To-Do: -assign issues to group - revise iteration & plan Implementation - DSD classes for all (planner, vendor, event, eventlist, vendorlist -logic nueded - unit tested for complex logic (Junit) All Tests. java - interface for database - Stub imp of db (List, Map) - GUI with at least one feature - same code style - divide into packages based on architecture

Architecture

- md file

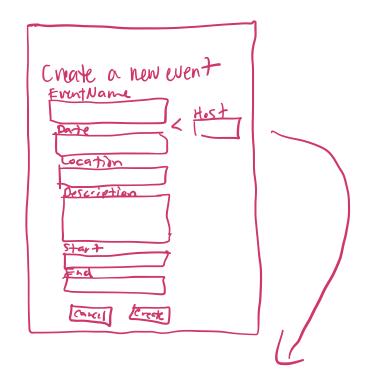
- sketch

- move incomplete issues to MS 2

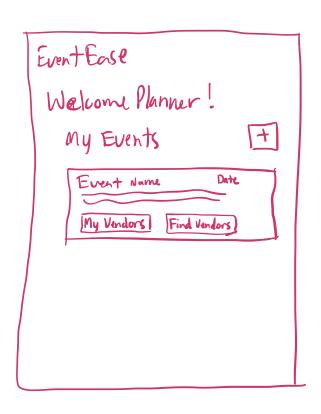
- Handin git

GUI - Add event





II I'm planning this event for myself
II I'm planning this event for a client



US: Add event
Class Event Event Host
Var: Event Name
Location
Date
Start Time
End Time
List Vendors
Description
Fun: Constructor

Edit event Class: event

Fun: Set Event Name
set Location
set Date
set Start Time
set End Time
Set Description
add Vendor To Event (Vendor)

Remove Event class: event

Fun: delete Event

Source Kindos

class = vendor

var: vendor Name Service Type description phone Number

email rate description

rating

tun: Constructor + add to vendoclist

vendorpictures [] Capacity - find vendors for event (show a list) class: vendor list

> var: vendorlist [] list title

fun: title by search add to list remove from list

- filter vendor options by criteria class: vendorlist

vor: fifterlist []

TIMOR (15) LJ

fun: add filter by service
remove filter by service
add filter by Cost
remove filter by Cost
sort Alpha Asc
sort Alpha Desc
sort Cost Low High
Sort Cost Low High
Sort Rating High Low
Sort Rating Low High

class: vendor

fun: getserviceType

add filter by Rating remove filter by Rating resetfilters

-view Vendor details class: vendor

fun: get Vendor Name
get Vendor Description
get Vendor Rate
get Vendor Rate
get Vendor Rate Description
get Vendor Number
get Vendor Email
get Vendor Pictures

-vendors respond to queries class: vendor dashboard

Var: Vendor [] Vendor Contact Name # Push to Id?
US: Accept/Rejuct
Event Task nequests
T Dupe?

class: vendor Requists

var: vindor bool status

planner regisest (Pending)

vendor requist (Action required)

vendoraccept (Accepted)

vendoraccept (Declined)

awaiting Event (Wait for Event)

order frogress (order in Progress)

Order Complete

fun: Set Status Constructor -> when planner sends request

Sort Events

- event sorting class: eventlist

> Var: events[] Num events

Fun: constructor -> when planner makes account
Sort by Event Date
Sort by Num Vendors Asc
Sort by Num Vendors Desc
Sort by Num Grust Asc
Sort by Num Grust Desc
add Event To List

remove Event From List

Architecture