VIRA | Virtual Insight into Real Athletics

Advisor

Dr. Diane Rover

Team Members

Katie Perkins	*Team Lead*
Nate Irmiter	*Stakeholder Communicator*
Bailey Righi	*Product Manager*
Willem Paul	*Standardization Manager*
Caroline Rankin	*Meeting Facilitator*

Email

sdmay20-33@iastate.edu

Website

sdmay20-33.sd.ece.iastate.edu

Revised

Friday, November 1, 2019
Version 2

Executive Summary

Development Standards and Practices Used

Software Development Practices

- Agile development process
- High cohesion, low coupling
- Modular code for maintainability and extensibility
- Good code formatting and documentation

Engineering Standards

- Unit testing
- Integration testing
- Acceptance testing
- Hardware I/O testing
- Documentation of design decisions, including UI and architecture
- Documentation of technologies and rationale behind them
- IEEE Standards
- Oculus application development standards

Summary of Requirements

- The user can log in and navigate to a guided special test module or injury evaluation quiz module
- The user can view the virtual limb in skin, muscle, or tendon modes to aid them in identifying what muscles and tendons lie underneath the skin
- The user can see their performance data from the modules they have worked on in the form of a statistics dashboard

Applicable Courses from Iowa State University Curriculum

We have all taken different courses, so every group members' experience is different. In general, however, the list of applicable courses includes (but is not limited to):

S E/CPR E 185	*Problem Solving in Software Engineering*
COM S 227	*Object-Oriented Programming*
COM S 228	*Introduction to Data Structures*
S E/COM S 309	*Software Development Practices*
S E 319	*Construction of User Interfaces*
S E 329	*Software Project Management*
S E 339	*Software Architecture and Design*
S E/COM S 362	*Object-Oriented Analysis and Design*

New Skills/Knowledge Acquired Not Taught in Courses

- Unity development environment
- Virtual reality application design
- Computer graphics concepts and terminology
- Knowledge of human anatomy and medical diagnosis procedures
- Better understanding of software development processes
- Creation of human models with MakeHuman
- Blender model development and animation

Table of Contents

List of Figures

List of Tables

1 | Introduction

1.1 | Acknowledgement

We would like to acknowledge the significant help of Joe Hubbard, an ISU athletic trainer who is our primary source of information related to athletic injury examinations. We would also like to thank Dr. Diane Rover for her guidance and expertise throughout the duration of our Senior Design project.

1.2 | Problem and Project Statement

There is nothing as good as hands-on, in-person, real-time experience in performing the tasks of an athletic trainer. When students in the athletic training program have made it to the point at which they are able to work in the training room with student athletes, their time in the training room becomes very valuable. However, while they are learning, a full-time athletic trainer must watch them and guide them through what they should do. This makes it very difficult for athletes to get the attention they need when the training room gets busy. On top of that, there are many traumatic injuries that a student will never have experience with until they are working full-time in the field.

A virtual reality training program would allow athletic training students to get the extra experience and feedback they need without having to interrupt or wait for a full-time athletic trainer to finish treating another athlete. We propose a virtual reality program that consists of a set of training modules that walk an athletic training student through injury evaluation and diagnosis scenarios.

Our goals are as follows:

1. Create a virtual reality (VR) training program that provides student athletic trainers with more experience outside of class and work,
2. Give student athletic trainers realistic simulations of assessing injured athletes without performing real (and critical) examinations, and
3. Allow for easy future extension of VIRA by using a modular design.

1.3 | Operational Environment

VIRA is primarily designed for stationary, indoor use. The VR headset we have chosen to use—an Oculus Quest—should therefore not be subjected to extreme conditions. As with any VR application, users should be aware of their surroundings at all times. The Oculus Quest has a built-in feature called the "Guardian", a virtual box around the user that alerts them when they step or reach outside of its bounds. This further protects the user and the headset.

1.4 | Requirements

Functional Requirements

Before the Module

- The user can select a user profile when starting the application
- The user can select a mode (guided or quiz)
- The user can select a specific module
  - Guided Mode modules will focus on special tests
  - Quiz Mode modules will focus on injury evaluation scenarios
  - Modules will cover traumatic injuries not often seen in training

During the Module

- The user can toggle view modes to see the skin, muscles, or tendons of the virtual limb
- The user can select a muscle or tendon and see its name
- The user can manipulate the limb in a realistic manner
- The application can guide the user through a special test
- The application can assess the user on an injury evaluation quiz
  - The user loses points for the following:
    - assuming the injury before evaluating,
    - palpating the wrong spot(s) or not palpating a certain spot,
    - doing incorrect or unnecessary tasks,
    - giving an incorrect diagnosis,
    - completing poor differential diagnoses, or
    - skipping steps of an evaluation
- The user can view the angle at which they are bending a joint when evaluating range or motion
- The user can exit the module and return to it at any time

After the Module

- The user can exit the module and return to it at any time
- The user can see their progress on a particular module over time
- The user can restart the module or return to the menu

Economic Requirements

- We shall not exceed the funding provided to us by the Department of Electrical and Computer Engineering (hereinafter "the Department" or "ECpE")

Environmental Requirements

- The application shall work in real-time and interact directly with the user

- The application shall accurately interpret the motion of the user's hands and realistically translate that motion to the virtual display
- The user must remain within the Guardian that is free of obstacles and other people while VIRA is in use

1.5 | Intended Users and Uses

VIRA is intended for use by student athletic trainers to practice injury evaluation and diagnosis. These are students who wish to further develop their clinical skills and gain experience working with traumatic injuries they wouldn't typically see as a student worker. Our intended customer is the Department of Sports Medicine, as they will be assisting us with module testing and will evaluate our accuracy. Our goal is to create an educational program consisting of modules that will augment students' studies and on-the-job experience.

1.6 | Assumptions and Limitations

Assumptions

For this project, we make the following assumptions:

- The Oculus Quest will allow a great enough range of motion to simulate interaction with an athlete,
- The Oculus Quest has a sufficient level of hand-tracking capabilities for VIRA,
- The maximum number of users will be at one at any given time,
- Multiple users will share the same device,
- The end product will not be used by anyone but athletic training students,
- The users have little experience in the specific diagnosis module(s) we create,
- The users have a sufficient understanding of English, and
- The users are physically capable of operating an Oculus Quest.

Limitations

As VIRA is a student-proposed project and due to complications acquiring hardware, our limitations are as follows:

- We are using an Oculus Quest, restricting us to VR technology,
- We are operating on a small budget from the Department,
- The number of VIRA modules created will be limited by development time and any schedule setbacks, and
- The quality of the graphics will be limited by the version of Unity we are able to use and the functionalities available to us in Unity as well as the quality of any other graphics programs we choose to utilize.

1.7 | Expected End Product Deliverables

VR Training Module

The final product will consist of at least one VR training module by the end of the school year. This will include both the standard training mode, in which the user is guided through a specific injury evaluation and provided with feedback (see below), as well as a quiz mode, in which the user will be assessed on their performance. The specific skills targeted by these modules are special tests, palpation, and injury evaluation or diagnosis.

Feedback Dashboard

The module(s) will implement a feedback component. This system will monitor the student athletic trainer's progress as they work through a module and will assess their technique, accuracy, etc. and provide them feedback after they complete the module. This information will be connected to a specific user's account so that user will be able to see what they did well and what they still need to improve upon.

2 | Specifications and Analysis

2.1 | Proposed Design

The program will be written for an Oculus Quest headset and controllers using Unity and will consist of modules that address scenarios that an athletic training student does not typically see while working in a clinic. The user will work through their chosen module alone, either standing still or sitting down. The module will consist of either a guided special test or a quiz evaluation of a virtual limb that is associated with the chosen scenario. The user will be able to manipulate the limb and gather more information by seeing different views of it, such as the muscle layer or tendon layer. The latter two enable further practice in palpation, which is key for injury evaluation.

For a guided special test, the program will prompt the user with visual cues to complete the evaluation process. For a quiz evaluation, there will be no guides for the reader, and they must manipulate the limb and make decisions about the options presented to them on their own. At the end of the training module, a virtual dashboard will appear to give the trainer feedback on how they performed during the training. It will also keep a history of how they have done on that particular module in the past so they can see the progress they are making towards doing it correctly. This project is intended to be scalable due to how unfamiliar the team is with augmented reality. We will begin with a single training module, then create others if time allows.

To test our modules, we plan on following IEEE 29119 standards for testing processes and techniques. We plan to utilize GitLab's continuous integration capabilities and in doing so, we will be organizing our test plan development, documenting, and discussing our plan. Additionally, many aspects of identifying, estimating, and handling risks have been a part of our design process, so we have been and will continue to take these steps into consideration. As for testing techniques, we plan to design our modules in a manner that heavily controls the program flow since VIRA is based on strictly defined injury tests. As such, there are some IEEE standards we plan to follow [1]. Most of the structure-based techniques will have a direct application to our modules and others will be very useful, such as boundary value analysis—we want to ensure that our models move correctly and have bounds on their movement—and state transition testing—we want to ensure that specific program or modules states flow to their correct successors.

Our project must also be compliant with the Oculus Quest Virtual Reality Check (VRC) Guidelines [2]. These detail the packaging, audio, performance, functional, and security requirements, among others. All applications submitted to the Oculus Store must comply with these requirements and we intend to develop VIRA as if we were submitting it to the Store.

2.2 | Design Analysis

We collectively decided that our process would follow the design thinking process presented in class. *Figure 1* (right) shows a diagram of our process, which we will repeat in iterations. In some ways, it has worked well so far, in others, we have found some flaws. As we are still in the early stages of the design process, we have yet to prototype our modules.

One issue we have discovered is that a complication in one area of the project has the potential to complicate another area. For instance, we met with an athletic trainer with the idea of creating an AR application, but due to difficulties acquiring a Microsoft HoloLens, we will instead be creating a VR application. This will have an impact on the potential functionality of the modules and as a result, on the information we need to gather from the trainer. It will likely manifest once we start development as we reconsider some of our planned functionalities. Currently, we do not plan to change our design process.

Figure 1 - Design Thinking Process
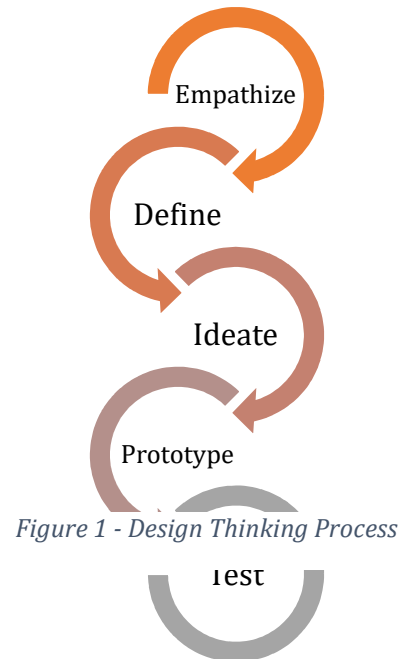
2.3 | Development Process

We will use an Agile process to develop VIRA. It is crucial that we receive lots of feedback from the end-users to ensure that we create exactly what they need. Working iteratively will allow us to check in with them regularly and will give us a good structure for our own time management. An Agile process will also ensure that we work collaboratively to have a shippable product at the end of each sprint, rather than working individually and hoping that one member's code will integrate with another's.

2.4 | Design Plan

Figure 2 (next page) shows our system block diagram, which we will use to guide our design. This diagram shows some of the use cases for our system as how different components of the system will communicate with each other.
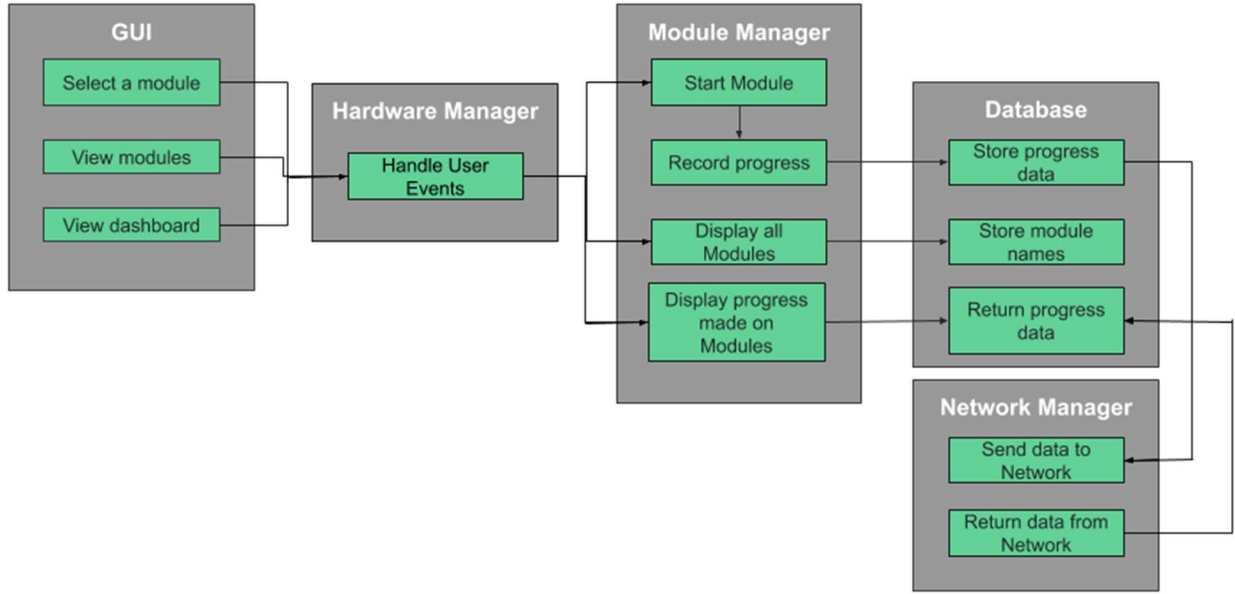
Figure 2 - VIRA Design Block Diagram

3 | Statement of Work

3.1 | Previous Work and Literature

There are several existing simulations for surgery (such as Touch Surgery [3]) and athletic skills training, but we did not find any specifically designed for athletic trainers. These similar products are able to map the body and calculate its movements, which are aspects that we will apply to VIRA.

3.2 | Technology Considerations

During the proposal phase, we were considering two different routes for the projects: augmented reality (AR) or virtual reality (VR). We initially chose AR as the preferred technology as it would allow the user to practice injury evaluation with a real partner. They would be able to see another person and manipulate their joints. Unfortunately, after following several possible leads, we were unable to acquire a Microsoft HoloLens. As a result, we are instead using an Oculus Quest, a VR headset. VR technology, while still useful, may not provide the best experience for the user, but after some testing, we have determined that it should still provide us with sufficient capabilities to successfully implement VIRA. Despite this, there are tasks that an athletic trainer would perform while testing and diagnosing a patient that are difficult or impossible to simulate in a VR environment. For instance, testing a joint's range of motion would require the trainer to provide resistance to the limb. While there is no way to recreate this force in VR, we can simulate it using the Quest controllers' haptic feedback. This will still allow the user to learn the steps required to perform a certain test.

3.3 | Task Decomposition

1. Create user profile selection
    a. Create screen sketches
    b. Develop basic graphics from screen sketches in Unity
    c. Code transition from user selection screen to modules screen
        i. Includes interaction with database
2. Create module selection
    a. Create screen sketches
    b. Develop basic graphics from screen sketches in Unity
    c. Code transition from module selection screen to module
3. Develop models for the chosen limb(s)
    a. Create different view modes
        i. Create skin view
        ii. Create muscle view

        iii.   Create tendon/ligament view

4. Create guided special test module
    a. Establish narration/guided content with help from athletic trainer
    b. Establish statistics to collect for performance dashboard
    c. Create screen sketches
    d. Develop basic graphics from screen sketches in Unity
    e. Code text and visuals for user guidance
        i. Limb movement indicators to guide user to move the limb in a certain way
        ii. Button selection for next step
    f. Code detection that the user has correctly followed the guidance
        i. Detect if the user has moved the limb to the correct position as indicated (if applicable)
        ii. Code redirect in case the use has done something wrong (may include more specific guidance or more explanations)
        iii. Code statistics collection and store information in the database
    g. Code muscle selection and corresponding identification
    h. Code tendon/ligament selection and corresponding identification
    i. Code transition from module completion to performance dashboard
    j. Choose which sound clips to use and where to use them
5. Create quiz module
    a. Establish narration with help from an athletic trainer
    b. Establish statistics to collect for performance dashboard
    c. Create screen sketches
    d. Develop basic graphics from screen sketches in Unity
    e. Code text and visuals for user guidance
        i. Decision tree
        ii. Button selections for next step or differential diagnosis (The user will be asked to choose what to do next or what diagnosis they want to make from a few options)
    f. Code detection that the user has correctly followed the guidance
        i. Detect if the user has moved the limb to the correct position as indicated (if applicable)
        ii. Detect is the user has chosen the next step incorrectly
        iii. Code statistics collection and store information in the database
    g. Code transition from module completion to feedback dashboard
    h. Choose which sound clips to use and where to use them
6. Create user statistics dashboard
    a. Establish which statistics will be used or not based on the given scenario
    b. Create screen sketches

    c. Develop basic graphics from screen sketches in Unity
    d. Code selection of statistics to use based on scenario
    e. Code user-specific data retrieval from database
    f. Code necessary calculations to display the correct statistics
    g. Code statistics display
7. Set up database
    a. Design database schema
    b. Create database
8. Establish connection between VIRA and database
    a. Create Spring backend
9. Enhance graphics quality in Unity

3.4 | Possible Risks and Risk Management

There are a few different risks inherent to our project. First, due to the fact that we were forced to switch to VR from AR, there is the risk that VIRA will not function as we initially envisioned. This means that we may be compromising on some implementation decisions to make VIRA work properly. Similarly, VIRA is designed for athletic trainers and will thus require medical/anatomical accuracy that none of us are too familiar with. In order to mitigate the risk of our module not working as the client intends, we need to work closely with them.

Another risk is the fact that VIRA is a student-proposed project. We do not have corporate funding, that is, we have a limited budget. Finally, as mentioned previously, none of us has experience in VR development, which introduces a level of risk regarding what we want to achieve versus what we can achieve.

3.5 | Proposed Project Milestones and Evaluation Criteria

Our milestones are the top-level items in our outline in Section 3.3. Our evaluation criteria are as follows:

- Create user profile selection: The user can select a profile under which to record their activities
- Create module selection: The user can select a module to enter
- Develop models for the chosen limb(s): The user can view the limb(s) in skin, muscle, and tendon modes
- Create guided special test module: The user can navigate through a module and perform the tasks necessary to complete it, learning how to implement the special test they have chosen along the way
- Create quiz module: The user can take a quiz and be evaluated on the tasks they perform without step-by-step guidance
- Create user statistics dashboard: The user can view statistics on their performance in the modules and quizzes they have completed
- Set up database: Data can be stored and retrieved properly
- Establish connection between VIRA and database: VIRA can correctly access the database for user profile selection and displaying the feedback dashboard
- Enhance graphics quality in Unity: Graphics are smooth and realistic enough that the user can easily determine what is being displayed

3.6 | Project Tracking Procedures

We are using GitLab as our code repository and to track progress during the project. High-level project progress will be tracked through GitLab via Milestones (see Section 3.5) and specific development tasks (See Section 3.3) will be represented as Issues, most of which will contribute to a certain Milestone. Unity also has version control available (Perforce, PlasticSCM), which we will decide upon as a team once development has begun.

3.7 | Expected Results and Validation

Our desired outcome is that we have at least one fully functional module that helps incorporate technology into an athletic training environment. We will know that VIRA works at a high-level through user acceptance testing. Once we begin the second semester, we plan to take the product to the Athletic Training Department at the end of each sprint to show our progress to the customers. We will use their feedback to ensure that we're working towards the correct solution. In the end, we will know that our solution is successful when it has been deemed medically accurate and effective by an athletic trainer.

4 | Project Timeline, Estimated Resources, and Challenges

4.1 | Project Timeline

All our tasks listed in Section 3.3 make up our Gantt chart (See Appendix 1). For August, September, and the first half of October, we focused on researching AR/VR software and hardware as well as similar projects that have been done elsewhere. We also met with an athletic trainer to gather the necessary steps for the training module and the expectations for the UI. From there, we created our requirements and design documentation. After that, we began learning Unity and ordered an Oculus Quest (after trying, unsuccessfully, to get a Microsoft HoloLens).

For the second half of October up until dead week of the Fall semester, we intend to design and set up our database, as well as the user profile selection and module selection screens. This is also the point where we will set up the connection to the database, likely through a Spring backend. After that, we will begin designing and coding our first guided special test module. This will include getting the necessary medical information from the athletic trainer and designing the narrative and screen sketches of the module. We will not do anything except make simple tweaks over Winter Break.

When school starts again for the Spring semester, we will focus on user interaction and user feedback. We will work on the code that determines whether the user's actions will be perceived as correct or incorrect and then gives feedback or collects data for the statistics dashboard. After that, we will design and code the statistics dashboard that will be present at the end of the module. This takes us into the first half of March, with the rest of the Spring semester either left as buffer time or time to create more modules.

4.2 | Feasibility Assessment

We expect to complete two fully functional modules completed by the end of the school year: a guided special test module and a quiz module. This is a reasonable goal, and anything less would not be useful to the client. Our biggest challenge is the team's inexperience with VR development. We anticipate some challenges in finding suitable tutorials for certain things such as coding for the Oculus Quest, as well as difficulty working with the graphics. Because none of us have worked with VR before, we have added in a rather large buffer of time at the end of the spring semester to accommodate schedule changes if development goes slower than expected.

4.3 | Personal Effort Requirements

Table 1 below lists the tasks we have done and expect to do for this project. The numbers provided are very rough estimates and will vary from person to person and from week to week. Following are descriptions of the tasks.

Table 1 - Individual Tasks

Task	Estimated Time (Hours Per Week)
Gathering requirements	2 (per sprint)
Researching	Variable
Acquiring hardware	4 (once)
Setting up development environment	< 0.5 (once)
Development	Variable
Testing	Variable
Writing design documents/presentations	3 – 5

Gathering Requirements

The bulk of this task involves meeting with athletic trainers to discuss the next steps for the sprint. The time required for this will vary based on what we plan to accomplish that sprint and how familiar we are with the material required to do those things.

Researching

We will need to research on a continual basis; from researching which hardware to choose to researching the best means of adding a model in Unity, we will constantly be researching to ensure that we do not reinvent the wheel.

Acquiring Hardware

This will only need to be done once. We have successfully acquired our Oculus Quest and should not need any further hardware.

Setting up Development Environment

This is also a one-time task. Once we have our Oculus Quest, we can all make sure that our development environments are set up similarly. Once we all have Unity properly set up and are all connected to our GitLab repository, we should not have to do anything further for this task.

Development

This is, of course, the bulk of our project's work. Most of our time should be spent working on the module and communicating what each of us has accomplished. Since we have spent most of our time as a team sorting out which technologies we will be using and working on presentations and design documents, we are uncertain of how much time we will need to put into developing a VR application.

Testing

This encompassed unit, system, regression, and acceptance testing and is part of the development process. As we have yet to begin development, we are unsure of how much time will be spent testing.

Writing Design Documents and Presentations

This includes working on all project management overhead, design documents, and presentations that are required for this course. Based on our time spent thus far, we estimate that this will take roughly three to five hours per week.

4.4 | Other Resource Requirements

The project will require an Oculus Quest VR headset, controllers, and a carrying case. Additionally, we will require time to meet with athletic trainers to discuss requirements, allow them to test the program, and receive feedback from them.

4.5 | Financial Requirements

As this project is student-proposed, we do not have company funding like some other projects might. Thus, we are financially restricted to what the Department can provide. We used $500 of our budget on the Oculus Quest. Additionally, we would like to have licenses to the Pro version of the Unity engine, which should be free for educational institutions.

5 | Testing and Implementation

5.1 | Interface Specifications

VIRA will be run solely on the Oculus Quest, so we will not need any software/hardware interface other than a USB cable.

5.2 | Hardware and Software

We will not need any additional hardware or software for testing VIRA. The Unity development environment provides us with unit testing and debugging capabilities. It is useful for modeling how the module will run on the Quest as well as making it obvious when there are visual bugs. Testing on the Quest depends solely on access to the Senior Design lab in Coover Hall, where we are storing it. Acceptance testing will be completed by the end-users.

5.3 | Functional Testing

We plan to use Unity to perform unit and module testing. Our integration testing will most likely be performed by an automated GitLab CI script that will perform a set of tests every time a team member pushes changes. System and acceptance testing will be completed by the team and the end-users.

5.4 | Non-Functional Testing

To test usability, we will analyze how first-time users navigate through VIRA to assess its intuitiveness. For performance testing, we will monitor running speed, response time, and battery consumption of repeated uses to ensure that VIRA runs efficiently on the Oculus Quest. Since VIRA will be developed specifically for the Oculus Quest, we do not expect any compatibility issues. Our main security concern regards the user accounts. We will ensure security in our database and will minimize network requests to reduce the chances that information is stolen.

5.5 | Process

As we have not yet begun development, we have not begun any sort of testing.

5.6 | Results

As we have not yet begun development, we have not begun any sort of testing.

6 | Closing Material

6.1 | Conclusion

So far, we have worked toward designing our project, gathering and honing our requirements, and deciding upon our technologies. We have also done some work using MakeHuman and Blender to experiment with creating realistic human models in Unity. Our goals are to:

1. develop at least two complete VIRA module,
2. design a modular product to allow for easy extensibility, and
3. learn about VR development.

We have decided upon the Oculus Quest as our hardware because it is the best option for our budget. We have chosen Unity as our development environment because it is an industry standard for developing 3D applications such as ours.

6.2 | References

[1] "IEEE Announces Virtual Reality (VR) and Augmented Reality (AR) Standards Projects In Advance of Participation at Augmented World Expo," *IEEE Standards Association*, 09-May-2017. [Online]. Available: https://standards.ieee.org/news/2017/ieee_p2408.html. [Accessed: 31-Oct-2019].

[2] "Quest Virtual Reality Check (VRC) Guidelines," *Oculus Developer Center*. [Online]. Available: https://developer.oculus.com/distribute/latest/concepts/publish-quest-req/. [Accessed: 31-Oct-2019].

[3] "Touch Surgery Simulations," *touchsurgery.com*. [Online]. Available: https://www.touchsurgery.com/simulations. [Accessed: 05-Oct-2019].

Appendices

Appendix 1 | Project Gantt Chart (Section 4.1)

Activity	Resource	Status
Make Project Plan		
Design Architecture & Ot...		
Create User Profile Sele...		
Create screen sketches		
Develop basic graphics f...		
Code transition from sel...		
Create Module Selection		
Create screen sketches		
Develop basic graphics f...		
Code transition from sel...		
Create Guided Module		
Establish narration/guide...		
Establish statistics to be ...		
Create screen sketches		
▾ Develop basic graphics f...		
Develop models for the l		
▸ Create different modes		
▸ Code text and other visu		
▾ Code detection that use		
Detect if user has mov		

Activity	Resource	Status	2019									November 2019																										
			18	21	22	23	24	25	28	29	30	31	01	04	05	06	07	08	11	12	13	14	15	18	19	20	21	22	25	26	27	28	29	02	03	04	05	06	09	10	11

Code module completior
Choose which sound clip

Create Quiz Module ▼

Establish narration (what...
Establish statistics to be ...
Create screen sketches
▼ Develop basic graphics f...
   Develop models for the l
   ▶ Create different modes
   ▼ Code text and other vis
      Decision tree
      Button selection for n
   ▼ Code detection that use
      Detect if user has mo
      Detect if user has cho
      Code collection of sta
   Code module completior
   Choose which sound clip

Create User Statistics D... ▼

Activity	Resource	Status	2019 November 2019

Activity	Resource	Status
Detect if user has mo...		
Detect if user has cho...		
Code collection of sta...		
Code module completio...		
Choose which sound cli...		
Create User Statistics D... ▼		
Establish which statistic...		
Create screen sketches		
Develop basic graphics f...		
Code selection of statisti...		
Code getting data from d...		
Code calculations neede...		
Code displaying the stat...		
Set Up Database ▼		
Create database archite...		
Create database		
Create tables		
Connect Unity and Data... ▼		
Create a Spring backend...		
Enhance Graphics Quality ▶		

Activity	Resource	Status	January 2020																February 2020															
			09	10	13	14	15	16	17	20	21	22	23	24	27	28	29	30	31	03	04	05	06	07	10	11	12	13	14	17	18	19	20	21
Create Guided Module ▼																																		
Establish narration/guide…																																		
Establish statistics to be …																																		
Create screen sketches																																		
▼ Develop basic graphics f…																																		
Develop models for the l																																		
▸ **Create different modes**																																		
▸ **Code text and other visi**																																		
▼ **Code detection that use**																																		
Detect if user has mov																																		
Detect if user has cho																																		
Code redirect in case																																		
Code collection of sta																																		
Code user muscle select																																		
Code user tendon select																																		
Code module completion																																		
Choose which sound clip																																		
Create Quiz Module ▼																																		
Establish narration (what…																																		
Establish statistics to be																																		

Activity	Resource	Status	January 2020	2020 / February 2020

2020

January 2020 · February 2020

09 10 13 14 15 16 17 20 21 22 23 24 27 28 29 30 31 03 04 05 06 07 10 11 12 13 14 17 18 19 20 21 24 25 26 27 28 02 03 04 05 06 09 10

Code text and other visi
- Decision tree
- Button selection for n
- ▼ Code detection that use
  - Detect if user has mov
  - Detect if user has cho
  - Code collection of sta
  - Code module completior
  - Choose which sound cli
- **Create User Statistics D...** ▼
- Establish which statistic...
- Create screen sketches
- Develop basic graphics f...
- Code selection of statisti...
- Code getting data from d...
- Code calculations neede...
- Code displaying the stat...
- **Set Up Database** ▼
- Create database archite...

Activity	Resource	Status	March 2020																					
			03	04	05	06	09	10	11	12	13	16	17	18	19	20	23	24	25	26	27	30	31	01 0
Detect if user has mov																								
Detect if user has cho																								
Code collection of sta																								
Code module completion																								
Choose which sound clip																								
Create User Statistics D...		▼																						
Establish which statistic...																								
Create screen sketches																								
Develop basic graphics f...																								
Code selection of statisti...																								
Code getting data from d...																								
Code calculations neede...			▓	▓	▓																			
Code displaying the stat...			▓	▓	▓																			
Set Up Database		▼																						
Create database archite...																								
Create database																								
Create tables																								
Connect Unity and Data...		▼																						
Create a Spring backend...																								
Enhance Graphics Quality		▶						▓	▓	▓	▓	▓	▓	▓	▓	▓								