Outline

- Meet the Team
  3
- The Problem
  4-8
- Our Solution
  9-17
- Software Design
  18-31
- Risk Analysis
  32-35
- References
  36
- Questions
  37
Meet the Team

Mentor: Scott Silsdorf

Mary Aman

David Kuckuck

Alyssa Grow

Christopher England

Anthony McCann

Mary Aman
The Problem
Problem Statement

There is no system in place that allows staff, students, and visitors to know when a parking garage or lot has available parking or if it is full. This leads to a backup of traffic in the garages and lots which makes people late while trying to find available parking.
Counting System In Place

T2 Systems

- ODU knows that parking is a problem for many of its students and visitors.
- ODU contracted with T2 Systems to install hardware that tracks how many cars fill each lot.
- T2’s hardware is available as metal counting loops in select garages and lots.
- T2 does not develop custom mobile apps and websites for customers. An outside app would need to be integrated into the system.
The Customer

● Developing for
  ○ ODU’s Transportation and Parking Services Department

● Used by
  ○ ODU Staff
  ○ ODU Students
  ○ ODU Visitors
Customer travels to campus

Customer chooses a parking location based on permit type

Customer enters garage/lot

Customer searches for an available parking space

Customer continues an open space in that location

Garage/lot appears to be full

Customer leaves location for another garage/lot

Customer finds a spot and parks

Process Flow - Before MonarchPark
Our Solution
New Technological Additions

- Metal counting loops installed
  - Lots 42 and 43
  - All levels of Garages A and D
- Yellow delineators installed to force drivers to go through correct loops
- Future plan for expansion throughout the campus
- T2 Systems already has occupancy reporting available where loops are installed. Currently, only Parking Services has access to this information. (1)
Our Solution

- Website and mobile application
- Raw occupancy data pulled from T2 Systems. This will be accomplished with direct supervision of Transportation and Parking Services
- Information displayed on a map with a red, yellow, green indicator for each lot and each level of the garages. Below is Parking Services’ view of the data.
Our Solution cont’d

It is also our plan that staff, students, and visitors using the website or application will be able to narrow their view based on permit type. Parking location availability will be based on time of day (i.e. for the evening program).
Our Solution cont’d

Finally we plan to include lot/garage specific information such as height restrictions and enforcement hours.

*parking garage A has a height restriction sign on it. Take pic of that and include here.*
Major Functional Component Diagram

- **Hardware**
  - Sensors in Garages A and D, Lots 42 and 43
  - User Cell Phone / Personal Computer
  - MonarchPark Server

- **Software**
  - HTML, CSS2, Javascript, Google Maps API, T2 API

- **System with which the software will interface**
  - T2 Systems
What the Solution Will Not Do

- Staff, students, and visitors will not be able to reserve spots through the app/website.
- Staff, students, and visitors will not be able to pay for parking permits or resolve parking tickets through the app/website.
  - However, T2 is the same system for permit purchasing.
  - Therefore, this is functionality could be incorporated later.
- Currently the solution will not provide forecasting.
  - T2 Systems is gathering this data.
  - This can be incorporated when the T2 counting system has been live longer and is installed in all parking locations.
## What is the Competition?

<table>
<thead>
<tr>
<th>Feature</th>
<th>MonarchPark</th>
<th>ParkMe (4)</th>
<th>Parkopedia (5)</th>
<th>StreetLine Parker (6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration supported by T2 Systems</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Filtering feature on permit type</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Capacity reporting on actual counted data</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>ODU specific information (height restrictions, enforcement hours, etc)</td>
<td>Yes</td>
<td>Yes(^{(\text{inaccurate})})</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Features available on a website</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Features available in an app</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Cost of Hosting/Integration</td>
<td>Free*</td>
<td>$$$</td>
<td>$$$</td>
<td>$$$</td>
</tr>
</tbody>
</table>

\* Free if website is hosted on ODU’s servers. $1-$15 a month for website hosting from GoDaddy if not hosted on ODU’s servers. (7)
Development Approach
Hardware Requirements

- Parking Sensors
  - Provided, installed, and configured by T2 Systems
  - Maintained by Parking and Transportation Services
- User’s Smartphone and/or Personal Computer
- Server to host our website/application
  - Required to run our solution
  - Can match the specs of the typical ODU servers
  - May not need new hardware, may just create a new VM on an existing server
Software Requirements

- User’s Smartphone must be at least HTML/CSS2 capable
- User’s Personal Computer must have an OS that can run a web browser
- Server to host our website/application
  - Will communicate directly with T2 to collect occupancy data
  - Can match the specs of the typical ODU servers, such as Linux or Windows Server 2008 for example
Software Design

Both the mobile application and the full desktop presentation will be built using progressive enhancement design philosophy. Our intent is to maximize usability and accessibility for the user.
Software Design cont’d

Notifications will be displayed above the map for parking events. They will appear at a set time and last for a specified duration. The user can click on the notification for more details.
Software Design cont’d

Filtering will be available based on permit type.
Software Design cont’d

Color indicators will be available to represent occupancy. This occupancy data will be retrieved from T2 Systems and interpreted on our back-end.
Software Design cont’d

Users will be able to click on the individual parking locations to gather more detailed information.
Design Logic

- The back-end will use the T2 API to communicate with T2.
- The back-end will use the Google Maps API for our map representation of the different lots.
- Each level of a garage will be treated as its own lot with its own occupancy thresholds, and these thresholds are already set by Parking Services preferences.
- We will develop our own logic for the time-of-day transition for evening classes.
Design Logic cont’d

- Our system will interpret T2’s data, and implement the red/yellow/green color coding for occupancy based on that data.
- We will develop our own logic to implement the permit filtering to grey out unwanted parking areas.
- For all combinations of lot types, hard code will be generated by the server so every possible (or reasonable) combination of permit types is represented.
Design Logic cont’d

The server polls the T2 database for changes, and it updates the internal model to match the current database.

When a user requests the site via HTTP/HTTPS, the server generates the website through its render engine.

It is at this point the parking logic will be rendered as hard code.
Capacity and Time of Day Logic

- The two primary things that have to be taken into account are capacity and time of day when determining if a lot fits a user's filter parameters.
- The basic unit will be a “lot”. A lot is the floor upon which cars are parked.
  - Lots have attributes such as permit type that can change based on events or time of day.
  - A parking garage is made up of lots.
Capacity Logic

- Capacity coloring will be determined by two values in the server config options, the Crowded threshold and Full threshold.
- If a lot’s car to parking space ratio is below Crowded, it is green, if it is over Crowded and below Full it is yellow, and if it is over Full it’s red.
- Garage color thresholds are calculated as a percentage of total cars over total spaces.
Time of Day Logic

- Time of day is known by the server (internal or external clock).
- Time of day and events are taken into account at render while the system determines parking permit type or lot accessibility.
- Parking information will be rendered to HTML upon request, and the specific information will be based on the current time.
- Evening classes have additional parking available after a certain time and that will be reflected to users accessing the system after that time.
Risk Analysis
# Risk Matrix

<table>
<thead>
<tr>
<th>Probability</th>
<th>Impact</th>
<th>Negligible</th>
<th>Low</th>
<th>Moderate</th>
<th>Significant</th>
<th>Catastrophic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequent</td>
<td>T1</td>
<td>C1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probable</td>
<td></td>
<td></td>
<td>T1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Occasional</td>
<td>T3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remote</td>
<td>T2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Improbable</td>
<td>C2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **T1 = Garage C may never have the hardware installed**
- **T2 = T2 system upgrades**
- **T3 = Imperfect counting**
- **C1 = App compatibility**
- **C2 = Occupancy data may be too limited and reduce use**
Risk Mitigation - Technical

- **T1** - Garage C may never have the hardware installed.
- **T2** - The T2 system could be upgraded, requiring fixes in the code.
  - Mitigation: Our solution is going to be handed off to ITS after it’s completed. ITS would react to any T2 system upgrades and rework the solution as necessary.
- **T3** - Imperfect counting is possible.
  - Example: Staff golf carts are recorded when they go over loops. They may not leave the way they came in.
  - Mitigation: Train staff to go out the way they came in.

![Risk Probability Table]

<table>
<thead>
<tr>
<th>Probability</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequent</td>
<td>C1</td>
</tr>
<tr>
<td>Probable</td>
<td>T1</td>
</tr>
<tr>
<td>Occasional</td>
<td>T3</td>
</tr>
<tr>
<td>Remote</td>
<td>T2</td>
</tr>
<tr>
<td>Improbable</td>
<td>C2</td>
</tr>
</tbody>
</table>
Risk Mitigation - Customer

- **C1** - The website will be available to all; however, the app will not be available to legacy cell phones.
  - Mitigation: Users with legacy cell phones will be limited to using a computer.
- **C2** - Users may choose to not use the app if it does not cover enough parking areas.
  - This will only be a risk if Transportation and Parking Services decides to not deploy the hardware in other parking locations.
  - Mitigation: This is a proof of concept. As they add more sensors, this will become more relevant to the parking situation.

<table>
<thead>
<tr>
<th>Probability</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequent</td>
<td>C1</td>
</tr>
<tr>
<td>Probable</td>
<td>T1</td>
</tr>
<tr>
<td>Occasional</td>
<td>T3</td>
</tr>
<tr>
<td>Remote</td>
<td>T2</td>
</tr>
<tr>
<td>Improbable</td>
<td>C2</td>
</tr>
</tbody>
</table>
References

1. Interviews with Scott Silsdorf, Director of Transportation and Services, and Alex Bockelman, Information Systems Coordinator.
2. ODU Interactive Campus Map [https://www.odu.edu/about/visitors/campus-map](https://www.odu.edu/about/visitors/campus-map)
4. ParkMe [https://www.parkme.com/los-angeles-parking](https://www.parkme.com/los-angeles-parking)
7. GoDaddy [http://tinyurl.com/gp5gf2u](http://tinyurl.com/gp5gf2u)
Questions?