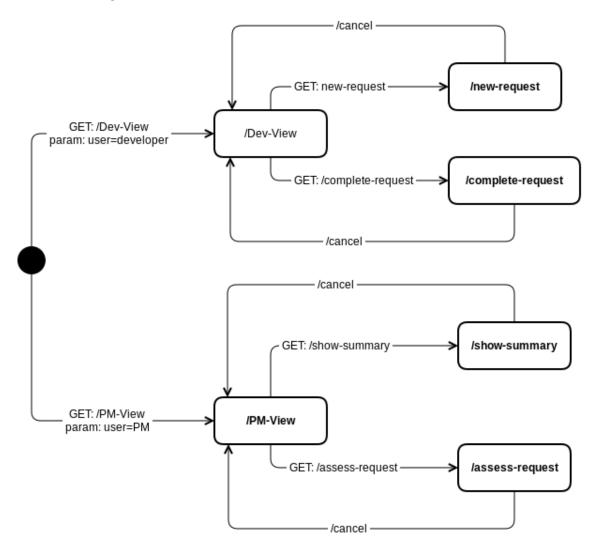
CO2006

# MINI PROJECT

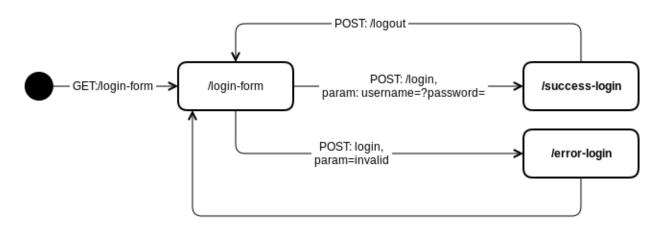
Ihtasham Chaudhry

# Question 1) a)

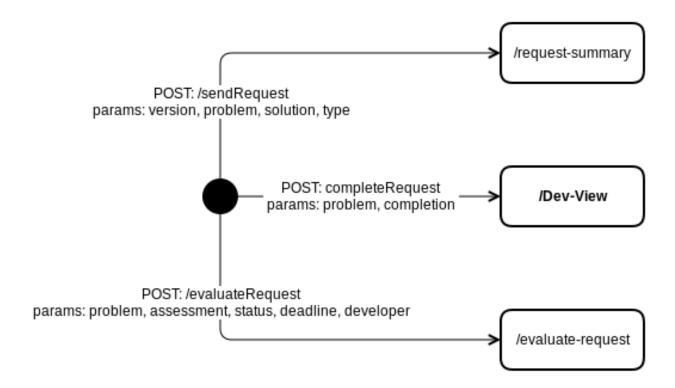
#### **Index Controller Navigational Model:**



#### **Authorization Controller Navigational Model**

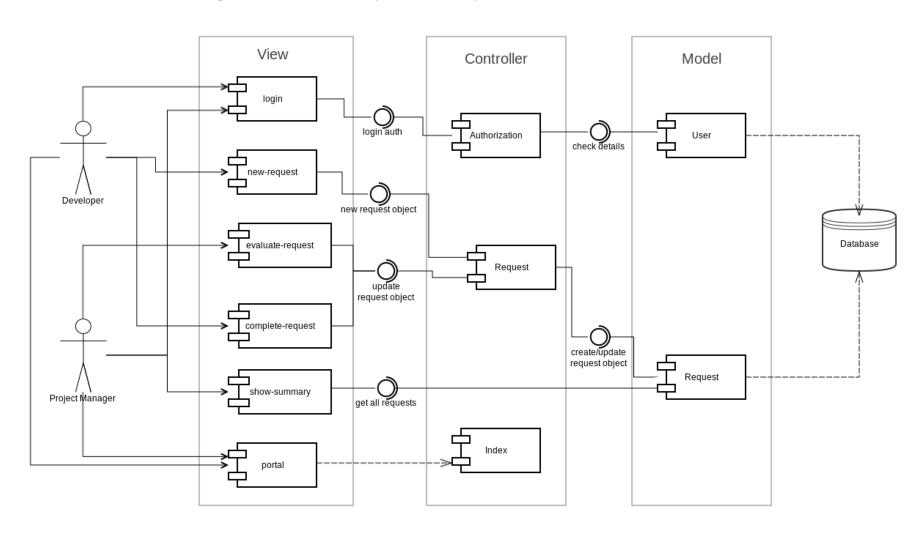


#### **Request Controller Navigational Model**



# Question 1) b)

The software architectural style I will use in my web application will be Spring MVC, thus the layers I will use are: Model, View And Controller and with the addition of using a SQL database where my models will be persisted.



## Question 2) a)

<<Java Interface>> <<Java Class>> <<Java Interface>> <<Java Class>> RoleRepository AuthorizationController SCIRepository **⊙** RequestController **⊙** IndexController bAppProject.persistence.repository webAppProject.controller AppProject.persistence.repository webAppProject.controller webAppProject.controller userRole: String findByld(int):Role findByld(int):SCI √IndexController() loginForm():ModelAndView findByName(String):SCI addRequest(String,String,String,String):ModelAndView invalidLogin():ModelAndView evaluateRequest(String,String,String,String,String,String):String index(HttpSession,HttpServletRequest,ModelMap):ModelAndView successLogin():ModelAndView completeRequest(String,String):String summary():ModelAndView logout():ModelAndView evaluate():ModelAndView error():String complete():ModelAndView newRequest():ModelAndView <<Java Class>> webAppProject.domain a id: Integer status: String problem: String solution: String reqType: String ~realRepositor deadline: String priority: String ersionRepository , ~requestRepository ~versRepository 10..1 -Role ~userRepository\ 0..1 developer: String assessment: String <<Java Interface>> <<Java Class>> completion: String VersionRepository RequestRepository UserRepository √Request() webAppProject.domain ebAppProject.persistence.repositor, webAppProject.persistence.repository bAppProject.persistence.repository setId(int):void id: Integer o findByld(int):Version a findByld(int):Request findByLogin(String):User getid():Integer role: String findByVersion(String):Version findByStatus(String):List<Request> getSolution():String findByDeveloper(String):List<Request> setSolution(String):void a findBvProblem(String):Request getld():Integer getProblem():String findByreqType(String):Request setId(Integer):void setProblem(String):void <<Java Class>> getRole():String getReqType():String setRole(String):void ∀ersion setReqType(String):void getUserRoles():Set<User> webAppProject.domain getCompletion():String setUserRoles(Set<User>):void a id: Integer setCompletion(String):void version: String getVersion():Version -role / 0. description: String setVersion(Version):void √Version() getStatus():String setStatus(String):void getld():Integer setId(Integer):void getDeadline():String setDeadline(String):void getVersion():String getPriority():String setVersion(String):void getDescription():String setPriority(String):void -userRoles 0..\* getDeveloper():String setDescription(String):void <<Java Class>> setDeveloper(String):void getsci():SCI ⊕ User setsci(SCI):void getAssessment():String webAppProject.domain toString():String setAssessment(String):void id: Integer login: String password: String enabled: boolean <<Java Class>> √User() webAppProject.domain getld():Integer a id: Integer setId(Integer):void name: String getLogin():String setLogin(String):void ofSCI() getPassword():String getid():Integer setPassword(String):void setId(Integer):void getRole():Role setName(String):void setRole(Role):void getName():String isEnabled():boolean setEnabled(boolean):void

### Question 2) b)

The MySQL script for the tables that will be used in the web application:

```
DROP TABLE IF EXISTS users;
DROP TABLE IF EXISTS roles;
DROP TABLE IF EXISTS version;
DROP TABLE IF EXISTS sci;
DROP TABLE IF EXISTS requests;
SET SQL SAFE UPDATES = 0;
CREATE TABLE roles (
     id INT NOT NULL AUTO_INCREMENT,
     roleName VARCHAR(45) NOT NULL,
     PRIMARY KEY (id)
);
CREATE TABLE sci (
     id INT NOT NULL AUTO INCREMENT,
     description VARCHAR (45) NOT NULL,
     PRIMARY KEY (id)
);
CREATE TABLE version (
     id INT NOT NULL AUTO INCREMENT,
     versionName VARCHAR(45) NOT NULL,
    scild INT NOT NULL,
     PRIMARY KEY (id),
     CONSTRAINT fk sciID FOREIGN KEY (sciId) REFERENCES sci(id)
);
CREATE TABLE users (
     id INT NOT NULL AUTO INCREMENT,
     userName VARCHAR(45) NOT NULL,
     password VARCHAR (200) NOT NULL,
     roleID INT NOT NULL,
     lastlogon VARCHAR (45) NULL,
     PRIMARY KEY (id),
     CONSTRAINT fk roleID FOREIGN KEY (roleID) REFERENCES roles(id)
);
CREATE TABLE requests (
     id INT NOT NULL AUTO INCREMENT,
     versionID INT NOT NULL,
     userID INT NOT NULL,
     problem VARCHAR (45) NOT NULL,
     solution VARCHAR (45) NOT NULL,
     type VARCHAR(45) NOT NULL,
     assessment VARCHAR (45) NOT NULL,
     PRIMARY KEY (id),
     CONSTRAINT fk userID FOREIGN KEY (userID) REFERENCES users (id),
     CONSTRAINT fk versionID FOREIGN KEY (versionID) REFERENCES
version(id)
);
```

This is the ERR diagram for my MySQL database:



### **README**

All steps have been implemented fully and are fully functional. By using MockMVC I have been able to send requests and use the pages to be able to access the different parts of the web application.