Virtual Reality Physio Monitor

BME 300/200
University of Wisconsin-Madison
Department of Biomedical Engineering
October 26, 2011

Team:
Hope Marshall – *Team Leader*
Roland Pomfret – *Communicator*
Gabriel Bautista – *BSAC*
Jiaquan Yu – *BWIG*

Clients:
Professor Robert Radwin
Dr. Benjamin Mandel

Advisor:
Professor John Webster
University of Wisconsin-Madison Department of Biomedical Engineering

Table of Contents

Abstract

Nurse education currently consists of theoretical learning and practice with patient models. In the ER Trauma Bay simulation in the Cave Automatic Virtual Environment (CAVE) at the Wisconsin Institutes of Discovery, one goal is to allow nurses to practice proper skills and techniques in a realistic scenario. Professor Robert Radwin and Dr. Benjamin Mandel are working with the CAVE to produce an ER tension pneumothorax scenario. Currently, the vitals monitor in the simulation displays vitals that do not change. In order to simulate a real-world clinical environment for training purposes, a dynamic virtual monitor display system must be created for the CAVE. The monitor should display electrocardiographs (EKG's or ECG's), respiration rate, blood pressure and SpO_2 levels that respond to changing physiological conditions in the virtual ER trauma bay. In order to complete the monitor design, three alternative programming languages were evaluated. The programming languages examined were Java, MATLAB, and C++. Based on chosen design criteria, the advantages of Java significantly outweighed the advantages of the alternate programming languages. Currently, the team has implemented a user interface for the monitor, which includes EKG signals, and thoroughly researched the tension pneumothorax case to develop pseudocode for the upcoming programming tasks.

Background

The CAVE (Figure 1) is a 10 x 10 x 10 ft. cube capable of rendering 3-dimensional virtual environments. Each side of the cube has two 1080p projectors which allow a given simulation to be displayed in high-definition. Audio in the CAVE is provided by a full surround sound system. The environment is viewed by wearing a special set of 3-D glasses, specially equipped with infrared sensors, which determine the direction the user is facing and adjust the image accordingly. The user interacts with the environment through a handheld device that allows him/her to point at certain objects in the simulation and move or use the object. Simulations for the CAVE are written in the CAVELib programming language (Radwin, Interview).

Figure 1: Image of a CAVE

http://en.wikipedia.org/wiki/Cave_Automatic_Virtual_Environment

The scenario being simulated involves a patient suffering from tension pneumothorax (Figure 2), commonly known as a collapsed lung. In the simulation, the user must follow a specific set of steps in order to bring the virtual patient to a stable condition. One element of the simulation is the physio monitor (Figure 3), or vitals monitor. The current monitor in the simulation is unrealistic in that it picks static values for all vitals. A more realistic monitor is

Figure 2: Tension pneumothorax scenario

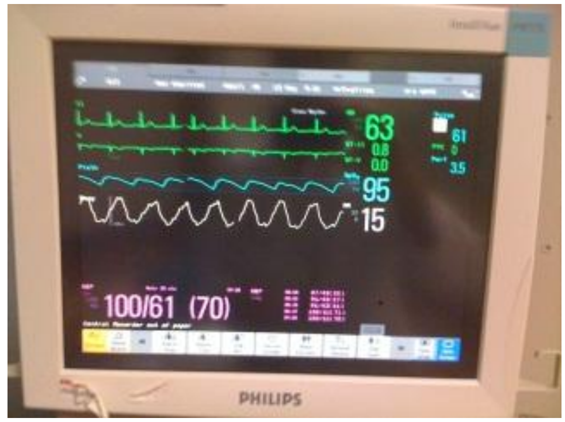


Figure 3: Philips Intellivue MP70

necessary in order to bring the simulation to the level of realism required in order to train nurses.

ER's are regularly equipped with physio monitors, although the types and brands vary widely. The clients have asked that the Philips Intellivue MP70 be used as a model for the monitor. Physio monitors can display many different readings, including but not limited to: heart rate, respiration rate, blood oxygen levels, carbon dioxide blood levels, blood pressure, and temperature.

Tension pneumothorax occurs when air accumulates in the chest cavity. This places pressure on the lungs, inhibiting their ability to expand, resulting in the patient experiencing symptoms such as shortness of breath, chest pain, fatigue, and a blue hue to the skin (Jain, Gosavi, & Jain). Air can enter the chest cavity for numerous reasons such as chest trauma from a car accident. This can also be caused by outdoor activities such as scuba diving or high-altitude climbing ("Collapsed Lung"). From a vitals perspective, heart rate and respiration rate increase, while blood pressure and blood oxygen decrease (Mandel, Interview). Tension pneumothorax is typically identified using an x-ray. However, there is usually not enough time to wait for these results in an emergency situation. If tension pneumothorax is suspected, prompt treatment with a needle and chest tube is performed, which allows air to escape from the chest cavity, returning the patient to a stable condition (Chen et al.).

Motivation

The tension pneumothorax 3-D scenario is being developed for nurse training. It is important for nurses to be trained in real cases and the CAVE is capable of simulating a realistic trauma scenario. Currently, nurse education consists of theoretical learning and practice with patient models. Neither of these methods provides an experience that is close to a real ER situation. The goal of the CAVE program is to allow nurses to practice proper skills and techniques in a realistic scenario.

In order to simulate real conditions in the ER trauma bay, the patient monitor must be displayed clearly and it should be identical to a real one both in appearance and function. Specifically, the monitor should display dynamic signals that change according to the patient's physiological conditions and nurses' reactions.

Problem Statement

The clients, Professor Robert Radwin and Dr. Benjamin Mandel, are working with the CAVE to produce an ER scenario that, in the future, will be used to train nurses. Currently, the vitals monitor in the simulation displays vitals that do not change. In order to simulate a real-world clinical environment for training purposes, a dynamic virtual monitor display system must be created for the CAVE. The monitor should display dynamic electrocardiographs (EKG's or ECG's), respiration rate, blood pressure and SpO_2 levels that respond to changing physiological conditions in the virtual ER trauma bay. These levels should also fluctuate based on general variability in humans. The virtual display must sound an alarm if it detects that the avatar patient's vitals are passing a dangerous threshold.

Client Requirements

The final monitor design must resemble a monitor used to train medical students. In the tension pneumothorax case, the monitor must simulate vitals that respond to the series of steps detailed by the flow chart provided by the clients. The monitor must display at least four dynamic vitals: heart rate, blood pressure, SpO_2, and respiration rate. The signals should look like the filtered signals commonly displayed by modern monitors in the ER. The vitals displayed must change based on user action, but they should also fluctuate based on general variance among the human race. In addition, the program must be flexible to allow for modification and implementation of other patient ailments in the future. The design should be based on the Philips Intellivue MP70. The program should also have the option for the user to enter vitals manually. Since this is a training program, a system should be in place to notify the user if the patient's vitals are declining at dangerous rates. This is meant to attract the user's attention to the monitor in order to properly address the situation.

Existing Designs

There are programs in existence that simulate patient vitals, but many of these programs are specific to certain scenarios. For example, the ANGIO Mentor product line from Simbionix is a series of hands-on simulators for cardiovascular surgeries. The user performs actions with their hands, such as inserting an endovascular balloon, that are reflected on a computer generated image of the heart. Simulated patient vitals appear on a large screen next to this image ("ANGIO Mentor").

Anesoft, a medical software company, has numerous programs available that simulate various medical emergencies in areas such as anesthesiology, Advanced Cardiovascular Life Support, and even bioterrorism (Anesoft). These programs include patient vitals simulators tied to the specific scenario the program is meant for.

Neither of these options, nor any other existing products found, have tension pneumothorax scenarios or are compatible with the CAVE.

Ethics

It is important to make this monitor simulation as realistic as possible so nurses who train in the CAVE will be better prepared for real surgery. If the simulated monitor is unrealistic, when it comes time for the user to perform in a real ER situation, wrong diagnoses may result which could cause harm to or death of the patient. Even if a realistic simulation is created, the user must be aware that it is merely a simulation. After practice in a CAVE scenario, the user should by no means consider his or her self ready to operate on a real person where a real life is at risk. In addition, any signal data acquired for programming use from real people will have to be made anonymous. Any programming assistance received will have to be properly documented and cited.

Ergonomics

Since the user does not physically interact with the monitor, the only relevant ergonomics are the aesthetics of the monitor and the audio it produces. The monitor will be designed after a Philips Intellivue MP70 image provided by the clients. The visual and audio components of this monitor should provide the user with an easy transition from the CAVE to the ER. Audio cues for heart rate and to indicate failing vitals must be implemented to attract the user's attention.

Design Proposal Overview:

The nature of the project is programming-oriented, so the alternative designs represent different programming languages. The clients have specific requirements for how the monitor should look and function, which leaves little flexibility for creativity in designing the monitor. The three alternative designs are described below.

Alternative Design 1-MATLAB

The first alternative design is the programming language MATLAB, which is intended for advanced numerical computing. The most notable feature of MATLAB is that each element is based on matrices. The advantages of MATLAB are that it is available as tethered software and the advanced numerical computing will be useful in simulating vitals (Mathworks). In addition, a realistic EKG simulator is available as a MATLAB program. MATLAB has good options for graphics, but the advanced graphics options are used primarily for mathematical applications (Mathworks). A disadvantage of MATLAB is that only a limited number of functions support multithreading, which means that multiple parts of the program can't be running at once (Mathworks). For example, in the program, each digital and graphical representation of the vitals will represent a separate thread. Since MATLAB doesn't have a multithreading option for the types of functions we would be using, the vitals would not update concurrently. For the program to run effectively, all vitals must update independently and concurrently.

Figure 4: MATLAB logo

http://media.photobucket.com/image/matlab+logo+/cacing_95/logo.jpg

Alternative Design 2-C++

The second alternative design is the programming language C++, which stems from the language C. C++ is an object-oriented version of C (Stroustrup). The advantages of C++ include that it uses less memory, and therefore is a fast language. In addition, C++ is widely used so many people have the skills to add on or modify the program. A disadvantage of C++ is that the language is more advanced and complex than languages that the team has experience with (Morris). Also, to program the necessary graphics, the team may need to purchase additional software for Graphical User Interface (GUI) applications. C++ has a limited standard library compared to Java, especially concerning graphics, which means there aren't as many *built-in* options for graphics (Programmers Stack Exchange).

Figure 5: C++ logo

http://www.pgpnet.dk/?Programming

Alternative Design 3-Java

The third alternative design is the programming language Java. Although Java is capable of robust applications, it is basic enough that it is regarded as a beginner language. Java is multithreaded, so multiple aspects of the program can update independently and concurrently. Java is the most accessible of the three alternative designs, as it runs on Eclipse, which is free and platform independent. In addition, an EKG simulator is available in Java. A disadvantage of Java is that it is slower than C++ and MATLAB and uses more memory (Jelovic). Therefore, the vital signals may not update as quickly as would be necessary.

Figure 6: Java logo

http://geektech.in/archives/1865

Design Criteria

Design criteria were determined based on the goal to create a monitor that meets client requirements. Figure 7 lists evaluation criteria for the alternative programming languages. The categories that were used to evaluate the designs are Universality, Accessibility and Cost, Ease of Use for Programming, and Graphics. The three designs were given a score for each category and then the total score was summed (Figure 8). The Universality category evaluates how common the programming language is. The more common the programming language is, the more likely a programmer will be able to implement new trauma scenarios in the future. Accessibility and Cost is important as the program for the monitor isn't copyrighted or protected. There is potential for other uses for the monitor outside of the uses for the CAVE in training, teaching or outreach. Ease of Use for Programming is the most heavily weighted category. As this design project involves relatively complex programming, the ability for the team to satisfactorily complete the project is contingent on programming skills and experience. The final category, Graphics, is significant because the virtual monitor must look realistic and the graphs must update precisely.

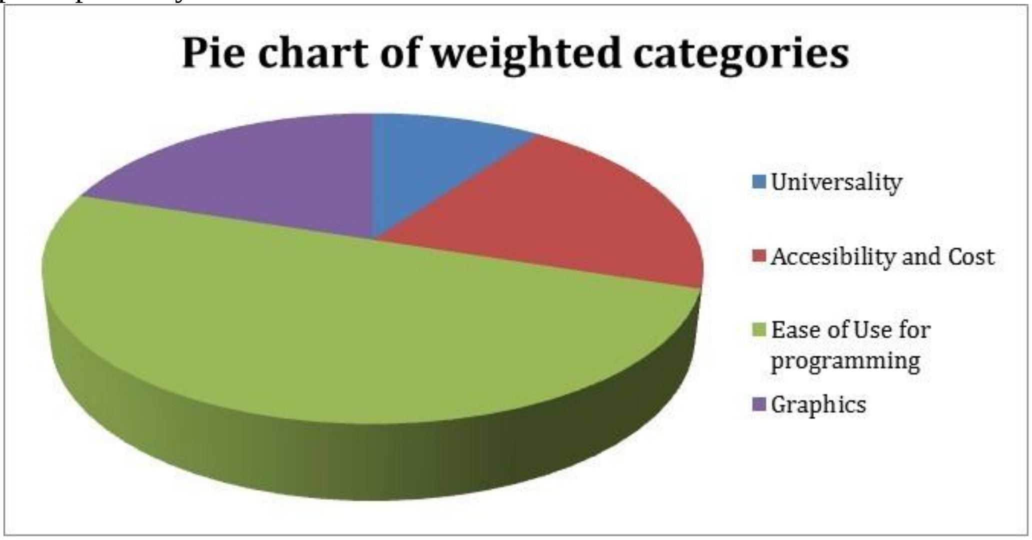


Figure 7: Pie chart of weighted categories

For the Ease of Use for Programming category, MATLAB scored the lowest because only one team member has had experience programming in MATLAB. C++ has similar syntax

	Universality (10)	Accessibility and Cost (20)	Ease of Use for programming (50)	Graphics (20)	Total
MATLAB	4	10	38	14	66
C++	6	12	40	19	77
Java	8	20	47	19	94

Figure 8: Decision Matrix

to Java, so it scored higher than MATLAB in this category. Java scored the highest since two team members have experience programming in Java and it is regarded as a beginner language.

MATLAB is extremely costly and therefore not accessible to the public, so it scored the lowest in the Accessibility and Cost category. In contrast, Java is free and platform independent, so it got a perfect score in this category.

After analyzing the Universality of each alternative design, MATLAB scored the lowest because it is the least common programming language out of the three chosen languages. Java scored the highest as it is the most commonly used programming language in comparison to the other two.

In researching the quality of graphics for each programming language, it was found that MATLAB has the fewest options for graphics while Java and C++ both have a large variety of options (Mathworks). Overall, MATLAB scored 66 points, C++ scored 77 points and Java scored 94 points. Evidently, Java is the strongest alternative design and was chosen as the final design in programming the Virtual Reality Physio Monitor.

Final Design

The final design will utilize a combination of Java and MATLAB in order to create a realistic monitor. Since the CAVE software cannot be run on most computers, a GUI will be created in Java to “simulate” the CAVE environment. This GUI (Figure 9) will display patient vitals, pictures of the scenario and a text description of the patient’s condition. It will also accept input via the “Actions” menu so that users can navigate through each step of the procedure. Vital signals will be simulated using MATLAB and these values will be stored in text files. The text files will then be read by the GUI software and stored in array lists. Various “Updater” and “Grapher” objects which run concurrently in the program will utilize the values in the array lists along with random number generators and user input to determine the signals the monitor needs to display. All of these values will have thresholds to make sure they do not stray from the specific simulated range. In order to efficiently add new scenarios to the software, an “Ailment” interface has been created. For the program to run, new scenario objects must adhere to the methods defined in the “Ailment” interface. Finally, the program will be translated into CAVELib in order to be implemented in the CAVE.

Future Work

In order to complete the Virtual Reality Physio Monitor, simulations of a pulse oximeter plethysmograph and respiration waves will have to be implemented. The updater and grapher classes still need to be written in order to update the GUI. Once the monitor is ready with dynamic signals displayed, work will begin on simulating the tension pneumothorax case. If time permits, other scenarios will be added to the program.

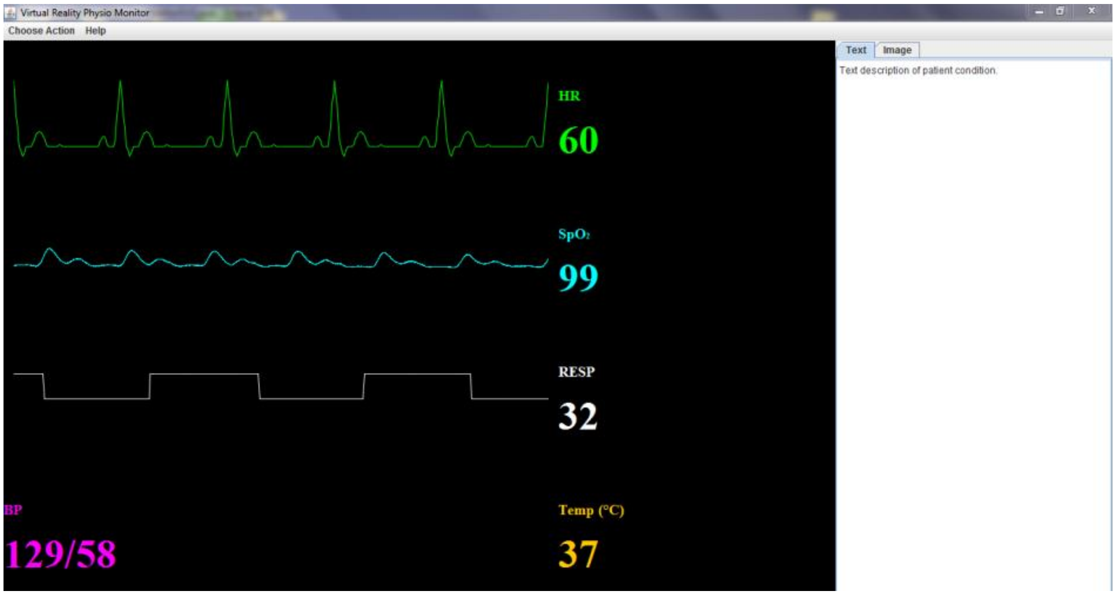


Figure 9: Image of user interface thus far

Works Cited

Chen, Ying-Lun, Chea-Ying Chen, and Jen-Kun Cheng. "Delayed Tension Pneumothorax During Surgery." Mackay Memorial Hospital, 2005. Web. 20 Oct. 2011. <http://homepage.vghtpe.gov.tw/~jcma/68/10/491.pdf>.

"Collapsed Lung - PubMed Health." *PubMed Health*. U.S. National Library of Medicine, 2011. Web. 22 Oct. 2011. <http://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0001151/>.

"Introduction to Nursing Theories." *Nursing Theories.* Web. 23 Oct. 2011 <http://currentnursing.com/nursing_theory/introduction.html>

Jain, Gosavi, and Jain. "Understanding and Managing Tension Pneumothorax." Indian Academy of Clinical Medicine, Jan. 2008. Web. 20 Oct. 2011. <http://medind.nic.in/jac/t08/i1/jact08i1p42.pdf>.

Jelovic, Dejan. "Why Java Will Always Be Slower than C++." Web. 23 Oct. 2011. <http://www.jelovic.com/articles/why_java_is_slow.htm>.

"MATLAB-The Language of Technical Computing." Mathworks. Web. 10 Oct. 2011. <http://www.mhttp://www.mathworks.com/products/matlab/athworks.com/products/matlab/>.

Morris, Richard (2 July 2009). "Niklaus Wirth: Geek of the Week". Retrieved 8 August 2009. "C++ is a language that was designed to cater to everybody's perceived needs. As a result, the language and even more so its implementations have become complex and bulky, difficult to understand, and likely to contain errors for ever."

Programmers Stack Exchange. Web. 10 Oct. 2011. <http://programmers.stackexchange.com/questions/80439/why-is-c-still-preferred-to-build-heavy-gui-apps-over-the-latest-dynamic-langu>.

Radwin, Robert, and Benjamin Mandel. Personal interview. 7 Sept. 2011.

Stroustrup, Bjarne (1997). "1". The C++ Programming Language (Third ed.). ISBN 0201889544. OCLC 59193992.

Anesoft.com - Screen-based Medical Simulation Software. Anesoft, 2010. Web. 26 Oct. 2011. <http://www.anesoft.com/>.

"ANGIO Mentor." *Simbionix - To Advance Clinical Performance*. Simbionix, 2011. Web. 26 Oct. 2011. <http://simbionix.com/simulators/angio-mentor/>.

Appendix A: Product Design Specification Report

Product Design Specification Report
Virtual Reality Physio Monitor

Client: Professor Robert Radwin
Dr. Benjamin Mandel
Advisor: John Webster
Team: Hope Marshall, hkmarshall@wisc.edu (Leader)
Roland Pomfret, pomfret@wisc.edu (Communicator)
Jiaquan (Jason) Yu, yu53@wisc.edu (BWIG)
Gabriel Bautista, gbautista@wisc.edu (BSAC)

Date: September 15th, 2011

In order to simulate a real-world clinical environment for training purposes, a dynamic virtual monitor display system must be improved for the 3D CAVE. The monitor should display dynamic EKG's, respiration rate, blood pressure and SpO_2 levels that respond to changing physiological conditions in the virtual ER trauma bay. The virtual display must sound an alarm if it detects that the avatar patient's vitals are passing a dangerous threshold.

Client requirements:

- Compatible with current virtual reality ER bay trauma room in CAVE
- Respond to a changing virtual physiological environment
- Clear and legible display
  - Display dynamic, realistic graphical readings for heart rate, respiration rate and SpO_2
  - Display dynamic, realistic digital readings for blood pressure, heart rate, respiration rate and SpO_2
- Sound an alarm if vitals cross a dangerous threshold
- Expandable to accommodate physiological scenarios that aren't already built in

Design requirements:

1. Physical and Operational Characteristics

a. Performance requirements: The monitor should display dynamic EKG's, respiration rate, blood pressure and SpO_2 levels that respond to changing physiological conditions and user actions in the virtual ER trauma bay.

b. Safety: The safety hazards are minimal as the environment in question is virtual.

c. Accuracy and Reliability: Blood pressure, SpO_2, respiration rate and heart rate should be accurate to the single digit.

d. Life in Service: The virtual monitor should last as long as the CAVE is in service.

e. Shelf Life: There are no shelf-life requirements.

f. Operating Environment: The virtual monitor must be compatible with Windows XP and CAVELib.

g. Ergonomics: All users must be able to understand and interact with the CAVE virtual monitor. The virtual device must be easy to use and see from the virtual distance of the user.

h. Size: The virtual monitor should be 49cm x 38cm.

i. Weight: This is not applicable.

j. Materials: Google SketchUp, Java, CAVELib, CAVE.

k. Aesthetics: The virtual monitor must appear as a real monitor does in an ER trauma bay. The monitor must be labeled clearly and concisely.

2. Production Characteristics

a. Quantity: 1

b. Target Product Cost: $0.00

3. Miscellaneous

a. Standards and Specifications: Not applicable

b. Customer: The clients require software that responds to changing physiological conditions and is expandable to accommodate physiological scenarios that aren’t already built in.

c. Patient-related concerns: Not applicable

d. Competition: Similar scenarios have been designed. However, the software is not available on the market

Appendix B: Original User Interface and Monitor Design

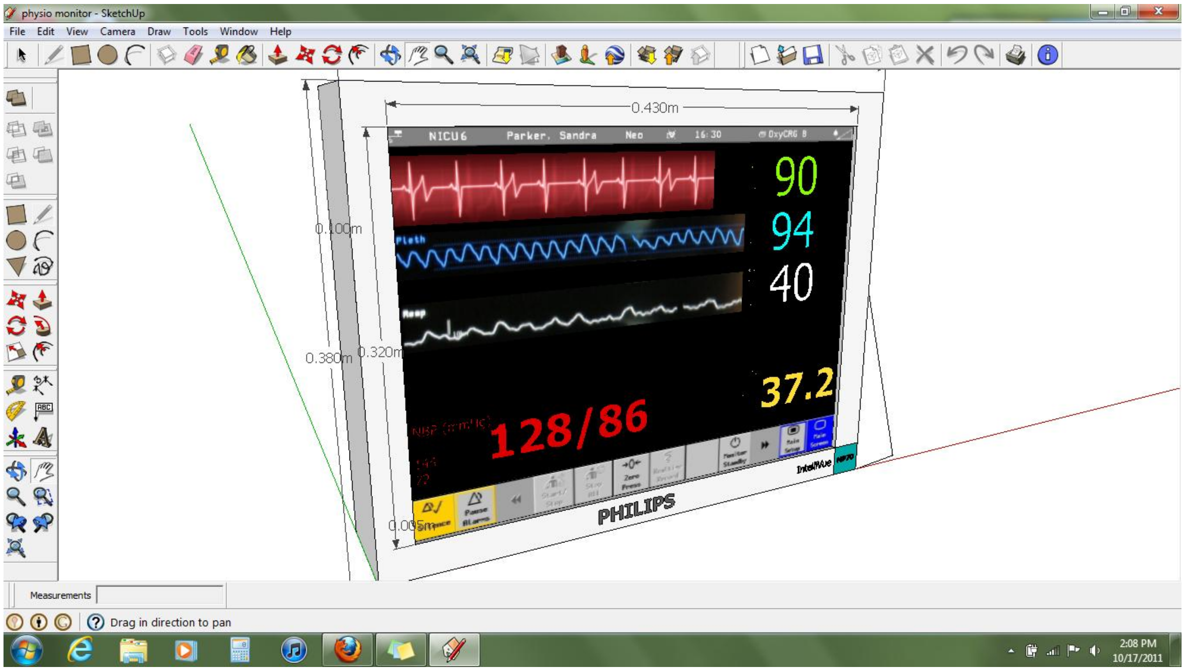


Figure 10: The above monitor design was made using Google SketchUp. The monitor design is a suggestion of what the monitor in the CAVE will look like.

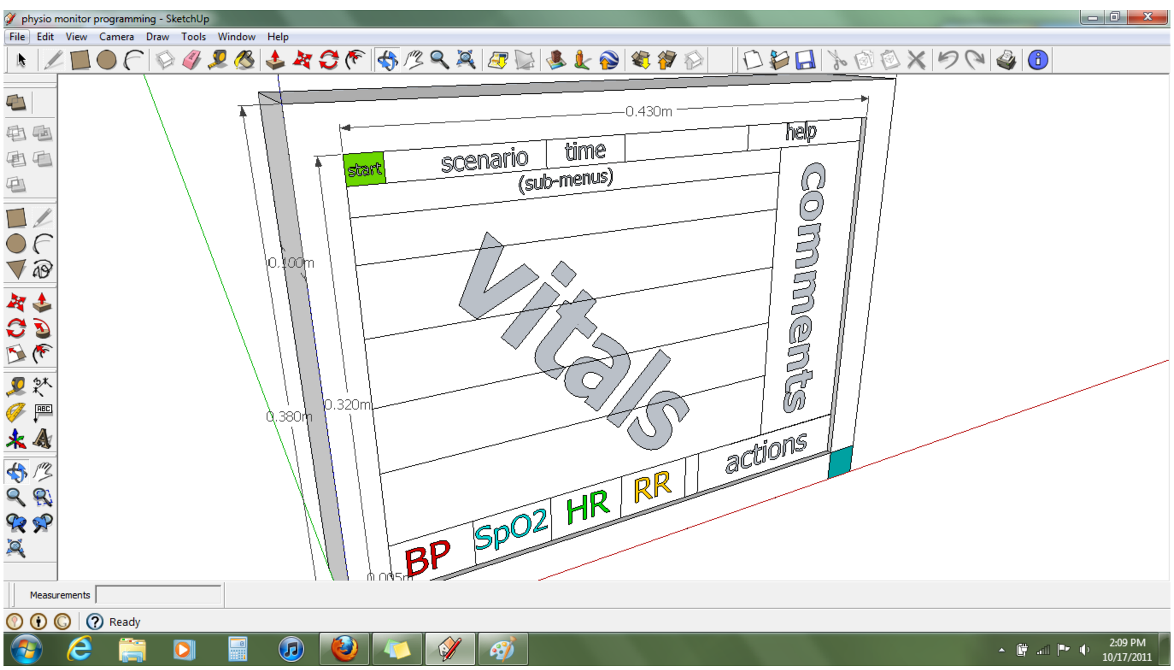
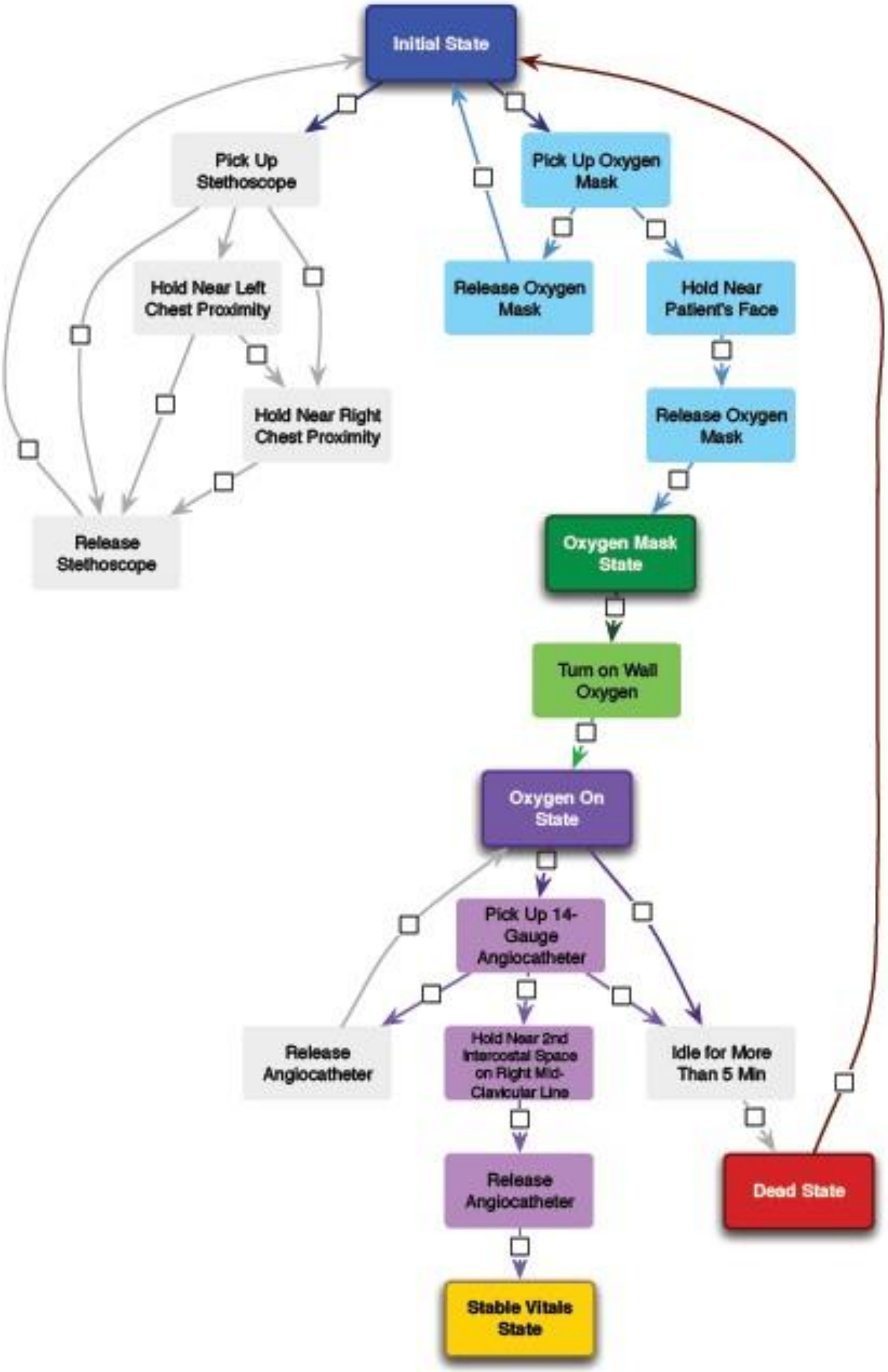


Figure 11: The above user interface design was made using Google SketchUp. The user interface design is a depiction of how the program will be run without actually being in the simulation environment.

Appendix C: Flowchart



Appendix D: Class Diagram