Project Title: CT Fastrack Website

Deliverable Name:	System Requirements
Team Name:	TeamSpark
Membership:	Lavanya Mattaparthi Kruthika UmmadahalliSomaraju Harshita Singh Khanom Meshkat Hassan Beiki
Submission Date:	10/06/16

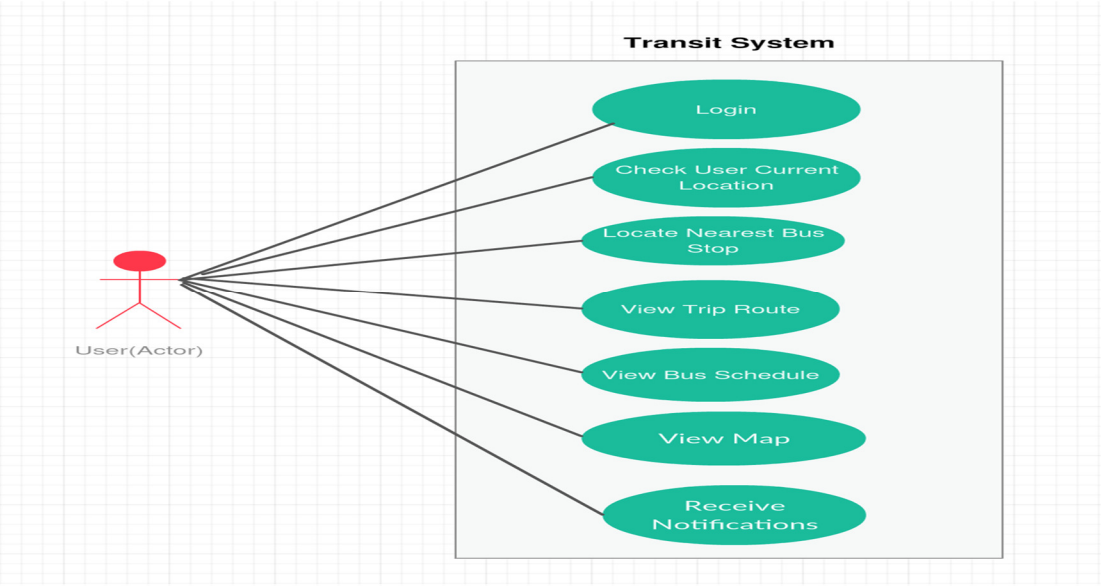
Functional Requirements

With the transit system we are helping Users to plan their ride, locate the bus stop closest to their location, check bus schedules, transit directions, get notifications and even store their favorites for later use. It delivers results in real-time: Up-to-the minute information on predicted departure/arrival times, keeps commuters up to date and on time.

The system takes information about the source and destination, selection of the bus, and displays the bus numbers along all the routes heading to the destination, generates maps as soon as the bus number is selected and most importantly tracks the location of the bus with the help of real time data feeds and sends the location and the minimum time required for the to arrive the bus stop.

All these tasks would be accomplished by the help of External Sources like Google API and JSON. JSON data feeds request for the raw data from the Service Provider and process them into the form of real data. Which will help the Users in showing the Routes, Map through GPS.

Use Case Diagram



Use cases

- ***Login:*** User needs to login to the System, for availing the transit facility, User can save the frequently visiting places in the form of favorites too.
- ***Check user current Location***: This feature helps identifying user current location on the Map
- ***Locate nearest Bus Stop:*** This feature helps to find the closest Bus stop to User's current location or to any given location. User can search by either a zip code or a full address.
- ***View trip Route***: This feature helps users showing all of the available public transit lines near their location and informs when the next vehicles are expected to arrive.
- ***View Bus Schedule***: It provides the bus timetable which makes the trip plan easy and convenient for the Users
- ***View map***: It provide real-time bus positions and arrival estimates.
- ***Receive Notifications***: Notifies the User when to leave and also let them know if they need to walk faster to make their connection.

User Stories

1. Login:

User role- Passengers who use this application.

Goal- Users who visit same place frequently, can create favorites for those locations.

Reason- User can click on favorites instead of inputting the same location every time, which saves time.

Pre-condition- User should create username and password.

Post condition- After logged in to the System, User should get 'favorites' options.

Breakdown- No Further Breakdown.

2. Check User's current Location:

User role- Passengers who use this application.

Goal- When a passenger enters their current location, their location will be identified on Map.

Reason- Passengers can identify their current location on the Map.

Pre-condition- User should have connected to the internet.

Post condition- User's device should be compatible to display Google Maps.

Breakdown-No further breakdown.

3. Locate Nearest Bus Stop

User Role- Passengers who use this application.

Goal- User should be able to identify nearby bus stops based on their current location and any given location.

Reason- With this feature, users will come to know how far is the nearest bus stop and accordingly they can plan their start time and commute to the bus stop.

Pre-condition- Users current location should be identified or desired valid location should be given as input.

Post condition-No Postcondition.

Breakdown- No further breakdown.

4. View trip Route:

User role- Passengers who use this application.

Goal- User should be able to identify their trip details visually on Map.

Reason- With this feature, User will be able to visualize the route and identify all the bus stops where the bus will stop on that route. Also, this feature will provide additional information like travel distance and travel time so that passenger can plan further commute at the destination location.

Pre-condition- User should provide valid source and destination.

Post condition- System should display route and display all bus stops on Maps.

Breakdown- No further breakdown.

5. View Bus Schedule:

User role- Users who use this application.

Goal- User's should be able to view any bus schedule.

Reason- Users would like to know the possible schedule for a given route so that they can plan when to start. With this feature, users will get all current and future schedule information of a bus.

Pre-condition- User should know which bus schedule they are going to search.

- Valid bus route (i.e. bus route number) should be provided.

Post condition- Only current and future bus schedules should be displayed.

Breakdown- No further breakdown.

6. View Map

User- Passengers who are using this application.

Goal-View routes on map, real-time bus locations, stop locations

Reason- See exactly where the bus is on the map in real-time and helps to plan the trip accordingly.

Pre-condition-No pre condition.

Post condition- Post condition is map has been changed.

Breakdown- It would be broken down into two, zoom in/zoom out.

7. Receive Notifications

User- Passengers who are using this application.

Goal- With this feature, passengers will be able to set a Reminder.

Reason- Many Passengers will keep an alarm to start early from home to reach the bus stop on time. With this feature, the passenger will get a notification before specified time by the user. This feature helps in eliminating users to manually set alarms on their devices.

Pre-condition- User should know the Bus and its departure time to set a notification.

Post condition- Notification should reach the user with a message.

Breakdown- No Further Breakdown.

Non-Functional Requirements

- Application should be user friendly.
- Application should be available.
- Application should be fast.
- Application should respond to the User's queries.
- Application should obey the privacy policy for protecting the data from the external sources.
- Application should have relevant documentation.
- Application must be secure.

Glossary

- Google API- Google Application Program Interface
- JSON-JavaScript Object Notation
- GPS-Global Positioning System