Project Proposal on



**Crazy Shooter (2D Game)**

**Submitted by:**

Rabin Khanal

00175079

Computing Project

Level 5 Diploma in Computing

Softwarica College of IT & E-Commerce

Batch 22C

Kathmandu, Nepal

April 8, 2019

**Submitted to:**

Sudip Lal Bajimaya

Module leader

Contents

[**Chapter 1** 3](#_Toc5691720)

[**Introduction** 3](#_Toc5691721)

[**1.1** **Introduction of my Project.** 3](#_Toc5691722)

[**1.2 Background of the Project** 3](#_Toc5691723)

[**1.3 Problem Statement** 3](#_Toc5691724)

[**1.4 Description of the Project** 3](#_Toc5691725)

[**1.5 Overview of the Project** 4](#_Toc5691726)

[**Chapter 2** 4](#_Toc5691727)

[**Scope of the project** 4](#_Toc5691728)

[**2.1 Scope** 4](#_Toc5691729)

[**2.2 Limitation** 5](#_Toc5691730)

[**2.3 Aim** 7](#_Toc5691731)

[**2.4 Objectives** 8](#_Toc5691732)

[**2.5 Overview of the scope** 8](#_Toc5691733)

[**Chapter 3** 8](#_Toc5691734)

[**Development Methodology** 8](#_Toc5691735)

[**3.1 Description of the methodology** 8](#_Toc5691736)

[**3.2 Design Pattern** 9](#_Toc5691737)

[**3.3 Architecture** 10](#_Toc5691738)

[**Chapter 4** 12](#_Toc5691739)

[**Project Planning** 12](#_Toc5691740)

[**4.1 Work Breakdown** 12](#_Toc5691741)

[**4.2 Milestones** 12](#_Toc5691742)

[**4.3 Gantt Chart** 12](#_Toc5691743)

[**Chapter 5** 13](#_Toc5691744)

[**Risk Managment** 13](#_Toc5691745)

[**Chapter 6** 13](#_Toc5691746)

[**Configuration Managment** 13](#_Toc5691747)

[**Chapter 7** 13](#_Toc5691748)

[**Conclusion:** 13](#_Toc5691749)

[**Chapter 8** 13](#_Toc5691750)

[**References** 13](#_Toc5691751)

# **Chapter 1**

## **Introduction**

### **Introduction of my Project.**

In this computing project I have created a 2D Game. In a 2D platform engine called Unity. Unity is the most essential software for creating an easy games. Among the so many games in the internet this game also contains so many advance features. This game is especially made for those peoples who feel boredom of the people. This 2D game helps them to stay sharp and concentrated. I have created this 2D Game that serves as a templates for the future games as well.

### **1.2 Background of the Project**

Without any background we could have so many problems at the future. We should have the clear vision of the background. The main thing that is need for 2D games the how people could enjoy the game. For the quick communication between the player and game character I have filled all the demands in a better way.

### **1.3 Problem Statement**

Problem statement are the clear statement that are in the existing system which need to be focused by the project developer team. The main focus of the problem statement is to find out what is wrong inside the system and analyze every aspects of the program we are designing.

* Sometimes there occurs a graphics problem while displaying the game.
* Program could not read or record the data of the game properly.
* Is not able to convey the message in proper way.

### **1.4 Description of the Project**

#### **1.4.1 Features of the Project**

The main features for this 2D game are as follows:

* Player can easy use this game.
* Players can easily use the menu bar.
* In this 2D Game
* Players could properly know about the score they have made.
* Players can clearly understand the mechanism of the game while playing it.
* Player can jump smoothly at the blocks while shooting.

### **1.5 Overview of the Project**

Before we use any kind of system we should have the clear vision of the project how it is going to be. Therefore Overview of the project is very necessary. In this 2D platform game the overall view is to provide a smooth surface game. And very Entertain to be played

# **Chapter 2**

## **Scope of the project**

### **2.1 Scope**

The main scope of this 2D game is to provide comfort while playing the game.

### **2.2 Limitation**

The limitations of the projects are:

* This is not much levels in the game system.
* Sometimes player may find problem shooting the objects...

### **2.3 Aim**

Followings are the aims of my projects:

• Aim to centralize every agricultural sector in a circle.

• To create better communication in between player and PC.

• To give every information to the farmers for the better productivity.

• To create a fine user-friendly website so that information sharing can be better.

### **2.4 Objectives**

Followings are the objectives to get above aims:

* Interview with the farmers and seller about how likely they are facing problem about their communication and information gathering system having manually.
* Educate them about the scope of automated system.
* Perform complete analysis and planning’s.
* Designing a website to fulfill the aims of the project.
* Meeting with the many farmers and market seller to make them know about this system its advantages over them.

### **2.5 Overview of the scope**

The main overview of the scope is to provide information about the future advantages and benefits.

It also provide information about how it will benefits to the players.

# **Chapter 3**

## **Development Methodology**

### **3.1 Description of the methodology**

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and trial high superiority software’s. The SDLC aims to yield a high-quality software that fulfills the customer prospects, reaches completion within times and cost evaluations. SDLC is a system describing task performed at each step in the software development procedure.

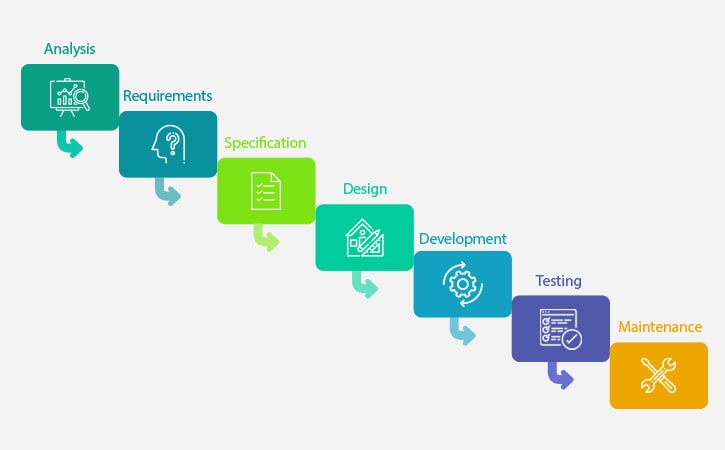


Fig 1.1 - Waterfall Model

Waterfall Methodology is one of the linear-sequential life cycle model which is very simple to understand and use. In a waterfall model, each phase must be completed fully before next phase can begin and basically used for the project which is small and there are no uncertain requirements. The sequential phases of waterfall model are:

Requirement gathering and analysis: This phase is the foremost phase of this model. Feasibility study, requirement analysis, planning, Software requirement specification (SRS), are the task performed in this phase.

System design: Requirement specification are studied and system design is made. It helps in specifying hardware and system requirement and helps in defining overall system architecture. Structural diagram, behavioral diagram, UI design, database design is made in this phase. Implementation: After design phases, system is implemented using concept of Object-Oriented Model and system is first developed in small units and then integrated in next phase for testing. Testing: The small units developed in the implementation phase is integrated to develop a complete system and different types of testing is done before deploying the: After doing various testing as unit-testing, functional non-functional testing system is deployed to the user or in the market.

Maintenance: After deploying of software many problems or bugs can be arise so to patch such errors and bugs maintenance is necessary.

I will be using waterfall model that is suitable for my project because of the following advantages:

* Simple and easy to understand and use.
* More paperwork helps providing a referenced document in project lifetime.
* One phases must be completed fully unless it will not allow to go to next phase so that phases do not overlap.
* Very beneficial for the small projects where requirements specification is better understood.

Disadvantages of using Waterfall method:

* Is not beneficial for haphazardly varying requirement.
* Not beneficial for large projects.
* It is very expensive to roll back in the initial phase in case of changing something.

### **3.2 Design Pattern**

Time is estimated for each activity of the system as follows:

Factory pattern, flyweight design pattern, abstract factory, MVC design pattern are some of the design patterns which are the layout for how to solve an issue that can be utilized as a part of a wide range of circumstances. In this project I am using MVC design pattern. MVC stands for model, view, and controller. Model handles the business logic, view accomplishes the display of the data, controller allows user to cooperate with model and is a structural design pattern.

Reason of using MVC pattern:

* Modules are divided into model, view controller so that management of different class are very easy.
* If arises any error or changes in the one model, no effect in other modules.



Fig 1.3 - MVC Pattern

### **3.3 Architecture**

System architecture is the conceptual model that defines the structure, behavior and more views of a system.

3-Tier Architecture:

* Is a kind of software architecture which is made of three “layers” of logical computing?
* Applicable in application

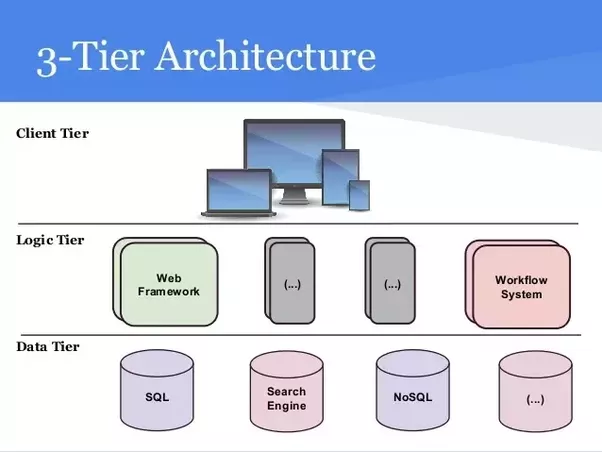


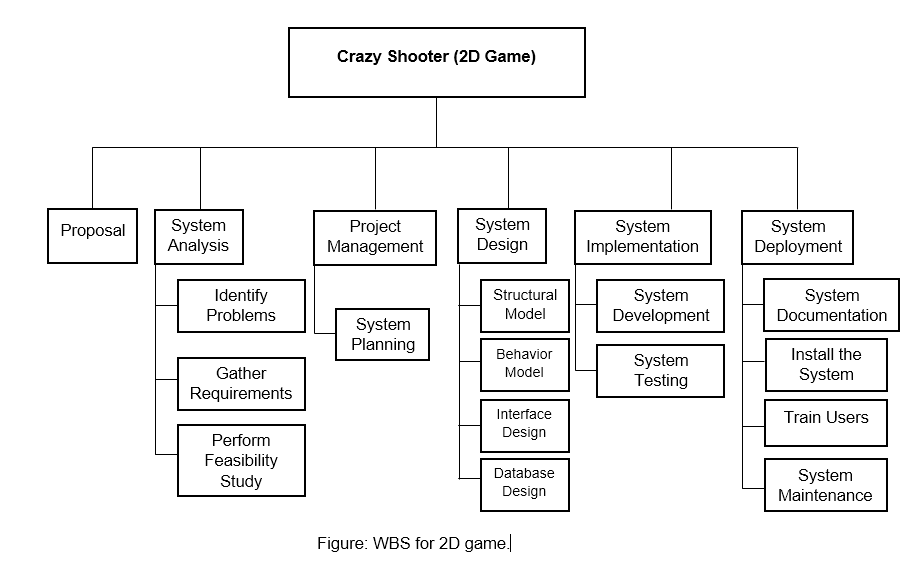
Fig 1.4: Three-tier architecture Diagram

# **Chapter 4**

## **Project Planning**

### **4.1 Work Breakdown**

A proposal for the system is developed through the use of various techniques and plans. The major aims, objectives, features and system development methodology for the system is properly mentioned. The system is break down into smaller components through the use of WBS for easy management. Similarly, time has been estimated to meet the project submission deadline. Various milestones is set to meet the deadline of the project. The risk in system development has been analyzed and further action that needs to be taken are well documented.



### **4.2 Milestones**

|  |  |
| --- | --- |
| Milestones | Date |
| **Project Management**  Risk Management  WBS  Configuration management  Proposal Submission | July 11, 2019  June 30, 2019  July 6, 2019  July 10,2019  July 11,2019 |
| **Analysis**  Feasibility study  Requirement analysis  Use Case  Architecture(Initial Class Diagram) | August 5, 2019  July 19, 2019  July 23, 2019  July 29, 2019  August 5, 2019 |
| **Design**  Structural Diagram  Behavioral diagram  UI Design  Database Design(Data Dictionary, ER Diagram) | September 4,2019  August 13, 2019  August 20, 2019  August 29, 2019  September 4, 2019 |
| **Implementation**  Build database  Coding | September 24,2019  September 10, 2019  September 24, 2019 |
| **Testing**  Unit Testing  Integration testing  Black box Testing  White box Testing | October 6, 2019  September 26, 2019  September 28,2019  October 2, 2019  October 6,2019 |
| **Deployment/Reporting**  User Training  Final report | October 18,2019  October 13, 2019  October 18, 2019 |

Fig 1.2- Milestone

### **4.3 Gantt Chart**

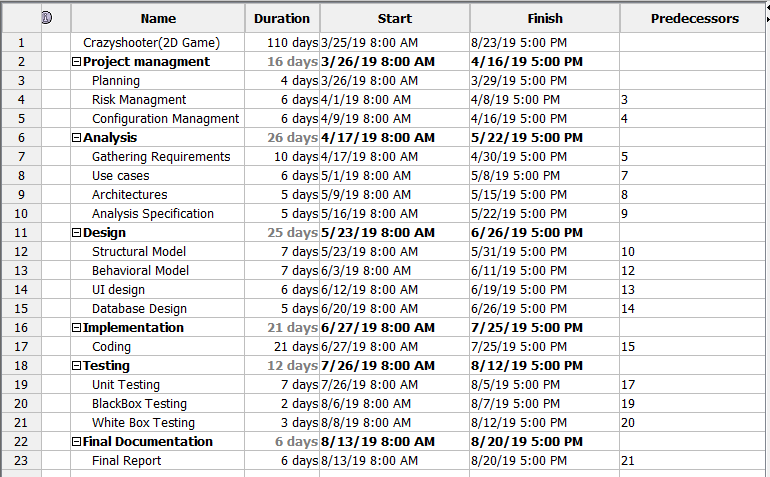
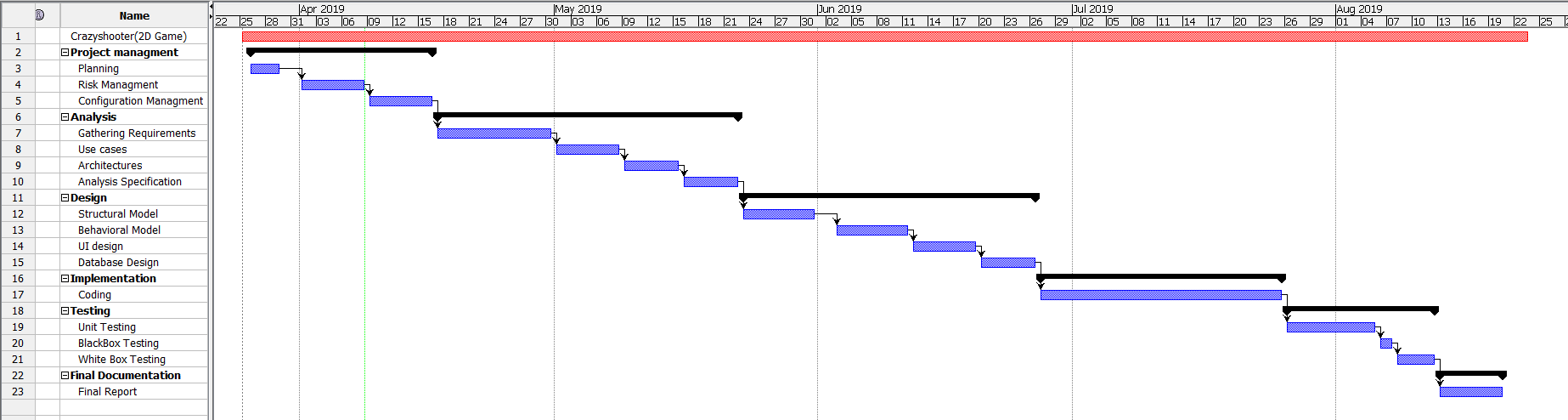


Figure: Gantt chart



# **Chapter 5**

## **5.1 Risk Management**

Lots of risk may occurs various risks may arise while doing the project. This may impact the project in terms of performance and time estimation. We have to identify and manage such risks with their likelihood of occurrence and their possible consequences.

The following table shows the corresponding values for likelihood and consequences of the risks:

Impact= Risk likelihood \* Risk Consequences

|  |  |
| --- | --- |
| Likelihood | Value |
| Low | 1 |
| Medium | 2 |
| High | 3 |

Table 1.3: Risk likelihood value

|  |  |
| --- | --- |
| Consequences | Value |
| Very low | 1 |
| Low | 2 |
| Medium | 3 |
| High | 4 |
| Very High | 5 |

Table 1.4: Risk consequences value

List of Risks:

There can be chances of arising so many risks during the phase of software:

* Cost forecasts are inaccurate.
* Under communication
* Ambiguous requirements
* Changes of requirements
* Inexperienced resources
* Natural disaster
* Evolvement of low quality software
* Game crash

Risk management table:

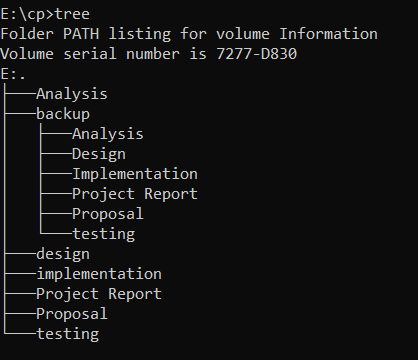
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Likelihood | Consequences | Impact | Action |
| Cost forecast are inaccurate. | 2 | 5 | 10 | Strong cost analysis should be done at the very beginning. |
| Under Communication | 2 | 5 | 10 | Good communication between player and computer must be made in order to remove horrific consequences. |
| Ambiguous requirements | 1 | 3 | 3 | Requirements must be clear and should be open to interpretation. |
| Changes of requirements | 1 | 2 | 2 | Regular meetings, interviews, feedbacks should be taken. |
| Inexperienced resources | 1 | 2 | 2 | Good training is a must. |
| Natural Disaster | 1 | 5 | 5 | Back up of data is a must. |
| Evolvement of low quality software | 1 | 4 | 4 | Well maintained development process should be used. |
| Software crash | 2 | 3 | 6 | Regular Back up of every data should be done using online cloud storage or other storage devices. |

Table 1.5: Risk management table

# **Chapter 6**

## **6.1 Configuration Management**

Configuration Management activity and its progress is recorded using separate sub directories under one main directory. Every details of the change in the project is recorded and managed using configuration management. A backup plan is developed which helps in restoring previous version if anything goes.



# **Chapter 7**

## **7.1 Conclusion**

I have prepared this proposal for the system through the use of various creative ideas. With lots of major aims, objectives, features and system development methodology for this 2D game I have mentioned everything in detail. With the help of unity engine I have tried to mention the Design Pattern, Architecture and different development methodology. I have also break down the system into different smaller components through the use of work breakdown system for easy management. And finally in the project planning I have estimated different time interval for analyzing, designing and implementing the 2D game through the help of milestones and Gantt chart.

# **Chapter 8**

## **8.1 References**