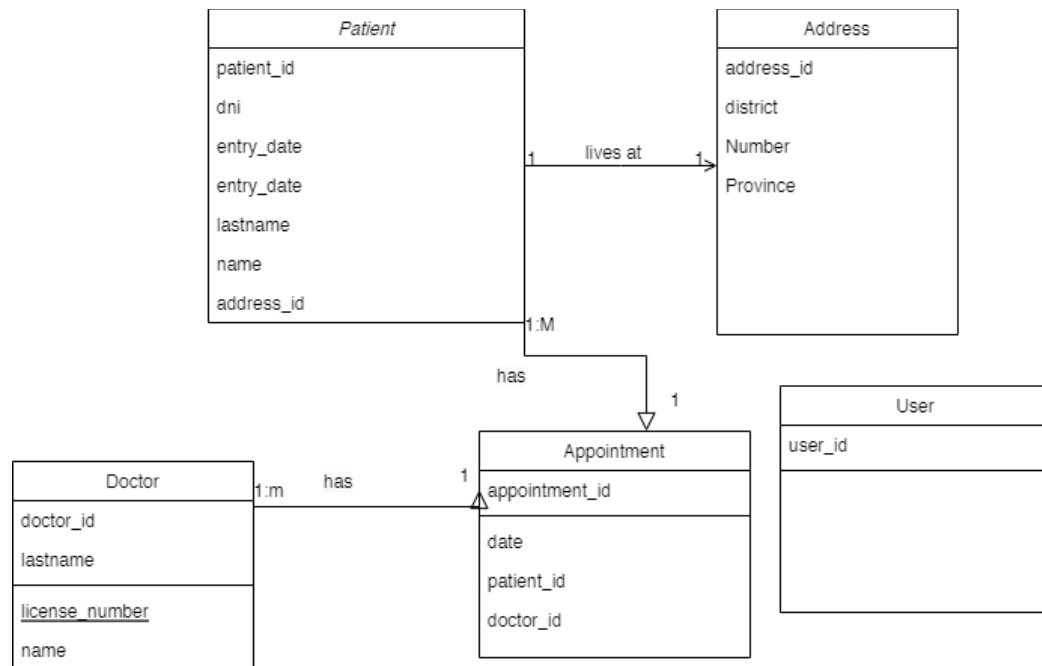


The Appointment Booking System

Class Diagram



Spring Boot Key Concepts Used.

- **Controller:** The Spring Boot, the controller class is **responsible for processing incoming REST API requests, preparing a model, and returning the view to be rendered as a response.** The controller classes in Spring are annotated either by the `@Controller` or the `@RestController` annotation.
- **Repository:** This is responsible to perform various operations on the object using the annotation `@Repository` from spring framework.
- **Service:** This components is the class file which contains `@Service` annotation. These class files are used to write business logic in a different layer.

```
public interface IAddressService { }
```

The OOP Concepts used

Inheritance

The user class extends the DateAduit class and thus able to inherit all the date values

Abstraction:

Abstraction is an interface and therefore all the repositories of the system created using the interface as it describes the objects of the model class.

Polymorphism:

The ability of an object to behave differently in different situations. For example I used getAll() method to find or return all the data about the appointment in the appointmentImplService.

Encapsulation:

In the doctor model, there is the use of encapsulation as show below where the name is hidden and made private of type string.

```
private String name;
```

The design pattern used:

Observer design pattern

For example in the doctor model where there is one-To-Many relationship between the doctor and the patient.

Code Refactoring:

In the doctor model, I changed the integer type of the license_number to use string data type.