Title: AR Application, Testing and Deployment Report  
  
Introduction:  
In this testing report, we shall document how we tested and deployed the AR application software which is an augmented reality project on android smart phone. This application entails visualization of a three-dimensional model onto the image target through Vuforia engine inside the unity development environment.

Making sure that the unity we used have the Vuforia engine attached and license acquired.

And the 3d model is already build.   
  
  
  
Adding Image Target:  
Image target added using Vuforia engine in Unity.  
AR one image dataset that contains 3D virtual object, called AR one target.  
Rescaling, making width and height adjustments.

Adding Cube as Test Object:  
Initial testing involved placing a cube on top of an image plane.  
Cube positioned and scaled appropriately.  
Establishing correct projection for cube to be child of image target.

Testing Output:  
Initiated play mode to check the reliability of AR projection.  
Generated lighting in response to adjusting to dark appearance.  
Cube correctly imaged projected onto the image target.

Adding 3D Model:  
  
1)Importing Barbarian 3D Model made:  
A Barbarian 3D Model is made from scratch and the animation is added.  
Ensuring correct importation through explaining import process.

2)Scaling and Positioning Barbarian Model:  
The barbarian model is superimposed over the image target.  
Scale adjusted for appropriate size.  
Images/barbarians set as children of image target.

3. Animating Barbarian Model:   
- Modified Barbarian model animation to legacy.  
- Set to loop the round kick animation selected.  
- Preview and selection of animations.  
  
4.Finalizing AR Output:  
- Round kick animation from the Barbarian model depicted on the image target.  
- Rotation adjusted for proper facing.  
  
Building and Deploying on Android:   
1. Setting Up JDK and SDK Paths:  
- Ensuring that Unity checks the JDK and SDK paths.  
- Described how users can put up these paths.  
  
2. Configuring Player Settings:  
- Company name, product name and an application logo.  
- Avoid transition problems by setting orientation to landscape and left.  
- Using company/product names in adjusted package name.  
- Specify minimal and optimal API levels for Android compatibility.  
- It had deleted Vulkan Graphics API due to possible incompatibility.  
  
3. Building the Application:  
- Plattform for Android in “Build Settings”.  
- I imported the sample scene into Build Settings.  
- Generate the .apk file by clicking Build.  
  
4. Transferring .apk to Android Smartphone:   
- I used Android File Transfer for Mac to connect and transfile files.  
- Analyzed the procedure for users of Windows systems.  
  
5. Installing and Testing on Android Smartphone:  
- I copied a file .apk for example onto my sd card on the smartphone.  
- Using the Files Android app, installed the application.  
- The ARONE app was successful testing on the android phone.  
  
Conclusion:  
It is proven that the mobile version of the ARONE is adapted for implementation and runs properly on the Android platform by testing it on a smart phone. It presents a detailed description of how to develop an AR application with Unity VUF-Engine.