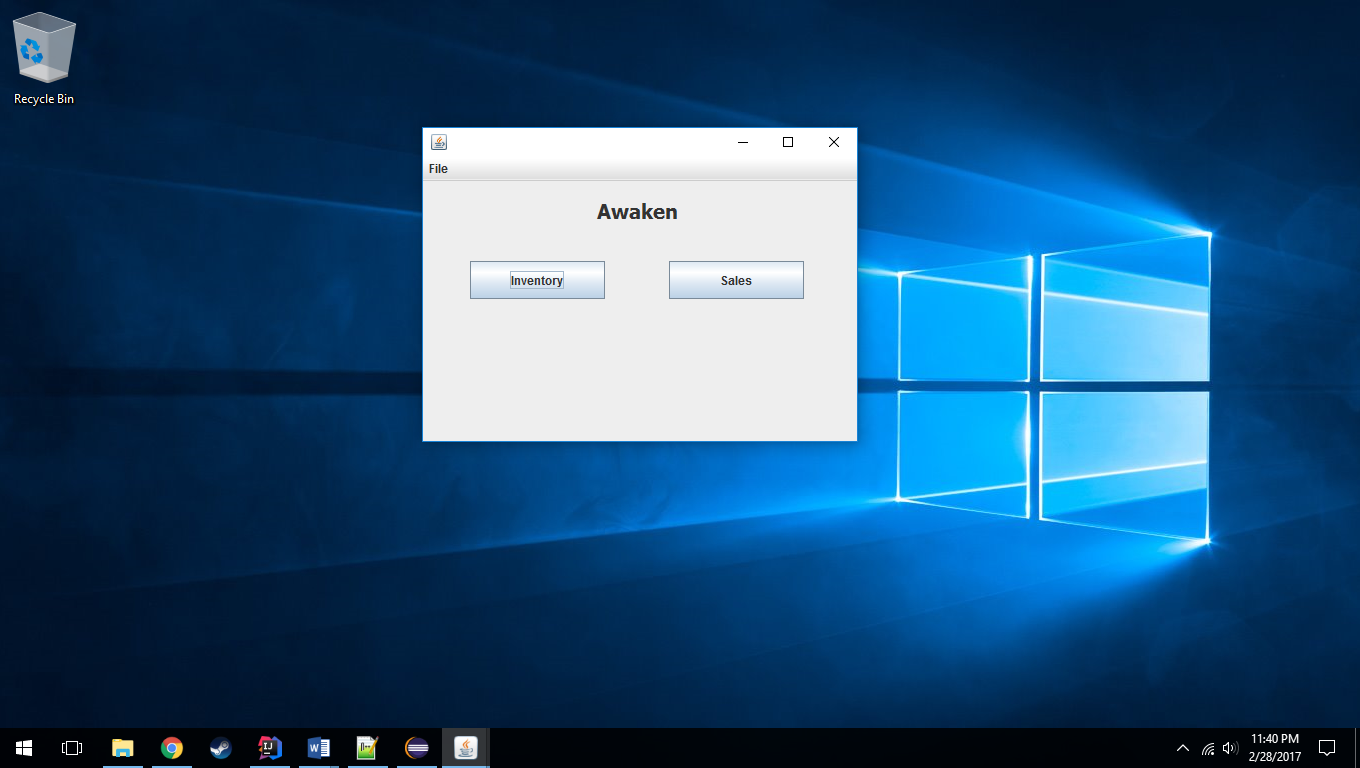
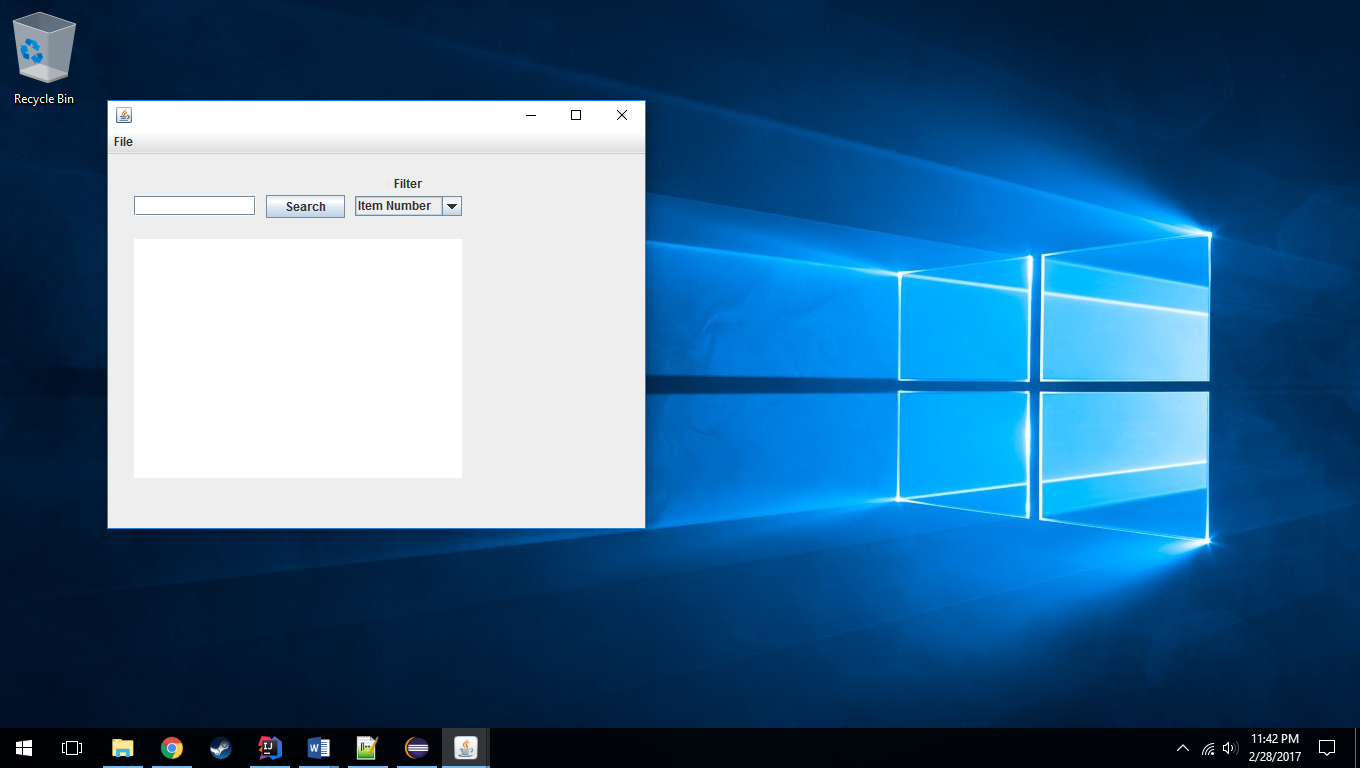
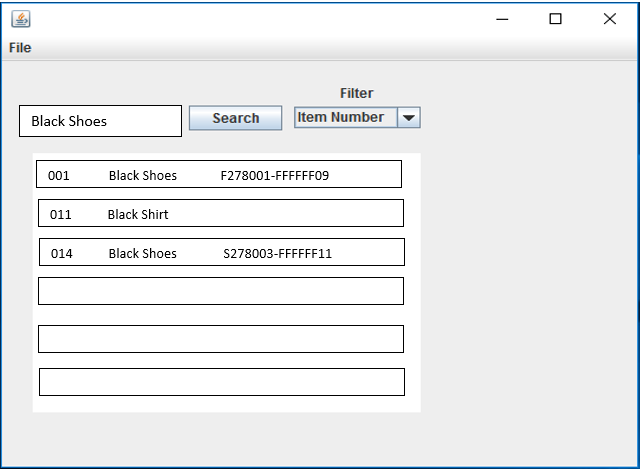
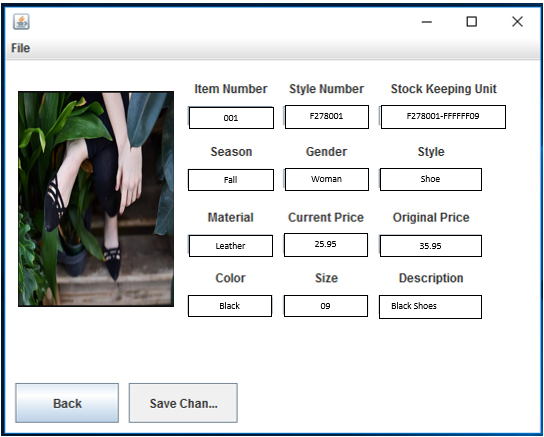
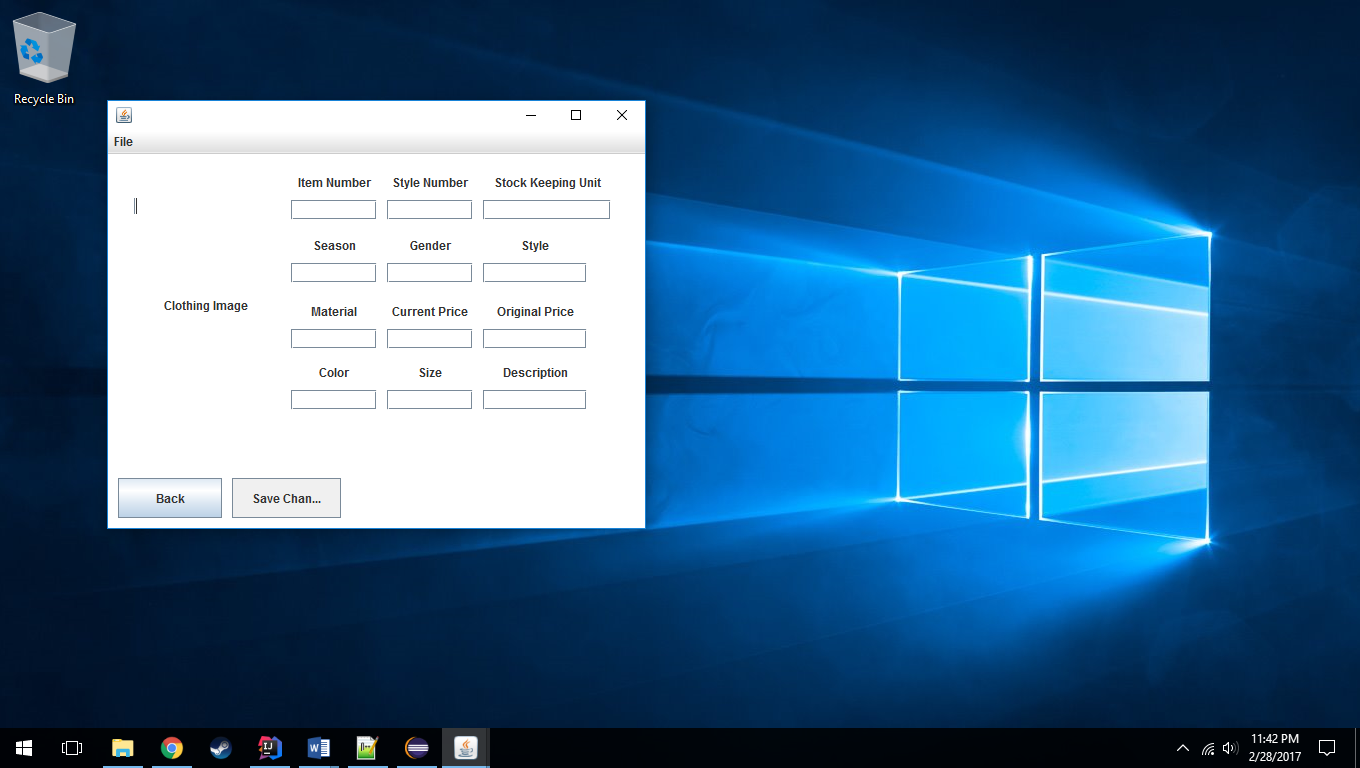
1. Introduction:
   1. We are creating a software solution to the problem of inventory management and sales information management using a web-based GUI, local backend Java, a web-based database and an online retail hosting site.
2. Assumptions:
   1. All future users will be familiar with the clothing industry as well as finance and marketing.
   2. Software will be used only on one system located in a business environment and will not need to be scaled at any point for any reason other than increased inventory quantity and increased desire for sales manipulation
3. Glossary:
   1. Style Number (Season (F or S)-Gender(1xxxxx)-Style(x1xxxx)-Material(xx1xxx)-Iteration(xxx001))
   2. Stock Keeping Unit (Style Number-Color (hex)-Size)
   3. OSS (Out of Stock)
   4. OS (Overstock)
4. Operating Environment:
   1. Desktop access only, local Java and HTML5/CSS3/JQuery code, will run in the hosts browser but will run locally and only connect to the internet for database storage and charting in google charts
5. Interfaces:
   1. Home

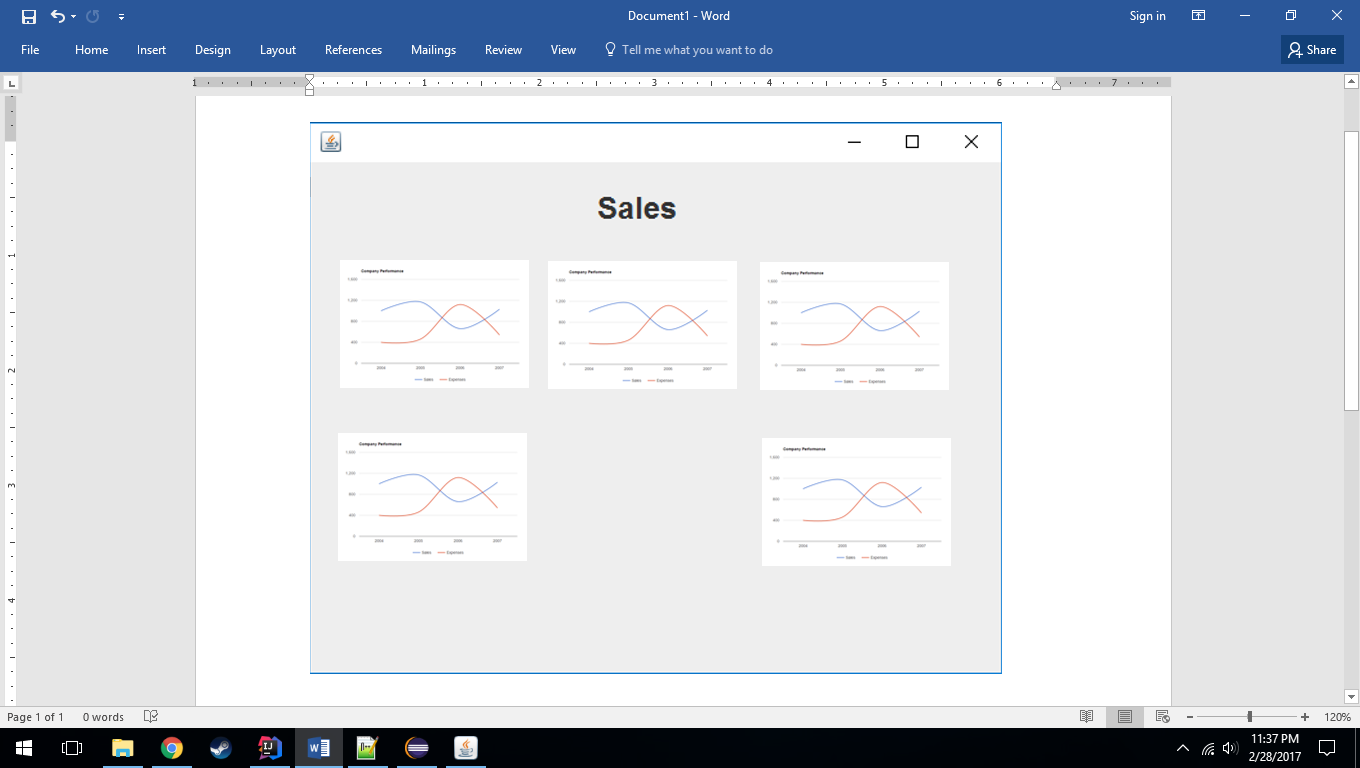


* 1. Inventory

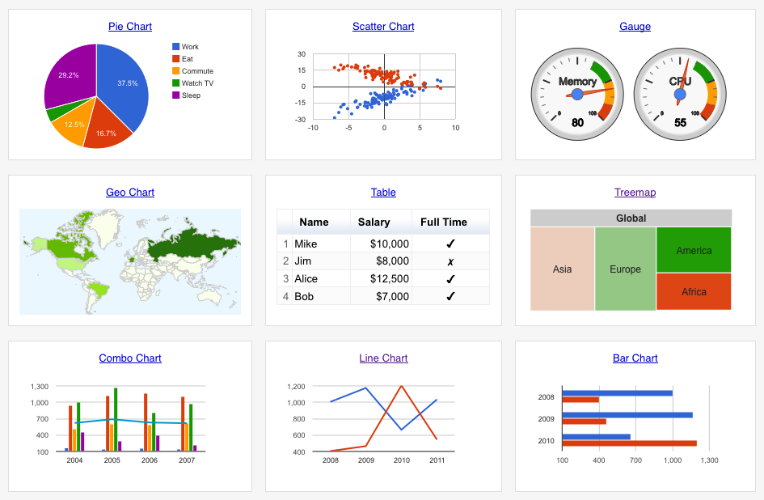


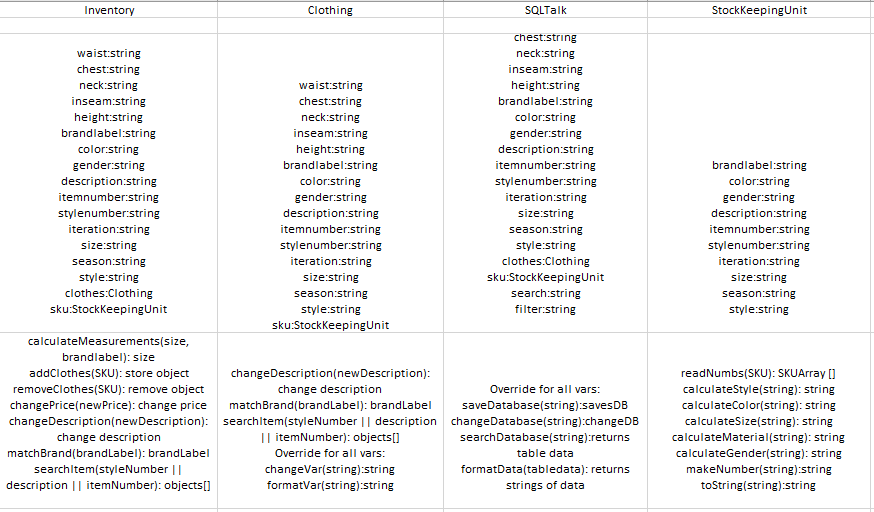


* 1. Sales



Examples of charts that can be called when relevant sales/inventory data exists



1. Object-Oriented Analysis:
   1. 
   2. Search an item
      1. This function takes in a search string from the user. It then parses the string and looks for any parts of the string used in both the search an in the database in the item number, style number, SKU, style, and description. Returns an array of clothing objects that can be displayed in a list with their toString methods.
   3. View an item’s information
      1. When an item from the returned array of item is selected, the information for that item is displayed in text fields. We accomplished this using several different call variable methods that formatted each string for output.
   4. Edit an item
      1. The text fields displaying the information are editable. These pass changes through any one of many edit variable methods that change the variable in the object.
   5. Save an item
      1. Pressing the save button calls a save method that takes the clothing object open, formats it with a format method and saves that to the database. A confirmation window will appear.
   6. Filter the search
      1. Any filter entered during the search step will be used as a qualifier and call a method to check for only clothes with that variable.
   7. View the sales dashboards
      1. Using google charts code in HTML5/CSS3/JQuery the dashboards will display as soon as the screen is displayed
   8. Alter dashboard 1
      1. Google Charts allows for the creation of editable graphs that can take in new local information via SVG file
   9. Alter dashboard 2
      1. Google Charts allows for the creation of editable graphs that can take in new local information via SVG file
   10. Alter dashboard 3
       1. Google Charts allows for the creation of editable graphs that can take in new local information via SVG file
   11. Alter dashboard 4
       1. Google Charts allows for the creation of editable graphs that can take in new local information via SVG file
   12. Alter dashboard 5
       1. Google Charts allows for the creation of editable graphs that can take in new local information via SVG file
2. Non-functional Requirements:
   1. Must be Java SE 8, HTML5, CSS3 or greater
   2. Nothing runs in parallel in this system
   3. Nothing can run in parallel in this system
   4. All access will be granted to all users and will be controlled locally by a door
3. Traceability Matrix: