each step of the iterative process detailed below, the team will be adding in the necessary software modules that will handle the central data repository, report generation, and other functionalities.

Below is the proposed outline of the iteration stages and milestones including what the team expects to have completed at each stage.

Process Outline

User testing throughout: At each milestone, the team will present the most current version of the software to the clients for their testing and evaluation. Although the software will not initially be fully complete during the early presentations, it will give the clients an approximation of the functionality of the final product.

1st iteration (March 10, 2006)

Requirements Document

The team will prepare a formal document that will detail the client's requirements for the software. The client will categorize these requirements into required, desired, and optional features. After the document is approved by the client, the team will design the initial user interface for the client's evaluation.

Mockups: Simple User and Administrator Interfaces

To ensure that the interface designs meet with the client's specifications, the team will develop a user level and an administrative level interface that will have all of the required data entry fields, but will be otherwise non-functional. User level interfaces will contain the proper entry fields to add reference statistics. Initially, these interfaces will be simplistic in nature and may not contain all of the intended fields and functions that the clients propose.

2nd iteration (April 7, 2006)

Design Document and Presentation

A formal document will be prepared that details both the design of the system and the code behind it. In describing the design of the system, the hardware and software needed from the client will be specified. Details about the program design will cover internal functionality so that the client can maintain and modify the end product in the future.

The team will also hold a forty-five minute presentation as a progress report for the client. As part of this progress report, the team will update its task schedules to ensure that the system will be satisfactorily complete by the end of the final iteration.

Revised User and Administrator Interfaces

Based on feedback from the client concerning the first iteration deliverables, the team will modify the design of the user and administrator interfaces as needed. This modification will include more planned features and advanced abilities that will aid in diagnosing users' interaction with the system. Again, during this iteration, the interface may not have all planned features, but will have all those required for this stage of development.

3rd iteration (May 11, 2006)

Final Testing Period

The team plans to reserve two weeks before the final presentation so that the client can test the product in its intended environment with real users and data. All functional requirements will have been met before this point; any changes hereafter will only cover small details, such as aspects of the user interface.

Final Documentation and Presentation

VII. RISK ANALYSIS

As with any project, this undertaking is not entirely risk-free. Three major risk categories have been identified: time, resource, and functionality.

Time Risks

As course requirements specify that the project must be completed within one academic semester, any extensions are not possible. This introduces the risk that the system may not be completed with the full functionality the client wants within the given timeframe of a semester. In this case, there is also a second risk of delayed implementation of the system if the client chooses to wait until the system is fully functional.

Resource Risks

Resource risks involve technologies the team has available for their use. Due to costs and other external constraints, the team may not be able to obtain the needed or best resources to complete parts of the system. For example, one identified resource is that team members will need laptops running a Windows operating system. Luckily, Cornell's Computer Science department has supplied this resource to those team members in need of laptops.

In addition, there are inherent risks in the resources, such as the software and hardware the team decides to use. Currently, to keep costs at a minimum, the team is considering open-source software, which is available without charge. Hardware for actual system implementation is also dependent on the hardware the library has available. Risks that stem from resources include hardware failures, system crashes, bugs in the code, etc., which may cause accidental data loss. Changes in the computer system of the library in the future that largely cannot be foreseen may also cause the system to malfunction. Since part of the system is web-based, slight variations in display of the user interface may also occur due to different internet browsers.

Functionality Risks

Functionality risks have to do with how the system works. Issues that fall under this category include developing a user interface that is not user-friendly or not well-liked by the client, or producing functions that have limited sustainability. The biggest risk comes from developing a system that does not do what the client wants it to do.

Out of the three risk categories, functionality risks are the easiest to reduce since functionality constraints are more flexible than time or resource constraints. However, minimizing functionality risk is usually accomplished by omitting specific parts and/or functions of the system, as decreasing functionality naturally decreases its associated risks. The team would like to avoid doing this as much as possible. The clients must be aware that it is possibile that this must be done in order to deliver the system by the due date at the end of the semester.

Risk Management/Minimization

Having outlined the basic risks associated with this project above, the team is prepared to take precautionary actions to minimize these risks. The principle plan is to develop and practice good management strategies. The team intends to divide the project into a series of iterative phases that have concrete milestones as discussed in previous sections. These milestones will provide project visibility and allow the client to see the team's progress at each stage. Multiple mockups will allow the client to evaluate usability, increase familiarity with the system, and shorten the learning curve.

Frequent communication and feedback from the client are also essential for client satisfaction with the user interface and functionality. The team will also constantly review their progress and modify goals if necessary to deliver a satisfactory system on time to the client.

VIII. BUSINESS CONSIDERATIONS