

Bus Tracker Website and Application

By: Group X

Matthewit Dechatech, Jonathan Kleehammer, Ashley Torres, Tristan White

Introduction

Purpose

The goal of this project is to create a web application accessible on mobile and desktop PC's for local Denton riders of DCTA (Denton County Transport Authority).

Scope

The "Denton Bus Tracker" is a web application that helps riders of DCTA quickly find bus routes to their destinations, view routes and stops with estimated arrival times, and see buses moving in real time.

Definitions, acronyms, abbreviations

DCTA: Denton County Transport Authority

Reference Documents

[Geodjango documentation](#)

[GDAL](#)

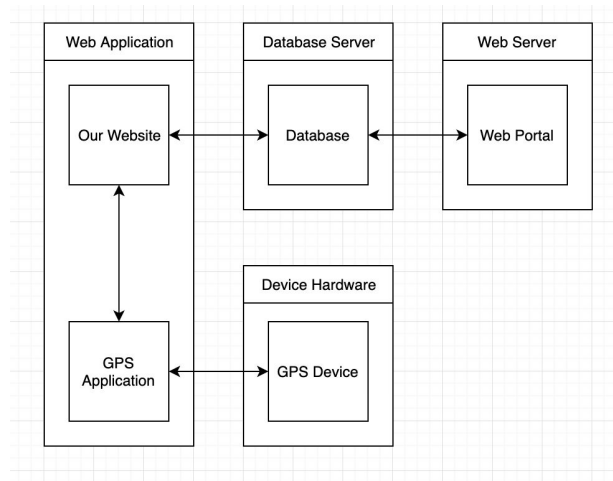
Overview

The following SRS document will outline the system's intended usage, who will be using it, and what features it will include upon completion. Section 1 will cover the overall description of the project, such as what the project is, it's functions, who will use the product, and constraints. Section 2 will contain the list of the software's specific requirements. This will include the various levels of interfaces the software is accessed with.

Section 1: Overall Description

1.1: Product Perspective

The goal of this software is to be a replacement to the current UNT Bustracker webapp, but will have a new self-contained product with no affiliation with the past bustracker. DCTA provides the bus route information, as well as real-time GPS information of each transport vehicle. We will have the website stored online which keeps a database to store information about users, routes, and buses. On the individual's device we'll need to retrieve their GPS information to enable route planning.



1.2: Product Functions

- Users will be able to click on the route they wish to ride on and view:
 - Closest bus
 - All stops for the bus routes
 - Bus schedule
- Routes can be sorted in multiple ways:
 - Favorite Routes
 - Closest to user
 - Alphabetically.
- Users will be able to give feedback to the administrators.

- Administrators will be able to use Django Admin (a site made by the django web framework that allows for changes in the website) to add or remove buses or routes
- Administrators will be able to view data that the bus drivers and rider feedback.
- Buses statuses can be identified as “in service” or “under maintenance”.
- Routes can be modified to reflect construction and road closures by the administrators.

1.3: User Characteristics

Bus riders will be the main users for this web application. The system is intended to be used by the general population of Denton, which includes all age ranges, education levels, and technical expertise. Because of this, the system will be very simple and intuitive.

DCTA will use the administration site to assist in running the transit services. It will have the features needed all on one screen for ease of adoption by DCTA.

1.4: Constraints

Because the real time tracking of the vehicles is provided by DCTA, we only have the information they provide. This means that we can only update the actual locations as well as the locations of vehicles as frequently as they provide.

Due to the inexperience of the developers with the software development process and the tools used to make the bustracker, the development process will be slowed down.

1.5: Assumptions and Dependencies

We are assuming that the user is using a device with at least 1gb of RAM, a 1.4 Ghz processor, and 1 Mbps download speed for loading the page. These are pretty low settings that even phones from several generations ago can surpass.

Section 2: Specific Requirements

2.1: External Interface

2.1a: User Interface

- The main screen will feature a map which will take up most of the screen with two icons on it.
 - One icon will open a sliding menu that will show the list of routes.
 - Second icon will send the user to either the login/sign up page or the user page if they're already signed in.
- The sliding routes menu will show tiles of each route.
 - Each tile features the title of the route and is assigned a unique color.
 - Clicking on the tile will expand downwards showing all the route stops in order of the estimated times of arrival.
- The user screen will default to asking the user to sign in if they're not already. It will feature input boxes for a username, a password, and a button for logging in. Below the login box there will be a link to sign up if the user does not have an account.
- Once logged in, users will be able to manage their account information, favorite routes, and feedback.

2.1b: Hardware Interface

- If the user will use the web browser, they will control it with a keyboard and mouse.
- If the user will use the mobile app version, they will control it with the touch screen.
- The sites interface will be able to adjust to the size of the screen.

2.1c: Software Interface

- Admin site that is made from Django will be used to modify the available routes and buses.
- User's data from their account will be stored anonymously in database for use interacting with the apps.
- A PostgreSQL database will be used to contain the data.

- A Google Maps-like tool will be used to give us a map of Denton, as well as a way to map the routes and stops of the buses.

2.1d: Communication Interface

- The product will be made with Django web framework, so it will require a network server.
 - Server will be uploaded to a TBD cloud service.
- User will give an email for their account allowing for features such as favorite routes.

2.2: Functional Requirements

- The user should be able to view a map of the Denton County transit service.
 - It will display a map that will contain the area that all the routes cover.
 - The map will have a column on the left side of the screen that will have a list of color coordinated bus routes.
- User will be able to select the route the wish to ride on
 - The route selected will be highlighted to ensure that the correct one is selected.
 - They will be able to see the time the closest bus arrives at on the selected route.
 - User will be able to see all the stops on the route and the estimated time that a bus or their current bus will arrive at it.

2.3: Performance Requirements

- App must be able to run smoothly.
 - The site will not crash when performing any actions on it, such as logging in, selecting a favorite route, etc.
 - Apps will be as bug free as possible by making sure that the information is right at all times.
 - The app will be able to run with the above conditions on both the desktop site and mobile site.

2.4: Design Constraints

- For mobile, the UI will potentially not be always mobile friendly as the site is designed for desktop web.
- Programmers are still learning the technologies

2.5: Other Requirements

- Security requirements