**INTRODUCTION**

CSV(Comma-Separated Values) file is a type of text file that stores tabular data in plain text format. Each line of the file represents a row of the table, with each field separated by a comma.

The CSV file reader is a Java application that reads the contents of a CSV file and displays it. It has a friendly user interface which was built using Java Swing GUI and it allows users to either select or drag and drop the CSV file to the user interface. The user must have an Integrated Development Environment and at least a JDK 19 to run the code successfully.

The CSV file reader features are:

1. Selecting of CSV file
2. Drag and Drop CSV file feature
3. Displaying of CSV file contents in Table format

APPROACH

The CSV file reader was built using Java and Java Swing GUI and it is made up of two JFrames. The first frame which will be displayed after the code has run successfully, displays the project name and a button for the user to begin. The second frame shows options for the user to either select a CSV file or drag and drop a CSV file. The first frame was written in the Main java file while the second frame was written in the ReaderUi java file.

An import of Swing and Awt was written in the Main java file in order to make use of Java Swing GUI methods. An Event listener was added to the button in the first frame to listen for clicks. When the user clicks on the button, “frame.dispose()” method is used to hide the frame and then a new Object from the ReaderUi class in the ReaderUi java file is created. A constructor was written in the ReaderUi class that displays the second frame using the “frame.setVisible()” method.

The ReaderUi java file contained imports of FileReader, File and BufferedReader class which was used to read the contents of a CSV file. The CSV file will be displayed on a JPanel in the second frame in a table format, JTable and TableModel was used to achieve this. The ReaderUi class implements the DropTargetListener which enables it to receive drag and drop events. After selecting a CSV file, the file path is passed as an argument to a method that reads the CSV file when the “read button” has been clicked. The challenging part of the project was getting individual values in the CSV file. The ‘.split(“\n”)’ method was used to separate the rows into individual values in a String array and then the ‘.split(“,”)’ method was used to split the individual items to a new String array and to remove the commas before adding them to the Table.

CONCLUSION

The CSV file reader provides a friendly user interface for the user to easily read the contents of a selected CSV file. It also handles errors in a case where the file the user selected is not a CSV file.