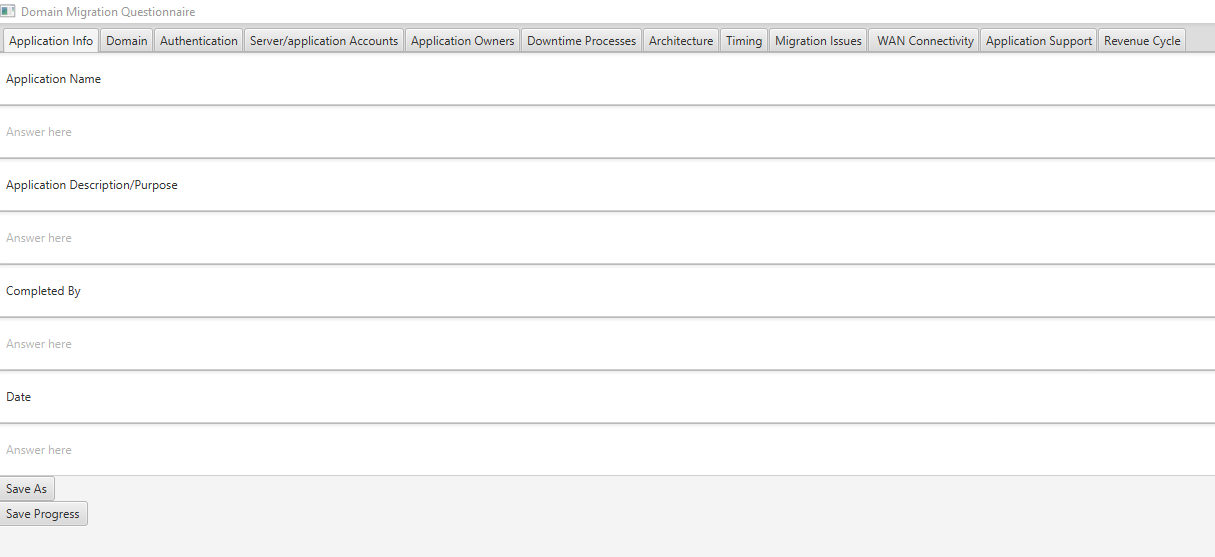
**Domain Migration Questionnaire Form Java GUI Documentation**

Developed by Chaitu Konjeti

Email: ckonjeti1@gmail.com



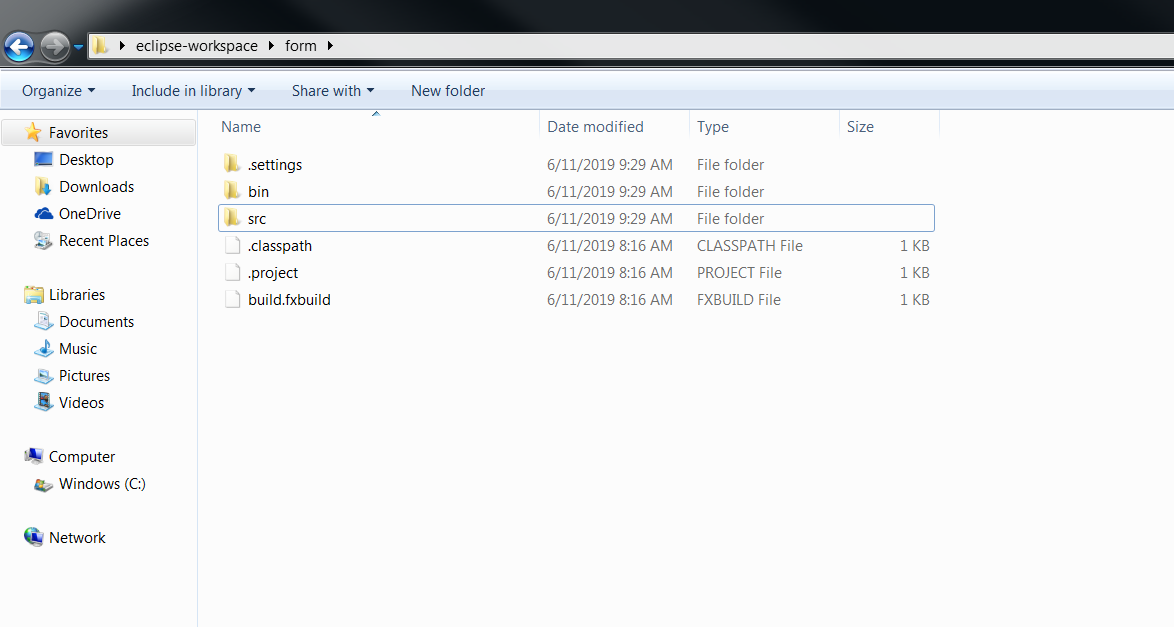
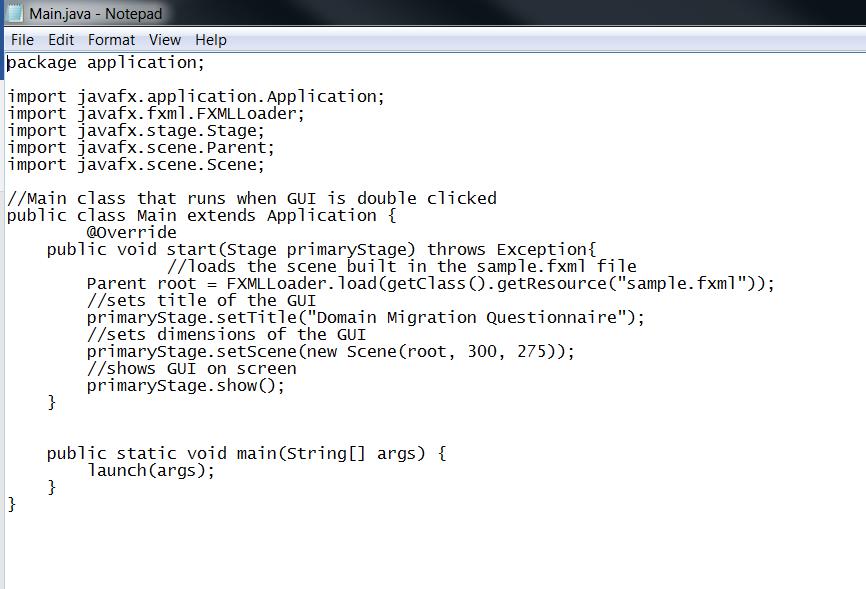
Transforms the Domain Migration Questionnaire excel form into an easy-to-use GUI using Java

Table of Contents

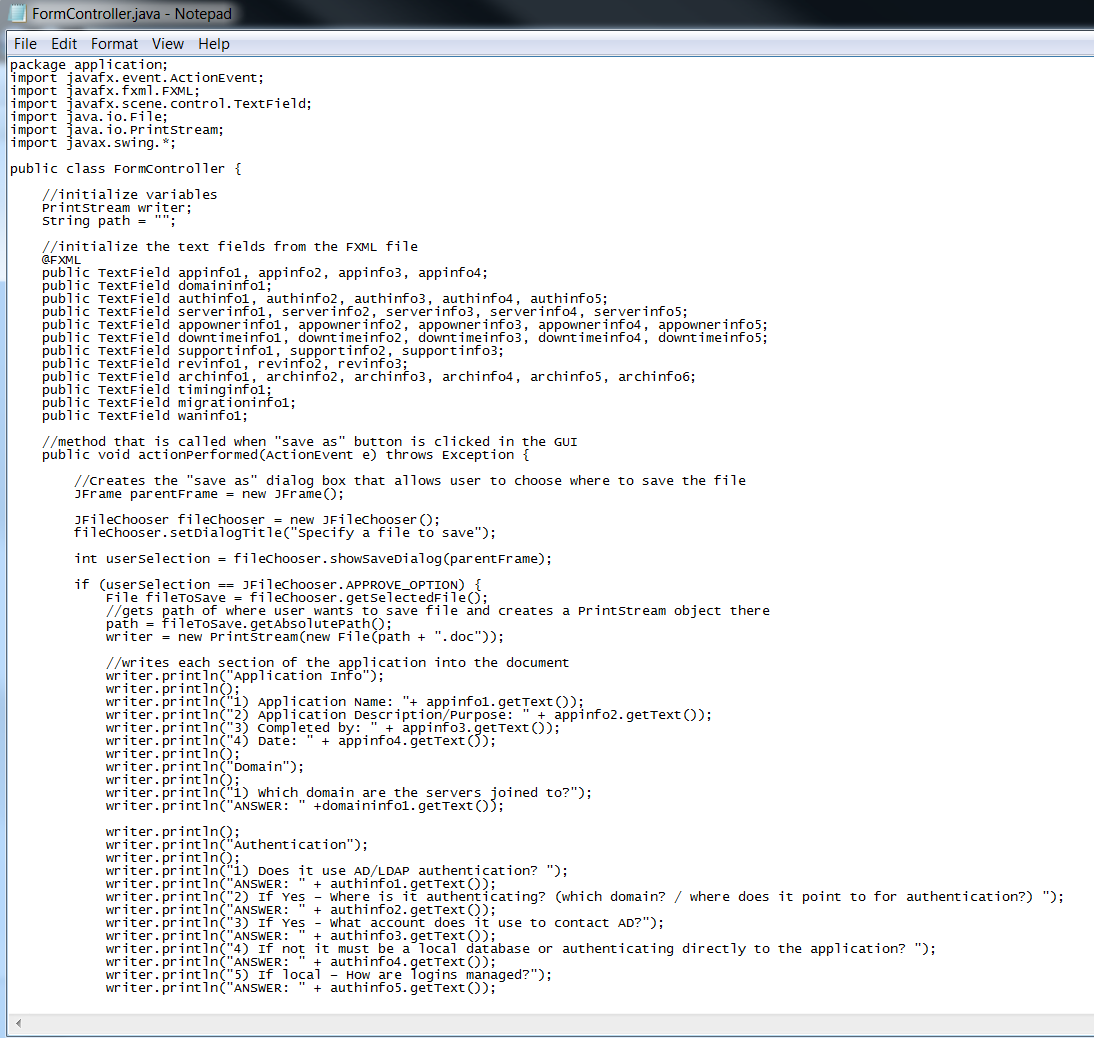
1. Project Directory
   1. Organization
   2. Main GUI Application Files
      1. Main.java
      2. FormController.java
      3. sample.fxml
      4. MAINFEST.mf
2. Exporting JavaFX Project as JAR File Using Eclipse IDE
3. Creating Batch File for Running JAR File
4. Using Launch4J to Bundle JRE with JAR File
5. Running the GUI and Understanding the Output
6. Appendix of Tools Used

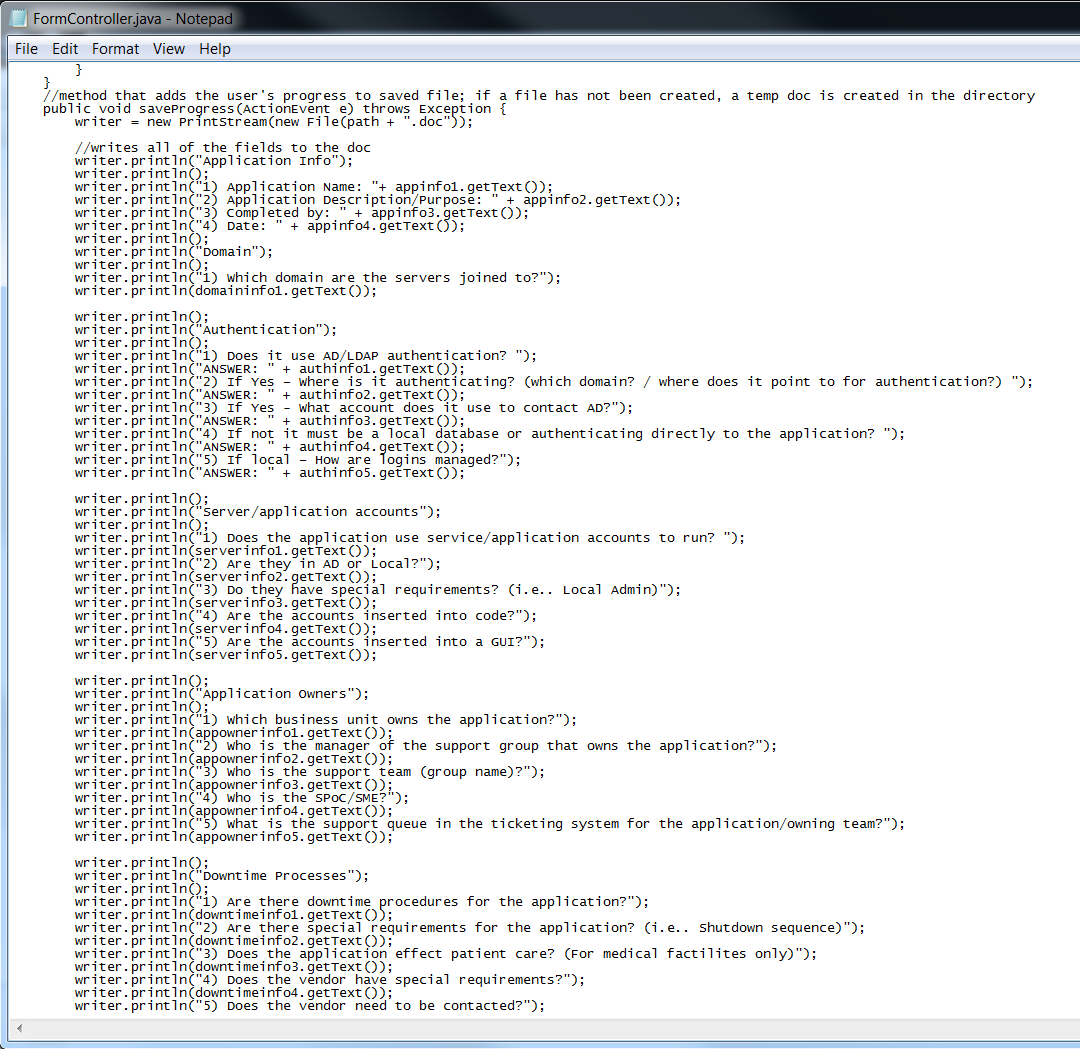
**Project Directory**

1. **Organization**

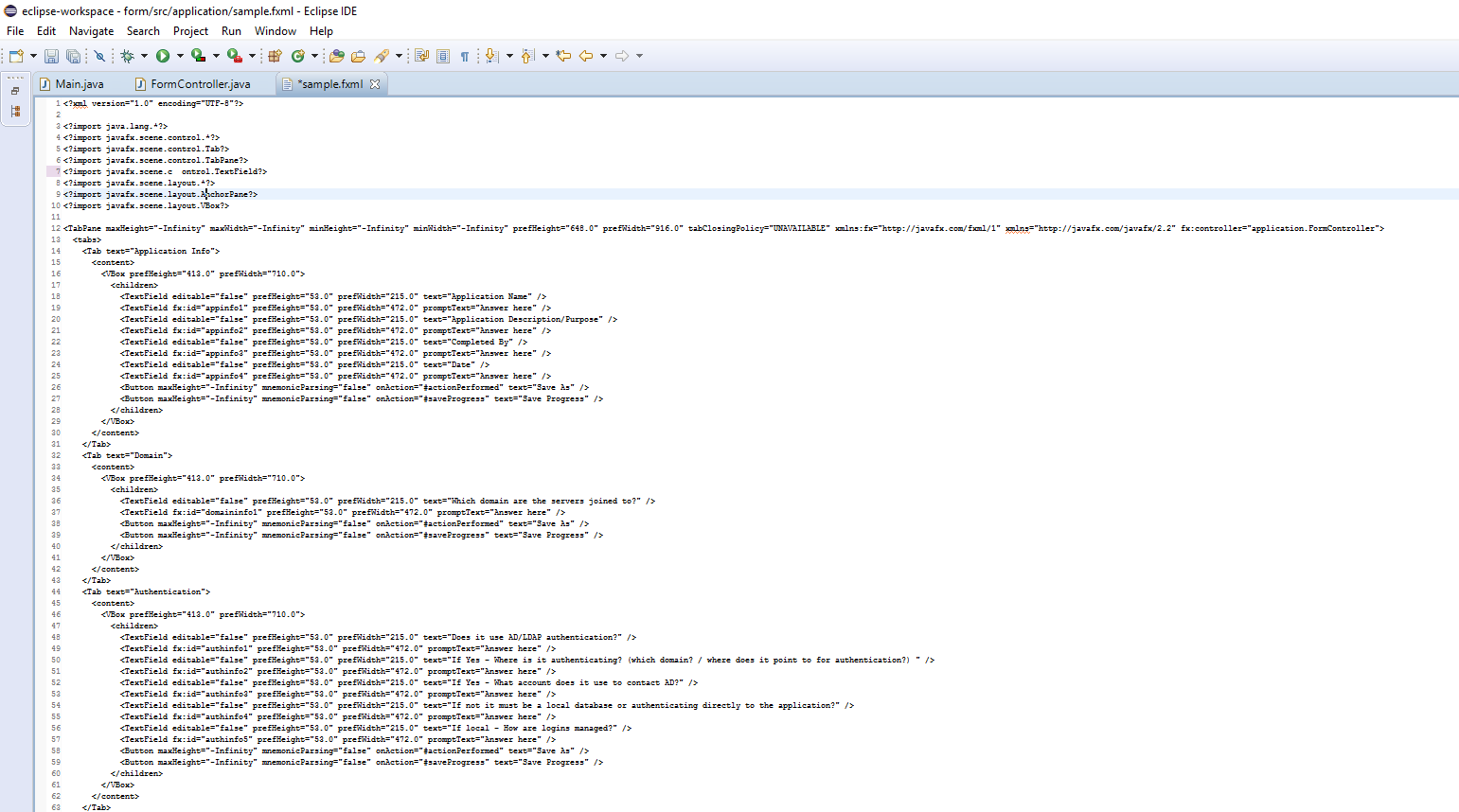
* 
* The src (source) folder contains all the Java, FXML, and Manifest files that were developed
  + The Java files control the functionality of the Text Fields and Buttons of the GUI created in the FXML file. The FXML file is an XML file that contains the layout of the GUI. The manifest file contains the path location of the Main java class in the project.
* The bin folder contains the .class files that were generated when the .java files in the src folder were compiled. These files should not be edited.
  1. **Main GUI Application Files**
     1. **Main.java**
* 
* This file is the first file to start up when the GUI is running. The start method initializes the GUI information – the FXML file, the title, and the dimensions. Then, it shows the GUI on the screen. The main method launches the start method.
* This file is in src/application

* + 1. **FormController.java**

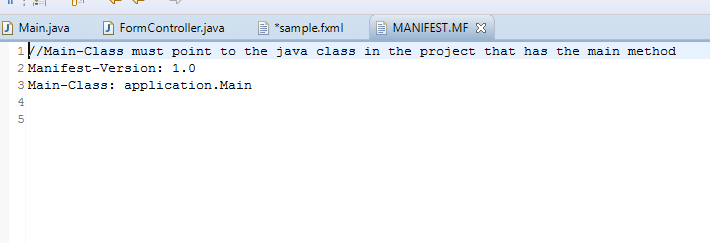




* The FormController file has two methods: actionPerformed() and saveProgress(). The actionPerformed method is called when the “Save As” button is clicked in the GUI. The link between the button and the method is established in the FXML file. First, a file chooser dialog box is created so that the user can choose where to save the file created from the GUI. Then, the method takes all of the user input from each of Text Fields in the GUI and writes them to a .doc file. The saveProgress method is called when the “Save Progress” button is clicked in the GUI. The link between the button and the method is established in the FXML file. When this button is clicked, a temp .doc file is created with the User’s answers so far.
* This file is in src/application
  + 1. **Sample.fxml**

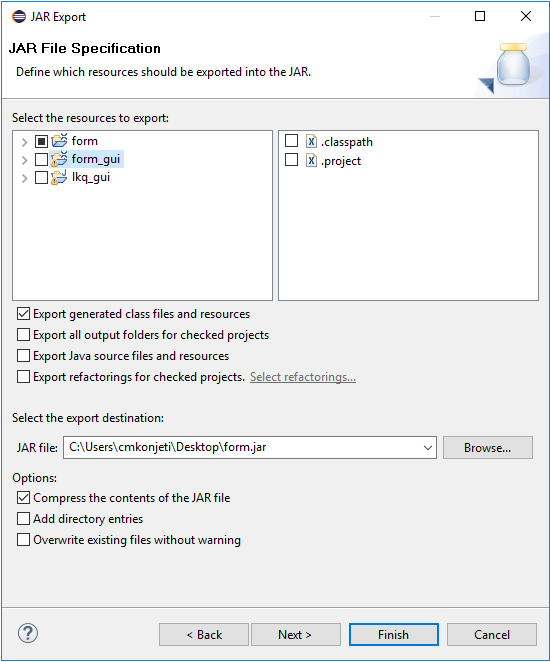
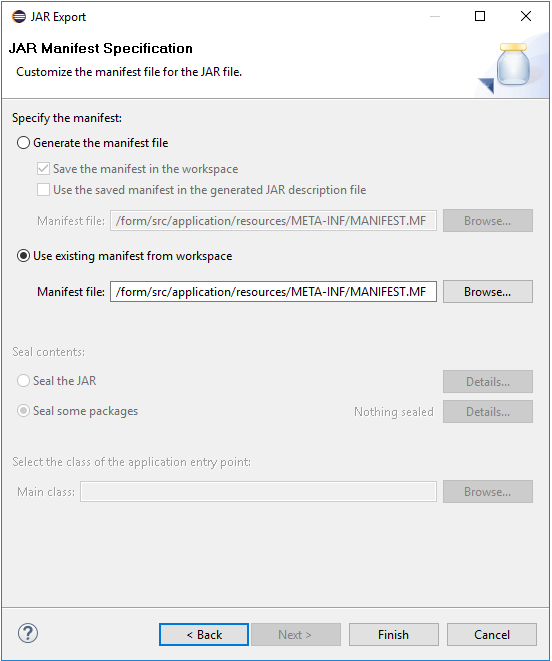


* The sample.fxml file is where the layout of the GUI is designed. This file can be opened up in SceneBuilder, where you can drag-and-drop the various items rather than code the XML file by hand. In the TabPane aspect, one of the parameters is “fx:controller: application.FormController.” This connects the GUI to the methods in the FormController Java class. Under the TabPane, several Tab items are created for the various categories of questions in the questionnaire. Under the Tabs, there is a VBox (Vertical Box) that holds the TextFields that display the questions, the TextFields where the user inputs his answer, and the 2 buttons. For each of the Buttons, there is a parameter that says “onAction.” This parameter connects the button to the specified method in the FormController.
* This file is in src/application
  + 1. **MANIFEST.MF**

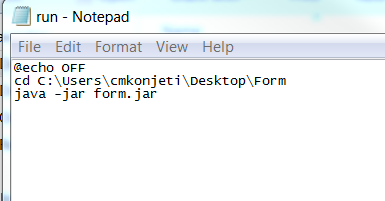


* The MANIFEST file simply allows the program to know which class is the Main-Class. When defining the location of the Main-Class, you must include the package name as well (application).
* This file is in src\application\resources\META-INF

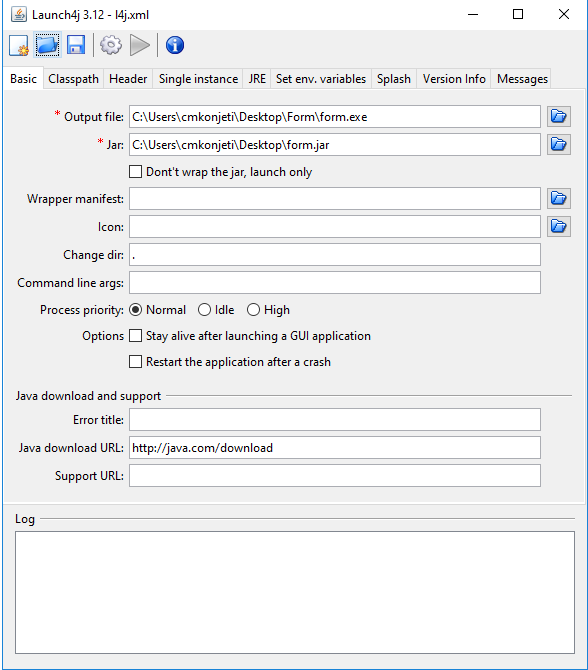
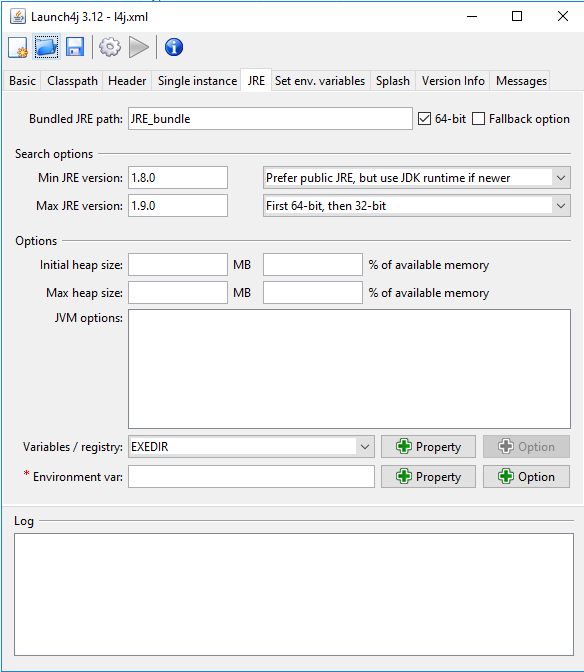
1. **Exporting JavaFX Project as JAR File Using Eclipse IDE**

* To export the JavaFX GUI as a JAR file that can be run on any other computer:
  + Open Eclipse
  + Click File -> Export
  + Click the Java tab and then click JAR file
  + 
  + At this point, the project that you want to export into a JAR file should be selected. You can also select the destination where you want the JAR file to be exported. Then, click next until you reach the JAR Manifest Specification section.
  + 
  + At this stage, you will need to select “Use existing manifest from workspace.” Then, browse the directory to find the location of your MANIFEST.MF file. Then click Finish.
  + The JAR file should have been created in the specified location.

1. **Creating Batch File for Running JAR File**

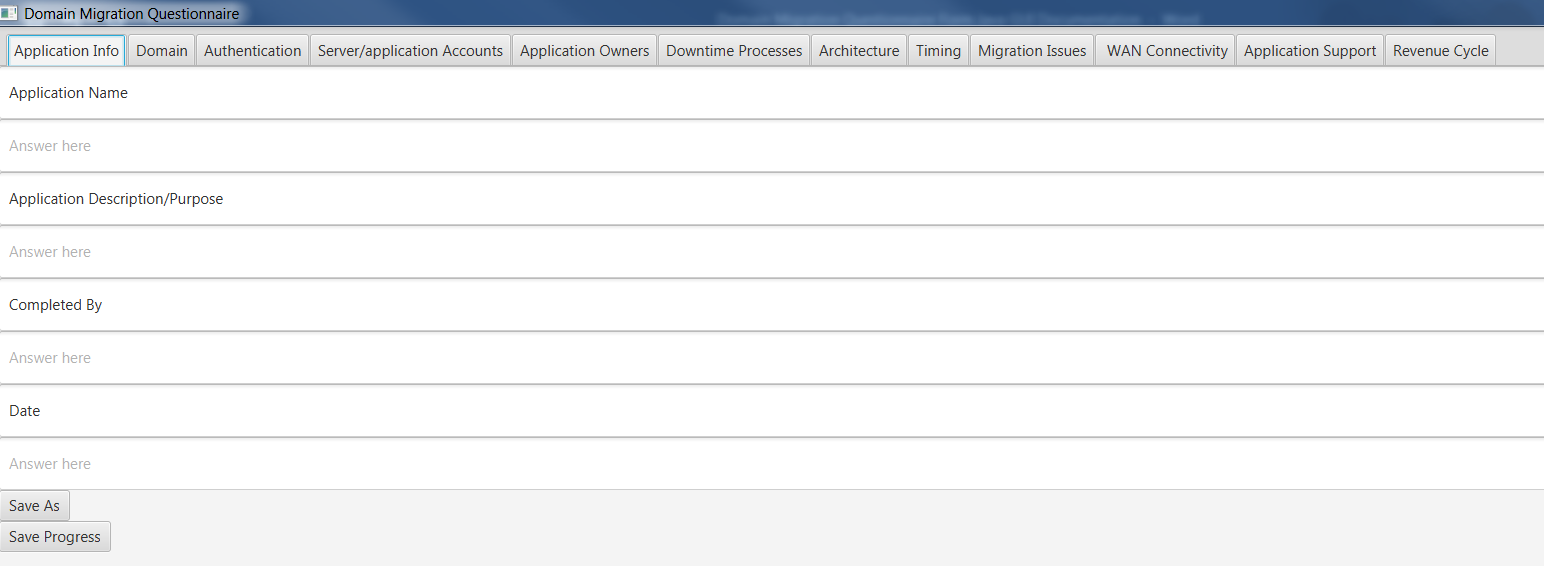
* Although the JAR file has been exported, you cannot run it by double clicking the file because a batch file to run the file has not been created. If you want to run the JAR file without the batch file, you will need to open Command Prompt and type java -jar form.jar.
* To create the batch file, go into the folder with the JAR file and create a new text document. Inside the text document, type the following:
* 
* Make sure to replace the location of the folder to where your JAR file is located
* When you save the file, make sure the extension of the file is “All Files” and save the document as “run.bat”
* Now, when you double click the JAR file, it should automatically run the GUI

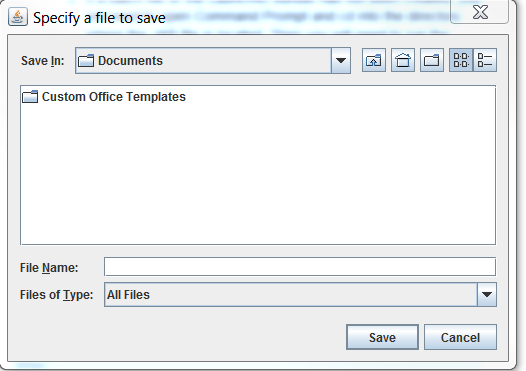
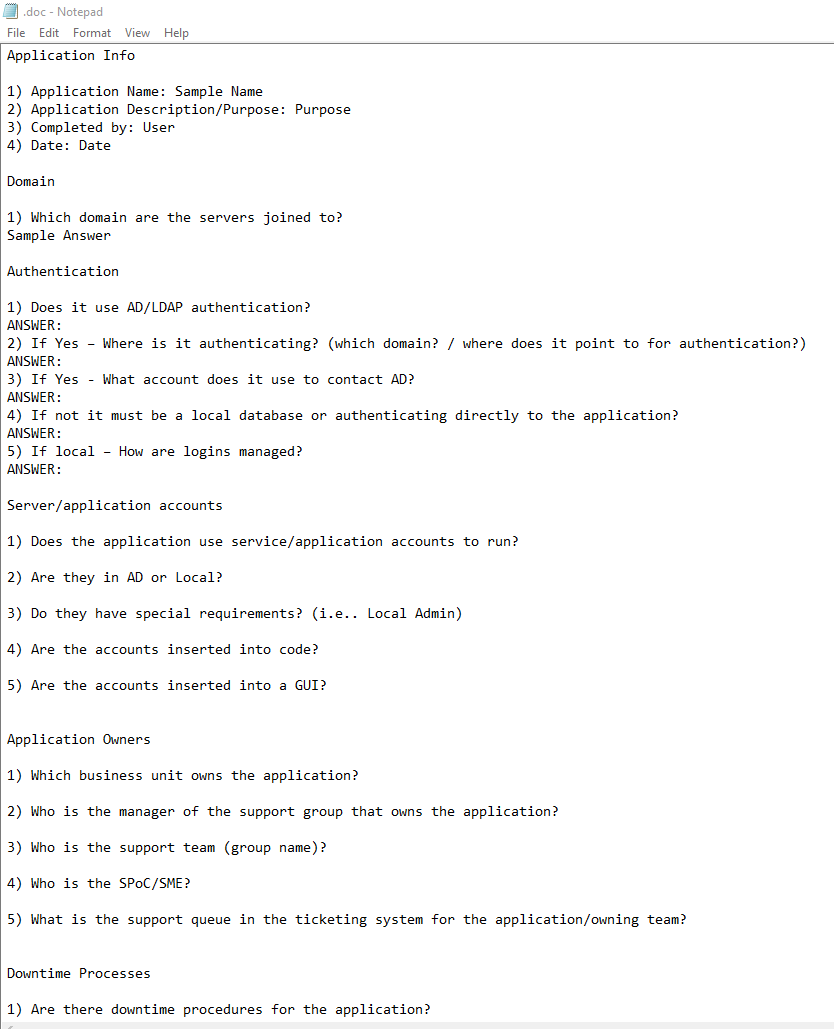
1. **Using Launch4J to Bundle the JRE with JAR File**

* If the machine that you want to deploy the GUI onto does not have Java installed, then you will need to bundle the Java Runtime Environment (JRE) with the JAR file using Launch4J
* Install the Launch4J software from the Internet
* Then, create a new folder in the folder where your JAR file is located and name it
* Then, go into C:\Program Files\Java and copy the bin and lib folders from the JRE that you are using for the GUI and paste into the JRE folder you created in the previous step.
* Then open Launch4J
* Fill in the opening page as follows:
* 
* Then click the JRE tab and fill it in as follows:
* 
* Then, click the gear icon in the top
* Now, a .exe file should have been created for the GUI
* This .exe file can be run on any computer because the JRE is bundled with it. Having Java on the machine is no longer a dependency for the GUI.

1. **Running the GUI and Understanding the Output**

* To run the GUI:
  + The Java Runtime Environment (JRE) that was used in this was 1.8.0
  + If a batch file or the Launch4J bundle has not been created, you will need to open Command Prompt and cd into the directory where the JAR file is located. Then you will need to run the command “java -jar form.jar.” You will need to have Java installed on the machine as well.
  + If you have the batch file but no Launch4J bundle, then you can double click the JAR file. Java will need to be on the machine for it work.
  + If you have the Launch4J Bundle, then you can double click the .exe file and the GUI should run.
* When the JAR file is opened, the following GUI should open:



* The user can fill in each of the text fields with the answers to the questions.
* At any point, the user can click the “save as” button. When this button is clicked, a file chooser dialog box is opened so that the user can choose where to save their output file.
* When the “save progress” button is clicked:
  + If the user has already clicked the “save as” button, it will update that file with their new answers
  + If the user has not clicked the “save as” button, then it creates a nameless file with the answers in the directory containing the JAR file
* The output file that the GUI creates looks like:
* 
* The questions along with the answers that the user writes are formatted into a .doc file.

1. **Appendix of Tools Used**

* The IDE used was Eclipse
  + Can be downloaded at <https://www.eclipse.org/downloads/>
* To bundle the JRE with GUI JAR file, Launch4J was used
  + Can be downloaded at <http://launch4j.sourceforge.net/>