Apr 11th, 9:45 AM - 11:45 AM

Smarter Cycling

Cory Koehler
Grant High School

Richard Smith
Grant High School

Sarah St. Clair
Grant High School

Alex Taylor
Grant High School

Aubrey Masten
Grant High School

See next page for additional authors

Presenter Information
Cory Koehler, Richard Smith, Sarah St. Clair, Alex Taylor, Aubrey Masten, and Konon Phillips

Smarter Cycling

Cory Koehler, Richard Smith, Sarah St. Clair, Alex Taylor, Aubrey Masten, Konon Phillips

Maseeh College of Engineering and Computer Science
PORTLAND STATE UNIVERSITY

Conclusion

Through the combination of the website and the pressure-sensing prototype, we can bikers, riders who know the city or are still learning their way around, BikeLock will rcreate a system that makes it easy for bikers to find available parking around the city. Whether they are veterans or beginning elieve some of the stress of biking in the city.

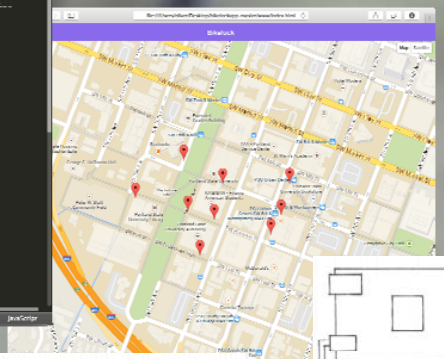
The Problem

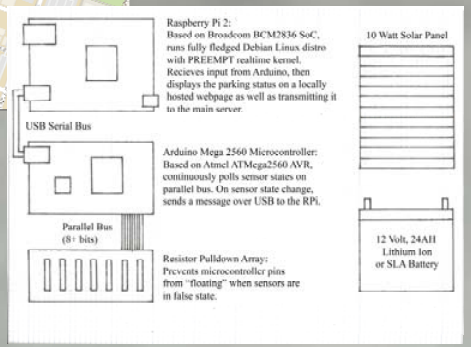
With the current trend of urbanization, the populations of major cities such as Portland are steadily increasing. This is causing a variety of problems, both within the city and in rural areas. In regards to the city, the major challenges facing city planners are the need for the expansion of residential neighborhoods and a rise in traffic throughout the city. One way to tackle the issue of an abundance of traffic, is to make alternate means of transportation more appealing to residents. We chose to focus on bicycling because of the bike-friendly culture already in place in Portland. An increase on the biking population would reduce the number of cars on the road, decreasing pollution, and cause drivers to be more aware of the bikes that ride beside them. The potential of our project to track bike theft throughout the city allows the bikers to feel more confident about their ability to avoid dangerous parking locations.

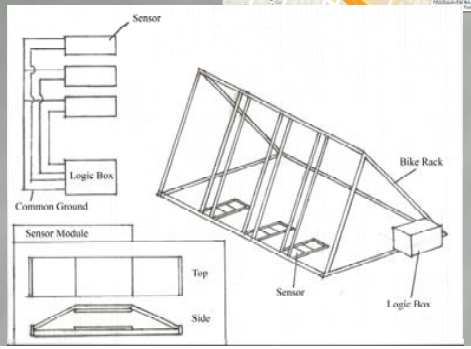
The Website and App

The website and mobile application work closely with the prototype. When a bike is parked in a space, the Raspberry Pi runs a Python script that sends an HTTP POST request containing JSON with the GPS location, the total number of parking spots and the number of spots available to the waiting Node.js server. If the parking space has special info, such as the area being protected from rain, that information is sent as well. The information is then stored in a PostgreSQL database where it can be queried at a later time. When a bike is removed from the space, the database is updated with the current number of available parking spaces.

The web application is accessed with a browser and the mobile application is available for both iOS and Android devices. Both make it easy for users to discover adequate parking nearby. Both the mobile and the web application are written using the Ionic mobile app framework, which in turn uses the AngularJS JavaScript framework. When a user opens either the mobile or the web application, their GPS location is sent to the server. The server then queries the database for nearby parking spots, returning their coordinates and the number of available parking spaces. That information is then displayed to the user, and the user can have the app direct them to the parking spot of their choice.







The Prototype

The BikeLock is designed to be easily deployable, simple in construction, and cost effective. It is based on the combination of a Raspberry Pi 2 for control and server communication, and an Arduino Mega 2560 for sensor polling and status. It all begins with the sensors. Each bike rack or parking space will have a custom pressure sensor located in each available space. The pressure sensor consists of two conductive plates sandwiched between a spring loaded non conductive plastic. When pressure is applied to the top by the weight of the bike's front wheel, the two conductive plates in the sensor are pushed together, completing a circuit which pulls the associated micro pin to a high state. The microcontroller, based on an Atmel ATmega2560, will send the information over a USB serial bus to the Raspberry Pi. Essentially, Raspberry Pi is a fully fledged computer, but in a credit-card-sized form factor. It is based on a Broadcom BCM2836 SoC and runs a complete 3.13-048 PREEMPT real time linux kernel and debian based operating system. The Raspberry Pi then relays the status of the sensors to the main server.