

Vintage Goes AR, Inc.

Asteroids Game Design

Document

Game Overview

Project background

Vintage Goes AR is a gaming company that transforms vintage games into exciting AR experiences. They have contracted with you to create an AR version of the classic Asteroids arcade game.

High-level concept

Game name: Asteroids AR

Target device: Mid- to high-end mobile phones, Android and iOS

Target audience: Action game players, fans of AR games and fans of classic arcade games.

Release date: Next summer

Publisher: Vintage Goes AR

In this mobile app-based game, players will be in an outer space scene, protecting Earth from incoming asteroids and UFO saucers. Players can earn points and mystery prizes for destroying them.

Game objective

The objective of the game is to destroy the asteroids and UFOs. Shooting a large asteroid breaks the asteroid into two medium sized asteroids. Shooting one of those pieces breaks it into two small asteroids. Asteroids will travel towards the Earth behind the Player with a small direction offset. Asteroids can damage the player and Earth. If the player's or Earth's life reaches zero, then the game is over.

Gameplay

Scoring

Players will earn the following scores upon destroying the asteroids and UFO saucers:

- Small Asteroid = 100
- Medium Asteroid = 50
- Large Asteroid = 20
- UFO = 100

Player mechanics

- The camera's view is the player, with the Earth appearing behind the player.
- Asteroids will spawn in front of the player and traverse towards the Earth.
- Players can move and rotate the phone to aim at asteroids.
- Player shoots by tapping the screen. An unlimited number of bullets are created at the center of the screen and will travel in a straight line from the direction of the shooter.
- If asteroids reach the camera (player's view) then they will damage the player.
- Players will lose a small amount of health each time an asteroid damages the player, and a health bar will indicate the player's remaining health.

Earth

- Is a large sphere behind the player.
- Asteroids will damage the Earth if they crash into it, with smaller asteroids causing less damage than larger asteroids.
- The Earth's health bar will appear on the screen.
- If the Earth is destroyed (reaches 0), the "game over" screen displays.

Asteroids

- Asteroids spawn on the opposite side of the Earth. Vertical spawning is limited to prevent asteroids from appearing too high or too low.
- Asteroids move towards Earth after spawning.
- Large and medium asteroids are split when they are hit.
- Asteroids are destroyed when they reach the player's view (the camera's position), damaging the player.
- Asteroids also are destroyed when they damage the Earth or are hit by a player's bullet.

UFO

- The UFO will spawn in a random position in front of the original player position.
- Each UFO traverses the player's eyesight (like on [Bayonetta's Angel Attack](#)).
- The UFO shoots bullets towards the player. These bullets can't be destroyed, they will always damage the player if they hit.
- The UFO can be destroyed by bullets.
- When a UFO is destroyed, a mystery box will appear in the scene for 10 seconds. If the player destroys the mystery box, this box will release a recovery Health Item that will start moving towards the player. When this item touches the player, a small amount of player health will be gained.

User interface

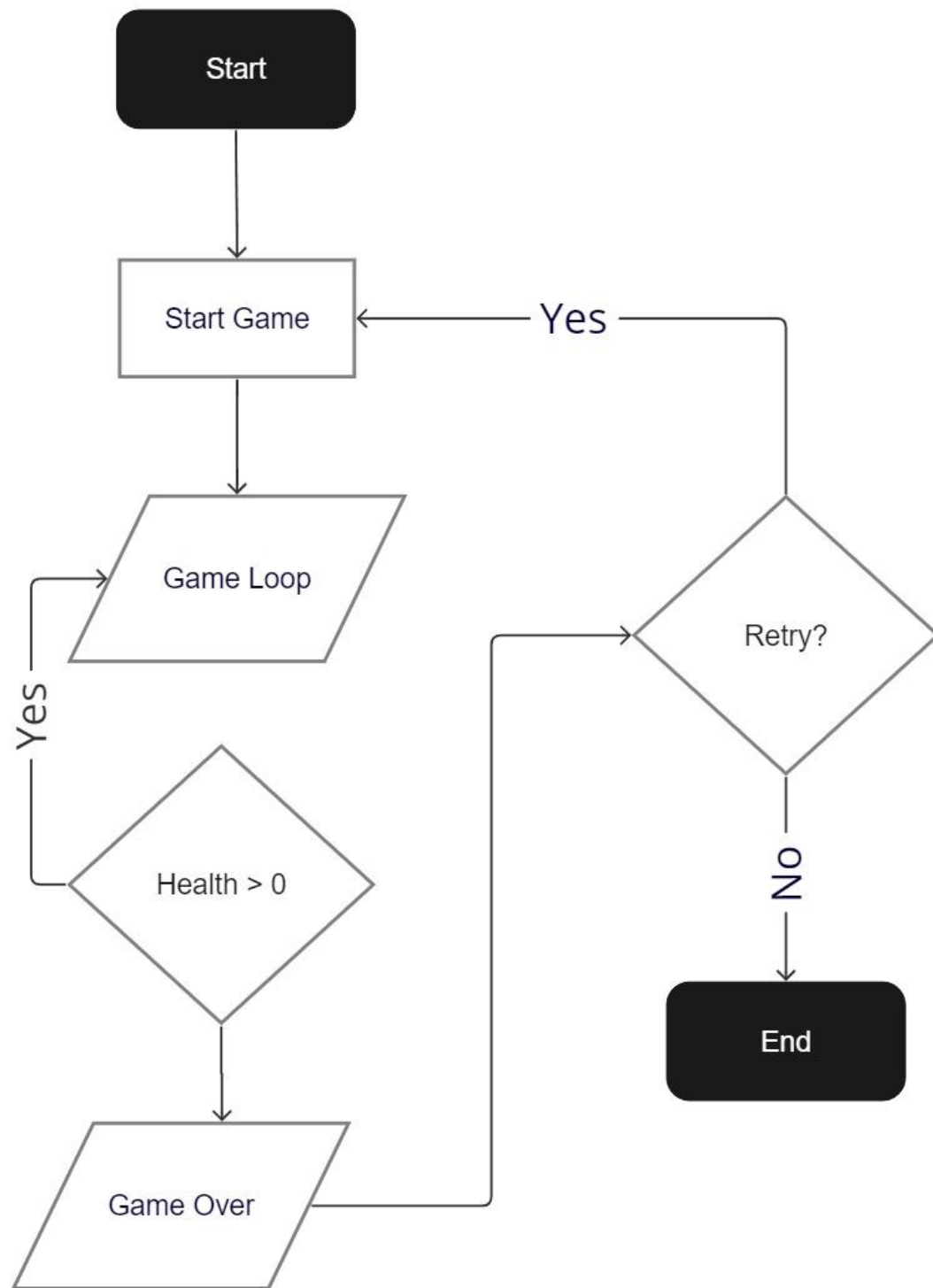
- Health bar indicating remaining life of Player
- Health bar indicating remaining life of Earth
- Score amount

User Feedback

Event	Action	Feedback
Player gun firing bullets	Tap on screen	Sound: Gun firing Visual: Bullet fired from player's view.
Asteroids hit by bullets	Bullets fired by the player hit Asteroid objects	Sound: explosion if hit Visual: If asteroid is hit, <ul style="list-style-type: none">• Large and medium asteroids break into pieces• Small asteroids are destroyed and disappear
UFOs hit by bullets	Bullets fired by the player hit UFOs	Sound: explosion if hit Visual: If a UFO is destroyed, the explosion releases a mystery box that moves towards the player. When this item touches the player, a small amount of player health will be gained.
Player Health	Asteroid hits player	Visual: Player Health bar decreases.

Bar Decreases		
Earth Health Bar Decreases	Asteroid hits Earth	Visual: Player Health bar decreases.
Asteroid Damages player	Asteroid hits player	Sound: An explosion. Visual: Player damage explosion.
Asteroid Damages Earth	Asteroid hits Earth	Sound: An explosion. Visual: Earth damage explosion.

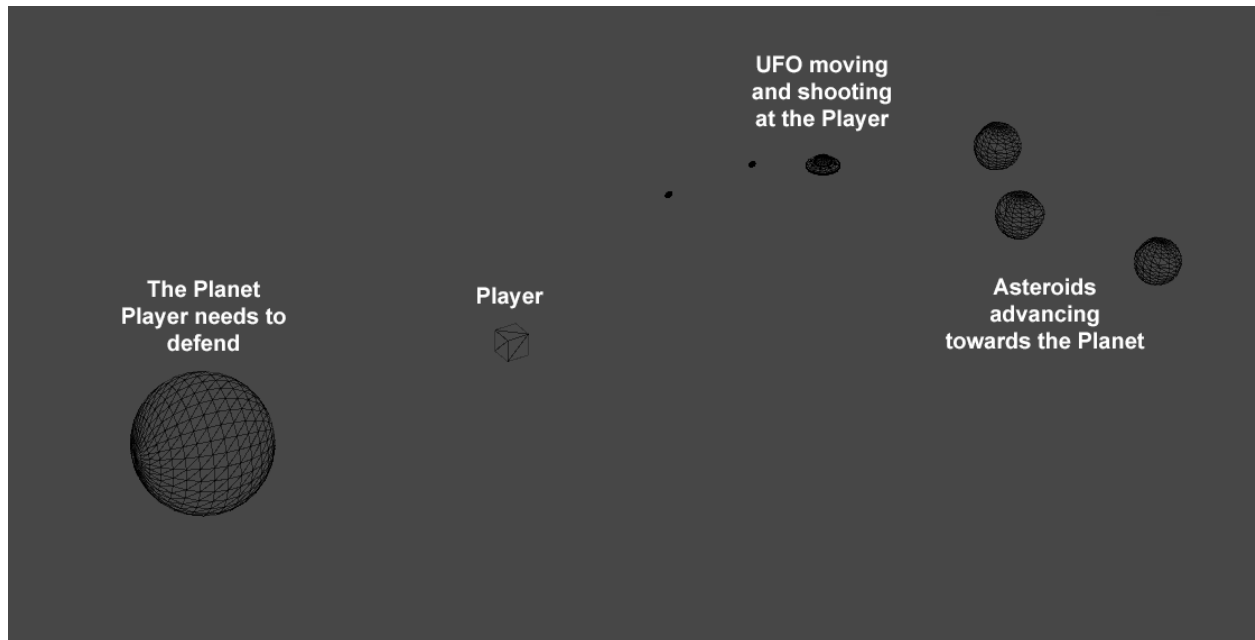
Game Flow



Theme / Setting / Visual Style

Vintage AR would like the Asteroids AR game to create the same feeling of action and danger as the vector graphic style [original Asteroids arcade game](#), but with modern 3D assets, VFX and audio.

Scene Mockup



Asset list

Visual assets

- 3D
 - Asteroids
 - Materials
 - Asteroid.mat
 - Source
 - AsteroidLargeMesh.fbx
 - AsteroidMediumMesh.fbx
 - AsteroidSmallMesh.fbx
 - Textures
 - Asteroid_Diffuse.png

- Asteroid_Metallic.png
 - Asteroid_Normal.png
- Earth
 - Materials
 - Earth.mat
 - Source
 - Earth.fbx
 - Textures
 - Earth_Diffuse.png
 - Earth_Metallic.png
 - Earth_Normal.png
- HealthItem
 - Materials
 - HealthItem.mat
 - Source
 - HealthItemModel.fbx
 - Textures
 - HealthItem_Diffuse.png
 - HealthItem_Metallic.png
 - HealthItem_Normal.png
- Mystery Box
 - Materials
 - MysteryBox.mat
 - Source
 - UFOlaserModel.fbx
 - Textures
 - UFOlaser_Diffuse.png
 - UFOlaser_Metallic.png
 - UFOlaser_Normal.png
 - UFOlaser_Emissive.png
- PlayerBullet
 - Materials
 - PlayerBullet.mat
 - Source
 - PlayerBulletModel.fbx
 - Textures
 - PlayerBullet_Diffuse.png
 - PlayerBullet_Metallic.png
 - PlayerBullet_Normal.png
 - PlayerBullet_Emissive.png
- UFO
 - Materials

- UFO.mat
 - Source
 - UFOModel.fbx
 - Textures
 - UFOlaser_Diffuse.png
 - UFOlaser_Metallic.png
 - UFOlaser_Normal.png
- UFOlaser
 - Materials
 - UFOlaser.mat
 - Source
 - UFOlaserModel.fbx
 - Textures
 - UFOlaser_Diffuse.png
 - UFOlaser_Metallic.png
 - UFOlaser_Normal.png
 - UFOlaser_Emissive.png
-
- Fonts
 - Thaleah_PixelFont
 - Materials
 - Thaleah_DemoBackground.png
 - ThaleahFat.mat
 - ThaleahFat.png
 - ThaleahFat_TTF.ttf
 - Thaleah_Demo.unity
 - Thaleah_Readme.pdf
 - ThaleahFat.fontsettings
 - ThaleahFat_TTF SDF.asset
- Scripts
 - Generic
 - AudioSfx.cs
 - DestroyOnGameOver.cs
 - FlashEffect.cs
 - GameState.cs
 - PlayAudioOnAwake.cs
 - RandomizeUIElements.cs
 - Rotate.cs
 - SceneLoader.cs
 - Utils
 - PlayerMouseController.cs

- UI
 - Prefabs
 - HUD.prefab
 - Textures
 - Asteroid_Original_1x.png Sprite Sheet containing:
 - Button
 - CrossHair
 - Earth
 - Heart
 - LargeAsteroid
 - ProgressBar
 - SmallAsteroid
 - SpaceShip
 - Star
 - Crosshair
 - AsteroidsUIBackground.png
- VisualEffects
 - ExplosionParticles.prefab
 - ExplosionMaterial.mat
 - SmokeMaterial.mat

Audio assets

Arcade SFX FREE

<https://assetstore.unity.com/packages/audio/sound-fx/arcade-sfx-free-42983>

Asteroids AR Technical Design Document

Engine

Unity 2021.3.1f1 has been chosen due to its ease of use for AR development. There is a free version, and it has a big list of features and the best target platform deploying capabilities.

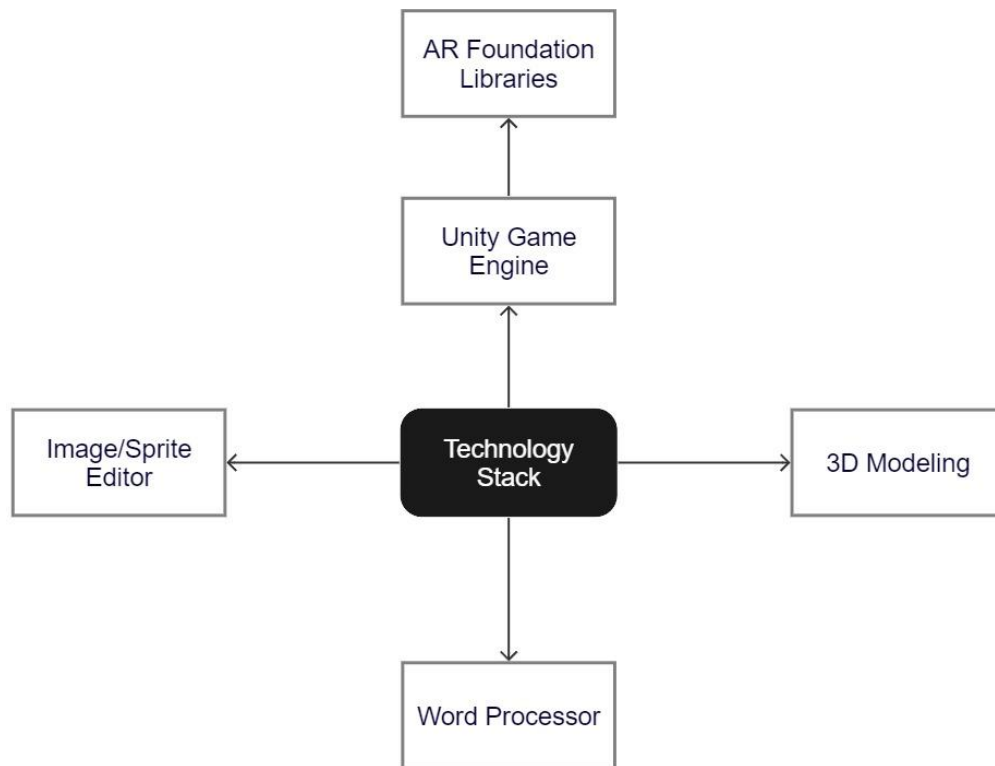
Target Platforms

- Android: Mid-End and High-End devices with Android 11 or newer
- iOS: iPhone 8 or newer
- Both versions will use a 9:16 aspect ratio and will target FullHD resolution

Main Features

- Game will be played using AR
- User will see game elements around their environment
- Asteroids will continuously and randomly spawn on scene
- Player is capable of shooting bullets
- UFO is randomly spawning and moving in front of player
- Mystery box contains healing item
- Earth spawns behind player (what user needs to protect)
- Scoring system
- Classic arcade look and feel
- UI
 - Score
 - Earth's health bar
 - Player's health bar
 - Crosshair

Technology Stack



miro

Asset Specifications

- Maximum of 500 triangles per object
- Maximum texture size of 512 x 512 pixels
- Prefer the use of sprite atlases
- Audio format should be .wav
- Fonts should be .ttf or .otf format

Naming Conventions

- For code, use the official C# naming convention provided by Microsoft:
<https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions>
 - Most common conventions:
 - Pascal Case for naming classes and methods
 - Camel Case for variables
 - Prefix variables with _ when they are private fields
 - Camel Case for local variables and parameters, do not use _ with these
- For asset names:
 - Use Pascal Case for all assets including Scripts, Models, Audio clips, Textures, Scenes, Prefabs, Materials and Scriptable Objects.