Cross-platform Application Development using Unity Game Engine

Pratik P. Patil[1]
Dept. of Computer Engineering,
P.V.P.I.T,
Pune, India

Ronald Alvares[2]
Dept. of Computer Engineering,
P.V.P.I.T,
Pune, India

Abstract: In this paper, we present the design and implementation of the Cross Platform game called Air Hockey. It is a very common game and was developed keeping the Android, Windows, and Windows Phone Operating System in mind. The aim behind this project was to connect two players/users and enable them to play against each other with the help of Wi-Fi, Bluetooth or Remote Server. The players too can play simultaneously unlike the alternate chance games like Pool, etc. The game has been designed and implemented and soon will be available on the Google Play Market and Windows Store. The game has been tested on Windows 7/8 for PC, Microsoft Lumia 620 running Windows Phone 8, Moto G (2nd Gen), Moto X (2nd Gen) running Android Lollipop and Samsung Note 3 running Android KitKat. So, it should run on other compatible devices as well.

Keywords: Cross-Platform, Unity Game Engine, Air Hockey, Software Development, Android Development, Windows Development, Wi-Fi, Bluetooth, Server

I. INTRODUCTION

Unity3D Game Engine is an integrated development tool used to develop interactive contents like video games, architectural visualization and real-time 3D animations. Its editor runs on Windows and Mac OS X platforms. Though it runs on only two platforms, it has the ability of developing applications for multiple platforms which are mentioned as follows – Windows, Windows Phone, Mac OS X, iOS, Wii, Linux, Android, Web Player, etc. All we need for it to develop the application on respective platforms is the software development kit (SDK) for it. There are various plugins to export the application to different web players like Flash Player, etc.

The functions that are supported by Unity3D are very abundant. All type of game developments are possible with Unity3d such as shaders, physics engine, network, terrain manipulation, audio, video and animation. Unity3D produces the applications based on JavaScript and/or C#. These are used to assign the animation or real-time transition of the Game-Objects defined in the application. GUI of Unity3D helps a new developer to approach easily, and script and program the transition of the Game-Object.

This study aims to design and develop applications on multiple platforms using Unity3D Game Engine and with the help of scripting and programming enable networking between any two devices so, the users of same/different devices running same/different platforms can play against each other. For the efficiency of the game development process, this study aims to plan, design, develop and test a smart game based on multi-platform game engine.

II. RELATED STUDIES

2.1 Introduction to Unity3D

Now-a-days the devices like Smartphones, tablets, iPad, iPhones have taken the application development to a whole another level. The cameras on these devices can be used to recognize business card and decrypt encrypted codes like barcodes, QR-codes, etc. by simply scanning them. Here in this era, networking is also a major factor, as many or mostly all applications running no-a-days require a proper network connectivity. So, we have come up with an idea of using this network connectivity to next level by using this cross-platform connectivity in the gaming applications. Usually, we have heard about handheld-to-handheld connectivity (i.e. Android - Android, iOS - iOS, Windows Phone - Windows Phone, Android-iOS, Android - Windows Phone, iOS - Windows Phone), but our aim is to enable all-to-all device connectivity so there can be Handheld – PC along with above connections.

Unity3D is a cross-platform game creation system developed by Unity Technologies containing game engine and an integrated development environment (IDE). It is used to develop games for websites, desktops, handheld devices and consoles. Previously released on Mac OS in 2005, it was then expanded to target more than fifteen platforms. It can be run on Windows or Mac OS X system. Supported Platforms include BlackBerry 10, Windows Phone 8, Windows, OS X, Linux(mainly Ubuntu), Android, iOS, Unity Web Player (including Facebook), Adobe Flash, PlayStation 3, PlayStation 4, PlayStation Vita, Xbox 360, Xbox One, Wii U, and Wii. It includes an asset server and Nvidia PhysX physics engine.

The latest version is the Unity 5 released on March 3rd 2015. It supports the real-time global illumination based on the Geometrics Enlighten technology. Other major changes include physically-based shaders, HDR sky-boxes, reflection probes, a new audio mixer with effects and enhanced animator workflows. Unity 5 brings support for Windows, Mac, Linux/Steam OS, Unity Web player, Android, iOS, Blackberry 10, Windows Phone 8, Tizen, Windows Store apps, WebGL, PlayStation 3, PlayStation 4, PlayStation Vita, Wii U. Xbox One, Xbox 360, Android TV, Samsung Smart TV, Oculus Rift, and Gear VR for a total of 21 supported platforms. The Unity3D game engine is downloadable from their website in two different versions – Unity and Unity Pro.

» Rendering:

The graphics engine uses Direct3D (Windows), OpenGL (Mac, Windows, Linux), OpenGL ES (Android, iOS), and proprietary APIs (Wii). There is support for bump mapping, reflection mapping, parallax mapping, screen space ambient occlusion (SSAO), dynamic shadows using shadow maps, render-to-texture and full-screen post-processing effects.[4] Unity supports art assets and file formats from 3ds Max, Maya, Softimage, Blender, Modo, ZBrush, Cinema 4D, Cheetah3D, Adobe Photoshop, Adobe Fireworks and Algorithmic Substance. These assets can be added to the game project, and managed through Unity's graphical user interface.[5]

The ShaderLab language is used for shaders, supporting both declarative "programming" of the fixed-function pipeline and shader programs written in GLSL or Cg. A shader can include multiple variants and a declarative fallback specification, allowing Unity to detect the best variant for the current video card, and if none are compatible, fall back to an alternative shader that may sacrifice features for performance.[6] Unity also has built-in support for Nvidia's (formerly Ageia's) PhysX physics engine, (as of Unity 3.0) with added support for real-time cloth simulation on arbitrary and skinned meshes, thick ray casts, and collision layers. [7]

» Scripting:

The game engine is based upon the JavaScript and C#. Unity's scripting is built on Mono, an open-source implementation of .NET Framework.[8] MonoDevelop is an open-source IDE for Linux, Mac OS X and Windows. It supports Boo, C, C++, C#, CIL, D, F#, Java, Oxygene, Python, Vala and VB .NET.[9][10]

» **Graphics:**

❖ Physics: Unity utilizes the NVIDIA PhysX physics engine, which supports:

1. Cloth

2. Soft and rigid-body interactions

3. Ragdolls

4. Joint systems

5. Wheel collision system[11]

❖ Terrain:

1. Terrain Painting

2. Detail Texture Painting

3. Tree Creator [11]

» **Networking:**

Unity has a number of built-in features to facilitate creation of standard multiplayer games.

Some of the features State Synchronization, Real-time Networking, Remote Procedure Calls, Backend Connectivity. Web integration allows the game to communicate with container web pages and services. Massive Multiplayer Online: Development requires additional work by the developer or the addition of one of several third-party software packages designed for that purpose. [11]

» **Asset Tracking:**

Unity also includes the Unity Asset Server - a version control solution for the developer's game assets and scripts. It uses PostgreSQL as a backend, an audio system built on the FMOD library (with ability to playback Ogg Vorbis compressed audio), video playback using the Theora codec, a terrain and vegetation engine (which supports tree bill boarding, Occlusion Culling with Umbra), built-in light mapping and global illumination with Beast, multiplayer networking using RakNet, and built-in pathfinding navigation meshes. [12]

» **Platforms:**

Unity supports deployment to multiple platforms. Within a project, developers have control over delivery to mobile devices, web browsers, desktops, and consoles.[13] Unity also allows specification of texture compression and resolution settings for each platform the game supports.[13]

Currently supported platforms include for Windows, Mac, Linux/Steam OS, Unity Web player, Android, iOS, Blackberry 10, Windows Phone 8, Tizen, Windows Store apps, WebGL, PlayStation 3, PlayStation 4, PlayStation Vita, Wii U. Xbox One, Xbox 360, Android TV, Samsung Smart TV, Oculus Rift, and Gear VR.[3]

» **Unity Asset Store:**

Launched in November 2010, the Unity Asset Store is a resource available within Unity Editor. Store consists of over 7000 asset packages, including 3D models, textures and materials, particle systems, music and sound effects, tutorials and projects, scripting packages, editor extensions and online services. The store also contains many extensions, tools and asset packages such as the package NGUI: Next-Gen UI by Tasharen Entertainment,[14] and the visual scripting extension uScript by Detox

Studios, Tidy Tile Mapper, a 2D/3D tile-based game design extension by Doppler Interactive and the input scripting package FingerGestures.

2.2 Autodesk 3ds Max:

Autodesk 3ds Max is a professional 3D computer graphics program for making 3D animations, models, games and images. It is developed and produced by Autodesk Media and Entertainment. It has modeling capabilities, a flexible plugin architecture and can be used on the Microsoft Windows platform. It is frequently used by video game developers, many TV commercial studios and architectural visualization studios. It is also used for movie effects and movie pre-visualization. In addition to its modeling and animation tools, the latest version of 3ds Max also features shaders (such as ambient occlusion and subsurface scattering), dynamic simulation, particle systems, radiosity, normal map creation and rendering, global illumination, a customizable user interface, and its own scripting language.[15]

Modeler and texture artist could increase the speed of working and efficiency because of extensive polygon modeling and texture mapping tool set. CAT (Character Animation Toolkit) that is completely integrated provides advanced rigging and animation system that can be used immediately. Artist could create very realistic or great image by using mental ray renderer and integrated 3ds Max scan line including mental ray network that can finish rendering faster. Since it supports C++ and .NET, developer could customize and expand 3ds Max and also integrate it into the existing pipeline. Also, developer could create 3D model with Autodesk 3ds Max, send it to Autodesk FBX exchange technology file after animating it, and then bringing it to Unreal Editor Contents production software.

2.3 Importing from 3ds Max to Unity:

The things that are imported from 3DS MAX from Unity3D can be largely divided into five parts. First, it is every node that has position, rotation, and scale. Second part includes pivot point, names, vertex, and meshes with colors. Third, they are normal and one or two UV sets. Fourth, they are diffuse texture and materials with colors. Fifth, they are many materials per mesh, animation, and bone based animations. Thus, these five parts are the ones that Unity recognizes from 3DS MAX. Also, Unity3D could import FBX file. FBX provides supports for software and applications that are both independent and from certain companies. FBX is used within entertainment pipeline or as a part of design production. Files can be sent more smoothly, more data is maintained, and the work flow is more efficient. To send a file from 3DS MAX to Unity3D, first convert to FBX file with export from 3DS MAX, and send it. Then the file can be imported from Unity3D. Only, if the name of FBX file is in Korean, Unity3D cannot recognize it [1] [2].

2.4 Photon Unity Networking Plugin:

Photon Unity Networking (PUN) plugin is used to provide applications multiplayer support using networking. It implements Unity's well known networking API on top of Photon. Keeping API simple, this gives you standalone, dedicated servers, fixes, tweaks and ongoing support. This plugin contains a setup wizard, the fully integrated API Editor extensions and a client library. It is compatible with a Photon Cloud Service i.e. Photon's Server. A developer can use this Photon Server or setup a custom server for his developed game. There are two types of packages for Photon – Photon PUN Free and Photon PUN+. [16]

The Photon PUN version can be downloaded for free from Unity Assets Store and has a few limitations. For a developer to export his game to Standalone, Web, Win 8 Store, Win 8 Phone, he can use Photon PUN Free. But then if he needs to export to Android and iOS, then he will need to buy Photon PUN+ or Unity Pro. Photon PUN+ has a limit of supporting over a 100 user playing concurrently. [16]

III. BODY

3.1 Summary of the Game Development:-

In this research paper

1. *Game Name:* Air Hockey.

2. *Game Genre:* Arcade, Board.

3. *Platform:* Supports Multi-platform (Windows PC, Android, etc.)

4. *Game Characteristics:*

Air hockey is a popular game of sliding the puck in the hole. Similar to Tennis and Football. It is played on a table top with one player at either end. Each player holds a small mallet in their hand and uses it to hit the disc, known as a puck across the table onto the opponent's goal. The puck travels when the air is pushed through small holes in the table reducing the friction between the puck and table. It then increases the speed of the puck enabling it to travel for a further or stronger distance as you swipe the puck across.

There are two players involved in the game. Each player has to defend his pocket and score a goal by hitting the puck in the opponent's pocket. Each player controls a paddle with which he can move to defend his pocket and score a goal.

Method of playing involves utilizing a touch function for the Android devices and a mouse pointer function or keyboard for a Windows PC.

5. *Core Idea of the Game:*

» Each player has the job of defending his own pocket in the game with the help of a round paddle.
» There are modes in the game:-

    a. Single Player – A single real-time player plays with an artificial intelligent opponent.

    b. Two Player – Two Players involved in one game, playing on the same device.

    c. Multiplayer Online – A single real-time player plays with a random opponent available on the server on their own respective devices. Option is also there for the player to connect to friends using Facebook connection.

» In the Single Player Mode, the Artificial Intelligence is coded in such a way that when a player is playing, it will adjust according to the game of the player making it hard and challenging for the player.

Pratik et al.,

International Journal of Advance Research in Computer Science and Management Studies
Volume 3, Issue 4, April 2015 pg. 19-27

The game working can be seen from following chart:
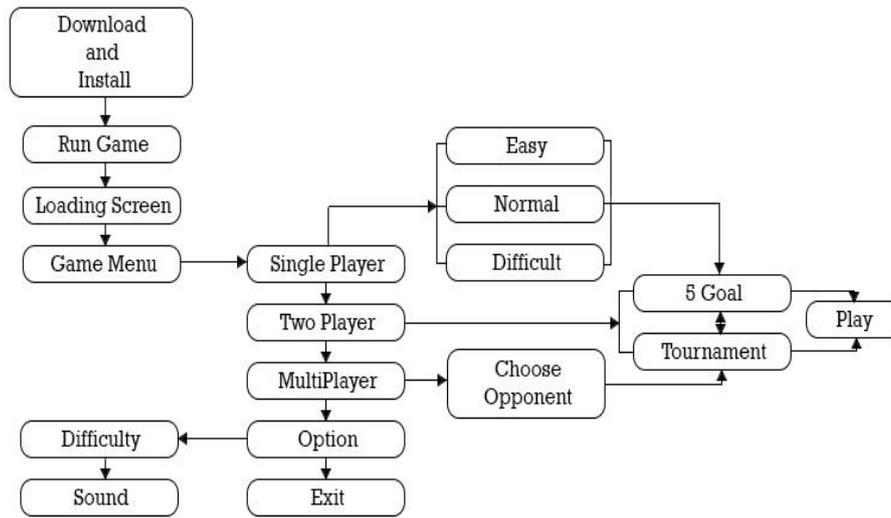
Fig. 1 Game

3.2 Graphic Concept Art:-

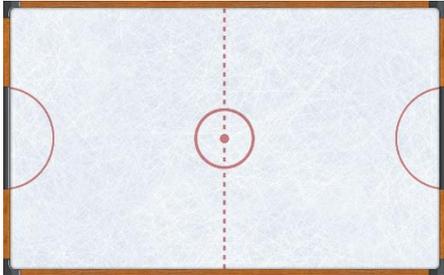**Table 1. Character Designing and Modelling**

Description	Image
This is the design of the how the table will look like. This will be in the background and there will be colliders added to it along the inner edges of table. These table surfaces are frictionless. So the paddles and puck move freely. This layout has been created in Adobe Photoshop and Autodesk 3ds Max	Fig. The Air Hockey Table
This is the design of Paddles that the players will be controlling. This will be moved by the players using touch on handheld devices or a mouse pointer on their computer systems. Designed in Adobe Photoshop and modeled in Autodesk 3ds Max.	Fig. The Paddles of Two Players
This is the design of the Puck that the players have to hit and score a goal and prevent it from entering their goal pocket. Designed in Adobe Photoshop and modeled in Autodesk 3ds Max.	Fig. Puck

Table 2. Animation

Description
Now, here we will apply animations to the paddles and puck using the scripts like Boo, JavaScript and/or C#. We can apply transition animation, assign keys to move the paddles. Adjust the animation keys with the auto key and put in the motions appropriate to the plan that has been set up.

Table 3. Interface

Description
The menu was designed in Adobe Photoshop with a next gen gaming font and animations. The game being played in the background of menu was pre-recorded. The buttons were also designed in Photoshop with an Inactive, Hover and Active State in mind. When clicked there is a sound and a special effect given to each button.

3.3 Game Development

3.3.1 Game UI Screen

1. Main Menu: It is the very first menu that is displayed when the game is executed. It shows the story and instructions of how to play the game. There are different modes for the user to select – Single Player, Two Player, Multiplayer, Options and Quit.

2. Single Player: When this mode is selected, the difficulty menu is displayed which displays three modes – Easy, Normal and Difficult. The opponent will then be adjusted according to the mode selected. Once selected there is one more menu displayed before game starts. This menu asks if you want to play 5 goals match or a tournament. Once selected, the game can be played.

3. Two Player: When this mode is selected, one more menu is displayed which asks if you want to play 5 goals match or a tournament. Once selected, the game can be played.

4. Multiplayer: Once selected there is a menu in which you can play with a random opponent or log in with Facebook and play with friends. After this one more menu displayed which asks if you want to play 5 goals match or a tournament. Once selected, the game can be played.

5. Options: This is a menu in which you can change the volume of music and sfx. You can also change the sensitivity of the paddle.

6. Quit: Confirm whether to end game or not.

3.3.2 Gameplay

Single Player:

Single Player: When this mode is selected, the difficulty menu is displayed which displays three modes – Easy, Normal and Difficult. The opponent will then be adjusted according to the mode selected. Once selected there is one more menu displayed before game starts. This menu asks if you want to play 5 goals match or a tournament. Once selected, the game can be played.

Two Player:
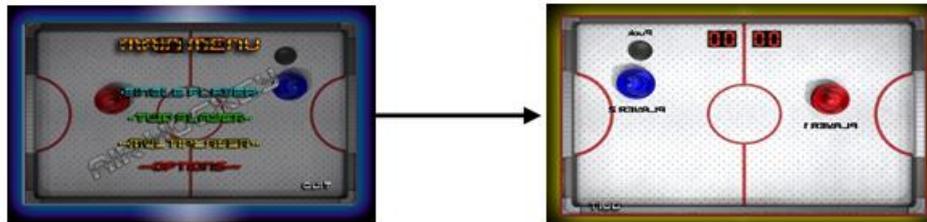
Two Player: When this mode is selected, one more menu is displayed which asks if you want to play 5 goals match or a tournament. Once selected, the game can be played.

Multi Player:

Multiplayer: Once selected there is a menu in which you can play with a random opponent or log in with Facebook and play with friends. After this one more menu displayed which asks if you want to play 5 goals match or a tournament. Once selected, the game can be played.

IV. FUTURE SCOPE

Cross platform Applications have a great future ahead. But there are few drawbacks when compared with single-platform regarding fluidity, seamless and device-optimized experience. When an application is developed on a single-platform, then using the same logic / code the developers can develop same app for multiple platforms and enable them to connect with each other. Hence there is bright future ahead regarding the Cross Platform Application Development.

V. CONCLUSION

This study aims to design and develop a cross-platform gaming application using Unity Game Engine. The functions that Unity3D supports autonomously are very abundant. All game developments are possible such as shader, physics engine, network, terrain manipulation, audio, video, and animation, and it is enabled so that it is possible to revise, meeting demand of user according to the need. For the efficiency of game development process, this study aims to plan, design, and develop a game based on multi-platform game engine using unity networking plugin called the Photon Unity Networking.

ACKNOWLEDGEMENT

References

1. "Is Unity Engine written in Mono/C# or C++?" http://answers.unity3d.com/questions/2187/is-unity-engine-written-in-mono-c-or-c. Retrieved 2011-04-26..

2. "Unity - Fast Facts". http://unity3d.com/company/public-relations/. Retrieved 5 February 2013

3. "Unity - Multiplatform". Unity Technologies. http://unity3d.com/unity/multiplatform/. Retrieved 5 February 2013.

4. "Using DirectX11 in Unity 4". Unity Technologies. http://docs.unity3d.com/Documentation/Manual/DirectX11.html. Retrieved 19 February 2013..

5. "How do I import objects from my 3D app?" Unity Technologies. http://docs.unity3d.com/Documentation/Manual/HOWTO-importObject.html. Retrieved 19 February 2013.

6. "Shaders". Unity Technologies. Retrieved 19 February 2013.. http://docs.unity3d.com/Documentation/Manual/Shaders.html.

7. "Physics". Unity Technologies. Retrieved 19 February 2013. http://docs.unity3d.com/Documentation/Manual/Physics.html.

8. "Using Scripts". Unity Technologies. http://docs.unity3d.com/Documentation/Manual/Scripting.html. Retrieved 19 February 2013

9. "MonoDevelop". Mono Technologies http://www.monodevelop.com/documentation/

10. "Getting started with Mono Develop". Unity Technologies. http://docs.unity3d.com/Documentation/Manual/HOWTO-MonoDevelop.html. Retrieved 19 February 2013.

11. "Unity Features", GameIndustry.com http://gameindustry.about.com/od/game-development/a/Unity-Engine-Feature-Guide.htm

12. "Asset Server (Pro Only)". Unity Technologies. http://docs.unity3d.com/Documentation/Manual/AssetServer.html. Retrieved 19 February 2013.

13. "Unleash your game with effortless deployment to 10 global platforms". Unity Technologies. http://unity3d.com/unity/multiplatform/. Retrieved 19 February 2013.

14. "NGUI: Next-Gen UI kit". Tasheren. http://www.tasharen.com/?page_id=140.

15. "Autodesk 3ds Max — Detailed Features", 25 March 2008 http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13567426

16. "Unity Networking with Photon", Exit Games   http://doc.exitgames.com/en/pun/current/getting-started/pun-intro