

Lisp in Summer Projects Submission

Submission Date	2013-10-20 22:48:35
Full Name	Javier Olaechea
Country	Peru
Project Name	trolling-success
Type of software	command-line/terminal app
General category	art
LISP dialect	Common Lisp
GitHub URL	https://github.com/PuercoPop/Trolling-Success
Did you start this project?	Yes, all the code is written by me
Project Description	I want to describe my project in this form.
Purpose	To be able to treat the contribution activity of github as a canvas
Function	It lets you write text on the github contribution activity.
Motivation	I was talking with a friend about the obsession of metrics and productivity, and although I like github I feel the contribution activity promotes the ideal of good programmer being the one that 'cranks out more code' so decided to subvert the contribution activity and treat it as a canvas.
Audience	My friends and anyone that shares my sense of humour. On second thought scratch my friends.
Methodology	<p>The basic idea is to map a position in the canvas to a date. The contribution activity displays in the days of the week along the y axis and the weeks along the x axis showing the activity in the last year. Hence the origin is set the current date in the previous year.</p> <p>In the original implementation uses a cursor that keeps track of the horizontal ('x') offset in the canvas. The letters are described in terms of a sequence of vertical bars. For</p>

convenience the constructor takes days instead of heights, sunday being the topmost and saturday the bottom one. using the cursor and the bar the method draw generates the appropriate git commits. Because all the letters require a space between them, initially, I decided to factor the boiler of placing an empty bar in an :around method. However after discussion with a more experienced CL brought the light that macros maybe opaque to runtime but around methods split the locality of implementation so it was rewritten as a macro define-letter. Another benefit is that being a macro the fact that it defines a method is hidden away so the information density of the code that established the binding between a letter and a group of bars is 100%, effectively creating a "DSL". And in the case of changing the implementation so that it doesn't use CLOS only the macro has to be rewriting leading to more flexible code.

Conclusion

It was a fun experience. After this implemented a game of life and intended to maintain a test account playing a game of life one tick an hour/day. However, github doesn't delete commit events when the history of a repo is rewritten that idea will have to wait for when one can 'erase' dots in the canvas.

Build Instructions

Clone it to ~/quicklisp/local-projects and then rename the version.num, file in the top level directory to version.lisp-expr

Test Instructions

There are no tests

Execution Instructions

1. Create a file similar to the the example provided

```
;;; Sample file
(ql:quickload :trolling-success)
(in-package :trolling-success)

(let ((canvas (new-github-cursor))
      (message "TROLLOLOL")
      (*git-author* "PuercoPolka")
      (*git-email* "pirata+polka@gmail.com"))
  (declare (special *git-author*)
           (special *git-email*))
  (write-to-canvas canvas message))
```

;; End of Sample File

2. sbcl --load .lisp
3. Create a Repo in github take not of the url.
4. git add remote origin (ie. git remote add origin https://github.com/PuercoPolka/demo.git)
5. git push origin master
6. Enjoy

Describe any bugs or caveats

For some reason I made the asd file depend on a file that isn't committed to the repo, I apologise for that crime against the judges time.

Also it requires interaction with github website as one can't create repos or delete repos through the API nor 'see' the result otherwise.

The whole alphabet is not implemented so one might have

to add some letters for a custom message.

Screen shots

□ [Screen Shot 2013-10-20 at 9.36.26 PM.png](#)

Official

I have read rules and have abided by them.
I am 18 years of age or older.
I am not living in Brazil, Quebec, Saudi Arabia, Cuba, Iran,
Myanmar (Burma), North Korea, Sudan, or Syria.