

RP SERVER DEVELOPMENT

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

SAYAN PRAMANIK (Reg.No - 39110913)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade “A” by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Sayan Pramanik** (Reg.No.- 39110913) who carried out the Project Phase-2 entitled "**RP SERVER DEVELOPMENT**" under my supervision from January 2023 to April 2023.

Internal Guide

Dr. A. Mary Posonia, M.E., Ph.D.

Head of the Department

Dr. L. LAKSHMANAN, M.E., Ph.D.



Submitted for Viva voce Examination held on 11-04-2023

Internal Examiner

External Examiner

DECLARATION

I, **Sayan Pramanik** (Reg.No- 39110913), hereby declare that the Project Phase-2 Report entitled "**RP SERVER DEVELOPMENT**" done by me under the guidance of **Dr. A. Mary Posonia, M.E., Ph.D.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 11 April 2023

PLACE: Chennai

SIGNATURE OF THE CANDIDATE



ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph. D, Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.A.Mary Posonia M.E.,Ph.D**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-1 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

GTA V roleplay is a multiplayer mod for the PC version of Grand Theft Auto V. Players who jump into one of the user-created RP servers become even more immersed in the game as they take on the form of a normal NPC, which can be customized in a number of different ways.

The level of roleplay depends on the server. Some servers are completely hardcore roleplaying — which means players are skirting the rules if they ask for tips on how to access the game's menus or even the button that corresponds to an action. Others are a bit less strict.

RAGE Multiplayer is an initiative started by an experienced team to provide an alternative, enjoyable, and stable multiplayer experience for Grand Theft Auto V. Started in May 2016, RAGE continues to be a very delicate project with great potential noticed by many. Our sole focus is to mold this project in a self-sustaining community that promotes creativity and originality. Thanks to the C++ foundation, RAGE doesn't depend on any external files or programs, since it's built with C++.

RAGE is a complete, stand-alone client with great functionality. The team focuses on continuous performance and stability improvement while simultaneously adding new features and expanding the API.

TABLE OF CONTENTS

Chapter No	TITLE	Page No.
	ABSTRACT	v
	LIST OF FIGURES	vii
1	INTRODUCTION	1
2	LITERATURE SURVEY	7
	2.1 Inferences from Literature Survey	9
	2.2 Open problems in Existing System	11
3	REQUIREMENTS ANALYSIS	13
4	DESCRIPTION OF PROPOSED SYSTEM	14
	4.1 Selected Methodology or process model	17
	4.2 Architecture / Overall Design of Proposed System	15
	4.3 Project Management Plan	16
	4.4 Financial report on estimated costing	21
5	IMPLEMENTATION DETAILS	22
	5.1 Development and Deployment Setup	22
	5.2 Testing	23
6	RESULTS AND DISCUSSION	24
7	CONCLUSION	28
	7.1 Conclusion	28
	7.2 Future work	28
	7.3 Implementation Issues	28
	REFERENCES	30
	APPENDIX	31
	A. SOURCE CODE	31
	B. SCREENSHOTS	46
	C. RESEARCH PAPER	51

LIST OF FIGURES

FIGURE NO	FIGURE NAME	Page No.
1.1	File Contents with all server end files	1
2.1	Server Config File Settings	10
2.2	CMD logs	10
2.3	CMD Incoming connections	11
2.4	Client Database Data	11
2.5	Client Data getting Downloaded	12
4.1	Server Architecture	14
4.2	System architecture of Client Side	15
4.3	System architecture of Server Side	16
4.4	Login Page of the Client Side	17
4.5	Character Selection for the Client	17
4.6	Spawn of the Client in the game	18
4.7	Bag of the user	19
4.8	Admin Panel for the Administrator	19
4.9	Property Buy Page	20
4.10	Phone Page of the user	20
5.1	My SQL workbench	22
5.2	Vantage SQL	22
5.3	Vantage Logs SQL	22
5.4	Stream-Distance vs Server Response	23

LIST OF ABBREVIATION

RP	Role-Play
NPC	non-player characters
GTA	Grand Theft Auto
Dll	Dynamic Link Library
Json	JavaScript Object Notation

CHAPTER 1

INTRODUCTION

There are 2 main players on GTA-RP Servers one among them is Rage Multiplayer and FiveM. RAGE is a platform to host servers, it has no limits and gives you full control over everything whereas FiveM is also a platform to host servers but with control admin panel, It has multiple limitations which has been set to the server's host.

When downloading the RAGE:MP Client and running it, you will get the server files.

The folder name will be called **server-files**. Open the folder and you will find this

VantageRP			
	Name	Date modified	Type
access	.vs	01-06-2022 18:36	File folder
top	.vscode	09-06-2022 15:10	File folder
loads	bin	31-05-2022 10:58	File folder
ments	client_packages	20-10-2022 20:24	File folder
res	DataBase	07-06-2022 23:27	File folder
3ase	dotnet	31-05-2022 10:59	File folder
ps	logo	30-05-2022 20:04	File folder
eMode	logs	03-06-2022 11:14	File folder
igeRP	maps	23-11-2021 21:22	File folder
ive	packages	03-12-2021 17:12	File folder
	plugins	23-10-2021 20:45	File folder
	BugTrap-x64.dll	28-04-2021 04:34	Application exten...
bjects	conf	20-10-2022 21:20	JSON File
top	coords	02-06-2022 19:48	Text Document
ments	debug	28-04-2021 04:34	Windows Batch File
loads	ragemp-server	08-06-2021 22:37	Application
	readme	14-05-2022 23:12	Text Document

Fig 1.1. File Contents with all server end files

From reference [3] we got to know that client packages is a folder in which client-side scripts are located and downloaded by the client. (Client-side allows you to draw GUI or do CEF work). Packages is a folder in which server-side scripts are located and loaded by the server. (Server-side allows you to manage the player and create your own world). Map is a folder in which map files (in JSON format) are loaded into the game world. Plugins is a folder which allows you to load custom .dll plugins into the server. conf.json contains the server configuration in JSON format. Read more about server settings here. Ragemp-server.exe is a server executable, which if you run, starts the game server.

If you're new to all this, it can prove to be quite daunting. You need to download the required mods to make it work, find and apply to join servers, and then figure out how to properly play the thing itself. On top of all that, you'll also come across a bunch of slang terms you may be unfamiliar with. To try and alleviate some of that stress, we've put together a handy starters guide to help anyone wanting to join in.

RP stands for role-play. In terms of GTA RP is completely different than standard GTA. You are suppose to create a character, fictional, and play out the character you create. It helps to think of RP as an alternate universe where everything is real. So using words like server, admins ect is frowned upon and considered immersion breaking.

My best suggestion is to watch some streams of people role playing, there are lots. If you decide to join a RP server make sure to read the rules. Rules are taken very seriously and the inability to follow rules, especially if the cause is not reading the rules in the first place, is cause for a swift ban.

A Grand Theft Auto Roleplay server takes the regular crime inspired personality of the GTA franchise and turns it into a clean slate. Thanks to the installation of a mod, you can spawn into a map populated with dozens of others and play as a cop, shop clerk, or even a criminal.

Most servers host a wide array of options and don't abide by a specific theme, per se. Typically, each site has their own website where they set out which specific mods to download to join, alongside various forums where you can chat to people who have taken up a similar profession to you.

Some servers are more hardcore than others regarding the amount of immersion they expect you to provide, but most ask you not to break character or engage in senseless violence. In fact, if you do, it's a good way to earn the ire of those roleplaying as cops.

Joining and maintaining your status on a GTA RP server takes some work. Most servers will ask you to fill out an application and read a set of rules – if you'd like an example, TwitchRP's ruleset is just a click away. Once your application has been accepted you'll join a queue where you can ascend the priority based on well you follow the rules.

You'll have to jump through some hoops, but it seems to be a necessary step to stop each server from crashing, or from letting anyone in who would ruin the fun.

Each server works to a similar cadence and aims to offer the most flexible experience possible by allowing you to play as anything from a cab driver to a news anchor. Because of that, a lot of these servers can be quite similar but they can differ in scale and accessibility. Because of that, we've sifted through a few to get you started.

NoPixel This is one of the most popular servers you could hope to join, but it's also the hardest. NoPixel only supports 32 players at a time and is frequented by Lirik, Moonmoon, and Summit1G. It's no surprise, then, that it had to suspend applications as over 4,000 people applied.

That said, applications won't be down forever and you could still be in a chance with roleplaying with some of Twitch's biggest streamers – especially if you can roleplay quite well.

Eclipse RP Another one of GTA RP's heavy hitters, this server hosts 200 players at one given time. One of its more impressive features is how detailed its forum section is – clearly laying out oodles of crime factions for you to join – ranging from the Los Santos Irish Mob to The Clowns. There's also post spots for specific jobs like Los Santos Customs and Los Santos Medical Services.

Do be warned though, Eclipse RP appears to be another one of GTA RP's premier servers, you're time on the waiting list may prove lengthy. GTA World This server stands out as it's entirely text-based, meaning those of us who aren't up to chatting with strangers have a place to go to give GTA RP a whirl. Much like the others, there's a sign-up process for GTA World, and according to the website there are currently 500 members.

Even if you're not an accepted member though, you can noodle around the forums and see what kind of factions are in play.

Mafia City This roleplay server is two years in the making and it's now available for signups. It comes with a "dynamic" legal and illegal group system, customisable player business and corporations, and every property is for sale, too. Apartments, houses,

hotels, and condos – you name it. Mafia City also gives you the option to play as a cop, gangster, or a money hungry CEO.

GTA V roleplay is a multiplayer mod for the PC version of Grand Theft Auto V. Players who jump into one of the user-created RP servers become even more immersed in the game as they take on the form of a normal NPC, which can be customized in a number of different ways. The level of roleplay depends on the server. Some servers are completely hardcore roleplaying — which means players are skirting the rules if they ask for tips on how to access the game's menus or even the button that corresponds to an action. Others are a bit less strict.

Anyone with a copy of GTA V on PC can play GTA V roleplay, but it is significantly different from the normal multiplayer mode that is built into Rockstar's hugely popular title.

You also need access to the modified game client, like FiveM or Grand Theft Multiplayer, to get the mod. Once there, you can join an RP server.

Some communities also require an invite — which players gain by proving themselves as a talented roleplayer (or a big name in the streaming space). Typically, you'll have to play on a prove yourself server in this case, but once you shine through, you'll get fast-tracked to the big leagues.

The world of Grand Theft Auto Online is no longer one multiplayer game mode. It's become an amorphous platform that players are constantly recontextualizing. The development and proliferation of Grand Theft Auto Roleplay (GTARP) gives players a place to join together in creating their own living, breathing cities. The players forge alternate lives online, pursuing jobs and establishing relationships — all with the goal of creating an environment for their role-playing.

Cone joined the “Legacy RP” server in February 2018 and was immediately hooked. “I’ve been consistently role-playing since I was 7 years old,” he says. “I’ve played every [pen and paper] game at this point.” He joined the Legacy RP police force, and quickly dedicated himself to the RP like a full-time job. Cone, a student with aspirations of becoming a full-time streamer, worked eight- to 12-hour shifts five times per week, handing out traffic citations, trying to arrest drug runners and occasionally being called to armed robberies.

The dedication led to a promotion, and Cone became a junior detective. The switch

came with a new host of challenges. The system wasn't built for evidence gathering and case building, which forced him to use an imperfect system of collaborating with other players, rather than traditional Grand Theft Auto mechanics, to play out intense criminal investigations.

GTARP was born out of the popular FiveM mod, a community-developed project that launched in 2015 to improve the established GTA Online experience. The mod gives role-players tools so they can blend traditional third-person action gameplay with a pseudo-Second Life experience — transporting them into a setting where they can be anyone and do anything, so long as they obey a series of guidelines called the "Federal Law" rules, designed to protect the RP.

Open-world games traditionally have overarching goals or win conditions to reward players for their skill or dedication. We're conditioned through multiplayer, sometimes even rewarded, to seek out a player who got the kill and get revenge. GTARP defies all these conventions. Ideally, nobody should try to "win"; instead, the goal is to appreciate the journey and build stories for the server. Other players govern the reward, not the game. Was the RP good? Is there now a storyline others can share in? Did everyone feel it was fair?

The concept of role-playing inside a multiplayer environment is nothing new. Its history can be traced to Ultima Online and takes a path through Ever Quest and World of Warcraft — but GTARP is real-world role-playing rarely seen in video games, with a depth that can become intoxicating thanks to how much it mimics real life.

Players drop into Los Santos with nothing. There's usually a police officer who meets a new player and guides them through the character creation process. They spawn at the job center, where their journey begins. Every new player begins with the same default, bald-headed avatar — which has led to regular players calling them "cue balls."

Cone arrives on the scene and assists. He uses a series of code words designed to keep the immersion. "Hold your N muscle for me," he says, trying to get a new player to press the N key to use voice chat. "Now, if your brain was a computer, you might press your escape key and check your audio settings." The player is responsive, role-playing immediately: "I just want to get a car, officer." Cone gives them \$5,000 out of his own pocket. It's enough to buy a basic car in the game. "It's not much, but it'll at least get you from Point A to Point B while you get settled," he says.

"That guy seemed really into it," Cone says on his stream. "I think he's going to be a good addition to the city."

As one of the most iconic streamers in the world, Ninja knows a thing or two about what makes a streaming platform great. He exploded in popularity during the Fortnite craze on Twitch, then signed a mega deal with Microsoft's now-defunct Mixer before moving back to Twitch exclusively for a while. Now, he streams anywhere he pleases.

Kick is the latest player in the streaming game, and although Ninja hasn't expressed any desire to stream on there yet, he believes it could actually succeed where Mixer eventually failed because it's less prone to bureaucracy, which, in turn, means important decisions can be made post-haste.

"The main issue with Mixer compared to Kick, and it's abundantly clear, is that it took too long to get things done," said Ninja. "There were like 80 billion different people that had to reach out to somebody else, who then had to reach out to somebody else, who had to reach out to somebody at Microsoft, who had to reach out to somebody up top at Microsoft even more to get confirmation to change anything."

CHAPTER 2

LITERATURE SURVEY

GTA V roleplay is a multiplayer mod for the PC version of Grand Theft Auto V. Players who jump into one of the user-created RP servers become even more immersed in the game as they take on the form of a normal NPC, which can be customized in a number of different ways. The level of roleplay depends on the server.

Rage Multiplayer is a modification for a popular open world game set in a fictional city, which is based on Los Angeles. As an Open World game, the player is free to do in the world as he likes. The Rage MP multiplayer mod allows you to play multiplayer on your own servers. Rage MP is a multiplayer mod based on servers hosted by the players themselves. What's special: Rage MP allows an unprecedented number of players. Via ZAP-Hosting you can rent servers with up to 1000 slots. Despite this high number of players, Rage MP runs with high performance and without FPS loss. In addition, Rage MP comes with its own API and is of course fully modifiable. A game like Rage MP is of course the most fun with your own friends or in your own clan. You can do what you want on your own server and leave it as it is. And of course, you can also make your own RP server with it. At ZAP-Hosting you can now also rent Rage MP servers. All our servers are of course fully DDoS protected, easy to use via our web interface and offer full mod support thanks to free FTP access.

From reference [4] encountered this problem, whenever I try to join server it says unknown file version, i have tried running updater.exe, I have easy anti-cheat installed, my drivers are up to date, also windows, I updated just minutes ago, i tried reinstalling Rage:MP, restarting pc several times, also I tried verifying integrity of game files, but it's not working, also seems like many of Rage:MP users encountered this problem cause many people are asking for fix.

Twitch serves as a conduit for viewers to get involved in the stories of GTARP. Servers are available as search terms, allowing a network like access to the show. Sometimes as many as 10 different streams create a collage of easily accessible stories, all operating in a communal space. Viewers are left with an occasionally cheesy but often charming soap opera that exists online 24 hours a day.

Legacy RP can only accommodate 32 players at a time but feels larger when so many people are role-playing in one space. It can be addictive to watch and can be even more difficult for players who become too invested in their role-playing.

Cone experienced this kind of crossover. After a week of intense RP, in which he spent 72 hours streaming, the line between life and character melted away. “I had to take a break,” Cone says. “I started to act like Chris the character, think like Chris the character. I couldn’t disconnect.” He found himself driving around his real-life town and questioning whether he needed to stop at red lights, or could roll through like in the game. Cone found himself missing turns because he was thinking too much about the RP.

Players follow a list of rules called “Federal Law,” which govern the separation between game and out-of-character (OOC) interactions. The section on how players must act when they are shot or incapacitated is very clear: If a character gets knocked out by blunt force, they are able to remember everything leading up to the moment they went down, but if they are shot, they must role-play as if everything 20 minutes prior to their death was lost to amnesia. The players role-playing as Cone and Renee both knew The Condemned ambushed them. They saw their motorcycles and signature red-and-black jackets, but neither player was allowed to remember any of it. As far as the characters knew, it was a random unprovoked attack, and nothing more.

The line Cone needed to walk, especially as a detective, was razor-thin. He was role-playing out the investigation of his own shooting, knowing full well who did it, but needing to act as if he had no clue. Every action he took had to make sense in the fiction. That led to tension a few hours after the shooting, when Cone joined several players — including members of The Condemned — at the city’s square, which operates as a social gathering hub. Killed and killer, making small talk about cars and the weather, both sides waiting to see if the other broke.

Cone had a clear path of investigation that made sense inside the role-play, but no way to execute it. For weeks the trail went dead, and Cone was ready to give up on the storyline. Then he got a message.

Each GTA roleplay server is different, but some are more strict than others. Generally, servers ask that you play by the rules and stick to your assigned role. But some servers are extra strict and ask that you never break character, which means even trying to ask for help from another roleplayer with something like controls has to be integrated into the storyline somehow.

When it comes to the art of roleplaying, there’s no time to think up a sentence, like WoW. With GTA roleplay, you have to take advantage of your acting skills, use your microphone, go with the flow, and improvise.

It's an experience that can be enjoyed alone, but it's much more entertaining and engaging with a couple of friends or even some new strangers by your side. It's a blast, and if you happen to own a copy of GTA V, it's well worth playing.

You could spawn into the server and take a bold path as a criminal mastermind or leader of a notorious gang. But you could just as easily become something as mundane as a chauffeur, singer, cop, restaurant worker, or milkman. The only limit when choosing what path to take your roleplaying in is your imagination.

Each GTA roleplay server is different, but some are more strict than others. Generally, servers ask that you play by the rules and stick to your assigned role. But some servers are extra strict and ask that you never break character, which means even trying to ask for help from another roleplayer with something like controls has to be integrated into the storyline somehow.

2.1 INFERENCES FROM LITREATURE SURVEY

There are multiple servers on Roleplay for GTAV. This problem to handle over 2000 players simultaneously is being solved by this project. Due to less availability of servers and constrain system requirements the server goes into queue and players needs to wait for join the server. hosting Cloud server and linking to Rage Multiplayer. It requires Connection to Server over a public port with various port forwarding. In figure 2.2 it shows all the files are ready and players ready to join into the server.

While there are many options for GTA roleplay servers, the NoPixel server is especially popular at the moment because it is a server where massive streaming stars regularly come together for intertwined storylines.

The NoPixel server is difficult to get into because of its massive popularity. But those in the GTA community can still enjoy watching some of the biggest names in streaming interact as unique characters on the server.

It also interacts with the database every 3min to Update the database.

```

1  {
2      "maxplayers": 5000,
3      "name": "Vantage RolePlay | [roleplay][voice] [Local]",
4      "gamemode": "roleplay",
5      "language": "eng",
6      "stream-distance": 350.0,
7      "announce": false,
8      "enable-nodejs": true,
9      "csharp": "enable",
10     "bind": "127.0.0.1",
11     "port": 22005,
12     "voice-chat": true,
13     "voice-chat-sample-rate": 48000,
14     "api-threading-debugging": false
15 }

```

Fig 2.1. Server Config File Settings

```

21:14:04.181 [Succ] | HouseFurniture => Furnitures has been saved to DB
21:14:04.182 [Info] | nInventory => Saving items...
21:14:04.193 [Succ] | nInventory => Items has been saved to DB.
21:14:04.284 [21:14:04.285 [21:14:04.287 [21:14:04.288 [SuccSuccSucc21:14:04.294 [Succ] | Stocks => Stocks has been saved to DB
] | Stocks => Stocks has been saved to DB
] | Stocks => Stocks has been saved to DB
[Succ] | Stocks => Stocks has been saved to DB
] | Stocks => Stocks has been saved to DB
21:15:42.207 [Info] | GM => Saving Database...
21:15:42.438 [Succ] | BusinessManager => Businesses has been saved to DB
21:15:42.468 [Succ] | GangCapture => Gang Regions has been saved to DB
21:15:42.712 [Info] | EatManager => Check Water.
21:15:44.505 [Succ] | HouseManager => Houses has been saved to DB
21:15:44.775 [Succ] | HouseFurniture => Furnitures has been saved to DB
21:15:44.775 [Info] | nInventory => Saving items...
21:15:44.777 [Succ] | nInventory => Items has been saved to DB.
21:15:44.787 [Succ] | Stocks => Stocks has been saved to DB
21:18:22.498 [Info] | EatManager => Check Eat.
21:18:42.242 [Info] | GM => Saving Database...
21:18:42.454 [Succ] | BusinessManager => Businesses has been saved to DB
21:18:42.485 [Succ] | GangCapture => Gang Regions has been saved to DB
21:18:44.529 [Succ] | HouseManager => Houses has been saved to DB
21:18:44.869 [Succ] | HouseFurniture => Furnitures has been saved to DB
21:18:44.873 [Info] | nInventory => Saving items...
21:18:44.876 [Succ] | nInventory => Items has been saved to DB.
21:18:44.888 [Succ] | Stocks => Stocks has been saved to DB

```

Fig 2.2. CMD logs

If a client joins a server admin can see the client IP address as this prevents user from multi accounting and allow us to make a prevention layer before an attack happens.

```

21:30:42.780 [Succ] | BUSINESSManager => BUSINESSES has been saved to DB
21:30:42.815 [Succ] | GangCapture => Gang Regions has been saved to DB
21:30:44.590 [Succ] | HouseManager => Houses has been saved to DB
21:30:44.843 [Succ] | HouseFurniture => Furnitures has been saved to DB
21:30:44.844 [Info] | nInventory => Saving items...
21:30:44.846 [Succ] | nInventory => Items has been saved to DB.
21:30:44.855 [Succ] | Stocks => Stocks has been saved to DB
21:33:42.584 [Info] | GM => Saving Database...
21:33:42.791 [Succ] | BusinessManager => Businesses has been saved to DB
21:33:42.821 [Succ] | GangCapture => Gang Regions has been saved to DB
21:33:42.886 [Info] | EatManager => Check Water.
21:33:44.579 [Succ] | HouseManager => Houses has been saved to DB
21:33:44.826 [Succ] | HouseFurniture => Furnitures has been saved to DB
21:33:44.827 [Info] | nInventory => Saving items...
21:33:44.828 [Succ] | nInventory => Items has been saved to DB.
21:33:44.841 [Succ] | Stocks => Stocks has been saved to DB
21:35:22.575 [Info] | EatManager => Check Eat.
[N] Incoming connection from 100.96.151.56
21:36:42.622 [Info] | GM => Saving Database...
21:36:42.827 [Succ] | BusinessManager => Businesses has been saved to DB
21:36:42.875 [Succ] | GangCapture => Gang Regions has been saved to DB
21:36:44.651 [Succ] | HouseManager => Houses has been saved to DB
21:36:44.907 [Succ] | HouseFurniture => Furnitures has been saved to DB
21:36:44.909 [Info] | nInventory => Saving items...
21:36:44.920 [Succ] | nInventory => Items has been saved to DB.
21:36:44.939 [Succ] | Stocks => Stocks has been saved to DB

```

Fig 2.3. CMD Incoming connections

accounts @vantage (vantage) - Table							
Objects	Text	Filter	Sort	Import	Export	Data Generation	Create Chart
socialclub	sayan	B67E61247ECA4D584D8697E493000	100.115.240.128	333333	-1	-2	sayan@gmail.com
TGBUTAL	e_rror_06	EDC809D0BAE8E1D85E9CD1199999998	100.77.194.66	506816	-1	-2	ksrp@gmail.com
Slayer5263	surojit	B67E61247ECA4D584D8697E0	100.96.151.56	-1	-1	-2	Surojit@gmail.com

Fig 2.4. Client Database Data

Figure 2.4 Shows the database with data where the user details are saved, it also contains the details of the cars, bag items, user played time, transaction made by the user, etc...

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

There are multiple servers on Roleplay for GTAV This problem to handle over 2000 players simultaneously is being solved by this project. Due to less availability of servers and constrain system requirements the server goes into queue and players needs to wait for join the server. Hosting Cloud server and linking to Rage Multiplayer. Managing

DLC and linking to database, providing customization to cars from client side Managing different types of DLCs Managing users' data and validating with Rockstar Game Services.

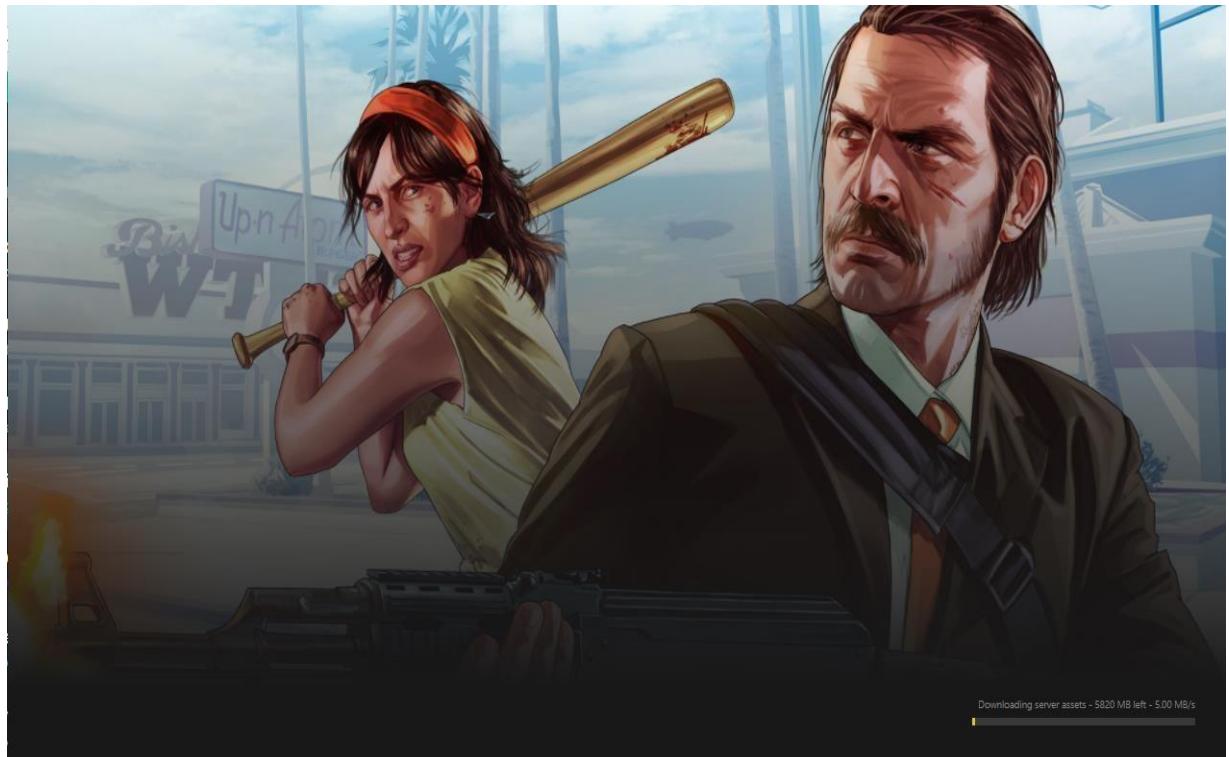


Fig 2.5. Client Data getting Downloaded

Figure 2.5 shows the attempt of user to connect and the download starts, this download speed depends on the host internet speed and client internet speed.

These include all the client packages required for the user to download in order to connect to the server, without downloading the server assets the client cannot connect to the server.

CHAPTER 3

REQUIREMENT ANALYSIS

Operating System is needed in such a way that the stem can be used to handle and run database with no error and limited bloatware which can be achieved by using Windows Server 2022. The tools needed to be installed in the OS are VS code to debug the code at any moment of fixing of code, The database has to be run on the database and manage by using MySQL Server along with this building the file on the specific OS it also needed a software called Visual Studio Enterprise.

The language needed to run and for debugging the code are C#, HTML, CSS, JavaScript. Running all this in a PC or a cloud host needed a PC with Dedicated SSDs which can store huge Data for the database and to create a backup at any point of time.

Running the server also needed a strong CPU of processing speed above 4GHz, without saying it also needed a strong internet connection speed with would let users to download data from the server at huge speed hence the seed should be above 100GBps which can be achieved on cloud computing via VPS.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

In the below mentioned Figure 4.1 we could see the system architecture of the process where the user makes an interaction with the Server after getting verified by the firewall it can be understood by reading the reference [1]. Admin has got the full control over the system.

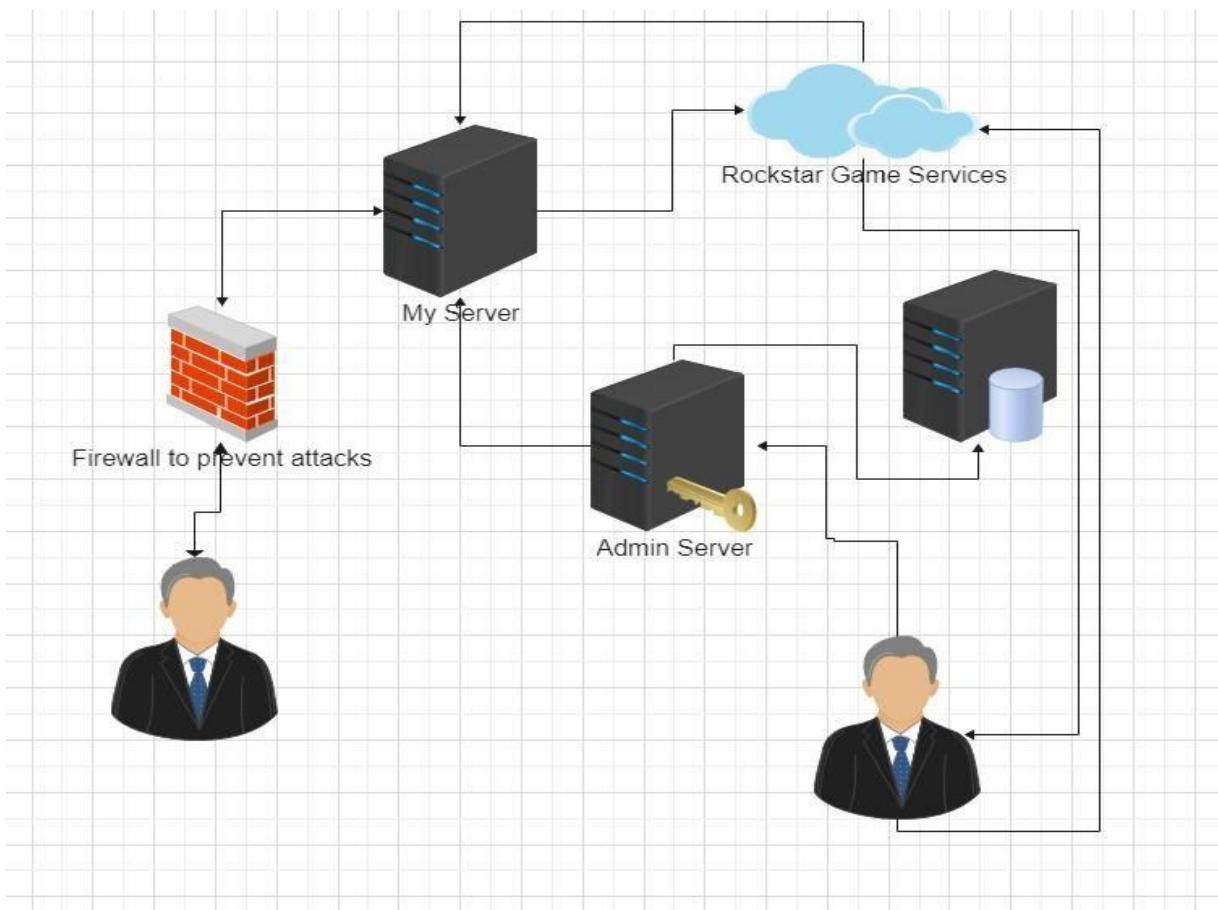


Fig 4.1. System Architecture

4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM

The block diagram of the proposed system has been shown in the above figures. The camera Here the user gets into the server. It's a Front-end Ideation Map. Once the player is spawning the activities of the player changed in respected to every player

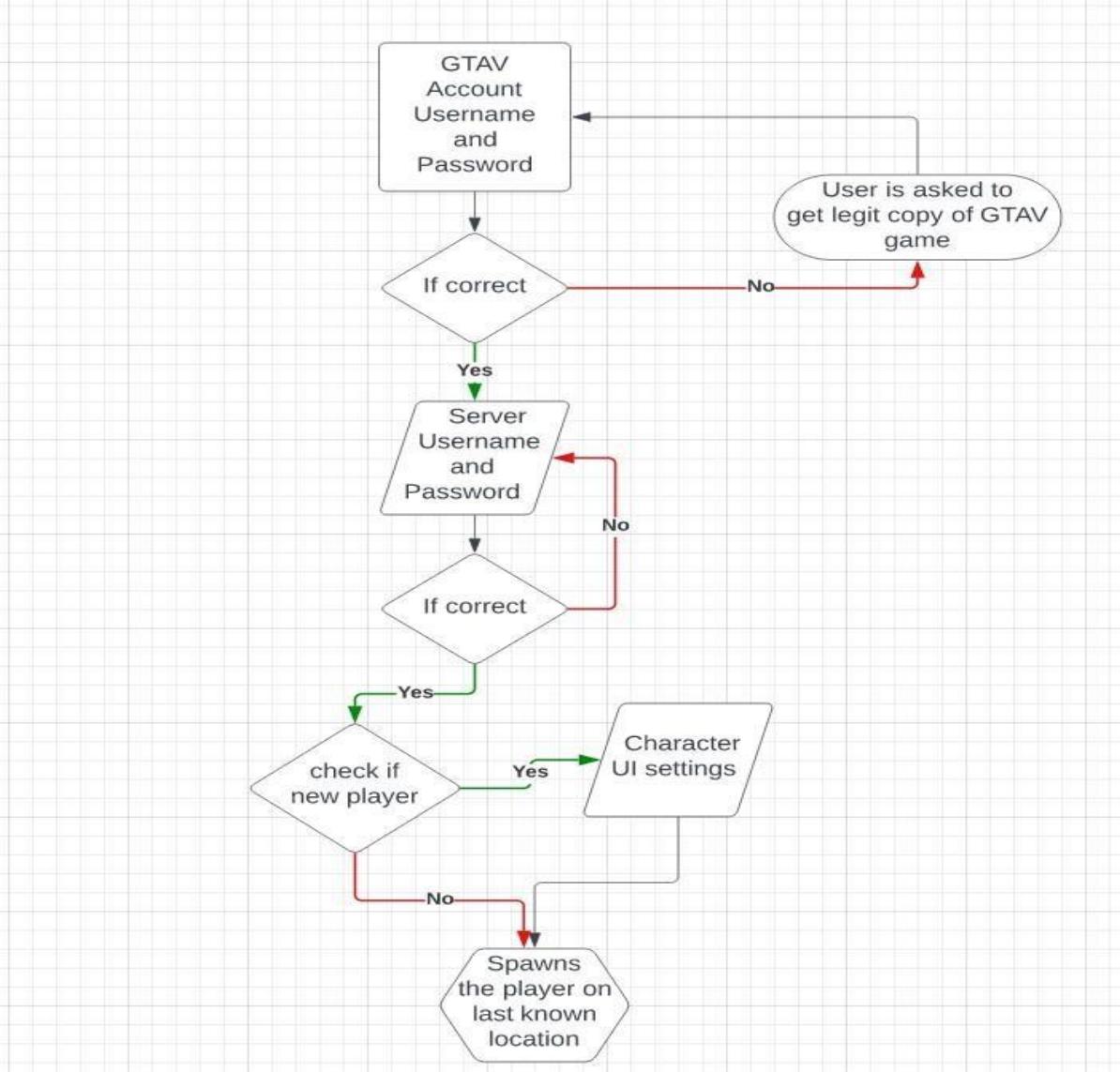


Fig 4.2. System architecture of Client Side

This is the ideation map for the backend. Here the player activities are all saved into database simultaneously Admin have different levels of access. Higher level admin has access to change in database directly.

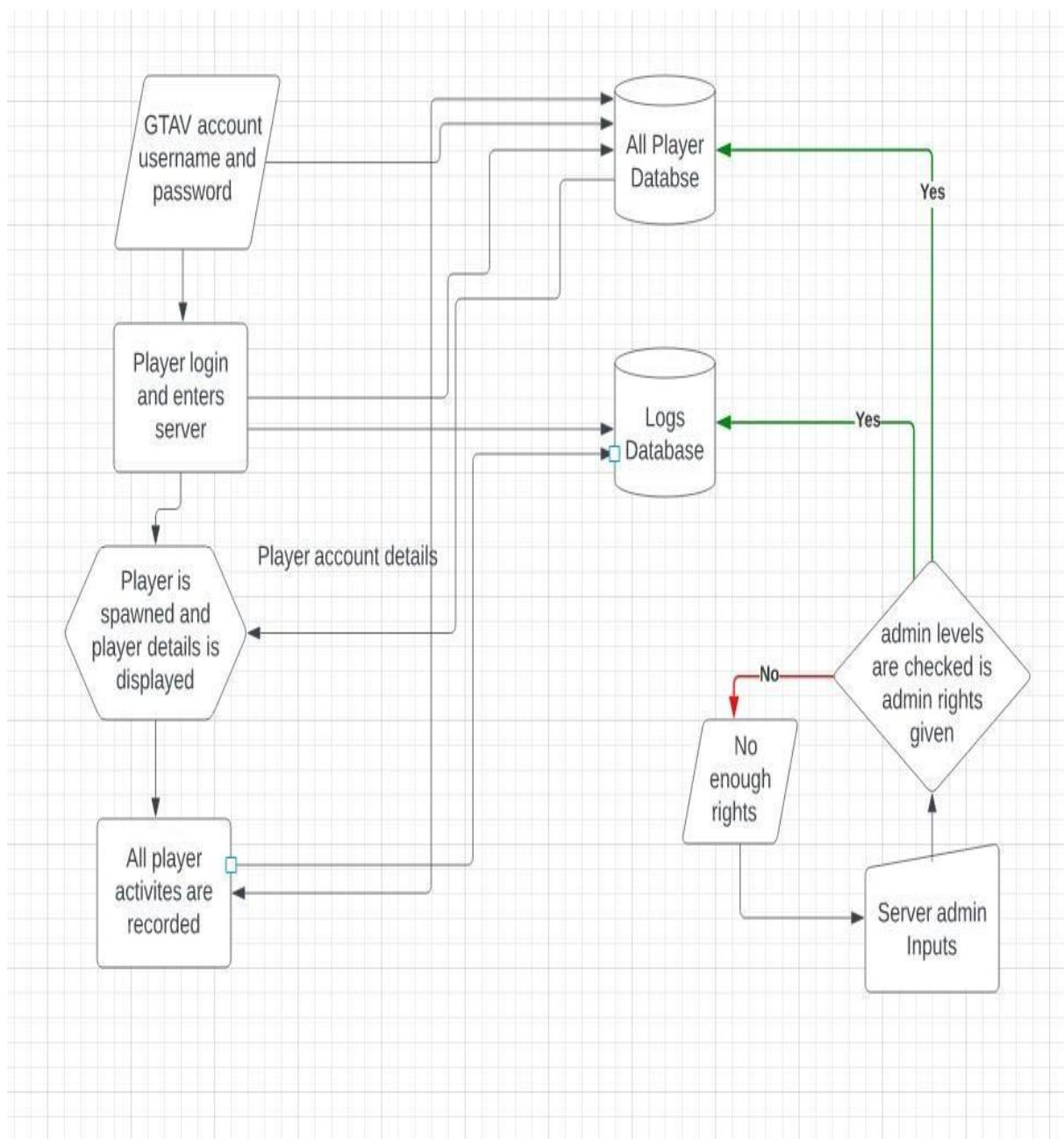


Fig 4.3. System architecture of Server Side

4.3 Project Management Plan

The picture below figure 4.4 shows the attempt of the user to login with the credentials. These helps the data of the client to be individually separated from the other client data

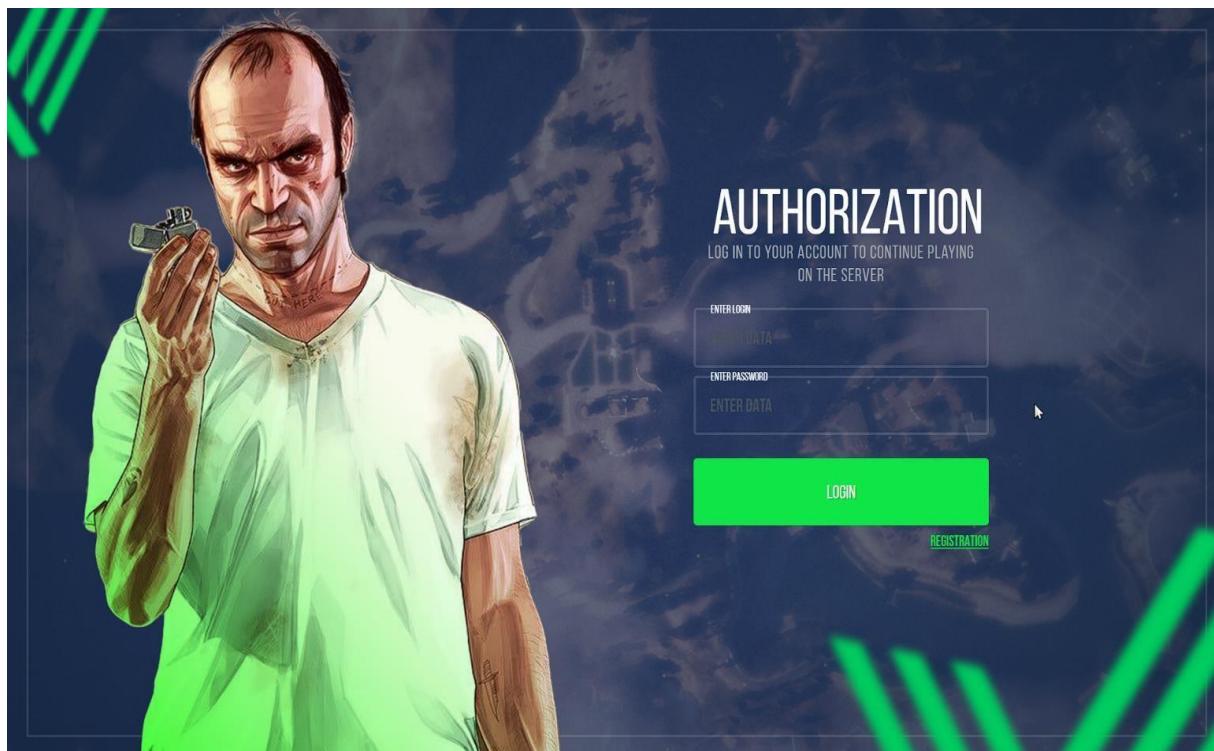


Fig 4.4. Login page of client Side

In the picture below figure 4.5 shows the user is given the option to create 3 characters for a single credentials which can prevent client form multi accounting

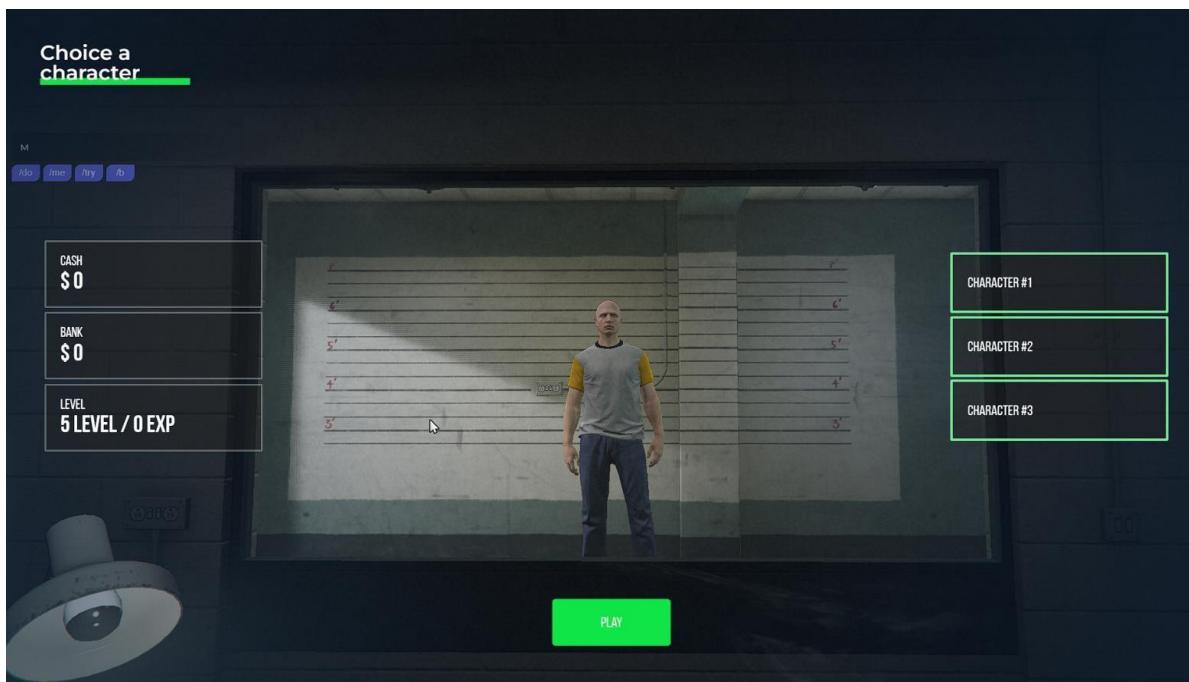


Fig 4.5. Character Selection for the client

Figure 4.6 shows the after-login page which makes the character spawn in the virtual

world, the figure shows the UI of the game, On the Top right it shows the server logo along with the server time Below it shows the money with the client in cash and in bank Below the money part there is a small time which reset every day with gives the player bonus for playing on the server for 5hrs Below the bonus section there is a small section which shows the key binds of the game which are needed more often. We could also see a green zone icon the right bottom which shows all illegal activities in game is prohibited only legal org members can perform the action as its near gov buildings. Left bottom show the map street name and a mic button which is off by default.



Fig 4.6. Spawn of the client in the game

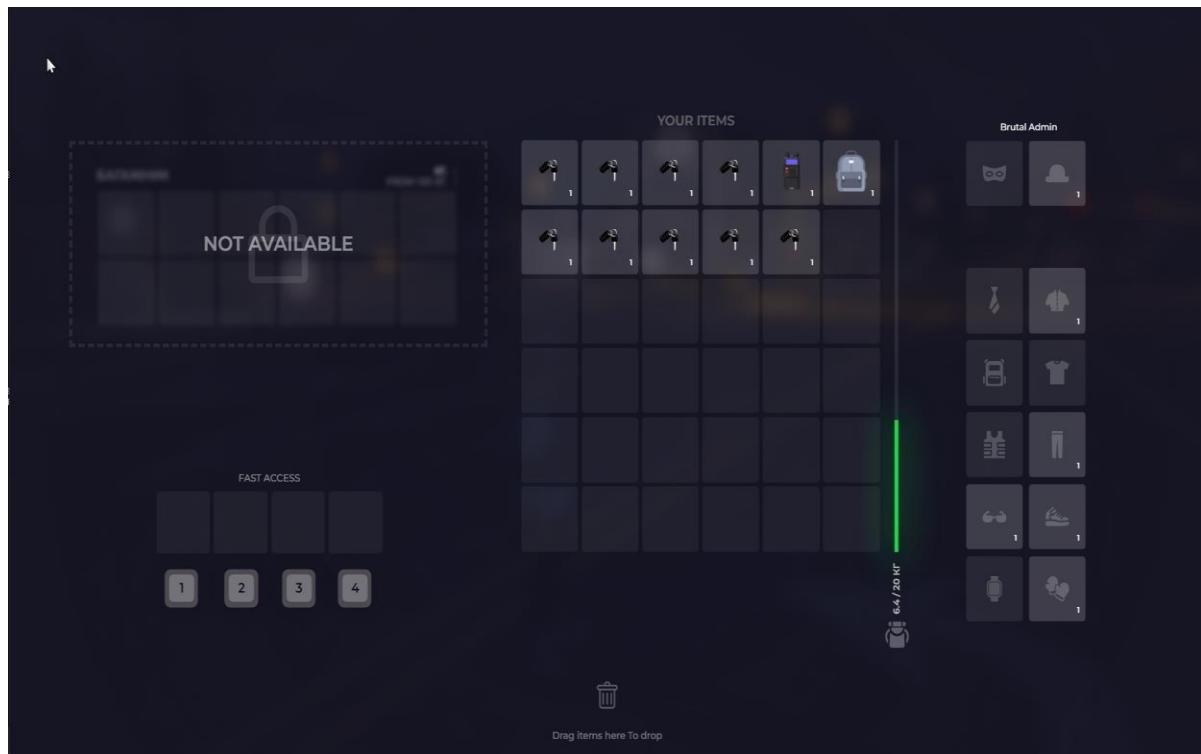


Fig 4.7. Bag of the user

Figure 4.7 shows the inventory of the client where the utility is saved. These can be modified by the user in the character, items in the bag cannot be modified by other users.

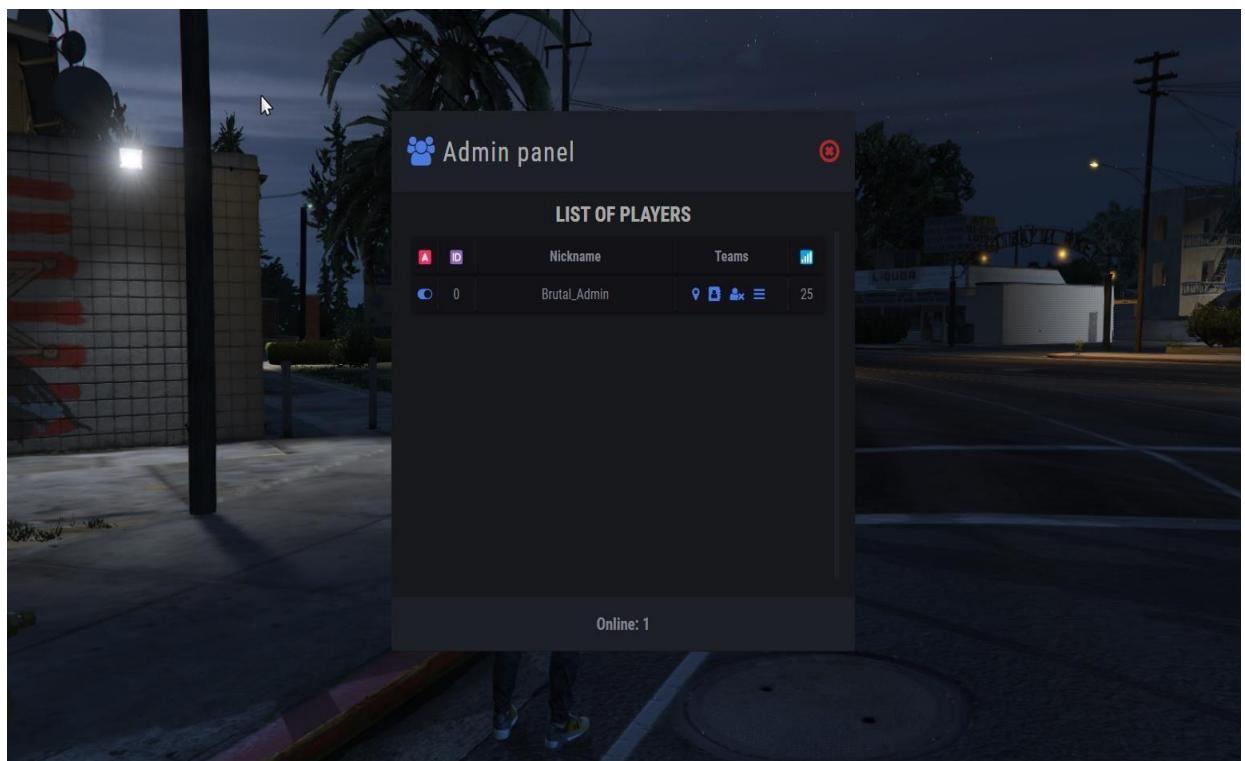


Fig 4.8. Admin panel for the administrator

Figure 4.8 There is also an Admin panel for the admins of the game which allows the admin for the smooth control of the game.



Fig 4.9. Property Buy Page

Figure 4.9 shows that the House can also be bought where the player can stay and clint has to remember to pay their rentals.



Fig 4.10. Phone page of the user

Figure 4.10 show the Phone menu which can be used to interact with other players on the server, navigate, creating advertisement, and booking taxi and performing other jobs. Every user has their own phone to setup their own contacts.

4.4 Financial report on estimated costing

The cost of the project is directly depended on the Developers, cyber security team, and the cost to host the server. Developers are needed to make the game environment bug free and to manage the server with admin rights. The role of cybersecurity team involves protecting the server from getting attacks like DDOS, Exploitation of bugs, etc. the final includes the role of hosting the server, it can be done in 2 means one by hosting it on a Cloud machine and another by hosting on the physical system and with a decent internet connection (2 network providers at least to prevent downtime).

Cloud server cost can vary from \$0.03398 / vCPU hour to \$0.09998 / vCPU hour. Running the system on a physical system includes electricity cost, Internet charges, and system maintenance cost.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

These server files and client codes are all written in C# language. The server is first run over the Rage multiplayer service, which is responsible for hosting the file. For deployment, first the Database server must be setup with 2 schemas named as vantage and the other for logs as vantage logs.



Fig 5.1. My SQL workbench

Vantage includes all the server assets like cars dictionary, house location, shop location, NPC location, user account details, etc. where as vantage logs include the daily logs of all the players like what they bought in game, the game currency they purchase, cars they bought at a instance, market value, etc.

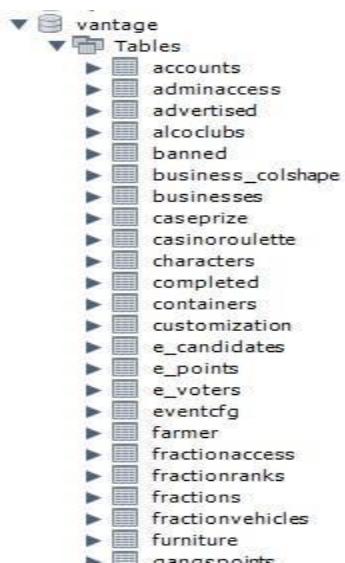


Fig 5.2. Vantage SQL



Fig 5.3. Vantagelogs SQL

5.2 TESTING

This server must be tested with overall a crowd of over 50 players, as it shows the stability of the server, server interaction and database update rate. These 50 players concurrently moving on the map helps in noticing the server stability and render rate which leads us to the graph of stream distance to server response time.

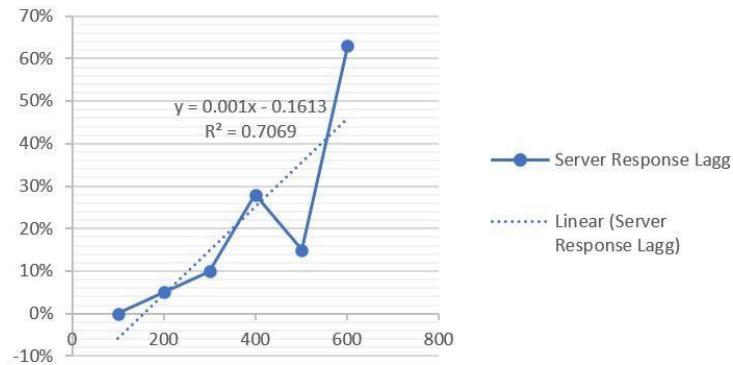


Fig.5.4. Stream-Distance vs Server Response

Server is also tested to have an ability to create 3 accounts at a time and the ability to perform role play with all having microphone open to have the facility to have VOIP.

CHAPTER 6

RESULTS AND DISCUSSION

As the result of hosting the server, player where able to join the server and communicate with voice chat and RP chats section in game which enabled the player to interact with other player and perform all activities available in game like buying cars, selling them to another user, putting on a number plate on the vehicle, and the most important save there belongs in their car and in their house warehouse. Players were also able to interact with their in-game bank account to save their in-game currency and redeem their red coins which can particularly be added in game by the purchase of real time money.

Players can also interact in their fraction with their other employees in their fraction, talking about fraction it includes Emergency medical services, Sheriffs, Gangs which are four gangs here, FIB, Government, and the jail administrator.

All levels organization people where able to perform their jobs like spawning their org cars and sheriff performing radio calls for their org alone, etc.

In your game mode folder, create an `index.js` file. After that, open the file and write the following:

```
//init Commands file
require('./commands.js')
```

Now save and close the file, then create another file named `commands.js` and open it.

Add these simple commands to your file:

```
mp.events.addCommand('hp', (player) => {
    player.health = 100;
});

mp.events.addCommand('armor', (player) => {
    player.armour = 100;
});
```

```
mp.events.addCommand('kill', (player) => {
    player.health = 0;
});
```

Now we have created a simple source with 3 basic commands. One that gives health, one that gives armor, and one that kills you.

Now save the file and close it.

Have you noticed anything wrong with the code we just wrote? If not, let me give you a hint. If you trigger the kill command, this means it will kill the player, but will you re-spawn back? I have the solution for you!

Create a **JSON** file and name it `spawn_points.json` and then add the following code:

```
{
    "SpawnPoints": [
        { "x": -425.517, "y": 1123.620, "z": 325.8544 },
        { "x": -415.777, "y": 1168.791, "z": 325.854 },
        { "x": -432.534, "y": 1157.461, "z": 325.854 },
        { "x": -401.850, "y": 1149.482, "z": 325.854 }
    ]
}
```

Done? Now save it and close.

Create a new file named `events.js` and add the following code:

```
let spawnPoints = require('./spawn_points.json').SpawnPoints;

mp.events.add('playerDeath', (player) => {
    player.spawn(spawnPoints[Math.floor(Math.random() * spawnPoints.length)]);
    player.health = 100;
});
```

We have defined our spawn points that we want the player to spawn in, and then we added an event for when the player dies. When this event is triggered, they re-spawn in a random location from the spawn points you defined. Also do not forget to restore the player's health 100 again or they can teleport while dead to the spawn point.

In the end, your `index.js` file should look this:

```
//init Commands file
require('./commands.js')
```

```
//init Events file
require('./events.js')
```

6.1 Scripted Jobs

Below is the list of few of jobs available inside the game and people or players can perform these jobs into the game so that they can earn the in game currencies and also leave the livelihood in game

- Store cashier
- Lawyer
- Government worker
- Police officer
- Fireman
- Trucker
- Farmer
- Metallurgist
- Mechanic
- Taxi driver
- Car thief
- Strict & realistic police department
- Fishing
- Coroners
- Sheriff
- County jail guard

- Brewers

6.2 Business system

Below is the list of all the business available in the games so that appear person or a player can perform in the game so that they can earn huge money and other player can interact with this business and the business owner would earn from it so that the economy of the server maintains.

- 24/7
- Ammunition
- Cinema
- Barbershop
- Clothing Store
- Plastic Surgery Clinic
- Tattoo Parlour
- Gas Station
- Mask Shop
- Electronic Shop
- Factory
- Garage / Chop Shop
- Restaurant
- Bar
- Night club
- Phone shop
- Liquor store
- Pawnshops

6.3 Property system

More than 5000 houses are available for players to own, with a garage system linked to your house and more than 300 interiors to choose from. Our furniture system allows you to completely create your own interior from A to Z, but you can also decide to use a premade interior and make some minor changes, and exterior furnishing!

CHAPTER 7

CONCLUTION

7.1 CONCLUSION

All the different parts present in the game were able to run fine in testing phase and hopefully it will also run flawlessly in the Beta release. All parts in games and library are truly to player needs and database can be as huge as player base. Database can be backed up before any changes in code and the update can be pushed.

7.2 FUTURE WORK

As a developing project there's always a chance of developing which could lead to more advanced player interaction and help the server to be more stable in features circumstances.

Huge specific feature works can be performed on this project such as making a black market where people can buy all such of stuffs in it currently people have to get their driving license by performing driving test in game whereas a black market available accessing through the phone can allow user to gain access by paying some in game currencies by which people can also avail their own licences such as gun, aeroplane, ship licences over the black market, for now only limited amounts of items are available in the shop whereas on upcoming update shops can also provide more amenities to the players which can be helpful at that moment.

In the current version of the game people must store the car vehicle in their hands and all the car and it can be seen inside the player profile whereas in the future upgrade this car keys can be made so that it is virtually seen only in one section of the players profile and not in their hands or bag.

On future upgrades many more roles can be provided on the organizations and the database has the ability to adjust all the following roles, people who have to quote the rules and their ranking with their abilities in the program.

7.3 IMPLEMENTATION ISSUE

Part of game cannot be made overnight, and it has to be adjusted overtime some implementation issues can be seen and can be rectified only upon long term use few implementation issues has been noticed such as for now the time to interact with the game and the database is set for 3 minutes that is every 3 minutes the game would

interact with the database and would upload the logs as well as the player activities player creating new accounts and all such into the database whereas upon long us implementation issues may occur and people might face issues where people create their own account and it doesn't get reflected in that database.

Another implementation issues can be seen search as if the data or the game is idle over a long time on the server the server might stop responding and stop accepting any input from the user if any user try to enter into the game there account might not be created or their interaction would not be registered over the server as the result people might not be able to perform any activities and some person physically has to come to the server and try to manually reactive the server.

Another implementation issues can be seen such as people's stopping or running over people in the green zones as marked in the game. Green Zone is designed in such a way that people won't be able to use their gun, ram over people and perform any kind of stuff which can be harmful for a player or would make the player unsafe in that region.

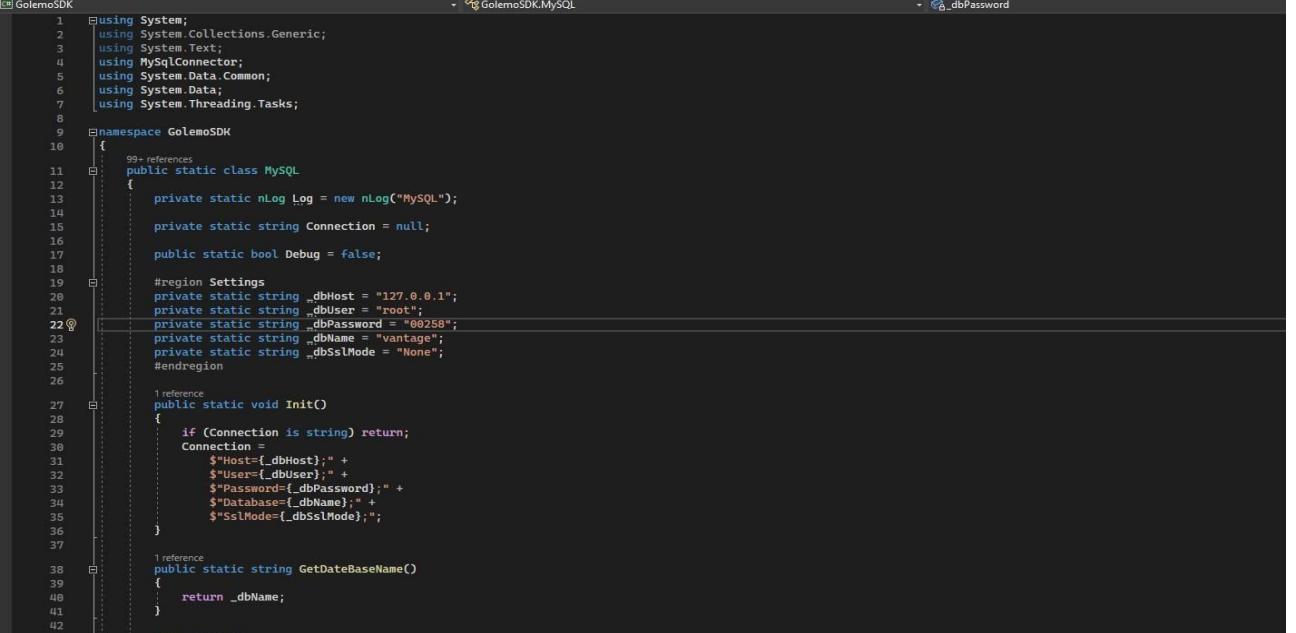
REFERENCES

1. Knowledge of Server Hosting- <https://ragemp.pro/>
2. Knowing RageMP- https://wiki.rage.mp/index.php?title=Main_Page
3. How to create server-
https://wiki.rage.mp/index.php?title=Getting_Started_with_Server#:~:text=Setting%20up%20the%20server,-
[1.&text=Open%20your%20server%2Dfiles%20folder,Read%20more%20about%20Server%20Settings.](https://wiki.rage.mp/index.php?title=Open%20your%20server%2Dfiles%20folder,Read%20more%20about%20Server%20Settings.)
4. Creating Basic Server- <https://rage.mp/forums/topic/2366-creating-a-basic-server/>
5. GTAV- https://en.wikipedia.org/wiki/Grand_Theft_Auto_V
6. GTAV Terms- <https://www.rockstargames.com/gta-v>
7. RageMP- <https://rage.mp/>
8. Roleplay- <https://dotesports.com/streaming/news/what-is-gta-v-roleplay>

APPENDIX

A. Source Code

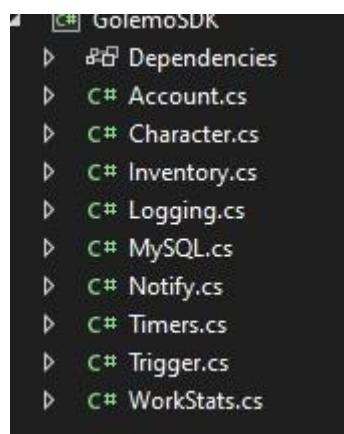
It's always best to start the source code by connecting the database to the game server below is the picture which shows how the database is connected to the game.



```
1  using System;
2  using System.Collections.Generic;
3  using System.Text;
4  using MySqlConnector;
5  using System.Data.Common;
6  using System.Data;
7  using System.Threading.Tasks;
8
9  namespace GolemoSDK
10 {
11     public static class MySQL
12     {
13         private static nLog _Log = new nLog("MySQL");
14
15         private static string Connection = null;
16
17         public static bool Debug = false;
18
19         #region Settings
20         private static string _dbHost = "127.0.0.1";
21         private static string _dbUser = "root";
22         private static string _dbPassword = "00258";
23         private static string _dbName = "vantage";
24         private static string _dbSslMode = "None";
25         #endregion
26
27         public static void Init()
28         {
29             if (Connection != string.Empty) return;
30             Connection =
31                 $"Host={_dbHost};"
32                 $"User={_dbUser};"
33                 $"Password={_dbPassword};"
34                 $"Database={_dbName};"
35                 $"SslMode={_dbSslMode};";
36         }
37
38         public static string GetDatabaseName()
39         {
40             return _dbName;
41         }
42
43     }
44 }
```

SQL connection

All the different entities which are required to pull into the games such as account details character details inventory which the people hold how the logging is done how were the notification of the messages are placed and timer of the game what triggers and when it should be triggers and all the workspace stats are all saved into one .cs file. View of the screenshots of all the entities available are given below.



Files with all variables

```

1  using System;
2  using System.Collections.Generic;
3
4  namespace GolemoSDK
5  {
6      public class AccountData
7      {
8          public string Login { get; protected set; }
9          public string Email { get; set; }
10         public string Password { get; set; }
11
12         public string HWID { get; protected set; }
13         public string IP { get; protected set; }
14         public string SocialClub { get; protected set; }
15
16         public long RedBucks { get; set; }
17         public int VipLvl { get; set; }
18         public DateTime VipDate { get; set; } = DateTime.Now;
19
20         public List<string> PromoCodes { get; set; }
21         public List<int> Characters { get; protected set; } // characters uuids
22
23         public bool PresentGet { get; set; } = false;
24     }
25 }
```

Accounts variables

```

1  using GTANetworkAPI;
2  using System;
3  using System.Collections.Generic;
4
5  namespace GolemoSDK
6  {
7      public class CharacterData
8      {
9          public int UUID { get; set; } = -1;
10         public int PersonID { get; set; } = 1;
11         public Vector3 SpawnPos { get; set; } = new Vector3(0, 0, 0);
12         public DateTime CreateDate { get; set; } = DateTime.Now;
13         public DateTime BirthDate { get; set; } = DateTime.Now;
14         public string FirstName { get; set; } = null;
15         public string LastName { get; set; } = null;
16         public bool Gender { get; set; } = true;
17         public int Health { get; set; } = 100;
18         public int Armor { get; set; } = 0;
19         public int LVL { get; set; } = 1;
20         public int EXP { get; set; } = 0;
21         public long Money { get; set; } = 500;
22         public int Bank { get; set; } = 0;
23         public int WorkID { get; set; } = 0;
24         public int FractionID { get; set; } = 0;
25         public int FractionLVL { get; set; } = 0;
26         public int ArrestTime { get; set; } = 0;
27         public int Fines { get; set; } = 0;
28         public int Water { get; set; } = 100;
29     }
30 }
```

Character used variables.

The screenshot shows a code editor window with a dark theme. The file is named 'GUILMOSDK\ItemTypes.cs'. The code defines a class with various properties representing item types. The properties are grouped by category: Liquids, Weapons, and Rifles. Each group contains several items with their corresponding numerical values.

```
57     LcnDrink2 = 27,  
58     LcnDrink3 = 28,  
59  
60     ArmDrink1 = 29,  
61     ArmDrink2 = 30,  
62     ArmDrink3 = 31,  
63  
64     /* Weapons */  
65     /* Pistols */  
66     Pistol = 100,  
67     CombatPistol = 101,  
68     Pistol50 = 102,  
69     SNSPistol = 103,  
70     HeavyPistol = 104,  
71     VintagePistol = 105,  
72     MarksmanPistol = 106,  
73     Revolver = 107,  
74     APPistol = 108,  
75     FlareGun = 110,  
76     DoubleAction = 111,  
77     PistolMk2 = 112,  
78     SNSPistolMk2 = 113,  
79     RevolverMk2 = 114,  
80     /* SMG */  
81     MicroSMG = 115,  
82     MachinePistol = 116,  
83     SMG = 117,  
84     AssaultSMG = 118,  
85     CombatPDW = 119,  
86     MG = 120,  
87     CombatMG = 121,  
88     Gusenberg = 122,  
89     MiniSMG = 123,  
90     SMGMk2 = 124,  
91     CombatMGm2 = 125,  
92     /* Rifles */  
93     AssaultRifle = 126,  
94     CarbineRifle = 127,  
95     AdvancedRifle = 128,  
96     SpecialCarbine = 129,  
97     BullpupRifle = 130,  
98     CompactRifle = 131,  
99     AssaultRifleMk2 = 132,  
100    CarbineRifleMk2 = 133,  
101    SpecialCarbineMk2 = 134,
```

Inventory variables

Any player can get any type of notification it can either be an error kind of notification or alert, info, success message or warning message below is a picture of all the various types of notifications a person can get.

```

25     Bottom,
26     BottomLeft,
27     BottomCenter,
28     BottomRight
29 }
30 99+ references
31 public static class Notify
32 {
33     99+ references
34     public static void Send(Player client, NotifyType type, NotifyPosition pos, string msg, int time = 2500)
35     {
36         Trigger.ClientEvent(client, "notify", type, pos, msg, time);
37     }
38 99+ references
39     public static void Error(Player client, string msg, int time = 2500)
40     {
41         Trigger.ClientEvent(client, "notify", NotifyType.Error, NotifyPosition.BottomCenter, msg, time);
42     }
43 32 references
44     public static void Alert(Player client, string msg, int time = 2500)
45     {
46         Trigger.ClientEvent(client, "notify", NotifyType.Alert, NotifyPosition.BottomCenter, msg, time);
47     }
48 12 references
49     public static void Info(Player client, string msg, int time = 2500)
50     {
51         Trigger.ClientEvent(client, "notify", NotifyType.Info, NotifyPosition.BottomCenter, msg, time);
52     }
53 81 references
54     public static void Succ(Player client, string msg, int time = 2500)
55     {
56         Trigger.ClientEvent(client, "notify", NotifyType.Success, NotifyPosition.BottomCenter, msg, time);
57     }
58 }

```

Notification types

Now let's investigate the main file which holds all the data of the player which is needed for the interaction which includes all the player names the player accounts the uids which will help the administrator of the server monitor which ID is assigned to which player and its easily allowed the admin to ban players any circumstances it also includes the level of the players the level in the fraction.

```

Notify.cs    What's New?    MySQL.cs
Golemo          Main.cs
19  using System.Net.Mail;
20  using Golemo.Voice;
21  using Golemo.Houses;
22  using Golemo.Fractions.Activity;
23  using Golemo.Fractions;
24  using Golemo.Scripts;
25  using Golemo.Entertainment;
26
27  namespace Golemo
28  {
29
30      99+ references
31      public class Main : Script
32      {
33          1 reference
34          public static string Codename { get; } = "Shadow";
35          1 reference
36          public static string Version { get; } = "1.0";
37          1 reference
38          public static string Build { get; } = "1000";
39          //////
40          1 reference
41          public static string Full { get; } = $"{Codename} {Version} {Build}";
42          1 reference
43          public static DateTime StartDate { get; } = DateTime.Now;
44          1 reference
45          public static DateTime CompileDate { get; } = new FileInfo(Assembly.GetExecutingAssembly().Location).LastWriteTime;
46
47          //////
48          public static List<int> UUIDs = new List<int>(); // characters UUIDs
49          public static List<int> PersonIDs = new List<int>(); // characters PersonIDs
50          public static Dictionary<int, string> PlayerNames = new Dictionary<int, string>(); // character uid - character name
51          public static Dictionary<string, int> PlayerBankAcc = new Dictionary<string, int>(); // character name - character bank
52          public static Dictionary<string, int> PlayerUUIDs = new Dictionary<string, int>(); // character name - character uid
53          public static Dictionary<string, int> PersonPlayerIDs = new Dictionary<string, int>(); // character name - character id
54          public static Dictionary<int, Tuple<int, int, int, long> > PlayersSlotsInfo = new Dictionary<int, Tuple<int, int, int, long>>(); // character uid - lvl,exp,fraction,money
55
56          public static Dictionary<string, Player> LoggedIn = new Dictionary<string, Player>();
57          public static Dictionary<Player, Character> Players = new Dictionary<Player, Character>(); // character in

```

Main Player code

The next 2 pictures you will look into include the main login page which player has to interact every single time they launch the game, and they wish to enter inside the game this 2-picture given below shows the login and a way to create a new account who is the player needs to undergo.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title></title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <meta name="viewport" content="width=device-width,initial-scale=1">

    <link rel="stylesheet" href="https://odnus.cloudflare.com/sjas/lib/font-awesome/5.12.0-2/css/all.min.css" integrity="sha256-46r060N2lrChLbb5zowXQ72/iK9Niw/lAmygHExk/o=" crossorigin="anonymous" />
    <script src="https://odnus.cloudflare.com/sjas/1/typed.js/2.0.11/typed.min.js"></script>
    <link href="https://fonts.googleapis.com/css?family=Montserrat:100,200,300,400,500,600,700,800,900" rel="stylesheet">

    <link rel="stylesheet" href="assets/css/noty.css">
    <link rel="stylesheet" href="assets/css/theme.css">
    <link rel="stylesheet" href="assets/auth/img/1v.png" class="logo1">
    <link rel="stylesheet" href="assets/auth/img/4v.png" class="logo4">
    <link rel="stylesheet" href="assets/auth/img/3v.png" class="logo3">
    <link rel="stylesheet" href="assets/auth/img/2v.png" class="logo2">
    <link rel="stylesheet" href="assets/auth_new.css" />
    <link rel="stylesheet" href="../../assets/tots/bobas-neone/styleSheet.css">

    <script src="https://cdn.jsdelivr.net/npm/vue@2.6.12"></script>
  </head>
  <body>
    <div class="switch-content authBody">
      <main class="authMain reg-and-restore restorepass rest-page show">
        <div class="border">
          <img alt="Assets/auth/img/4v.png" class="logo4">
          <img alt="Assets/auth/img/3v.png" class="logo3">
          <img alt="Assets/auth/img/2v.png" class="logo2">
        </div>
        <div class="text1"><b>Enter</b> to continue</div>
        <div class="text2">You are in open beta testing, so please report any bugs you find, in our discord:&ltb style="font-weight: 600; font-size: 3.5vh;"> https://discord.gg/W4WpDs8z3 </b></div>
      </main>
      <transition name="fade">
        <main class="authB authorization auth-page">
          <form id="auth-form" action="" method="POST">
            <img alt="Assets/auth/img/1v.png" class="personage">
            <img alt="Assets/auth/img/2v.png" class="logos">
            <img alt="Assets/auth/img/3v.png" class="logos">
            <div class="border">
              <div class="box">
                <div class="text1">AUTHORIZATION</div>
                <div class="text2">Log in to your account to continue playing on the server</div>
                <div class="inputbox">
                  <div class="label">ENTER LOGIN</div>
                  <input type="text" class="the-input entry-login" name="entry-login-id" id="entry-login-id" placeholder="ENTER DATA" required>
                </div>
                <div class="inputbox">
                  <div class="label">ENTER PASSWORD</div>
                  <input type="password" class="the-input" id="entry-password-id" name="entry-password" placeholder="ENTER DATA" required>
                </div>
              </div>
            </div>
          </form>
        </main>
      </transition>
    </div>
  </body>
</html>
```

HTML Login page

```
</transition>
<transition name="fade">
<main class="authreg reg-and-restore registration reg-page">
  <div class="border"></div>
  <img alt="Assets/auth/img/7.png" class="logo6">
  <img alt="Assets/auth/img/5.png" class="logo5">
  <img alt="Assets/auth/img/6.png" class="logo4">
  <div class="border">
    <form id="reg-form" action="" method="POST">
      <div class="text1">CREATING A NEW ACCOUNT</div>
      <div class="text2">CREATE YOUR SNULL CHARACTER AND BUILD HIS FUN LIFE IN LOS SANTOS</div>
      <div class="blockinput">
        <div class="inputbox">
          <div class="label">ENTER MAIL</div>
          <input class="the-input" type="email" name="new-user_email" id="new-user_email-id" placeholder="ENTER DATA" required>
        </div>
        <div class="inputbox">
          <div class="label">ENTER LOGIN</div>
          <input type="text" class="the-input new-user_login" name="new-user_login" id="new-user_login-id" placeholder="ENTER DATA" required>
        </div>
        <div class="inputbox">
          <div class="label">ENTER PASSWORD</div>
          <input class="the-input" type="password" name="new-user_pw" id="new-user_pw-id" placeholder="ENTER DATA" required>
        </div>
        <div class="inputbox">
          <div class="label">REPEAT PASSWORD</div>
          <input class="the-input" type="password" name="new-user_pw-repeat" id="new-user_pw-repeat-id" placeholder="ENTER DATA" required>
        </div>
        <div class="inputbox">
          <div class="label">PROMO CODE</div>
          <input type="text" class="the-input" name="new-user_promo-code" id="new-user_promo-code" placeholder="ENTER DATA">
        </div>
      </div>
      <div class="btn btn-register-end">REGISTRATION</div>
      <div class="btnreg js-btn-back">AUTHORIZATION</div>
    </form>
  </div>
</main>
</transition>
```

HTML create account page.

Now let us look at the part were all the vehicles are stored in the house, inside the house there is a separate section known as garage every house has different types of garages depending upon the cost of the house there are 3 types of houses present in

the ecosystem one includes the basic one which has only limited 6 garages other comes with 15 and followed by 50 garages people have the option to buy as many cars as they want but provided they should have their spacing in the garage if there is no space in the garage person cannot buy any vehicles and henceforth he has to either sell the vehicle to a player or has to sell the vehicle towards the government

the below picture describes the hi houses which are defined by the garage via ID followed by which there is also a part which includes parking fine fusspots in the map are located or designed in such a way that the parking fine would automatically be given to the player if the vehicle is parked at a location than a certain amount of time this would be notified to the player by the notification section.

```

namespace Golemo.Houses
{
    #region GarageType Class
    class GarageType
    {
        public Vector3 Position { get; }
        public List<Vector3> VehiclesPositions { get; }
        public List<Vector3> VehiclesRotations { get; }
        public int MaxCars { get; }

        public GarageType(Vector3 position, List<Vector3> vehiclesPositions, List<Vector3> vehiclesRotations, int maxCars)
        {
            Position = position;
            VehiclesPositions = vehiclesPositions;
            VehiclesRotations = vehiclesRotations;
            MaxCars = maxCars;
        }
    }
    #endregion

    #region Garage Class
    class Garage
    {
        public int ID { get; }
        public int Type { get; }
        public Vector3 Position { get; }
        public Vector3 Rotation { get; }
        [JsonIgnore] public int Dimension { get; set; }

        [JsonIgnore]
        private ColShape shape;

        [JsonIgnore]
        private ColShape intShape;
        [JsonIgnore]
        private Marker intMarker;

        [JsonIgnore]
        public Dictionary<string, Tuple<int, Entity>> entityVehicles = new Dictionary<string, Tuple<int, Entity>>();
        [JsonIgnore]
        public Dictionary<string, Entity> vehiclesOut = new Dictionary<string, Entity>();
        private nLog Log = new nLog("Garage");

        public Garage(int id, int type, Vector3 position, Vector3 rotation)
        {
            ID = id;
            Type = type;
            Position = position;
            Rotation = rotation;

            shape = NAPI.ColShape.CreateCylinderColShape(position - new Vector3(0, 0, 1), 1, 3, 0);
            shape.OnEntityEnterColShape += (s, ent) =>
            {
                try
                {
                    NAPI.Data.SetEntityData(ent, "GARAGEID", id);
                }
                catch { }
            };
        }
    }
}

```

Garage in house

Parking in in this game refers to all the entities such as parking the car in a space rearrange in the car in the slot which includes all the vehicles in the organization, while paying the parking feet also includes the money present in the players hand or not in parking the garage facility is also defined in the players section where a player can call their vehicle from the garage near them and they can also tell they are car sell their keys, make a new duplicate key for their own car or handover the key to his friends so that he can use the vehicle.

```

        }

    }

    [ServerEvent(Event.PlayerEnterVehicle)]
    public void Event_OnPlayerEnterVehicle(Player player, Vehicle vehicle, byte seatid)
    {
        try
        {
            if (!vehicle.GetData<bool>("PARKING") || !vehicle.HasData("PARKING"))
            {
                return;
            }
            vehicle.ResetData("PARKING");
            Trigger.ClientEvent(player, "deleteWorkSlip");
            Trigger.ClientEvent(player, "deleteCheckpoint", 900);
        }
        catch (Exception e) { Log.Write("PlayerEnterParkingVehicle: " + e.Message, nLog.Type.Error); }
    }

    [RemoteEvent("takerentparking")]
    public static void BuyParkPlace(Player player, string name, string number)
    {
        if (!Main.Players.ContainsKey(player)) return;
        if (!player.HasData("PARKFINE"))
        {
            Notify.Alert(player, "You must be near a valet", 4500);
            return;
        }
        ParkFine park = _parkFines.Find(x => x.ID == player.GetData<int>("PARKFINE"));
        if (park == null)
        {
            Notify.Error(player, "Unexpected error", 5000);
            return;
        }
        int costcar = park.Price;
        if (Main.Players[player].Money < costcar)
        {
            Notify.Send(player, NotifyType.Error, NotifyPosition.BottomCenter, $"Insufficient funds, you need: ${costcar}", 3000);
            return;
        }
        if (number == "" || number == null)
        {
            Notify.Send(player, NotifyType.Error, NotifyPosition.BottomCenter, $"You don't have a car", 3000);
            return;
        }
        foreach (var v in NAPI.Pools.GetAllVehicles())
        {
            if (v.HasData("ACCESS") && v.GetData<string>("ACCESS") == "PERSONAL" && NAPI.Vehicle.GetVehicleNumberPlate(v) == number)
            {
                Notify.Send(player, NotifyType.Error, NotifyPosition.BottomCenter, $"Machine already called", 3000);
                return;
            }
        }
        MoneySystem.Wallet.Change(player, -costcar);
        SetCarInFreeParkPlace(player, name, number);
    }
}

```

Parking file

parking file also includes the necessity of the player to buy a garage for the first time as the land into the city a player can buy a parking space in the parking slot over the map and they get 3 slots father car they can buy 3 cars even without having their own house and garage the car automatically spawn over the parking area and they can take use of their vehicle and access all the vehicle facilities from those parking spots marked on the map.

Coming to the next part all houses has their own furniture, in the furniture section there are 3 things which can be stored in the wardrobe the first one comes with the gun store where the player has able to store all the gun and the ammos inside them the second part comes with the clothing were the person or the player can store all the clothes inside their wardrobe and the 3rd thing which includes the objects a person can collect different entities present into the world and they can store it inside their wardrobe the below picture defines the same as we go through the code

```

        {
            obj = NAPI.Object.CreateObject(NAPI.Util.GetHashKey(Model), Position, Rotation, 255, Dimension);
            obj.SetData("ID", ID);
            return obj;
        }
    }

    class FurnitureManager : Script
    {
        private static nLog Log = new nLog("HouseFurniture");
        public static Dictionary<int, Dictionary<int, HouseFurniture>> HouseFurnitures = new Dictionary<int, Dictionary<int, HouseFurniture>>();
        public static Dictionary<int, Dictionary<int, List<Item>>> FurnitureItems = new Dictionary<int, Dictionary<int, List<Item>>>();
        public FurnitureManager()
        {
            var result = MySQL.QueryRead("SELECT * FROM `furniture`");
            if (result == null || result.Rows.Count == 0)
            {
                Log.Write("DB return null result.", nLog.Type.Warn);
                return;
            }
            foreach (DataRow Row in result.Rows)
            {
                Dictionary<int, HouseFurniture> furnitures = JsonConvert.DeserializeObject<Dictionary<int, HouseFurniture>>(Row["furniture"].ToString());
                Dictionary<int, List<Item>> item = JsonConvert.DeserializeObject<Dictionary<int, List<Item>>>(Row["data"].ToString());
                int id = Convert.ToInt32(Row["uuid"].ToString());
                HouseFurnitures.Add(id, furnitures);
                FurnitureItems.Add(id, item);
            }
            Log.Write($"Loaded {HouseFurnitures.Count} players furnitures.", nLog.Type.Success);
        }
        public static Dictionary<string, string> NameModels = new Dictionary<string, string>()
        {
            { "Gun safe", "prop_id_int_safe_01" },
            { "A wardrobe", "prop_rub_cabinet02" },
            { "Closet With object", "prop_rub_cabinet01" },
        };
        public static void Save()
        {
            foreach (var data in HouseFurnitures)
            {
                string furniture = JsonConvert.SerializeObject(data.Value);
                string item = JsonConvert.SerializeObject(FurnitureItems[data.Key]);
                MySQL.Query($"UPDATE `furniture` SET `furniture` ='{furniture}', `data` ='{item}' WHERE `uuid`='{data.Key}'");
            }
            Log.Write("Furnitures has been saved to DB", nLog.Type.Success);
        }
        public static void Create(int id)
        {
            if (!HouseFurnitures.ContainsKey(id))
            {
                HouseFurnitures.Add(id, new Dictionary<int, HouseFurniture>());
                FurnitureItems.Add(id, new Dictionary<int, List<Item>>());
                MySQL.Query($"INSERT INTO `furniture`(`uuid`, `furniture`, `data`) VALUES ({id}, '{JsonConvert.SerializeObject(new Dictionary<int, HouseFurniture>())}', '{JsonConvert.SerializeObject(new Dictionary<int, List<Item>>())}'");
            }
        }
    }
}

```

Wardrobe

There are also few files which include the dll files this dll files are used by the game to render the objectives over the map this dll files are basically the graphic base designs which are set in the game this set of dll files are has to be first linked into the code given a separate amount of values and then this dll files has to be set inside the database which allows it to get directly linked with the dictionary after successfully setting up the dictionary and the code a person can use the car or the vehicle and also the building inside the game below is the picture was represents view of the dll files present inside the game.

map_props_newyear	7/9/2022 11:38 AM	File folder
map_props_ship	7/9/2022 11:38 AM	File folder
map_props_xmas	7/9/2022 11:38 AM	File folder
map_props2	7/9/2022 11:38 AM	File folder
map_props3	7/9/2022 11:38 AM	File folder
map_rancho	7/9/2022 11:38 AM	File folder
map_xmas_ru	7/9/2022 11:38 AM	File folder
marabunta	7/9/2022 11:38 AM	File folder
ppgolfrm	7/9/2022 11:38 AM	File folder
sheriffpaleto	7/9/2022 11:38 AM	File folder
tattoos_pack	7/9/2022 11:38 AM	File folder
vagos	7/9/2022 11:38 AM	File folder
veh_17r35	7/9/2022 11:38 AM	File folder
veh_18rs7	7/9/2022 11:38 AM	File folder
veh_18Velar	7/9/2022 11:38 AM	File folder
veh_63gls	7/9/2022 11:38 AM	File folder
veh_63gls2	7/9/2022 11:38 AM	File folder
veh_350z	7/9/2022 11:38 AM	File folder
veh_2019M5	7/9/2022 11:38 AM	File folder

dll files

This file includes all the map options which includes also predefined entities such as hospitals Sharif fractions design it also includes tattoo packs which a player can have them on the body of the players it also includes the vehicle design view of the vehicle names can be seen in the figure 9.13 search as 63glsl it also includes all the clothing packs which can be acquired by the player this clothing pack includes all the clothing clothes design which clothes can be bought by the player from the shop the clothing shop. This dll files can be repeatedly edited using OpenIV tool Which enables us to edit or rename the objective of a game.

Few dll files such as car dll is directly linked with the database admin has the permission to spawn this dll car and test at even this dll cars can also be added to the fraction such as if he add a different car design for an emergency medical services it can also be spawned for the certain level of rank into the game. Game directly accepts the dll file name in order to spawn the vehicle.

Admin panel for all the players has been turned off only the player with certain promotion level has the access to open the admin panel the design of the admin panel is specified in the client side. Player can access the admin panel by clicking on F8 button on their keyboard. Below is the represented code for popping up the admin panel for the player.

```

let adminPanelState = false;

mp.keys.bind(0x77, false, function () {
    if(adminPanelState) global.adminPanel.execute("admlist.closepanel()");
    if (!loggedin || chatActive || editing || new Date().getTime() - lastCheck < 1000 || global.menuOpened || !localplayer.getVariable("IS_ADMIN")) return;
    else mp.events.callRemote('openAdminPanel');
    lastCheck = new Date().getTime();
});

mp.events.add("openAdminPanel", (json, json2) => {
    if (!loggedin || chatActive || editing || cuffed) return;
    global.adminPanel = mp.browsers.new('package://browser/modules/AdminPanel/index.html');
    global.adminPanel.active = true;
    global.menuOpen();
    adminPanelState = true;
    global.adminPanel.execute(`admlist.active=true`);
    global.adminPanel.execute(`admlist.cmdlist=${json}`);
    global.adminPanel.execute(`admlist.items=${json2}`);
});

mp.events.add("closeAdminPanel", () => {
    if(global.adminPanel)
    {
        global.menuClose();
        adminPanelState = false;
        global.adminPanel.active = false;
        global.adminPanel.destroy();
    }
});

mp.events.add("getPlayerInfo", (id) => {
    mp.events.callRemote('getPlayerInfoToAdminPanel', id);
});

mp.events.add("loadPlayerInfo", (json) => {
    global.adminPanel.execute(`admlist.player=${json}`);
});

```

Admin Panel.

Below is the picture of the UI which is design in order to open the admin panel this

primary panel enables the player to see how many players are offline and their functions to perform on the players and many more.

```

1 R:\www\index.html>
2 <html lang="ru" dir="ltr">
3   <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="css/adminpanel.css">
6     <link rel="stylesheet" href="../../assets/css/font.css">
7     <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.css" integrity="sha512-5A8nwdMOWrSz20fDajczgUiUdUBR8liPYU+WyTZP1lmY9G6Oc7H1Zv156XqmsgNUz" crossorigin="anonymous">
8     <script src="https://cdn.jsdelivr.net/npm/vue@2.6.12"></script>
9     <title>AdminPanel</title>
10    </head>
11
12  <body>
13    <div id="app" v-if="active" v-cloak>
14      <div class="container">
15        <header>
16          <i class="fa fa-users" aria-hidden="true"></i>
17          <h1>Admin panel</h1>
18          <span>
19            <i @click="closepanel()" class="fa fa-times-circle-o" aria-hidden="true"></i>
20          </span>
21        </header>
22
23        <div class="main pageMembers">
24          <h2>List of players</h2>
25          <div class="items">
26            <table border="1">
27              <thead>
28                <tr class="item">
29                  <th class="adminItem">[&lt;/th>
30                  <th class="idItem" style="text-align: center; width: 100px; vertical-align: middle; padding: 5px; border: none; background-color: #f0f0f0; border-bottom: 1px solid black; border-radius: 5px; font-size: 10px; font-weight: bold; color: black; text-decoration: none; transition: all 0.3s ease-in-out; position: relative; z-index: 1; ">[&lt;/th>
31                  <th class="nickItem" style="text-align: center; width: 100px; vertical-align: middle; padding: 5px; border: none; background-color: #f0f0f0; border-bottom: 1px solid black; border-radius: 5px; font-size: 10px; font-weight: bold; color: black; text-decoration: none; transition: all 0.3s ease-in-out; position: relative; z-index: 1; ">[&lt;/th>
32                  <th class="cmdItem" style="text-align: center; width: 100px; vertical-align: middle; padding: 5px; border: none; background-color: #f0f0f0; border-bottom: 1px solid black; border-radius: 5px; font-size: 10px; font-weight: bold; color: black; text-decoration: none; transition: all 0.3s ease-in-out; position: relative; z-index: 1; ">[&lt;/th>
33                  <th class="pingItem" style="text-align: center; width: 100px; vertical-align: middle; padding: 5px; border: none; background-color: #f0f0f0; border-bottom: 1px solid black; border-radius: 5px; font-size: 10px; font-weight: bold; color: black; text-decoration: none; transition: all 0.3s ease-in-out; position: relative; z-index: 1; ">[&lt;/th>
34                </tr>
35                <tr class="item" v-for="item in items">
36                  <td>
37                    <i class="fa fa-toggle-on" aria-hidden="true" v-if="item[0] > 0" :title="item[0]></i>
38                    <i class="fa fa-toggle-off" aria-hidden="true" v-else></i>
39                  </td>
40                  <td>{{item[1]}}</td>
41                  <td>{{item[2]}}</td>
42                  <td>
43                    <span>
44                      <i @click="command('tp ' + item[1])" title="Телепорт к игроку" class="fa fa-map-marker" aria-hidden="true"></i>
45                      <i @click="getPlayerInfo(item)" class="fa fa-address-book" title="Получить информацию" aria-hidden="true"></i>
46                      <i @click="command('kick ' + item[1])" title="Кикнуть игрока" class="fa fa-user-times" aria-hidden="true"></i>
47                      <i @click="selectPlayer(item)" title="Другие команды" class="fa fa-bars" aria-hidden="true"></i>
48                    </span>
49                  </td>
50                <td>{{item[3]}}</td>
51              </tr>
52            </table>
53          </div>
54        </div>
55      </div>
56    </div>
57  </body>
58</html>

```

Admin Panel UI

This above admin panel UI design enables the admin to add health, remove gun from the player, add ban to the user, give the player a jail time, etc all this data is mentioned in the admin panel UI which can easily be given or taken away from a player via admin. There is also option for the admin to teleport to the player and also have a flying look over the person to observe all the activities if any suspicious or any active player reports to the admin.

As to know earlier bad there are different rankings for different fraction levels such as emergency medical services sheriff levels fib every tracking levels have their own amount and their own transaction limit every organization people have their own ranking which can be seen specially on their different menu infraction tablet the ranking of the fraction people as well as their payday is saved directly on to the database and its pulled automatically from the database towards the game logs that is vantage logs.

	idkey	fraction	rank	payday	name	clothesm	clothesf
	226	13	3	300	Suspicion		
	227	13	4	400	Anzagorts		
	228	13	5	500	Jepkir		
	229	13	6	600	Ehpair		
	230	13	7	700	Nayoh		
	231	13	8	800	Goh		
	232	13	9	900	Cherry		
	233	13	10	1000	Bair		
	242	14	1	55000	Recruit	army_1	army_1
	243	14	2	60000	Private	army_2	army_2
	244	14	3	65000	Specialist	army_3	army_3
	245	14	4	70000	Corporal	army_4	army_4
	246	14	5	75000	Sergeant	army_5	army_5
	247	14	6	80000	Sergeant	army_6	army_6
	248	14	7	85000	Petty officer	army_7	army_7
	249	14	8	90000	Sergeant Major	army_8	army_8

Fraction ranking

Play has also been given option to perform animation in the game which helps the player for better interaction over gesture there are many gestures which can be performed in games such as rubbing neck, carrying a person, performing a dance emote, waving hand, throwing cigarettes, Bowing down, thumbs up, begging,etc.

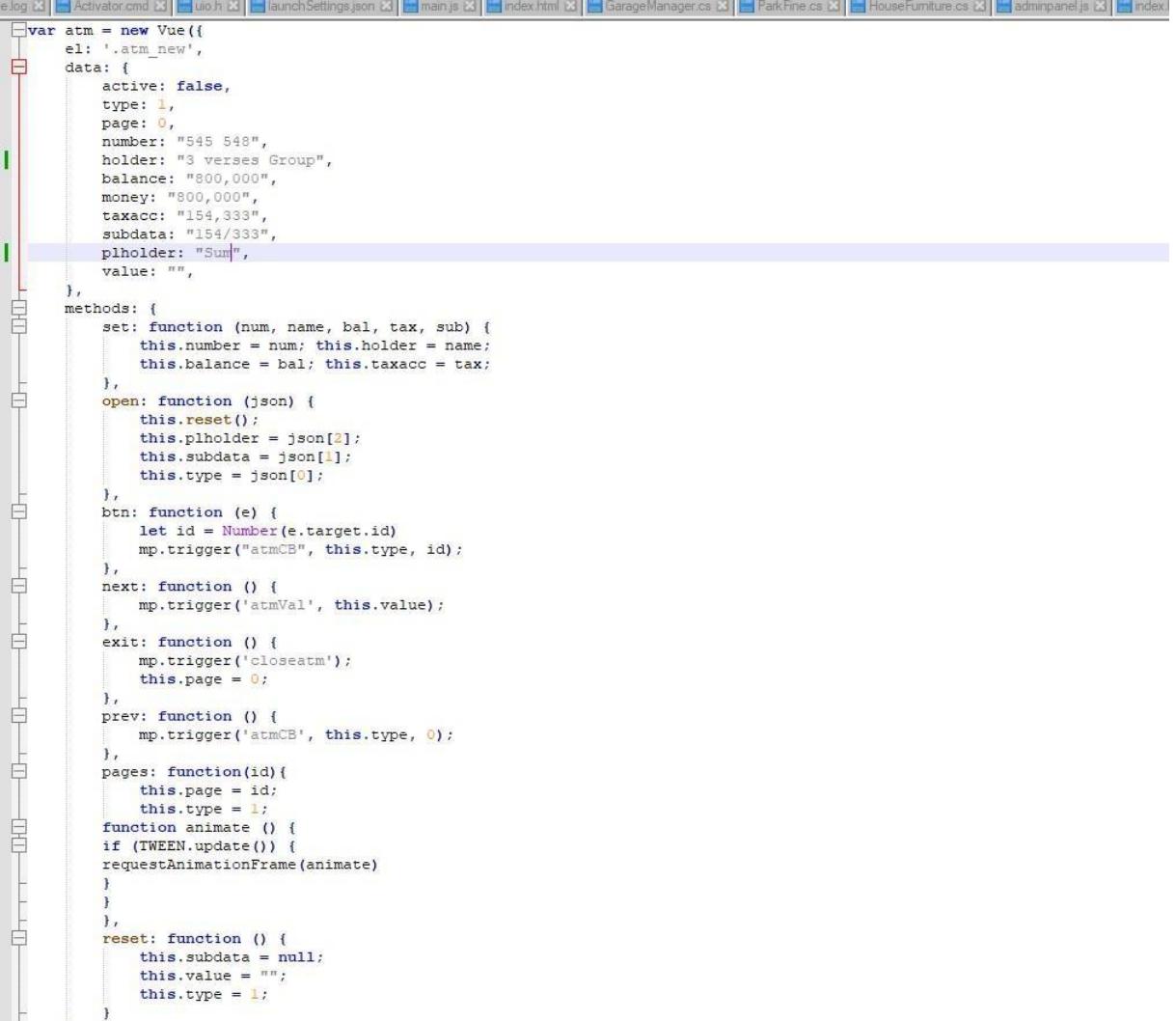
```

1 var animationList = [
2   {
3     id: 0,
4     name: 'Actions',
5     img: 'actions',
6     special: false,
7     animations: [
8       {
9         id: 0,
10        name: 'Scratch ass',
11        ad: 'anim@heists@team_respawn@respawn_02',
12        an: 'heist_spawn_02_ped_d',
13        af: 8
14      },
15      {
16        id: 1,
17        name: 'Rub neck',
18        ad: 'amb@world_human_cop_idles@female@idle_a',
19        an: 'idle_c',
20        af: 8
21      },
22      {
23        id: 2,
24        name: 'Rub palms',
25        ad: 'amb@world_human_cop_idles@female@idle_b',
26        an: 'idle_d',
27        af: 8
28      },
29      {
30        id: 3,
31        name: 'Rub arms',
32        ad: 'move_action@p_m_one@unarmed@idle@variations',
33        an: 'idle_a',
34        af: 8
35      },
36      {
37        id: 4,
38        name: 'Pick your nose',
39        ad: 'anim@mp_player_intcelebrationfemale@nose_pick',
40        an: 'nose_pick',
41        af: 8
42      },
43      {
44        id: 5,
45        name: 'Cross your arms-1',
46        ad: 'rcmme_amandal',
47        an: 'stand_loop_cop',
48        af: 9
49      },
50      {
51        id: 6,
52        name: 'Cross your arms-2',
53        ad: 'amb@world_human_cop_idles@female@idle_b',
54        an: 'idle_e',
55        af: 8
56      }
57    ]
58  }
59]

```

Emote animations.

Does every player has to do some activity in the game to earn money and all the money earned can be saved into their bank account there is a difference system of a banking system in the game this banking system code has been given below which represents a bank or an ATM name which is saved into the game and their details or the transaction made by them to give a person money or to earn from it or in order to save the money from the head towards their bank account is saved into the database.



```

var atm = new Vue({
  el: '.atm_new',
  data: {
    active: false,
    type: 1,
    page: 0,
    number: "545 548",
    holder: "3 verses Group",
    balance: "800,000",
    money: "800,000",
    taxacc: "154,333",
    subdata: "154/333",
    pholder: "Sun",
    value: ""
  },
  methods: {
    set: function (num, name, bal, tax, sub) {
      this.number = num; this.holder = name;
      this.balance = bal; this.taxacc = tax;
    },
    open: function (json) {
      this.reset();
      this.pholder = json[2];
      this.subdata = json[1];
      this.type = json[0];
    },
    btn: function (e) {
      let id = Number(e.target.id)
      mp.trigger("atmCB", this.type, id);
    },
    next: function () {
      mp.trigger('atmVal', this.value);
    },
    exit: function () {
      mp.trigger('closeatm');
      this.page = 0;
    },
    prev: function () {
      mp.trigger('atmCB', this.type, 0);
    },
    pages: function(id){
      this.page = id;
      this.type = 1;
      function animate () {
        if (TWEEN.update()) {
          requestAnimationFrame(animate)
        }
      }
    },
    reset: function () {
      this.subdata = null;
      this.value = "";
      this.type = 1;
    }
  }
})

```

Banking

Since every player wants to use the fortune once or the other into the game there is a system of container opening which enables the player to use their luck to see which car or which item they win from the container opening this containers are located at the dockyard of the game Ware all the containers are present there are 3 categories of opening containers basically the first type of container which a person can open is a basic one which requires a very less amount of game currencies second is the medium level which enables player to use a better car or a better items and the 3rd

but not the least person can win a super fast or a very expensive car which can be held by the person after putting a number plate on it towards the government which enables the player to earn huge amount of money below is the picture of the code which enables the container opening.

```

1  <template>
2  	<div el="#boom">
3  		data: {
4  			active: false,
5  			contNumber: '',
6  			type: '',
7  			contClass: '',
8  			bet: '',
9  			minBet: '',
10            betPlayerName: '',
11            info: "НОСТАВИТЬ",
12            put: false,
13            putInput: "0",
14
15            win: false,
16            gain: '',
17            winCost: 1750000
18
19        },
20        methods: {
21            addInfoContainer: function(contID, Type, Name, Price, MinBet, betPlayerName){
22                this.contNumber = contID;
23                this.type = Type;
24                this.contClass = Name;
25                this.bet = Price;
26                this.minBet = MinBet;
27                if(betPlayerName == "null") return;
28                this.betPlayerName = betPlayerName;
29            },
30            addWin: function(prizName, sellPrice){
31                this.gain = prizName;
32                this.winCost = sellPrice;
33                this.info = "ВЫИГРЫШ"
34            },
35            addInfoWinner: function(contID, Type, Name, Price, MinBet, betPlayerName){
36                this.contNumber = contID;
37                this.type = Type;
38                this.contClass = Name;
39                this.bet = Price;
40                this.minBet = MinBet;
41                this.betPlayerName = betPlayerName;
42                this.info = "ПОБЕДА"
43            },
44            setBet: function(){
45                if(this.info == "ВЫИГРЫШ"){
46                    this.win = true
47                    return;
48                }
49                else if(this.info == "ПОБЕДА"){
50                    mp.trigger('OpenContainer')
51                    this.closeContMenu();
52                    return;
53                }
54                this.put = true
55                mp.trigger("ReadBet")
56            },
57            setNewBet: function(){
58                mp.trigger('setNewBet', this.putInput)
59            }
60        }
61    </div>
62  </template>
63
64  <script>
65  export default {
66      name: 'Container'
67  }
68  </script>
69
70  <style>
71  .cont{
72      width: 100px;
73      height: 100px;
74      border: 1px solid black;
75      margin: auto;
76  }
77  </style>

```

Container Opening

Since every player needs some practice on the game for performing any world event in the game there is a specific portion known as arena which can be accessed by going to the arena which is present in the centre of the getto area. In this area player can select how many players plus which area they want to fight there are specific predefined places which are player can fight such as Cayo Perico, Kortz Center, Airport, Camp, Sawmill and Port this doesn't allow any effect on the world section but it creates a specific hidden portion in the game where the people are made to fight in this version people are given a specific number of guns ordered to practice below is the piece of gold which allows player to select which arena they want to play in.

```

page.log.cs | Activator.cmd.cs | Gio.h.cs | launchSettings.json.cs | main.cs | index.html.cs | GarageManager.cs.cs | ParkFine.cs.cs | HouseFurniture.cs.cs | adminpanel.js.cs | index.html.cs | commands.js.cs | adminpanel.js.cs | index.html.cs | animList.js.cs | atm.js.cs
  ↳ var hudis = new Vue({
    el: ".hud_gun",
    data: {
      hud: false,
      kills: 0,
      time: "1s",
      deaths: 0,
      time2: "0s",
      localmap: "Cayo Perico",
    }
  })
  ↳ var gangarena = new Vue({
    el: ".gangarena",
    data: {
      active: false,
      adm: false,
      pass: 0,
      style: 0,
      inlobby: false,
      weapon: 0,
      map: 0,
      lobby: [],
      lobbies: null,
      players: [],
      winners: [],
      arena: "Cayo Perico",
      arenas: ["Cayo Perico", "Kortz Center", "Airport", "Camp", "Sawmill", "Port"],
      modal: false,
    },
    methods: {
      hide: function () {
        mp.trigger('client::closemenu');
      },
      hides: function () {
        this.active = false;
        this.style = 0;
        if (this.inlobby)
        {
          this.adm = false;
          mp.trigger('client::disconnectlobby');
        }
      },
      hidesno: function () {
        this.active = false;
        this.style = 0;
      },
      show: function() {
        this.active = true;
        this.style = 0;
        this.inlobby = false;
        this.adm = false;
        this.weapon = 0;
        this.map = 0;
        this.lobby = [];
      },
      gostyle: function(index) {
        this.style = index;
      }
    }
  })

```

Arena Section

```

16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

```

Arena Game Section

After people doing different activities in the game people also have to do fishing has to run their own shop in the black market and many more this all things comes under the market section of the client side which enables the player to set up their own shop below is a piece of code which represents the fishing market.

```

panel.js | index.html | animList.js | atm.js | containers.js | script.js | index.js | gangarena.js | vue.js | gangarena.html | market.js |
Market
var fishMarket = new Vue({
  el: '#app',
  data: {
    active: false,
    header: "Anthony Williamson",
    page: 0,
    onWork: false,
    buyitems: [],
    sellitems: [],
    curs: 1,
    hays: 1,
    seedcount: 1,
    buyValue: 1,
    sellValue: 1,
    timeout: null,
  },
  methods: {
    setinfo(json) {
      this.curs = json[0]
      this.hays = json[1]
      this.seedcount = json[2]
    },
    changePage(value) {
      this.page = value
      mp.trigger("changePage3", value);
    },
    buy(item) {
      this.buyValue = parseInt(this.buyValue);
      if(this.buyValue <= 0 || this.buyValue == null) this.buyValue = 1;
      mp.trigger("farmerBuy3", item, this.buyValue);
    },
    sell(item) {
      this.sellValue = parseInt(this.sellValue)
      if(this.sellValue <= 0 || this.sellValue == null) this.sellValue = 1;
      mp.trigger("farmerSell3", item, this.sellValue);
    },
    closeMenu() {
      this.active = false
      this.page = 0
      // mp.trigger("closeMarketMenu")
      let main = document.getElementById("mainblock");
      main.style.top="-120vh";
      main.style.opacity="0";
      main.style.transition="all 0.50s";
      this.timeout = setTimeout( () => {
        this.active = false
        this.page = 0
        mp.trigger("closeMarketMenu3");
      }, 300);
    }
  }
})

```

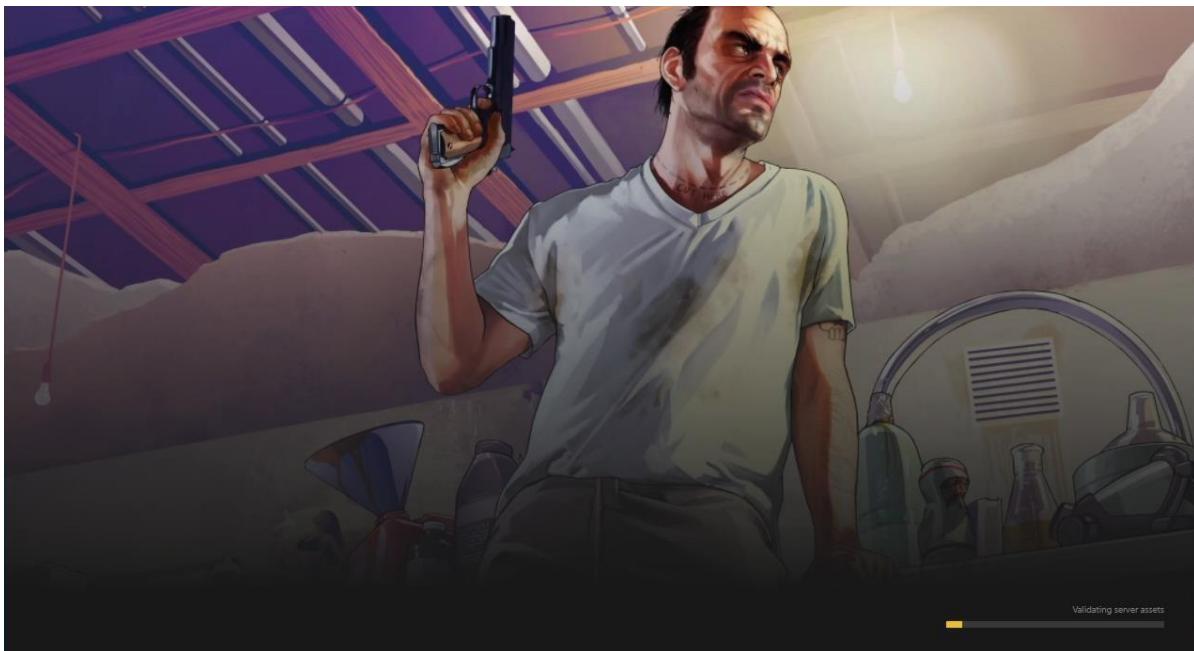
Fish Market

People running their own vehicles would always need to top up their petrol, this is an RP server people has to top up the vehicles keeping the engine on will also consume the vehicle petrol from the vehicle below is the piece of code which allows a petrol pump to roll petrol into the engine's petrol pump is also a business which has to be top up by the business owner materials has to be brought by the business owner towards their own businesses so that the petrol in the in the petrol pump is always full and up to date so that the player can always come and load the petrol from the petrol pump if at any given time the petrol pump runs out of fuel the map would turn the petrol pump in red colour stating the fuel pump is now out of fuel.

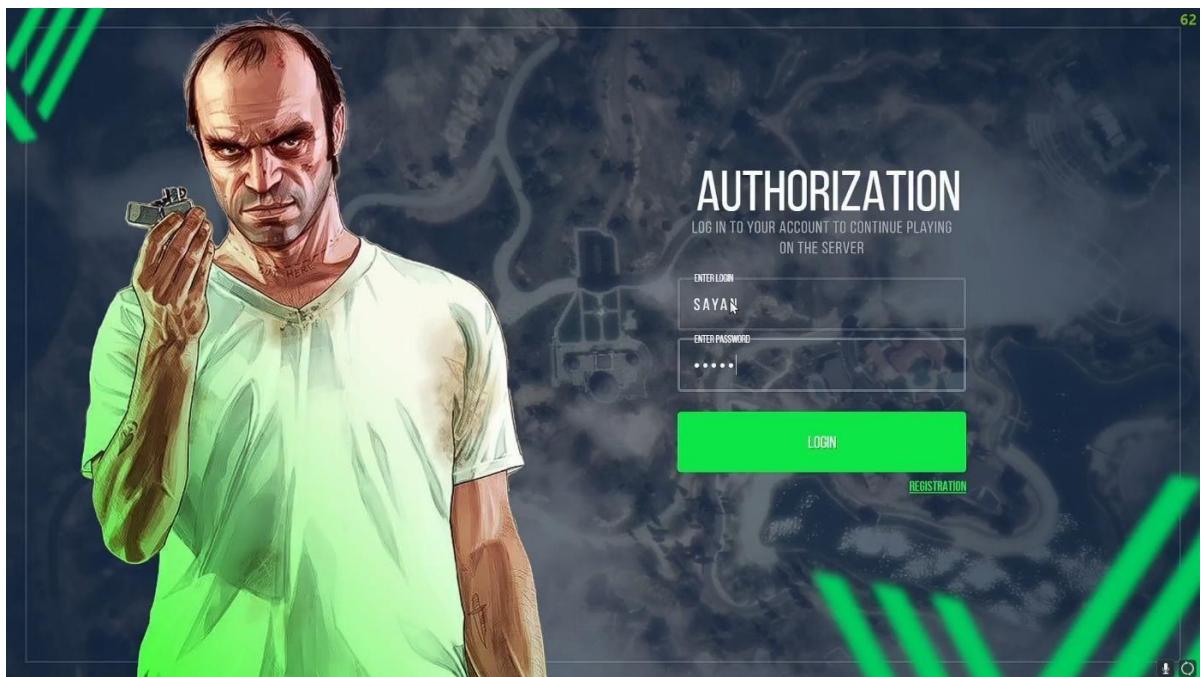
```
1 var petrol = new Vue({
2   el: ".petrol",
3   data: {
4     active: false,
5     price: 45,
6     input: "",
7     inputelectro: 0,
8     street: "Vinewood Hills",
9     fuel: "gasoline",
10    style: "gasoline",
11    maxfuel: 120,
12    multi: 1.15,
13    btns: [false,true,false,false,fals...],
14    regular: false,
15    premium: false,
16    diesel: false,
17    card: false,
18  },
19  methods: {
20    gstyle: function(index) {
21      this.style = index
22    },
23    gov: function () {
24      //console.log("full")
25      mp.trigger('petrol.gov', this.diesel, this.regular, this.premium)
26    },
27    full: function () {
28      //console.log("full")
29      mp.trigger('petrol.full', this.diesel, this.regular, this.premium)
30    },
31    yes: function () {
32      console.log(this.input * this.multi)
33      mp.trigger('petrol', this.input * this.multi, this.diesel, this.regular, this.premium)
34    },
35    no: function () {
36      //console.log("no")
37      mp.trigger('closePetrol')
38    },
39    cardbn: function() {
40      if (this.card == 'false') {
41        this.card = true;
42      }
43      else if (this.card == 'true') {
44        this.card = false;
45      }
46      console.log(this.card);
47    },
48    checkbox: function(id){
49      let ind = this.btns.indexOf(true);
50      if (ind > -1) this.btns[ind] = false;
51      this.btns[id] = true;
52      this.multi=id;
53      this.active=false;
54      this.active=true;
55      if (id == -1) {
56        this.regular = true;
57      }
58    }
59  }
60})
```

Petrol pump

B. ScreenShots



Validating server files



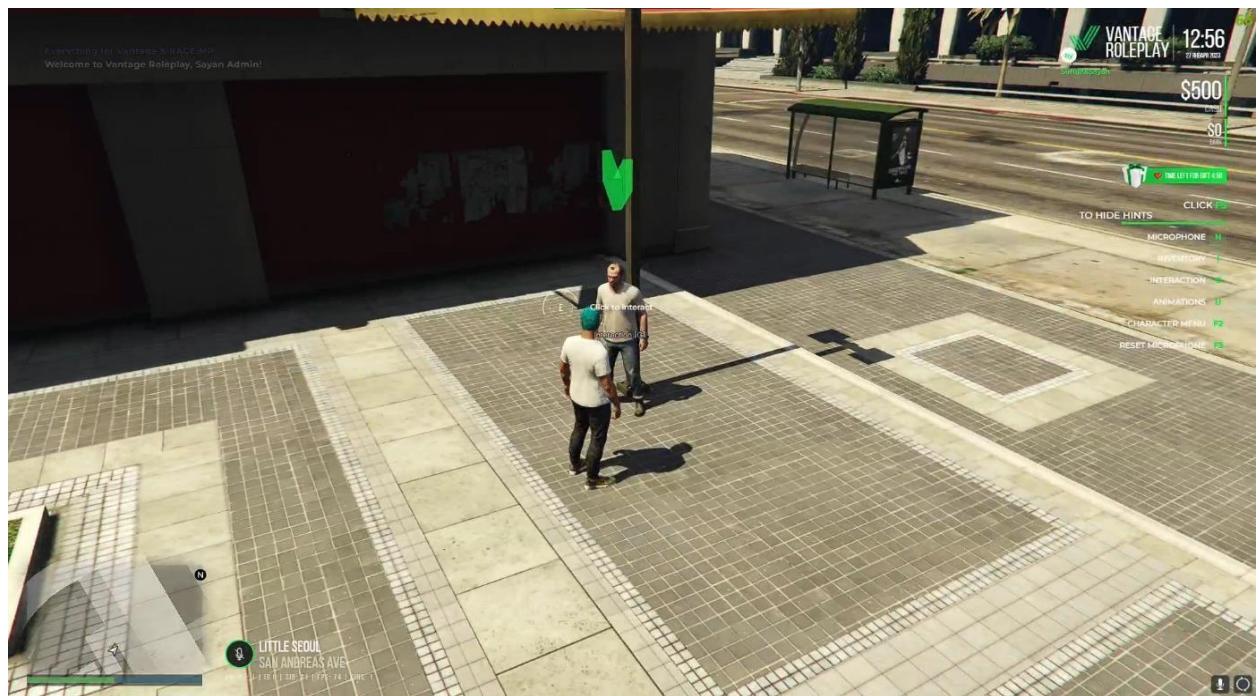
Authentication page



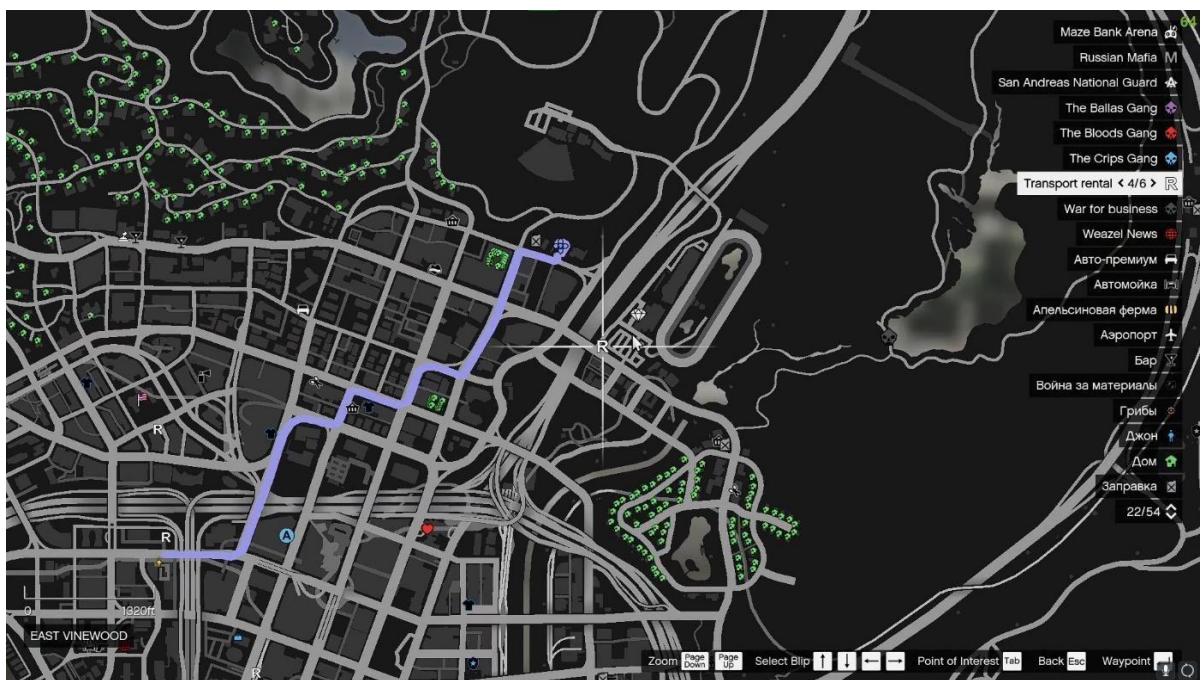
Character creation page



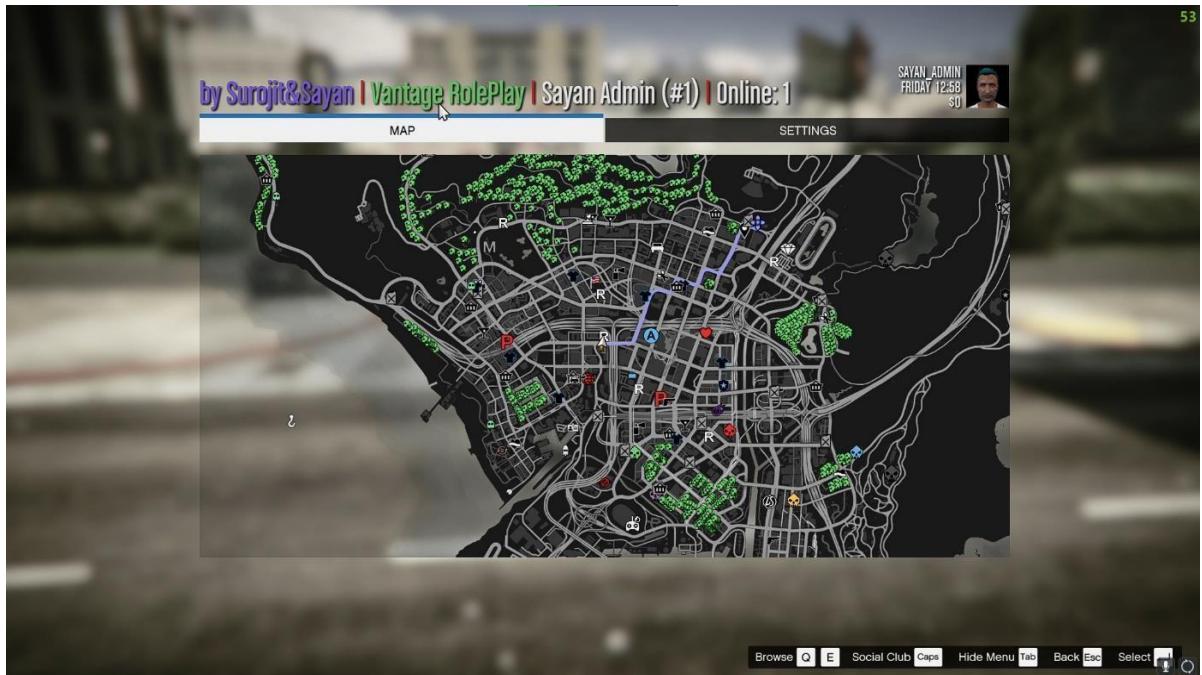
Game environment



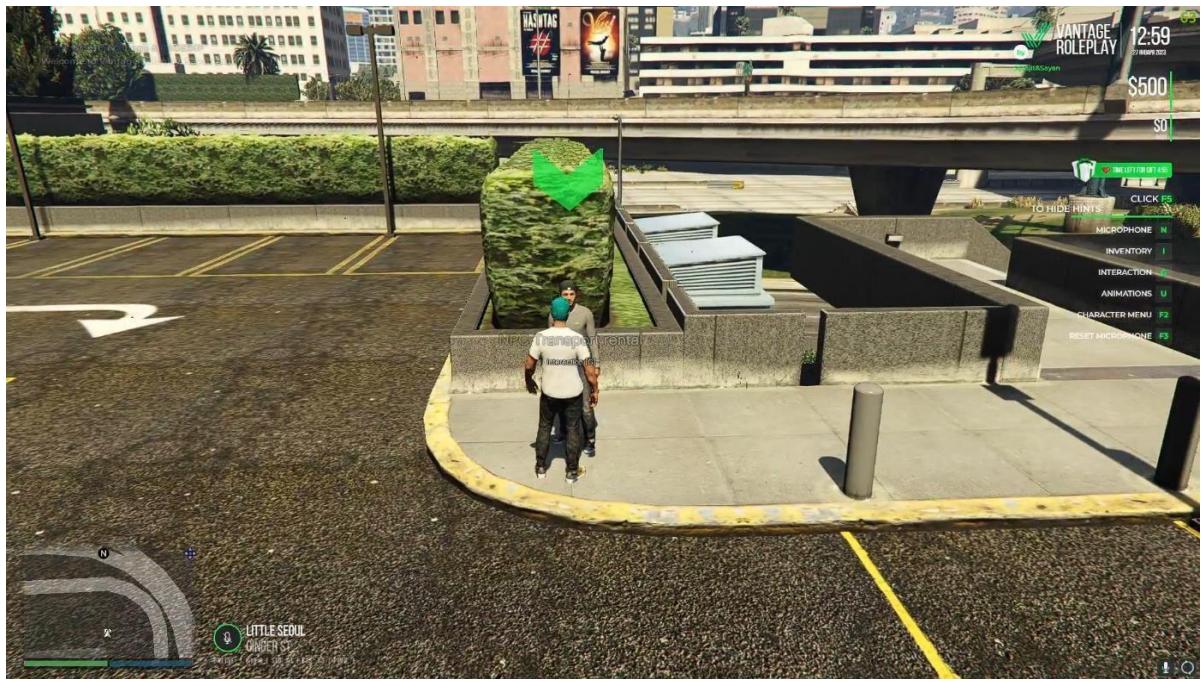
1st NPC interaction



World environment map



Map loading menu



Transportation in NPC interaction

C. RESEACH PAPER

Server Development: GTAV-RP

Sayan Pramanik¹

Department of Computer Science
and Engineering,
Sathyabama Institute of
Science and Technology,
Chennai, India.
sayanpramanik2012@gmail.c
om

Dr. A. Mary Posonia²

Department of Computer Science
and Engineering, Sathyabama
Institute of Science and
Technology, Chennai, India.
maryposonia.cse@sathyabama.
ac.in

ABSTRACT—In this server development we have obtained a position where we have taken the Game GTAV as a model and developed a server on its bases of a Roleplay on a community server called as Rage Multiplayer. Here all players come into interaction and communicates to every other player via voice chat or in game-rp chat. All RP chat is monitored by the Server admins continuously. This server will have a capacity to handle more than 5000 players simultaneously with an active radius of 500. Here all players have their data stored on the Database of the server and players are limited to the number of devices they can login with their legit copy of their GTAV original game. This RP server is coded on language such as C#, and for database handling we used My SQL Server as the platform.

Keywords: GTAV-Roleplay, Server, Voice Communication, Player entities, DDos-attack prevention.

I. INTRODUCTION

There are 2 main players in the Server development technique for the GTAV Roleplay Server Hosting and development the one come with all preinstalled themes for all the UI and then the server consoles such as “txAdmin” [3] which provides all the facility to obtain the output but sometime not the desired one so we have chosen to develop the server on the Rage Multiplayer which enables full customization of the server starting from scratch to UI to game console, etc. RAGE Multiplayer is an initiative started out through an experienced crew to offer an opportunity, exciting, and strong multiplayer revel in for Grand Theft Auto V. Commenced in May 2016, RAGE remains a completely delicate challenge with splendid ability noticed by using many. Our sole focus is to mold this project in a self-sustaining community that promotes creativity and originality. Thanks to the C++ foundation, Rage doesn't depend on any outside documents or programs, since it's built with C++. RAGE is a complete, stand-alone client with great functionality. The team makes a specialty of continuous overall performance and stability development whilst simultaneously adding new functions and expanding the API.

In this project we were able to achieve the position to allow the players to join the server and connect and interact with the game entities, admins can also interact with e players fly around and can span cars admins are also set to different levels where deferent level admis are given different roles to perform and admin with higher level can perform action to ban players and remove ban from players. There is also a defined console which helps to track the players movement with will help to trace out any unwanted movement. Plugins is a folder which lets in you to load custom “.dll” plugins into the server, conf.json contains the server configuration in JSON format. Ragemp-server.exe is a server executable, which if you run, starts the game server.

Name	Date modified	Type	Size
access	01-06-2022 18:36	File folder	
top	09-06-2022 15:10	File folder	
clouds	31-05-2022 10:54	File folder	
ments	20-10-2022 20:24	File folder	
res	07-06-2022 23:27	File folder	
ase	07-06-2022 23:29	File folder	
ps	31-05-2022 20:04	File folder	
rMode	09-06-2022 11:14	File folder	
rgemp	23-11-2021 21:22	File folder	
ive	03-12-2021 17:12	File folder	
;	23-10-2021 20:45	File folder	
;	28-04-2021 04:34	Application exten...	35 KB
objects	28-10-2022 21:30	JSON File	1 KB
top	02-06-2022 19:48	Text Document	1 KB
ments	28-04-2021 04:34	Windows Batch File	1 KB
clouds	08-06-2021 22:37	Application	35,500 KB
debug	14-05-2022 23:12	Text Document	1 KB
ragemp-server			
readme			

Fig.1. Server End System files

II. LITERATURE SURVEY

There are few existing systems based on the RP server on GTAV based but those are either based on the low-level RP experienced or have lot of delay on addressing player problems. Rage Multiplayer is a modification for a popular open world game set in a fictional city, which is based on Los Angeles. As an Open World game, the player is free to do in the world as he likes. Rage MP is a multiplayer mod based on servers hosted by the players themselves. What's special: Rage MP allows an unprecedented number of players. Via ZAP- Hosting you can rent servers with up to 1000 slots. Despite this high number of players, Rage MP runs with high performance and without FPS loss. In addition, Rage MP comes with its own API and is of course fully modifiable. A game like Rage MP is of course the most fun with your own friends or in your own clan. You can do what you want on your own server and leave it as it is. And of course, you can also make your own RP server with it. At ZAP-Hosting you can now also rent Rage MP servers. All our servers are of course fully DDoS protected [13], easy to use via our web interface and offer full mod support thanks to free FTP access. There are multiple servers on Roleplay for GTAV. This problem to handle over 2000 players simultaneously is being solved by this project. Due to less availability of servers and constrain system requirements the server goes into queue and players needs to wait for join the server. Hosting Cloud server and linking to Rage Multiplayer.

Some problem on existing systems is Due to less availability of servers and constrain system requirements the server goes into queue [5] and players needs to wait for join the server. Hosting Cloud server and linking to Rage Multiplayer, all the client packages required for the user to download in order to connect to the server, without downloading the server assets the client cannot connect to the server.

III. RESEARCH METHODOLOGY

In the Figure 2 we could see the system architecture of the process where the user makes an interaction with the Server after getting verified by the firewall it can be understood by reading. Admin has got the full control over the system. All the data here are encrypted and

password protected so it becomes difficult for the attackers to attack our machine files.

The admin also is bounded to different levels which provides them different access levels. Admin server is directly connected to the database server hosted on that machine itself, admin have the right to edit and modify the data in the database only at a specific position.

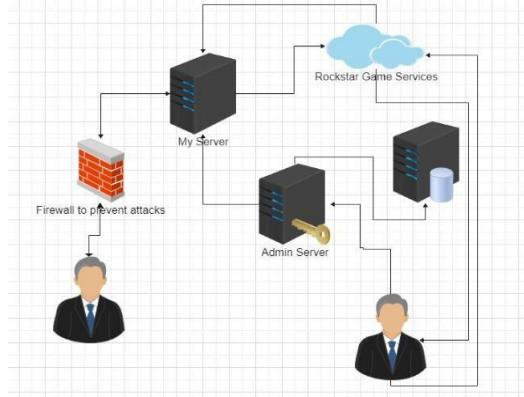


Fig.2. Server Architecture

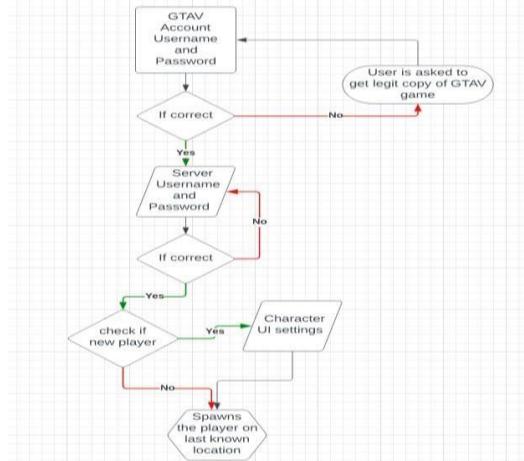


Fig.3. Front-end Ideation Map

In the Figure 3 we could [1] see the block diagram of the proposed system. Here the user gets into the server and all the rest processes is given in the ideation map starting from user interaction to admin interaction. Once the player is spawning the activities of the player changed in respected to every player. In the next ideation map for the backend. Here the player activities are all. Saved into the database simultaneously Admin have different levels of access. Higher level admin has access to change in database directly.

In the backend [15] of the machine the user enters the server by entering the server username and password and a link is created to interact with the database. If the player is new a character making

screen is shown to the player is new a character making screen is shown to the player or the player will be spawned in the world and any activities of his choose can be done.

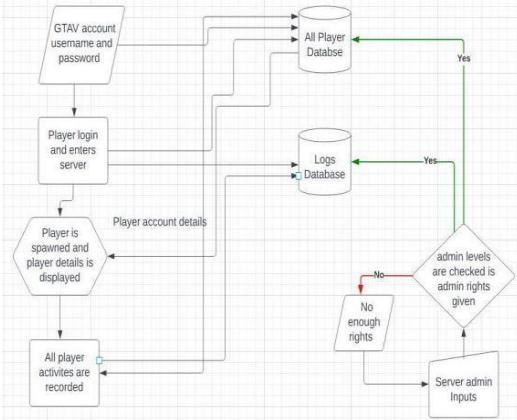


Fig.4. Back-end Ideation

In the Figure 5 we can determine the disk usage by the files which together hold up to 7GB in initial stage which includes DLCs, network configuration over RAGE, Client data this data capacity can change depending on the further development.

Content type	Disk Usage
DLCs	5
Client Data	2
Database	3
Network	2

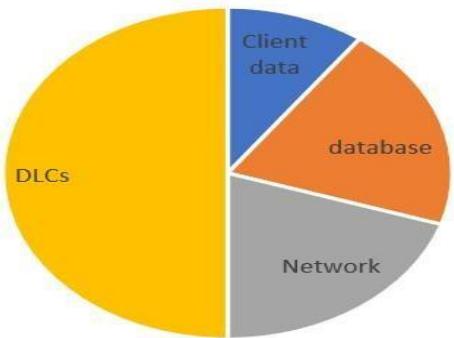


Fig.5. Server-Disk Usage

All activities which affect the economy of the world, or the experience of the player are saved in database and the steps performed to achieve that is saved in the logs, the admin has a major role here to look after the logs any skipping of the steps to achieve the certain set then the admin has the right to freeze, kick, ban the player from the server.

Some coins in the server can be earned by doing activities in the world itself and some can be done by Doing transaction which is bought to the main way of earning by the creators. [2] The payment gateway will be secured by razor pay as that offers the secure payment gateway and it secures the payment in game and its directly linked to the database, if the payment gets successful then the value gets added into the database.

IV. RESULT

In this project, I obtained a result where a user can make an account on the server which will automatically get linked to the rockstar game service and login to his account, their password is encrypted and saved in database which can neither be changed or retrieved by the admin nor the owner. This server enables the client or the player to make 3 character which helps in preventing the player to make multiple accounts based on 1 rockstar game ID.



Fig.6.Login Page for user

Also, the secured payment system helps the player pay to the game server to buy in game currency and use them in the RP world, this in game payment method will help to run the server maintenance and to hire and upgrade security system of the machine to be prevented from DDOS attacks.



Fig.7. Player spans in game

At the end [12] we have achieved a position to provide crate opening, car spawning with

different DLCs and set location of various house and entities. Set price of different house to XYZ coordinates

Fig8 shows the best 500 render distance is best for this [10] kind of server for optimum response time of 5.2ms

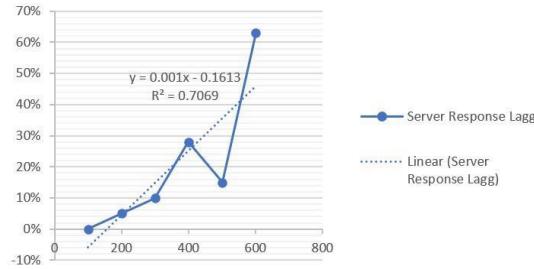


Fig.8. Stream-Distance vs server Response

```

12:52:55.749 [Info] | Entity-System => Long-Region has been saved to DB
12:52:55.749 [Info] | EntityManager => Check Water.
12:52:57.455 [Info] | HouseManager => Houses has been saved to DB
12:52:57.706 [Info] | HouseFurniture => Furnitures has been saved to DB
12:52:57.706 [Info] | HouseEntity => Every Moving Items...
12:52:57.710 [Info] | StockManager => Stocks has been saved to DB
[4] Incoming connection from 127.0.0.1
12:54:10.666 [Info] | GM => WeirdWeebie try to signin step 1
12:54:10.666 [Info] | GM => WeirdWeebie try to signin step 2
12:54:11.815 [Info] | GM => WeirdWeebie try to signin step 3
12:54:22.429 [Info] | GM => WeirdWeebie select char
12:54:29.590 [Info] | BankSystem => Created new Bank Account! ID:78697
12:55:03.469 [Info] | GM => Playing
12:55:03.469 [Info] | BusinessManager => BustHouses has been saved to DB
12:55:55.448 [Info] | GangCapture => Gang Regions has been saved to DB
12:55:55.448 [Info] | HouseManager => Houses has been saved to DB
12:55:58.447 [Info] | HouseFurniture => Furniture has been saved to DB
12:55:58.447 [Info] | HouseEntity => Every Moving Items...
12:55:58.448 [Info] | HouseEntity => Saving Items...

```

Result Grid	Filter Rows	Edit	Print	Report/Imports	Wrap Cell Contents	Character	Result Grid	Form
result.xls	Logon	hwd	refid.xls	b	character1	character	Result Grid	Form
TOTAL	sayan	80761247ECA4D584D69737021F00166E1...	493000	105.115.240.128	333333	-1		
e_mor_06	sayan	EDC95000A8E9E1D8E9ECD10E17667AA0E3C...	19999998	105.77.194.66	509316	-1		
•	sayan							

Fig.9. People getting into server

VI. CONCLUSION

Giving description of my work done and taking the samples of the available works, we could say that the system architecture and the system proposed is best in the present environment and can be used in the next coming years. This system has the capability [11] to handle more than 10000 players and record all the user activities on every 3 min basis and not take huge load resulting in freezing of servers.

REFERENCES

- [1] M. Chawla, K. Singh and C. Kumar, "Attitudinal data based server job scheduling using genetic algorithms: Client-centric job scheduling for single threaded servers," 2016 Ninth International Conference on Contemporary Computing (IC3), Noida, India, 2016, pp. 1-7, doi: 10.1109/IC3.2016.7880230.
- [2] M. Hisyam and I. B. K. Manuaba, "Integration Model of Multiple Payment Gateways for Online Split Payment Scenario,"

2022 International Conference on Information Management and Technology (ICIMTech), Semarang, Indonesia, 2022, pp. 122-126, doi: 10.1109/ICIMTech55957.2022.9915168.

[3] J. Yang, J. Han, Y. Xing, Y. Zhang, W. Wei and K. Xue, "SSMP: Server Selection for Multipath TCP in CDN Environments," GLOBECOM 2020 - 2020 IEEE Global Communications Conference, Taipei, Taiwan, 2020, pp. 1-6, doi: 10.1109/GLOBECOM42002.2020.9348043.

[4] S. Segawa, H. Masuda and M. Mori, "Proposal and Prototype of DNS Server Firewall with Flexible Response Control Mechanism," 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Toyama, Japan, 2019, pp. 466-471, doi: 10.1109/SNPD.2019.8935681.

[5] Mary Posonia A, Ankayarkanni B, Usha Nandhini D, Albert Mayan J, Nagarajan G (2022), "An Efficient Algorithm for Traffic Congestion Control", Advances in Intelligent Computing and Communication, Lecture Notes in Networks and Systems, Vol 430, Springer.

[6] X. Han, X. Meng, D. Zhai and G. Chen, "A Resilient Response Method of DDoS Attack Based on TCP Proxy," 2021 2nd International Conference on Computer Engineering and Intelligent Control (ICCEIC), Chongqing, China, 2021, pp. 46-49, doi: 10.1109/ICCEIC54227.2021.00017.

[7] Z. Zeng and B. Veeravalli, "On the Design of Distributed Object Placement and Load Balancing Strategies in Large-Scale Networked Multimedia Storage Systems," in IEEE Transactions on Knowledge and Data Engineering, vol. 20, no. 3, pp. 369-382, March 2008, doi: 10.1109/TKDE.2007.190694.

[8] A. Rost, B. Venkatesh and C. P. Diduch, "Distribtuion System with Distributed Generation Load Flow," 2006 Large Engineering Systems Conference on Power Engineering, Halifax, NS, Canada, 2006, pp. 55-60, doi: 10.1109/LESCPE.2006.280360.

[9] Joshila Grace L.K,Vigneshwari S,Sathyra Bama Krishna R,Ankayarkanni B,Mary Posonia A,"A Joint Optimization Approach for Security and Insurance Management on the Cloud", Lecture Notes in Networks and Systems,Vol.430, pp. 405–413..

[10] S. C. Deshmukh and S. S. Deshmukh, "Improved load balancing for distributed file system using self acting and adaptive loading

- data migration process," 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Noida, India, 2015, pp. 1-6, doi: 10.1109/ICRITO.2015.7359283.
- [11] R. T. Bhimarasetti and A. Kumar, "Distributed generation placement in unbalanced distribution system with seasonal load variation," 2014 Eighteenth National Power Systems Conference (NPSC), Guwahati, India, 2014, pp. 1-5, doi: 10.1109/NPSC.2014.7103786.
- [12] Z. Zhang and C. Lu, "Research and implementation for data synchronization of heterogeneous databases," 2010 International Conference on Computer and Communication Technologies in Agriculture Engineering, Chengdu, China, 2010, pp. 464-466, doi: 10.1109/CCTAE.2010.5544320.
- [13] O. B. Al-Khurafi and M. A. Al-Ahmad, "Survey of Web Application Vulnerability Attacks," 2015 4th International Conference on Advanced Computer Science Applications and Technologies (ACSAT), Kuala Lumpur, Malaysia, 2015, pp. 154-158, doi: 10.1109/ACSAT.2015.46.
- [14] H. Kitakami et al., "Building and search system for a large-scale DNA database," IEE Colloquium on Molecular Bioinformatics, London, UK, 1994, pp. 6/1-6/9.
- [15] P. Yang, M. Wu, Y. Liu and S. Yao, "Research and Development of Embedded EIB ARM Linux Server Gateway," 2006 4th IEEE International Conference on Industrial Informatics, Singapore, 2006, pp. 1375-1379, doi: 10.1109/INDIN.2006.275616.
- [16] Z. Wang et al., "Detecting Malicious Server Based on Server-to-Server Realation Graph," 2016 IEEE First International Conference on Data Science in Cyberspace (DSC), Changsha, China, 2016, pp. 698-702, doi: 10.1109/DSC.2016.79.
- [17] A. Riva, R. Bellazzi and S. Montani, "A knowledge-based Web server as a development environment for Web-based knowledge servers," IEE Colloquium on Web-Based Knowledge Servers (Digest No. 1998/307), London, UK, 1998, pp. 5/1-5/5, doi: 10.1049/ic:19980533.
- [18] H. Yanagisawa, "Extension of Web-Based Software Development Environment," 2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, Gwangju, Korea (South), 2015, pp. 463-466, doi: 10.1109/WAINA.2015.117.
- [19] Suthir S, S. Janakiraman, M. Srividya and N. Anusha, "A contemporary network security technique using smokescreen SSL in huddle network server," 2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), Chennai, India, 2016, pp. 673-676, doi: 10.1109/AEEICB.2016.7538376.
- [20] Z. Lian et al., "Design of Communication System of Underwater Hybrid Voice Communication System," 2021 IEEE International Conference on Electronic Technology, Communication and Information (ICETCI), Changchun, China, 2021, pp. 325-329, doi: 10.1109/ICETCI53161.2021.9563507.