Transportation for Visual Impaired  
Cycle 1 Report

By

Aymeric Zuurhout

Tyler Espy

Taylor Lucy

Submitted in partial fulfillment of the requirements for  
COMP 4710 Senior Design  
to the Department of Computer Science and Software Engineering,  
Samuel Ginn College of Engineering, Auburn University

Auburn, Alabama

October 4, 2015

# Table of Contents

Table of Contents ii

2 Executive Summary (System Metaphor) 1

3 Project Introduction 2

3.1 Previous Development 2

3.2 Intent This Cycle 2

3.3 Future Work 2

4 Requirements / User Stories 3

4.1 User Stories 3

4.2 User Stories 3

4.2.1 Off-Screen Ghost Markers 3

5 Design Documentation 5

6 Management Plan 6

6.1 Task Assignments 6

6.2 Development Schedule 6

6.3 Planned Code / Feature Freeze 6

7 Risk Mitigation 7

8 Test Plan and Test Procedures 8

8.1 Test Plan 8

8.2 Test Procedures 9

8.2.1 Procedure 1: Sample Test Procedure 9

9 Lessons Learned 10

References 11

Appendix A 12

A.1 Status Reports 12

A.2 Meeting Minutes 12

A.3 Size Estimation Documentation 12

A.4 Problem Reports / Change Requests 12

A.5 Correspondence 12

A.6 Source Code 12

General Instructions for your Cycle Report

Basically, any written material you use or generate should be included in your cycle binder. This may include:

* Copies of all email correspondence related to the project.
* Design and process documents such as user stories, use cases, UML diagrams, database diagrams, user interface storyboards, requirements documents, etc.
* Copies of all status reports for the cycle.
* Your system metaphor, management plan, project schedule, etc.
* Your cycle Powerpoint presentation.
* All source code, database or XML schema, scripts, etc.

### Printouts

Bring TWO (2) printed copies of your cycle binder to your presentation. We will write grading comments on these and return them to you for your review. You may omit lengthy non-process sections of the cycle binder (such as source code) from the printed copy.

Bring TWO (2) printed copies of your PDF slides for the instructors.

# Executive Summary (System Metaphor)

Alabama Institute for the Deaf and Blind has a transportation service located in the greater Talladega area, which currently does all their reservations manually by hand. Our goal is to modernize, optimize, and automate their reservation process with an online database system and a webpage and IPad application frontend.

# Project Introduction

In this project we are working for the Alabama Institute for the Deaf and Blind. They are located in Talladega and work with the two schools in the area. They have a transportation system, which aids those in the community who are unable to commute on their own. Currently, the system they use to track appointments and verify that they have drivers for these commutes are by Outlook calendar and hand writing the information about each appointment and are stored in a binder. They point of our project is to allow them to all of this, all in one place. We want allow the user to use a calendar and visually see all event happening, either by choice of month, week, or day. From there the user can select a date and create a new event, or reservation. Doing this would allow them to have all the information needed in one location, as well as have a database for them that they can search instead of everything being on paper and in binders.

## Previous Development

Being that we are creating a project from the beginning the only development before this cycler were those in the Architectural Spike. During the Architectural Spike we had set a goal to have a working form, rough calendar layout, and database. We completed the form, partially the calendar, and only set up the server. The reason being that we were unable to finish all of it was because we underestimated how much work there goes behind a calendar, and the database was not set because we were told old by our sponsor’s that we would be able to have access to their server, but was later told that they would like to see a function product before giving us access to theirs.

## Intent This Cycle

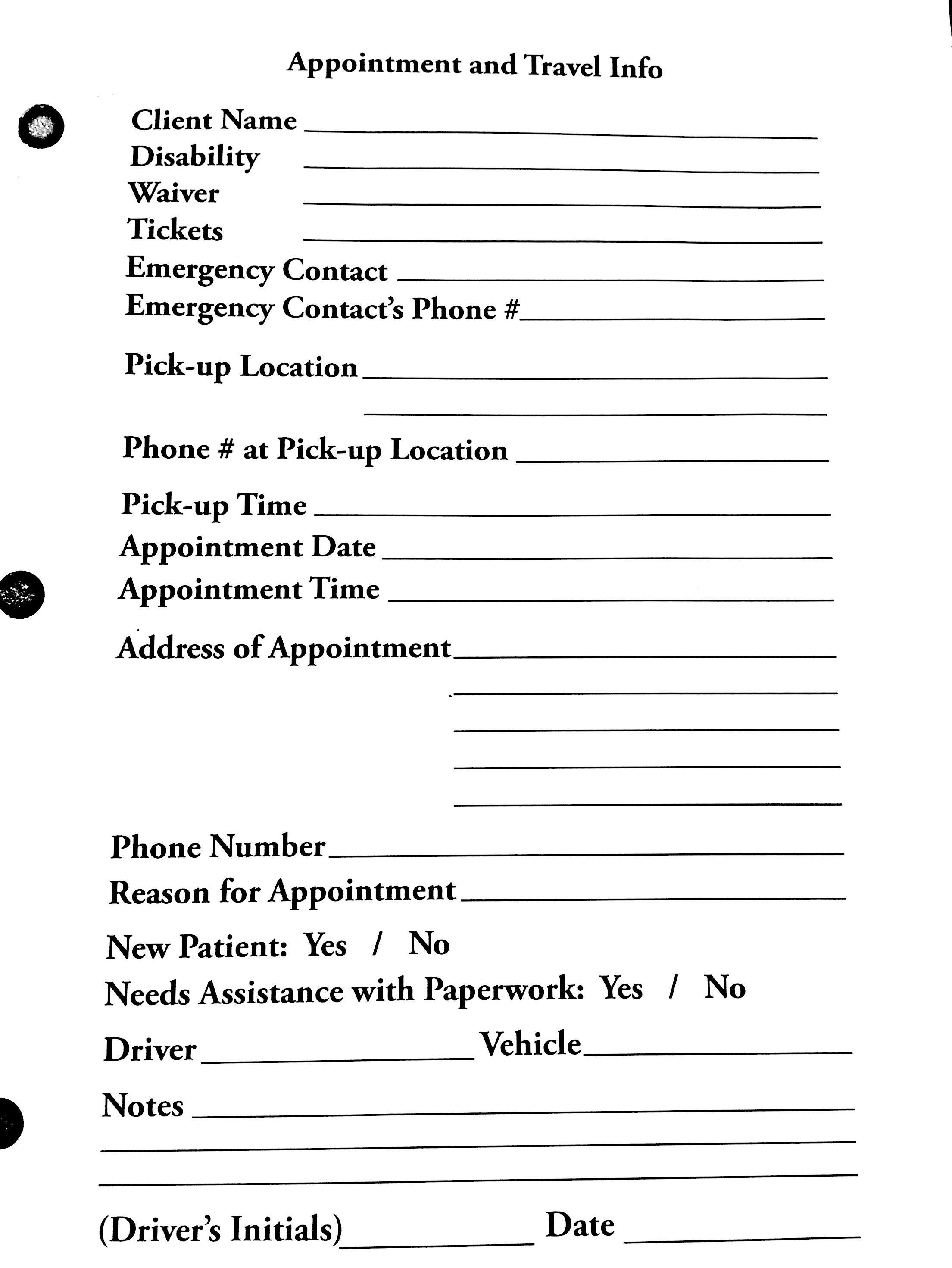
The intent of this cycle was to continue working on the form, improve the calendar, and have a working database. However, meeting with our sponsors and with them review what we currently had, a lot of changes were to be made and features to be added. With the form, they were alright with the layout on the web-page, but they told us they would like to still have the ability to print it out. The reason being because they still want to make their drives physically sign off on the reservation they have accepted. So we implemented that, as well as being able to select an event off the calendar and a filled out form with all the information about that event would come up. This of course comes along side that our database is working and interacting properly. The calendar this time is to have the ability to see all event happening in months, week, or day organization look at, which the user will be able to select based on their preference. This detail allows one of the users, who is visually impaired themselves, to read the information easier, instead of having to remember their location on the screen.

## Future Work

Our future plans, involve having the backend of our product working. We are currently working on all the frontend and having all the features our sponsor wants. However if we want to be able to test our product and make sure all functionalities are working, we need our backend up and running. So the very first thing we will be doing next cycle is making sure our .php along with out server is communicating properly. We will work on making sure that information is saved when the user submits a form, the date is accurately place on the calendar, and that information can be retrieved and edited. Once we have that all working, overall our general concept of our project should be working. The next steps will be all the little feature and add-ons that will improve the usability for our sponsor.

# Requirements / User Stories

Below is the form they current use right now to create reservations, and the form we are to implement in our project. They want to have the ability to print it out and look similar to this.



## User Stories

### Adding a New Reservation

Summary: User receives a call and asking the questions needed to fill in the form, and user saves it and it is displayed on the calendar.

Description: Customer wants to be able to fill in the form when they receive a call and all information is saved.

Hours: Total Planned: 30  
Planned this cycle: 10  
Total Actual: 15  
Actual this cycle: 9

Coder: Aymeric Zuurhout, Tyler Espy, Taylor Lucy

Tester: Tyler Espy

Reviewer: Entire team

Status: Still in progress. Since this is the main user story and allows all other functions and interactions, all things relate to this working properly. We currently do not have full connectivity between making a reservation and the calendar, which are our most important focus right now and next step to be completed.

### User has the ability to see current reservations

Summary: User can look at the calendar and see all reservations that have been planned and the information about each one.

Description: Customer wants to be able to see on any day on the calendar what reservations are planned. From here they can make an executive decision on whether they can make a new reservation on that day, or what available time there is that day.

Hours: Total Planned: 30  
Planned this cycle: 10  
Total Actual: 5  
Actual this cycle: 5

Coder: Aymeric Zuurhout, Tyler Espy,

Tester: Tyler Espy

Reviewer: Entire team

Status: Still in progress. This relates mainly to user story 1 to work, and therefore for it to happen we need a database that is working. We believe right now we have the functionality in place for it work, but we do not have a running database that has proper connectivity.

# Design Documentation

The purpose of the application is to allow the user to input information concerning their client’s reservation. This application will allow them to move away from the old use of pen, paper, and carbon paper to the modern, and commonly used, digital logging. Doing this will allow them to easily create and find reservations for their clients, instead of searching though file cabinets.

Our process discussion was that our initial design thought was to create a calendar like function that would have access to the form need to create a reservation. However, after a bit of thought, we realized that this needed to be user friendly to people who are either blind or deaf. Since we are dealing with the deaf and blind, features that they believe will help them use our product easier is very important to them. Simple things like drop down menus, to having all reservations next to each other in order, and having reservations color-coded are just some of the simple features that will make their life much easier. We implemented most of these in order to demonstrate to our customer to make sure that is what they wanted.

At the code level, we are currently using .JS (JavaScript) files for the functionality, .HTML (Hypertext Markup Language) files to display text using the web, .CSS (Cascading Style Sheets) files to format and display the elements within the HTML files, and .PHP (Hypertext Preprocessor) which is a server-side scripting language designed for web development. On an organization level we have a form.html and a form.css for the form that the user sees and inputs text. Then we have a calendar.html along with calendar.css and calendar.js to display the calendar and have all the functionality within to work. Together the form.html and calendar.html communicate to make the interaction the user sees. One of the future files that we will create is the basic form look, so the user can see what the print out will look like. Other future files may include the displaying of car and drivers that are available at certain times and dates, but that will be a decision made when that feature comes up.

At an overview of how we modeled it, we set up to where the user would select a certain date. From there you can see all current reservations for that date, get a driver, get a vehicle, or if there isn’t current a reservation to create one. Selecting one of the current reservations will allow you to edit that form. However if select to create a new reservation, a blank form will appear and will be used to fill out all the information need for the customer.

How the system works:

1. Customer will view calendar from AIDB website.
2. User selects whether to add a new reservation or current reservation.
3. If user selects to add a new reservation, they will sent to form asking questions to be filled in.
4. Once the user fills in the information they will submit it and it will be saved.
5. The event created should then be displayed on the calendar.
6. From the calendar point of view, the user will have to see all reservations scheduled.
7. Also from the calendar the user will have the ability to select one of the reservations created, and from there it will pull up the form filled out with the information previously put in.
8. Lastly the user will have the ability to print out the form, in a much simplistic look.

During this cycle we had to take a few steps back, as well as add a few things that we didn’t not expect. In the beginning we had a general belief of what our sponsor wanted. However when we went to meet them in Talladega we has misunderstood some desires. We took notes and made many changes. But we believe to be back on track and progress was of course made.

# Management Plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | User Story | Cycle planned for completion | Total planned hours | Status | Actual hours this cycle |
| 1 | User wants to be able to when they receive a call add a new reservation. | 1 | 10 | In Review | 10 |
| 2 | User picks data and sees real time car availability. | 3 | 25 | Unstarted | 0 |
| 3 | User wants to be able to see all current/available time to make reservation. | 2 | 15 | In Review | 20 |
| 4 | User's information is stored within a database | 2 | 15 | Collaboration | 5 |
|  |  | **Planned Total** | 65 | **Actual Total** | 35 |

## Task Assignments

**Calendar**

Hours: Aymeric Zuurhout – 14

Tyler Espy – 20

Taylor Lucy – 0

**Form**

Hours: Aymeric Zuurhout – 8

Tyler Espy – 14

Taylor Lucy – 0

**Database**

Hours: Aymeric Zuurhout – 0

Tyler Espy – 0

Taylor Lucy – 4

**Visit Sponsor**

Hours: Aymeric Zuurhout – 7

Tyler Espy – 7

Taylor Lucy – 0

# Risk Mitigation

During this cycle we didn’t take to many risks since when we meet with our sponsor we had a lot more on our plate then we expected. So we tried to minimalize as many

# Test Plan and Test Procedures

Since we are constructing a web-based application, testing our product mostly happens when we functionally do something. So testing would require our product to first off be in place with the able to read and write what we specify it to. Connectivity between all our files must be in place too. All functionalities right now are connected, but we are trouble shooting our database right now. Information is not being saved; therefore, almost all our user stories are put on halt until we get our problem with our database resolved.

# Lessons Learned

Lessons learned is an invaluable section of your report for teams that come after you. Anything you tried that didn’t work, any technologies or solutions you considered or attempted and then abandoned, any problems with parts, components, vendors, software APIs, etc., should be documented here.

We are not interested in lessons such as “we learned how to better communicate as a team” or “we learned how to set up a database using Microsoft SQL Server.” That is pointless – you’re expected to develop team skills in this course and you’re expected to develop or improve your skills with new tools.

Instead, this section, which is arguably one of the most important in your report, should serve as a roadmap for future work and help future engineers avoid some of the problems or roadblocks you encountered.

# Appendix A Supporting Documents

## Status Reports

XXXXXX

## Meeting Minutes

XXXX

## Size Estimation Documentation

XXX

## Problem Reports / Change Requests

XXXX

## Correspondence

Include all correspondence, such as email, chat logs, message boards, etc., between:

* The team and the customer.
* The team and the instructor(s)/manager(s).
* Individual team members.

## Source Code

Finally, you should include source code in your PDF. This includes:

* All source code
* Database create scripts, stored procedures, etc.
* Administration info (IP addresses, server/machine names, user names, passwords, gmail lists, dropbox or sharepoint accounts, etc.)
* Version information (e.g. README.TXT)
  + A Version Description is the “README” for the delivered product.
  + A Version Description should contain the following:
    - Version number
    - Description of the application
    - Key features
    - Known bugs/issues

You DO NOT and SHOULD NOT print this lengthy portion of your report and hand it in to the instructors. Save a tree, and perhaps the planet.