IPO Chart

Component: Utility Services

Module Name: Error

Author: Mia Park

Reviewers: Team 15, Bob Dahlberg

Inputs	Process	Outputs
Error_Code Language	1. Input from the Application Service Broker is checked for incorrect input. 2. SB will be called with language.txt file and error_code.	Error_Message

Inputs:
- Error_Code
- Language

Process Description:
1. Input from the Application Service Broker is checked for incorrect input.
2. SB will be called with language.txt file and error_code.

Outputs:
- Error_Message

APIs/Objects:
- Service Broker
- Text Broker

Pseudocode:
```
{ //Start Main
  /*------------------------------------------------------------------------------
   *      argsArray - String[] that holds string arguments passed at runtime
   *    language - String that holds passed language
   *    code - int that holds passed error code
   *    filer - String that holds the language.txt file
   *    set - String that combines TB + filer + code
   *     pb - Process builder that calls SB
   *    reader - var that reads from pb
   *     line - String that stores each line read from pb
   *------------------------------------------------------------------------------*/
  String[] argsArray = args[0].split(",");          // Arguments passed at runtime
```

IPO Chart

```
String language = "";                        // Passed language
int code = 404;                               // Error code
String filer = "msgEng.txt";                    // Language file
String set = ""                              // Parameter formatter
Process pb;                                   // Process Builder that calls SB
var reader;                                   // Reads from Process Builder
String line = ""                             // Printed line from reader
/*------------------------------------------------------------------------------
* @code
* Parse program arguments and store in argsArray
* IF (argsArray.length == 1)
*    Parse argsArray[0] into int and store in code
* END IF
* ELSE IF (argsArray.length == 2)
*     Set language = argsArray[0].substring(0,3)
*     Parse argsArray[1] into int and store in code
*     Set filer = "msg" + language + ".txt"
* END ELSE
* Set set = "TB" + filer + code
* Set pb = ProcessBuilder("java", "src/modules/SB.java", set)
* TRY (set reader = BufferedReader(InputStreamReader(pb.getInputStream)))
*     WHILE ((line = reader.readLine) != NULL)
*      Print line
*    END WHILE
* END TRY
*------------------------------------------------------------------------------*//End Main }
```