**Indiana University Southeast - CSCI Department**

**CS Capstone II - Programmer’s Manual**

**Spring 2017**

**TruckPup - A Louisville Food Truck Finder**

**Team Intereus:**

**Todd Ayers**

**Jason Lytle**

**Clayton McGuire**

**Branden Wagner**

**Vision:**

The vision for this project was to create a software suite for the administration and location of food trucks in the Louisville, KY area and to do so from the perspective of truck owners. Many similar apps are available, but seem to only exist to fill the needs of the user. Based on the needs of the sponsor and on the idea that to gain the truck owners support of the app, it was decided to build the system with features that they would prioritize over features the user would prioritize. In most ways these needs paralleled each other, but also required the addition of features that might not be needed if the software was only built for users.

**Introduction:**

In achieving the vision set for the project, a set of software systems were created. A mobile application for users and truck owners and a website for truck owners and administrators. A name was decided on, TruckPup, and some early visual elements were gathered. A number of tools, languages and resources were used to bring the project together, which will be discussed in more detail below. Ultimately, the scope of the elements intended for the first run of the project became too great and some initially planned upon features had to be left out. Work that could progress the project further will also be shown and ideas for future features listed.

**Components:**

The system is composed of two major software implementations. The first is a mobile application designed to run on the Android and iOS operating systems that will give profile functions to truck owners and information to users. The second is a website that gives functional control of a database to the systems administrator and profile control to truck owners.

**Mobile Application:**

As stated, the mobile application will be available to Android OS users and iOS users. The app is intended for use by truck owners and users (who in the view of this project are people searching for location information, event and profile information about food trucks in the Louisville, KY area). This splits the use of the app for two distinct purposes:

* truck owners will use the app to sign in using a username/password combination and then broadcast the location of their truck using the phone’s built in GPS system. At the time of broadcasting, truck owners are also able to decide how long they want the truck to list as active (in this case meaning available for business) by choosing a time value from a drop down menu.
* users will use the app to scan the area nearby for active trucks, click on the map truck icons to get abbreviated information about the truck, check the events list for where trucks will be in the future, and check the truck list for extended truck profile information (including menu and “About” page)

**Website Application**

The website has a front end for user interaction and a database backend for holding information about the truck’s owners. Truck owners and administrators will have functional access to this database through the front end, which like the mobile app separates the website for two purposes:

* administrators will sign on to the website with an authorized username/password combination. They will then have access to an interface that will allow them to invite new truck owners using their email address, deactivate current truck owner’s accounts, and permanently delete current truck owner’s accounts.
* truck owners will sign on to the website with an authorized username/password combination. They will have access to their profile where they can edit the truck name, add a profile picture, add “About” information, and add social media account information (for this release Facebook and Twitter only). Truck owners can also create, edit and delete menu items and their prices in one of four categories (Combo, Entree, Side, Drink).

**Tools:**

A variety of tools, languages and resources were used over the course of the project. Below is a fairly comprehensive list used in the programming and presenting done throughout.

**Drawing tool for ER diagrams, network diagrams, for files, data entry screens and other user interfaces, and algorithm designs for processes etc. :**

Balsamiq for mockups, MS SQL Studio for database diagram

**Database for local data:**

HTML LocalStorage

**Database for system-wide data:**

MS SQL Express

**User diagrams:**

Balsamiq

**Word processing for reports tool:**

Google Docs

**Computation tools: e.g., web page development, jbuilder, sourceforge:**

*Web development:* Visual Studio  
*Packaging for iOS:* Xcode  
*General text editors:* Visual Studio Code, Notepad++, Atom

**Language of implementation (for different segments of the project):**

JavaScript (Angular, TypeScript), HTML (Material Design for Bootstrap), SCSS, SQL, ASP.NET MVC (C#)

**Cross-platform framework (for mobile application development):**

Ionic (NodeJS for CLI, Cordova for native components, Gulp for build automation)

**Project management (Gantt charts, timing charts):**

Google Docs

**Webblog for team communications:**

GitHub’s “Issues” section

**Any other tool being used:**

GitHub for source control, WinSCP to update team’s ADA page, Firebase for authentication, Pusher for real-time updates of mobile app

**Project Repository:**

* **Software**
  + The project can be cloned from the repository [here](https://github.com/ephemerant/445-Final).
* **Documentation**
  + All documentation pertaining to the project (including testing materials, presentations and project structure can be found [here](https://github.com/ephemerant/445-Final/tree/master/docs) and [here](http://ada.ius.edu/~cmcguire/).

**Installation for New Install**

New installations of the mobile apps can currently be created by cloning the repository from the above link and following the steps on the [Ionic Framework](http://ionicframework.com/docs/v1/guide/testing.html) page. As can be seen by reading the documentation, installing to an Android device is very easy, but installing to an iOS device requires ownership of an Apple Developer Account. Steps for testing the apps using a browser can also be found at the above link.

New installations of the website would require server space and adjustment to the code to redirect calls to the current server space and current profiles for Firebase and Pusher.

**Installation for New Platform**

The mobile app has no persistent data. It can be deleted and downloaded without loss of functionality. The website also has no persistent data. The database should be backed up before attempting to install it onto a new server.

**Further Development**

If given time and resources, there are a number of different additions that could improve the overall experience of the system or just increase the functionality. Over the limited time that was available, only some of the intended features were able to be implemented and tested and in some cases, new features became desirable during development or through conversations with the sponsor.

**Mobile Application**

For the mobile application, there were a number of features that made it to the user interface, but were not completely implemented in code. There were also quality of life improvements that could be made.

* Social Networking (User)
  + This was an item that was requested by the sponsor. She wanted users to have the ability to post directly from the app to social media outlets of their choice about their experience with a truck.
* Social Networking (Truck Owner)
  + This was an item that was requested by the sponsor. Similar to the user version, she wanted truck owners to be able to post directly to various social media accounts. The difference was another feature that there was trouble with, scheduling, which will be covered shortly in the website section.
* Map Improvements
  + This was an item discovered while testing. There was an issue with truck icons on the map screen being too close together if the map was zoomed out. Although it can be fixed currently by zooming in, there are better methods that could be implemented.
* Hotspot Button (User)
  + This was an item brainstormed with the sponsor. There was an idea to create a button to allow users to try to request that trucks come to a spot where none currently come. If enough users in an area made this request, this could allow truck owners to enter a new market with less risk.
* User Accounts
  + This was an item created by the development team. The idea was to have users have their own accounts so that they could save information to their device, such as a “Liked” truck list for filtering purposes.
* User Loyalty Reward Program
  + This was an item created by the development team. Users could gain rewards by eating with app affiliated trucks, like discounts or coupons.

**Website Application**

* Social Media Post Scheduling
  + This was an item requested by the sponsor. She wanted to be able to create a schedule in advance for when trucks would be active and automatically post to social media at those times and make the truck active in the system.

**Both Applications**

* Design Work
  + Although some design materials were collected, more effort could be put into making the mobile application visually appealing.

Although there are a number of items on this list, it is less indicative that not everything planned was accomplished in the time given and more indicative of the amount of growing that the app could do.