

CS6400 Database Project Spring Semester 2018



Purpose of the Project

Analyze, specify, design, implement, document and demonstrate an information system application to support a new online auction system called GTBay. GTBay is similar to, but simpler than, existing online auction sites such as eBay. You are required to use methodology presented in the videos. The system should be implemented using a DBMS that supports standard SQL queries. The professor must approve other alternative implementations. *In no circumstances can you use a tool that automatically generates SQL or automatically maps programming objects into the database.*

Deliverables

(There will be more specific information posted about deliverables later)

Phase I

The deliverables include:

1. Enhanced Entity Relationship (EER) Diagram
2. Information Flow Diagram
3. Task Decomposition Diagram
4. Abstract Code
5. A list of logical constraints that will be enforced. Do not include any constraints that can be shown in the ER diagram, but rather semantic, business logic related constraints. **You are required to include at least five constraints, although a fully-specified system will have more than that. Constraints that can be specified directly using ER notation will not count toward the five required.**
6. Any assumptions made including explanations. **In particular, include a paragraph about how you will calculate auction winners. For example, will winners be calculated once and stored explicitly? Or will winners be calculated from the bid history on the fly? When will winners be calculated? There are multiple ways to implement this functionality. For this phase, you should explain conceptually how you plan on implementing it.**

Notes:

1. The EER must capture the constraints of the system as much as possible whenever applicable, i.e. total participation, super/sub class, weak entities.
2. The design of your system must satisfy all the constraints. You are allowed to make up additional assumptions and constraints as long as they do not conflict with the specified constraints and requirements. If possible, those additional assumptions and constraints should be included in the ER diagram.

Phase II

1. Cover Page
2. Copy of the ER Diagram (either from phase I (with any revisions) or from the solution)
3. Copy of the Information Flow Diagram from phase I (either from phase I (with any revisions) or from the solution)
4. Relational Schema Diagram (with primary and foreign keys identified, referential integrity is shown by arrows)
5. Create Table statements, including domain constraints, integrity constraints, primary keys, and foreign keys
6. Abstract Code incorporating SQL statements for each task (*follow the template in the phase II design methodology*)

Notes: A set of SQL statements may be required in order to complete one task. However, in such cases, the last SQL statement should show the output according to the specification. If mentioned, the returned tuples must be ordered according to the specification. The last SQL should resemble the output as much as possible.

Phase III

Prior to the demo, the database has to be populated with a dataset. We will provide more details for what data needs to be present before demonstrations begin.

Deliverables for Phase 3 are:

1. Copy of the Create Table statements from phase II (with any revisions)
2. Source code (documented) for your system
3. A functional application with embedded SQL statements that accesses your database
4. A system demo to one of the TAs.

GTBay Overview

GTBay is a simple online auction site. The site supports two kinds of users: **Regular Users** and **Administrative Users**. Both kinds of users can list items they have for sale, search for items, bid on items, rate items, comment on items, and purchase items. Administrative users have the additional option of viewing two administrative reports.

The following sections contain a functional description of the GTBay application along with some screen mockups. *The user interfaces depicted in this project description merely serve as examples to guide your thinking.* Your project's interface may look different. For example, you might choose to split up some interfaces we have shown on a single screen into multiple screens. You might choose to use popup windows instead of refreshing the page. A different user interface is acceptable as long as your application supports the same functionality as described below. You may implement the project as a traditional standalone application (e.g., using Java GUIs) or as a web application (e.g., using a web scripting language like PHP). All screens are static and do not need to be dynamically updated if they are already loaded and other users make updates to the database.

Logging In to GTBay

Figure 1 shows the GTBay login screen. All users are uniquely identified by his or her **Username**. Providing a valid **Username** and **Password** combination will log the user into the system. Providing invalid login credentials should display an error message and return the user to the login screen.

After a user logs in to the system, he or she should be presented with a menu with the following options:

- Search for Items
- List an Item for Sale
- View Auction Results



Figure 1 - GTBay Login Screen

Administrative Users have an additional attribute that your system should keep track of: the **Position** of the administrator in the company. There is not a fixed list of positions, so you should allow any value for this field. When the administrator logs in, his or her **Position** should be displayed somewhere on the screen. Also, administrators have two additional menu options:

- View Category Report
- View User Report


New User Registration

Users who are new to GTBay must register first. A *Register* button is provided directly on the login page. Clicking this button displays the new user registration form. Figure 2 shows the new user registration form. All fields on this form are required and cannot be changed at a later time. (Users who forget their passwords are just plain out of luck!)

After the user clicks **Register**, the system should verify that all fields are filled in, that the **Username** has not already been registered, and that the **Password** and Confirm **Password** fields are equal. If any of these validations fail, the user should be returned to this screen to make corrections. The user should be provided with meaningful error messages so he or she knows what to correct.

Users who register through this interface do NOT have administrative privileges. The only way a user receives administrative rights is if the database administrator gives these permissions behind the scenes.

GTBay New User Registration

 *It's not junk if someone will pay for it.*

First Name

Last Name

Username

Password

Confirm Password

All fields are required

Returns to the login screen

Figure 2 – GTBay New User Registration Form

Selling Items on GTBay

A major feature of GTBay is the ability for users to list items they would like to sell. Figure 3 shows the New Item for Auction form. The user enters important information about the item so that other users can decide whether or not to bid on it. The fields include the **Item Name**, a **Description** of the item, an item **Category**, the **Condition** of the item, and whether or not the item is **Returnable**. The user also enters price information including the **Starting Bid**, the **Minimum Sale Price** that the user will accept, the **Auction Length**, and an optional **Get It Now Price**. All price fields are in US Dollars and should only allow two decimal places. All fields are required to be filled in except for the **Get It Now Price**.

The list of Categories is fixed by the system. A dropdown list is provided to the user so that he or she does not have to type in the **Category** name. Even though the categories are fixed, it should be possible for the database administrator to change category names or add new ones behind the scenes. *In other words, do not hard code these names into your application.* The available categories are:

- Art
- Books
- Electronics
- Home & Garden
- Sporting Goods
- Toys
- Other

Likewise the list of **Conditions** is also fixed by the system. The possible values are:

- New
- Very Good
- Good
- Fair
- Poor

The list of conditions is guaranteed never to change. Also, conditions have a total ordering and can be compared with each other. Keeping this in mind, choose an appropriate data representation during the implementation phase that will allow for easy comparisons (e.g., “at least good”).

The auction rules are as follows:

- The **Starting Bid** is the minimum amount required for the first bid. No one can make a bid below this price.
- The **Minimum Sale Price** is the smallest amount that the user is willing to set for the item. This information is visible only to the user listing the item. Users bidding on the item will not know if the minimum sale price has been met until the auction is over. This gives the listing user peace of mind that an item will not sell too cheaply.
- The user also determines **Auction Length**. The options are 1, 3, 5, and 7 days. The **Auction End Time** is calculated when the user submits the item by adding the selected number of days to the current date and time. For example, if the user creates a new item on March 1, 2018 at 5:00PM and chooses an auction length of 1 day, the auction will end on March 2, 2018 at exactly 5:00PM.
- Optionally, the user can set a **Get It Now Price**. This is the price at which the user is willing to sell the item immediately. If another user chooses to Get It Now, then the auction is immediately over and that user is declared to be the winner.

When the user clicks *List My Item*, your application should verify that all required fields are filled in and that the data is valid. In particular, you should consider what constraints should hold among the **Starting Bid**, **Minimum Sale Price**, and **Get It Now Price**. If the user enters invalid data the item should not be listed. Instead, show an appropriate error message and return the user to the form in order to make corrections.

All items are assigned a unique numerical **Item ID** so that the item can be identified easily. The ID is automatically generated by the system and cannot be edited by the user. When the user submits the form, your application should choose a unique ID for the new item. The ID will be shown on the Search Results screen and the Item Description page.

The screenshot shows a web form titled "New Item for Auction". The form contains the following fields and callouts:

- Item Name:** A text input field containing "Garmin GPS 255W".
- Description:** A text area containing "You'll never be lost thanks to the Garmin 255W. It features a 4" screen, up to 8 hours battery life, and a suction cup to stick it to your windshield." A callout points to this field: "Choose from a fixed list of categories."
- Category:** A dropdown menu showing "Electronics". A callout points to this field: "Conditions include: New, Very Good, Good, Fair, and Poor".
- Condition:** A dropdown menu showing "Very Good".
- Start auction bidding at \$:** A text input field containing "35.00". A callout points to this field: "This is the starting bid price."
- Minimum sale price \$:** A text input field containing "60.00". A callout points to this field: "This is the minimum price you will accept for the item. It is not seen by bidders."
- Auction ends in:** A dropdown menu showing "3 days". A callout points to this field: "This indicates when the auction will end. Possible values are 1 day, 3 days, 5 days, and 7 days."
- Get It Now price \$:** A text input field containing "79.00", followed by the text "(optional)". A callout points to this field: "A user can immediately buy the item at this price."
- Returns Accepted?** A checkbox that is checked.
- Buttons:** "Cancel" and "List My Item" buttons at the bottom right.

Figure 3 - New Item for Auction

Searching for Items

Users can search for items for sale using a search interface like the one shown in Figure 4. The following search criteria are available:

- **Keyword:** An item is returned if the keyword appears anywhere in the **Item Name** or **Description**.
- **Category:** If a **Category** is selected, only items in that category will be returned. If no category is selected, then items in any category may be returned. The drop down should contain a blank row in case the user does not wish to search on category.
- **Minimum Price:** If the minimum price field is filled in, items returned from the search must have a current high **Bid** of at least this amount. If no one has bid on the item, then the **Starting Bid** price should be used instead.
- **Maximum Price:** If the maximum price field is filled in, items returned from the search must have a current high **Bid** of less than or equal to the maximum price. Obviously, if an item's **Starting Bid** is higher than the maximum price, then the item would not be returned in the search. If an item has no bids but the **Starting Bid** is lower than the maximum price, it should be included in the results.
- **Condition at least:** If this field is selected, then only those items with *the selected condition or better* are returned. For example, if the user chooses the condition “Good,” then items with conditions of “Good,” “Very Good,” or “New” should be returned. This drop down should have a blank row in case the user does not wish to search on **Condition**.

The screenshot shows a dialog box titled "Item Search" with a close button (X) in the top right corner. It contains five input fields and two buttons at the bottom. Callouts provide additional context for each field:

- Keyword:** A text box containing "GPS". A callout states: "Searches Item Name and Description fields".
- Category:** A dropdown menu showing "Electronics".
- Minimum Price \$:** A text box with a cursor. A callout states: "Current bid must be at least this price".
- Maximum Price \$:** A text box. A callout states: "Current bid cannot be more than this price".
- Condition at least:** A dropdown menu showing "Good". A callout states: "Minimum acceptable condition".
- Buttons:** "Cancel" and "Search". A callout pointing to both buttons states: "Search results must match ALL criteria (i.e., these fields are ANDed together)".

Figure 4 - Item Search

When the user clicks *Search*, only those items that meet ALL of the criteria will be returned. Not all criteria must be entered for a search. For example, if the user selects the **Category** “Arts” and **Condition** at least “Fair”, then only those items that are in the “Arts” category AND have a condition of at least “Fair” will be returned. If the user enters no criteria, then a listing of all items will be returned. The user will be redirected to the *Search Results* screen as depicted in Figure 5.

The screenshot shows a 'Search Results' page with a table of auction items. Callouts provide additional context: 'Current high bid or blank if no bids yet' points to the 'Current Bid' column; 'User who has the current high bid' points to the 'High Bidder' column; 'Results are sorted by time, with the item ending soonest on top' points to the 'Auction Ends' column; 'Clicking on item name takes the user to the item page where she can view details and place a bid' points to the 'Item Name' column; and 'Return to the search screen for another search' points to the 'Back to Search' button.

ID	Item Name	Current Bid	High Bidder	Get It Now Price	Auction Ends
1001	Garmin GPS 255W	\$44.00	lostnfound99	\$79.00	3/3/2018 4:30PM
1012	Garmin nuvi 200	\$66.00	sistaSuzy	\$99.00	3/3/2018 4:41PM
1054	TomTom XL 340S	-	-	\$109.00	3/4/2018 8:15AM
1067	Magellen RoadMate	\$28.00	thinkmcfly	\$55.00	3/6/2018 9:22PM

Back to Search

Figure 5 - Search Results

The *Search Results* screen lists all those items that match the user's criteria and are still for sale (i.e., the auction has not yet ended). Items from auctions that have already ended should NOT appear in the search results. The fields listed in the search results include the **Item ID**, **Item Name**, the current **Bid Amount** (if there is one), the **Username** of the user who has the high bid (if there is one), the **Get It Now Price** (if one is set), and the **Auction End** date and time. The results should be sorted in order of the auction end date with the auction ending the soonest listed first. The list of results can be displayed in its entirety, or paginated.

If the user clicks on the **Item Name**, then he or she is taken to the Item Description page as shown in Figure 6. From this screen the user can see information about the item for sale, including the **Item ID**, **Item Name**, **Description**, **Category**, **Condition**, if the item is **Returnable**, and the **Get It Now Price** if one has been set. Information about the latest four bids (at most) is also shown. The bidding information includes the **Bid Amount**, **Time of Bid**, and the **Username** of the person who made the bid. The bids are sorted by the **Time of Bid** with the latest one listed first. If no one has bid on the item, then the latest bids section will be empty.

GTBay Item for Sale

Item ID: 1001 [View Ratings](#) (Takes the user to the ratings screen for this item.)

Item Name: Garmin GPS 255W

Description: You'll never be lost thanks to the Garmin 255W. It features a 4" screen, up to 8 hours battery life, and a suction cup to stick it to your windshield.

[Edit Description](#) (This option only available if the current user is also the listing user)

Category: Electronics

Condition: Very Good

Returns Accepted? ☒

Get It Now price: \$79.00 (This row only appears if a "Get It Now" price has been set. Clicking the button purchases the item immediately.)

Auction Ends: 3/3/2018 4:30PM

Latest Bids

Bid Amount	Time of Bid	Username
\$44.00	3/3/2018 1:14PM	lostnfound99
\$40.00	3/3/2018 1:00PM	bigfella
\$39.00	3/3/2018 9:27AM	lostnfound99
\$35.00	3/2/2018 12:15PM	fliznip2010

(Displays the latest four bids with the highest (and most recent) bid on top.)

Your bid: \$ 50.00 (minimum bid \$45.00)

How much the user will bid for the item. It must be at least \$1 above the current bid and less than the Get It Now price (if one has been set).

Figure 6 - Item Description

From this screen, the user has several options:

- The user can click *cancel* to return to the *Search Results* screen.
- If the user viewing the item is also the user who listed the item, then the user is allowed to edit the item description. When the user clicks *Edit Description*, display a popup window asking for the new description. The existing description should then be replaced with the new one. Note that this is the only field that the user can update after the item has been created. If the user viewing the item is not the listing user, then the *Edit Description* option should not be visible.
- The user can click *View Ratings* to see ratings and comments about the item (see Figure 7).
- The user can purchase the item immediately if a **Get It Now Price** has been set. In this case a button should appear allowing the user to make the purchase. This immediately ends the auction (even if the time is not yet expired) and sets the current user as the **Winner** of the auction.
- The user can bid on the item. You are allowed to decide if a user can bid on their own item they have listed, but it's probably a good idea to prevent it. The user's bid must be at least one dollar higher than the current (highest) **Bid Amount**. The bid must also be less than the **Get It Now Price**. If the user attempts to bid higher than the **Get It Now Price**, an error message should be shown suggesting that the user click Get It Now instead of bidding.

Item Ratings

The screenshot shows a web interface titled "Item Ratings". At the top, it displays item information: Item ID 1001, Item Name Garmin GPS 255W, and Average Rating 3.5 stars. A callout points to the average rating, stating "Average of all ratings for this item." Another callout points to the list of ratings, stating "Ratings are listed with the most recent rating listed first." Below this, a list of ratings is shown. The first rating is by user "gojackets" on "3/2/2018 12:22PM" with a 3-star rating. A callout points to the "Delete My Rating" link, stating "This option only available if this is the current user's rating". The second rating is by user "thinkmclv" on "1/2/2018 10:57AM" with a 4-star rating. Below the ratings, there is a "My Rating" section with a 4-star rating and a "Comments" text area. A callout points to the star rating, stating "Users rate items from 0 to 5 stars." At the bottom, there are two buttons: "Cancel" and "Rate This Item". A callout points to the "Cancel" button, stating "Returns user to item description page without making a rating." Another callout points to the "Rate This Item" button, stating "Submits the user's rating and comment."

Figure 7 - Item Ratings

Users have the ability to rate items on a scale of zero to five stars. Users can also provide **Comments** on the item. When a new rating is submitted, the system will record the user who made the **Rating**, the **Number of Stars**, the **Comment**, and the **Date and Time** of the **Rating**.

The Item Ratings screen (see Figure 7) also shows existing ratings for all items with the same name as the listed item, sorted in the order that the ratings were entered with the most recent rating listed first. In addition to listing each rating separately, the average star rating is also calculated and listed at the top of the page. If the user merely wants to view existing ratings and not enter a new rating, he or she clicks *Cancel* to return to the Item Description page.

Your system should make sure that a user may only rate an item listing once. The user also has the ability to delete any of their own ratings. Any ratings that the user has submitted should have a *Delete My Rating* option present. If the user chooses this option, that rating should be deleted from the database and the page should be refreshed.

Auction Results

The Auction Results screen (see Figure 8) shows information about auctions that have ended. The fields shown are the **Item ID**, the **Item Name**, the final sale price, the **Username** of the **Winner**, and the **Date and Time** that the auction ended. This list should only include information about auctions that have already ended. The list should be sorted by the auction end date/time with the most recently ended auction listed first and can display all results at once, or if desired, paginated. The list is static; navigation to any other screens from rows in the list is not possible.

The screenshot shows a window titled "Auction Results" with a table of auction data. Three callout boxes provide additional context: one points to the "Sale Price" column stating it's either the final bid or the Get It Now price; another points to the "Auction Ended" column stating auctions are sorted by end date/time; a third points to the "Sale Price" and "Winner" columns for the "Danish Art Book" row, stating they are blank because the item did not sell. A "Done" button is located at the bottom right of the window.

<u>ID</u>	<u>Item Name</u>	<u>Sale Price</u>	<u>Winner</u>	<u>Auction Ended</u>
988	Canon Powershot	\$170.00	thinkmcfly	3/2/2018 3:32PM
983	Nikon D3	\$1950.00	sistaSuzy	3/2/2018 2:41PM
990	Danish Art Book	-	-	3/2/2018 12:15PM

Figure 8 - Auction Results

The sale price and winner fields should be calculated in the following way:

- If the item was purchased using the Get It Now option, then the sale price is the **Get It Now Price** and the winner is the **Username** of the person who purchased the item.
- If, at the time the auction is over, the high **Bid** is greater than or equal to the **Minimum Sale Price**, then the sale price is the value of the high **Bid** and the winner is the **Username** of the person with the high **Bid**.
- If no one purchased the item using the Get It Now option, and if the highest **Bid** is still lower than the **Minimum Sale Price** (or there are no **Bids** at all), then both the sale price field and the winner field should be left blank.

An implementation decision you will have to make is when to actually calculate auction winners. A recommended strategy is the following: Whenever a user chooses to view the Auction Results screen, before showing any results, first search the list of items for any auctions that do not yet have a declared winner but have already ended (based on the current time). Then, update those auctions with the winner (if there is a winner) before showing the list of results.

An alternative but more difficult implementation is to create a background process that will periodically check for auctions that have ended and update the database with the auction winners. In a "real" system, this option would be the best implementation. However, because of the added complexity of creating a separate process, you are not required to use this implementation option.

GTBay Administrative Reports

Administrative users have the ability to view two reports: the Category Report and the User Report.

When writing SQL statements for these reports, use as few SQL statements as possible. In some cases, you will need to use a separate SQL statement to calculate totals. The important thing to remember is to let the database do the work. Don't simply pull in all the needed information and do the grouping/aggregation using programming language constructs. The query processor can do it much more efficiently than you can by hand. In other words, you want the output returned from your SQL query to be as close as possible to the information in the final report.

The Category Report (see Figure 9) aggregates information about items by **Category**. The fields in the report are the **Category** name, the total number of items in the category, the minimum **Get It Now Price** across all items in the category, the maximum **Get It Now Price** across all items in the category, and the average **Get It Now Price** across all items in the category. The report should be sorted alphabetically by category name. Keep in mind that some items may not have a **Get It Now Price**. They should not be included in the min, max, or average prices. (In other words, if an item does not have a **Get It Now Price**, do NOT assume it has a **Get It Now Price** of \$0). This report should include both items that have already been sold (the auction has ended) or that are still for sale.

The screenshot shows a window titled "Category Report" with a table of data. Callouts point to specific values: "Get It Now Price of the most expensive item in the category" points to the Max Price for Art (\$2500.00); "Average Get It Now Price of items in the category" points to the Average Price for Art (\$108.24); "Get It Now Price of the cheapest item in the category" points to the Min Price for Art (\$3.00); and "Categories listed in alphabetical order" points to the list of categories.

<u>Category</u>	<u>Total Items</u>	<u>Min Price</u>	<u>Max Price</u>	<u>Average Price</u>
Art	35	\$3.00	\$2500.00	\$108.24
Books	22	\$5.50	\$150.00	\$34.02
Electronics	76	\$3.50	\$2200.00	\$59.09
Home & Garden	11	\$6.00	\$120.00	\$13.22
Other	39	\$1.00	\$550.00	\$14.77
Sporting Goods	23	\$8.75	\$350.00	\$76.93
Toys	14	\$10.00	\$95.00	\$33.03

Figure 9 - Category Report

The second administrative report shows some aggregate information about users. The fields in the report are **Username**, the total number of items listed by the user (whether sold or not), the total number of items sold by the user (either via Get It Now or auction), the total number of items purchased by the user

(either via Get It Now or auction), and the total number of items that the user rated. Figure 10 shows an example User Report.

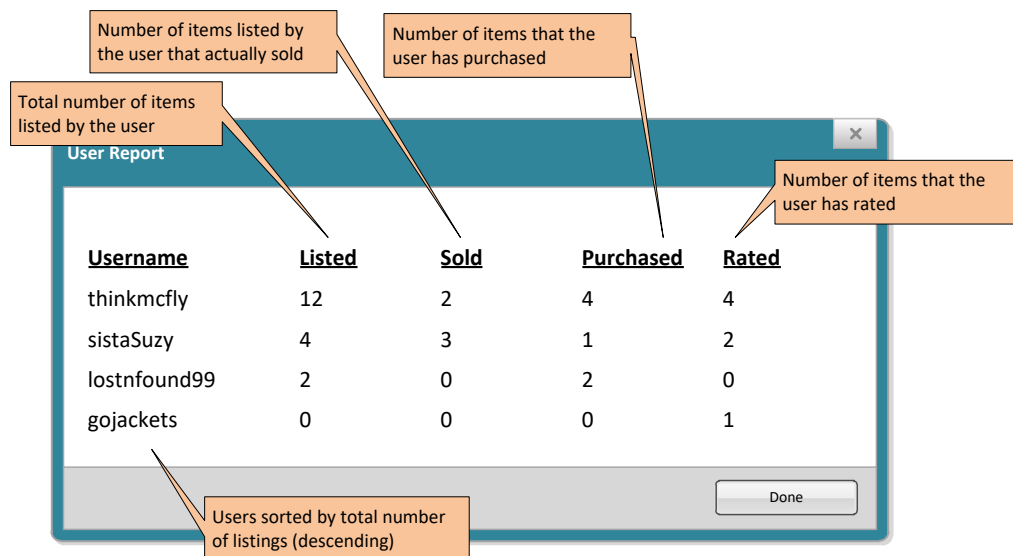


Figure 10 - User Report

Document Version Info

<u>Version</u>	<u>Notes</u>	<u>Date</u>
1.0	Original version	1/15/2018
1.3	Minor updates to a few graphics	1/15/2018
1.4	More minor graphic updates	1/15/2018
1.5	Added abstract code to list of deliverables	1/21/2018
1.6	Clarification on condition, searching, search results, bidding, ratings, and auction results.	3/26/2018