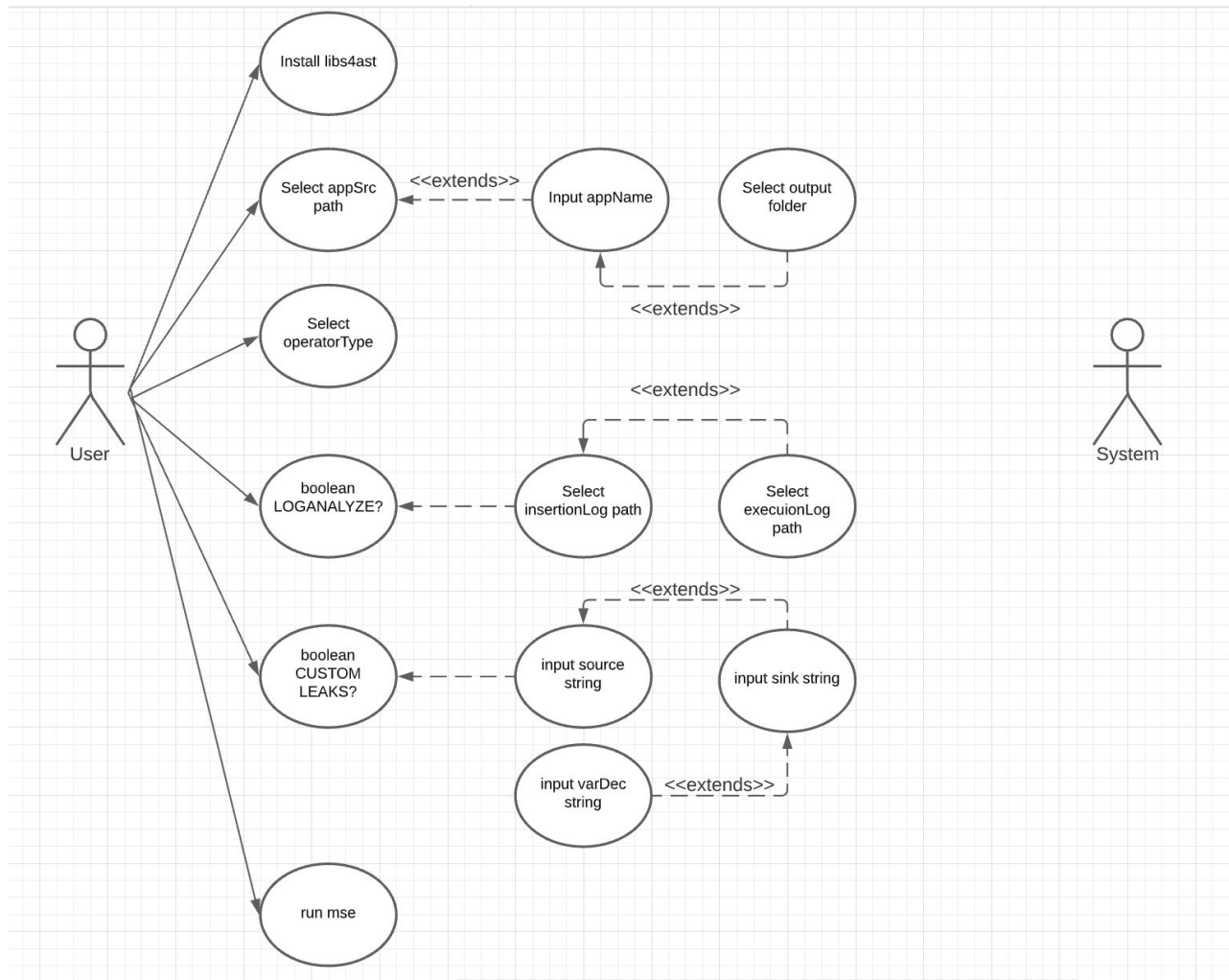


MSE GUI Brainstorm

Team Members: Phil & Johnny



Initial

- Sketches created
- **Question:** Will this GUI run within Eclipse only or outside?
- Frames created
 - Main Frame (Title screen; choose whether to make a new config or use existing one)
 - Create New Configuration Frame (Asks user to input information for the config file)
 - Loading/execution tracking Frame (progress bar, updates as muse runs)
 - Crash report Frame (tells user muse crashed--gives info about why)
 - Successful completion Frame (tells where mutated code is, how long it took to run, which operator was used, and other information about the execution that's useful).

Summary of potential next steps

1. Automate the installation of mdroid (bash script?)
2. Brainstorm GUI features
3. Add a .java file for validating the configuration (check if file paths, operators, etc are valid)
4. Design and implement the GUI code (interfaces, classes, layouts)
5. Examine what JavaFX has to offer (layouts, buttons, pop-up windows, user input fields, drop down menu)
6. Choose color scheme
7. Make issues for all the things we plan to do, and decide who is doing what

New feature with GUI

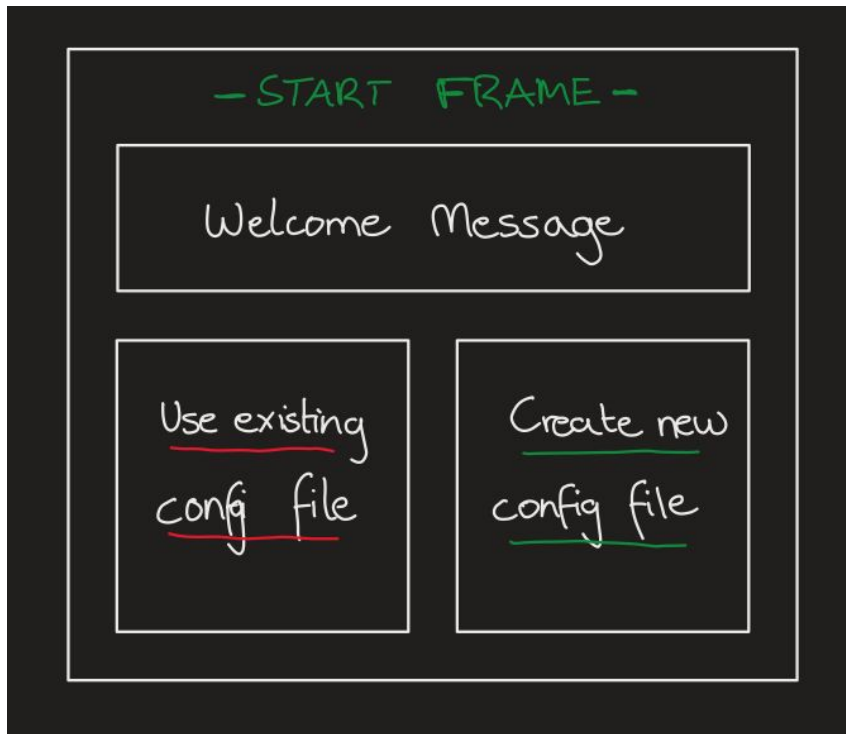
1. Configuration validation (green check if input path and format are valid)

Bill To / Billing Address

Full Name	John Newman	✓
Street Address	2125 Chestnut st	✓
	<i>optional</i>	
Zip Code	9412	Enter Zip for City & State The specified ZIP is invalid
Phone		
Email		

Send me exclusive offers, deals and expert reviews ☒

Start Frame



- AKA First Activity
- Import config.properties or create a new properties file
- Able to navigate to **either**:
 - Main Frame
 - Loading Frame

Main Frame

- To set up “config.properties”
- Able to navigate to Loading Frame

A hand-drawn sketch of a software interface titled "MAIN FRAME". The interface is divided into several sections:

- CHOOSE APPLICATION LOCATION:** A section with a long horizontal button containing a right-pointing triangle.
- APPNAME:** A label followed by a text input field and a button labeled "SELECT OUTPUT FOLDER" with a right-pointing triangle.
- SELECT OPERATOR TYPE:** A section with a long horizontal button containing a right-pointing triangle.
- LOGANALYSE?:** A checkbox with a green checkmark and a red 'x'.
- CUSTOMLEAKS?:** A checkbox with a green checkmark and a red 'x'.
- insertionLog path:** A label followed by a text input field.
- executionLog path:** A label followed by a text input field.
- source_string:** A label followed by a text input field.
- sink_string:** A label followed by a text input field.
- varDec_string:** A label followed by a text input field.
- RUN:** A large red button at the bottom left.

Loading Frame



- Gives application progression
- Able to navigate:
 - Back to previous frame
 - Error Frame (if called)
 - Finish Frame

Error Frame

- Gives specifics on error
- Some error has occurred
 - Exit application
 - Return to Start (reconfigure)



Finish Frame



- “Completion Activity”
- Gives result summary
- Able to navigate:
 - Back to Start Frame
 - Exit application

New files (and probably a GUI folder where src and test are)

GuiMain.java (extends Application)

See:

<https://www.opensourceforu.com/2017/07/developing-basic-gui-application-using-javafx-eclipse/>

ConfigBuilder.java (separate ConfigValidator class?)