

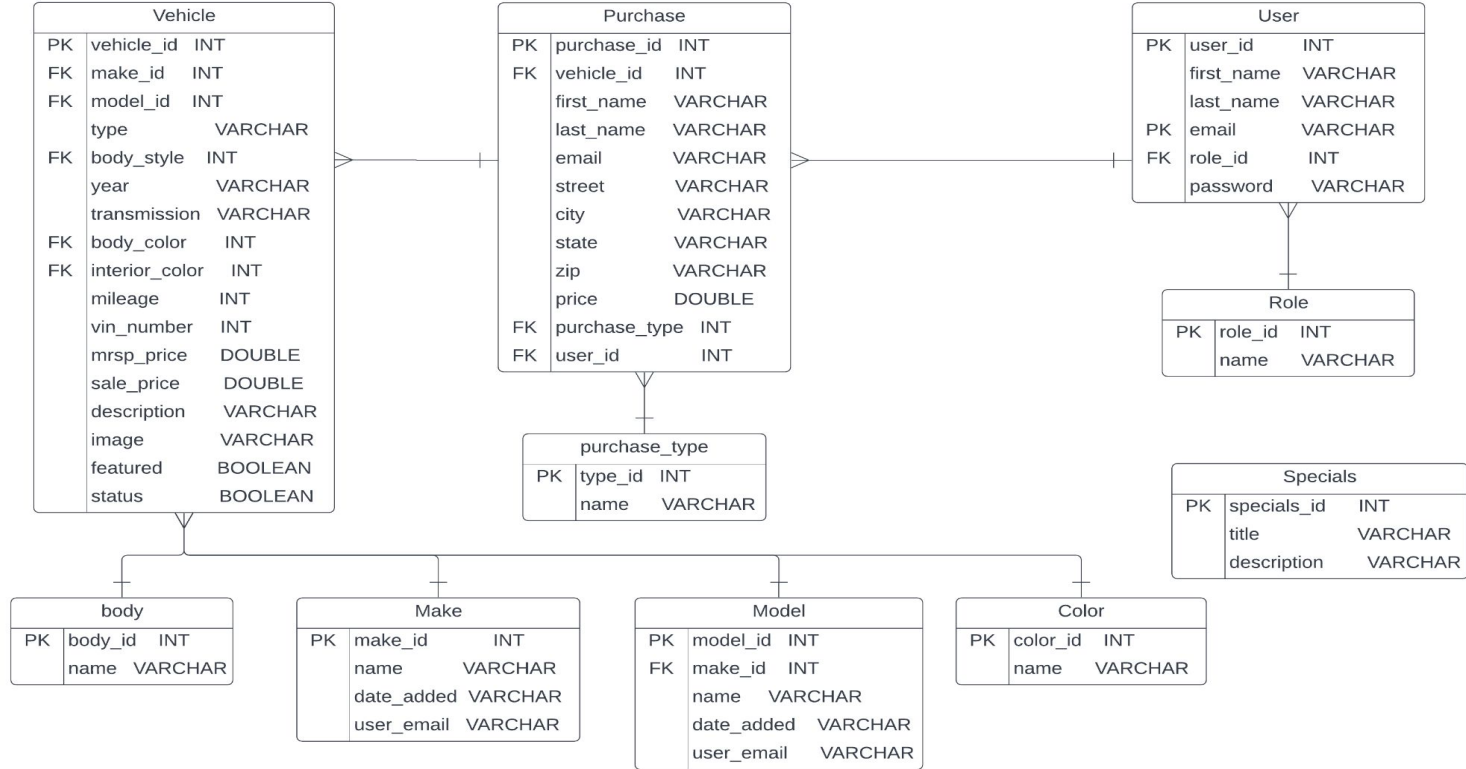


Car Dealership

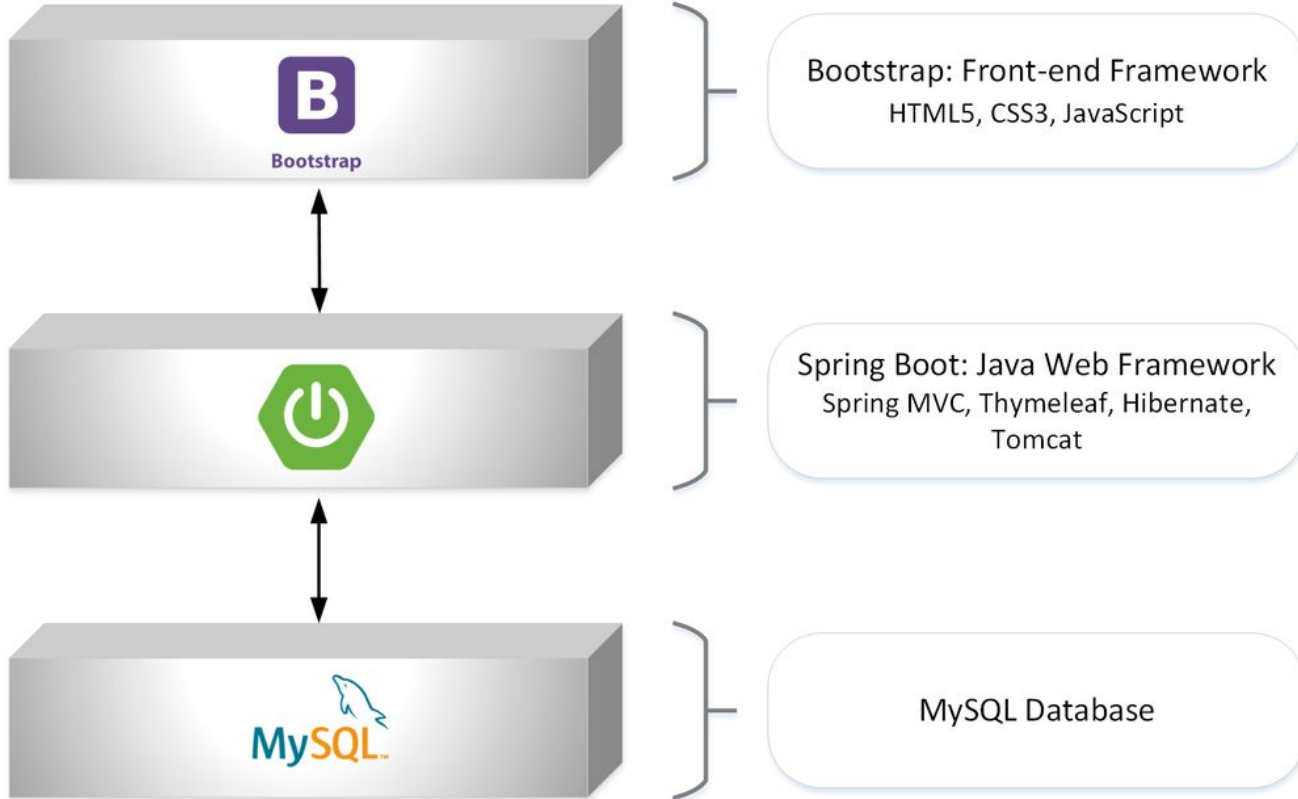
Capstone Project



Entity Relationship Diagram



Tech-Stack



Tables

- Vehicle
- Role
- User
- Make
- Model
- Color
- Body
- Purchase_type
- Purchases
- Special

Views:

- Sales
- Inventory

Database Scripts

```
DROP DATABASE IF EXISTS dealership;  
CREATE DATABASE dealership;  
USE dealership;
```

```
DROP TABLE IF EXISTS vehicle;  
CREATE TABLE vehicle (  
vehicle_id int primary key auto_increment,  
make int,  
model int,  
type varchar(10),  
body int,  
year varchar(10),  
transmission varchar(15),  
body_color int,  
interior_color int,  
mileage int,  
vin_number varchar(20),  
mrsp_price int,  
sale_price int,  
description varchar(500),  
image varchar(50),  
featured boolean,  
sold boolean,  
user_id int,  
foreign key (user_id) references user(user_id),  
foreign key (make) references make(make_id),  
foreign key (model) references model(model_id),  
foreign key (body) references body(body_id),  
foreign key (body_color) references color(color_id),  
foreign key (interior_color) references color(color_id)  
);
```

Database Scripts

DROP TABLE IF EXISTS role;

CREATE TABLE role (

role_id int primary key auto_increment,
name varchar(10)

);

DROP TABLE IF EXISTS user;

CREATE TABLE user (

user_id int primary key auto_increment,
first_name varchar(50),
last_name varchar(50),
email varchar(50) NOT NULL,
role int NOT NULL,
password varchar(50) NOT NULL,
foreign key (role) references role(role_id)

);

DROP TABLE IF EXISTS make;

CREATE TABLE make (

make_id int primary key auto_increment,
name varchar(50),
date varchar(10),
user_email varchar(50)

);

DROP TABLE IF EXISTS model;

CREATE TABLE model (

model_id int primary key auto_increment,
make_id int,
name varchar(20),
date varchar(10),
user_email varchar(50),
foreign key (make_id) references make(make_id)

);

DROP TABLE IF EXISTS color;

CREATE TABLE color (

color_id int primary key auto_increment,
name varchar(15)

);

DROP TABLE IF EXISTS body;

CREATE TABLE body (

body_id int primary key auto_increment,
name varchar(50)

);

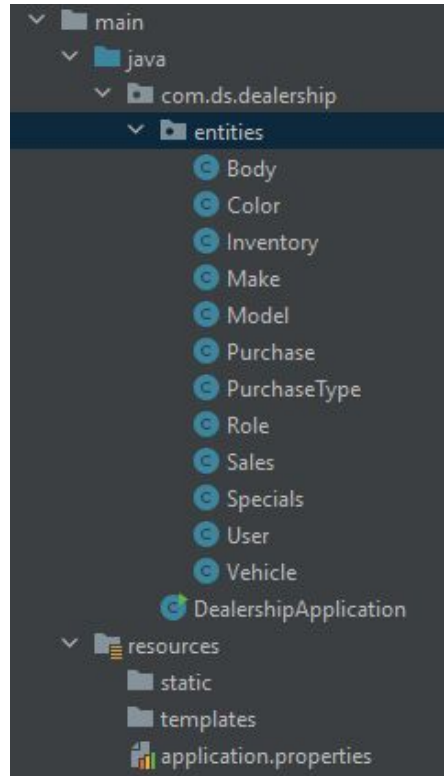
Database Scripts

```
DROP TABLE IF EXISTS purchase_type;  
CREATE TABLE purchase_type (  
  type_id int primary key auto_increment,  
  name varchar(50)  
);
```

```
DROP TABLE IF EXISTS special;  
CREATE TABLE special (  
  special_id int primary key auto_increment,  
  title varchar(50),  
  description varchar(500)  
);
```

```
DROP TABLE IF EXISTS purchase;  
CREATE TABLE purchase (  
  purchase_id int primary key auto_increment,  
  vehicle_id int,  
  first_name varchar(50),  
  last_name varchar(50),  
  email varchar(50),  
  street varchar(50),  
  city varchar(50),  
  state varchar(50),  
  zip varchar(5),  
  price int,  
  purchase_type int,  
  foreign key (vehicle_id) references vehicle(vehicle_id),  
  foreign key (purchase_type) references purchase_type(type_id)  
);
```

DAOs



JPA Annotations

```
@Entity
public class Inventory {
    @GeneratedValue (strategy = GenerationType.IDENTITY)
    @Id
    @Column(name = "inventory_id")
    private int id;
    @Column
    private String year;
    @Column
    private int count;
    @Column (name = "stock_value")
    private double stockValue;
    @Column
    private String type;
    @ManyToOne
    @JoinColumn(name = "make_id")
    private Make make;
    @ManyToOne
    @JoinColumn(name = "model_id")
    private Model model;
```

```
@Entity
public class Purchase {
    @GeneratedValue (strategy = GenerationType.IDENTITY)
    @Id
    @Column(name = "purchase_id")
    private int id;
    @Column(name = "first_name")
    private String firstName;
    @Column(name = "last_name")
    private String lastName;
    @Column
    private String email;
    @Column
    private String street;
    @Column
    private String city;
    @Column
    private String state;
    @Column
    private String zip;
    @Column
    private int price;

    @OneToMany
    @JoinColumn(name = "vehicle_id")
    private Vehicle vehicleId;
    @ManyToOne
    @JoinColumn(name = "purchase_type")
    private PurchaseType purchaseType;
```

JPA Annotations

```
@ManyToOne
@JoinColumn(name = "body_color")
private Color bodyColor;

@ManyToOne
@JoinColumn(name = "interior_color")
private Color interiorColor;

@ManyToOne
@JoinColumn(name = "body")
private Body body;

@ManyToOne
@JoinColumn(name = "make")
private Make make;

@ManyToOne
@JoinColumn(name = "model")
private Model model;
```

```
@Entity
public class Vehicle {

    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Id
    @Column(name = "vehicle_id")
    private int id;

    @Column
    private String type;

    @Column
    private String year;

    @Column
    private String transmission;

    @Column
    private int mileage;

    @Column(name = "vin_number")
    private String vinNumber;

    @Column(name = "mrsp_price")
    private int mrspPrice;

    @Column(name = "sale_price")
    private int salePrice;

    @Column
    private String description;

    @Column
    private String image;

    @Column
    private boolean featured;

    @Column
    private boolean sold;
```

JPA Annotations

```
@Entity
public class User {

    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Id
    private int id;
    @Column(name = "first_name")
    private String firstName;
    @Column(name = "last_name")
    private String lastName;
    @Column(nullable = false)
    private String email;
    @Column(nullable = false)
    private String password;

    @ManyToOne
    @JoinColumn(name="role_id", nullable = false)
    private Role role;
```

Repositories

- Repositories to handle Custom Queries

Controllers

- REST Controllers To handle REST API calls
- Group Controllers for specific DAOs

J-Unit Testing

- Test Daos
- Test Service Layer



Spring Security

- User Access Permissions
 - Admin
 - Sales
 - Disabled



Spring Security