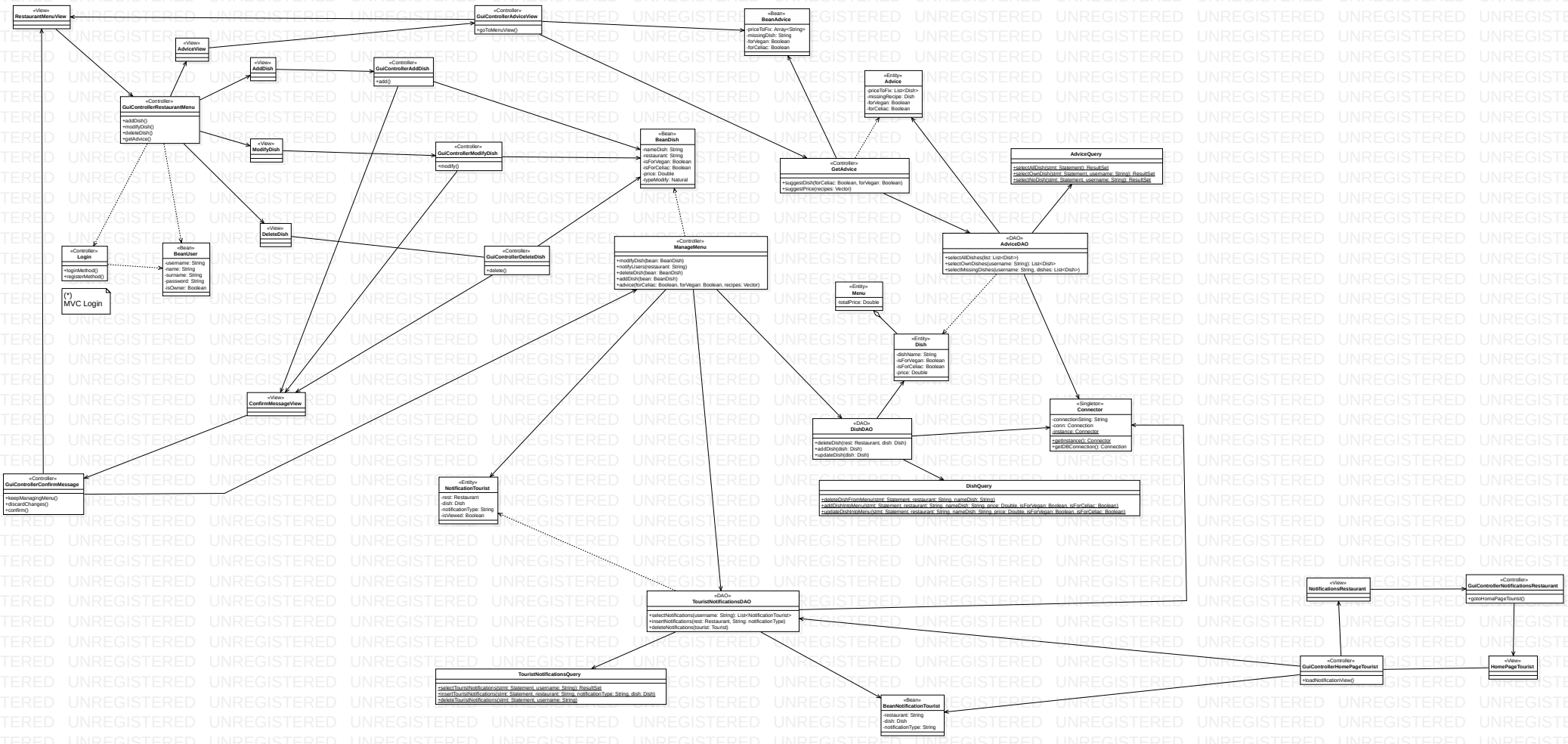
Model.mdm


«View»
RestaurantMenuView
«View»
AdviceView
«View»
AddDish
«View»
ModifyDish
«View»
DeleteDish
«View»
ConfirmMessageView
«View»
NotificationsRestaurant
«View»
HomePageTourist
«Controller»
GuiControllerRestaurantMenu
+addDish()
+modifyDish()
+deleteDish()
+getAdvice()
«Controller»
GuiControllerAdviceView
+goToMenuView()
«Controller»
GuiControllerAddDish
+add()
«Controller»
GuiControllerModifyDish
+modify()
«Controller»
GuiControllerDeleteDish
+delete()
«Controller»
Login
+loginMethod()
+registerMethod()
(*)
MVC Login
«Controller»
GuiControllerConfirmMessage
+keepManagingMenu()
+discardChanges()
+confirm()
«Controller»
GuiControllerNotificationsRestaurant
+gotoHomePageTourist()
«Controller»
GuiControllerHomePageTourist
+badNotificationView()
«Bean»
BeanAdvice
-priceToFix: Array<String>
-missingDish: String
-forVegan: Boolean
-forCeliac: Boolean
«Bean»
BeanDish
-nameDish: String
-restaurant: String
-isForVegan: Boolean
-isForCeliac: Boolean
-price: Double
-typeModify: Natural
«Bean»
BeanUser
-username: String
-name: String
-surname: String
-password: String
-isOwner: Boolean
«Bean»
BeanNotificationTourist
-restaurant: String
-dish: Dish
-notificationType: String
«Entity»
Advice
-priceToFix: List<Dish>
-missingRecipe: Dish
-forVegan: Boolean
-forCeliac: Boolean
«Entity»
Menu
-totalPrice: Double
«Entity»
Dish
-dishName: String
-isForVegan: Boolean
-isForCeliac: Boolean
-price: Double
«Entity»
NotificationTourist
-rest: Restaurant
-dish: Dish
-notificationType: String
-isViewed: Boolean
«Controller»
GetAdvice
+suggestDish(forCeliac: Boolean, forVegan: Boolean)
+suggestPrice(recipes: Vector)
«Controller»
ManageMenu
+modifyDish(bean: BeanDish)
+modifyUserRestaurant: String)
+deleteDish(bean: BeanDish)
+addDish(bean: BeanDish)
+advice(forCeliac: Boolean, forVegan: Boolean, recipes: Vector)
AdviceQuery
+selectAllDishes(stmt: Statement): ResultSet
+selectOwnDishes(stmt: Statement, username: String): ResultSet
+selectMissingDishes(stmt: Statement, username: String): ResultSet
«DAO»
AdviceDAO
+selectAllDishes(list: List<Dish>)
+selectOwnDishes(username: String): List<Dish>
+selectMissingDishes(username: String, dishes: List<Dish>)
«DAO»
DishDAO
+deleteDish(rest: Restaurant, dish: Dish)
+addDish(dish: Dish)
+updateDish(dish: Dish)
«Singleton»
Connector
-connectionString: String
-conn: Connection
-instance: Connector
+getInstance(): Connector
-getDbConnection(): Connection
DishQuery
+deleteDishFromMenu(stmt: Statement, restaurant: String, nameDish: String)
+addDishIntoMenu(stmt: Statement, restaurant: String, nameDish: String, price: Double, isForVegan: Boolean, isForCeliac: Boolean)
+updateDishIntoMenu(stmt: Statement, restaurant: String, nameDish: String, price: Double, isForVegan: Boolean, isForCeliac: Boolean)
«DAO»
TouristNotificationsDAO
+selectNotifications(username: String): List<NotificationTourist>
+insertNotifications(rest: Restaurant, String: notificationType)
+deleteNotifications(tourist: TouristId)
TouristNotificationsQuery
+selectTouristNotifications(stmt: Statement, username: String): ResultSet
+insertTouristNotifications(stmt: Statement, restaurant: String, notificationType: String, dish: Dish)
+deleteTouristNotifications(stmt: Statement, username: String)