## **Spring MVC 3-level architecture** Note: Every dependency without label here is : Manage Data(Service -> Data Access ) Note: Each class in this architecture is a sinleton and the property is set automatically by Spring framework Data Service Layer Access Controller Layer AccountController AccountService Uses + getAllUsers(): List<User> + getAllUsers(): List<User> register(request:RegisterRequest): Result register(request:RegisterRequest): Result + registerAdmin(adminId:String, email:String, password:String): Result + login(email:String, password:String): Result registerAdmin(adminId:String, email:String, password:String): Result login(email:String, password:String): Result UserRepository + findByEmailaddress(emailaddress:String): User + findByStudentId(studentId:String):User **ArtifactService** + findUserByFirstname(firstname:String): List<User> ArtifactController + findUserByLastname(lastname:String): List<User> + showCategoryArtifactAttachments(): Map<String, Map<String, List<String>>> + findUserByUsername(username:String): List<User> + showCategoryArtifactAttachments(): Map<String, Map<String, List<String>>> + upload(email:String, category:String, title:String, description:String, + findUserByLink(link:String): User + upload(email:String, category:String, title:String, description:String, attachment:List<String>, privacy:String): Result Uses attachment:List<String>, privacy:String): Result +deleteArtifact(request: ViewArtifact): Result +deleteArtifact(request: ViewArtifact): Result +viewAttachment(request:ViewAttachmentRequest): ViewAttachmentResult +viewAttachment(request:ViewAttachmentRequest): ViewAttachmentResult +switchPrivacy(email:String, category:String, artifact:String, privacy:String): Result +switchPrivacy(email:String, category:String, artifact:String, privacy:String): Result CacheService CacheController Uses uploadCache(email:String, file:MultipartFile): Result CacheRepository - uploadCache(email:String, file:MultipartFile): Result + clearCache(email:String):Result - clearCache(email:String):Result + findByEmailaddress(emailaddress:String): UserCache + deleteByEmailaddress(emailaddress:String): Void + findAllByEmailaddress(emailaddress:String): List<UserCache> CategoryController CategoryService + addCategory(request:CategoryRequest): Result - addCategory(request:CategoryRequest): Result Uses + deleteCategory(request:CategoryRequest): Result - deleteCategory(request:CategoryRequest): Result + showCategory(request:AllCategoryRequest): CategoryRequest + showCategory(request:AllCategoryRequest): CategoryRequest GuestService GuestController Uses -findPubByLink(link:String): Map<String, Map<String, List<String>>> + findPubByLink(link:String): Map<String, Map<String, List<String>>> + findProfileByLink(link:String): Profile + findProfileByLink(link:String): Profile AdminRepository + findByEmailaddress(emailaddress:String): Administrator **ProfileController ProfileController** + findAllByAdminId(adminId:String): Administrator Uses updateAvatar(email:String, image:MultipartFile): Result + updateAvatar(email:String, image:MultipartFile): Result + deleteAvatar(email:String): Result + deleteAvatar(email:String): Result + getProfile(email:String): Profile + getProfile(email:String): Profile + updateProfile(email:String, firstname:String, lastname:String/username:String): Result + updateProfile(email:String, firstname:String, lastname:String, username:String): Result + generateLink(email:String): String + generateLink(email:String): String SearchService SearchController Uses + showUser(info: String): List<User> + showUser(info: String): List<User>

