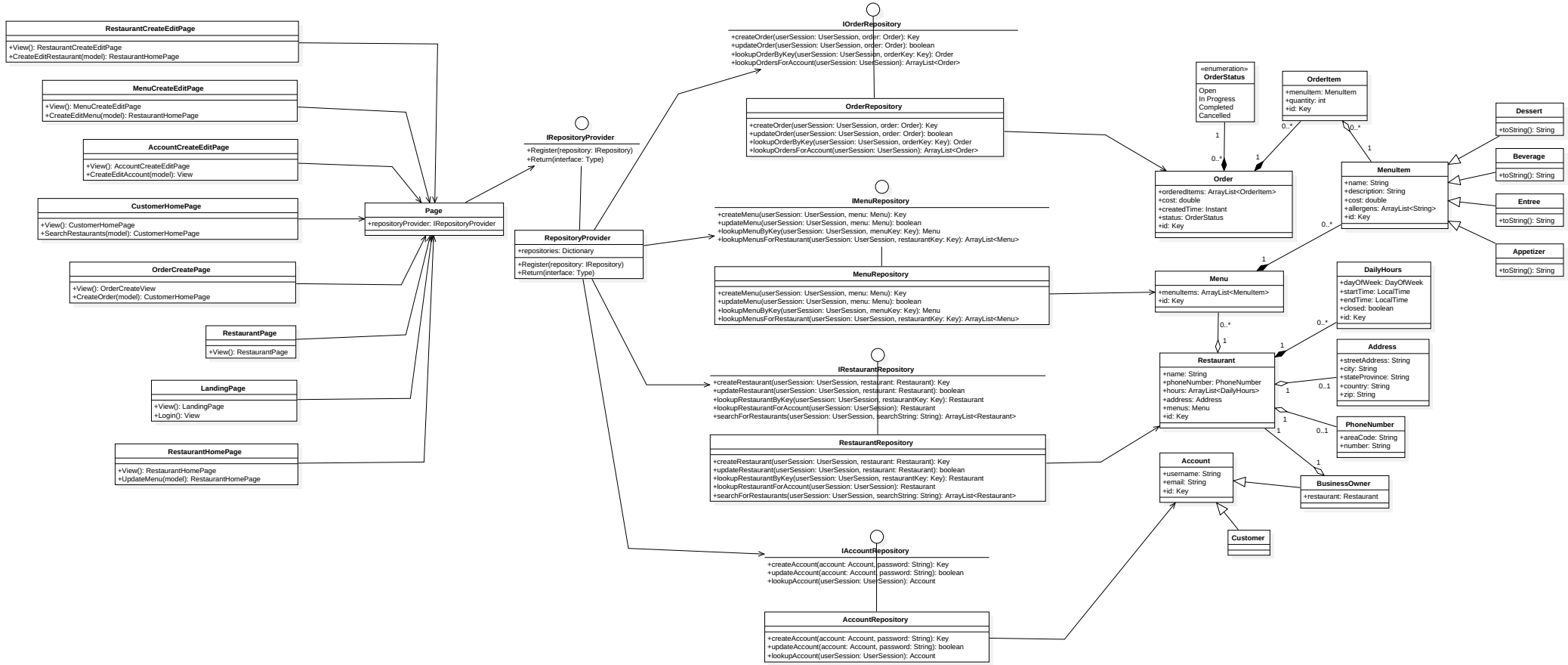
Model::Main


RestaurantCreateEditPage
+View(): RestaurantCreateEditPage
+CreateEditRestaurant(model): RestaurantHomePage
MenuCreateEditPage
+View(): MenuCreateEditPage
+CreateEditMenu(model): RestaurantHomePage
AccountCreateEditPage
+View(): AccountCreateEditPage
+CreateEditAccount(model): View
CustomerHomePage
+View(): CustomerHomePage
+SearchRestaurants(model): CustomerHomePage
OrderCreatePage
+View(): OrderCreateView
+CreateOrder(model): CustomerHomePage
RestaurantPage
+View(): RestaurantPage
LandingPage
+View(): LandingPage
+Login(): View
RestaurantHomePage
+View(): RestaurantHomePage
+UpdateMenu(model): RestaurantHomePage
Page
+repositoryProvider: IRepositoryProvider
IRepositoryProvider
+Register(repository: IRepository)
+Return(interface: Type)
RepositoryProvider
+repositories: Dictionary
+Register(repository: IRepository)
+Return(interface: Type)
IOrderRepository
+createOrder(userSession: UserSession, order: Order): Key
+updateOrder(userSession: UserSession, order: Order): boolean
+lookupOrderByKey(userSession: UserSession, orderKey: Key): Order
+lookupOrdersForAccount(userSession: UserSession): ArrayList<Order>
OrderRepository
+createOrder(userSession: UserSession, order: Order): Key
+updateOrder(userSession: UserSession, order: Order): boolean
+lookupOrderByKey(userSession: UserSession, orderKey: Key): Order
+lookupOrdersForAccount(userSession: UserSession): ArrayList<Order>
IMenuRepository
+createMenu(userSession: UserSession, menu: Menu): Key
+updateMenu(userSession: UserSession, menu: Menu): boolean
+lookupMenuByKey(userSession: UserSession, menuKey: Key): Menu
+lookupMenusForRestaurant(userSession: UserSession, restaurantKey: Key): ArrayList<Menu>
MenuRepository
+createMenu(userSession: UserSession, menu: Menu): Key
+updateMenu(userSession: UserSession, menu: Menu): boolean
+lookupMenuByKey(userSession: UserSession, menuKey: Key): Menu
+lookupMenusForRestaurant(userSession: UserSession, restaurantKey: Key): ArrayList<Menu>
IRestaurantRepository
+createRestaurant(userSession: UserSession, restaurant: Restaurant): Key
+updateRestaurant(userSession: UserSession, restaurant: Restaurant): boolean
+lookupRestaurantByKey(userSession: UserSession, restaurantKey: Key): Restaurant
+lookupRestaurantForAccount(userSession: UserSession): Restaurant
+searchForRestaurants(userSession: UserSession, searchString: String): ArrayList<Restaurant>
RestaurantRepository
+createRestaurant(userSession: UserSession, restaurant: Restaurant): Key
+updateRestaurant(userSession: UserSession, restaurant: Restaurant): boolean
+lookupRestaurantByKey(userSession: UserSession, restaurantKey: Key): Restaurant
+lookupRestaurantForAccount(userSession: UserSession): Restaurant
+searchForRestaurants(userSession: UserSession, searchString: String): ArrayList<Restaurant>
IAccountRepository
+createAccount(account: Account, password: String): Key
+updateAccount(account: Account, password: String): boolean
+lookupAccount(userSession: UserSession): Account
AccountRepository
+createAccount(account: Account, password: String): Key
+updateAccount(account: Account, password: String): boolean
+lookupAccount(userSession: UserSession): Account
«enumeration»
OrderStatus
Open
In Progress
Completed
Cancelled
OrderItem
+menuItem: MenuItem
+quantity: int
+id: Key
Order
+orderedItems: ArrayList<OrderItem>
+cost: double
+createdTime: Instant
+status: OrderStatus
+id: Key
MenuItem
+name: String
+description: String
+cost: double
+allergens: ArrayList<String>
+id: Key
Dessert
+toString(): String
Beverage
+toString(): String
Entree
+toString(): String
Appetizer
+toString(): String
Menu
+menuItems: ArrayList<MenuItem>
+id: Key
DailyHours
+dayOfWeek: DayOfWeek
+startTime: LocalTime
+endTime: LocalTime
+closed: boolean
+id: Key
Restaurant
+name: String
+phoneNumber: PhoneNumber
+hours: ArrayList<DailyHours>
+address: Address
+menus: Menu
+id: Key
Address
+streetAddress: String
+city: String
+stateProvince: String
+country: String
+zip: String
PhoneNumber
+areaCode: String
+number: String
Account
+username: String
+email: String
+id: Key
BusinessOwner
+restaurant: Restaurant
Customer
1
0..*
1
0..*
0..*
1
1
0..*
1
0..*
1
0..*
1
1
0..1
1
0..1
1
1