BASIC LEGAL CITATION

Feasibility Study

SEPTEMBER 25, 2015

FOR LEGAL INFORMATION INSTITUTE

Basic Legal Citation

for Legal Information Institute

Feasibility Study

Harish Sethumadhavan
Aiswarya Jami
Akshay Tata
Brenda Martis
Devi Snigdha Muppala
Sania Nagpal
Joao Curcio
Yihui Fu

September 25, 2015

TABLE OF CONTENTS

Basic Legal Citation

1 INTRODUCTION

1.1 EXECUTIVE SUMMARY

Writing a book as complex as Introduction to Basic Legal Citation and maintaining the book updated every year is a high demanding endeavor. If only the book was maintained as a website, would the workload of updating it be a matter of writing an HTML file with the contents of legal writing. However, the author Prof. Peter Martin has another three formats for his book, and each format has its own specific characteristics. Therefore, he has to re-write the same book four times, one for each format. He also needs to keep track of his progress, so that all versions will be consistent. These tasks are accomplished manually. Furthermore, when the book was published in 1993, there was not a concern to create a mobile-friendly and interactive design, which is almost a necessity nowadays. That's where our team enters.

Our team's goal is to design and to build a modern, mobile-friendly version of the Introduction to Basic Legal Citation site, and to create solutions to improve the content maintenance process to support efficient updating and output to the website and to various eBook formats.

The main client will be Sara Frug (sf6@cornell.edu), from the Legal Information Institute, with whom we will maintain a bi-weekly communication to present our progress. The system will be mainly used by the author, Prof. Peter Martin, but our success also benefits his audience: law students, paralegals, non-lawyers reading the law for professional purposes.

1.2 GROUP INFORMATION

The group is composed of 8 CS 5150 students, namely:

1) Harish Sethumadhavan hs643@cornell.edu
2) Aiswarya Jami aj493@cornell.edu
3) Akshay Tata at758@cornell.edu
4) Brenda Martis bom6@cornell.edu
5) Devi Snigdha Muppala dm754@cornell.edu
6) Sania Nagpal sn579@cornell.edu
7) Joao Curcio jfc293@cornell.edu
8) Yihui Fu yf263@cornell.edu

1.3 CLIENT INFORMATION

Our client is Sara Frug, from the Legal Information Institute of Cornell University. Her contact information provided is ssf6@cornell.edu. The Legal Information Institute (LII) is a well-known publisher of open-access legal information. It operates the single most active web site at Cornell, with approximately 150,000 unique visitors per day and over 100 million page views during the last calendar year. Since 1992, the LII has been a leader in the application of Internet-based technologies to legal data.

1.4 BENEFITS

The main beneficiated by this project would be the author, Prof. Peter Martin, and his audience: law students, paralegals, non-lawyers reading the law for professional purposes. The audience for the resulting product potentially numbers in the millions.

2 PRELIMINARY REQUIREMENTS ANALYSIS

2.1 APPLICATION OVERVIEW

This project involves implementation of a web based application that hosts the documentation regarding basics of legal citation. The application will have two interfaces, the author side and the user side. The author is the person who owns the content on the website and is also responsible for the perpetual updating of the same. The user base would be hundreds to thousands of law students who use this website as a resource to understand the legal writing and learn how to cite the legal sources.

2.1.1 Author View

The author view will consist of an editor interface that will allow him to edit the content of the website. This updated content is to be reflected in four formats (HTML, ePub, Mobi and PDF). Once the author decides to publish these changes, he will have an option to reflect the changes in any of the other formats without manually having to repeat the updating process.

2.1.2 User View

The application will allow the user to view the content hosted on the website in the form of classified web pages and markups.

2.2 FUNCTIONAL REQUIREMENTS

- The application will have an interface to update the website content;
- The client should be able to generate different formats of the same content (HTML, ePub, Mobi and PDF);
- The author interface may not need to be web-based;
- The users will have an interface to view the legal citation content;
- The user interface can be either a website, eBook on kindle, a mobile document reader or a pdf file;
- A website with good user experience and also a mobile friendly version has to be implemented;
- The functionality for the user to search for a citation/keyword in the website may be included as an optional feature.

2.3 NON-FUNCTIONAL REQUIREMENTS

- **Reusability and Maintenance**: The website must be easily maintainable once it is handed over back to the client with a descriptive user manual.
- **Scalability**: The website already has up to a million users. The website must be scalable enough to cater to an even larger number of users in future.

- **Consistency**: One of the major concerns of the client is to maintain consistency among the different formats. The feature provided for generating different formats should keep a uniform data across all.

2.4 TECHNICAL REQUIREMENTS

- **Version Control:** Each update on the website content can be treated as a version. We may use Git to keep a track of these versions.
- **Server**: The web page is going to be integrated on the main website for the Cornell Law School and uses the same server.

The formatted output generation will involve the following technologies:

- ePub – XML version
- Mobi – HTML
- PDF – MS Word
- Website – XML, AngularJS, HTML

3 SUGGESTED DELIVERABLES

3.1 MANAGEMENT DELIVERABLES

3.1.1 Requirements Analysis

A document that details the formal requirements of the client, both functional and nonfunctional. This document ensures that the team has understood the client's requirements and serves as a contract that outlines the list of tasks that are to be implemented by the team, to deem the project as complete.

This document may be subject to modification in each iteration of the project. During the review phase of the software development process, the client can choose to request changes to the requirements. The new requirement edits will be closely studied and analyzed for feasibility by the development team and subsequently, a new version of the requirements document will be written.

3.1.2 Design Document

This document details the architecture and design considerations of the system, both hardware and software. This document is written after the client's approval of the requirements document. This ensures that the system design is competent and efficient to satisfy the client's requirements. The design document will be written only after all technical intricacies of the system has been decided upon.

3.1.3 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.

3.2 TECHNICAL DELIVERABLES

A proof-of-concept of the requirements were done and the group is now confident that this project is feasible. Based on the findings of the proof-of-concept, the following technical deliverables have been charted out.

3.2.1 Part A

A generic framework that allows instant conversion of markup text into HTML, Mobi, ePub and PDF formats.

3.2.1.1 Language Markup

Define a new XML with tags specific to this project. Each tag in the new XML will have an equivalent mapping to an HTML tag, Mobi tag, ePub tag and PDF. The mappings of these tags will be stored in "config" file and this deliverable will be used in rest of the deliverables. This deliverable involves thorough research of the existing website's source code and ensuring that all possible HTML tag types have been accounted for in the config file.

If there are some HTML tags, like highlight for instance, that can't be mapped into Mobi format, another equivalent scheme of conversion must be specified in the config file. This can include a mapping like, "if there is highlighted text in the web content", "increase the font size of highlighted text in Mobi format", so that new XML tags are cross-compatible across all formats and devices.

If there is a tag that can have no conversion into a particular format, that too must be documented in the config file. For instance, if the web page has an embedded video, this can't be embedded into a PDF or an eBook. So, the video embed tag will be translated to nothing, and will be ignored when trying to create an ePub or Mobi file.

3.2.1.2 The Mapper

A script that will accept the "config" file from the previous deliverable as an input, the input markup file and the output type. This script will convert the input markup file into a required output markup type by replacing tags according to the mappings specified in the "config" file. This mapper will essentially serve as the script that converts documents from one type to another, without any manual labor.

3.2.1.3 Front-end UI editor for the mapper tool

The front end UI tool will help add/delete text to the original document and apply the newly defined XML tags for each section of the document. It will also have buttons – "Generate HTML", "Generate Mobi", "Generate ePub", "Generate PDF". Each of these can be clicked to generate the required type of output file.

In addition to these features, the UI editor will also recognize the "config" file and present the tag mappings in a very intuitive readable format. This will facilitate the user to add new tags, edit existing one, or to modify the mapping for any given tag type. This will ensure that the framework is agnostic and that the tags that the user requires in the output file are easily configurable, without any modification of source code.

3.2.1.4 Git Integration for the document version control

The document is very large and contains mostly static content. Using a database on the backend will be in-appropriate for our use cases, given the size of the text and low transaction frequency.

We instead decide to use Git on the backend for document version control. The first commit to the Git server will be the entire document, after it has been re-tagged using our newly defined XML tags. Subsequent updates will be backed up on the Git server, by computing only the difference between the last 2 revisions of the document. This model will scale well because the entire document is rarely replaced and most updates happen on specific sections of the documents or when new sections are added.

Version control will also enable the user to "roll back" to a particular document version, instantly generate HTML, ePub or Mobi formats for that document version, etc.

3.2.2 Part B

Enhancing the LII website's user-interface, making it mobile device friendly and optimal for all screen sizes.
The proof-of-concept for this part of deliverables were done using AngularJS, an interactive JavaScript framework used for web design.

3.2.2.1 Converting frames to tabbed panes

Currently, the website has been designed as 2 frames, one with the index on the left and the main frame has the content. The frames will be replaced with tabbed panes and effective navigation tabs that are user-friendly.

3.2.2.2 Adding a search bar

The current LII site doesn't have an integrated search bar that allows users to easily search the entire site for content. The search bar will be responsive and the search results page will have only snippets of document sections where the search keywords were found.

3.2.2.3 Replacing inline frames

Currently, the site has several inline frames and the content for each of these frames are being taken from separate static input files. These will be replaced by different design layouts within the current document.

3.2.2.4 Usability, SEO

Improve the overall usability of the website. This may include changing the coloring scheme where appropriate, adding tabbed panes, navigation tabs, popup menus and clickable keywords, where required.

This deliverable also focuses on search engine optimization to make the LII site ranked higher in search engine results.

4 SOFTWARE DEVELOPMENT PROCESS

The project will undertake spiral development model. We chose spiral development for our design model because we feel it is the best suitable for our current standing. Some parts of our system are clearly well defined and understood but there are few parts which need more clarification in the requirements and the design options.

For all the components which are clearly understood, we will use the sequential process of development, whereas for those components which still need more understanding, we will use an iterative approach. We will try to integrate all components in such a way where we have a functioning version of the given system at any given point of time. Given the requirements and the available resources, we feel that Spiral Development Model is more suited for our group.

The following are the benefits of using a Spiral Model:

- High amount of work analysis and hence, avoidance of risk is enhanced. Since we are putting a lot of effort in understanding the work process and the requirements, there is a high chance of avoiding the risk.
- We have a strong approval and documentation control among the members of the group.
- Software is produced early in the software life cycle.

5 OUTLINE PLAN

5.1 MILESTONES WITH EARLIEST START DATES

I. **Milestone 1 (September 18, 2015)** – Feasibility Study: In this step, we identify the problem and do the analysis to check if it can be solved efficiently. This involves in-depth study of issues involved and few implementation decisions. This has to be discussed with the client in order to confirm the formal requirements of the project.

II. **Milestone 2 (September 25, 2015)** – Study of structure of files: Legal citation text format is mainly categorized into four parts. Web pages where HTML is the markup language, eBook format (ePub standard), Mobi format used mostly on the kindle and pdf format on a variety of computer platforms. This step includes study of structure of different file formats.

III. **Milestone 3 (September 25, 2015)** – Creation of configuration files: The mapping of above file format is stored in "configuration" files. This involves a lot of research on the existing text code and all the tag types should be included in the configuration file.

IV. **Milestone 4 (October 2, 2015)** – Implementation of generic file generator: This step is used to create a generic script that will take the above generated configuration files from different version of the file format into a common markup type and in turn generate four different documents without manual intervention.

V. **Milestone 5 (September 25, 2015)** – User interface design: The users already have a website to view the legal citation content. The task here is to make it more user friendly and presentable.

VI. **Milestone 6 (October 5, 2015)** – Test the consistency of files – Phase 1: This is the step where we test the outcome of phase 1 thoroughly with unit testing, before it goes for acceptance test.

VII. **Milestone 7 (October 12, 2015)** – Presentation 1 – Phase 1 presentation will be prepared as a part of this milestone.

VIII. **Milestone 8 (October 19, 2015)** – Incremental study of structure of files – After the review of phase 1, the team will modify and add some more changes to the configuration file and a mapper. This step will include more improved features.

IX. **Milestone 9 (October 19, 2015)** – Iterative implementation – This step will include fully implemented environment and will cover desired and additional features as suggested by client.

X. **Milestone 10 (November 2, 2015)** – Configuration text editor - This step will include Git configuration for the document version control and front end UI editor for the mapper tool.

XI. **Milestone 11 (October 30, 2015)** – Test the enhanced product – Phase 2 – Before system goes for client acceptance, the system will be thoroughly tested out on various test cases and make it fully functional for end users to access it.

XII. **Milestone 12 (November 20, 2015)** – Review the final product – In this step, software will be complete before the final presentation, it will give the client expected functionality of the final product.

XIII. **Milestone 13 (November 2, 2015)** – Presentation 2 – This will cover the content maintenance process to support efficient updating and output to the website and various eBook formats

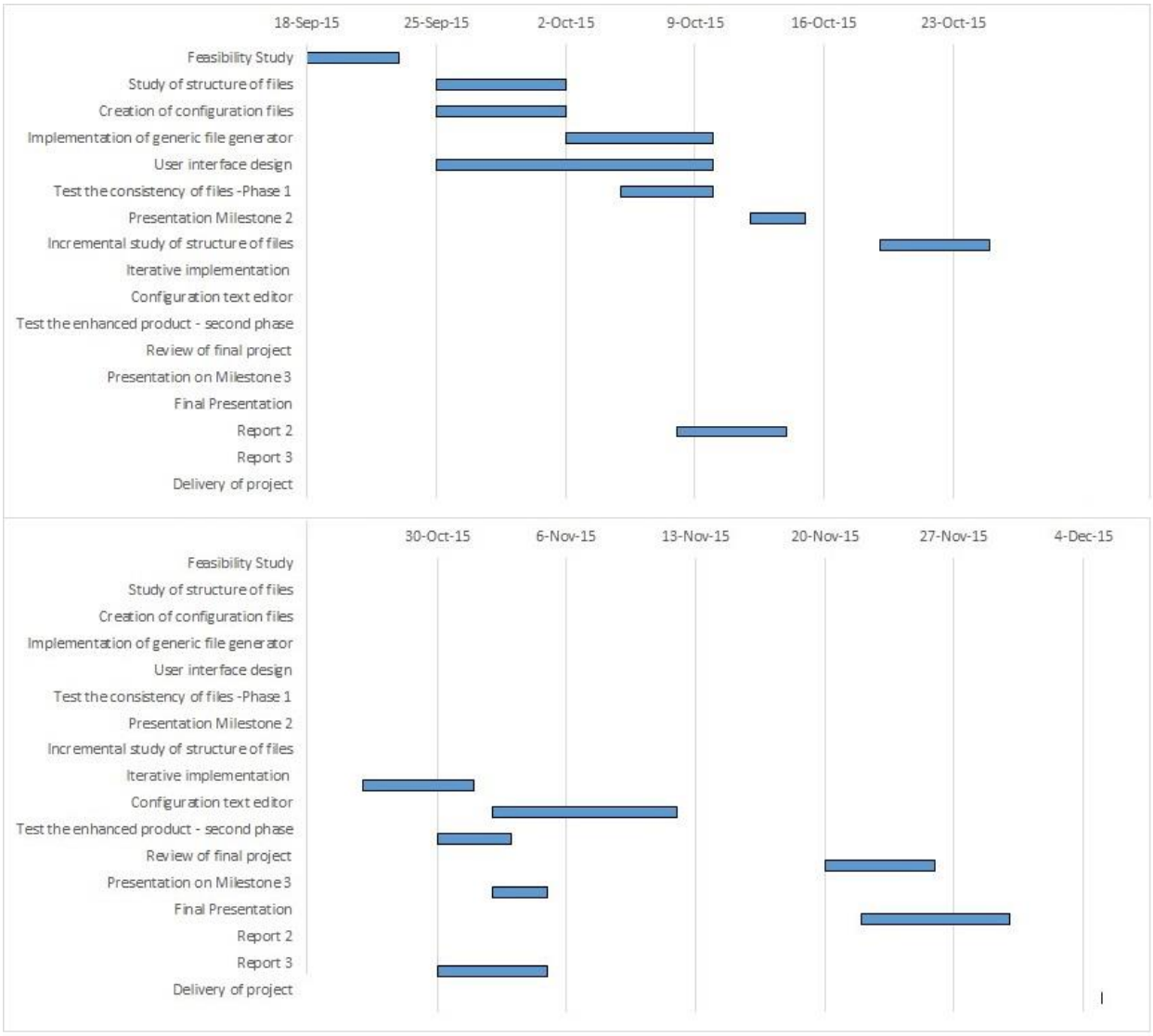
XIV. **Milestone 14 (November 22, 2015)** – Final Presentation – We prepare for final presentation along with the final deliverable wraps up where we conclude final implementation upon requirement specified.

XV. **Milestone 15 (October 8, 2015**) – Report 2 – We make the report ready for phase 1.

XVI. **Milestone 17 (December 5, 2015)** – Delivery of Project – This is the step where we prepare for live demonstration, presentation and final report. The final goal is to deliver a satisfactory system on time to the client.

5.2 GANTT TABLE & GANTT CHART

No	Task	Start Date	End Date	Duration	Slack	Status
1	Feasibility Study	18-Sep-15	25-Sep-15	5	2	Completed
2	Study of structure of files	25-Sep-15	5-Oct-15	7	3	In Progress
3	Creation of configuration files	25-Sep-15	5-Oct-15	7	3	In Progress
4	Implementation of generic file generator	2-Oct-15	12-Oct-15	8	2	In Progress
5	User interface design	25-Sep-15	12-Oct-15	15	3	New
6	Test the consistency of files -Phase 1	5-Oct-15	12-Oct-15	5	2	New
7	Presentation Milestone 2	12-Oct-15	16-Oct-15	3	2	New
8	Incremental study of structure of files	19-Oct-15	26-Oct-15	6	2	New
9	Iterative implementation	26-Oct-15	2-Nov-15	6	2	New
10	Configuration text editor	2-Nov-15	16-Nov-15	10	4	New
11	Test the enhanced product - second phase	30-Oct-15	4-Nov-15	4	2	New
12	Review of final project	20-Nov-15	28-Nov-15	6	2	New
13	Presentation on Milestone 3	2-Nov-15	6-Nov-15	3	1	New
14	Final Presentation	22-Nov-15	4-Dec-15	8	4	New
15	Report 2	8-Oct-15	16-Oct-15	6	2	New
16	Report 3	30-Oct-15	6-Nov-15	6	2	New
17	Delivery of project	5-Dec-15	12-Dec-15	5	2	New


18-Sep-15
25-Sep-15
2-Oct-15
9-Oct-15
16-Oct-15
23-Oct-15
Feasibility Study
Study of structure of files
Creation of configuration files
Implementation of generic file generator
User interface design
Test the consistency of files -Phase 1
Presentation Milestone 2
Incremental study of structure of files
Iterative implementation
Configuration text editor
Test the enhanced product - second phase
Review of final project
Presentation on Milestone 3
Final Presentation
Report 2
Report 3
Delivery of project
30-Oct-15
6-Nov-15
13-Nov-15
20-Nov-15
27-Nov-15
4-Dec-15
Feasibility Study
Study of structure of files
Creation of configuration files
Implementation of generic file generator
User interface design
Test the consistency of files -Phase 1
Presentation Milestone 2
Incremental study of structure of files
Iterative implementation
Configuration text editor
Test the enhanced product - second phase
Review of final project
Presentation on Milestone 3
Final Presentation
Report 2
Report 3
Delivery of project

6 VISIBILITY PLAN

The team aims to maintain an efficient visibility plan of the project progress and ensure that the progress meets the client's requirements. Deviations from the requirements will be reported in time and new project subtask might be created based on client's feedback. Visibility plans with the client and within the team are discussed below.

6.1 EXTERNAL PLAN

The team will conduct weekly meetings with the client for the first three weeks and then meet biweekly for all the other meetings.

The team will also conduct weekly meetings within the group.

The meeting will take place in the Cornell Law School. If there is any situation which needs immediate attention, the group will communicate via email and depending on the emergency we will try to schedule a meeting in between, contingent upon the availability of other members in the group. Since we have used the spiral model for development, this allows the client to periodically evaluate the output of the project before the development and testing phases are done.

6.2 INTERNAL PLAN

The team as a whole will meet weekly on Monday evening from 8.30 to 10pm to report progress and discuss problems. We will also keep track of the meeting minutes which will be shared with all the team members. We will also setup a collaborative communication platform which we will use for group discussions and one-on-one chat. Code will be uploaded to GitHub, an online repository for the project that helps with version control and document sharing. All the code will be documented carefully before being pushed to the repository. The workflow and progress of all milestones will be closely monitored by all group members and compared with the schedule in regular meetings.

We will also use Gantt charts which will illustrate the start and the finish dates of the terminal elements and summary elements of the project.

7 BUSINESS CONSIDERATION

In general, as Cornell students, the team owns the copyright in the software that we create in this project. The team agrees 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.

A detailed draft agreement is attached in the appendix. The team will confirm this with the client and has it signed.

8 RISK ANALYSIS

8.1 MUTABLE REQUIREMENTS

Risk: The requirements of the project are not concrete, which means that the client could think of an idea that he feels may have been important and was not collected as a part of initial requirement gathering. These changing requirements may lead to variations in the design and implementation in the later stages of the software development process.

Solution: To minimize the risk, the team needs to establish a clear plan with the client and fix the requirements permanently.

8.2 INDEFINITE REQUIREMENTS

Risk: The client and the project team may be on a different page and have a different understand of the requirements.

Solution: To ensure uniformity, the project team intends to meet the client and provide a reverse knowledge transition. This will overcome any gaps in the understanding, if there is any, and make the understanding of requirements more robust, clear and reduce inconsistency.

8.3 TECHNICAL BOUNDARIES

Risk: The team has not planned what tools we are exactly going to use for building the project. There is fair but not a concrete idea about the details of the tools and if we need to develop our own tools to solve problems that will occur in the project.

Solution: To minimize the risk we have scheduled work among different members of the teams who will target specific brackets of the problem and see which technology fits right to solve a particular problem. Also we have decided to ask for client insight to minimize the risk.

8.4 SYSTEM INTEGRATION

Risk: Because our team is not building the system from ground up, we have to integrate some our ideas into the existing system. We have anticipated that this may lead to issues while merging. This may lead to some unexpected roadblocks.

Solution: To ensure smooth integration we have anticipated loopholes and areas where the system is likely to breakdown and will work on this to make it robust.

8.5 GROUP POTENTIAL

Risk: While our team members are not experts in all technologies, the understanding of technologies overlap among most members. As we work through the project, the team members have to pick up new technical skills which might affect time lines.

Solution: To tackle this, we have come to an understanding that we will spend a week's time on researching on solving problems using a divided approach and identify concrete technologies most relevant to this project to minimize the risk.

8.6 SCHEDULE RISKS

Risk: Project schedules slip when project tasks and schedule release risks are not addressed properly. Schedule risks may affect the project and lead to frustrating situation and could affect the quality of deliverables.

Solution: To tackle this, we have underestimated our time line and have planned to keep two weeks as a buffer to handle any delay in time lines

9 CONCLUSION

Legal Citation is undoubtedly a challenging project. Many new tools will have to be learned and some implemented. Nevertheless, our feasibility study strongly agrees that the project is feasible and should be carried on. Therefore our team is continuing with the development of the project and is prepared to face whichever tasks are proposed, in order to deliver a complete software by the deadline stipulated.

10 APPENDIX

10.1 APPENDIX A: AGREEMENT

Cornell Law School: Legal Information Institute

The undersigned agree to the following:

That all code, documentation and other copyright-protected material produced in the course of this CS5150 project (Project Material) shall be understood by all to be the work of joint authors and not as a work made for hire;

That should one or many concepts developed from the project could be patented, the joint authors collectively own the rights to all patents associated with the Project Material. Sara Frug and the Legal Information Institute will have non-exclusive rights to use the system and full rights to use and modify the system regardless of patents held by the joint authors;

That the joint authors shall include all the undersigned, the CS 5150 students working on the project and Sara Frug;

That despite joint authorship there will be no duty on the part of the student authors, individually or as a group, to account for any return on subsequent commercial use or development of the Project Material;

That, in contrast, should Sara Frug or the Legal Information Institute realize royalties or other direct financial return from licensing any of the Project Material there will be a duty to account to the other joint authors for any such revenue net of costs; and that the undersigned will use care to assure that the Project Material does not incorporate code covered by copyright and licensed on terms that are inconsistent with unlimited noncommercial distribution;

There is no Warranty; however Developers will do their best to fulfill requirements, but have no legal duties to do so.

[Signatures]