

# Virtual Reality Concept for entertainment

Anusha Sajjan Gandla, Hari Kishan Darapaneni

**Abstract—** This project is to provide entertainment which is one of the extensions to the virtual reality concepts among many. This android game for Android mobile device platforms is to function as a motion gaming emulator, a simulated 3-D virtual world rendered with the Open Wonderland environment. Our android game is intended to give a lag free and seamless motion gaming experience and may also be transferrable for future applications.

This would actually be very interesting as it's a complete new concept and the user would enjoy playing this. This extensions development would have a great future due to its better concept for graphics and real time applications. This simulates the users and gives them a whole lots of real time experience. This provides user friendly environment and also helps to gain a good knowledge on modeling and animation concepts. This has a great vision on future development and exposes users a good environment to work on. This project brings a good application to users and has a real time gaming experience and also to know more about 3-D applications and work with them.

**Index Terms—** Virtual reality, emulator, seamless, perspectives, oculus rift, gear VR, tracker, maze, packaging, SDK (Software Development Kit).

## INTRODUCTION

Since today majority of the smart phone devices run on Android, thus this game targets the users as well as the Android market. As a viewer of virtual game world, this application renders the areas of 3-D world not immediately visible on a set of player's screens. The main aim is to allow the user to "view" and render the segments of the game world not currently visible on-screen but by means of traversing the space between sets of monitors— corresponding to actual space in the game environment and rendering this view onto an Android mobile device. This will provide players with a complete cycle i.e. 360-degree view of the game world without the restrictions imposed by having a fixed number of perspectives as one could find in a typical multi-monitor setup.

The game in our project contains the spherical ball which moves in the direction of tilt of the phone. The user interface will be easy to use and allow the user to navigate within the game with ease. The android operating system uses a market place to sell applications for the phone.

The Open Wonderland Java environment doesn't render efficiently— or in some cases —on the Android or IOS operating systems. Hence it is necessary that we render the Open Wonderland software on a remote PC and utilize OpenGL to transmit this data to our mobile devices. OpenGL is a set of protocols and methods that allow a mobile device to render 3D objects optimally; this will serve as our primary means of interfacing between the hardware and software components of the products.

The input to this game is the attention of the users and their strategies; and the output is the entertainment which will be the consequence of the input. This has a greater future scope into the process of providing entertainment using virtual reality concepts and it's cost effective and efficient and encourages developers and users towards it.

## Research methodology:

Our system is based around a main menu through which the game can be accessed. At any time the user can exit a game, which brings the use back to the main menu.

## Main Menu description

The main menu is the first thing a user will see when the application is opened on their Android mobiles.

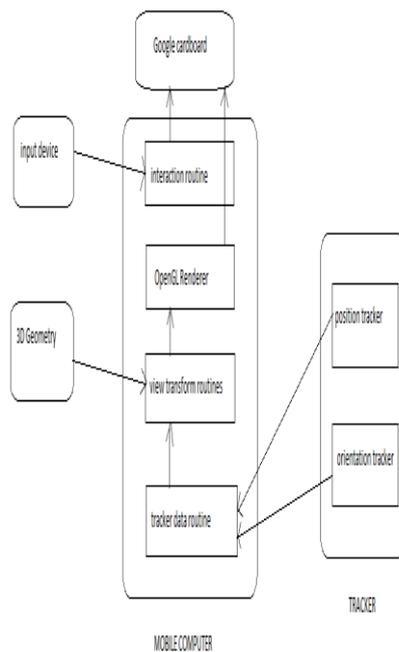
## Processing narrative for Main Menu

From the main menu, the user can select the game or choose to exit the application. Upon selecting the game, the particular activity of that game will be launched. If the user chooses to exit, the application will be closed and the user is returned to the Android OS interface.

## Description of the Main Menu interface

The main menu will consist of a simple icon for starting the game as well as for choosing to exit the application.

**Description of proposed framework**



**ALGORITHM (END USERS)**

- 1) The user must take the .apk file in his/her mobile phone with Android OS above 4.0
- 2) Then it should be installed in the device
- 3) This brings an application VR\_Mobile in the device
- 4) On opening the application we get the game started which is developed by Unity engine
- 5) The user must fix the mobile in the cardboard exactly according to the requirement
- 6) User perspective environment is seen here
- 7) There is a point of vision in the game world which is the actual game simulator
- 8) The user has to point that point on to the arrows of the ball to apply force on the ball
- 9) Immediately on pointing, a force is applied on the ball in the applied direction shown with an arrow symbol
- 10) So, this moves the ball
- 11) There is a path in the environment which the user has to complete
- 12) The end point is shown with a green light indicating the level is over
- 13) This game has 2 levels which automatically switches after each level completion.
- 14) Thus, the game is complete when the user completes both the levels
- 15) This has a loop repeated if to be demonstrated continuously to other users.

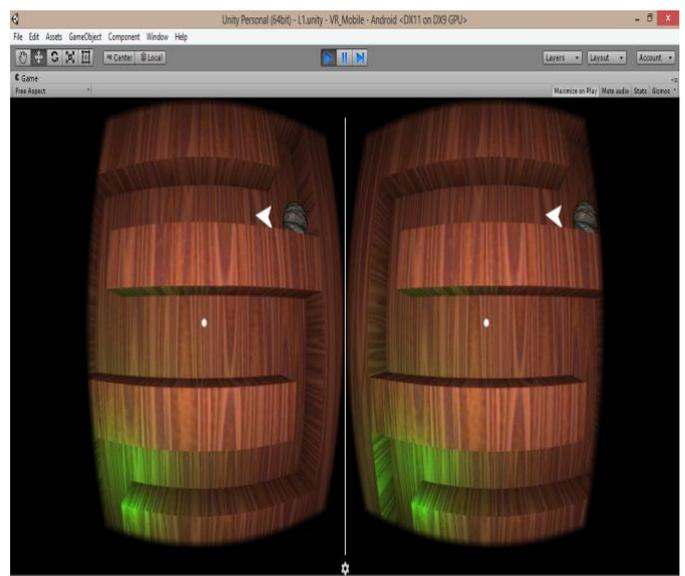
Certain software development kits for developing Cardboard applications would be provided by the google one for

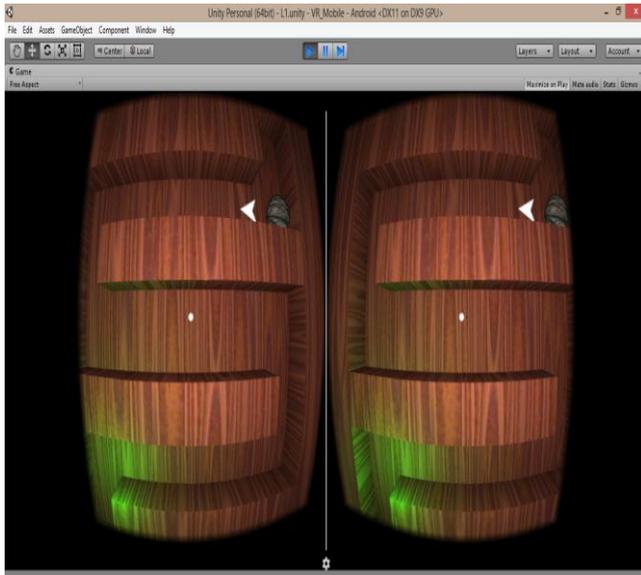
the iOS operating, one for Android operating system using Java, and one for the game engine Unity using C#, and. Firstly, google supported only android platform, but later Google announced its iOS support for the Unity plugin at the Google I/O 2015 conference. Other apps with Cardboard support are available on App Store for iOS and the Google Play store. Also with the traditional native Cardboard apps, there are Google Chrome VR Experiments implemented using WebGL; phones, including Apple's IOS, that support WebGL can run Google's web based platform experiments. A sample demonstration of the app to iOS was released at Google I/O 2015. In January 2016, It was announced by Google that the software development kits would also support spatial audio, a virtual reality effect that is intended to simulate audio coming from outside of the listeners head located anywhere in 3D space.

**Implementation details**

- This android game supports only a single user at a time.
- Android devices vary in technology supported/ capabilities, and hence we can't guarantee the universal access to the application across all the Android platforms.
- Google updates the Android operating system regularly and future iterations may not be compatible with current versions of Unity3D.
- If the developer doesn't maintain the software, following release may result in compatibility issues in the future.

**Screen\_shots**





**Unity engine**

**User Interface**

**1. The Main Menu**

<b>Component</b>	<b>Action</b>
Ball	This is the game object under the user control. It is programmed to mimic the user movements.
Direction Arrows	They determine the direction of the ball movement when a user imposes the dot over them. There are two arrows, each at either side of the ball
Dot	This game object always stays at the center of your screen. It is mainly programmed for the movement. Once the user overlaps this dot over the any of the arrows, the ball moves in that direction.
The Maze	This is the game component that changes from level to level. The ball is to be made to pass this maze for the user to play (unlock) the next level.
Green Light	Every maze has a green light which denotes the goal. For an user to unlock the next maze he has to make the ball reach the present goals maze.

**2. game interface**

<b>Button</b>	<b>Action Performed</b>
Play Game	Start a fresh game and clear all the data from the previous game including state, score and level orientation.
Load Game	Fetch the previous game state variables including orientation, level and score.
Exit Game	Save the present game state and quit the game and go back to the Android main menu

**Results and discussion**

**Installation and packaging issues**

The Software will be packaged and distributed as an Android application Package kit (APK) in the Android Marketplace. It will be available to all users of mobile devices running the Android Operating System with a support for Google Cardboard library.

**Legal considerations**

We will be using Unity3D 5.0 Open license with Android SDK loaded and extended to Google Cardboard libraries in accordance to the Android SDK License Agreement distributed by Google.

**Conclusion**

In order for the user to play the game he must primarily be an owner of a Android device with support for cardboard API. That intuitively suggests that the user should have a Google Cardboard to have the experience he game is designed for. Access is a must for the users to the Android Market place which is Google Play Store, in our case where the game apk is available. All development of this game were done on a Windows 8.1 machine. The game objects were modelled in Blender\*ver\*. The game physics and controls scripts were all developed in Unity3D 5.0 with Google Cardboard API loaded. Testing of the application was done on the Unity Emulator and on various Android devices compatible for this game. As long as the play store is available to Google’s Android mobile Operating System users, the software will be marketable, functional and maintainable to both developers and users.

**RELATED WORK (STUDY ABOUT EXISTING WORK, RESEARCH GAPS)**

<b>Model available</b>	<b>User friendly and user experience</b>	<b>Mobility</b>	<b>Efficiency</b>	<b>More advantages</b>	<b>Limitations</b>
play station VR by sony	Yes excellent	partially	Very efficient	High end graphics support with embedded audio and video.	Very costly and installation problems.
Oculus Rift	Yes Excellent	Partially	Very efficient	High refresh rate, low-persistence display, visual fidelity	Costly and other PC requirements like graphic card, RAM, USB ports and OS we use.
Samsung Gear VR	Yes Excellent	Fully mobile	Efficient	Easy to transport, play games, virtual vacation, movies.	Supports only Samsung Galaxy mobiles like note5, S6, S6edge, S6 edge+.
Google Cardboard for Videos	Yes Good	Fully mobile	Efficient	Brand new way to experience movies and other videos Cost effective.	Some apps can't be moved to SD card and supports only limited mobiles.

**Proposed Model**

<b>Model Proposed</b>	<b>Use Friendly and user experience</b>	<b>Mobility</b>	<b>Efficiency</b>	<b>Other advantages</b>	<b>Limitations</b>
Virtual Reality for entertainment using Google cardboard.	Yes Excellent	Fully mobile	Very efficient	Can be used for gaming, images, videos and exploring and is very cost effective.	Supports limited apps.

**References**

[1] "Sony announces Project Morpheus, a virtual reality headset coming to PlayStation 4". Polygon. Retrieved July 16, 2014.

[2] Tony Sperry. *Beyond 3D TV*, Lulu Pres, Inc., November 2003.

[3] "The Inside Story of Oculus Rift and How Virtual Reality Became Reality".

[4] "Oculus User Guide" (PDF). Static.oculus.com. Retrieved July 17, 2015.

[5] VR". Unrealengine.com. Retrieved July 17, 2015.

[6] "Samsung Gear VR - Specs". samsung.com. Retrieved 2015-10-10.

[7] "Samsung Gear VR review". Engadget UK. Retrieved 2015-12-10.

[8] "Introducing the Samsung Gear VR Innovator Edition". www.oculus.com. Retrieved 2015-11-24.

[9] "Google Cardboard – Google". Google. Retrieved June 2, 2015.

[10] "Virtual Reality". Chrome Experiments. Retrieved August 19, 2014. 7

[11] Roudavski, S. (2010). *Virtual Environments as Techno-Social Performances: Virtual West Cambridge Case-Study*, in *CAADRIA2010: New Frontiers, the 15th International Conference on Computer Aided Architectural Design Research in Asia*, ed. by Bharat Dave, Andrew I-kang Li, Ning Gu and Hyoung-June Park, pp. 477-486

[12] Higgins, T., Main, P. & Lang, J. (1996). "Imaging the Past: Electronic Imaging and Computer Graphics in Museums and Archaeology", Volume 114 of *Occasional paper*, London: British Museum. ISSN 0142-4815

[13] Horowitz, Ken (December 28, 2004). "Sega VR: Great Idea or Wishful Thinking?". *Sega-16*. Archived from the original on 2010-01-14. Retrieved 21 August 2010.

[14] Cline, Mychilo Stephenson (2005). *Power, Madness, & Immortality: the Future of Virtual Reality*. [Virtualreality.universityvillagepress.com](http://Virtualreality.universityvillagepress.com). Retrieved 2009-10-28.

[15] Rosenberg, L., "Virtual fixtures as tools to enhance operator performance in telepresence environments," *SPIE Manipulator Technology*, 1993.

#### *Authors*



*Anusha Sajjan Gandla*  
*Department of Computer Science*  
*Vellore Institute Of Technology*  
*Vellore, Tamil\_Nadu, India*



*Hari Kishan Darapaneni*  
*Department of Computer Science*  
*Vellore Institute Of Technology*  
*Vellore, Tamil\_Nadu, India*