

**SAVITRIBAI PHULE PUNE UNIVERSITY**

**A PROJECT REPORT**

**ON**

**“CLOUD BASED DEVOPS SKILL  
ASSESSMENT APPLICATION”**

**Submitted toward the partial fulfillment of the Requirement of**

**BACHELOR OF ENGINEERING  
(COMPUTER ENGINEERING)**

**BY**

<b>NISHANT KUMAR</b>	<b>B150284273</b>
<b>CHETAN PAWAR</b>	<b>B150284279</b>
<b>ROSHAN PATIL</b>	<b>B150284278</b>
<b>YOGESH MAHAJAN</b>	<b>B150284260</b>

**UNDER THE GUIDANCE OF**

**PROF. AMAR MORE**

**MIT**

[An Autonomous Institute affiliated to Savitribai Phule Pune University]

**Academy of  
Engineering**

**School of Computer Engineering and Technology**

**MIT Academy of Engineering**

**Alandi(D), PUNE - 412105**

**2018-2019**

**A PROJECT REPORT  
ON  
“CLOUD BASED DEVOPS SKILL ASSESSMENT  
APPLICATION”**

**Submitted to  
SAVITRIBAI PHULE PUNE UNIVERSITY**

**Submitted toward the partial fulfillment of the Requirement of**

**BACHELOR OF ENGINEERING  
(COMPUTER ENGINEERING**

**BY**

<b>NISHANT KUMAR</b>	<b>B150284273</b>
<b>CHETAN PAWAR</b>	<b>B150284279</b>
<b>ROSHAN PATIL</b>	<b>B150284278</b>
<b>YOGESH MAHAJAN</b>	<b>B150284260</b>

**UNDER THE GUIDANCE OF  
PROF. AMAR MORE**

**MIT**

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

**Academy of  
Engineering**

**School of Computer Engineering and Technology**

**MIT Academy of Engineering**

**Alandi(D), Pune - 412105**

**2018-2019**



[An Autonomous Institute affiliated to Savitribai Phule Pune University]

Academy of  
Engineering

**School of Computer Engineering and Technology**  
**MIT Academy of Engineering**  
**ALANDI(D), PIN CODE-412105**

## **CERTIFICATE**

This is certify that the project entitled  
**“CLOUD BASED DEVOPS SKILL ASSESSMENT  
APPLICATION“**

submitted by

<b>NISHANT KUMAR</b>	<b>B150284273</b>
<b>CHETAN PAWAR</b>	<b>B150284279</b>
<b>ROSHAN PATIL</b>	<b>B150284278</b>
<b>YOGESH MAHAJAN</b>	<b>B150284260</b>

is a bonafide work carried out by students under the supervision of Prof. Amar More and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering).

**Date:**     /     /

**(Prof. Amar More)**  
**Project Guide**

**(Prof. Ranjana Badre)**  
**Dean, SCET**

**(Dr. Yogesh Bhalerao)**  
**Director**

**Name of Internal Examiner**

**External Examiner**

# MIT

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

# Academy of Engineering

## PROJECT APPROVAL SHEET

Is successfully completed by

**“CLOUD BASED DEVOPS SKILL ASSESSMENT  
APPLICATION“**

submitted by

<b>NISHANT KUMAR</b>	<b>B150284273</b>
<b>CHETAN PAWAR</b>	<b>B150284279</b>
<b>ROSHAN PATIL</b>	<b>B150284278</b>
<b>YOGESH MAHAJAN</b>	<b>B150284260</b>

at

**School of Computer Engineering and Technology**

**MIT Academy of Engineering, Alandi(D)  
SAVITRIBAI PHULE PUNE UNIVERSITY,PUNE**

ACADEMIC YEAR 2018-19

Prof. Amar More  
Internal Guide  
SCET.

Prof. Ranjana Badre  
Dean  
SCET.

## Acknowledgements

We are profoundly grateful to **Prof. Amar More** for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion.

We would like to express deepest appreciation towards **Dr. Yogesh Bhalerao**, Director, Mit Academy of Engineering, **Prof. Ranjana Badre**, Dean, School of Computer Engineering and Technology and **Prof. Sumit Khandelwal**, Project Co-ordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

NISHANT KUMAR  
CHETAN PAWAR  
ROSHAN PATIL  
YOGESH MAHAJAN

# ABSTRACT

Now a days many online tools are available to test the programming knowledge of the person like codechef. But in order to test the knowledge of the DevOps there is no such online tool available. So the aim is to develop the cloud based infrastructure to test the knowledge of DevOps of the examinee. The questions related to the DevOps will be given to the candidate along with the access to the terminal. The candidate has to do all the steps required to solve the problem given. The terminal Provided to the candidate is the communication link between the candidate and the allotted container.

We are using containers rather than VMs, because containers are small,light-weighted and fast, one application can be packed in each container image. The Kubernetes will manage the containerized applications such as database storage and user specific command across a set of containers or hosts and provides mechanisms for deployment, maintenance, and application-scaling. The container runtime packages, instantiates, and runs user commands on containerized application. The output generated will be stored in a temporary file which will be verified with the desired output stored in a database.

**Keywords:** DevOps, Orchestration, Containerization, Linux, Cloud computing, Parallel Computing, Networking

# Contents

<b>1</b>	<b>Synopsis</b>	<b>3</b>
1.1	Project Title . . . . .	3
1.2	Project Option . . . . .	3
1.3	Internal Guide . . . . .	3
1.4	Sponsorship and External Guide . . . . .	3
1.5	Technical Keywords (As per ACM Keywords) . . . . .	3
1.6	Problem Statement . . . . .	4
1.7	Abstract . . . . .	4
1.8	Goals and Objectives . . . . .	4
1.9	Relevant mathematics associated with the Project . . . . .	4
1.10	Names of Conferences / Journals where papers can be published . .	6
1.11	Review of Conference/Journal Papers supporting Project idea . . . .	6
1.12	Plan of Project Execution . . . . .	9
<b>2</b>	<b>Technical Keywords</b>	<b>10</b>
2.1	Area of Project . . . . .	10
2.2	Technical Keywords . . . . .	10
<b>3</b>	<b>Introduction</b>	<b>11</b>
3.1	Project Idea . . . . .	11
3.2	Motivation of the Project . . . . .	11
3.3	Literature Survey . . . . .	11
<b>4</b>	<b>Problem Definition and scope</b>	<b>14</b>
4.1	Problem Statement . . . . .	14
4.1.1	Goals and objectives . . . . .	14
4.1.2	Statement of scope . . . . .	14
4.2	Major Constraints . . . . .	15
4.3	Methodologies of Problem solving and efficiency issues . . . . .	15
4.4	Outcome . . . . .	15

4.5	Applications . . . . .	15
4.6	Hardware Resources Required . . . . .	16
4.7	Software Resources Required . . . . .	16
<b>5</b>	<b>Project Plan</b>	<b>17</b>
5.1	Project Estimates . . . . .	17
5.1.1	Reconciled Estimates . . . . .	19
5.1.2	Project Resources . . . . .	19
5.2	Risk Management w.r.t. NP Hard analysis . . . . .	19
5.2.1	Risk Identification . . . . .	19
5.2.2	Risk Analysis . . . . .	19
5.2.3	Overview of Risk Mitigation, Monitoring, Management . .	20
5.3	Project Schedule . . . . .	22
5.3.1	Project task set . . . . .	22
5.3.2	Task network . . . . .	22
5.3.3	Timeline Chart . . . . .	22
5.4	Team Organization . . . . .	22
5.4.1	Team structure . . . . .	23
5.4.2	Management reporting and communication . . . . .	23
<b>6</b>	<b>Software requirement specification</b>	<b>25</b>
6.1	Introduction . . . . .	25
6.1.1	Purpose and Scope of Document . . . . .	25
6.1.2	Overview of responsibilities of Developer . . . . .	25
6.2	Usage Scenario . . . . .	26
6.2.1	User profiles . . . . .	26
6.2.2	Use-cases . . . . .	26
6.3	Data Model and Description . . . . .	27
6.3.1	Data Description . . . . .	27
6.3.2	Data objects and Relationships . . . . .	28
6.4	Functional Model and Description . . . . .	28
6.4.1	Data Flow Diagram . . . . .	28
6.4.2	Activity Diagram: . . . . .	30
6.4.3	Non Functional Requirements: . . . . .	30
6.4.4	State Diagram: . . . . .	31
6.4.5	Design Constraints . . . . .	31
6.4.6	Software Interface Description . . . . .	31



<b>7</b>	<b>Detailed Design Document using Appendix B and C</b>	<b>33</b>
7.1	Introduction . . . . .	33
7.2	Architectural Design . . . . .	33
7.3	Data design (using Appendices B and C) . . . . .	34
7.3.1	Database description . . . . .	34
7.4	Component Design . . . . .	34
7.4.1	Class Diagram . . . . .	34
<b>8</b>	<b>Project Implementation</b>	<b>36</b>
8.1	Introduction . . . . .	36
8.2	Tools and Technologies Used . . . . .	36
8.3	Methodologies/Algorithm Details . . . . .	38
8.4	Verification and Validation for Acceptance . . . . .	39
<b>9</b>	<b>Software Testing</b>	<b>41</b>
9.1	Type of Testing Used . . . . .	41
9.2	Test Cases and Test Results . . . . .	41
<b>10</b>	<b>Results</b>	<b>44</b>
10.1	Screenshots . . . . .	44
10.2	Outputs . . . . .	46
<b>11</b>	<b>Deployment and Maintenance</b>	<b>48</b>
11.1	Installation and un-installation . . . . .	48
11.2	User help . . . . .	49
<b>12</b>	<b>Conclusion and future scope</b>	<b>51</b>
12.1	Conclusion . . . . .	51
12.2	Future Scope . . . . .	51
<b>A</b>	<b>References</b>	<b>53</b>
<b>B</b>	<b>Laboratory assignments on Project Analysis of Algorithmic Design</b>	<b>56</b>
<b>C</b>	<b>Laboratory assignments on Project Quality and Reliability Testing of Project Design</b>	<b>58</b>
<b>D</b>	<b>Project Planner</b>	<b>61</b>

<b>E</b>	<b>Reviewers Comments of Paper Submitted</b>	<b>63</b>
<b>F</b>	<b>Plagiarism Report</b>	<b>65</b>
<b>G</b>	<b>Term-II Project Laboratory Assignments</b>	<b>67</b>
<b>H</b>	<b>Information of Project Group Members</b>	<b>83</b>
<b>I</b>	<b>Publication Details</b>	<b>87</b>

# List of Figures

1.1	Plan of Project Execution . . . . .	9
5.1	Agile methodology . . . . .	17
5.2	Plan of Project Execution . . . . .	22
6.1	Use Case Diagram . . . . .	27
6.2	Entity Relationship Diagram . . . . .	28
6.3	DFD Level 0 . . . . .	29
6.4	DFD Level 1 . . . . .	29
6.5	Entity Relationship Diagram . . . . .	30
6.6	State Diagram . . . . .	31
7.1	Architecture diagram . . . . .	33
7.2	Class Diagram . . . . .	34
9.1	Functional Testing . . . . .	42
10.1	Deployed Containers . . . . .	44
10.2	Results . . . . .	45
10.3	Output . . . . .	46
C.1	Unit Testing . . . . .	58
C.2	Reliability Testing . . . . .	59
D.1	Plan of Project Execution . . . . .	61
F.1	Plagiarism Report . . . . .	65
G.1	State Diagram . . . . .	68
G.2	Kubernetes container orchestration . . . . .	69
G.3	Sequence Diagram . . . . .	73
G.4	State Diagram . . . . .	74
G.5	Deployment Diagram . . . . .	75
G.6	Sign Up . . . . .	77

G.7 Login . . . . .	77
G.8 Create Test . . . . .	78
G.9 Test Details . . . . .	78
G.10 Ongoing Test . . . . .	79
G.11 Test Result . . . . .	79
G.12 Selenium Testing . . . . .	81

# List of Tables

4.1	Hardware Requirements . . . . .	16
5.1	Risk Analysis . . . . .	19
5.2	Risk Probability definitions . . . . .	20
5.3	Risk Impact definitions . . . . .	20
6.1	Use Cases . . . . .	27

# **Synopsis**

# Chapter 1

## Synopsis

### 1.1 Project Title

Cloud Based DevOps Skill Assessment Application

### 1.2 Project Option

Internal Project

### 1.3 Internal Guide

Prof. Amar More

### 1.4 Sponsorship and External Guide

Non sponsored

### 1.5 Technical Keywords (As per ACM Keywords)

1. D. Software

(a) D.2 SOFTWARE ENGINEERING

i. D.2.6 Programming Environments

A. Graphical environments

B. Integrated environments

C. Interactive environments

D. Programmer workbench

## **1.6 Problem Statement**

To develop a web based application to assess DevOps and Linux administration skills.

## **1.7 Abstract**

Now a days many online tools are available to test the programming knowledge of the person like codechef. But in order to test the knowledge of the DevOps there is no such online tool available. So the aim is to develop the cloud based infrastructure to test the knowledge of DevOps of the examinee. The questions related to the DevOps will be given to the candidate along with the access to the terminal. The candidate has to do all the steps required to solve the problem given. The terminal Provided to the candidate is the communication link between the candidate and the allotted container. We are using containers rather than VMs, because containers are small,light-weighted and fast, one application can be packed in each container image. The Kubernetes will manage the containerized applications such as database storage and user specific command across a set of containers or hosts and provides mechanisms for deployment, maintenance, and application-scaling. The container runtime packages, instantiates, and runs user commands on containerized application. The output generated will be stored in a temporary file which will be verified with the desired output stored in a database.

## **1.8 Goals and Objectives**

- Assess DevOps skills
- Assess Linux administration skills
- To provide a web based platform for assessment

## **1.9 Relevant mathematics associated with the Project**

### **System Description:**

- Input:  $X$  = Answers for the questions in the test.
- Output:  $Y$  = Relative rank of candidate according to his performance in the test.



- Our Project is having distributed environment. We are using DevOps Tools such as Kubernetes for generation of master and slave nodes. System will perform task of the evaluation of skills of the candidate.

**Mathematical Model:**

- Let  $S = \{U, N, T, S, \text{status}, \text{result}, F\}$   
Where,
- $U = \{u_1, u_2, u_3, \dots, u_i\}$  Finite set of users (Candidates).
- $N = \{ \text{Master}, \text{Slave} \}$   
Where,  
Master = Kubernetes master node.  
Slave = Kubernetes worker node.
- $\text{Slave} = \{ \text{slave}_1, \text{slave}_2, \dots, \text{slave}_i \}$
- $T = \{ t_1, t_2, t_3, \dots, t_i \}$  Finite set of test Scenarios.
- $S = \{ s_1, s_2, s_3, \dots, s_i \}$  Finite set of scores of a user.
- $\text{status} = \{ \text{status}_1, \text{status}_2, \dots, \text{status}_i \}$  Test status.
- Functionalities:  
Yes/No = authenticate (uname, passwd)  
Interface Candidate Web Brower (Wi)  
 $S_i = \text{get test Scenario Score}(t_i)$   
  
Rank = apply ranking algorithm ( $t_i, s_i$ )  
Result = generate test result.
- Success conditions : Correct Evaluation of candidates as per performance.
- Failure conditions : Incorrect Evaluation consisting rank of candidates.

## 1.10 Names of Conferences / Journals where papers can be published

- Computer Society of India Communications, 2019
- CCGrid May, 2019 - 19th Annual IEEE/ACM International Symposium in Cluster, Cloud, and Grid Computing, Larnaca, Cyprus

## 1.11 Review of Conference/Journal Papers supporting Project idea

### 1. Cloud Computing Innovation in India: A Framework and Roadmap - White Paper 2.0,” in Cloud Computing Innovation in India: A Framework and Roadmap - White Paper 2.0

Explores the market opportunities for cloud computing in India. Cloud Computing is a new paradigm in information technology (IT) and IT-enable services (ITES) that transform computing as a resource to computing as a service. It is a disruptive technology with influence pervading across all aspect of a modern economy. While this has the potential of leapfrogging the economy of emerging market like India, the adoption and deployments in such countries poses a unique sets of technological, business, and regulatory challenges. Examines the viability of developing cloud computing markets, applications, and services in India.

### 2. Understanding DevOps & bridging the gap from continuous integration to continuous delivery,” Fifth International Conference on the Innovative Computing Technology (INTECH 2015), Pontevedra, 2015

As part of Agile transformations in past few years we have seen IT organizations adopting continuous integration principles in their softwares delivery lifecycle, which has improved the efficiency of development teams. With the time it has been realized that this optimization as part of continuous integration 'alone' is just not sufficient to make the entire delivery lifecycle efficient or is not driving the organizations efficiency. This paper tries to cover all aspects of Devops applicable to various phases of SDLC and specifically talks about business need, ways to move from continuous integration to continuous delivery and its benefits. Continuous delivery transformation in this paper is explained with a real life case study that how infrastructure can be maintained.

**3. DevOps: Introducing Infrastructure-as-Code,” 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C), Buenos Aires, 2017**

DevOps result in a series of software engineering tactics aimed at shortening the actionable operation of software design changes. One of these many tactic is to harness infrastructure-as-code, that is, writing a blueprint that contain deployment specifications ready for orchestration in the cloud. It discusses all necessary elements and abstractions in writing and maintaining that blueprints, revolving around a key standard for its expression, namely, the OASIS Topology and Orchestration Specification for Cloud Applications (TOSCA) industrial standards adopted by as many as 60+ big industrial players worldwide.

**4. Improve software quality through practicing DevOps,” 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, 2017**

DevOps is extended from certain agile practices with a mix of patterns intended to improve collaboration between development and operation teams. The main purpose of this paper is to conduct a study on how DevOps practice has impacted to software quality. The secondary objective is to find how to improve quality efficiently. Automation is the most critical factor to improve the software quality. As per the results of multiple regression analysis, it has proved culture, automation, measurement and sharing are important factors to consider to improve quality of the software. In conclusion it can be recommended to use DevOps to achieve high quality software.

**5. DevOps: Making It Easy to Do the Right Thing,” in IEEE Software, vol. 33, no. 3, pp. 53-59, May-June 2016**

Wotif Group used DevOps principles to recover from the downward spirals of manual release activities that many IT departments face. Its approach involved the concept of making it efficient to do the working thing By defining the right things (deployment standards) for developments and operations teams and making it easy to adopt, Wotif drastically and effectively improved the average release cycle time.

**6. ”Continuous practices and devops: beyond the buzz, what does it all mean?,” 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Vienna, 2017**

DevOps and continuous practices are attracting steadily growing attentions by

both practitioners and researchers in the software engineering community. The terms are often used inconsistently, interchangeably and with unclear meaning, however. By taking the positions that, this ambiguity and miscommunication renders the community great harm, their effects and interplay between them, reduce ambiguity.

**7. GNU/Linux shell access through a web-browser for an embedded Linux e-learning system,” 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari, 2011**

The internet is growing rapidly and has a more impact on the education sector than it had ever before. This paper aims at describing a novel way to extend the e-Learning techniques used to the area of embedded Linux education. For these techniques, Open Source Software technologies such as AJAX, PHP, Apache have been used in an innovative way to impart embedded Linux education just by use of normal web-browsers which make the learning system as easy as checking a mail. In this paper, ways of enabling the embedded devices Linux shell access through a web-browser is explained, which is a key feature of the system. This improves scalability and accessibility for multiple users.

**8. Establish new concept to develop evaluation system of examination questions and examination result,” 2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIM-SEC), Dengleng, 2011**

This paper aims to establish a reasonable, objective, quantized evaluation standard of analyzing examination and score, and develop the evaluation index system of examination questions and examination result analyzing. A lot of reasonable and objective ideas such as examination difficulty, estimate score, objective difficulty and so on are risen and defined, and some related quantized calculation methods are given, and the examination result analyzing and examination principle software system which can make a quantized decision is programmed.

**9. Task Based Automatic Examination System for Sequenced Test,” 2009 International Conference on Electronic Computer Technology, Macau, 2009**

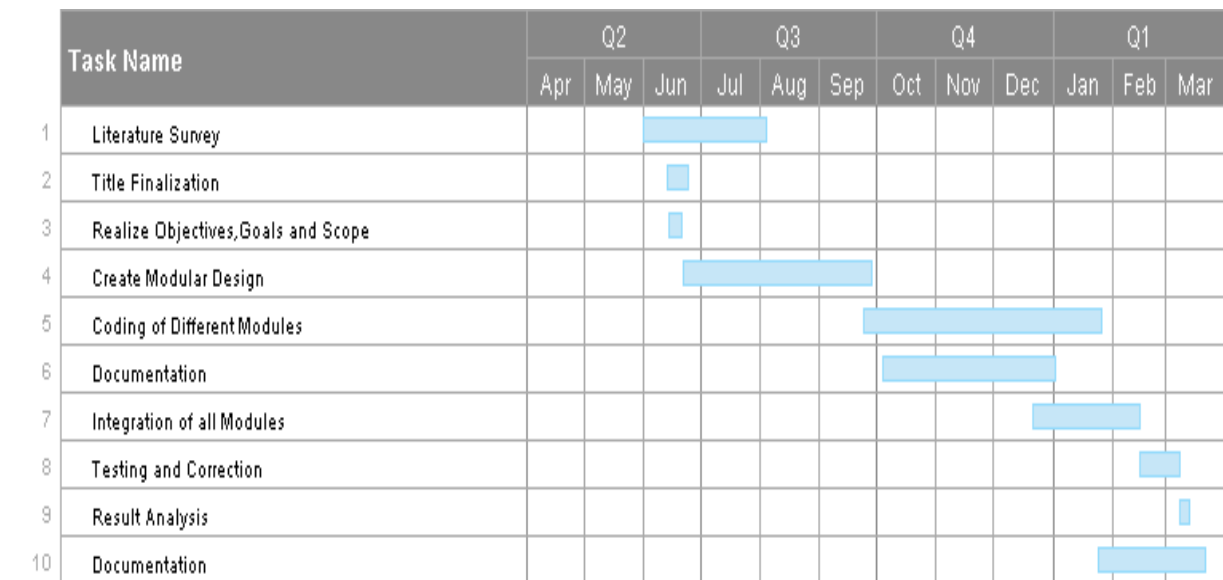
Computer greatly influences our educational environment. Over the last few years, online automatic computer examination systems have been widely used for computer-based tests, but these systems are based on traditional question-answer examination style which is not fit for the sequenced test. The sequenced

test should consider the context of the examinee.ex. the order of questions or the permissions of the examinee, to grade an examinee.This paper propose an effective and practical automatic examination architecturebased on task. The task is to abstract from the examination process and meet the requests of the sequenced test, such as order and dependency.

**10. "Containerization and the PaaS Cloud," in IEEE Cloud Computing, vol. 2, no. 3, pp. 24-31, May-June 2015.**

Containerization is a lightweight virtualization solution. Apart from exhibiting benefits over traditional virtual machines in the cloud, containers are especially relevant for PaaS clouds to manage and orchestrate applications through containers as an application packaging mechanism.

## 1.12 Plan of Project Execution



**Figure 1.1:** Plan of Project Execution

## Chapter 2

### Technical Keywords

#### 2.1 Area of Project

System Application

#### 2.2 Technical Keywords

##### 1. D. Software

##### (a) D.2 SOFTWARE ENGINEERING

##### i. D.2.6 Programming Environments

- A. Graphical environments
- B. Integrated environments
- C. Interactive environments
- D. Programmer workbench

## **Chapter 3**

### **Introduction**

#### **3.1 Project Idea**

The aim is to develop a web application which offers quick and easy way to appear for the test which also provides the results immediately after the examination with 100 percentage accuracy and security. Assessment of Linux and DevOps skills can be done immediately.

#### **3.2 Motivation of the Project**

Now a days many online tools are available to test the programming knowledge of the person like codechef. But in order to test the knowledge of the Linux and DevOps there is no such online tool available. So the aim is to develop the cloud based infrastructure to test the knowledge of Linux and DevOps of the examinee. This project aims to establish a reasonable, objective, quantized evaluation standard of analyzing examination and score, and develop the evaluation index system of examination questions and examination result analyzing.

#### **3.3 Literature Survey**

- Number of literature pertaining techniques to DevOps and cloud computing published already and are available for public usage. As per Wotif Group [2] used DevOps principles to recover from the downward spirals of manual release activities that many IT departments face. Its approach involved the concept of making it efficient to do the working thing By defining the right things (deployment standards) for developments and operations teams and making it easy to adopt, Wotif drastically and effectively improved the average release cycle time. Containerization is a lightweight virtualization solution.

- The paper presented in conference 2017 Seventeenth International Conference on Advances in ICT [3], DevOps is extended from certain agile practices with a mix of patterns intended to improve collaboration between development and operation teams.
- The main purpose of this paper is to conduct a study on how DevOps practice has impacted to software quality. The secondary objective is to find how to improve quality efficiently. Automation is the most critical factor to improve the software quality. As per the results of multiple regression analysis, it has proved culture, automation, measurement and sharing are important factors to consider to improve quality of the software.
- In the paper[7] Explores the market opportunities for cloud computing in India. Cloud Computing is a new paradigm in information technology (IT) and IT-enable services(ITES) that transform computing as a resource to computing as a service. It is a disruptive technology with influence pervading across all aspect of a modern economy.
- In the paper presented in Euromicro Conference on Software Engineering and Advanced Applications [4] DevOps and continuous practices are attracting steadily growing attentions by both practitioners and researchers in the software engineering community. The terms are often used inconsistently, interchangeably and with unclear meaning, however. By taking the positions that, this ambiguity and miscommunication renders the community great harm, their effects and interplay between them, reduce ambiguity.
- GNU/Linux shell access through a web-browser for an embedded Linux e-learning system [5] represents The internet is growing rapidly and has a more impact on the education sector than it had ever before. This paper aims at describing a novel way to extend the e-Learning techniques used to the area of embedded Linux education. For these techniques, Open Source Software technologies such as AJAX, PHP, Apache have been used in an innovative way to impart embedded Linux education just by use of normal web-browsers which make the learning system as easy as checking a mail. In this paper, ways of enabling the embedded devices Linux shell access through a web-browser is explained, which is a key feature of the system. This improves scalability and accessibility for multiple users.
- The paper represented in Establish new concept to develop evaluation system



of examination questions and examination result. [6] This paper aims to establish a reasonable, objective, quantized evaluation standard of analyzing examination and score, and develop the evaluation index system of examination questions and examination result analyzing. A lot of reasonable and objective ideas such as examination difficulty, estimate score, objective difficulty and so on are risen and defined, and some related quantified calculation methods are given, and the examination result analyzing and examination principle software system which can make a quantified decision is programmed.

- Containerization and the PaaS Cloud [9] focus on Containerization is a lightweight virtualization solution. Apart from exhibiting benefits over traditional virtual machines in the cloud, containers are especially relevant for PaaS clouds to manage and orchestrate applications through containers as an application packaging mechanism.
- Task Based Automatic Examination System for Sequenced Test [8] represents Computer greatly influences our educational environment. Over the last few years, online automatic computer examination systems have been widely used for computer-based tests, but these systems are based on traditional question-answer examination style which is not fit for the sequenced test. The sequenced test should consider the context of the examinee.ex. the order of questions or the permissions of the examinee, to grade an examinee. This paper propose an effective and practical automatic examination architecture based on task. The task is to abstract from the examination process and meet the requests of the sequenced test, such as order and dependency.
- Our Project is having distributed environment. We are using DevOps Tools such as Kubernetes for generation of master and slave nodes. System will perform task of the evaluation of skills of the candidate.

## Chapter 4

### Problem Definition and scope

#### 4.1 Problem Statement

To develop a web based application to assess DevOps and Linux administration skills.

##### 4.1.1 Goals and objectives

**Goals :**

- To design a web based application where candidate can interact with live linux environment and perform task given.

**Objective :**

- To provide a web based platform for assessment
- Assess DevOps skills
- Assess Linux administration skills

##### 4.1.2 Statement of scope

- This application can be used to assess Linux and DevOps skills like configuration ,management etc.
- The project will efficiently make use of containers with efficient scheduling and management to assess Linux administration skills and DevOps skills.
- The project will provide a user friendly web based platform for assessment.

## 4.2 Major Constraints

- Use of containers restrict the use of hardware resource as they are disabled and do not allow hardware related task execution such as configuration of network adapter.
- The state of containers cannot be saved and hence cannot be reviewed in future once it has been evaluated.

## 4.3 Methodologies of Problem solving and efficiency issues

- The physical(live) environment can be provided using virtual machines or by containers.
- Virtual machines and containers differ in several ways, but the primary difference is that containers provide a way to virtualize an OS so that multiple workloads can run on a single OS instance. With VMs, the hardware is being virtualized to run multiple OS instances. Containers speed, agility, and portability make them yet another tool to help streamline software development.

## 4.4 Outcome

- Assessment of the candidate's practical knowledge regarding Linux and DevOps skills.
- Organization can choose or hire the best candidates.

## 4.5 Applications

- Web based Environment where organization can choose the right candidates through practical assessment.
- Users can perform self evaluation and work harder to achieve their career goals.
- Assessment of Linux Skills, DevOps Skills, Configuration management of servers.

## 4.6 Hardware Resources Required

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	2 GHz	For fast processing
2	RAM	3 GB	Complete process faster

**Table 4.1:** Hardware Requirements

## 4.7 Software Resources Required

### Platform :

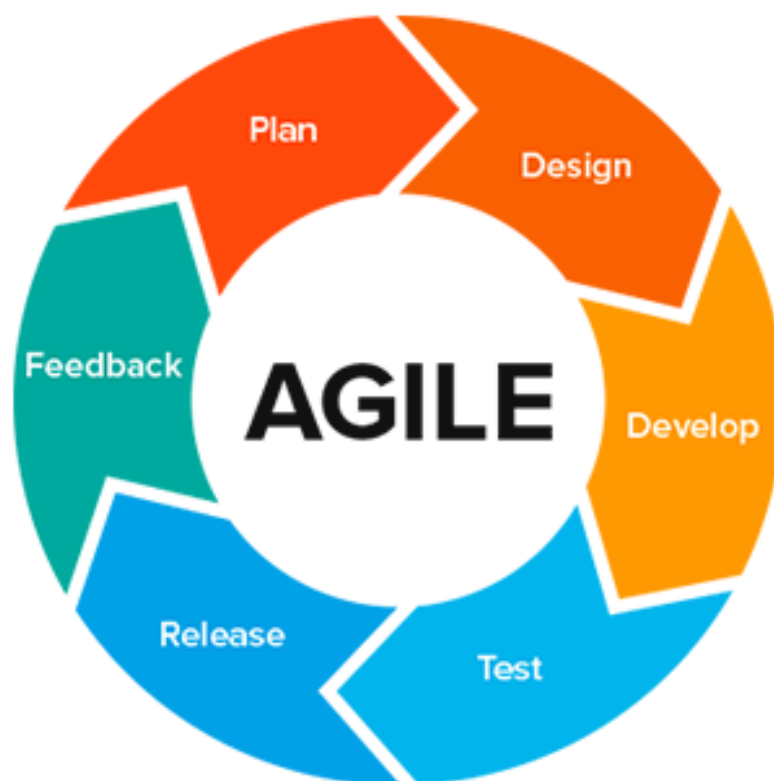
1. Operating System: Ubuntu 18 LTS (bionic)
2. IDE: Pycharm
3. Programming Language: Python, Django, HTML5
4. Database : MySql

## Chapter 5

### Project Plan

#### 5.1 Project Estimates

We are using Agile methodology to implement our project. AGILE methodology is a practice that promotes continuous iteration of development and testing throughout the software development lifecycle of the project. Both development and testing activities are concurrent unlike the Waterfall model.



**Figure 5.1:** Agile methodology

- **Scope out and prioritize projects :**

During the first step of the agile software development life cycle, the team scopes out and prioritizes projects. Some teams may work on more than one project at the same time depending on the departments organization.

- **Diagram requirements for the initial sprint :**

Once you have identified the project, work with stakeholders to determine requirements. You might want to use user flow diagrams or high-level UML diagrams to demonstrate how the new feature should function and how it will fit into your existing system.

- **Construction/iteration :**

Once a team has defined requirements for the initial sprint based on stakeholder feedback and requirements, the work begins. UX designers and developers begin work on their first iteration of the project, with the goal of having a working product to launch at the end of the sprint. Remember, the product will undergo various rounds of revisions, so this first iteration might only include the bare minimum functionality. The team can and will have additional sprints to expand upon the overall product.

- **Release the iteration into production :**

Test the system.

Your quality assurance (QA) team should test functionality, detect bugs, and record wins and losses.

Address any defects.

Finalize system and user documentation.

- **Production and ongoing support for the software release :**

This phase involves ongoing support for the software release. In other words, your team should keep the system running smoothly and show users how to use it. The production phase ends when support has ended or when the release is planned for retirement.

- **Retirement**

During the retirement phase, you remove the system release from production, typically when you want to replace a system with a new release or when the system becomes redundant, obsolete, or contrary to your business model.

### 5.1.1 Reconciled Estimates

#### Cost Estimate

As the software we have used is open source software. Our project is free of cost.

#### Time Estimates

The time required for our project is 10 months.

### 5.1.2 Project Resources

PC/Laptop with i5 or above Processor and 8 GB RAM Ubuntu 16 or higher. Pycharm IDE, Django, Python Interpreter.

## 5.2 Risk Management w.r.t. NP Hard analysis

This section discusses Project risks and the approach to managing them.

### 5.2.1 Risk Identification

Risk identification will involve the project team, appropriate stakeholders, and will include an evaluation of environmental factors, organizational culture and the project management plan including the project scope. Careful attention will be given to the project deliverables, assumptions, constraints, WBS, cost/effort estimates, resource plan, and other key project documents. A Risk Management Log will be generated and updated as needed and will be stored electronically in the project library located at file location.

### 5.2.2 Risk Analysis

The probability and impact of occurrence for each identified risk will be assessed by the project manager, with input from the project team using the following approach:

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Management of environment becomes an issue when no of candidates is large.	High	Low	High	Medium
2	Proper internet connection must be available	Low	Medium	Low	Low
3	Container failure	Low	Low	Medium	Low

**Table 5.1:** Risk Analysis

Probability	Value	Description
High	Probability of occurrence is	> 75%
Medium	Probability of occurrence is	26 – 75%
Low	Probability of occurrence is	< 25%

**Table 5.2:** Risk Probability definitions

Impact	Value	Description
Very high	> 10%	Schedule impact or Unacceptable quality
High	5 – 10%	Schedule impact or Some parts of the project have low quality
Medium	< 5%	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

**Table 5.3:** Risk Impact definitions

### 5.2.3 Overview of Risk Mitigation, Monitoring, Management

The level of risk on a project will be tracked, monitored and reported throughout the project life cycle. A Top 10 Risk List will be maintained by the project team and will be reported as a component of the project status reporting process for this project. All project change requests will be analyzed for their possible impact to the project risks. Management will be notified of important changes to risk status as a component to the Executive Project Status Report. Following are the details for each risk.



Risk ID	1
Risk Description	Management of environment becomes an issue when no of candidates is large.
Category	Development Environment.
Source	Software requirement Specification document.
Probability	High
Impact	Medium
Response	Mitigate
Strategy	Better scheduling of containers will resolve this issue
Risk Status	Occurred

Risk ID	2
Risk Description	Proper internet connection must be available
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	Low
Response	Mitigate
Strategy	Connect to good ISP.
Risk Status	Identified

Risk ID	3
Risk Description	Container failure
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Low
Response	Accept
Strategy	Better container orchestration
Risk Status	Identified

## 5.3 Project Schedule

### 5.3.1 Project task set

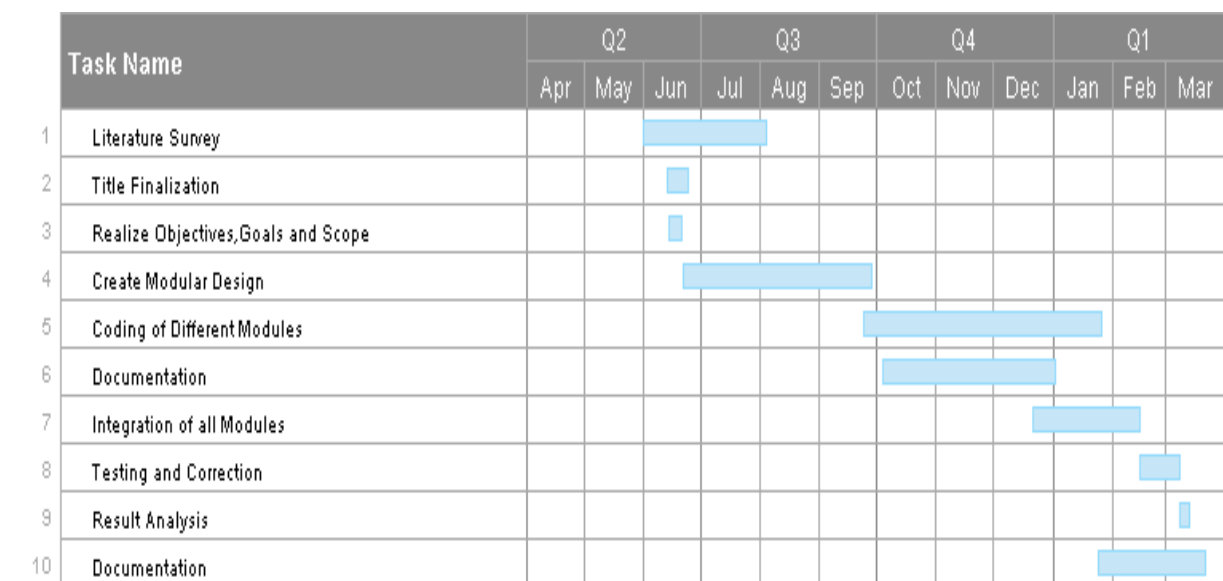
Major Tasks in the Project stages are:

- Task 1: Create container as per all dependencies.
- Task 2: Provide IP to candidate of the corresponding container.
- Task 3: Evaluate after task execution.
- Task 4: Create Result based on all task completed.
- Task 5: Display result to both candidate and organization.

### 5.3.2 Task network

Project tasks and their dependencies are noted in this diagrammatic form.

### 5.3.3 Timeline Chart



**Figure 5.2:** Plan of Project Execution

## 5.4 Team Organization

Our Internal Guide is Mr. Amar More. He always provides us helpful guidance, suggestions and immense support and also provided us with valuable suggestions.

#### **5.4.1 Team structure**

Our team consists of four members Yogesh Mahajan, Roshan Patil, Chetan Pawar and Nishant Kumar . All the four members contributed in each and every phase of the project, i.e. research work and survey and documentation. Even though the work was equally distributed we all cooperated with each other to get the job done faster. This has helped towards the development of project successfully.

#### **5.4.2 Management reporting and communication**

We all share a great bond with each other and try to help each other in the time of needs. We have a cordial relation with our guide. We keep them informed of our weekly progress and updates in the project. Their valuable advice are always helpful whenever we face any problem.

## **Software requirement specification**

## Chapter 6

# Software requirement specification

### 6.1 Introduction

#### 6.1.1 Purpose and Scope of Document

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to users. Also, we shall predict and sort out how we hope this product will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops.

In short, the purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the projects software requirements specification (SRS) document is specifically designed to create application that can be used to assess DevOps skills like configuration ,management etc.The project will demonstrate how containers can be efficiently scheduled and managed to assess Linux administration skills.This provides a user friendly web based platform for assessment. Developer and end user interested in this documnetation would include but not be limited to the system owners, the system users, the project manager and the design team.

#### 6.1.2 Overview of responsibilities of Developer

The key responsibilities of a developer are :-

- To understand the problem that the software is supposed to solve.
- To Design a solution.
- To develop and test it before releasing it to customers.

## 6.2 Usage Scenario

- **Company:**

Registration : Enter company test details.

CreateTest : Create test.

SelectQuestions : Select questions depending on domain.

GetResult : Get results of candidate.

Monitor: Monitor candidate live environment test.

- **Admin**

CreateTestScenario: Create test environment for candidates.

Monitor : Monitor status of live containers.

Evaluate : Evaluate and save results of candidates.

- **Candidate**

Registration: Enter new users details.

Login: Find correct user and allow to login.

Apply :Apply for test.

Result : Get individual result.

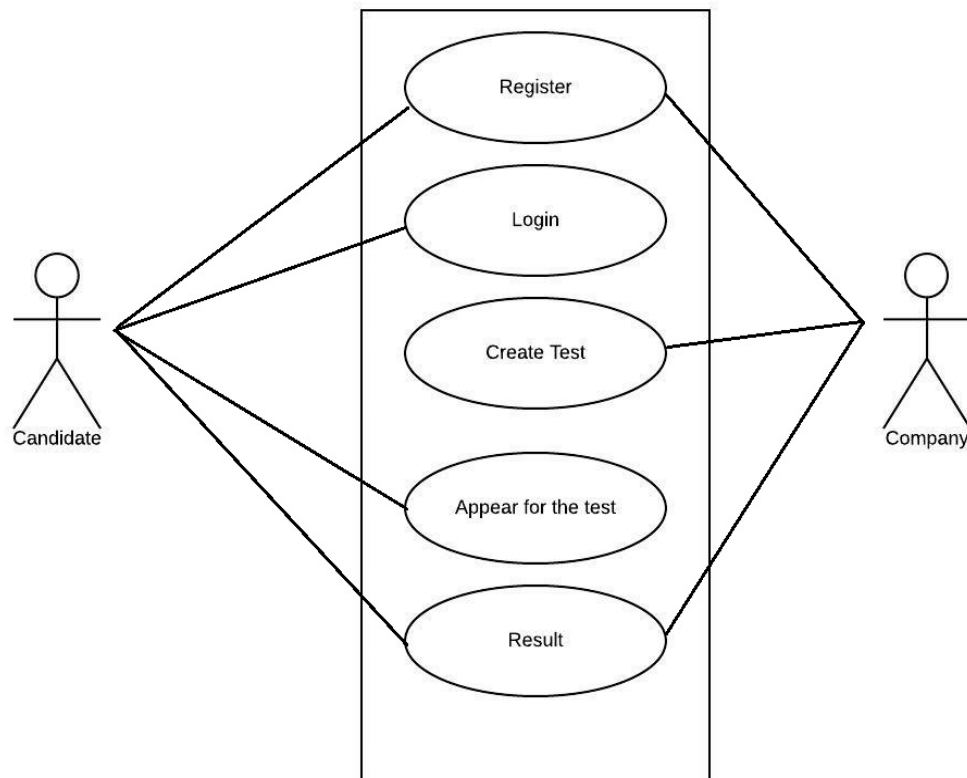
### 6.2.1 User profiles

The user can login in the system, select the company exam from the company list provided by the system. Then complete the exam as per task given. User can view previous results.

### 6.2.2 Use-cases

All use-cases for the software are presented. Description of all main Use cases using use case template is provided.

Sr No.	Use Case	Description	Actors	Assumptions
1	Test Creation	Create Test	Organization	Test creation based on selected questions
2	Take Exam	Terminal command to be executed	Student	Commands executing successfully

**Table 6.1:** Use Cases**Figure 6.1:** Use Case Diagram

## 6.3 Data Model and Description

### 6.3.1 Data Description

Data objects that will be managed/manipulated by the software are described in this section. The database entities or files or data structures required to be described. For data objects details can be given as below

### 6.3.2 Data objects and Relationships

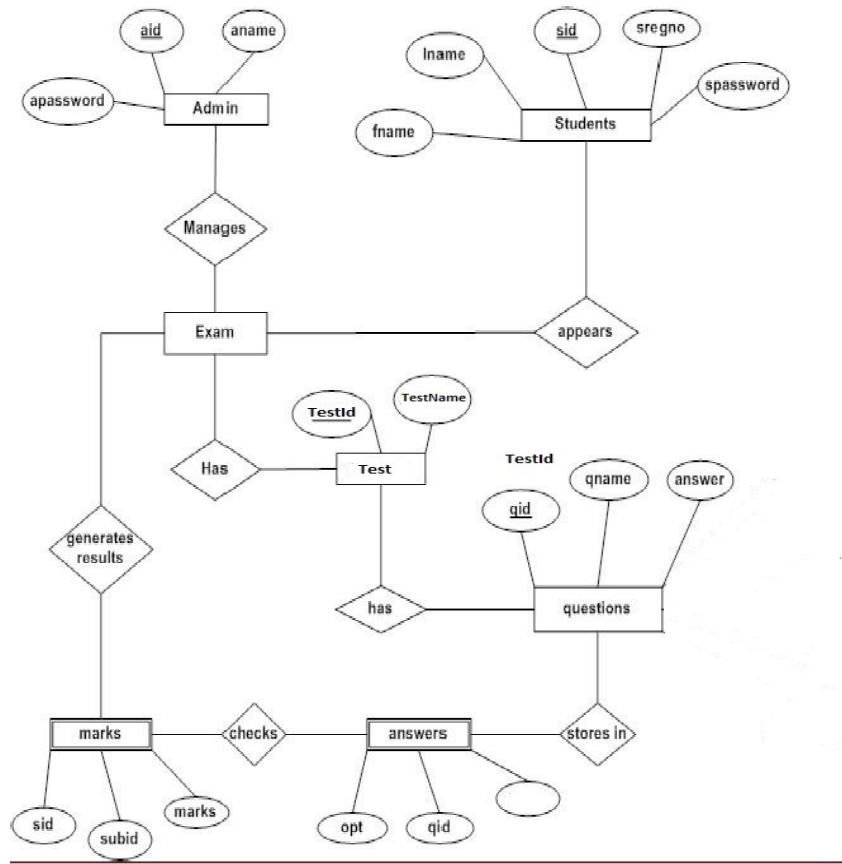


Figure 6.2: Entity Relationship Diagram

## 6.4 Functional Model and Description

A Data Flow Diagram (DFD) is a graphical representation of the flow of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system. It shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored.

### 6.4.1 Data Flow Diagram

#### Level 0 Data Flow Diagram



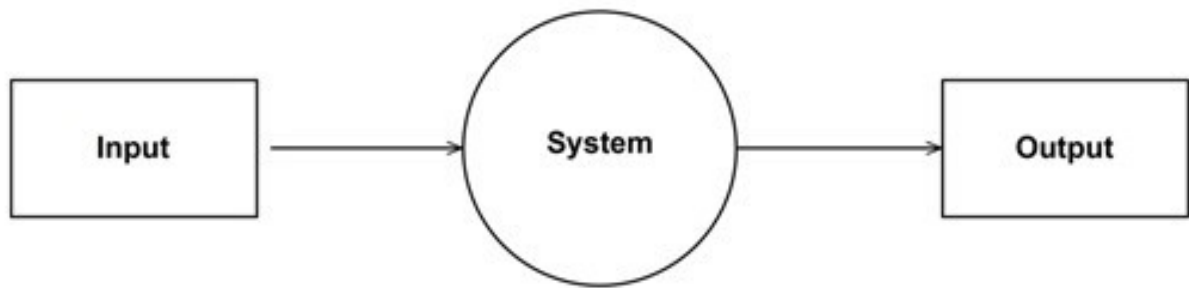


Figure 6.3: DFD Level 0

### Level 1 Data Flow Diagram

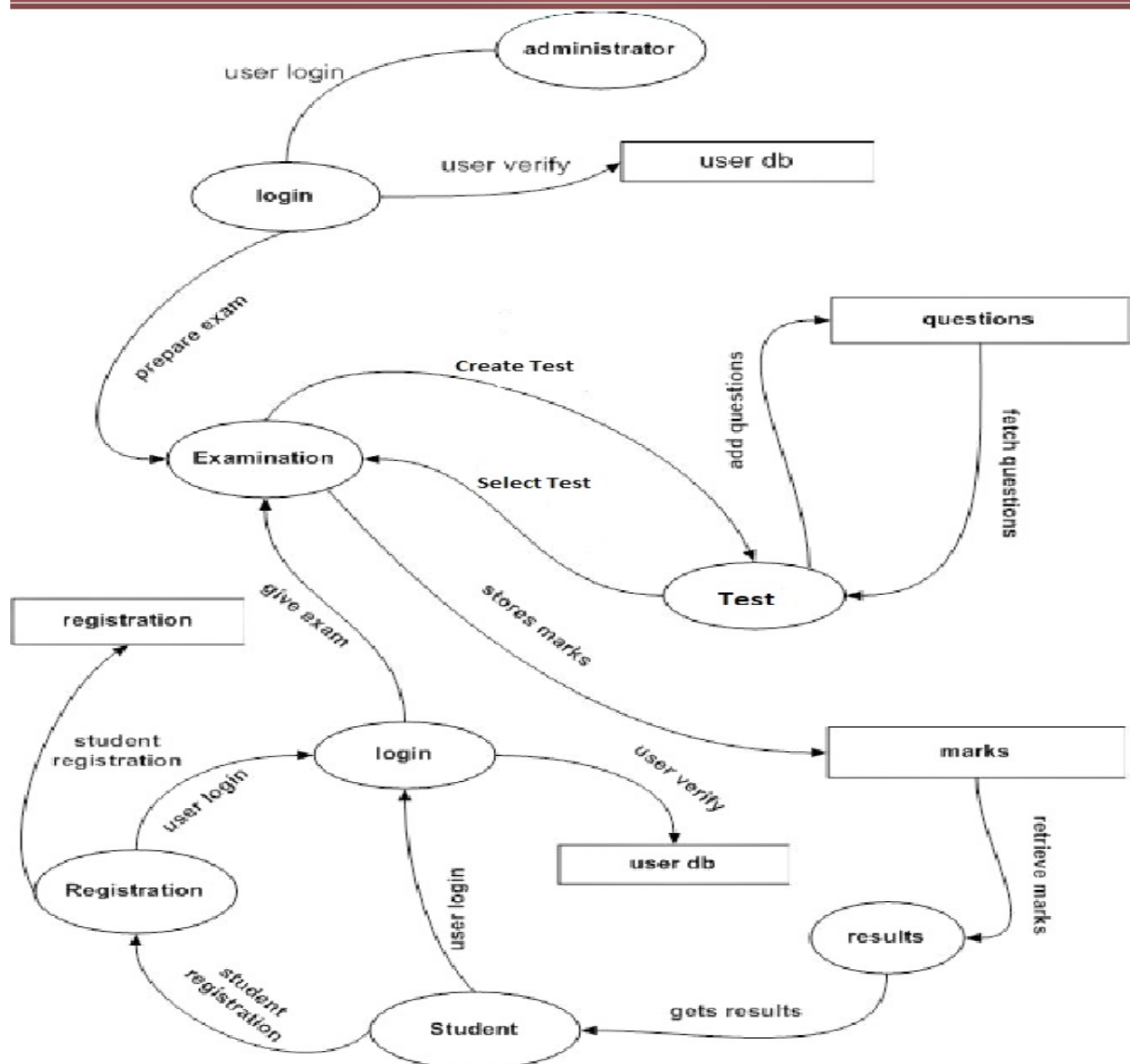


Figure 6.4: DFD Level 1

### 6.4.2 Activity Diagram:

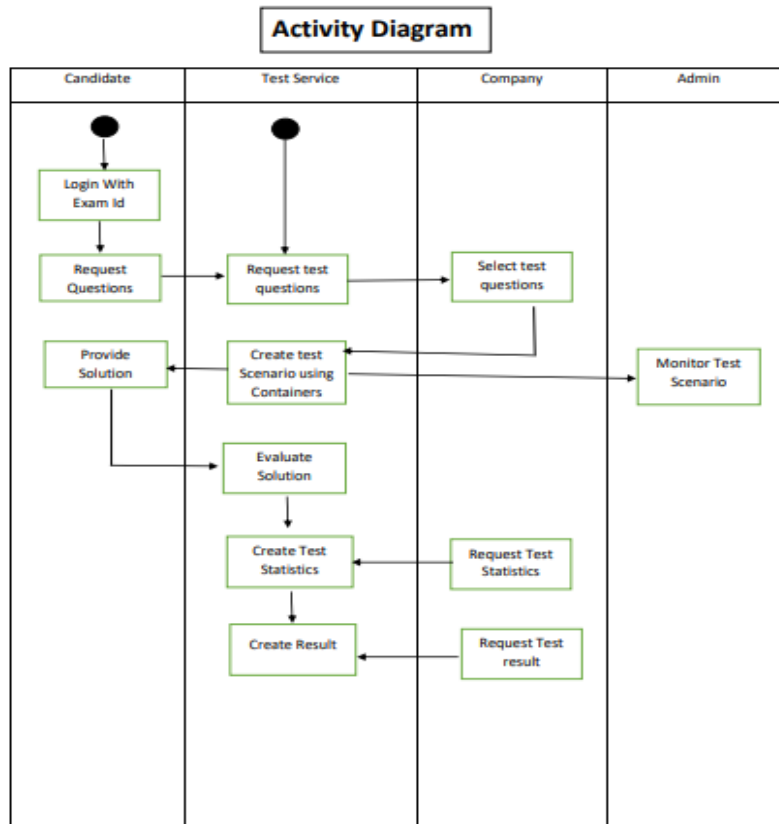


Figure 6.5: Entity Relationship Diagram

### 6.4.3 Non Functional Requirements:

#### Interface Requirements

- Our Project belongs to web based, so connecting user at online with request and response.
- Since we need to provide terminal access through web based , we have used tornado socket to implement Terminado.tornado.websocket Bidirectional communication to the browser. WebSockets allow for bidirectional communication between the browser and server.

#### Performance Requirements

- System can produce results faster on 16 GB of RAM.
- It may take more time for peak loads at master server.

- The system will be available 100% of the time. Once there is a fatal error, the system will provide understandable feed back to the user.

#### 6.4.4 State Diagram:

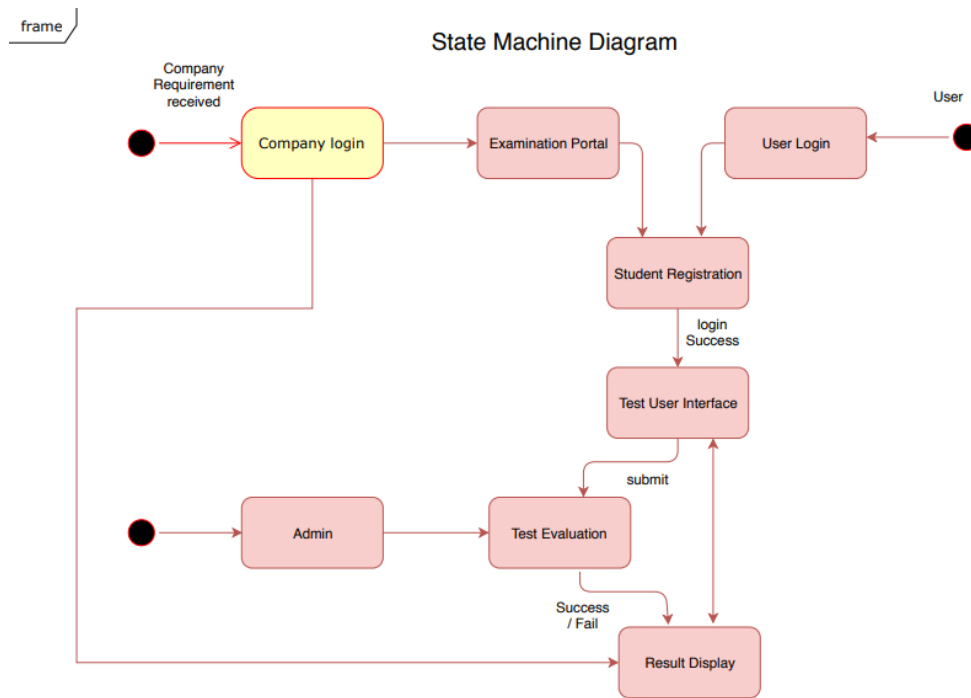


Figure 6.6: State Diagram

#### 6.4.5 Design Constraints

Use of HTML5 in our project design limit it's compatibility with different devices.

#### 6.4.6 Software Interface Description

User can interact with the system through the web application made in HTML5 and CSS using Django web framework.

## **Detailed Design Document using Appendix B and C**

## Chapter 7

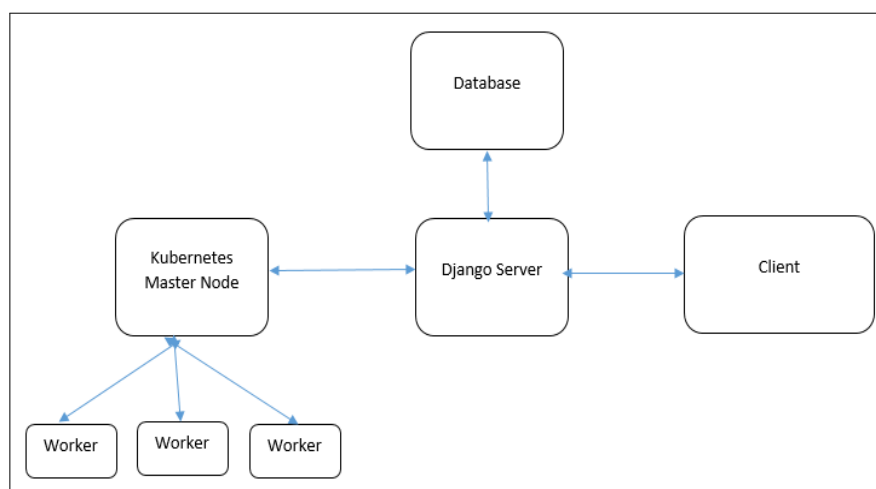
# Detailed Design Document using Appendix B and C

### 7.1 Introduction

This application can be used to assess DevOps skills like configuration ,management etc.The project will demonstrate how containers can be efficiently scheduled and managed to assess Linux administration skills.This provides a user friendly web based platform for assessment.

### 7.2 Architectural Design

This application can be used to assess DevOps skills like configuration ,management etc.The project will demonstrate how containers can be efficiently scheduled and managed to assess Linux administration skills.This provides a user friendly web based platform for assessment.



**Figure 7.1:** Architecture diagram

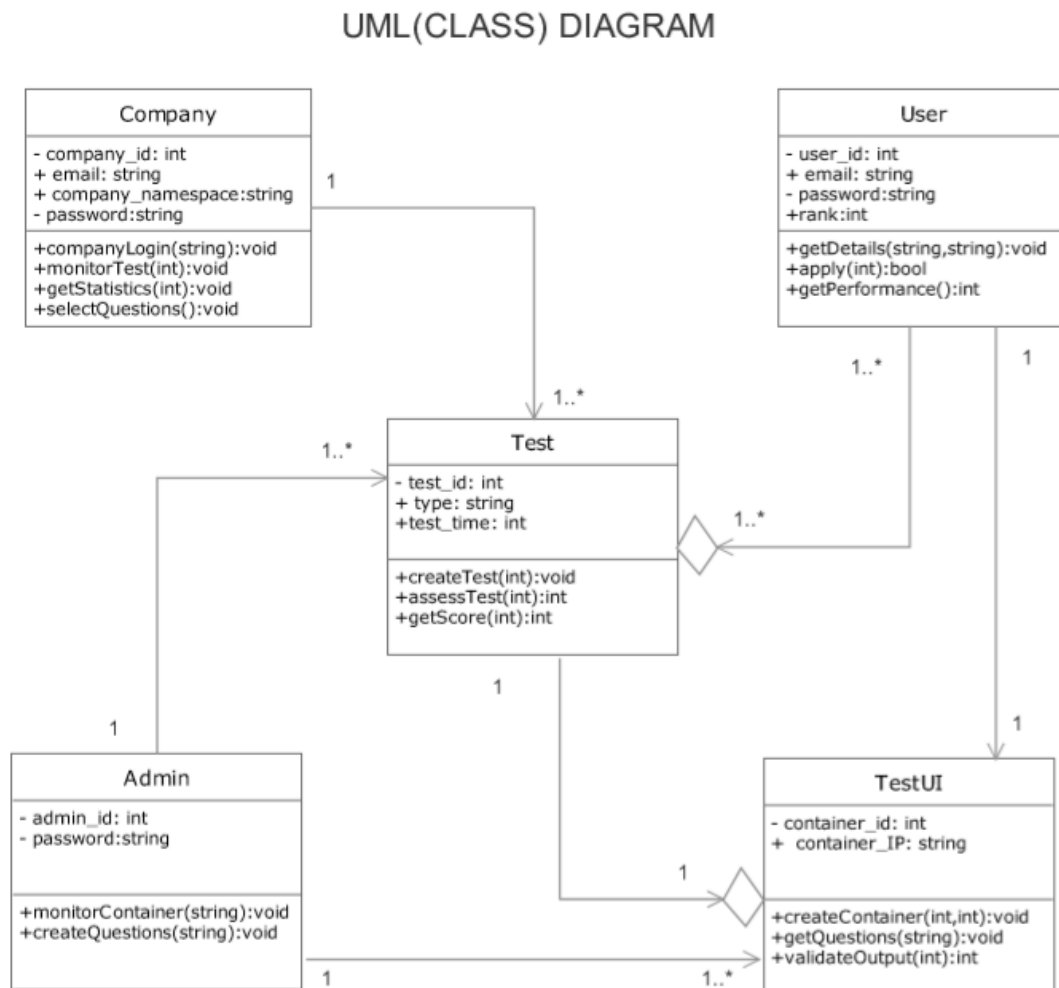
## 7.3 Data design (using Appendices B and C)

### 7.3.1 Database description

MySQL : For database SQL(Structured Query Language) is used which will store all the information regarding candidate and organization. The table maintains and stores all details of containers being deployed and in execution.

## 7.4 Component Design

### 7.4.1 Class Diagram



**Figure 7.2:** Class Diagram

## **Project Implementation**

# Chapter 8

## Project Implementation

### 8.1 Introduction

The questions related to the DevOps will be given to the candidate along with the access to the terminal. The candidate has to do all the steps required to solve the problem given. The terminal Provided to the candidate is the communication link between the candidate and the allotted container. We are using containers rather than VMs, because containers are small,light-weighted and fast, one application can be packed in each container image. The Kubernetes will manage the containerized applications such as database storage and user specific command across a set of containers or hosts and provides mechanisms for deployment, maintenance, and application-scaling.

### 8.2 Tools and Technologies Used

- **Terminado :**

This is a Tornado websocket backend for the Xterm.js Javascript terminal emulator library.

It evolved out of pyxterm, which was part of GraphTerm (as lineterm.py), v0.57.0 (2014-07-18), and ultimately derived from the public-domain Ajax-term code, v0.11 (2008-11-13) (also on Github as part of QWeb).

Modules:

- **terminado.management** : controls launching virtual terminals, connecting them to Tornado's event loop, and closing them down.
- **terminado.websocket** : Provides a websocket handler for communicating with a terminal.
- **terminado.uimodule** : Provides a Terminal Tornado UI Module.



JS: terminado/\_static/terminado.js: A lightweight wrapper to set up a term.js terminal with a websocket.

- **Docker :**

Docker is a platform for developers and sysadmins to develop, deploy, and run applications with containers. The use of Linux containers to deploy applications is called containerization. Containers are not new, but their use for easily deploying applications is.

Containerization is increasingly popular because containers are:

- **Flexible:** Even the most complex applications can be containerized.
- **Lightweight:** Containers leverage and share the host kernel.
- **Interchangeable:** You can deploy updates and upgrades on-the-fly.
- **Portable:** You can build locally, deploy to the cloud, and run anywhere.
- **Scalable:** You can increase and automatically distribute container replicas.
- **Stackable:** You can stack services vertically and on-the-fly.

- **Kubernetes :**

Kubernetes is an open source system for managing containerized applications across multiple hosts, providing basic mechanisms for deployment, maintenance, and scaling of applications.

The open source project is hosted by the Cloud Native Computing Foundation

- **Planet Scale :** Designed on the same principles that allows Google to run billions of containers a week, Kubernetes can scale without increasing your ops team.
- **Never Outgrow :** Whether testing locally or running a global enterprise, Kubernetes flexibility grows with you to deliver your applications consistently and easily no matter how complex your need is.
- **Run Anywhere :** Kubernetes is open source giving you the freedom to take advantage of on-premises, hybrid, or public cloud infrastructure, letting you effortlessly move workloads to where it matters to you.

- **Python :**

A high-level taken programming language for all-purpose programming. Python's design philosophy emphasizes code readability, especially when using large amounts of white space. It provides a structure that allows for clear programming on a small scale and on a large scale.

Python contains a dynamic sort system and automatic memory management. It supports a variety of programming paradigms, including object-oriented, imperative, functional, and procedural, and has a large and complete standard library.

- **Django :**

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. Its free and open source.

- **Ridiculously fast** :Django was designed to help developers take applications from concept to completion as quickly as possible.
- **Reassuringly secure** :Django takes security seriously and helps developers avoid many common security mistakes.
- **Exceedingly scalable** :Some of the busiest sites on the Web leverage Django's ability to quickly and flexibly scale.

## 8.3 Methodologies/Algorithm Details

- **Configuration management :**

Configuration management solves the problem of manually install and configure packages once the hardware is in place. The benefit of using configuration management solutions is that servers are deployed exactly the same way every time . If you need to make a changes across 100 thousand servers you only need to make the change in one system . In the operations environments we have worked in there were always strict controls on who could access production environment , who could make change, when changes could be made, who could physically touch hardwares , and who could access what data centers . In these highly regulated and process oriented enterprises the thought of blurring the lines between development and operation seems like a non-starter.

- **Kubernetes container orchestration:**

Containers support VM-like separation of concerns but with far less overhead and far greater flexibility. As a result, containers have reshaped the way people think about developing, deploying, and maintaining software. In a containerized architecture, the different services that constitute an application are packaged into separate containers and deployed across a cluster of physical

or virtual machines. But this gives rise to the need for container orchestration tool that automates the deployment, management, scaling, networking, and availability of container-based applications.

## **8.4 Verification and Validation for Acceptance**

### **1. V and V tasks Planning:**

- (a) Verification of contract
- (b) Evaluation of Concept document
- (c) Performing risk analysis

### **2. V and V tasks- Requirement phase:**

- (a) Evaluation of software requirements
- (b) Evaluation / analysis of the interfaces
- (c) Generation of systems test plan
- (d) Generation of Acceptance test plan

### **3. V and V tasks Design Phase:**

- (a) Evaluation of software design
- (b) Evaluation / Analysis of the Interfaces (UI)
- (c) Generation of Integration test plan
- (d) Generation of Component test plan
- (e) Generation of test design

### **4. V and V Tasks Implementation Phase:**

- (a) Evaluation of source code
- (b) Evaluation of documents
- (c) Generation of test cases
- (d) Generation of test procedure
- (e) Execution of Components test cases

## **Software Testing**

# Chapter 9

## Software Testing

### 9.1 Type of Testing Used

Functional Testing

### 9.2 Test Cases and Test Results

#### 1) Signup

<b>Test cases</b>	Registration Screen Signup
<b>Objective</b>	Click on sign up button then check all required/ mandatory fields with leaving all fields blank
<b>Expected Result</b>	All required/ mandatory fields should display with symbol * Instruction line * field(s) are mandatory should be displayed

#### 2)Test Environment

<b>Test cases</b>	Test Environment Creation
<b>Objective</b>	Selection of Examination questions
<b>Expected Result</b>	Test questions selected successfully and get respective test id

**3)Login**

<b>Test cases</b>	Candidate login to give test
<b>Objective</b>	Pods should deploy and user should get terminal
<b>Expected Result</b>	Pods deployed successfully and user got terminal

**4)Test Submission**

<b>Test cases</b>	Test submission
<b>Objective</b>	Check respective command can execute in terminal
<b>Expected Result</b>	Move to the next question or test completed successfully

**5)Test Evaluation**

<b>Test cases</b>	Test Evaluation
<b>Objective</b>	Evaluation script should run in pod and return some value
<b>Expected Result</b>	Candidate get marks for respective question

**6)Test Result**

<b>Test cases</b>	Result declaration
<b>Objective</b>	Marks of the candidate should store in database in descending order
<b>Expected Result</b>	Candidate and organization both can see the result and candidate with higher marks must be in top of the list

**Figure 9.1:** Functional Testing

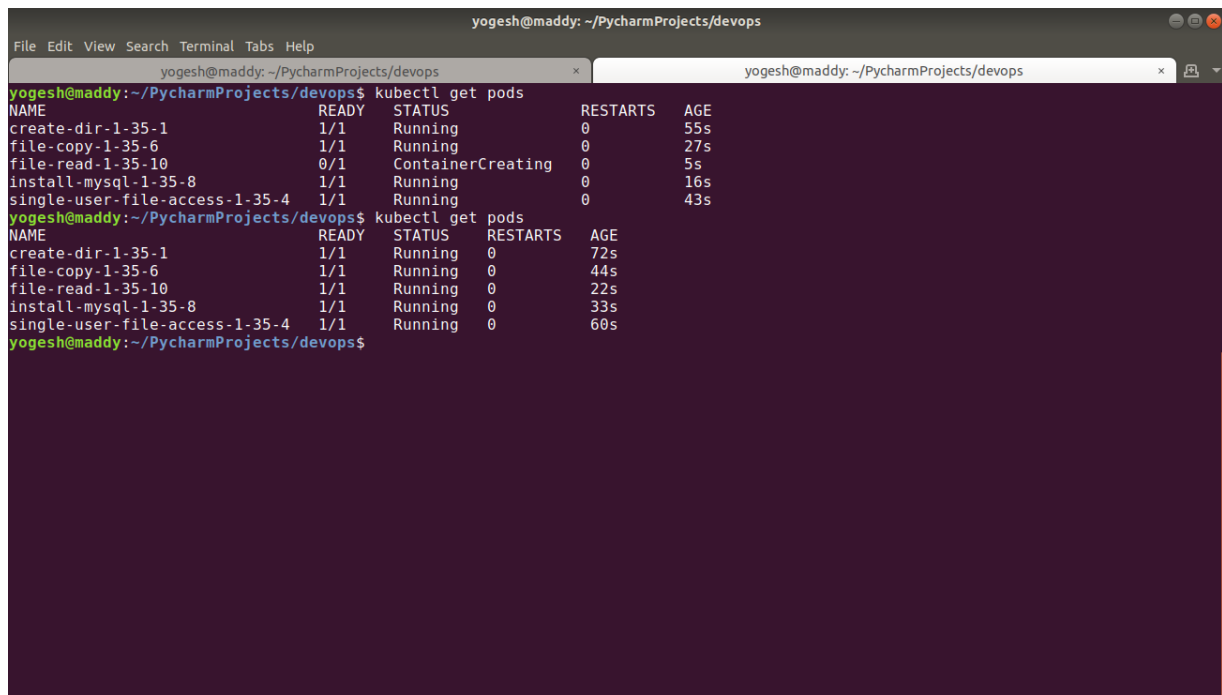
## **Result**

# Chapter 10

## Results

### 10.1 Screenshots

The containers will be created for each questions .

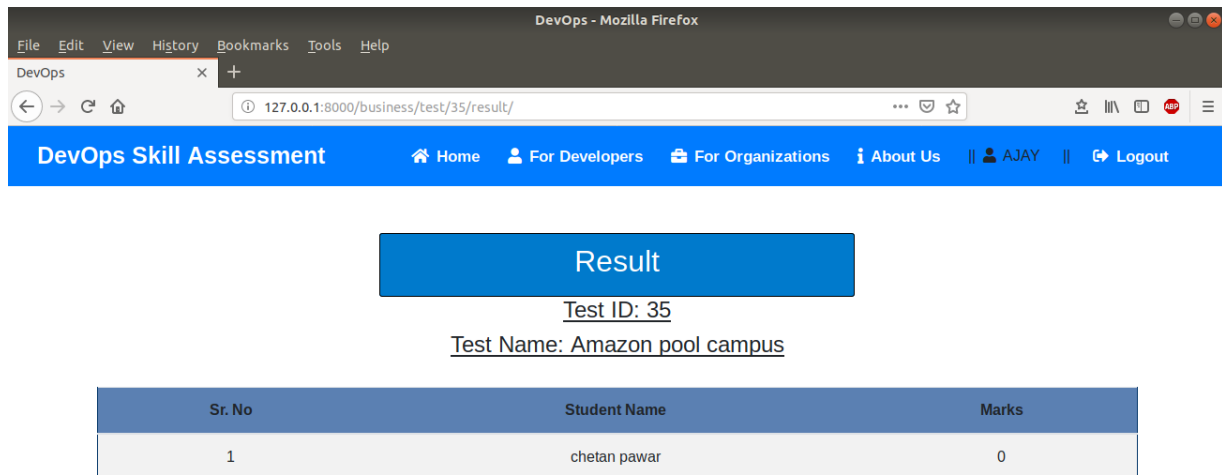


```
yogesh@maddy: ~/PycharmProjects/devops
File Edit View Search Terminal Tabs Help
yogesh@maddy: ~/PycharmProjects/devops
yogesh@maddy:~/PycharmProjects/devops$ kubectl get pods
NAME                READY   STATUS              RESTARTS   AGE
create-dir-1-35-1    1/1     Running             0           55s
file-copy-1-35-6     1/1     Running             0           27s
file-read-1-35-10    0/1     ContainerCreating   0           5s
install-mysql-1-35-8 1/1     Running             0           16s
single-user-file-access-1-35-4 1/1     Running             0           43s
yogesh@maddy:~/PycharmProjects/devops$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
create-dir-1-35-1    1/1     Running   0           72s
file-copy-1-35-6     1/1     Running   0           44s
file-read-1-35-10    1/1     Running   0           22s
install-mysql-1-35-8 1/1     Running   0           33s
single-user-file-access-1-35-4 1/1     Running   0           60s
yogesh@maddy:~/PycharmProjects/devops$
```

**Figure 10.1:** Deployed Containers



Result available to the individual candidate.



DevOps Skill Assessment

Home For Developers For Organizations About Us AJAY Logout

Result

Test ID: 35

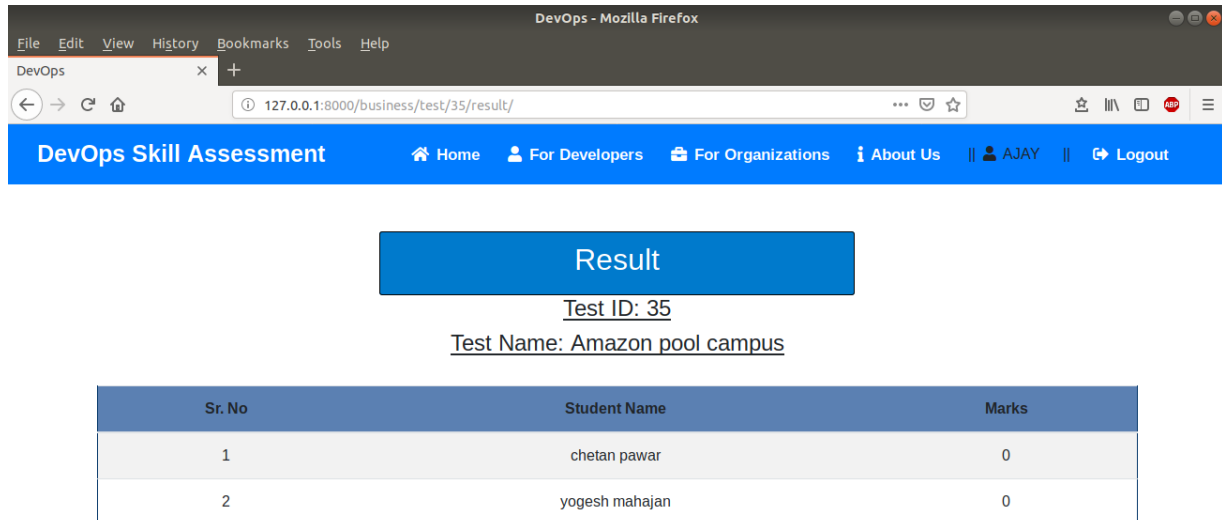
Test Name: Amazon pool campus

Sr. No	Student Name	Marks
1	chetan pawar	0

**Figure 10.2:** Results

## 10.2 Outputs

The Result of the candidates is available to organization in the sorted order of there performances.



**Figure 10.3:** Output

## **Deployment and Maintenance**

# Chapter 11

## Deployment and Maintenance

### 11.1 Installation and un-installation

#### Docker Installation:

- Update the apt package index:

```
1 sudo apt-get update
```

- Install packages to allow apt to use a repository over HTTPS:

```
1 sudo apt-get install /  
2 apt-transport-https /  
3 ca-certificates /  
4 curl /  
5 gnupg2 /  
6 software-properties-common
```

- Add Docker's official GPG key:

```
1 curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key  
add  
2 sudo apt-key fingerprint 0EBFCD88 pub 4096R/0EBFCD88 2017-02-22 Key  
fingerprint = 9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88 uid  
Docker Release (CE deb) <docker@docker.com> sub 4096R/F273FCD8  
2017-02-22
```

- Use the following command to set up the stable repository

```
1 sudo apt-get install docker-ce docker-ce-cli containerd.io
```

#### Kubernetes installation

- Installing kubeadm, kubelet

```
1 apt-get update && apt-get install -y apt-transport-https curl curl -s
  https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key add
  - cat <<EOF >/etc/apt/sources.list.d/kubernetes.list deb https://apt
    .kubernetes.io/ kubernetes-xenial main EOF apt-get update apt-get
    install -y kubelet kubeadm kubectl apt-mark hold kubelet kubeadm
    kubectl systemctl daemon-reload systemctl restart kubelet
```

- Install kubectl on Linux:Ubuntu

```
1 sudo apt-get update && sudo apt-get install -y apt-transport-https curl
  -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt
  -key add - echo "deb https://apt.kubernetes.io/ kubernetes-xenial
    main" | sudo tee -a /etc/apt/sources.list.d/kubernetes.list sudo apt
  -get update sudo apt-get install -y kubectl
```

## 11.2 User help

- The user can monitor all services from the homepage of web application.
- A user guide will be provided while deploying the project to user.

## **Conclusion and future scope**

## **Chapter 12**

### **Conclusion and future scope**

#### **12.1 Conclusion**

A cloud-based devOps skill assessment tool, which integrates containers and their orchestration , will be developed for the assessment of linux skills and configurations skill along with DevOps skills. This project presents computational issues of accessing terminal through web.

Companies that incorporate the DevOps operations to get more done, plain and simple. With a single team composed of cross-functional members all working in collaboration, DevOps systems can deliver the maximum speed, functionality, and innovation. Companies that uses this application will be easily able to create test depending on their specific requirements. This test scenarios will help company to easily evaluate candidate knowledge and thus they can find a right candidate for the desired post.

#### **12.2 Future Scope**

- We can extend our work into practice website for DevOps and Linux skills.
- Potential improvement can be made to our Container management.

## **Appendices**



# Appendix A

## References

- [1] Cloud Computing Innovation in India: A Framework and Roadmap - White Paper 2.0, in Cloud Computing Innovation in India: A Framework and Roadmap - White Paper 2.0
- [2] Understanding DevOps & bridging the gap from continuous integration to continuous delivery, Fifth International Conference on the Innovative Computing Technology (INTECH 2015), Pontevedra, 2015
- [3] DevOps: Introducing Infrastructure-as-Code, 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSEC), Buenos Aires, 2017
- [4] Improve software quality through practicing DevOps, 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, 2017
- [5] Improve software quality through practicing DevOps, 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, 2017
- [6] Continuous practices and devops: beyond the buzz, what does it all mean?, 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Vienna, 2017
- [7] GNU/Linux shell access through a web-browser for an embedded Linux e-learning system, 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari, 2011
- [8] Establish new concept to develop evaluation system of examination questions and examination result, 2011 2nd International Conference on Artificial Intelli-

gence, Management Science and Electronic Commerce (AIMSEC), Dengleng, 2011

- [9] Task Based Automatic Examination System for Sequenced Test, 2009 International Conference on Electronic Computer Technology, Macau, 2009
- [10] Containerization and the PaaS Cloud, in IEEE Cloud Computing, vol. 2, no. 3, pp. 24-31, May-June 2015.

# **Laboratory assignments on Project Analysis of Algorithmic Design**

## Appendix B

### Laboratory assignments on Project Analysis of Algorithmic Design

- Input: Tasks based questions to candidate.
- Output: Result of the test.
- Functions : Container creation Function, Container evaluation Functions, Result generation Functions.
- Success Conditions: Correct result of candidates sorted as per performance.
- Failure Conditions: Inability to evaluate a particular task.

## **Laboratory assignments on Project Quality and Reliability Testing of Project Design**

## Appendix C

### Laboratory assignments on Project Quality and Reliability Testing of Project Design

#### Test Cases and Test Results:

##### 1) Signup

<b>Test cases</b>	Registration Screen Signup
<b>Objective</b>	Click on sign up button then check all required/ mandatory fields with leaving all fields blank
<b>Expected Result</b>	All required/ mandatory fields should display with symbol * Instruction line * field(s) are mandatory should be displayed

##### 2)Test Environment

<b>Test cases</b>	Test Environment Creation
<b>Objective</b>	Selection of Examination questions
<b>Expected Result</b>	Test questions selected successfully and get respective test id

**Figure C.1:** Unit Testing

**3)Login**

<b>Test cases</b>	Candidate login to give test
<b>Objective</b>	Pods should deploy and user should get terminal
<b>Expected Result</b>	Pods deployed successfully and user got terminal

**4)Test Submission**

<b>Test cases</b>	Test submission
<b>Objective</b>	Check respective command can execute in terminal
<b>Expected Result</b>	Move to the next question or test completed successfully

**5)Test Evaluation**

<b>Test cases</b>	Test Evaluation
<b>Objective</b>	Evaluation script should run in pod and return some value
<b>Expected Result</b>	Candidate get marks for respective question

**6)Test Result**

<b>Test cases</b>	Result declaration
<b>Objective</b>	Marks of the candidate should store in database in descending order
<b>Expected Result</b>	Candidate and organization both can see the result and candidate with higher marks must be in top of the list

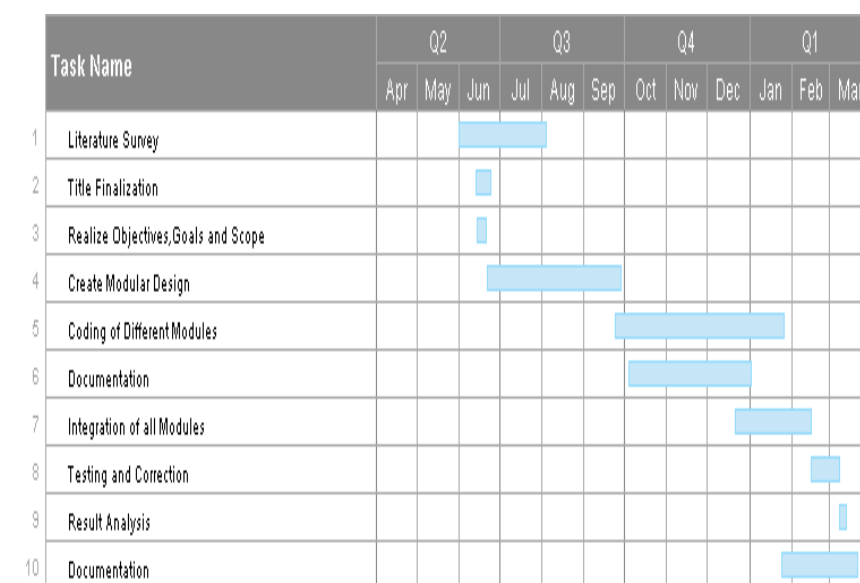
**Figure C.2:** Reliability Testing

## **Project Planner**



## Appendix D

### Project Planner



**Figure D.1:** Plan of Project Execution

## **Reviewers Comments of Paper Submitted**

## Appendix E

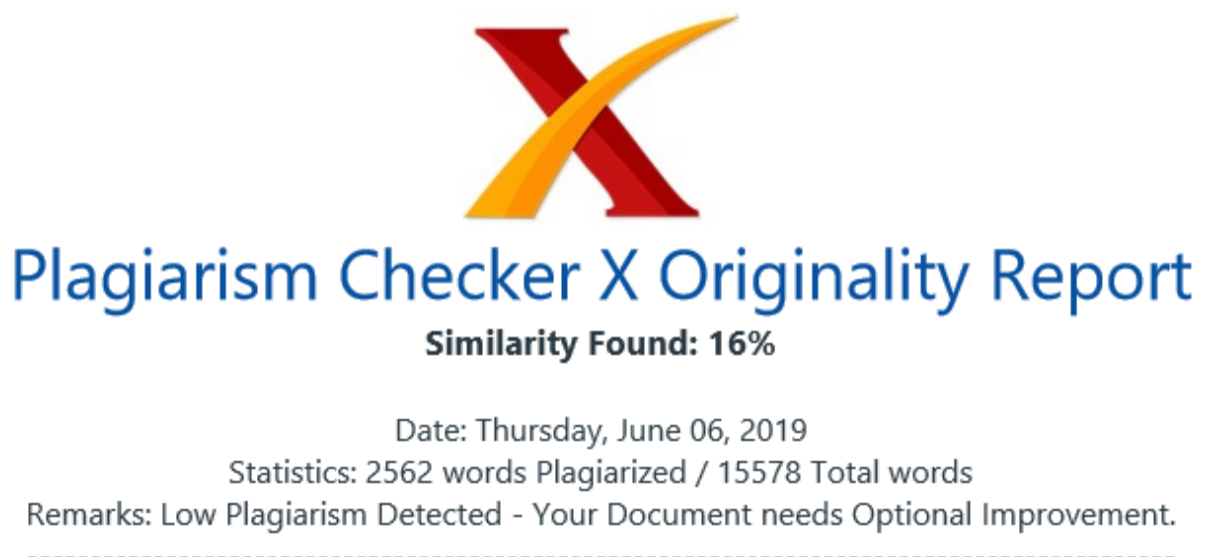
### Reviewers Comments of Paper Submitted

1. Paper Title: Cloud Based DevOps Skill Assessment Application
2. Name of the Conference/Journal where paper submitted : Computer Society of India Research Journal Of Engineering Technology
3. Paper accepted/rejected : Accepted
4. Review comments by reviewer : Good Presentation, Identify more services.
5. Corrective actions if any : Paper Was Reviewed And Submitted.

# **Plagiarism Report**

## Appendix F

### Plagiarism Report



**Figure F.1:** Plagiarism Report

## **Term-II Project Laboratory Assignments**

# Appendix G

## Term-II Project Laboratory Assignments

### Assignment 1

- **Problem statement:** Review of proposed design and necessary corrective action is taking to consider and submit publication/presentation details with review report.
- **Project Architectural Design:**

The aim is to develop the cloud based infrastructure to test the knowledge of DevOps of the examinee. The questions related to the DevOps will be given to the candidate along with the access to the terminal. The candidate has to do all the steps required to solve the problem given. The terminal Provided to the candidate is the communication link between the candidate and the allotted container. We are using containers rather than VMs, because containers are small, light-weighted and fast, one application can be packed in each container image. The Kubernetes will manage the containerized applications such as database storage and user specific command across a set of containers or hosts and provides mechanisms for deployment, maintenance, and application-scaling. The container runtime packages, instantiates, and runs user commands on containerized application. The output generated will be stored in a temporary file which will be verified with the desired output stored in a database. To work with Kubernetes, we use Kubernetes API objects to describe our clusters desired state: what application or other workloads we want to run, what container images we use, the number of replicas, what network and disk resources we want to make available, and more. The Kubernetes Master is a collection of three processes that run on the single node in our cluster, which is designated as a master node. Those processes are: kube-apiserver, kube-scheduler and kubecontroller-manager. Each individual non-master node in our cluster runs two processes kube-proxy kublet. DevOps is a software development method-

ology that combines software development with information technology operations . The goal of it is to shorten the systems development phases while also delivering fixes, features, and updates frequently in close alignment .The DevOps approach is to include automation and event monitoring at all steps of the software development .The focus on the developer collaboration enables a new approach to managing the complexity of real world problems. I believe the operations complexity breaks down into a few categories: configuration management , infrastructure and deployment automation, log and performance management, and monitoring.

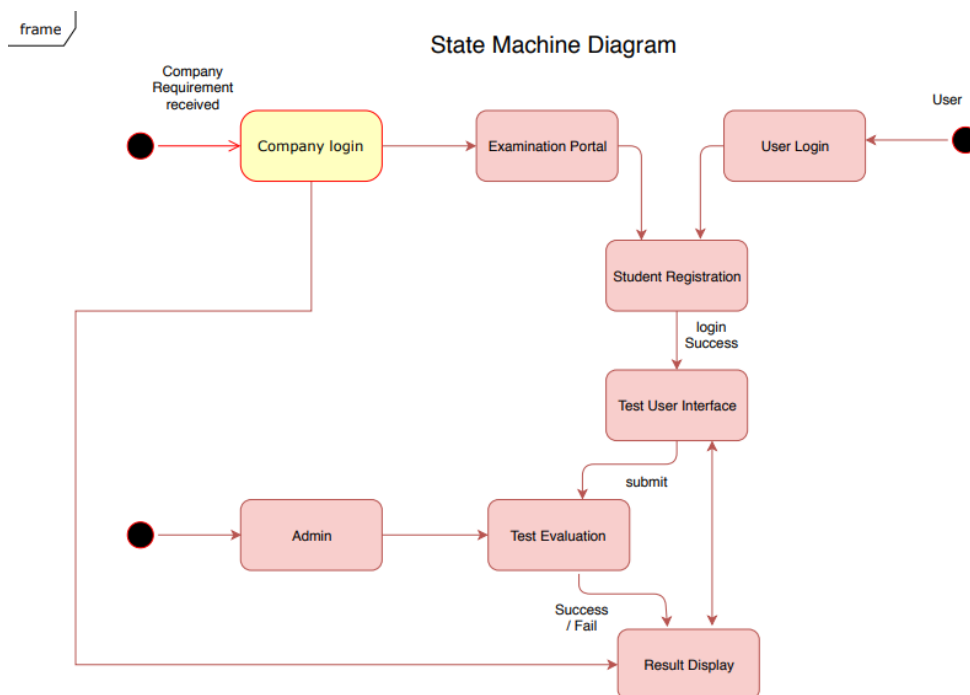


Figure G.1: State Diagram

- Methodologies of Problem solving:

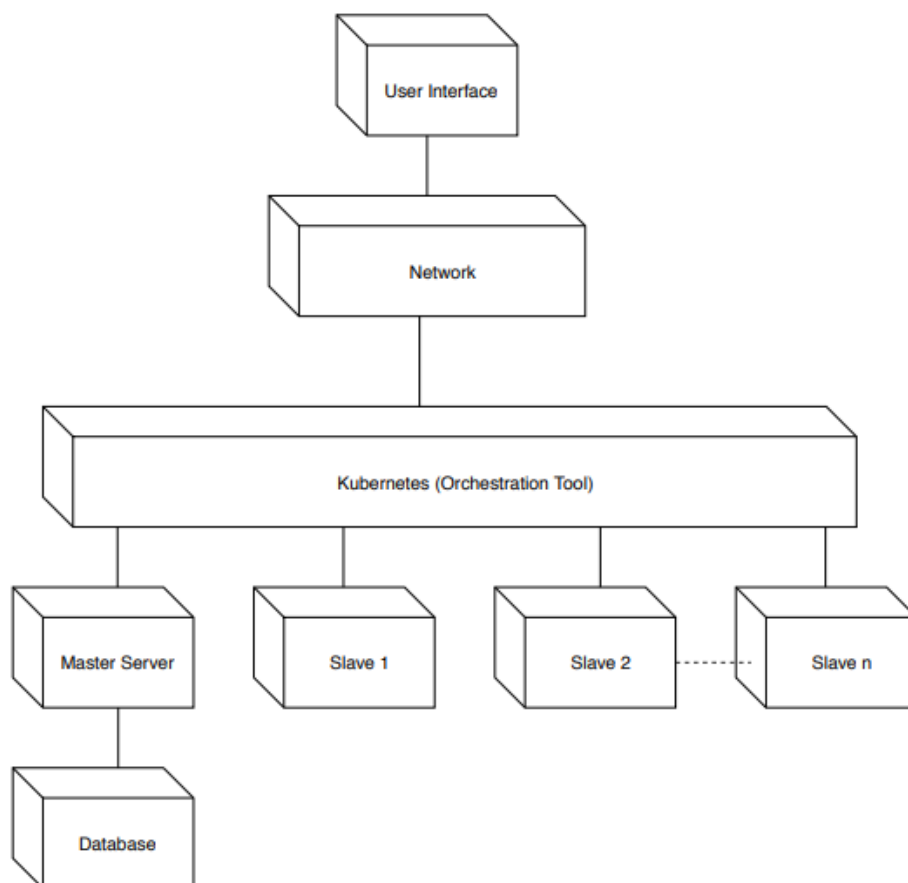
1. **Configuration Management:** Configuration management solves the problem of manually install and configure packages once the hardware is in place. The benefit of using configuration management solutions is that servers are deployed exactly the same way every time . If you need to make a changes across 100 thousand servers you only need to make the change in one system . In the operations environments we have worked in there were always strict controls on who could access production environment , who could make change, when changes could be made, who could physically touch hardwares , and who could access what data centers . In



these highly regulated and process oriented enterprises the thought of blurring the lines between development and operation seems like a non-starter.

2. **Kubernetes container orchestration** Containers support VM-like separation of concerns but with far less overhead and far greater flexibility. As a result, containers have reshaped the way people think about developing, deploying, and maintaining software. In a containerized architecture, the different services that constitute an application are packaged into separate containers and deployed across a cluster of physical or virtual machines. But this gives rise to the need for container orchestrationa tool that automates the deployment, management, scaling, networking, and availability of container-based applications.

**Deployment Diagram**



**Figure G.2:** Kubernetes container orchestration

- **Publication/presentation details with review report**

**Journal Name:** Computer Society of India (CSI) Communications.

**Paper Name:** Cloud Based DevOps Skill Assessment Application

**Author Name:** Roshan Patil,Chetan Pawar,Yogesh Mahajan,Nishant Kumar

**Impact Factor:** 6.2

**Paper Status:** Published

**Assignment No.: 2:**

- **Problem statement:** Project Workstation selection, Installations details with setup and Installation procedure.
- **System and Hardware Requirements :** Operating System: Ubuntu any distro  
IDE: Pycharm  
Programming Language: Python3  
Framework : Django  
Packages :Terminado ,Tornado Websocket,crispyforms,  
Scripting Language : Bash,CSS,Bootstrap4,HTML5  
Database: MySql  
Technologies Used: Docker,Kubernetes
- **Installation details: Docker Installation:**

- Update the apt package index:

```
1 sudo apt-get update
```

- Install packages to allow apt to use a repository over HTTPS:

```
1 sudo apt-get install /
2 apt-transport-https /
3 ca-certificates /
4 curl /
5 gnupg2 /
6 software-properties-common
```

- Add Dockers official GPG key:

```
1 curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-
  key add
2 sudo apt-key fingerprint 0EBFCD88 pub 4096R/0EBFCD88 2017-02-22 Key
  fingerprint = 9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF
  CD88 uid Docker Release (CE deb) <docker@docker.com> sub 4096R/
  F273FCD8 2017-02-22
```

- Use the following command to set up the stable repository

```
1 sudo apt-get install docker-ce docker-ce-cli containerd.io
```

**Kubernetes installation**

### – Installing kubeadm, kubelet

```
1 apt-get update && apt-get install -y apt-transport-https curl curl
  -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-
  key add - cat <<EOF >/etc/apt/sources.list.d/kubernetes.list
  deb https://apt.kubernetes.io/ kubernetes-xenial main EOF apt-
  get update apt-get install -y kubelet kubeadm kubectl apt-mark
  hold kubelet kubeadm kubectl systemctl daemon-reload systemctl
  restart kubelet
```

### – Install kubectl on Linux:Ubuntu

```
1 sudo apt-get update && sudo apt-get install -y apt-transport-https
  curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg |
  sudo apt-key add - echo "deb https://apt.kubernetes.io/
  kubernetes-xenial main" | sudo tee -a /etc/apt/sources.list.d/
  kubernetes.list sudo apt-get update sudo apt-get install -y
  kubectl
```

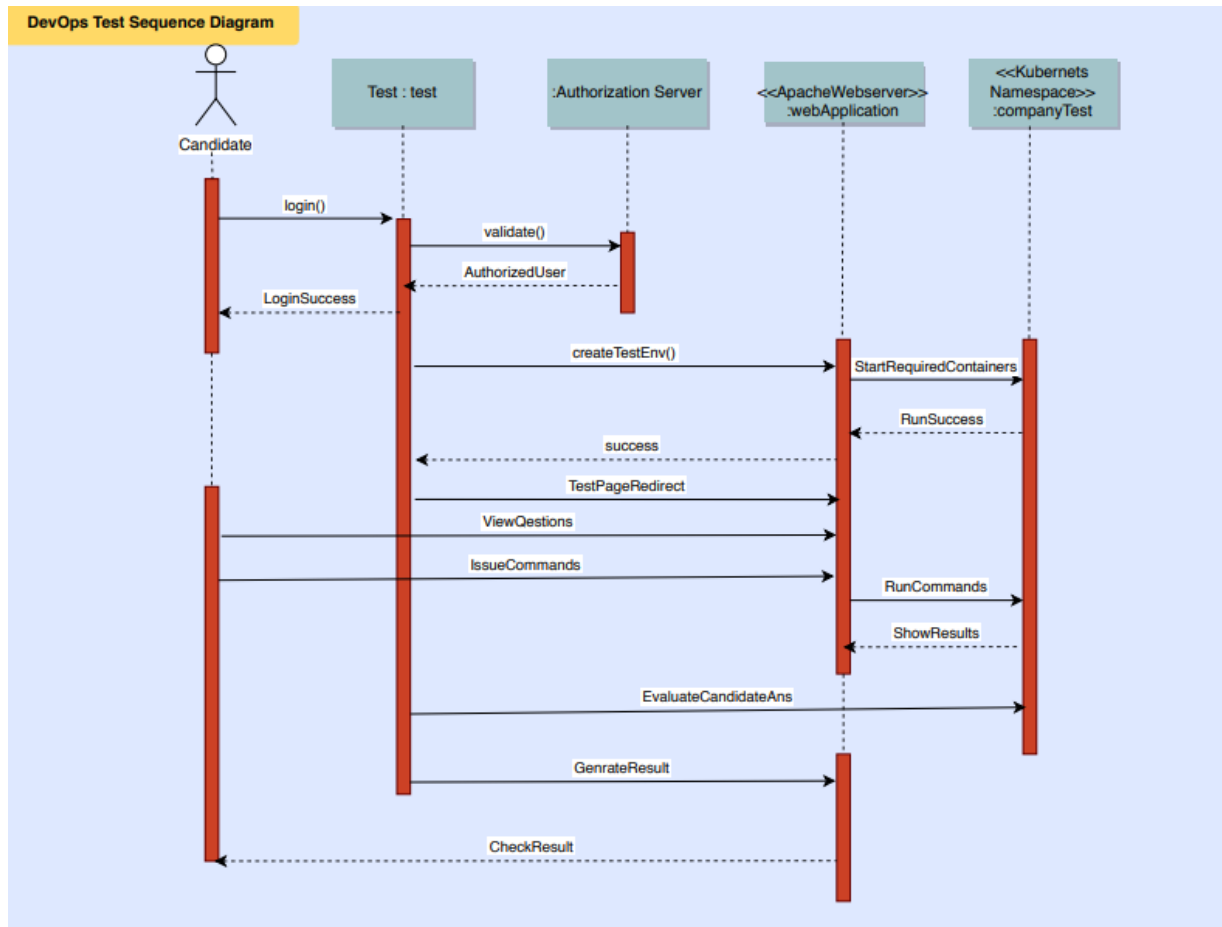
**Assignment No. 3:**

- **Problem statement:**

Programming of project functions as per UML diagram mention in earlier submission.

**UML Diagram:**

- **Sequence Diagram**



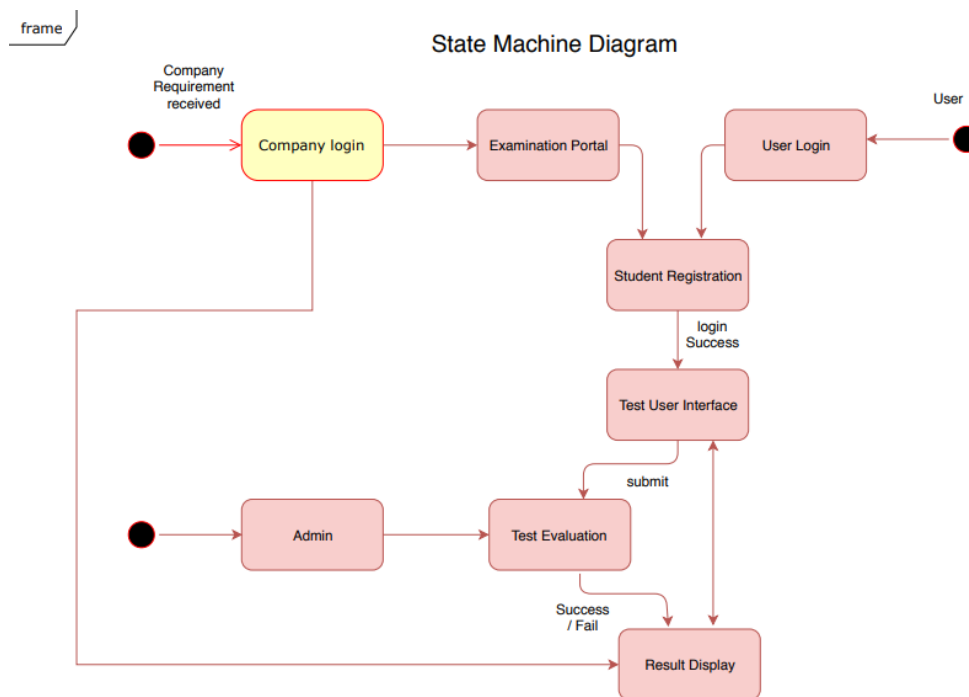
**Figure G.3:** Sequence Diagram

Lifeline represents typical instances of the components or classes in our system. Messages are shown as arrows. They can be complete, lost or found; synchronous or asynchronous; call or signal. Activate is used to denote participant activation. Once a participant is activated, its lifeline appears. Objects are model elements that represent instances of a class or of classes. Classes in UML show architecture and features of the designed system. Actor specifies a role played by a user or any other system that interacts with the subject. Candidate logs in to the system and lifeline gets activated. Candidate can give exam using the test instance. The Authorization server issues commands given by

the candidate to actual running container. The candidate and the actual running container are isolated from each other. They communicate using a webserver. After each completion of task as

per the question, the page is redirected to the next question. Once the test is completed. A request for test generation is passed to Company test portal to generate the result.

### • State Diagram

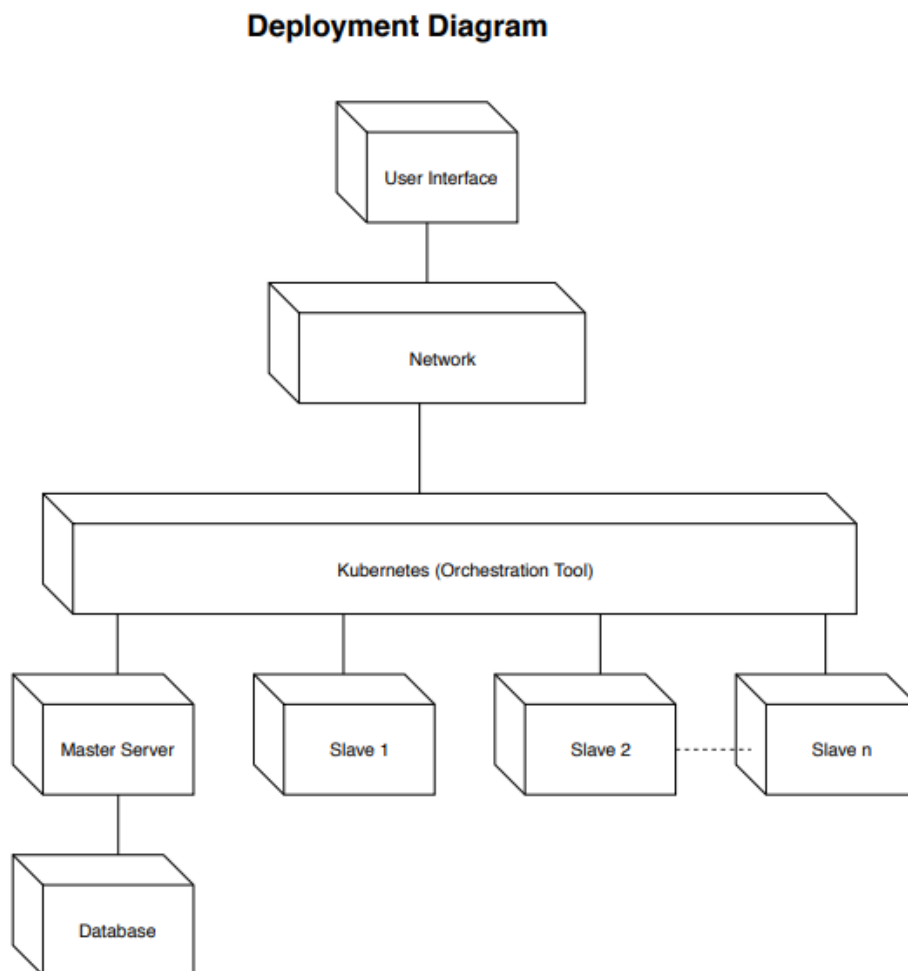


**Figure G.4:** State Diagram

A state is a condition during the life of an object during which it satisfies some condition, performs some activity, or waits for some external event. A start state is the state that a new object will be in immediately following its creation. An end state is a state that represents the object going out of existence. A transition is a relationship between two states indicating that an object in the first state will perform certain actions and enter the second state, when a specified set of events and conditions are satisfied. Candidate logs in to the framework and lifeline of object gets initiated. Candidate can give test utilizing the test occurrence. The Authorization server issues directions given by the candidate to real running compartment. The candidate and the genuine running compartment are detached from one another. They convey utilizing a webserver. After every fulfillment of undertaking according to the inquiry, the page is diverted

to the following inquiry. When the test is finished. A solicitation for test age is breezed through to Company test entryway to produce the outcome.

- **Deployment Diagram**



**Figure G.5:** Deployment Diagram

Nodes represent either hardware devices or software execution environments. They could be connected through communication paths to create network systems of arbitrary complexity. Component represents a modular part of a system. A component defines its behavior in terms of provided and required interfaces. Dependency relationship is a relationship in which one element, the client, uses or depends on another element, the supplier.

**Master node:** Runs multiple controllers that are responsible for the health of the cluster, replication, scheduling, endpoints (linking Services and Pods), Kubernetes API, interacting with the underlying cloud providers etc. Generally it makes sure everything is running as it should be and looks after worker nodes.

**Worker node (minion):** Runs the Kubernetes agent that is responsible for

running Pod containers via Docker or rkt, requests secrets or configurations, mounts required Pod volumes, does health checks and reports the status of Pods and the node to the rest of the system.

**Pod:** The smallest and simplest unit in the Kubernetes object model that you can create or deploy. It represents a running process in the cluster. Can contain one or multiple containers.

**Deployment:** Provides declarative updates for Pods (like the template for the Pods), for example the Docker image(s) to use, environment variables, how many Pod replicas to run, labels, node selectors, volumes etc.

**DaemonSet:** Its like a Deployment but instead runs a copy of a Pod (or multiple) on all (or some) nodes. Useful for things like log collection daemons (sumologic, fluentd), node monitoring daemons (datadog) and cluster storage daemons (glusterd).

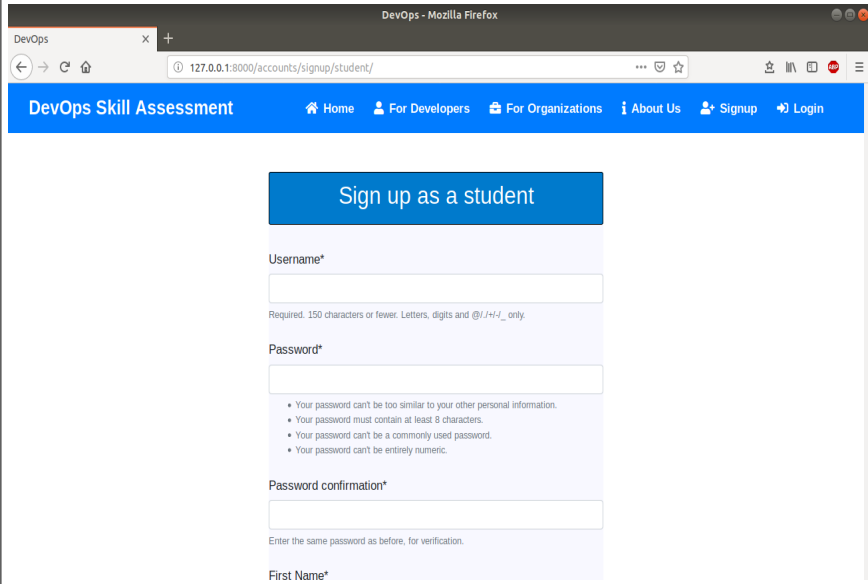
**ReplicaSet:** Controller that ensures a specified number of Pod replicas (defined in the Deployment) is running at any given time.

**Service:** An abstraction which defines a logical set of Pods and a policy by which to access them (determined by a label selector). Generally its used to expose Pods to other services within the cluster (using targetPort) or externally (using NodePort or LoadBalancer objects).

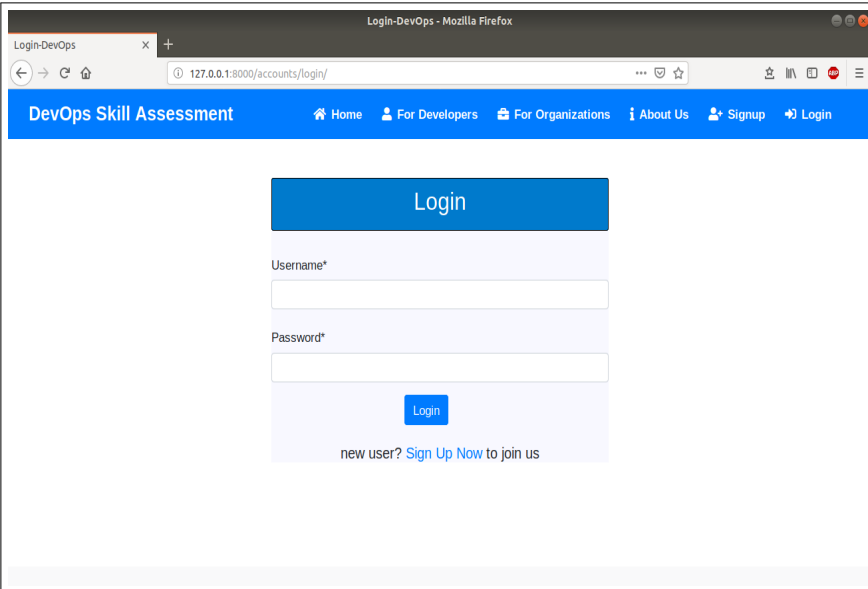


**Assignment No. :4**

- **Title:** Interface and GUI of project work, after necessary corrective action taking as per earlier submission.



The screenshot shows a web browser window titled "DevOps - Mozilla Firefox" with the address bar displaying "127.0.0.1:8000/accounts/signup/student/". The page has a blue header with the text "DevOps Skill Assessment" and navigation links: Home, For Developers, For Organizations, About Us, Signup, and Login. The main content area features a blue button labeled "Sign up as a student". Below this button are input fields for "Username\*", "Password\*", and "Password confirmation\*", each with a light blue border. The "Username\*" field has a small text requirement below it: "Required. 150 characters or fewer. Letters, digits and @/./+/-/\_ only." The "Password\*" field has a list of password requirements: "Your password can't be too similar to your other personal information.", "Your password must contain at least 8 characters.", "Your password can't be a commonly used password.", and "Your password can't be entirely numeric." The "Password confirmation\*" field has a note: "Enter the same password as before, for verification." At the bottom of the form is a "First Name\*" input field.

**Figure G.6: Sign Up**

The screenshot shows a web browser window titled "Login-DevOps - Mozilla Firefox" with the address bar displaying "127.0.0.1:8000/accounts/login/". The page has a blue header with the text "DevOps Skill Assessment" and navigation links: Home, For Developers, For Organizations, About Us, Signup, and Login. The main content area features a blue button labeled "Login". Below this button are input fields for "Username\*" and "Password\*", each with a light blue border. Below the "Password\*" field is a blue "Login" button. At the bottom of the form, there is a link: "new user? [Sign Up Now](#) to join us".

**Figure G.7: Login**

The screenshot shows a web browser window titled 'DevOps - Mozilla Firefox' with the address bar displaying '127.0.0.1:8000/business/test/create/'. The page features a blue header with the text 'Create Test'. Below the header, there is a form with a 'Name\*' field containing the text 'Amazon pool campus'. Underneath, a 'Questions\*' section lists several tasks with checkboxes: 'Create Directory' (checked), 'Bash Text File Creation' (unchecked), 'Change Port of Apache Server' (unchecked), 'Single User File Access' (checked), 'tmpfs File System Creation' (unchecked), 'File Copy' (checked), 'File Move' (unchecked), 'My-Sql Installation' (checked), 'Check for File Write Mode' (unchecked), 'Create File in Read Mode' (checked), and 'Create File in Executable Mode' (unchecked). A green 'Create' button is located at the bottom right of the form.

Figure G.8: Create Test

The screenshot shows a web browser window titled 'DevOps - Mozilla Firefox' with the address bar displaying '127.0.0.1:8000/business/test/35/'. The page has a blue header with the text 'DevOps Skill Assessment' and navigation links: 'Home', 'For Developers', 'For Organizations', 'About Us', 'AJAY', and 'Logout'. Below the header, there is a blue section titled 'Test Details'. This section displays 'Test ID : 35' and 'Test Name : Amazon pool campus'. A 'Questions :' section lists five tasks: '1. Create Directory', '2. Single User File Access', '3. File Copy', '4. My-Sql Installation', and '5. Create File in Read Mode'. A 'View Result' button is located on the right side of the page.

Figure G.9: Test Details

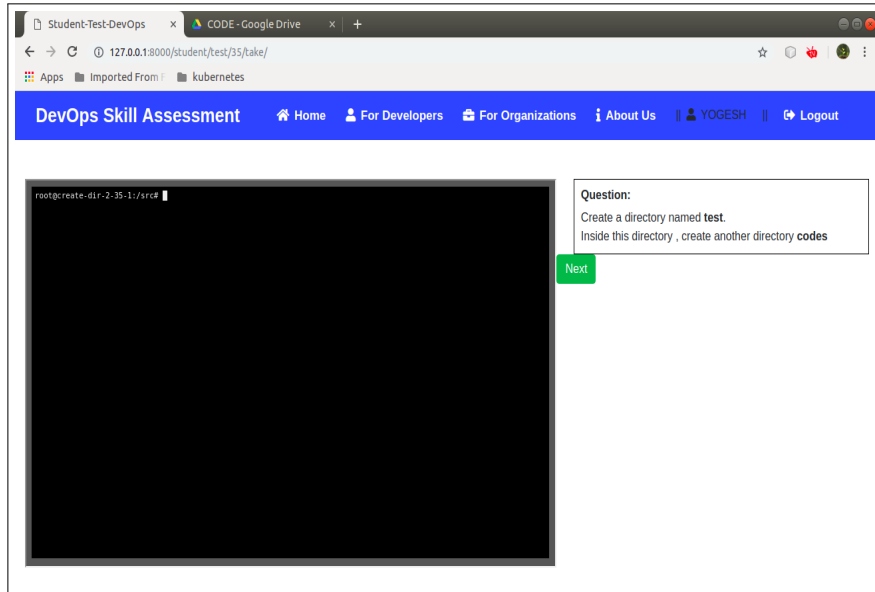


Figure G.10: Ongoing Test

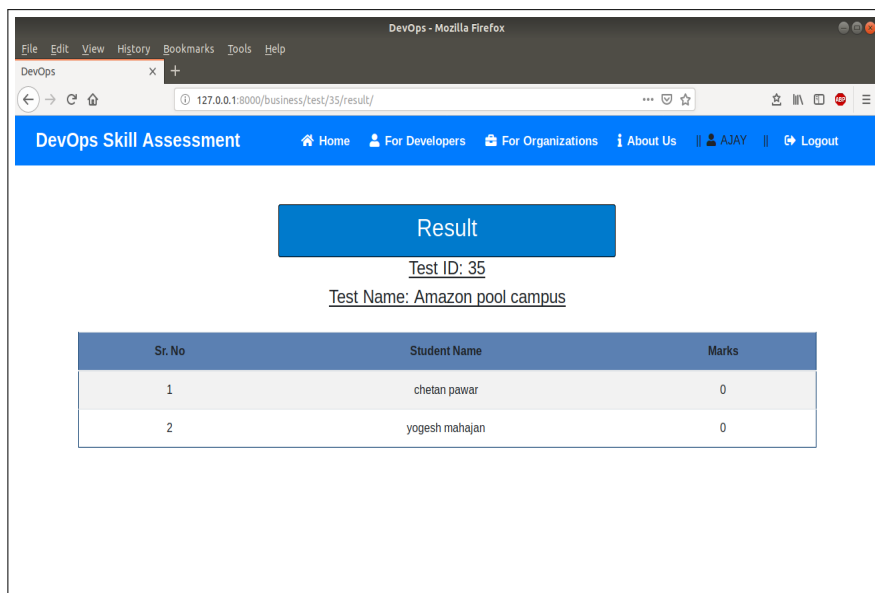


Figure G.11: Test Result

**Assignment No.: 5**

- Title: Selection of tools and various testing cases for project perform.

- **Testing in DevOps:**

DevOps is a set of practices that highlights collaboration, communication, and automation throughout the application development lifecycle. Continuous Delivery allows teams to release a build after continuous testing automatically any time. The continuous repeatable release of software ensures speedy delivery of stable quality software to end user. Automation has an essential role to play in both DevOps and Continuous Delivery. Automating DevOps and Continuous Delivery helps your organization provide continuous delivery to the users faster and manage a business with low risk. Testing needs to be fast and dynamic to match with the goals of DevOps and Continuous Delivery. With traditional testing approaches and tools, this is a seemingly difficult task. And, without the right testing approach, it will be difficult to meet the DevOps and Continuous Delivery standards.

**Impact of DevOps on Testing**

Testers are now expected to have at least the following knowledge and skills to be able to effectively carry out testing activities Basic networking knowledge Basic Unix/Shell scripting, e.g. bash, python Continuous Integration/Continuous Delivery e.g. Jenkins, Performance testing tools such as JMeter or Gatling Ready for Operations and Resilience Testing Knowledge of containers, Docker, Kubernetes Querying monitoring tools such as Splunk Clouds services, e.g. AWS, Azure

**Tool for Testing:** The right automation testing tool must be used after evaluating the capabilities of the tool closely. It should be capable of meeting end-to-end automation testing requirements. Thus make the test automation program really effective in the agile environment. The right automation testing tool must be used after evaluating the capabilities of the tool closely. It should be capable of meeting end-to-end automation testing requirements. Thus make the test automation program really effective in the agile environment. Automated testing tools that can help ship code and deliver quality software on time. Selenium is one of the open source tool used in testing.

**(1) Selenium**

One of the main reasons for its widespread use is that it is open source. There are a number of frameworks implemented in different languages using Sele-

nium, which offer more capabilities in automation testing.

### 1) Signup

<b>Test cases</b>	Registration Screen Signup
<b>Objective</b>	Click on sign up button then check all required/ mandatory fields with leaving all fields blank
<b>Expected Result</b>	All required/ mandatory fields should display with symbol * Instruction line * field(s) are mandatory should be displayed

### 2)Test Environment

<b>Test cases</b>	Test Environment Creation
<b>Objective</b>	Selection of Examination questions
<b>Expected Result</b>	Test questions selected successfully and get respective test id

### 3)Login

<b>Test cases</b>	Candidate login to give test
<b>Objective</b>	Pods should deploy and user should get terminal
<b>Expected Result</b>	Pods deployed successfully and user got terminal

### 4)Test Submission

<b>Test cases</b>	Test submission
<b>Objective</b>	Check respective command can execute in terminal
<b>Expected Result</b>	Move to the next question or test completed successfully

### 5)Test Evaluation

<b>Test cases</b>	Test Evaluation
<b>Objective</b>	Evaluation script should run in pod and return some value
<b>Expected Result</b>	Candidate get marks for respective question

### 6)Test Result

<b>Test cases</b>	Result declaration
<b>Objective</b>	Marks of the candidate should store in database in descending order
<b>Expected Result</b>	Candidate and organization both can see the result and candidate with higher marks must be in top of the list

**Figure G.12:** Selenium Testing

**Assignment No.: 6**

- **Problem statement:** Result as per various test cases perform for project work and submit result charts, table of result and graph including reliability testing.
- **Solution:**

**Test Chart**

<b>Test no</b>	<b>Test Case</b>	<b>Expected Result</b>	<b>Comment</b>
1	Check for Validation / Error Messages on all the Screen	Validation error messages Shall be displayed properly at correct position	Pass
2	Check for all the Fields Label on all Forms	Field labels Shall be standard e.g. field accepting users first name should be labeled properly as First Name	Pass
3	Check functionality of buttons available on all pages.	All Buttons on the Forms should be Functional	Pass
4	Check all the Fields on the Page/ Forms	All fields on page (e.g. text box, radio options, dropdown lists) shall be aligned properly	Pass
5	Check if correct data is getting saved in database upon successful page submit	Correct data shall be saved in database	Pass

Table 1: Test Cases

## Appendix H

### Information of Project Group Members



1. Name : Yogesh Pradeep Mahajan
2. Date of Birth : 31/03/1997
3. Gender : Male
4. Permanent Address :9, Laxmi Narayan R/H, Ashwin Nagar, CIDCO, Nashik,422009
5. E-Mail : mahajan.yogesh205@gmail.com
6. Mobile/Contact No. : 9011480011
7. Placement Details : Placed at MiniOrange
8. Paper Published : Computer Society Of India



1. Name : Roshan Dinesh Patil
2. Date of Birth : 30/04/1997
3. Gender : Male
4. Permanent Address :3-A,Raisoni nagar,Mahabal,Jalgaon
5. E-Mail : roshanpatil3004@gmail.com
6. Mobile/Contact No. : 8888872919
7. Placement Details : Placed at Vignet
8. Paper Published : Computer Society Of India





1. Name : Chetan Kisan Pawar
2. Date of Birth : 04/10/1997
3. Gender : Male
4. Permanent Address : A/P-Walhe(Bahirjichiwadi), Tal-Purandar, Dist-Pune-412305
5. E-Mail : chetanpawar968@gmail.com
6. Mobile/Contact No. : 9767340536
7. Placement Details : Placed at Amdocs
8. Paper Published : Computer Society Of India



1. Name : Nishant Kumar
2. Date of Birth : 13/01/1997
3. Gender : Male
4. Permanent Address : S/O S K Suman, old durga sthan, vill-Mohanpur, dict-Samastipur 848506
5. E-Mail : phy.nishant@gmail.com
6. Mobile/Contact No. : 9504088709
7. Placement Details : Placed at Amdocs
8. Paper Published : Computer Society Of India

# **Appendix I**

## **Publication Details**

# Cloud Based DevOps Skill Assessment Application

1<sup>st</sup> Yogesh Mahajan  
School of computer engineering and  
technology  
MIT academy of engineering  
Pune, India  
mahajan.yogesh205@gmail.com

4<sup>th</sup> Nishant Kumar  
School of computer engineering and  
technology  
MIT academy of engineering  
Pune, India  
phy.nishant@gmail.com

2<sup>nd</sup> Roshan Patil  
School of computer engineering and  
technology  
MIT academy of engineering  
Pune, India  
roshanpatil3004@gmail.com

5<sup>th</sup> Amar More  
School of computer engineering and  
technology  
MIT academy of engineering  
Pune, India  
ahmore@comp.maepune.ac.in

3<sup>rd</sup> Chetan Pawar  
School of computer engineering and  
technology  
MIT academy of engineering  
Pune, India  
chetanpawar968@gmail.com

**Abstract**—Now a days many online tools are available to test the programming knowledge of the person like codechef. But in order to test the knowledge of the DevOps there is no such online tool available. So the goal is to develop the cloud based infrastructure to test the knowledge of DevOps of the examinee. The questions related to the DevOps will be given to the candidate along with the access to the terminal. The candidate has to do all the steps required to take care of the issue given. The terminal Provided to the candidate is the correspondence interface between the candidate and the allotted container. We are using containers rather than VMs, because containers are small, light-weighted and what's more, quick, one application can be pressed in every container image. The Kubernetes will deal with the containerized applications, for example, database stockpiling and client explicit order over a lot of containers or has and gives systems to sending, support, and application-scaling. The container runtime bundles, instantiates, and runs client directions on containerized application. The yield produced will be put away in an impermanent document which will be checked with the ideal yield put away in a database.

**Keywords**— DevOps, Orchestration, Containerization, Linux, Cloud computing, Parallel Computing, Networking.

## I. INTRODUCTION

DevOps is an item progression framework that combines programming improvement with data advancement assignments. Its objective is to abbreviate the frameworks improvement stages while additionally conveying fixes, highlights, and updates every now and again in close arrangement. The DevOps approach is to incorporate robotization and occasion observing at all means of the product advancement. The attention on the engineer cooperation empowers another way to deal with dealing with the multifaceted nature of certifiable issues. It trusts the tasks multifaceted nature stalls into a couple of classifications: setup the executives, foundation and arrangement mechanization, log and execution the board, and checking. The following are a few apparatuses we have used to help explain these errands. As a component of Agile changes as in the paper "Understanding DevOps and conquering any hindrance from continuous integration to continuous conveyance" [1] in recent years we have seen IT associations embracing continuous integration standards in their programming projects conveyance lifecycle, which has improved the effectiveness of advancement groups. With the time it has been understood that this advancement as a component of ceaseless joining alone is simply not adequate to make the whole conveyance lifecycle proficient or isn't driving the associations efficiency. This paper attempts to

cover all parts of DevOps pertinent to different periods of SDLC and explicitly discusses business need, approaches to move from consistent combination to nonstop conveyance and its advantages.

## II. LITERATURE REVIEW

Number of literatures pertaining techniques to DevOps and cloud computing published already and are available for public usage. According to Wotif Group [2] used DevOps standards to recuperate from the descending spirals of manual discharge exercises that numerous IT offices face. Its methodology included the idea of making it productive to do the working thing. By characterizing the correct things (organization measures) for advancements and tasks groups and making it simple to receive, Wotif radically and viably improved the normal discharge process duration. Containerization is a lightweight virtualization arrangement.

The paper presented in conference 2017 Seventeenth International Conference on Advances in ICT [3], DevOps is stretched out from certain deft practices with a blend of examples planned to improve joint effort among advancement and task groups.

The principal explanation behind this paper is to direct an examination on how DevOps practice has affected to programming quality. The optional goal is to find how to improve quality efficiently. Automation is the most critical factor to improve the software quality. As per the results of multiple regression analysis, it has proved culture, automation, measurement and sharing are important factors to consider to improve quality of the software.

In the paper [7] Explores the market opportunities for cloud computing in India. The principal explanation behind this Cloud Computing is another worldview in data innovation (IT) and IT-empower services (ITES) that change processing as an asset to figuring as an administration. It is a problematic innovation with impact infesting over all part of a cutting edge economy.

In the paper presented in "Euromicro Conference on Software Engineering and Advanced Applications" [4] DevOps and continuous practices are attracting steadily growing attentions by both practitioners and researchers in the software engineering community. The terms are regularly utilized conflictly, reciprocally and with misty significance, in any case. By taking the positions that, this equivocality and miscommunication renders the network incredible mischief, their belongings and interaction between them, decrease uncertainty.

GNU/Linux shell access through an internet browser for an inserted Linux e-learning framework [5] speaks to The web is developing quickly and more affects the training area than it had ever previously. This paper goes for depicting a novel method to expand the e-Learning procedures used to the zone of implanted Linux training. For these strategies, Open Source Software advancements, for example, AJAX, PHP, Apache have been utilized in an inventive manner to bestow inserted Linux instruction just by utilization of typical internet browsers which make the learning framework as simple as checking a mail. In this paper, methods for empowering the implanted gadgets Linux shell access through an internet browser is clarified, which is a key component of the framework. This improves adaptability and availability for numerous clients.

The paper addressed in Establish new plan to make evaluation course of action of examination questions and examination result. [6] This paper means to set up a sensible, objective, quantized assessment standard of breaking down examination and score, and build up the assessment list arrangement of examination questions and examination result dissecting. A ton of sensible and target thoughts, for example, examination trouble, gauge score, target trouble, etc are risen and characterized, and some related quantized estimation strategies are given, and the examination result breaking down and examination rule programming framework which can settle on a quantized choice is modified.

" Containerization and the PaaS Cloud " [9] center around Containerization is a lightweight virtualization arrangement. Aside from displaying benefits over customary virtual machines in the cloud, holders are particularly pertinent for PaaS mists to oversee and organize applications through compartments as an application bundling component.

" Task Based Automatic Examination System for Sequenced Test " [8] speaks to Computer extraordinarily impacts our instructive condition. Throughout the latest couple of years, online programmed PC examination frameworks have been generally utilized for PC based tests, however these frameworks depend on conventional inquiry answer examination style which isn't fit for the sequenced test. The sequenced test ought to consider the setting of the examinee.ex. the request of inquiries or the consents of the examinee, to review an examinee. This paper propose a compelling and viable programmed examination design dependent on assignment. The undertaking is to extract from the examination procedure and meet the solicitations of the sequenced test, for example, request and reliance.

Our Project is having appropriated condition. We are utilizing DevOps Tools, for example, Kubernetes for age of ace and slave nodes. System will perform task of the evaluation of skills of the candidate.

### III. METHADODOLOGY

#### A. Configuration Management

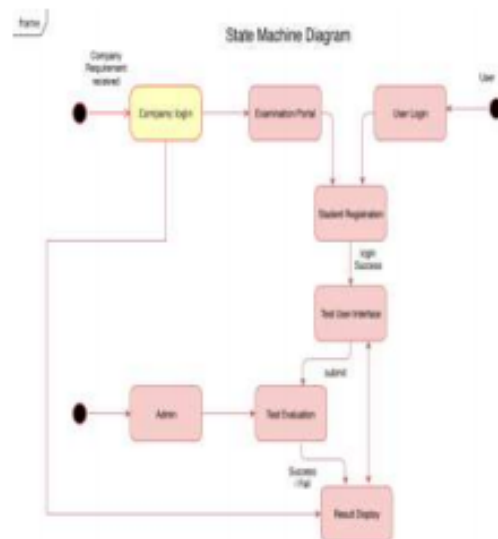
Configuration management tackles the issue of physically introduce and design bundles once the equipment is set up.

The advantage of utilizing configuration management arrangements is that servers are conveyed the very same way every time . In case you have to roll out an improvements crosswise over 100 thousand servers you just need to roll out the improvement in one framework . In the activities conditions we have worked in there were constantly exacting controls on who could get to generation condition , who could make change, when changes could be made, who could physically contact fittings , and who could get to what data centers . In these very directed and process situated undertakings the prospect of obscuring the lines among improvement and task appears like a non-starter.

#### B. Working of Kubernetes

To work with Kubernetes, we use Kubernetes API items to depict our bunches wanted state: what application or different outstanding tasks at hand we need to run, what container images we use, the quantity of imitations, what system and disk resources we need to make accessible, and more. The Kubernetes Master is a gathering of three procedures that keep running on the single node in our group, which is assigned as a master node. Those processes are: kube-apiserver, kube-scheduler and kubecontroller-manager. Each individual non-master node in our cluster runs two processes kube-proxy kublet.

#### C. Architectural Design



### IV. MATHEMATICAL MODEL

Input: X = Answers for the questions in the test.

• Output: Y = Relative rank of candidate according to his performance in the test.

Let  $S = \{U, N, T, S, \text{status}, \text{result}, F\}$

Where, •  $U = \{u_1, u_2, u_3, \dots, u_i\}$  Finite set of users (Candidates).

•  $N = \{ \text{Master}, \text{Slave} \}$



Where, Master = Kubernetes master node.

Slave = Kubernetes worker node.

- Slave = { slave1, slave2, .., slave i }
- T = { t1, t2, t3, .., ti } Finite set of test Scenarios.
- S = { s1, s2, s3, .., si } Finite set of scores of a user.
- status = { status1, status2, .., status i } Test status.
- Functionalities: Yes/No = authenticate (uname, passwd)
- Interface Candidate Web Brower (Wi)
- Si = get test Scenario Score(ti)
- Rank = apply ranking algorithm (ti, si)
- Result = generate test result.

#### V. CONCLUSION

Organizations that consolidate the DevOps activities to accomplish increasingly, plain and straightforward. With a single gathering made out of cross-valuable people all working in collaboration, DevOps structures can pass on the most extraordinary speed, value, and advancement. Companies that utilize this application will be effectively ready to make test contingent upon their particular prerequisites. This test scenarios will help company to easily evaluate candidate knowledge and thus they can find a right candidate for the desired post

#### ACKNOWLEDGMENT

We would like to express our deep sense of gratitude and respect towards our guide Prof. Amar More, School of Computer Engineering, MIT Academy of Engineering. We have received from him while collecting data on this paper and throughout our studies. We whole-heartedly thank to our Dean Mrs. Ranjana R. Badre for their guidance. We also indebted to all Sr. Engineers and others who gave us their valuable time and guidance. The various information and sources we used during my report completion find place in our report.

#### REFERENCES

- [1] Manish Virmani, "Understanding DevOps & bridging the gap from continuous integration to continuous delivery" Fifth International Conference on the Innovative Computing Technology, 20-22 May 2015, Pontevedra, Spain. IEEE, 2015. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/7173368> [Accessed: 03 August 2015].
- [2] Matej Artac, Tadej Borovssak, Elisabetta Di Nitto, Michele Guerriero, Damian Andrew Tamburri, "DevOps: Introducing Infrastructure-asCode" 39th International Conference on Software Engineering Companion (ICSE), 20-28 May 2017, Buenos Aires, Argentina. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/7965401> [Accessed: 24 August 2017].
- [3] Pulasthi Perera, Roshali Silva, Indika Perera, "Improve software quality through practicing DevOps, 2017 Seventeenth" International Conference on Advances in ICT for Emerging Regions, 6-9 Sept. 2017, Colombo, Sri Lanka. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/8257807> [Accessed: 15 January 2018].
- [4] Daniel Stahl, Torvald Martensson, Jan Bosch, "Continuous practices and devops: beyond the buzz, what does it all mean?" 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 30 Aug.-1 Sept. 2017, Vienna, Austria. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/8114695> [Accessed: 20 November 2017].
- [5] Suchakrpani Datt Sharma, D. N. Sonawane, Tanushri Chakravorty, Tushar Patil, "GNU/Linux shell access through a web-browser for an embedded Linux e-learning system" 3rd International Conference on Electronics Computer Technology, 8-10 April 2011, Kanyakumari, India. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/5941713> [Accessed: 07 July 2011].
- [6] Zhang Ruilan, Hua Jing, Liu Yingjia, An Weili "Establish new concept to develop evaluation system of examination questions and examination result" 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 8-10 Aug. 2011, Dengfeng, China. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/6009959> [Accessed: 06 September 2011].
- [7] "Cloud Computing Innovation in India: A Framework and Roadmap - White Paper 2.0" 31 Dec. 2014, Pontevedra, Spain. IEEE, 2015. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/7039186>
- [8] Song Luo, Jianbin Hu, Zhong Chen, "Task Based Automatic Examination System for Sequenced Test" International Conference on Electronic Computer Technology, 20-22 Feb. 2009, Macau, China. Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/4795912> [Accessed: 27 February 2009].
- [9] Claus Pahl, "Containerization and the PaaS Cloud", IEEE Cloud Computing, Volume: 2, Page(s): 24 - 31, Issue: 3, May-June 2015, Available: IEEE Xplore, <https://ieeexplore.ieee.org/document/7158965>. [Accessed: 15 July 2015].

## Certificates:



