

# Mann Library SmartMap

---

## Requirements Analysis and Preliminary Design

### **The Client**

Howard Raskin  
[hbr1@cornell.edu](mailto:hbr1@cornell.edu)

### **Team Lasera**

Amanda Li  
Sara Lin  
Evan Sperling  
Andrew Tourtellot  
Lisa Wang  
Robert Xiao

# Table of Contents

- Requirements..... 3
  - General Overview ..... 3
    - Purpose and Scope..... 3
    - Objectives and Goals..... 3
  - Current Systems ..... 3
  - Proposed System Requirements..... 4
    - Overview ..... 4
    - Functional Requirements..... 4
    - Usability Requirements..... 5
    - Non-functional Requirements ..... 5
  - Requirements Models..... 6
    - Use Case Diagrams ..... 6
    - Use Case Specifications..... 7
    - Scenario 1 – Student Browsing Library Map ..... 14
    - Scenario 2 – Student Searching for Animal Psychology 2<sup>nd</sup> Edition ..... 14
    - Scenario 3 – Mann Librarian Changing Exhibit Information ..... 15
    - Data Models Simplified ER Diagram ..... 16
- Preliminary Design ..... 19
  - User interface..... 19
    - Window ..... 19
    - Header..... 19
    - Map ..... 20
    - Tabs ..... 21
    - Footer..... 21
    - Mockup Design ..... 21
- System Design ..... 23
  - Model View Controller..... 23
  - Database Design..... 24

# Requirements

## General Overview

### Purpose and Scope

The purpose of this new software system is to provide an interactive guide to the newly expanded Mann Library. The expansion project will greatly increase the size of the library and will also add many new features such as cafes and public exhibits. The library will also provide numerous public computing centers as well as private study rooms and group presentation rooms. Patrons of the library will be greatly assisted by a system that they can access off of the Mann Library website that will allow them to search for specific rooms and to learn about the events happening in the library.

The interactive map being designed will encompass all of Mann Library. There will be detailed interactive maps of each floor of the library. The system will also provide locations of extra features such as bathrooms and elevators so patrons can determine which rooms or locations would be most convenient for them. In addition, pictures of study rooms will be provided so that patrons can decide to choose rooms based on the view from that area. At this time, the map will not interact with existing Mann Library databases. As a result, there will not be a feature allowing users to search the library catalog nor will there be a feature allowing users to check on availability of library staff.

### Objectives and Goals

The main objective of this project is to provide patrons with an online interactive guide to successfully navigate the new Mann Library. This will help patrons get acclimated to the new building and can also foster visibility for the library by providing details of public events being held. Secondary objectives for this utility are to provide library staff with extensible and well documented code to allow for easy implementation of additional features and to provide the library staff with a comprehensive administration tool to allow for editing of table content and map data without having to dig through the source code.

Success of this project will be dependent on two things: The first is an easy to use map navigation system. Since this is mainly for patrons unfamiliar with the library, the interface should be such that it is intuitive for them to use. The second is that the map should be able to provide all necessary information to the user in a manner such that it is not overwhelming. Well organized text as well as use of pop up boxes will provide users with a quick overall look at the library and then allow them to go into more depth with areas that they are interested in.

## Current Systems

There is currently no interactive map system available at Mann Library. There is a site on the library website that provides patrons with static images of the library layout. However, it does not provide patrons with any more details of the room.

## Proposed System Requirements

### Overview

There are two key pieces of SmartMap which must seamlessly interact with one another in order to present a cohesive, robust user experience. The first is the user interface, which consists of the interactive map and various methods of manipulating it. The second is the metadata about each room or facility, which is used to display relevant information to the user.

Because SmartMap will be web-based, it must be built with accessibility in mind. In addition, we must ensure that its underlying technologies are compatible with the library's computer system.

### Functional Requirements

The system must be able to perform the following functions:

- **User Interface**
  - Zoom levels                      Users must be able to view the map at both building and floor levels.
  - Discrete rooms                 The map must clearly outline each room.
  - Labels                             Rooms must be labeled by name and/or type.
- **Functionality**
  - Information retrieval         Users should be able to view information about each particular room or facility, including upcoming activities and events, as described in the data section below.
  - Call number search             Users must be able to search for books by call number. The map will highlight the location of the stack holding the requested book.
  - Room search                     Users must be able to search for and locate rooms based on certain preferences, such as seating capacity.
  - Admin functions                Map administrators must be provided with an interface that allows them to update metadata for all rooms and facilities, as well as the items on the map itself.
- **Data**

There are 32 different items, which can be a room, an area or something in a room such as a printer. All of these tables share a set of standard fields which include ID, name, item type, coordinates, floor, Mann web site URL, additional URLs, images, description, and collaborative space, some of which may be empty. In addition, some items have fields that are unique. (See table below) For example, the café item has the additional fields menu URL, events URL and Hours URL.

Item Type	Extra Fields
Bailey Hortorium	<i>None</i>
Bird Watching Station	<i>None</i>
Bissett Collaborative Area	<i>None</i>
Café	menu URL, events URL, hours URL
Classroom	Class Schedule URL, Computer Type, no. of computers, software, software URL, computer URL, seating capacity, hours URL, reserve URL
Drinking Fountain	<i>None</i>
Exhibit Case	exhibit ID
Lockers	locker type
Outdoor Patio	events URL
Photocopier	<i>None</i>
Printer	color, duplex, format
Public Computing	<i>None</i>
Rest Room	Gender
Seminar Room	seating capacity, schedule URL
Service Desk	hours URL, goHereFor, phone number, email address,
Stacks	start call num, end call num, collection type
Staff Offices	departments, contact name, contact phone number, contact email
Stairwell/Elevator	take me to floors, take me to POIs,
Student Expo Space	schedule URL, exhibitID
Study Area	soft seats, grad student
Study Room	Room #, study room type, view type, projector, BIB ID/Item ID

## Usability Requirements

### Accessibility

The technologies used to build the map must be supported by a typical user's web browser (a typical user is assumed to be a college student or faculty member whose computer is relatively up-to-date).

## Non-functional Requirements

### Compatibility

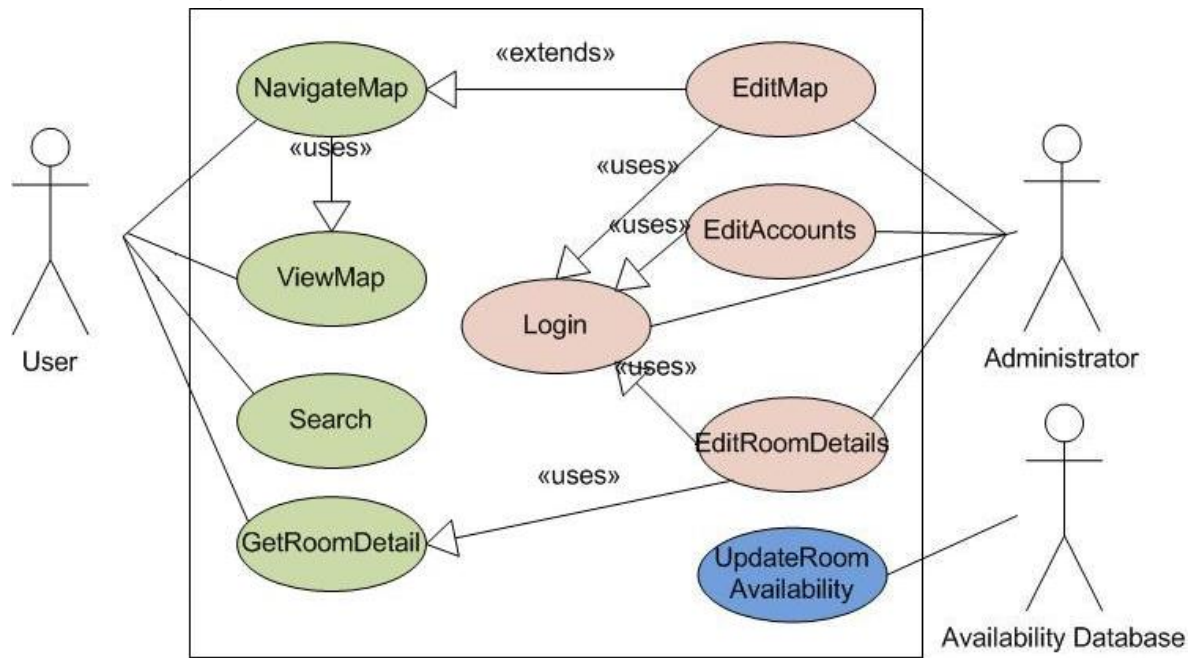
The application must function correctly when ported to the library's servers.

### Extensibility

SmartMap must be extensible so as to allow administrators to add features not available in the initial release.

## Requirements Models

### Use Case Diagrams



## Use Case Specifications

### ***NavigateMap***

**Summary** The user performs basic navigation of the library map, including zooming, scrolling, and map item identification.

**Preconditions** The SmartMap window must be open in a browser meeting the system requirements. The map itself must be displayed.

**Trigger** Certain types of mouse interaction with the map display or map controls.

#### **Course of events**

1. SmartMap retrieves and displays the initial zoomed-out map.
2. SmartMap begins an iteration of the user interaction loop. The user initiates any of the following actions:
  - a. Zoom: The user selects a "zoom in" or "zoom out" option.
  - b. Scroll: The user instructs the map to scroll its display via a selectable option or a dragging action with the mouse.
  - c. Identify: The user requests the identity of a map item.
3. SmartMap responds to the user request. The following responses correspond to the actions above:
  - a. The system fetches the map data required to display the new floor, changes the display parameters to correspond to the new zoom level, and redraws the map.
  - b. The system changes the display parameters so the map is shifted according to the scroll action performed, and redraws the map.
  - c. The system displays an unobtrusive label for the item chosen by the user that identifies it and possibly includes other useful, concise information. The label does not interfere with further map viewing and navigation.
4. The user continues interacting with the map.
  - a. Return to step 2.
  - b. Return to step 2.
  - c. When the user de-selects the map item, the system removes the label for it from the display. Return to step 2.

#### **Alternate paths**

##### *Map display limits*

Depending on the current state of the map, the map may not respond as described in step 3 above:

3. SmartMap response:
  - a. If the map is at maximum or minimum zoom and a "zoom in" or "zoom out", respectively, was requested, then the map will do nothing.
  - b. If the requested display of the map goes beyond the maximum scroll range of the map, the SmartMap will adjust the display so it is bounded by the scroll range.

## ***ViewMap***

**Summary** The user looks at the map display.

**Trigger** The user opens the SmartMap window in a browser meeting the system requirements.  
Or, the system redraws the map in response to a user request.

### **Course of events**

1. SmartMap fetches necessary map data and images from library server.
2. SmartMap displays either floor map or building map on screen. The building map is displayed by default.

Floor map includes:

- a. Rooms and other items, with items falling into the same category or function distinguishable.
- b. Outline of hallways and rooms.
- c. Labels for rooms and items with a display area large enough for such a label.
- d. Controls for map manipulation.
- e. (Under consideration) Highlighted color for the item that is selected, if any.

Full-zoom building map map includes:

- a. Small versions (thumbnails) of each floor.
  - b. Labels for each floor.
  - c. Controls for map manipulation.
  - d. (Under consideration) Highlighted colors/image for the floor that is selected, if any.
3. The user views the map as desired.



## **Search**

- Summary** The user searches for a book by call number or a room by various criteria.
- Preconditions** The SmartMap window must be open in a browser meeting the system requirements.
- Trigger** The user selects the search option or inputs into a search field on the main SmartMap display.

### **Course of events**

1. SmartMap displays a search field allowing text entry.
2. The user enters search terms into the field and confirms the search.
3. SmartMap determines if the search is for a book call number, otherwise it treats the query as a room search.
4. SmartMap generates search results from the map item database. If the query was a call number the stacks are searched, otherwise all map items are searched.
5. SmartMap displays search results with relevant information: Item description, search relevance.
6. User selects search result to jump to.
7. SmartMap displays the floor map, centered on the selected item.

### **Alternate paths**

#### *No results*

- 3b. SmartMap does not find any search results.
- 4b. SmartMap displays a message indicating there are no results and options to refine the search or return to the previous display.
- 5b. User selects the "refine search" option; return to 2.

#### *Cancel search*

This could occur in place of any of steps 2, 6, or 5b.

- 1c. User cancels search or closes search display.
- 2c. SmartMap displays the map shown before the user initiated Search.

## ***GetRoomDetail***

**Summary** The user views more information on a selected map item.

**Preconditions** The SmartMap window must be open in a browser meeting the system requirements.  
The map itself must be displayed.

**Trigger** Certain types of interaction with a map item or map controls.

### **Course of events**

1. The user selects a room or other map item about which to get more information.
2. SmartMap retrieves the room information from the server.
3. SmartMap shows room information in a new display section. Information displayed may include:
  - a. Images, possibly thumbnails.
  - b. Item name and description.
  - c. Item availability.
  - d. Additional item information depending on the item type.
  - e. External link(s) relevant to the item.
4. The user closes the information display section.

### **Alternate paths**

#### *Text scrolling*

4b. Depending on the amount of item information, the user may scroll through the text. Continue to 4.

#### *Thumbnail expansion*

4c. The user selects a thumbnail to expand.

5c. SmartMap displays the full image corresponding to the thumbnail.

6c. The user closes the image. Return to 3.

## ***Login***

**Summary** An administrator securely logs into the system.

**Preconditions** The SmartMap window must be open in a browser meeting the system requirements.

**Trigger** The administrator selects the login option.

### **Course of events**

1. SmartMap prompts the administrator for a username and password.
2. The administrator enters said information.
3. SmartMap checks the information against its database of accounts.
4. If the information matches, continue to 5. If it does not, return to 1.
5. SmartMap sets the system to administrator mode and redisplay the same screen as before login but with administrator options. Options displayed in administrator mode are:
  - a. An edit option added to the main map window.
  - b. An edit option to each item information window.
  - c. An option to edit administrator accounts.
  - d. The login option changes to a logout option.
6. The administrator may navigate the system and make editing changes as desired before continuing.
7. The administrator selects logout.
8. SmartMap disables administrator mode and removes administrator options from screen.

## **EditMap**

**Summary** An administrator changes layout of map items.

**Preconditions** The SmartMap window must be open in a browser meeting the system requirements.  
The map itself must be displayed.

**Trigger** The administrator logs in and selects edit mode.

### **Course of events**

1. SmartMap displays editing controls and locks out other administrators from editing the map.

The path branches here:

#### *A. Editing an existing item.*

2. The administrator selects an item to edit.
3. SmartMap displays the item as highlighted (possibly by a color change or special display of edges or vertices).
4. The administrator changes the location or shape of the selected item, or selects the delete option.

#### *B. Adding a new item.*

2. The administrator selects the add new item option.
3. The administrator defines the new item's shape and location.
4. SmartMap prompts the administrator for the new item's name and description.

The branches merge back together here:

5. Return to 2 if there are more changes, or continue to 6.
6. The administrator selects a preview option.
7. SmartMap displays map as it would appear to a user viewing the map with all the edits made.
8. The administrator selects to return to editing (return to 1) or save changes (continue to 9).
9. SmartMap saves the changed data to the server and unlocks editing for other administrators.
10. SmartMap displays the new map to the screen. The system is still in administrator mode, but not map edit mode.

### **Alternate paths**

#### *Cancel edit*

This branch could occur in place of any step where the administrator gives any input

- 1b. The administrator selects a cancel editing option
- 2b. SmartMap displays the old map. It exits edit mode but remains in administrator mode.

#### *Editing in progress*

- 1c. If another administrator has a lock on map editing, SmartMap displays a message to that effect and does not enter edit mode.

## **EditAccounts**

**Summary** An administrator makes changes to the list of administrator accounts.

**Preconditions** The SmartMap window must be open in a browser meeting the system requirements.

**Trigger** The administrator logs in and selects the option to edit accounts.

### **Course of events**

1. SmartMap retrieves a list of administrator accounts from the server.
2. SmartMap displays the login and real name of each account.
3. The administrator selects an account to edit or selects to create a new account.
4. SmartMap acquires a lock on the account.
5. SmartMap displays a full listing and options for the account:
  - a. Login name
  - b. Real name
  - c. Reset or set password option
  - d. (Under consideration) Permissions options
  - e. Delete account option
  - f. Save changes option
6. The administrator makes changes to the account and saves changes.
7. SmartMap saves changes to the server and releases the account lock.
8. Return to 2.

### **Alternate paths**

#### *Editing in progress*

- 4b. If the selected account is in use (unless by the administrator that is logged in) or is being edited, return to 2.

#### *Deleting account*

- 6c. The administrator selects the delete account option.
- 7c. If the administrator is not editing his own account, SmartMap prompts for a delete confirmation; otherwise return to 5.
- 8c. The administrator selects confirm or cancel.
- 9c. If the deletion was confirmed, SmartMap saves changes to the account listing; return to 2. If it was cancelled, return to 5.

#### *Cancelling account changes*

- 6d. The administrator selects cancel, possibly after making changes.
- 7d. SmartMap releases the account lock; return to 2.

#### *Cancelling account editing*

- 3e. The administrator closes the account edit display.
- 4e. SmartMap returns the display to the main map.

## ***EditRoomDetails***

**Summary** An administrator edits details on a particular map item or room.

**Preconditions** The SmartMap window must be open in a browser meeting the system requirements.  
The map itself must be displayed.

**Trigger** The administrator logs in, gets room details on a particular map item, and selects edit in the room detail window.

### **Course of events**

1. SmartMap acquires a lock on the room.
2. SmartMap displays item information with all text information displays changes to entry fields. It adds
3. entry fields allowing picture file selection in the picture area.
4. The administrator makes changes to the room information.
5. The administrator selects preview (continue to 5) or cancel (continue to 8).
6. SmartMap displays item information without editing fields but using the new information.
7. The administrator selects confirm (continue to 7) or continue editing (return to 2).
8. SmartMap saves changes to the server.
9. SmartMap releases the room edit lock.
10. SmartMap returns to the room information, exiting editing mode.

### **Alternate paths**

#### *Editing in progress*

- 1b. If the room is locked for editing, SmartMap displays a message to that effect.
- 2b. The administrator selects OK. Skip to 9.

## ***UpdateRoomAvailability***

**Summary** The room availability database provides periodic updates to the map rooms that store availability.

**Preconditions** The library database containing room availability is available for the SmartMap system to connect to.

**Trigger** A periodic timer has been activated.

### **Course of events**

1. SmartMap establishes a connection to the availability database.
2. SmartMap copies room availability to its own database for each room to which it applies.
3. SmartMap closes the connection.

### **Alternate paths**

#### *Room data missing*

- 2b. If the availability database does not contain data for a particular room SmartMap attempts to update, that room's availability is marked as unknown.

## **Scenario 1 – Student Browsing Library Map**

**Individual** Sara Lin, senior at Cornell, major in computer science, location Duffield Hall

**Equipment** Toshiba laptop with Firefox 2.0 browser, JavaScript enabled

### **Scenario**

1. SL turns on laptop and opens browser
2. SL types the URL of Mann Library and clicks on the “SmartMap” link on the library homepage
3. SmartMap application is displayed with a floor plan overview of Mann Library
4. SL chooses the first floor; the floor plan is expanded to full map size
5. First floor shows rooms and facilities
6. Sara wants to see the Café on the lower right corner, she drags the map upward to center on the Café
7. SL maximizes the map to see an enlarged version
8. SL clicks on the Café to see more room specific details
9. An information area appears with the detailed room information
10. SL reads the hours of the Café
11. Finished, SL closes the SmartMap application

## **Scenario 2 – Student Searching for Animal Psychology 2<sup>nd</sup> Edition**

**Individual** Danielle Dubois, freshman at Cornell, major in animal science, location Mann Library

**Equipment** Library desktop in CIT Computing Lab in Mann

### **Scenario**

1. DD opens IE7 browser
2. Mann Library homepage appears
3. DD clicks on the SmartMap link on the homepage
4. SmartMap application is displayed with a floor plan overview of Mann Library
5. DD goes back to Mann Library homepage
6. DD finds the call number of Animal Psychology 2<sup>nd</sup> Edition using Library Gateway
7. DD copies the call number and pastes it on the search input area on the SmartMap application
8. An information area appears with the result stack location
9. DD selects the location
10. The map changes to show the selected location
11. DD recognizes that this is the correct location, records the location
12. DD leaves to find her Psychology book

### **Scenario 3 – Mann Librarian Changing Exhibit Information**

**Individual** Emily Carr, Mann Library Librarian, location Man Library

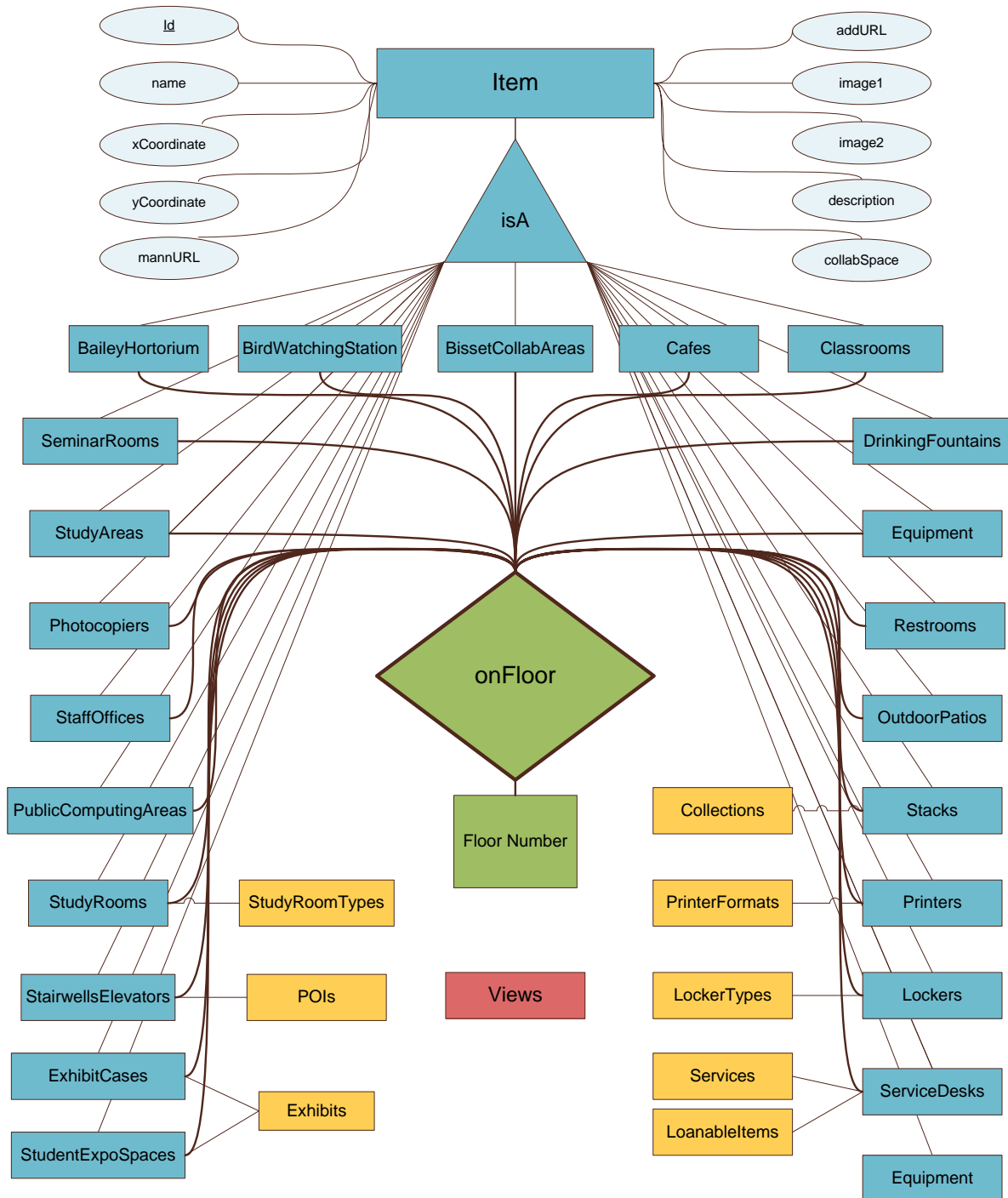
**Equipment** Library desktop in Admin Office in Mann

#### **Scenario**

1. EC opens her IE7 browser
2. Mann Library homepage appears
3. EC clicks on the SmartMap link on the homepage
4. SmartMap application is displayed with a floor plan overview of Mann Library
5. EC clicks on Admin link on SmartMap
6. Admin sign in form appears
7. EC types her username and password, the database authenticates her
8. SmartMap changes to Admin mode
9. EC selects the first floor plan in the floor overview map
10. First floor expands
11. EC selects the exhibition button
12. Exhibition information area shows the current information
13. EC selects to edit the exhibition information
14. Now each entry in the tab becomes an editable input area
15. EC changes the name, date, and description of the new exhibition and uploads two new images
16. EC submits the changes
17. EC receives a preview of the changes
18. EC confirms the changes
19. The new information is submitted to the database
20. EC signs out and closes the application

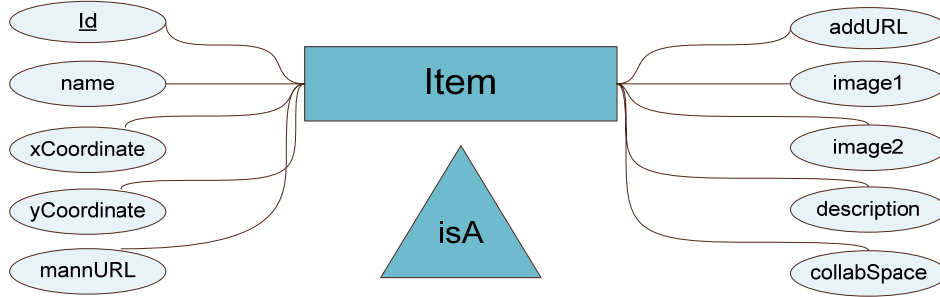
# Data Models

## Simplified ER Diagram

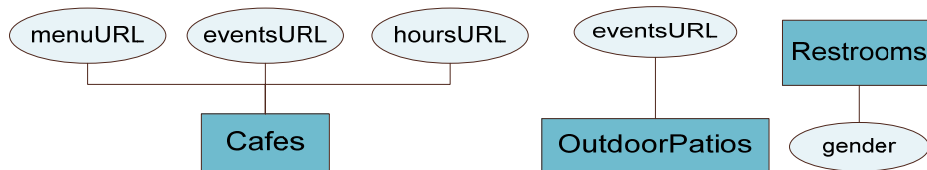




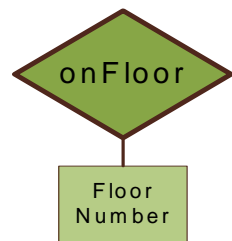
## Breakdown of Simplified ER Diagram



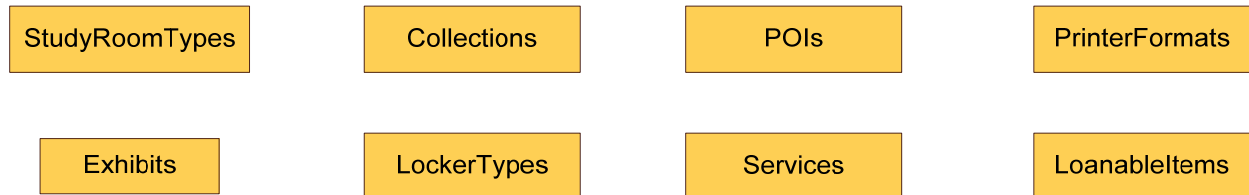
An Item is an object that can be visible on the SmartMap. Each Item will have a set of core characteristics, such as X and Y coordinates, a name, and a description.



There will be many subclasses of Items: Café's OutdoorPatios, etc. Some of these subclasses will have addition attributes unique to that object, such as *gender* for a Restroom or an *hoursURL* for a Café.



Each Item will be mapped to a floor. The only exceptions to this are Stairwells and Elevators, which will be mapped to all the floors they span.

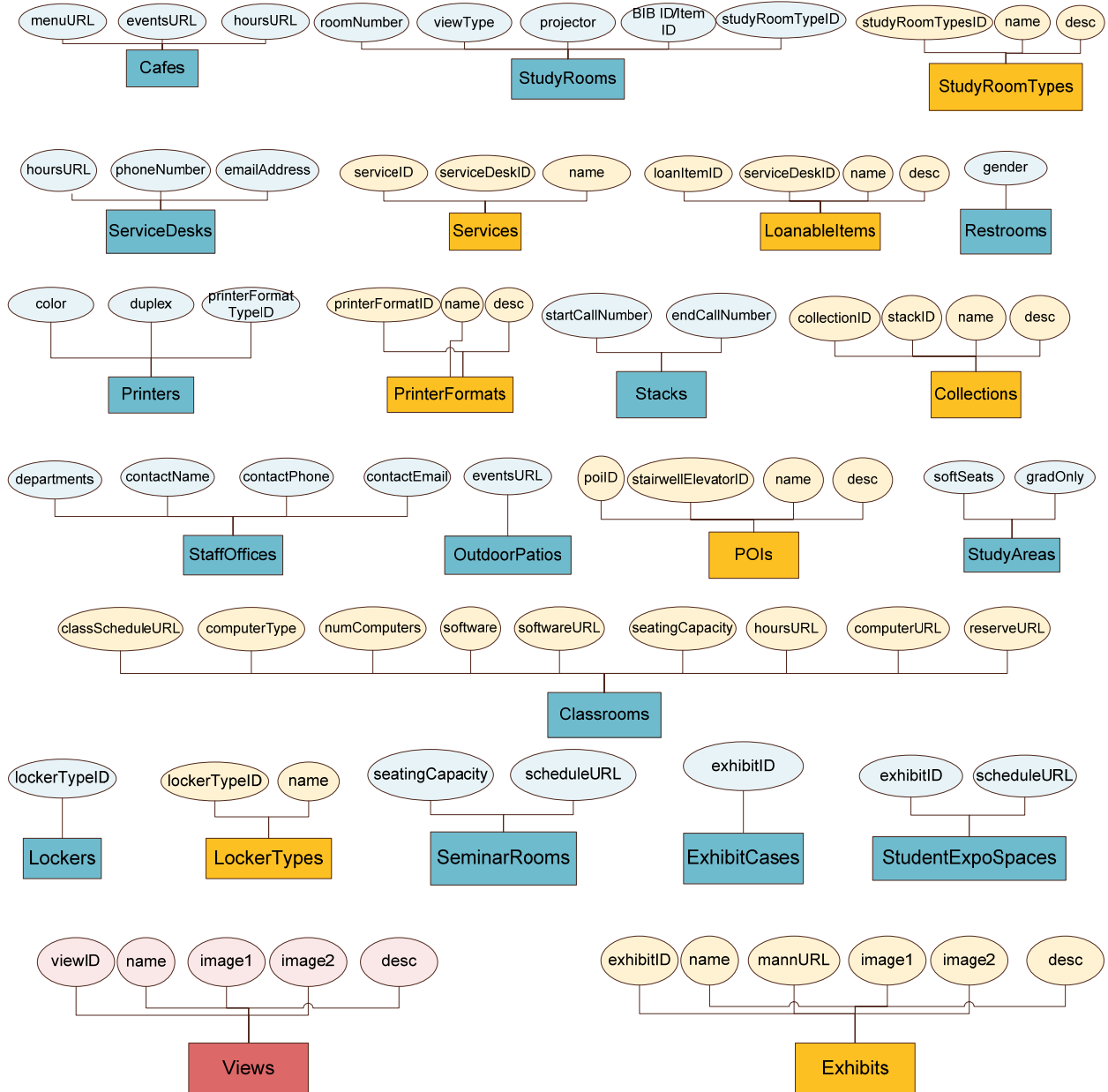


There are several types of entities which will not be directly visible on the SmartMap. They are shown on the large ER Diagram as orange. These entities will most likely be used or linked to by other Items or entities. For example, an Exhibit will not show up on the SmartMap but the ExhibitCase or StudentExpoSpace that houses the exhibit will be visible and have more information about the Exhibit.

### Views

A View will be the type of view available from a room. For example, a StudyRoom could have a View of the woods or the parking lot. These Views will be available to those Items that are literally rooms that have a view.

## Additional Attributes from ER Diagram



The above models show the additional attributes and entities with detail that could not be covered in the Simplified ER Diagram. The colors match those from the Simplified ER Diagram as well.

# Preliminary Design

## User interface

The user interface is divided into four primary areas: header, map, tab, and footer. The following sections will provide a brief overview of each feature included and a preliminary mock-up with some key features.

## Window

After a user clicks on the SmartMap link in the Mann Library Homepage, the SmartMap application will be displayed in a new pop up window. This way the user can view the SmartMap and continue navigation the Mann Library website.

## Header

- |                     |                                                                                                                                                      |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Header/Title</b> | Affiliation with the Cornell University library system is emphasized by following the standard website logo and title style.                         |
| <b>Search</b>       | Since it is located in the same place as the standard Cornell search box, the SmartMap search is easily recognized and accessed.                     |
| <b>Links</b>        | Advanced Search and Help links are displayed right below the search box. If clicked, additional forms and information will be shown in the tab area. |

## Mockup



## Map

### Color-coding

Each room type should be associated with a color. All rooms in the library are divided into the following four categories: public area, private offices, collaboration area, and facilities. There colors to represent these areas will be green, yellow, blue, and brown respectively.

### Labeling

Most rooms are labels on the map. However, smaller item will have easily identifiable icons.

### Hovering

When the user hovers over an item, a dialogue box will appear on top of the room with basic information.

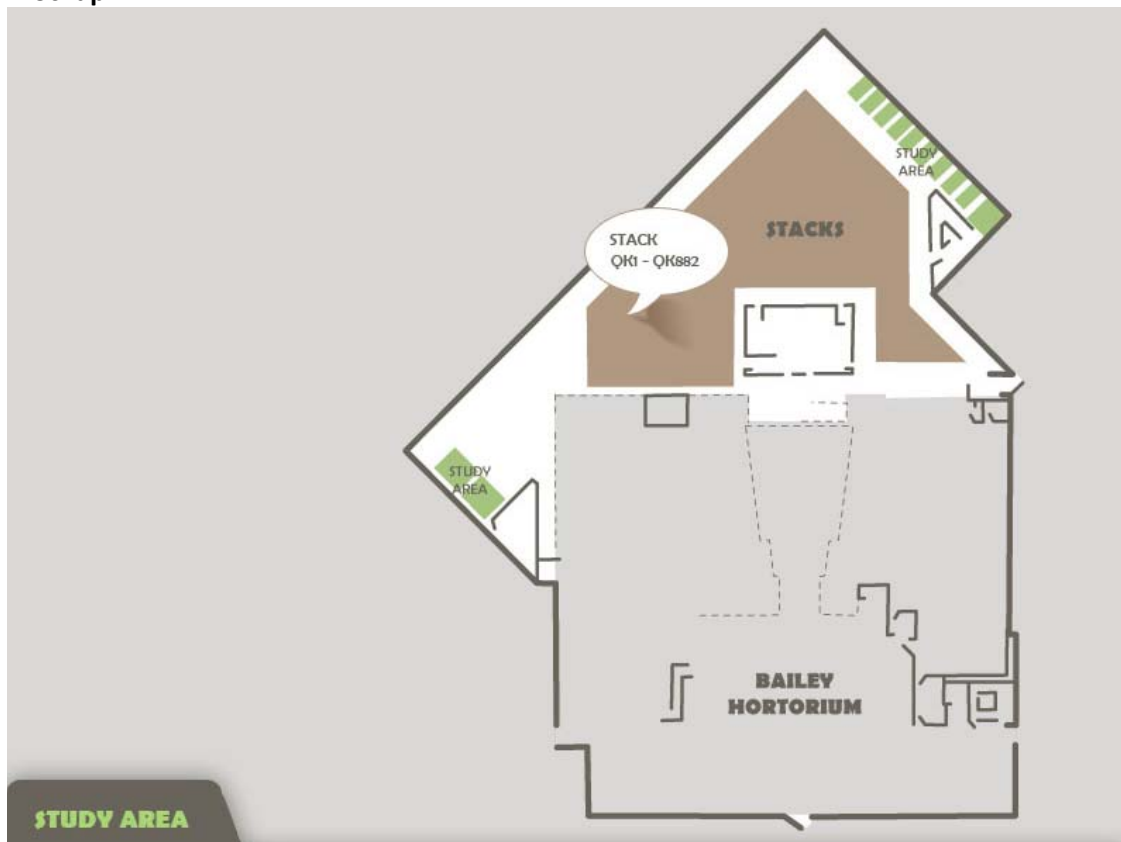
### Clickability

When a user clicks on a room, facility, or piece of equipment on the map, the map will bring up a display showing more in-depth information. The display is represented as a tab which will be elaborated in the next section.

### Toolkit

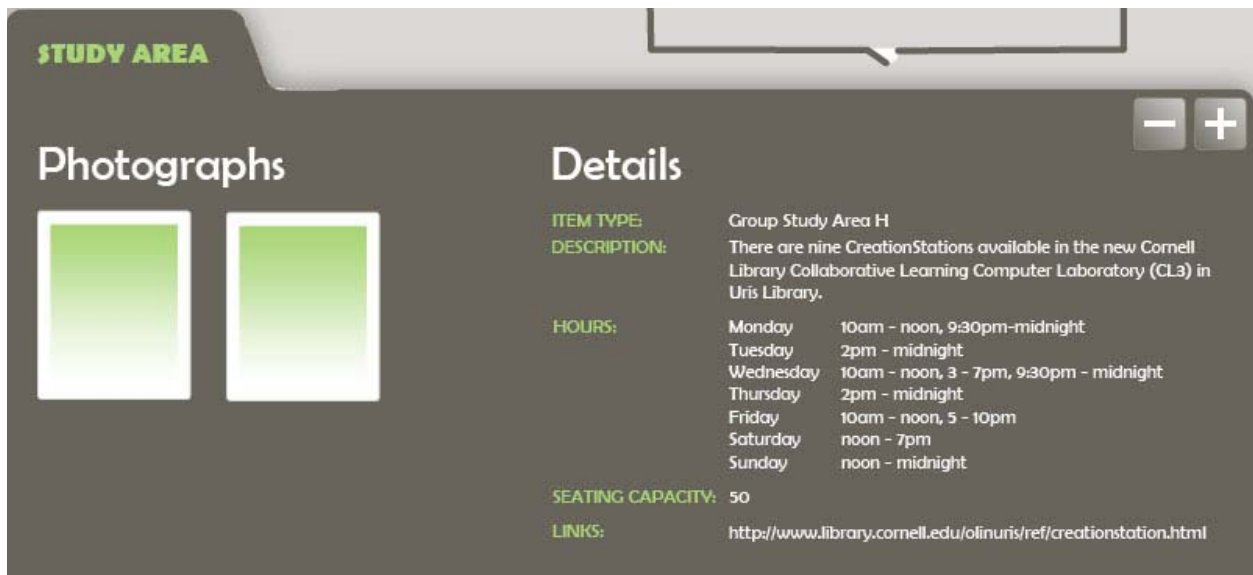
A toolkit will be provided on the left side of the map area. It will include a home, zoom buttons, and navigation buttons. It will also include a legend with description of the map color code.

## Mockup



## Tabs

- Label** The upper left corner of the tab will contain the room name in its color code.
- Photographs** Photographs of the room will be shown here if they are provided. Otherwise, the area will be replaced by the details section.
- Details** This section contains detailed room information such as the item type, description, hours, additional fields, and links.
- Buttons** The upper right corner may have several buttons, such as maximize and minimize buttons. In the admin mode, it will also have editing buttons.



## Footer

The footer will have additional links: Mann Library Homepage, Help, Admin, Contact, About, and Feedback. A new tab will appear for each one of these links with the exception of the Mann Library Homepage link.

## Mockup



## Mockup Design

The colors on the mockup page are chosen to match the current Mann Library color scheme. In addition, different shades of gray will be used to differentiate between areas of the application. SmartMap visitors will immediately be attracted to the map area because it will have more vibrant colors in comparison with the background.

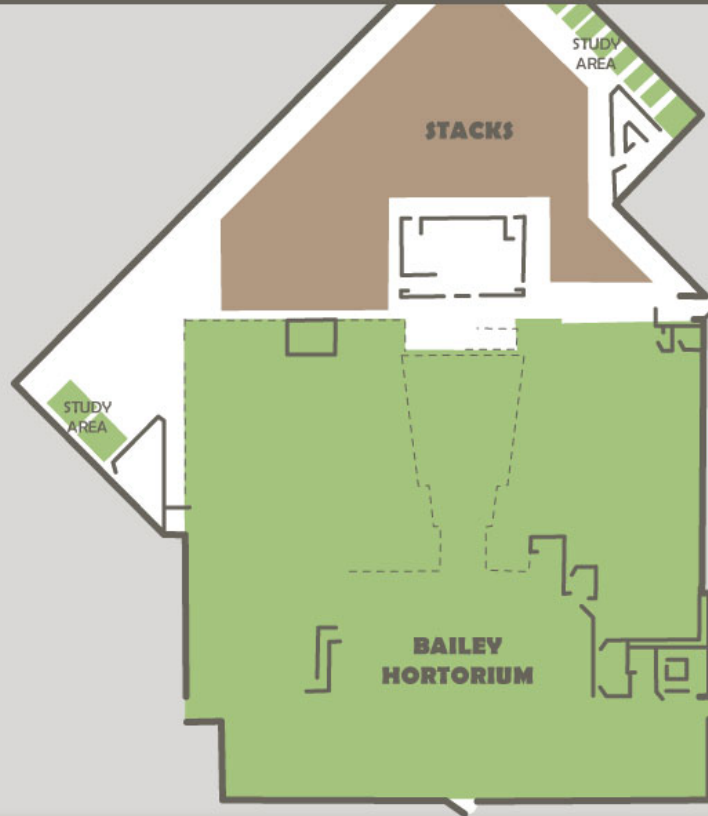


SEARCH MAP:

go

[Advanced Search](#)

[Help](#)



## STUDY AREA

### Photographs



### Details

**ITEM TYPE:** Group Study Area H

**DESCRIPTION:** There are nine CreationStations available in the new Cornell Library Collaborative Learning Computer Laboratory (CL3) in Uris Library.

**HOURS:**

Monday	10am - noon, 9:30pm-midnight
Tuesday	2pm - midnight
Wednesday	10am - noon, 3 - 7pm, 9:30pm - midnight
Thursday	2pm - midnight
Friday	10am - noon, 5 - 10pm
Saturday	noon - 7pm
Sunday	noon - midnight

**SEATING CAPACITY:** 50

**LINKS:** <http://www.library.cornell.edu/olinuris/ref/creationstation.html>

## System Design

### Model View Controller

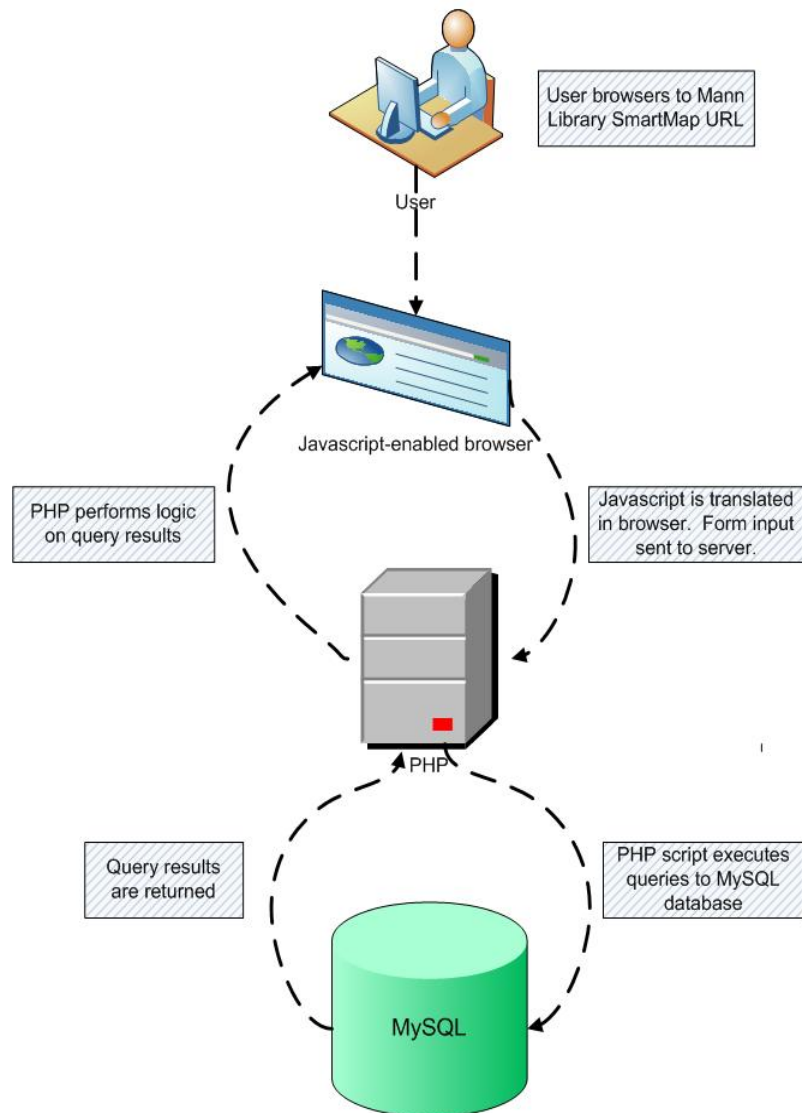
The system will utilize a three-tier architecture, as illustrated in the diagram.

The presentation tier will consist of XHTML, CSS, and JavaScript. These technologies are executed by the user's browser. JavaScript will be used to handle the interactive features of the map to increase accessibility - one of client's requirements. Users' browsers will have JavaScript enabled by default. Other technologies such as Flash or Java applets require proprietary software that users may not have installed.

PHP 5 will be used in the application tier. This part of the system queries the database so that pages can be dynamically generated based on user input. Unlike the technologies in the presentation tier, PHP is a server-side scripting language.

The database tier will use MySQL 5, a relational database. The database server will hold all of the metadata about the rooms and facilities in the library. A relational database is used so that we can model the associations between different items ( e.g. each photograph is of a certain room, each room is on a particular floor).

Both PHP 5 and MySQL 5 were picked because they will be compatible with the setup of the production server.



## Database Design

These tables represent entities from the previous ER diagrams. Instead of having one table with all possible items listed in it, the items will be separated by type of items. This method was chosen since some items will have additional attributes beyond the common set. This will allow the system to better address those items. Every item is uniquely identified by its id and its item type. For example, when mapping items to floors, the floor number, item id, and item type id is stored. That way, a café with id 001 can be differentiated from a bathroom with the same id.

The item types table will list all of the item types in our database. Instead of having an item type in every single item table (every record in the study rooms table would have the same item type), there will be separate table that links the study room table name to the item type id. There will also be a mapping of type name to type id so that it can be displayed on the map.

### Proposed Tables

BaileyHortorium	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

BirdWatchingStation	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

Cafes	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace menuURL eventsURL hoursURL

DrinkingFountains	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

OutdoorPatios	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace eventsURL

Restrooms	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace gender

Equipment	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace exhibitID

Photocopiers	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

Views	
PK	viewID
	name description image1 image2

StudyAreas	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace softSeats gradOnly

StaffOffices	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace departments contactName contactPhone contactEmail

StairwellsElevators	
PK	id
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

POIs	
PK	poiID
PK	stairwellElevatorID
	name description



BissetCollabAreas	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

Classrooms	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace classScheduleURL computerType numComputers software softwareURL computerURL seatingCapacity hoursURL reserveURL

StudyRooms	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace roomNumber studyRoomTypeID viewType projector BIB ID/Item ID

SeminarRooms	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace seatingCapacity scheduleURL

Lockers	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace lockerTypeID

LockerTypes	
PK	<u>lockerTypeID</u>
	name

StudyRoomTypes	
PK	<u>studyRoomTypeID</u>
	name description

ExhibitCases	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace exhibitID

StudentExpoSpaces	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace scheduleURL exhibitID

Exhibits	
PK	<u>exhibitID</u>
	name mannURL addURL image1 image2 description

PublicComputingAreas	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace

ServiceDesks	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace hoursURL phoneNumber emailAddress

Services	
PK	<u>serviceID</u>
PK	<u>serviceDeskID</u>
	name

LoanableItems	
PK	<u>loanItemID</u>
PK	<u>serviceDeskID</u>
	name description

Printers	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace color duplex printerFormatID

Stacks	
PK	<u>id</u>
	name xCoordinate yCoordinate mannURL addURL image1 image2 description collabSpace startCallNumber endCallNumber

Collections	
PK	<u>collectionID</u>
	stackID name description

PrinterFormats	
PK	<u>printerFormatID</u>
	name description

ItemTypes	
PK	<u>itemTypeID</u>
	tableName realName

Every Item that will be put onto the map (i.e. has X and Y coordinates) will have their itemTypeID here and a name. For example, an entry would look like: itemTypeID 003, tableName "BissetCollabArea", realName "Bisset Collaboration Area"

onFloor	
PK	<u>floorNumber</u>
PK	<u>itemTypeID</u>
PK	<u>id</u>

Every itemType is uniquely identified by its itemTypeID and it's id. In the onFloor table, you can list these items on the floor where it exists. Some items exist on multiple floors, such as elevators, so you would have 4 records in this table that shows that an Elevator 1 is on Floor 1, 2, 3, and 4.

This is the preliminary table design. Changes still have to be made to the schema to accommodate a separate images and URL table so that each item can be mapped to a dynamic number of images and URLs.