**Integrity as a Service**

**BITS ZG628T: Dissertation**

by

SREE RAM GANTA

2016HT12642

**Dissertation work carried out at**

**TATA CONSULTANCY SERVICES, HYDERABAD**



**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE**

**PILANI (RAJASTHAN)**

April 2018

**Integrity as a Service**

### BITS ZG628T: Dissertation

by

SREE RAM GANTA

2016HT12642

# Dissertation work carried out at

**Tata Consultancy Services Limited, Hyderabad**

Submitted in partial fulfillment of M.Tech. Software Systems degree programme

Under the Supervision of

Srihari Rao P, Assistant Consultant

Tata Consultancy Services Limited, Hyderabad

****

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE**

**PILANI (RAJASTHAN)**

April 2018

#### CERTIFICATE

This is to certify that the Dissertation entitled Integrity as a Service and submitted by Sreeram Ganta having ID-No: 2016HT12642 for the partial fulfillment of the requirements of M.Tech. Software Systems degree of BITS, embodies the bonafide work done by him/her under my supervision.

 Signature of the Supervisor

Place: Hyderabad Srihari Rao. P, Assistant Consultant   
Date: March 16, 2018 Tata Consultancy Services, Hyderabad   
 **Name, Designation & Organization &Location**

**Birla Institute of Technology & Science, Pilani**

**Work-Integrated Learning Programmes Division**

**Second Semester 2017-2018**

**BITS ZG628T: Dissertation**

**ABSTRACT**

**BITS ID No. :** 2016HT12642

**NAME OF THE STUDENT :** SREERAM GANTA

**EMAIL ADDRESS :** <2016HT12642@wilp.bits-pilani.ac.in>

**STUDENT’S EMPLOYING :** TATA Consultancy Services Limited,

**ORGANIZATION & LOCATION** Hyderabad

**SUPERVISOR’S NAME :** SRIHARI RAO PINNINTI

**SUPERVISOR’S EMPLOYING :** TATA Consultancy Services Limited,

**ORGANIZATION & LOCATION** Hyderabad

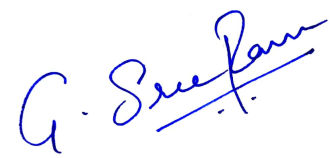
**SUPERVISOR’S EMAIL ADDRESS:** [sriharirao.p@tcs.com](mailto:sriharirao.p@tcs.com)

**DISSERTATION TITLE :** INTEGRITY AS A SERVICE

**ABSTRACT :** With the help of Blockchain, one can create an Integrity between two untrusted parties to exchange goods/products, which is the main objective of this project. Project make use of unique barcode/QR-code of the product to represent certain block of the encrypted Blockchain. Service provided will decrypt the block into readable format to the consumer/customer.

**Broad Academic Area of Work:** Software Engineering & Management

**Key words:** Blockchain, Docker, postgres, pycam, python, ubuntu, zbar

**Signature of the Student Signature of the Supervisor**

**Name:** SREERAM GANTA **Name:** SRIHARI RAO P

**Date:** March 16, 2018 **Date:** March 16, 2018

**Place:** Hyderabad  **Place:** Hyderabad

**ACKNOWLEDGEMENTS**

I would like to thank my Supervisor Sri Hari Rao for guiding me in design process and suggesting the right tools to use for design.

I would also like to thank my team member Naveen Surisetty for suggesting the architecture and giving the constructive feedback, which helped to progress this project.

Thanking my Examiner Naveen Kumar Maremanda for his encouragement for this project.

* Sreeram Ganta

Contents

[1. Introduction 1](#_Toc509224074)

[2. Approach to Requirements Analysis 1](#_Toc509224075)

[3. Technologies 2](#_Toc509224076)

[4. About Hyperledger Sawtooth 2](#_Toc509224077)

[5. Project Analysis 3](#_Toc509224078)

[6. Design of Blockchain using Barcode 4](#_Toc509224079)

[7. Pre Configuration 7](#_Toc509224080)

[8. Creation of Blockchain using Barcode 8](#_Toc509224081)

[9. Summary 13](#_Toc509224082)

[10. Direction for future work 13](#_Toc509224083)

[11. References 13](#_Toc509224084)

[12. Checklist of Items 14](#_Toc509224085)

Table of Figures

[Figure 1: Reading Barcode using webcam 6](#_Toc509178766)

[Figure 2: Create block chain 9](#_Toc509178767)

Table of Tables

[Table 1: Barcode\_cli help screen 5](#_Toc509178499)

[Table 2: Setup 7](#_Toc509178500)

[Table 3: Add user 8](#_Toc509178501)

[Table 4: Listing product information in postgres DB 8](#_Toc509178502)

[Table 5: Create block chain 9](#_Toc509178503)

[Table 6: List of blocks 10](#_Toc509178504)

[Table 7: List of states 10](#_Toc509178505)

[Table 8: Show chain 11](#_Toc509178506)

[Table 9: Update block chain 11](#_Toc509178507)

[Table 10: Show Updated block chain 12](#_Toc509178508)

## Introduction

**1.1** **Blockchain**

Blockchain is a continuous layered blocks, which are linked and secured by cryptography. Mostly heard cryptocurrency, Bitcoin’s underlying technology is Blockchain.  It is "an open [distributed ledger](https://en.wikipedia.org/wiki/Distributed_ledger) that can record transactions between two parties efficiently and in a verifiable and permanent way".

Linux foundation started business Blockchain frameworks hosted with Hyperledger like Fabric, Burrow, and Sawtooth etc. All of them provide an interface to create cryptographic ledger(Blockchain) with variety of consensus algorithms for different applications.

* 1. **Purpose of the document**

This document explains about the software, which creates an Integrity between two untrusted parties to exchange goods/products. Software provides below possibilities to the user and detailed more in further chapters.

**Migrate Product details to Blockchain:** Manufacturer/Owner will have scope to migrate all the product information into Blockchain.

**Manage product details in Blockchain:** User can create, update and read the product details into/from the block chain.

**User Management:** Owner will have scope to manage users i.e., Owner can add admin or supplier type of users into blockchain.

Software designed to use command line interface using python language. It uses Hyperledger Sawtooth framework as a core base.

## Approach to Requirements Analysis

Software will make use of unique barcode of the product to represent certain block of the encrypted Blockchain. Service provided will decrypt the block into readable format to the consumer/customer. The approach to requirements analysis is as follows.

* + To create a Blockchain there are multiple frameworks designed by Linux Foundation community, Sawtooth is one of them. Sawtooth is applicable to this requirement as it is concentrated more in supply chain management.
  + To any product owner there shall have a possibility to migrate details to Blockchain, which indicates software should be possibility to read data from owner’s database using unique barcode of the product.
  + In addition, there shall be possibility to the retailers to update the Blockchain details in owner’s server.

## Technologies

From the approach discussed in previous chapter, below technologies chosen to design the software.

* + Ubuntu 16.04 will be the working environment due to its flexibility and open source characteristics.
  + As mentioned earlier base API used is Hyperledger Sawtooth framework.
  + Since the core base is in python language, python confirmed as coding language.
  + To read the barcode in python using webcam, modules like zbar and pygame are used.
  + Postgres DB will act as DB for product details.
  + To access data from postgres DB, psycopg2-binary module is used.
  + PyCharm IDE used to design.
  + GIT as code repository

## About Hyperledger Sawtooth

Hyperledger Sawtooth is an enterprise blockchain platform for building distributed ledger applications and networks. The design philosophy targets keeping ledgers *distributed* and making smart contracts *safe*, particularly for enterprise use.

Sawtooth simplifies blockchain application development by separating the core system from the application domain. Application developers can specify the business rules appropriate for their application, using the language of their choice, without needing to know the underlying design of the core system.

Sawtooth is also highly modular. This modularity enables enterprises and consortia to make policy decisions that they are best equipped to make. Sawtooth’s core design allows applications to choose the transaction rules, permissioning, and consensus algorithms that support their unique business needs.

* 1. **About Distributed Ledgers**

A “distributed ledger” is another term for a blockchain. It distributes a database (a ledger) of transactions to all participants in a network (also called “peers” or “nodes”). There is no central administrator or centralized data storage. In essence, it is:

* + **Distributed:** The blockchain database is shared among potentially untrusted participants and is demonstrably identical on all nodes in the network. All participants have the same information.
  + **Immutable:** The blockchain database is an unalterable history of all transactions that uses block hashes to make it easy to detect and prevent attempts to alter the history.
  + **Secure:** All changes are performed by transactions that are signed by known identities.

These features work together, along with agreed-upon consensus mechanisms, to provide “adversarial trust” among all participants in a blockchain network.

* 1. **Transaction Families**

In Sawtooth, the data model and transaction language is implemented in a *transaction family*. It describes the validation and way to store the details in block chain.

## Project Analysis

* 1. **Business Requirement**

The requirement is to provide any manufacturer of any product to migrate their product information to blockchain to increase the integrity with customers.

This can be a value add for the customer.

* 1. **Scope of Project**

Hyperledger Sawtooth framework as a base, a service is designed to provide an Integrity between two untrusted parties seller and consumer. Below pointers, provide the scope of the project.

* + Software will allow manufacturer to add the details of the product into the blockchain, by reading the barcode of the product.
  + Multiple sellers/dealers can modify same product data.
  + Modified data/Transaction history of a product shall also have maintained by the service.

## Design of Blockchain using Barcode

Software designed has two modules.

1. Barcode - CLI
2. Barcode- Transaction Processor
   1. **Barcode Command line Interface**

This module is the interface provided to the user to perform creating, updating and reading operations on block chains with the given barcode. This module in bundled to an executable named ‘barcode\_cli’.

Two scripts where designed as part of this module

* barcode\_cli.py
* barcode\_reader.py

**barcode\_cli.py:** All actions to be performed are designed in this python file.

barcode\_cli is responsible for

* + - creating the admin setup
    - adding admin/supplier users
    - creating block chain
    - show block chain
    - updating block chain

Below is the help screen of the same. This module in bundled to an executable named ‘barcode\_cli’. Detailed explanation for each responsibility of barcode\_cli.py done in later chapters.

root@ubuntu: ~# barcode\_cli -help

BarCodeReaderCli.

Usage:

barcode\_cli.py setup

barcode\_cli.py add (supplier|admin) <name> [-k <keypath> | --keypath <keypath>]

barcode\_cli.py create chain (-u <user> | --username <user>) [-b <barcode> | --barcode

<barcode>]

barcode\_cli.py show chain (-u <user> | --username <user>) [-b <barcode> | --barcode <barcode>]

barcode\_cli.py update chain (-u <user> | --username <user>) (-l <location> | --location <location>) [-b <barcode> | --barcode <barcode>]

barcode\_cli.py (-h | --help)

barcode\_cli.py --version

Options:

-h --help Show this screen.

-u --username username

-l --location updating location

-b –barcode input barcode through cli

--version display version

Table : Barcode\_cli help screen

**barcode\_reader.py:** This script is designed to read the barcode from the webcam. Called from barcode\_cli.py whenever barcode is taken as an input. Pygame module is used to read the images from webcam, to read the actual barcode from the image zbar module is used. Refer the below demo screen shot.

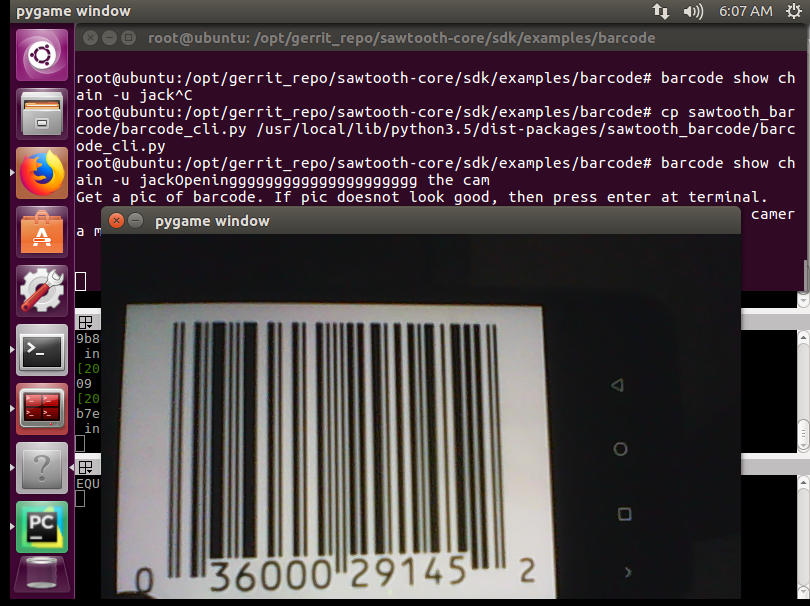


Figure : Reading Barcode using webcam

* 1. **Barcode Transaction Processor**

Transaction processor plays the key role in storing the product details into the block chain, it is specifically designed to create separate transaction family ‘barcode’.

BarcodeTransactionProcessor class in barcode\_processor.py is inherited from the TransactionProcessor class from the Sawtooth framework.

Block chain is updated with respective to the inputs given from the user.

This module in bundled to an executable named ‘sawtooth-barcode’

## Pre Configuration

* 1. **Sawtooth Setup**

Sawtooth installation required to perform prior to barcode cli setup. It includes below sequence of steps. Detailed installation steps were discussed in sawtooth wiki page [2].

1. Install sawtooth binaries
2. Start validator – responsible for validating and creating blockchain
3. Start rest api – responsible to accept rest calls from the users
4. Start settings transaction processor – has the detailed information about transaction families
   1. **Install Barcode Binaries**

Barcode python module can be installed using below command, where <path> is path where software is downloaded

# python3 <path>/setup.py build install

After installation barcode\_cli and barcode\_tp binaries will be available.

To start the barcode\_tp processor run below command in separate terminal.

# sudo -u sawtooth barcode\_tp –v

* 1. **Barcode cli Setup**

As part of setup admin user is created, the respective admin private key is added to the blockchain. Admin user key is used as signer to all the user’s information in the blockchain.

# barcode\_cli setup

|  |
| --- |
| root@ubuntu:~# barcode\_cli setup  writing file: /root/.sawtooth/keys/admin.priv  writing file: /root/.sawtooth/keys/admin.pub  Response: {  "link": http://127.0.0.1:8008/batch\_statuses?id=c420f579017a5a0eecf8d5021d089b3ff9594706b66f8ee4f7fe643b4a250ee95a8fed2ad5e276bec5b21f9b350ba80708cf8046315d6cc4ed314e29ebffd703  }  root@ubuntu:~# |

Table : Setup

* 1. **Add user**

To create/start any chain in blockchain DB user details should be present in blockchain DB. Two kind of users can be added

admin – Has permissions to create, read and update blockchain information.

supplier – Has permission to only update the blockchain information that too only location field of the product.

Run below command to add user information into blockchain where tag is either admin/supplier and <user> is username.

# barcode\_cli add <tag><user>

|  |
| --- |
| root@ubuntu:~# barcode\_cli add admin user1  writing file: /root/.sawtooth/keys/user1.priv  writing file: /root/.sawtooth/keys/user2.pub  Response: {  "link": http://127.0.0.1:8008/batch\_statuses?id=c420f579017a5a0eecf8d5021d089b3ff9594706b66f8ee4f7fe643b4a250ee95a8fed2ad5e276bec5b21f9b350ba80708cf8046315d6cc4ed314e29ebffd703  }  root@ubuntu:~# |

Table : Add user

## Creation of Blockchain using Barcode

* 1. **Prerequisite**

As part of pre-requisite product details are added in postgres DB with barcode.

Separate user and database name were used for postgres DB like barcode-user and barcode DB respectively.

root@ubuntu:~# sudo -i -u barcode\_user psql -d barcode

Password:

psql (9.5.11)

Type "help" for help.

barcode=# select \* from barcode\_details;

barcode\_id | name | mfg\_date | location

--------------+-------+---------------------+----------

36000291452 | maagi | 2014-04-28 12:30:00 | Delhi

671860013624 | kikat | 2016-04-29 12:30:00 | Gurgaon

(2 rows)

barcode=#

Table Listing product information in postgres DB

* 1. **Create Chain**

1. To create chain run below command, which will open webcam to scan the barcode from the user, where <username> is admin type username.

Only admin type users have permission to create chain in the block chain. Command fails if the supplier type user is given as input.

#barcode\_cli create chain –u <username>

**Note:** Use –b option to pass barcode to the command.

#barcode\_cli create chain –u <username> -b <barcode\_id>

root@ubuntu: ~# barcode\_cli create chain -u jack

Get a pic of barcode. If pic does not look good, then press enter at terminal.

Camera will take another pic. When done press q and enter to quit camera mode

q

INFO: Barcode read: 671860013624

Validation Successful

Response: {

"link": "http://127.0.0.1:8008/batch\_statuses?id=98965b14748f185726a15a8bea8c46c954acaeb678b6d554ac9ffc3dfd763d585a5d253df6da15b1daf576c7f1be37f8ed0039e50ffcf921f89ea06a430cfbf8"

}

Table : Create block chain

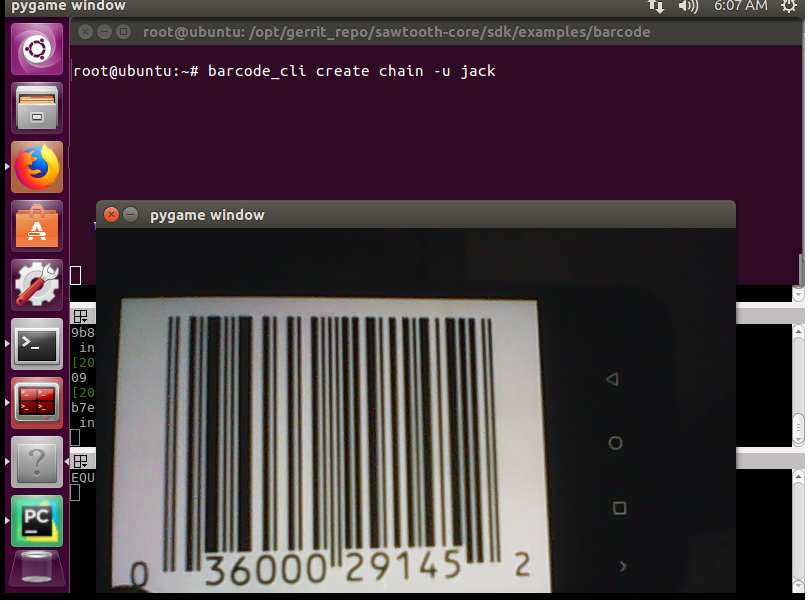


Figure : Create block chain

1. To list the created blockchain use the below command

# sawtooth block list

root@ubuntu:~# sawtooth block list

NUM BLOCK\_ID BATS TXNS SIGNER

3 d9591700aab11c6e7e089d5a0990b93083474964ba16b5ca5e822e9f4821d1ef3aa6583a01b0d98d2ff978705ed52fe4c8be891b01a775f18124399faebd3778 1 1 0329cb970786dd...

2 6b0e43c9e68c039499c4829e5c9cba2b4650bcb64e5b1f139137ed4a43cb9a6a33f3f66ab2754b05059ffb1a8b57f3ecf6a50c21317a6f46240816497fa90950 1 1 0329cb970786dd...

1 f0d75e4cfaaf8e5f21a2a90cf226bc6acd1b81885e2cc19c437861182098a70f5f019db50a561eb0b4a9dc20ed419c2e78d2997f839836e40c15cf308f74b4c8 1 1 0329cb970786dd...

0 2789efc2e231eab0a3ddb31e6c0cfee21bee657ffd9c8e65a61c85268dde8a856286358cca81300af853b4485ad9ee02277284675a0866ef157d402990998f42 1 1 0329cb970786dd...

root@ubuntu:~#

Table : List of blocks

1. To list the states of blocks use below command

# sawtooth state list

root@ubuntu:~# sawtooth state list

ADDRESS SIZE DATA

000000a87cb5eafdcca6a8cde0fb0dec1400c5ab274474a6aa82c12840f169a04216b7 110 b'\nl\n&sawtooth.settings.vote.authorized\_keys\x12B02101354de52cf04d0f8b24033b93d37...

000000a87cb5eafdcca6a8f82af32160bc531176b5001cb05e10bce3b0c44298fc1c14 86 b'\nT\n\'sawtooth.validator.transaction\_families\x12)[{"family": "barcode", "versio...

b47a371d7f3710bde250f1249b80f02efb45a55bd5b2f778eeb7f4b94f1e0584e72daa 46 b'671860013624,kikat,2016-04-29 12:30:00,Gurgaon' ...

b47a373251aba0a7c56ed8a2b7e1a22d8c20b10f8aa74d2991b78f6d61868d20b1af18 43 b'36000291452,maagi,2014-04-28 12:30:00,Delhi' ...

HEAD BLOCK: "d9591700aab11c6e7e089d5a0990b93083474964ba16b5ca5e822e9f4821d1ef3aa6583a01b0d98d2ff978705ed52fe4c8be891b01a775f18124399faebd3778"

Table : List of states

* 1. **Show Chain**

1. Show chain will fetch the details from blockchain. Before displaying the raw information, it decodes the data into human readable format.

# barcode\_cli show chain –u <username>

**Note:** Use –b option to pass barcode to the command.

#barcode\_cli show chain –u <username> -b <barcode\_id>

root@ubuntu:~# barcode\_cli show chain -u jack

Get a pic of barcode. If pic does not look good, then press enter at terminal. Camera will take another pic. When done press q and enter to quit camera mode

q

INFO: Barcode read: 036000291452

Barcode Number: 036000291452

Product Name: maagi

Manufacturing Date: 2014-04-28 12:30:00

Locations Crossed: Delhi

Table : Show chain

* 1. **Update Chain**

Updation of chain is mainly designed for supplier users, who will update the location of the product to the blockchain up on reaching to them.

1. Run below command to update the location of the barcode, where <username> is either admin or supplier username, <location> is the location of the supplier and <barcode> is the barcode id

# barcode\_cli update chain –u <username> -l <location> -b <barcode>

root@ubuntu: ~# barcode\_cli update chain -u jack –l Hyderabad –b 36000291452

INFO: Barcode read: 36000291452

Validation Successful

Response: {

"link": "http://127.0.0.1:8008/batch\_statuses?id=98965b14748f185726a15a8bea8c46c954acaeb678b6d554ac9ffc3dfd763d585a5d253df6da15b1daf576c7f1be37f8ed0039e50ffcf921f89ea06a430cfbf8"

}

Table : Update block chain

1. Check the updation using show command

root@ubuntu: ~# barcode\_cli show chain -u jack -b 36000291452

Validation successful

INFO: Barcode read: 36000291452

Barcode Number: 36000291452

Product Name: maagi

Manufacturing Date: 2014-04-28 12:30:00

Locations Crossed: Delhi-> hyderabad

Table : Show Updated block chain

## Summary

* To migrate details from manufacturer/seller data base to blockchain a software is created
* Hyperledger Sawtooth as base software is designed in python language
* Pre configuration like setup and create users to be done
* Command line interface is used to create / read the details in the block chain
* To list states and block chains Sawtooth commands are used
* Updating the information in blockchain privilege is given to supplier type users to update location of the product.

## Direction for future work

* All the design done is CLI as of now, which can be extended to GUI
* Software can be enhanced to migrate the entire product owner’s DB in one shot
* Support to read QR code can be done
* Improve to provide Integrity as a service to multiple customers.
* Can design mobile application to read the barcode for consumers to see the status of the product

## References

1. [Hyperledger org](https://www.hyperledger.org/)
2. [Sawtooth wiki](https://sawtooth.hyperledger.org/docs/core/releases/latest/app_developers_guide/ubuntu.html)
3. [Blockchain](https://www.blockchain.com/)
4. [Python zbar](https://pypi.python.org/pypi/zbar-py/1.0.4)
5. [GIT repo](https://github.com/sreeram514/IntegrityasaService)

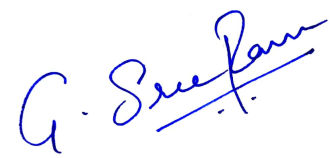
## Checklist of Items

# Checklist of items for the Final Dissertation Report

|  |  |  |
| --- | --- | --- |
|  | Is the final report neatly formatted with all the elements required for a technical Report? | Yes |
|  | Is the Cover page in proper format as given in Annexure A? | Yes |
|  | Is the Title page (Inner cover page) in proper format? | Yes |
|  | (a) Is the Certificate from the Supervisor in proper format?  (b) Has it been signed by the Supervisor? | Yes  Yes |
|  | Is the Abstract included in the report properly written within one page? Have the technical keywords been specified properly? | Yes  Yes |
|  | Is the title of your report appropriate? The title should be adequately descriptive, precise and must reflect scope of the actual work done. Uncommon abbreviations / Acronyms should not be used in the title | Yes |
|  | Have you included the List of abbreviations / Acronyms? | NA |
|  | Does the Report contain a summary of the literature survey? | Yes |
|  | Does the Table of Contents include page numbers?   1. Are the Pages numbered properly? (Ch. 1 should start on Page # 1) 2. Are the Figures numbered properly? (Figure Numbers and Figure Titles should be at the bottom of the figures) 3. Are the Tables numbered properly? (Table Numbers and Table Titles should be at the top of the tables) 4. Are the Captions for the Figures and Tables proper? 5. Are the Appendices numbered properly? Are their titles appropriate | Yes  Yes  Yes  Yes  Yes  NA |
|  | Is the conclusion of the Report based on discussion of the work? | NA |
|  | Are References or Bibliography given at the end of the Report?  Have the References been cited properly inside the text of the Report?  Are all the references cited in the body of the report | Yes  Yes  Yes |
|  | Is the report format and content according to the guidelines? The report should not be a mere printout of a Power Point Presentation, or a user manual. Source code of software need not be included in the report. | Yes |

**Declaration by Student:**

I certify that I have properly verified all the items in this checklist and ensure that the report is in proper format as specified in the course handout.



**Place:** Hyderabad **Signature of the Student**

**Date:** March 16, 2018 **Name:** SREERAM GANTA

**ID No.:** 2016HT12642