sprint went, but we expected to run into a few issues, progress will speed up once this is solved.

**Sprints Incompleted:** Update Firebase with GPS Coordinates

## Sprint 3

This week Andy continued updating the firebase with the information from the Arduino GPS. He continued solving the issue with Authentication tokens, but unfortunately while he solved the issue he ran out of time before finishing the implementation of the fix.

Overall this Sprint could have went better, but at least we solved the issue. Progress is expected to go faster.

**Sprints Incompleted:** Update Firebase with GPS Coordinates

## Sprint 4

Sprint Review: This week Andy continued updating the firebase with the information from the Arduino GPS.

This week he implemented the fix for the Authentication early in the week. We made progress but needed to get further in the other sections before we knew the data structure we would need to send.

Overall we are happy with the progress on this sprint. In retrospect we should have probably started the hardware section a week later, but we have learned from this.

**Sprints Completed:** Update Firebase with GPS Coordinates

## Sprint 5

This week Andy continued updating the firebase with the information from the Arduino GPS. This week we knew the data structure required and Andy implemented sending the GPS information in this type. We thought we had an issue with the type not sending correctly, but after a few hours of investigating we realized the firestore was indeed sending correctly. Overall this Sprint could have went better, but we weren't the only group that made this mistake. It is reassuring to know everything is working correctly. Andy has been moved to creating dummy data in the firebase.

**Sprints Completed:** Update Firebase with GPS Coordinates to correct location

# Website Retrospectives

## Sprint 1

In this week Nash began implementing the ability to securely log in. He started by reading up on React Framework and used this knowledge to implement the user interface groundwork by making the State and Prop for the log in gui. He did not finish the implementation but as he had to learn React.js from scratch we are very happy with how this first sprint went. As he gets more familiar with the technology he will speed up the workflow.

**Sprints Incompleted:** Securely log in by giving my admin ID and password.

## Sprint 2

In this week Nash finished up the Login feature. You can now log in using the successful credentials and validation is implemented for incorrect log in. He also created a very basic landing page just to show the login was successful. The Sprint was finished on time, so we are getting better at estimating completion times. Overall we are happy with this Sprint

**Sprints Completed:** Securely log in by giving my admin ID and password.

## Sprint 3

This week Nash completed the landing page. You can now check the trip/vehicle information. Next Sprint will let you load up specific details by hitting the details button, but for now you can see it in a table view. This Sprint was finished on time. We are happy with how this Sprint went.

**Sprints Completed:** As an admin I want to see a landing page with info on the trips

## Sprint 4

This week Nash implemented the details button. Now when you click details next to a trip in the display it loads up a prop with more information on the selected item. He also implemented the Google Maps API allowing Admins to see the location of the user and vehicle on Google Maps.

He finished this Sprint in the time we allotted we are happy with this Sprint.

**Sprints Completed:** Implement Google Maps API

Display Customer location on Google Maps

Display Customer details

## Sprint 5

This week Nash implemented the Directions of the Customer from the Car into the Google Maps for the Admin Panel. This allows the map to display distance from car to customer.

**Sprints Completed:** Implement the Google Directions API into Google Maps

## Sprint 6

This week Andy was told to add dummy data to the Firebase as Seamus thought it was be a nicer display. He added 15 new vehicles and 15 new customers.

Nash implemented the ability to edit customer and vehicle details through the panel, the panel now also displays information on the vehicles and updates there location in real time. The google maps locations now also update in real time through the info from the firebase. Overall we got a ton of work done on time this week, I don't think this Sprint could have gone better.

**Sprints Completed:** Display the vehicle location on the map.

Edit Vehicle details.

Enter Dummy data into Firebase.

Edit Customer details.

Update Google Maps based on Firestore Coordinates.

Display Vehicle Info.

## Sprint 7

This week Nash implemented a safer log in system. Before you could bypass the login page, now you cannot access the other props before you log in.

He also added a log out button that allows an admin to log out.

This Sprint was finished on time, we are happy with how this Sprint went.

**Sprints Completed:** Implement Secure login with route protection.

Implement Logout.

## Sprint 8

This week Nash added 3 boxes to the homepage as per Seamus' suggestion to fill up a blank area.

One calculated and displayed the customer that takes the most trips with us, the second one allows an Admin to change the cost per km traveled in real time, and the 3rd one calculated and displayed the vehicle with highest user rating.

He also added the company Logo the team had designed into the Admin Panel.

Overall we are very happy with this Sprint, it was finished in the allotted time.

**Sprints Completed:** Add logo to the website.

Allow Admin to change the cost per kilometer.

Display highest rated vehicle.

Display most valuable customer.

## Sprint 9

This week Nash added the logo to the tab.

He also set up the web app on everyone's device for testing.

Dan and Andy went through and tested the website features up until now.

**Sprints Completed:** Added logo to the website tab.

Testing the Website.

## Sprint 10

This week was our final push after Christmas to finish the website. Nash managed to get a lot more done as we had no other projects. This week he got the server side of Stripe payments working, he added the contact us page for users looking for support. He added the feature where the website can now dynamically allocate the nearest available vehicle to the user when the ride is accepted by the admin. Finally he edited the Ratings system to allow it to work in tandem with the application.

Overall we are very happy with the work we got finished this week.

**Sprints Completed:** Dynamically allocated nearest car to a trip order

Add Contact Page

Implement Stripe on the server end

Fine Tune the Rating

# Testing

## Application Testing

### Sign Up Validation

**Sign Up**

Ron Smith

ron@email.com

Password

Confirm Password

0876543210

**SUBMIT**

Please do not leave ANY field Empty.

Don't have an account? [Sign Up Here](#)

**Sign Up**

Ron Smith

ron@email.com

Password

Confirm Password

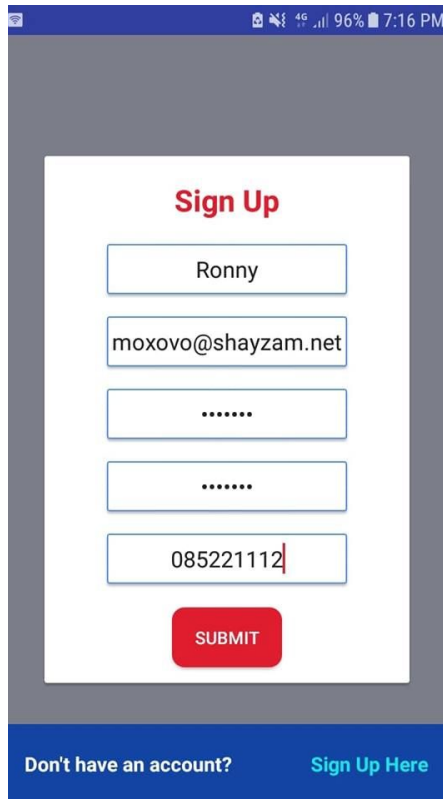
0852828122

**SUBMIT**

Make sure passwords match

Don't have an account? [Sign Up Here](#)

## Sign Up Successful: Data Input and appears on Database



A mobile application interface for signing up. At the top, the status bar shows 4G signal, 96% battery, and 7:16 PM. The app title "Sign Up" is in red. Below it are five input fields: a name field with "Ronny", an email field with "moxovo@shayzam.net", two password fields with masked characters ".....", and a phone number field with "085221112". A red "SUBMIT" button is at the bottom of the form. Below the form, there is a link "Don't have an account? Sign Up Here" in blue.

**Sign Up**

Ronny

moxovo@shayzam.net

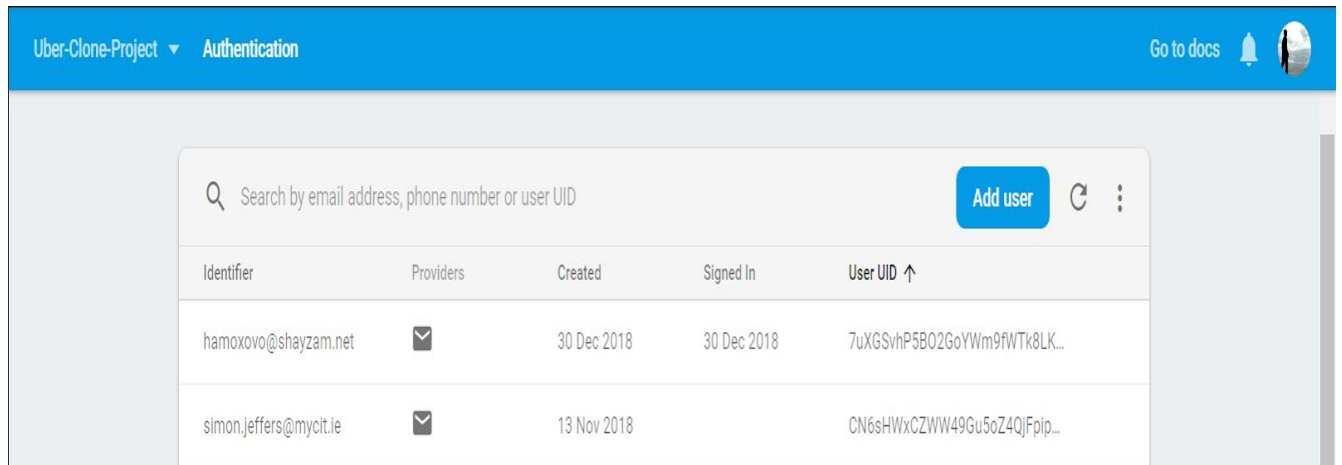
.....

.....

085221112

**SUBMIT**

Don't have an account? [Sign Up Here](#)



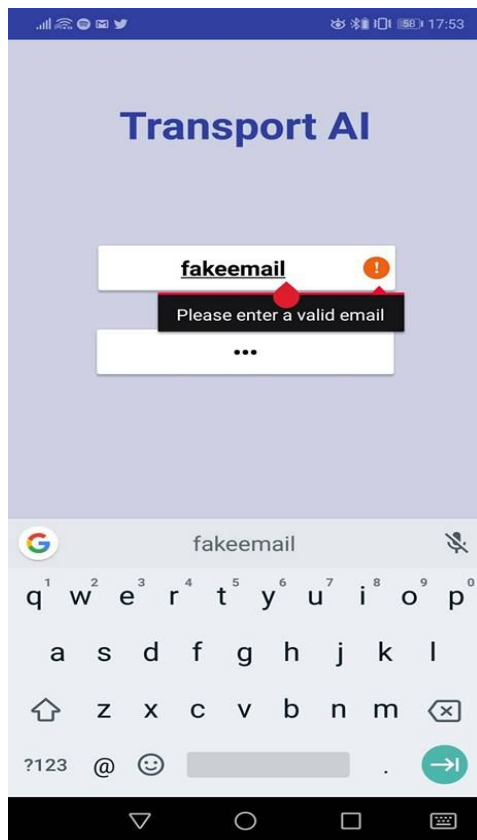
A web dashboard interface for user management. The top navigation bar is blue with "Uber-Clone-Project" and "Authentication" on the left, and "Go to docs" and a user profile icon on the right. Below the navigation bar is a search bar with the placeholder "Search by email address, phone number or user UID" and an "Add user" button. Below the search bar is a table with columns: Identifier, Providers, Created, Signed In, and User UID. The table contains two rows of user data.

Uber-Clone-Project Authentication Go to docs

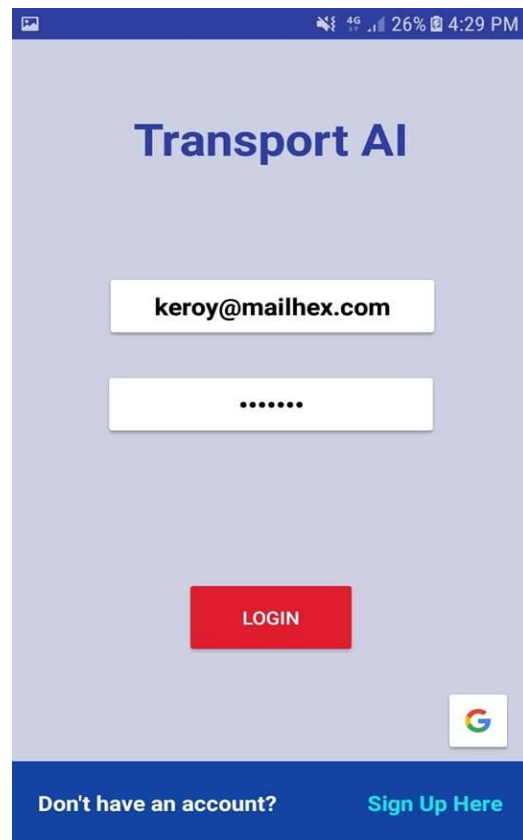
Search by email address, phone number or user UID Add user

Identifier	Providers	Created	Signed In	User UID ↑
hamoxovo@shayzam.net	📧	30 Dec 2018	30 Dec 2018	7uXGSvhP5B02GoYWM9fWTk8LK...
simon.jeffers@mycit.ie	📧	13 Nov 2018		CN6sHWxCZWW49Gu5oZ4QjFpip...

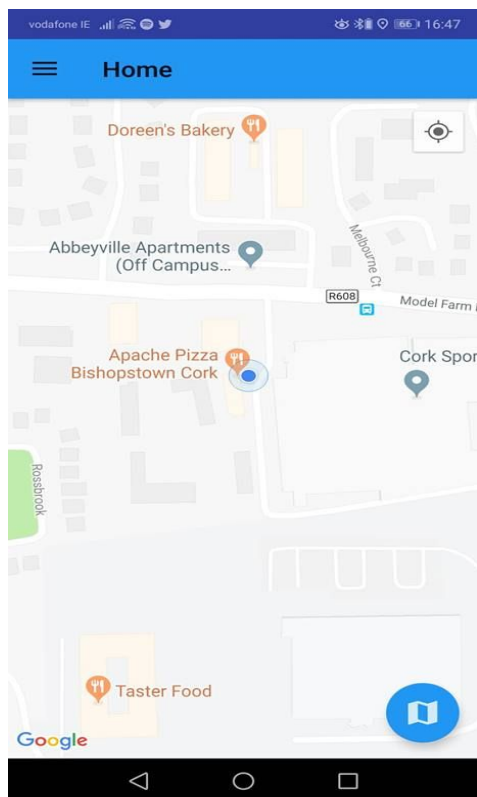
Login: Picture 1: invalid email.



Picture 2: Inputting successful email

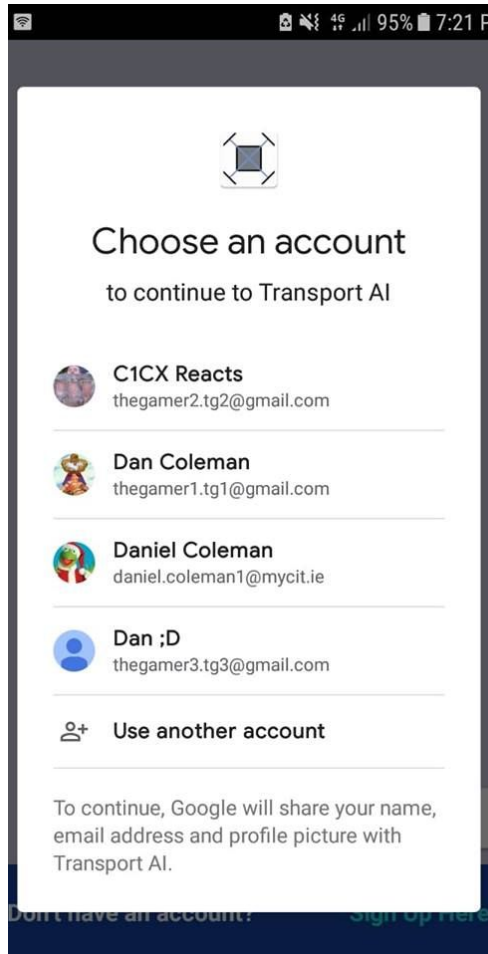


Picture 3: Login Successful brought to homepage

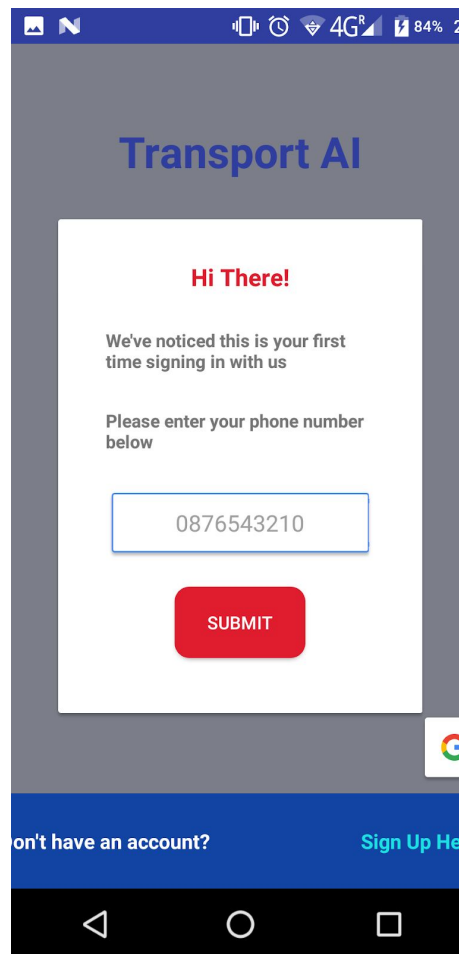


**Social Media Sign-up/Login:** First the google logo is clicked [see Picture 2 above]

Picture 1: Account is selected

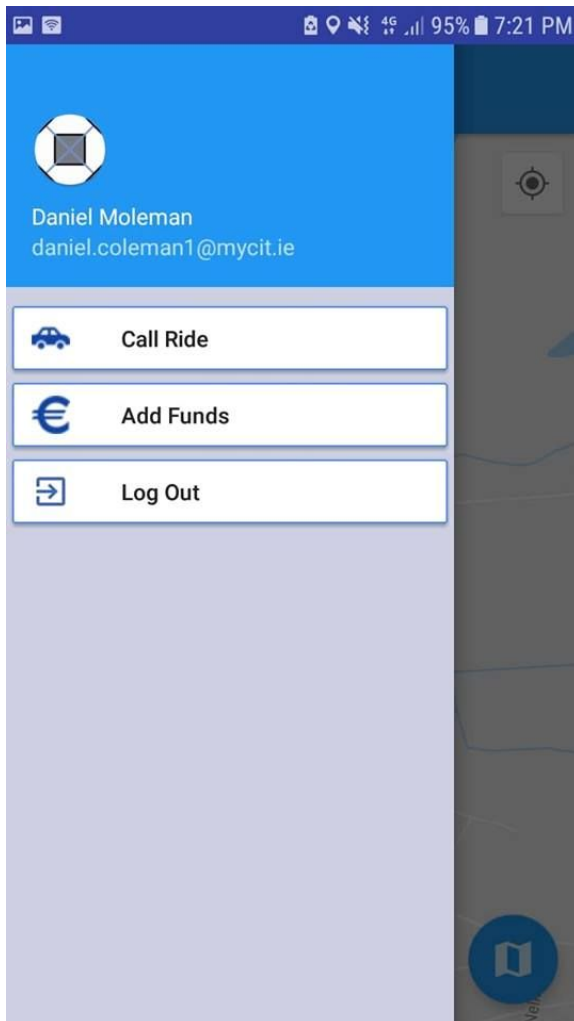


Picture 2: Prompted for mobile if new user



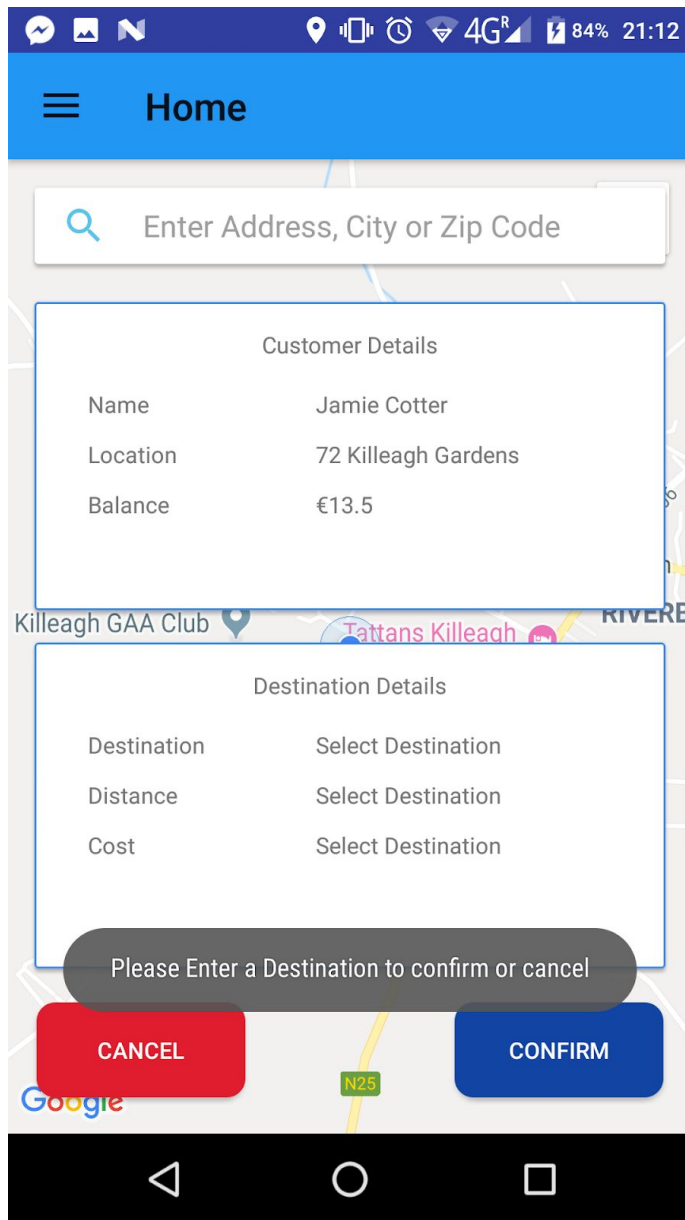


Picture 3: Brought to Homepage

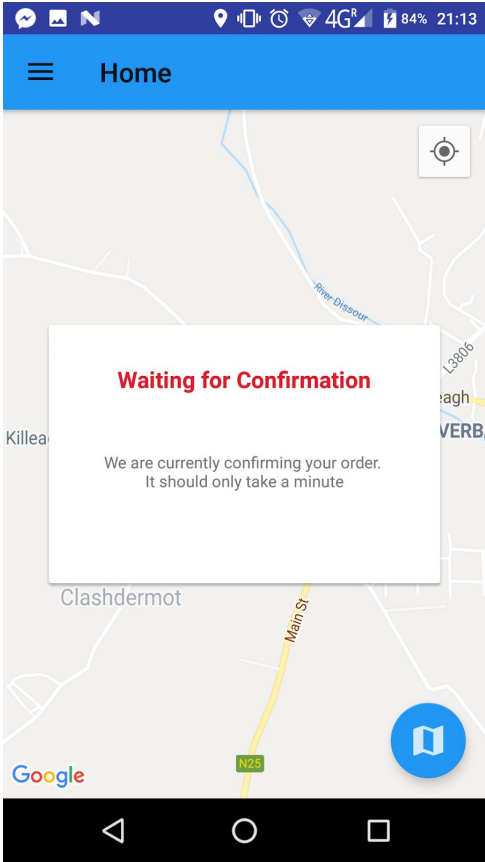


**Calling a ride:** Location is already taken from GPS[See login picture 3].  
Then use the nav bar to call a ride [See sign-up/login for social media picture 4]

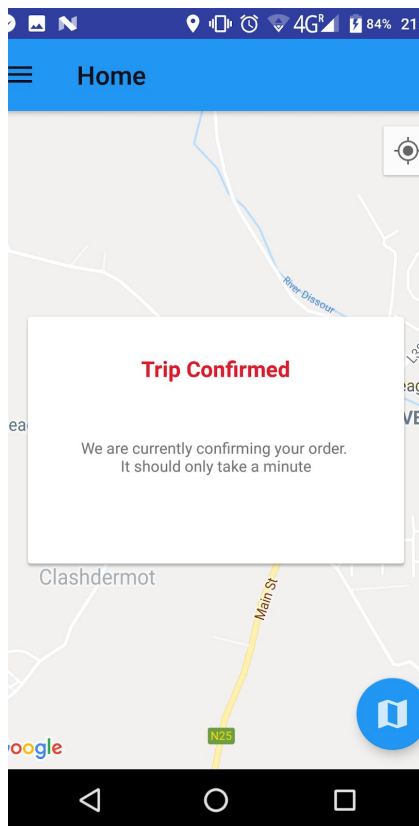
Picture 1: Shows customer details and prompts destination and validation if you try to hit confirm before the destination is chosen.



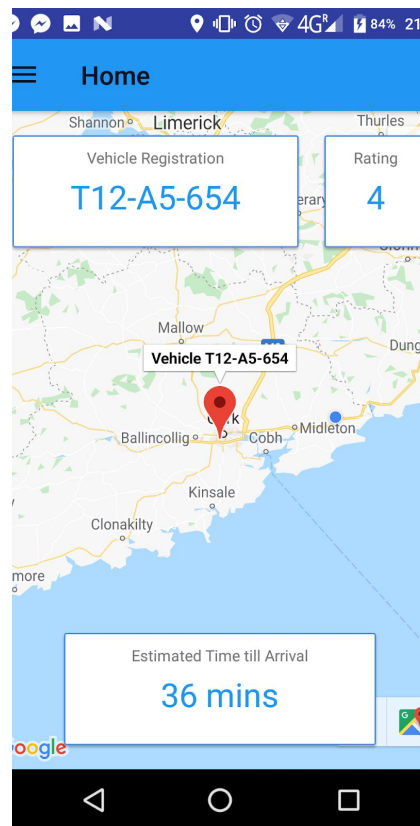
Picture 2:Destination picked, distance and cost calculated. Picture 3: Wait for confirmation



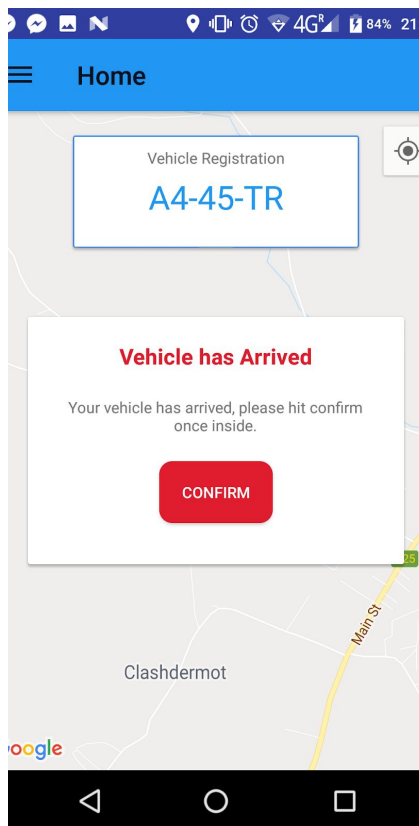
Picture 4: Confirmation received



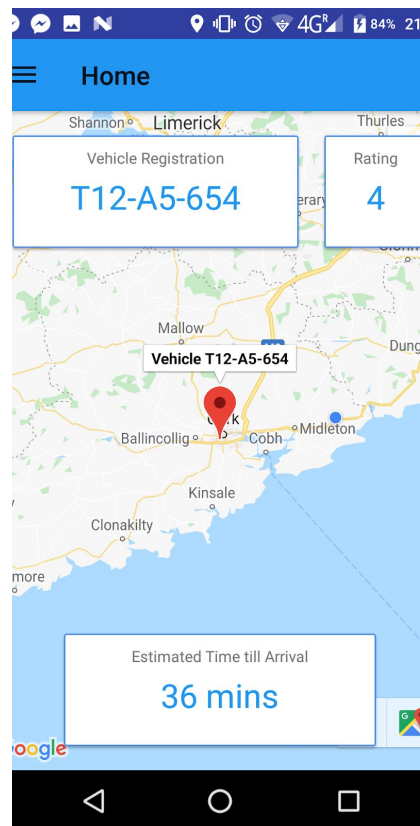
Picture 5: Vehicle incoming. ETA Shown.



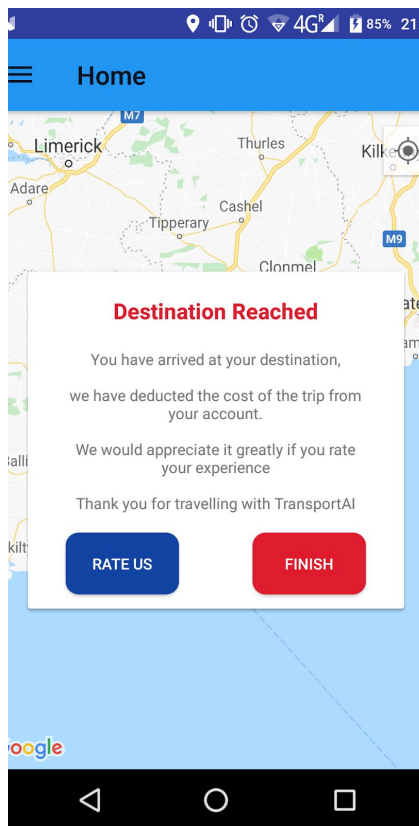
Picture 6: Vehicle arrival screen



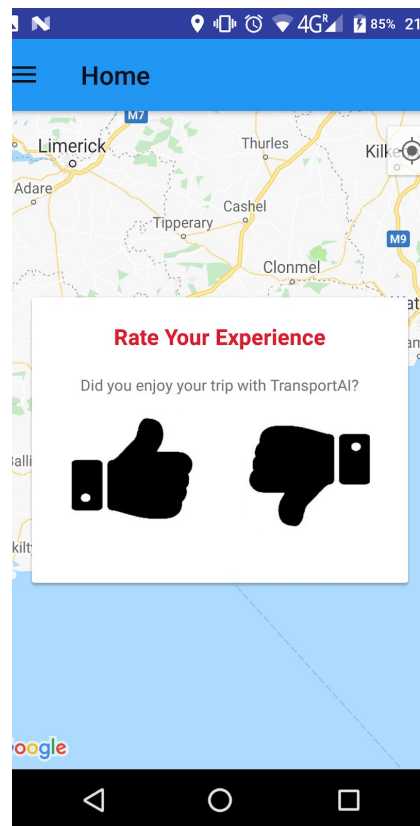
Picture 7: Route to location.



Picture 8: Destination Reached



Picture 9: Rate the Vehicle fragment.



## Funds

Picture 1: Funds Main Page

The screenshot shows the 'Home' screen of a mobile app. At the top is a blue header with a hamburger menu icon and the text 'Home'. Below this is a white box titled 'Account Details' containing the following information:

Name	Jamie Cotter
Current Balance	€-7.5
Balance After	€0

Below the account details is a card input section with a Visa logo, a card number '1234 1234 1234 1234', and an expiration date 'MM/YY'. Underneath are three input fields for 'Billing Name', 'Billing Address', and 'Top Up Amount'. At the bottom is a red 'CONFIRM' button.

Picture 2: Validation for incorrect card info

This screenshot is identical to Picture 1, but the card number is '5401 05/74 613'. A dark grey error message 'Invalid Card Data' is displayed over the 'Top Up Amount' input field. The 'CONFIRM' button remains red.

Picture 3: Inputting information on card

The screenshot shows the 'Home' screen for a different user, Daniel Moleman. The 'Account Details' box shows:

Name	Daniel Moleman
Current Balance	€70
Balance After	€0

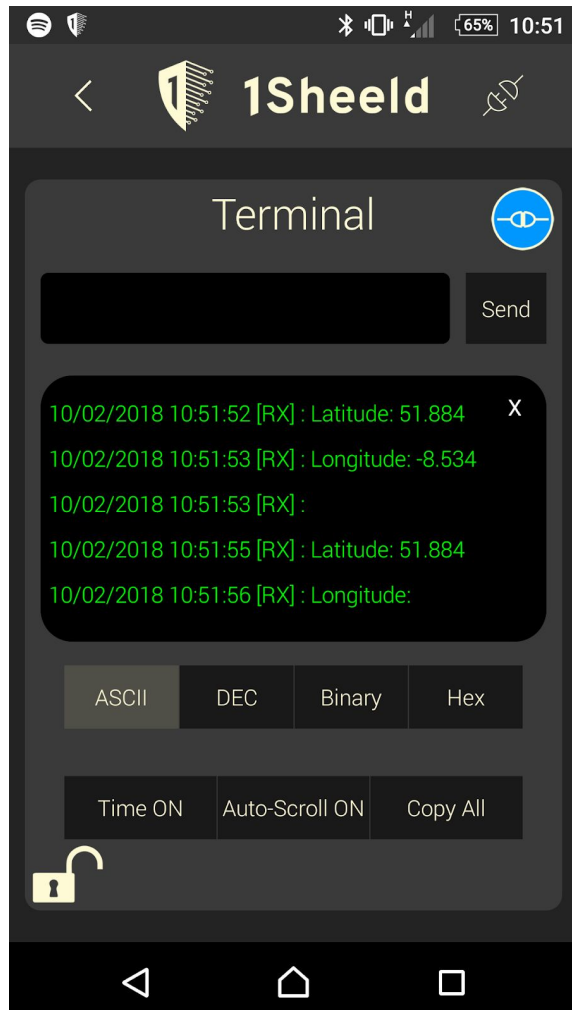
The card input section shows a Visa logo, card number '4242 04/24 424', and the name 'Daniel Moleman'. The 'Billing Address' field contains 'Moleman Manor, Ballyvolane'. The 'Top Up Amount' field contains '500'. The 'CONFIRM' button is red.

Picture 4: Funds successfully updated

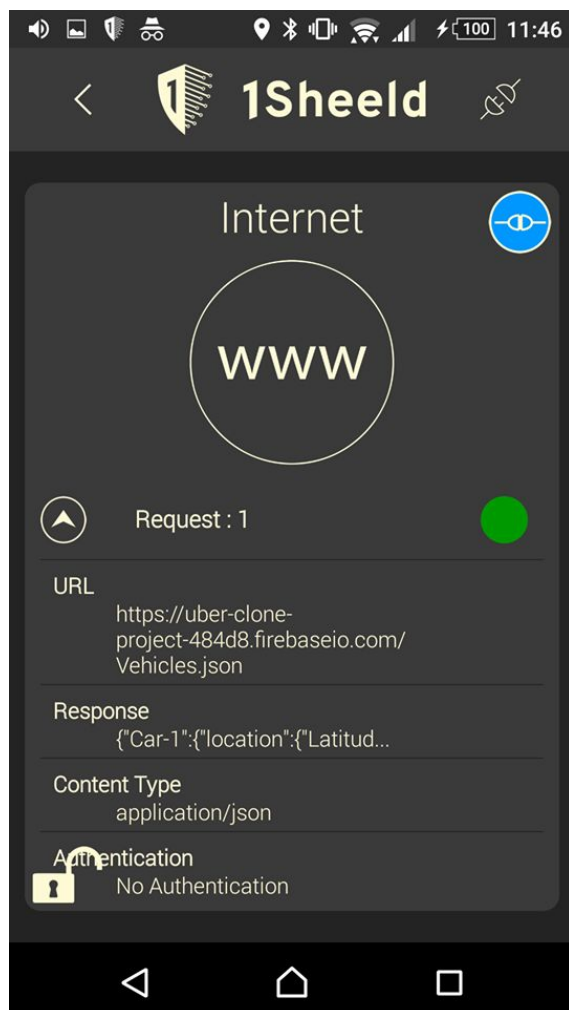
This screenshot is identical to Picture 3, but the input fields for 'Billing Name', 'Billing Address', and 'Top Up Amount' are now empty. The 'CONFIRM' button remains red.

# Hardware Testing

## Getting the GPS Coordinates



## http request successful

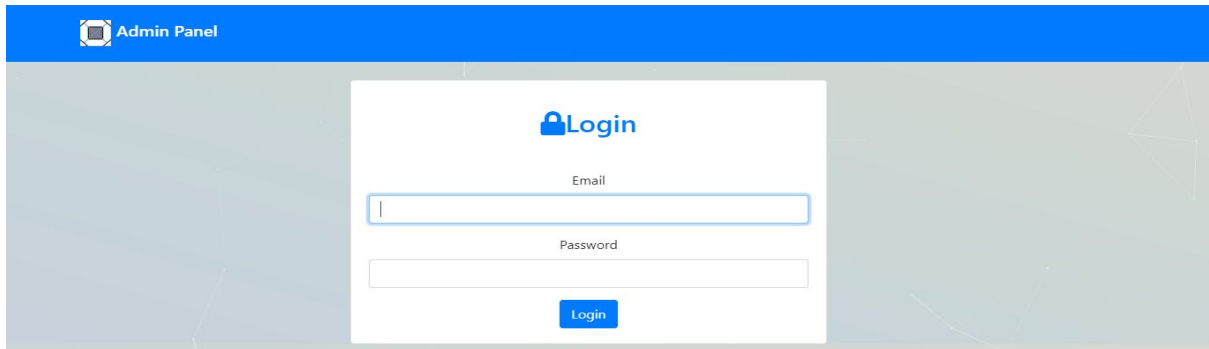


## Updates the database



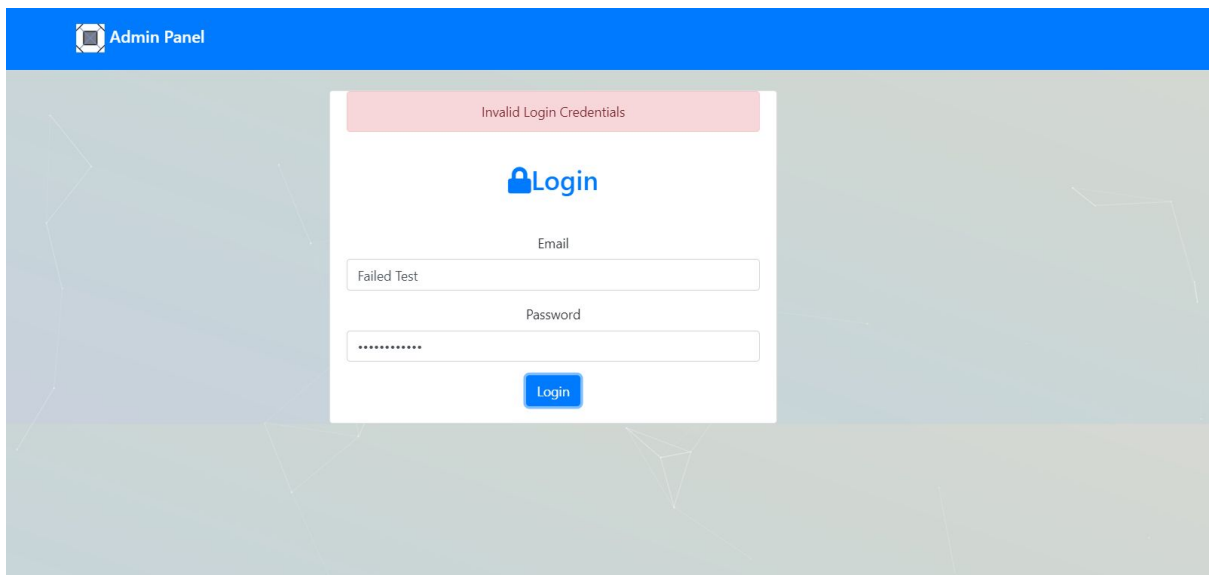
# Website Testing

## Admin Login



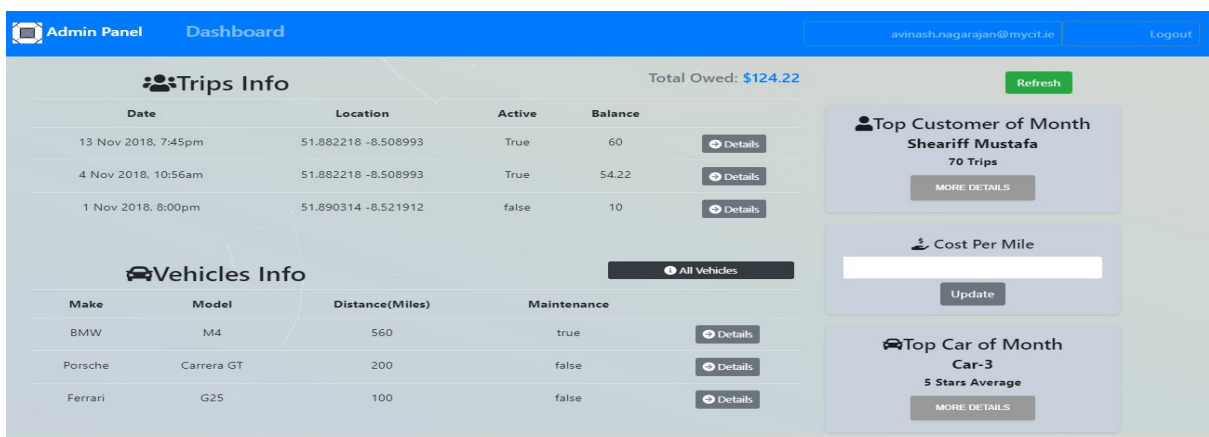
The screenshot shows the Admin Panel login page. It features a blue header with the 'Admin Panel' logo and title. The main content area is a light gray background with a white login form in the center. The form has a blue lock icon and the word 'Login' in blue. Below this, there are two input fields: 'Email' and 'Password'. A blue 'Login' button is at the bottom of the form.

## Admin Login, Bad Login Info



The screenshot shows the Admin Panel login page after an unsuccessful login attempt. A red error message 'Invalid Login Credentials' is displayed at the top of the form. The 'Email' input field now contains the text 'Failed Test'. The 'Password' input field is filled with dots. The 'Login' button remains visible.

## Home Panel



The screenshot shows the Admin Panel dashboard. It features a blue header with the 'Admin Panel' logo, 'Dashboard' title, and user information 'avinash.nagarajan@mycot.se' with a 'Logout' button. The main content area is a light gray background with a blue sidebar on the left. The dashboard includes a 'Trips Info' table, a 'Vehicles Info' table, and three summary cards on the right.

Date	Location	Active	Balance	
13 Nov 2018, 7:45pm	51.882218 -8.508993	True	60	<a href="#">Details</a>
4 Nov 2018, 10:56am	51.882218 -8.508993	True	54.22	<a href="#">Details</a>
1 Nov 2018, 8:00pm	51.890314 -8.521912	false	10	<a href="#">Details</a>

Make	Model	Distance(Miles)	Maintenance	
BMW	M4	560	true	<a href="#">Details</a>
Porsche	Carrera GT	200	false	<a href="#">Details</a>
Ferrari	G25	100	false	<a href="#">Details</a>

**Top Customer of Month**  
Sheariff Mustafa  
70 Trips  
[MORE DETAILS](#)

**Cost Per Mile**  
  
[Update](#)

**Top Car of Month**  
Car-3  
5 Stars Average  
[MORE DETAILS](#)



## Car Info

Admin Panel

Dashboard

avinash.nagarajan@mycity.ie

Logout

[Back to Dashboard](#)

Edit

Car-1

Rated 4.00 stars

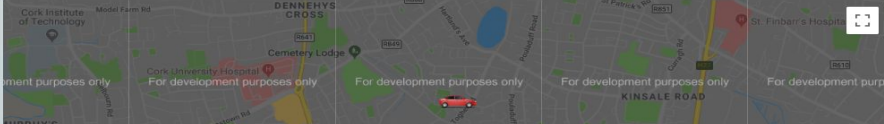
Distance Travelled: 560 miles

Needs Maintenance?: **TRUE**

Registration: T12-A5-654

Make: BMW

Model: M4



## Client Edit

Admin Panel

Dashboard

avinash.nagarajan@mycity.ie

[Back To Dashboard](#)

Edit Client

**Name:**

Sarah Dully

**Email:**

sarah.dully@gmail.com

**Phone:**

0847136458

**Balance:**

40

**Trips:**

12

Submit

## Customer Balance Update

Admin Panel

Dashboard

avinash.nagarajan@myc

[Back to Dashboard](#)

EditCancel

Sarah Dully

Customer Balance: \$40.00

Passenger ID: ABOoQMwa9Z49JBkfA35C

Balance: \$60.00

66

Update

Contact Email: sarah.dully@gmail.com

Contact Phone: 0847136458

Car Info: Car-2

Needs Maintenance?: FALSE

Registration: A4-45-TR

Make: Porsche

## Customer balance Updated

Admin Panel

Dashboard

avinash.nagarajan@myc

[Back to Dashboard](#)

EditCancel

Sarah Dully

Customer Balance: \$40.00

Passenger ID: ABOoQMwa9Z49JBkfA35C

Balance: \$66.00

66

Update

Contact Email: sarah.dully@gmail.com

Contact Phone: 0847136458

Car Info: Car-2

Needs Maintenance?: FALSE

Registration: A4-45-TR

Make: Porsche

## Trip Info

Admin Panel

Dashboard

avinash.nagarajan@mycit.ie

Logout

[Back to Dashboard](#)

EditCancel

Sarah Dully

Customer Balance: \$40.00

Passenger ID: ABOoQMwa9Z49JBkfA35C

Balance: \$60.00

Contact Email: sarah.dully@gmail.com

Contact Phone: 0847136458

Car Info: Car-2

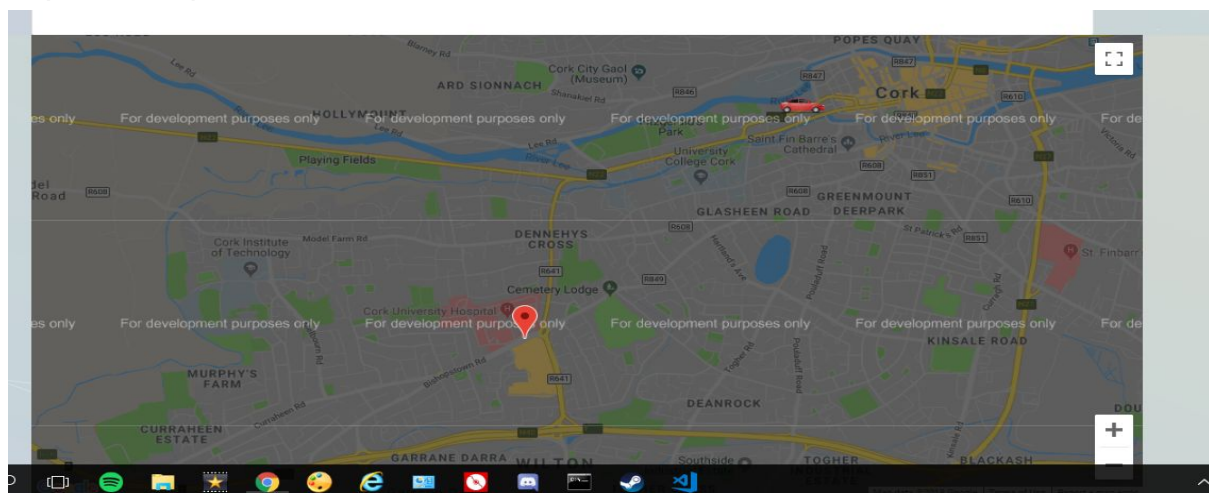
Needs Maintenance?: FALSE

Registration: A4-45-TR

Make: Porsche

Model: Carrera GT

## Trip Info Map



## Vehicle Distance Edit

Admin Panel

Dashboard

avinash.nagarajan@mycit.ie

[Back To Dashboard](#)

Edit Vehicle

Make:

BMW

Model:

M4

Registration:

T12-A5-654

Distance:

560

Submit

## Vehicle Distance Edited

**Admin Panel** Dashboard avinash.nagarajan@mycity.ie

[Back to Dashboard](#) Edit

**Car-1**  
Rated **4.00** stars

Distance Travelled: 600 miles Needs Maintenance?: **TRUE**

Registration: T12-A5-654
Make: BMW
Model: M4

Google  
This page can't load Google Maps correctly.  
Do you own this website? OK

## Vehicle Info

**Admin Panel** Dashboard avinash.nagarajan@mycity.ie

[Back to Dashboard](#) Edit


**Car-1**  
Rated **4.00** stars

Distance Travelled: 560 miles Needs Maintenance?: **TRUE**

Registration: T12-A5-654
Make: BMW
Model: M4

Map showing vehicle location (red car icon) and surrounding areas like DENNEHY'S CROSS, KINSALE ROAD, and St. Finbarr's Hospital.

## User Contact

 Admin Panel

Dashboard

avinash.nagarajan@mycit.ie

Logout

[Back to Dashboard](#)

Edit

Nioe Snekiw

Customer Balance: \$51.00

Passenger ID: w50YYlivCyerEsfKZK1e

Contact Email: eoin.wickens@mycit.ie

Contact Phone: 0841256349