Allocation and Sharing of Compute Resources via Social Network based on Social Cloud

MSc Research Project
Cloud Computing

Aniket Agrawal
Student ID: X18140904

School of Computing
National College of Ireland

Supervisor: Prof. Vikas Sahni

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Aniket Agrawal
Student ID:	X18140904
Programme:	Cloud Computing
Year:	2020
Module:	MSc Research Project
Supervisor:	Prof. Vikas Sahni
Submission Due Date:	17/08/2020
Project Title:	Allocation and Sharing of Compute Resources via Social Network based on Social Cloud
Word Count:	5155
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	16th August 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Allocation and Sharing of Compute Resources via Social Network based on Social Cloud

Aniket Agrawal
x18140904
MSc Research Project in Cloud Computing

16th August 2020

Abstract

Social Networking platforms are tremendously changing the way people communicate and share the resources. Online relationship is based on Friend trust relationship factor. The integration of cloud computing with the social network help people to share their resources efficiently and transparently. It is easier for the user to share computing power to their friends via social network. To substantiate this, Social compute cloud is used where resources like CPU, RAM and storage are shared on social networks through friends relationship. In the social cloud, resource owners offer virtualized environment on their resources to the social network. The key finding of this work helps to utilized personal compute resources efficiently and effectively. This paper gives a new direction towards social network using cloud computing infrastructure.

keywords: Cloud Computing, Social Cloud, Social Network, Friend-Trust Relationship

1 Introduction

Cloud Computing is widely used in today's industry for many reasons. It reduces overheads and cost of the consumers by providing infrastructure, platform and software as a service. It helps to deliver on-demand computing resources. Cloud has boosted the computing power in small as well as medium scale business with features like the pay-as-you-go model, where the user pays only for services used in a specific amount of time. Cloud Computing becomes an essential part of the computing world as it had made life easy as anyone can access their data from anywhere in the world. Cloud is everywhere, including phones, tablets and high-tech devices. As the proposal is based on the allocation of personal resources like RAM, CPU and storage to another user by Social Networks, Cloud Computing is playing a crucial role in allocating and de-allocating resources in a real-time environment. The main concern using the cloud is trust and accountability of the user Caton et al. (2014).

This paper introduces Open Source Social Networks (OSSN) with social compute cloud for better utilization of personal computing resources. Social Cloud is a cloud-based sharing platform which dynamically allocates resources into social networks. The concept of Volunteering Computing is implemented for donating user compute powers to other users who required higher configuration to perform the task. Volunteering to compute helps to share unused compute power for scientific innovation projects like BOINC and SETI@Home. As there is an under-utilization of compute resources, the concept called Social Compute Cloud is introduced. Social Compute Cloud helps to allocate and de-allocate resources dynamically.

Nowadays, Social Networking is emerging as the sharing platform where millions of users share various type of resources such as file, music, videos, etc. According to Gartner peer insights of 2019, there are more than 2.6 Billion active users on social networking website like Facebook. On the other hand, Cloud Computing is useful to share various kind of resources on social platforms. The integration of social networking platform with cloud computing is introduced in this project. Virtualization plays an essential role in sharing the Physical machine to a virtual machine. This virtual machine can be shared on social platforms using the Personal Computer Over Internet Protocol (PCoIP). Kouril and Lambertova (2010). For this project, VMware Workstation software Huber et al. (2011) is used, which helps to virtualized resources. User can share their resources by directly virtualizing physical machine into a virtual machine or can create another virtual machine with desire configuration and OS image.

While sharing resources online with friends on social platforms, there is always a security risk which could affect the host machine. To minimize this risk, Friend Trust Relationship approach has been introduced. This model allows user to share their resources to trusted friends. There are various players in the market which provides a different kind of cloud services. To host the social network and its database and storage, Microsoft Azure services are used. User can also save their data using Azure Blob Storage services which can be integrated with the host machine.

1.1 Relationship between Social Networking platforms and Cloud Computing

With the immense amount of data generated every minute on social platforms, it is difficult to manage the data. Cloud computing plays a vital role to manage all the data and provides it when demanded. Cloud Computing offers services which are affordable to the user. Services like Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS) are useful for deploying servers, software, etc. As this project is based on the allocation of computing resources to another user via social networking platform, Cloud services help to manage the traffic flow of the system and dynamically allocate the resources to the user when needed.

1.2 Social Network OSSN

OSSN is an Open Source Social Networking platform where the developer can build their own social platform using PHP language. OSSN support various types of component which is used for enhancement in user social experience. This Component acts as a service for the user, which is written in PHP. OSSN allows cloud implementation with third-party web-application integration. For the research project, OSSN helps to integrate the

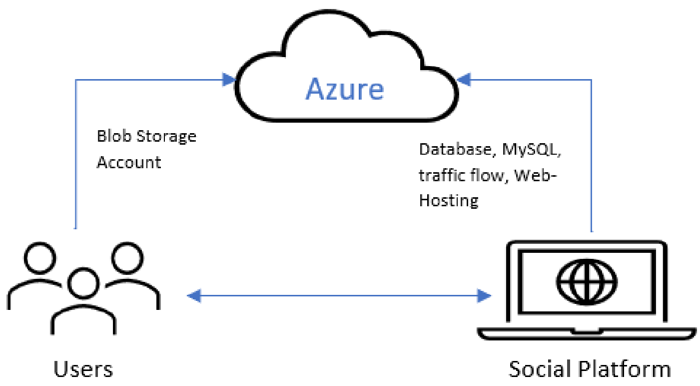


Figure 1: Social Cloud

BOINC server with the cloud platform. Also, OSSN help to connect Personal Computer over Internet Protocol (PCoIP) which is used to share virtualized system over the network.

1.3 Research Question

What is the most significant method to utilise 100% hardware in the personal system that can be deployed in the cloud using the social networking platform?

1.4 Objective and Motivation

The main objective is to increase the utilization rate of the personal computer and to reduce the wastage of computing power. According to the survey, only 15% of the total compute power is utilized in a personal system; rest 85% is not used. This paper present an idea to utilize maximum CPU power in a personal system using social cloud approach. It aims to provide a technique through which the utilization rate can be increased.

1.5 Report Structure

This report consists of the following sections: Section 2 has Related work. Section 3 explains the Research Methodology. In section 4, Design Specification is explained. Section 5 consist of the Implementation followed by section 6, which consist of Evaluation. Section 7 Concludes the research and shows the scope of Future work.

2 Related Work

This section presents all the related work paper review with the critical comparison, drawbacks, issues, future scope. All the relevant work by authors is divided and reviewed based on advantages, shortcomings, problems which helps in proposal study. There are several examples of integrating Cloud Computing with a social networking platform. Nowadays, most of the social networking sites use cloud platform to host their application for scalability and reliability. For Example, Facebook uses Amazon Web Services (AWS) to build a highly scalable cloud-based application.

2.1 Friend Trust Relationship Model

While reviewing numerous papers on the social cloud, the most relevant paper signifies the friend-trust relationship model Guo et al. (2014). Online Social Network (OSN) users share millions of resources online to their friends. This social relationship is based on the 1-hop trust relationship factor. In paper Guo et al. (2014) authors give multi-hop trust chain factor to share resources more securely and safely. Still, privacy concerns can occur if resources are shared with strangers. To overcome this, the authors proposed a trust-based privacy-preserving friend recommendation scheme. In this scheme, OSN users can apply various filters to find friends and establish a social relationship to share resources. Author Kumar et al. (2010) use OSN close friend factor to establish an anonymous communication channel. They have used trust level derivation to enable strangers to obtain a secure connection on the bases on the trust chain. This paper is very niche for the project as it gives various factors to secure social networks resource sharing methods.

2.2 Virtualization Techniques

Cloud Computing is growing exponentially, the need for virtualization resources is increasing. Cloud Computing provides virtualized services to consumers. In Singh (2018) paper author explained the need for virtualization in cloud computing for better utilization of resources. Paper gives a broad perspective that enables the user to share their resources on the internet and intranet. The most transparent way of accessing this virtual resources is using Remote Desktop Protocol (RDP) Casanova et al. (2017) access. RDP helps to take the remote access of the system fast and securely. In paper Singh (2018), the author suggests various ways to virtualize the compute resources. Using VMware, virtualization can be use to transfer physical to the virtual machine state where user physical machine acts as a virtual machine. Another way is to create a virtual machine separately and share it with others using methods like RDP, PCoIP. This study is a niche to research proposal as it is based on sharing resources using virtual layer for more efficiently and securely.

2.3 Volunteering Computing

In today's world, the importance of the social world has increased. Numerous amount of data are being shared on social networks. People voluntarily share data for little or no benefits. Authors McMahon and Milenkovic (2011) proposed architecture for sharing infrastructure resources by Social Volunteer Computing method via the internet through friend-trust relationship. While critically reviewing this paper, it gives the reader a broad perspective of the social world and sharing infrastructure. Authors suggest Volunteering Computing could be the future of the IT world as it could be used to carry more significant operations with no cost on hardware. This paper gives the reader the idea about the distributed computing model where users donate their computing power to another user. Currently, many companies are using volunteering computing for sharing resources. Latest projects like storj Labs Wilkinson et al. (2014) are using Volunteering Computing where user can share their storage like hard-drive to cloud via Dropbox. The focus of volunteering to compute is to utilize 100% personal compute resources.

2.4 BOINC and SETI@Home

As the need for sharing resources keeps on increasing through virtualization and cloud computing, BOINC helps to share resources. BOINC is a Berkeley Open Infrastructure for Network Computing which supports volunteering computing. In paper McGilvary et al. (2013) authors, discussed the need of BOINC server that allow projects with large computational requirements. BOINC is usually used to share resources for scientific purpose. Authors tell about the problem of sharing an existing deployment machine directly to the BOINC. This can cause problem in machine hardware and its configuration. To solve this problem, authors introduce virtual BOINC (V-BOINC) where virtual instances can be created in VMWare, VirtualBox, KVM, in the local system and that virtual instance can be shared to BOINC server. The sharing of the virtual machine over a network is more secure than the sharing of physical resources directly. In paper Javadi et al. (2011) authors gives a broad idea about the availability of the SETI@Home project where tens of thousands personal system are shared. This papers study is essential to the research proposal as it gives the idea to share personal resources using virtualization techniques.

2.5 Cloud Platform

Nowadays, there are various computing services which are provided by the different cloud platform. In paper Velkoski et al. (2013), the authors give a detailed study of the Microsoft Azure platform and its services. Azure provides services like web-hosting, blob storage for sharing storage virtually, virtual instances and many more. Azure comes with various API framework Chappell et al. (2010) that allows user to access and share their systems. Authors give a brief idea about the fabric controller that allows connecting with third party server. The paper study is crucial because this proposal has used the Azure platform to host website and to manage and configure the MySQL database.

2.6 Altruistic Sharing

Altruistic Sharing is an approach in which the user can share computing resources with other users who need little or no benefits. This comes under self-volunteering and crowdsourcing model. In recent years, authors Inagaki and Shinkuma (2018) proposed architecture for altruistic sharing using the social relationship for sharing resources and managing it. The study of this paper is pivotal to this proposal as it gives a brief suggestion on sharing and managing resources. Authors have also developed a prototype to measure system authentication overhead. This system works on social networks to verify that proposed system has limited access to resources which is not close to the resource owner and authentication process does not affect the guest to share resources.

2.7 Middleware technology for Sharing and Managing Resources

In Anderson (2019) paper, authors gives various aspect to use BOINC (Berkeley Open Infrastructure for Network Computing) as it is an open-source middleware system which supports Volunteering Computing. Using BOINC can reduce the workload and overhead cost of the infrastructure. The paper suggests BOINC can handle all the virtually HTC application as well. The study of this paper is very critical for this research project as it helps to implement BOINC with web-server for resource virtualization. Another approach in paper Ferreira et al. (2011) gives the idea of sharing personal resources via virtualized container image. This image can be used to access owner resources.

2.8 Approach for sharing resources on Social Cloud

In Gan et al. (2017) paper, authors have proposed an algorithm for resource sharing using social crowdsourcing model. The main concern for the authors was the shortage of resources for game-based computing. As the games required high configuration with highly power RAM and CPU, they have developed a crowdsourcing mechanism which helps for multi-resources sharing facility. It uses a social networking platform like Facebook to share compute power to social friends where user can utilize the system in a better way. As the resources are being shared on the social platform, the user can enjoy their gaming without actually up-scaling their environment. This paper is beneficial for the research work as it gives advantages like:

- Algorithm for multi-resource sharing.
- To perform complex computing task, microelectronics technique is used.
- Using crowd-sourcing model Murturi et al. (2015) to deploy on social platforms.
- Mechanism like Vickrey-Clarke-Groves(VCG) which helps to allocate process in social cloud.
- For non-interruption of services, DRF based algorithm Martin et al. (2010) has been developed.

This benefits are important to research area as it gives the idea for sharing resources on social cloud platform.

2.9 Conclusion

Many Social Cloud platforms enable the sharing of resources to improve the feasibility and reliability of the system. The significant gap while reviewing these papers is that it is not entirely utilizing personal resources. In this paper, the purpose is to utilize personal resources using social cloud approach via social networks.

Comparative Analysis of Related Work on Sharing Method			
Researchers [Ref.]	**Approach**	**Benefits**	**Limitations**
Guo et al. (2014) Kumar et al. (2010)	Friend Trust Relationship.	privacy-preserving friend recommendation scheme.	Security and Accountability issues.
Casanova et al. (2017) Singh (2018)	Virtualization using RDP approach.	Fast and easy to share.	Trust Factor and Latency.
McMahon and Milenkovic (2011) Wilkinson et al. (2014)	Social Volunteer Computing method.	Sharing resources on social networks.	Generic Algorithm not developed which results in damaging resources.
Anderson (2019) Ferreira et al. (2011)	BOINC Sharing Method.	Increase utilization rate.	Uses only one-way bi-directional communication.

3 Methodology

Cloud Computing social platform provides a low-budget execution of various services like storage, web-hosting, databases. In addition to this, Volunteering Computing can be performed using the middlewares, which helps in the connection between Social cloud and Client. There are two different strategies which are used for volunteering computing resources. The first is using middleware BOINC server. The ideal time of computer when resources are not in use does not allow direct allocation of computing power for volunteering to compute. To save this wastage of computing power, BOINC server is set up in such a way that when a computer is ideal for more than 5 minutes, it is directly allocated the computer for volunteering computing for the scientific purpose at SETI@HOME. Another approach which has been used is sharing RAM, CPU power, Customize OS via Chrome Remote Desktop using virtualization layer of VMWare workstation. The sharing of the system via the virtualization approach not only safeguard the system from security vulnerabilities but also can be deployed quickly and efficiently on the social cloud platform.

This Research project helps to share customized Operating System, RAM and Processor using virtualization approach and is deployed in Social Networking Platform. This approach helps users to share their system securely and efficiently, which solves the research question to utilize 100% hardware in a personal computer using the social platform. Moreover, In paper[1] authors have integrated sharing API within Facebook. There was no dedicated platform where users can share their system to another user. To solve this issue, Social Networking Cloud-based platform has been created where user can share their system directly to the BOINC server or via Chrome Remote method. To know about the concept of volunteering to compute, it is important to understand the workflow of the BOINC server along with the social cloud. This Research project is an expansion of the previous work done by S.Caton, C.Haas, K.Chard, K.Bubendorfer, O.Rana (2014) Caton et al. (2014). The previous methodology was performed using Seattle Peer-to-Peer computing middleware using the Facebook platform as a social sharing option. Following are the different methods that has been used for this project:

3.1 BOINC Architecture

BOINC is an open-source middleware system which is used for Volunteering Computing purpose. To volunteer nodes, resource sharing user need to install BOINC client and can choose projects to give the compute support. This client software supports Windows, Mac and Linux OS. Once the setup is completed, the operation takes place from client to server-side. The Volunteering nodes transfer the task from the client to the BOINC server. The problem with the BOINC server is that it uses a bi-directional way of communication, which means either the volunteer node can communicate with the BOINC server or server can communicate with the volunteer node. Both cannot communicate at the same time. BOINC works on client node request, which means if the nodes are available and do not request for the new task than the process will remain idle until the client request for the new task. This factor leads to the wastage of resources. To eradicate this wastage, a social cloud layer between client and BOINC server has been implemented that can help to handle faster and two-way communication. This request can be handle by MySQL server which can control multiple requests at a time.

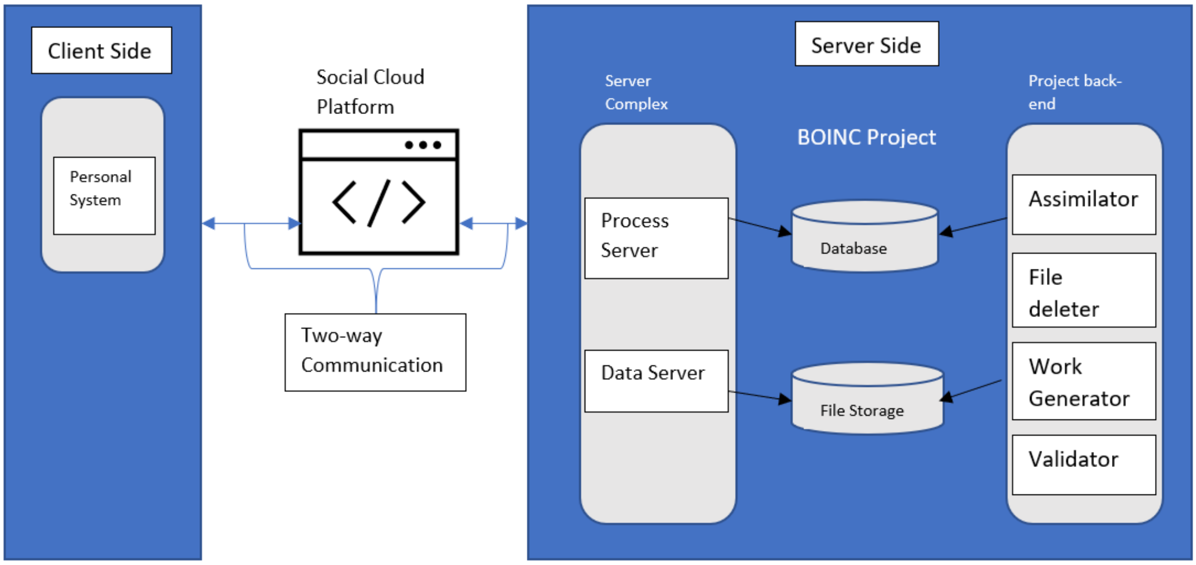


Figure 2: An Approach for BOINC

3.2 Remote Connection

The approach of Personal Computer over Internet Protocol (PCoIP) Remote Desktop Protocol is used to connect another computer over a network. With this approach, RDP can be use for volunteering purpose also. Today there are many cloud provider like Microsoft Azure, AWS, GCP, which gives access to the virtual image via RDP by using a pay-as-you-go model. This proposal is one-way, which required the user to pay for using services. Research proposal is based on Volunteering Computing, where user can share their virtual system on social networking platform using the remote desktop protocol. The problem with standard remote sharing, for example, Team Viewer application where one user gives remote access of the system to another user which lead to high-security risk and also only one user can use the system. With Virtual Desktop Infrastructure (VDI) Yang et al. (2018), virtualization layer is used to share the system. User can create a virtual image with specifying specification in VMWare workstation. This VM can be share in social networking cloud platform to other users. This helps better utilization on the personal computer as well as a secure way to transmit data over cloud storage.

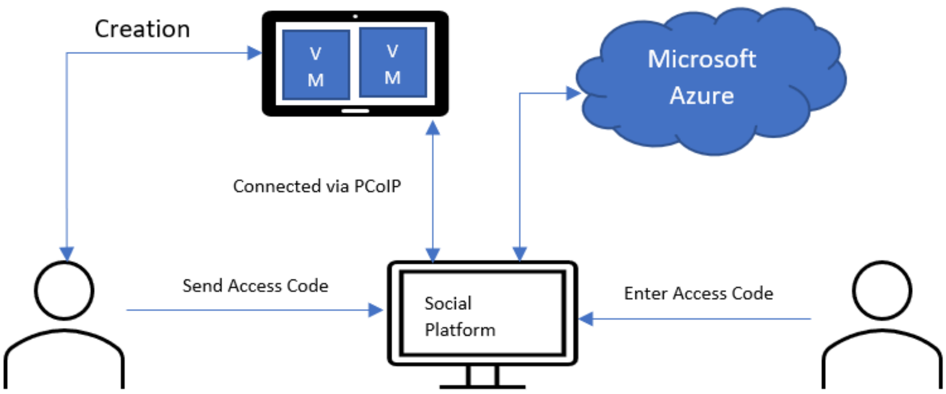


Figure 3: PCoIP Sharing Method using Cloud

3.3 Proposed Architecture

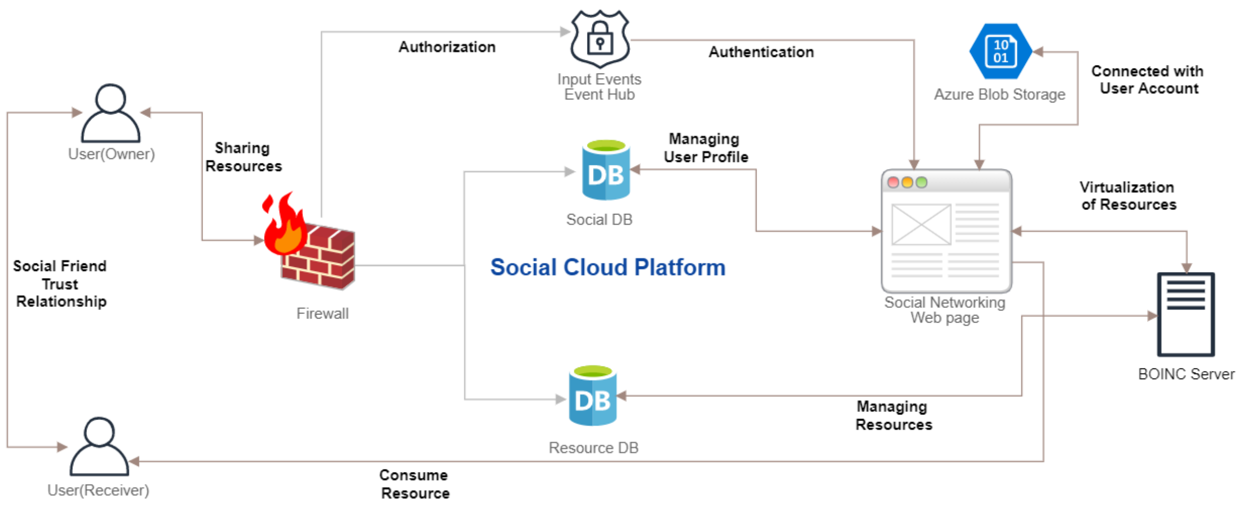


Figure 4: System Architecture

Figure 4 shows the working architecture of this project. It demonstrate various aspects that are critical and required to accomplished research project. The process is as follows:

- First, the User (Owner) acquires a secure connection for social networking cloud website by going through the firewall layer with username and password.
- The Social Networking page is connected with Social Database, which is used to manage user profile and system information.
- Resources Database manages and keep a record of the running time of the BOINC server and also keep track of the services allocated.
- For Cloud Storage Option, it is directly connected with Azure Blob Storage. The User needs to run a script (provided in Azure) to get connected with network storage.
- BOINC server is directly implemented with web server where User can share their system securely.
- User (Receiver) can demand the compute services from User (Owner) by posting or using personal message section present in social website.

4 Design Specification

Figure 5 shows the research workflow diagram. The first level is to initialize the process in order to share/consume the resources. The user first need to verify their identity with correct credentials. After that, user can request middleware BOINC server to share the services for a specific project, BOINC server then process the request from the user and automatically virtualize the resources in the background process of the user's system. The CPU processor of the user system do not get completely blocked while sharing resources on BOINC. User can use some amount of CPU power for personal use while sharing to BOINC server. All the services are managed and record by the social networking platform layer where the consumer can request CPU and compute power from resources owner. After successfully using the services user can suspend or end the service.

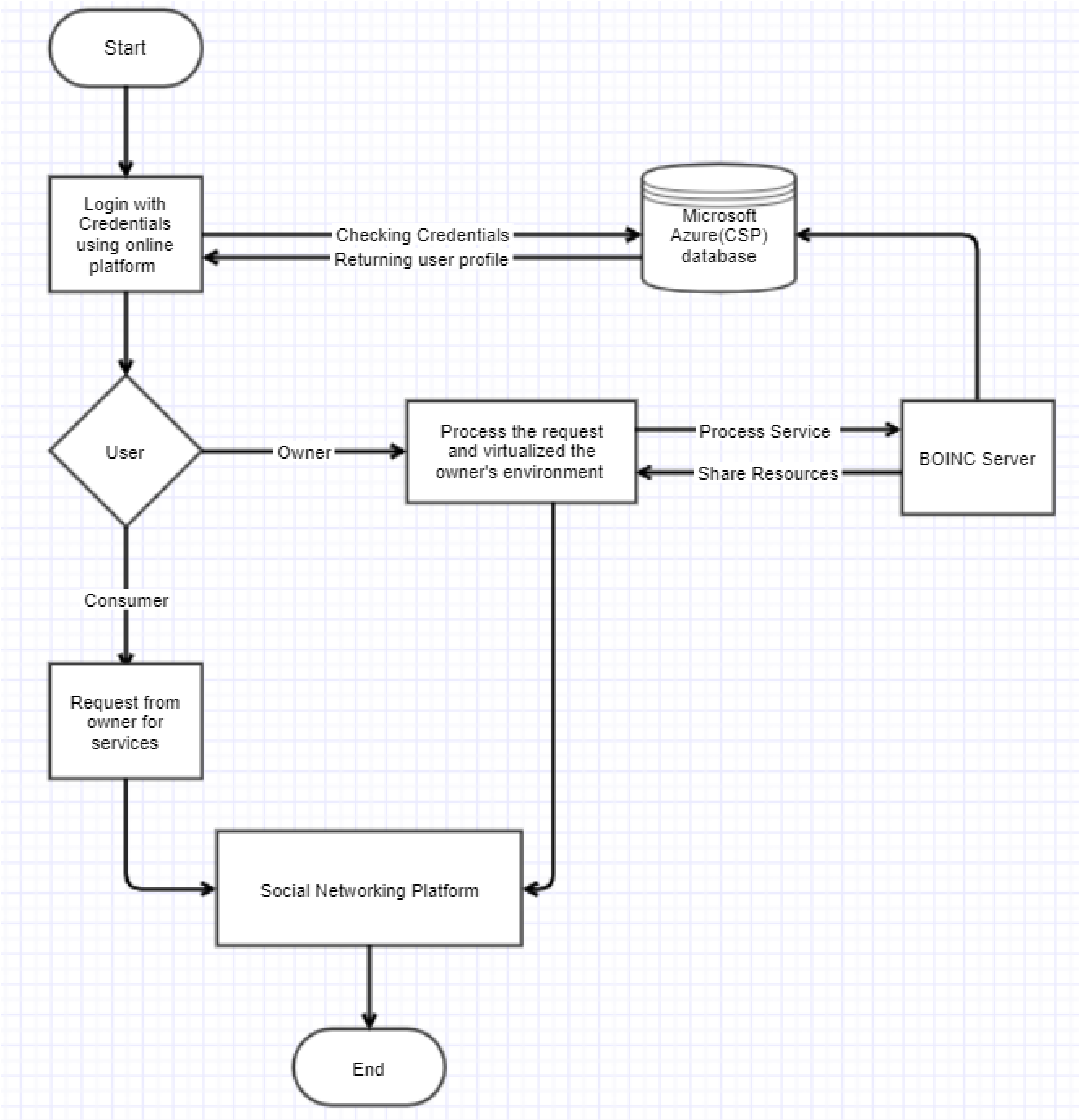


Figure 5: Work Flow Diagram

5 Implementation

To investigate and increase the CPU utilization rate of the personal computer on the social cloud, the following implementation method was conducted: The Social Networking Platform was developed in OSSN (Open Source Social Networking) using PHP language. This platform was configured on Microsoft Azure Virtual Machine with an integrated database. To manage the web-server, the Apache Tomcat server has been configured on localhost. For managing the user database and system information, MySQL server has been configured. Both Apache and MySQL server are managed using the XAMPP control panel. While implementing the BOINC server, some specific criteria were considered, if the volunteer nodes are available to share or not. Also, the most appropriate node has been allocated for sharing directly in the BOINC server. Configuration has been done on 2 different nodes: the first is SETI@HOME and the second is nanoHUB_at_home. These nodes are usually available to share personal compute resources. For remote connection configuration, PCoIP method is used. The remote session is configured in Chrome remote support with OSSN direct implementation. If the user logout the social account, all the nodes and remote connection will be lost for higher security. The proposed implementation was performed on Cloud virtual machine with Windows 10 OS.

5.1 Implementation OSSN Social Platform

In this section, a social networking website cloud-based platform is proposed where users can share resources with their friend using Friend-trust relationship and Generic algorithm. To implement the platform, it is an essential step to setup MySQL, Apache Tomcat server, in order to host the platform on the web.

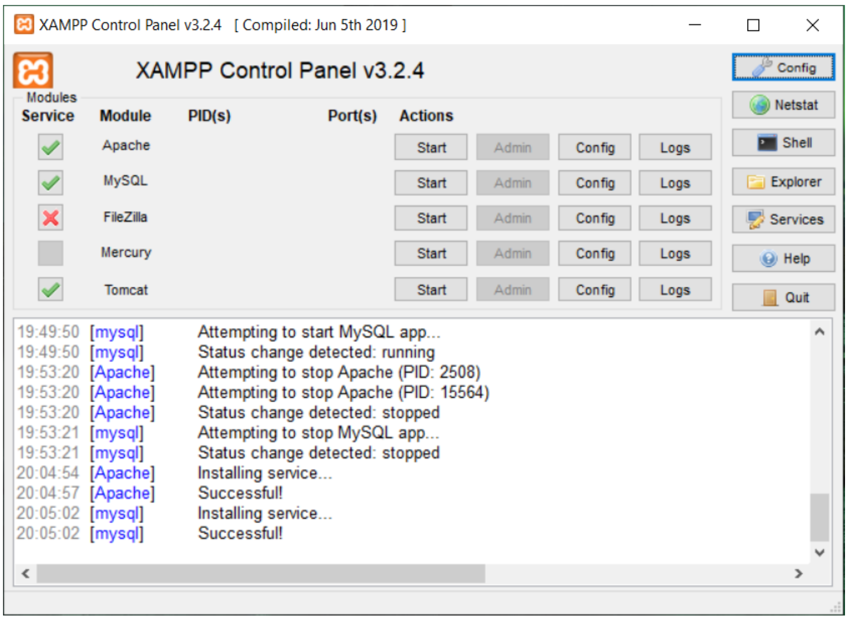


Figure 6: Server Manager Control Panel

Server Specification can be found in the configuration manual. Once the Server setup is done, validation is required for OSSN nodes to start the services successfully.

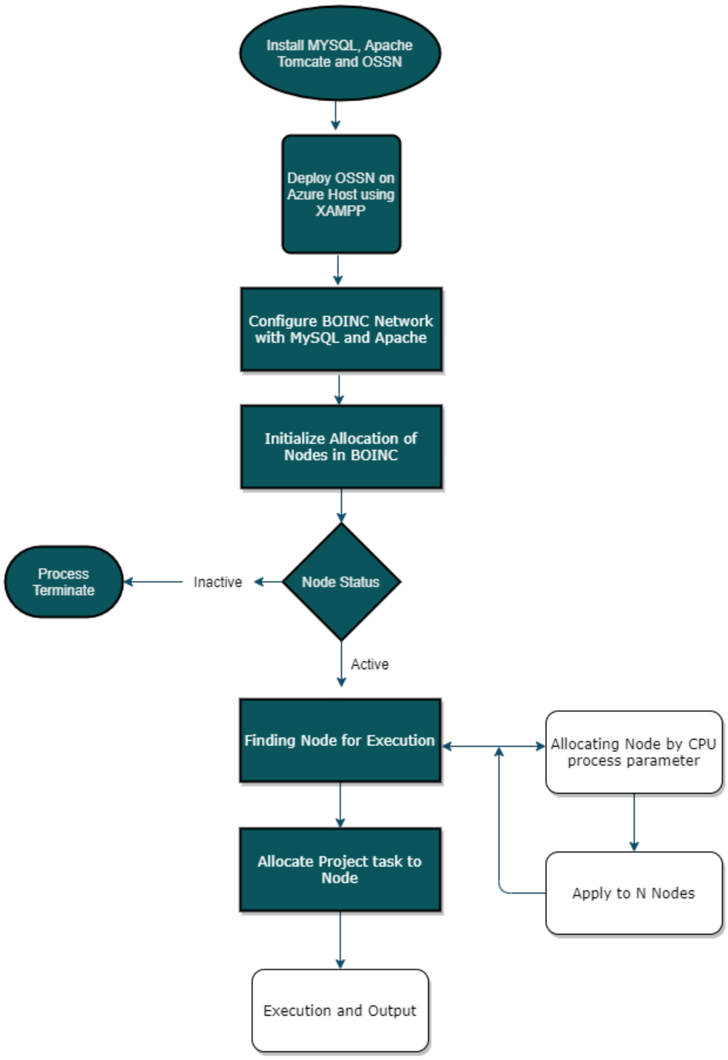


Figure 7: Implementation Workflow

Figure [7] explains the process of Node allocation for better utilization in detail. Once the MYSQL, Apache Tomcat server, and OSSN are set up, it goes to the configuration stage in BOINC. Social Cloud-first evaluates the process by initializing and allocating nodes in BOINC. The next process is to check the status of the node is an active or inactive state. If the node is inactive, then the process get terminated, which results in nodes that cannot be allocated to a selected project in BOINC. If it is an active state, it goes to the next step to find which node is most suitable for execution. The process then compares different nodes with the BOINC project parameters and Processor of the user's system. If the user's processor is greater than or equal to the project parameters, it allocates nodes to the BOINC projects.

To successfully run the service from social cloud to BOINC server it is essential to integrate the BOINC server on social platform.

- **MOD_REWRITE:** Use to rewrite and modify URL. Also different rules can be setup for URL.
- **PHP CURL:** It is use for Client URL. It enables variety of web resources from PHP scripts. Also it helps to send HTTP request which is essential for remote sharing of

system. It supports 3rd party API system that has been used to implement BOINC server directly to website.

- **PHP openssl:** This is use to secure the web communication over the network. It uses HTTPS secure protocol.
- **PHP json:** It is a Java Script Object Notation which help to store and exchange information on the web-server. It is used to store user profile data.

Algorithm 1 Resource Sharing Matching Algorithm(RSMA) Genetic

Phase 1. Parameter Detection
The Node allocation on BOINC via social platform is based on user processor capacity.
Phase 2. Initialization
Each resource owner construct their own preference list of BOINC project;
Construct the set of users that are not matched with Node Type, *UNMATCH*
Phase 3. Matching algorithm
do
all BOINC Project (x) *MATCH* which have Processor and RAM (y) and have never rejected it before;
for all resource provider user x, x=1 **do**
if $x \leq y$ **then**
resource provider x Allocates all the resources to BOINC server;
Remove those *users* from the *UNMATCH* list;
Else
resource provider x not eligible to share node on chosen BOINC Project,
$x \geq y$, reject the request;
Kept the users in the *UNMATCH* list;
end if
end for
End of algorithm;

Algorithm 1 explains the execution process in BOINC by directly allocation of Processor and RAM. First, the algorithm will check the Processor and RAM capacity which is required in BOINC project. If the sharing user matches the parameters with the chosen project, node will be allocated automatically. It also removes user from *UNMATCH* list. If user is not eligible to share on the specific BOINC project, it goes to the *UNMATCH* list and finds new BOINC project to allocate nodes.

5.2 Model Developed

Figure 9 shows the process flow of a node. The project has two types of sharing approach. The first approach is volunteering computing, where users share their compute power for scientific purposes. From figure 8, User 1 is sharing their system to BOINC server using social Networking platform. This platform is pre-configured with BOINC server projects and node allocation. Another approach is the sharing system using Personal Computer over IP. Here, User 2 is sharing the system with User 3, which are both social friends. This session is run in web-platform itself, which is highly secure, and the latency rate is very low. The cloud-oriented social platform also supports Azure storage service, where the user can save their data in the cloud.

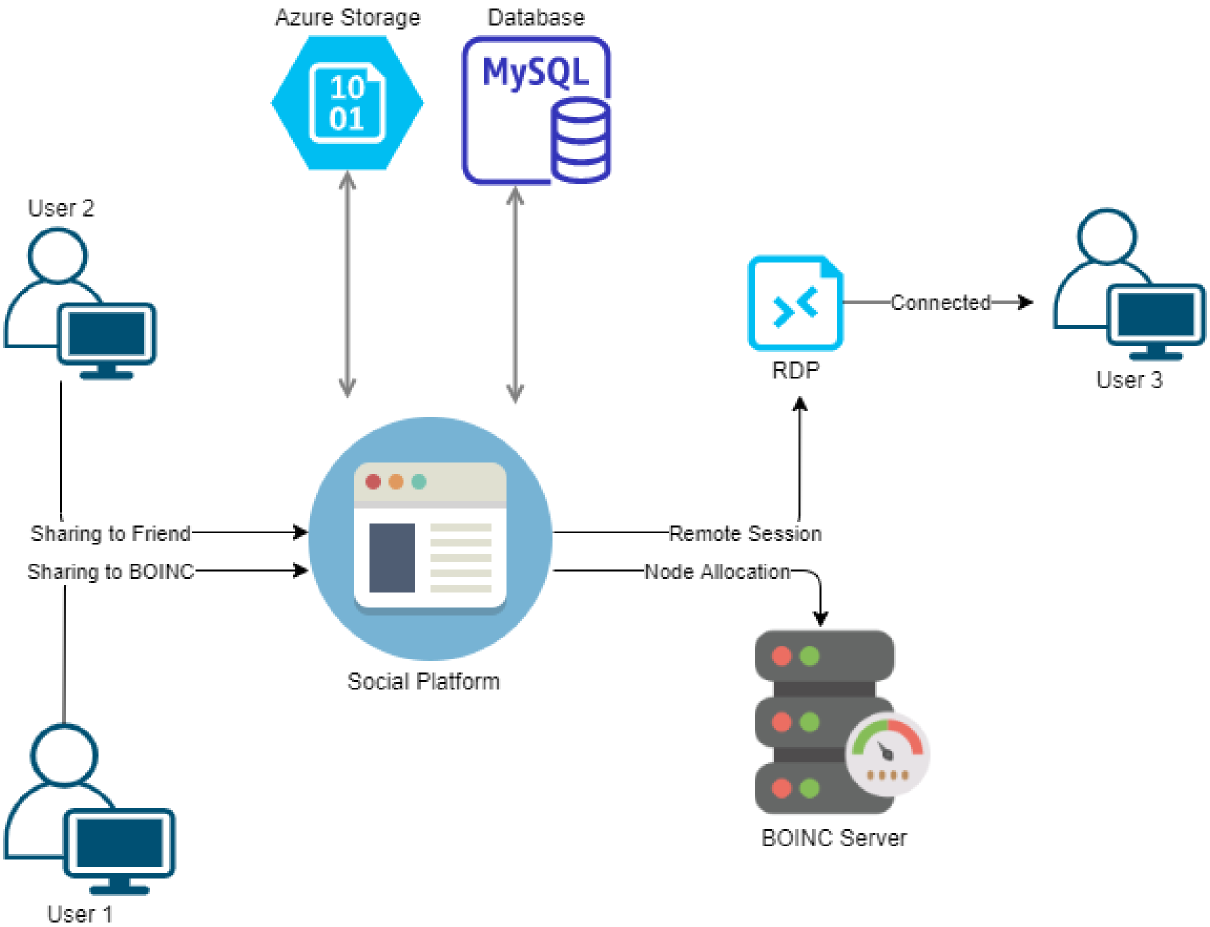


Figure 8: Working Model

6 Evaluation

The primary goal of this research project is to analyze and utilize personal computing power by virtually allocating on social networks. To evaluate a social cloud method, a platform has been proposed with which it helps to allocate different resources, various size, etc. To critically evaluate the functionality and working, tools like Ranorex and different experiments are used. Following are the experiment conducted to analyze results:

6.1 Experiment 1

This Experiment shows the CPU utilization rate in a standard personal system and also shows the utilization rate when the nodes are allocated in the BOINC server.

Explanation: From figure 9, it can be observed that the utilization rate keeps on fluctuating from 5% to 50% depending on the usage. After running 255 processes, the utilization rate was just 11%. Now in figure 10, it can be observed that the CPU utilization rate increases rapidly with the approach to allocate nodes virtually on the BOINC server. This method helps to utilizes a personal system between 80% to 100%.

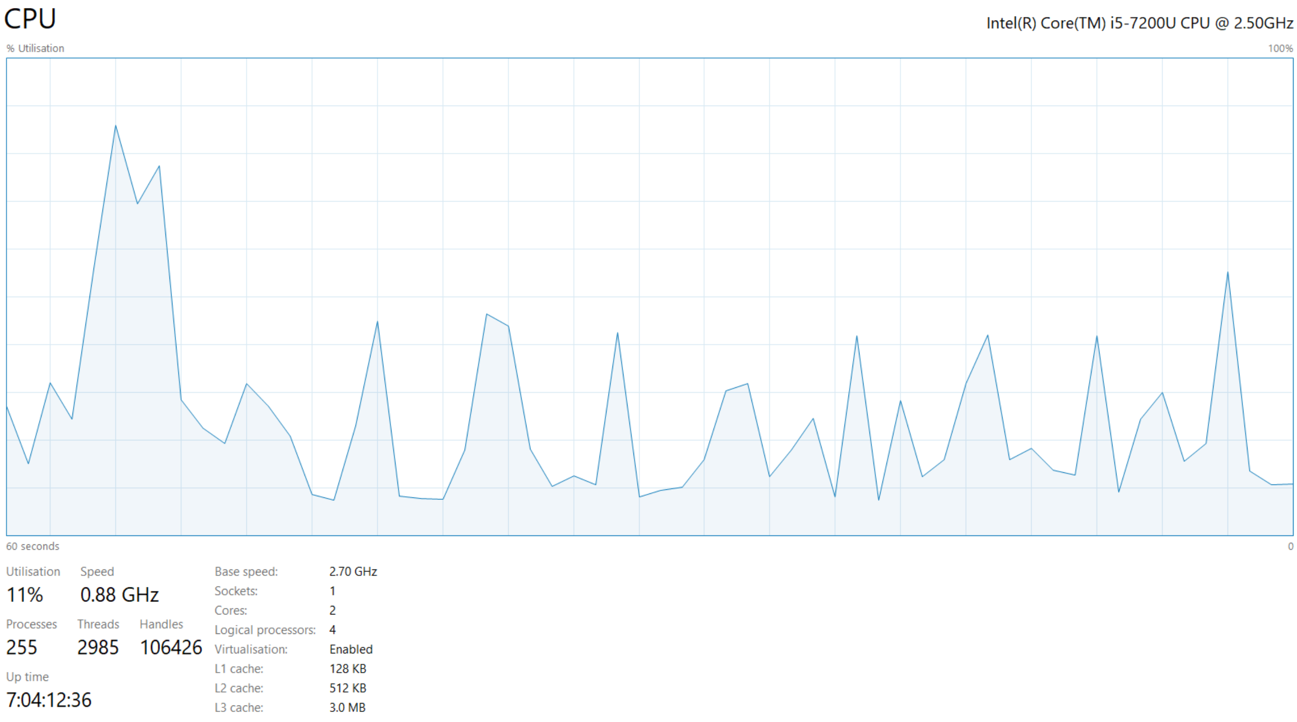


Figure 9: Utilization rate without Node Allocation

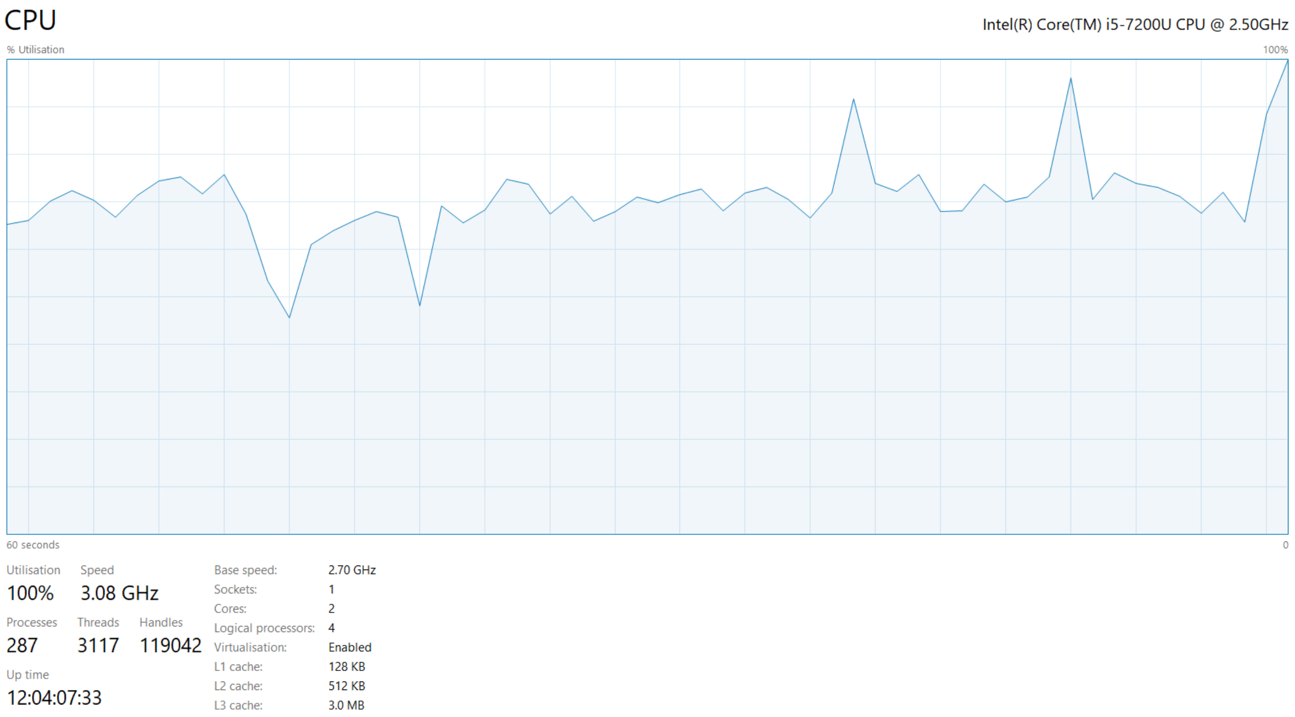
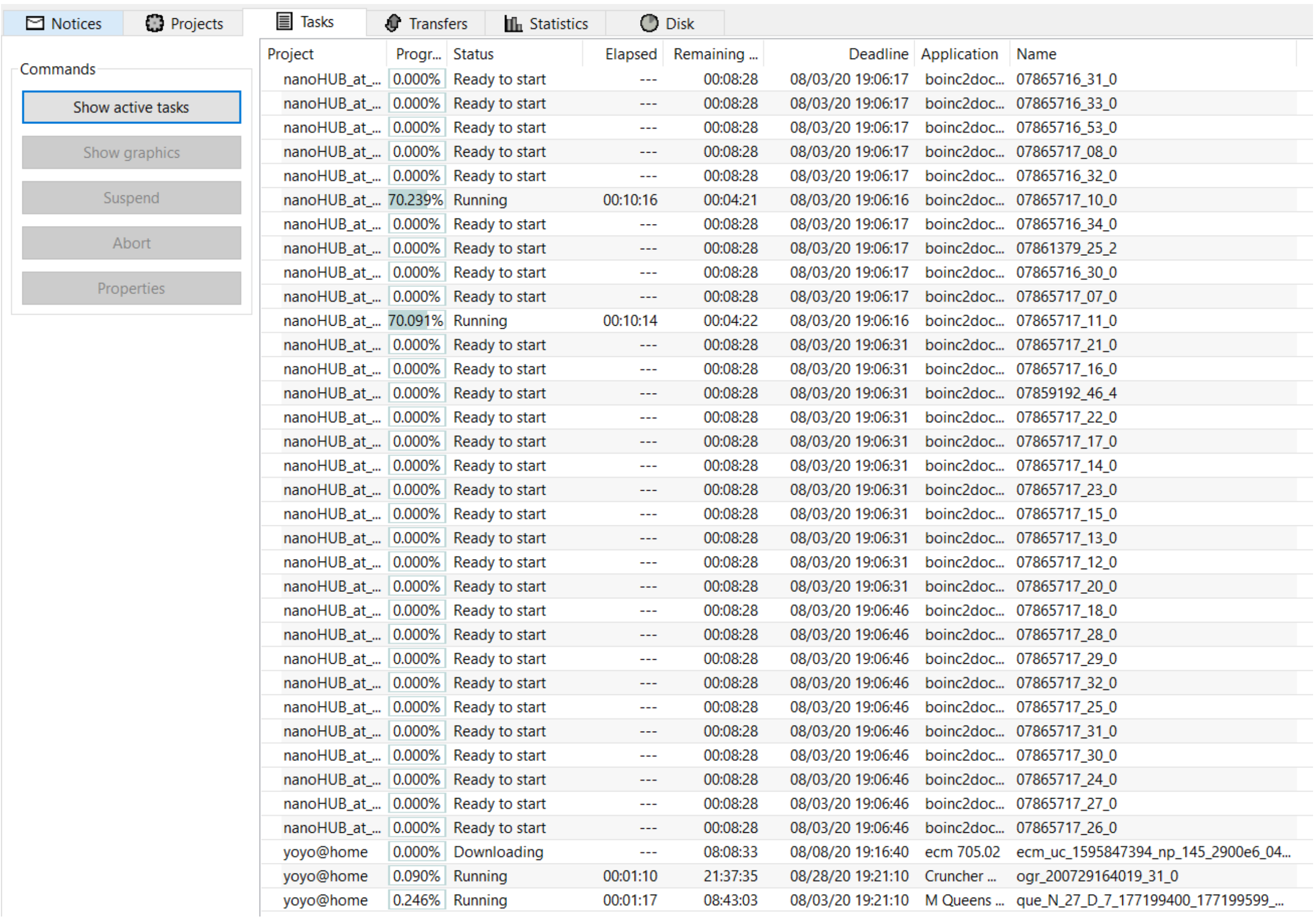


Figure 10: Utilization rate with Node Allocation

6.2 Experiment 2

In this experiment, different Node allocation is checked with the status of each BOINC server project. This concludes that the nodes can be allocated frequently and simultaneously without wasting the CPU and disk power.



Project	Progr...	Status	Elapsed	Remaining ...	Deadline	Application	Name
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865716_31_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865716_33_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865716_53_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865717_08_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865716_32_0
nanoHUB_at_...	70.239%	Running	00:10:16	00:04:21	08/03/20 19:06:16	boinc2doc...	07865717_10_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865716_34_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07861379_25_2
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865716_30_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:17	boinc2doc...	07865717_07_0
nanoHUB_at_...	70.091%	Running	00:10:14	00:04:22	08/03/20 19:06:16	boinc2doc...	07865717_11_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_21_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_16_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07859192_46_4
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_22_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_17_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_14_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_23_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_15_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_13_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_12_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:31	boinc2doc...	07865717_20_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_18_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_28_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_29_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_32_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_25_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_31_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_30_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_24_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_27_0
nanoHUB_at_...	0.000%	Ready to start	---	00:08:28	08/03/20 19:06:46	boinc2doc...	07865717_26_0
yoyo@home	0.000%	Downloading	---	08:08:33	08/08/20 19:16:40	ecm 705.02	ecm_uc_1595847394_np_145_2900e6_04...
yoyo@home	0.090%	Running	00:01:10	21:37:35	08/28/20 19:21:10	Cruncher ...	ogr_200729164019_31_0
yoyo@home	0.246%	Running	00:01:17	08:43:03	08/03/20 19:21:10	M Queens ...	que_N_27_D_7_177199400_177199599_...

Figure 11: Frequent Node Allocation

6.3 Experiment 3

This Experiment shows the utility rate of each node when it is allocated for sharing in the BOINC server. This concludes that disk space can be temporarily shared for volunteer purpose.

Explanation: In graph 12, the grey shaded circle shows the Total disk usage by BOINC nodes. The black layer indicates the space used by nodes which is 1.78GB in this Experiment. The dark grey shaded portion shows free space available to nodes which is 20.61GB. The red circle shows the disk usage of the different BOINC projects. For this Experiment, three different node allocation are considered. This Experiment shows the dynamic way to allocates the disk space for a better utilization rate of the resources.

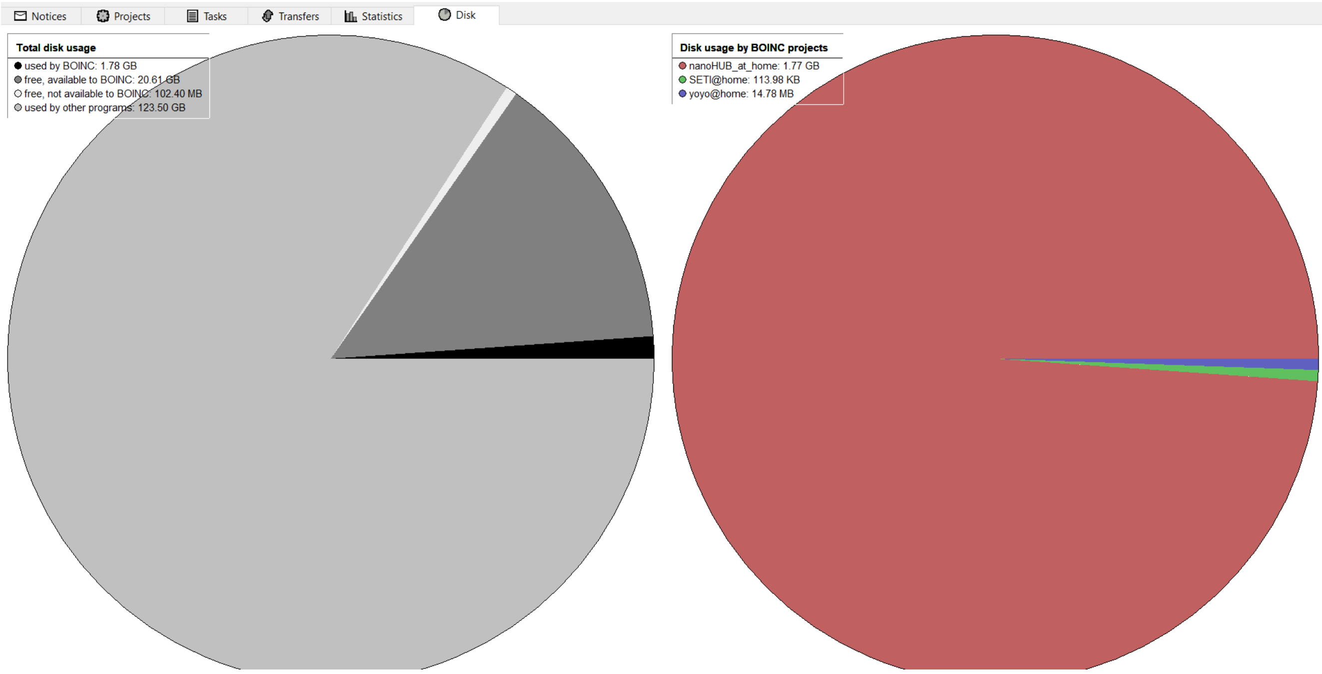


Figure 12: Disk Usage

6.4 Discussion

From the above experiment, it can be conclude that the BOINC server has a tremendous utilization rate of personal resources. The BOINC server, when integrating with social cloud, gives the best result which help users to allocate nodes to the server dynamically. However, When a large number of the nodes are allocated to the physical system it can be nonfunctional. Due to this, user may be unable to use the system until the nodes are entirely allocated and the process is finished . It is more feasible to suspend the node directly for the functionality of the system.

A Social sharing of resources using PCoIP in cloud computing relies on various factors. It is an excellent option to use PCoIP and BOINC with cloud deployment. However, while sharing resources using PCoIP(RDP method), if the host system stops, the functionality of the guest system also stops. This can lead to loss of data. To overcome this, direct connectivity of cloud storage with the social platform is introduced in this report. This helps to retrieve the lost data for the guest systems. Therefore Social network platform is a more feasible option for allocation and security purpose.

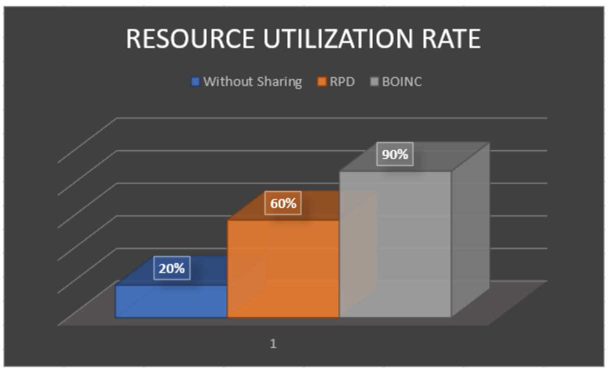


Figure 13: Personal System Resource Utilization Rate

7 Conclusion and Future Work

The main objective of this research was to improve the utilization rate of personal systems using the social cloud. The social platform using OSSN networks was used to create a platform where users can share their system directly for volunteer purpose or for sharing it to friends via Friend Trust Relationship factor. The BOINC is used as a middleware in the process of sharing nodes. The main drawback of the BOINC server was that it cannot be allocated task dynamically and efficiently as it functions unidirectional. To improve this, Social Platform was created with an integrated BOINC server which helps to allocate node dynamically, and the request is communicated in a two-way manner. Another approach that is implemented is sharing resources to a friend via PCoIP, which works as a remote desktop protocol. The problem with RDP is that it is less secure to share on an open network. Moreover, it can harm the host system directly. To improve this, integration of social web-platform with PCoIP is conducted which run RDP session within the website itself. As the website is secure with TLS, HTTPS protocol, it becomes more secure. The host can share the VM directly created in VMWare. The main goal of sharing the personal system to friends is to increase the utilization rate and reduce wastage of computing resources. Through these 2 approaches, the utilization rate of the personal system is enhanced.

In the future to enhance the utilization rate, direct node access from host to guest can increase the accuracy to determine the correct node for processing. The social cloud can play a significant role for resources sharing techniques. Also, There is numerous algorithm available that could be use to reduce resource wastage for better efficiency to the existing solution.

Acknowledgement

I want to express my gratitude and thank my supervisor Prof.Vikas Sahni who always motivated me to do better and helped me throughout the coursework with his critical inputs. His valuable suggestion has helped me to complete the project and implementation in the right direction. I would also like to thank all the Professors and staff member of the National College of Ireland for providing excellent guidance and support throughout my coursework.

References

Anderson, D. P. (2019). Boinc: A platform for volunteer computing, *Journal of Grid Computing* pp. 1–24. `https://www.researchgate.net/publication/331543735_BOINC_A_Platform_for_Volunteer_Computing`.

Casanova, L., Kristianto, E. et al. (2017). Comparing rdp and pcoip protocols for desktop virtualization invmware enviroment, *2017 5th International Conference on Cyber and IT Service Management (CITSM)*, IEEE, pp. 1–4. `https://doi.org/10.1109/CITSM.2017.8089272`.

Caton, S., Haas, C., Chard, K., Bubendorfer, K. and Rana, O. F. (2014). A social compute cloud: Allocating and sharing infrastructure resources via social networks, *IEEE*

Transactions on Services Computing **7**(3): 359–372. https://doi.org/10.1109/TSC.2014.2303091.

Chappell, D. et al. (2010). Introducing the azure services platform, *White paper, Oct* **1364**(11). http://www.davidchappell.com/Azure_Services_Platform_v1.1--Chappell.pdf.

Ferreira, D., Araujo, F. and Domingues, P. (2011). libboincexec: A generic virtualization approach for the boinc middleware, *2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum*, IEEE, pp. 1903–1908. https://doi.org/10.1109/IPDPS.2011.349.

Gan, X., Li, Y., Wang, W., Fu, L. and Wang, X. (2017). Social crowdsourcing to friends: An incentive mechanism for multi-resource sharing, *IEEE Journal on Selected Areas in Communications* **35**(3): 795–808. https://doi.org/10.1109/JSAC.2017.2672379.

Guo, L., Zhang, C. and Fang, Y. (2014). A trust-based privacy-preserving friend recommendation scheme for online social networks, *ieee transactions on dependable and secure computing* **12**(4): 413–427. https://doi.org/10.1109/TDSC.2014.2355824.

Huber, N., von Quast, M., Hauck, M. and Kounev, S. (2011). Evaluating and modeling virtualization performance overhead for cloud environments., *CLOSER*, pp. 563–573. https://www.researchgate.net/publication/220865632_Evaluating_and_Modeling_Virtualization_Performance_Overhead_for_Cloud_Environments.

Inagaki, Y. and Shinkuma, R. (2018). Shared-resource management using online social-relationship metric for altruistic device sharing, *IEEE Access* **6**: 23191–23201. https://doi.org/10.1109/ACCESS.2018.2823300.

Javadi, B., Kondo, D., Vincent, J.-M. and Anderson, D. P. (2011). Discovering statistical models of availability in large distributed systems: An empirical study of seti@ home, *IEEE Transactions on Parallel and Distributed Systems* **22**(11): 1896–1903. https://doi.org/10.1109/TPDS.2011.50.

Kouril, J. and Lambertova, P. (2010). Performance analysis and comparison of virtualization protocols, rdp and pcoip, *Proceedings of the 14th WSEAS international conference on Computers*, pp. 782–787. http://www.wseas.us/e-library/conferences/2010/Corfu/COMPUTERS/COMPUTERS2-62.pdf.

Kumar, R., Novak, J. and Tomkins, A. (2010). Structure and evolution of online social networks, *Link mining: models, algorithms, and applications*, Springer, pp. 337–357. https://doi.org/10.1007/978-1-4419-6515-8_13.

Martin, P., Li, H.-Y., Zheng, M., Romanufa, K. and Powley, W. (2010). Dynamic reconfiguration algorithm: Dynamically tuning multiple buffer pools, *International Conference on Database and Expert Systems Applications*, Springer, pp. 92–101. https://doi.org/10.1007/3-540-44469-6_9.

McGilvary, G. A., Barker, A., Lloyd, A. and Atkinson, M. (2013). V-boinc: The virtualization of boinc, *2013 13th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing*, IEEE, pp. 285–293. https://doi.org/10.1109/CCGrid.2013.14.

McMahon, A. and Milenkovic, V. (2011). Social volunteer computing, *Journal on Systemics, Cybernetics and Informatics (JSCI)* **9**(4): 34–38. https://www.researchgate.net/publication/267707234_Social_Volunteer_Computing.

Murturi, A., Kantarci, B. and Oktug, S. F. (2015). A reference model for crowdsourcing as a service, *2015 IEEE 4th International Conference on Cloud Networking (CloudNet)*, pp. 64–66. https://doi.org/10.1109/CloudNet.2015.7335281.

Singh, M. (2018). Virtualization in cloud computing- a study, *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pp. 64–67. https://doi.org/10.1109/ICACCCN.2018.8748398.

Velkoski, G., Simjanoska, M., Ristov, S. and Gusev, M. (2013). Cpu utilization in a multitenant cloud, *Eurocon 2013*, IEEE, pp. 242–249. https://doi.org/10.1109/EUROCON.2013.6624993.

Wilkinson, S., Boshevski, T., Brandoff, J. and Buterin, V. (2014). Storj a peer-to-peer cloud storage network. https://storj.io/storj2014.pdf.

Yang, C., Liu, J., Lee, J., Chang, C., Lai, C. and Kuo, C. (2018). The implementation of a virtual desktop infrastructure with gpu accelerated on openstack, *2018 15th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN)*, pp. 366–370. https://doi.org/10.1109/I-SPAN.2018.00069.