GAME DEVELOPMENT: PROBLEMS

Sakshi Shukla, Rayan Dutta, Rudrani Mishra

B.Tech Undergraduate Student, Computer Science Engineering, SRM University, Tamil Nadu, India
B.Tech Undergraduate Student, Computer Science Engineering, SRM University, Tamil Nadu, India
B.Tech Undergraduate Student, Computer Science Engineering, SRM University, Tamil Nadu, India

ABSTRACT

Many people can be found playing games on their mobile phones when they're taking the bus, the tube, or just wanting to pass the time. Games can help people socialize and learn, and they're extremely fun to play and enjoy, and the fact that this can be achieved on a mobile means we can always have games at the ready for those boring periods in our lives. Its said, " All work and no play, makes Jack a dull boy". 'Space chase' is a game designed keeping this in mind including budget, schedules, so that the user can play it anytime. The game can be played by a 1 year old kid to a 70 years old man. Space chase is made simple keeping many things in ming There will be a rocket in space being chased by asteroids and meteoroids. It will be given some power ups at times which can be used to destroy asteroids and meteoroids. If the rocket is attacked by asteroids or meteoroids, the game is over. The specialty of this game will be its GUI. It will be designed in Unity using C# codes. The GUI of the game is designed using Photoshop. There will be three modules in the game which will be the play screen, the home screen and the high score screen. As the name suggests, the play screen is the main screen where the user will play. The home screen is the like the menu card, one can choose what he must do. The game over screen will come once the user quits or he loses. The main aim of the game is to make the player relax and just simply play the game in his leisure time.

Keyword : *Game, Space Chase, development, problems*

1. INTRODUCTION

The project aims to provide some fun to user's life whenever he is free or is bored. Nowadays everyone's life is a race. But it's said, "All work and no play, makes Jack a dull boy." That play can be indoor or outdoor. Today, indoor games are not limited to carrom or chess. With the advancement in technology, new games are being developed. From Mario to FIFA, gaming is at utmost development. As people moved from Nokia 1100 to iPhone 8, games also moved from 'Snake' to 'Mini Militia'[2].

Keeping the people's interest in mind, we have designed a simple game whose sole purpose is make players relax. The aim of the game is simple. The name of the game is 'Space Chase'. As the name suggests, the game is based on space and spaceship. There is no successful movie without a villain, how can a game be successful without enemy. Our game has a very rough, tough and solid enemy, obviously that can be asteroids, as game set up is in Space. So, with no surprises left, there will be a spaceship and lots of enemies, whom the player must destroy or they will destroy the spaceship, hence the player will die.

Also, the best part of the game is its design, which has been designed using Adobe Photoshop. The background consists of various constellations with fusion of two colors. The spaceship is simple. The asteroids are in the shape of big rocks. The player can shoot the asteroids to destroy them. The shoot is like a glimpse of light passing very fastly. Once the player is unable to defend its spaceship and is attacked by the asteroids, the game gets over. Also this is a single level game.

The game is designed in Unity using C# codes.

1.1 Problem Statement

As it is said, no system can be ideal, i.e. 100% efficient. Game development is an extremely complex activity, and a much harder task than one can initially imagine[1]. Creating a game is a different experience from what it was in 1994, and certainly much more difficult, having grown in complexity in recent years.

The main goal of this project is to collect and discuss the problems that afflict the electronic game industry, since surprisingly very little attention has been paid to collecting information about them, to organizing this information in a systematic form, and to quantifying the frequency with which they occur. To achieve our goal, we have taken the following steps:

- reanalyzed the problems of the traditional software industry, collecting this
- information to compare it with the game industry;
- reanalyzed the problems of the game industry that are cited in specialized literature, making it possible to map such problems when analyzing the post-mortems;
- recollected via postmortem analysis (focusing on aspects of software engineering)[3], the real problems cited by professionals in the game industry
- a qualitative analysis of the outcome;
- compared the results obtained from the game industry to the data from the traditional industry and searched for similarities and idiosyncrasies.

- The specialized literature on electronic games describes in exhaustive detail the scheduling problems in game projects. Althoughthey usually begin with reasonably structured schedules, bright ideas, and enthusiasm, something always goes wrong.
- The knowledge of necessary software and scripts is also important[8]

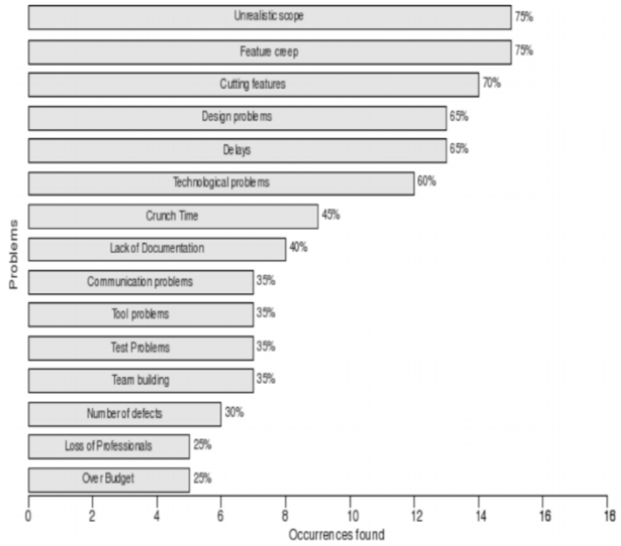

Figure 1- Problems found

1.2 Objective

The objective of this project is to simplify the problems related to the development of game. Many a times, we forget about the time, and things needed to grow tree, what we remember is the fruit. If the basic requirements are considered properly, won't the fruit be tastier? It is well said, "what you sow, so you reap".

The development of the game should be looked upon. Projects with budget problems are characterized by demands for investments above the estimates[4]. Therefore, the budget of the project should be as minimum as it can be. It helps both user and developer. The time required by the project should be utilized in making good designs. From a customer's point of view, a project has quality problems when the final product is different from the one expected. Quality is associated with the customer's satisfaction, aesthetic aspects, and the system's ability to meet its requirements. Here, the customer is the player/user who will play the game.

Management problems happen in several ways, including bad communication, lack of investment in the team's training, and neglecting to inspect the project. Communication is important in real life as well as in completing projects. Communication can be anything, between computer and programmer or between two programmers. Communication between the programmers is important as the whole project resides on them. Thus, keeping these things in mind, the evolution of game development can be made simple[5].

2. LITERATURE SURVEY

Game Design is the art of applying design and aesthetics to create a game for entertainment or for educational, exercise, or experimental purposes. Increasingly, elements and principles of game design are also applied to other interactions, particularly virtual ones

Game design creates goals, rules and challenges to define a board game, card game, dice game, casino game, role-playing game, sport, video game, war game or simulation that produces desirable interactions among its participants and, possibly, spectators[1][6].

Academically, game design is part of game studies, while game theory studies strategic decision making (primarily in non-game situations). Games have historically inspired seminal research in the fields of probability, artificial intelligence, economics, and optimization theory. Applying game design to itself is a current research topic in metadesign.

A game concept is an idea for a game, briefly describing its core play mechanisms, who the players represent, and how they win or lose.

A game concept may be "pitched" to a game publisher in a similar manner as film ideas are pitched to potential film producers. Alternatively, game publishers holding a game license to intellectual property in other media may solicit game concepts from several designers before picking one to design a game, typically paying the designer in advance against future royalties.

During design, a game concept is fleshed out. Mechanisms are specified in terms of components (boards, cards, on-screen entities, etc.) and rules. The play sequence and possible player actions are defined, as well as how the game starts, ends, and what is its winning condition. In video games, storyboards and screen mockups may be created[7].

Game testing is a major part of game development. During testing, players play the game and provide feedback on its gameplay, the usability of its components or screen elements, the clarity of its goals and rules, ease of learning, and enjoyment to the game developer. The developer then revises the design, its components, presentation, and rules before testing it again. Later testing may take place with focus groups to test consumer reactions before publication.

During testing, various balance issues may be identified[9], requiring changes to the game's design. Video game testing is a software testing process for quality control of video games. The primary function of game testing is

the discovery and documentation of software defects (aka bugs). Interactive entertainment software testing is a highly technical field requiring computing expertise, analytic competence, critical evaluation skills, and endurance.

2.1 Existing System

Video game development is the process of creating a video game. The effort is undertaken by a game developer, which may range from a single person to an international team strewn across the globe. Traditional commercial PC and console games are normally funded by a publisher, and can take several years to reach completion. Indie games can take less time and can be produced at a lower cost by individuals and smaller developers. The independent game industry has seen a substantial rise in recent years with the growth of new online distribution systems, such as Steam and Uplay, as well as the mobile game market, such as for Android and iOS devices.

The first video games were noncommercial, and were developed in the 1960s. They required mainframe computers to run and were not available to the general public for play. Commercial game development began in the 1970s with the advent of first-generation video game consoles and early home computers like the Apple I. Due to low costs and low capabilities of computers, a lone programmer could develop a full game. However, approaching the 21st century, ever-increasing computer processing power and heightened consumer expectations made it difficult for a single person to produce a mainstream console or PC game. The average cost of producing a triple-A video game slowly rose from US$1–4 million in 2000 to over $5 million in 2006, then to over $20 million by 2010.

Mainstream PC and console games are generally developed in phases. First, in pre-production, pitches, prototypes, and game design documents are written. If the idea is approved and the developer receives funding, a full-scale development begins. This usually involves a team of 20–100 individuals with various responsibilities, including designers, artists, programmers, and testers.

Game development is a software development process, as a video game is software with art, audio, and gameplay. Formal software development methods are often overlooked. Games with poor development methodology are likely to run over budget and time estimates, as well as contain a large number of bugs. Planning is important for individual and group projects alike.

Overall game development is not suited for typical software life cycle methods, such as the waterfall model. One method employed for game development is agile development. It is based on iterative prototyping, a subset of software prototyping. Agile development depends on feedback and refinement of game's iterations with gradually increasing feature set. This method is effective because most projects do not start with a clear requirement outline. A popular method of agile software development is Scrum.

Another successful method is Personal Software Process (PSP) requiring additional training for staff to increase awareness of project's planning. This method is more expensive and requires commitment of team members. PSP can be extended to Team Software Process, where the whole team is self-directing.

Game development usually involves an overlap of these methods. For example, asset creation may be done via waterfall model, because requirements and specification are clear, but gameplay design might be done using iterative prototyping. Development of a commercial game usually includes the following stages:

- Pre-production or design
- High concept
- Pitch
- Concept
- Game design document
- Prototype
- Programming
- Level creation
- Art production

- Audio production
- Testing

2.2 Issues In Existing System

The analysis of problems in the development of software systems goes back to the origins of software engineering. The famous NATO conference in 1968 [NATO 1968], in which the term software engineering[9] was created, had as its basic goal the analysis of the key problems in the critical areas of program construction.

Although the academic community has being discussing the problems that afflict the development of computational systems for almost 40 years, we still have difficulties in elaborating elegant and correct software that satisfies customers and meets estimates, as much in time as in budget. The current status of software development can be seen as far from ideal. It has been suggested that software is one of the most problematic aspects of the corporate world: budgets and deadlines are not met and errors are more common in large software projects than in almost any other business area. Quality is associated with the customer's satisfaction, aesthetic aspects, and the system's ability to meet its requirements. From the technical point of view, the most common reason for disasters in software projects is poor quality control. The development teams get lost when the scope of the project grows very large and complex, and they have to deal with the highly specific requirements of the games domain as well.

Another common form of feature creep happens when external code (a new component, for example) is incorporated without planning so as to save time. However, it frequently becomes necessary to work extremely hard to integrate this new component. A third form of feature creep takes place when developers decide, despite having a set of solid libraries, to implement their own algorithms.

However, there are a lot of examples in the games industry where features discovered during the development phase transformed a game into a success. Game development is not a linear process. Hence, if an interesting function is discovered, it must be analyzed in terms of risk and, if viable, be added to the project's schedule.

All games depend on technology. Moreover, the technologies used in games are so advanced that game companies are the leaders in graphical computation. However, cutting-edge technology brings risks as well, since using it may frequently involve great effort and a large investiment in time. The technological risks are generally higher when the team is working on a new platform that has not been completely delivered or consolidated. The first of these risks is that no developer has ever worked in it; the second is that frequently the platform hardware still contains problems that are only found by development team at the launch title.

Feature creep is the process by which new modules are added without planning during software construction. As well as feature creep there is another problem that is mentioned frequently: that of cutting features during the development process. Due to the fact that projects initiate an overly ambitious number of functionalities, some of which are even implemented, but for a variety of reasons they end up being cut further on in the project. Although designing is a common practice in game projects, mainly by using a design document or game bible, many reports claim that there are problems in this phase.

Many game projects mention delayed schedules.

The failures of third-party application program-

The failures of third- party application programming interfaces (APIs) for the platform or hardware. Although many projects reported sucess with documentation processes, mainly with the project document, many postmortems cite congenital problems due to lack of documentation[3].

Some projects were characterized by the great number of defects found in a development phase. Some commented that their projects were seriously affected by the loss of professionals.

3. SYSTEM ANALYSIS

Specification of a project comprises of the particular needs to be satisfied, or essential characteristics that a project needs to be completed successfully. Specifications are written usually in a manner that enables both parties and/or an independent certifier to measure the degree of conformance. They are, however, not the same as control limits which allow fluctuations within a range and conformance to them does not necessarily mean quality which is a predictable degree of dependability and uniformity. Specifications are divided generally into two main categories:

- Performance specifications: Conform to known customer requirements such as keeping a room's temperature within a specified range.
- Technical Specifications: It express the level of performance of the individual units, and are subdivided into a)Individual Unit specification which state boundaries/parameters of the unit's performance consisiting of a nominal value and tolerance departure from the nominal value, (b) acceptable quality level which states limits that are to be satisfied by most of the units, but a certain percentage of the units is allowed to exceed those limits, and c) Distribution specifications which define an acceptable statistical distribution for each unit, and are used by a producer to monitor its production processes.

Likewise, Space Chase is time-consuming, concise and require more thinking, lesser algorithms and codes. The specification of Space Chase is basically a game design software, codes and the supportable device to configure the gaming app.

The specification based on the performance and technical specifications are:

- Hardware Specifications: A laptop of 64 Bit processor with a minimum of 8 GB RAM.
- The game shall be designed in Unity Engine mostly using C# codes.

Game Design is the art of applying design and aesthetics to create a game for entertainment or for educational, exercise, or experimental purposes. Increasingly, elements and principles of game design are also applied to other interactions, particularly virtual ones. Game design creates goals, rules and challenges to define a board game, card game, dice game, casino game, role playing game, sport video game, war game or simulation that produces desirable interactions among its participants and possible, spectators.

Academically, game design is part of game studies, while game theory studies strategic decision making (primarily in non- game situations). Games have historically inspired seminal research in the fields of probability, artificial intelligence, economics and optimization theory. Applying game design to itself is a current research topic in metadesign[2].

What do games do to people? Using tools and methods such as surveys, controlled laboratory experiments, and ethnography researches have investigated both the positive and negative impacts that playing games could have on people. More sociologically informed research has sought to move away from simplistic ideas of gaming as either 'negative' or 'positive', but rather seeking to understand its role and location in the complexities of everyday life.

The scope of a project is its breadth and depth, what are you making, and how complex does it have to be achieve what you want? Scope is something to be decided, not something to be discovered post-facto. Scope, most simply put, is the understanding of how vision meets execution. The goal behind making this project was to design something that catches attraction to the user with its graphics and gaming sequence. The project results into a perfect game that can be useful to the user for killing time when one is sitting idle. This report describes all the requirements of the project. The player can play the game as long as he wants as there are no levels. The game encourages creativity and daring via branching pathways. We demonstrate the action flow between inputs, script, display(output). The project is the result of the creativity and graphics designed using C# codes.

3.1 Overall Description

The overall description of a project includes the features, components software and hardware specification comprising of which the project is successfully made. The aim of the project is to design a gaming application that helps the user to get some new gaming experience[5]. The project is developed using Unity project and the necessary C# codes. The project constitutes of a rocket and meteorites shower, the game proceeds with shooting the meteorite and moving ahead by making sure that the path of rocket is not hindered by any of the meteorite. The body of the rocket and meteorites is developed using Unity graphic features.

3.2 Product features

The features of a successful project should be its accessibility, platform independent and user friendly. Similarly Space Chase serves proper accessibility to the user. The gaming application is expected to not to lag while using it and this project completely fulfills the need of lag-free gaming experience. The major features of this project is the High Score Board; the high score module shows the score achieved by the user. More the path is covered without any hindrance, more score is gained by the user.

The other feature of Space Chase is replaying the game once you have lost, the replay option comes on the same window after you hit some meteorite and lose the game. With the replay option, user can also view his score card and can even quit the game. The application provides a smooth gaming experience. The reason behind providing lag-free experience is because of its 2Dimensional development[10]. The 2-D game provides the user with amazing graphical interface through which the gaming interface looks interesting. The GUI consists of the background of space, a rocket and shower of meteors and asteroids.

4. CONCLUSION

Thus, the game industry can grew bigger and better with the change in these problems. Also, the game developed can be given more levels, 3D effects and better user interface. This can also increase the demand of game industry.

5. REFERENCES

[1] Multiplayer Online Battle Arena, December 12th, 2015, http://www.giantbomb.com/multiplayer-online-battle-arena/3015- 6598/

[2] Our Games | Riot Games, http://www.riotgames.com/our-games

[3] Craighead, J., Burke, J. and Murphy, R., Using the Unity Game Engine to Develop SARGE: A Case Study (2008)

[4] Delap, S., Stafford, R., League of Legends: Scaling to Millions of Ninjas, Yordles, and Wizards, 7th April, 2011., https://www.infoq.com/presentations/League-of-Legends

[5] Prabha, C., Login and Networking Services for Online Multiplayer Games, Master's Thesis, Helsinki Metropolia University of Applied Sciences (2015)

[6] Bharambe, A., Pang, J., Seshan, S., Colyseus: A Distributed Architecture for Online Multiplayer Games, 3rd Symposium on Networked Systems Design & Implementation, San Jose, California, United States of America (2006)

[7] Assiotis, M., Tzanov, V., A Distributed Architecture for MMORPG, The 5th Workshop on Network & System Support for Games, Singapore, (2006)

[8] Hsu, C., Ling, J., Li, Q., Kuo, C.C.J., On the Design of Multiplayer Online Video Game Systems, University of Southern California (2003)

[9] Photon Unity 3D Networking Framework SDKs and Game Backend | Photon: Multiplayer Made Simple, https://www.photonengine.com/en-US/PUN

[10] Custom Authentication | Exit Games, https://doc.photonengine.com/en/realtime/current/reference/custo m-authentication