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

1 Executive Summary (System Metaphor) 1

2 Project Introduction 2-3

2.1 Previous Development 4

2.2 Intent This Cycle 4

2.3 Future Work 4

3 Requirements / User Stories 5

3.1 User Stories 5

3.2 User Stories 6

3.2.1 Off-Screen Ghost Markers 6

4 Design Documentation 7-8

5 Management Plan 9

5.1 Task Assignments 6

5.2 Development Schedule 6

5.3 Planned Code / Feature Freeze 6

6 Risk Mitigation 7

7 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

# 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 IOS application frontend.

# Project Introduction

In this project we are building a transportation system for the Alabama Institute for the Deaf and Blind. Located in Talladega with the two functioning schools in the area. Their current transportation system is directed towards aiding those in the community who are unable to commute on their own due to their disabilities. The Transportation Department consists of three individuals, Zoe Rose, Jennifer Reeve and Donte Little. Both Ms. Rose and Ms. Reeves later referred hereafter as “schedulers” handle scheduling responsibilities with Mr. Little as their manager. There current system is mostly done by hand. They field calls from clients needing transportation to and from a specific location anywhere in Alabama on a specific date. The date, location and time varies and is handled on a first come first serve basis. During the call schedulers, follow a simple process described below: (Steps followed in sequential order)

1. Check for availability on an Outlook calendar
   1. If opening is available they ask for client information
   2. If opening is unavailable they decline give the client available openings
   3. When opening is chosen schedulers move on to step 2
2. Ask client for information and handwrite the information on Appointment sheet
3. Create and event in outlook on the date chosen
4. Insert scheduled information in event
   1. Client Name
   2. Start Time
   3. End Time
   4. Scheduled driver
   5. Vehicle
5. Store hand written Appointment sheet in binder for future use
6. Field further calls repeating step 1-6

At the time of an appointment

1. Schedulers find Appointment sheet in binder
2. Transfer information from Appointment sheet to Driver Information Sheet
3. Driver picks up the sheet from schedulers and conducts the transport
4. Upon successful transport Driver initials form and returns it to schedulers
5. Schedulers file form record keeping
6. Steps 7 – 9 continue until all transports have been conducted

Our project is devoted to making this process easier by digitalizing it, replacing hardcopies with digital forms. Our system will feature connectivity between three primary forms:

1. Calendar Form
2. Scheduling Form
3. Driver Information Form

All of which retain an ability for easy printing to make record keeping easier, an auto-fill feature will be implemented to reduce the redundancy of filling in the same information across forms. Our users (schedulers) themselves are visually impaired, to facilitate their disabilities all forms are made compatible with programs such as JAWS and Zoomtext designed to help visually impaired read and use information. The calendar also contains the aesthetic qualities to help our customers see events, by choice of month, week, or day. Each event is placed close the one prior, removing white space that previously inhibited the schedulers. With the calendar the schedulers will be able to select a date, view currently scheduled appointments, and if an opening exists create a new appointment. Each day contains a link to the Scheduling Form to make scheduling a new appointment easy. After an appointment is saved it is viewable from the calendar. On the day of an appointment the scheduler, can simply click the event upon which a link transfers them to an auto-filled Driver Information Form. From there they can view appointment information and simply print out the form for Drivers Use. A new step by step process is shown below:

1. Check for availability on calendar
   1. If opening is available they ask for client information
   2. If opening is unavailable they decline give the client available openings
   3. When opening is chosen schedulers move on to step 2
2. Ask client for information and enter information in Appointment sheet
3. Print out Appointment Sheet and save appointment to calendar
4. Store Appointment sheet in binder for record keeping
5. Field further calls repeating steps 1-4

At the time of an appointment

1. Schedulers click on appointment in calendar
2. Print out Driver Information Form
3. Driver picks up the sheet from schedulers and conducts the transport
4. Upon successful transport Driver initials form and returns it to schedulers
5. Schedulers file form record keeping
6. Steps 7 – 10 continue until all transports have been conducted

## Previous Development

Starting from scratch during our Architectural Spike we set a goal to have an initial working form, rough calendar layout, and set up development tools. We completed the initial form, a date selector and set up development tools listed below.

1. Git repository for team sharing
2. AptanaStudio3 workspace for html, css, php, JavaScript development
3. Development Server

## Intent This Cycle

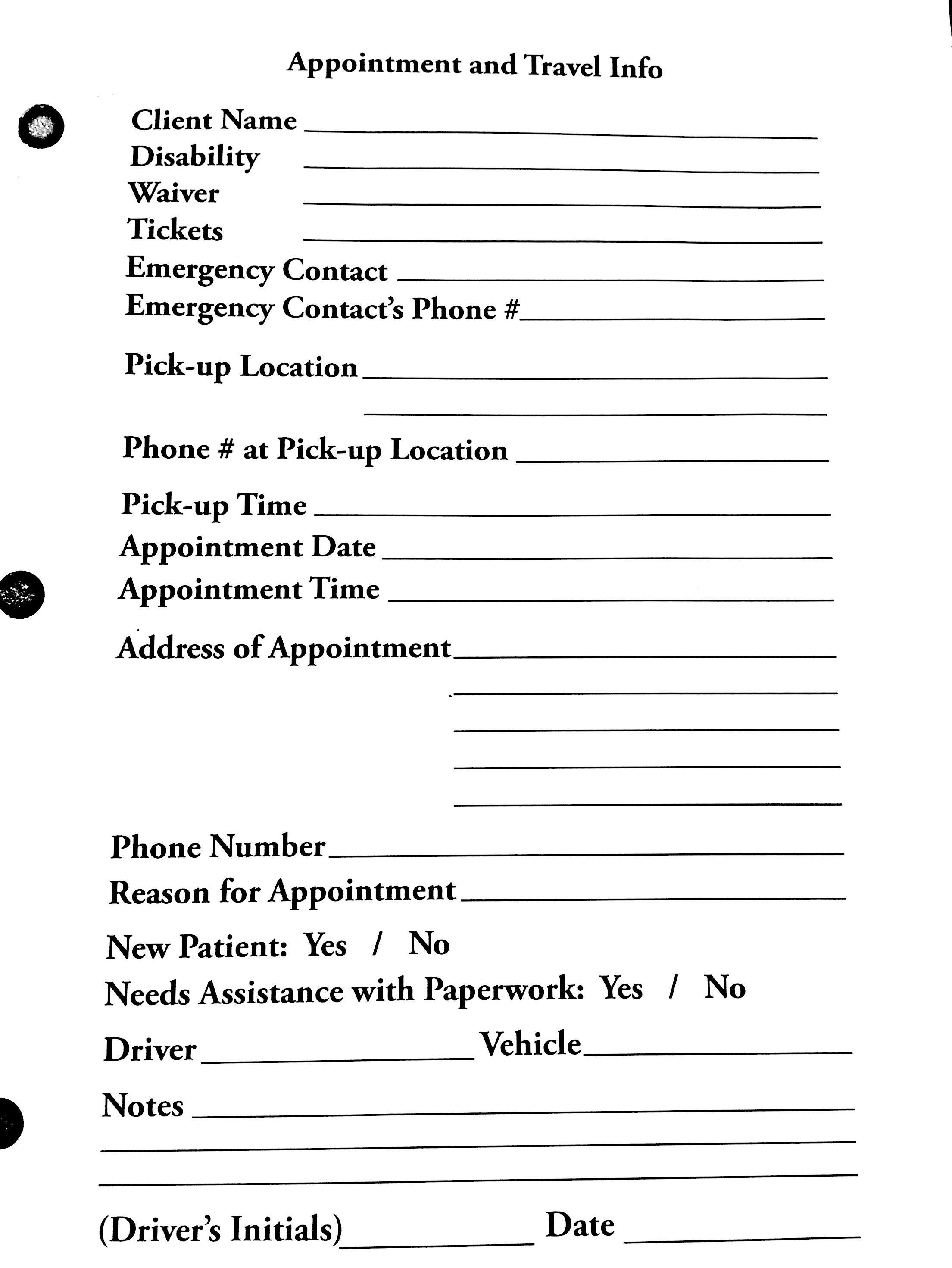
The initial intent of this cycle was to review initial form and calendar with clients making changes where needed, and create initial database schema. We had the opportunity to meet our sponsors in Talladega and demo our product. After seeing their system first hand a lot of needed changes to the original product were discovered. For example originally we had only planned for one form and a date selection service, but upon seeing their system we realized a need for a visual calendar design, as well as an addition form devoted to informing the driver. They still wanted to make their drivers physically sign off on the reservation they had accepted. We devoted considerable time to calendar implementation and creating the desired additional form. We created the form to desired specifications, and a calendar with the ability to see all events organized by months, weeks, or days which the user will be able to select based on their preference. After seeing the schedulers use their system we also expended considerable effort implementing attributes within our calendar to help schedulers with limited vision. We also created a MySQL database to store reservations, drivers, and vehicle information.

## Future Work

With our Front End interface almost complete our future plans, revolve around implementation of the backend of our product. An important feature of our product is record keeping. We have initial schemas implemented in a MySQL database but have not implemented an ability to store or access information from our web service. Our next step is implementing the ability to store information from our Scheduling Form, upon saving, to the reservation table created in our MySQL database. With that completed we will populate calendar days with reservation events. Afterward we will add auto-filling attributes to the Driver Information Form, viewed from the calendar. Upon completing these steps, we can begin to further enhance, the calendar form to better serve our sponsors.

# Requirements / User Stories

Below is the form currently in use as of now to create reservations, and the template for our Driver Information 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 asks questions needed to fill out Appointment Form. User completes and saves form which is then 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 and added to the calendar.

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 is 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 decisions 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 copies 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.

Design Specs:

1. Calendar is viewable via link from AIBD webpage
2. Calendar is populated via information stored on AIBD servers
   1. As of now this information is stored on our development server using MySQL
   2. When ready for production our scripts will move required schemas to their database
3. Calendar is linked to two forms:
   1. Appointment Form
   2. Driver Information Form
4. Appointment form is linked to each day on the calendar
   1. Using php, Appointment dates are auto-filled
   2. Upon saving Appointment Information is stored in Reservation Table created in MySQL database
5. Driver Information Form is linked to each event on a given day
   1. Mapped to Reservation Table in MySQL database
   2. Form is auto-filled using data stored in MySQL database
   3. Form is printable
   4. It is NOT however editable

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 needed progress was 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.