

To: *Dr. Rod Johnson, Professor, EECS 481*
Dr. David Chesney, Professor, EECS 481

CC: *Mr. Chris McMeeking, Instructor, EECS 481*
Mr. David Sabourin, Instructor, EECS 481

From: *Team SET*
Anthony Chiang, Student
Deng Ke Teo, Student
Steven Uy, Student

Subject: *Software Requirements Document for Eye Tracking Communication Application for People with Cerebral Palsy, EECS 481 Project*

Date: *8 October 2013*

<i>Version</i>	<i>Date</i>	<i>Revision</i>	<i>Author</i>
1	2 October 2013	First Draft	Steven Uy
2	5 October 2013	Second Draft	Deng Ke Teo
3	6 October 2013	Third Draft	Anthony Chiang
4	14 October 2013	Fourth Draft	Anthony Chiang
5	17 October 20123	Fifth Draft	Deng Ke Teo
6	19 October 2013	Final Version	Steven Uy

Project Definition

In this section, we will discuss the background, objectives, and considered stakeholders behind our project.

Background

Communication is an important aspect of everyone's life. Being able to communicate allows people to share their thoughts and feelings and express their intent. Unfortunately, many people with disabilities lack the ability to communicate either through speech or gestures. Moreover, many of them lack tools that can aid them in communicating or the ability to use communication tools.

People with cerebral palsy are often unable to fully enjoy the benefits of technology. For example, because of their lack of motor skills, they struggle with operating devices such as tablets, personal computers or even household appliances such as microwave ovens. Moreover, due to their conditions, they are sometimes unable to properly interact with the people around them. Being able to interact with others is important for everyone, even for people with disabilities. To improve the quality of life of people with cerebral palsy, we propose a communication tool using eye-tracking technology that will allow cerebral palsy patients to potentially be more expressive in communicating their thoughts. The tool we are proposing leverages the eye-tracking capabilities of the Creative Intel Gesture Camera to enable people with cerebral palsy to be better at communicating.

In the context of Grace, a teenage girl with cerebral palsy, and her family, who we are working closely with on this project, daily communication between Grace and the people around her is usually binary-she would reply yes or no to questions by waving her hand or shaking her head. In school, she uses a table of pictures to communicate her thoughts, which usually takes some time because of her limited movements. Our communication application, Speller using Eye Tracking (SET), that serves as a hybrid between an on-screen keyboard and communication grid will allow Grace to communicate with family members at home and friends and teachers in school more expressively.

The eye-tracking communication tool we will be developing will have features that will allow Grace and people with cerebral palsy to communicate expressively using eye motions. Our tool will feature multiple modes that allow for communication with the most generic mode being a keyboard layout that will allow users to create text sentences. Other modes available include a customizable layout of expressions or commonly used phrases that will allow users to communicate their thoughts quickly. We also plan to integrate day-to-day online features into our tool such as e-mail functionality and have the tool be fully customizable.

The software development kit for the Intel Creative Camera is for the C++ programming language. That will be our main programming language for configuring the Creative Camera to capture users' eye motion and obtaining input from the users. For the GUI, we have chosen to work with the Windows Presentation Foundation (WPF) Framework which is in C# as it is highly configurable.

Objectives

Our objective for this project is to have a working and properly tested eye-tracking communication tool that will allow people with cerebral palsy to express their thoughts beyond what they normally can. In Grace's case, the tool will allow her to form proper sentences to communicate with; thus enhancing her expressiveness and allowing her to communicate beyond binary gestures. We have until the end of this semester to complete the project, with our final demos and presentations being held on 13 December 2013. That translates to about two and a half months of work and we want our tool to at least have basic communicative functions such as a keyboard layout to create sentences with.

Stakeholders

This section introduces the stakeholders for our project and explains how each stakeholder is relevant to our product.

Grace and her parents

This project revolves around creating a tool that will help Grace, a teenage girl with cerebral palsy, in four key areas of her life, namely, play, communication, school-life and home-life. Since Grace's parents have a part to play in each one of these areas, their feedback and evaluation on the direction and progress of our tool is necessary for us to meet Grace's needs.

Cerebral palsy patients and their caregivers

Doctors who work with cerebral palsy patients can use our tool to perform empirical analyses on their patients and find out how much our tool will aid them in their everyday life and to find out more about their patients' needs. For cerebral palsy patients, this tool will help them like they will help Grace- it will allow them to be more expressive in their communication.

Dr. David Chesney

The instructor for EECS 481, Software Engineering, and overseer of all the projects in the class is Dr. David Chesney. He provides us with the necessary tools and guidance needed to come up with a final project. All forms of documentation regarding the project, as well as the development and testing of the different stages of the project have to undergo his evaluation.

Technical Issues

This section discusses the user analysis, and variety of requirements that define the project on a technical level.

User analysis

The application is directed towards a universal user, whether Grace, those with disabilities, or others. Any user has the same functional requirements and features. The user is able to configure the eye tracking and select characters on the qwerty or T-9 interface organized by tabs. Additionally, the user can adjust selection time and idle timeout. More details are listed in the requirements definition section.

Requirements definition

In this section, we will discuss the fundamental requirements and supporting ideas in our product.

Project Design and Constraints

Resources

The Intel Creative Gesture Camera will be used for eye tracking. The camera connects to a Windows platform via USB, preferably a laptop with a 15" screen. The camera will be using the Intel Perceptual Computing SDK and openCV library.

Input method

Selection and hovering will be controlled using pupil and head tracking. A user will be able to spell out a sentence, select a method of output, all through eye tracking. Eye tracking and head tracking provide information for a gaze estimation to place the cursor where the user is looking at the screen. The cursor can still be controlled using a mouse or track pad if necessary.

Core values in design

Spelling charts are commonly used for communication purposes for those with a disability. Using a spelling chart commonly involves pointing out letter by letter until one can guess or complete the word or sentence. This process can be very slow and difficult for the user with poor motor control. By using the eyes, the user can search for the letter and select it at the same time, rather than searching and then pointing for selection. This is meant to increase communication while decreasing effort required.

Functional Requirements

On-screen input

The user interface contains several methods of character input such as an onscreen QWERTY keyboard, a T-9 grid, and previously saved words or phrases. Each of these methods have individual views, all organized by a tab management system. Creating a sentence can use any combination of each method.

Cursor control

The Intel Creative Camera pulls the following information, the location of the head, eye corners, pupils, and the pitch and yaw of the head. Each of these variables is used to provide gaze estimation to relate to where the cursor currently is. Selection is made where a user focusing on a button for an adjustable amount of time.

Text options

Entered text into the text window can be used in many ways involving word completion, common sentences, and output or saving methods. As a user begins to type part of a word, a list of possible words will appear. The same is true for sentences, where saved sentences may be selected if several words in the string are entered. A user can additionally save common words or phrases to enter instantly in different tabs. Once a message is entered, the user has options of text to speech or text to a file.

Non-Functional Requirements

Ease of use

For any communication aid, ease of use should be an essential focus. In the general case for our stakeholders, movement involving the eyes is easier than an arm or hand. Additionally a laptop with an Intel Camera can fit on a tray or table working at varying distances. The interface should be simple enough to navigate, even on a first use basis. As a user traverses through the process, searching, selecting, and completion, each step of the way should be comfortable and natural.

Accommodation for scenarios

Several scenarios arise when using tracking algorithms with the camera. Scenarios that may lead to tracking error involve multiple faces on screen, movement in background, poor lighting, and glasses blocking pupil detection. Specific to cerebral palsy, unintentional head movements can be accommodated by an adjustable idle timer, so a user can resume typing a sentence if they turn away from the screen.

Open Issues

In this section, we will discuss the management of the risks associated with the entire SET project. Risks are listed in table format with each table describing the risk, how the risk can be detected, how it can be avoided and how it can be mitigated.

Risk Description: Grace's intermittent involuntary head movements may hinder the focus of her pupils on the screen.
Risk Detection: The group will observe and record how long it takes for her to return her head to the correct position.
Risk Avoidance: The risk will be lowered by adjusting the selection and focus of the pupils, with users deliberately turning their head away at varying speed to allow for adjustments.
Risk Mitigation Plan: To prevent accidental selection, as the camera detects any turn of the head outside a rotation range, the program will remain idle for a time period determined by the user until the head returns within the range and pupil's are focused on the screen.

Risk Description: Pupil tracking is impossible with the Creative Intel gesture camera.
Risk Detection: If the existing SDK for Creative Intel camera fails pupil tracking nearing a release date, a hardware change will be discussed.
Risk Avoidance: The risk will be decreased by starting early testing and research with the existing SDK for the Creative Intel camera.
Risk Mitigation Plan: Many hardware alternatives exist, such as the Kinect with an external lens or a high definition webcam.

Risk Description: Grace's eye movements may not be stable enough for our implementation.
Risk Detection: The group will work with Grace, observing her pupil focus while reading text on a screen.
Risk Avoidance: This risk can be avoided through extensive testing of the eye tracking feature to ensure that Grace's eye movements match up with the cursor on the screen. Initial testing without Grace will need to focus on simulating movements similar to Grace's.
Risk Mitigation Plan: If Grace's movements do not match our expectations, we will have to adjust the eye tracking to accommodate them.

Risk Description: Multiple objects or people in the background being detected may interfere with the eye tracking and selection.
Risk Detection: This risk can be detected through tests with the Intel camera such as multiple people or moving objects in the background.
Risk Avoidance: This risk could be avoided by designing application only to track the pupils of Grace and ignore background images.
Risk Mitigation Plan: If this risk does occur, the environment can be controlled without additional objects or eye tracking can be designed specifically for Grace.

Risk Description: Student who is recently fired decides to join our team
Risk Detection: Student requests to join the team.
Risk Avoidance: Refuse to allow student to join the team.
Risk Mitigation Plan: New team member will be assigned simple tasks until he or she is fully familiar with the project and the team's source code.

Risk Description: Loss of a team member
Risk Detection: As per the group contract, members may not leave the team one week prior release (alpha/beta/final). Moreover, they must inform all team members before leaving.
Risk Avoidance: A group meeting will be held and members will be deterred from leaving.
Risk Mitigation Plan: Using the Pivotal Tracker tool, tasks can be easily reassigned to the remaining members of the team.

Project management

In this section, we will discuss the task breakdown and scheduling for deliverables, changes to the original scope, assessment of progress and coding style standards.

Updated task breakdown & schedule

Our project has two main aspects: eye-tracking and user interface. For the majority of our application's development, these two aspects are decoupled, meaning that they can be developed independent of each other. Every few weeks, work done on these two aspects will be integrated to ensure coherency.

Because our team is relatively large, we have decided to divide our team into two sub-teams. Sub-teams model the Agile project management approach and work in parallel. Each sub-team can be working on one of the two aspects concurrently. This will reduce the inefficiency caused by a large team and allow us to capitalize on our team's size and develop rapidly.

Figures 1 and 2 are Gantt Charts showing the start and end dates per task split between the interface team and the eye tracking team. To manage these tasks in a structured manner, we are using Pivotal Tracker, an agile project management tool to keep track of our tasks.

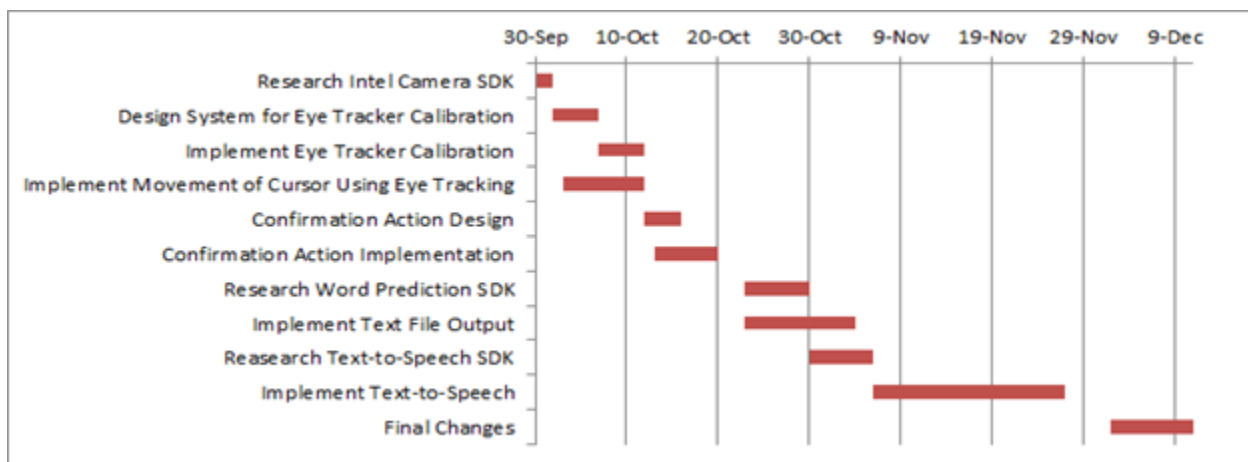


Figure 1: Schedule of eye tracking team's milestones and deliverables

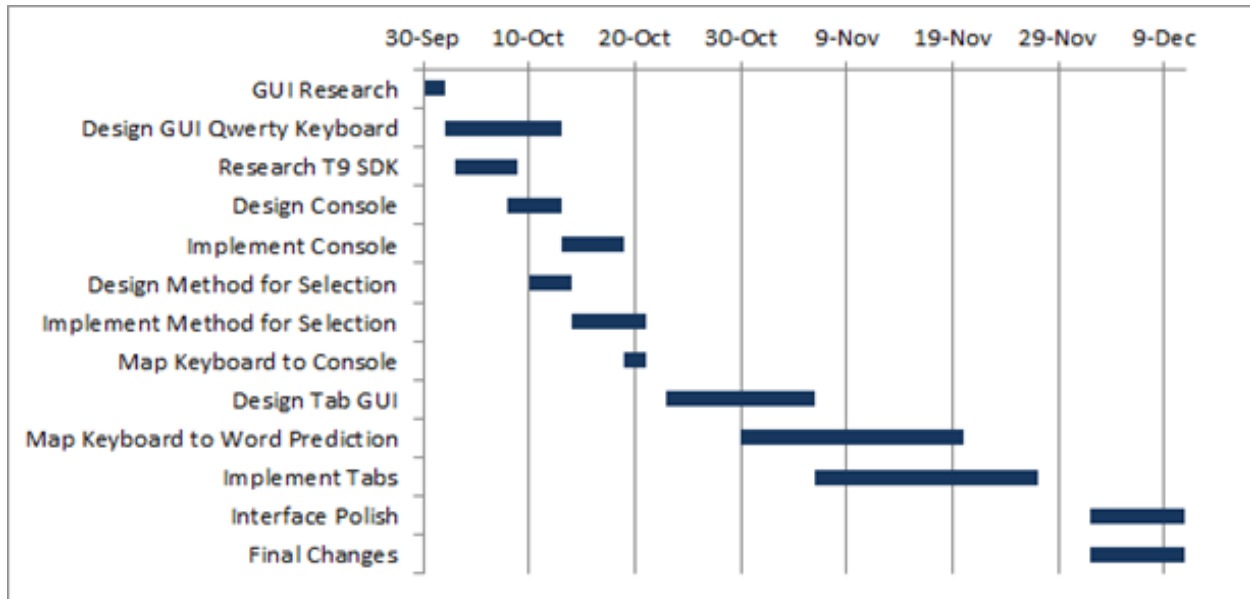


Figure 2: Schedule of UI team's milestones and deliverables

Changes of scope

The scope has not changed since the submission of our scopedoc. We continue to follow the roadmap model that is depicted by Figure 2:

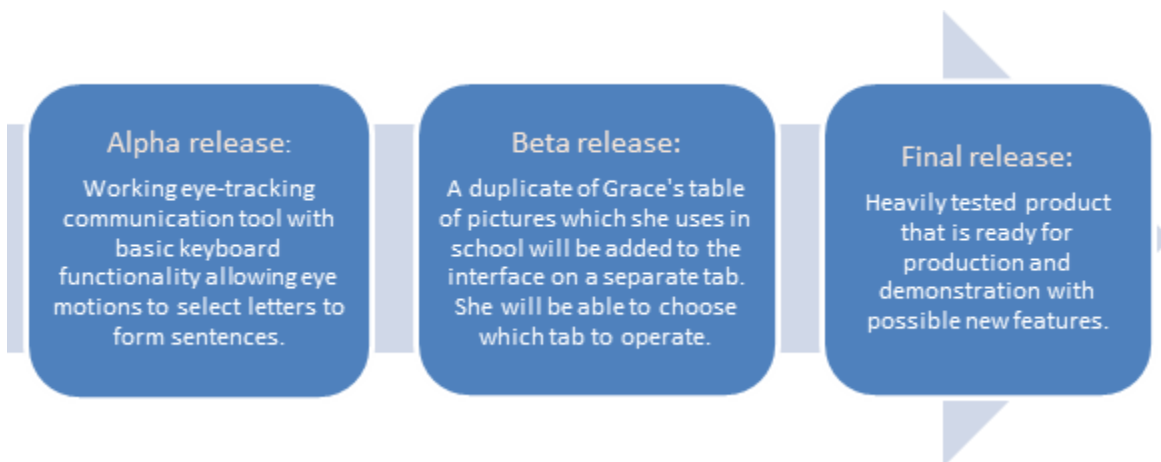


Figure 2: Roadmap for version releases of eye-tracking communication tool.

Assessment of progress

To date, the backend sub-team working on eye-tracking has had to work with only one camera. Due to this, the backend team started off with inertia as members had to rotate the use of the camera. We have requested for another camera to work with and the course instructor has gladly agreed to provide us another one. This will lead to higher productivity.

Development standards

Coding style

Functions, classes, and variables should follow the camel case standard. Classes should have their first letter capitalized and should be in singular form. Variables should have meaningful names, for example 'foo' and 'bar' are not particularly enlightening.

Containers should have plural names, for example a vector holding Person objects should be named people.

Examples:

```
void keyPress(//function arguments);  
class KeyButton;  
int counter;  
vector< Person> people;
```

Version control

Our team will be using an open-source distributed Version Control System called “Git”. With Git, every user has a local copy of the entire repository (including its entire history). This solves common problems of centralized Version Control Systems such as:

- 1.If the central repository goes down, you can't commit code to the repository, nor can you pull code from the repository.
- 2.If the central repository explodes, you have lost the record of your code. Any previous versions are lost.

Conclusion

SET, which serves as a hybrid between an on-screen keyboard and communication grid will allow Grace to communicate with family members at home and friends and teachers in school more expressively. Set will allow Grace to communicate with her eyes by tracking her eye gaze and recording input based on where she is focusing her gaze on the UI keyboard.. It is certainly true that the SET is designed with Grace's interest as its foremost objective. However, the nature by which it is designed will hopefully prove to be useful for others affected by cerebral palsy. Some aspects that demonstrate this principle include the adjustable parameters for feature selection and validation. We are confident that by keeping in mind the importance of customization and extensibility, our class will succeed in developing a communication application that meets Grace's needs.

If you have any questions with regard to SET tool, please send an e-mail to teamset@gmail.com and allow us 2 business days to get back to you, thank you.