Controller

* LoanController

Service

* LoanService

Model

* User

Repository

* UserRepository

**Model**

Fields:

private String middlename

private String lastname

private Date dob (doubt in the data type for date)

private String martial status

private String ssnNumber

private long loanAmount

private String loanPurpose

private String description

private String addressLine1

private String addressLine2

private String city

private String state

private String postalCode

private String homePhone

private String officePhone

private String mobile

private Sting email

private String employerName

private long annualSalary

private int workExpMonth

private int workExpYear

private String designation

private String eaddressLine1

private String eaddressLine2

private String ecity

private String estate

private String epostalCode

private String status (By default=”In progress”)

private Date submittedAt (it will be system time and the format will be timestamp)

getters and setters

constructor

**Controller (LoanRestController)**

**Method to display all applications when clicked on View On Application**

@GetMapping(“/all-app”)

public Iterable<User> showAllApp(){ return service.getAllApp()}

**Method to display a particular application by using Application ID**

@GetMapping(“/get-app/{id}”)

public User getById(@Path………){return service.getAppById(int id)}

**Method to register a particular application**

@PostMapping(“/save-app”)

@Transactional

public void saveApp(User user){ service.saveApplication(user)}

**Repository (UserRepository)**

* Will extend CrudRepository
* We will add update custom queries

**Service (LoanService)**

**Method display all applications**

public Iterable<User> getAllApp(){ return repo.findAll()}

**Method to get application by id**

Public User getAppById(int id){

return repo.findById(id).get();

}

**Method to save application**

public void saveApplication(User user){

repo.save(user) By default status is in progress

Check for Front End Decline Rules

If(checkFrontEndRules(user)){ call python model }

else{ update the status to declined}

}

public boolean checkFrontEndRules(User user){}