

FEASIBILITY STUDY

The Group

Douglas Tak-Lai Wong, dtw9@cornell.edu
Gregor Charles Carrigan, gcc26@cornell.edu
James Ioannidis, jti4@cornell.edu
Jeffery Zhang, jz87@cornell.edu
Talitha Lynn Forcier, tlf23@cornell.edu

The Client

Bob Kibbee, Map & GIS Librarian, Olin Library, rk14@cornell.edu

The Task to be Undertaken

The project is to develop an interactive web-based mapping application to search the complete range of geo-referenced materials in the Cornell University Library. The interactive map would allow a user to find maps through maps, providing a more intuitive interface to cartographic catalog browsing.

The project consists of three main parts: determining a database solution to hold the metadata for the library's map resources, developing an administrative interface that would allow for adding and changing metadata for map holdings, and developing a UI to expose this metadata to the end user (library patrons).

Benefits:

Currently library patrons have a difficult time locating maps due to the fact that the patron is in a "geographic" mindset and the library catalog requires that they translate their query into an often obscure textual query.

1. The proposed mapping application would allow additional metadata to be appended to current catalog information to aid indexing and searchability.
2. The proposed mapping application would provide a framework for library patrons to visually and intuitively locate paper maps and atlases, digital maps scanned from the collection, and GIS resources; thus overcoming the challenges of the current catalog system and promoting the accessibility of these valuable resources.

A Preliminary Requirements Analysis:

The system needs to meet the following functional requirements:

1. Web Interface
 - a) Administrator Side
 - i. Allow admin to modify, delete, or add the cartographic information for a particular map
 - ii. Search by region name
 - iii. Display necessary metadata information for a particular map
 - b) Public Side
 - i. Display interactive map
 1. Must be able to zoom to at least three levels: world, continent, and country
 2. Must provide a minimum level of geographic detail including country boundaries, country names, and a distinction between land and water
 3. Must provide both catalog information and possibly some additional metadata on holdings for the current region based on click location and zoom-level
 4. Must be able to make several key distinctions between maps, the only mandatory requirement here being the ability to distinguish topographic map series and their resolutions
2. Database to store cartographic information
 - a) Must be pre-populated from the library catalog
 - b) Must allow additional information to be added for each map
3. Be easily extensible from both the administrator's perspective and from future developers' perspectives

The system may have the following functional requirements:

Undecided

1. An interface to updating the map database with a more current catalog version, while maintaining additional metadata may be necessary.
2. The scope of and the amount of metadata we display for each cataloged map. It is possible that for basic purposes we may only present a user with a link to the library catalog entry for a particular map.
3. The administrative interface may not need to be web-based.

4. The functionalities to search, delete and modify metadata may need to be available.

Optional

5. The zooming level may expand to the state/province level.
6. The boundaries of a region may be near-precise given the Group has access to a geo-decoding tool.

Non-functional requirements are unspecified at this point. Foreseeable non-functional requirements may include: up-time, reliability, number of concurrent users supported, response time of the system, etc.

Technical Requirements – Feasibility:

1. *Server* – The system is going to be running on a server in Olin Library. The Group is currently working on a contact to find out exactly who would be in charge of this server. If nothing else, the Group could definitely be given a test server in Mann to test the development code on. Most of the libraries servers are Linux servers.
2. *Database* – The current library catalog system is hosted on an Oracle database. The Group will have access to this database, and permission to add tables as needed for the inventory control system.
3. *Web* – The system will need to be integrated with the current library web page, which uses common web technologies. The library staff is flexible about loading on new technologies if needed.
4. *Map* – The interactive map can be done by using Google Maps API, Yahoo! Maps API, TerraServer, and others. After some research of the various technologies, the Group has decided that Google Maps API works best for the project.
5. *Geo-decoder* – The system needs a geo-decoder to map longitude-latitude pairs into region/country codes/names. Currently, the Group has not found a public tool or service that provides this functionality. Without this tool, the Group will have to reside on a rough workaround, by first finding out the maximum and minimum longitude and latitude pairs of the various regions, and define a rectangular region in which a user may click on to select the region that the rectangular region is approximating.

Scope:

For the purposes of this project we are solely concerned with exposing the Olin library

map collection through an intuitive UI supported by an administrative interface. Development to the extent called out in the above requirements section is all that is planned. The end product will run on an Olin library web-server and will be viewable through all standard web-browsers.

We DO NOT plan to make other library resources available through this interface.

We DO NOT plan to expose the additional metadata to the library catalog nor any interface other than a map-based UI.

We understand that it is NOT necessary to dynamically link with the library's central database, so maintaining a separate database is a viable option.

Suggested Deliverables:

Management Deliverables:

1. *Requirements Analysis* – a document and a presentation to go over the formal requirements of the project, both functional and non-functional. This deliverable ensures that the Group is working on a system that closely matches to the wishes of the Client. This deliverable gives the Client a chance to modify and correct items that were mis-communicated or missed out before allowing the Group to proceed further in the design.
2. *Design Document* – a document and a presentation to go over the design of the system. This is the Group's opportunity to go over how the project is to be implemented to the Client. This deliverable is done by the more technical and experienced in the Group, based on the understanding of the requirements established in the previous deliverable.
3. *Source Code* – a document, presentation along with the source code of the final completed project. This final deliverable wraps up and concludes the project. In this deliverable, the Group delivers the final implementation based on the requirements specified and the design developed in previous stages. The system would have been tested thoroughly with unit tests and with a final acceptance test and would be ready for deployment to the production system.

Technical Deliverables:

1. A *database* with the required tables to support the inventory system—a database needs to be set up on the library servers with the tables needed in the system to store the inventory information, geo-referenced materials, and other cartographic data.
2. An *administrative interface* to add, modify, delete and search for inventory—a

webpage designed to allow the administrator (ie., the Client) of the system to add information to the inventory system for every map that is found in the library and to build up an electronic record of the resources that are found in the library.

3. An *interactive map* with labeled countries and clear national boundaries—a map of the world with zooming capabilities and re-centering functions, labeled with names corresponding to the current view of the map, that has clear boundary lines (ie., country border) on a web page.
4. A *side menu* that is populated with cartographic information based on the inventory—a portion of the web page that shows available cartographic resources for the selected region, based on information in the inventory database. The information that will be displayed has yet to be decided.

Walk-Through:

In order to ensure that the Client and the Group are on the same page, the following walk-through has been prepared to illustrate the Group's understanding of the product desired by the Client. The walk-through is not necessarily a reflection of the exact interactions for the final product; rather it should serve as a rough overview of the functionality required by the final product as the Group currently understands it.

Library Patron walk-through

The Patron is first presented with a map of the world labeled so that all countries are named and clickable. When a client selects the country of interest an info-page with a list of the types of maps (topological, street, etc.) or other resources (web links, etc.) relevant to that country along with a list of larger geographical regions which that the country is part of, each of which would link to their own info-page. Also close-up of the country is presented which could be used to add another level of detail to the interface (counties, cities, etc.). A click on the info page will use the library catalogue and the database the Group will be implementing to generate a page listing all the relevant maps. A click to any particular map will present a page containing all the information necessary to physically get the client to the map and anything appropriate that we can glean from the catalogue or database.

Administrative walk-through

The administrator needs a way to add, modify and delete entries in the database. To add entries the administrator will enter data on a simple text-box, pull-down menu populated page with all the necessary fields. To modify or delete entries the

Administrator could either use the 'client' interface to reach a particular entry and then modify it, or search on any of the fields in order to find an entry.

Software Development Process:

The project will undertake the *modified waterfall model* because there is a well-defined set of requirements. As the Client has very specific needs for the system which will not likely change in a short timeframe, and given that this is a production system (not a research project), the modified waterfall model should be better suited and gives the Group the following benefits:

1. *Process visibility* – both the Client and the Group are certain which stage of the development process the project is in.
2. *Separation of tasks* – the Group may concentrate on one area at a time, especially since some members of the Group have less experience in coding and in large scale software projects.
3. *Quality control* – a modified waterfall model allows the Group to spend more time on the requirements, understanding the design, and on developing better code (a programmer with less experience may have a difficult time delivering in short iterations in an iterative refinement model).

Outline Plan (Principal activities and Milestones)

- I. Milestone 1 (March 3, 2006) – Requirements Analysis (draft). An initial draft of the requirements analysis should be done as Milestone 1. This should come after a formal requirements gathering meeting with the Client.
- II. Milestone 2 (March 10, 2006) – Requirements Analysis (final). The final draft of the requirements analysis should be done for Milestone 2. In addition, a presentation will be prepared as a part of this milestone.
- III. Milestone 3 (March 24, 2006) – Software Architecture and Design (draft). An initial draft of the software architecture and design should be done as Milestone 3. A meeting with the Client should follow Milestone 3 to get feedback on the design of the system.
- IV. Milestone 4 (April 7, 2006) – Software Architecture and Design (final). A final draft of the software architecture and design document should be done for

- Milestone 4. A presentation should be prepared for the Client.
- V. Milestone 5 (April 14, 2006) – Database. The database is the most important part of the system, as it is the center of all information. All subsequent system components depends on this deliverable. A database schema needs to be fixed for Milestone 5 to provide a basis for the other components to be based on.
- VI. Milestone 6 (April 21, 2006) – Inventory Control. As the menu of cartographic information needs to be published using information in the database, the next bottleneck is the inventory control, which is a graphical interface to allow the administrator to enter, modify, and delete data.
- VII. Milestone 7 (April 28, 2006) – Map and Menu. The map and the menu are the front-end graphical web interface that the public user sees and interacts with. Milestone 7 is to reach feature-completion on the requirements.
- VIII. Milestone 8 (May 5, 2006) – Testing, Debugging and Integration. The system needs to be well-tested, debugged at this milestone. Also, once the system has passed the acceptance test, it needs to be integrated to the actual production system for this milestone.
- IX. Milestone 9 (May 11, 2006) – Project Deadline. The project source code should be handed over to the Client for the final milestone. A presentation is presented to the Client.

Visibility Plan

External – The Group will conduct regular biweekly meetings with the Client at the Olin library. If situations arise or if a problem needs to be addressed between the meetings, the Group will conduct any further necessary communication via email. Because a modified waterfall model will be used, a report will be issued to the Client at the end of every step to ensure that both parties are in-sync and to minimize any miscommunication in the requirements.

Internal – The Group will meet weekly on Wednesday evenings from 7:30 pm to 9:00 pm to discuss progress and problems. Meeting minutes will be kept track of and sent to all members of the Group for reference. Any additional communication will be done via email or through other collaboration tools such as document sharing. In

addition, the source code will be stored at gforge, a repository for the project. All source code will be documented carefully before being submitted to the repository. The progress of the principal activities and major milestones will be closely monitored and compared with the schedule. During the regular meetings, progress will be compared to a Gantt chart, which will be drafted after the completion of the requirements study when the tasks are identified and assigned.

Business Considerations

As Cornell students, the Group owns the copyright in the software that we create in this project. The Group agrees to transfer the copyright to the Client and to provide the Client with unrestricted license to use the system.

It is just possible that a project may develop concepts that could be patented. If such a situation arises, the Group collectively owns the rights to all patents associated with the System.

We understand that the use of *open-source solutions IS a viable option* and that there are not any serious licensing issues to this extent.

Risk Analysis

1. Changing Requirements:

Risk: The Client may have different ideas about the system during the course of the project. Depending on the situation, the changes that the Client wishes to have implemented may require little or major changes to the architecture.

Solution: To reduce the possibility of this occurring, the Group needs to establish a clear visibility plan with the Client.

2. Incomplete Requirements:

Risk: It is possible that requirements may be implied but not discussed or misunderstood. This frequently occurs after meetings.

Solution: The Group's interpretation of the Client's requirements will be presented back to the Client to get a confirmation on whether the Group has understood the Client. Frequent client updates and a high level of visibility will also help call attention to any misunderstandings.

3. Lack of Resources, Tools:

Risk: For the project to meet one of the functional requirements (clicking an area of a map within the boundaries of a country should select that country), a

geo-decoding tool is needed. However, at the time of writing of this document, no public/free tools can be found that can do this longitude-latitude pair and country conversion.

Solution: A workaround is proposed that would meet the minimal level of this requirement by defining the borders of the selection area to be a rectangle defined by the maximum and minimum longitude and latitude pairs.

4. System Integration:

Risk: Depending on the level of access to the servers that the Group receives, the Group may need to work on the system offline and eventually integrate with the production system when it is ready and thoroughly tested. Due to different software configuration, there may be unpredictable obstacles.

Solution: To ensure a smooth system integration, the Group needs to be aware of as much about the configuration as early as possible.

5. Technical Requirements:

Risk: The software and hardware server environment are not perfectly certain at this point. The client is not aware of the technical aspects of the project. The technical server configuration may have an affect on system architecture and design.

Solution: To resolve this problem, the Group has requested the client to refer the Group to the technical staff working at the library for further inquiries.

6. Non-functional Requirements:

Risk: Similar to incomplete requirements, non-functional requirements is something that has not been brought up in the initial meeting with the Client. These include requirements on the number of users that the system expects to support concurrently, and the response time of the database lookup.

Solution: A follow up meeting is needed to specify the non-functional requirements.

7. Human resources:

Risk: The Group is relatively small consisting of only 5 members, some members are not technically oriented and almost all members have limited knowledge of relevant web-technologies.

Solution: For these reason the Group acknowledges that a slow design and implementation phase may be inevitable, and are planning accordingly.

Conclusion:

Based on the analysis of this feasibility study, the Group has collectively agreed that

this project *IS FEASIBLE* and the group *IS WILLING* to take on the afore mentioned project. The benefits are significant enough to justify the development effort required. Currently, the costs of the project are believed to only involve the labor costs of the group. The *preliminary deadline* for the project will be set for *April 7*. On this date a functional system should be live on an Olin library server. The next stage of development will be the completion of a requirements study, which will be conducted and written up by March 3, 2006.