Phase 1 Report | CS 6400 - Fall 2017 | Team 091

Table of Contents

Tools-4-Rent! Data Types

User

Attribute	**Data type**	**Allow Null**
first-name	String	Not Null
middle-name	String	Not Null
last-name	String	Not Null
username	String	Not Null
password	String	Not Null
e-mail	String	Not Null

Customer

Attribute	**Data type**	**Allow Null**
street	String	Not Null
city	String	Not Null
state	String	Not Null
zip-code	String	Not Null
number	Int	Not Null
name-on-card	String	Not Null
expiration-month	Int	Not Null
CVC	Int	Not Null
expiration-year	Int	Not Null
area-code	Int	Null
phone-number	Int	Null
extension	Int	Null
primary	boolean	Not Null

Clerk

Attribute	**Data type**	**Allow Null**
date-of-hire	date	Not Null
employee-number	int	Not Null
temp-password	String	Not Null

Reservation

Attribute	Data type	Allow Null
start-date	date	Not Null
end-date	date	Not Null
pick-up-date	date	Null
drop-off-date	date	Null
reservation-id	Int	Not Null
total-deposit	float	Not Null
total-rental	float	Not Null
num-days	Int	Not Null

Rental

Attribute	Data type	Allow Null
tool-deposit-price	float	Not Null
tool-rental-price	float	Not Null

Sale order

Attribute	Data type	Allow Null
for-sale-date	date	Not Null
sale-date	date	Null
sale-price	float	Not Null

Service order

Attribute	**Data type**	**Allow Null**
service-order-id	Int	Not Null
service-cost	float	Not Null
service-start-date	date	Not Null
service-end-date	date	Not Null

Tool

Attribute	**Data type**	**Allow Null**
length	float	Null
width-diameter	float	Null
material	String	Null
tool -number	Int	Not Null
sub-option	String	Not Null
sub-type	String	Not Null
power-source	String	Not Null
original-price	float	Not Null
for-sale	boolean	Null

Power tool

Attribute	Data type	Allow Null
volt-rating	float	Not Null
amp-rating	float	Not Null
min-rpm-rating	int	Not Null
max-rpm-rating	int	Not Null
max-torque-rating	int	Null
min-torque-rating	int	Null
adjustable-clutch	boolean	Null
blade-size	float	Null
dust-bag	boolean	Null
power-rating	float	Null
tank-size	float	Null
pressure-rating	float	Null
motor-rating	float	Null
drum-size	float	Null

Tool accessory

Attribute	Data type	Allow Null
accessory-name	string	Null
accessory-quantity	int	Null
accessory-description	string	Null

Battery

Attribute	Data type	Allow Null
voltage	int	Null
amperage	float	Null

Hand tool

Attribute	Data type	Allow Null
sae-size	float	Null
drive-size	float	Null
deep-socket	boolean	Null
screw-size	int	Null
drive-size	float	Null
adjustable	boolean	Null
anti-vibration	boolean	Null
gauge-rating	int	Null
capacity	int	Null

Tools-4-Rent! Business Logic Constraints

General

- Columns in table should be sortable by clicking on header
- Several tasks require searches based on either combination of attributes or keyword. For attribute combinations, pulldown menus should be populated as selections are made
- The existence of sales orders and service orders should determine the status of a tool for a given date--no status field for each tool

Clerk

- Clerk username, e-mail, temporary password, employee number and date of hire entered by system administrator
- All clerk e-mails contain the domain name: '@tools4rent.com'

Tool

- Purchase price is entered with each tool; rental price is 15% of purchase price; deposit is 40% of purchase price; sale price is 50% of purchase price
- All tools are available for daily rental; one day is 24 hours
- After 50 rents, a tool is automatically for sale
- All dimensions are stored as inches; all ratings stored as standard units (no "milli", "kilo", etc)
- Short tool description and full tool description are a concatenation of attributes into a string

Login

- If a customer logs on with a username that does not exist, they will be prompted to register
- If customer logs in as clerk, display error
- If it is the first time a clerk logs in with a temporary password, the clerk is prompted to change the password by entering it twice

Registration

- If a customer exists with an existing username, the user should not be allowed to register as a new customer
- User is required to select one phone as primary before exiting registration
- Need to account for hyphens and extensions when entering phone numbers
- Entering credit card information is required during registration

View Profile

- List all known information about user

- Rental history listed from most recent to oldest and includes the name of clerk who handled the reservation

Check tool availability

- Form has search feature
- If more than ten tools returned from a search--prompt for more restrictive search

Reservation

- Reservation form has search feature
- If identical tool to be returned within 24 hours, notify customer
- Fewer than ten tools per reservation
- Total price is rental price per tool times number of days rented; Deposit price is sum of deposit per tool; total price is rental price minus deposit price
- Unique reservation number assigned when reservation is completed
- Concurrency: the same tool can appear on multiple pending reservations, but tool is assigned to whoever completes registration first. Other reservations should show error if tool is not available

Reservation pick-up

- Customer has option of updating credit card information at time of pick-up

Add Tool

- Deposit and rental prices automatically determined based on the purchase price
- Accessories need to be separately listed for power tools
- Clerk should not be allowed to choose subtype without first selecting tool category and power source; inapplicable data fields should be hidden/disabled to prevent data corruption
- Selections in each drop down to be dynamically determined
- Clerk should not be allowed to mismatch voltage requirements/battery type for cordless tools--they must be guided to select cordless first
- Cordless battery dropdowns contain ‘Li-ion’, ‘NiCd’ or ‘NiMH’ only
- Chuck/drive sizes in drop down are ¼”, ½”, ⅜”,¾” and 18,20,22,24G

Repair tool/service status

- Repair and service status forms have search feature
- Tool is not available for rent during the dates the tool is in repair
- Repair override: A clerk can “fix now” to update the end repair and return tool to “available”; Clerk on record overwritten to be clerk performing the override
- Clerk determines repair cost when creating service request
- Clerk cannot duplicate service request on the same tool

Sell tool/sale status

- Sell tool and view sale status forms have search feature
- Tools are sold after rented 50 times

- Tools may be reserved but not picked up; in this case, a rental should not be counted
- Tools are automatically put up for sale after the 50th rental with the clerk listed as “Jill Watson” (system’s automated virtual clerk)
- Once a tool is sold a unique transaction number is created which tracks the tool number, customer who bought the tool, clerk who marked the tool for sale, sale price, for sale date and sale date
- If a tool has not been marked for sale, a customer cannot purchase it
- A tool being changed to “for sale” cannot be in service or have an open reservation

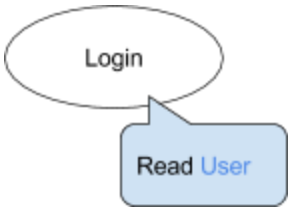
Reports

- Generate three reports: clerk report, customer report and tool report
- Clerk report lists clerk’s employee number, full name, email, hire date, number of pick-ups and drop offs handled this month and sum of two; ordered by number of pick-ups and drop offs
- Customer report lists all customers who rented a tool over the last month. Shows username, link to full profile, full names, e-mail address, primary phone number, and number of rentals. List is sorted by number of rentals
- Tool report lists everything in inventory with tool ID, current status of tool, relevant date, description, rental profit, cost of toll and total profit (rental profit: # of days rented times cost/day; total profit: rental profit minus tool cost). List is ordered by total profit

Task Decomposition/Abstract Code

Login

Task Decomposition



Lock Types: Read-only on User
Number of Locks: Single
Enabling Conditions: None
Frequency: ~200 log-ins per day
Consistency (ACID): not critical, order is not critical
Subtasks: Mother Task is not needed. No decomposition needed.

Abstract Code

- User enters *username*, *password* and selects "**Customer/Clerk**" radio button.
- When ***Sign in*** button is clicked:
    - If ***"Customer"*** radio button was clicked:
        - If *username* and *password* verified, then **Customer Main** form
        - If *username* exist, but *password* not verified then "Error: incorrect password"
        - If username does not exist, then **Registration** form
        - If existed "Clerk" user login as "Customer", then error message would be displayed.
    - If ***"Clerk"*** radio button was clicked:
        - If *username* and *password* verified, then **Clerk Main** form
        - If *username* exist, but *password* not verified then "Error: incorrect password"
        - If *username* exists and *password* is temporary password (first login), then prompt to reset password by entering new password twice
        - If *username* is an existing customer and *password* is verified, then display "Error: Login as customer"

Main Menu / Navigation Bar

Abstract Code

If user login as "**Customer**":

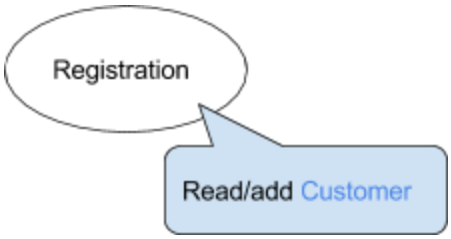
- Show ***"View Profile", "Check Tool Availability", "Make Reservation", "Purchase Tool"*** and ***"Exit"*** tabs.
- Upon:
    - Click ***View Profile*** button- Jump to the **View Profile** task.
    - Click ***Check Tool Availability*** button- Jump to the **Check Tool Availability** task.
    - Click ***Make Reservation*** button- Jump to the **Make Reservation** task.
    - Click ***Purchase Tool*** button- Jump to the **Purchase Tool** task.
    - Click ***Exit*** button- exit main menu and go back to the **Login** form.

If user login as "**Clerk**":

- Show ***"Pick-Up Reservation", "Drop-Off Reservation", "Add New Tool", "Service Order", "Service Status", "Sell Tool", "Sale Status", "Generate Reports"*** and ***"Logout"*** tabs.
- Upon:
    - Click ***Pick-Up Reservation*** button- Jump to the **Pick-Up Reservation** task.
    - Click ***Drop-Off Reservation*** button- Jump to the **Drop-Off Reservation** task.
    - Click ***Add New Tool*** button- Jump to the **Add New Tool** task.
    - Click ***Service Order*** button- Jump to the **Service Order** task.
    - Click ***Service Status*** button- Jump to the **Service Status** task.
    - Click ***Sell Tool*** button- Jump to the **Sell Tool** task.
    - Click ***Sale Status*** button- Jump to the **Sale Status** task.
    - Click ***Generate Reports*** button- Jump to the **Generate Reports** task.
    - Click ***Log Out*** button- Invalidate login session and go back to the **Login** form.

Registration

Task Decomposition



Lock Types: Read / add **Customer**
Number of Locks: Single schema construct
Enabling Conditions: Both two are enabled when **Customer Registration Form** was opened.
Frequency: medium, projected users up to 100,000 registered Customers per store, so possible several hundred new registrations per day
Consistency (ACID): is critical, if a username was used by a existed valid "**Customer**", such user was not able to register as a new customer.

Subtasks: All tasks must be done and "Customer" personal information input should be done first. Mother task is required to coordinate subtasks.

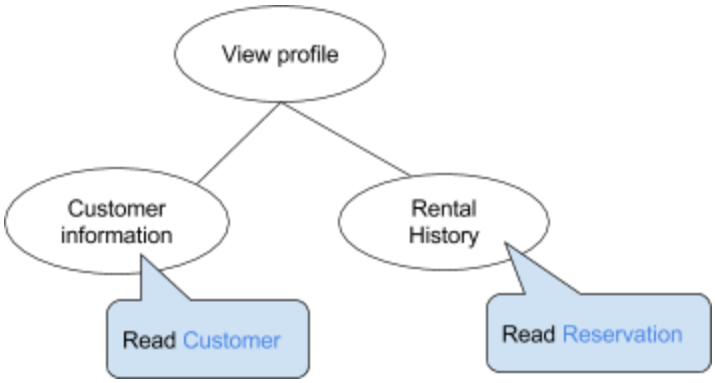
Abstract Code

"Customer" opened **Customer Registration Form**:

- GET username, e-mail address, full name (first, middle, last), home phone, work phone and cell phone information.
- PARSE phone numbers to area code, number and extension.
- CHECK if primary phone number was declared by **"Customer"**.
- GET Customer's street address, city state, 9-digit zip code with hyphen.
- GET **Customer's** credit card number and name on card.
- GET credit card expiration month and year from dropdown menu.
- GET credit card CVC 3-digit number.
- IF username was used by valid existing **"Customer"**, THEN display "Error: Customer already exists".
- CLICK ***Register*** button:
    - IF all required fields were filled THEN return to **Login** form.
    - ELSE display "Error: missing required value(s)".

View Profile

Task Decomposition



Lock Types: Two read-only lookups of personal information in Customer and rental history in Reservation
Number of Locks: Several different schema constructs are needed**.**
Enabling Conditions: Both two are enabled by a user's login.
Frequency: Low- Both two have the same frequency.
Consistency (ACID): is not critical and there is no option to edit a customer's profile.
Subtasks: All tasks must be done, but can be done in parallel. Mother task is required to coordinate subtasks. Order is not necessary.

Abstract Code

"Customer" clicked on ***View Profile*** button from **Main Menu**:

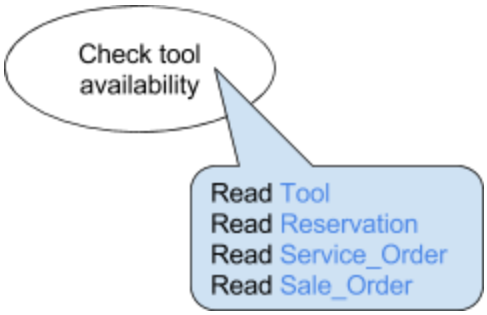
- Run the **View Profile** task: extracting information about the rental customer and their profile where *username* is the identifier of the current **"Customer"** using the system from the HTTP Session/Cookie.
- Find the current **"Customer"** using the *username*; Display user's first and last name; Display user's E-mail; Display user's all phone numbers; Display user's Address.
- Find the rental history of the current **"Customer"**, listing all reservations made by the user, ordered from most recent to oldest.
- For each **Reservation ID:**
- When ready, user selects next action from choices in **Main Menu**
    - Display all tools reserved in this single reservation.
    - Display reservation start date and end date.
    - Display the names of **"Clerk"** who handled the tools pick-up / drop-off sessions.
    - Display the number of reservation days.

- Display the total deposit price and total rental price.

When ready, user selects next action from choices in **Main Menu.**

Check Tool Availability

Task Decomposition



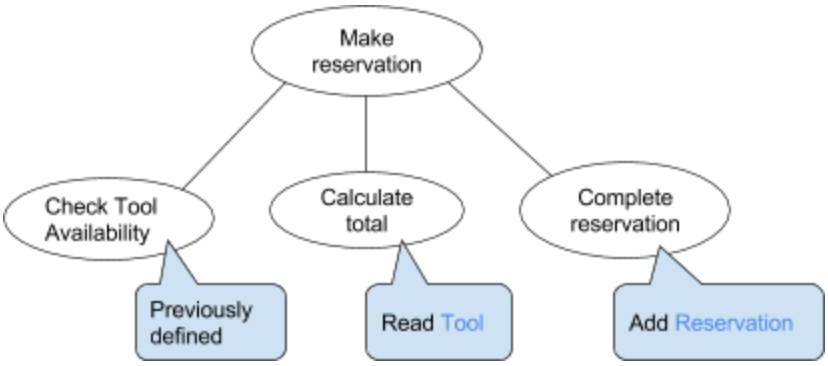
Lock Types: read on Tool, Reservation, Service_Order, Sale_Order
Number of Locks: One on each schema needed
Enabling Conditions: “Check Tool Availability” option is selected from the main menu
Frequency: ~50/day
Consistency (ACID): Tasks do not need to be completed in order
Subtasks: No mother tasks needed

Abstract Code

GET SearchValues (StartDate, EndDate, Keyword, Type, PowerSource, SubType)
List=Query of SearchValues with ToolAvailable==TRUE
IF len(List)>10 THEN “Specify more unique search”
ELSE:
Description=FORMAT(Dimensions+PowerSource+SubOption+SubType+Other+Manufacturer)
DepositPricePerReservation=0.4*PurchasePrice
RentalPricePerDay=0.15*PurchasePrice
DISPLAY ToolID, Description, DepositPricePerReservation, RentalPricePerDay
WHILE no buttons pressed, do nothing
IF more details linked pressed, display full description
IF Column header pressed, sort column

Make Reservation

Task Decomposition



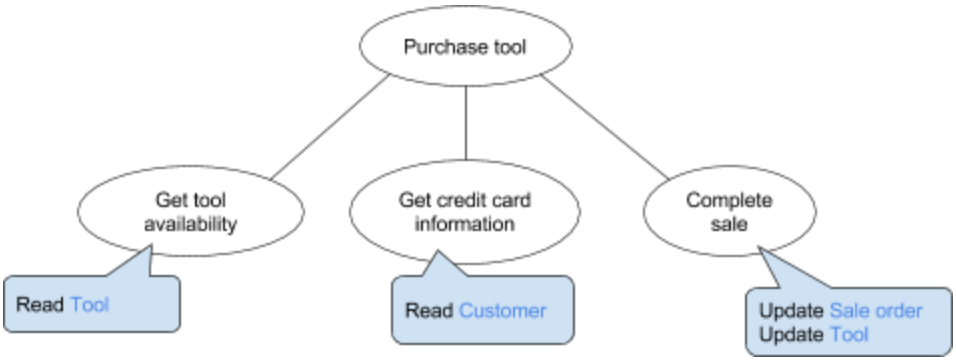
Lock Types: read/update on Customer, read/update on Tool, read/update on ReservationList, read on SaleOrder (PurchasePrice),
Number of Locks: Several schemas needed.
Enabling Conditions: Reservation Button is selected from the Main Menu
Frequency: ~50/day
Consistency (ACID): Tasks do not need to be completed in order.
Subtasks: No mother task needed.

Abstract Code

WHILE no buttons pressed, do nothing
IF Search pressed:
GET SearchValues (StartDate, EndDate, ToolCatagory, PowerSource/SubTypes and/or Keyword)
AvailableList=Query of SearchValues with ToolAvailable==TRUE OR ToolAvailableWithin24Hours==TRUE
Description=FORMAT(Dimensions+PowerSource+SubOption+SubType+Other+Manufacturer)
DepositPricePerReservation=0.4*PurchasePrice
RentalPricePerDay=0.15*PurchasePrice
DISPLAY ToolID, Description, DepositPricePerReservation, RentalPricePerDay
IF Add pressed:
IF len(Reservation)==10 THEN “Error: Max number of tools exceeded”
ELSE Add Tool to Reservation
IF Remove pressed:
Remove tool from Reservation
If Calculate Total pressed
CALCULATE Deposit Total and Rental Total
DISPLAY Reservation Summary Form

Purchase Tool

Task Decomposition



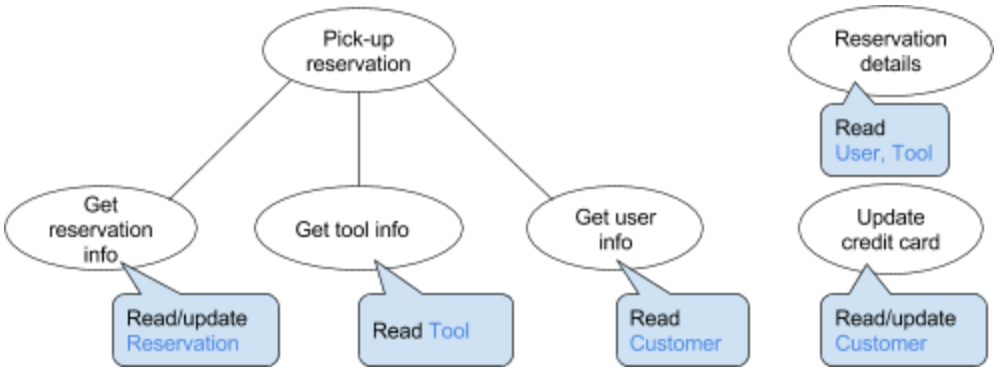
Lock Types: read/update on Tool, read on Customer, update on Sale Order,
Number of Locks: several schema constructs involved
Enabling Conditions: “Purchase Tool” button is selected from main menu
Frequency: ~30/day
Consistency (ACID): Critical/order is critical
Subtasks: Mother Task is needed.

Abstract Code

WHILE no buttons pressed, do nothing
IF Search pressed:
GET SearchValues (ToolCategory, PowerSource, SubType, Keyword)
List=Query of SearchValues with ToolForSale==TRUE
SalePrice=0.5*PurchasePrice
DISPLAY ToolID, Status, Description, SalePrice
IF PurchaseTool pressed:
IF CreditCardOnFile==FALSE, DISPLAY Credit Card info form
ELSE:
SAVE purchase
DISPLAY Purchase Confirmation form

Pick-up Reservation

Task Decomposition



Lock Types: Read/update on Reservation; read on Tool; read on Customer (no credit card update), read/update on Customer (with credit card update); read on User
Number of Locks: Several schema constructs needed
Enabling Conditions: Enabled by selection from Clerk main menu
Frequency: ~200/day
Consistency (ACID): not critical, order is not critical
Subtasks: No mother task is need--task order driven by user.

Abstract Code

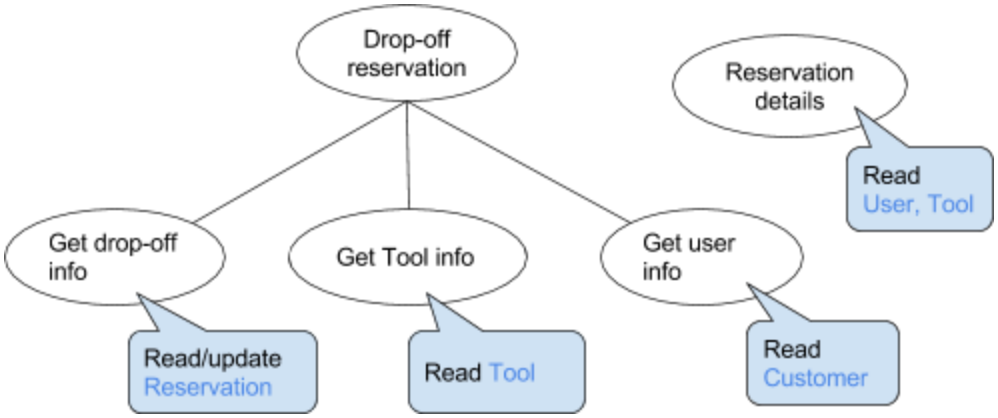
- Search Reservation for all reservations waiting to be picked up
- Find start-date and end-date in Reservation for each reservation to be picked up
- Find username and customer-id in Customer for each reservation to be picked up
- Display reservation-id, customer, customer-id, start-date and end-date for each reservation to be picked up

    - Upon click reservation-id
        - Find full-name in User
        - For each tool on reservation, find original-price in Tool
        - Calculate total deposit and rental price for each tool
        - Calculate total deposit price and total rental price for reservation-id
        - Display pop-out detail with reservation-id, full-name-total deposit and total-rental price
    - Upon entering reservation id and click ***PickUp***:
        - Display reservation summary
            - If credit card info in summary NULL and Click ***Confirm PickUp***
                - Find credit-card in Customer

        - Charge credit card with credit-card information
        - Update pick-up-date in Reservation
        - Display Rental Contract
            - If click ***Print Contract***--print contract
- If credit card info in summary Not Null and click Confirm PickUp
    - Update credit-card in Customer with information from form
    - Charge credit card with credit-card information
    - Update pick-up date in Reservation
    - Display Rental Contract
        - If click ***Print Contract***--print contract

Drop-off Reservation

Task Decomposition



Lock Types: Read/insert on Reservation; read/update on Tool; read on Customer; Read on User
Number of Locks: Several schema constructs are required
Enabling Conditions: Enabled by selection from Clerk Main Menu
Frequency: ~200/day
Consistency (ACID): not critical, order is not critical
Subtasks: Task order and completion driven by user

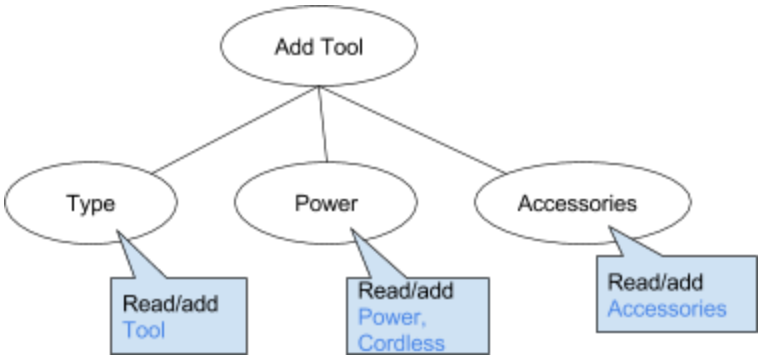
Abstract Code

- Search Reservation for all reservations that are ready for return (have been picked up but not returned)
- Find start-date and end-date in Reservation for each reservation to be returned
- Find username and customer-id in Customer for each reservation to be returned
- Display reservation-id, customer, customer-id, start-date and end-date for each reservation to be returned

- Upon click reservation-id
  - Find full-name in User
  - For each tool on reservation, find original-price in Tool
  - Calculate total deposit and rental price for each tool
  - Calculate total deposit price and total rental price for reservation-id
  - Display pop-out detail with reservation-id, full-name-total deposit and total-rental price
- Upon entering reservation id and click ***DropOff***:
  - Find full-name in User
  - For each tool on reservation, find original-price in Tool
  - Calculate total deposit and rental price for each tool
  - Calculate total deposit price, total rental price and total due for reservation-id
  - Display Drop off Reservation with reservation-id, full-name-total deposit, total-rental price, total due
    - If click ***Drop Off***
      - Update drop-off-date in Reservation
      - Display Final Receipt
        - If click ***Print Receipt***-print contract

Add Tool

Task Decomposition



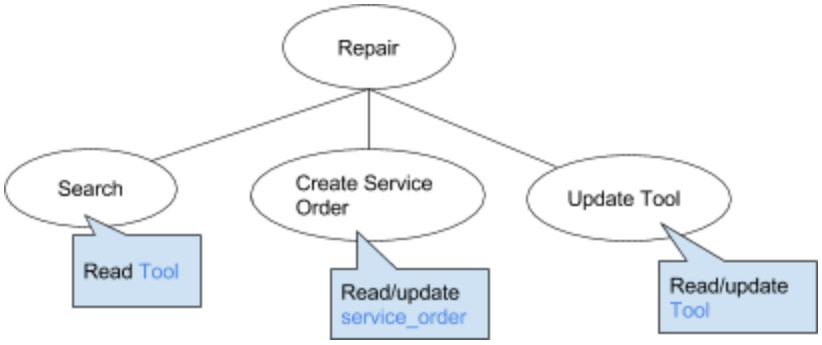
Lock Types: Read/insert on Tool, Power, Cordless, Accessory, Battery
Number of Locks: Several schema constructs are needed
Enabling Conditions: Enabled by selection from Clerk Main menu
Frequency: Low (~10/week)
Consistency (ACID): Not critical
Subtasks: Mother task is needed. Order is not necessary.

Abstract Code

- On click of ***Type [radio button]*** get options for Sub-Type field from Tool
    - If Type is Power Tool, display Power Tool suboption fields
        - Dynamically determine options from Power
        - Dynamically determine accessory options from Accessory
    - If Type is Cordless, display cordless suboptions
        - Dynamically determine options from Cordless
- Upon selection of Sub-Type, get options for Sub-Option field from Tool
- Upon click of Confirm
    - If width or length is feet, convert to inches
    - If amp, volt or power is “milli” or “kilo”, convert to decimal
    - Add tool to Tool

Repair Tool

Task Decomposition



Lock Types: Read/write on Tool, Service_Order
Number of Locks: Two
Enabling Conditions: Selected from Clerk main menu
Frequency: Low (~10/week)
Consistency (ACID): not critical, order is not critical
Subtasks: Tasks are done in series driven by user. No mother task is required.

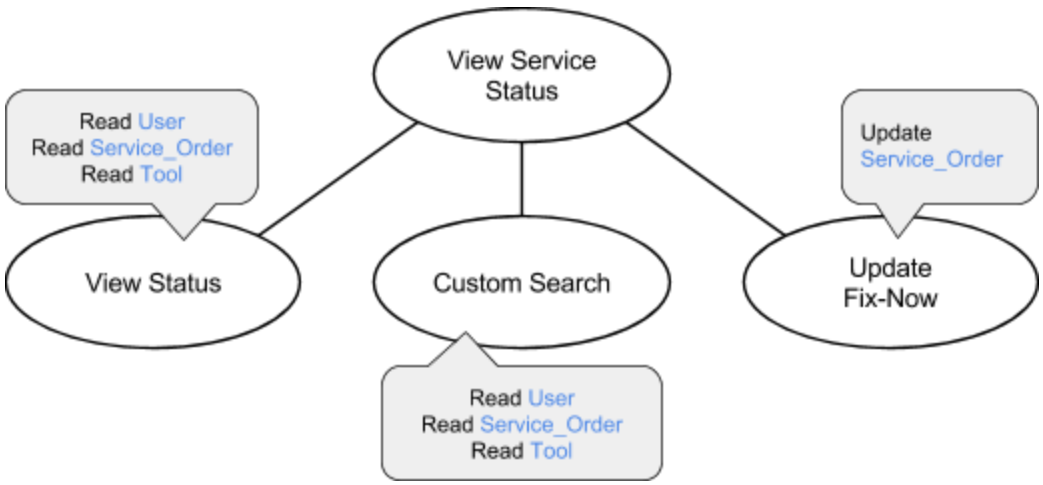
Abstract Code

- Upon click ***Search***
    - Get tool-id, short-desc (aggregate), original-price, from Tool that matches keyword in
    - Calculate Rental Price and Deposit Price
    - Display list
- Upon click ***Type [radio button], Power Source, Sub-Type***

    - Get tool-id, short-desc, original-price from Tool that matches search criteria
    - Calculate Rental Price and Deposit Price
    - Display list
- Upon click ***Service Tool***
    - Load associated tool-id into “Tool ID” field
- Upon click ***Confirm***
    - If “Tool ID” field is Null, error “No tool ID”
    - If “Enter Service Cost” is Null, error “Enter repair cost”
    - Verify “Start Date” and “End Date” fields valid
    - Create Service-order
        - Increment last service-id from Service_Order to create new id
        - Add new record in Service_Order

View Service Status

Task Decomposition



Lock Types: Read/Update Service_Order, Read-Only User, Read-Only Tool
Number of Locks: 3
Enabling Conditions: Clerk activates form from menu. Active service order exists. Type and Custom Search have matching values in service order records.
Frequency: Low. Repairs are non-deterministic and occur as needed.
Consistency (ACID): Not Critical
Subtasks:

- Display all tools currently being serviced
- Click “fix now” to update the end repair date to “now” and return tool to “available”

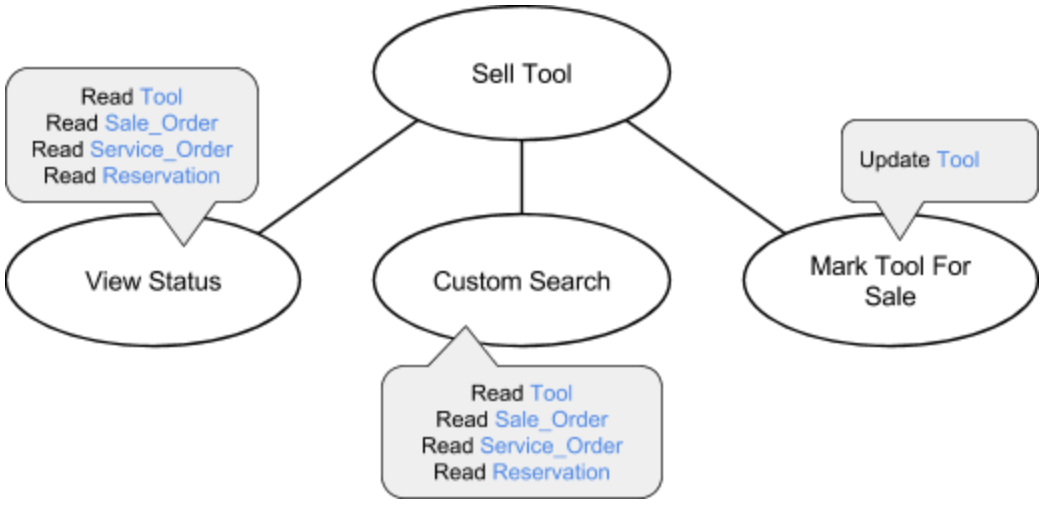
- Clerk on record overwritten to be clerk performing the override

Abstract Code

- Search Service_Order for active records and return service-id, service-cost, service-start-date, service-end-date, service-tool-id and service-clerk-id. Format dates as datetime and repair-cost as $X.XX.
- Using sale-tool-id from Service_Order, read Tool for sub-type and short-desc (aggregate), filtering results that match the selected sub-type in ***Type [radio button]***.
- Using service-clerk-id from Service_Order, read username from User for the matching user record.
- If the ***Custom Search [input field]*** is not empty, search short-desc (aggregate), service-start-date, service-end-date, service-cost and username for matching values.
- If a ***Column Header [label]*** has been selected, order the results by the corresponding field for the selected column header, either ascending or descending
- Upon click ***Type [radio button]***
    - Search Tool for matching tool-type values
    - Read tool-id, short-desc (aggregate) from Tool
    - Search Service_Order for service-tool-id equal to tool-id
    - Read service-id, service-start-date, service-end-date, service-cost, service-clerk-id from Service_Order
    - Format service-start-date and service-end-date as datetime and service-cost as $X.XX.
    - Search User for user-id equal to service-clerk-id
    - Read username from User
- Upon click ***Search***
    - Remove non-alphanumeric characters from search string
    - Read service-id, service-tool-id, service-start-date, service-end-date, service-cost, service-clerk-id from Service_Order
    - Search Tool for tool-id equal to service-tool-id
    - Read tool-id, short-desc (aggregate) from Tool
    - Search User for user-id equal to service-clerk-id
    - Perform case insensitive search on tool-id, short-desc (aggregate), service-start-date, service-end-date, service-cost, tool-type and username for search string
- Upon click ***Fix-Now***
    - Update Service_Order service-end-date to ‘now()’ using the record service-id

Sell Tool

Task Decomposition



Lock Types: Read-Only Tools, Sale_Order, Service_Order, Reservation, Update Sale_Order
Number of Locks: 5
Enabling Conditions: Clerk activates form from menu. Tools are available in inventory to rent (not sold, reserved or in service). Type, Custom Search, Power Source and Sub-Type have matching values in the available Tool records.
Frequency: Low ~10x / week
Consistency (ACID): Update of Tool for-sale must occur before View Status or Custom Search
Subtasks:

- Tools rented 50 times automatically marked “for sale” by system and updated to show the sale clerk as the virtual assistant, “Jill Watson”
- Tools can manually be marked for sale
- Allow for tool search

Abstract Code

- Search Tool for all records not currently “for-sale”
- Read tool-id, short-desc (aggregate), original-price from Tool
- Calculate rental-price to 15% of original-price and deposit-price to 40% of original-price
- Search Sale_Order for records with sale-tool-id equal to tool-id. Limit records from Tool to those without a sale-date.
- Search Service_Order for records with service-tool-id equal to tool-id and service-start-date less than today and service-end-date in the future. Limit records from Tool having no active record in Service_Order.
- Search Reservation for records with renting-tool-id equal to tool-id, start-date less than ‘now()’, end-date greater than ‘now()’ or pick-up-date less than ‘now()’ and drop-off-date

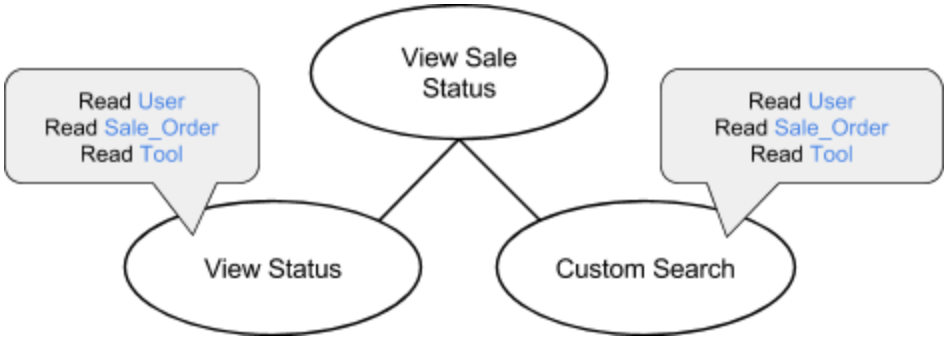
greater than ‘now()’ or without a value. Limit records from Tool having no active record in Reservation.

- Format rental-price and deposit-price as $X.XX
- Populate ***Power Source [dropdown]*** reading distinct power-source values from Tool
- Populate ***Sub-Type [dropdown]*** reading distinct sub-type values from Tool
- Upon click ***Type [radio button]***
    - Search Tool for matching tool-type values
    - Read tool-id, short-desc (aggregate), original-price from Tool
    - Calculate rental-price to 15% of original-price and deposit-price to 40% of original-price
    - Search Sale_Order for records with sale-tool-id equal to tool-id. Limit records from Tool to those without a sale-date.
    - Search Service_Order for records with service-tool-id equal to tool-id and service-start-date less than today and service-end-date in the future. Limit records from Tool that do not have an active record in Service_Order.
    - Search Reservation for records with renting-tool-id equal to tool-id, start-date less than ‘now()’, end-date greater than ‘now()’ or pick-up-date less than ‘now()’ and drop-off-date greater than ‘now()’ or without a value. Limit records from Tool having no active record in Reservation.
    - Format rental-price and deposit-price as $X.XX
- Upon click ***Search***
    - Remove non-alphanumeric characters from search string
    - Read tool-id, short-desc (aggregate), original-price, power-source, sub-type, tool-type from Tool
    - Calculate rental-price to 15% of original-price and deposit-price to 40% of original-price
    - Search Sale_Order for records with sale-tool-id equal to tool-id. Limit records from Tool to those without a sale-date.
    - Search Service_Order for records with service-tool-id equal to tool-id and service-start-date less than today and service-end-date in the future. Limit records from Tool that do not have an active record in Service_Order.
    - Search Reservation for records with renting-tool-id equal to tool-id, start-date less than ‘now()’, end-date greater than ‘now()’ or pick-up-date less than ‘now()’ and drop-off-date greater than ‘now()’ or without a value. Limit records from Tool having no active record in Reservation.
    - Perform case insensitive search on tool-id, short-desc (aggregate), rental-price, deposit-price, tool-type, sub-type and power-source for search string
    - Format rental-price and deposit-price as $X.XX
- Upon click ***Power Source [dropdown]***
    - Search Tool for matching power-source values
    - Read tool-id, short-desc (aggregate), original-price from Tool
    - Calculate rental-price to 15% of original-price and deposit-price to 40% of original-price

    - Search Sale_Order for records with sale-tool-id equal to tool-id. Limit records from Tool to those without a sale-date.
    - Search Service_Order for records with service-tool-id equal to tool-id and service-start-date less than today and service-end-date in the future. Limit records from Tool that do not have an active record in Service_Order.
    - Search Reservation for records with renting-tool-id equal to tool-id, start-date less than ‘now()’, end-date greater than ‘now()’ or pick-up-date less than ‘now()’ and drop-off-date greater than ‘now()’ or without a value. Limit records from Tool having no active record in Reservation.
    - Format rental-price and deposit-price as $X.XX
- Upon click ***Sub-Type [dropdown]***
    - Search Tool for matching sub-type values
    - Read tool-id, short-desc (aggregate), original-price from Tool
    - Calculate rental-price to 15% of original-price and deposit-price to 40% of original-price
    - Search Sale_Order for records with sale-tool-id equal to tool-id. Limit records from Tool to those without a sale-date.
    - Search Service_Order for records with service-tool-id equal to tool-id and service-start-date less than today and service-end-date in the future. Limit records from Tool that do not have an active record in Service_Order.
    - Format rental-price and deposit-price as $X.XX
- Upon click ***Sell Tool [button]***
    - Create a new record in Sale_Order
        - Create a new, unique sale-order-id
        - Set sale-tool-id to the selected tool-id
        - Set for-sale-date to ‘now()’
        - Set sale-price to 50% of original-price
        - Set sale-clerk-id to current session user-id
    - Update Tool record as “for-sale”

View Sale Status

Task Decomposition



Lock Types: Read-Only User, Read-Only Sale_Order, Read-Only Tool
Number of Locks: 3
Enabling Conditions: Clerk activates form from menu. Tools are for sale or a sale has been completed. Type and Custom Search have matching values in sale order records.
Frequency: Low ~10x / week
Consistency (ACID): Not Critical
Subtasks:

- List of tools either sold or for sale
- Allow search by customer username, sale price and/or keyword

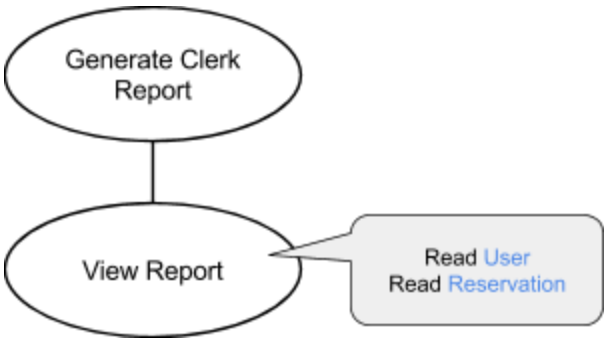
Abstract Code

- Search Sale_Order for active records and return sale-id, sale-tool-id, sale-price, for-sale-date, sale-date, sale-clerk-id and sale-customer-id. Format dates as datetime and sale-price as $X.XX.
- Using sale-tool-id from Sale_Order, read Tool for sub-type and short-desc (aggregate), filtering results that match the selected sub-type in ***Type [radio button]***.
- Using sale-customer-id from Sale_Order, read username from User for the matching user record.
- If the ***Custom Search [input field]*** is not empty, search sale-id, sale-tool-id, short-desc (aggregate), for-sale-date, sale-date, sale-price and username for matching values.
- If a ***Column Header [label]*** has been selected, order the results by the corresponding field for the selected column header, either ascending or descending
- Upon click ***Type [radio button]***
    - Search Tool for matching tool-type values
    - Read tool-id, short-desc (aggregate) from Tool
    - Search Sale_Order for sale-tool-id equal to tool-id
    - Read sale-id, for-sale-date, sale-date, sale-price, sale-clerk-id and sale-customer-id from Sale_Order

    - Format sale-date and for-sale-date as datetime and sale-price as $X.XX.
    - Search User for user-id equal to sale-customer-id
    - Read username from User
- Upon click ***Search***
    - Remove non-alphanumeric characters from search string
    - Read sale-id, sale-tool-id, for-sale-date, sale-date, sale-price, sale-customer-id from Sale_Order
    - Search Tool for tool-id equal to sale-tool-id
    - Read tool-id, short-desc (aggregate) from Tool
    - Search User for user-id equal to sale-customer-id
    - Perform case insensitive search on sale-id, sale-tool-id, sale-clerk-id, short-desc (aggregate), for-sale-date, sale-date, sale-price, tool-type and username for search string

Generate Clerk Report

Task Decomposition



Lock Types: Read-Only User, Reservation
Number of Locks: 2
Enabling Conditions: Clerk activates form from menu
Frequency: Low to Medium - Up to ~ 400 x / day (once on pickup, once on dropoff for all reservations)
Consistency (ACID): Not Critical
Subtasks:

- Generate clerk report with users' employee number, full name, email, hire date, number of pick-ups and drop offs handled this month and sum of two; ordered by number of pick-ups and drop offs

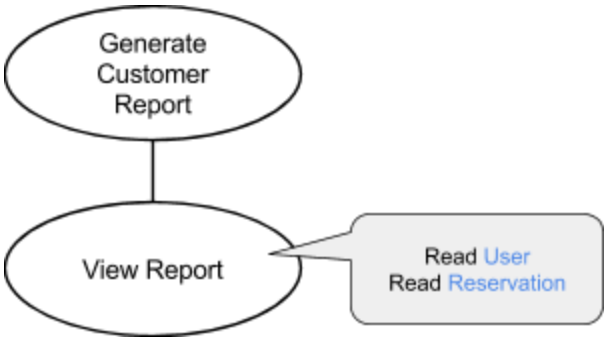
Abstract Code

- Select user-id, first-name, middle-name, last-name, email, hire-date from User
- Search Reservation for pick-up-id, drop-off-id equal to user-id

- Read pick-up-id, drop-off-id from Reservation
- For each user-id, count number of pick-up-id equal to user-id into number-of-pickups, count number of drop-off-id equal to user-id into number-of-dropoffs
- Calculate combined-total by adding number-of-pickups and number-of-dropoffs
- Format hire-date as datetime
- Sort results by combined-total descending
- Upon click ***Back To Report Menu***
    - Leave Clerk Report form
- Upon click ***Reload Results***
    - Select user-id, first-name, middle-name, last-name, email, hire-date from User
    - Search Reservation for pick-up-id, drop-off-id equal to user-id
    - Read pick-up-id, drop-off-id from Reservation
    - For each user-id, count number of pick-up-id equal to user-id into number-of-pickups, count number of drop-off-id equal to user-id into number-of-dropoffs
    - Calculate combined-total by adding number-of-pickups and number-of-dropoffs
    - Format hire-date as datetime
    - Sort results by combined-total descending

Generate Customer Report

Task Decomposition



Lock Types: Read-Only User, Reservation
Number of Locks: 2
Enabling Conditions: Clerk activates form from menu
Frequency: Low to Medium - Up to ~ 400 x / day (once on pickup, once on dropoff for all reservations)
Consistency (ACID): Not Critical
Subtasks:

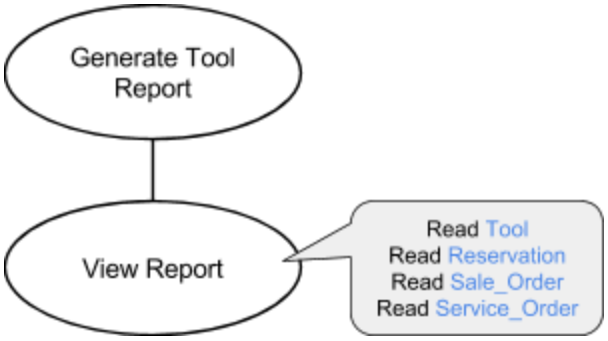
- Generate customer report containing all customers who rented a tool over the last month. Shows username, link to full profile, full names, e-mail address, primary phone number, and number of rentals. List is sorted by number of rentals

Abstract Code

- Read user-id, first-name, middle-name, last-name, email, phone (where primary is true) from User
- Search Reservation for renting-customer-id equal to user-id. Limit records from User by matching records with ‘now()’ - pick-up-date < 30 days.
- Read tool-id, start-date, pick-up-date from Reservation
- For each user-id, count number of renting-customer-id where start-date is not null into number-of-reservations, count number of tool-id where pick-up-date is not null equal into number-of-tools-rented
- Sort results by number-of-tools-rented descending
- Upon click ***Back To Report Menu***
    - Leave Customer Report form
- Upon click ***Reload Results***
    - Read user-id, first-name, middle-name, last-name, email, phone (where primary is true) from User
    - Search Reservation for renting-customer-id equal to user-id. Limit records from User by matching records with ‘now()’ - pick-up-date < 30 days.
    - Read tool-id, start-date, pick-up-date from Reservation
    - For each user-id, count number of renting-customer-id where start-date is not null into number-of-reservations, count number of tool-id where pick-up-date is not null equal into number-of-tools-rented
    - Sort results by number-of-tools-rented descending
- Upon click **View Profile**
    - Navigate to View Customer Profile

Generate Tool Report

Task Decomposition



Lock Types: Read-Only Tool, Reservation, Sale_Order, Service_Order
Number of Locks: 2
Enabling Conditions: Clerk activates form from menu
Frequency: High - Up to ~1240 x / day (after every transaction, for every tool)
Consistency (ACID): Not Critical
Subtasks:

- Generate tool report containing everything in inventory with tool ID, current status of tool, relevant date, description, rental profit, cost of toll and total profit (rental profit: # of days rented times cost/day; total profit: rental profit minus tool cost). List is ordered by total profit

Abstract Code

- Read tool-id, short-desc (aggregate), original-price from Tool
- Select from Reservation where renting-tool-id equals tool-id
- Read tool-id, tool-rental-price, start-date, pick-up-date, end-date, drop-off-date from Reservation
- Select from Sale_Order where sale-tool-id equals tool-id
- Read for-sale-date, sale-date, sale-price from Sale_Order
- Select from Service_Order where service-tool-id equals tool-id
- Read service-start-date, service-end-date, service-cost from Service_Order
- For each tool-id, calculate the number of days rented for each matching reservation record as ceiling(drop-off-date (or ‘now()’ if null) - pick-up-date) * tool-rental-price into rental-profit.
- For each tool-id, calculate the value of total-cost by taking the difference of the original-price and the sum of all matching service order records.
- For each tool-id, calculate the total-profit as the difference between rental-profit and total-cost.
- For each tool-id, calculate the current-status as

    - If matching sale order has sale-order-id not null and sale-date not null, set to “Sold”
        - Set status-date to sale-date
    - If tool has for-sale set to True, set to “For-Sale”
        - Set status-date to for-sale-date
    - If matching service order has service-start-date < ‘now()’ and service-end-date > ‘now()’ or null, set to “In-Repair”
        - Set status-date to service-start-date
    - If matching reservation has start-date < ‘now()’ or pick-up-date < ‘now()’ and end-date > ‘now()’ and drop-off-date is null, set to “Rented”
        - Set status-date to pick-up-date if pick-up-date is not equal to start-date, else start-date
    - If not any other condition, set to “Available”
        - Set status-date to null
- Sort results by total-profit descending
- Format rental-profit, total-cost and total-profit as $X.XX
- Format dates as date (mm/dd/yyyy)
- Upon click ***Type [radio button]***
    - Read tool-id, short-desc (aggregate), original-price from Tool where tool-type matches selected ***Type [radio button]***
    - Select from Reservation where renting-tool-id equals tool-id
    - Read tool-id, tool-rental-price, start-date, pick-up-date, end-date, drop-off-date from Reservation
    - Select from Sale_Order where sale-tool-id equals tool-id
    - Read for-sale-date, sale-date, sale-price from Sale_Order
    - Select from Service_Order where service-tool-id equals tool-id
    - Read service-start-date, service-end-date, service-cost from Service_Order
    - For each tool-id, calculate the number of days rented for each matching reservation record as ceiling(drop-off-date (or ‘now()’ if null) - pick-up-date) * tool-rental-price into rental-profit.
    - For each tool-id, calculate the value of total-cost by taking the difference of the original-price and the sum of all matching service order records.
    - For each tool-id, calculate the total-profit as the difference between rental-profit and total-cost.
    - For each tool-id, calculate the current-status as
        - If matching sale order has sale-order-id not null and sale-date not null, set to “Sold”
            - Set status-date to sale-date
        - If tool has for-sale set to True, set to “For-Sale”
            - Set status-date to for-sale-date
        - If matching service order has service-start-date < ‘now()’ and service-end-date > ‘now()’ or null, set to “In-Repair”
            - Set status-date to service-start-date

        - If matching reservation has start-date < ‘now()’ or pick-up-date < ‘now()’ and end-date > ‘now()’ and drop-off-date is null, set to “Rented”
            - Set status-date to pick-up-date if pick-up-date is not equal to start-date, else start-date
        - If not any other condition, set to “Available”
            - Set status-date to null
    - Sort results by total-profit descending
    - Format rental-profit, total-cost and total-profit as $X.XX
    - Format dates as date (mm/dd/yyyy)
- Upon click ***Search***
    - Remove non-alphanumeric characters from search string
    - Read tool-id, short-desc (aggregate), original-price from Tool where tool-type matches selected ***Type [radio button]***
    - Select from Reservation where renting-tool-id equals tool-id
    - Read tool-id, tool-rental-price, start-date, pick-up-date, end-date, drop-off-date from Reservation
    - Select from Sale_Order where sale-tool-id equals tool-id
    - Read for-sale-date, sale-date, sale-price from Sale_Order
    - Select from Service_Order where service-tool-id equals tool-id
    - Read service-start-date, service-end-date, service-cost from Service_Order
    - For each tool-id, calculate the number of days rented for each matching reservation record as ceiling(drop-off-date (or ‘now()’ if null) - pick-up-date) * tool-rental-price into rental-profit.
    - For each tool-id, calculate the value of total-cost by taking the difference of the original-price and the sum of all matching service order records.
    - For each tool-id, calculate the total-profit as the difference between rental-profit and total-cost.
    - For each tool-id, calculate the current-status as
        - If matching sale order has sale-order-id not null and sale-date not null, set to “Sold”
            - Set status-date to sale-date
        - If tool has for-sale set to True, set to “For-Sale”
            - Set status-date to for-sale-date
        - If matching service order has service-start-date < ‘now()’ and service-end-date > ‘now()’ or null, set to “In-Repair”
            - Set status-date to service-start-date
        - If matching reservation has start-date < ‘now()’ or pick-up-date < ‘now()’ and end-date > ‘now()’ and drop-off-date is null, set to “Rented”
            - Set status-date to pick-up-date if pick-up-date is not equal to start-date, else start-date
        - If not any other condition, set to “Available”
            - Set status-date to null

- Perform case insensitive search on tool-id, current-status, status-date, short-desc (aggregate), rental-profit, total-cost, total-profit for search string
- Sort results by total-profit descending
- Format rental-profit, total-cost and total-profit as $X.XX
- Format dates as date (mm/dd/yyyy)