VR Ping Pong Team

Logan Herman, Henry Yi, William Wang

Use Case

Problem: Long distance, virtual interactions are not as personal as physical interactions. This makes digital social interactions less interesting and engaging to everyone involved.

Solution: Creating a virtual reality ping pong game to play against other people around the world in real-time. The VR aspect creates a pseudo-presence that will make the interactions more fun.

Use Case cont.

Our product consists of two main parts: the paddle and the VR headset.

Wireless paddle: Uses sensors to track the user’s motions in trying to hit the ping pong ball, and tries to capture sensor data to simulate the ball’s flight

Virtual Reality Application: Runs on your phone to be like any other VR experience Displays graphics simulating the game and communicates with sensors and game server to provide a real-time ping pong gaming experience.

ECE Areas Covered: Software, Hardware, Signals

Use Case Requirements: User Experience

- End-to-End Latency
  - Average human reaction time ~250 ms
  - Professional ping pong averages ~ 2 hits per second = ~500 ms per hit
  - ~100-150 ms is considered “workable” according to some gaming sites
  - Target latency is around 50% of human reaction time, and 25% of duration of ping pong hit
- Smooth Frame Rate
  - 30 FPS is acceptable, allows for 15 frames to show ball flight path of a professional-speed rally
- Moderate resolution (~360p)

Use Case Requirements: Paddle

- Paddle power lifespan
  - Allow for 1 hour of continuous gameplay
  - Paddle should be wireless in order to maximize freedom of motion
- Accurate Paddle Motion Tracking
  - Paddles are around 6 inches x 6 inches for the face size, 1 inch thick
  - Balls are 1.5 inches diameter
  - +/- 3 inches in terms of position
  - +/- 7.5 degrees in terms of orientation

Technical Challenges Part 1: Sensing + Device Application

- Accurately tracking the paddle's orientation and position
  - Calibration of paddle orientation and position
  - Filtering out sensor noise and keeping sensor error to a minimum
- Powering our paddle's sensors without a large power source, while keeping the paddle powered for sufficient time
- Outputting a desirable FPS at a decent resolution
  - Making sure that graphics rendering can be done at a high enough speed

Solution Part 1: Sensing System + Device Application

- Use a combination of Inertial Measurement Units and Computer Vision for paddle motion sensing
- Sensing system will need to be connected to a microcontroller
- Use of battery to power microcontroller and sensing system
- Google Cardboard VR development kit for the VR mobile application
  - Can be done with Unity or Android Studio
  - Provide libraries for rendering graphics
  - Android libraries for Bluetooth communications as well

Technical Challenges Part 2: End-to-end Latency

- Two factors of latency
  - Sensor to Device
    - Choosing a fast communication medium
    - Having enough processor speed to quickly process sensor data
    - Sending data packets that are efficiently sized
  - Device to Device
    - Biggest challenge in latency
    - Ensuring nearly real time data reception
    - If using the internet, have to transmit data efficiently to reduce lag in player experience

Solution Part 2: End-to-End Communications

- Sensor to Device Communication:
  - From sensor to microcontroller: wired serial communication (I2C, UART, etc…)
  - From sensing system to device: Bluetooth transmission is the obvious solution
  - Full-duplex communication between device and sensor through bluetooth
- Device to Device Communication:
  - Use networking to send data between devices (e.g. Android Sockets)
  - Potential use of a server to receive data from device and relay to other device
    - Allows for players to connect through a central server
    - Need to check how latency is affected
  - Decide what data is absolutely necessary to be sent real time, and what data is not as urgent to be sent through network, and what data can be processed locally
  - Possibly Cache some data

Testing, Verification, and Metrics

- Testing latency:
  - Get ping times for data from device to device
  - Measure delay from sensor to device
- Testing Paddle Sensor Accuracy:
  - Measure error of expected position and the actual physical position of paddle
  - Measure error of expected orientation and the actual physical orientation of the paddle
  - Measure how the ball trajectory reacts to real physical contact with a paddle compared to how the graphics simulate such reactions
    - Where the ball hits on the table
    - Location of the apex of the ball
    - Time it takes for the ball to complete its trajectory
    - Reaction of the ball to spin

Division of Labor

Milestone 1: Design and Purchasing

- Research CV - Logan
- Research IMU sensor and computers (rPi, Arduino, etc.) - Will
- Research VR/Bluetooth/Graphics - Henry

Milestone 2: Design Proposal

- Basic computer vision tracking - Logan
- Test accuracy of IMU - Will
- Measure latency of headset-to-headset and headset-to-camera communication - Henry

Division of Labor cont.

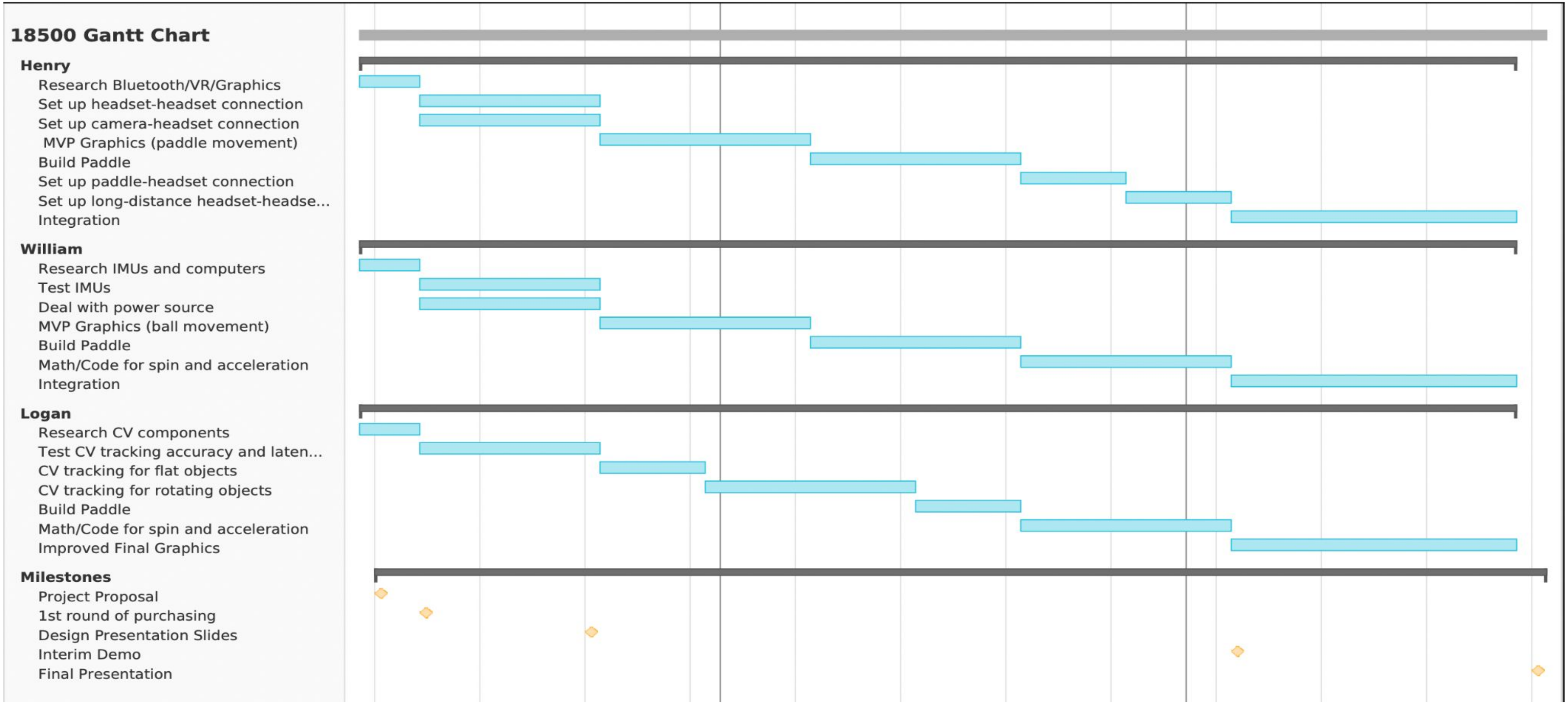
Milestone 3: MVP

- Computer vision tracking for flat objects and MVP graphics- Logan
- Project ball movement - William
- Project paddle movement - Henry

Milestone 4: Final Product

- Build paddle, Computer vision tracking for rotating objects, Final graphics - Logan
- Build paddle, Project ball movement - Will
- Build paddle, Headset-to-paddle communication, Long distance headset-to-headset communication - Henry

Schedule


18500 Gantt Chart
Henry
Research Bluetooth/VR/Graphics
Set up headset-headset connection
Set up camera-headset connection
MVP Graphics (paddle movement)
Build Paddle
Set up paddle-headset connection
Set up long-distance headset-headse...
Integration
William
Research IMUs and computers
Test IMUs
Deal with power source
MVP Graphics (ball movement)
Build Paddle
Math/Code for spin and acceleration
Integration
Logan
Research CV components
Test CV tracking accuracy and laten...
CV tracking for flat objects
CV tracking for rotating objects
Build Paddle
Math/Code for spin and acceleration
Improved Final Graphics
Milestones
Project Proposal
1st round of purchasing
Design Presentation Slides
Interim Demo
Final Presentation