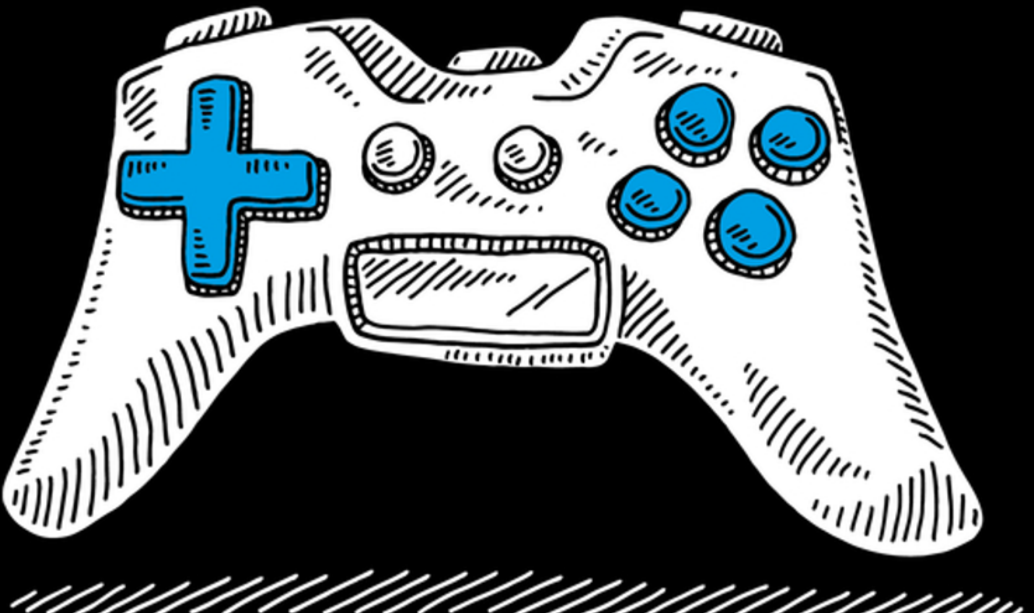
PATRICK McCABE

INSTRUCTOR AT THE DIGITAL ARTS EXPERIENCE

From the makers of for dummies®

# CREATE COMPUTER GAMES

DESIGN AND BUILD YOUR OWN GAME



WILEY

# CREATE COMPUTER GAMES — DESIGN AND BUILD YOUR OWN GAME

by Patrick McCabe

WILEY

Published by: **John Wiley & Sons, Inc.,** 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com

Published simultaneously in Canada

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit https://hub.wiley.com/community/support/dummies.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at http://booksupport.wiley.com. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2017959162

ISBN: 978-1-119-40418-7 (pbk); ISBN: 978-1-119-40422-4 (ebk); ISBN: 978-1-119-40424-8 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

# CONTENTS

# INTRODUCTION

This book explains the basics of game design using the free game development tool Unity. This book was created to make Unity accessible for young adults who are interested in the process of game design but don't know where to begin.

Game design is tough. It involves figuring out and understanding everything from design to coding. Knowing where to begin is daunting and can feel inaccessible. Just keep in mind that no one starts off great at game design. You don't need to know everything to start, and thinking that you do is one of the hardest challenges to overcome.

This book was written to get people like me to start thinking about game design in a different way. This book isn't about making a perfect game or about teaching you everything about game development or even Unity. This book is a starting line. It teaches the basics and encourages you to think about games differently.

## ABOUT THIS BOOK

This book was written with the thought that games are tough to make and that coding is confusing. Think of this like a cookbook that explains why you use two eggs instead of three and what the pepper does to help bring out the flavor. The codes and game development that this book walks you through are approached from the bottom up. Things make sense when you understood *why* certain codes or components are needed. Just knowing the ingredients to a recipe doesn't teach you how to cook.

## FOOLISH ASSUMPTIONS

This book was written for teenagers who have an interest in developing games. People who have spent years working in game design will find this book repetitive and probably not useful at all. I don't pretend that this teaches everything about game design — I know for a fact that it doesn't. This book is an introduction to the field, so it was made for people who have little to no idea where to even begin when they design their games. Some rudimentary typing skills and access to a computer and the Internet are all you really need to read this book. You'll also need a copy of Unity, which you can download for free (I'll show you how).

## ICONS USED IN THIS BOOK

Throughout this book, I use the following icons:

TIP

Tips give some clarifications or offer shortcuts. I use tips to help you understand the program better or do things in an easier way.

REMEMBER

Anything marked by the Remember icon reminds you of principles or ideas that you should think about throughout your game development.

## WHERE TO GO FROM HERE

This book takes you step by step through the process of designing a 3D platformer. If you have a clear idea of what game development is and know about Unity's interface, you can skip the first two chapters and start with Chapter 3, where I dive into actually developing a game.

Chapters 8 through 10 use a separate program known as Blender and go into the basics of animation. If you aren't interested in adding animation or you already have animations you want to include in your game, you can skip those chapters.

All the other chapters build off of each other and are designed to take you through the steps necessary for developing an example game that you'll build with this book.

# CHAPTER 01

# What Is Game Design?

In this chapter, you'll ask yourself questions about not just *your* games but *all* games. When you're designing computer games, you need to think about the fundamental reasons people play games in the first place. The reasons people play strategy games aren't the same reasons they play horror games. Both kinds of games are fun, but for very different reasons. The goal of this chapter is to help you understand what those reasons are so that the games you design deliver what your players are looking for.

Understanding how game mechanics and themes work together to create a gaming experience will help you better visualize and create your own game. If the mechanics aren't fun or don't fit, it doesn't matter how cool the theme is — the game won't be fun. If the theme doesn't match the mechanics, the game might be fun but it won't be memorable. If you aim to create great games, you need to understand both mechanics and themes before you even open the software you'll use to build your games.

## THINKING ABOUT WHAT MAKES FUN GAMES FUN

Have you ever played a game that you couldn't put down? One that gave you the ability to shut off your mind without a care in the world because you were just hooked on it? A game that was just a lot of fun? I'm sure you have! Otherwise, you wouldn't be reading this book.

But why were you sucked into the games you love? Why are they so addicting? What makes them fun?

Fun is where games live and die. If a game isn't fun, nobody plays it. The first thing you have to understand, though, is that there are different types of fun, and different kinds of games:

- **Fighting games** reward quick thinking and reading your opponent's moves and figuring out the proper combo to counter it with.
- **Strategy games** (like XCOM 2, shown in Figure 1-1) challenge you to think and plan for a variety of situations, sometimes in an instant.

**Figure 1-1:** XCOM 2.

- **Adventure games** push you to explore and discover more about the world around you.
- **Horror games** (like Slender: The Arrival, shown in Figure 1-2) get your adrenaline pumping and push your natural curiosity to its limits.
- **Puzzle games** reward you for solving different complex puzzles and for problem solving.
- **Platformers** (like Super Meat Boy, shown in Figure 1-3) work off of reward and punishment and challenge you to master the controls and the timing to perfectly execute a level.

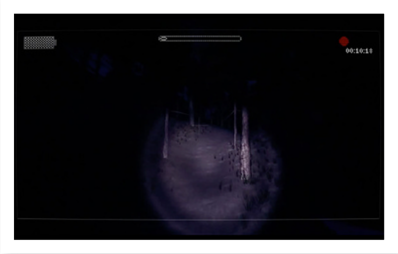


**Figure 1-2:** Slender: The Arrival.

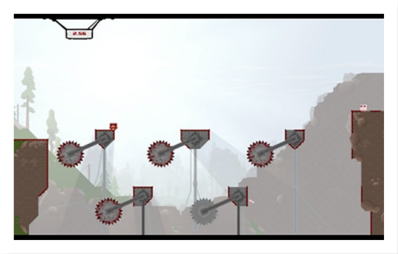


**Figure 1-3:** Super Meat Boy.

And these are just a fraction of the types of games that are available to play. Like films, there are many different genres, some broad (like action games) or specific (like puzzle-based fighting games — look up Super Puzzle Fighter).

Whether it's to feel accomplished after beating a challenging level, or to feel powerless as you wander through the woods while an unknown monster hunts you, you want a game to provoke some sort of emotion in you. Fun comes from a game provoking the right emotion at the right time.

If you think about it, this is the same reason you watch a movie or read a book. When you want to laugh, you watch a comedy. When you want to cry, you read a tragedy. Understanding that fun comes in many different forms and goes beyond a single emotion can open up a variety of game design that you may not have thought about before.

TIP

**When you're designing your game, ask yourself what type of emotion your game is trying to provoke — and capitalize on it! Sometimes that feeling is the excitement you feel in the heat of a battle. Other times, it's the sadness you feel after a game forces you to question your own mortality and life choices, like in one of my favorite games, To the Moon (see Figure 1-4).**

Figure 1-4: To the Moon.

# ASKING THE RIGHT QUESTIONS BEFORE YOU BEGIN

REMEMBER

**You don't have to answer these questions in this order. Often in game design, you start by answering any one of these questions and build out from there. There is no right or wrong order to answer these questions, but you should answer all of them before you start designing your game.**

Think of game design like making something good to eat. You wouldn't just throw any ingredient into a pan and hope for the best. To make it taste great, you need to start by asking yourself what kind of food you want to make. Are you making a salad? An appetizer? A soup or dessert? Once you know what the purpose of the food is — to be a satisfying entrée or a sweet treat at the end of a meal, for example — you can choose a recipe that will get you to your goal. And after you take it out of the oven or finish combining the ingredients, you might decorate it or add a garnish to make it look good.

Similarly, before you design a game, you need to ask yourself what type of game you want to make and who will be playing the game. In this section, I've listed some of the questions you should ask yourself.

## WHAT IS YOUR GAME ABOUT?

Do you want to make a war game that puts players in the middle of World War II or a game about a haunted children's restaurant where animatronic machines are trying to kill you? What your game is about can help you think about the story and help contextualize some of the decisions you'll make as you're designing your game.

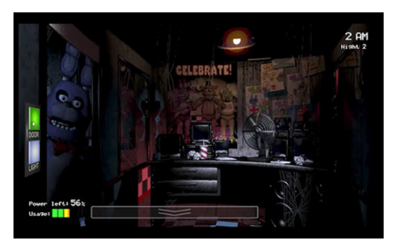


**Figure 1-5:** Five Nights at Freddy's.

## WHAT TYPE OF GAME ARE YOU MAKING?

Is your game a choice based role-playing game (RPG) or a more linear platformer? There are many different types of games that you can make. The type you choose can drastically change how people see your game.

Imagine if Skyrim were a 2D platformer, or if Five Nights at Freddy's (shown in Figure 1-5) gave you the ability to move around. Imagine how different those games would be.

**REMEMBER**

**Mechanics complement story. Whatever type of game you aim to make, keep in mind how different mechanics can change the audience for the game.**

## WHAT TYPE OF FEELING ARE YOU AIMING TO PROVOKE WITH YOUR GAME?

Do I want my player to feel powerful or powerless? A game that encourages exploration in a vast wasteland, like Fallout 4 (shown in Figure 1-6), provokes different feelings than a game that puts players in a cramped hallway. You want to go into your game knowing what type of feelings you want the player to feel. Knowing this can influence mechanic and story decisions.

**REMEMBER**

**No matter what, the player will feel *something* when she plays your game. And one way to ensure that she isn't bored when she's supposed to be excited is to make the game with the type of feeling in mind you aim to provoke.**

## WHO IS YOUR PLAYER PLAYING AS?

Is he a soldier in the war fighting for his country, or is he the civilian just trying to survive as the war happens around him? When you make your game, think of how you want to frame your story or your characters. The story of a brave knight trying to vanquish her nemesis, the evil warlock, is far different than a story of an evil knight bent on defeating the good wizard.

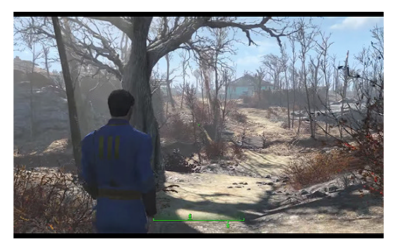


**Figure 1-6:** Fallout 4.

## WHAT IS THE PLAYER'S OBJECTIVE?

Games have goals. Sometimes the goal is to defeat the villain. Other times, it's to survive until morning. Think of your game as a story. Characters need a purpose. Mario isn't just running around the mushroom kingdom for fun. He's going to rescue the princess. Setting a clear objective for your player will give her direction in your game. Even games that focus on exploration set objectives that the player can strive for. Even if the objective isn't the most important part of your game, it's important to have one.

## WHAT OBSTACLES WILL YOUR PLAYER FACE?

What is stopping your player from reaching his goal? Games aren't fun without any challenge. You never want your player to be bored in your game. The easiest way to prevent boredom is to understand what types of obstacles the player needs to overcome. This can take the shape of enemy characters trying to kill your player or just puzzles that the player has to solve.

## WHO IS YOUR PLAYER?

The most important question of all is who you're designing your game for. The type of person who plays a peaceful game like Minecraft (shown in Figure 1-7) may not be the same type of person who plays a fast-paced fighting game like Street Fighter. Knowing the type of person you want to play your game can help determine the type of game you create.

**Figure 1-7:** Minecraft.

TIP

**If you're ever in doubt about who your player base is, ask yourself if you would want to play the game you're making. Don't design the game you think people want to play. Design the game *you* want to play.**

# CREATING YOUR GAME ON PAPER

A good exercise is to create a game on paper before you go into designing a game on the computer. This will help you understand the importance of mechanics in your game without your getting caught up with all the bells and whistles that come with a theme. Try thinking of a simple objective-based game that a person can play with just a pen and paper.

REMEMBER

**Paper games are tough because you need to get the player interested in the game without the fluff or theme. But in the end, if a game isn't fun, a theme — no matter how cool — won't fix it. Themes are important, but if you don't have an interesting game without the theme, no one will want to play it.**

# CHAPTER 02

## Unity: The Software You'll Use to Build Your Game

Unity is a program that you can download online for free — just go to www.unity3d.com/get-unity/download. Developed by Unity Technologies, this open-source program has opened up game development for a whole new generation of game developers (like you!). You can use Unity to develop 2D and 3D games, but for the purposes of this book, I show you how to develop a 3D game.

In this chapter, you'll learn some basic organization techniques that will help you manage your game creation. I show you how to create a new file in Unity, introduce you to the basic layouts and controls of Unity, and explain how to create an in-game object. This chapter may not be the most exciting, but you really need to understand these basics before beginning your game, so don't skip ahead!

Similar to the games you play (and will create!), game making is modular. You have to take things one step at a time. In later chapters, you'll dive into more difficult parts of game creation that have way more moving pieces and files to keep track of. Organizing your files and knowing the layout of Unity will save you the headache of trying to locate your files or tools, not to mention hours trying to retrace your steps.

## GETTING ORGANIZED

Whether you're baking cookies or changing the oil in your car or developing a computer game, you need to get organized first. Getting organized isn't the fun part of any job, but it makes every job easier.

Unity does a lot of heavy lifting when it comes to organizing the files you need to create your computer game. But before you begin creating your game, you need to create a directory to store all your games in. This directory serves two purposes:

- **It creates an easy-to-remember spot on your computer for you to find all your files.** The last thing you want to have to do is dig around through a bunch of folders looking for where you saved your game.
- **Creating backups and transferring your files is much easier when they're all in the same spot.** You don't want some files in one place and other files in another. All your files for your games should be in the same area. Think of it the way you think of the notebooks you keep for your classes. You wouldn't put your biology notes in your English notebook and that history quiz in your French folder — at least not if you want to pass those classes! The same goes for the files you use to build your computer games.

REMEMBER

**Computers are stupid — they can't find files if you move them. To save yourself a ton of time and frustration, store all your files in one location.**

**Every time you start to create a new game, make a simple directory to store your files in. Unity automatically creates simple directories for your files, but for the purposes of the game you'll be making in this book, you also need a directory that will include files outside the ones that Unity uses. To create a directory, follow these steps:**

1. **Create a folder called `Unity_Games` in the `My Documents` folder on your computer.**
2. **Inside the `Unity_Games` folder, create a folder called `Boxo_3D_Platformer`.**
3. **Inside the `Boxo_3D_Platformer` folder, create two folders called `Blender_Files` and `Unity_Files`.**

When you create games on your own, apart from this book, you can make these directories as specific or as broad as you want, to include things like music files, sound files, image files, and so on. For the purposes of the game you're building in this book, the directory described here (shown in Figure 2-1) will serve you just fine.

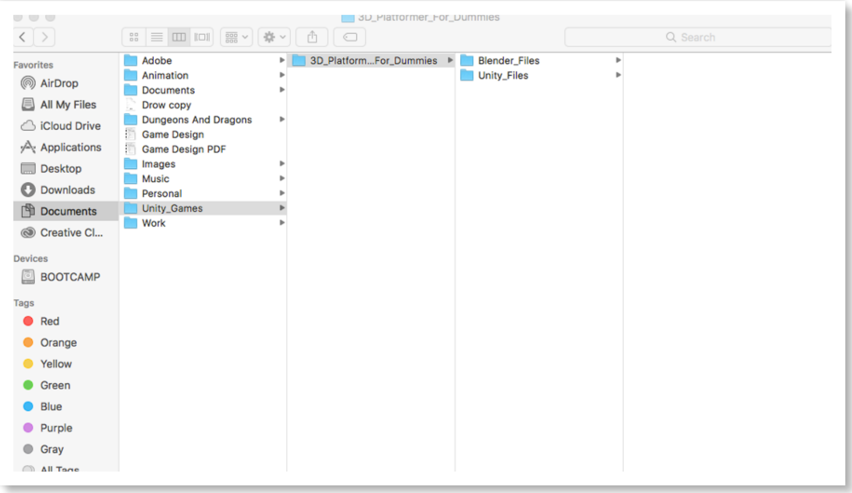


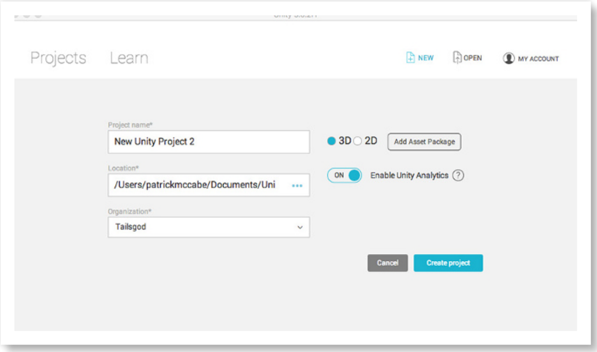
**Figure 2-1:** The directory for your new game.

## CREATING A NEW FILE

When you have your directory set up (see the preceding section), you're ready to create a new file. Follow these steps:

1. **Open Unity.**
2. **Click Create a New File.**

   The new project screen appears (see Figure 2-2).



Project name*: New Unity Project 2

● 3D ○ 2D

Location*: /Users/patrickmccabe/Documents/Uni

ON Enable Unity Analytics

Organization*: Tailsgod

Figure 2-2: The New Project screen.

3. **In the Project Name field, enter** Boxo_3D_Platformer.
4. **In the Location field, enter** Unity_Files.

   That's the folder you create in the preceding section.
5. **Select the 3D button.**
6. **Click Create Project.**

   A new folder is created within the `Unity_Files` folder that contains all your game information, as well as all your *assets* (components that are used within the game, game objects, characters, music, images, and other types of files used in our game are all examples of assets).

## UNDERSTANDING HOW UNITY IS LAID OUT

When you open Unity, you're greeted with a scary-looking screen (shown in Figure 2-3). *Do not be scared of Unity.* The best way to get over your natural fear is to understand what every part of the screen is and how it relates to the game you're creating.

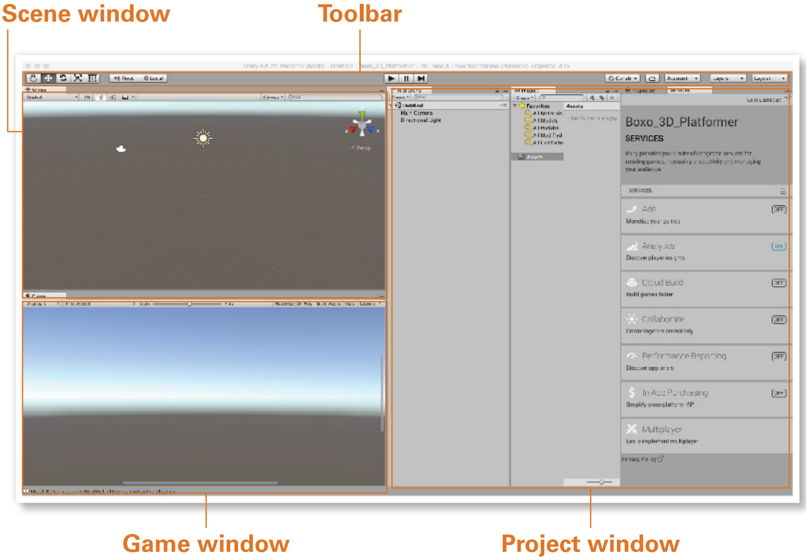


Figure 2-3: This is what you see when you open Unity.

Here's what you're looking at in Figure 2-3:

- **Scene window:** The Scene window, one of the two main windows in Unity, is where you place and set up your game objects.

TIP

**Game objects include everything that helps visualize your game. This includes objects that the player sees within the game, such as 3D objects and game text, as well as aspects of the scene that help bring the scene to life, such as lights and cameras.**

- **Game window:** The Game window, the other main window in Unity, gives you a sense of how your game will look like when you finish. It's a preview that lets you see what adjustments you need to make.
- **Project window:** The Project window allows you to manage the assets of your project. It organizes files by type. In the upper-right corner of the Project window, you can search for specific assets. Later on, the Project window will allow you to access your materials, scenes, prefabs, scripts, and other assets.

- **Toolbar:** The Toolbar (shown in greater detail in Figure 2-4) is located at the top of the screen. It gives you control over the scene, allowing you to transform the objects and move around the entire scene area. On the Toolbar, you find the controls for playing and pausing the game and accessing your account on the Unity cloud.

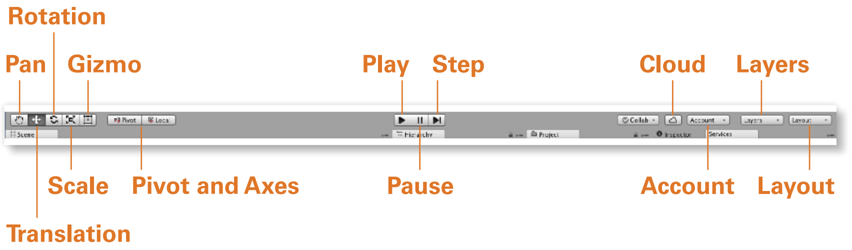


**Figure 2-4:** The Toolbar.

REMEMBER

**Your object will be rotated and moved a lot throughout this and other projects. Understanding the differences between local and global rotations will help you better control your scene. The *global axes* never change and remain in a fixed direction onscreen, no matter where or in what direction the object is. The *local axes* are attached to the object itself and change to match the angle the object is facing.**

Here are the tools you find on the Toolbar:

- **Pan tool:** Grabs onto the screen itself and moves the camera to have a better view of the game objects or world.
- **Translation tool:** Moves your objects along the three axes of 3D (X, Y, and Z).
- **Rotation tool:** Rotates objects around the three axes.
- **Scale tool:** Allows you to adjust the size of your object, making it bigger or smaller along the three axes. The square around the dot is also a scale tool, but it scales from the corners of the object, as opposed to just the center or pivot point.
- **Gizmo tool:** Adjusts where the *pivot point* (the point that your object rotates or scales from) is on the object, as well as what direction of the axes you follow.
- **Play button:** Starts the game, along with any physics or events that start when the game starts. Pressing the Play button again resets the game back to the start.