LEaD Design:
Team Portfolio

Adrian Beehner
Budget Director

Andrew Butler
Documenter

Kevin Dorscher
Client Liaison

Paul Martin
Designer

CS 480/481: Fall 2017 - Spring 2018
Senior Capstone Design Project
UI CS - Wireless Tower of Lights

University of Idaho
College of Engineering

Contents

1 Introduction

1.1 Project Summary

The University of Idaho has, for several years, done various projects involving the Tower of Lights Show and equipping the marching band with light-up glasses. The current "TowerLights" product involves LED-based light bars that are placed in front of front-facing widows of a large buildling (Theophilus Tower) and are then illuminated to play animations alongside/synchronously with music. The goal is to enhance the current "TowerLights" product. The current implementation of this product uses the ethernet wiring system in the building to control the LEDs. The goal of the project described in this document is to convert this part of the system to a wireless operation. This in turn requires the development of a wireless module that would be attached to each of the light bars. Thus this module has to sleep and wake up, as well as respond to wireless signals from a computer, and since it's wireless, these modules will need to be battery powered. Battery power must also be conserved by staying in the sleep state until needed. The purpose of this enhancement is to provide a certain level of portability to have "TowerLights" at other locations.
The product will give the user the ability to run a program that reads in .tan files and .wav files, have this program communicate with a XBee Wireless module on an Arduino that is attached to a computer via USB, then communicate wirelessly with each Arduino receiver, that is battery powered. Each of these Arduino receivers are attached to an LED board, that will then communicate with each LED on that board through wired communication from the Arduino (same one that holds the receiver) to the LEDs. The program that broadcasts the shows will be available for OSX, Windows, and Linux based operating systems.
This documentation lives at https://github.com/YupHio/LEaD_Design/tree/master/Doc/TeamPortfolio_LEaD_Design.tex
The code for the project can be found at https://github.com/YupHio/LEaD_Design/tree/master/Code

1.2 Document Purpose

This document is a team portfolio for the Fall 2017-Spring 2018 CS 480/481: Senior Capstone Design project at the University of Idaho. The purpose of this document is to outline the methodology, design, and keep a record of this project. It defines terms used, outlines the scope of the project, details specific design choices, meeting minutes, project learning, design goals, specification and constraints, system diagrams, analysis of alternatives, engineering modeling, manufacturing/assembly plan, experimental design, data analysis, balance sheet, and other items.

1.3 Definition of Terms

- **Arduino** - open source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world
(https://en.wikipedia.org/wiki/Arduino)
- **Arduino Shield** - Shields are boards that can be plugged on top of the Arduino PCB extending its capabilities. The different shields follow the same philosophy as the original toolkit: they are easy to mount, and cheap to produce.
(https://www.arduino.cc/en/Main/ArduinoShields)
- **Xbee** - The Arduino Xbee shield allows multiple Arduino boards to communicate wirelessly over distances up to 100 feet (indoors) or 300 feet (outdoors) using the Maxstream Xbee Zigbee module.
(https://www.arduino.cc/en/Main/ArduinoShields)

1.3.1 Arduino IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them. https://www.arduino.cc/en/Main/Software

1.3.2 Pulse

PulseAudio is a sound system for POSIX OSes, meaning that it is a proxy for your sound applications. It allows you to do advanced operations on your sound data as it passes between your application and your hardware. Things like transferring the audio to a different machine, changing the sample format or channel count and mixing several sounds into one are easily achieved using a sound server. https://www.freedesktop.org/wiki/Software/PulseAudio/l

2 Meeting Minutes

Weekly action items and summaries of progress made are detailed below. Furthermore, subsections discuss what was helpful and what was not during these meetings. Discussion of attendance and participation, as well as contribution and discussion topics are discussed below.

2.0.1 9/08/2017 Meeting Notes - With Dr.Rinker

Meeting started at 3:31, all members present, Adrian on Zoom meeting

Question and Answer with Rinker:

General schedule: Rinker's in CDA start of the week, always in Moscow on Friday, in between depends on events.

Current system in the tower: 3 high powered LEDs (in series) in each room facing the proper direction, controlled over CAT cable from the basement. LEDs prefer constant current over constant voltage, using constant current power source. Each color takes 270 mili-amps. Constant current circuit used here.

Future objectives: Convert system to be wireless. Nodes will need to sleep for a couple days before the show begins, using low power, and should then be remotely wake-able. Power and LED configurations are up to us.

Current goofy lights: broadcast from laptop to Arduino like board, transmits out to the glasses. Wireless protocol related to Zigby. Not wifi, but 802.15.4 (ad-hoc sensor network). Devices can sleep, wakeup, reconnect to network, etc. Zigby handles errors when reconnecting, etc. We avoid using Zigby as were broadcasting in real time, and do not want the error handling. Broadcasts on 2.4ghz, regular wifi frequency. 9v lith-ion batteries are being used in the glasses. LEDs in the glasses are in series. Uses a resistor to deliver the correct voltage. Uses the chip from an Arduino, straight up programmed from the Arduino IDE. Atmega328p. Broadcasts to all glasses i.e. DMX. Uses 16 different channels for groups of lights.

802.15.4 only goes at 250kbps. Might want to reduce each channel to 2 bytes instead of 3?

DMX protocol: Used in theater lights, wired protocol, goes through each light sequentially.

Current code is all available for use, we're going to get that from Rinker and put on GitHub(?)

Mouser.com parts. Superbrightleds.com

Meeting adjourned at 4:39

2.0.2 9/14/2017 Meeting Notes

Meeting started at 3:30, Adrian lost Internet almost immediately (bad ISP). This Team meeting had the team generally discuss the schedule of the project for the entire year. This meeting was very important in outlining the goals and deadlines needed on a monthly basis. The figures for these schedules is shown below. Meeting ended at 4:15

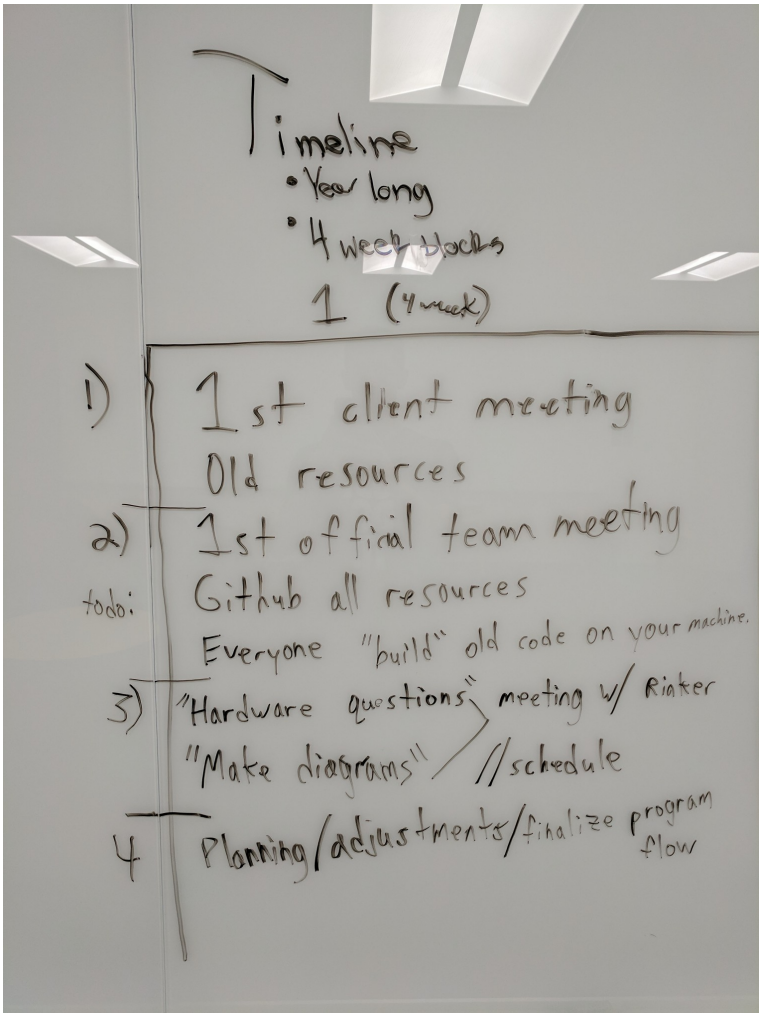

Figure 1: 9/14 Meeting Project Schedule

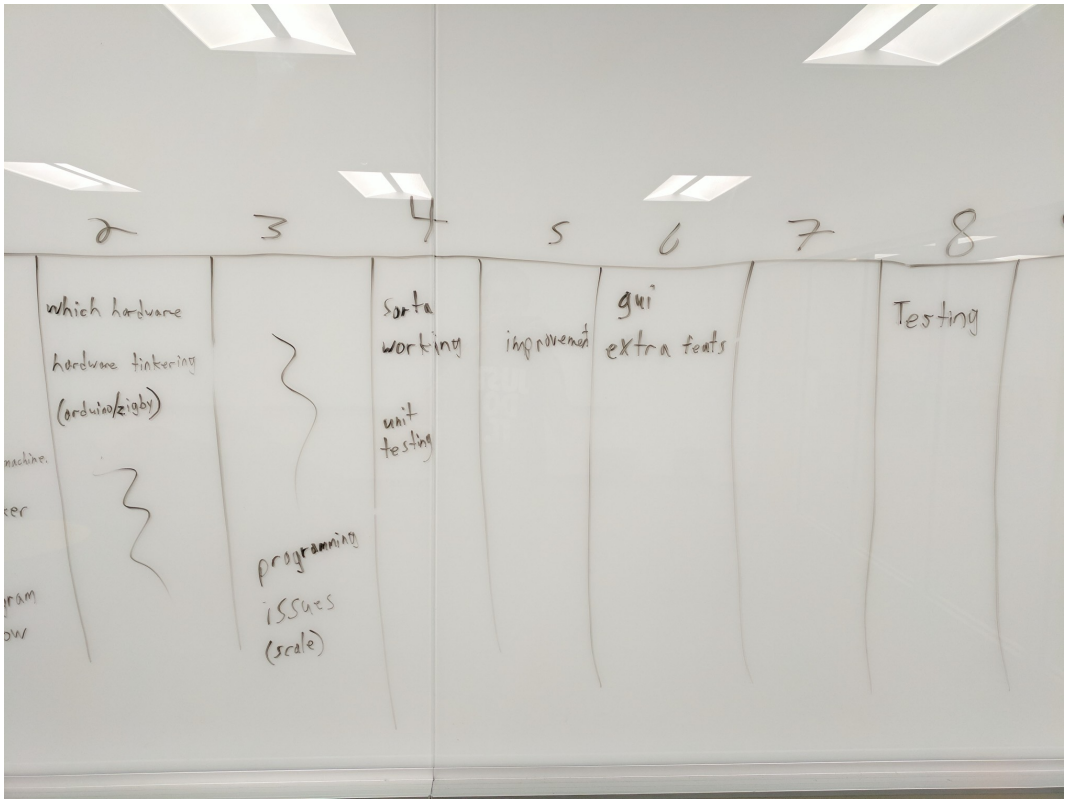

Figure 2: 9/14 Meeting Project Schedule

2.0.3 9/22/2017 Meeting Notes - With Dr.Rinker

All members present, Adrian via Zoom, meeting started at 3:32

We only need to deal with .tan file to hardware, theres another group redesigning the .tan file creator. Theyre finishing up in December. Also may be redesigning the interface for the player?

Current implementation uses xbee to transmit to receiver, then transmits to the light controller serially.

We can probably use old CSAC space to store hardware, work. This has a soldering station too, along with some goofy glasses and the old tower hardware being stored.

328p chips are super cheap, could definitely use one of those for each light bar.

Arduino IDE supports turning off bootloader now, etc, which should make development even easier.

Current player is Linux specific, supposedly has Mac and Windows equivalent libraries though. Pulseaudio and FTDI. Look into making this cross platform compatible.

Main thread of player sends wav bytes to the audio thread, updates lights once the program reaches the proper time.

Parts needed: transmitter, shield, USB to serial, receiver chips, light bars themselves, batteries.

Meeting adjourned at 4:33

3 Project Learning

Technologies used to solve problems are described below. Further discussion of these technologies are left in each section's subsections.

4 Design Goals

Client needs and project goals are discussed below. A Timeline for these is also included. Discussion of revision of goals, and addition of any new goals is also discussed below.

4.1 Client Needs

The needs of the Client (University of Idaho) are as follows:

- LED Light Bars
- Microprocessor communication (Arduino)
- LEDs bright enough to be a coherent display, visible from a distance in the dark
- Wireless Protocol SPI
- Battery powered
- Receiver Module for Arduino (802.15.14 chip)
- AdrProcessor for designing chip
- Low power mode (sleep mode)
- Wake up remotely
- 1-bit for each color for each window
- 802.15.4 protocol, channels 3 bytes (1 for each color, RGB)
- Avoid wifi (we don't want to have interference)
- Design module
- Expand channels (for expanding bandwidth)
- 15-20 stories, need to support enough windows
- WAV file support
- OSX, Windows, and Linux support (Cross-Platform)
- .tan file support -

4.2 Project Goal

The goal of the project is to the extend the versatility of the Tower Of Lights project, which at the moment, gives the user the ability to run a program which reads in a .tan file (animation files for the lights) and .wav files. Then this program communicates with a Arduino via Ethernet. Now, the Arduino communicates to each of the LEDs, and tells them which

color and brightness to be, from the .tan file (thus it basically reads in animation info). The enhancement of the project involves providing cross-platform support, which means having to rework some of the TowerPlayer code so it doesn't use the Pulse library (which is Linux-specific). Also, the enhancement requires making the wired connection to the Arduinos on the LED bar to wireless, this is accomplished by having an Arduino Receiver on each LED Board that receives info sent out from the Arduino connected to the main computer running the program, that Arduino has a XBee Shield attached, which is a wireless module to transmit the info to each Arduino on a board. The Arduino now requires a portable power supply, which needs to be a 9V battery for each Arduino on an LED Board. The final enhancement is that since the LED Boards are running off battery, they require some kind of sleep mode, where they will still be able to receive info (so they can wake up).

The product will give the user the ability to run a program that reads in .tan files and .wav files, have this program communicate with a XBee Wireless module on an Arduino that is attached to a Computer via USB, then communicate wirelessly with each battery powered Arduino receiver, on each LED board, that will then communicate with each LED on that board through wired communication from the Arduino (same one that holds the receiver)to the LEDs. The program that runs through this procedure will be available for the OSX, Windows, and Linux based operating systems.

4.3 Timeline

This was the timeline for UIdaho's Fall 2017 - Spring 2018 CS 480/481: Senior Capstone Design class.

- September • Planning/Adjustments/Finalize Program Flow 1
- October • Hardware Decision/Hardware Tinkering (Arduino/Xbee)
- November • Programming Issues (Scale)
- December • Prototype Product/Unit Testing
- January • Product Improvement/Evaluation
- February • GUI Implementation/Extra Features 2
- March • TBD
- April • Testing
- May • Ship/Manufacture (Deliver product)

5 Specifications and Constraints

Discussion of client interviews, pictures, measurements, etc. are provided below. Design specifications and constraints are also presented. Reasoning for any constraints is also mentioned.

6 System Diagrams

Discussion of symbols used, the diagrams themselves, and the software used for the diagrams is discussed below.

6.0.1 Current Product

The current product flow in regards to the final product is shown below in Figure below. The current setup does not have any battery setup, and requires a wired connection. Changing this is the core of this project, which will improve the versatility of the TowerOfLights product.

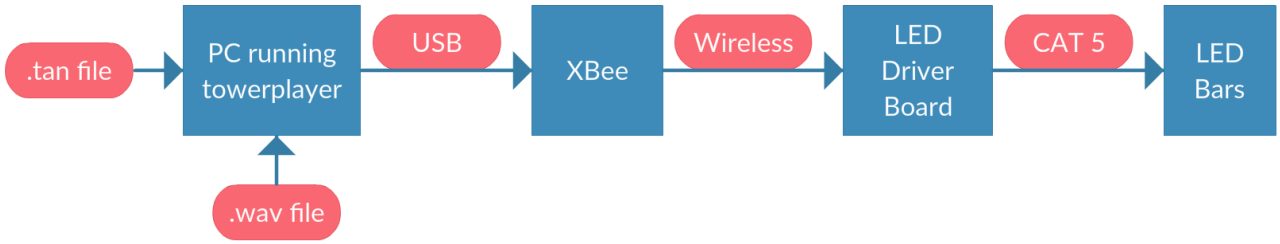

Figure 3: Current Product Flow

6.0.2 Desired Product

The desired product flow is shown in the figure below. The main focus is on the battery that should power each Arduino reciever, as well as the SPI protocol from XBee to the Receiver. This is to make the process wireless instead of wired, which is the main goal of this endeavor.

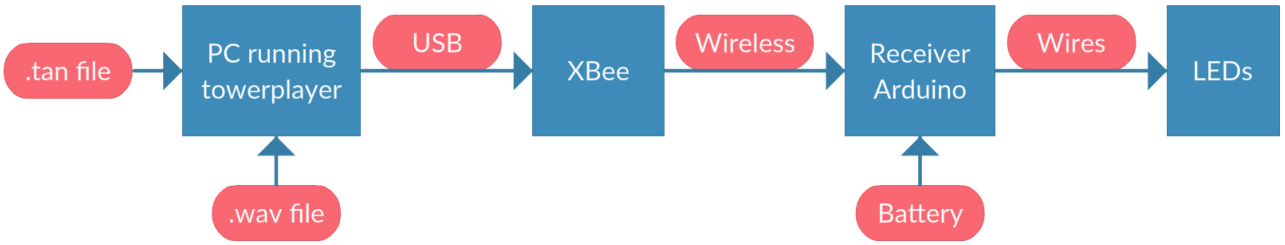

Figure 4: Desired Product Flow

6.0.3 PC Running TowerPlayer

The diagram for a flow chart depicting the sequence of actions for running the TowerPlayer program on a computer is shown in the figure below. This diagram helps with understanding

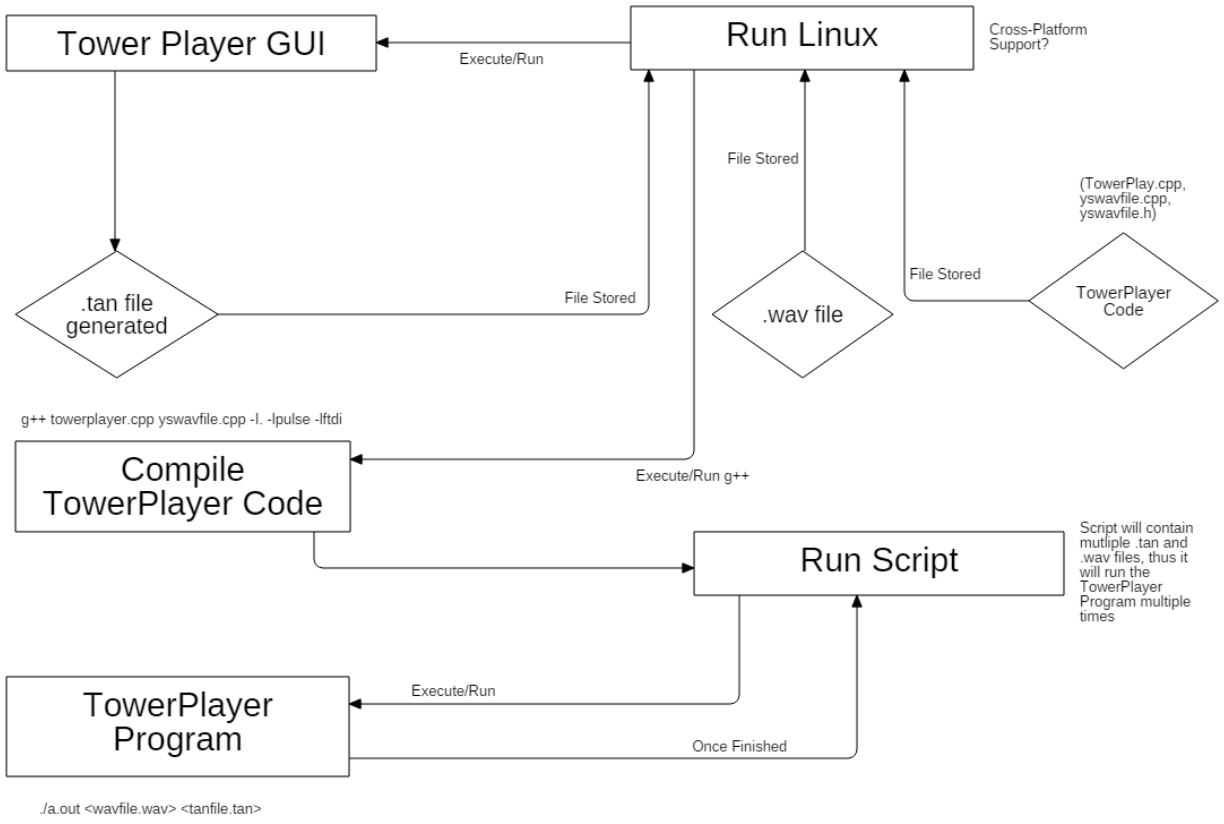

Figure 5: Flow Chart Diagram for PC Running TowerPlayer

the underlying software that needs to be setup and used before the hardware can successfully work together.

7 Analysis of Alternatives

Discussion of possible alternatives and why some alternatives are better is described below. These topics include: safety, moving parts, cost, durability, compatibility, and reliability.

8 Engineering Model

Discussion of the physical, chemical, and biological system modeling. Also discusses modeling criteria, expected accuracy, and pitfalls. Section of modeling software used is present, as well as data needed and how the data was obtained. Lastly a validation scheme for the model is shown.

9 Manufacturing/Assembly Plan

Discussion of the fabrication need, a flowchart of process oriented projects, a bill of materials, and the estimated manufacturer and delivery time is discussed below.

10 Experimental Design

The characterization of the purpose of the experiment, model validation, data gaps, and performance measurement are discussed below. Also the details on documentation, instrumentation, and measurements are also described.

11 Data Analysis

Documentation on statistical tools used, accuracy of data, and experiments shown below. Discussion on confidence is results also discussed below.

12 Balance Sheet

Discussion on initial budget, estimated cost for materials, components, labor, and spending plan are all described below.

13 Other Items

File management, archiving, documenting any issues, reports of accidents/incidents/near misses/precautions are described below.

13.1 LEaD Design Team Contract

The team contract for the *LEaD Design* is shown below. The contract discusses the various professional approaches the team will be held accountable to act towards during the time spent on the project. The contract is an important document, as it dicusses the various inner workings of how the team will work on assignments, resolve conflicts, manage work, make decisions and so on. The contract is four pages total.

LEaD Design Team Contract

Section 1: Team Name and Mission
Team Name: LEaD_Design
Mission of Team: To provide a quality product of portability and functionality that meets and/or exceeds the expectations of the client. Also to provide a professional level of conduct and documentation for swift and successful communication between all parties involved.

Section 2: Membership
Team Members: Adrian Beehner, Andrew Butler, Kevin Dorscher, Paul Martin
Consultants/Mentors/Instructional Staff: Dr. Robert Rinker

Section 3: Roles and Responsibilities
Budget: Adrian Beehner
Primary Client Contact: Kevin Dorscher
Organize Team Meetings: Paul Martin
Team Documentation: Andrew Butler

Additional Roles: Scheduling, Project Management, Online Repository Management, Communication Management, Designing (Goals, Specification, Experimental), Prototyping, Testing, Researching, Diagraming, Analyzing (Data, Alternatives), Modeling, Manufacturing/Assembly

Roles Will Be Selected/Assigned By: Team consensus, with evaluating the strengths and weaknesses of teammates and accordingly assigning roles based on these. Discussion/volunteering for responsibilities will be the primary method, if this proves inefficient, a variation of team voting will be required. Some roles will not be individual responsibilities however, but instead a collaborative effort that requires the professional coordination and responsibility of the entire team.

What are the key responsibilities associated with each role?

Budget: Initial Budget, estimating cost for materials/components/labor/etc., and create Spending Plan

Primary Contact Client: Professional Communication with Client, Schedule Times to meet with Client, Discuss/Report Progress

Organize Team Meetings: Communication with Team Members about meeting times, Meeting Agenda, Primary Figure in Meeting

Team Documentation: Record and Document Essential Meetings, Events, etc., Organize Documentation, Monitor teammate documentation

Scheduling: Create weekly and monthly schedules, plan goals, due dates, meetings, etc., factor in all teammates when scheduling

Project Management: Evaluate and monitor team progress, tasks, etc., rectify any project issues or roadblocks

Online Repository Management: Set up/monitor online repository, keep backups, correct any mistakes in repository

Communication Management: Set up/monitor team communications (email, Discord, etc.), promote professional team communication

Designing: Design with Client's needs goals, specifications, and constraints in mind, promote and

justify experimental design, set new goals

Prototyping: Simulate beginning aspects of design, provide only basic client needs and specifications, evaluate what aspects to disregard

Testing: Run tests to generate quantitative data about product, provide objective view of product in current state, create extensive tests for possible bugs in product

Researching: Find current technologies to use, determine how the technologies work, correct implementation of them, evaluating design, determine needed changes

Diagraming: Use standard symbols, properly labeling and referencing, documenting which member drew what and software used to draw the diagram

Analyzing: Determine possible alternatives in design, conceptualize testing data, document statistical tools used, document accuracy of data, determine confidence in results

Modeling: System Modeling, document modeling criteria, expected accuracy, and pitfalls, document modeling software used, document required data and where it was acquired, provided validation scheme

Manufacturing/Assembly: Provide fabrication, create flowcharts for process oriented items, Provide bill of materials, drawing, manufacturer and estimated delivery time

Section 4: Team Relationships
Relationships among team members must support full and respectful engagement of all members for the benefit of the entire team.

Members Will: Promote professional and respectful relationships among each other. Each member will have a positive and respectful attitude towards one another. Every member will have a voice and respected opinion in the team, and other teammates will listen in a respectful manner. Members will openly listen to any concerns a teammate has in regards to the project and promote solutions.

Members Will NOT: Create unprofessional and disrespectful relationships among each other. Each member will not have a negative and confrontational attitude towards one another. Members will not be isolated and have their opinion ignored, and teammates will not ignore opinions. Members will not shutdown any concerns a teammate has in regards to the project and forsake the concern.

Section 5: Joint Work
5a. **Purposes of Joint Work**
Team members will work together to establish collective goals and to produce decisions and work products that advance teamwork and project success.

Members Will: Work together in a professional collaborative environment, to help promote team decisions and products, while creating positive teamwork and successful progress in the project. Members will communicate and complete joint work in a timely and professional fashion. Team members will include all members in decisions and actions regarding the project, such as deciding goals, schedules, events and so on. Each team member will ensure that their collaborative work is up to the standards of the team.

Members Will NOT: Work individually in an unprofessional environment with no collaboration, to dismiss team decisions and products, while ignoring teamwork and progress in the project. Members will not withhold and ignore joint work in a tardy and unprofessional manner. Team members will not exclude members in decisions and actions regarding the project, such as deciding goals schedules, events, and so on. Each team member will not neglect to check that their collaborative work is up to standards of the team.

5b. **Team Meetings**
Team meetings are an important example of working together.

* *Where and When for Team Meetings:* Large Study Room, in Library at 3:30pm, on Thursdays

* *Components Required in Team Meeting Agenda:* Current progress in project, current due dates, weekly schedule, topics to be discussed, upcoming due dates, progress to be made by next meeting

* *How Meeting Minutes Be Taken/Circulated:* Documented by teammate in charge of team documentation, when meeting transcript is written minutes will be calculated as well.

Section 6: Individual Work
Team members are expected to work alone in many cases to complete work important to the team.

* *How Will Work Assignments Be Made:* Decided in team meeting when planning the schedule for the week/month, team will discuss/decide/designate what assignments will be suitable for each member.

* *How Will Quality Expectations Be Established and Verified:* Everything a team member does will be pushed onto the team's GitHub repository, where all other members will have access to view and evaluate work. Also, during meetings, team will go over last week's progress, which will involve evaluating each member's individual work and determining its quality.

* *How Will Due Dates Be Established and Verified:* During team meetings, the weekly schedule will verify due dates. Since each team meeting discusses progress made previously, due dates for weekly work assignments will be every Thursday before 3:30 pm.

* *How Will Status of Work in Progress Be Communicated:* Team members will communicate status of work in progress through the current pushed work on the GitHub repository, as well as communicating any updates about individual work through the Discord server chat. Team members will be sure to provide meaningful commit messages.

Section 7: Documentation and Communication
The team must maintain timely and accurate documentation of its individual and collective achievements, while also communicating needed information to one another and key project stakeholders.

* *Individual Documentation to Be Kept:* Diagrams, research, reports, analyses, sketches, ideas, design processes, math calculations, data and results

* *Team Documentation to Be Kept:* Transcripts for meetings, schedules, team portfolio, client meetings transcript, events documentation

* *Process for Communicating with Other Team Members:* Using team members' email or utilizing the Discord "LEaD Designers" server for communication between one or more members. Each team member must make sure that their means of communication and effective and that their message is clear and concise, for quick and simple understanding

* *Process of Communicating with Your Client/Outside Stakeholders:* The designated Primary Contact Client will be responsible for contacting and communicating with the client/outside stakeholders. Thus is a team member needs to contact the client, they must notify and discuss it with the teammate responsible for being the Primary Contact Client

Section 8: Conflict Resolution
The team will strive to resolve conflicts quickly and to the satisfaction and benefit of everyone involved.

* *Process Will Be Used to Address Conflicts:* Each team member is responsible for bring up conflicts in

weekly team meetings, or on the Discord Server, if needing to be resolved quickly. All team members will discuss the conflict, and decide upon a reasonable conclusion, based on everyone's opinions. If team members cannot resolve the conflict, the instructor/mentor should be notified to help progress towards conflict resolution.

* *Team Dynamics Be Communicated to Instructors/Mentors:* All team members are responsible for coordinating information together to be presented to the instructor/mentor, on a monthly basis. Include information such as progress of project, current deadlines, team performance, behavior, and so on. Communicating to the instructor/mentor any shortcomings and issues on the project will also be an essential asset.

Section 9: Amendments

* *This Team Contract Will Be Kept: Online in the team's GitHub Repository:*
  https://github.com/YupHio/LEaD_Design
* *Contract Will Be Reviewed:* On a Monthly Basis.

* *Contract Will Be Amended:* If teammate or team as a whole desire to amend the contract, the whole team must be notified, such as a group message, and then the teammate or will be amending the contact will pull the current contract from the online repository, make the needed changes, and push the change to the repository with an appropriate commit message.

Section 10: Affirmation of Compliance

We, the members of this team, affirm that we have established this contract with input and consensus of all members. By our signatures, we commit to compliance with the contract for the benefit of all members and the team as a whole.

Name	Signature	Date
Adrian Beehner	AB	9/21/17
Kevin Dorscher	KD	9/21/17
Paul Martin	PM	9/21/17
Andrew Butler	AB	9/21/17