**Final Project Report**

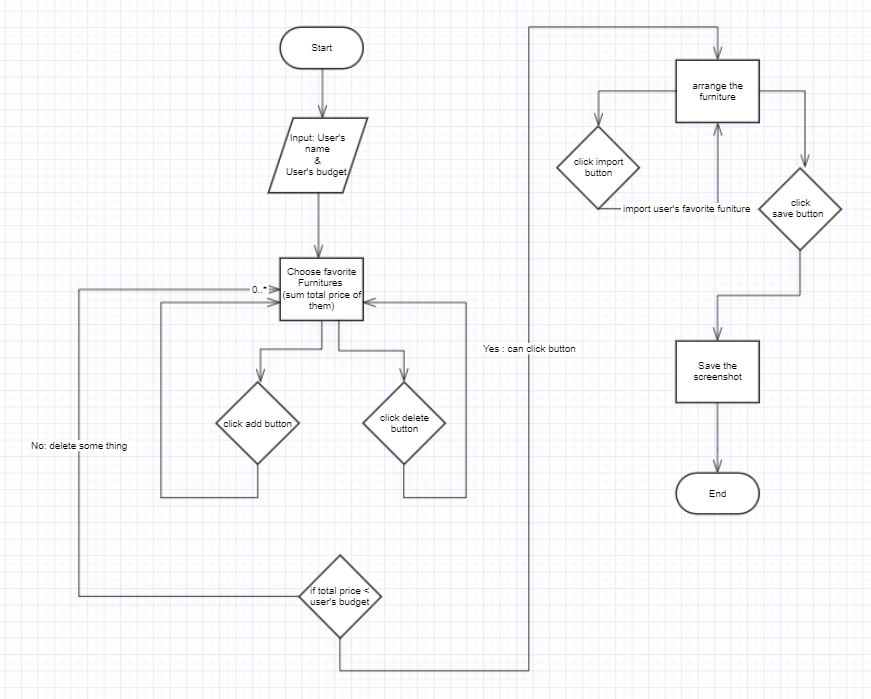
Student Name: 2019314680

Student ID: Son Sejeong

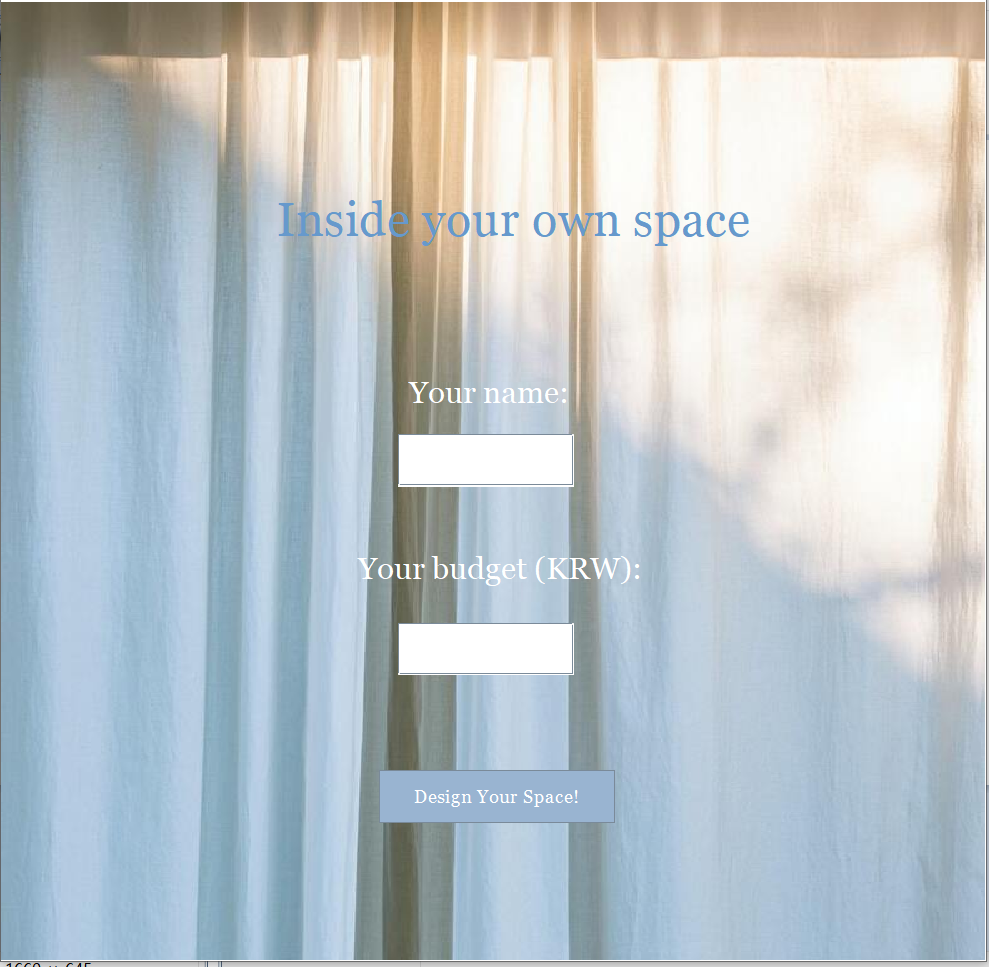
**1. Briefly describe the project purpose:**

Recently I moved, so I was interested in furniture and interior design. To apply these interests to the Java Project, I try to make a GUI which can check furnitures I liked and store a image after arranging them in GUI frame. It's for someone who moves in like me or tries to decorate their own room for a change.

**2. Draw the logic flow of the program (with flowchart):**

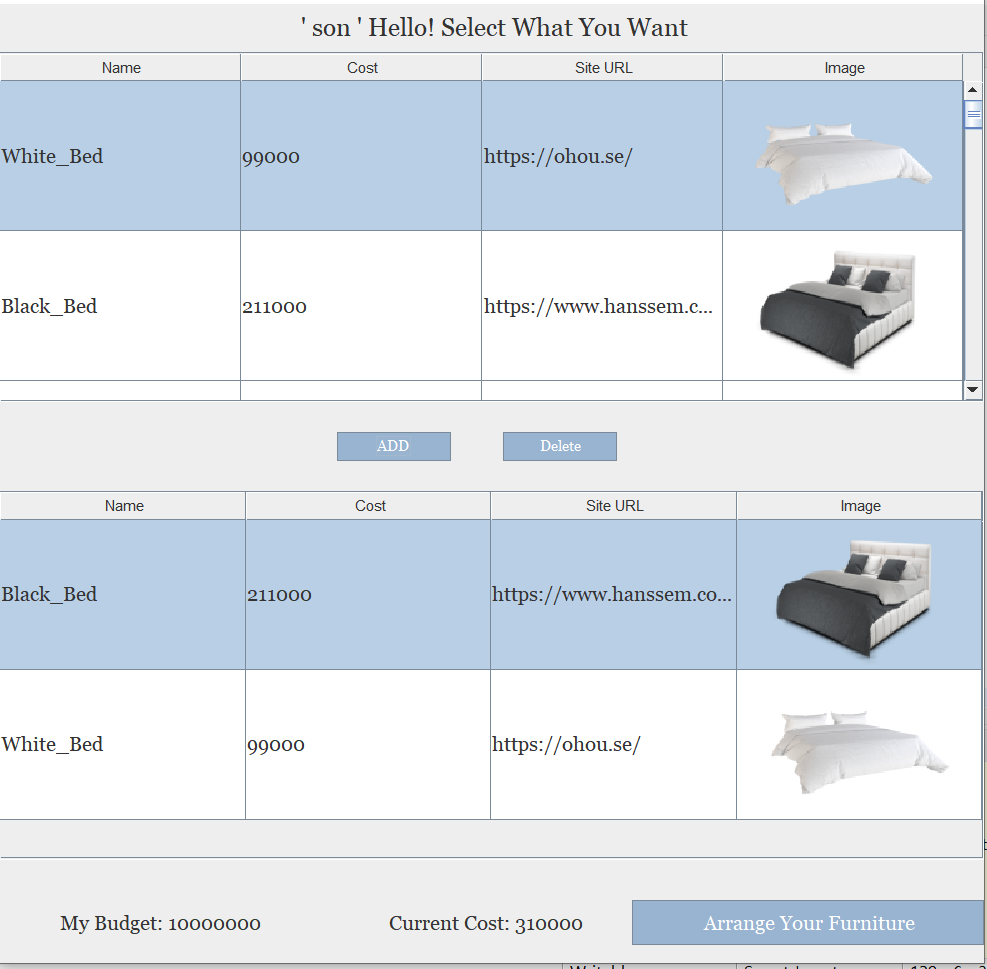
****

**3. Provide screenshots for each screen with brief description:**



In first screen, you can enter user name and your budget to buy furnitures.

If you don't enter any of them, the system returns message to ask fill the blank.



On the top, you can see a list of furnitures. If you want to buy some of them, you can choose it and add to your own favorite list. But if the sum of cost exceeds your budget, program returns the warning message to notice you to overspend budget.



From your Favorites list, you can import the images of funitures you want.

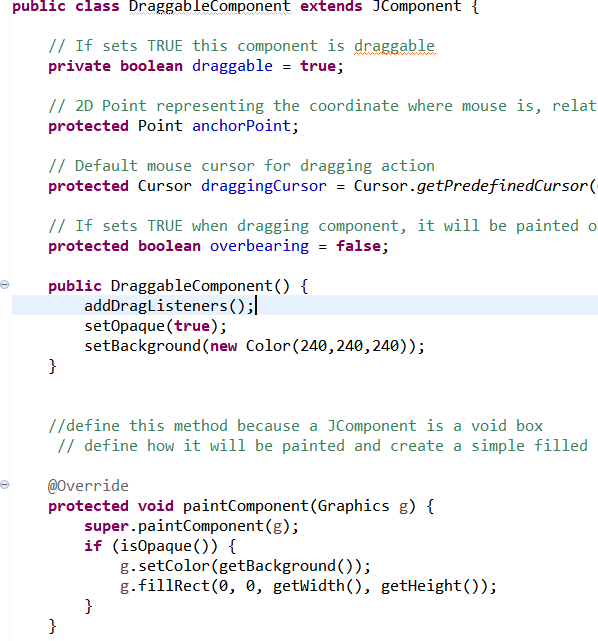
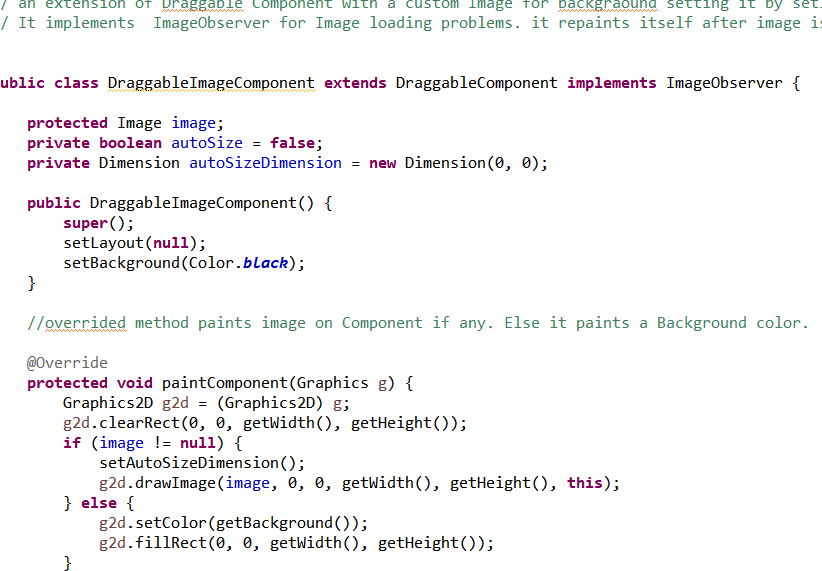
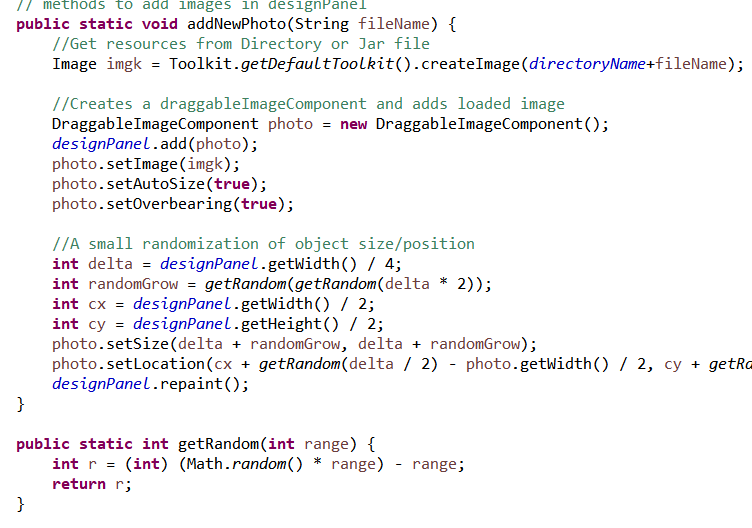
Also, images are draggable. So you can arrange the images on the screen to simulate your room. After arrange, click a button to save the screenshot.

**4. Explain the code of the main functionalities**

This gui has three main functions. First, it can check and store the furnitures that you want. So I use Jtables and Jbuttons to implement this function. After make a table which store total information of furnitures, I implement two buttons, to add and to delete. For buttonAdd, take a data of a row which i selected and put them in to second table. Also sum the price of furnitures and show it in the label which is in the button of frame. In the contrast for buttonDelete, delete a data of a row which i selected and substract the price of furniture from the current cost.



Second, you can drag the images of furnitures and arrange them on the frame. For draggable images, i have to define new classes about image attributes. So extending Jcomponent, allocate the functions to be draggable. From these two classes, now can make images whatever you want to take and arrange them on the frame.



Third, you can save the screenshot that you simulate your room. So, I use BufferedImage to make new image data and save it. paint() method make new image and ImageIO.write() helps to save the img to current directoty.



**5. Explain what is included in your project and why it is used (Polymorphism, Inheritance, File I/O, etc)**

This GUI attempted to follow **the MVC structure, which is one of the OOP designs**. The Model-View-Controller is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application.

First, the Model component corresponds to all the data-related logic that the user works with. In this GUI, Class TakeTable works the function of the Model component. It directly manages the data about furniutures, logic and rules of the application .

Next, the View component is used for all the UI logic of the application. In this GUI,

Class View totally have a role of the View component. It shows any representation of information such as tables, images, and buttons to user to handle and get what users want.

Finally, controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. Class DraggableImageComponet and Class DraggableComponent works a role of controllers. These two classes are related to make a image draggable in the View and manipulate data from the Model.

**File I/O** was also used to load images and text data. In terms of input, I use a txt file to take data of furnitures. Also, I save a screenshot of arranged furnitures. This is a kind of File output.

**Threading** is also used to set view. With EventQueue.invokeLater method, it post an event at the end of Swings event list and is processed after all previous GUI events are processed. And when the program import the images, EventQueue. invokeLater method works again to post the event to import the images.

In detail, displaying the furnitures on the screen is an example of **inheritance and polymorphism**.In terms of inheritance, to make a image draggable, Class DraggableImageComponet inherits Class DraggableComponent. From the view of ploymorphism, to generate each different functions of Panels, ImagePanel extends JPanel which is normally used. So in this project, each Panels works with different functions