

The background is a light beige color with various green foliage and abstract shapes. There are several green leaves and branches scattered around the edges. In the top left, there are concentric blue circles. In the bottom left, there are dark green dots. In the bottom right, there are dark green abstract shapes.

Burnett's Landscaping

Faith Burnett
CIS355a
DeVry University

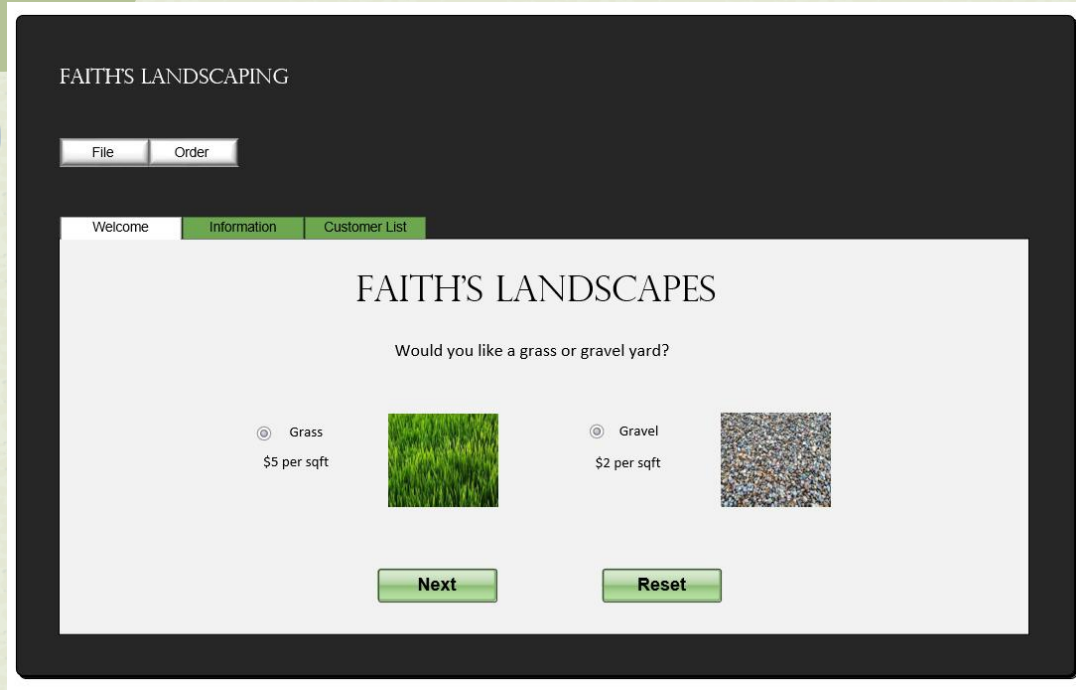
Introduction

Part 1	Design application by building wireframe and uml diagram
Part 2	Begin creating the GUI starting with the information page
Part 3	Continue building out the GUI by creating the welcome page
Part 4	Combining frames into single frame using panels and creating the customer page
Part 5	Setup file storage to track orders and customer information
Part 6	Setup database for storing orders and customer information to replace file storage

Design

- Create wireframe to map out components location and to serve as a guide in the design process
- Create UML diagram to organize data to serve as a guide for building classes and storing data in either file or database

Wireframe



- The welcome page is the first pane created
- Radio buttons are added to give customer landscaping option
- To navigate panes the next button is added
- Reset will be used to unselect radio button for fresh order

Wireframe

FAITH'S LANDSCAPING

File Order

Welcome Information Customer List

FAITH'S LANDSCAPES

Please enter your information:

Name:

Address:

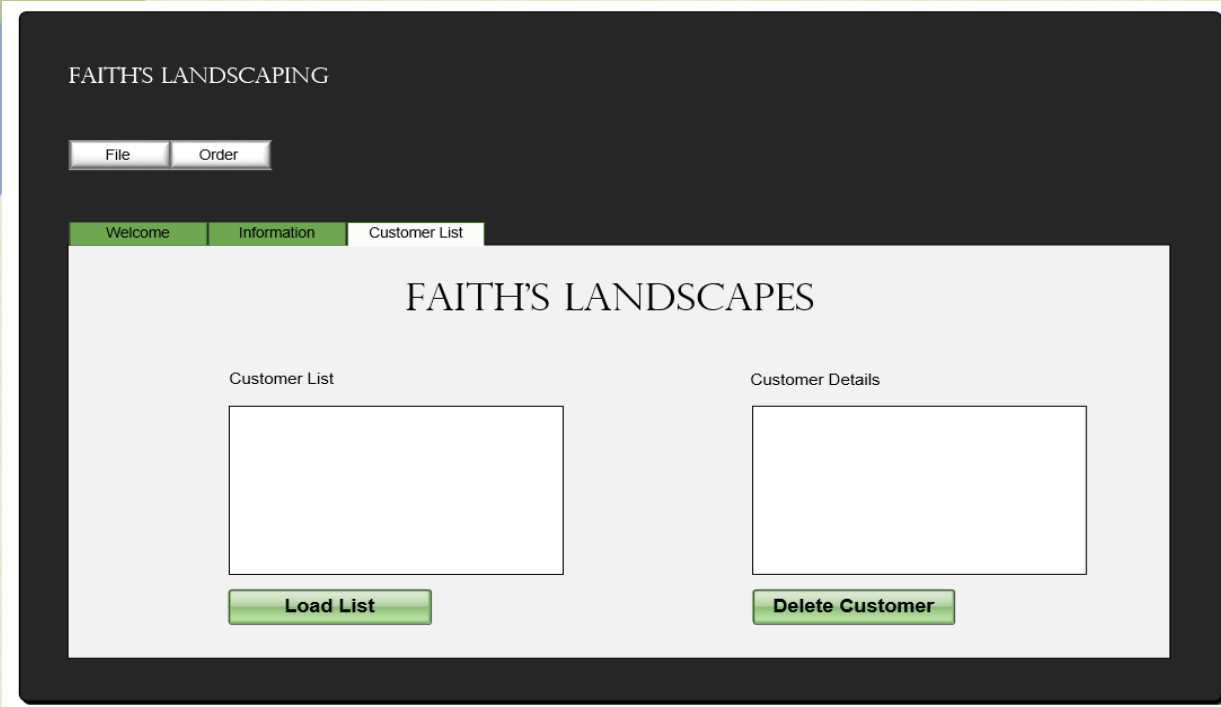
Enter the length and width of your yard:

Length (ft): Width (ft):

Order Summary:

- The information pane is created
- Labels and textfields are added for customer information to be entered
- Calculate button is needed for creating the order total and for displaying details

Wireframe



- The customer details page is created
- Listbox is needed to display individual customers
- Textarea is needed to display customer order details
- To retrieve customers the load list button is added
- Delete button will be needed to delete customer after selection

UML

- In the customer class the customer must include yard specifications and location as well as name and the totalCost attribute to store the cost of an order
- To display details for user we need a method for outputting the details and retrieving the details
- In the data class we need attributes for database connection including name of the database and login information
- Methods to add, delete and retrieve customers are needed for the data class

Customer



```
-customerId:int  
-name:String  
-address:String  
-yardType:String  
-length:double  
-width:double  
-totalCost:double  
+toString():String  
+getDetails():String
```

DataIO

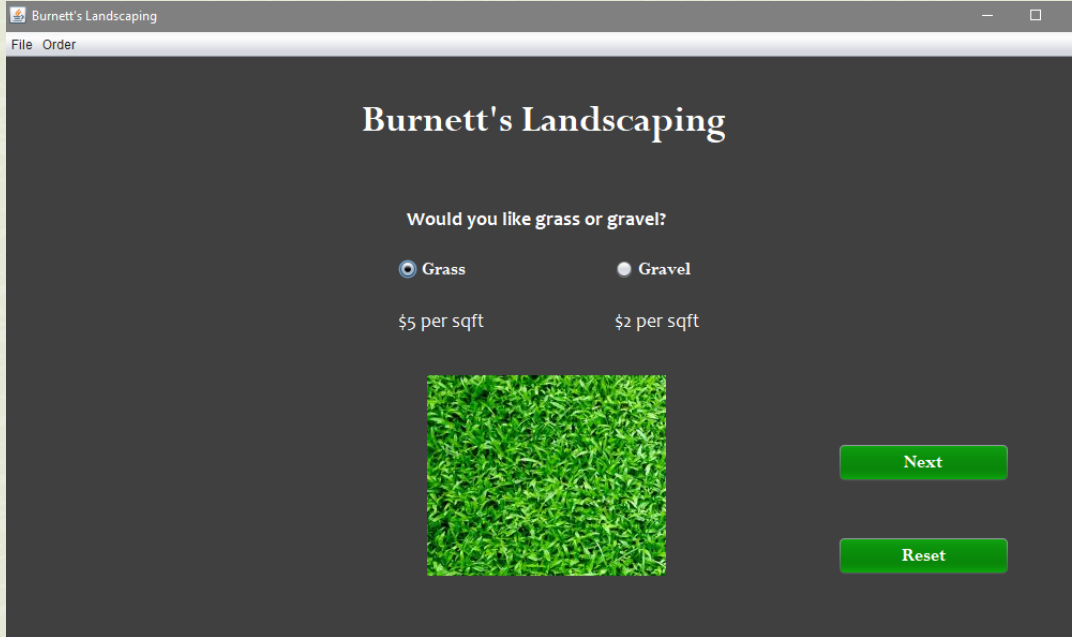
```
-DATABASE_NAME:String  
-CONNECTION_STRING:String  
-USER_NAME:String  
-PASSWORD:String  
+add(cust:Customer):void  
+delete(customerId:int):void  
+getList():Array List<Customer>
```


Welcome Page



- Create a new JFrame for entering customer order options
 - Create navigation from frame to frame
 - Perform a reset on selected options for new order
- 
- 

Select Option



The image shows a web browser window titled "Burnett's Landscaping". The browser's address bar shows "File Order". The web page has a dark grey background. At the top, the text "Burnett's Landscaping" is displayed in a white serif font. Below this, the question "Would you like grass or gravel?" is shown in a white sans-serif font. There are two radio button options: "Grass" with a selected radio button and "Gravel" with an unselected radio button. Below the "Grass" option, the price "\$5 per sqft" is listed. Below the "Gravel" option, the price "\$2 per sqft" is listed. A square image of green grass is positioned below the "Grass" option. At the bottom right of the form, there are two green buttons with white text: "Next" and "Reset".

Burnett's Landscaping


File Order

Burnett's Landscaping

Would you like grass or gravel?

☒ Grass ☐ Gravel

\$5 per sqft \$2 per sqft



Next

Reset

The user will select a landscape option when next is pressed you are directed next frame to enter customer information and calculate order details. If reset is hit option selected is cleared for a fresh order to be placed.

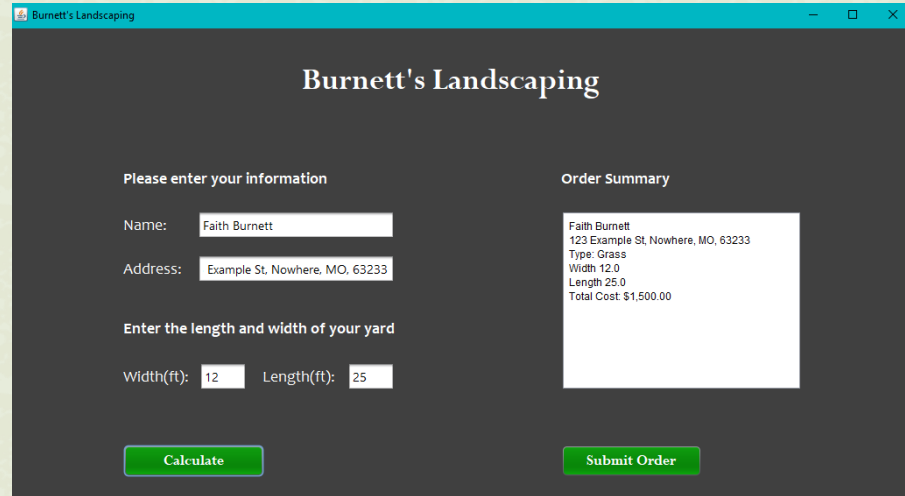


Information Page

- Using the UML class diagram for customer build a java class
- Create a new JFrame for entering customer order information
- Create an instance of the customer object
- Perform validation on input fields from JFrame

Preparing Order

At this stage of the application user input is retrieved and the cost of the order is calculated and displayed when calculate button is pressed. Submit is not functional at this time.



The screenshot shows a web application window titled "Burnett's Landscaping". The interface is divided into two main sections: "Please enter your information" and "Order Summary".

Please enter your information

Name:

Address:

Enter the length and width of your yard

Width(ft): Length(ft):

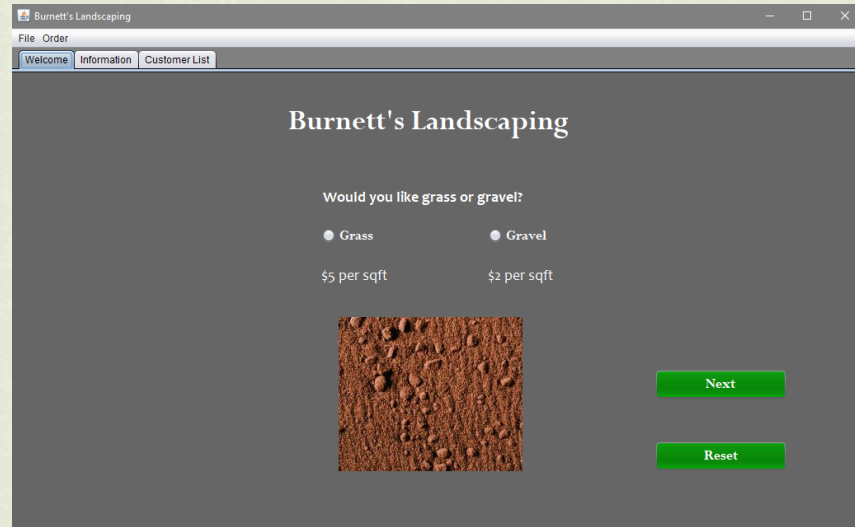
Order Summary

Faith Burnett
123 Example St, Nowhere, MO, 63233
Type: Grass
Width 12.0
Length 25.0
Total Cost: \$1,500.00

At the bottom, there are two buttons: "Calculate" (green) and "Submit Order" (green).

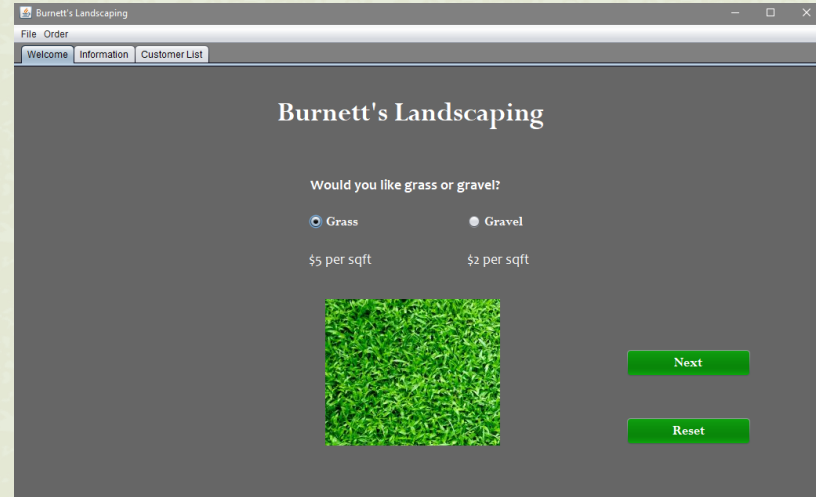
Completed Design

After adding a menu bar and tabbed panel all frames are now included in this one frame.

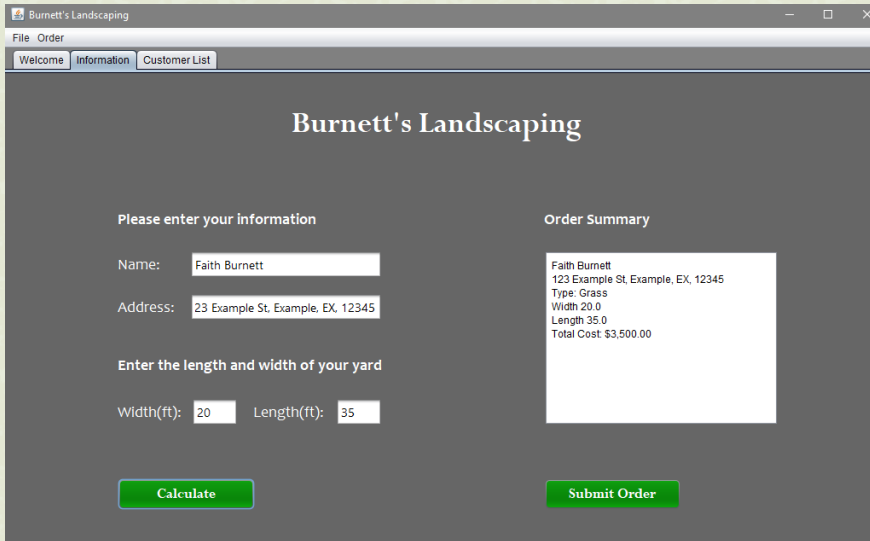


Current Functionality

When the application is opened the user has the option to select a grass landscape or gravel landscape then the user will hit next.



Current Functionality cont..



The screenshot shows a web browser window titled "Burnett's Landscaping". The browser's address bar shows "File Order". The application has three tabs: "Welcome", "Information", and "Customer List". The "Information" tab is active. The page has a dark grey background with white text. The title "Burnett's Landscaping" is centered at the top. Below the title, there are two main sections. The left section is titled "Please enter your information" and contains three input fields: "Name:" with the value "Faith Burnett", "Address:" with the value "23 Example St. Example, EX, 12345", and "Enter the length and width of your yard" with two sub-inputs: "Width(ft):" with the value "20" and "Length(ft):" with the value "35". Below these inputs are two green buttons: "Calculate" and "Submit Order". The right section is titled "Order Summary" and contains a white box with the following text: "Faith Burnett", "123 Example St. Example, EX, 12345", "Type: Grass", "Width 20.0", "Length 35.0", and "Total Cost \$3,500.00".

Burnett's Landscaping

Please enter your information

Name:

Address:

Enter the length and width of your yard

Width(ft): Length(ft):

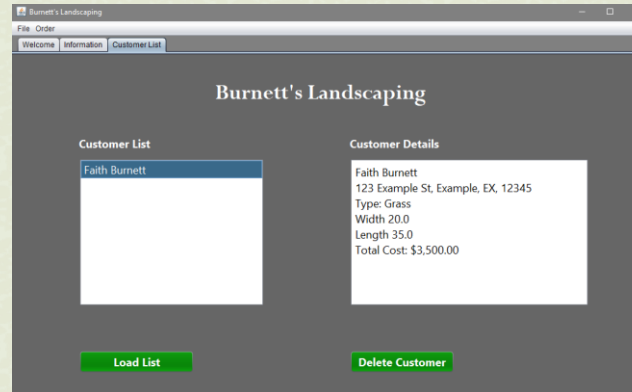
Order Summary

Faith Burnett
123 Example St. Example, EX, 12345
Type: Grass
Width 20.0
Length 35.0
Total Cost \$3,500.00

From the welcome page the user is navigated to the information page where customer details are entered and the price of the order is calculated by pressing calculate. Once submit order is pressed they are navigated to the customer page.


Current Functionality cont..

From the information page here the customer that was added is displayed and the details of the order are displayed. When you select a customer from the list you can hit delete and that customer order will be deleted.

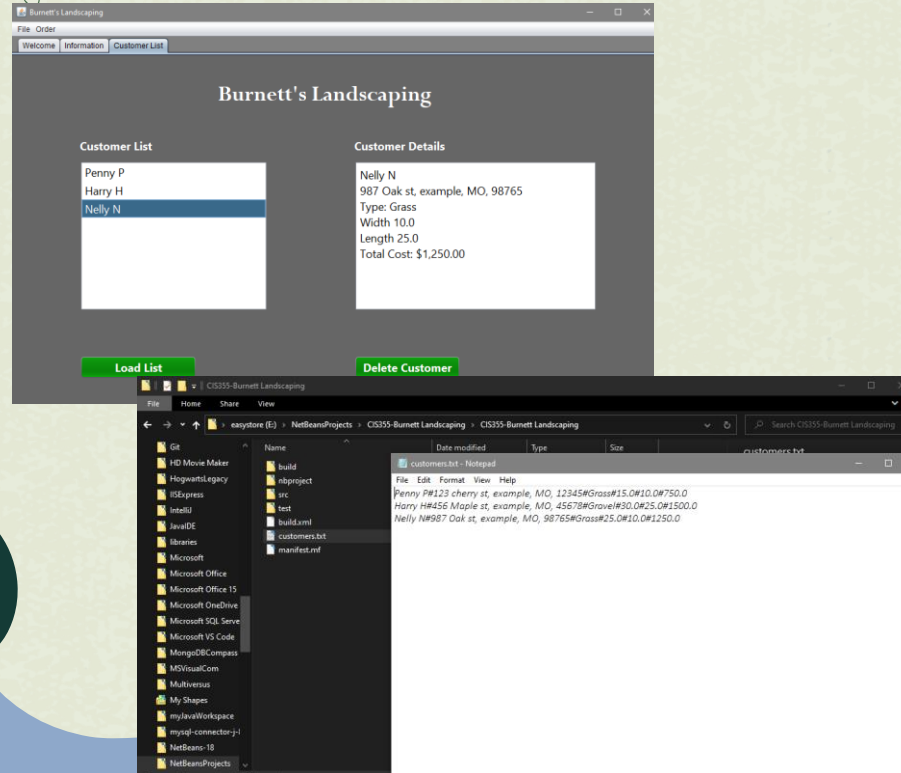




File Storage

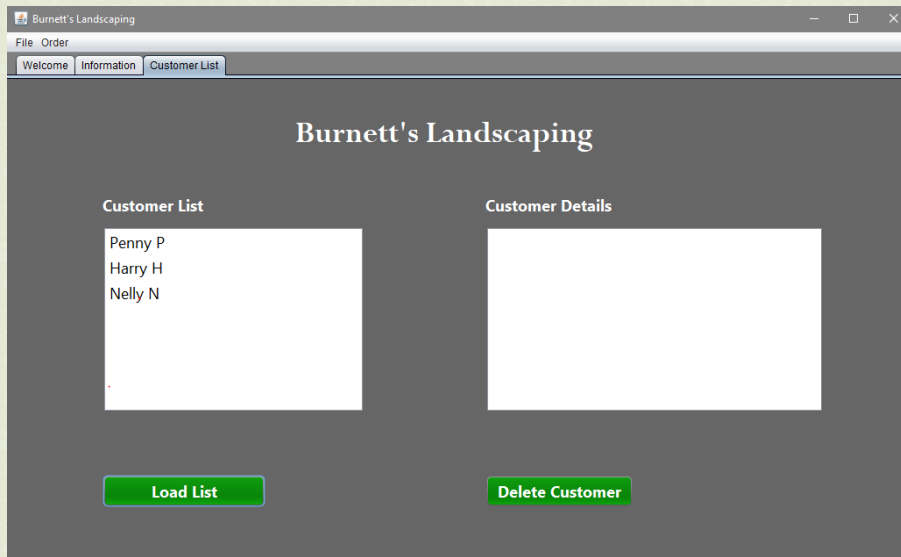
- Using the UML diagram create the DataIO class it will be used to store details of each order
- 

Saving to File



The order is submitted from the information page then displayed on the customer page. At the same time the data is being saved to a local text file.

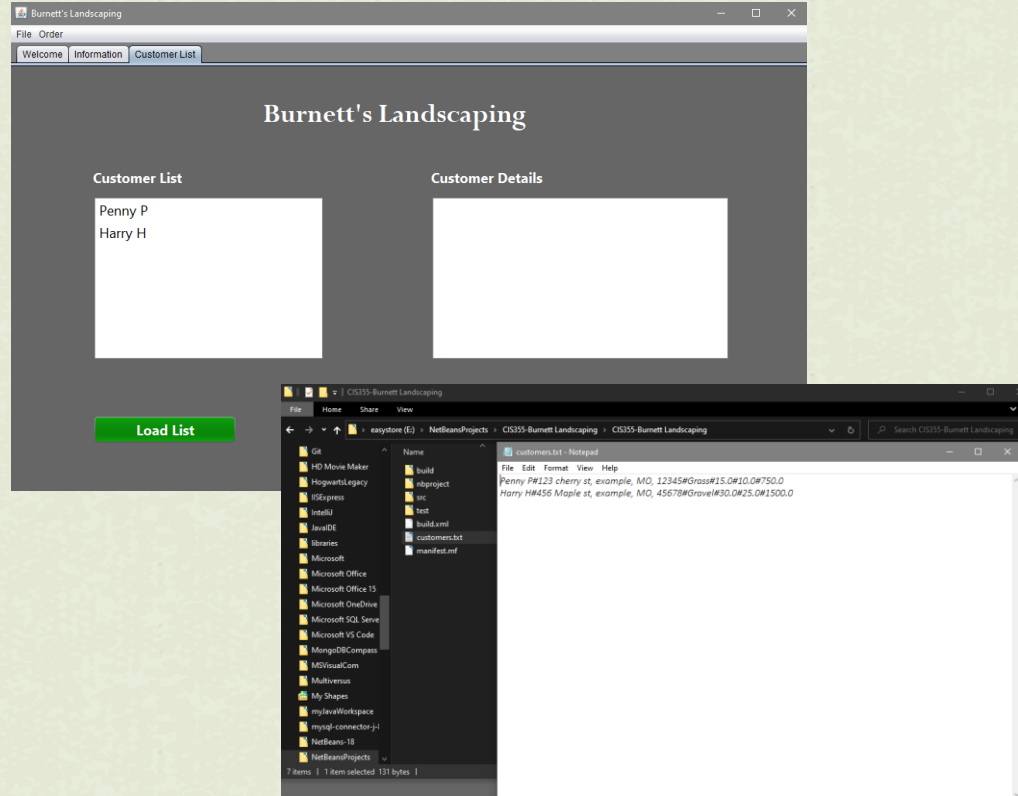
Load Customer



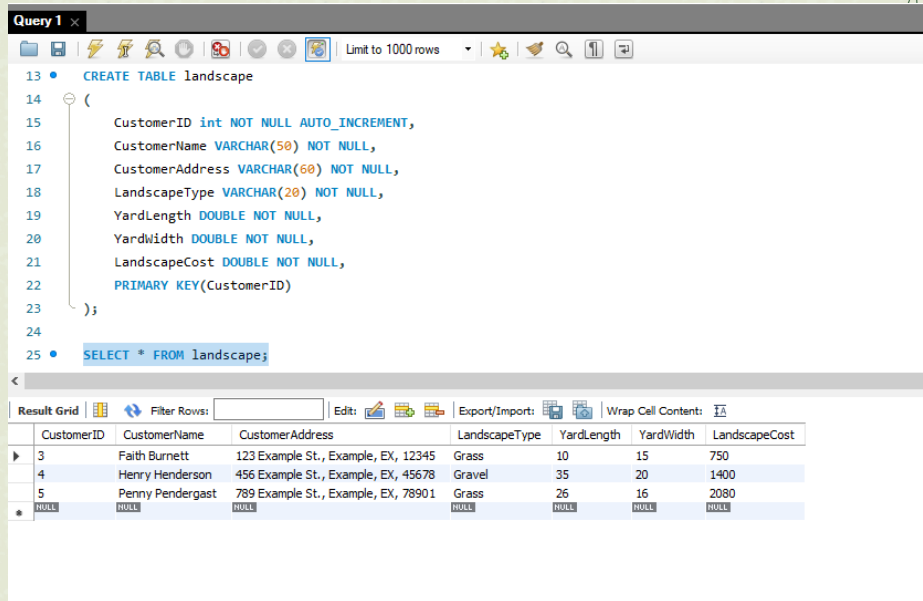
If you open the application and go straight to the customer page you can hit load list and the customer data is retrieved from the text file and displayed.

Delete Customer

Now that the customers are loaded if you select one and hit delete the customer is deleted from the list and the text file.



Switch Storage Method



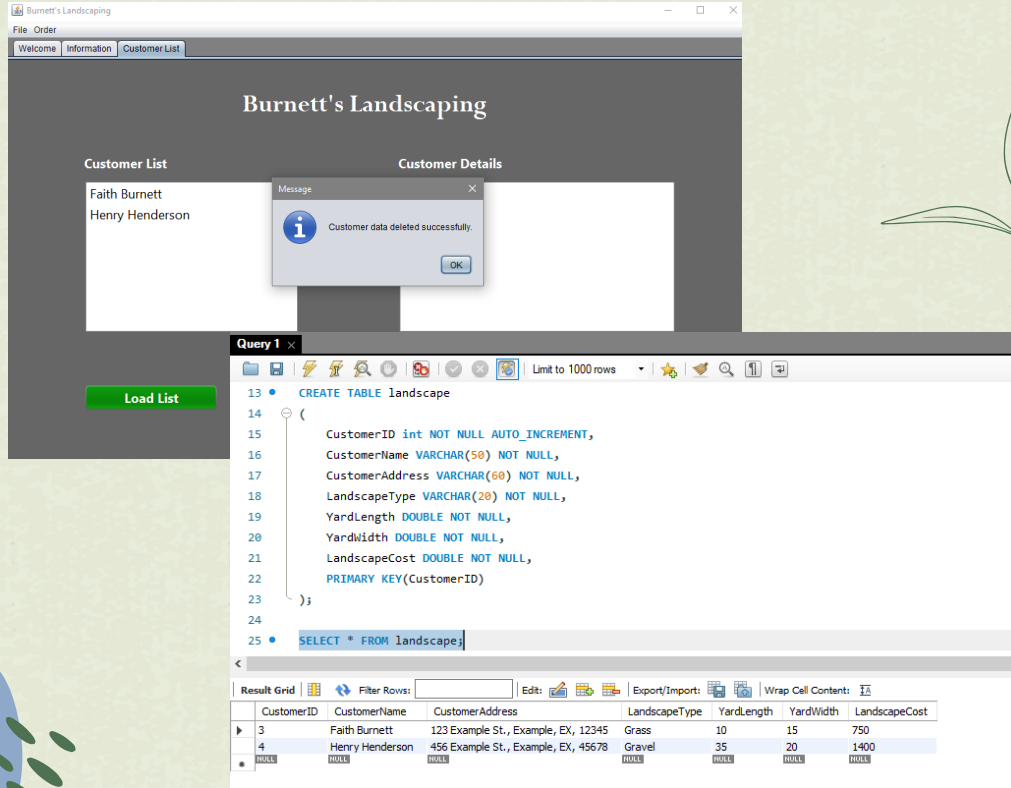
```
Query 1
Limit to 1000 rows

13 • CREATE TABLE landscape
14 (
15     CustomerID int NOT NULL AUTO_INCREMENT,
16     CustomerName VARCHAR(50) NOT NULL,
17     CustomerAddress VARCHAR(60) NOT NULL,
18     LandscapeType VARCHAR(20) NOT NULL,
19     YardLength DOUBLE NOT NULL,
20     YardWidth DOUBLE NOT NULL,
21     LandscapeCost DOUBLE NOT NULL,
22     PRIMARY KEY(CustomerID)
23 );
24
25 • SELECT * FROM landscape;
```

CustomerID	CustomerName	CustomerAddress	LandscapeType	YardLength	YardWidth	LandscapeCost
3	Faith Burnett	123 Example St., Example, EX, 12345	Grass	10	15	750
4	Henry Henderson	456 Example St., Example, EX, 45678	Gravel	35	20	1400
5	Penny Pendergast	789 Example St., Example, EX, 78901	Grass	26	16	2080
NULL	NULL	NULL	NULL	NULL	NULL	NULL

In this part we have created a connection to a MySQL database and instead of saving the customer object to a text file we'll be saving it to the landscape table inside of the database.

Loading and Deleting



The screenshot displays the Burnett's Landscaping application interface. The main window has tabs for 'Welcome', 'Information', and 'Customer List'. The 'Customer List' tab is active, showing a list of customers: Faith Burnett and Henry Henderson. A 'Load List' button is visible at the bottom left. A 'Message' dialog box is open in the center, displaying the text 'Customer data deleted successfully.' with an 'OK' button. Below the application window, a 'Query 1' window is open, showing a SQL query and its results.

Query 1

```
13 CREATE TABLE landscape
14 (
15     CustomerID int NOT NULL AUTO_INCREMENT,
16     CustomerName VARCHAR(50) NOT NULL,
17     CustomerAddress VARCHAR(50) NOT NULL,
18     LandscapeType VARCHAR(20) NOT NULL,
19     YardLength DOUBLE NOT NULL,
20     YardWidth DOUBLE NOT NULL,
21     LandscapeCost DOUBLE NOT NULL,
22     PRIMARY KEY(CustomerID)
23 );
24
25 SELECT * FROM landscape;
```


Result Grid

CustomerID	CustomerName	CustomerAddress	LandscapeType	YardLength	YardWidth	LandscapeCost
3	Faith Burnett	123 Example St., Example, EX, 12345	Grass	10	15	750
4	Henry Henderson	456 Example St., Example, EX, 45678	Gravel	35	20	1400
NULL	NULL	NULL	NULL	NULL	NULL	NULL

As before with the text file we can load customers and delete them and changes are made in both the list and the database.




Challenges

- Making sure components were named to uniquely so that I could remain organized
 - Remembering to add comments so that someone else could know what each block does
- 



Skills



- Using OOP principles
 - Designing wireframe
 - Creating UML diagrams
 - Debugging
 - Building a GUI
- 



Thanks!

Do you have any questions?

faithburnett@outlook.com

CREDITS: This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)

