Scratch Data Tools Extension

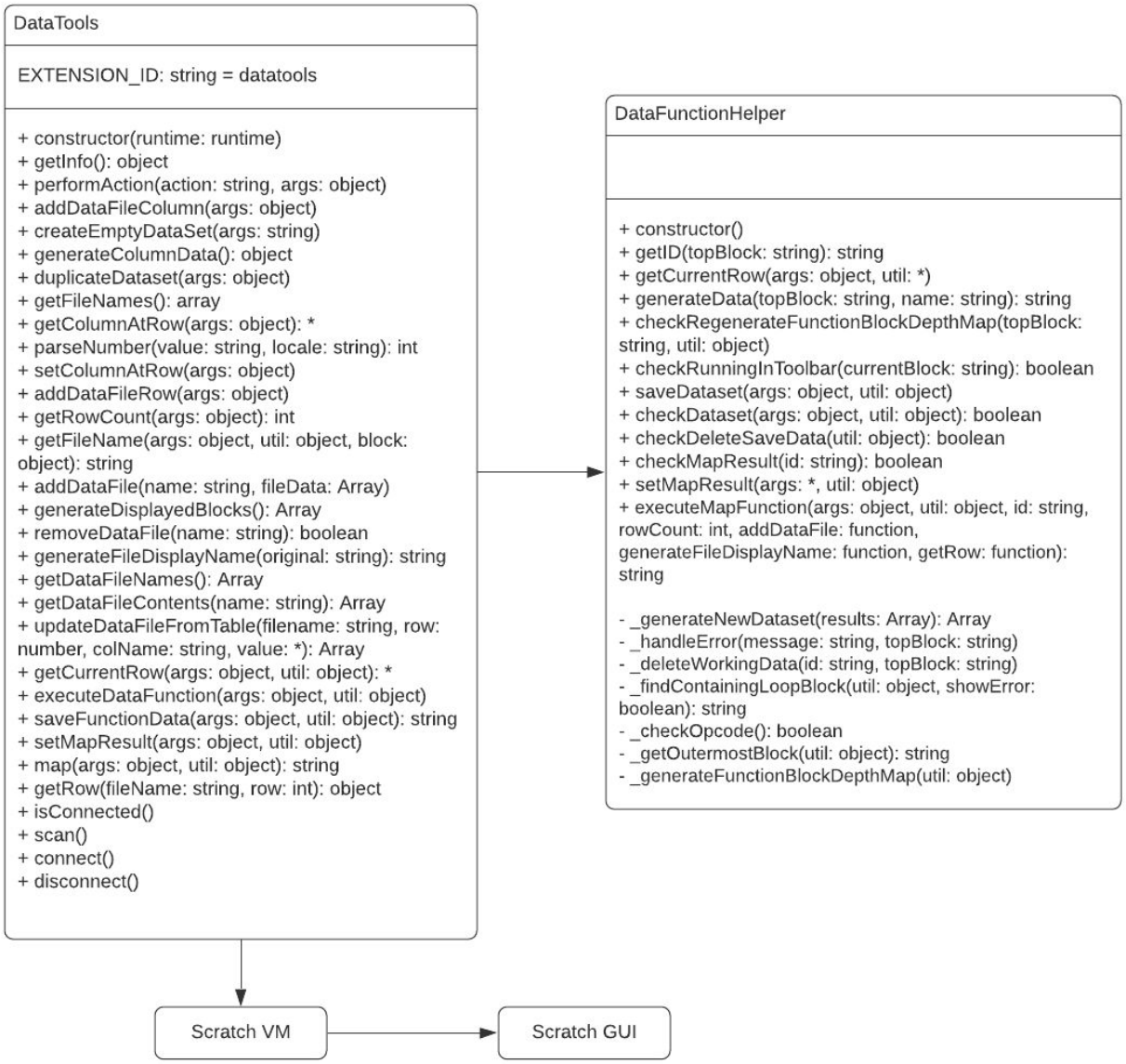
Developer Documentation

Program Structure

Scratch as a project is a combination of three repositories: the Virtual Machine, GUI, and Blocks. The majority of the code we write is in one dedicated extension folder in the virtual machine repository, currently split between two files - a main extension file responsible for the creation and functionality of most blocks in the extension, and a separate helper file containing the code responsible for some of the more advanced functionality. Some changes have been made to the GUI repository to add extra menus and help display the blocks correctly.

When an extension is selected by the user, an instance of that extension's index.js page is loaded into the extensions_manager which in turn talks to the runtime object for the vm. The GUI consumes an instance of the vm and uses this instance to make function calls to the backend as the user moves blocks and runs the program. We have used this concept to include functions in the runtime object that help our gui components, mainly the file selector menu that is loaded with the extension and the file viewer, communicate with the vm where the user's data is stored. This is also how files are uploaded, the extension is loaded by the user, a modal appears asking to upload a file, they select a file, the data is then processed by the gui to be represented in the proper format(generally speaking each file is an array of row objects, each row object is made of a series of column and value pairs, for example: [{name: 'Donald Duck', age: 15}, {name: 'Mickey Mouse', age: 55}]). This data is then sent to the extension to be stored using a call to the runtime.

Below is a brief overview of the project architecture, focused on the two extension files mentioned above in the VM repository.



Setting up a Dev Environment

Scratch is a node based project, to compile and run the code you must have node installed. To work on this project, you will need to host Scratch locally on your machine using the modified code of Scratch 3.0 hosted on github by the KSU Scratch Data Tools team. To begin, you will need to pull the **master** branch of these repositories:
https://github.com/papaphil/DataTools-scratch-vm
https://github.com/papaphil/DataTools-scratch-gui
https://github.com/papaphil/DataTools-scratch-blocks

Simply install scratch locally as you would scratch, but using these repositories. Instructions on how to install Scratch locally are maintained by the LLK developers and can be found here: https://github.com/LLK/scratch-gui/wiki/Getting-Started

We will be following existing guidelines for the development of Scratch extensions found here: https://github.com/LLK/scratch-vm/blob/develop/docs/extensions.md

In the future, the KSU Scratch Data Tools team hopes to host the modified version of Scratch publicly on the Kansas State University network. For the time being if you wish to use Scratch without the data tools extension you must go to https://scratch.mit.edu/.

Deployment

As this is a locally hosted project for an unofficial extension, there is no deployment method used. All use of the extension is described in the “Getting Started” link here: https://github.com/LLK/scratch-gui/wiki/Getting-Started