Bluetooth Bicycle Tracker

Trillium Health
Problem Description:

- Numerous race management mobile applications exist, BUT few with bluetooth interaction and intermittent network capability.

- Design a system built for collecting and sharing data that could operate both online and off, while showing the user the most recent information available.
Background

Considerations:

- Mobile Application must be functional with intermittent connectivity.
- Application must estimate ETAs as well as positional information to the best of it’s ability.
- Mobile Application must automatically check-in/check-out racers via Bluetooth Beacons.
- Along with the iOS application we would also need an API for the mobile application to hit.
Background

Designed Solution, Architecture:

MySQL Database  Sails.js Server  iOS App  Bluetooth Beacon
Background

Tradeoffs:

- Sails has no native front-end framework
- Sails is not good at supporting two different authentication types
- iOS has security limitations in establishing communications
- MySQL has a rigid schema
Welcome!

About Us
The RaceChipper™ app is designed for the enjoyment of the most adventurous drivers and offers a set of features that can help you analyze performance and receives your vehicle data. This application makes it easier for you.

Safety First
By collecting data with the app, you can monitor your vehicle's data continuously. This information is essential for all car owners. There are no integrated data features. You can access your vehicle data for all time, easily, and conveniently.

How it Works
[Diagram showing the process]
Testing - General

- Field testing
  - Dry run with actual racers to test automatic checkin/checkouts

- Unit testing
  - Framework in place, runs on build
  - Lagging due to time constraints to finish features
Testing - Usability

- SUS (System Usability Scale)
  - Alternative to SUMI
  - 10 questions, Likert scale; users follow script
  - RESULTS:
    - Sample of 15 participants
    - Average SUS score of 80.33
Testing - Usability

- **Heuristic Analysis** (developed by Jakob Nielsen)
  - Evaluate 10 heuristics: categories of consistency, errors, information design, navigation, & operation

---

**Severity**

- One: 23%
- Two: 31%
- Three: 46%

**Violation Categories**

- Errors: 23%
- Consistency: 38%
- Operation: 38%
- Navigation: 23%
- Information Design: 38%
Testing - Accessibility

- ChromeVox Chrome extension
  - Alternative to JAWS screen reader
- NoCoffee Chrome extension
  - Vision simulator
- WAVE Chrome extension
- Keyboard-based navigation
Status - What was Accomplished

Completed 95% P1 Requirements, 70% P2, 38% P3, 0% P4

- **iOS:**
  - Racer Registration
  - Automatic/Manual Check-In & Check-out
  - Route Map with Racer Locations
  - Racer ETA
  - Chat
  - Offline functionality

- **Webapp:**
  - Account Registration
  - Events CRUD
  - Admin Functionalities
Status - Future Work

- Refined ETA calculations
- Optimized Bluetooth beacon tracking
- Streamlined Bluetooth device registration
- Racer accounts, statistics, social integration
- Fundraising
Challenges

- iOS development
- Bluetooth beacon interactions
- iOS testing bluetooth & offline functionality automated
- Offline capabilities
Reflection

Positive:

- Majority (95%) of P1 requirements completed, & other priority reqs
- Effective process selection
- Frequent sponsor input and consideration

Less Positive:

- Changing requirements caused delays
- Testing was largely neglected until end
- Communication could be better
Questions?