MCTS3000

Connor Henke, Ryan Gahart,
Tony DiPerna

Target Audience & Value

- Anyone looking for a better bus riding experience
- People who use Milwaukee buses regularly
- Visiting tourists

- Easy to plan out travel across the city to make the most out of people’s time

Why use it?

- Save favorite bus stops for quick, easy navigation
- See all active routes in MCTS
- Able to see realtime bus locations and stop times
- You get the same functionality as MCTS3000 for Android

Competition

- Bus MKE
- Transit Tracker (Milwaukee)
- MCTS Tracker
- Google Maps

Carthage Packages Used

Alamofire

- Used to make asynchronous HTTP requests to MCTS API

ObjectMapper

- Used to easily convert JSON responses to defined Swift objects

Realm

- Mobile database that uses its own persistence engine

Demo

Contributions

Tony

- Project setup, data persistence, favorites and routes page

Ryan

- UI fixes, loading and working with the map

Connor

- Predictions page, loading vehicles and stops

What did we learn?

- Alamofire and ObjectMapper make working with APIs very easy
- MapKit lets you deeply customize your map
- Realm is simple and straightforward to start using
- Auto layouts makes building responsive views easy

Difficulties

- Merge conflicts when multiple people are working on the same Storyboard
- Unhelpful errors from Realm
- Issues with dependencies and Carthage that held us back initially
- Other issues with the shared development environments