

Design and Development of OMPS Game

N. Thirupathi Rao, Debnath Bhattacharyya, Tai-hoon Kim

Abstract: *The current Model “Online Multiplayer Strategy Game” is an online multiplayer strategy game which is developed as a web application. Each player starts the game as the leader of a small undeveloped village, surrounded by undeveloped resource fields. Creating military units will allow them to attack a person or defend from enemy attacks. Players can join as a team. Allies may trade resources through trade market or send reinforcements when others are being attacked. An alliance can win the game by destroying the enemy alliance completely. The main motive of the game is to gather resources with villagers, developing village by constructing new buildings, creating as many troops as you can and destroying the enemy alliance with your alliance. The current Model is developed using mean stack. A standard java stack called MEAN is used for designing and building the dynamic web pages. Also the same software is used for MongoDB and other sources etc.,. The current work sparks the player’s creativity, develops problem solving skills, and improves one’s planning, management and foresight. The game is portable and can be accessed from anywhere.*

Keywords: *Online game, multiplayer’s, villagers, strategy and game.*

I. INTRODUCTION TO BROWSER GAMES

Browser games are getting famous day by day and also the games which were developed targeting mobile devices also getting day to day [1]. Browser games are those users can use the personal computers to play the games. The development of such games includes the usage of various web technologies and other technologies for further more interactive with the end users. These games consist of all types of various video games and other types of related games. These games include the combination of both single players and sometimes double players and in some games more than two players also possible to play the games [2]. Program diversions are regularly allowed to-play and don't require any customer programming to be introduced separated from an internet browser or program module. Now and again an amusement might be free, however charge for additional in-diversion highlights. Multiplayer program recreations have an extra spotlight on social association, either between a few players or on a huge scale. Because of the availability of program diversions, they are regularly played in progressively visit, shorter sessions contrasted with conventional PC recreations [3,4]. Since program diversions run separated from equipment in an internet browser, they can keep running on a wide range of working frameworks without being ported to every stage. In a diversion Model the item is amusement, which is substantially more than just programming [5]. It needs to give agreeable substance.

Revised Manuscript Received on May 07, 2019.

N. Thirupathi Rao, Department of Computer Science and Engineering Vignan’s Institute of Information Technology (A) Visakhapatnam-530 049, AP, India.

Debnath Bhattacharyya, Department of Computer Science and Engineering Vignan’s Institute of Information Technology (A) Visakhapatnam-530 049, AP, India.

Tai-hoon Kim, Department of Computing and IT, University of Tasmania, Churchill Ave, Hobart TAS 7005, Australia

The nature of the diversion likewise has a noteworthy job in its notoriety. Standard web innovations, for example, HTML, CSS, PHP, and JavaScript can be utilized to make program diversions, yet these have had constrained achievement on account of issues with program similarity and quality. These advancements take into consideration amusements that can be kept running in all guidelines consistent programs. Furthermore, committed designs advances, for example, SVG and canvas take into consideration the quick rendering of vector and raster illustrations separately. Also, WebGL takes into consideration equipment quickened 3D support in the program. Program modules were utilized to give diversion advances subsequent to being introduced by the client. Starting at 2017 most organizations (Oracle for Java module, Adobe for Flash Plug-in) are thinking about to end support for their modules. Additionally internet browser makers are leaving utilizing modules in their items later on. Some examples of such browser based multiplayer games are as follows,

A. Castle Of Heroes

Castle of Heroes, is an enormously multiplayer online dream program based procedure amusement created by the Chinese designer Snail Games and distributed in the United States by its U.S. distributing division, Snail Games USA [6, 7]. The decision is between Human, Elf, Orc and Undead. Each race has distinctive advantages for a beginning player, for example, people having an early extended unit, or early undead units being less expensive to deliver than different races. Further racial advantages are discovered later in the diversion. In the wake of picking and naming a character players will have the alternative to take a short instructional exercise to acquaint players with the diversion. Errands for the player to finish help to direct the player through the amusement once the instructional exercise is finished [8, 9, 10]. Players begin with a little however shifted supplement of troops and a blue quality Hero to order them. Some of the other set of games are “Empire and State” and “Clash of Clans”. The advantages and disadvantages of these games are discussed in detail in the following sections. The client side programming Languages is JavaScript while the server side programming language is PHP [11, 12, 13]. The JavaScript Library used is jQuery 1.8.3 and Modernizr with HTML5 as the mark up language.

II. OBSERVATIONS FROM THE EXISTING MODELS

1. “Age of Empires” uses PHP and ASP.NET for server side scripting and jQuery javascript Library for client side programming.



Design and Development of OMPS Game

- The Web Hosting provider for Age of Empires is Microsoft.
- Age of Empires has 16.6 % of visitors from US, 13.6% from India and 6.6% from France
- The major drawback in Age of Empires is the limitation on the number of villagers and limitation on the number of troops.
- Travian's first release won the "super browser game" award in 2006
- Travian uses PHP for server side scripting and javascript Library for client side programming.
- The major drawback in Travian is that the production cannot be changed

A. Problem Statement

All the existing models have certain limitations on the number of villagers that can be added to the game and number of troops that can be created with the help of the villagers. Also, the production cannot be changed in these existing models. Also, we are adding additional features like "Watch Tower" to make this more interesting. Watch tower is a building which helps to locate the enemies at a particular distance based on the level of watch tower which can be upgraded.

B. Advantages of Proposed System

- Unlimited number of villagers.
- Unlimited number of troops.
- Production can be changed.
- Comes with the new feature of "Watch Tower" methodology model

III. MODULES IN THE MODEL

The modules involved in this Model are

- Interface Designing
- Design Implementation
- Server side Scripting
- Database Development

A. Interface Designing

User interface design (UI) or user interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing usability and the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design).

B. Server Side Scripting

A database is a collection of information that is organized so that it can be easily accessed, managed and updated. This will include the specification of an appropriate storage schema, security enforcement, and external schema and so on. Implementation is heavily influenced by the choice of available DBMSs, database tools and operating environment.

IV. Implementation, Results and Discussions

The results that were arrived from after implementation are discussed with each screen shot in the following figures and tables.

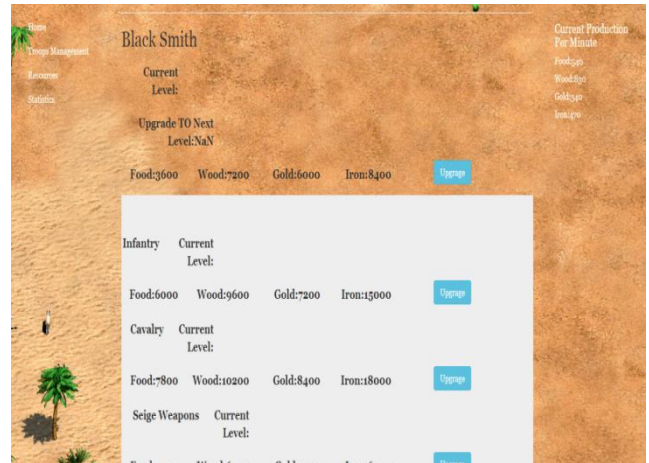


Fig 1. Blacksmith

This is displayed whenever the user clicks on the blacksmith building.

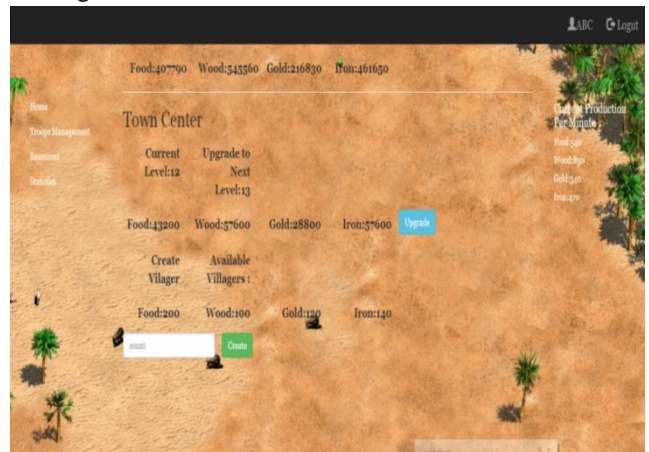


Fig 2. Town Center

This is displayed whenever the user clicks on the town center building.

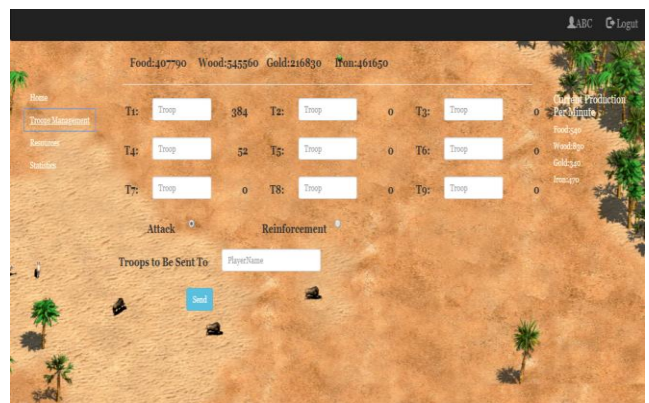


Fig 3. Manage Troops

This is used to manage troops. User can send troops to other members in the game



Fig 4. Resources

These are the different resources available in the game. Every resource has a particular production per minute.

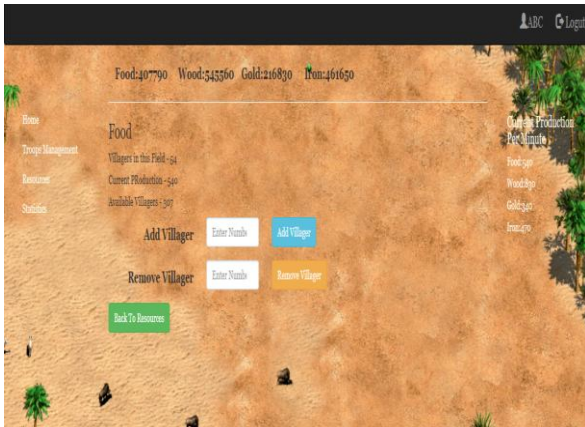


Fig 5. Manage Villagers

This is for managing the number of villagers. User can add or remove villagers from a particular field.

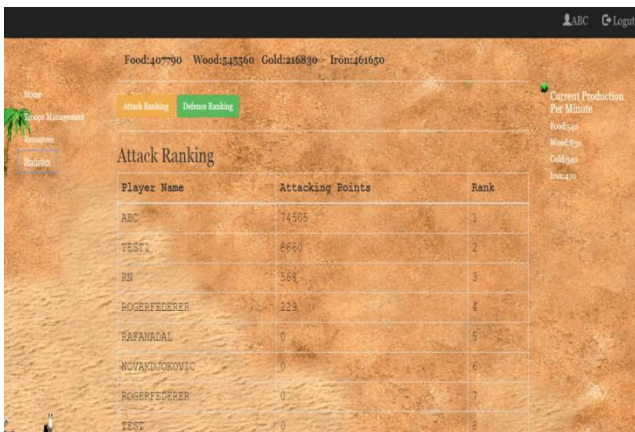


Fig 6. Attack Ranking

This gives the attack ranking of the user.

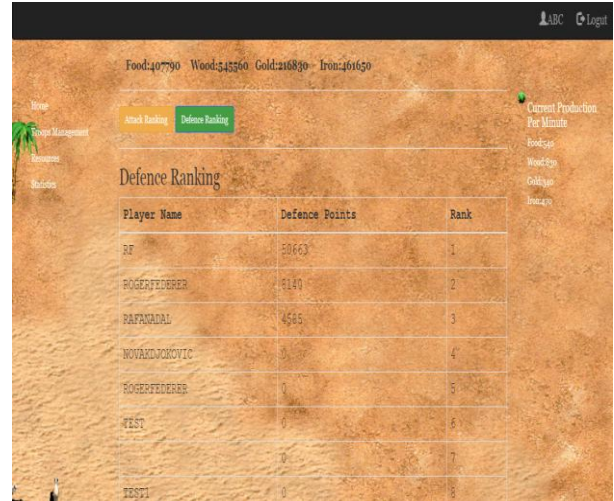


Fig 7. Defense Ranking

This displays the defence ranking of the user. Before adding villagers

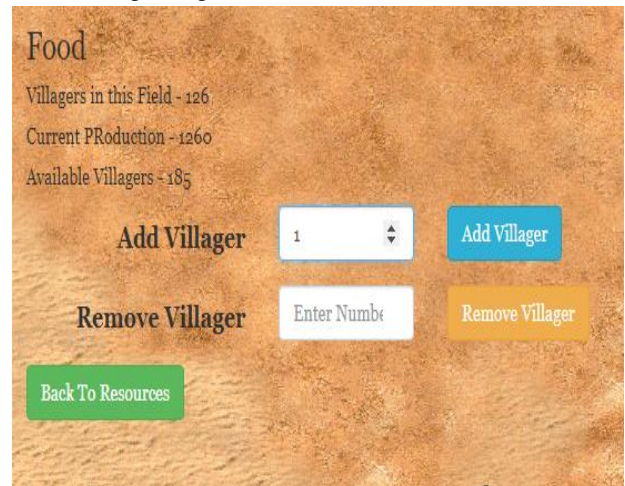


Fig 8. Before Adding Villagers

Adding Villagers

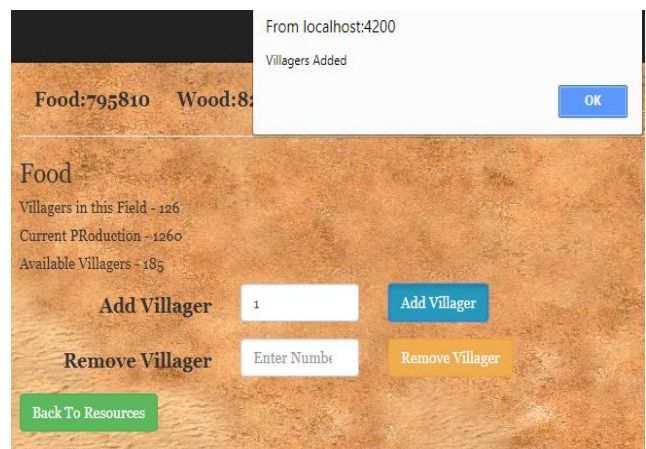


Fig 9. After Adding Villagers

After Adding Villagers



Fig 10. After Adding Villagers

The performance of the currently developed online strategy was tested in various cases. Several cases considered are like the login of the user to the game, login of several users at a time, checking of the correctness of the username and password of the users with less amount of time and other issues were tested and verified with various scenarios under the actual processing of the game. These scenarios are considered under the actual number of users or the actual number of players playing the game. But, after adding the new set of online players or the new set of players to the existing set of players might have some impact on the performance of the game. The new set of players was added in the form of adding new villagers in the game or the new set of players playing the game. The performance of the online game strategy was observed the performance and the results are studied in the form of execution time. These scenarios are given in the following as tables with various scenarios. The scenarios are given as before adding the number of villagers to the existing game and the performance of the game after adding the new set of users or the new villagers adding to the existing set of villagers. These scenarios can be seen as follows, Before adding the Villagers to the actual number of users in the game,

Table 1. Test Case 1

S.No. of Test Case	1	Execution Time
Name of Test	User Login Success	5 Seconds
Sample Input	Here The user Enter Username, Password For Entering In To His Account	
Expected Output	Displays Message As "user Login Success"	
Actual Output	Same As Expected This Component Clearly Tells That user Is: Login Success	

Table 2. Test Case 2

S.No. of Test Case	2	Execution Time
Name of Test	User Login fails	4 Seconds
Sample Input	Here The user Enter Username, Password For Entering In To His Account	
Expected Output	Displays Message As "user Login failure"	
Actual Output Remarks	Same As Expected This Component Clearly Tells That user Is Login failure	

Table 3. Test Case 3

S.No. of Test Case	3	Execution Time
Name of Test	Assign Villager	3 Seconds
Sample Input	Here the user enters the number of villagers to be assigned	
Expected Output	Displays Message As "Villagers Added"	
Actual Output	Same as expected Output	
Remarks	This component clearly tells that villagers are added successfully	

After adding the new villagers to the existing set of villagers for playing the game and analyzing the performance of the game as follows,

Table 4. Troops Creation

S.No. of Test Case	4	Execution Time
Name of Test	Creating Troops	6 Seconds
Sample Input	Here the user enters the number of troops to be created	
Expected Output	The "Currently Available" troop count should be increased	
Actual Output	Same as expected Output	
Remarks	This component clearly tells that troops are created successfully	

Table 5. Resources for Creation of Troops

S.No. of Test Case	5	Execution Time
Name of Test	Insufficient resources for creating troops	6 Seconds
Sample Input	Here the user enters the number of troops to be created	
Expected Output	Displays Message as "not Enough Resources"	
Actual Output	Same as expected Output	
Remarks	This component clearly tells that the troops are not created with insufficient number of resources.	

Table 6. Sending Troops to a User



S.No. of Test Case	6	Execution Time
Name of Test	Send troops to a valid user	7 Seconds
Sample Input	Here the user enters the name of the user to whom the troops are to be sent	
Expected Output	Displays Message as "Attack has been sent"	
Actual Output	Same as expected Output	
Remarks	This component clearly tells that the attack has been sent successfully	

Table 7. Overall Performance Comparison

S.No	Consolidated Cases	Average Execution Time
1	Before adding new villagers to the game players	5 Seconds
2	After adding the new villagers to the existing game players	6.5 Seconds

From all the above six cases or six scenarios of various cases, it is understood that the execution time for the first three cases have less time than the next level three cases of after adding villagers to the actual number of villagers or players playing the game. The time taking by the machine for implementing or doing such tasks was more than the regular time that was expected. Hence, the execution time for the above six cases can be understood very clear that the time is more when more number of villagers are being added to the existing set of players or with the existing set of game players.

V. CONCLUSION

The current Model "design and development of online multiplayer strategy game" successfully combines all the interesting features of "Age of Empires" and "Travian" and overcomes all the mentioned disadvantages in these two games and successfully passed all the test cases in the testing phase. Unlike "Age of Empires", there are no limitations on the number of villagers and number of troops. The production can be changed easily in the current Model. This also has an additional feature of "Watch Tower" which enables the user to notice enemies from a particular distance depending upon the level of the watch tower. The user can login to access his game status and continue playing the game successfully from any browser provided with an active internet connection. The current Model also sparks the player's creativity, develops problem solving skills, and improves one's planning, management and foresight. The game is portable and can be accessed from anywhere.

REFERENCES

1. Getting MEAN with Mongo, Express, Angular, and Node -Book by Simon Holmes. [Accessed on 10-12-2018]

2. Mean Web development book by Amos Q.Haviv. [Accessed on 10-12-2018]
3. <https://www.mongodb.org>. [Accessed on 10-12-2018]
4. <https://www.nodejs.org> [Accessed on 12-12-2018]
5. <https://npmjs.com/> [Accessed on 11-12-2018]
6. <http://www.tutorialspoint.com> [Accessed on 05-12-2018]
7. <http://www.javatpoint.com> [Accessed on 05-12-2018]
8. [http:// android-developers.blogspot.in](http://android-developers.blogspot.in) [Accessed on 05-12-2018]
9. <https://design.google.com> [Accessed on 05-12-2018]
10. <https://www.google.com/design/spec/material-design> [Accessed on 15-12-2018]
11. <https://www.youtube.com/watch?v=upgjCMHGpwo> [Accessed on 15-12-2018]
12. <https://www.youtube.com/watch?v=EMiU8zACVgA> [Accessed on 15-12-2018]
13. <https://blog.jetbrains.com/webstorm/2014/01/getting-started-with-node-js-in-webstorm/>. [Accessed on 15-12-2018]

AUTHORS PROFILE



Dr N. Thirupathi Rao received PhD (Tech., CSE) from Andhra University, Visakhapatnam, India. Currently, Dr N.Thirupathi Rao associated with Vignans Institute of Information Technology, Visakhapatnam-530049, India as Associate Professor of Computer Science and Engineering of the Institute since the year 2016. His research areas include Communication Networks, Queuing Models, Stochastic Modeling, Image Processing, Pattern Recognition, Bio-Informatics, Evolutionary Computing and Security. He published 45+ research papers in various reputed International Journals and Conferences. He is the member of ACM, IE, CSI, and ISPS.



Dr Debnath Bhattacharyya received PhD (Tech., CSE) from University of Calcutta, Kolkata, India. Currently, Dr Bhattacharyya associated with Vignans Institute of Information Technology, Visakhapatnam-530049, India as Dean R&D of the Institute since the year 2015. His research areas include Image Processing, Pattern Recognition, Bio-Informatics, Computational Biology, Evolutionary Computing and Security. He published 200+ research papers in various reputed International Journals and Conferences. He published six textbooks for Computer Science as well. He is the member of IEEE, ACM, ACM SIGKDD, IAENG, and IACSIT.



Dr Tai-hoon Kim received B.E., and M.E., degrees from Sungkyunkwan University in Korea and PhD degrees from University of Bristol in UK and University of Tasmania in Australia. Now he is working for Department of Computing and IT, University of Tasmania, Churchill Ave, Australia. His primary research areas are security engineering for IT products, IT systems, development processes, and operational environments. He published 400+ research papers in various reputed International Journals and Conferences. He published ten textbooks for Computer Science as well. He is the member of IEEE, ACM, etc.

