Designing and implementing a user-focused Web-based database system

Rachel Noiseux
Colby College

Follow this and additional works at: http://digitalcommons.colby.edu/honorstheses

Part of the Computer Sciences Commons

Colby College theses are protected by copyright. They may be viewed or downloaded from this site for the purposes of research and scholarship. Reproduction or distribution for commercial purposes is prohibited without written permission of the author.

Recommended Citation
http://digitalcommons.colby.edu/honorstheses/183

This Honors Thesis (Open Access) is brought to you for free and open access by the Student Research at Digital Commons @ Colby. It has been accepted for inclusion in Honors Theses by an authorized administrator of Digital Commons @ Colby. For more information, please contact enhodes@colby.edu.
Designing and Implementing a User-Focused Web-Based Database System

Rachel Noiseux

Honors Thesis
Advisors: Clare Bates Congdon and Marc Smith
Department of Computer Science
Colby College

Spring 2004
Acknowledgements

First and foremost, I would like to thank my two advisors, Clare Bates Congdon and Marc Smith. Without the two of them, this project would not have been possible. The numerous hours they spent meeting with me, reading and editing my papers, as well as guiding me in the right direction with this project was a tremendous help.

Also a special thank-you to Humphrey Sorenson, who so eagerly jumped into this project mid-year. Without him, the database section would not have been nearly as complete. His time and energy spent meeting with me and helping me fix problems is much appreciated.

Many thanks to my CS 352 group, Marc Attiyeh and Chris Roberts, for their initial work on this project. Their ideas and work throughout the semester provided me a great basis for which to build on. An additional thanks to Marc, for helping me learn PHP and MySQL, as well as getting me out of many sticky coding situations.

A special thanks to Leanne Burnham for her time spent meeting with me over the past year and a half. Her willingness to participate in user testing was extremely beneficial to my project. Also thanks to those who did expert user testing for me: Marc Attiyeh, Adrian LaRochelle, Skyler Place, and numerous members of this spring's CS 352. Their time spent testing the interface, and providing comments helped shape the system.

Lastly, thanks to Skyler Place for his help with creating the diagrams and figures. His efforts alleviated many hours of frustration for me. Without him, my diagrams would not be nearly as beautiful.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table of Figures</td>
<td>v</td>
</tr>
<tr>
<td>Chapter 1 - Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Chapter 2 - Background on User-Centered Design</td>
<td>3</td>
</tr>
<tr>
<td>2.1 Principles of Design</td>
<td>3</td>
</tr>
<tr>
<td>2.1.1 Visibility</td>
<td>3</td>
</tr>
<tr>
<td>2.1.2 Feedback</td>
<td>4</td>
</tr>
<tr>
<td>2.1.3 Constraints</td>
<td>4</td>
</tr>
<tr>
<td>2.1.4 Mapping</td>
<td>5</td>
</tr>
<tr>
<td>2.1.5 Consistency</td>
<td>5</td>
</tr>
<tr>
<td>2.1.6 Affordances</td>
<td>5</td>
</tr>
<tr>
<td>2.2 Overview of the UCD Cycle</td>
<td>6</td>
</tr>
<tr>
<td>2.3 Design Phase</td>
<td>7</td>
</tr>
<tr>
<td>2.3.1 User Analysis</td>
<td>7</td>
</tr>
<tr>
<td>2.3.2 Task Analysis</td>
<td>7</td>
</tr>
<tr>
<td>2.3.3 Competitive Analysis</td>
<td>8</td>
</tr>
<tr>
<td>2.3.4 Social Impact Statement</td>
<td>9</td>
</tr>
<tr>
<td>2.3.5 Testable Usability Goals</td>
<td>9</td>
</tr>
<tr>
<td>2.3.6 Concrete Example Tasks</td>
<td>9</td>
</tr>
<tr>
<td>2.4 Prototype Phase</td>
<td>10</td>
</tr>
<tr>
<td>2.5 Evaluation Phase</td>
<td>11</td>
</tr>
<tr>
<td>Chapter 3 - Background on Databases</td>
<td>14</td>
</tr>
<tr>
<td>3.1 Entity-Relationship Model</td>
<td>14</td>
</tr>
<tr>
<td>3.1.1 Entity</td>
<td>14</td>
</tr>
<tr>
<td>3.1.2 Attributes</td>
<td>15</td>
</tr>
<tr>
<td>3.1.3 Entity Set</td>
<td>15</td>
</tr>
<tr>
<td>3.1.4 Relationships</td>
<td>16</td>
</tr>
<tr>
<td>3.1.5 Entity-Relationship Diagram</td>
<td>18</td>
</tr>
<tr>
<td>3.2 Mapping an ER Diagram to a Relational Database</td>
<td>19</td>
</tr>
<tr>
<td>3.3 Data Normalization</td>
<td>21</td>
</tr>
<tr>
<td>3.4 Implementation</td>
<td>23</td>
</tr>
<tr>
<td>Chapter 4 - Details of the Task</td>
<td>25</td>
</tr>
<tr>
<td>4.1 User Analysis</td>
<td>25</td>
</tr>
<tr>
<td>4.1.1 Administration</td>
<td>25</td>
</tr>
<tr>
<td>4.1.2 COOT Leaders</td>
<td>27</td>
</tr>
<tr>
<td>4.1.3 First-Year Students</td>
<td>27</td>
</tr>
<tr>
<td>4.2 Task Analysis</td>
<td>29</td>
</tr>
<tr>
<td>4.2.1 Administration</td>
<td>29</td>
</tr>
<tr>
<td>4.2.2 COOT Leaders</td>
<td>30</td>
</tr>
<tr>
<td>4.2.3 First-Year Students</td>
<td>32</td>
</tr>
<tr>
<td>4.3 Concrete Example Tasks</td>
<td>35</td>
</tr>
<tr>
<td>4.4 Social Impact Statement</td>
<td>36</td>
</tr>
<tr>
<td>4.5 Competitive Analysis</td>
<td>41</td>
</tr>
<tr>
<td>4.6 Testable Usability Goals</td>
<td>50</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>2-1</td>
<td>UCD Cycles</td>
</tr>
<tr>
<td>3-1</td>
<td>Example of 1:1 Relationship</td>
</tr>
<tr>
<td>3-2</td>
<td>Example of 1:* Relationship</td>
</tr>
<tr>
<td>3-3</td>
<td>Example of <em>:</em> Relationship</td>
</tr>
<tr>
<td>3-4</td>
<td>Examples of ER Symbols</td>
</tr>
<tr>
<td>3-5</td>
<td>ER Diagram to Table Conversion, Step 1</td>
</tr>
<tr>
<td>3-6</td>
<td>ER Diagram to Table Conversion, Step 2</td>
</tr>
<tr>
<td>3-7</td>
<td>Data Normalization, 1NF</td>
</tr>
<tr>
<td>3-8</td>
<td>Data Normalization, 3NF</td>
</tr>
<tr>
<td>4-1</td>
<td>Harvard’s FOP Main Page</td>
</tr>
<tr>
<td>4-2</td>
<td>Yale’s FOOT Main Page</td>
</tr>
<tr>
<td>4-3</td>
<td>William’s WOOLF Main Page</td>
</tr>
<tr>
<td>4-4</td>
<td>William’s WOOLF Page – Frames Noticeable</td>
</tr>
<tr>
<td>4-5</td>
<td>Middlebury’s MOO Main Page</td>
</tr>
<tr>
<td>4-6</td>
<td>Tufts Wilderness Orientation Main Page</td>
</tr>
<tr>
<td>4-7</td>
<td>Tufts Registration Form</td>
</tr>
<tr>
<td>4-8</td>
<td>Dartmouth’s DOC Trips Main Page</td>
</tr>
<tr>
<td>4-9</td>
<td>DOC Registration Main Page</td>
</tr>
<tr>
<td>4-10</td>
<td>DOC Registration Part I</td>
</tr>
<tr>
<td>4-11</td>
<td>DOC Registration Confirmation</td>
</tr>
<tr>
<td>5-1</td>
<td>Organizational Overview of System</td>
</tr>
<tr>
<td>5-2</td>
<td>Overview of First-Year Section</td>
</tr>
<tr>
<td>5-3</td>
<td>Breakdown of Application Page</td>
</tr>
<tr>
<td>5-4</td>
<td>First-Year Students General Information, COOT Application Section I</td>
</tr>
<tr>
<td>5-5</td>
<td>First-Year Students Emergency Contact Information, COOT Application Section II</td>
</tr>
<tr>
<td>5-6</td>
<td>First-Year Students Medical Information, COOT Application Section III</td>
</tr>
<tr>
<td>5-7</td>
<td>Experience, Application Section IV</td>
</tr>
<tr>
<td>5-8</td>
<td>Trip Selection, COOT Application Section V</td>
</tr>
<tr>
<td>5-9</td>
<td>Overview of Leader Section</td>
</tr>
<tr>
<td>5-10</td>
<td>COOT Leader Application, page 1</td>
</tr>
<tr>
<td>5-11</td>
<td>COOT Leader Application, page 2</td>
</tr>
<tr>
<td>5-12</td>
<td>COOT Leader Application, page 3</td>
</tr>
<tr>
<td>5-13</td>
<td>COOT Leader Application, page 4</td>
</tr>
<tr>
<td>5-14</td>
<td>COOT Leader Summer Contact Info</td>
</tr>
<tr>
<td>5-15</td>
<td>Overview of Administration Section</td>
</tr>
<tr>
<td>5-16</td>
<td>Administration Tasks with First-Year Students</td>
</tr>
<tr>
<td>5-17</td>
<td>Administration Page to Enroll First-Year Students in COOT</td>
</tr>
<tr>
<td>5-18</td>
<td>Administration Page to Search for First-Year Student</td>
</tr>
<tr>
<td>5-19</td>
<td>Administration Page to Assign First-Year Student to COOT Trip</td>
</tr>
<tr>
<td>5-20</td>
<td>Administration Page to Modify First-Year Student Info, Step 1</td>
</tr>
<tr>
<td>5-21</td>
<td>Administration Page to Modify First-Year Student Info, Step 2</td>
</tr>
<tr>
<td>5-22</td>
<td>Administration Page to Modify First-Year Student Info, Step 3</td>
</tr>
<tr>
<td>5-23</td>
<td>Administration Tasks with COOT Leaders</td>
</tr>
</tbody>
</table>
Chapter 1 - Introduction

The goal of this project was to combine two different fields within computer science to produce a system that would be beneficial to a wide variety of people. The first area, user-centered design, studies how users interact with an interface and the motivation behind the decisions they make when interacting with the interface. The second area, databases, explores different techniques for storing and retrieving large amounts of data.

My first experience with databases was the Introduction to Databases class in the fall of my sophomore year. In this class, I was introduced to the relational database and its widespread use in society. I soon realized that databases were prevalent everywhere, yet building a “good” one took a lot of careful planning. As a result of this class, I began to seek out different databases in various areas of my life (work, student groups, etc) that I could enhance or explore to get a deeper understanding of their purpose and functionality.

User-centered design, on the other hand, was a part of the computer science field that I did not realize existed until just over a year ago. It was an approach to designing interfaces that made complete sense once understanding the motivation behind it. Reading about the results of previous user tests was fascinating and I was eager to try the same process.

Last spring, as part of my User-Centered Design class, I began reworking the Colby College Orientation database used to manage the orientation trips that first year students are sent out. The database we reviewed was used by the administrators and coordinators to file the information sent in by first-year students as well as to keep track of the trip leaders and the specifics of each trip.

Our group focused on beginning to redesign the coordinators system to ensure that the data entry went more smoothly. We began by interviewing the users and investigating their different tasks. From this information, we were able to create prototypes and then evaluate them with the users. By the end of the user-centered design process, we had a prototype that showed improvement in a number of different areas for the users. Data entry time had decreased, reports were generated more efficiently, and the user’s overall satisfaction with the system had increased.

At the end of the class, the project was far from finished. I was interested in further pursuing expansion of the system, more user testing, and then taking it to the final step and
actually working on developing the system. This project was a perfect culmination of both of my interests in computer science. It also allowed me to gain experience into the software development process. This project presented an opportunity for me to further update my knowledge in the database world, more specifically looking at web-based databases.

Part of the reason I initially chose to do this project was due in part to my work in Student Activities. Throughout the past few years, I have been asked numerous times to assist with technical difficulties with the system. Each year as a new student coordinator is hired, I watch them struggle to learn the system. Working with the current system seems to add a lot of unnecessary stress to their lives as well as require redundant data entry. My hope is that this project will fix many of these issues and aid in accommodating the expansion of the registration system.

This book details the steps I took to create this system. Chapter Two begins by giving a basic overview of the user-centered design field. Chapter Three introduces the background information about databases. The next chapter explains the details of who the users of the system are and the tasks they wish to accomplish with it. Chapter Five looks at developing prototypes and evaluating them based off of the information gathered in the previous chapter. The process and choices made when designing the database are detailed in Chapter Six. Finally, Chapter Seven summarizes what was learned throughout this process and is followed by the details of the work that could still be done on the system.
Chapter 2 - Background on User-Centered Design

Humans interact with computer interfaces on a daily basis, oftentimes without even recognizing the fact that they are doing so. Anytime a person uses an ATM or checks their voicemail, they are interacting with a different interface. Many times if a person stumbles when using these types of systems they blame themselves. However, it is often not the user’s fault; instead, the system was poorly designed. Here lies the motivation for study of user-centered design.

Originally, user-centered design was not even considered a facet in the computer science field. As the popularity in using computers grew from an esoteric audience to widespread use and the number of tasks accomplished using a computer increased, so did the need for usable systems. Out of this need, grew the study of users and their interactions with the system, now known as user-centered design.

This chapter explores the topics that encompass the user-centered design field. It begins by looking at the different design principles that have evolved over the years as points to keep in mind when designing an interface. These principles can then be applied throughout the user-centered design process of designing, prototyping, and evaluating the interface. The steps involved in each of the user-centered design phases are defined, explaining what each entails as well as its purpose.

2.1 Principles of Design

Design principles are abstractions that have evolved from experience and theory. These principles have become guidelines for designers when initially creating a system. They are not steps to make a perfect interface, rather just important aspects to keep in mind. These design principles should be used to view the different aspects of the interface from various angles. Donald Norman discusses the common ones in his book, The Design of Everyday Things (1988).

2.1.1 Visibility

One of the most important parts of a well-designed system is making sure the features are visible or at least intuitive. Visibility examines whether the user can see how to interact with the system to accomplish a task. For example, a button that is not labeled is not helpful to the user
and can often lead to a huge amount of frustration. However, if the button is clearly labeled “next” than the user will understand the action of the button. By making the actions of your system clearly visible, the user will have a better understanding of how to go about accomplishing the task he/she wishes to complete.

2.1.2 Feedback

Feedback can easily be defined as letting the user know what is going on. This feedback can come in numerous forms. Many times a sound or quick beep will play to indicate a button has been pushed or the button may change colors to indicate which selection has been made. If a user is filling out a form, once the submit button is pushed another page should load to indicate that the form has been successfully submitted. These little clues allow the user to know that some action has occurred and indicates what is happening with the system. Without any feedback, users may continually push the same button or sit and wonder what is happening with the system.

2.1.3 Constraints

There are numerous different kind of constraints in today’s world. Norman categorizes them into the following groups: physical, logical, and cultural. Physical constraints restrict the user from performing an action. For example, trying to fit a key into a door that it does not unlock is a physical constraint. If the user attempts to do this, he/she will find that the key will not fit or turn in the lock. Similarly, when a user goes to plug a mouse into a computer, it can only fit in certain slots. On the other hand, logical constraints rely on the user’s reasoning. If the user is asked for a telephone number, logically the user should only be entering numbers into that field. Lastly, cultural constraints have been learned through society. A simple example would be the red on a sign. Through society, red has been associated with stop signs and danger signs. Therefore, when a user sees the color red a warning flag should be raised in their mind. Often times, it may be a combination of the constraints that prohibit the user from accomplishing a task.
2.1.4 Mapping

When an interface has numerous controls it is helpful for the user to be able to associate each one with a specific effect. This association is known as mapping. A good example of this can be seen by looking at the combination of arrow keys for a remote-control car. In the example below it is far easier to use the left set of arrows than it is use to the arrows on the right. This is due to the fact that the arrows in the diagram on the left have a better intuitive mapping to the direction they go [9].

This design principle can also be applied to web pages. When users are looking for more information, they naturally scroll down. However, if the web page requires them to scroll horizontally it may be possible that they will miss this information.

2.1.5 Consistency

When designing an interface, consistency throughout the system is crucial. If on one page a button does one thing and then on another page a button that looks the same does something completely different, the user will be lost. Now there is no way to predicate what the button might be used for in another part of the interface. By keeping different elements within the system consistent, the user will have an easier time of learning to use the system. This way they do not have to learn each part of the system separately.

Consistency also applies to elements such as menus or the navigation system throughout the system. Menus should be fairly consistent from page to page, keeping similar items grouped together. This same principle should be applied to the navigation system. The navigation should appear consistently in the same place, so that the user can easily locate it no matter where they are in the system. This consistency will once again allow the users to easily learn and use the system without feeling overwhelmed.

2.1.6 Affordances

Norman defines affordances as “providing strong clues to the operation of things” [9]. The theory behind this design principle is that directions or a label will be redundant if
affordance is taken into consideration. Examples of affordance might be a cup handle affords holding, or a door handle affords pushing, or slots afford putting items into them. Affordances should be incorporated and used in a design whenever possible as they are instinctive clues to the user.

2.2 Overview of the UCD Cycle

Once the designer understands the principles and goals of user-centered design, the user-centered design cycle can begin. The cycle is most easily broken up into three phases: the design phase, the prototype phase, and the evaluation phase. These three steps work to identify who the users are and what their needs are, then take those needs and develop ideas of how to achieve them, and finally test these ideas with the users to see what works the best.

Each of the phases is intended to build on one another and be cycled through numerous times. Going through each phase only once will not have the same result because the input learned from the users in the evaluation phase would never be incorporated into the system. As a result of cycling through the phases numerous times, it is easy to identify and fix problems with the interface. Also, by involving the users at numerous different points in the cycle, the designers will have a better understanding of who the users are and what they are looking to achieve. If this process is not followed, there is a strong probability that bad design will occur. Therefore by using this cycle, the users will be more satisfied with the system for it will be tailored to people who are actually using it.
2.3 Design Phase

The design phase is a key part of the user-centered design process. During this phase, the designer learns about the user, the tasks, other products that accomplish similar tasks, the impact the system will have on the user, as well as goals that can be evaluated further in the cycle.

2.3.1 User Analysis

Producing a user analysis is the designer’s first attempt at describing who the users are. “Know the user” is a very basic user-centered design principle but is of vital importance. If the users are not clearly known or defined, the product may be built with the wrong set of people or no one particular in mind, which will result in a system that could be very frustrating to use. Getting to know the users is done by visiting where the user works, interviewing, and/or handing out questionnaires. The approach that is taken depends mainly upon how large and accessible the user group is. If at all possible, it is strongly encouraged to actually visit the user’s work place.

The goal throughout this process is to learn about different individual characteristics of the users. Characteristics include background of the user, motivation for the task, work experience, and work environment. Learning about these qualities will allow the designer to predict any learning difficulties the user may have and compensate for them early on in the process. Also by visiting the user’s work place, the designer can receive clues into integrating features such as sound into the interface. It is possible that sounds may be distracting to the user or other employees around the user. It is also helpful to investigate whether the user uses the system uninterrupted or if he/she is constantly switching between tasks. This will affect how much the designer can rely on the user remembering what they were doing before being interrupted [8].

2.3.2 Task Analysis

Besides learning about the users and their characteristics, it is also important to understand their needs for the system. This includes understanding specifically the tasks they wish to accomplish with the system. It is helpful to have users describe how they currently do the task and then watch them actually perform the task. Using this two-step process compensates for the fact that many times the way a user describes doing a task and the way it is actually done
is completely different. If the opportunity is given, it is also valuable to observe especially effective users of the current system. By watching them interact with the system, the designer may get some hints to incorporate into the new system. Equally important is watching where current users fail or spend excessive time in using the old system. These are areas that should be improved in the new system. Lastly, the way that the user deals with emergencies or exceptional circumstances with the current system should be investigated. This will become an important part of the social impact statement (described in Section 2.3.4) [8].

The functional analysis of the interface often is closely tied with the task analysis. It examines the functionality of the system being developed. In other words, what the system must do in order for the user to complete the tasks that were previously defined. For example, when at an ATM the task may be to withdraw money. The functionality would be accessing your account securely. Another aspect that the functionality analysis will look at is what part of the system should be automated. For example, some larger pizza chains now allow you to order your food online and have it delivered. However, smaller stores do not have an automated system but instead may have their menu and phone number listed online. In many cases automation is dependant upon the resources available to create such an interface or the personal goals of the company. In this case of the smaller pizza store, they might strive for interaction with their customers and feel that ordering over the web has a negative impact on their interaction strategies.

2.3.3 Competitive Analysis

Many times a system will already be in place similar to the one that is being designed. These systems should not be ignored but rather learned from. If possible, a few user-centered design experts should critique the different systems available. These critiques should reflect what works well in the system as well as where the system fails. These critiques can then be used to help design a new system, enhancing or fixing features that need improvement and hopefully further building upon the solid features. This stage of the design process is crucial when designing for a company that is building software for a user to buy. A well designed interface may be the selling point of the software, as most users would like an interface that is simple to use rather than a complex one.
2.3.4 Social Impact Statement

Social impact statements examine the biases that are present in computer systems [5]. This document also addresses many of the concerns of the users and business. From the document, the users and business will be able to see how the current system will be changing and the overall time frame for this change. Changes in job security, training, hardware, and staff are all factors that should be discussed in this document. It is also important to have a plan for the security of the system, if necessary, as well as a plan for backing up the data within the system. Each of these factors should be considered for their impact differs for each project. This is also where any biases towards a diverse audience should be addressed. For example, if there are users who are blind or deaf there needs should be incorporated into this document. Also decisions about the security of information need to be discussed. This entails, who should have access to the information, especially if the information is confidential. Lastly, the development process of the project should be outlined. This will not only keep the users informed but will set deadlines for the designers to follow.

2.3.5 Testable Usability Goals

Testable usability goals should be set before the prototyping phase begins. These specific goals define how to measure the success of the system. Normally, the goals are constructed by looking at the current system, identifying areas of improvement, and then setting a specific goal for this improvement. For example, a testable usability goal might be for a user to enter a certain amount of information into the system in a specific amount of time. Another way to measure the usability of the system might be to count the number of errors per task for the user. By having usability goals, the designers will be able to judge what progress has been made through the development process and where more work is still needed [8].

2.3.6 Concrete Example Tasks

Concrete example tasks are an important part of understanding what the tasks of the system are. They should be very clear and concise stating who the users are and describing the complete job they wish to accomplish. These tasks should not explain how the task is to be completed, so that the interface is not biased before it is constructed. However, the tasks need to detail the information that is received, where the information is going, and what happens with
that information. For example, a receptionist at a college admissions office needs to look up the address and print out a mailing label for Jane Doe in order to send her an application. In addition to Jane's name, the receptionist also has Jane's phone number at hand. This example states concretely who the user is, what they wish to accomplish, and the information that they have available to them. The concrete example tasks can be formed after meeting with the users and gathering information from them. Once they are formed, the concrete example tasks will play a role in developing the prototypes to present to the users [6].

2.4 Prototype Phase

Once the requirements of the user have been gathered, different ideas of how the interface should be designed can be formed. This phase of the user-centered design process is known as prototyping. Prototyping is a valuable tool because it can often save companies a lot of time and money. One of the advantages of testing with prototypes is that they are fast and easy to change. With no implementation behind most of them, new ideas can be easily incorporated and problems found can be easily changed.

The first level of prototyping is known as low-fidelity prototyping. Low-fidelity prototypes lack a significant amount of the detail. There are numerous ways to do this. One way is to draw paper sketches. These sketches are quick drafts of different ways the layout of the interface can be presented. Another method is to do a storyboard. A storyboard is similar to sketches but traces through the users' different movements when completing a task. Using the storyboard approach is good for finding ambiguity in the definition of the tasks.

After low-fidelity prototyping is complete, it is often common to move to high-fidelity prototyping. High-fidelity prototyping resembles the final system much more than low-fidelity prototyping. High-fidelity prototypes have a greater amount of detail and closely resemble the actual interface being designed. When developing a high-fidelity prototype there are two different options. The designer can choose to develop a vertical prototype or a horizontal prototype. A vertical prototype focuses on a few features that are implemented and working. On the other hand, horizontal prototypes have limited functionality but show more of the features. Though no tasks can be accomplished on a horizontal prototype, they allow the user to get the feeling for the interface as a whole with respect to placement of the features. It is often easier for users to give feedback on a high-fidelity system due to the resemblance to the real system.
In both cases, the prototypes should be built off of the requirements gathered in the describing phase of the user-centered design cycle. The low-fidelity prototypes are the most appropriate starting point for developing an interface. They are cheap to produce and can be changed quickly and easily. Once a low-fidelity prototype has been decided upon, then a high-fidelity prototype can be developed from this. As prototypes are evaluated (discussed in Section 2.5), the users’ requirements and problems will become clearer and the prototype will develop to reflect this. With little functionality behind each prototype, changes can be made much more rapidly than changing a fully implemented system. Discovering problems before a program is released is much more cost efficient for a company than discovering the problem afterwards.

2.5 Evaluation Phase

Once a prototype has been constructed, it is important that it undergoes a series of tests or evaluations. Evaluations of the different prototypes are critical because the designer is not the user. Designers often become too close to their system in order to see difficulties that other users may have and can not evaluate whether the interface follows the principles of user-centered design. The user-centered design process places strong emphasis on having the users’ input during all of the phases and this one is especially critical. Even if a designer follows all of the principles of design discussed earlier this does not mean the interface has great usability. These principles are guidelines that should be considered but are not a formula for creating an instant user-friendly interface. Different types of evaluations will bring to light problems that designers have often not even considered.

In the user-centered design process, the first round of evaluations is often a heuristic evaluation. This evaluation asks usability experts, those people familiar with the user-centered design principles and goals, to evaluate the system based upon these principles. There should be between five and ten experts evaluating the system, independently of each other [8]. This number is based upon a study Nielsen did comparing the number of usability experts to the number of usability problems found. The experts might look to see how accurate the correlation is between the system and the real world. They also may review how error messages are given to the user. During the test, they can take notes on the consistency and layout of the system. Since the experts are not actual users, the person running the experiment may need to provide background information specifically about the users or their tasks in order for full
comprehension of the system. Once the expert users are done testing the system, they may gather as a group to discuss their results. However, if the tests are done on an individual basis all at separate times, then each expert user may sit down and share the results with the experimenter and offer suggestions for changes within the system.

It is often good to have the expert users complete the tasks that will be presented to the user. This gives the experimenter an opportunity to assess the testing method to ensure that the directions are clear and accurate. If problems are found, the experimenter can then change them before the actual user testing starts.

Once heuristic evaluations have been completed, the cycle of describing and prototyping will occur again. Changes to the necessary documents and prototypes will occur before the next round of testing. When these changes are complete, the next round of testing often involves user testing. Selecting users to complete the test can often be very difficult. When designing for a large user group, the selected users need to be what is considered “typical” users and not expert users of the current system that is in place. Having users biased or chosen inaccurately can affect the reliability of the test results.

Ideally, the user test will be complete by two to three users representing the user group [8]. This number of users is calculated by Nielson measured by studying the benefits and costs of having a different number of users test an interface. These users will be asked to complete one or two tasks with the system that model the tasks they would use the system for. There are numerous ways in which to observe the experiment. It is possible to use a video camera and tape the entire evaluation but this may make some users uncomfortable. The people running the test may choose to have two people observing and one person running the test. The people observing would be responsible for taking notes about the areas the user had problems in and what seemed to work well for the user. Many times a “thinking aloud” approach is taken, which involves the user talking through the decision he/she is making when completing the task. Once the tasks are completed, it is often beneficial to follow up with a questionnaire or interview to debrief and receive some feedback from the user.

Much like the heuristic evaluation, the results from the user evaluation will then be used to update and change the prototype and description of the user. The cycle may need to be repeated a few more times before an adequate interface is created. However, it is important that user testing is not overdone so that the users do not become tired of testing. Users that test the
same interface a couple of times often begin to adapt to the system. This makes them less likely to act on instinct and base their comments more upon past testing experiences. For this reason, if money and time allows, it is beneficial to bring in a new set of users for the next user evaluation. Even if only one user test is completed, the amount learned from this test will be far more beneficial than creating the system without any input from the users. Fully understanding the users and incorporating them into the user-centered design cycle allows for a much more usable system to be developed that users will happier with.
Databases can be seen everywhere you look today. They are found in libraries, hospitals, colleges, and even in many homes. They have become the de facto way to store data. Originally they began by storing primarily text, such as the card catalogue system in the library. Today, they also store multimedia files such as maps, weather data, and satellite images.

A relational database stores all of the information inside tables. These tables are made up of rows and columns. Each row is an individual entry into the database that holds one value for each column. A row can simply be thought of as a single record in the database. If the database has multiple tables these tables may be related by columns that hold similar values.

It is very important to look at the way the databases are organized and constructed. This construction of tables, which store the data within the relational model, will affect the efficiency of data storage and retrieval within the database. A well organized and defined database facilitates efficient information storage and retrieval.

This chapter discusses one modeling technique for building a database in Section 3.1. Section 3.2 examines the steps in taking this model and creating a relational database from it. Then, the process for normalizing these tables is looked at. Finally, the different pieces of software that are used to implement this system are studied.

3.1 Entity-Relationship Model

The first step in creating such a database is to look at the information in a high-level abstract manner. This can be done by creating a conceptual model known as an entity-relationship (ER) model or a unified modeling language (UML) diagram. For this project, an ER model was chosen as the type of conceptual model to build. An ER model looks at the data as well as the associations between different types of data. Once these groups have been defined, a formal ER diagram can be constructed. The following sections define the different parts of an entity-relationship model.

3.1.1 Entity

An entity can be viewed as an object, physical or conceptual [4]. For example, a car, house, company, or job can all be viewed as different entities. Each of these objects can be
uniquely identified through different characteristics. These characteristics, or properties, are known as the attributes. For example, if you have an employee entity, some attributes might be first name, last name, or address.

3.1.2 Attributes

An attribute can be classified according to three different categories: simple or composite, single-valued or multi-valued, and stored or derived. A simple attribute is one that only has one component. In other words, it cannot be subdivided or broken down into smaller pieces. A composite attribute, however, can be subdivided. For example, consider an attribute for storing a full address; this could be broken down into the following simple attributes: street, city, state, and zip code. The type of attribute that is used, simple or composite, is dependent on what the database is designed for [4].

A single-valued attribute is one in which only one value is possible, like first name. On the other hand, a multi-valued attribute can hold multiple values, such as college degrees for a particular person. Multi-valued attributes may have a limitation on the minimum and maximum entries that are expected within the attribute.

Derived attributes are values that can be determined from stored attributes. Stored attributes, such as birth date, is static information. From this value, it is possible to derive an attribute such as age. Another example of a derived attribute is counting the number of occurrences of a staff entity within a staff table. So, derived attributes can be calculated from one or more stored attributes.

3.1.3 Entity Set

With each entity now clearly defined by numerous different attributes, it is possible to group related entities together to form an entity set. Each entity within an entity set is a unique object but is described using the same properties as other entities in the set. To uniquely define each entity a key must be defined. The key is the value (or set of values) that distinctively identifies each entity within a set. The minimum number of attributes needed to uniquely identify an entity is called a candidate key. For example, if a bank has different branch numbers which are all unique, these branch numbers could form the candidate key. If there are numerous candidate keys, one must be chosen to be the primary key. In a case where numerous candidate
keys are available, the designer can chose a primary key based upon attribute length and future
certainty of the data. In some cases, more than one attribute will be needed in order to uniquely
identify an attribute. When two or more attributes are needed, they form a composite key. Keys
will play an important part later on when designing the specific tables [3].

An entity set can also be classified into two different categories: strong or weak. A
strong entity set is one that can stand on its own. This independence is determined by whether or
not the entity set can uniquely identify itself. On the other hand, a weak entity set, A, is
dependent on another entity set, B. For a weak entity set, A, there is no way to define a primary
key without looking at the entity set, B [3].

3.1.4 Relationships

Once the entity sets are all clearly defined, the associations between them can be more
closely looked at. These associations are known as relationships. These relationships are often
formed when one entity set refers to another entity set. For example, a manager entity could
refer to an employee entity to get specific information about the manager.

Relationships are further defined by their cardinality. Cardinality defines the maximum
number of entity associations that exists between the two groups. The different cardinalities can
be broken up into three groups: one-to-one relationships, one-to-many relationships, and many-
to-many relationships. Cardinality captures a real-world constraint that may apply to entity
associations. These constraints are often made by the user or a business in accordance with their
ultimate goals when working with the data [3].

One-to-One Relationships (1:1)

![Figure 3-1 - Example of 1:1 Relationship](image)
A one-to-one relationship is the simplest form of the different entity relationships. In this kind of relationship, an entity in one entity set is related to only one entity in another entity set and vice versa. For example, in Figure 3-1 there are two entities, a person entity and a social security number entity. They are connected by a relationship called possess. Each entity, or person, within the person entity set can possess only one social security number. The converse is also true. This is shown in Figure 3-1 by the line that connects the person entity to the social security entity. Each person maps directly to one social security number. This type of cardinality can also be written in short form as 1:1.

One-to-Many Relationships (1:*)

Slightly more complex than a one-to-one relationship is a one-to-many relationship. A one-to-many-relationship, also written as 1:* or 1:N, is one where one entity is related to numerous entities in another entity set but each of these is linked back to only that one entity. This can be seen in Figure 3-2 where there are two entities, professor and course. These entities are related by a relationship titled teaches. The first professor, entity A, teaches only one course, 101. However, this is not the case with the second entity, B. In this case, the entity is related to two courses, 102 and 103. This shows that an entity within the professor entity set is allowed to teach more than one course but each course can only have one professor.

The asterisk in the notation may be replaced with a number if a maximum value is known. If in the example above, it was defined that a professor was not allowed to teach more than five classes the relationship would be defined as 1:5 (one-to-five). It may also be the case
that a professor is not required to teach any courses at all. In this instance, the relationship would be defined as 0:* or zero-to-many, assuming there is no maximum number of courses. This cardinality, or restrictions, would be defined specifically by the interpretation of the data being stored within the database.

**Many-to-Many Relationship (**:*)**

![Figure 3-3 - Example of **:* Relationship](image)

There will often be cases when a relationship does not fit into the one-to-one or one-to-many relationship category. In this case, there is also a many-to-many relationship, known as *:* or M:N. In a many-to-many relationship, the entities in both entity sets are related to numerous other entities in the opposite set.

An example of a many-to-many relationship can be seen in Figure 3-3. In this case, there are two entities, student and class, who are connected by a registers relationship. Student A has registered for classes 101 and 103 and student B has registered for class 103. This shows that each student is allowed to register for multiple classes. The diagram also shows that each class is allowed to have multiple students registered for it. Since both entities can have multiple connections with the other entity set, this example satisfies the definition of a many-to-many relationship.

**3.1.5 Entity-Relationship Diagram**

Now that the elements of an entity-relationship model have been introduced, it is possible to present them in a short hand notation in a diagram. This notation has become standard, so that it is consistent from one ER diagram to the next.
**Entity Symbols**

- Entity
- Weak Entity

**Attributes Symbols**

- Attribute
- Key Attribute
- Multi Valued Attribute
- Composite Attribute
- Derived Attribute

**Relationship Symbols**

Types of Relationships

- One to Many
- Many to Many
- Zero to Many

Figure 3-4 Examples of ER Symbols
3.2 Mapping an ER Diagram to a Relational Database

Once an ER Diagram has been constructed it is fairly easy to convert it into tables for a relational database. Each different symbol shown above plays some role in the construction of tables. These tables will later be tested to see if they uphold the rules of data normalization.

The first step of transforming an ER diagram into a relational database is to take the regular entity types and make them a table. The key of the table will be the key attribute [4]. In Figure 3-5, the Employee entity has become the employee table. According to the ER diagram, Name is a composite attribute. When a composite attribute is present, each attribute that makes up the composite attribute becomes a field in the table. In the example above, LName and FName both become fields in the employee table as well as Sex and SSN.
The next step is to map weak entities to tables. This step is similar to the first step of mapping regular entities to tables. Begin by creating a table with the entity set name and making all of its attributes fields within the table. Since the entity is weak, and therefore dependant on another entity, this dependency needs to be shown. To do this, take the key attribute from the regular entity and make it a field in the weak entity table. This field, in addition to another field in the table, will then become the composite key of the table [4]. This process is shown in Figure 3-6. The dependent entity is a weak entity dependant upon the employee entity. So, the initial table created for the dependent entity consists of all of its attributes as fields. Then, the key attribute from the employee table, SSN, was taken and added as part of the key, noted as EmpSSN. Adding EmpSSN to the dependent table ensures uniqueness among the records in the table.

The different types of relationships may also affect the creation of tables. This, however, depends on the specific database implementation and the relationship that has been defined. Relationships may cause the need for other tables to be created or additional foreign keys to be added within a table. The designer of the database should evaluate the ER diagram and the data with care in order to determine if any of these options are necessary.

3.3 Data Normalization

Once the tables have been constructed, it is important to verify if they follow the rules of data normalization. Data normalization consists of a series of tests upon the data. These tests look at the association between the attributes. The goal of data normalization is to reduce redundancy among the attributes. By upholding the rules of data normalization, data insertion, deletion, and retrieval is ultimately faster and more efficient. Also by reducing the redundancy among the attributes, the space required to store the data will be minimized.

Data normalization has five forms of which the first three are used in practice. The details of these rules are discussed in the remainder of this section. The first normal form is most often required and done instinctively when creating a database. The second and third normal form are still often used, but may be strayed away from depending on the specific database. Forms higher than the third form are used only in rare situations. It is assumed that if a database is in second normal form than it has also met the criteria for first normal form. In this sense, all of the normal forms above the first normal form build upon each other.
Data that has not been normalized is said to be in unnormalized form (UNF) [3]. Since none of the data normalization tests have been done on the attributes, it is assumed that tables in this form may have one or more repeating groups. The first normal form eliminates repeating groups within a table.

In Figure 3-7, the Customer Order table has the repeating field Item number, so that the company can track the items each customer orders. However, the customer is limited to ordering only three items per customer ID. In order to eliminate the repeating rows, the table can be broken down into two separate tables, as seen on the right. This 1NF conversion allows each customer to order any number of items, not just three.

Now that the conversion to first normal form has been accomplished, each table can be tested against second normal form. Second normal form tests that each column must be dependant on the entire primary key [3]. This is important only when a composite key is present in a table. If 2NF is found to be violated, separating the data into two separate tables as shown in Figure 3-7, is often the solution to the problem.
Third normal form addresses transitive dependency. Third normal form ensures that every attribute is dependent on the primary key directly. For example, Figure 3-8 shows a table with the attributes: staffNo, branchNo, bAddress where staffNo is the primary key. The StaffBranch table does not satisfy the 3NF. This is due to the fact that staffNo and branchNo are dependant upon each other and branchNo and bAddress are dependent on each other. However, bAddress and staffNo have no dependency, thus violating the third rule. By separating these fields into two tables, one with staffNo and branchNo, and the other with branchNo and bAddress, the third rule of normalization is satisfied [3].

As mentioned before, normalization above the third form is very rare, though these forms do exist. It is important not to over normalize the tables. This can often lead to difficulties with data retrieval and storage. After normalizing, there may be times when denormalization is necessary in order to make the database store the correct information. This discretion is left up to the database designer, though the rules of data normalization should be upheld whenever possible.

3.4 Implementation

One of the most popular ways to implement a web-based database system today is using a combination of an Apache server with PHP and MySQL as programming languages. This section will look at the roles each of these pieces play in creating the bigger pictures of connecting a database to the web.

The Apache server began in 1995 when many people were becoming discouraged at the performance of web servers. These people decided to take the matter into their own hands, and so created Apache, an open source code software. Within time it grew to become the most popular web server today. Since it has an open source code, numerous developers continually work on the code making it known as one of the most reliable and stable web servers. Lastly, Apache is also cross-platform, making it easily accessible (and free) to all [2].

PHP is a recursive acronym that stands for PHP: Hypertext Preprocessor. PHP was developed to be embedded within HTML. The syntax of PHP is based strongly upon C, but some influences of Perl and Java are also evident. When PHP is embedded within an HTML file, the server reads through the HTML file until it comes across PHP code indicated by a <<?. The server then evaluates the PHP code, which ends with a ?>, and returns the result in HTML to
be displayed. The benefit of using PHP is that it is dynamic code which allows for customization of web pages [1].

PHP is extremely useful for retrieving information from HTML forms. This is done through global PHP variables that pass information from one page to the next. Variables in PHP are not assigned originally assigned a type, rather it is automatically declared once some information has been assigned to it. This makes PHP a very flexible and easy to work with language [1].

When working with databases, MySQL is commonly used in conjunction with PHP. MySQL is an open source database server that implements Structured Query Language (SQL). SQL is the most common language used for retrieving, changing, and storing information from a database. PHP is the middle ground that communicates to both MySQL and HTML. Connecting to the MySQL database and conducting queries can all be done using PHP. The results of these queries are then handled and formatted by PHP, so that they are correctly displayed on the screen [11].

Overall if the three pieces are put together, this is what happens when someone visits a database driven website. Once the URL is typed in the Apache server recognizes the file as a PHP file. The PHP commands connect to the MySQL database and perform the requested queries. The results are returned and stored in PHP variables. The server interprets this PHP, which then results in the HTML that is shown to the user [11]. It easily seen, why these three tools are so powerful due to their powers in creating dynamic web pages as well as interactive web-based database.
Chapter 4 - Details of the Task

This chapter explores the describing phase of the user-centered design process. The proposed COOT system is one that a large group of users will be using. It is easiest to split these users up and group them according to the tasks they wish to perform with the system. This results in three user groups: the administration, the first-year students, and the COOT leaders. When exploring the specifics of the users and their tasks, it is easiest to look at these groups individually and then how they overlap and interact with one another. The following subsections look in depth at the characteristics of each user group, the tasks and goals they need to complete with the system, other orientation systems in the surrounding area, as well as the testable usability goals for the system.

4.1 User Analysis

The Colby Outdoor Orientation Trips (COOT) system is a complex system with many different sections. There are numerous different people who contribute different data to be input and more people who require a variety of data out of the system. Each of these users (or user groups for those who use the database in similar fashion) must be looked at in-depth to understand his/her goals and expectations of the system. An overall goal for the entire system is that it should be a self-explanatory system. This means that there will be no training necessary or reading of any manuals. However, help will be available should the user become stuck or have a question.

4.1.1 Administration

The first user group to look at is the coordinators of the orientation program. This user group consists of deans, two college students, and a secretary. The Assistant Director of Student Activities and one college student will be the primary users of the system for this group. The other college student and secretary will have some use with the system but it will be minimal and at random points. The rest of the deans will have indirect contact with the system, but their needs will need to be taken into consideration when examining the tasks to be completed.

The current dean in charge of the orientation program (Assistant Director of Student Activities) has been working with the program for four years. She understands the program and
its organization well and uses the current system primarily to pull information to give to other sources or to complete other projects. In her job, she interacts daily with a computer and is familiar with basic internet navigation. Many of her day-to-day tasks involve her using e-mail and the internet in order to research or order necessary supplies for students. Therefore, she is comfortable with the basic operations of software such as a web browser (Internet Explorer or Netscape), Eudora, and Microsoft Word.

The student, who works for the dean and is in charge of the program, uses the current system to input all of the information and then also retrieves data out of the system to pass onto other sources. Unlike the dean, this student is not the same person each year. Most years, a new student is hired, so the computer knowledge varies from year to year. In both cases, we can assume that the users are familiar with minimal web browsing and entering data into forms on the web. However, the student may encounter some difficulty as the office computer is a Mac, while most students are more familiar with PCs. Hopefully, this obstacle will be minimized as most web browsers are fairly universal. Since the student is also a Colby student, we can assume he/she has had some exposure to software such as Microsoft Word, as well as an e-mail application. This helps us gauge the computer experience of the user.

The second college student does not work with the program for the entire summer. Therefore, his/her interaction with the system is minimized to a short four-week period. We can assume that he/she has the same experience as the student discussed above. However, since he/she will only be interacting with the system for such a short period of time, it is important that it does not take a week or two in order to learn how to use the system. This would be counterproductive to the program.

The secretary will also have a very minimal role interacting with the system. Since she is not employed during the summer, her time working with the system will also be minimal. The current secretary is very comfortable using computers and has taken numerous computer classes at a local business college. She uses her computer daily for numerous tasks and is familiar with programs such as Microsoft Word, Excel, web browsers, Eudora, as well as the school’s scheduling software. This summer she also worked for ITS. Therefore, she has a good knowledge base about computers and should be able to pick up simple software easily.

Lastly, the rest of the deans and administration staff will not have direct contact with the system. However, they will be requesting information from it, such as emergency contact
information or specific trip information. It is important to keep their needs in mind so that any of the users described above will be able to complete the task for them. Many times they will need the information almost immediately therefore the turn around time needs to be minimized.

4.1.2 COOT Leaders

The next user group is the COOT leaders. COOT leaders are Colby students who are selected to lead the different orientation trip. Each year roughly a hundred students are picked. Since it is such a large group it is also hard to categorize the “typical” users. However, there are some assumptions that can be made about this group. Since all of the users are Colby students, we can assume that they have had exposure to using the internet, and most are probably fairly comfortable with it. Since there are numerous computer labs spread throughout campus, COOT leaders are also probably familiar with software such as Microsoft Word and at least one type of e-mail application. They should be fairly comfortable using a computer, though once again there level of comfort between a Mac and PC may differ. However unlike the coordinators, COOT leaders will not be using a specific computer to complete their task. They will be free to choose whatever computer they feel most comfortable with. This will work to our advantage when designing the system because we do not need to worry about people being placed in an uncomfortable or unfamiliar environment. COOT leaders will also have very little interaction with the system and it may not all be at the same time. For example, they may use the system in the spring to enter information about themselves but then not use it again until the middle of the summer to see specific trip information. This large gap of time will affect the memorability of the system and is an important factor to consider in the design phase.

4.1.3 First-Year Students

The third user group is perhaps the hardest to describe. This user group consists of the people who will be using the system to register for orientation trips. We can not assume that it is only the first year students who will be registering for the trips, because many times parents or other family members may do it for the first-year students. This broadens our user group lowering our expectation of experience with the internet.

We will need to consider three classifications for this user group. The first group we will call the novice group. These users are unfamiliar with the internet and maybe computers in
general. Typically they will have had up to three months of use on a computer and exposure to the internet and web browsing.

The intermediate users will have had a bit more computer usage. These users will have used a computer anywhere from three months to up to a year. They will have had roughly this much exposure to the internet. By this time most users are comfortable with the internet and for basic websites they should be able to maneuver through them fairly easily.

The experienced users of this group will have had a lot of use with computers and particularly with the internet. They will be able to navigate websites and find the information they need fairly easily. They will have had exposure to a variety of online forms as well as many different web site designs. The difference between the experienced and novice users is substantial. We may assume that those falling in the novice group are still a little leery of online registration and might opt for the paper form. However, when building the interface this user group must be at the forefront of the design. For if novice users can easily use a web site, most likely intermediate and experienced users will be able to as well.

For this group of users, the primary goal will be to research the different trips offered and then register for orientation. Users may decide to return to this site once trip assignment have been made, to see what trip they are assigned to, as well as get specific information about equipment that is necessary. Visits to the web site can vary from someone who registers quickly and never returns, to someone who continually checks back for new news. Because of the number of different types of users in this group, this section will be the hardest to test as well as design for.

In each of the user groups, we want the user interface to be simple and as self explanatory as possible. In most cases, the user will be using it only a few times and will not need to have constant interaction with it. No training should be necessary though help should be easily accessible if the user is still confused by the options presented. We can also assume the atmosphere the user uses the system in can be quite variable. Some people may do it in the quiet of their own home while others (such as the coordinators) will be in the midst of an active office full of distractions. These factors emphasize the need for simplicity and the ability to have the user shift his/her attention from the task at hand and be able to easily regain his/her place without starting over.
The number of users who are familiar with the old way are a minority in the entire user pool. Primarily this will affect only the present Student Activities deans and secretary who have done registration the old way for the last couple of years. However, first-year students and parents may have familiarity with other online registration programs, so it is best to keep it consistent with other forms on the web. This consistency should help the ease of use of the program. Consistency throughout each section of the system (one section per user group) will be especially helpful for identifying when users stray into other sections. However, sections should also be tied together so that the system does not appear chaotic. Rather the changes in the sections should be subtle but noticeable. These small hints in addition to a solid user interface will help guide the user in the right direction. The user should walk away from the system feeling as if he/she successfully completed the task at hand with little to no aggravation.

4.2 Task Analysis

In order to fully understand what the new COOT system is required to do we must take a look at the tasks that each of the users (or user groups) wish to complete. These tasks must be analyzed by looking at the current way they are accomplished and then deciding how to improve it. After the tasks are identified, testable usability goals will need to be set to see if the new system meets the standards (goals) that have been set.

Though none of the user groups will be using the system to complete the same task, the information one user group enters will often be displayed for another user group’s output or manipulation. With this said, the system must carefully be designed so that the information is in one central location, which will help maintain the consistency of the data. For example, one of the old COOT systems was split into two databases. These two databases had some overlapping fields but often times one database would be updated and the other would not. This led to decisions being made based on inconsistent data. By keeping data in a central location, the new system will prevent this from happening.

4.2.1 Administration

The coordinators will be primarily using the system to assign trips to each student. They will need to view the information each first-year student has entered, review their trip choices, and then assign them a trip. They will also need to have access to the information about the
different leaders so that they can assign them to lead particular trips as well. In addition to manipulating the data, the coordinators will need to retrieve information such as all of the first-year students who are on a particular trip, and a list of all of the current leaders.

Information regarding actual trips, such as campsite information, equipment information, arrival and departure times will also be entered by the coordinators. This database is their primary way for organizing all of the little details about trips. Any confirmations or deposits sent to campsites or companies (for example canoe rentals) will also need to be noted. Information regarding travel (travel time, mode of transportation, etc) is also associated with each trip. Lastly, information regarding supplies for each trip must be kept track of. These supplies include items such as stoves, number of tents, whistles, medkits, maps, etc.

The functionality of a search feature for the coordinators is one that will be extremely helpful. The administration needs to be able to search by any sort of data on first-year and COOT leaders. This allows them to quickly access data without going through each entry or flipping through paper forms. Another addition to the system would be allowing the coordinators to search through wait-listed COOT leaders. These are COOT leaders who were not chosen in the first round but are substitutes in case a chosen COOT leader is unable to fulfill the position. Being able to search these records will alleviate time for the coordinators by eliminating the need to find the original COOT applications and reading through them. Instead, they can search based upon what kind of trip leader they are looking for.

The current system for the coordinators was revamped about a year ago as part of my CS352 project. Though the system that was tested with the users was never implemented, a similar system, which was not web based, was utilized over the summer. The system which is currently in use is based in Filemaker and the file is then shared to the appropriate computers. Though the system is an improvement over past systems, the networking of computers is often difficult and prevents easy access to the information at any location other than a few computers in the Student Activities Office.

4.2.2 COOT Leaders

COOT leaders will initially use the system to apply to be a COOT leader. The application asks for person information (name, box, phone) and then goes into a series of questions. These questions are used to judge the interest of the student applying to be a COOT leader.
leader. At the end of the application, the student is asked to rate the different types of COOT trips that are offered, so that the coordinators can get a general sense of what trip the applicant is interested in leading.

The application at the present is a paper application that students must go to the Student Activities Office to get. There is a period of roughly two weeks when applications can be picked up, filled out, and returned. A majority of the applications that are returned are typed. The application is then used as a guide in the interviewing and finally choosing the COOT leaders for the upcoming year.

Once chosen as a COOT leader, the student will need to enter the system to enter summer contact information. This information is used by the coordinators to do appropriate mailings over the summer. Once this information is entered, COOT leaders will primarily use the system to pull data from the system. This information can include things such as first-year students assigned to their trip, their e-mail address, and dorm assignment once on campus. Another valuable piece of information is a list of supplies that each first-year student is sent for each trip. This differs from trip to trip and would not change from year to year, so would be an addition to the COOT website, rather than the system.

Currently, the COOT leaders fill out their summer contact information by going to the Student Activities Office and signing a piece of paper. This once again creates extra paperwork for the COOT coordinator who then takes the information and enters it into the system. The new system will allow COOT leaders to update their information if it changes without going through the hassle of finding the COOT coordinator.

When the COOT leaders use the system, they enter their preference for which trips they would like to lead. It is possible to create a system that can try to assign COOT leaders to trips based upon their preference once they have been chosen. However, upon review of how COOT leaders are chosen by the committee this does not seem to be an item of the system that would benefit from being automated. The committee chooses COOT leaders after having interviewed them and understanding their outdoor skills. When making decisions, the committee picks leaders with a specific trip in mind, not just a general group of students.
4.2.3 First-Year Students

The first-year students and families will be using the system primarily to register for a COOT trip. This will be most likely a one-time interaction with system. They will be asked for information such as the student's address, trip choices (top three), allergies, and emergency contact information. In return, users should receive a confirmation that the information has been successfully submitted and once the information has been processed, they should receive notification of which trip they are assigned to.

Once trip assignments have been made, students should be able to return to the site to gather more information about other first-year students assigned to the trip, trip leaders, and equipment required for the trip. Information regarding other trip members (first-year students) will be fairly basic, such as student's name and Colby e-mail address. Since some first-year students may not wish for this information to be shared, the registration form will have an option to opt-in or opt-out of this feature.

First-year students have been registering via mail in the past. Once they have been accepted to Colby, they receive a packet detailing all of the trips. They are then asked to return the sign-up form as well as the health form indicating their first, second, and third choice trips. Since trips are filled on a first-come, first-served basis it is easy for the coordinators to place the students on the appropriate trip.

Automation might also be possible for assigning first-year students to specific trips. Unlike COOT leaders, every first-year student who registers for COOT will get a place on a trip. The rules for assigning students to trips are much more structured, as there can only be a certain number of students per trip, and first-year students rank their top three choices. However, after talking with the coordinators about this option, they were very hesitant. Assigning first-year students by hand to a trip allow them to review all of their information and make sure the student is suited for the correct trip. Therefore, this task seems best not automated for the time being.

These decisions about the functionality of the system are made from the information gathered in the user and task analysis and with input from the user where necessary. While many of the proposed new functionalities originally seemed like a good idea, after further analysis they seemed to complicate the task more than necessary. However, it is good to examine the functionality behind the tasks in order to find areas of improvement. The functionality of the system will evolve over time as the system and its users grow and develop.
Since the tasks differ from group to group, the testable tasks will also differ from group to group. Some groups will require more testing than others and will also be easier to test. For example, the coordinator's group will be easy to test since the staff member is consistent from year to year unless he/she leaves the job. On the other hand, COOT leaders change every year, so students who were leaders last year will not necessarily be COOT leaders in the future.

To test the success of the system with the coordinator's group (i.e. COOT coordinator, Associate Dean of Student Activities, and other staff members of the Student Activities office), the Associate Dean of Student Activities will be asked to navigate through the system to accomplish certain tasks. These tasks include:

- Retrieve and print a list of the first-year students, the trip they are on, and their COOT leaders within two minutes
- Locate a student's personal information (address, phone number, trip preferences, and medical history) within two minutes
- Enter campsite information for a specific trip within two minutes
- Find a specific trip and add confirmation about the campsite within a minute
- Enter a student who registered via mail into the system within five minutes

It may also be worthwhile to have the COOT coordinator from the past year complete these tasks. Since she has already had experience with the previous system, these tasks might be easier for her to complete. However, it would be good to have input from someone with similar experience (age and schooling) as the COOT coordinator will have.

COOT leaders will also be fairly easy to test. Though we can not predict who will be chosen to lead a trip in the future, we can use past leaders as a basis for testing. These students will be chosen at random and will be asked to participate in minor testing. It might also be beneficial to test different leaders for different tasks of the system that will be used by COOT leaders. The first task to be tested will be having a COOT leader enter his/her summer contact info in less than five minutes. The user will be directed to the general COOT website and then asked to find his/her way to enter information. Since this task will only be completed once (not reoccurring), it is important that navigation to the correct page is simple and quick. Once at the correct page, the following information will be requested:

- Name (First, Last, and Nickname)
- Summer Address (City, State, Zip)
The second task will be to find a list of the first-year students assigned to the leader's trip. This task is more of a navigation test and should not take any longer than three minutes especially since there should be some familiarity with the site from entering summer contact information months earlier. Since these tasks will not be completed at the same time in actuality, it would be ideal if they were not tested together.

The first-year students and their families will be the hardest group to test. Since we will not know who is accepted we can not test them directly. It is possible to test current first-year students or just other family members at random. The system should be geared so that anyone over the age of 16 should be able to complete the following tasks. The first task is to register for orientation. The following information will be requested from the user:

- Name (First, Last, and Nickname)
- Address (City, State, Zip)
- Home Telephone Number
- Emergency Contact (Two)
- Date of Birth
- Gender
- Medical Notes/Allergies
- Trip Choices (Top 3)
- An option to opt-in or opt-out of listing their name and Colby e-mail address for other first-year students assigned to the same trip to see.

The second task pertains only to first-year students which will ask them to use the site to find information about the other members on their trip, their name, e-mail address, and leader's contact information. Both of these tasks should be completed in a reasonable short time. Navigating to and filling out the registration form should take no longer than 20 minutes. However, this time might be slightly higher or lower depending on how readily available information such as emergency contacts is. The second task of finding out about other trip members should take no more than five minutes. For in this case, it should simply be the process of navigating the right spot on the website.
It is important to note that COOT leaders and first year students or their families must be chosen at random in order to eliminate biases that may occur when testing. By choosing them at random a better picture of the user interaction with the system will be formed.

4.3 Concrete Example Tasks

With the users and their tasks clearly investigated and defined above, it is now possible to form concrete example tasks. These tasks define specifically what the user is looking to do with the system. They will be used when forming a prototype and in evaluating the success of the system in addition to the testable usability goals. The following concrete example tasks were made for the COOT system:

1. Elizabeth Smith enters the system to register for COOT. She enters personal information about herself (name, address, phone), emergency contact information for two separate people (name, relationship, and phone number), and then picks three types of COOT trips in which she would like to participate. However, before picking her trips, she learns about the various options.

2. Ned White enters the system to apply to be a COOT leader. He fills in his personal information (name, campus box, phone) as well as answering a series of questions pertaining to his outdoor qualifications and why he is interested in becoming a COOT leader. Finally, he indicates which type of trip he is interested in leading.

3. Ned White now chosen to be a COOT leader enter the system to accept the position and to provide summer contact information (mailing address and phone number) before leaving school in May.

4. Julie Brown, the student coordinator, enters specific information pertaining to each COOT trip into the system during the summer. This information entails details such as the type of transportation, time of departure from Colby, time of departure to return to Colby, campsite addresses for three nights, campsite confirmations for three nights, and details regarding the number of stoves, tents, canoes, paddles, etc that are necessary for each trip.

5. Julie Brown, student coordinator, receives an application for a first-year student in the mail. The application has specific details about the first-year student's
personal information, trip choices, outdoor experience, and two emergency contacts. Julie must then enter this information into the system so that all records are in the same central location.

6. Leanne Burnham, director of COOT, enters the system to assign a COOT leader, Ted Varley to lead the trip Moxie Bald A.

7. Leanne Burnham, director of COOT, enters the system to obtain a list of all of the first-year students as well as COOT leaders for Acadia Exploration A. She wishes to take this print out and post it on the bulletin board in Cotter Union.

4.4 Social Impact Statement

The Colby Outdoor Orientation Trips is organized by the Student Activities Office. They hire a student coordinator each year to organize the fifty plus trips, the five hundred incoming students, and hundred trip leaders. Currently the database storing all of this information is used by two people. These two people, the student coordinator and Associate Director of Student Activities, enter all of the data pertaining to the trips, first-year students, and trip leaders as it is received. The primary goal of the system is to expand it to a larger user base. The system was expanded to allow first-year students to register online and allow COOT leaders to enter their contact information online as well. This reduced the amount of data entry the student coordinator and Associate Director of Student Activities will have to do. A secondary goal was to create an easy learnable system so that users who interact with the system once can do so with ease. The last goal was to create a system that would make organizing COOT trips easier and more efficient.

For the COOT system the primary stakeholders are the coordinators of the COOT trips. Other people such as first-year students and families, and COOT leaders will also be affected directly by the new system. The information the first-year students or their families enter into the system will directly affect which orientation trip the coordinators assign them to. The new system will also indirectly affect other groups on campus such as the Dean of Students office and other college staff, who often ask for information stored within the system.

The benefits of the system include faster performance, reduced errors, and increased user satisfaction. The coordinators will be able to assign students to trips faster and with fewer errors using the new system. The new system would also require the coordinators to spend less time on
data entry, which allowed them to spend more time planning the itinerary for each trip. First-year students would be able to register quickly online and COOT leaders would have access to current information constantly.

Since the new system would alleviate much of the need for manual data entry, the job of the student coordinator may change to reflect this. However, the position will not be eliminated, rather the focus of the work maybe changed to another area of the trip. It will still be important to have this person to enter any registrations that are received via mail, to answer concerns and questions about the program, and to assign students to appropriate trips. The tasks of the other users of the system will be similar to the past with the major change being that they are done electronically instead of on paper.

With information such as medical history being stored in the database, security and privacy issues will be a major concern for the system. Since the database has the possibility of growing fairly large, it is important that we do not store unnecessary information. The data that is stored must then be password protected, encrypted, and the system should monitor the logins. By using password protection, there will be greater control over the information the user is allowed to view. An integration approach with CARS (Colby’s authentication system) might be one approach. With CARS current (including first-year enrolled students) students and staff all have user names and passwords. This would eliminate the need to create separate user logins specifically for this system.

The system should also be monitored for excessive logins by inappropriate people. This might be added to the responsibilities of the coordinators or the Associate Dean of Student Activities as they will be the ones who will have full access to all of the data. The monitoring of the system is another way the data will be protected to help ensure privacy.

Given the way the COOT program is structured, a centralized system seems to be the best approach for designing our systems. Though many users will interact with the system, only a few (the coordinators) will be using the data in the system to make decisions. For this reason, there is no need to burden the other users with additional responsibilities of backing up the system. This would create unnecessary confusion when interacting with the system. Instead, the coordinators will be responsible for backing up the system at their discretion. During the summer months, when the system is heavily used, the database may need to be backed up daily. However, during the winter months, the database may only need to be backed up once a month.
Certain data within the system may cause the coordinators to make biased judgments. For example, if a student signs up for a particular trip and spaces are available he/she should be given a spot. If the coordinators look at the student’s medical history, they may think that this trip is inappropriate for the student. Though this judgment is biased, it is important for the coordinators to make these decisions in order to ensure the safety of all of the students on the trip.

Another potential bias was the software, computer system, and screen size that the system is run on. Since this system was designed to be web based it can be viewed with different programs such as Netscape Navigator, Microsoft Internet Explorer, or Mozilla. This in combination with the screen size could change how the pages are viewed. Though this was taken into consideration when designing the pages, there was no way to eliminate all of these changes.

The order of the data on the forms that first-year students and COOT leaders fill out was also be a bias. It was important to watch the order of the fields, so that people were not turned away from the system based on what they originally see. It was important that the fields had a flow to them, making them easier to fill out, and eliminating unnecessary confusion.

As was mentioned earlier, individual rights were an issue that must be addressed within the scope of this project. While password protection was one way to ensure privacy, it was also important to thoroughly test the system to ensure that no one has access to information that he/she should not have. Password protecting the system would also help ensure the data quality of the system.

The current implementation of the COOT system has very little accommodation for people with handicaps such as sight loss. Since the new COOT system is expanding its user base by an enormous amount, this also expands the possibility of a handicap person using the system. While it is important to consider how these people will be able to interact with the system, additions or modifications to the system based on their needs will be made, time permitting. The concern may also be raised for those families living in rural areas, who do not have internet access to register for COOT or for those people uncomfortable registering online. In this case, these families and students may register the old way via the mail.

The coordinators section of the current program was recently revamped as part of my CS352 class last spring. It is important to use these results to help design the layout of the new
system for the coordinators. The organization of the fields and input forms work well for the
coordinators, so there is no need to change them. However, for the other users it is a bit harder
to understand what will work well since nothing is in place besides paper forms. Because of this,
the development of the interface will be kept simple yet still meet the needs of the users.

The new COOT system was currently in late requirements gathering phase and early
design phase in December. Basic prototypes (with little to no features) were developed by the
end of January. The first round of user testing was scheduled to be done during January into the
first few weeks of February. After these results were gathered, full implementation was to begin
and carry through the next few months. Once the majority of the site had been implemented,
another round of user testing was to be conducted before making the final changes. The
completion date was set for the beginning of May.

Decisions were made primarily by the designer. However, these decisions were based on
input from the users, those who test the system, as well as computer science faculty at Colby. Input was received from potential users via the user testing and these comments and problems
were taken into consideration to change the layout or functionality of the system.

The primary stakeholders in the system, the coordinators, were asked to participate in two
or more user testing sessions. The goal of these testing sessions was to receive input about
oversights that may have been made when designing the system, as well as input on what works
and does not work in comparison with the current system. Other users were randomly chosen to
try out the other sections of the system (the registration and COOT leader section). Though
users were asked at random, participation is voluntary.

Since the goal of the current COOT system was to create an easy learnable system, the
new system will adopt this goal as well. Since there is minimal amount of training provided on
the system, there will be no required training to use the new system. A manual will be provided
outlining how to do specific common tasks for each of the groups, which will help eliminate any
confusion. The system can be used on any computer that is connected to the internet. This will
allow users to work in an environment where they are comfortable with the setting and the
hardware.

As discussed earlier, routine backups are essential when the system is being used
constantly (during registration time). Even with the constant backups, a long-term plan needs to
be put in place in cause of long-term failure. The option of having a paper back up system
should also be explored, so that nothing registration can continue as normal while the system is being fixed. This may mean that once a registration is received a copy of the registration is emailed to the student coordinator, who then prints it and files it. These options will need to be further discussed with all of the coordinators to figure out the best plan.

Since the current way of registration (paper forms) can not be completely phased out (due to those who do not have access to a computer or the internet) there will be an overlap of systems. Since paper forms will still be mailed to all families, if there is a problem with the online registration, families can be instructed to fill out the paper form. The major change will be for the coordinators who will go from using Filemaker to using a web based interface to manage the trips. These two systems will not be able to overlap as the data needs to be stored all in one spot. The old system should still be kept in because the new system fails during migration or data from the past years needs to be viewed. The best time to migrate to the new system would be in October, a few weeks after the trips return. This will allow a few months for the coordinators to become familiar with the system. Then in December and January, Winter COOT (orientation for students coming to Colby in January) registration can begin. These orientation trips are much smaller and are only geared to about forty students. This small testing should be enough to identify any major bugs in the system and repair them before hundreds of people try to use the system from May-August.

With such a wide base of users, it may be difficult to measure the success of the new system. Having COOT leaders and those using the system to register fill out a questionnaire might be one possible way. However, in this case the comments received might be only from those who had problems with the system or who particularly enjoyed the system. A questionnaire might be appropriate for the coordinators, since it will be easy to have each one of them complete one. Another statistic to look at might be the number of people who used the system to register online instead of using the paper form. All of these, in addition to comments received, can be taken into consideration to determine the success of the system.

In conclusion, the new system broadens the user base of the COOT database which in turn affects how the program is organized. While these modifications will not change how the trips are run, they will change the interaction of many people with the system. The new system will create better organization and efficiency when organizing COOT trips.
4.5 Competitive Analysis

A good way to receive insight into designing a system is to look at other similar systems that are currently put into place. These systems will often be able to reflect good and bad qualities that you want to the system you are designing to contain or not contain. In order to do this, I reviewed ten schools in the New England area who had similar programs such as the Colby Outdoor Orientation program. While not all of them had online orientation, it was helpful to look at the layout of their site and see the information that they provided.

The first school explored was Bates. It was rather disappointing to see that Bates had very little online about orientation. There was a specific site devoted to it, but it only had the schedule of orientation events. While this proved helpful, it had nothing pertaining to any outdoor orientation trips. However, there were numerous articles about the outdoor orientation trips Bates ran, and the outing club had a short blurb on it. All of this information, which was very scarce, was hard to come by.

Similar to Bates, the Bowdoin outdoor orientation program is hosted by the outing club. So while there is a main orientation page, this page provides no link to the “pre-orientation” program. Once at the Bowdoin Outing Club page, there is a page devoted to the pre-orientation program. While it is not a fancy page, a good chunk of information is present. From this page, students can download the necessary registration and health forms, see pictures from past trips, and look at equipment lists for the different trips. This page seems to be similar to what Colby has implemented at the moment.

Harvard’s program, First-Year Outdoor Program (FOP), goes into more detail than the Bowdoin or Bates page (Figure 4-1). The page is designed using frames. The frame on the left is the menu bar, while the frame on the right contains the bulk of information. The website is divided into six sections: About FOP, FOP Participants, Alumni, Parents, Leaders, and Trainees. The participant page had information about trips and registration information. However, Harvard is still doing mail-in registration by downloading PDFs. The leaders page simply consisted of upcoming events that leaders needed to be aware of during orientation. The trainees page was devoted to students who were interested in becoming FOP leaders. It included information about the training process, FAQ, as well as the downloadable application. The FOP pages were easy to navigate since the menu on the left was static. The pages from section to
section were consistent with similar fonts and layouts. However, the layout was not the most appealing and eye pleasing layout possible.

![Figure 4-1 - Harvard's FOP Main Page](image)

Yale broke down their website in a similar manner to Harvard. Their website for Freshman Outdoor Orientation Trips (FOOT) was extremely easy to navigate, had consistent layout throughout the sections, was easy to read and pleasing to the eye (Figure 4-2). Their menu bar was also on the left. It appears as though they did their site using frames as well, but they are less noticeable then in other sites reviewed. They took a different approach to breaking up their menus by having sub headlines beneath each group. For example, there is a section called “Leader Information”. Under this headline, there are links to the leader facebook, calendar, application questions, and the leader application. This was extremely helpful in quickly locating information. A downfall to their site is all forms are still PDF format and can only be sent in via mail.
Williams Outdoor Orientation for Living as a First year (WOOLF) was an extremely vivid site using mainly purple and yellow for colors (Figure 4-3). Their site covered far less information than Yale or Harvard's but there were some interesting features. For example, the section with the leaders provided links to each of their names which then showed a picture of them. However, this did not tell any further information about the leader or about which trip he/she was leading. There was also no place to register or download the form to register. This site was also constructed using frames which look great when the window is open to maximum size. Yet, when the window is not maximized, scroll bars appear throughout the screen which is very confusing (Figure 4-4).
The Middlebury Outdoor Orientation (MOO) site is very similar to the Bowdoin page (Figure 4-5). The main page has the bulk of the information. It took a couple of visits to the site to realize there was slightly more information about the trips besides what was on the main page.
The main page is very hard to read with the main font in bold and the headlines as the same font but not bold faced. Links to download PDF versions of all registration forms are also found on this page. The layout and fonts are not consistent from page to page which causes some confusion as to which pages are related to MOO and which pages are related to the February Outdoor Orientation (FOO) program. Lastly, the packing lists for all of the trips are on one website. So, if you try to print one, unless you know how to print only a section of a page, you will get all of the trips’ packing lists.

Figure 4-5 - Middlebury's MOO Main Page

Tufts orientation program like many others is run by the college Outing Club. It contained a lot of the information that was common between the sites. In this case, the menu bar was only an inch wide which for a menu is extremely small (Figure 4-6). The other unusual part of this website was the fact that the registration sheet was part of a website instead of a downloadable PDF file (Figure 4-7). Normally, this would be a good idea to make the user experience easier. However, there was no easy way to print the form unless printing the entire web page which then caused a page break in the middle of the form. This page break makes the form seem very unofficial and could create hesitancy for users trying to use it (of whether it is valid or not).
TUFTS WILDERNESS ORIENTATION

Applications will be in the mail in late May. They will arrive with your Freshmen Welcome packets along with many other pamphlets.

Please be aware that space is very limited. Participants will be drawn by lottery and late applications will be considered on a space available basis only. We will inform you by July 10th whether you have been placed on a trip. Along with your acceptance, we will send more details about your trip and the program.

Once again, we encourage you to apply...Tufts Wilderness Orientation is an amazing opportunity, and we want you along! NO EXPERIENCE NECESSARY

Sincerely
Ed Edison
Annemarie Braun
Tufts Wilderness Coordinators

Figure 4-6 - Tufts Wilderness Orientation Main Page

Figure 4-7 - Tufts Registration Form

Dartmouth's site was the only site I was able to find that allowed and actually strongly encouraged online orientation. This site went far more in depth than many of the other sites above did and is the closest to the site envisioned for COOT. The orientation trips are offered by Dartmouth Outing Club, so the registration site is part of the outing club site, which can cause
some confusion. Like many other sites, there are specific sections for leaders, students, and parents. The leaders section includes applications (when available), list of accepted leaders, and other important information. The student section allows for registration online, posts the trip each student is assigned to, as well as gives gear lists for each different type of trip. The parent section just gives general information about contact information and making arrangements to get to Dartmouth.

![Dartmouth's DOC Trips Main Page](image)

The DOC leader section was fairly simple. At this time, the application is not available online, so it is unclear on whether or not it is an online application or a PDF file. The list of accepted leaders simply is a long list of every leader in alphabetical order, with the trip, section of the trip, and arrival and departure dates. It appears to be a table pulling from a database and does not allow for any searching by trip name, leader name, etc.

Much like the DOC leader section, the new student section of the website contains a list of all first year students who have registered to be on trips and their assignment. This list is also not searchable and the column titles of the table are very confusing using labels such as AppPersonal.Name.Last.

The online registration link brings you first to a page of instructions (Figure 4-9). This explains which paper work must be signed and returned via mail and the correct steps to follow.
The registration page is broken up into fifteen sections. Each section is supplemented by a help page which explains exactly what information they are looking for. The first section asks for general information about the student (Name, Address, Phone, E-Mail) (Figure 4-10). The second section asks whether or not the student is applying for financial assistance. This assistance is only for the DOC trips and not for attending Dartmouth in general, which is at first a bit confusing, however the help section spells this out clearly. The next few sections deal with choosing a trip and borrowing equipment. This is followed by medical history. It is interesting to note that this is not a secure form so they do not guarantee the confidentiality of any information submitted. The form then returns to trip selection asking the student to describe his/her experience. Once the information has been filled out, you simply click a submit button and a thank you page will appear confirming that your registration has been submitted (Figure 4-11).

![Registration Welcome]

Figure 4-9 - DOC Registration Main Page
While the Dartmouth system is by far the most advanced for online registration of all of the schools reviewed, it does have its flaws. The lack of security of the form could become a major issue, especially with medical history. The layout of the site is not as well done as the Yale site for example. By looking at each of the sites, it gives a better idea of what information
each college finds necessary to post, different approaches to presenting this information, and different user experiences at each place. It is important to keep in mind both the good and the bad things of each site when constructing a new site so that these can help determine the layout of the interface and the functionality of the site.

4.6 Testable Usability Goals

These testable usability goals are ways that the performance of the system can be measured. For this project, there should be usability goals for each of the different user groups as they will be tested individually. These testable usability goals will be closely connected to the tasks each user group wishes to accomplish with the system.

Many times, usability goals can be set based upon other similar systems or the current system. However, in the case of the first-year students, the current system is a paper form and there are not many similar systems. Ideally, the form to register would be quick to fill out but since there are parts of the form that may require information to be looked up it is hard to put a time limit on filling it out. However, it is important that the first-year students can easily locate the information about the different trips in order to select one. The following testable usability goals have been set for the first-year students:

1. Can locate information about different trips offered within a minute.
2. Can register for COOT without making more than three mistakes in providing the information asked for.

The COOT leaders are in a similar situation to the first-year students. The application requires time in order to answer each of the questions, so it is not feasible to measure the usability of the system by this. Instead, the following testable usability goals have been set:

1. The number of times the user expresses clear frustration is less than two.
2. The number of user errors in navigating the application is less than three.

The administration, unlike the other two groups, needs to be able to enter and retrieve specific data in a timely manner. The navigation of the system must also be clear, so that time is not wasted wandering around the system. With this in mind, the following usability goals were set:

1. Must be able to enter information regarding transportation, campsite information, or supplies for a specific trip in less than two minutes.
2. Must be able to find the appropriate part of the system to enter the information in within two errors.

3. Retrieve and print a list of the first-year students, their trip they are on, and their COOT leaders within two minutes

4. Locate a student's personal information (address, phone number, trip preferences, and medical history) within two minutes

5. Enter a student who registered via mail into the system within five minutes
Chapter 5 – Prototype and Evaluation

This chapter begins by exploring the prototype development process for this project. It examines the choices that were made in terms of grouping information and the relationship between these groups. Once the prototype is developed, the evaluation phase begins. This evaluation phase includes designing a test, running the test, and then interpreting the results. These results are then used to influence changes in the prototype before another round of evaluation begins.

5.1 Prototype Development

As previously discussed, the COOT system has three main user groups. This needs to be kept in the front of our mind when designing a system to accommodate these groups. The overlap of common tasks to be accomplished between the groups will also need to be kept in mind.

When breaking up the content to be on specific pages, the way the main site was configured, the way the registration is configured, and the way other similar sites were broken up all factored into the decision-making process.

The diagram above depicts the different paths that can be taken from the main page of the system. This path will depend on the role of the user and what he/she wishes to accomplish with the system.
5.1.1 First-Year Students

The current website online for COOT is designed specifically for those interested in learning more about the orientation program. For the most part, the people using this website will be similar to our first-year student user group. Therefore, keeping this in mind, we can absorb a lot of the content from the current website into the new site as it will be helpful for users to be able to learn more about the trips before registering.

<table>
<thead>
<tr>
<th>First-Year Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>About</td>
</tr>
<tr>
<td>Equipment</td>
</tr>
<tr>
<td>Administration</td>
</tr>
<tr>
<td>History</td>
</tr>
<tr>
<td>FAQ</td>
</tr>
<tr>
<td>Application</td>
</tr>
<tr>
<td>Leaders</td>
</tr>
</tbody>
</table>

Section 5-2 Overview of First-Year Section

The diagram above represents the seven sections that are accessible to someone who has entered the system as first-year student. Each section is linked to all of the other sections allowing easy navigation from one area to another. The rectangular blocks indicate static pages, the upside down triangle represents input into the database and the triangle represents dynamic information being retrieved from the database. The About, Equipment, Photo Gallery, History, and FAQ sections are static pages that give information pertaining to COOT trips in general. They will not read or write to the database in any way. These are all areas that exist on the current COOT website. The leaders section will only be retrieving information from the
database. This area is designated a place for COOT leaders pictures and brief information about
them (name, major, and e-mail address).

The application section is based off of the current paper application that is sent to
students. The paper application divides the information into five sections. These five sections
are shown in the diagram below.

![Figure 5-3 Breakdown of Application Page](image)

Since so much information is requested from the user, each section will become its own page.
Users will be able to navigate back and forth through the pages to change the information if
necessary.

The general information page asks the user for their name, gender, mailing address, home
phone, summer phone, age, and t-shirt size. Once this section is completed, the user moves onto
the emergency contact information. This involves giving the name, relationship, and phone
numbers for two people who are designated emergency contacts. These people are contacted if
something happens while the first year student is on a trip.

The next section, medical information, asks for a list of allergies, whether the student has
asthma, if the student is insulin-dependant, if the student is insulin dependant or if the student is
epileptic. Here students registering are also asked to list any prescription medications they are
on, as well as describe any other medical conditions that might affect what trip they are placed
upon. Security and privacy will be a concern when implementing this page.

The information entered on the experience page is used to assign students to appropriate
COOT trips. Students are asked to rank their outdoor skill/comfort level on a scale from one to
ten (one being novice and ten being expert). They are also asked to list any outdoor experiences
(individual and group) that are relevant to their trip choice.
Finally, once the information regarding their experiences has been entered they enter the trip choice page. Here students rank their top three types of trips they would like to participate in. They can choose between three different levels of backpacking, two levels of basecamp, two levels of canoeing, civil service, combination, excursion, fishing, fly-fishing, road biking, rock climbing, sea kayaking, theater/improve, trail biking, and trail work. This will be an important point to provide a link to the trip descriptions while ensuring that the information already entered in the registration process is not lost.

In January, I completed high-fidelity prototypes of the system based upon the information gathered and feedback received from my low-fidelity sketches. The following screenshots were developed for the first-year student COOT application.

**Figure 5-4 – First-Year Students General Information.**
COOT Application Section 1
II. Emergency Contact Information

Contact 1:
We will contact this person first in the event of an emergency.

Name:
Relationship to you:
Home Phone:
Work Phone:
Pager:
Cell Phone:
* If applicable

Contact 2:
We will contact this person in the event of an emergency if we are unable to reach your first emergency contact person.

Name:
Relationship to you:
Home Phone:
Work Phone:
Pager:
Cell Phone:
* If applicable

Figure 5-5 – First-Year Students Emergency Contact Information, COOT Application Section II

III. Medical Information

Please list any serious allergies (i.e., bee stings, nuts, shellfish, penicillin, etc.)

Do you have asthma? □ Yes □ No
If yes, do you carry an inhaler? □ Yes □ No
Are you an insulin-dependent diabetic? □ Yes □ No
Do you have epilepsy or any other seizure disorder? □ Yes □ No
Please list any prescription medications you take regularly.

Do you have any other medical conditions (i.e., orthopedic problems, eating disorders, sun sensitivity, etc.) that might affect your participation in outdoor activities such as hiking, swimming, or canoeing? □ Yes □ No
If yes, please explain.

Figure 5-6 – First-Year Students Medical Information, COOT Application Section III
IV. Experience

Trips appropriate for all skill and comfort levels are offered. In order to best meet your individual needs, please answer the following questions as accurately as possible. Vague or misleading information may result in a trip placement that is beyond your skill/comfort level.

On a scale of one (1) to ten (10), please rate your outdoor skill/comfort level.

- [ ] 1  -  [ ] 2  -  [ ] 3  -  [ ] 4  -  [ ] 5  -  [ ] 6  -  [ ] 7  -  [ ] 8  -  [ ] 9  -  [ ] 10  -  [ ] Expert

In the space provided, please list experience you have which is relevant to the trip preferences you will indicate. Please include both organized trip experience (e.g., NOLS, Outward Bound, Wilderness Summer Camp) and individual experiences. The information you provide will be used to place you on an appropriate trip.

---

V. Trip Selection

Please select your top three trip preferences and difficulty level (if applicable). Each selection must be in a different trip category (e.g., Backpacking Level 1, Canoeing Level 1, Fly-Fishing).

1. 
2. 
3. 

You will be responsible for providing a sleeping bag and frame pack for all trips unless specifically notified otherwise. For hiking trips, you are expected to provide your own bicycle and helmet. For fishing trips, you must provide your own rod and tackle. A complete trip-specific equipment list will be included in your trip placement mailing.

- [ ] Please check here if you think you might need assistance obtaining equipment for your trip. If you indicate that you will need help, the COOT2 Coordinator will contact you.

All trips are available on a first-come, first-serve basis. Trip preference and past experience will be used to place students on the most appropriate COOT2 trips.

---

5.1.2 Leaders

The leaders section was developed for Colby students who wish to become COOT leaders and then later on after being chosen to become a COOT leader. The leaders will also
need access to much of the information that first-year students will have access to. The following diagram depicts the information that leaders may need to use.

Figure 5-9 – Overview of Leader Section

The pages that are rectangular are static pages and the information will not change. The triangle indicates information being retrieved from the system, while the upside down triangle refers to information being entered into the database. The diamond indicates that information will be retrieved and inserted into the database. The equipment page provides the list of equipment necessary for the different trips. Though this differs from type of trip, it is consistent from year to year. The trip description may be helpful for potential COOT leaders as they are asked to rank the different trip they would like to lead. This will help give them the necessary information in order to make this decision.

The application for COOT leaders is available to all Colby students during March. The first part of the application is basic information including name, class year, major(s), phone at Colby, box number, e-mail address, home address, gender, and any experience leading COOT trips. The questions then change to longer thoughtful questions, drawing information from the person about why they want to lead COOT and what they see the role of COOT being. The following questions are asked of any potential COOT leader:

- What sort of leadership and/or experiences have you had at Colby?
- What do you feel is the purpose of COOT?
- Please explain why you think you would be a good COOT leader.
• Please discuss, with specific examples, how you would continue your leadership role once the outdoor trip is over and you are back on campus.

• Given your experience, either as a COOTer or a COOT² leader, what were some negative aspects of your trip or experience that should be addressed and/or changed?

• In your opinion, what are some of the main “issues” of particular relevance for first-year students?

• What are your thoughts about the diversity component of the COOT² program? Have you had any experience facilitating workshops or programs for students or other campus groups?

• Please list any other experiences you feel qualify you to be a good COOT² leader, including organization and/or personal undertakings.

The application is roughly two pages long with three to four lines of blank space to answer each question, except for the last one which asks for it to be answered on a separate page of paper. This was helpful in judging the size of the textbox that should be created for answering the questions as many people determine how much to write based on the amount of space they are given. The other difficult part about this application was that it is slightly long to be placed on just one page. It was very difficult to even logically break the questions into groups. If for instance, there was one question per page the application process would be extremely long (over 10 pages). A compromise was to put two or three questions per page, except for the last question, which would require a page to itself.

Potential leaders are also asked in their application to rank the type of trips they would like to lead. They are given a list of the twelve different types of trips and are asked to rate them in descending order with one being the first choice. On the paper application, they are given no description of the trips other than what they know from their own experience. However, it would be nice to build into the prototype a way for them to get a brief description of each type of trip in case there is some uncertainty as to what each trip includes.
2004 COOT\textsuperscript{2} Leader Application

In order to complete required COOT\textsuperscript{2} Leader training, you will be obligated to return to Colby on Friday, August 20. Training is absolutely mandatory; there will be no exceptions.

It is important to note that if you have served as a COOT\textsuperscript{2} leader in the past, you may not automatically be chosen to lead COOT\textsuperscript{2} again. Please take time to address these questions thoroughly, as the selection committee has adequate information upon which to base their decision.

It is also important to note that COOT\textsuperscript{2} leaders will be expected to play an active role with Orientation and their COOT\textsuperscript{2} was once they return to campus, after the Outdoor Program has concluded. COOT\textsuperscript{2} leaders, by way of applying for this position, are committed to participating and assisting with all of COOT\textsuperscript{2} training, the outdoor program, and the on-campus orientation component with equal enthusiasm.

There will be approximately 100 COOT\textsuperscript{2} leaders chosen from a very competitive pool of applicants. Decisions will be made in April, and you will be notified by April 9th.

Figure 5-10 – COOT Leader Application, page 1

2004 COOT\textsuperscript{2} Leader Application

First Name: \\
Middle Initial: \\
Last Name: \\
ID: \\
Gender: \\
Major(s): \\
Phone: \\
Box: \\
E-Mail: \\
Home Address: \\
City: \\
State: \\
Zip Code: \\
Have you ever served as a COOT\textsuperscript{2} leader? \(\square\) Yes \(\square\) No \\
If so, which trip and what year? \\
What sort of leadership positions and/or experiences have you had at Colby?

Figure 5-11 – COOT Leader Application, page 2
Another part of the system for the COOT leaders will be entering their summer contact information, if it is different than the home address entered on their application. Currently, this is done by having the COOT leaders stop in the office and write down their name and address on
a piece of paper. This is also the way they indicate that they accept the position of COOT leader. The prototype created (Figure 5-14) allows them to complete both by using the system.

An additional feature for the COOT leaders would be for them to get the list of their COOT students online. Currently, they receive a list of their names when they get on campus for summer training. If there are any changes to the list, they need to be notified and receive a new list. Having an online list allows it to be automatically updated as well as provide the leaders with information such as email addresses.

5.1.3 Administration

The administration can be viewed as the super users of the system. They will need access to everything mentioned above as well as more detailed information. They take all of the information entered by first-year students and leaders, evaluate it, and then assign them to the correct trip. The information that they need to access and enter can be broken down into three groups, shown below.
At the start, the administration needs to be able to enter first-year students who have mailed in their application rather than filling out the online version. This means that they will need to have access to a form similar, if not identical, to the registration form seen by first-year students. Once students have registered, it is also the job of the administration to look at their information and assign them to trips. The administration will also need to be able to quickly locate a student if they need contact information or have a question about something located in their application. These tasks can be broken down under the first-years sphere to be represented below.

As discussed earlier, the first-year registration online is based off of the paper version. Therefore, when receiving applications by mail it will be easy to input them into the system. The layout of the screens can be similar to those seen above for first-year registration. An example is shown in Figure 5-17.
The search function is convenient when locating a student quickly. Mainly students are searched for based on a few select criteria, their name, ID number, or trip they are assigned to. The administration will have the choice of entering one or more than one search criteria based on what they know about the student in order to locate their record. It might helpful to look at expanding this feature to be able to search other criteria but the criteria mentioned above are the most prominent.

Assigning a first-year student to a COOT trip was something that looked as though it may be able to become automated. The first-year student ranks his/her top three choices in the application and then the trips are filled on a first come, first serve basis. However, the COOT
coordinators read through the first-year student’s trip choices and then compare it with the first-year’s experience in order to assign them to an appropriate level trip. If their experience is vague, the coordinator will make the necessary phone calls in order to get the information needed. Due to this process, it is not practical to have the process automated.

There are many factors that will need to be taken into consideration when assigning to a trip. Besides looking at the experiences and trip choices mentioned above, the number already on the trip also must be taken into consideration. All trips have a maximum number of students, normally between 12-15. All of this information needs to be readily available so that the coordinator can make a decision without having to switch screens numerous times.

The prototype shown in Figure 5-19 lists the name of the first-year student, his/her trip choices, experience, and then allows for the coordinator to choose a trip. Once a trip is selected, the number of spots available will be shown. If the coordinator needs to reach the student to get more information, he/she can click on the first-year student’s name to see the contact information. This page will show all students who have not been assigned to a trip, based on the time their registration was submitted.

![COOTer Trip Assignment](image)

**Figure 5-19 – Administration Page to Assign First-Year Student to COOT Trip**

In order to modify a first-year student’s information, that student record must be located. Therefore, there needs to be a built-in search function to the modify a COOTer information section. The search function returns the possible matches, as well as their ID number and trip that they assigned to in order to help locate the correct student. Once the student is identified, clicking on his/her name will bring the coordinator to the registration form filled in with the information already in the system. The coordinator can then make any necessary changes and
save them. These changes can range from a change of address to reassigning the student to a different COOT trip.

Figure 5-20 – Administration Page to Modify First-Year Student Info, Step 1

Figure 5-21 – Administration Page to Modify First-Year Student Info, Step 2

Figure 5-22 – Administration Page to Modify First-Year Student Info, Step 3
The leader section, like the COOTer section, can also be broken down into subsections. Like the first-year section, the administration needs to be able to add leaders, search leaders, assign them to lead a particular trip, and modify their contact information. Therefore, the leader section can be broken down as shown in the diagram below.

![Figure 5-23 - Administration Tasks with COOT Leaders](image.png)

The add leader section can be slightly different from the form that the potential COOT leaders see online. There is no need for the coordinators to enter the answer to all of the questions on the application. The coordinators will only be entering those people who have been chosen and did not fill out the online application into the system. The only information that they will need is the leader's name, address (home, mailing, and school), ID number, e-mail, campus phone, and whether he/she has been a COOT leader before.
When looking for information regarding a COOT leader, the most commonly asked questions are who leads this trip or what trip is he/she leading? With this in mind, the search page can be constructed to answer these. In order to answer these questions, the coordinators must be able to search by trip name, COOT leader’s name, and possibly their ID number. This is very similar to the COOTers search page.

The decision about which Colby student is leading which specific COOT trip is decided by April. The COOT committee, which is a group of past COOT leaders who interview potential COOT leaders, decide what trip each leader that is selected is leading. So the COOT coordinator
only needs to enter the trip that the new leader has been assigned to. The coordinator does not need to see any information in order to make a decision. Therefore, the coordinator simply needs to locate the appropriate person and then match them to the trip they have been assigned to.

**Assign COOT Leader to a Trip**

*Step 1. Locate COOT Leader*

ID: 
Last Name: and/or
First Name: and/or

![Figure 5-26 – Administration Page to Assign a COOT Leader to Lead a Trip, Step 1](image)

**Assign COOT Leader to a Trip**

*Step 2. Assign Trip*

<table>
<thead>
<tr>
<th>Name</th>
<th>ID</th>
<th>Trip</th>
</tr>
</thead>
<tbody>
<tr>
<td>Christopher Johnson</td>
<td>254978</td>
<td></td>
</tr>
<tr>
<td>Bill Johnson</td>
<td>120776</td>
<td></td>
</tr>
</tbody>
</table>

![Figure 5-27 – Administration Page to Assign a COOT Leader to Lead a Trip, Step 2](image)

The coordinators must also be able to modify the COOT leader’s information during the course of the summer. It is possible that a COOT leader changes their summer address or gets reassigned to another trip. The modify information section will allow the coordinators to make these changes. Set up similar to the COOTers modify information section, the coordinator will need to search for the leader whose information needs to be changed. Once the search results have been returned, the coordinator can identify the correct person and then click on their name in order to change their information. The information available to change is the same information that a coordinator uses to enter a COOT leader into the system (see above) as well as their current trip assignment.
Modify COOT Leader Contact Info

Step 1. Locate COOT Leader
ID: [ ]
Last Name: [ ] and/or [ ]
First Name: [ ] and/or [ ]

Search

Figure 5-28 – Administration Page to Modify a COOT Leader’s Information, Step 1

Modify COOT Leader Contact Info

Step 2. Click on COOT Leader’s Name to see and modify contact info
Name: [ ]
ID: [ ]
Trip: [ ]

Christopher Johnson 447829
Riki Johnson 123676

Figure 5-29 – Administration Page to Modify a COOT Leader’s Information, Step 2

Modify COOT Leader Contact Info

Step 3. Modify Info and Click Submit
First Name: [ ]
Middle Initial: [ ]
Last Name: [ ]
Trip: [ ]
ID: [ ]
Gender: [ ]
Major(s): [ ]
Phone: [ ]
Box: [ ]
E-Mail: [ ]
Home Address: [ ]
City: [ ]
State: [ ]
Zip Code: [ ]
Summer Mailing Address: [ ]
City: [ ]
State: [ ]
Zip: [ ]

Has he/she ever served as a COOT leader? [ ] Yes [ ] No

Figure 5-30 – Administration Page to Modify a COOT Leader’s Information, Step 3
The coordinators are also in charge of specific trip details. These details include what supplies are assigned to each trip, booking the campsites, as well as organizing transportation for each trip. These separate sections allow the trip information to be broken down into subcategories shown below.

![Diagram](image)

**Figure 5-31 – Administration Tasks Organizing Trips**

A COOT trip is described by three factors. Each trip is given an unique name, normally according to the place it is taking place in (ex Belgrade Lakes A). A trip type describes what kind of trip it is (basecamp, canoeing, hiking, etc). Lastly, the level indicates how difficult the trip is and can range anywhere from 1A-3B. This is the basic information that is needed in order to create or add a new trip to the system.

![Add a COOT Trip](image)

**Figure 5-32 – Administration Page to Add a COOT Trip**
This basic information is then built upon as the summer progresses. Once a route is figured out (most follow the same route every year), campsites will be booked for each night. The name and address of the campsite for each night, for each trip is then recorded. The check number as well as confirmation from the campsite are recorded once they are available. This information will be recorded in the campsite information section.

**Campsite Info**

![Campsite Info Diagram](image)

*Figure 5-33 – Administration Page to Add Campsite Info*

Getting the first-year students and COOT leaders to each different trip site also differs depending on trip. Some trips take vans while others take buses. If vans are being taken, information regarding the size and number of vans must be recorded. If a bus is dropping students off, the bus departure time from Colby must be recorded as well as its pickup time on the last day of the trip. There must also be spaces to make notes about the bus (confirmation, other trips that may be sharing the same bus, etc).

**Trip Transportation**

![Trip Transportation Diagram](image)

*Figure 5-34 – Administration Page to Store Trip Transportation Info*
The supplies that are taken/required for each trip must also be kept track of. This will differ from trip to trip. For example, those trips who are not canoeing will not need paddles. The following supplies need to be tracked:

- Number of tents
- Number of stoves
- Medkit
- Pot
- Water Purification
- Map
- Atlas
- Compass
- Bathroom Kit
- Whistles
- Number of Canoes
- Number of Life Jackets
- Number of Paddles
- Number of Rubbermaid Bins
- Misc

While not every trip uses each of these supplies, the coordinator goes through and marks which are needed for the trip. This list is then used to gather and sort the supplies before giving them to the COOT leaders before their departure.
5.2 Description of Heuristic Evaluation

Heuristic evaluation is the process of testing a potential user interface with people who are familiar with the user-centered design principles. It is used to point out flaws with the design, offer different suggestions for organization of the information, and a general critique of the system. These test users are often not the real users of the system but are asked to hypothetically take the role of the different users when critiquing the system.

The implemented prototype is known as a horizontal prototype. It has no system at the backbone, but rather fakes the storing of the data. When filling out a continuous form, such as the first-year application, the data appears to be saved from page to page. This is done so that the user's interaction with the system is as real as possible. Also, this allows the users to interact more naturally with the system; their responses will be more genuine, instead of having them say, “I think I would react this way.”

This first round of testing looked purely at the input of information by each of the different user groups. A later test round will look at the way information is returned to the user upon request. The return and summary of collected information is most important to the administration. The other two user groups (those registering for COOT and COOT leaders) are less dependant on output in order to complete their tasks with the system.
The users were first presented with the COOT application screen. This screen is intended to be used by people (first-year students) who wish to register for a COOT trip. Before the test, test users were made aware that none of the information that they enter will be saved. They were then asked to use the system as if they themselves were registering for a COOT trip. Test users could choose to enter their own information, a fake alias, or ask for information about a fake person. If information was asked for, it was presented in a paragraph format to avoid a data type entry task. Test users were specifically not given a list of fake data to enter so that interaction with the system is as real as possible. It is important to understand the users thought process of collecting the data instead of just reading it off of a paper.

The COOT registration form is divided into five sections. The first section asks the user for general information such as name, address, phone, e-mail, etc. On the next page, the user is asked for the emergency contact information of two people. This is a possible location where users may have to leave the computer to locate specific information, such as the phone number of someone. Users may once again have to locate information when asked to enter medical information on the next page. After the basic medical information has been entered, users were asked to explain their experience on a scale from one to ten and then further as a descriptive paragraph. Finally users were asked to rank their top three trips. A link is placed on this page to the trip descriptions so that the users may look up and explore the different types of trips without losing their entire registration form.

Users were asked to use the think aloud process. This entails talking to explain the thoughts that are running through the users head to make different decisions while filling out the form. In addition, notes were taken of the user’s process of filling out the form. These notes detailed any hesitancy or indications of confusion, the way the user maneuvered through the pages (returning to previous pages), and a general time frame for filling out each section of the registration.

The next section of the system that was tested was the COOT leader’s application page. This similar to the COOT registration page is divided up into many pages. This application is very different from the COOT registration application as it requires more thought to fill out. There are four pages that explore qualifications, reasons, and expectations of students who wish to become a COOT leader. These answers may vary in length from one paragraph to many.
Since we know that all the test users are Colby students, they were asked to fill out the form as if they were personally applying to become a COOT leader. If for any reason they felt uncomfortable disclosing information such as home address they can simply substitute false information. Information asked for on this form should not require users to look up any information except possibly their ID number.

Thinking aloud was once again used to explore the layout and order of the questions in the application. Timing was less important on this task for when the task is completed in reality users may choose to dedicate a significant amount of time to answering the longer questions. It was more beneficial to note the way the user navigates and interacts with the form.

The last section of the system to be tested was the administration section. However, not all of the input screens were tested. The administration has a COOT registration page, which mirrors the page tested earlier as well a leader application page. Since these pages will have been tested earlier, it is redundant to test them again with the same users.

Users were asked to test the trip information section of the administration page. This involves receiving the details about a new COOT trip that has just been created. They would then need to take this information and record it in the system. Once the trip had been added into the system, they would need to fill out other information about the trip. This information would include the supplies that are needed for the trip, campsite information for each of the three nights, as well as transportation to the necessary locations.

Unlike the previous two tests, users were given the information needed to enter into the system. However, they were not guided to the exact location of where they need to enter their information. Instead, it was part of the test to observe how easy it is to locate the correct spot to enter the information.

Once again, users were asked to think aloud and explain the reasoning they go through when entering information. In addition to taking notes on their actions, users were also timed to see how long it takes to complete each stage of the data entry. Speed was a factor in this case, as the administration needs to be able to complete these tasks quickly without getting bogged down by the program.

For all three tests, the user were given little to no information on how the system works. This was due to the fact that the user is not expected to have any training in order to complete the
tasks necessary. The test should be self explanatory, though the test user may ask for clarification if he/she is confused by the task to be accomplished.

Following each task, a short questionnaire was presented to the user to fill out (see Appendix A.2). This served as a wrap up of each section, allowing for smoother transitions in between sections of the system. When all the tests had been completed, the test user was asked to fill out a short final questionnaire about his/her experience with the system (see Appendix A.3). This gave the user a chance to comment on the system as a whole, changes he/she would make, color choices, and other factors that may have affected his/her performance completing the tasks. This information along with the notes taken throughout the tests were used to make changes to the interface. The way the pages are organized, and more specifically the way the fields were grouped would be further analyzed based upon test results for all the areas. In the administration section, the grouping of information and navigation of the trip information section could be reviewed and revised as necessary.

After the changes from this study have been made the next step was to present a similar test to each of the different test users to receive their feedback. The expert test users may also need to be tested again depending on the number of problems that are found in this initial pilot study.

5.3 Heuristic Evaluation Results

For my heuristic evaluations, I had seven “usability experts” critique the system. These experts were students who had taken or are taking the user-centered design class or have had experience in graphic design. Two of the expert users did all three of the tasks mentioned above. The remaining five users did anywhere between one and two of the tasks depending on time availability. In order to comprehend the results, I am going to look at the successfulness of completing each task separately.

The first task looked at registering for COOT as a first-year student. Every person who completed this task ranked it as easy to very easy to complete on their post task survey. People often stopped to comment and ponder the format in which to enter the phone number. They then made their own inference about what the system was looking for and continued onwards.

In other tests, users complained that when using the tab button to move between fields, it skipped over the gender selection menu. This then caused them to stop using only the keyboard
to enter information and use the mouse to navigate back to the missed field. Users also expressed frustration when they reached the application question regarding outdoor experience. Lastly, in the first-year section, there was some hesitancy in using the form navigation system. When going to click next, the mouse did not turn to a hand to indicate a button but rather a cursor, which caused some hesitation. Also, when the user came to the last page and pushed the next button it goes immediately back to the first page.

These minor problems observed during this task did not substantially effect completing the task at hand. However, some minor changes that could be made to the system were discovered throughout the testing process. With the field for the telephone number, it was suggested that the box be split or an example be given in order to clarify this. Doing this would alleviate the confusion of what format the system prefers and would allow for greater data integrity within the database.

The users also felt that choosing their trip preferences before listing their outdoor experience would be a better order. This way, when entering the experience, there would be some basis on which to write about. The information then provided in the experience box would be more specific to the type of trips the users chose.

The next task involved applying to be a COOT leader. Some of the same concerns from the first-year students page resurfaced, such as the navigation system having a next button on the last page and having the cursor change to a hand when over a button. Some users commented on the text box size being too large for the information that was being asked for. Confusion also arose in determining what page of the application the user was on.

Generally, this task went very well and no major problems were encountered. The text box sizes needed to be revisited so that they were not misleading. This was especially important for school ID field and box number field, which are receiving information of a known set length. It was also suggested that page number in the navigation bar at the bottom be shaded to indicate your position in the application. Overall, four out of the five test users ranked the task as being very easy to complete. There were no major problems in entering the information and the instructions seemed clear and concise.

A lot of the confusion for the third task stemmed from the fact that the usability experts did not know enough about the coordination of the COOT trips to fully understand the system. The task purposefully set people on the wrong page to begin, in order to see their logic in
navigating the system. People expressed their confusion and then began reasoning their way through each tab to pick the next best choice. For most users, it took about five clicks to locate or realize they were on the right page. Many test users were on the right page but continued to click, for the option they wished to choose was not clickable since they were already there.

The next round of confusion came when entering the campsite confirmation information. The separation of fields and information was modeled after the current system. Since the test users had never used this system, many started to enter the confirmation for each campsite in the wrong text box. However, midway through the task this was realized and corrected.

The transportation page also provided a minor confusion. Since trips take either buses or vans, it was intended to have two separate sections on the page. However, these sections were not clear, and information was placed into the wrong fields. The information regarding the bus was often inserted into the field designated for van information without the users even realizing their mistake.

Out of the three tasks, this task caused the most difficulty for users. Some mistakes that were made the users did not even realize they were making. The frustration level for the users seemed escalate in points where they were struggling to find the correct page to enter the information. Part of this is due to the fact that the test users have had no experience organizing COOT trips and therefore did not realize the process that is taking place.

Users did not have any specific recommendations for fixing the navigation problem they encountered. One suggested renaming the Trip Info tab. Others said that while there was some navigation issues originally, once they had browsed through the pages, it was easy to know where specific information went. It was also suggested that the transportation page be dynamic to reflect the choice of whether taking a bus or a van. This could then only show the fields that were relevant to a bus or a van.

Overall, all of the tests went very smoothly. The mistakes that were made can be fixed with minor changes but will help enhance the system. It was also helpful to see how the instructions were interpreted so that if they were unclear they could be modified before the user testing.
5.4 User Testing Instructions

Unlike the previous round of testing, user testing involved the actual users. These users were asked to perform tasks with the system that are similar to their real life tasks. Instead of offering suggestions to fix the problems, the experimenter is in charge of analyzing the results of the test.

In this round of testing, the administration portion of the system was tested. Leanne Burnham, the assistant director of student activities, was asked to test the new system. She had been involved with the development of the system since the very beginning of last semester. This made it easier to do testing as she is familiar with the process and understood the point of the testing.

The heuristic testing tested all three parts of the system individually. However, in this round of testing, not all three parts can be tested as Leanne is a member of only one user group. The administration user group has the highest interaction with the system and may in fact have to enter first-year students or COOT leaders if paper applications are received. Their interaction with the system is also constant, unlike the other two groups, making their user testing slightly more important.

Though the administration is the smallest user group, in terms of people, they have the most tasks to accomplish with the system. When choosing the tasks to test, I focused on tasks that were different from the current system and had not been tested as part of last year's class. It was my hope that these tests in addition to last year's results would create a full comprehensive system for the administration.

The user was first asked to assign a COOT leader to a specific trip. The system currently being used by the coordinators has no specific section for doing this. Rather, it is done in a round about way through another screen. This test was useful to see if the new process is helpful, and self-explanatory or if the user tried to immediately revert back to doing it in the same way the current system does.

As the next part of the test, a COOT leader's information needed to be modified. This is an important part of the system to look at because it is new and will need to be used more often than in previous systems. With COOT leaders entering their own information online, the coordinators will spend less time entering new leaders and more time updating information if necessary.
Similar to modifying a COOT leader's information was the modification of a first-year student's information. Once again, the same scenario is present, as the coordinators will be spending less time entering individual information and more time modifying, if necessary. However, in this case, the information regarding the first-year student is spread out over numerous pages, similar to the application. Therefore, it is necessary to see if this information is easily located or if the user is deterred by not immediately finding it on the first page.

A major flaw with the system currently in place is the lack of flexibility in adding a new trip. Adding a new trip creates a huge string of problems and cannot be done by the coordinators themselves. In the new system, this was presented as a task for the coordinators to do on their own. They were given basic information and asked to add the trip to the database with the understanding that it is a newly created trip.

Once the trip had been added, they continued on to add specific information pertaining to the trip. This information involved different campsites and their locations as well as confirmation numbers, transportation arrangements, and the number of supplies. These tasks may be slightly misleading for in real life, they would not be entered subsequently. However, the test was focused on looking at navigating to the pages and the organization of information.

These tasks were spread out to equally represent each section of the system. The user was started off on a random page so that the navigation of the system can be fully tested. Any excessive amounts of time navigating to find a correct page should be noted for this will slow down the process of completing any task.

Similar to the heuristic evaluation, the user was asked to "think aloud". This gave the experimenter some insight into the user's thought process. This in turn can be used to analyze the different observations and results from the tests. The experimenter took notes on obstacles the user encounters as well as comments of particular relevance.

The user was provided with all of the information that needed to be entered into the system. This was due to the fact that in real life the users will be taking information from other sources and be entering it into the system. Therefore, there was no need to make them make up the information as this situation will never occur in real life.

At the end of the test, the user was asked to complete the post test survey shown in Appendix A.3. The post task surveys were not administered as part of this user test due to the
large number of tasks but brief nature. Also unlike the heuristic tests, it was natural for the coordinators to flow between these tasks interchangeably without any break.

The observations and surveys were then analyzed by the experimenter to determine problems with the system. The experimenter noticed places where the user spends an excessive amount of time looking for the right page, where long pauses occurred when completing tasks, or when confusion or aggravation was expressed by the user. These were indications of places the system needed to be worked on and may determine whether another round of user testing is necessary before implementing the prototype.

5.5 User Evaluation Results

The user evaluation was done on the administration portion of the system. As discussed earlier, this part of the system is all encompassing as the coordinators may have to complete the same tasks as the first-year students or COOT leaders. This group was also easier to test as the users are a small, defined user group, who are easy to locate. For this test, Leanne Burnham, Assistant Director of Student Activities, who is the staff member in charge of COOT, completed the tasks explained above.

The test began and there was immediate confusion on where to begin with the system. As with the heuristic evaluation, the system was purposefully not brought up on the correct page, so that the navigation of the user could be watched. Leanne started to jump around and ended up in the portion of the system designed for adding a leader. She began to enter the information that was part of the second task and then realized she was not in the right place. She explained that she had initially read the directions wrong while navigating to the correct page to make the trip assignment.

The trip assignment section of the system is a multi-step section. The first step is identifying and searching for the COOT leader you wish to assign to a trip. When Leanne got to this page her reaction was “I don’t need to search” but then continued on to using the page with no difficulties. It was interesting to note that despite the fact the search page says you do not need to enter information in all of the fields, she did so, as the information was provided on the sheet. This slowed down the desired time in completing this task.

The next decision involved whether to use the “Add Leader” or “Modify Contact Info” button to add a summer address into the system. The directions purposely did not use the word
"modify" to see if it was intuitive as to which direction in which to go. Once again, Leanne was presented with the search feature in order to identify the right COOT leader, but this time breezed through it and then had no problem adding the information given.

After completing this section, the test moved into the COOTers fields where emergency contact information was given. Once again, it was not specified whether this first-year student was already entered into the system. Leanne chose to use the modify page and then added all the pertinent information. She was also given other information that the system did not store in order to see her reaction. She made a comment about how there was no place to store information such as address and then moved on as it did not seem relevant.

It was interesting to note that in the heuristic evaluation many users had problems with adding information about the transportation for each trip. However, once Leanne got to this task, she quickly filled out the correct fields with no problems. This may be due to the fact that this page closely resembles the previous system so it works for her but is not intuitive for others.

Some shortcuts and abbreviations were also noticed during the testing. When the information repeated in the campsite section, Leanne used the copy and paste feature in order to save time. In the supplies section, she abbreviated the word for money into the symbol and described the number of tents differently than the presented on the instructions. These abbreviations and shortcuts seemed as though they had been picked up over the past few years from working with the current system, so were a natural reaction for her. The shortcuts that were used may be taken into consideration when modifying the fields as to make the information requested more obvious.

Each task was also timed as it was completed. It was a bit hard to time the first task due to the interweaving with the second task but once Leanne was on the correct page it took about two minutes. Adding a new trip took under a minute as did adding the transportation information. Tasks that involved using the search screen took longer than expected, so while both were completed in less than five minutes, it was not as efficient as I had planned. Adding the information about the supplies took about two minutes due to the large amount of information that needed to be entered. Lastly, adding the campsite information took slightly longer than two minutes.

These times are helpful in judging how successful the system was according to the usability goals originally set. Though supplies and campsite information were border line time-wise, there
was no indication of confusion on using the system. Leanne appeared to be extra cautious in order to make sure she entered the information correctly, as she was not informed that it was timed.

From the survey following the test, Leanne once again indicated her mistake in misreading the first question. Overall, she ranked the system easy to use and noted that she did not encounter any errors while completing the tasks. As noticed, she felt the most difficult part of the system was searching for students (either first-year or COOT leaders) in order to changing their information or adding them to a trip. She also noted that the tabs were easy to work with and navigate through the different sections of the system. Lastly, unlike the users in the heuristic evaluation, she would like a help tab for reference in case errors do occur or for quick reference.

Through this round of user testing, it seems as though the user felt the search part of the system for locating records to modify is not efficient. However, what Leanne did not realize is that when using the system currently in place she utilizes the same type of search; it is just not as noticeable a step in the process of completing a task. Therefore, the search feature may need to be enhanced but can not be eliminated from the system.

Other options to fix this problem would be to have a drop down menu with every person’s name in it that is in the system. However, this may get to be a bit much due to the number of students who register to go on COOT trips. Therefore, it seems that although the user may not understand why a search needs to take place it is a crucial section of the task. Perhaps then the search page needs to be modified or clarified so that the user understands that not all three fields (ID, first name, and last name) need to be filled out in order to locate a student. These are all options which should be looked and revised for further testing.

5.6 Modified Prototypes

Both the heuristic evaluation and user evaluation teach the designer a lot about the system. The designer must then take this feedback and associate how to make changes appropriate for the system. With the heuristic evaluation, these changes might be easier as the test users can often suggest alternative ways. The user evaluation often shows major flaws with the flow of the system which causes the user and task analysis to be updated. In both cases, small minor changes are often noticed as well. This section highlights the major changes that were made to parts of the system.
5.6.1 First-Year Students

The major change from the original screen shot (Figure 5-4) is the new buttons at the bottom of the page. These buttons are changed from the original ones which caused problems during the heuristic evaluation. Other changes include adding a second street address line for those people with apartment numbers, etc. Lastly, the last two questions were changed from check boxes to radio buttons, as radio buttons only allow you to choose one option instead of multiple ones.
In Figure 5-37, the most dramatic change is the phone number boxes. These boxes used to be one field (see Figure 5-5) but caused numerous expert users confusion on the format in which to enter the data. When the new format was tested out, the test ran smoothly with no confusion. Also, the name and relationship to you fields have been resized to as to indicate the amount of information that should be entered into each.
5.6.2 COOT Leaders

When the COOT leader application was originally planned, all of the questions appeared on one page. This made for an extremely long application with not a whole lot of room to answer the questions. It was decided that a better approach would be to make it a multi-page application allowing for adequate space for each question as shown above in Figure 5-38. This new version of the application tested out well in the heuristic evaluations and only the navigation between pages, as discussed in the first-year student section, needed to be revised.
5.6.3 Administration

The search part of the first-year and leader information caused some problems during user testing. The original design of the search page (Figure 5-20) was very hard to understand. Figure 5-39 shows an updated version of this same page that looks cleaner and worked very well in the user testing, despite the confusion as to why a search was being performed. Further changes are still being looked at for this page but this is a step in the right direction.

Figure 5-39 – Administration Page to Modify First-Year Student Info

Trip Supplies

Enter the quantity and description, if necessary, for the following supplies needed for the selected trip:

- Tents:
- Stove:
- Cannons:
- Life Jackets:
- Paddles:
- Rubbermaid Bin:
- Misc:

Check the supplies that are required for the trip:
- Medkit
- Water Purification
- Atlas
- Bathroom Kit
- Pot
- Map
- Compass
- Whistles

Figure 5-40 – Administration’s Page for Trip Supplies
This page underwent dramatic changes after a heuristic evaluation. Originally, as shown in Figure 5-35, there were numerous drop down menus. One of the original heuristic evaluators found these extremely confusing and suggested check boxes instead. This change was made and then tested in both further heuristic evaluations and in the user testing. In both cases, there were no mistakes in completing the tasks and the user's time decreased from the previous system.

There are numerous other changes that went into each section of the system. This section simply highlighted a few major changes that had noticeable impact on the users in future tests. As always, more testing can be done and these prototypes could be further developed before implementation occurs. However, at this point we have reached a satisfactory usability status, as most of our usability goals have been reached.
Chapter 6 - Architecture Design

Architecture design refers to the plans to build the system that will support the interface designed in previous chapters. For a database, this means organizing the data into an entity-relationship diagram, and then converting this diagram into tables. The next sections look at the decisions that were made when constructing the entity-relationship diagram as well as the tables. The changes that were made to the tables due to the data normalization rules are also discussed.

6.1 ER Diagram

The information gathered from the user and task analysis needs to be organized to fit into groups that will form the entity sets of the ER diagram. In order to organize these entity sets, it is easiest to look at each user group separately. This will allow us to see the information each different user group will enter into the system.

One group of users is the first-year students. First-year students are using the system primarily to enter information about themselves. This information provides details about their address, medical history, emergency contacts, and trip preferences. Since all first-years registering are asked to provide this same information, it seems relevant to group all of it into one entity set, which we will call COOTer. However, since there is more than one emergency contact needed for each COOTer this would be best split off into another entity set. So by examining the first-years information, two entity sets have been defined: COOTer and Emergency Contact.

Since each emergency contact is related to a specific entity of the COOTer entity set, a relationship will need be formed between the two. From the user and task analysis, it has been found that each first-year student provides two emergency contacts. The first-year student provides the information regarding their emergency contact through the application. Since this emergency contact is associated with only one first-year student, a one-to-many relationship seems to be the best choice. This relationship can be seen in Figure 6-1 below. However, a person may be an emergency contact for more than one first-year student. It is unlikely this would happen in the same year, but still a possibility. Since the first-year student has no knowledge of whether or not the emergency contact has already been entered into the system by another student, the one-to-many relationship will still be the most adequate choice.
The next user group to look at is those Colby students applying to be COOT leaders. The information these students enter into the system is information about themselves, such as address at school and during the summer, and answers to the questions on the application. This group can constitute another entity set called Leader. These students applying to be a COOT leader are often former COOTers. However, since the information gathered on COOT leaders and first-year students is dramatically different a connection between the two is unnecessary. Therefore since there is no connection between leader and the previous two entity sets defined, no more relationships need to be added to the diagram.

The administration user group has much overlap with the previous two groups defined. The user and task analysis show how the administration may need to enter first-year students and leaders into the database themselves if paper applications are used. Since these two entities are already defined, this will not affect the current state of our entity model.

Another task of the administration is to organize the specific details about each COOT trip. Trips are defined in numerous ways. First they have a type and a level. Trips are also defined by the supplies and the method of transportation for getting to and from the trip. Lastly, each trip has at most three campsites associated with it. From the user and task analysis, it is understood that many trips may be instances of the same trip type and level, so these two entities will become an entity set called Trip Type. Campsites fall into a similar situation. The same campsite may be used for numerous different trips and each trip has at most three campsites. Therefore, another entity called campsites can be created. The rest of the information compromising trips will be attributes of the base entity set, Trip.

These three newly-formed entity sets seem to have an association with one another. For
The main page is very hard to read with the main font in bold and the headlines as the same font but not bold faced. Links to download PDF versions of all registration forms are also found on this page. The layout and fonts are not consistent from page to page which causes some confusion as to which pages are related to MOO and which pages are related to the February Outdoor Orientation (FOO) program. Lastly, the packing lists for all of the trips are on one website. So, if you try to print one, unless you know how to print only a section of a page, you will get all of the trips’ packing lists.

Figure 4-5 - Middlebury’s MOO Main Page

Tufts orientation program like many others is run by the college Outing Club. It contained a lot of the information that was common between the sites. In this case, the menu bar was only an inch wide which for a menu is extremely small (Figure 4-6). The other unusual part of this website was the fact that the registration sheet was part of a website instead of a downloadable PDF file (Figure 4-7). Normally, this would be a good idea to make the user experience easier. However, there was no easy way to print the form unless printing the entire web page which then caused a page break in the middle of the form. This page break makes the form seem very unofficial and could create hesitancy for users trying to use it (of whether it is valid or not).
instance, each trip is defined by a specific trip type but the same trip type may apply to many different trips. This will form a one-to-many relationship between the trip and trip type entities. It is also known that each trip has at most three campsites for each of the nights during the trip. This suggests an association between the trip and campsites entity. The relationship would be a many-to-many relationship since a campsite may accommodate numerous different trips and each trip may stay at up to three different campsites during the duration of the trip. These newly formed relationships and entity sets are depicted in Figure 6-2.

Additionally, entities in the leader entity set are people chosen to lead a specific COOT trip. This creates another association with the trip entity. Each trip has at least two leaders, some have three. Students chosen to be a COOT leader are often COOT leaders in future years though not necessarily for the same trip. Therefore, the relationship will need to be a many-to-many relationship, shown below in Figure 6-3.
Finally, there is also a relationship between the first-year students and the trips. Each student is asked to rank their top three trip choices. Students are subsequently assigned to one trip. Each trip will have anywhere between ten and twenty first year students on it. Since there are two different actions being performed within the system, two separate relationships will need to be formed. These relationships can be seen in Figure 6-4.
The entity-relationship diagram shown in Figure 6-4 is the final version for the COOT database. This shows an overview of how the system should be set up and the interactions between the different entity sets, which will eventually be used to form the tables of the database. Since there are so many attributes associated with each of the entity sets they have been omitted from this diagram, but are included in separate diagrams in Section 6.2 below.

6.2 Data Normalization

Once the entity sets have been defined it is easy to transfer them into tables to create a relational database as discussed in Section 3.3. However, these tables might not follow the rules of data normalization. Therefore, it is important to ensure that the tables meet at least 1NF, and ideally up to, 3NF.

The tables in Figures 6-5 through 6-8 were generated using the relationship view in Microsoft Access. Primary keys of the tables are indicated in bold, and relationships are shown with connecting lines. The cardinality of the relationship can also be seen in the diagram by
looking at the number or symbol that appears on both ends of the line. For example, the line connecting the COOTer table to the Medical table in Figure 6-5a, shows two ones on each end of the line, indicating a one-to-one relationship. On the other hand, in Figure 6-5b, the relationship between COOTer and TripChoice, shows a one and an ∞, demonstrating a one-to-many relationship.

In Figure 6-5a, the tables show the original arrangement for the COOTer and Emergency Contact tables. The line connecting the two tables indicates a one-to-one relationship. However, upon first glance it is evident that this does not satisfy requirements for 1NF, so some alterations must be done to the tables and their relationships. The emergency contact table violates the 1NF rule because the same information about the two contacts is stored within two separate fields. To fix this problem, the relationship between COOTer and EmergencyContact is changed to a one-to-many relationship and the fields are condensed to store the name, relationship, and phone numbers of the emergency contacts. This change is shown in Figure 6-5b.

A similar problem occurs with the TripSelection table in Figure 6-5a. TripChoice is a repeating field, which violates 1NF. So, TripChoice is broken off into a new separate table as seen in Figure 6-5b. The relationship between COOTer and TripChoice is changed to be a one-to-many relationship. In the TripChoice table a new field called Rank is created, so the order of trip choices can be saved. There is also a compound key composed of COOTerID and TripChoice in the TripChoice table. COOTerID is a foreign key to the COOTer table while TripChoice is a foreign key to the TripType table.

It was also discovered that, since the relationship between Medical and COOTer is a one-to-one relationship, it can be condensed into one table. All of the information stored in the Medical table is directly related to COOTerID. Since COOTerID is also the key for the COOTer table, it makes sense to combine the two tables. In addition, the fields from TripSelection, ExpScale and Experience, are also needed only once for each COOTer. All of these fields were then condensed into the COOTer table, as shown in the Figure 6-5b.
A similar data analysis was done on the Leader table as shown in Figure 6-6a. The original table stored all of the personal information about the leaders, any previous trip they had lead, as well as the current trip they were leading. However, this was limiting the users to only entering one previous trip per leader rather than all such trips. To fix this problem, a new table called PreviousTrips was created and related to the leader table with a one-to-many relationship between the LeaderID and ID located in the Leader table. PreviousTrip held the ID of the trip, the year, and the leader number. The field tripLeading in the Leader table then became a derived attribute based upon the year.

Figure 6-7 All tables and their relationships in normalized form
Once these changes had been made a full view of the tables and their relationship was constructed as shown in Figure 6-7. These tables and relationships were once again evaluated to ensure that they satisfied the rules of data normalization, as well as holding the necessary information correctly and accurately. In order to achieve this, some tables had to be denormalized.

It was decided that the TripChoice table was better off not normalized and so some repeating fields were added back in. Since there were only three repeating fields it was not a huge amount of information that was being replicated. Also, by keeping these three fields, it was easier to pull the trip choices for the first-year students, in the administration interface so that they could be assigned appropriately to the right trip.

In the Confirmation table, three yes/no fields were added in order to clarify the data. In Figure 6-7, the fields in the tables did not allow the user to determine the nights that the campsites had confirmed being booked for. These yes/no fields corrected this problem. Also incorrect in Figure 6-7, the COOTer table is related to the Trip table by the field TripID. Instead, this relationship should exist between the COOTer and TriplInfo tables. Since the records held in the TriplInfo table are year specific, these records creates a better relationship with the first-year student records which are also year specific. Lastly, a date field was added to the TripChoice table. This supports the ability to monitor and assign trips on a first come, first serve basis.

These corrections or modifications were the last changes to the data model. It is the set of tables seen in Figure 6-8 that need to be implemented. Though not all of the tables follow the rules of data normalization, it is important to realize that these rules were carefully considered and applied where appropriate.
Figure 6-8 Tables after denormalization
Chapter 7- Conclusion

The goal of using a user-centered design approach is to create an interface that is easier for the user to use. In this project, the users were integrated in the development from the very beginning of the design phase. It began by investigating who they were and what they wanted to accomplish and then progressed to building and testing prototypes.

From the design phase of the project, I learned that the users and their tasks are not always as obvious as one would like. By having the designer observe and interview the client, the designer get a much better feel for the project. This makes the designer understand what the users are trying to accomplish and gives the designer a connection to the project. This connection makes the project more significant and pushes the designer to strive for the best possible solution to designing the interface.

In the prototyping phase, it is a chance to throw out any ideas that exist for creating the system. Since I was the only one designing this system, it was very easy to become trapped and not see the other options for designing a system. Once an idea came into my head, it was easy to just stick with it and run. This, however, quickly ended by the time the heuristic evaluations began. These evaluations emphasized the idea that the “designer is not the user”. Parts of the system that I thought were really good turned out to have huge problems with them and needed to be revised and fixed.

By the time the user testing came around, it was important to enter the test with an open mind. The mistakes that were made were beneficial to learning more about the system and the user. It was far better to catch the errors in testing than after the system was fully implemented.

More specifically, the success of the system can be measured by looking at the usability goals set before the testing process began. For the first-year students, we were interested in making sure they could easily locate information about the different trips if necessary. The heuristic evaluations confirmed that this was possible, and many people comfortably browsed through the different trips before making their selection on the application. Through testing it was also learned that some of the questions were worded awkwardly or unclearly. The answers to these questions were still correct on the application; however it was good to learn that these were confusing.
For the COOT leaders, I was testing for overall frustration as well as number of errors. In all tests, both were very low. There was very little confusion what the application was asking for and the tests only revealed a few minor flaws. In this case, both usability goals were easily reached.

The administration section of the program was by far the largest. The tests revealed some flaws with some of the pages but after the changes were made the user testing ran very smoothly. Information was smoothly entered and right around the time period aimed for in the usability goals. In the testing, there was only one error in navigation to the wrong section of the system mainly due to an error in reading instructions rather than due to the system.

Overall, the user-centered design process of designing this system was very successful. The administration was very pleased with the system when presented with it and realized its potential. The number of errors in using the system was extremely low in all tests without requiring any training or directions with the system. The decrease in entry time will now allow the administration to focus on other aspects of the program. The user-centered design process was beneficial for all user groups in making their tasks easier and more efficient to complete.

The strong interface was then complemented by a well thought out and designed implemented system. The database design was carefully planned by utilizing much of the information gathered through the user-centered design process. This allowed me to plan the tables with a greater understanding of what information needed to be stored, how the different information was related, and the various combinations of information that needed to be retrieved.

From this project, I learned the benefits of weaving together a user-centered design and database project. It was extremely useful to be able to have a thorough understanding of the users and their tasks when designing the database. The two disciplines in computer science complement one another extremely well.
Chapter 8 – Future Work

There are numerous aspects of this project that could be further developed if time permitted. As seen earlier, the user-centered design cycle is a continuous cycle and could ultimately go on forever. The same could be said about data normalization for the database tables. However, in both cases, a stopping point needs to be reached in order to make progress.

The next step in this project would be to do some user testing with both the first-year students and the potential COOT leaders. The first-year student user group is slightly hard to do user tests with due to the group’s large size, yet would still be beneficial to the system. The COOT leaders group is much easier to do user testing with as Colby students can be easily found on campus. These test results may show different problems than were discovered in the heuristic evaluation.

The database component also needs much work. The tables were built but never connected with the interfaces. This step is fairly easy as most of the PHP code is already written for the pages, so the mySQL simply needs to be added. However, the wrong version of mySQL is installed on the current server, which does not allow the tables to be correctly linked with foreign keys.

The administration section is still missing the Printable Reports section. This part of the system is how the administration is able to generate reports based on the data within the system. These reports include COOT leaders and their assigned trips, first-year students and their assigned trips, mailing labels, etc. The new system can not be used in the real time environment if this section is not developed. In order to develop this section, numerous more cycle of the user-centered design process would need to be done pertaining specifically to the reports.

At the beginning of this project, there was also discussion about allowing first-year students to view their trips as well as fellow classmates assigned to their trip. By providing this information, students would have the option to bond with others on their trips before actually embarking on their adventure. For many students, this may also ease their transition to college life.

The COOT leaders may also benefit from similar information. Before returning home for the summer, each pair of COOT leaders writes a generic letter to be sent to the first-year students on their trip. It would be great for these leaders to have access to a list throughout the summer,
so that initial contact could be made before arriving on campus. Also, once the COOT leaders arrive on campus, they are constantly looking up information about their first-year students (such as dorm room, email address, etc) that could be easily provided to them from this system.

Lastly, it would be interesting to study whether users prefer a tab navigation or a side menu. The current system has one of each, depending on the section of the system. The tabbed navigation approach is becoming extremely popular in today's society while the side menu is becoming less and less used on web pages. Doing different user tests with both options and then exploring which has better usability would be an interesting experiment. However, the preference in navigation may be due mainly to the user and the tasks they are trying to complete with the system.

Overall, many of these expansions to the system are extra features that would enhance the experience for some particular users. The main parts that are crucial to finish in the future are the implementation of the currently developed interface as well as printable reports for the administration. With these two sections completed, the system would be ready to be put into production.
Appendix A – User Testing Methodology

A.1 Heuristic Evaluation Instructions

Task #1 – Register to go on a COOT Trip
1. Go to [www.cs.colby.edu/~rtnoiseu/COOT/trial.html](http://www.cs.colby.edu/~rtnoiseu/COOT/trial.html)
2. Fill out the form either using your own information or fake information (you may ask for information if you can’t think of any). Note: None of the information entered will be saved after you finish.
3. While filling out the form, please talk aloud and explain decisions that you make, or anything you find confusing.
4. Please print the report you receive after submitting your responses.
5. Once you have finished please fill out the short survey about this task.

Task #2 – Apply to be a COOT Leader
2. Fill out the form as if you were applying to be a COOT leader for the upcoming year. Any information that you do not wish to share may be made up. Note: Information will not be saved after you finish.
3. Use the same approach as above, by talking through each step and decision you make while completing this task.
4. Please print the report you receive after submitting your responses.
5. Once you have finished please fill out the short survey about this task.

Task #3 – Working for the COOT Coordinator
You have been hired as an assistant to the COOT coordinator. He/she directs you to the following webpage ([http://www.cs.colby.edu/~rtnoiseu/COOT/addcooter1.htm](http://www.cs.colby.edu/~rtnoiseu/COOT/addcooter1.htm)) and asks you to complete the following tasks. Please explain what you are doing as you complete each step.

1. A new COOT trip has been added for this year. Please add it into the system. It will be called Acadia Exploration A and is a hiking trip with a difficulty level of 1a.
2. The following campsites have been confirmed for the Acadia Exploration A trip.
   a. Day 1: Camp Fire
      14 Oak Lakes Road
      Rockland, ME 12321
      Conf #:3456 on 6/12/03
      Check #674839
   b. Day 2: Camp-Land
      72 Pine View Lane
      Rockport, ME 54321
      Conf #:12 on 6/12/03
      Check #:345687
   c. Day 3: Same as day 2
3. The transportation for Acadia Exploration A has also been confirmed. It will be one school bus, leaving at 7:15AM and picking up at 10:30AM. This bus will also stop to pick up Acadia Exploration B on the way back.
4. The following supplies will be needed for the Acadia Exploration Trip:
d. Tents: 2 – 6 man tents
e. Medkit: Yes
f. Bathroom Kit: Yes
g. Map: Yes
h. Compass: Yes
i. Pot: Yes
j. Atlas: Yes
k. Misc: $100 travel/emergency money
l. Stove: 2 – Gas Stoves
m. Canoes: N/A
n. Life Jackets: N/A
o. Paddles: N/A
p. Rubbermaid Bin: N/A
q. Whistles: Yes

5. Once you have finished please fill out the short survey about this task.

Task #4 – Survey

Please complete a brief survey about your experience. Thanks!
A.2 Post Task Survey

1. How easy or difficult was it to complete the task? (Circle your answer)
   
   
   
   Very Difficult Neither Easy Easy Very Easy
   Difficult Nor Difficult

   Comments: __________________________________________________________

2. How easy or difficult was the navigation system for the page? (Circle your answer)
   
   
   
   Very Difficult Neither Easy Easy Very Easy
   Difficult Nor Difficult

   Comments: __________________________________________________________

3. How effective were the page labels/terminology?
   
   
   
   Very Unclear Some Clear Very Clear
   Unclear Clear/Some Unclear

   Comments: __________________________________________________________

4. Would you have benefited from online help? Yes No
   Comments: __________________________________________________________
A.3 Post Test Survey

Please rate the following aspects of the system you have just used:

1. Were the instructions, options and commands on using the system easy to follow and use?

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difficult</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Easy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Easy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difficult</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments: __________________________

2. Was the layout and presentation of information clear?

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unclear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Some</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clear/Some</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unclear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments: __________________________

3. Was it easy to move around different parts of the system?

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difficult</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Easy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Easy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difficult</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments: __________________________

4. Did the system allow you sufficient flexibility to work in the way you wanted?

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inflexible</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flexible</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Very</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inflexible</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments: __________________________
5. Was the system helpful in coping with any errors that were made?

<table>
<thead>
<tr>
<th></th>
<th>1 Very Unhelpful</th>
<th>2 Unhelpful</th>
<th>3 Neither Helpful Nor Unhelpful</th>
<th>4 Helpful</th>
<th>5 Very Helpful</th>
</tr>
</thead>
</table>

Comments: ____________________________________________

6. Did you find the system responsive to your inputs?

<table>
<thead>
<tr>
<th></th>
<th>1 Very Unresponsive</th>
<th>2 Unresponsive</th>
<th>3 Neither Responsive Nor Unresponsive</th>
<th>4 Responsive</th>
<th>5 Very Responsive</th>
</tr>
</thead>
</table>

Comments: ____________________________________________

7. What was most difficult to do or understand with any part of the system?

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

8. What was easiest to do or understand with any part of the system?

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

9. What is one thing you would change about the system?

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________
A.4 User Instructions

1. The following COOT leader needs to be assigned to lead Acadia Exploration A. Christopher Johnson 234578

2. Christopher Johnson, ID Number 467829, leading COOT trip Moxie Bald A, needs to have his summer address added to the system. It is as follows:
   50 Winter St
   Greenville, NJ 09798

3. The emergency contact information for COOTer Howard Reagal, ID Number 341793 on Trip: Excursion A has changed. It is now as follows:

   Emergency Contact 1: Sue Reagal
   Mother
   345 North St
   Waterville ME 04901

   Emergency Contact 2: Doug Serdgy
   Uncle
   867 White Plains
   Houston TX 45686

4. A new COOT trip has been added for this year. Please add it into the system. It will be called Acadia Exploration A and is a hiking trip with a difficulty level of 1A.

5. The following campsites have been confirmed for the Acadia Exploration A trip. Please record these details in the system.
   a. Day 1: Camp Fire
      14 Oak Lakes Road
      Rockland, ME 12321
      Conf #:3456 on 6/12/03
      Check #674839
   b. Day 2: Camp-Land
      72 Pine View Lane
      Rockport, ME 54321
      Conf #:12 on 6/12/03
      Check #345687
   c. Day 3: Same as day 2

6. The transportation for Acadia Exploration A has also been confirmed. It will be one school bus, leaving at 7:15AM and picking up at 10:30AM. This bus will also stop to pick up Acadia Exploration B on the way back.

7. The following supplies will be needed for the Acadia Exploration Trip:
   a. Tents: 2 – 6 man tents
   b. Medkit: Yes
c. Bathroom Kit: Yes
d. Map: Yes
e. Compass: Yes
f. Pot: Yes
g. Atlas: Yes
h. Misc: $100 travel/emergency money
i. Stove: 2 – Gas Stoves
j. Canoes: N/A
k. Life Jackets: N/A
l. Paddles: N/A
m. Rubbermaid Bin: N/A
n. Whistles: Yes
References


