# CrazyBid Documentation

## Logical Data Model

## Entities

**Customer**

This entity is to record the information of each customer, so that he or she may utilize the auction client.

**Attributes:**

* customerId
* auto generated Primary Key for the entity
* firstName
* stores first name of the customer
* lastName
* stores last name of the customer
* username
* stores the username of the customer, which is used to log into the client
* password
* stores the password of the customer
* phoneNumber
* stores the phone number of the customer
* email
* stores the email of the customer
* creditBalance
* stores the credit balance of the customer
* premium
* stores the membership status of the customer e.g. true if customer is a premium member, false if normal member
* addressList
* stores the addresses of the customer
* customer has a OneToMany relationship with addressList
* bidList
* stores the bids of the customer
* customer has a OneToMany relationship with bidList
* productList
* stores the products that the customer has won
* customer has a OneToMany relationship with productList
* topUpList
* stores the credit top up history of the customer
* customer has a OneToMany relationship with topUpList

**AuctionListing**

This entity is to record the information of each product listing for auction

**Attributes:**

* id
* auto generated Primary Key for the entity
* startDate
* stores the starting date of the listing
* endDate
* stores the ending date of the lising
* product
* stores the name of the product
* productDescription
* stores the description of the product
* startingPrice
* stores the starting bid price of the listing
* expectedPrice
* stores the ending bid price of the listing
* status
* stores the availability of the listing
* e.g. true if available for customers to bid, false if not available for bidding
* bidList
* stores the amounts of bids received for this listing
* AuctionListing has a OneToMany relationship with bidList
* owner
* stores the winning customer of the listing
* AuctionListing has a ManyToOne relationship with customer
* address
* stores the winning address of the listing
* AuctionListing has a ManyToOne relationship with address

**Address**

This entity stores the address of the customer that it is affiliated with.

**Attributes:**

* id
* auto generated Primary Key for the entity
* streetAddress
* stores the address of the customer
* country
* stores the country of the customer
* city
* stores the city of the customer
* postalCode
* stores the postal code of the customer
* phoneNumber
* stores the phone number of the customer
* status
* stores the status of the address
* customer
* stores the customer for the address
* Address has a ManyToOne relationship with customer
* productList
* stores the winning bids affiliated with this address
* Address has a OneToMany relationship with productList

**Bid**

This entity stores Bid information for each bidding transaction.

**Attributes:**

* id
* auto generated Primary Key for the entity
* bidTime
* stores the time where the bid was completded
* bidAmount
* stores the bid amount
* auctionListing
* stores the listing that is affiliated with the bid
* Bid has a ManyToOne relationship with AuctionListing
* customer
* stores the customer that is affiliated with the bid
* Bid has a ManyToOne relationship with customer

**CreditPackage**

This entity stores the information for each credit package, which are for customers to purchase for usage on the bidding client.

**Attributes:**

* id
* auto generated Primary Key for the entity
* packageName
* stores the name of the credit package
* amount
* stores the amount of credits for the credit package
* soldAmount
* stores the number of times the credit package was sold
* status
* stores the availability of the credit package
* transactionList
* stores the transaction that the credit package is affiliated with
* CreditPackage has a OneToMany relationship with transactionList

**Staff**

This entity stores the information for CrazyBid’s staff

**Attributes:**

* staffId
* auto generated Primary Key for the entity
* firstName
* stores first name of the staff
* lastName
* stores last name of the staff
* username
* stores the username of the staff, which is used to log into the client
* password
* stores the password of the staff
* staffAccessRight
* stores the access rights of the staff

**TopUpTransaction**

This entity stores the credit top up transactions of a customer

**Attributes:**

* id
* auto generated Primary Key for the entity
* createdOn
* stores the date of the transaction
* customer
* stores the affiliated customer of the transaction
* creditPackage
* stores the affiliated creditPackage of the transaction

## Physical Architecture

### Customer Client

**MainApp**

This is the main menu that is displayed when the customer client is first

Includes methods to:

* Login
  + Allows user to log in with a registered account
* Register
  + User can register for an account with a unique username and password
* Exit
  + Exits the application

**CustomerBidModule**

This module handles the bidding related operations for the customer

Includes methods to:

* View all auction listings
  + Includes refreshing of the list
* View details of an auction listing
  + If the customer has enough credits, he can place a bid for an item
* View won bids
  + If there are any won bids, customer can select the address for the item to be sent to.

**CustomerCreditModule**

This module handles the credit related operations for the customer

Includes methods to:

* View credit balance
* View transaction history
  + Includes both bidding history and credit top up history
* Purchase credits

**CustomerProfieModule**

This module handles the profile related operations for the customer

Includes methods to:

* View customer profile
  + Allows the customer to view the details of his profile
* Update profile
  + Allows customer to update his profile, including the changing of password
* Create address
  + Creates an address for the customer, to be used as a delivery address for won biddings
* View address details
  + After viewing the address, customer has the option to update or delete the address
* View all addresses
  + Allows customer to view all his saved addresses

### Staff Client

**MainApp**

This is the main menu that is displayed when the staff client is first started

Includes methods to:

* Login
  + Allows a staff to log in with a registered account
* Exit
  + Exits the application

**AdminOperationModule**

This module handles the admin related operations for admin staff.

Includes methods to:

* Create new staff
  + Allows the admin staff to create a new staff
* View employee details
  + Allows the admin staff to view and update the details of the staff, or delete the staff record
* View all employees
  + Allows the admin staff to view all employee records in the database

**FinanceStaffOperationModule**

This module handles the finance related operations for finance staff.

Includes methods to:

* Create credit package
  + Allows finance staff to create new credit packages for customers to purchase
* View credit package details
  + Allows the finance staff to view and update the credit package, and to delete or disable the credit package
* View all credit packages
  + View all credit packages, active or inactive

**SalesStaffOperationModule**

This module handles the sales related operations for sales staff.

Includes methods to:

* Create auction listing
  + Allows the sales staff to create a product listing to be available for bidding
* View auction listing details
  + Allows the sales staff to view and update the auction listing, or delete it from the database
* View all auction listings
  + Allows the sales staff to view all auction listings, active or inactive
* View all auction listings with bids but below reserved price
  + Allows the sales staff to view any auction listings that received bids but are below reserved price. Sales staff will then have to manually assign the winning bid to a customer