|  |  |
| --- | --- |
| D:\PAL\Logo_4_21_15\Logo_4_21_15\Primary Logo\png_files\EPAM_LOGO_Full_Color_RGB.png | CDP Automated Testing Mentoring Program  2018-Q2 |

MODULE 13: test reporting with JS libs

## home task

1. Add logging of every step (with log4js or winston lib) for your framework from the BDD module home task.
   1. Configure logs format in an informative way
   2. Demonstrate usage of different log levels (debug, action, error, etc)
   3. Configure ability to write logs in console and to save logs in a file (a new file should be created for each day)
2. Configure your framework to save test run results in a file and add the ability to create an

Html report based on that file.

1. Your framework should take a screenshot on test completion
2. Highlight at least one element before taking a screenshot.

## Bonus TAsk

* Configure transport logstash for Winston or log4js and send your test logs to a server by http

## acceptance criteria

1. Format of logs is informative (contains info about log level, time (e.g. YYYY-MM-DD hh:mm:ss,SSS), string of log, etc)
2. Different log levels are used (debug, info, action, warning, error)
3. By default logs are written in console and are stored in a file (a file should be created automatically – a new one for a new date)
4. Html report with screenshots should be provided after each test run
5. At least one highlighted element should be visible on the screenshot in the Html report

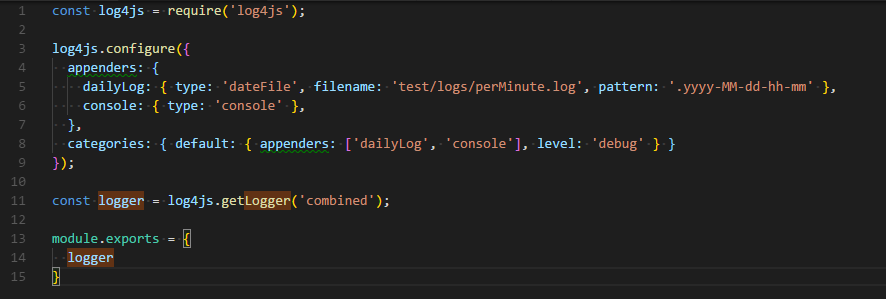
## home task

As a basis for Module 13 task I used my current project.

To run and view result for actions and executors part:

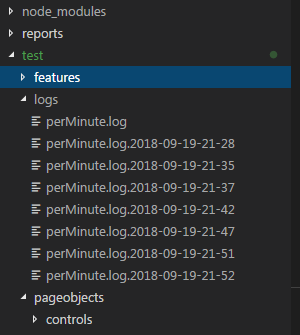
* copy code from repository **https://github.com/Alexey-Kachemirov/js\_atm**
* open **‘module13’** folder
* install project from copied folder running **‘npm install’**
* copy file with credentials **‘creds.js’** in the root project folder.
* run command from saml-idp project
* run test using grunt task runner, use command **‘grunt’.**

**Logger file is set up in ./test/support/loggerConfig.js**

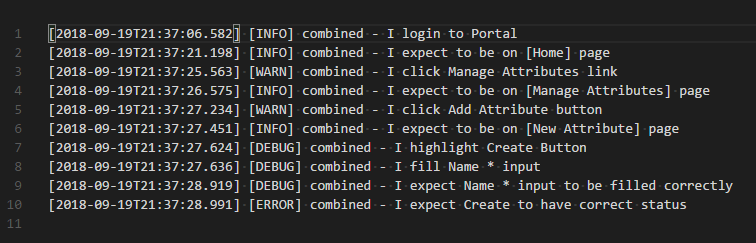


**1.2 – 1.3.**

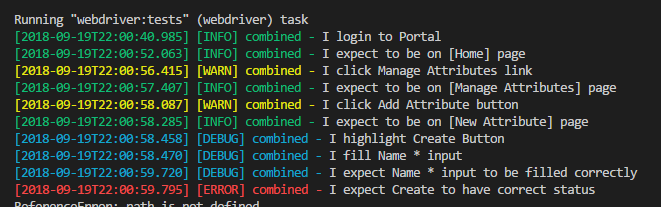
I set up writing logs in console and log file. New log file is created each minute. It uses pattern **‘.yyyy-MM-dd-hh-mm’**. Files are saved to the file **test/logs/perMinute.log**



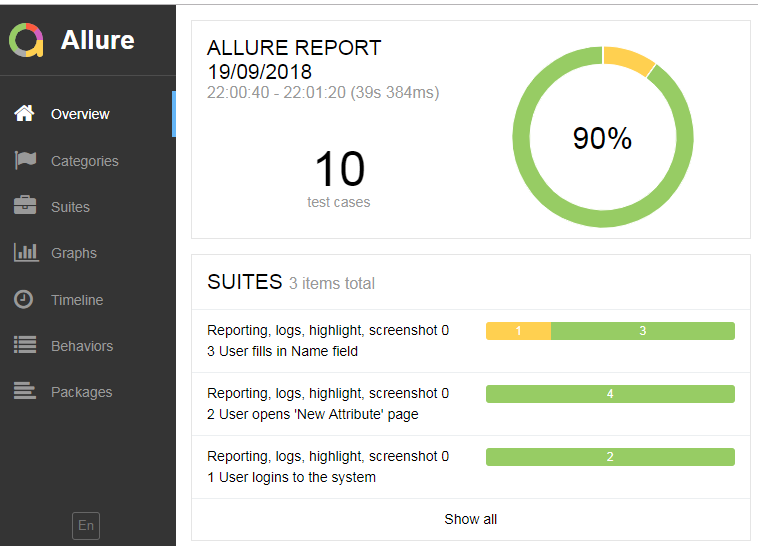
Each file contains proper logs:



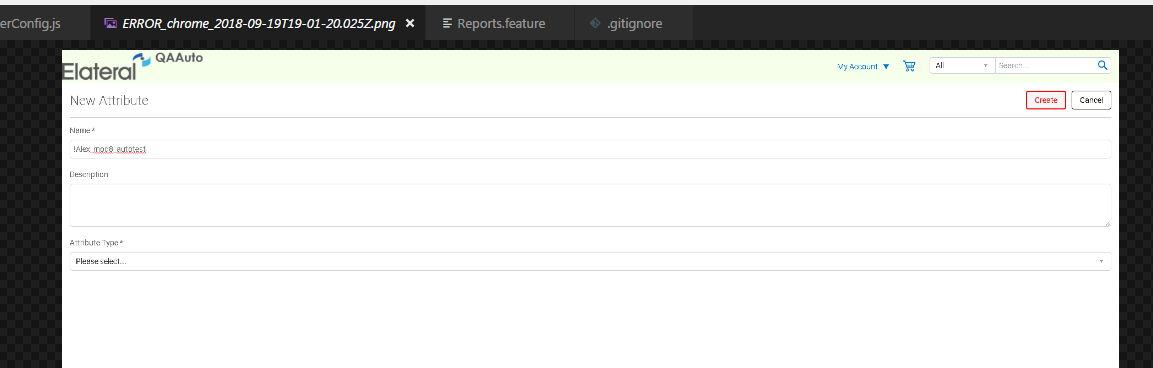
There are different logs levels are using during test run. For example:



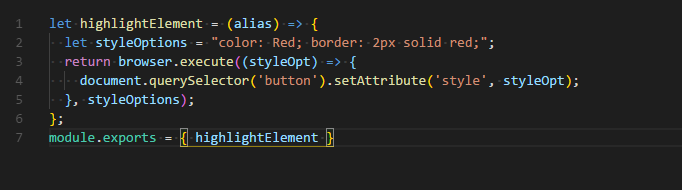
1. After each test run results are saved in json format file by the path: .**\test\reports\json-results\** and html report is built.



1. Screenshot is taken on test failure and saved by path: **./errorshots/**



1. Also added element highlighting. It is visible on previous screenshot. And implemented in file: **\test\support\highlightElement .** See screenshot below:

****