# Debugger Project

System built for the analysis and evaluation of software systems to aid development.

**Stage 1: Partition Source Code into Readable, Executable Chunks**

Procedure

* User specified project path as input
* Module to scan path for all source files (file scanner object)
  + Generates a list of paths to each source file
  + Loads source files into source file object (considering making a file loader class to handle this part)
  + \*Maybe should change to directory scanner
  + Consideration of .class files (ignore)?
  + Consideration of .xml files (ignore)?
  + Consideration of Libraries/bin?
  + Focus on .java files?
  + What about interfaces? Abstract classes? Inheritance?
* For each source file:
  + Partitioner class reads a source file one line at a time
  + Reads lines until designated stopping points (brackets, keywords)
  + Partitions designated code blocks
  + Returns list of partition objects
  + Consideration: User input based results?
* Returned objects will contain all the information needed to run each

Partition Object:

* Represents a function (method in Java)
* Contains the following data members:
  + Source file name
  + Extracted header block from source file
  + Extracted class name and modifiers from source file
  + May include extended class name separately
  + Dictionary of lines (key will be source file line #)
* Must be able to be recombined with other partitions or run individually

Source File Object:

* Represents a whole, complete source file
* Contains the following data members:
  + Source file name
  + Source file language
  + Source file lines
  + List of partition objects (not sure about this yet)

Goal for Stage 1:

1. Successfully scan for all target source files
2. Generate partition objects
3. Recombine partition objects accurately