Product Requirements Document (PRD)

Problem Description

Many people still don’t know the capabilities of Augmented Reality. AR is an underutilized technology for its many possibilities. It is hard to access and confusing to new users. Because of this, there is not much interest among the general public.

Scope

The scope of this project is to develop an Augmented Reality experience that blends with real-world physics. There will be virtual rubber balls that the player will be able to interact with in different ways. The user will manipulate a physical object (e.g. a box) and see the effect that has on the balls. We will include some

features such as changing environmental factors like gravity, bounciness, or the box itself.

Use Cases

The user will grab a ball and move it at their leisure, then they will throw the ball and watch it fall. They will try to throw the ball into a goal in the artificial environment.

The user will search for a setting and be able to set different physics parameters. Then the user will modify the ball variables to their preference.

The user will be able to see the graphically generated balls inside a physical box. Then the user will be able to move or rotate the physical box and see the reaction of the digital balls.

The user will play a level with the objective of getting all balls into a designated hole. The hole will also move and change size as time passes. The user will lose the level if they don't make all the balls into the hole before the time runs out.

Purpose and Vision (Background)

The purpose of this project is to create a game where players can interact with the objects within to deliver a fun and memorable Augmented Reality experience for the user.

We want to use the Meta Quest 3 to experience the wonders of AR, and provide a way to seamlessly integrate real world physics and virtual objects.

Stakeholders

- People who are interested in games
- Engineering Expo
- People who are interested in art
- Our Project Partner
- Developers (Us)

Preliminary Context

AR stands for Augmented Reality, which is a software that displays objects in the real world around you.

Assumptions

We can create a virtual object in an environment.

We can develop an app that works on an AR platform (Meta Quest 3).

We have access to users who can test and give valuable feedback.

We have until the end of the school year to develop a shippable product.

We will have a Quest 3 available for testing major releases.

We assume that most users will have little/no experience with AR so the product will be simple and understandable.

There is a well-tested and reliable development environment for the Meta Quest 3 readily available for use.

Constraints

We are a small team, and should focus on making a small product at first.

We need to make sure that we have access to hardware that can run an AR application for testing.

The project is constrained by the capabilities and limitations of the Meta Quest 3 and AR development tools, which may impact the complexity of physics simulations and graphical quality. Also limited by capabilities of computers running the software.

There is a lot to discover about AR development for this project.

Time constraints, team members might need to prioritize other classes over this project at times.

We need to make sure that we can deploy the application to be used on the Quest 3.

Dependencies

Meta Quest 3 Availability: The project relies on the use of Meta Quest 3. The availability of these devices for development and testing is a critical hardware dependency.

Software Dependencies: The project depends on the integration of AR development tools and 3D computer graphics software to create the AR experience.

Data Sources for Physics Simulations: If the project includes real-world data for physics simulations (e.g. gravitational constants for different planets). Access to and accuracy of these data sources are crucial.

We need to depend on our knowledge of Computer Graphics, since it is a prerequisite for working on this project.

Market Assessment and Competition Analysis

Hardware:

Microsoft Hololens 2

There is some documentation for Hololens development within Unreal Engine, but it is not as fully fleshed out as other products like the Quest. Hololens is designed for professional environments, so it can be expected to be a slightly higher quality than other products. It is also much more expensive.

Apple Vision

Needs more research but the features and implementation and capabilities look awfully similar to that of Hololens. However I'm not sure what opportunities there are for development of outside software.

Meta Quest 3

Needs more research but there is documentation for Quest application development within Unreal Engine that seems to be more up to date than HoloLens development. Quest development would be more ideal for developing a game as Hololens is designed more for professional workflows and Quest is more of a consumer grade AR device.

Software:

There do not appear any direct alternative AR games on the Quest store to what we are planning to create. There are other physics based applications for the Quest 3, however none appear to be directly comparable with our plans. Further research needs to be done on this, however.

There are some existing physics applications that are open-source, such as Microsoft's "MRTK Examples Hub", which has several environments where you can manipulate objects. Some of this could certainly be re-used for our project.

Non-AR games: There may be some games out there that share some of the requirements that ours needs to follow, but they don't use AR functionality.

Target Demographics (User Persona)

Types of users most likely to use this product:

- Individual users
- Businesses

Users:

- John is a 25 year old that is interested in AR, but wants to see something moving in the real world instead of among other simulated objects.
- A business based in the augmented reality industry wants to hold a seminar to try to get more people interested in the field. They need some applications that demonstrate the capabilities of AR to new users.
- Erin is curious about augmented reality and wants to try out some applications. There are plenty of productivity focused ones, but they want something more fun and interactive to try.
- Joe is a 20 year old and likes to play video games. He wants to play something unique and something that can offer a good challenge.
- Steve is 27 years old and wants to see some creative work. He is especially interested in art that he can interact with.

Requirements

The product should:

- Have simulated objects moving around the real world around you
- Make the simulated objects react to the user
- Have an optional goal for the user

- Realistically simulate the physics of an object

We need to discuss what other features are wanted and realistic.

User Stories and Features (Functional Requirements)

User Story	Feature	Priority	GitHub Issue	Dependencies
As someone who wants to learn more about physics, I want the game to have controllable physics so that I can see what difference each change makes.	Adjustable parameters	Must Have	TBD	N/A
As someone new to Augmented Reality, I want to experience something unique that shows the capabilities of AR so that I can find if it is something I am interested in.	Accurate physics simulation	Must Have	TBD	N/A
As a science educator, I want to use this AR application as a teaching tool, so that I can explain complex physics concepts to my students in an interactive and immersive way.	Create an educational mode within the AR application that provides explanations and real-world examples for different physics scenarios.	Should Have	TBD	TBD
As someone who plays games, I want to have a goal in the simulation so that I can stay engaged with the product.	Point system for game	Should Have	TBD	N/A

Non-Functional Requirements

Documentation and Coding Standards

- Code should be well-documented, following established coding standards and best practices to ensure maintainability and collaboration within the development team.
- User guides and documentation should be provided to explain the application's features and functionality to users and educators.

Performance

- It should render virtual objects smoothly, maintaining a minimum frame rate of 30 frames per second for a seamless AR experience
- The system should calculate fast enough to be able to make objects move realistically
- Response time for the user should be less than 300 milliseconds
- Visuals respond to motion fast enough as to not cause discomfort/confusion (unsure of specific number)
- Simulation should be free of visual/physics bugs
- Simulation should be resistant to unexpected actions/movement while running
- The system should recover from a failure within 3 minutes

Accessibility

- The game should be playable for anyone with an AR device
- We shouldn't spend too much money on developing this game

Data Requirements

The product needs to collect data about the environment around it in real time.

Physics Simulation Data

- Data Type: Real-time physics simulation parameters
- Data Fields: Parameters for gravity, friction, ball characteristics, and challenge settings.
- Purpose: To define the physics behavior of virtual objects in the AR environment.

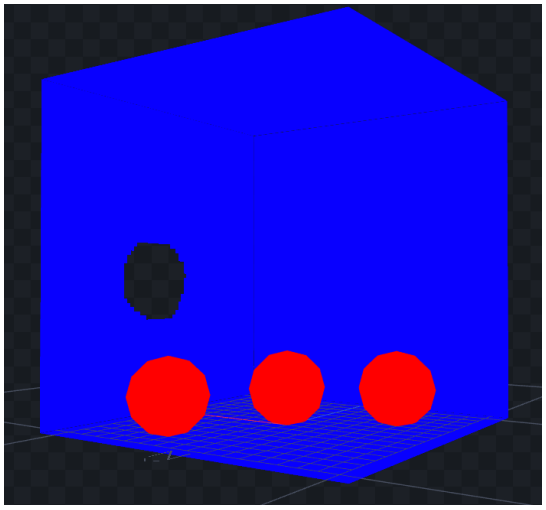
Integration Requirements

- The product needs to integrate with Meta Quest 3, as that will be the platform we are developing for.
- This product will not work without the platform. Microsoft Store (if we intend to make it publicly available that way).

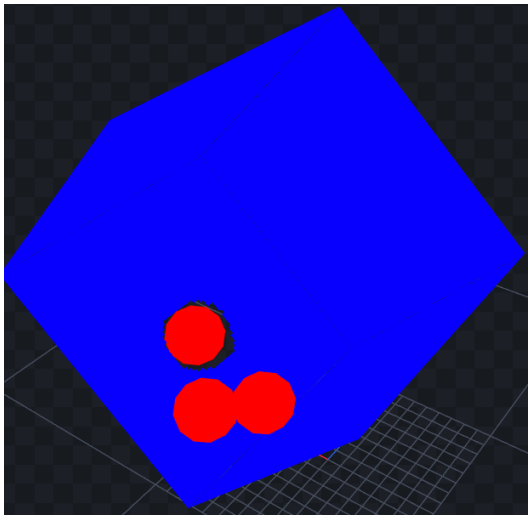
- 3D software integration: Utilizing 3D AR development tools for the creation of the AR environment and physics simulations.

User Interaction and Design

The user should be able to see the objects in the space without anything other than settings on their screen.



You have a box and a certain number of balls. The goal of the game is to get all of the balls through the hole within a certain amount of time, by moving the box. While not shown, the gravity and bounciness of the balls can be altered.



The hole will change its position and size after a certain amount of time passes. There's also a chance that you'd have to put the balls through the hole in a specific order.

Milestones and Timeline

End of Fall Term

- Get familiar with Unreal Engine
- Movable box
- Balls within the box are able to move with realistic physics
- Basic game where you put the balls through the hole within a certain amount of time

End of Winter Term

- Gravity, bounciness of balls, and number of balls can be altered
- Move simulation into AR environment
- Have hole change position and size depending on time
- Add difficulty where you need to get the balls through the hole in a specific order
- Have a game that is mostly complete

End of Spring Term

- Have a game that is polished and fully finished, ready to be shipped out.

Goals and Success Metrics

Goal	Metric	Baseline	Target	Tracking Method
Increase interest in AR	Average session duration	1 min 30 sec	5 minutes	Plausible Analytics
Ensure Technical Reliability and Quality	Bug-Free Experience	No initial bug-free experience	Achieve a 98% bug-free experience within the first month of deployment, with no critical bugs impacting user gameplay.	Monitor bug reports, identify and resolve issues promptly.
Have the game be beatable	Average win rate	Win rate < 60%	Win rate > 60%	Interview

Keep the player immersed	Plausibility of Simulation	Simulated balls bounce randomly within the parameters on the physical box	Simulated balls bounce according to the environment and user controlled parameters.	Monitor how simulated balls bounce within the environment. The balls should interact with the user as they want. Interview.

Out of Scope

Scale and Performance: The project will prioritize the proof-of-concept and functionality over extensive scalability and performance optimizations. These optimizations, if necessary, can be considered in later iterations. (Can we make this “game” more complicated depending on the size of the box we are looking at?)

Create a non-AR version of the game so that it becomes more accessible to play.