**INDEX**

|  |
| --- |
| CHAPTER 1- INTRODUCTION |
|  |
| 1.1 Introduction about Project |
| 1.2 Objectives |
| 1.3 Background of the Project |
| 1.4 Scope |
|  |
| CHAPTER 2- SYSTEM ANALYSIS |
|  |
| 2.1 Feasibility Study |
| 2.2 Fact Finding Technique |
| 2.3 Requirement |
| 2.4 Tools and Technologies |
| 2.5 Learning the Interface |
|  |
| CHAPTER 3- PROJECT DESIGN |
|  |
| 3.1 Use-Case Diagram |
| 3.2 Entity relationship diagram |
| 3.3 Activity Diagram |
| 3.4 Class Diagram |
| 3.5 Sequence Diagram |
| 3.6 Planning Scheduling |
| 3.7 Process Model |
| 3.8 Data Deployment Diagram |
| 3.9 Advantage |
| 3.10 Disadvantage |
| 3.11 Bibliography |
| 3.12 Conclusion |

**CHAPTER-1**

**INTRODUCTION**

**1.1 Introduction**

Developing software applications is a time-consuming process, and with time-consuming processes come high costs. During the last years, several software development methodologies, often known as *agile software* *development*, have become widely usedby software developers to address this issue. Many different development methodologies can be more or less good, depending of the task and application type.

One of the software development methodologies is the *evolutionary* *software method*, which, as the namehints, takes on an evolutionary approach to the problem, and allows the project to evolve through different stages of the project. Our case study will show how well this evolutionary approach worked on our project where we choose to develop a 3D graphic computer game. Some requirements for the computer game were given from the beginning, such as:

**2D graphics** –The game mustcontain 2D models, and render these in the game. 2D environments were never a requirement, and platform games with

2D environment could still open up for 2D objects.

**Impressive result –** The game resultmust impress whoever plays the game. It should last long, and make the players come back and play it over and over again.

**Graphical effects** –To achieve animpressive result, we would need to add modern graphical effects, such as real-time rendered soft shadows, motion blur, and ambient occlusion.

Working with these requirements, we decided to use Unity 3D as our platform to develop our 3D game with. This decision was made with regard to that the platform had many in-built tools and provided a good framework for us to get started with the development as fast as possible. The fact that Unity 3D also used javascript as development language was also in consideration, since we wanted to learn this newly developed javascript language.

The requirement for the game to contain 2D graphics introduced an interesting challenge for the project group, since all had none or little experience in 2D modelling. Spending time learning how to model proper 2D models for our game was therefore necessary. During the research to find out what 2D modelling program to use, we found that we could use different studios to create models that we could later import to our game project. The complete game contains models made in both Blender and 3D Studio MAX.

With these choices made, we soon had our development environment set to use Unity 3D supporting the framework, and Blender and 3D Studio MAX for modelling the graphical components. For some of the sound effects we also made use of Adobe Audition 2.0.

**1.2 Objectives**

The objective of this project is to develop a android application for 2D Platformer Game.

Level to scoring system like High score and with low score.

Moving path your Player when play with this Game. To safe your Player by hand’s on touching system and Defeat the Enemy.

Play this game with your list score number with high score and low score name.

**1.3 Background Studies**

To implement this Game project I have to study more about programming language.

This project is design and Development on the basis of certain programming language like C#.

**1.4 Scope**

This Report describes all the requirements for the project. The purpose of this research is to provide a virtual image for the combination of both structured and unstructured information of our project “Ellen’s World”. “Ellen's World” is a single-player strategy game on the Android platform. The player will progress through levels which require precise manipulation of the environment, though the game Encourages creativity and daring via branching pathways.

The episodic structure of the game facilitates the pace of the story. I am demonstrating the action flow between inputs, script, display (output). I am working mainly with story, levels, object, animation, graphics, scripts, game engine facilities. I am working with web launching, free hand programming, carton making.

**CHAPTER-2**

**SYSTEM ANALYSIS**

**2.1 Feasibility study exciting system**

A Feasibility study is undertaken to determine the possibility or portability of either improving the existing system or developing a completely new system.

Feasibility is the measure of how beneficial or practical the development of information system will be to an organization. The feasibility study involves following criteria.

* + Whether the identified user needs may be satisfied using current software and hardware technologies.
  + The study will decide if the proposed system will be cost-effective and if it can be developed given existing budgetary constraints.
  + The result should inform the decision of whether to go ahead with a more detailed analysis.

**There are three methods of feasibility study**

* + 1. Technical feasibility
    2. Economic feasibility
    3. Behavior feasibility

**Technical feasibility:**

It is measure of the practicality of specific technical solution and the availability of technical resources and expertise. Technical feasibility is computer oriented. The “P2D Platformer Game” is technical feasible because of the following reasons.

* + In Game hardware and software requirement are easily available.
  + The games have a good GUI interface.
  + The games will have user friendly form and screen.

**Economic feasibility: -**

It is a measure of the cost-effectiveness of a project or solution. This is often called a cost-benefit analysis. Economic feasibility deals with the cost and benefits of the information system.

In the economic feasibility, the development cost of the system is evaluated weighing it against the ultimate benefit derived from the new system. It is found that the benefit from the new system would be more than the cost and time involved in its development.

**The Game is Economical feasible because of the following reasons.**

Game requires less time to react for the user.

The cost of the hardware and software are normal.

GUI interface.

The system provides the services for the decision making. As this not begins a conversion of the present module into and rather begins creating a new module from scratch, the cost of the module includes cost of the module development; implementation and it not include the maintenance.

**Operational Feasibility:**

Operational feasibility covers two aspects. One is the technical performance aspect and other is the acceptance. Operational feasibility determines how the proposed system will fit in the current operations and what, if any job restructuring and retraining may be needed to implement the system.

In the system operational feasibility checks, whether the user who is going to use the system is able to work with the software’s with which the system is coded and also the mind of the user going to use the system. If the user does not understand or is able to work on the system further development is of waste.

**Behavioral feasibility:-**

Behavior feasibility determines how much effort will go in the proposed information system and in education and training efficiency on the new system.

It is a measure of how well the solution will work in the organization. It is also a measure of how people feel about the application.

**2.2 Fact Finding Technique**

The Fact-Finding method adopted are as follows:

**Interview:**

An interview with Mr. Arden Fernandez who is a Student at “Vasai College of Science and Technology” was conducted. The Questions related to the entire procedure of the current system was asked. Some of the questions about the job and reports for the month were asked.

Some of the related Questions are:

**What will be the genre of the game?**

In an interview with Mr. Arden Fernandez, I came to know that the 2D platformer Game is the game

That People Required in mobile devices. The Most Famous 2d platformer game is “Mario” .

**How will the game works?**

The works in 2D Environment. The game can be control by touch gestures on the mobile phone. The game doesn’t need active connection to work, It’s an offline game.

**What will the content of the game?**

It’s a story based 2d platformer game, In Which the player has to complete the story of game and to defeat the final boss to won.

**What is the unique features in your platformer game than other games?**

First thing that makes the game different from other games is that it can works without the active network connection other than pubg or other games which will required active connection all the time to play. The other thing that differentiate our game from other game is that it can works on almost all devices, Which means everyone can play awesome game with barely low end devices.

**What are the requirements of the game?**

In the planning phase, The recommended System Requirements will be an Android Phone with an Android version(5 or more) 512mb ram and 500mb storage. The actual requirements will know after the development of the game.

**2.3 System Requirement**

## 2. 2D Platformer Game: Unity3D Game

**Hardware:**

* 512 Mb Ram
* 1 GHZ Processor
* Android Device/Windows

**Software:**

* Android/Window

**2.4 Tools and Technologies**

**1. Android:**

Android is a mobile operating system (OS) based on the Linux kernel and currently developed by Google. With a user interface based on direct manipulation, Android is designed primarily for touchscreen mobile devices such as smartphones and tablet computers, with specialized user interfaces for televisions (Android TV), cars (Android Auto), and wrist watches (Android Wear). The OS uses touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, and a virtual keyboard. Despite being primarily designed for touchscreen input, it also has been used in game consoles, digital cameras, regular PCs (e.g. the HP Slate 21) and other electronics.

Android is popular with technology companies which require a ready-made, low-cost and customizable operating system for high-tech devices. Android's open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which add new features for advanced users or bring Android to devices which were officially, released running other operating systems.

Android, Inc. was founded in Palo Alto, California in October 2003 by Andy Rubin (co-founder of Danger), Rich Miner (co-founder of Wildfire Communications, Inc.), Nick Sears (once VP at T-Mobile), and Chris White (headed design and interface development at WebTV) to develop.

**Android SDK:** The Android SDK provides the API libraries and developer tools necessary to build, test, and debug apps for Android.

***Why Android?***

**A secure mobile experience:** A dedicated work profile, hardware-based encryption and sharing restrictions ensure business data - calendars, contacts, files and apps - are separate and safe from malware while personal information stays private.

**Easy to manage:** IT has full control of all work related policies, profiles and data - from distributing apps to wiping business information - and the standard Enterprise Mobility Management (EMM) framework delivers a consistent experience across all devices.

**All the apps you need:** Find and deploy business apps easily with Google Play and create apps quickly with the Android app framework. Seamlessly integrate with existing IT systems like Microsoft Exchange, IBM Notes and Google Apps for Work.

**A seamless experience:** Switch between work and personal activities intuitively and seamlessly. **Data separation with multiple profiles:** Work and personal information is securely separated and managed on a single device, with multiple profiles.

**Standard APIs for EMM providers:** A full range of management APIs will be available on all

Android devices, ensuring a consistent user experience.

**Secure app deployment:** All business apps are managed and distributed to the work profile through a managed version of Google Play.

**Flexibility:** Internally developed enterprise apps can be self-hosted on premise or hosted by Google.

**Full app management support:** APIs for complete app management via EMM solutions. Whitelist apps and control installs and updates for business users.

**Collaborate anyplace:** Connect wherever you are with enterprise apps for mail, contacts, calendar and tasks. Full support for Microsoft Exchange, IBM Notes and Google Apps for Work.

**Work anytime:** Create files and edit in real-time from your smartphone or tablet using Docs,

Sheets and Slides designed and optimized for mobile devices.

**Browse online with confidence:** Use the internet for work safely using the intuitive and secure Chrome browser, built on top of Android's security framework.

1. **Unity3D:**

Unity is an entire ecosystem of tools and services designed for people who want to build a successful business by creating multiplatform games and interactive content. The Unity ecosystem is available to anyone who downloads the Unity engine. The Unity engine integrates into one unparalleled platform the tools to create 2D and 3D interactive content; collaboration solutions; rapid multiplatform deployment, and retention, advertising and analytics services to grow your business.

Unity is a cross-platform game creation system developed by Unity Technologies, including a game engine and integrated development environment (IDE). It is used to develop video games for web sites, desktop platforms, consoles, and mobile devices. First announced only for Mac OS, at Apple’s Worldwide Developers Conference in 2005, it has since been extended to target more than fifteen platforms. It is now the default software development kit (SDK) for the Nintendo Wii U.

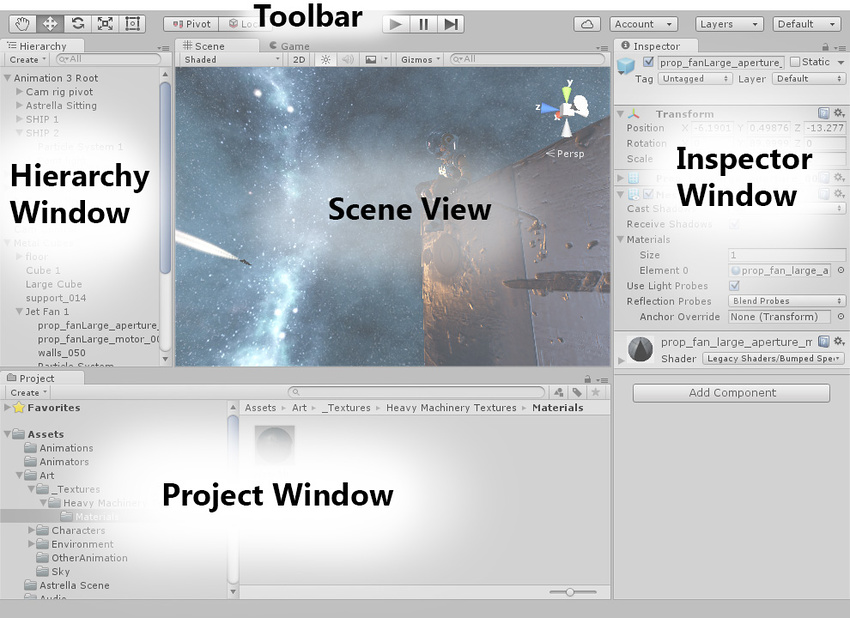
Unity is notable for its ability to target games to multiple platforms. Within a project, developers have control over delivery to mobile devices, web browsers, desktops, and consoles. Supported platforms include BlackBerry 10, Windows Phone 8, Windows, OS X, Linux (mainly Ubuntu)[citation needed], Android, 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's PhysX physics engine. Unity Web Player is a browser plugin that is supported in Windows and OS X only. Unity is the default software development kit (SDK) for Nintendo's Wii U video game console platform, with a free copy included by Nintendo with each Wii U developer license. Unity Technologies calls this third party provisioning of a free SDK, an "industry first".

# **2.5 Learning the interface**

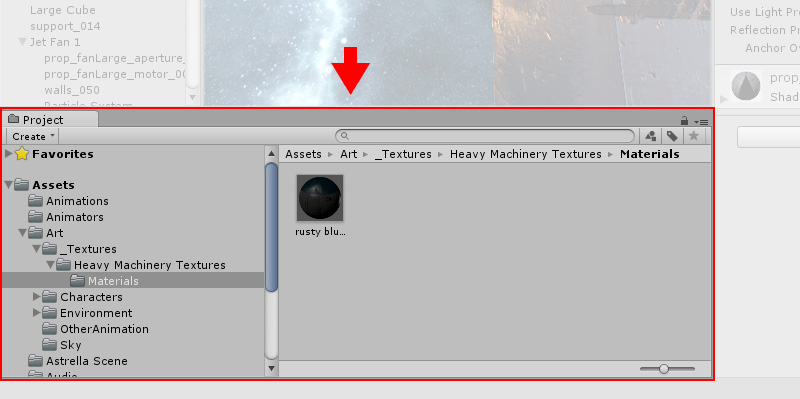
Take your time to look over the editor interface and familiarize yourself with it. The main editor window is made up of tabbed windows which can be rearranged, grouped, detached and docked.

This means the look of the editor can be different from one project to the next, and one developer to the next, depending on personal preference and what type of work you are doing.

The default arrangement of windows gives you practical access to the the most common windows. If you are not yet familiar with the different windows in Unity, you can identify them by the name in the tab. The most common and useful windows are shown in their default positions, below:

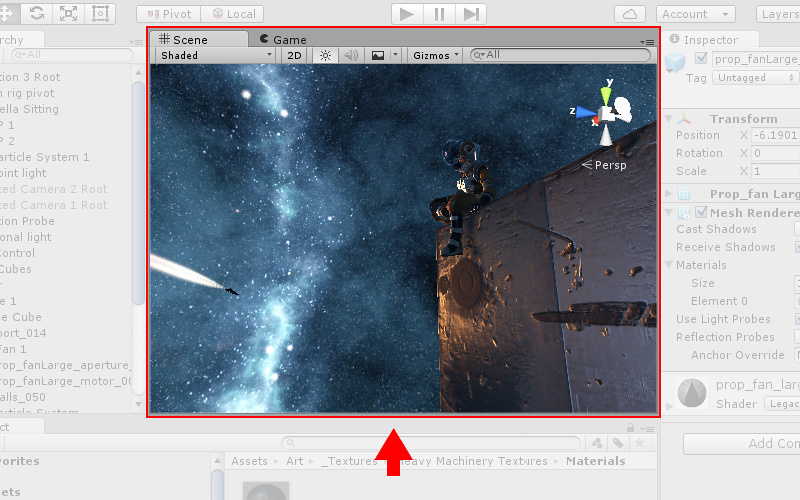


## **The Project Window**



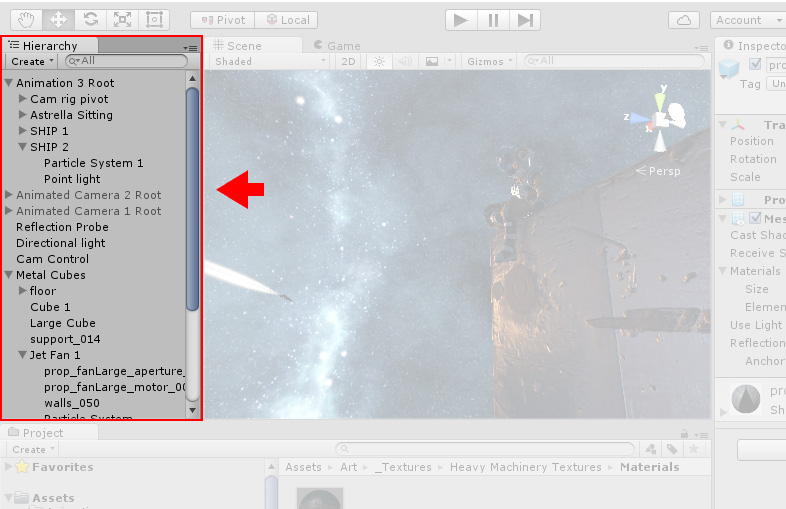
**The Project Window** displays your library of assets that are available to use in your project. When you import assets into your project, they appear here. Find out more about the [Project Window](https://docs.unity3d.com/Manual/ProjectView.html).

## **The Scene View**



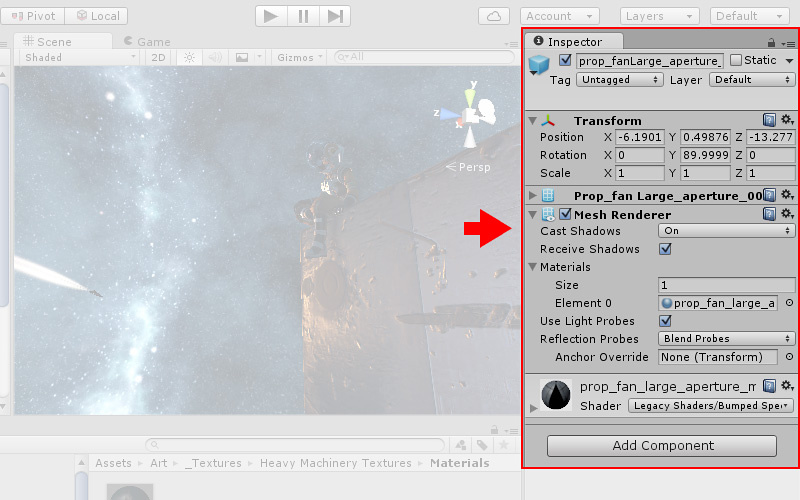
**The Scene**  
**View** allows you to visually navigate and edit your **scene**. The **scene** view can show a 3D or 2D perspective, depending on the type of project you are working on. Find out more about the [Scene View](https://docs.unity3d.com/Manual/UsingTheSceneView.html) and the [Game View](https://docs.unity3d.com/Manual/GameView.html).

## **The Hierarchy Window**



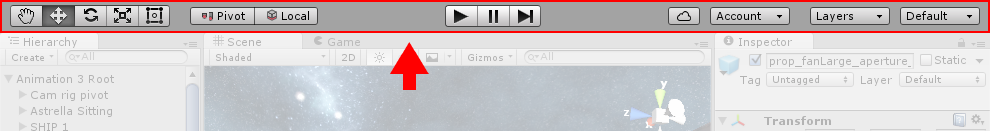
**The Hierarchy Window** is a hierarchical text representation of every object in the scene. Each item in the scene has an entry in the hierarchy, so the two windows are inherently linked. The hierarchy reveals the structure of how objects are attached to one another. Find out more about the [Hierarchy Window](https://docs.unity3d.com/Manual/Hierarchy.html).

## **Inspector Window**



**The Inspector**  
**Window** allows you to view and edit all the properties of the currently selected object. Because different types of objects have different sets of properties, the layout and contents of the **inspector** window will vary. Find out more about the [Inspector Window](https://docs.unity3d.com/Manual/UsingTheInspector.html).

**The Toolbar**



**The Toolbar** provides access to the most essential working features. On the left it contains the basic tools for manipulating the scene view and the objects within it. In the centre are the play, pause and step controls. The buttons to the right give you access to your Unity Cloud Services and your Unity Account, followed by a layer visibility menu, and finally the editor layout menu (which provides some alternate layouts for the editor windows, and allows you to save your own custom layouts).

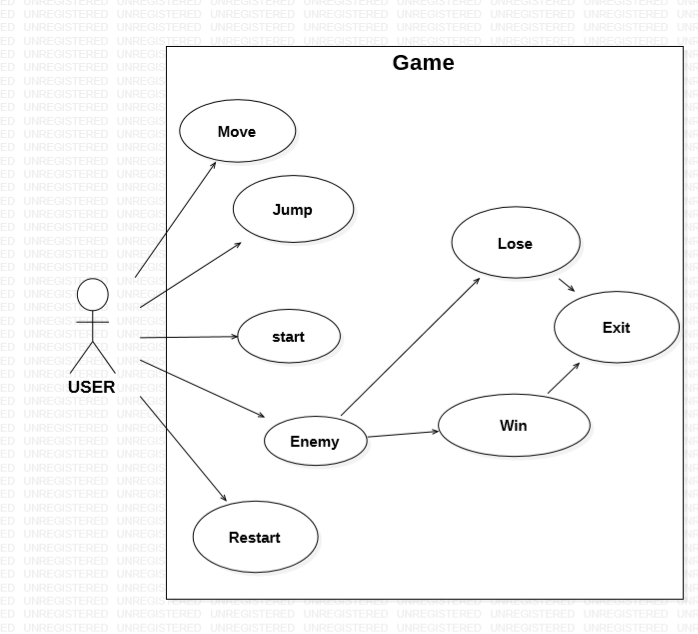
The toolbar is not a window, and is the only part of the Unity interface that you can’t rearrange.

**CHAPTER-3**

**PROJECT DESIGN**

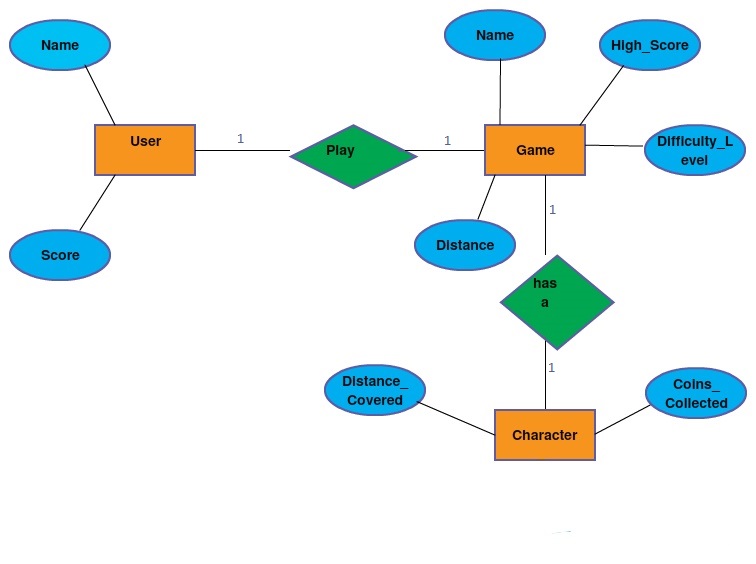
**3.1 Use Case Diagram:**

A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, Secondary Actor Receiver.



**3.2 Entity Relationship Diagram:**

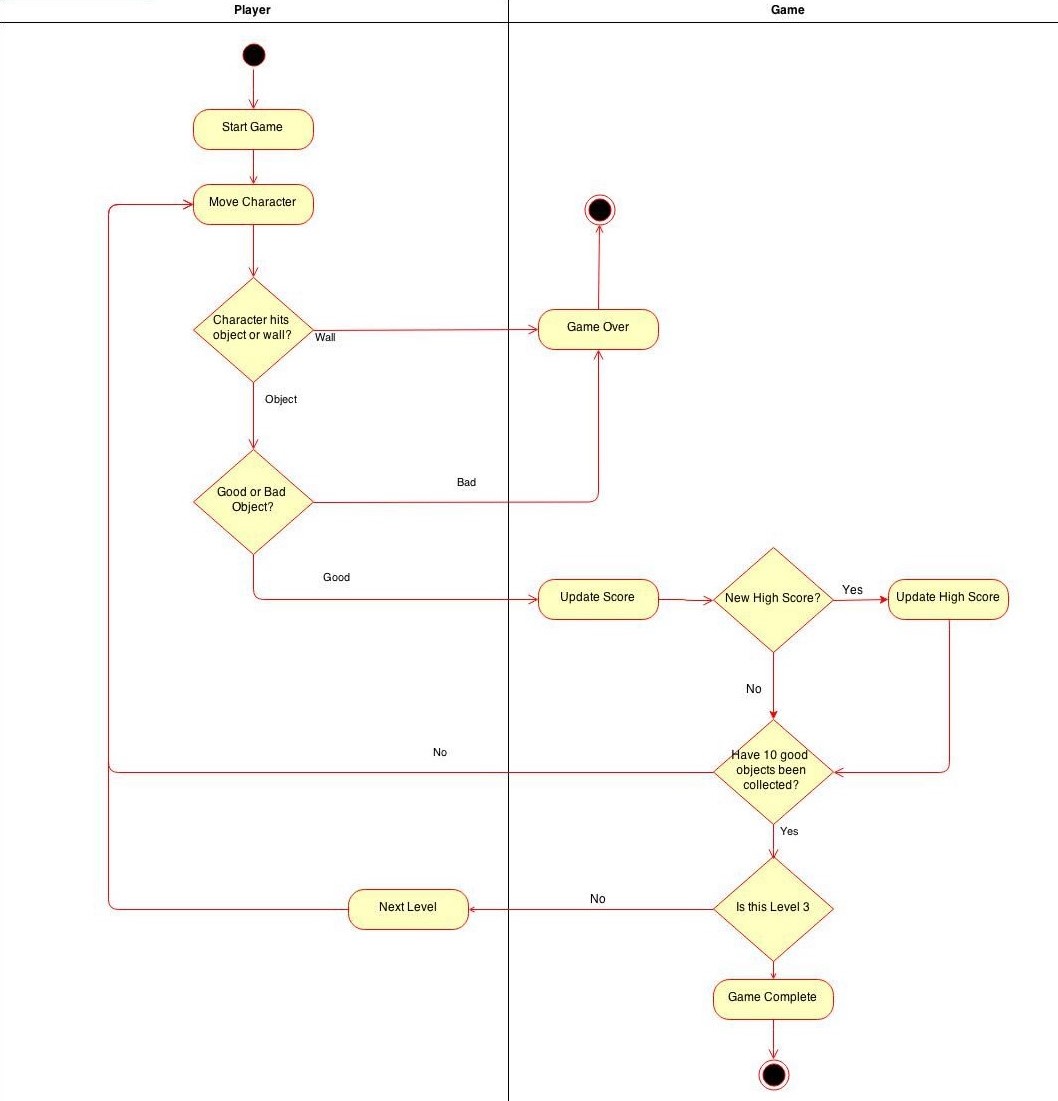
Entity Relationship Diagram, also known as ERD, ER Diagram or ER model, is a type of structural diagram for use in database design. An ERD contains different symbols and connectors that visualize two important information: The major entities within the system scope, and the inter-relationships among these entities.



**3.3 Activity Diagram:**

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

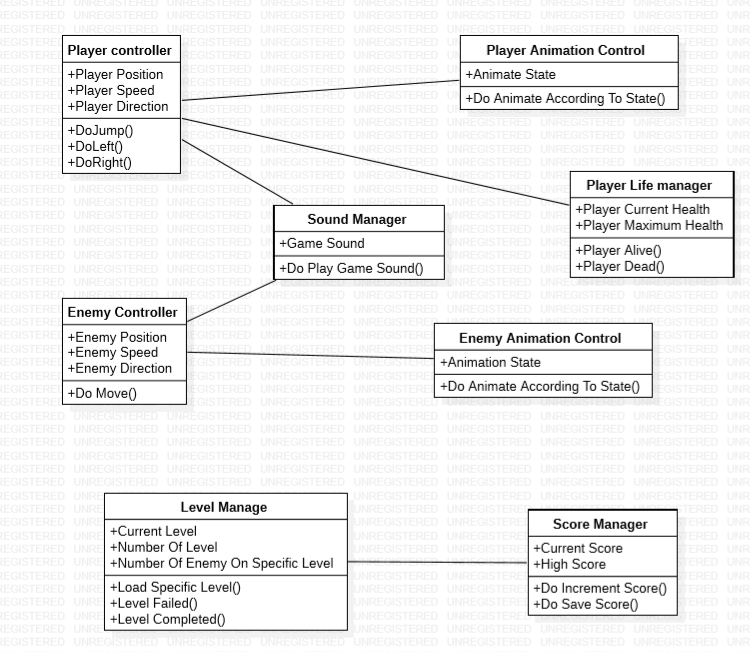
The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.



**3.4 Class Diagram:**

Class is nothing but a structure that contains both variables and methods. The Class Diagram shows a set of classes, interfaces, and collaborations and their relating ships. There is most common diagram in modeling the object-oriented systems and are used to give the static view of a system. It shows the dependency between the classes that can be used in our system.

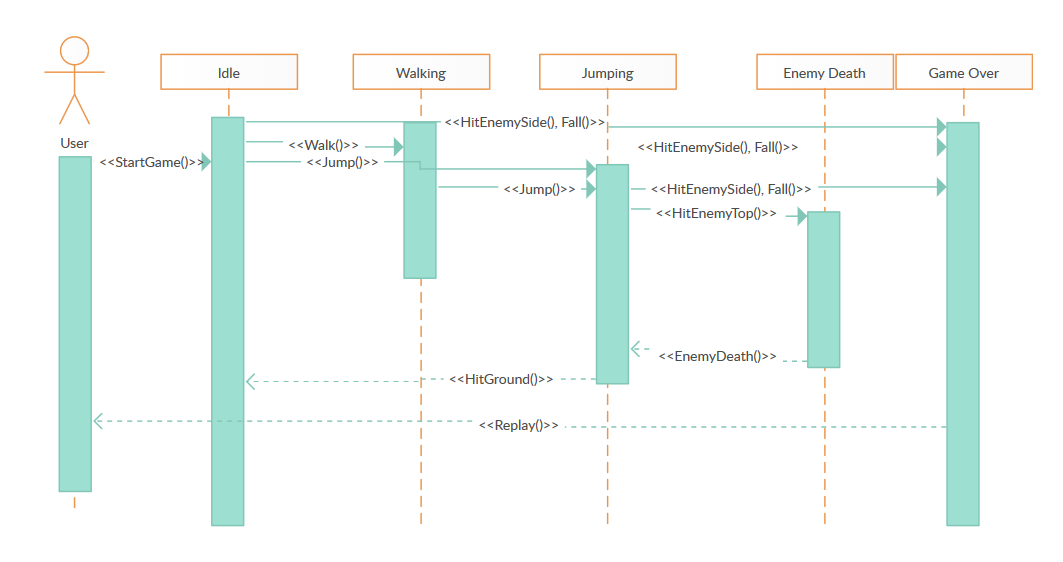
The interactions between the modules or classes of our projects are shown below. Each block contains Class Name, Variables and Methods.



**3.5 Sequence Diagram:**

Sequence diagram and collaboration diagram are called INTERACTION DIAGRAMS. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them

A sequence diagram is an introduction that empathizes the time ordering of messages. Graphically a sequence diagram is a table that shows objects arranged along the X-axis and messages ordered in increasing time along the Y-axis



**3.6 Planning Scheduling**

**3.7 Process Model**

The simplest process model is the waterfall model, which states the Phases are organized in the linear order. The model was originally proposed by Royce, though variations of the model have evolved depending on the nature of activities and the flow of control between them. In this model, a project begins with feasibility analysis.

**Steps: -**

* Project Planning Phase
* Analysis Phase
* Design Phase (architecture, system, detailed)
* Coding Phase
* Testing Phase
* Software manuals (e.g. – user, installation, etc.)

**Project Planning Phase**

**Design Phase**

**Analysis Phase**

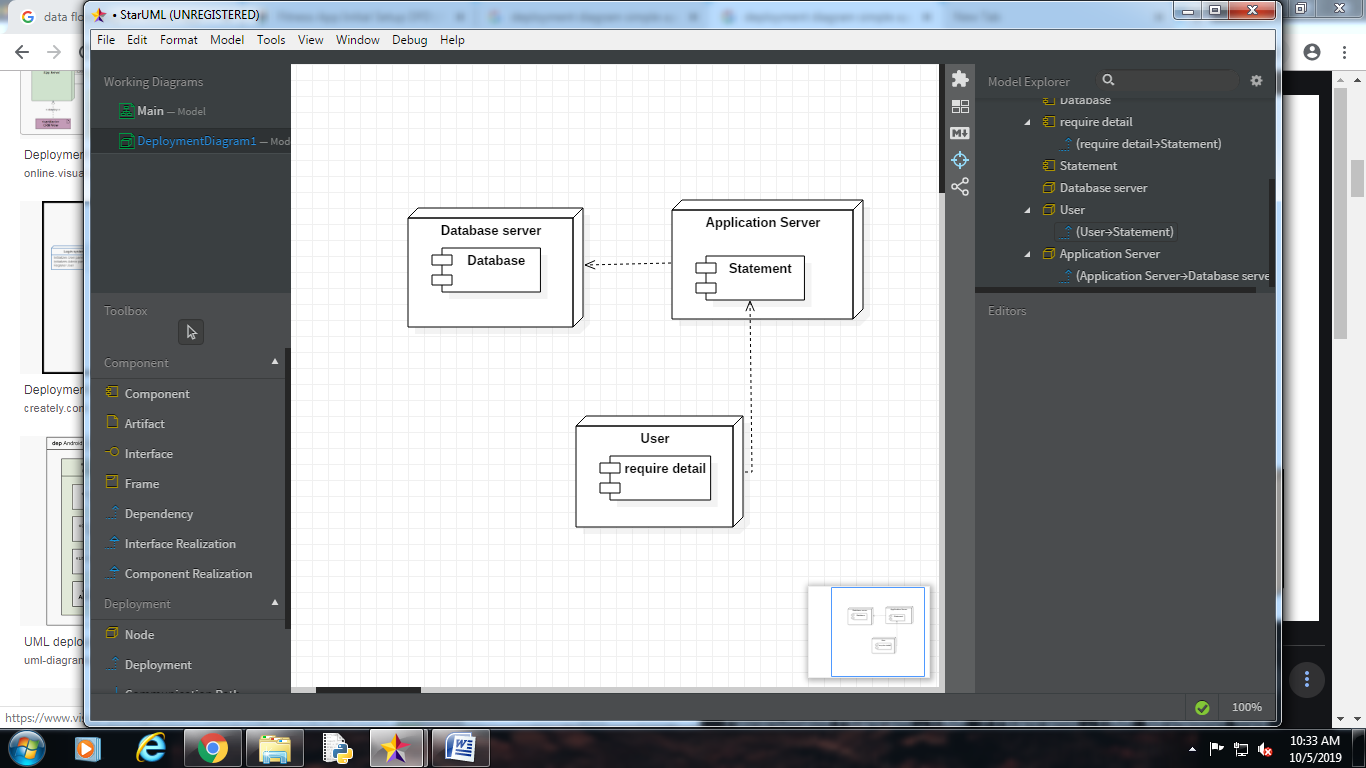
**Coding Phase**

**Testing Phase**

**Implementation Phase**

**3.8 Deployment Diagram**

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system



**3.9 ADVANTAGES**

• It is Free To Play.

• It has Great 2d Graphics.

• It gives immersive experience to the end user.

• It is Compatible with almost all Android phones.

**3.10 DISADVANTAGES**

• It is only Available for Android Phones.

• It has only 2d graphics.

• It supports Android ver 5.0 or higher.