Meeting Notes: February 27th, 4p

Get coding!

Everett: kivy

images are widgets, can’t be put into stencils

works in float, but not scatter mode

stencils only for graphic instructions

Ashley: kivy

experimentation with url request

asynchronous, which can be confusing

wiki: has an API! (in references)

network requests as long as they’re in sequence, not in parallel

try to get kivy working on iPad

unittest: test class for each class

assertion for every method within classes

use Git!!!!!!!!!!

git clone <url>

make changes, add changes, commit w/message, push (git push <origin> master)

Everett & Sarah: deal with it

trygithub

add meaningful commit messages, atomic commits

create branches for volatile code, then merge into master branch

both functional and unit tests

functional: over-arching, to test bigger picture

unit test: test for each method

meta-classes & singletons

singleton: ensures only one instance of a given class

“@class method” definition

unit tests: supposed to be totally self-contained

don’t normally take parameters, just use mocked up data

coding tasks: just keep adding to github

GOAL: run tests before committing, should always pass

don’t commit something that doesn’t compile

continue to adapt to schedule

prioritization:

get main page, and then query whole ring of related nodes

visit all pages to get image and title

build a node for each related page

UI: find out when model is ready, then draw the nodes

\*\*Wikipedia API sandbox\*\*\*

make sure you use @classmethod

once we start coding, might need to meet and check in more often

**References:**

Unit testing: <http://docs.python.org/2/library/unittest.html>

Test driven design: <http://www.openp2p.com/pub/a/python/2004/12/02/tdd_pyunit.html>

Assert methods: <http://docs.python.org/2/library/unittest.html#assert-methods>

Github tutorials:

<https://help.github.com/articles/set-up-git>

<https://help.github.com/articles/create-a-repo>

Wikipedia API:

<http://www.mediawiki.org/wiki/API>