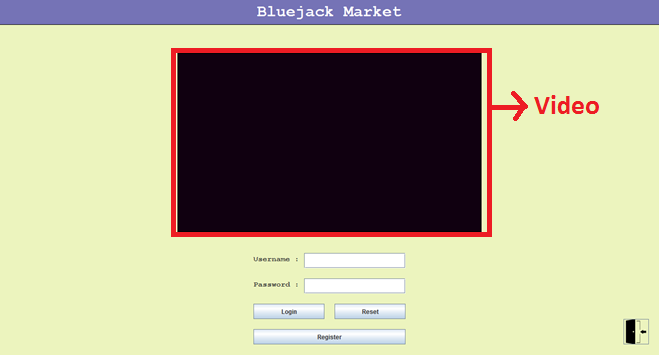
**Bluejack Market**

**Bluejack Market** is a company that sell many items that people need. As a programmer, you asked to make an application for handle user transaction using **Java Programming** for **Bluejack Market**.

1. **Login Page**

****

The program will display a **Login Page** when started.

**List of components:**

* **JFrame** for **Login Window**
* **JPanel** for **panels**
* **JLabel**
* **Text**: Username and Password
* **JTextField** to input **username**
* **JPasswordField** to input **password**
* **JButton** for **Login** **Button**, **Reset Button**, **Register Button**, **Exit Button** (icon: exit.png)
* **Video** todisplay promotional video about **Bluejack Market** (video: about.mpg)

**Description:**

* If user clicks the **Login Button**, the program will
* Validate **username cannot be empty**. If **username** is **empty**, show message “**Username must be filled**”.
* Validate **password cannot be empty**. If **password** is **empty**, show message “**Password must be filled**”.
* Validate **username and password** **must be valid** according to the registered users list (when user registers, his / her username and password will be saved in the list). If username and password is **not valid**, show message “**Username or password is invalid**”.
  + If all fields are **validated** **successfully**, the **Login page** will be **closed** and then **Main page** will be displayed.
* If user clicks the **Reset Button**, the program will
  + Clear all fields.
* If user clicks the **Register Button**, the program will
  + Closethe **Login page** and open **Register page**.
* If user clicks the **Exit Button**, the program will
  + Close the application.

1. **Register Page**



The program will display a **Register Page** used to register new user data

**List of components:**

* **JFrame** for **Register Window**
* **JPanel** for **panels**
* **JLabel**
* **Text**: Username, Password, and Confirm Password
* **JTextField** to input **username**
* **JPasswordField** to input **password** and input **confirm password**
* **JButton** for **Register Button**, **Reset Button**, **Back Button**, **Exit Button** (icon: exit.png)

**Description:**

* If user clicks the **Register Button**, the program will
* Validate **username cannot be empty**. If **username** is **empty**, show message “**Username must be filled**”.
* Validate **password cannot be empty**. If **password** is **empty**, show message “**Password must be filled**”.
* Validate **password must be alphanumeric**. If **password** is **not alphanumeric**, show message “**Password must be alphanumeric**”.

Example:

abcabc **(invalid)**

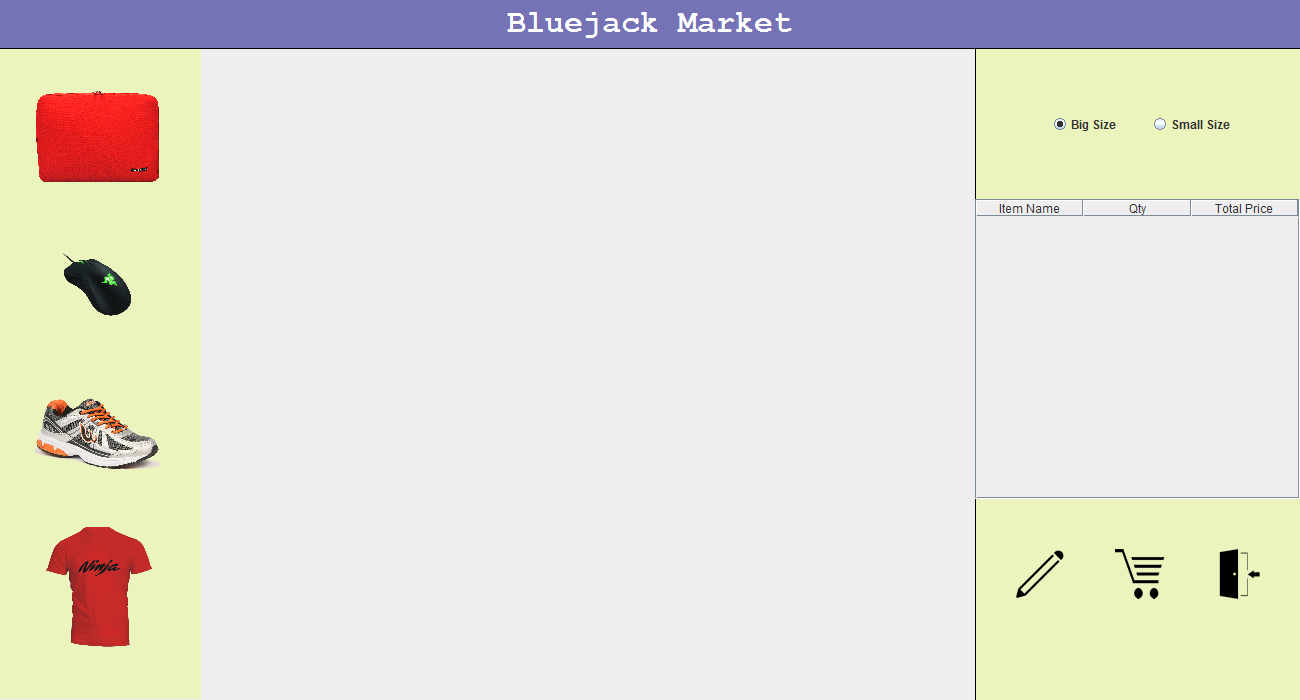
123123 **(invalid)**

abc123 **(valid)**

123abc **(valid)**

* Validate **confirm password must match with password**. If **confirm password** **doesn’t match with password**, show message “**Confirm Password must be filled**”.
  + If all fields are validated **successfully**, register the user data in the system. Then, the **Register page** will be **closed** and **Login page** will be **displayed**.
* If user clicks the **Reset Button**, the program will
  + Clear all fields.
* If user clicks the **Back Button**, the program will
  + Closethe **Register page** and open **Login page**.
* If user clicks the **Exit Button**, the program will
  + Close the application.

1. **Main Page**



**List of components:**

* **JFrame** for **main window**
* **JTable** for **list of** **transactions**
* **JRadioButton** for choosing **image size** (**big size** and **small size**)
* **JLabel**
* **Icon**: Edit Icon (edit.png), Buy Items Icon (shop.png), Exit Icon (exit.png)

**Descriptions:**

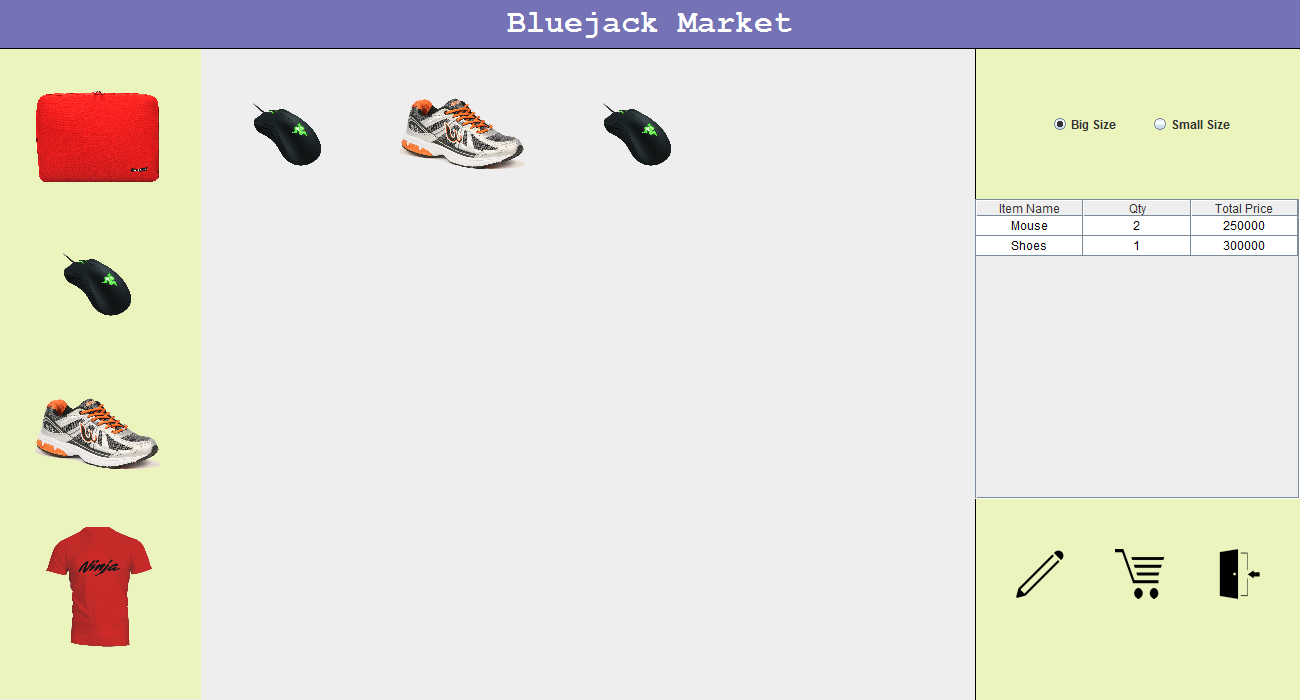
* There is **music** that will be **played during the form opens** (music: canon.wav)
* Display **list of available items** to buy using **Graphics2D**. There are **4 currently available items** in the store (the items list could be edited in the Edit Items page):

|  |  |  |
| --- | --- | --- |
| **Name** | **Price** | **Image** |
| Notebook Case | 25000 | case.png |
| Mouse | 125000 | mouse.png |
| Shoes | 300000 | shoes.png |
| T-Shirt | 75000 | t-shirt.png |

* Draw **a line** for separating **between list of available items and user’s cart**
* Every **item on list item** have the following effect when **dragged by mouse**:
* The dragged item will follow the user’s mouse cursor.
* Validate **user** must **put the image** that he / she drags **inside** the **cart area**.

****

* If user **puts the image inside the cart:**
* If the **cart already contains 16 items**, show message “**The cart is full**” and **do not put the item** inside the cart.
* If the **item has not added to the cart before**, **insert a new row** in the table.
* If the **item** has already **added to the cart**, **update the quantity** and **total price** in the table.



* There are **2 kinds of image size**:
* **Big** (default image size)

Width:125 px

Height: 125 px

* **Small**

Width: 0.75 x Big size (93.75 px)

Height: 0.75 x Big size (93.75 px)

* When user **changes** item thumbnail size (using the **RadioButton**)**:**
* If user choose “**Big Size**”, then the **size of the icon** will be **Big.**
* If user choose “**Small Size**”, then the **size of the icon** will be **Small.**

**Notes:** You must use affine transform, for transformation between Big and Small image size.

* If user clicks **Buy Icon**, the program will:
* If the table is **empty**, then show message “**There is no item**”
* If the User’s cart is **not empty**, then **show** message “**You will buy [Total Price]**” with confirm dialog “**Yes**”, “**No**”, and “**Cancel**”

**Total Price = Total of Price of all Items in the user’s cart**

* If user choose “**Yes**”, then show message “**Buy Success**”, **clear table’s data**, and **clear user’s cart**
* If user choose “**No**” or “**Cancel**”, then **nothing happen** and all **data are not changed**.
* If user clicks **Edit Icon**, the program will:
  + Closethe **Main page** and open **Edit Items** **page**.
* If user clicks **Exit Icon**, the program will:
* The program will be **closed**.

1. **Edit Items Page**



The program will display an **Edit Items Page** used to edit the items list shown in the left panel on the Main page.

**List of components:**

* **JFrame** for **Edit Items Window**
* **JPanel** for **panels**
* **JLabel**
* **Text**: Item Image, Item Name, and Item Price
* **Images** to display **4 items** in the list.
* **JTextField** to show **item image**, input **item name**,and **item price**
* **JButton** for **Remove Buttons**, **Reset Button**, **Add Button**, **Back Button**

**Description:**

* When the form opens for the first time, the **initial states** of the form are:
  + **Item Image** text field is **disabled**.
  + **Items Images** will be shown with a **Remove Button** below the Images.
  + **Item Name** and **Item Price** fields are **empty**.
* If user clicks the **Remove Button**, the program will
* Remove the item from the items list.
* If user clicks the **Browse Button**, the program will
  + Open a File Chooser for selecting the new item image.
  + If user selects **Open** Button **in the File Chooser dialog**:
    - Save the selected file’s path in the Item Image text field.
  + Otherwise:
* The Item Image text field will not be changed.
* If user clicks the **Reset Button**, the program will
  + Reset the form to the initial state.
* If user clicks the **Add Button**, the program will
* Validate **item name cannot be empty**. If **item name** is **empty**, show message “**Item Name must be filled**”.
* Validate **item price cannot be empty**. If **item price** is **empty**, show message “**Item Price must be filled**”.
  + Validate **item price must be a number**. If **item price** is **not a number**, show message “**Item Price must be a number**”.
  + Validate **item image must be a** “**jpg**” **image or** “**png**” **image**. If **item image** is **not** a “**jpg**” **or** “**png**” image, show message “**Item Image must be ‘jpg’ or ‘png’**”.
  + Validate **there are less than 4 items in the items list**. If there are **already 4 items** in the items list, show message “**Items List is full**”.
  + If all fields are validated **successfully**, register the items data in the system. Then, the **Edit Items page** will **reset back to the initial state**.
* If user clicks the **Back Button**, the program will
  + Closethe **Edit Items page** and open **Main** **page**.

Here are the rules that you must follow to create your project:

1. Use appropriate software for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya.
2. Use the techniques taught during practicum.
3. Collect appropriate files for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya.
4. Include the other files that can support your project, such as:
   * All files in your project
   * Other files (image, audio, video, etc.) used in your project
   * \*.DOC file (documentation of your project) that contains the reference links of additional files (image, audio, video, etc.) used in your project

If there are some hidden creativities, please note them in the existing documentation because they can greatly affect your project score

**References:**

**Images:**

* https://encrypted-tbn3.gstatic.com/images?q=tbn:ANd9GcSHASugpCVM0rVDuOoqJGTHo-2HY84ghV5CSJ7u29zpI8T8HrTD
* https://encrypted-tbn2.gstatic.com/images?q=tbn:ANd9GcQJ787YonOCQDXM0EHdEeqp7KLjUY0NyMdCv9aj2UXBuIi1Kq2k
* http://pngimg.com/upload/running\_shoes\_PNG5827.png
* http://assets.razerzone.com/eeimages/products/6924/razer-deathadder-gallery-3.png
* https://2.bp.blogspot.com/-7X1pFeid\_4Q/V1pl2Jo1yZI/AAAAAAAACgg/AmNTU\_G-h6YFvuLtPHEq52he-oAmrgsKwCLcB/s1600/ninja%2Btransparan.png
* http://icons.iconarchive.com/icons/icons8/ios7/256/Editing-Edit-icon.png
* https://s0.bukalapak.com/img/588066861/large/1.jpg

**Music:**

* http://freemusicarchive.org/music/Steve\_Combs/Masters\_Remastered/02\_Canon\_in\_D