Datacenter Management as a Service (DMaaS)

Project Work Book
Version 2.0

By
Debasis Dash
Juthika Dash
Shivani Patil
Tanushree Chaubal

Date
10th Nov 2015

Advisor: Dr. Youngee Park

Table of Contents

Table of figures

List of tables

Version History

Version	Date	Contributor	Changes
1.0	10/25/2015	DMaaS Team	Chapter 1, 2, 3, 4, 5,8, 9
2.0	11/10/2015	DMaaS Team	Chapter 1, 6,7,8,9, 10

Related Documents

Sl. No	Author	Document Name
1	Debasis Dash	Function Specification document for Virtual Abstraction Interface
2	Juthika Dash	Function Specification document for Access Control List and Decision Tree
3	Shivani Patil	Function Specification document for DDoS detection and prevention
4	Tanushree Chaubal	Function Specification document for Recommendation engine

Chapter 1. Literature Search, State of the Art

Literature Search

Currently security and resource management in virtualized data centers is one of the major challenge. Virtualized Data Centers can be thought of a pool of cloud resources used for enterprise needs that mostly are related to storage and processing. The papers [6] "The study on Data Security on Cloud Computing based on Virtualization" and [7] "Security Implications of Virtualization: A Literature Study" gives an introduction to why using virtualization is beneficial and what are the security issues that are currently faced while using virtualized data centers. Some issues faced while managing virtual infrastructure include maintenance, configuration and updates for software, maintaining consistency and integrity, so on [6]. Maintaining the performance of data center is also important factor since many computations are dependent on this. Also, recovery from failure should be handled carefully as one data center might be responsible for processing many different tasks simultaneously. Some security concerns for a virtualized environment are Confidentiality, Integrity, Authenticity, Availability and Non-Repudiation [7]. A solution, which handles all these factors appropriately, can be considered as an excellent solution. The data center traffic is great source of information for security monitoring. The network traffic needs to be captured and analyzed using modern tools like big data tools [8].

Large-scale distributed system consolidation has become a pivotal component and enabling technology for infrastructure-as-a-service cloud computing environment. Virtualized datacenter is a popular approach to achieve such an aggregation of individual systems in the network. This consolidation of systems will inevitably cause threats to the security of the network as a whole. This gives rise to on-demand security monitoring of guest systems, to track and eliminate threats, which may jeopardize the operation of the Virtualized Datacenter. An on-demand, mandatory security monitoring system in Virtualized Datacenter environment is required, which won't depend on pre-installed guest components. The ultimate aim is to utilize, deploy, work and monitor the security in a large-scale Virtualized Datacenter cloud environment, like IaaS cloud [1].

On-demand, rapid scalability and resource pooling are the main characteristics of any cloud-computing environment. Virtualization of Datacenter in any cloud environment, host's load balance, power saving, failure recovery and dynamic resource allocation is dependent on live migration of virtual machines in the datacenter. So studying and modeling live migrations [2] is important to predict its impact on the datacenter performance and to take the migration decision at the appropriate times. Hence the study of live migration impact on datacenter resources utilization given the virtual machine and network characteristics, becomes a necessity. Resource management techniques of live migration are important to have less cost, higher resource utilization, higher availability and green cloud-computing environment. Live migration is an integral part of virtualization of a Datacenter, but overheads in live migration may adversely affect datacenter CPU, network and power consumption. Hence it's necessary to implement live migration in a virtualized datacenter and ensure elimination of migration overheads simultaneously [2].

A powerful monitoring and reporting tool for VMWare VSphere and Microsoft Hyper-V called Veeam ONE is a pre-existing tool which enables "Availability for Modern Datacenter TM" which provides backup and performance issue notifications before they affect the users of the VMs. Veeam ONE provides real-time, agent free and unattended 24x7 monitoring of backup and virtual infrastructures, notifying the manager of backup and performance issues before applications and users are negatively impacted. It performs fast troubleshooting by providing information to point out the root cause and easily resolve issues, providing advanced up-to-date alerting capabilities to guarantee that business remains working at all times [13].

A free virtualization monitoring software, called VM Manager Plus monitors unlimited VMs, hosts, and data stores across VMware, Hyper-V and XenServer platforms. It monitors critical performance metrics such as CPU, memory, and disk utilization, network usage, memory swap, data store read/write latency, memory ready, and much

more. VM Manager Plus also monitors live migration of VMs and updates the VM-host relationship map automatically [14].

A latest trend is the integration of recommendation systems in the applications. Netflix recommendation system [15] emphasizes the importance of using both data and algorithms to create the best possible experience for Netflix members. The article provides information on how to create a software architecture that can deliver fast recommendations. The main goal of their system was coming up with a software architecture that handles large volumes of existing data, is responsive to user interactions, and makes it easy to experiment with new recommendation approaches. The data can be stored for later offline processing or online. Model training is another form of computation that uses existing data to generate a model that will later be used during the actual computation of results. Another part of the architecture describes how the different kinds of events and data need to be handled by the Event and Data Distribution system and how to combine intermediate Recommendation Results in a way that makes sense for the user.

Denial of Service attacks in clouds is a major issue faced nowadays. These attacks are referred to as Distributed Denial of Service (DDoS) attacks. When a system is under DDoS attack, the usage of the system shoots up, degrading its performance. Researchers have been trying hard to find an optimal solution. One such solution proposed in the paper [17] is called FCMDPF (Flexible, Collaborative, Multi-layer, DDoS Prevention Framework). This solution aims at preventing DDoS attacks by using three layers. The three layers are: An outer blocking scheme, which aims at blocking an IP if it is listed in the blacklist. The next layer is Service Traceback Oriented Architecture (STOA), which is responsible to check if the input request is created by human or is automated. The last layer is Flexible Advanced Entropy Based scheme to help eliminate flash-crowd and DDoS attacks. [17]

Another approach to mitigate the DDoS attacks is by using dynamic allocation strategy [16]. When a system is under DDoS attack, the dynamic allocation strategy for resources can be used in order to mitigate the attacks and avoid performance degradation. The solution suggests using the idle resources in the cloud resource pool for cloning the

intrusion prevention servers. This will help in filtering the attack-packets and will help in maintaining the quality of service for users [16].

An anomaly detector using "Least Mean Square- Forward Linear Predictor" has been proposed which helps in detecting anomalies by predicting the Entropy-Ratio of the IP flows [18]. The entropy-ratio is then compared with the values of real measures for detecting anomalies. For calculating the entropy-ratio, the traffic is divided in groups according to the source and destination IP, source and destination port numbers and type of protocol. The groups are then used to create matrices called as "Randomly Aggregated Flow Matrix" and "Randomly Aggregated Traffic Matrix". The entropy-ratio doesn't change and thus helps in predicting future values using past values [18].

A security model called Trusted Private Virtual Datacenter was proposed which offers centralized security management for cloud computing resources used for storage and processing. This model provides security for resources distributed in different locations within an IaaS cloud. The IaaS cloud provides capabilities of on-demand allocation of resources. The objective of this model was to provide solution to the security issues like privacy, data integrity and better management of the virtualized infrastructure. The architecture composed of IaaS data center, which provided an infrastructure to host multiple customers, trusted virtual clusters and trusted virtual machines, which are used for processing the customer requests [4].

The efficient Security metrics reporting system should be able to access raw security data sets (such as security logs) and have efficient drill-down functionalities – a generation of more specific reports on user defined queries, and the identification of individual entities in raw security data (such as log messages or Netflow records) [9]. Major security metrics includes Network IDS Alarm logs, NetFlow Data, server logs and other logs which can be used to monitor the security of the data center. The need of capturing the live data and process is needed for security monitoring. These data are so large that it needs big data tools like Hadoop [10] or spark for distributed processing and anomaly detection.

Another model proposed for securing virtual environments is Virtualized Environment Security [5]. The aim is to provide security not only to the Virtual Machines but to the entire environment. The model gives security measures considering different dimensions of a virtualized environment. Some dimensions used for evaluating security measures include Management Dimension, Organization Dimension, Disaster Recovery Dimension and Technical Dimension. For instance, the Disaster Recovery Dimension mentions that backup and recovery policies should be clearly defined taking into consideration the critical factors [5].

Citrix XenServer [12] provides a robust set of virtualization technologies to help improve the resilience and usefulness of an organization's infrastructure, to ensure that not just critical resources remain available but that they are always accessible to all users in any location using any device. By implementing Citrix virtualization solutions, the risks due to both IT service outages and workforce continuity disruptions - such as diminished efficiency, lost revenue, missed opportunity, failure to meet service level agreements and customer defections - can be reduced, while simultaneously providing a wide range of benefits, including reduced complexity and costs, a stronger foundation for security and compliance, and greater adaptability to changing business conditions [3].

State-of-the-Art Summary

Paper Title	**State-of-the-Art Summary**
EagleEye: Towards Mandatory Security Monitoring in Virtualized Datacenter Environment [1]	In order to achieve on-demand security monitoring on a large-scale Virtualized Datacenter, the paper "EagleEye: Towards Mandatory Security Monitoring in Virtualized Datacenter Environment" came up with the technique of high-level representation replication to address the semantic gap and the inconsistent system state problems. This technique is powerful enough to deal with complex black-box mechanisms such as disk caching. The requirement for synchronous monitoring is supported by the stealthy hook mechanism, which is transparent (to the guest) and scalable. The authors proposed the deferred

	introspection technique as an enhancement of memory introspection to deal with inconsistent guest memory states due to on-demand paging or memory swapping. The goals of mandatory security monitoring prevent the use of guest kernel synchronization mechanisms to implement efficient "blockingwait" for security monitoring. The authors envisioned the "In-VM idle loop mechanism" to improve the performance of security monitoring due to the lack of such synchronization mechanisms [1].
Live Migration Impact on Virtual Datacenter Performance [2]	Live migration is one of the essential features in virtualized datacenter environments that drives virtual machine load balance, power saving and fault tolerance. In the paper "Live Migration Impact on Virtual Datacenter Performance" [2], empirical models supported by mathematical formulae are proposed to predict the live migration time, power consumption and network throughput before taking the VM migration decision.
Implementation of Xenserver to ensuring business continuity through power of virtualization for cloud computing [3]	Details on the techniques available to incorporate, administer, and automate a virtual data center allowing it to deliver cost-effective server consolidation and business permanence. The authors observe that Administration Pack for Citrix XenServer for Microsoft System Center Operations Manager effectively covers the complex mix and requirements of technologies in today's typical hybrid environments of data centers. The XenServer product improves on open source Xen primarily in the area of manageability. They have streamlined and automated common tasks while retaining most of the transparency of open source Xen.
Trusted Virtual Private Datacenter: A Model	The second model Trusted Virtual Private Datacenter was proposed for securing and managing cloud resources and

Toward Secure IaaS Cloud [4]	services. The objective was to propose a security mechanism considering the trusted relationship between the client and the IaaS provider. The model uses this relationship so as to allow both parties to set security controls to protect data and infrastructure within the cloud and virtualized data center.
A Virtualized Environment Security (VES) Model for a Secure Virtualized Environment [5]	VES model takes into consideration the security aspects of an entire environment and suggests measures for each dimension. The entire environment including the virtual machines and the virtual data centers should be secure enough to store and process data.
The Study on Data Security in Cloud Computing based on Virtualization [6]	Why using virtualization is beneficial and what are the security issues that are currently faced while using virtualized data centers. Some issues faced while managing virtual infrastructure include maintenance, configuration and updates for software, maintaining consistency and integrity, so on. The model provides a high degree of integrity, privacy and confidentiality- some major concerns of a virtualized data center.
Security Implications of Virtualization: A Literature Study [7]	Maintaining the performance of data center is also important factor since many computations are dependent on this. Also, recovery from failure should be handled carefully as one data center might be responsible for processing many different tasks simultaneously. Some security concerns for a virtualized environment are Confidentiality, Integrity, Authenticity, Availability and Non-Repudiation.
A Big Data Architecture for Large Scale Security Monitoring [8]	Some security concerns for a virtualized environment are Confidentiality, Integrity, Authenticity, Availability and Non-Repudiation [7]. A solution, which handles all these factors appropriately, can be considered as an excellent solution. The data center traffic is great source of information for security monitoring. The network traffic needs to be captured and

	analyzed using modern tools like big data tools [8].
Using Security Logs for Collecting and Reporting Technical Security Metrics [9]	The efficient Security metrics reporting system should be able to access raw security data sets (such as security logs) and have efficient drill-down functionalities – a generation of more specific reports on user defined queries, and the identification of individual entities in raw security data (such as log messages or Netflow records) [9]. Major security metrics includes Network IDS Alarm logs, NetFlow Data, server logs and other logs which can be used to monitor the security of the data center. The need of capturing the live data and process is needed for security monitoring.
Hadoop-Based Network Traffic Anomaly Detection in Backbone [10]	The need of capturing the live data and process is needed for security monitoring. These data are so large that it needs big data tools like Hadoop [10] or spark for distributed processing and anomaly detection.
vSphere from VM Ware [11]	Based on the prediction model results, the network admin can confirm the migration decision and move ahead with it. This paper provides a detailed study of impact of live migration, of datacenter resources network and power consumption for VMware environment [11].
xenServer from citrix [12]	Citrix XenServer [12] provides a robust set of virtualization technologies to help improve the resilience and usefulness of an organization's infrastructure, to ensure that not just critical resources remain available but that they are always accessible to all users in any location using any device.
Veeam ONE Availability Suite [13]	A pre-existing monitoring and reporting tool, “Veeam ONE Availability Suite” for VMWare VSphere and Microsoft Hyper-V, was instrumental in providing a detailed case study of 24x7 monitoring of virtual machines and environments, notification and alert mechanisms for failure and backup and troubleshooting

	capabilities to ensure that the system remains working and is up at all times. This study helped to describe the challenges, which can be faced by a system or system admin, while monitoring a VM or data center. The tool not only carries out monitoring tasks, but also provides alerts when any performance issues are encountered. This is an effective tool for monitoring as well as failure recovery and backup for the system.
VM Manager Plus [14]	Monitoring of critical performance metrics and providing VM statistics to the user, is an essential performance monitoring tool called "VM Manager Plus", which can increase the efficiency of the system by providing backup and alert notifications and help in easy recovery after failure. CPU, memory, and disk utilization, network usage, memory swap, data store read/write latency, memory ready etc. are the various but not limited metrics, which are monitored by the tool. If the monitored results show any discrepancy between the required and current values of the metrics, then remedial suggestions can be made and the recovery of the VMs can be carried out.
Netflix System Architectures for Personalization and Recommendation [15]	Recommendation systems are required to provide recommendations to the user based on results computed/trained by the system. The article "Netflix System Architectures for Personalization and Recommendation", provides information on the training of a system in order for it to give appropriate suggestions and recommendations to the user based on their previous historical data as well as interests. The training of the system can be done using machine learning algorithms and can be used to produce two computation models- Online computation and Offline computation. Online computation responds better to recent events and user interaction, but has to

	respond to requests in real-time. This can limit the computational complexity of the algorithms employed as well as the amount of data that can be processed. Offline computation has fewer limitations on the amount of data and the computational complexity of the algorithms since it runs in a batch manner with relaxed timing requirements. A combination of offline and online computations provides a better-trained model, which in turn can provide better recommendations.
Can We Beat DDoS Attacks in Clouds? [16]	Distributed Denial of Service attacks are a major problem faced by cloud companies today. Many researchers are working to find an optimal solution for detecting and mitigating these attacks. The research paper [16], "Can We Beat DDoS Attacks in Clouds?" gives information about how dynamically allocating idle resources from a cloud resource pool can be helpful in mitigating these attacks [16].
A Novel Protective Framework for Defeating HTTP-Based Denial of Service and Distributed Denial of Service Attacks [17]	Another approach of defeating DDoS attacks using entropy is proposed in the paper [17] "A Novel Protective Framework for Defeating HTTP-Based Denial of Service and Distributed Denial of Service Attacks". The framework is flexible, collaborative, multilayer, DDoS prevention framework and helps in detecting attacks using entropy very accurately [17].
An Online Anomaly Detection Method Based on a New Stationary Metric-Entropy Ratio [18]	Another interesting technique proposed in the article [18] "An Online Anomaly Detection Method Based on a New Stationary Metric- Entropy Ratio" is "LMS-FLP (Least-Mean Square-Forward Linear Predictor)". The detector is built to work online and has given accurate results in specific cases. The reliability of the detector is dependent on the proportion of malicious traffic in overall traffic [18].

Table 1 - State-of-the-Art summary

References

[1] Yu-Sung Wu, Pei-Keng Sun, Chun-Chi Huang, Sung-Jer Lu, Syu-Fang Lai and Yi-Yung Chen, "EagleEye: Towards Mandatory Security Monitoring in Virtualized Datacenter Environment", ISBN- 978-1-4673-6471-3, Dependable Systems and Networks (DSN), 2013 43rd Annual IEEE/IFIP International Conference on 24-27 June 2013.

The paper provides an on-demand, mandatory security monitoring system in Virtualized Datacenter environment, which won't depend on pre-installed guest components. The ultimate aim is to utilize, deploy, work and monitor the security in a large-scale Virtualized Datacenter cloud environment.

[2] Mohamed Esam Elsaid and Christoph Meinel, "Live Migration Impact on Virtual Datacenter Performance", 2014 International Conference on Future Internet of Things and Cloud.

Empirical models supported by mathematical formulae are proposed in this paper, to predict the live migration time, power consumption and network throughput before taking the VM migration decision. Based on the prediction model results, the network admin can confirm the migration decision and move ahead with it.

[3] N.Staalinprasannah and S.Suriya II, "Implementation of Xenserver to ensuring business continuity through power of virtualization for cloud computing", IEEE - 31661.

This paper provides details on the techniques available to incorporate, administer, and automate a virtual data center allowing it to deliver cost-effective server consolidation and business permanence.

[4] Xin Wan, ZhiTing Xiao and Yi Ren, "Trusted Virtual Private Datacenter: A Model Toward Secure IaaS Cloud ", 2012 Fourth International Conference on Multimedia Information Networking and Security.

The paper proposed architecture composed of IaaS data center, which provided an infrastructure to host multiple customers, trusted virtual clusters and trusted virtual machines, which are used for processing the customer requests.

[5] Annette Tolnai and Sebastiaan von Solms, "A Virtualized Environment Security (VES) Model for a Secure Virtualized Environment", 2010 Conference for Internet Technology and Secured Transactions (ICITST).

This paper proposed a "VES model" which takes into consideration the security aspects of an entire environment and suggests measures for each dimension. The entire environment including the virtual machines and the virtual data centers should be secure enough to store and process data.

[6] Fu Wen and Li xiang, "The Study on Data Security in Cloud Computing based on Virtualization", IT in Medicine and Education (ITME), 2011 International Symposium on, Volume2.

This paper highlights some issues faced while managing virtual infrastructure, which include maintenance, configuration and updates for software, maintaining consistency and integrity.

[7] Andre van Cleeff, Wolter Pieters, and Roel Wieringa, "Security Implications of Virtualization: A Literature Study", 2009 International Conference on Computational Science and Engineering

This paper provides the implications of using virtualized data centers and why virtualization is beneficial from security point of view and why using virtualization is beneficial and what are the security issues that are currently faced while using virtualized data centers

[8] Samuel Marchal, Xiuyan Jiang, Radu State, Thomas Engel "A Big Data Architecture for

Large Scale Security Monitoring", 2014 IEEE International Congress on Big Data

This paper highlights that the data center traffic is great source of information for security monitoring. The network traffic needs to be captured and analyzed using modern tools like big data tools.

[9] Risto Vaarandi, Mauno Pihelgas, "Using Security Logs for Collecting and Reporting Technical Security Metrics", 2014 IEEE Military Communications Conference

This paper provides insight into the efficient Security metrics reporting system to access raw security data sets (such as security logs) and have efficient drill-down functionalities and the identification of individual entities in raw security data (such as log messages or Netflow records).

[10] Jishen Yu, Feng Liu, Wenli Zhou, Hua Y, "Hadoop-Based Network Traffic Anomaly Detection in Backbone", Proceedings of CCIS2014

This paper asserts that data used in applications is so large that it needs big data tools like Hadoop or spark for distributed processing and anomaly detection.

[11] vSphere from VM Ware http://www.vmware.com/products/vsphere

This study provides a detailed study of impact of live migration, of datacenter resources network and power consumption for VMware environment.

[12] xenServer from citrix http://xenserver.org/open-source-virtualization-download.html

This is a downloading site for citrix xenserver.

[13] Veeam ONE Availability Suite- http://www.veeam.com/virtualization-management-one solution.html

This study entails a powerful monitoring and reporting tool for VMWare VSphere and Microsoft Hyper-V called Veeam ONE.

[14] VM Manager Plus- https://www.manageengine.com/virtualization-management/

This study provides details about a free virtualization monitoring software, called VM Manager Plus monitors unlimited VMs, hosts, and data stores across VMware, Hyper-V and XenServer platforms.

[15] Netflix System Architectures for Personalization and Recommendation- http://techblog.netflix.com/2013_03_01_archive.html

The article provides information on how to create a recommendation software architecture that can deliver fast recommendations to the users of the system.

[16] Shui Yu, Yonghong Tian, Song Guo and Dapeng Oliver Wu, "Can We Beat DDoS Attacks in Clouds?, IEEE Transactions On Parallel And Distributed Systems, Vol. 25, No. 9, September 2014

The paper focuses on proposing a DDoS mitigation strategy. The strategy proposed in this paper is dynamic resource allocation when a system is under DDoS attack.

[17] Mohammed A. Saleh and Azizah Abdul Manaf, "A Novel Protective Framework for Defeating HTTP-Based Denial of Service and Distribute Denial of Service Attacks", The Scientific World Journal Volume 2015 (2015), Article ID 238230.

In this article, the authors have proposed a framework called as FCMDPF (Flexible Collaborative Multilayer DDoS Prevention Framework) which helps in defeating the DDoS attacks by using entropy.

[18] Ziyu Wang, Jiahai Yang, and Fuliang Li. "An on-line anomaly detection method based on a new stationary metric - entropy-ratio", Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on, pages 90–97, Sept 2014.

The paper focuses on proposing an anomaly detector called as LMS-FLP- Least Mean Square-Forward Linear Predictor, which works online for detecting malicious users using the entropy-ratio.

[19] Matrix factorization algorithm from http://www.maths.uq.edu.au/~gjm/nhnrm_spl11.pdf

The document describes machine learning algorithm. Machine algorithm is known as matrix factorization algorithm.

Chapter 2. Project Justification

Monitoring datacenter performance - a synergy of server performance, application performance, and security threats, is a great challenge. The challenge rises multifold when disparate virtualization platforms and monitoring tools are used. Unified approach of data center management and security monitoring is an essential requirement for the enterprises. The proposed solution gives a unified solution to **manage**, **collaborate** and **monitor** heterogeneous datacenter.

2.1 Virtualization Abstraction Interface (VAI):

The proposed solution will provide an abstraction interface to manage datacenters with disparate virtualization infrastructure. Users can use a single console to monitor and manage their datacenters. The solution will support VMWare, Xenserver in the initial phase, which can be extended to other virtualization platform in a similar fashion.

2.2 Access Control List support:

The proposed solution will provide access to the monitoring resources based on access control policies. ACL policy will include role based access control and attribute-based access control.

2.3 Anomaly detection using resource consumption analysis (RCA):

The solution will include an unparalleled feature called resource consumption analysis (RCA). Using RCA, the system can predict abnormal behavior inside the datacenter by analyzing combined data from system log, resource usage and network activity. Example: If the system starts consuming more CPU and memory than what it normally uses then the analysis algorithm will process the data from system log and network activity to find the cause of the spike and provide an alarm for any suspicious activity.

2.4 Graphical user interface with proper dashboard:

The solution will present resource usage like CPU usage, memory usage, swap memory usage, power state and security threats statistics though graphical user interface. Features like addition of new user, addition of new datacenter, configuring the monitoring parameters, management of the user's role and permission will be part of the user interface.

2.5 Failure detection and auto recovery of VMs:

Detection of virtual machine failures and recovery will be an advantageous feature for the users. The solution will monitor the health of the individual VMs to detect the system failure and execute the appropriate steps to recover it.

2.6 Alert mechanism:

The proposed solution will generate alarms for the unfavorable situations like unauthorized user activity, malicious access and network attack like denial of service (DDoS) from outside parties.

Chapter 3. Project Requirements

The project requirements are as follows:

- User Story
- Requirement List
- Use Case

User Story:

User story is a tool used in agile software development to capture description of a software feature from an end-user perspective. A user story helps to create a simplified description of a requirement.

- As a user I can login to cloud based datacenter management service.
- As a user I can add and manage datacenters with disparate virtualization environment using a single console.
- As a user I can see the system details like IP address, power state, etc. and resource usage like CPU usage, memory usage and disk usage.
- As a user I can configure the monitoring parameters, type of security threats that user wants to monitor.
- As a user I can manage and query the activity log of the proposed service.
- As a user I can see the generated alarm, for a resource that crosses the threshold limit.
- As a user I can add and manage a new data center in easy and effortless manner.
- As a user I can power on/off my server in the data center using service functionality.
- As a user I can see the network security statistics for my data center.
- As a user I can access the resource information based on my role.

Requirement list:

The project requirements are arranged based on functionality; example: requirement under user interface and configuration, security, interface requirement, statistics, logging, performance, etc.

Requirements are ranked as per priority of the task and mentioned in table-1. These are divided in the following three categories:

1. Essential: Mandatory requirements come under this category, which need to be agreed upon in a consensus.
2. Desired: These requirements will enhance the software product usage, these are the second in line of required features for the customer.
3. Optional: Optional requirements are nice-to-have features. These requirements are not necessary for the software functionality.

Sl. No	Requirement Type	ID	Description
Essential Requirements			
3.1	**User Interface and Configuration**	REQ_1	The proposed software solution shall provide an login interface to the hosted service
		REQ_2	User shall be able to see only their own virtual machines under there login
		REQ_3	Administrative user shall be able to add new user and assign access control policy.
		REQ_4	The solution shall provide a user interface to create profile to configure threshold parameters
3.2	**Security**	REQ_5	The multi-tenant solution shall provide the secured data access, where only authorized users can access their own data center information
		REQ_6	The solution shall provide role based account for a given client. Example include administrator, audit user, etc.
3.3	**Access Interface**	REQ_7	The proposed solution shall provide the secured interface to transfer the data from the data center premises to hosted solution for the monitoring and analysis purpose
3.4	**Request Processing**	REQ_8	The proposed solution shall be able to monitor the availability of the virtual machine in the client data center
		REQ_9	The proposed solution shall be able to detect the failure of the virtual machine and recover it back in the client data center, if configured in the profile
		REQ_10	The proposed service shall provide the functionality to detect the http flood, to keep the service running all the time
		REQ_11	The solution shall be able to collect CPU usage, Memory Usage and Disk usage for each system inside a data center in the configured interval to analyze the resource usage
		REQ_12	The proposed solution shall be able to recommend the additional resource requirement. Example: if the system usage is high then it will recommend to add new system for their processing

		REQ_13	The solution shall give the capability to switch on/off the server remotely
		REQ_14	The solution shall allow creation of different threshold profile for the application monitoring and network security monitoring
		REQ_15	The proposed solution shall be able to scale based on the number of request. It should be able to increase the request handling capacity to 50% more when it reached 70% of the configured load
		REQ_16	The solution shall be able to collect the network activity and system log for security analysis
		REQ_17	The solution shall be able to give provide faster analysis by using tools like apache spark
3.5	**Statistics**	REQ_18	The solution shall provide the interface to visualize server availability
		REQ_19	Authorized user shall be able to see the system details and resource usage like CPU, Memory, Disk usage
		REQ_20	Authorized user shall be able to see the security statistics like number HTTP flood attacks
		REQ_21	User shall be able to see graphical representation of the request from suspicious region
		REQ_22	User should be able to see the statistics of the file system permission grant
Desired Requirements			
3.6	**Configuration parameters**	REQ_23	User shall be able to select or view the monitoring parameters
3.7	**Logging**	REQ_24	The solution shall be able to access the hosted service log to track the previous activity in the service
		REQ_25	Administrator should be able access the previous 7 days of activity log
3.8	**Performance**	REQ_26	The application shall be able to handle 50GBs of data with no degradation in performance
3.9	**Reliability**	REQ_27	The solution shall provide better level of reliability by making the application fault tolerant
3.1	**Availability**	REQ_28	Service shall be available for 99.99% percentage of the time
Optional Requirements			
3.11	**Billing**	REQ_29	User shall be able to configure the billing type
		REQ_30	User shall have the tools to see the billing details
3.12	**Training and documentation**	REQ_31	User manual with the complete steps of configuration and usage should be developed
		REQ_32	Smart client development

Table 2- List of requirements for DMaaS service development

Use Case:

Use Case -1: Company A acquires company B with different IT environment

Let's say company A, using set of IT management tools and virtualized environment acquires company B with different set of IT environment. In this situation parent company need to update one of the data center tools to make it unified or need a solution which can manage both datacenters without any change. Scenario depicted in figure -1.

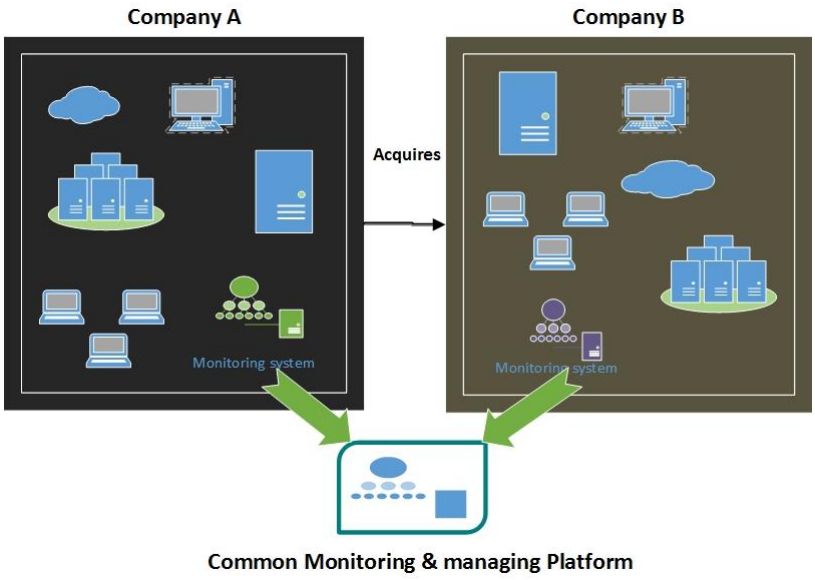


Figure 1: Use case diagram

Use Case -2: Company has multiple datacenters in different geographical locations with different IT environment. Figure -2 show depicts the required scenario.

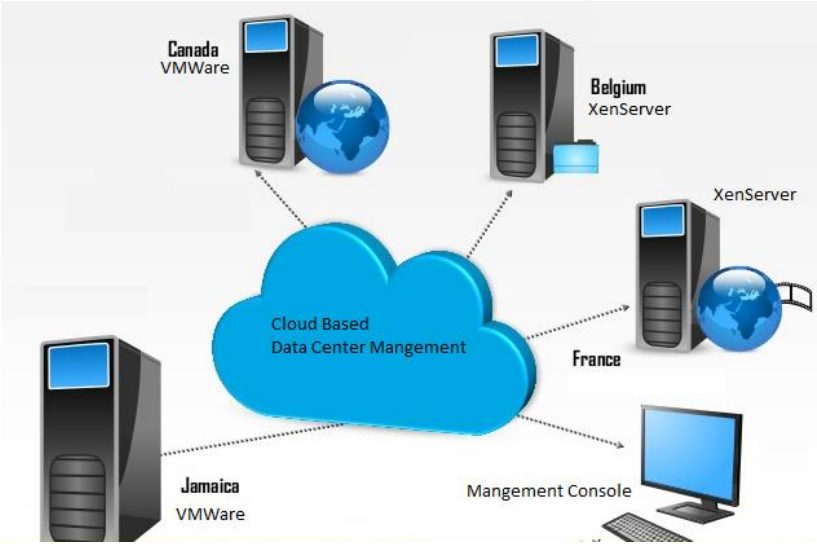


Figure 2: Use case 2

Company A has multiple datacenters with different capabilities and different set of management environments. Managing all the data centers in unified manner will be challenging in terms of resources and security. If the company uses a solution as proposed by us, they can manage their multiple data centers and their security in unified manner by controlling it from single location.

Chapter 4. Dependencies and Deliverables

Dependencies

- Single point of failure - As multiple datacenters and their security will be controlled in a unified manner using the Cloud based data center management node, there arises a single point of failure which could bring the entire system to a halt, if it fails.
- Hardware capacity of system should be high - The system will be processing Big data, so the capacity of the system should not fail to store and process the data.
- Ability to connect to data center - This will depend upon two factors:
    - Internet connectivity
    - Permissions/Licenses

Deliverables

- Survey paper for publication
- Fully developed and tested product ready to use by a company.

Chapter 5. Project Architecture

The proposed solution will handle large amount of data to process and monitor the resource usage and threats in the data center. The solution will be implemented based on the below designed architecture; namely "Multi-tenant Service Oriented Architecture".

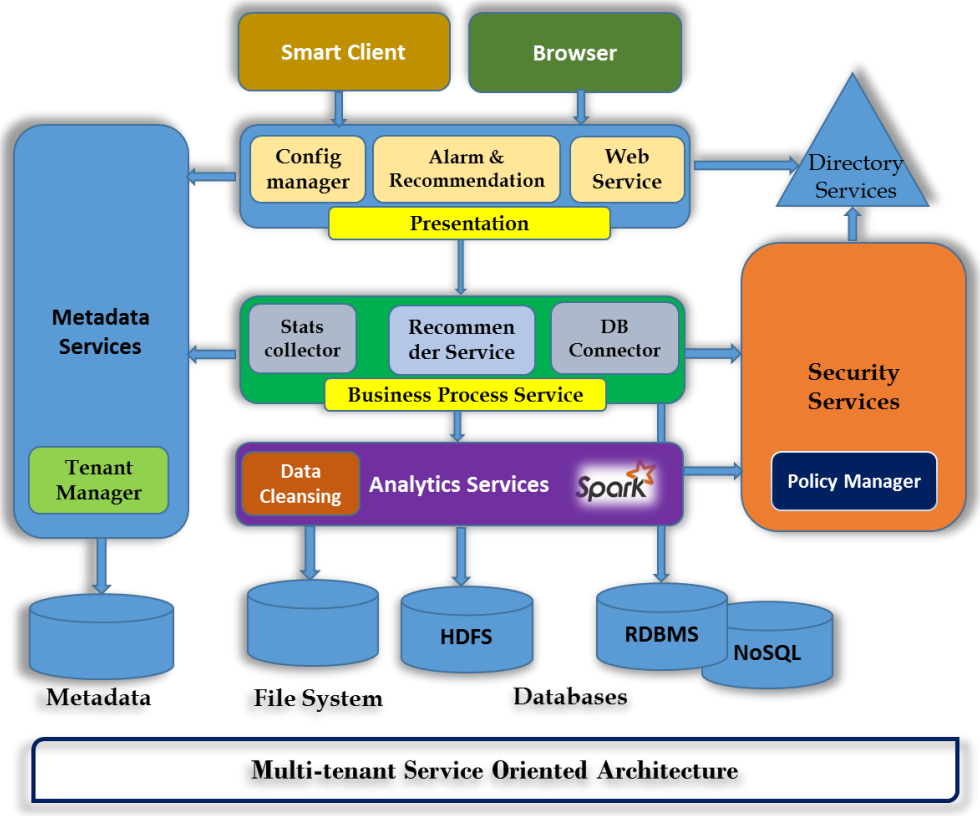


Figure 3 : DMaaS Architecture block diagram

The proposed solution comes under the SaaS model. We have designed our model to utilize the **"Service oriented architecture in a multi-tenant"** environment. Figure-3, proposed architecture has better **performance**, is **scalable** and **cost effective** in comparison to traditional MVC architecture. The above depicted components are explained below:

Metadata Services: It provides the primary means of customization and configuring the application to meet customer need. This design allows customers to configure the service, monitoring parameters and work-flow and business rule. The major component will be tenant manager, which will ensure the sanity among users from different enterprises.

Smart client: Smart client, is a mobile application to visualize given statistics of the datacenter. Currently enterprises want to track their resource usage and anomalies in most convenient manner.

Browser: This is a web client to connect to the datacenter management service in a secured manner and able to visualize the information and statistics on a computer screen.

Presentation Layer: This layer provide the access to the service configuration, alarms, and recommendation using modern web service model. Third party charting tool will be integrated to provide data visualization in a most effective manner.

Directory Services: Directory service will provide the access control functionality to the service. This can include role based authentication, attribute based authentication, etc. Each user will have attached profile based roles and responsibilities.

Business Process Service: This layer provides the functionality to connect to the datacenter to collect system logs, network log, and interaction with the hypervisors for collecting usage metrics. The recommendation engine will be part of the business process layer and will interact with the analytics layer to query required data and provide recommendation using these data. These services would be implemented as REST web services and/or SOAP based services depending on the need.

Analytics Services: Data cleansing and analysis is the primary functionality of this service layer. Analytics on the resource usage and network data will be done. We will be using spark engine for analysis of large amount of data most efficient manner by using distributed computation. This will interact with both business process layer and data storage layer.

Database: This layer take account of various type of database that we are going to use. We will be using HDFS for the data analysis, NoSQL for storing the activity log and Profile related log in the RDBMS or NoSQL system.

Security service module: This service will have the functionality to monitor application security and network security. Application security will be measured by taking into account the resources usage pattern. In terms of the network security, it will monitor network activity and permission changes in the file system and syslog.

Chapter 6. Project Design

DMaaS cloud hosted service provides a multi-tenant based datacenter management service, users can create/delete virtual machines, find recommendation on resource usage and create pattern in secured environment.

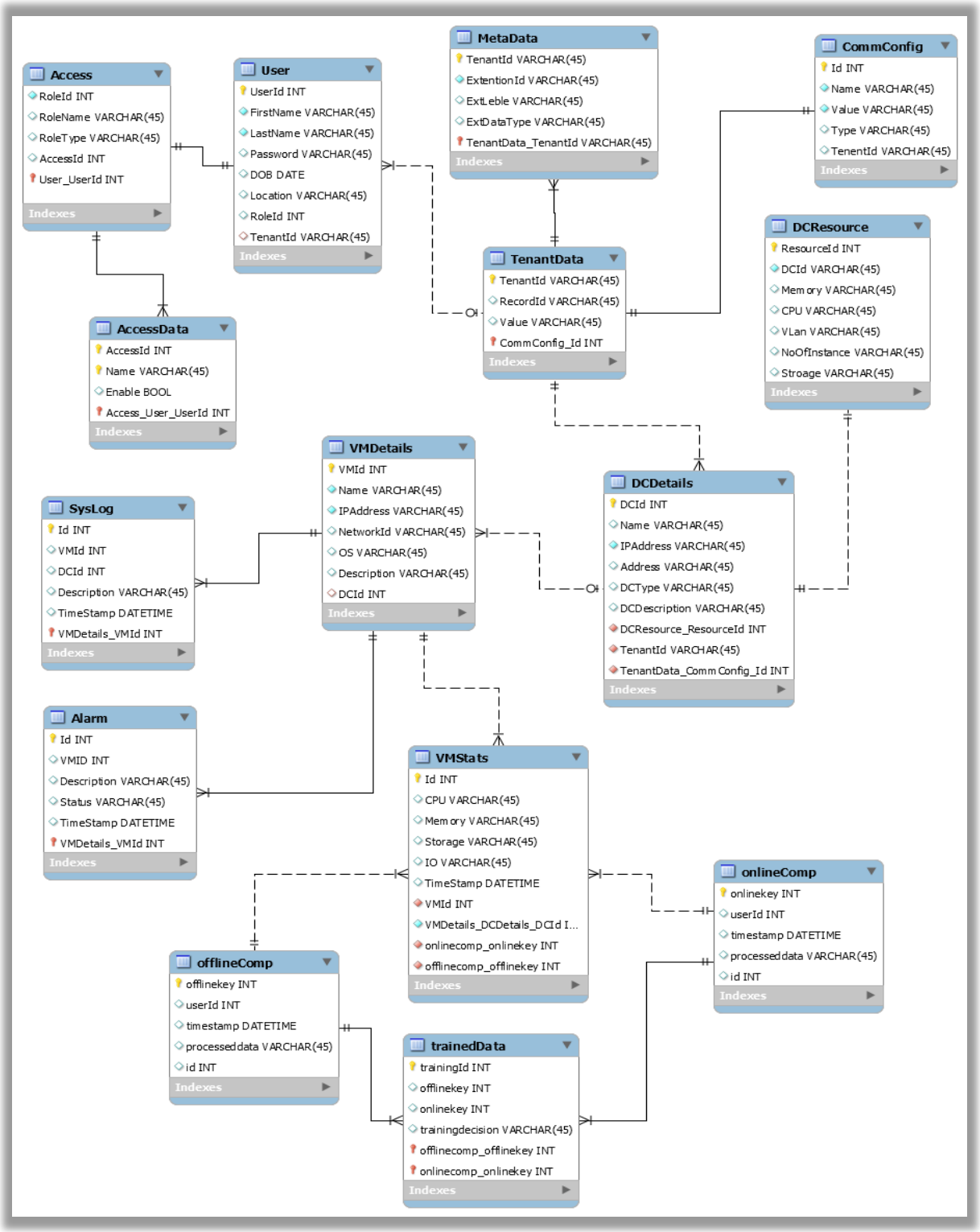


Figure 4 : DMaaS Entity relation diagram

Figure - 4 gives the entity-relationship diagram for DMaaS. The VMDetails, VMStats table is responsible for storing all the VM related statistics for a user. The user details will be stored in the User table. For differentiating between user roles, the Access will be used. Datacenter information will be maintained in DCDetails and DCResources tables. For Recommendation System, the tables used will be offlineComp, onlineComp, trainedData. The Alarm table will be used for storing the Alarms created along with their status.

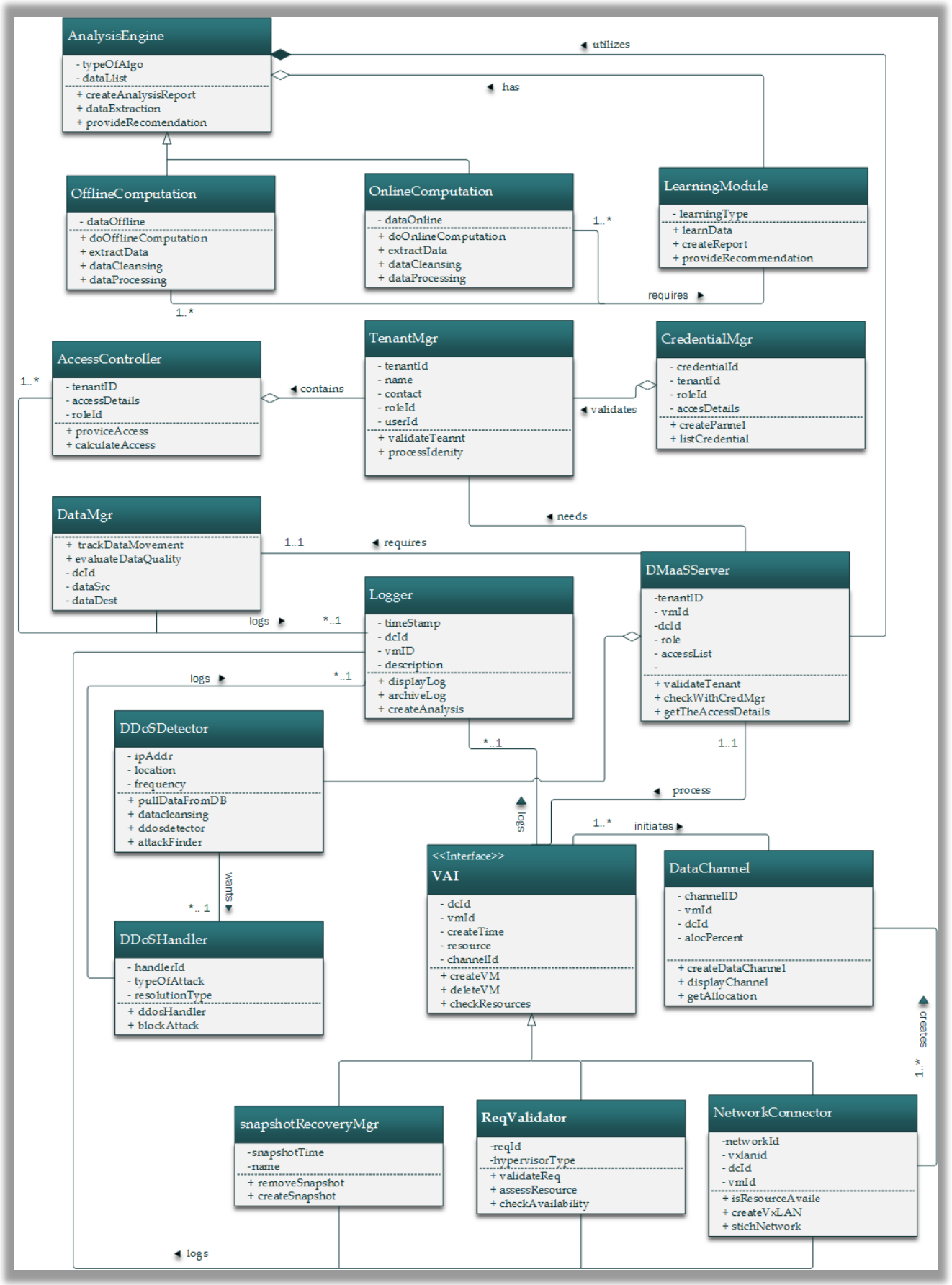


Figure 5 : DMaaS Class diagram

Figure - 5, gives the class layout for DMaaS. The AccessController, TenantManager and CredentialMgr classes will be used for handling Access Control Lists. DMaaSServer is the main class connecting all the modules (AccessControl, DDoS Detection, Virtual Abstraction Interface and Recommendation System) together. The VAI will provide an interface for snapshotRecoveryMgr, ReqValidator and NetworkConnector classes. The DDoSDetector and DDoSHandler will be responsible for detecting and mitigating DDoS attacks. For the recommendation system, classes used will be AnalysisEngine, which has the LearningModule, and will help OnlineComputation and OfflineComputation classes. The DataMgr and Logger will be responsible for handling logging requests for the system.

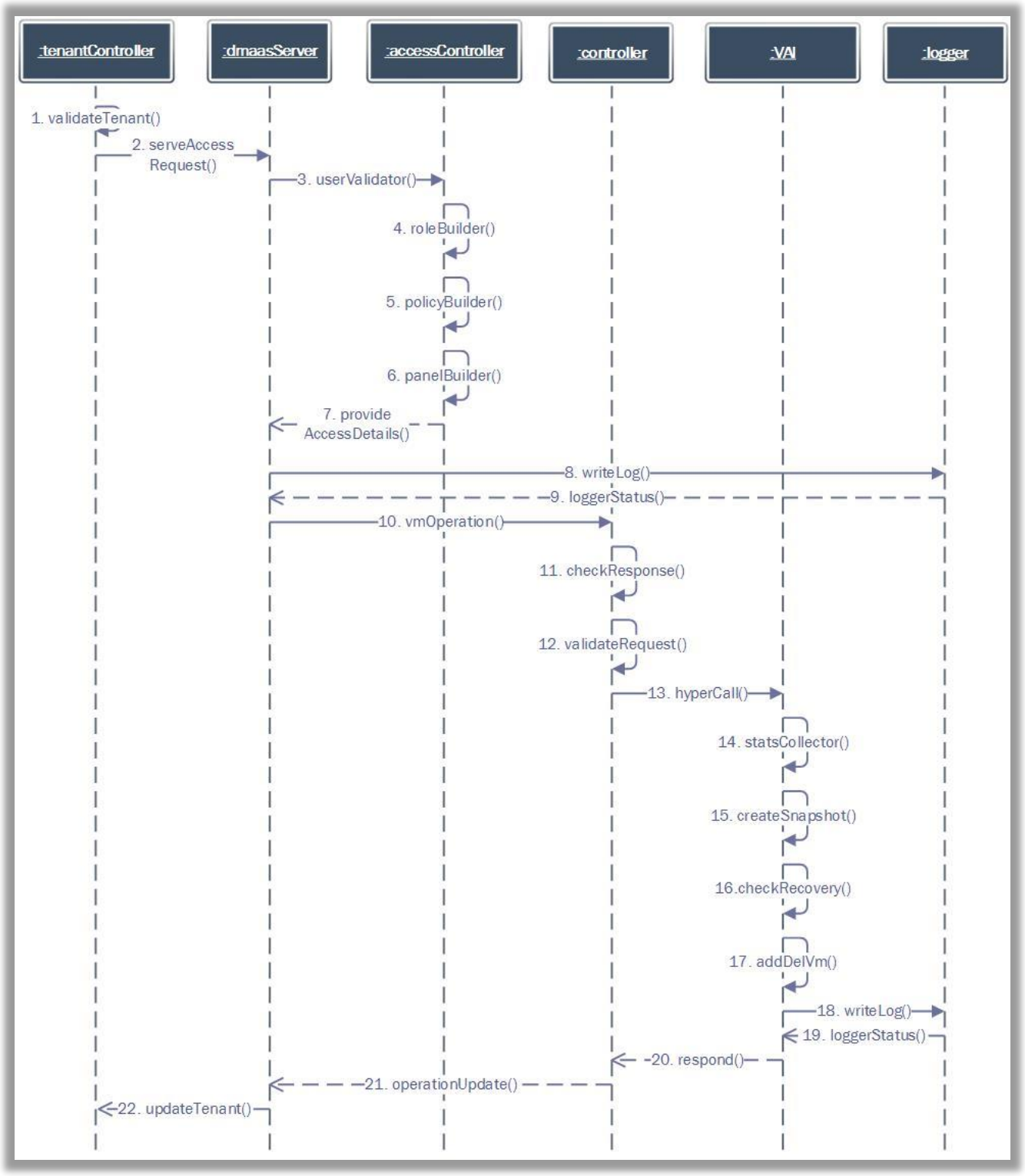


Figure 6: Sequence diagram for accessing a service for VM operation

Figure - 6 describes the sequence diagram for accessing service for hyper operations. When a user attempts to login, first step is to validate the tenant. The second stage of validation is access and role validation of the user. When a user is authenticated with the generated secure token VIA controller, then the hyper call can be initiated. Hyper call includes the stats collection, VM addition, VM deletion, snapshot creation, VM recover. After the hyper call takes place, the response will be

written to logger in step 18 and step 19. The response is again reported to tenant and to do DMaaS follow from step 20 to step 22.

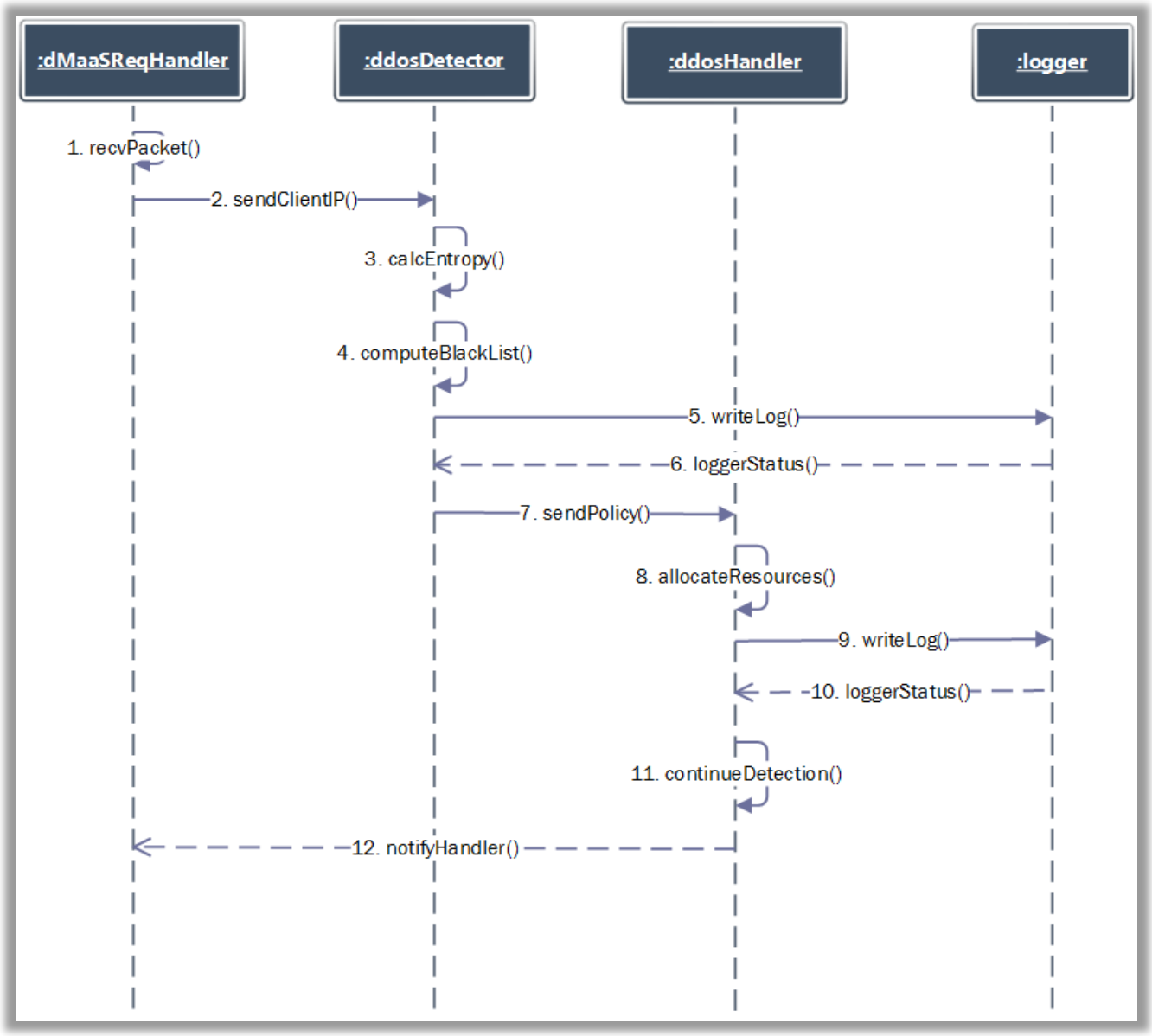


Figure 7 : Sequence diagram for DDoS

Figure - 7 gives a sequence diagram for the DDoS detection and mitigation module. The DMaaS Request Handler receives a packet and sends the clientIP to the DDoS Detector. The DDoS Detector then calculates entropy and checks the BlackList. The status is written to the logger, which sends back the loggerStatus. The DDoS detector then sends the policy to the DDoS Handler, which is responsible for mitigating the attack. The Handler will allocate resources dynamically and will write the status to the log. The process of detection will continue and the DMaaS Handler will get notified.

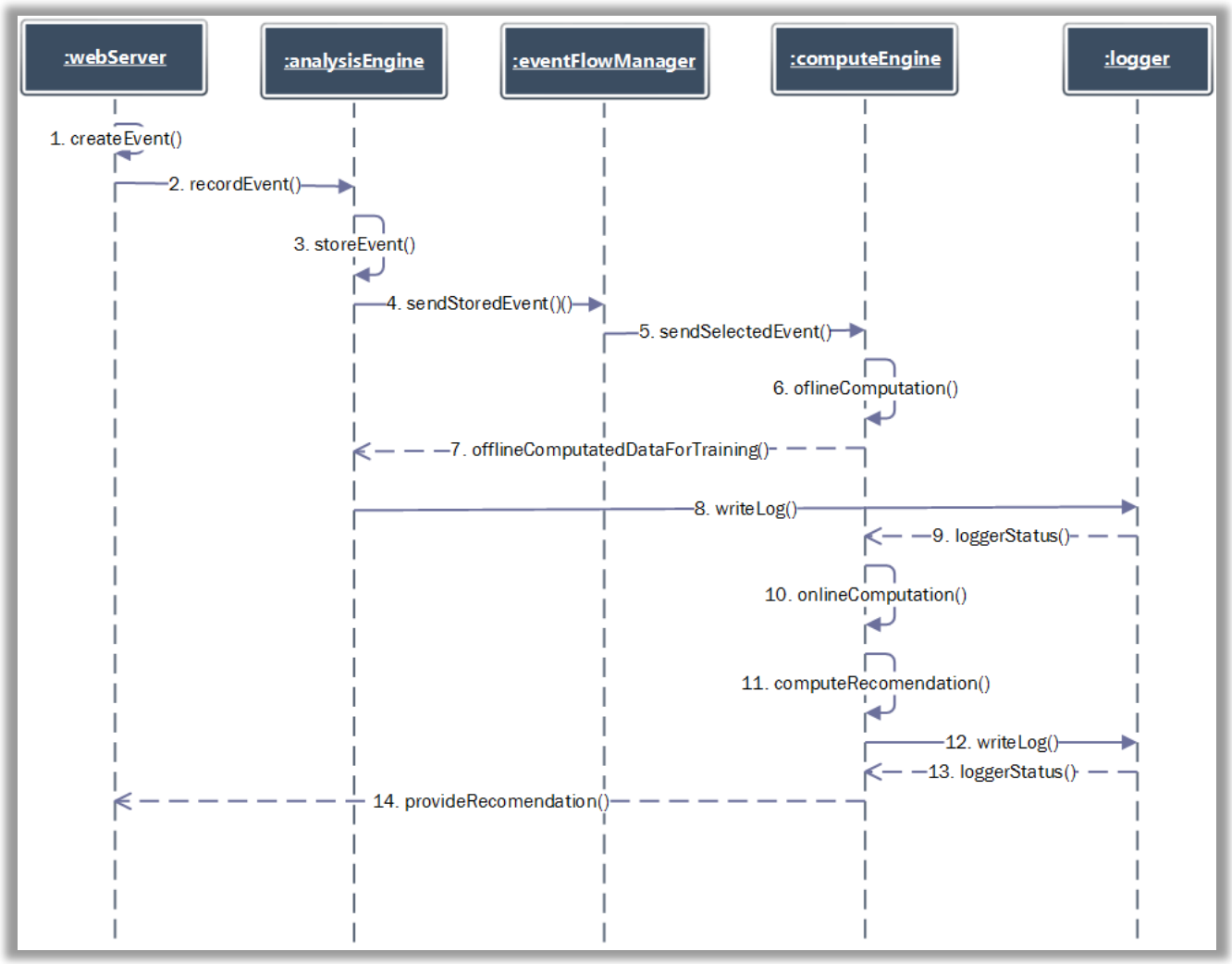


Figure 8: Sequence diagram for recommendation

As per figure - 8, webserver will create events and send the recorded events to the Analysis Engine. The analysis engine will store the events and send these events to the Event Flow Manger. The Event Flow Manager will send selected events to the ComputeEngine for offline computation. The ComputeEngine will send the offline computed data for training back to the Analysis Engine.

The analysis engine will then write log and the logger will send the status to the ComputeEngine. The ComputeEngine will then perform online computation and computations to get recommendations. The recommendations computed will be logged. The webserver will then be provided with these recommendations.

Mockup User Interface:

Expected user interface for this service has depicted below. All web pages are tentative web pages. This includes login page, registration page, VM details, VM statistics, DDoS reporting page, recommendation page, etc.

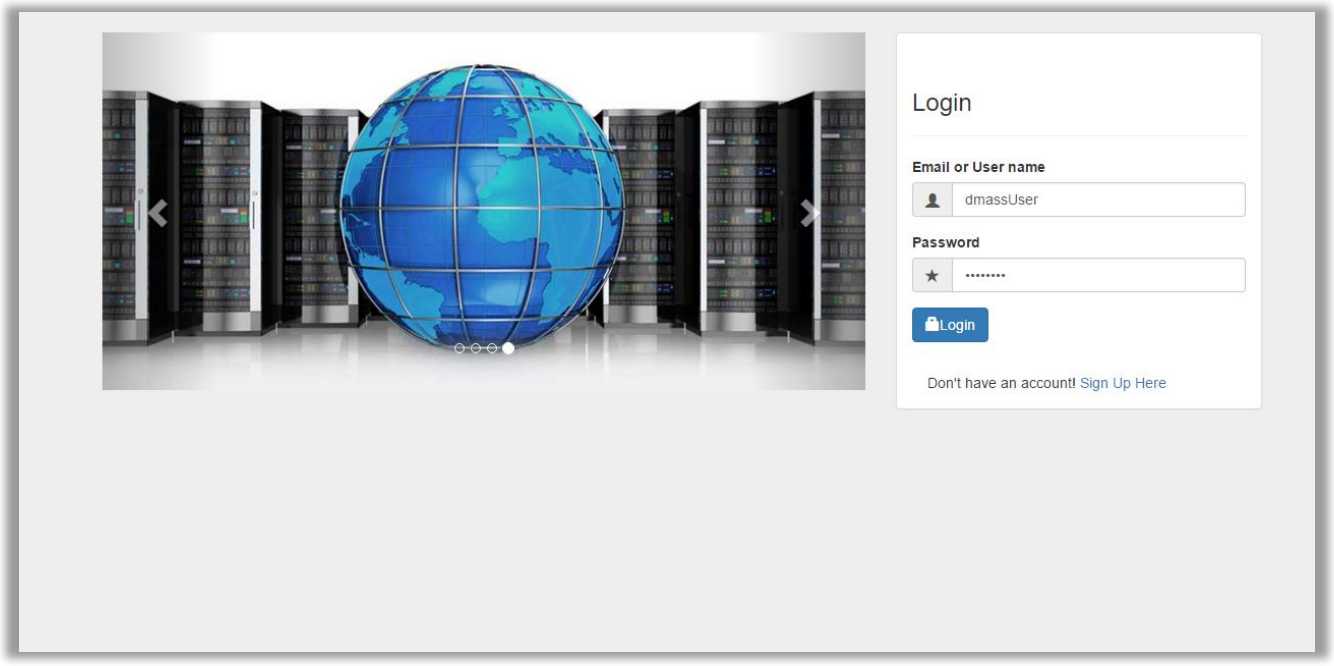


Figure 9: Mock up - DMaaS Login page

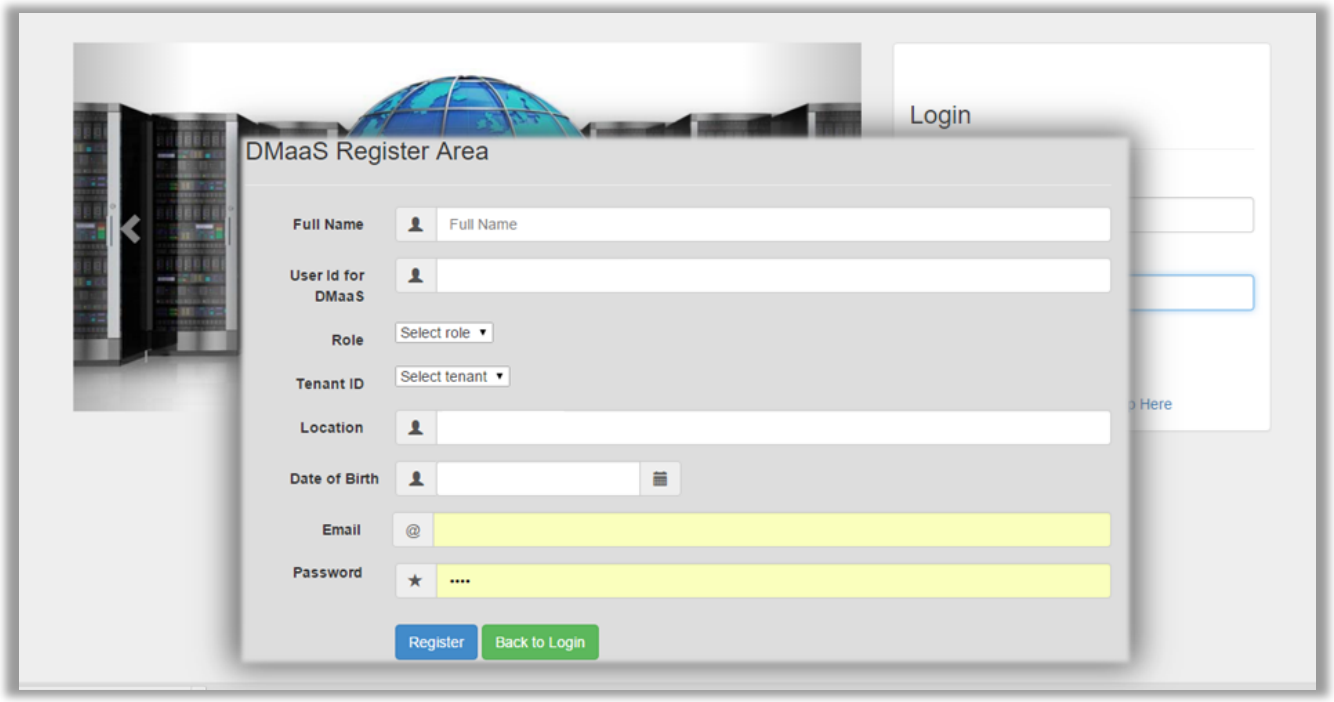


Figure 10: Mock up - Register Page

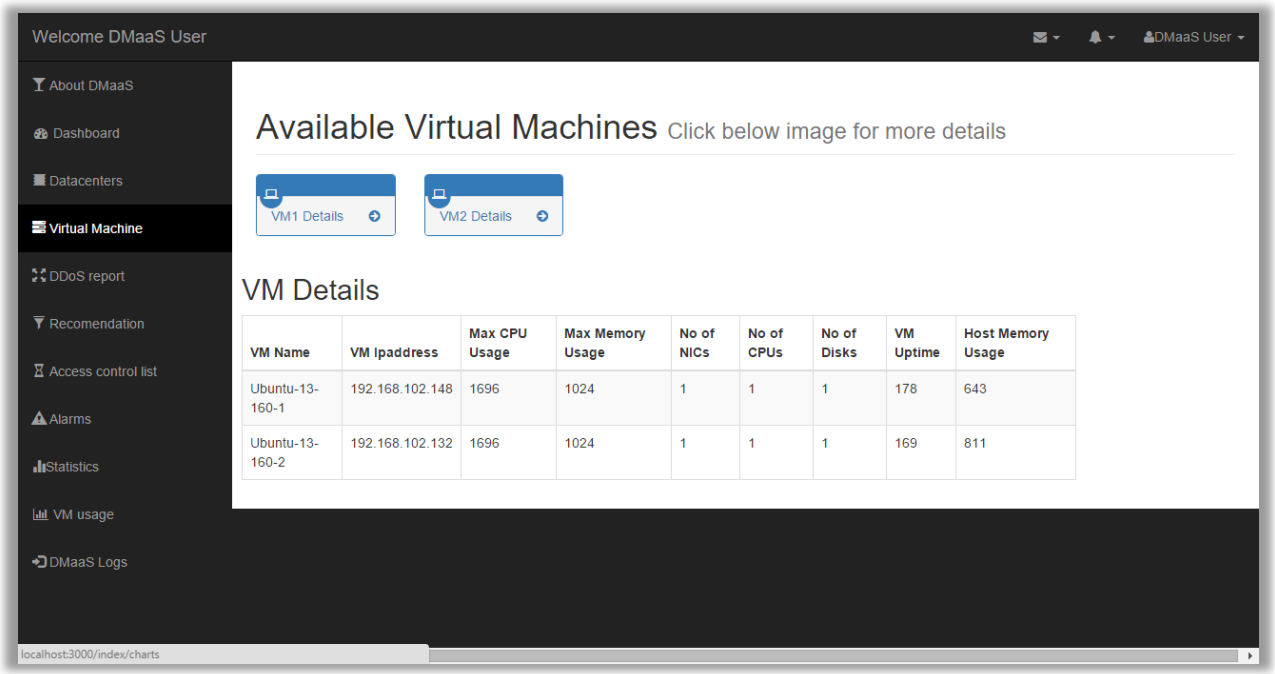


Figure 11: Virtual machine list and details page

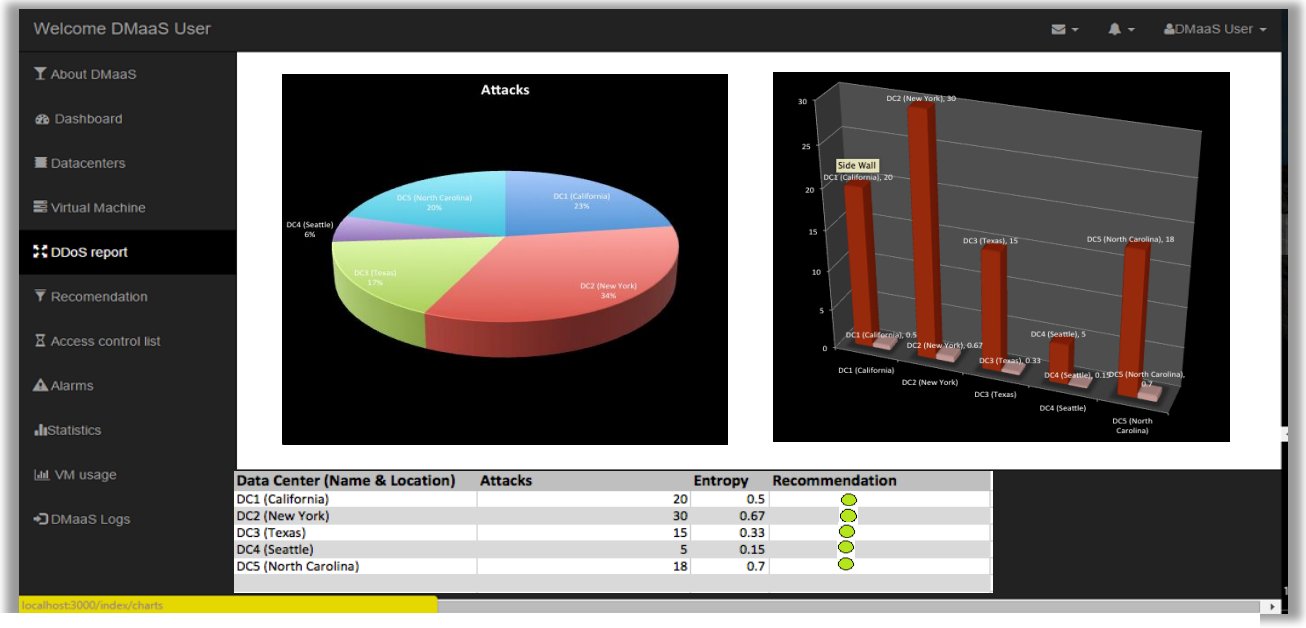


Figure 12: DDoS report page for DMaaS Service

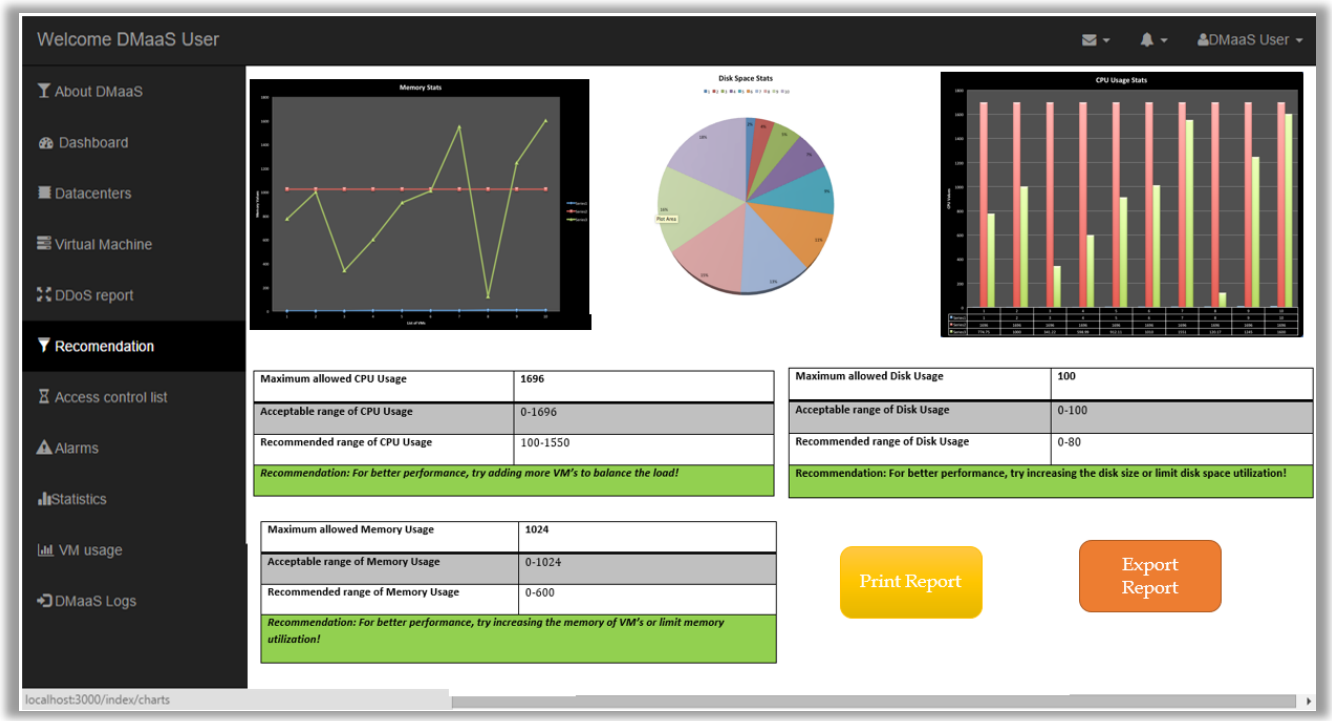


Figure 13: Recommendation page for the DMaaS service

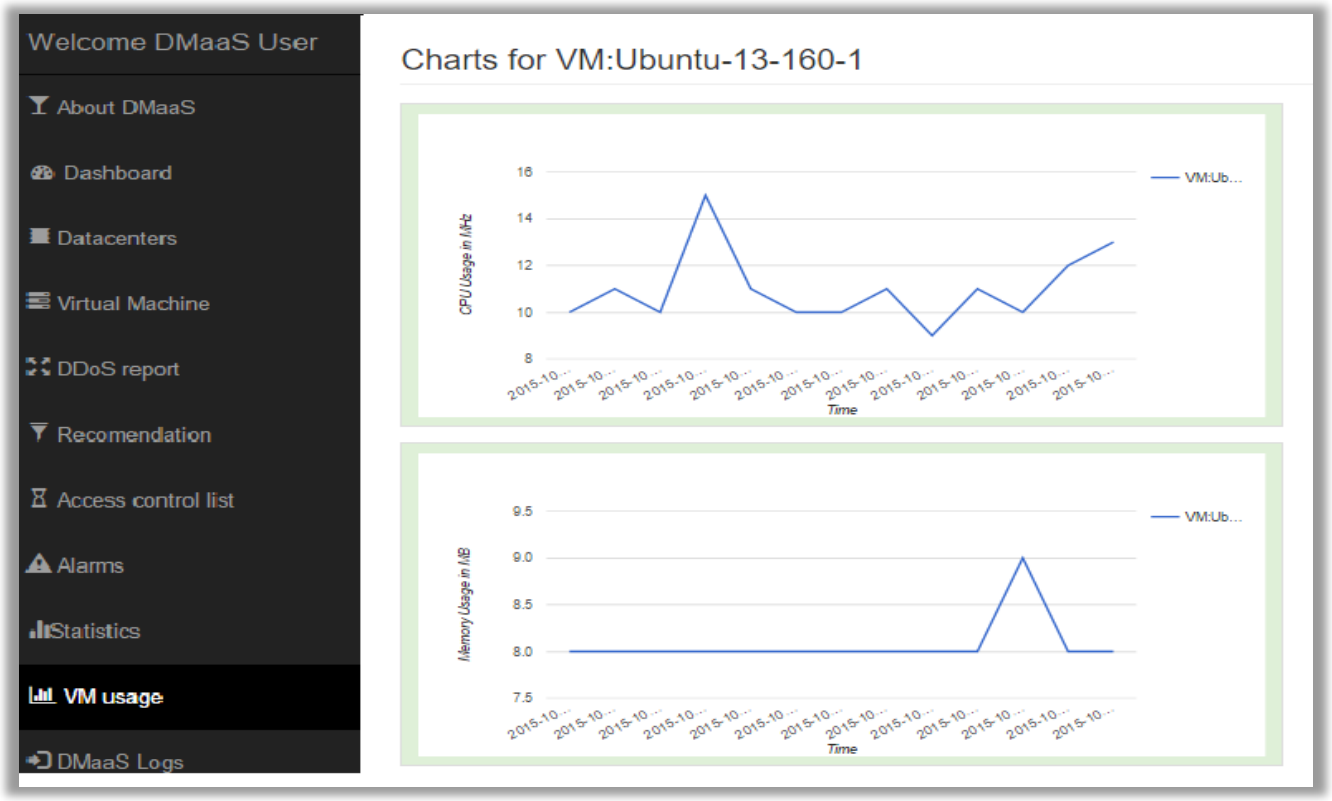


Figure 14: virtual machine usage page

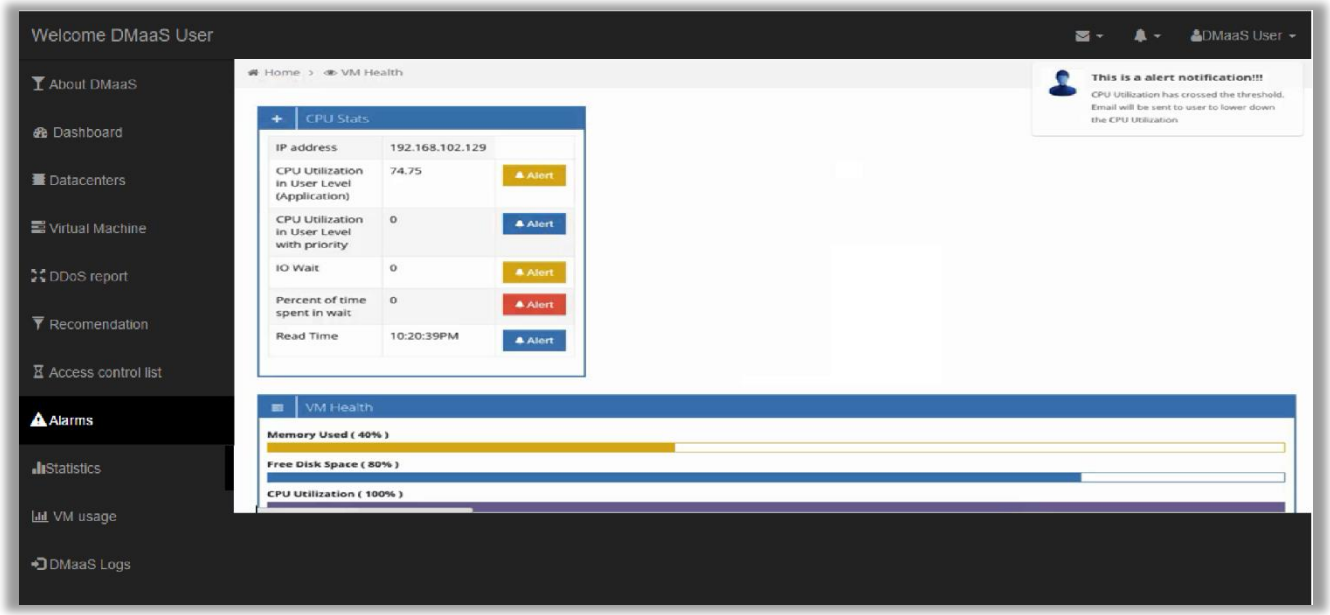


Figure 15: Alarm Page for DMaaS

Figure 16: Stats page for DMaaS

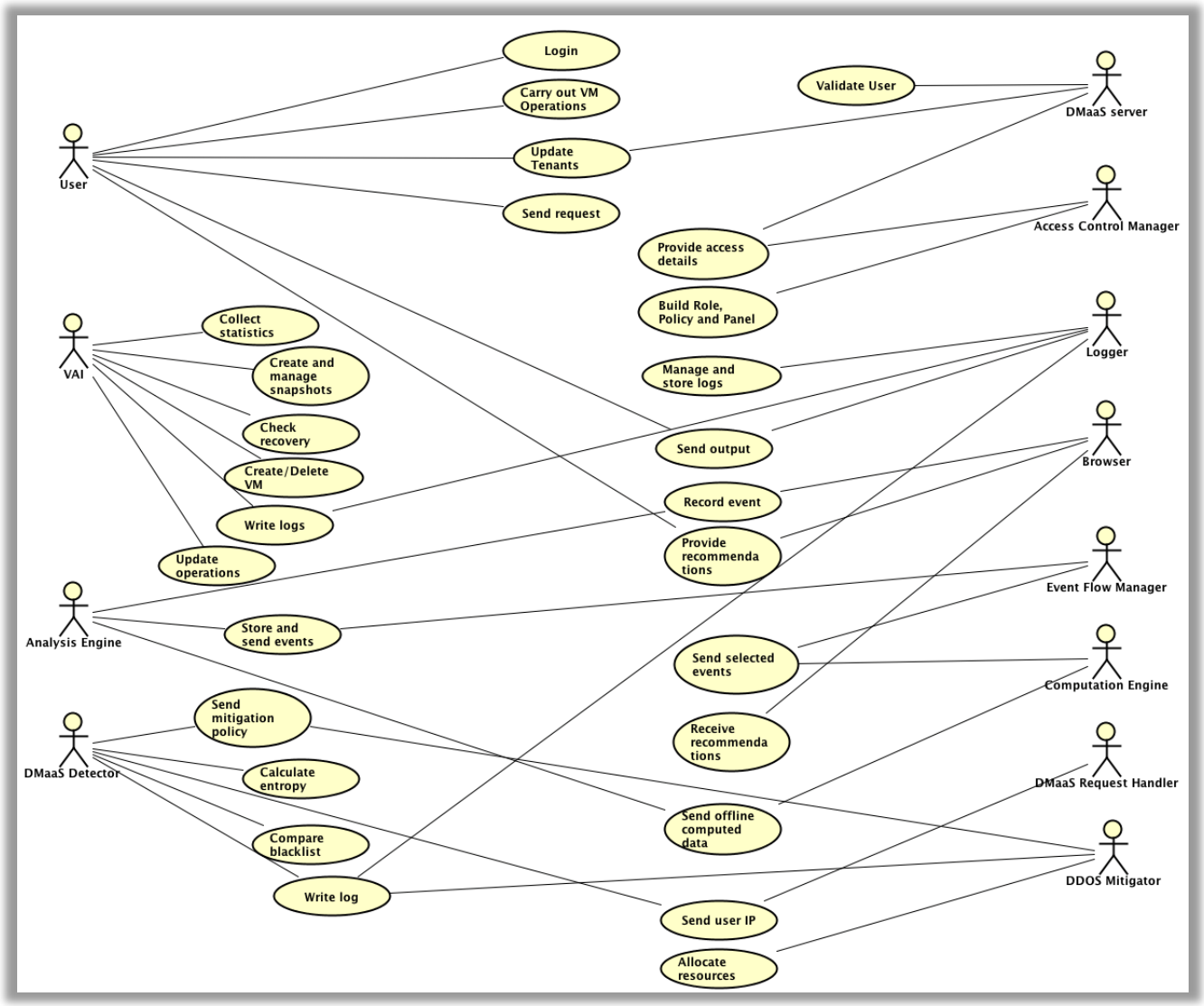


Figure 17: Use case diagram for DMaaS service

Figure 17 depicts the use case diagram for DMaaS server. Here users include VAI, Analysis Engine, DMaaS detector, DDoS handler, DMaaS request handler, computation engine browsers and logger. All the events and major functionality is show above.

Chapter 7. QA, Performance, Deployment Plan

Quality Assurance and performance measurement are most vital step between development and project delivery. The following table presents the various testing strategy that will be followed with details like what will be part of the coverage and benefit of the test.

Test Strategy	Test Coverage	Benefit
Unit testing	Unit testing will carried by each individual team members on their part of module. This test require deep knowledge of the implementation. The test case will cover each line code of the function. This will help is remove the logical errors in the code.	Logical errors and faults can be identified and addresses in the module level
Functional testing	Each team members will run set of test cases related to their functionality. This major goal of this testing is to identify the logical errors in the component level. Complete setup is not required for this test. Tester will be using own development setup to run these set of test cases	Functionality validation can be verified in this stage
Incremental integration testing	As a part of the testing, we will be doing the incremental-integration testing. All the team members will integrate their code on 3 weeks plan. We will be running set up test case to see the sanity of the solution at global level. We can remove / rectify the issues related to inoperability.	Backward compatibility can be maintained
Integration testing	At the end of the solution development, team will run the integration test cases. It might include test case from both black box and white box testing. This will cover end - to -end functionality of DMaaS. As a user we will be able to achieve all the functionality. This test case can be run in the test setup or read cloud setup.	Errors and faults can be identified and fixed between modules communication and dependency
regression testing	At each phase of development team will keep adding test cases which need to pass for any code change. This will helps to maintain the backward compatibility. This will reduce the rework time by adding less errors in the code for new development.	Backward compatibility can be maintained
acceptance testing	Acceptance test cases will have one -to - one mapping with the list of requirements. The test cases development is already started. Some of the test case from the integration testing can also be part of the acceptance test. This will set of feature that will demoed to the customer and required customer signoff for project delivery. This test will help the team be in the customer expectation.	Successful mapping of the requirements and project delivery can be verified

stress testing	This will test the availability and functionality with high usage of the application resource. This will help us to define the performance benchmark. The test includes sending large number of request to web application, simulate the DDOS attack. The web application should able to run and work as expected at large service request.	Stability of the system can verified
performance testing	Performance test is will validate all the performance counters for DMaaS service. The performance measurement includes: 1) number of parallel access to the service 2) Number of datacenter each tenant can handle 3) Number of tenants that system can handle 4) amount of data the analytics engine can process per unit time 4) accuracy of recommendation 5) latency for accessing virtual machines	Service capacity and benchmarking
usability testing	Team will run the usability test cases to test the user friendliness of the web service. This test will validate the user experience with the web service. We working on a tool to measure the time spent by a user per page and how is the sequence page access. This will help the team to design more user friendly pages	Provide user experience details
recovery testing	As a part of recovery testing. The system will be configured in fault tolerance mode. The service will accessed using load balancer. Where we will be testing the system by randomly making components of the system down. The system should sustain any kind of failure in the network. This is one of the important test strategy as we should able to give a continuous service to the customer	Fault tolerant system
security testing	DMaaS service include security features like Access Control List, detection of DDoS attack and detection of hacking by learning the user behavior in the web. All the aspect of the above mentioned will be validated. The service will manage the customer's datacenter and their data. Hence this kind of testing is very helpful to mitigate security threat for the service and customers	Secured service
alpha testing	This test will be done towards the end of service development cycle. We might allow users to use this to learn their reaction and expectation from the product. This test cycle is generally little longer cycle. This is very important as the quality and meeting the requirement of the customer will be measured in this cycle. This gives an over idea on the usage of the product,	Readiness to launch

	quality and performance.	

Table 3- Testing strategy for DMaaS service

Some of the sample test cases that will used for various testing shown as below. The test cases are identified based on four major module like: web service with ACL, virtual abstraction interface, DDoS detection module and recommendation engine.

Case ID	Functionality	Description	Expected Result	Actual Result	Remark
1	**Recommendation System**	Loading historical user data	The data collector should be able to load user's historical data for training the model		
2		Training the data	The model should be able to train the data when inputted with appropriate user data.		
3		Giving Recommendations based on user input	The system should be able to give recommendations based on user's requirements		
4		Giving Recommendations based on usage	Given a user's usage for resources, the system should be able to give recommendations for balancing the load.		
5		Alerting the user	Given that a user has set threshold values for resources, the system should be able to give alerts when the usage reaches threshold.		
6		Updating the user's data for better recommendations	The system should continuously monitor changes for a user and update data accordingly, in order to increase accuracy in recommendations		

7		Testing samples	The system should be able to handle testing samples correctly in order to apply the algorithm on real data		
8		Similarity between users	The system must be able to calculate similarity between users and their preferences in order to give accurate recommendations		
9		Constructing the utility matrix	The system should have the ability to construct a good utility matrix, with appropriate values to provide user with better recommendations		
10		User Ratings for increasing accuracy	There should be a facility to take user input for recommendations given which should be used for further recommendations. This will help increase the accuracy of recommendation system		
11	**DDoS Detection and Mitigation System**	Detecting Attack	The system should have the ability to detect DDoS attack accurately before it affects the resources		
12		Mitigating Attack	In case DMaaS is under DDoS attack, the DDoS detection & mitigation system should have the ability to mitigate this attack		
13		Calculating Entropy accurately	The system should have the ability to calculate entropy correctly and continuously to detect attacks		
14		Searching in blacklist	The system should have the ability to check a user's IP in the blacklist.		
15		Prohibiting access	If a user's IP is found in the blacklist, the system should have the ability to prohibit access to avoid DDoS attack		
16		Dynamic allocation strategy	The system should be able to allocate resources dynamically in order to mitigate the attack		

17		Intrusion Prevention System	The system should have intrusion prevention subsystem to help detect and mitigate attack		
18		Updating blacklist	The system should have ability to update the blacklist in cases where an IP is not listed in this blacklist but was a DDoS attacker.		
19		Monitoring resources	The system should have the ability to monitor resources continuously and detect any abnormal behavior which can be a cause of DDoS attack.		
20		Moving users to new resources efficiently	In case of an attack, the system should have the ability to move the users to dynamically added resources efficiently, without interrupting user processes.		
21	**Virtual Abstraction Interface**	Loading VAI configuration	The virtual abstraction interface should able read all the user configured parameters and load to the system. It might also include to dynamic loading of the update parameters		
22		Single instruction for different hypervisors	VAI shall able to map the DMaaS server's request to appropriate platform specific calls.		
23		Snapshot Manager	System able to create snapshot for each virtual machines and hosts. Snapshot manager should remove the previous snapshot before creating one		
24		Recovery Agent	Recovery agent is able to detect the failure and initiate the recovery procedure		
25		Request validator	Each request is able to validated based on the available resource		

26		Resource manager	Resource manager is able to provide the availability of the resource across datacenters for a given tenant		
27		RabbitMQ	VAI's components are able to use the Rabbit message queue properly		
28		Activity logger	VAI is able to log all the activity messages in the defined log file		
29		Token validation	VAI controller is able to contact the access manager and validate the token before it sent the request to request validator process		
30		Resolving platform specific hyper calls	VAI layer is able to resolve type of virtual environment from the request and able to map the platform specific API calls to manage the virtual environment using openStack and VMWare APIs		
31	**Web service with ACL**	Tenant Manager	Able to load tenant data, validate them using metadata service.		
32			Tenant Manager should update or add or delete tenant details to metadata using metadata Service.		
33		Meta service	able to delete, add and update through meta service dB proxy		
34		DMaaS authenticator	DMaaS authenticator should interface to tenant manager and credential manager		
35		Credential Manager	Should able to create/update/delete credential for tenant which are new or existing user		
36		Resource Manager	It should able to authorize the resource logging user and can able to restrict the user to access more resources. It also should able to provide the access details to the user.		

37		Data Manager	Should able to validate and store data files against created template present in the system		
38		VAI Controller	Should able to create/validate the system created decision tree per user which is based on the usage and access of DMaaS system		

Table 4- Sample test case for DMaaS service

Deployment

DMaaS is a cloud based service, will be deployed in Amazon Web Service (AWS). The major components required for the deployment will as below:

Elastic Cloud Compute (EC2):

Solution will be using multiple EC2 instances. This includes both micro and large instance. All the micro instance will be used for the web service hosting, virtualization layer and security analysis. We will be using large instance for the data analysis and recommendation system.

Elastic Load Balancer (ELB):

DMaaS will required ELB for load balancing purpose. ELB will be configured to spin new instances if the traffic rate increases from the configured value. It will provide easy scaling with respect to traffic and resource load.

MySQL database:

All the transaction and tenant related information will be stored in the MySQL. We will use efficient way to implement multi-tenant functionality.

NoSQL database:

Information required for analysis and recommendation purpose will be stored in the NoSQL database. All the VM statistics will also be stored in the NoSQL database like MongoDB. We are

Firewall:

To add security towards the enterprise data pipeline, firewall can be used. It will provide additional security for accessing the data service from the enterprise towards DMaaS service.

Figure 18 depicts a typical deployment scenario. Two companies- Company A and Company B are using DMaaS platform. The companies are able to create, monitor and get recommendations for VMs across different hypervisors like VMWare, OpenStack, and Xenserver using proposed solution.

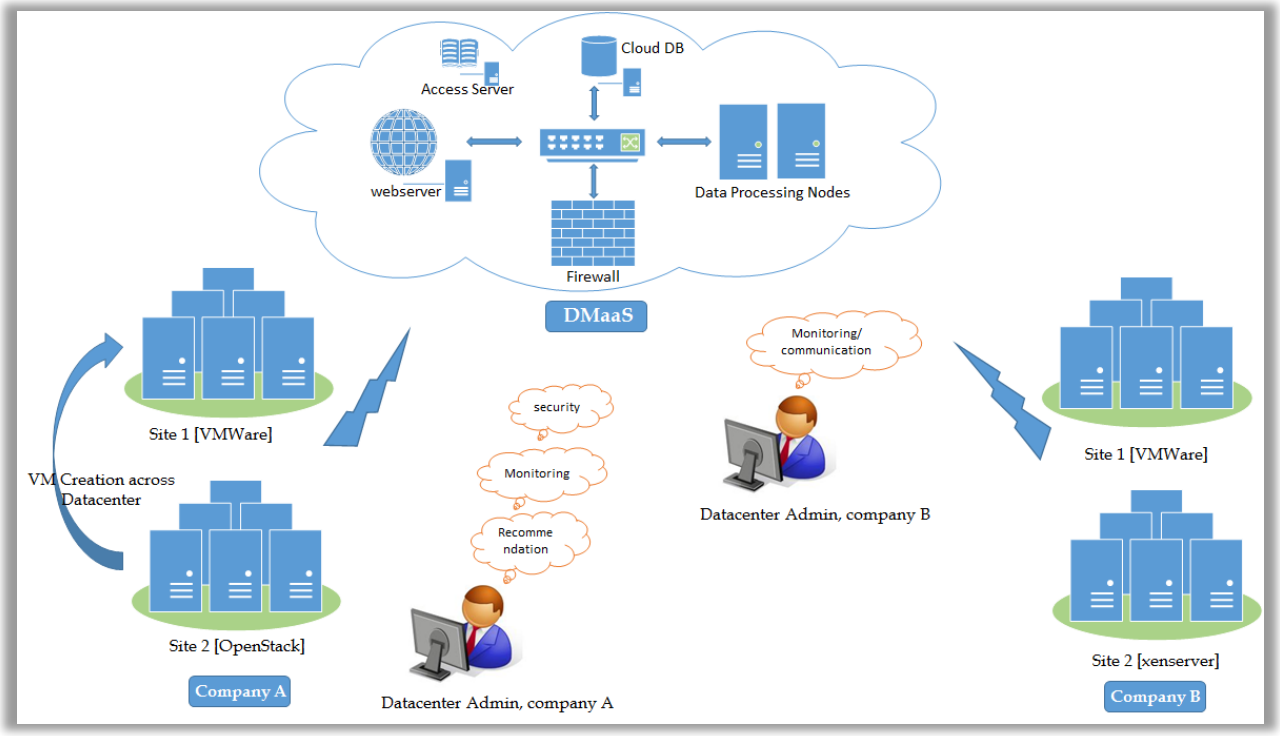


Figure 18: Deployment Diagram

The datacenter admins of both these companies have ease in maintaining security, monitoring liveliness and managing load using recommendations provided.

Algorithm:

One of the example algorithm from our reference [19] is as shown below:

The algorithm can be implemented as follows.

Gradient-based framework for matrix factorization

1: Input: $\boldsymbol{X}$-matrix of microarrays.
2: Select m-number of global iterations; q-number of factors; $\lambda > 0$-initial learning rate, $0 < \xi < 1$-correction rate, L_S-initial value of the target function.
3: Initial matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ may be generated randomly.
4: Global cycle: repeat m times the following steps 5–17:
5: genes-cycle: for $i = 1$ to p repeat steps 6–15:
6: tissues-cycle: for $j = 1$ to n repeat steps 7–15:
7: compute prediction $S = \sum_{f=1}^{q} a_{if} b_{fj}$;
8: compute error of prediction: $E = x_{ij} - S$;
9: internal factors-cycle: for $f = 1$ to q repeat steps 10–15:
10: compute $\alpha = a_{if} b_{fj}$;
11: update $a_{if} \Leftarrow a_{if} + \lambda \psi(E) b_{fj}$;
12: $E \Leftarrow E + \alpha - a_{if} b_{fj}$;
13: compute $\alpha = a_{if} b_{fj}$;
14: update $b_{fj} \Leftarrow b_{fj} + \lambda \psi(E) a_{if}$;
15: $E \Leftarrow E + \alpha - a_{if} b_{fj}$;
16: compute $L = L(\boldsymbol{A}, \boldsymbol{B})$;
17: $L_S = L$ if $L < L_S$; otherwise: $\lambda \Leftarrow \lambda \cdot \xi$.
18: Output: $\boldsymbol{A}$ and $\boldsymbol{B}$-matrices of loadings and metagenes.

The following partial derivatives are necessary for the above algorithm (see steps 11 and 14):

$$\frac{\partial \Psi(E_{ij})}{\partial a_{if}} = -\psi(E_{ij}) b_{fj}, \tag{8}$$

$$\frac{\partial \Psi(E_{ij})}{\partial b_{fj}} = -\psi(E_{ij}) a_{if}. \tag{9}$$

Chapter 8. Implementation Plan & Progress

This section describes the project implementation and progress of the proposed solution for Datacenter Management as a Service (DMaaS). This solution will provide a unified approach to manage data center with disparate virtualization platforms by providing control plane abstraction to the data center. Modern approach of recommendation system will be used for the resource management. The algorithm will track the usage of the system and behavior.

Summary of development tools:

- Development toolkit: Eclipse and C9 (Cloud based collaborative development tool)
- Programing language: Java, Node.js, Ruby, Python
- Cloud platform: C9 and Amazon web services
- Graph visualization: Google Charts and high charts
- Project management tools: Microsoft excel, Kanbanize and Lucid charts

High level implementation plan is as follows:

1. Environment setup:

The environment setup includes local development and test setup and cloud-based setup for final deployment.

   a. The local development environment includes IDE for development, installation of vSphere to create the local simulated data center (Software installed already).
   b. The deployment setup will be using Amazon Web Service environment. (Requirement finalized).
   c. For data processing, we will be installing single node HDFS partition for proof-of-concept development.

2. Acquiring development tools:

Initial level of development tools already installed, which includes:

   a. Eclipse and C9 cloud platform, as IDE.
   b. Java 8.0 for web services development.
   c. Python 2.7 for test case development.
   d. Node.js, Ruby and Boot strap with express framework from front-end development.
   e. MongoDB, to store activity log, MySQL for transaction based data and logs. HDFS to analyze datacenter data.
   f. RESTful APIs, data access framework.

3. Simulation environment:

Simulation environment will be created using Amazon Web Services instances, includes HDFS instances for data analysis, two instances for web service hosting in active-active mode. The instances will also provide high-availability.

4. Understanding algorithms, sample programs and executing them on dataset being used:
   a. Each team member is exploring relevant algorithms for the said objective.
   b. Team members are creating individual development plan.
   c. Sample code will be ready for the 295A final report submission.
5. Proof of concept will include:
   a. Basic user portal development to allow multi-tenant login.

b. Accessing at least one virtual machine data and configure from the front-end.
c. Upload data for cleansing and analysis.
d. Environment setup for the recommendation engine.

6. Implementing the Essential feature set:

Essential features, mentioned in section 3 will be studied thoroughly and high level design will be ready for 295A final report.

a. Identification of essential features is completed.
b. We will work on connectivity between web service and data center.
c. Loading data to HDFS file system for analysis.
d. Detailed design of business logic will be prepared.

7. Implementation of desired and optional features:

Desired features, mentioned in section 3 will be analyzed and high level design will be ready for review.

a. Desired and optional features are identified
b. Detailed design for the user interface will be created.
c. Logging module design will be created.
d. Development plan will be created for all the optional features.

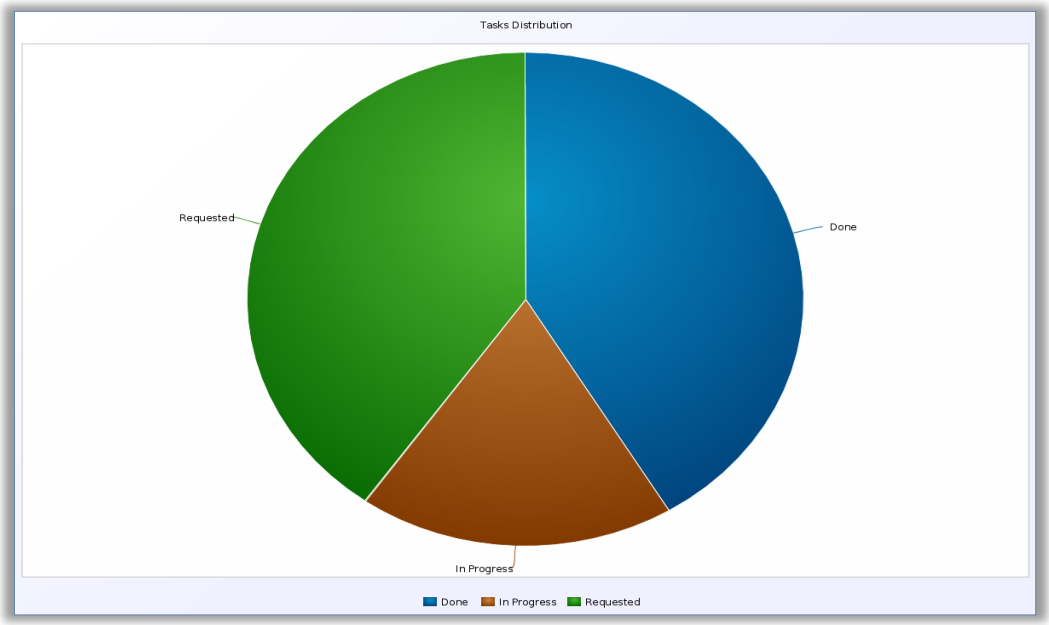


Figure 19 Task distribution diagram using task status

Currently there are 87 tasks for all the modules for DMaaS, out of which 41.38% are in done state, 18.39% are in progress and 40.23% are in requested state.

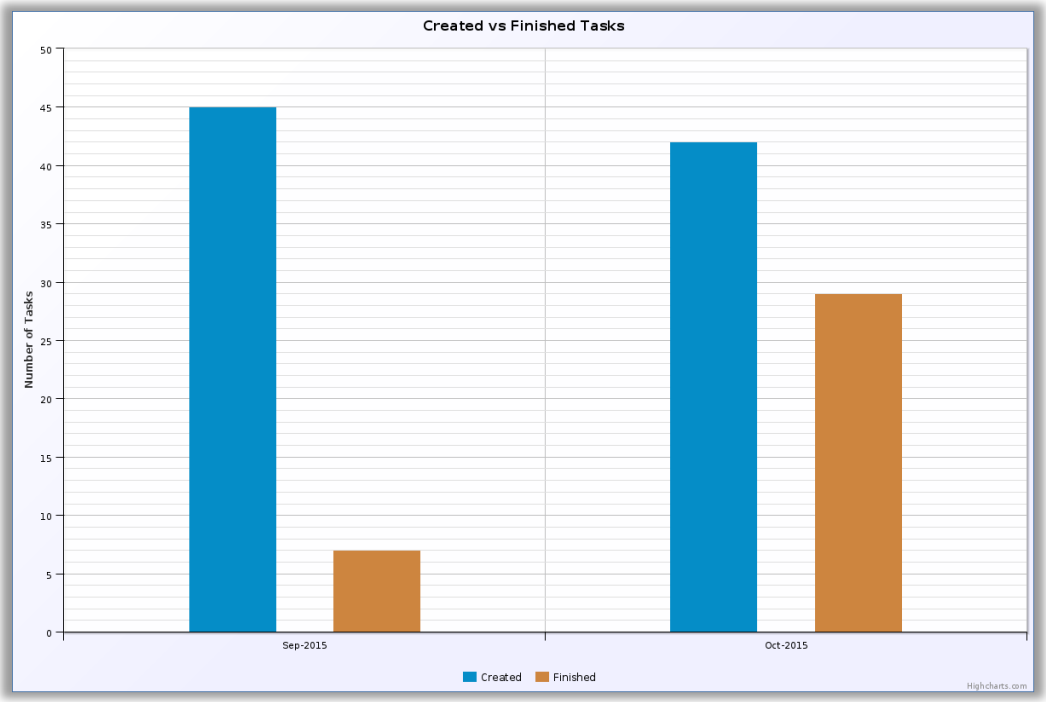


Figure 20 : Created vs finished diagram for the month of September and October.

The Created vs. Finished chart shown in figure 20 for the month of September, show that 45 tasks are created and 7 tasks finished. But in October we could able to contribute more as a team and created 42 tasks, out of which we finished 29 tasks.

Chapter 9. Project Schedule

Project schedule has been planned using the Gantt chart, PERT chart and Kanban board. Project milestone and release plan will be tracked using Gantt and PERT chart. Individual work assignment and status of the task will be managed by online Kanbanize tool, which is Kanban board for project management.

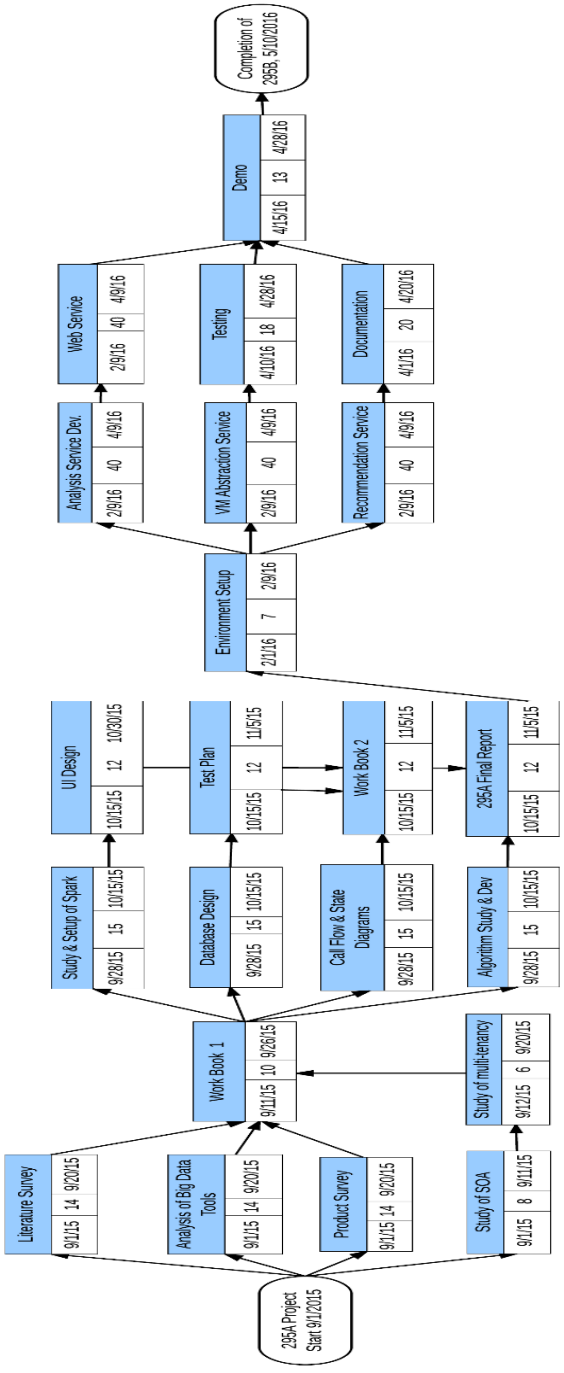
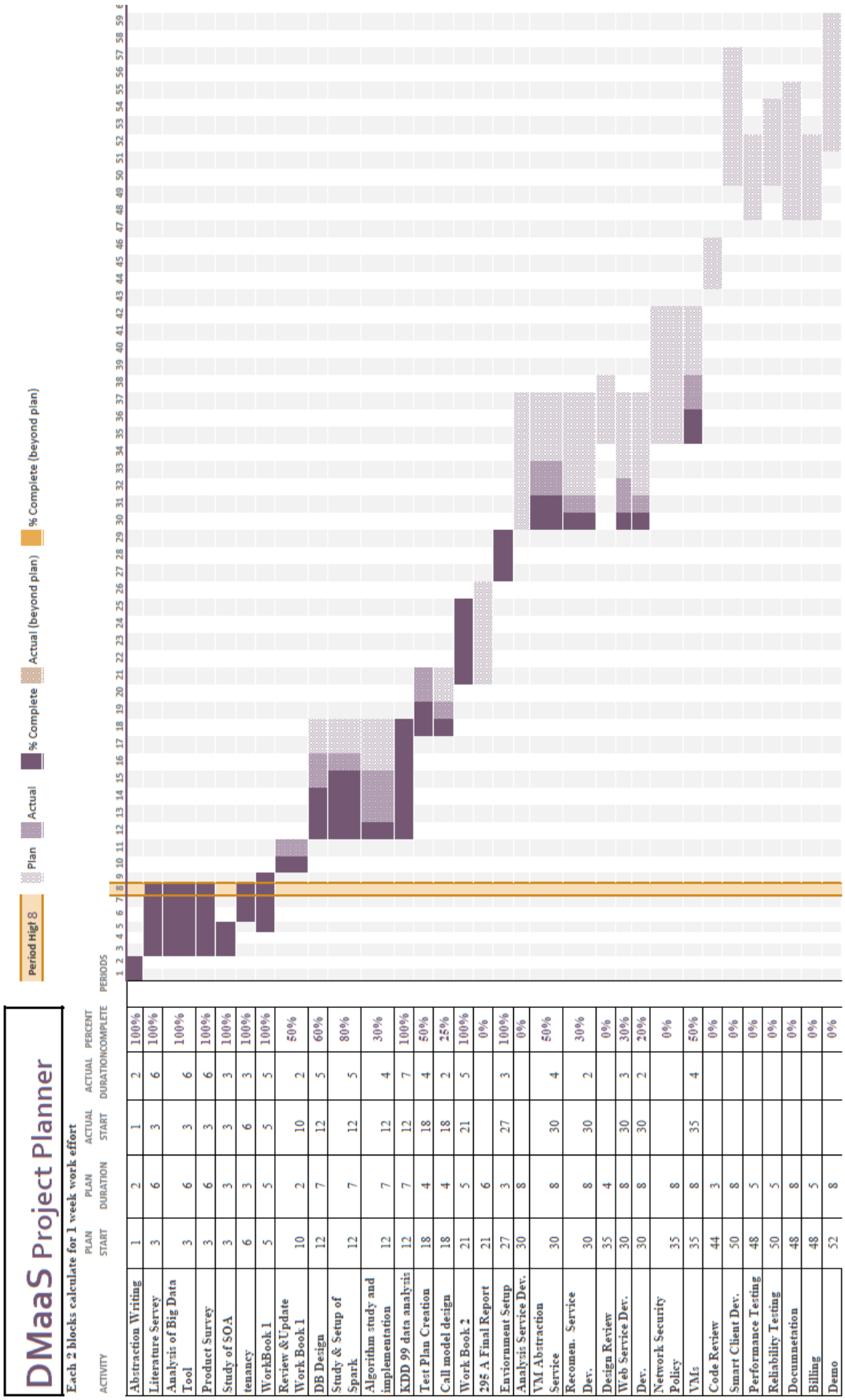


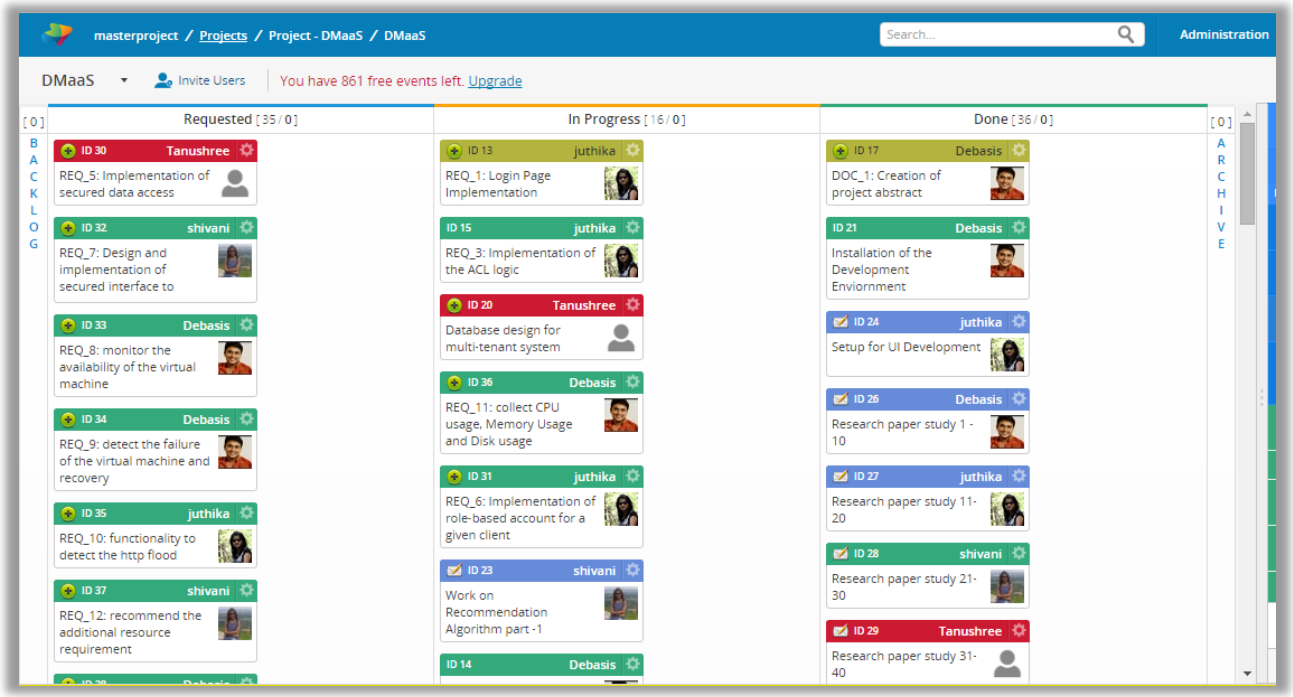
Figure 21: Project PERT chart



ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Abstraction Writing	1	2	1	2	100%
Literature Servey	3	6	3	6	100%
Analysis of Big Data Tool	3	6	3	6	100%
Product Survey	3	6	3	6	100%
Study of SOA	3	3	3	3	100%
tenancy	6	3	6	3	100%
WorkBook 1	5	5	5	5	100%
Review &Update Work Book 1	10	2	10	2	50%
DB Design	12	7	12	5	60%
Study & Setup of Spark	12	7	12	5	80%
Algorithm study and implementation	12	7	12	4	30%
KDD 99 data analysis	12	7	12	7	100%
Test Plan Creation	18	4	18	4	50%
Call model design	18	4	18	2	25%
Work Book 2	21	5	21	5	100%
295 A Final Report	21	6			0%
Enviornment Setup	27	3	27	3	100%
Analysis Service Dev.	30	8			0%
VM Abstraction Service	30	8	30	4	50%
Recomen. Service Dev.	30	8	30	2	30%
Design Review	35	4			0%
Web Service Dev.	30	8	30	3	30%
Dev.	30	8	30	2	20%
Network Security Policy	35	8			0%
VMs	35	8	35	4	50%
Code Review	44	3			0%
Smart Client Dev.	50	8			0%
Performance Testing	48	5			0%
Reliability Testing	50	5			0%
Documnetation	48	8			0%
Billing	48	5			0%
Demo	52	8			0%

Figure 22: Project Gantt chart

Updated Gantt chart for DMaaS project shown above, described the activity, its planned duration and actual duration and planned start time and actual start time. This char also explains DMaaS activities percentage of completion.



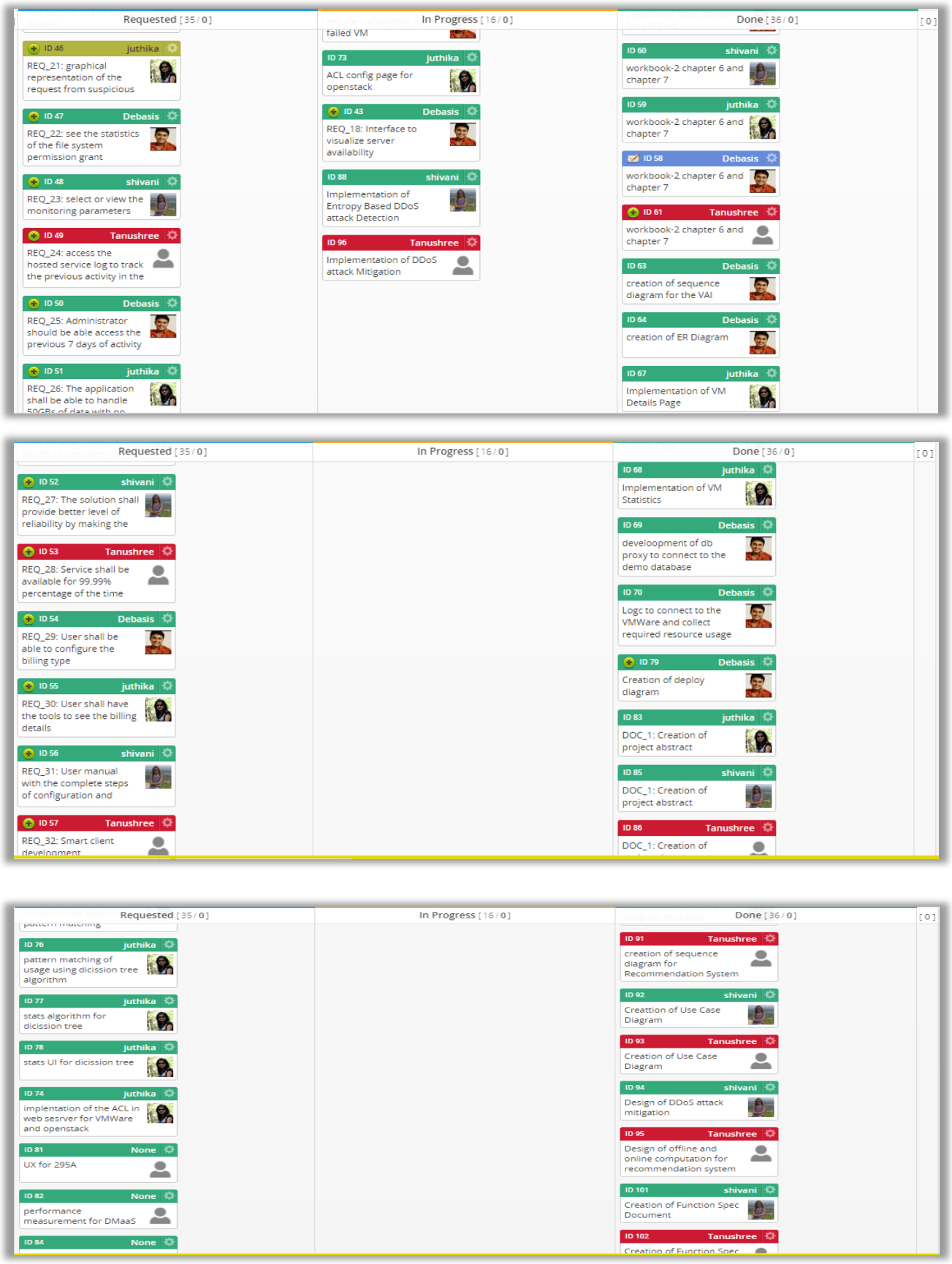


Figure 23: Project task's progress using Kanbanize

Figure 23 shows all the tasks created for DMaaS in "Kanbanize" web tool. There three swim line as "Requested" (means created but not yet started), "In Progress" (means work in progress for the task) and done which implies the completed tasks.

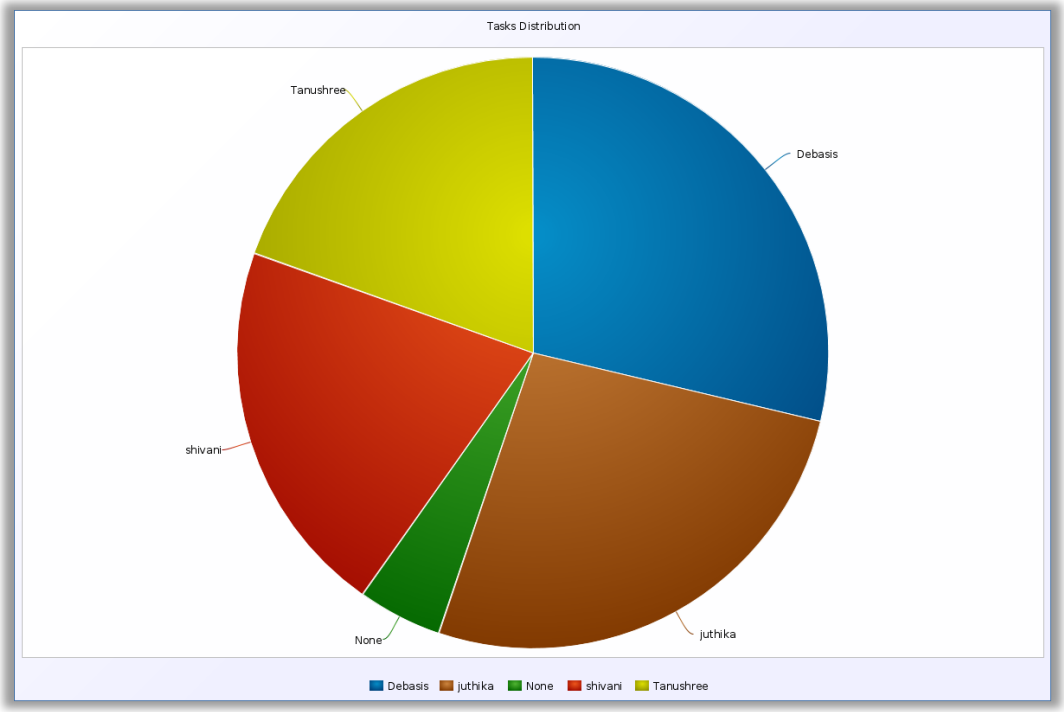


Figure 24: Task distribution for each assignee.

Currently we have 87 tasks and can be increased based on the task criticality. The graph in figure 24 shows the tasks distribution for each assignee.

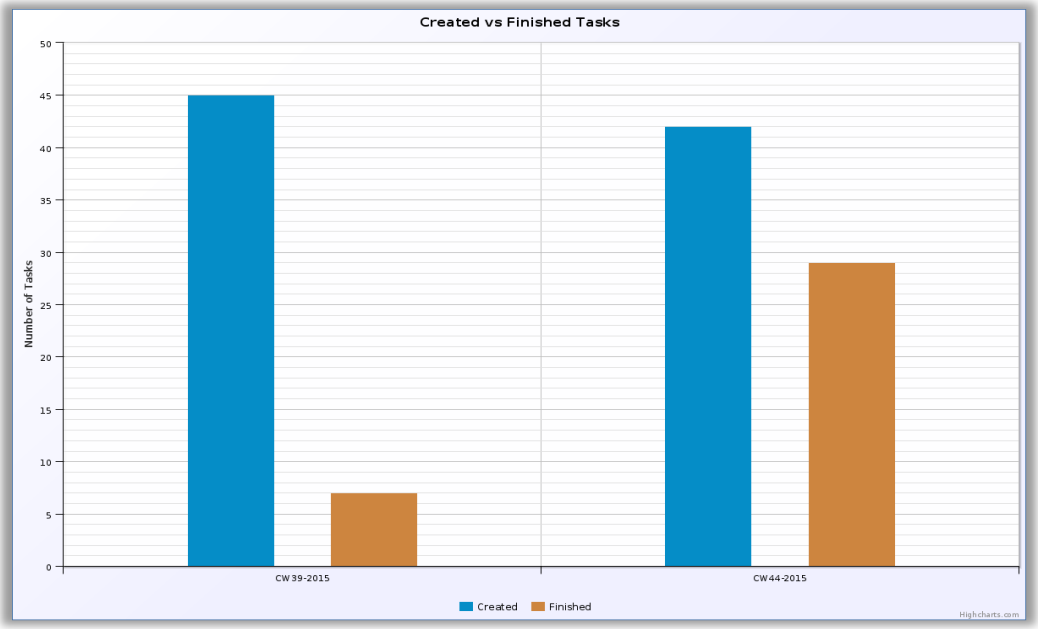


Figure 25: Created vs finished tasks for 39 calendar week and 44 calendar week

Figure 25, shows the created vs finished analytics for the 39 and 44 calendar week.

Chapter 10. Conclusion

DMaaS is a future ready software as a service, which will be solving many issues related to datacenter consolidation. The multi-tenant nature of software will reduce the per user cost. Fixed cost for setting up the service will shared across multiple users. Enterprise can reduce their operational cost by, pay as you go model. This will be very helpful to small and medium class business in addition to the large enterprises. Enterprise can now focus on main goal of their business rather than spending time and money to manage their datacenters.