

Requirements for ActionLogger

Functional Requirements

Must Haves

- Logger must output to file
- Logger must add a timestamp
- Logger must add the origin (classname) of the log
- Logger must be able to be turned off
- Logger must be able to disable writing to file
- Logger must open a new file with every game start
- Logger must log all errors
- Logger must print additional text if a log is an error
- Logger must log all navigation inside the game
- Logger must log all collisions
- Logger must log all eat and die actions
- Logger must log score
- Logger must log growth
- Logger must log in the format: “[timestamp] – [classname] – [action]”

Should Haves

- Logger should be able to only print errors

Could Haves

- Logger could log errors to a different file (logs in log.txt and errors in error.txt)

Won't haves

- Logger will not manipulate existing log files

Non-functional requirements

- Logger must be written in java 1.8
- Logger must be delivered on 18th of september 2015
- Logger must be tested with a minimum of 75% line coverage

ActionLogger	
Superclass(es):	
Subclasses:	
Write given text to file	FileWriter
Get timestamp	Current Time
Open new log file with data & time in filename	FileWriter
Close logger	FileWriter

<<user interface>>	
ActionLogger	
- filewriter: FileWriter - dateFormat: SimpleDateFormat	
+ logLine(text: String, className: String) + logLine(text: String, className: String, isError: boolean) + close() - getTimeStamp()	