HAL-VT report

NAME:- TAMMINAINA SAI SOMASEKHAR

# About HAL:-

Hindustan Aeronautics Limited is an Indian state-owned aerospace and defence company, headquartered in Bengaluru, India. Established on 23 December 1940, HAL is one of the oldest and largest aerospace and defence manufacturers in the world today.

# History of HAL:-



The history and growth of Hindustan Aeronautics Limited is synonymous with the growth of Aeronautical industry in India for more than 79 years.

The Company which had its origin as Hindustan Aircraft Limited was incorporated on 23 Dec 1940 at Bangalore by **Shri Walchand Hirachand**, a farsighted visionary, in association with the then Government of Mysore, with the aim of manufacturing aircraft in India. In March 1941, the Government of India became one of the shareholders in the Company and subsequently took over its management in 1942. In collaboration with the Inter Continental Aircraft Company of USA, the Company commenced its business of manufacturing of Harlow Trainer, Curtiss Hawk Fighter and Vultee Bomber Aircraft.

In January 1951, Hindustan Aircraft Limited was placed under the administrative control of Ministry of Defence, Government of India.

# Customers of HAL:-

|  |  |
| --- | --- |
| DEFENCE & SPACE | CIVIL |
| Indian Air Force | Border Security Force |
| Indian Army | Oil & Natural Gas Cooperation of India |
| Indian Navy | Govt. of Karnataka |
| Indian Coast Guard | Govt. of Jharkhand |
| Indian Space Research Organisation | Govt. of Maharshtra |
| Defence Research & Development Organisation | Geological Survey of India |
| Ordnance Factory Board | Bharat Heavy Electricals Ltd. |

# International customers of HAL:-

| EXPORTS:(In Alphabetical Order) |
| --- |
| Airbus Industries, France |
| Boeing, USA |
| Coast Guard, Mauritius |
| Ecuadorian Air Force, Ecuador |
| ELTA, Israel |
| GE Aviation, USA |
| Hamilton Sundstrand, USA |
| Honeywell International, USA |
| Israel Aircraft Industries, Israel |
| Mauritius Police Force, Mauritius |
| Moog Inc. USA |
| Namibian Air Force, Namibia |
| Nepal Army, Nepal |
| RAC MIG, Russia |
| Rolls Royce Plc, UK |
| Royal Air Force, Oman |
| Royal Malaysian Air Force, Malaysia |
| Royal Thai Air Force, Thailand |
| Ruag, Germany |
| Rosoboronexport, Russia |
| Suriname Air Force |
| Turbomeca, France |
| Vietnam Air Force, Vietnam |

# PROJECT at HAL:-

## Title: Movie ticket booking

There is a theatre at HAL and we made front-end pages for booking movie tickets there.

## Pages:-

There are 3 main pages:-

* Home page
* Seat selection page
* Payment details page

# Tools used :-

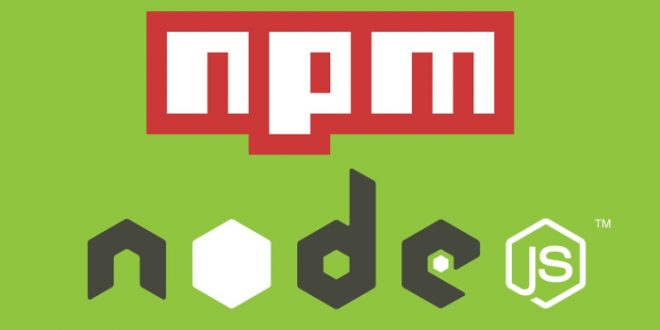
* VS code (Visual Studio Code)
* Npm installer

## VS Code:-



* Visual Studio Code is free, open-source, and cross-platform. This means that it works on Windows, Linux, and macOS.
* Visual Studio Code can be classed as an integrated development environment (IDE), meaning that developers can write and test code at the same time.
* Visual Studio Code has an in-built debugger, making the development flow less ‘clicky’ and maintains a single view with code and debugger.
* You don’t need to have multiple screens to run the different consoles and rearrange them each time you need to minimize something. It’s built into the design and your desired workspace set up.
* In addition to this, there is intelliSense but built into the code editor. intelliSense is a form of predictive coding. With the addition of framework, library, and/or language plugin extensions, you can leverage this even further with ready-made boiler-plates.

## Npm installer:-



* Npm is installed with Node Js.
* You have to install Node.js to get npm installed on your computer.
* All npm packages are defined in files called package.json.
* The content of package.json must be written in JSON.
* At least two fields must be present in the definition file: name and version.
* Npm can manage dependencies.
* Npm can (in one command line) install all the dependencies of a project.
* Dependencies are also defined in package.json**.**

# Languages used:-

* Node Js
* Svelte kit
* Tailwind css

## Svelte:-

Like React and Vue, **Svelte is a JavaScript framework/library**. In general, it's a set of components, tools and rules for creating the structure of websites and applications with the use of JavaScript.

Svelte is a compiler that generates minimal and highly optimized **JavaScript** code from our sources; you'll need a terminal with node + npm installed to compile and build your app.

## Svelte vs React :-

Compared to React, **Svelte is simpler to understand and get started with**, because the major portion of Svelte is plain JavaScript, HTML, and CSS. Svelte also sticks closely to JavaScript's classic web development models, and introduces only a few extensions to HTML, making it much easier to learn.

Here’s a sample of a basic application made with Svelte:

<script>

let a = 1;

let b = 2;

</script>

<input type="number" bind:value={a}>

<input type="number" bind:value={b}>

<p>{a} + {b} = {a + b}</p>

Below is a code snippet from the same application built with React:

import React, { useState } from 'react';

export default () => {

const [a, setA] = useState(1);

const [b, setB] = useState(2);

function handleChangeA(event) {

setA(+event.target.value);

}

function handleChangeB(event) {

setB(+event.target.value);

}

return (

<div>

<input type="number" value={a} onChange={handleChangeA}/>

<input type="number" value={b} onChange={handleChangeB}/>

<p>{a} + {b} = {a + b}</p>

</div>

);

};

Spacing included, Svelte is nine lines of code, while React is 21. That is a significant difference! The beautiful thing about the Svelte app is that it is in no way less functional.

Additionally, with Svelte, you don’t need to worry about forgetting to export components, as they are exported by default, and ready to be imported by other components. With React, you have to do that manually.

Svelte also comes with built-in [effects and animations](https://blog.logrocket.com/essential-transitions-and-animations-in-svelte/). You don’t need to use a third-party library to create animations like you would in React. svelte/motion, svelte/transition, svelte/animate, and svelte/easing provide us with powerful modules to create amazing animations out of the box. Amazingly, these built–in effects won’t increase the size of your app.

With Svelte, you don’t have to worry about writing unique classes, or styles leaking out of their components; styles are component-scoped in their style tags, which allows for flexible styling.

## Tailwind css:-

Tailwind CSS is basically **a utility-first CSS framework for rapidly building custom user interfaces**. It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override.

**Advantages of Tailwind CSS:**

* No more silly names for CSS classes and Id’s.
* Minimum lines of Code in CSS file.
* We can customize the designs to make the components.
* Makes the website responsive.
* Makes the changes in the desired manner.   
  CSS is global in nature and if make changes in the file the property is changed in all the HTML files linked to it. But with the help of Tailwind CSS we can use utility classes and make local changes.

# Directions of use for users:-

Login page:-

* When a user opens the website, the user is redirected to the login page.
* Then the user must click on login as a user and login using his/her login credentials (mobile number/email).
* Upon entering the correct login credentials and clicking on the login button, the user gets redirected to the home page.

Home page:-

* In home page, there are cards of the movies scheduled to play in the theatre.
* Under each card of the movie, movie name, genre, time, etc is given.
* The user can select the name of the movie with the correct time slot he wanted to attend.
* Upon clicking the card, the user gets redirected to the seat selection page.

Seat selection page:-

* In seat layout page, the user can find all the rows in the theatre.
* There different categories of rows.
* Each category has different cost and a name.
* User can select the seats which he wanted, provided those were not booked before.
* The seat colors:-
  + Black color:- Seats available for booking
  + Light grey color:- Seats that are already booked
  + Green:- Seats that are selected by the user
* The user must select at least one seat to go to payment details page.
* The confirm button only appears when at least one seat is selected, otherwise, “select at least one seat message” is shown.
* When the user clicks on the confirm button, he/she gets redirected to payment details page.

Payment details page:-

* There are two sections in the payment details page.
* Ticket details:-
  + The name of the movie
  + Number of seats booked
  + Names of the booked seats
  + Time slot
  + Contact details (to be entered by user):-
    - Mobile number
    - Email
* The ticket details will be sent to the contact details given above.
* If the user does not give any contact details, then the ticket details will be sent to the default contact details, i.e., login credential mobile number/email.
* Payment split:- The following things are shown in payment split
  + Cost of the seats
    - Total cost of all seats from all categories
  + Convenient fee
    - Fixed service charge for each seat
  + IGST
    - Fixed percentage
  + Promo code
    - Upon entering the right promo code, the user gets instant x% discount
  + Donation Rs y
    - Upon clicking checkbox, an addition of y RS will be added to the grand total for donation purpose
  + Grand Total
    - Total amount to be paid after adding IGST, convenient fee, donation to the seats cost and removing the discount.
* Then there are two buttons
  + Continue
    - To go to payment gateway page
  + Cancel
    - To go back to seat selection page

# Admin Rights:-

* The admin must first log in on log in page by clicking on log in as an admin and then enter his credentials and login.
* He can add or delete a movie and its time slots from both the front end and backend.
* Can change seat layout.
* Can change the rates of each category of seats.
* He can change the promo code.
* He can change the discount percentage.
* He can change the IGST percentage.
* He can change the cause of the donation and its value.

# Printing the details of the movie:-

* The details of the movie are stored in Json format.
* Every movie is stored in an array.
* Each movie is given some properties:-
  + Movie name
    - Name of the movie
  + Genre
    - Genre of the movie
  + Slot
    - Slot of the movie in which it is playing
* These movie details are then fetched from back-end and printed in a for loop.

# Printing the seat layout:-

* The details of each seat are stored in Json format.
* First the array stores all the categories.
* Then in each category, the rows of that particular category are stored.
* Then in each row, the seats of that row are stored
* In each seat, the properties of that seat are stored.
* The properties are:-
  + Name
    - Name of the seat.
  + Ghost
    - Weather the seat is present or not.
  + Booked
    - Weather that particular seat is already booked or not.
  + Selected
    - Weather the seat is selected by the user or not.
* Then we fetch these details and print them in the for loop.

# Payment page details:-

* The promo code is stored in the backend and is fetched.
* The discount, IGST percentage, donation amount and cause are also fetched from the backend.
* The details of the movie, booked seats, etc, are sent as parameters in the URL and then fetched.

# The Files:-

# Functions Implemented:-

# Conclusion:-