

INTERNSHIP REPORT

(Feb 2022-June 2022)

Internship report submitted in partial fulfilment of the requirement for the
degree of Bachelor of Technology

in

Computer Science and Engineering

By

Akhilesh

181201



Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat, 173234,

Himachal Pradesh, INDIA

DECLARATION

I hereby declare that this submission is my own work carried out at Cognizant Technology Solutions Corporation, Noida from February 2022 to June 2022 and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma from a university or other institute of higher learning, except where due acknowledgment has been made in the text.

Submitted by:

Akhilesh

181201

Computer Science & Engineering and Information Technology Department
Jaypee University of Information Technology

ACKNOWLEDGEMENT

I would like to thank Mrs. Anjali and all the people that worked along with me at Cognizant Technology Solutions Corporation, Noida with their patience and openness. They created an enjoyable working environment.

I also would like to thank Mr. Reejesh for mentoring me throughout my internship.

It is indeed with a great sense of pleasure and immense sense of gratitude that I acknowledge the help of these individuals.

I am highly indebted to Mr. Pankaj Kumar, Training & Placement Coordinator of our college for the facilities provided to accomplish this internship. I would also like to thank the Head of our Department Dr.Vivek Kumar Sehgal and the faculty for teaching us the skills required for this internship.

Finally, I must acknowledge the constant support and patients of my parents.

Akhilesh

181201

Jaypee University of Information Technology

TABLE OF CONTENT

Content	Page No.
Declaration By Candidate	I
Acknowledgement	II
List of Figures	IV
Chapter 01: INTRODUCTION	1
1.1 Introduction	1
1.2 Job Description	1
1.3 Objectives	2
1.4 Internship Schedule	2
Chapter 02: COMPANY DESCRIPTION	3
2.1 Cognizant	3
Chapter 03: TOOLS AND TECHNOLOGIES USED	6
3.1 React	6
3.2 Github	9
3.3 Postman	10
3.4 C#	11
3.5 .NET	12
3.6 ASP.NET	12
3.7 AWS	13
Chapter 04: LIVE PROJECT	15
4.1 Description	15
4.2 Steps	18
4.4 Code	21
4.5 Results	29
Chapter 05: CONCLUSION	31
5.1 Conclusion	31
REFERENCES	32
PLAGIARISM REPORT	33

LIST OF FIGURES

Fig no.	Figures	Page no.
1	About Cognizant	3
2	React App	7
3	C# Code snippet	12
4	Input Image	15
5	Expected Output	16
6	Result Image 1	29
7	Result Image 2	29
8	Result Image 3	30

Chapter 01: INTRODUCTION

1.1 Introduction

An internship is a professional teachable moment that gives students real-world experience in their field of study or professional goals. An internship enables students to learn new skills while exploring and enhancing their careers. This report is a description of my ongoing internship at Cognizant Technology Solutions Corporation. This internship report details the activities that helped me achieve a bunch of my stated objectives. I was assigned the profile of “.NET and Full Stack Developer” for my internship. First 2 months were learning of the basic concepts of engineering and web development following the 3rd month based on a company project.

1.2 Job Description

The introduction of engineering concepts to the creation, implementation, and technical management is known as software engineering. Software engineering was designed to address the challenges of low-quality software initiatives. Problems arise when software exceeds schedules, budgets, and quality expectations. It assures that the software is built in a consistent, correct, timely, cost-effective, and specification-compliant manner. Software engineering became vital to keep up with the rapid changing of user needs and the context in which the program is expected to operate.

The Software Engineer Trainee is responsible to assist in the design and development of software. The Software Engineer Trainee works with other members of the team to develop secure and dependable software solutions. The duties and responsibilities of this position are as follows:

- Application development (coding, programming)
- Code debugging and testing
- New software application documentation and testing
- Investigating, diagnosing, and resolving a wide range of technical challenges
- Working with senior executives
- Identifying issues and developing solutions
- Learning about new technology ahead of time

1.3 Objectives

The objective of the internship is to learn more about the programming languages like Node.js, Express.js, MongoDB, and ReactJS. These four languages are the basic requirements for my job profile. After the learning process, I was assigned to work on a live project for the company. This internship provides experience to the freshers so that they can learn more about the industry. The perspective was to gain enough knowledge so that we can work easily on the project assigned to us.

1.4 Internship Schedule

The internship plan was as follows:

➤ March 2022

This month, I revised the HTML, CSS, and Javascript which was followed by learning about Bootstrap. I was assigned to develop simple web page using HTML, CSS and JavaScript

➤ April 2022

I studied ReactJS for frontend development in April, and I was assigned to create simple projects.

➤ May 2022

This month I was assigned for a task in the Business Unit. I had to design the front-end components of a website using React.

➤ June 2022

In this coming I will be learning C#, .NET and ASP.NET.

Chapter 02: COMPANY DESCRIPTION

2.1 Cognizant

Cognizant is an American MNC service and consulting company headquartered in Teaneck, New Jersey, United States. Cognizant provides IT consulting and business for industries like banking, health care, manufacturing, media and entertainment.

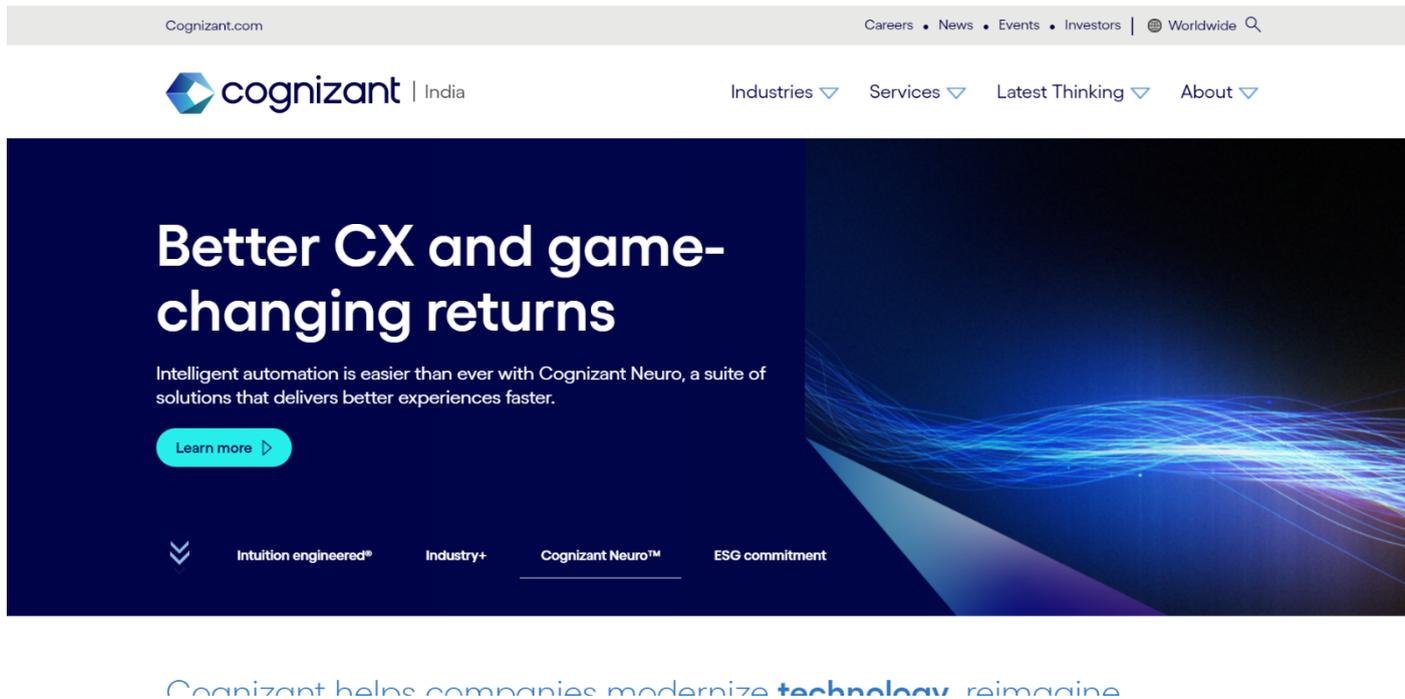


Figure 1. About Cognizant

2.1.1 Our Services

Cloud Enablement - Wherever you are on your cloud transformation journey, Cognizant is there to support you. We collaborate with you to build the best approach for your organisation so you can reap the full benefits and value of cloud computing. Our vision, industry experience, and client-centric approach deliver end-to-end cloud services and innovative solutions that result in short-term success, measurable business outcomes, and happy customers. Learn more about how Cognizant can help you accelerate your cloud journey.

Application modernization services-Application modernization services cover the migration of legacy applications or platforms to new applications or platforms, as well as the integration of new functionality to give the business the most up-to-date features.

Data, Analytics and AI- Artificial intelligence (AI) analytics is a subset of business intelligence that employs machine learning techniques to uncover new patterns and relationships in data. In practice, AI analytics is the process of automating much of the work that would normally be performed by a data analyst.

2.2 History of Cognizant

Cognizant was founded as Dun & Bradstreet Satyam Software (DBSS), with Srini Raju as the original CEO and MD, as Dun & Bradstreet's in-house technology arm focused on completing huge IT projects for Dun & Bradstreet enterprises. The company began targeting customers outside of Dun & Bradstreet in 1996.

Dun & Bradstreet split up some of its companies in 1996, including Erisco, IMS International, Nielsen Media Research, Pilot Software, Strategic Technologies, and DBSS, to form Cognizant Corporation, based in Chennai, India. DBSS changed its name to Cognizant Software Solutions months later, in 1997. Dun & Bradstreet paid \$3.4 million for Satyam's 24 percent ownership in DBSS in July 1997. The company's headquarters were relocated to the United States, and Kumar Mahadeva was chosen CEO in March 1998. The company focuses on Y2K-related tasks and web development as a part of Cognizant Corporation.

Cognizant Corporation, the parent corporation, divided into two entities in 1998: IMS Health and Nielsen Media Research. Cognizant Technology Solutions had become a public unit of IMS Health after this restructuring. IMS Health partially split out the company in June 1998, launching a public offering of Cognizant stock. The firm can raise \$34 million, which was less than the underwriters at IMS Health had hoped for. They set aside the funds for debt repayment and office upgrades.

Kumar Mahadeva opted to lessen the company's reliance on Y2K contracts, with Y2K projects accounting for only 26% of revenues in Q1 1999, compared to 49% in early 1998. Kumar Mahadeva opted to avoid large-scale ERP deployment projects because he believed the \$16.6 billion enterprise software market was saturated. Instead, he concentrated on application management, which contributed for 37% of Cognizant's sales in the first quarter of 1999. In 2002, Cognizant's revenues were \$229 million, and it had no debt and \$100 million in total. The company flourished during the dotcom bust by taking on maintenance work that major IT services companies didn't want.

IMS Health bought its entire 56 percent ownership in Cognizant in 2003, and the company

implemented a poison pill policy to avoid aggressive takeovers. In 2003, Kumar Mahadeva stepped down as CEO and was succeeded by Lakshmi Narayanan. The company's products portfolio gradually expanded beyond IT to include outsourced (BPO) and consulting firms. Francisco D'Souza succeeded Lakshmi Narayanan in 2006. Cognizant achieved rapid expansion in the 2000s, as evidenced by its ten consecutive appearances on Fortune magazine's "100 Fastest-Growing Companies" list from 2003 to 2012.

Cognizant made its biggest acquisition in September 2014, when it paid \$2.7 billion for healthcare IT services business TriZetto Corp. In pre-market trading, shares of Cognizant surged about 3%..

The company struck a multimillion-dollar arrangement with Escorts Group in India on June 24, 2015, to assist Escorts' enterprises with digital transformation and modernization across all business areas.

It teamed with Singapore-based market chain NTUC FairPrice on 30 June 2015 to digitally modernise NTUC's operations in order to improve personalised and consistent customer experience across numerous channels.

Cognizant and a group of Indian life insurers stated in April 2018 that they were working on a blockchain solution to improve productivity by allowing cross-company data sharing. The platform, which is based on Corda, a distributed ledger technology platform developed by R3, is said to eliminate reliance on third-party data brokers and aggregators for KYC reasonable care, clinical and financial assessment, risk assessment, fraud detection, and regulatory compliance.

Cognizant sold its Oy Samlink acquisition to Kyndryl in January 2022.

Chapter 03: TOOLS AND TECHNOLOGIES

3.1 React

ReactJS is a Javascript framework for constructing declarative, fast, and adaptable user interface components. It's a component-based front-end framework that manages the app's view layer exclusively. Jordan Walke, a software engineer at Facebook, conceived the idea. Facebook invented and maintains it, and it's later used in Facebook products like WhatsApp and Instagram. Although ReactJS was designed around 2011 for Facebook's feed area, it was first publicly published in May 2013.

The most common website architecture today is MVC (model view controller). React is the view in the MVC design, whereas Flux or Redux is the architecture.

A ReactJS application is made up of numerous components, each being responsible for generating a reusable piece of HTML code. Components are the foundation of all React programs. Complex applications can be developed using simple building blocks by nesting these components with other components. ReactJS interconnects the DOM-based technique for populating data in the HTML DOM. Because the virtual DOM alters specific DOM components rather than refreshing the full DOM, it is fast.

To create a React app, we create React components that correspond to various aspects. The application structure is made up of these components, which are arranged into higher-level components. Consider a form containing the required fields, labels, and buttons, among other things. Each form component can be created as a React element, which we then combine to build the form component. The form elements would describe the structure and parts of the form.

ReactJS' major purpose is to construct a User Interface that aids program performance. It makes use of a virtual DOM (JavaScript objects), that speeds up the app. In JavaScript, the virtual DOM is quicker than the traditional DOM. ReactJS can also be used on server and client sides, as well as with other frameworks. Component and data structures are used to frame and make app maintenance easier.

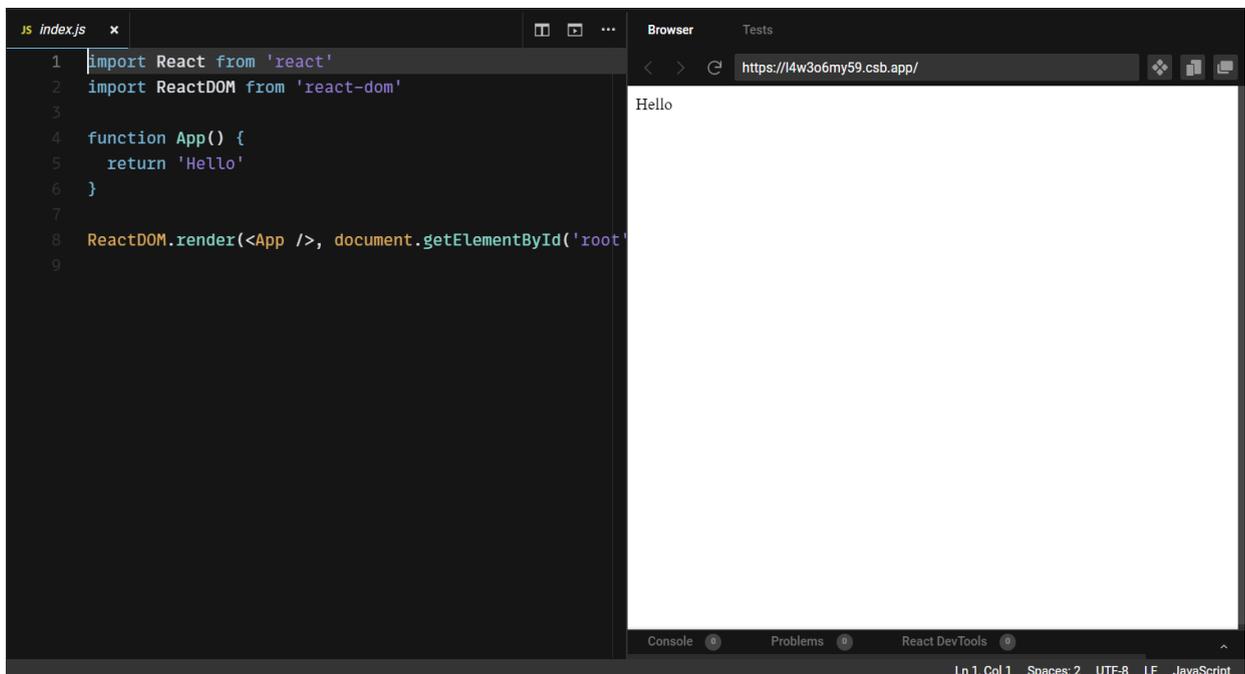


Figure 2. An example of a ReactJS application

3.1.1 Features of ReactJs

ReactJS is quickly becoming the most popular JavaScript framework among web developers. It is an important part of the front-end ecology. The following are some of ReactJS' key features:

- JSX

JavaScript XML is abbreviated as JSX. It's a syntax extension for JavaScript. ReactJS uses an XML or HTML-like syntax. This syntax is converted into React Framework JavaScript calls. It enhances ES6 to allow HTML-like text and JavaScript react code to coexist. It is not required to utilise JSX, however, it is strongly recommended in ReactJS.

- Components

Components are at the heart of ReactJS. A ReactJS application consists of several components, each with its logic and controls. These components are reusable, which makes it easier to keep the code clean while working on larger projects.

- One-way Data Binding

ReactJS is designed to handle one-way data binding or unidirectional data flow. One-way

data-binding allows you to have more control over your application. If the data flow occurs oppositely, additional characteristics are required. Because components are designed to be immutable, and the data they contain cannot be modified, this is the case. Flux is a pattern that aids in data unidirectionality. This increases efficiency by making the application more versatile.

- Virtual DOM

The original DOM object is represented by a virtual DOM object. It functions similarly to a one-way data binding. The complete UI is re-rendered in virtual DOM representation whenever any changes are made to the web application. Then it compares the differences between the old and new DOM representations. After that, the true DOM will only update the things that have changed. This speeds up the application and eliminates memory waste.

- Simplicity

ReactJS makes use of a JSX file, which makes the program easy to code and comprehend. We already know that ReactJS is a component-based solution that allows you to reuse code as needed. This makes it easy to use and understand.

- Performance

The speed of ReactJS is well-known. This distinguishes it from other frameworks currently available. This is due to the fact that it manages a virtual DOM. The Document Object Model (DOM) is a cross-platform computer programming API for HTML, XML, and XHTML. The DOM is purely memory-based. As a result, we didn't even write directly to the DOM while creating a component. Instead, we'll create virtual components that will be converted into the DOM, resulting in smoother and faster performance.

3.2 Github

Software developers and engineers can use GitHub to build public-facing cloud repositories for free. You can transfer a GitHub repository to your device, add and alter files locally, and then "push" your changes back to the repository, where they will be visible to the public.

Repository

A repository (sometimes shortened as "repo") is a site where all of the files for a project are kept. Each project has its repository, which you can access using a specific URL.

Forking a Repo

When you establish a new project based on an existing project, this is known as "forking." This is a fantastic tool that greatly fosters the development of new applications and projects. If you find a project on GitHub that you'd want to contribute to, you can fork it, make the modifications you want, and then republish it as a new repo. You may quickly add changes to your current fork if the original repository that you forked to create your new project is updated.

Git Commit

You're ready to commit once you've staged the files you want to add. The commit message is important regardless of whether you commit using GitHub Desktop or the command line. Short commit statements that describe your change should be used. The commit messages will lead you through the history of your repository, therefore they should be informative. The following message format can be used in command-line commits:

```
git commit -m "git commit message example"
```

Pull Requests

You've forked a repository, made a fantastic change to the project, and want the original devs to notice it—perhaps even put it in the original project/repository. Create a pull request to accomplish this. The original repository's authors can view your work and decide whether or not to accept it into the official project. When you submit a pull request, GitHub provides an excellent communication channel between you and the primary project's maintainer.

3.2 Postman

Postman is a well-known API testing tool. APIs may be simply created, tested, shared, and documented with the help of this tool. Postman is an Application Program Interface (API) testing tool that may be used to create, test, develop, change, and document APIs. It's a basic graphical user interface for executing and analysing HTTP requests and responses. When utilising Postman for testing, you don't need to write any HTTP connection code. Instead, we use Postman to connect with the API and create test suites called collections. Almost every feature a developer can require is included in this program. This tool can do GET, POST, PUT, and PATCH HTTP queries, as well as convert APIs.

Postman is built around a set of powerful technologies that are extremely easy to use. Postman has become a useful tool for more than 8 million users. Postman is used for the following reasons:

1. Accessibility- After installing Postman on the device, simply login to the account to use it anywhere.
2. Use Collections- Users can create collections for their API calls using Postman. Each set can create multiple requests and subfolders. It will aid in the organisation of test suites.
3. Test development- Every API request should include verification of a successful HTTP response status to test checkpoints.
4. Automation Testing- The Collection Runner or Newman can conduct tests in multiple repetitions or iterations, saving time for repeated tests.
5. Creating Environments- The usage of numerous environments reduces test replication because the same collection can be used in different settings.
6. Debugging- The postman console assists in successfully debugging the tests by tracking what data is being obtained.
7. Collaboration- To improve file sharing, you can import and export collections and environments. You can also share the collections via a direct connection.
8. Continuous integration- It can support continuous integration.

It's as simple as typing a URL into your browser to send a request. In Postman, we can quickly

send requests to APIs. You can use an API request to access or deliver data from a data source.

An HTTP method is required to send the API request. POST, GET, DELETE, PUT, and PATCH are some of the most widely used methods.

- GET: This HTTP method is used to access the data from an API.
- POST: This method transmits new data.
- DELETE: This is used to remove or delete the existing data.
- PATCH: This method is used to update the existing data.
- PUT: This method is used to update the existing data.

3.3 C#

C# is an object-oriented programming language developed by Microsoft for their .NET project. C# is very popular and high in demand for the following reasons:-

- C# is a high-level language that is similar to other popular programming languages such as C, C++, and Java, making it simple to learn for everyone.
- C# is commonly used to create desktop and web apps. It is one of the most widely used languages on professional desktops. If you want to make Microsoft programmes, C# is the language to use.
- Community: The greater the community, the better, as new tools and software are developed to improve it. Because C# has a big community, efforts are made to ensure that it continues to exist in the system and does not become extinct.
- C# is a popular game development language that will continue to grow in popularity. Because C# is integrated with Microsoft, it has a broad audience. C# is a popular game development language because of features like Automatic Garbage Collection, interfaces, and object-oriented programming.

A simple addition program in C#:-

```

1  using System;
2
3
4  namespace ConsoleApp1
5  {
6      0 references
7      internal class Program
8      {
9          0 references
10         static void Main(string[] args)
11         {
12             int firstNum, secondNum;
13             int sum;
14             Console.Write("Enter First number");
15             firstNum = Convert.ToInt32(Console.ReadLine());
16             Console.Write("Enter Second number");
17             secondNum = Convert.ToInt32(Console.ReadLine());
18
19             sum = addition(firstNum, secondNum);
20             Console.WriteLine("Sum is " + sum);
21             Console.Read();
22         }
23
24         1 reference
25         public static int addition(int a,int b)
26         {
27             return a+b;
28         }
29     }
30 }

```

Figure 3: An example C# code

3.4 .NET

Microsoft's .NET framework is a software framework that they designed and built. The initial version of the .NET framework, 1.0, was released in 2002. It is a virtual machine for compiling and executing programmes written in various languages such as C#, VB.Net, and others.

Form-based apps, Web-based applications, and Web services are all built with it. The .NET platform supports a number of programming languages, the most popular of which being VB.Net and C#. It is used to create programmes for Windows, phones, the web, and other platforms. It has a lot of features and also adheres to industry standards.

The .NET Framework supports more than 60 programming languages, with Microsoft designing and developing 11 of them.

With .NET, you can use multiple languages, editors, and libraries to build for web, mobile, desktop, games, and IoT.

3.5. ASP .NET

ASP.NET is a server-side web application framework that is open-source and designed to build dynamic web pages. Microsoft created it to help programmers create dynamic web pages, applications, and services. Active Server Pages Network Enabled Technologies is the name of the technology.

It is the successor to Microsoft's Active Server Pages (ASP) technology and was first launched in January 2002 with version 1.0 of the .NET Framework. The Common Language Runtime (CLR) underpins ASP.NET, allowing programmers to build ASP.NET code in any supported .NET language. ASP.NET components can process SOAP messages using the ASP.NET SOAP extension framework.

ASP.NET Core is the successor of ASP.NET. It's a modular web framework that combines ASP.NET and additional frameworks like Entity Framework. The new framework is cross-platform and uses the new open-source .NET Compiler Platform (codename "Roslyn"). ASP.NET MVC, ASP.NET Web API, and ASP.NET Web Pages (a platform that solely uses Razor pages) have all been combined into a single MVC 6 platform.

3.6 AWS

Amazon provides a fantastic suite of web services that enables developers to create dynamic and robust applications. Deploying on AWS can save you time, money and manpower compared to building and maintaining more traditional systems.

Amazon Web Services delivers a number of benefits for IT organisations and developers alike, including:

- **Cost-effective:** Pay only for what you use, as you use it, with no up-front commitments. As the Amazon Web Services cloud grows, our operations, management and hardware costs shrink, and we pass the savings onto you.
- **Dependable:** Utilise a battle-tested, web-scale infrastructure that handles whatever you throw at it. The Amazon Web Services cloud is distributed, secure and resilient, giving you reliability and massive scale.
- **Flexible:** Build any application you want using any platform or any programming model. You control the resources you consume and fit them into your application as you see fit.
- **Comprehensive:** Don't start from scratch. Amazon Web Services gives you a number of services you can incorporate into your applications. From databases to payments, these services help you build great applications cost effectively and with less up-front investment.

Amazon Simple Storage Service (S3): Amazon S3 is storage for the Internet. It is designed to make web-scale computing easier for developers. Amazon S3 provides a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of websites. The service aims to maximise benefits of scale and

to pass those benefits on to developers.

Amazon Elastic Compound (EC2): EC2 is the computing part of the Amazon services. EC2 provides the CPU, memory, operating system and transient storage. EC2 is the equivalent of a barebones PC. You get to pick the amount of RAM you need (from a predefined list of configurations), the amount of transient storage you need (also from a list) and the number of CPUs you need (from a series of compute options). For the operating system, you can choose from various flavours of Linux, Solaris or Microsoft Windows Server. The basis of EC2 is the Amazon Machine Image (AMI). An AMI is a virtual machine with your chosen operating system and applications bundled together. You can create your own AMIs from scratch if you want to. To get started though, Amazon offers hundreds of public AMIs with many operating systems and pre-installed applications.

Amazon CloudWatch: Amazon CloudWatch is a web service that provides monitoring for AWS cloud resources, starting with Amazon EC2. It provides customers with visibility into resource utilisation, operational performance, and overall demand patterns—including metrics such as CPU utilisation, disk reads and writes, and network traffic. To use Amazon CloudWatch, simply select the Amazon EC2 instances that you'd like to monitor; within minutes, Amazon CloudWatch will begin aggregating and storing monitoring data that can be accessed using the AWS Management Console, web service APIs or Command Line Tools.

Chapter 04: LIVE PROJECT

TASK 1

4.1 Description

This task includes drawing on a canvas in which the AI will detect what you have drawn and based on that I've to draw the AI will generate a graph for you, if AI can't detect what you have drawn or you've drawn something wrong then I have to send data to frontend and display the error.

Expected Input-

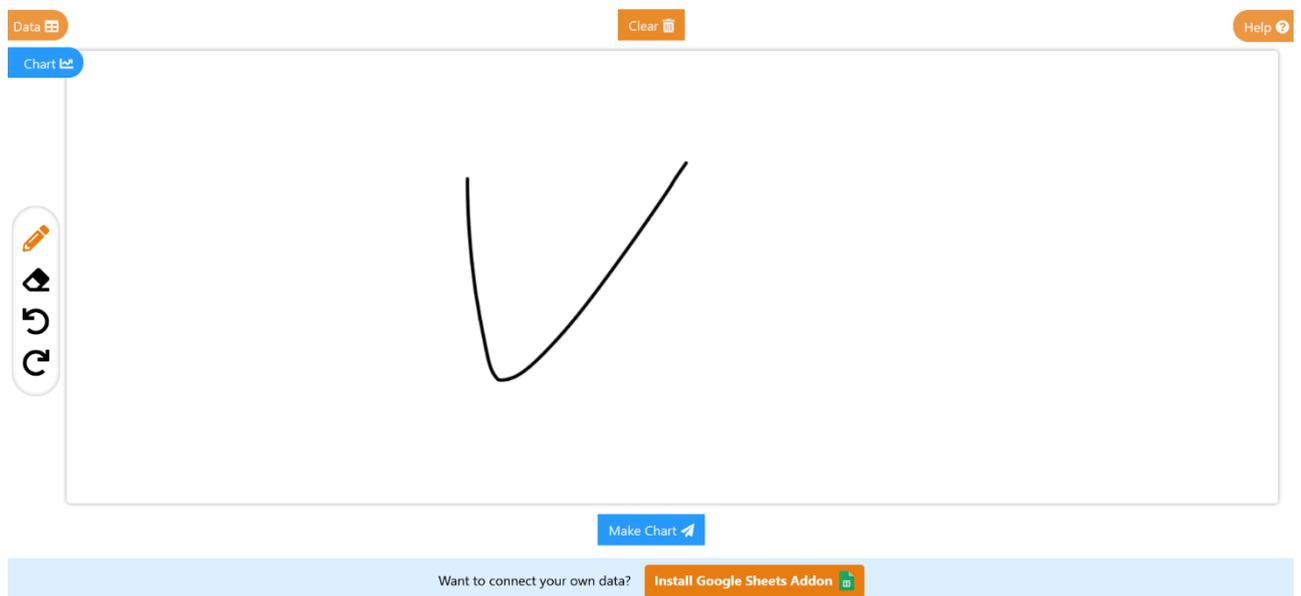


Figure 4: Input image

Expected Output-

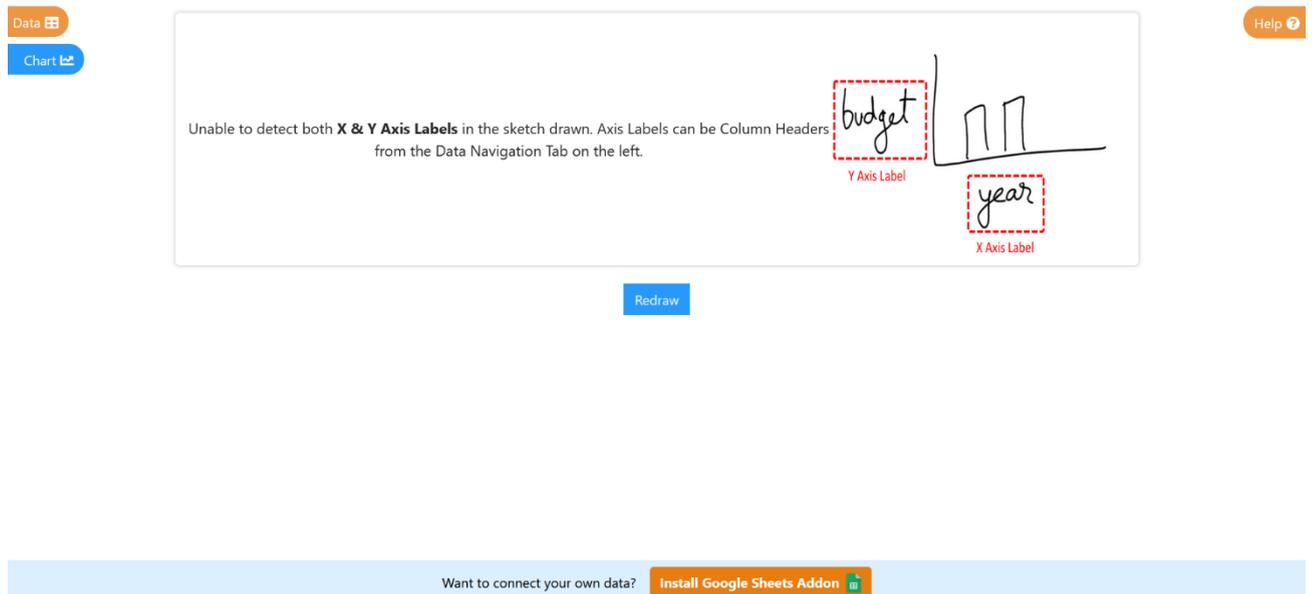


Figure 5: Expected output

4.2 Tech Stack Used

- Amazon EC2 - Amazon Elastic Cloud known as EC2 is a part of amazon cloud services that allows users to run virtual computers on their system with high performance. We have used EC2 for backend purposes to run AI operations in the backend.
- Amazon S3 - Amazon S3 is amazon's database storage. We have used amazon s3 to store data that we have drawn in .png format.
- React – For our frontend purpose we've used React to display content.
- Redux – To manage react state globally we have used redux.

4.2.1 Importing necessary libraries

Starting with importing libraries, to use React and to manage state we have to use useState hook. To use canvas I've imported the Canvas component and to use redux I've imported react-redux.

```

import React, { useState } from "react";
import Canvas from "./Canvas";
import { connect } from "react-redux";
import Spinner from "../utilities/Spinner";
import Plot from "../content/Plot";
import Alert from "../utilities/Alert";
import LoadText from "../utilities/LoadText";
import yAxisError from "../../assets/images/Yaxiserror.png";
import xyAxisError from "../../assets/images/XYaxiserror.png";
import xAxisError from "../../assets/images/Yaxiserror.png";
import { clearImageData, clearImageError } from "../../actions/imageActions";
import { processData } from "../../actions/dataActions";

```

4.2.2 Getting backend response

After drawing the graph, the backend will send a response, to retrieve the response and to store the value we are using redux, here we are getting the response and storing them in imageData variable.

```

const DrawCharts = ({
  imageData,
  clearImageData,
  clearImageError,
  processData,
  showRating,
}) => {
  const [showAlert, setShowAlert] = useState({
    show: false,
    message: "",
  });
};

```

4.2.3 Steps

After retrieving the response there can be three possibilities :-

1. **imageData outdated** :- After drawing we might have updated the data, that is the .csv file uploaded to the server. Then we have to redraw the image.

```
// callback function called by plot, when imagedata is outdated
const reprocessData = () => {
  processData();
  displayAlert(
    "Your Data seems to have updated recently. You might want to redraw"
  );
};
```

2. **Uploading** :- After drawing we have to show some graphics while in the backend AI is working his job.

```
// loading
if (imageData.loading) {
  return (
    <div className="drawCharts-spinner-container">
      <Spinner />
      <p className="status" style={{ fontSize: "1.1rem", opacity: "0.9" }}>
        {imageData.uploading ? (
          <span>Uploading Your Sketch...</span>
        ) : imageData.processing ? (
          <LoadText />
        ) : (
          ""
        )}
      </p>
    </div>
  );
}
```

3. **Error** :- If error received from the backend then we have to get the error code and display the information according to that error

```
//get error
const getImageDataError = () => {
  let imageDataError = imageData.imageData.message.split("$");
  return imageDataError[0];
};
//get error code
const getImageDataErrorCode = () => {
  let imageDataArray = imageData.imageData.message.split("$");
  if (imageDataArray[1] == "0") return xyAxisError;
  else if (imageDataArray[1] == "1") return xAxisError;
  else return yAxisError;
};
```

4. **Display** :- After getting the error then we have to display on page with that image.

```
// success
if (imageData.imageData) {
  if (!imageData.imageData.success) {
    return (
      <React.Fragment>
        <div className="container text-center mt-4 shadow-container error-box">
          <p
            className="plot-error"
            dangerouslySetInnerHTML={{
              __html: getImageDataError(),
            }}
          ></p>
          <img
            src={getImageDataErrorCode()}
            style={{ height: "300px", width: "350px" }}
          />
        </div>
        <div
          className="btn-container1 text-center"
          style={{ marginTop: "15px" }}
        >
          <button
            className="primary-btn btn redraw"
            onClick={() => {
              clearImageError();
            }}
          >
            Redraw
          </button>
        </div>
      </React.Fragment>
    );
  }
}
```

```

return (
  <div style={{ marginTop: "15px" }}>
    <div className="btn-container text-center">
      <button
        className="primary-btn btn redraw"
        onClick={() => {
          clearImageData();
        }}
      >
        Redraw
        <i style={{ marginLeft: "5px" }} className="fas fa-sync"></i>
      </button>
      <button
        className="primary-btn btn redraw"
        onClick={() => {
          clearImageError();
        }}
        style={{ marginLeft: "1rem" }}
      >
        Edit Sketch
        <i style={{ marginLeft: "5px" }} className="fas fa-edit"></i>
      </button>
    </div>
    <Plot plotData={imageData.imageData} reprocessData={reprocessData} />
  </div>
);

```

TASK 2

4.1 Description

In this task I had to create a Data Tab which was a kind of feedback form in which we get information from the user about what kind of data they want to integrate and then I had to send mail to the specified email address.

4.2 Code

- Use hooks to create a portion of dataTab to display accordingly to a given situation.

```
Akhilesh, 4 months ago | 1 author (Akhilesh)
1  import React, { useState } from "react";
2  import "../assets/css/popup.css";
3  import gsheetsLogo from "../assets/images/gsheets.png";
4  import dataLogo from "../assets/images/database.png";
5  import excelLogo from "../assets/images/excel.png";
6  import appLogo from "../assets/images/apps.png";
7  import { Row, Col } from "react-bootstrap";
8  import { Button, Form } from "react-bootstrap";
9
10 const Popup = () => {
11   const [showPopup, setShowPopup] = useState(true);
12   const [sheetContent, setSheetContent] = useState(false);
13   const [otherContent, setOtherContent] = useState(false);
14   const [showSubmitted, setShowSubmitted] = useState(false);
15   const [showContent, setShowContent] = useState(true);
16   const [emailContent, setEmailContent] = useState(false);
17   const [input, setInput] = useState("");
18   const [dataSource, setDataSource] = useState("");
19   if (showPopup == false) return null;
20
21   const showSheetContent = () => {
22     setSheetContent(!sheetContent);
23     setOtherContent(false);
24     setShowSubmitted(false);
25   };
26   const showOtherContent = () => {
27     setOtherContent(!otherContent);
28     setSheetContent(false);
29     setShowSubmitted(false);
30   };
31   const toggleSubmitted = () => {
```

```

31  ✓  const toggleSubmitted = () => {
32      setOtherContent(false);
33      setSheetContent(false);
34      setShowContent(false);
35      setEmailContent(true);
36  };
37
38  ✓  const sendData = (e) => {
39  ✓    const message = `
40      <p>User EmailID: ${input}</p>
41      <p><strong>User wants to connect ${dataSource}</strong></p>
42  `;
43  ✓    window.Email?.send({
44        Host: "smtp.gmail.com",
45        Username: "dhwanil@drawcharts.co",
46        Password: "awdbsrtmenimnptc",
47        To: "help@drawcharts.com",
48        From: "help@drawcharts.com",
49        Subject: `Datasource Feedback `,
50        Body: message,
51  ✓    }).catch((e) => {
52        console.error(e.message);
53    });
54  };
55  ✓  const onSubmit = () => {
56      setShowSubmitted(true);
57      setEmailContent(false);
58      sendData();
59  }
60  return (
61  ✓  <div className="popup-prompt text-center">

```

```

60 return (
61   <div className="popup-prompt text-center">
62     <div className="overlay" id="popup-overlay"></div>
63     <div className="overlay-inner popup-inner">
64       <button
65         aria-label="Close"
66         className="btn close-rating rating-submit"
67         onClick={() => setShowPopup(false)}
68       >
69         <i className="fas fa-2x fa-times text-align-right"></i>
70       </button>
71       {showContent && (
72         <div>
73           <h2>Connect Data Source</h2>
74           <br />
75           <Row>
76             <Col>
77               <div
78                 className="bn3"
79                 onClick={showSheetContent}
80                 variant="warning"
81                 style={{
82                   backgroundColor: sheetContent ? "#ff8000" : "transparent",
83                 }}
84               >
85                 <img
86                   className="sheets-logo"
87                   src={gsheetsLogo}
88                   alt="google sheets"
89                 />{" "}
90                 Google Sheets{" "}

```

```

90     Google Sheets{" "}
91   </div>
92 </Col>
93 <Col>
94   <div
95     className="bn3"
96     onClick={showOtherContent}
97     variant="warning"
98     style={{
99       backgroundColor: otherContent ? "#ff8000" : "transparent",
100    }}
101   >
102     <img className="sheets-logo" src={dataLogo} alt="database" />{" "}
103     Other Data Source{" "}
104   </div>
105 </Col>
106 </Row>
107 </div>
108 )}
109 {sheetContent == true && (
110   <div className="selected-item">
111     <ul>
112       <li>
113         <p>
114           1. Install the{" "}
115           <a
116             href="https://workspace.google.com/marketplace/app/drawcharts/728433098170"
117             target="_blank"
118           >
119             DrawCharts addon for google sheets
120           </a>{" "}

```

```

116 < a href="https://workspace.google.com/marketplace/app/drawcharts/728433098170"
117     target="_blank"
118     >
119     DrawCharts addon for google sheets
120 < /a>{" "}
121 < /p>
122 < /li>
123 < li>
124 < p>
125     {" "}
126     2. Open your{" "}
127 < a
128     href="https://docs.google.com/spreadsheets/u/0/?tgif=d"
129     target="_blank"
130     >
131     Google Sheets{" "}
132 < /a>
133 < br />
134     and click Extensions &rarr; DrawCharts &rarr; Create Chart
135 < /p>
136 < /li>
137
138 < li>
139     < p>3. Click "Validate" and "Open in Drawcharts"< /p>
140 < /li>
141 < /ul>
142 < /div>
143 )}
144 {emailContent == true && (
145 < div>
146 < Form>

```

```

144 {emailContent == true && (
145     <div>
146         <Form>
147             <Form.Group className="mb-3" controlId="formBasicEmail">
148                 <Form.Label>Email address</Form.Label>
149                 <Form.Control
150                     type="email"
151                     placeholder="Enter email"
152                     value={input}
153                     onInput={(e) => setInput(e.target.value)}
154                     required
155                 />
156                 <Form.Text className="text-muted">
157                     We'll never share your email with anyone else.
158                 </Form.Text>
159             </Form.Group>
160             <Button
161                 className="btn btn-primary install-btn ml-3"
162                 variant="primary"
163                 type="submit"
164                 onClick={onSubmit}
165             >
166                 Submit
167             </Button>
168         </Form>
169     </div>
170 )}
171 {otherContent == true && (
172     <div className="selected-item">
173         <ol>
174             <li>
175                 <Button

```

```

71 {otherContent == true && (
72   <div className="selected-item">
73     <ol>
74       <li>
75         <Button
76           onClick={() => {
77             toggleSubmitted();
78             setDataSource("Database");
79           }}
80           className="buttons"
81           variant="warning"
82           size="sm"
83         >
84         <img className="sheets-logo" src={dataLogo} alt="database" />{" "}
85         Database
86       </Button>
87     </li>
88     <li>
89       <Button
90         onClick={() => {
91           toggleSubmitted();
92           setDataSource("Excel");
93         }}
94         className="buttons"
95         variant="warning"
96         size="sm"
97       >
98       <img className="sheets-logo" src={excelLogo} alt="excel" />{" "}
99       Excel
100     </Button>
101   </li>

```

```

220     {showSubmitted && (
221       <div>
222         <React.Fragment>
223           <h1 className="rating-heading mb-4">Thank You!</h1>
224           <p>
225             Your response has been submitted.
226             <br />
227             We'll get back to you and help connect your datasource{" "}
228           </p>
229           <div className="success-checkmark">
230             <div className="check-icon">
231               <span className="icon-line line-tip"></span>
232               <span className="icon-line line-long"></span>
233               <div className="icon-circle"></div>
234               <div className="icon-fix"></div>
235             </div>
236           </div>
237         </React.Fragment>
238       </div>
239     )}
240   </div>
241 </div>
242 );
243 };
244

```

4.5 Results

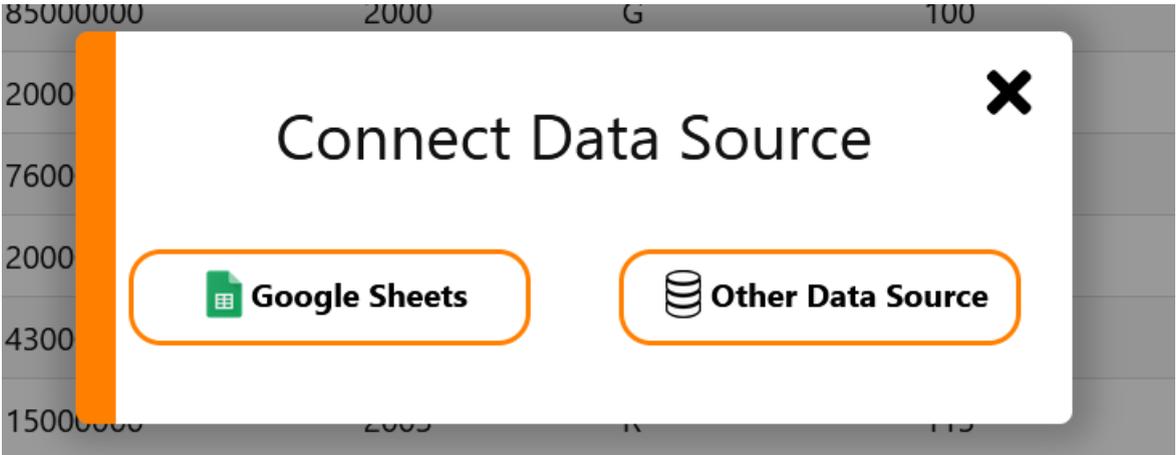


Figure 6: Result 1

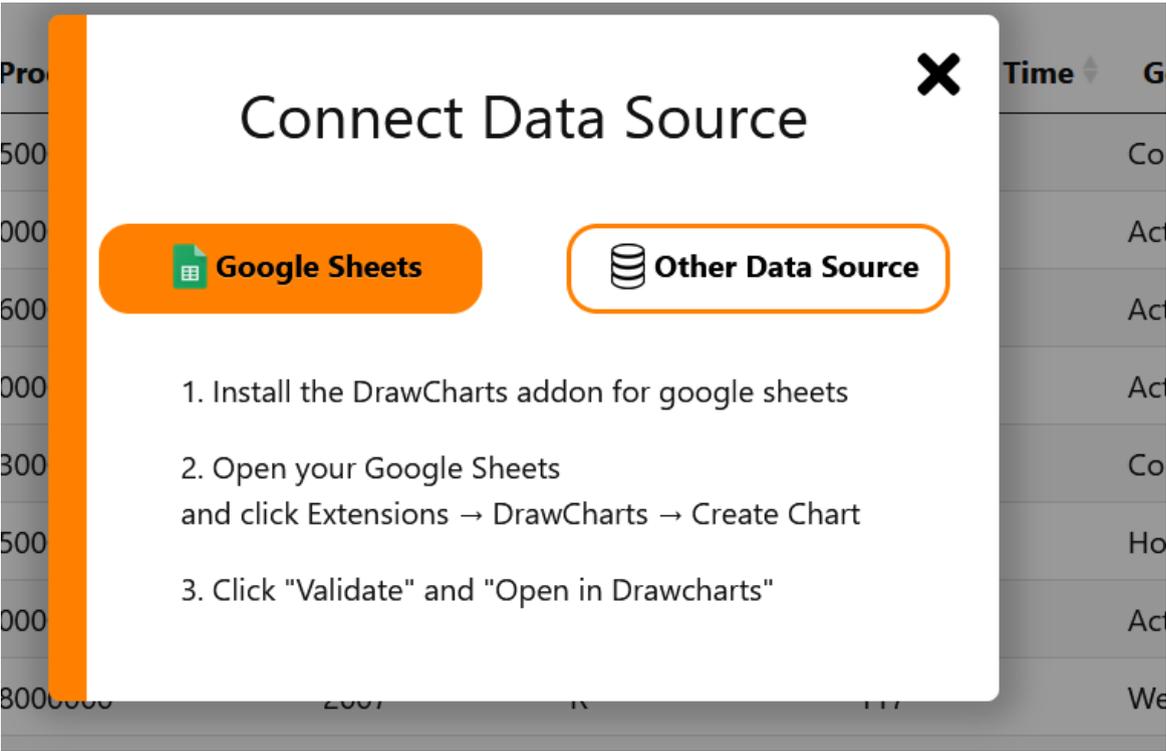


Figure 7: Result 2

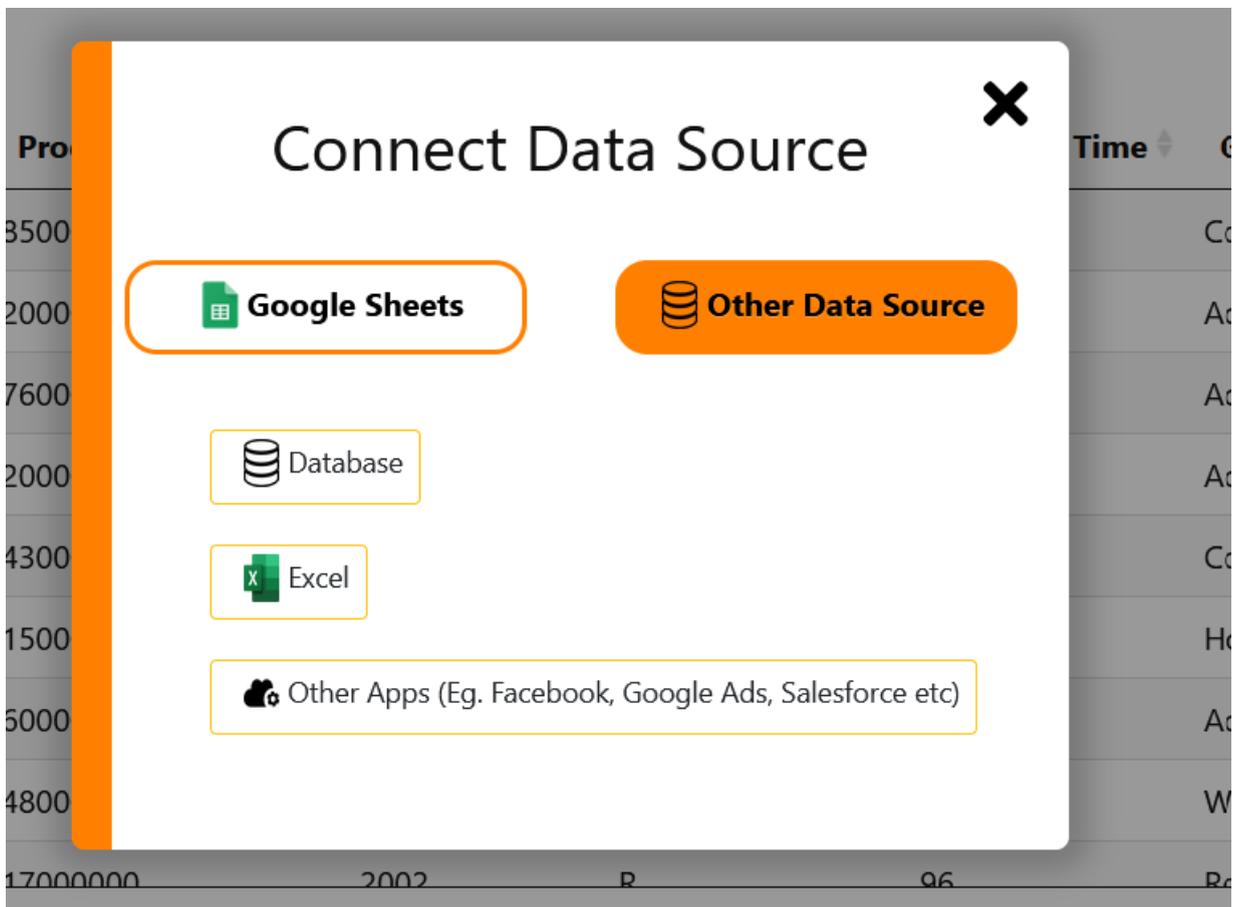


Figure 8: Result 3

Chapter 05: CONCLUSION

5.1 Conclusion

In conclusion, this internship has become a wonderful and rewarding experience. I can say that my stay with Cognizant Company was quite beneficial to me. I was offered a job as a .NET full stack developer and I must say I've learned a lot of new things and still the internship is going on yet I have to study about ASP.NET which is very exciting.

I also got the opportunity to do a Business project which was very interesting and got to learn a lot of new things by working on that project.

Overall it was a very rewarding experience.

REFERENCES

1. <https://www.w3schools.com/html/default.asp>
2. <https://www.w3schools.com/css/default.asp>
3. <https://www.w3schools.com/js/default.asp>
4. <https://www.w3schools.com/nodejs/>
5. <https://www.youtube.com/watch?v=SccSCuHhOw0>
6. <https://www.npmjs.com/package/express>
7. <https://www.mongodb.com/docs/>
8. <https://www.w3schools.com/REACT/DEFAULT.ASP>
9. <https://stackoverflow.com/>
10. <https://github.com/>

PLAGIARISM REPORT

AKHILESH 181201

ORIGINALITY REPORT

24% SIMILARITY INDEX	15% INTERNET SOURCES	3% PUBLICATIONS	19% STUDENT PAPERS
--------------------------------	--------------------------------	---------------------------	------------------------------

PRIMARY SOURCES

1	en.wikipedia.org Internet Source	5%
2	Submitted to Indian institute of Management, Udaipur Student Paper	4%
3	Submitted to Campbellsville University Student Paper	3%
4	Submitted to York St John University Student Paper	3%
5	www.javatpoint.com Internet Source	2%
6	Submitted to Glasgow Caledonian University Student Paper	2%
7	Submitted to United Colleges Group - UCG Student Paper	2%
8	Submitted to University of Central Lancashire Student Paper	1%
9	skinpacks.com Internet Source	1%

10	Submitted to Nexford University Student Paper	1 %
11	Submitted to Management Development Institute Of Singapore Student Paper	1 %
12	Submitted to Teaching and Learning with Technology Student Paper	1 %
13	prasannakumar-putha.blogspot.com Internet Source	1 %

Exclude quotes On

Exclude matches < 14 words

Exclude bibliography On