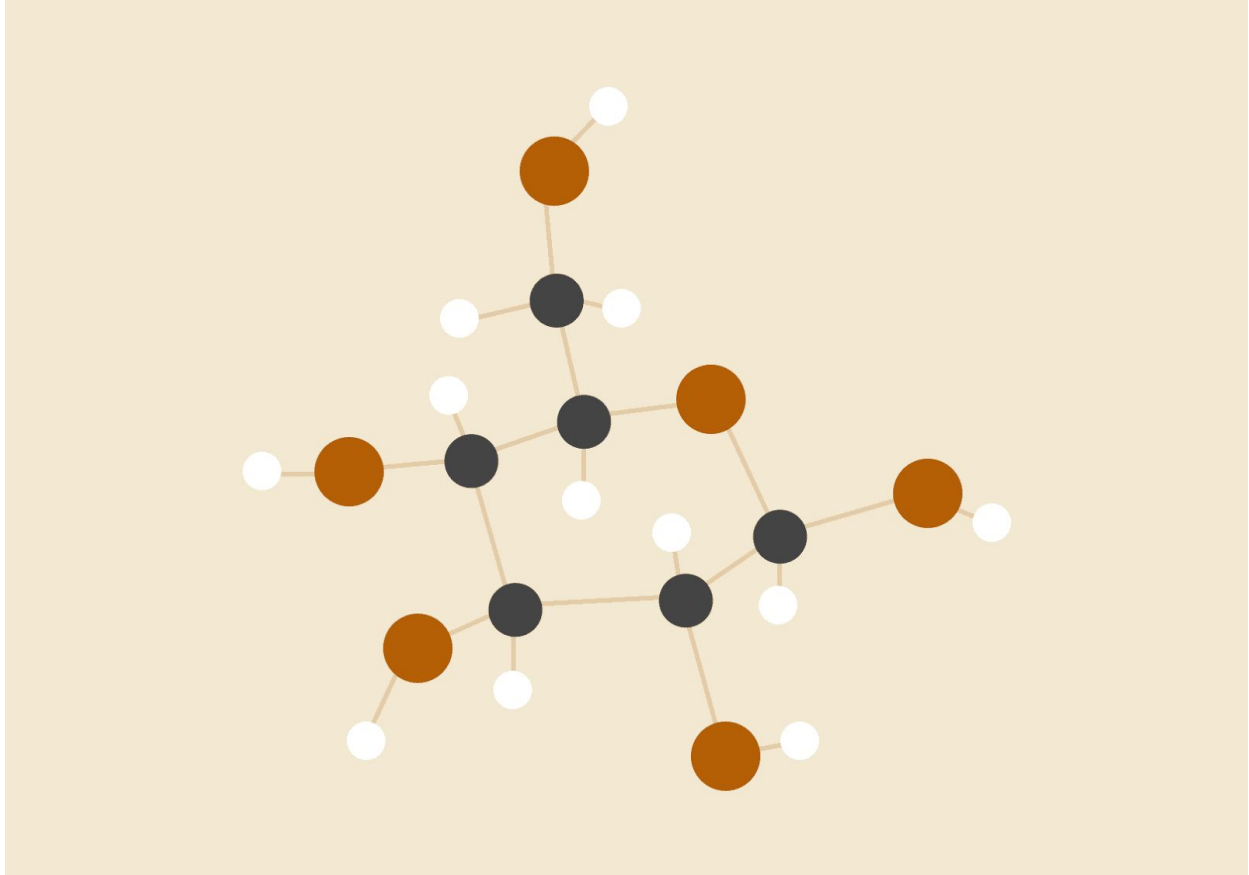


Phase 1: Report

CS6400: Team 105



Jung, Sungjin; sjung93@gatech.edu

Kazkayasi, Mehmet O; mkazkayasi3@gatech.edu

Katnis, Nichole Marlene; nkatnis3@gatech.edu

Khodke, Pallavi Vijay;

February 11, 2018

Online Master of Science in Computer Science
Georgia Tech

TABLE OF CONTENTS

[GTBay Data Types](#)

[GTBay Constraints](#)

[GTBay Task Decomposition with Abstract Code](#)

DATA TYPES

USER

Attribute	Data Type	Nullable
username	String	Not Null
password	String	Not Null
first_name	String	Not Null
last_name	String	Not Null
listed	Integer	Not Null
sold	Integer	Not Null
purchased	Integer	Not Null
rated	Integer	Not Null

ADMIN USER

Attribute	Data Type	Nullable
username	String	Not Null
position	String	Not Null

ITEM

Attribute	Data Type	Nullable
item_id	Integer	Not Null
item_name	String	Not Null
description	String	Not Null
active	Boolean	Not Null
condition	Integer	Not Null
returnable	Boolean	Not Null
starting_bid	Decimal(19,2)	Not Null
get_it_now_price	Decimal(19,2)	Not Null
auction_start_date_time	DateTime	Not Null
auction_end_date_time	DateTime	Not Null
sale_price	Decimal(19,2)	Nullable
buyer_name	String	Nullable

CATEGORY

Attribute	Data Type	Nullable
category_name	String	Not Null

total_items	Integer	Not Null
min_price	Decimal(19,2)	Not Null
average_price	Decimal(19,2)	Not Null
max_price	Decimal(19,2)	Not Null

RATING

Attribute	Data Type	Nullable
rating_date_time	String	Not Null
number_of_stars	Integer	Not Null
comment	String	Null

BID

Attribute	Data Type	Nullable
time_of_bid	DateTime	Not Null
bid_amount	Decimal(19,2)	Not Null

BUSINESS LOGIC CONSTRAINTS

GTBay User

- Users who are new to GTBay must register first.
- Users who have an existing GTBay account will not be able to register.
- User information submitted at registration cannot be changed.

GTBay Item

- No other attribute except **Description** can have any change after first register.
- Only Item Winners can review Item?
- Relisted items are considered new items with new Item_ID
- Minimum Sale Price has to be lesser than Get It Now Price

Auction

- Bids can't be lower than **Start Auction Bidding at**.
- Bids can't be lower than (highest bid on the item + \$1).
- Bids can't be higher than Get It Now Price (if specified).
- New bids are not accepted after Auction End Date.

Ratings

- Users can only rate an item once.
- Users cannot rate items listed by themselves.

ASSUMPTIONS

1. Auctions that end when a user chooses the “Get it Now” option will populate the winner and save the date and time of the transaction as the Auction_End_Date_Time.
2. Auction winners will be calculated and stored when a user attempts to view the results of the auction after the auction end date. When a user want to see the Auction Results, all items are filtered through their Auction End Date & Time. If Current Time is ahead of their property, this means their Auction ended in one of three ways:
 - a. Some User uses Get It Now function: Even though the Auction End Date & Time is not reached yet, this will change the Auction End Date & Time attribute of the Item to the time Get It Now button is clicked. So, it will be among Items of which Auction End Time & Date < Current Time.
 - b. Some User wins the Auction: We will decide on the winner based on the last bid made, which has to be the highest bid.
 - c. No User made any bid or highest bid is below Minimum Sale Price: In that case, no user purchased the item. So, there is no auction winner.
3. An auction will expire/become inactive after the Auction_End_Date_Time lapses. A closed auction will be filtered from item search results.
4. Inactive offers will not accept new bids.
5. Items that have no winner at the auction close must be relisted before a new auction for the item can begin.
6. Relisted items are considered new items, independent of any previous listings.

TASK DECOMPOSITION with ABSTRACT CODE

Login

Task Decomposition

Lock Types: No lock is needed.

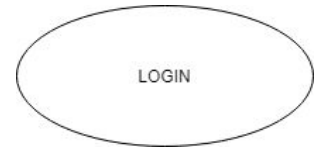
Number of Locks: -

Enabling Conditions: None

Frequency: Around 200 logins per day

Consistency (ACID): not critical, order is not critical.

Sub Tasks: Mother task is not needed. No decomposition needed.



Abstract Code

- User enters *username* ('\$Username'), *password* ('\$Password') input fields.
- When **Login** button is clicked:
 - If User record is found but `User.password != '$Password'`:
 - Go back to **Login** form, with error message.
 - If User record did not found: Go back to **Login** form, with error message.
 - Else:
 - Store login information as session variable '\$Username'
 - Go to **Main Menu**
- Upon click **Register** button - Jump to **Register** task

Register

Task Decomposition

Lock Types: Read-lock on [User](#) table

Number of Locks: Single

Enabling Conditions: None, order is not critical.

Frequency: Around 5 registers per day

Consistency (ACID): Not critical, order is not critical

Sub Tasks: Mother task is not needed. No decomposition needed.



Abstract Code

- User enters *First Name* ('\$FName'), *Last Name* ('\$LName'), *Username* ('\$Username'), *Password* ('\$Password') and *Confirm Password* ('\$CPassword') input fields.
- When **Register** button is clicked:
 - If any of the fields is not filled in: Display **Register** form, with error message.
 - If '\$Password' != '\$CPassword': Display **Register** form, with error message.
 - If '\$Username' is used by another User: Display **Register** form, with error message.
 - Else, store new user information in User table: Go to **Log In** form
- When **Cancel** button is clicked: Display **Login** form.

New Item For Auction

Task Decomposition

Lock Types: Read-lock on **Item** table

Number of Locks: Single

Enabling Conditions: Trigger by successful login.

Frequency: Around 10 item added per day

Consistency (ACID): Not critical, order is non-critical

Sub Tasks: Mother task is not needed. No decomposition needed.



Abstract Code

- Get categories from **Category**.Category_Name values in dropdown list.
- User:
 - Enters *Item Name* (**'\$ItemName'**), *Description* (**'\$Description'**), *Start Auction Bidding* (**'\$AuctionStartPrice'**), *Minimum Sale Price* (**'\$MinSalePrice'**), *Minimum Sale Price* (**'\$AuctionStartPrice'**), and optionally *Get It Now Price* (**'\$GetItNowPrice'**).
 - Selects *Category* (**'\$Category'**), *Condition* (**'\$Condition'**), *Auction Length* (**'\$AuctionLength'**) input fields from dropdown lists,
 - Chooses to click or not on *Returns Accepted* (**'\$ReturnsAccepted'**) field.
- When **List My Item** button is clicked:
 - If any of the fields except *Get It Now Price* field is not filled in: Display **New Item for Auction** form, with error message.
 - If **'\$GetItNowPrice' < '\$MinSalePrice'**: Display **New Item for Auction** form, with error message.
 - Assign **Item**.Item ID, Assign 1 to **Item**.Active.
 - Calculate **Item**.AuctionEndDate using Current date and time along with **'\$AuctionLength'**
 - Get the User using **'\$Username'** and increment **User**.Listed by 1
 - Get Category that Item is in using **'\$Category'** and :
 - Increment **Category**.TotalItems by 1.
 - If the **Item**.GetItNowPrice < **Category**.MinPrice: Assign **Category**.MinPrice = **Item**.GetItNowPrice
 - If **Item**.GetItNowPrice > **Category**.MaxPrice: Assign **Category**.MaxPrice = **Category**.GetItNowPrice
 - Modify **Category**.AveragePrice using **Category**.AveragePrice, **Category**.TotalItems and **Item**.GetItNowPrice
 - Store new item information in **Item** table
 - Display **Main Menu** form.
- When **Cancel** button is clicked: Display **Main Menu** form.

Item Search

Task Decomposition

Lock Types: No locks needed

Number of Locks: -

Enabling Conditions: Trigger by successful login.

Frequency: Around 30 searches per day

Consistency (ACID): not critical, order is not critical

Sub Tasks: Mother task is not needed. No decomposition needed.

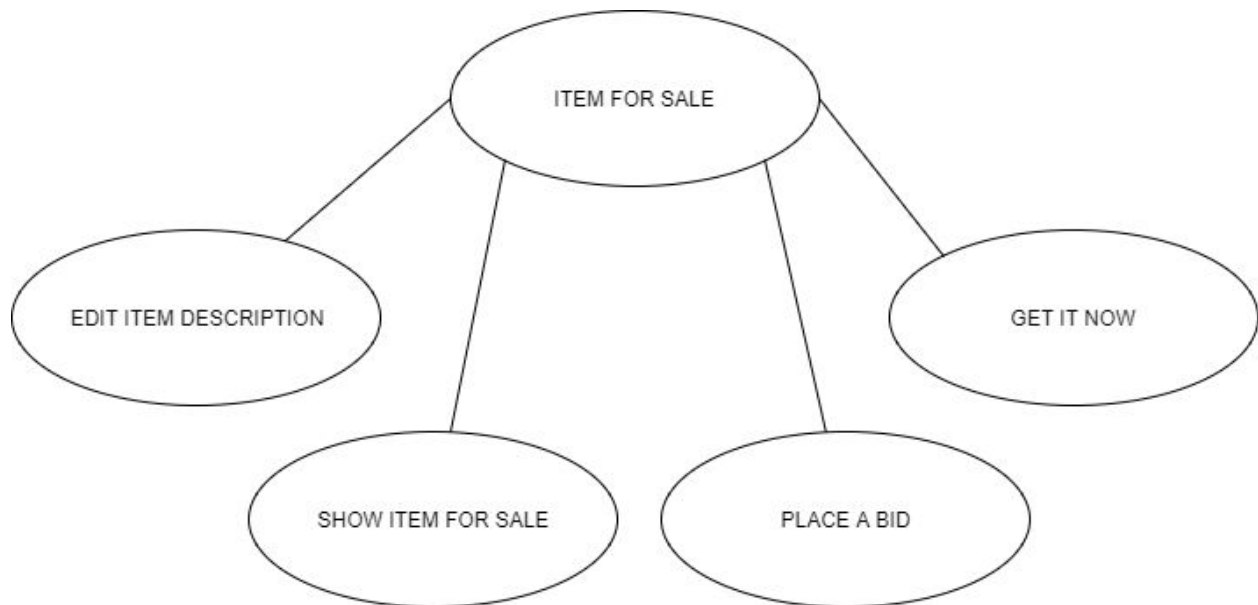


Abstract Code

- Get **Category**.Category_Name values along with “**Any**” option in dropdown list.
- User:
 - Enters *Keyword* (“**Keyword**”),
 - Optionally user selects: *Minimum Price* (“**MinPrice**”), *Maximum Price* (“**MaxPrice**”), *Category* (“**Category**”) and *Condition At Least* (“**ConditionAtLeast**”)
- When **Search** button is clicked:
 - Find Items with **Item**.Active = 1 in the selected category which has ‘**Keyword**’ in ‘**ItemName**’ or ‘**Description**’
 - Optional search criteria used to further filter results
 - that have a current high Bid of at least ‘**MinPrice**’ and at most ‘**MaxPrice**’
 - better ‘**Condition**’ than ‘**ConditionAtLeast**’
 - For each Item:
 - Get **Item**.ItemId, **Item**.ItemName, **Item**.GetItNowPrice, **Item**.AuctionEndTime
 - Get **Bid** with max **Bid**.Time associated with Item and as Current Bid,
 - Get Bid Owner associated with Current bid as High Bidder
 - Display form **Search Results**
 - When any **Item Name** button is clicked: Display **Item for Sale** form.
- When **Back to Search** button is clicked: Display **Item Search** form.

Item for Sale

Task Decomposition



Lock Types: Read-lock on [Bid](#) table and [Item](#) table

Number of Locks: Several different schema constructs are needed.

Enabling Conditions: Trigger by clicking on an Item in Item Search results

Frequency: Around 50 items are seen per day

Consistency (ACID): critical, order matters

Sub Tasks: Not all tasks can be done in parallel. Mother task is required to coordinate subtasks. Order is not necessary.

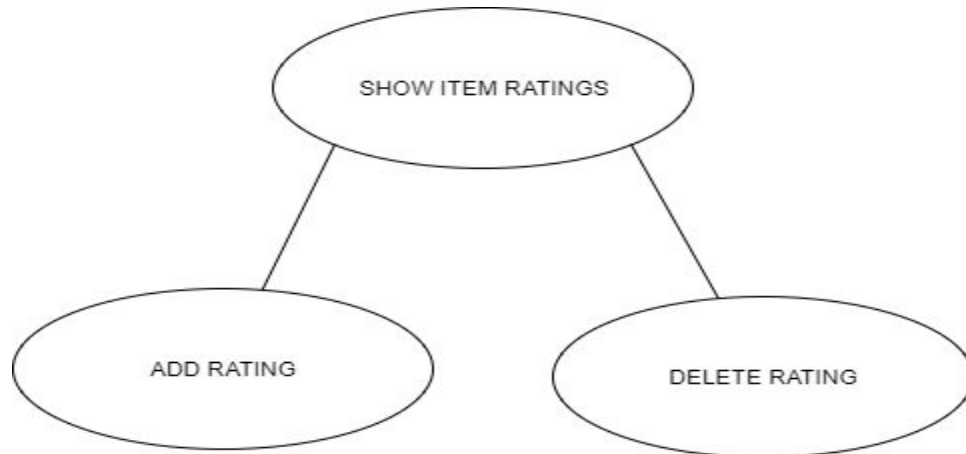
Abstract Code

- Get [Item](#).ItemName, [Item](#).ItemID, [Item](#).Description, [Item](#).AuctionEndDate, [Category](#).Category_name that [Item](#) is related, [Item](#).Condition, [Item](#).GetItNowPrice (if not NULL), get top 4 bids with bids sorted by descending [Bid](#).Time that [Item](#) is related as well as usernames associated with the [Bid](#)
- Show Edit Description button if '\$UserID' belongs to the seller.
- Calculate ('\$MinBid'), by adding one dollar on the highest bid.
- Show **Item for Sale** form
- Upon clicked **Get It Now** button:
 - Assign [Item](#).Active = 0
 - Assign [Item](#).SalePrice = [Item](#).GetItNowPrice
 - For user that listed the item increment [User](#).Sold by 1
 - For user buying the product increment [User](#).Purchased by 1

- Modify `Item.AuctionEndDate` as Current Time.
 - Assign `Item.SalePrice = Item.GetItNowPrice`
 - Assign `Item.BuyerName = '$UserID'`
- Upon **View Ratings** button is clicked: Display form **Item Ratings**
- User can choose to bid on item. So, User enters Your Bid (`'$YourBid'`). Upon clicked **Bid On This Item** button:
 - If `'$YourBid'` is not populated Display Item For Sale form with error message.
 - If `'$YourBid'` is not larger than one dollar added to highest bid Item has: Display Item For Sale form with error message.
 - If `'$YourBid'` is greater than or equal to the `Item.GetItNowPrice` Display **Item For Sale** form with error message.
 - Else, create a new bid for the item. Assign `Bid.BidAmount = '$YourBid'` and `Bid.TimeofBid = Current Time`.
 - Display **Item For Sale** form.
- Upon **Edit Description** button is clicked: Enable textbox containing `Item.Description` for edit.
- When **Cancel** button is clicked: Display **Search Results** form.

Item Ratings

Task Decomposition



Lock Types: Read-lock on [Rating](#)

Number of Locks: Single

Enabling Conditions: Trigger by clicking on View Ratings button on Item For Sale form

Frequency: Around 10 item ratings are seen per day and 2 ratings are given

Consistency (ACID): not critical, order is not critical

Sub Tasks: Mother task is required to coordinate subtasks. Decomposition is needed: **Add Rating** and **Delete Rating**.

Abstract Code

- Find and display [Item](#).ItemName, [Item](#).ItemID,
- Get ratings that are associated with the Item and calculate the average rating
- Get all Rating information related to the Item: [Rating](#).RatingTimeDate, [Rating](#).Comment, and [Rating](#).NumberOfStars sorted by descending [Rating](#).RatingDateTime
- Check '\$UserID' has a Rating and if it has: Display **Delete My Rating** button on '\$UserID's rating
- Upon **Delete My Rating** button is clicked: Remove the Rating belongs to '\$UserID' on
- User fills Comments ('\$Comment') and selects My Rating ('\$Rating').
- Upon **Rate This Item** button is clicked:
 - Store [Rating](#).RatingTimeDate as Current time, [Rating](#).Comment, and [Rating](#).NumberOfStars in Rating table.
 - Display **Item Ratings** form.
- Upon **Cancel** button is clicked: Display **Item for Sale** form.

Auction Results

Task Decomposition

Lock Types: No locks needed

Number of Locks: -

Enabling Conditions: Trigger by successful login.

Frequency: Around 2 times per day

Consistency (ACID): not critical, order is not critical

Sub Tasks: Mother task is not needed. Decomposition is not needed



Abstract Code

- Fetch Items that Item.AuctionEndDate has passed.
- Set [Item](#).Active = 0 for returned Items
- If [Item](#).SalePrice is NULL and Item has bids and highest bid is greater than or equal to [Item](#).MinimumSalePrice:
 - Set [Item](#).SalePrice = Current Bid.
 - Set [Item](#).Buyer = Current Bid's Owner
- Display [Item](#).ItemID, [Item](#).ItemName, [Item](#).SalePrice ('-' in case NULL), and [Item](#).Buyer ('-' in case NULL)
- Display **Auction Results** sorted by descending AuctionEndDate
- Upon ***Done*** button is clicked: Display **Item for Sale** form.

Category Report

Task Decomposition

Lock Types: No locks is needed

Number of Locks: -

Enabling Conditions: None

Frequency: Around 2 times per day

Consistency (ACID): not critical, order is not critical

Sub Tasks: Mother task is not needed. No decomposition needed.



Abstract Code

- Fetch Categories and sort by `Category.CategoryName`
- Display `Category.Listed`, `Category.MinPrice`, `Category.TotalItems`, and `Category.MaxPrice`, `Category.AveragePrice`.
- Upon clicked Done button: Display Main Menu form.

User Report

Task Decomposition

Lock Types: No locks is needed

Number of Locks: -

Enabling Conditions: None

Frequency: Around 2 times per day

Consistency (ACID): not critical, order is not critical

Sub Tasks: Mother task is not needed. No decomposition needed.



Abstract Code

- Fetch Users and sort by `User.Listed`.
- Display `User.Listed`, `User.Purchased`, `User.Sold`, and `User.Rated`.
- Upon clicked Done button: Display Main Menu form.