

# **System Requirements: CTFastTrak Live Map and Routes**

Software Engineering CTFastTrak API Project  
Bryan Davis, Robert Rotaru, Matthew Shafran, Brian Tardiff

## **Project Overview**

The system will have a Map API (such as Google Maps), an interface for the JSON data from CTFAstrak, and an interface for the GTFS data from CTFAstrak. The system needs to be accessible by a user who connects to it via an internet browser. The user is able to interact with a map interface in order to display all current CTFAstrak data, to find the nearest bus stop to a point on the map, set a desired destination to navigate to, and pick a route from a number of possible routes. Additionally, the user is able to get individual bus information and bus stop information from the system. If there are alerts, updates, or notifications, the user is able to receive them live from the system. The GTFS interface will listen and accept notification data from CTFAstrak and make this available to the system. The GTFS interface will also accept bus information data and bus stop information data and make this available to the system. The JSON interface will be able to request bus stop information and bus information from CTFAstrak and make it available to the system.

## **User Stories**

User Story #1: User is displayed current information

Pre-condition: System is online and reachable. User is connected to the system.

Post-condition: User is able to view current information.

User role: Traveler

Goal: To get up-to-date CTFAstrak information from the system.

Reason: Traveler is better able to plan a route.

User Story #2: User gets the nearest bus stop location.

Pre-condition: System is accessible; Traveler is logged into the system; Locations are available

Post-condition: Traveler is able to retrieve nearest stop location

User role: Traveler

Goal: Traveler can access the application's get nearest location feature and the app will determine the stop location nearest to the traveler

Reason: Traveler should be able to access the nearest bus stop location so that they can plan their route accordingly

User Story #3: User is able to set a destination.

Pre-condition: System is accessible; Traveler is logged into the system; Locations are pre-loaded into the system

Post-condition: Traveler is able to set a destination

User role: Traveler

Goal: Traveler can access the application's set destination feature and specify a destination that they would like to travel to

Reason: Traveler should be able to set a destination that they would like to travel to in order to accommodate travel plans

User Story #4: User chooses a route from among one or more.

Pre-condition: System is accessible; Traveler is logged into the system; Routes have

been calculated based on traveler's input

Post-condition: Traveler is able to pick a route

User role: Traveler

Goal: Traveler can access the application's pick route feature and select from a list of calculated routes that the application has generated based on the user's input

Reason: Traveler should be able to pick a preferred route to facilitate travel needs

User Story #5: User is displayed bus information

Pre-condition: System is online and reachable. User is connected to the system.

Post-condition: User is able to view most current bus information

User role: Traveler

Goal: To get up-to-date location of bus stops

Reason: Traveler is better able to plan a route

User Story #6: Get bus stop information

Pre-condition: System is online and reachable. User is connected to the system.

Post-condition: The user is able to receive information about bus stop locations and projected arrival times.

User role: Traveler

Goal: The user will be able to look at nearby bus stop locations and figure out which are closest. The next upcoming bus arrival times will also be listed for each location.

Reason: The user will be able to figure out which stop to go to.

User Story #7: Receive GTFS Notifications

Pre-condition: System is online and reachable. User is connected to the system. The user is either allowing automatic notifications or requests an update of GTFS data.

Post-condition: The user will receive all up to date information from the GTFS data, including any changes to route information.

User role: Traveler

Goal: The user will be able to receive any updated information about the bus arrival times, ensuring that the data shown to the user is as accurate as possible. The notifications will update the user to any important changes (i.e. a delayed bus).

Reason: The user will be alerted of any changes in bus routes and be able to plan a better route if the alerts affect him or her.

User Story #8: JSON getting bus stop information from CTFastTrack

Pre-condition: CTFastTrack provides bus stop information in JSON format

Post-condition: Bus stop data gets sent to user

User role: JSON Data

Goal: To get up-to-date location of bus stops

Reason: Responding to User's request for bus stop location

User Story #9: JSON getting bus information from CTFastTrack

Pre-condition: CTFastTrack provides bus information in JSON format

Post-condition: Bus data gets sent to user

User role: JSON Data

Goal: To get up-to-date location of buses

Reason: Responding to User's request for bus location

User Story #10: GTFS interface accepts live event/delay data.

Pre-condition: GTFS API is available. Interface is able to access GTFS data.

Post-condition: System listens and receives live event and delay information. This information is displayed in the user interface.

User role: The GTFS interface.

Goal: Live event data is relayed from the CTFAstrak API to the user interface.

Reason: Traveler is able to receive notifications about route conditions and events.

User Story #11: GTFS interface accepts live bus information.

Pre-condition: GTFS API is available. Interface is able to access GTFS data. There is bus information available.

Post-condition: System listens and receives live bus status information. This information is displayed in the user interface.

User role: The GTFS interface

Goal: Live bus data is accepted from the CTFAstrak API and displayed to the user interface.

Reason: Traveler is able to view buses live on the map.

User Story #12: GTFS interface accepts live information about stops.

Pre-condition: GTFS API is available. Interface is able to access GTFS data. There is bus stop information available.

Post-condition: System listens and receives live bus status information. This information is displayed in the user interface.

User role: The GTFS interface.

Goal: Live bus stop information is accepted from the CTFAstrak API and displayed to the user interface.

Reason: Traveler is able to view stops and live stop conditions on the map.

## **Nonfunctional Requirements**

1. The system will be deployed as a web application, for both mobile and desktop browsers.
2. The web application will be available to anyone with an internet connection, focused mainly on commuters in CT.
3. Once deployed, the web application should have near 100% uptime, taken down only for maintenance.
4. The web application will be able to scale up if given more data (through JSON or GTFS).
5. The web application will be free to all users.
6. The user can specify their current location, but this will be kept completely private.
7. The web application will be as reliable as the data given to the application (from CTFAstrak).
8. The user can turn off automatic updates in order to make the application more

resource efficient (especially for mobile, to save data).

## **Glossary**

1. BRT (Bus Rapid Transit) System - A system which provides quality and time-efficient transportation throughout urban, suburban, and rural communities
2. Bus Terminal - A location in which a traveler can get on or off a CT Fastrak bus
3. CT Fastrak - A transportation system featuring high quality and cost-effective bus transit services featuring many of the Bus Rapid Transit (BRT) best design and operation practices
4. Google Maps - A service that allows a map to be displayed on a web application
5. Google Maps API - An application programming interface that enables developers to customize maps and the information displayed on the map
6. GTFS - GTFS (General Transit Feed Specification) is a feed specification that allows public transportation agencies to provide realtime updates about their fleet to application developers
7. Service Maps - Maps generated by CT Fastrak which display local, express, and connector routes and destinations, park and ride locations, and multi-use recreational trail access
8. Servicing area - A city or town that provides access to CT Fastrak transportation
9. System Map - A map of the entire regional CT Fastrak system, and is posted at various CT Fastrak stations
10. Traveler - A user who intends to use the CT Fastrak application to assist in obtaining information related to CT Fastrak transit options