



WarehouseModelApplication - klasa główna aplikacji która posiada statyczną metodę run() wywoływaną w metodzie main(), która uruchomi całą aplikację.

Interface: **UIComponent** - interface z wzorca dekorator który posiada metodę display(), której implementacje będą odpowiadać za konkretne elementy UI, zastosowanie tego wzorca pozwala utworzyć menu konsolowe użytkownika.

ConcreteComponent - implementacja interfejsu UIComponent, tworząca nagłówek tego interface.

StorageManagementMenuDecorator – implementacja interfejsu UIComponent odpowiadający za dodanie elementów UI magazynu.

PackageManagementMenuDecorator - implementacja interfejsu UIComponent odpowiadający za dodanie elementów UI paczek.

BaseDecorator - impl. interfejsu UIComponent dodaje inne, ozdobne elementy UI.

Interface: **Command** - jest to interface z wzorca komenda, dzięki któremu będzie można tworzyć różne implementacje komend, które będą uruchamiane w z klas dekoratora.

StorageManagerCommand - implementacja interface Command - umożliwia modyfikację lub usunięcie istniejącej przestrzeni magazynowej.

DefineStorageCommand - implementacja interface Command - umożliwia dodanie nowej przestrzeni magazynowej.

DisplayCapacityCommand - implementacja interface Command - umożliwia wyświetlenie zajętości wielu przestrzeni magazynowych.

UpdatePackageCommand - implementacja interface Command - umożliwia modyfikację istniejącej paczki.

RemovePackageCommand - implementacja interface Command - umożliwia usunięcie istniejącej paczki z przestrzeni magazynowej.

AddPackageCommand - implementacja interface Command - umożliwia dodanie nowej paczki do przestrzeni magazynowej.

WarehouseStateManager - klasa która, będzie mogła posiadać tylko jedną instancję (singleton), realizująca logikę zarządzania kolejkami (przestrzeniami magazynowymi) oraz będzie przechowywać te przestrzenie magazynowe.