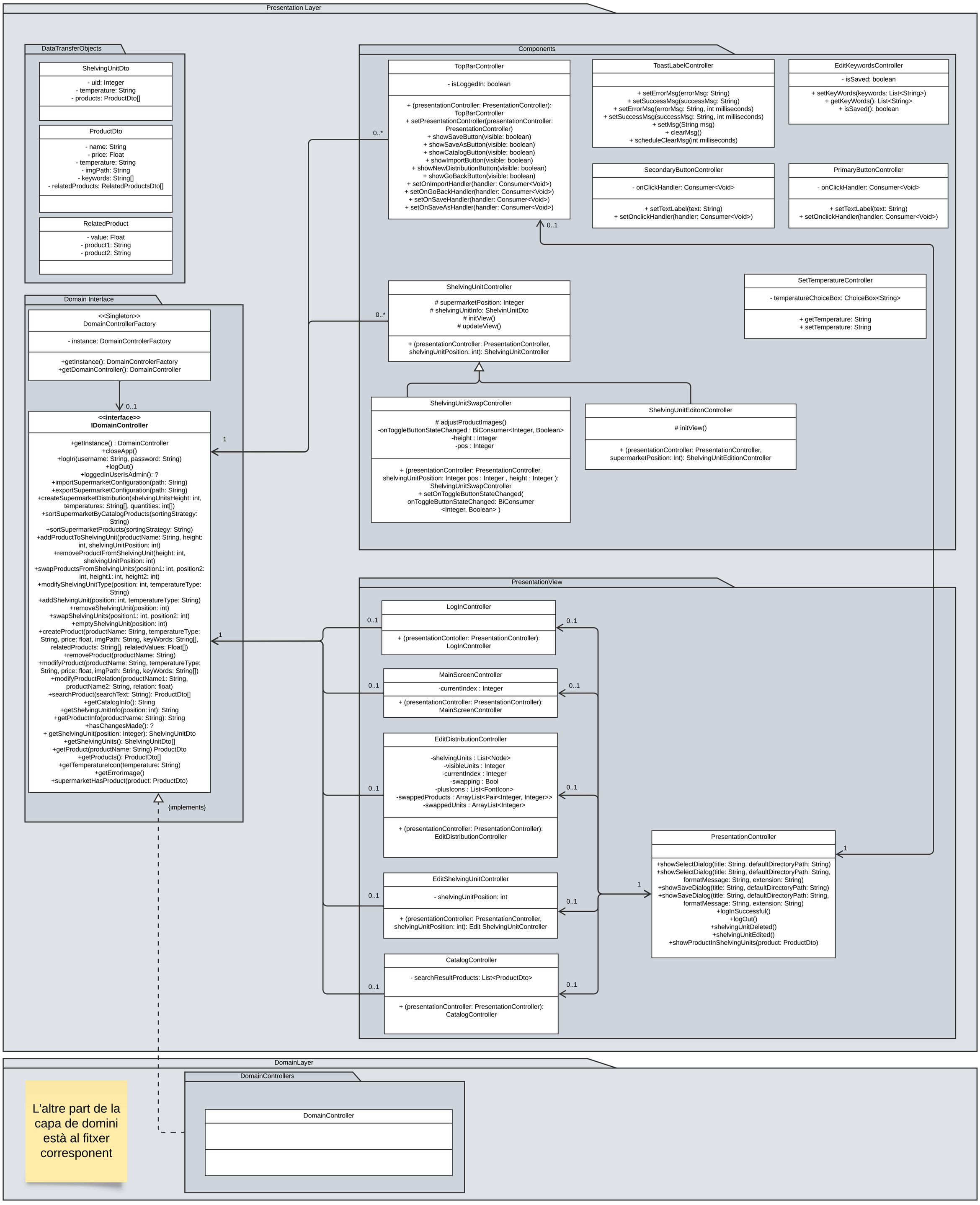
Presentation Layer
DataTransferObjects
ShelvingUnitDto
- uid: Integer
- temperature: String
- products: ProductDto[]
ProductDto
- name: String
- price: Float
- temperature: String
- imgPath: String
- keywords: String[]
- relatedProducts: RelatedProductsDto[]
RelatedProduct
- value: Float
- product1: String
- product2: String
Domain Interface
<<Singleton>>
DomainControllerFactory
- instance: DomainControlerFactory
+getInstance(): DomainControlerFactory
+getDomainController(): DomainController
0..1
<<interface>>
IDomainController
+getInstance() : DomainController
+closeApp()
+logIn(username: String, password: String)
+logOut()
+loggedInUserIsAdmin(): ?
+importSupermarketConfiguration(path: String)
+exportSupermarketConfiguration(path: String)
+createSupermarketDistribution(shelvingUnitsHeight: int, temperatures: String[], quantities: int[])
+sortSupermarketByCatalogProducts(sortingStrategy: String)
+sortSupermarketProducts(sortingStrategy: String)
+addProductToShelvingUnit(productName: String, height: int, shelvingUnitPosition: int)
+removeProductFromShelvingUnit(height: int, shelvingUnitPosition: int)
+swapProductsFromShelvingUnits(position1: int, position2: int, height1: int, height2: int)
+modifyShelvingUnitType(position: int, temperatureType: String)
+addShelvingUnit(position: int, temperatureType: String)
+removeShelvingUnit(position: int)
+swapShelvingUnits(position1: int, position2: int)
+emptyShelvingUnit(position: int)
+createProduct(productName: String, temperatureType: String, price: float, imgPath: String, keyWords: String[], relatedProducts: String[], relatedValues: Float[])
+removeProduct(productName: String)
+modifyProduct(productName: String, temperatureType: String, price: float, imgPath: String, keyWords: String[])
+modifyProductRelation(productName1: String, productName2: String, relation: float)
+searchProduct(searchText: String): ProductDto[]
+getCatalogInfo(): String
+getShelvingUnitInfo(position: int): String
+getProductInfo(productName: String): String
+hasChangesMade(): ?
+ getShelvingUnit(position: Integer): ShelvingUnitDto
+getShelvingUnits(): ShelvingUnitDto[]
+getProduct(productName: String) ProductDto
+getProducts(): ProductDto[]
+getTemperatureIcon(temperature: String)
+getErrorImage()
+supermarketHasProduct(product: ProductDto)
{implements}
Components
TopBarController
- isLoggedIn: boolean
+ (presentationController: PresentationController): TopBarController
+ setPresentationController(presentationController: PresentationController)
+ showSaveButton(visible: boolean)
+ showSaveAsButton(visible: boolean)
+ showCatalogButton(visible: boolean)
+ showImportButton(visible: boolean)
+ showNewDistributionButton(visible: boolean)
+ showGoBackButton(visible: boolean)
+ setOnImportHandler(handler: Consumer<Void>)
+ setOnGoBackHandler(handler: Consumer<Void>)
+ setOnSaveHandler(handler: Consumer<Void>)
+ setOnSaveAsHandler(handler: Consumer<Void>)
ToastLabelController
+ setErrorMsg(errorMsg: String)
+ setSuccessMsg(successMsg: String)
+ setErrorMsg(errorMsg: String, int milliseconds)
+ setSuccessMsg(successMsg: String, int milliseconds)
+ setMsg(String msg)
+ clearMsg()
+ scheduleClearMsg(int milliseconds)
EditKeywordsController
- isSaved: boolean
+ setKeyWords(keywords: List<String>)
+ getKeyWords(): List<String>
+ isSaved(): boolean
SecondaryButtonController
- onClickHandler: Consumer<Void>
+ setTextLabel(text: String)
+ setOnclickHandler(handler: Consumer<Void>)
PrimaryButtonController
- onClickHandler: Consumer<Void>
+ setTextLabel(text: String)
+ setOnclickHandler(handler: Consumer<Void>)
ShelvingUnitController
supermarketPosition: Integer
shelvingUnitInfo: ShelvinUnitDto
initView()
updateView()
+ (presentationController: PresentationController, shelvingUnitPosition: int): ShelvingUnitController
SetTemperatureController
- temperatureChoiceBox: ChoiceBox<String>
+ getTemperature: String
+ setTemperature: String
ShelvingUnitSwapController
adjustProductImages()
-onToggleButtonStateChanged : BiConsumer<Integer, Boolean>
-height : Integer
-pos : Integer
+ (presentationController: PresentationController, shelvingUnitPosition: Integer pos : Integer , height : Integer): ShelvingUnitSwapController
+ setOnToggleButtonStateChanged(onToggleButtonStateChanged: BiConsumer <Integer, Boolean>)
ShelvingUnitEditonController
initView()
+ (presentationController: PresentationController, supermarketPosition: Int): ShelvingUnitEditionController
0..*
0..1
PresentationView
LogInController
+ (presentationContoller: PresentationController): LogInController
MainScreenController
-currentIndex : Integer
+ (presentationController: PresentationController): MainScreenController
EditDistributionController
-shelvingUnits : List<Node>
-visibleUnits : Integer
-currentIndex : Integer
-swapping : Bool
-plusIcons : List<FontIcon>
-swappedProducts : ArrayList<Pair<Integer, Integer>>
-swappedUnits : ArrayList<Integer>
+ (presentationController: PresentationController): EditDistributionController
EditShelvingUnitController
- shelvingUnitPosition: int
+ (presentationController: PresentationController, shelvingUnitPosition: int): Edit ShelvingUnitController
CatalogController
- searchResultProducts: List<ProductDto>
+ (presentationController: PresentationController): CatalogController
PresentationController
+showSelectDialog(title: String, defaultDirectoryPath: String)
+showSelectDialog(title: String, defaultDirectoryPath: String, formatMessage: String, extension: String)
+showSaveDialog(title: String, defaultDirectoryPath: String)
+showSaveDialog(title: String, defaultDirectoryPath: String, formatMessage: String, extension: String)
+logInSuccessful()
+logOut()
+shelvingUnitDeleted()
+shelvingUnitEdited()
+showProductInShelvingUnits(product: ProductDto)
1
0..1
DomainLayer
DomainControllers
DomainController
L'altre part de la capa de domini està al fitxer corresponent