INDUSTRY INTERNSHIP PROJECT

CAPGEMINI

Training Report

Submitted in partial fulfilment of requirements for the award if degree of

B.Tech

Computer Science and Engineering

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB

From 01/20/20 to 05/08/20

SUBMITTED BY

Name of student: Gurubelli Vikram

Registration Number: 11613655

Signature of the student:Vikram

Submitted to: Sawal Tandon

Name of Supervisor: Sawal Tandon

Designation: Assistant Professor Head

Declaration by the Supervisors

To whom so ever it may concern

This is to certify that **Gurubelli Vikram, 11613655** from Lovely Professional University, Phagwara, Punjab, has worked as a trainee in CAPGEMINI on **"JEE-FS TRAINEE"** under my supervision from **Jan, 2020 to May, 2020**. It is further stated that the work carried out by the student is a record of original work to the best of my knowledge for the partial fulfilment of the requirements for the award of the degree, degree name.

Name of External Supervisor	Name of Internal Supervisor
Varsha Lonkar	**Sawal Tandon**
Designation of the External Supervisor	Designation of the Internal Supervisor
Consultant-India core L&D-JEE	**Assistant Professor Head**
Signature of the external Supervisor	Signature of the Internal Supervisor
Dated:16/06/2020	Dated:16/06/2020

Student Declaration

To whom so ever it may concern

I, **Gurubelli Vikram, 11613655,** hereby declare that the work done by me on "**JEE-FS TRAINING**" from **Jan, 2020** to **May, 2020**, under the supervision of **Varsha Lonkar, Consultant-India Core L&D-JEE**, and **Sawal Tandon, Asst. Professor and Head (CSE)**, Lovely professional University, Phagwara, Punjab, is a record of original work for the partial fulfilment of the requirements for the award of the degree, **B.Tech (Computer Science and Engineering)**.

Gurubelli Vikram(11613655)

Signature of the student

Dated: 16/06/2020

TRAINING CERTIFICATE FROM ORGANIZATION

ACKNOWLEDGMENT

The training opportunity I had with Capgemini was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals who led me through this internship period.

Bearing in mind previous I am using this opportunity to express my deepest gratitude and special thanks to the **JEE-FS TRAINER, Mr. BABU** who guides and keep me on the correct path and allowing me to carry out my project at their esteemed organization and extending during the training.

I express my deepest thanks to **Varsha Lonkar,Consultant-India Core L&D-JEE** for taking part in useful decision & giving necessary advices and guidance and arranged all facilities to make life easier. I choose this moment to acknowledge her contribution gratefully.

I perceive as this opportunity as a big milestone in my career development. I will strive to use gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives. Hope to continue cooperation with all of you in the future,

Sincerely,

Name: Gurubelli Vikram

Regd. Number: 11613655

Date: 16/06/2020

LIST OF CONTENTS

Chapter 1

1.1 Introduction of Capgemini

Capgemini is a global leader in consulting, digital transformation, technology and engineering services. The Group is at the forefront of innovation to address the entire breadth of clients' opportunities in the evolving world of cloud, digital and platforms. Building on its strong 50- year+ heritage and deep industry-specific expertise, Capgemini enables organizations to realize their business ambitions through an array of services from strategy to operations. Capgemini is driven by the conviction that the business value of technology comes from and through people. Today, it is a multicultural company of 270,000 team members in almost 50 countries. With Altran, the Group reported 2019 combined revenues of €17billion.

1.2 VISION AND MISSION

At Capgemini, we live and breathe the philosophy that "people matter and results count".

Our Vision: The business value of technology comes from and through people.

Capgemini understands that business value cannot be achieved through technology alone. It starts with people: experts working together to get to the heart of your individual business objectives and develop the most adapted solutions to fit these requirements. We believe this human-centered approach to technology is what makes the difference for your business.

Our Mission: with you, we create and deliver business and technology solutions that fit your needs and drive the results you want, Capgemini enables you to transform your organization and improve performance.

We aim to empower you to respond more quickly and intuitively to changing market dynamics. By bolstering your ability to harness the right technology, we help you become more agile and competitive.

Collaboration is central to the way we do business. Our experts join forces with your people to form a cohesive team. More than just a promise, our capacity to collaborate has become a key client expectation. We call this approach the Collaborative Business Experience®. It shows in our every interaction and is our way of forging closer, more effective relationships.

People matter, results count.

1.3 HISTORY

Capgemini was founded by Serge Kampf in 1967 as an enterprise management and dataprocessing company. The company was inaugurated as the Société pour la Gestion de l'Entreprise et le Traitement de l'Information (Sogeti).

In 1974 Sogeti acquired Gemini Computers Systems, a US company based in New York.In 1975, having made two major acquisitions of CAP (Center d'Analyze et de Programmation) and Gemini Computer Systems, and following resolution of a dispute with the similarly-named CAP UK over the international use of the name 'CAP', Sogeti renamed itself as CAP Gemini Sogeti.

Cap Gemini Sogeti launched US operations in 1981, following the acquisition of Milwaukeebased DASD Corporation, specializing in data conversion and employing 500 people in 20 branches throughout the US. Following this acquisition, The U.S. Operation was known as Cap Gemini DASD.

In 1996, the name was simplified to Cap Gemini with a new group logo. All operating companies worldwide were re-branded to operate as Cap Gemini.

Ernst & Young Consulting was acquired by Cap Gemini in 2000. It simultaneously integrated Gemini Consulting to form Cap Gemini Ernst & Young.

In 2017, Cap Gemini S.A. became Capgemini SE, and its Euronext ticker name similarly changed from CAP GEMINI to CAPGEMINI.

In 2019, Capgemini acquires Altran bringing the total employee count to over 250,000. This is the largest acquisition in the companies history.

1.4 CODE OF BUSINESS ETHICS

Our Code of Business Ethics is our ongoing commitment to maintaining and promoting worldclass standards of business integrity and trust, wherever we operate.

Since our Group was founded, we have always believed that a profitable and sustainable business cannot exist without sound ethics and integrity. This belief lies at the very heart of our decentralized organization. Our Code of Business Ethics encapsulates our respect for the law and for individuals, and our responsibilities to our clients and stakeholders. It provides the basis for our ethical culture.

The Group has always been a values-based organization: our Group Values are a part of our DNA. Guided by our Values, we seek to adopt an ethical mindset and ethical behavior in all our daily business activities. Our Code of Business Ethics sets out, explains, and formalizes our Values, principles of action, and rules of conduct and behavior regarding:

- **Our People:** We are committed to providing a safe, inclusive work environment.
- **Business Integrity:** As a business we are committed to acting responsibly in the market place.
- **Business Relationships:** We are committed to delivering value and building longstanding relationships based on mutual trust with our clients, and to achieving profitable and sustainable growth for the Group.
- **Group and Third-Party Assets:** We recognize our responsibility to maintain the security and integrity of the assets of our Group and of the third parties we work with, by handling these assets responsibly and professionally.

- **Corporate Social Responsibility:** As responsible citizens, we support the communities and respect the environment in which we operate.

We follow our Code of Business Ethics no matter where we operate and whatever the economic circumstances. Following the Code of Business Ethics is mandatory for Group compliance.

1.5 COMPANY’S BUSINESS

Overview of our Consulting, Technology, Outsourcing and other Managed Services.

Consulting Services: (Capgemini Consulting), which help to enhance the performance of organizations, based on in-depth knowledge of client industries and processes;

Application Services: which devise, develop, implement and maintain IT applications covering the Group’s system integration and application maintenance activities;

Technology and Engineering Services (Sogeti): which provide assistance and support to internal IT teams within client companies;

Other Managed Services: which integrate, manage and/ or develop either fully or partially, clients’ IT Infrastructure systems (or that of a group of clients), transaction services and on demand services and/or business activities (Business Services).

The speed of change is throwing traditional business methods into question and disrupting the relevance of entire industries. With over 270,000 team members in almost 50 countries countries and partnerships with a broad spectrum of technology vendors, Capgemini has both the resources and expertise to react quickly to client needs — a key asset in a digital era when deployment needs to be measured in weeks rather than months.

1.6 CORPORATE GOVERNANCE

The Board of Directors

The Board of Directors sets the strategic direction of the Company and the Capgemini Group. It appoints the executive corporate officer(s) responsible for implementing this strategy, approves the financial statements, convenes the Shareholders’ Meeting, and proposes the annual dividend. It takes decisions on the major issues concerning the day-to-day operation and future of Capgemini, to promote sustainable value creation for its shareholders and all stakeholders.

1.7 APPROACH

Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

We offer an array of integrated services that combine top-of-the-range technology with deep sector expertise and a strong command of our four key businesses.

The Collaborative Business Experience™

People Matter, Results Count.

Technology is advancing, borders are disappearing and competition is intensifying. More and more, successful businesses must be ready to respond to complex and unpredictable challenges. With Capgemini's team of experts at your side, your business will become more agile and competitive.

The best way for us to work for you is to work with you. The Collaborative Business Experience™ helps you define the rules of the game, rather than be limited by them.

Finding Unique Solutions Together

Collaboration is central to the Capgemini philosophy and a pillar of our service delivery. From strategy development through to implementation, clients benefit from our tailored approach.

Working beside you every step of the way, we analyze your challenges and guide you through your transformation.

Our collaborative tools and methodologies give you the freedom to transform and overcome resistance to change. We channel the expertise of our leading technology partners, putting the right tools in the hands of your teams.

Guided by the 4 dimensions of collaboration, we target value, mitigate risk, optimize capabilities, and align the organization to achieve the objective.

Collaboration in Action

Capgemini is the partner of choice for leading businesses across all sectors. Our client Success Stories show how the Collaborative Business Experience™ helps organizations discover the freedom to increase performance through innovation.

Through the experience that comes from working with thousands of companies over the past three decades, Capgemini has identified four key elements of collaboration. They form a picture of an organization's strengths, weaknesses, needs, and priorities. They help us collaborate with our clients to achieve better, faster, and more sustainable results.

1.8 Rightshore®

In today’s challenging economic environment, businesses are seeking to reduce costs and enhance growth. They need business and IT solutions that promote innovation and transformation, while helping to create and sustain a competitive advantage Businesses can achieve these goals with a scalable approach to global delivery and sourcing that combines quality, efficiency, talent and collaboration.

Rightshore® Benefits

We draw on our global resources, cost-efficient processes and extensive experience in diverse geographies, disciplines and industries to deliver:

- **Cost reductions:** Save up to 40% on IT costs and improve your bottom line by leveraging the right balance of locations.
- **Streamlined processes:** Improve productivity and reduce operating costs by implementing agile, efficient processes.
- **Innovation:** Free your energies from IT and focus on innovating and transforming your business.
- **Competitive advantage:** Stay ahead of the curve with solutions that employ the latest technology to improve quality in delivery.
- **Growth:** Implement solutions that increase productivity and help you expand your business.

1.9 Accelerated Solutions Environment(ASE)

What is our Accelerated Solutions Environment and why do more than a third of FTSE 100 and Fortune 500 companies use it to solve complex business issues?

Solving Complex Business Challenges

Solving problems is a part of doing business. But bringing together key stakeholders from multiple business areas and geographies to find solutions can compound the problem and take months, or even years. Moreover, organizational and cultural conflicts can make it difficult for employees to commit to and manage change quickly.

Organizations have to engage with a growing and changing list of client groups and third parties. They need an innovative way to unite their stakeholders to collaborate on sustainable solutions that meet with consensus within hours and days.

Collaboration to Drive Project Acceleration

The ASE brings together the right people, with the right content, underpinned by a rigorous method and process. This collaborative way of working offers a different look and feel to a traditional change program. As you start to infuse greater collaboration into the day-to-day, you will find improvements in your work culture and ways of working between functional areas.

Working with the ASE facilitation team and large-scale change model, we will propel your organization from solution design to implementation.

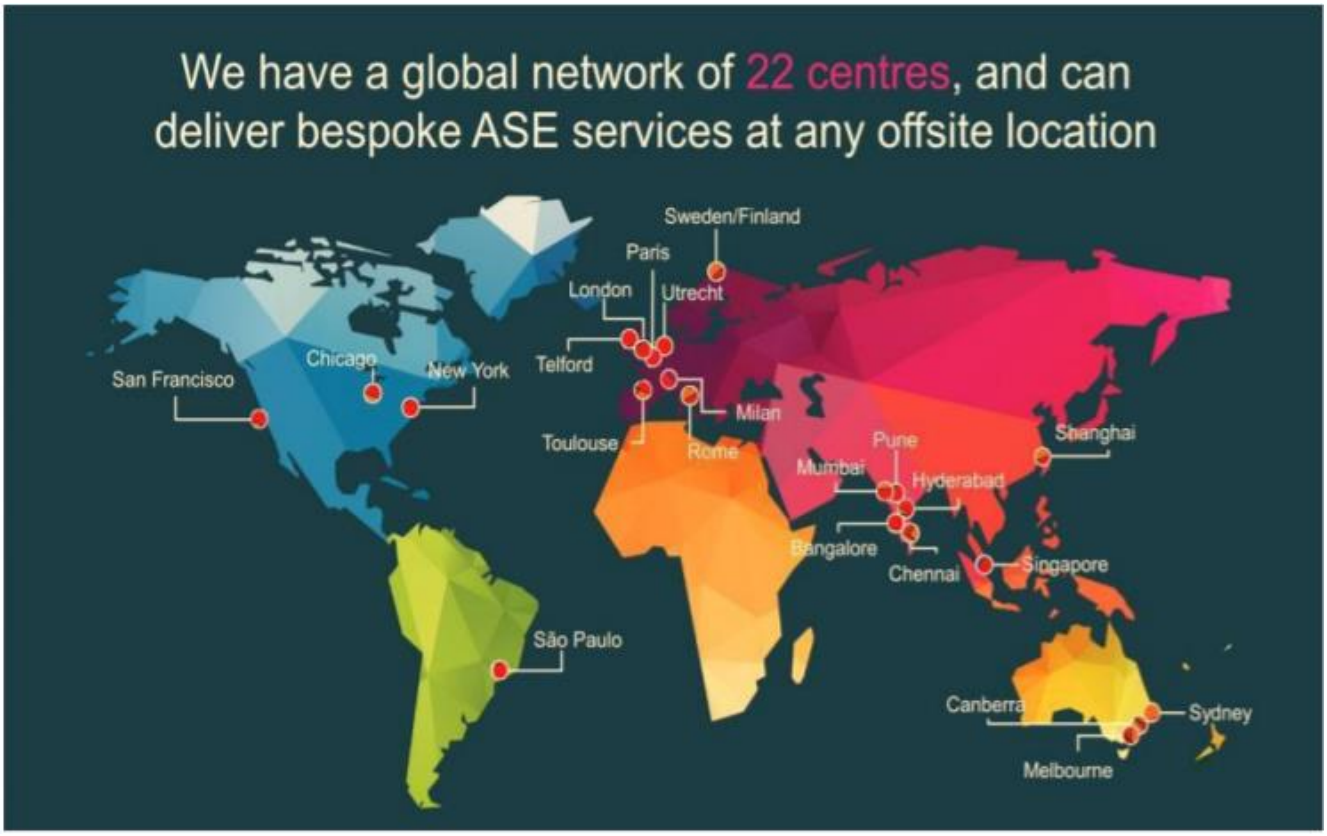
We organize events of one to three or more days called ASE DesignSessions. In many cases, our dedicated teams will scope, design and facilitate multiple ASE DesignSessions over a project lifecycle to help you and your key stakeholders to align and implement change successfully.

We also offer an “Acceleration Zone”, a custom built collaborative workspace that is established on-site at your location.

We deliver the following project outcomes:

- **Acceleration:** make business decisions in hours and days.
- **Quality:** collaborate effectively to develop robust, fit-for-purpose solutions.
- **Ownership:** involve and align multiple stakeholders to enable their commitment to change
- **Mitigation:** use proven tools and methodologies to improve the certainty of program delivery
- **Innovation:** use our Co-Innovation Labs to deliver Rapid Ideation Sessions, Hackathon Sessions, Crowd Sourcing and Rapid Prototyping.

1.10 Worldwide Presence



We have a global network of 22 centres, and can deliver bespoke ASE services at any offsite location

1.11 Capgemini Awards and Recognition

2019

- Capgemini positioned as a leader among Global SAP HANA and Leonardo Ecosystem Partners for SAP S/4HANA – September 2019.
- 2019 SAP Pinnacle Award – FINALIST in the Customer Experience Partner of the Year – Large Enterprise category.
- Capgemini named a Leader in two IDC MarketScape 2019 Vendor Assessments for Utility Services.
- Capgemini named Leader in Next-generation IT Infrastructure Services by Everest.
- Capgemini named a Leader by Everest Group for BFS.
- Capgemini named a Leader in Advanced Analytics BPS by NelsonHall.
- Capgemini wins 2019 Artificial Intelligence Breakthrough award for best virtual agent solution.
- Capgemini named EMEA Systems Integrator (SI) & Value Added Reseller (VAR).

2018

- Capgemini named a 2018 World's Most Ethical Company® by the Ethisphere Institute for the 6th consecutive year.
- Capgemini Research Institute ranked Number 1 for the quality of its research.
- Capgemini recognized as a Leader in Gartner's Magic Quadrant for CRM and Customer Experience Implementation Services, Worldwide.
- Capgemini named a Leader for Digital Services in Property and Casualty (P&C) Insurance Capabilities by Everest Group.
- Capgemini named a Leader in Digital Services in Consumer Banking in inaugural Everest Group PEAK MatrixTM.
- Capgemini named a Leader in Risk and Compliance Applications Services by Everest Group.
- Capgemini's innovative learning programs are recognized with six awards for excellence from the Brandon Hall Group.
- Source Global Research ranks Capgemini's Digital Transformation Institute in the top 3 for the quality of its research.
- Capgemini named a Leader in Gartner's Magic Quadrant for Data and Analytics Service Providers, Worldwide 2018.

2017

- Capgemini receives Citrix System Integrator Award 2017 for its My Workspace Services.
- Capgemini named a 2017 World's Most Ethical Company® by the Ethisphere Institute for the 5th consecutive year.
- Capgemini receives Partner Award Winners at CONNECT 2017.
- Capgemini Group onoured by IBM with 2017 Outstanding Partnership Award for Security.
- Capgemini University wins 2017 Skillsoft Innovation award for its Engagement Managers Transformation program.
- Capgemini Receives 2017 SAP® Pinnacle Award: Customers' Choice Partner of the Year.
- Capgemini wins the 2017 Pega Partner Award for Partner Excellence in Accelerating Growth.
- Capgemini recognized as Informatica's 2017 Big Data Partner of the Year.
- Capgemini Group named a Leader in Gartner's Magic Quadrant for Application Testing Services, Worldwide 2017.

2016

- Capgemini University is recipient of the prestigious Brandon Hall Group Excellence Awards 2015.
- The Ethisphere Institute: Capgemini named a 2016 World's Most Ethical Company® for the 4th consecutive year.
- Capgemini receives SAP® North America Partner Excellence Award 2016 for Industry Services.
- Capgemini receives 2016 SAP® Pinnacle Award: Customers' Choice – Service.
- Capgemini positioned in the Winner's Circle by HfS Research Blueprint report for Finance & Accounting As-a-Service Business Process Outsourcing.
- Capgemini wins Pegasystems Partner Award 2016 for Partner Excellence in Driving Growth.
- Capgemini University receives Skillsoft Innovation Award 2016 for Empowering Continuous Talent Development.

2015

- Capgemini featured in the "Winner's Circle" for Enterprise Mobility Services in the HfS Blueprint Report.
- SAP: Capgemini Receives 2015 SAP® Pinnacle Award: Services Transformation Partner of the Year.
- Capgemini Consulting is Awarded "Best Performing Brand" by the Managing Partners' Forum for Management Excellence 2015.
- HfS: Capgemini positioned in the Winner's Circle by HfS Research Blueprint report for Finance and Accounting BPO.

- Capgemini positioned as a ‘Leader’ amongst Large Enterprise Mobile Applications Service Providers by Independent Research Firm.
- Capgemini Named as a 2015 World’s Most Ethical Company® by the Ethisphere Institute for the 3rd Consecutive Year.
- Capgemini Receives 2014 Cisco Supplier Quality Award for Delivery Excellence.
- Capgemini wins two prestigious awards at the Informatica Partner Summit for outstanding performance.

2014

- The Ethisphere Institute: World’s Most Ethical Company Award Again!.
- Forrester: Forrester Ranks Us as Leader in Latest SAP Services Report.
- Forrester: Leader Forrester WaveTM-Bericht SAP Services.
- Gartner: Leader in Global F&A BPO Magic Quadrant from Gartner.
- Nelson Hall: Capgemini positioned as a Leader in Software Testing Services by NelsonHall.
- Gartners: Capgemini Group positioned in the “Leaders” Quadrant in Gartner’s Magic Quadrant for Application Testing Services 2014.
- Forrester: Capgemini named a ‘Leader’ amongst Global Infrastructure Outsourcing Providers in an independent report by Forrester Research, Inc.
- Gartners: Capgemini positioned in “Leaders” Quadrant in Gartner’s Magic Quadrant for SAP Application Management Service Providers, Worldwide.
- Gartners: Capgemini positioned in “Leaders” Quadrant in Gartner’s Magic Quadrant for SAP Implementation Service Providers, Worldwide.
- Source: Capgemini Consulting has been recognized by Source1 for the consistently high quality of its thought leadership content.

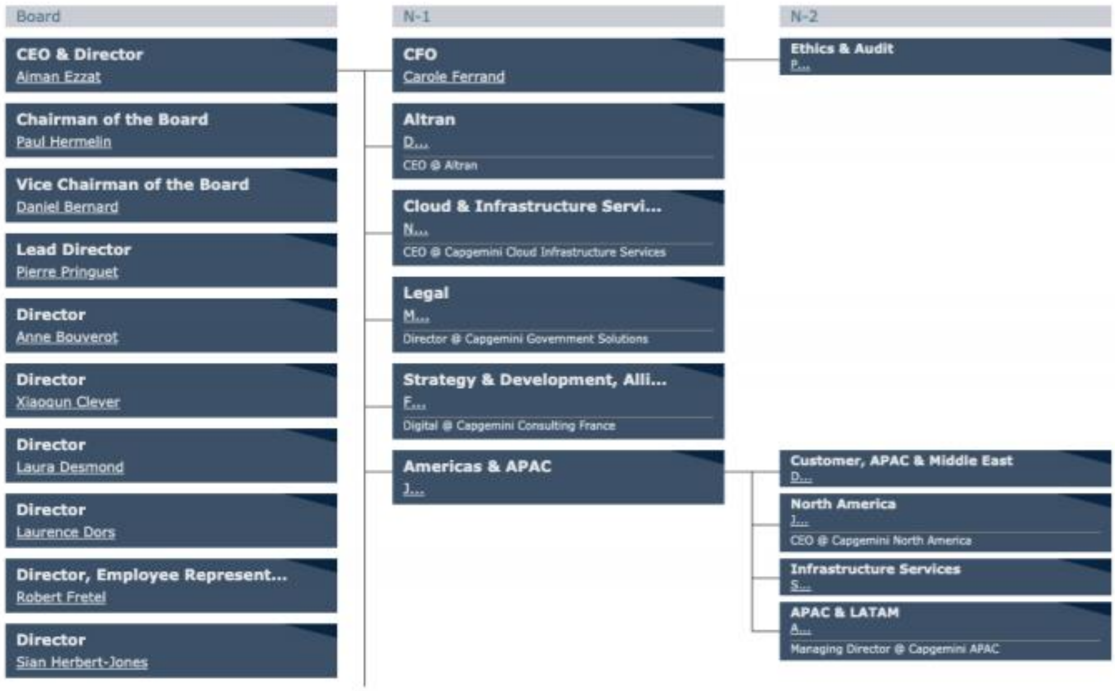
2013

- Capgemini rated as ‘Strong Positive’ in Gartner Marketscope for Western European Business Intelligence & Information Management Services1.
- IDC MarketScape: Positioned as a Leader for IT services in the EMEA Utilities Market 2013.
- HP: AllianceOne Mission-Critical Computing Partner of the Year Award.
- Pegasystems Inc: Two Pegasystems’ Global Strategic Alliance Partner Awards.
- Kennedy Consulting Research & Advisory: Positioned as Leader in the Vanguard of Digital Strategy Consulting Services.
- SAP: SAP® Pinnacle Award as Mobile Co-Innovation Partner of the Year.
- Ethisphere Institute: One of the World’s Top Most Ethical Companies.
- ASTDs: Our University Wins ASTD’s 2013 Excellence in Practice Award – with Capgemini Consulting!.

2012

- Gartner, Inc.: Positioned in the leaders quadrant in the « Magic Quadrant for Help Desk Outsourcing, Europe » (1).
- Gartner, Inc.: Positioned in the leaders quadrant in the “Magic Quadrant for SAP Implementation Service Providers, N America” (2).
- Gartner, Inc.: Positioned in the leaders quadrant in the “Magic Quadrant for SAP Application Services Providers, Europe” (3).
- Gartner, Inc.: Positioned in the leaders quadrant in the “Magic Quadrant for Oracle Application Service Providers, Europe » (4).
- Ovum: #1 for outsourced testing services.
- SAP Pinnacle Award in 2 categories : “Transformational Award: Unwired Enterprise” and “Business Process Outsourcing and Business Process as a Service Provider of the Year”.
- SAP: Global SAP certification in Run SAP® methodology.
- SAP: Partner Impact Award for Mobility.
- Microsoft: Outstanding Collaboration Partner of the Year.
- Microsoft: SAP-Microsoft Unite Innovating for Customer Success Award.
- Pegasystems: Partner Excellence Award for Customer Centricity.

Organization chart of the company



Board

CEO & Director
Aiman Ezzat

Chairman of the Board
Paul Hermelin

Vice Chairman of the Board
Daniel Bernard

Lead Director
Pierre Pringuet

Director
Anne Bouverot

Director
Xiaoqun Clever

Director
Laura Desmond

Director
Laurence Dors

Director, Employee Represent...
Robert Fretel

Director
Sian Herbert-Jones

N-1

CFO
Carole Ferrand

Altran
D...
CEO @ Altran

Cloud & Infrastructure Servi...
N...
CEO @ Capgemini Cloud Infrastructure Services

Legal
M...
Director @ Capgemini Government Solutions

Strategy & Development, Alli...
F...
Digital @ Capgemini Consulting France

Americas & APAC
J...

N-2

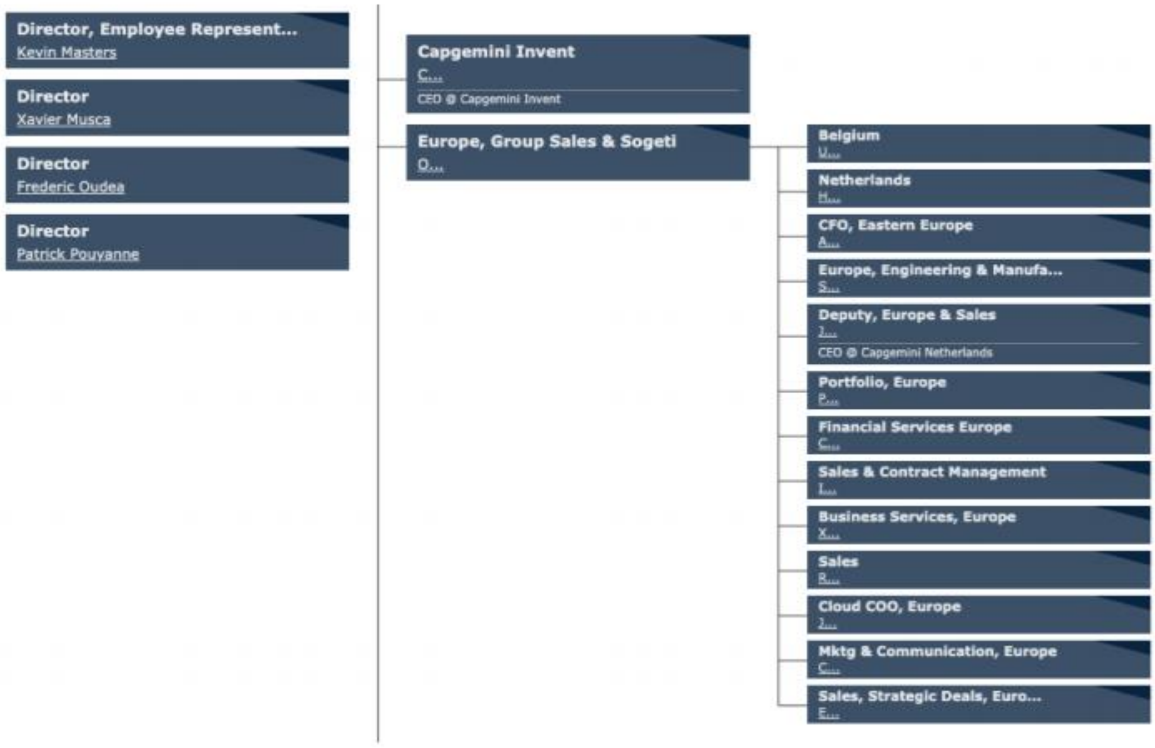
Ethics & Audit
P...

Customer, APAC & Middle East
D...

North America
J...
CEO @ Capgemini North America

Infrastructure Services
S...

APAC & LATAM
A...
Managing Director @ Capgemini APAC



Director, Employee Represent...
Kevin Masters

Director
Xavier Musca

Director
Frederic Oudea

Director
Patrick Pouyanne

Capgemini Invent
C...
CEO @ Capgemini Invent

Europe, Group Sales & Sogeti
O...

Belgium
U...

Netherlands
H...

CFO, Eastern Europe
A...

Europe, Engineering & Manufa...
S...

Deputy, Europe & Sales
J...
CEO @ Capgemini Netherlands

Portfolio, Europe
P...

Financial Services Europe
C...

Sales & Contract Management
L...

Business Services, Europe
X...

Sales
R...

Cloud COO, Europe
J...

Mktg & Communication, Europe
C...

Sales, Strategic Deals, Euro...
E...

CHAPTER 2

INTRODUCTION OF THE PROJECT UNDERTAKEN

1.FLIGHT MANAGEMENT SYSTEM:

Problem Statement:

To develop an Flight Management System with various defined roles for user as well as admin.

Description:

This project is aimed at developing Flight Management System for user and admin. This is a web based application that can be accessed throughout the web. This system can be used to add Flight, Update flight, Modify flight, delete flight, add airport, delete airport, modify airport, scheduling flight, searching scheduled flight, booking tickets etc. . This is an integrated system that contains both the user component and the administration component.

Scope:

Following are the functionality provided by the system:

There are two categories of people who would access the system: customer and administrator. Each of these would have some privileges.

1. The customer can:
   - Create his user account.
   - Login into the application.
   - Check for available flights.
   - Make a booking.
   - View the bookings made.

2. The administrator can:
   - Login into the application.
   - Add flight, schedule and route details.
   - View the flight, schedule and route details.
   - Cancel or modify the flight, schedule and route details.

The following functionalities have not been covered under the application:

- The application does not cover boarding pass generation and seating plans.
- Third party applications like email & sms integrations.
- Payments are not yet accepted by the application.
- User can not delete booking by Id.

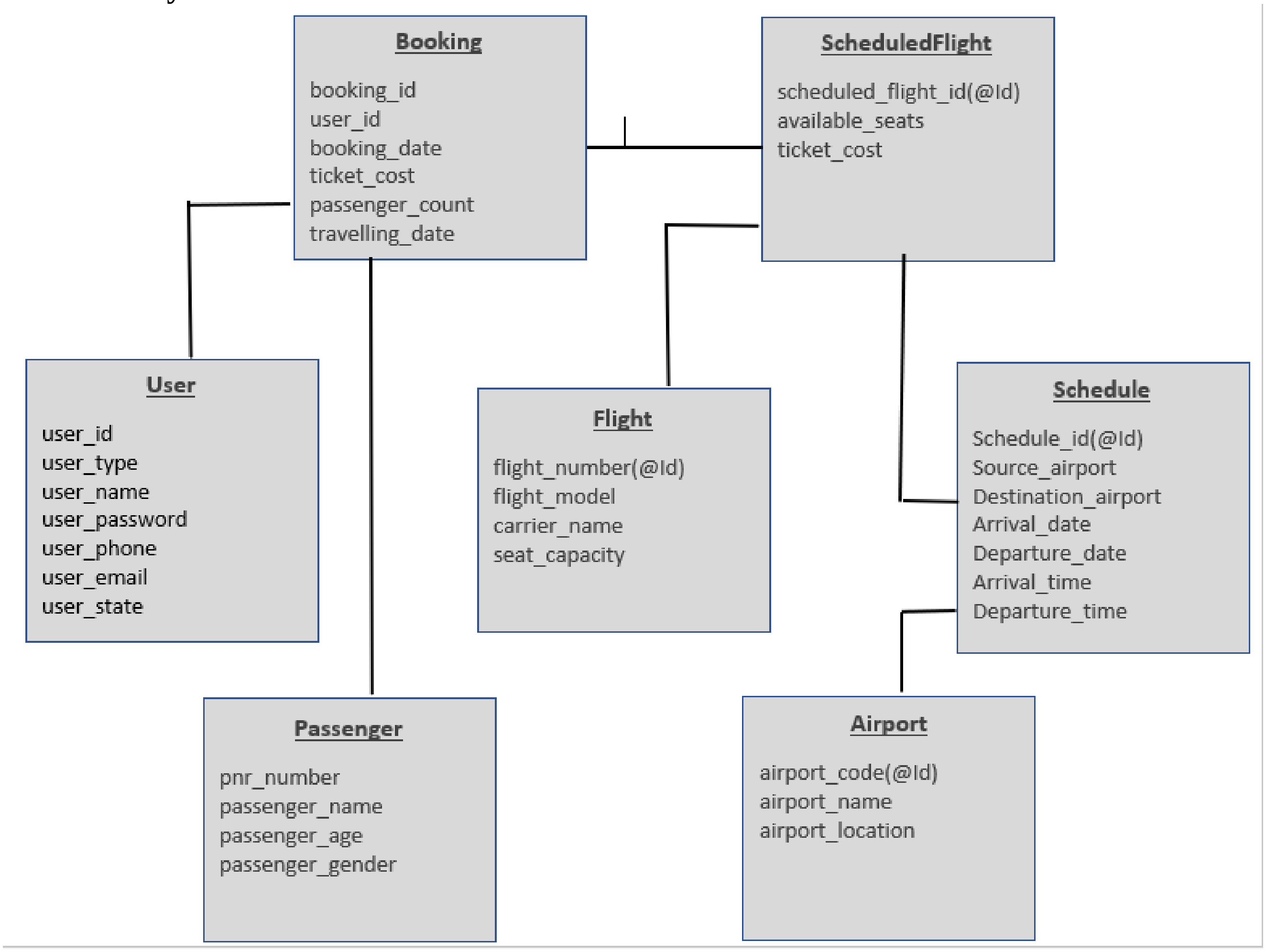
Role and profile:

In the project we were working with a team of four members and everyone was assigned different parts to work. As I was also the TL in the team my role was enhanced as I need to maintain the progress and work culture during the development of the project.

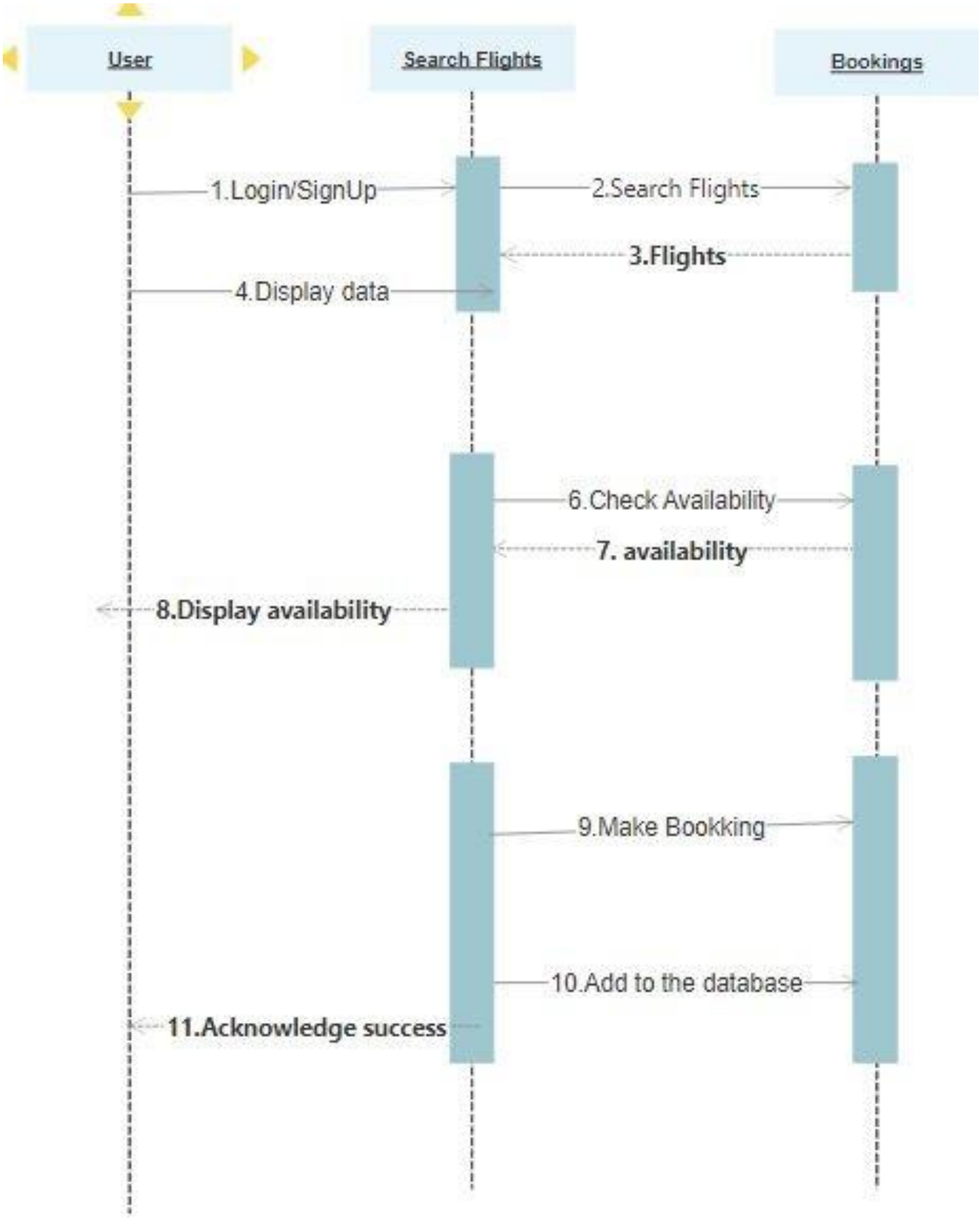
In the assigned project I was handling Scheduling in which I had to schedule a flight, view all scheduled flights, view flight, view schedule and search scheduled flight.

Class Diagram:

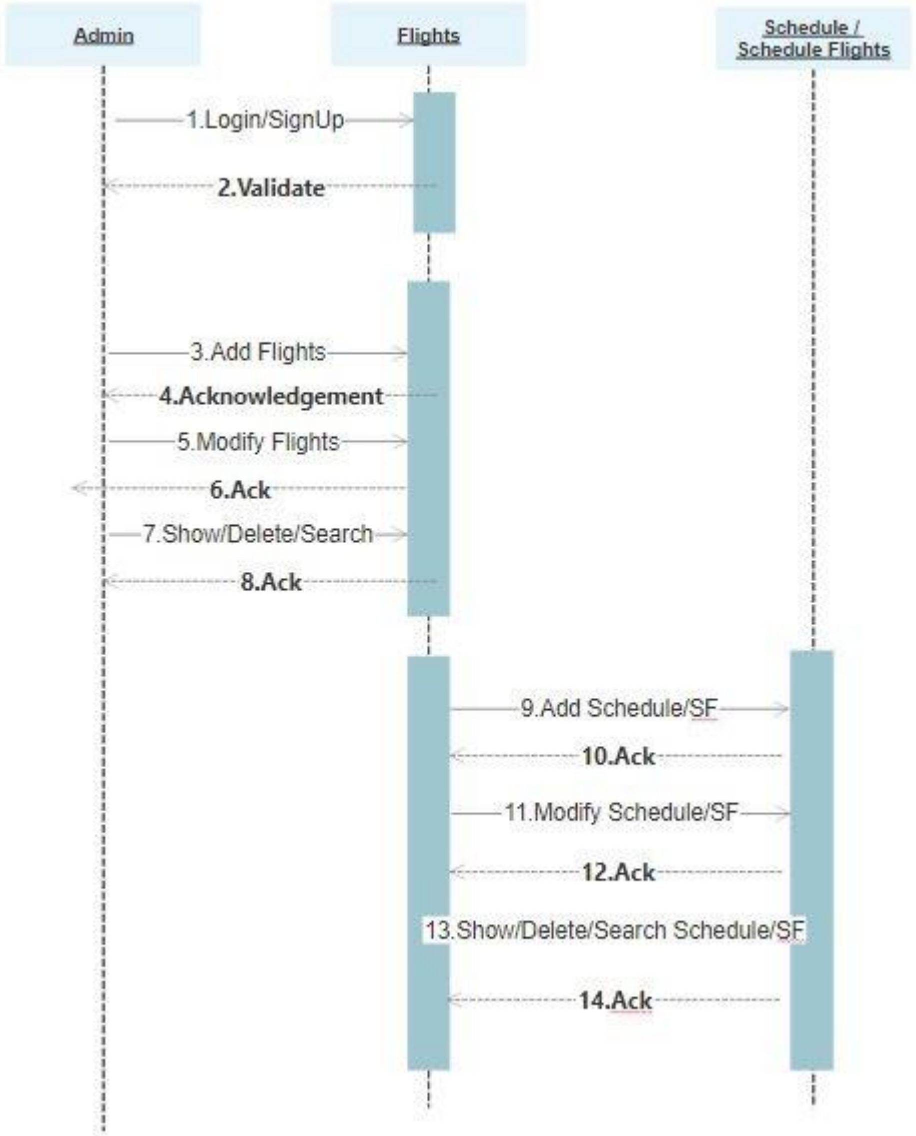
DTO Layer-



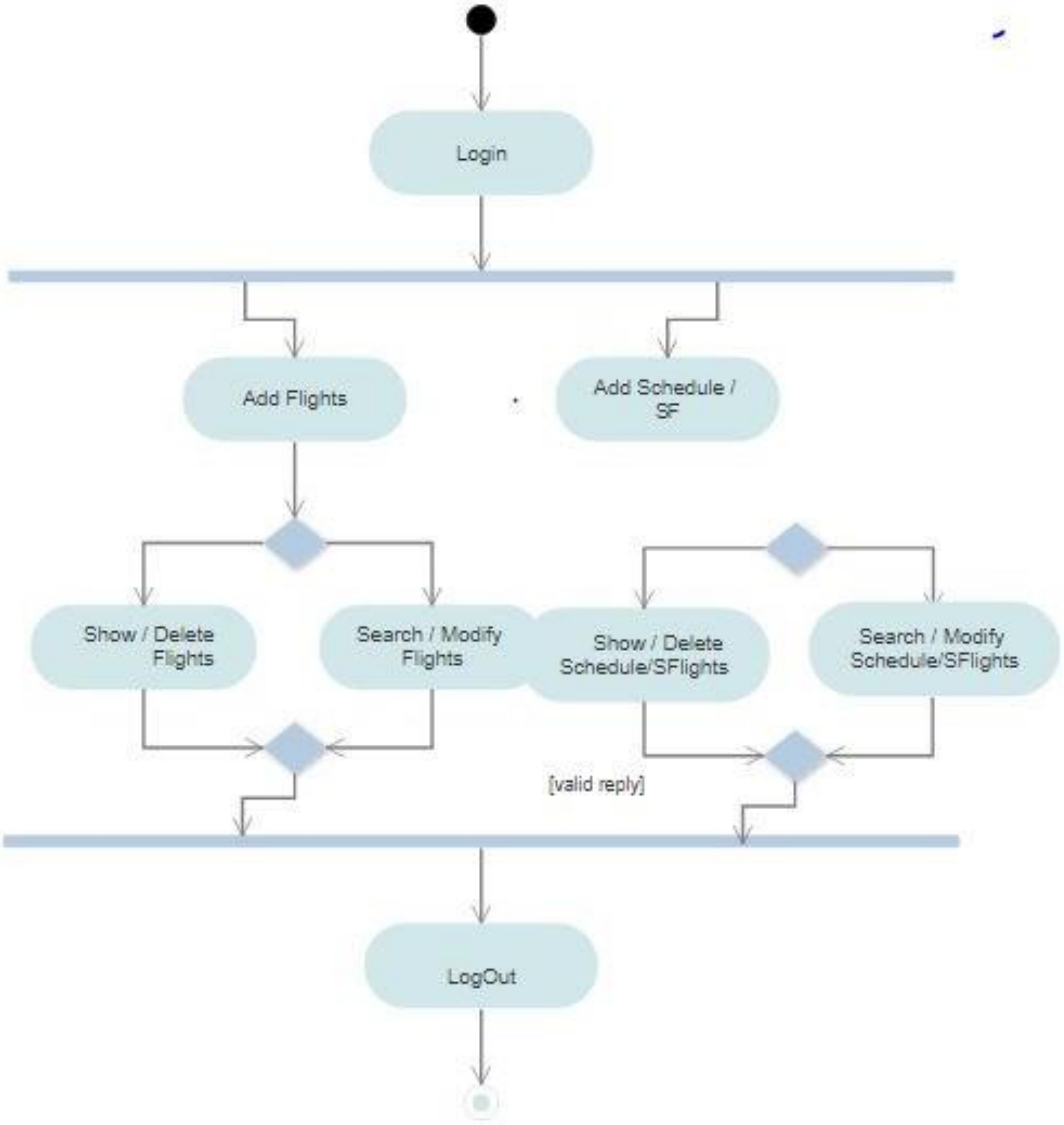
Sequence Diagram for User:



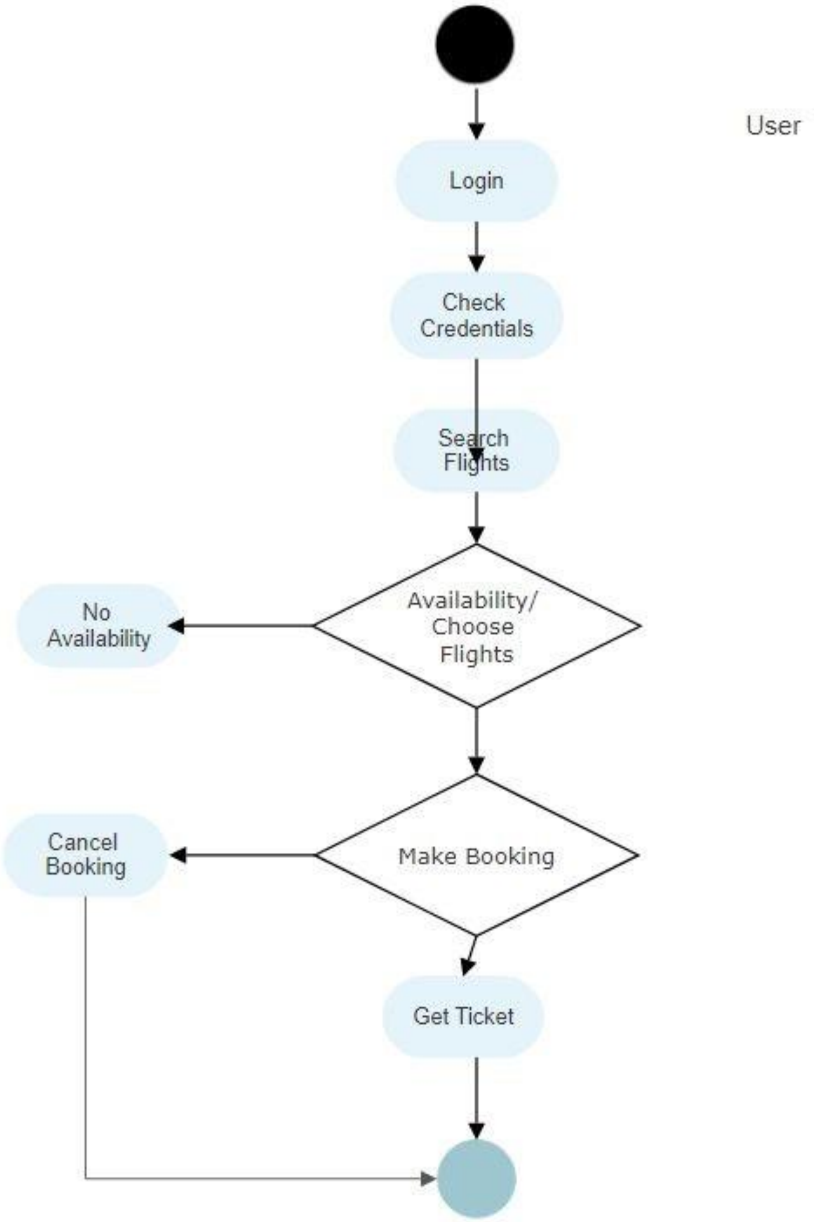
Sequence Diagram for Admin:



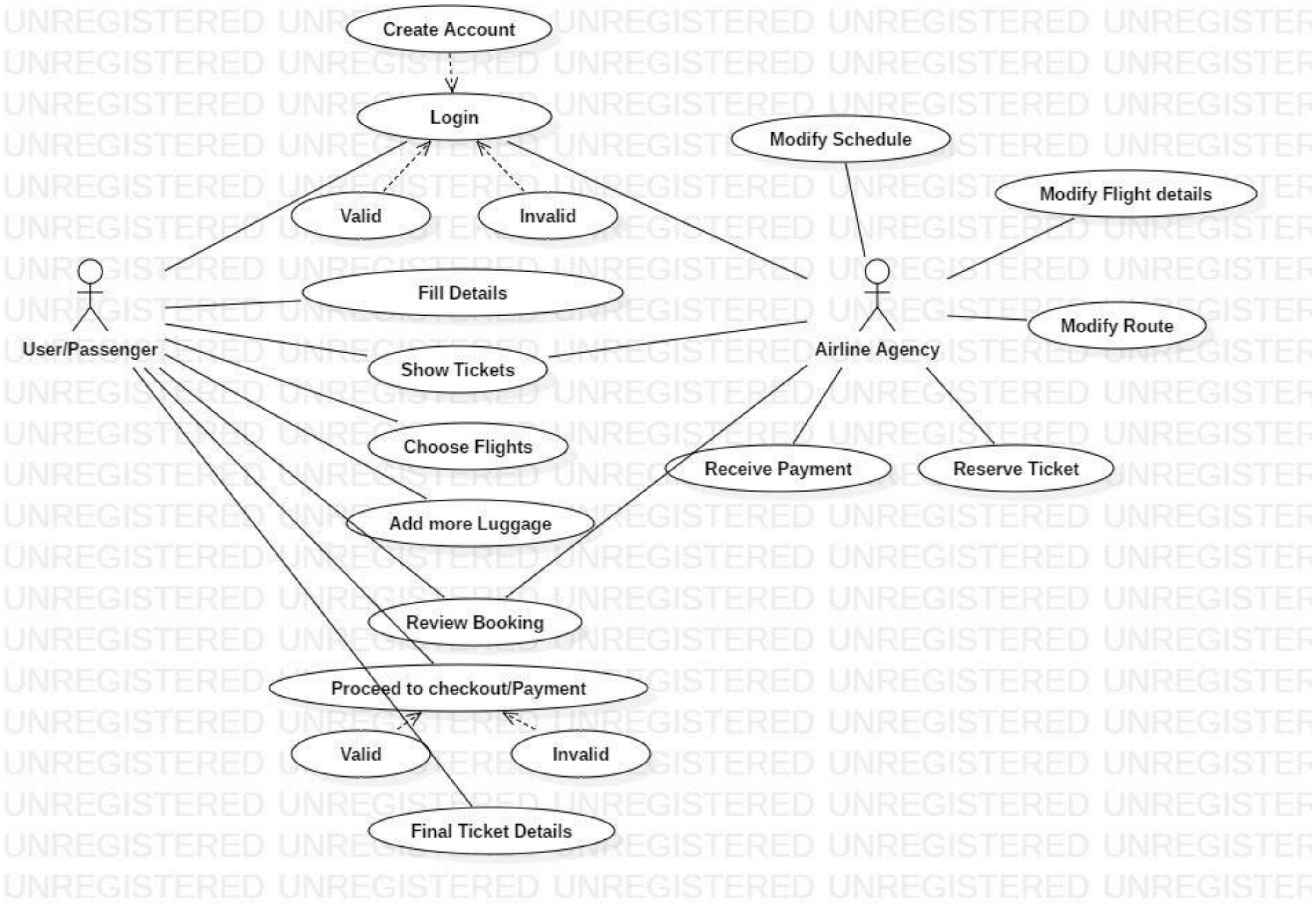
Activity Diagram for Admin:



Activity Diagram for User:



Use Case Diagram:



2.TECHNOLOGY USED

2.1 INTRODUCTION

The technology which is learned and performed is Java Fullstack and Angular 6 which is the entirety of a computer system or application, comprising both the front end and the back end. The topics which were covered under this are: Front End: Angular 6.

Back End: Java Fullstack.

2.1.1 FRONT END TECHNOLOGY

The front end of a website is the part that users interact with. Everything that you see when you're navigating around the Internet, from fonts and colors to dropdown menus and sliders, is a combo of HTML, CSS, and TypeScript being controlled by your computer's browser. Front-end web development is the practice of converting data to graphical interface for user to view and interact with data through digital interaction using HTML, CSS and TypeScript.

2.1.1.1 ANGULAR 6

Angular 6 is a JavaScript (actually a TypeScript based open-source full-stack web application) framework which makes you able to create reactive Single Page Applications (SPAs). Angular 6 is completely based on components. It consists of several components which forms a tree structure with parent and child components. Angular's versions beyond 2+ are generally known as Angular only. The very first version Angular 1.0 is known as AngularJS.

2.1.1.1.1 SPA(SINGLE PAGE APPLICATION)

A single page application is a web application or a website which provides users a very fluid, reactive and fast experience similar to a desktop application. It contains menu, buttons and blocks on a single page and when a user clicks on any of them; it dynamically rewrites the current page rather than loading entire new pages from a server. That's the reason behind its reactive fast speed.

2.1.1.1.2 FEATURES OF ANGULAR

- Updates regarding Application Performance.
- Angular Material & CDK.
- Virtual Scrolling.
- Improved Accessibility of Selects.
- Supports Content Projection using web standard for custom elements.
- Dependency updates regarding TypeScript 3.1, RxJS 6.3, Node 10.

Following are the Angular CLI commands to create the first Angular app.

- npm install -g @angular/cli
- ng new my-dream-app
- cd my-dream-app
- ng serve

2.1.1.1.3 Angular 6 Architecture

Angular 6 is a platform and framework which is used to create client applications in HTML and TypeScript. Angular 6 is written in TypeScript. Angular 6 implements core and optional functionality as a set of TypeScript libraries which you import in your app. NgModules are the basic building blocks of an Angular 6 application. They provide a compilation context for components. An Angular 6 app is defined by a set of NgModules and NgModules collect related code into functional sets. An Angular 6 app always has at least a root module which enables bootstrapping, and typically has many other feature modules. Components define views, which are the sets of screen elements that are chosen and modified according to your program logic and data by Angular 6. Components use services, which provide specific functionality not directly related to views. Service providers can be injected into components as dependencies, making your code modular, reusable, and efficient.

Components: Components and services both are simply classes with decorators that mark their types and provide metadata which guide Angular to do things. Every 2 Angular application always has at least one component known as root component that connects a page hierarchy with page DOM. Each component defines a class that contains application data and logic, and is associated with an HTML template that defines a view to be displayed in a target environment.

Metadata of Component class: The metadata for a component class associates it with a template that defines a view. A template combines ordinary HTML with Angular directives and binding markup that allow Angular to modify the HTML before rendering it for display. The metadata for a service class provides the information Angular needs to make it available to components through dependency injection (DI).

Modules: Angular 6 NgModules are different from other JavaScript modules. Every Angular 6 app has a root module known as AppModule. It provides the bootstrap mechanism that launches the application. Generally, every Angular 6 app contains many functional modules.

Some important features of Angular 6 Modules: Angular 6 NgModules import the functionalities form other NgModules just like other JavaScript modules. NgModules allow their own functionality to be exported and used by other NgModules. For example, if you want to use the router service in your app, you can import the Router NgModule.

Template, Directives and Data Binding: In Angular 6, a template is used to combine HTML with Angular Markup and modify HTML elements before displaying them. Template directives provide program logic, and binding markup connects your application data and the DOM.

There are two types of data binding:

Event Binding: Event binding is used to bind events to your app and respond to user input in the target environment by updating your application data. Read more about event binding.

Property Binding: Property binding is used to pass data from component class and facilitates you to interpolate values that are computed from your application data into the HTML. Read more about property binding.

Services and Dependency Injection: In Angular 6, developers create a service class for data or logic that isn't associated with a specific view, and they want to share across components. Dependency Injection (DI) is used to make your component classes lean and efficient. DI doesn't fetch data from the server, validate user input, or log directly to the console; it simply renders such tasks to services.

Routing: In Angular 6, Router is an NgModule which provides a service that facilitates developers to define a navigation path among the different application states and view hierarchies in their app. It works in the same way as a browser's navigation works. i.e Enter a URL in the address bar and the browser will navigate to that corresponding page. Click the link on a page and the browser will navigate to a new page. Click the browser's back or forward buttons and the browser will navigate backward or forward according to your seen history pages.

How does Router work: The router maps URL-like paths to views instead of pages. Whenever a user performs an action, such as clicking a link that would load a new page in the browser, the router intercepts the browser's behavior, and shows or hides view hierarchies. If the router finds that the current application state requires particular functionality, and the module that defines it hasn't been loaded, the router can lazy-load the module on demand. The router interprets a link URL according to your app's view navigation rules and data state. You can navigate to new views when the user clicks a button or selects from a drop box, or in response to some other stimulus from any source. The router logs activity in the browser's history, so the back and forward buttons work as well. To define navigation rules, you associate navigation paths with your components. A path uses a URL-like syntax that integrates your program data, in much the same way that template syntax integrates your views with your program data. You can then apply program logic to choose which views to show or to hide, in response to user input and your own access rules.

2.1.1.1.4 Angular 6 Directives:

Directives are instructions in the DOM. They specify how to place your components and business logic in the Angular. Directives are js class and declared as @directive. There are 3 directives in Angular.

Component Directives: Component directives are used in main class. They contain the detail of how the component should be processed, instantiated and used at runtime.

Structural Directives: Structural directives start with a * sign. These directives are used to manipulate and change the structure of the DOM elements. For example, *ngIf and *ngFor.

Attribute Directives: Attribute directives are used to change the look and behavior of the DOM elements. For example: ngClass, ngStyle etc.

2.1.1.1.5 Angular 6 Databinding:

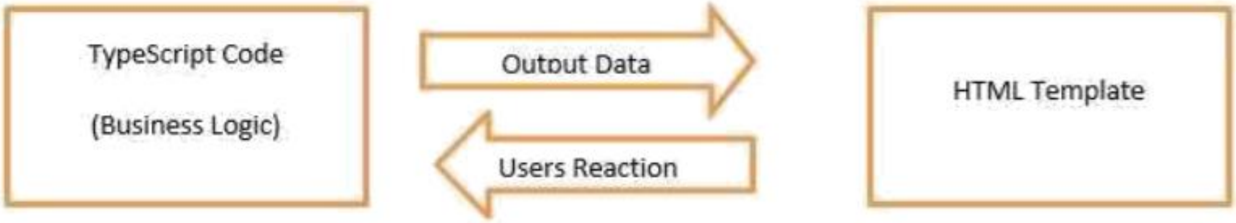
Databinding is a powerful feature of Angular. Angular Databinding is used for communication. It is used to communicate between your TypeScript code (your business logic) and the other component which is shown to the users i.e. HTML layout. Databinding is necessary because when we write the code in TypeScript, it is compiled to JavaScript and the result is shown on HTML layout. Thus, to show the correct and spontaneous result to the users, a proper communication is necessary. That's why databinding is used in Angular.
There is two type of databinding:

One-way databinding:

One-way databinding is a simple one way communication where HTML template is changed when we make changes in TypeScript code or In one-way databinding, the value of the Model is used in the View (HTML page) but you can't update Model from the View. Angular Interpolation / String Interpolation, Property Binding, and Event Binding are the example of one-way databinding.

Two-way databinding:

In two-way databinding, automatic synchronization of data happens between the Model and the View. Here, change is reflected in both components. Whenever you make changes in the Model, it will be reflected in the View and when you make changes in View, it will be reflected in Model. This happens immediately and automatically, ensures that the HTML template and the TypeScript code are updated at all times.



2.1.1.1.6 Angular 6 String Interpolation:

In Angular, String interpolation is used to display dynamic data on HTML template (at user end). It facilitates you to make changes on component.ts file and fetch data from there to HTML template (component.html file). String Interpolation and Property binding both are used for same purpose i.e. one-way databinding. But the problem is how to know which one is best suited for your application. String Interpolation and Property Binding doth are about one-way data binding. They both flow a value in one direction from our components to HTML elements. String Interpolation is a special syntax which is converted to property binding by Angular. It's a convenient alternative to property binding. When you need to concatenate strings, you must use interpolation instead of property binding.

2.1.1.1.7 Angular 6 Event Binding:

Angular facilitates us to bind the events along with the methods. This process is known as event binding. Event binding is used with parenthesis ().

2.1.1.1.8 Anguar 6 Forms:

Angular forms are used to handle user's input. We can use Angular form in our application to enable users to log in, to update profile, to enter information, and to perform many other data-entry tasks. In Angular 6, there are 2 approaches to handle user's input through forms:

Reactive forms

Template-driven forms

Both approaches are used to collect user input events from the view, validate the user input, create a form model and data model to update, and provide a way to track changes. Both Reactive forms and Template-driven forms manage and process data differently. Each offers different advantages.

Reactive Forms: Reactive forms are more robust. Reactive forms are more scalable, reusable, and testable. They are most preferred to use if forms are a key part of your application, or your application is already built using reactive patterns. In both cases, reactive forms are best to use.

Template-driven Forms: Template-driven forms are best if you want to add a simple form to your application. For example: email list signup form. Template-driven forms are easy to use in the application but they are not as scalable as Reactive forms. Template-driven forms are mainly used if your application's requires a very basic form and logic. It can easily be managed in a template.

Similarity between Reactive Forms and Template-driven Forms.

There are some building blocks which are shared by both reactive and template-driven forms:

FormControl: It is used to track the value and validation status of an individual form control.

FormGroup: It is used to track the same values and status for a collection of form controls.

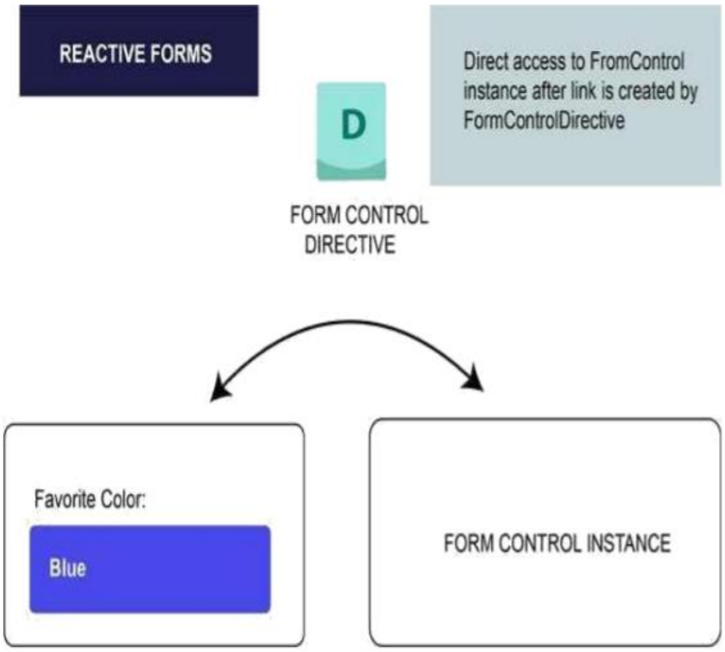
FormArray: It is used to track the same values and status for an array of form controls.

ControlValueAccessor: It is used to create a bridge between Angular FormControl instances and native DOM elements.

Form Model Setup Form model setup is used to track value changes between Angular forms and form input elements. Let's take an example to see how the form model is defined and created.

Form model setup in Reactive forms: See the below component with an input field for a single control implemented using reactive forms.

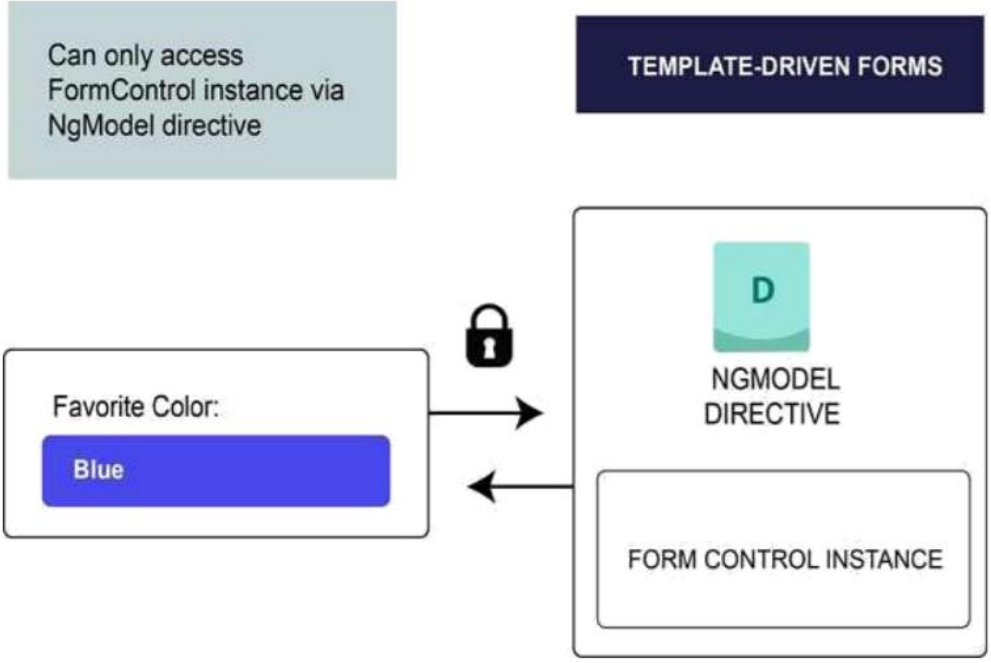
In reactive forms, the form model is the source of truth. The source of truth provides the value and status of the form element at a given point in time. Here, in the example above, the form model is the **FormControl** instance.


REACTIVE FORMS
D
FORM CONTROL DIRECTIVE
Direct access to FromControl instance after link is created by FormControlDirective
Favorite Color:
Blue
FORM CONTROL INSTANCE

In reactive forms, the form model is explicitly defined in component class. After that the reactive form directive (here, it is: FormControlDirective) links the existing FormControl instance to a specific form element in the view using a value accessor (ControlValueAccessor instance).

Form model setup in Template-driven Forms: See the same above component with an input field for a single control implemented using template-driven forms.

In template-driven forms, the source of truth is template itself.



The form model abstraction promotes simplicity over structure. The template-driven form directive NgModel creates and manages the FormControl instance for a given form element. It's less explicit, but it removes the direct control over the form model.

2.2 BACKEND

The back end of a web application is the part that controls all the functionalities, business logics of the web application. All the requests made by the customer are processed in the backend and the results are shown back to the customers. The backend contains the core java, spring, database, hibernate.

2.2.1 CORE JAVA

Java is a programming language developed by James Gosling with other team members named Mike Sheridan and Patrick Naughton as also called as Green Team in 1995 for Sun Microsystems for digital devices such as set-top boxes, televisions etc. Java 9 is a general-purpose, concurrent, object-oriented, class-based, and runtime environment (JRE) which consists of JVM which is the cornerstone of the Java platform.

2.2.1.1 Features of JAVA

Simple: Java has made life easier by removing all the complexities such as pointers, operator overloading as you see in C++ or any other programming language.

Portable: Java is platform independent which means that any application written on one platform can be easily ported to another platform.

Object-oriented: Everything is considered to be an “object” which possess some state, behavior and all the operations are performed using these objects.

Secured: All the code is converted in bytecode after compilation, which is not readable by a human. and java does not use an explicit pointer and run the programs inside the sandbox to prevent any activities from untrusted sources. It enables to develop virus-free, tamper-free systems/applications.

Dynamic: It has the ability to adapt to an evolving environment which supports dynamic memory allocation due to which memory wastage is reduced and performance of the application is increased.

Distributed: Java provides a feature which helps to create distributed applications. Using Remote Method Invocation (RMI), a program can invoke a method of another program across a network and get the output. You can access files by calling the methods from any machine on the internet.

Robust: Java has a strong memory management system. It helps in eliminating error as it checks the code during compile and runtime.

High Performance: Java achieves high performance through the use of bytecode which can be easily translated into native machine code. With the use of JIT (Just-In-Time) compilers, Java enables high performance.

2.2.1.2 COMPONENTS IN JAVA

JVM (Java Virtual Machine): It is an abstract machine. It is a specification that provides a run-time environment in which Java byte code can be executed. It follows three notations:

Specification: It is a document that describes the implementation of the Java virtual machine. It is provided by Sun and other companies.

Implementation: It is a program that meets the requirements of JVM specification.

Runtime Instance: An instance of JVM is created whenever you write a java command on the command prompt and run the class.

JRE (Java Runtime Environment): JRE refers to a runtime environment in which Java bytecode can be executed. It implements the JVM (Java Virtual Machine) and provides all the class libraries and other support files that JVM uses at runtime. So JRE is a software package that contains what is required to run a Java program. Basically, it's an implementation of the JVM which physically exists.

2.2.2 SPRING FRAMEWORK

The Spring Framework was first released in 2004. After that there has been a significant major revision, such as Spring 2.0 provided XML namespaces and AspectJ support, Spring 2.5 provide annotation-driven configuration, Spring 3.0 provided a Javabased @Configuration model. The latest release of the spring framework is 4.0. it is released with the support for Java 8 and Java EE 7 technologies.

Spring framework uses various new technologies such as Aspect-Oriented Programming (AOP), Plain Old Java Object (POJO), and dependency injection (DI), to develop enterprise applications, thereby eliminating the complexities involved while developing enterprise applications using EJB, Spring is an open source lightweight framework that allows Java EE 7 developers to build simple, reliable, and scalable enterprise applications. This framework is a great way to help you manage your business objects. It is the development of Web applications, as well as much of the simplest Java frameworks and application programming interfaces (APIs), such as Java database connectivity (JDBC), JavaServer Pages (JSP), and Java Servlet.

The Spring framework can be considered as a collection of sub-frameworks, also called layers, such as Spring AOP. Spring Object-Relational Mapping (Spring ORM). Spring Web Flow, and Spring Web MVC. You can use any of these modules separately while constructing a Web application. The modules may also be grouped together to provide better functionalities in a Web application.

2.2.2.1 Features of the Spring Framework

The features of the Spring framework such as IoC, AOP, and transaction management, make it unique among the list of frameworks. Some of the most important features of the Spring framework are as follows:

IoC container: Refers to the core container that uses the DI or IoC pattern to implicitly provide an object reference in a class during runtime. This pattern acts as an alternative to the service locator pattern. The IoC container contains assembler code that handles the configuration management of application objects.The Spring framework provides two packages, namely org.springframework.beans and org.springframework.context which helps in providing the functionality of the IoC container.

Data access framework: Allows the developers to use persistence APIs, such as JDBC and Hibernate, for storing persistence data in database. It helps in solving various problems of the developer, such as how to interact with a database connection, how to make sure that the connection is closed, how to deal with exceptions, and how to implement transaction management It also enables the developers to easily write code to access the persistence data throughout the application.

Spring MVC framework: Allows you to build Web applications based on MVC architecture. All the requests made by a user first go through the controller and are then dispatched to different views, that is, to different JSP pages or Servlets. The form handling and form validating features of the Spring MVC framework can be easily integrated with all popular view technologies such as ISP, Jasper Report, FreeMarker, and Velocity.

Transaction management: Helps in handling transaction management of an application without affecting its code. This framework provides Java Transaction API (JTA) for global transactions managed by an application server and local transactions managed by using the JDBC Hibernate, Java Data Objects (JDO), or other data access APIs. It enables the developer to model a wide range of transactions on the basis of Spring’s declarative and programmatic transaction management.

Spring Web Service: Generates Web service endpoints and definitions based on Java classes, but it is difficult to manage them in an application. To solve this problem, Spring 12 Web Service provides layered-based approaches that are separately managed by Extensible Markup Language (XML) parsing (the technique of reading and manipulating XML). Spring provides effective mapping for transmitting incoming XML message request to an object and the developer to easily distribute XML message (object) between two machines.

JDBC abstraction layer: Helps the users in handling errors in an easy and efficient manner. The JDBC programming code can be reduced when this abstraction layer is implemented in a Web application. This layer handles exceptions such as DriverNotFound. All SQLExceptions are translated into the DataAccessException class. Spring’s data access exception is not JDBC specific and hence Data Access Objects (DAO) are not bound to JDBC only.

Spring TestContext framework: Provides facilities of unit and integration testing for the Spring applications. Moreover, the Spring TestContext framework provides specific integration testing functionalities such as context management and caching DI of test fixtures, and transactional test management with default rollback semantics.

2.2.3 Hibernate Framework

Hibernate is a framework which provides some abstraction layer means programmer dont have to worry about the implementations, Hibernate do implementations for you internally like Establishing a connection with the database, writing query to perform CRUD operations etc. It is a java framework which is used to develop persistence logic. Persistence logic means to store and process the data for long use. More precisely Hibernate is a open source, non-invasive, light-weight java ORM(Object relational mapping) framework to develop objects which is independent of the database software and make independent persistence logic in all JAVA, JEE. Framework means it is a special install-able software that provides abstraction layer on one or more technologies like JDBC ,Servlet etc to simplify or reduce the complexity for development process.

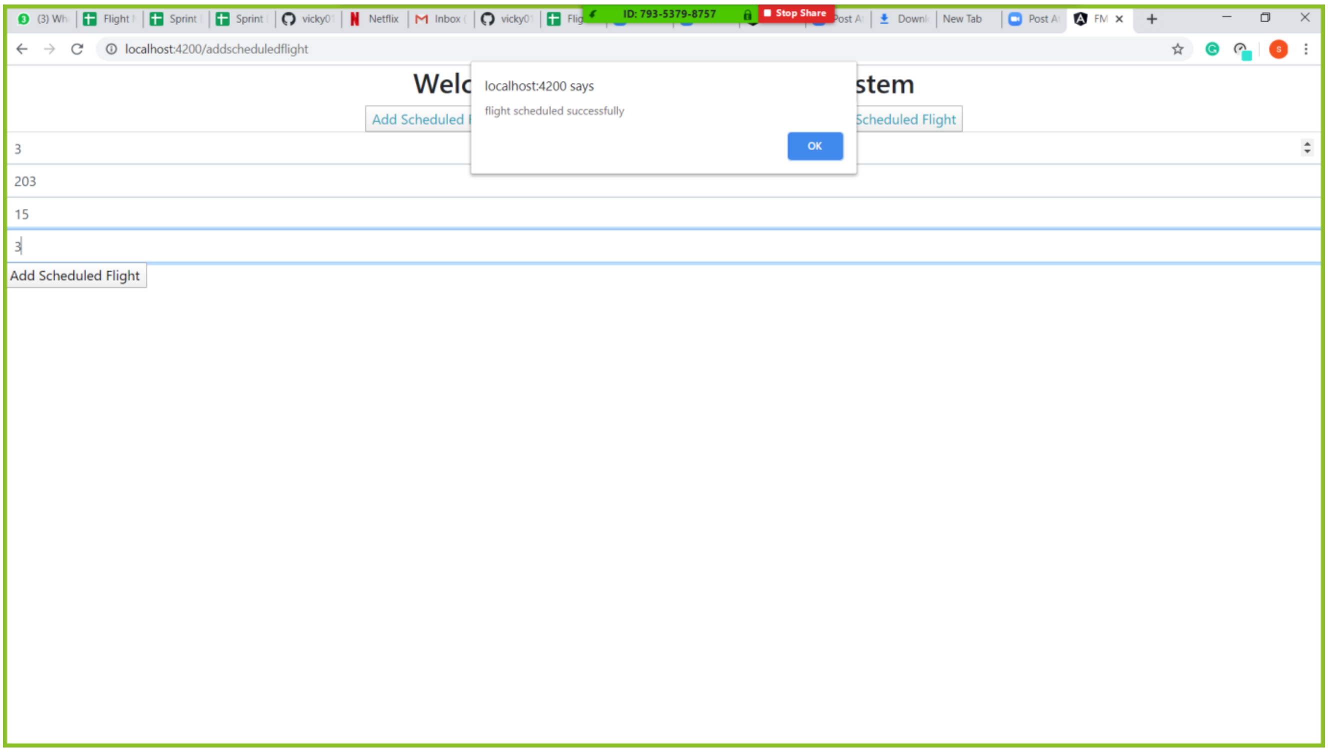
2.2.3.1 Hibernate Functionalities

- Hibernate framework support Auto DDL operations. In JDBC manually we have to create table and declare the data-type for each and every column. But Hibernate can do DDL operations for you internally like creation of table, drop a table, alter a table etc.
- Hibernate supports Auto Primary key generation. It means in JDBC we have to manually set a primary key for a table. But Hibernate can this task for you.
- Hibernate framework is independent of Database because it supports **HQL (Hibernate Query Language)** which is not specific to any database, whereas JDBC is database dependent.
- In Hibernate, Exception Handling is not mandatory, whereas In JDBC exception handling is mandatory.
- Hibernate supports Cache Memory whereas JDBC does not support cache memory.
- Hibernate is a ORM tool means it support Object relational mapping. Whereas JDBC is not object oriented moreover we are dealing with values means primitive data. In hibernate each record is represented as a Object but in JDBC each record is nothing but a data which is nothing but primitive values.

  An ORM tool simplifies the data creation, data manipulation and data access. It is programming technique that maps the object to the data stored in the database.

CHAPTER 3

SNAPSHOTS OF PROJECT



localhost:4200/addscheduledflight

localhost:4200 says

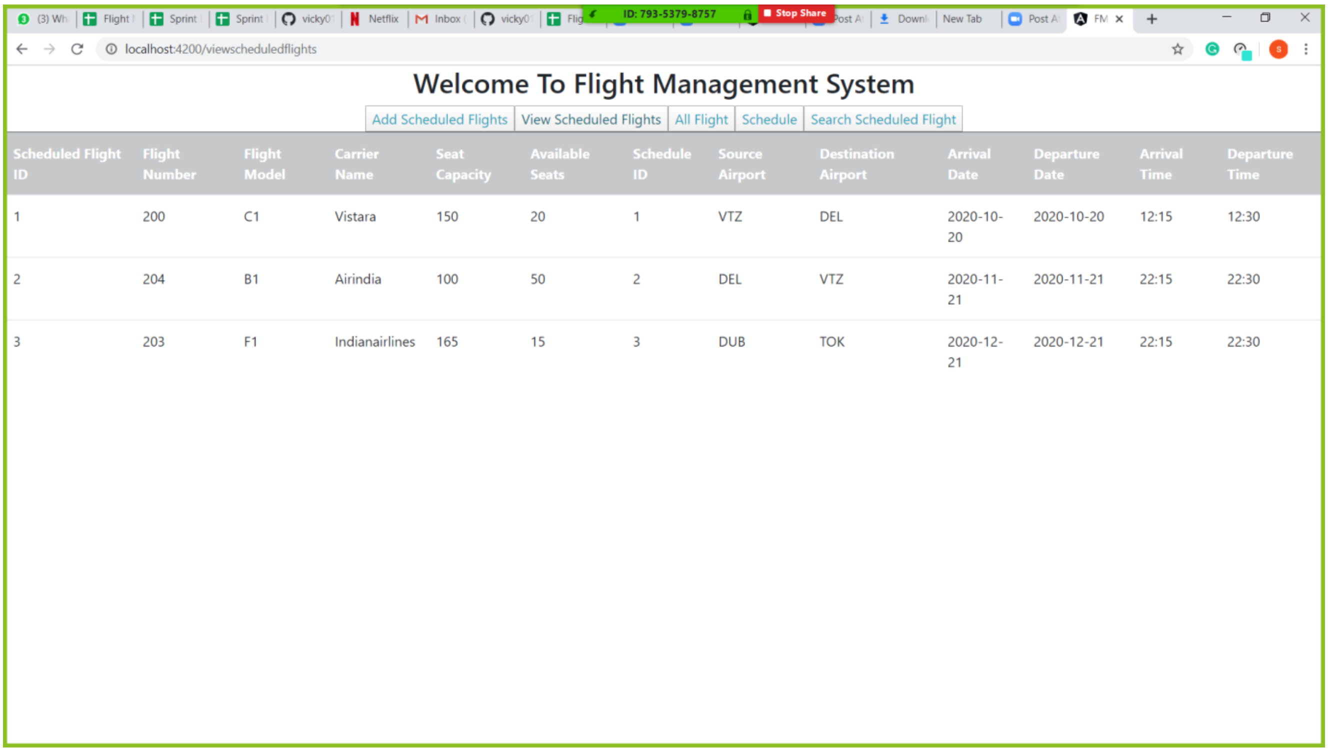
flight scheduled successfully

3

203

15

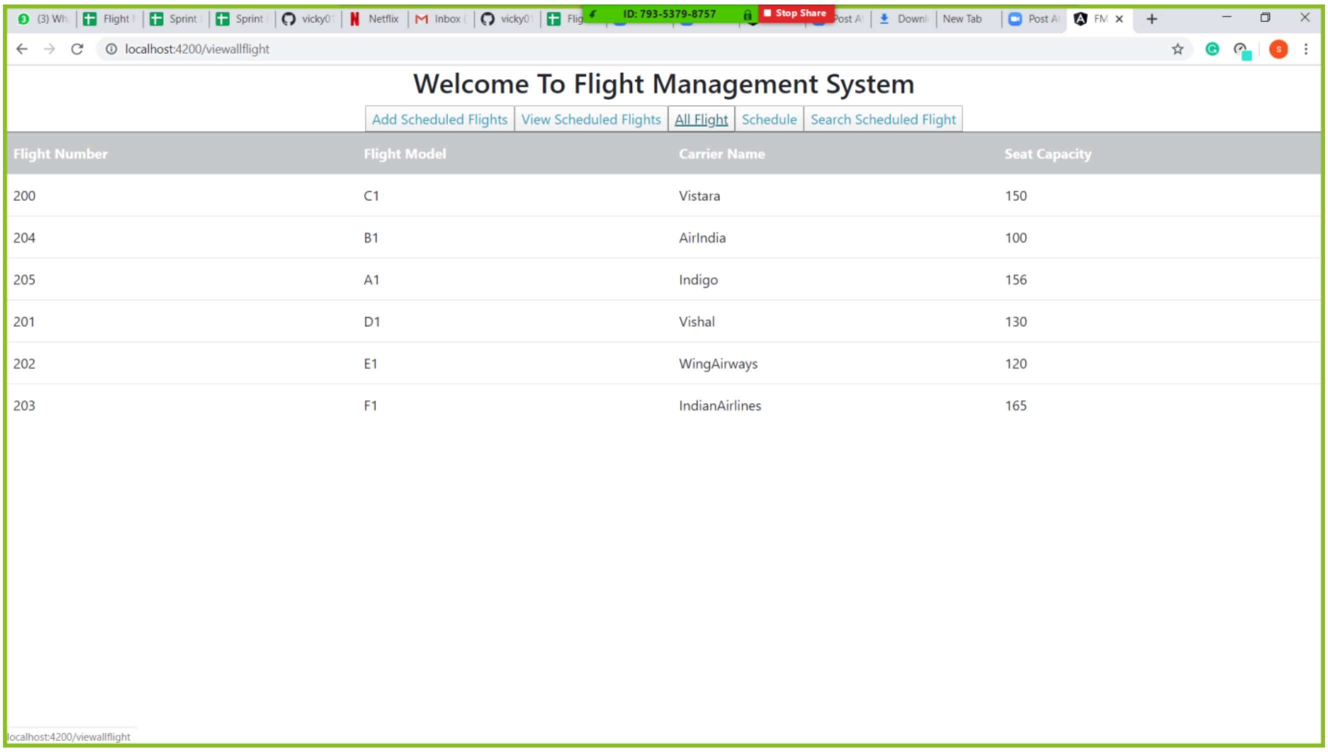
3



localhost:4200/viewscheduledflights

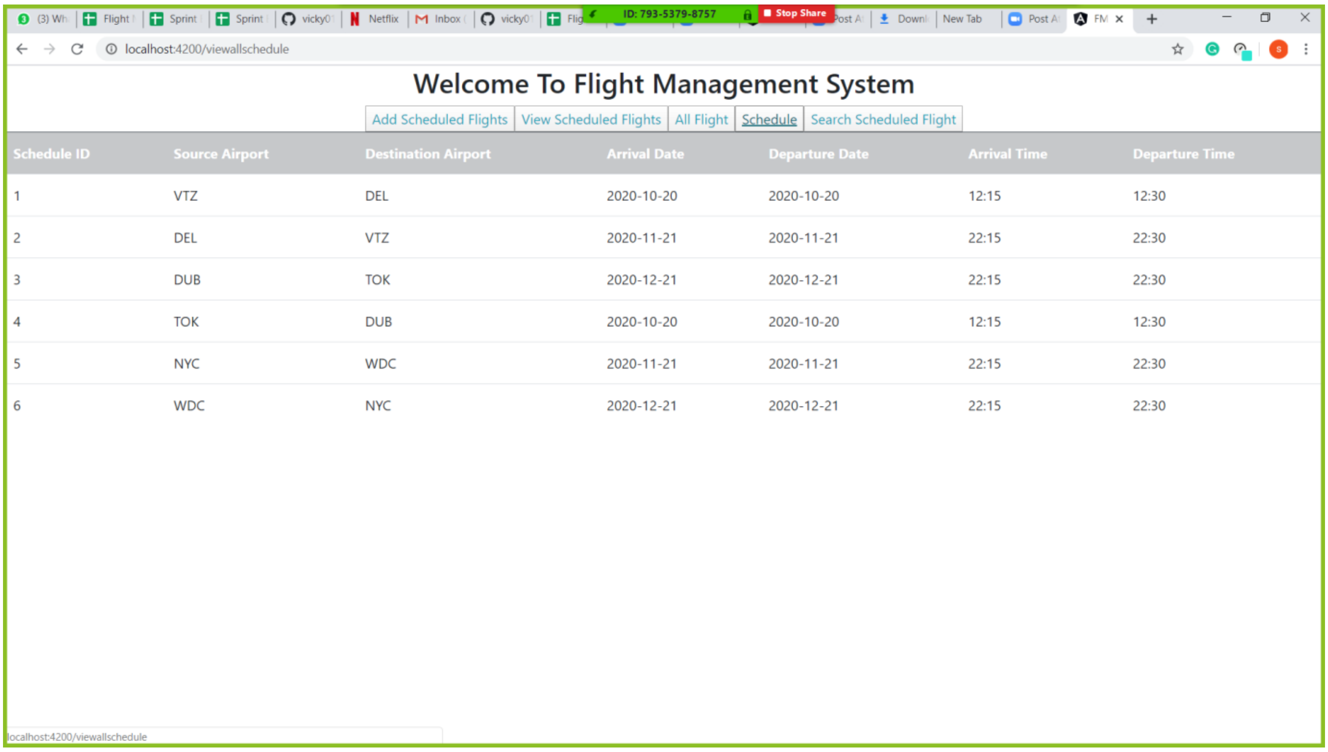
Welcome To Flight Management System

Scheduled Flight ID	Flight Number	Flight Model	Carrier Name	Seat Capacity	Available Seats	Schedule ID	Source Airport	Destination Airport	Arrival Date	Departure Date	Arrival Time	Departure Time
1	200	C1	Vistara	150	20	1	VTZ	DEL	2020-10-20	2020-10-20	12:15	12:30
2	204	B1	Airindia	100	50	2	DEL	VTZ	2020-11-21	2020-11-21	22:15	22:30
3	203	F1	Indianairlines	165	15	3	DUB	TOK	2020-12-21	2020-12-21	22:15	22:30



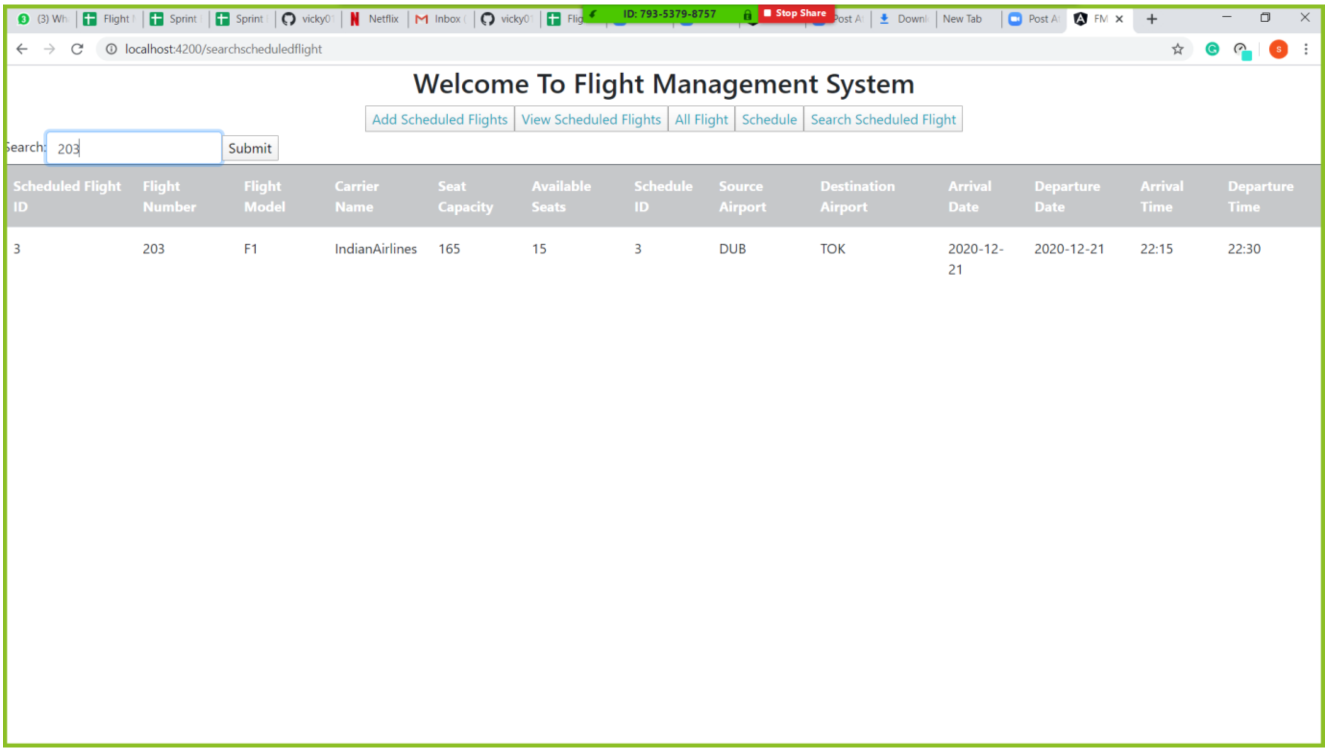
Welcome To Flight Management System

Flight Number	Flight Model	Carrier Name	Seat Capacity
200	C1	Vistara	150
204	B1	AirIndia	100
205	A1	Indigo	156
201	D1	Vishal	130
202	E1	WingAirways	120
203	F1	IndianAirlines	165



Welcome To Flight Management System

Schedule ID	Source Airport	Destination Airport	Arrival Date	Departure Date	Arrival Time	Departure Time
1	VTZ	DEL	2020-10-20	2020-10-20	12:15	12:30
2	DEL	VTZ	2020-11-21	2020-11-21	22:15	22:30
3	DUB	TOK	2020-12-21	2020-12-21	22:15	22:30
4	TOK	DUB	2020-10-20	2020-10-20	12:15	12:30
5	NYC	WDC	2020-11-21	2020-11-21	22:15	22:30
6	WDC	NYC	2020-12-21	2020-12-21	22:15	22:30



Welcome To Flight Management System

Search: 203 Submit

Scheduled Flight ID	Flight Number	Flight Model	Carrier Name	Seat Capacity	Available Seats	Schedule ID	Source Airport	Destination Airport	Arrival Date	Departure Date	Arrival Time	Departure Time
3	203	F1	IndianAirlines	165	15	3	DUB	TOK	2020-12-21	2020-12-21	22:15	22:30

CHAPTER 4

CONCLUSION AND ARCHITECTURE OF PROJECT

Architecture is kind of an overloaded term, so we should probably dig deeper into what the term really means in the context of layers. For implementation of Flight Management System I used **Layered Architecture**. The main idea behind Layered Architecture is a separation of concerns – as we said already, we want to avoid mixing domain or database code with the UI stuff, etc. The actual idea of separating a project into layers suggests that this separation of concerns should be achieved by source code organization. This means that apart from some guidance to what concerns we should separate, the Layered Architecture tells us nothing else about the design and implementation of the project. This implies that we should complement it with some other architectural processes, such as some upfront design, daily design sessions, or even full-blown Domain-Driven Design. Whichever option we choose doesn't matter, at least for the sake of layering, but we need to remember: Layered Architecture gives us nothing apart from a guideline on how to organize the source code.

Layered architecture is the organization of the project structure into four main categories: presentation, application, domain, and infrastructure. Each of the layers contains objects related to the particular concern it represents.

The presentation layer contains all of the classes responsible for presenting the UI to the end-user or sending the response back to the client (in case we're operating deep in the back-end).

- The application layer contains all the logic that is required by the application to meet its functional requirements and, at the same time, is not a part of the domain rules. In most systems that I've worked with, the application layer consisted of services orchestrating the domain objects to fulfill a use case scenario.
- The domain layer represents the underlying domain, mostly consisting of domain entities and, in some cases, services. Business rules, like invariants and algorithms, should all stay in this layer.
- The infrastructure layer (also known as the persistence layer) contains all the classes responsible for doing the technical stuff, like persisting the data in the database, like DAOs, repositories, or whatever else you're using.

There are two important rules for a classical Layered Architecture to be correctly implemented:

- All the dependencies go in one direction, from presentation to infrastructure. (Well, handling persistence and domain are a bit tricky because the infrastructure layer often saves domain objects directly, so it actually knows about the classes in the domain).
- No logic related to one layer's concern should be placed in another layer. For instance, no domain logic or database queries should be done in the UI.

References:

- https://www.capgemini.com/in-en/
- https://en.wikipedia.org/wiki/Capgemini
- https://www.capgemini.com/in-en/careers/
- https://www.javacodegeeks.com/hibernate-tutorials
- https://howtodoinjava.com/spring-mvctutorial/
- https://www.java2novice.com/jdbc/
- https://angular.io/tutorial
- https://www.tutorialspoint.com/angular6/
- https://www.google.com /java/