Augmented Education

Senior Design Final Documentation

Augmented Education

Aaron Alphonsus Cheldon Coughlen Daniel Hodgin Kenneth Petry
Savoy Schuler Brady Shimp

November 27, 2017

Contents

List of Figures

List of Tables

List of Algorithms

Document Preparation and Updates

Current Version [0.0.1]

Prepared By:
Kenneth Petry
Brady Shimp

Revision History

Date	***Author***	***Version***	***Comments***
2/2/12	*Team Member #1*	*1.0.0*	*Initial version*
3/4/12	*Team Member #3*	*1.1.0*	*Edited version*

1

Overview, Description and Deliverables

The overview should take the form of an executive summary. Give the reader a feel for the purpose of the document, what is contained in the document, and an idea of the purpose for the system or product.

1.1 Team Members and Team Name

Team Name: Augmented Education
Team Members:

- Aaron Alphonsus
- Cheldon Coughlen
- Daniel Hodgin
- Kenneth Petry
- Savoy Schuler
- Brady Shimp

1.2 Client

A description of the client or customer. Description of sponsor if different than client. Brief statement of customer's problem or goal for this project List the customer needs

1.3 Project

A high level description of the project. Project environment ... Project boundaries. Project context. Technical Environment. Current systems overview.

1.3.1 Mission Statement

Mission statement for the team inserted here.

1.3.2 Elevator Pitch

Elevator Pitch for your project inserted here.

1.3.3 Purpose of the System

What is the purpose of the system or product?

1.4 Business/Market Need

Use this section to define what business need exist and how this software will meet and/or exceed that business need. How do you make money!! What is the revenue model? What is the market? Who are customers?
Example: Mouse Detector Phone App

Product Description: iPhone based app that can detect the high frequency sounds of mice and locate them.

Key Business Goals: Product introduced in the second quarter 2009

- 50% gross margin
- 15% share of mouse trap market

Primary Market: Consumers

Secondary Markets: Lazy cats

Assumptions:

- Available from App store
- Survillence mode
- Low power consumption
- Autodial on detection

Stakeholders:

- User
- Retailer
- Sales Force
- Production
- Legal department

Certifications: Apple, Cat Fancy Magazine

1.5 Deliverables

Provide a complete description of the client requested deliverables. This section should be the section that your software contract refers to. (e.g. prototype, documentation, code, users manual, ...)

1.5.1 Software

The main deliveralbe is a swoftware toolchain to save, retrieve, and view 3D models produced in popular modeling software. The two main components are:

1. A website to manage users' files
   - Upload files
   - Save files
   - Run software to convert between 3D file types
   - Serve files back to users
2. A file conversion program to convert a users uploaded file into a viewable file type
   - Convert a given 3D model into a common file type to be stored on the website
   - Convert the common file type to the type needed to be viewd on an Augmented Reality device

1.5.2 Hardware

Test the flow of the website and file conversion software on popular Augmented Reality devices, witch may include:

- Microsoft Hololens
- Meta 2
- Mobile devices running IOS and/or Android

1.5.3 Documentation

And so on. Anything that your contract states that you will deliver to the client.

2

User Stories, Requirements, and Product Backlog

2.1 Overview

The overview should take the form of an executive summary. Give the reader a feel for the purpose of the document, what is contained in the document, and an idea of the purpose for the system or product.

The user stories are provided by the stakeholders. You will create he backlogs and the requirements, and document here. This chapter should contain details about each of the requirements and how the requirements are or will be satisfied in the design and implementation of the system.

Below: list, describe, and define the requirements in this chapter. There could be any number of sub-sections to help provide the necessary level of detail.

2.2 User Stories

This section can really be seen as the guts of the document. This section should be the result of discussions with the stakeholders with regard to the actual functional requirements of the software. It is the user stories that will be used in the work breakdown structure to build tasks to fill the product backlog for implementation through the sprints.

This section should contain sub-sections to define and potentially provide a breakdown of larger user stories into smaller user stories. Each component must have a test identified, meaning you need to know how you plan to test it. If a requirement is not testable, then some justification needs to be made on why the requirement has been included. The results of the tests should go in the testing chapter.

2.2.1 Sprint Zero

Main Goal:

View a Maple 3D model on a Microsoft Hololens

2.2.1.a AR Rendering

- As a faculty member, I want a Maple file to be automatically converted into an AR Tag on a cloud server.
- As a user, I want to be able to view an AR tag through a Microsoft Hololens to render a 3D model.

2.2.1.b Website Hosting

- As a faculty member, I want to upload a Maple 3D model to a cloud server.
- As a faculty member, I want to be able to download the AR tag for my document from a cloud server.

2.2.1.c Sprint Zero Breakdown

User stories can be broken down into two main categories: AR Rendering and Website Hosting. Half of the team will primarily work on the AR Rendering stories, and the other half the Website Hosting stories. The mail goal of these user stories is to view a 3D model from the Maple software on a Microsoft Hololens where the files are stored and managed in the cloud.

2.2.2 Sprint One

2.2.3 Sprint Two

2.2.4 Sprint Three

2.2.4.a AR Rendering

- As a user, I want to be able to view surface materials
- As a user, I want to be able to slice a 3D model and view a section of the model
- As a user, I want to be able to be able to switch between models quickly in the AR device

2.2.4.b Sprint Three Breakdown

The clients shared some of their requests for how the files are rendered and viewed in an AR device. These include viewing surfact

2.2.5 Sprint Four

2.2.6 User Story #2

2.2.6.a User Story #2 Breakdown

User story #2 ....

2.2.7 User Story #3

2.2.7.a User Story #3 Breakdown

User story #3 ....

2.3 Requirements and Design Constraints

Use this section to discuss what requirements exist that deal with meeting the business need. These requirements might equate to design constraints which can take the form of system, network, and/or user constraints. Examples: Windows Server only, iOS only, slow network constraints, or no offline, local storage capabilities.

2.3.1 System Requirements

The basic system requirements to use the website are to have a web browser installed with internet access. The user must have access to modeling software or a method to create/provide 3D models to the website.

In order to fully use the product, a user must have an Augmented Reality device. Each device may have different system requirements.

For example, the Meta 2 requires a seperate computer in order to run. The minimum and recommended specifications are listed below. The list was crated in November 2017.

	Minimum	Recommended
OS	Windows 10 (64 bit)	Windows 10 (64 bit)
CPU	Intel i7-4770	Intel i7-6700
RAM	8GB DDR3	16GB DDR4
GPU	NVIDIA GTX 960	NVIDIA GTX 970
Hard Drive	2GB Free Space	2GB+ Free Space
I/O Ports	1X HDMI 1.4b and 2X USB 3.0 ports	1X HDMI 1.4b and 2X USB 3.0 ports
3D Engine	Unity 5.6 or higher	Unity 5.6 or higher

More up to date requirements can be found on the Meta 2 website at: `https://buy.metavision.com/`

2.3.2 Network Requirements

What are they?

2.3.3 Development Environment Requirements

What are they? Is the system supposed to be cross-platform?

2.3.4 Project Management Methodology

The stakeholders might restrict how the project implementation will be managed. There may be constraints on when design meetings will take place. There might be restrictions on how often progress reports need to be provided and to whom.

2.4 Specifications

Any specifications that need to be understood? Put it here.

2.5 Product Backlog

The full initial product backlog should go here. The sprint backlogs are located in the prototypes chapter.

- What system will be used to keep track of the backlogs and sprint status?
- Will all parties have access to the Sprint and Product Backlogs?
- How many Sprints will encompass this particular project?
- How long are the Sprint Cycles?
- Are there restrictions on source control?

2.6 Research or Proof of Concept Results

This section is reserved for the discussion centered on any research that needed to take place before full system design. The research efforts may have led to the need to actually provide a proof of concept for approval by the stakeholders. The proof of concept might even go to the extent of a user interface design or mockups.

2.7 Supporting Material

This document might contain references or supporting material which should be documented and discussed either here if appropriate or more often in the appendices at the end. This material may have been provided by the stakeholders or it may be material garnered from research tasks.

3

Project Management

This section provides some housekeeping type of information with regard to the team, project, environment, etc.

3.1 Team Member's Roles

The team is divided into two main parts:

1. Website
    - Daniel Hodgin
    - Brady Shimp
    - Savoy Schuler
2. Conversion Software
    - Aaron Alphonses
    - Cheldon Coughlen
    - Kenneth Petry

The website team is responsible for creating the web portal that users will interact with. The duties of the website include: uploaded file management, user authentication, running the conversion software, etc.

The conversion software is responsible for converting 3D models from an uploaded file type, to one usable by Augmented Reality devices. The software will be run by the website when a user or AR device requests a file. The team will also be responsible for determining which file type to stored in the backend of the website.

3.2 Project Management Approach

This section will provide an explanation of the basic approach to managing the project. Typically, this would detail how the project will be managed through a given Agile methodology. The sprint length (i.e. 2 weeks) and product backlog ownership and location (ex. Trello) are examples of what will be discussed. An overview of the system used to track sprint tasks, bug or trouble tickets, and user stories would be warranted.

3.3 Stakeholder Information

This section would provide the basic description of all of the stakeholders for the project. Who has an interest in the successful and/or unsuccessful completion of this project?

3.3.1 Customer or End User (Product Owner)

Who? What role will they play in the project? Will this person or group manage and prioritize the product backlog? Who will they interact with on the team to drive product backlog priorities if not done directly?

3.3.2 Management or Instructor (Scrum Master)

Who? What role will they play in the project? Will the Scrum Master drive the Sprint Meetings?

3.3.3 Investors

Are there any? Who? What role will they play?

3.3.4 Developers –Testers

Who? Is there a defined project manager, developer, tester, designer, architect, etc.?

3.4 Budget

Describe the budget for the project including gifted equipment and salaries for people on the project.

3.5 Intellectual Property and Licensing

Describe the IP ownership and issues surrounding IP.

3.6 Sprint Overview

If the system will be implemented in phases, describe those phases/sub-phases (design, implementation, testing, delivery) and the various milestones in this section. This section should also contain a correlation between the phases of development and the associated versioning of the system, i.e. major version, minor version, revision.

All of the Agile decisions are listed here. For example, how do you order your backlog? Did you use planning poker?

3.7 Terminology and Acronyms

Provide a list of terms used in the document that warrant definition. Consider industry or domain specific terms and acronyms as well as system specific.

3.8 Sprint Schedule

The sprint schedule. Can be tables or graphs. This can be a list of dates with the visual representation given below.

3.9 Timeline

Gantt chart or other type of visual representation of the project timeline.

3.10 Development Environment

Both teams agreed to use the Microsoft ecosystem to develop the product.

3.11 Development IDE and Tools

The IDE of choice for the website and file conversion team is Visual Studio 2017 Enterprise Edition.

To compile the web conversion software two libraries are needed.

- Autodesk's FBX SDK is required to export .fbx files. It must be installed in a folder located in the project directory nmed F̈BX SDK. The download can be found at: `http://usa.autodesk.com/adsk/servlet/pc/item?siteID=123112&id=26416244`. The Windows VS2015 version must be installed.
- Open Asset Import Library supports a wide variety of import and export file types. The download can be found at: `http://assimp.org/main_downloads.html`. Version 3.1.1 is what was used in the project.

3.12 Source Control

Which source control system is/was used? How was it setup? How does a developer connect to it?

3.13 Dependencies

Website

File Conversion

FBX SDK A library produced by Autodesk that converts from a select few file types to the .fbx file that is easily viewed on Microsoft supported software (Windows 10, Hololens).

Open Asset Import Library A library the reads and writes multiple file types (does not export to .fbx).

3.14 Build Environment

How are the packages built? Are there build scripts?

3.15 Development Machine Setup

If warranted, provide a list of steps and details associated with setting up a machine for use by a developer.

4

Design and Implementation

4.1 Systems Goals

Briefly describe the overall goals this system plans to achieve. These goals are typically provided by the stakeholders. This is not intended to be a detailed requirements listing. Keep in mind that this section is still part of the Overview.

4.2 System Overview and Description

Provide a more detailed description of the major system components without getting too detailed. This section should contain a high-level block and/or flow diagram of the system highlighting the major components. See Figure 4.1. This is a floating figure environment. LaTeX will try to put it close to where it was typeset but will not allow the figure to be split if moving it can not happen. Figures, tables, algorithms and many other floating environments are automatically numbered and placed in the appropriate type of table of contents. You can move these and the numbers will update correctly.

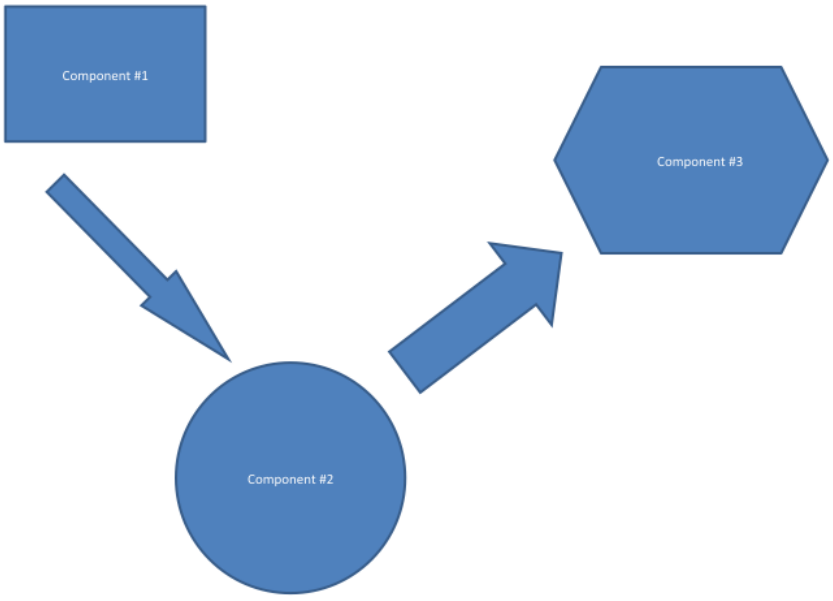

Figure 4.1: A sample figure .... System Diagram

4.2.1 Major System Component #1

Describe briefly the role this major component plays in this system.

4.2.2 Major System Component #2

Describe briefly the role this major component plays in this system.

4.2.3 Major System Component #3

Describe briefly the role this major component plays in this system.

4.3 Technologies Overview

This section should contain a list of specific technologies used to develop the system. The list should contain the name of the technology, brief description, link to reference material for further understanding, and briefly how/where/why it was used in the system. See Table 4.1. This is a floating table environment. LaTeX will try to put it close to where it was typeset but will not allow the table to be split.

Table 4.1: A sample Table ... some numbers.

7C0	hexadecimal
3700	octal
11111000000	binary
1984	decimal

4.4 Architecture and System Design

This is where you will place the overall system design and the architecture. This section will be very detailed and should be image rich. There is the old phrase *a picture is worth a thousand words*, in this class it could be worth hundreds of points (well if you sum up over the entire team). One needs to enter the design and why a particular design has been done. THIS IS THE CORE OF THE COURSE.

It is important for you to say why as much as what.

4.4.1 Design Selection

Failed designs, design ideas, rejected designs here.

4.4.2 Data Structures and Algorithms

Describe the special data structures and any special algorithms.

4.4.3 Data Flow

4.4.4 Communications

4.4.5 Classes

4.4.6 UML

4.4.7 UX

4.4.8 UI

4.4.9 MVVM, etc

4.5 Major Component #1

If the following makes sense, use this outline, if not then modify the outline

This section is used to describe the design details for each of the major components in the system. Note that this chapter is critical for all tracks. Research tracks would do experimental design here where other tracks would include the engineering design aspects. This section is not brief and requires the necessary detail that can be used by the reader to truly understand the architecture and implementation details without having to dig into the code. Sample algorithm: Algorithm 1. This algorithm environment is automatically placed - meaning it floats. You don't have to worry about placement or numbering.

Algorithm 1 Calculate $y = x^n$

Require: $n \geq 0 \vee x \neq 0$
Ensure: $y = x^n$
  $y \Leftarrow 1$
  if $n < 0$ **then**
    $X \Leftarrow 1/x$
    $N \Leftarrow -n$
  else
    $X \Leftarrow x$
    $N \Leftarrow n$
  end if
  while $N \neq 0$ **do**
    if N is even **then**
      $X \Leftarrow X \times X$
      $N \Leftarrow N/2$
    else $\{N$ is odd$\}$
      $y \Leftarrow y \times X$
      $N \Leftarrow N - 1$
    end if
  end while

Citations look like [2, 1, 3] and [6, 4, 5]. These are done automatically. Just fill in the database `designrefs.bib` using the same field structure as the other entries. Then pdflatex the document, bibtex the document and pdflatex twice again. The first pdflatex creates requests for bibliography entries. The bibtex extracts and formats the requested entries. The next pdflatex puts them in order and assigns labels. The final pdflatex replaces references in the text with the assigned labels. The bibliography is automatically constructed.

4.5.1 Technologies Used

This section provides a list of technologies used for this component. The details for the technologies have already been provided in the Overview section.

4.5.2 Component Overview

This section can take the form of a list of features.

4.5.3 Phase Overview

This is an extension of the Phase Overview above, but specific to this component. It is meant to be basically a brief list with space for marking the phase status.

4.5.4 Architecture Diagram

It is important to build and maintain an architecture diagram. However, it may be that a component is best described visually with a data flow diagram.

4.5.5 Data Flow Diagram

It is important to build and maintain a data flow diagram. However, it may be that a component is best described visually with an architecture diagram.

4.5.6 Design Details

This is where the details are presented and may contain subsections. Here is an example code listing:

```
#include <stdio.h>
#define N 10
/* Block
 * comment */

int main()
{
    int i;

    // Line comment.
    puts("Hello world!");

    for (i = 0; i < N; i++)
    {
        puts("LaTeX is also great for programmers!");
    }

    return 0;
}
```

This code listing is not floating or automatically numbered. If you want auto-numbering, but it in the algorithm environment (not algorithmic however) shown above.

4.6 Major Component #2

4.6.1 Technologies Used

This section provides a list of technologies used for this component. The details for the technologies have already been provided in the Overview section.

4.6.2 Component Overview

This section can take the form of a list of features.

4.6.3 Phase Overview

This is an extension of the Phase Overview above, but specific to this component. It is meant to be basically a brief list with space for marking the phase status.

4.6.4 Architecture Diagram

It is important to build and maintain an architecture diagram. However, it may be that a component is best described visually with a data flow diagram.

4.6.5 Data Flow Diagram

It is important to build and maintain a data flow diagram. However, it may be that a component is best described visually with an architecture diagram.

4.6.6 Design Details

This is where the details are presented and may contain subsections.

4.7 Major Component #3

4.7.1 Technologies Used

This section provides a list of technologies used for this component. The details for the technologies have already been provided in the Overview section.

4.7.2 Component Overview

This section can take the form of a list of features.

4.7.3 Phase Overview

This is an extension of the Phase Overview above, but specific to this component. It is meant to be basically a brief list with space for marking the phase status.

4.7.4 Architecture Diagram

It is important to build and maintain an architecture diagram. However, it may be that a component is best described visually with a data flow diagram.

4.7.5 Data Flow Diagram

It is important to build and maintain a data flow diagram. However, it may be that a component is best described visually with an architecture diagram.

4.7.6 Design Details

This is where the details are presented and may contain subsections.

5

System and Unit Testing Design

This section describes the approach taken with regard to system and unit testing. This chapter does not describe the outcome of those tests. That will be described in the prototypes chapter.

5.1 Overview

Provides a brief overview of the testing approach, testing frameworks, and general how testing is/will be done to provide a measure of success for the system.

Each requirement (user story component) should be tested. A review of objectives and constraints might be needed here.

5.2 Dependencies

Describe the basic dependencies which should include unit testing frameworks and reference material.

5.3 Test design and setup

Describe how test cases were developed/designed, setup, and how they connect to the requirements. This section can be extremely involved if a complete list of test cases was warranted for the system. One approach is to list each requirement, module, or component and describe the test.

The unit test framework is described here.

5.4 System Testing

5.5 System Integration Analysis

5.6 Risk Analysis

5.6.1 Risk Mitigation

6

Sprint Results and Prototypes

This chapter is for recording the results of each sprint and documenting the evolving product. It is a historical record of what you accomplished in 464/465. This should be organized according to Sprints. It should have the basic description of the sprint deliverable and what was accomplished. Screen shots, photos, captures from video, etc should be used. Expect this to be a long chapter.

6.1 Sprint 0 Report

6.1.1 Sprint Backlog

- Define the Minimum Viable Product (MVP)
- Define the toolchain
- Identify use cases and create user stories
- Estimate a timeline for the development process

6.1.2 Deliverable

- Timeline (tentative)
  - 9/18:
    - Define user stories.
    - Define the MVP.
    - Define the toolchain.
    - Create tentative timeline.
  - 10/2:
    - Sample website setup with file upload to web root directory.
  - 10/16:
    - First client presentation documents.
    - Standalone file conversion.
  - 10/30:
    - File conversion on Azure.
    - Download original and converted file from Azure.
  - 11/13:
    - Generate unique AR Tags that map to each converted file on Azure.
  - 11/27:
    - User accounts and login on the Azure site.
  - 12/4 - 12/11:
    - File sharing permissions between users

6.1.3 Successes and Failures

6.1.3.a Successes

- Defined the MVP

Upload a Maplesoft 3D object file to a website. Have the website perform an automatic file conversion and present the user with an AR Tag. When the user views the AR Tag through an AR device, download and render the 3D object file.

- Decided to develop with the Microsoft Hololens being the primary supported AR device.

A conscious decision was made in accordance with current technologies and the Mobile Computing Grant awarded to the South Dakota School of Mines and Technology that this service aims to satisfy to keep development centered around the Hololens and other Microsoft supported or compatible tools and services.

- Decided to use Azure for cloud hosting services

The decision was between using Azure cloud services or Amazon Web Services. Azure was voted as the better of the two options on the grounds that the primary device we intend to make work using this service is the Microsoft Hololens and compatibility conflicts should be avoided.

- Created initial user stories
  - As a faculty member, I want to upload a maple file to a cloud server.
  - As a faculty member, I want the maple file to be automatically converted to an AR tag on the cloud server.
  - As a faculty member, I want to be able to download the AR tag for my document from the cloud server.
  - As a user of this product, I want to be able to view the AR tag through a Microsoft Hololens to render a 3D model.
- Estimated tentative development timeline. (See 6.1.2)

6.1.4 Sprint Review

6.1.5 Sprint Retrospective

6.1.6 Sprint Analytics

Place your burndown charts, team velocity information, etc here if they are not discussed above.

6.2 Sprint 1 Report

6.2.1 Sprint Backlog

6.2.2 Deliverable

6.2.3 Results of testing

6.2.4 Successes and Failures

6.2.5 Modifications required (product backlog, design, requirements, etc)

6.2.6 Sprint Review

6.2.7 Sprint Retrospective

6.2.8 Sprint Analytics

Place your burndown charts, team velocity information, etc here if they are not discussed above.

6.3 Sprint 2 Report

6.3.1 Sprint Backlog

6.3.2 Deliverable

6.3.3 Results of testing

6.3.4 Successes and Failures

6.3.5 Modifications required (product backlog, design, requirements, etc)

6.3.6 Sprint Review

6.3.7 Sprint Retrospective

6.3.8 Sprint Analytics

Place your burndown charts, team velocity information, etc here if they are not discussed above.

6.4 Sprint 3 Report

6.4.1 Sprint Backlog

6.4.2 Deliverable

6.4.3 Results of testing

6.4.4 Successes and Failures

6.4.5 Modifications required (product backlog, design, requirements, etc)

6.4.6 Sprint Review

6.4.7 Sprint Retrospective

6.4.8 Sprint Analytics

Place your burndown charts, team velocity information, etc here if they are not discussed above.

6.5 Sprint 4 Report

6.5.1 Sprint Backlog

6.5.2 Deliverable

6.5.3 Results of testing

6.5.4 Successes and Failures

6.5.5 Modifications required (product backlog, design, requirements, etc)

6.5.6 Sprint Review

6.5.7 Sprint Retrospective

6.5.8 Sprint Analytics

Place your burndown charts, team velocity information, etc here if they are not discussed above.

7

Release – Setup – Deployment

This section should contain any specific subsection regarding specifics in releasing, setup, and/or deployment of the system.

7.1 Deployment Information and Dependencies

Are there dependencies that are not embedded into the system install?

7.2 Setup Information

How is a setup/install built?

7.3 System Versioning Information

How is the system versioned?

8

User Documentation

This section should contain the basis for any end user documentation for the system. End user documentation would cover the basic steps for setup and use of the system. It is likely that the majority of this section would be present in its own document to be delivered to the end user. However, it is recommended the original is contained and maintained in this document.

8.1 User Guide

The source for the user guide can go here. You have some options for how to handle the user docs. If you have some `newpage` commands around the guide then you can just print out those pages. If a different formatting is required, then have the source in a separate file `userguide.tex` and include that file here. That file can also be included into a driver (like the senior design template) which has the client specified formatting. Again, this is a single source approach.

8.2 Installation Guide

8.3 Programmer Manual

9

Research Results

This chapter describes the results and conclusions of your research. This would be the final report for a research project.

9.1 Result 1

9.2 Result 2

9.3 Conclusions

9.4 Further work

Bibliography

[1] R. Arkin. *Governing Lethal Behavior in Autonomous Robots*. Taylor & Francis, 2009.

[2] Howie Choset, Kevin M. Lynch, Seth Hutchinson, George A Kantor, Wolfram Burgard, Lydia E. Kavraki, and Sebastian Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, Cambridge, MA, June 2005.

[3] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.

[4] V. Lumelsky and A. Stepanov. Path planning strategies for point mobile automation moving amidst unknown obstacles of arbirary shape. *Algorithmica*, pages 403–430, 1987.

[5] S.A. NOLFI and D.A. FLOREANO. *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. A Bradford book. A BRADFORD BOOK/THE MIT PRESS, 2000.

[6] Wikipedia. Asimo — Wikipedia, the free encyclopedia. `http://upload.wikimedia.org/wikipedia/commons/thumb/0/05/HONDA_ASIMO.jpg/450px-HONDA_ASIMO.jpg`, 2013. [Online; accessed June 23, 2013].

SDSMT SENIOR DESIGN
SOFTWARE DEVELOPMENT AGREEMENT

This Software Development Agreement (the "Agreement") is made between the SDSMT Computer Science

Senior Design Team _____,
("Student Group")

consisting of team members _____,
("Student Names")

and Sponsor _____,
("Company Name")

with address: _____.

[Note: Bracketed material is included to suggest content that will vary with each agreement. I STRONGLY SUGGEST THAT THE INSTRUCTOR LOOK AT THE COMPLETED AGREEMENT BEFORE YOU SIGN IT!!]

1 RECITALS

1. Sponsor desires Senior Design Team to develop software [for use in Sponsor's simulation platform for optical fiber transmissions of digitized video signals] (the "Field").
2. Senior Design Teams willing to develop such Software.

NOW, THEREFORE, in consideration of the mutual covenants and promises herein contained, the Team and Sponsor agree as follows:

2 EFFECTIVE DATE

This Agreement shall be effective as of _____ (the "Effective Date").

3 DEFINITIONS

1. "Software" shall mean [the computer programs in machine readable object code form and any subsequent error corrections or updates supplied to Sponsor by Senior Design Team pursuant to this Agreement.] [Depending on the particulars of each agreement, any or all of the following may need to be specified. If they are relevant, they should be used throughout, modifying the standard form as appropriate.]
2. "Acceptance Criteria" means the written technical and operational performance and functional criteria and documentation standards set out in the [project plan.]
3. "Acceptance Date" means [the date for each Milestone when all Deliverables included in that Milestone have been accepted by Sponsor in accordance with the Acceptance Criteria and this Agreement.]
4. "Deliverable" means a deliverable specified in the [project plan.]
5. "Delivery Date" shall mean, [with respect to a particular Milestone,] the date on which University has delivered to Sponsor all of the Deliverables [for that Milestone] in accordance with [the project plan and] this Agreement.

6. "Documentation" means the documents, manuals and written materials (including end-user manuals) referenced, indicated or described in [the project plan] or otherwise developed pursuant to this Agreement.
7. "Milestone" means the completion and delivery of all of the Deliverables or other events which are included or described in [the project plan] scheduled for delivery and/or completion on a given target date; a Milestone will not be considered completed until the Acceptance Date has occurred with respect to all of the Deliverables for that Milestone.

4 DEVELOPMENT OF SOFTWARE

1. Senior Design Team will use its best efforts to develop the Software described in [the project plan.] The Software development will be under the direction of or his/her successors as mutually agreed to by the parties ("Team Lead") and will be conducted by the Team Lead. The Team will deliver the Software to the satisfaction of the course instructor that reasonable effort has been made to address the needs of the client. The Team understands that failure to deliver the Software is grounds for failing the course.
2. Sponsor understands that the Senior Design course's mission is education and advancement of knowledge, and, consequently, the development of Software must further that mission. The Senior Design Course does not guarantee specific results or any results, and the Software will be developed only on a best efforts basis. The Software is considered PROOF OF CONCEPT only and is NOT intended for commercial, medical, mission critical or industrial applications.
3. The Senior Design instructor will act as mediator between Sponsor and Team; and resolve any conflicts that may arise.

5 COMPENSATION

[This is entirely subject to negotiation. Normally NO COMPENSATION occurs in a Senior Design Project. On occasion an intern status and wage is appropriate.]

6 CONSULTATION AND REPORTS

1. Sponsor's designated representative for consultation and communications with the Team Lead shall be

   _____ or such other person as Sponsor may from time to time designate to the Team Lead ("Designated Representative").
2. During the Term of the Agreement, Sponsor's representatives may consult informally with course instructor regarding the project, both personally and by telephone. Access to work carried on in University facilities, if any, in the course of this Agreement shall be entirely under the control of University personnel but shall be made available on a reasonable basis.
3. The Team Lead will submit written progress reports. At the conclusion of this Agreement, the Team Lead shall submit a comprehensive final report in the form of the formal course documentation at the conclusion of the Senior Design II course.

7 CONFIDENTIAL INFORMATION

1. The parties may wish, from time to time, in connection with work contemplated under this Agreement, to disclose confidential information to each other ("Confidential Information"). Each party will use reasonable efforts to prevent the disclosure of any of the other party's Confidential Information to third parties for

a period of three (3) years after the termination of this Agreement, provided that the recipient party's obligation shall not apply to information that:

(a) is not disclosed in writing or reduced to writing and so marked with an appropriate confidentiality legend within thirty (30) days of disclosure;

(b) is already in the recipient party's possession at the time of disclosure thereof;

(c) is or later becomes part of the public domain through no fault of the recipient party;

(d) is received from a third party having no obligations of confidentiality to the disclosing party;

(e) is independently developed by the recipient party; or

(f) is required by law or regulation to be disclosed.

2. In the event that information is required to be disclosed pursuant to subsection (6), the party required to make disclosure shall notify the other to allow that party to assert whatever exclusions or exemptions may be available to it under such law or regulation.

8 INTELLECTUAL PROPERTY RIGHTS

[Negotiated on a case-by-case basis. This must address who owns the algorithms and who owns the source code. For example: All deliverables become property of the Sponsor. Roughly: If the idea originates with the sponsor, or if a sponsor pays you to develop an idea, then they have legitimate claim to the IP. If the idea originates from the University (through faculty or staff) then the University has legitimate claim. If the idea is yours (student) and you develop it without external compensation then you have legitimate claim.]

9 WARRANTIES

The Senior Design Team represents and warrants to Sponsor that:

1. the Software is the original work of the Senior Design Team in each and all aspects;
2. the Software and its use do not infringe any copyright or trade secret rights of any third party.

No agreements will be made beyond items (1) and (2).

10 INDEMNITY

1. Sponsor is responsible for claims and damages, losses or expenses held against the Sponsor. [Sponsor may have something to add here.]
2. Sponsor shall indemnify and hold harmless the Senior Design Team, its affiliated companies and the officers, agents, directors and employees of the same from any and all claims and damages, losses or expenses, including attorney's fees, caused by any negligent act of Sponsor or any of Sponsor's agents, employees, subcontractors, or suppliers.
3. NEITHER PARTY TO THIS AGREEMENT NOR THEIR AFFILIATED COMPANIES, NOR THE OFFICERS, AGENTS, STUDENTS AND EMPLOYEES OF ANY OF THE FOREGOING, SHALL BE LIABLE TO ANY OTHER PARTY HERETO IN ANY ACTION OR CLAIM FOR CONSEQUENTIAL OR SPECIAL DAMAGES, LOSS OF PROFITS, LOSS OF OPPORTUNITY, LOSS OF PRODUCT OR LOSS OF USE, WHETHER THE ACTION IN WHICH RECOVERY OF DAMAGES IS SOUGHT IS BASED ON CONTRACT TORT (INCLUDING SOLE, CONCURRENT OR OTHER NEGLIGENCE AND STRICT

LIABILITY), STATUTE OR OTHERWISE. TO THE EXTENT PERMITTED BY LAW, ANY STATUTORY REMEDIES WHICH ARE INCONSISTENT WITH THE PROVISIONS OF THESE TERMS ARE WAIVED.

11 INDEPENDENT CONTRACTOR

For the purposes of this Agreement and all services to be provided hereunder, the parties shall be, and shall be deemed to be, independent contractors and not agents or employees of the other party. Neither party shall have authority to make any statements, representations or commitments of any kind, or to take any action which shall be binding on the other party, except as may be expressly provided for herein or authorized in writing.

12 TERM AND TERMINATION

1. This Agreement shall commence on the Effective Date and extend until the end of classes of the second semester of Senior Design (CSC 467), unless sooner terminated in accordance with the provisions of this Section ("Term").
2. This Agreement may be terminated by the written agreement of both parties.
3. In the event that either party shall be in default of its materials obligations under this Agreement and shall fail to remedy such default within thirty (30) days after receipt of written notice thereof, this Agreement shall terminate upon expiration of the thirty (30) day period.
4. Any provisions of this Agreement which by their nature extend beyond termination shall survive such termination.

13 ATTACHMENTS

Attachments A and B are incorporated and made a part of this Agreement for all purposes.

14 GENERAL

1. This Agreement constitutes the entire and only agreement between the parties relating to the Senior Design Course, and all prior negotiations, representations, agreements and understandings are superseded hereby. No agreements altering or supplementing the terms hereof may be made except by means of a written document signed by the duly authorized representatives of the parties.
2. This Agreement shall be governed by, construed, and enforced in accordance with the internal laws of the State of South Dakota.

15 SIGNATURES

Replace with name of student #1	Date
Replace with name of student #2	Date
Replace with name of student #3	Date
Replace with name of sponsor's representative	Date

A

Product Description

Write a description of the product to be developed. Use sectioning commands as neccessary.

NOTE: *This is part of the contract.*

B

Class Index

1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

C

Class Documentation

1 Poly Class Reference

Public Member Functions

- Poly ()
- ∼Poly ()
- int myfunction (int)

1.1 Constructor & Destructor Documentation

1.1.a Poly::Poly ( )

My constructor

1.1.b Poly::∼Poly ( )

My destructor

1.2 Member Function Documentation

1.2.a int Poly::myfunction (int *a*)

my own example function fancy new function

new variable

The documentation for this class was generated from the following file:

- hello.cpp

D

Business Plan

Use a plan outline that works for your venture!

1 Business Model

2 Market and Competition

3 Regulatory environment

4 Intellectual Property and Freedom to Operate

5 Management Team and Advisors

6 Sources and Uses of Capital

7 Financial Statements

8 Metrics and Milestones

9 Exit Plan

E

Experimental Log

For research projects one needs to keep a log of all research/lab activities.

10/15/15 Ran modified filter on data sets 1 - 6. Results were ...

10/17/15 Changed tolerance on sensor and collected data. These ...

F

Publications

Research Track: This chapter will include any publications generated from the research. Most likely these will be preprints and one will just include the pdf.

G

Acknowledgment

Thanks

H

Supporting Materials

This document will contain several appendices used as a way to separate out major component details, logic details, or tables of information. Use of this structure will help keep the document clean, readable, and organized.

LaTeX Example

LaTeX sample file: Remove from submitted materials

1 Introduction

This is a sample input file. Comparing it with the output it generates can show you how to produce a simple document of your own.

2 Ordinary Text

The ends of words and sentences are marked by spaces. It doesn't matter how many spaces you type; one is as good as 100. The end of a line counts as a space.

One or more blank lines denote the end of a paragraph.

Since any number of consecutive spaces are treated like a single one, the formatting of the input file makes no difference to TeX, but it makes a difference to you. When you use LaTeX, making your input file as easy to read as possible will be a great help as you write your document and when you change it. This sample file shows how you can add comments to your own input file.

Because printing is different from typewriting, there are a number of things that you have to do differently when preparing an input file than if you were just typing the document directly. Quotation marks like "this" have to be handled specially, as do quotes within quotes: " 'this' is what I just wrote, not 'that' ".

Dashes come in three sizes: an intra-word dash, a medium dash for number ranges like 1–2, and a punctuation dash—like this.

A sentence-ending space should be larger than the space between words within a sentence. You sometimes have to type special commands in conjunction with punctuation characters to get this right, as in the following sentence. Gnats, gnus, etc. all begin with G. You should check the spaces after periods when reading your output to make sure you haven't forgotten any special cases. Generating an ellipsis ... with the right spacing around the periods requires a special command.

TeX interprets some common characters as commands, so you must type special commands to generate them. These characters include the following: $ & % # { and }.

In printing, text is emphasized by using an *italic* type style.

A long segment of text can also be emphasized in this way. Text within such a segment given additional emphasis with Roman *type. Italic type loses its ability to emphasize and become simply distracting when used excessively.*

It is sometimes necessary to prevent TeX from breaking a line where it might otherwise do so. This may be at a space, as between the "Mr." and "Jones" in "Mr. Jones", or within a word—especially when the word is a symbol like *itemnum* that makes little sense when hyphenated across lines.

Footnotes[1] pose no problem.

TeX is good at typesetting mathematical formulas like $x - 3y = 7$ or $a_1 > x^{2n}/y^{2n} > x'$. Remember that a letter like x is a formula when it denotes a mathematical symbol, and should be treated as one.

[1] This is an example of a footnote.

3 Displayed Text

Text is displayed by indenting it from the left margin. Quotations are commonly displayed. There are short quotations

> This is a short a quotation. It consists of a single paragraph of text. There is no paragraph indentation.

and longer ones.

> This is a longer quotation. It consists of two paragraphs of text. The beginning of each paragraph is indicated by an extra indentation.
>
> This is the second paragraph of the quotation. It is just as dull as the first paragraph.

Another frequently-displayed structure is a list. The following is an example of an *itemized* list.

- This is the first item of an itemized list. Each item in the list is marked with a "tick". The document style determines what kind of tick mark is used.
- This is the second item of the list. It contains another list nested inside it. The inner list is an *enumerated* list.
  1. This is the first item of an enumerated list that is nested within the itemized list.
  2. This is the second item of the inner list. LaTeX allows you to nest lists deeper than you really should.

  This is the rest of the second item of the outer list. It is no more interesting than any other part of the item.
- This is the third item of the list.

You can even display poetry.

> There is an environment for verse
> Whose features some poets will curse.
>
> For instead of making
> Them do *all* line breaking,
> It allows them to put too many words on a line when they'd rather be forced to be terse.

Mathematical formulas may also be displayed. A displayed formula is one-line long; multi-line formulas require special formatting instructions.

$$x' + y^2 = z_i^2$$

Don't start a paragraph with a displayed equation, nor make one a paragraph by itself.

4 Build process

To build LaTeX documents you need the latex program. It is free and available on all operating systems. Download and install. Many of us use the TexLive distribution and are very happy with it. You can use a editor and command line or use an IDE. To build this document via command line:

```
alta>  pdflatex SystemTemplate
```

If you change the bib entries, then you need to update the bib files:

```
alta>  pdflatex SystemTemplate
alta>  bibtex SystemTemplate
alta>  pdflatex SystemTemplate
alta>  pdflatex SystemTemplate
```

The template files provided also contain a Makefile, which will make things much easier.

Acknowledgment

Thanks to Leslie Lamport.