**Traceability diagram:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirements** | **Packages** | **Class** | **Method** |
| Create Model | ui | Main | createController() |
| model | Controller | Controller() |
| Req#1: register a community | ui | Main | registerCommunity() |
| model | Controller | searchCommunity(name:String) |
| model | Community | getName() |
| model | Controller | availableCommunity() |
| model | Controller | displayChallenges() |
| model | Community | getChallenges() |
| model | Challenge | getChallenges() |
| model | Challenge | Challenge(description : String) |
| model | Challenge | getDescription() |
| model | Controller | displayCommunityTypes() |
| model | Community | getCommunityTypes() |
| model | CommunityType | getCommunityTypes() |
| model | CommunityType | CommunityType(description: String) |
| model | CommunityType | getDescription() |
| model | Challenge | intToChallenge(intChallenge : int) |
| model | CommunityType | intToCommunityType(intType: int) |
| model | Controller | registerCommunity(name : String, representantName : String, representantPhone : String, population : int, intChallenge : int, intType : int) |
| model | Community | Community(name : String, representantName : String, representantPhone : String, population : int, intChallenge : int, intType : int) |
| Req#2: Register a place | ui | Main | registerPlace() |
| model | Controller | getCommunities() |
| model | Controller | oneMinCommunity() |
| model | Community | getName() |
| model | Controller | intToCommunity() |
| model | Controller | displayPlaceTypes() |
| model | Place | getPlaceTypes() |
| model | PlaceType | getPlaceTypes() |
| model | PlaceType | PlaceType(description: String) |
| model | PlaceType | getDescription() |
| model | Controller | displayDepartments() |
| model | Place | getDepartments() |
| model | Department | getDepartments() |
| model | Department | Department(description: String) |
| model | Department | getDescription() |
| model | Controller | searchPlace(name:String) |
| model | Place | getName() |
| model | Controller | availablePlace() |
| model | Controller | registerPlace(intCommunity: int, placeName : String, area : double, inaugurationDate : Calendar, photo : String, resources : double, intDepartment : int, intType : int) |
| model | Place | Place(name : String, area : double, inaugurationDate : Calendar, photo : URL, resources : double, department : Department, type : PlaceType, caregivingCommunity: Community) |
| model | PlaceType | intToPlaceType(intType : int) |
| model | Department | intToDepartment(intDepartment : int) |
| Req#3: Add a product to a community | ui | Main | addProduct() |
| model | Controller | oneMinCommunity() |
| model | Controller | getCommunities() |
| model | Community | getName() |
| model | Controller | displayProductTypes() |
| model | Community | getProductTypes() |
| model | Product | getProductTypes() |
| model | ProductType | getProductTypes() |
| model | ProductType | PlaceType(description: String) |
| model | ProductType | getDescription() |
| model | Controller | addProduct(intCommunity : int, productName : String, naturePercent : double, handcraft : String, intType : int) |
| model | Controller | intToCommunity() |
| model | Product | Product(name : String, naturePercent : double, handcraft : String, intType: int) |
| Model | ProductType | intToProductType(intType : int) |
| model | Community | addProduct(newProduct : Product) |
| model | Community | searchProduct(productName : String) |
| model | Product | getName() |
| model | Community | availableProduct() |
| Req#4: Eliminate a product from a community | ui | Main | deleteProduct() |
| model | Controller | searchCommunity(communityName:String) |
| model | Community | getName() |
| model | Community | searchProduct(productName: String) |
| model | Product | getName() |
| model | Controller | deleteProduct(communityName:String, productName: String) |
| model | Community | deleteProduct(productName:String) |
| Req#5: Add species to a place | ui | Main | addSpecies() |
| model | Controller | searchPlace(placeName: String) |
| model | Place | getName() |
| model | Place | availableSpecies() |
| model | Place | searchSpecies(speciesName: String) |
| model | Species | getName() |
| model | Controller | addSpecies(communityName : String, speciesName : String, photo : String, localPop : int, type : SpeciesType) |
| model | Place | addSpecies(speciesName : String, photo : String, localPop: int, type : SpeciesType) |
| model | Species | Species(name : String, photo : String, localPop: int, type : SpeciesType) |
| Req#6: Modify a species data in a place | ui | Main | modifySpecies() |
| model | Controller | searchPlace(placeName : String) |
| model | Place | getName() |
| model | Place | searchSpecies(oldSpeciesName:String) |
| model | Species | getName() |
| model | Controller | modifySpecies(placeName: String, oldSpeciesName : String, newSpeciesName, photo : Image, localPop : int, type:SpeciesType) |
| model | Place | modifySpecies(oldSpeciesName : String, newSpeciesName, photo : Image, localPop : int, type:SpeciesType) |
| model | Place | searchSpecies(newSpeciesName: String) |
| model | Species | getName() |
| model | Species | setName(newSpeciesName : String) |
| model | Species | setType(type : SpeciesType) |
| model | Species | setPhoto(photo : String) |
| model | Species | getPhoto(photo : String) |
| model | Species | setLocalPop(localPop : int) |
| Req#7: Access a place information. | ui | Main | accessPlace() |
| model | Controller | searchPlace(name: String) |
| model | Place | getName() |
| model | Controller | accessPlace(name:String) |
| model | Place | toString() |
| model | Community | toString() |
| Req#8: Access the information of the communities in a department. | ui | Main | accessDeptCommunity() |
| model | Controller | searchDepartment(department : Department) |
| model | Place | getDepartment() |
| model | Controller | accessDeptCommunity(department:Department) |
| model | Place | getCaregivingCommunity() |
| model | Community | getName() |
| model | Place | searchCommunity(name : String) |
| model | Community | toString() |
| Req#9: Access the information of communities based on their biggest problematic. | ui | Main | accessChallengeCommunity() |
| model | Controller | searchChallenge(challenge:Challenge) |
| model | Community | getChallenge() |
| model | Controller | accessChallengeCommunity(challenge1 : Challenge, challenge2 : Challenge) |
| model | Community | toString() |
| Req#10: Display the name of the place with the most species diversity. | ui | Main | mostDiverse() |
| model | Controller | mostDiverse() |
| model | Place | speciesCount() |
| model | Place | getName() |
| Req#11: Display the three places with the biggest area. | ui | Main | biggestArea() |
| model | Controller | biggestArea() |
| model | Place | getArea() |
| model | Place | getName() |