

Unity tutorial report: Making an action role playing game (Diablo II style)

T. M. Horey (50%), J. Moran-Perez (50%)

Table of Contents

Introduction.	1
Unity.	1
Unity3D: RPG Tutorial (Diablo Style).	1
Unity Animation Tutorials.	3
Unity3D:Terrain Tutorials.	4
Unity3D:Using Height Maps.	5
Demonstration.	5
Summary.	16
References.	18

Introduction

For our Project 3, we set out to complete tutorials for Unity so that we could use what we learned to build a small Diablo II style action role playing game. We were fortunate enough to find a tutorial that follows almost exactly what we wanted to do and along the way we supplemented with other tutorials from Unity to fill in the gaps. We go over why we chose Unity, and then cover what we did and what we learned from each of the tutorials. After this, we describe the game we made upon the completion of the tutorials. Most of the later pages are screenshots, which we fit one to each page so they could be as large as possible for detail. In the end, we decide that using Unity had its ups and its downs, but overall we liked the experience and what we accomplished.

Unity

After considering between Unity and Unreal Engine, Unity became the game engine of our choice for several reasons. First, we like how Unity is very flexible and able to support so many platforms. It is convenient to have one engine that can build for PC, consoles, mobile, and web. Second, we have been exposed to Unity more than Unreal Engine, especially because in the last couple years as there have been many indie games released that developed with Unity. Lastly, Unity seemed to have a lot more helpful content and tutorials available. Finding help online was simple, either from the Unity documentation or from other users on the forums. Since we both had no experience developing with modern game engines, Unity seemed like the way to go.

Unity3D: RPG Tutorial (Diablo Style)

The main tutorial that we followed was a youtube video series entitled “Unity3D : RPG Tutorial Diablo Style” (Jain). There are 16 videos in all, each 20 - 40 minutes in length. The video titles and content description are as follows:

- Unity3D: RPG Tutorial (Diablo Style) Session 1(Updated Version): Click To Move
 - Setting up a new Unity project & learning how to move your character by clicking on the terrain
- Unity3D : RPG Tutorial (Diablo Style) Session 2: Animation and Camera Follow

- Learning animation and setting the camera to follow your character
- Unity3D: RPG Tutorial (Diablo Style) Session 3: Combat System (1)
 - Create an enemy and have it chase your character
- Unity3D: RPG Tutorial (Diablo Style) Session 4: Combat System (2)
 - Attacking enemies and related animation
- Unity3D: RPG Tutorial (Diablo Style) Session 5: Combat System (3)
 - Continued from previous video
- Unity3D: RPG Tutorial (Diablo Style) Session 6: Combat System (4)
 - Implementing a health system and death + animations
- Unity3D: RPG Tutorial(Diablo Style) Session 7: Enemy health bar
 - Using Unity's GUI system to display enemy health
- Unity3D: RPG Tutorial(Diablo Style) Session 8: Enemy health bar Part 2
 - Continued from previous video
- Unity3D: RPG Tutorial(Diablo Style) Session 9: Leveling System
 - Earning experience from enemy kills and increasing stats on level up
- Unity3D: RPG Tutorial(Diablo Style) Session 10: Stunning
 - How to interrupt an enemy and stun for a certain time on attack
- Unity3D: RPG Tutorial(Click To Move Style) Session 11: Special Attacks
 - Continued from previous video + extra damage
- Unity3D: RPG Tutorial(Diablo Style) Session 12: Projectiles
 - How to implement different types of projectiles to damage an enemy
- Unity3D Tutorial: Object Pooling
 - Creating an object pool to save resources
- Unity3D: RPG Tutorial(Diablo Style) Session 13 : Action Bar
 - Creating an action bar to keep track of character stats
- Unity3D: RPG Tutorial(Diablo Style) Session 14 : Action Bar 2
 - Continued from previous video
- Unity3D: RPG Tutorial(Diablo Style) Session 15 : Saving
 - How to saving gamestate info between game sessions

The videos were mostly easy enough to follow, except for the content related to animations and special attacks.

Regarding animations, this tutorial is several years old and Unity has since deprecated the old animation system so all of that content was completely unusable. We were able to work around it by learning the new animation system. The new system seemed more difficult to learn and use, but it has a lot of functionality.

Regarding the special attacks, we tried to implement them as shown in the video - in a separate script attached to the player character. This never worked for us and the attacks would not register, as if the special attack button was not pressed. We think this may be related to the change in how we implement animations, as some of our methods to control attacks are a bit different. We ended up programming the special attacks alongside the basic attacks in the main fighting script just so we could continue with the tutorial. We would like to come back to it and have special attacks be its own script eventually.

The only content that we did not implement into our game is object pooling and saving. We know how they are done, but our game is so small we decided not to spend too much time on this because the benefit would be minimal. Object pooling is one of the last videos and it is demonstrated using a new project with 3D shapes, not the tutorial project. If this were one on the first videos we think it would have been easier to implement.

Unity Animation Tutorials

To learn how the latest version of Unity does animation, we watched nearly all of the videos concerning animation from the tutorials on the Unity website ("Animation."). The most helpful section was "Controlling Animation." The old animation system seemed much more simple: animations could be played by having them added as assets and just calling them with "animation.Play()". The new animation system requires an animator controller and an avatar for each object to be animated. With this you can

build an animation state machine for your avatar. The conditions to switch from one animation state to another can be changed in the code. In the end, the new system is more complex but has a lot of customization ability.

Unity3D Terrain Tutorials

This tutorial basically explained what you needed to do to create terrain in your scene. It went over things. Basically every terrain is a 3D game object that is added to your scene. From there you can manipulate it and add different textures to the specific part of the terrain that you are working on. So for example for the ground we used a grassy terrain as our base and template. Then we added different textures for the mountains and for the road. For the mountains we used a rock texture that we got from the asset store and for the road we used an asphalt texture for the road. Unity allows you to blend different textures in order to create variation between your textures. We did this for the mountain texture. We used two rock textures one light and one dark and blended them together to create the effect that we wanted. We also added trees to our scene. We used two different types of trees in our terrain. We used a medium pine tree model in the left side of the map and a medium maple tree in the right side of the map.

Overall there was a lot of tools to use in this tutorial. Some were a lot harder to implement and where more for intermediate users so we didn't end up including these in our project. Additionally some where kinda unnecessary and didn't fit into the theme of our project. For example Unity also allows you to edit the wind in your scene but we felt that this wasn't necessary so it wasn't included in our project. Additionally there are tools to edit your trees and slant them and change the the size of the leaves and the density of the branches but we didn't implement these thing because we felt that the users would barely notice these things.

Unity Tutorial Using Height Map

Overall this was a very long tutorial and it was over an hour long. This was a tutorial aimed towards more advanced user. We didn't implement the majority of the things showed in this tutorial. We should implemented the basics in the beginning.

So for example the main takeaway from this project was that there are tools that you can use to edit your terrain by using a brush to scroll over your terrain and that raises points on your terrain. The longer you hold it the more height you get and this is how you can create sharp points and edges in your terrain. This is what we used to make that huge mountain in our scene.

For the other parts of our scene that weren't too steep we used the smoothing feature tool that allowed us to smooth the edges of our terrain to make this more walkable and not too jagged. Overall, this tutorial was helpful because it helped us make the various mountain and hill features that gave our terrain variety and added to the complexity of the scene.

Demonstration

The game we made is one scene with a player character and two types of enemies. The player character has an idle and run animations and a basic attack, a special attack (more damage and stun), and a magic attack (more damage and long range). The enemies also have idle and walk/run animations and a basic attack. When in combat, a health bar for the current enemy is displayed at the top of the screen. An action bar is always displayed at the bottom of the screen and contains the following info: health, mana, experience, and keys for the special and magic attack. Mana is used when casting a magic attack. When health reaches 0, the player "dies" but the game does not end because the player re-spawns with full health after 10 seconds. There are no health or mana potions so they both constantly replenish and do so more quickly when not in combat. The player starts at level 0 and earns experience by killing enemies. At level up the following stats are automatically increased, either linearly or exponentially depending on the stat: player maximum health, player damage, enemy

maximum health, enemy damage, experience required for next level, and experience gained from enemy kill.

The scene in our game is a mountainous/forested terrain with a path winding through it, from one end to the other. Along the path are abandoned houses and hordes of enemies. The lighting is dark and gloomy to mimic a setting sun with smoke in the air.

There are quite a few sound effects in the game. There is a background track that loops with music borrowed from the original Diablo II game. There are various enemy sounds (attack, pain, roar, death), player sounds (attack, pain, death), attack sounds (blade sounds, magic casting sounds), and footsteps for the player.

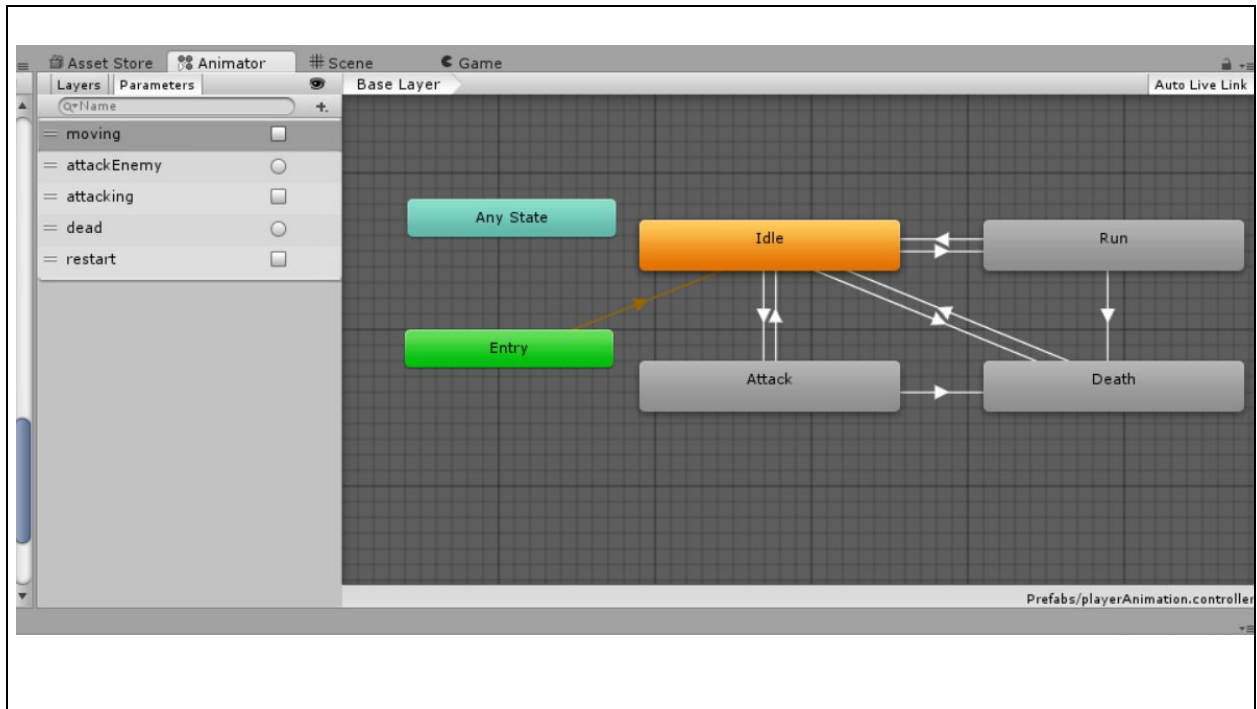
All models and animations were downloaded free from the Unity asset store. The attack animation for the player is wonky because it was made using an older version of Unity and it does not translate well to Unity 5. During a swing, the sword leaves the player's hands, does an arc, then returns to his hands. This can be seen on the screenshot on page 15. We don't mind this too much because it is a free asset and the player does wield magic so maybe that is the explanation.

At the moment, there is no end game. Once all the enemies are cleared there is nothing else to do and nowhere to go. We planned to have another scene with a boss fight but what we currently have took us longer to do than we expected. So the next thing on our list for Unity tutorials would be adding more scenes and transitions between scenes. What we also lack is an in-game menu for things like pausing the game, restarting, saving and exiting, or rebinding key commands.

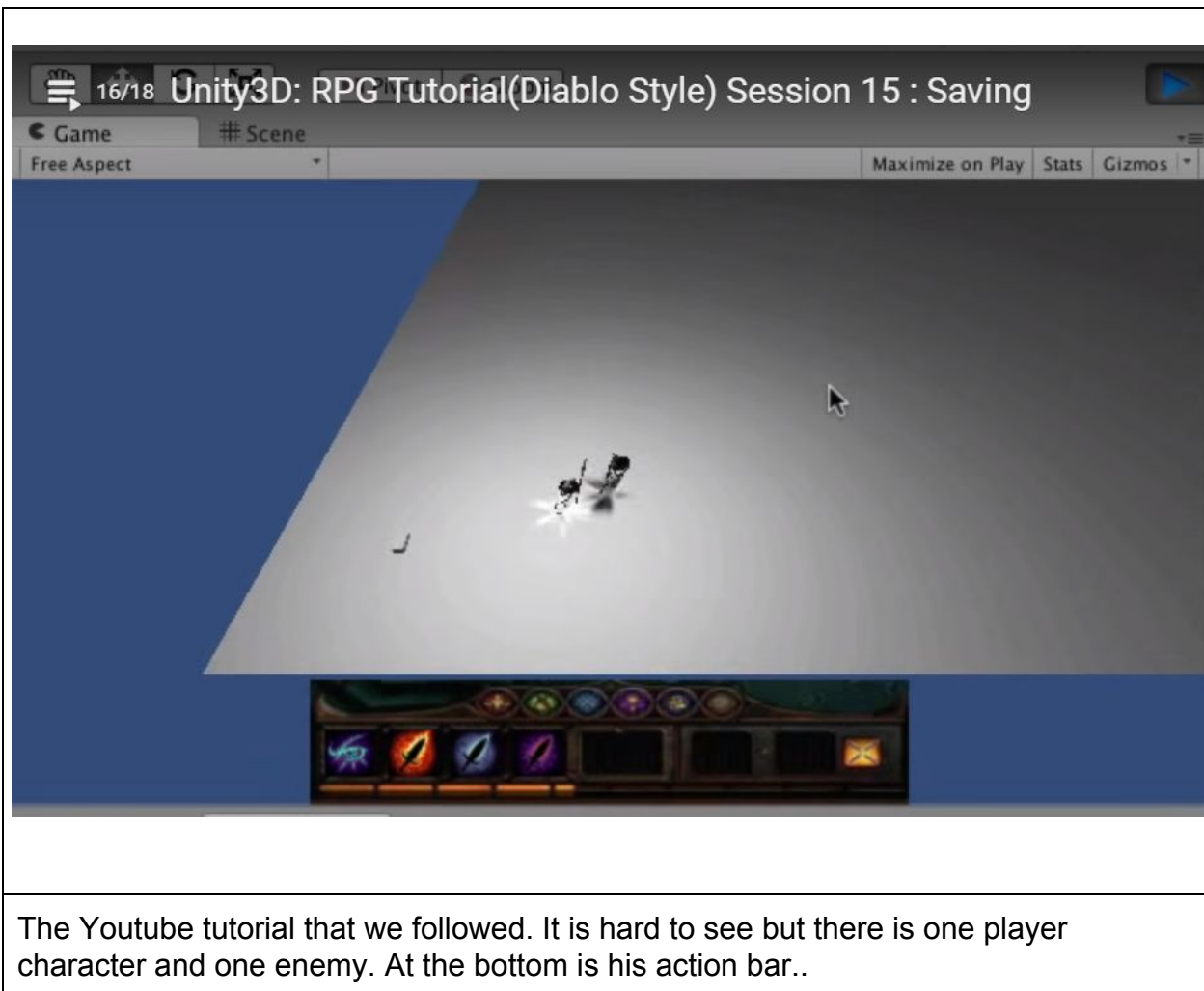
Controls:

- Left mouse click on enemy - basic attack
- Left mouse click on terrain - move to location
- 1 key - special attack
- 2 key - magic attack

Below are all the screenshots:

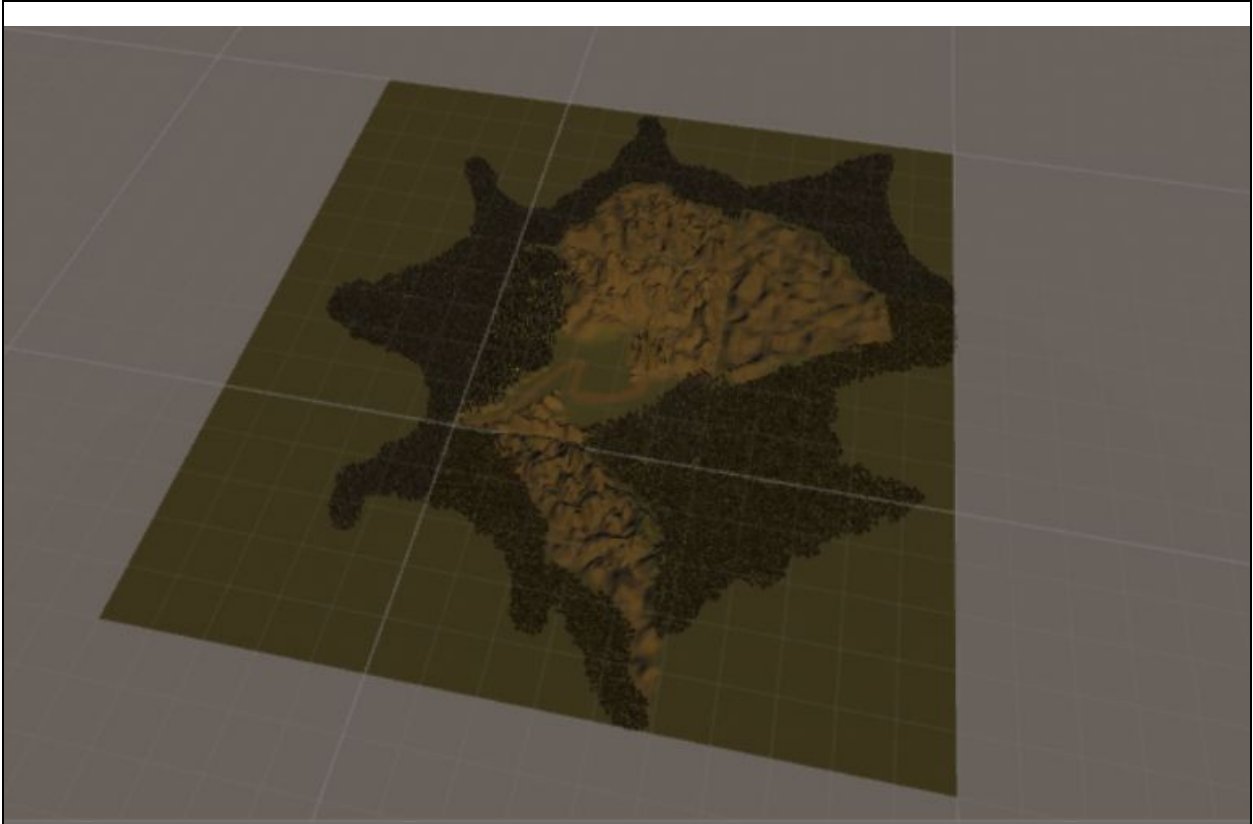


The animation state machine for the player character. The parameters on the left side control the transitions from one state to another.





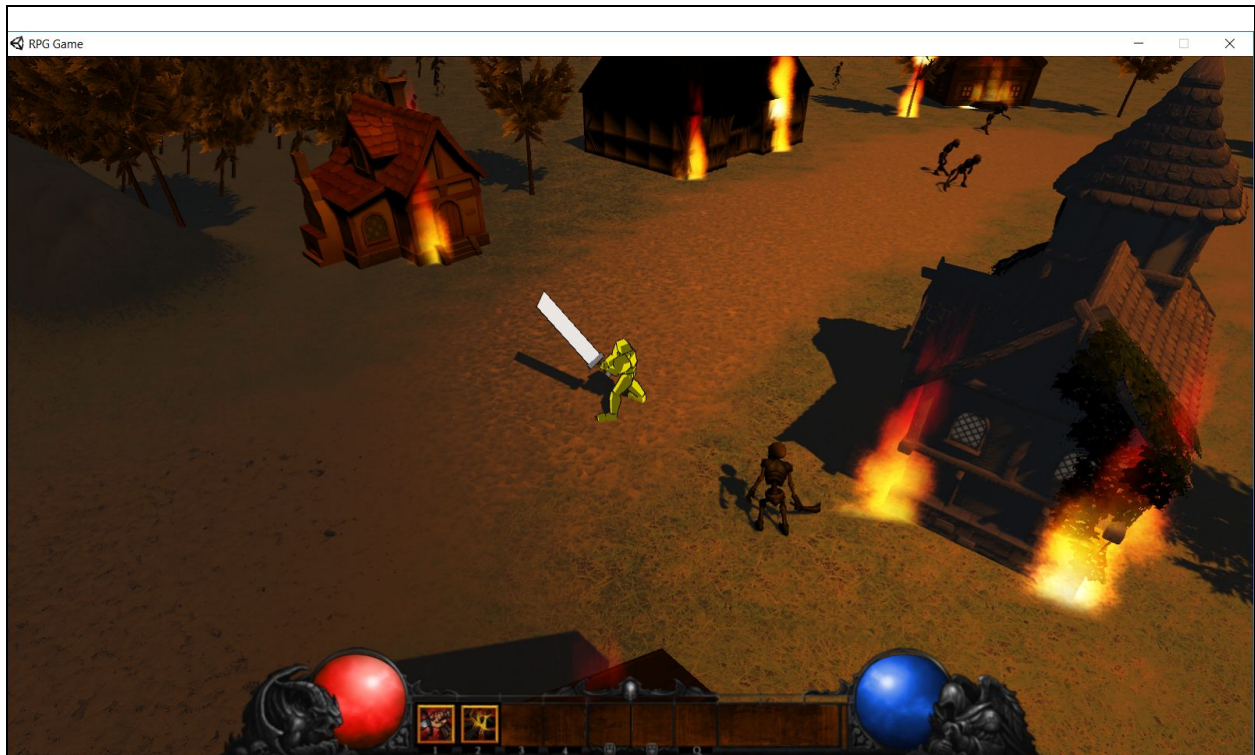
Our project at about half-way through the tutorials.



Our terrain. The area with the path winding through is the walkable area.



Our project. The view when the game first loads. The orbs represent our health and mana. The two tiles represent the special and magic attack for keys 1 and 2. The currently empty bar above the tiles is the experience bar.



A view of the abandoned houses and enemies.



The base of the mountainous area where the path ends. This would have been the place where we would transition to another scene.





Summary

Overall, this was a real fun project to implement using Unity. However there were many pitfalls and obstacles that we had to deal with. For example one of the biggest hurdles was using Github. When making the scene there were various times when the scene would get deleted whenever you pressed the play button. Additionally sometimes when you tried to commit there would be conflicts and it just won't push. Other times Unity would freeze in the middle of doing an action and the only way to fix it was to exit and restart. One time we didn't notice that this had happened when we were deleting an

object we didn't want in the scene, but somehow we ended up deleting the scene itself and lost days of work. We did learn to keep better backups. Overall, there seem to be various problems with the version with Unity and it made working on this project a lot more difficult.

However, we did learn a lot and had a good time with building our own little game. Using Unity was a new experience and we would like to continue discovering more about game development and modern game engines. We found out that it was relatively simple to get the framework of a game going, but putting in the details and the final touches to make it look and feel polished is a large amount of work. Even if there is no end game, we are pleased with what we have been able to accomplish for Project 3.

References

"Animation." *Unity3d*. N.p., n.d. Web. 24 Feb. 2016.

<<http://unity3d.com/learn/tutorials/topics/animation>>

"Introduction to HeightMaps Difficulty:Intermediate" *Unity3d*. N.p., n.d Web 20 Feb. 2016

<<https://unity3d.com/learn/tutorials/modules/intermediate/live-training-archive/introduction-to-heightmaps>>

Jain, Naman. "Unity3D: RPG Tutorial (Diablo Style)." *Game Designers Vault*. Ed. Pete Ellis. N.p., n.d. Web. 12 Feb. 2016.

<<http://www.gamedesignersvault.com/unity3d-rpg-tutorial-diablo-style/>>.

"Unity Documentation Manual." *Unity3d*. N.p., n.d Web. 18 Feb. 2016

<<http://docs.unity3d.com/Manual/terrain-UsingTerrains.html>>