

# Hidden Guardian

## PROJECT PLAN

Team 9

Kelli Rout

Swamy Ponpani

Jennifer Frank - Team Lead

Keng-Yik Ho - Chief Engineer

Matthew Pedretti - Hardware Engineer

Tom Kirby - Backend Development

Jacob Stilwell - Computer Programmer

[sddec18-09@iastate.edu](mailto:sddec18-09@iastate.edu)

<https://sddec18-09.sd.ece.iastate.edu/>

Revised: 3/26/18

# Table of Contents

<b>1</b>	<b>Introductory Material</b>	<b>5</b>
1.1	Acknowledgement	5
1.2	Problem Statement	5
1.3	Operating Environment	5
1.4	Intended Users and Intended Uses	5
1.5	Assumptions and Limitations	6
1.6	Expected End Product and Other Deliverables	6
<b>2</b>	<b>Proposed Approach and Statement of Work</b>	<b>7</b>
2.1	Objective of the Task	7
2.2	Functional Requirements	7
2.3	Constraints Considerations	8
2.4	Previous Work And Literature	8
2.5	Proposed Design	8
2.6	Technology Considerations	9
2.7	Safety Considerations	9
2.8	Task Approach	9
2.9	Possible Risks And Risk Management	10
2.10	Project Proposed Milestones and Evaluation Criteria	11
2.11	Project Tracking Procedures	11
2.12	Expected Results and Validation	12
2.13	Test Plan	12
<b>3</b>	<b>Project Timeline, Estimated Resources, and Challenges</b>	<b>12</b>
3.1	Project Timeline	12
3.2	Feasibility Assessment	13
3.3	Personnel Effort Requirements	13

3.4 Other Resource Requirements	14
3.5 Financial Requirements	14
<b>4 Closure Materials</b>	<b>15</b>
4.1 Conclusion	15
4.2 References	15
4.3 Appendices	15

## List of Figures

Figure 1: Speaker/Microphone system

Figure 2: Complete system

Figure 3: GitLab Issues Board

Figure 4: Semester 1 Gaant chart

Figure 5: Semester 2 Gaant chart

Figure 5: Timeline Diagram for Spring Semester

## List of Tables

Table 1: Timeline of proposed schedule for the Spring semester.

## List of Symbols

Not Applicable.

## List of Definitions

Not Applicable.

### 1.1 ACKNOWLEDGEMENT

Professor Swamy Ponpandi provided technical support and shall be acknowledged for assisting Team 9 in the development of Hidden Guardian. Similarly, Professor Zambreno and the SE, CPR E and EE departments provided financial support in the creation of this product.

### 1.2 PROBLEM STATEMENT

Kid's lives today involve online interactions more than ever. Their interactions with strangers invoke the risk of sharing unwanted personal information, password theft, receiving viruses, being cyberbullied and more. However, currently there is no practical parental monitoring system to keep track of these unwanted videogame interactions.

Our solution is Hidden Guardian. Hidden Guardian is a combination of a speaker/microphone for a gaming console. It will connect to the gaming console as well as the Hidden Guardian's mobile app. Through the mobile app the parent user will be able to input keywords, such as meet, phone number, address and will be notified if they are used in chats. The app will also provide live text streams of the child's chats and specific text paragraphs of chats with those keywords. Hidden Guardian will only be able to be enabled or disabled by the parent.

Overall, our goal is to provide a system so that parents can have peace of mind that their kids videogame interactions are safe.

### 1.3 OPERATING ENVIRONMENT

The typical operating environment of Hidden Guardian is a household setting. Given that houses are typically well climate controlled the hardware component won't be subject to any extreme conditions. It will however be handled frequently and should be at least somewhat robust in order to withstand moderate impacts.

### 1.4 INTENDED USERS AND INTENDED USES

The intended user for this project will be for 2 targets, parents and their children. The children will be able to use to bluetooth speaker for their gaming uses or listening to music. The speaker will provide immersive gaming experience by providing high quality audio. The device will also have a mic to enable children communicate with other players online to improve gameplay.

Parents will be using it as a device to monitor their children activities. By using the mic on board, the device will record all conversations made by the kid when the device is turned on. Parents will be able to access logs of recordings using the app and find keywords of the recording such as phone numbers, address, to help improve the safety of their children.

## 1.5 ASSUMPTIONS AND LIMITATIONS

### Assumptions:

- We don't have to worry about privacy laws, only the Xbox TOS.
- The maximum number of tracked voices will be 8 at a time.
- At the end of development, this product will not be commercially ready to consume
- The maximum amount of hours/week of data we will store is 15. This is the assumption because 9 is the average per child, so we will store the upper limit of this data point.

### Limitations:

- We will only be able to cover one language, that being english.
- The amount of audio data that can be stored in the local object.
- Our console application is running in the background with the limitation that it cannot to be sold on the Windows platform. This can be fixed later, but currently this acts as a limitation.
- Following Xbox guidelines can be restricting
  - User's ability to exit out of a background application
  - Ability to access text chat logs

## 1.6 EXPECTED END PRODUCT AND OTHER DELIVERABLES

### - Console App

The console app will be a data logging application for the xbox one console. Parents will be able to set up the app to record their child's audio conversations over xbox live and the console application will then convert it from speech to text. Access to the app will be restricted so that only the parent account can enable or disable it.

### - Smartphone App

The smartphone app will allow users to review and search through the data collected by the console app. It will have a user friendly interface and provide a variety of tools for filtering through the stored data. These tools include things like being able to search for keywords or users and having all relevant conversations pop up.

### - Hardware

A wireless speaker and microphone attachment that allow a user to communicate with a gaming console. The microphone attachment on the controller will have a set of controls for changing different channels of volume (e.g. game volume, chat volume). The speaker will run on a rechargeable battery and may be rechargeable via a usb cord.

The initial stages of each of the mobile and console app will be delivered by May and the final versions of those components as well as the hardware will be delivered by the beginning of December.

## 2 Proposed Approach and Statement of Work

### 2.1 OBJECTIVE OF THE TASK

Our goal is to create a functional speaker/microphone, an android applications and a console application.

#### **Hardware**

Functional speaker and microphone - this hardware component will connect to the gaming console and mobile applications. It will act as a replacement headset for the child user and will receive all chat data to be sent to the software components.

#### **Software**

Console application - The console application will store all voice conversations, friends list and gather data about the child's gaming. It will be protected from allowing the child to enable or disable Hidden Guardian.

Android Mobile application - this application will communicate with the console app and hardware to present all the information to the parent. It will be very user friendly.

### 2.2 FUNCTIONAL REQUIREMENTS

The device will have the function to record conversations done by the kids and double as bluetooth speaker with wifi connection capabilities. Basic features of the speaker will be bluetooth connectivity, a volume button, a mute button, game volume, a power button and powered by a battery which is rechargeable. The speaker will send information in a non-standard way such that to work with our program you have to have our speaker. The main feature of the device would be hidden from the kids which is the ability to record conversations. Other feature will be an app the will enable parents to access the the conversations and chat logs from multiple platforms and find key words and phrases with a parental lock on the app so the kids do not remove the app accidentally or purposely.

### 2.3 CONSTRAINTS CONSIDERATIONS

We will want our product to be user friendly for parents who may not be very tech-savvy. Having a help option and using a language that is understandable to them.

We also have to consider the privacy of this product. Online members who participate in video game chat sign a release with XBox that they are willing to extend their privacy rights when using this product. This is meeting a standard, though may be considered unethical. The product is helping parents monitor unsafe actions, cyberbullying and avoiding password theft but it is also recording others so it is in a grey area.

### 2.4 PREVIOUS WORK AND LITERATURE

#### **Android Application**

From our research, you plan to use android studio to host the android application. We will connect our java application to a mysql database using JDBC (Vogella). We realized that the sql database will not be able to store the actual audio files so it will just be used to store usernames, passwords, users that are blocked, keywords and the titles of audio files.

#### **Bluetooth Speaker/Recording device**

From our research, there have been several diy projects made by electronics enthusiast that are similar in functions but different in purpose. I realize that a simple Bluetooth Module is able to establish connection with any bluetooth device and so the speaker is not only limited to connecting to the xBox one. The research on the recording device made us realize that using a Arduino microcontroller is able to record and store the recordings in an SD card.

#### **Current Similar Devices**

Currently, there is no device with both functions. But there are devices that have seperated functions

### 2.5 PROPOSED DESIGN

Some potential solutions we came up with included having a speaker that listened to the audio, adapted that audio to text and then sent that to the mobile application. The issue with this proposal was that clarity would most likely be poor and there would be no



guarantee of which user was saying what. It also posed for some issues with storage on the console application as it would require a lot of storage.

Another design idea we had is that once we gathered the data to directly send it to the mobile application without converting it to text. The issue with this idea was that it would require potentially 2.5 GB of data that would be hard to store on the speaker and also nearly impossible to feasibly store on a mobile device. We also considered having a device that would act as a blocker and would collect the audio data that was being sent to the headphones and send it to the mobile application, but the logistics of having that device/our speaker do that would get complicated for security reasons. These ideas all had many roadblocks and didn't seem as feasible or inexpensive as our proposed design concept.

Our proposed design consists of a bluetooth wireless speaker that allows a user to communicate with other players while online gaming hands free. The speaker will come with a code that gives it access to the console application code. The console application will gather the user's audio data when communicating with other user and convert that data from audio to text on the console application that is running in the background of the console. It will then transmits that data to a database which can be accessed and searched using a secure smartphone app. The search tools will allow the user to filter through the gamer's chat data via text transcripts and will notify the user/guardian based on specific keywords they have set up. Similarly, if the same user is being flagged for using the keywords marked as malicious it can notify the user/guardian of the flagged user. We also plan to have a microphone that is attached to the controller to provide pause/play and volume change functionality as requested by our client.

## 2.6 TECHNOLOGY CONSIDERATIONS

- Mic might be too sensitive that it will pick up audio from the speaker but if using a less sensitive mic, might risk not be able to record voices
- One of the main choices made on the software side of the project is the choice of programming languages for the different software components. Different languages have will have varying levels of support and available libraries to reference. We also want to optimize on our current knowledge of languages so that we don't spend too much time starting from scratch with a new language.
  - We decided the console we develop with should be Xbox One because it has the least amount of development license issues - however, it's coding language (UWP) has minimal documentation.

## 2.7 SAFETY CONSIDERATIONS

- Hardware needs to be safe to handle
  - no loose circuits
  - meets safety regulations of other speaker and microphone standards

- The information collected and stored must be encrypted as to not allow unauthorized access or tampering.
  - password encryption with database storage
  - hidden ability to enable/disable Hidden Guardian if you are not the parental user

## 2.8 TASK APPROACH

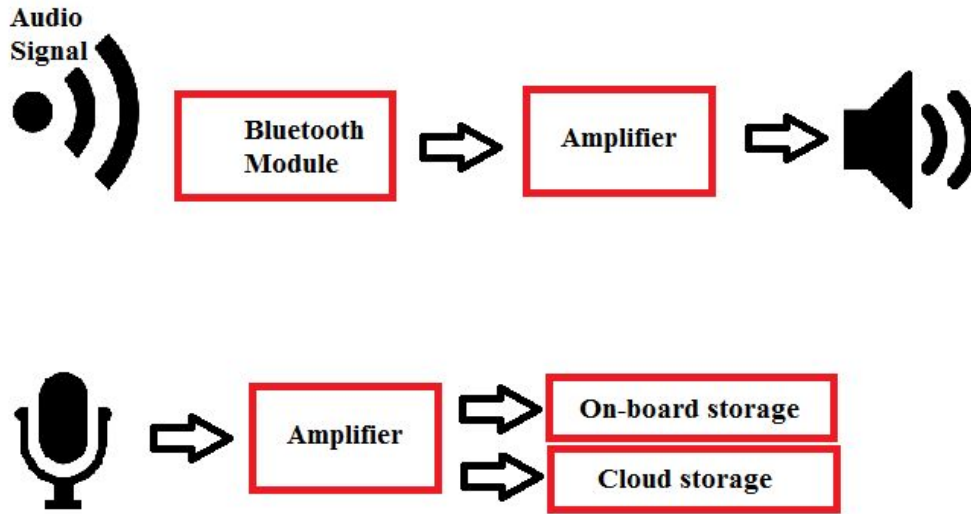


Figure 1: Speaker/Microphone system

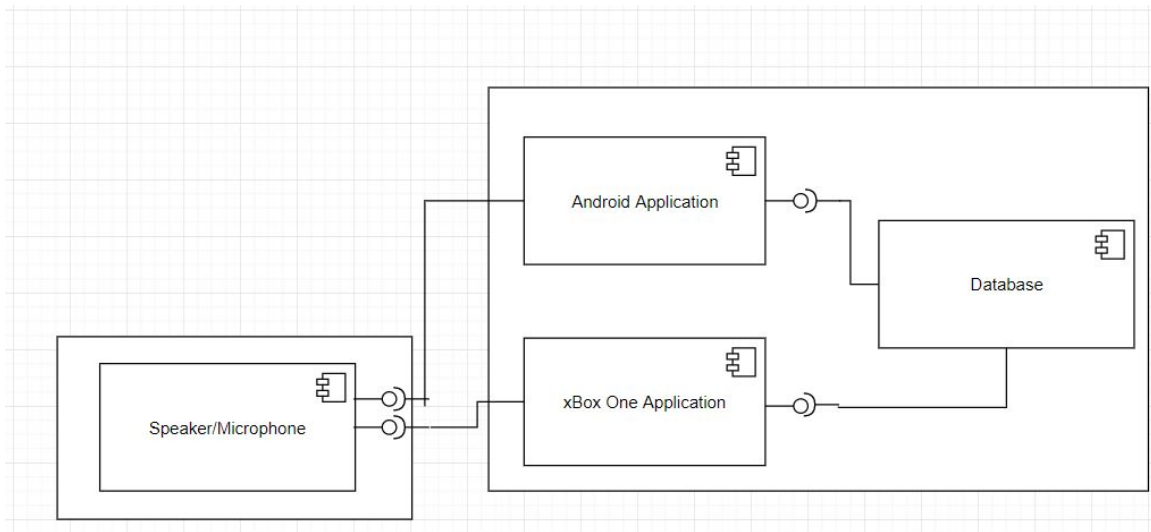
The device will contain 2 systems. One will be the speaker system and one will be the microphone system.

- **Speaker system**

Signal from the console will be paired with the bluetooth module. The audio signal will be small so the signal will have to go through an amplifier. The amplified signal will be sent to the speaker providing high quality sound

- **Microphone system**

Conversations will be picked up by the microphone as signals. The signals will then be amplified through a amplifier. The signal will then be stored in storage device or cloud storage to be accessed by the app.



**Figure 2: Complete system**

The xBox One application will access the chat logs and store them in the database. The Android application will then be able to access the database including the recordings recorded by the microphone.

### 2.9 POSSIBLE RISKS AND RISK MANAGEMENT

- Team’s lack of experience with console programming and inability to estimate development time costs
- Potential inability to receive essential information from the XBox Console
- Potential expensive costs in purchasing an XBox One console and development license
- Potential excessive costs to meet the audio storage requirements
- Muffled voices, extreme volumes (quiet or loud), distance away from speaker, and accents may affect the clarity of video recordings, false positives for keyword notifications and other accuracy issues

### 2.10 PROJECT PROPOSED MILESTONES AND EVALUATION CRITERIA

- Successfully collecting audio data from the console
  - Test conversations can be gathered by the app and correlate information like the current speaker and time of the conversation
- Successfully collecting text chat data from the console
  - Send test messages and verify that the app is gathering relevant information e.g. the content of the message, time and source of the message
- Storing collected data as a reference in a database
  - Attempt to store test data in the database
- Accessing data from the database on a separate app
  - Attempt to read back test data that is stored

- Voice to text parsing
  - Parse sample audio recordings and ensure the audio is interpreted correctly
- Making stored data searchable
  - Search through data with known characteristics
  - Ensure all relevant data appears when those characteristics are filtered for
- Interfacing a wireless speaker with the console
  - Play a known sound file from the console and ensure it sounds correct from the speaker
- Interfacing a wireless microphone with the console
  - Use an existing console app with voice chat capabilities and use it to verify correct audio

### 2.11 PROJECT TRACKING PROCEDURES

Our project tracking procedures will be done through gitLab issues. We will have an agile board where we assign tasks and are able to comment about the issues and tasks via this platform. Gitlab issues allows us to tag items to certain milestones and goals, assign due dates, and add certain “weights” so we all know the priority. We have just started using GitLab Issues as our Project Tracking Platform, but with time we will start utilizing their features and filling up our task board. An example of our current board is shown in Figure 3.

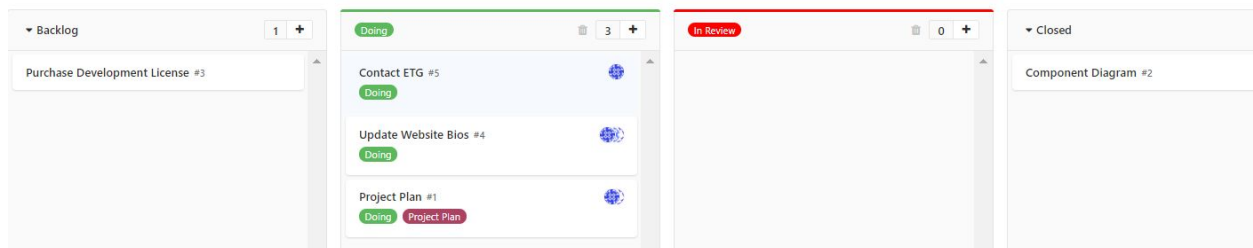


Figure 3

### 2.12 EXPECTED RESULTS AND VALIDATION

Our desired outcome of this project is a working speaker and microphone, a console app with the ability to log and transfer audio and chat communications, and a computer/smartphone app that provides a variety of tools for quickly and intuitively navigating the logged data. To ensure that the components each accomplish their task sufficiently we plan to focus on testing the functionality of each component individually before combining them into a single functioning system.

### 2.13 STANDARDS

For standards relating to privacy and security of data we will meet general standards of encryption of data being sent to and from the database and applications. However, before this product is able to go into production it will need a firewall system and further more intensive privacy and security implementations that will be handled by our client.

One thing that we've considered with ethical standards is whether it is ethical to distribute a chat record of audio conversations between one's child and potential other children or strangers. Though this is of concern, it falls under the same category as other gaming streaming services. Those who sign Xbox's privacy license are willingly consenting to a potential distribution. I also think it is worth considering that this product's main consumers will be parents of children whose main intent behind purchasing is to keep their kids safe.

### 2.13 TEST PLAN

Functional Testing:

We plan to test whether the speaker can receive a bluetooth signal and amplify it based on the volume setting. We will also test the distance the bluetooth speaker can withstand before losing signal so that we can record that for our consumers. We will test whether the controller extension can pair with the speaker and test transmitting an audio signal and receiving microphone data through the controller extension. We also plan on manually sending a message on the console to test whether it can be read into the application. We will test dumping all data on app into database. We plan to use unit tests on the database server to make sure querying works. We also plan to use unit testing to test the keyword functionality on the mobile application.

Non Functional Testing:

We will test the best location to put the speaker and the switches (for easy usability). We will test different ways to set up the controller extension so that it fits in the space between the controller handles. We will use unit tests to test our login feature to guarantee security. We will test compatibility between database and both applications and usability will be tested by making sure that the app fully runs in the background (we will use a lot of corner cases to test this with a simple gaming application). We will also do stress tests to make sure the console application can send a continuous stream of data.

## 3 Project Timeline, Estimated Resources, and Challenges

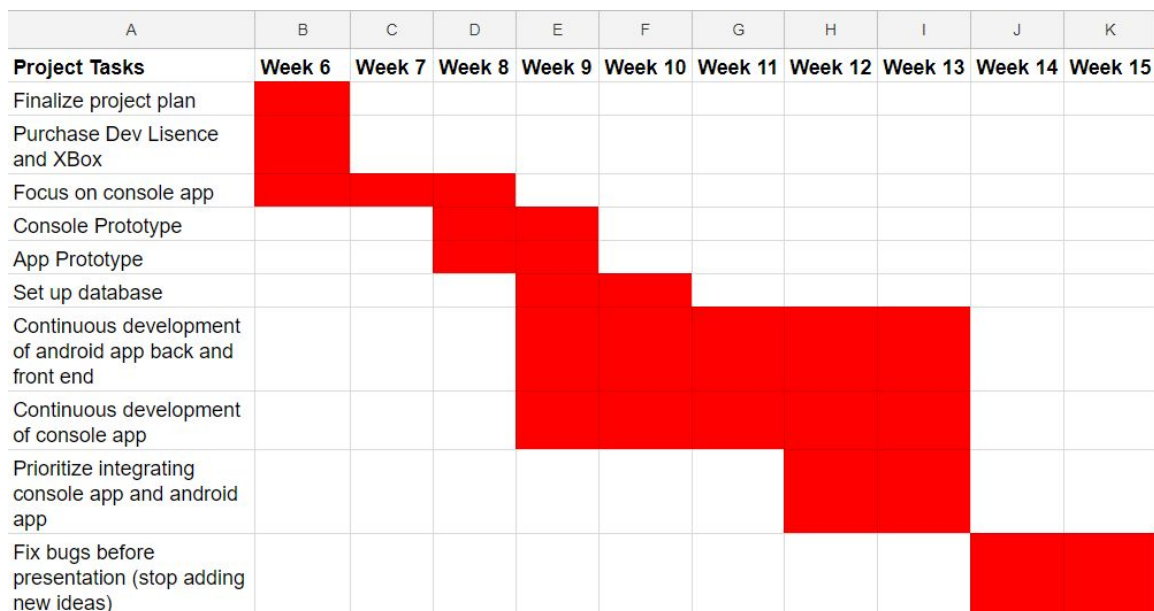
### 3.1 PROJECT TIMELINE

<b>Week 4</b> 1/27 - 2/2	Create schedule, come up with unique ideas, write down research notes on voice recognition api, what is accessible in Xbox dev mode, hardware, systems that currently do something similar
--------------------------------	--

<b>Week 5</b> 2/3 - 2/9	Practice using git, get our “board of tasks” up and running, finalize development first draft decisions (this matches up with our task to finish the project plan). (UML Diagram - component), write requirements (functional and nonfunctional)
<b>Week 6</b> 2/10 - 2/16	Continue testing software implementation, and researching the same topics as in week 4.
<b>Week 7</b> 2/17 - 2/23	Begin development testing on console while simultaneously continuing research
<b>Week 8</b> 2/24 - 3/2	Continue development testing on console while simultaneously continuing research
<b>Week 9</b> 3/3 - 3/9	Thomas finishes database connection from xbox console Jenn, Keng-Yik focus on getting audio data Jacob focus on extracting text data Matthew focus on speech to text
<b>Week 10</b> 3/10 - 3/16	Spring Break - continue development when possible Thomas finishes database connection from xbox console Jenn, Keng-Yik focus on getting audio data Jacob focus on extracting text data Matthew focus on speech to text
<b>Week 11</b> 3/17 - 3/23	Thomas and Jacob will have a constant flow of information from mobile to console and console to mobile app via the database Matthew - Speech to text Keng-yik - Getting the audio files Jenn - Working on front-end, making the mobile application usable
<b>Week 12</b> 3/24 - 3/30	Finish Speech to text Finish gathering audio Thomas - help with console application (work on making a simulation game app and utilize forums and calling microsoft) Jenn - focus on keywords
<b>Week 13</b> 3/31 - 4/6	Integrate Keng-yik and Matthew’s work on console application Thomas - help with console application (work on making a simulation game app and utilize forums and calling microsoft)

<b>Week 14</b> 4/7 - 4/13	<p>In complete development mode at this point. Working on both mobile and console app. Team will stop introducing new work/ features and focus on producing a proof of concept and fixing bugs.</p> <p>Thomas, Keng-Yik and Matthew make things clean on the xbox app, introduce graphics, look into how it can be used as a background application.</p> <p>Console application team will start testing with the console applications</p> <p>Mobile Application will test keywords</p>
<b>Week 15</b> 4/14 - 4/20	<p>Focus on producing a proof of concept and fixing bugs and preparing for presentation</p>
<b>Week 16</b> 4/21 - 4/27	<p>Dead Week - presentation. At this point we want to have a proof of concept that consists of at basic: showing that we can get data from xbox and receive data from our speaker/app side, potentially without integration. At most we would have a general speaker, arduino, with a bread board for button/control usage, and simple functionality on XBox.</p>

**Table 1**



**Figure 4 - Semester 1 Gaant Chart**

Project Tasks	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14
Android app is receiving text from console application	█	█	█											
Create and improve on Hardware Prototype			█	█	█									
Android app's keyword feature is successfully working			█	█	█									
Work on console application being hidden to child user				█	█	█	█	█						
Implement a way to have context to words (tags or audio files)					█	█	█							
Focus on testing								█	█	█	█	█		
Spend extra time for new ideas (implement iOS app or website)								█	█	█	█	█	█	
Prepare for final presentation													█	█

Figure 5 - Semester 2 Gaant Chart

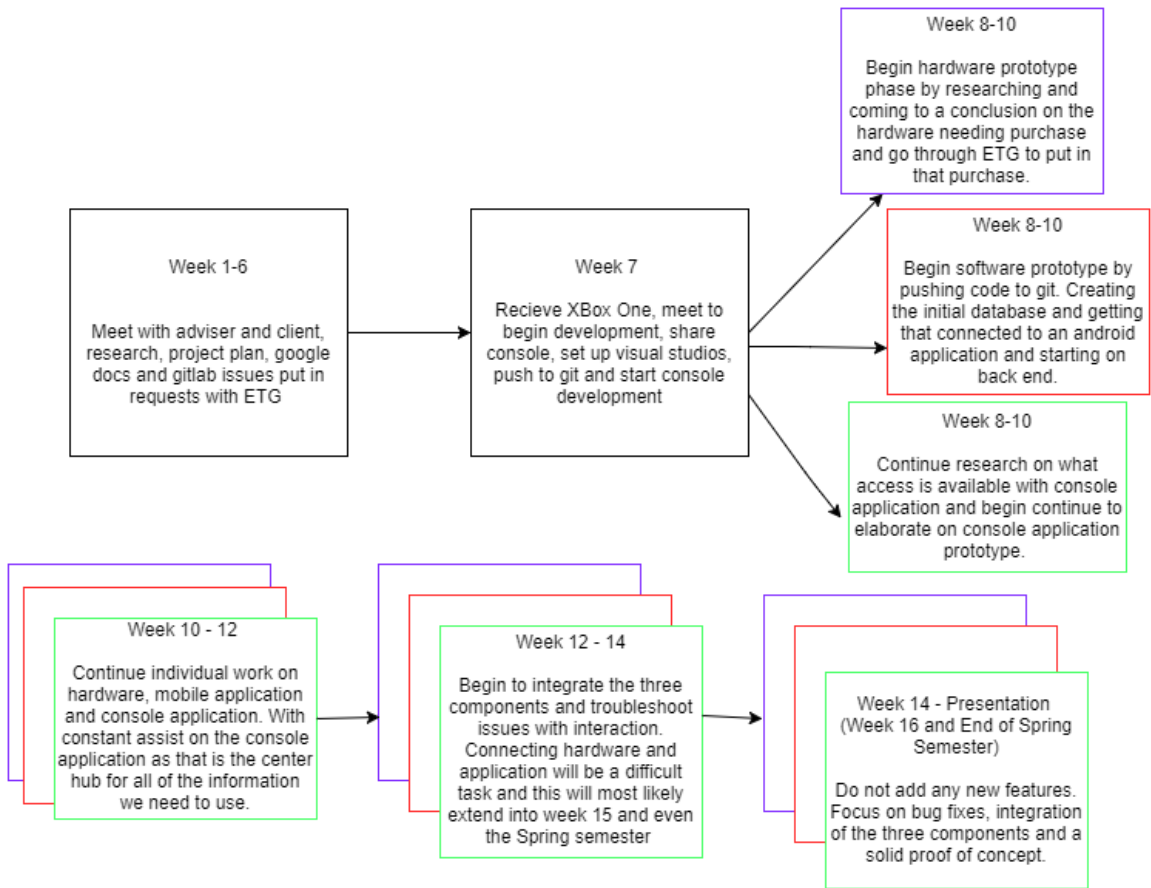


Figure 6 - Timeline Diagram for Spring Semester



We have made a rough draft of our Gantt chart and WBS. They currently contain our main deliverables with some breakdown.

Our main goal with our project timeline is to get interfacing with the consoles to work first then extend to app, site, and hardware with an emphasis on starting back end first. Then after we are fully structured, focusing more on the frontend.

Our timeline breakdown focuses on the gaming console first because without knowing our accesses with the Xbox Development system, we will not be able to know the scope of our abilities with our app as a whole. After that, we want to split our team apart in working on the gaming console as well as the mobile application to capitalize on our strengths.

We decided two weeks before our first semester presentation is when we should stop adding to our workload and focusing on bug fixes and making things look cleaner and presentable. We also decided that our bare minimum proof of concept would be showing our ability to receive and send data from the xbox and the same with our speaker application, even if they aren't integrated together quite well. We hope for more to be accomplished but this minimum goal will at least guarantee us a good start to finish implementation by the end of second semester.

In the Fall semester we plan to implement as much as we possibly can. Our first goal will be to get fully functioning console and mobile application that is user friendly and has modular code. From there we will extend our mobile application to a website and potentially iOS. As well as look into extending our Xbox system platform to others. However, those are additional tasks that will be added once we get our bare minimum baseline down.

### 3.2 FEASIBILITY ASSESSMENT

We will have a speaker and microphone that works as intended and a mobile app that will work with the speaker. The main challenge is making sure we can gather the audio data from the console application. We have done thorough research but cannot find much documentation so we will need to start reverse engineering this so that it can be confirmed feasible. We can confirm feasibility in getting a background application running on the Xbox One development platform and that it can connect to our database using SQLite. We also can confirm feasibility of creating a mobile app that will connect and receive and send data to and from the database.

Any feasibility issues we are still working to find work-arounds with are revolved around the console application. There is not much documentation for UWP and though this helps prove that what we are trying to do is new and innovative, it is slow moving to find documentation on how to access audio data and convert it from speech to text. However, from our research that is already completed (found at

<https://docs.microsoft.com/en-us/windows/uwp/> and it's following paths) we do believe that completing our console application goals will be feasible.

### 3.3 PERSONNEL EFFORT REQUIREMENTS

Not fully implemented.

Create speaker
Create microphone
Group chat interfacing
party chat interfacing
Create Database structure
Develop DB interfacing
Create website front end
Create website back end
Create app front end
Create app back end
connect app to DB
connect site to DB
connect DB to console
connect Hardware to console

### 3.4 OTHER RESOURCE REQUIREMENTS

Not applicable.

### 3.5 FINANCIAL REQUIREMENTS

We will potentially need an XBox One to develop on - **\$200 - \$300**

XBox Development License - **\$19.99**

Speaker/Microphone - **\$60-\$80**

## 4 Closure Materials

### 4.1 CONCLUSION

Our goal for Hidden Guardian is to create a platform where parents can have their minds at ease about their child's video game play. We plan to have a microphone/speaker that replaces the gamers headset, a console application and a mobile application. Between the communication of these three components the parental user will be able to listen for keywords, listen to chats, and effectively monitor their child's video game play on a user-friendly mobile app.

## 4.2 REFERENCES

DIY Bluetooth Speaker/Voice Recorder

<http://www.instructables.com/id/DIY-Bluetooth-Speaker-PartyBar/>

<http://www.instructables.com/id/Make-Your-Own-Spy-Bug-Arduino-Voice-Recorder/>

Android Application

<https://www.androidauthority.com/android-app-development-complete-beginners-658469/v>

<http://www.vogella.com/tutorials/MySQLJava/article.html>

Financial Resources

<https://www.gamestop.com/browse/consoles/xbox-one?nav=28-xuo,13ffff2412-1e0>

<https://www.xbox.com/en-US/developers>

Speaker Research

<https://www.parts-express.com/cat/midrange-midbass-drivers-full-range-speakers/16>

<https://www.digikey.com/>

IEEE Ethical Code

<https://www.ieee.org/about/corporate/governance/p7-8.html>

Microsoft Windows UWP Forums

<https://docs.microsoft.com/en-us/windows/uwp/>