

# Taxi-Booking Service

## app-routes.ts

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

## Index.html

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

## App

### App.component.ts

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

### App.component.html

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

## App.component.css

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

Home

## Home.component.ts

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

## Home.component.html

```
<div class="row">
  <div class="col-md-4" style="margin-top: 300px; margin-left: 50px;">
    <h1 class="con">TRAVEL WITH TRUST</h1>
    <p>Book affordable and comfortable rides</p>
    <button type="button">Booking</button>
  </div>

  <div class="col-md-7">
    <div class="media-object">
      
    </div>
  </div>
</div>
```

## Home.component.css

```
img{
  margin-top: 100px;
}
```

## Booking

### Booking.component.ts

```
import { CommonModule } from '@angular/common';
import { HttpClient, HttpClientModule } from '@angular/common/http';
import { Component, OnInit } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { Router, RouterLink, RouterModule } from '@angular/router';

declare var $: any;

@Component({
  selector: 'app-booking',
  standalone: true,
  imports: [CommonModule, RouterModule, RouterLink, HttpClientModule, ],
  templateUrl: './booking.component.html',
  styleUrls: ['./booking.component.css']
})
```

```

})
export class BookingComponent implements OnInit{
  drivers: Driver[] = [];
  selectedDriver: any;

  constructor(private http: HttpClient,private router: Router) {}

  ngOnInit(): void {
    this.getDrivers().subscribe((response) => {
      this.drivers = response;
    });
  }

  getDrivers(){
    return this.http.get<Driver[]>('http://localhost:3000/nearbyDrivers');
  }

  openpopup() {
    $('#exampleModelCenter').modal('show');
  }

  details(driver: any) {

    this.selectedDriver = driver;
  }

  booking(driver: any) {
    if (this.selectedDriver) {
      // Navigate to the booking details page with the selected driver details
      this.router.navigate(['/app-booking-details'], { queryParams:
this.selectedDriver });
    } else {
      console.log("No driver selected for booking.");
    }
  }
}
// Define the Driver class outside the component
class Driver {
  driverName!: string
  vehicleModel!:string
  arrivalTime!:string
  duration!:string
  driverRating!:string
  fare!:string
}

```

## Booking.component.html

```
<!-- app.component.html -->

<router-outlet></router-outlet>

<div class="container">
  <h1 style="text-align: center;">Cab Booking</h1>

  <div class="card-container" >
    <div class="card" style="height:100px">
      <div class="row">
        <div class="col-6">
          <label>From</label><br>
          <select style="width: 300px" >
            <option>Delhi</option>
            <option>Pune</option>
            <option>Goa</option>
            <option>Gurugram</option>
            <option>Karnataka</option>
            <option>Odisha</option>
          </select>
        </div>

        <div class="col-6">
          <label>To</label><br>
          <select style="width: 300px">
            <option>Delhi</option>
            <option>Pune</option>
            <option>Goa</option>
            <option>Gurugram</option>
            <option>Karnataka</option>
            <option>Odisha</option>
          </select>
        </div>
      </div>
    </div>

    <br>

    <div class="card">
      <div class="row">

        <div class="col-12">
          <div class="card" *ngFor="let driver of drivers" >
            <div class="row" *ngIf="drivers.length > 0">
              <div class="col-2" >
```

```

        
    </div>
    <div class="col-8">
        <ul>
            <li>
                <strong>{{ driver.driverName }}</strong>
                <ul>
                    <li><strong>Vehicle Model:</strong> {{
driver.vehicleModel }}</li>
                    <li><strong>Arrival Time:</strong> {{
driver.arrivalTime }}</li>
                    <li><strong>Duration:</strong> {{
driver.duration }}</li>
                    <li><strong>Rating:</strong> {{
driver.driverRating }}</li>
                    <li class=""><strong>Fare:</strong>
&#x20B9;
                        {{ driver.fare }}</li>
                </ul>
            </li>
        </ul>
    </div>
    <div class="col-2" style="margin-top: 50px;">
        <button type="button" class="btn btn-primary "
(click)=details(driver) data-toggle="modal" data-target="#exampleModal">
            Details
        </button>
    </div>

</div>
<p *ngIf="drivers.length === 0">No drivers available</p>
</div>
<br>
</div>
</div>
</div>

<!-- Modal -->
<div class="modal fade" id="exampleModal" tabindex="-1" aria-
labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Driver
details</h5>

```

```

        <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
            <span aria-hidden="true">&times;</span>
        </button>
    </div>
    <div class="modal-body">
        <div class="col-2">
            
        </div>
        <div class="col-8">
            <ul>
                <li>
                    <strong>{{ selectedDriver?.driverName }}</strong>
                    <ul>
                        <li><strong>Vehicle Model:</strong> {{
selectedDriver?.vehicleModel }}</li>
                        <li><strong>Arrival Time:</strong> {{
selectedDriver?.arrivalTime }}</li>
                        <li><strong>Duration:</strong> {{
selectedDriver?.duration }}</li>
                        <li><strong>Rating:</strong> {{
selectedDriver?.driverRating }}</li>
                        <li class=""><strong>Fare:</strong> &#x20B9;
{{ selectedDriver?.fare }}</li>
                    </ul>
                </li>
            </ul>
        </div>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-primary"
(click)="booking(drivers)">Booknow</button>
    </div>
</div>
</div>
</div>

```

## Booking.component.css

```

.container{
    margin-top: 150px;
}

```



## Booking-details

### Booking-details.component.ts

```
import { FormsModule } from '@angular/forms';

import { Component, OnInit } from '@angular/core';
import { CommonModule } from '@angular/common';
import { ActivatedRoute, Router, RouterModule } from '@angular/router';
import { MatSnackBar } from '@angular/material/snack-bar';

interface Driver {
  driverName?: string;
  vehicleModel?: string;
  arrivalTime?: string;
  duration?: string;
  driverRating?: string;
  fare?: string;
  to?: string;
  from?: string;
}

@Component({
  selector: 'app-booking-details',
  standalone: true,
  imports: [FormsModule, CommonModule, RouterModule],
  templateUrl: './booking-details.component.html',
  styleUrls: ['./booking-details.component.css'] // Use styleUrls instead of
styleUrl
})
export class BookingDetailsComponent implements OnInit {
  selectedDriver: Driver = {};
  bookingSuccessful = false;

  constructor(private route: ActivatedRoute, private router: Router, private
snackBar: MatSnackBar) {}

  ngOnInit(): void {
    this.route.queryParams.subscribe((params) => {
      this.selectedDriver = params as Driver;
      this.showBookingSuccessModal();
    });
  }

  backHome() {
    this.router.navigate([''], { state: { bookingSuccessful: true } });
  }
}
```

```

showBookingSuccessModal() {
  this.bookingSuccessful = true;
}
}

```

## Booking-details.component.html

```

<!-- booking-details.component.html -->
<div class="container" style="margin-left: 500px;">
<p class="top">Happy Journey</p>
<div class="row">
  <div class="col-3"><strong>From</strong>:{{selectedDriver.from}}</div>

  <div class="col-6"><strong>To</strong>:{{selectedDriver.to}}</div>
</div>
<div class="card" style="text-align: center;">
<h3 style="margin-right: 205px;">Driver Details</h3>
<div class="col-8">
  <ul>
    <li>

      <strong>{{ selectedDriver.driverName }}</strong>

      <ul>
        <li><strong>Vehicle Model:</strong> {{ selectedDriver.vehicleModel
}}</li>
        <li><strong>Arrival Time:</strong> {{ selectedDriver.arrivalTime
}}</li>
        <li><strong>Duration:</strong> {{ selectedDriver.duration }}</li>
        <li><strong>Rating:</strong> {{ selectedDriver.driverRating }}</li>
        <li class=""><strong>Fare:</strong> &#x20B9; {{ selectedDriver.fare
}}</li>
      </ul>
    </li>
  </ul>
  <button (click)="backHome()">go to home</button>
</div>
</div>
<!-- booking-details.component.html -->

<!-- ... your existing content ... -->
<div *ngIf="bookingSuccessful" class="booking-success-snackbar">
  <i class="fas fa-check-square"></i> Booking Successful
</div>

```

## Booking-details.component.css

```
.top{
  margin-top: 100px;
  font-size: 65px;
  font-weight: bold;
}
.card{
  width:900px ;
  height:300px ;
  text-align: center;
  margin-left: 250px;
}
ul{
list-style-type: none;
}
button{
  background: bisque;
}
/* booking-success-modal.component.css */

.modal {
  display: none;
  position: fixed;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background-color: rgba(0, 0, 0, 0.5);
}

.modal-content {
  background-color: #fff;
  padding: 20px;
  margin: 10% auto;
  width: 50%;
  text-align: center;
}
/* Add this to your styles.css or styles.scss file */

.booking-success-snackbar {
  position: fixed;
  top: 80px; /* Adjust as needed */
  right: 20px; /* Adjust as needed */
  background-color: #4caf50; /* Green */
  color: white;
  padding: 15px;
  border-radius: 5px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}
```

```
z-index: 1;
animation-duration: 3000;
}
```

---

## Signup

### signup.component.ts

```
import { HttpClient, HttpClientModule } from '@angular/common/http';
import { Component } from '@angular/core';
import { FormsModule, ReactiveFormsModule, FormGroup, FormControl, Validators
} from '@angular/forms';

@Component({
  selector: 'app-signup',
  standalone: true,
  imports: [FormsModule, ReactiveFormsModule, HttpClientModule],
  templateUrl: './signup.component.html',
  styleUrls: ['./signup.component.css'] // Corrected property name
})
export class SignupComponent {
  constructor(private http: HttpClient) {}

  users: user[] = [];

  controlForm = new FormGroup({
    name: new FormControl('', [Validators.minLength(10)]), // Added initial
value ('')
    age: new FormControl(''),
    username: new FormControl(''), // Added initial value ('')
    password: new FormControl(''), // Added initial value ('')
    address: new FormControl(''),
    referralCode: new FormControl(''),
    isCabRider: new FormControl('')
  });

  addUser() {
    return this.http.post<user>('http://localhost:3000/user', {
      name: this.controlForm.controls.name.value,
      age: this.controlForm.controls.age.value,
      username: this.controlForm.controls.username.value,
```

```

        password: this.controlForm.controls.password.value,
        address: this.controlForm.controls.address.value,
        referralCode: this.controlForm.controls.referralCode.value,
        isCabRider: this.controlForm.controls.isCabRider.value,
    });
}
}

class user {
    name!: string;
    age!: string;
    username!: string;
    password!: string;
    address!: string;
    referralCode!: string;
    isCabRider!: string;
}

```

## signup.component.html

```

<p style="margin-top: 150px;"></p>

<div class="card">
    <div class="container">
        
        <h3 style="text-align: center;">Register</h3>
        <form>
            <form [formGroup]="controlForm">
                <div class="row">
                    <div class="col-6">
                        <label for="firstname">Name</label>
                        <input type="text" id="firstname" name="firstname"
formControlName="firstname" required="">
                    </div>
                    <div class="col-6">
                        <label for="age">age</label>
                        <input type="text" id="age" name="age" formControlName="age"
required="">
                    </div>
                </div>
                <br>
                <div class="row">
                    <div class="col-6">
                        <p><label for="username">username</label>

```

```

<input type="text" id="username" name="username" formControlName="username"
required="">
</div>
<div class="col-6">
<p><label for="password">password</label>
    <input type="password" id="password" name="password"
formControlName="password" required="">

</div>

</div>

<div class="row">
    <div class="col-12">
<p><label for="address">address</label>
<input type="text" id="address" name="address" formControlName="address"
required="" style="width: 350px;">
</div>

</div>
<div class="row">
    <div class="col-6">
<label for="referralcode">referralcode</label>
<input type="text" id="referralcode" name="referralcode"
formControlName="referralcode" required="">
</div>
<div class="col-5">
<label for="cabride">cabride</label>
<input type="text" id="cabride" name="cabride" formControlName="cabride"
required="">
</div>
</div>
<br>
<button class="btn btn-primary" style="width: 450px;margin-left:
65px;"(click)="addUser()">Register</button>
    </form>
</form>

```

## signup.component.css

```

.card{
    width: 600px;
    height:550px;

```

```
    align-content: center;
    margin-left: 500px;
}
label{
    font-weight: bold;
}
```

---

## Signin

### signin.component.ts

```
import { HttpClient, HttpClientModule } from '@angular/common/http';
import { Component } from '@angular/core';
import { FormsModule, ReactiveFormsModule, FormGroup, FormControl, Validators
} from '@angular/forms';

@Component({
  selector: 'app-signup',
  standalone: true,
  imports: [FormsModule, ReactiveFormsModule, HttpClientModule],
  templateUrl: './signup.component.html',
  styleUrls: ['./signup.component.css'] // Corrected property name
})
export class SignupComponent {
  constructor(private http: HttpClient) {}

  users: user[] = [];

  controlForm = new FormGroup({
    name: new FormControl('', [Validators.minLength(10)]), // Added initial
value ('')
    age: new FormControl(''),
    username: new FormControl(''), // Added initial value ('')
    password: new FormControl(''), // Added initial value ('')
    address: new FormControl(''),
    referralCode: new FormControl(''),
    isCabRider: new FormControl('')
  });

  addUser() {
    return this.http.post<user>('http://localhost:3000/user', {
      name: this.controlForm.controls.name.value,
      age: this.controlForm.controls.age.value,
```

```

        username: this.controlForm.controls.username.value,
        password: this.controlForm.controls.password.value,
        address: this.controlForm.controls.address.value,
        referralCode: this.controlForm.controls.referralCode.value,
        isCabRider: this.controlForm.controls.isCabRider.value,
    });
    }
}

class user {
    name!: string;
    age!: string;
    username!: string;
    password!: string;
    address!: string;
    referralCode!: string;
    isCabRider!: string;
}

```

## signin.component.html

```

<p style="margin-top: 150px;"></p>

<div class="card">
    <div class="container">
        
        <h3 style="text-align: center;">Register</h3>
        <form>
            <form [formGroup]="controlForm">
                <div class="row">
                    <div class="col-6">
                        <label for="firstname">Name</label>
                        <input type="text" id="firstname" name="firstname"
formControlName="firstname" required="">
                    </div>
                    <div class="col-6">
                        <label for="age">age</label>
                        <input type="text" id="age" name="age" formControlName="age"
required="">
                    </div>
                </div>
                <br>
                <div class="row">
                    <div class="col-6">

```



```

    <p><label for="username">username</label>
<input type="text" id="username" name="username" formControlName="username"
required="">
</div>
<div class="col-6">
<p><label for="password">password</label>
    <input type="password" id="password" name="password"
formControlName="password" required="">

</div>

</div>

<div class="row">
    <div class="col-12">
<p><label for="address">address</label>
<input type="text" id="address" name="address" formControlName="address"
required="" style="width: 350px;">
</div>

</div>
<div class="row">
    <div class="col-6">
<label for="referralcode">referralcode</label>
<input type="text" id="referralcode" name="referralcode"
formControlName="referralcode" required="">
</div>
<div class="col-5">
<label for="cabride">cabride</label>
<input type="text" id="cabride" name="cabride" formControlName="cabride"
required="">
</div>
</div>
<br>
<button class="btn btn-primary" style="width: 450px;margin-left:
65px;"(click)="addUser()">Register</button>
    </form>
</form>

```

signin.component.css

```

.card{
    width: 600px;
    height:550px;
    align-content: center;
    margin-left: 500px;
}

```

```
label{
    font-weight: bold;
}
```

## Db.json

```
{
  "user": [
    {
      "name": "Steve",
      "age": "21",
      "username": "steve123",
      "password": "qwerty123",
      "address": "Street 12, New Delhi",
      "referralCode": "R123",
      "isCabRider": "Yes",
      "panCard": "GUE377IEI0",
      "rc": "R123",
      "dl": "DL59839",
      "vehicle": "Toyota Innova"
    }
  ],

  "nearbyDrivers": [{
    "driverName": "Jeremy",
    "vehicleNumber": "DL-CS7278",
    "driverRating": "4",
    "vehicleModel": "Polo",
    "arrivalTime": "9:32 am",
    "duration": "43 mins",
    "fare": "345",
    "from": "Delhi",
    "to": "Gurugram"
  },
  {
    "driverName": "Martha",
    "vehicleNumber": "DL-CL6828",
    "driverRating": "5",
    "vehicleModel": "Swift",
    "arrivalTime": "9:35 am",
    "duration": "44 mins",
    "fare": "350",
    "from": "Gurugram",
```

```
    "to": "Noida"
  },
],
"booking": [
  {
    "driverName": "Martha",
    "from": "Gurugram",
    "to": "Noida",
    "vehicle": "Toyota Innova",
    "vehicleNumber": "DL-CL6828",
    "driverRating": "5",
    "vehicleModel": "Swift",
    "arrivalTime": "9:35 am",
    "duration": "44 mins",
    "fare": "350"
  }
]
```