Lifecycle for Game Development to Ensure Enhanced Productivity

K. Subhash Babu[1]   R.Maruthi[2]

Professor & Head, Dept. of MCA, SSN college of Engineering, Chennai, Tamilnadu, India[1]

Associate Professor, Dept. of MCA, SSN college of Engineering, Chennai, Tamilnadu, India[2]

ABSTRACT: Today the major player in the world of entertainment is games. Gaming as a hobby is no longer restricted to young children; it is equally enjoyed by older people. This can be attributed to their easy availability, not only on various traditional platforms but also on emerging mobile platforms. Games therefore have become an integral part of everyone's life. It is seen that playing games not only provides entertainment to the people it also has the potential to influences their behaviour. It can trigger new ideas and strategies and in the long run can also influence the personality of the game player. Like any other software development, the development of games also needs to be done in a structured manner. In this paper an overview of the game development lifecycle is discussed. It also reviews the designs of some of the game engines that are available and also shows how to code the games in a structured manner.

K*eyword*s: Games, Multimedia, Game Engines

I. INTRODUCTION

The gaming industry has seen unprecedented growth in recent times. In order to be ahead of the competition the game developer needs to bring innovative ideas to the table. Today's user is very demanding, hence there is a need for the designers to understand and keep pace with his/her demands. Game development is a creative activity that undergoes continuous evolution; hence it becomes difficult for the developer to stick to any specific development lifecycle. This paper addresses this need of the games developer by providing him with a games lifecycle that can adapt to the changing needs of the user.

The games development lifecycle consists of different phases which are similar to the phases in a typical software development process. These phases are influenced by many factors. The most important ones being the software and hardware requirements. In addition, the lifetime of the game also plays an important part in the lifecycle. The lifetime of the game is dependent on the version of the software that is available and the duration of its availability [1]. A given phase in the lifecycle for a specific game may or may not be applicable to another game. This paper suggests a way to design phases in the lifecycle which can be applied to almost all the games. Section 2 describes the phases in the games development followed by the conclusion and references.

II. GAME DEVELOPMENT PHASES

Many researchers have described the game development lifecycle in accordance with different software development models. All these models describe only the important phases in the lifecycle but no single model explains all the phases. Based on all these models it can be said that the three important phases in the game development process are the idea, proposal and design phases [2]. The phases in the game development life cycle are discussed in the following sections and the flow is shown in figure-1.

```
┌──────────────────┐                    ┌──────────────────┐
│  Developing the  │                    │ High, Module and │
│      Story       │                    │ Low Level Design │
└──────────────────┘                    └──────────────────┘
         │                                        │
         ▼                                        ▼
┌──────────────────┐                    ┌──────────────────┐
│Developing the Script│                 │     Coding       │
└──────────────────┘                    └──────────────────┘
         │                                        │
         ▼                                        ▼
┌──────────────────┐                    ┌──────────────────┐
│ Feasibility Study│                    │    Testing       │
└──────────────────┘                    └──────────────────┘
         │                                        │
         ▼                                        ▼
┌──────────────────┐                    ┌──────────────────┐
│ Promotional Demo │                    │   Debugging      │
└──────────────────┘                    └──────────────────┘
         │                                        │
         ▼                                        ▼
┌──────────────────┐                    ┌──────────────────┐
│    Designing     │                    │   Integration    │
└──────────────────┘                    └──────────────────┘
         │                                        │
         ▼                                        ▼
┌──────────────────┐                    ┌──────────────────┐
│ Layout Designing │                    │  Game Testing    │
└──────────────────┘                    └──────────────────┘
         │
         ▼
┌──────────────────┐
│    Modeling      │
└──────────────────┘
         │
         ▼
┌──────────────────┐
│    Texturing     │
└──────────────────┘
         │
         ▼
┌──────────────────┐
│    Animation     │
└──────────────────┘
```
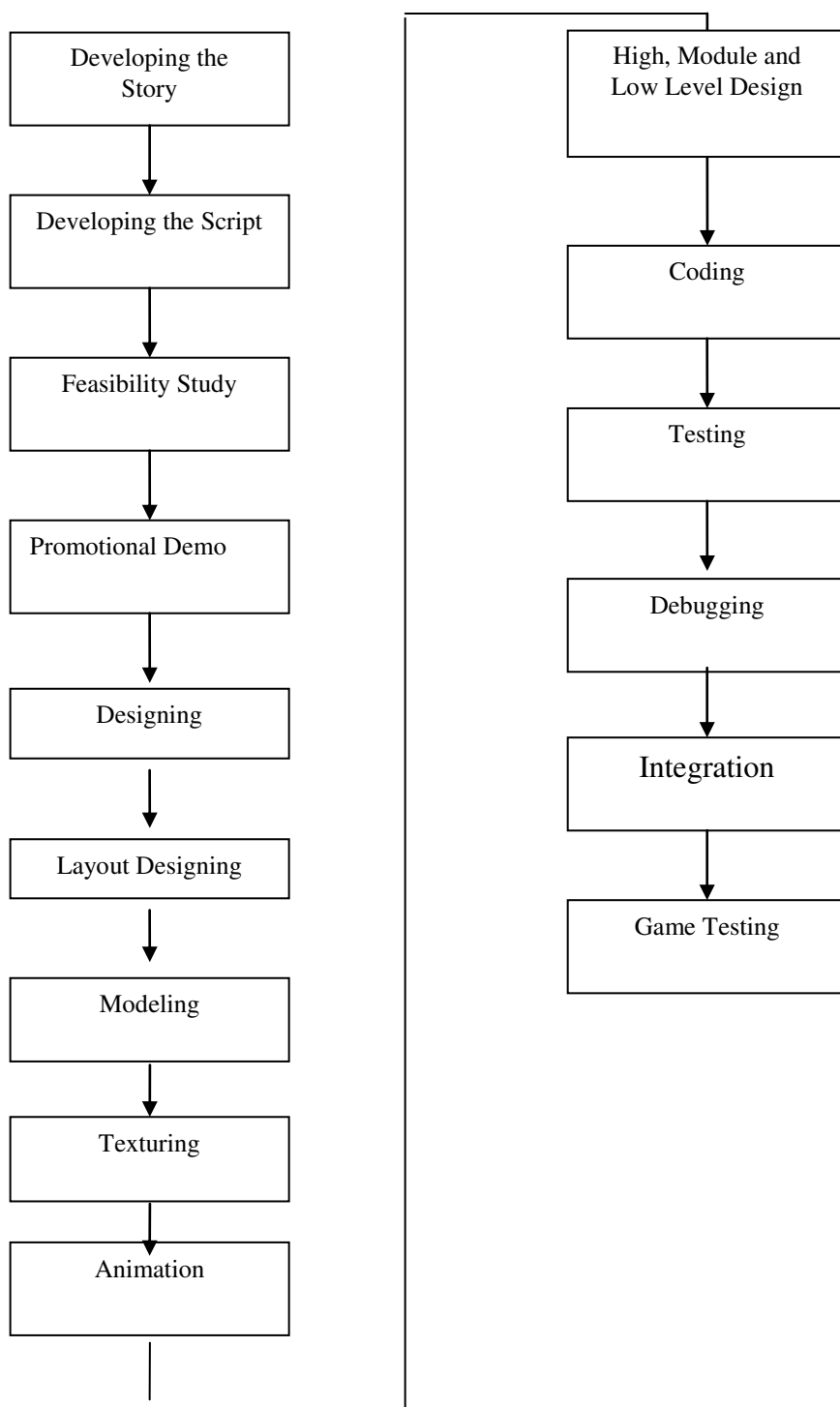
Figure-1: Various Phases in Games Development

A. DEVELOPING THE STORY

To start with, a core theme is developed with a well defined story line. The story is generally targeted at a specific age group. Every story and game starts with an idea or a concept. This idea is then developed further. Characters are decided, situations, events and places are created, relationships determined and so on. Following that, various conflict and resolution scenarios are added. All the stories follow a similar structure. Apparently the translation of a story into a game is mere conflict resolution that can be easily achieved based on the designer's creativity [3].

B. DEVELOPING THE SCRIPT

The story is then fine tuned to obtain a tight script that ensures a seamless flow of the game from one level to the other level. The better the script the better will be the game. The script writing needs a great deal of imagination and creativity. The key to write a good script is to capture the imagination of the audience. The characters and the backdrops are created and decided during this phase.

C. FEASIBIBLITY STUDY

This study is needed to assess the feasibility of a game. In addition it helps to freeze the requirements, the scope, profitability and other inferences of the proposed game [4]. The areas to be reviewed within the feasibility study are − overall analyses of the requirements, the pricing, technical, organizational, cultural and legal issues and the schedule of the project. This is a phase where project leaders, project design and development personnel and research group will be working together to create a game as a complete product [5]. The programmers analyze and explain the programming limitations to the project managers. The outcome of the feasibility study is the feasibility study report, which comprises of two parts - management summary and technical specification [4]. The findings and recommendations are tabulated in the report.

D. PROMOTIONAL DEMO

A plan for the promotional demo for the game is prepared and is executed. Promotional demo is needed to attract the potential customers. At this stage, since the game is not implemented the live demonstration is not possible. Hence a basic trial version is created for promotions.

E. DESIGNING

The three main designing elements are the character design, background design and sets & props design. Computer games utilize characters to a large extent. In Character Design, while designing a convincing character for games, emphasis is given to characters with strong personality and who can be of visual interest. This designing phase requires various techniques like use of visual design themes, silhouettes, poses, facial expressions and behavior of the characters [6,7]. Some of the character designs were depicted in figure-2
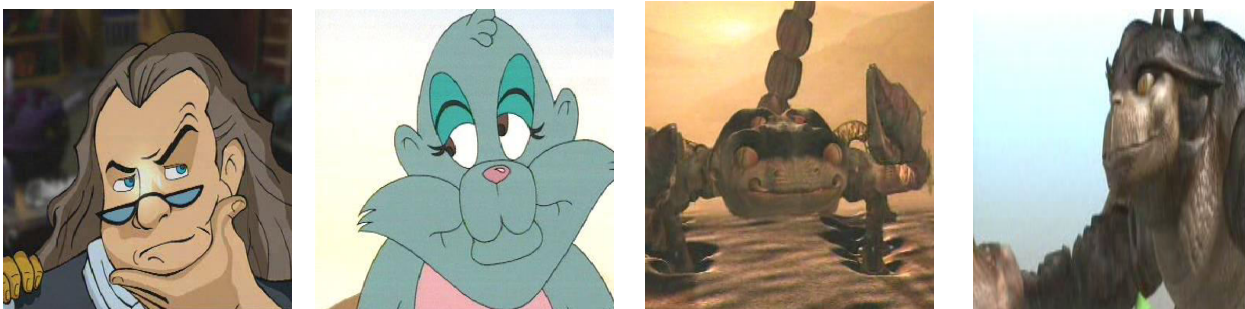

Figure-2: Character designs

In background design, it is enough to keep the background simple, if the emphasis is on the character and the characters are visually strong. Care has to be taken to ensure the background looks real. . A background design depicting the flow of water in a jungle is in figure-3(a) and figure-3(b) gives the background of the underwater scene.

Figure -3(a) Background design-1

Figure-3(b) Background Design-2

In sets and props design, the designers plan and define all the elements and components of a game like settings, layouts, constraints; story flow, characters, sets and props, and other paraphernalia available to the characters[8]. Some of the sets and props layouts are given in figure-4.

Figure-4 Sets and Props Designs

F. LAYOUT DESIGNING

In this stage the layout for the entire game and for all the levels is designed. The layout represents the various sets, the passage ways with hurdles and the escape routes.. Every type of game design map will have certain constraints. Covering all the constraints in the map is quite challenging [9]. The figure-5 shows the layout design of a game

Figure-5: Layout Design

G. MODELING

Modeling is of three types *character modeling, sets modeling* and *props modeling*. Models have to provide enough data so that the finer details can be rendered in an effective manner. There are many advanced techniques which are not very expensive, and they offer lots of features for creating models. One such advanced technique is the use of normal mapping[10]. Some of the props modeling were represented in figure-6.

Figure- 6 Props Modeling

H. TEXTURING

Texturing is a technique for adding detail, providing surface texture or color so that a realistic look can be given to the characters, sets and props. It is very common in almost all the 3D applications. An example of textured design is given in figure-3.

Figure-7: Textured designs

I. ANIMATION

After designing, modeling and texturing stages, the characters have to be animated to represent the sequence of actions in the game. Because the flow of the game is not linear, mesh methodology is used to animate a character. Mesh methodology supports the deformations of Muscles & Joints. (eg., facials, biceps, skin movement on rib-joints for animals, etc).

The animation requires some criteria for the selection of methods like 2D or 3D, and also requires expert tips on key framing and production management. The figure-8(a) and (b) shows the sequence of intermediate frames in frame by frame animation.

Figure-8(a) Frame by Frame Animation-1

Figure-8(b) Frame by Frame by Animation-2

The animation phase includes motion capture (mocap) animation and key frame animation. In mocap animation each and every movement of the characters or objects enacted by humans or trained animals with either optical or magnetic sensors placed on them , is captured and recorded to create a series of scenes. These movements are later applied to the models. The only disadvantage is that it requires use of expensive camera/magnetic systems and very specific hardware and software. Key frames are snapshots of an image at a single point in time. Key frame animation is the cycling of key frames to give the illusion of movement.

J. HIGH AND LOW LEVEL DESIGN

The high level design gives the overall design of the game by identifying all the elements. It shows the abstract view of the system and it hides some of the details. The design also gives the relationships between the various modules involved in the development process in the form of data flow, flow charts, data structures etc., In module design high level design is segregated into various modules. The low level design details the high level design (i.e) the logic behind each of the module is defined .

K. CODING

Coding is an important part of game development process. This phase includes not only coding for making the gaming interactive but also for implementing the interface design. Various game engines are available in the market to code a game and execute the code and edit the game. There are a separate sets of tools for making 2D games and 3D games. The 2D games tools include, GameMaker, Multimedia Fusion 2, Construct, Flixel, FlashPunk and Stencyl. 3D games tools are Unity , UDK , XNA , BlitzMax , jMonkeyEngine and   Torque to name a few[11]. Most games today on the PC platform are written using C++.

GameMaker

GameMaker for Windows is the ideal economical tool for use with Microsoft Windows for fast and simple games development. . Its features include drag and drop interface, providing background designs, sprites (process of keeping multiple images as a single image) and animated graphics. The image editor comes with 3D graphics, advanced text drawing, sound effects music etc[12]. It is a fully integrated development environment(figure:9) with a built in scripting language called GameMarker Language (GML
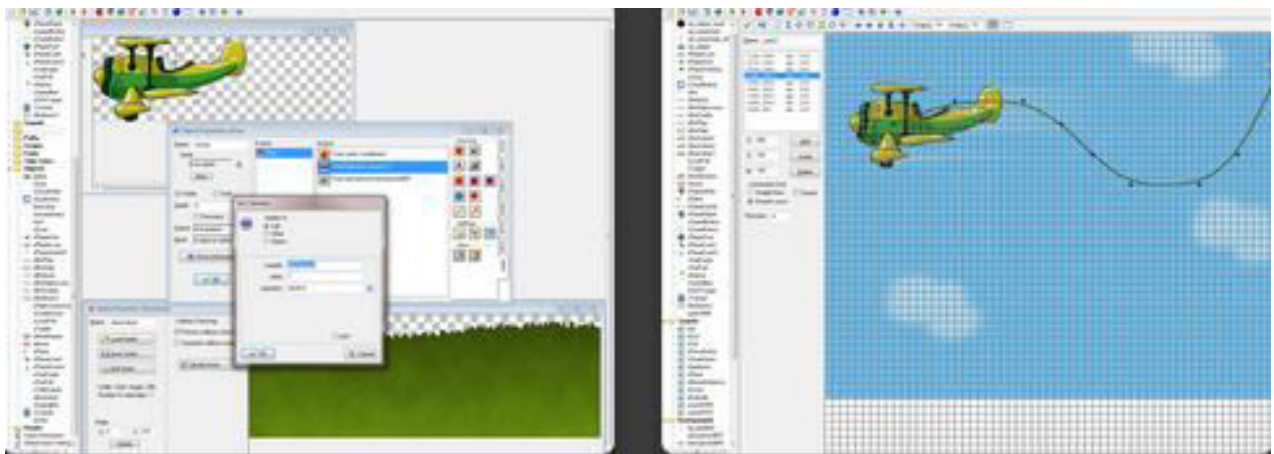
Figure:9 GameMaker IDE

Here is an example of GML script. The script below is written to display a text in the specified co-ordinate position if 'a=1', otherwise to draw a rectangle.

```
if a=1 {
        draw_text(100,100,"A = 1");
            //draw "A = 1" at      coordinate(100,100)
   }
else
  {
      draw_rectangle(x1,x2,y1,y2,colour)
    }
```

Multimedia Fusion 2(MMF2)

Multimedia Fusion 2 is a gaming tool by Click Team. It provides a more user friendly environment for building games with a creative flexibility. The games can be published in multiple platforms like iOS, Android, Adobe flash Player, Java etc,. The MMF2 is built upon the visual programming interface with a frame editor, data editor, event editor, Picture and Animation Editor, Event List Editor etc,.[13]. It supports most of the image file formats and sound file formats like Wav, mod and OGG Vorbis

Flixel

Flixel is an open source game-making library. It is totally free for personal or commercial use and it is written entirely in Actionscript 3. Like other game engines and libraries, it supports simultaneous motion of objects, collisions between objects, grouping of objects and generation & emission of particles. It allows the developer to create game levels using tilemaps, enable scrolling and usage of Mouse & keyboard input and so on.[14]

International Journal of **I**nnovative **R**esearch in **C**omputer and **C**ommunication **E**ngineering

(An ISO 3297: 2007 Certified Organization)

Vol. 1, Issue 8, October 2013

A simple code segment to display HelloWorld in Flixel is given below
Helloworld.as

```
package
{
import org.flixel.*;
[SWF(width="640", height="480",

backgroundColor="#000000")]
public class HelloWorld extends FlxGame
{
public function HelloWorld()
{
    super(320,240,PlayState,2);   }
            }
}
```
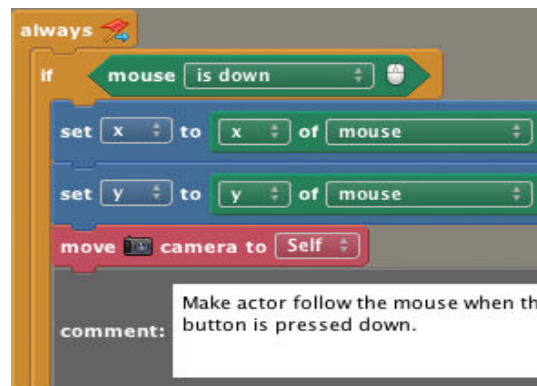
PlayState.as

```
package
{
import org.flixel.*;
public class PlayState extends FlxState
{
override public function create():void
{
add(new FlxText(0,0,100,"Hello, World!"));
} } }
```

Stencyl

Stencyl is an attractive, intuitive toolset for game developers. Once it is designed, it can run anywhere without code and can be published in platforms like iOS, Android, Flash, HTML5, Windows and Mac[15]. The Design interface and the coding interface in Stencyl software is given in figure: 10(a) and (b)

Figure:10(b) Design interface   **(b)Stencyl- Coding Interface**

Unity – Code Hero

"Code Hero" developed by Primer labs, San Francisco is a tool which is used to code games in JavaScript with the Unity 3D game engine. A Simple code snippet to enter into the game world, send a message and play the audio is given below

```
var world : World;
function Start() {
world=GameObject.Find("World").GetComponent
                              ("World");
 }
function OnTriggerEnter( other : Collider ) {
   world.SendMessage("AddCoin");
   renderer.enabled = false;
   collider.enabled = false;
   if ( audio ) {
          audio.Play();    }
}
```

UDK- Unreal Development kit

UDK - Unreal Engine 3 – a complete professional development framework. UDK is used in all aspects of the development process. The process includes planning of production and programming, creation of content and various levels of games. It also helps in profiling, optimizing, Packaging and Distribution of games[16]. It's all round utility makes it an awesome games development tool on diverse platforms such as PC and Mac.

Blitz Max

BlitzMax is the new, next generation game programming language from Blitz Research. It can be used in all versions of Operating Systems like Windows, Linux and Mac OS. It comes with a BlitzMax compiler, core BlitzMax, Max2D graphics, MaxGUI, Integrated Development Environment, debugger, sample code and documentation and Source code for all modules.

 It supports function pointers, dynamic arrays, inheritance and polymorphism etc,. It has a flexible 'collection' system for dealing with linked lists, low level pointer handling and Unicode Transformation Format 16 (UTF16) strings,. It also has the ability for interfacing with C/C++/Objective C or assembly code [17]. The BlitzMax IDE is given in figure:-11
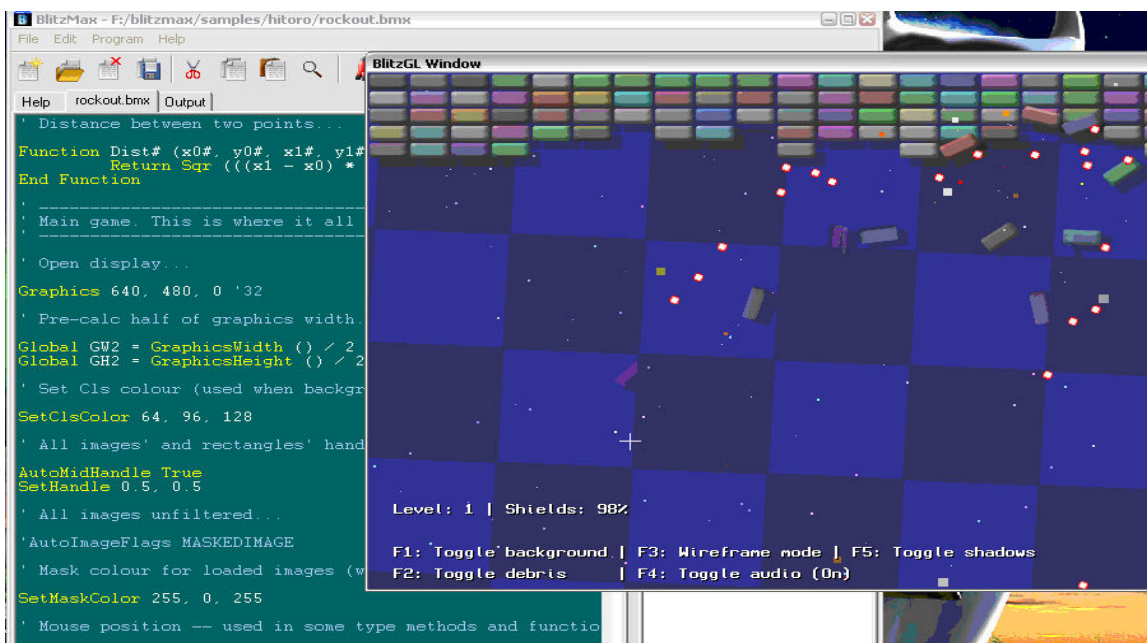

Figure:11 BlitzMax IDE

The typical code to draw a triangle in OpenGL is as follows.
GLGraphics 640,480

While Not KeyHit(KEY_ESCAPE)
        glClear GL_COLOR_BUFFER_BIT

        glBegin GL_TRIANGLES
        glVertex2i 0,1
        glVertex2i 1,-1
        glVertex2i -1,-1
        glEnd
        Flip
Wend.

jMonkey Engine 3.0

        It is a JAVA OpenGL game engine to make any type of game for all modern platforms. jMonkeyEngine is a collection of libraries that is bundled in SDK. It is a software system designed for the creation and development of video games. It supports shader-based architecture which has an excellent compliance with next generation graphics standards and its provides OpenGL Shading Language (GLSL), shader libraries etc,. It is bundled with functions for lighting, binding, special effects, asset importing and texturing [18]. The GUI features allows creating an interface, which   enables user to navigate and interact with the games in a flexible manner. The jME3 development environment is given in figure:12
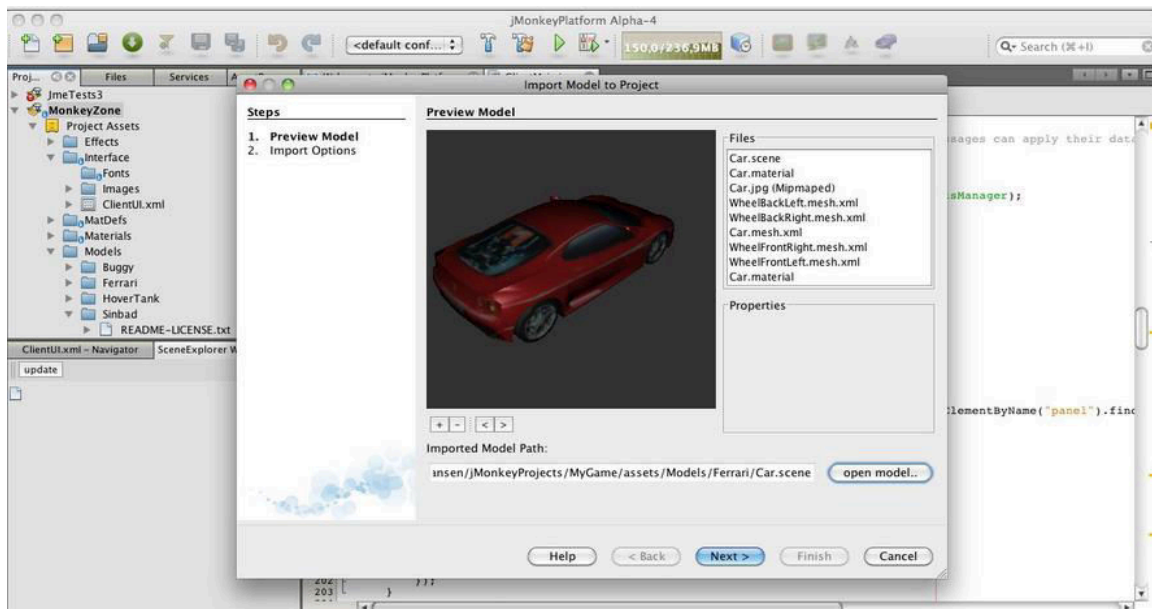

Figure:12 jME3 development Environment

Adventure Game Studio

        The AGS Editor is a **Windows-based IDE** for creating games as quickly and easily as possible. The IDE allows to import graphics, write game scripts and also test the game[21]. The AGS IDE for the design editor and script editor is given in the figure:13.
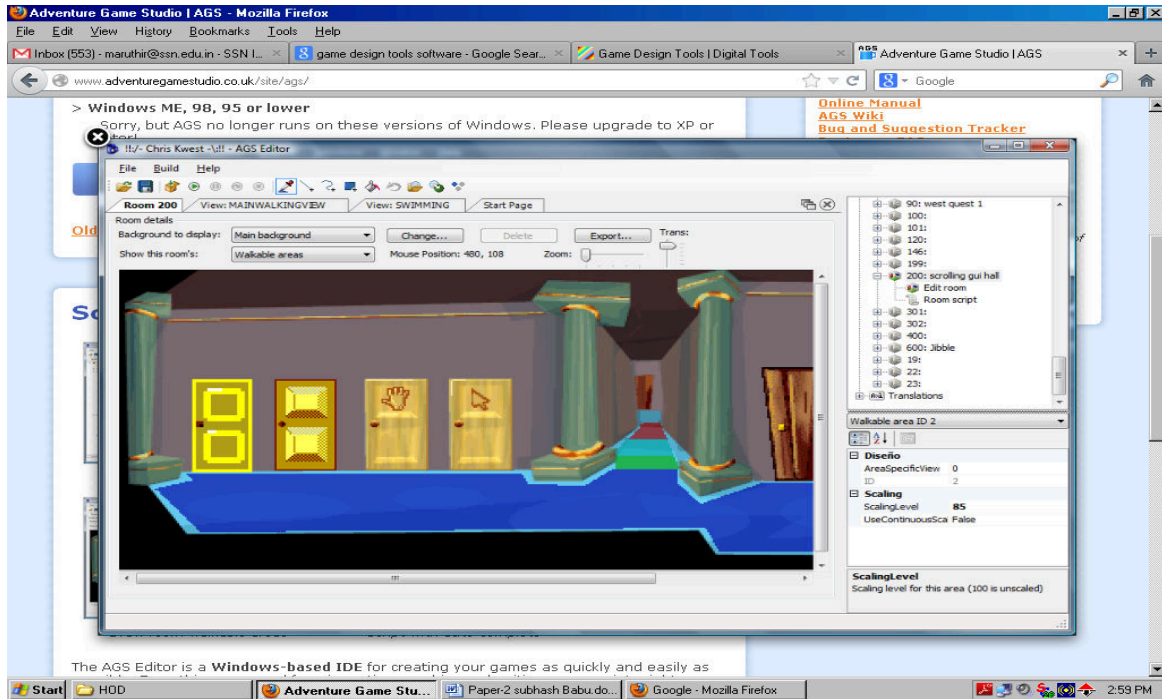
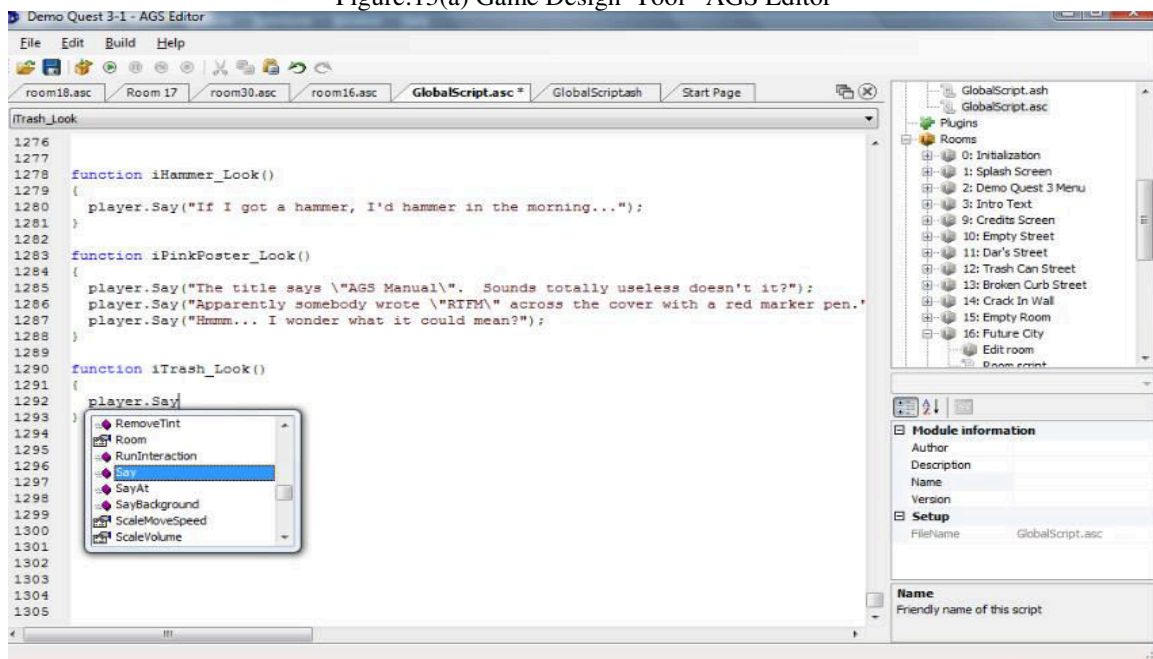Figure:13(a) Game Design Tool –AGS Editor

Figure:13(b) Game Design Tool –AGS Script Editor

Stratagus

It is a free open source software GUI IDE for creating games which is licensed under GNU GPL. It includes support for playing over the internet/LAN, or playing a computer opponent. A port of Stratagus real-time strategy gaming engine for Android can be used to play several games, including: Wargus - Warcraft II: Tides of Darkness & Beyond the Dark Portal (expansion pack) clone[22,23].

The whole environment is mainly written in C or C++ with the configuration language being Lua. SDL(Simple DirectMedia Layer) , gzip and bzip2.. Lua has a relatively simple C Application Programming Interface (API) compared to other scripting.

A code fragment in Lua to enter the game world or to quit the game is given below

```
class RPG
 {
        Lua lua = new Lua();
         bool quit = false;
       [STAThread]
 static void Main(string[] args)
 {
        RPG rpg = new RPG();
        rpg.Go();
 }
 public void Go()
 {
      Console.WriteLine("Welcome to the cool RPG Game");
      Console.WriteLine("\tPress q to quit!");

 while(quit == false)
 {
       Console.Write(">");
       string ans = Console.ReadLine();

  if(ans == "q")
       quit = true;
    }
     Console.WriteLine("Goodbye");
    }
 }
```

It successfully runs under all most all Operating systems like Windows, Linux, Mac OS and BSD[22,23] .

L. **TESTING AND DEBUGGING**

This phase is carried out at the module level before integrating all the modules. Testing is to be done to make each and every module in the game to be free from errors and bugs. Most of the Games designing tools come with built in testing and debugging tools.

M. INTEGRATION

After testing, all the modules are integrated to get the final product. It is very similar to assembling the various parts of a product to get a complete product.

N. GAME TESTING

Game Testing is a subset of the game development process. The game is tested for the complete flow from the beginning to the end of the game rigorously to identify the bugs and correct. Since it is a complex task which involves twists and turns, it is tested thoroughly to meet the expectations of the users. Some of the software's for testing are as follows

Game Tester 1.1

Game Tester is an open source free software that allows the user to create and test games. This software is designed to run on all the versions of the Windows Operating system platform.

SeeTest

SeeTest provides a complete solution for test automation of games. It supports image, animation, non standard elements etc., and full support for HTML, Silverlight, Flash, Flex, and HTML5. It is a simple record and play type and no programming is required. It is supported in almost all browsers like Internet Explorer, FireFox, Chrome, Safari, Opera and Plugs into Quick Test Professional (QTP), TestComplete, Rational Function Tester (RFT), C#, Java, Python, and Perl[19]

eggPlant

eggPlant is a graphical user interface (GUI) test tool that enables data-driven testing, load testing and regression testing at the same time. It is a perfect tool for the media industry because it can automate application and GUI testing across various browsers, mobile devices and platforms by means of the same scripts. The testing cost and quality assurance are reduced to a great extent. Producing and supplying products to the market is made very fast using this tool [20]. The eggPlant interface for game testing is given in the figure:14
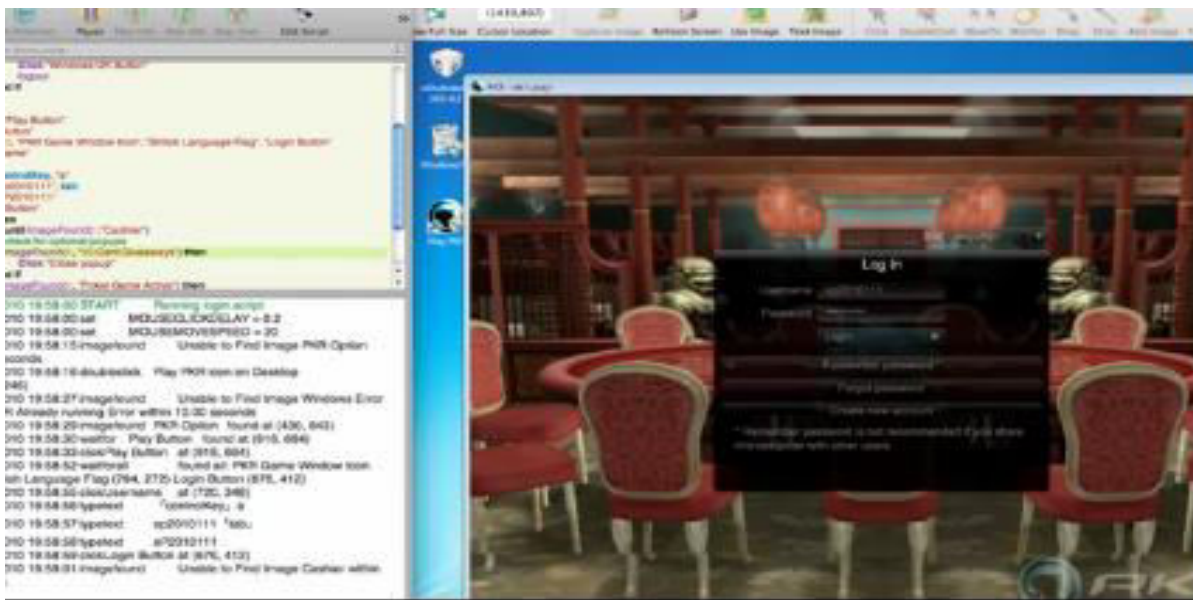

Figure:14 eggPlant interface for game testing

III. CONCLUSION

This paper gives an outline of the phases involved in the game development process and briefly discusses some of the tools for designing and coding games. The suggested framework and the tools make the development process systematic and efficient. The game designers and developers can be benefited by this and as they can proceed their game development process without worrying about any defects and shortcomings.

IV . REFERENCES

[1]	http://game.ezinemark.com/the-life-cycle-of-a-game-7d2d85a67203.html

[1]	http://www.docstoc.com/docs/39557145/The-Game-Development Process_ PowerPoint_---The-Game-Development.

[2]	http://www.rpgrevolution.com/tutorial/developing-a-game-storygame writing_3.html

[3]	http://www.ogcio.gov.hk/en/infrastructure/methodology/software_life_cycle/

[4]	http://www.techbaba.com/q/196feasibility+study+software+development+life+cycle.aspx

[5]	http://www.cgmwonline.com/character-design-for-games.html

[6]   Petri Lankoski, Character-Driven Game Design: Characters, Conflict, and Game play more

[7]   http://www.creativeskillset.org/games/careers/article_4724_1.asp

[8]   http://www.raphkoster.com/2009/03/10/some-zone-design-lessons/

[9]   James Dargie, Modeling Techniques: Movies vs. Games

[10]   http://www.pixelprospector.com/the-big-list-of-game-making-tools/

[11]   http://www.yoyogames.com/gamemaker/windows

[12]   http://www.clickteam.com/website/world/multimedia-fusion-2

[13]   ww.flixel.org

[14]   http://www.stencyl.com/

[15]   http://www.unrealengine.com/udk/

[16]   http://www.blitzbasic.com/Products/blitzmax.php

[17]   http://jmonkeyengine.com/engine/

[18]   http://experitest.com/web-test-automation-seetest/gaming-test-automation/

[19]   http://www.testplant.com/solutions/media-and-gaming/

[20]   http://www.adventuregamestudio.co.uk/site/ags/

[21]   http://stratagus.com/

[22]   https://play.google.com/store/apps/details?id=com.drodin.stratagus&hl=en

BIOGRAPHY

K. Subhash Babu
Professor & Head in the Department of Computer Applications, SSN College of Engineering, Kalavakkam, Chennai, He   has 35 years of industrial (28 years) and teaching (9 years) experience, received his B.E (ECE) degree first class with distinction from Andhra University, M. Tech. Electrical Engineering from Indian Institute of Technology. His research interests includes multimedia, video compression etc,.

R.Maruthi
received her **M.C.A** degree from Shanmugha College of Engineering.Bharathidasan University, Trichy, in 2000 and **Ph.D** degree from Mother Teresa Women's University, KodaiKanal., in 2010 and currently   working as an Associate   Professor in the Department of Computer Applications in SSN College of Engineering, Kalavakkam, Chennai. Her research interests includes Image Processing, Computer Graphics, Multimedia and Web Graphics