iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** III **Month of publication:** March 2022

DOI: https://doi.org/10.22214/ijraset.2022.40800

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue III Mar 2022- Available at www.ijraset.com

Universal Real-Time Strategy Game in Unreal Engine

Sudharsan S
B-Tech, Computer Science, Dr. M.G.R Educational and Research Institute, Chennai, India

Abstract: ***In the rapid growing field of game development and game design, the RTS genre of games continues to attract substantial number of gamers. RTS genre games not only provide entertainment for gamers but also the ability to yest and grow their real time application of tactical and logical decision-making skills. RTS games are lately very popular but a brief number of games are available with similar concepts. This reputation in the story line of the available games makes the long-term game play less desirable. The project intents to develop a spin-off RTS game with major emphasis on level design, gameplay mechanics, user interface development and resource management.***
Keywords: Unreal Engine, Blender, Gaea, Mesh, the Home Line.

I. INTRODUCTION

Real-Time Strategy games (RTS) are a genre of computer and video games. They take place in real-time and involve resource gathering, base building, technology development and high-level control over individual units. Within a real-time strategy game, players become the leader of a colony or military base. For example, in Age of Empires, the player controls an explorer, which is a unit capable of exploring the map, creating a settlement and acquiring special buildings for the player. At the beginning of the game several players are teleported into the world, where they begin by constructing a base. Real-time strategy games have a strong economic side, with players in Age of Empires constructing, upgrading and managing a variety of buildings that produce the three basic resources – food, wood and coin. Players invest these basic resources into improving their economy, fortifying their base, strengthening their military, and developing various forms of technology. Once constructed, players maneuver their military units around the world, attacking and engaging enemy units, the ultimate goal being to destroy the opponent's explorer and level the base of any opponents foolish enough to challenge their supremacy.

Underneath the surface game-play, video games are fundamentally about making decisions and exercising skills. A car racing simulation involves a great deal of skill in controlling the vehicle along with decisions involving the choice and setup of the vehicle. Real-time strategy games, while varying greatly in content and style, are unified by a set of common decisions that their players make. These decisions involve a variety of challenging problems that players are simultaneously solving: resource allocation - developing their economy; force composition - training and equipping an effective military; opponent modelling - estimating the location and composition of enemy forces; spatial reasoning - predicting incoming attacks or defensive vulnerabilities, while hiding and misleading their enemies about their own intentions. In contrast to other types of games, the decisions involved in RTS games concentrate player involvement around making high level, long term strategic decisions. The paradigm is designed to draw players into the game world giving them a set of interesting decisions to make, along with compelling reasons about why they are making them and their resulting consequences. Real-Time strategy games present a unique opportunity for research, containing a variety of interesting research problems within an integrated and motivated whole. traditional arduous methods. Deep learning techniques are very successful in image classification problems.

In developing a real-time strategy game, the following aspects are essential:

1) The playing fields.
2) The graphics engine.
3) The game interfaces.
4) Management of unit data.
5) Artificial Intelligence Agent Navigation for successfully moving units from point A to point B.
6) Unit states and actions.
7) Unit interaction.
8) Designing the game mechanics.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue III Mar 2022- Available at www.ijraset.com

During the last decade, the scientific community has acknowledged that real-time strategy games constitute rich environments for researchers to develop complex resource management schemes and battle strategies which can be related to real world events. Much of the research that can be found about the development of real-time strategy games concerns the mixture of genres and subgenres of video games to create an immersive experience for players. An example is the combination of the role-playing game genre and the real-time strategy genre in the popular game, Ages of Empire.

II. PROPOSED SYSTEM

The methodological approach taken in developing the real-time strategy for this project is divided into 8 specific areas listed below, in no specific order. While some aspects of the approaches can be simultaneously combined, others may require spontaneous instalments or preceding approaches. For example, the steps taken in designing the Real-Time Camera can be combined with those taken in rendering it through the User Interface. However, the Game Design has to precede the Level Design.

A. Game Design

To develop this game, CryEngine 3, Unity 4, Shiva 3D and Unreal Development Kit (UDK) were compared to choose the most suitable engine for real-time strategy. At the end of the analysis, UDK was selected based on availability of previous work and system capabilities. The UDK is a game engine developed by Epic Studios and it is the free version of Unreal Engine 4. A previous real-time strategy game built using UDK is Hostile Worlds and this project will combine elements from that game. A number of assets developed using 3D applications like Fuse, 3DS Ma, Maya and/or Blender. After purchase, the assets are then imported into the UDK for implementation. There are a lot of permutations in terms of AI for real-time strategy. Programmers are known to use algorithms like A*. But for this project, the explicit use of path nodes and navigation points are required. Unreal Kismet, a visual scripting tool in UDK, and Unreal Script, UDK's proprietary programming language were used to provide the artificial intelligence features in the game. The cinematic and animation sequences for the game were done in the Unreal Development Kit using Unreal Matinee and Unreal Kismet. Unreal Kismet is a built-in application in Unreal Development Kit that allows developers to design games using visual scripting. Unreal Matinee can read the nodes in Kismet and use them as triggers to play the sequences. Levelling-up is and RPG feature which was incorporated in the game. The algorithm is precise and simply indicates that when the player reaches a certain pre-defined score, his level increases and his 'stats' also increase. His stats may include the population of his army, the strength of his army and his base and others. Levelling-up is an excellent way to increase the play and replay value for the game. The score system will be dynamic and will be triggered by different actions in-game such as defeating monsters, collecting artifacts and deploying units. When this increases to a certain level, the player may level-up.
For the user interface, Scale form was extensively used in the design. The workflow is as thus: sketches of the concept design of the interface were rendered and cleaned in Photoshop and then imported into Flash Professional integrated with Scale form for animation. Real-time strategy games are usually played using a combination of the keyboard and mouse and that method will also be followed in this project.

B. Resource Management

Economy is rather important in real-time strategy. It is possible for the player to upgrade his army by using resources which will be available either at specific points or throughout the game. The primary resources in the game are shards, which may be used to purchase units. The game allows for the use of various strategies to be used as regards economic development. For the long-run, it is possible to develop heavily in economic growth while leaving military weak at first only to balance this out in the end-game. This strategy allows the player to have excess resources which are always useful in a real-time strategy game. Another strategy is to develop both economic growth and military strength side by side. Here, the player has just enough resources to either build an army or to develop infrastructure but not both. Players are advised to experiment and come up with their own strategies as well.

C. Battle Systems

Real-time strategy veterans always have various techniques for battle. One such method may be to use a combination of different types of soldiers to try to achieve a balance. Another may be to invest heavily in just one specific rank and upgrade that rank to the maximum. Also, defence systems such as turrets and towers may be included in the game to increase the options for military tactics in-game. Deploying and upgrading units require the use of resources and the amount of resources required may vary, depending on the type of unit and its rank. The units deployed by the player follow scripts written in Kismet and Unreal Script that allow the units to open fire on enemies, gather resources and protect the base. Navigation points and path nodes are also developed for the game.

©IJRASET: All Rights are Reserved

D. Real-Time Camera

The real-time aspect of the game requires a camera that delivers a top-down or isometric view of the game. This will be one of the first problems to solve in developing the game as the default camera is usually a first-person camera or a third-person camera.

E. Level Design

Level design is a complex process which requires numerous sketches and editing as well as experimenting within the editor to get it right. But Unreal Development Kit is a high-end engine that can deliver stunning environments in record time. Hence, designing and sculpting the level does not take too long.

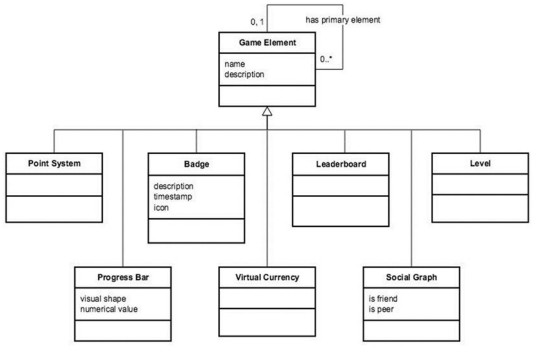
F. User Interface Design

The user interface is comprised of the menu screens that control the game mode and the Heads-Up-Display screens that show the player's status in-game. They were designed with Photoshop, Flash Professional and Scaleform. The flow for the user interface development. The heads-up-display interface can either be very simple or very complex, depending on the style or theme intended to be used. For this project, the HUD includes a mini-map interface, a shard interface, a score interface, an abilities interface and a status interface. The mini-map interface consists of height maps calculated using the levels created in the UDK and a cursor and a compass to show the location of the player and the locations of targets and objectives. The score and shard interfaces display the player's scores and shards respectively. The abilities interface provides a means of access to the player's unit abilities and skills. The status interface shows the current status of the player's units in terms of health and ammunition. The game modes were designed in a menu format, with each mode having its own menu and submenu. The front-end menu is the first menu that the player interacts with as it leads to the other menus. Next is the main menu which has the options menu, credits menu and game modes as sub menus. The options menu has its submenu composed of radio buttons, sliders and check boxes, and the credits menu simply rolls out the cast of the production team. The game modes are the join game and host game modes which are used to join or host a game respectively, which use the architecture provided in the original AEO.

G. Sound Design

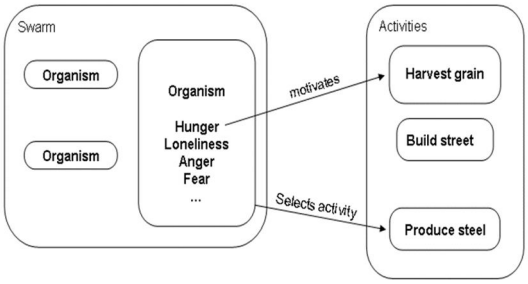
Designing and developing the sound files for the game is a complex process that actually starts with conceptualization of what the atmosphere in the game should 'feel' like. The inclusion of the Reverb Volume in the game allows direct manipulation of any sound file introduced into the game at runtime. The reverb effect, along with every other kind of effect can also be produced in Fruity Loops Studio. A recommended alternative for sound producers who cannot acquire or use FL Studio is Adobe Audition, although Audition is not free either. FL Studio is the audio editing software of choice used for this project because of its complexity, efficiency, effectiveness and robustness. For the voice-over and announcer systems, the recording feature is used to produce high quality audio for feedback. The background audio is mastered in FL Studio using a combination of the playlist, step sequencer, piano roll, browser and the mixer. The playlist contains all the patterns, audio clips and automation clips included in the soundtrack. The step sequencer contains the individual elements and instruments imported from the browser used in creating current the patterns and audio clips. The browser contains a list of all the instruments, sound files, audio clips and elements that could be used in FL Studio. The mixer is used to master the beat and contains slots for the application of effects such as reverb, panning, volume, compressor, equalizer, filter, mute, stereo enhancer amongst others.

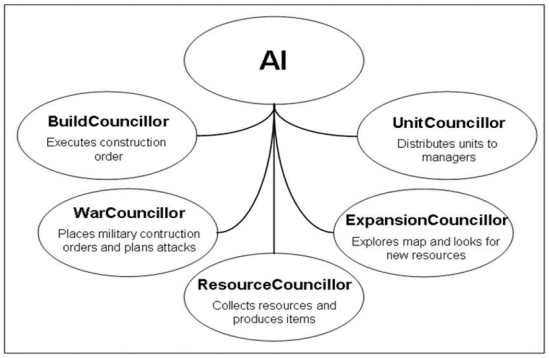
III. CLASS DIAGRAM



ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue III Mar 2022- Available at www.ijraset.com

IV. ORGANIZATIONAL DIAGRAM





V. MODULE DIAGRAM AND DESCRIPTION

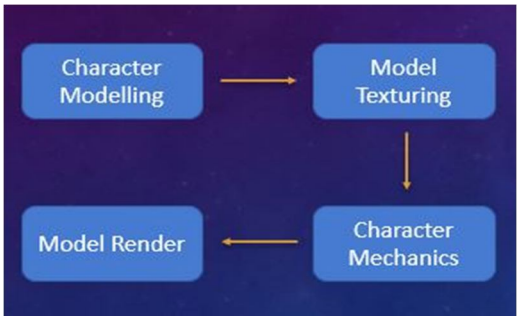
A. Module 1

The following module contains various facilities like Height Map Generation, Master Material Modelling, Landscape Texturing which makes up the landscape. Asset Modelling, Asset Texturing, Asset Render in making the assets needed. Level Mechanics and Level Render which is compiling and fine tuning the lighting and colour systems.



B. Module 2

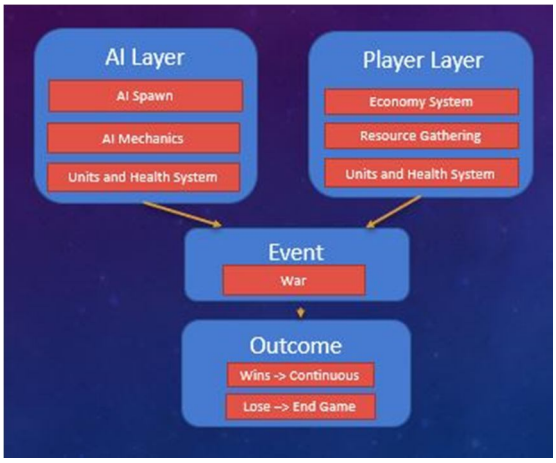
The following module contains various facilities like Character Modelling which is making the 3D model. Model Texturing is adding textures to the model. Character Mechanics include the movement and other physics systems for the player model and assets built in for. Model Render is computing the lighting and colour systems.



©IJRASET: All Rights are Reserved

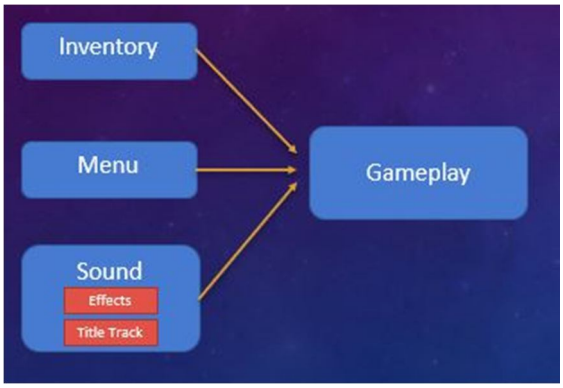
C. Module 3

The following module contains various facilities like AI Spawn, AI Mechanics and AI Unit systems along with the health systems together making the AI Layer. The Economy, Resource Gathering and Resource management, Units and Heath Systems for the player. The Event which is War and Outcome depends upon the factors like Player inputs and Player Layer.

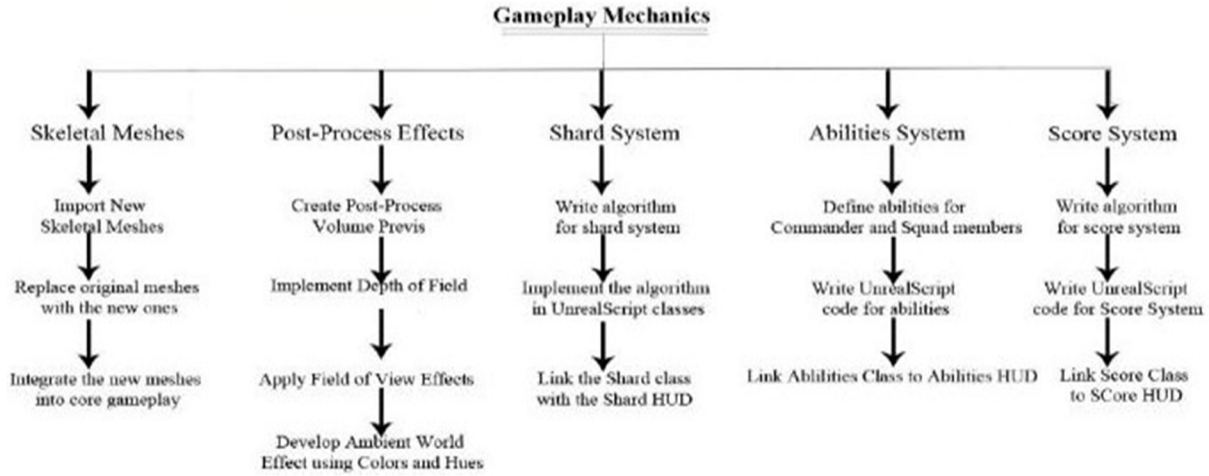


D. Module 4

The following module contains various facilities like the Inventory which is an UI allowing the user to manage and uses the resources collected. The Menu is the where the user would be being the game with and the options to select the quality, sound controls and to load the saved game or to begin and new one. Sound deals with the sound effect in the game play and Title track which would be played while accessing the menu.



VI. GAME MECHANICS



VII. RESULT



VIII. CONCLUSION

A. *Game Engine*

The Unreal Engine 5 offers the most features of all found engines that are relevant for this project. The compatibility of these features is thoroughly proven by the community; we don't run the risk of having to use unverified and unpredictable 3rd party components. Unreal has a very big community and offers a wealth of knowledge that developers can rely on. Even if the Unreal Engine 3,4 might be cumbersome at first, it is a great opportunity to really get to know this state-of-the-art game engine and explore it to its core. Another point not to neglect is UDK's revenue share model, which provides free access to this great game engine on development – payment is only necessary if the game results in a financial success later on.

B. *User Interface and Level Design*

The main textures and low-poly models like in AOE. The colour scheme for each level is not very rich but saturated and contrasted with a lot of black. All units the player can interact with are distinguished from the background. This way, a very bright, mysterious, dreamy atmosphere is achieved. The main level shows an earth planet with a very bright sun. The colour scheme is mostly blue and green. It feels like an Island where a life and resources to help the life strive exists. The realistic landscape and architectural structures.

C. *Gameplay*

To collect the resources and develop the economy and charge against the enemy. The player would have the control to select and cultivate troops and city building according to the resources collect and stored in the inventory. There are wide range of Units to choose from. AI which is spawned would have random resources and units. The goal of the player would be to build and win the war.

IX. FUTURE SCOPE

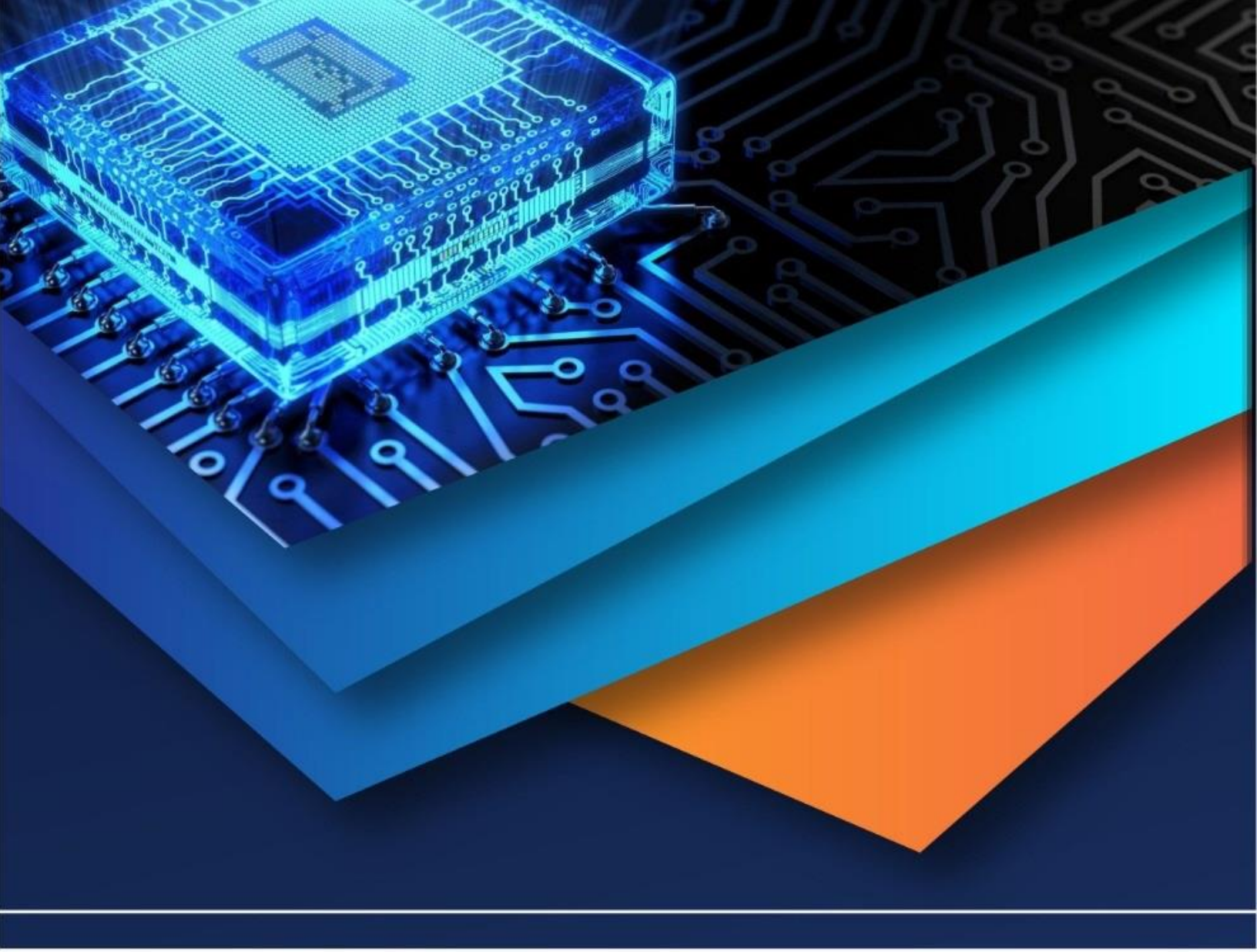
Further new game modes and compatibility to every device intended to be achieved once the game acquires a substantial player. Especially mobile devices as very less RTS game available for the players with mods support. Also, mods support, new maps and multiplayer would be introduced.

REFERENCES

[1] Christopher E. Miles, "Co-evolving Real-Time Strategy Game Players", ProQuest, pp. 7-10, USA, 2007.
[2] William Cassidy, "Utopia", GameSpy (Hall of Fame Article), USA, 2004
[3] Alan Satori-Angus, "Cosmic Conquest", Byte Magazine, USA, 1982.
[4] Dan Adams, "The State of the RTS", IGN Magazine, 2006.
[5] Lindsay Fleay, "The Historical RTS List", The RTSC Games List, USA, 2011.
[6] Matt Barton, "The History of Computer Role-Playing Games Part 2: The Golden Age (1985-1993)", pp. 3, USA, 2007.
[7] Dani Bunten Berry, "Game Design Memoir", USA, 2007.

[8] Kurt Kalata, “So What the Heck is Silver Ghost”, Hardcore Gaming 101, 2010.
[9] Levi Buchanan, “Top 10 Renovation Games”, IGN Magazine, USA, 2008.
[10] Anoop Gantayat “Sega Ages: Gain Ground”, IGN Magazine, 2004.
[11] Ian Erickson, “Herzog Zwei”, IGN Entertainment: Classic Gaming (Genesis - Game of the Week Article), USA, 2012.
[12] Scott Sharkey, “Hail to the Duke”, 1UP Magazine, USA, 2011.
[13] IGN, “Amiga Reviews: Battlemaster”, ZZap! Issue 68, pp. 45, USA, 1990.
[14] Bruce Geryk, “A History of Real-Time Strategy Games”, GameSpot, USA, 2006.
[15] Blizzard Entertainment, “StarCraft’s 10-Year Anniversary: A Retrospective”, Blizzard Entertainent, USA, 2008.
[16] Interactive Entertainment Today,“Top 100 Games”, Edge, USA, 2007.
[17] Boba Fatt, “The 52 Most Important Video Games”, GamePro, USA, 2007
[18] Andrew Park, “The Greatest Games of All Time”, GameSpot (PlanetScape: Torment), USA, 2005.
[19] Greg Kasavin, “StarCraft Strategy Guide: The Protoss Conclave – Units and Structures”, GameSpot, USA, 1998.
[20] Greg Kasavin, “StarCraft Strategy Guide: The Zerg Swarm – Units and Structures”, GameSpot, USA, 1998.
[21] Greg Kasavin, “StarCraft Strategy Guide: The Terran Dominion – Units and Structures”, GameSpot, USA, 1998.
[22] Blizzard Entertainment, “Blizzard Support: StarCraft”, Battle.Net, USA, 2008.
[23] Prima Fast Track Guide, “StarCraft – StarEdit Tutorial”, CreepColony, USA, 2008
[24] Age Of Empire , Ensemble Studios, Relic Entertainment , MacSoft

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)