

**INTERSHIP AT ZOHO CORP.**

**VIRTUAL INTERNSHIP REPORT**

*Submitted by*

**RISHI VARMAN R T  
(18C080)**

**BACHELOR OF ENGINEERING**

*in*

**Computer Science and Engineering**

**THIAGARAJAR COLLEGE OF ENGINEERING,  
MADURAI – 625 015**

**18<sup>th</sup> May – 07<sup>th</sup> JULY 2021**



**THIAGARAJAR COLLEGE OF ENGINEERING(TCE),  
MADURAI- 625 015**



### **BONAFIDE CERTIFICATE**

Certified that this Virtual Internship report is the bonafide work of Rishi Varman R T (18C080) 7<sup>th</sup> sem. Department of Computer Science and Engineering who carried out the Virtual Internship at ZOHO Corp. between **8<sup>th</sup> MAY – 07<sup>th</sup> JULY 2021.**

Submitted for Evaluation held at Thiagarajar College of Engineering on .....

**EXAMINER 1**  
(Name with Signature)

**EXAMINER 2**  
(Name with Signature)

**EXAMINER 3**  
(Name with Signature)

## **ABSTRACT**

Internship is a great opportunity to know how the industry works. I got an opportunity to do a summer internship at ZOHO Corp. , Chennai. I worked in ZOHO CREATORS TEAM. This team mainly focused on technologies with JAVA , JDBC , MYSQL , SERVLETS , STRUTS2, OAUTH 2.0. My team leader wanted me to strengthen my knowledge in their technologies so I was asked to study as well as implement what I have studied and was asked to build few applications based on the technologies. This internship was conducted online which had daily evaluation using ZOHO MEET. I was asked to work on visual studio code and eclipse IDE. Daily working hours were from morning 9:00 AM to evening 6:00 PM. I would have to check in for work in ZOHO CLIQ web/mobile app.

My team leader will conduct meet around 5:00 PM or 6:00 PM and will evaluate my progress daily. Along with my team leader , 2 other senior members will be present in the meet and will evaluate my progress. The meet usually takes around 30 to 40 minutes. During the 2 months I was asked to build 2 apps.

1.Employee Management Application

2.Railway Ticket Booking Application

I also implemented a Rest API performing CRUD operations in their application ‘ZOHO CREATOR’.This journey of two months made me explore new things and made me implement what I have learnt for the past years in BE.

### **List Of Abbreviation**

DBMS	Database Management System
OOP	Object Oriented Programming
JDBC	JAVA Database Connectivity
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheet

## **Table of Contents**

<b>Chapter No</b>	<b>Title</b>	<b>Page No</b>
	<b>Abstract</b>	<b>3</b>
	<b>List of Abbreviations</b>	<b>3</b>
<b>1.</b>	<b>Introduction</b>	<b>5</b>
<b>2.</b>	<b>Background</b>	<b>8</b>
<b>3.</b>	<b>Objectives</b>	<b>9</b>
<b>4.</b>	<b>Purpose of the work</b>	<b>9</b>
<b>5.</b>	<b>Problem Formulation</b>	<b>10</b>
<b>6.</b>	<b>Conceptual Design</b>	<b>12</b>
<b>7.</b>	<b>Results and Discussion</b>	<b>15</b>
<b>8.</b>	<b>Conclusion and Future Enhancements</b>	<b>30</b>
	<b>References</b>	

## 1. Introduction

### **JAVA :**

Java is one of the most popular and widely used programming language. Java has been one of the most popular programming language for many years. Java is Object Oriented. However it is not considered as pure object oriented as it provides support for primitive data types (like int, char, etc)The Java codes are first compiled into byte code (machine independent code). Then the byte code runs on Java Virtual Machine (JVM) regardless of the underlying architecture. Java syntax is similar to C/C++. But Java does not provide low level programming functionalities like pointers. Also, Java codes are always written in the form of classes and objects. Java is used in all kind of applications like Mobile Applications (Android is Java based), desktop applications, web applications, client server applications, enterprise applications and many more. When compared with C++, Java codes are generally more maintainable because Java does not allow many things which may lead bad/inefficient programming if used incorrectly. For example, non-primitives are always references in Java. So we cannot pass large objects (like we can do in C++) to functions, we always pass references in Java. One more example, since there are no pointers, bad memory access is also not possible.

## **SERVLETS :**

Servlets are the Java programs that run on the Java-enabled web server or application server. They are used to handle the request obtained from the webserver, process the request, produce the response, then send a response back to the webserver.

The server-side extensions are nothing but the technologies that are used to create dynamic Web pages. Actually, to provide the facility of dynamic Web pages, Web pages need a container or Web server. To meet this requirement, independent Web server providers offer some proprietary solutions in the form of APIs(Application Programming Interface).

These APIs allow us to build programs that can run with a Web server. In this case, Java Servlet is also one of the component APIs of Java Platform Enterprise Edition which sets standards for creating dynamic Web applications in Java.

## **DBMS :**

The software which is used to manage database is called Database Management System (DBMS). For Example, MySQL, Oracle etc. are popular commercial DBMS used in different applications. DBMS allows users the following tasks:

- i. Data Definition: It helps in creation, modification and removal of definitions that define the organization of data in database.
- ii. Data Updation: It helps in insertion, modification and deletion of the actual data in the database.
- iii. Data Retrieval: It helps in retrieval of data from the database which can be used by applications for various purposes.

- iv. User Administration: It helps in registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control and recovering information corrupted by unexpected failure.

## **STRUTS2 :**

Struts is used to create a web applications based on servlet and JSP. Struts depend on the MVC (Model View Controller) framework. Struts application is a genuine web application. Struts are thoroughly useful in building J2EE (Java 2 Platform, Enterprise Edition) applications because struts takes advantage of J2EE design patterns. Struts follows these J2EE design patterns including MVC.

In struts, the composite view manages the layout of its sub-views and can implement a template, making persistent look and feel easier to achieve and customize across the entire application. A composite view is made up by using other reusable sub views such that a small change happens in a sub-view is automatically updated in every composite view.

Struts consists of a set of own custom tag libraries. Struts are based on MVC framework which is pattern oriented and includes JSP custom tag libraries. Struts also supports utility classes.

## **2. Background**

### **Employee Management Application :**

Develop/build a employee management system using struts2 framework. This is a web application which is implemented in eclipse IDE. MySQL is used as the DBMS as well TomCat Server is used as the backend server. This application is implemented using struts2 MVC framework. This application can be used to manage the employees of a company. Operations like adding, deleting and updating the employee details can be done in this application.

### **Railway Ticket Reservation System:**

Develop/build an train ticket booking app using servlets and struts2 framework. There are two pages, admin page and reservation page. The admin page can manage all the trains and the tickets booked in each train. Also he can set the message to be sent to each user who reserves the ticket based on the points. The reservation fetches the details from the xml file which contains the details of the train, like its seating capacity and the travel source and destinations. The user can reserve and cancel their tickets in the reservation page. Also there is a waiting list column where the tickets which are pending to be reserved are in the waiting list table. It automatically gets confirmed if any tickets are cancelled.

### **REST API :**

Design a REST API and integrate with Zoho Creator application which is a cloud software to create custom applications on your own without any prior coding experience or IT expertise. Implement the api which performs the CRUD operations using POSTMAN.



### 3. Objectives

#### **Employee Management Application:**

- Use Struts2 framework
- Use ajax to get the data from backend
- To make an user friendly interface
- Connect Database with frontend

#### **Railway Ticket Reservation System :**

- Use Struts2 framework
- Use java collections
- Use ajax to get the data from backend
- To make an user friendly interface
- Connect Database with frontend
- Handle XML files using JAVA

#### **Rest Api:**

- Perform CRUD operations using the API
- To Use POSTMAN
- To integrate an API with the application
- Integrate the OAuth 2.0 authentication with the API

### 4. Purpose of Work

#### **Employee Management Application:**

- To test the knowledge and understanding of OOPS concepts
- To use and familiarize java collections

- To explore complex system designs and design them accordingly
- To Use ajax to get the data from backend

### **Railway Ticket Reservation System:**

- To build web pages using html css
- To use jsp and scriptlets
- To use mysql and connect using jdbc
- To use struts2 framework
- To Handle XML files using JAVA
- To use and familiarize java collections

### **Rest Api:**

- To Perform CRUD operations using the API
- To Use POSTMAN
- To integrate an API with the application
- To Integrate the OAuth 2.0 authentication with the API

## **5. Problem Formulation**

### **1.Employee Management Application**

- There is a admin page
- Admin page consists of four buttons.
  - Add Employee
  - Remove Employee
  - Update Employee
  - View Employees
- Admin will be able to add employees in the database
- Admin will be able to update the details of the employees in the database
- Admin will be able to delete the employees in the database
- Admin will be able to view all the employee details from the database which will be displayed as a table

### **2.Railway Ticket Reservation System**

- User will be able to book tickets for the train which is required for him
- Points have to be given to the user while he makes his booking
- Users must be able to cancel the booked tickets
- The Reservation page must show the train list and the tickets available in the train.
- If the tickets have been completely booked then the tickets have to be put in the waiting list
- When the confirmed tickets are cancelled then the waiting list tickets are confirmed

- The details of the train are fetched from the xml file
- The admin can view all the passenger details
- The admin can set the message which can be displayed to the user while he books the tickets

### **3.Rest API**

- Must be able to perform the CRUD operations
- Integrate with the Zoho Creator Application
- Use POSTMAN to implement the API
- Integrate OAuth 2.0 for authentication

## **6. Conceptional Design**

### **1.Employee Management Application**

- This is an servlet web app based on MVC concept.
- HTML , CSS , JS , Scriptlets were used for front end
- Mysql server was used for back end
- Struts2 Framework was used for connecting front end and backend.
- Jdbc is used to connect front end and backend
- 2 Packages were created one for all actions classes and other for all class files.
- All operations that were processed were implemented in action class
- Ajax is used to get the data from the backend

### **2. Railway Ticket Reservation System**

- This is an servlet web app based on MVC concept.
- HTML , CSS , JS , Scriptlets were used for front end
- Mysql server was used for back end
- Struts2 Framework was used for connecting front end and backend.
- Jdbc is used to connect front end and backend
- 2 Packages were created one for all actions classes and other for all class files.
- All operations that were processed were implemented in action class
- Ajax is used to get the data from the backend
- All are stored as objects while moving them through the backend

EG:

A class Train is created with 8 member variables and Getter&Setter

were created for the member variables.

1. private int Uid;
2. private int total;
3. private String Source;
4. private String Destination;
5. private int price;
6. private String Status;
7. private String phone;
8. private String trainid;

- The train details are stored in xml file in the format

```
<train name="Madurai Express" value="tr1">
```

```
    <station name="Madurai">
```

```
        <id>101</id>
```

```
        <seats>70</seats>
```

```
        <points>15</points>
```

```
    </station>
```

## 7. Results and Discussions

### a) Employee Management Application:

#### Structure of struts. Xml file:

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts
3 Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">
4
5 <struts>
6   <constant name="struts.devMode" value="true" />
7   <package name="default" extends="struts-default" namespace="/">
8
9     <action name="Register" class="com.struts.action.AddAction">
10
11   </action>
12
13   <action name="Delete" class="com.struts.action.DeleteAction">
14
15   </action>
16
17   <action name="Edit" class="com.struts.action.EditAction">
18
19   </action>
20
21   <action name="viewrecords" class="com.struts.action.ViewAction">
22
23   </action>
24 </package>
25 </struts>

```

localhost:8080/ajax/

STR's Enterprise

Add User

Name

Email Address

Age

Salary

Add

Eid	Ename	Age	Email	Salary	Edit	Delete
	Rishvarman R T	20	risvarrt@gmail.com	200000	<a href="#">Edit</a>	<a href="#">Delete</a>

localhost:8080 says

Resgistered

OK

STR's Enterprise

Add User

Eid	Ename	Age	Email	Salary	Edit	Delete
1	Rishivarman R T	20	rishvarrt@gmail.com	200000	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	Rathi Mala	45	rat@gmail.com	122222	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

STR's Enterprise

localhost:8080 says

Edited...!

OK



localhost:8080 says  
Deleted risvarrt@gmail.com

OK

STR's Enterprise

Add User

Rathi Mala

rat@gmail.com

45 122222

Add

Eid	Ename	Age	Email	Salary	Edit	Delete
1	Rathi Mala	45	rat@gmail.com	122222	Edit	Delete

## b) Railway Ticket Reservation System:

### Structure of struts. Xml file:

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE struts PUBLIC "-//Apache Software Foundation/DTD Struts
3 Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">
4
5 <struts>
6 <constant name="struts.devMode" value="true" />
7 <package name="default" extends="struts-default" namespace="/">
8
9 <action name="Buy" class="com.struts.action.BuyAction">
10
11 </action>
12 <action name="viewc" class="com.struts.action.ViewAction">
13
14 </action>
15 <action name="viewp" class="com.struts.action.ViewAction1">
16
17 </action>
18 <action name="viewavail" class="com.struts.action.AvailabilityAction">
19 </action>
20
21 <action name="stlist" class="com.struts.action.StationAction">
22
23 </action>
24 <action name="Cancelc" class="com.struts.action.CancelcAction">
25
26 </action>
27 <action name="Cancelp" class="com.struts.action.CancelpAction">
28
29 </action>
30 <action name="gettrain" class="com.struts.action.GetTrain">
31
32 </action>
33 </package>
34
35
36 <package name="admin" namespace="/" extends="struts-default">
37
38
39 </package>
40
41 </struts>

```

## XML file of train:

```

1 <?xml version="1.0"?>
2 <trains>
3   <train name="Madurai Express" value="tr1">
4     <station name="Madurai">
5       <id>101</id>
6       <seats>70</seats>
7       <points>15</points>
8     </station>
9     <station name="Dindugal">
10      <id>102</id>
11      <seats>40</seats>
12      <points>25</points>
13    </station>
14    <station name="Trichy">
15      <id>103</id>
16      <seats>40</seats>
17      <points>25</points>
18    </station>
19    <station name="Coimbatore">
20      <id>104</id>
21      <seats>40</seats>
22      <points>25</points>
23    </station>
24    <station name="Chennai">
25      <id>105</id>
26      <seats>40</seats>
27      <points>25</points>
28    </station>
29  </train>
30
31 </trains>
32 <train name="Pandian Express" value="tr2">
61 <train name="Chennai Express" value="tr3">
90 </trains>

```



## Available trains

Madurai Express
Pandian Express
Chennai Express

← → ↻ http://localhost:8080/railways2/ 🔍 ☆ ⚙️ 👤

### STR's Public Transport Reservation System

#### Ticket Availability

Station- id	Station	Tickets Available
101	Madurai	70
102	Dindugal	40
103	Trichy	40
104	Coimbatore	40
105	Chennai	40

#### Passenger List

MADURAI

DINDUGAL

TRICHY

COIMBATORE

CHENNAI

#### Buy Ticket

Phone Number

Total Number of Tickets

Source ▾ Destination ▾

Buy

← → ↻ http://localhost:8080/railways2/ 🔍 ☆ ⚙️ 👤

#### Buy Ticket

9842110473

5

Salem ▾ Thoothukudi ▾

Buy

localhost:8080 says  
Pandian Express--Tickets Purchased:5--Points:250--CALLING--Ticket  
Purchased--Calling tr2--Mailing--Ticket Purchased--Mailing tr2--  
OK

localhost:8080 says

STATUS:PENDING

OK

Canceled ticket ticket ID :324

OK

[illegible]

STR's ADMIN PANEL

Buy Action

Points

Cancel

STR's ADMIN PANEL

Buy Action

Enter Points Range

Train-ID

FROM

TO

Next

Buy Action Table

Action-ID	Train-ID	Points Range	E-Title	E-Message	M-Title	M-Message	P-Title	P-Message	Edit	Delete
28	tr2	100-400	Ticket Purchased	Mailing tr2	NIL	NIL	Ticket Purchased	Calling tr2	Edit	Delete
33	tr1	300-500	Ticket Purchased	mailing	Ticket Purchased	mesaging	NIL	NIL	Edit	Delete
34	tr1	200-400	NIL	NIL	Ticket Purchased	MESSAGING	Ticket Purchased	CALING	Edit	Delete

STR's ADMIN PANEL

Buy Action

Enter Points Range

☐

Email

☒

Call

☒

Message

Prev

Next

localhost:8080 says

Admin Message has been set

OK

## Enter Points Range

### Message

Title

Ticket Purchased

Message

Messaging for Ticket Confirmation

### Call

Title

Ticket Purchased

Message

Callin for confirmation

Prev

Submit

## Database of the confirmed tickets:

Server: 127.0.0.1 » Database: struts » Table: trainc

Browse

Structure

SQL

Search

Insert

Export

Import

Privileges

Operations

Tracking

Triggers

Showing rows 0 - 13 (14 total, Query took 0.0006 seconds.)

SELECT \* FROM `trainc`

☐ Profiling [\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Explain SQL \]](#) [\[ Create PHP code \]](#) [\[ Refresh \]](#)

☐ Show all

Number of rows: 25

Filter rows: Search this table

Sort by key: None

+ Options

Edit

Copy

Delete

312

tr1

8870056293

12

102

104

360

CONFIRMED

600

Edit

Copy

Delete

313

tr1

8870056293

10

101

102

300

CONFIRMED

150

Edit

Copy

Delete

314

tr1

8870056293

12

101

102

360

CONFIRMED

180

Edit

Copy

Delete

315

tr1

8870056293

10

101

102

300

CONFIRMED

150

Edit

Copy

Delete

316

tr2

9842160473

10

201

202

300

CONFIRMED

250

Edit

Copy

Delete

317

tr1

8870056293

12

101

102

360

CONFIRMED

180

Edit

Copy

Delete

318

tr1

8870056293

12

101

102

360

CONFIRMED

180

Edit

Copy

Delete

319

tr2

8870056293

12

201

202

360

CONFIRMED

300

Edit

Copy

Delete

320

tr1

8870056293

12

101

102

360

CONFIRMED

180

Edit

Copy

Delete

321

tr1

8870056293

12

101

104

360

CONFIRMED

780

Edit

Copy

Delete

322

tr1

8870056293

8

101

103

240

CONFIRMED

320

Edit

Copy

Delete

323

tr1

8870056293

12

101

102

360

CONFIRMED

180

Edit

Copy

Delete

325

tr2

9842110473

11

201

202

330

CONFIRMED

275

Edit

Copy

Delete

326

tr2

9842110473

5

203

205

150

CONFIRMED

250

Check all

With selected:

Edit

Copy

Delete

Export

☐ Show all

Number of rows: 25

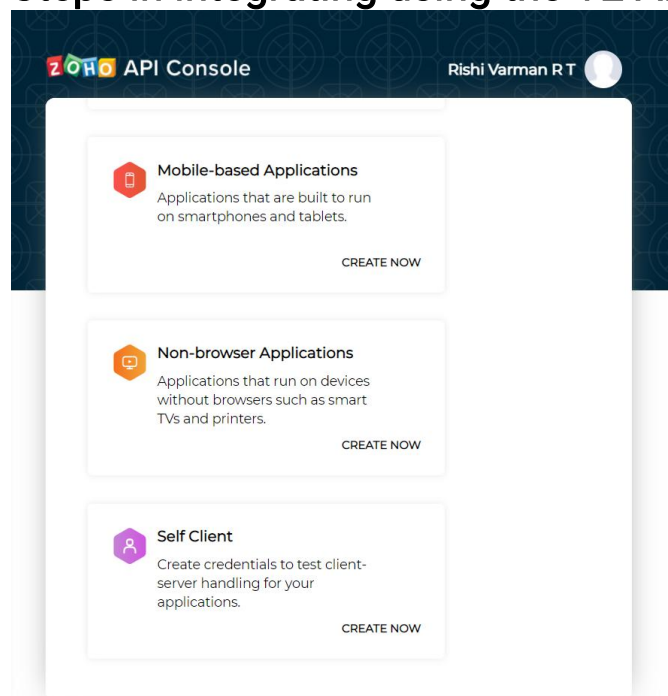
Filter rows: Search this table

Sort by key: None

Console

### c) Rest API:

## Steps in integrating using the V2 API:



Generate Access and refresh token

Examples 0 BUILD

POST [https://accounts.zoho.com/oauth/v2/token?grant\\_type=authorization\\_code&client\\_id=1000.1OAGGIXXU44G5...](https://accounts.zoho.com/oauth/v2/token?grant_type=authorization_code&client_id=1000.1OAGGIXXU44G5...) Send Save

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies Code

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	grant_type	authorization_code			
<input checked="" type="checkbox"/>	client_id	1000.1OAGGIXXU44G9S28HT7A1VLU9ROW0C			
<input checked="" type="checkbox"/>	client_secret	f2b334108382d8d2048d076d81dab624e8e378aaf4			
<input type="checkbox"/>	redirect_uri	<redirect_uri>			
<input checked="" type="checkbox"/>	code	1000.be509c696ffc6c8e07611de574bb68a0.04048...			

Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "access_token": "1000.94f6df256ba912d7bfe459d989648ceb.e1b64ae7c2cb791eb5a216972a26023a",
3   "refresh_token": "1000.ad8ca293431284a5dad06766ca436337.ea4f06d158daba91df3bf033afb215d6",
4   "api_domain": "https://www.zohoapis.com",
5   "token_type": "Bearer",
6   "expires_in": 3600
7 }
```



POST <https://creatorapp.zoho.com/api/v2/risvarrt/visitor-management/form/Contact> Send Save

Params Authorization Headers (11) Body Pre-request Script Tests Settings Cookies Code

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1 {
2   "data": {
3     "Email": "risvarrt@gmail.com",
4     "Phone_Number": "+918870056293",
5     "Name": {
6       "first_name": "Rishi",
7       "last_name": "varman"
8     },
9     "Date_of_Birth": "10-Jan-2020"
10  }
11 }

```

Body Cookies (3) Headers (15) Test Results Status: 200 OK Time: 1217 ms Size: 628 B Save Response

Pretty Raw Preview Visualize JSON 🔍

```

1 {
2   "code": 3000,
3   "data": {
4     "ID": "4192570000000020005"
5   },
6   "message": "Data Added Successfully"
7 }

```

## Visitor Management

Rishi Varman R T



EMPLOYEES



VISITORS



VISIT REQUESTS FOR WEB



CONTACTS

### All Contacts



	<input type="checkbox"/>	Name	Date of Birth	Email	Phone
		Rathi Mala	12-Dec-1977	rat@gmail.com	+919942340216
		Rishi varman	10-Jan-2020	risvarrt@gmail.com	+918870056293

## **8. Conclusions and Future Enhancements:**

### **a) Employee Management Application**

- This Web app is hosted on tomcat server and runs on localhostin web browsers.
- The admin can add,delete and update the employees
- The admin can view all the employees
- All the data are stored in tables using mysql database.

#### **Future Enhancements:**

- Sign in feature can be added to the employees who are added by the admin
- Multiple admin feature can be added

### **b) Railway Ticket Reservation System:**

- User will be able to book tickets for the train which is required for him
- Points are given to the user while he makes his booking
- Users will be able to cancel the booked tickets
- The Reservation page must show the train list and the tickets available in the train.
- If the tickets have been completely booked then the tickets have to be put in the waiting list
- When the confirmed tickets are cancelled then the waiting list tickets are confirmed
- The details of the train are fetched from the xml file
- The admin can view all the passenger details

- The admin can set the message which can be displayed to the user while he books the tickets
- All the data are stored in tables using mysql database.

**Future Enhancements:**

- Specific seat booking feature can be added where the user can book which seat he wants in the train
- Mailing or messaging feature can be added so that the user the message of his ticket confirmation

**c) Rest API:**

- It is able to perform the CRUD operations
- It is Integrated with the Zoho Creator Application
- POSTMAN to used implement the API
- Integrated with OAuth 2.0 for authentication

**Future Enhancements:**

- A simple front end can be used to perform the CRUD operations with a UI

## 9. References:

- Java - What, Where and Why? - <https://www.javatpoint.com/java-tutorial>
- OOPs - <https://www.javatpoint.com/java-oops-concepts>
- Object class - <https://www.javatpoint.com/object-class>
- Java Array - <https://www.javatpoint.com/array-in-java>
- Wrapper Class - <https://www.javatpoint.com/wrapper-class-in-java>
- Collection - <https://www.javatpoint.com/collections-in-java>
- Multithreading - <https://www.javatpoint.com/multithreading-in-java>
- SQL - <https://www.w3schools.com/sql/>
- JDBC - <https://www.javatpoint.com/java-jdbc>
- STRUT2 - <https://www.javatpoint.com/struts-2-tutorial>
- HTML - <https://www.w3schools.com/html/>
- CSS - <https://www.w3schools.com/css/>

Below is my internship completion certificate.

**Zoho Corporation Private Limited**

Plot 140, 151, Estancia IT Park, Vallancheri,  
Chengalpattu District, Tamilnadu, 603 202.

Ph: +91 - 44 - 6744 7070

[www.zohocorp.com](http://www.zohocorp.com)

SEZ Unit

Dated: 03-Nov-2021

**TO WHOMSOEVER IT MAY CONCERN**

This is to certify that Mr/Ms. **Rishi Varman R T, SI-732;** has undergone his/her internship training in **Zoho Corporation Private Limited**, from 18-May-2021 to 07-Jul-2021. During this period, his/her performance and conduct were found to be good.

**Saajudeen S**

**Associate HR**



Corporate Identification No: **U40100TN2010PTC075961**

e-mail ID: [hr-team@zohocorp.com](mailto:hr-team@zohocorp.com)