Project #3 UML Solutions

Version 1.20

General Information

Author	Alfonso de la Morena
Name of Project	Project 3
Instructor	Dr. Jelena Tešić
Class	CS3354: Object-Oriented Design and Programming
Due date	03/07/18

Revision History

Date	Author	Description of change
2/21/2018	Alfonso	Started project
2/26/2018	Alfonso	Parts 1 and 2
2/28/2018	Alfonso	Parts 3 and 4
3/1/2018	Alfonso	Parts 5 and 6
3/3/2018	Alfonso	Revised
3/4/2018	Alfonso	Final Revision and Submission

Part 1: Use Cases

Use Case: Customer purchases item in store

Id: UC- 001

Description
Customer will purchase an item from the store in person. Employee will take payment, record the sale in the store database, record the sale in the customer's account and give the item to customer.

Primary Actor
Customer

Supporting Actors
Employee

Pre-Conditions
Customer must have an account. Employee must have an account. Item must be in stock. Customer must have enough money to purchase the item.

Post Conditions

Success end condition
Item quantity must be removed from stock. Money must be registered in the system. Information on the sale must be recorded to the database. Information on the sale must be recorded to the customers account.

Failure end condition:
Sale is not completed. No change must be made in the database. No change must be made in the customer's account. If unable to connect to the store database the item will not be sold.

Trigger
Customer comes to store and interacts with employee.

Variations

Customer creates account but does not purchase an item. Customer edits account but does not purchase an item. Customer creates account and purchases item. Customer edits account and purchases item. Customer purchases item and does not want a receipt. Customer purchases an item and receives a receipt.

Frequency: 10000 per week

Assumptions
Customer can interact with employee without use of our system. Customer will not give system money without purchasing items. The sale will be recorded in the correct customer's account. The employee performing the sale will be logged when the sale is processed.

Use Case: Customer creates order in website

Id: UC- 002

Description

- Customer will log in to the website.
- Select the item's he/she wishes to purchase.
- Create an order.

Primary Actor
Customer

Supporting Actors
N/A

Pre-Conditions
Customer must have an account. Item must be in stock.

Post Conditions

Success end condition
Item is ordered to be sent to store. Admin must approve the item and quantities being requested. Information on the order must be recorded to the customer's account.

Failure end condition:
Order is not created or canceled. For canceled orders it must be recorded in the customer's account and it must happen before the item is ready for pickup or being sent. No changes occur if the order is not created.

Trigger
Customer logs into website and creates an order.

Variations

Customer creates account but does not create an order. Customer edits account but does not create an order. Customer creates account and orders item. Customer edits account and orders item.

Frequency: 5000 per week

Assumptions
Customer can select item's in website via computer prompts. Customer will be using his own account to create an order. Items will be delivered to a store close to the customer's location.

Use Case: Customer will pick up a website order.

Id: UC- 003

Description

- Customer will arrive at store to pick up an order made via the website.
- Employee will take payment.
- Record the sale in the store database.
- Record the sale in the customer's account.
- Give the item to customer.

Primary Actor
Customer

Supporting Actors
Employee

Pre-Conditions
Customer must have an account. Customer must have previously ordered an item to get delivered to the store. Item must have arrived. Employee will have an account. Customer will have enough money to purchase item.

Post Conditions

Success end condition
Item has arrived in store. The customer who made the order shows up to pick it up. An employee services the customer. The customer has enough money to purchase the item. The sale is recorded in the system. The sale is recorded in the customer's account.

Failure end condition:
Order is created but the customer does not show up to pick it up. Customer cancels order.

Trigger
Customer shows up at store to pick up an item they have previously ordered.

Variations

Customer edits account but does not complete order. Customer edits account and completes order. Customer completes order and does not want a receipt. Customer completes order and receives a receipt.

Frequency: 5000 per week

Assumptions
Customer can interact with employee without use of our system. Customer will not give system money without purchasing items. The sale will be recorded in the correct customer's account. The employee performing the sale will be logged when the sale is processed.

Use Case: Customer contact customer service via the website

Id: UC- 004

Description

- Customer log in to the website.
- Customer will select the option to interact with employee.
- Customer will be placed in a FIFO Queue that will place him/her in contact with the next available employee.
- Customer will interact with employee via an on-screen prompt.

Primary Actor
Customer

Supporting Actors
Employee

Pre-Conditions
Customer must have an account. There must be employees available. The queue must have a reasonable wait time.

Post Conditions

Success end condition
Customer logs in. Customer gets placed in queue. Customer interacts with employee. Customer issue is resolved.

Failure end condition:
Customer cannot log in. Customer leaves queue before employee interacts with them. Customer's issue is not solved by interacting with employee.

Trigger
Customer selects to be placed in queue to speak with an employee via the website.

Variations

Customer logs in and leaves before his turn in the queue. Customer is in contact with employee but does not resolve his issue.

Frequency: 5000 per week

Assumptions
Customer can interact with employee with use of our system. Customer requesting assistance will be the customer that is logged in. Customer will interact with an active employee.

Use Case: Customer create/edit account in person at the store

Id: UC- 005

Description

- Customer interact with an employee and inform him he/she wishes to make changes to her account.
- Customer will provide proof he/she is the customer in question.
- The change will be recorded in the system.

Primary Actor
Customer

Supporting Actors
Employee

Pre-Conditions
Customer must be in the store. Customer must know basic information about themselves. There must be employees available.

Post Conditions

Success end condition
Customer arrives at store. Informs employee of changes that he/she wishes to be made. Employee verifies customer. Employee updates customer database.

Failure end condition:
Changes are not made. Account is not created.

Trigger
Customer arrives in store and informs employee of desire to change or create a customer profile.

Variations

Customer cannot be verified and therefore unable to edit account.

Frequency: 1000 per week

Assumptions
Customer can interact with employee without use of our system. Employee will be able to verify customer without use of our system. Changes will be requested by the owner of the account.

Use Case: Customer create/edit account via the website

Id: UC- 006

Description

- Customer access the website.
- Select the option to create or edit and account.
- Customer will follow the prompts to complete this action.

Primary Actor
Customer

Supporting Actors
N/A

Pre-Conditions
Customer have access to the website. Customer must be able to verify they themselves if they wish to edit an account with a password. Customer must know basic information about themselves to create a new account.

Post Conditions

Success end condition
Customer accesses website. Customer follows prompts and achieves desired changes.

Failure end condition:
Changes are not made. Account is not created.

Trigger
Customer accesses website with a desire to edit or create an account.

Variations

Customer cannot be verified and therefore unable to edit account.

Frequency: 1000 per week

Assumptions
Customer can interact with on screen prompts with use of our system. Customer editing information will be the account owner.

Use Case: Employee will update the database after a completed sale

Id: UC- 007

Description

- Customer will purchase an item from the store.
- Employee will log in to terminal.
- Employee will select the customer that is buying from the database or create a new account if it does not exist.
- Employee will select the items being sold and add them to the transaction via on screen prompts.
- Employee will take money from the customer.
- Employee will log the sale in the database.

Primary Actor
Employee

Supporting Actors
Customer

Pre-Conditions
Employee must have an account. Item sold must be in stock. Customer must have enough money to purchase item.

Post Conditions

Success end condition
Item is sold. Database is successfully updated.

Failure end condition:
Item is not sold. No changes are made in database.

Trigger
Customer comes to store to purchase item.

Variations

Customer purchases item and creates account. Customer purchases item and edits account.

Frequency: 5000 per week

Assumptions
Customer can interact with employee without use of our system. Customer can verify themselves or create an account. Employee have an account.

Use Case: Employee will update the database after a customer picks up an order

Id: UC- 008

Description

- Customer will come in to pick up an online order.
- Employee will log in to terminal.
- Employee will select the customer that is picking up the order from the database.
- Employee will select the order being sold and add it to the transaction via on screen prompts.
- Employee will take money from the customer.
- Employee will log the order as completed in the database.

Primary Actor
Employee

Supporting Actors
Customer

Pre-Conditions
Employee must have an account. Item sold must be in stock. Customer must have enough money to purchase item. Customer must have previously ordered the item and it must have arrived.

Post Conditions

Success end condition
Order is sold. Database is successfully updated.

Failure end condition:
Order is not sold. No changes are made in database.

Trigger
Customer comes to store to complete and order.

Variations

Customer completes order and edits account.

Frequency: 5000 per week

Assumptions
Customer can interact with employee without use of our system. Customer can verify themselves. Order must have arrived. Employee have an account.

Use Case: Admin accesses reports from database.

Id: UC- 009

Description

- Admin will log in to data base
- Admin will select what information he wishes to see from the following list:
  - Income
  - Expenses and profit
  - Financial Documents (Balance Sheets, Statements of Cash Flow and Invoices.)
- Admin will also have the option of printing the Financial Documents.

Primary Actor
Admin

Supporting Actors
N/A

Pre-Conditions
Admin must have an account. Admin must have access to a computer that has the option to log in to the database.

Post Conditions

Success end condition
Information is accessed. Optional: documents are printed.

Failure end condition:
Unable to access information.

Trigger
Admin logs in to data base.

Variations

Admin prints documents.

Frequency: 21 per week

Assumptions
Admin has access to database. Data in database is updated periodically. Printer will be handled independently from our system.

Use Case: Admin adds/edits employee user.

Id: UC- 010

Description

- Admin will log in to data base
- Admin will select what a choice from the following list:
    - Add Employee information
    - Edit Employee information
- The change will be logged in database.

Primary Actor
Admin

Supporting Actors
N/A

Pre-Conditions
Admin must have an account. Admin must have access to a computer that has the option to log in to the database. Employee must know basic information about themselves.

Post Conditions

Success end condition
Employee information changes are logged.

Failure end condition:
No changes are made.

Trigger
Admin logs in to data base.

Variations

Admin adds more than one employee. Admin edits more than one employee.

Frequency: 2 per week

Assumptions
Admin has access to database. Data in database is updated periodically. Employee information entered will be correct.

Use Case: Admin orders supplies

Id: UC- 011

Description

- Admin will log in to data base
- Admin will select what a choice from the following list:
  - Order supplies.
- Items and their quantities will be selected via on screen prompts.
- The order for new items will be created.
- Money will be deducted to order the items.
- Database will be updated once items arrive.

Primary Actor
Admin

Supporting Actors
N/A

Pre-Conditions
Admin must have an account. Admin must have access to a computer that has the option to log in to the database. Admin must have enough purchasing power to make order. Items ordered must exist or get added to database.

Post Conditions

Success end condition
Order is made. Money is deducted. Items arrive and are added to stock.

Failure end condition:
No changes are made.

Trigger
Admin logs in to data base.

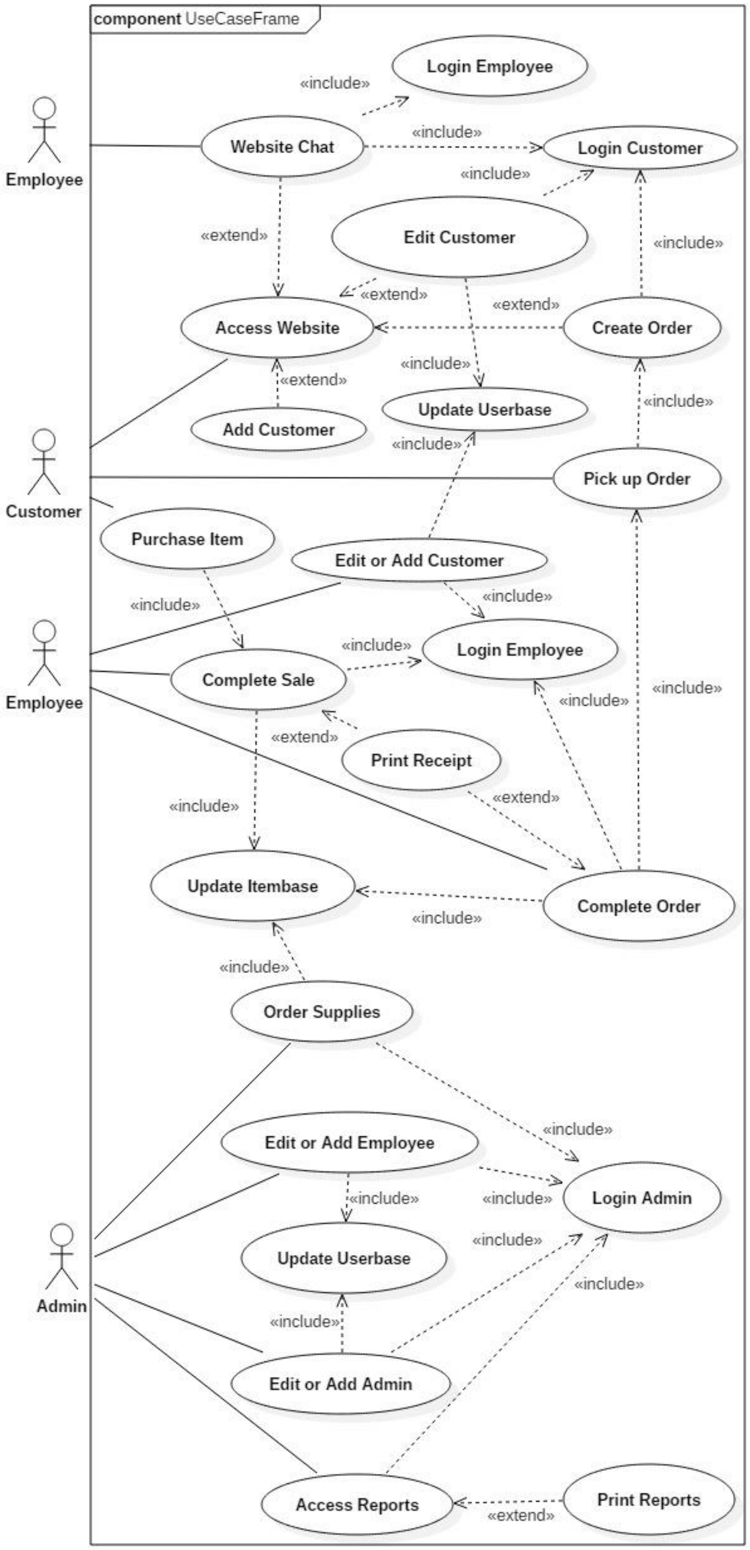
Variations

Admin orders more than one item. Admin adds new items to database.

Frequency: 14 per week

Assumptions
Admin has access to database. Data in database is updated periodically.

Part 2: Use Cases Diagram



Part 3: Class Analysis & CRC Cards

Entity Objects:

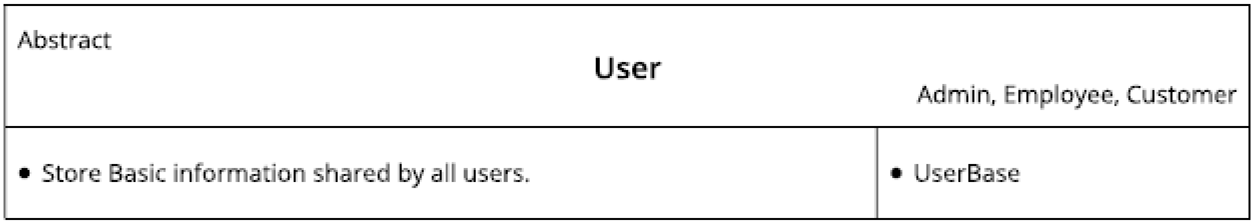
- User
  - Admin
  - Employee
  - Customer
- Database
  - User
  - Item
  - Orders
  - Transactions
  - Reports
- Transaction
- Item
  - Small Hardware
  - Appliance
- Order
  - Customer
  - Admin
- Reports

Boundary Objects:

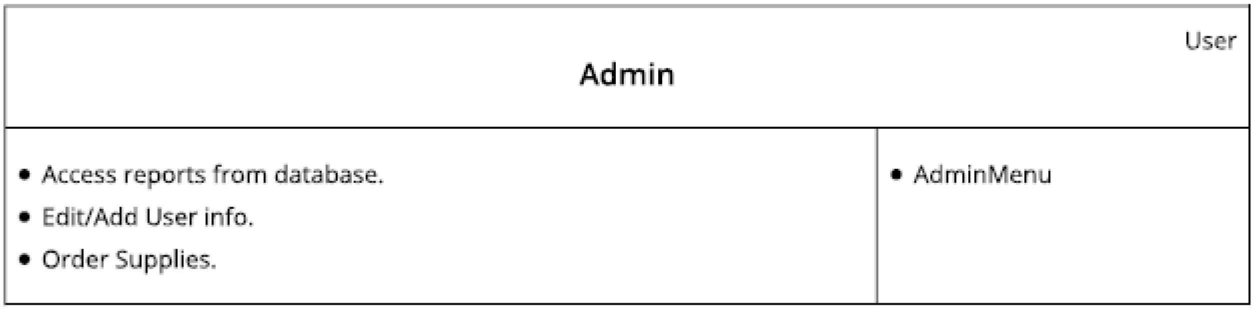
- Store Menu
  - Sale
  - Customer Info
  - Employee Login
  - Transaction (for printing transaction information)
- Website Menu
  - Customer Order
  - Customer Login
  - Customer Info
  - Employee Chat
- Admin Menu
  - User information
  - Reports
  - Admin Order

Control Objects:

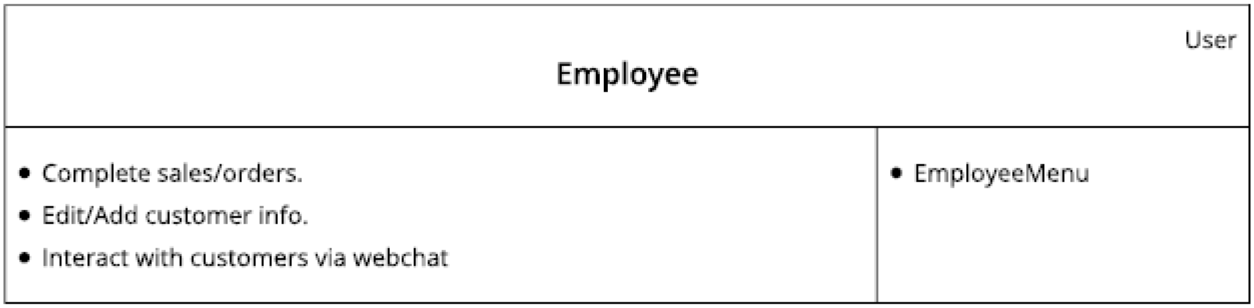
- Store Control
  - Print Transaction
  - Save Transactions
  - Save Transaction changes
  - Save Customer Changes
  - Save Item Changes
  - Save Order Changes
  - Save Report Changes
- Website Control
  - Save Orders
  - Save Customer Changes
- Admin Control
  - Print Reports
  - Save User Changes
  - Save Item Changes
  - Save Order Changes
  - Save Report Changes



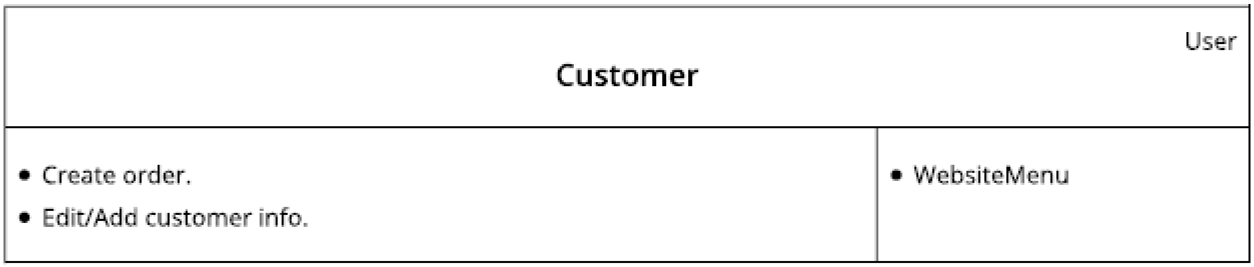
Abstract **User** Admin, Employee, Customer	
• Store Basic information shared by all users.	• UserBase



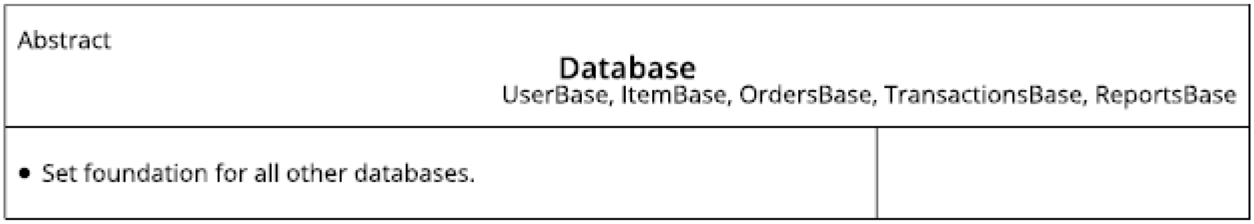
User **Admin**	
• Access reports from database. • Edit/Add User info. • Order Supplies.	• AdminMenu



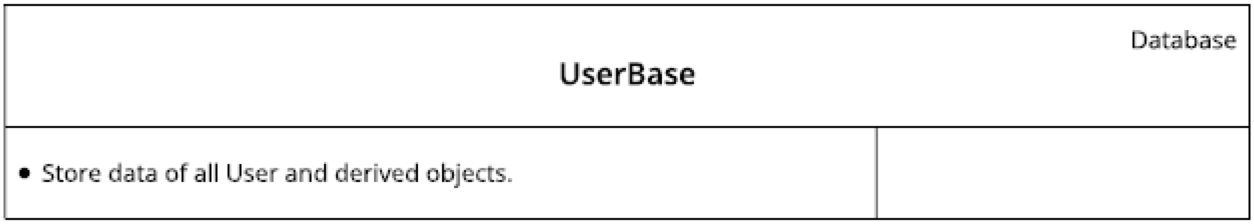
User **Employee**	
• Complete sales/orders. • Edit/Add customer info. • Interact with customers via webchat	• EmployeeMenu



User **Customer**	
• Create order. • Edit/Add customer info.	• WebsiteMenu



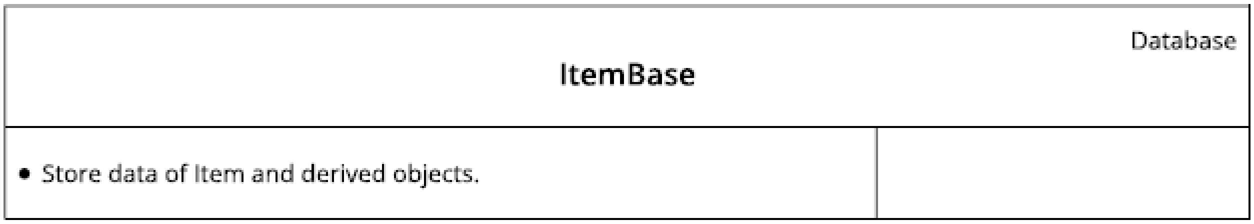
Abstract **Database** UserBase, ItemBase, OrdersBase, TransactionsBase, ReportsBase	
• Set foundation for all other databases.	



Database

UserBase

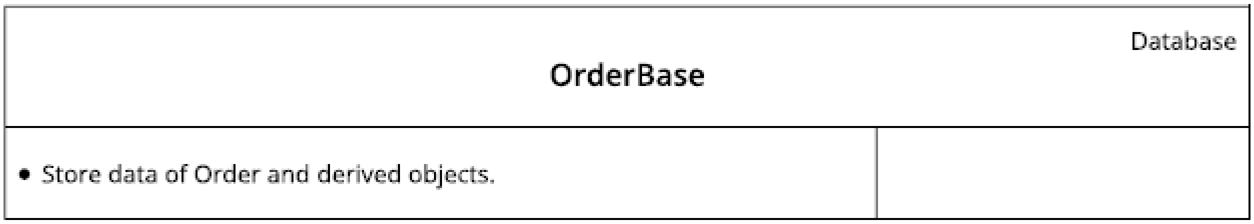
• Store data of all User and derived objects.	



Database

ItemBase

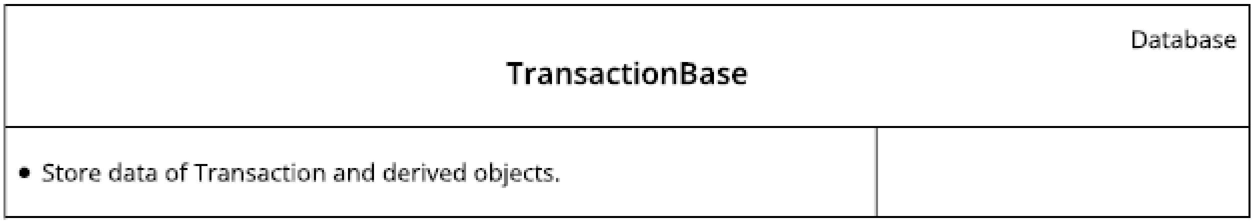
• Store data of Item and derived objects.	



Database

OrderBase

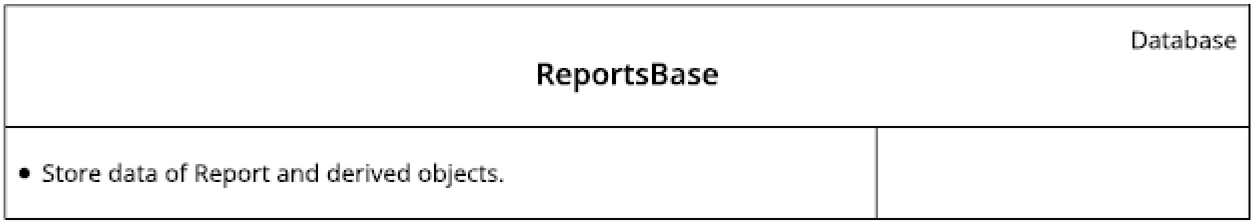
• Store data of Order and derived objects.	



Database

TransactionBase

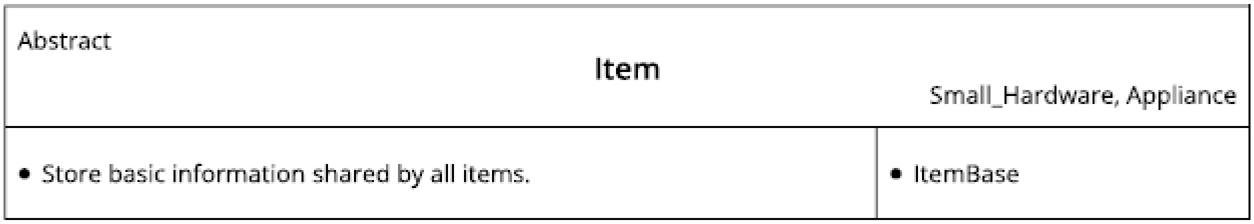
• Store data of Transaction and derived objects.	



Database

ReportsBase

• Store data of Report and derived objects.	

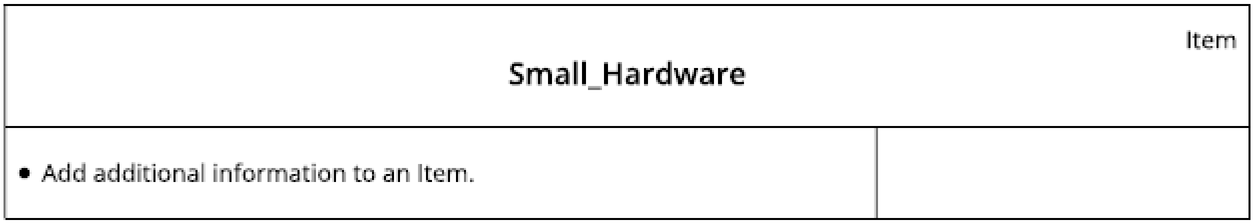


Abstract

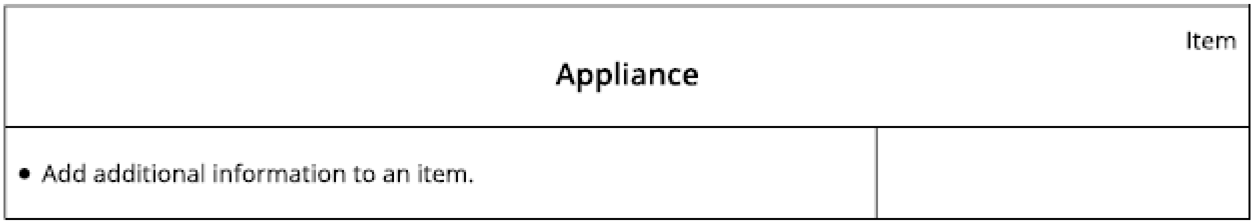
Item

Small_Hardware, Appliance

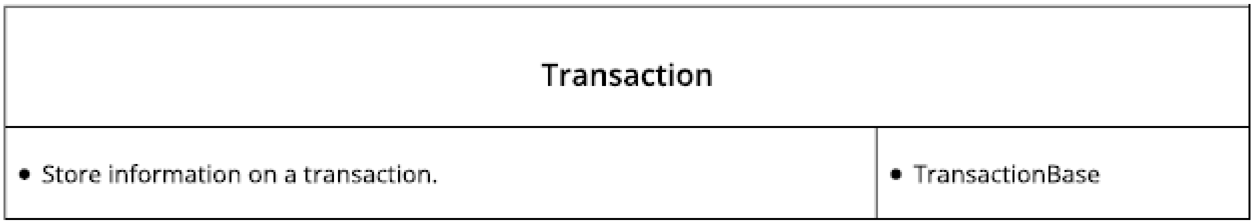
• Store basic information shared by all items.	• ItemBase



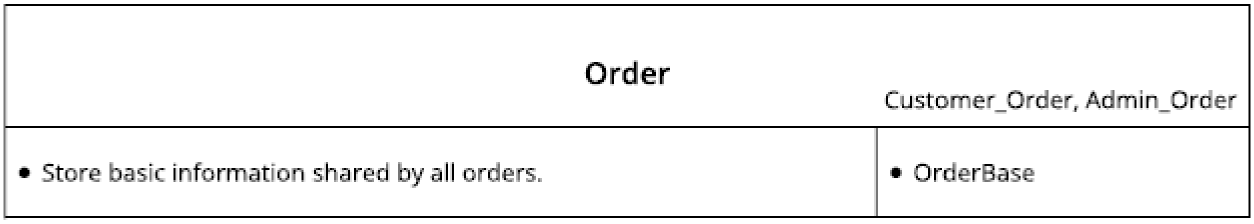
Small_Hardware	Item
• Add additional information to an Item.	



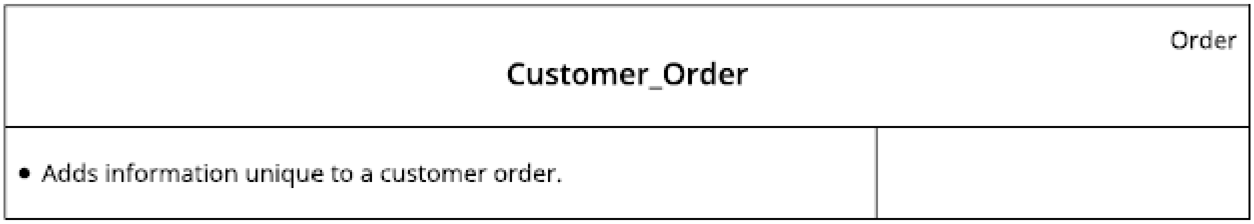
Appliance	Item
• Add additional information to an item.	



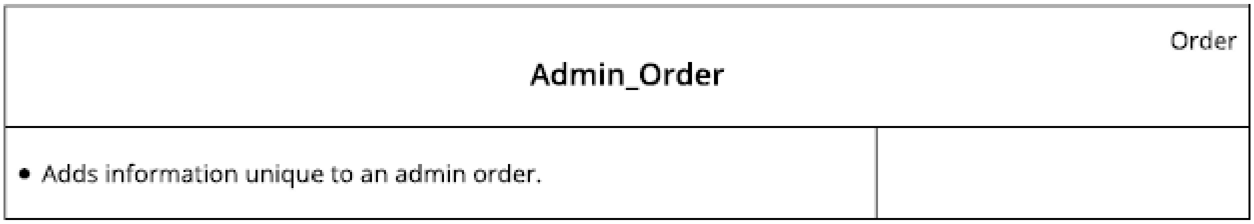
Transaction	
• Store information on a transaction.	• TransactionBase



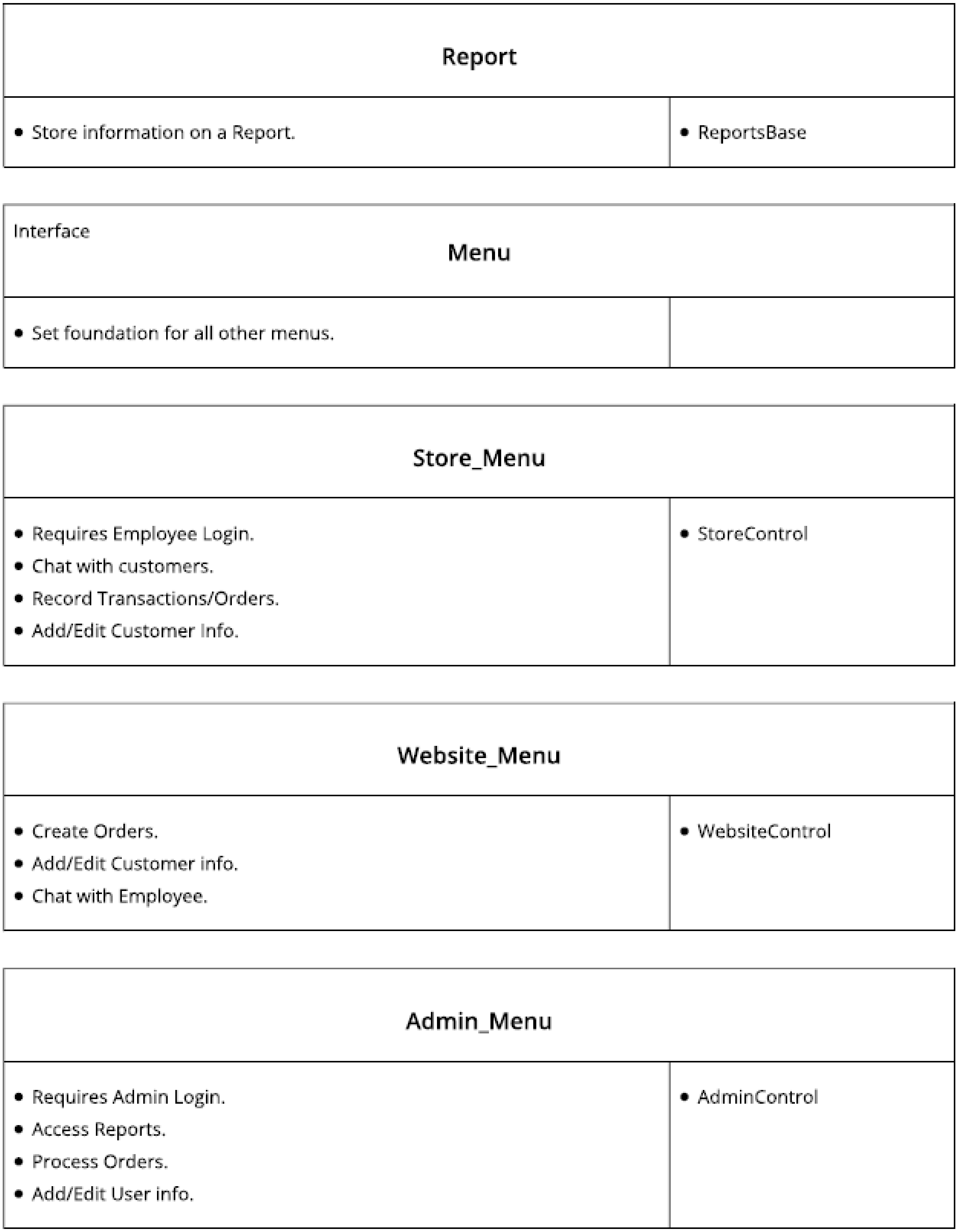
Order	Customer_Order, Admin_Order
• Store basic information shared by all orders.	• OrderBase



Customer_Order	Order
• Adds information unique to a customer order.	



Admin_Order	Order
• Adds information unique to an admin order.	



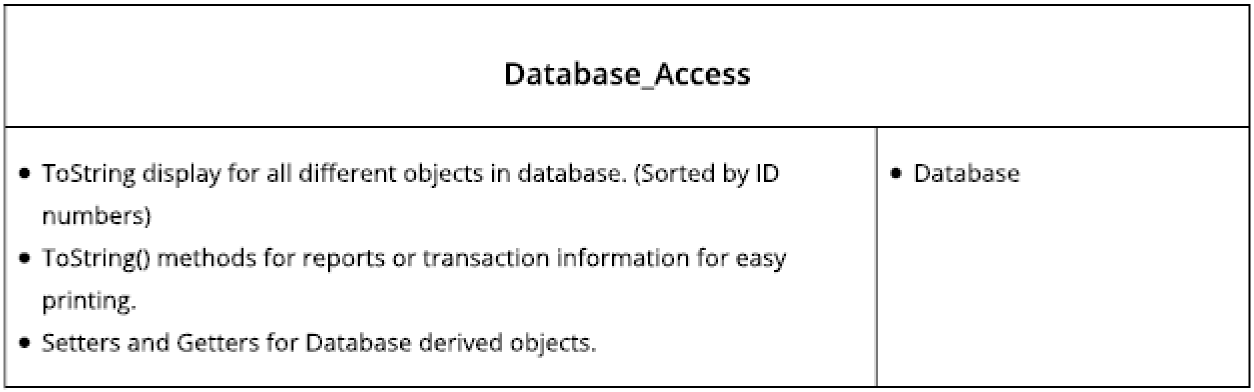
Report	
• Store information on a Report.	• ReportsBase

Interface Menu	
• Set foundation for all other menus.	

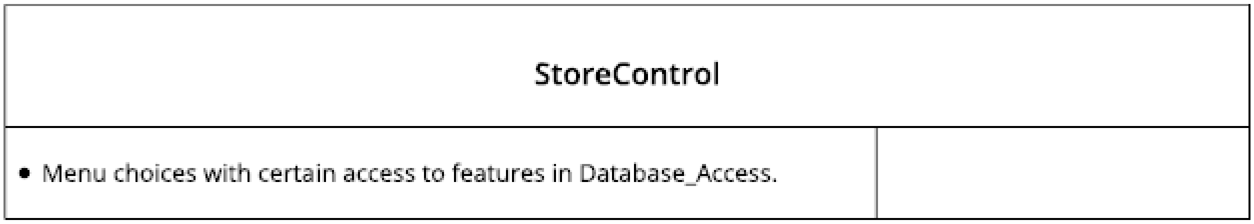
Store_Menu	
• Requires Employee Login. • Chat with customers. • Record Transactions/Orders. • Add/Edit Customer Info.	• StoreControl

Website_Menu	
• Create Orders. • Add/Edit Customer info. • Chat with Employee.	• WebsiteControl

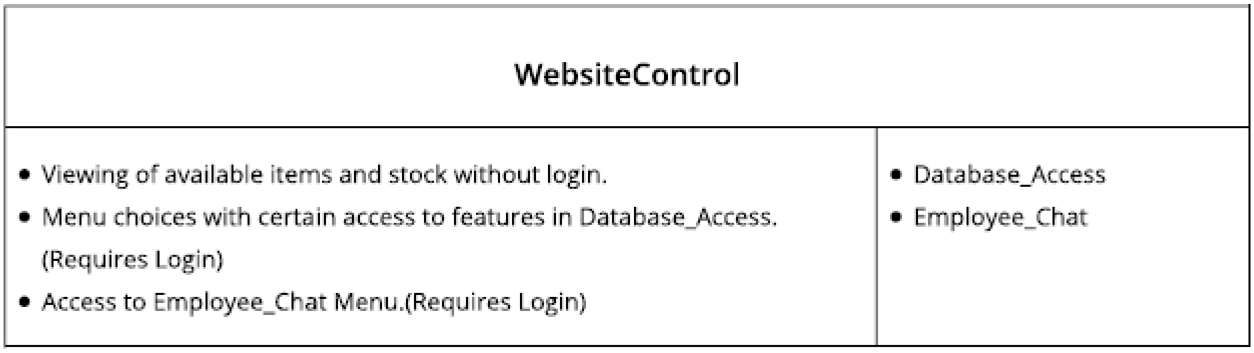
Admin_Menu	
• Requires Admin Login. • Access Reports. • Process Orders. • Add/Edit User info.	• AdminControl



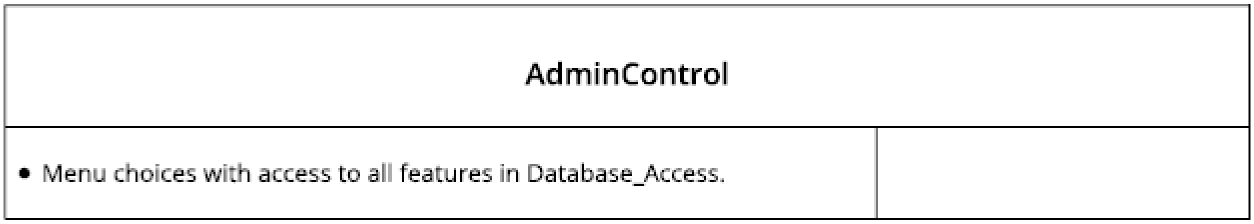
Database_Access	
• ToString display for all different objects in database. (Sorted by ID numbers) • ToString() methods for reports or transaction information for easy printing. • Setters and Getters for Database derived objects.	• Database



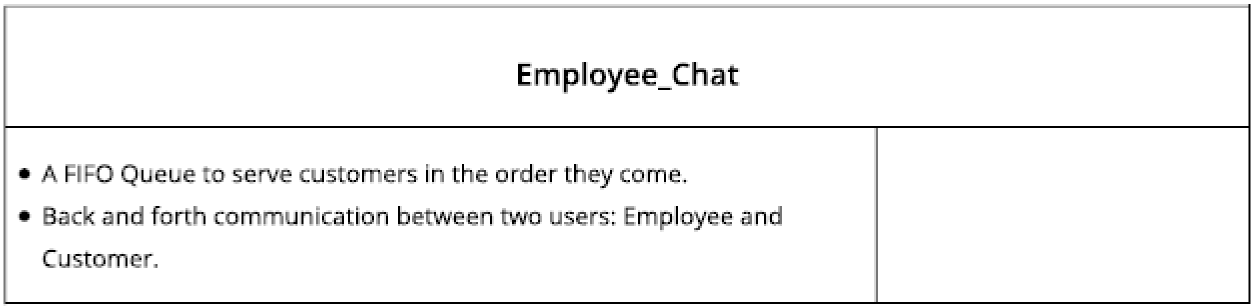
StoreControl	
• Menu choices with certain access to features in Database_Access.	



WebsiteControl	
• Viewing of available items and stock without login. • Menu choices with certain access to features in Database_Access. (Requires Login) • Access to Employee_Chat Menu.(Requires Login)	• Database_Access • Employee_Chat

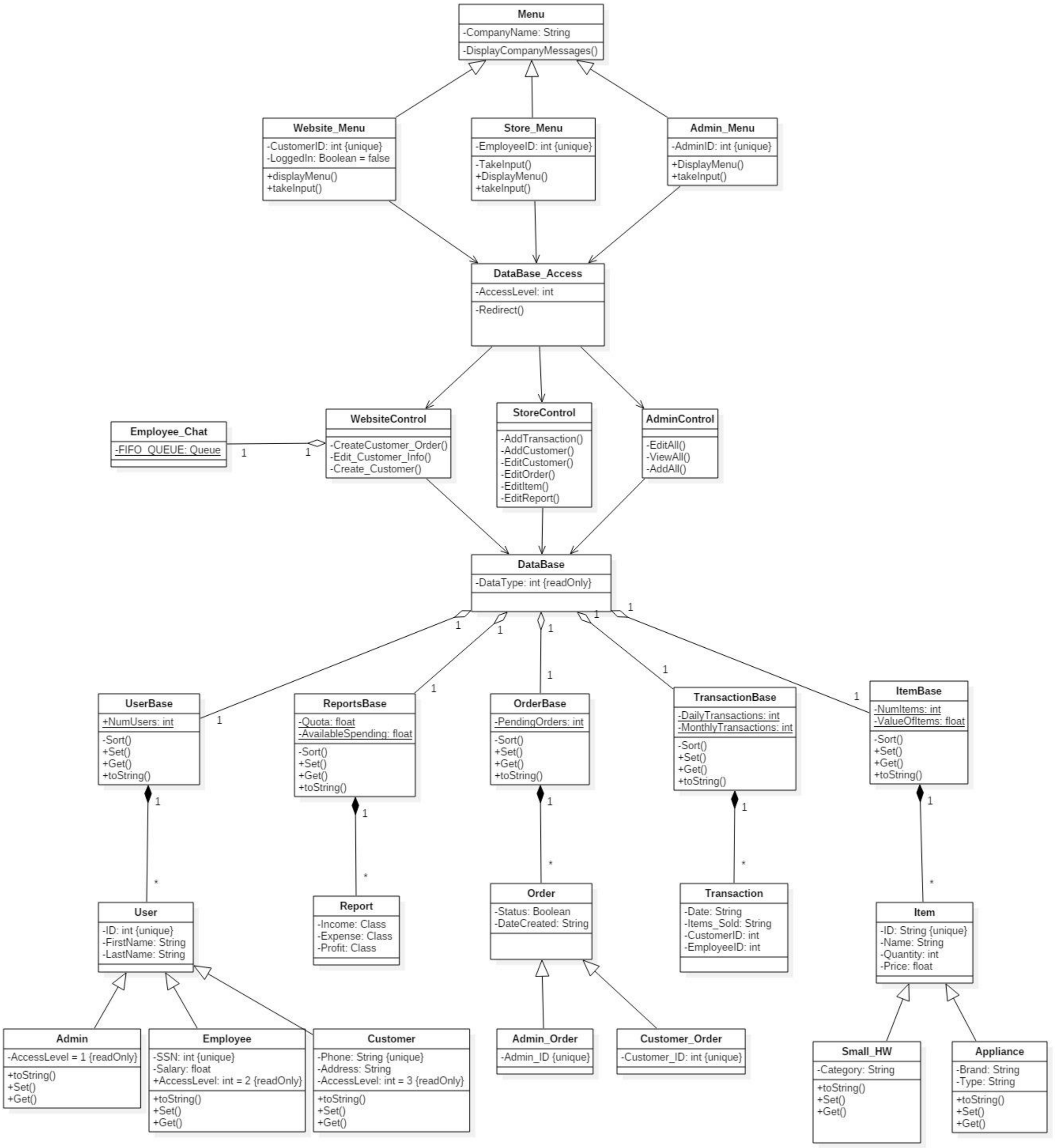


AdminControl	
• Menu choices with access to all features in Database_Access.	

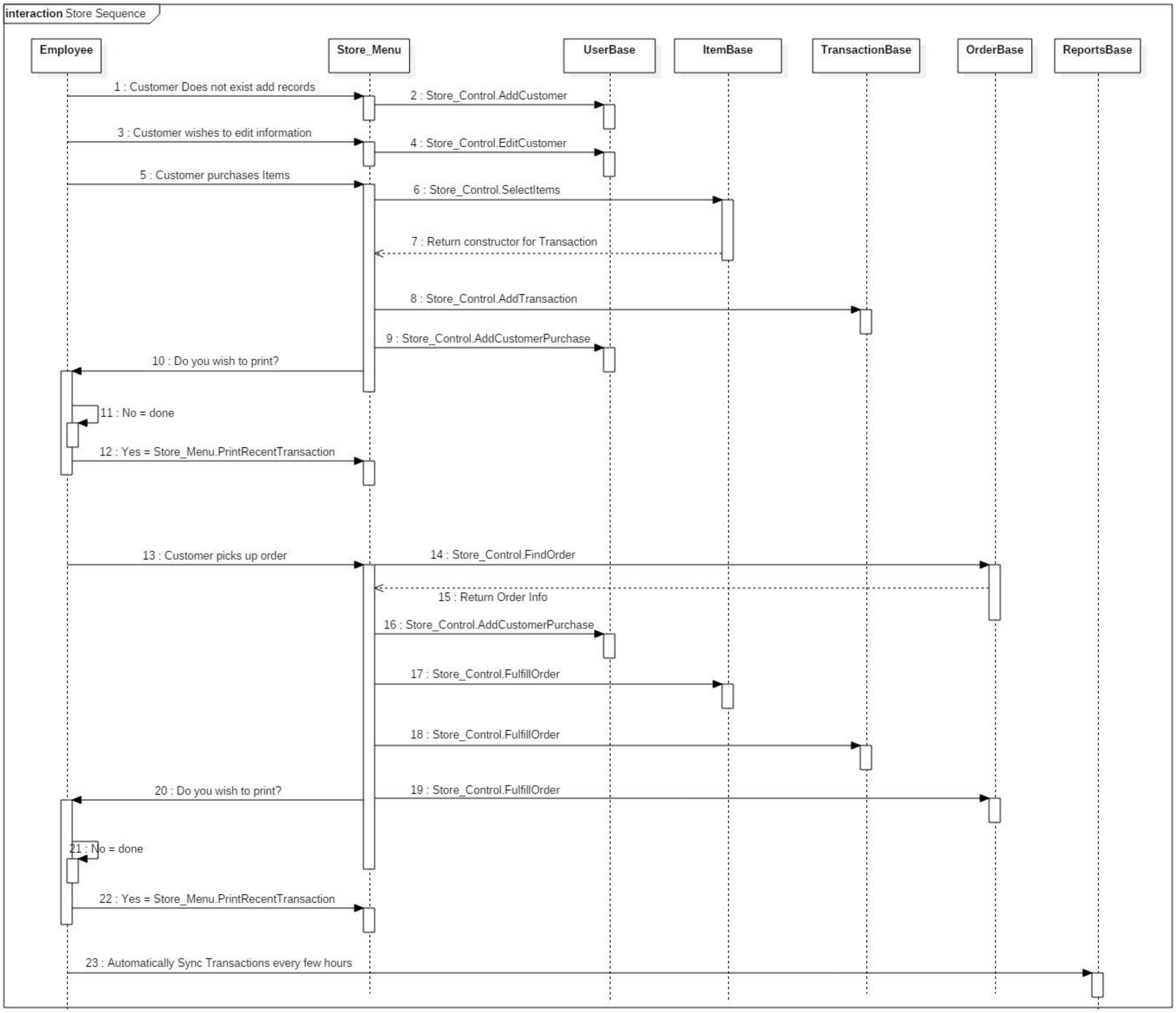


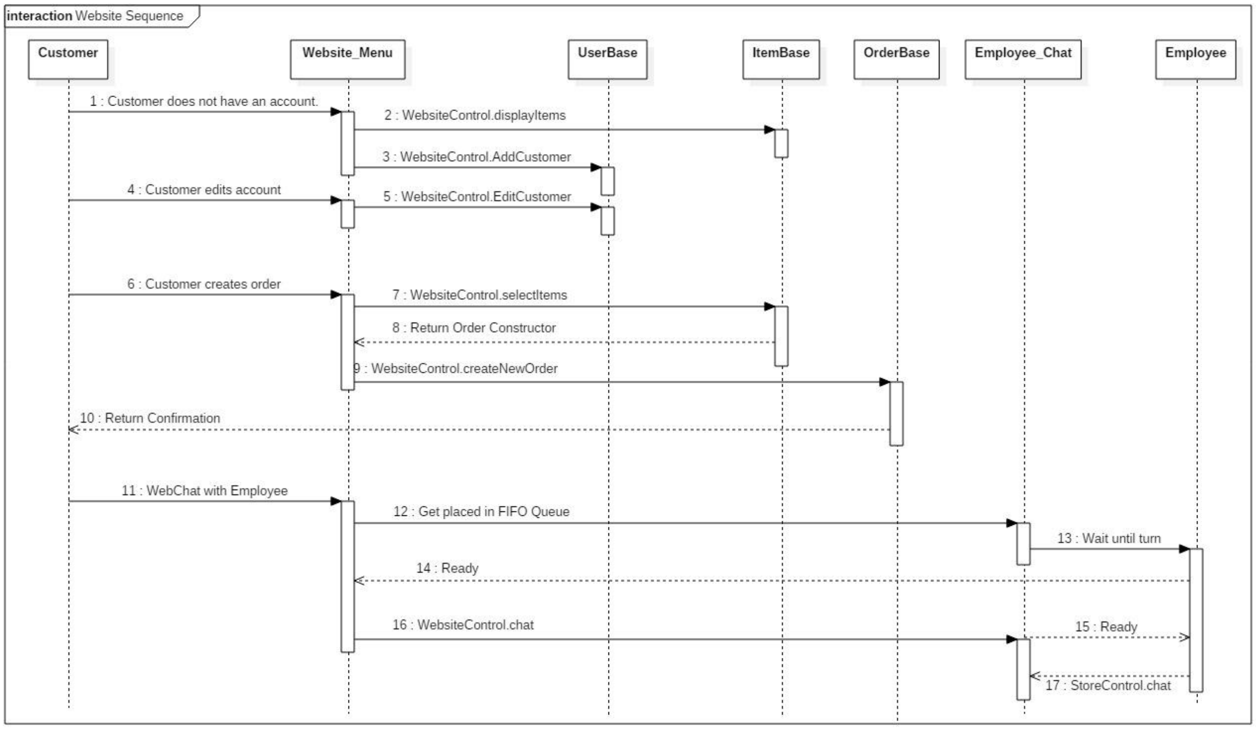
Employee_Chat	
• A FIFO Queue to serve customers in the order they come. • Back and forth communication between two users: Employee and Customer.	

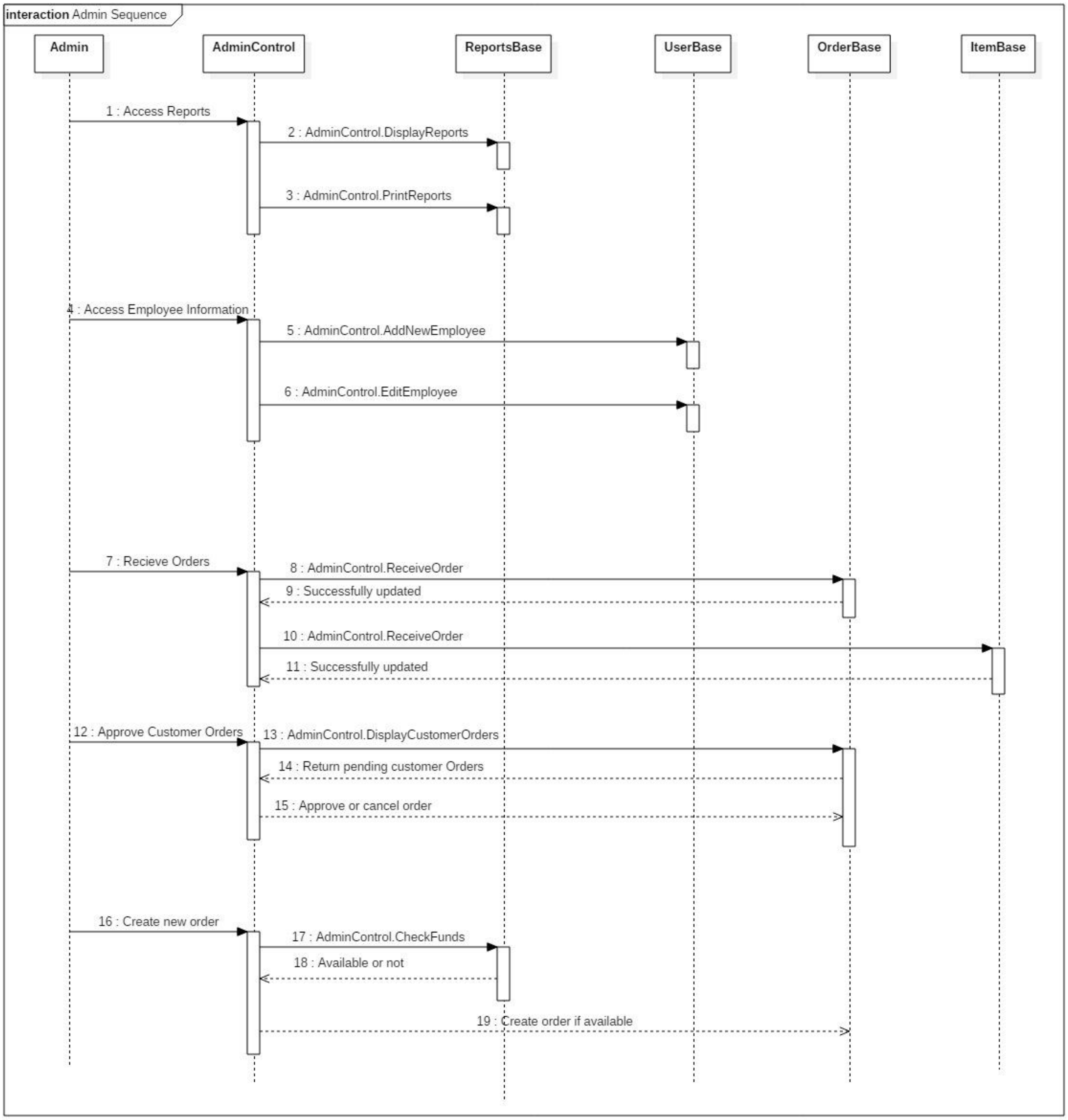
Part 4: Class Diagram



Part 5: Sequence Diagrams


interaction Store Sequence
Employee
Store_Menu
UserBase
ItemBase
TransactionBase
OrderBase
ReportsBase
1 : Customer Does not exist add records
2 : Store_Control.AddCustomer
3 : Customer wishes to edit information
4 : Store_Control.EditCustomer
5 : Customer purchases Items
6 : Store_Control.SelectItems
7 : Return constructor for Transaction
8 : Store_Control.AddTransaction
9 : Store_Control.AddCustomerPurchase
10 : Do you wish to print?
11 : No = done
12 : Yes = Store_Menu.PrintRecentTransaction
13 : Customer picks up order
14 : Store_Control.FindOrder
15 : Return Order Info
16 : Store_Control.AddCustomerPurchase
17 : Store_Control.FulfillOrder
18 : Store_Control.FulfillOrder
19 : Store_Control.FulfillOrder
20 : Do you wish to print?
21 : No = done
22 : Yes = Store_Menu.PrintRecentTransaction
23 : Automatically Sync Transactions every few hours


interaction Website Sequence
Customer
Website_Menu
UserBase
ItemBase
OrderBase
Employee_Chat
Employee
1 : Customer does not have an account.
2 : WebsiteControl.displayItems
3 : WebsiteControl.AddCustomer
4 : Customer edits account
5 : WebsiteControl.EditCustomer
6 : Customer creates order
7 : WebsiteControl.selectItems
8 : Return Order Constructor
9 : WebsiteControl.createNewOrder
10 : Return Confirmation
11 : WebChat with Employee
12 : Get placed in FIFO Queue
13 : Wait until turn
14 : Ready
16 : WebsiteControl.chat
15 : Ready
17 : StoreControl.chat


interaction Admin Sequence
Admin
AdminControl
ReportsBase
UserBase
OrderBase
ItemBase
1 : Access Reports
2 : AdminControl.DisplayReports
3 : AdminControl.PrintReports
4 : Access Employee Information
5 : AdminControl.AddNewEmployee
6 : AdminControl.EditEmployee
7 : Recieve Orders
8 : AdminControl.ReceiveOrder
9 : Successfully updated
10 : AdminControl.ReceiveOrder
11 : Successfully updated
12 : Approve Customer Orders
13 : AdminControl.DisplayCustomerOrders
14 : Return pending customer Orders
15 : Approve or cancel order
16 : Create new order
17 : AdminControl.CheckFunds
18 : Available or not
19 : Create order if available