**CS544**

**Enterprise Architecture Final**

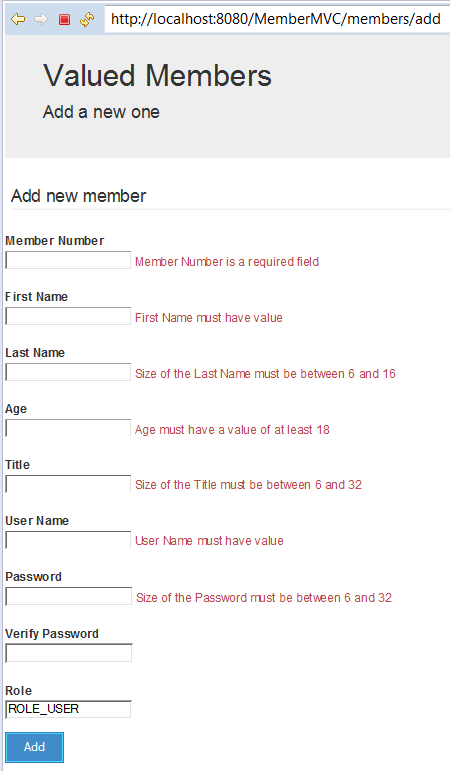
## Exam 2 Example

Name\_\_\_\_\_\_\_\_\_Pheakdey Luk\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID \_\_\_\_\_986591\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**NOTE: This material is private and confidential. It is the property of MUM and is not to be disseminated.**

1. [15 Points] The following screen displays the member entry screen. Annotate The Member & User Credentials domain objects to reflect the Validation requirements. The provided Member Controller also needs to be annotated. Also the controller methods need to be completed as necessary [ including validation support].



**Here is the relevant part of the Member Domain Class:**

@Entity

**public** **class** Member {

@Id

@GeneratedValue(strategy=GenerationType.***AUTO***)

**private** **long** id;

@Column(length = 16)

@NotEmpty(message=”{String.empty}”)

**private** String firstName;

@Column(length = 16)

@Size(min=6, max=16, message=”{Size.name.validation}”)

**private** String lastName;

@Min(value=18, message=”{Min.age}”)

**private** Integer age;

@Column(length = 32)

@Size(min=6, max=32, message=”{Size.title.validation}”)

**private** String title;

@NotNull(message=”{Integer.empty}”)

**private** Integer memberNumber;

@OneToOne(fetch=FetchType.***EAGER***, cascade = CascadeType.***ALL***)

@JoinColumn(name="member\_id")

UserCredentials userCredentials;

**Here is the UserCredentials:**

@Entity(name = "Authentication")

**public** **class** UserCredentials {

@Id

@Column(name = "USER", nullable = **false**, unique = **true**, length = 127)

@NotEmpty(message=”{String.userName}”)

String userName;

@Column(name = "PASSWORD", nullable = **false**, length = 32)

@Size(min=6, max=32, message=”{Size.password.validation}”)

String password;

@Column( nullable = **false**, length = 32)

String verifyPassword;

Boolean enabled;

@OneToOne(mappedBy="userCredentials", cascade = {CascadeType.***PERSIST***, CascadeType.***MERGE***})

**private** User user;

**ErrorMessage.properties**

String.empty = {0} must have a value

Size.name.validation = size of {0} must be between {2} and {1}

Mix.age = {0} must have a value of at least {1}

Size.title.validation = {0} must have a value

Size.password.validation = {0} must be between {2} and {1}

Integer.empty = {0} is required field

MemberNumber = Member Number

fistName = Fist Name

age = Age

title = Title

UserCredentials.userName = User Nmae

UserCredentials.password = Password

**MemberController.java**

**public** **class** MemberController {

@Autowired

**private** MemberService memberService;

@RequestMapping(value=”/add”, method=RequestMapping.GET)

**public** String getAddNewMemberForm(@ModelAttribute("newMember") Member newMember) {

**return** "addMember";

}

@RequestMapping(value=”/add”, method=RequestMapping.POST)

**public** String processAddNewMemberForm(@Valid @ModelAttribute(“newmeber”) Member memberTobeAdded, BindingResult result) {

**if(result.hassError()){**

**return “addMember”;**

**}**

memberService.save(memberToBeAdded);

**return** "redirect:/members";

}