Ministry of Electronics and Information Technology

Digital India Internship Scheme

INTERNSHIP REPORT

Broad Area : eGovernance
Sub Area : Cloud Computing

INTERN
ABHISHEK CHHABRA

Under the Guidance of
Mr. Sanjay Goyal, Mrs. Uma Chauhan, Mr. Kshitij Kushagra

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance.I am extremely privileged to have got this all along the completion of my project.

I respect and thank Mr. Sanjay Goyal and Mrs. Uma Chauhan, for providing me an opportunity to do the project work in Ministry of Electronics and Information Technology, broad area eGovernance and sub area Cloud Computing.

I owe my deep gratitude to my project guide Mr. Kshitij Kushagra, who took keen interest on my project work and guided me all along, till the completion of my project work by providing all the necessary information for developing a good system.

TABLE OF CONTENTS

TASKS ASSIGNED

Following were the tasks assigned :-

1. A study on interoperability and portability in Cloud Computing.

2. Android Application Development for MeitY Canteen.

TASK 1

ABSTRACT

Evolution of processing power, storage technologies and availability of high quality broadband speed and big data have enabled the realization of cloud computing.Resources such as computing power, infrastructure, application platform are provided through the internet by cloud service providers.The existence of multiple cloud providers put the cloud service customer into vendor lock-in scenario.To avoid this interoperability and portability standards have to be established.This same situation exists in the Government of India's cloud initiative GI Cloud.In order to solve this problem a framework has been devised that categories the IaaS cloud offerings into levels of conceptual interoperability model.

INTRODUCTION

With proliferation of cloud computing and competition between Cloud Service Providers (CSPs) increase, the customers should have the freedom to switch between service providers for ensuring sectoral growth. The customer should not be locked on a single cloud provider. Every cloud service provider creates its own processes for a user or application interface with the cloud leading to cloud Application Programming Interface propagation.This leads to vendor lock-in, portability and inflexibility to use multiple vendors in the cloud including the inability to use an organization's own data centres resources seamlessly.This scenario is a major concern not just at indian level but at global level.

OBJECTIVE

For communicated information to be understood by the receiving system there is a need of interoperability. There is an need for consistent data handling and predictable performance across disparate cloud providers within a cloud ecosystem enabling hybrid cloud. For Indian system the problem is addressed with respect to interoperability, portability and vendor lock-in. The objective is to propose a framework to develop an ecosystem for interoperable cloud for Indian system.

SCOPE

Government of India has empanelled 13 Cloud Service Providers under the GI Cloud. As a result a potent problem of vendor lock-in has been identified for the use of cloud services within the government organizations. Project has been performed for avoiding vendor lock-in among the Cloud Service Providers in government organization. For this interoperability and portability characteristics have been studied.

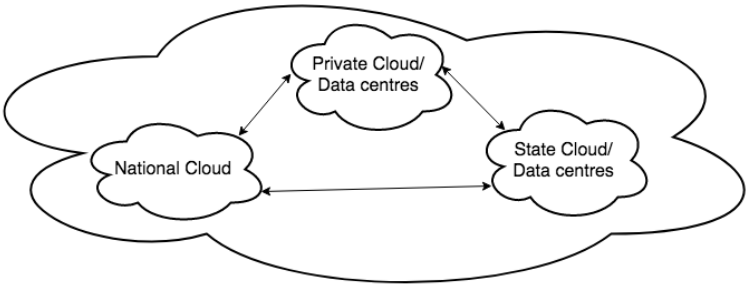

Figure 1 - GI Cloud Architecture

CHALLENGES IN CLOUD INTEROPERABILITY

In the context of cloud computing, interoperability is providing the ability for a cloud service customer system to interact with a cloud service and exchange information. It is moving of applications from one environment to another or run in multiple clouds. It involves software and data simultaneously active in more than one cloud infrastructure, interacting to serve a common purpose.

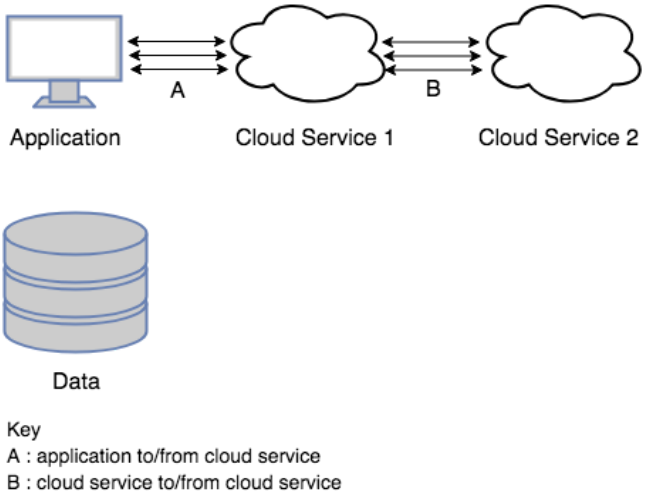

Figure 2 - High level view of cloud interoperability

Figure 2 indicates that cloud interoperability takes place between a Cloud Service Customer's application and cloud services and also takes place between cloud services.There are multiple interfaces involved in both of these cases, as indicated by multiple arrows.

1. Virtualization technology needs to be same for both the vendors or else virtual machine has to be converted to other format. Various Disk Formats are supported by the vendors, some of them are OVF, OVA, VMDK, raw. In such a case, performance level of the application and its interoperability with management, network and storage components needs to be addressed.
    a. On similar lines, due to lack of established industry standards within the cloud computing industry, public clouds are commonly propriety to various extents. For cloud consumers that have custom-built solutions with dependencies on these propriety environments, it can be challenging to move from one cloud provider to another.
2. Operating system versions and hypervisors that do match can produce multiple conflicts. Cloud providers define the relationship between servers and storage.
3. Every cloud has unique infrastructure for providing network services between servers and applications and servers and storage. Differences are likely in network addressing, directory services, firewalls, routers, switches, identity services, naming services and other resources. Target cloud are usually going to have a network architecture that differs from the source cloud network architecture.
4. The greatest level of interoperability is found in IaaS cloud services, where functionality is often broadly equivalent and there are number of standard interfaces such as CDMI. PaaS cloud services have lower levels of interoperability as there are few interface standards and some open source platforms where CSPs use same open source platform. SaaS cloud services represent the greatest level of interoperability challenge today as there are very few interface standards for SaaS applications - even switching from one SaaS application to another SaaS application with comparable functionality involves a change in interface.
5. Migrating an application across two different cloud environments means separating it from its original ecosystem; this may require re- engineering based on the components/parts that the target cloud provides. Data movement and encryption of data needs to be handled. Sequence

between source and target cloud result in a sequence of integration issues and requires rebuilding the application and application stack in target cloud.

6. During closure of business, cloud provider should be obligated to support porting of its customers to another provider. If cloud provider is unable to do so then a regulatory mechanism may be required to port customers to another provider. Similarly, in case the service ceases i.e. the cloud provider is prohibited from providing cloud service due to any legal issue or government decision, a regulatory mechanism is needed to deal with issues that emerge thus.

CHALLENGES IN CLOUD PORTABILITY

Portability is the ability to migrate applications between different clouds.

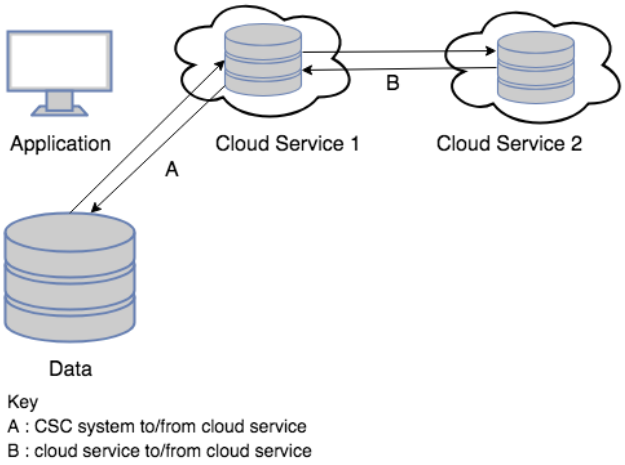

Figure 3 - High level view of cloud data portability

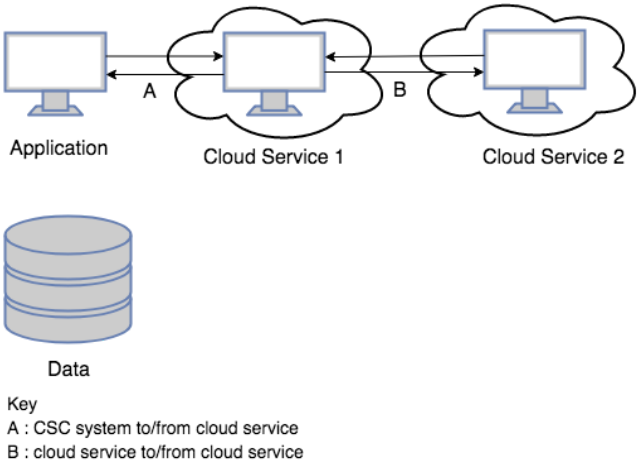

Figure 4 - High level view of cloud application portability

Figure 3 indicates porting of data between a CSC's system and a cloud service and porting of data from one cloud service to another.The arrows in both direction indicate the potential to port data to and from any of those places. Figure 4 indicates porting of application between a CSC's system and a cloud service and porting of application from one cloud service to another.The arrows in both direction indicate the potential to port applications to and from any of those places.

1. For system portability, an application description is required along with the description of the platform services. The interoperability requires secure translation of these semantics while porting from one cloud to another. These semantics need to be addressed across various interfaces involved in deployment of cloud. There can be four types of semantics[1] describing applications:
    a. System Semantics: Semantics pertaining to system characteristics like load balancing and deployment.
    b. Data Semantics: Semantics pertaining to data like manipulations restriction and storage.
    c. Non-Functional Semantics: Semantics pertaining to QoS characteristics like performance and security.
    d. Logic & Process Semantics: Semantics pertaining to core functions of the application like programming language, runtime, and exception handling.
2. For functional portability, the ability to define application functionality QoS details in platform-agnostic manner. Portability of a virtual machine images is being addressed by the DMTF Open Virtualization Format (OVF) [4].It provides a good basis for limited portability but does not address complex configurations or interactions with any supporting system.

3. For data portability, the ability for a customer to retrieve application data retrieve application data from one provider and import this into an equivalent application hosted by another provider. It is necessary to provide a platform-independent data representation.

VENDOR LOCK-IN

Vendor lock-in problem in cloud computing is the situation where customers are dependent on a single cloud provider technology implementation and cannot easily move in the future to a different vendor without substantial costs, legal constraints or technical incompatibilities[5].The lock-in situation is evident in the applications developed by specific cloud platforms, cannot be migrated to other cloud platforms and users become vulnerable to any changes made by their providers.

Interoperability and portability are essential qualities that affect the cloud under different perspectives. For many companies there is a large amount of sensitive data and IT assets in-house which can deter them to migrate to the cloud due to risks of vendor lock-in, security and privacy issues. Integration, interoperability and portability between software on-premise and in the cloud has to be taken into account.

WORKING

An attempt to create a framework by applying the Level of Conceptual Interoperability Model[13] to the cloud services offered by CSPs has been made. Our study design consists of distinct phases as depicted in Figure 5.

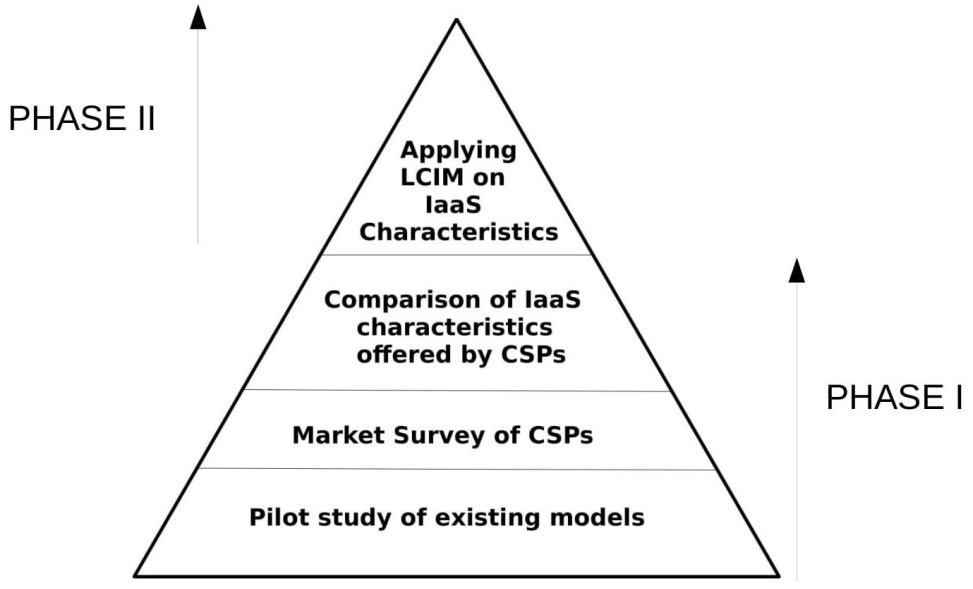

Figure 5 - Explanatory Design

PHASE I

Phase I is a pilot study of the existing models followed by a market survey. After doing this a comparison of IaaS level offerings by Cloud Service Providers is formulated. The data is gathered and analyzed from the technical specifications, white papers and a thorough research of the cloud product offered by various CSPs.

PILOT STUDY OF EXISTING MODELS

ISO/IEC 19941:2017[6] specifies three models for the context of cloud interoperability and portability.
Table 1 describes cloud interoperability facet model.It defines five facets within the context of cloud interoperability being transport, syntactic, semantic data, behavioral, policy.This model is derived by combining and abstracting the European Interoperability Framework[12] and the Levels of Conceptual Interoperability Model (LCIM)[13].
Table 2 describes cloud data portability facet model.It defines three facets of cloud data portability within the context of cloud computing being data syntactic, data semantic and data policy.This model is based on the cloud interoperability facet model.Data portability relates to the capability of moving data in and out of the cloud service environment.
Tabel 3 describes cloud application portability facet model.It defines five facets of cloud application portability being instruction, syntactic, metadata, behavior and policy.The cloud application portability facet model is inspired by the cloud interoperability facet model.Application portability involves Virtual Machine (VM) Images Migration.It relates to the capability of moving the App code to or from the cloud service.

Facet	**Aim**	**Objects**	**Solutions**
Transport	Data transfer	Signals	Protocols of data transfer, e.g. REST over HTTP;MQTT
Syntactic	Understand format of transferred data	Data	Standardized data exchange formats, e.g XML
Semantic Data	Interpretation of transferred data using a data model	Information	Common data models, e.g OData, OWL
Behavioral	Get anticipated outcomes when making service requests	Programmatic interface	UML models, pre-conditions, post-conditions, constraint specifications
Policy	Ensure that interacting systems conform to applicable laws, regulations and organizational policies	Laws, regulations, policies	Conditions for operation

Table 1 - 5 facet model for cloud interoperability

Facet	Aim	Objects	Examples
Syntactic	Receive data in a readable structured format	Data	JSON, XML
Semantic	Understand the meaning of ported data	Information	OWL
Policy	Meet applicable laws, regulations and policies	Laws, regulations, policies	Personal data regulations, Cross border data transfer laws, Security policies

Table 2 - 3 facet model for data portability

Facet	Aim	Objects	Examples
Instruction	Execute application instructions correctly	Executable artifacts	Java, C++, BPEL
Syntactic	Understand and use format of application artifacts	All application artifacts	Zip, tar, jar
Metadata	Understand and use the metadata that specifies environmental dependencies for executing the application	Metadata artifacts	YAML, JSON,Script, XML
Behaviour	Produce the expected results when executing the application	Application functional and nonfunctional behaviors	Verified by test studies
Policy	Meet applicable laws, regulations and policies	Laws, regulations, policies	Personal data regulations, Cross border data transfer laws, Security policies

Table 3 - 5 facet model for cloud application portability

Cloud Computing Interoperability Forum (CCIF) is planning to come up with a unified cloud interface[7].Unified cloud interface performs unification of various cloud APIs and abstract it behind an open standardized cloud interface.Key driver of unified cloud interface is to create an API about other API.It acts as an singular abstraction/programmatic point of contact that encompasses the entire infrastructure as well as emerging cloud centric technologies through a unified interface.
The model suggests the use of semantic web and OWL.Purpose of unified cloud interface is to serve as a common interface for interaction between remote platforms, networks, systems, applications, services, identity and data.

Enterprise Cloud Orchestration Platform[8], different cloud services providers can register cloud services that the offer with orchestration layer.The layer can then dynamically select and bind to services based on criteria algorithm that determine the best cloud service for a particular job based on factors like highest performance, lowest cost.It uses User-Computer Interaction for interacting with different cloud service providers or have similar functionality built-in to be able to understand and interact with different CSPs via different APIs.

mOSAIC[9], consists of building an open source API offering layer of abstraction, and applying semantic to interface and component level.The application can be separated into application logic-layer and cloud layer.A descriptor of the application is used to select the proper clouds.The applications are decomposed in components runnable of top of different clouds.Exchanges between clouds are done using cloud based message queue technologies.mOSAIC gives a set of APIs that enables the developers to build vendor independent applications that consist of multiple cloud components.

Approach	**Focus**	**Key Operations**
ISO/IEC 19941:2017	Exchange of information between connecting systems.	Categorises interoperability and portability characteristics into various facets.
Cloud Computing Interoperability Forum (CCIF)	Standardisation approach that proposes a common management API.	API for common management tasks e.g.cloud monitoring, deployment
Enterprise Cloud Orchestration Platform	Uses user-computer interaction to interact with different CSPs.	Selects the best cloud service for particular job.
mOSAIC	mOSAIC API serves as an intermediate layer between developers and cloud platforms.	On run-time, applications are decomposed into components and the platform decides which cloud implementation is better for each component.

Table 4 - Interpretation from Existing Models

MARKET SURVEY

In order to get insights into the problem and to identify and analyze the Cloud Service Providers, market survey was conducted. Data was gathered in the form of technical specifications from CSPs through an Email request. The outcome of the survey is a capability matrix.

COMPARISON OF IaaS CHARACTERISTICS

Service models of cloud computing consists of Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [15].The difference between them is in the complexity of provisioned, and how customers can use or access them.
IaaS is the capability provided to the cloud consumer is to provision processing, storage, networking, and other fundamental computing resources, where the consumer is able to deploy and run arbitrary software include operating systems and applications.IaaS is marked through the provision of basic computing capabilities for customers. In PaaS, users have control only of their applications, and to a certain extend the configuration of hosting environments. In the SaaS model, customers just use a certain application running on a cloud infrastructure [16].
Our focus is on IaaS level interoperability, Teckelmann and Reich [11, 16, 17] have presented a taxonomy of interoperability for IaaS clouds. The taxonomy discusses all important issues aiming for IaaS interoperability. Each of the IaaS taxonomy is further parametrized using Notable Service Traits [10].
The parameters explored are Storage, VM resources, Billing, Network, Security, Access Mechanism, SLA, Computing Power, Database Management.

PHASE II

Phase II is an attempt to fit the IaaS characteristics of 3 CSPs in Level of Conceptual Interoperability Model has been made. Thus the result quantified is based on the identification of technology stack used by the CSPs.

Level	Description
Level 0 - No interoperability	No interoperability
Level 1 – Technical	Communication protocol exists at the bits and bytes level
Level 2 – Syntactic	Uses a common structure, such as the data format
Level 3 – Semantic	Uses a common information exchange reference model, such as word meanings are the same in each system
Level 4 – Pragmatic	All interoperating systems are aware of each other's methods and procedures for using the data
Level 5 – Dynamic	State changes (including assumptions and constraints) in one system are comprehended by all interoperating systems
Level 6 – Conceptual	A shared meaningful abstraction of reality is achieved

Table 7 - LCIM's Seven Levels of Interoperability

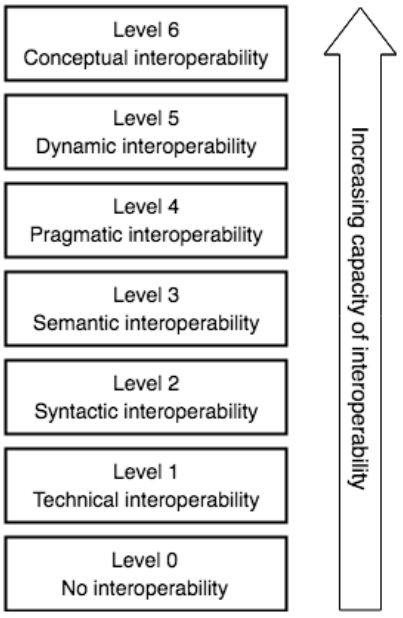

Figure 5 - The Levels of Conceptual Interoperability Model

Tolk and Muguira [14] originally proposed LCIM. The model used is the evolution of LCIM [13] as depicted in Figure 5. Table 7 describes seven layers from 'no interoperability' to 'conceptual interoperability'. Prescriptive role [13] of LCIM model is used as the sole purpose of applying the model is to act as an interoperability guidance model to prescribe and guide the interoperability of IaaS characteristics offered by CSPs.

A framework was is devised by mapping the IaaS characteristics offered by the CSPs onto LCIM. The characteristics fall into 'no interoperability', 'technical interoperability' and 'syntactic interoperability'.

CONCLUSION

To avoid the problem of vendor lock-in interoperability and portability levels among cloud providers were examined.It was determined from pilot study of existing models that the greatest level of interoperability is found for IaaS cloud services.PaaS cloud services have lower levels of interoperability and the SaaS applications present the greatest interoperability challenge today. As the IaaS cloud services show a greater level of interoperability a framework on IaaS characteristics was developed.

FUTURE SCOPE

Using the framework created two prominent findings can be established.First, rating of Cloud Service Providers based on the level of interoperability they demonstrate.For this a questionnaire will be prepared for various government organisations and depending on the answers a critical analysis will be carried to aid the rating of Cloud Service Providers.Second, creation of Interoperability Test Bed for cloud services provided by Government of India.

REFERENCES

[1] A Survey on Approaches for Interoperability and Portability of Cloud Computing Services", Kostas Stravoskoufoset al.

[2] Consultation Paper on Cloud Computing, TRAI

[3] Oberle, K., Fisher, M.: ETSI CLOUD–initial standardization requirements for cloud services. In: Altmann, J., Rana, O.F. (eds.) GECON 2010. LNCS, vol. 6296,pp. 105–115. Springer, Heidelberg (2010)

[4] Open Virtualization Format Specification, DMTF Document Number DSP0243 (2010)

[5] Justice Opara-Martins, Reza Sahandi and Feng Tian, "Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective", Journal of Cloud Computing: Advances, Systems and Application, DOI 10.1186/s13677-016-0054-z

[6] ISO/IEC 19941:2017 https://www.iso.org/obp/ui/#iso:std:66639:en

[7] Unified Cloud Interface project https://code.google.com/archive/p/unifiedcloud/

[8] Infosys Research on Orchestration layer approach.

[9] Dana Petcu, *"Portability and Interoperability between Clouds: Challenges and Case Study"*

[10] Lydia Leong, "Critical Capabilities for Public Cloud Infrastructure as a Service, Worldwide".

[11] Magdalena Kostoska, Marjan Gusev, Sasko Ristov,*"An Overview of Cloud Interoperability"*, Proceedings of the Federated Conference on Computer Science and Information Systems.

[12] European Interoperability Framework https://ec.europa.eu/isa2/sites/isa/files/eif_brochure_final.pdf

[13] Wenguang WANG,Andreas TOLK, Weiping WANG, "The Levels of Conceptual Interoperability Model: Applying Systems Engineering Principles to M&S", Spring Simulation Multiconference (SpringSim'09). San Diego, CA, USA.

[14] Tolk A, Muguira JA. "The levels of conceptual interoperability model". Fall Simulation Interoperability Workshop. Orlando, Florida: Simulation Interoperability Standards Organization, 2003

[15] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," National Institute of Standards & Technology, Tech. Rep., Oct. 7, 2009.

[16] R. Teckelmann, C. Reich, and A. Sulistio, "Mapping of Cloud Standards to the Taxonomy of Interoperability in IaaS," Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference.

[17] R. Prodan and S. Ostermann, "A survey and taxonomy of infrastructure as a service and web hosting cloud providers," Grid Computing, 2009 10th IEEE/ACM International Conference.

TASK 2

OBJECTIVE

An Information System application is created for facilitating the MeitY canteen. The end users can come to know what dishes are available in the canteen through the use of a SmartPhone.

USE CASE DIAGRAM

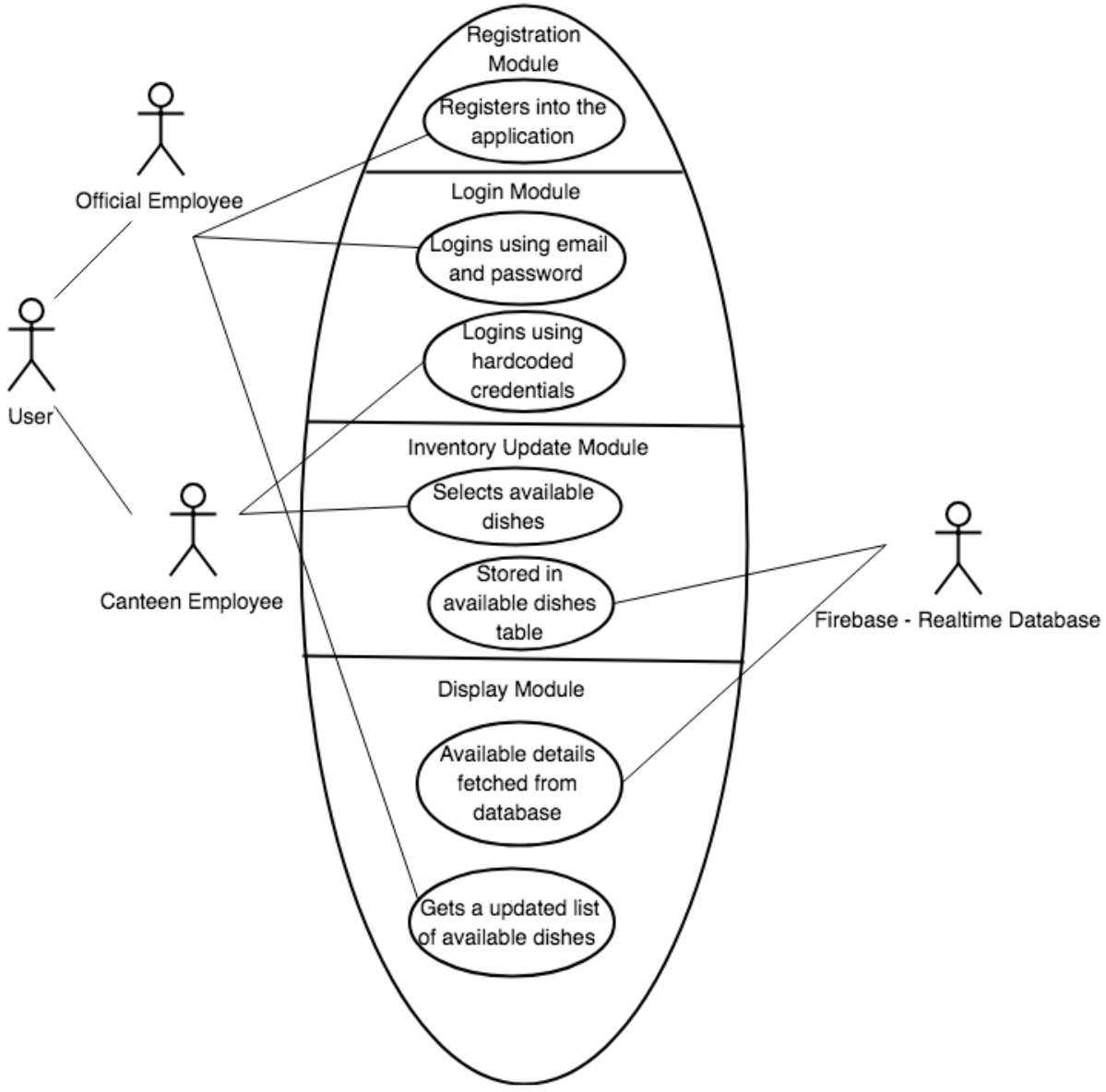

Figure 6 - Use Case Diagram

TECHNOLOGY STACK

Android Studio - Android Application Development
Firebase - Email Authentication, Realtime Database

OUTPUT

SCREENSHOTS OF THE APPLICATION

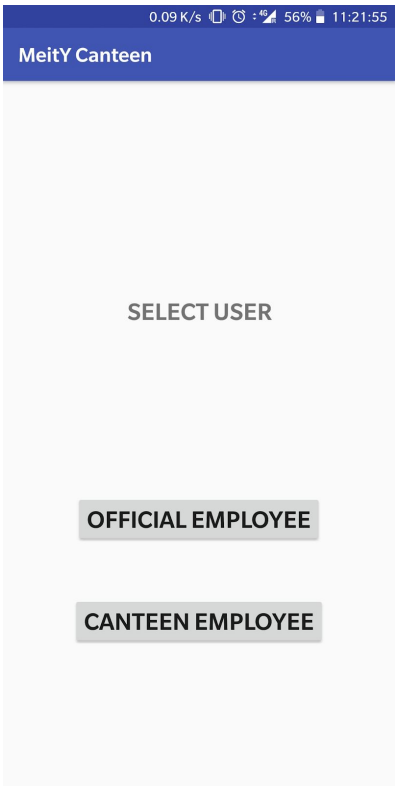

Figure 7 - Initial Screen

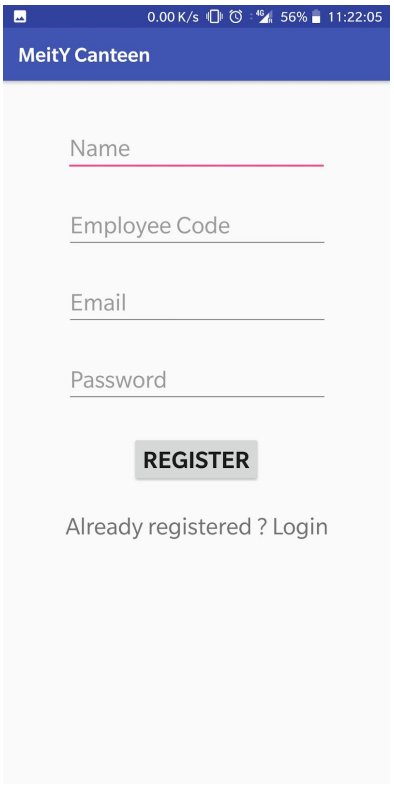

Figure 8 - Official Employee Register Screen

Figure 9 - Canteen Employee Login Screen

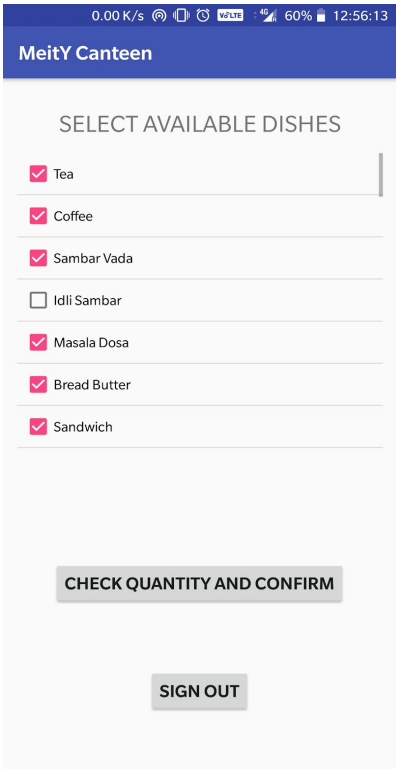

Figure 10 - Canteen Emp Inventory Update Screen

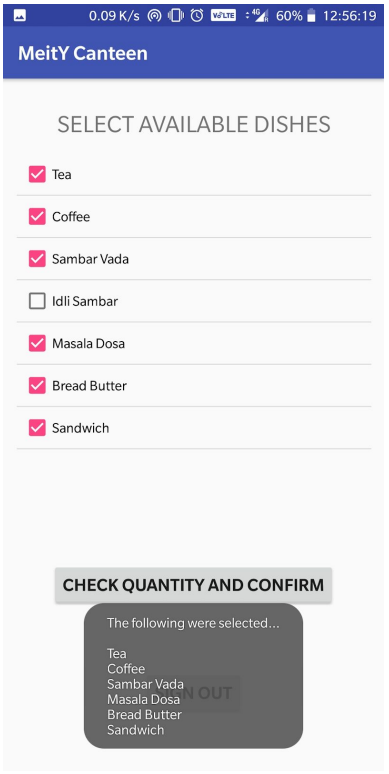

Figure 11 - Inventory Confirmation

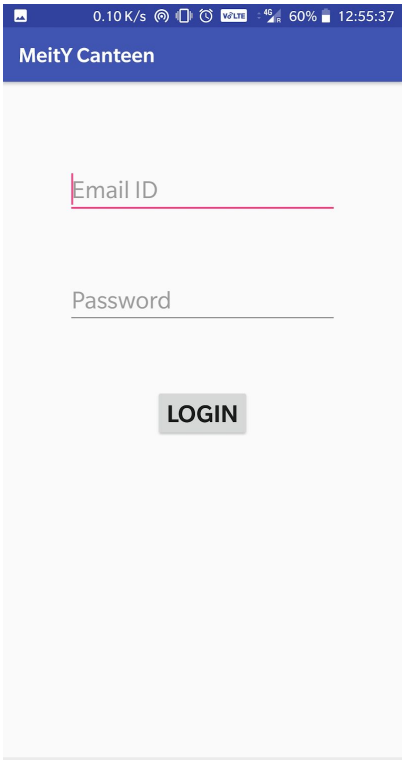

Figure 12 - Off Emp Login Screen

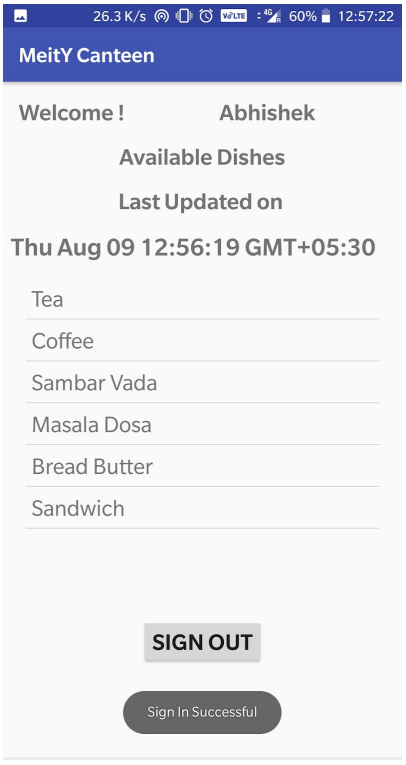

Figure 13 - Available Dishes Screen

SCREENSHOTS OF THE FIREBASE CONSOLE

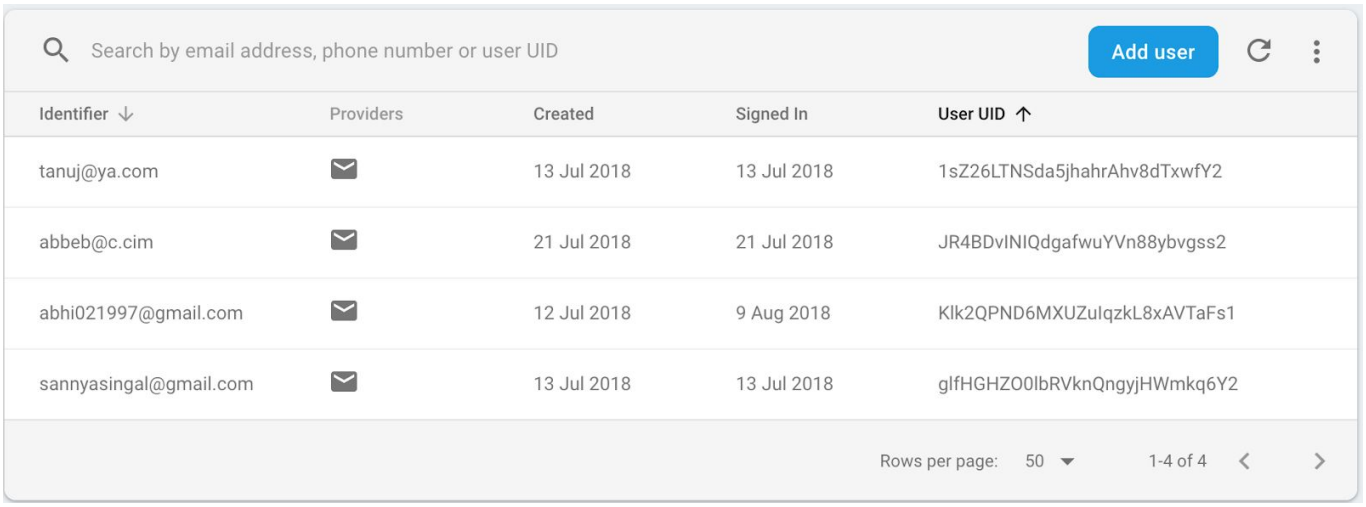

Figure 14 - Authentication Tab Firebase

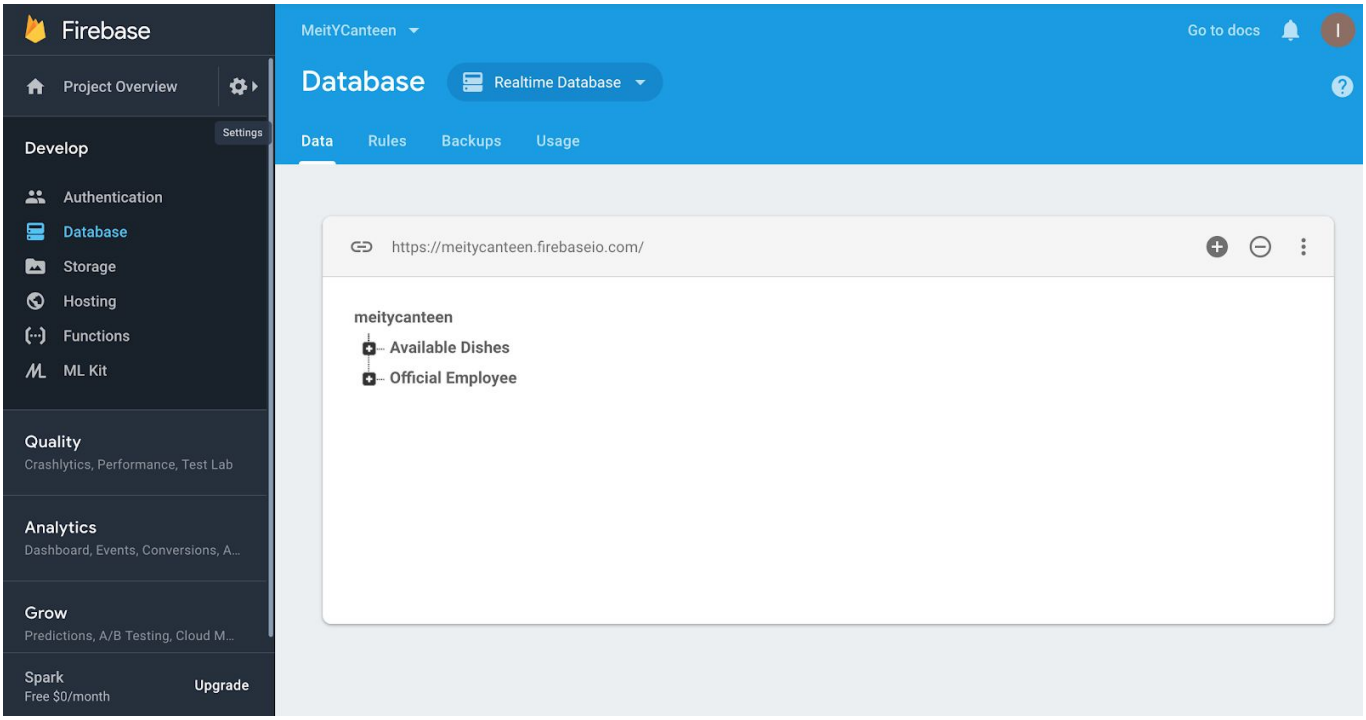

Figure 15 - Overview of Realtime Database

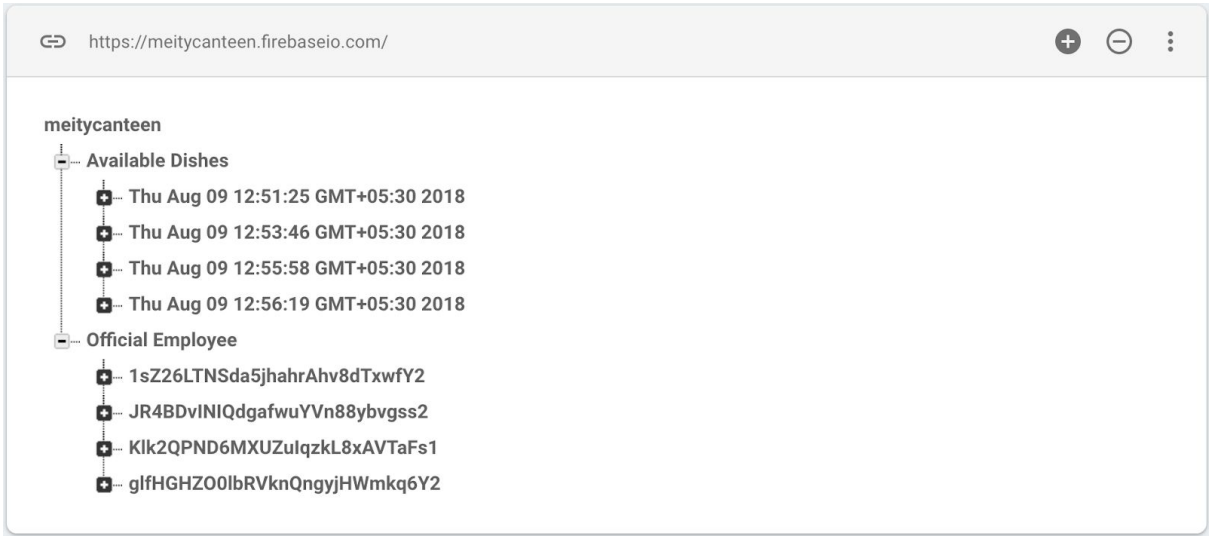

Figure 16 - Current Status of Realtime Database

Figure 17 - Available Dishes, Updated by Canteen Employee

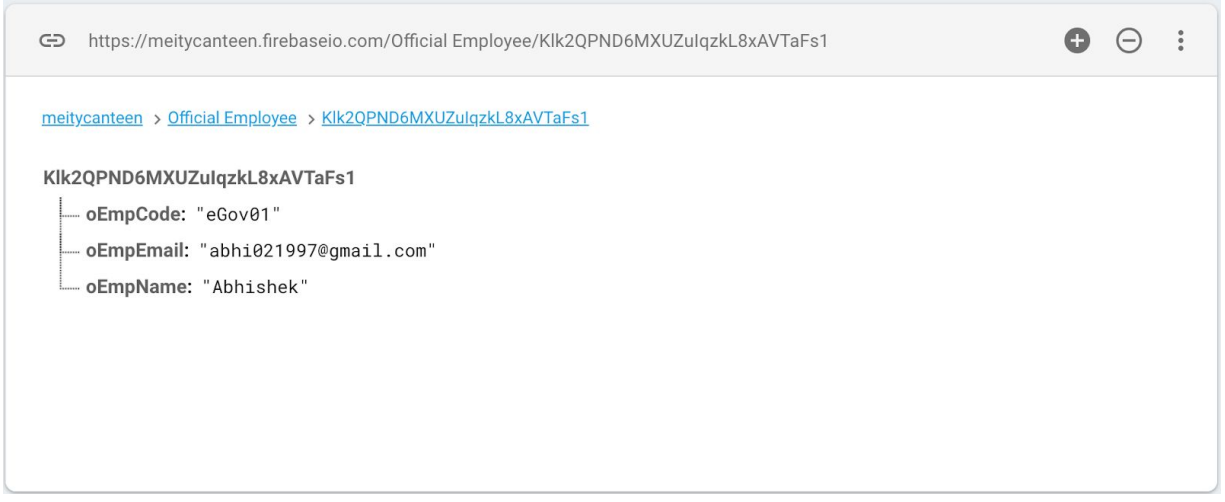

Figure 18 - Official Employee Details, Single user

FUTURE SCOPE

The application can be made for full fledged order placement. This can be achieved by Integrating BHIM Interface using Deep Linking UPI provided by National Payments Corporation of India.

REFERENCES

[1] Deep Linking UPI
https://blog.deazzle.in/enable-upi-payments-in-your-app-without-the-need-to-integrate-with-a-bank-c911019f3b2d

[2] List View https://developer.android.com/guide/topics/ui/layout/listview

[3] Firebase for Android https://firebase.google.com/docs/android/setup

[4] Constraint Layout https://developer.android.com/training/constraint-layout/