Department of Computer Science & Engineering
The University of Texas at Arlington

Project Charter
CSE 4316: Senior Design I
Spring 2019

The EyeT Guys
The Eye Tracker

Jonathan El-Khoury
George Hinkel
Jay Denton
Dharampreet Gill
Paul Wafula

Revision History

Revision	Date	Author(s)	Description
0.1	02.22.2019	JE	document creation
0.2	02.24.2019	JE, GH	complete draft
0.3	05.02.2019	JD	minor errors fixed

Contents

List of Figures

1 Vision

Our vision is to create a new eye tracking system for Marci to replace her current system which no longer has continued support from the original manufacturer and is dangerously close to failure. By creating this system, it will allow Marci to continue to be able to communicate with her family even in the event of failure of her current system. However, if her current system continues to function then she will continue to use that and keep our system as an emergency backup.

2 Mission

In order to realize our vision, we will construct a new eye tracking system. This will involve using an embedded computer development board (Nvidia TX2) and attaching two remote cameras to it. Our initial plan is to utilize and retrofit two PlayStation Eye cameras. Each camera will be responsible for tracking one of Marci's eyes thus giving Marci the ability to communicate with her family and others. Each camera will be outfitted with infrared lights as well as infrared filters (which will be used to block out all visible light). Upon completion, this system will then be mounted to Marci's current battery-powered wheelchair.

3 Success Criteria

Upon completion of the prototype eye tracking system, we expect the following success indicators to be observed:

- In the event of a catastrophic failure of Marci's current system, our system will be available for immediate use.
- 15 minutes of 100% eye tracking reliability before the need for re-calibration.
- Basic functionality is implemented. Marci will be able to type and communicate with her family and others.

Within 6 months after the prototype delivery date, we expect the following success indicators to be observed:

- In the event of a catastrophic failure of Marci's current system, our system will be available for immediate use.
- Implementation of additional functional requirements to more closely mirror her current system.
- An additional 5% increase in eye tracking reliability before the need for re-calibration.

Within 12 months after the prototype delivery date, we expect the following success indicators to be observed:

- In the event of a catastrophic failure of Marci's current system, our system will be available for immediate use.
- Implementation of all of functional requirements matching what her current system already does.
- An additional 15% increase in eye tracking reliability before the need for re-calibration.

4 BACKGROUND

Imagine being trapped in your own body, unable to move a muscle or even talk, all you can do is look around... this may remind some of you of sleep paralysis, but there are thousands of patients suffering from ALS and other locked-in diagnoses living this reality on a daily basis. However, there is hope in technology to assist these people in communicating and living a full life. For many decades now, computers, digital cameras, and certain image processing techniques have enabled patients with locked-in syndrome to communicate via typing by tracking their eyes as they look at a digital keyboard.

The problem is that the market for medical eye tracking and eye typing is very small, limited to the patient population of those with various locked-in diagnoses, which has led to many older smaller manufacturers of eye typing systems being bought out by bigger companies, leaving fewer choices for the end consumer of these medical products.

Marci is a former UTA Alumnus and the product owner. She has been living with ALS for years now and needs an eye tracker and typing system to communicate. She has been using the same system for years since it was reliable and she was skilled at using it, however the manufacturer was bought out and support for the device was dropped, so they are in need of a new system as their current eye typing system decays without software or hardware support. They tried many of the newest versions of these sorts of machines but found them all to be less effective than the old machine, which led them to asking the UTA Computer Science and Engineering department who assigned two Senior Design teams to the task. Our team, EyeTGuys, was tasked with the creation of the hardware eye tracking system consisting of remote cameras, and a machine to run the image processing. Our sister team, EyeType was tasking with creating the accessibility tool and keyboard app extension.

5 RELATED WORK

After researching on multiple sites that provide patients with references to alternative communication options, I discovered a number of expensive eye gaze augmentative and alternative communication systems that were feature rich and robust. Many of them include features such as predictive typing, and saved phrases and responses.

For example, Tobii Dynavox provides its I-Series of trackers as well as the EyeMobile series of trackers. The I-Series+ trackers offer wake-on-gaze and sleep-on-gaze features, as well as IR signal integration for environmental control. [5]

Forbes AAC's Winslate 12D with Enable Eyes is another option for those patients with only eye mobility. It is another all in one package, and also another system that uses Windows for its accessibility features. [3]

Talk To Me Technologies also makes an eye gaze AAC, the eyespeak. It includes preset vocabulary sets and phrase saving features, as well as predictive typing/word prediction and realistic voice synthesis features. [4]

Eyegaze makes the Eyegaze Edge, another eye gaze AAC tablet system. It includes a short calibration duration feature, IR signal integration to control various remote controlled applications, and various design features that attempt to reduce eye strain and eye fatigue. [2]

These options are all great and work, but for our customer they do not fit the bill. For Marci, the other eye trackers she has tried have had unreliable eye tracking. She also is able to use her current system to communicate faster than any of the new ones she has tried. Reliability is the only sort of goal that our team can meet to better create a product for Marci. Marci and her care givers indicated that perhaps the fact that Marci has been using a singular eye tracker and the new ones are mostly double eye trackers, and perhaps they do not subjectively feel the same when used, leading to why Marci might be slower with newer systems.

6 System Overview

For our system, there will be two modules that interact, the EyeTGuys module, which our team is responsible for, and the EyeType module, which the EyeType team is responsible for. Overall the system will perform the function of a medical grade eye-tracking and eye-typing accessibility communication device.

The EyeTGuys module will consist of six hardware and four software components. Hardware includes the microcontroller/machine that will do the computational workload and it's power supply, one infrared lens filtered cameras, one infrared light, one character LCD display, and a physical casing and mount structure. Software includes an eye tracker, an eye gesture recognizer, a gaze-coordinate converter, and a state machine with calibration routine. The cameras will be pointed at and focused the user's individual eyes, while the infrared light will be pointed at the face in order to illuminate the eye images without bothering the user with a light in their eye. They will constantly be taking real time video and streaming that data to the microcontroller, which will be performing real time image processing within the eye tracking program, using the images of the eyes to estimate a "gaze vector". If the microcontroller is in the calibrated state, the machine will feed that gaze vector data into the gaze-coordinate converter which, based on a previous calibration session's data, estimates the screen coordinate that corresponds with the observed gaze vector and then streams that data to the other module for use in the eye-typing program. The machine will also always stream the gaze vector data to the eye-gesture recognizer which will use patterns of relative changes in the gaze vector to enable basic control functionality, like ordering a calibration or calling for assistance, without the need for accurate calibration data. Finally, when in the uncalibrated state or when an outside event triggers a calibration order, the machine enters a calibration routine which receives screen coordinate data from the other module as it displays a calibration screen and correlates it with the system's observed gaze vector.

The EyeType module will be designed and constructed by the EyeType team, but it will, in its most basic form, include a computer or microcontroller capable of running a consumer OS, and a screen/display that will act as the display for the computer while also being the screen which the EyeTGuys module is supposed to return the gaze screen coordinates.

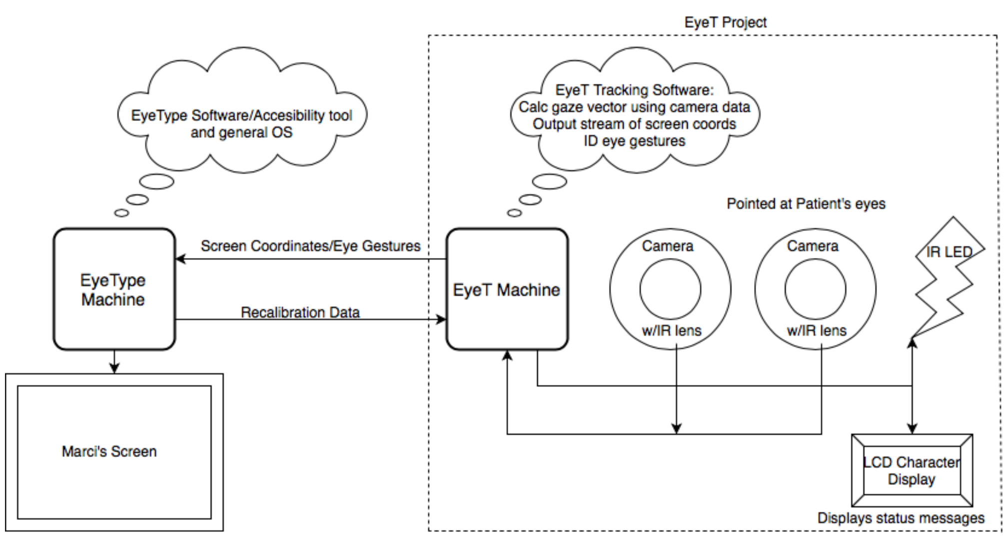

Figure 1: EyeT System Overview

7 Roles & Responsibilities

The stakeholder for this project is Marci and her family who are we are building the eye tracker hardware system for. The other stakeholder for this project is University of Texas at Arlington.

The point of contact from the customer side is Marci's father and mother. So if we need any kind of information, our team lead(George) will contact Marci's family to set up a time for the team to meet and gather the information we need.

- George Hinkel - PCB design/ hardware design
- Paul Wafula - will assist George with the hardware design and work with other team members assisting in software designs
- Jonathan El-Khoury - Implementing the software part: calibrations of the eye gestures (how the pupil moves, left, right, up or down)
- Jay Denton - Implementing the software part: Getting the data from the eye tracking software and helping other team member in gathering data from just the pupil
- Dharampreet Gill - Implementing the software part: Cutting out the outer portion of the eye from the software images data and help to gather data from just the pupil

Product owner with remain same through out the project. Which is UTA and Marci, since the hardware system will be built for her. As for scrum master, for right now the scrum master is Jonathan, who is making all the criteria list of what needs to be done in the future and also working with the team lead George on making crucial decisions.

8 Cost Proposal

Our approximate budget for the project is ends up being 1100 due multiple for expensive that we need to take into consideration for the whole hardware system design. It might end up being more then 1100 because initially we might just build prototype and build a final product afterward. Initially UTA gives 800 for senior design project. So, that money will come from UTA and from our sponsor if needed, since this a a sponsor project. the money will be spent on the external SSD that we need to run our project which will have Ubuntu running. Also the cost of making the hardware model, and the nvidia tx2 embedded computer. All of the product that we will be buying will be justified by our professor too.

8.1 Preliminary Budget

Table 1: Budget Table

Items	Cost
nvidia tx2	800
SSD	100
PSEye camera	15(x2)
IR filters	20

8.2 Current & Pending Support

The funding source for the project is 800 dollars giving by UTA. Since this project is sponsored if we need anymore funding our sponsors will take care of that.

9 Facilities & Equipment

The Eye Tracking project is made up of two teams: a software team - EyeType, and a hardware team - EyeT Guys. The lab space that can contain two team members is required. It is for the convenience of two teams to work and communicate with each other. For programming and testing of the eye tracking algorithm and the Eye Tracker, at least two desktops are needed. Makerspace is considered to be needed at the end of semester for packaging and product delivery with use of a 3-D printer.

To successfully complete the final project. The other support equipment that are required includes, but not limited to, an embedded computer TX2, two PSEye cameras, a few IR LED lights, couple IR filters, a SSD hard drive, a Printed Circuit Board and some cables.

The embedded computer is for better performance of running program. And it can be put on a electrical wheelchair without taking too much space. The NVIDIA® Jeston™TX2, Linux computer, has already present in the lab, is needed to processing pupil tracking algorithm and controll IR cameras and IR lights. Another embedded computer required by EyeType is Intel® NUC, which is used to develop eye tracking softwares. These two embedded computers are connected through USB servers locally.

One hard drive is needed to install or dual boot desktops in the lab with Linux system. IR LED lights are needed to illuminate the patient‘s facial area. IR filters attached to the PSEye cameras are needed to locate and find the patient‘s pupils. These materials can be purchased from Amazon, or Fry‘s Electronics.

10 Assumptions

- Couple weeks are needed for eye tracking algorithm research.
- The wheelchair purchased would be the same size and style of Mrs. Marci‘s.
- The Eye Tracking remote which includes cameras and LED lights can be packed into a case made by 3D printer.
- This product is compatible with the software applications implemented by Eye Type team.
- A basic setup would be done in the fourth sprint.

11 Constraints

Constraints are limitations imposed on the project, such as the limitation of cost, schedule, or resources, and you have to work within the boundaries restricted by these constraints. All projects have constraints, which are defined and identified at the beginning of the project.

Constraints are outside of your control. They are imposed upon you by your client, organization, government regulations, availability of resources, etc. Occasionally, identified constraints turn out to be false. This is often beneficial to the development team, since it removes items that could potentially affect progress.

This section should contain a list of at least 5 of the most critical constraints related to your project. For example:

The following list contains key constraints related to the implementation and testing of the project.

- Final prototype demonstration must be completed by August 7th, 2019
- Total budget can not go past $1500
- Customer must be able to control everything in her own environment
- Hardware used must be able to run the system software

- System can not cause any type of discomfort to the customer
- Must be able to send out alert when software freezes

12 Risks

There are five critical risks related to our project. The probability of occurrence, size of loss, and risk exposure related to each of risks are listed and organized in a table shown below:

Risk description	Probability	Loss (days)	Exposure (days)
Software is not ready for testing	0.30	5	1.5
Patient is not available for testing	0.20	5	1
Hardware is not capable of operating given software	0.30	10	3
Delay on certain parts needed for prototype	0.10	10	1
A sudden major revision to the plans	0.10	5	.5

Table 2: Overview of highest exposure project risks

13 Documentation & Reporting

13.1 Major Documentation Deliverables

The initial version of the project charter will be delivered on February 25th, 2019. The initial version of the system requirements specification document will be delivered at the end of sprint 2 on March 18th, 2019. The initial version of the architectural design specification document will be delivered at the end of sprint 3 on April 8th, 2019. The detailed design specification document along with all final document revisions will be submitted by the end of the semester on May 3rd, 2019.

13.1.1 Project Charter

This document will be maintained and updated as necessary. We predict that we will need to update the charter once per sprint cycle as we learn and figure out what we need. The initial version will be submitted on February 25th, 2019.

13.1.2 System Requirements Specification

This document will be maintained and updated as necessary. We predict that we will need to update the SRS once per sprint cycle as we learn about and receive updated system requirements. The final version will be submitted on April 29th, 2019.

13.1.3 Architectural Design Specification

This document will be maintained and updated as necessary. We predict that we will need to update the ADS once per sprint cycle as we make necessary changes to our system architecture. The final version will be submitted on April 29th, 2019.

13.1.4 Detailed Design Specification

This document will be maintained and updated as necessary. We predict that we will need to update the DDS once per sprint cycle as we make necessary changes to our system's design. The final version will be submitted on August, 2019.

13.2 Recurring Sprint Items

The following will be recurring sprint items: the Project Charter, the System Requirements Specification document, the Architectural Design Specification document, and the Detailed Design Specification document.

13.2.1 Product Backlog

As new requirements arise, we will update our SRS accordingly. These items will be prioritized according to difficulty and time required to complete. Decisions regarding this will be made by the product owner. We will use Google Sheets to keep track of our product backlog.

13.2.2 Sprint Planning

The sprint plan will be decided by the scrum master and will occasionally change each sprint depending on the workload of each group member. There will be a total of 7 sprints.

13.2.3 Sprint Goal

Sprint goals are decided by our scrum master, who is Jonathan. He will come up the criteria that we must meet over each sprint. Our customer won't be involved in every sprint but ones we have extensive amount of stuff done then we will notify the customer for a trial run.

13.2.4 Sprint Backlog

Which ever team member is working on the item for that sprint, if they are able to finish up the item that they are working on for that sprint early, they have the option to start early on item that is listed on the sprint backlog. For now our team has decided that the sprint backlog be maintained with a spreadsheet. Where we will list the task needs to be that done for that week and the team member that are suppose to be working on it.

13.2.5 Task Breakdown

The team members will be assigned task based on their skill set. And most importantly which major they are. Our computer engineering team members will be working on the hardware tasks, while computer science will be working on the software aspect of our project. While still helping the engineering team with the hardware part. The team lead will assigned parts to the team members, where we will have due date listed on our sprint log. Whoever works on a task that is listed on the scrum sheet will also record their time on that.

13.2.6 Sprint Burn Down Charts

George, the team leader, will be responsible for generating the burn-down charts for each sprint. The total amount of effort expended by each individual team member will be determined from two metrics, the total number of tasks completed by the member during the sprint as well as the sum of the difficulty estimations of the completed tasks. The burn down chart will be formatted as a stacked line area chart, tasks remaining versus date. However, care must be taken by the scrum master to ensure that any team member is not artificially inflating their effort metrics.

13.2.7 Sprint Retrospective

To handle the retrospective we will holding team meetings of the task that we weren't able to finish. Soon after each sprint is finish we will hold a meeting right after looking at tasks/item that needs more attention. We will document this as a separate log items that needs to be worked on. They will be due on their impertinences.

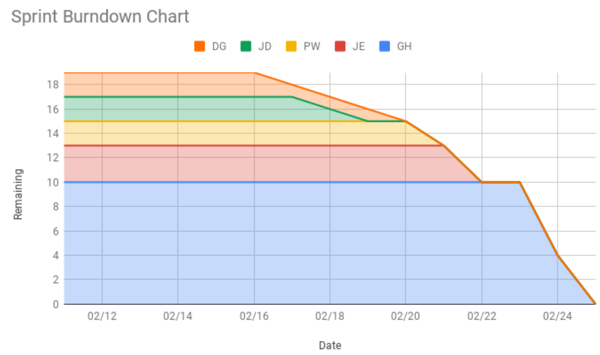

Figure 2: Example sprint burn down chart

13.2.8 Individual Status Reports

Each team member will update the sprint log whenever they are done with their task. Also we will have a team GitHub, so soon after an individual finishes their task they will push their content on to git so it can be accessed by everyone. The key item will be just how much time they spent on it and if it works efficiently as planned or where there any changes that needs to be be notified to the other team members.

13.2.9 Engineering Notebooks

The engineering notebook will be updated after every team meeting by whole team and as for per individual it will be updated after they have finished the assigned task. The minimum amount of pages that should be completed for each sprint is 2-4. Each interval will be a sprint long. Each member is accountable for their own task, if the assigned task is not done, then they will document that. Each team members will ask other team members to sign their notebook after each documentation.

13.3 Closeout Materials

13.3.1 System Prototype

The final system prototype will have everything that our customer need, which is a complete working remote eye tracking system that will be attached to her wheelchair. This will hopefully be done at end of summer. Yes, just for our prototype, we will conduct a test with the customer and see how they like it. The test will most likely be conducted off-site at our clients house.

13.3.2 Project Poster

Project poster will include the picture of our hardware system, the dimension of that will 2 feet by 4 feet. It will be done after our final product is done.

13.3.3 Web Page

There will be a blog web page, on school website, that has general information about the EyeT Guys group members, sponsors, project abstract and background, project requirements and system overview. Links attached to project documentations are also included.

13.3.4 Demo Video

The demo video will show one of our team member using our product and how its used and all of it features. The video will approximately be 3-5min long demonstrating all of our products features. We won't any B-reel footage for future because its not needed. Our product is mainly for our client use.

13.3.5 Source Code

Our source code will be maintained using Git version control and Github to synchronize across devices and team members. Source code will be provided to the customer as well as clear directions on how to change or flash the embedded controller with new software if so desired. The source code will be provided as a link, since it will be open source and publicly available on Github. The project code will be open source under the GNU GPL v3.0

13.3.6 Source Code Documentation

We will document our code to such a standard that any competent engineer would be able to understand and build upon our code. We will probably use something along the lines of Doxygen and we will provide the final documentation in browsable HTML for the structure that provides.

13.3.7 Hardware Schematics

- Abstract Circuit Schematic: All the electrical connections between major components
- Motherboard Circuit Schematic: Circuit schematic of "motherboard" PCB to mount the TX2, power connections, as well as any peripheral connections we need, cooling, etc to enable the software and usability goals of this project.
- Motherboard E-CAD PCB file: E-CAD PCB design of the "motherboard", this is the actual layout and specifications for the PCB itself

13.3.8 CAD files

We will use Solid Works or PTC Creo (undecided currently) to produce the 3-D CAD files necessary for the physical casing and mounting hardware. We will provide STEP files in our closeout materials.

13.3.9 Installation Scripts

There should be no need for the customer to ever deploy our software to a new installation, however we will provide source code, documentation, and instructions on how to install or reinstall the software.

13.3.10 User Manual

We will provide a digital user manual in either HTML of PDF format, whichever allows for the best explanation of how to use our product.

References

[1] DOD Sample Phase I Proposal. Technical report, U.S. Department of Defense, 2011.

[2] https://eyegaze.com. Eyegaze edge, 2019.

[3] https://www.forbesaac.com/winslate series. Forbes aac winslate, 2019.

[4] https://www.talktometechnologies.com/pages/eyespeak. Talk to me technologies eyespeak, 2019.

[5] https://www.tobiidynavox.com/products/devices/. tobiidynavox i series and eyemobile, 2019.

[6] Michael Lant. How To Make Your Project Not Suck by Using an Agile Project Charter, 2010.

[7] Kenneth S Rubin. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley Professional, 1st edition, 2012.