Rose-Hulman Institute of Technology

University of Wisconsin–Madison | Department of Computer Sciences
Human-Computer Interaction Laboratory

Milestone 2

Trey Cahill Chris Gropp Samad Jawaid Kevin Risden

October 5, 2011

Contents

1 Executive Summary

This document's purpose is to detail the participant scheduling system proposed by the Human-Computer Interaction Lab of Wisconsin-Madison. It is the second document describing this project, and details project design through use cases, data flow diagrams, and storyboards. The project exists because the lab wishes to unify their schedule information and provide a simple, intuitive interface for prospective participants to sign up for experiments.

2 Introduction

The Human-Computer Interaction Lab at the University of Wisconsin-Madison wants a web-based system to better manage the scheduling of participants for their studies. These studies range from one-on-one experiments to group interactions, and many of them involve the robot used by the lab. Currently, each researcher arranges studies independently via email and is responsible for scheduling rooms, avoiding conflicts, and notifying participants of changes; unifying this information onto one system simplifies all of these tasks. To the client, the most important benefit of a unified system is the ability for participants to easily browse all available experiments, which is not possible over email. However, a variety of other functionality should be integrated into this utility to take advantage of the unity of information; most notable is recognizing room conflicts when scheduling studies, since the lab has only one robot and it cannot be moved.[1]

Project information will be documented as follows: Milestone 1 provides an overview of the project, from client background to key features and requirements. Milestone 2 covers the behavior of the system, including use cases and data flow diagrams. Milestone 3 details constraints, back-end requirements, and elaborates upon the user interface. Testing and maintenance information can be found in Milestone 4. Milestone 5 will include usability data and interface re-design related to such data.

3 Client Background

The client is the Human-Computer Interaction Lab at the University of Wisconsin-Madison. Their research focus is the on the way humans perceive computers, and how this perception influences their actions. The main goal is to learn about this interaction through making hypotheses, experimenting, analysing the data, and then publishing papers on the results. They draw the participants for their experiments from a wide range of people, usually ranging from 18-65 years of age and from diverse technical backgrounds. As such, any system they use must be designed for all levels of technical competency.

4 Current System

Each researcher has their own method of handling participant scheduling. For most the current system is to have the participants email the individual researcher and then that researcher records the time slot in some sort of excel spreadsheet. Other researchers have tried Google Calendar appointment slots; while this is a better system, not everyone uses it and the client believes it is too complex for most participants and some researchers. Addressing the lack of unified data and superfluous effort on the part of the participants is the primary goal of the project.

5 Product Overview

This section provides a high-level view of the product capabilities, interfaces to other applications, and system configurations.

5.1 Product perspective

The participant scheduling system will be a new product. It will be used to schedule experiments and participants in the Human-Computer Interaction Lab at the University of Wisconsin-Madison. The product is independent and totally self-contained; it is not a component of a larger system.

5.2 Elevator Statement

For the researchers in the Human-Computer Interaction Lab at the University of Wisconsin-Madison who currently schedule experiments and participants with rudimentary tools such as pencil and paper, email, or Google Calendar, the participant scheduling system will be a web application that will streamline the lab's scheduling process. Unlike current solutions, this application will be the same for every researcher, so it will also be easier for participants to be a part of multiple experiments.

5.3 Summary of Capabilities

Here are the major benefits and features the product will provide.

Customer Benefit	Supporting Feature
List of participants for an experiment	Reports
Room availability (avoid conflicts)	Overall lab schedule
Simple sign up	Intuitive user interface
Track all experiments	Experiments manager
Access from anywhere at any time	Web application

5.4 Assumptions and Dependencies

- The participant scheduling system will be a web application.
- The server has the necessary operating system and software.
- There is no integration with any other system.
- There is no import of existing data.

5.5 Rough Estimate of the Cost

There is no monetary cost for this project, because the software development, as part of a college class, is free. Furthermore, the client will be provided with free servers through the University of Wisconsin-Madison for the finished product. The client will perform maintenance and management on their own.

6 Features

ID	Feature	Status	Priority	Effort	Risk	Stability	Target Release	Reason
1	Browse Experiment	Approved	Critical	Medium	Medium	Medium	1st	Lets experiments be advertised better and to display the experiments
2	Persistent Experiment Storage	Approved	Critical	Medium	Low	High	1st	Store experiment for the data to be web based.
3	Levels of Authentication	Approved	Useful	Medium	High	Medium	3rd	Have levels of administers, workers and participants in order to control privacy issues and other sensitive data
4	Participant Schedule Experiment	Approved	Critical	Medium	Medium	Medium	1st	Participant can schedule experiment slot
5	Filter Experiments	Approved	Useful	Medium	Low	High	2nd	Filter the experiments when browsing according to Time, Date, Payment, etc.
6	Experiment Participants	Approved	Important	Low	Low	High	2nd	View all of the participants by admins and workers only of individual experiments
7	Cancel Experiment Appointment	Approved	Useful	Medium	Medium	Medium	3rd	Cancel participant scheduled appointment
8	Add Experiment	Approved	Important	Medium	Medium	Medium	2nd	Add experiment from admins view
9	Modify Experiment	Approved	Important	Medium	Medium	Medium	2nd	Modify or Edit experiment from admins view
10	Notify Participant when Creating Appointment	Approved	Useful	Medium	Low	High	4th	Send an email reminding participants of participation in an experiment
11	Notify Participant Appointment Reminder	Approved	Useful	Medium	Low	High	4th	Send an email or text reminding participants for their experiments
12	Notify Participant Appointment Cancellation Reminder	Approved	Useful	Medium	Low	High	4th	Send an email or text reminding/telling participants of cancellation of their experiments
13	Export Experiment Participant List	Approved	Useful	High	Low	High	4th	Reports on experiments scheduled with an option for Individual experiments reports
14	All Experiments Calendar	Approved	Useful	Medium	Low	Low	4th	Have an overall schedule viewer
15	Remove Experiments	Approved	Important	Low	Low	High	4th	Allow for workers or administers to remove schedules
16	Tracking of Consent Payment Forms	Rejected	Useful	Medium	Low	Medium	N/A	Allow for workers to check off participants when filling out consent/payment forms
17	User Report	Rejected	Useful	Medium	Low	Medium	N/A	Allow participants to have a report on new experiments
18	Accounts	Approved	Critical	High	Low	Medium	N/A	Accounts for participant
19	Prevent Scheduling Conflicts (Participant)	Approved	Useful	High	Low	Medium	N/A	Prevent participants from scheduling 2 experiments at the same time
20	Prevent Scheduling Conflicts (Administer)	Approved	Useful	High	Low	Medium	N/A	Prevent 2 rooms from being scheduled at the same time
21	Install Scripts	Proposed	Useful	High	Low	Low	TBD	Install scripts for installation
22	Documentation for Maintenance and User	Approved	Useful	High	High	Low	Ongoing	Documentation

The Browse Experiments feature and Persistent Experiment Storage both had a Priority of Critical since they both must be implemented for even a very basic version of the scheduling System. The effort on both was a medium as with a team of two, there would be a manageable amount of work. Browse Experiments has a stability of medium since it is up for change upon the client seeing the UI. Persistent Experiment Storage has a stability of high since once implemented has little chance of being changed.

Participant Schedule Experiment has a priority of critical since the participants must be able to sign themselves up for an experiment for the project to be successful. Again, the effort is medium since with two people the work would be manageable. The risk is high on this feature, since the success of the project has a dependence on the feature. The stability would be medium since the steps are unlikely to change, but the UI could easily change.

Levels of Authentication is a useful priority because it would not be necessary for there to be an actual Admin since the users trust each other, but this would be a nice feature. The effort and stability are medium since the feature may change some, but only smaller parts of the feature, while still being a very manageable task.

Filter Experiments has a priority of useful, since it might only apply to users in certain situations. Filer Experiments and Experiment Participants have a low risk, since the project does not depend on their success. They both also have a stability of high, since changes are unlikely to happen. The effort on Filter Experiments is medium, since there are some areas of the feature, such as what to filter by, that have not been established. The effort on Experiment Participants is low since a simple SQL query will do most of the job.

Cancel Experiment Appointment, Modify Experiment, and Add Experiment get and effort of Medium, since most deal with SQL and some logic on the back end. They also have a risk of medium, since a mistake while implementing these features could create a difficult to find bug elsewhere. The stability is medium, since parts of the database could change slightly.

Notify Participant When Creating Appointment, Notify Participant Appointment Reminder, and Notify Participant Appointment Cancellation Reminder all have a priority of useful since they would be nice to have, but are not vital to the projects success. They all have an effort of medium since they involve a persons email, but could be reduced to low, depending of the framework used. Their risk is low, since a failure here creates no problems else where in the project, nor does a mistake spread else where in the project. The stability is high on these since they are unlikely to change.

User Information Form has a priority of critical since the user must enter their information when scheduling for an experiment. The effort is low since this will be a simple UI, but the Risk is High since the User information must be stored for the experiment. The stability is high since all that is needed is name, phone number, and email.

Export Experiment Participant List is a useful feature, that has a high effort due to formatting of the report. The risk is low though, since the feature is not critical in the release of the product. Stability is high due to the feature being very specific.

An All Experiments Calendar would be useful for the future participants. The effort is medium because it would be an extension of the Browse Experiments feature.

Remove Experiments has a priority of Important, since, although rare, experiments maybe cancelled. The effort is low since most of it will be taken care with an SQL query. Also, a stability of high is given because of how specific the feature is described.

Accounts, Prevent Scheduling Confilicts for the Participant, and Prevent Scheduling Conflicts for the Administer all have a priority of useful, except for Accounts which Critical, since the other two features mentioned rely upon the Accounts along with cancellation of the experiment slots. The effort is high for all the features due to the logic needed when implementing the features.

Tracking of Constent Payment Forms and User Report have all be rejected, since the client does not need theses features.

Documentation and Install scripts are both useful priority. The effort will be high, since there is complexity associated with the Install scripts and Documentation is difficult to keep up to date. The stability would be low since the definition could change.

Items 1, 8, 15, 18, and 21 will be assigned to Kevin Risden. Items 3, 4, 10, 11, and 12 will be assigned to Samad Jawid. Items 6, 9,14 and 20 will be assigned to Chris Gropp. Items 2, 5, 7, 13 and 19 will be assigned to Trey Cahill. The entire team will work on item 22.

7 Use Cases

7.1 Experiments

1. Name: Authentication
   (a) Brief Description: User logs in, logs out, or creates an account.
   (b) Actors: User
   (c) Basic Flow:
      i. Step 1: User clicks the "login/logout/create account" button from any page.
         A. The system will record where the user entered this use case from, as well as any entered information. This is so the system might seamlessly return the user to what they were doing after authentication is complete.
         B. Alternate Flow: If the user is already logged in, the system will log them out. They are no longer considered authenticated and may not be able to view their previous page; if so, they will be shunted to the main page. Exit the use case here.
      ii. Step 2: User prompted for Email and Password. There is a button labeled "Create New Account" which will alter the interface, displaying instructions and revealing the fields Name, Phone, and Confirm Password. After completing these fields, user clicks "Submit" button.
         A. If the user didn't expand the "Create New Account" option, the system sends their login information to the database. If it matches, continue to step 3.
         B. Alternate Flow: If the database didn't recognize the email, return to step 2 and expand "Create New Account".
         C. Alternate Flow: If the password didn't match the one stored in the database, return to step 2 and display a message informing the user their password was incorrect.
      iii. If the "Create New Account" area was open, the system will create a new account.
         A. If the pasword and confirm password fields don't match, return to step 2 and display a message saying so. Keep all entered information except the password and confirm password fields.
         B. Query the database to see if it already has an account with that email. If it does, return to step 2 display a message saying so.
         C. Send an email to the address listed in the Email field. If this fails, return to step 2 and display a message saying so.
         D. Send all entered information to the database to be added. Continue to step 3.
      iv. Step 3: System displays a message confirming account creation or successful login. The user is now logged in. The user will be returned to the page they entered authentication from automatically after 10 seconds, or can click a link to do so immediately.

(d) Pre-conditions:

i. System is functional.

(e) Post-conditions:

i. User is logged in or out, the opposite of what they were before.

(f) Special Requirements:

i. N/A

(g) Feature Mapping:

i. Levels of Authentication

ii. Accounts

2. Name: Sign Up for Experiment

(a) Brief Description: Prospective participant chooses and signs up for an experiment.

(b) Actors: Participant (henceforth "user")

(c) Basic Flow: (User can return to a previous step with the "back" options on their browser.) Each page has a login/logout button, which will follow the use case "Authentication" if clicked.

i. Homepage: The system homepage has a button for login, and a table displaying currently running experiments. Each experiment on the table can be clicked for further details.

ii. Step 1: User clicks on an experiment button. This will navigate them to that experiment's page.

A. Possible substep: The experiment listing table has options for sorting and filtering. The user may click these buttons and enter filters without altering flow.

iii. Experiment Page: Each experiment has a webpage with its name, description, and a list of timeslots. Each timeslot can be clicked to move to the confimation screen. There is also a button to join the experiment which will send the user to the confirmation screen without a timeslot selected.

iv. Step 2: User clicks on a timeslot or the join button. The system will navigate them to the confirmation page.

A. Possible substep: The timeslot listing has options for sorting and filtering. The user may click these buttons and enter filters without altering flow.

v. Step 3: If the system attempts to move to the confirmation page but the user is not authenticated, it will follow the use case "Authentication" first. Once they have successfully logged in, they will be sent directly to the confirmation page.

vi. Confirmation Page: This page can only be accessed while logged in. It displays the experiment name, required qualifications, and a checkbox certifying that the user meets these requirements. The timeslot list from the experiment page is also on this page, with the timeslot remaining selected if the user did so in step 2. There is a large "Confirm Appointment" button at the bottom of the page.

vii. Step 4: The user checks the box, selects a timeslot if they have not already, and clicks the "Confirm Appointment" button.

viii. Final actions: The system will send the appointment data to the database and return the user to the homepage, with a popup confirming their successful registration. The system will also send an email with the experiment and timeslot information to the account they used to register.

(d) Alternate Flows:

i. User logs out while on the confirmation page.

A. As a user cannot return to this page while logged out (the normal behavior for the Authentication use case), they will be shunted back to the experiment page. Information provided in step 2 is not guaranteed to remain selected.

ii. Database reports an error when receiving appointment data.

A. Unlikely to occur outside of concurrent modification (two people signing up at the same time), the system will shunt the user back to the confirmation page with the updated available timeslots and provide a popup explaining why this happened.

(e) Pre-conditions:

i. System is functional.

(f) Post-conditions:

i. Database has added appointment to user and experiment data.

ii. (Side effect) User is authenticated.

(g) Special Requirements:

i. N/A

3. Name: Add Experiment

(a) Brief Description: Experiments can be created by Administrators and Researchers

(b) Actors: Administrators and Researchers

(c) Basic Flow: (user can cancel at any time and follow Alt Flow 1)

i. User must click on Add New Experiment link from the Administration "home" page

ii. System will display a screen with text boxes to enter experiment name, description, and qualifications, multiple date/time choosers for the schedule times, and a drop down list to specify the length of the schedule slots

iii. User must enter the experiment information for name, description, qualifications, and schedule slots

iv. User can then begin setting up the schedule times by choosing date, begin, and end time for each slot they want to run the experiment

v. User then must save the experiment by clicking the Save button

vi. System will then save the experiment to persistent storage and provide the user with confirmation that the experiment was created successfully and redirect user to all experiment view

(d) Alternate Flows:

i. User cancels out of creating an experiment

ii. Saving an experiment fails

(e) Pre-conditions:

i. User is an Administrator and/or a Researcher and has authenticated

(f) Post-conditions:

i. System will have recorded the experiment or the system will notify the user why the creation of the experiment failed

(g) Special Requirements:

i. N/A

4. Name: Modify Experiment (user can cancel at any time and follow Alt Flow 1)

(a) Brief Description: Experiments can be modified by Administrators and Researchers to change all assets of the experiment

(b) Actors: Administrators and Researchers

(c) Basic Flow:

i. System will display experiment fields (name, description, qualifications, schedule time, schedule slots, and participant list)

ii. User will click on desired field to modify [Alt Flow 3]

iii. System will allow field that user choose to be editable in line

iv. User will then change field as desired and click out or save when finished

v. System will update the database with the modified experiment information

(d) Alternate Flows:

i. User cancels out of creating an experiment. System will return user to the page where user came from

ii. Saving an experiment fails

iii. User deletes an experiment. System will remove experiment from database after user confirmation and display a message to the user indicating this was successful

(e) Pre-conditions:

i. User is an Administrator and/or a Researcher and has authenticated

ii. User choose experiment through one of the experiment views

(f) Post-conditions:

i. System will have recorded the modifications to the experiment or the system will notify the user why the modification of the experiment failed

(g) Special Requirements:

i. N/A

5. Name: List Experiment Participants

(a) Brief description: Researcher logs in and views a list of all participants for a selected experiment.

(b) Actors: Researcher

(c) Basic flow

i. Researcher logs in

ii. System displays table of researcher's experiments

iii. Researcher selects experiment from table

iv. System displays list of all participants for selected experiment

(d) Alternate flows

i. Researcher does not own any experiments

A. (2) displays an empty table

Name	Phone number	Email address	Date/time
First Last	(123) 123-4567	first.last@example.com	2011-9-1 13:00
First Last	(123) 123-4567	first.last@example.com	2011-9-1 13:00
First Last	(123) 123-4567	first.last@example.com	2011-9-1 13:00
First Last	(123) 123-4567	first.last@example.com	2011-9-1 13:00
First Last	(123) 123-4567	first.last@example.com	2011-9-1 13:00

B. He cannot proceed past (2) until he creates an experiment or is added to another researcher's

ii. Selected experiment has no participants

A. (4) displays an empty table

B. Nothing is displayed in (4) until a participant signs up for the selected experiment

(e) Pre-conditions

i. System is running

ii. System is in ready state

iii. Researcher has account with correct permissions/groups

(f) Post-conditions

i. Researcher knows who is signed up to participate in his selected experiment or there are no experiments/participants

(g) Special Requirements:

i. N/A

6. Name: Cancel Experiment Appointment

(a) Brief description: Participant logs in and cancels an appointment.

(b) Actors: Participant (User)

(c) Basic flow

i. Participant logs in

ii. System displays table of participant's appointments

iii. Participant selects appointment from table

iv. System displays details for selected appointment

v. Participant selects delete

vi. System displays confirmation prompt

vii. Participant selects confirm: appointment is deleted and system returns to (2) with an affirmation message

viii. Participant selects cancel: system returns to (5)

(d) Alternate flows

i. Participant has no appointments

A. (2) displays an empty table

B. He cannot proceed past (2) until he signs up for an experiment

(e) Pre-conditions

i. System is running

ii. System is in ready state

iii. Participant has account

(f) Post-conditions

i. Participant deleted selected appointment or participant cancelled operation

ii. Researcher(s) owning said appointment's experiment are notified via email

(g) Special Requirements:

i. N/A

7. Name: Report Experiment Participant Lists

(a) Brief Description: When the user is a researcher, the user will be able to export a csv file, filed with the Experiment name and participant and times.

(b) Actors: Researcher

(c) Basic Flow:

i. The researcher will check what experiments to export to the csv file from the list of experiments in the researcher side view

ii. The researcher will click Export to CSV

iii. The system will generate a csv file from the selected experiment displaying the name of the experiment and the names of participants with their times

iv. The system will then start the download of the file to the researchers computer

v. When the system has completed 3 and 4, the system will display a message box Export Complete!

vi. The researcher will click OK or the exit button on the message box

vii. The system will be back in the researcher side view.

(d) Alternative Flow:

i. The researcher did not select any experiment. An error window will appear.

ii. The system encounters an error when pulling data from the database. An error window will appear

iii. The system encounters any error when creating the csv file. An error window will appear

iv. The system cannot download the file to the researchers computer. An error window will appear

v. The researcher will deny the download of the csv. A message box will appear

vi. The user exits the browser

(e) Preconditions:

i. The researcher must be logged in as a researcher

ii. The system is in the researcher side view

iii. The researcher must already have experiments scheduled

(f) Postconditions:

i. The system is back in the researcher side view

(g) Feature mapping:

i. Export Experiment Participant List

8. Name: Calendar/List of All Experiments

(a) Basic Description: The list will show the soonest to be conducted experiments (10) and will allow for a user to click and view more information on the experiment

(b) Basic Flow:

i. The system loads the first 10 experiments

ii. The user can scroll down the list

iii. The user selects an experiment

(c) Alternative Flow:

i. The system errors when loading the first 10 experiments from the database

ii. The user goes to another website

iii. The user attempts to log into the researcher page and fails

iv. The user attempts to log into the researcher page and has success

v. The user clicks on the next 10 experiments and repeats basic flow 1 to 3

vi. The user exits the browser

(d) Precondition:

i. The user is on the web page

(e) Postcondidtion:

i. The system is showing an experiment or the browser is on a new page

(f) Feature mapping:

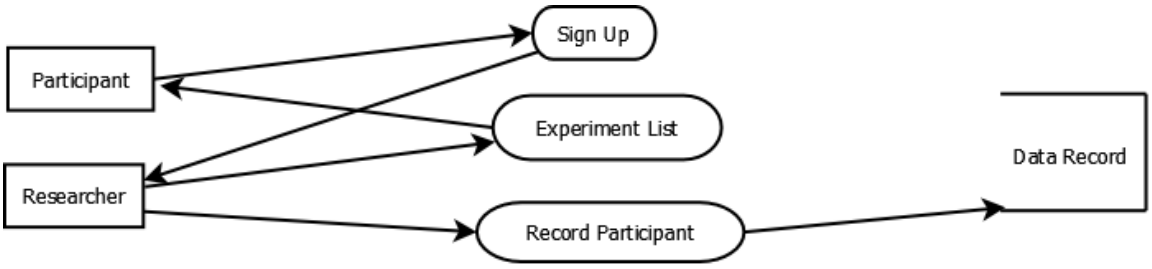
i. All calendar Experiments

ii. Browse Experiments

iii. Persistent Experiment Storage

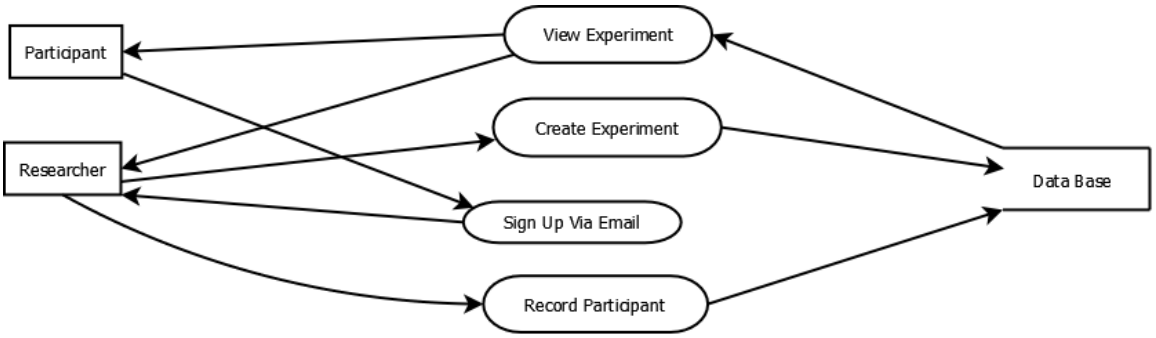
8 Data Flow Diagrams

8.1 Current System Context Diagram



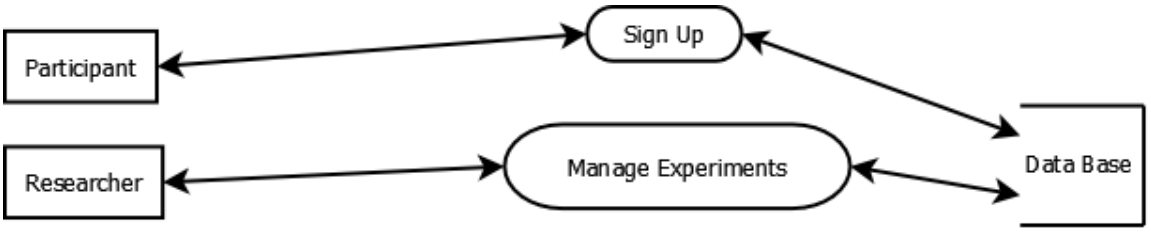
The current system requires the researcher to post an experiment (Experiment List), with the Participant then seeing the experiment from the experiment list and signing up. The researcher then gets the participants information and records the participant information into their current data record system.

8.2 Current System Level 0 Data Flow Diagram



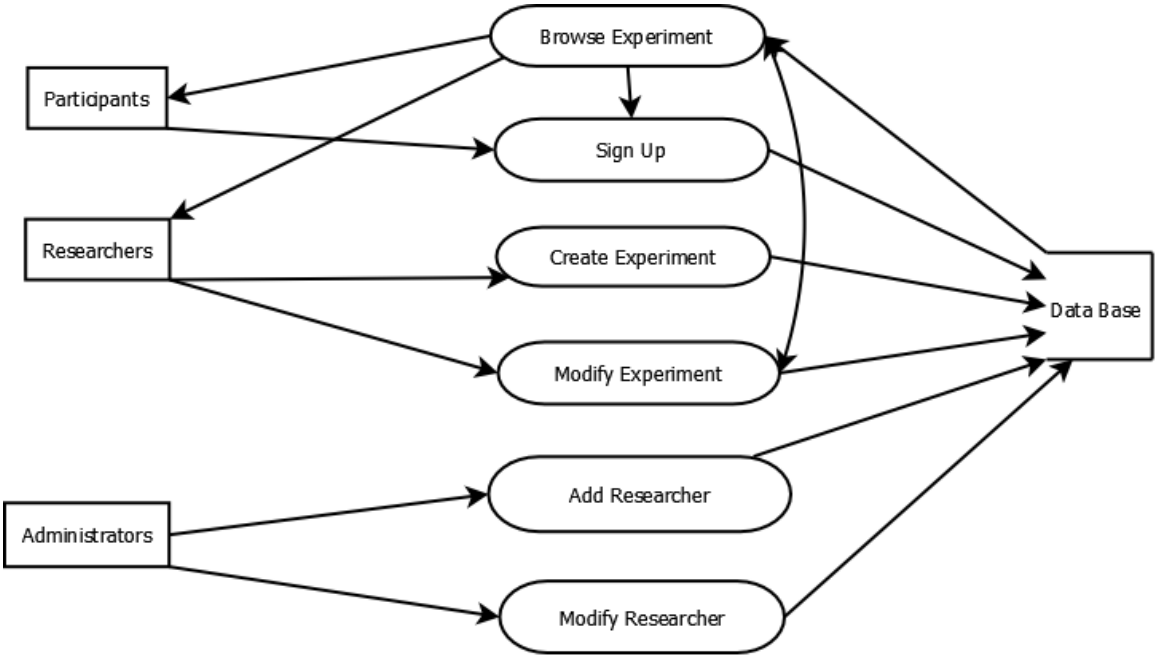
In the level zero data flow diagram, the flow remains the same, but the participant is signing up via email, while the researcher must explicitly create the experiment to be put on the data base (for this definition of data base the team is including a peg board of information or a campus email or similar types of information distribution).

8.3 Proposed System Context Diagram



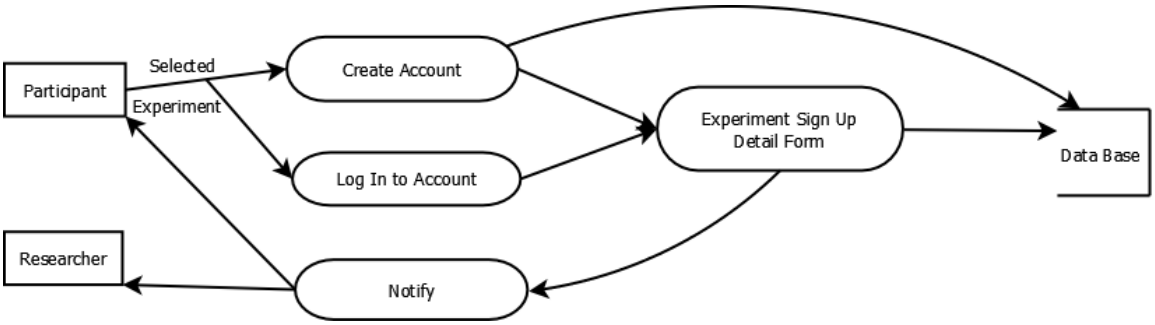
In the proposed system, the researcher will manage experiments via an computer data base. This same data base will let the user sign up and keep their information for the experiment.

8.4 Proposed System Level 0 Data Flow Diagram



Now, in the level 0 diagram, creating and experiment and modifiying an experiment are explicitly different, but both have data being stored on the computer data base. The participant now browses the experiment and then signs up, instead of the two being combined. The data entered by the participant will be saved on the data base. Another actor, the administrator, will have the ability to add a researcher or modifiy a researcher. All of the data about the researchers will be placed on the computer data base.

8.5 Proposed System Level 1 Data Flow Diagram for Sign up

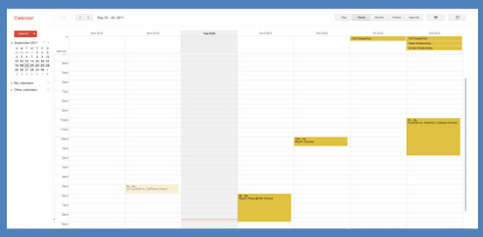


When signing up the participant has selected a specific experiment to begin with. Then, the participant creates an account, with the information being stored in the computer data base, or logs into their account. Next the participant will select a time and confirm that they conform to the qualifications of the experiment. This information will be stored in the computer data base. After finishing sign up, the researcher and participant will be notified via email of the new participant.

9 Storyboard

Human Computer Interaction Labs

Browse Experiments



Researcher Log In

Sign up Information for Experiment X

Name:
Cell Phone:
Email:

☐ By clicking this I have agreed that I meet all qualifications

Submit

Experiment X

- Qualification #1
- Qualification #2
- Qualification #3
- Qualification #4
- Qualification #5

Notes:

Dates
- Date 1
  - Time1
  - Time2
  - Time3
- Date 2
  - Time1
  - Time2
  - Time3

This Experiment is ran by <Researcher1, Researcher2,…,Researchern>

Sign up for this Experiment

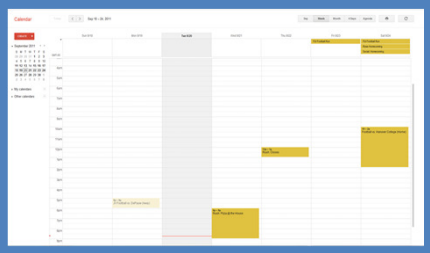
Researcher Log In

Username:
Password:

Log In

Experiments of <Username>

Filters
- Date
- Room
- Etc
- Etc
- …



Create New Experiment

Experiments of <Username>

Experiment	Date	Room
Robot Stuff	5/5	F 217
UI Stuff	5/6	F 322
10001010110	5/10	O 231

Create New Experiment

Experiment X

Save

Qualifications:
- Age
- Language
- Etc
- etc

Description/Notes

Participant List

Name	Cell Phone	Date/Time	Email
Bill Bob	(555)555555	5/5 – 14:00	bob@ma.com
Sparks Mcgee	(403)6185546	5/6 – 13:30	sparks@ma.com
Shane Jermey	(209)4723882	5/5 – 13:30	jermy@ma.com

Human Computer Interaction Labs

Browse Experiments

Experiment Name	Date – Times	Notes
Robot Stuff	5/5 – 13:00	Etf
Less Robot Stuff	5/5 – 14:00	Etc
More Robot Stuff	5/6 – 12:00	Etc
UI	5/6 – 9:00	Etc
Dan Guiney	5/8 – 20:00	Etc
Computer Runs	5/9 – 15:00	etc

Researcher Log In

10 References

[1] University of Wisconsin-Madison. Human-Computer Interaction Laboratory, 2010.

11 Appendix

12 Glossary

13 Index