**VIETNAM NATIONAL UNIVERSITY HOCHIMINH CITY**

**UNIVERSITY OF INFORMATION TECHNOLOGY**

**ADVANCED PROGRAM IN INFORMATION SYSTEMS**

--------------------------------



**BÙI NGUYỄN HOÀNG TÂN – VÕ TRƯỜNG VŨ**

**RESEARCH AND IMPLEMENT CUSTOMER RELATIONSHIP MANAGEMENT ON CLOUD-BASED COMPUTING ARCHITECTURE**

**BACHELOR OF ENGINEERING IN INFORMATION SYSTEMS**

**HO CHI MINH CITY, 2015**

**VIETNAM NATIONAL UNIVERSITY HOCHIMINH CITY**

**UNIVERSITY OF INFORMATION TECHNOLOGY**

**ADVANCED PROGRAM IN INFORMATION SYSTEMS**

--------------------------------



**BÙI NGUYỄN HOÀNG TÂN - 10520019**

**VÕ TRƯỜNG VŨ – 10520651**

**RESEARCH AND IMPLEMENT CUSTOMER RELATIONSHIP MANAGEMENT ON CLOUD-BASED COMPUTING ARCHITECTURE**

**BACHELOR OF ENGINEERING IN INFORMATION SYSTEMS**

**THESIS ADVISOR**

**Dr. HỒ BẢO QUỐC**

**HO CHI MINH CITY, 2015**

# **ACKNOWLEDGMENTS**

The author wishes to express sincere appreciation to Dr. Hồ Bảo Quốc for his assistance in the preparation of this dissertation. In addition, special thanks to Dr. Ngô Thanh Hùng, whose facilitate with the consultancy in design and implementation the system. Thanks also to the members of the school council for their valuable input.

Finally, we would like to express our greatest gratitude towards our families for always believe and support us.

# **EVALUATION OF ADVISOR**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

# **EVALUATION OF CRITIC**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**TABLE OF CONTENTS**

**🙢🕮🙠**

[**ACKNOWLEDGMENTS** i](#_Toc411418665)

[**EVALUATION OF ADVISOR** ii](#_Toc411418666)

[**EVALUATION OF CRITIC** iii](#_Toc411418667)

[**LIST OF FIGURES** vii](#_Toc411418668)

[**LIST OF TABLES** x](#_Toc411418669)

[**ABSTRACT** xi](#_Toc411418670)

[**Chapter 1 : Introduction** 1](#_Toc411418671)

[**1.1** **Background** 1](#_Toc411418672)

[**1.2** **Problem statement** 1](#_Toc411418673)

[**1.3** **Scope of this thesis** 2](#_Toc411418674)

[**1.4** **Structure of this thesis** 2](#_Toc411418675)

[**Chapter 2 : Fundamental Theories** 3](#_Toc411418676)

[**2.1** **Introduction** 3](#_Toc411418677)

[**2.2** **Cloud Computing** 3](#_Toc411418678)

[**2.2.1** **The era of Cloud Computing** 3](#_Toc411418679)

[**2.2.2** **Definition of Cloud Computing:** 3](#_Toc411418680)

[**2.2.3** **Advantages of Cloud Computing** 4](#_Toc411418681)

[**2.2.4** **Essential Characteristics** 4](#_Toc411418682)

[**2.2.5** **Cloud Computing Service Model** 5](#_Toc411418683)

[**2.2.6** **Deployment Model** 6](#_Toc411418684)

[**2.3** **Customer Relationship Management (CRM)** 9](#_Toc411418685)

[**2.3.1** **Definition of CRM** 9](#_Toc411418686)

[**2.3.2** **Some benefits of CRM** 9](#_Toc411418687)

[**2.3.3** **Why is CRM?** 10](#_Toc411418688)

[**Chapter 3 : Cloud Computing Solutions** 11](#_Toc411418689)

[**3.1** **Introduction** 11](#_Toc411418690)

[**3.2** **The Overview of some CC solution** 11](#_Toc411418691)

[**3.3** **OpenStack** 12](#_Toc411418692)

[**3.3.1** **Definition of OpenStack** 12](#_Toc411418693)

[**3.3.2** **History Founder of OpenStack** 12](#_Toc411418694)

[**3.4** **Why we choose OpenStack** 13](#_Toc411418695)

[**3.5** **Main component of OpenStack-IceHouse** 15](#_Toc411418696)

[**3.6** **Open Source CRM solution** 17](#_Toc411418697)

[**Chapter 4 : System Implementation** 19](#_Toc411418698)

[**4.1** **Introduction** 19](#_Toc411418699)

[**4.2** **OpenStack installation and system design** 19](#_Toc411418700)

[**4.2.1** **Design system** 19](#_Toc411418701)

[**4.2.2** **The deployment model** 20](#_Toc411418702)

[**4.2.3** **Install the following service on each appropriate node – install OpenStack IceHouse** 22](#_Toc411418703)

[**4.3** **Deploy some module of CRM** 32](#_Toc411418704)

[**4.3.1** **Install sugarCRM on virtual machine** 32](#_Toc411418705)

[**4.3.2** **Deploy module account** 36](#_Toc411418706)

[**4.3.3** **Deploy module Leads - Potential customers** 40](#_Toc411418707)

[**4.3.4** **Deploy module Opportunity** 44](#_Toc411418708)

[**4.3.5** **Deploy Email Marketing** 47](#_Toc411418709)

[**4.4** **Summary** 49](#_Toc411418710)

[**Chapter 5 : System Testing and Result** 50](#_Toc411418711)

[**5.1** **The content of the test** 50](#_Toc411418712)

[**5.2** **Data preparation** 50](#_Toc411418713)

[**5.3** **The result of the test performance** 50](#_Toc411418714)

[**5.4** **Analysis** 53](#_Toc411418715)

[**Chapter 6 : Conclusion and Future Work** 54](#_Toc411418716)

[**6.1** **Conclusion** 54](#_Toc411418717)

[**6.2** **Future Work** 54](#_Toc411418718)

[**REFERENCES** 55](#_Toc411418719)

[**APPENDICES** 56](#_Toc411418720)

# **LIST OF FIGURES**

**🙢🕮🙠**

[Figure 2.1 Service models of cloud computing 5](#_Toc411418721)

[Figure 2.2 Public Cloud. 7](#_Toc411418722)

[Figure 2.3 Private cloud. 7](#_Toc411418723)

[Figure 2.4 Hybrid cloud. 8](#_Toc411418724)

[Figure 2.5 Community cloud. 8](#_Toc411418725)

[Figure 3.1 The position of OpenStack. 12](#_Toc411418726)

[Figure 3.2 Monthly number of participant [7] 13](#_Toc411418727)

[Figure 3.3 Accumulated community populations [7] 14](#_Toc411418728)

[Figure 3.4: Accumulated developer population [7] 15](#_Toc411418729)

[Figure 3.5 Top 10 CRM solution 17](#_Toc411418730)

[Figure 4.1 The model to install OpenStack IceHouse. 21](#_Toc411418731)

[Figure 4.2 Services in each nodes. 22](#_Toc411418732)

[Figure 4.3 Verify NTP synchronization. 23](#_Toc411418733)

[Figure 4.4 The database name of the services. 23](#_Toc411418734)

[Figure 4.5 The users of the system. 24](#_Toc411418735)

[Figure 4.6 The image of the OpenStack. 24](#_Toc411418736)

[Figure 4.7 Hypervisor of compute node. 25](#_Toc411418737)

[Figure 4.8: Network agents. 25](#_Toc411418738)

[Figure 4.9 Compute services. 26](#_Toc411418739)

[Figure 4.10 The login of horizon. 26](#_Toc411418740)

[Figure 4.11 The overview of horizon. 27](#_Toc411418741)

[Figure 4.12 Network topology. 28](#_Toc411418742)

[Figure 4.13 Create a new volume. 29](#_Toc411418743)

[Figure 4.14 Swift store an object. 29](#_Toc411418744)

[Figure 4.15 Resource usage overview. 29](#_Toc411418745)

[Figure 4.16 Create a template. 30](#_Toc411418746)

[Figure 4.17 The instance created by template above. 30](#_Toc411418747)

[Figure 4.18 Attached the instance to the volume. 30](#_Toc411418748)

[Figure 4.19 The swift interface stored an image. 31](#_Toc411418749)

[Figure 4.20 Resources usage overview. 31](#_Toc411418750)

[Figure 4.21 The configuration of the instance*.* 32](#_Toc411418751)

[Figure 4.22 The Ubuntu instance in Horizon. 32](#_Toc411418752)

[Figure 4.23 Using PuTTY to add new user. 32](#_Toc411418753)

[Figure 4.24 Login to the instance. 33](#_Toc411418754)

[Figure 4.25 Setup sugarCRM through web browser. 35](#_Toc411418755)

[Figure 4.26 The login interface of sugarCRM. 35](#_Toc411418756)

[Figure 4.27 The main interface of sugarCRM. 36](#_Toc411418757)

[Figure 4.28 Module account overview. 36](#_Toc411418758)

[Figure 4.29 Add a new customer. 37](#_Toc411418759)

[Figure 4.30 Finding a customer named James. 38](#_Toc411418760)

[Figure 4.31 Edit a customer. 38](#_Toc411418761)

[Figure 4.32 Two companies we want to merge. 38](#_Toc411418762)

[Figure 4.33 Merge two companies. 39](#_Toc411418763)

[Figure 4.34 The information after merging. 40](#_Toc411418764)

[Figure 4.35 Create an activity. 40](#_Toc411418765)

[Figure 4.36 Module Leads. 41](#_Toc411418766)

[Figure 4.37 Create a potential customer - Mr. Mitsue Toller. 42](#_Toc411418767)

[Figure 4.38 Mass update 42](#_Toc411418768)

[Figure 4.39 Convert leads. 43](#_Toc411418769)

[Figure 4.40 Create log call. 44](#_Toc411418770)

[Figure 4.41 Prompt dialog box. 44](#_Toc411418771)

[Figure 4.42 Module opportunity overview. 45](#_Toc411418772)

[Figure 4.43 Create an opportunity. 45](#_Toc411418773)

[Figure 4.44 Create an activity for the opportunity. 46](#_Toc411418774)

[Figure 4.45 Task: Make a report. 46](#_Toc411418775)

[Figure 4.46 Task finished. 46](#_Toc411418776)

[Figure 4.47 The content of the task after finished. 47](#_Toc411418777)

[Figure 4.48 Campaign Travel 01/2015. 48](#_Toc411418778)

[Figure 4.49 The email content. 48](#_Toc411418779)

[Figure 4.50 Email queue. 49](#_Toc411418780)

[Figure 4.51 Email send success 49](#_Toc411418781)

[Figure 5.1 Real machine IO time. 51](#_Toc411418782)

[Figure 5.2 Virtual machine IO time. 52](#_Toc411418783)

# **LIST OF TABLES**

**🙢🕮🙠**

[Table 4.1 The host to install OpenStack 20](#_Toc411418784)

[Table 5.1 Configuration of two machines 50](#_Toc411418785)

[Table 5.2 The CPU benchmark 51](#_Toc411418786)

[Table 5.3: Real and virtual machine IO time. 52](#_Toc411418787)

[Table 5.4: Time install sugarCRM. 53](#_Toc411418788)

# **ABSTRACT**

Cloud Computing (CC) is becoming a popular term in IT industry and it’s making a massive change to computation platform. It makes deploying and maintaining an application easier and faster than ever. Although CC is still in its early state, most enterprises are begin to adopt it. CC helps reduce a huge cost for deploy and maintain applications and its infrastructure. Thus, CC brings a chance for Small and Medium Enterprises to apply and benefit from modern and latest technologies without so much effort and increase their competent.

On this thesis, we research and learn about Cloud Computing technology and how to create clouds (private or public) by using open source cloud operating system – OpenStack. Then we select an open source application – SugarCRM - a Customer Relationship Management application, one of the most needed application for any enterprise and deploy it on our cloud platform. Finally, we testing the system on some sample data for the usability and performance.

Our motivation is to help Vietnamese Enterprises change the way they deploy applications. Setting up and environment, physical server and deploying application on it require a high cost and an amount of time and not effective in resource management. However, with our idea about applying Cloud Computing, Enterprises now can deploy an application in a short time, lower cost and can be proactive in management the resource. Furthermore, this solution opens an approach is to provide an software solution for Vietnamese Small-And-Medium Enterprise (SME) without have to invest for their own IT infrastructure, IT department with are too costly and might be impossible. From this, Vietnamese SME will have a better chance to benefit from software solutions like CRM and increase their business effectiveness.

When bringing the system to test phrase, the system perform well on sample data and the experience is not so different from between virtual cloud system with real physical system. The result proves the high possibility of our solution to apply in real system of companies. However, it still in an early state with raw model of service we can provide. In the future, we will study more about develop more suitable and convenient solutions for this system and make it available to be access anywhere with the internet (our system in this time can only be access with the same intranet).

# **: Introduction**

1. **Background**

Nowadays, for enterprises, how to manage company, their customer information and the software that support the company’s process is one of the difficult and high priority problem almost every enterprise have to face. In order to solve this, in the ordinary way, a company have to pay a high costs for buying hardware, licensed software, network infrastructure, hiring salary for IT employees to manage, maintain the entire complex system. Moreover, they have to investigate and calculate all the problems such as the extensibility, upgradable, security and availability of the system. Now with the raising of technology, Cloud Computing becomes the answer the problem. Cloud Computing is not a just concept anymore, people are using cloud-based services every day without notice that they are on the cloud from a wide range of applications such as web mail, cloud storage (dropbox, google drive), social network. With cloud computing, businesses can solve the infrastructure and manage resources problem, and to be able to focus more on the business matter, boost up the profit.

In this Thesis, we focus on how to apply Cloud Computing techniques to help people deploy applications the faster ways and manage their resources effectively.

1. **Problem statement**

Information technology is becoming an important part of any business with a wide range of enterprise applications such as ERP, mail service, Data Mining etc. It helps enterprises reduce the cost and make the business process faster, easier. However, the cost enterprises have to pay when apply IT into their business is still high and complex with all the things like: infrastructure – servers, data center, IT people for implement and maintain the system Therefore it becomes a big obstacle for any Vietnamese enterprises especially Medium and Small one when consider implement IT in their business. In a decade, the term “Cloud Computing” is rising as one of the biggest technology trend that evolve the way deploy and using IT application. In Vietnam, “Cloud Computing” is still a new concept. But if enterprises understand it clearly, this technology could be a potential answer that help solving the challenges when apply information technology into their business.

1. **Scope of this thesis**

In this thesis, we research and study about Cloud Computing architect, cloud computing techniques and how to apply one these and implement a CRM application on cloud machine and evaluate the possibility to apply in practice.

1. **Structure of this thesis**

To accomplish this scope, a suitable and ordinal structure should help.

First, the thesis must have basic knowledge about Cloud Computing, Virtualization and their characteristics. What need to be concern when apply Cloud Computing and to choose a form of Cloud Computing services. That is the work of chapter 2 – bringing some fundamental theories to the reader.

Second, from the fundamental theories, we research on different ways to apply Cloud Computing, reasons to choose one and research the way this solution working.

Third is when the real implementation comes to play. The knowledge and the plan are all introduced in previous section, but how to bring it into real life is a matter. This session will have reader understand components of the solution and how to deploy it on a real computer system, after that we try to deploy a real enterprise CRM software on the implemented cloud. This phase is presented in chapter.

The testing on sample data and the performance of the system, availability and characteristic of Cloud should be mention when the implementation is done, some strategies that business can apply the solution and future development can be made.

# **: Fundamental Theories**

1. **Introduction**

In this chapter, we will go through the fundamental concept of cloud computing and learn how can it help to utilize the current system and type of services it provides. We also research about one types of enterprise application which is rising as an important part of any business: Customer relationship management

1. **Cloud Computing**
2. **The era of Cloud Computing**

Unlike the early years of the era of information technology, for now, the total investment cost for the annual information technology, management component for increasing investment in comparison with the device. Indeed, an urgent demand for the enterprise is to reduce the complexity and cost due to the reverse side of the development of information technology gives enterprises. Typically, the growth of unwanted acreage for equipment, energy consumed, number of administrative staff, license fees, and the cost of maintaining the system.

In another aspect, the mobile device is growing strongly. The trend of the users is used tablets, smartphones at work replacing traditional PCs. Although dominant in terms of flexibility, the majority of mobile devices are equipped with hardware does not meet the deployment of all applications directly onto it. Therefore, there should be a system on which the application is installed, the server handles most and give the terminal (mobile, desktop) as a "service", and these devices play a role input and output.

To respond to the demand above, cloud computing technology based on virtualization was appeared. With this technology, technical, network-based infrastructure development, can respond well the "service". In addition, the costs of managing and maintaining information technology systems are substantially reduced.

1. **Definition of Cloud Computing:**

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models. [1]

1. **Advantages of Cloud Computing**

* Fast processing speed, providing the user with fast service and cheap price based on centralized infrastructure (cloud).
* The cost of the initial investment in infrastructure, machinery and human resources of the use of cloud computing is reduced to a minimum.
* No longer dependent on the equipment and geographical location, allowing users to access and use the system via a web browser anywhere and on any device they use (such as a PC or mobile phone ...).
* Sharing resources and costs on an extensive area, bringing benefits to the user.
* Scalability is to help improve the quality of services offered on the "cloud".
* The security is improved by focusing on the data.
* The application of cloud computing easier to repair and improve the features because they are not fixed installation on a computer.

1. **Essential Characteristics**

There are five essential characteristics:

* *Rapid elasticity*: The providers easily specify resource recovery as well as the user very quickly. To the user is allowed to request a resource "unlimited" and only paying what they used.
* *Broad network access*: access to computing resources easily through standard mechanisms for network.
* *Measured service*: provider to ensure the calculation of customer consumption. Model towards a "pay as you go".
* *On-demand self-service*: allowing customers to customize the resources used without any notification or through the intervention of the provider.
* *Resource pooling*: the physical resources and virtual resources of the CC is shared and automatically granted to users.

1. **Cloud Computing Service Model**

There are three service models of cloud computing:

• Infrastructure as a Service (IaaS).

• Platform as a Service (PaaS).

• Software as a Service (SaaS).

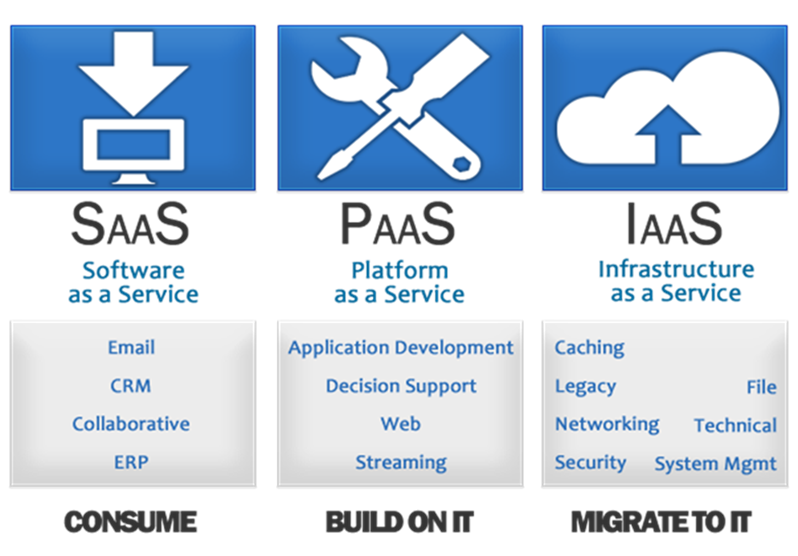


Figure 2.1 Service models of cloud computing

* *Software as a Service (SaaS)*: SaaS provides complete applications as an on-demand service for clients with just one version installed. Customers choose the appropriate application needs and use that does not care about or put effort managing computing resources below.
* *Platform as a Service (PaaS)*: PaaS services provide computing platform that allows customers to develop software, service computing needs or build a Cloud-based services there. PaaS services can be built separately and provide customer through a private API. Customers build applications and interact with CC infrastructure through this API. At PaaS, customers do not manage or Cloud platform resources such as the operating system layer, kept in the lower layer.
* *Infrastructure as a Service (IaaS)*: Services IaaS provides basic services including computing power, storage space, and network connectivity to customers. Customers (individuals or organizations) can use this infrastructure resources to meet the computing needs or install applications for users. With this service, customers do all operating systems, storage and applications installed by the customer. Typically, customers of IaaS services can be anyone who want to have a computer and manually install applications.

1. **Deployment Model**

There are four deployment models:

• Public cloud.

• Private cloud.

• Hybrid cloud.

• Community cloud.

* *Public Cloud*: This is the model that infrastructure CC is an organization owned and offers extensive services to all customers through the Internet infrastructure or public wide area networks. The Public Cloud services towards large number of customers often have high capacity infrastructure to meet the computational demands flexibility, offering customers low cost.

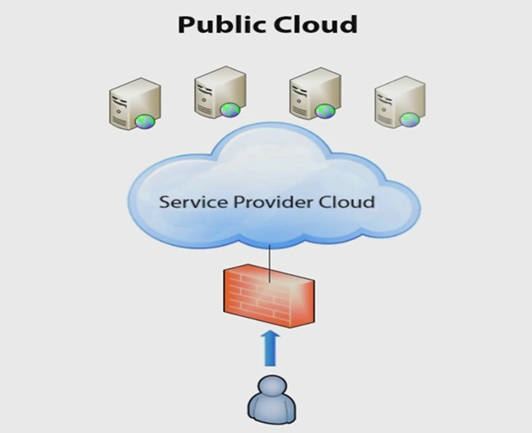


Figure 2.2 Public Cloud.

* *Private Cloud*: Private Cloud organizations, large construction firms to exploit their advantages are organizations, large construction firms to exploit their advantages in technology and management capabilities of CC. With Private Cloud, businesses optimize their IT infrastructure, improve efficiency, and manage the allocation and recovery of resources, thereby reducing the time-to-production business market.

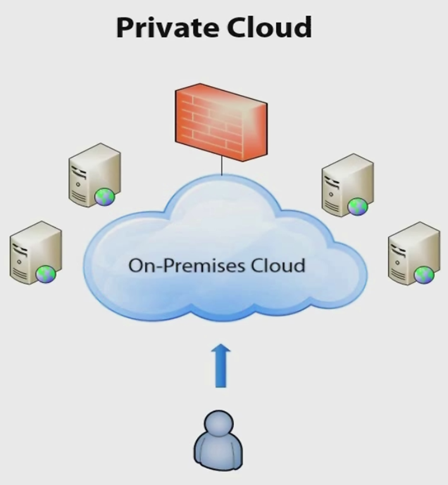


Figure 2.3 Private cloud.

* *Hybrid cloud*: model consists of two or more clouds on integrated together. Hybrid Cloud model allows sharing of infrastructure or meet the needs of data exchange.



Figure 2.4 Hybrid cloud.

* *Community cloud*: model in which cloud infrastructure is shared by several organizations for a community of users in the organization. These organizations share a CC infrastructure to improve investment efficiency and used.

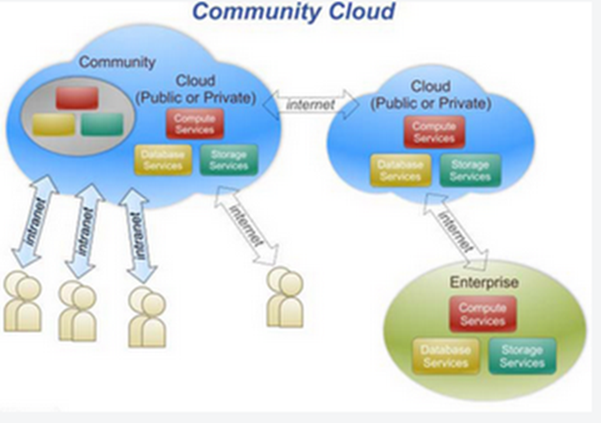


Figure 2.5 Community cloud.

1. **Customer Relationship Management (CRM)**
2. **Definition of CRM**

CRM (Customer Relationship Management) is a method that helps businesses reach and communicate with customers in a systematic and effective management of customer information, such as information about account, demand, communication ... to serve our customers better.

Through the system of customer relations, customer information will be updated and stored in the system database management. Through a search engine specific data, businesses can analyze, forming a list of potential customers and long-term strategy to set the appropriate customer care. Additionally, businesses can handle the issues and problems of customers quickly and efficiently.

1. **Some benefits of CRM**

* Improving business processes according to customer-oriented strategy.
* Develop a customer database structure and complete information items, at least easy to update and use.
* Raising the efficiency of handling customer information, helping clients better understand the approach should better customer.
* Enhanced sharing and updating customer information exchange between relevant departments.
* There are tools to evaluate the performance of the business unit in the company.
* Solve the contradiction between the values that customers and profits of the business (for understanding what the customer needs to provide the right services and products for customer).
* Increased profits from the complete process of customer service, increase customer numbers and revenue increases.
* Reduced cost of sales, marketing costs, management costs.

1. **Why is CRM?**

Recently, with the rapid revolution of technologies, the world is flatter and enterprises now have to face a great amount of competitions. Therefore, Customers have a wide range of choices and the customer-oriented is becoming one of the most important strategy for enterprises. Hence, Customer Relationship Management System has become a “must have” for every enterprises. It supports enterprises on the Customer-oriented strategy, maintaining the happy of current customer and bring in more new customers. In Vietnam, Customer service is one of the weak points of Vietnamese Enterprises and make them fail when facing big international enterprises with outstanding services. Therefore, bringing a chance to use CRM system can becomes the key for the competition problems. Successful combination of these two hot trends: Cloud Computing and CRM could make a huge change for the Vietnamese in the market. So in this thesis, we choose CRM among other enterprises software like SCM, ERP etc.

After this chapter, we gain a basic knowledge of cloud computing, CRM system and the advantages they can brings to business. With this knowledge, in the next chapter, we can find out the best solution and type of service that can help we solve our problem.

# **: Cloud Computing Solutions**

1. **Introduction**

In order to find the solution for building CRM application on the cloud, we will go through different cloud computing solutions and CRM open source applications on the market and their features. Hence, we can choose out the best ones to implement a potential and suitable solution for building CRM on cloud architect.

1. **The Overview of some CC solution**

* *CloudStack*: is open source software designed to deploy and manage large networks of virtual machines, as a highly available, highly scalable Infrastructure as a Service (IaaS) cloud computing platform. CloudStack is used by a number of service providers to offer public cloud services, and by many companies to provide an on-premises (private) cloud offering, or as part of a hybrid cloud solution. [2]
* *OpenNebula*: is a cloud computing platform for managing heterogeneous distributed data center infrastructures. The OpenNebula platform manages a data center's virtual infrastructure to build private, public and hybrid implementations of infrastructure as a service. OpenNebula is free and open-source software, subject to the requirements of the Apache License version2. [3]
* *Eucalyptus*: Eucalyptus is a free and open-source computer software for building Amazon Web Services (AWS)-compatible private and hybrid cloud computing environments marketed by the company Eucalyptus Systems. Eucalyptus is the acronym for Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems. Eucalyptus enables pooling compute, storage, and network resources that can be dynamically scaled up or down as application workloads change. [4]
* *OpenStack*: is a free and open-source cloud computing software platform. Users primarily deploy it as an infrastructure as a service (IaaS) solution. The technology consists of a series of interrelated projects that control pools of processing, storage, and networking resources throughout a data center-which users manage through a web-based dashboard, command-line tools, or a RESTful API. OpenStack.org released it under the terms of the Apache License. [5]

1. **OpenStack**
2. **Definition of OpenStack**

OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering their users to provision resources through a web interface. [6]

OpenStack project is cloud computing open source for building public and private cloud. OpenStack platform provide an IaaS (infrastructure as a service).



Figure 3.1 The position of OpenStack.

From Figure 3.1 we can see:

* The bottom is the hardware of OpenStack which is virtualized to share application, users.
* The top is your application, i.e. the software that we use.
* OpenStack is in the middle section 2 above, the components of OpenStack, but in different modules illustration of basic components: Dashboard, Compute, Networking, API, storage...

1. **History Founder of OpenStack**

The product was created by hosting company Rackspace and the NASA in 2010, OpenStack has the advantage of an open source software, is contributing built by a community of users worldwide.

1. **Why we choose OpenStack**

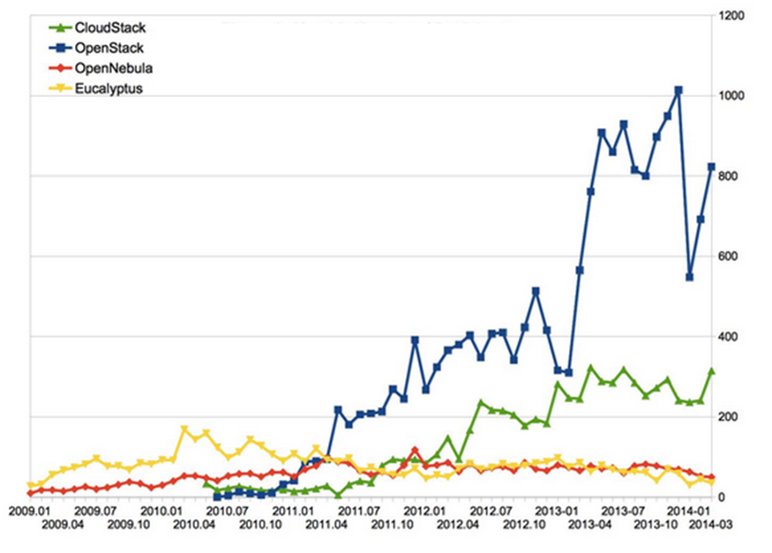


Figure 3.2 Monthly number of participant [7]

In four projects, we can see that the number of active participation of the OpenStack project is higher than the remains. The growth of high OpenStack. At March, 2014, people’s participation is higher than twice the CloudStack OpenStack, and give away Eucalyptus, OpenNebula.

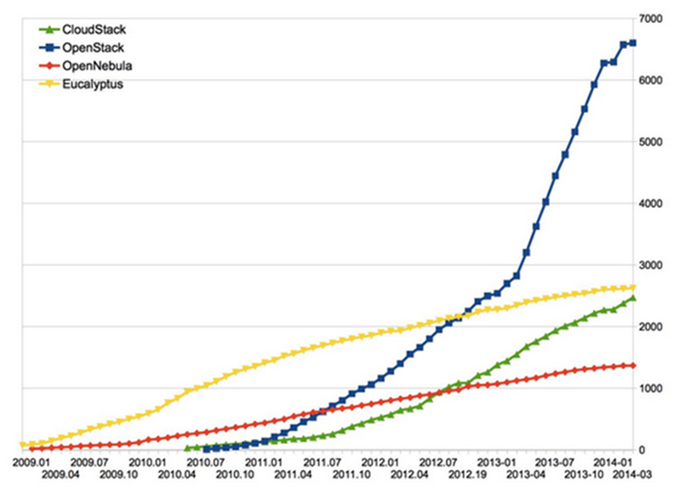


Figure 3.3 Accumulated community populations [7]

Accumulated community populations refers to the total capacitance and developers to contribute to the project. OpenStack community populations have projected significant growth, two times CloudStack, Eucalyptus, and approximately four times OpenNebula.



Figure 3.4: Accumulated developer population [7]

Accumulated developer population mention of the fine contributions and project code. OpenStack significant growth, far beyond the competition ten times.

* From Figure 3.2, 3.3, 3.4 we can see OpenStack is outperformed than the other clouds.

1. **Main component of OpenStack-IceHouse**

OpenStack IceHouse consist of ten components:

* *Compute (Nova):* The management module and provides a virtual machine. The name of its development Nova, so when you read about the understanding that OpenStack Nova compute. It supports multiple hypervisors including KVM, QEMU, LXC ... Compute XenServer is a powerful tool that can control the entire work: networking, CPU, storage, memory, creation, control and elimination machine virtualization, security, access control. You can control all command or from the web interface on the dashboard.
* *Object Storage (Swift):* Redundant storage system expansion, ensuring data integrity and replication across cluster, with high fault tolerance, easy to extend. It provides storage services, much like the other services such as stored mediafire ... but here is dropbox service comes in OpenStack, and other storage method.
* *Networking (Neutron):* manage system and network IP addresses, enabling people to self-serving, add the network configuration. Neutron provides various network management network as flat, flat network DHCP, VLAN. The mission of the Neutron is the connection between the instances together, and the instance with public network.
* *Dashboard (Horizon):* provides administrators and users a graphical interface to access, provides automated and cloud resources. The design can be easily extended to help add to the product as well as external service billing, monitoring and other monitoring tools.
* *Identity Service (Keystone):* is composition for authentication, token, policy and service catalog for all other services of OpenStack. It is implemented through the OpenStack API Identity.
* *Image Service (Glance):* is managing the virtual disk image. Glance supports RAW images, Hyper-V (VHD), Virtual Box (VDI), qemu (qcow2) and VMware (VMDK, OVF). You can do: update the virtual disk images, configure the public and private image and control access to them, and of course you can create and delete them.
* *Telemetry (Ceilometer):* is a development project aims to provide a single point of contact for billing system to get all the measurements needed to set up the payment to the customer, the module operates on all present the main components of OpenStack and it can support new components of OpenStack future.
* *Orchestration (Heat):* Heat supply-Orchestration template allows users to create most of the resources OpenStack. It also includes other features such as automatic scaling.
* *Database (Trove):* Provides a database to easy management for both relational and non-relational database engines

1. **Open Source CRM solution**

CRM system is rising to become a must have software for every enterprises. Hence, the development of open source CRM also become stronger with variable choices in the market. The below figure shows the top 10 Open Source CRM system in the market and their capability.

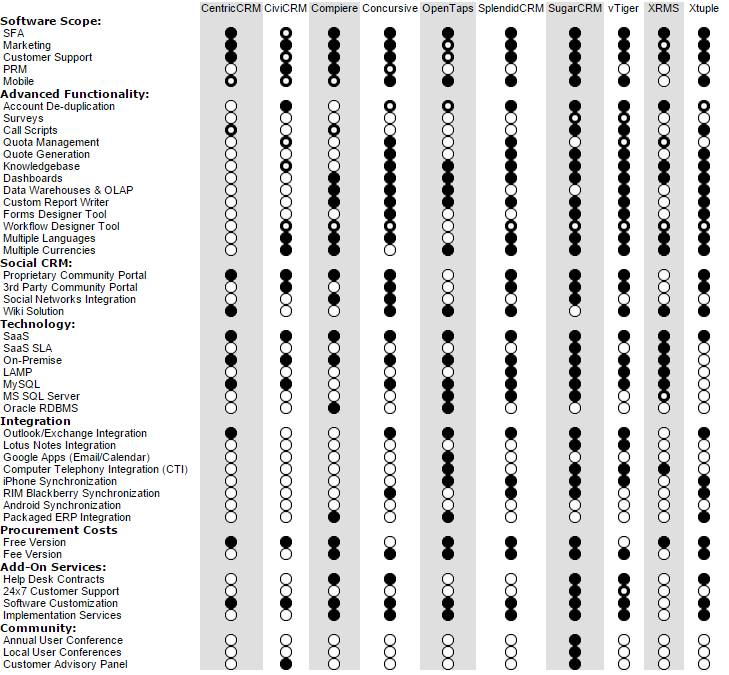


Figure 3.5 Top 10 CRM solution

With open source solution, we can greatly increase the cost for the solution, which is one of the main targets to solve the problem to help bring a chance for Medium and Small Enterprises.

In the figure, we can see that SugarCRM is supply most of the features and it also the biggest Open Source CRM standard right now. Choosing SugarCRM, we can have a great support from big community and SugarCRM team. In the future we can upgrade the software license or develop our own module based on the current Software.

From different available solutions, we can pick out the most suitable ones: Openstack and SugarCRM, they are well known open source software and can be integrate with each other to solve our problem.

# **: System Implementation**

1. **Introduction**

After choose out 2 solutions on the last chapter - OpenStack for building cloud system and SugarCRM, we will implement them on the real physical system to estimate the possibility to apply it to solve the problem. We will deploy 4 modules supported by SugarCRM Community edition includes: Customer, Potential Customer, Opportunity and Email Marketing on a cloud machine of OpenStack running Linux operation system.

1. **OpenStack installation and system design**
2. **Design system**

* The basic architecture in main page docs.openstack.org is two (controller and compute node) or three nodes (controller, network and compute node), but that model is not optimal because it is difficult to expand later. For example, image storage controller, if you want to add another node, the new controller node must contain the image of the old node (the image Windows Server 2012 RC is 16GB), leading to the accrual data.
* Ideal model proposed of my group is 5 nodes (controller, network, compute, block and swift node).
* Reduce the load on the controller nodes. The image will be saved elsewhere.
* Easy to expand. Block node will take over the task of providing additional storage space for the image if necessary.
* Not to be redundant data. The image is stored in Swift node. The architecture of the data will be difficult swift failure.
* Easily identify and correct the error as necessary.

Table 4.1 The host to install OpenStack

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **HOST** | **CPU** | **RAM** | **DISK** | **NICs** |
| Controller | 4 | 3 GB | 50 GB | Eth0: 192.168.32.71  Eth1: 10.10.10.71 |
| Network | 1 | 512 MB | 20 GB | Eth0: 192.168.32.72  Eth1: 10.10.10.72  Eth2: 10.10.20.72 |
| Compute | 8 | 4 GB | 100 GB | Eth0: 192.168.32.73  Eth1: 10.10.10.73  Eth2: 10.10.20.73 |
| Block | 1 | 512 MB | 20 GB | Eth0: 192.168.32.80  Eth1: 10.10.10.80 |
| Swift | 1 | 512 MB | 40 GB | Eth0: 192.168.32.90  Eth1: 10.10.10.90 |

1. **The deployment model**

* OpenStack components are installed and deployed on 5 nodes:

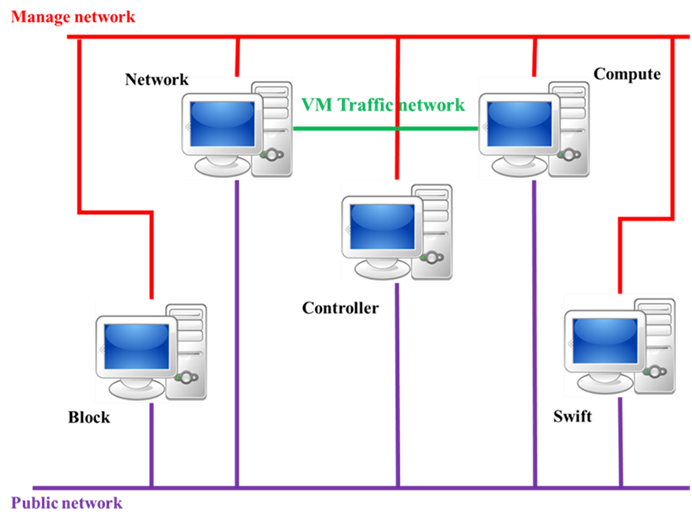
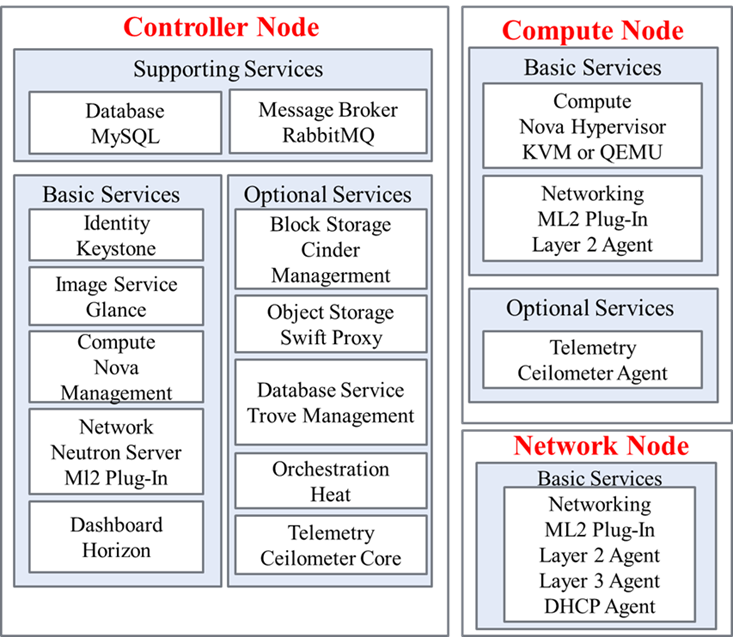


Figure 4.1 The model to install OpenStack IceHouse.

* Manage network (10.10.10.x): A network for administration, not accessible to the public Internet.
* Public network (192.168.32.x): This network is to connect to the Internet to download some packages necessary for OpenStack.
* Virtual machine (VM) network (10.10.20.x): This network is using for traffic network between network node and compute node.
* Service in each nodes



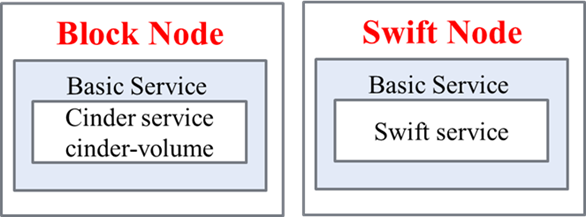


Figure 4.2 Services in each nodes.

1. **Install the following service on each appropriate node – install OpenStack IceHouse**

* ***Controller node:***
* Setup IP, hostname: *run: bash controller-1-IP.sh*
* Setup ntp service : *bash controller-2-NTP.sh*

Network Time Protocol (NTP): We need to synchronize services across multiple nodes. It makes sure that all node can be action in the same time.



Figure . Verify NTP synchronization.

* Setup database service: *bash controller-3-Database.sh*

OpenStack services need a database to store information. We use a MySQL install in controller node to store service database.

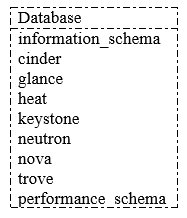


Figure . The database name of the services.

* Setup Keystone service: *bash controller-4-Keystone.sh*

The Identity Service performs the following functions:

* Provides the ability to validate, set the decentralization policy for the OpenStack project.
* Keystone provide policies, token, and validate the entire cloud infrastructure
* Setup users, role, tenant: *bash controller-5-CreateUser.sh*

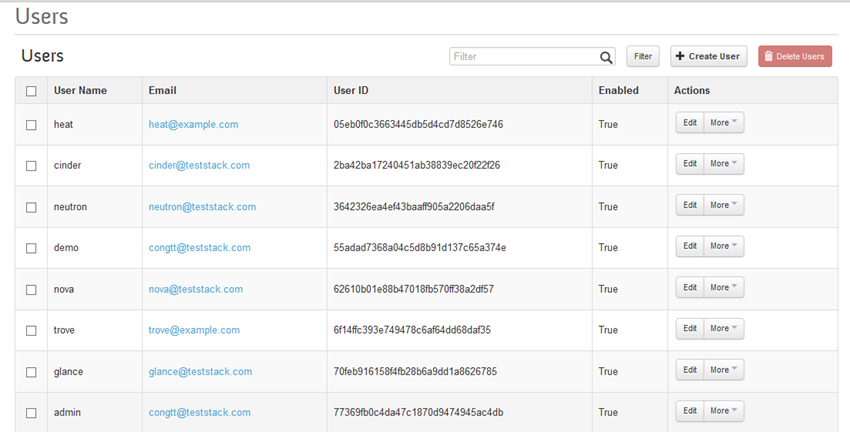


Figure 4.5 The users of the system.

* Setup Glance service: *bash controller-6-Glance.sh*

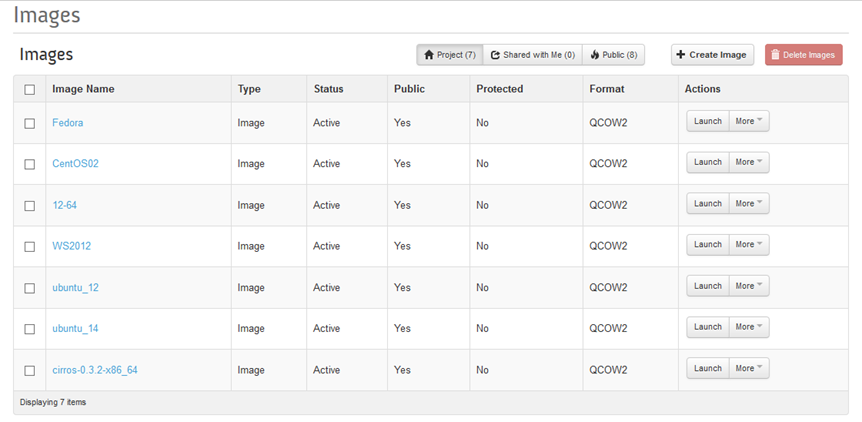


Figure 4.6 The image of the OpenStack.

* Setup Nova service: *bash controller-7-Nova.sh*

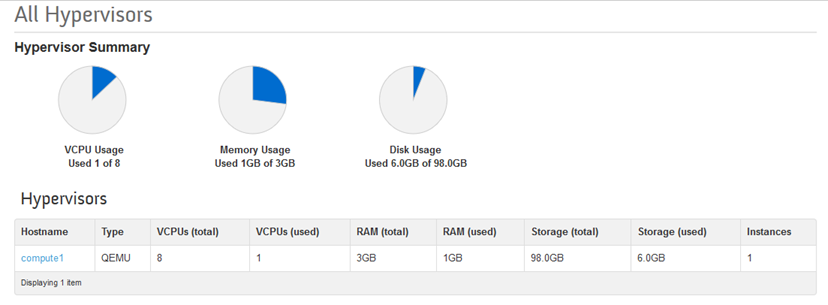


Figure . Hypervisor of compute node.

* Setup Neutron service: *bash controller-8-Neutron.sh*
* ***Network node:***
* Configuration IP, hostname: *bash network-1-IP.sh*
* Setup Neutron service: *bash network-2-Neutron.sh*

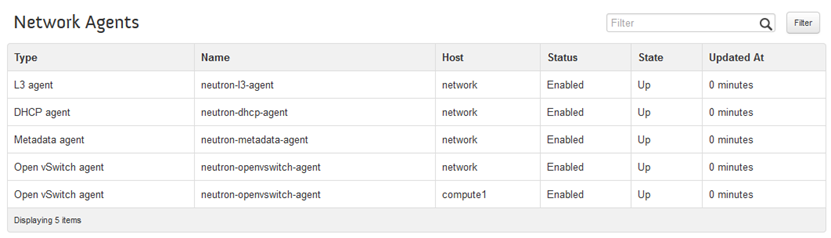


Figure 4.8: Network agents.

* ***Compute node:***
* Configuration IP, hostname: *bash compute-1-IP.sh*
* Setup Nova service: *bash compute-2-Nova.sh*

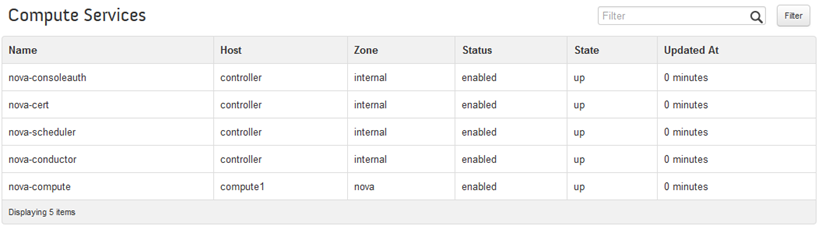


Figure 4.9 Compute services.

* ***Controller node:***
* Setup Dashboard-Horizon: *bash controller-9-Horizon.sh*

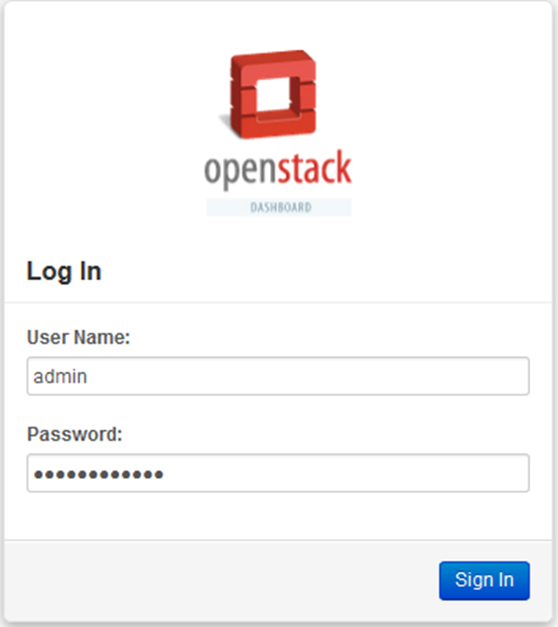


Figure 4.10 The login of horizon.

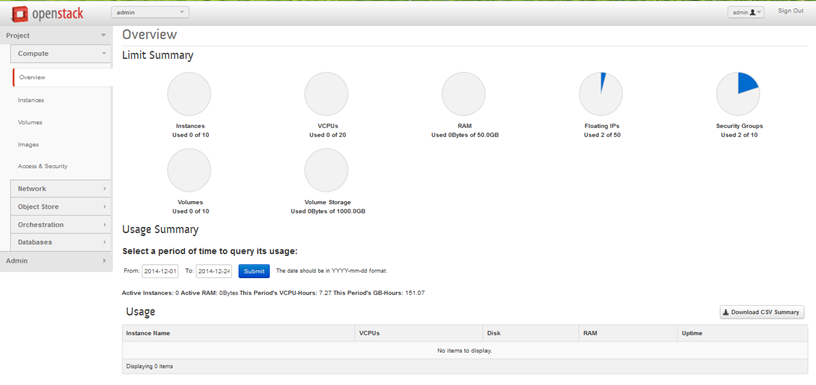


Figure 4.11 The overview of horizon.

* Create network: *bash controller-10-CreateNetwork.sh*

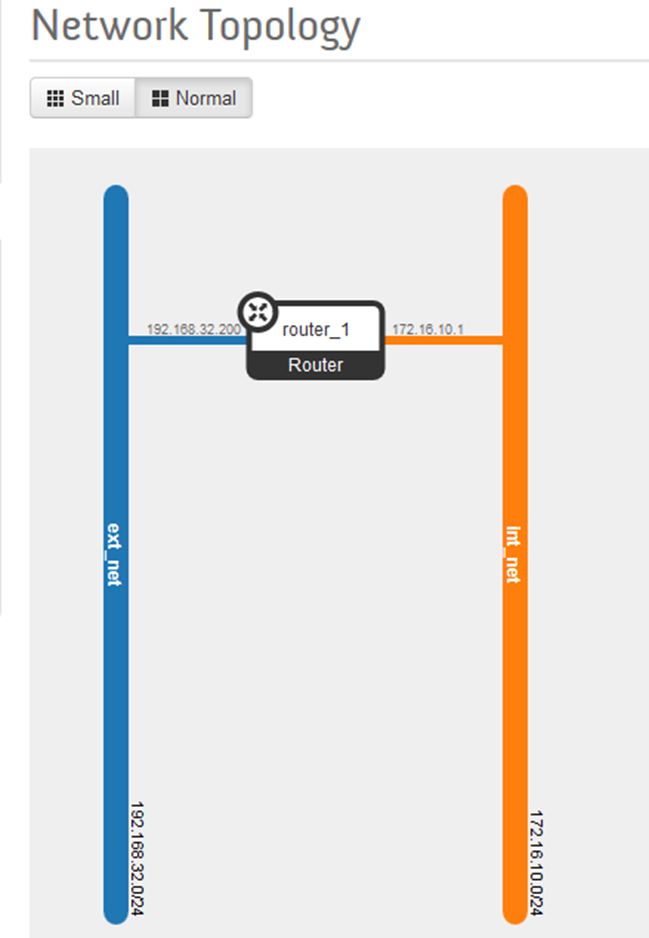


Figure 4.12 Network topology.

* Install Cinder service: *bash controller-11-Cinder.sh*

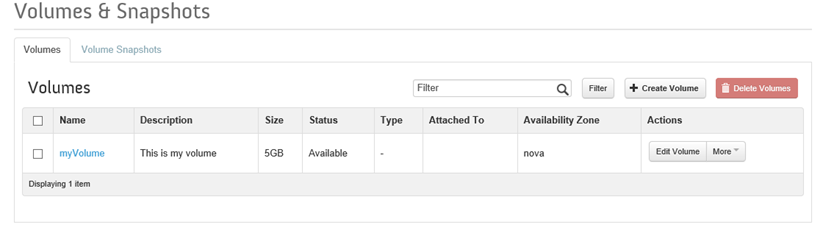


Figure 4.13 Create a new volume.

* Install Swift service: *bash controller-12-Swift.sh*

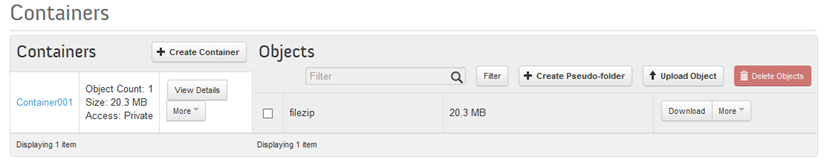


Figure 4.14 Swift store an object.

* Install Ceilometer service: *bash controller-13-Ceilometer.sh*

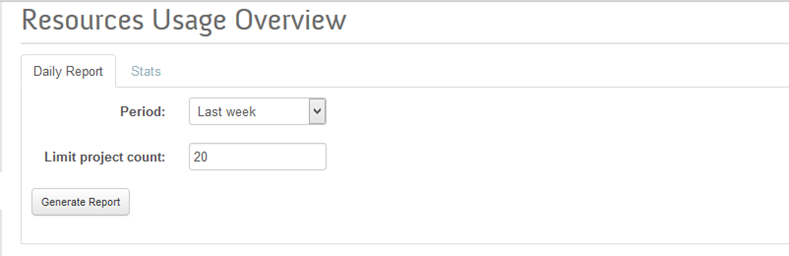


Figure 4.15 Resource usage overview.

* Setup Heat service: *bash controller-14-Heat.sh*

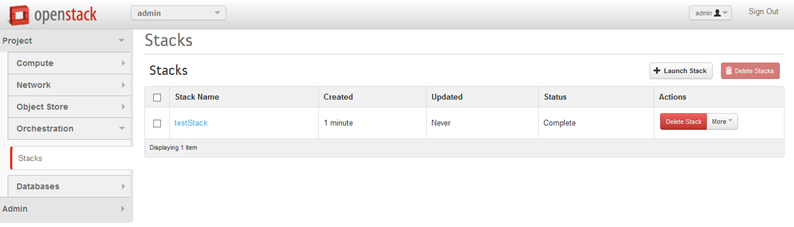


Figure 4.16 Create a template.

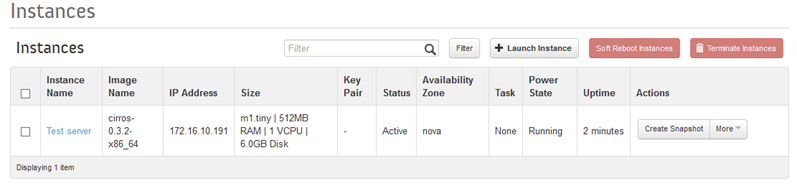


Figure 4.17 The instance created by template above.

* Setup Trove service: *bash controller-15-Trove.sh*
* ***Block node:***
* Configuration IP, hostname: *bash block-1-IP.sh*
* Setup Cinder service: *bash block-2-Cinder.sh*



Figure 4.18 Attached the instance to the volume.

* ***Swift node:***
* Configuration IP, hostname: *bash swift-1-IP.sh*
* Setup Swift service: *bash swift-2-Swift.sh*

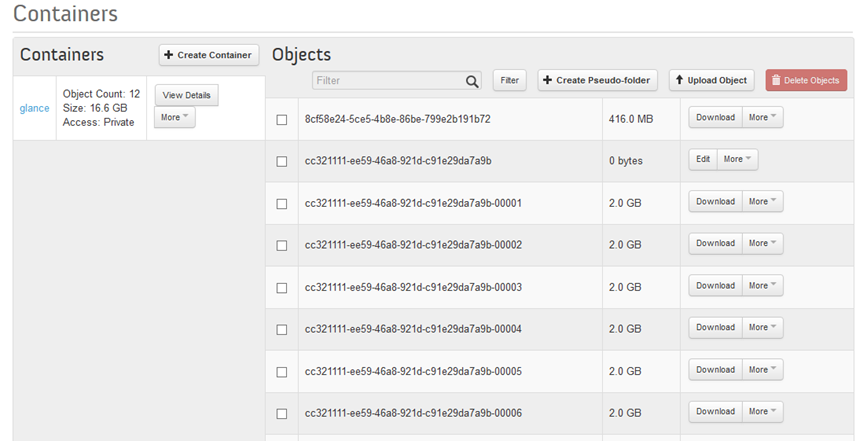


Figure 4.19 The swift interface stored an image.

* ***Compute node:***
* Setup Ceilometer service: *bash compute-3-Ceilometer.sh*

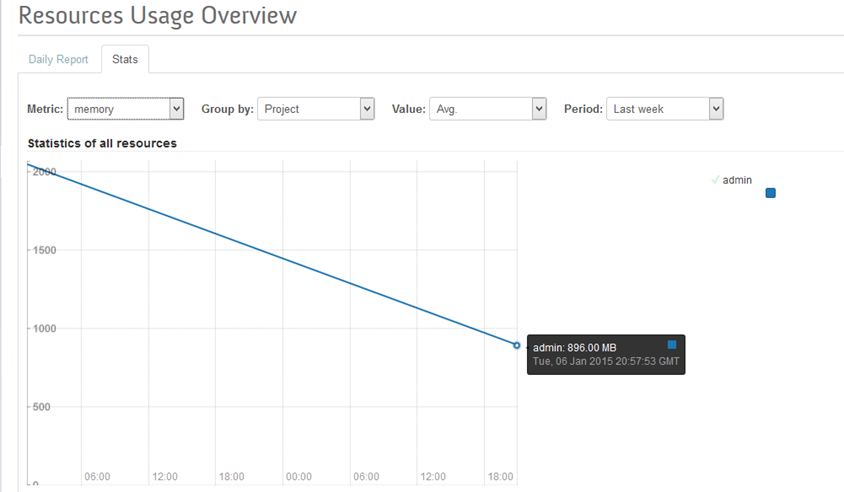


Figure 4.20 Resources usage overview.

1. **Deploy some module of CRM**
2. **Install sugarCRM on virtual machine**

* Create a virtual machine (instance)
* We create an instance with the following configuration



Figure . The configuration of the instance*.*

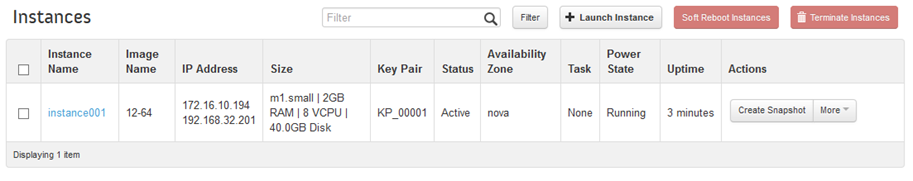


Figure 4.22 The Ubuntu instance in Horizon.

* We are using PuTTy to create user through KeyPair to access into the Ubuntu.

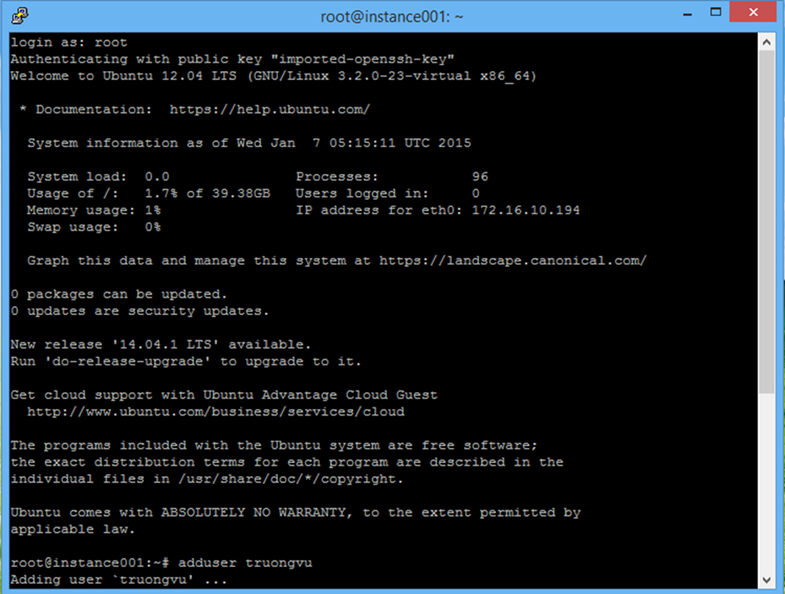


Figure 4.23 Using PuTTY to add new user.

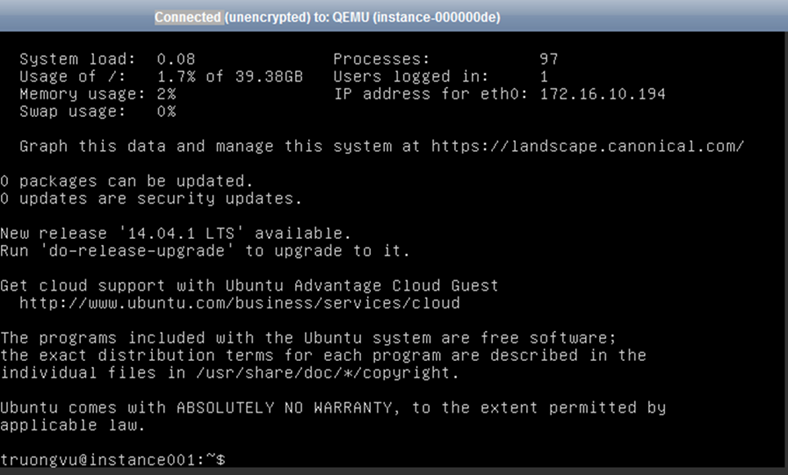


Figure 4.24 Login to the instance.

* Bringing sugarCRM into cloud
* We install sugarCRM and some package necessary by one script. This script make sure that at time we install, this is the latest version.

#!/bin/bash -ex

DEFAULT\_PASS='openstack123'

MYSQL\_PASS="$DEFAULT\_PASS"

echo mysql-server mysql-server/root\_password password $MYSQL\_PASS | debconf-set-selections

echo mysql-server mysql-server/root\_password\_again password $MYSQL\_PASS | debconf-set-selections

apt-get install lamp-server^ -y

apt-get install libapache2-mod-php5 libapache2-mod-perl2 php5 php5-cli php5-common php5-curl php5-dev php5-gd php5-imap php5-ldap unzip php5-mhash php5-mysql php5-odbc curl libwww-perl imagemagick -y

cat <<EOF | mysql -u root -p$MYSQL\_PASS

create database sugarcrmdb;

create user 'sugar'@'localhost' identified by 'openstack123';

grant all privileges on sugarcrmdb.\* to 'sugar'@'localhost';

flush privileges;

EOF

cd /var/www/

wget -O sugarCE.zip http://sourceforge.net/projects/sugarcrm/files/latest/download?source=files

unzip sugarCE.zip

mkdir sugarcrm

sgcrm=$(ls|grep SugarCE)

path="$sgcrm/."

cp -a $path /var/www/sugarcrm/

chown -R www-data:www-data sugarcrm

cd /var/www/sugarcrm/

chmod 766 config.php

chmod 766 custom

chmod -R 766 data

chmod -R 766 cache

chmod -R 766 modules

sed -i 's/2M/64M/g' /etc/php5/apache2/php.ini

sed -i 's/html//g' /etc/apache2/sites-available/000-default.conf

/etc/init.d/apache2 restart

echo "=====> COMPLETE <====="

* After that we have:

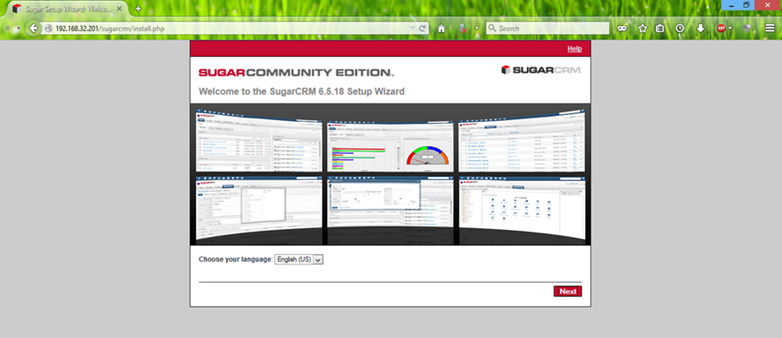


Figure 4.25 Setup sugarCRM through web browser.



Figure 4.26 The login interface of sugarCRM.

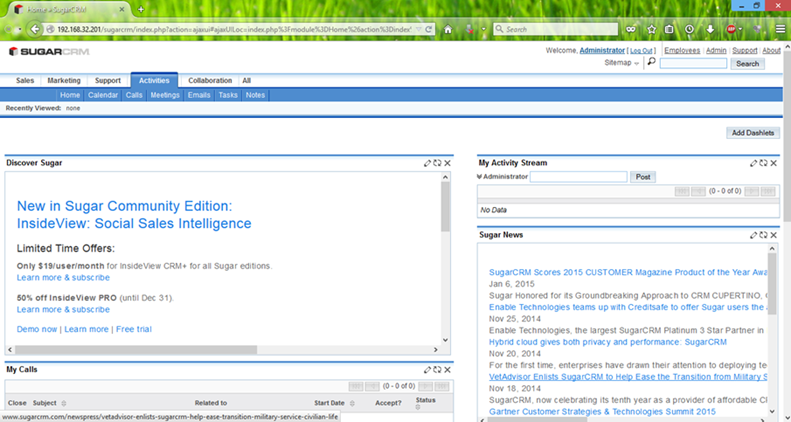


Figure 4.27 The main interface of sugarCRM.

1. **Deploy module account**

* Overview of module:

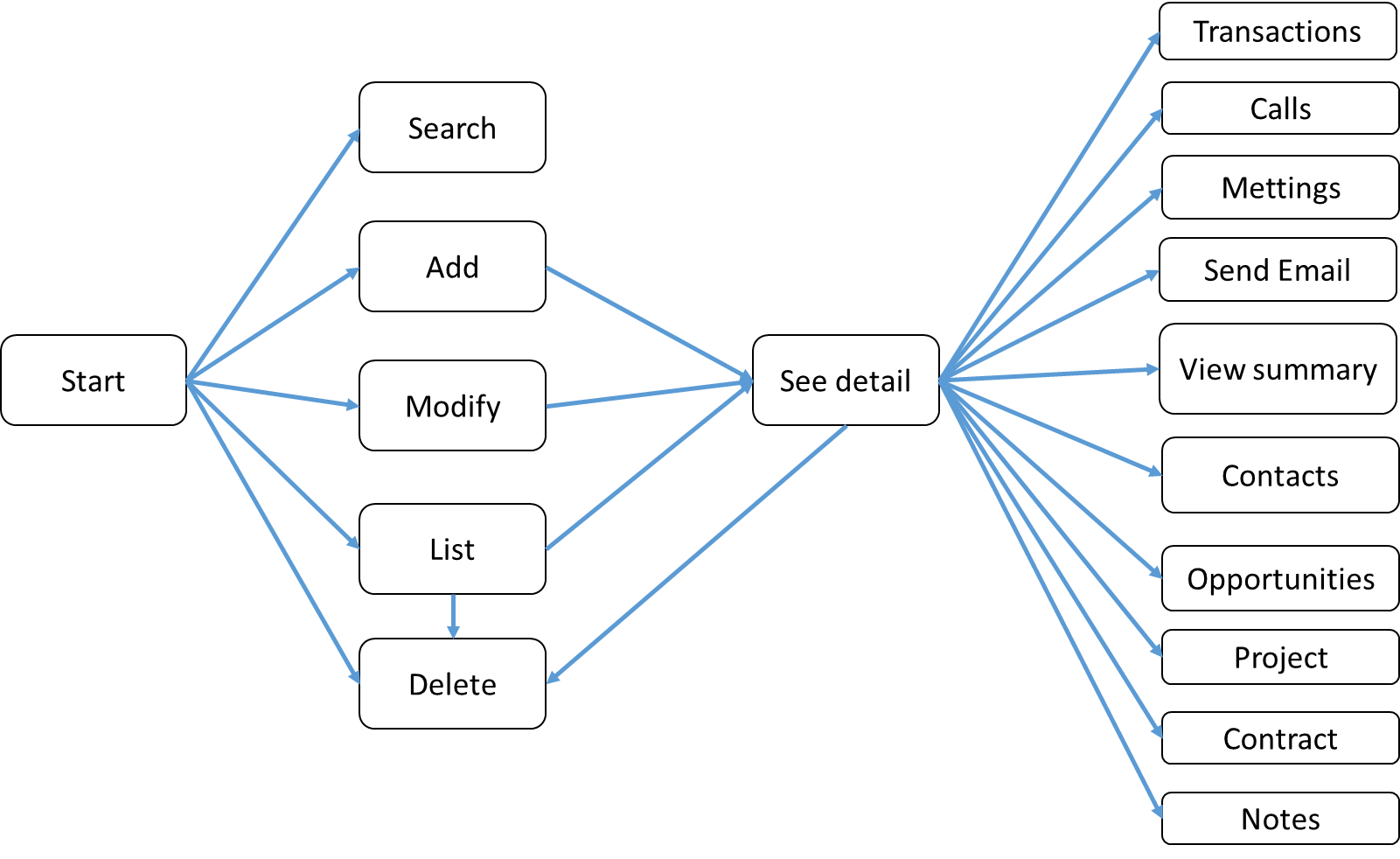


Figure 4.28 Module account overview.

Module account is where the account creation and management:

* Management of information relating to customers, such as customer name, phone number, address…
* Manage the activities of meetings, calls, tasks… in the past and what to do, what has been done.
* Management notes, file attachments and email.
* Manage contact information related to customers.
* Manage sales opportunities, contracts.
* Management of cases such as complaints, feedback.

From Figure 4.28, we can see:

* There are some function such as: search, add, modify, list, delete customer.
* When we add, modify it will return a list of customer. When we see a detail, we can also delete a customer.
* With see detail screen, we can manage transactions, calls, meetings, send email…
* Add a new customer.

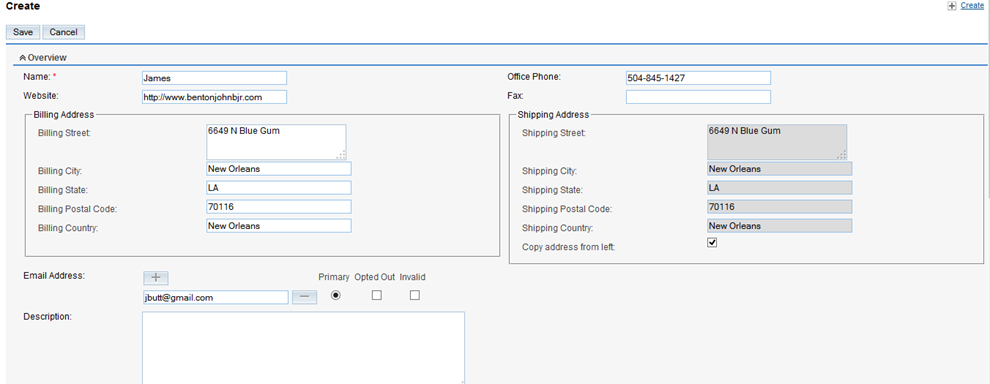


Figure 4.29 Add a new customer.

* Finding customer

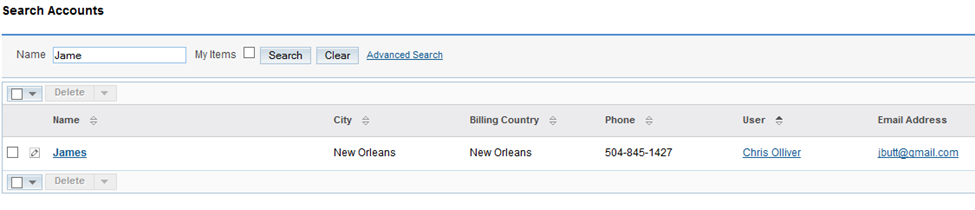


Figure 4.30 Finding a customer named James.

* Edit a customer

We change a customer named James by James Butt.



Figure 4.31 Edit a customer.

* Merge two or more customers.

Sometimes we add a new customer who existing in the database, and we would like to update the information, we can you merge function.

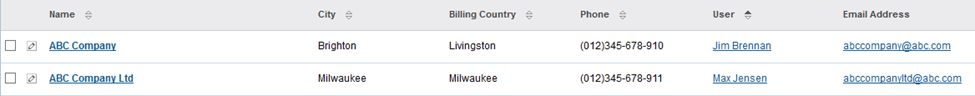


Figure 4.32 Two companies we want to merge.

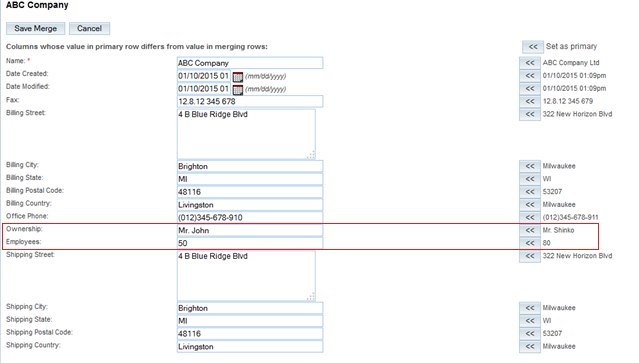


Figure 4.33 Merge two companies.

ABC company is in the left (information will be save) and ABC company Ltd information is in the right.

We would like to update employees into 80 in the right, and the ownership is Mr. Shinko.

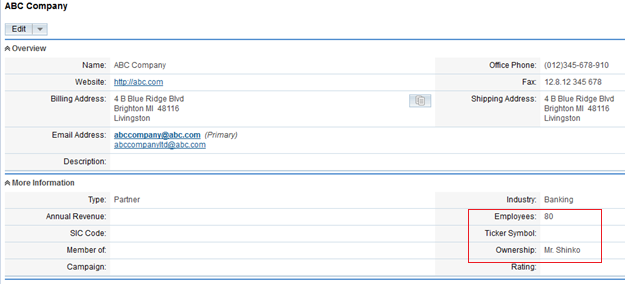


Figure 4.34 The information after merging.

* Create an activity:

We can create a task and assign it for concrete employee. When the employee login into system, he or she will see the task was assigned and will finished it.

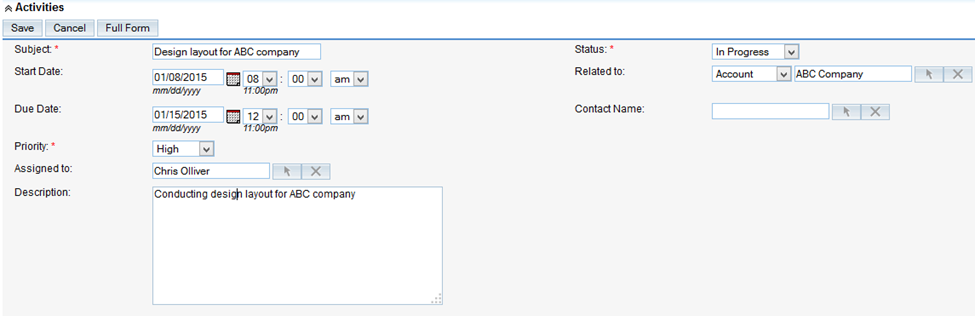


Figure 4.35 Create an activity.

1. **Deploy module Leads - Potential customers**

* Overview of module:

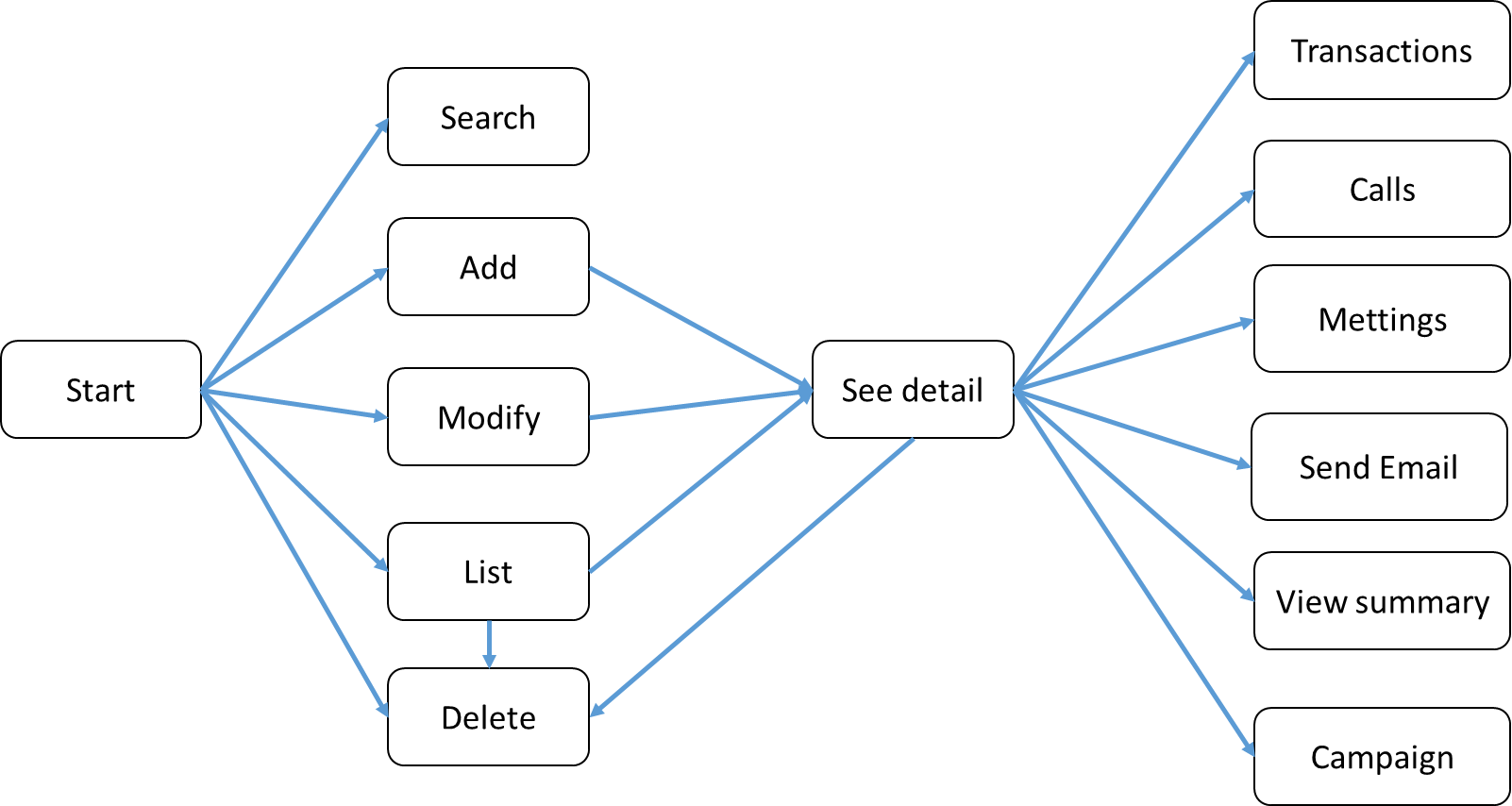


Figure 4.36 Module Leads.

Leads will be contacted early in the sales process. Use module Leads to create, manage and monitor of business processes, the assembly unqualified and related activities, business strategies.

Potential customers are customers in the future, not the present time, or have already, but not yet exploited, or customer value provides very little, the customer is not available.

For example, we are a provider TV 4K, it has high price, so it should be a few buyers, but in the future this product may be at any home. We can say this product has a large number of "potential customers" in the future.

The amount of potential customers are customers bring little value at this time, but can bring great value in the future.

* Create a potential customer

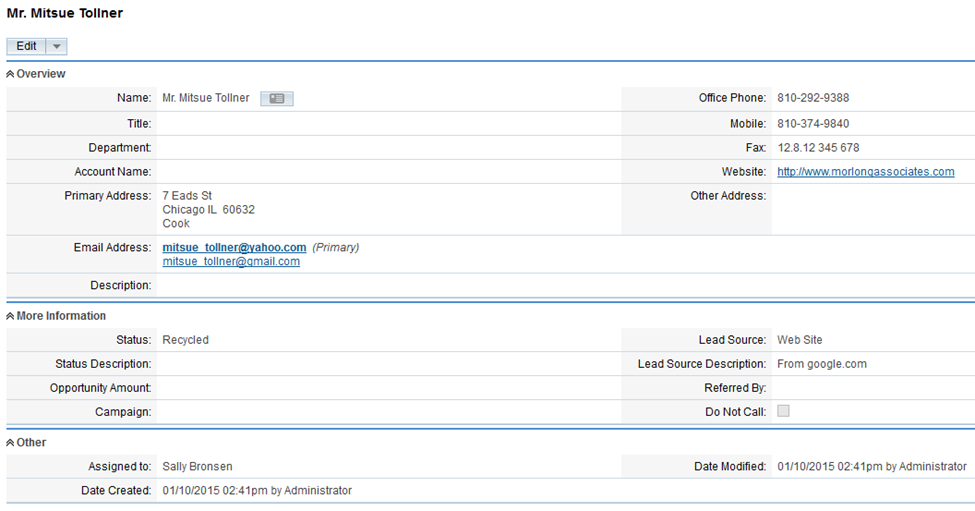


Figure 4.37 Create a potential customer - Mr. Mitsue Toller.

* Mass update

We used mass update to update some record in the same time.

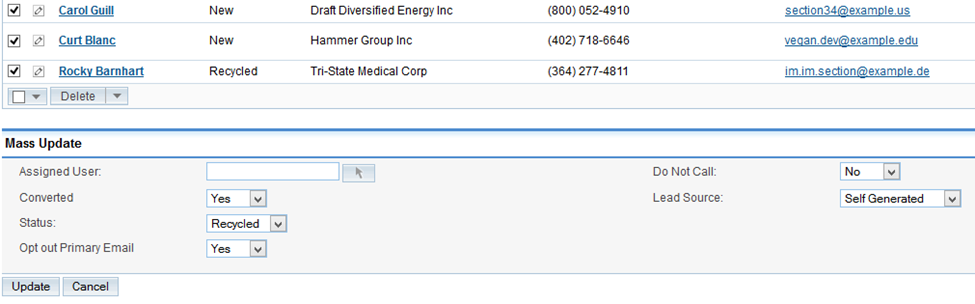
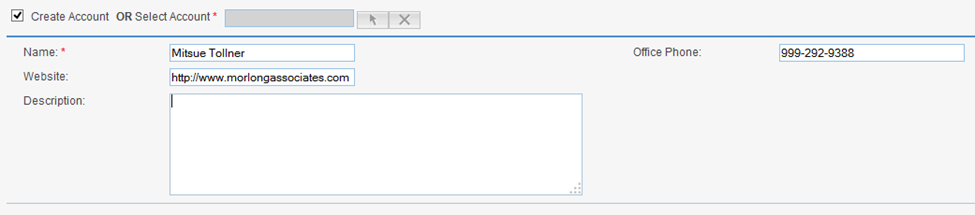
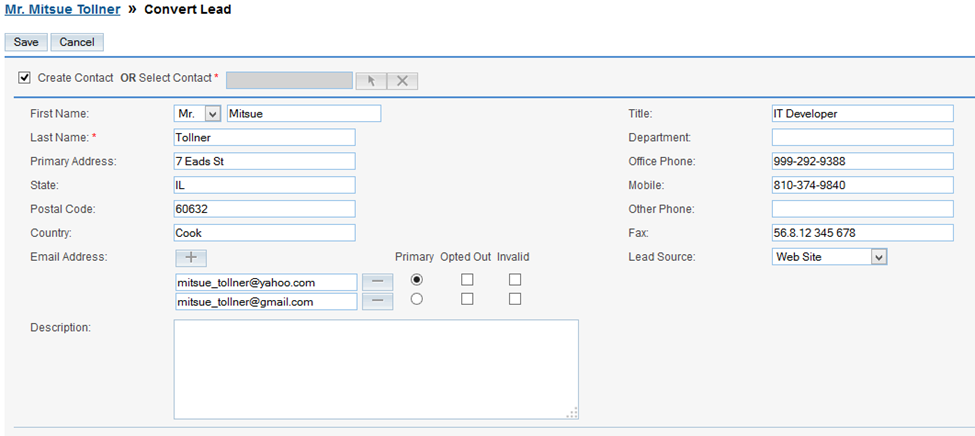


Figure 4.38 Mass update

* Convert leads

Converting leads into customers. In the business process approach leads, business staff will evaluate the opportunity leads, really need it or not. This function allows us to move from leads to the customer. During the conversion process, we can do more such as opportunities, create node, call logs, meeting scheduler, create task.



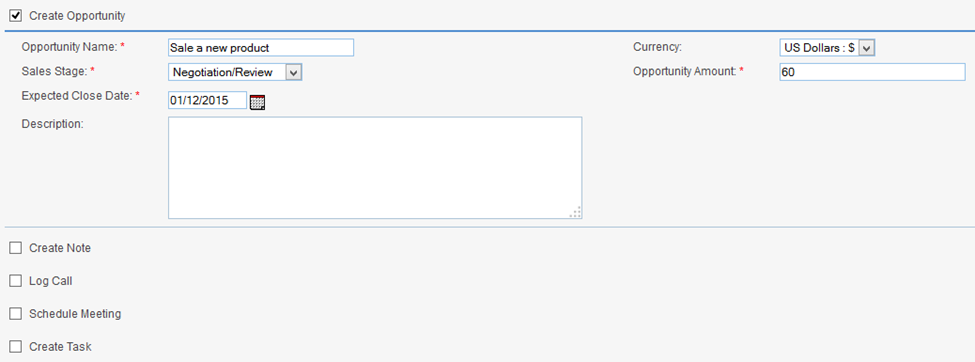


Figure 4.39 Convert leads.

* Create task

After we convert it, we can setup a call for some purpose such as to inform a price.

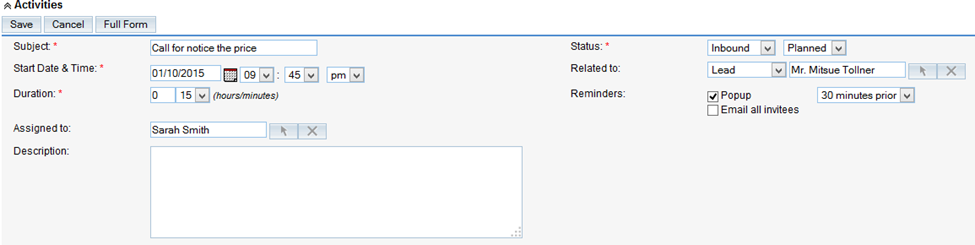


Figure 4.40 Create log call.

Nearly time related the task, the prompt dialog box will appear, make sure you do not forget the task.

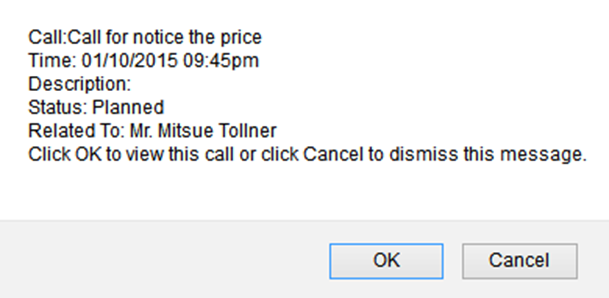


Figure 4.41 Prompt dialog box.

1. **Deploy module Opportunity**

* Overview of module

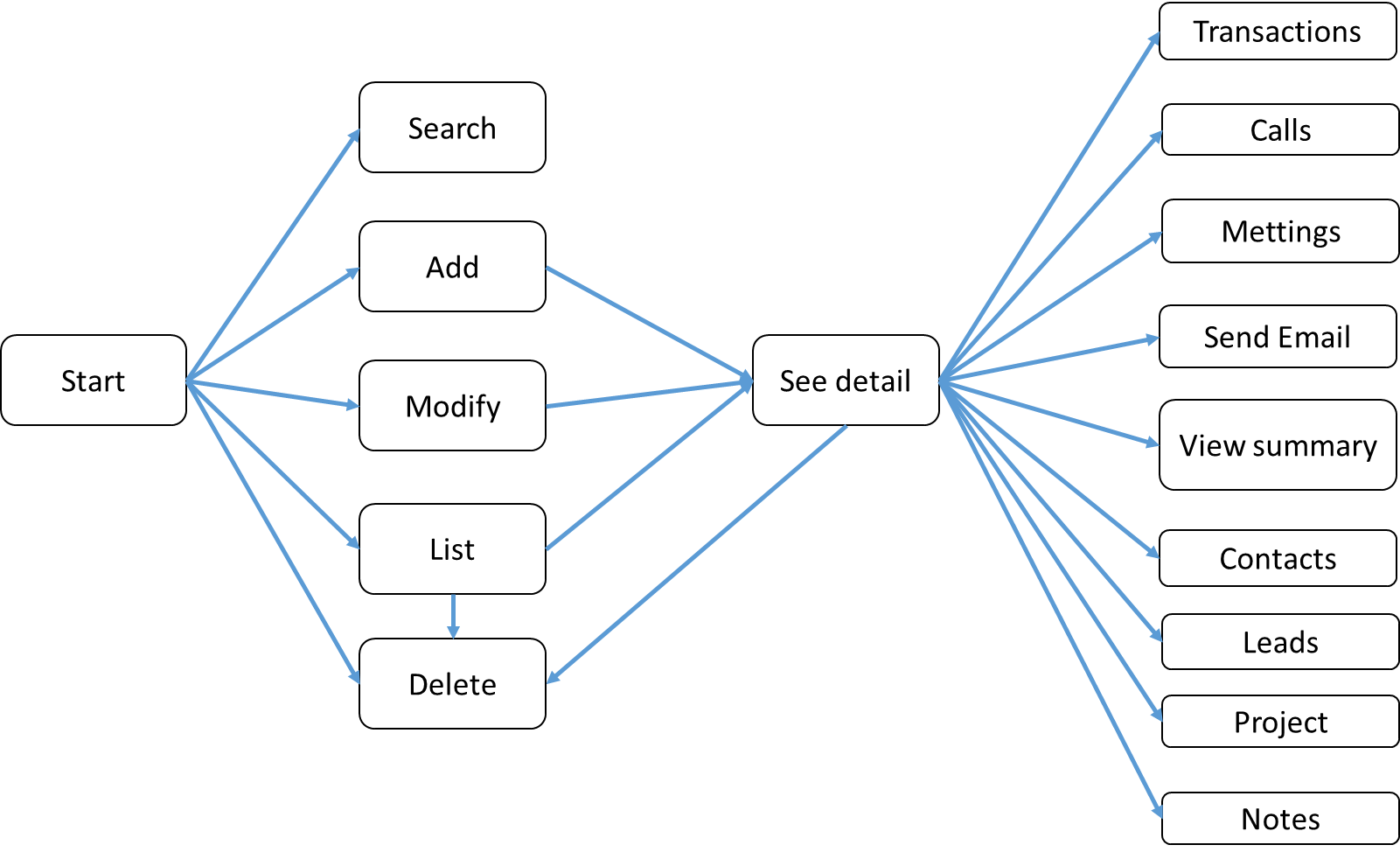


Figure 4.42 Module opportunity overview.

Use module opportunity to create and manage business opportunities, and potential leads related records.

An opportunity can only be associated with a single customer, but a customer can have multiple opportunities. And an opportunity can only be associated with a single campaign.

* Create an opportunity

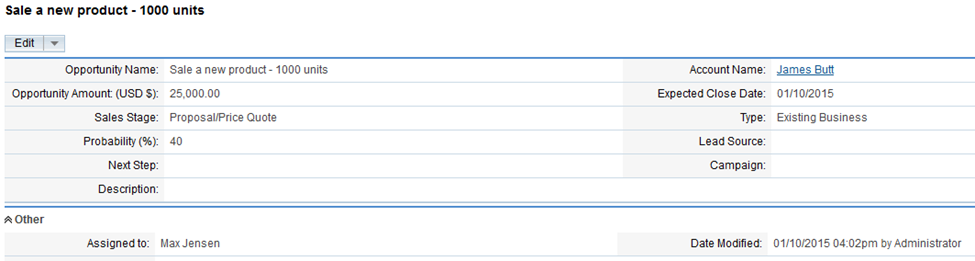


Figure 4.43 Create an opportunity.

* Create a task

We create an activity for the opportunity and assign it to the employee.

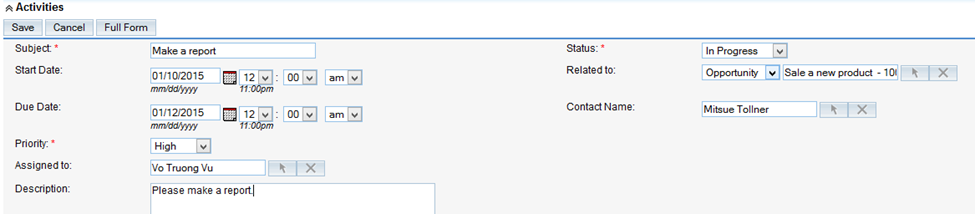


Figure 4.44 Create an activity for the opportunity.

This activity will save in the activities.

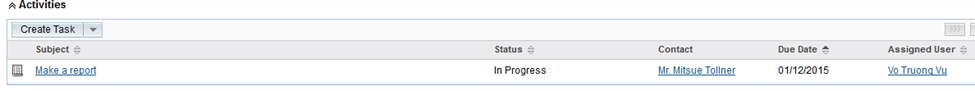


Figure 4.45 Task: Make a report.

When the employee login into the system, he or she will see the task and do it. When the employee finished and updated the status from in progress to completed, the task will save in history.



Figure 4.46 Task finished.

The report was created and attached into Notes

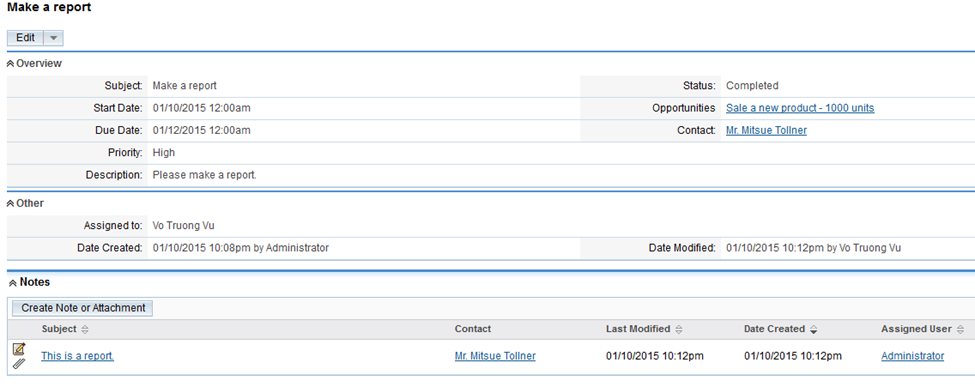


Figure 4.47 The content of the task after finished.

1. **Deploy Email Marketing**

* Overview of email marketing

Email Marketing is a form of direct advertising, including electronic mail (email) is used as a means of communicating with customers during transmission advertising information. The purpose of email marketing is to build relationships with new customers and maintain and care for the traditional customers, this will help the business is ongoing.

* Some benefits of email marketing:
* *Low-cost*: One of the most obvious benefits of email marketing is its lower cost compared to mainstream marketing channels. There are no print or postage costs and no fees paid in exchange for exposure on a certain billboard, magazine or television channel.
* *Improving tracking and feedback*
* Most types of marketing and advertising are difficult to measure.
* With email, everything can be tracked.
* See who clicked on certain links and much more.
* Get valuable feedback as you strive to grow your business.
* *Increasing customer loyalty and profitability*
* Deeper relationships drive more profitability
* Email is an easy and inexpensive way of establishing early and long-lasting relationships.
* Send newsletters, promotions, new service announcements or event invitations.
* Create a campaign

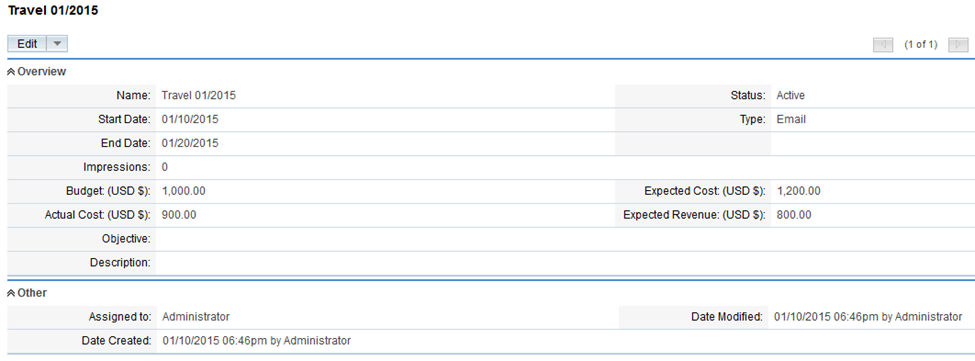


Figure 4.48 Campaign Travel 01/2015.

* Create a content of email

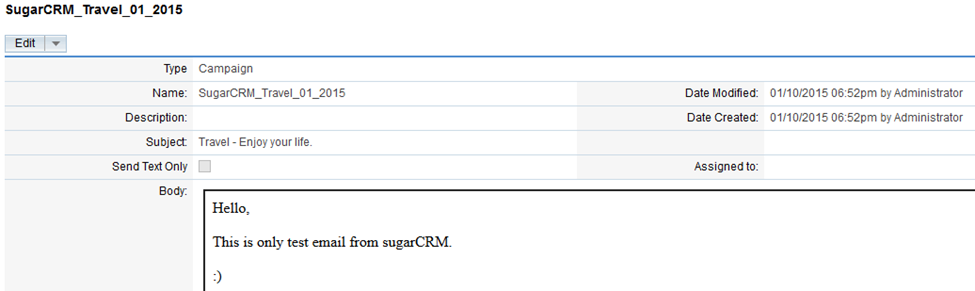


Figure 4.49 The email content.

* Send email into queue, it will send in suitable time you configured.

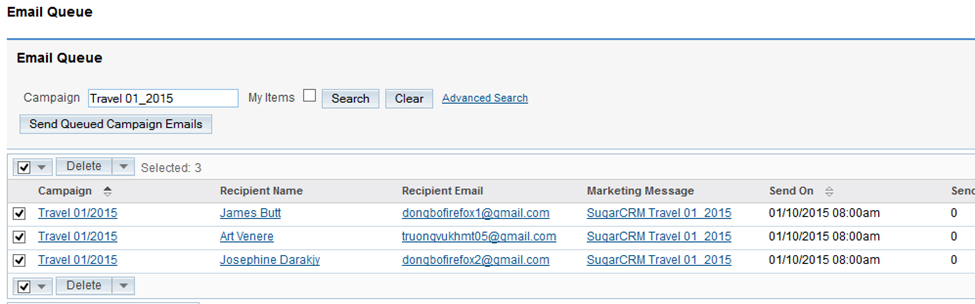


Figure 4.50 Email queue.

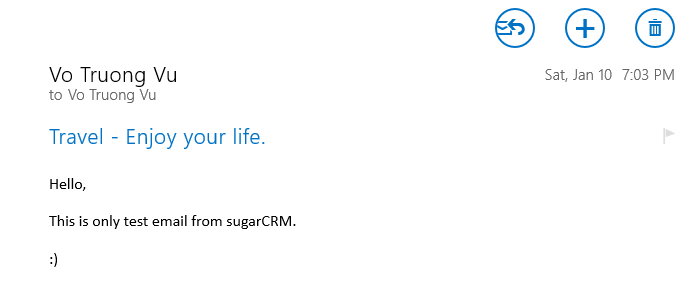


Figure 4.51 Email send success

1. **Summary**

* After this chapter, we known the way to deploy OpenStack IceHouse, bring SugarCRM from local into cloud.
* Deploy success 4 modules of SugarCRM that can be used for any businesses.

# **: System Testing and Result**

1. **The content of the test**

We want to test the performance of the virtual machine - instances created against real machine.

We are particularly interested in CPU performance and speed read and write (I/O) than other factors because it is two of the really important factor determining the overall virtual machine performance.

1. **Data preparation**

We test our system real computer vs virtual machine (instance) with sysbench, which is a well-known benchmark tool, allowing you to easily check the performance of the system such as CPU performance. I/O.

To measure file IO performance, we use this command:

*dd if=/dev/zero of=test bs=64k count=16k conv=fdatasync*

Testing time to install sugarCRM without sysbench

Configuration real and virtual machine-instance

Table . Configuration of two machines

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | OS | CPU | RAM | Disk |
| Real machine | Ubuntu | 8 | 4 GB | 100 GB |
| Virtual machine | Ubuntu | 8 | 2 GB | 40 GB |

1. **The result of the test performance**

* CPU Benchmark:

We used prime number checked

Table . The CPU benchmark

|  |  |  |
| --- | --- | --- |
| Prime number | Real machine | Virtual machine |
| 2000 | 21.8292s | 21.9196s |
| 4000 | 56.3301s | 56.2320s |
| 6000 | 99.5974s | 99.0916s |
| 8000 | 147.1970s | 147.4159s |
| 10000 | 201.7416s | 201.6541s |

.

* File IO Time

IO time of real machine:

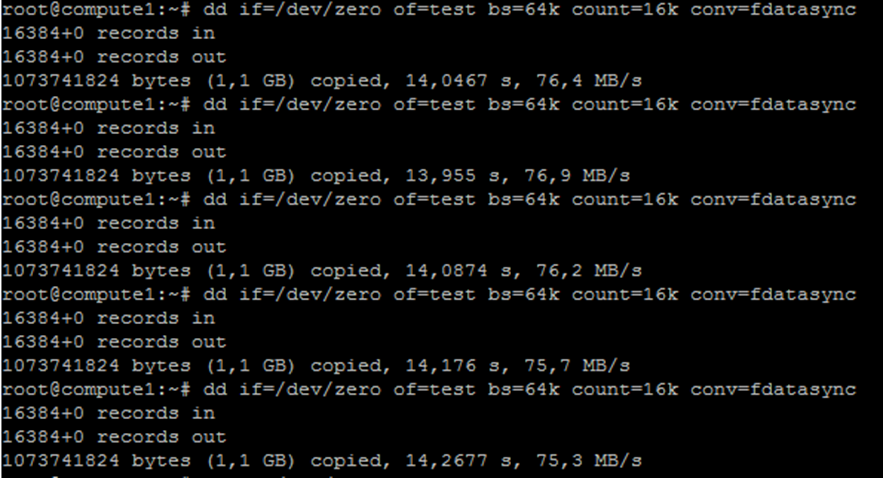
.

Figure 5.1 Real machine IO time.

IO time of virtual machine:



Figure 5.2 Virtual machine IO time.

We are summary it into the below table:

Table 5.3: Real and virtual machine IO time.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Real machine  (MB/s) | Virtual machine  (MB/s) |
| Attempt | 1 | 76.4 | 73.4 |
| 2 | 76.9 | 71.5 |
| 3 | 76.2 | 71.9 |
| 4 | 75.7 | 74.6 |
| 5 | 75.3 | 72.2 |
| **Average time** | | **76.1** | **72.72** |

* SugarCRM

This time is not include download package time.

Table 5.4: Time install sugarCRM.

|  |  |  |
| --- | --- | --- |
|  | Real machine | Virtual machine |
| **TIME** | 2m 32s | 2m 34s |

1. **Analysis**

* The processing speed of the CPU virtual machine close to the speed of the real machine.
* File IO of virtual machine is less than real machine because it is virtual disk, it through a virtual machine manager to interact to system resources, so it will take a longer time.
* The performance of the system is quite well and almost the same as the real system. It also bring the benefits which are mentioned in the theories that help we solve our problem. From this result, with further development, we can have a possible chance to apply this solution in practice to help solve the problem.

# **: Conclusion and Future Work**

1. **Conclusion**

* In this thesis, we gain a lot of knowledge about:
* Research:
* We understand the technical CC.
* We gain knowledge about the CRM.
* Deployment:
* We are complete to install and deploy a lot of module of the CC.
* Complete to bring CRM from local to CC.
* We deploy some modules of CRM.

We can apply the knowledge we studied to solve the practice problem

1. **Future Work**

* Future objectives of the research wants to build a larger test system, expansion of service providers such as diversification of the operating system, providing storage services, ... towards building a model form of open source cloud computing in fact complete, an IaaS provider for service operators.
* We would like to expand the Cloud system to serve not only the company that can give users leased use.
* Develop the solution to a complete package of service: as a service supplier, we can bring a chance to apply CRM system to all the Vietnamese enterprises with a reasonable prices without worry about all the technical matters to focus more in business problems.
* Develop our own SugarCRM module that suitable for each problem of enterprises. Apply more enterprises solution to bring different types of services such as SaaS.

**REFERENCES**

|  |  |
| --- | --- |
| [1] | P. Mell and T. Grance, "The NIST Definition of Cloud," *The NIST Definition of Cloud,* p. 2, 2011. |
| [2] | "Apache CloudStack™," Apache Software Foundation, [Online]. Available: http://cloudstack.apache.org/. [Accessed 05 January 2015]. |
| [3] | Ahmad-Al-Shishtawy and A. H. Payberah, "felix.site.ac.upc.edu," 2013 June 2013. [Online]. Available: felix.site.ac.upc.edu/Clommunity\_deliverables/D2.1.pdf. [Accessed 05 January 2015]. |
| [4] | Rajamanickam, "http://www.rtoz.org," QualityPoint Technologies, 12 September 2014. [Online]. Available: http://www.rtoz.org/2014/09/12/hp-acquires-cloud-sftware-startup-eucalyptus/. [Accessed 05 January 2015]. |
| [5] | P. Porecha, "https://zenodo.org," August 2014. [Online]. Available: https://zenodo.org/record/11782/files/CERN\_openlab\_Parin\_Porecha.pdf. [Accessed 05 January 2015]. |
| [6] | "http://docs.openstack.org," 07 January 2015. [Online]. Available: http://docs.openstack.org/training-guides/content/operator-getting-started.html. [Accessed 08 January 2015]. |
| [7] | Q. Jiang, "http://www.qyjohn.net," [Online]. Available: http://www.qyjohn.net/wp-content/uploads/2014/04/CY14-Q1-IaaS-Community-Analysis.pdf. [Accessed 05 Januray 2015]. |

**APPENDICES**