Practical – 1

Aim: Prepare the project abstract and identify the appropriate process model.

Abstract

Domain:

Productivity and event manager

Specific software reference:

Google Calendar

Features of Google Calendar:

- Scheduling events or google meets
- Create Recurring reminders/ events/ tasks
- Overview of the day/ week/ month/ year and to plan the tasks in efficient manner
- Collaboration and sharing of schedules between group
- Providing abstraction to users for Google meets
- Deactivate and activate additional calendars (for example: display of holidays - religious, national)
- Mail-notifications and alarms for events, tasks and reminders

Impact on society

- It is a perfect tool for not only that we can see the deadline but also we can receive some alerts due to the proximity of one deadline, and everybody can see the evolution of other projects and how the rest of the groups are working. It helps in organizing meets and increasing productivity.

Appropriate process model

Spiral model

Why Spiral model:

- The project is long
- Project needs user feedback and interaction *(For example, for events creation, the developers working on adding "create an event" feature but the client or user might be wishing to add recurring events feature)*
- Risk analysis is needed to be done
- Multiple iterations required as there should be multiple rounds of improvement
- Requirements are not clear from initial phase

Practical – 2

Aim: To Prepare Software Requirement Specification for your project Definition

Table of Contains

1. Introduction

1.1 Purpose

The main purpose of Google Calendar is to quickly schedule meetings and events and get reminders about upcoming activities, so as to always know what's next. Calendar is designed for teams, so it's easy to share the schedule with others and create multiple calendars that the team can use together.

1.2 Document Conventions

HTML – HyperText Markup Language
CSS – Cascading Style Sheets
JS – JavaScript
CRUD – Create, Read, Update, Delete
RAM – Random Access Memory
CPU – Central Processing Unit

1.3 Intended Audience and Reading Suggestions

This document is intended for an audience such as developers, project managers, marketing staff, users, testers, and documentation writers. The sequence for reading the document is beginning with the overview sections and proceeding through the sections.

- **Developers:** In order to be sure, they are developing the right project that fulfil requirements provided in this document.
- **Testers:** in order to have an exact list of the features and functions that have to respond according to requirements and provided diagrams.
- **Users:** in order to get familiar with the idea of the project and suggest other features that would make it even more functional.
- **Documentation writers:** To know what features & in what way they have to explain. What security technologies are required, how the system will respond in each user’s action etc.
- **Admin, Manager and Customers:** in order to know exactly what they have to expect from the system, right inputs and outputs and response in error situations.

1.4 Project Scope

The scope of the project includes adding new features. It will be a new version of the Google Calendar. Also, it will include features covering post meeting insights

2. Overall Description

2.1 Product Perspective

Even the most unorganized person can become the most productive with Google Calendar. This calendar does more than just remind you what day it is today, from arranging appointments to connecting individuals from all around the world. You can use it to arrange remote meetings with video links, coordinate numerous calendars to locate the optimum time, and integrate it on your website so that your clients may book time with you.

2.2 Product features

- Google Calendar allows users to create and edit events. Events have a set start time and stop time, with an option for an "All-day event".
- Users can enable a "Recurring" functionality with optional parameters for frequency.
- Users can add a color to an event for recognition or to distinguish the event from others.
- Events are viewable in different types of setups, including day, week, month, or schedule.
- Users can optionally set notifications, with options for type (email, mobile push notification) and time.
- Users can invite other people to events; for other Google Calendar users, the event becomes visible in their calendar, and for non-Google Calendar users, an email will have options for "Yes", "No", or "Maybe".
- Privacy settings allow the user to define the levels of public visibility of the entire calendar or individual events. Although the calendar defaults to showing users event times in their local time, users can specify a different time zone for an event.
- Users can enable or disable the visibility of special calendars, including a Birthdays calendar, that automatically retrieves dates of births from a user's Google contacts and displays the dates on a yearly basis, and a Holidays calendar, a country-specific calendar featuring dates of special occasion

2.3 User classes and Characteristics

- **End-user class:** This class comprises naïve users, who will have no knowledge of the internal architecture, they will follow a menu-driven program.
- **Sophisticated-user class:** This class comprises users having some knowledge of the software. They have some knowledge of how to use software and update software and data.
- **Experts-user class:** This class comprises users who will have full knowledge of software as well as their interfaces.

2.4 Operating Environment

2.4.1 Hardware

For best viewing, set the computer's screen resolution to 800x600 pixels or higher. The user's cell phone browser must have cookies enabled and be XHML compliant. Also, the user's cell phone network needs to allow secure sockets layer (SSL) traffic and cookies.

2.4.2 Operating System

Google Calendar is fully compatible with most Apple and PC computer browsers: Microsoft Internet Explorer versions 6–7, Mozilla Firefox version 2 and later, and Apple Safari 3.1. Users must enable cookies and JavaScript. Google Calendar can sync with the 2003 and 2007 Microsoft Outlook Calendar versions, as long as one has Windows Vista or Windows XP, but not the Windows XP 64-bit edition. Users can import events from their Apple iCal (versions 2.0 and newer).

2.5 Design and Implementation constraints

Usage via IMAP clients is subject to other restrictions as well. For example, messages can be sent to only up to 100 recipients at once (including To, CC, and BCC fields). Apart from this, there are bandwidth restrictions as well, which set limits on the data you can download from the servers each day – 2500MB (IMAP) and 1250MB (POP).
Google also sets out bandwidth restrictions on your usage. If you're using the web client, you can download 750MB each hour, and a maximum of 1250MB each day. At the same time, uploads are capped at 300 MB per hour and 500MB per day. Apart from this, you should also be aware of the 25MB attachment limit for every Gmail message. It is recommended that you use Google Drive or other services such as Dropbox to transfer large amounts of data or attachments.

2.6 User Documentation

It will include user manuals to guide users on how to use Google Calendar. Manuals will provide information to users about various functionalities of players like Movie play, audio, etc.

3. System Features

- Creating Separate Calendars
- Sharing & customizing Google Calendar
- Add your non-negotiables
- Block Time for Calls or Emails
- Google Calendar Integration

4. External User Interfaces

4.1 User Interfaces

For the user interface, it provides a full-color display screen. It supports a double-screen display in-phase. It can support Touch and wheel mechanisms for the interface for mobiles. It also supports shortcut key startups.

4.2 Hardware Interfaces

This product requires any digital screen on which the internet is accessible..

4.3 Software Interfaces

It supports Microsoft Windows 98/2000/ME/XP OS, Mac OS 10,linux. It also provides an Image interface, management taskbar. It also provides an interface to share the calendar invite and import it to local or apple calendar or any other calendar.

4.4 Communication Interfaces

No communication is needed.

5. Other Non-Functional Requirements

5.1 Performance Requirements

All calendar scheduling, viewing, and options commands must execute instantaneously, except for the following:

- the time to execute a meeting scheduling operation must be on the order of seconds
- the time to execute viewing another user's calendar must be instantaneous, except for any delays due to network transmission of data from the central host to the local computer

The time to execute all administrative commands must be instantaneous, except for any delays due to network transmission of data from the central host.

In general, the performance of Calendar commands that must access data across a network is subject to potential network delays. The Calendar Tool must transmit data as fast as is possible on any given network, but delays due to external network circumstances are beyond the control of the Calendar.

There are no specific requirements for the allowable size of a calendar in terms of the number of scheduled items. A Calendar on the order of thousands of items is the maximum expected size during normal use.

5.3 Safety Requirements

If you use Google Calendar with your personal Gmail Account, the content you save on Google Calendar is private to you, from others, unless you choose to share it. If you use Google Calendar for work or school, your administrator may be able to view your calendar.

If you share your calendar, your events have the same privacy settings as your calendar. You can change what others can see about certain events.

5.3 Security Requirements

Data is encrypted in-transit and at-rest. To help you access your calendar offline, we also store your calendar entries on your device.

Your Google Account comes with built-in security designed to detect and block threats like spam, phishing and malware. Your activity is stored using strong industry standards and practices.

5.4 Software Quality Attributes

- Reliability: Good validations of inputs from user will be done on both client and server side to avoid incorrect storage of records.
- Maintainability: During the maintenance stage, SRS document can be referred for any validations.
- Portability: This system can be easily viewed in any browser.
- Flexibility: The system keeps on updating the data according to the actions that takes place.
- Timeless: The system carries out all the operations with consumption of very less time.
- Security: Security of the system is maintained by giving access to only authenticated user.

Practical – 3

Aim: Estimation of Project Metrics for your project definition**.**

Project Metrics:

Project size: What would be the size of the code written say, in number of lines, files, modules?

Cost: How much would it cost to develop a software? A software may be just pieces of code, but one has to pay to the managers, developers, and other project personnel.

Duration: How long would it be before the software is delivered to the clients?

Effort: How much effort from the team members would be required to create the software?

In this experiment we will focus on two methods for estimating project metrics i.e using Basic & Intermediate Constructive Cost Model (COCOMO):

Constants for Basic COCOMO:

Type	A	B	C	D
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Constants for Intermediate COCOMO:

Type	A	B
Organic	3.2	1.05
Semi-detached	3.0	1.12
Embedded	2.8	1.20

Equations:

Effort for Basic COCOMO, E = a*(Size)^b (person-month)

Effort for Intermediate COCOMO, E = a*(Size)^b*EAF (person-month)

Duration/Time for development, D = c*(E)^d (month)

Person for Development, N = E/D (person)

Note: For Intermediate COCOMO we will not calculate the duration as it will be similar to Basic COCOMO

Assumptions/Estimated Values:

No. of modules = 4

Project Size = 12 KLOC

EAF = 1.7

1. Organic Development Mode

A. Basic COCOMO Model:

Effort,

E = a*(Size)^b person-month

E = 2.4*(12)^1.05 person-month

E = 32.61 person-month

Time required,

D = c*(E)^d months

D = 2.5*(32.61^0.38) months

D = 9.39 months

Person required,

N = E/D person

N = 32.61/9.39 person

N = 3.47 ~ 4 person

B. Intermediate COCOMO Model:

Effort,

E = a*(Size)^b*EAF person-month

E = 3.2*(12)^1.05*1.17 person-month

E = 73.91 person-month

Time required,

D = 9.39 months (from: same as basic cocomo)

Person required,

N = E/D person

N = 73.91/9.39 person

N = 7.87 ~ 8 person

2. Semi Detached Development Mode

A. Basic COCOMO Model:

Effort,

E = a*(Size)^b person-month

E = 3.0*(12)^1.12 person-month

E = 48.50 person-month

Time required,

D = c*(E)^d months

D = 2.5*(48.50^0.33) months

D = 8.97 months

Person required,

N = E/D person

N = 48.5/8.97 person

N = 5.40 ~ 5 person

B. Intermediate COCOMO Model:

Effort,

E = a*(Size)^b*EAF person-month

E = 3.0*(12)^1.12*1.7 person-month

E = 82.46 person-month

Time required,

D = 8.97months (from: same as basic cocomo)

Person required,

N = E/D person

N = 82.46/8.97 person

N = 9.19 ~ 9 person

3. Embedded Development Mode

A. Basic COCOMO Model:

Effort,

E = a*(Size)^b person-month

E = 3.6*(12)^1.20 person-month

E = 71.01 person-month

Time required,

D = c*(E)^d months

D = 2.5*(71.01^0.32) months

D = 9.78 months

Person required,

N = E/D person

N = 71.01/9.78 person

N = 7.26 ~ 7 person

B. Intermediate COCOMO Model:

Effort,

E = a*(Size)^b*EAF person-month

E = 2.8*(12)^1.20*1.7 person-month

E = 93.89 person-month

Time required,

D = 9.78months (from: same as basic cocomo)

Person required,

N = E/D person

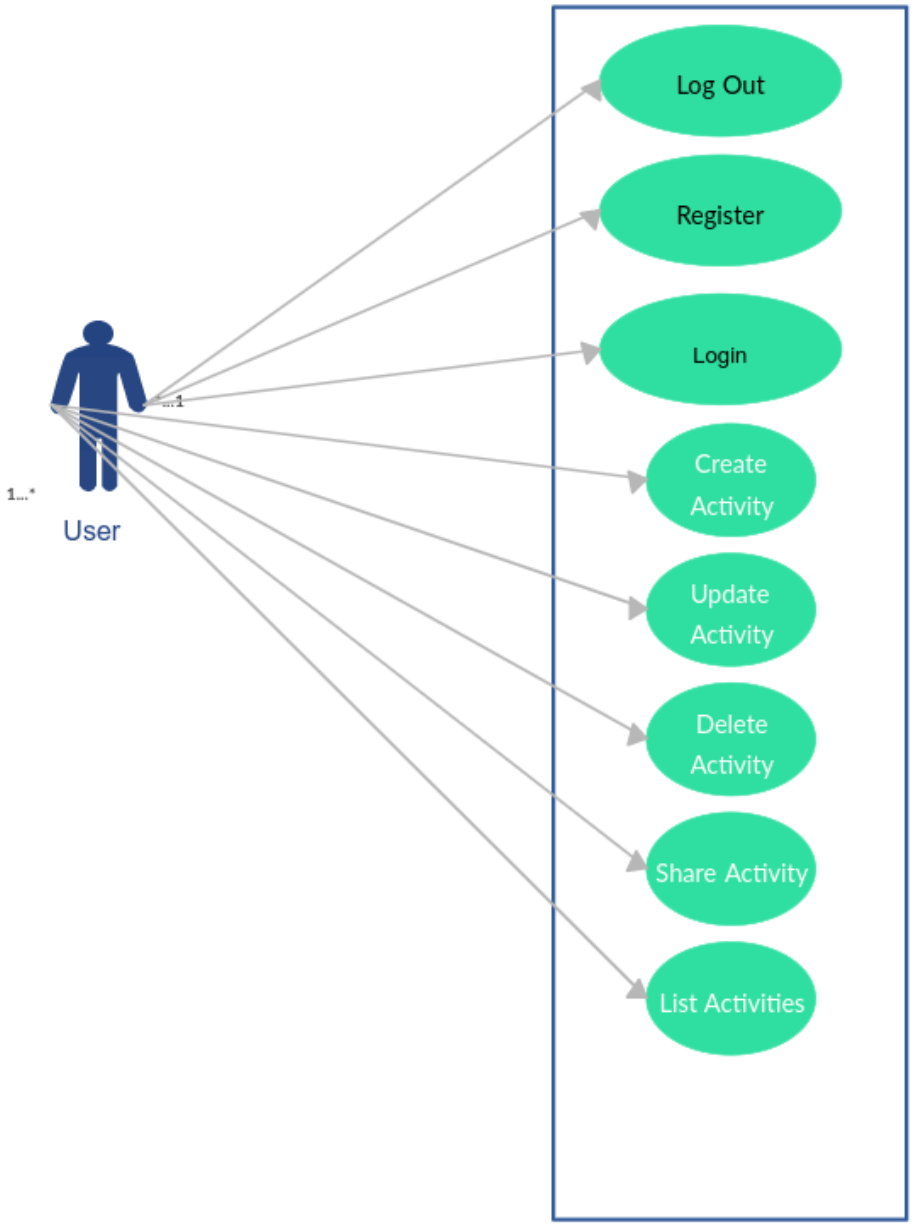
N = 93.89/9.78person

N = 9.60 ~ 10 person

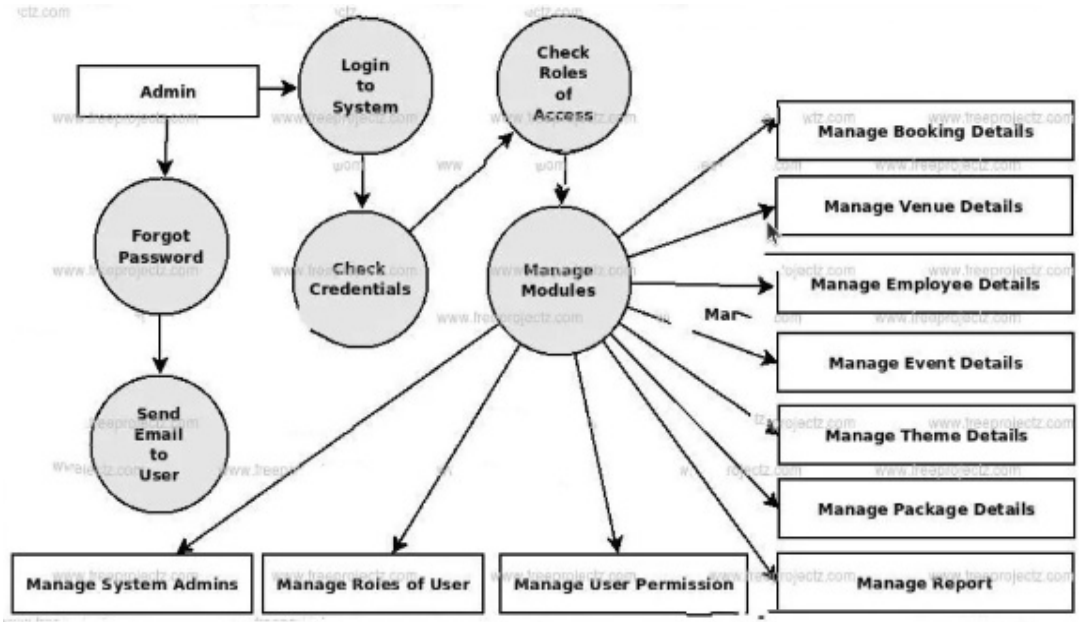
Practical – 4

Aim: To Prepare Use Case Diagrams and Data Flow Diagrams for your Project definition.

Use Case Diagrams:



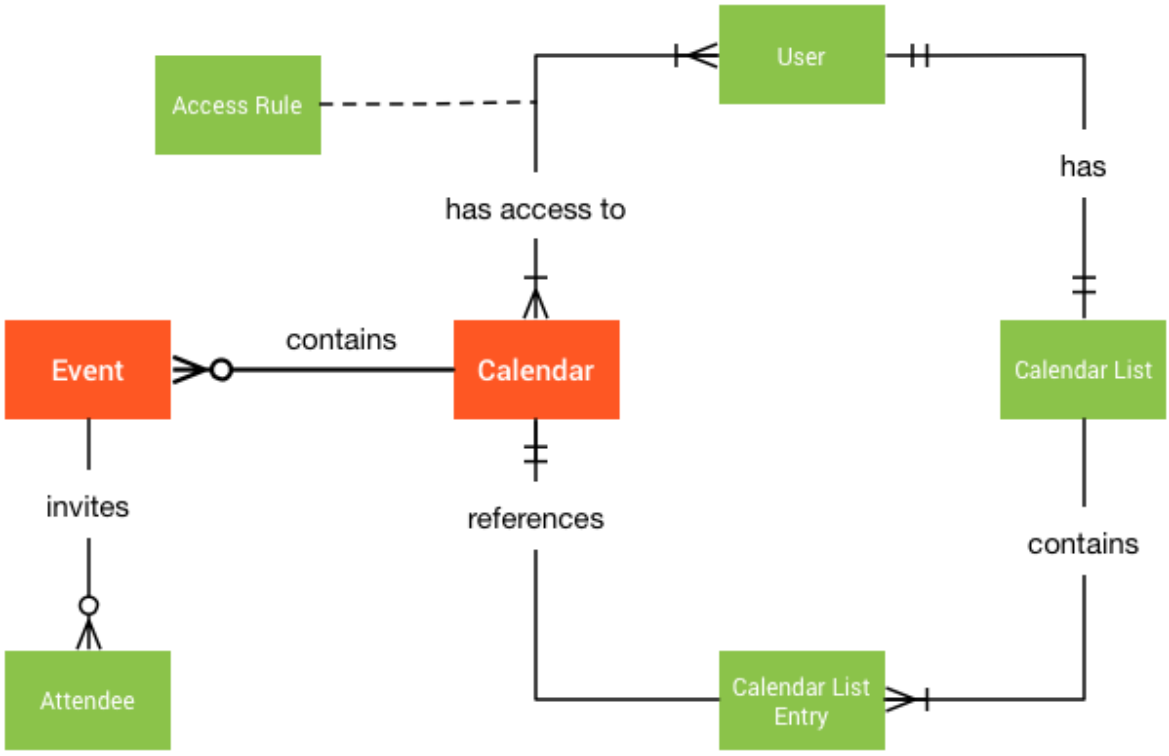
Data Flow Diagram:



Practical – 5

Aim: To Prepare an E-R diagram and data dictionary for your project definition

E-R Diagram



Data Dictionary

SIGNUP TABLE

Attribute Name	Key	Type	size	required	default values
id	Primary	objectID	12	yes	automatic
email address	UNIQUE	string	100	yes	no
password		string	100	yes	none
cpassword		string	100	yes	none

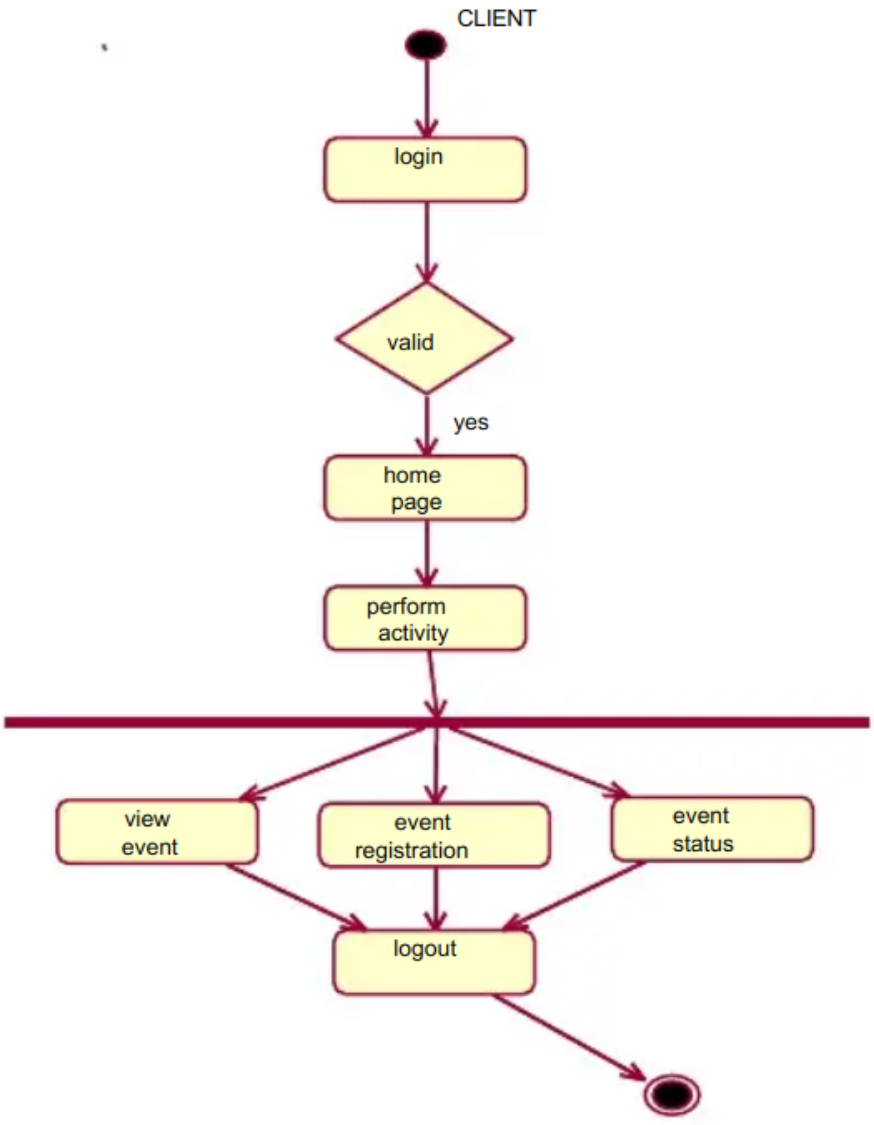
EVENT TABLE

Attribute Name	Key	Type	size	required	default values
MeetId	Primary	string	12	yes	automatic
date		string	100	yes	no
startTime		string	100	yes	none
endTime		string	100	yes	none
description		string	500	no	none
guestList		arrayList<Event>		no	empty
category		string	2	no	private

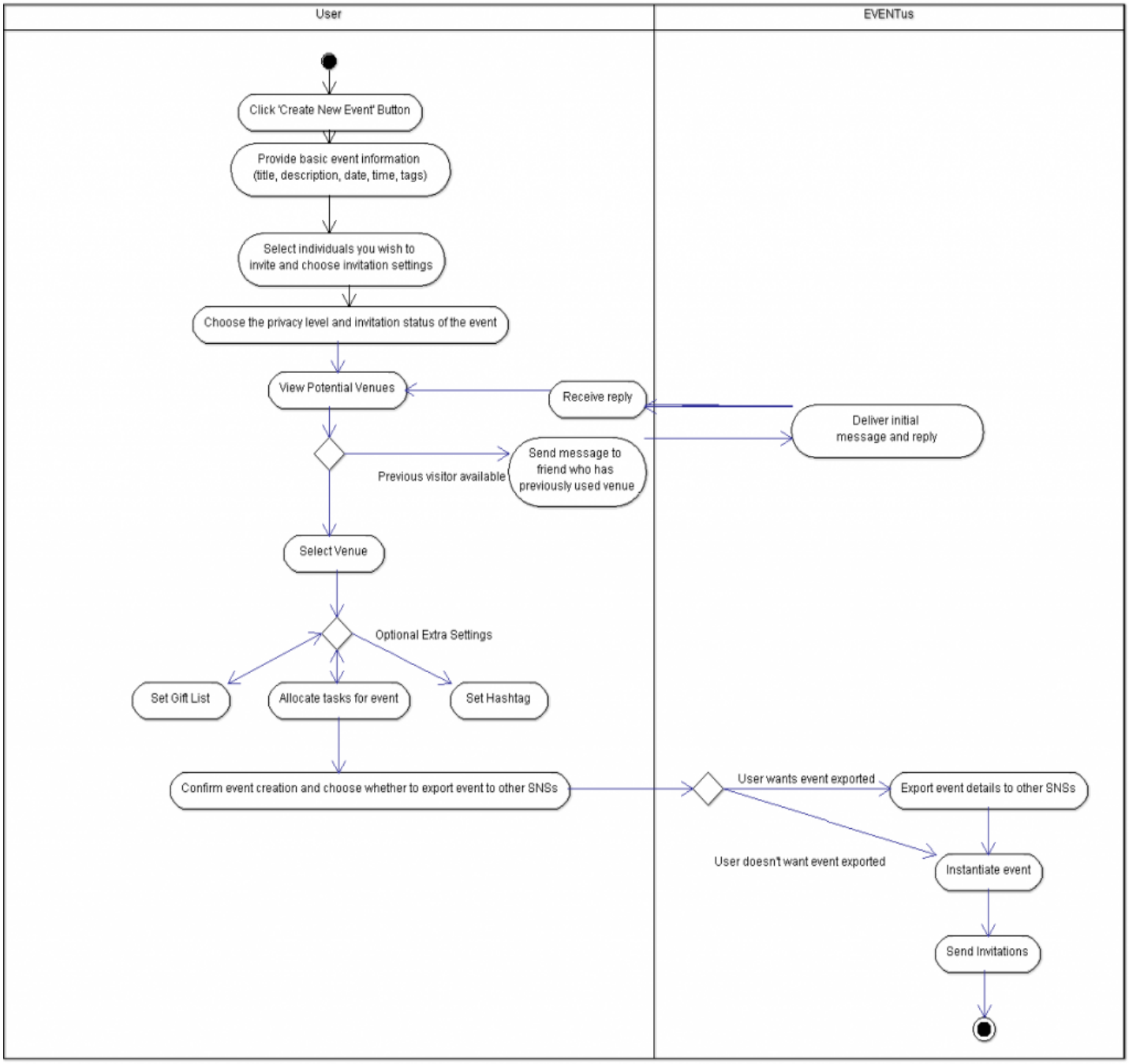
Practical: 6

Aim: To Prepare the Activity Diagram and swim lane diagram for the Project.

Activity diagram:



Swim Lane Diagram:

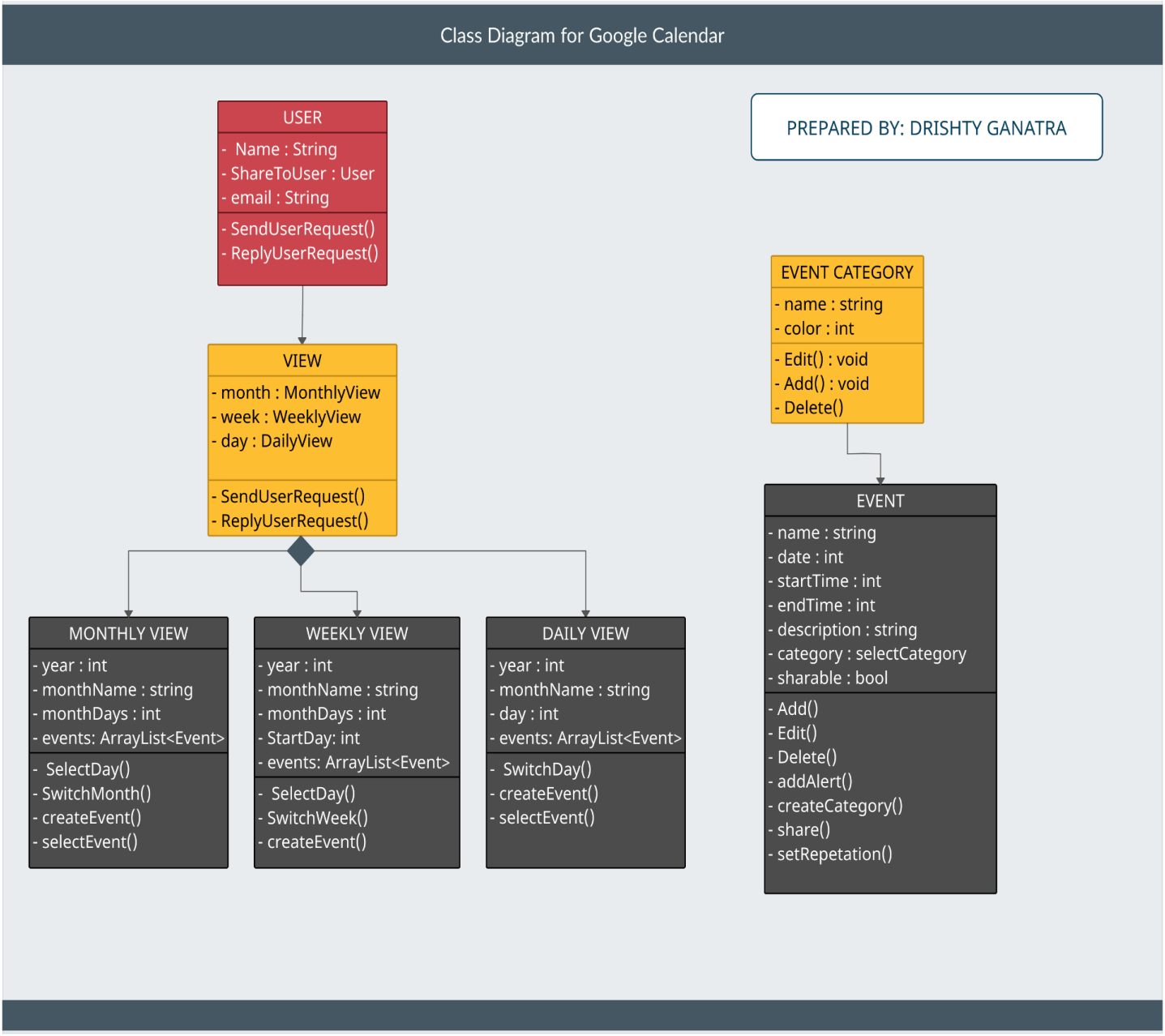


Practical 7

Aim: Identify Domain Classes from the Problem Statements and prepare Class Diagram

Domain classes: user, view, monthly view, weekly view, daily view, event category, event

Class Diagram:



Practical – 8

Aim: Designing Test cases for your project

Test case	Test case description	Test Data	Expected results	Actual results	pass/ fail
1.	Check user login with valid data	email: drishty.ganatra@gmail.com password: abraKaDabra	User should be logged in	user logs in	pass
2.	Check user login with invalid data	email: drishty.ganatra@gmail.com password: 123	User should not be able to log in	user unable to log in	pass
3.	Check user signup with valid data	email: drishty.ganatra@gmail.com password: abraKaDabra	User should be able to sign up	user signs up	pass
4.	Check user signup with invalid data	email: drishty.ganatra@gmail.co password: abraKaDabra	show pop up of invalid mail	user does not sign up	pass
5.	one user trying to view calendar events of other user	events of drishty.ganatra@gmail.com	only the public events should be visible and not the private ones	only the public event is shown	pass
6.	non-existing email address added in guest list	drishty1235@gmail.com	undelivered report should be sent to organizer	organizer receives undelivered mail report	pass
7.	a user/ administrator trying to update calendar for an event of user having a higher priority event already in that duration	existing event details: startTime : 04:00:00 endTime : 05:00:00 event to be added by another user/ administrator's details: startTime : 4:30:00 endTime : 4:45:00	send a mail to user to update the calendar according to their choice and the calendar should not override high priority event	mail sent to the user with calendar invite and user asked to clear the clutter to update the calendar with current events.	pass

Aim: To prepare test cases using testing tools.

Testing tools:

In the market, various types of software testing tools are available for different purposes. Some of them are open source and others are paid. Tools from a software testing context can be defined as a product that supports one or more test activities right from planning, requirements, creating a build, test execution, defect logging and test analysis.

Some Best Testing Tools for Software Testing

- Tosca Testsuite
- Watir
- Sahi
- Ranorex
- TestComplete
- HPE Unified Functional Testing (HP – UFT formerly QTP)
- TestingWhiz
- Selenium

Here, I have used Selenium as a testing tool for my project.

Selenium:

Selenium is one of the most widely used open-source Web UI (User Interface) automation testing suite. It was originally developed by Jason Huggins in 2004 as an internal tool at Thought Works. Selenium supports automation across different browsers, platforms and programming languages.

Selenium can be easily deployed on platforms such as Windows, Linux, Solaris and Macintosh. Moreover, it supports OS (Operating System) for mobile applications like iOS, windows mobile and android.

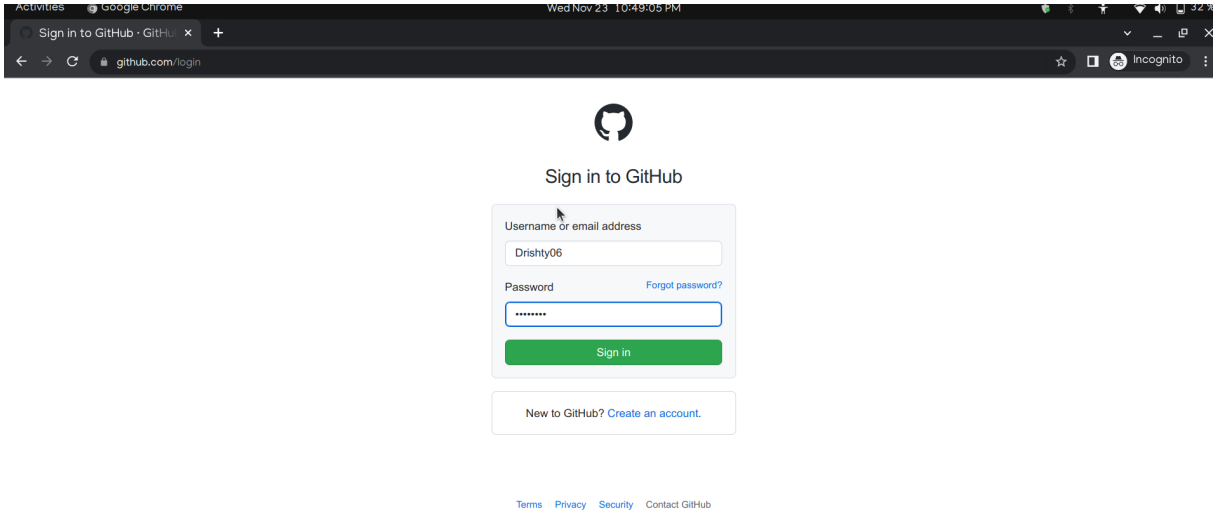
Selenium supports a variety of programming languages through the use of drivers specific to each language. Languages supported by Selenium include C#, Java, Perl, PHP, Python and Ruby. Currently, Selenium Web driver is most popular with Java and C#. Selenium test scripts can be coded in any of the supported programming languages and can be run directly in most modern web browsers. Browsers supported by Selenium include Internet Explorer, Mozilla Firefox, Google Chrome and Safari.

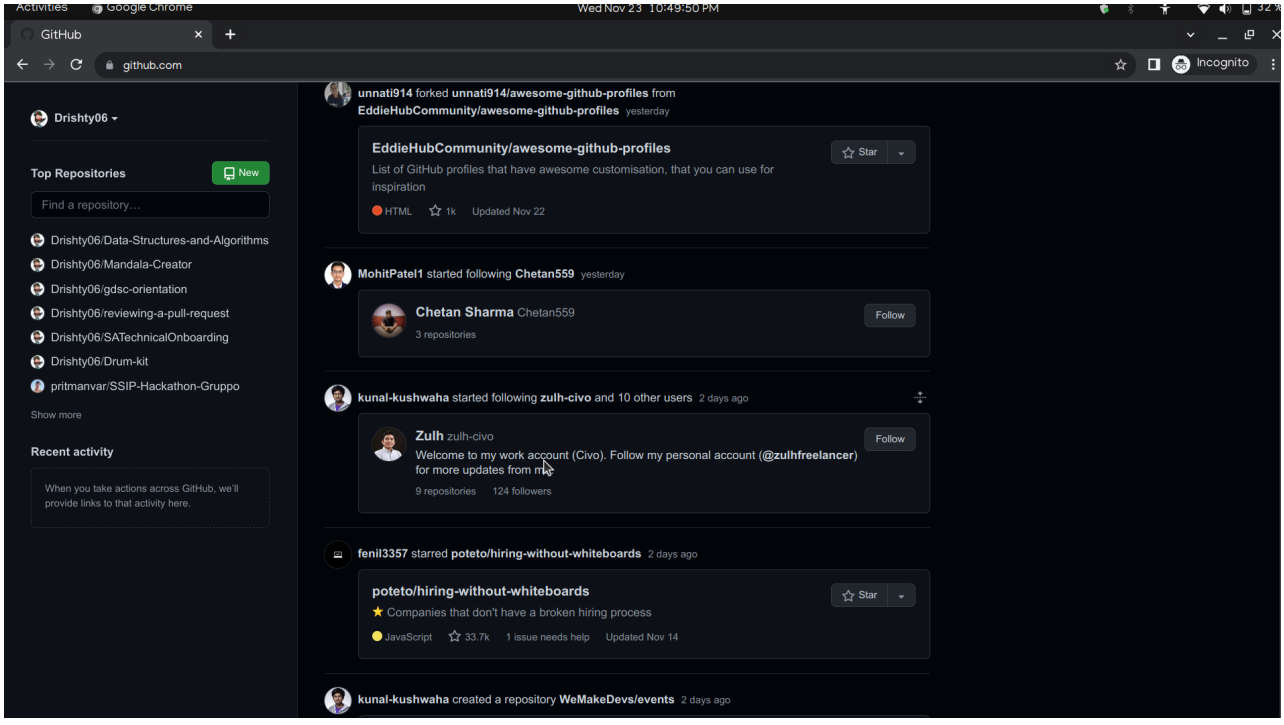
Performing testing using selenium on a demo project:

Code:

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.Test;
public class LoginAutomation {
@Test
public void login() {
System.setProperty("webdriver.chrome.driver", "path of driver");
WebDriver driver=new ChromeDriver();
driver.manage().window().maximize();
driver.get("https://www.github.com/users/sign_in");
WebElement username=driver.findElement(By.id("user_email_Login"));
WebElement password=driver.findElement(By.id("user_password"));
WebElement login=driver.findElement(By.name("commit"));
username.sendKeys("abc@gmail.com");
password.sendKeys("your_password");
login.click();
String actualUrl="https://live.github.com/dashboard";
String expectedUrl= driver.getCurrentUrl();
Assert.assertEquals(expectedUrl,actualUrl);
}
}
```

Result:

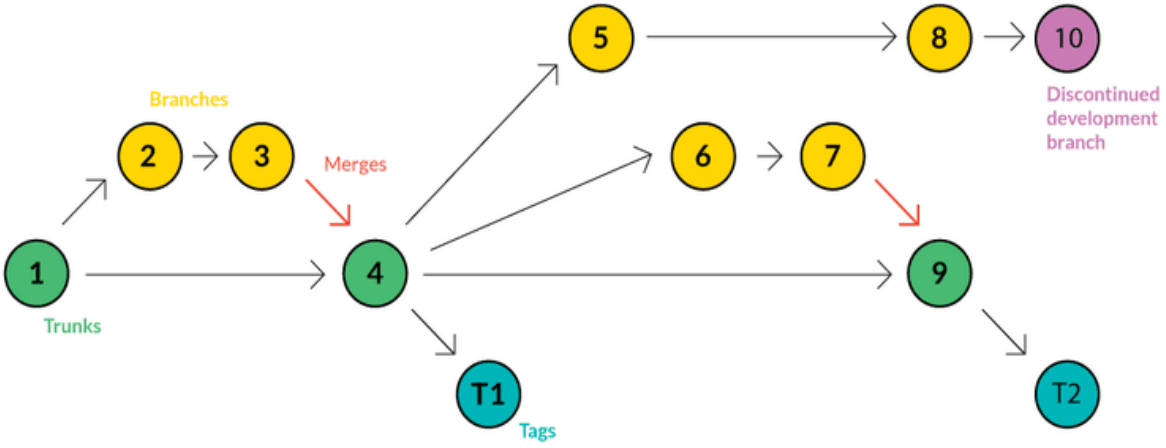

Activities
Google Chrome
Wed Nov 23 10:49:05 PM
Sign in to GitHub · GitHub
github.com/login
Incognito
Sign in to GitHub
Username or email address
Drishty06
Password
Forgot password?
Sign in
New to GitHub? Create an account.
Terms
Privacy
Security
Contact GitHub


Activities
Google Chrome
Wed Nov 23 10:49:50 PM
GitHub
github.com
Incognito
Drishty06
Top Repositories
New
Find a repository...
Drishty06/Data-Structures-and-Algorithms
Drishty06/Mandala-Creator
Drishty06/gdsc-orientation
Drishty06/reviewing-a-pull-request
Drishty06/SATechnicalOnboarding
Drishty06/Drum-kit
pritmanvar/SSIP-Hackathon-Gruppo
Show more
Recent activity
When you take actions across GitHub, we'll provide links to that activity here.
unnati914 forked unnati914/awesome-github-profiles from EddieHubCommunity/awesome-github-profiles yesterday
EddieHubCommunity/awesome-github-profiles
List of GitHub profiles that have awesome customisation, that you can use for inspiration
HTML
1k
Updated Nov 22
Star
MohitPatel1 started following Chetan559 yesterday
Chetan Sharma Chetan559
3 repositories
Follow
kunal-kushwaha started following zulh-civo and 10 other users 2 days ago
Zulh zulh-civo
Welcome to my work account (Civo). Follow my personal account (@zulhfreelancer) for more updates from me
9 repositories
124 followers
Follow
fenil3357 starred poteto/hiring-without-whiteboards 2 days ago
poteto/hiring-without-whiteboards
Companies that don't have a broken hiring process
JavaScript
33.7k
1 issue needs help
Updated Nov 14
Star
kunal-kushwaha created a repository WeMakeDevs/events 2 days ago

Practical: 9

Aim: To install version control tool and study its features.

What is Version Control?



- Version control, also known as source control, is the practice of tracking and managing changes to software code. Version control systems are software tools that help software teams manage changes to source code over time. As development environments have accelerated, version control systems help software teams work faster and smarter. They are especially useful for DevOps teams since they help them to reduce development time and increase successful deployments.
- Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

What are the benefits of using version control?

- Using version control software is a best practice for high performing software and DevOps teams. Version control also helps developers move faster and allows software teams to preserve efficiency and agility as the team scales to include more developers.
- Version Control Systems (VCS) have seen great improvements over the past few decades and some are better than others. VCS are sometimes known as SCM (Source Code Management) tools or RCS (Revision Control System). One of the most popular VCS tools in use today is called Git. Git is a Distributed VCS, a category known as DVCS, more on that later. Like many of the most popular VCS systems available today, Git is free and open source.

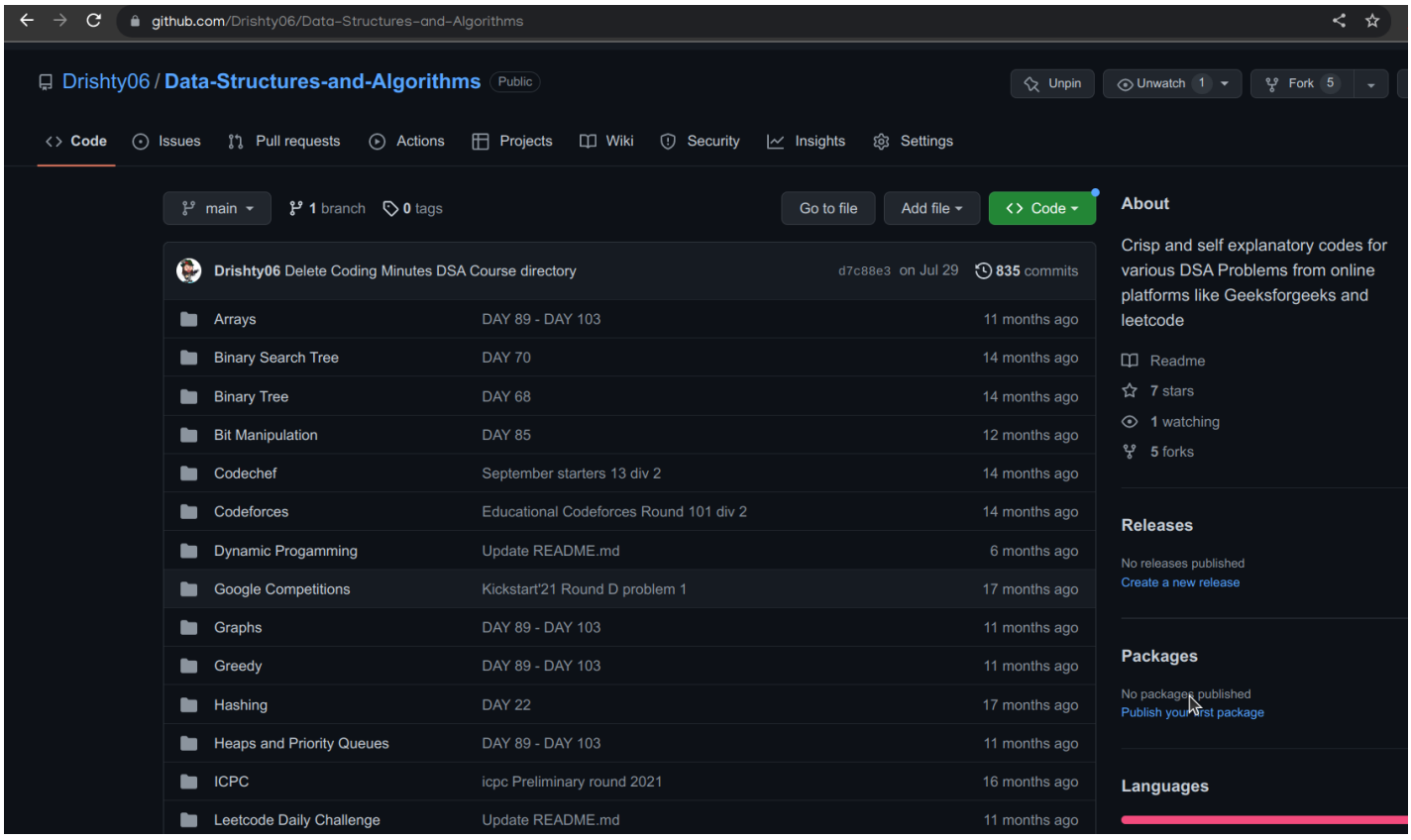
Git and GitHub as version Control Tool

What is Git?

- **Git** is a free, open-source distributed version control system. It keeps track of projects and files as they change over time with the help of different contributors.
- **Version control** is a system that records changes to a file, or set of files, over time so that you can recall specific versions later.
- Git helps keep track of changes made to a code. If at any point during coding you hit a fatal error and don't know what's causing it, Git allows you to revert back to a stable state. It also helps you see what changes have been made to the code over time.

What is GitHub?

- GitHub is a code hosting platform for collaboration and version control.
- GitHub lets you (and others) work together on projects.



Getting started with git

To get started with Git, go to your terminal and run the following command in your project directory. This initializes a project directory.

```
git init
```

Run the following command to add files for Git to track. This will add these files to the staging area.

```
git add <filename_one>
```

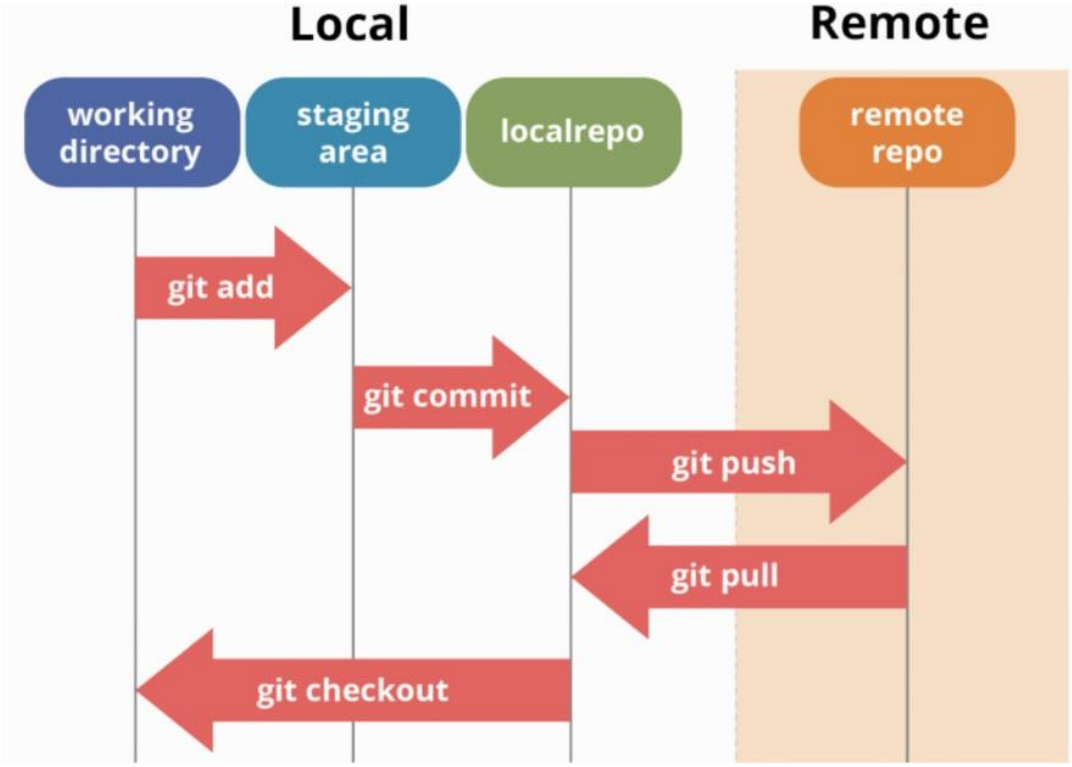
Run the following command to commit your changes to these files.

```
git commit -m "<add a commit message here>"
```

We can push our changes through once we're done.

```
git push
```

Making any more changes in the master branch will require these changes to be committed again.



Practical: 10

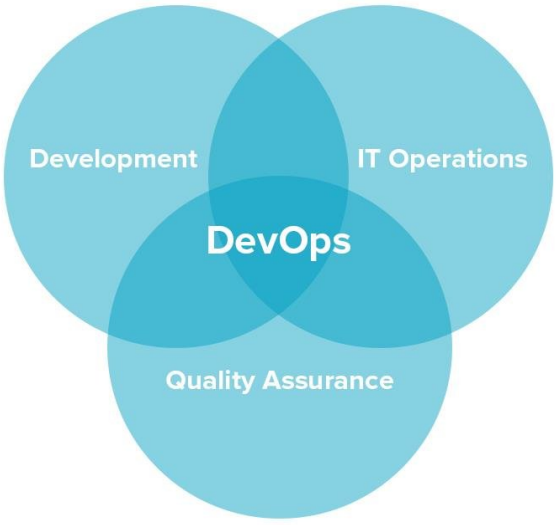
Aim:

i) Study Principles, Practices, tools of DevOps, and find out various advantages and disadvantages of DevOps.

ii) Study of any two Open source tools in DevOps

- DevOps stands for development and operations. It's a practice that aims at merging development, quality assurance, and operations (deployment and integration) into a single, continuous set of processes. This methodology is a natural extension of Agile and continuous delivery approaches.

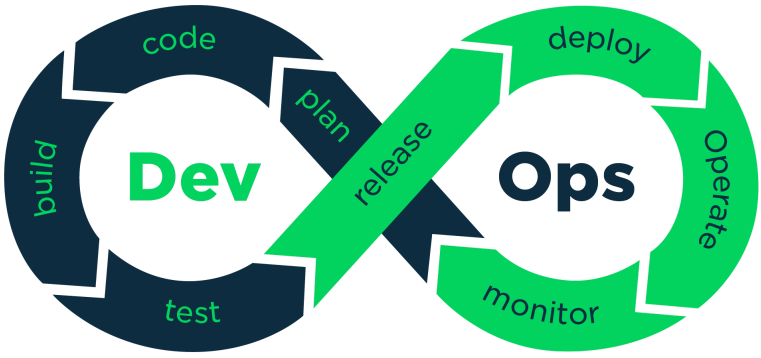
Principles Of Devops:



- **Constant collaboration and communication.** These have been the building blocks of DevOps since its dawn. Your team should work cohesively with the understanding of the needs and expectations of all members.
- **Gradual changes.** The implementation of gradual rollouts allows delivery teams to release a product to users while having an opportunity to make updates and roll back if something goes wrong.

- **Shared end-to-end responsibility.** When every member of a team moves towards one goal and is equally responsible for a project from beginning to end, they work cohesively and look for ways of facilitating other members' tasks
- **Early problem-solving.** DevOps requires that tasks be performed as early in the project lifecycle as possible. So, in case of any issues, they will be addressed more quickly.

Devops Practices:



- **Agile planning**

  In contrast to traditional approaches of project management, Agile planning organizes work in short iterations (e.g. sprints) to increase the number of releases. This means that the team has only high-level objectives outlined, while making detailed planning for two iterations in advance. This allows for flexibility and pivots once the ideas are tested on an early product increment.

- **Continuous development**

  The concept of continuous "everything" embraces continuous or iterative software development, meaning that all the development work is divided into small portions for better and faster production. Engineers commit code in small chunks multiple times a day for it to be easily tested. Code builds and unit tests are automated as well.

- **Continuous automated testing**

  A quality assurance team sets committed code testing using automation tools like Selenium, Ranorex, UFT, etc. If bugs and vulnerabilities are revealed, they are sent back to the engineering team. This stage also entails version control to detect integration problems in advance. A Version Control System (VCS) allows developers to record changes in the files and share them with other members of the team, regardless of their location.

- **Continuous integration and continuous delivery (CI/CD)**

  The code that passes automated tests is integrated in a single, shared repository on a server. Frequent code submissions prevent a so-called “integration hell” when the differences between individual code branches and the mainline code become so drastic over time that integration takes more than actual coding. Continuous delivery is an approach that merges development, testing, and deployment operations into a streamlined process as it heavily relies on automation.

- **Continuous monitoring**

  The final stage of the DevOps lifecycle is oriented to the assessment of the whole cycle. The goal of monitoring is detecting the problematic areas of a process and analyzing the feedback from the team and users to report existing inaccuracies and improve the product’s functioning.

- **Containerization**

  Containers are more lightweight and packaged with all runtime components (files, libraries, etc.) but they don’t include whole operating systems, only the minimum required resources. Containers are used within DevOps to instantly deploy applications across various environments and are well combined with the IaC approach described above. A container can be tested as a unit before deployment. Currently, Docker provides the most popular container toolset.

Advantages Of DevOps:

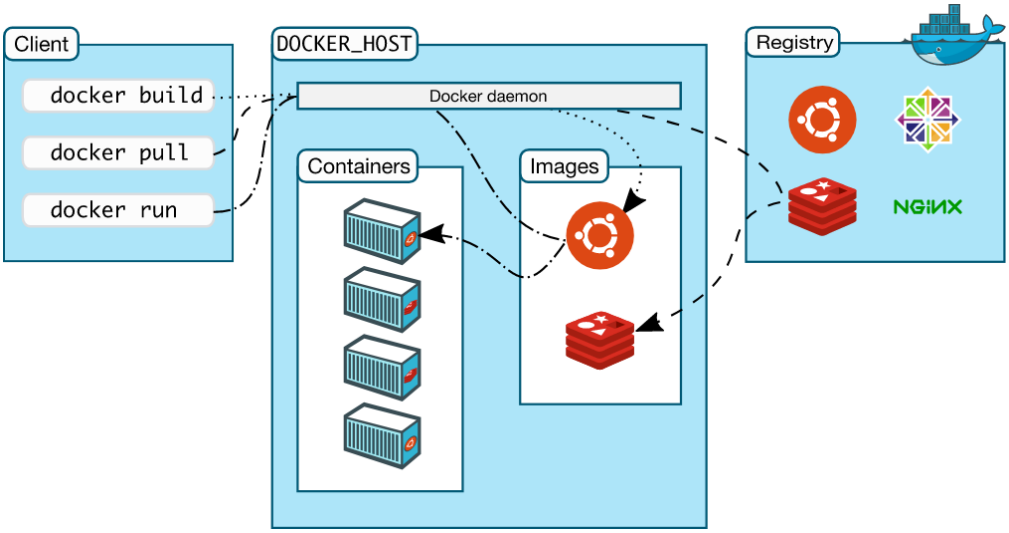
- Faster development and deployment of applications.
- Faster response to the market changes to improve business growth.
- Business profit is increased as there is a decrease in software delivery time and transportation costs.
- Improves customer experience and satisfaction.
- Simplifies collaboration as all the tools are placed in the cloud for customers to access.
- Leads to better team engagement and productivity due to collective responsibility.

Disadvantages Of DevOps:

- Less availability of DevOps professionals.
- Infrastructure cost is high for setting by DevOps environment.
- Lack of DevOps knowledge can lead to problems in the continuous integration of automation projects.

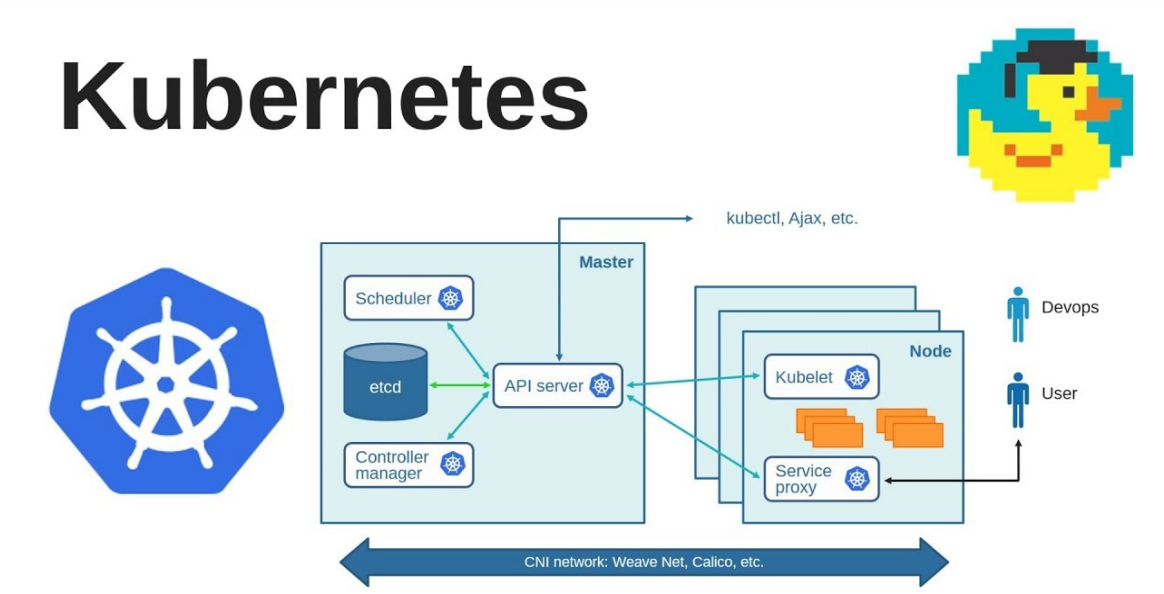
Open Source Tools for Devops

Docker:



- As a desktop user you can use one of Docker's tools (like Docker Desktop) to run applications that have been packaged into container form. There are thousands of container images available in Docker Hub.
- As a developer, you can use Docker to build a container image for your own application, and share the image with others.
- As a server administrator, you can use Docker to run applications in containers, whether that's apps from your developers, or third-party apps like MySQL.

Kubernetes:



- Kubernetes is a tool that allows one to manage multiple Docker containers as a single unit to make development occur more quickly and simplify operations overall.
- Kubernetes lets you build your entire infrastructure as code (a pattern known as IaC). Kubernetes can define and automatically provision all aspects of your applications and tools, including access control, networking, databases, storage, and security.
- You can similarly manage environment configuration in code. Instead of running a script every time you need to deploy a new environment, you prepare a source repository with environment configuration, and Kubernetes and use this declarative configuration to set up environments automatically.