Meeting Minutes: April 5th, 2:30pm

Progress report:

Ashley: completed effort hours

TODO:

node placement

model

right now: kivy finds nodes, stores information there

but could let the model handle all that

kivy calls all the nodes on the screen in a touch event

touch down, touch move, touch up

touches propagate down object tree

can you select which nodes will respond?

can control whether it propagates down or not

each node has one parent

could send model coordinates of node to respond, wouldn’t have to search thru all nodes

don’t let kivy search through the children => stop search at node container

use hash table to store node coordinates?

putting in model would make testing easier

might not need to have event dispatch for the model

just call the node function?

whenever it's updated, call display

pass parent node reference

send page url, keyword, position

no sibling connections

model checks to see if it already has a given node; if so, draws node b/w child and that node

animating node restructuring would be relatively easy

still need a function to do graph of nodes and get coordinates

different ideas of the model’s function

model like a DB?

or presentation/UI layer just gets commands

controller routes to model

controller feeds from view to model

isolate these layers?

original plan: display can use the model, but never updates it

user event => controller => model => event => display

send event back through controller? but prob doesn’t hurt to send it directly to display?

kivy UI not a singleton

model needs to access UI object

send a reference to parent node

currently, zooming does not scale font within nodes

zoom in: bottom left-hand corner coordinate, zoom scalar

test communication b/w model and display

recenter camera to active node

run kivy full screen?

don’t worry about small screen devices

lock screen size

min/max/default zoom levels

when you’re choosing a new node, you need to know the coordinates of the old nodes so you

don’t intersect nodes/lines

buckets: divide coordinate plane into “buckets” to contain different nodes

network pretty slow

load html directly instead of using API

compare load time

can get images alone, would be faster for initial display

user has to wait for two network requests: search, and page

ashley: get two homework questions finished by next meeting

sarah: add a value for max number of links (priority-wise)

implement prioritization function

parser & network are now hooked-up!

ULTIMATE GOAL: (triforce) connect display to everything else!

Get 1) and 2) done by 4/10

Next meeting: 4/10, 4pm