About me

And my AR project

Introduction

You will get to know me a bit better

01

Academic experience

Awards and achievements while studying

02

Vuforia Project

In-depth analysis of my AR dissertation project

Game engine dev.

Creating a pool game with custom graphics and physics

03

I'm Constantine

01

- **BSc in Games Technology**
  Graduated: Nov 2019

- **BSc in Economics**
  Graduated: Nov 2018

Do I finish my studies in a year, or do I start studying another subject and complete both in parallel?

—My dilemma

Final year field exploring!

Global Game Jam

Best Game
Best Lighting

Debugging

1st class on my final exam
Debug events

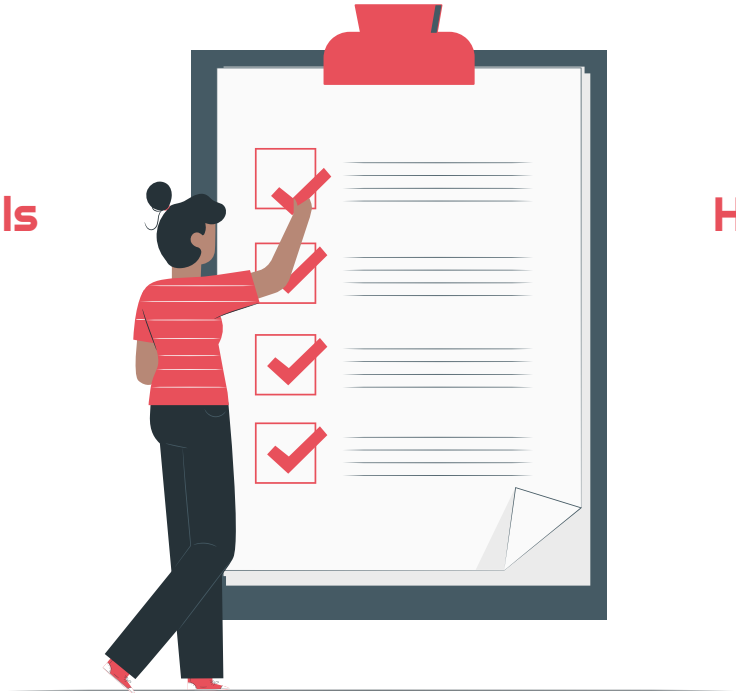
Coventry Game Jam

Top 3 position for the
developed game

Developed skills

Soft Skills

- Communication
- Teamwork
- Time management



Hard Skills

- C++ ,C#
- Git collab
- Game engines

Further development

German language

Improving my German

SLAM Algorithms

Reading literature about SLAM

02

Vuforia vs ARCore

My dissertation project

Watching AR applications for the first time

At this moment I knew I wanted to learn more about AR

Integrating AR into my studies

Researching AR and deciding my Dissertation project

Dissertation Timeline

Literature research and finalising the content

Complete development and performance documentation

Decide the project and get ethics approval

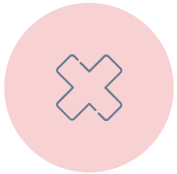
Application development

January

Project Evolution

HoloLens AR Game

-Initial idea to develop a game on HoloLens



Limitations

-Project lacked research question
-Considering available resources

Android AR SDK comparison

-Research question requirement fulfilled
-Android device available

January | February

FInalising the project content

Choosing SDKs

Vuforia and ARCore
Most documentation and tutorials

Figuring out the question to answer

A performance comparison between the two most popular Android SDKs

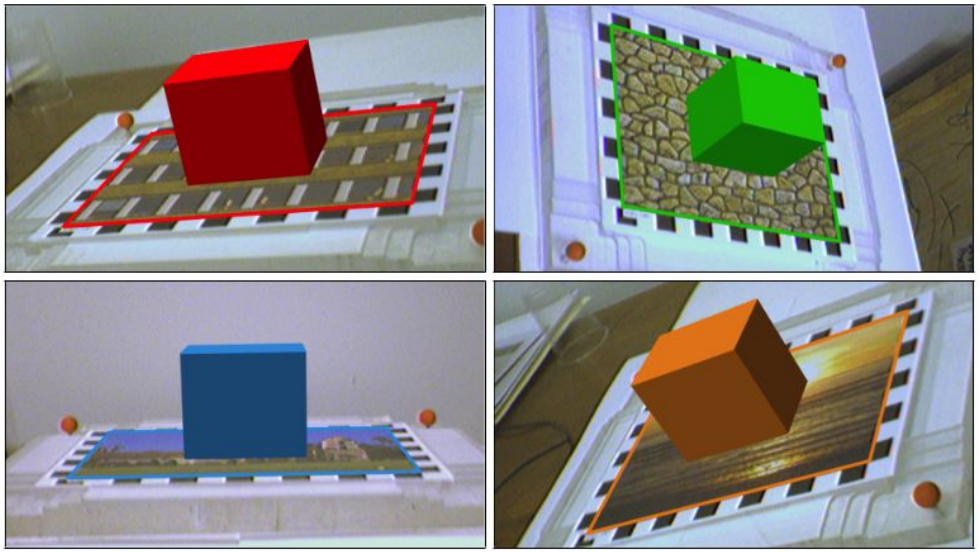
Deciding on the game design

-Time limitation

-Do the work twice

-Easy to replicate

-Tap and destroy



January | February | March

Developing with Vuforia

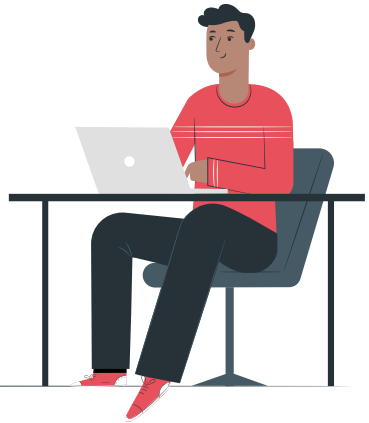
-Unity game engine

-Image targets

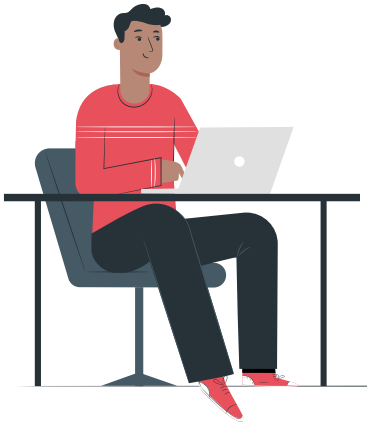
-Create and download image database

-Integrate into the game engine

-Develop the game



Developing with ARCore



-Find resources

-Enable compatibility with Unity

-Create image database and bitmaps

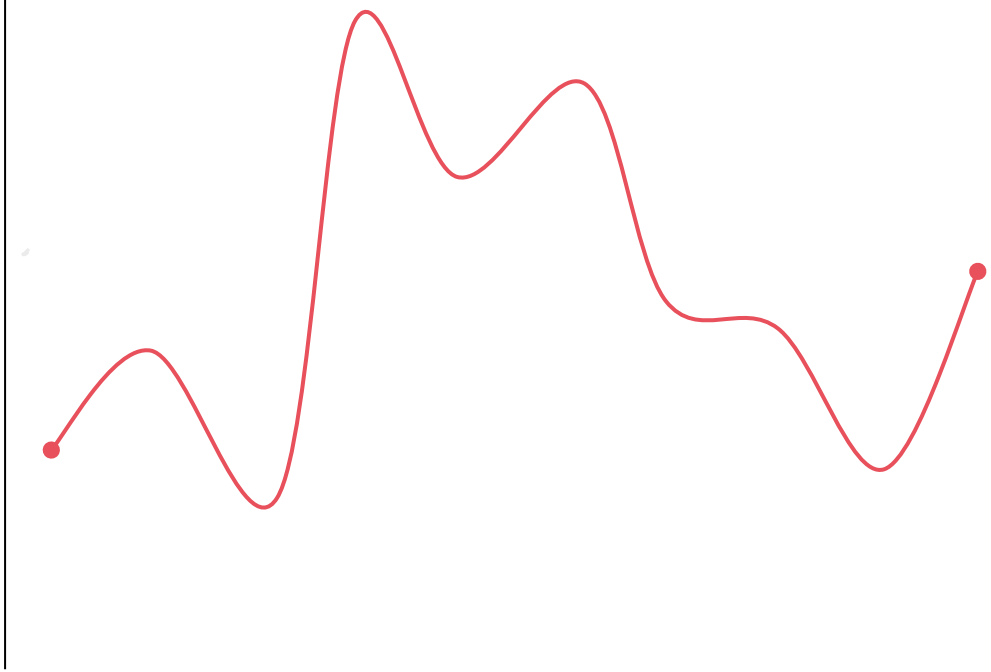
-Add "listeners"

-Start developing the game

Profiling

Using the Unity profiler tool to measure performance

- CPU
- Memory
- FPS



Sampling Methodology

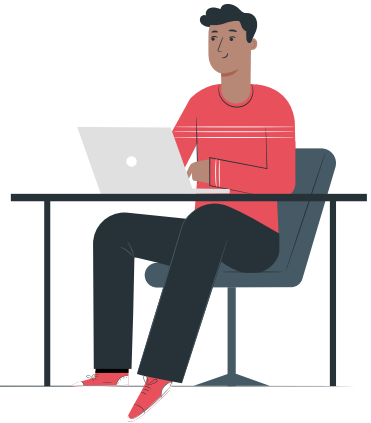
-Measure each SDK performance for 30 seconds

-Calculate their performance with 1 image target

-Calculate their performance with 3 image targets

-Find the average stats for the duration

-Compare



Results Analysis

Vuforia Image targets was the winner

14.26%

Less memory used and less CPU use compared to ARCore Augmented Images

With two weeks left it was time to document the results

Conclusions

Vuforia Image Targets

- Limited experience friendly
- Native compatibility makes development process straightforward
- Better performance

ARCore Augmented Images

- Limited amount of online tutorials
- Intermediate skills required in order to start the development
- Similar FPS outcome to Vuforia

Things I would do differently

Compare the performance of Surface detection instead of image detection

Develop the applications on a HoloLens device

03

Game engine dev.

Creating graphics and physics from scratch

Graphics programming

Year 1

Mathematics for computer graphics
Introduction to graphics programming

Year 2

OpenGL development

Year 3

Advanced graphics
Ray Tracing and Ray casting

Developing without a Game engine

Decide on the game

Come up with a game idea that would be viable for a semester coursework

Create OpenGL graphics

Use modern openGL in order to create the meshees

Create realistic physics

Make collisions and friction to have a realistic outcome

Developing process

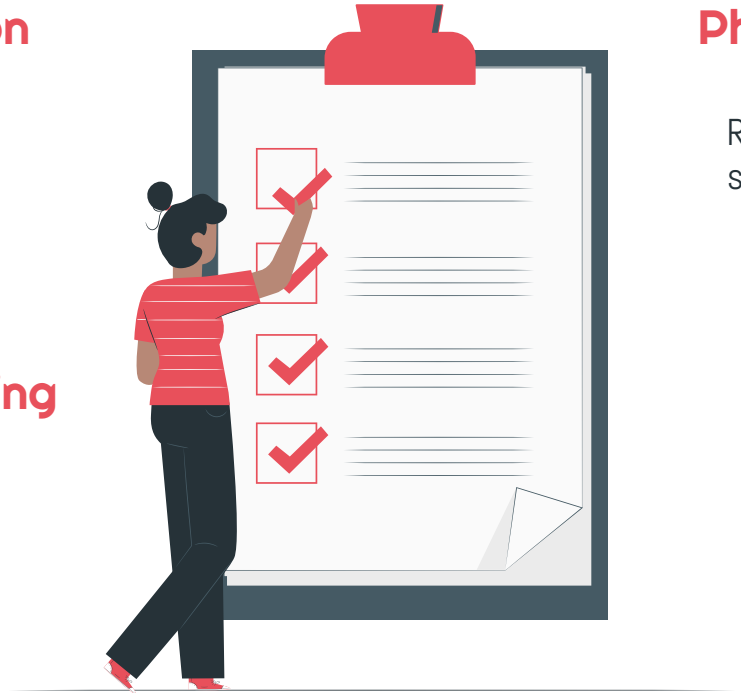
Universal application

GLFW library

GLEW library

Graphics programming

Create the meshes for the spheres and the pool table



Physics programming

Realistic transfer of the speed and adding friction

Game polish

Ambient, Diffuse and Specular lighting

Phong's Algorithm formula

Final thoughts on game engines

Game engines are making the development faster and less complicated

People with no graphics knowledge can work with primitive objects and create amazing results

Developing graphics from an empty C++ project is a time consuming task but not impossible!

Thanks!

Do you have any questions?