**Justification of choices**

Contents

[Language for Android app 1](#_Toc468711919)

[Java 1](#_Toc468711920)

[Kotlin 1](#_Toc468711921)

[C#/Unity/Vuforia 1](#_Toc468711922)

[Language for admin app 2](#_Toc468711923)

[Server side 2](#_Toc468711924)

[.NET C# with Entity Framework 2](#_Toc468711925)

# Language for Android app

Options:

* Java
* Kotlin
* C# /Unity/Vuforia

## Java

Pros:

* Familiarity – studied for 2 years and already used in Android Projects
* Used in current Sabisu apps, familiar to the client
* More standard

## Kotlin

Pros:

* Removes a lot of boilerplate code
* Null safety, no more unexpected null pointer exception
* Interoperability between java and kotlin, can use existing Java libraries

Cons:

* Something new to learn, could take up time
* Not much support from people I know

## C#/Unity/Vuforia

Pros:

* Vuforia SDK is meant specifically for AR
* Has the potential to create a very impressive looking app

Cons:

* Would have to learn Android development in Unity
* No support for existing java libraries
* Image recognition is not good for simple shapes
* Possible cost of unity pro

# Language for admin app

## Server side

### .NET C# with Entity Framework

Pros:

* Familiarity – Would be quick to develop
* Lots of templating from Visual Studio and other tools
* Standard to the client – would fit with their existing infrastructure

### Node.js

Pros:

* Lightweight
* Easy to deploy
* More modern technology
* Already familiar with javascript
* Good for low cpu intensive tasks

Cons:

* Still some things to learn

# IDE for Android App

## Android Studio

* Industry standard IDE
* Fully supported by google
* I have experience in using it

# IDE for .NET applications

## Visual Studio

# IDE for Node.js applications

## Visual Studio

## Visual Studio Code