CS 3330 FINAL PROJECT DOCUMENTATION

UI Development: This project was created using JavaFX. It generates parings for participants in a Secret Santa gift exchange. It then allows participants to start shopping for their based on the preliminary data collected on Amazon. A hyperlink is updated to contain a link to an amazon search page with the gift preference listed in the data file as a query. You can then save your pairings and your match's gift preference to a file.

Architecture:
Model { SecretSantaModel.java }
Views { FXMLDocument.fxml}
Controllers { FXMLPlayerController.java}

Required Elements

1. Object Oriented Elements :
    a. Classes
    SecretSantaModel.java, FileHandling.java, CSVHandling.java, Pairs.java, Am9fqSecretSanta.java.
    b. Subclasses
    CSVHandling is a subclass of Pairs. Seen on Line 26 of CSVHandling.java. SecretSantaModel.java is a subclass of CSVHandling.
    c. At least one Abstract Class
    Pairs.java is an abstract class
    d. At least one Interface
    FileHandling.java is an interface and it is implemented by CSVHandling.java.
2. Code elements that you utilize:
    a. One or more collection classes
    List and HashMap are used throughout the application. For example, List<String> in the CSVHandling.java function convertToList() (Beginning on line 28 in CSVHandling.java). A HashMap is used in the same file in the method initKey() (Line 56).
    b. Exception Handling
    There are multiple cases of using try-catch statements for exception handling. One of them being in the saveFile() method in CSVHandling.java beginning on line 137.
    FXMLDocumentController.java uses a throws statement in the method openLink() on line 127.
3. The application must have a clearly defined model
    The model is contained in the SecretSantaModel.java file.

4. The UI must utilize multiple scenes and at least one of the scenes will have the contents of the scene graph changed based on the application state

   The main scene for this application is the "Get Started" scene. It contains multiple scenes including info and error scenes that are activated based on the click of the button. For example, entering a name in the shopping search bar that was not submitted in the original data file will show a new error scene explaining that there was an error. This is shown in the method searchError() in FXMLDocumentController.java beginning on line 97. Another example is when a file is successfully saved after pressing the save button, an info scene alerting the user that the file was successfully created and stored is shown (saveSuccess() in FXMLDocumentController.java starting on line 131).

   An example of the scene graph changing based on the application state is the File upload status text changing to "File Uploaded" once the user uploads a CSV file. Line 77 in FXMLDocumentController.java.

5. There must be a way to access "About" information that includes information about you and the application
   There is a tab labeled "About Me" that contains information about the application and myself.
6. The application must save and load data. The target for saving/loading data can be files, a network service, or a database

   This application loads data from a CSV file that the user selects to upload. The one created for testing is called participants.csv (uploadCSV() in FXMLDocumentController.java beginning line 70). It is in the form "Name, Gift". The user has the choice to save the pairings generated to a file called "savedpairs.csv" which is automatically saved when the save button is pressed (CSVHandling.java, saveFile() method beginning line 137).