



# Final Design Presentation

July 29, 2008

# AGENDA

1. The Team
2. Problem Space
3. Research
4. Design Process
5. Design&Rationale

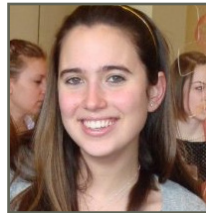


# THE TEAM

# CMU



**GEM BLEASDELL**  
Project Manager



**KORINA LOUMIDI**  
User Testing Lead



**ADAM MATTHEWS**  
Client Liason &  
Documents Lead



**BHARATHI PITTI**  
Technical Lead



**CEREN SAKIZLI**  
Design Lead &  
Web Master



# OUTSYSTEMS

Carlos Alves

Rodrigo Castelo

Rodrigo Coutinho

Lúcio Ferrão

António Melo

Irene Montenegro

# PROBLEM SPACE





# PROBLEM SPACE

## Agile Environment

- Fast Paced
- Rapid Turnover

## Target User

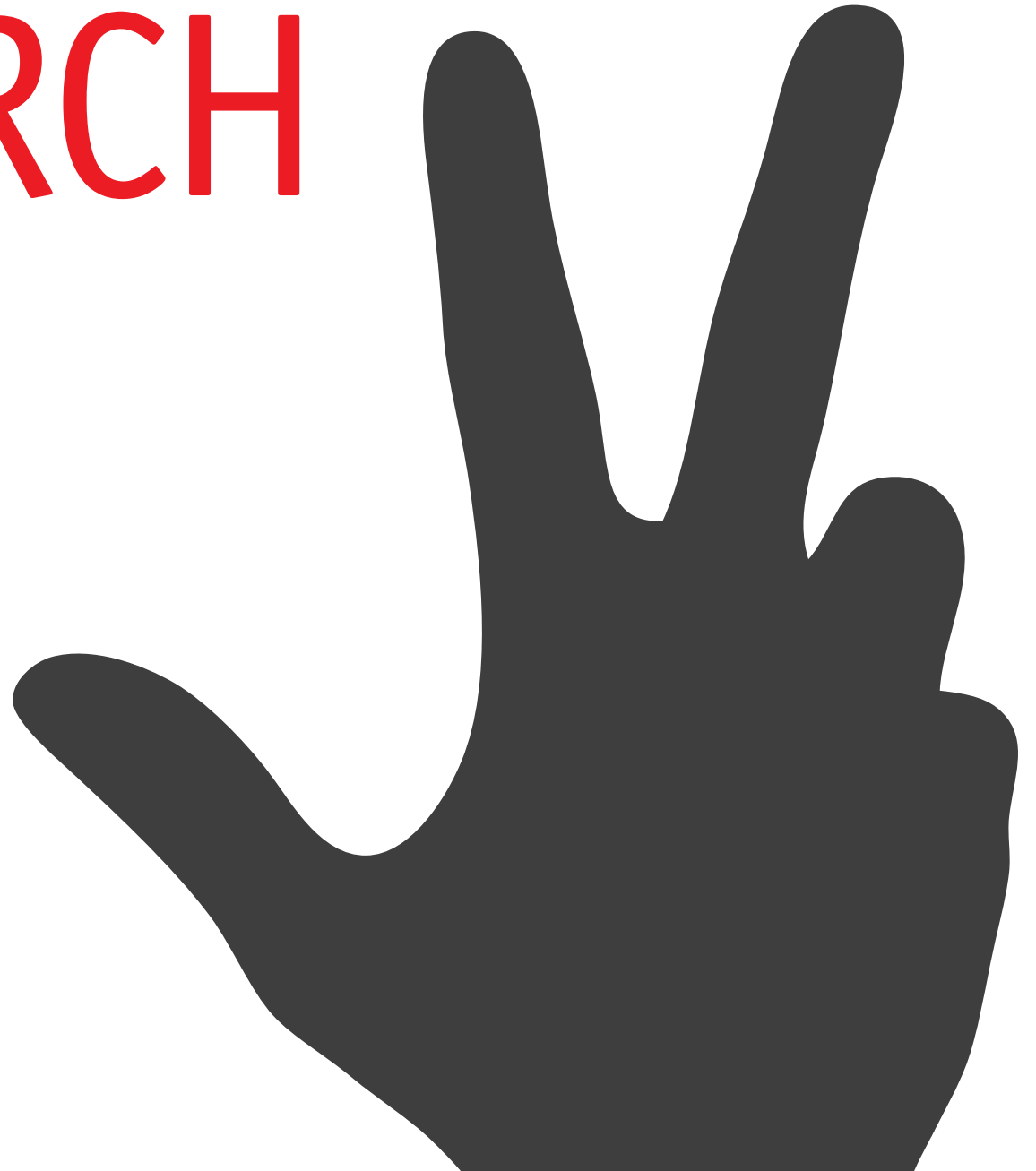
- Business Developer
- Less Technical
- Interested in results

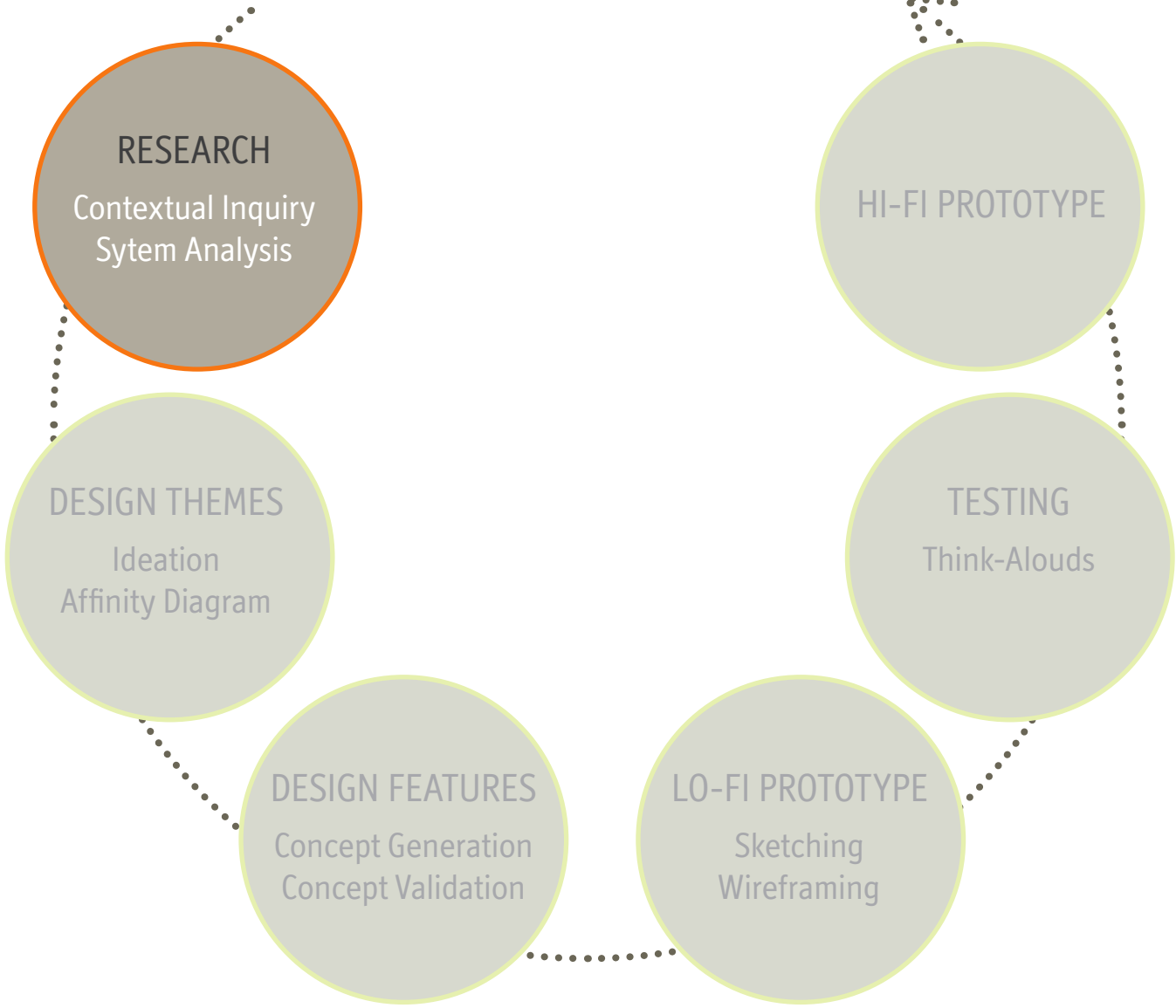
## Current System

- Image Look & Feel
- Layout Organization
- Interaction Paradigm
- Information Visibility



# RESEARCH



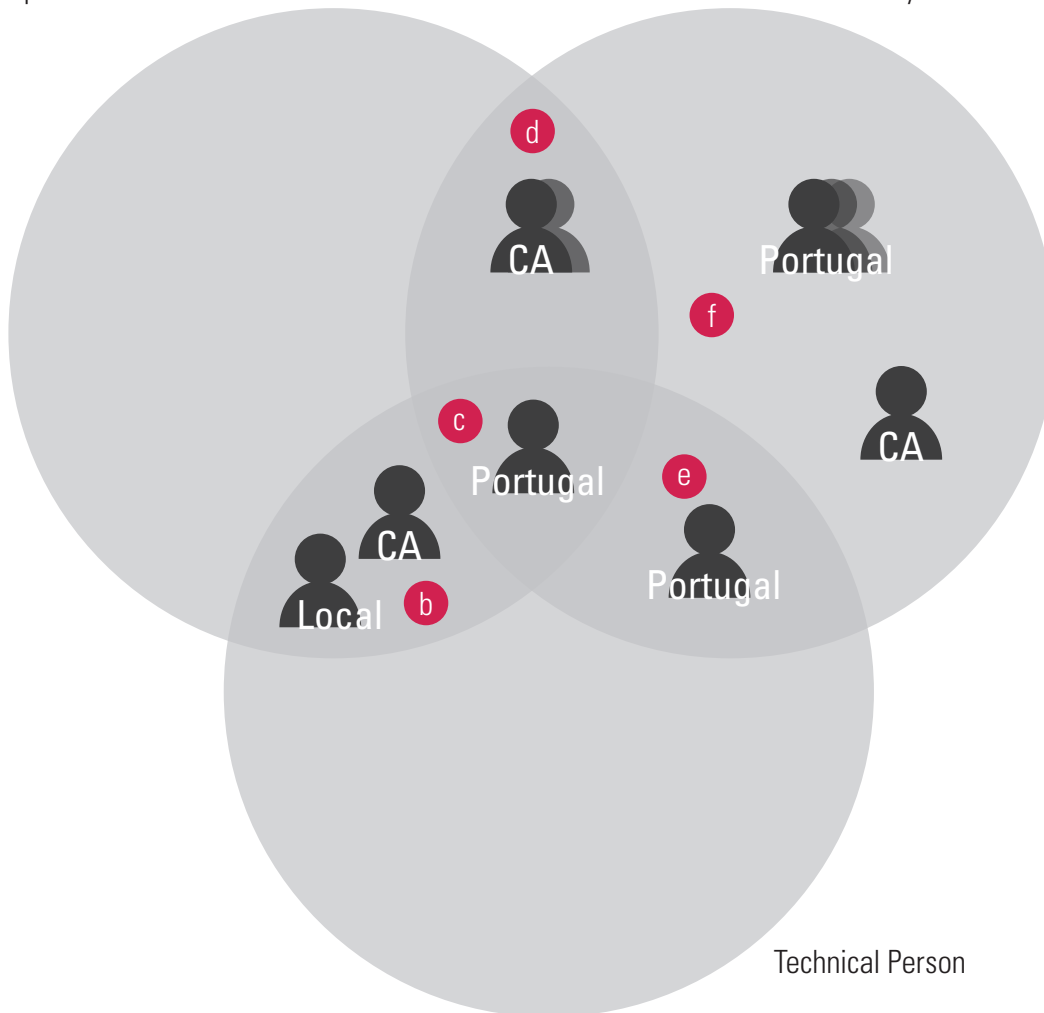


— Spring  
— Summer

# CI PROCESS

Experienced Developer  
in other platforms

Experienced Developer  
in OutSystems Platform



- b** One (1) local CI and one (1) CI in California with technical developers who are experienced in other platforms
- c** One (1) CI in Portugal with technical developers who are experienced in multiple platforms
- d** Two (2) CIs in California with non-technical developers who are experienced in multiple platforms
- e** One (1) CI in Portugal with technical developers who are experienced only in OutSystems platform
- f** Three (3) CIs in Portugal and one (1) CI in California with non-technical developers who are experienced only in OutSystems platform

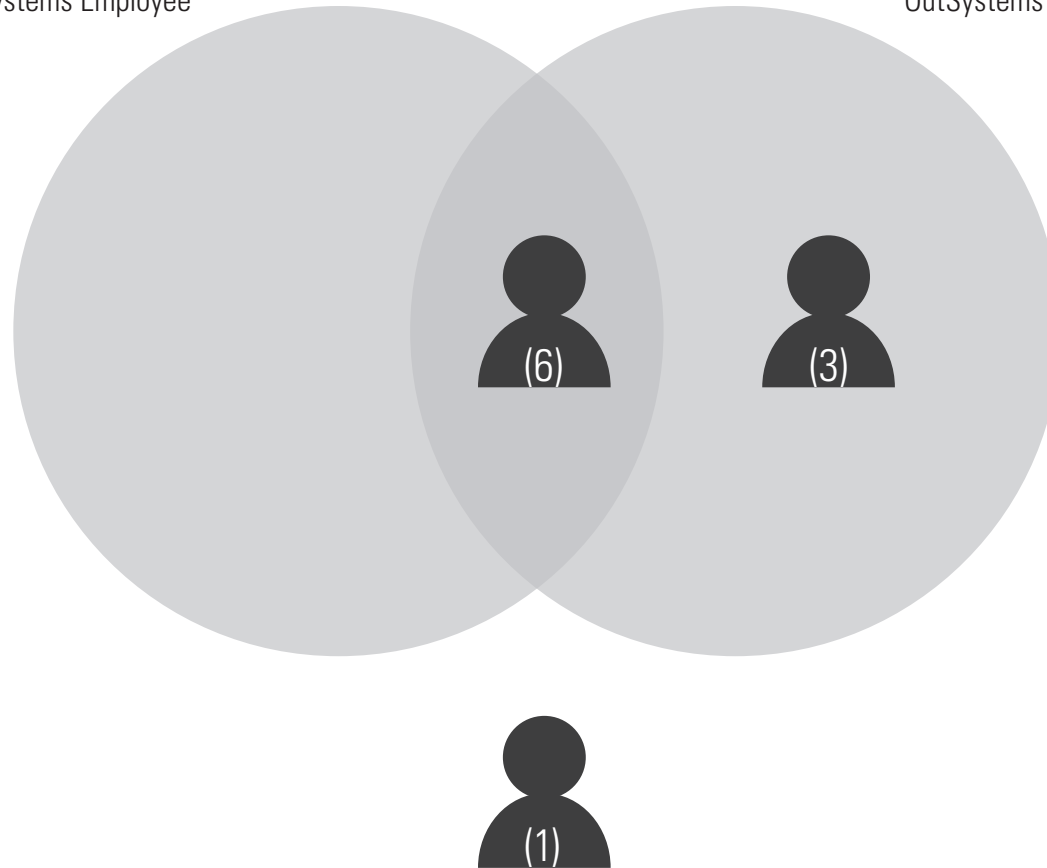
---

Total = Ten (10) CIs

# CI RESULTS

OutSystems Employee

OutSystems Platform user





# SYSTEM ANALYSIS PROCESS

MS Visual Studio 2005

MS Sharepoint

MS BizTalk

Salesforce.com

Apex

Force.com

PAAS

Skyway Software

Composite Software

Wave Maker

Nexaweb

Cape Clear

Inuvia

Yahoo! Pipes

IBM QEDWiki

Microsoft PopFly

Serena

CogHead

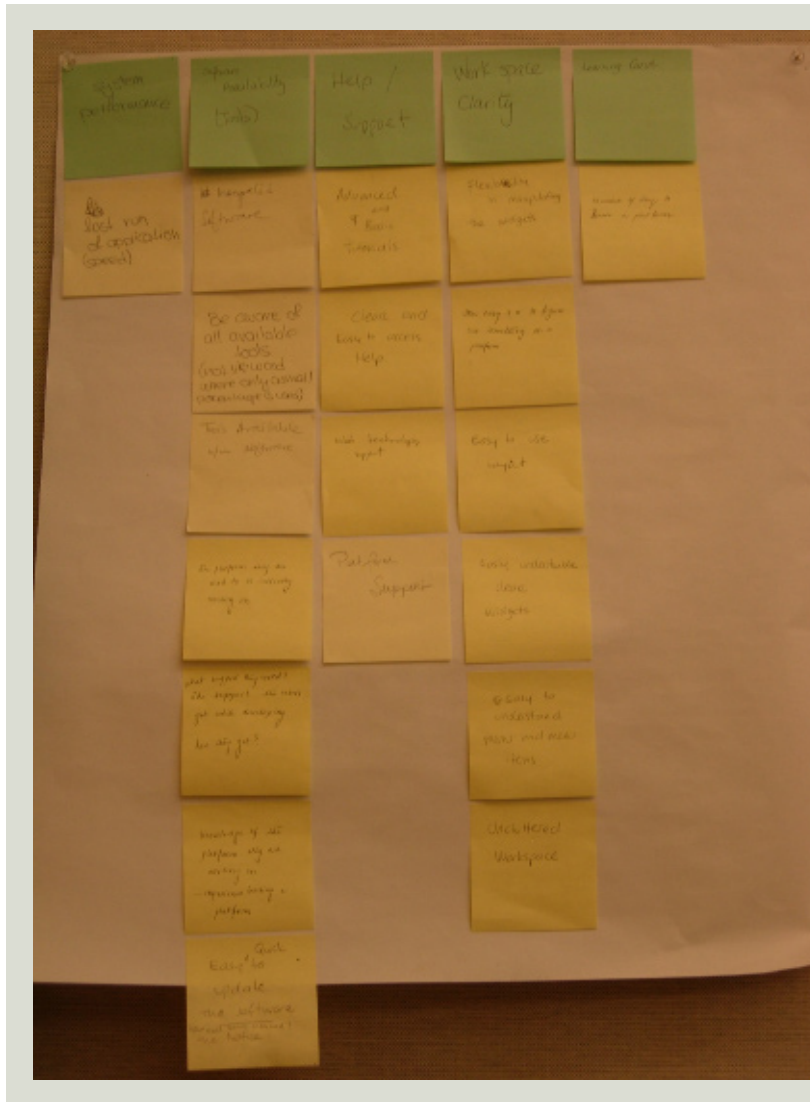
WinDev

Adobe Flex

# SYSTEM ANALYSIS RESULTS

Competitors	Biz Talk	Sharepoint	Wavemaker	Force.com	Capeclear
Features					
Customize applications	X			X	
Programming skills required	X	X	X		X
Integration	X	X		X	
Installation required	X	X		X	X
Visual Display	X		X		X
Colloboration		X			

# AFFINITY DIAGRAM



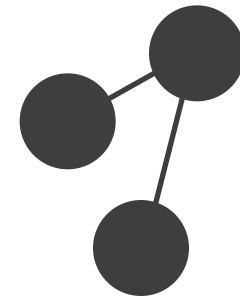
# FOCI



communication



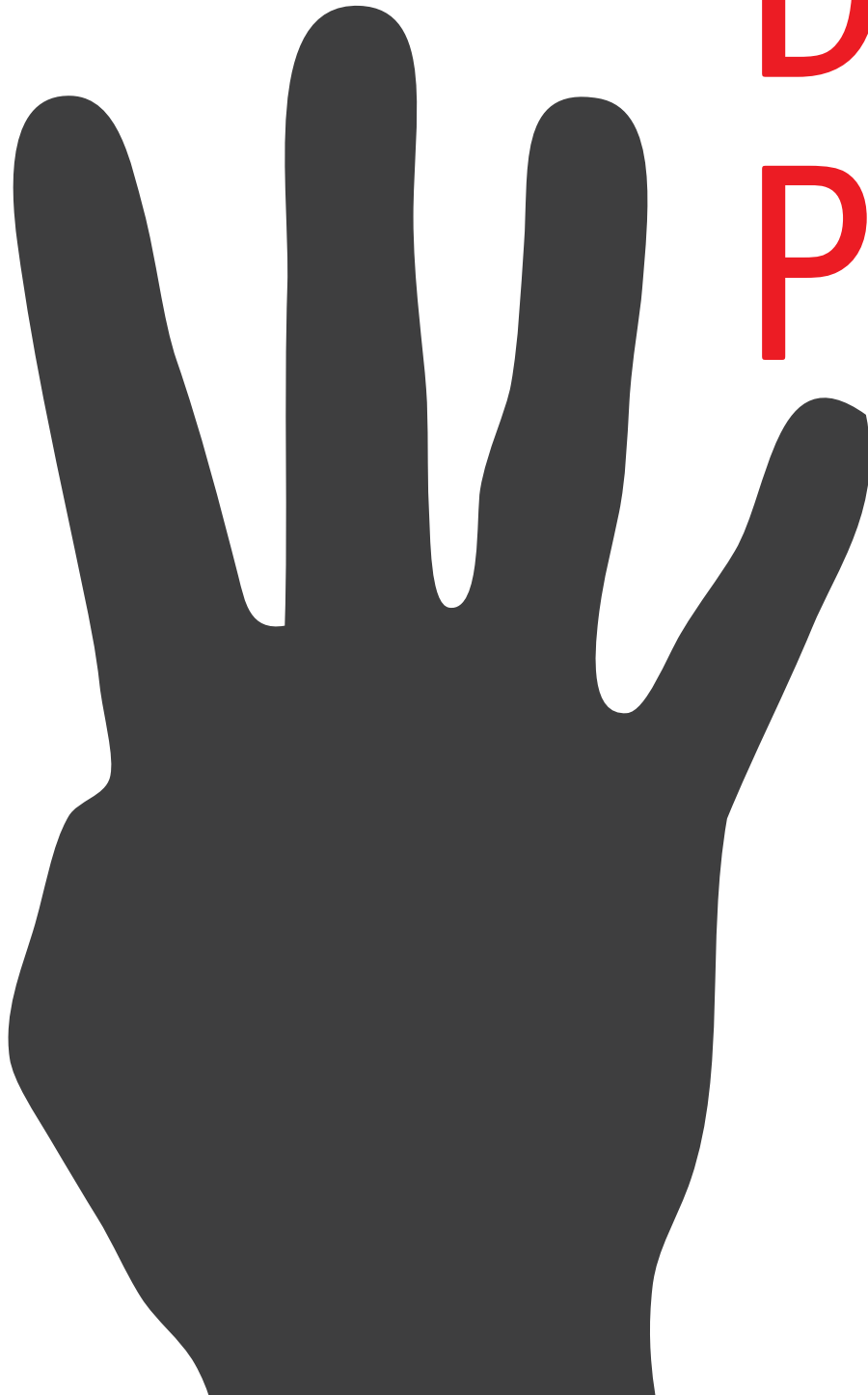
maintenance

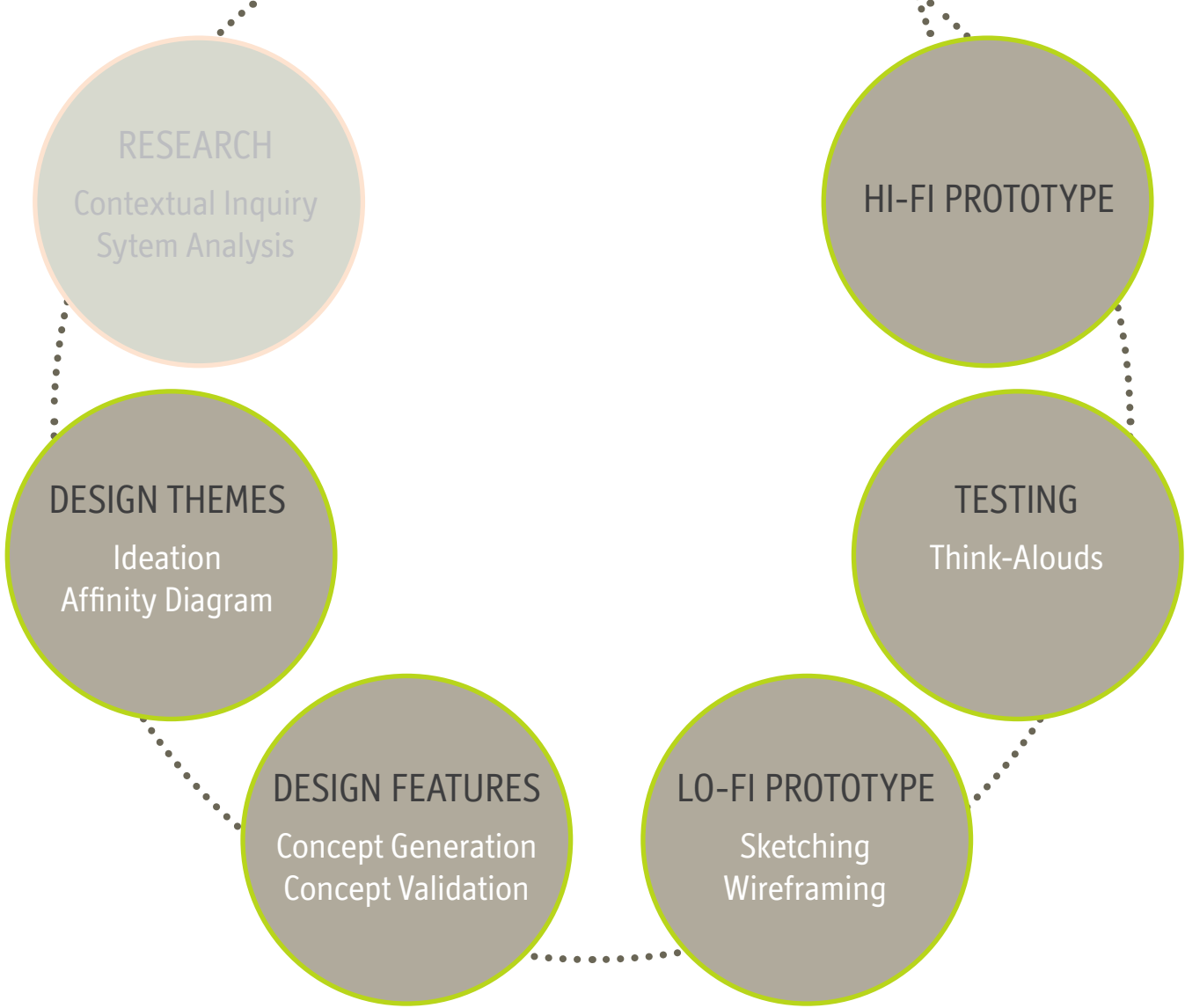


visual  
representation  
of code

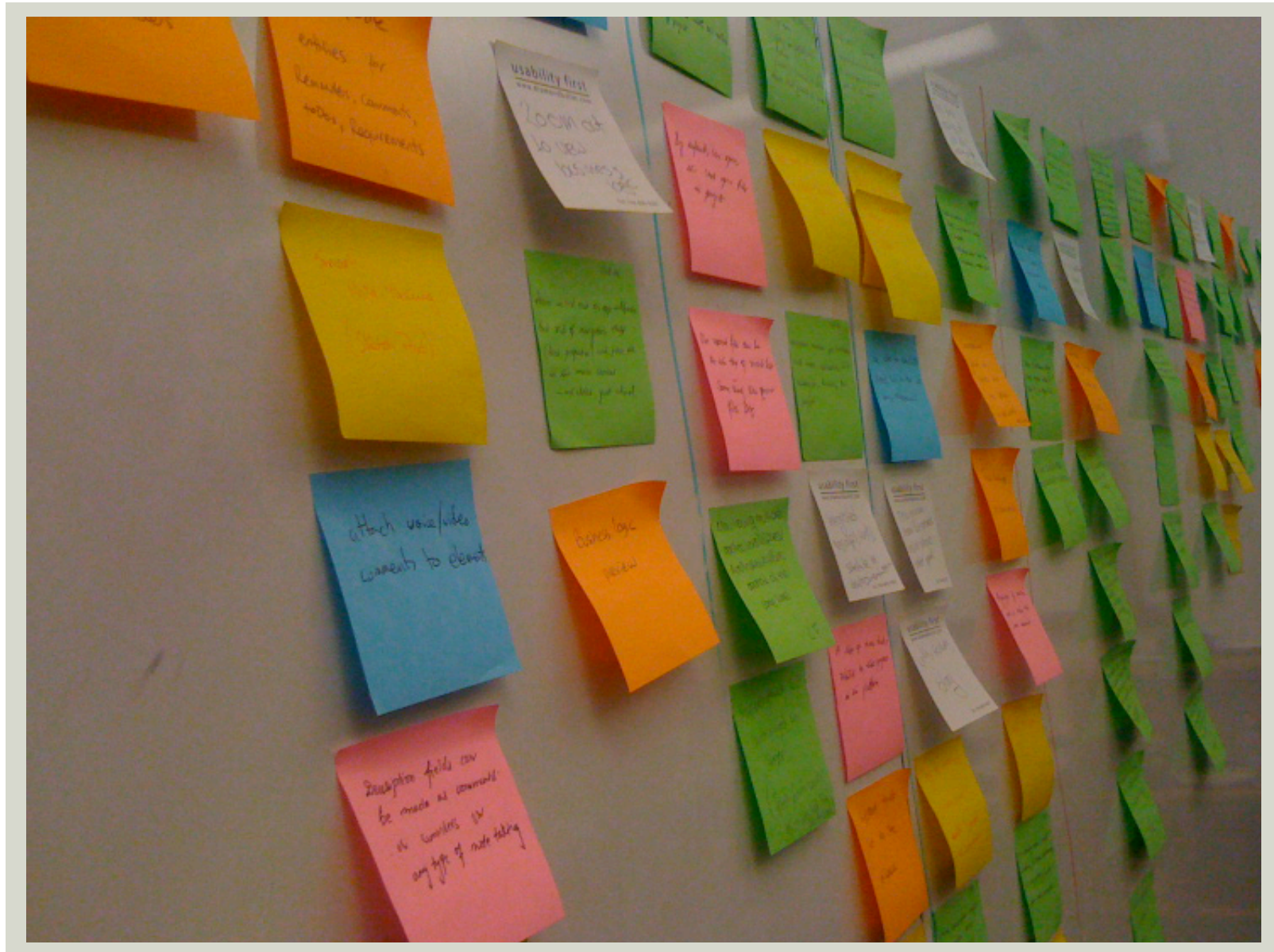


# DESIGN PROCESS





# BRAINSTORMING





# DESIGN THEMES

Daily Cycle

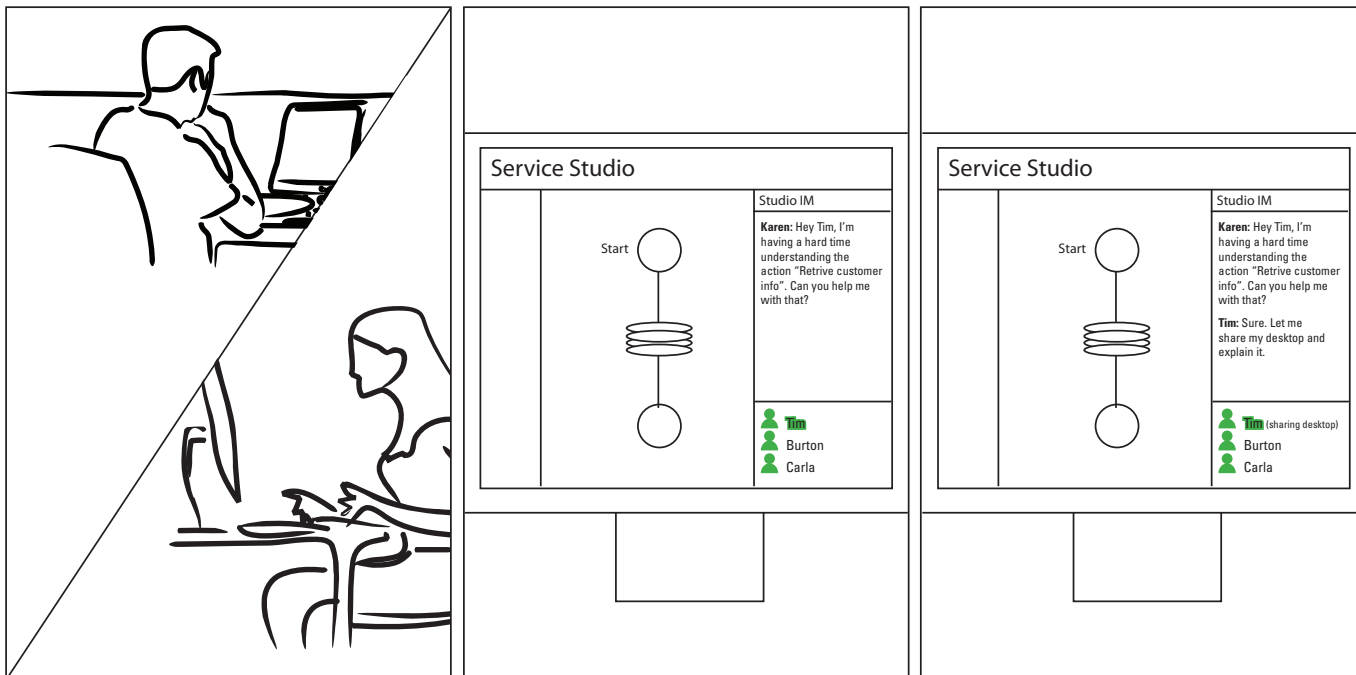
Flexibility

Adaptation

Intuitiveness

# CONCEPT VALIDATION

## 4. Instant Messaging with other developers

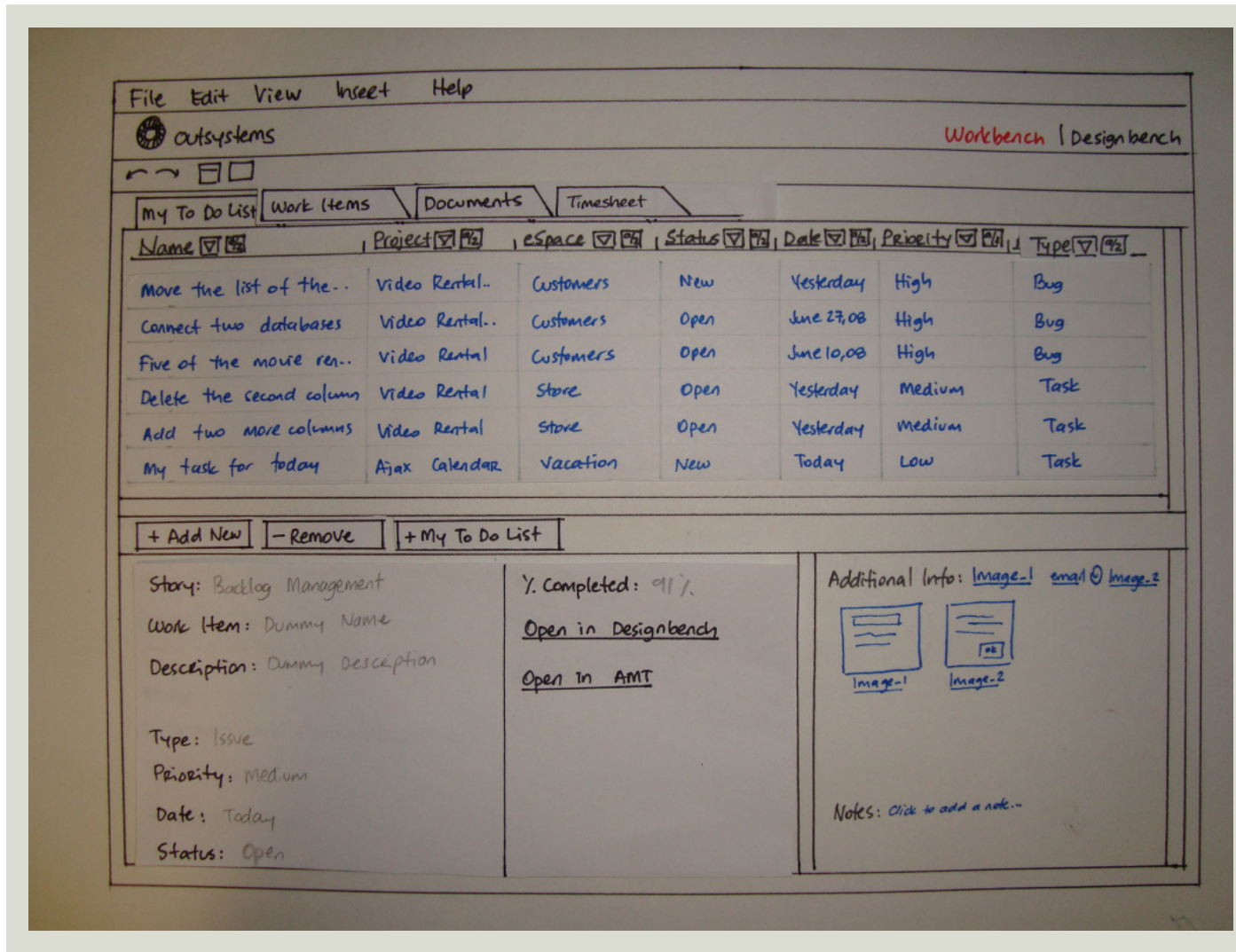


Karen is collaborating with other developers working on the same project. She can see in real-time with her built-in chat how many developers are actually working on the same

Karen doesn't understand the part of the code written by another developer. She realizes that it was written by Tim. She starts a chat with Tim asking for help with the code.

Tim shares his desktop with Karen and explains the business logic he had created.

# WIREFRAMES





# PROTOTYPES

Low-Fi (Paper)

Med-Fi (Digital)

Hi-Fi (Digital)



# DESIGN & RATIONALE





# DEMO

+ explanations



My To Do List

- Today
  - Backlog Management combobox
  - Visual distinction of WIs by type
  - Add row allocation view
  - View more items without paginati
  - Use in Backlog screen username, i
- Yesterday
- July 18, 2008

PROPERTIES

Name: Action 1  
Description: Lorem ipsum  
Type: Action  
Input: Lorem ipsum  
Output: Lorem ipsum

NOTES

1/5

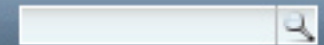
SupplierServices (eSpace)

Root > SupplierServices > LoginFlow > Delete



DEBUG AREA





My History

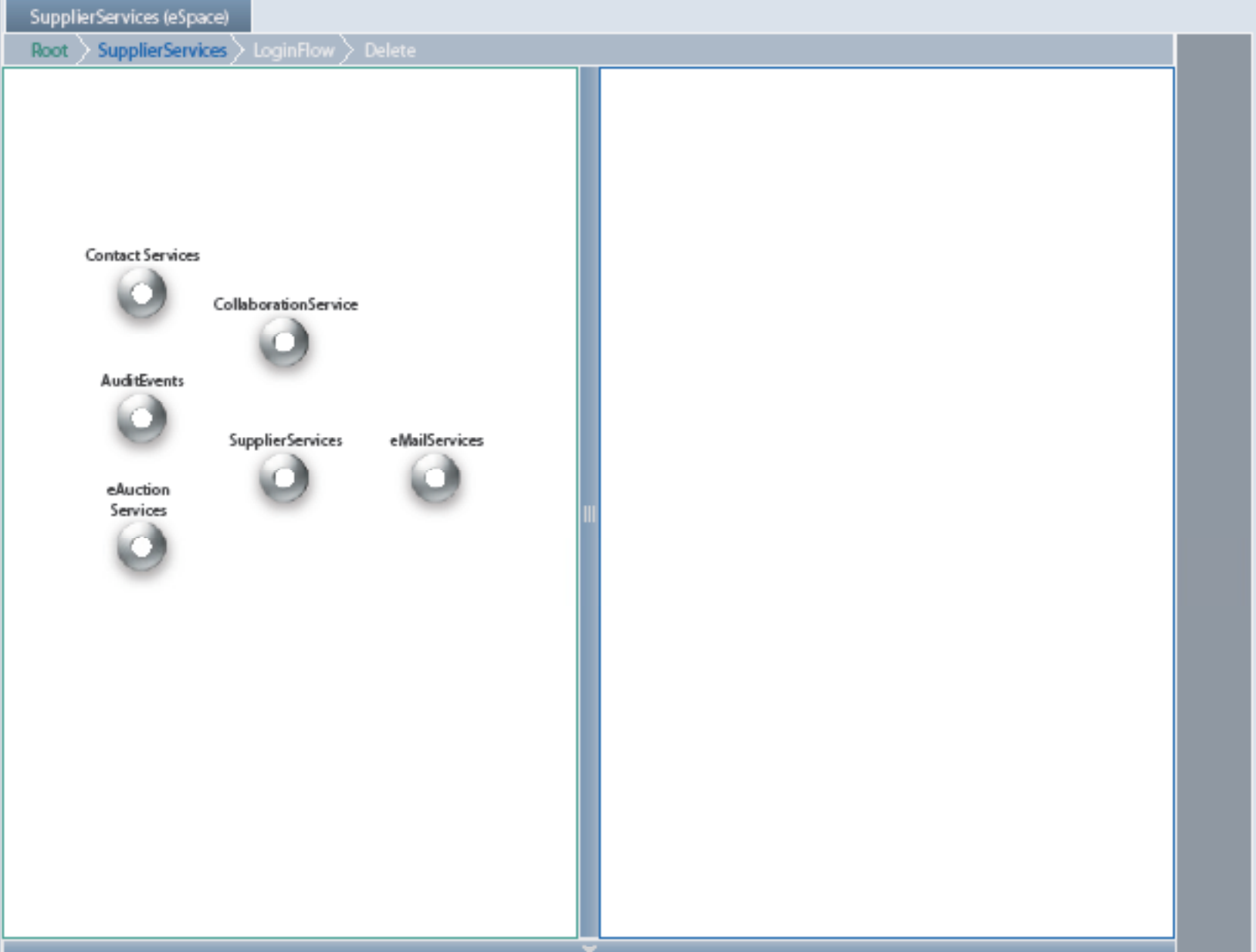
- Today
  - Database Customers created
  - Login\_Flow web screen created
  - Added Customer entity to Login
  - OK, Cancel button added to Log
  - Merged two screen flows entities
  - Deleted the Cancel button from
- Yesterday
- July 18, 2008

PROPERTIES

Name: Action 1  
Description: Lorem ipsum  
Type: Action  
Input: Lorem ipsum  
Output: Lorem ipsum

NOTES

1/5



! DEBUG AREA



**My History**

Today

- Database Customers created
- Login\_Flow web screen created
- Added Customer entity to Login
- OK, Cancel button added to Log
- Merged two screen flows entities
- Deleted the Cancel button from

Yesterday

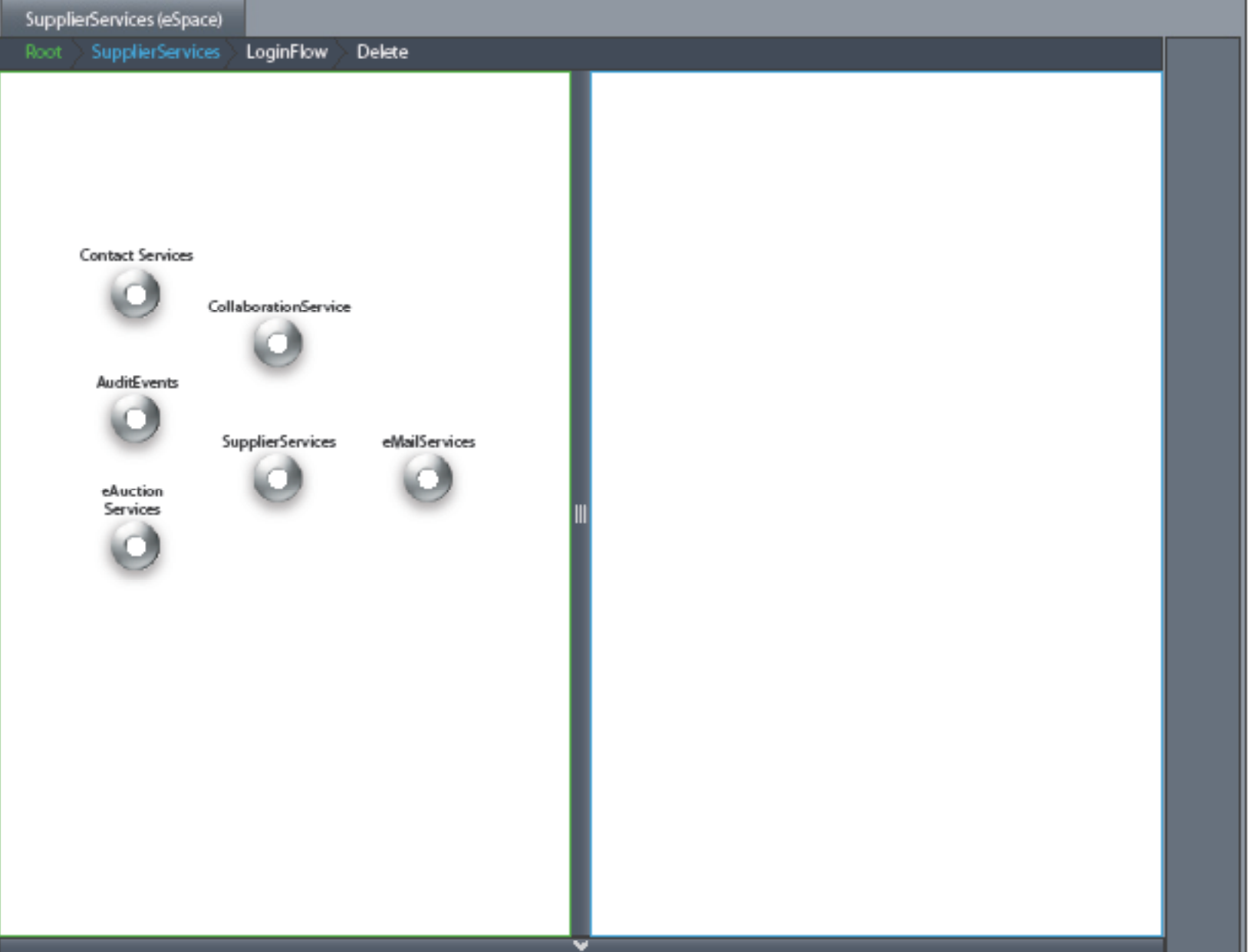
July 18, 2008

**PROPERTIES**

Name: Action 1  
Description: Lorem ipsum  
Type: Action  
Input: Lorem ipsum  
Output: Lorem ipsum

**NOTES** + - < 15 >

Empty note area



! DEBUG AREA



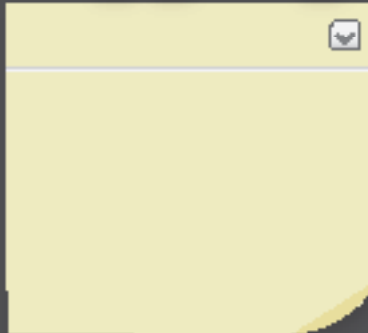
My History

- Today
  - Database Customers created
  - Login\_Flow web screen created
  - Added Customer entity to Login
  - OK, Cancel button added to Log
  - Merged two screen flows entities
  - Deleted the Cancel button from
- Yesterday
- July 18, 2008

PROPERTIES

Name: Action 1  
Description: Lorem ipsum  
Type: Action  
Input: Lorem ipsum  
Output: Lorem ipsum

NOTES + - < 15 >



SupplierServices (eSpace)

Root > SupplierServices > LoginFlow > Delete



! DEBUG AREA



# ADJUSTABLE SPLIT PANELS

## Solves

- Maintenance
- Layout Organization
- Information Visibility
- Interaction Paradigm

## User Research

- Need for Comparison
- Code Reuse

## User Feedback

- Dual Purposed
- A favorite
- Screen real estate must be considered



# CONTEXTUAL TOOLBARS

## Solves

- Layout Organization
- Maintenance
- Image Look & Feel

## User Research

- Too many items in tree
- Difficult to find items needed
- Items arranged by type
- Tree did not represent relationships

## User Feedback

- Apprehensive about losing tree
- Good to have what you need when you need it



# SEARCH

## Solves

- Maintenance
- Visual Representation of the Code
- Interaction Paradigm

## User Research

- Code Reuse
- Trouble locating items
- Easy Access to code

## User Feedback

- Not tested





# INDEPENDENT/DEPENDENT BROWSING

## Solves

- Maintenance
- Information Visibility
- Interaction Paradigm

## User Research

- Code Reuse
- Code Comparison
- Easy Access to code
- User loses context

## User Feedback

- The mix is good
- Both functionalities necessary
- Helps maintain context



# THUMBNAIL VIEW

## Solves

- Information Visibility
- Layout Organization

## User Research

- User loses context

## User Feedback

- Good to keep context
- Must be able to turn off



# TOGGLING SCREEN REAL ESTATE

## Solves

- Layout Organization
- Maintenance
- Visual Representation of Code

## User Research

- Complex & Extensive code
- Difficulty Navigating

## User Feedback

- Good to be able to maximize screen real estate



# BREADCRUMBS

## Solves

- Layout Organization
- Maintenance
- Information Visibility
- Interaction Paradigm

## User Research

- User loses context
- Cannot jump between items

## User Feedback

- Great to see where you came from



# BREADCRUMB PREVIEWS

## Solves

- Information Visibility

## User Research

- User loses context

## User Feedback

- Not Tested



# NOTES

## Solves

- Communication
- Maintenance
- Information Visibility

## User Research

- Commenting is poor
- Commenting is a nuisance
- User takes a long time to figure out what code is doing
- Many types of notes created

## User Feedback

- Users like the notes
- Needed to be more visible



# COLLABORATION

## Solves

- Maintenance
- Communication

## User Research

- Users worked on same items
- Changes could clash

## User Feedback

- Would like to be able to see in line of code
- Not always on
- Real-time not always visible



# TODO

## Solves

- Communication
- Maintenance

## User Research

- Tasks often looked at to start day
- Users interact with tasks constantly
- Users often build a list of tasks from the assigned

## User Feedback

- Tasks available in the morning





# HISTORY

## Solves

- Maintenance
- Information Visibility

## User Research

- User forgets where they left off
- Introduced bugs difficult to locate

## User Feedback

- Very Useful
- Multi functional
- Would like to see all related items



# DEBUGGING

## Solves

- Maintenance

## User Research

- Time loss

## User Feedback

- Real-time debugging preferred
- May not notice
- Tested on Paper



# DIAGRAMMING

## Solves

- Maintenance
- Visual Representation of the Code
- Information Visibility
- Interaction Paradigm

## User Research

- Code Reuse
- Code Comparison
- Easy Access to code
- Too many items in tree

## User Feedback

- More comfortable about tree
- Still missing items from tree



# ZOOMING/PANNING

## Solves

- Maintenance
- Information Visibility
- Interaction Paradigm

## User Research

- Code Reuse
- Code Comparison
- Easy Access to code
- User loses context

## User Feedback

- leaving the body text here for you to quickly change it.



# FUTURE STEPS

Thumbnail View

Previews

Toolbar Design

Sonar Mode

Magnification Mode

Adding Entities/Actions/Web Blocks

Map

Search

Debugging

THANK YOU  
OBRIIGADO