INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal - 500 043, Hyderabad, Telangana

Examinations Control Office

Examination	B TECH VI SEMESTER END EXAMINATIONS REGULAR JUNE 2025 REG UG20		
Month & Year	1-Jun	**Date**	20/06/2025
Course Name	DEVOPS		
Course Code	ACSC42	**E-Code**	6701

Instructions to Evaluators

- Evaluators should spend at least 3-5 minutes on one answer booklet during the evaluation.
- Evaluators should cross check that marks are allotted for all the attempted questions.
- The marks should be assigned fairly according to the mark distribution specified in the scheme of evaluation.
- For questions that were attempted incorrectly, evaluators are required to award zero marks.
- The evaluator must give a proper justification in case of any mistakes identified in the marks provided.

START WRITING FROM HERE

Q.No.

2a> Ans:

In developing a real world projects we need a team to help each other and look out for each other while developing a real time project.
By working in a team there would be more quality checks, the project would be more refined and most importantly the work load would be divided / amount of work done by a person would be divided among teammates.
Now for every person to work on the project some kind of environment is required. This is where online directiores come into place.
The most commonly used DIRECTORIES are:

1> SVN

2> Git

1> SVN:

SVN uses a centralized architecture. Which means that every person who has access to that directory can make

Q.No.

changes which would be reflected on actual code / product.
This directory becomes a headache when multiple users are trying to use the code.
There is alot of work / It is quite hard to test some test cases in this architecture.
Suddenly switching tools in between the project is quite hard

2) Git:

Git uses a decentralized architecture. Which means when trying to use a code / directory it actually clones the directory where you can work on your project and make changes on it (clone) without worrying that some one can change your idea.
Using Git becomes easy when there are multiple users on your team and each have a unique idea. We can / Everyone can work

Q.No.

on their idea without actually making changes on original code. This makes more number of choices to ~~choice~~ to choose one idea which is more user friendly, reliable and efficient idea from all the ideas of team members Before deploying it on to the web. One can comparatively find it easy to switch tools for version control and ~~collabra~~ collaborate with other techs easily using Git.

Over all both techs have their own advantages and disadvantages.
It depends on the user needs and type of users to use which technology they prefer.
Both have their own unique architectures. They both have their own way of doing things.

Q.No.

3a) Ans:- **Dockers**

A docker is a tool / a type of software which is used for generating containers.
These containers are then used to switch services.
Eg: Migrating an existing monolithic application to microservices using docker containers.
These containers can also be used to deploy code.
These containers can store data, etc.
There are various components that are present in docker. They are:

- Docker engine
- Docker hub
- Docker images
- Containers
- Networks and volumes.

Docker engine:

The docker engine is actually the main part of the docker. Here the main code which does

Q.No.

all the processes in a docker is present It acts as a heart to the docker.

- **Docker hub:**

Docker hub is an important component in the docker. This is where all the data is stored. We can also retrive the stored data here

- **Docker images:**

Docker can actually store images. By using docker images one can store, transfer and share images using docker images.

- **Containers:**

A container is like a jar filled with data or files. These files can be stored, transfered and accessed when ever needed using containers in docker.

Q.No.	

•Networks and volumes:

Networks are a type / medium by which docker files / containers can be transfered.

Volumes are a collection of containers / files present in docker which are all stored together.

In general, volumes contain same / similar / related data and files.

2b) Ans:

There are many ways to identify a critical bug in Git.

But the my prefered way would be through branching and merging to find that bug.

First is to divide each branch individually and check each branch individually.

After checking if you find an error in a branch then go to that branch and divide the code into different segements.

Now check each and every

Q.No.

segements, individually you can use white box testing also.
After finding the error fix the error.
Now run that branch individually and ~~run~~ check for errors.
If no errors are found then try to run it with another branch Like this slowly merge one one branch to the code and run the code.
After that combine all branches and run the code.
This is a lengthy process but it shows how each an every element in the code is working.
This helps the developer to update the parts or branches of code where every ~~neccesary~~ required.
It also teaches a lesson for developer to how to efficiently code and check your code for future projects.
By doing this the developer learns a lesson that the code should be organi-zed and comments are needed for every while, for loops and if-else statements

Q.No.

5a) Ans:

We use cloud provisioning and configuration management to write/create and manage chef cookbooks.

We use could services like AWS (Amazon Web Services), EC2 and Assure for cloud storage and management.

We use dockers to write recipes, define attributes and testing cookbooks before deployment.

We use CD or Continous deployment to automate the deployment process.

We try to automate the deployment process to have less human intervention. This is because if there is human intervention there would be bound ~~errors~~ to be errors while deployment. To minimize the risk of errors we try to automate not only deployment process but all the testing process to.

Steps to create and manage Chef cookbook

Q.No.

1) We use docker to write & create a container.

2) Then we use these containers to write recipes.

3) We define all the attributes we use while creating a docker.

4) We use different types of testing to test our cook book before deployment.

These different types of testing are:

1) White box testing:

This type of testing is done to check the code and check for errors in the code.

This testing is usually done by developers.

2) Black box testing:

This type of testing is done to check the outputs for given inputs whether the expected output matches actual output.

Q.No.

This type of testing is done by tester or can be automated by testing tools

3) Grey box testing:
This type of testing is a mix of both black box and white box testing.
It includes best of both worlds

7a) Ans: Creating a manual test plan for a complex web application is quite a hard task.
One needs to ensure the test cases are realistic and commonly occured during real time while writing all test cases.
Some of the key elements to include is lare:
1) Black-box testing
2) White-box testing
3) Grey box testing.
4) GUI testing

Q.No.

1) Black-box testing:-

This is a type of testing done by the tester.

By doing this testing one would ensure all test cases are passed during testing.

If any test case if failed required changes are made by developer.

This type of testing mainly checks the outputs of the code for given inputs.

2) White-box testing:-

This type of testing is done by the code developer.

By doing this testing one would ensure that the code is error free.

The errors in the code like logical error, syntax error are checked and corrected by the developer.

3) Grey-box testing:-

This type of testing can either be done by developer or tester.

This testing combines the best of both worlds of black-box and white-box

Q.No.

testing.

4) GUI testing:

This type of testing is done by the tester.

In this testing each and every UI component is checked.

The testing ensures the proper working of UI elements

The CSS style of web application is checked.

7b) Ans:- Creating a comprehensive test plan to ensure full coverage of new feature in your application. Testing your application in various ways:-

1) Unit testing.
2) Integration testing
3) GUI testing

1) Unit testing:-

This is a type of testing where each and every element is checked

Q.No.

individually

2) Integrated testing:-

This type of testing test whether each and every component in your application is properly connected to one another.

One can reduce errors and save time while maintaining consistance by using automated integration testing here.

3) GUI testing:

This testing checks each and every UI component used on the frontend to check each and every element is working as intented or not.

To finish it all we can included Black box testing and White box testing.

Black box testing:

This checks the outputs for the given inputs in the code. It mainly checks test cases.

Q.No.

White box testing:

This type of testing checks the errors present in the code. This testing is also called internal testing.

This testing is done by developers.

5b) Ans- Chef plays an important role in cloud provisioning and configuration management.

Chef makes the chefbooks.

The creation involves in writing recipies, defining attributes and testing cook-books before deployment.

Chef uses automated testing tools like automated integration tests to ensure data consistancy across tests.

Chef uses tools like Cd and Continous deployment to make the process of deployment error free and maintain consistancy across environments.

Continous deployment is a deployment tool to deploy data into server.

Q.No.

or cloud / web server; where one can deploy the data as soon as the data is ready for deployment.
One cannot stop and overlook each data set before it is deployed.

Cd is another tool where it always brings data ready for deployment. It does not deploy the data automatically. The testers reviewes the data and then proceeds to deploy the data.

10) Ans: Branching and merging in Git is a process where each data is divided into different sections.
These sections contain all its related code in one place.
These sections are called branches.
There is a main branch where each branch joins in the end.
Merging is the process of connecting different branches.
One of the best practices for managing branches and resolving

Q.No.

conflicts is through implementing IaC (Infrastructure as Code) to maintain an error free and consistant code.

Git is an application where coders/ developers share ideas on a problem statement.

One can find ideas on how to work on that problem statement and find an example code for that problem.

One can also learn how to solve problems which are challenging for them by watching how others solved that problem.

It is a community where every one can learn.

Github is used to connect and store Git files from where this all is possible.

Q.No.	
3b) Ans:	Dockers can be used to migrate data. Docker containers are used to migrate an existing monolithic application to microservices architecture. To convert this one need to follow certain steps: 1) One needs to build a docker where they can store all the data 2) Ensure all the monolith is broken down into smaller data files which are to stored in containers. 3) Try to independently deploy these containe We can use tools like CI and Continues deployment while deploying. By using these tools the human intervention is reduced Errors are reduced. The process is automa Data is consistant.

Q.No.

1b) Ans: To implement infrastructure as Code one needs to follow steps:

To maintain consistency in infrastructure configurations across multiple environments one needs to must an sure implement IaC (Infrastructure as Code). This manages the consistency across all multiple environments. It is quite a challenge to implement this without following certain steps.

1) One needs to create ~~dock~~ docker

2) Create containers to ~~store~~ store data.

3) These containers can manage data.

4) We can manage containers to maintain consistency.

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

ROUGH WORK

Content written here will not be considered for valuation