

# **Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY**

(An Autonomous Institute, affiliated to VTU, Belagavi, Accredited by NAAC with 'A' Grade)

**Near Jnana Bharathi Campus, Bengaluru - 560056**



Aided By Govt. of Karnataka

## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

### **Cloud Computing Laboratory Report**

*Submitted in partial fulfilment of the requirement for the award of the Degree of*

*Submitted By*

**KARTHIK B**

**1DA20CS054**

*Submitted to,*

**Dr. Smitha Shekhar B**

Associate Professor,  
CSE Dr AIT

**Dr. Asha K N**

Associate Professor  
CSE Dr AIT

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**2023-2024**

# **Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY**

(An Autonomous Institute, affiliated to VTU, Belagavi, Accredited by NAAC with ‘A’ Grade)

**Near Jnana Bharathi Campus, Bengaluru – 560056**



## **CERTIFICATE**

### **“Cloud Computing Laboratory”**

This is to certify that the submitted document in the partial fulfillment of the requirement of the 7th semester Cloud Computing laboratory curriculum during the year 2022-23 is a result of Bonafede work carried out by-

**KARTHIK B**

**1DA20CS054**

**Signature of the guide,**

---

**DR .Smitha Shekar B**

Associate Professor

CSE DR AIT

**DR. Asha K N**

Associate Professor

CSE DR AIT

1. Internal Examiner \_\_\_\_\_

2. External Examiner \_\_\_\_\_

---

**Dr. Siddaraju**

**Head of Department**

**CSE Dr. AIT**

## **TABLE OF CONTENTS**

<b>Chapter No.</b>	<b>Title</b>	<b>Page No.</b>
<b>PART A</b>		
Chapter 1	Salesforce	1
Chapter 2	CloudAnalyst	63
Chapter 3	Google App Engine	104
Chapter 4	Amazon Web Services – RDS	112
Chapter 5	VMware Workstation	126
	File Transfer in Virtual Machines	147
<b>PART B</b>		
Chapter 6	Mini Project	167
Chapter 7	Industry Certification - Salesforce	

---

# PART A

# **SALESFORCE**

Salesforce is an American cloud-based software company headquartered in San Francisco, California. It provides customer relationship management (CRM) service and also provides enterprise applications focused on customer service, marketing automation, analytics, and application development.

## **History:**

The company was founded on February 3, 1999 by former Oracle executive Marc Benioff, together with Parker Harris, Dave Haelterhoff, and Frank Dominguez as a software as a service (SaaS) company, and was launched publicly between September and November 1999.

In June 2004, the company had its initial public offering on the New York Stock Exchange under the stock symbol CRM and raised US\$110 million. Early investors include Larry Ellison, Magdalena Yesil, Halsey Minor, Stewart Henderson, Mark Iscaro, and Igor Sill, a founding member of Geneva Venture Partners.

In October 2014, Salesforce announced the development of its Customer Success Platform to tie together Salesforce's services, including sales, service, marketing, analytics, community, and mobile apps. In October 2017, Salesforce launched a Facebook Analytics tool for business-to-business marketers. In September 2018, Salesforce partnered with Apple intended on improving apps for businesses.

In February 2020, co-chief executive officer Keith Block stepped down from his position in the company. Marc Benioff remained as chairman and chief executive officer.

On December 1, 2020, it was announced that Salesforce would acquire Slack for \$27.7 billion. The acquisition closed on July 21.

In February 2021, Salesforce announced that CFO Mark Hawkins would be retiring from his position after six years of working for the company; however, retaining a position as CFO emeritus until October. Amy Weaver was selected as his replacement.

Salesforce.com's customer relationship management (CRM) service comprises several broad categories: Commerce Cloud, Sales Cloud, Service Cloud, Data Cloud (including Jigsaw), Marketing Cloud, Community Cloud (including Chatter), Manufacturing Cloud, Analytics Cloud, App Cloud, Vaccine Cloud, IoT and Work.com with over 100,000 customers.

Main services

Salesforce's main services are tools for case, task and issue management. It also gives customers tracking abilities for their raised cases and conversation features for social networking Web sites, provides analytical tools and other services including email alert, Google search, and access to customers' entitlement and contracts. They also partner with companies like IBM, Accenture, and Saggezza to help integrate Salesforce's cloud-based services into their businesses.

### **Lightning Platform**

Lightning Platform (also known as Force.com) is a platform as a service (PaaS) that allows developers to create add-on applications that integrate into the main Salesforce.com application. [failed verification] These third-party applications are hosted on Salesforce.com's infrastructure.

Force.com applications are built using declarative tools, backed by Lightning [further explanation needed] and Apex, a proprietary Java-like programming language for Force.com, as well as Visualforce, a framework including an XMLsyntax typically used to generate HTML. The Force.com platform typically receives three complete releases a year. As the platform is provided as a service to its developers, every single development instance also receives all these updates.

In 2015, a new framework for building user interfaces – Lightning Components – was introduced in beta. Lightning components are built using the open-source Aura Framework but with support for Apex as the server-side language instead of Aura's JavaScript dependency. This has been described as an alternative to, not necessarily a replacement for, Visualforce pages.

As of 2013, the Force.com platform has 1.4 million registered developers. Lightning Base Components is the component library built on top of Lightning Web Components.

### **Experience Cloud**

Experience Cloud (formerly Community Cloud) provides Salesforce customers the ability to create online web properties for external collaboration, customer service, channel sales, and other custom portals in their instance of Salesforce. Tightly integrated to Sales Cloud, Service Cloud, and App Cloud, Experience Cloud can be quickly customized to provide a wide variety of web properties. Experience Cloud combines the functionality of the former Salesforce Customer and Partner Portals with some additional features.

## **Work.com**

Work.com, previously Rypple, is a social performance management platform for managers and employees. It allows continuous coaching, real-time feedback, and recognition. It is aimed at sales management, customer service, marketing, and can be utilized by human resource departments.

Work.com, then known as "Rypple", was founded by Daniel Debow and David Stein, to create a simple way of asking for feedback anonymously at work. The company was formed in May 2008 and their client list included Mozilla, Facebook, LinkedIn and the Gilt Groupe. Rypple aims to get employees to build and manage their own coaching networks.

In September 2011, Rypple announced that they had hired Bohdan Zabawskyj as its Chief Technology Officer. In 2011, Rypple developed a more formalized management methodology called OKR ("Objectives and Key Results") for Spotify. Rypple also partnered with Facebook to create "Loops", short for "feedback loops", which gathers feedback from co-workers, including praise, progress against goals, and coaching from supervisors into one channel.

In December 2011, Salesforce.com announced that they would acquire Rypple. The transaction was completed in 2012 and Rypple was rebranded as Work.com in September 2012.

## **AppExchange**

Launched in 2005, the Salesforce AppExchange is an online application marketplace for third-party applications that run on the Force.com platform. Applications are available for free, as well as via yearly or monthly subscription models. Applications available range from integrations with SharePoint to mobile approval management. As of June 2016, it features 2,948 applications which have driven 3+ million installs. The "AppExchange" is also a place customers can search for cloud consulting partners to help them implement the technology in their own organization. Cloud consulting partners for Salesforce include large companies like IBM's "Bluewolf" and Accenture as well as smaller ones like Cloudrreach.

## **myTrailhead**

Launched in 2019, Salesforce's myTrailhead is an online training platform that can be customized for the specific needs of its customers. The platform extends functionality to provide users with training content specific to their usage of Salesforce and enables them to create and publish their own training content and programs.



## **Technologies:**

Salesforce is powered by the Model–view–controller architecture.

### **Apex**

Apex is a proprietary programming language provided by the Force.com platform to developers similar to Java and C#. It is a strongly typed, object-oriented, case-insensitive programming language, following a dot-notation and curly-brackets syntax. Apex can be used to execute programmed functions during most processes on the Force.com platform including custom buttons and links, event handlers on record insertion, update, or deletion, via scheduling, or via the custom controllers of Visualforce or Lightning Experience pages.

Due to the multitenant nature of the platform, the language has strictly imposed governor limitations to guard against any code monopolizing shared resources. Salesforce provides a series of asynchronous processing methods for Apex to allow developers to produce longer-running and more complex Apex code.

### **Lightning**

In 2014, Salesforce made public the front end of its platform, called Lightning. This component-based framework is what the Salesforce mobile app is built on. Salesforce built on this framework in 2015 by releasing the Lightning Design System, an HTML style framework with default CSS styling built in. This framework allows customers to build their own components to either use in their internal instances or sell on the AppExchange.

The Salesforce Lightning App Builder is a tool for rapid application development of responsive web interfaces. This interface allows for different screens to be put together based on Lightning components. This can be used as layouts for records or specific applications.

Lightning Experience, released in 2016, is the new redesigned interface in Salesforce for processes enhancement. Since then all the apps available on AppExchange need to be Lightning and those built on Classic have to migrate to Lightning as Classic is not to be updated any more by Salesforce. The platform offers an option for developers to employ migration techniques to enable the new user-friendly interface and switch to Lightning.

What is Salesforce used for?

- Engage customers with relevant, empathetic digital marketing from anywhere.
- Sell smarter and grow your business faster from anywhere.
- Quickly launch and scale ecommerce built around your customer — from anywhere.
- Provide great customer service from anywhere.
- Go digital fast and empower your teams to work from anywhere.



(Followed By 10 Playgrounds)

# 1) Create a web application to enter student details like Name, USN, Semester, Section and CGPA to a database on Salesforce Cloud Platform.

Steps:

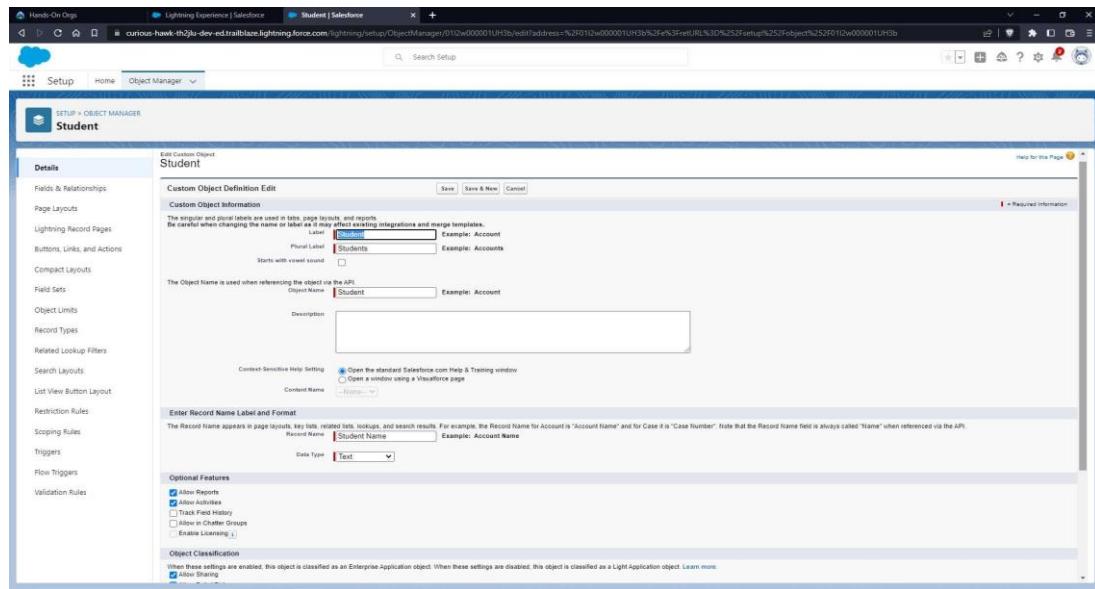
1. Launch Salesforce Trailhead Playground by going to <https://trailhead.salesforce.com>. Now switch to Hands on orgs by clicking your picture in the top right corner and then click on “Hands on orgs”.

The screenshot shows the Trailhead Hands-On Orgs interface. At the top, there are three tabs: "Recently Viewed", "Flight", and "Salesforce". Below the tabs, there's a search bar and a navigation menu with links to "Products", "Industries", "Customers", "Learning", "Support", "Company", and "Salesforce+". The main content area is titled "Hands-On Orgs" and displays "Connected Orgs (3)". It lists three orgs: "lab1 LAST USED", "DreamHouse Playground", and another unnamed org. Each org entry includes fields for Organization ID, Username, Type, Created, and Last Activity, along with "Rename", "Disconnect", and "Launch" buttons. The bottom of the screen shows a taskbar with various icons and system status information.

2. Click on create playground to create a new one. And follow the instructions.
3. After creating the playground click on Launch.
4. Click on the gear icon on top right menu and click “setup.”
5. Now is the time to create our objects, hence we navigate to “Object Manager” hover on “Create” and click on “Custom Object” under it.

The screenshot shows the Salesforce Setup Object Manager page. The title bar says "Hands-On Orgs", "Lightning Experience | Salesforce", and "Object Manager | Salesforce". The main content area is titled "Object Manager" and shows a table of standard objects. The columns are labeled "LABEL", "API NAME", "TYPE", "DESCRIPTION", and "LAST MODIFIED". The table includes rows for Account, Activity, Alternative Payment Method, API Anomaly Event Store, Asset, Asset Action, Asset Action Source, Asset Relationship, Asset State Period, Associated Location, Authorization Form, Authorization Form Consent, Authorization Form Data Use, Authorization Form Text, Business Brand, Campaign, Campaign Member, Card Payment Method, Case, and Case Related Issue. On the right side of the table, there are buttons for "Quick Find", "Schema Builder", "Create", "Custom Object", and "Custom Object from AppExchange". The URL at the bottom of the browser window is <https://curious-hawk-th2qk-dev-trailblaze.lightning.force.com/lightning/setup/ObjectManager/home>.

6. Name the object as “Student”. Allow reports and search
7. Check in front of “Launch New Custom Tab Wizard after saving this custom object”



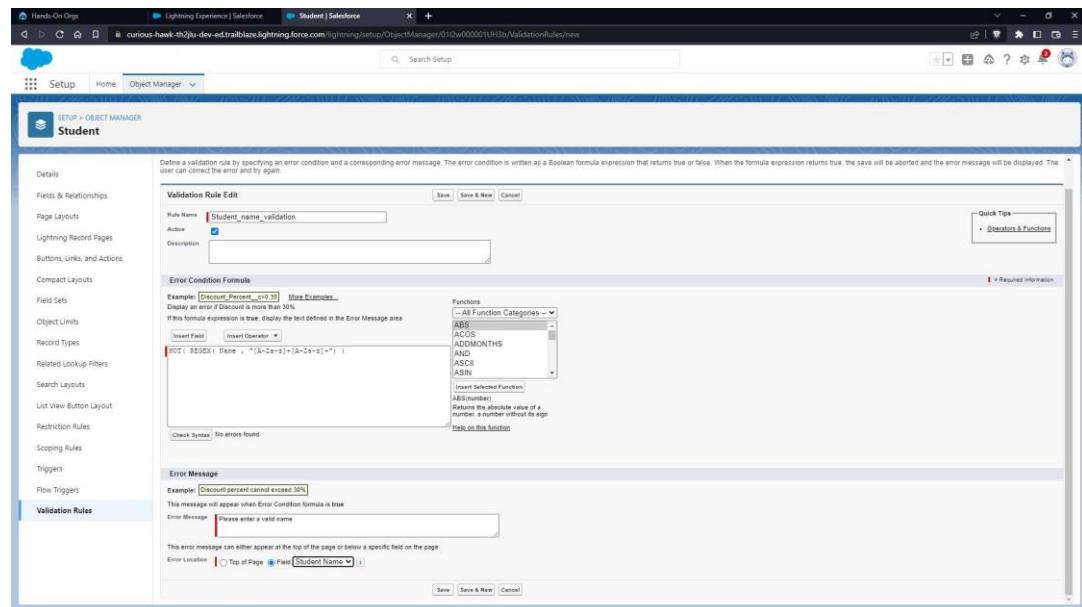
8. Go back to home, search for Tab, navigate to custom object tab and click on new. Create a tab for “student” object.
9. Now let’s add fields to the object: Go to “Fields & Relationships” option of student object and click “New”.
10. Add the following fields one by one:
  - Field Label: **USN**, Length: **10**, Data Type: **Text**, provide an example as Help Text, make it as required field and don’t allow duplicate values and make it as case insensitive.
  - Field Label: **Section**, Data Type: **Picklist**, provide the sections you could think of, make it as required.
  - Field Label: **Semester**, Data Type: **Picklist**, provide 1-8 numbers, make it as required.
  - Field Label: **CGPA**, Length: **2**, Decimal Place: **2**, Data Type: **Number**, make it as required field.
  - Optionally we can also add phone number and email ID.

## 11. Let’s start validating each data

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	CGPA	CGPA__c	Number(2, 2)		
Lightning Record Pages	Created By	CreatedById	Lookup(User)		
Buttons, Links, and Actions	Email	Email__c	Text(80)		
Compact Layouts	Last Modified By	LastModifiedById	Lookup(User)		
Field Sets	Mobile	Mobile__c	Text(10)		
Object Limits	Owner	OwnerId	Lookup(User Group)		
Record Types	Section	Section__c	Picklist		
Related Lockup Filters	Semester	Semester__c	Picklist		
Search Layouts	Student Name	Name	Text(80)		
List View Button Layout	USN	USN__c	Text(10) (Unique Case Insensitive)		
Restriction Rules					
Scoping Rules					
Triggers					
Flow Triggers					
Validation Rules					

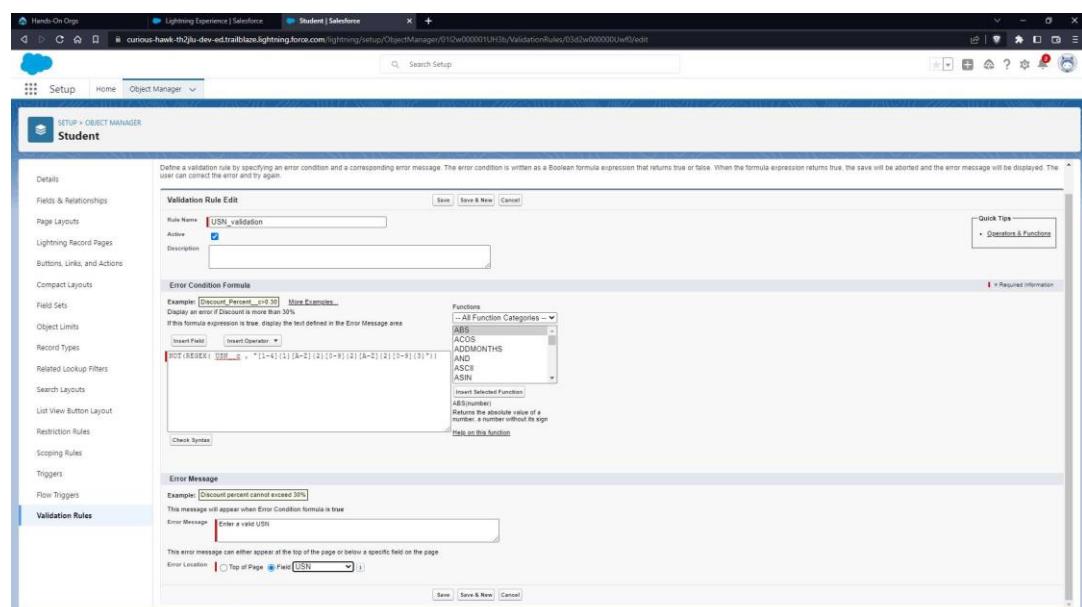
### a. Name validation

- Go to validation rule and click “New”
- Name it as “Student name validation”
- Error condition Formula: “NOT(REGEX(Name, “[a-zA-Z]+[a-zA-Z]+”))”
- Error Message: “Please Enter a valid name”.
- Error Location: Field – Student name
- Click Save



### b. USN validation

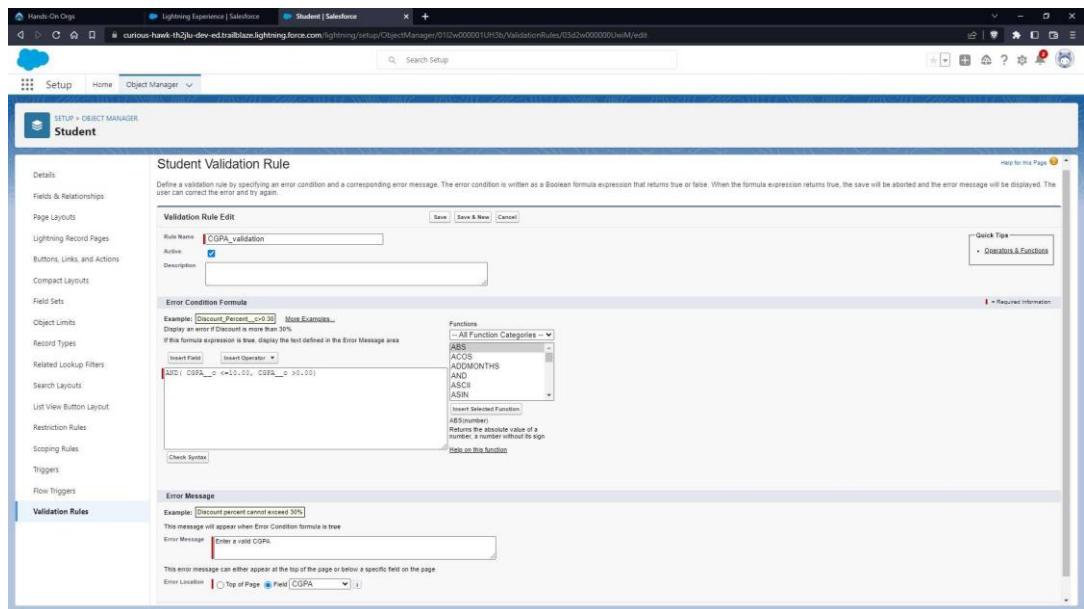
- Go to validation rule and click “New”
- Name it as “USN validation”
- Error condition Formula: “NOT(REGEX( USN\_c , "[1-4]{1}[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{3}"))”
- Error Message: “Please Enter a valid USN”.
- Error Location: Field – USN
- Click Save



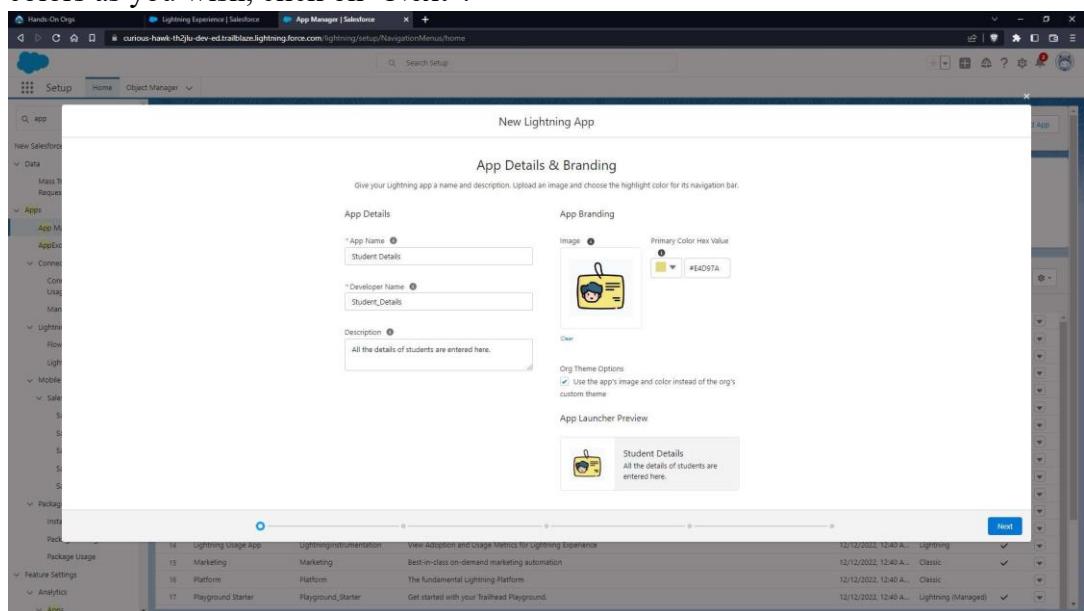
### c. CGPA validation

- Go to validation rule and click “New”

- Name it as “CGPA validation”
- Error condition Formula: “AND(CGPA\_\_c<=10.00, CGPA\_\_c>0.00)”
- Error Message: “Please Enter a valid CGPA”.
- Error Location: Field – CGPA
- Click Save

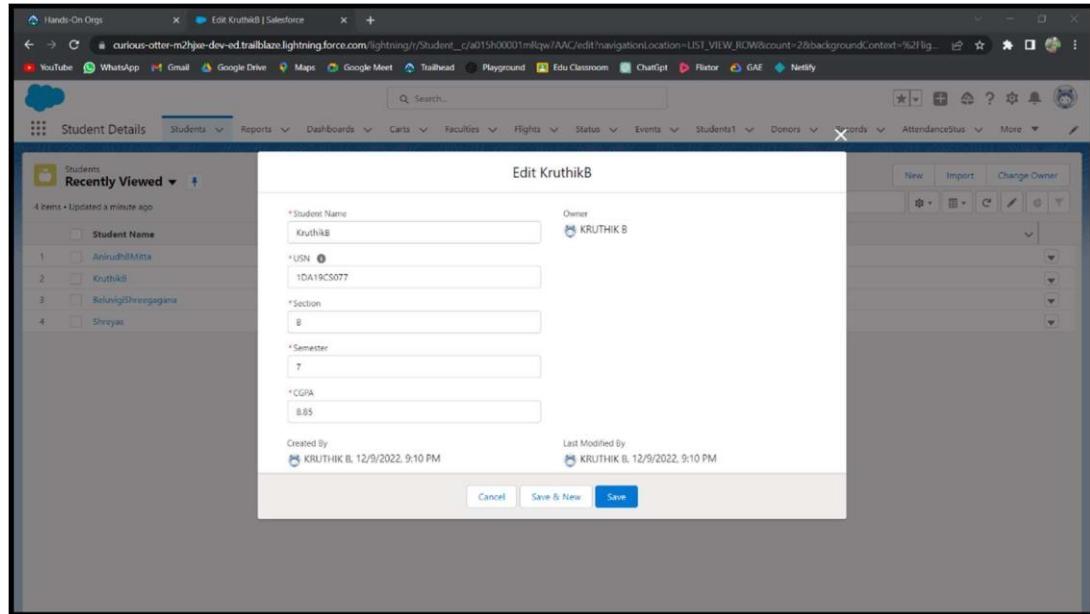


- d. Similarly give validation rules for email and phone number if you have added.
12. Now create an application: navigate to home and search for “App Manager”. Click on it
13. Click on “New Lightning App” to create a Lightning Application.
14. Name it as “Student Details”, give your description, upload an image and change colors as you wish, click on “Next”.

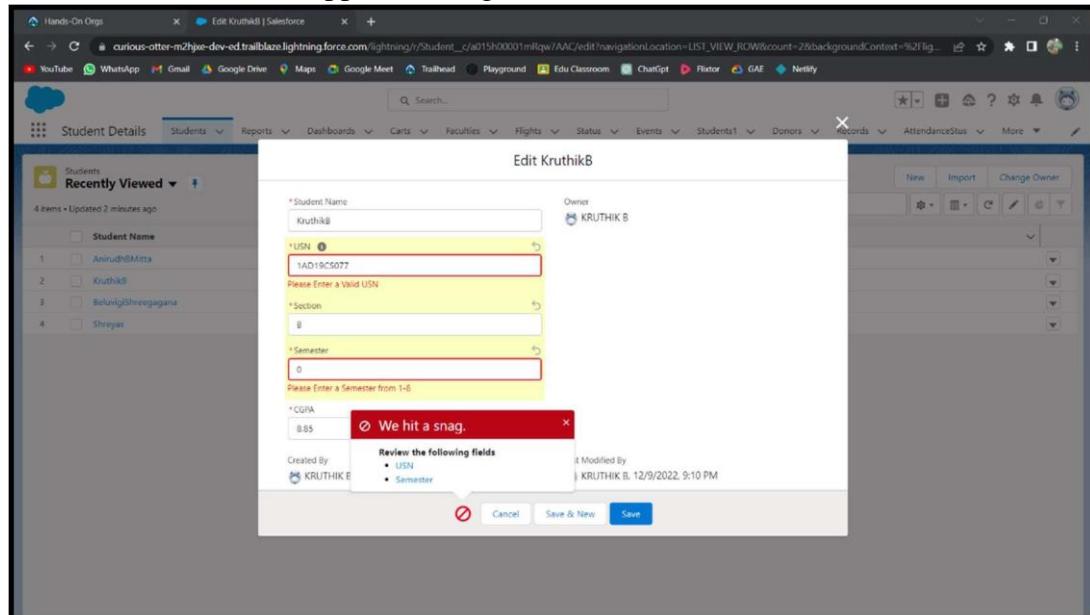


15. Set navigation style to standard navigation click next, let the utility bar have defaults so click next.
16. In navigation items add Dashboard, Reports and Student click next.
17. Assign it to System Administrator profile then click save and finish.
18. Click on app launcher (3x3 dots on top left corner) then click on view all, you'll see your application. Click on it.
19. Now navigate to Students tab and click on “New”.

20. Enter the details.



21. Now let's check what happens if we give invalid details.



22. Enter few more details. Now we can use the reports and dashboard

23. First let us create a student report

- a. Navigate to “Reports” tab and click on “New folder”, give any name and click save.
- b. Click on “New Report” and search for “Students”, select it and click on “Start Report”
- c. On left you see Add column option. From there add all the columns you require.
- d. Also add a group if you wish to.
- e. Click on save and name the report as “Student Report” and save it in folder you had created.
- f. Click on Save and then Run.

The screenshot shows the 'Create Report' dialog box in the Salesforce Report Builder. The 'Report Type Name' is set to 'Activities with Students' and the 'Category' is 'Standard'. The 'Report Type' dropdown shows 'Students' selected. On the right, the 'Details' panel shows the report is for 'Students' (Standard Report Type) and has no reports yet. It also lists 'Objects Used in Report Type' including 'Owner', 'Student', 'User', and 'Role'.

The screenshot shows the 'New Students Report' preview. The report includes fields for Semester, Student Name, CGPA, Email, Mobile, and Section. The preview table shows data for students Luna Lovegood, Neville Longbottom, Ginny Weasley, Hermione Granger, Harry Potter, Krumka B, Anush B Mitta, and Belugi Sheegagan. The report is currently in 'Outline' mode.

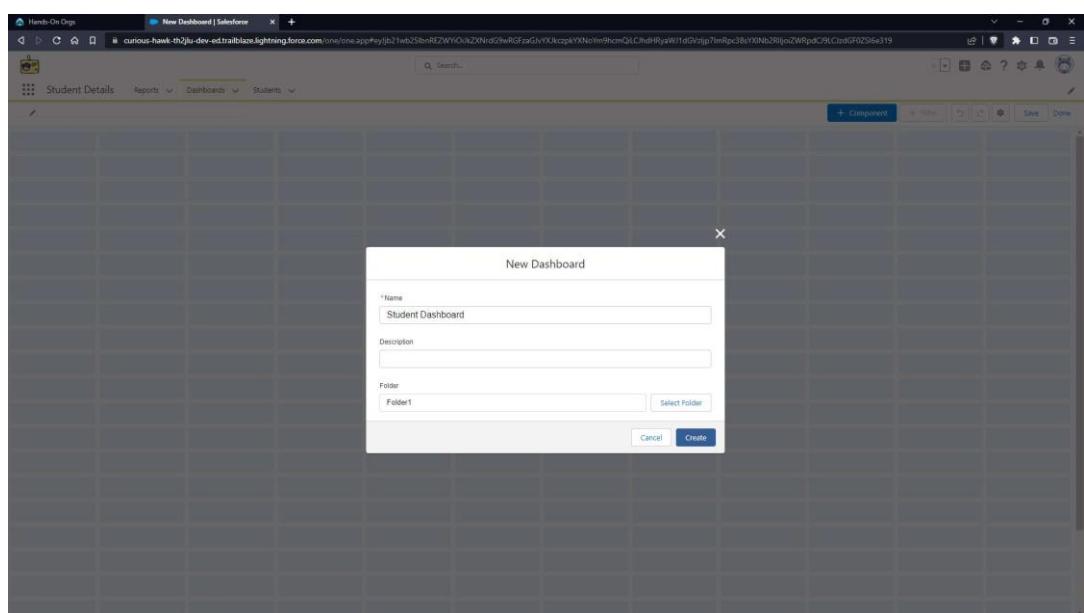
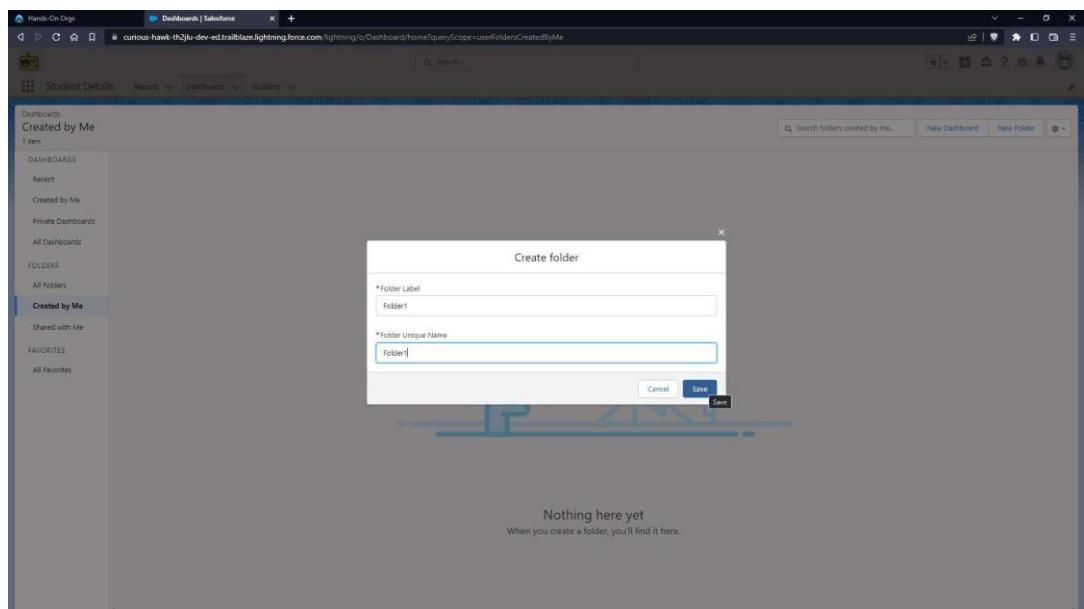
  

The screenshot shows the 'Select' dialog box for creating a new folder. The folder name is 'Folder1' and it is being created under 'All Folders > Folder1'. The message 'Nothing here yet. This folder has no subfolders.' is displayed. The 'Select Folder' button is highlighted.

24. Now let's create a dashboard

- Go to "Dashboard" tab and then click on "New folder" and give it any name.

- b. Click on “New Dashboard” and then name it as “Students Dashboard” and select folder that you have created Click on Create.
- c. Click on the report that you have created and click on that and click select.
- d. Select any style to represent the data in dashboard.
- e. Add any filter(s), otherwise it is optional.
- f. Click on Save and Click Run.



The image displays three screenshots of a Salesforce dashboard builder interface, showing the creation and final state of a circular chart component.

**Screenshot 1: Add Component Dialog**

This screenshot shows the "Add Component" dialog box. The "Report" dropdown is set to "Students Report". Under "Display As", the "Record Count" option is selected. On the right, a preview of a donut chart titled "Record Count" is shown, divided into four segments labeled 2, 2, 3, and 3. Below the preview, a "View Report (Students Report)" link is present. To the right of the preview, a "Semester" section shows four categories: 1 (blue), 2 (orange), 5 (green), and 7 (purple). At the bottom of the dialog are "Cancel" and "Add" buttons.

**Screenshot 2: Dashboard Preview**

This screenshot shows the dashboard after the component has been added. A success message "Dashboard saved" is displayed above the chart. The chart's preview and configuration options are visible, along with the "View Report (Students Report)" link and the "Semester" section.

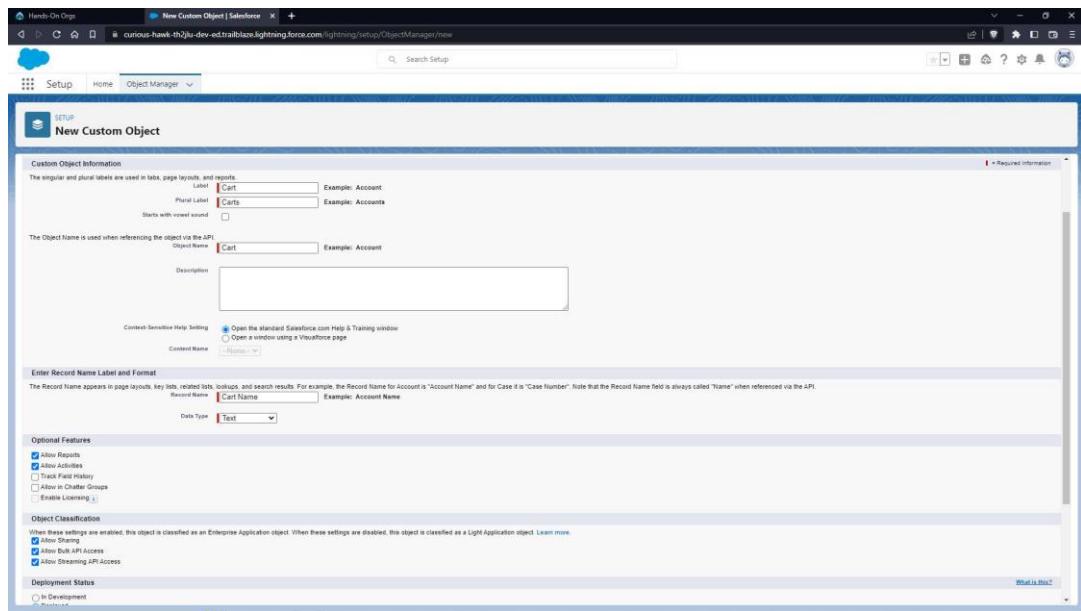
**Screenshot 3: Final Dashboard View**

This screenshot shows the final published dashboard. The chart component is now part of the main dashboard layout, displaying the same data and configuration as in the previous screens. The "View Report (Students Report)" link and the "Semester" section are also present.

## 2) Create a web application to implement an online cart for adding items to a shopping cart and deleting it.

### Steps:

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”.
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create> Custom Object” to create new Custom Object.
4. Name the object “Cart”.
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”



7. To create a Tab for the Object: Select any Tab Style for the object “Cart”. Click Next, Next, leave the defaults and save.
8. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: Item Name, Data Type: Text Area, make it as Required Field.
  - Field Label: Category, Data Type: Picklist, click radio button in front of Enter values, with each value separated by a new line Values are Books, Electronics & Accessories, Furniture & Home Appliances, Fashion – Men, Fashion – Women, Fashion – Kids, Footwear and Others.
  - Make it as Required Field and Restrict the values to the values in the picklist.
  - Field Label: Quantity, Data Type: Number, make it as Required Field.
  - Field Label: Price, Data Type: Currency (Length 16, Decimal Places 2), Make it as Required Field.

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Category	Category__c	Picklist		
Lightning Record Pages	Created By	CreatedById	Lookup(User)		
Buttons, Links, and Actions	Item Name	Name	Text(50)		
Compact Layouts	Last Modified By	LastModifiedById	Lookup(User)		
Field Sets	Owner	OwnerId	Lookup(User Group)		
Object Limits	Price	Price__c	Number(16, 2)		
Record Types	Quantity	Quantity__c	Number(2, 0)		
Related Lookup Filters					
Search Layouts					
List View Button Layout					
Restriction Rules					
Scoping Rules					
Triggers					
Flow Triggers					
Validation Rules					

To create an application:

10. Go to “Setup” and type “App Manager” in Quick Find Box.
11. Click on “New Lightning App” to create a Lightning Application.
12. Name it as “Online Shopping Cart”, give the description for your application.
13. Uploading Image and changing colours are optional, then click Next.
14. Navigation Style: Standard Navigation, click Next.
15. No need to add any Utility Bar, click Next.
16. Add the following Items: Carts, Reports and Dashboards, click Next.
17. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
18. Go to App Manager, select your application and select Carts and click “New” to add some details to your application.

Recently Viewed	
8 Items • Updated a few seconds ago	
<input type="checkbox"/>	Item Name
1	<input type="checkbox"/> The War of Lanka
2	<input type="checkbox"/> Mobile
3	<input type="checkbox"/> Laptop
4	<input type="checkbox"/> Cushion Couch
5	<input type="checkbox"/> Raavan Enemy of Aryavrat
6	<input type="checkbox"/> Sita Warrior of Nithila
7	<input type="checkbox"/> Jacket
8	<input type="checkbox"/> Ram Soion of Ikkavalu

19. Create Report

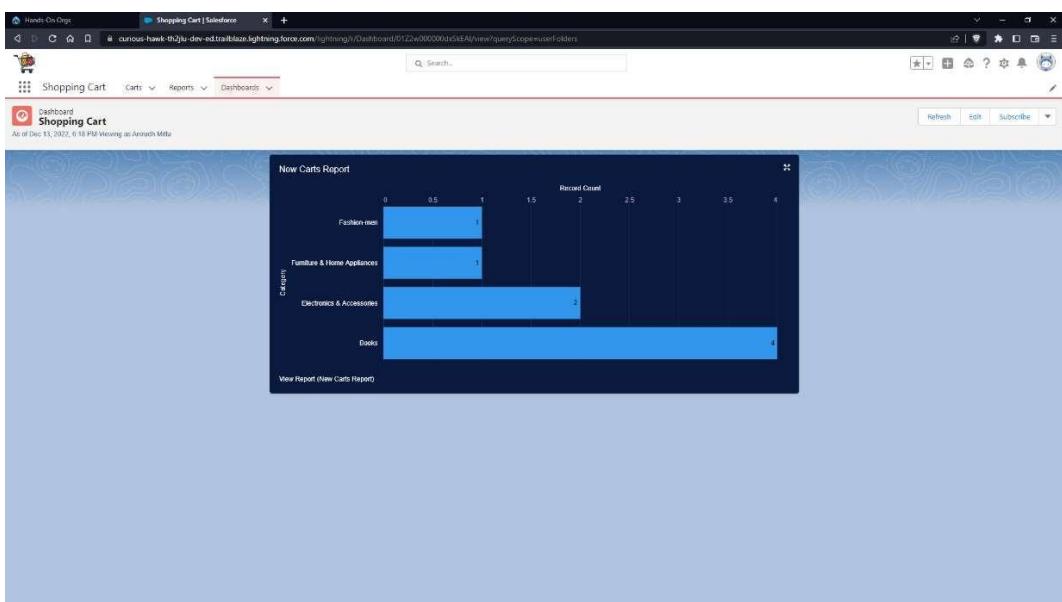
- Go to “Reports tab” Click on “New Folder” And give it any name and then click Save.
- Click on “New Report” and from search bar Search for “Carts” and then select it and click Continue.
- Add the required Columns to get the Completed Entered data.

- If you Want the report to be grouped by any specific fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Carts Report” and then select the folder which you have created.
- Click Save and then Click Run.

Category	Cart Item Name	Price	Quantity
Books (4)	Sita Warior of Mimila	249.00	3
	Ramayan Enemy of Aranyakath	449.00	1
	Ram Scion of Ikshvaku	320.00	2
	The War of Lanka	499.00	2
Electronics & Accessories (2)	Mobile	34999.00	2
	Laptop	49999.00	1
Fashion-men (1)	Jacket	999.00	1
Furniture & Home Appliances (1)	Cushion Couch	15999.00	1

## 20. Create Dashboard

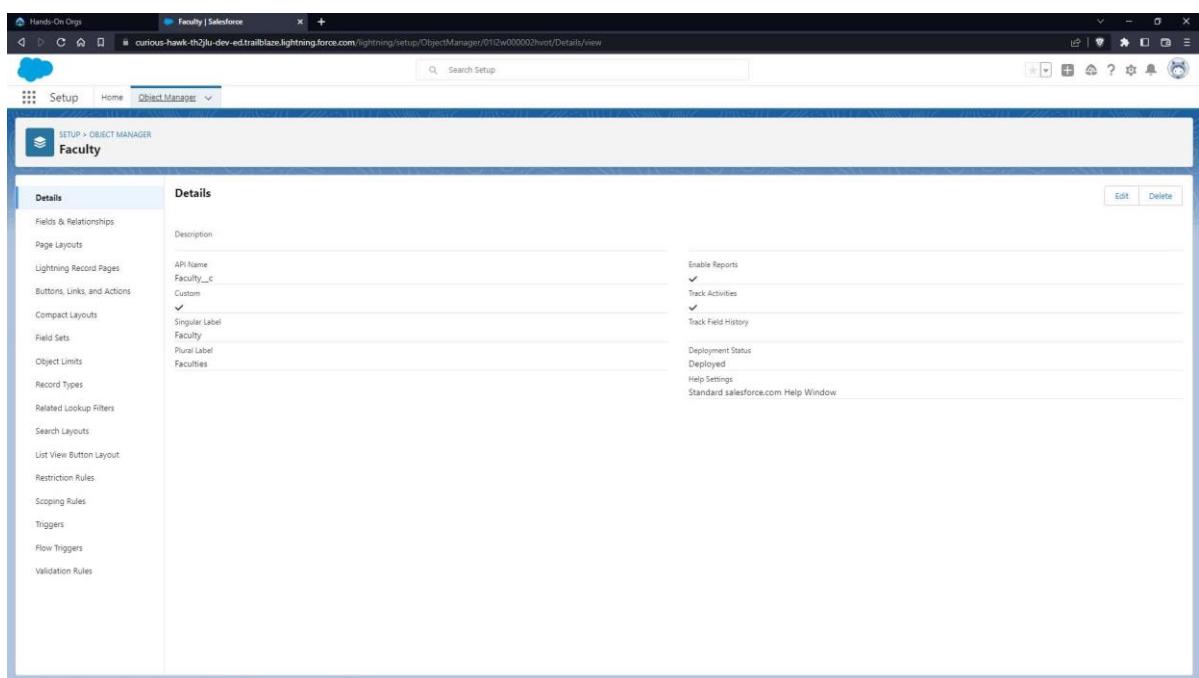
- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Shopping cart” and select folder that you have created Click on Create.
- Click on the report that you have created and click on that and click select.
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run.



**3) Create a web application to enter the faculty details like faculty ID, faculty name, and salary to a database and calculate the income tax to be paid by the faculty at the end of financial year.**

**Steps:**

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”.
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create> Custom Object” to create new Custom Object
4. Name the object “Faculty” 2. Allow Reports and Allow Search.
5. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”
6. To create a Tab for the Object: Select any Tab Style for the object “Faculty”. Click Next, Next, leave the defaults and save.



7. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”. Add the following fields one after the other:
  - Field Label: ID (Length 10), Data Type: Text, provide an example ID as Help Text, make it as Required Field, don’t allow Duplicate Values, make it as Case Insensitive and Set this field as the unique record identifier from an external system
  - Field Label: Salary, Data Type: Currency (Length 16, Decimal Places 2), Make it as Required Field.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Faculty Name	Name	Text(80)		
Faculty ID	Faculty_ID_c	Text(10) (External ID) (Unique Case insensitive)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		
Salary	Salary_c	Number(16, 2)		

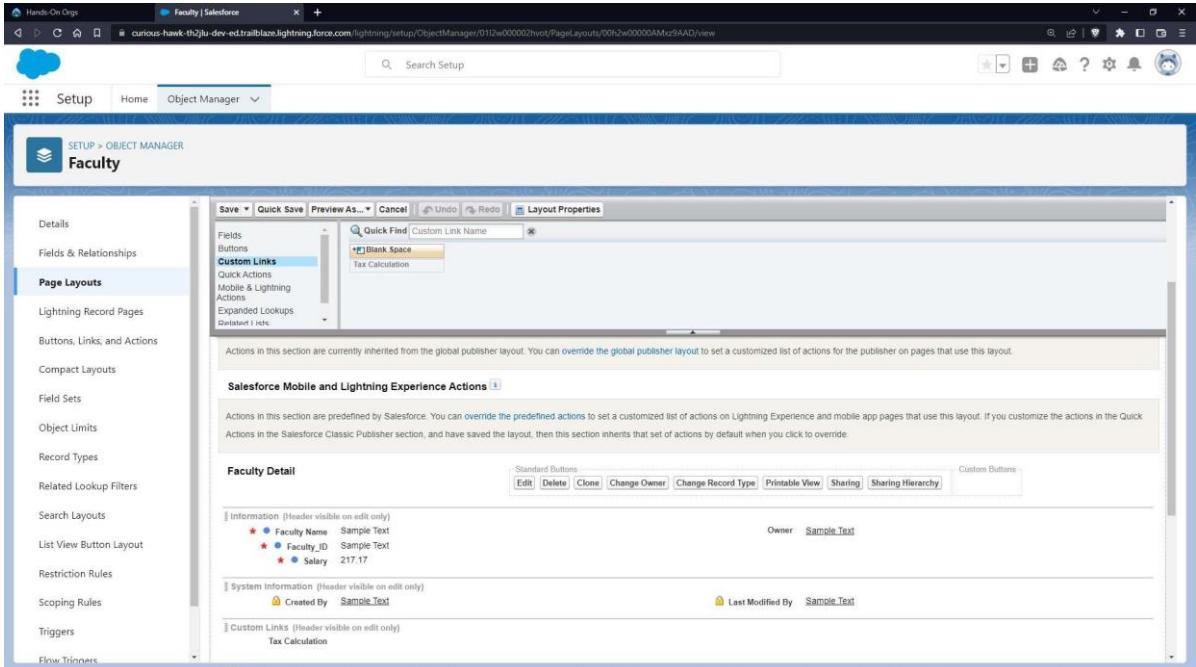
To calculate Income Tax to be paid:

8. Go to “Buttons, Links and Actions” of Faculty Object and click “New Button or Link”
9. Name it as “Tax Calculation”. Select the radio button “Detail Page Link” as it is a website link.
10. Behaviour: Display in new window.
11. Content Source: URL.
12. Field Type: Faculty
13. In the empty space provided, type  
<https://www.incometaxindia.gov.in/Pages/tools/income-tax-calculator-234ABC.aspx>  
 It is a link which redirects to the income tax calculation website.
14. Link Encoding: Unicode (UTF-8).
15. Click Save.

Label	Tax Calculation	Object Name	Faculty
Name	Tax_Calculation	Link Encoding	Unicode (UTF-8)
Behavior	Display in new window	Display Type	Detail Page Link
Button or Link URL	https://www.incometaxindia.gov.in/Pages/tools/income-tax-calculator-234ABC.aspx	Show Address Bar	<input type="checkbox"/>
Height (in pixels)	600	Show Scrollbars	<input checked="" type="checkbox"/>
Width (in pixels)		Show Toolbars	<input type="checkbox"/>
Window Position	No Preference	Show Menu Bar	<input type="checkbox"/>
Resizable	<input checked="" type="checkbox"/>	Show Status Bar	<input type="checkbox"/>
Description	Calculating tax based on salary	Created By	Anirudh Mitta, 12/13/2022, 6:42 PM
		Modified By	Anirudh Mitta, 12/13/2022, 6:42 PM

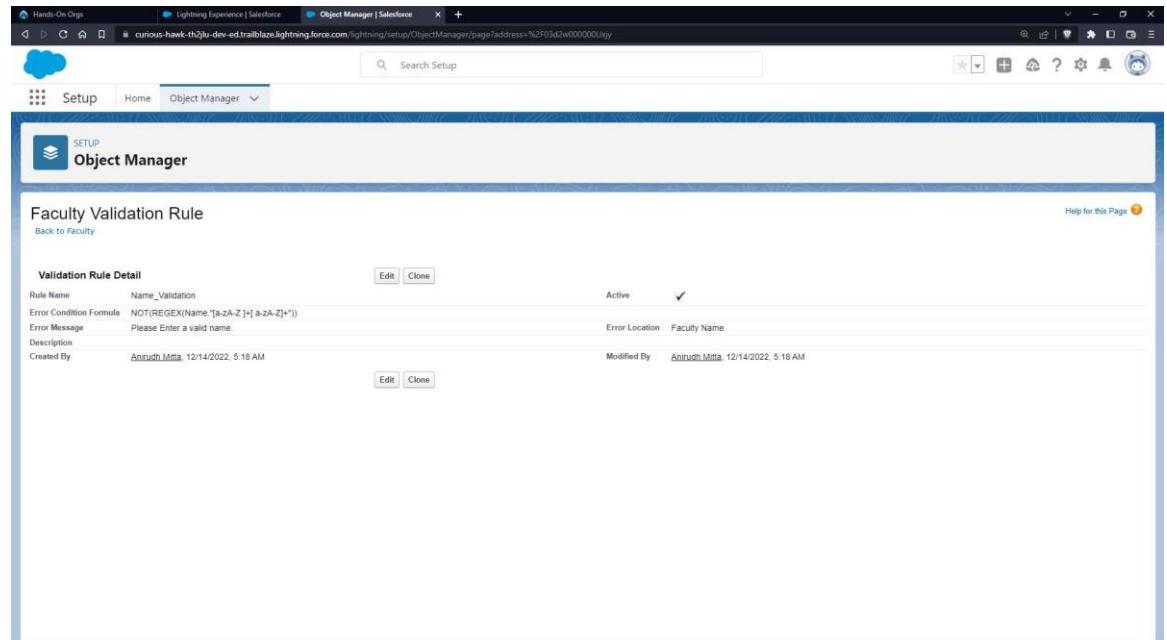
16. Go to Page Layout, Click Faculty Layout.

17. Click Custom Links, Drag and drop the “Tax Calculation” link in the Custom Link area.
18. Click Save.



19. To add a rule to the faculty name so that it should take only valid names:

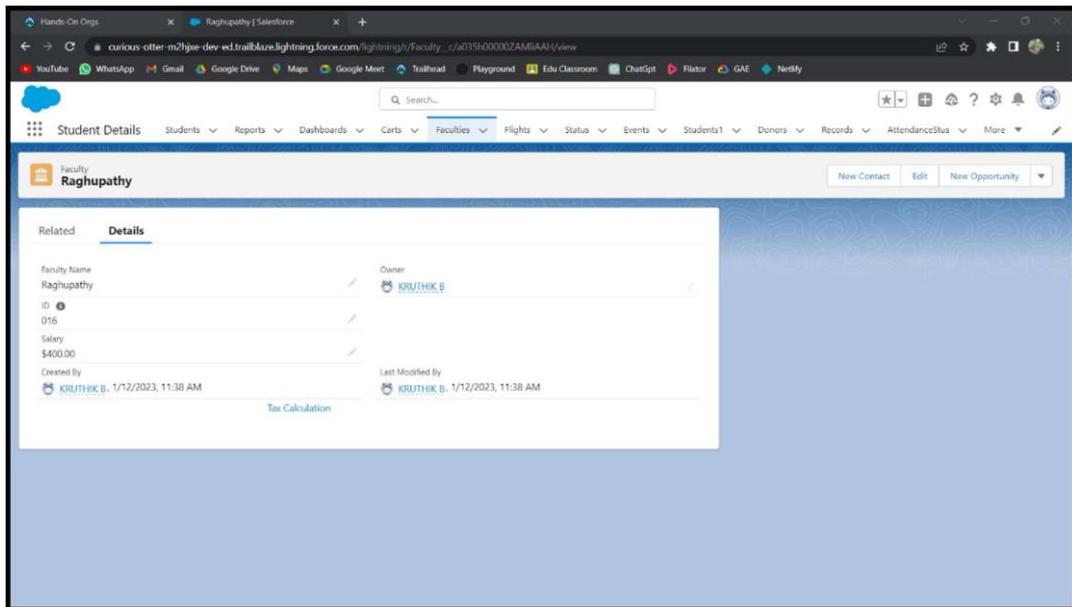
- Go to Validation Rule of Flight Object and click “New”
- Name it as “Name Validation”.
- Error Condition Formula: NOT(REGEX(Name,”\*a-zA-Z ]+[ a-zA-Z++”)).
- Error Message: Please Enter a valid name.
- Error Location: Field – Faculty name.
- Click Save.



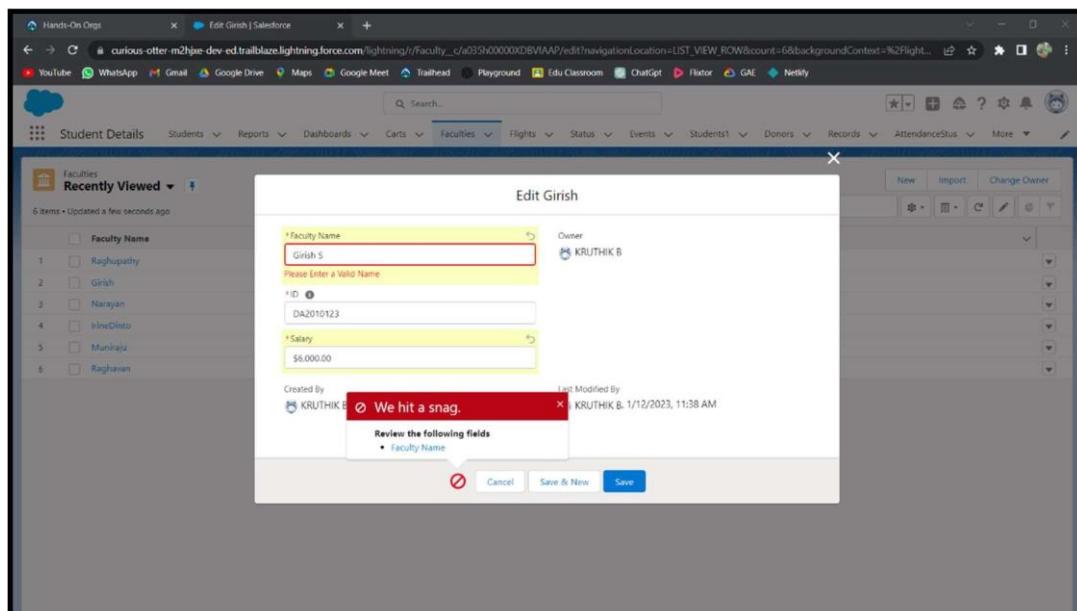
Create an application:

20. Go to “Setup” and type “App Manager” in Quick Find Box.
21. Click on “New Lightning App” to create a Lightning Application.
22. Name it as “Faculty Database”, give the description for your application.
23. Uploading Image and changing colours are optional, then click Next.

24. Navigation Style: Standard Navigation, click Next.
25. No need to add any Utility Bar, click Next.
26. Add the following Items: Faculties, Reports and Dashboards, click Next.
27. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
28. Go to App Manager, select your application and select Faculties and click “New” to add some details to your application.



29. Click the entry you added, go to details. Make sure you will get an error message when you enter an invalid name and invalid id.



30. Press the “Tax Calculation” link to calculate income tax. Click OK so that it will redirect you to the income tax calculator website.
31. Enter the required Details and press “Calculate”.

The screenshot shows the Income Tax Department's website with the title "INCOME AND TAX CALCULATOR". It includes dropdown menus for "Assessment Year", "Tax Payer", "Male / Female / Senior Citizen", "Residential Status", and "Income from Salary". There are also sections for "Income From House Property" and "Capital Gains" with "Show Details" buttons. To the right, there are three expandable sections: "TAX INFORMATION AND SERVICES", "TAX LAWS & RULES", and "INTERNATIONAL TAXATION".

### 32. To Create a Faculty Report:

- Go to “Reports tab” Click on “New Folder” And give it any name and click on Save.
- Click on “New Report” and from search bar Search for “Faculty” and then select it then Click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Faculties Report” and then select the folder which you have created.
- Click Save and then Click Run.

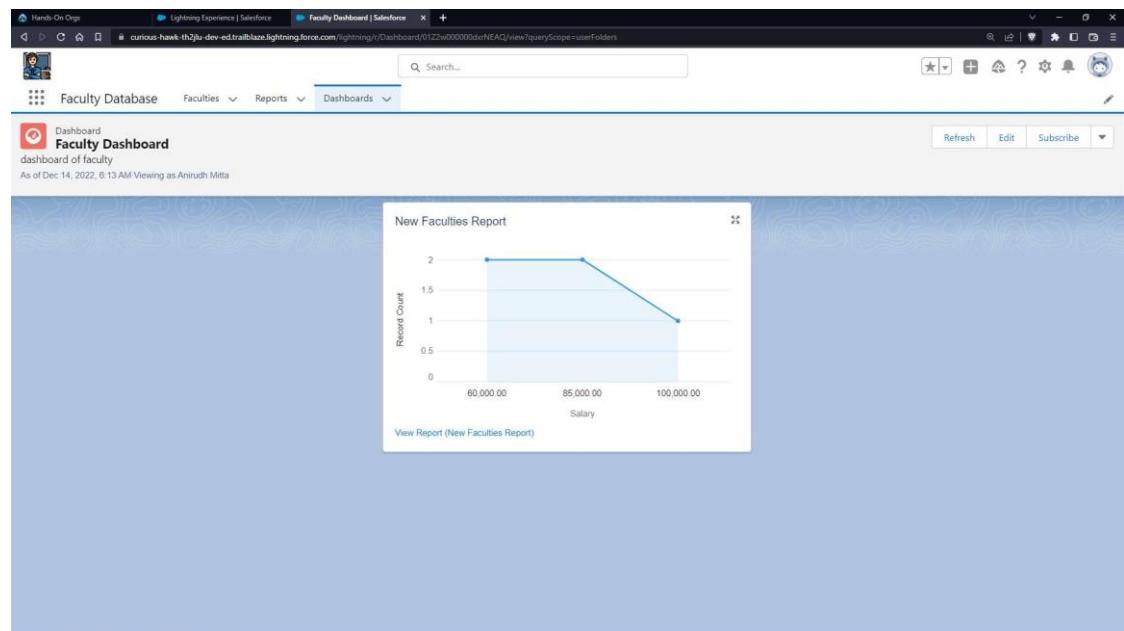
The screenshot shows a Salesforce report titled "Report: Faculties New Faculties Report". The report displays a table with the following data:

Salary	Faculty Name	Faculty ID
60,000.00 (2)	Minerva McGonagall Remus Lupin	FAC1MME657 FAC12AI879
85,000.00 (2)	Albus Dumbledore Severus Snape	FAC10CS113 FAC15E485
100,000.00 (1)	Professor Umbridge	FAC09C9998

### 33. To Create a Faculty Dashboard:

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Income Tax Dashboard” and select folder that you have created Click on Create.
- Click on the report that you have created and click on that and click select.

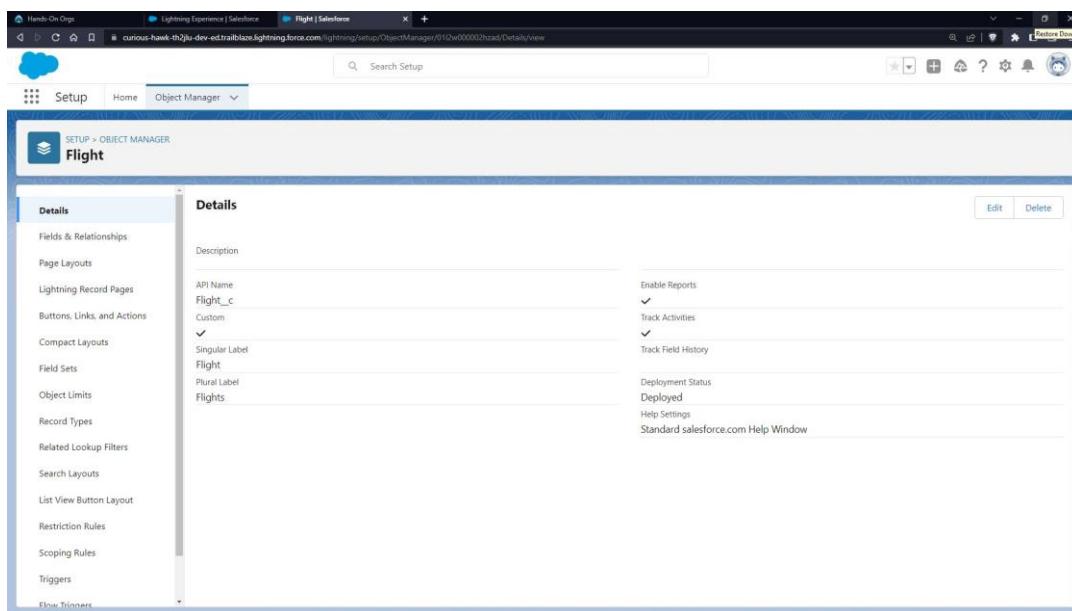
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run.



**4) Create a web application to book a flight from a source to destination and store the status of flight, and departure timings on database.**

**Steps:**

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create > Custom Object” to create new Custom Object.
4. Name the object “Flight”.
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
7. To create a Tab for the Object: Select any Tab Style for the object “Flight”. Click Next, Next, leave the defaults and save.



8. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: Source, Data Type: Text Area, make it as Required Field.
  - Field Label: Destination, Data Type: Text Area, make it as Required Field.
  - Field Label: Departure Timing, Data Type: Date/Time, make it as Required Field.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
Departure Time	Departure_Time__c	Date/Time		
Destination	Destination__c	Text Area(255)		
Flight Name	Name	Text(80)		✓
Last Modified By	LastModifiedBy	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		✓
Source	Source__c	Text Area(255)		

## 10. Validation rules:

### a. Date and time validation

- To add a rule to the departure timing so that it is greater than today's date and the present time:
- Go to Validation Rule of Flight Object and click "New"
- Name it as "Date Time should be in Range"
- Error Condition Formula: `Departure_Time__c < NOW()`
- Error Message: Departure Date/Time cannot be in past or present.
- Error Location: Field – Departure Timings.
- Click Save.

Rule Name	Date_Time_Validation	Active	
Error Condition Formula	<code>Departure_Time__c &lt; NOW()</code>		
Error Message	Departure Date/Time cannot be in past or present	Error Location	Departure Time
Description		Modified By	Anirudh.Mita. 12/14/2022, 6:39 AM
Created By	Anirudh.Mita. 12/14/2022, 6:39 AM		

### b. Source and destination validation:

- To add a rule to the Source and destination so that Source and destination should not be equal:
- Go to Validation Rule of Flight Object and click "New"

- Name it as “Source and destination Checking”
- Error Condition Formula: Source\_c = Destination\_c
- Error Message: Source and destination can never be equal.
- Error Location: Top of the Page.
- Click Save

Flight Validation Rule

Validation Rule Detail

Rule Name	Source_and_Destination_Validation	Active	<input checked="" type="checkbox"/>
Error Condition Formula	Source_c = Destination_c	Error Location	Top of Page
Error Message	Source and destination cannot be same		
Description		Modified By	Anirudh Mitta, 12/14/2022, 6:49 AM
Created By	Anirudh Mitta, 12/14/2022, 6:49 AM		

Create one more object to provide status of the flight:

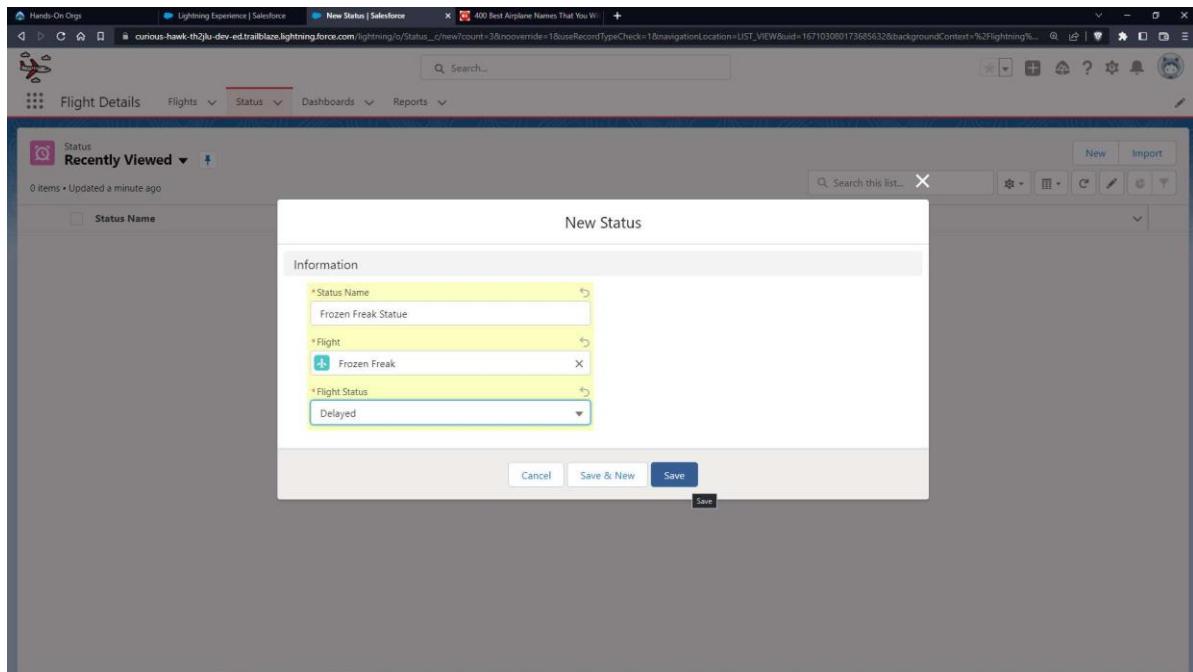
11. Name the Object “Status”
12. Allow Reports and Allow Search.
13. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”
14. Create a Tab for the Object.
15. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
16. Add the following fields one after the other:
  - Field Label: Flight Name, Data Type: Master-Detail Relationship, Related to: Flight.
  - Sharing Setting: Read-Only. Leave the defaults and save.
  - Master – Detail relationship is provided to enter status only to the existing flights.
  - Field Label: Flight Status, Data Type: Picklist, click radio button in front of Enter values, with each value separated by a new line.
  - Values are: Arrived, Cancelled, Delayed and Departed.
  - Make it as Required Field and Restrict the values to the values in the picklist.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
Flight	Flight_c	Master-Detail(Flight)		✓
Flight Status	Flight_Status__c	Picklist		▼
Last Modified By	LastModifiedBy	Lookup(User)		
Status Name	Name	Text(80)		✓

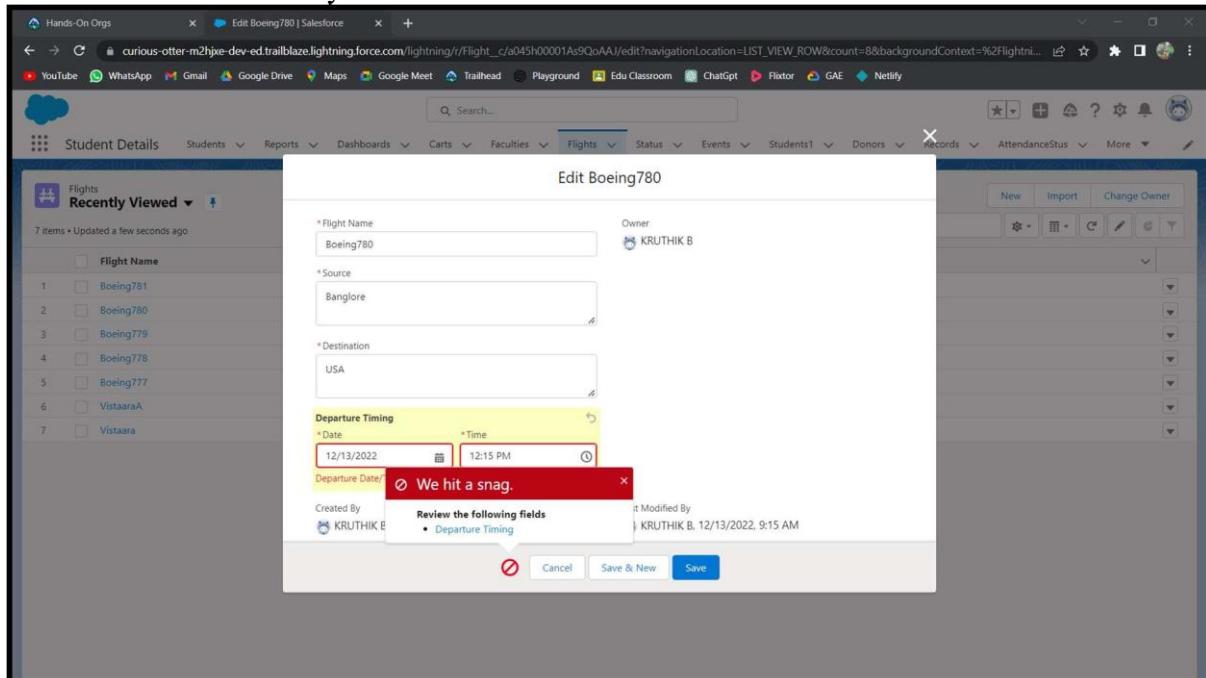
To create an application:

17. Go to “Setup” and type “App Manager” in Quick Find Box.
18. Click on “New Lightning App” to create a Lightning Application.
19. Name it as “Flight Details”, give the description for your application.
20. Uploading Image and changing colours are optional, then click Next.
21. Navigation Style: Standard Navigation, click Next.
22. No need to add any Utility Bar, click Next.
23. Add the following Items: Flights, Statuses, Reports and Dashboards, click Next.
24. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
25. Go to App Manager, select your application and select Flights and click “New” to add some details to your application.

Flight Name	Owner
Boeing781	KRUTHIK B



Make sure you will get an error message when you try to give the Departure Timing less than the current time and today's date.



## 26. Create a flight report

- Go to “Reports tab” Click on “New Folder” And give it any name and then click Save.
- Click on “New Report” and from search bar Search for “Flights” and then select it and Click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Flights Report” and then select the folder which you have created.
- Click Save and then Click Run

	Flight	Flight Name	Source	Destination
12/14/2022 (1)	Evil Voodoo	Rome	Pukket	
12/15/2022 (5)	Ironpunch	Kabul	Istanbul	
	Prime Eagle	Denver	Mosco	
	Driftight	Osaka	Tokyo	
	Shadowfire	Berlin	Nairobi	
	Frozen Freak	Bengaluru	Paris	
12/16/2022 (3)	Dragonrise	Jerusalem	Hyderabad	
	Darkfreak	Mumbai	Washington	
	Frozen Freak	Paris	Bengaluru	
12/17/2022 (1)	Skypulse	Kolkatta	Dubai	
12/18/2022 (1)	Sharp Legionnaire	Palermo	Lisbon	
12/19/2022 (2)	Bansheeward	Helsinki	Oslo	
	Roguerush	NewYork	Delhi	

## 27. Create a status report

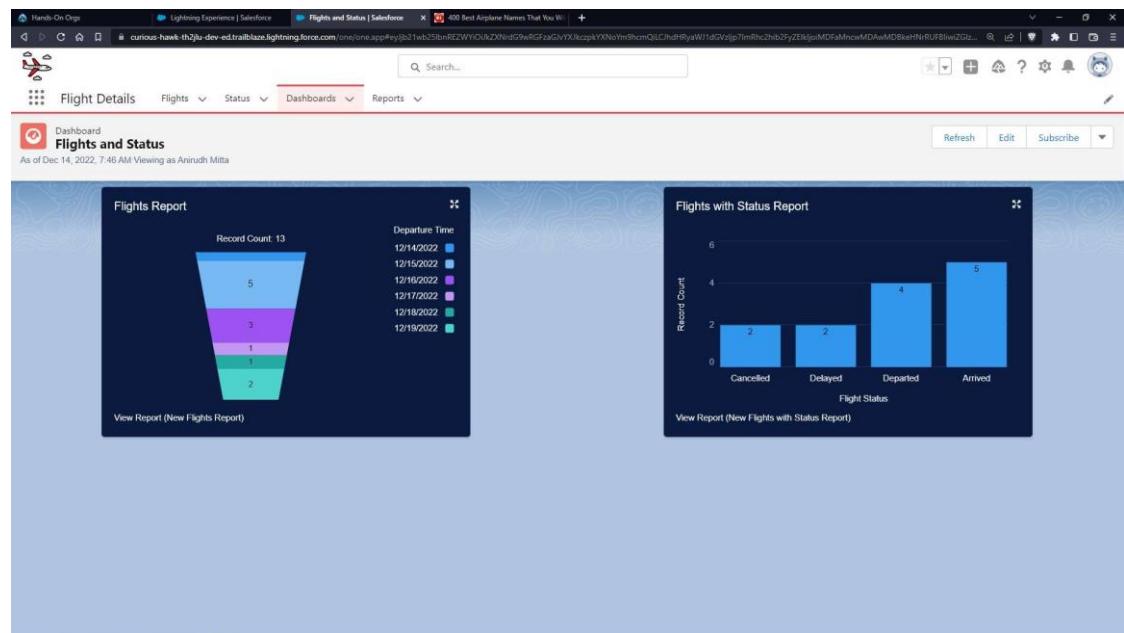
- Go to “Reports tab” Click on “New Folder” And give it any name and then click Save.
- Click on “New Report” and from search bar Search for “Flights with Status” and then select it and then click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Flights with status Report” and then select the folder which you have created.
- Click Save and then Click Run.

	Flight	Flight Name	Status	Status Name
Arrived (5)	Skypulse	Skypulse Status		
	Driftight	Driftight Status		
	Bansheeward	Bansheeward Status		
	Roguerush	Roguerush Status		
	Frozen Freak	Frozen Freak Status1		
Cancelled (2)	Evil Voodoo	Evil Voodoo Status		
	Dragonrise	Dragonrise Status		
Delayed (2)	Frozen Freak	Frozen Freak Status		
	Sharp Legionnaire	Sharp Legionnaire Status		
Departed (4)	Darkfreak	Darkfreak Status		
	Shadowfire	Shadowfire Status		
	Prime Eagle	Prime Eagle Status		
	Ironpunch	Ironpunch Status		

## 28. Create a Dashboard with both reports

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Flights and Status” and select folder that you have created Click on Create.
- Click on the report that you have created and click on that and click select.

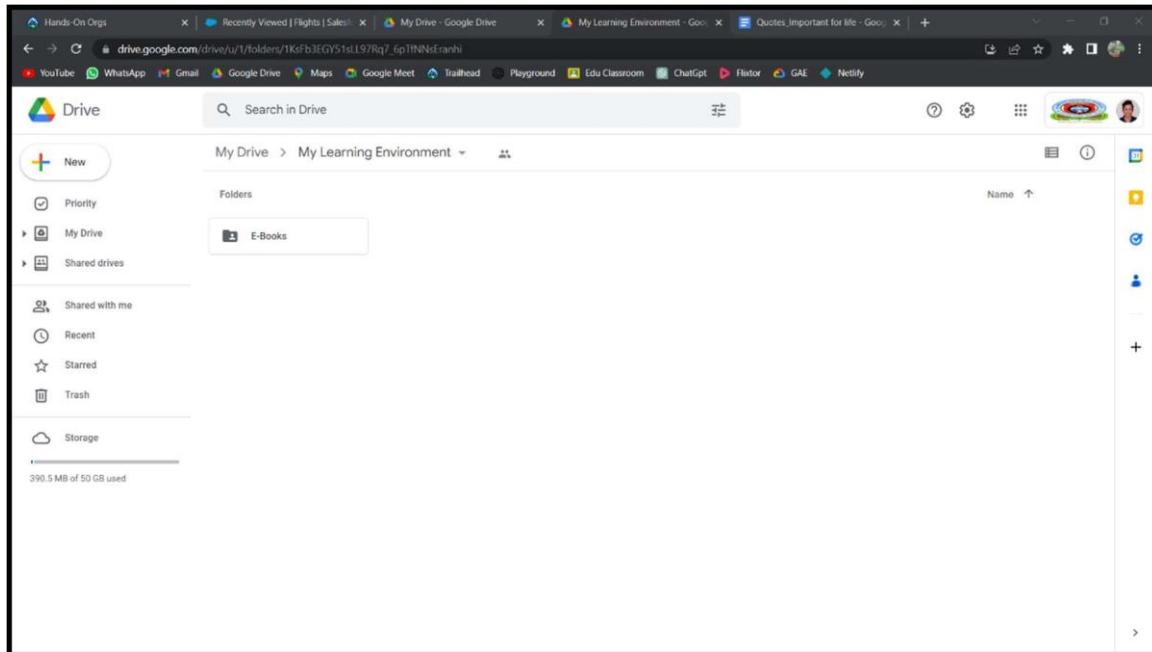
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run.



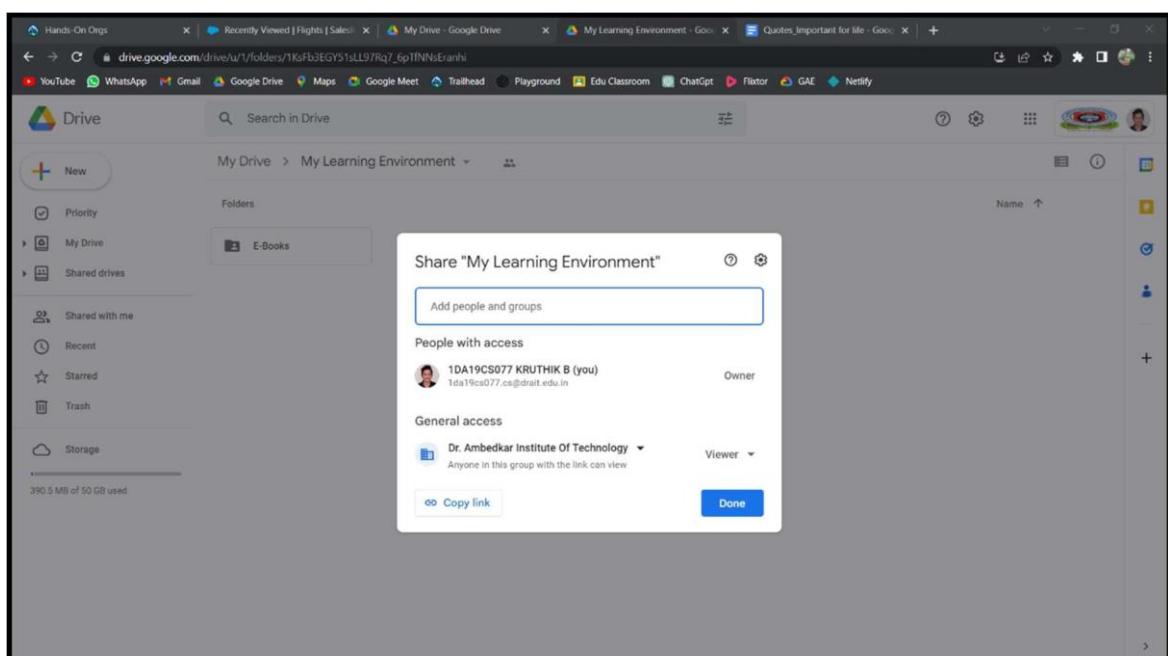
**5) Create a Collaborative learning environment for a particular learning topic using Google Apps. Google Drive, Google Docs and Google Slides must be used for hosting e-books, important articles and presentations respectively.**

**Steps:**

1. Open <http://drive.google.com/> and Sign In with your google account.
2. Create a new folder named “My Learning Environment” by clicking “New” button on the top left corner.

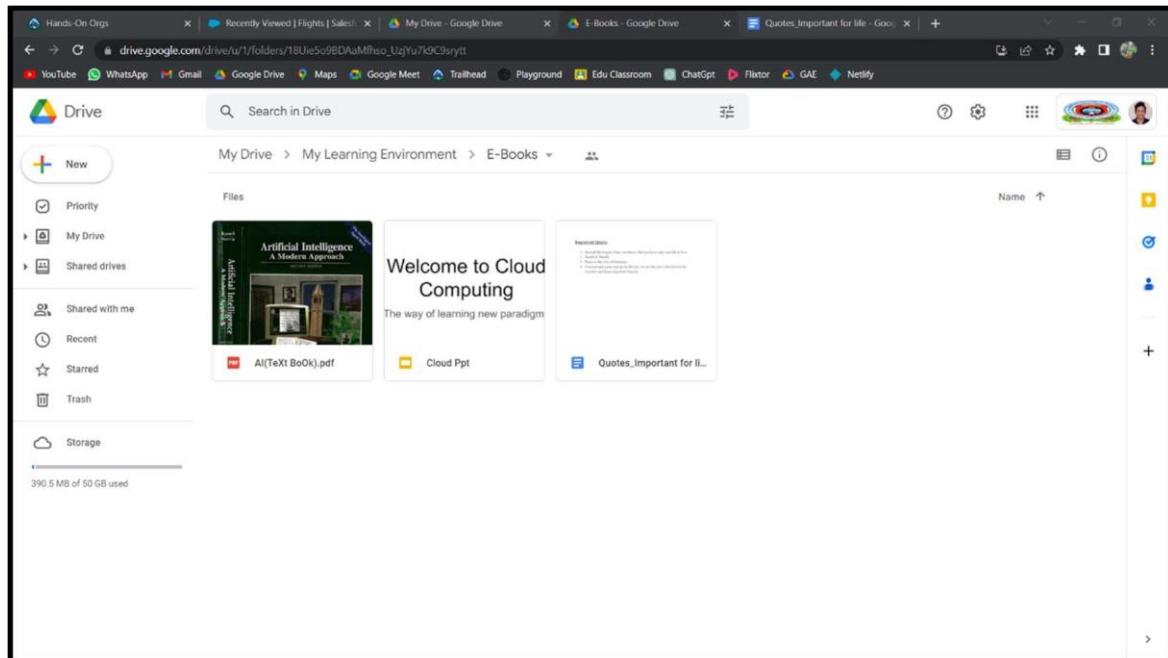


3. Right click on the folder created and tap “Share” then click on “Advanced”.
4. Under the “Who has Access” section click on “Change” of the first option.
5. Now check on the “On- Public on the web” option & set the Access to “Can View Only” and Click Save. This will make your folder to be accessible by anyone on the internet to view its contents and download them.

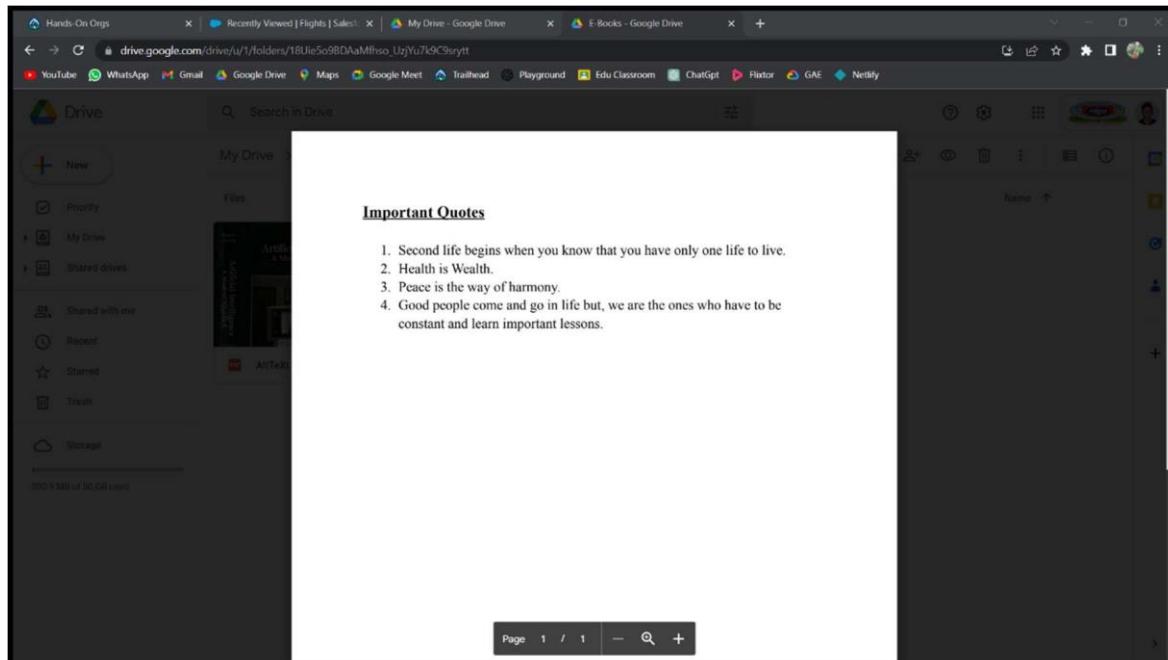


6. Copy the link and post it or share it to anyone you like.

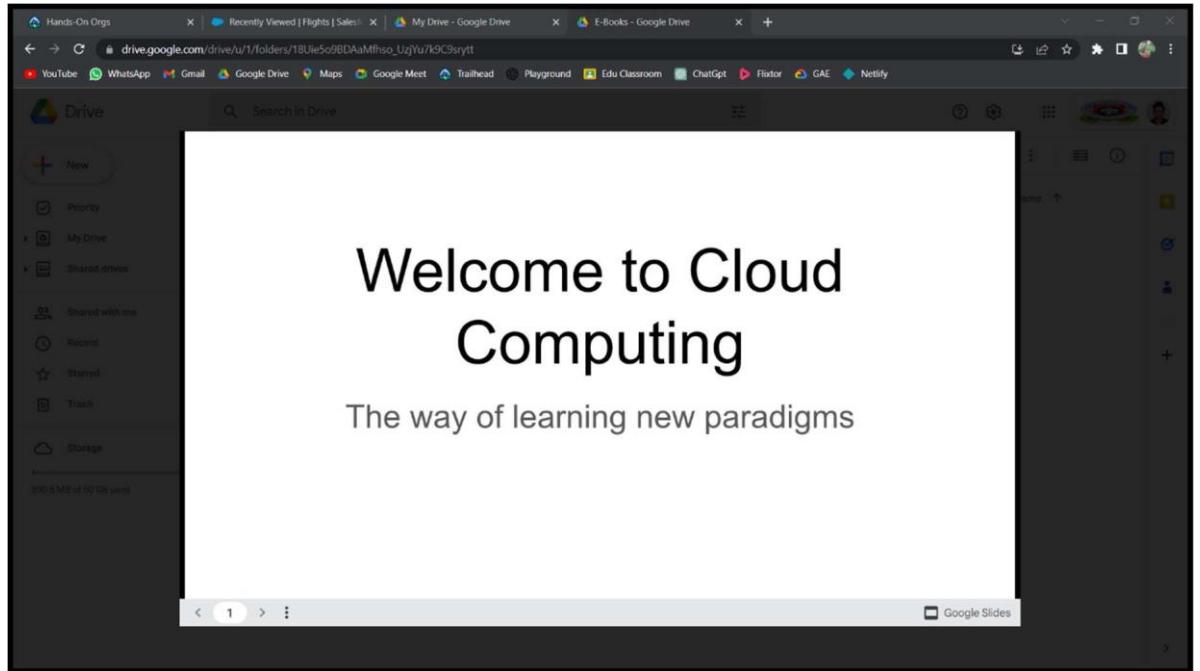
7. Double click on the folder you just created and click “New” button again.
8. Add these items:
  - a. Folder: Named “E-Books” where you will open the folder and click “New” and “Upload a file” like a Book from your hard drive.



- b. Google Docs: Named “Important Quotes” where you will add some important links to the doc file. The file will keep saving hence you need to press save. click “Share” if you wish to share it.



- c. Google Slide: Named “Welcome Folks” make a few changes and add your content and choose your theme.



- d. You may also add many other items as necessary.
- At any point in future if you wish to share this environment right click on the folder and click share. Copy the link and share.

([https://drive.google.com/drive/folders/1KsFb3EGY51sLL97Rq7\\_6pTfNNsEranhi?usp=share\\_link](https://drive.google.com/drive/folders/1KsFb3EGY51sLL97Rq7_6pTfNNsEranhi?usp=share_link))

**6) Develop Department events' registration app with an object containing event name, date/time, venue as parent relationship, another object containing student name, branch, event name , date/time , venue as child relationship.**

**Steps:**

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”.
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create > Custom Object” to create new Custom Object.
4. Name the object “Event”.
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
7. To create a Tab for the Object: Select any Tab Style for the object “Event”. Click Next, Next, leave the defaults and save.
8. To add fields to the Object:
  - Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: Event Time, Data Type: Date/Time, make it as Required Field.
  - Field Label: Event Venue, Data Type: Text Area, make it as Required Field.

The screenshot shows the Salesforce Object Manager interface for the 'Event' object. On the left, there's a sidebar with various setup options like Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Field Sets, Object Limits, Record Types, etc. The main area is titled 'Fields & Relationships' and lists six fields. The fields are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Event Name	Name	Text(80)		
Event Time	Event_Time__c	Date/Time		
Event Venue	Event_Venue__c	Text Area(255)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		

10. To add a rule to the Event Date/Time so that it is greater than today's date and the present time:
  - Go to Validation Rule of Event Object and click “New”.
  - Name it as “Date and Time in Range”.
  - Error Condition Formula: Event\_Time\_\_c<NOW ()
  - Error Message: Date or time less than the current one.
  - Error Location: Field – Event Time.
  - Click Save.

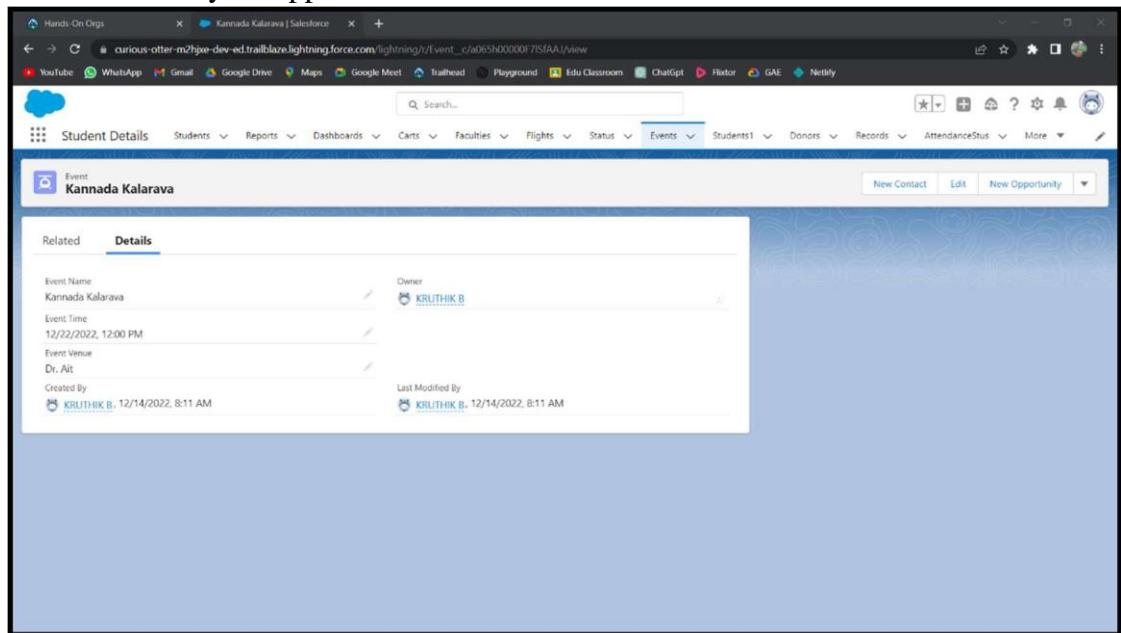
Create one more object to store student details:

11. Name the Object “Student”.
12. Allow Reports and Allow Search.
13. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
14. Create a Tab for the Object.
15. To add fields to the Object:
  - Go to “Fields & Relationships” option of Student object and Click “New”.
16. Add the following fields one after the other:
  - Field Label: Event Name, Data Type: Text.
  - Field Label: Event Time, Data Type: Master-Detail Relationship, Related to: Event. Sharing Setting: Read-Only. Leave the defaults and save.
  - Field Label: Event Venue, Data Type: Look up Relationship, Related to: Event.
  - Field Label: Branch, Data Type: Text, Make it as a Required Field.

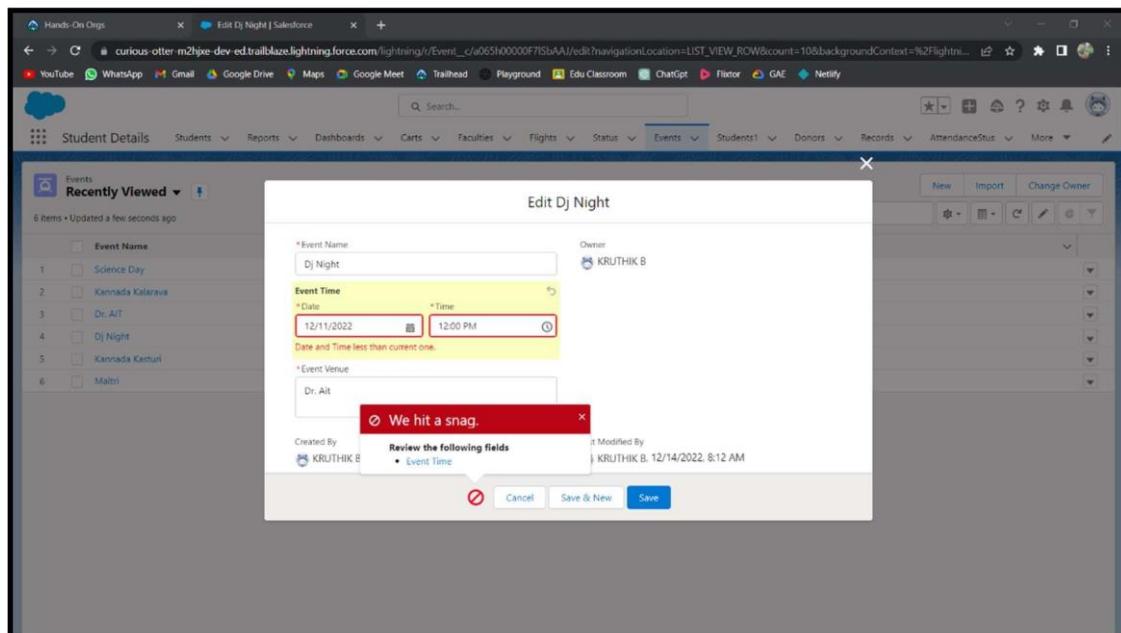
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Branch	Branch_c	Text(80)		
Created By	CreatedById	Lookup(User)		
Event	Event_c	Master-Detail(Event)		
Event name	Event_name_c	Text(80)		
Event Venue	Event_Venue_c	Lookup(Event)		
Last Modified By	LastModifiedById	Lookup(User)		
Student Name	Name	Text(80)		

17. To create an application:
18. Go to “Setup” and type “App Manager” in Quick Find Box.

19. Click on “New Lightning App” to create a Lightning Application.
20. Name it as “Events’ Registrations”, give the description for your application.
21. Uploading Image and changing colours are optional, then click Next.
22. Navigation Style: Standard Navigation, click Next.
23. No need to add any Utility Bar, click Next.
24. Add the following Items: Events, Students, Reports and Dashboards, click Next.
25. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
26. Go to App Manager, select your application and select Events and click “New” to add some details to your application.



27. Make sure you will get an error message when you try to give the Event Time less than the current time and today’s date and also you can’t select events which are not there in the list.



28. To Create an Event Report:
  - Go to “Reports tab” Click on “New Folder” And give it any name and Click Save.

- Click on “New Report” and from search bar Search for “Events” and then select it then Click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Events Report” and then select the folder which you have created.
- Click Save and then Click Run.

The screenshot shows the Salesforce Report Builder interface. The report is titled "New Events Report" and is set to preview records for the "Events" object. The report structure includes "Event Venue" as a group row, "Event: Event Name" as a column, and "Event Time" as another column. The data table contains the following information:

Event Venue	Event: Event Name	Event Time
JPN Auditorium, Dr.AIT (3)	Kannada Kalarava	12/17/2022, 12:00 PM
	Freshers Day	12/20/2022, 11:15 AM
	Interrupt	12/18/2022, 1:15 PM
Online (1)	Intellectual Property Rights	12/19/2022, 12:00 PM
Open Auditorium, Dr.AIT (2)	Maithri 2k23	12/18/2022, 9:00 AM
	Ethic Day	12/19/2022, 9:15 AM

## 29. To Create a Students Report:

- Go to “Reports tab” Click on “New Folder” And give it any name and Click Save.
- Click on “New Report” and from search bar Search for “Students with Events” and then select it then Click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Events with Students Report” and then select the folder which you have created.

The screenshot shows the Salesforce Report Builder interface. The top navigation bar includes tabs for 'Events Registration', 'Events', 'Students', 'Reports', and 'Dashboards'. The current page is 'Events with Students'. The report preview shows a table with four columns: 'Event Time', 'Event: Event Name', 'Student: Student Name', and 'Event Venue'. The data in the table is as follows:

Event Time	Event: Event Name	Student: Student Name	Event Venue
12/17/2022 (1)	Kannada Kalarava	Katniss	JPN Auditorium, Dr.AIT
12/18/2022 (1)	Interrupt	Anirudh B Mitta	JPN Auditorium, Dr.AIT
12/19/2022 (2)	Ethic Day	Pete	Open Auditorium, Dr.AIT
	Intellectual Property Rights	Gale	Online
12/20/2022 (1)	Freshers Day	Heymich	JPN Auditorium, Dr.AIT

The left sidebar contains sections for 'Fields' (Groups, Columns), 'Outline', and 'Filters'. The bottom of the screen has buttons for 'Row Counts', 'Detail Rows', 'Subtotals', and 'Grand Total'.

### 30. To Create a Dashboard:

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Event Dashboard” and select folder that you have created, Click on Create.
- Click on the report that you have created and click on that and click select.
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on components
- Click another report you created and select a style to represent.
- Click on Save and Click Run.

The screenshot shows the Salesforce Dashboard titled 'Event Registration Dashboard'. The dashboard displays two reports: 'Events Report' and 'Events with Students Report'. The 'Events Report' section features a donut chart with three segments labeled 1, 2, 3, and 6. A legend indicates the segments correspond to 'JPN Auditorium, Dr.AIT' (blue), 'Online' (purple), and 'Open Auditorium, Dr.AIT' (yellow). The 'Events with Students Report' section features a gauge chart with a scale from 0 to 10, currently at 5. Both reports include a 'View Report' link.

**7) Develop Blood donation registration app with an object which records donors' name, age and blood group as parent relationship and another object containing haemoglobin level, donated or not details (if age>18) child relationship.**

**Steps:**

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create > Custom Object” to create new Custom Object.
4. Name the object “Donor”
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
7. To create a Tab for the Object: Select any Tab Style for the object “Donor”. Click Next, Next, leave the defaults and save.
8. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: Age, Data Type: Number (3,0).
  - Field Label: Blood Group, Data Type: Picklist, click radio button in front of Enter values, with each value separated by a new line.
  - Values are A+ve, B+ve, A-ve, B-ve, O+ve, O-ve, AB+ve, AB-ve.
  - Make it as Required Field and Restrict the values to the values in the Picklist.
  - Field Label: Gender, Data Type: Picklist, click radio button in front of Enter values, with each value separated by a new line.
  - Values are Male, Female, Others.
  - Field Label: Weight, Data Type: Number (3,2).

Fields & Relationships				
8 Items, Sorted by Field Label				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Age	Age__c	Number(2, 0)		
Blood Group	Blood_Group__c	Picklist		
Created By	CreatedById	Lookup(User)		
Donor Name	Name	Text(80)		✓
Gender	Gender__c	Picklist		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Weight	Weight__c	Number(3, 2)		

Create one more object to store Collection details:

10. Name the Object “Record”.
11. Allow Reports and Allow Search.

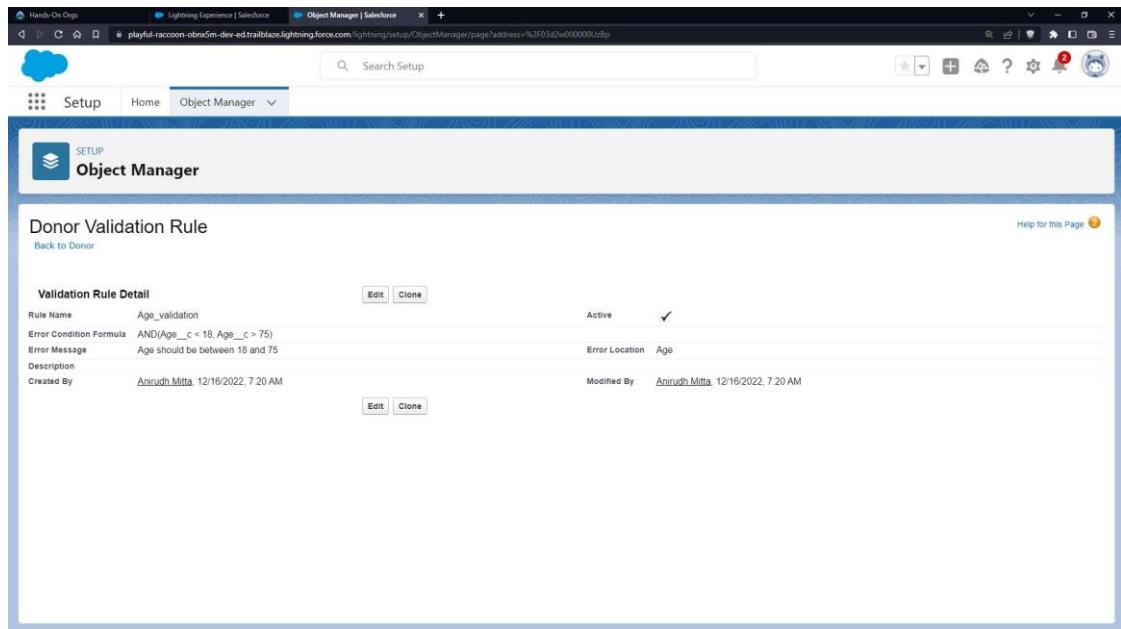
12. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”
13. Create a Tab for the Object.
14. To add fields to the Object:
  - Go to “Fields & Relationships” option of Student object and Click “New”.
15. Add the following fields one after the other:
  - Field Label: Haemoglobin level, Data Type: Number (2,2).
  - Field Label: Donor Name, Data Type: Look up Relationship, Related to: Donor.
  - Field Label: Blood Group, Data Type: Look up Relationship, Related to: Donor.
  - Field Label: Age, Data Type: Master-Detail Relationship, Related to: Donor. Sharing Setting: Read-Only. Leave the defaults and save.
  - Field Label: Status, Data Type: Picklist, Values: Donated, Not Donated, Make it as a Required Field.

Fields & Relationships					
	FIELD LABEL	DATA TYPE	CONTROLLING FIELD	INDEXED	
Page Layouts	Age	Age__c	Master-Detail(Donor)	✓	
Lightning Record Pages	Blood Group	Blood_Group__c	Lookup(Donor)		
Buttons, Links, and Actions	Created By	CreatedById	Lookup(User)	✓	
Compact Layouts	Donor Name	Donor_Name__c	Lookup(Donor)	✓	
Field Sets	Hemoglobin Level	Hemoglobin_Level__c	Number(2, 2)		
Object Limits	Last Modified By	LastModifiedById	Lookup(User)		
Record Types	Record Name	Name	Text(80)	✓	
Related Lookup Filters	Status	Status__c	Picklist		
Search Layouts					
List View Button Layout					
Restriction Rules					
Scoping Rules					
Triggers					
Flow Triggers					
Validation Rules					

## 16. Validation rules

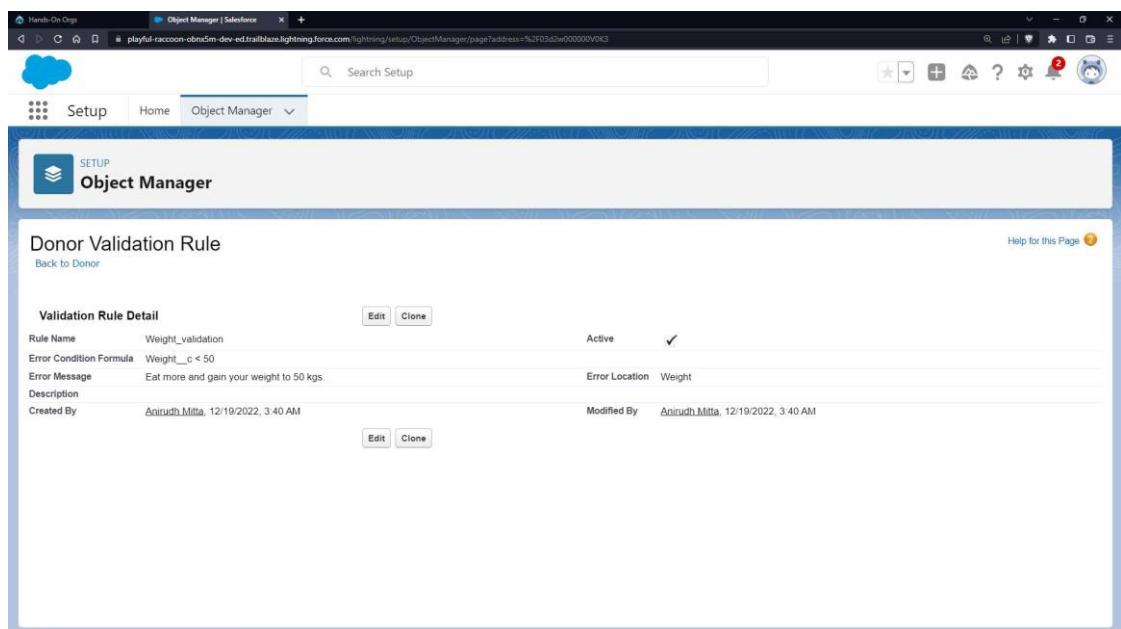
### a. Age Validation:

- To add a rule to the Donor age so that it is greater than 18 years:
- Go to Validation Rule of Donor Object and click “New”
- Name it as “Age Validation”.
- Error Condition Formula: AND (Age\_\_c < 18, Age\_\_c > 75).
- Error Message: Age must be greater than 18.
- Error Location: Field – Age.
- Click Save.



### b. Weight validation:

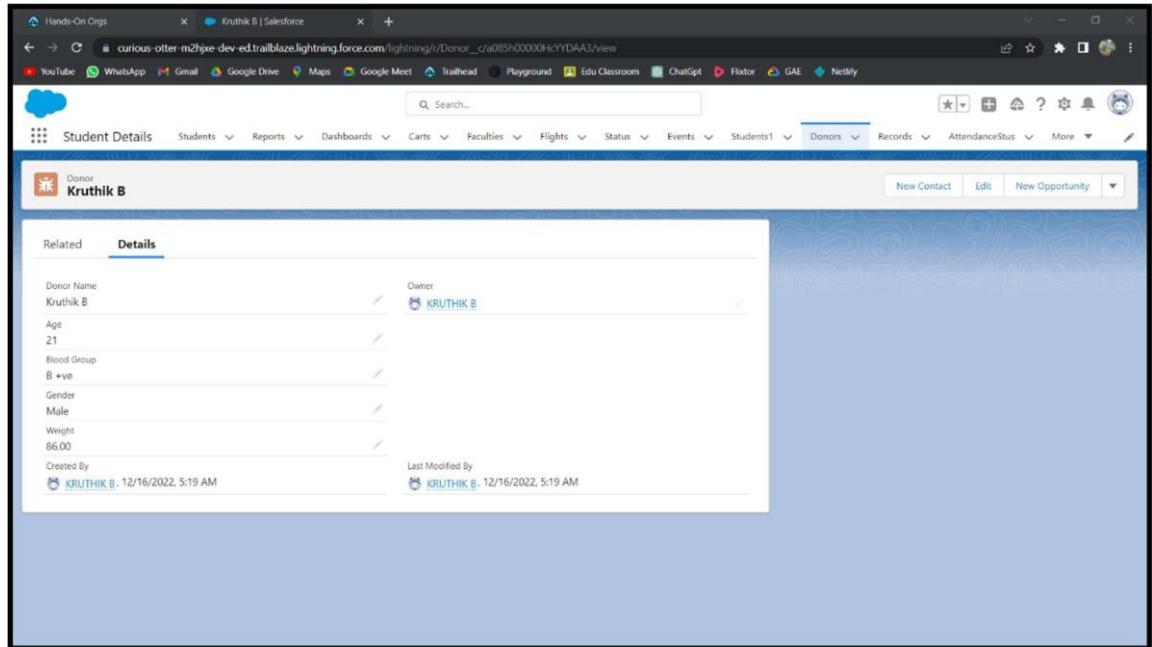
- To add a rule to the Donor Weight so that it Should be greater than 50:
- Go to Validation Rule of Donor Object and click “New”
- Name it as “Weight Validation”.
- Error Condition Formula: weight\_\_c < 50.
- Error Message: Eat more and gain your weight to 50 kgs.
- Error Location: Field –weight.
- Click Save.



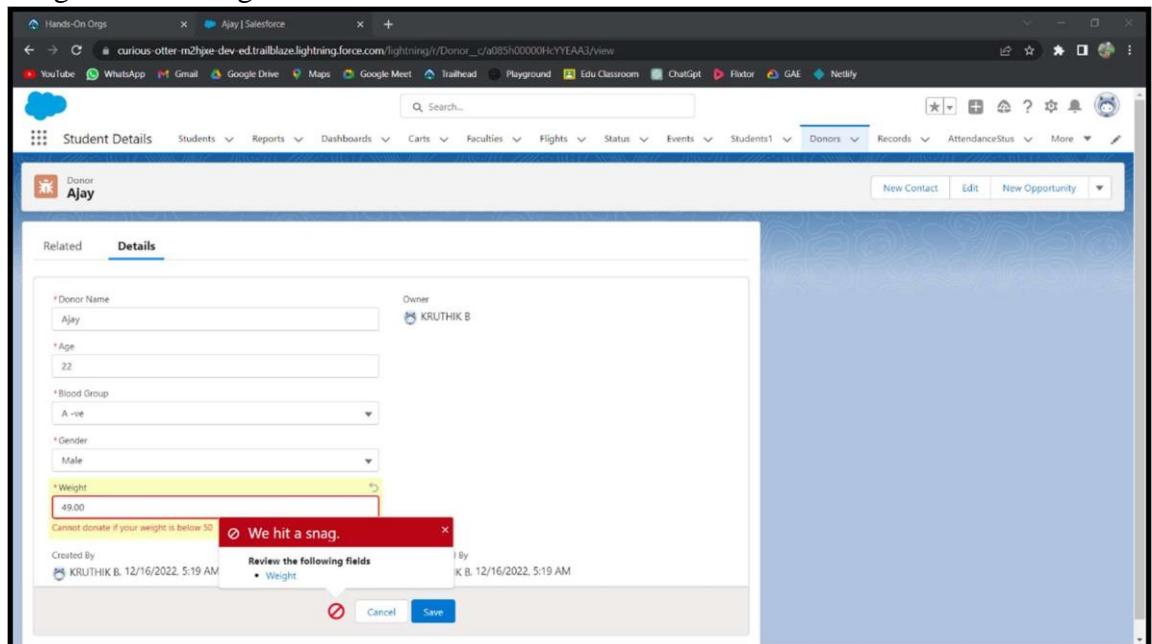
### To create an application:

17. Go to “Setup” and type “App Manager” in Quick Find Box.
18. Click on “New Lightning App” to create a Lightning Application.
19. Name it as “Blood Donation”, give the description for your application.
20. Uploading Image and changing colours are optional, then click Next.
21. Navigation Style: Standard Navigation, click Next.
22. No need to add any Utility Bar, click Next.

23. Add the following Items: Donors, Records, Reports and Dashboards, click Next.
24. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
25. Go to App Manager, select your application and select Donors and click “New” to add some details to your application.



26. Make sure you can't donate when your age is less than 18 years and when your weight is not 50 kgs.



27. Create a Donor Report:
  - Go to “Reports tab” Click on “New Folder” And give it any name then click on Save.
  - Click on “New Report” and from search bar Search for “Donors” and then select it and then click on Continue.
  - Add the required Columns to get the Complete Entered data.
  - If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.

- Click on save and name the report as “New Donor Report” and then select the folder which you have created.
- Click Save and then Click Run.

Blood Group	Donor: Donor Name	Age	Gender	Weight
A+ve (1)	Anirudh B Mitta	21	Male	63.00
B-ve (2)	Joye Reacher	39	Male	69.00
	Paul Hubble	45	Male	75.00
O+ve (2)	Frances Neagle	30	Female	68.00
	Josephine Hubble	40	Female	68.00
AB-ve (3)	Jack Reacher	35	Male	95.00
	Rosco Connin	29	Female	50.00
	Kiner	52	Male	65.00

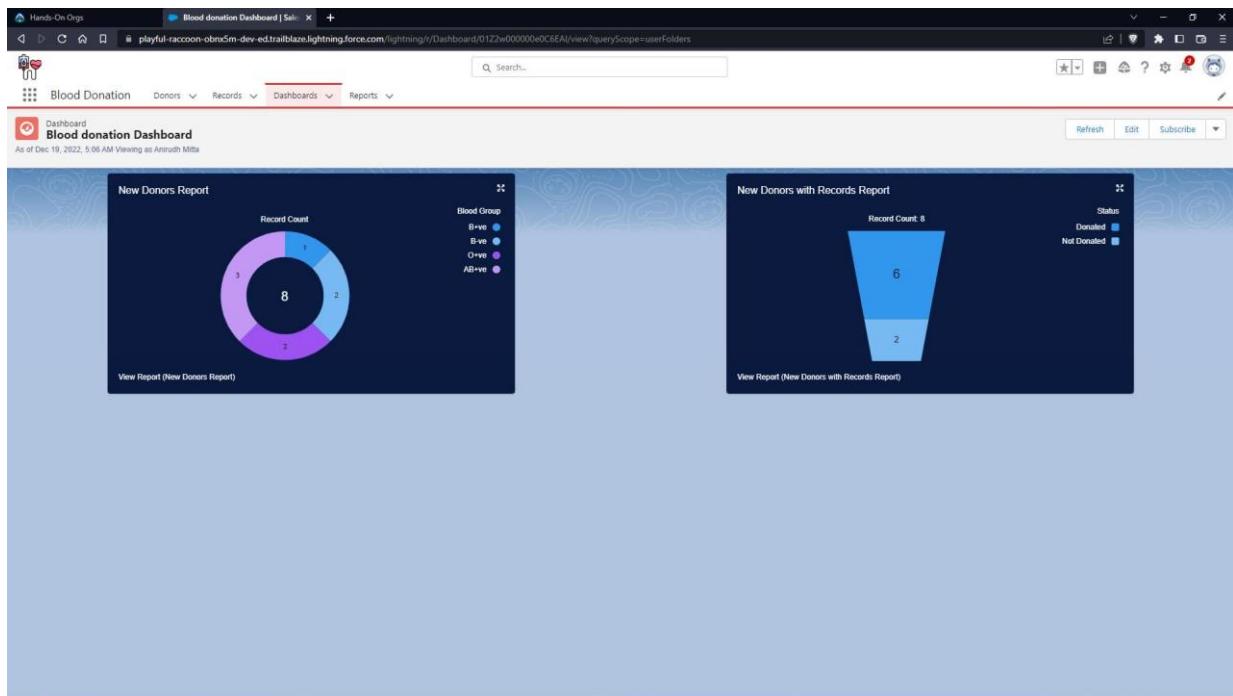
#### 28. Create a records report:

- Go to “Reports tab” Click on “New Folder” And give it any name then click on Save.
- Click on “New Report” and from search bar Search for “Donors with records” and then select it and then click on Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Donors with record Report” and then select the folder which you have created.
- Click Save and then Click Run.

Status	Donor: Donor Name	Record: Record Name
Donated (6)	Josephine Hubble	Record 1
	Joye Reacher	Record 4
	Kiner	Record 3
	Anirudh B Mitta	Record 8
	Rosco Connin	Record 6
	Jack Reacher	Record 7
Not Donated (2)	Frances Neagle	Record 2
	Paul Hubble	Record 5

## 29. Create a dashboard

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Donors Dashboard” and select folder that you have created Click on Create.
- Click on the report that you have created of donors and click on that and click select.
- Click on the report that you have created of records and click on that and click select.
- Select any style to represent the data of donors in dashboard.
- Select any style to represent the data of records in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run.



## 8) Develop Attendance maintenance app with an object to record student details and attendance and a provide a link to college websites' results webpage.

### Steps:

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create > Custom Object” to create new Custom Object.
4. Name the object “Student”.
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
7. To create a Tab for the Object: Select any Tab Style for the object “Student”. Click Next, Next, leave the defaults and save.
8. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: USN (Length 10), Data Type: Text, provide an example USN as Help Text, make it as Required Field and Don’t allow Duplicate Values and make it as Case Insensitive.
  - Field Label: Technical Branches, Data Type: Picklist, click radio button in front of Enter values, with each value separated by a new line.
  - Values are: CSE, ISE, TCE, EEE, EC etc.
  - Make it as Required Field and Restrict the values to the values in the picklist.
  - Field Label: Total Attendance %, Data Type: Percent (3,2).
  - Field Label: Semester, Data Type: Picklist.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Student | Salesforce', a search bar, and various setup icons. The main area is titled 'SETUP > OBJECT MANAGER' and shows the 'Student' object. On the left, there's a sidebar with links like 'Details', 'Fields & Relationships', 'Page Layouts', 'Lightning Record Pages', etc. The 'Fields & Relationships' section is active, displaying a table of fields. The table has columns for 'FIELD LABEL', 'FIELD NAME', 'DATA TYPE', 'CONTROLLING FIELD', and 'INDEXED'. The fields listed are: Branch (Branch\_\_c, Picklist), Created By (CreatedById, Lookup(User)), Last Modified By (LastModifiedById, Lookup(User)), Owner (OwnerId, Lookup(User,Group)), Semester (Semester\_\_c, Picklist), Student Name (Name, Text(80)), Total Attendance (Total\_Attendance\_\_c, Percent(3,2)), and USN (USN\_\_c, Text(10) (Unique Case Insensitive)).

To include Custom Links:

10. Go to “Buttons, Links and Actions” of “Student” Object and click “New Button or Link”.
11. Name it as “Check Details”.

12. Select the radio button “Detail Page Link” as it is a website link.
13. Behaviour: Display in new window.
14. Content Source: URL.
15. Field Type: Student.
16. In the empty space provided, type <http://www.drait.edu.in/>  
It is a link which redirects to the income tax calculation website.
17. Link Encoding: Unicode (UTF-8)
18. Click Save.

The screenshot shows the Salesforce Setup interface for creating a custom button. The left sidebar lists various setup categories like Page Layouts, Lightning Record Pages, and Buttons, Links, and Actions. The 'Buttons, Links, and Actions' tab is currently selected. On the right, a form titled 'Custom Button or Link Detail' is displayed for a 'Check Details' button. The button's properties include:

- Label:** Check Details
- Name:** Check\_Details
- Behavior:** Display in new window
- Button or Link URL:** http://www.drait.edu.in/
- Height (in pixels):** 600
- Width (in pixels):** 600
- Window Position:** No Preference
- Resizable:** ✓
- Description:** (empty)
- Object Name:** Student
- Link Encoding:** Unicode (UTF-8)
- Display Type:** Detail Page Link
- Show Address Bar:** (unchecked)
- Show Scrollbars:** (checked)
- Show Toolbars:** (unchecked)
- Show Menu Bar:** (unchecked)
- Show Status Bar:** (unchecked)

At the bottom of the form, it shows the button was created by Anirudh Mitta on 12/19/2022 at 9:11 AM and modified by Anirudh Mitta on the same date and time.

19. Go to Page Layout, Click Student Layout.
20. Click Custom Links, Drag and drop the “Check Details” link in the Custom Link area.
21. Click save.
22. To create an application:
23. Go to “Setup” and type “App Manager” in Quick Find Box.
24. Click on “New Lightning App” to create a Lightning Application.
25. Name it as “Attendance Management”, give the description for your application.
26. Uploading Image and changing colours are optional, then click Next.
27. Navigation Style: Standard Navigation, click Next.
28. No need to add any Utility Bar, click Next.
29. Add the following Items: Student, Records, Reports and Dashboards, click Next.
30. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
31. Go to App Manager, select your application and select Student and click “New” to add some details to your application.

Make sure you are redirected to the college website when you click the check details link.

### 32. Create a student report

- Go to “Reports tab” Click on “New Folder” And give it any name then click Save.
- Click on “New Report” and from search bar Search for “Attendance Management” and then select it then click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Students Report” and then select the folder which you have created.
- Click Save and then Click Run.

USN	Student Name	Branch	Semester	Total Attendance
1	Sheldon Lee Cooper	ECE	1	65.00%
2	Anirudh B Mitta	CSE	7	100.00%
3	John Sturges	TCE	5	77.00%
4	Missy Cooper	EEE	3	95.00%
5	Conn Tucker	ISE	8	85.00%

### 33. Create student dashboard

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Attendance Management” and select folder that you have created Click on Create.
- Click on the report that you have created and click on that and click select.
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run.

As of Dec 19, 2022, 9:45 AM Viewing as Anirudh Mitta

Refresh Edit Subscribe

New Students Report

View Report (New Students Report)

**9) Create a web application with objects to maintain database of an art gallery which contains objects like artists, arts, inventory and provide a link to any of the art gallery website.**

**Steps:**

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”.
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create > Custom Object” to create new Custom Object.
4. Name the object “Artist”.
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
7. To create a Tab for the Object: Select any Tab Style for the object “Artist”. Click Next, Next, leave the defaults and save.
8. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: ID (Length 10), Data Type: Text, provide an example ID as Help Text, make it as required Field, don’t allow Duplicate Values, make it as Case Insensitive and Set this field as the unique record identifier from an external system.
  - Field Label: Art Name and Details, Data Type: Text, Make it as a required field.
  - Field Label: Style, Data Type: Picklist,
  - Values: Pencil Sketching, craft design, free hand, Human portrait, drawing, painting etc.
  - Make it as Required Field and restrict values to the values in the picklist.

The screenshot shows the Salesforce Object Manager interface for the 'Artist' custom object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The main area is titled 'SETUP > OBJECT MANAGER' and shows the 'Artist' object. A table titled 'Fields & Relationships' displays seven items, sorted by Field Label. The columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The fields listed are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Art name and details	Art_name_and_details_c	Text Area(255)		
Artist Name	Name	Text(80)	✓	
Created By	CreatedById	Lookup(User)		
ID	ID_c	Text(10) (External ID) (Unique Case Insensitive)	✓	
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)	✓	
Style	Style_c	Picklist		

Create one more object to store Collection details:

10. Name the Object “Art”.

11. Allow Reports and Allow Search
12. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
13. Create a Tab for the Object.
14. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
15. Add the following fields one after the other:
  - Field Label: Art Name and Details, Data Type: Master-Detail Relationship, Related to: Artist. Sharing Setting: Read-Only. Leave the defaults and save.
  - Field Label: ID (Length 10), Data Type: Text, provide an example ID as Help Text, make it as required Field, don’t allow Duplicate Values, make it as Case Insensitive and Set this field as the unique record identifier from an external system.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Art Name	Name	Text(80)		✓
Art Name and Details	Art_Name_and_Details_c	Master-Detail(Artist)		✓
Created By	CreatedById	Lookup(User)		
ID	ID_c	Text(10) (External ID) (Unique Case Insensitive)		✓
Last Modified By	LastModifiedById	Lookup(User)		

Create one more object to store inventory details:

16. Name the Object “Inventory”.
17. Allow Reports and Allow Search.
18. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”
19. Create a Tab for the Object.
20. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
21. Add the following fields one after the other:
  - Field Label: Quantity, Data Type: Number. Make it as a required field.

To give a link to any art gallery website:

22. Go to “Buttons, Links and Actions” of Art Object and click “New Button or Link”.
23. Name it as “Gallery”.
24. Select the radio button “Detail Page Link” as it is a website link.
25. Behaviour: Display in new window.
26. Content Source: URL.
27. Field Type: Gallery.
28. In the empty space provided, type, <https://www.nga.gov/>

29. Link Encoding: Unicode (UTF-8).
30. Click Save.
31. Go to Page Layout, Click Art Layout.
32. Click Custom Links, Drag and drop the “Gallery” link in the Custom Link area.
33. Click save

**Inventory Custom Button or Link**  
**Gallery**

**Custom Button or Link Detail**

Label	Gallery	Object Name	Inventory
Name	Gallery	Link Encoding	Unicode (UTF-8)
Behavior	Display in new window	Display Type	Detail Page Link
Button or Link URL	https://www.nga.gov/	Show Address Bar	<input type="checkbox"/>
Height (in pixels)	600	Show Scrollbars	<input checked="" type="checkbox"/>
Width (in pixels)		Show Toolbars	<input type="checkbox"/>
Window Position	No Preference	Show Menu Bar	<input type="checkbox"/>
Resizable	<input checked="" type="checkbox"/>	Show Status Bar	<input type="checkbox"/>
Description			

Created By: Anirudh Mitta, 12/19/2022, 6:35 PM      Modified By: Anirudh Mitta, 12/19/2022, 6:35 PM

To add a rule to the Artist id so that it should take valid id:

- Go to Validation Rule of Artist Object and click “New”.
- Name it as “Artist id validation”.
- Error Condition Formula: NOT(BEGINS(ID\_\_c, 'ART')).
- Error Message: Please Enter a Valid id of an artist.
- Error Location: Field –id.
- Click Save.

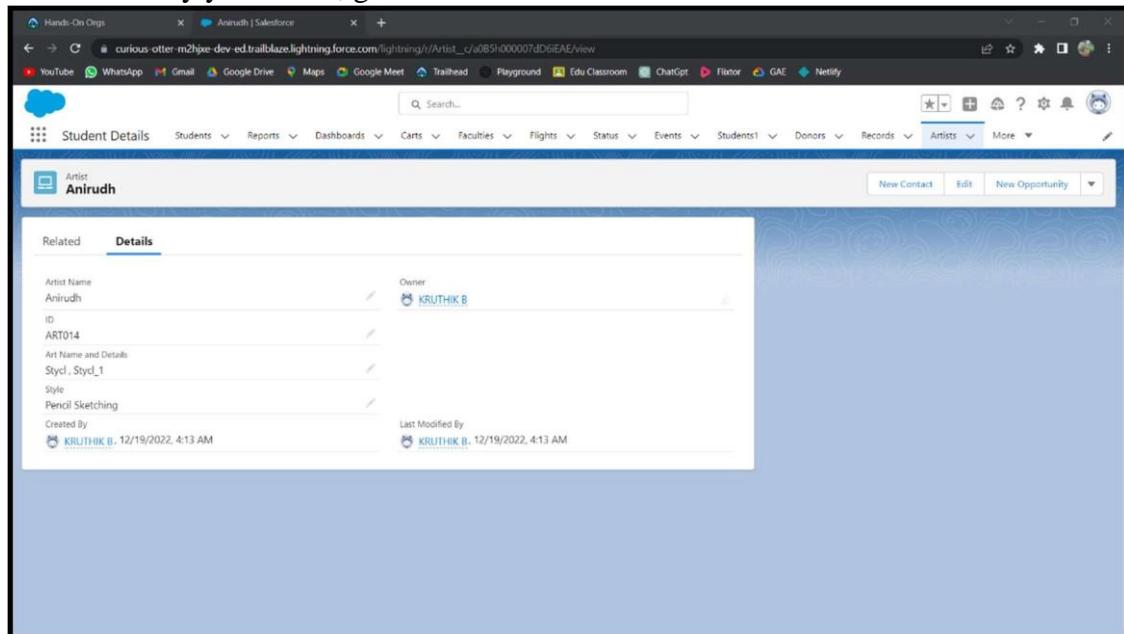
**Validation Rule Detail**

Rule Name	Artist_id_validation	Active	<input checked="" type="checkbox"/>
Error Condition Formula	NOT(BEGINS(ID__c, 'ART'))	Error Location	ID
Error Message	Please enter a valid ID	Created By	Anirudh Mitta, 12/19/2022, 6:39 PM
Description			
Created By	Modified By: Anirudh Mitta, 12/19/2022, 6:39 PM		

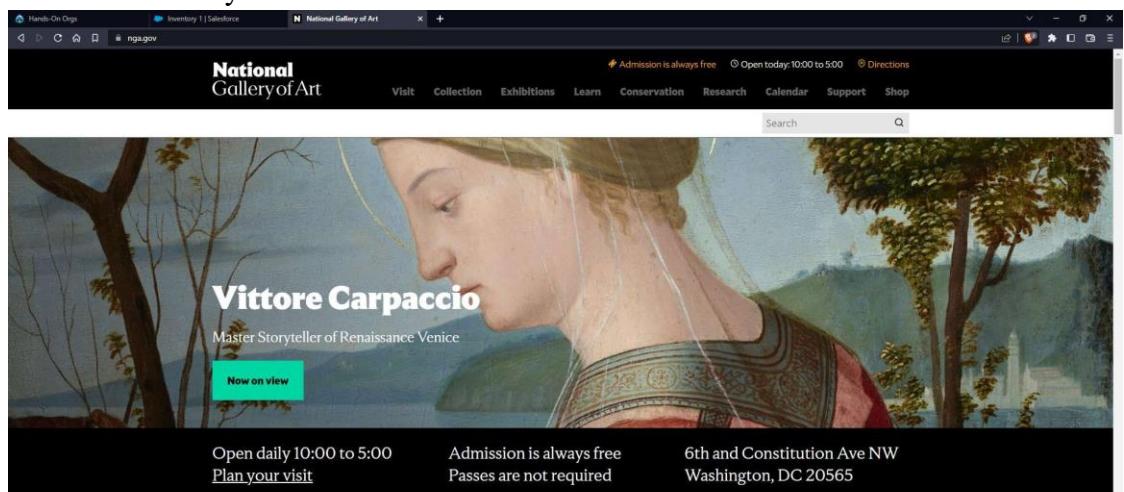
To create an application:

34. Go to “Setup” and type “App Manager” in Quick Find Box.
35. Click on “New Lightning App” to create a Lightning Application.

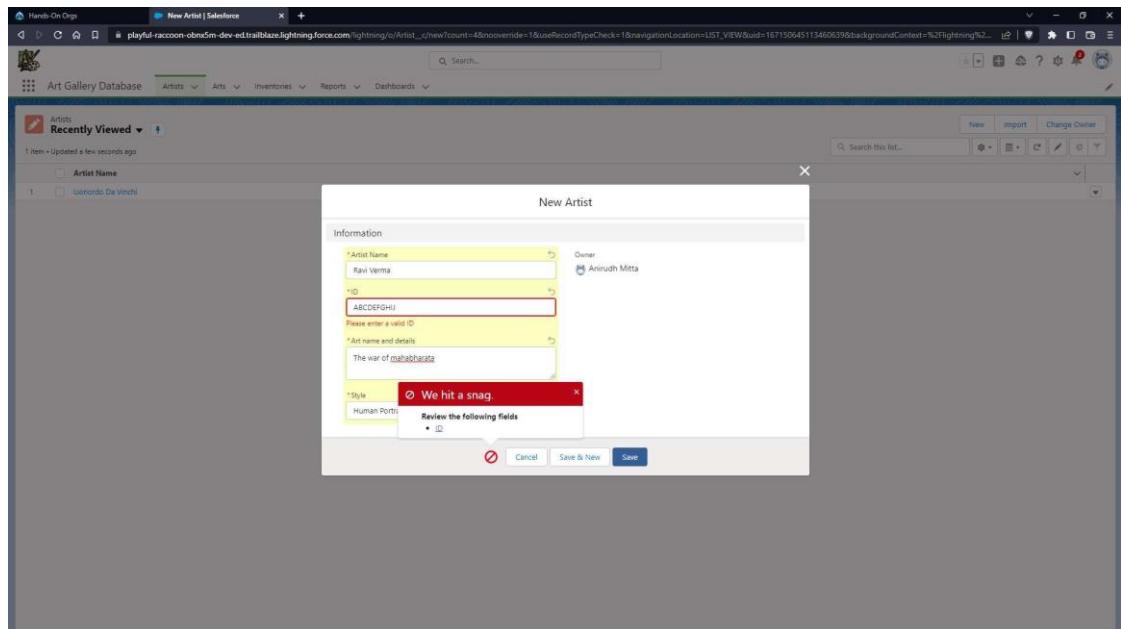
36. Name it as “Art Gallery Database”, give the description for your application.
37. Uploading Image and changing colours are optional, then click Next.
38. Navigation Style: Standard Navigation, click Next.
39. No need to add any Utility Bar, click Next.
40. Add the following Items: Artists, Arts, Inventories, Reports and Dashboards, click Next.
41. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
42. Go to App Manager, select your application and select Art gallery database and click “New” to add some details to your application.
43. Click the entry you added, go to details.



44. Press the “Gallery” link to check the details.



45. Click OK so that it will redirect you to the website.  
Make sure it should an error when an invalid id is given.



#### 46. Create an Artist report

- Go to “Reports tab” Click on “New Folder” And give it any name and Click Save.
- Click on “New Report” and from search bar Search for “Artists” and then select it and then Click Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Artists Report” and then select the folder which you have created.
- Click Save and then Click Run.

Report: Artists New Artists Report			
Total Records 5			
Style	Artist: Artist Name	ID	Art name and details
Pencil Sketching (1)	Rana Prathap	ART2000076	Sketch of Maharani
Craft Design (1)	Rama Krishna	ART2000013	Craft Model
Human Portrait (1)	Ravi Verma	ART2000043	The war of mahabharata
Painting (2)	Lionorda Da Vinci	ART1000035	Monalisa Painting
	Anirudh B Mitta	ART4000087	The painting of Emotions

#### 47. Create an Arts report

- Go to “Reports tab” Click on “New Folder” And give it any name.
- Click on “New Report” and from search bar Search for “Artists with Arts” and then select it.
- Add the required Columns to get the Complete Entered data.

- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Artists with Arts Report” and then select the folder which you have created.
- Click Save and then Click Run.

The screenshot shows a Salesforce Lightning interface with a report titled "New Artists with Arts Report". The report table has the following data:

Artist: Owner Name	Artist: Artist Name	Art: Art Name	ID
Anirudh Mitta (5)	Anirudh B Mitta	The painting of emotions	ART4000087
	Rana Prathap	Sketch of MahaRani	ART2000076
	Ravi Verma	The war of Mahabharat	ART2000043
	Rama Krishna	Craft Model	ART2000013
	Lionordo Da Vinchi	Monalisa Painting	ART1000035

#### 48. Create an Inventory report

- Go to “Reports tab” Click on “New Folder” And give it any name.
- Click on “New Report” and from search bar Search for “Inventories” and then select it.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Inventories Report” and then select the folder which you have created.
- Click Save and then Click Run.

Total Records   Total Quantity  
5      36

Inventory: Owner Name	Inventory: Inventory Name	Quantity
Anirudh Mitta (5)	Inventory 5	4
	Inventory 1	5
	Inventory 2	5
	Inventory 4	15
	Inventory 3	7
<b>Total (5)</b>		<b>36</b>

Row Counts  Detail Rows  Subtotals  Grand Total

#### 49. Create a Dashboard

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Art Gallery Dashboard” and select folder that you have created Click on Create.
- Click on the report that you have created on Artists and click on that and click select.
- Select any style to represent the data in dashboard.
- Click on the report that you have created on Arts and click on that and click select.
- Select any style to represent the data in dashboard.
- Click on the report that you have created on Inventory and click on that and click select.
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run

As of Dec 19, 2022, 7:41 PM Viewing as Anirudh Mitta

New Artists Report

Record Count

Style

- Pencil Sketching
- Craft Design
- Human Portrait
- Painting

5

New Artists with Arts Report

5

New Inventories Report

36

## 10) Create a web application with objects to record details about staff, syllabus and activities of a department and provide a link to college website from any of the objects.

### Steps:

1. Launch your Salesforce Trailhead Playground by opening any module and Switch to Lightning Experience if you are currently in Salesforce Classic by clicking your picture in the right top corner and then click on “Switch to Lightning Experience”
2. Then go to Setup gear icon and click “Setup”.
3. Click on “Object Manager” and click “Create > Custom Object” to create new Custom Object.
4. Name the object “Staff”.
5. Allow Reports and Allow Search.
6. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”.
7. To create a Tab for the Object: Select any Tab Style for the object “Staff”. Click Next, Next, leave the defaults and save.
8. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
9. Add the following fields one after the other:
  - Field Label: ID (Length 10), Data Type: Text, provide an example ID as Help Text, make it as Required Field, don’t allow Duplicate Values, make it as Case Insensitive and Set this field as the unique record identifier from an external system.
  - Field Label: Branch, Data Type: Text, Make it as a required field.
  - Field Label: Salary, Data Type: Number, Make it as a required field.

The screenshot shows the Salesforce Setup interface with the URL [playful-raccoon-obm5m-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01l2w000002U8T/FieldsAndRelationships/view](https://playful-raccoon-obm5m-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01l2w000002U8T/FieldsAndRelationships/view). The page title is "SETUP > OBJECT MANAGER" and the object name is "Staff". On the left, there's a sidebar with links like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The main content area is titled "Fields & Relationships" and shows a table with 7 items, sorted by Field Label. The table has columns: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The data is as follows:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Branch	Branch_c	Text(50)		
Created By	CreatedById	Lookup(User)		
ID	ID_c	Text(10) (External ID) (Unique Case Insensitive)	✓	
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)	✓	
Salary	Salary_c	Number(15, 0)		
Staff Name	Name	Text(80)	✓	

Create one more object to store Syllabus details:

10. Name the Object “Syllabus”.
11. Allow Reports and Allow Search.
12. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”
13. Create a Tab for the Object.

14. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
15. Add the following fields one after the other:
- Field Label: Subject Code, Data Type: Text.
  - Field Label: Credits, Data Type: Number.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Credit	Credit_c	Number(1, 0)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)	✓	
Subject Code	Subject_Code_c	Text(6) (External ID) (Unique Case Insensitive)	✓	
Syllabus Name	Name	Text(80)	✓	

Create one more object to store department activities’ details:

16. Name the Object “Activities”.
  17. Allow Reports and Allow Search.
  18. Check the box in front of “Launch New Custom Tab Wizard after saving this custom object”
  19. Create a Tab for the Object.
  20. To add fields to the Object: Go to “Fields & Relationships” option of Student object and Click “New”.
  21. Add the following fields one after the other:
- Field Label: Details, Data Type: Text. Make it as a required field.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Activity Name	Name	Text(80)	✓	
Created By	CreatedById	Lookup(User)		
Details	Details_c	Text Area(255)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)	✓	

## 22. Validation rules

### a. Staff ID Validation:

- To add a rule to the Staff is so that it should take only valid ids:
- Go to Validation Rule of Staff Object and click “New”
- Name it as “Staff id validation”.
- Error Condition Formula: NOT(BEGINS(ID\_\_c,’STAFF’)).
- Error Message: Please enter a valid staff Id.
- Error Location: Field – ID.
- Click Save.

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Hands-On Orgs', 'Object Manager | Salesforce', and a search bar. Below the navigation is a toolbar with icons for Home, Object Manager, and various setup functions. The main content area is titled 'Object Manager' and shows a 'Staff Validation Rule'. The rule details are as follows:

Validation Rule Detail	
Rule Name	Staff_ID_validation
Error Condition Formula	NOT(BEGINS(ID__c,’STAFF’))
Error Message	Please enter a valid staff Id.
Description	
Created By	Anirudh.Mitta, 12/21/2022, 4:12 AM
Active	<input checked="" type="checkbox"/>
Error Location	ID
Modified By	Anirudh.Mitta, 12/21/2022, 4:12 AM

### b. Subject Code Validation

- To add a rule to the Subject code so that it should not take invalid sub code:
- Go to Validation Rule of Syllabus Object and click “New”.
- Name it as “Sub code validation”
- Error Condition Formula: NOT(REGEX(Subject\_Code\_\_c, “\*[a-zA-Z]+[0-9+]”)).
- Error Message: Please Enter a valid Subject code.
- Error Location: Field – Subject Code.
- Click Save.

**Syllabus Validation Rule**

**Validation Rule Detail**

Rule Name	Subject_Code_Validation	Active	
Error Condition Formula	NOT(REGEX(Subject_Code__c, "[a-zA-Z][0-9]+"))	<input checked="" type="checkbox"/>	
Error Message	Enter a valid subject code	Error Location	Subject Code
Description		Created By	Anirudh Mitta, 12/21/2022, 4:15 AM
Created By	Anirudh Mitta, 12/21/2022, 4:15 AM	Modified By	Anirudh Mitta, 12/21/2022, 4:15 AM

### c. Credits validation

- To add a rule to the Credits so that it Should not take more than 4 credits:
- Go to Validation Rule of Syllabus Object and click “New”.
- Name it as “Credits validation”.
- Error Condition Formula: OR(credits\_\_c >4 ,Credits\_\_c <=0).
- Error Message: Please Enter the credits which is less than 4.
- Error Location: Field –Credit.
- Click Save.

**Syllabus Validation Rule**

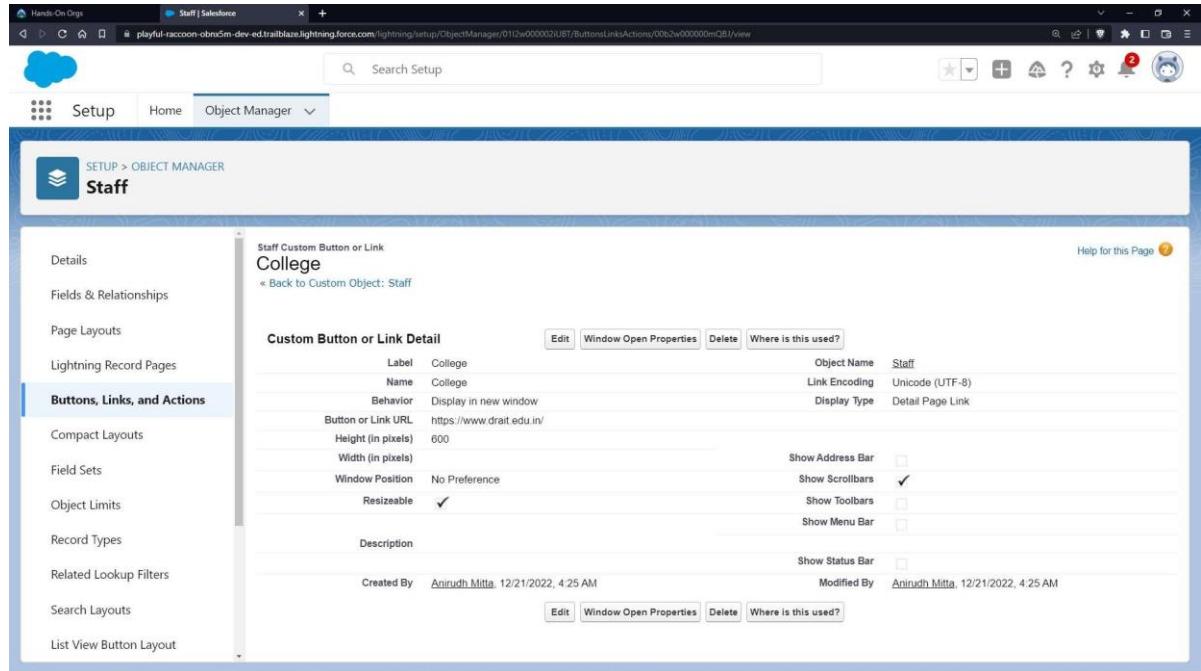
**Validation Rule Detail**

Rule Name	Credit_Validation	Active	
Error Condition Formula	OR( Credit__c >4 , Credit__c <=0)	<input checked="" type="checkbox"/>	
Error Message	Enter credits between 1 and 4	Error Location	Credit
Description		Created By	Anirudh Mitta, 12/21/2022, 4:19 AM
Created By	Anirudh Mitta, 12/21/2022, 4:19 AM	Modified By	Anirudh Mitta, 12/21/2022, 4:19 AM

To give a link to college website:

- Go to “Buttons, Links and Actions” of Art Object and click “New Button or Link”.
- Name it as “College”. Select the radio button “Detail Page Link” as it is a website link.
- Behaviour: Display in new window.

26. Content Source: URL.
27. Field Type: College.
28. In the empty space provided, type <http://www.drait.edu.in/>
29. Link Encoding: Unicode (UTF-8).
30. Click Save.



31. Go to Page Layout, Click Activities Layout.
32. Click Custom Links, Drag and drop the “College” link in the Custom Link area.
33. Click Save.

To create an application:

34. Go to “Setup” and type “App Manager” in Quick Find Box.
35. Click on “New Lightning App” to create a Lightning Application.
36. Name it as “Department Details”, give the description for your application.
37. Uploading Image and changing colours are optional, then click Next.
38. Navigation Style: Standard Navigation, click Next.
39. No need to add any Utility Bar, click Next.
40. Add the following:
  41. Items: Staff, Syllabuses, Activities, Reports and Dashboards, click Next.
  42. Assign it to System Administrator Profile by selecting System Administrator and pressing right arrow and then click Save & Finish.
  43. Go to App Manager, select your application and select Faculties and click “New” to add some details to your application.
  44. Click the entry you added, go to details.
  45. Press the “College” link to check the details.
  46. Click OK so that it will redirect you to the website.

Make Sure You will get an error when u give invalid staff id, credits and Subject code.

#### 47. Create a Staff Report

- Go to “Reports tab” Click on “New Folder” And give it any name then select Save.
- Click on “New Report” and from search bar Search for “Staffs” and then select it then select Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Staffs Report” and then select the folder which you have created.
- Click Save and then Click Run.

Branch	Staff: Staff Name	Salary
CSE (2)	Vasistha	100,000
	Vyasa	80,000
ECE (1)	Bhrigu	100,000
ME (2)	Agastya	80,000
	Viswamitra	80,000

#### 48. Create a syllabus report

- Go to “Reports tab” Click on “New Folder” And give it any name then select Save.
- Click on “New Report” and from search bar Search for “Syllabus” and then select it then select Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Syllabus Report” and then select the folder which you have created.
- Click Save and then Click Run

The screenshot shows a Salesforce Lightning interface with a report titled "Report: Syllabus New Syllabus Report". The report displays two records:

	Syllabus: Syllabus Name	Credit
1	Android Programming	4
2	Cloud Computing	4

#### 49. Create an activity report

- Go to “Reports tab” Click on “New Folder” And give it any name then select Save.
- Click on “New Report” and from search bar Search for “Activities” and then select it then select Continue.
- Add the required Columns to get the Complete Entered data.
- If you want the report to be grouped by any specific Fields, then Search for the field in “Add groups” otherwise it is optional.
- Click on save and name the report as “New Activities Report” and then select the folder which you have created.
- Click Save and then Click Run.

Activity: Owner Name	Activity: Activity Name	Details
Anirudh Mitta (3)	Experiment 2	The details of experiment 2
	Activity 3	The details of activity 3
	Experiment 1	The details of experiment 1

## 50. Create a dashboard

- Go to “Dashboard tab” and then click on “New Folder” and give it any Name.
- Click on “New Dashboard” and then name it as “Staff dashboard” and select folder that you have created Click on Create.
- Click on the report that you have created on staff and click on that and click select.
- Select any style to represent the data in dashboard.
- Click on the report that you have created on syllabus and click on that and click select.
- Select any style to represent the data in dashboard.
- Click on the report that you have created on activity and click on that and click select.
- Select any style to represent the data in dashboard.
- Add any filter(s), otherwise it is optional.
- Click on Save and Click Run.

**New Staff Report**

Sum of Salary

CSE	180k
ECE	100k
ME	160k

[View Report \(New Staff Report\)](#)

**New Activities Report**

View Report (New Activities Rep...)

**New Syllabus Report**

Syllabus: Syllabus Name	Credit
Android Programming	4
Cloud Computing	4

[View Report \(New Syllabus Report\)](#)

# CLOUD ANALYST

## Cloud Analyst

Cloud Analyst is a tool developed at the University of Melbourne whose goal is to support evaluation of social networks tools according to geographic distribution of users and data centers. In this tool, communities of users and data centers supporting the social networks are characterized and, based on their location; parameters such as user experience while using the social network application and load on the data centre are obtained/logged.

Cloud Analyst is developed by Bhathiya Wickremasinghe et al. at the CLOUDS Laboratory. It is built on top of CloudSim and separates the simulation experimentation from a programming task enabling one to concentrate on the simulation parameters rather than the technicalities of programming. Simulation in Cloud Analyst involves the following steps:

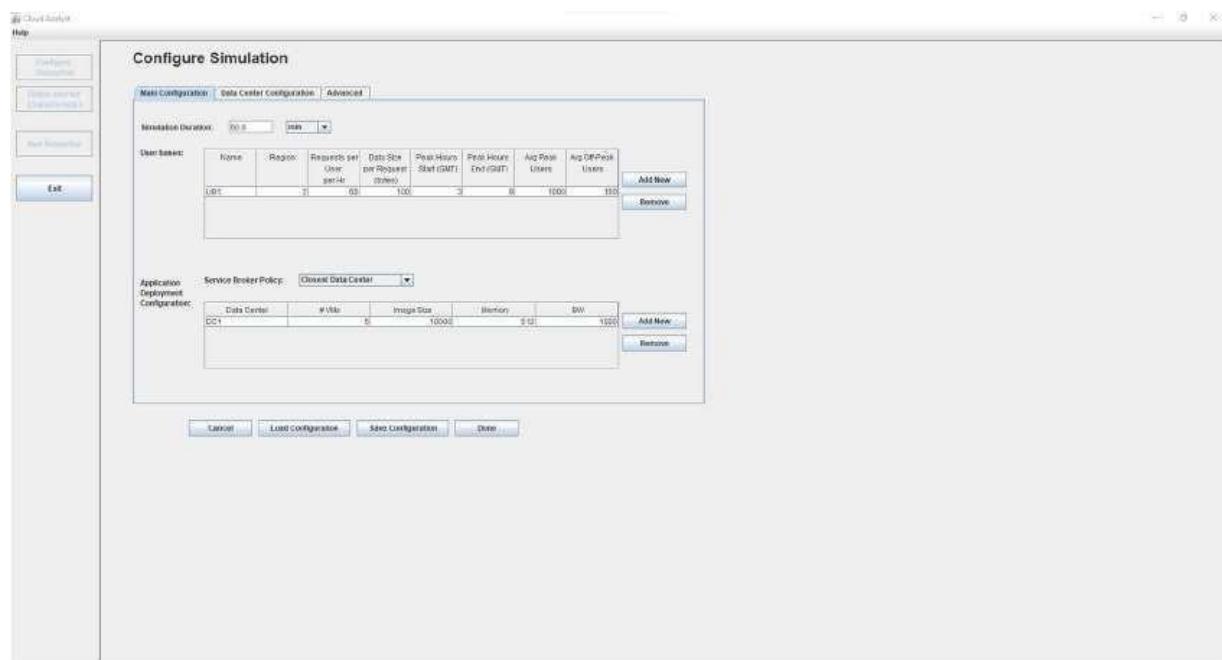
- i. Defining and configuration of User Bases.
- ii. Defining and configuring Data Centers
- iii. Allocating of Virtual Machines in Data Centers.
- iv. Review and Adjustment of various other parameters such as Packet size, Number of packets, Bandwidth, and Load balancing policies.



The Cloud Analyst enables us to model different scenarios of CSPs and User Bases, and provides a comprehensive output detailing the response time, Data Center processing time and total cost involved in the communication and computation.

#### Installing and Running Cloud Analyst:

1. Download Cloud Analyst
2. Extract the files from the zip file which will give following folder structure.
3. Click on run.bat file.
4. The user can then configure the simulation which includes
  - Simulation Duration
  - Number of User Bases
  - Service broker policy
  - Data Center Configuration
  - Load Balancing Policy
5. To run the simulation, click on Run Simulation.



**Configure Simulation**

Main Configuration | Data Center Configuration | Advanced

Data Center	Name	Region	Arch	GB	VMs	Cost per VM/hr	Memory Cost/hr	Storage Cost/hr	Data Transfer Cost/Gb	Physical HW Units
DC1	QX2	Latex	Van	8	3.1	0.02	0.1	0.1	0.1	2

Add New | Remove

**Cancel | Load Configuration | Save Configuration | Done**

**Configure Simulation**

Main Configuration | Data Center Configuration | Advanced

User grouping factor in Data Center:  
(Equivalent to number of simultaneous users from a single monitor)

Request processing factor in Data Center:  
(Equivalent to number of simultaneous requests a single application server instance can support)

Executable instruction length per request:  
(8000)

Load balancing policy across VMs in a single Data Center:  
Round Robin

**Cancel | Load Configuration | Save Configuration | Done**

# CLOUD ANALYST SIMULATIONS

1)

Use the following userbase configuration to simulate following scenarios for the given data centre and virtual machine configuration and answer to the following questions.

**Scenario-1:** Nearest data center with round robin policies

**Scenario-2:** Optimize response time with round robin policies

User base	Region	Data center	Peak-hour users	Off-peak hour users	Virtual machines
UB1	North America	--	1000	500	DC1-50
UB2	South America	--	800	1200	
UB3	Europe	DC1	2000	1000	
UB4	Africa	--	500	300	
UB5	Asia		3000	300	
UB6	Ocenia		1500	150	

- i) Tabulate the overall response time of all the scenarios and plot a line graph
- ii) Plot a bar graph for the data processing time of all the scenarios
- iii) Compare average response time by regions of all scenarios by plotting line graph
- iv) Using Pie chart show the total cost spent for each scenario

2)

Simulate the following scenarios for the given userbase, data centre and virtual machine configuration and answer to the given questions

Scenario	Scenario Description	Load Balancing algorithm	Service broker policy
1	One data center with 50 Virtual Machines for UB1	Nearest Data Centre	Round robin
2	Two data centers with 25 and 50 Virtual Machines respectively for UB1		
3	Three data centers with 100,75 and 25 Virtual Machines respectively for UB1		

- i) Tabulate the overall response time and data processing of all the scenarios and plot the bar graph
- ii) Plot a line graph of data center request servicing time of all the data centers for all the scenarios
- iii) Compare average response time by regions of all scenarios by plotting line graph
- iv) Mention the data centers used by the UB2, UB3, UB4 and UB5

3)

Simulate the following scenarios for given data centre, data centre and virtual machine configuration and answer the following questions

**Scenario 1:** closest data center and round robin policies

**Scenario 2:** optimize response time and round robin policies

Use the following userbase configuration for all the scenarios

User base	Region	Data center	Peak-hour users	Off-peak hour users	Virtual machines
UB1	North America	DC1, DC3	1000	500	DC1-50 DC3-100
UB2	South America	---	800	1200	
UB3	Europe	DC4	2000	1000	DC4-150
UB4	Africa	--	500	300	

- i) Tabulate and compare the Average response time and data processing time of all the scenarios by plotting the line graph
- ii) Tabulate the response time of user bases in all scenarios and compare these by plotting bar graph. Which user base is taking maximum time among three scenarios? Why
- iii) Calculate the data transmission time from DC1 to UB2
- iv) Plot the bar graph for data center cost of all scenarios

5)

Analyze the various load balancing algorithms for the given userbase, data centre and virtual machine configuration and answer the following questions. Consider the following userbase configuration for all load balancing algorithms

Number of User bases	06
Region for the userbases	UB1-South America, UB2-Asia, UB3-North America, UB4-Europe, UB5-Africa, UB6-Ocenia
Average peak users for all the user bases	10000
Average off-peak users for all the user bases	100
Peak hours' time	Depends on the region
Data centers in each user base	UB1-1, UB2-2, UB3-1, UB4-3, UB5-2, UB6-1
Virtual machines in each data center	6
Simulation time	10 mins
Service broker policy	Nearest data center

- a) Tabulate and compare the data processing time of load balancing algorithms by plotting the line graph
- b) Tabulate the response time of load balancing algorithms by plotting the bar graph
- c) Tabulate the response time by region for load balancing algorithms and plot bar graph
- d) Which load balancing algorithm is best and why?

4)

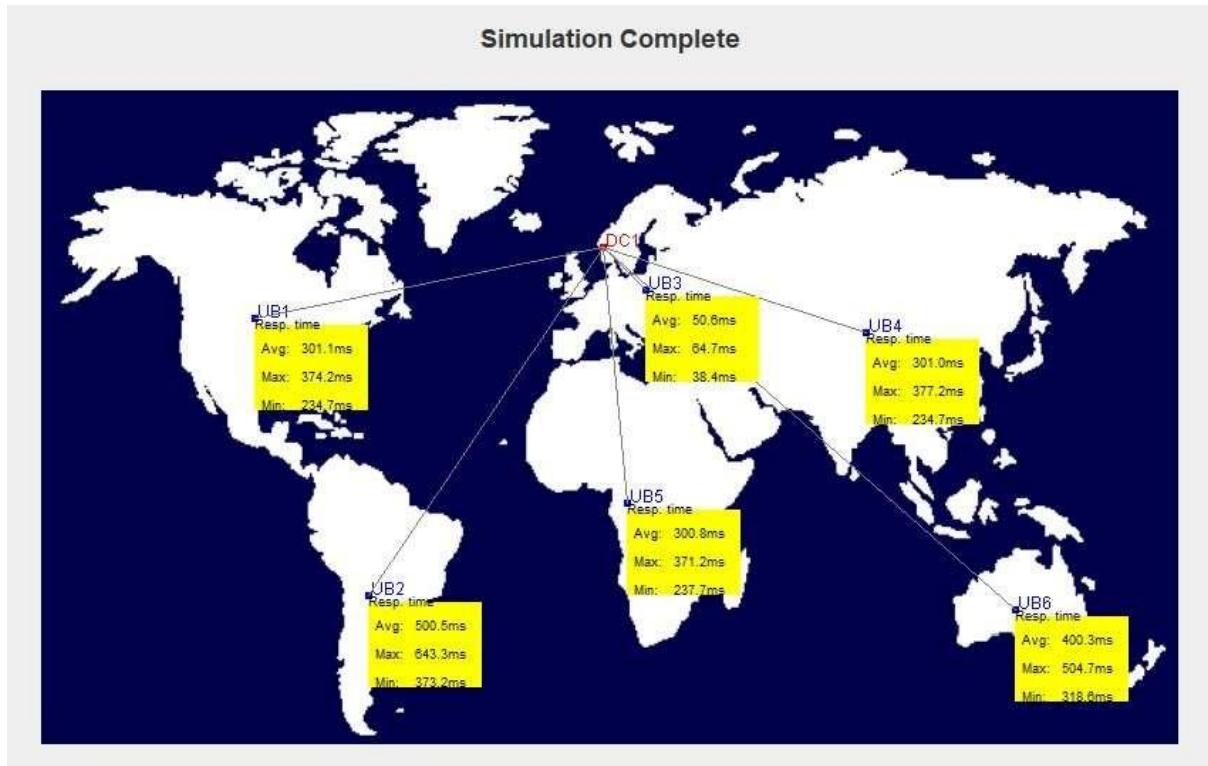
Analyze the various service broker policies for the following configuration and answer the following questions.

Parameter	Value Used
UB Name	UB1
Region	2
Request Per User Per Hour	60
Data Size Per Request	100
Peak hour start(GMT)	3
Peak hour end (GMT)	9
Avg Peak Users	40000
Avg Off Peak Users	4000
DC 1 – No Of VM	75
DC 2 – No Of VM	50
DC 3 – No Of VM	25
VM Image Size	10000 MB
VM Memory	512 MB
VM Bandwidth	1000 bps
DC 1 – No Of Physical Machine	2
DC 2 – No Of Physical Machine	2
DC 3 – No Of Physical Machine	2
DC – Memory Per Machine	204800 Mb
DC – Storage Per Machine	100000000 Mb
DC – Available BW Per Machine	1000000
DC – No Of Processors Per Machine	4
DC – Processor Speed	10000 MIPS
DC – VM Policy	Time Shared
User Grouping Factor	1000
Request Grouping Factor	100
Executable Instruction Length	500
Load Balancing Policy	Throttled

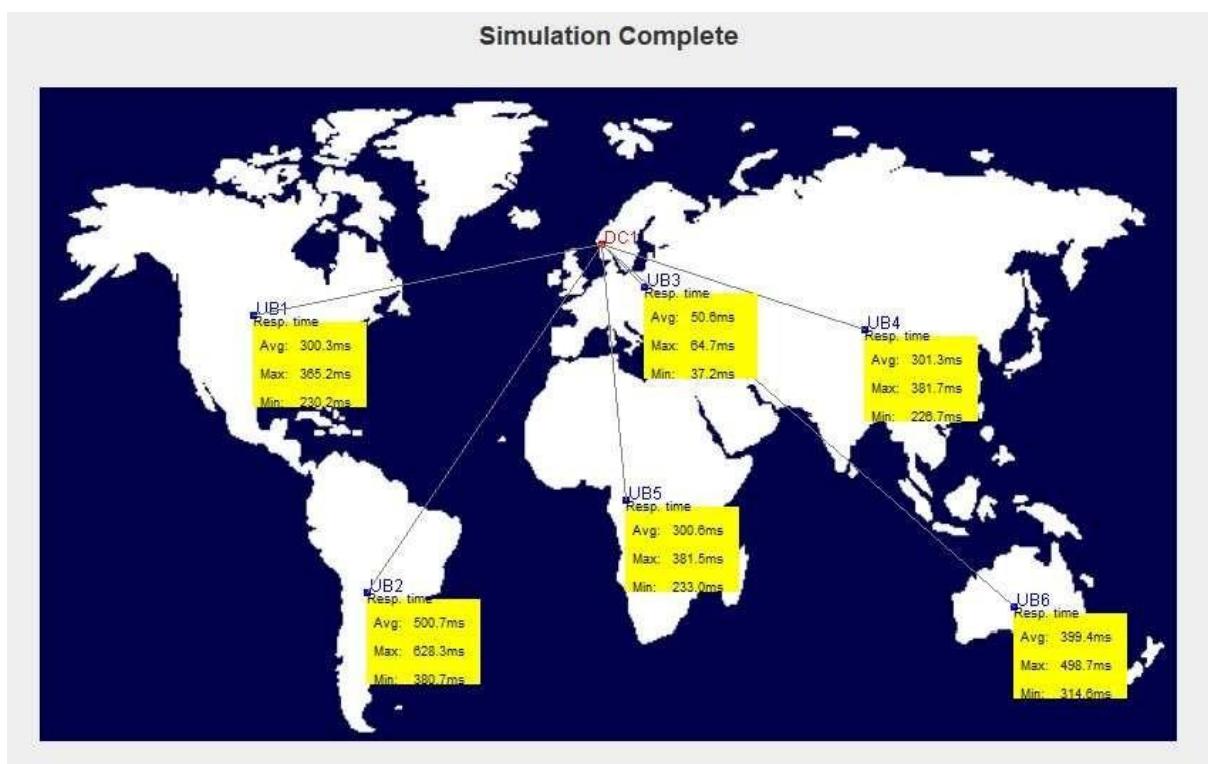
- a) Tabulate and compare the data processing time of service broker policies by plotting the line graph
- b) Tabulate and compare response time of service broker policies by plotting the bar graph
- c) Tabulate the cost for service broker policies and represent it using pie chart
- d) Which service broker policy is best and why?

1)

### Closest



### Optimized



Results of the Simulation Completed at: 29/11/2022 09:48:20

### Overall Response Time Summary

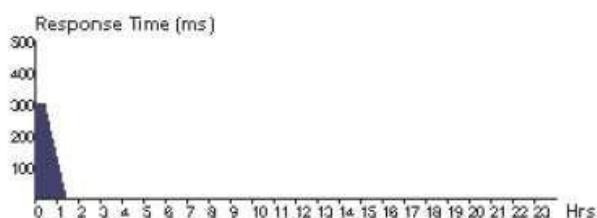
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	277.25	38.42	643.27
Data Center processing time:	0.92	0.07	1.43

### Response Time by Region

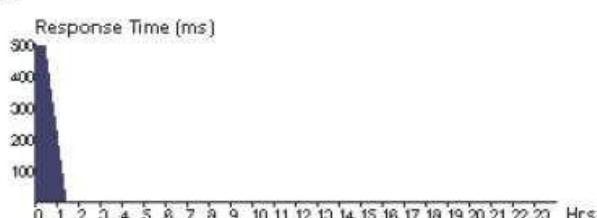
Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	301.08	234.68	374.23
UB2	500.50	373.24	643.27
UB3	50.57	38.42	64.68
UB4	300.97	234.66	377.17
UB5	300.85	237.69	371.18
UB6	400.31	318.64	504.65

### User Base Hourly Response Times

UB1

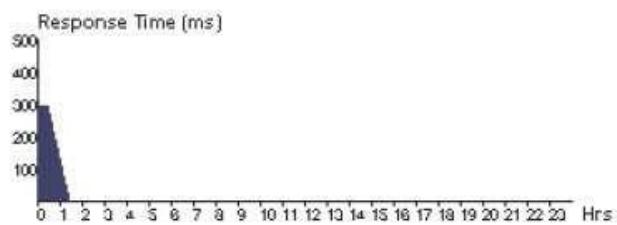
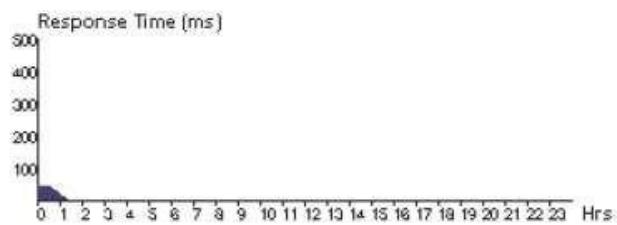


UB2

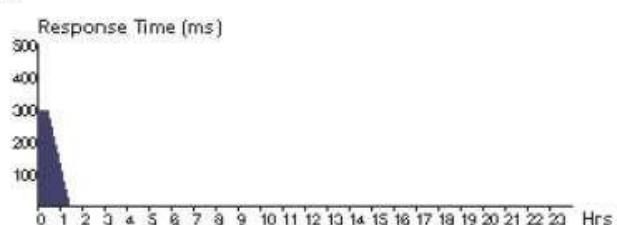


UB3

UB4



UB5



UB6



#### Data Center Request Servicing Times

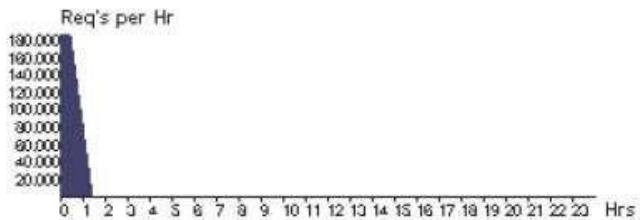
Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.92	0.07	1.43

#### Data Center Hourly Average Processing Times

DC1

#### Data Center Hourly Loading

### DC1



### Cost

Total Virtual Machine Cost (\$): 5.02

Total Data Transfer Cost (\$): 1.95

Grand Total: (\$) 6.97

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC1	5.02	1.95	6.97

Results of the Simulation Completed at: 29/11/2022 09:53:43

#### Overall Response Time Summary

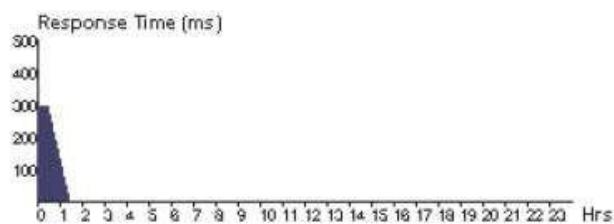
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	277.15	37.18	628.30
Data Center processing time:	0.92	0.07	1.43

#### Response Time by Region

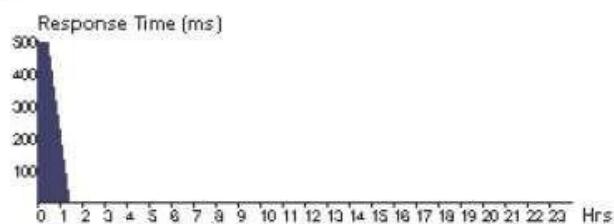
Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	300.29	230.17	365.17
UB2	500.65	380.71	628.30
UB3	50.62	37.18	64.66
UB4	301.33	226.74	381.69
UB5	300.63	233.00	381.55
UB6	399.44	314.64	498.65

#### User Base Hourly Response Times

UB1

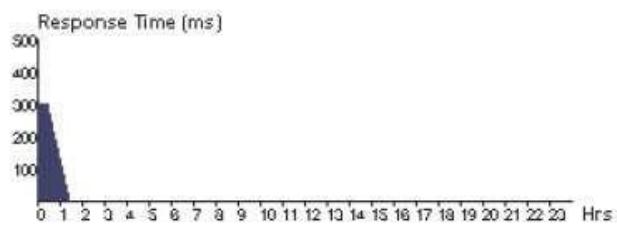
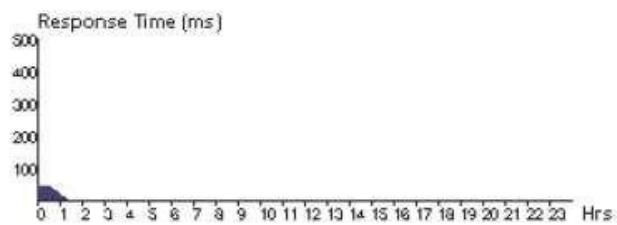


UB2

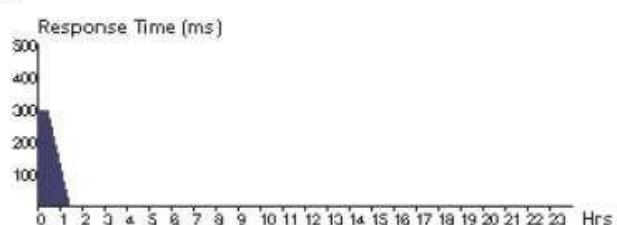


UB3

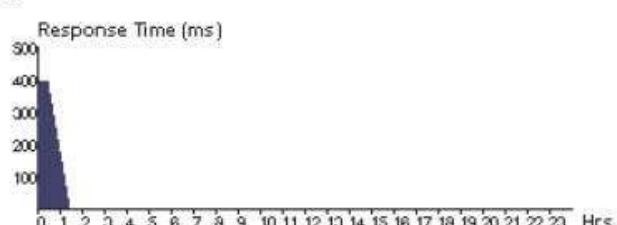
UB4



UB5



UB6



#### Data Center Request Servicing Times

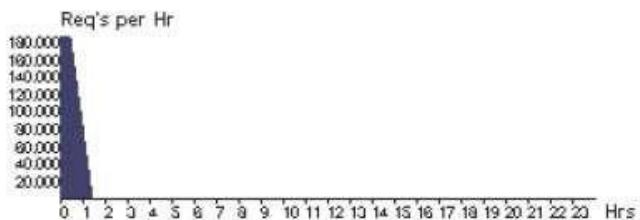
Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.92	0.07	1.43

#### Data Center Hourly Average Processing Times

DC1

#### Data Center Hourly Loading

### DC1



### Cost

Total Virtual Machine Cost (\$): 5.02

Total Data Transfer Cost (\$): 1.95

Grand Total: (\$) 6.97

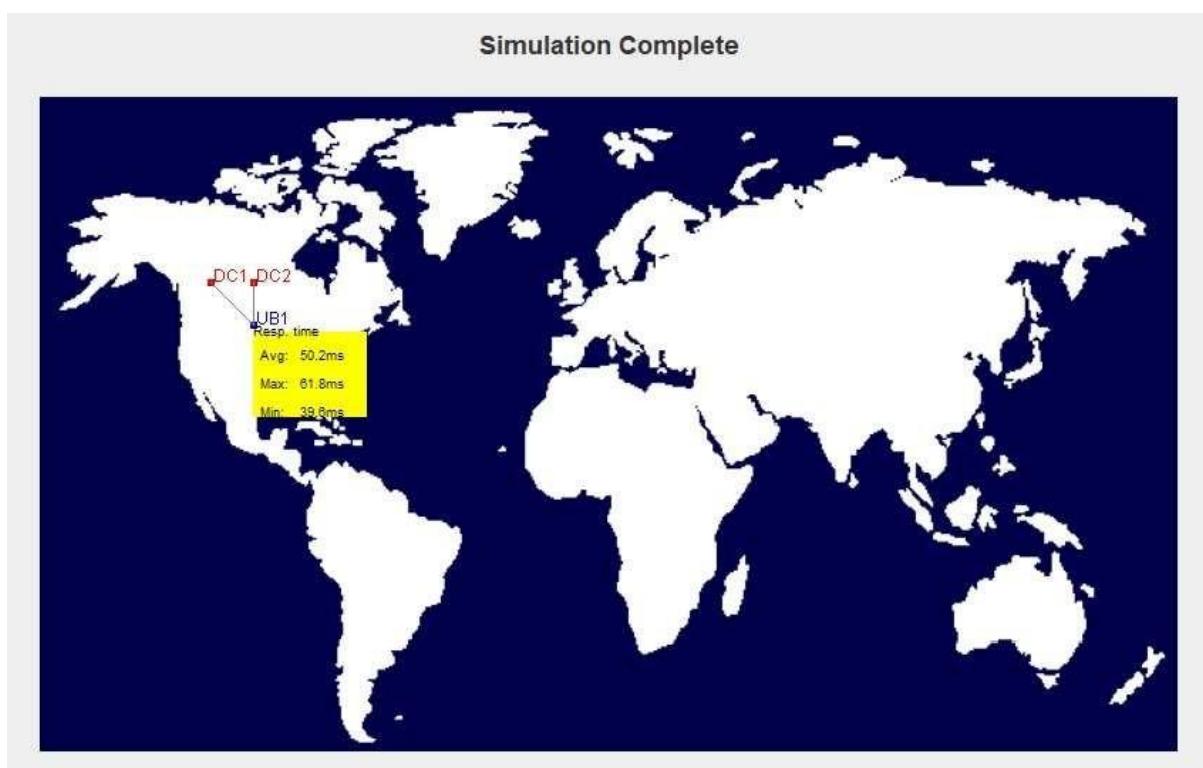
Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC1	5.02	1.95	6.97

2)

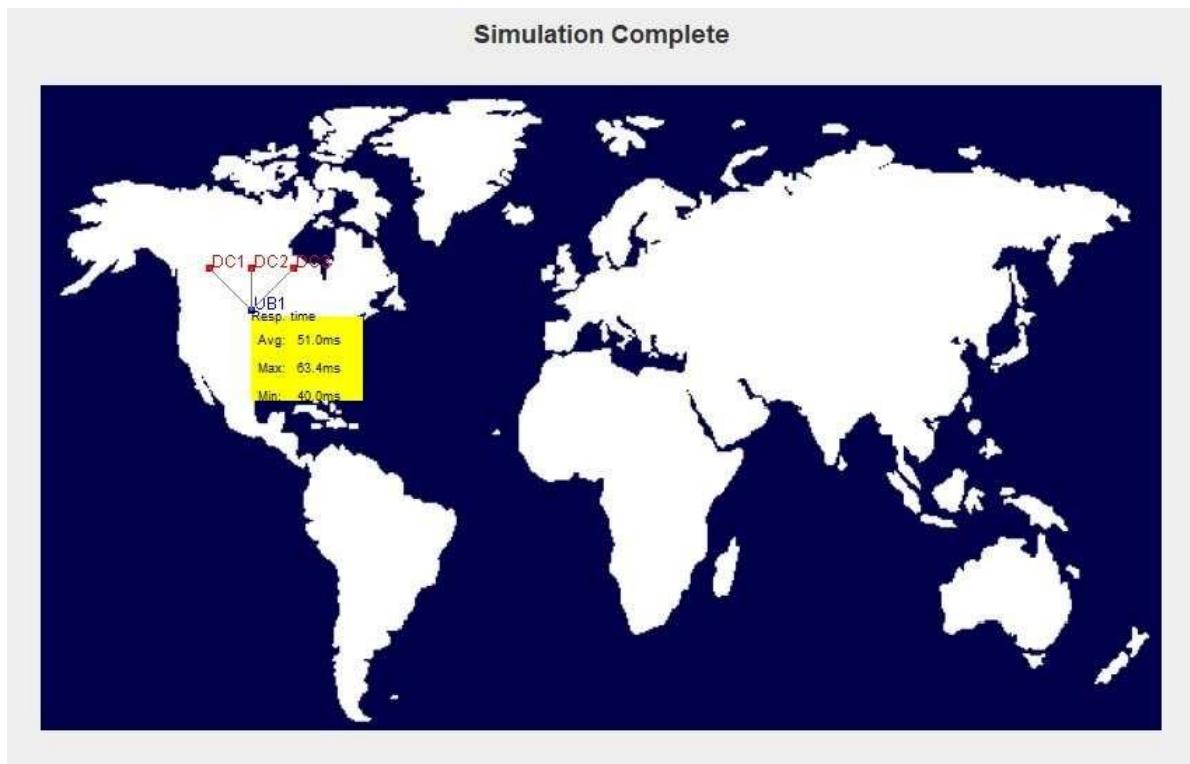
### Scenario 1



### Scenario 2



### Scenario 3



Results of the Simulation Completed at: 29/11/2022 09:58:12

#### Overall Response Time Summary

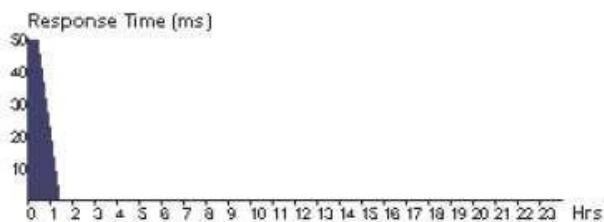
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	50.09	39.55	61.61
Data Center processing time:	0.48	0.01	0.86

#### Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	50.09	39.55	61.61

#### User Base Hourly Response Times

UB1



#### Data Center Request Servicing Times

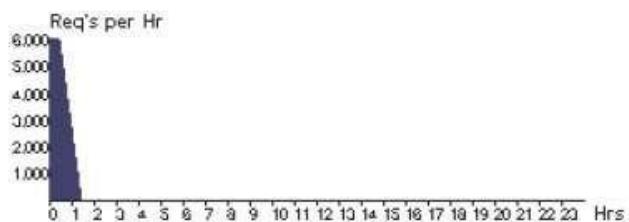
Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.48	0.01	0.86

#### Data Center Hourly Average Processing Times

DC1

## Data Center Hourly Loading

DC1



Cost

Total Virtual Machine Cost (\$): 0.51

Total Data Transfer Cost (\$): 0.06

Grand Total: (\$) 0.57

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC1	0.51	0.06	0.57

Results of the Simulation Completed at: 29/11/2022 10:08:23

### Overall Response Time Summary

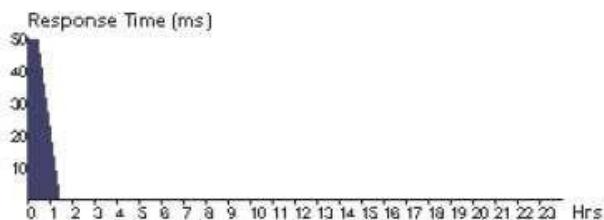
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	50.22	39.63	61.83
Data Center processing time:	0.60	0.02	1.08

### Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	50.22	39.63	61.83

### User Base Hourly Response Times

UB1



### Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.68	0.07	1.08
DC2	0.51	0.02	0.88

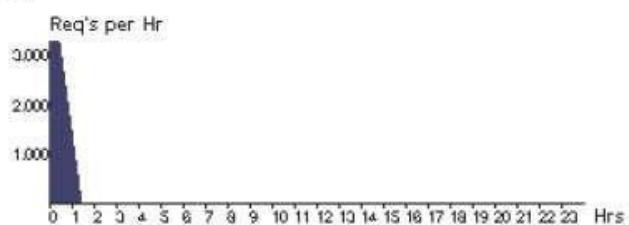
### Data Center Hourly Average Processing Times

DC1

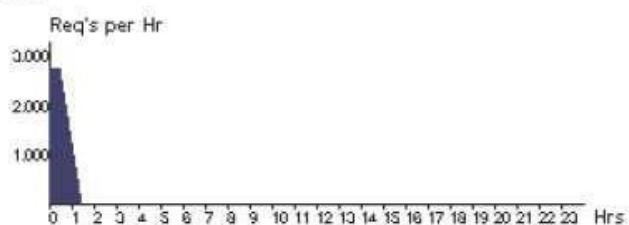
DC2

#### Data Center Hourly Loading

DC1



DC2



Cost

Total Virtual Machine Cost (\$): 3.04

Total Data Transfer Cost (\$): 0.06

Grand Total: (\$) 3.11

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC2	0.51	0.03	0.54
DC1	2.54	0.03	2.57

Results of the Simulation Completed at: 29/11/2022 10:11:21

### Overall Response Time Summary

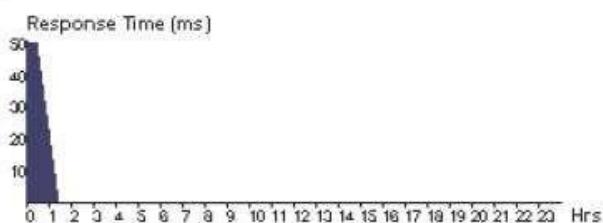
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	51.03	40.01	63.38
Data Center processing time:	1.41	0.03	2.63

### Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	51.03	40.01	63.38

### User Base Hourly Response Times

UB1



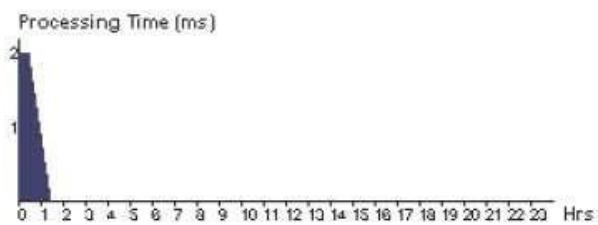
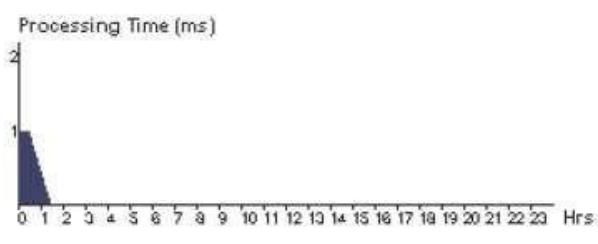
### Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	1.57	0.13	2.00
DC2	2.18	0.44	2.63
DC3	0.50	0.03	0.88

### Data Center Hourly Average Processing Times

DC1

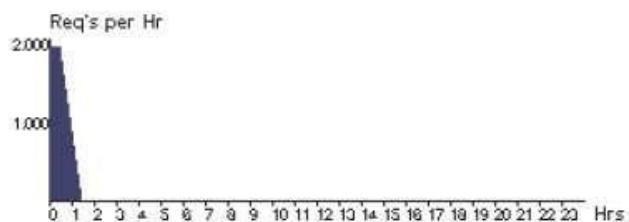
DC2



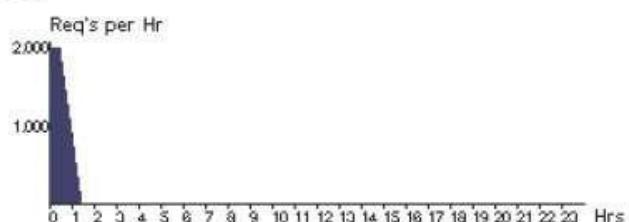
DC3

#### Data Center Hourly Loading

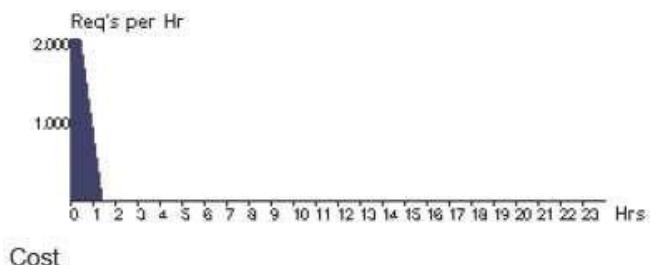
DC1



DC2



DC3



Total Virtual Machine Cost (\$): 18.25

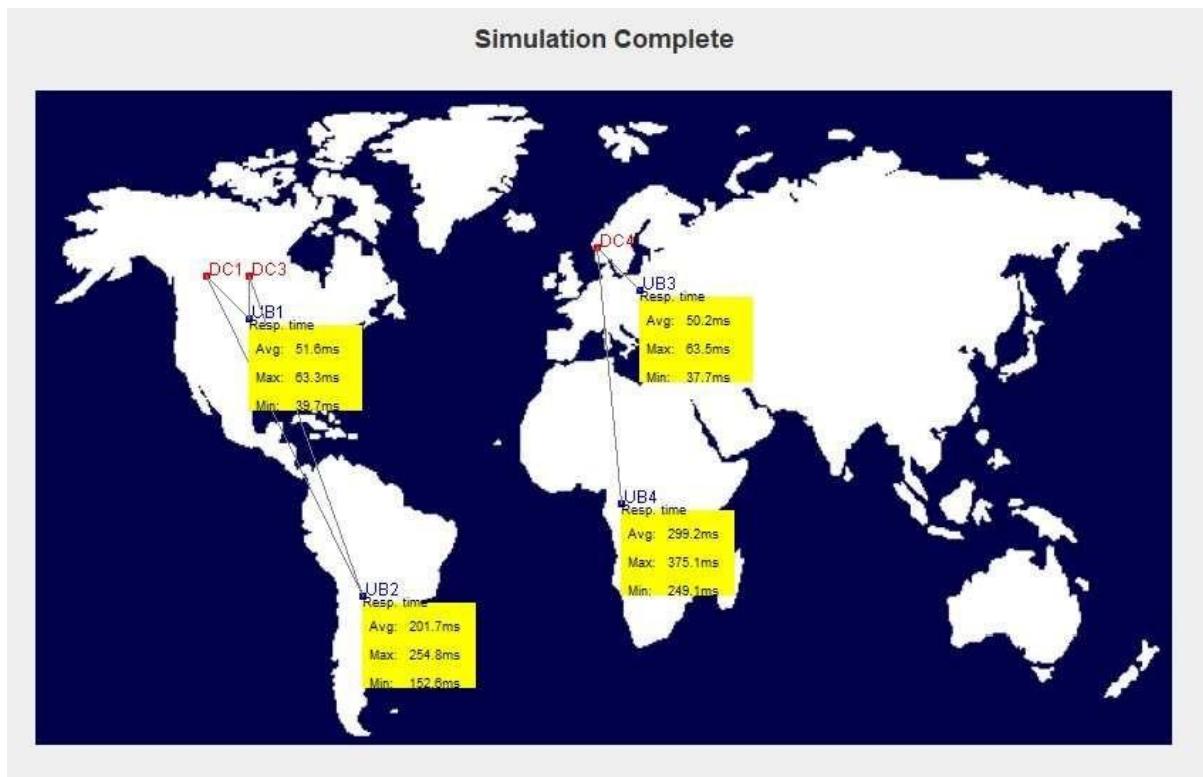
Total Data Transfer Cost (\$): 0.06

Grand Total: (\$) 18.32

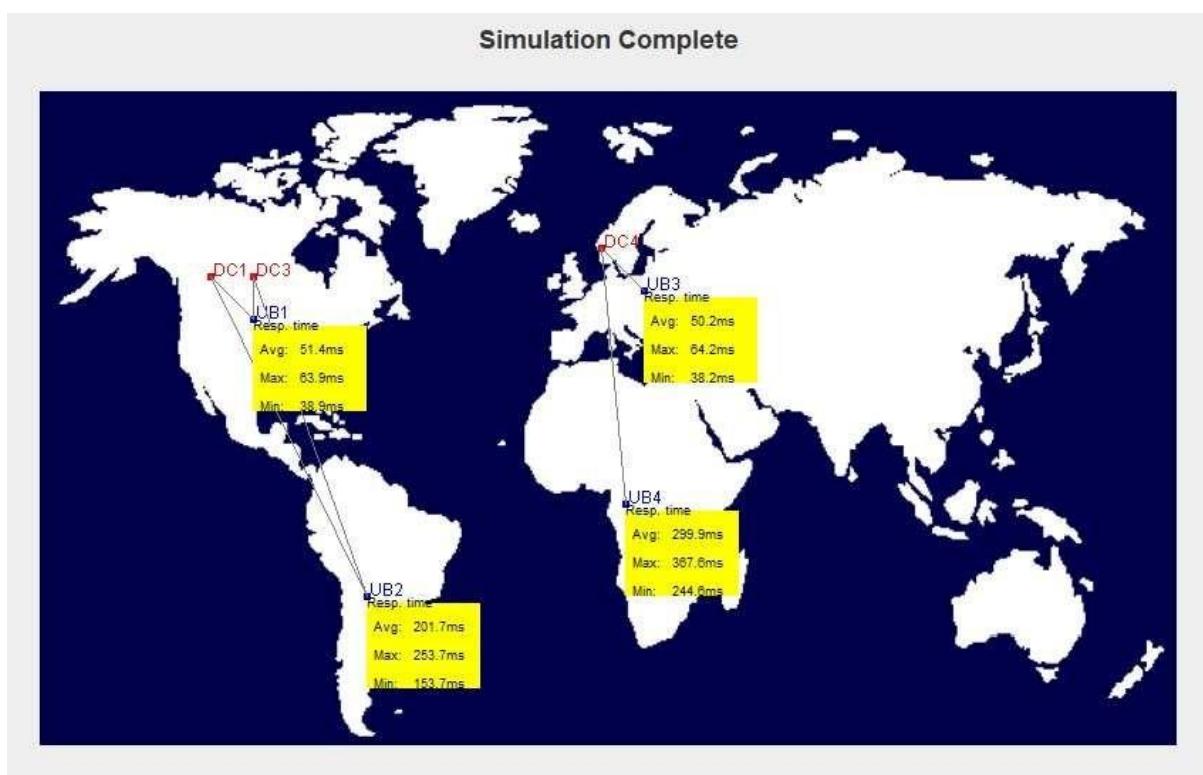
Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC2	7.61	0.02	7.63
DC1	10.14	0.02	10.16
DC3	0.51	0.02	0.53

3)

### Closest



### Optimized



Results of the Simulation Completed at: 29/11/2022 10:21:20

### Overall Response Time Summary

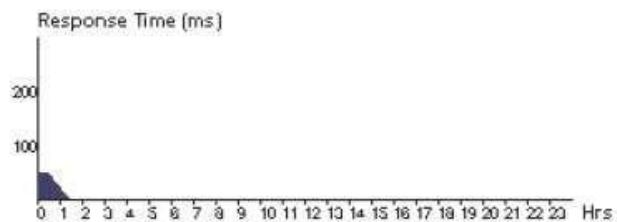
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	125.37	37.68	375.14
Data Center processing time:	1.24	0.02	3.28

### Response Time by Region

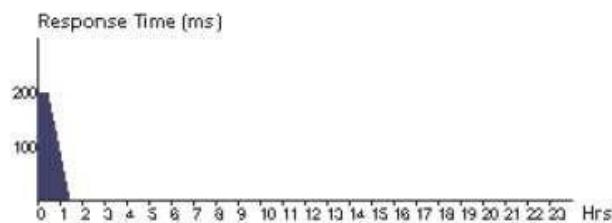
Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	51.56	39.66	63.29
UB2	201.66	152.60	254.76
UB3	50.15	37.68	63.45
UB4	299.18	249.14	375.14

### User Base Hourly Response Times

UB1

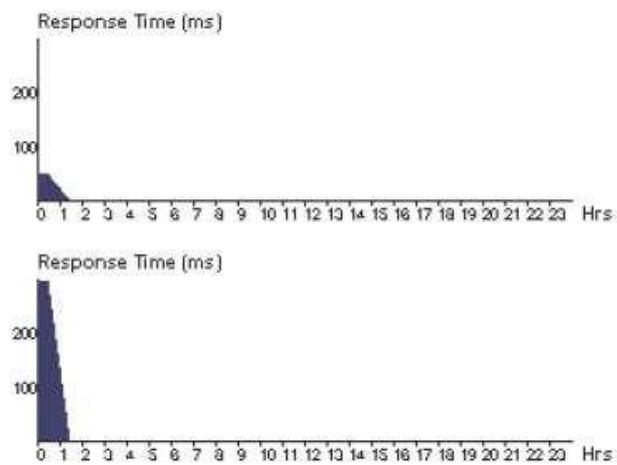


UB2



UB3

UB4



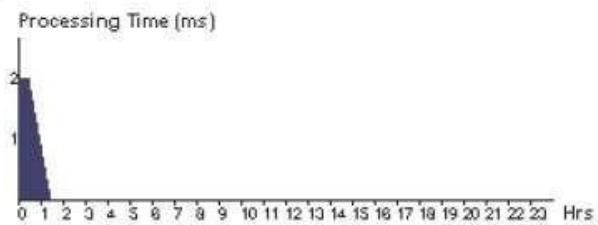
#### Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.77	0.24	1.40
DC3	2.65	0.59	3.28
DC4	0.51	0.02	1.26

#### Data Center Hourly Average Processing Times

DC1

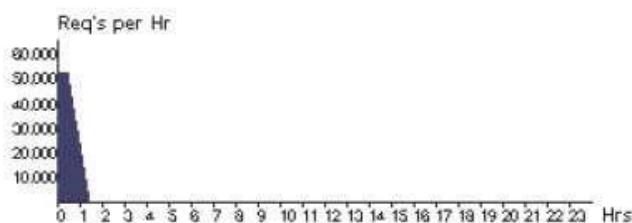
DC3



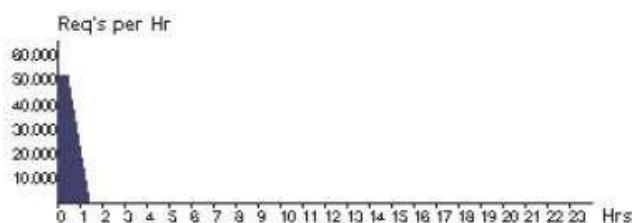
DC4

### Data Center Hourly Loading

DC1



DC3



DC4



### Cost

Total Virtual Machine Cost (\$): 15.56

Total Data Transfer Cost (\$): 1.78

Grand Total: (\$) 17.34

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC1	5.02	0.55	5.57

DC4	0.50	0.69	1.19
DC3	10.04	0.54	10.58

Results of the Simulation Completed at: 29/11/2022 10:25:46

### Overall Response Time Summary

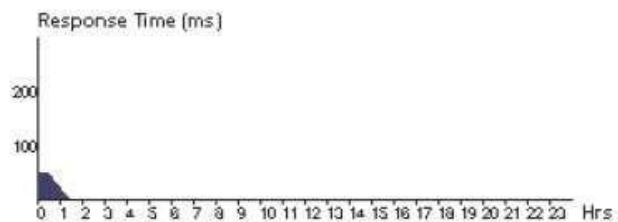
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	125.38	38.18	367.63
Data Center processing time:	1.19	0.02	3.28

### Response Time by Region

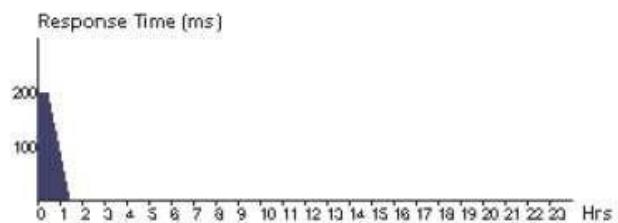
Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	51.37	38.92	63.92
UB2	201.68	153.66	253.72
UB3	50.17	38.18	64.19
UB4	299.93	244.64	367.63

### User Base Hourly Response Times

UB1

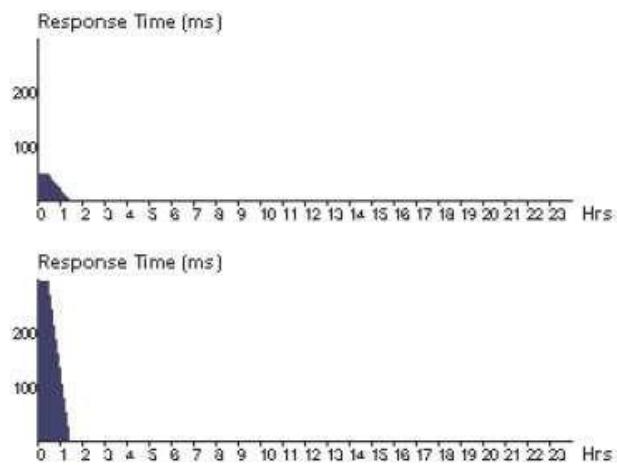


UB2



UB3

UB4



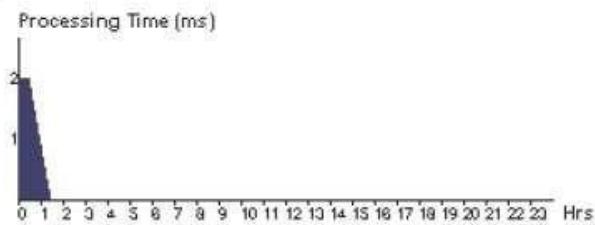
Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.78	0.22	1.41
DC3	2.64	0.61	3.28
DC4	0.51	0.02	1.08

Data Center Hourly Average Processing Times

DC1

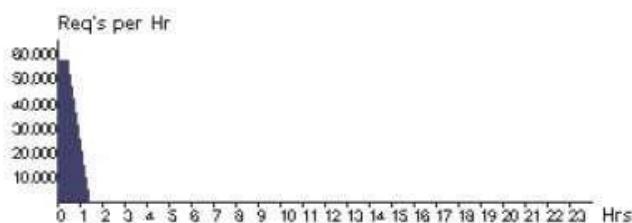
DC3



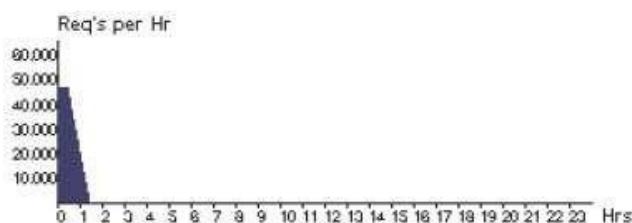
DC4

## Data Center Hourly Loading

DC1



DC3



DC4



## Cost

Total Virtual Machine Cost (\$): 15.56

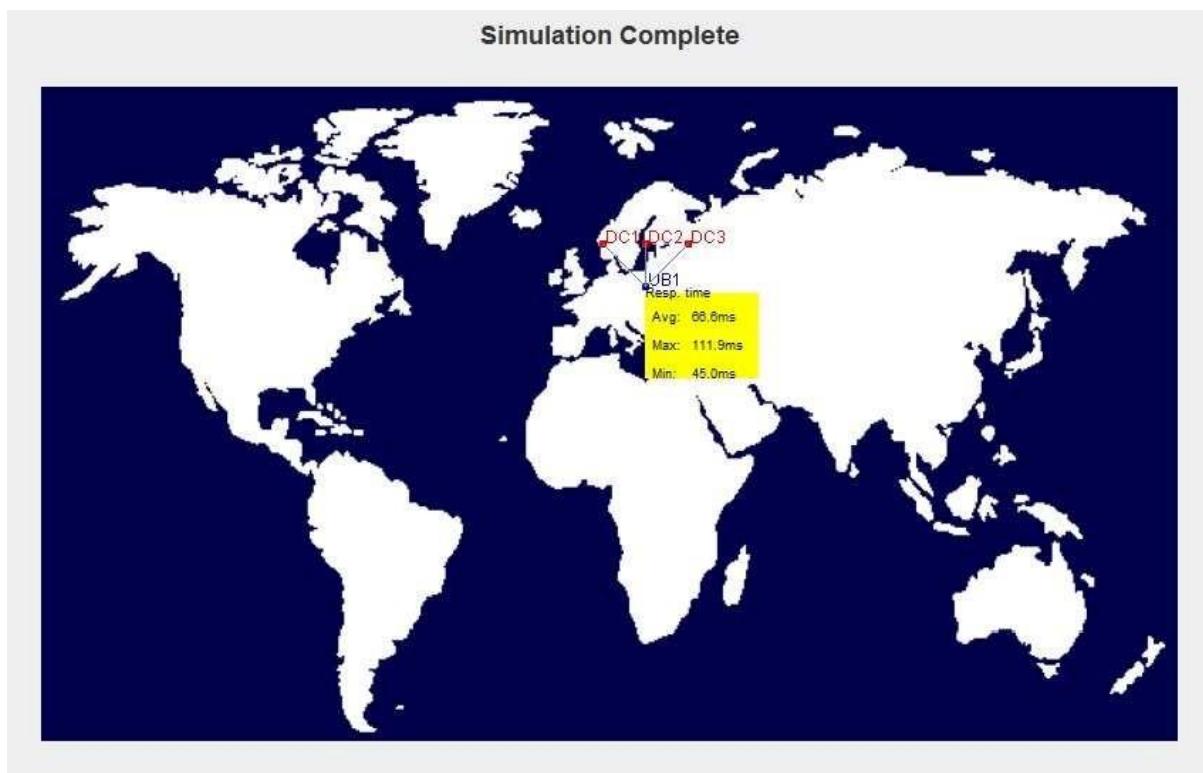
Total Data Transfer Cost (\$): 1.78

Grand Total: (\$) 17.34

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC1	5.02	0.60	5.62

DC4	0.50	0.69	1.19
DC3	10.04	0.49	10.53

4)



Results of the Simulation Completed at: 29/11/2022 10:38:39

### Overall Response Time Summary

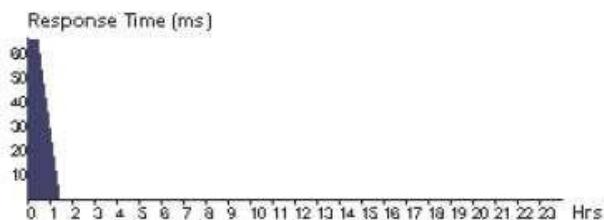
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	66.62	44.97	111.86
Data Center processing time:	17.01	0.05	47.53

### Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	66.62	44.97	111.86

### User Base Hourly Response Times

UB1



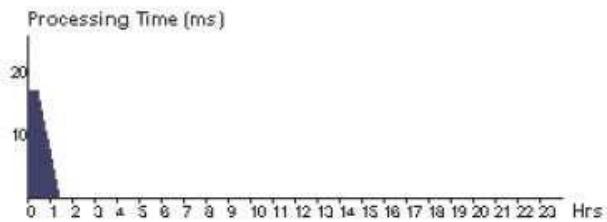
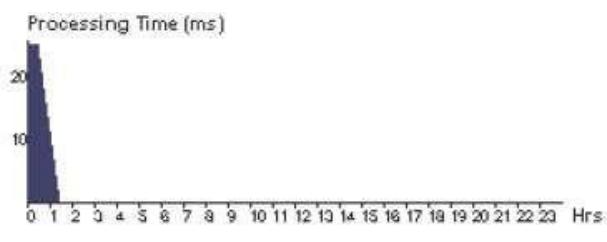
### Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	25.52	0.93	47.53
DC2	17.82	0.88	32.00
DC3	2.77	0.05	5.40

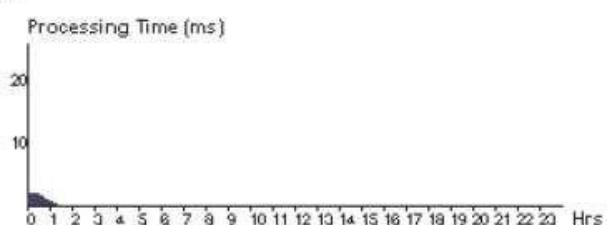
### Data Center Hourly Average Processing Times

DC1

DC2

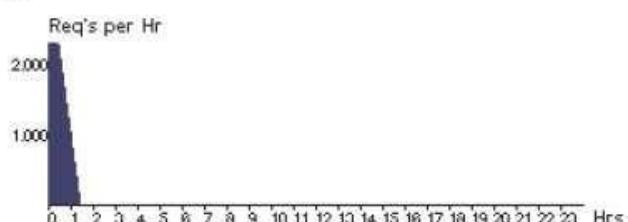


DC3

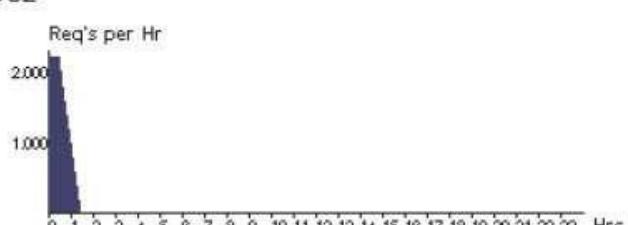


#### Data Center Hourly Loading

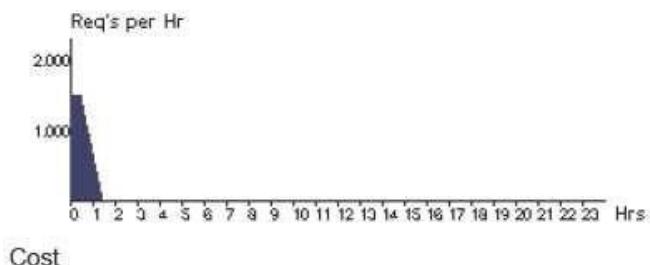
DC1



DC2



DC3



Cost

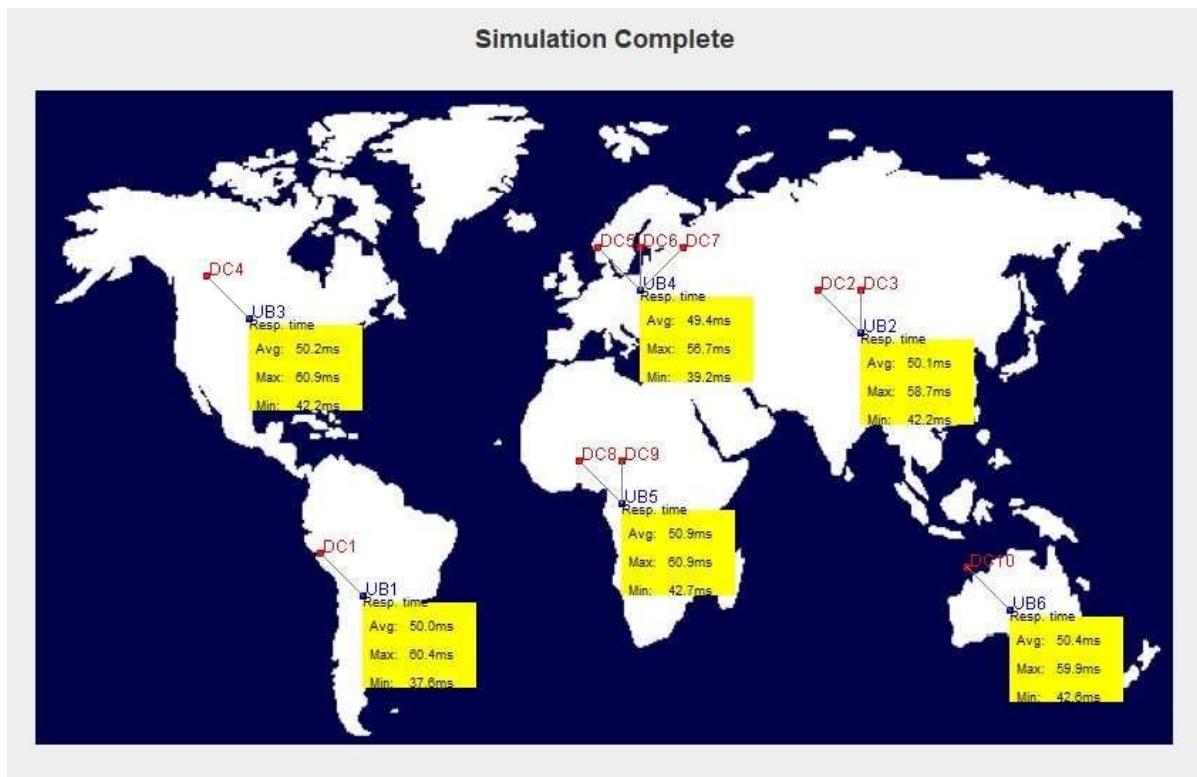
Total Virtual Machine Cost (\$): 13.18

Total Data Transfer Cost (\$): 0.06

Grand Total: (\$) 13.24

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC2	5.07	0.02	5.09
DC1	7.61	0.02	7.63
DC3	0.51	0.01	0.52

5)



Results of the Simulation Completed at: 29/11/2022 10:57:04

#### Overall Response Time Summary

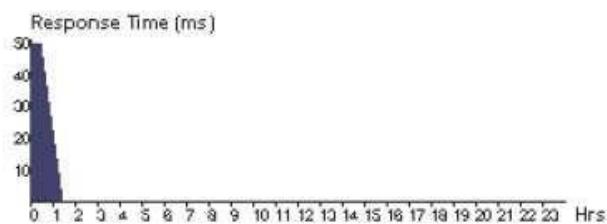
	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	50.16	37.62	60.93
Data Center processing time:	0.50	0.02	0.92

#### Response Time by Region

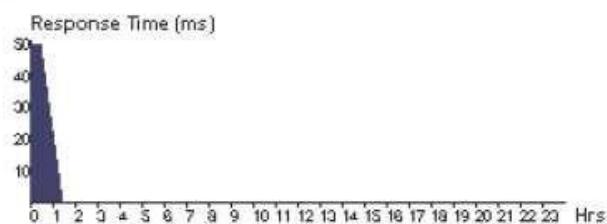
Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	50.02	37.62	60.37
UB2	50.06	42.16	58.66
UB3	50.24	42.16	60.91
UB4	49.43	39.16	56.66
UB5	50.90	42.68	60.93
UB6	50.37	42.61	59.91

#### User Base Hourly Response Times

UB1

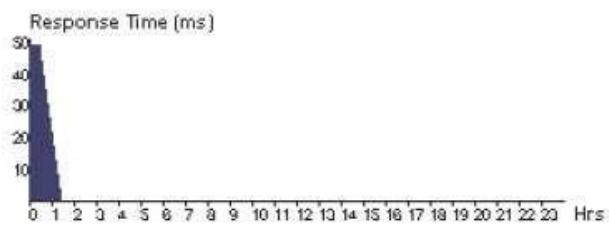
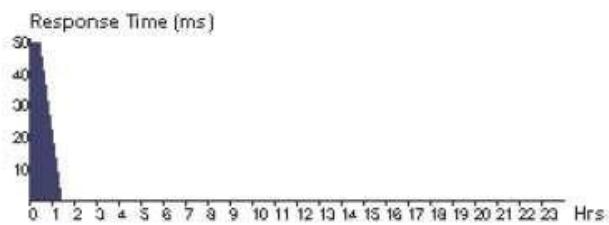


UB2

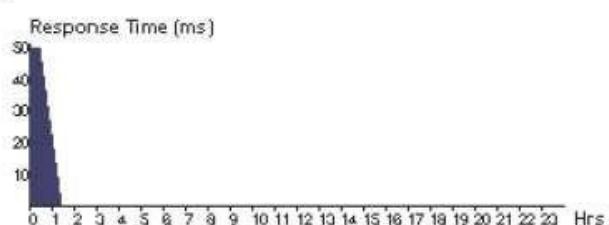


UB3

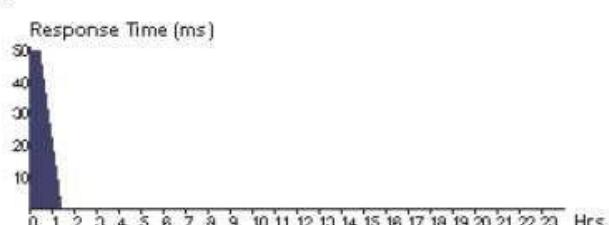
UB4



UB5

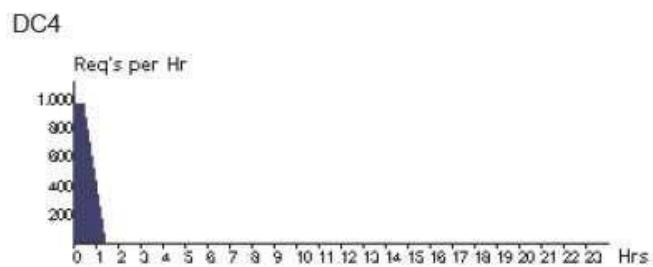
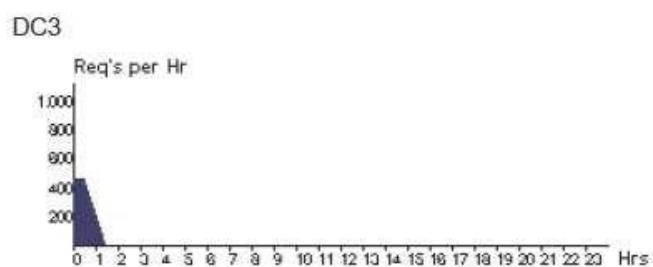
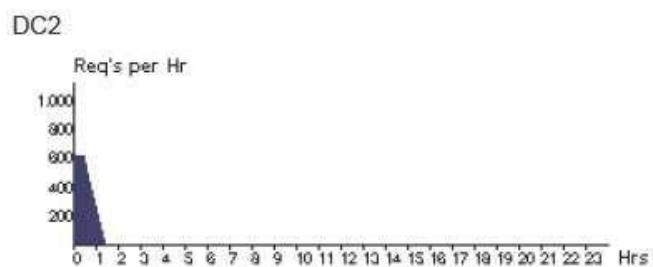
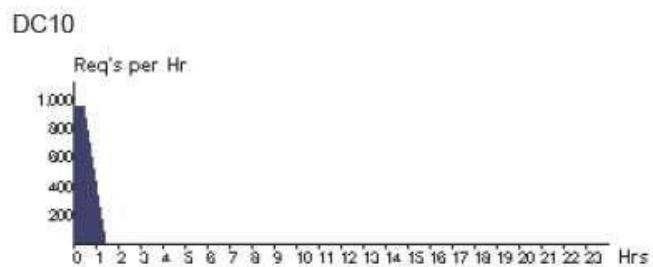
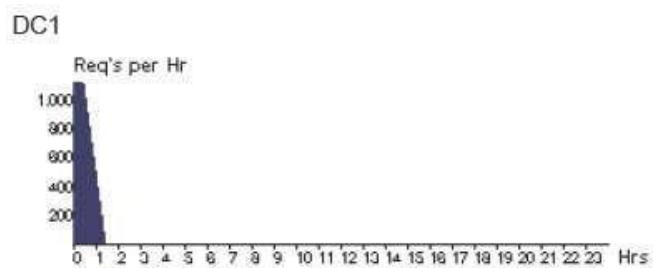


UB6



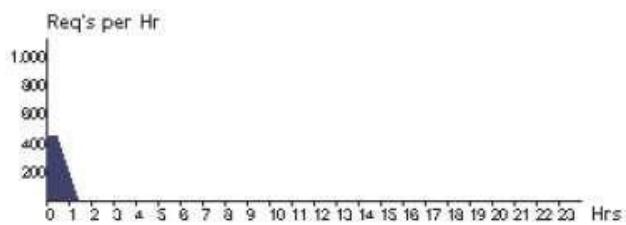
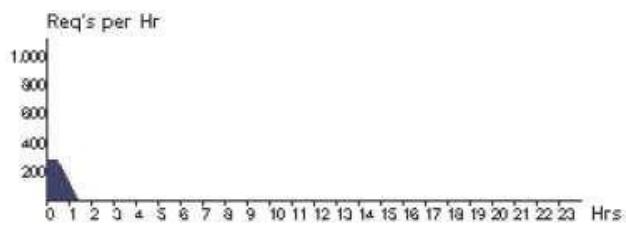
#### Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC10	0.51	0.11	0.90
DC1	0.42	0.03	0.86
DC2	0.48	0.08	0.91
DC3	0.59	0.11	0.91
DC4	0.52	0.02	0.91
DC5	0.52	0.07	0.88
DC6	0.45	0.09	0.90
DC7	0.50	0.03	0.90
DC8	0.52	0.03	0.92

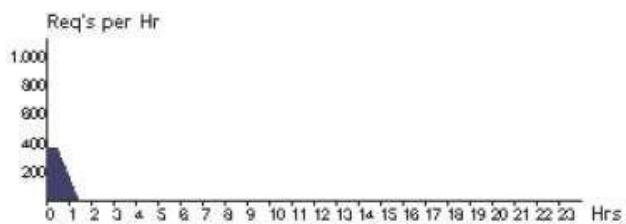


DC5

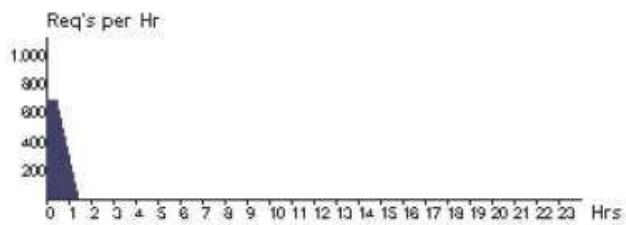
DC6



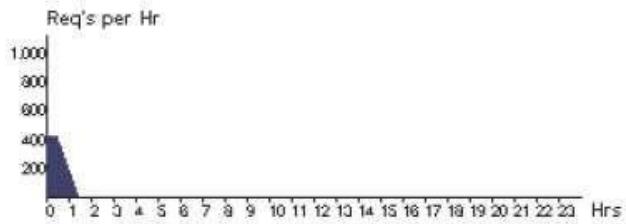
DC7



DC8



DC9



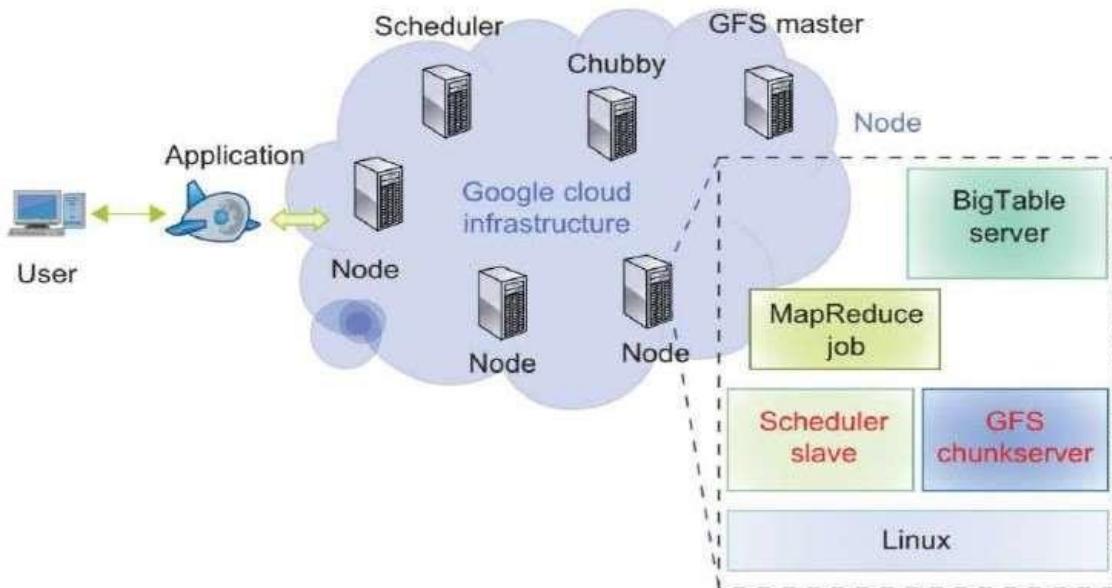
Cost

Total Virtual Machine Cost (\$):	0.99
Total Data Transfer Cost (\$):	0.07
Grand Total: (\$)	1.05

Data Center	VM Cost \$	Data Transfer Cost \$	Total \$
DC10	0.10	0.01	0.11
DC2	0.10	0.01	0.11
DC1	0.10	0.01	0.11
DC4	0.10	0.01	0.11
DC3	0.10	0.00	0.11
DC6	0.10	0.00	0.11
DC5	0.08	0.00	0.09
DC8	0.10	0.01	0.11
DC7	0.10	0.00	0.10
DC9	0.10	0.00	0.10

# GOOGLE APP ENGINE

Google App Engine (often referred to as GAE or simply App Engine) is a cloud computing platform as a service for developing and hosting web applications in Google-managed data centers. Applications are sandboxed and run across multiple servers. App Engine offers automatic scaling for web applications—as the number of requests increases for an application, App Engine automatically allocates more resources for the web application to handle the additional demand. The service is free up to a certain level of consumed resources and only in standard environment but not in flexible environment. Fees are charged for additional storage, bandwidth, or instance hours required by the application. It was first released as a preview version in April 2008 and came out of preview in September 2011.



Google App Engine primarily supports Go, PHP, Java, Python, Node.js, .NET, and Ruby applications, although it can also support other languages via "custom runtimes". Python web frameworks that run on Google App Engine include Django, CherryPy, Pyramid, Flask, web2py and webapp2, as well as a custom Google-written webapp framework and several others designed specifically for the platform that emerged since the release. Any Python framework that supports the WSGI using the CGI adapter can be used to create an application; the framework can be uploaded with the developed application. Third-party libraries written in pure Python may also be uploaded.

Google App Engine supports many Java standards and frameworks. Core to this is the servlet 2.5 technology using the open-source Jetty Web Server, along with accompanying technologies such as JSP. Java Server Faces operates with some workarounds. A newer release of App Engine Standard Java in Beta supports Java8, Servlet 3.1 and Jetty9. Though the integrated database, Google Cloud Datastore, may be unfamiliar to programmers, it is accessed and supported with JPA, JDO, and by the simple low-level API. There are several alternative libraries and frameworks you can use to model and map the data to the database such as Objectify, Slim3 and Jello framework. The Spring Framework works with GAE. However, the Spring Security module (if used) requires workarounds. Apache Struts 1 is supported, and Struts 2 runs with workarounds. The Django web framework and applications running on it can be used on App Engine with modification. Django-nonrel aims to allow Django to work with non-relational databases and the project includes support for App Engine.



### **Reliability and support**

- All billed App Engine applications have a 99.95% uptime SLA.
- App Engine is designed in such a way that it can sustain multiple datacentre outages without any downtime. This resilience to downtime is shown by the statistic that the High Replication Datastore saw 0% downtime over a period of a year.
- Paid support from Google engineers is offered as part of Premier Accounts

## **Bulk downloading**

SDK version 1.2.2 adds support for bulk downloads of data using Python. The open-source Python projects gaebar, appocket, and gawsh also allow users to download and back up App Engine data. No method for bulk downloading data from GAE using Java currently exists.

### Restrictions

- Developers have read-only access to the filesystem on App Engine. Applications can use only virtual filesystems, like gae-filestore.
- App Engine can only execute code called from an HTTP request (scheduled background tasks allow for self-calling HTTP requests).
- Users may upload arbitrary Python modules, but only if they are pure-Python; C and Pyrex modules are not supported.
- Java applications may only use a subset (The JRE Class White List) of the classes from the JRE standard edition. This restriction does not exist with the App Engine Standard Java8 runtime.
- A process started on the server to answer a request can't last more than 60 seconds (with the 1.4.0 release, this restriction does not apply to background jobs anymore).
- Does not support sticky sessions (a.k.a. session affinity), only replicated sessions are supported including limitation of the amount of data being serialized and time for session serialization.

App Engine provides more infrastructure to make it easy to write scalable applications, but can only run a limited range of applications designed for that infrastructure. App Engine's infrastructure removes many of the system administration and development challenges of building applications to scale to hundreds of requests per second and beyond. Google handles deploying code to a cluster, monitoring, failover, and launching application instances as necessary. While other services let users install and configure nearly any \*NIX compatible software, App Engine requires developers to use only its supported languages, APIs, and frameworks. Current APIs allow storing and retrieving data from the document-oriented Google Cloud Datastore database; making HTTP requests; sending e-mail; manipulating images; and caching. Google Cloud SQL can be used for App Engine applications requiring a relational MySQL compatible database backend. Per-day and per-minute quotas restrict bandwidth and CPU use, number of requests served, number of concurrent requests, and calls to the various APIs, and individual requests are terminated if they take more than 60 seconds or return more than 32MB of data.

Google App Engine's integrated Google Cloud Datastore database has a SQL-like syntax called "GQL" (Google Query Language). GQL does not support the Join statement. Instead, one-to-many and many-to-many relationships can be accomplished using ReferenceProperty (). Google Firestore is the successor to Google Cloud Datastore and replaces GQL with a document-based query method that treats stored objects as collections of documents.

Google App Engine Features:

- Blob store for serving large data objects;
- GAE Cloud Storage for storing data objects;
- Page Speed Service for automatically speeding up webpage load times;
- URL Fetch Service to issue HTTP requests and receive responses for efficiency and scaling; and
- Memcache for a fully managed in-memory data store.

### **Benefits of GAE**

- Ease of setup and use. GAE is fully managed, so users can write code without considering IT operations and back-end infrastructure. The built-in APIs enable users to build different types of applications. Access to application logs also facilitates debugging and monitoring in production.
- Pay-per-use pricing. GAE's billing scheme only charges users daily for the resources they use. Users can monitor their resource usage and bills on a dashboard.
- Scalability. Google App Engine automatically scales as workloads fluctuate, adding and removing application instances or application resources as needed.
- Security. GAE supports the ability to specify a range of acceptable Internet Protocol (IP) addresses. Users can allow list specific networks and services and blocklist specific IP addresses.

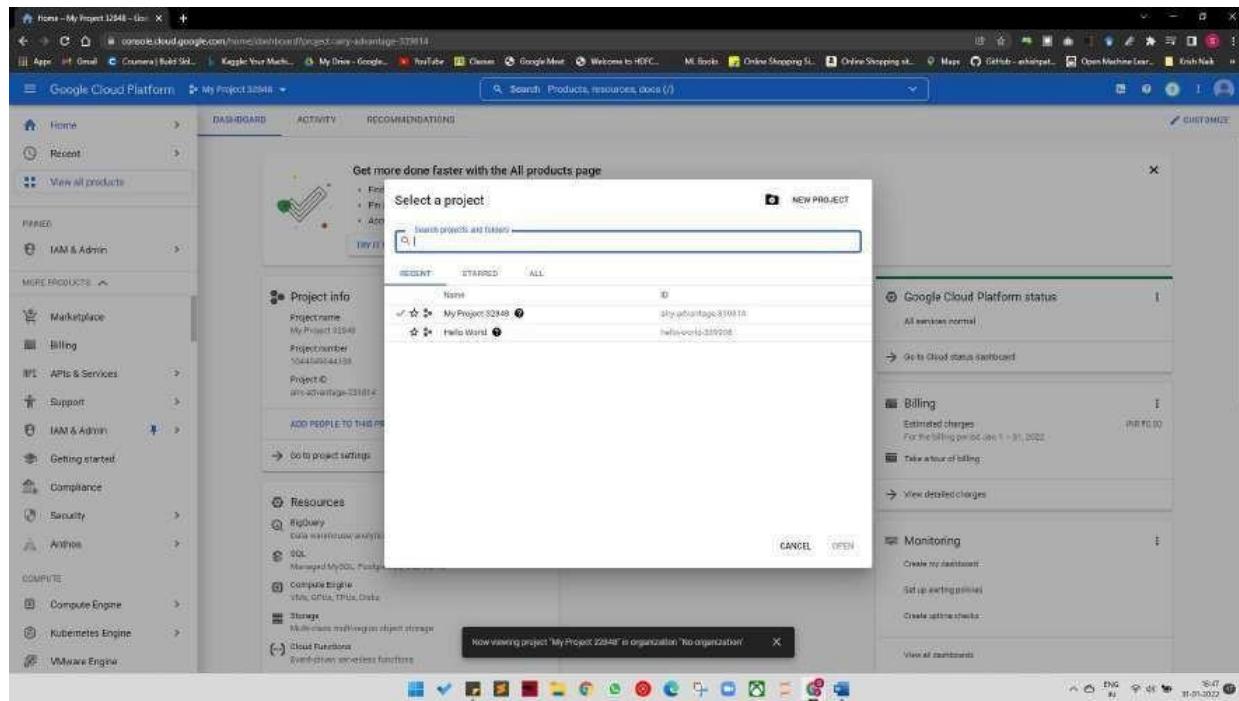
### **GAE challenges**

- Lack of control. Although a managed infrastructure has advantages, if a problem occurs in the back-end infrastructure, the user is dependent on Google to fix it.
- Performance limits. CPU-intensive operations are slow and expensive to perform using GAE. This is because one physical server may be serving several separate, unrelated app engine users at once who need to share the CPU.
- Limited access. Developers have limited, read-only access to the GAE filesystem.
- Java limits. Java apps cannot create new threads and can only use a subset of the Java runtime environment standard edition classes.

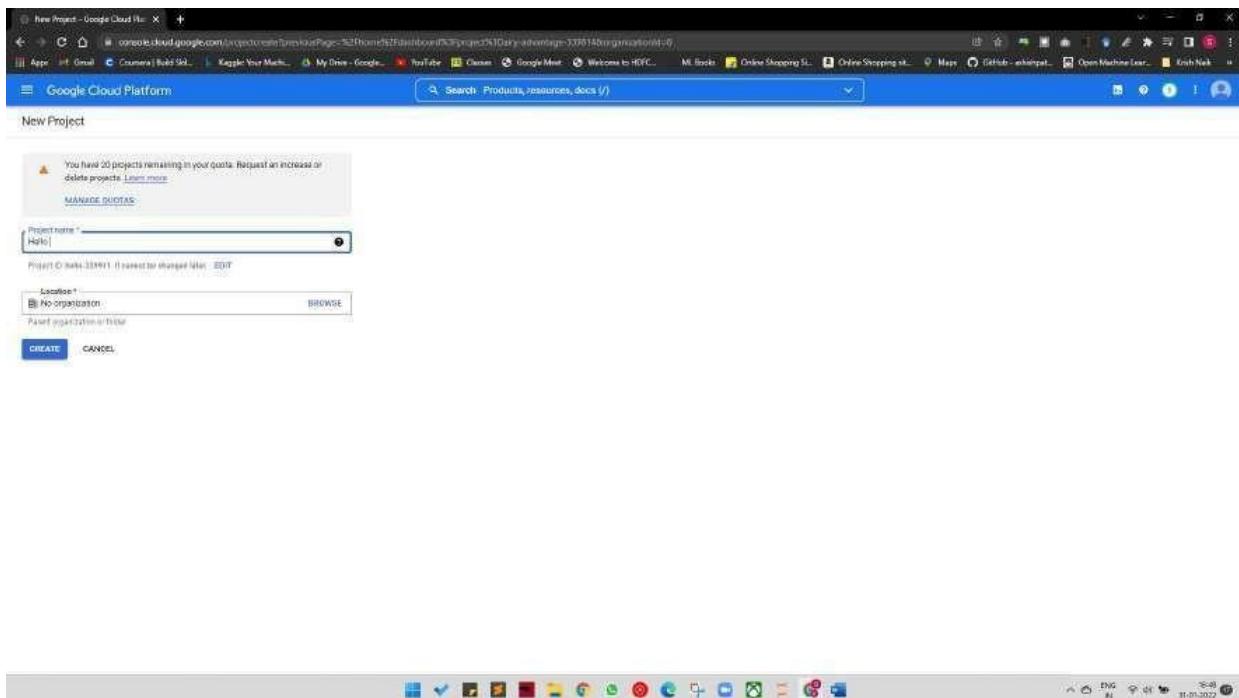
## Install Google App Engine. Create hello world app and other simple web applications using python/java

**Step 1:** Create an account on Google App Engine.

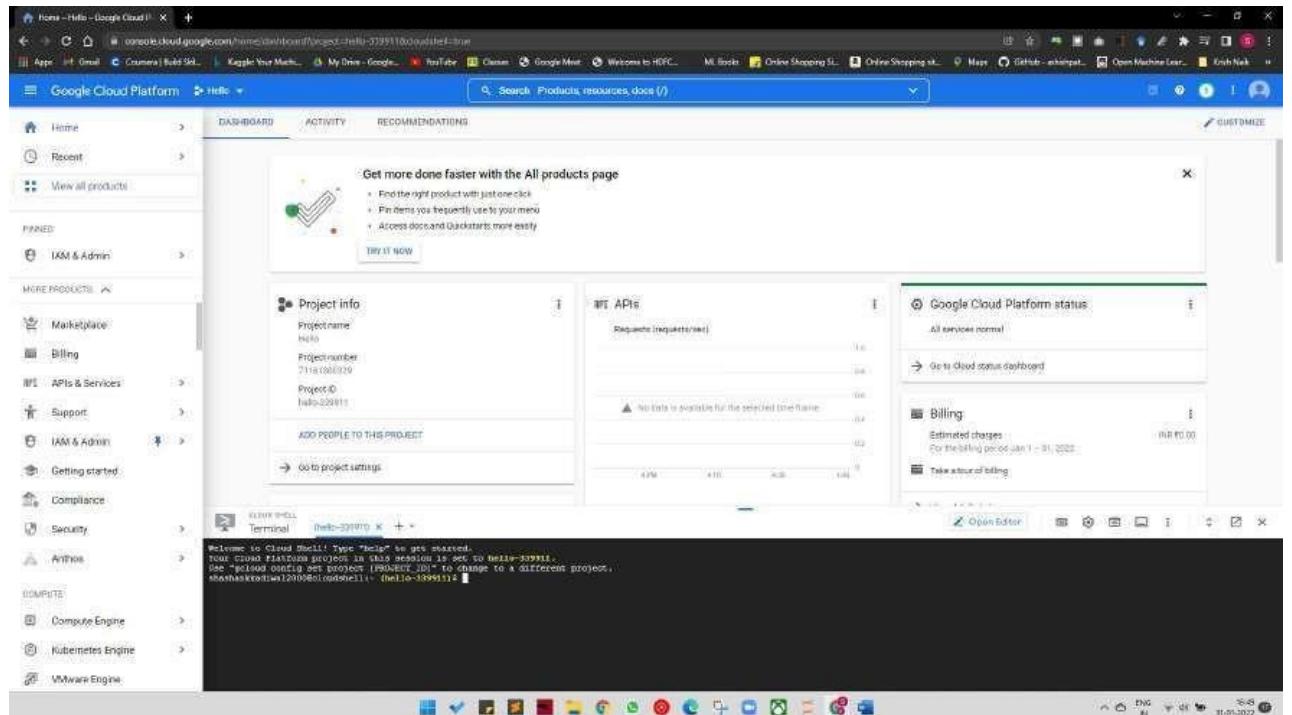
**Step 2:** Log in to your Google App Engine Account.



**Step 3:** Create a New Project and give a Project Name and click on create.



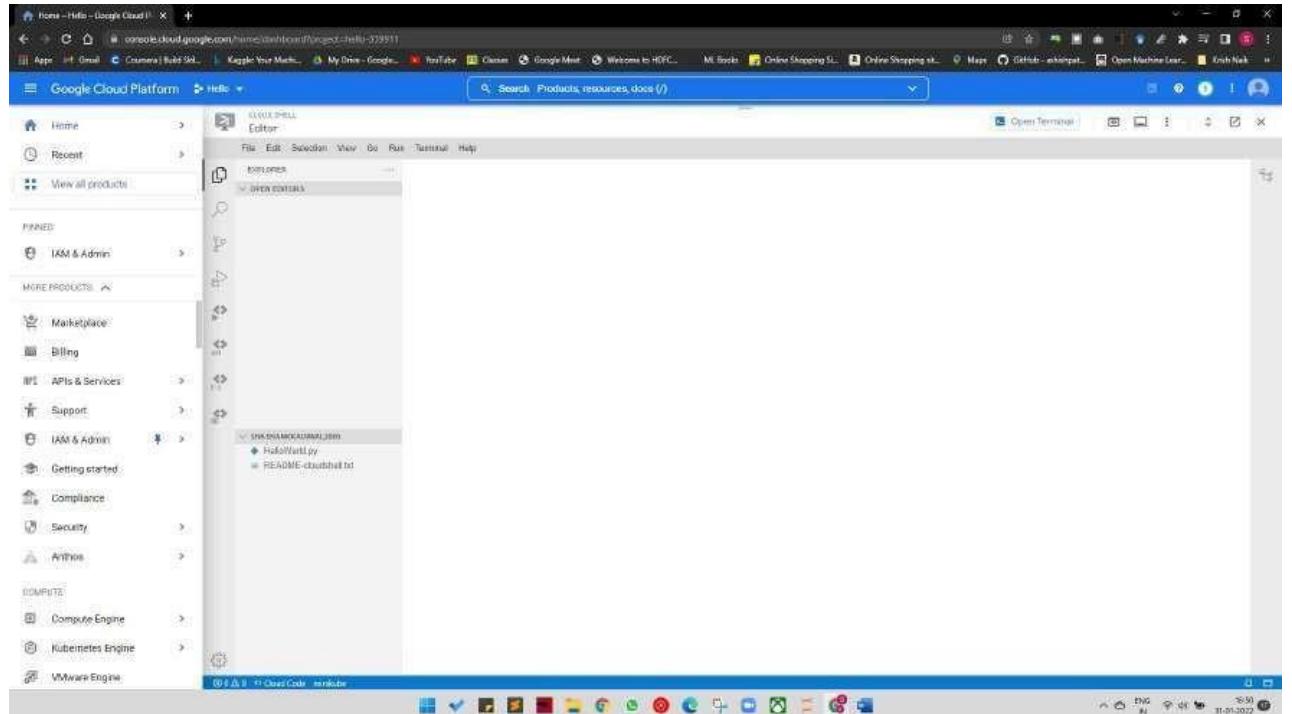
**Step 4:** Once the project is created click on Activate Cloud Shell on the top right corner.



**Step 5:** Once the cloud shell gets activated type the following command in the terminal and choose the region as South Asia

**gcloud app create**

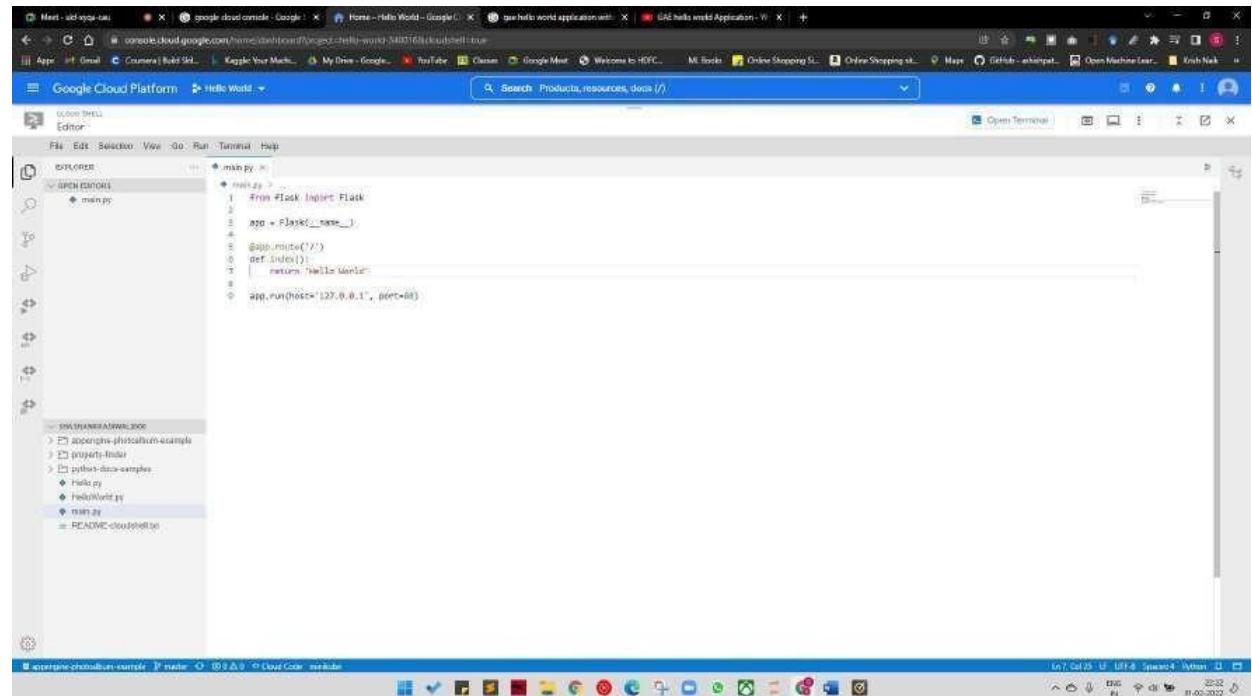
**Click on Open Editor.**



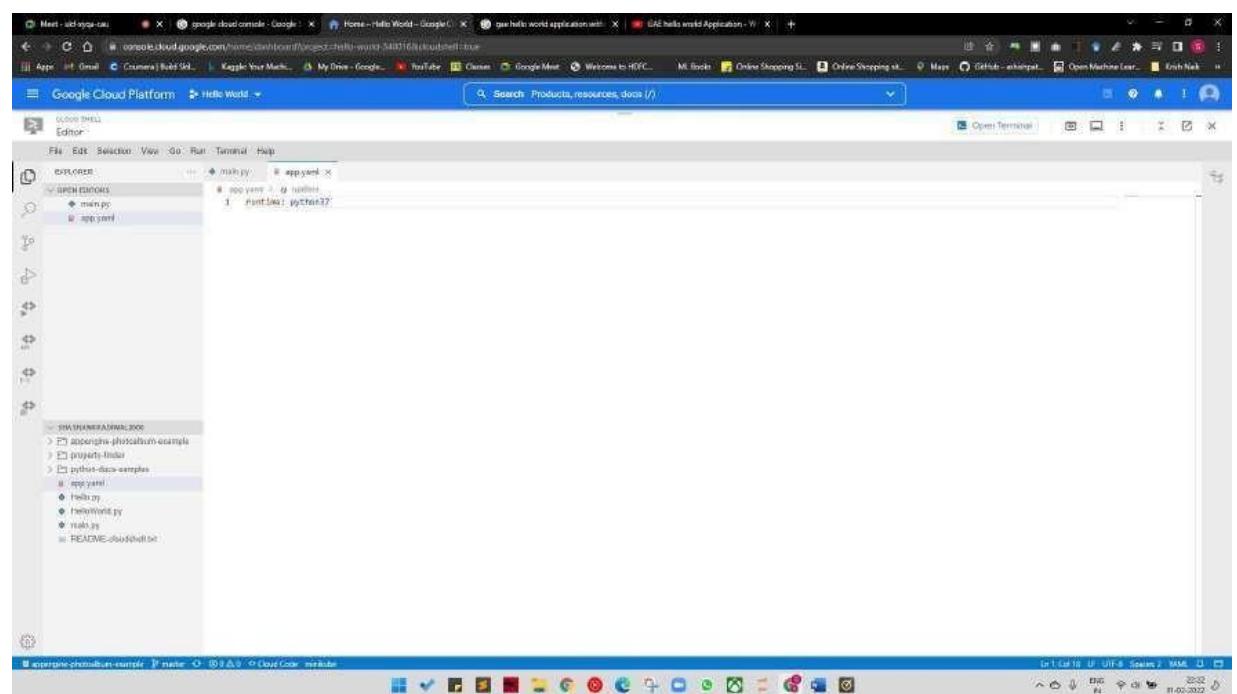
**Step 6:** Click on File and New File.

**Step 7:** Name the File as main.py and type the following code:

```
from flask import Flask
app = Flask(__name__)
def index():
    return 'Hello World'
app.run(host='127.0.0.1', port=80)
```

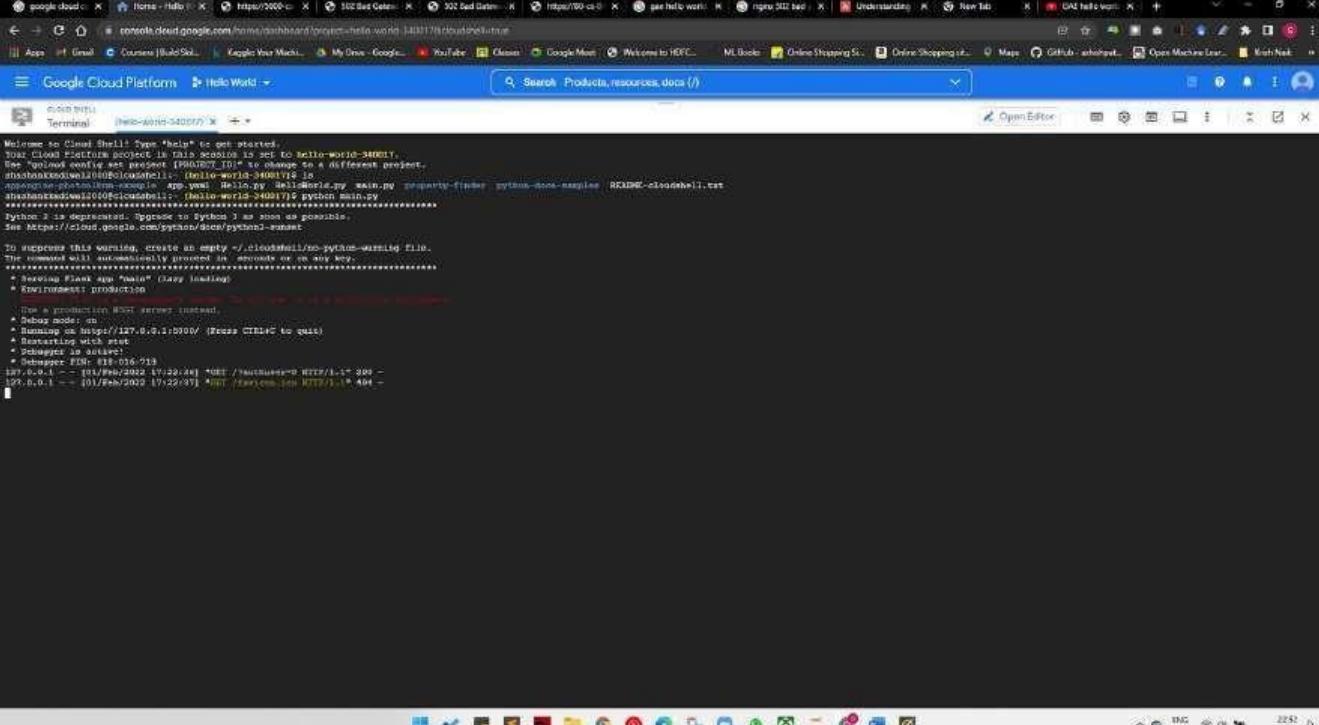


**Step8:** Create a app.yaml file



**Step 9:** Go to terminal and type the following command

**python main.py**



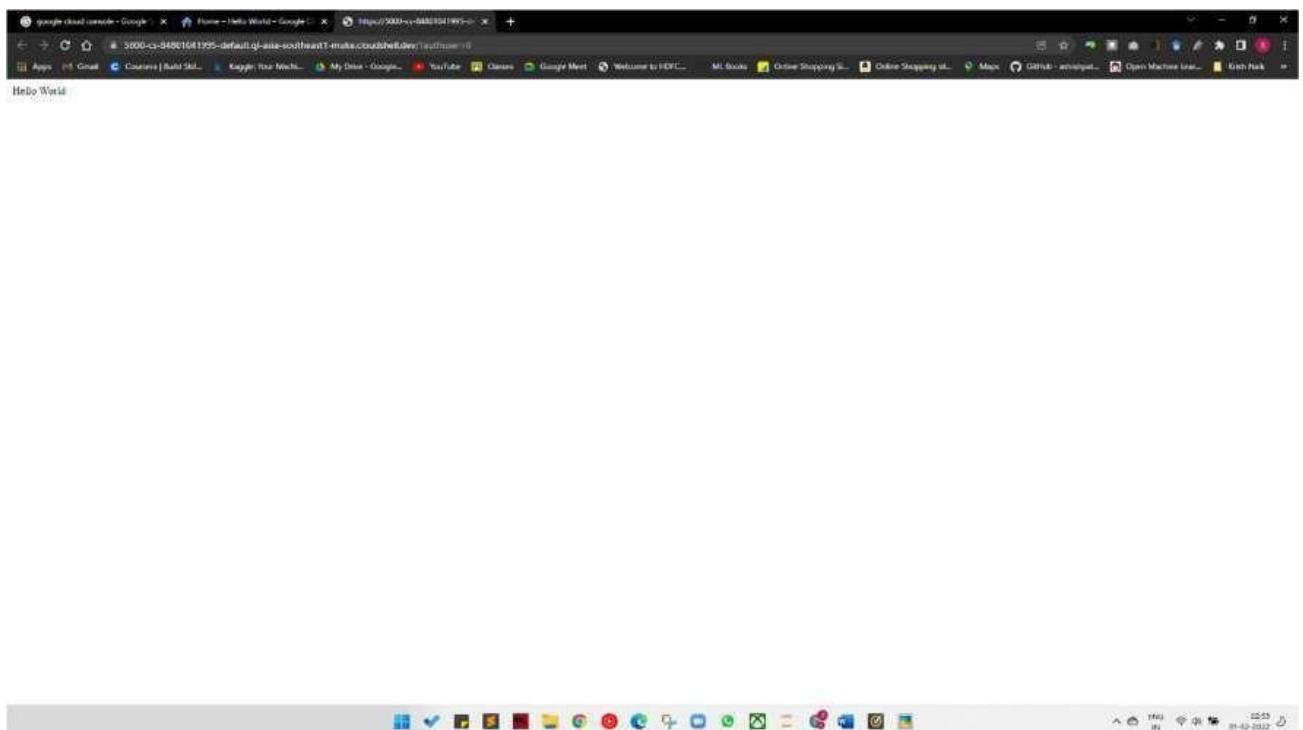
```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to Hello-World-340610.
To change projects, run "gcloud config set project PROJECT_ID" or switch to a different project.
ssh -L 5000:localhost:5000 cloudshell: (Hello-World-340610) ~
app-engine-python-hello-world: app.yaml, HelloWorld.py, main.py, property-finder, python-django-example README-cloudshell.txt
ssh -L 5000:localhost:5000 cloudshell: (Hello-World-340610) ~ python main.py
-----
Python 3 is deprecated. Upgrade to Python 3 as soon as possible.
See https://cloud.google.com/python/docs/python-version

To suppress this warning, create an empty ~/.cloudshell/python-warning file.
The command will automatically proceed in seconds or on any key.
-----
* Starting Flask app "main" (lazy loading)
* Environment: production
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Reloader is active!
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [01/Feb/2022 17:22:46] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [01/Feb/2022 17:22:47] "GET /favicon.ico HTTP/1.1" 404 -
```

**Note:** If Flask is not installed, please use the following command

**pip install flask**

**Step 10:** Open the https link



## **AMAZON WEB SERVICES**

Amazon Web Services, Inc. (AWS) is a subsidiary of Amazon providing on- demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered pay-as-you-go basis. These cloud computing web services provide a variety of basic abstract technical infrastructure and distributed computing building blocks and tools. One of these services is Amazon Elastic Compute Cloud (EC2), which allows users to have at their disposal a virtual cluster of computers, available all the time, through the Internet. AWS's virtual computers emulate most of the attributes of a real computer, including hardware central processing units (CPUs) and graphics processing units (GPUs) for processing; local/RAM memory; hard-disk/SSD storage; a choice of operating systems; networking; and pre-loaded application software such as web servers, databases, and customer relationship management (CRM).

The AWS technology is implemented at server farms throughout the world, and maintained by the Amazon subsidiary. Fees are based on a combination of usage (known as a "Pay-as-you-go" model), hardware, operating system, software, or networking features chosen by the subscriber required availability, redundancy, security, and service options. Subscribers can pay for a single virtual AWS computer, a dedicated physical computer, or clusters of either. As part of the subscription agreement, Amazon provides security for subscribers' systems. AWS operates from many global geographical regions including 6 in North America.

Amazon markets AWS to subscribers as a way of obtaining large scale computing capacity more quickly and cheaply than building an actual physical server farm. All services are billed based on usage, but each service measures usage in varying ways. As of 2017, AWS owns 33% of all cloud (IaaS, PaaS) while the next two competitors Microsoft Azure and Google Cloud have 18%, and 9% respectively, according to Synergy Group.

As of 2021, AWS comprises over 200 products and services including computing, storage, networking, database, analytics, application services, deployment, management, machine learning, mobile, developer tools, RobOps and tools for the Internet of Things. The most popular include Amazon Elastic Compute Cloud (EC2), Amazon Simple Storage Service (Amazon S3), Amazon Connect, and AWS Lambda (a serverless function enabling serverless ETL e.g., between instances of EC2 & S3).

Most services are not exposed directly to end users, but instead offer functionality through APIs for developers to use in their applications. Amazon Web Services' offerings are accessed over HTTP, using the REST architectural style and SOAP protocol for older APIs and exclusively JSON for newer ones



## Applications of AWS services

Amazon Web services are widely used for various computing purposes like:

- Web site hosting
- Application hosting/SaaS hosting
- Media Sharing (Image/ Video)
- Mobile and Social Applications
- Content delivery and Media Distribution
- Storage, backup, and disaster recovery
- Development and test environments
- Academic Computing
- Search Engines
- Social Networking

## **RELATIONAL DATABASES**

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks, such as hardware provisioning, database setup, patching, and backups. It frees you to focus on your applications so you can give them the fast performance, high availability, security, and compatibility they need.

Amazon RDS is available on several database instance types - optimized for memory, performance, or I/O - and provides you with six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and SQL Server. You can use the AWS Database Migration Service to easily migrate or replicate your existing databases to Amazon RDS.



## **Benefits of Amazon RDS**

- Easy to Administer

Amazon RDS makes it easy to go from project conception to deployment. Use the Amazon RDS Management Console, the AWS RDS Command-Line Interface, or simple API calls to access the capabilities of a production-ready relational database in minutes. No need for infrastructure provisioning, and no need for installing and maintaining database software.

- Highly Scalable

You can scale your database's compute and storage resources with only a few mouse clicks or an API call, often with no downtime. Many Amazon RDS engine types allow you to launch one or more Read Replicas to offload read traffic from your primary database instance.

- Available and Durable

Amazon RDS runs on the same highly reliable infrastructure used by other Amazon Web Services. When you provision a Multi-AZ DB Instance, Amazon RDS synchronously replicates the data to a standby instance in a different Availability Zone (AZ). Amazon RDS has many other features that enhance reliability for critical production databases, including automated backups, database snapshots, and automatic host replacement.

- Fast

Amazon RDS supports the most demanding database applications. You can choose between two SSD-backed storage options: one optimized for high- performance OLTP applications, and the other for cost-effective general-purpose use. In addition, Amazon Aurora provides performance on par with commercial databases at 1/10th the cost.

- Secure

Amazon RDS makes it easy to control network access to your database. Amazon RDS also lets you run your database instances in Amazon Virtual Private Cloud (Amazon VPC), which enables you to isolate your database instances and to connect to your existing IT infrastructure through an industry-standard encrypted IPsec VPN. Many Amazon RDS engine types offer encryption at rest and encryption in transit.

- Inexpensive

You pay very low rates and only for the resources you actually consume. In addition, you benefit from the option of On-Demand pricing with no up-front or long-term commitments or even lower hourly rates via our Reserved Instance pricing.

## **VPC**

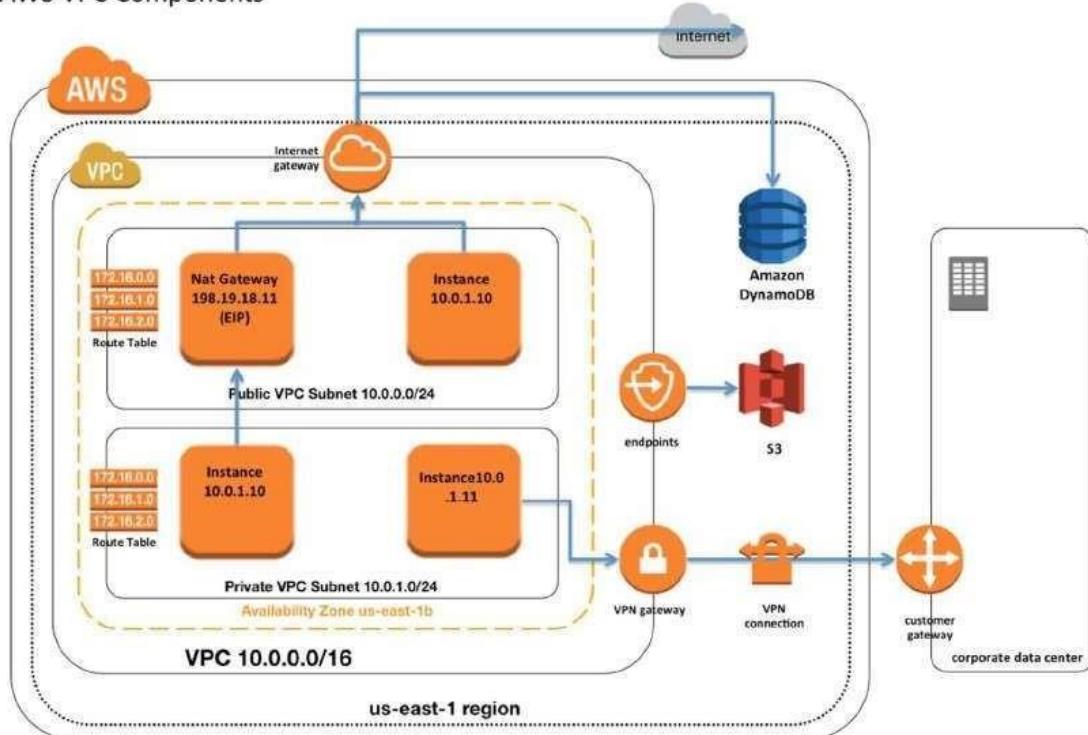
Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data centre, with the benefits of using the scalable infrastructure of AWS. Amazon VPC is the networking layer for Amazon EC2.

The following are the key concepts for VPCs:

- Virtual private cloud (VPC) — A virtual network dedicated to your AWS account.
- Subnet — A range of IP addresses in your VPC.
- CIDR block — Classless Inter-Domain Routing. An internet protocol address allocation and route aggregation methodology. For more information, see [Classless Inter-Domain Routing](#) in Wikipedia.
- Route table — A set of rules, called routes, that are used to determine where network traffic is directed.
- DHCP options sets: Configuration information (such as domain name and domain name server) passed to EC2 instances when they are launched into VPC subnets.
- Internet gateway — A gateway that you attach to your VPC to enable communication between resources in your VPC and the internet.
- Egress-only internet gateways: A type of internet gateway that allows an EC2 instance in a subnet to access the internet but prevents resources on the internet from initiating communication with the instance.
- VPC endpoint — Enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service.
- NAT gateway: A managed AWS service that allows EC2 instances in private subnets to connect to the internet, other VPCs, or on-premises networks.
- NAT instance: An EC2 instance in a public subnet that allows instances in private subnets to connect to the internet, other VPCs, or on-premises networks.
- Carrier gateways: For subnets in Wavelength Zones, this type of gateway allows inbound traffic from a telecommunication carrier network in a specific location and outbound traffic to a telecommunication carrier network and the internet.

- Prefix lists: A collection of CIDR blocks that can be used to configure VPC security groups, VPC route tables, and AWS Transit Gateway route tables and can be shared with other AWS accounts using Resource Access Manager (RAM).
- Security groups: Acts as a virtual firewall to control inbound and outbound traffic for an AWS resource, such as an EC2 instance. Each VPC comes with a default security group, and you can create additional security groups. A security group can be used only in the VPC for which it's created.

AWS VPC Components

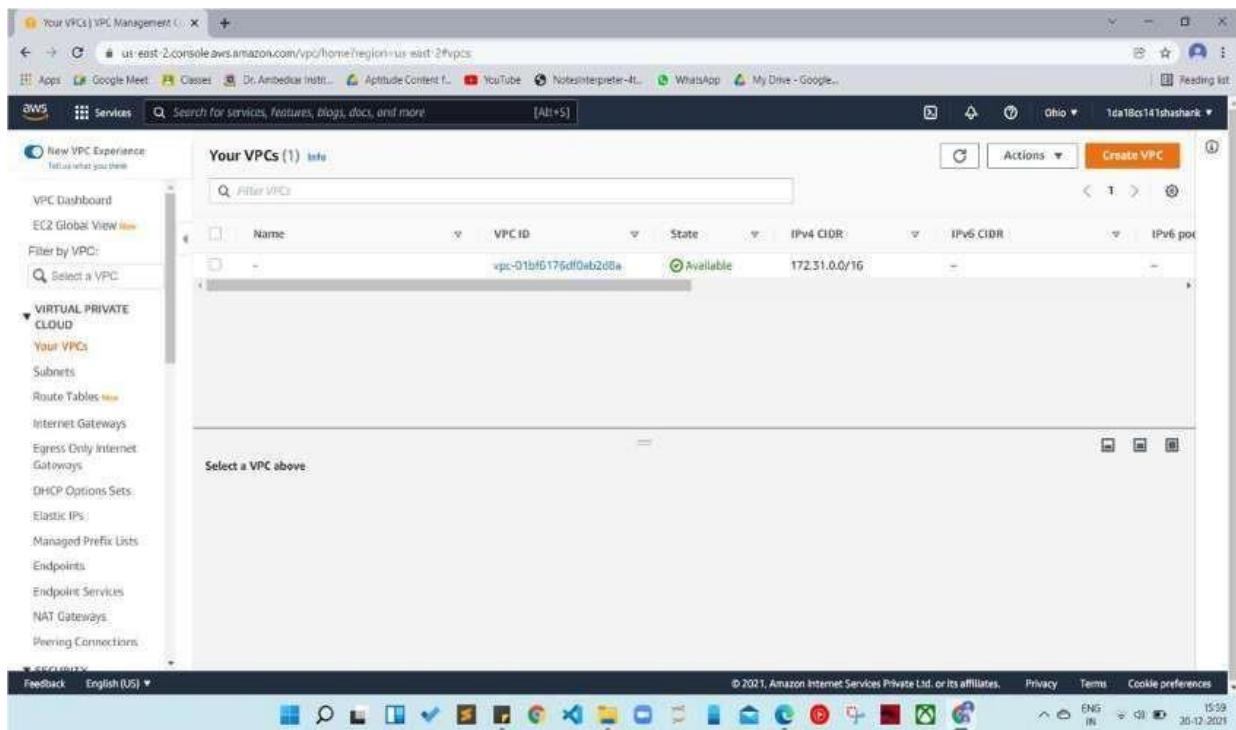


## Create a RDS and launch in your custom VPC network

**Step 1:** Open “AWS Management Console”. Click on “VPC” service.

**Step 2:** Click “Subnets” on the left panel.

**Step 3:** Now you can see there is one subnet group (Public Subnet) created in your VPC (Your VPC id/11.0.0.0/16). Now Click on “Create Subnet” button



**Step 4:** Give a Name to the subnet and select you own VPC from the “VPC drop down”. Select an “availability zone” (Which is not used by “Public Subnet” of your VPC). Give CIDR block range. Click on “Yes, Create” button.

**Step 5:** Now you can see the subnet is created in your VPC.

**Step 6:** Click on “Services”. Click on “RDS” service.

**Step 7:** Click on “Subnet Groups” on the left panel. (Note: Before creating the subnet groups you need to note down your VPC ID and subnets for that VPC).

**Step 8:** Click on “Create DB Subnet Group” button.

**Step 9:** Give a name to the “Subnet”. Select your own “VPC ID”. Select the “Availability Zone” and “subnet”. Click on “Add” button.

**Step 10:** Now select another “Availability Zone” and “Subnet”. Click on “Yes, Create” button. (Note: Before clicking on “Yes, create” check that two subnets are added or not)

**Step 11:** Now you can see that “DB subnet group” is created.

**Step 12:** Click on “RDS Dashboard” on the left panel. Click on “Get Started” button

**Step 13:** Select your desired “Database Engine” from the list and click on “Select” button.

**Step 14:** If you want multi-AZ Deployment, select the “first radio button” otherwise select the “second radio button”. Click on “Next” button.

Subnet ID	Subnet ARN	State	IPv4 CIDR
subnet-07c4baa3fbf019c49	arn:aws:ec2:us-east-2:881380715570:subnet/subnet-07c4baa3fbf019c49	Available	172.31.64.0/24
251			
vpc-01bf6176df0ab2d8a			
Auto-assign public IPv4 address: No	Route table: rtb-069b0b82e7e500e7e	Availability Zone: us-east-2a	Customer-owned IPv4 pool: No
Output ID: -	Auto-assign IPv6 address: No	Network ACL: acl-03a16c6400ea1909c	Default subnet: No
		IPv6 CIDR reservations: No	

**Step 15:** Select your desired DB instance from the “DB Instance Class” drop down. Select Multi-AZ option from the drop down. Select the storage type from the drop down and give 95 your desired storage space range. Enter the Details (DB instance Identifier, Master user name, password and confirm password). Click on “Next” button.

Availability zone	Subnet ID	CIDR block
us-east-2a	subnet-0819dfe582fc96219	172.31.0.0/20
us-east-2b	subnet-0b1bb051855021d47	172.31.16.0/20
us-east-2c	subnet-05a273a0c7af62103	172.31.32.0/20
us-east-2a	subnet-07c4baa3fbf019c49	172.31.64.0/24

**Step 16:** Select your own VPC ID from the “VPC Dropdown”. Select the “public accessibility” from the dropdown. Select your desired Availability Zone from the dropdown. Give a name to your Database and check the database port is and mention the “Backup retention period” as per your needs and if you want a Time frame for your backup, configure the time frame from “Backup Window” Dropdown. Click on “Launch DB Instance” button.

The screenshot shows the AWS RDS Management Console. A green success message at the top right says "Successfully created Mydbsubnet. View subnet group". The main pane displays a table titled "Subnet groups (1)". The table has columns: Name, Description, Status, and VPC. One row is shown: "mydbsubnet", "db-subnet", "Complete", and "vpc-01bf6176df0eb2d8a". The left sidebar shows navigation links for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Events, and Event subscriptions. The bottom status bar shows the date as 20.12.2021 and the time as 16:37.

The screenshot shows the "Create database" wizard for MySQL. At the top, it says "Choose a database creation method": "Standard create" (selected) and "Easy create". Below that is the "Engine options" section, which lists "Engine type": "MySQL" (selected), "Amazon Aurora", "MariaDB", "PostgreSQL", "Oracle", and "Microsoft SQL Server". To the right, there is a "MySQL" section with a description: "MySQL is the most popular open-source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database." It also lists several bullet points about MySQL support. The left sidebar is identical to the previous screenshot, showing the same navigation links. The bottom status bar shows the date as 20.12.2021 and the time as 16:45.

Screenshot of the AWS RDS Management Console showing the "Engine options" section for creating a new DB instance.

**Amazon RDS**

**Engine type:** MySQL (selected)

**Edition:** MySQL Community

**Known issues/limitations:** Review the Known issues/limitations link to learn about potential compatibility issues with specific database versions.

**Version:** MySQL 8.0.23

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 5 Read Replicas per instance, within a single Region or cross-region.

Screenshot of the AWS RDS Management Console showing the "Settings" section for creating a new DB instance.

**Amazon RDS**

**DB instance identifier:** Mydb

**Credentials Settings**

**Master username:** Sheshank

**Master password:** (Input field)

**Confirm password:** (Input field)

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 5 Read Replicas per instance, within a single Region or cross-region.

Servers | VPC Management Con... RDS Management Console

us-east-2.console.aws.amazon.com/rds/home?region=us-east-2&source=dbinstance&db=false&3...import=false

get 2/2 Ohio Tda18cs14tishank

**Amazon RDS**

Dashboard Databases Query Editor Performance insights Snapshots Automated backups Reserved instances Proxies Subnet groups Parameter groups Option groups Custom engine versions Events Event subscriptions Recommendations

Feedback English (US)

Confirm password: Info:

**DB instance class**

DB instance class: Info:  Standard classes (includes m classes)  Memory optimized classes (includes r and x classes)  Burstable classes (includes t classes)

db.t3.micro 2 vCPUs 1 GiB RAM Network: 2.085 Mbps

Include previous generation classes

**Storage**

Storage type: Info: General Purpose SSD (gp2) Baseline performance determined by volume size

Allocated storage:  GiB

© 2021, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 16:47 20-12-2021

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 5 Read Replicas per instance, within a single Region or cross-region.

Servers | VPC Management Con... RDS Management Console

us-east-2.console.aws.amazon.com/rds/home?region=us-east-2&source=dbinstance&db=false&3...import=false

get 2/2 Ohio Tda18cs14tishank

**Amazon RDS**

Dashboard Database Query Editor Performance insights Snapshots Automated backups Reserved instances Proxies Subnet groups Parameter groups Option groups Custom engine versions Events Event subscriptions Recommendations

Feedback English (US)

**Storage**

Storage type: Info: General Purpose SSD (gp2) Baseline performance determined by volume size

Allocated storage:  GiB

(Minimum: 20 GiB, Maximum: 16,384 GiB) Higher allocated storage may improve IOPS performance.

Provisioning less than 100 GiB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. Learn more ↗

**Storage autoscaling** Info: Provides dynamic scaling support for your database's storage based on your application's needs.

Enable storage autoscaling Enabling this feature will allow the storage to increase once the specified threshold is exceeded.

Maximum storage threshold: Info: Charges will apply when your database autoscales to the specified threshold

1000 GiB

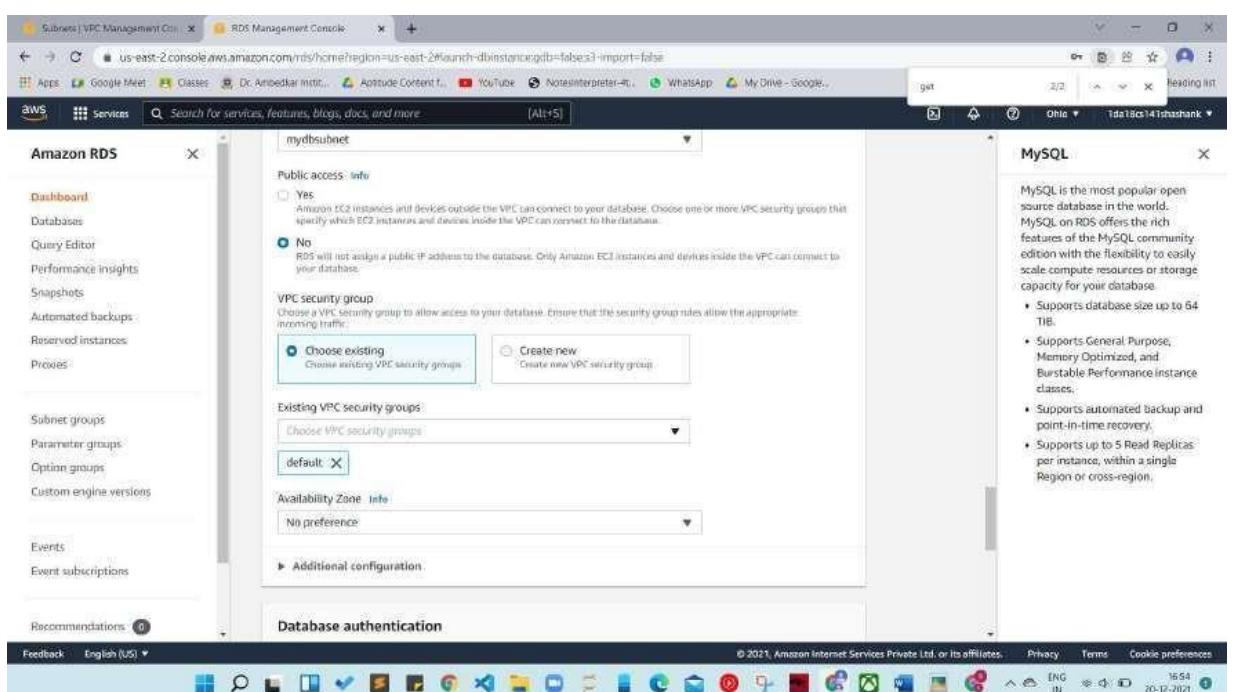
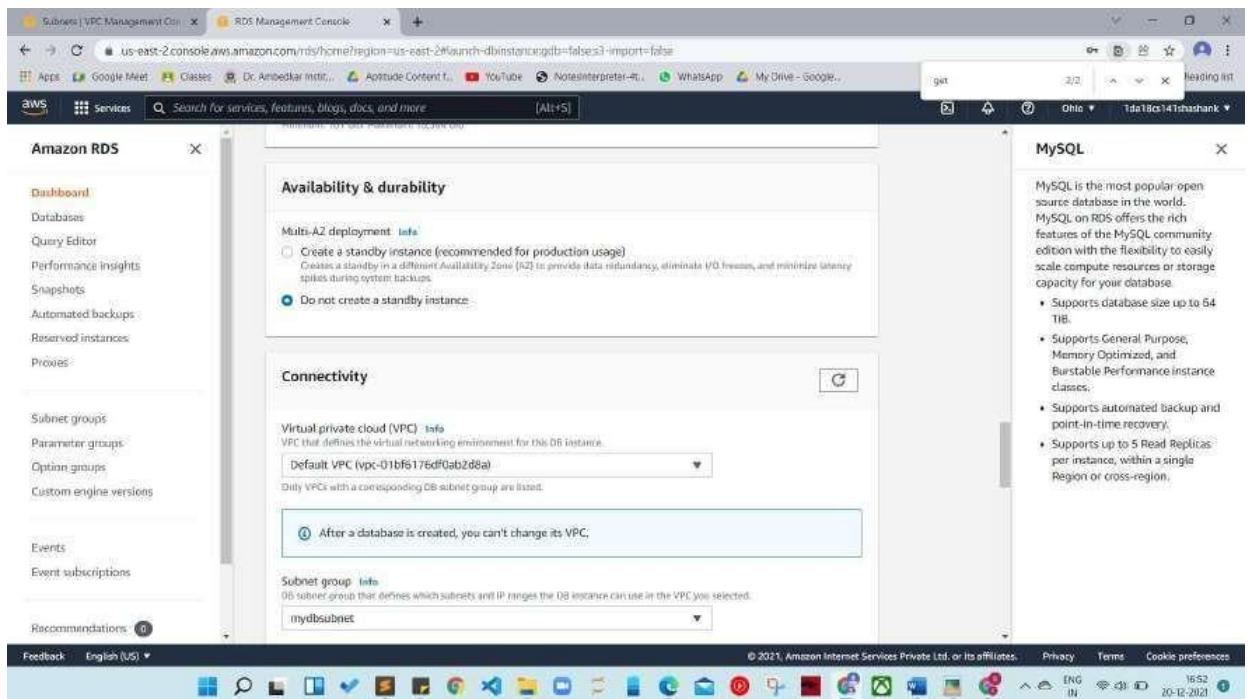
(Minimum: 21 GiB, Maximum: 16,384 GiB)

© 2021, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences ENG IN 16:54 20-12-2021

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 5 Read Replicas per instance, within a single Region or cross-region.



**Step 17:** Click on “Close” button.

**Step 18:** Now you can see one DB instance is created. (It will take 5-10min to create the database instance). You will have the DB name, VPC ID, End Point (which is used to connect to the DB Instance from your EC2 instance) and etc...

The screenshot shows the AWS RDS Management Console for creating a new MySQL database. The left sidebar lists various RDS services like Databases, Query Editor, and Performance Insights. The main area is titled 'MySQL' and shows the configuration for a new instance. It includes sections for 'Additional configuration' (with options for Password and Kerberos authentication), 'Estimated monthly costs' (showing DB instance at 12.41 USD, Storage at 2.50 USD, and Total at 14.71 USD), and a note about third-party rights. A 'Create database' button is visible at the bottom right.

The screenshot shows the AWS RDS Management Console after a database named 'mydb' has been created. The left sidebar remains the same. The main area shows a confirmation message 'Creating database mydb' and a 'View credential details' button. Below it, the 'Databases' section displays a table with one row for 'mydb'. The table columns include DB identifier, Role, Engine, Region & AZ, Size, Status, and CPU. The database is listed as 'Backing up'.

The screenshot shows the AWS RDS Management Console. A green success banner at the top right says "Successfully created database mydb". The left sidebar has "Databases" selected. The main "Databases" page lists "mydb" with details: Instance: MySQL Community, Region & AZ: us-east-2a, Class: db.t2.micro, Status: Available, CPU: 3.8%. The bottom status bar shows "Feedback English (US)" and the date "20-12-2021".

This screenshot shows the detailed view for the "mydb" database. The "Summary" tab is active, displaying basic information: DB identifier: mydb, Role: Instance, CPU: 3.87%, Status: Available, Engine: MySQL Community, Class: db.t2.micro, Region & AZ: us-east-2a. Below it, the "Connectivity & security" tab is selected, showing the endpoint: mydb.cimimc1tuy6.us-east-2.rds.amazonaws.com, and networking details like Availability Zone: us-east-2a. The bottom status bar shows "Feedback English (US)" and the date "20-12-2021".

Note: To connect to the Database from your Ec2 Instance, you need the following.

- RDS end point.
- Database Name.
- Master username.
- Master Password.
- Port Number.

# **VM WARE, VIRTUAL BOX AND GUEST OS**

## **Virtual Box**

VirtualBox is a powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use. Not only is VirtualBox an extremely feature rich, high-performance product for enterprise customers, it is also the only professional solution that is freely available as Open-Source Software under the terms of the GNU General Public License (GPL) version 2. See "About VirtualBox" for an introduction.

Presently, VirtualBox runs on Windows, Linux, Macintosh, and Solaris hosts and supports a large number of guest operating systems including but not limited to Windows (NT 4.0, 2000, XP, Server 2003, Vista, Windows 7, Windows 8, Windows 10), DOS/Windows 3.x, Linux (2.4, 2.6, 3.x and 4.x), Solaris and Open Solaris, OS/2, and OpenBSD.

VirtualBox is being actively developed with frequent releases and has an ever-growing list of features, supported guest operating systems and platforms it runs on. VirtualBox is a community effort backed by a dedicated company: everyone is encouraged to contribute while Oracle ensures the product always meets professional quality criteria.



## **History:**

VirtualBox was first offered by Innotek GmbH from Weinstadt, Germany, under a proprietary software license, making one version of the product available at no cost for personal or evaluation use, subject to the VirtualBox Personal Use and Evaluation License (PUEL). In January 2007, based on counsel by LiSoG, Innotek GmbH released VirtualBox Open-Source Edition (OSE) as free and open-source software, subject to the requirements of the GNU General Public License (GPL), version 2. Innotek GmbH also contributed to the development of OS/2 and Linux support in virtualization and OS/2 ports of products from Connectix which were later acquired by Microsoft. Specifically, Innotek developed the "additions" code in both Windows Virtual PC and Microsoft Virtual Server, which enables various host–guest OS interactions like shared clipboards or dynamic viewport resizing.

Sun Microsystems acquired Innotek in February 2008. Following the acquisition of Sun Microsystems by Oracle Corporation in January 2010, the product was re-branded as "Oracle VM VirtualBox". In December 2019, VirtualBox started supporting only hardware-assisted virtualization, dropping support for Software-based one.

## **Features:**

- Snapshots of the RAM and storage that allow reverting to a prior state.
- Screenshots and screen video capture
- "Host key" for releasing the keyboard and mouse cursor to the host system if captured (coupled) to the guest system, and for keyboard shortcuts to features such as configuration, restarting, and screenshot. By default, it is the right-side CTRL key.
- Mouse pointer integration, meaning automatic coupling and uncoupling of mouse cursor when moved inside and outside the virtual screen, if supported by guest operating system.
- Seamless mode – the ability to run virtualized applications side by side with normal desktop applications
- Shared clipboard
- Shared folders through "guest additions" software
- Special drivers and utilities to facilitate switching between systems
- Ability to specify amount of shared RAM, video memory, and CPU execution cap
- Ability to emulate multiple screens
- Command line interaction (in addition to the GUI)

- Public API (Java, Python, SOAP, XPCOM) to control VM configuration and execution
- Nested paging for AMD-V and Intel VT (only for processors supporting SLAT and with SLAT enabled)
- Limited support for 3D graphics acceleration (including OpenGL up to (but not including) 3.0 and Direct3D 9.0c via Wine's Direct3D to OpenGL translation)
- SMP support (up to 32 virtual CPUs per virtual machine), since version 3.0
- Teleportation (aka Live Migration)
- 2D video output acceleration (not to be mistaken with video decoding acceleration), since version 3.1
- EFI has been supported since version 3.1 (Windows 7 guests are not supported).

## **VM Ware**

VMware, Inc. is an American cloud computing and virtualization technology company with headquarters in California. VMware was the first commercially successful company to virtualize the x86 architecture. VMware's desktop software runs on Microsoft Windows, Linux, and macOS, while its enterprise software hypervisor for servers, VMware ESXi, is a bare-metal hypervisor that runs directly on server hardware without requiring an additional underlying operating system. VMware's most notable products are its hypervisors. VMware became well known for its first type 2 hypervisor known as GSX. This product has since evolved into two hypervisor product lines: VMware's type 1 hypervisors running directly on hardware and their hosted type 2 hypervisors. VMware software provides a completely virtualized set of hardware to the guest operating system. VMware software virtualizes the hardware for a video adapter, a network adapter, and hard disk adapters. The host provides pass-through drivers for guest USB, serial, and parallel devices. In this way, VMware virtual machines become highly portable between computers, because every host looks nearly identical to the guest. In practice, a system administrator can pause operations on a virtual machine guest, move or copy that guest to another physical computer, and their resume execution exactly at the point of suspension. Alternatively, for enterprise servers, a feature called vMotion allows the migration of operational guest virtual machines between similar but separate hardware hosts sharing the same storage (or, with vMotion Storage, separate storage can be used, too). Each of these transitions is completely transparent to any users on the virtual machine at the time it is being migrated.

VMware's products predate the virtualization extensions to the x86 instruction set, and do not require virtualization-enabled processors. On newer processors, the hypervisor is now designed to take advantage of the extensions. However, unlike many other hypervisors, VMware still supports older processors. In such cases, it uses the CPU to run code directly whenever possible (as, for example, when running user-mode and virtual 8086 mode code on x86). When direct execution cannot operate, such as with kernel-level and real-mode code, VMware products use binary translation (BT) to re-write the code dynamically. The translated code gets stored in spare memory, typically at the end of the address space, which segmentation mechanisms can protect and make invisible. For these reasons, VMware operates dramatically faster than emulators, running at more than 80% of the speed that the virtual guest operating system would run directly on the same hardware. In one study VMware claims a slowdown over native ranging from 0–6 percent for the VMware ESX Server.

## **Products:**

### **Desktop software**

- VMware Workstation, introduced in 1999, was the first product launched by VMware. This software suite allows users to run multiple instances of x86 or x86-64 -compatible operating systems on a single physical personal computer. Workstation Pro version 15.5.1 was released in Nov 2019
- VMware Fusion provides similar functionality for users of the Intel Mac platform, along with full compatibility with virtual machines created by other VMware products.
- VMware Workstation Player is freeware for non-commercial use, without requiring a license, and available for commercial use with permission. It is similar to VMware Workstation, with reduced functionality.

### **Server software**

- VMware ESXi, an enterprise software product, can deliver greater performance than the freeware VMware Server, due to lower system computational overhead. VMware ESXi, as a "bare-metal" product, runs directly on the server hardware, allowing virtual servers to also use hardware more or less directly. In addition, VMware ESXi integrates into VMware vCenter, which offers extra services.

## **Cloud management software**

- VMware vRealize Suite – a cloud management platform purpose-built for a hybrid cloud.
- VMware Go is a web-based service to guide users of any expertise level through the installation and configuration of VMware vSphere Hypervisor.
- VMware Cloud Foundation – Cloud Foundation provides an easy way to deploy and operate a private cloud on an integrated SDDC system.
- VMware Horizon View is a virtual desktop infrastructure (VDI) product.

## **Application management**

- The VMware Workspace Portal was a self-service app store for workspace management.

## **Storage and availability**

VMware's storage and availability products are composed of two primary offerings:

- VMware vSAN (previously called VMware Virtual SAN) is software-defined storage that is embedded in VMware's ESXi hypervisor. The vSphere and vSAN software run on industry-standard x86 servers to form a hyper-converged infrastructure (or HCI). However, network operators need to have servers from HCL (Hardware Compatibility List) to put one into production. The first release, version 5.5, was released in March 2014. The 6th generation, version 6.6, was released in April 2017. New features available in VMware vSAN 6.6 include native data at rest encryption, local protection for stretched clusters, analytics, and optimized solid-state drive performance. The VMWare 6.7 version was released in April 2018, Users now have improved monitoring tools and new workflows, it is closer to feature parity. The vCenter Server Appliance architecture is moving around to an easy deployment method.
- VMware Site Recovery Manager (SRM) automates the failover and failback of virtual machines to and from a secondary site using policy-based management.

## **Networking and security products**

- VMware NSX is VMware's network virtualization product marketed using the term software-defined data centre (SDDC). The technology included some acquired from the 2012 purchase of Nicira. Software Defined Networking (SDN) allows the same policies that govern Identity and Access Management (IAM) to dictate levels of access to applications and data through a totally converged infrastructure not possible with legacy network and system access methods.

## Other products

- Workspace ONE allows mobile users to access apps and data.
- The VIX (Virtual Infrastructure eXtension) API allows automated or scripted management of a computer virtualized using either VMware's vSphere, Workstation, Player, or Fusion products. VIX provides bindings for the programming languages C, Perl, Visual Basic, VBscript and C#.
- Herald is an communications protocol from VMware for more reliable Bluetooth communication and range finding across for mobile devices. Herald code is available under an Open-source license and was implemented in the Australian Government's COVID Safe app for contact tracing on 19 December 2020.

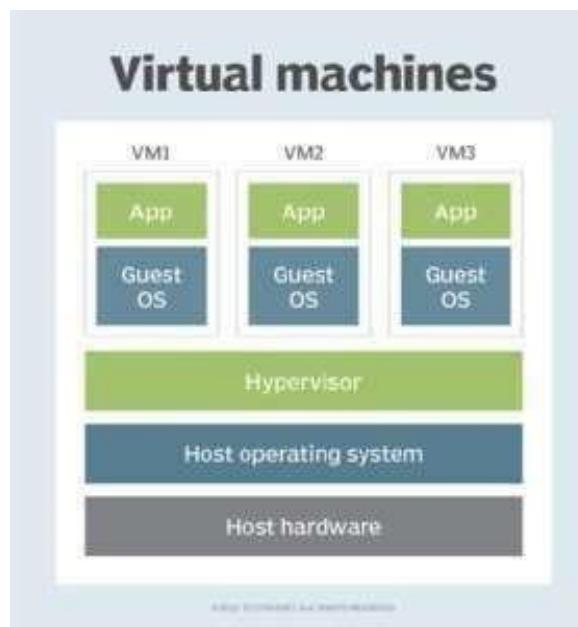


## Guest OS

A guest operating system (guest OS) is an operating system (OS) that is secondary to the OS originally installed on a computer, which is known as the host operating system. The guest OS is either part of a partitioned system or part of a virtual machine (VM) setup. A guest OS provides an alternative OS for a device.

In disk partitioning, a guest OS is simply another instance of the same operating system that can boot up for controlling a certain partitioned memory set. A virtual machine (VM) process is much different, in that a guest OS can be a different OS alternative. In VM setups, a guest OS is delivered through a virtual machine environment through a tool called a hypervisor. Again, the machine will typically have a host OS, where the guest OS will operate "within" the host OS. This can lead to limitations on file saving and other operations within the guest OS, depending on whether the guest OS is said to be "persistent."

Part of the emergence of guest operating systems in VM systems has to do with the benefits provided by virtualization. These revolutions in computing coincide with the more general concept of cloud computing, where resources are delivered, rather than hosted, in physical local hardware setups. In addition, a guest OS often takes advantage of a lean OS build, where memory requirements are further alleviated. VM setups can help with licensing issues, system requirements and more, making these an attractive part of an outsourced computing service.



# **HOST OPERATING SYSTEM VERSUS GUEST OPERATING SYSTEM**

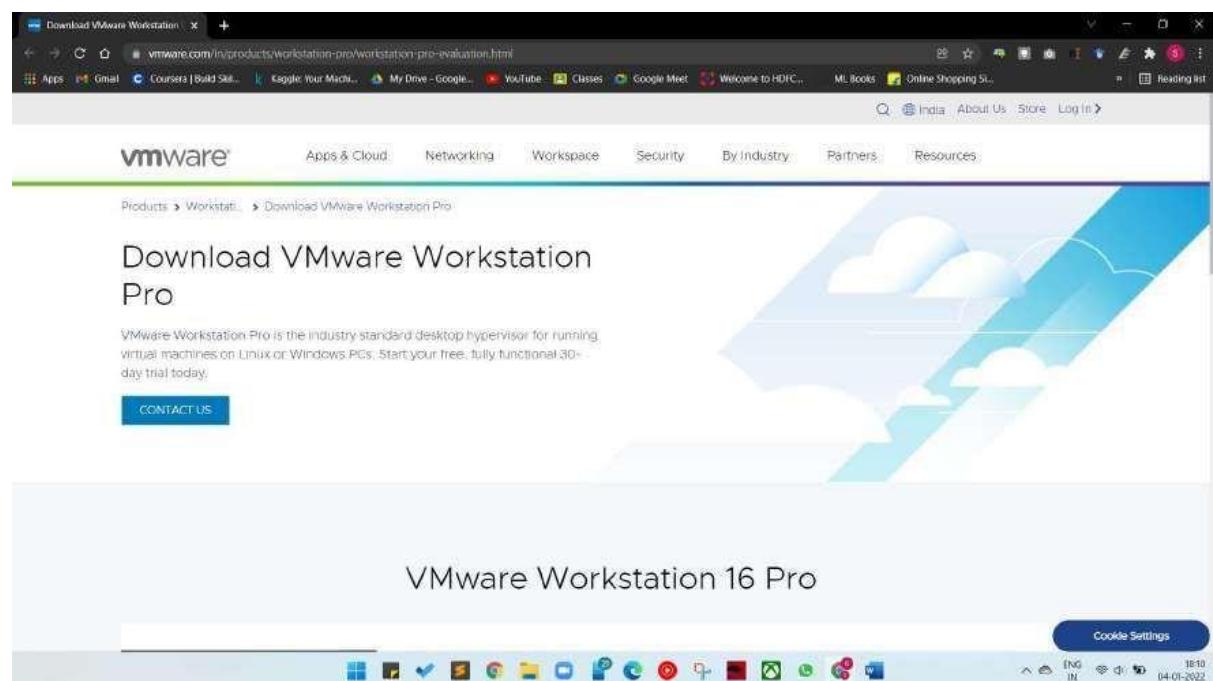
HOST OPERATING SYSTEM	GUEST OPERATING SYSTEM
A software installed on a computer that interacts with the underlying hardware	A software installed in a virtual machine
Host OS interacts on the hardware	Guest OS runs on a virtual machine
There is a single host operating system	There can be a single or multiple guest operating systems

Visit [www.PEDIAA.com](http://www.PEDIAA.com)

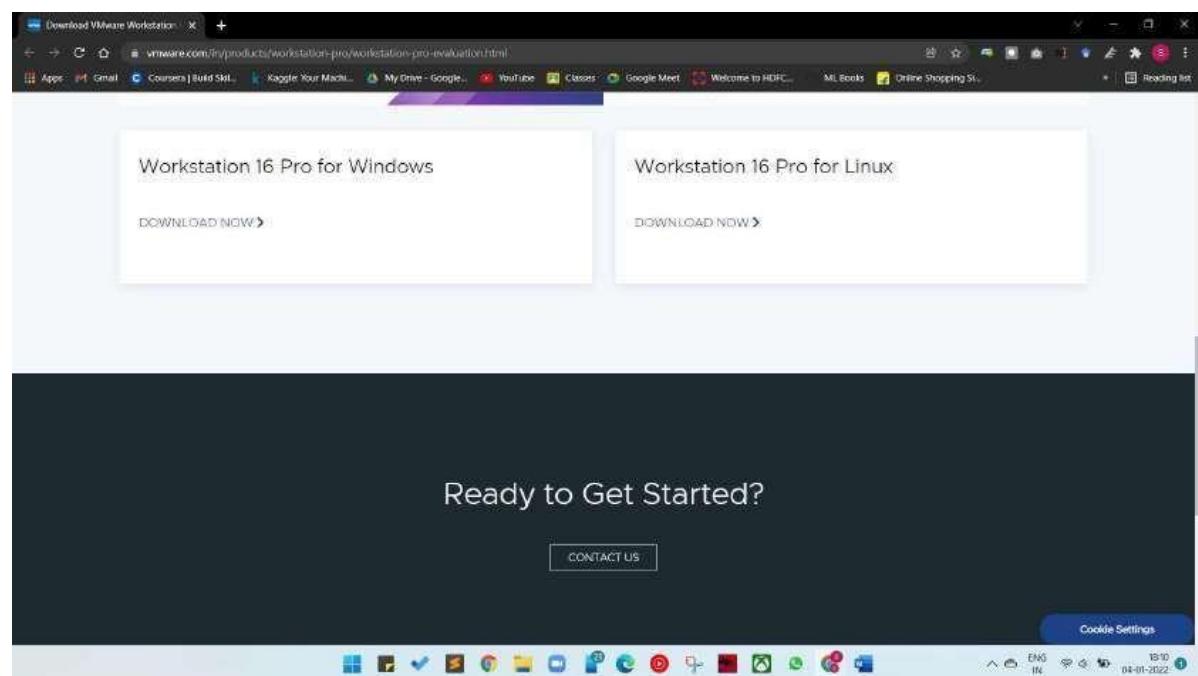
# Install VirtualBox/VMware Workstation with different flavors of Linux and execute some C/C++ programs

**Step 1:** To download and install the VMware product visit the official website of VMware.

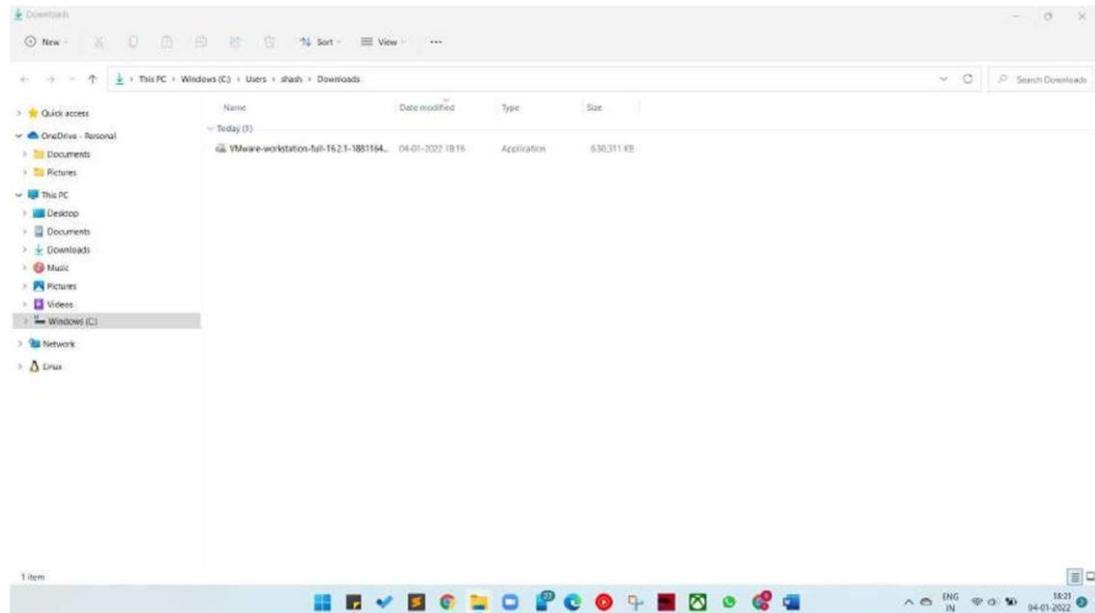
<https://www.vmware.com/in/products/workstation-pro/workstationproevaluation.html>



**Step 2:** Click on Download VM WorkStation for Windows.



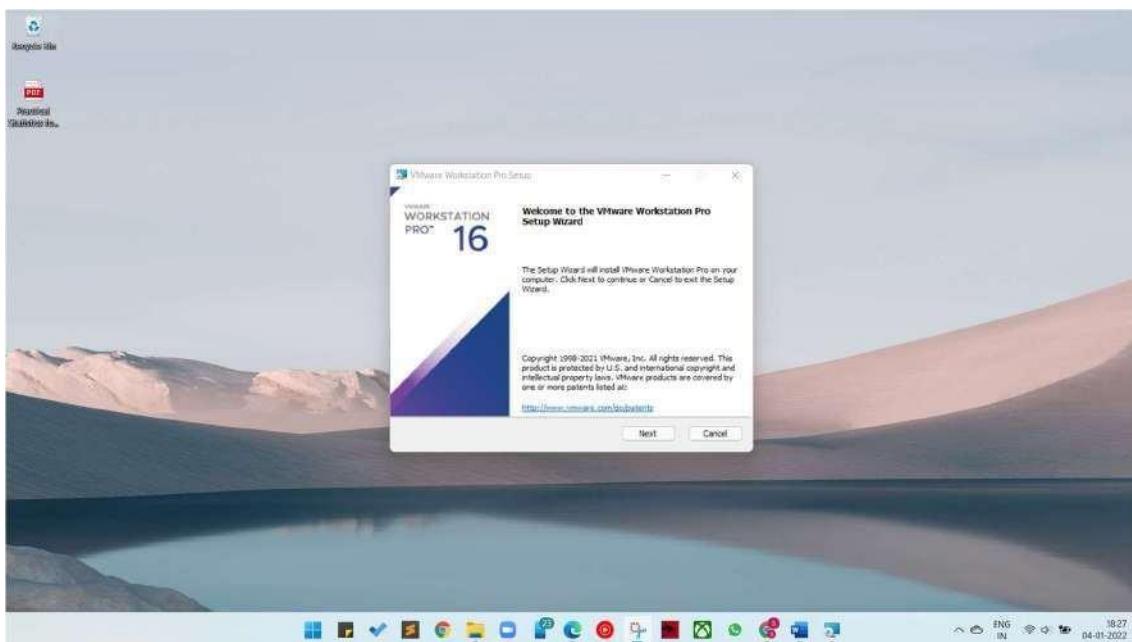
The installation file gets downloaded in the specified location and is now ready for installation.



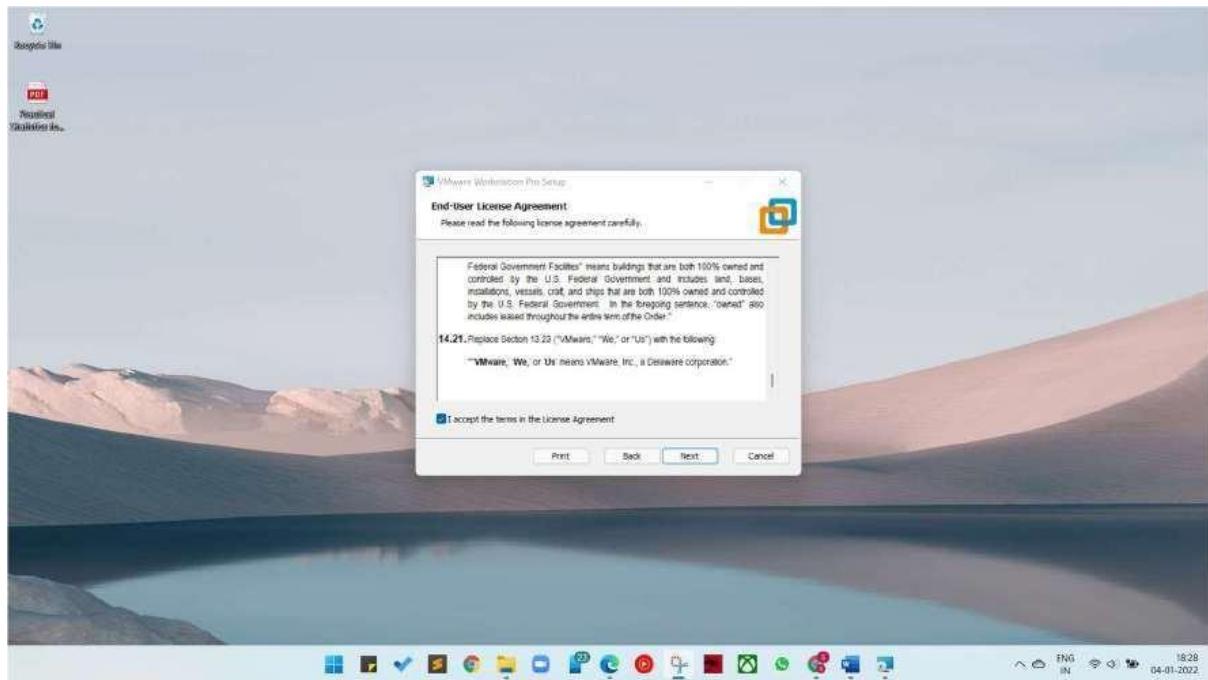
**Step 3:** Click on the download file to install the VMWare Workstation 16 Pro. Popup will appear



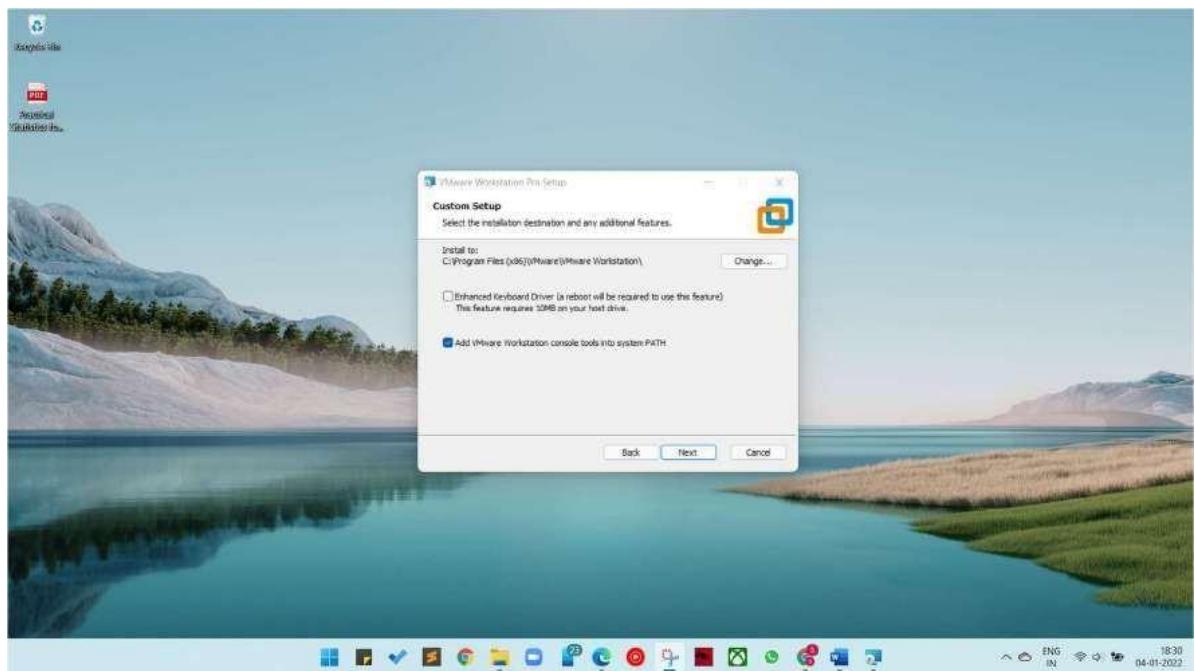
**Step 4:** Once the initialization gets completed, Click Next



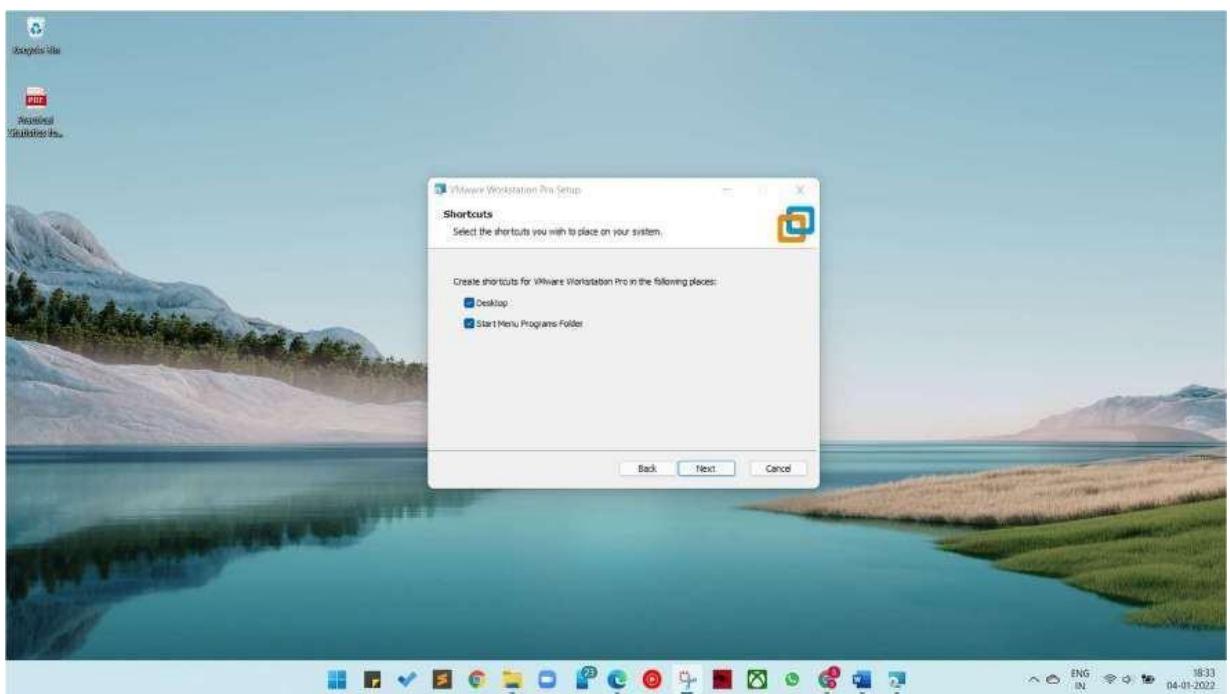
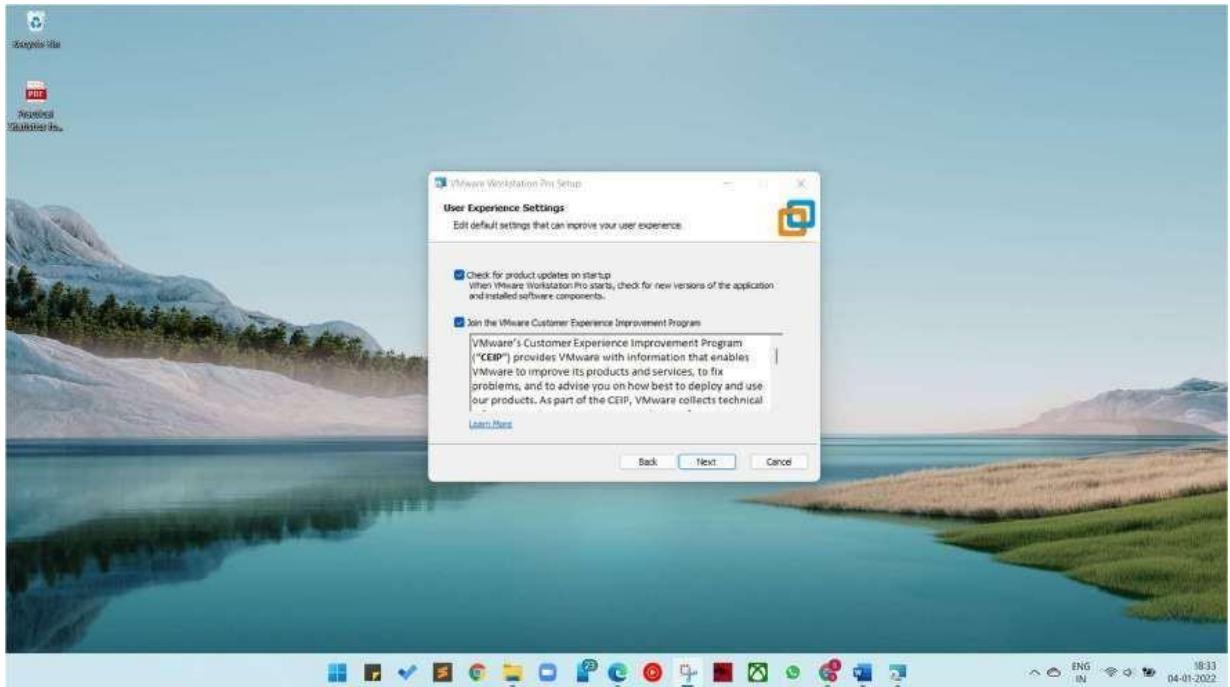
## Step 5: Accept the terms and click on Next



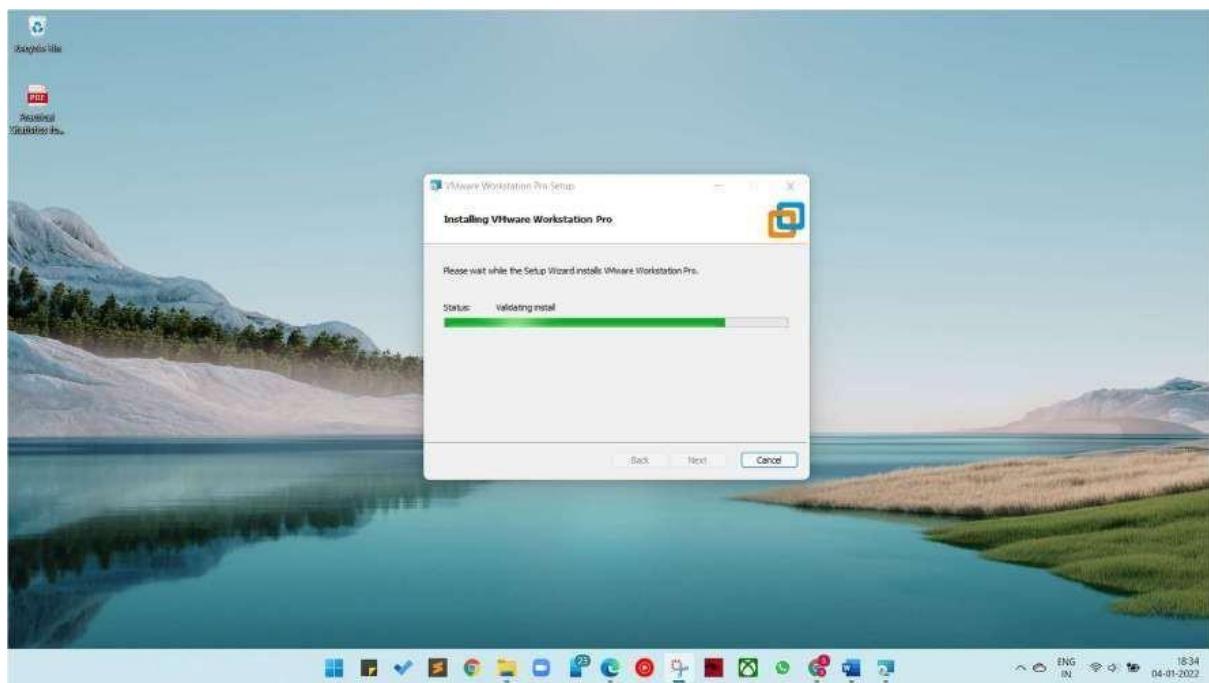
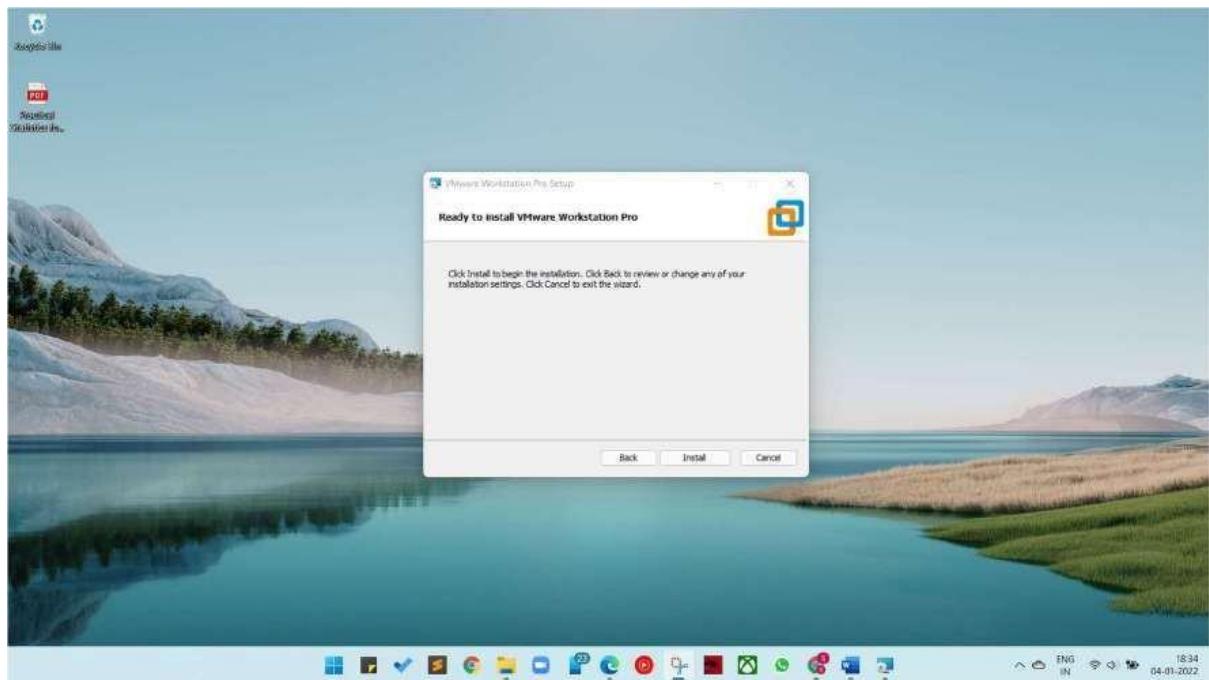
In the next screen, it will ask for some additional features, it is not mandatory to check this box.  
Click on Next.



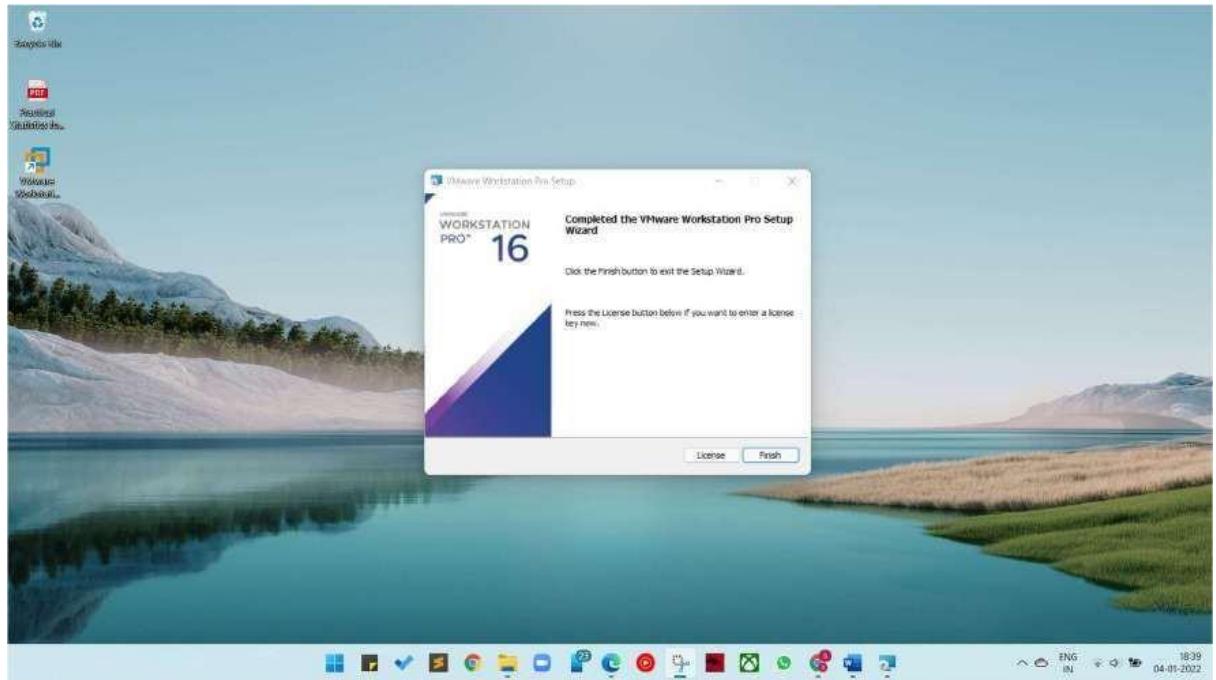
**Step 6:** On the next screen, some checkboxes are populated, Check them as per your requirement. Click on Next.



**Step 7:** At this step, VMware Workstation is ready to install. Click on Install.

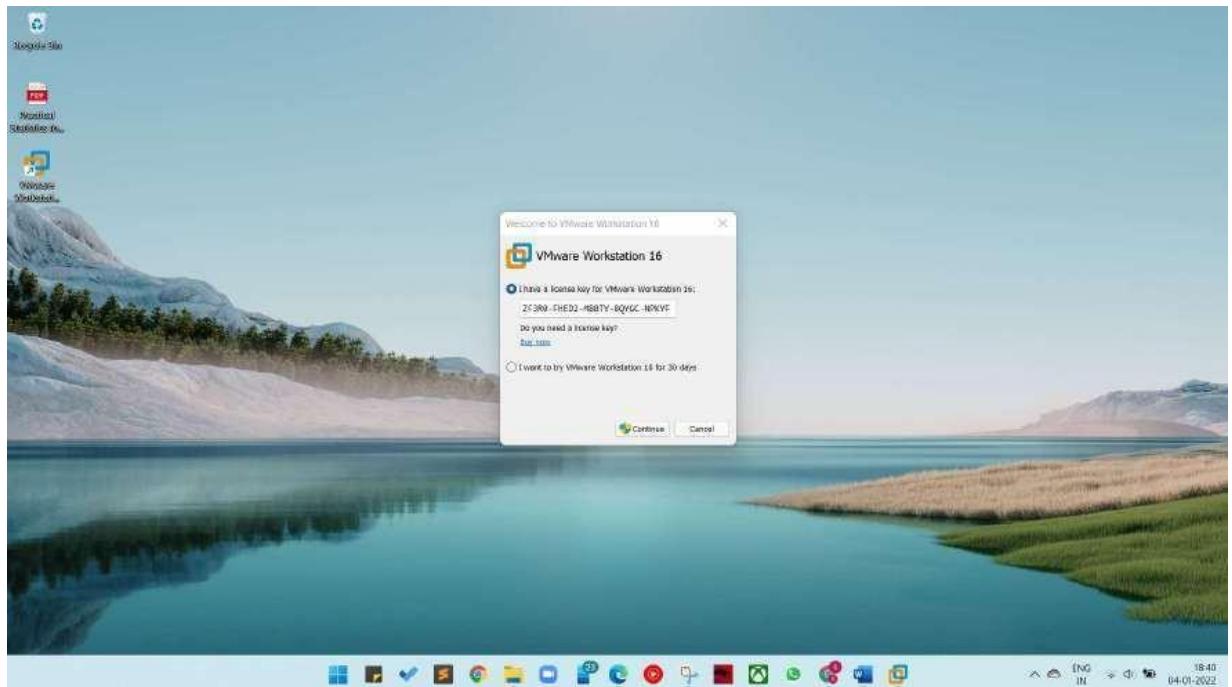


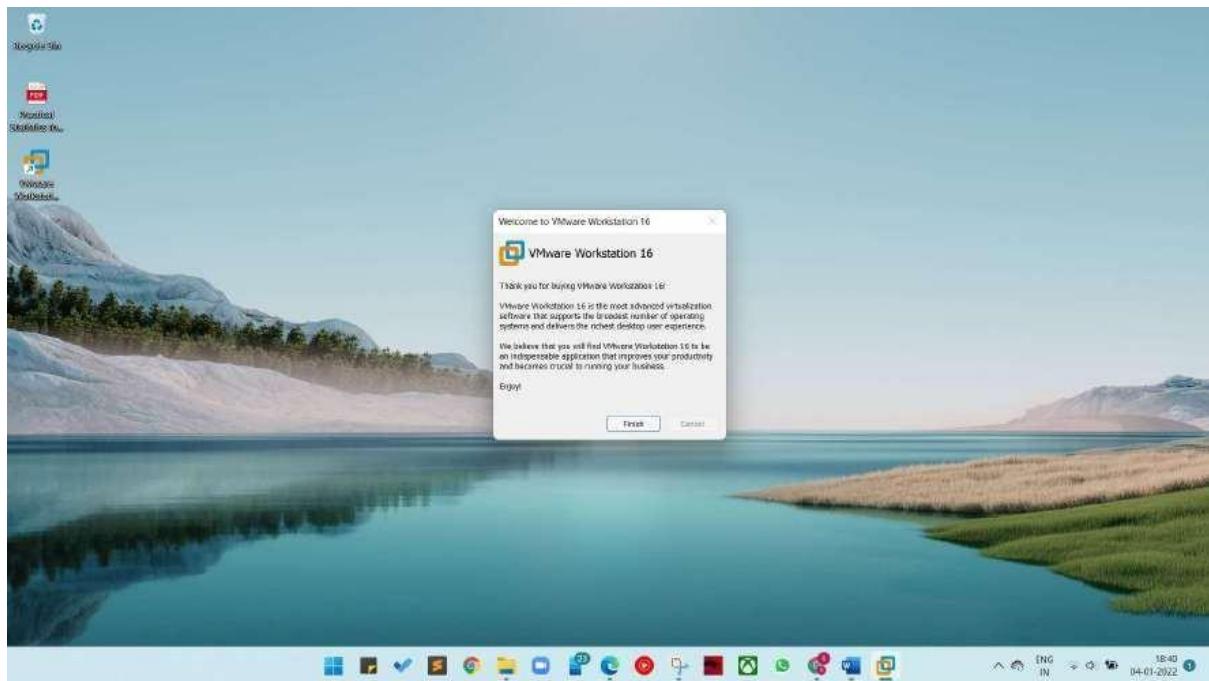
**Step 8:** Once the installation gets completed you will see the following dialogue box. Click on Finish. If you have purchased the product and have a license key, then you can click on License to enter the key.



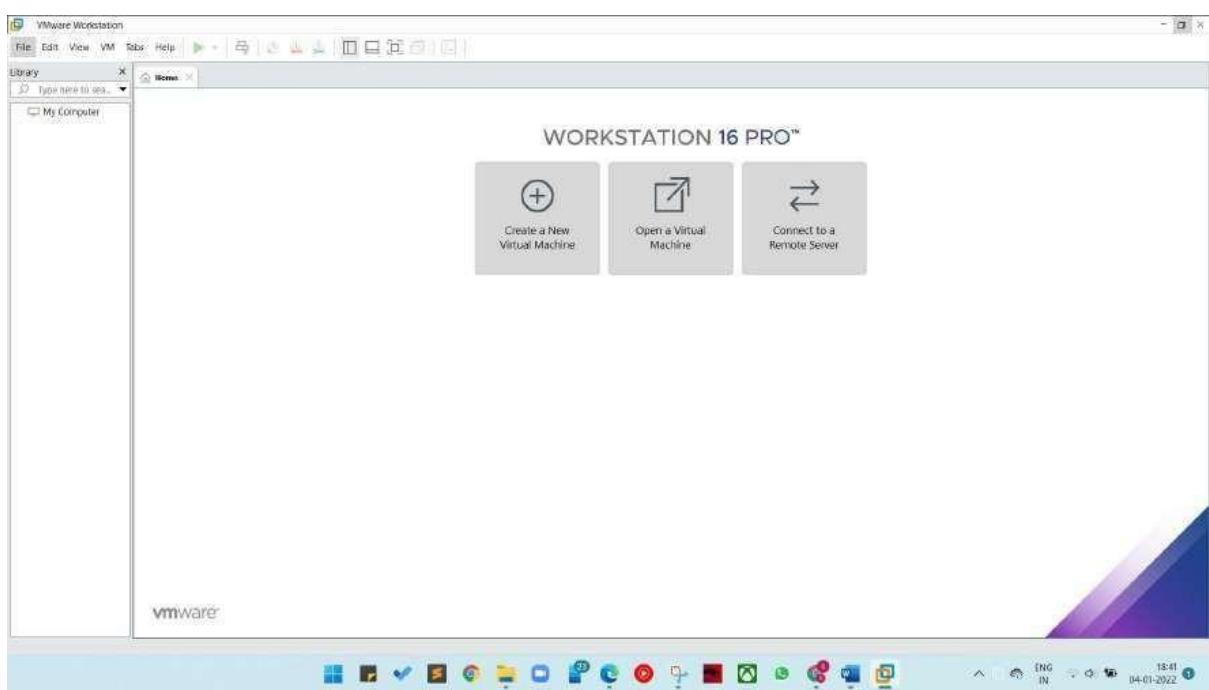
**Step 9:** Upon Finish, the window will close, and You can see VMware Workstation installed icon on your Desktop. Double Click on the Icon to open the application.

**Step 10:** For the first time opening, if you have not entered the License key in the previous steps, then it will ask for a license key.

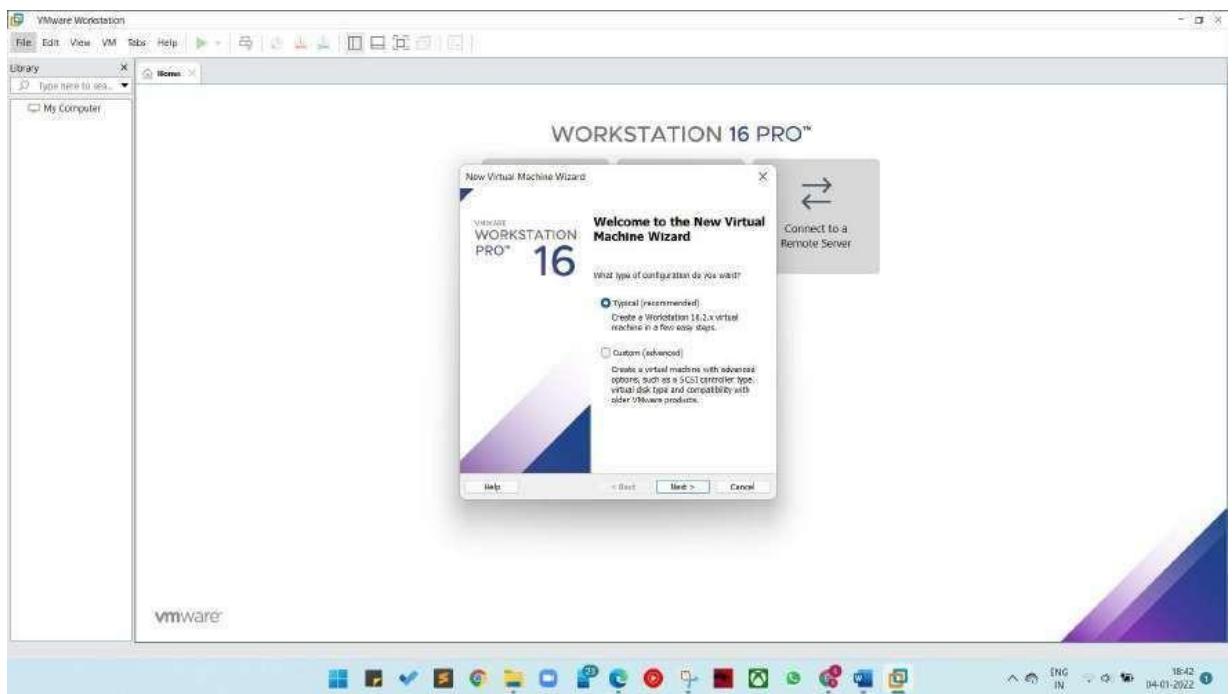




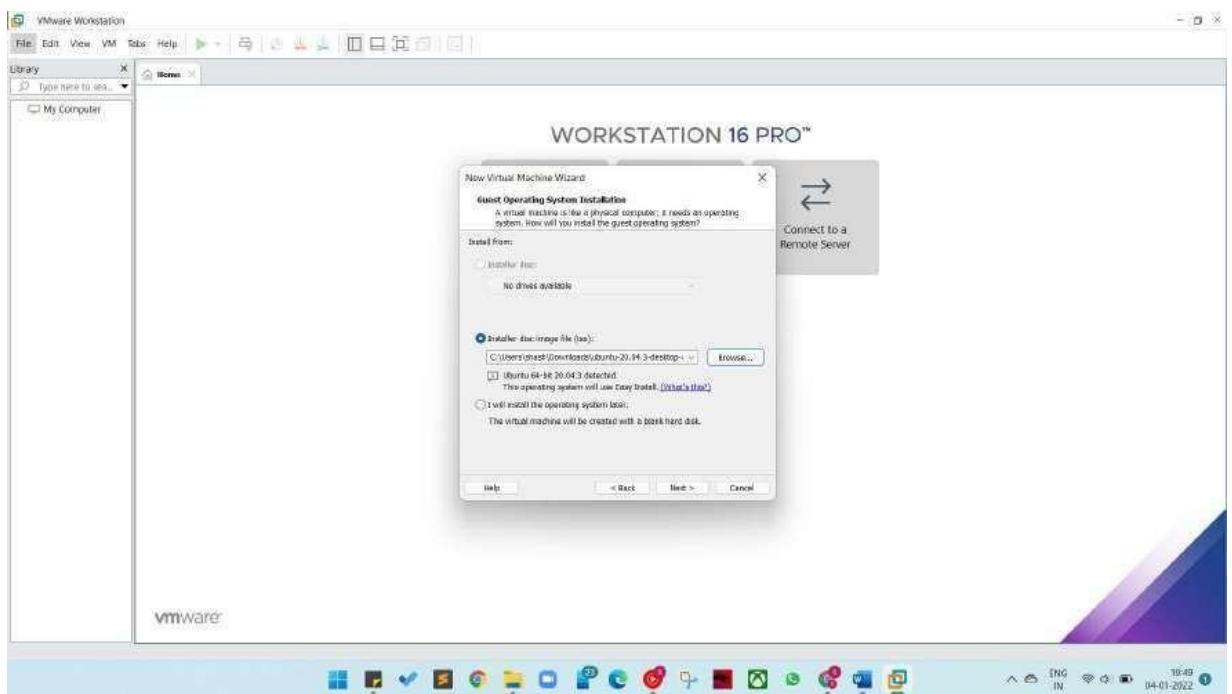
Finally, this will open a window of VMware Workstation Pro.



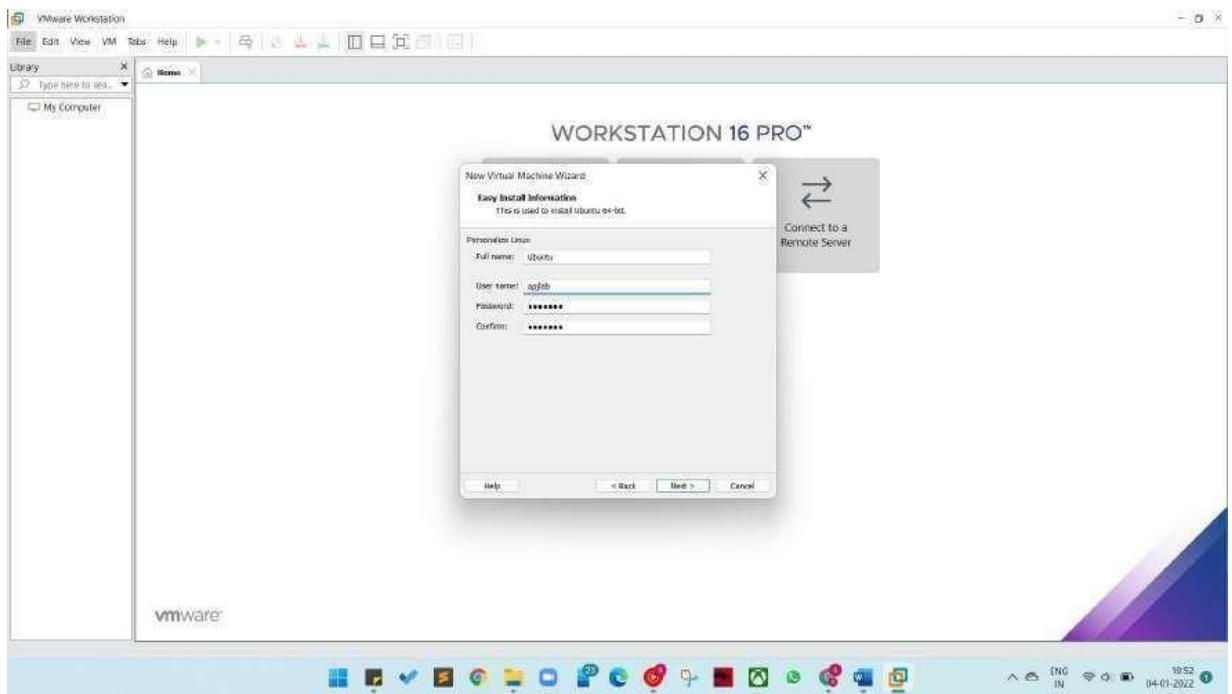
**Step 11:** Click on File New Virtual Machine. A New Virtual Machine Wizard will appear.  
Click on Typical.



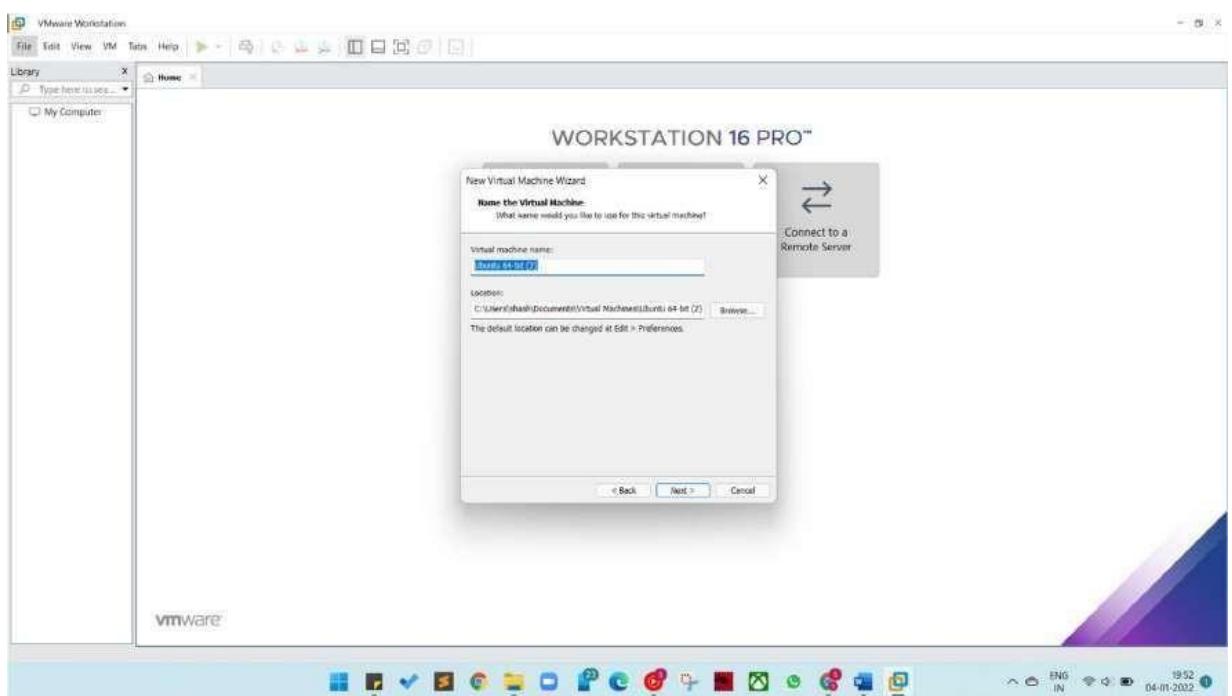
### Step 12: Select the ISO File and click on Next



### Step 13: Fill in the required details such as username, password and click on next.

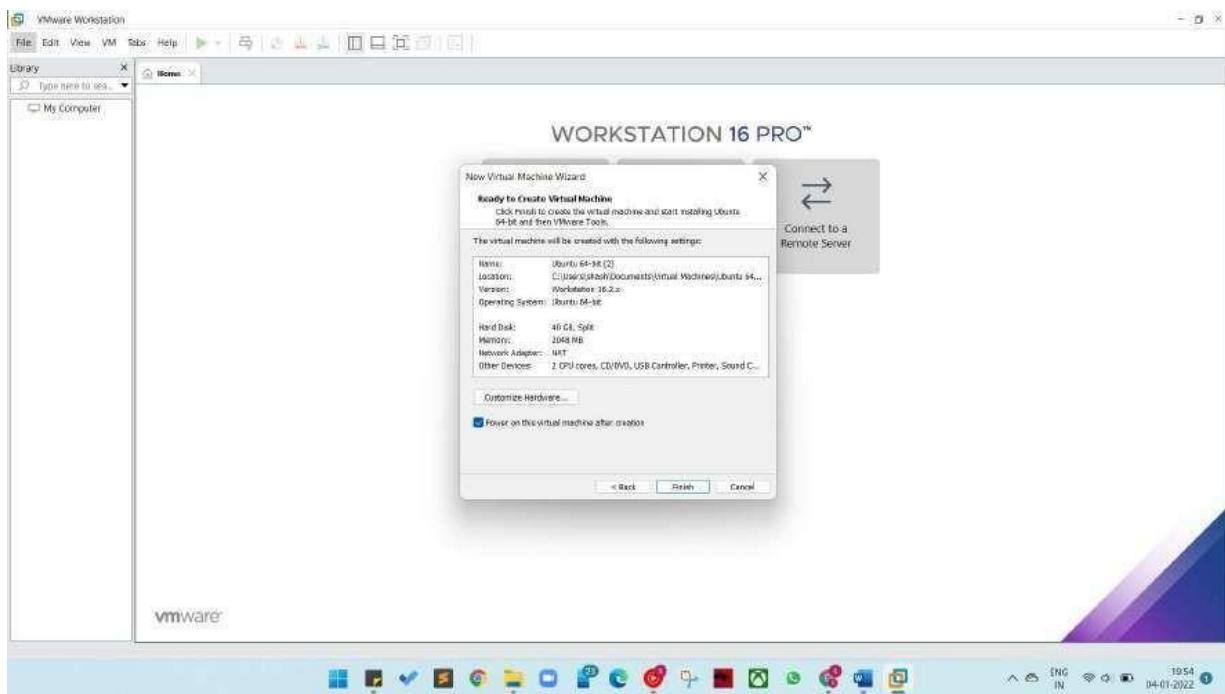


**Step 14:** Name the virtual machines.

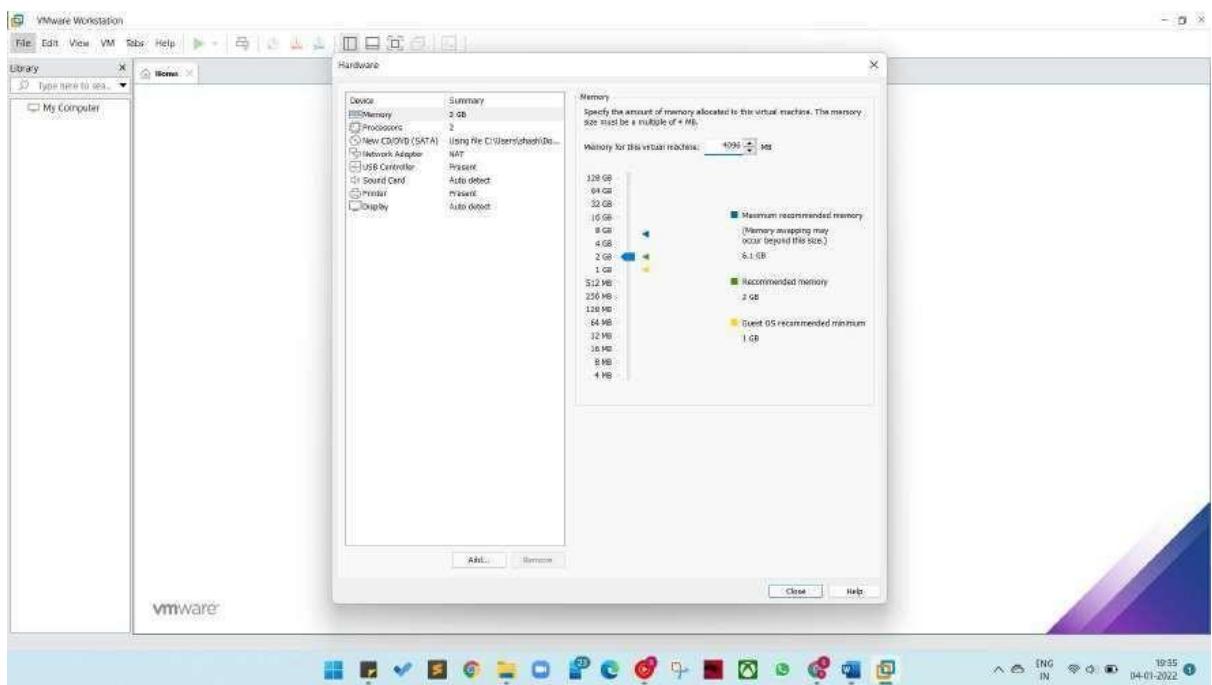


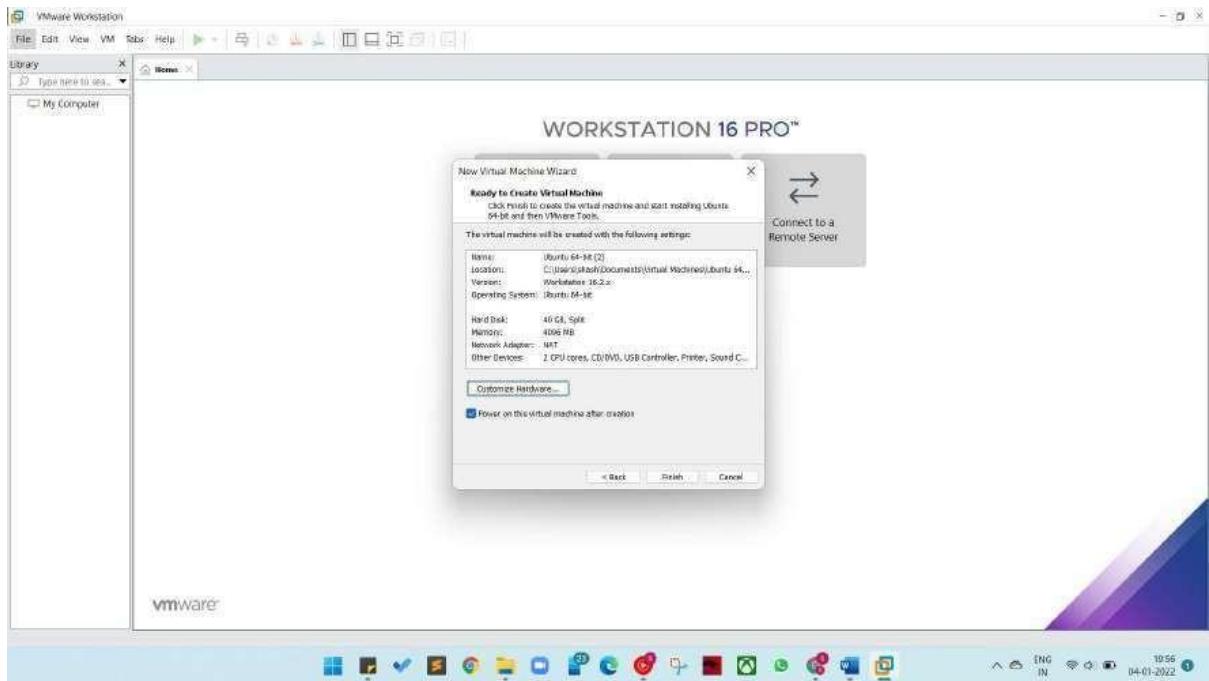
**Step 15:** Allocate the memory and select Split virtual disk into multiple files and click on next.

Click on Customize Hardware.

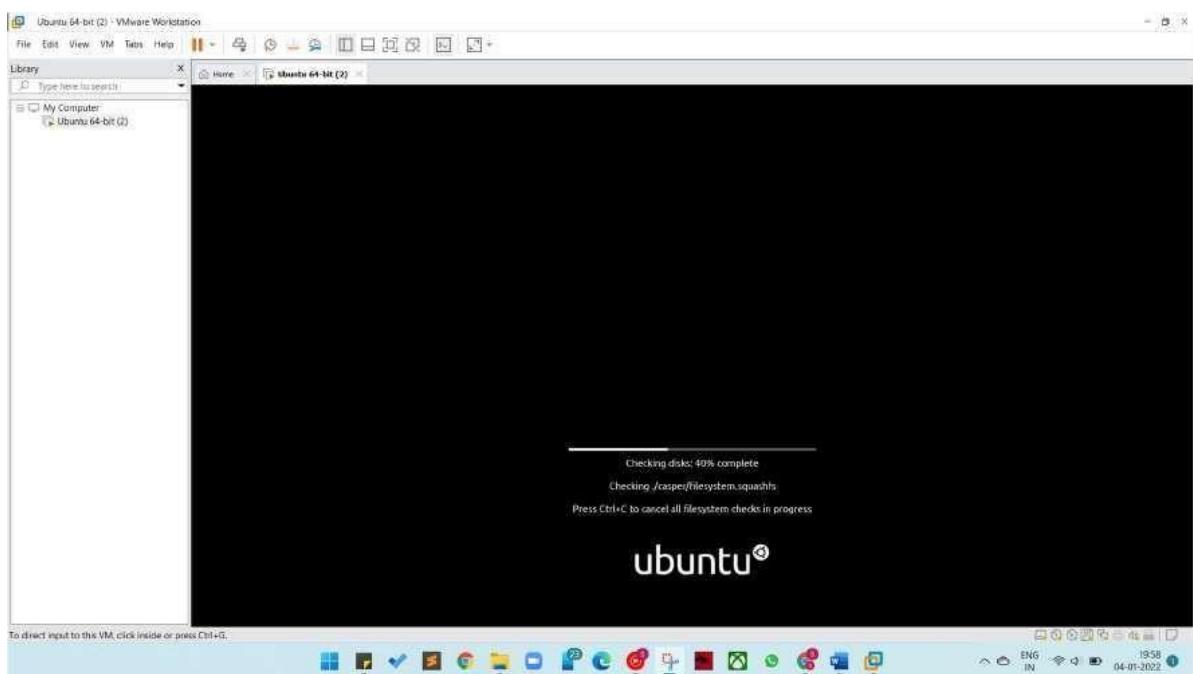


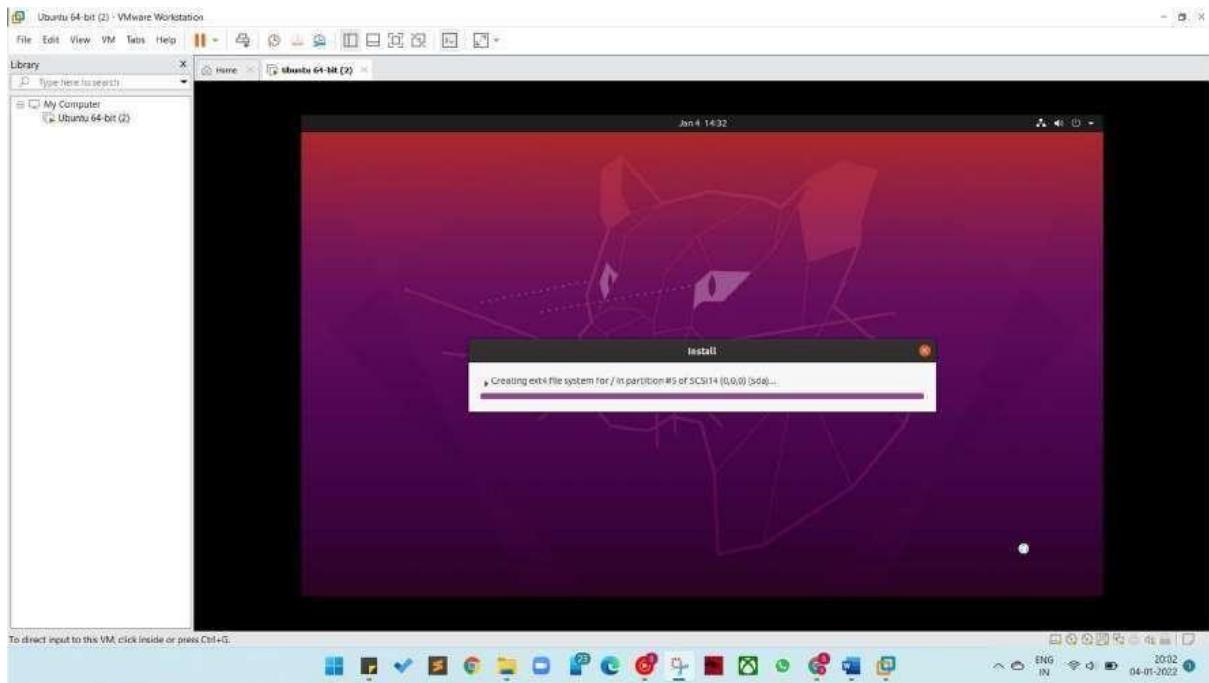
Set the memory size to 4GB



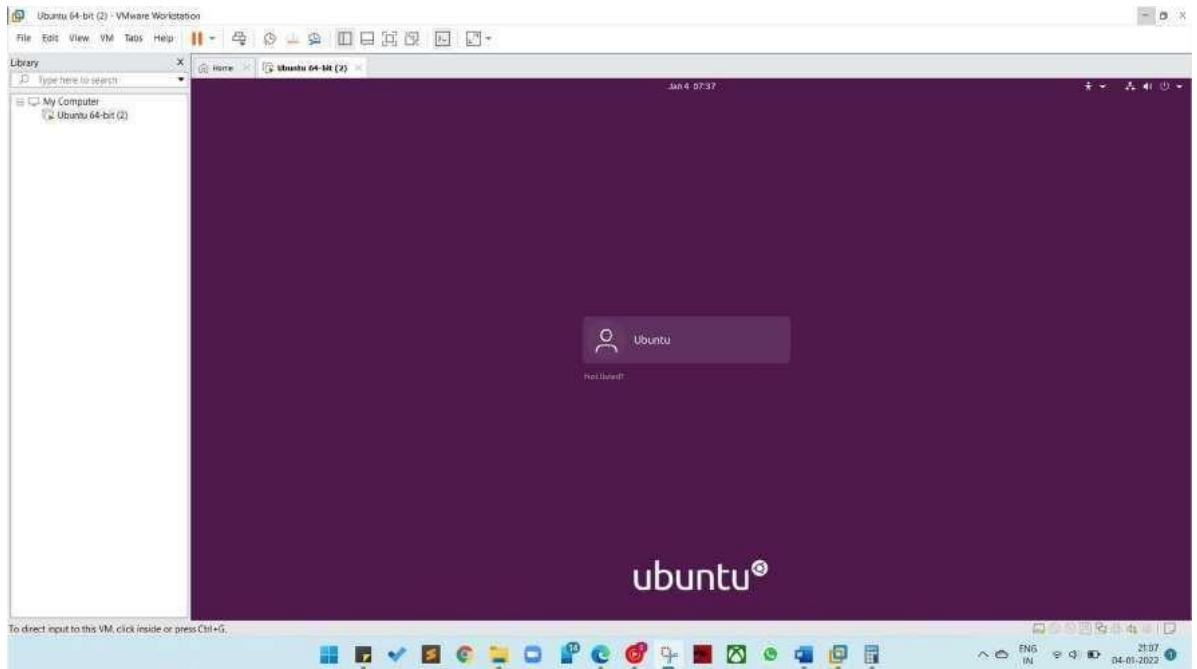


**Step 16:** Click on Finish. You can see that Ubuntu gets installed in the VM Ware Workstation.



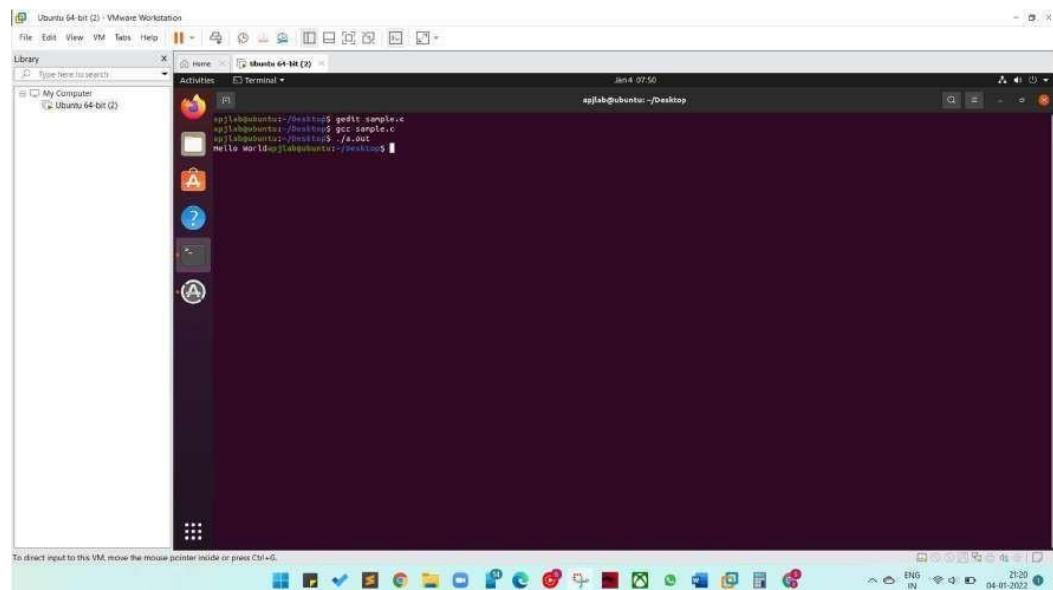
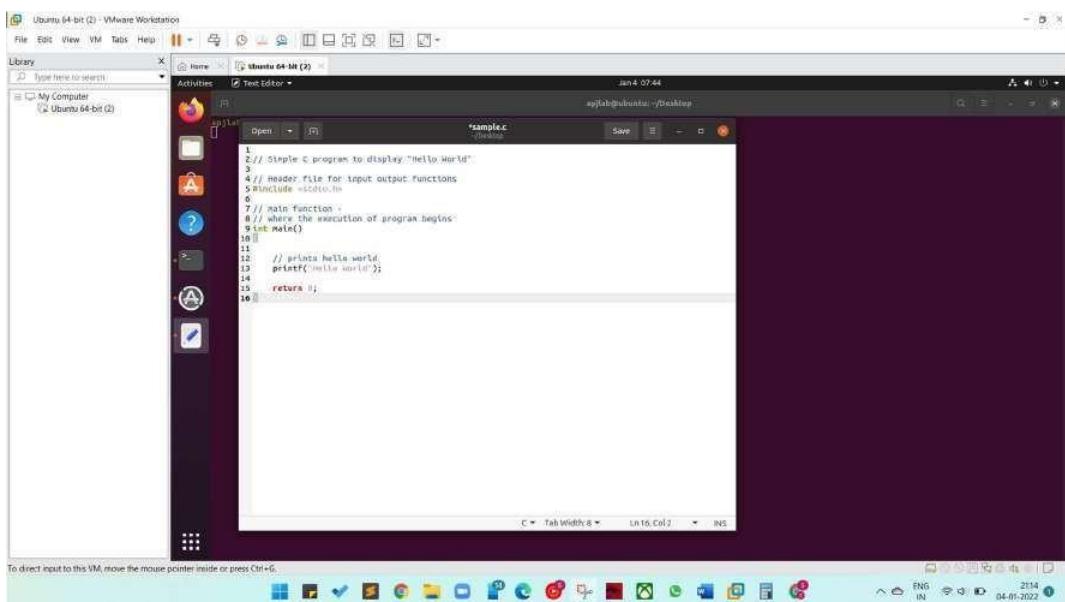
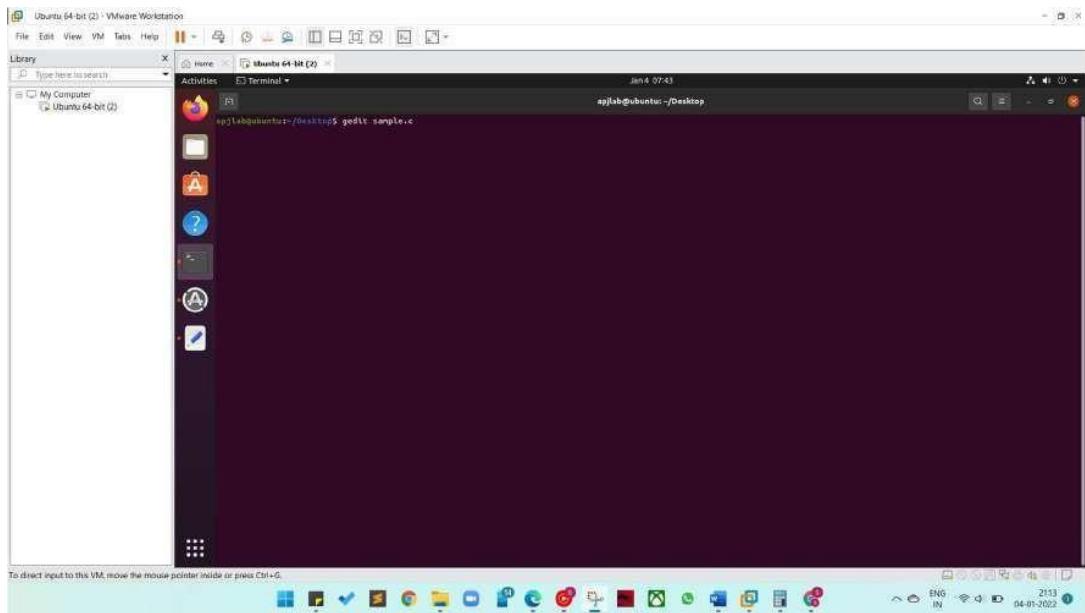


### Step 17: Login and Open Terminal and Execute a simple C Program



### Step 18: Using gedit write a simple C Code. Use the following commands on the terminal:

- To Open the Editor gedit sample.c
- To Compile the code gcc sample.c
- To Get the output ./a.out



# VIRTULIZATION AND FILE SHARING

## Virtualization

Virtualization is the process of running a virtual instance of a computer system in a layer abstracted from the actual hardware. Most commonly, it refers to running multiple operating systems on a computer system simultaneously. To the applications running on top of the virtualized machine, it can appear as if they are on their own dedicated machine, where the operating system, libraries, and other programs are unique to the guest virtualized system and unconnected to the host operating system which sits below it.

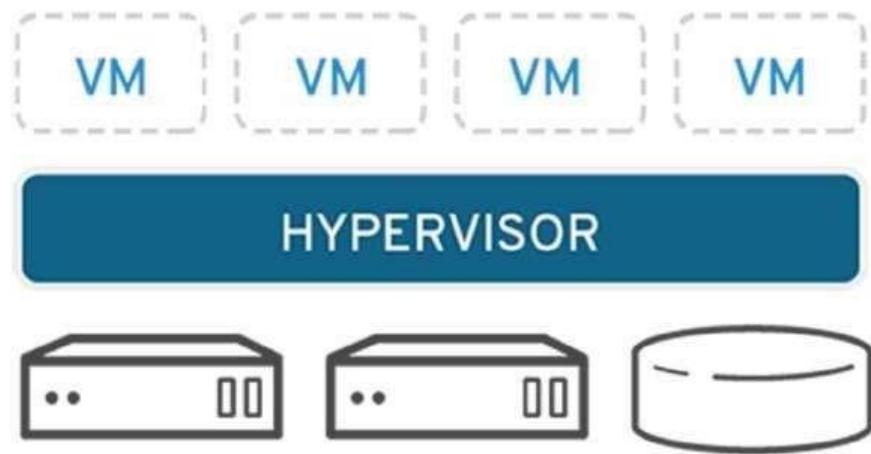
There are many reasons why people utilize virtualization in computing. To desktop users, the most common use is to be able to run applications meant for a different operating system without having to switch computers or reboot into a different system. For administrators of servers, virtualization also offers the ability to run different operating systems, but perhaps, more importantly, it offers a way to segment a large system into many smaller parts, allowing the server to be used more efficiently by a number of different users or applications with different needs. It also allows for isolation, keeping programs running inside of a virtual machine safe from the processes taking place in another virtual machine on the same host.

## Working of Virtualization

Software called hypervisors separate the physical resources from the virtual environments—the things that need those resources. Hypervisors can sit on top of an operating system (like on a laptop) or be installed directly onto hardware (like a server), which is how most enterprises virtualize. Hypervisors take your physical resources and divide them up so that virtual environments can use them.

Resources are partitioned as needed from the physical environment to the many virtual environments. Users interact with and run computations within the virtual environment (typically called a guest machine or virtual machine). The virtual machine functions as a single data file. And like any digital file, it can be moved from one computer to another, opened in either one, and be expected to work the same.

When the virtual environment is running and a user or program issues an instruction that requires additional resources from the physical environment, the hypervisor relays the request to the physical system and caches the changes—which all happens at close to native speed (particularly if the request is sent through an open-source hypervisor based on KVM, the Kernel-based Virtual Machine).



## **FILE SHARING**

Transferring Files to and from Virtual Machines can be done in the following ways:

1. Creating a Shared Folder in VirtualBox
2. Dragging and Dropping Files in VirtualBox
3. Managing Files with NextCloud

### **1. Creating a Shared Folder in VirtualBox**

A shared folder is a folder that makes its files available on both the guest machine and the host machine at the same time. Creating a shared folder between the guest and the host allows you to easily manage files that should be present on both machines. The course virtual machines are ready to use shared folders right away, but if you are using the virtual machine on your personal computer, then you will need to specify which folder to use as shared storage.

**Shared Folders on SCS Lab Computers using Course VMs:** If you are using a course VM on a lab computer, it is likely that a shared folder has already been set up for you. On the desktop of your course VM, you should notice a folder titled Shared Folders. Inside this folder, you will find any folders that have been shared between the course VM and lab computers. You should see two folders that have already been configured for you: Z\_DRIVE and Temp. Z\_DRIVE gives you access to your Windows Account Z:\ drive. This is storage that is persistent to your SCS account and available as a network drive on the lab computers. Temp gives you access to the folder found at D:\temp on the lab computer. Files stored in this folder are local to the machine, meaning that they can be accessed faster but will delete from the system when you log out. If you are working with data that you will need to use again, use the Z\_DRIVE for your shared folder. If you need a faster read/write speed, use the Temp folder, but remember to back up your files, or they will be deleted when you log off the computer.

## **2. Shared Folders on Personal Computers**

If you are using your own personal machine, you will need to configure VirtualBox to look in the right place for your shared files. First, click on the guest machine you intend to share files with. From there, you can select the guest Settings and navigate to Shared Folders on the left side menu. To create a new shared folder, either click the New Folder icon on the right menu or right-click the empty list of shared folders and click Add Shared Folder. From here, there are six options:

- Folder Path: The folder name on the host machine. Click the drop-down menu and navigate to the folder you would like to share.
- Folder Name: This is the name of the folder as it will appear on the guest machine.
- Read-Only: If you check read-only, the guest machine will be unable to write changes to the folder. This is valuable when you only want to send files to the virtual machine, but do not want to risk having the files modified by the guest.
- Auto-Mount: When any external storage is connected to a computer, it must be mounted in order to be used. It is recommended that you turn on auto-mounting unless you are familiar with the process of mounting a drive yourself.
- Mount Point: Unless you already know about mount points, leave this blank.
- Make Permanent: If you check this, the shared folder will be a permanent machine folder. If it is not checked, the folder will not be shared after a shutdown.

On the course virtual machines, when you load into the desktop, you should see a folder labelled Shared Folders. In there, you will see any folders that are currently mounted and being shared.

### **3. Dragging and Dropping Files in VirtualBox**

If you only need to transfer a few files quickly, you can simply drag and drop the files in. On the top bar of the running guest machine, click on Devices > Drag and Drop and make sure that Bidirectional is selected. This means that you will be able to drag files from the host to the guest and from the guest to the host. Once bidirectional drag and drop is checked, you should be able to begin dragging and dropping files.

NOTE: Sometimes when dragging files into the course VM, you may not be able to drag them into the file browser directly. If you encounter this issue, you should drag your files onto the Desktop and move the files around from there. You should see the cursor change when it is ready to drop files.

You can also drag files from the guest machine into the host. To do this, simply open the file browser on the host to where you would like to drop the files and drag the files from the virtual machine into the file browser of the host. File transfers should be pretty quick; if the virtual machine seems stuck when transferring, simply cancel the transfer and try again.

## **Managing Files with NextCloud**

On any virtual machine, including VirtualBox, VMWare, or the virtual machines hosted on the SCS OpenStack, you can access the SCS NextCloud services to move files between multiple machines and your SCS Windows Account storage. NextCloud offers you all of your SCS storage in one remote location, similar to how you might use other file hosting services like Dropbox or Google Drive. Before trying to use NextCloud, you should check that you can access the service by logging in here. If you can access the NextCloud services, you can browse the various file storage services available to you:

Linux Home: These are the files from your SCS Linux Account

Windows Home: These are the files from your SCS Windows Account and your lab Z:\ drive.

NextCloud: In addition to the other storage accounts provided to you by the SCS, you can also upload up to 20GB of files directly to NextCloud.

With NextCloud, you can upload your files from any machine with an internet connection and download them onto any other machine with an internet connection. For example, you can move project files off of your virtual machine, onto the NextCloud storage, and then download them on your personal laptop. Alternatively, you can upload files from your personal PC onto the NextCloud storage, place them into the Windows home folder, and access those files from either the lab Z:\ drive or download them on a virtual machine like VirtualBox or OpenStack.

### **Uploading Files to NextCloud from a Lab Computer**

If you would like to upload files from a lab computer, the easiest way to do this is to place the files you would like to transfer into your Z:\ drive. These files will be automatically backup into your NextCloud storage under the Windows home folder. After that, you can move them into the main NextCloud storage or choose to keep them in your Z:\drive.

### **Uploading Files to NextCloud from a VM or Other PC**

If you would like to upload files from either a VM or any other computer, you can log in to the NextCloud service using any of the available interfaces, such as the web interface. Press the “+” icon in the top left of the file browser and select Upload File. From here, you can choose to keep it in the main NextCloud storage, move it into your Windows Account storage (the Windows home folder), or into your Linux Account storage (the Linux Home folder).

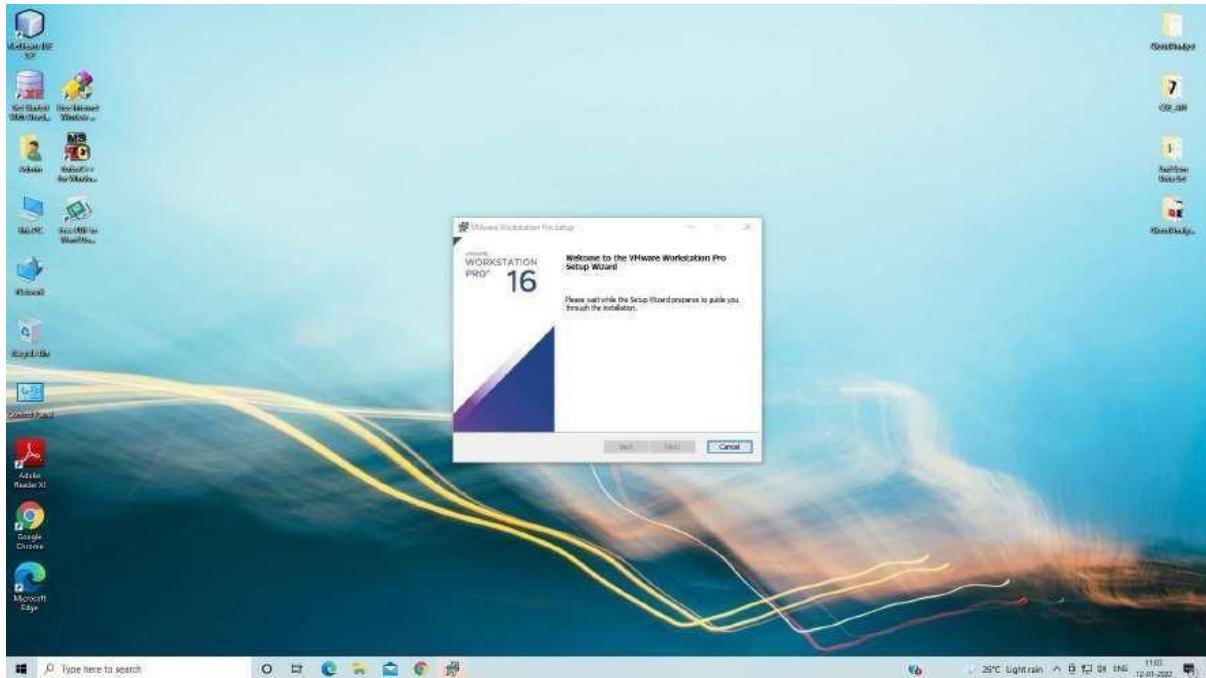
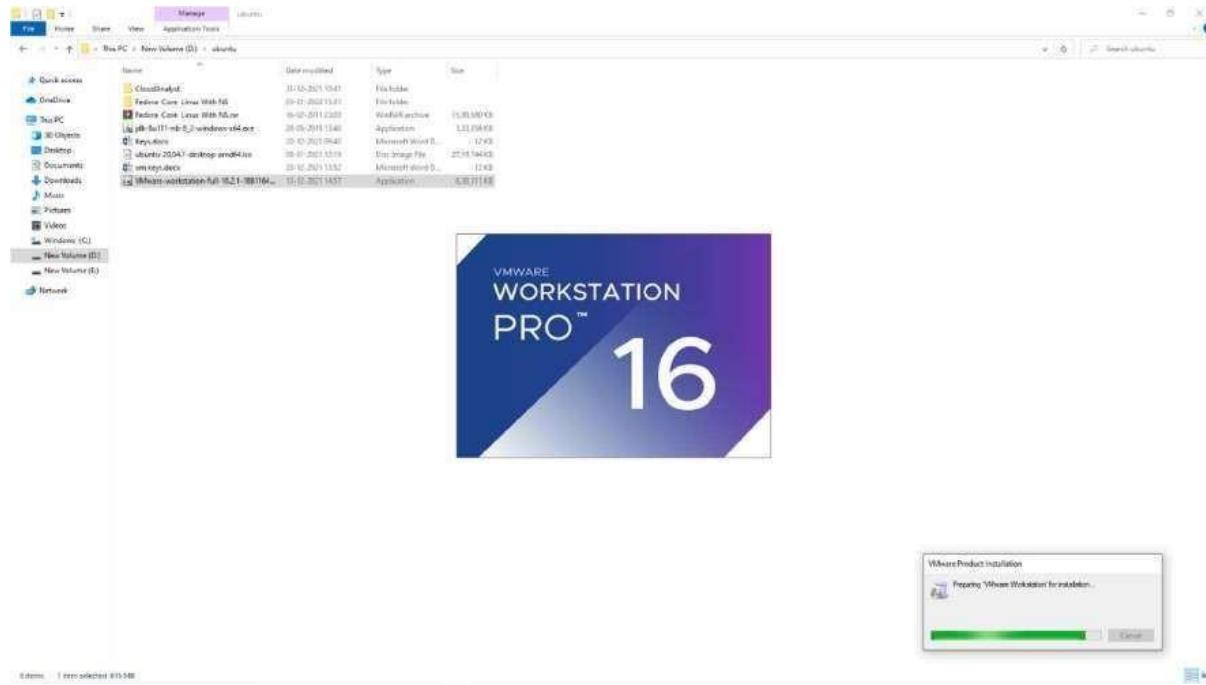
### **Downloading NextCloud Files to a VM or Other PC**

Once your files are uploaded, you will be able to download those files onto any machine, which can connect to NextCloud. First, log in to your preferred NextCloud interface (e.g., the web interface). Navigate to the folder which contains the files you would like to download. Once you are in the target folder, click the checkbox next to each file you would like to download. Above the file listing, you should notice the context bar changing to tell you how many files you have selected and a button labelled Actions. Click Actions > Download.

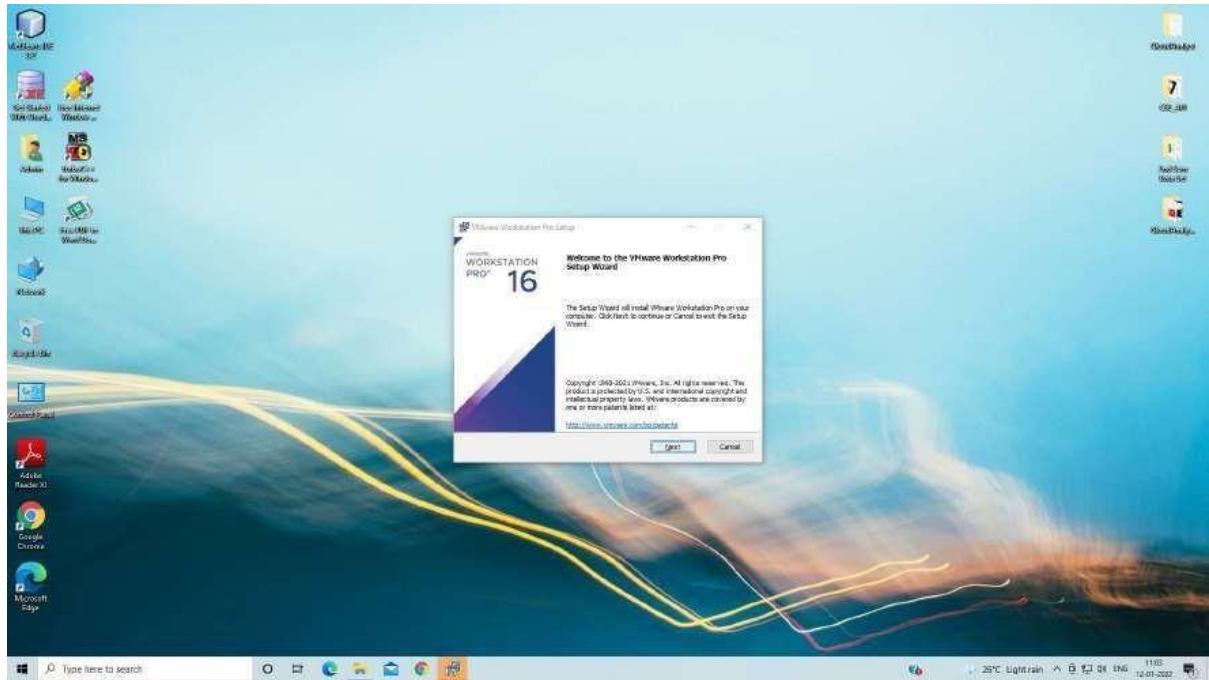
If you selected a single file, it will prompt you to confirm the download. If you have chosen more than one file, NextCloud will place all of the selected files into a zip archive. Before you can use the files, you will need to extract them from the archive. Once you have downloaded your file or extracted your archive, you are ready to use your files on your machine

## Create a file in one virtual machine and share it on a host machine

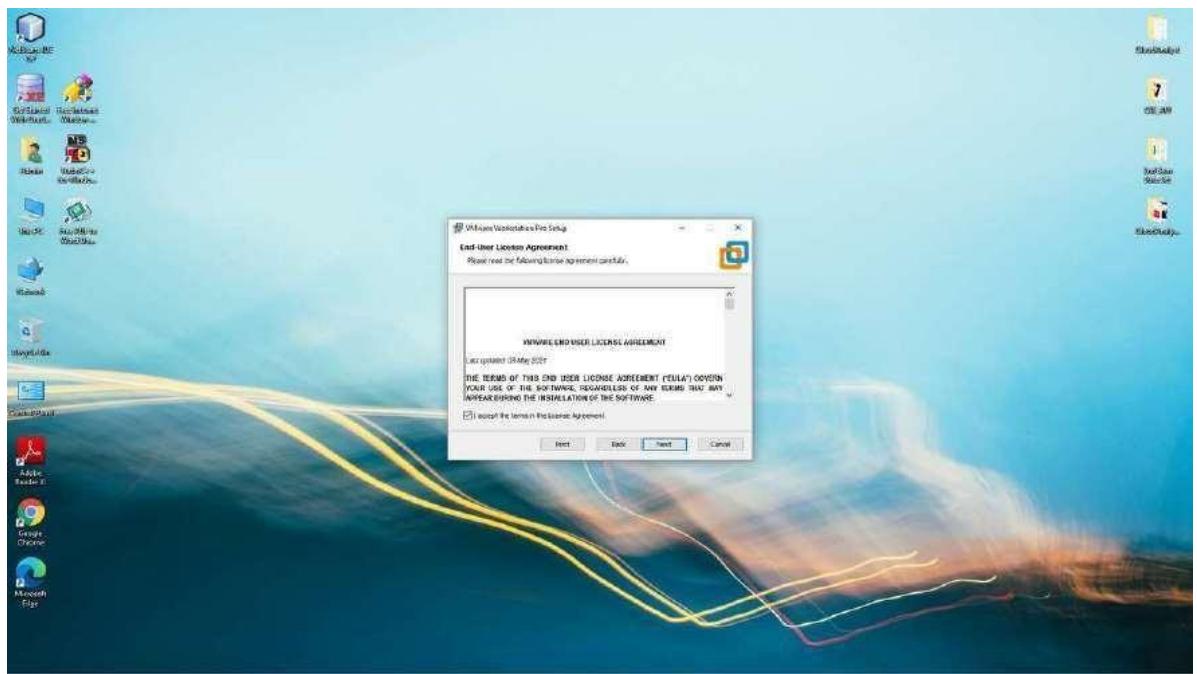
**Step 1:** Install VM Work Station Pro 16 by clicking on the application file.



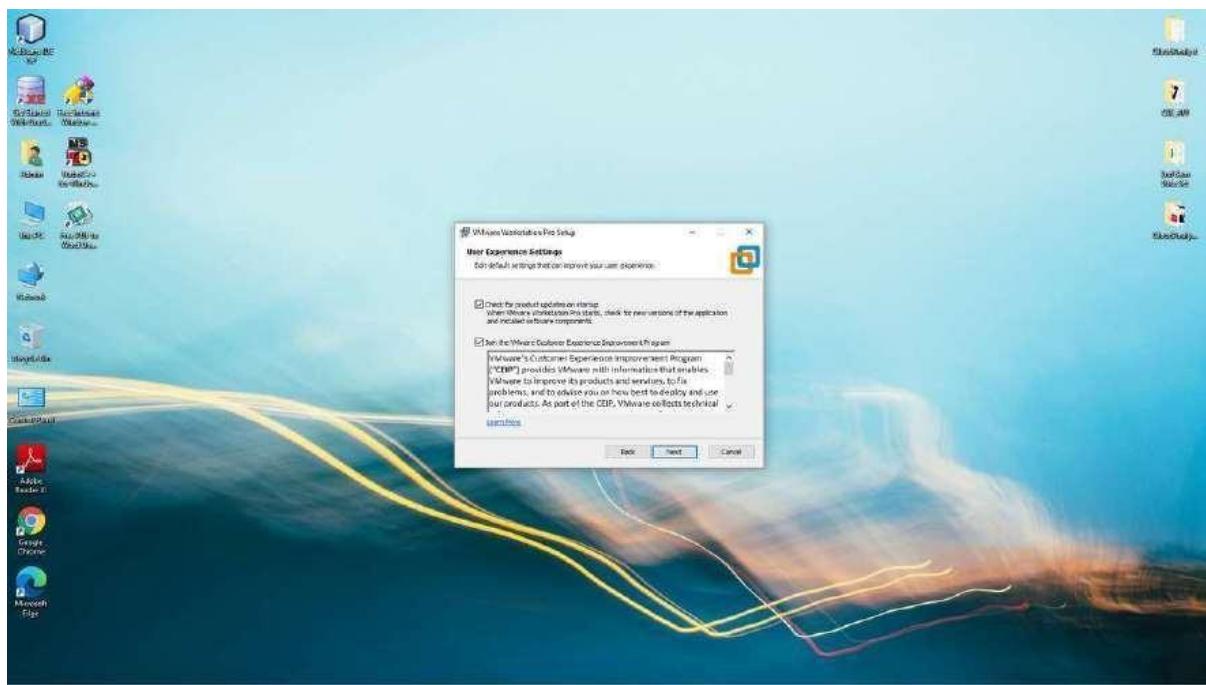
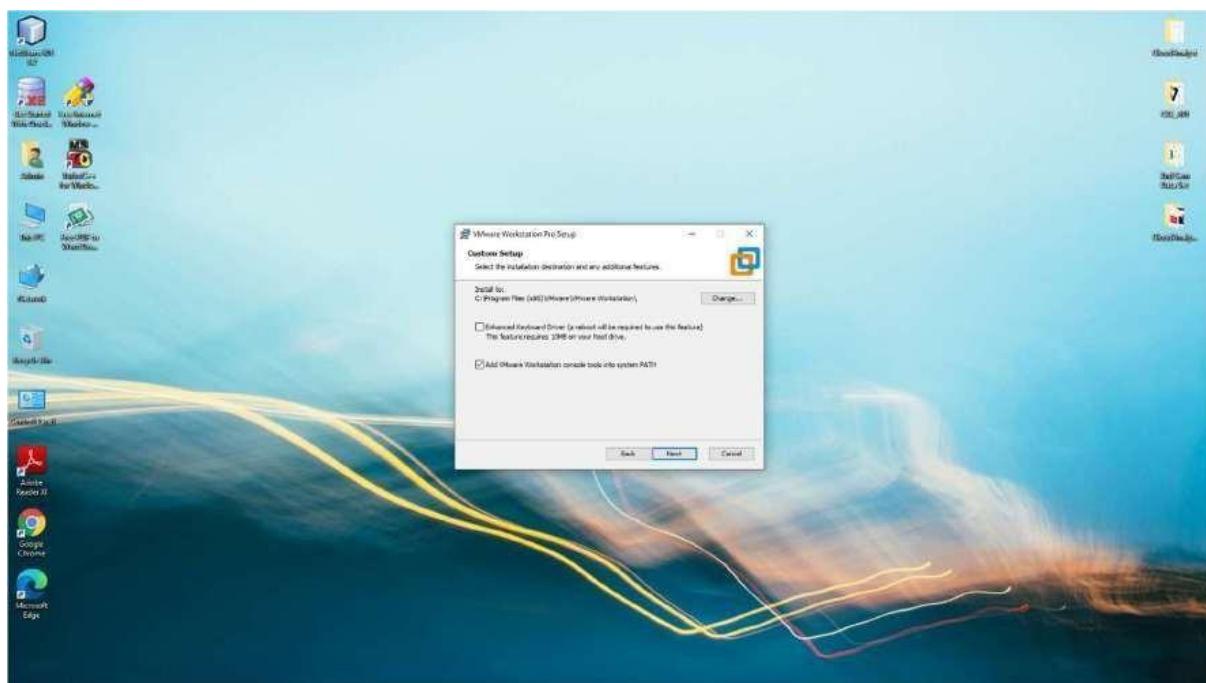
A window appears as shown in the picture. Click on the Next and follow the installation steps.

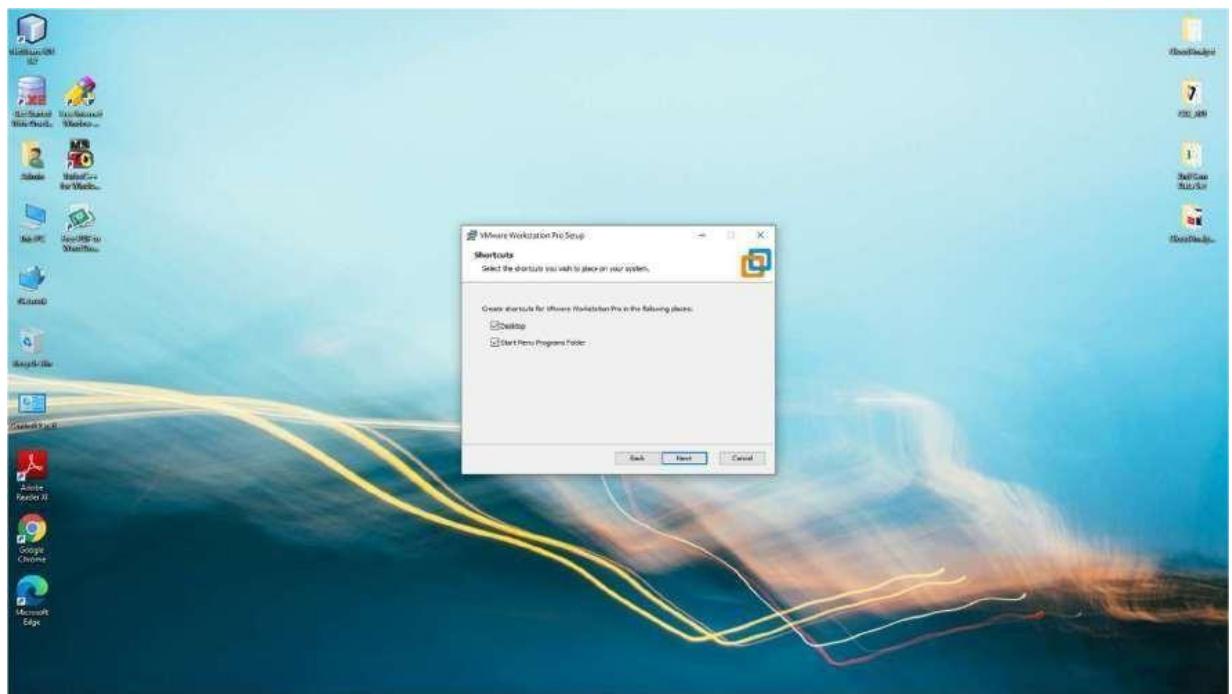


Accept the terms to continue the installation.

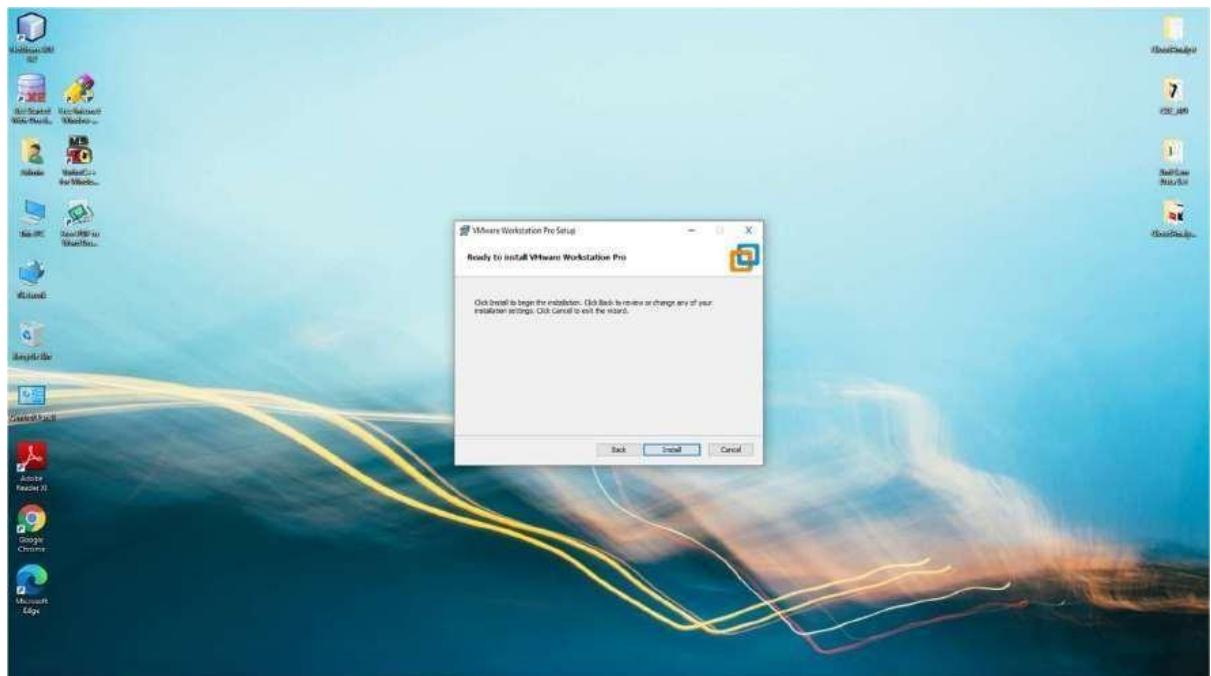


Click on Next.

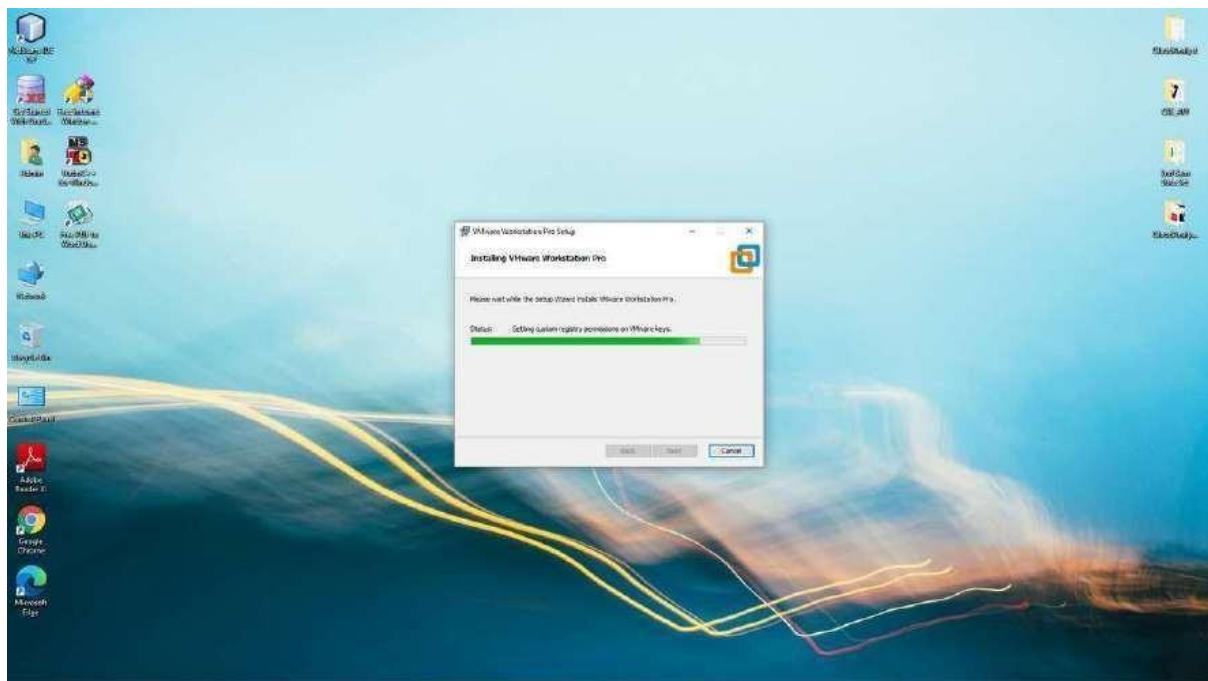




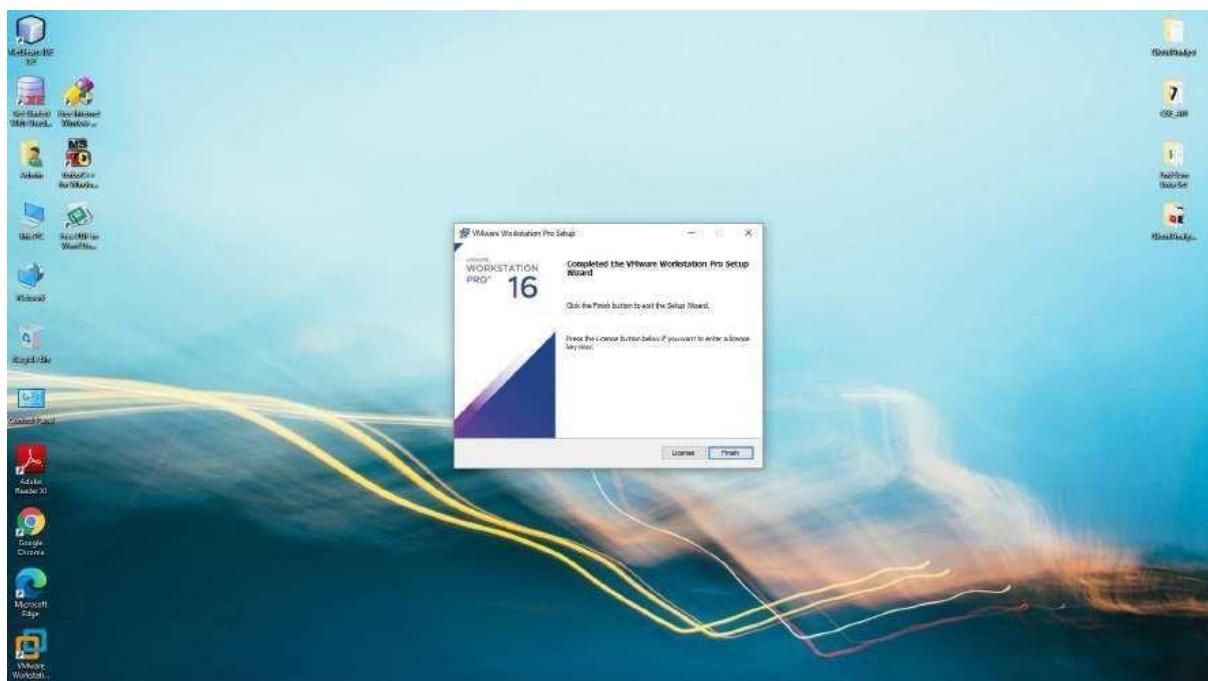
## Click on Install



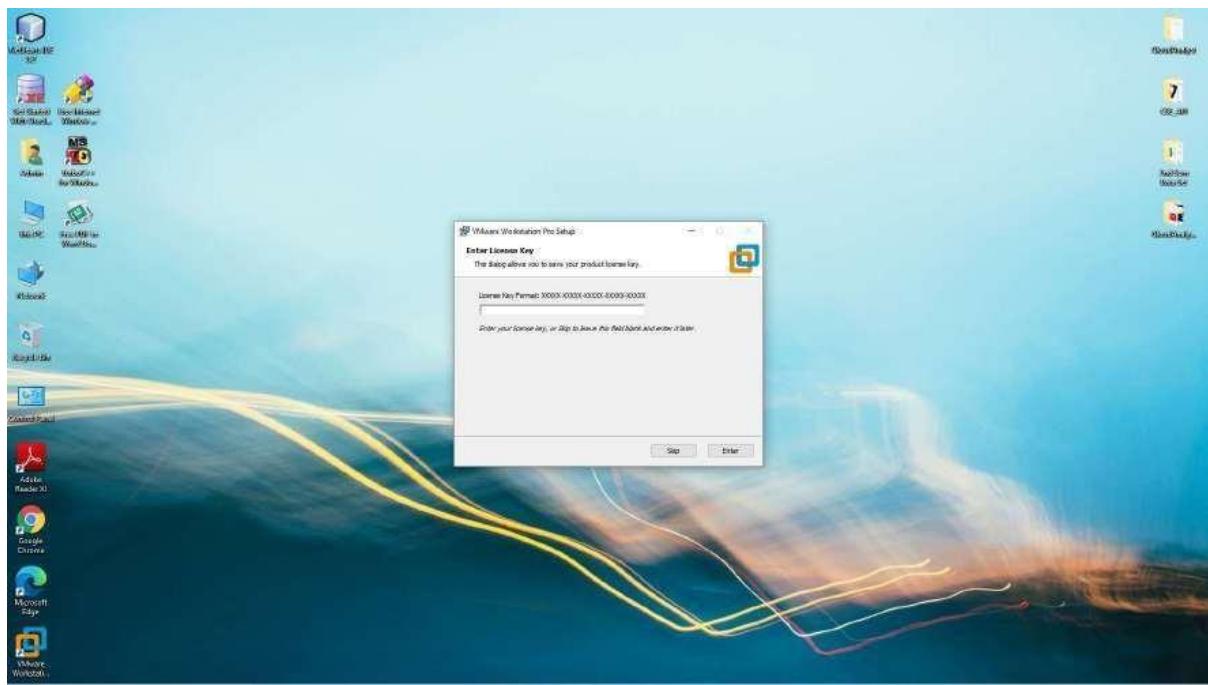
Wait for the installation to complete.



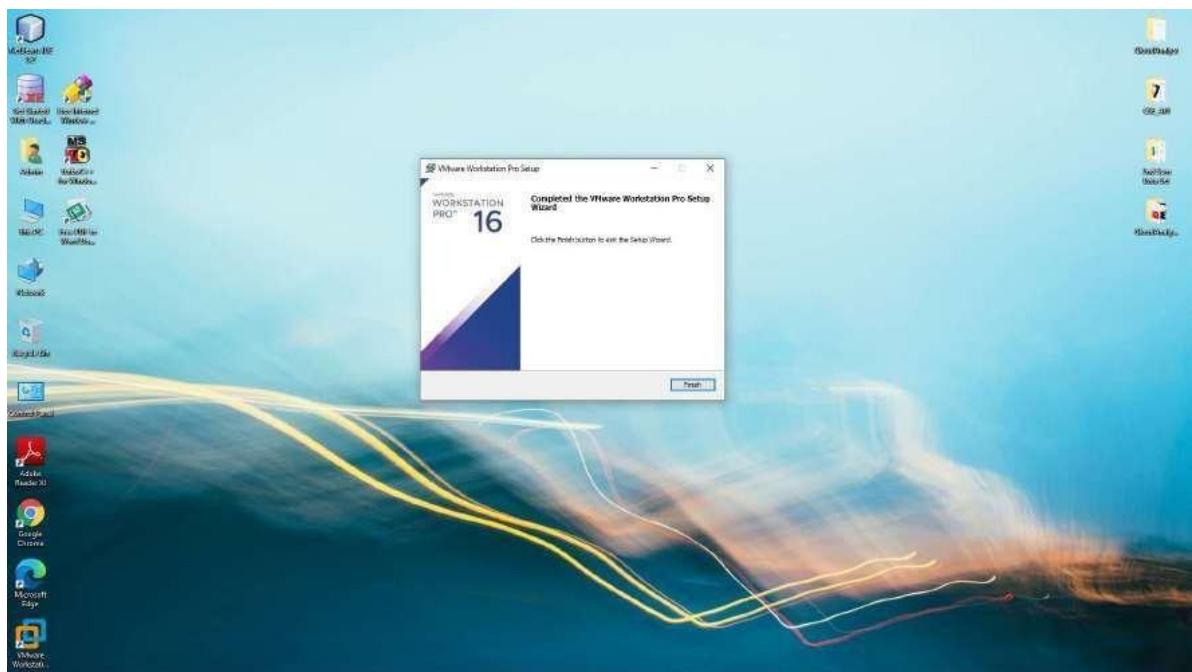
Click on Finish



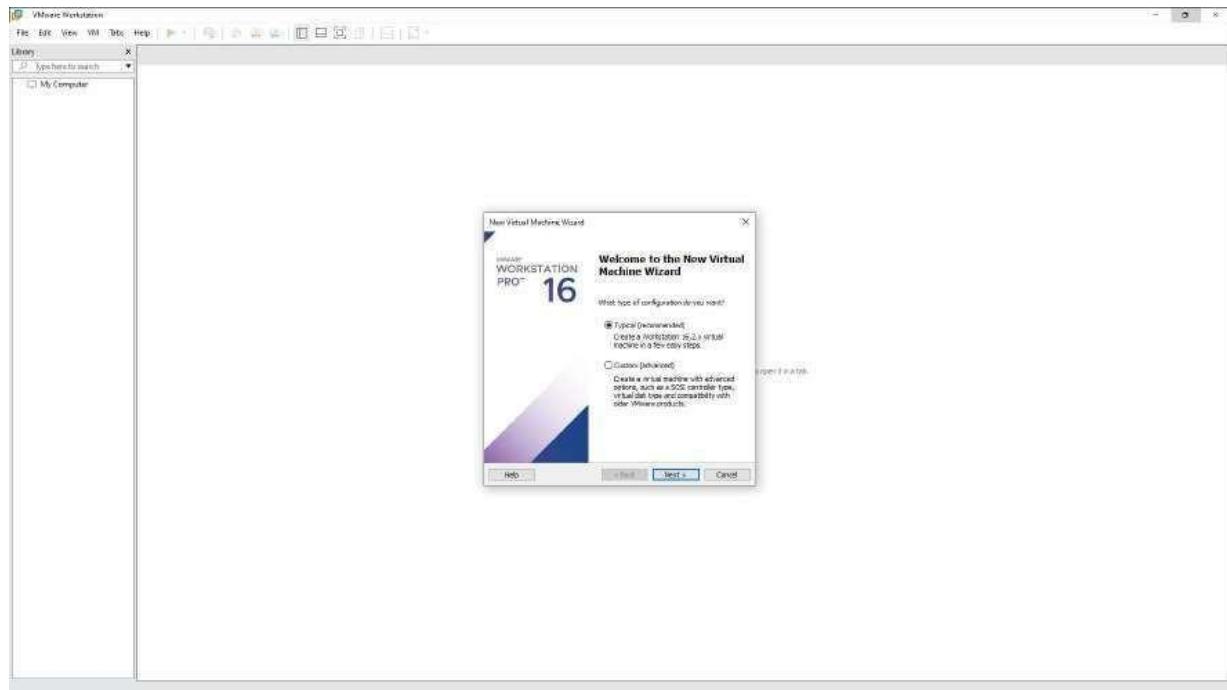
**Step 2:** Enter the license key received and click on Enter.



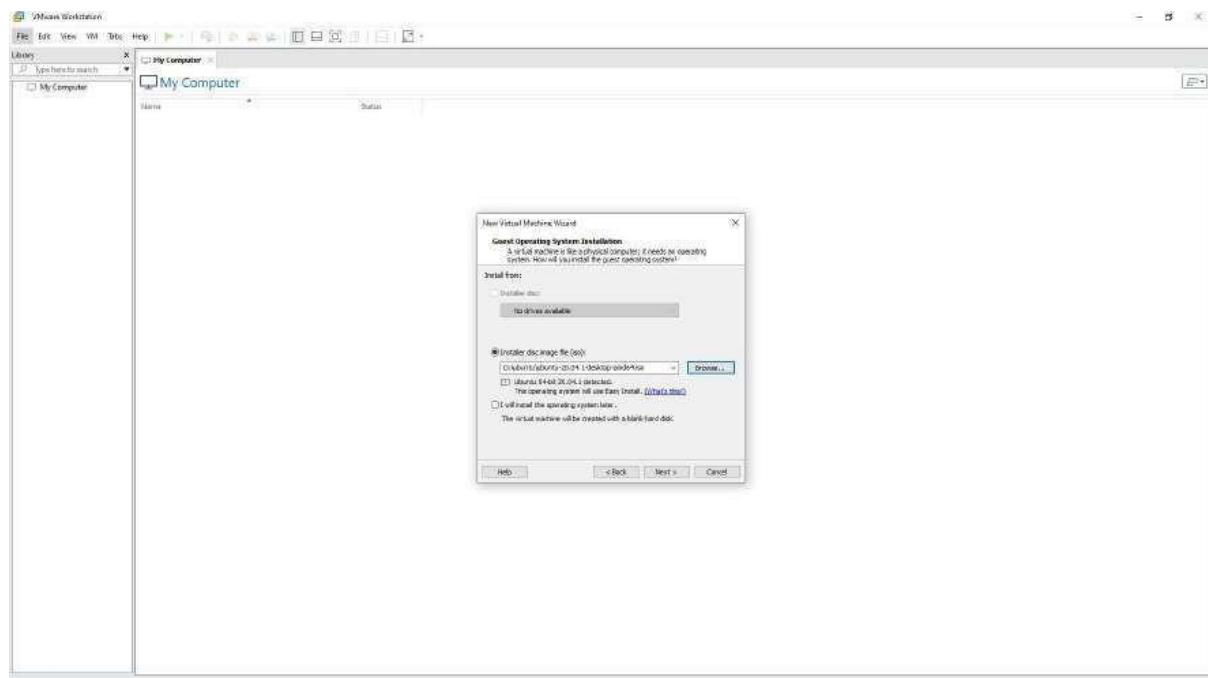
Click on Finish.



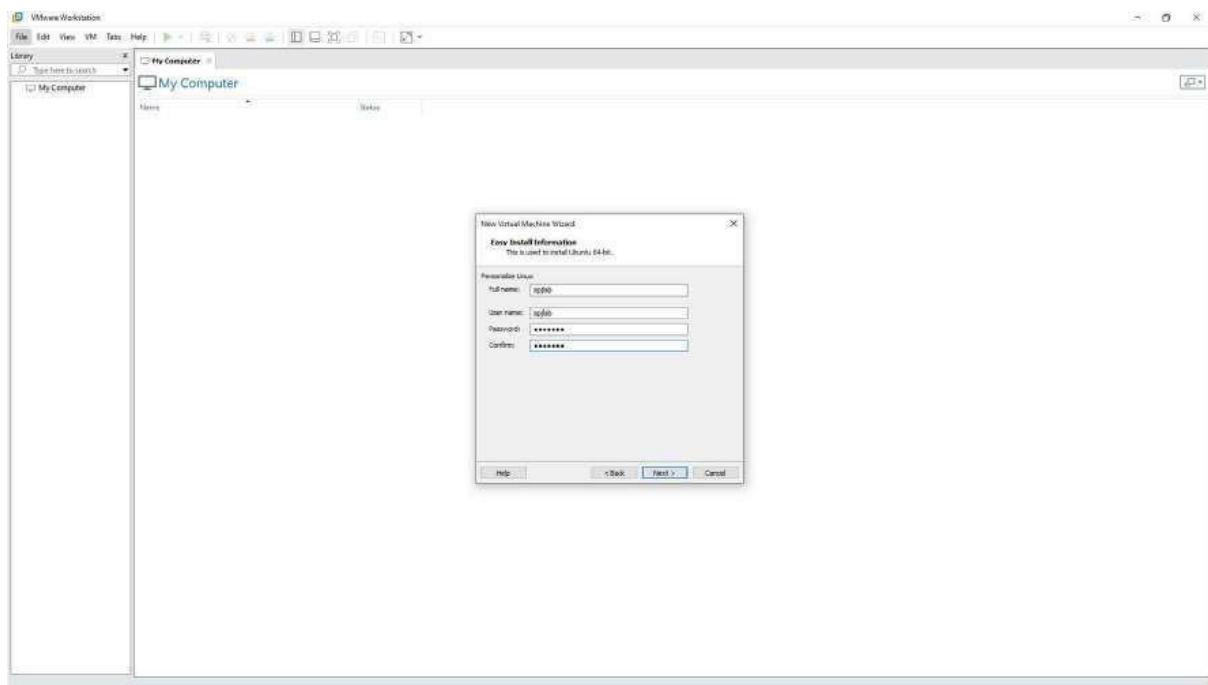
**Step 3:** Click on File and Click on New Virtual Machine. A window appears as shown in the picture below. Select Typical and click on Next.



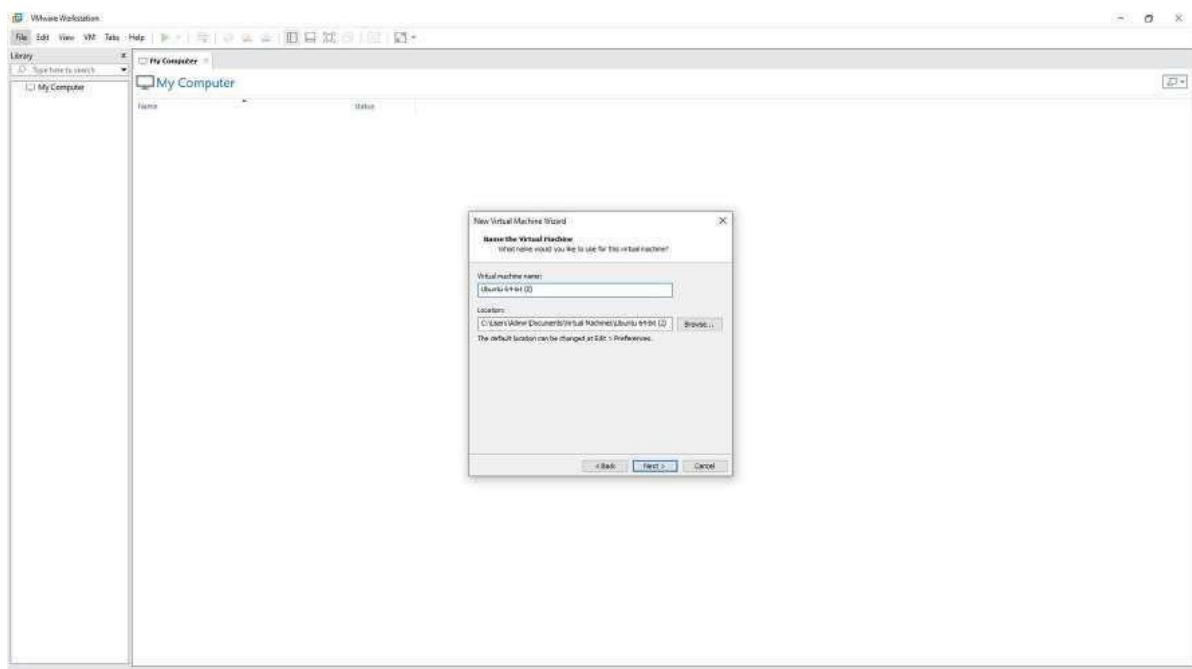
**Step 4:** Select the Ubuntu ISO file present in the hard disk of your computer. Click on Next.



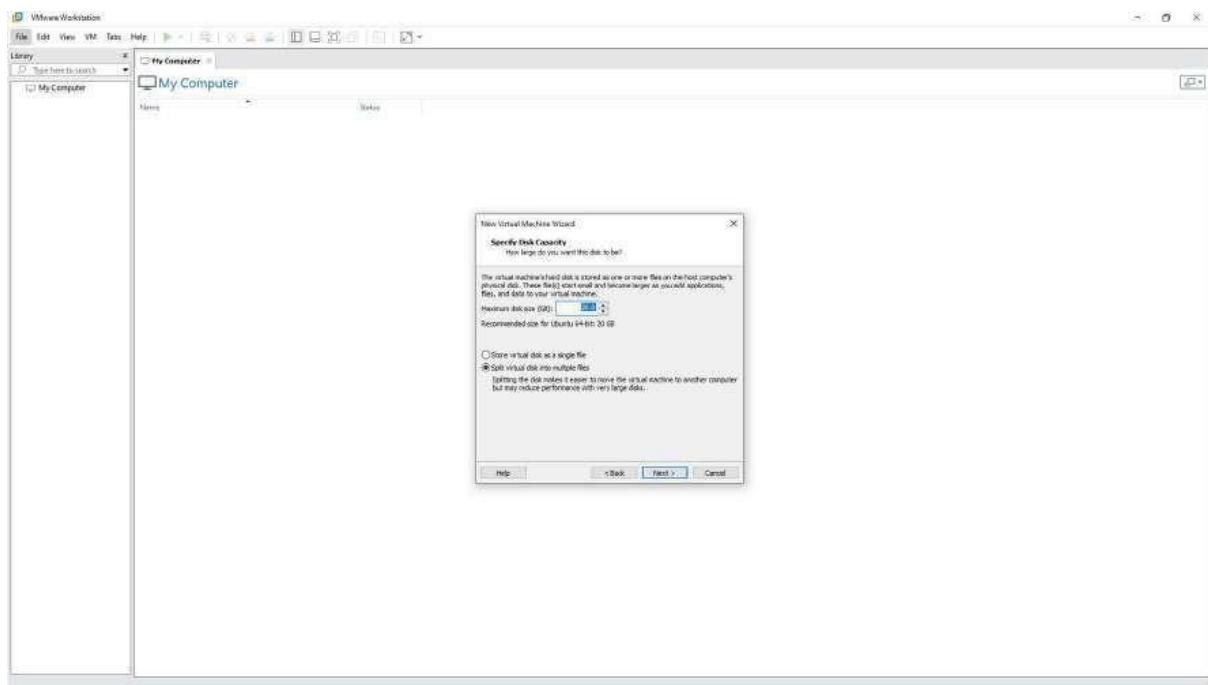
**Step 5:** Set the Username and Password for your virtual machine and click on next.



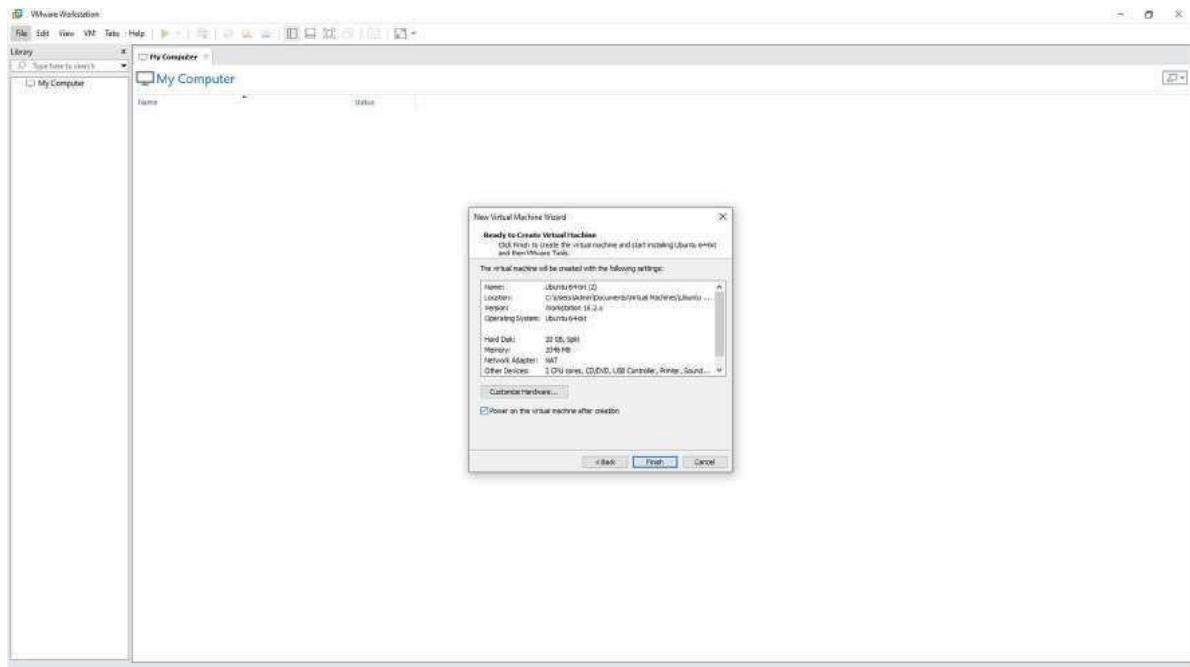
**Step 6:** Name the virtual machine and click on next.



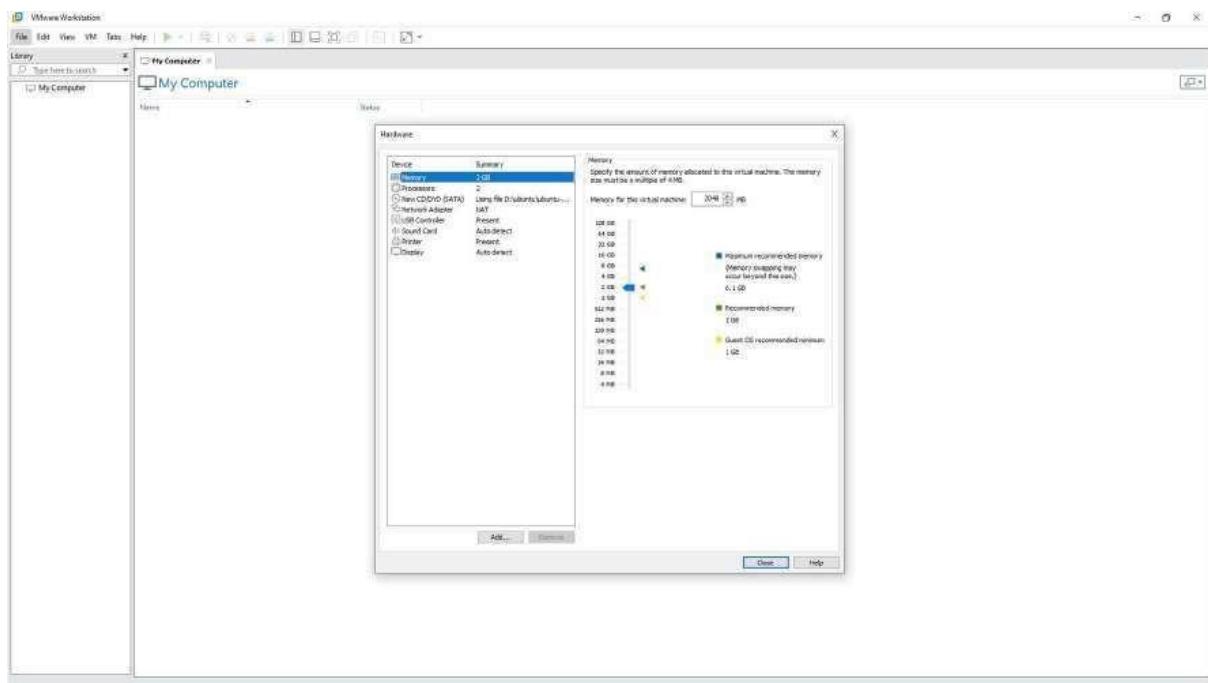
**Step 7:** Set the disk space to 40GB and click on Next.



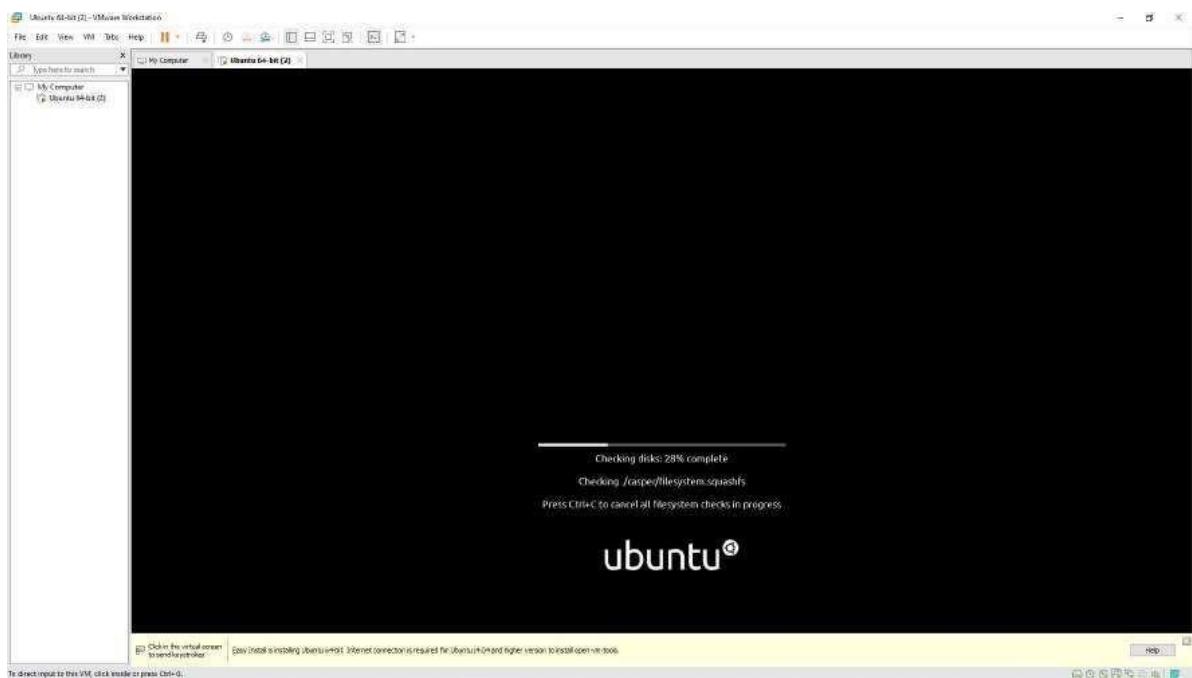
**Step 8:** Click on Customize Hardware.

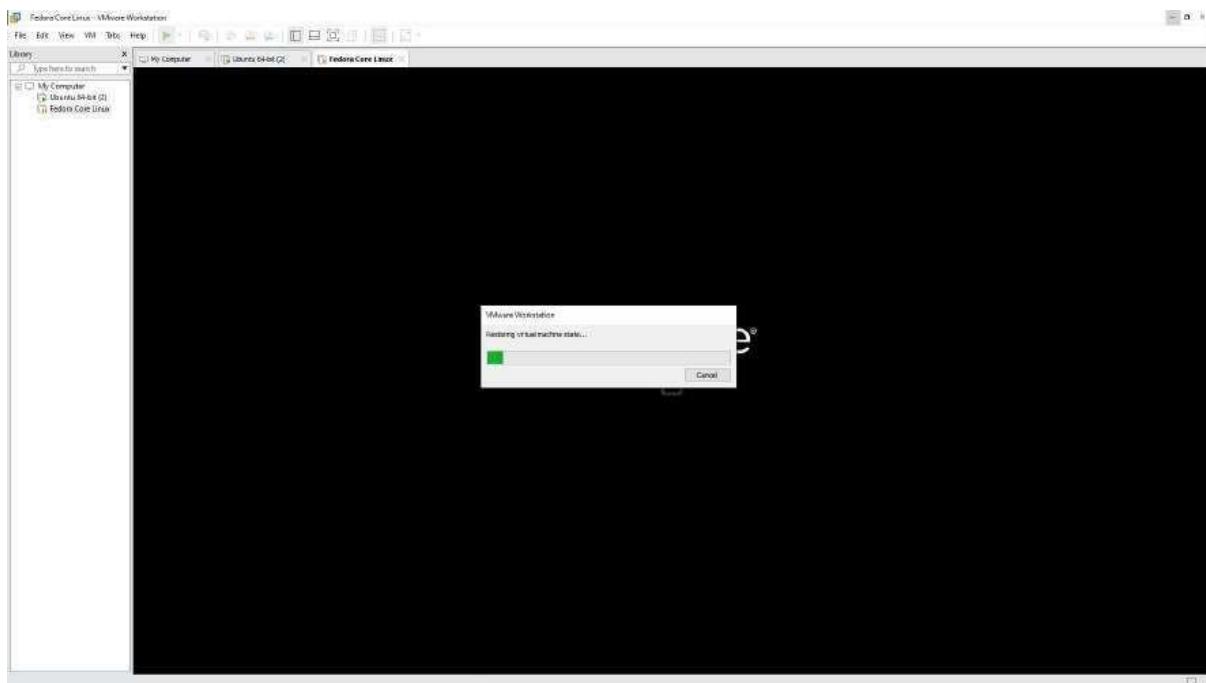


**Step 9:** Set the RAM Size to 4GB i.e., 4096 MB and click on Close and click on Finish.



Wait for the installation to complete.

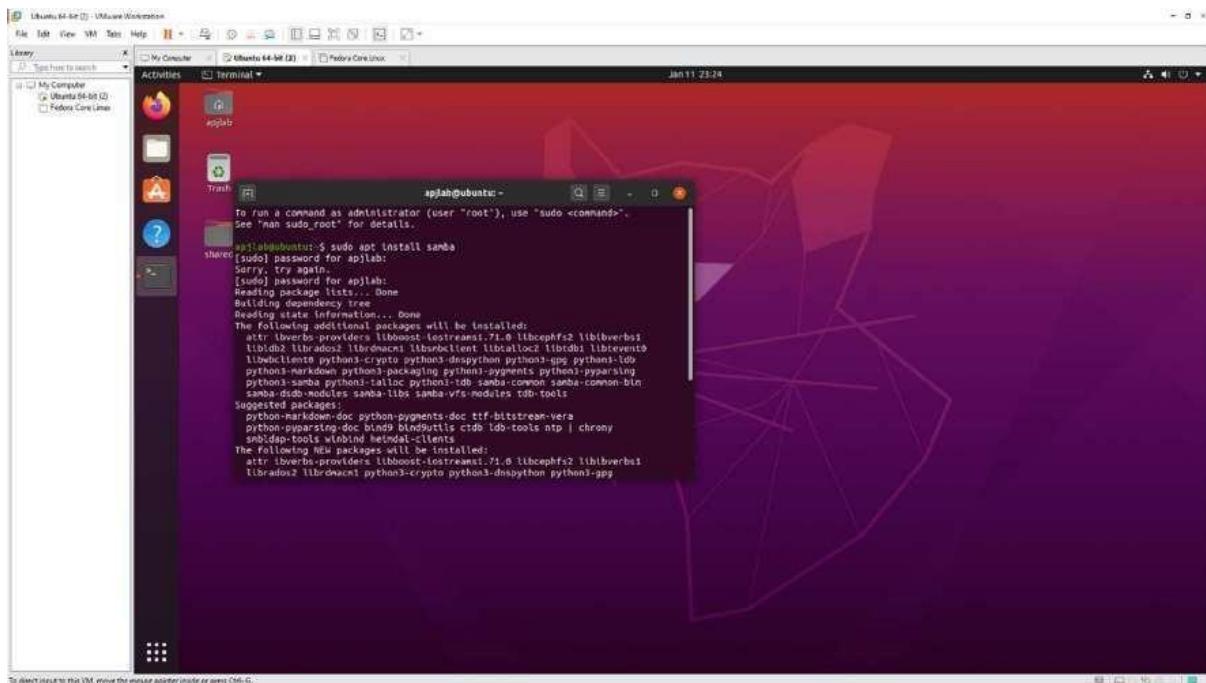




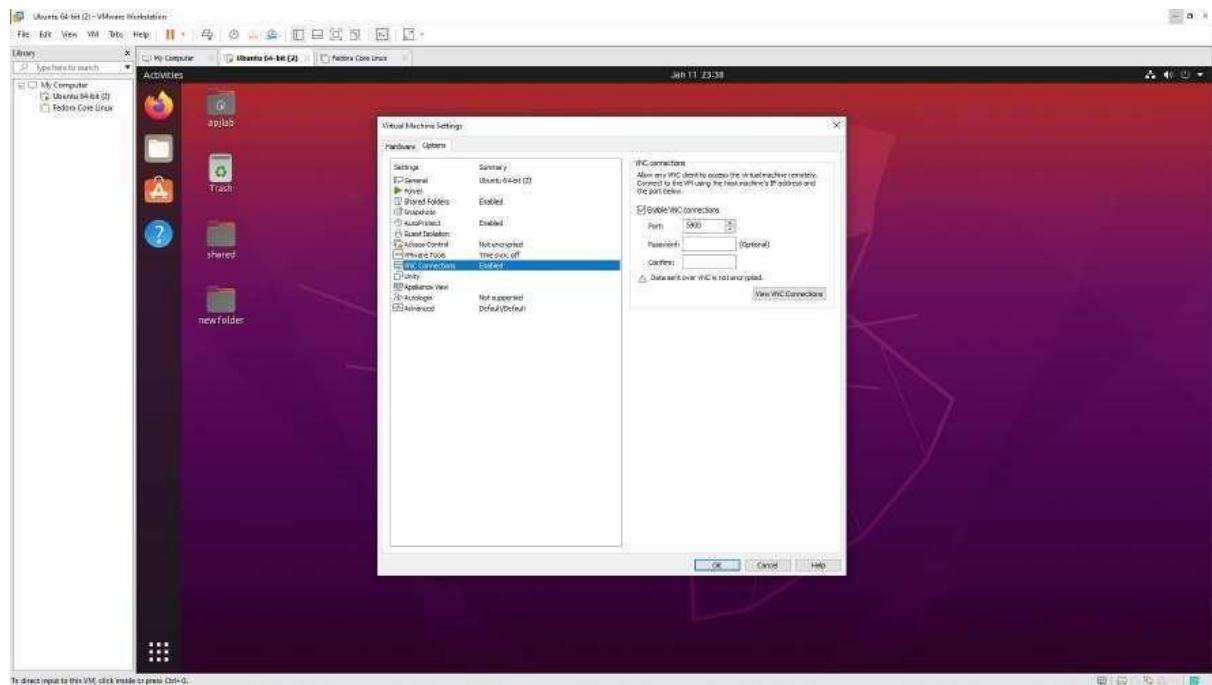
**Step 10:** Open the terminal and type the following command:

**sudo apt install samba**

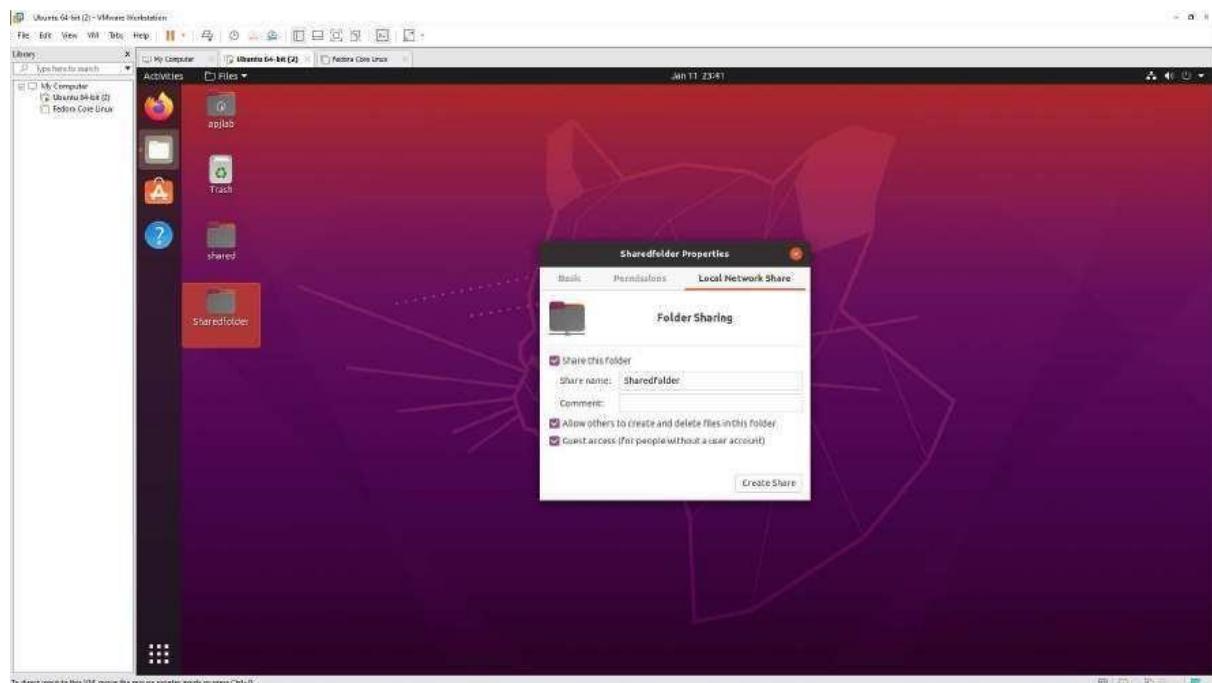
Wait for the installation to complete and close the terminal.



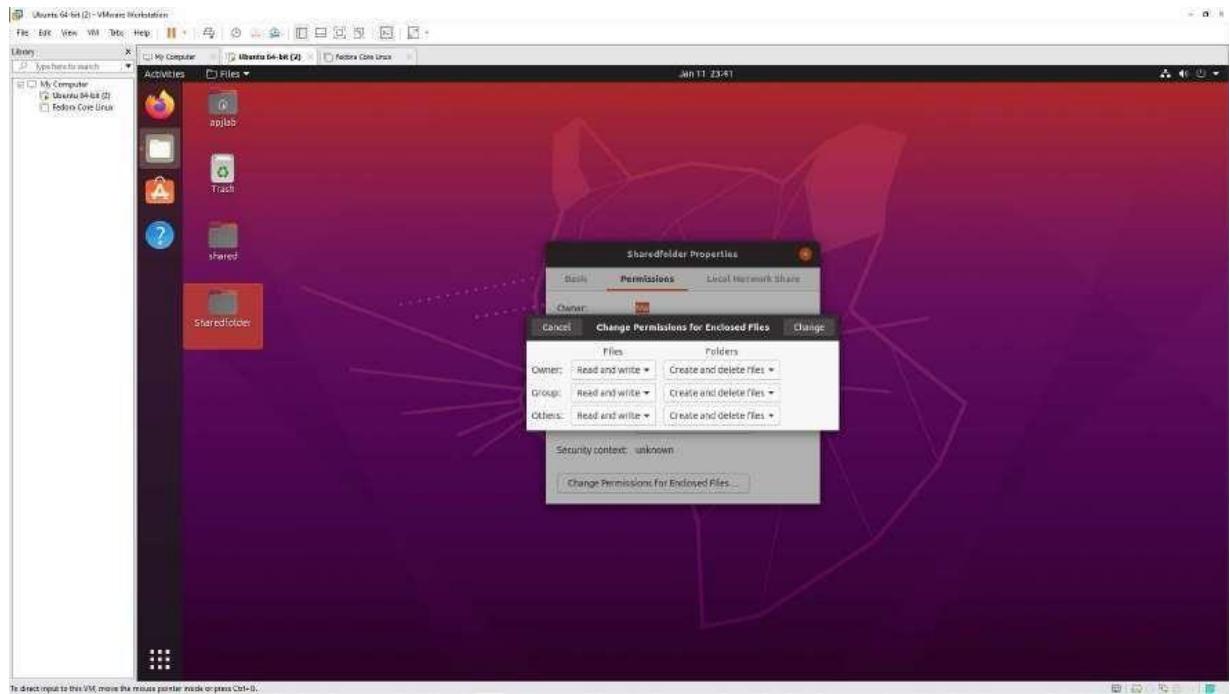
**Step 11:** Click on VM on top and click on Settings. A window appears. Click on Options and Enable Shared Folders, Auto Protect, VNC Connections and click on OK



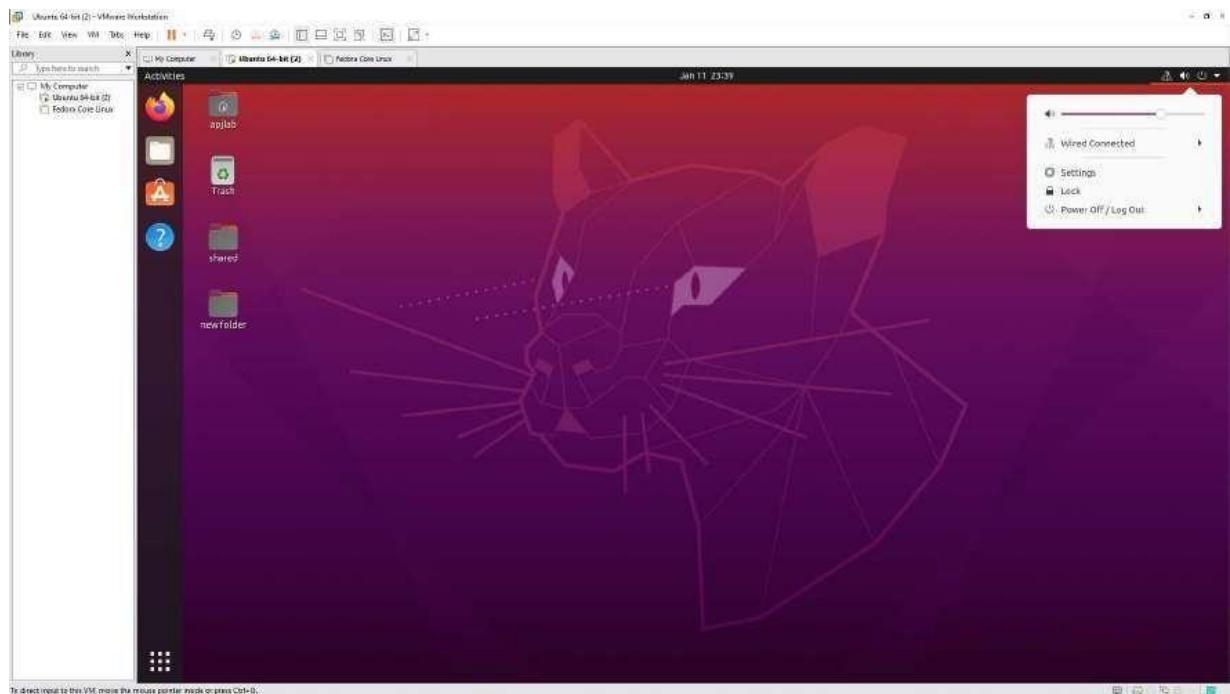
**Step 12:** Create a new folder that has to be shared and right click on the folder and click on properties. Select Share this folder, allow others to create and delete files in this folder and Guest access

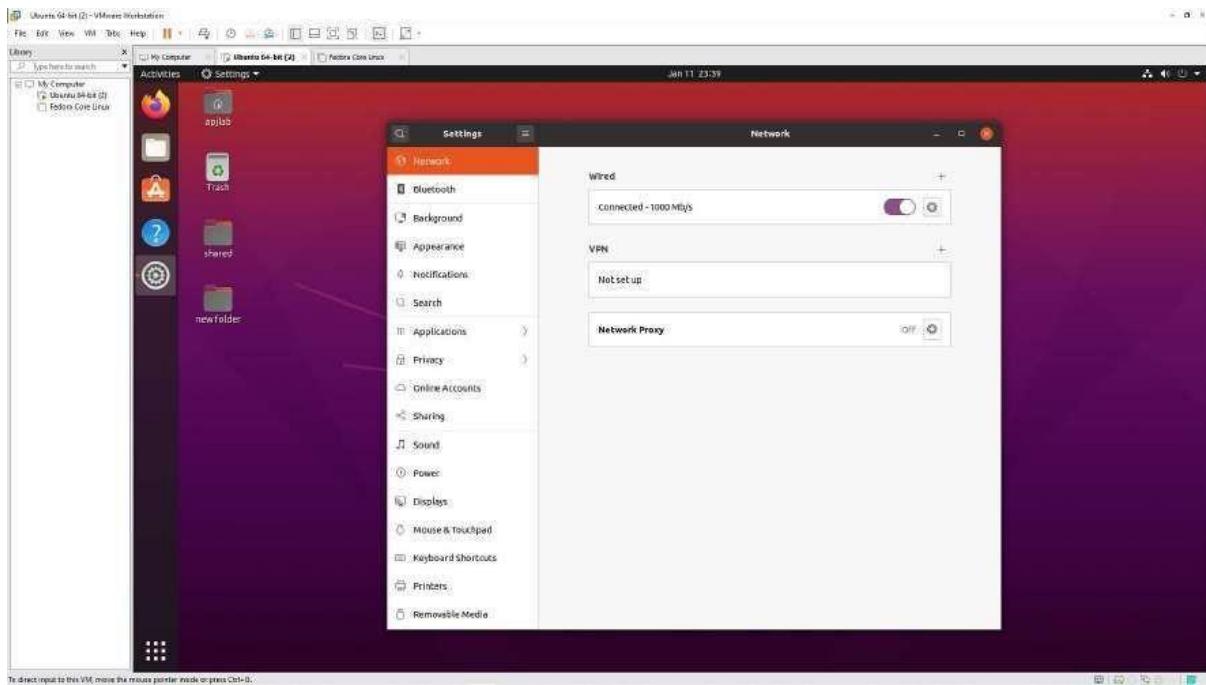


**Step 13:** Click on permissions and set Read and Write Permissions for Owner, Group and Others.

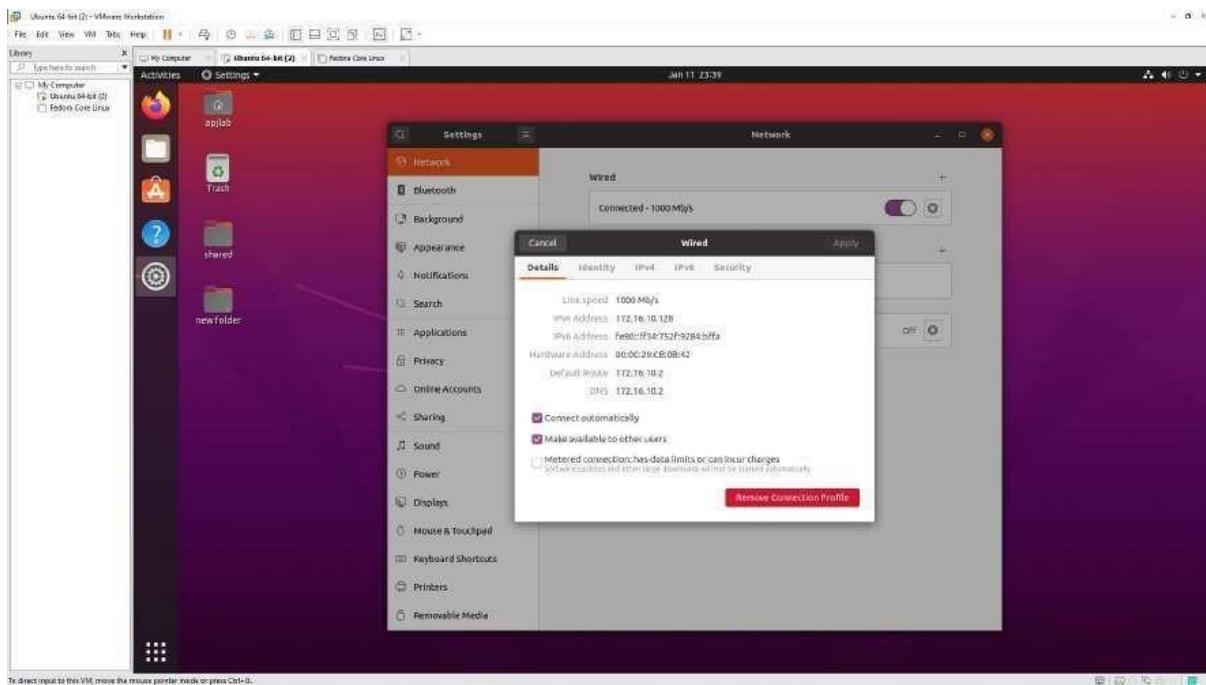


**Step 14:** Close all the windows and click on Settings

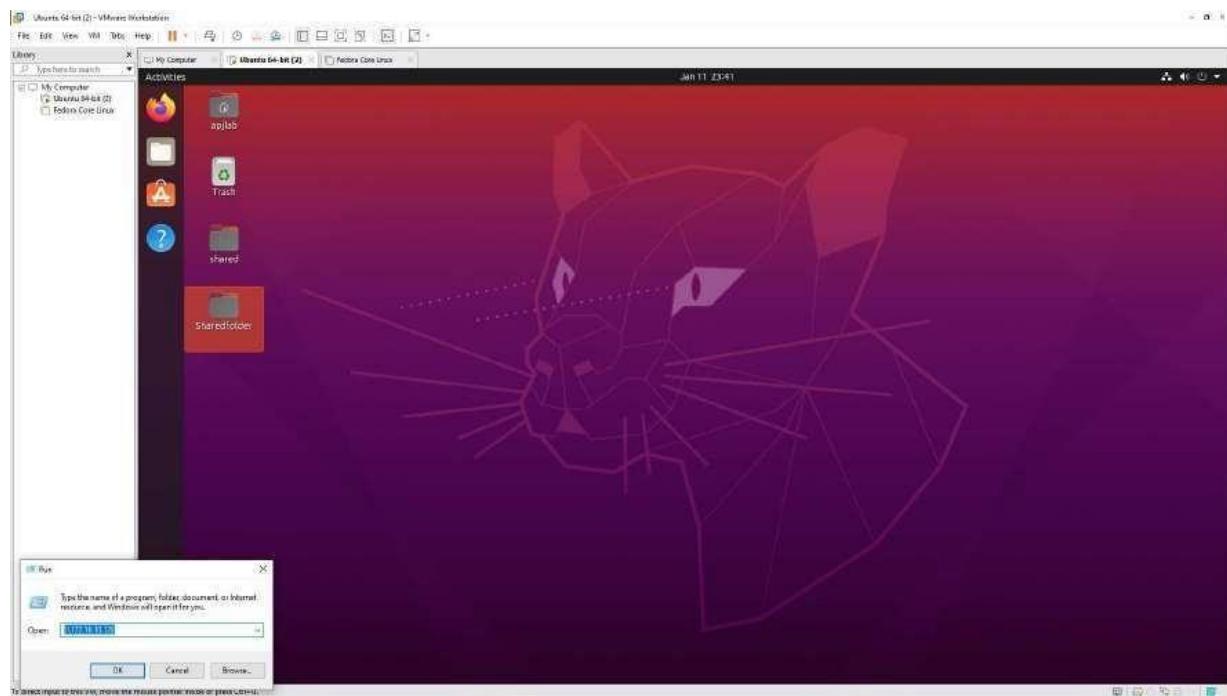




**Step 15:** Click on Wired Settings and window appears as shown in the below picture. Copy the IPv4 address.



**Step 16:** Open RUN on Windows and type the IP Address with the “\\”. For example, if the IPv4 Address copied is 172.16.10.128 in the Run window enter \\172.16.10.128



**Step 17:** A window appears showing the folder that was shared. The user can now access the folder.



---

# **PART B**

# **Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY**

(An Autonomous Institute, affiliated to VTU, Belagavi, Accredited by NAAC with ‘A’ Grade)

**Near Jnana Bharathi Campus, Bengaluru – 560056**



## **MINI PROJECT REPORT ON**

**“WhiteBoard APP”**

**Bachelor of engineering in**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted by,**

**KARTHIK B**

**1DA20CS054**

**Under the guidance of,**

**DR. Smitha Shekar B  
Associate Professor  
CSE DR AIT**

**DR. Asha K N  
Associate Professor  
CSE DR AIT**

**Department of Computer Science and Engineering**

**2023-2024**

# **Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY**

(An Autonomous Institute, affiliated to VTU, Belagavi, Accredited by NAAC with ‘A’ Grade)

**Near Jnana Bharathi Campus, Bengaluru – 560056**



## **CERTIFICATE**

### **“White Board”**

This is to certify that the project entitled **“Whiteboard APP”** submitted in the partial fulfillment of the requirement of the 7th semester Cloud Computing laboratory curriculum during the year 2021-22 is a result of Bonafede work carried out by-

**KARTHIK B**

**1DA20CS054**

**Signature of the guide,**

---

**Dr .Smitha Shekar**  
Asst Professor  
CSE Dr. AIT

---

**Dr. Asha K N**  
Associate Professor  
CSE DR AIT

1. Internal Examiner \_\_\_\_\_ copy
2. External Examiner \_\_\_\_\_

---

**Dr. Siddaraju**  
Head of Department  
CSE Dr. AIT

## **ACKNOWLEDGEMENT**

The satisfaction that accompanies to this project would be incomplete without the mention of the people who made it possible, without whose constant guidance and encouragement would have made our efforts go in vain.

We consider ourselves privileged to express our gratitude and respect towards all those who guided us through the project, "**White Board App**"

We would like to express our gratitude to **Dr. Meenakshi M, Principal Dr. AIT**, for providing us the congenial environment to work in.

We would like to express our profuse gratitude to **Dr. Siddaraju, HOD, Dept. of Computer Science & Engineering, Dr. AIT**, for giving us the support, encouragement and providing us the required lab facilities that was necessary for the completion of this project.

As a token of gratitude, we would like to acknowledge our sincere gratefulness to the internal guide **Dr.Smitha Shekar B and Dr.Asha K N Dept. of CSE, Dr. AIT**, for her unlimited support and encouragement provided throughout the process.

We also express our gratitude and sincere thanks to all the teaching and non-teaching staff of **Computer Science & Engineering Department**.

Finally, yet importantly, we would like to express our heartfelt thanks to our beloved **Parents** for their blessings and our **Friends** for their help and wishes for the successful completion of this project report.

**KARTHIK B**

## **ABSTRACT**

The White Board project is a web-based application developed using HTML, CSS, and JavaScript, designed to provide users with a digital platform for collaborative and interactive drawing. The primary objective of this project is to replicate the traditional whiteboard experience in a virtual environment, enabling users to draw, write, and collaborate in real-time. The application features a user-friendly interface with a canvas area where users can express their creativity using a variety of drawing tools, such as pens, brushes, and colors. The project leverages HTML for structuring the web page, CSS for styling and layout, and JavaScript for implementing dynamic functionalities.

# TABLE OF CONTENTS

<b>Chapter No</b>	<b>Title</b>	<b>Page Number</b>
Chapter 1	Introduction	1-3
1.1	Existing System	2
1.2	Proposed System	2
1.2.1	Scope of Work	2
1.2.2	Problem Statement	3
Chapter 2	Requirement Specification	4-5
2.1	Functional Requirements	4
2.2	Non-Functional Requirements	5
2.3	Hardware Specification	5
2.4	Software Specification	5
Chapter 3	Methodology and Design	6-10
3.1	Architecture Design	6
3.2	Back End	7
3.3	Front End	8
Chapter 4	Snapshots	11-13
4.1	Front End	11
4.2	Back End	13
	Development (Code)	14-17
	Deployment (Frontend and Backend)	18-21
	Conclusion & Future Enhancements	22
	References	22

# TABLE OF CONTENTS

<b>Chapter No</b>	<b>Title</b>	<b>Page Number</b>
Chapter 1	Introduction	167-169
1.1	Existing System	168
1.2	Proposed System	168
1.2.1	Scope of Work	168
1.2.2	Problem Statement	169
Chapter 2	Requirement Specification	170-171
2.1	Functional Requirements	170
2.2	Non-Functional Requirements	171
2.3	Hardware Specification	171
2.4	Software Specification	171
Chapter 3	Methodology and Design	172-176
3.1	Architecture Design	172
3.2	Back End	173
3.3	Front End	174
Chapter 4	Snapshots	177-179
4.1	Front End	177
4.2	Back End	179
	Development (Code)	180-183
	Deployment (Frontend and Backend)	184-187
	Conclusion & Future Enhancements	188
	References	188

# **CHAPTER 1**

## **INTRODUCTION**

In a world increasingly connected by digital technologies, the White Board project emerges as a versatile and interactive solution to facilitate collaborative drawing and brainstorming in a virtual space. This web-based application reimagines the traditional whiteboard experience, providing users with a dynamic platform to express their creativity, share ideas, and collaborate in real-time. Developed using a combination of HTML, CSS, and JavaScript, the White Board project aims to bridge the gap between physical and digital collaboration, catering to diverse user needs across education, business, and creative endeavors.

The essence of a whiteboard lies in its ability to serve as a canvas for free-form expression, idea generation, and collaborative problem-solving. This project seeks to capture and enhance these qualities in an online environment, enabling users to draw, write, and interact with each other seamlessly. Leveraging the power of web technologies, the application ensures accessibility from various devices, fostering a collaborative space that transcends geographical boundaries.

Key elements of the White Board project include a responsive and intuitive user interface, a rich set of drawing tools, real-time collaboration features, user authentication for personalized experiences, and the ability to save and share creations. As we delve into the details of the project, we will explore how HTML structures the web page, CSS enhances its visual appeal and layout, and JavaScript brings the dynamic functionalities to life.

This project not only caters to the demands of remote collaboration but also serves as a creative outlet for individuals to express themselves in a digital realm. By embracing the flexibility of web technologies, the White Board project endeavors to empower users to engage in collaborative drawing sessions, virtual meetings, or educational activities with ease and efficiency. Join us on this exploration of the White Board project, where the lines between the physical and digital worlds blur, and creativity knows no bounds.

## **1.1 EXISTING SYSTEM**

Traditionally, collaborative drawing and brainstorming activities have been conducted on physical whiteboards in meeting rooms, classrooms, and collaborative workspaces. While physical whiteboards serve their purpose well, they come with inherent limitations, especially in a world that increasingly relies on digital solutions. The transition to virtual platforms has been driven by the need for remote collaboration, flexibility, and the ability to archive and share creations efficiently. Here, we explore the existing systems that have paved the way for the White Board project:

### Physical Whiteboards:

Pros: Physical whiteboards are tangible, easy to use, and foster face-to-face collaboration. They have been a staple in educational institutions and offices for group discussions and brainstorming sessions.

Cons: Limited space, difficulty in archiving content, and the need for participants to be physically present in the same location hinder the scalability and flexibility of physical whiteboards.

### Digital Whiteboard Applications:

Pros: Several digital whiteboard applications exist, offering features like online collaboration, drawing tools, and the ability to save and share content. Examples include Microsoft Whiteboard, Miro, and Google Jamboard.

Cons: Some applications may require software installations, have a learning curve, or lack certain features. Additionally, they might be part of larger productivity suites, making them more complex than a dedicated, lightweight solution.

## **1.2 PROPOSED SYSTEM**

The proposed White Board system is a comprehensive web-based application that seeks to address the limitations of existing systems while harnessing the advantages of digital collaboration. With a focus on user-friendliness, real-time interaction, and robust features, the proposed system aims to redefine the virtual whiteboard experience.

- User-Friendly Interface: The system will feature an intuitive and responsive user interface, ensuring ease of use for individuals across different skill levels and devices.
- Saving and Sharing Functionality: Users will have the ability to save their whiteboard creations, enabling them to revisit and continue their work at a later time. Moreover, the system will facilitate easy sharing of boards, fostering collaboration beyond real-time sessions.

### **1.2.1 SCOPE OF THE WORK**

The scope of the White Board project encompasses the development and implementation of a feature-rich, web-based application that provides users with a collaborative and interactive platform for digital drawing. The project will cover the following aspects:

- User Interface Design: Designing an intuitive and user-friendly interface that accommodates drawing tools, collaboration features, and user authentication functionalities.
- Drawing Tools Implementation: Implementing a diverse set of drawing tools, including pens, brushes, colors, and erasers, to offer users a wide range of creative options.
- Real-Time Collaboration Features: Developing mechanisms for real-time collaboration, enabling multiple users to interact with the whiteboard simultaneously. This includes features for live drawing, annotations, and synchronous updates.
- User Authentication and Personalized Boards: Integrating user authentication to ensure secure access. Users will have personalized boards associated with their accounts, allowing them to save and revisit their creations.
- Saving and Sharing Functionality: Implementing a system for users to save their whiteboard sessions and share them with others. This may involve saving the data in a persistent format and generating shareable links.
- Responsive Design for Multi-Device Access: Ensuring that the application is responsive, providing a consistent and optimal user experience across various devices, including desktops, tablets, and smartphones.
- Rich Media Integration: Supporting the integration of images and other media into the whiteboard to enhance the creative possibilities and collaborative potential of the platform.
- User Feedback and Interaction Features: Implementing features that allow users to provide feedback, comment on drawings, and interact with each other during collaborative sessions.
- Session History and Playback: Developing a system to record and store the history of whiteboard sessions, enabling users to playback or review the progression of drawings and annotations.
- Accessibility and Inclusivity: Adhering to accessibility standards to ensure inclusivity for users with different abilities. This involves designing the application with features that make it usable and navigable for a diverse user base.
- Testing and Quality Assurance: Conducting thorough testing to identify and address any bugs, glitches, or performance issues. Ensuring the reliability and stability of the application under various usage scenarios.

### **1.2.2 PROBLEM STATEMENT**

Current collaborative drawing solutions, including physical whiteboards and existing digital platforms, lack an optimal balance of user-friendliness, real-time collaboration features, and accessibility. These systems often have limitations in terms of complexity, personalization, and drawing tools. The challenge is to develop a web-based White Board application that offers a seamless and intuitive platform for real-time interactive drawing, addressing these limitations and providing a user-centric solution for diverse collaborative scenarios.

## CHAPTER 2

# REQUIREMENT SPECIFICATION

In this chapter, the requirements for the White Board project will be outlined, providing a detailed specification to guide the development process. The requirements cover various aspects of the system, including functionality, performance, and user experience.

## 2.1 FUNCTIONAL REQUIREMENTS

- User Authentication and Account Management: The system should provide user authentication mechanisms to secure access. Users should have the ability to create accounts, log in, and manage their profiles.
- Whiteboard Creation and Management: Users should be able to create new whiteboards for collaborative drawing sessions. The system must support the management of multiple whiteboards per user account.
- Drawing Tools: The system should offer a variety of drawing tools, including pens, brushes, colors, and erasers. Users must be able to switch between drawing tools seamlessly.
- Real-Time Collaboration: Multiple users should be able to collaborate in real-time on the same whiteboard .Changes made by one user should be instantly reflected on the screens of other collaborators.
- User Interaction and Feedback: Users should have the ability to provide feedback on drawings and interact with each other. The system must support comments and collaborative annotations.
- Saving and Sharing :Users should be able to save their whiteboard sessions for future reference. The system must provide options to share whiteboards with others through links.

## 2.2 NON-FUNCTIONAL REQUIREMENTS

- **Performance:** The system must be responsive, with minimal latency during real-time collaboration. It should support a reasonable number of simultaneous users without significant performance degradation.
- **Scalability:** The application should be designed to scale with an increasing number of users and whiteboard sessions.
- **Security:** User authentication must be secure, and communication between users and the server should be encrypted. Personal data and whiteboard content should be stored securely.
- **Cross-Browser Compatibility:** The application should be compatible with popular web browsers, including Chrome, Firefox, Safari, and Edge.
- **Mobile Responsiveness:** The user interface must be responsive and user-friendly on various devices, including desktops, tablets, and smartphones.

## 2.3 HARDWARE REQUIREMENTS

Hardware refers to the physical elements of the computer. This is sometimes called the machinery or equipment of the whole model. In contrast to software, hardware is a physical entity. Hardware and software are interconnected. Without software, the hardware of the system would have no function. Here is a list of hardware and software components used in our system along with their functionalities and use.

- **Processor:** Intel i7 10<sup>th</sup> Gen
- **Processor Speed:** 2.4 GHz
- **Ram:** 8 Gb
- **Hard Disk:** 10 Gb

## 2.4 SOFTWARE REQUIREMENTS

The software requirement specifies the pre-installed software needed to run the code being implemented in this project. Software requirements for the proposed system are given as follows:

- **Operating System:** Windows 7/8/10/11
- **Integrated Development Environment:** Visual Studio Code, Postman.
- **Language:** Html, CSS, JavaScript.

## CHAPTER 3

### METHODOLOGY AND DESIGN

Framework outline is worried about making another framework in the spot of an old one. This is the most innovative and testing stage and essential as well. A powerful plan enhances procedural subtle elements important furthermore the understanding capacity for executing a shiny new framework.

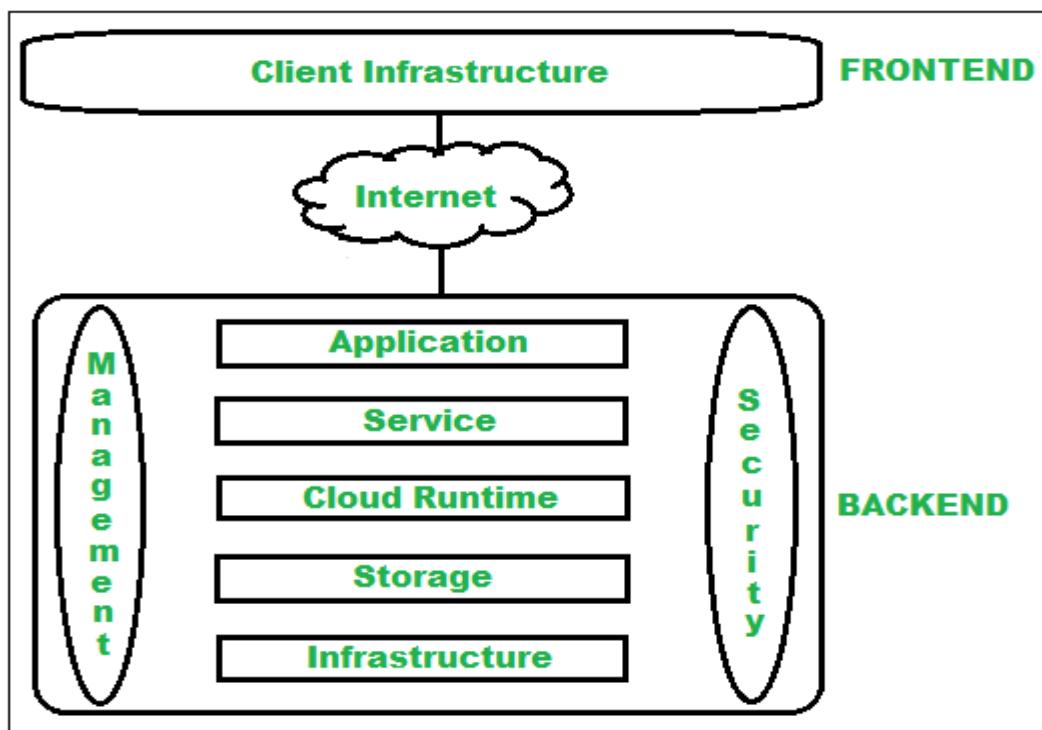


Figure 3.1.1 Architecture Design

#### 3.1 ARCHITECTURE DESIGN

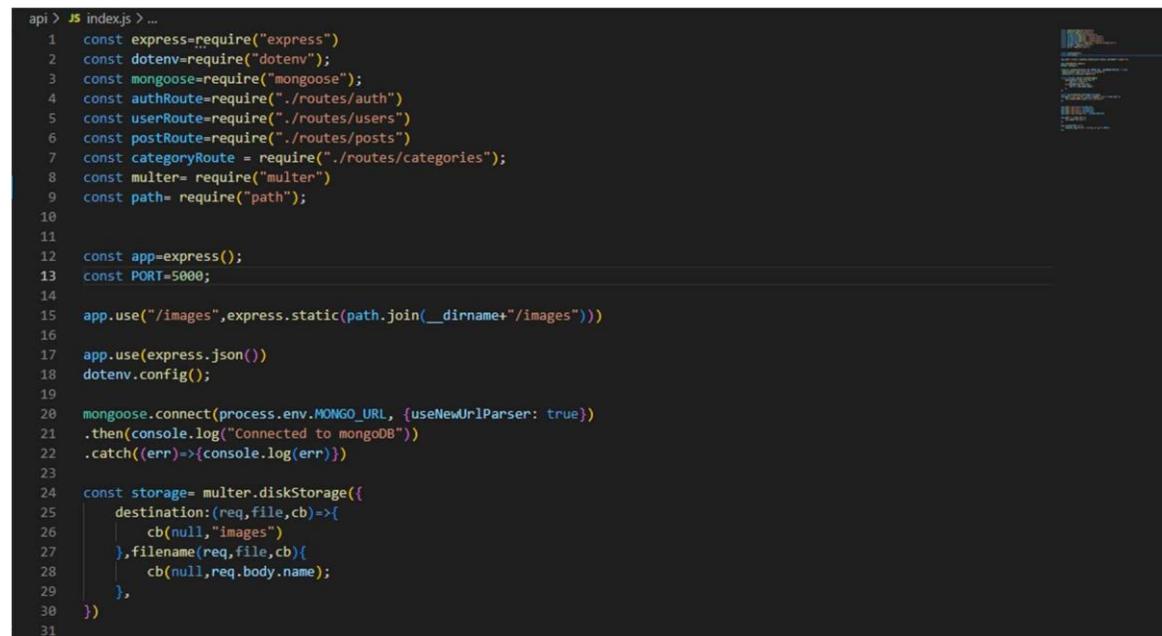
In the above figure we can see that the system architecture works on the model Request-Response. Firstly, we have React which is used to create various components under a similar category which in turn requests for the outline and data for its components. Secondly, Node.js uses routers and controllers to model the data while it loads data into react components through object document mapping. Finally, the mongo DB which uses mongoose module to connect to node is a NoSQL database which does not consist of tables, rows and columns.

## 3.2 BACKEND (Technologies)

Back-End refers to the server-side development. It focuses on databases, scripting, website architecture. It contains behind-the-scene activities that occur when performing any action on a website. It can be an account login or making a purchase from an online store. Code written by back-end developers helps browsers to communicate with database information.

### JavaScript

JavaScript is a versatile programming language that is primarily used for web development. It enables the creation of dynamic and interactive content on websites.



```
api > JS index.js > ...
1 const express=require("express")
2 const dotenv=require("dotenv");
3 const mongoose=require("mongoose");
4 const authRoute=require("./routes/auth")
5 const userRoute=require("./routes/users")
6 const postRoute=require("./routes/posts")
7 const categoryRoute = require("./routes/categories");
8 const multer= require("multer")
9 const path= require("path");
10
11
12 const app=express();
13 const PORT=5000;
14
15 app.use("/images",express.static(path.join(__dirname+"/images")))
16
17 app.use(express.json())
18 dotenv.config();
19
20 mongoose.connect(process.env.MONGO_URL, {useNewUrlParser: true})
21 .then(console.log("Connected to mongoDB"))
22 .catch((err)=>{console.log(err)})
23
24 const storage= multer.diskStorage({
25   destination:(req,file,cb)=>{
26     cb(null,"images")
27   },filename:(req,file,cb){
28     cb(null,req.body.name);
29   },
30 })
31
```

Figure 3.2.1 Index.js main file

```
JS script.js U X
JS script.js > ...
    ♫ Click here to ask Blackbox to help you code faster
1  let canvas = document.querySelector("#canvas");
2
3  canvas.width = window.innerWidth;
4  canvas.height = window.innerHeight - 100;
5
6  window.addEventListener("resize", function() {
7      canvas.width = window.innerWidth;
8      canvas.height = window.innerHeight - 100;
9      drawLinesFromDB();
10 });
11
12
13 let ctx = canvas.getContext("2d");
14
15 let linesDB = [];
16 let redoLinesDB = [];
17 let isPenDown = false;
18 let line = [];
19
20 canvas.addEventListener("mousedown", function(e) {
21     if (redoLinesDB.length) {
22         redoLinesDB = [];
23     }
24     console.log("Inside mouse down");
25     isPenDown = true;
26     let x = e.clientX;
27     let y = e.clientY - 100;
28     ctx.beginPath();
29     ctx.moveTo(x, y);
30
31     let pointObject = {
32         x: x,
33         y: y,
34         type: "md",
35         lineWidth: ctx.lineWidth,
36         strokeStyle: ctx.strokeStyle,
```

Figure 3.2.2 Index.js File

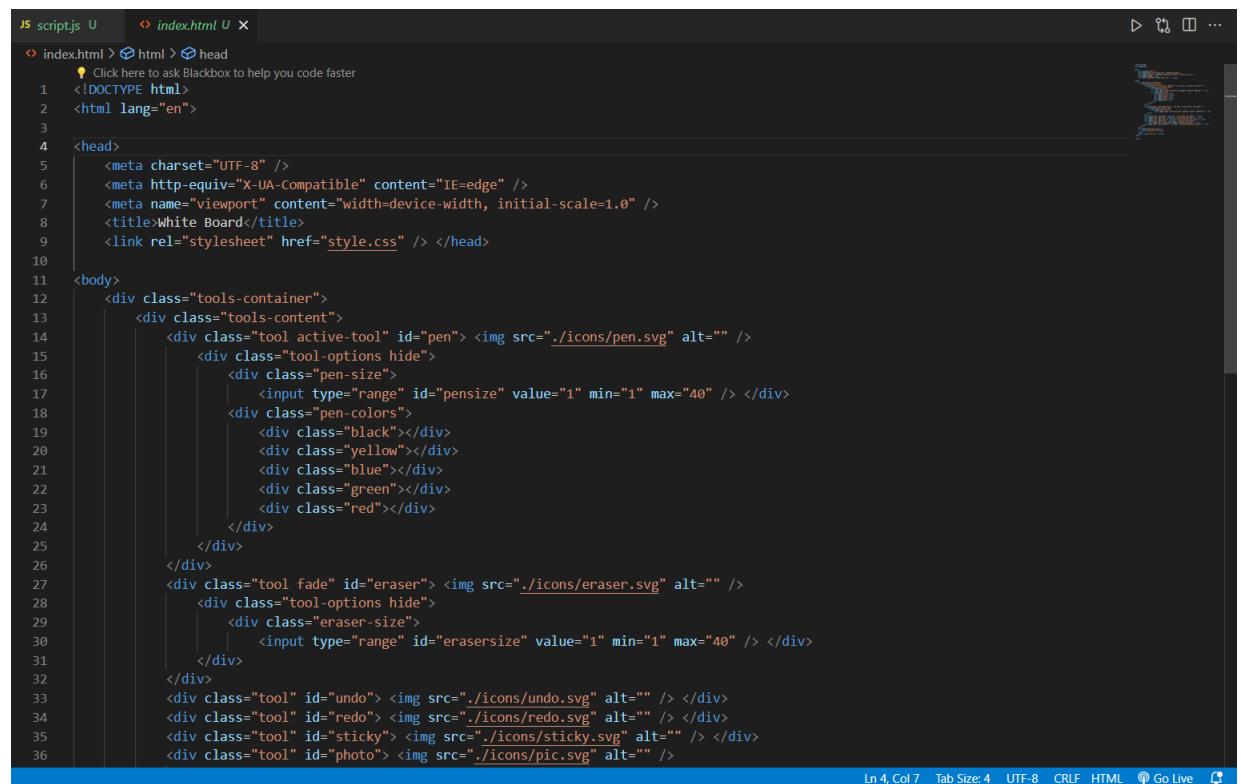
### 3.3 FRONTEND (Technologies)

Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.

#### HTML

HTML, which stands for Hypertext Markup Language, is the standard markup language used to create and structure content on the World Wide Web. It serves as the backbone of web development, providing a standardized way to define the structure and presentation of web pages.

HTML is a markup language, not a programming language. This means it consists of tags that define elements on a webpage. Each tag represents a specific piece of content or a structural element, allowing browsers to interpret and display the information accordingly.



The screenshot shows a code editor window with the file 'index.html' open. The code is for a whiteboard application. It includes DOCTYPE declaration, HTML5 opening tag, meta tags for charset, X-UA-Compatible, and viewport, a title, and a link to a stylesheet. The main content is a 'body' section containing a 'div' with class 'tools-container'. Inside this container are two 'div' elements with classes 'tool active-tool' and 'tool fade'. The first tool contains a 'pen' icon, a 'pen-size' input range, and four color swatches ('black', 'yellow', 'blue', 'green'). The second tool contains an 'eraser' icon, an 'eraser-size' input range, and three other icons ('undo', 'redo', 'sticky'). The code uses inline styles and external image links for icons.

```
JS script.js U  index.html U X
index.html > html > head
Click here to ask Blackbox to help you code faster
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8" />
6   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
7   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
8   <title>White Board</title>
9   <link rel="stylesheet" href="style.css" /> </head>
10
11 <body>
12   <div class="tools-container">
13     <div class="tools-content">
14       <div class="tool active-tool" id="pen"> 
15         <div class="tool-options hide">
16           <div class="pen-size">
17             <input type="range" id="pensize" value="1" min="1" max="40" /> </div>
18           <div class="pen-colors">
19             <div class="black"></div>
20             <div class="yellow"></div>
21             <div class="blue"></div>
22             <div class="green"></div>
23             <div class="red"></div>
24           </div>
25         </div>
26       </div>
27       <div class="tool fade" id="eraser"> 
28         <div class="tool-options hide">
29           <div class="eraser-size">
30             <input type="range" id="erasersize" value="1" min="1" max="40" /> </div>
31           </div>
32         </div>
33       <div class="tool" id="undo">  </div>
34       <div class="tool" id="redo">  </div>
35       <div class="tool" id="sticky">  </div>
36       <div class="tool" id="photo"> 
```

Figure 3.3.1 Html Main file

## **JSX**

JavaScript XML (JSX) is an extension to the ECMAScript syntax without any defined semantics. (JSX 2014) React embraces the fact that rendering logic is inherently coupled with another UI logic. Instead of separating technologies, React uses loosely coupled units called components that contain both. JSX is optional and not required to use React. However, JSX is a good visual aid when working with UI inside JavaScript. It also allows React to show more useful error and warning messages.

## **Virtual Document Object**

Model The HTML DOM was originally intended for static pages and thus was not optimized for creating dynamic UI. When the DOM updates, it has to update every node and re-paint the page with the corresponding CSS and layout. It is common for a single page application to contain thousands of dynamically generated nodes that have event listeners attached to them. In dynamic pages, the HTML DOM must check for changes in every node data at a regular interval. This is considerably reducing application performance. The Virtual DOM was invented as a solution to this inefficiency. The Virtual DOM is an abstraction of the HTML DOM. It is lightweight and detached from the browser.

While building client-side apps, a team of Facebook developers realized that the DOM is slow (The Document Object Model (DOM) is an application programming interface (API) for HTML and XML documents. It defines the logical structure of documents and the way a document is accessed and manipulated.).

## **CSS**

CSS, or Cascading Style Sheets, is a stylesheet language used for describing the presentation and styling of HTML documents.

It enables the separation of content from design, allowing developers to control the layout, colors, fonts, and other visual aspects of a webpage. CSS consists of rulesets, each comprising a selector and a declaration block.

Selectors target HTML elements, and the declaration block contains property-value pairs.

```
# style.css > ↗ body
  * {
    box-sizing: border-box;
    margin: 0;
    padding: 0;
  }
  body {
    overflow: hidden;
  }
  .tools-container {
    height: 100px;
    display: flex;
    justify-content: center;
    align-items: center;
  }
  .tools-content {
    height: 70%;
    width: 70%;
    display: flex;
    justify-content: space-evenly;
    align-items: center;
    background: #rgba(255, 154, 255, 0.652);
    border-radius: 10px;
    transform: translateY(5px);
    box-shadow: 0 5px 10px #191d1db3;
  }
  .tool {
    height: 55%;
    width: 10%;
    position: relative;
  }
  .tool img {
```

Ln 8, Col 22 Spaces: 2 UTF-8 CRLF CSS ⚙ Port : 5500 🔍

## Context API

In a typical React application, data is passed top-down (parent to child) via props, but such usage can be cumbersome for certain types of props (e.g. locale preference, UI theme) that are required by many components within an application. Context provides a way to share values like these between components without having to explicitly pass a prop through every level of the tree.

# CHAPTER 4

## 4.1 Front End

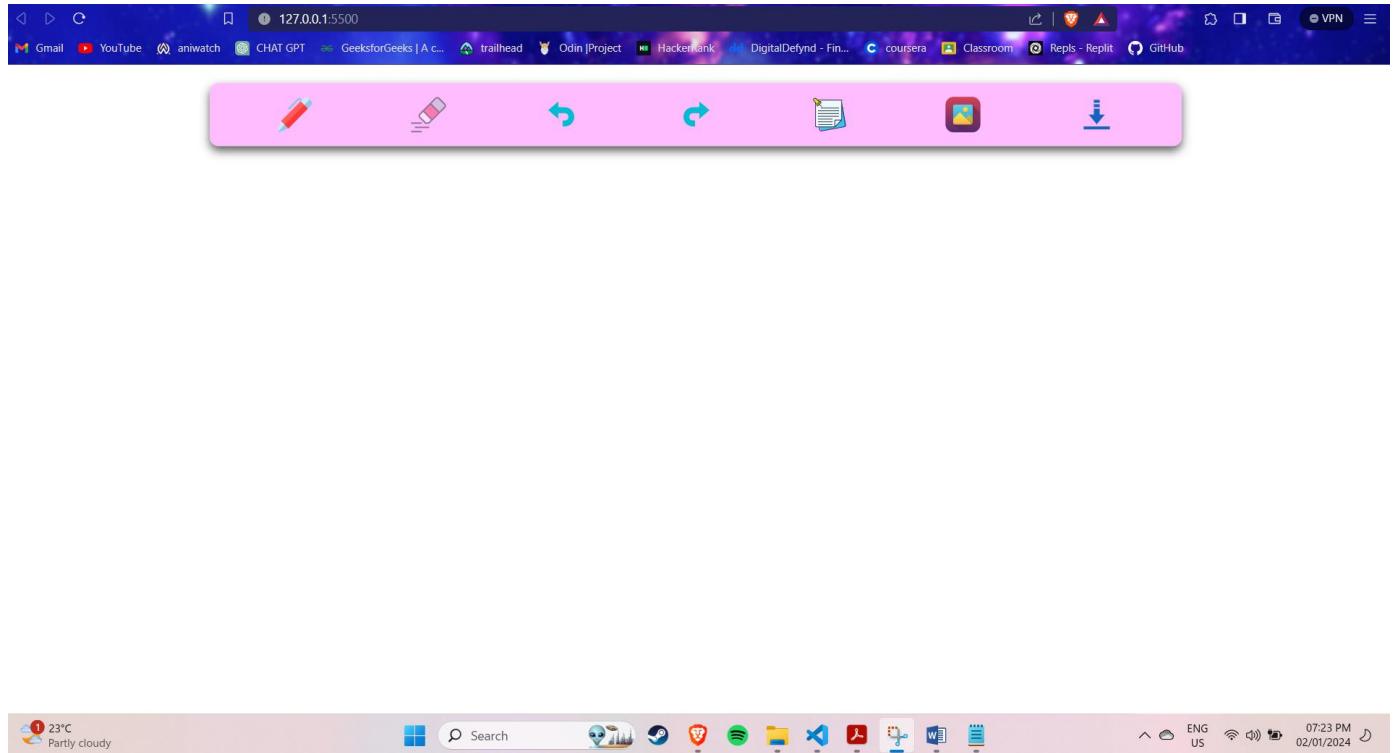


Figure 4.1.1 Front End Design Layout

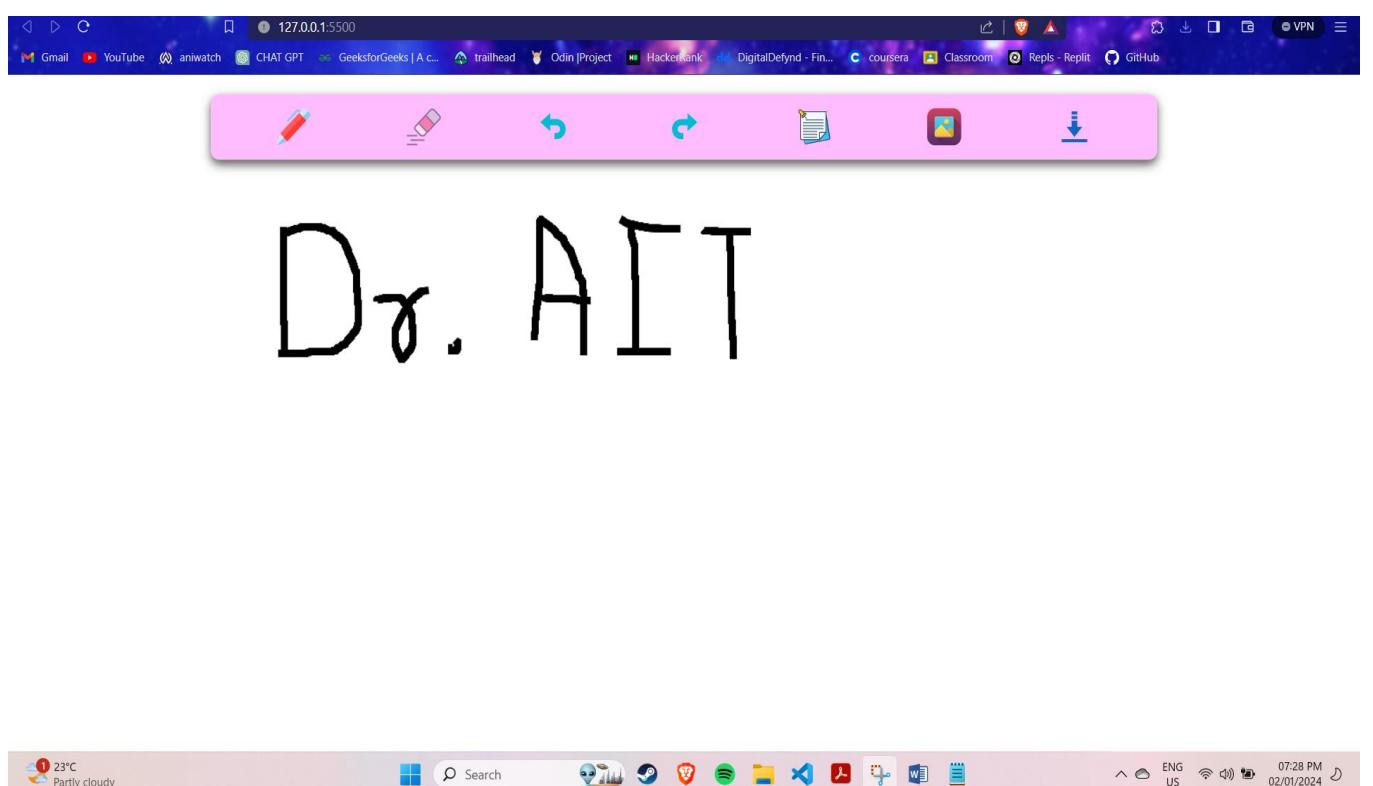
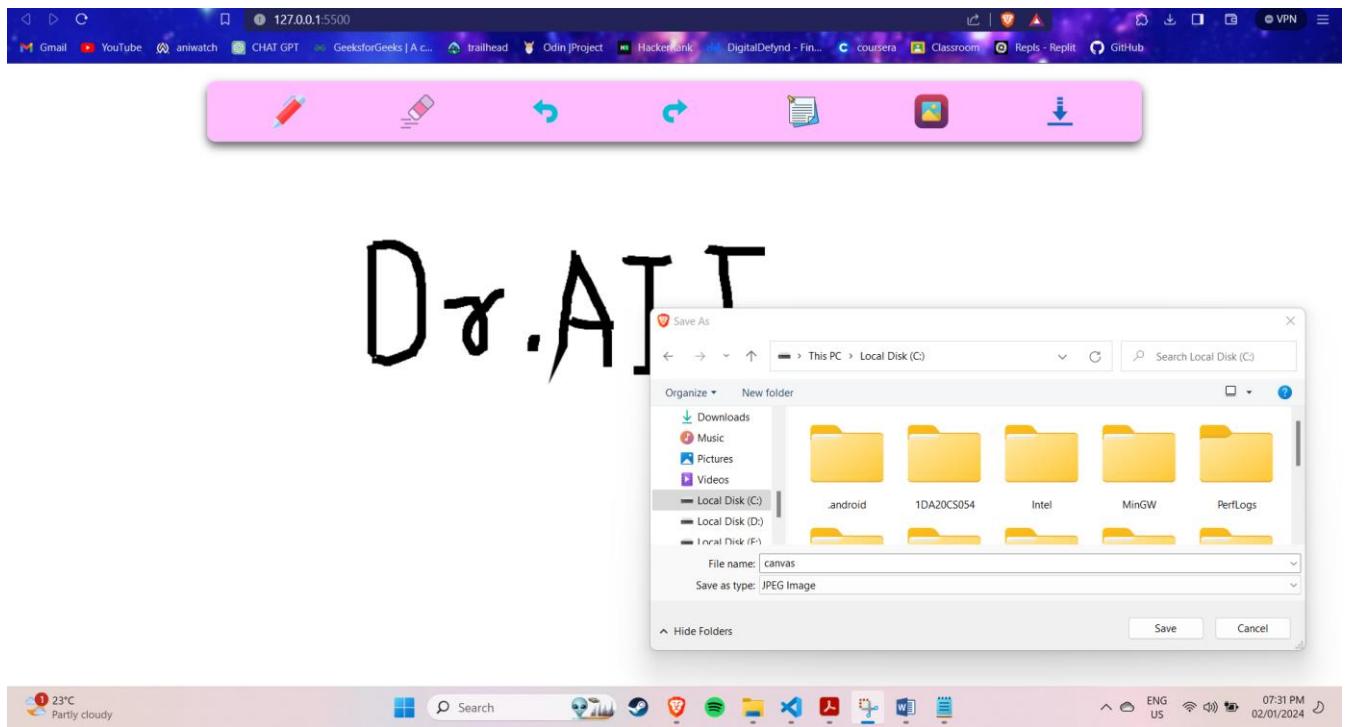


Figure 4.1.2 Inputting the content



**Figure 4.1.3 Basic layout of taking input from user**

## 4.2 Development Code

### Index.html

```
html>
<html lang="en">

<head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>White Board</title>
    <link rel="stylesheet" href="style.css" /> </head>

<body>
    <div class="tools-container">
        <div class="tools-content">
            <div class="tool active-tool" id="pen"> 
                <div class="tool-options hide">
                    <div class="pen-size">
                        <input type="range" id="pensize" value="1" min="1" max="40" />
                    </div>
                    <div class="pen-colors">
                        <div class="black"></div>
                        <div class="yellow"></div>
                        <div class="blue"></div>
                        <div class="green"></div>
                        <div class="red"></div>
                    </div>
                </div>
            </div>
            <div class="tool fade" id="eraser"> 
                <div class="tool-options hide">
                    <div class="eraser-size">
                        <input type="range" id="erasersize" value="1" min="1" max="40" />
                    </div>
                </div>
            </div>
            <div class="tool" id="undo">  </div>
            <div class="tool" id="redo">  </div>
            <div class="tool" id="sticky">  </div>
            <div class="tool" id="photo"> 
                <input type="file" class="hide" name="" id="photo-upload" />
            <div class="tool" id="download"> 
        </div>
        </div>
        <div class="drawing-container">
            <canvas id="canvas"></canvas>
        </div>
        <script src="script.js"></script>
    </body>
</html>
```

## Style.css

```
* {  
    box-sizing: border-box;  
    margin: 0;  
    padding: 0;  
}  
  
body {  
    overflow: hidden;  
}  
  
.tools-container {  
    height: 100px;  
    display: flex;  
    justify-content: center;  
    align-items: center;  
}  
  
.tools-content {  
    height: 70%;  
    width: 70%;  
    display: flex;  
    justify-content: space-evenly;  
    align-items: center;  
    background: rgba(255, 154, 255, 0.652);  
    border-radius: 10px;  
    transform: translateY(5px);  
    box-shadow: 0 5px 10px #191d1db3;  
}  
  
.tool {  
    height: 55%;  
    width: 10%;  
    position: relative;  
}  
  
.tool img {  
    height: 100%;  
    width: 100%;  
}  
  
/* .drawing-container {  
} */  
  
/* #canvas {  
    background-color: aqua;  
} */  
  
.sticky {  
    position: absolute;  
    top: 200px;  
    left: 300px;  
    height: 200px;  
    width: 250px;  
    /* background: bisque; */
```

```
}

.sticky-header {
  height: 13%;
  background: lightcyan;
  display: flex;
  justify-content: flex-end;
  align-items: center;
}

.sticky-header div {
  height: 70%;
  width: 7%;
  margin-right: 10px;
  border-radius: 50px;
}

.minimize {
  background: lightskyblue;
}

.close {
  background: lightcoral;
}

.sticky-content {
  height: 87%;
  border-bottom-right-radius: 20px;
  background: lightsteelblue;
  outline: none;
  padding: 10px;
}

.sticky-image-div {
  height: 87%;
}

.sticky-image {
  height: 100%;
  width: 100%;
  object-fit: cover;
}

/* .active-tool {
  transform: scale(1.2);
} */

.fade {
  opacity: 0.5;
}

.tool-options {
  height: 100px;
  width: 100px;
  position: absolute;
  /* background: aquamarine; */
```

```
top: 60px;
display: flex;
border-radius: 10px;
flex-direction: column;
justify-content: space-evenly;
padding: 2px;
background: aliceblue;
}

.hide {
  display: none;
}

/* .pen-size {
}
*/
.pen-size input {
  width: 100%;
}

.eraser-size input {
  width: 100%;
}

.pen-colors {
  display: flex;
  justify-content: space-evenly;
  width: 100%;
}

.pen-colors div {
  height: 15px;
  width: 14px;
  border-radius: 50px;
}

.black {
  background: black;
}

.yellow {
  background: yellow;
}

.blue {
  background: blue;
}

.green {
  background: green;
}

.red {
  background: red;
}
```

## Script.js

```
let canvas = document.querySelector("#canvas");

canvas.width = window.innerWidth;
canvas.height = window.innerHeight - 100;

window.addEventListener("resize", function() {
    canvas.width = window.innerWidth;
    canvas.height = window.innerHeight - 100;
    drawLinesFromDB();
});

let ctx = canvas.getContext("2d");

let linesDB = [];
let redoLinesDB = [];
let isPenDown = false;
let line = [];

canvas.addEventListener("mousedown", function(e) {
    if (redoLinesDB.length) {
        redoLinesDB = [];
    }
    console.log("Inside mouse down");
    isPenDown = true;
    let x = e.clientX;
    let y = e.clientY - 100;
    ctx.beginPath();
    ctx.moveTo(x, y);

    let pointObject = {
        x: x,
        y: y,
        type: "md",
        lineWidth: ctx.lineWidth,
        strokeStyle: ctx.strokeStyle,
    };
    line.push(pointObject);
});

canvas.addEventListener("mousemove", function(e) {
    if (isPenDown) {
        console.log("Inside mousemove");
        let x = e.clientX;
        let y = e.clientY - 100;
        ctx.lineTo(x, y);
        ctx.stroke();

        let pointObject = {
            x: x,
            y: y,
            type: "mm",
        };
        line.push(pointObject);
    }
});
```

```

    });

canvas.addEventListener("mouseup", function() {
    console.log("mouseup");
    isPenDown = false;

    linesDB.push(line);
    line = [];

    console.log(linesDB);
});

// Drawing

let pen = document.querySelector("#pen");
let eraser = document.querySelector("#eraser");

let penOptions = pen.querySelector(".tool-options");
let eraserOptions = eraser.querySelector(".tool-options");

let penSize = penOptions.querySelector("#pensize");
let eraserSize = eraserOptions.querySelector("#erasersize");

let penColors = penOptions.querySelectorAll(".pen-colors div");

let currentPenSize = 1;
let currentPenColor = "black";
let currentEraserSize = 1;

penSize.addEventListener("change", function() {
    // handle pen size
    let penSizeValue = penSize.value;
    // console.log(penSizeValue);
    // pensize set hogा
    currentPenSize = penSizeValue;
    ctx.lineWidth = currentPenSize;
});

eraserSize.addEventListener("click", function() {
    let eraserSizeValue = eraserSize.value;
    currentEraserSize = eraserSizeValue;
    ctx.lineWidth = currentEraserSize;
});

for (let i = 0; i < penColors.length; i++) {
    penColors[i].addEventListener("click", function(e) {
        let penColor = e.target.className;
        currentPenColor = penColor;
        ctx.strokeStyle = currentPenColor; // for lines
    });
}

pen.addEventListener("click", function() {
    if (pen.classList.contains("active-tool")) {

```

```

// pen already active hai
// pen tool options open honge
if (penOptions.classList.contains("hide")) {
    penOptions.classList.remove("hide"); // remove hide class from penOptions
} else {
    penOptions.classList.add("hide");
}
} else {
    // pen is not active
    // make pen active
    eraser.classList.remove("active-tool");
    eraser.classList.add("fade");
    eraserOptions.classList.add("hide");

    pen.classList.remove("fade");
    pen.classList.add("active-tool");

    ctx.lineWidth = currentPenSize;
    ctx.strokeStyle = currentPenColor;
}
});

eraser.addEventListener("click", function() {
    if (eraser.classList.contains("active-tool")) {
        // eraser already active
        if (eraserOptions.classList.contains("hide")) {
            eraserOptions.classList.remove("hide"); // remove hide class from penOptions
        } else {
            eraserOptions.classList.add("hide");
        }
    } else {
        // eraser not active
        pen.classList.remove("active-tool");
        pen.classList.add("fade");
        penOptions.classList.add("hide");

        eraser.classList.add("active-tool");
        eraser.classList.remove("fade");

        ctx.strokeStyle = "white";
        ctx.lineWidth = currentEraserSize;
    }
});
});
```

// Sticky Note

```

let sticky = document.querySelector("#sticky");

sticky.addEventListener("click", function() {
    addSticky();
});
```

```

function addSticky(imageElement) {
    let stickyDiv = document.createElement("div");
    stickyDiv.classList.add("sticky");
    stickyDiv.innerHTML = `<div class="sticky-header">
<div class="minimize"></div>
<div class="close"></div>
</div>`;
    let minimize = stickyDiv.querySelector(".minimize");
    let close = stickyDiv.querySelector(".close");
    let stickyHeader = stickyDiv.querySelector(".sticky-header");
    let stickyContent;
    if (imageElement) {
        let stickyImageDiv = document.createElement("div");
        stickyImageDiv.classList.add("sticky-image-div");
        stickyDiv.append(stickyImageDiv);
        stickyImageDiv.append(imageElement);
        stickyContent = stickyImageDiv;
    } else {
        // <div class="sticky-content" contenteditable="true"></div>
        let stickyContentDiv = document.createElement("div");
        stickyContentDiv.classList.add("sticky-content");
        stickyContentDiv.setAttribute("contenteditable", "true");
        stickyDiv.append(stickyContentDiv);
        stickyContent = stickyContentDiv;
    }
}

minimize.addEventListener("click", function() {
    stickyContent.style.display == "none" ?
        (stickyContent.style.display = "block") :
        (stickyContent.style.display = "none");
});

close.addEventListener("click", function() {
    stickyDiv.remove();
});

let stickyHold = false;
let initialX;
let initialY;
stickyHeader.addEventListener("mousedown", function(e) {
    stickyHold = true;
    initialX = e.clientX;
    initialY = e.clientY;
});

stickyHeader.addEventListener("mousemove", function(e) {
    if (stickyHold) {
        let finalX = e.clientX;
        let finalY = e.clientY;

        let dx = finalX - initialX;
        let dy = finalY - initialY;

        let {
            top,
            left
        }
    }
});

```

```

        } = stickyDiv.getBoundingClientRect();
        // sticky => top + dy
        // sticky => left + dx
        stickyDiv.style.top = top + dy + "px";
        stickyDiv.style.left = left + dx + "px";

        initialX = finalX;
        initialY = finalY;
    }
});

stickyHeader.addEventListener("mouseup", function() {
    stickyHold = false;
});

document.body.append(stickyDiv);
}

// Undo redo

let undo = document.querySelector("#undo");
let redo = document.querySelector("#redo");
undo.addEventListener("click", undoLine);
redo.addEventListener("click", redoLine);

function undoLine() {
    if (linesDB.length) {
        let undoLine = linesDB.pop();
        redoLinesDB.push(undoLine);

        // clear canvas
        ctx.clearRect(0, 0, canvas.width, canvas.height);

        drawLinesFromDB();
    }
}

function redoLine() {
    if (redoLinesDB.length) {

        let currentLineWidth = ctx.lineWidth;
        let currentStrokeStyle = ctx.strokeStyle;

        let redoLine = redoLinesDB.pop();
        for (let i = 0; i < redoLine.length; i++) {
            let pointObject = redoLine[i];
            if (pointObject.type == "md") {
                ctx.lineWidth = pointObject.lineWidth;
                ctx.strokeStyle = pointObject.strokeStyle;
                ctx.beginPath();
                ctx.moveTo(pointObject.x, pointObject.y);
            } else {
                ctx.lineTo(pointObject.x, pointObject.y);
                ctx.stroke();
            }
        }
    }
}

```

```

        linesDB.push(redoLine);

        ctx.lineWidth = currentLineWidth;
        ctx.strokeStyle = currentStrokeStyle;
    }

}

function drawLinesFromDB() {
    let currentLineWidth = ctx.lineWidth;
    let currentStrokeStyle = ctx.strokeStyle;

    for (let i = 0; i < linesDB.length; i++) {
        let line = linesDB[i];
        for (let i = 0; i < line.length; i++) {
            let pointObject = line[i];
            if (pointObject.type == "md") {
                ctx.lineWidth = pointObject.lineWidth;
                ctx.strokeStyle = pointObject.strokeStyle;
                ctx.beginPath();
                ctx.moveTo(pointObject.x, pointObject.y);
            } else {
                ctx.lineTo(pointObject.x, pointObject.y);
                ctx.stroke();
            }
        }
    }

    ctx.lineWidth = currentLineWidth;
    ctx.strokeStyle = currentStrokeStyle;
}

// Img Upload

let photoDiv = document.querySelector("#photo");
let photoUploadInput = document.querySelector("#photo-upload");
let downloadDiv = document.querySelector("#download");

photoDiv.addEventListener("click", function() {
    photoUploadInput.click();
});

photoUploadInput.addEventListener("change", function(event) {
    console.log(event);
    let fileObj = event.target.files[0];
    console.log(fileObj);
    let filePath = URL.createObjectURL(fileObj, {
        type: "image/jpg"
    });
    let img = document.createElement("img");
    img.setAttribute("src", filePath);
    img.classList.add("sticky-image");
    addSticky(img);
});

```

```
downloadDiv.addEventListener("click", function() {  
    let imagePath = canvas.toDataURL("image/jpg");  
    console.log(imagePath);  
    // <a href="" download="canvas.jpg"></a>  
    let aTag = document.createElement("a");  
    aTag.download = "canvas.jpg";  
    aTag.href = imagePath;  
    aTag.click(); })
```

# **DEPLOYMENT**

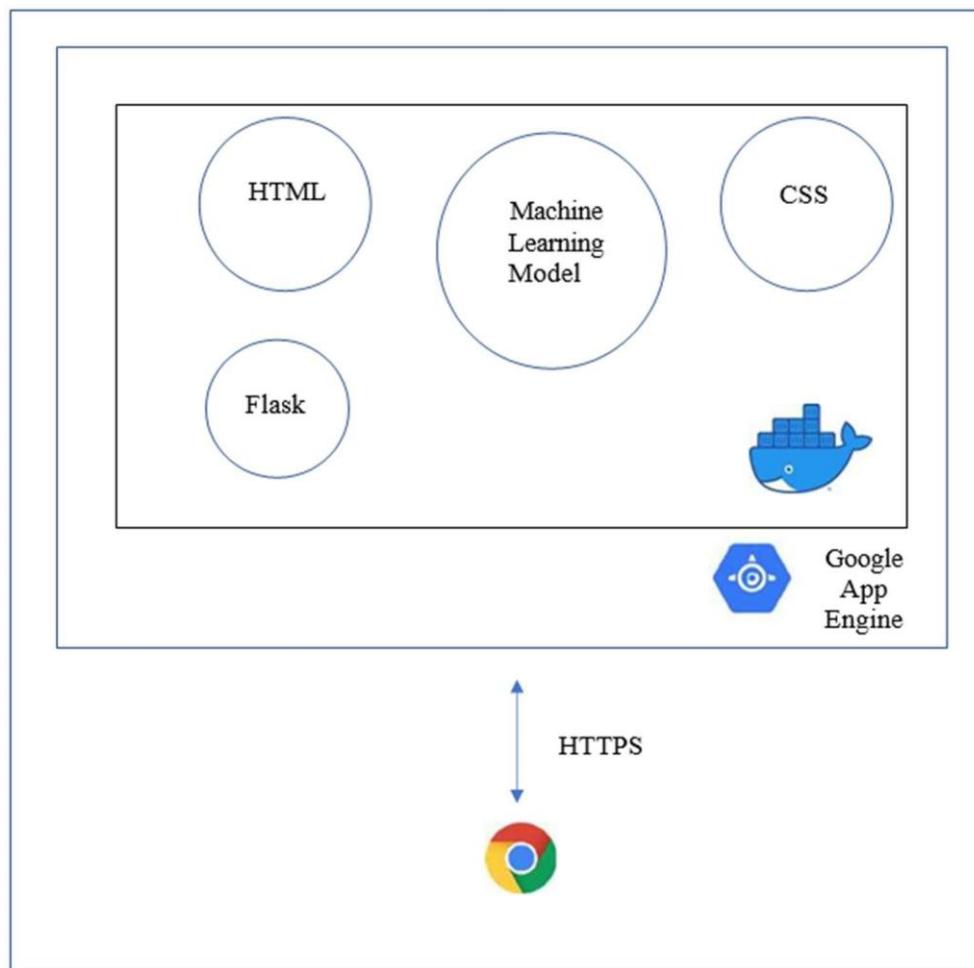
Software deployment includes all of the steps, processes, and activities that are required to make a software system or update available to its intended users. Today, most IT organizations and software developers deploy software updates, patches and new applications with a combination of manual and automated processes. Some of the most common activities of software deployment include software release, installation, testing, deployment, and performance monitoring.

Software development teams have innovated heavily over the past two decades, creating new paradigms and working methods for software delivery that are designed to meet the changing demands of consumers in an increasingly connected world. In particular, software developers have created workflows that enable faster and more frequent deployment of software updates to the production environment where they can be accessed by users.

## **Advantages of Cloud Deployment:**

- Faster and simplified deployments. Automate builds that deploy code, databases and application releases, including resource provisioning.
- Cost savings. Control costs using consumption-based pricing and eliminate capex-heavy on-premises environments.
- Platform for growth. Leverage the global infrastructure provided by cloud service providers (CSPs) to seamlessly expand the business into other geographies.
- New digital business models. Exploit the continuous release of features and services by CSPs, incubate new technologies and innovate digital business models.
- Business resiliency. Architect for the availability and fault-tolerance CSPs offer and ensure disaster recovery and business continuity of applications to make the business resilient.
- Agility and scalability. Use autoscaling and scalability to meet peak demands of the business without provisioning for excess capacity.
- Geographic reach. Access applications from any location, on any device, leveraging the connectivity backbone of CSPs.
- Operational efficiency. Use the inherent automation enabled by cloud to increase operational efficiency and reduce human effort.
- A competitive edge. Leverage infrastructure as code and development, security and operations to reduce the time to market for new features and stay ahead of the competition.
- Empowered users. Increase productivity by empowering users with self-service options on cloud, such as portals, DevOps pipelines, and executive and operational dashboards.

For our application we will be using Google App Engine for deployment of the application which provides a platform as a service to host our applications



**Fig: Deployment of the application on Google App Engine**

### **Deploying backend on google app engine**

- Create a new project in the Google Cloud Console. Once you have created a new project, you should enable the App Engine API for your project.
- Set up a local development environment by installing the Google Cloud SDK, which includes the gcloud command-line tool that you'll use to deploy your app to App Engine.
- Create a new Node.js app. You can use the Express.js framework or any other framework you prefer. Make sure that your app listens to a specific port, such as 8080.
- Create an app.yaml file in the root directory of your app. This file is used to configure your App Engine app and its deployment settings. A basic app.yaml file for a Node.js app might look something like this:

```
runtime: nodejs14
```

```
env: flex
```

```
handlers:
```

```
- url: /*
```

```
script: auto
```

- The runtime field should be set to nodejs14, and the env field should be set to flex, which will automatically scale your app based on incoming traffic. The handlers field is used to specify how URLs are handled by the app, in this case, any URL request will be handled by the script file.
- Test your app locally by running the Node.js server and visiting the localhost address in the browser.
- Deploy your app to App Engine using the gcloud command-line tool. Make sure that you are authenticated and that your project is set to the correct project using gcloud command-line and then run the command

### **gcloud app deploy**

- This command will deploy your application to App Engine and you'll see the details of the process on the command line.
- Once the deployment is complete, your Node.js backend will be live and accessible to users via a URL provided by App Engine. You can check the Application URL in the Cloud Console under the App Engine's dashboard.

## **Deploying frontend on Netlify**

- Build your React application using a command such as npm run build, this command creates a production-ready version of your app in a directory called build.
- Initialize a new git repository in the root of your project by running git init
- Connect your git repository to your Netlify account by running netlify init or you can connect it from the website by creating a new site from git.
- Add the build folder to your git repository and push it to your connected remote repository.

```
git add .
```

```
git commit -m "Initial Deployment"
```

```
git push
```

- Netlify will detect the new push to your repository and start building and deploying your React app.
- Once the deployment is complete, your React frontend will be live and accessible to users via a URL provided by Netlify. You can see the URL from your site dashboard on the Netlify website.
- You can further configure the build process and the environment of your site by adding a `netlify.toml` file. This file is used to configure the build settings and environment variables.
- You can also configure your custom domains, continuous deployment, and more by using Netlify's features.

## **CONCLUSION & FUTURE ENHANCEMENTS**

Adding support for multimedia: this includes adding the ability to take photos and videos within the app, and attaching them to notes. It could also include the ability to record audio and embed it within notes. Improving organization and categorization: this could include adding tagging and categorization functionality to notes, as well as the ability to create notebooks or folders to group related notes together. Adding note-sharing functionality: this could include the ability to share notes with others, either through a built-in sharing feature or by integrating with other services like email or social media. Enhancing search capabilities: this could include making the search feature more powerful and intuitive, and allowing users to search for specific keywords or phrases within notes. Incorporating machine learning/AI functionality: for example, the app could use machine learning to suggest notes or categorize them based on their contents, and also suggest notes based on their previous notes Improving note editing: This could include adding the ability to format text, create checklists, and add diagrams or other multimedia elements to notes. Are some of the features that can be added in the future to this project.

## **REFERECNES**

- <https://www.youtube.com>
- <https://www.google.co.in>
- <https://www.wikipedia.org>
- <https://cloud.google.com/appengine/docs/standard/python/getting-started/hosting-a-static-website>
- <https://happycoding.io/tutorials/java-server/hosting-google-app-engine>

