

Mobile applications (e.g., Android) are heavily driven by their graphical user interfaces (GUIs), as this is primary means by which users interact with apps. However, building these GUIs can be difficult as they are typically constructed according to a complex hierarchical tree-based structure, where leaf-level components in this tree represent objects with which a user would typically interact. Your goal for this programming assignment is to build a small tool that highlights these leaf level GUI-components in an Android application screenshot by parsing and processing metadata (in the form of xml files) that describes the hierarchical screen structure, thus making it easier for developers to understand the GUI. The input to your tool will be the set of screenshot/xml file pairs attached to this assignment, and the output should be a new set of annotated screenshots (see examples below) where leaf-level components are highlighted with a yellow box. The file names delineate corresponding screenshots and xml files, where file names are in the format: <app.package>-<screen#>.ext, where the extension is either .xml or .png; note that only some apps have multiple screens. The xml files were captured from android using the dump feature of the uiautomator framework in Android, therefore, you may want to research this representation before starting your implementation.

When you are finished with the assignment, please send it to [svelascodimate@wm.edu](mailto:svelascodimate@wm.edu). Ideally, please have your project in a GitHub repository and just send us the link to it. However, if you are not able to do that, please send us a zip file. In your e-mail please indicate a SUBJECT line: CSci 435 Programming assignment #1. In the body of the e-mail please indicate your full name and W&M ID.

Some notes:

1. Your repository should have a README file that stipulates how to compile and run your code.
2. Your repository should have the set of annotated screenshots that are output from your tool.
3. Your repository should have your code and any libraries used (unless you are using a dependency management system like maven).
4. Please note that you can use any libraries as well as any programming languages to implement it.
5. A brief natural language description of your solution and why you made certain design decisions.

