HOME SERVICES SOFTWARE

A Literature report submitted in partial fulfillment of the requirement for degree of

Bachelor of Technology

in

Computer Science Engineering

By

C.TEJADEEP(R161817)
J.SIREESHA(R161177)

Under the guidance of

Satyanandaram N

Asst.professor in Department of Computer Science Engineering

Rajiv Gandhi University of Knowledge Technologies
RGUKT
R.K.Valley

Kadapa(Dt), Andhra Pradesh(516329), India

Rajiv Gandhi University of Knowledge Technologies

RGUKT
R.K.Valley

<u>CERTIFICATE</u>

This is to certify that the report entitled “**Home Service Software**” submitted by CHAKALI TEJADEEP, bearing ID.NO. R161817 and JALIPATI SIREESHA, bearing ID.NO R161177 in partial fulfillment of the requirements for the award of Bachelor of Technology in Computer Science is a bonafied work carried out by him under my supervision and guidance.

The report has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

C.RathnaKumari
Head of the Department,
Computer Science Department,
RGUKT, R.K Valley.

N.Sathyanandaram,
Project internal guide,
Computer Science Department,
RGUKT, R.K Valley.

DECLARATION

We CHAKALI TEJADEEP,JALIPATI SIREESHA hereby declare that this report entitled “Home service Software” submitted by us under the guidence and supervision of N.Sathyanandarama is a bonafide work. We also declare that it has not been submitted previously in part or in full to this University or Institution for the award of any degree or diploma.

Place:	(chakali Tejadeep)
Date:	R161817
	(Jalipati Sireesha)
	R161177

Acknowledgments

I would like to express my sincere gratitude to ***N.Sathyanandaram garu***, my project Guide, for valuable suggestions and keen interest through out the progress of my course of research.

I am greateful to T.Sandeep kumar, HOD of CSE, for providing excellent computing facilities and a congenial atmosphere for progressing with my project.

At the outsechnologies,R.K Valley for providing all the necessary resources for the successful completion of my course work. At last, but not the least I thank my classmates and other students for their physical and moral support.

With Sincere Regards,

Chakali Tejadeep.

Jalipati Sireesha.

Home Services Software

Under Graduation Level:-

Project Title: Home Services Software

Project Guide: Satyanandaram N

Project Focus:Provide details of the category of workers availability near by our

sourroundings

University:RGUKT,RK.VALLEY

Project Type: Website

Abstract:

The Home Services Software shows category of workers availability in online.The user can directly get option to select and send request to workers based on their needs.This is a Website that allows a user to register his details through register option.After registering the user will get login.User can seclect the type of worker or category of worker to their needs.After selcting the worker category the system allows to enter the user details and address.Then go back to the previous page which shows types of workers.And this app allows to select the worker based on pincode of the user area.This provides workers details near by our sourroundings based on pincode.User get opportunity to send request through mail/msg/call.This Home Services Software provides language switching option for uneducated people.And it provides online&offline payment method.

Contents

Acknowledgments

Abstract

Amazon Web Services

- 1.Introduction
- 2.History of Cloud
- 3.Architecture of Cloud Computing
- 4.Cloud Deployment Models
- 5.Cloud Service Models
- 5.Advantages of Cloud Computing
- 6.AWS Storage Services
- 7.Databases
- 8.Security And Identity Services
- 9.Amazon EC2

GitHub

- 1.Introduction
    - -What is GitHub
    - -Advantages of GitHub
    - -Features of GitHub
    - -Why GitHub
    - -What is Git
    - -Benfits of Git
    - -Why Git

-Git vs GitHub

-Essential components of GitHub

Devops

-Why Devops

-Waterfall Model & Challenges

-Devops History

-Devops Architecture Features

-Devops Components

-Devops WorkFlow

-Devops Principles & Practices

-Advantages

-DevOps Lifecycle

-DevOps Pipeline

-DevOps Methodology

-Azure DevOPs

-AWS DevOps

-Agile

-DevOps vs Agile

Deploying Map into project

Article

Project

Introduction

The Home Services Software shows category of workers availability in online.The user can directly get option to select and send request to workers based on their needs.This is a Website that allows a user to register his details through register option.After registering the user will get login.User can seclect the type of worker or category of worker to their needs.After selcting the worker category the system allows to enter the user details and address.Then go back to the previous page which shows types of workers.And this app allows to select the worker based on pincode of the user area.This provides workers details near by our sourroundings based on pincode.User get opportunity to send request through mail/msg/call.This Home Services Software provides language switching option for uneducated people.And it provides online&offline payment method.

LITERATURE SURVEY

AMAZON WEB SERVICES:

1.INTRODUCTION

AWS stands for Amazon Web Services which uses distributed IT infrastructure to provide different IT resources on demand.The AWS service is provided by the Amazon that uses distributed IT infrastructure to provide different IT resources available on demand. It provides different services such as infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS). Amazon launched AWS, a cloud computing platform to allow the different organizations to take advantage of reliable IT infrastructure.

Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud platform, offering over 200 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies—are using AWS to lower costs, become more agile, and innovate faster.

Uses of AWS

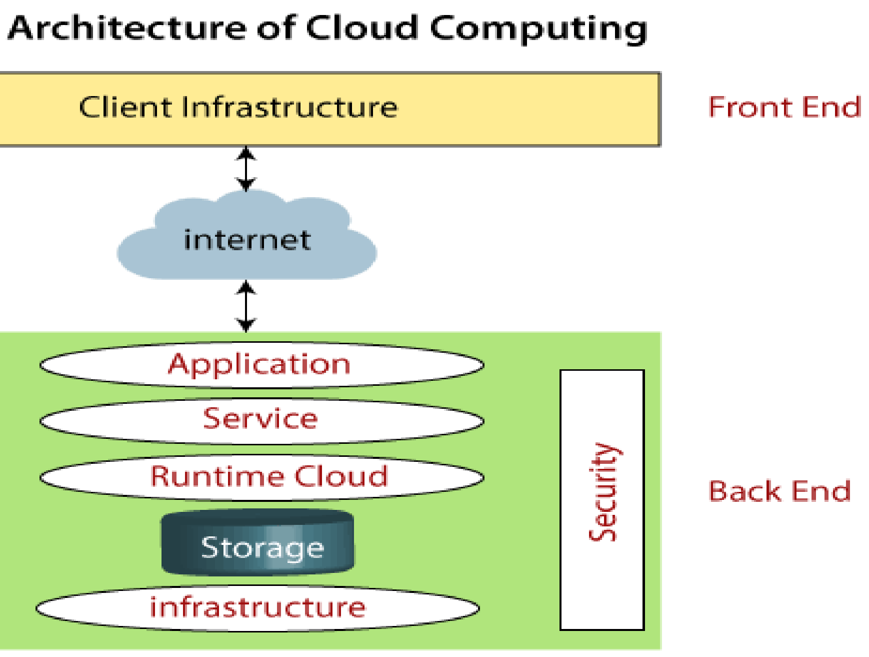
- A small manufacturing organization uses their expertise to expand their business by leaving their IT management to the AWS.
- A large enterprise spread across the globe can utilize the AWS to deliver the training to the distributed workforce.
- An architecture consulting company can use AWS to get the high-compute rendering of construction prototype.
- A media company can use the AWS to provide different types of content such as ebox or audio files to the worldwide files.

2.HISTORY OF CLOUD

Cloud Computing:
Cloud computing is the on-demand delivery of IT resources over the Internet with pay-as-you-go pricing. Instead of buying, owning, and maintaining physical data centers and servers, you can access technology services, such as computing power, storage, and databases, on an as-needed basis from a cloud provider like Amazon Web Services (AWS).

1. *In 1999, Salesforce.com started delivering of applications to users using a simple website. The applications were delivered to enterprises over the Internet, and this way the dream of computing sold as utility were true.*
2. *In 2002,* **Amazon** *started Amazon Web Services*, providing services like storage, computation and even human intelligence. However, only starting with the launch of the Elastic Compute Cloud in 2006 a truly commercial service open to everybody existed.
3. *In 2009,* **Google Apps** *also started to provide cloud computing enterprise applications.*
4. Of course, all the big players are present in the cloud computing evolution, some were earlier, some were later. *In 2009,* **Microsoft** *launched Windows Azure*, and companies like Oracle and HP have all joined the game. This proves that today, cloud computing has become mainstream.

3.ARCHITECTURE OF CLOUD COMPUTING



Front End

The front end is used by the client. It contains client-side interfaces and applications that are required to access the cloud computing platforms. The front end includes web servers (including Chrome, Firefox, internet explorer, etc.), thin & fat clients, tablets, and mobile devices.

Back End

The back end is used by the service provider. It manages all the resources that are required to provide cloud computing services. It includes a huge amount of data storage, security mechanism, virtual machines, deploying models, servers, traffic control mechanisms, etc.

1. Client Infrastructure

Client Infrastructure is a Front end component. It provides GUI (Graphical User Interface) to interact with the cloud.

2. Application

The application may be any software or platform that a client wants to access.

3. Service

A Cloud Services manages that which type of service you access according to the client's requirement.

Cloud computing offers the following three type of services:

i. Software as a Service (SaaS) – It is also known as **cloud application services**. Mostly, SaaS applications run directly through the web browser means we do not require to download and install these applications. Some important example of SaaS is given below –

Example: Google Apps, Salesforce Dropbox, Slack, Hubspot, Cisco WebEx.

ii. Platform as a Service (PaaS) – It is also known as **cloud platform services**. It is quite similar to SaaS, but the difference is that PaaS provides a platform for software creation, but using SaaS, we can access software over the internet without the need of any platform.

Example: Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.

iii. Infrastructure as a Service (IaaS) – It is also known as **cloud infrastructure services**. It is responsible for managing applications data, middleware, and runtime environments.

Example: Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

4. Runtime Cloud

Runtime Cloud provides the **execution and runtime environment** to the virtual machines.

5. Storage

Storage is one of the most important components of cloud computing. It provides a huge amount of storage capacity in the cloud to store and manage data.

6. Infrastructure

It provides services on the **host level**, **application level**, and **network level**. Cloud infrastructure includes hardware and software components such as servers, storage, network devices, virtualization software, and other storage resources that are needed to support the cloud computing model.

7. Management

Management is used to manage components such as application, service, runtime cloud, storage, infrastructure, and other security issues in the backend and establish coordination between them.

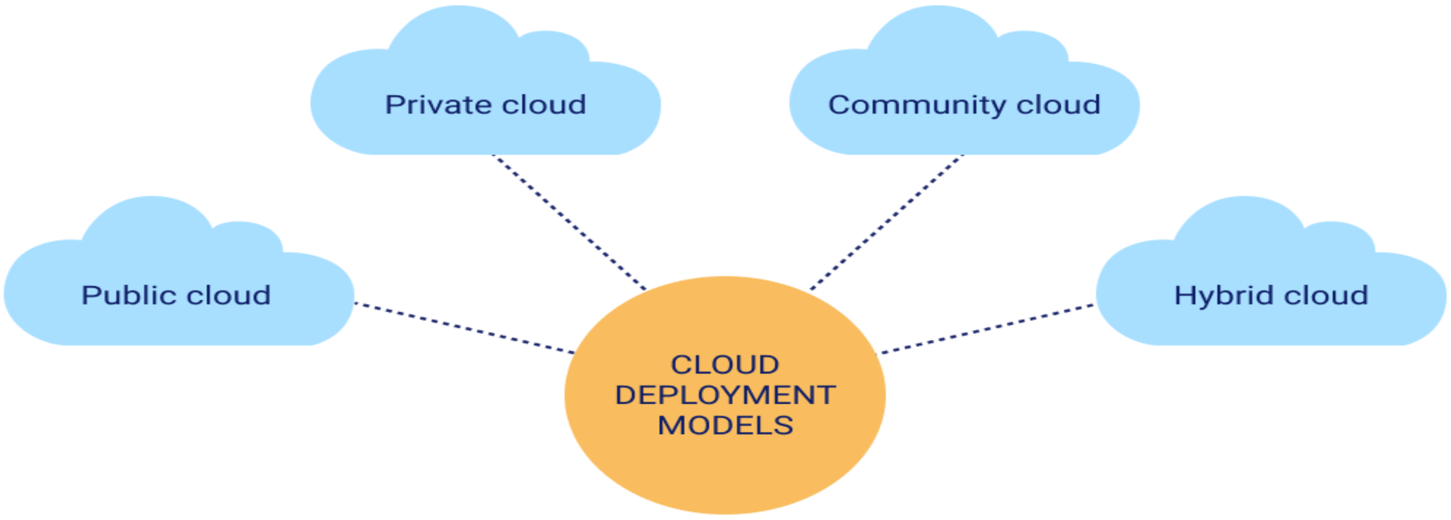
8. Security

Security is an in-built back end component of cloud computing. It implements a security mechanism in the back end.

9. Internet

The Internet is medium through which front end and back end can interact and communicate with each other.

4.CLOUD DEPLOYMENT MODELS

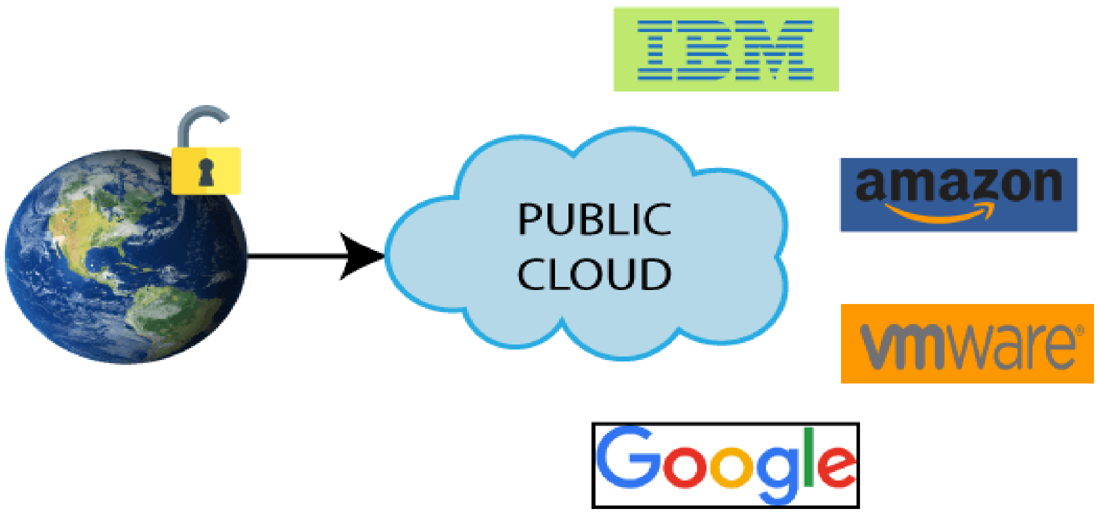


Public Cloud

Public cloud is **open to all** to store and access information via the Internet using the pay-per-usage method.

In public cloud, computing resources are managed and operated by the Cloud Service Provider (CSP).

Example: Amazon elastic compute cloud (EC2), IBM SmartCloud Enterprise, Microsoft, Google App Engine, Windows Azure Services Platform.

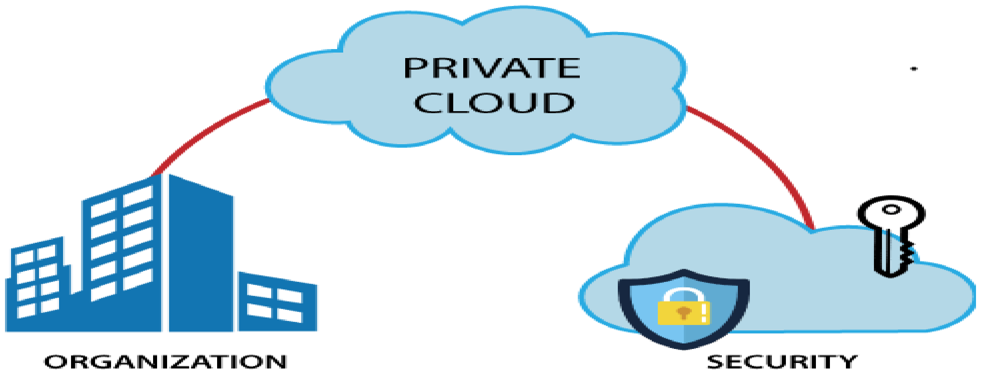


Private Cloud

Private cloud is also known as an **internal cloud** or **corporate cloud**. It is used by organizations to build and manage their own data centers internally or by the third party. It can be deployed using Opensource tools such as Openstack and Eucalyptus.

Based on the location and management, National Institute of Standards and Technology (NIST) divide private cloud into the following two parts-

- On-premise private cloud
- Outsourced private cloud



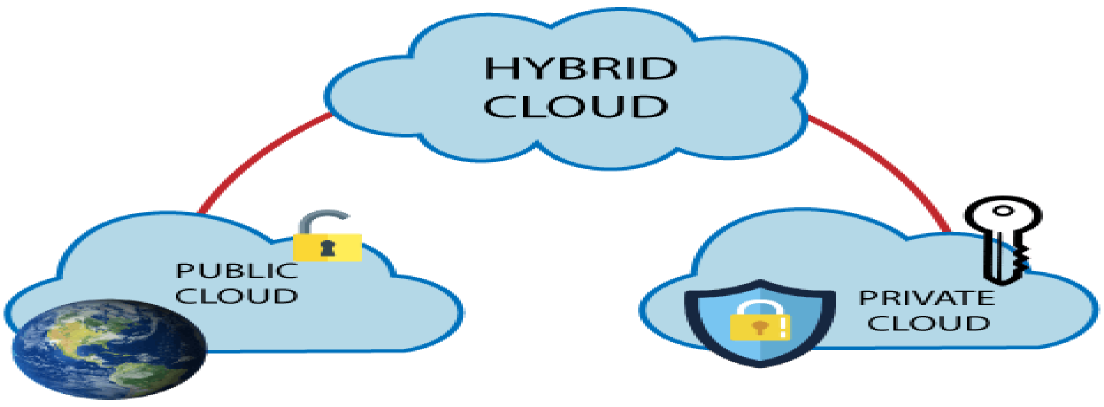
Hybrid Cloud

Hybrid Cloud is a combination of the public cloud and the private cloud. we can say:

Hybrid Cloud = Public Cloud + Private Cloud

Hybrid cloud is partially secure because the services which are running on the public cloud can be accessed by anyone, while the services which are running on a private cloud can be accessed only by the organization's users.

Example: Google Application Suite (Gmail, Google Apps, and Google Drive), Office 365 (MS Office on the Web and One Drive), Amazon Web Services.



Community Cloud

Community cloud allows systems and services to be accessible by a group of several organizations to share the information between the organization and a specific community. It is owned, managed, and operated by one or more organizations in the community, a third party, or a combination of them.

Example: Health Care community cloud

5.CLOUD SERVICE MODELS

Infrastructure as a Service (IaaS)

IaaS is also known as **Hardware as a Service (HaaS)**. It is a computing infrastructure managed over the internet. The main advantage of using IaaS is that it helps users to avoid the cost and complexity of purchasing and managing the physical servers.

Characteristics of IaaS

There are the following characteristics of IaaS -

- Resources are available as a service
- Services are highly scalable
- Dynamic and flexible
- GUI and API-based access
- Automated administrative tasks

Platform as a Service (PaaS)

PaaS cloud computing platform is created for the programmer to develop, test, run, and manage the applications.

Characteristics of PaaS

There are the following characteristics of PaaS -

- Accessible to various users via the same development application.
- Integrates with web services and databases.
- Builds on virtualization technology, so resources can easily be scaled up or down as per the organization's need.

- Support multiple languages and frameworks.
- Provides an ability to "**Auto-scale**".

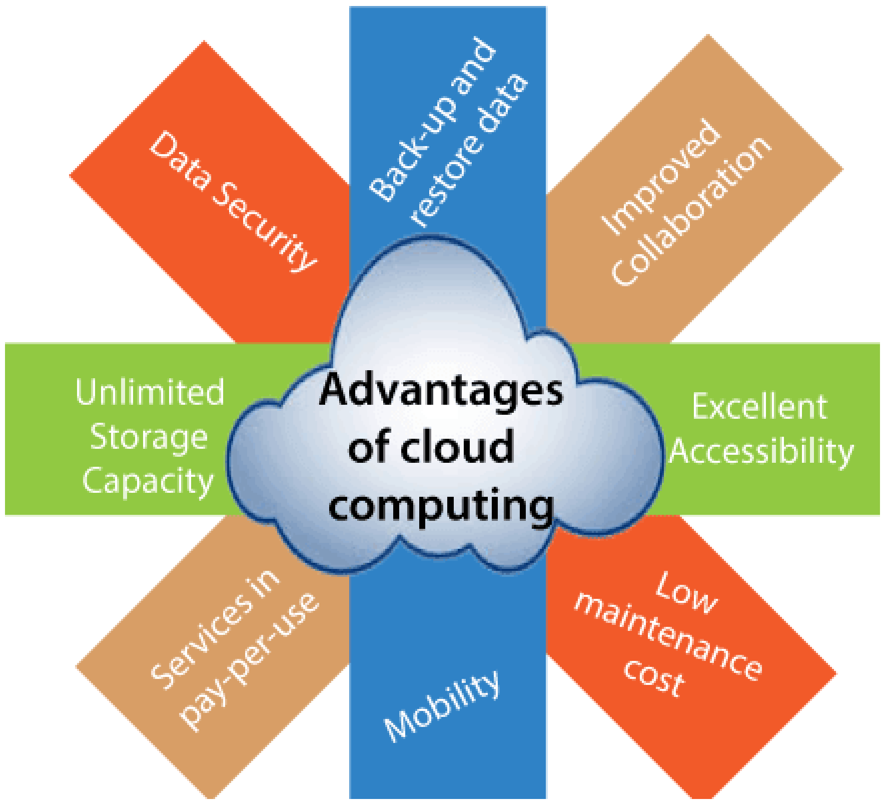
Software as a Service (SaaS)

SaaS is also known as "**on-demand software**". It is a software in which the applications are hosted by a cloud service provider. Users can access these applications with the help of internet connection and web browser.

Characteristics of SaaS

There are the following characteristics of SaaS -

- Managed from a central location
- Hosted on a remote server
- Accessible over the internet
- Users are not responsible for hardware and software updates. Updates are applied automatically.
- The services are purchased on the pay-as-per-use basis



5.ADVANTAGES OF CLOUDCOPUTING

1) Back-up and restore data

Once the data is stored in the cloud, it is easier to get back-up and restore that data using the cloud.

2) Improved collaboration

Cloud applications improve collaboration by allowing groups of people to quickly and easily share information in the cloud via shared storage.

3) Excellent accessibility

Cloud allows us to quickly and easily access store information anywhere, anytime in the whole world, using an internet connection. An internet cloud infrastructure increases organization productivity and efficiency by ensuring that our data is always accessible.

4) Low maintenance cost

Cloud computing reduces both hardware and software maintenance costs for organizations.

5) Mobility

Cloud computing allows us to easily access all cloud data via mobile.

6) IServices in the pay-per-use model

Cloud computing offers Application Programming Interfaces (APIs) to the users for access services on the cloud and pays the charges as per the usage of service.

7) Unlimited storage capacity

Cloud offers us a huge amount of storing capacity for storing our important data such as documents, images, audio, video, etc. in one place.

8) Data security

Data security is one of the biggest advantages of cloud computing. Cloud offers many advanced features related to security and ensures that data is securely stored and handled.

6.AWS STORAGE SERVICES

S3 (Simple Storage Service) — *Storage service of AWS inwhich we can store objects like files, folders, images,documents, songs, etc. It cannot be used to installsoftware, games or Operating System.*

• ***EFS (Elastic File System) —*** *Provides file storage for usewith your EC2 instances. It uses NFSv4 protocol and can beused concurrently by thousands of instances.*

• ***Glacier —*** *It is an extremely low-cost archival service tostore files for a long time like a few years or even decades.*

• ***Storage Gateway —*** *It is a virtual machine that you installon your on-premise servers. Your on-premise data can bebacked up to AWS providing more durability.*

7.DATABSES

RDS (Relational Database Service) — *Allows you torun relational databases like MySQL, MariaDB, PostgreSQL,Oracle or SQL Server. These databases are fully managed byAWS like installing antivirus and patches.*

• ***DynamoDB —*** *It is a highly scalable, high-performance NoSQL database. It provides single-digit millisecond latency at any scale.*

• ***Elasticache —*** *It is a way of caching data inside the cloud. It can be used to take load off of your database by cachingmost frequent queries.*

• ***Neptune —*** *It has been launched recently. It is a fast,reliable and scalable graph database service.*

• ***RedShift —*** *It is AWS's data warehousing solution that can be used to run complex OLAP queries.*

8.SECURITY AND IDENTITY SERVICES

IAM (Identity and Access Management) *— Allows youto manage users, assign policies, create groups to manage multiple users.*

• ***Inspector*** *— It is an agent that you install on our virtual machines, which then reports any security vulnerabilities.*

• ***Certificate Manager*** *— It gives free SSL certificates for your domains that are managed by Route53.*

• ***Directory Service*** *— A way of using your company's account to log in to AWS.*

9.AMAZON EC2

Introduction to AWS EC2

Amazon Elastic Compute Cloud, EC2 is a web service from Amazon that provides re-sizable compute services in the cloud. They are re-sizable because you can quickly scale up or scale down the number of server instances you are using if your computing requirements change.

WHAT IS INSTANCE

An instance is a virtual server for running applications on Amazon's EC2. It can also be understood like a tiny part of a larger computer, a tiny part which has its own Hard drive, network connection, OS etc. But it is actually all virtual.You can have multiple "tiny" computers on a single physical machine, and all these tiny machines are called Instances.

Why AWS EC2 ?

Suppose you are a developer, and since you want to work independently you buy some servers, you estimated the correct capacity, and the computing power is enough. Now, you have to look after the updating of security patches every day, you have to troubleshoot any problem which might occur at a back end level in the servers and so on.But if you buy an EC2 instance, you don't have to worry about any of these things as it will all be managed by Amazon; you just have to focus on your application.

Types of EC2 Computing Instances

Computing is a very broad term, the nature of your task decides what kind of computing you need. Therefore,AWS EC2 offers 5 types of instances which are as follows:

1. General Instances

For applications that require a balance of performance and cost.E.g email responding systems, where you need a prompt response as well as the it should be cost effective, since it doesn't require much processing.

2. Compute Instances

For applications that require a lot of processing from the CPU.

E.g analysis of data from a stream of data, like Twitter stream

3. Memory Instances

For applications that are heavy in nature, therefore, require a lot of RAM.

E.g when your system needs a lot of applications running in the background

i.e multitasking.

4. Storage Instances

For applications that are huge in size or have a data set that occupies a lot of

space.

E.g When your application is of huge size.

5. GPU Instances

For applications that require some heavy graphics rendering.

E.g 3D modeling etc.

GitHub

INTRODUCTION

What is GitHub?

GitHub is an immense platform for code hosting. It supports version controlling and collaboration and allows developers to work together on projects. It offers both distributed version control and source code management (SCM) functionality of Git. It also facilitates collaboration features such as bug tracking, feature requests, task management for every project.

Advantages of GitHub

GitHub can be separated as the Git and the Hub. GitHub service includes access controls as well as collaboration features like task management, repository hosting, and team management.

- The key benefits of GitHub are as follows.
- It is easy to contribute to open source projects via GitHub.
- It helps to create an excellent document.
- You can attract the recruiter by showing off your work. If you have a profile on GitHub, you will have a higher chance of being recruited.
- It allows your work to get out there in front of the public.
- You can track changes in your code across versions.

Features of GitHub

GitHub is a place where programmers and designers work together. They collaborate, contribute, and fix bugs together. It hosts plenty of open source projects and codes of various programming languages.

GitHub
Collaboration
Bug Tracking
Branches
Git Repositories
Project Management
Team Management
Code Hosting
Track and assign tasks

Some of its significant features are as follows.

- Collaboration
- Integrated issue and bug tracking
- Graphical representation of branches
- Git repositories hosting
- Project management
- Team management
- Code hosting
- Track and assign tasks
- Conversations

Why Git Hub

Github is like social media for programmers. They can upload there code and showoff there skill. They can have back up of all the files here. Git provides them facility to undo their work infinite times so this is one of the most important reason why we use git. Like social media programmer can create group here and do projects it also gives chat option. So all who working in project can communicate there itself. Like admin in what's app here admin have all powers to accepts new members and changes or modifications done by others.

What is Git?

Git is an **open-source distributed version control system**. It is designed to handle minor to major projects with high speed and efficiency. It is developed to co-ordinate the work among the developers. The version control allows us to track and work together with our team members at the same workspace.

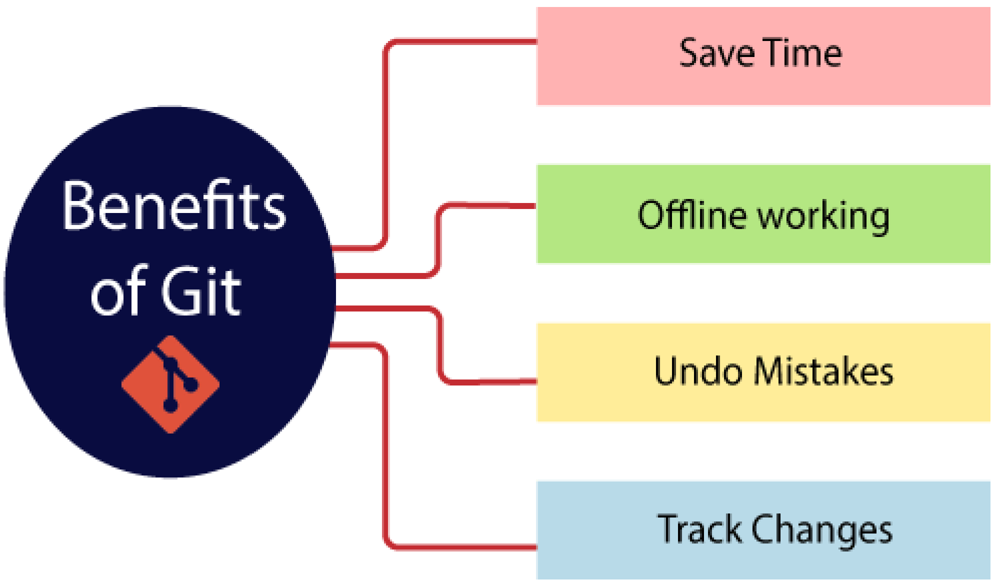
Git is foundation of many services like **GitHub** and **GitLab**, but we can use Git without using any other Git services. Git can be used **privately** and **publicly**.

Git was created by **Linus Torvalds** in **2005** to develop Linux Kernel. It is also used as an important distributed version-control tool for **the DevOps**

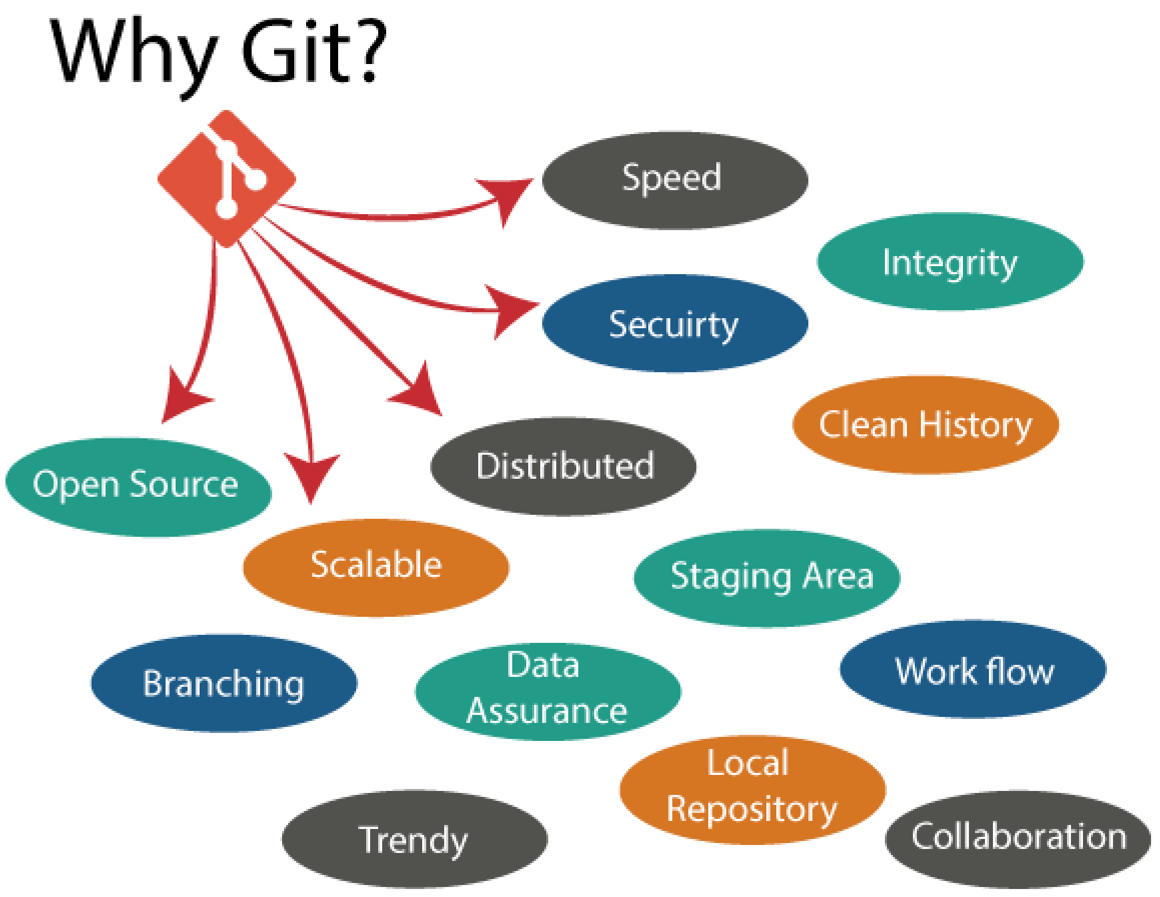
Benefits of Git

A version control application allows us to **keep track** of all the changes that we make in the files of our project. Every time we make changes in files of an existing project, we can push those changes to a repository. Other developers are allowed to pull your changes from the repository and continue to work with the updates that you added to the project files.

Some **significant benefits** of using Git are as follows:



- **Saves Time**
  Git is lightning fast technology. Each command takes only a few seconds to execute so we can save a lot of time as compared to login to a GitHub account and find out its features.
- **Offline Working**
  One of the most important benefits of Git is that it supports **offline working**. If we are facing internet connectivity issues, it will not affect our work. In Git, we can do almost everything locally. Comparatively, other CVS like SVN is limited and prefer the connection with the central repository.
- **Undo Mistakes**
  One additional benefit of Git is we can **Undo** mistakes. Sometimes the undo can be a savior option for us. Git provides the undo option for almost everything.
- **Track the Changes**
  Git facilitates with some exciting features such as **Diff, Log,** and **Status**, which allows us to track changes so we can **check the status, compare** our files or branches.



- **Git Integrity**
  Git is **developed to ensure** the **security** and **integrity** of content being version controlled. It uses checksum during transit or tampering with the file system to confirm that information is not lost. Internally it creates a checksum value from the contents of the file and then verifies it when transmitting or storing data.
- **Trendy Version Control System**
  Git is the **most widely used version control system**. It has **maximum projects** among all the version control systems. Due to its **amazing workflow** and features, it is a preferred choice of developers.
- **Everything is Local**
  Almost All operations of Git can be performed locally; this is a significant reason for the use of Git. We will not have to ensure internet connectivity.
- **Collaborate to Public Projects**
  There are many public projects available on the GitHub. We can collaborate on those projects and show our creativity to the world. Many developers are collaborating on public projects. The collaboration allows us to stand with experienced developers and learn a lot from them; thus, it takes our programming skills to the next level.
- **Impress Recruiters**
  We can impress recruiters by mentioning the Git and GitHub on our resume. Send your GitHub profile link to the HR of the organization you want to join. Show your skills and influence them through your work. It increases the chances of getting hired.

Git vs Git Hub

Git	GitHub
Git is a distributed version control tool that can manage a programmer's source code history.	GitHub is a cloud-based tool developed around the Git tool.
A developer installs Git tool locally.	GitHub is an online service to store code and push from the computer running the Git tool.
Git focused on version control and code sharing.	GitHub focused on centralized source code hosting.
It is a command-line tool.	It is administered through the web.
It facilitates with a desktop interface called Git Gui.	It also facilitates with a desktop interface called GitHub Gui.
Git does not provide any user management feature.	GitHub has a built-in user management feature.
It has minimal tool configuration feature.	It has a market place for tool configuration.

Essential components of the GitHub are:

- Repositories
- Branches
- Commits
- Pull Requests
- Git (the version control tool GitHub is built on)

GitHub Repository

A GitHub repository can be used to store a development project.
It can contain **folders** and any type of **files** (HTML, CSS, JavaScript, Documents, Data, Images).
A GitHub repository should also include a **licence** file and a **README** file about the project.
A GitHub repository can also be used to store ideas, or any resources that you want to share.

Branch

A GitHub branch is used to work with different **versions** of a repository at the same time.By default a repository has a **master** branch (a production branch).Any other branch is a **copy** of the master branch (as it was at a point in time). New Branches are for bug fixes and feature work separate from the master branch. When changes are ready, they can be merged into the master branch. If you make changes to the master branch while working on a new branch, these updates can be pulled in.

Commits

The git commit command will save all staged changes, along with a brief description from the user, in a "commit" to the local repository. Commits are at the heart of Git usage. You can think of a commit as a snapshot of your project, where a new version of that project is created in the current repository

Full Request

You make local code changes and then submit those changes to a remote project maintainer for review before those changes are implemented, or merged. This is called a pull request; you are requesting that someone reviews and approves your changes before they become final.

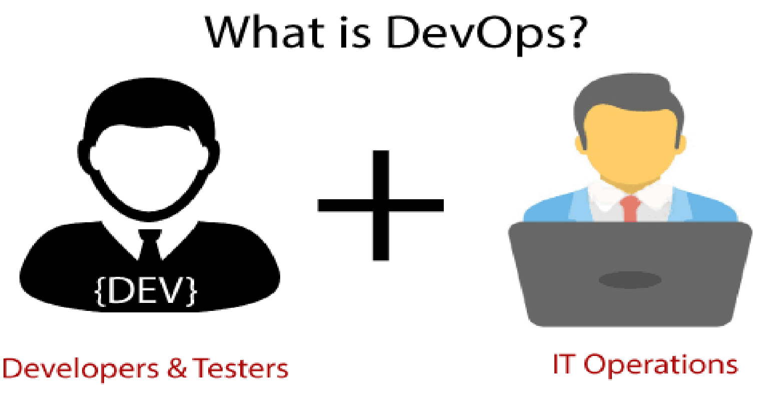
A "**pull request**" is you **requesting** the target repository to please grab your changes.

DevOps

INTRODUCTION

What is DevOps?

The DevOps is a combination of two words, one is software Development, and second is Operations. This allows a single team to handle the entire application lifecycle, from development to **testing**, **deployment**, and **operations**. DevOps helps you to reduce the disconnection between software developers, quality assurance (QA) engineers, and system administrators.

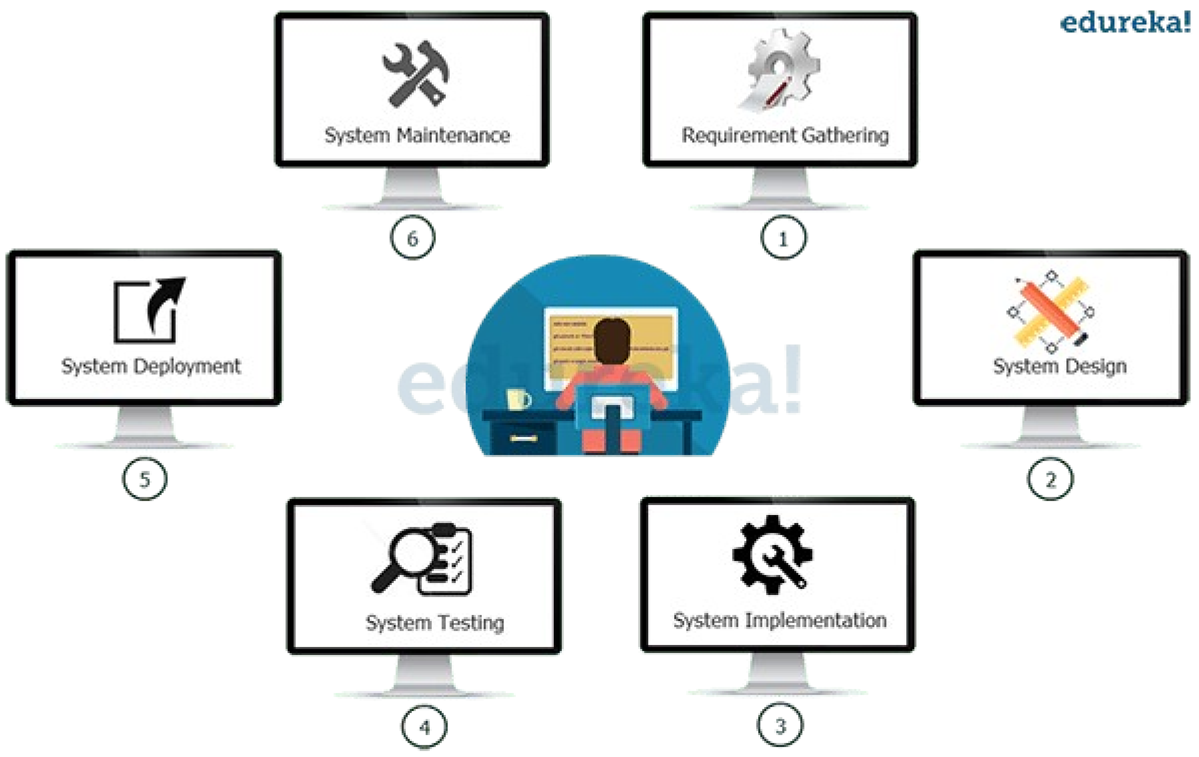


Why DevOps?

1. The operation and development team worked in complete isolation.
2. After the design-build, the testing and deployment are performed respectively. That's why they consumed more time than actual build cycles.
3. Without the use of DevOps, the team members are spending a large amount of time on designing, testing, and deploying instead of building the project.
4. Manual code deployment leads to human errors in production.
5. Coding and operation teams have their separate timelines and are not in synch, causing further delays.

Waterfall Model

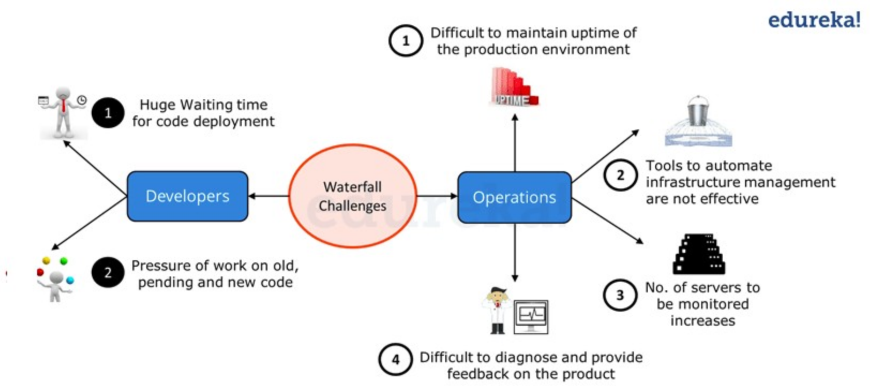
Let’s consider developing software in a traditional way using a Waterfall Model.



In the above diagram you will see the phases it will involve:

- In phase 1 – Complete Requirement is gathered and SRS is developed
- In phase 2 – This System is Planned and Designed using the SRS
- In phase 3 – Implementation of the System takes place
- In phase 4 – System is tested and its quality is assured
- In phase 5 – System is deployed to the end users
- In phase 6 – Regular Maintenance of the system is done

Waterfall Model Challenges


edureka!
1 Difficult to maintain uptime of the production environment
UPTIME
1 Huge Waiting time for code deployment
Developers
Waterfall Challenges
Operations
2 Tools to automate infrastructure management are not effective
2 Pressure of work on old, pending and new code
3 No. of servers to be monitored increases
4 Difficult to diagnose and provide feedback on the product

The Water-fall model worked fine and served well for many years however it had some challenges. In the following diagram the challenges of Waterfall Model are highlighted.

In the above diagram you can see that both Development and Operations had challenges in the Waterfall Model. From Developers point of view there were majorly two challenges:

1 After Development, the code deployment time was huge.

2 Pressure of work on old, pending and new code was high because development and deployment time was high.

On the other hand, Operations was also not completely satisfied. There were four major challenges they faced as per the above diagram:

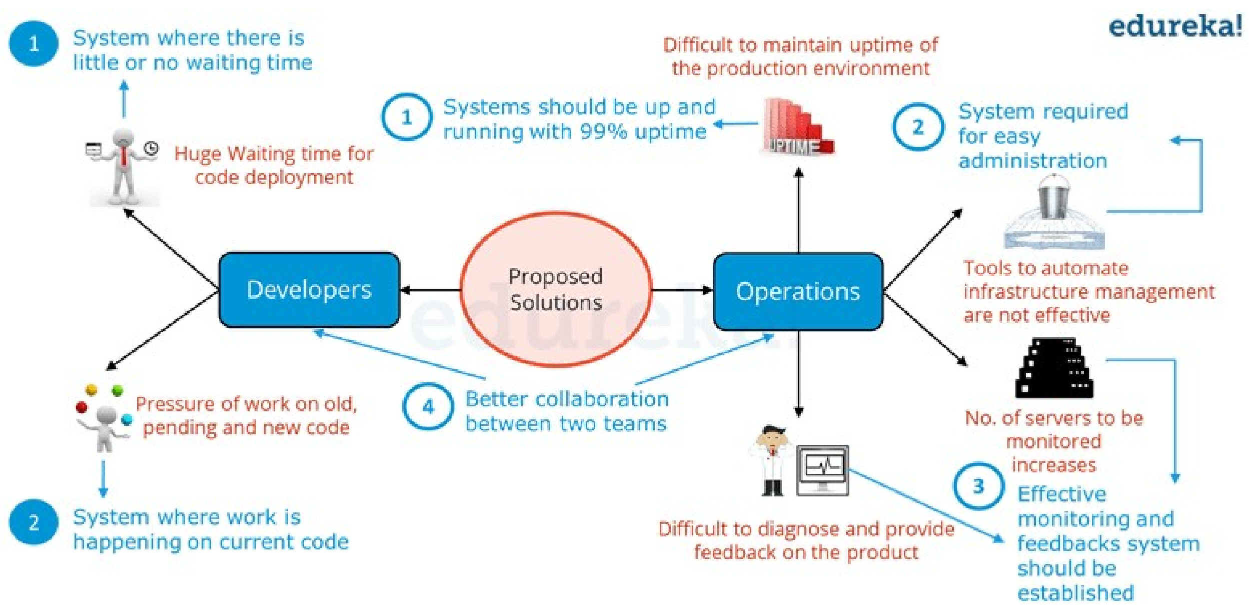
1 It was difficult to maintain ~100% uptime of the production environment.

2 Infrastructure Automation tools were not very affective.

3 Number of severs to be monitored keeps on increasing with time and hence the complexity

4 It was very difficult to provide feedback and diagnose issue in the product.

In the following diagram proposed solution to the challenges of Waterfall Model are highlighted.


edureka!
1 System where there is little or no waiting time
Huge Waiting time for code deployment
Difficult to maintain uptime of the production environment
1 Systems should be up and running with 99% uptime
UPTIME
2 System required for easy administration
Tools to automate infrastructure management are not effective
Developers
Proposed Solutions
Operations
Pressure of work on old, pending and new code
4 Better collaboration between two teams
No. of servers to be monitored increases
2 System where work is happening on current code
3 Effective monitoring and feedbacks system should be established
Difficult to diagnose and provide feedback on the product

In the above diagram, Probable Solutions for the issues faced by Developers and Operations are highlighted in blue. This sets the guidelines for an Ideal Software Development strategy.

From Developers point of view:

1 A system which enables code deployment without any delay or wait time.

2 A system where work happens on the current code itself i.e. development sprints are short and well planned.

From Operations point of view:

1 System should have at-least 99% uptime.

2 Tools & systems are there in place for easy administration.

3 Effective monitoring and feedbacks system should be there.

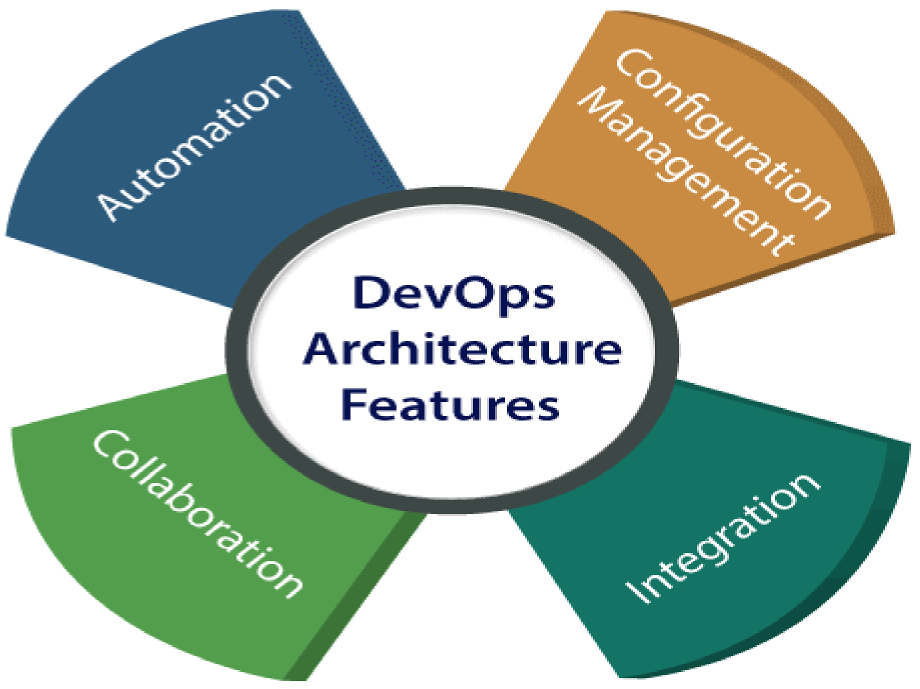
4 Better Collaboration between Development & Operations and is common requirement for Developers and Operations team.

DevOps integrates developers and operations team to improve collaboration and productivity.

DevOps History

- In 2009, the first conference named **DevOpsdays** was held in Ghent Belgium. Belgian consultant and Patrick Debois founded the conference.
- In 2012, the state of DevOps report was launched and conceived by Alanna Brown at Puppet.
- In 2014, the annual State of DevOps report was published by Nicole Forsgren, Jez Humble, Gene Kim, and others. They found DevOps adoption was accelerating in 2014 also.
- In 2015, Nicole Forsgren, Gene Kim, and Jez Humble founded DORA (DevOps Research and Assignment).
- In 2017, Nicole Forsgren, Gene Kim, and Jez Humble published "Accelerate: Building and Scaling High Performing Technology Organizations".

DevOps Architecture Features



1) Automation

Automation can reduce time consumption, especially during the testing and deployment phase. The productivity increases, and releases are made quicker by automation. This will lead in catching bugs quickly so that it can be fixed easily. For contiguous delivery, each code is defined through automated tests, cloud-based services, and builds. This promotes production using automated deploys.

2) Collaboration

The Development and Operations team collaborates as a DevOps team, which improves the cultural model as the teams become more productive with their productivity, which strengthens accountability and ownership. The teams share their responsibilities and work closely in sync, which in turn makes the deployment to production faster.

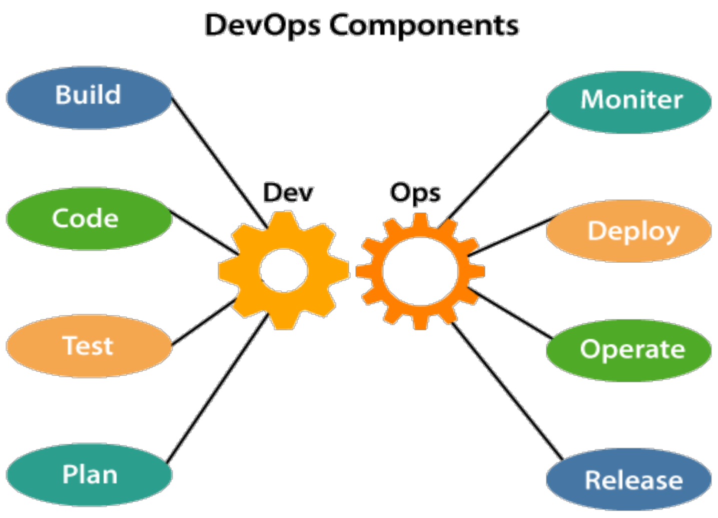
3) Integration

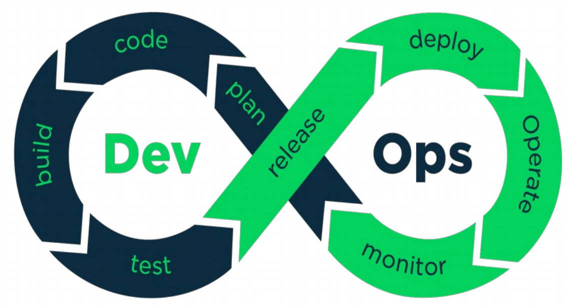
Applications need to be integrated with other components in the environment. The integration phase is whore the existing code is combined with new functionality and then tested. Continuous integration and testing enable continuous development. The frequency in the releases and micro-services leads to significant operational challenges. To overcome such problems, continuous integration and delivery are implemented to deliver in a **quicker, safer**, and **reliable manner**.

4) Configuration management

It ensures the application to interact with only those resources that are concerned with the environment in which it runs. The configuration files are not created where the external configuration to the application is separated from the source code. The configuration file can be written during deployment, or they can be loaded at the run time, depending on the environment in which it is running.

DevOps Components





1) Build

Without DevOps, the cost of the consumption of the resources was evaluated based on the pre-defined individual usage with fixed hardware allocation. And with DevOps, the usage of cloud, sharing of resources comes into the picture, and the build is dependent upon the user's need, which is a mechanism to control the usage of resources or capacity.

2) Code

Many good practices such as Git enables the code to be used, which ensures writing the code for business, helps to track changes, getting notified about the reason behind the difference in the actual and the expected output, and if necessary reverting to the original code developed. The code can be appropriately arranged in **files, folders**, etc. And they can be reused.

3) Test

The application will be ready for production after testing. In the case of manual testing, it consumes more time in testing and moving the code to the output. The testing can be automated, which decreases the time for testing so that the time to deploy the code to production can be reduced as automating the running of the scripts will remove many manual steps.

4) Plan

DevOps use Agile methodology to plan the development. With the operations and development team in sync, it helps in organizing the work to plan accordingly to increase productivity.

5) Monitor

Continuous monitoring is used to identify any risk of failure. Also, it helps in tracking the system accurately so that the health of the application can be checked. The monitoring becomes more comfortable with services where the log data may get monitored through many third-party tools such as **Splunk**.

6) Deploy

Many systems can support the scheduler for automated deployment. The cloud management platform enables users to capture accurate insights and view the optimization scenario, analytics on trends by the deployment of dashboards.

7) Operate

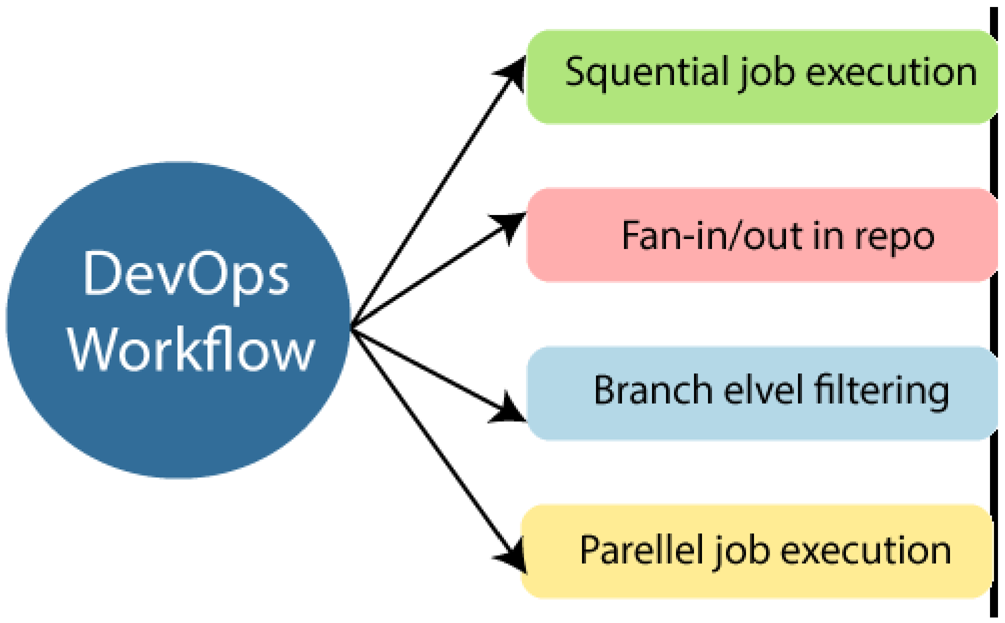
DevOps changes the way traditional approach of developing and testing separately. The teams operate in a collaborative way where both the teams actively participate throughout the service lifecycle. The operation team interacts with developers, and they come up with a monitoring plan which serves the IT and business requirements.

8) Release

Deployment to an environment can be done by automation. But when the deployment is made to the production environment, it is done by manual triggering. Many processes involved in release management commonly used to do the deployment in the production environment manually to lessen the impact on the customers.

DevOps Workflow

DevOps workflow provides a visual overview of the sequence in which input is provided. Also, it tells about which one action is performed, and output is generated for an operations process.



DevOps workflow allows the ability to separate and arrange the jobs which are top requested by the users. Also, it gives the ability to mirror their ideal process in the configuration jobs.

DevOps Principles

The main principles of DevOps are Continuous delivery, automation, and fast reaction to the feedback.

1. **End to End Responsibility:** DevOps team need to provide performance support until they become the end of life. It enhances the responsibility and the quality of the products engineered.
2. **Continuous Improvement:** DevOps culture focuses on continuous improvement to minimize waste. It continuously speeds up the growth of products or services offered.
3. **Automate Everything:** Automation is an essential principle of the DevOps process. This is for software development and also for the entire infrastructure landscape.
4. **Custom Centric Action:** DevOps team must take customer-centric for that they should continuously invest in products and services.
5. **Monitor and test everything:** The DevOps team needs to have robust monitoring and testing procedures.
6. **Work as one team:** In the DevOps culture role of the designers, developers, and testers are already defined. All they needed to do is work as one team with complete collaboration.

These principles are achieved through several DevOps practices, which include frequent deployments, QA automation, continuous delivery, validating ideas as early as possible, and in-team collaboration.

DevOps Practices

Some identified DevOps practices are:

- Self-service configuration
- Continuous build
- Continuous integration

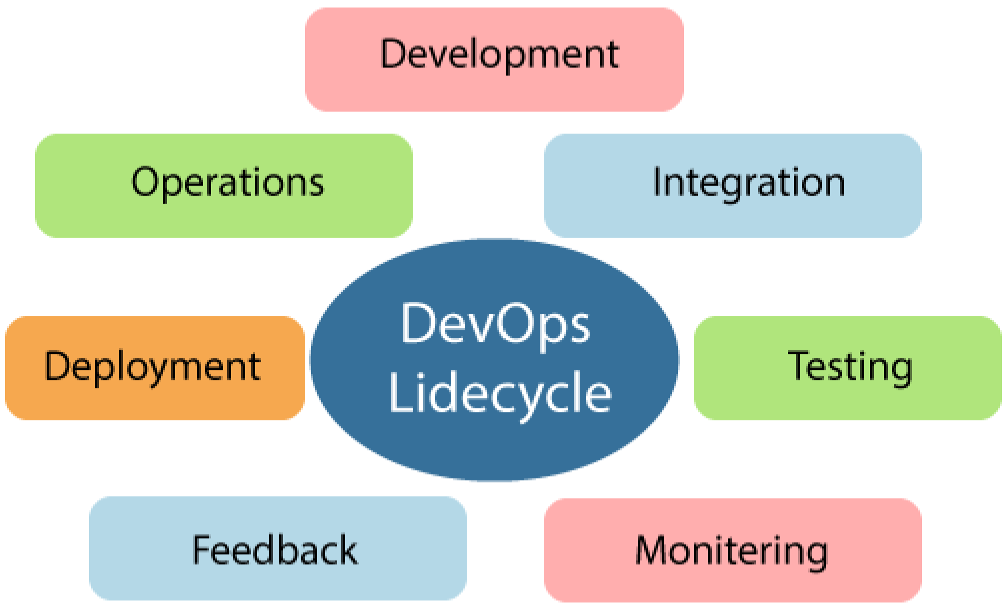
- Continuous delivery
- Incremental testing
- Automated provisioning
- Automated release management

DevOps Advantages

- DevOps is an excellent approach for quick development and deployment of applications.
- It responds faster to the market changes to improve business growth.
- DevOps escalate business profit by decreasing software delivery time and transportation costs.
- DevOps clears the descriptive process, which gives clarity on product development and delivery.
- It improves customer experience and satisfaction.
- DevOps simplifies collaboration and places all tools in the cloud for customers to access.
- DevOps means collective responsibility, which leads to better team engagement and productivity.

DevOps Lifecycle

DevOps defines an agile relationship between operations and Development. It is a process that is practiced by the development team and operational engineers together from beginning to the final stage of the product.



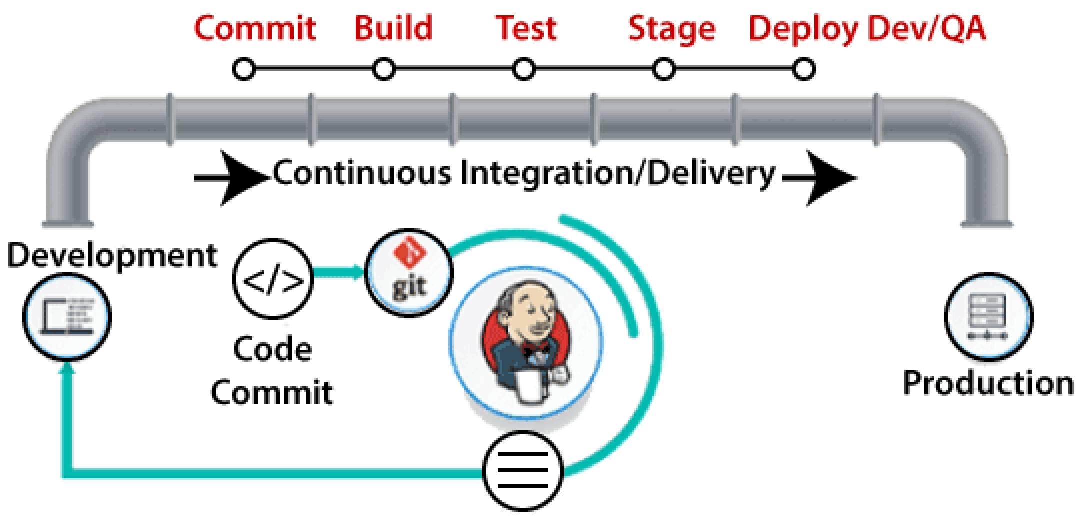
1) Continuous Development

This phase involves the planning and coding of the software. The vision of the project is decided during the planning phase. And the developers begin developing the code for the application. There are no DevOps tools that are required for planning, but there are several tools for maintaining the code.

2) Continuous Integration

This stage is the heart of the entire DevOps lifecycle. It is a software development practice in which the developers require to commit changes to the source code more frequently. This may be on a daily or weekly basis. Then every commit is built, and this allows early detection of problems if they are present. Building code is not only involved compilation, but it also includes **unit testing, integration testing, code review**, and **packaging**.

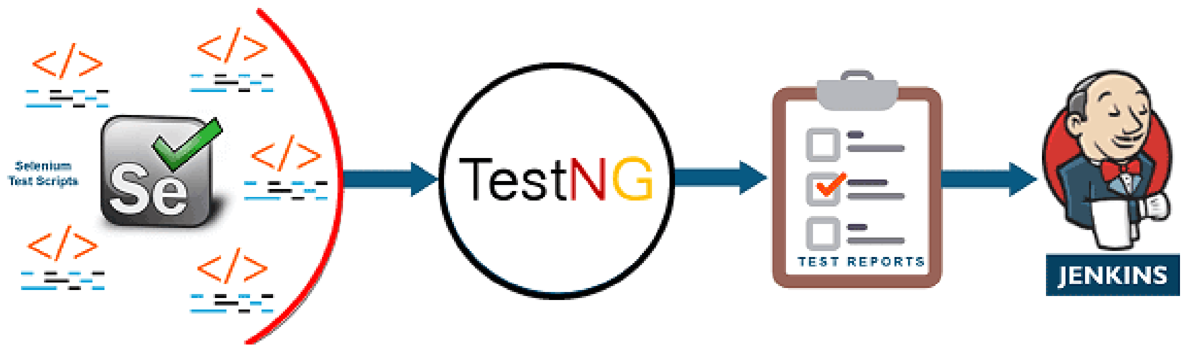
The code supporting new functionality is continuously integrated with the existing code. Therefore, there is continuous development of software. The updated code needs to be integrated continuously and smoothly with the systems to reflect changes to the end-users.



Jenkins is a popular tool used in this phase. Whenever there is a change in the Git repository, then Jenkins fetches the updated code and prepares a build of that code, which is an executable file in the form of war or jar. Then this build is forwarded to the test server or the production server.

3) Continuous Testing

This phase, where the developed software is continuously testing for bugs. For constant testing, automation testing tools such as **TestNG, JUnit, Selenium**, etc are used. These tools allow QAs to test multiple code-bases thoroughly in parallel to ensure that there is no flaw in the functionality. In this phase, **Docker** Containers can be used for simulating the test environment.



Selenium does the automation testing, and TestNG generates the reports. This entire testing phase can automate with the help of a Continuous Integration tool called Jenkins.

Automation testing saves a lot of time and effort for executing the tests instead of doing this manually. Apart from that, report generation is a big plus. The task of evaluating the test cases that failed in a test suite gets simpler. Also, we can schedule the execution of the test cases at predefined times. After testing, the code is continuously integrated with the existing code.

4) Continuous Monitoring

Monitoring is a phase that involves all the operational factors of the entire DevOps process, where important information about the use of the software is recorded and carefully processed to find out trends and identify problem areas. Usually, the monitoring is integrated within the operational capabilities of the software application.

It may occur in the form of documentation files or maybe produce large-scale data about the application parameters when it is in a continuous use position. The system errors such as server not reachable, low memory, etc are resolved in this phase. It maintains the security and availability of the service.

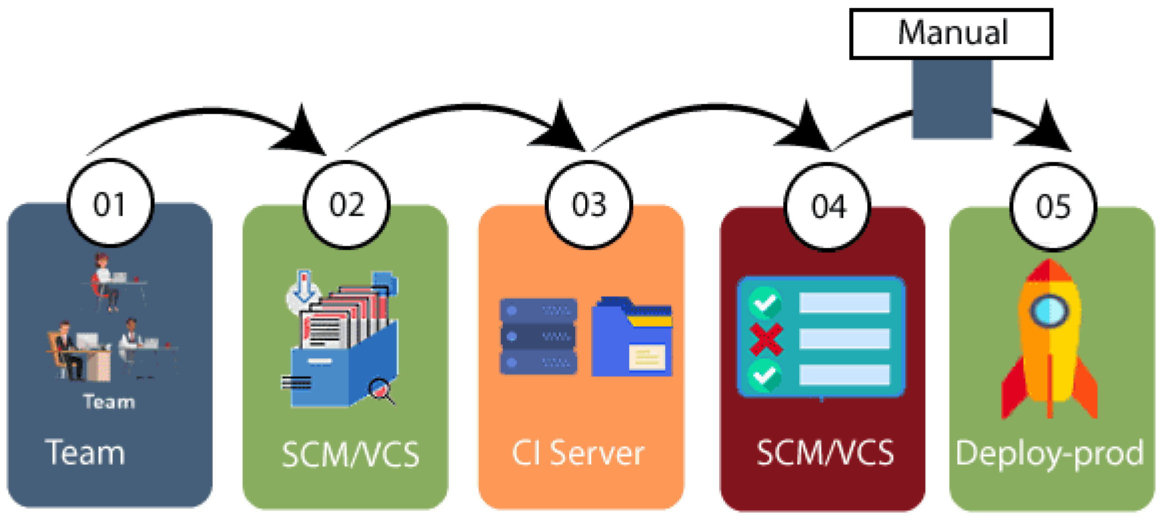
5) Continuous Feedback

The application development is consistently improved by analyzing the results from the operations of the software. This is carried out by placing the critical phase of constant feedback between the operations and the development of the next version of the current software application.

The continuity is the essential factor in the DevOps as it removes the unnecessary steps which are required to take a software application from development, using it to find out its issues and then producing a better version. It kills the efficiency that may be possible with the app and reduce the number of interested customers.

6) Continuous Deployment

In this phase, the code is deployed to the production servers. Also, it is essential to ensure that the code is correctly used on all the servers.



The new code is deployed continuously, and configuration management tools play an essential role in executing tasks frequently and quickly. Here are some popular tools which are used in this phase, such as **Chef, Puppet, Ansible**, and **SaltStack**.

Containerization tools are also playing an essential role in the deployment phase. Vagrant and Docker are popular tools that are used for this purpose. These tools help to produce consistency across development, staging, testing, and production environment. They also help in scaling up and scaling down instances softly.

Containerization tools help to maintain consistency across the environments where the application is tested, developed, and deployed. There is no chance of errors or failure in the production environment as they package and replicate the same dependencies and packages used in the testing, development, and staging environment. It makes the application easy to run on different computers.

7) Continuous Operations

All DevOps operations are based on the continuity with complete automation of the release process and allow the organization to accelerate the overall time to market continuingly.

It is clear from the discussion that continuity is the critical factor in the DevOps in removing steps that often distract the development, take it longer to detect issues and produce a better version of the product after several months. With DevOps, we can make any software product more efficient and increase the overall count of interested customers in your product.

DevOps Pipeline

A pipeline in software engineering team is a set of automated processes which allows DevOps professionals and developer to reliably and efficiently compile, build, and deploy their code to their production compute platforms.

The most common components of a pipeline in DevOps are build automation or continuous integration, test automation, and deployment automation.

A pipeline consists of a set of tools which are classified into the following categories such as:

- Source control
- Build tools
- Containerization
- Configuration management
- Monitoring

Continuous Integration Pipeline

Continuous integration (CI) is a practice in which developers can check their code into a version-controlled repository several times per day. Automated build pipelines are triggered by these checks which allows fast and easy to locate error detection.

Some significant benefits of CI are:

- Small changes are easy to integrate into large codebases.
- More comfortable for other team members to see what you have been working.
- Fewer integration issues allowing rapid code delivery.
- Bugs are identified early, making them easier to fix, resulting in less debugging work.

Continuous Delivery Pipeline

Continuous delivery (CD) is the process that allows operation engineers and developers to deliver bug fixes, features, and configuration change into production reliably, quickly, and sustainably. Continuous delivery offers the benefits of code delivery pipelines, which are carried out that can be performed on demand.

Some significant benefits of the CD are:

- Faster bug fixes and features delivery.
- CD allows the team to work on features and bug fixes in small batches, which means user feedback received much quicker. It reduces the overall time and cost of the project.

DevOps Methodology

We have a demonstrated methodology that takes an approach to cloud adoption. It accounts for all the factors required for successful approval such as people, process, and technology, resulting in a focus on the following critical consideration:

- **The Teams:** Mission or project and cloud management.
- **Connectivity:** Public, on-premise, and hybrid cloud network access.
- **Automation:** Infrastructure as code, scripting the orchestration and deployment of resources.
- **On-boarding Process:** How the project gets started in the cloud.
- **Project Environment:** TEST, DEV, PROD (identical deployment, testing, and production).
- **Shared Services:** Common capabilities provided by the enterprise.
- **Naming Conventions:** Vital aspect to track resource utilization and billing.
- **Defining Standards Role across the Teams:** Permissions to access resources by job function.

Azure DevOps

Azure DevOps is also known as Microsoft visual studio team services (VSTS). It is a set of collaborative development tools built for the cloud. VSTS was commonly used as a standalone term, and Azure DevOps is a platform which is made up of a few different products, such as:

- Azure Test Planes
- Azure Boards
- Azure Repos
- Azure Pipeline
- Azure Artifacts

Azure DevOps is everything that needs to turn an idea into a working piece software. You can plan a project with azure tools.

The azure pipeline is the CI component of azure DevOps. The azure pipeline is Microsoft's cloud-native continuous integration server, which allows teams to continuously build, test, and deploy all from the cloud. An azure pipeline can connect to any number of source code repositories such as Azure Repos, GitHub, Tests, to grab code and artifacts for application delivery.

Azure DevOps Server

Azure DevOps Server is a Microsoft product that provides version control, requirements management, reporting, lab management, project management, testing, automated builds, and release management capabilities. It covers the entire application of lifecycle and enables DevOps capabilities.

Azure DevOps can be used as a back-end to the numerous integrated development environments, but it is modified for Microsoft visual studio and eclipse on all platforms.

Azure DevOps Services

Microsoft announced the release of the software as a service offering of visual studio on the Microsoft Azure platform at the time Microsoft called it a visual studio online.

Microsoft offers visual studio, basic, and stakeholder subscriber access levels for the Azure DevOps services. The basic plan is free of cost for up to five users. Users with a visual studio subscription can be added to a project with no additional charge.

AWS DevOps

AWS is the best cloud service provider, and DevOps is the implementation of the software development lifecycle.

Here are some reasons which make AWS DevOps a highly popular combination, such as:

- AWS CloudFormation
- AWS EC2

- AWS CloudWatch
- AWS CodePipeline

Let's see all of these by one in brief such as:

AWS CloudFormation

DevOps team is required to create and release cloud instances and services more frequently in comparison to development teams. Templates of AWS resources such as EC2 instances, ECS containers, and S3 storage buckets let you set up the entire stack without having to bring everything together.

AWS EC2

You can run containers inside EC2 instances. Hence you can leverage the AWS security and management features.

AWS CloudWatch

This monitoring tool tracks every resource that AWS has to offer. It makes it easy to use third-party tools for monitoring such as sumo logic etc.

AWS CodePipeline

Code Pipeline is an essential feature from AWS, which highly simplifies the way you manage your CI/CD toolset. It integrates with tools such as **Jenkins, GitHub**, and CodeDeploy that enable you to visually control the flow of app updates from build to production.What is Agile?

The Agile involves continuous iteration of development and testing in the **SDLC** process. Both development and testing activities are concurrent, unlike the waterfall model. This software development method emphasizes on incremental, iterative, and evolutionary development.

It breaks the product into small pieces and integrates them for final testing. It can be implemented in many ways, such as **Kanban, XP, Scrum**, etc.

The Agile software development focus on the four core values, such as:

- Working software over comprehensive documentation.
- Responded to change over following a plan.
- Customer collaboration over contract negotiation.
- Individual and team interaction over the process and tools.

What is Agile?

The Agile involves continuous iteration of development and testing in the **SDLC** process. Both development and testing activities are concurrent, unlike the waterfall model. This software development method emphasizes on incremental, iterative, and evolutionary development.

It breaks the product into small pieces and integrates them for final testing. It can be implemented in many ways, such as **Kanban, XP, Scrum**, etc.

The Agile software development focus on the four core values, such as:

- Working software over comprehensive documentation.
- Responded to change over following a plan.
- Customer collaboration over contract negotiation.
- Individual and team interaction over the process and tools.

DevOps vs Agile

Parameter	DevOps	Agile
Definition	DevOps is a practice of bringing development and operation teams together.	Agile refers to the continuous iterative approach, which focuses on collaboration, customer feedback, small, and rapid releases.
Purpose	DevOps purpose is to manage end to end engineering processes.	The agile purpose is to manage complex projects.
Task	It focuses on constant testing and delivery.	It focuses on constant changes.
Team size	It has a large team size as it involves all the stack holders.	It has a small team size. As smaller is the team, the fewer people work on it so that they can move faster.
Team skillset	The DevOps divides and spreads the skill set between development and the operation team.	The Agile development emphasizes training all team members to have a wide variety of similar and equal skills.
Implementation	DevOps is focused on collaboration, so it	Agile can implement within a range

	does not have any commonly accepted framework.	of tactical frameworks such as **safe, scrum**, and **sprint**.
Duration	The ideal goal is to deliver the code to production daily or every few hours.	Agile development is managed in units of sprints. So this time is much less than a month for each sprint.
Target areas	End to End business solution and fast delivery.	Software development.
Feedback	Feedback comes from the internal team.	In Agile, feedback is coming from the customer.
Shift left principle	It supports both variations left and right.	It supports only shift left.
Focus	DevOps focuses on operational and business readiness.	Agile focuses on functional and non-functional readiness.
Importance	In DevOps, developing, testing, and implementation all are equally important.	Developing software is inherent to Agile.
Quality	DevOps contributes to creating better quality with automation and early bug removal. Developers need to follow Coding and best Architectural practices to maintain quality standards.	The Agile produces better applications suites with the desired requirements. It can quickly adapt according to the changes made on time during the project life.
Tools	**Puppet, Chef, AWS, Ansible**, and team City OpenStack are popular DevOps tools.	**Bugzilla, Kanboard, JIRA** are some popular Agile tools.
Automation	Automation is the primary goal of DevOps. It works on the principle of maximizing efficiency when deploying software.	Agile does not emphasize on the automation.
Communication	DevOps communication involves specs and design documents. It is essential for the operational team to fully understand the software release and its network implications for the enough running the deployment process.	Scrum is the most common method of implementing Agile software development. Scrum meeting is carried out daily.
Documentation	In the DevOps, the process documentation is foremost because it will send the software to an operational team for deployment. Automation minimizes the impact of insufficient documentation. However, in the development of sophisticated software, it's difficult to transfer all the knowledge required.	The agile method gives priority to the working system over complete documentation. It is ideal when you are flexible and responsive. However, it can harm when you are trying to turn things over to another team for deployment.

Index.html

```
<!DOCTYPE html>
<html>
<head>
	<title>Access Google Maps API in php</title>
	<link rel="stylesheet" href="css/style.css">



</head>
<body>

	<div id="map"></div>
	<script async defer
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBL1LscQ_XQMN5hs-Pu8abiG7fkI3ATTHU&callback=initMap"></script>
<script src="googlemap.js"></script>



</body>

</html>
```

googlemap.js

```
function initMap() {
	//map option
	var options = {
		center: {lat:20.5937 ,lng:78.9629};//lat=lattitude,lng=
		zoom: 10
	}
	//new map
	map = new google.maps.Map(document.getElementById("map"),options)
	/*
```

```
        //Marker
        const marker = new google.maps.Marker({
                position:{lat: 37.9922, lng:-1.1307},
                map:map
                icon:"https://img.icons8.com/nolan/2x/marker.png"
        });
        //Infowindow
        const detailWindow = new google.maps.Infowindow({
                content:'<h2>Murcia</h2>'
        });
//click,mouseover,onload
        marker.addListener("click",()=>{
                detailWindow.open(map,marker);
        });
        */

        //Add Marker
        /*function addMarker(property){
                const marker = new google.maps.Marker({
                        position:{lat: 37.9922, lng: -1.1307},
                        map:map,
                        icon: property.imageIcon
                });
                //check for custom icon
                if(property.imageIcon){
                        //set imageIcon

                        marker.setIcon(property.imageIcon)
                }
                detailWindow.open.(map,marker);
                addMarker({{location:lat: 37.9922, lng: -1.1307},
                    imageIcon: "https://img.icons8.com/nalan/2x/marker.png"
                    content:'<h2>Murcia City</h2>'});
                addMarker({{location:lat: 39.4699, lng: -0.3763}});
                addMarker({{location:lat: 38.5411, lng: -0.1225}});

        }*/
}
```

style.css

```
#map{
        height: 700px;
        width: 100%;
}
```

Loading the Maps JavaScript API:

The maps JavaScript API is loaded using a script tag,which can be added inline in HTML file or JavaScript file.

```
<script async

src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap">

</script>
```

Script Tag attributes:

-src:The URL where the Maps JavaScript API is loaded from,including all of the symbols and defnitions you need for using Maps JavaScript API.

In the above Script Tag we used two parameters key and callback.

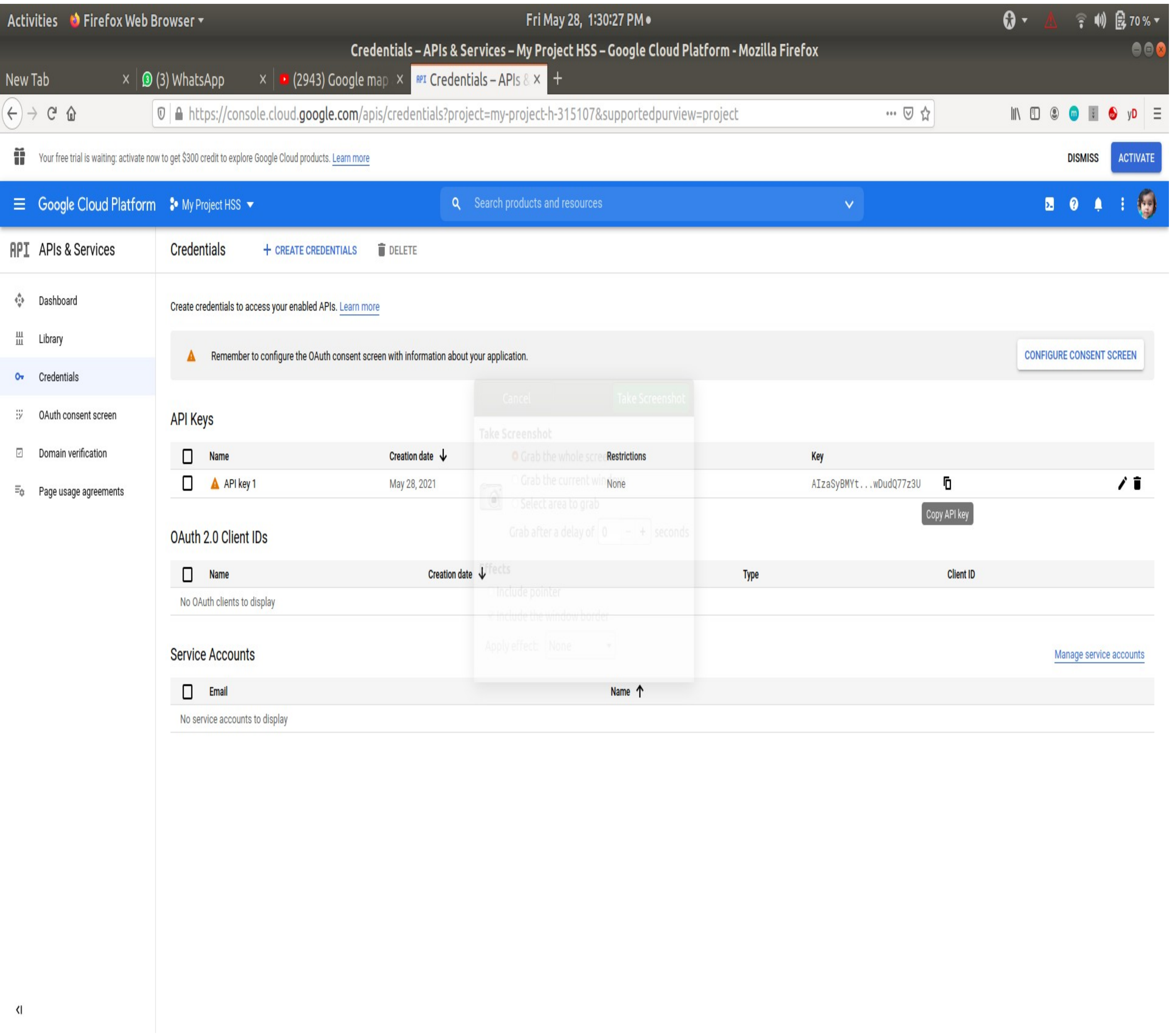
Key :where you provide your API key.

Callback:where you specify the name of the global function to be called once the Maps JavaScript API loads completely.

-in the above example callback is initMap

-async: Asks the browser to asynchronously download and execute the script. When script is executed,it will call the function specified using the callback parameter.

API created in google cloud platform



Activities Firefox Web Browser ▾ Fri May 28, 1:30:27 PM ● 70 % ▾

Credentials – APIs & Services – My Project HSS – Google Cloud Platform - Mozilla Firefox

New Tab × (3) WhatsApp × (2943) Google map × Credentials – APIs & ×

https://console.cloud.google.com/apis/credentials?project=my-project-h-315107&supportedpurview=project

Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more DISMISS ACTIVATE

Google Cloud Platform My Project HSS ▾ Search products and resources

APIs & Services Credentials + CREATE CREDENTIALS DELETE

Dashboard
Library
Credentials
OAuth consent screen
Domain verification
Page usage agreements

Create credentials to access your enabled APIs. Learn more

Remember to configure the OAuth consent screen with information about your application. CONFIGURE CONSENT SCREEN

API Keys

Name	Creation date ↓	Restrictions	Key
API key 1	May 28, 2021	None	AIzaSyBMYt...wDudQ77z3U

Copy API key

OAuth 2.0 Client IDs

Name	Creation date ↓	Type	Client ID
No OAuth clients to display			

Service Accounts Manage service accounts

Email	Name ↑
No service accounts to display	

Map document object model Elements

<div id= “map”></div>

-for the map to display on a web page,we must reserve a spot for it.Commonly we do this by creating a named div element .

Map Options

There are two required options for every map: center and zoom.

```
map=new google.maps.Map(document.getElementById('map'),{
        center: {lat:-34.397, lng: 150.644},
        zoom:8});
```

Zoom levels

The initial resolution at which to display the map is set by zoom property,where zoom 0 corresponds to a map of the Earth fully zoomed out, and larger zoom levels zoom in at a higher resolution. Specify zoom level as an integer.

Ex: zoom:8

Offering a map of the entire Earth as a single image would either require an immense map, or a small map with very low resolution. As a result, map images within Google Maps and the Maps JavaScript API are broken up into map "tiles" and "zoom levels." At low zoom levels, a small set of map tiles covers a wide area; at higher zoom levels, the tiles are of higher resolution and cover a smaller area. The following list shows the approximate level of detail you can expect to see at each zoom level:

1: World

5: Landmass/continent

- 10: City
- 15: Streets
- 20: Buildings

The Map Object

```
map = new google.maps.Map(document.getElementById("map"),options)
```

The JavaScript class that represents a map is the Map class.Objects of this class define a single map on a page.We create a new instance of this class using the JavaScript new operator.

When you create a new map instance, you specify a <div> HTML element in the page as a container for the map. HTML nodes are children of the JavaScript document object,and we obtain a reference to this element via the document.getElementById() method.

REFERENCES:

1.www.google.com/

2.https://www.javatpoint.com/aws-tutorial

3. https://aws.amazon.com

4.https://github.com/quora

5.https://www.javatpoint.com/github

6.https://www.edureka.co/blog/devops-tutorial

7.https://www.javatpoint.com/devops

ARTICLE

Limitations in DEVOPS Implementation Techniques Of CI/CD

Tejadeep(r161817)
Sireesha(r161177)

December 2021

Abstract

This article provides the information about DevOps techniques and implementation process of CI/CD .DevOps is all about Cul- ture,Automation,Measurement and Shar- ing(CAMS),It's gaining popularity because of its Continuous approach.Several factors prevent the organization from adapting these approches like lack of fully automated acceptance test,the rollback methodol- ogy,manually driven quality check etc.This reasearch paper provides various test meth- ods like regression,smoke testings and tools that can constitute an effective CD/CI pipeline.To provide security we are using se- curity testing techniques like (Web Appli- cation Security Scanning)WAST,(Security API Scanning)SAS,(Behaviour Driven Se- curity Testing)BDST.

1 Introduction

Continuous Integration (CI) and Continu- ous Delivery (CD)are software engineering processes used in DevOps [10] in order to improve the efficiency of projects . The CI/ CD processes can be implemented at one of three overarching degrees of automa- tion. The first covering development and testing(Continuous Integration), the second extending this with auto-mated integration testing (Continuous Delivery) and the third adding continuous deployment to a produc- tion environment (continuous Deployment).

The Continuous Integration stage starts with a commit,followed by a build of the modified application which is then verified using unit testsW˙ hen all test cases pass, the tested application is deployed to a test- ing environment. If Continuous Delivery is implemented, a set of automated accep- tance tests is executed to verify that there are no regressions in the system's features. This step also helps to identify any errors that may occur due to a difference in run- time environment because the testing en- vironment is usually a server with similar configu-ration to the production environ- ment. Depending on the level of automa- tion, this step can also involve manual test- ing and approval before the pipeline ad- vances to the next stage. When all pre- vious stages have passed, the system is au- tomatically deployed to the production en- vironment in the Continuous Deployment stage, where the users will have access to the new version. If a test fails, or when an error occurs during build or deployment, the pipeline is automatically stopped and developers are notified of the error. When a fix has been committed the pipeline will start all over again in order to test the en- tire application.

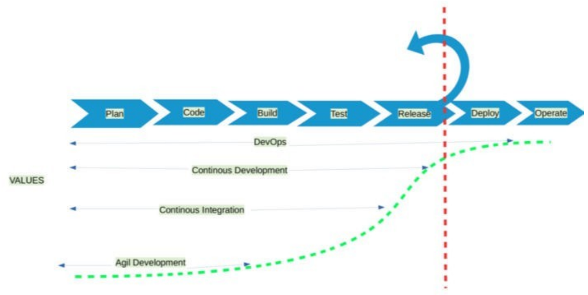


Figure 1: Collaboration Diagram

The ideal release pipeline:

- Increases product quality
- Increases the agility of the product's development
- Reduces the stress and last-minute panic associated with product releases.

DevOps method of color-coding the vari- ous stages of the release pipeline, while the next section identifies five critical points of the pipeline where automation can signifi- cantly boost productivity, agility, and prod- uct quality

A standard release pipeline consists of the main elements or environments listed below. Depending on the software and compliance requirements, environments might be added or removed from the pipeline, either permanently or by demand. 1.Local development environments; the individual developers' personal computers

2. A Continuous Integration (CI) server, or development server
3. Test servers for functional UI testing of the product.A production environment.

A gate is a set of requirements that must be met for a build (a piece of code) to pass to the next stage, or environment, of the pipeline. For example, a gate could set the following requirements:

- The code should build without errors
- All unit tests must pass
- All functional UI tests must pass.

all stages and gates of the CD release pipeline are color-coded to make it very easy to identify and understand the flows and checks of the process.

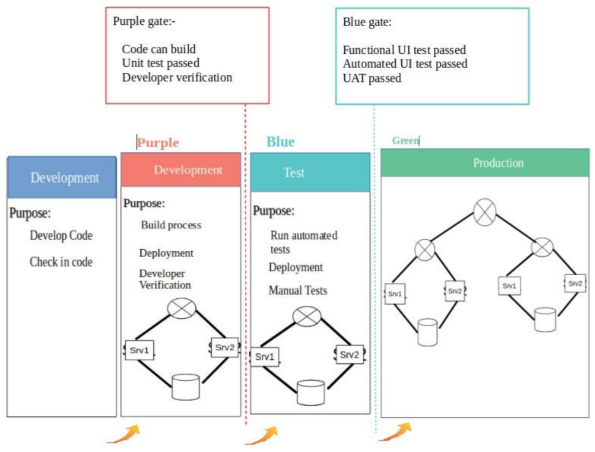


Figure 2: An example of a release pipeline

1 DEVELOPMENT

This is where the code is built on a build server and deployed to a CI server [3].

2.1 Purple Gate

After development, the code is ‘checked in’ at the Purple Gate. To pass this gate, the following requirements must be met:

- The code can build.
- The code must have passed all unit tests.
- The code must have passed a local verification by the individual developer.

The checks of the Purple Gate are very crit- ical because the Development environment is designed to imitate the Production environment. This means, if the code works in Development, it is likely to work in Production.

More requirements can be added to the Purple Gate. In any case, if any of the checks fail, the check-in should be rolled back and the move to Test should be can- celled.

3 TEST:

This environment is for product tests of all kinds [4]. It can contain multiple server setups to accommodate for various test types: functional tests, performance tests, load tests, etc.

3.1 Blue Gate

To pass this gate, the following require- ments must be met:

- The code must have passed all automated functional UI tests (regression testing)
- The product must have passed a verifica- tion of its visual appearance
- The server log files must have been inspected after regression tests
- Performance data must have met an acceptable benchmark

Again, more requirements can be added to this gate. If any of the checks fail,the check-in should be rolled back and the move to Production should be cancelled.

4 PRODUCTION

Smoke testing is for identifying any bugs before the end-user does [8]. Obviously, this improves the end-user experience as the product is better received.

Automating the continuous delivery pipeline:

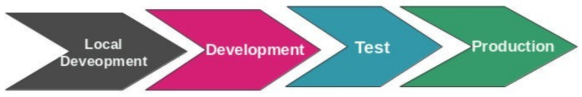


Figure 3: The color-coded release pipeline, summarized

The effectiveness of a CD release pipeline depends on how fast the results of the Red and Blue Gate checks can feed back to rel- evant stakeholders.

For example, if a unit test fails in Development, the developer who did the check-in should be notified immediately by an email alert. Another example, if the automated UI tests in Test fail, an alert could appear on a dashboard monitored by the team re- sponsible.

The faster the feedback mechanism, the more likely it is to find a quick solution to the given issue. If a developer is notified weeks after error detection in a checked- in piece of code, he or she has most likely moved on to other tasks and projects, and the context and relevant requirements of the given code is not on top of his or her mind. The developer would have to spend time re-adjusting to the right context, re- sulting in significant productivity loss.

The answer to how to speed up the feedback mechanism in the release pipeline is automation. Ideally, all the automation needs of the product delivery process should be manageable by using one tool that enhances the work of all the roles involved in the pipeline; from testers to product and business owners.

4.1 REGRESSION TEST- ING

The primary area of automated testing is regression testing [7]. This helps ensure that changes to a product base in not creat- ing bugs in the existing product. Regression testing usually takes in place in the Test en- vironment.

By relying on test, and without having to having a write a single line of code, the test team can automate and maintain the test cases required for passing the Blue Gate. The cases can run on a schedule, i.e. ev- ery night, repeatedly 24/7, or on an ad hoc basis.

The Test Automation Platform comes with native plugins to the common build and release services such as Jenkins, TFS, Atlassian Bamboo, and TeamCity. This makes it very easy to trigger test cases to run as part of a release plan. Furthermore, testing returns all test case results in JUnit format, which is natively supported by all platforms, enabling an overview of all test cases directly via any release platform an organization is using.

As mentioned, automation is meant for making feedback instantly available. After test execution, all case results are available in the Test Automation Platform, and it is possible to filter them by projects, sched- ules, and more. The results can also be emailed to designated stakeholders depend- ing on the case outcome.

Finally, test results can be automatically pushed to whichever bug management system is used (TFS, Jira, HP Quality Center, etc.).

EARLY REGRESSION / DEVEL-OPER VERIFICATION

Running regression [1] tests against the Development environment serves two purposes:

- Making sure that the build is testable, that is, ensuring that it makes sense to push to Test.
- Providing feedback to developers about whether the Development environment is reliable; if the code works here it can be assumed that it also works in Test and Production.

Early regression testing only covers a small part of the total regression tests avail- able, for two reasons.

Firstly, because there is a higher fre- quency of deployment, or check-ins of code, to Development, there is an upper limit to the number of early regression tests that can be executed per day. Secondly, veri- fication at this stage is usually kept at a limited number of basic test cases focusing on critical features.

The schedules for early regression testing are usually triggered as part of the same processes that generates the build and deploys the code to the Development server.

FEATURE REGRESSION / INDIVIDUAL DEVELOPER VERIFICATION

The third type of regression testing that is easily automated with feature verification on the individual developer's local installation [5]. With teams of testers, developers, and DevOps professionals can share and collaborate on test cases, and everyone can run them on their individual PCs if needed. The results of local runs provide valuable feedback to the entire team.

4.1 SMOKE TESTING

"Errors will occur." This is a well-known and accepted truth in software production. So, during a release cycle, it is never a ques- tion of "if errors will occur", but rather how the organization will react on them. This is about running automated test cases against the Production eof smoke testing [6]. This is about running automated test cases against the Production environment to ensure that the software indeed runs as intended. Tests done at this stage are the closest thing to real user interaction. An example of a test case could be to perform a user login to a web application and then validate the login. The test cases used in smoke testing are solely used for this pur- pose.

The outcomes of the tests are not allowed to dictate any drastic changes to the prod- uct. Smoke tests should be able to run both frequently and quickly, and they should not be allowed to push the production system to its limits.

Advantages:Defects will be identified in early stages,Saves test effort and time,Easy to detect critical errors and correction of errors,runs quickly,minimizes integration risks.

Challenge: Some defects may be encoun- tered in later stages where it can be cost effective.

The three dynamic ap- plication security test- ing techniques we in- tegrated into a CI/CD pipeline are:

1. Web Application Security Scanning (WAST) using Zed Attack Proxy (ZAP) 1
,
2. Security API Scanning (SAS) with JMe- ter 2 and
3. Behaviour Driven Security Testing (BDST) using SeleniumBase automation framework.

5.1 1.Wed Application Secu- rity Scanning(WAST):

-This testing technique is an automated web security test that attacks a web application through its user interface. -It includes 3 steps performed by WAST [2] component:

a. Spider scan : The spider Scan explores the whole application in order to determine all URLs/resourse available.
b. Active scan: The active Scan performes malicious requests againest every identified

resourse and evaluates every response of the application in order to determine possible security issues on the targeted URL.

c. Result reporting: Once the active scan is completed,the results are aggrigated into a report.

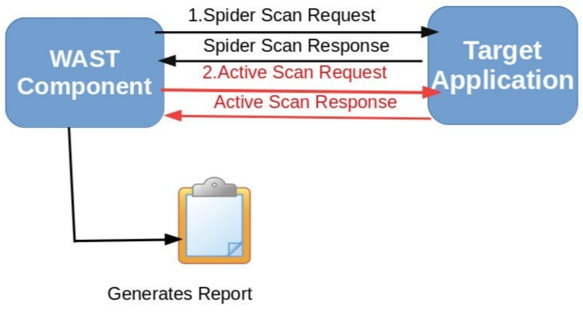


Figure 4: Overview of WAST security test- ing technique: The spider scan determines all available components and the active scan attacks them.

5.1 2.Security API Scan- ning(SAS):

-The WAST technique scans the entire web application but may not detect flaws of the underlying back-end (web) services.

-This technique allows testing of every endpoint in great detail and can cover multiple security relevant cases such as authentication, input validation, or error handling. -In SAS [2], a parameterized request is generated and sent to the API of the web service that is under test through a request component. -The input data can vary from credentials for authentication to malicious payloads such as SQL injection (SQLi). -All the requests go through a proxy component that intercepts traffic between the request component and the target application. -The proxy component evaluates all the intercepted traffic for any security issues. -After the test is performed, the proxy component reports the result of the evaluation. - SAS testing is especially useful when generated fuzz data is used as input. Fuzz data can be a list of the most common passwords, a bulk of random data in order trigger unexpected behaviour of the system,or malicious input for SQLi.

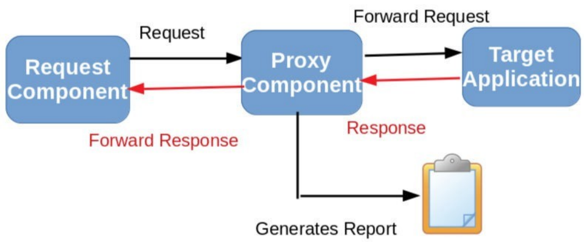


Figure 5: Overview of SAS testing tech- nique: Request component sends request through proxy component to target appli- cation. Proxy component evaluates traffic and generates report.

5.2 3.Behaviour Driven Secu- rity Testing(BDST):

-Behaviour Driven Development(BDD) is an extension of Test Driven Devel- opment(TDD) and follows the idea of integrating business insights into test- ing. -BDD uses a natural language approach in order to define behaviour

and expected outcome of test cases. - Behaviour Driven Security Testing(BDST) [9] applies idea of BDD to the domain of security testing for the added benefit that non-security experts can understand the security tests,further improving the collaboration between security experts and DevOps teams. -Aditionally,BDST provides a dynamic security documentation of the whole software system due to the GWT(Given,When,Then) format of the test specifiactions. -In UI testing, BDD frameworks are used to automate standard UI tests that mimic the user behavior.
-This approach can be used to automate the execution of attack scenarios from the hacker’s perspective. Because this technique is executed against the system as a whole, it enables the identification of vulnerabilities that target multiple entry- points to the system. -BDST combines several security testing techniques such as SAS or WAST in order to mimic attack scenarios by a hacker, as well as to find security issues during normal system usage.

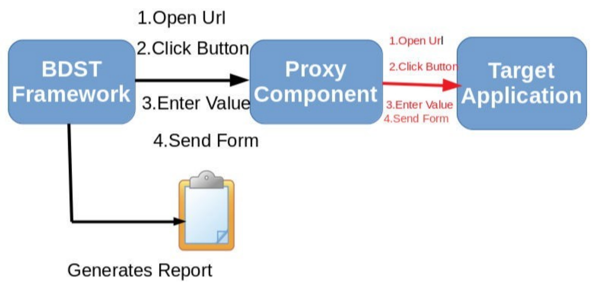


Figure 6: Overview of BDST testing tech- nique: The BDST framework sends be- haviour driven requests to the target appli- cation through the proxy component which then scans for security flaws.

Advantages:Testing can be auto- mated,the fixed bugs and issues do not reoccur.

6 Challenges in DevOps

By understanding all the above tech- niques and implementation process we encounter some challenges those are Ambiguous,Management Struc- ture,training,Experience,Complexity,Time Consumtion etc.

S.No	Challenge	Description
1.	Ambiguous:	Since DevOps is a new concept that's unclear on its definition, goals
2.	Management Structure:	Between developers and operators, they lack management structure
3.	Training:	As it's a new concept most operators and Developers lack proper training of the whole DevOps operations and Principles hence new technology tools and methods of DevOps
4.	Experience:	There is a shortage of DevOps experienced individuals hence the whole concept is learned and practices to obtain experience at present time and apply in the project,
5.	Complexity:	with addition of functionality the product gets more com- plex.Hence,the amount of regression testing to be
6.	Time Con- suming:	Regression testing involves running existing test cases,it takes a lot

Table 1: Challenges in Devops

7 Conclusion

This Paper Provides Information about Continuous Integration and Continuous Deployment of code.
Continuous Integration doesn't get rid of bugs,but it does make them dramatically easier to find and remove.
The main purpose of Continuous Deployment is to increase the speed at which we are getting feedback from our customers and clients and responding to that feedback.
Regression Testing is an essential aspect of a dynamic,iterative development and deploymentsoftwareprocess.Withsmoketesting, we simplify the detection and correction of major defects.By using CI/CD techniques we can get efficient product or application.

References

[1] Dirk Beyer, Stefan Lo¨we, Evgeny Novikov, Andreas Stahlbauer, and Philipp Wendler. Precision reuse for efficient regression verification. In *Pro- ceedings of the 2013 9th Joint Meeting on Foundations of Software Engineer- ing*, pages 389–399, 2013.

[2] Remco v Buijtenen and Thorsten Rangnau. Continuous security testing: A case study on the challenges of integrating dynamic security testing tools in ci/cd. *17th SC@ RUG 2019-2020*, page 45, 2019.

[3] Martin Fowler and Matthew Foemmel. Continuous integration, 2006.

[4] Chaitanya Kallepalli and Jeff Tian.

Measuring and modeling usage and reliability for statistical web testing. *IEEE transactions on software engi- neering*, 27(11):1023–1036, 2001.

[5] M Karagiannopoulos, D Anyfantis, SB Kotsiantis, and PE Pintelas. Feature selection for regression problems. *Proceedings of the 8th Hellenic European Research on Computer Math- ematics & its Applications, Athens, Greece*, 2022:13, 2007.

[6] Jesse D Kosiba, Emily L Zale, and Joseph W Ditre. Associations between pain intensity and urge to smoke: Test- ing the role of negative affect and pain catastrophizing. *Drug and alcohol de- pendence*, 187:100–108, 2018.

[7] Hareton KN Leung and Lee White. In- sights into regression testing (software testing). In *Proceedings. Conference on Software Maintenance-1989*, pages 60– 69. IEEE, 1989.

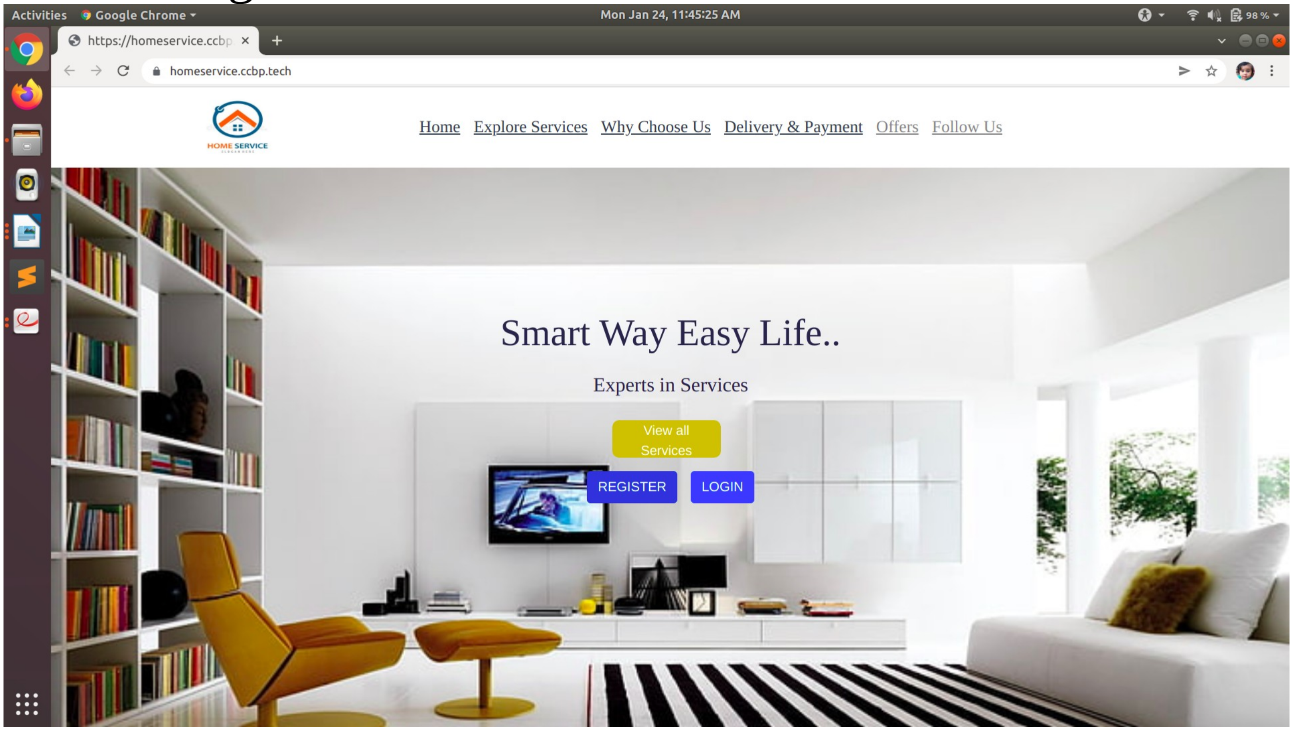
[8] Ian Molyneaux. *The art of application performance testing: from strategy to tools*. " O'Reilly Media, Inc.", 2014.

[9] Sandeep Sivanandan et al. Agile de- velopment cycle: Approach to design an effective model based test- ing with behaviour driven automation framework. In *20th Annual In- ternational Conference on Advanced Computing and Communications (AD- COM)*, pages 22–25. IEEE, 2014.

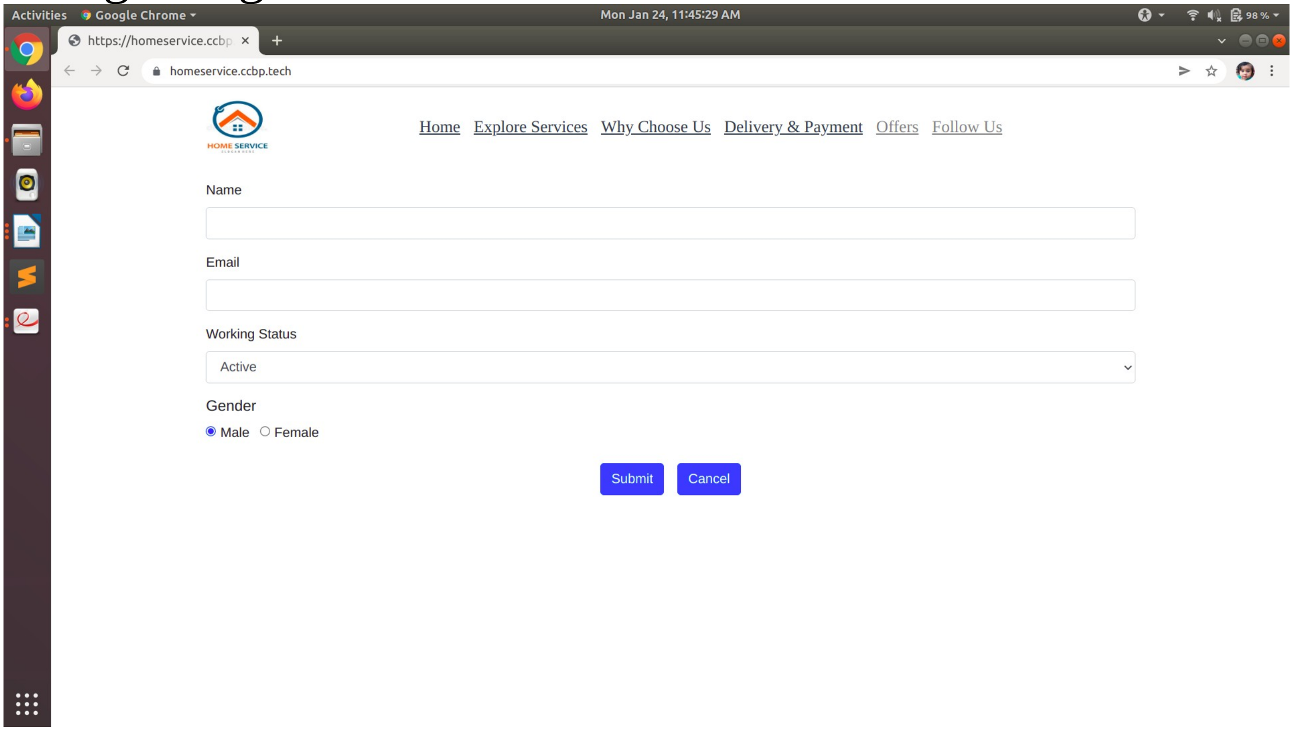
[10] Manish Virmani. Understanding de- vops & bridging the gap from contin- uous integration to continuous deliv- ery. In *Fifth international conference on the innovative computing technol- ogy (intech 2015)*, pages 78–82. IEEE, 2015.

Project

1.HomePage



2.Login Page:



Name

Email

Working Status

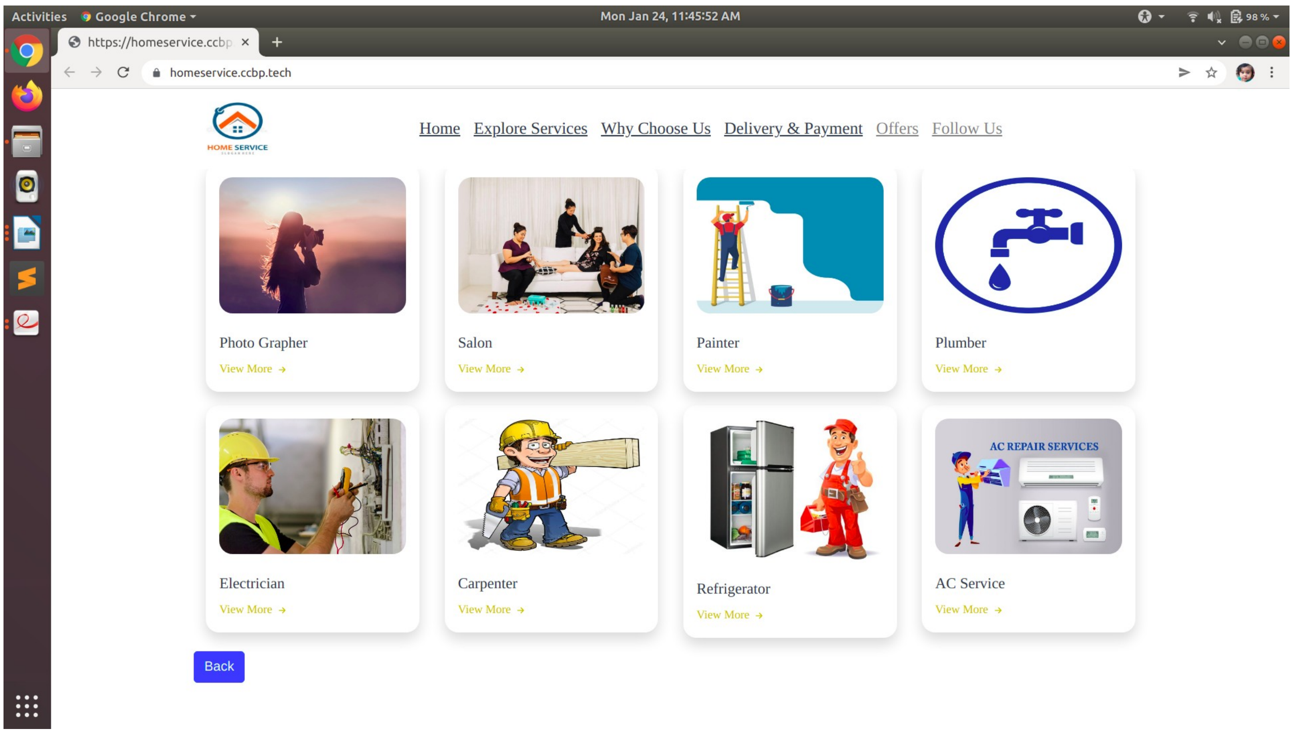
Active

Gender

[✓ Male] Female

Submit Cancel

3.Services Page:



Code:

```
<!DOCTYPE html>
<html>

<head>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/cs
s/bootstrap.min.css" integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN
0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous" />
    <script src="https://code.jquery.com/jquery-
3.5.1.slim.min.js" integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBn
```

```
E+IbbVYUew+OrCXaRkfj"
crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/um
d/popper.min.js"
integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H
66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN"
crossorigin="anonymous"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/b
ootstrap.min.js" integrity="sha384-
B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofv
L8/KUEfYiJOMMV+rV"
crossorigin="anonymous"></script>
    <script src="https://kit.fontawesome.com/20c5629a29.js"
crossorigin="anonymous"></script>
</head>

<body>
    <nav class="navbar navbar-expand-lg navbar-light bg-
white fixed-top">
        <div class="container">
            <a class="navbar-brand" href="#">
                <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/v
1627539197/Screenshot_from_2021-07-29_11-42-
37_ujvxrg.png" class="food-munch-logo" />
            </a>
            <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNavAltMarkup"
```

```
aria-controls="navbarNavAltMarkup" aria-
expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse"
id="navbarNavAltMarkup">
            <div class="navbar-nav m-auto navheading ml-
5">
                <a class="nav-link active" id="navItem1"
href="#">
                    Home
                    <span class="sr-only">(current)</span>
                </a>
                <a class="nav-link" href="#services"
id="navItem2">Explore Services</a>
                <a class="nav-link" href="#WhyChooseUs"
id="navItem3">Why Choose Us</a>
                <a class="nav-link" href="#Del"
id="navItem4">Delivery & Payment</a>
                <a class="nav-link" href="#Offers"
id="navItem5">Offers</a>
                <a class="nav-link" href="#followUs"
id="navItem6">Follow Us</a>
            </div>
        </div>
      </div>
    </nav>
    <div id="sectionbanner">
      <div class="banner-section-bg-container d-flex justify-
content-center flex-column ">
        <div class="text-center">
```

```
            <h1 class="banner-heading mb-3">Smart Way
Easy Life..</h1>
            <p class="banner-caption mb-4">Experts in
Services</p>
            <button class="custom-button"
onclick="display('section1')">View all Services</button>
         </div>
         <div class="d-flex flex-row justify-content-center
mt-3">
            <button class="btn btn-primary mr-3"
onclick="display('sectionregisterPage')">REGISTER</butt
on>
            <button id="login" class="btn btn-
primary">LOGIN</button>
         </div>
      </div>
   </div>
   <div id="sectionregisterPage">
      <div class="container">
         <h1 class="form-heading">New User
Registration</h1>
         <form id="myForm">
            <div class="mb-3">
               <label for="name">Name</label>
               <input type="text" class="form-control"
id="name" />
               <p id="nameErrMsg" class="error-
message"></p>
            </div>
            <div class="mb-3">
               <label for="email">Email</label>
```

```
                <input type="text" class="form-control"
id="email" />
                <p id="emailErrMsg" class="error-
message"></p>
            </div>
            <div class="mb-3">
                <label for="status">Working Status</label>
                <select id="status" class="form-control">
                    <option value="Active">Active</option>
                    <option value="Inactive">Inactive</option>
                </select>
            </div>
            <div class="mb-3">
                <h1
class="gender-field-heading">Gender</h1>
                <input type="radio" name="gender"
id="genderMale" value="Male" checked />
                <label for="genderMale">Male</label>
                <input type="radio" name="gender"
id="genderFemale" value="Female" class="ml-2" />
                <label for="genderFemale">Female</label>
            </div>
            <div class="d-flex flex-row justify-content-
center">
                <button class="btn btn-primary mr-
3">Submit</button>
                <button class="btn btn-primary"
onclick="display('sectionbanner')">Cancel</button>
            </div>
        </form>
    </div>
</div>

<div id="services" class="d-none">
    <div class="bg">
        <div class="container">
            <div class="row">
                <div class="col-12">
                    <h1 class="h1 pt-5">Our Services</h1>
                    <p class="p">Most online stores offer lower
prices.Online shopping makes price comparision simpler
and quicker.It is very convenient to shop from where you
are located.It saves you the cost of driving to stores, as well
as parking fees.</p>
                </div>
                <div class="col-12 col-md-4">
                    <div class=" shadow card text-center mb-
3">
                            <div class="d-flex flex-row justify-
content-center">
                                <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/ecommerce-services-delivery-img.png"
class=" img" />
                            </div>
                            <h1 class="h2">Fast and Free Delivery</
h1>
                            <p class="p2">Fast,free, and convenient
delivery choices on millions of times.</p>
                    </div>
                </div>
                <div class="col-12 col-md-4">
                    <div class=" shadow card text-center mb-3
">
                        <div class="d-flex flex-row justify-
content-center">
                            <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/ecommerce-services-money-back-
img.png" class=" img" />
                        </div>
                        <h1 class="h2">100% Money Back
Guarantee</h1>
                        <p class="p2">This is probably The most
popular guarantee in the world.</p>
                    </div>
                </div>
                <div class="col-12 col-md-4">
                    <div class=" shadow card text-center mb-3
">
                        <div class="d-flex flex-row justify-
content-center">
                            <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/ecommerce-services-24-by-7-support-
img.png" class=" img" />
                        </div>
                        <h1 class="h2">Online Support
24/7</h1>
                        <p class="p2">Our online support will
provide you with many useful functions,valuble
information and tips.</p>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="wcu-section pt-5 pb-5 d-none"
id="WhyChooseUs">
    <div class="container">
        <div class="row">
            <div class="col-12">
                <h1 class="wcu-section-heading">Why
Choose Us?</h1>
                <p class="wcu-section-description">
                    We provide best services with fast manner
and we shoud take care about safety,comfort to customers
                </p>
            </div>
            <div class="col-12 col-md-4">
                <div class="wcu-card p-3 mb-3">
                    <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/v
1628348638/images_vpmdlg.jpg" class="wcu-card-
image" />
                    <h1 class="wcu-card-title mt-3">Home
Services</h1>
                    <p class="wcu-card-description">
                        Experience workers in all services . All
our orders are carefully arranged to give you the nothing
less than perfect.
                    </p>
                </div>
            </div>
            <div class="col-12 col-md-4">
               <div class="wcu-card p-3 mb-3">
                  <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_225,w_225/v1628408936/residential-cleaning-
services_vaqsuz.png" class="wcu-card-image" />
                  <h1 class="wcu-card-title mt-3">Deep
Cleaning</h1>
                  <p class="wcu-card-description">
                     Deep cleaning is the most basic and
necessary part of keeping our house clean. It helps us to
control the growth and spread of infection and viruses.
                  </p>
               </div>
            </div>
            <div class="col-12 col-md-4">
               <div class="wcu-card p-3 mb-3">
                  <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_225,w_225/v1628409100/special-offer-star-
7527442_hexyrt.jpg" class="wcu-card-image" />
                  <h1 class="wcu-card-title mt-3">Best
Offers</h1>
                  <p class="wcu-card-description">
                     Service Coupons & Offers upto
                     <span class="offers">50% OFF</span>
                     and Exclusive Promo Codes on All Online
Service Orders.
                  </p>
               </div>
            </div>
          </div>
        </div>
      </div>
      <div class="delivery-and-payment-section pt-5 pb-5 d-
none" id="Del">
        <div class="container">
          <div class="row">
            <div class="col-12 col-md-5 order-1 order-md-
2">
              <div class="text-center">
                <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/delivery-payment-section-img.png"
class="delivery-and-payment-section-img" />
              </div>
            </div>
            <div class="col-12 col-md-7 order-2 order-md-
1">
              <h1 class="delivery-and-payment-section-
heading">
                Delivery and Payment
              </h1>
              <p class="delivery-and-payment-section-
description">
                Enjoy hassle-free payment with the
plenitude of payment options
                available for you. Get live tracking and
locate your order on a
                live map. It's quite a sight to see your order
arrive to your door.
```

```
                        Plus, you get a 5% discount on every order
every time you pay
                        online.
                    </p>
                    <button class="custom-button">Order
Now</button>
                    <div class="mt-3">
                        <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/visa-card-img.png" class="payment-
card-img" />
                        <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/master-card-img.png" class="payment-
card-img" />
                        <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/paypal-card-img.png" class="payment-
card-img" />
                        <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/american-express-img.png"
class="payment-card-img" />
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div class="thanking-customers-section pt-5 pb-5 d-
none" id="Offers">
        <div class="container">
```

```
      <div class="row">
        <div class="col-12 col-md-7 d-flex flex-column
justify-content-center">
          <h1 class="thanking-customers-section-
heading">
            Thank you for being a valuable customer to
us.
          </h1>
          <p class="thanking-customers-section-
description">
            We have a surprise gift for you
          </p>
          <div class="d-md-none">
            <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/thanking-customers-section-img.png"
class="thanking-customers-section-img" />
          </div>
          <div>
            <button class="custom-button">Redeem
Gift</button>
          </div>
        </div>
        <div class="col-12 col-md-5 d-none d-md-
block">
          <img
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-
responsive-website/thanking-customers-section-img.png"
class="thanking-customers-section-img" />
        </div>
      </div>
```

```
        </div>
    </div>
    <div id="followUs" class="d-none">
        <div class="follow-us-section pt-5 pb-5"
id="followUsSection">
            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <h1 class="follow-us-section-
heading">Follow Us</h1>
                    </div>
                    <div class="col-12">
                        <div class="d-flex flex-row justify-content-
center">
                            <div class="follow-us-icon-container">
                                <i class="fab fa-twitter icon"></i>
                            </div>
                            <div class="follow-us-icon-container">
                                <i class="fab fa-instagram icon"></i>
                            </div>
                            <div class="follow-us-icon-container">
                                <i class="fab fa-facebook icon"></i>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
        <div class="footer-section pt-5 pb-5">
            <div class="container">
                <div class="row">
```

```
                    <div class="col-12 text-center">
                        <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/v
1627539197/Screenshot_from_2021-07-29_11-42-
37_ujvxrg.png" class="food-munch-logo" />
                        <h1 class="footer-section-mail-
id">orderservice@service.com</h1>
                        <p class="footer-section-address">
                            123 Y.S.R.Kadapa,Andhrapradhesh, India.
                        </p>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div id="section1">
        <div class="explore-menu-section pt-5 pb-5">
            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <h1 class="menu-section-
heading">ExploreServices</h1>
                    </div>
                    <div class="col-12 col-md-6 col-lg-3">
                        <div class="shadow menu-item-card p-3
mb-3">
                            <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_800,w_1100/v1627626639/e1_gxo0a4.jpg"
class="menu-item-image w-100" />
```

```
                    <h1 class="menu-card-title">Photo
Grapher</h1>
                    <a href="" class="menu-item-link">
                      View More
                      <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right-short"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                        <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                      </svg>
                    </a>
                  </div>
                </div>
                <div class="col-12 col-md-6 col-lg-3">
                  <div class="shadow menu-item-card p-3
mb-3">
                    <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_800,w_1100/v1627626637/e8_qtaeqn.jpg"
class="menu-item-image w-100" />
                    <h1 class="menu-card-title">Salon</h1>
                    <a href="" class="menu-item-link">
                      View More
                      <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                        <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
```

```
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                        </svg>
                     </a>
                  </div>
               </div>
               <div class="col-12 col-md-6 col-lg-3">
                  <div class="menu-item-card shadow p-3
mb-3">
                     <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_800,w_1100/v1627626637/e2_ok5bgl.jpg"
class="menu-item-image w-100" />
                     <h1
class="menu-card-title">Painter</h1>
                     <a href="" class="menu-item-link">
                        View More
                        <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                           <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                        </svg>
                     </a>
                  </div>
               </div>
               <div class="col-12 col-md-6 col-lg-3">
                  <div class="menu-item-card shadow p-3
mb-3">
```

```
                    <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_800,w_1100/v1627626638/e3_vj5aln.jpg"
class="menu-item-image w-100" />
                    <h1
class="menu-card-title">Plumber</h1>
                    <a href="" class="menu-item-link">
                      View More
                      <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                        <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                      </svg>
                    </a>
                  </div>
                </div>
                <div class="col-12 col-md-6 col-lg-3">
                  <div class="menu-item-card shadow p-3
mb-3">
                    <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/v
1627626638/e5_a24rgu.jpg" class="menu-item-image w-
100" />
                    <h1 class="menu-card-title">Electrician</
h1>
                    <a href="" class="menu-item-link">
                      View More
                    <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                      <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                    </svg>
                  </a>
                </div>
              </div>
              <div class="col-12 col-md-6 col-lg-3">
                <div class="menu-item-card shadow p-3
mb-3">
                  <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_800,w_1100/v1627626638/e4_fhfsed.jpg"
class="menu-item-image w-100" />
                  <h1 class="menu-card-title">Carpenter</
h1>
                  <a href="" class="menu-item-link">
                    View More
                    <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                      <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                    </svg>
                  </a>
```

```
            </div>
          </div>
          <div class="col-12 col-md-6 col-lg-3">
            <div class="menu-item-card shadow p-3
mb-3">
              <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/v
1627626636/e6_zyebwd.jpg" class="menu-item-image w-
100" />
              <h1 class="menu-card-
title">Refrigerator</h1>
              <a href="" class="menu-item-link">
                View More
                <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                  <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                </svg>
              </a>
            </div>
          </div>
          <div class="col-12 col-md-6 col-lg-3">
            <div class="menu-item-card shadow p-3
mb-3">
              <img
src="https://res.cloudinary.com/dopqeywgh/image/upload/c
_scale,h_800,w_1100/v1627626637/e7_wpvgzy.jpg"
class="menu-item-image w-100" />
```

```
                            <h1 class="menu-card-title">AC
Service</h1>
                            <a href="" class="menu-item-link">
                                View More
                                <svg width="16px" height="16px"
viewBox="0 0 16 16" class="bi bi-arrow-right"
fill="#d0b200" xmlns="http://www.w3.org/2000/svg">
                                    <path fill-rule="evenodd" d="M4
8a.5.5 0 0 1 .5-.5h5.793L8.146 5.354a.5.5 0 1 1 .708-.708l3
3a.5.5 0 0 1 0 .708l-3 3a.5.5 0 0 1-.708-.708L10.293
8.5H4.5A.5.5 0 0 1 4 8z" />
                                </svg>
                            </a>
                        </div>
                    </div>
                    <buttong class="btn btn-primary"
onclick="display('sectionbanner')">Back</button>
                </div>
            </div>
        </div>
    </div>
    <script type="text/javascript"
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-static-
website/js/ccbp-ui-kit.js"></script>
</body>

</html>
```