Use Cases and Logical Architecture

- XID: X00167646
- Name: Taiwo Obadare
- Project Title: Torch

1 Use Cases:

Title (goal)	User Signs Up
Primary Actor	Unregistered User
Story	User that has not been registered will have the option to sign up to the application as a registered user. 1. Customers is at homepage 2. Customer Clicks Sign Up Link 3. Customer is brought to Sign Up page. 4. Customer enters First Name, Last Name, email address, password information. 5. Customer accepts Terms and Conditions. 6. Customer clicks sign up button 7. Customer is redirected to home page. 8. Customer is now successfully registered.

Title (goal)	Book a Scooter
Primary Actor	Verified User
Story	Verfied users will have the ability to rent a scooter out from a host at a set price per day calculated with insurance. 1. User is on electric scooter detail page. 2. User chooses pickup date & use chooses return date. 3. User Chooses pickup location or remains with host location. 4. User elects continue button. 5. User proceeds to book vehicle page. 6. User confirms price and insurance amount. 7. User selects pay now button. 8. Host receives booking request email. 9. Host navigated into email request link. 10. Host is brought to Host dashboard of requested vehicle. 11. Host clicks accept button. 12. User is verified to continue with booking via email. 13. User receives transaction id number in email. 14. User clicks into completion link. 15. User is brought to booking information page.

Title (goal)	User Applies to be a Host
Primary Actor	Verified User
Story	Verified users will have the ability to apply for a host status to be able to rent out there scooter and make money. 1. User will be brought to homepage 2. User click on Become a host link on navigation bar 3. User will be brought to host information page. 4. User will click accept link 5. User sees host confirmation page. 6. User has been elevated to host access. 7. Email will be sent detailing host information.

Title (goal)	List Scooter Out
Primary Actor	Host User
Story	Host users will be able to rent out their scooter in the marketplace to any verified user of the website. 1. Host will initially be on homepage. 2. Host clicks on admin dashboard link in nabber. 3. Host clicks on vehicles sub bar 4. Host clicks on list vehicle button. 5. Host is brought to add scooter page. 6. Host enters Scooter, Make, Model, Speed, Weight capacity, color, wheel type, upload vehicle of scooter. 7. Host enters scooter availability time slots. 8. Host enters vehicle description, cost per day and insurance provider. 9. Host clicks confirm. 10. Scooter has been successfully listed on the marketplace.

Title (goal)	User cancels order
Primary Actor	Verified User
Story	Verified users will be able to cancel their order a day before the scooter is needed. 1. User has vehicle listing booked. 2. User can cancel a day before official booking. 3. User is currently on home page. 4. User clicks Trips on nabber page. 5. User is brought to all trips section of page. 6. User clicks cancel trip. 7. User is prompted with a warning.

	8. User clicks confirm. 9. Host received notification of cancellation. 10. Host is on admin dashboard and confirms cancellation 11. Tips has been successfully cancelled.

Title (goal)	User Verification
Primary Actor	Registered User
Story	Register users will have the opportunity to verify themselves in order to rent out an electric scooter. 1. Registered User is at homepage 2. Registered User clicks register link. 3. Registered User is Brough to registration page 4. Registered User Uploads profile photo 5. Registered User enters mobile number 6. Registered User enters PPSN 7. Registered User enters in payment method 8. Registered User clicks review button 9. Torch admin verifies credentials 10. Registered User is elevated to Verified user. 11. Verified user receives email of verification.

Title (goal)	Filters scooter by speed, weight & make
Primary Actor	User
Story	1. User is at scooter results page. 2. User clicks textbox by a certain speed 3. User clicks textbox, by given weight 4. User selects dropdown of given make 5. User clicks filter button. 6. Results lists is retrieved 7. User sees filtered scooters

Title (goal)	Search available Scooter by Locality & Date
Primary Actor	User
Story	Users of the site will be able to 1. User is at home page 2. User clicks into homepage search box 3. User selects scooter location 4. User selects available from date 5. User selects available to date 6. User clicks search button 7. User is brought to results list

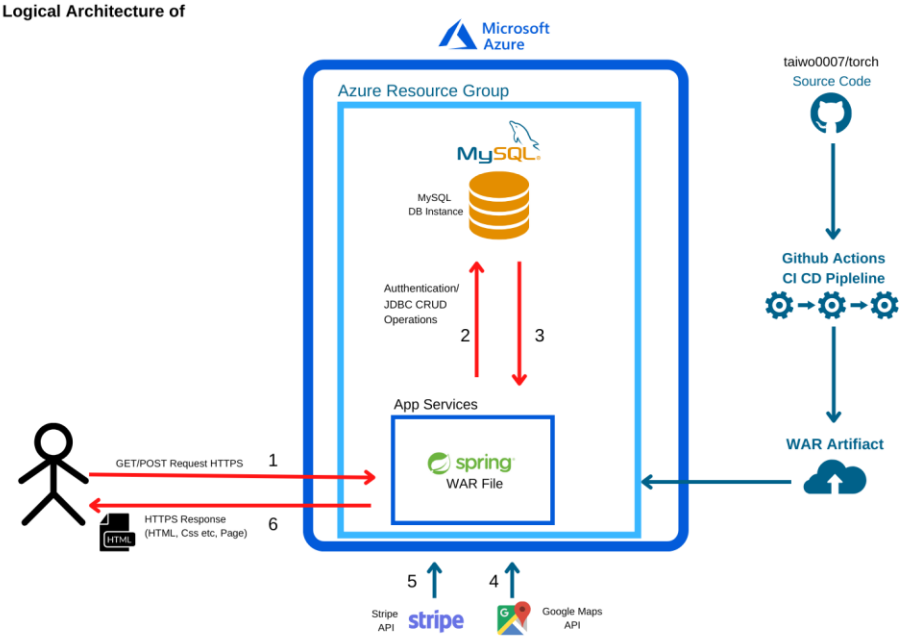
Title (goal)	Browse Scooter Listing.
Primary Actor	User

Story	1. User is at scooter results page. 2. This page comprises of pictures of scooter and cost. 3. User clicks into desired scooter. 4. Scooter detail page is loaded. 5. Scooter detail page shows reviews of scooter. Insurance details to include insurer name. Picture of scooter to see and cost, location information.

Title (goal)	Review a scooter
Primary Actor	User
Story	1. User is brought to scooter detail page. 2. User clicks on review textbox. 3. User writes review in the checkbox. 4. User selects the star rating of the trip. 5. User clicks submit review. 6. Page is reloaded. 7. Review submitted banner is displayed.

2 Logical Architecture

Figure 1: Logical Architecture of Torch



This application is hosted on Microsoft Azure. Microsoft Azure allows us to create a resource group that will host hour application on the cloud. The resource group will also contain our database.
The system will updated on an ongoing basis using a CI/CD pipeline. When changes have been made by the developers, the developer will commit the code to a git service. This code will be pushed to GitHub and this will trigger a build using the Git Actions CI CD capabilities. Once the build is successful it should release an artifcat. This artifact will be sent to Azure and azure will update the application with any necessary changes.

User sends a request HTTP request to our azure server which hosts the spring boot web application (1) The web application processes the request and ties it to one of our controller methods that routes it to a service. The service will process the request and go to the repository layer (2) where it will retrieve the relevant data before it sends it out with the template The request is for a data in the database and the database is connected in the application using relevant credentials such as host and password. This database is a MySQL database hosted on azure. The database understands the request and returns the relevant data, this data could be user credentials or scooter information. (3) The application will also bring back data from the database to talk to the google maps Api (4) Stripe will be used to process payments that may have been initiated (5) The Controller layer that initiated the request to the service layer and then the service to the repository later returns the data to the controller layer. The controller layer maps the relevant data to the HashMap object and combines it with the relevant template to show the user. The application then returns the requested resources, in this case website resources and displays it to the user in a HTTP Response.(6)