

# Zeek2Find

# Zeek2Find

*CIS 491-101*

*Professor Osama Eljabiri*

*The NJIT Library  
Videogame Project*

*Created by:*

*Robert Miller*

*Andrew Luchnick*

*Dominick Badalamenti*

*Nathaniel Hirsch*

*Richard Urban*

*Michael Ronquillo*

# Zeek2Find

1	INTRODUCTION .....	7
1.1	Important Notice:.....	7
1.2	Abstract .....	7
1.3	About the “As-Is” System:.....	7
1.4	Problems with the “As-Is” System:.....	8
1.5	Key Designers in Zeek2Find .....	9
2	PROBLEM STATEMENT .....	10
3	GLOSSARY .....	11
3.1	Development and CASE Tool Definitions .....	11
3.2	General Definitions .....	12
4	METHODOLOGIES .....	14
4.1	Prototyping .....	14
4.2	Win-Win Spiral.....	15
4.3	Use Case Diagram.....	16
4.4	Data Flow Diagram.....	16
5	SDLC METHODOLOGIES EVALUATION.....	17
5.1	Overview of the SDLC Evaluation Matrix.....	17
5.2	The SDLC Evaluation Matrix.....	18
6	LITERATURE REVIEW .....	19
6.1	The LC Call Number Quiz.....	19
6.2	Fake Out.....	20
6.3	Dr. Clue .....	21
7	PRIORITIZATION.....	22
7.1	Table Breakdown of Project Priorities.....	22
7.2	Bar Graph Breakdown of Project Priorities .....	23
7.3	Pie Chart Breakdown of Project Priorities.....	24
7.4	Project Priority Order Ranking.....	24
8	PROJECT MANAGEMENT .....	25
8.1	Project Team and Roles.....	25
8.2	Resources Management .....	27
8.2.1	Work Breakdown Structure.....	27

# Zeek2Find

8.2.2	Milestones .....	29
8.2.3	Gantt Chart .....	30
8.2.4	PERT chart .....	31
9	FEASIBILITY ANALYSIS .....	35
9.1	Technical Feasibility .....	35
9.1.1	Familiarity with Technology.....	35
9.1.2	Familiarity with Project Size.....	35
9.1.3	Familiarity with Application.....	35
9.1.4	Compatibility .....	35
9.2	Financial Feasibility.....	35
9.2.1	Development Costs .....	35
9.2.2	Annual Operating Costs .....	36
9.2.3	Annual Benefits.....	36
9.3	Organization Feasibility .....	36
9.3.1	Resource .....	36
9.3.2	Benefits.....	36
10	COST ANALYSIS ESTIMATION .....	37
10.1	Function Point Estimation.....	37
10.1.1	Total Unadjusted Function Points.....	37
10.1.2	Project Complexity .....	37
10.1.3	Adjusted Project Complexity & Total Adjusted Function Points .....	38
10.2	COCOMO Analysis Calculation .....	38
10.2.1	Lines of Code .....	38
10.2.2	Schedule Time Estimation.....	38
11	RISK MANAGEMENT .....	40
11.1	Technical Risk Assessment.....	41
11.1.1	Identify Uncertainties.....	41
11.1.2	Analyze Risks.....	41
11.2	Risk Control:.....	42
11.3	Risks Facing Basic Elements of Project Success: .....	42
12	STAKEHOLDERS.....	44
13	REQUIREMENTS GATHERING.....	46
13.1	Vord Method .....	46
13.1.1	Brainstorming for View Points.....	46
13.1.2	Hierarchy Diagram .....	47
14	ASSUMPTIONS.....	48
14.1	Assumptions Client.....	48
14.1.1	Hardware.....	48

# Zeek2Find

14.1.2	Software.....	48
14.1.3	Other.....	48
14.2	Assumptions Server.....	48
14.2.1	Hardware.....	48
14.2.2	Software.....	48
14.2.3	Other.....	48
15	REQUIREMENTS WRITING.....	49
15.1	Functional Requirements.....	49
15.2	Non-Functional Requirements.....	49
16	USE CASE SCENARIO DIAGRAM.....	50
16.1	Use Case Specification.....	50
16.2	Use Case Diagram.....	51
17	REQUIREMENTS MAPPING.....	53
17.1	AS-IS DFD.....	53
17.1.1	Context Diagram.....	53
17.1.2	Level-0 Diagram.....	54
17.1.3	Level-1 Decomposition: 1.0.....	55
17.1.4	Level-1 Decomposition: 2.0.....	56
17.1.5	Level-1 Decomposition: 3.0.....	57
17.1.6	Level-1 Decomposition: 4.0.....	58
17.1.7	Level-1 Decomposition: 5.0.....	59
17.1.8	Level-2 Decomposition: 1.2.....	60
17.1.9	Level-2 Decomposition: 3.3.....	61
17.1.10	Level-2 Decomposition: 5.1.....	62
17.2	TO-BE DFD.....	63
17.2.1	Context Diagram.....	63
17.2.2	Level-0 Diagram.....	64
17.2.3	Level-1 Decomposition: 1.0.....	65
17.2.4	Level-1 Decomposition: 2.0.....	66
17.2.5	Level-1 Decomposition: 3.0.....	67
17.2.6	Level-1 Decomposition: 4.0.....	68
17.2.7	Level-1 Decomposition: 5.0.....	69
17.2.8	Level-2 Decomposition: 2.1.....	70
17.2.9	Level-2 Decomposition: 2.2.....	71
17.2.10	Level-2 Decomposition: 3.2.....	72
17.2.11	Level-2 Decomposition: 4.1.....	73
17.2.12	Level-2 Decomposition: 4.2.....	74
18	PROCESS SPECIFICATIONS.....	75
18.1	Structured English.....	75
18.2	Decision Tree.....	77

# Zeek2Find

18.2.1	Login .....	77
18.2.2	Engage Scavenger Hunt.....	78
18.2.3	Administrate Users .....	79
18.2.4	Administrate Scavenger Hunt.....	80
18.2.5	Submit Final Answer .....	81
Decision Table .....		82
18.2.6	Process 1.0 - Login .....	82
18.2.7	Process 2.0 – Engage Scavenger Hunt.....	82
18.2.8	Process 3.0 – Administrate Users.....	82
18.2.9	Process 4.0 – Administrate Scavenger Hunt .....	82
18.2.10	Process 5.0 – Submit Final Answer .....	83
19	ENTITY RELATIONSHIP .....	84
19.1	Entity Relationship Model (ERM) .....	84
19.2	Entity Relationship Diagram (ERD) .....	85
20	STRUCTURE CHART .....	86
20.1	AS-IS Structure Chart.....	86
20.2	TO-BE Structure Chart .....	87
21	SEQUENCE DIAGRAM.....	88
22	STATE TRANSITION DIAGRAM.....	89
23	ABSTRACT MACHINE MODEL .....	90
24	DATA DICTIONARY .....	91
24.1	Data Dictionary.....	91
25	USER INTERFACE DESIGN .....	93
25.1	Users Page.....	93
25.2	Administration Page.....	98
26	SURVEY SECTION .....	100
26.1	Survey.....	101
26.2	Survey Results.....	102
27	SOFTWARE TESTING.....	108
27.1	Introduction.....	108
27.2	Known Issues.....	108
27.3	Test Cases.....	109
28	REFERENCES.....	111

28.1	Publications.....	111
28.2	Websites.....	111

**1 INTRODUCTION**

**1.1 Important Notice:**

When viewing this document in Microsoft Word, you are hyperlinked to each section by the appropriate reference in the Table of Contents. To follow the link, either click directly on the section, or hold CTRL and click on the link and you will be taken to the referenced location in the document.

**1.2 Abstract**

Zeek2Find is an interactive learning tool that will allow for students to acclimate themselves with the research tool at their disposal in the NJIT library. This is done through an interactive, dynamic, web page that places the student at the center of a murder mystery. By answering questions regarding the libraries databases systems as well as general library related questions the user will be rewarded with clues that are used to solve the murder mystery.

Although the project sounds very specific the code has been written general enough to allow for it to be used in many different ways. One such example of how this code can be reused is by making into a scavenger hunt. As opposed to questions about the library the questions can lead the user to places in which the user will have to find something and report back on it. Instead of clues for a murder the clues can lead to the location of the prize that the person has to find.

**1.3 About the “As-Is” System:**

The NJIT Van Houten Library does not have a dedicated learning system for teaching students on how to use the library's resources. Currently, there are only two ways to learn: studying the online documentation and contacting the library staff for consultation.

The online documentation consists of various How-To guides and manuals for the various sections of the library. Guides on searching the periodicals databases exist as well as tutorials on researching information on specific subjects. In addition, general information on library protocol (ex. Call Number Structure, etc.) and general searching (such as Boolean and wild card searching) and paper writing information.

Contacting the library staff is another option for inquiring students. A student can contact a member of the staff via online means (ex. Email, 24/7 Live Chat via AOL Instant Messenger). A student can also contact the library by phone or visit the library and inquire at the front desk in person. Scheduling an appointment for a formal consultation with a subject expert is also available.

**1.4 Problems with the “As-Is” System:**

The resources to teach the new student on how to utilize the databases available to them are lacking and discouraging. The online documentation available on the NJIT Van Houten Library website is terse and confusing and does not adequately leave a lasting impression on the student. It is because of this that the new student will often ignore the documentation and hopelessly fumble around.

Much of the online documentation on the library website is scattered and cluttered. Much of the online tutorials are not easily found and vaguely referred to. Also, the guides on searching



the databases are brief and often confusing for the students not familiar with subjects such as Boolean searching.

Learning to use the substantial database resources of the NJIT Van Houten Library is ultimately an uninteresting and intimidating task for incoming students new to the library due to the unorganized and monotonous documentation and material currently available. Consequently, many new students are never fully taught how to use the periodical databases provided by NJIT and some are not even aware of their existence.

**1.5 Key Designers in Zeek2Find**

<b>NJIT</b>
Robert Miller
Andrew Luchnick
Dominick Badalamenti
Nathaniel Hirsch
Richard Urban
Michael Ronquillo

## 2 PROBLEM STATEMENT

Libraries can be daunting for new students and are not generally look upon as fun and interesting places to augment their education. Since most incoming students are high school graduates, many arrive with memories of their old high school library: boring and full of books. As a university library is much more vast and affluent than the typical high school library, incoming students are not aware of the extensive amount of periodicals and journals available to the eager researcher.

To improve upon the current system of learning, we propose to design Zeek2Find, an interactive web-based scavenger hunt game for the Van Houten Library that is fun, stimulating, and encourages new students to learn and use the resources available to them at the NJIT Van Houten Library.

The scavenger hunt game will be web-based and consist of solving a murder mystery in which students have to find answers to specific questions by researching via the library's database resources and acquire clues in order to find the killer in the murder mystery. Each murder mystery will have a story describing the background and situation of each character involved in the mystery. This gaming environment will give students an incentive to find answers to the questions and ultimately acquire valuable researching skills.

Our goal is to provide an interesting and focused experience for new students and give them a general sense of what their university library can offer to them.

## 3 GLOSSARY

### 3.1 Development and CASE Tool Definitions

#### 1. php

PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.

#### 2. MySQL

MySQL is a relational database management system, which means it stores data in separate tables rather than putting all the data in one big area adding flexibility, as well as speed. The SQL part of MySQL stands for "Structured Query Language," which is the most common language used to access databases.

#### 3. phpMyAdmin

PhpMyAdmin is a tool written in PHP intended to handle the administration of MySQL over the Web.

#### 4. Macromedia Dreamweaver

A web page designing tool that utilizes the Hypertext Markup Language used throughout the Internet.

#### 5. Macromedia Flash

A tool used to create animated movies that can be shown on the Internet.

#### 6. TechSmith Camtasia Studio

A program that takes screen capture movies on a computer, and then saves the movies in various Internet-ready formats.

#### 7. Microsoft Publisher

A helpful tool in designing posters, brochures, and even webpages.

## 3.2 General Definitions

### 1. Administrator

Someone, more than likely a librarian, who is going to login to the system to perform updates.

### 2. Murder Mystery

A game genre in which the user receives clues about a murder and has to piece together the situation to find who the murderer is.

### 3. Scavenger Hunt

A mainly group-oriented genre of game where users race to find items on a list. The winners are those that finish quickest with the most items found.

### 4. Journals

Magazine-like media containing articles that are specific to one occupation or hobby, usually written by experts in that field.

### 5. Online Databases

Used by many libraries, these databases contain journal articles in either abstract form or full-text form.

### 6. Data Flow

Data in motion, moving from one process in a system to another.

### 7. Data Flow Diagram

A diagram of the various logical processes of a system and the data that flows to and from each process.

### 8. Entity-relationship data model (E-R model)

A detailed, logical representation of entities, associations, and data elements for an organization or business area.

### 9. Entity-relationship diagram

<i>Zeek2Find 2004</i>	<h1>Zeek2Find</h1>
<i>CIS 491 Final Project</i>	

A graphical representation of an E-R model.

## 10. Stakeholders

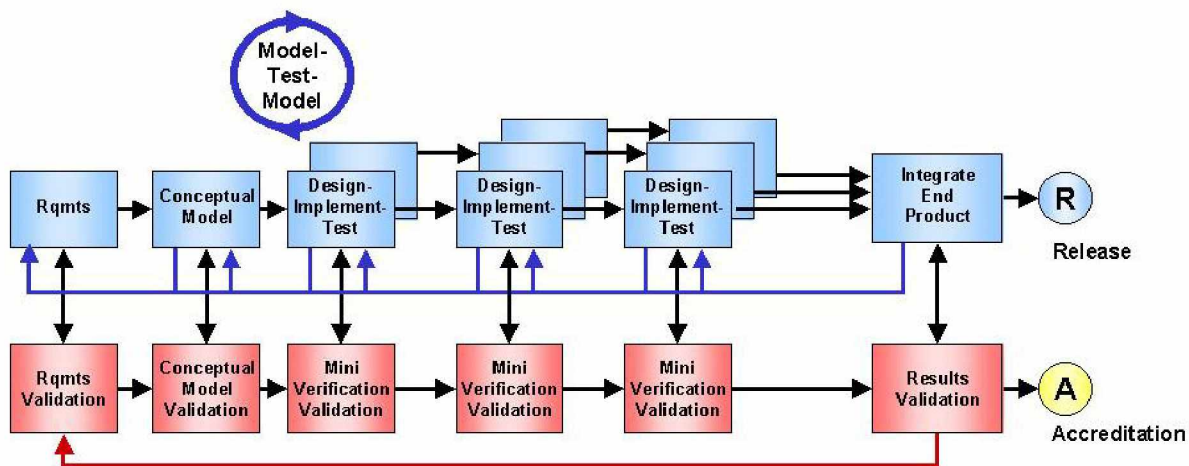
A person who has an interest in an existing or new information system.

## 11. Software Development Life Cycle (SDLC)

The logical process used by a systems analyst to develop and build an information system, including functional requirements, validation, training, and user ownership. An SDLC should result in a high quality system that meets or exceeds customer expectations, within time and cost estimates, works effectively and efficiently in the current and planned Information Technology infrastructure, and is cheap to maintain and cost-effective to enhance.

## 4 METHODOLOGIES

### 4.1 Prototyping



**Prototyping Paradigm**

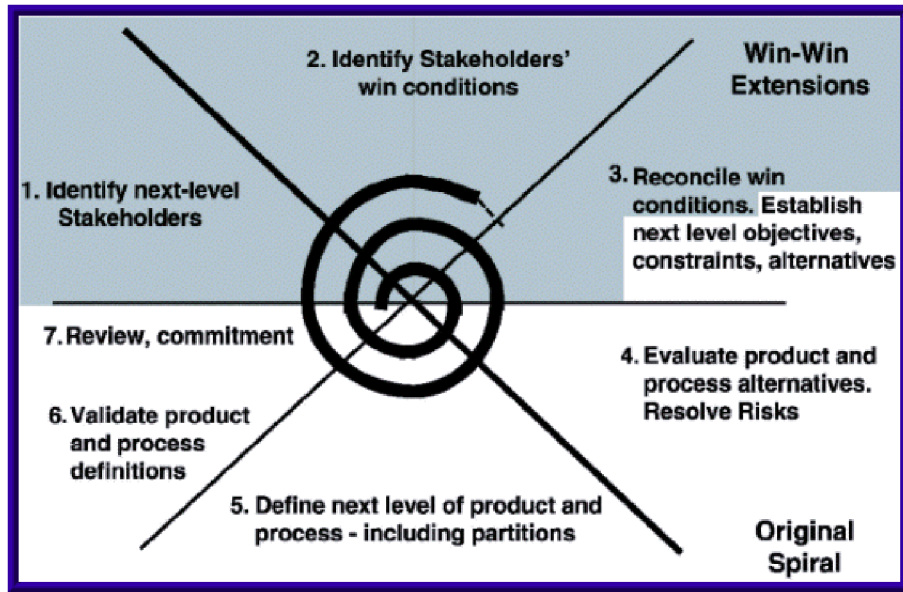
Prototyping is the construction of a model of a system. The prototype allows the developer and the user to test certain aspects of the overall design. It allows them both to test for usability and functionality as well as appearance and usability. After the original prototype is tested the user can state changes that should be made and the programmers can immediately start to implement them. As the prototype changes more of the design specifications are fixed. Prototyping is extremely useful in developing a system.

Our group used prototyping very efficiently. Every two weeks we implemented and developed a new prototype, which was more advanced than the last. This allowed us to show our sponsor exactly where our project was headed and allowed our sponsor to interject with her opinions on what should be changed. Also by using prototyping, a good pace was set, and we

# Zeek2Find

knew exactly what needed to be done every week. Prototyping turned out to be a useful tool for us to use in the Zeek2find project.

## 4.2 Win-Win Spiral



The win-win spiral methodology is a newer technique of software engineering. It expands on the Boehm-Spiral methodology by adding a priority setting step, the win-win process, at the beginning of each spiral cycle and by forming intermediate goals called anchor points. The win-win process identifies a decision point. For each decision point, the objectives, constraints, and alternatives are established and a win-win condition is established. The anchor points establish three intermediate goals (life cycle objective, life cycle architecture, initial operational capability).

This was a very effective technique for our project. We met with our sponsor every week to continually refine our project requirements. This project developed more steadily because it

was an agreement between the sponsor’s needs and wants, and the programmer’s needs and wants. This allowed our group to be creative yet constrained because of the constant input from our sponsor. Also, this technique kept us on track of where and how the project was moving and set a good tone for our project.

**4.3 Use Case Diagram**

A use case diagram is used to analyze the functional requirements of a system without worrying about the implementation of the requirements. It is usually constructed in the early stages of system development and focuses on what the system should do. A use case diagram is made up of actors and use cases. An actor (represented by a stick figure) is an external entity that exchanges information with the system. A use case (represented by an oval) represents related actions started by the actor.

**4.4 Data Flow Diagram**

A data flow diagram is the most common form of a process model. These diagrams use only four simplistic symbols (data flows, data stores, processes, and sources) to represent both physical and logical information systems. It depicts the movement of data (using pictures) between external entities and the processes and data stores within a system. The data flow diagrams then can be broken down even further to show of how the data flows.



**5 SDLC METHODOLOGIES EVALUATION**

**5.1 Overview of the SDLC Evaluation Matrix**

This evaluation matrix shows that for our design needs the best way to manage our project is by using Prototyping and Win-Win Spiral. Prototyping works out great for our needs as the split management approach works very well for the makeup of the group. Having a prototype every two weeks allows our group to see exactly how and what should be changed during each session. This methodology also helps distinguish who needs to do what tasks and by when. This helps with the flow of the project and makes sure everything is done one time.

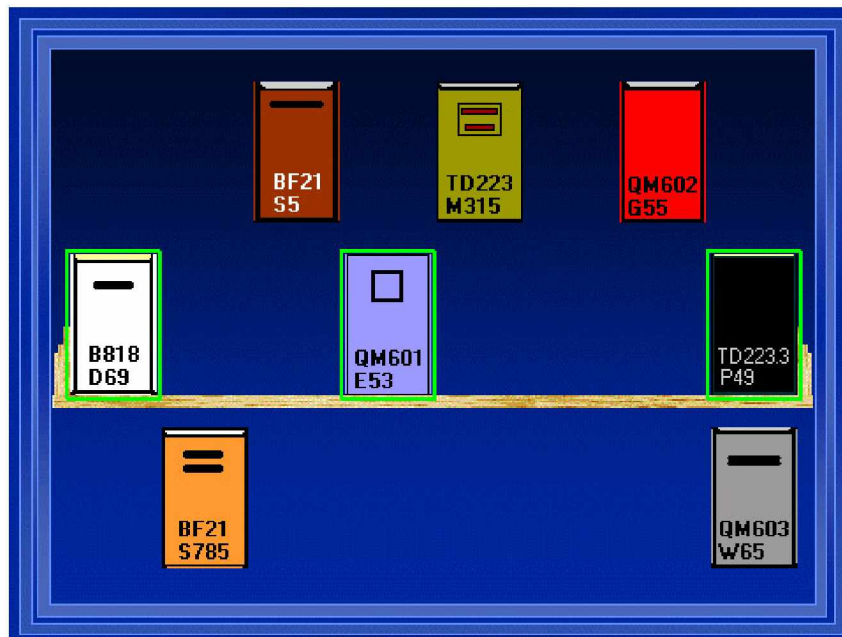
Another methodology that is valuable to our group is Win-Win Spiral. This methodology makes sure our stakeholders and sponsors get what they want. It prevents risk by making sure nothing is done that the sponsor or the programmer does not want. These sessions also help by making the project flexible so that both the programmer and the sponsors are happy.

## 5.2 The SDLC Evaluation Matrix

	Max Points	Waterfall	Extreme Programming	Spiral	Prototyping	Win-Win Spiral
Criteria	Weight					
Rapid Development	30	0	15	10	20	25
Control Schedule	20	20	5	0	20	0
Flexibility of Development	30	0	0	15	15	20
Risk Assessment	10	0	0	0	5	10
Separation of Management	30	5	10	5	25	10
Process Concurrency in Design	5	0	5	5	5	5
Identify Process Stakeholders	5	0	0	0	0	5
<b>Totals</b>	<b>130</b>	<b>25</b>	<b>35</b>	<b>35</b>	<b>90</b>	<b>75</b>

## 6 LITERATURE REVIEW

### 6.1 The LC Call Number Quiz



The first piece of software we examined was actually sent to us by our sponsor, Ms. Scharf. It is called The LC Call Number Quiz, and is a Java applet located online at <http://www.lcsc.edu/library/SatchLCall/>. Right off the bat, we enjoyed this game. It is written to show you the hassles that librarians have to go through to return books to shelves. In this game, you have to put books back, in correct alphabetical as well as call order. We liked the ease of use, intuitiveness, and simplicity of the game. Using this as an original idea for what Ms. Scharf wanted from us, we began to plan for the Zeek2Find system. However, there were a few things we didn't enjoy about the game. It was very easy to cheat in the game. If you were unsure as to where a book goes, you can keep trying until it is highlighted in green. With this in mind, we wanted to make sure our game is impossible to cheat, but it will have hints to help you if you get

# Zeek2Find

stuck. The most distinguishing feature was that it was Java-based, so anyone online is able to play the game. This was important, so we could have our game online, instead of making a cd that would have to be distributed to students.

## 6.2 Fake Out



Using the first game as an example of the genre, we looked up another online java game. We found Fake Out!, located at <http://www.eduplace.com/fakeout>. This is a very fun game where you try to guess the correct definition of a word. We liked this game because there are different levels based on age. We definitely wanted to incorporate levels into our game after playing this one for a while. What we did not like was that although it was fun, it is very much textual. Ms. Scharf told us as a requirement, that Zeek2Find must not be all textual. The best, most distinguishing feature of Fake Out! is the ability for users to submit fake definitions to "fake out" other users. This is important to us, because we want to make sure Ms. Scharf and other administrative level users can add more questions, stories, etc. to our game.

# Zeek2Find

## 6.3 Dr. Clue



In order to get a feel for an online "scavenger hunt" style game, we looked at a few scavenger hunt sites. We first looked at Dr. Clue (<http://www.drclue.com>). We especially liked the fact that instead of collecting items, like on a regular scavenger hunt, you complete complex puzzles and brain-teasers. We didn't like the fact that they charged you to use their ideas, so we stressed the fact that our game was free. The most important feature of this "game" was that it is exciting, you really want to complete the puzzles which are marketed as "brains over brawn".

## 7 Prioritization

For our prioritization of our project we decided to use a point system. We gave each member of our group one hundred points and divided them up into six categories: education, entertainment, difficulty, speed, security, and user interface. The more important the group member thought of the category the more points he would assign to that specific area. We then tabulated a chart and added up all the points for each area and assessed which priorities were the highest and which priorities were the lowest. We also drew two charts to have an easier vision of what priorities were most important. The six categories are defined as follows:

**Education** - students should learn as much as possible from this game

**Entertainment** - students should enjoy playing this game

**Difficulty** – the questions in the game should be of an appropriate difficulty

**Speed** – the game should not take a long time to load or lag between portions

**Security** – the administrative functions of the game should be secure

**User Interface** – the game should have an intuitive appealing user interface

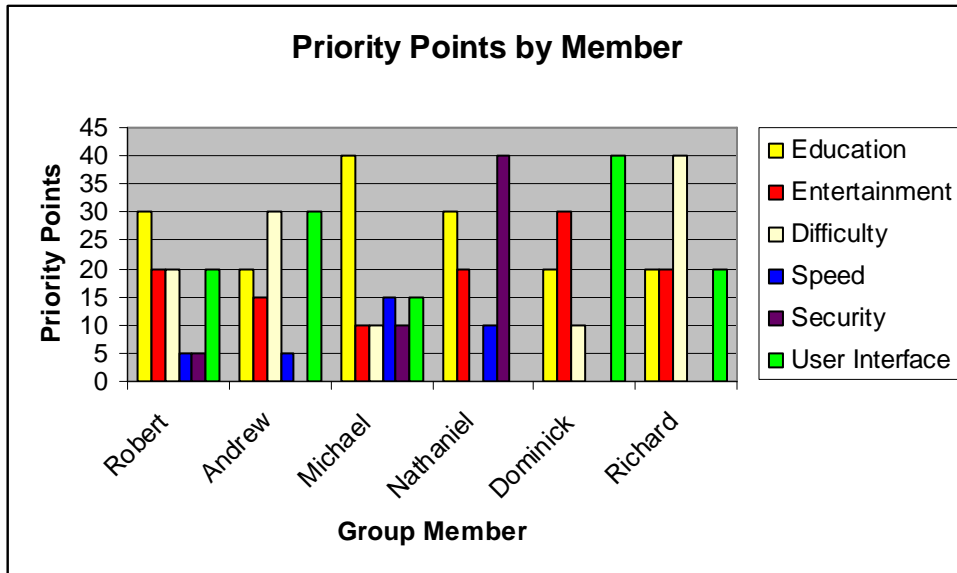
### 7.1 Table Breakdown of Project Priorities

Member	Education	Entertainment	Difficulty	Speed	Security	User Interface	
Robert	30	20	20	5	5	20	<b>100</b>
Andrew	20	15	30	5	0	30	<b>100</b>
Michael	40	10	10	15	10	15	<b>100</b>
Nathaniel	30	20	0	10	40	0	<b>100</b>
Dominick	20	30	10	0	0	40	<b>100</b>
Richard	20	20	40	0	0	20	<b>100</b>
<b>Totals</b>	<b>160</b>	<b>115</b>	<b>110</b>	<b>35</b>	<b>55</b>	<b>125</b>	<b>600</b>

# Zeek2Find

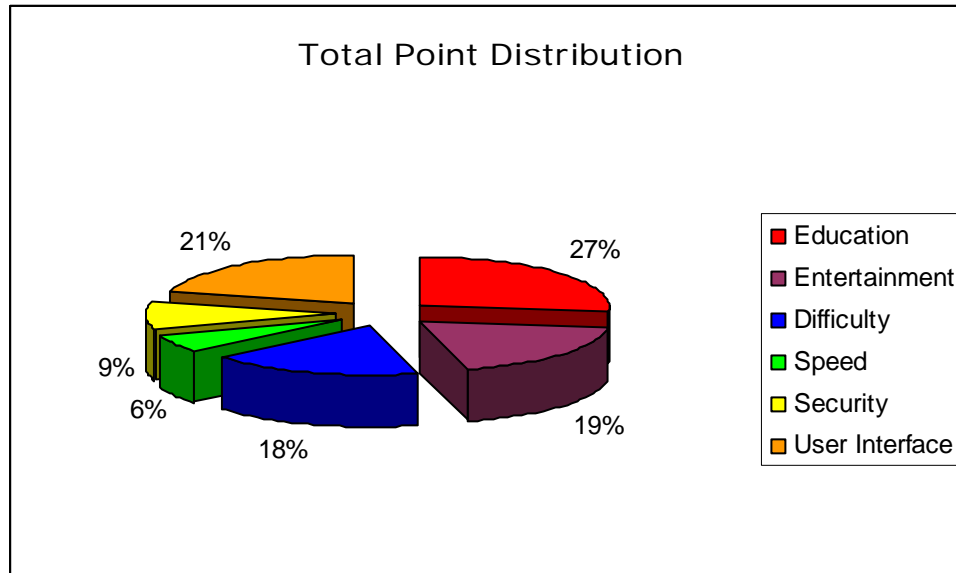
## 7.2 Bar Graph Breakdown of Project Priorities

This graph shows a comparison of each group member's priorities on this project. It is clear that everybody feels that education, entertainment, and a good user interface are important to this project. We can also see that speed and security are not a concern to most group members. The group appears to be split over the importance of how difficult the questions are for students.



# Zeek2Find

## 7.3 Pie Chart Breakdown of Project Priorities



This broken pie chart clearly shows that, once again, education is overall the most important priority in our project. Entertainment and user interface are very close in priority with the user interface coming out just ahead for second highest. These results appear to be reasonable because the game is mainly an educational tool, but we would like to make it as enjoyable for the students as possible.

## 7.4 Project Priority Order Ranking

1. Education
2. User Interface
3. Entertainment
4. Difficulty
5. Security
6. Speed



**8 PROJECT MANAGEMENT**

**8.1 Project Team and Roles**

**Dominick Badalamenti (System Analyst)**

Dominick was the database engineer for the project. He linked the database of questions and answers to the user interface page. He designed and developed questions for the system, and wrote and made up clues for the storyline. He also worked with Andrew with the design of the user interface webpage.

**Nathaniel Hirsch (System Administrator)**

Nathaniel designed and implemented the administration page for the Zeek2Find project. He developed this page for the librarians to add and modify questions to the system. He also designed many security features so that the system would be kept safe from tampering.

**Andrew Luchnick (Programmer)**

Andrew was the primary designer of the user interface webpage. He developed the code for the functionality of the main page. He also helped in the design of how the main page was going to look and how the user was going to interact with the main page.

**Robert Miller (Project Manager)**

Robert was the project manager for the Zeek2Find project. He designated tasks for all the members of the Zeek2Find project team. He set up goals and due dates for all applications and made sure that everything was done well and on time. He also met with the professor to show the growth and progress of the project.

**Michael Ronquillo (System Architect)**

<i>Zeek2Find 2004</i>	<h1>Zeek2Find</h1>
<i>CIS 491 Final Project</i>	

Michael was the stress tester for the Zeek2Find project. He ran tests on the system to see how it would handle certain situations that would arise with the system. He also tested the system for bugs and any errors that might occur within the system.

**Richard Urban (System Analyst)**

Richard helped with the data collecting and documentation of the Zeek2Find project. He developed and compiled results from surveys and other forms of data collection. He also helped with setting up the layout of the main page so that it would look better.

## 8.2 Resources Management

### 8.2.1 Work Breakdown Structure

Task Name	Duration	Start Date	End Date	Predecessor
Group Creation	14d	Tue 8/31/04	Mon 9/13/04	
Project Manager Application	7d	Tue 8/31/04	Mon 9/6/04	
Group Member Applications	5d	Tue 9/7/04	Sat 9/11/04	2
Team Certification Form	2d	Sun 9/12/04	Mon 9/13/04	3
Initialization	21d	Tue 8/31/04	Mon 9/20/04	
Choose Project Idea	14d	Tue 8/31/04	Mon 9/13/04	
Project Approval	0d	Mon 9/13/04	Mon 9/13/04	6
Methodology	1d	Tue 9/14/04	Tue 9/14/04	
Abstract	1d	Tue 9/14/04	Tue 9/14/04	8
Problem Statement	1d	Tue 9/14/04	Tue 9/14/04	7
Brainstorm Ideas For Videogame	7d	Tue 9/14/04	Mon 9/20/04	7
Previous Work	7d	Tue 9/14/04	Mon 9/20/04	7
Research Videogames in Education	7d	Tue 9/14/04	Mon 9/20/04	
Research Scavenger Hunts	7d	Tue 9/14/04	Mon 9/20/04	
Designation of Tasks	1d	Tue 9/14/04	Tue 9/14/04	7
Meet with Sponsor	3d	Tue 9/14/04	Thu 9/16/04	7
Progress Report #1	0d	Tue 9/21/04	Tue 9/21/04	
Prepare for Introductory Presentation	7d	Tue 9/14/04	Mon 9/20/04	
Introductory Presentation	0d	Tue 9/21/04	Tue 9/21/04	
Project Management	28d	Tue 9/14/04	Mon 10/11/04	7
Work Breakdown Structure	24d	Tue 9/14/04	Thu 10/7/04	
Gantt Chart	24d	Tue 9/14/04	Thu 10/7/04	
Pert Chart	24d	Tue 9/14/04	Thu 10/7/04	
Cost Benefit Analysis	28d	Tue 9/14/04	Mon 10/11/04	
Cost Estimation	28d	Tue 9/14/04	Mon 10/11/04	
Risk Management	28d	Tue 9/14/04	Mon 10/11/04	
Project Milestones	24d	Tue 9/14/04	Thu 10/7/04	
Progress Report #2	0d	Tue 10/5/04	Tue 10/5/04	
Prepare for 1st Prototype Release	7d	Tue 10/5/04	Mon 10/11/04	
1st Prototype Release	0d	Tue 10/12/04	Tue 10/12/04	
Analysis	14d	Tue 10/5/04	Mon 10/18/04	
Stakeholders Analysis	14d	Tue 10/5/04	Mon 10/18/04	
Requirements Gathering	14d	Tue 10/5/04	Mon 10/18/04	
VORD Method	14d	Tue 10/5/04	Mon 10/18/04	
Use Case Scenario	14d	Tue 10/5/04	Mon 10/18/04	
Requirements Definition	14d	Tue 10/5/04	Mon 10/18/04	
As-Is Data Flow Diagrams (DFDs)	14d	Tue 10/5/04	Mon 10/18/04	
Functional Requirements	14d	Tue 10/5/04	Mon 10/18/04	
Non-Functional Requirements	14d	Tue 10/5/04	Mon 10/18/04	
Process Specification	14d	Tue 10/5/04	Mon 10/18/04	
Structured English	14d	Tue 10/5/04	Mon 10/18/04	
Decision Trees	14d	Tue 10/5/04	Mon 10/18/04	

<i>Zeek2Find 2004</i>	<h1>Zeek2Find</h1>
<i>CIS 491 Final Project</i>	

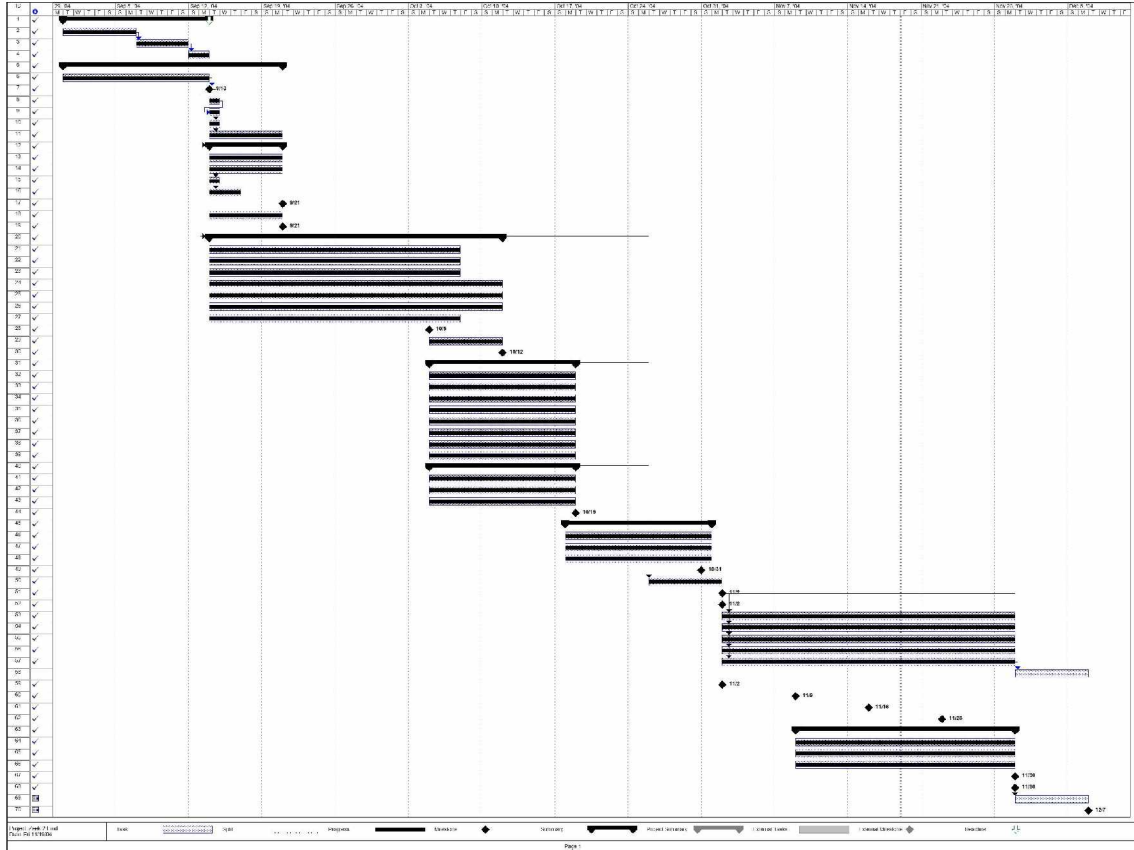
Decision Tables	14d	Tue 10/5/04	Mon 10/18/04	
Progress Report #3	0d	Tue 10/19/04	Tue 10/19/04	
Prepare for Halloween Open House	14d	Mon 10/18/04	Sun 10/31/04	
Open House Brochure	14d	Mon 10/18/04	Sun 10/31/04	
Open House Poster	14d	Mon 10/18/04	Sun 10/31/04	
Open House Presentation	14d	Mon 10/18/04	Sun 10/31/04	
Halloween Open House	0d	Sun 10/31/04	Sun 10/31/04	
Prepare for 2nd Prototype Release	7d	Tue 10/26/04	Mon 11/1/04	"31,40,20"
2nd Prototype Release	0d	Tue 11/2/04	Tue 11/2/04	
Midterm Documentation Report	0d	Tue 11/2/04	Tue 11/2/04	
User Interface Design	28d	Tue 11/2/04	Mon 11/29/04	46
System Design	28d	Tue 11/2/04	Mon 11/29/04	46
ERM Model	28d	Tue 11/2/04	Mon 11/29/04	46
Structure Chart	28d	Tue 11/2/04	Mon 11/29/04	46
System Development	28d	Tue 11/2/04	Mon 11/29/04	46
System Testing	7d	Tue 11/30/04	Mon 12/6/04	52
Progress Report #4	0d	Tue 11/2/04	Tue 11/2/04	
Progress Report #5	0d	Tue 11/9/04	Tue 11/9/04	
Progress Report #6	0d	Tue 11/16/04	Tue 11/16/04	
Progress Report #7	0d	Tue 11/23/04	Tue 11/23/04	
Prepare for Poster Session	21d	Tue 11/9/04	Mon 11/29/04	
Poster Session Presentation	21d	Tue 11/9/04	Mon 11/29/04	
Poster Session Posters	21d	Tue 11/9/04	Mon 11/29/04	
Poster Session Brochure	21d	Tue 11/9/04	Mon 11/29/04	
Poster Session	0d	Tue 11/30/04	Tue 11/30/04	
Progress Report #8	0d	Tue 11/30/04	Tue 11/30/04	
Prepare for Final Presentation	7d	Tue 11/30/04	Mon 12/7/04	53
Final Presentation	0d	Tue 12/7/04	Tue 12/7/04	

## 8.2.2 Milestones

Milestone	Date
Group Formation	9/13/2004
Project Approval	9/13/2004
Progress Report #1	9/21/2004
Introductory Presentation	9/21/2004
Project Management	10/11/2004
Progress Report #2	10/5/2004
1st Prototype Release	10/12/2004
Analysis	10/18/2004
Process Specification	10/18/2004
Progress Report #3	10/19/2004
Halloween Open House	10/31/2004
2nd Prototype Release	11/2/2004
Midterm Documentation Report	11/2/2004
Progress Report #4	11/2/2004
Progress Report #5	11/9/2004
Progress Report #6	11/16/2004
Progress Report #7	11/23/2004
Poster Session	11/30/2004
Progress Report #8	11/30/2004
Final Presentation	12/7/2004

# Zeek2Find

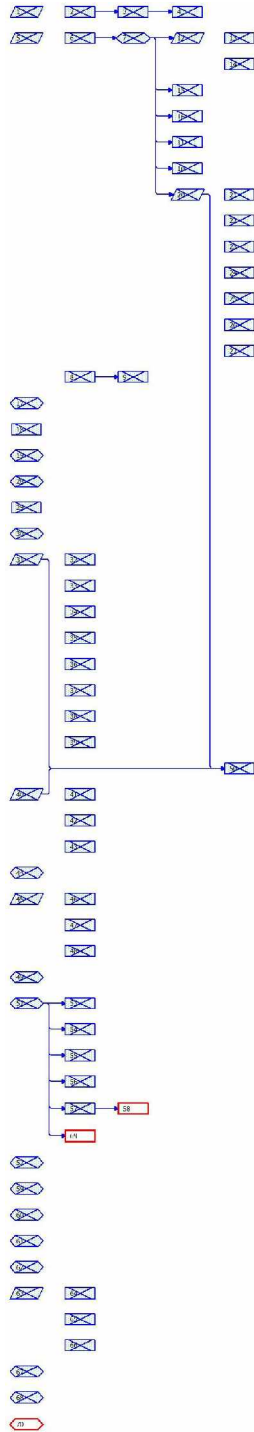
## 8.2.3 Gantt Chart



# Zeek2Find

## 8.2.4 PERT chart

### Entire Chart

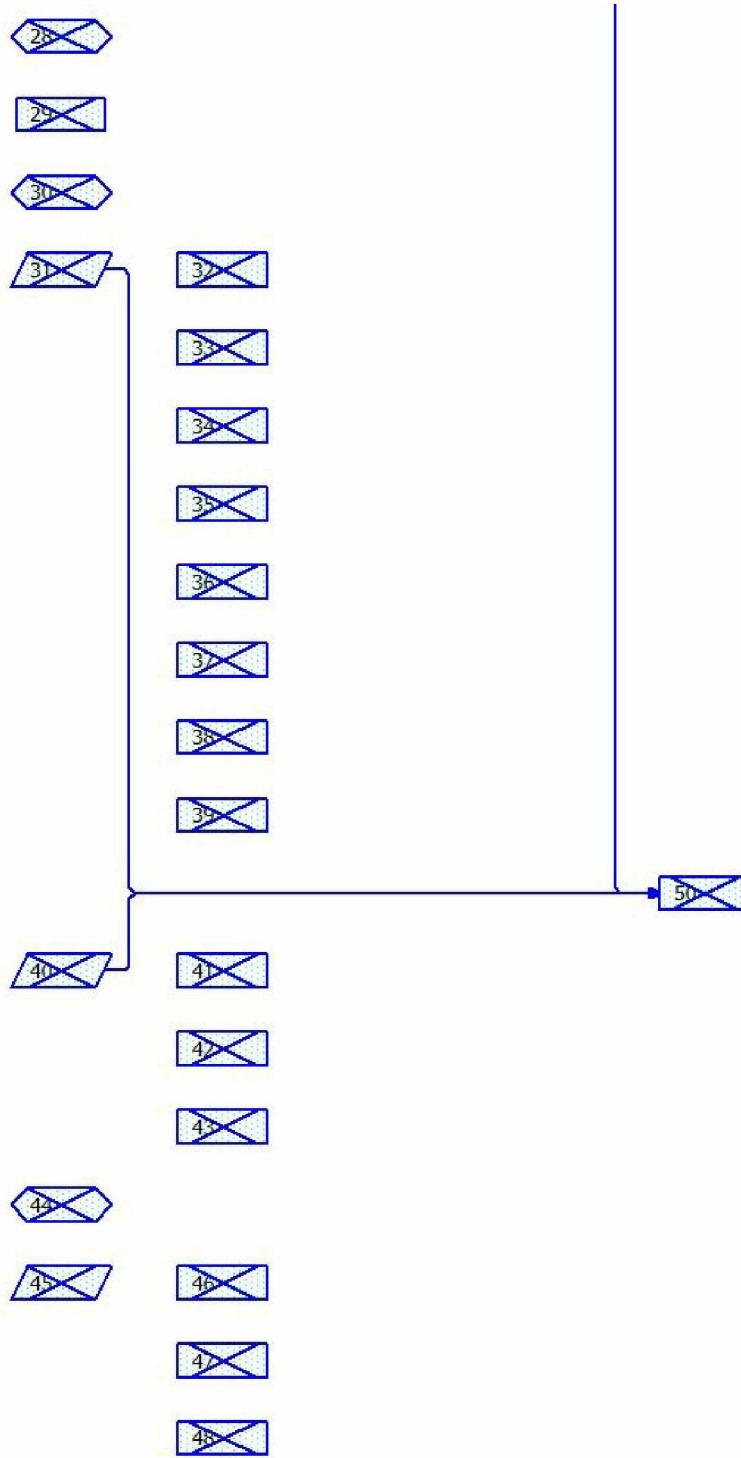






# Zeek2Find

## Upper Section of PERT Chart





**9 FEASIBILITY ANALYSIS**

**9.1 Technical Feasibility**

**9.1.1 Familiarity with Technology**

Administration should have no problem using Zeek2Find when updating questions and stories. The admin page is extremely intuitive and easy to use. Anyone who has filled out a form on a webpage will have no worries updating the system.

**9.1.2 Familiarity with Project Size**

The bulk of the work will be done prior to administrators' use. To the administrator, the project is rather small. If they can run a library, they can use the system with ease.

**9.1.3 Familiarity with Application**

This is an application that will run in web browsers, which most people use daily. It is very similar to filling out a form online, which people who have used browsers, should be completely at ease with.

**9.1.4 Compatibility**

Since Zeek2Find runs as php web pages, it is completely platform independent.

**9.2 Financial Feasibility**

**9.2.1 Development Costs**

The development costs are negligible. The main resource being consumed will be the minimal time it takes to develop the database and code the php pages.

**9.2.2 Annual Operating Costs**

Again, as with the development, the annual costs are negligible. Time is the only resource needed to keep the system going, and that is only necessary when administrators wish to update the system. Otherwise, the system is self-sustaining, and needs no adjustments.

**9.2.3 Annual Benefits**

The benefits can be seen in the students, who will be better researchers, and be much more familiar with the massive amounts of data stored within the library.

**9.3 Organization Feasibility**

**9.3.1 Resource**

The only resources needed are space required on the server, and time to make updates at the administrator’s leisure.

**9.3.2 Benefits**

The main benefit is that students will feel extremely confident in their ability to fully utilize the library as a research tool, making them better students, and smarter individuals. In turn, NJIT will have smarter graduates, which adds prestige to the name and its alumni.

## 10 COST ANALYSIS ESTIMATION

### 10.1 Function Point Estimation

#### 10.1.1 Total Unadjusted Function Points

Complexity					
Description	Low	Medium	High		Total
Inputs	X3	8X4		X6	32
Outputs	X4		5X5	X7	25
Queries			X4	10X6	
	X3				60
Files	X7	6X10		X15	60
Program Interfaces	5X5	X7		X10	25
<b>TOTAL</b>	<b>UNADJUSTED</b>	<b>FUNCTION</b>	<b>POINTS( TUFPP)</b>		<b>202</b>

#### 10.1.2 Project Complexity

	Scale of 1 to 5
Data Communications	5
Heavy Use Configuration	2
Transaction Rate	3
End-User efficiency	4
Complex Processing	1
Installation Ease	1
Multiple Sites	2
Performance	5
Distributed functions	1
On-line data entry	5
On-line update	5
Reusability	5
Operational Ease	4
Extensibility	3
<b>Project Complexity (PC)</b>	<b>46</b>

### 10.1.3 Adjusted Project Complexity & Total Adjusted Function Points

$\begin{aligned} \text{Project Complexity (PCA)} &= 0.65 + (0.01 * PC) \\ &= 0.65 + (0.01 * 46) \\ &= 0.65 + 0.46 \\ &= 1.01 \end{aligned}$
$\begin{aligned} \text{Total Adjusted Function Points (TAFP)} &= \text{PCA} * \text{TUFP} \\ &= 1.01 * 202 \\ &= 204.02 \end{aligned}$

## 10.2 COCOMO Analysis Calculation

### 10.2.1 Lines of Code

Function Points	Lines of Code
PHP	50

LOC:

$$\begin{aligned} C &= \text{TAFP} * 50 \\ &= 204.02 * 50 \\ &= 10201 \end{aligned}$$

KLOC = 10.201

### 10.2.2 Schedule Time Estimation

$$\text{Effort} = 1.4 * 10.201 = 14.2814 \text{ Person Months}$$

$$\begin{aligned} \text{Schedule time} &= 3 * (14.201)^{1/3} \\ &= 7.27855 \text{ Months} \end{aligned}$$

$$\begin{aligned} \text{People} &= 14.201 / 7.27855 \\ &= 1.96212 \text{ People} \end{aligned}$$

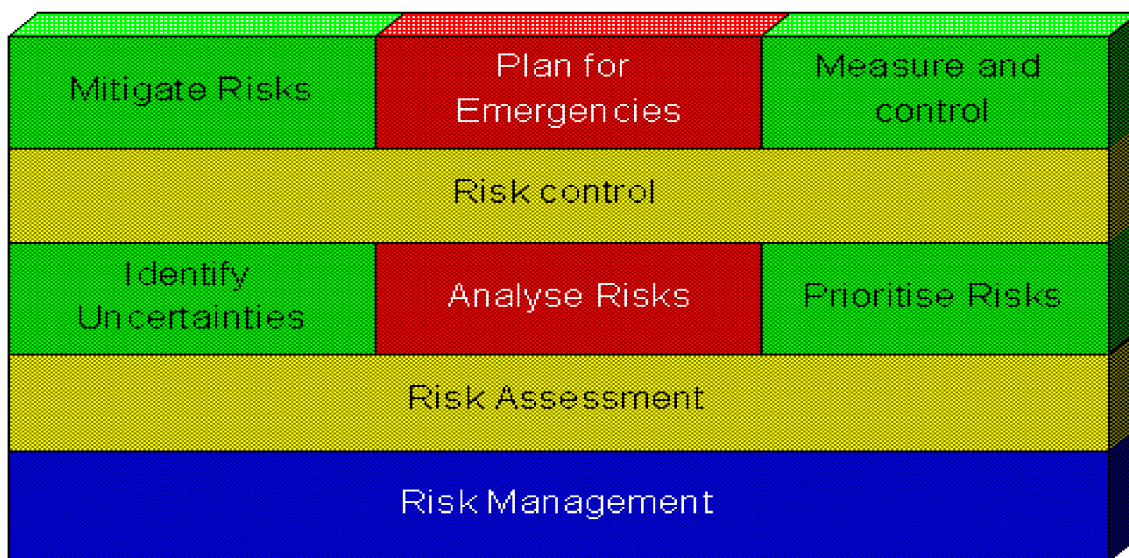
<i>Zeek2Find 2004</i>	<h1>Zeek2Find</h1>
<i>CIS 491 Final Project</i>	

With a well trained staff, it will take 1.96212 people to develop the system. Although many regard CoCoMo as obsolete, we believe that given the fact that this project will be completely finished over the course of the semesters, the schedule time is right on, and, the amount of people determined by CoCoMo to actually do the coding is when rounded up the amount of people we have writing code.

## 11 RISK MANAGEMENT

Effective risk management is made up of risk assessment and risk control. To assess risk we will first identify uncertainties, then analyze the risks, and finally prioritize them. We will then control our risk by mitigating risks, planning for emergencies, and considering how we can measure and control our risk.

### Effective Risk Management



Copyright Coblands consulting Aug 1996



**11.1 Technical Risk Assessment**

**11.1.1 Identify Uncertainties**

- We must decide what web technologies should we use
- We must decide what web server to run on
- We must decide what back end database to use

**11.1.2 Analyze Risks**

**Technology Choice:**

If we choose the wrong technology the project can be a nightmare to implement, and maintain. If no one in the group is familiar with it they time will be lost learning the new technology.

**Web Server:**

If we pick a web server that is non standard the code may not be portable to work on other web servers. Also if the web server is a slow one then the page will appear to be sluggish and will not hold the users attention.

**Database:**

If the wrong back end database is used then the site will appear to be slow. If the database is not easily incorporated into the web technology we choice then it may be a nightmare to implement.

**Prioritize Risks**

The selection on the web server is important, but not as much as the database. This is because most web servers will run the same. And if we make the code standard enough it should work on all web servers. The database is very important, however it really depends on what language we choice to code the site in. Many databases work well with different

# Zeek2Find

technologies so that should be catered towards the one thing that we have the most control over, the language.

The priority of our risks is as follows:

1. Web Technology
2. Back End Database
3. Web Server

## 11.2 Risk Control:

### Mitigate Risks

To minimize our risks we chose to use PHP and mysql. The reason for this is that all most every web server out there supports PHP, and mysql is a very common DBMS that is widely available.

### Plan for Emergencies

In the event that the server we are on can not use PHP then we have to set out to install PHP and mysql on the web server.

### Measure and Control

We plan to control these risks through good communication and research. If any group member recognizes a significant problem with any part of the project is their responsibility to bring it to the attention of the group whether the problem directly relates to their work or not.

## 11.3 Risks Facing Basic Elements of Project Success:

Risk is anything that jeopardizes the success of a project. The success of our project can be measured in the following three categories: scheduling, budget, and user requirements.

### **Schedule**

Schedule constraints present a very large risk. All of us are graduating this semester so anything that is not finished will remain that way. This is a big problem as the system should go live at the end of the semester.

### **Budget**

Since all the tools we are using are available for free on line, and since we are doing all the work in making this project there is little risk in running over budget on this project.

### **User Requirements**

There is great risk in meeting the user requirements for the Zeek-2-Find project. Since the success of this project will be determined by how well it is accepted by the students and librarians it is important to meet the needs of all potential users. If Zeek-2-find is not fun, informative, and easy to use it will not be widely used if at all.

**12 STAKEHOLDERS**

**I. Libraries and librarians**

Libraries and librarians are some of the larger stakeholders of the Zeek2find software. They interact directly with the project. By using this software the librarians jobs are easier. for the students know how to navigate and find things they want in the library easier. Also libraries will be much more efficient because the librarian’s time will be much more well spent.

- a) NJIT library
- b) Various libraries throughout the country

**II. Businesses**

Businesses interact directly with the system as stakeholders. Businesses can utilize this system to improve the learning time of certain programs for employees or customers. The breakdown of the business stakeholders are as following:

- a) Upper management
- b) new employees
- c) old employees using new system

**III. Students**

Students are very important stakeholders of this system. This software allows the students to have a fun and entertaining way to learn how to navigate the library and its

systems. Also, it allows for a speedier more general knowledge of the system and its components. The breakdown of the student stakeholders is as follows:

- a) new students
- b) transfer students
- c) graduate students
- d) students who are new to the library

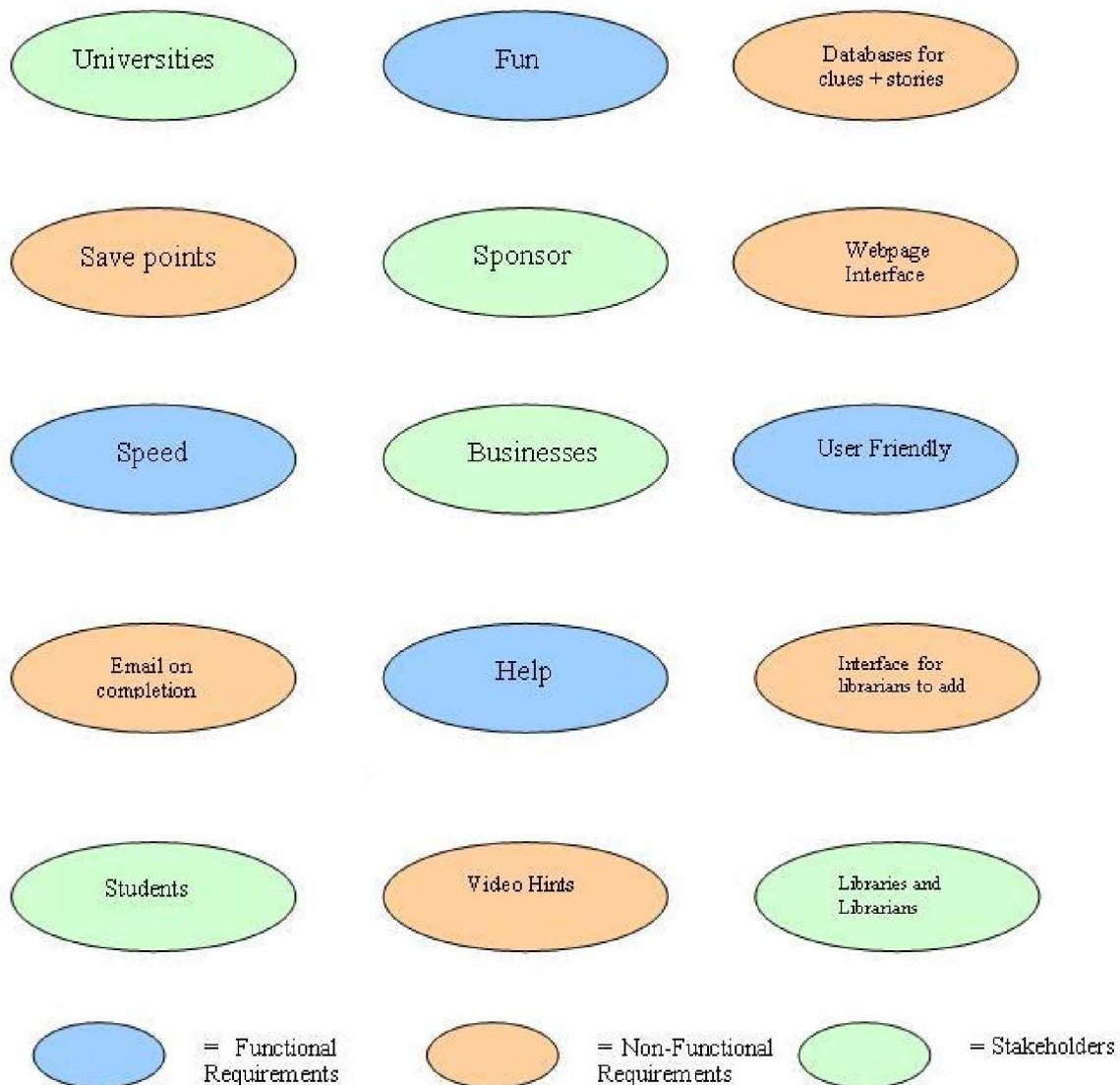
### **III. Sponsor**

The last stakeholder is our sponsor. She has an invested interest in the completion of this project. She will interact with the software directly and choose ways in which the students who use the system will react.

## 13 REQUIREMENTS GATHERING

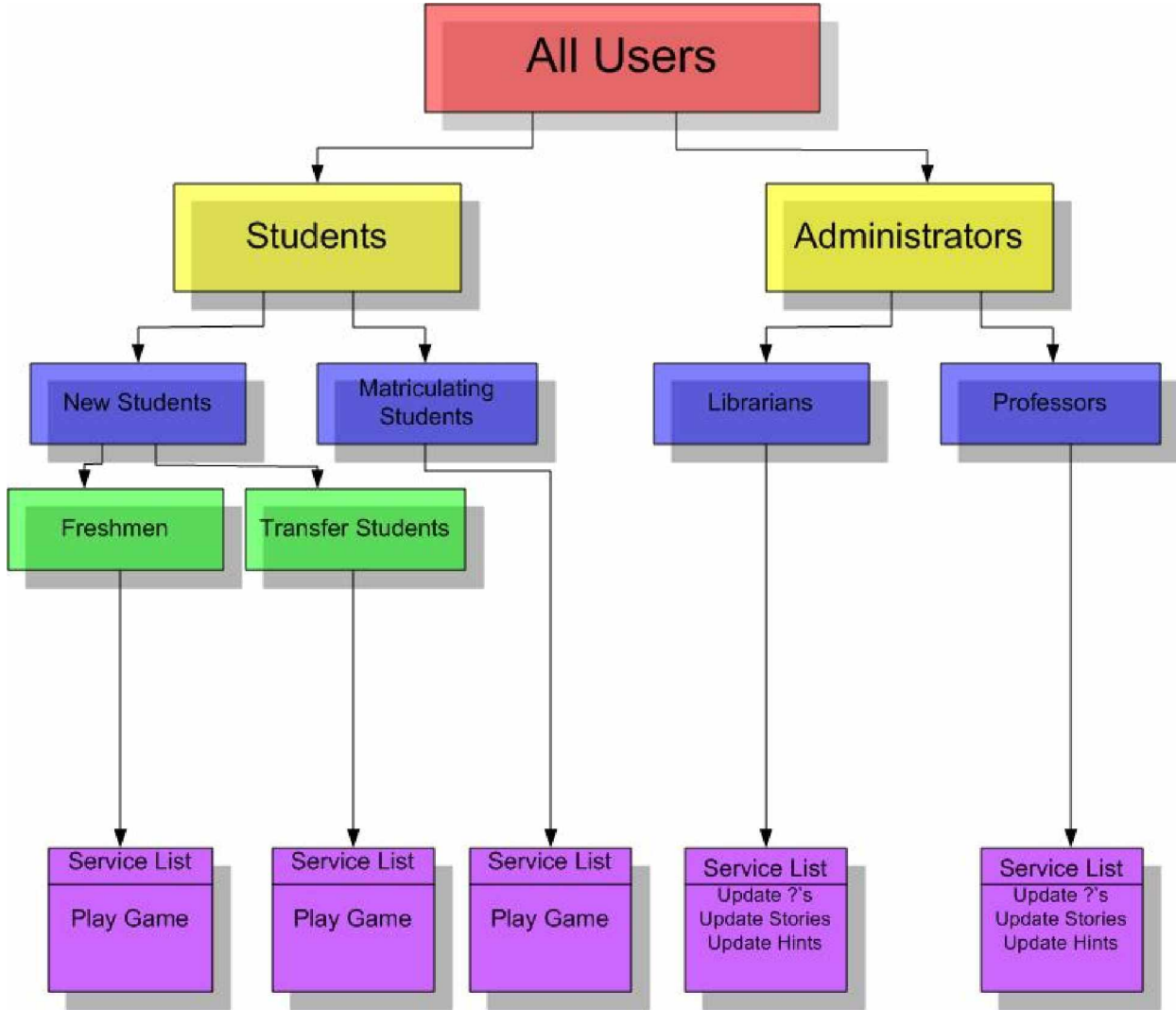
### 13.1 Vord Method

#### 13.1.1 Brainstorming for View Points



# Zeek2Find

## 13.1.2 Hierarchy Diagram



## 14 ASSUMPTIONS

### 14.1 Assumptions Client

#### 14.1.1 Hardware

- There will be a reliable connection to the Internet.

#### 14.1.2 Software

- The user has a modern web browser.
- The user has flash 5 or higher in stalled.

#### 14.1.3 Other

- The user has a good grasp on the English language.

### 14.2 Assumptions Server

#### 14.2.1 Hardware

- There will be a reliable connection to the Internet.
- The machine is fast and robust enough to handle a plethora of simultaneous connections.

#### 14.2.2 Software

- The server has a web server that supports PHP.
- The server is running PHP version 4.x.
- The server has access to a mysql database.

#### 14.2.3 Other

- The admin can come up with stories and questions to ensure the game is entertaining.



**15 REQUIREMENTS WRITING**

**15.1 Functional Requirements**

- Zeek2Find must allow for administrators to login in order to change the system.
- Zeek2Find must keep records of users score.
- Zeek2Find must email administration when user has successfully completed the game.
- Zeek2Find must be fully updatable, to add new stories and murderers.
- Server must have MySQL version 3.x or higher installed.
- Clients must have high speed internet connection.
- Server must run version 4.x of php.
- Client’s browser must support Flash animations version 5 or higher.

**15.2 Non-Functional Requirements**

- Zeek2Find must most importantly be entertaining and informative, at the same time.
- Zeek2Find must run excellently on all browsers.
- Zeek2Find must load quickly.
- Zeek2Find should be intellectually stimulating.
- Zeek2Find must allow for administrators to login in order to change the system.

**16 USE CASE SCENARIO DIAGRAM**

**16.1 Use Case Specification**

**Actors:**

User-	The user, usually a student, who will be playing the game.
Administrator-	The admin, usually a librarian or professor, who will be adding, modifying, or deleting questions, stories, clues, and hints into the game.

**Pre-conditions:**

- Users must not have access to the administration page.
- Administrators must be able to modify every aspect of the game.

**Post-conditions:**

- The system will return exactly to its original state.
- Inserts, modifications, and deletions must take place immediately.

**Basic Flows:**

- The user answers questions to complete to receive hints, and then complete the game.
- The administrator modifies questions, stories, clues, and hints one at a time.

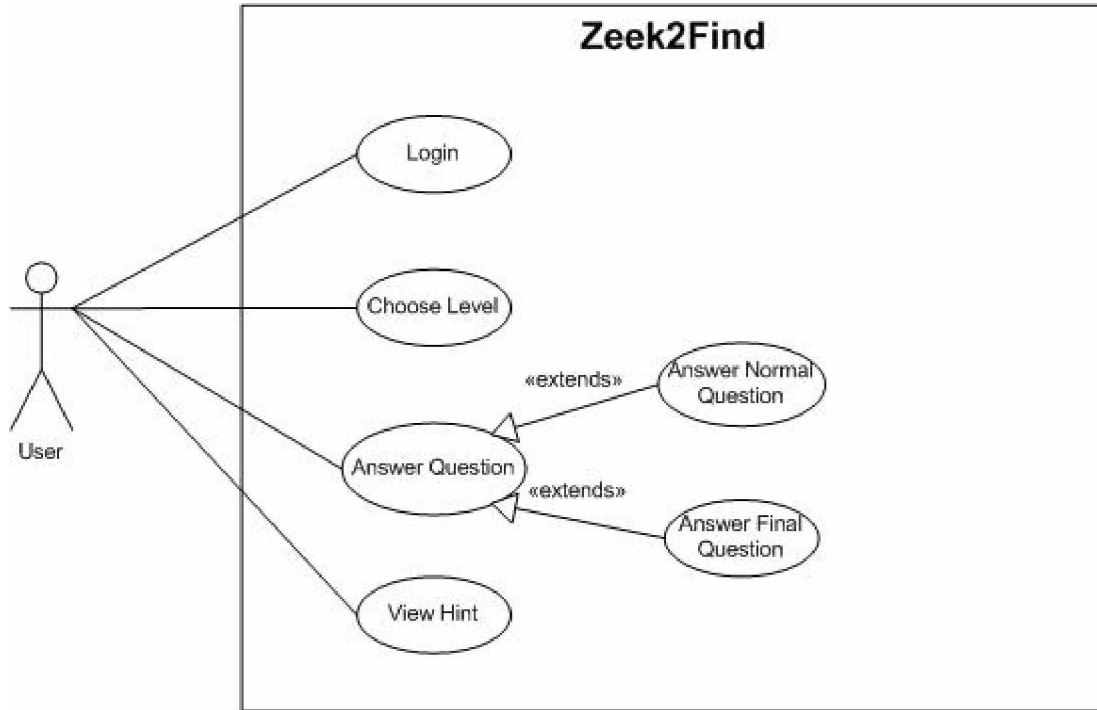
**Special Requirements:**

- Administrator usernames and passwords should require special permissions to access.

# Zeek2Find

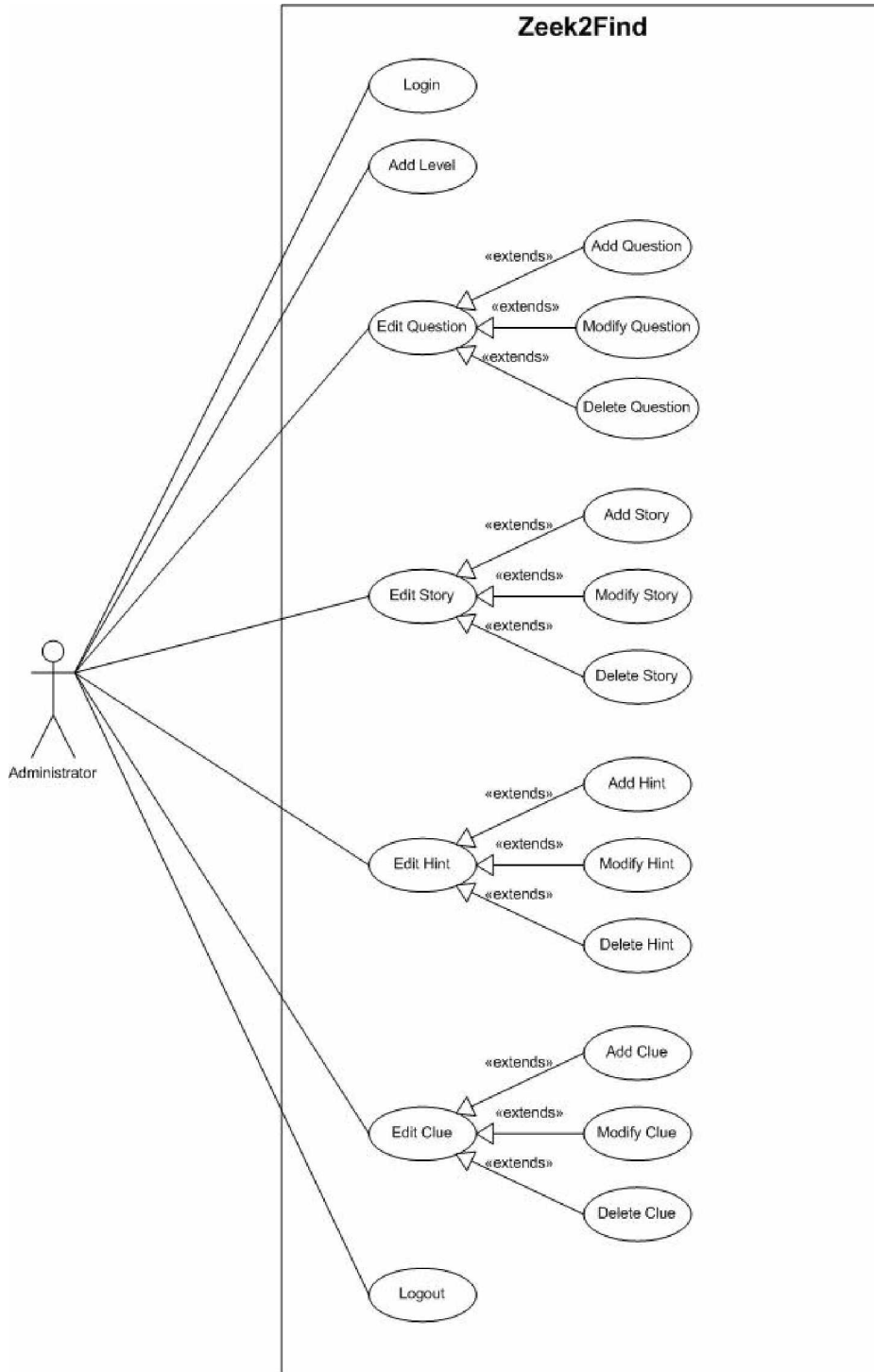
## 16.2 Use Case Diagram

### Use Case Diagram for User



# Zeek2Find

## Use Case Diagram for Administrator



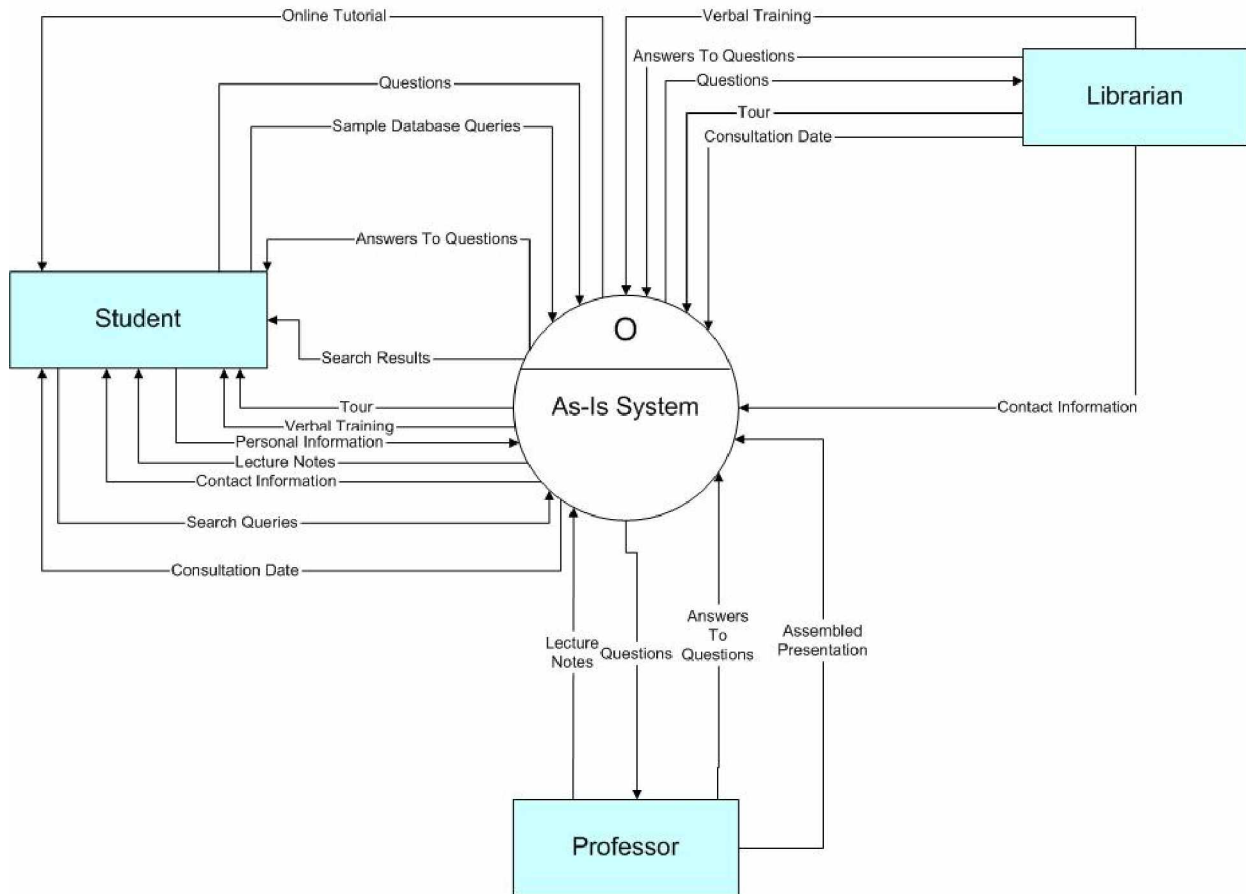
# Zeek2Find

## 17 REQUIREMENTS MAPPING

### 17.1 AS-IS DFD

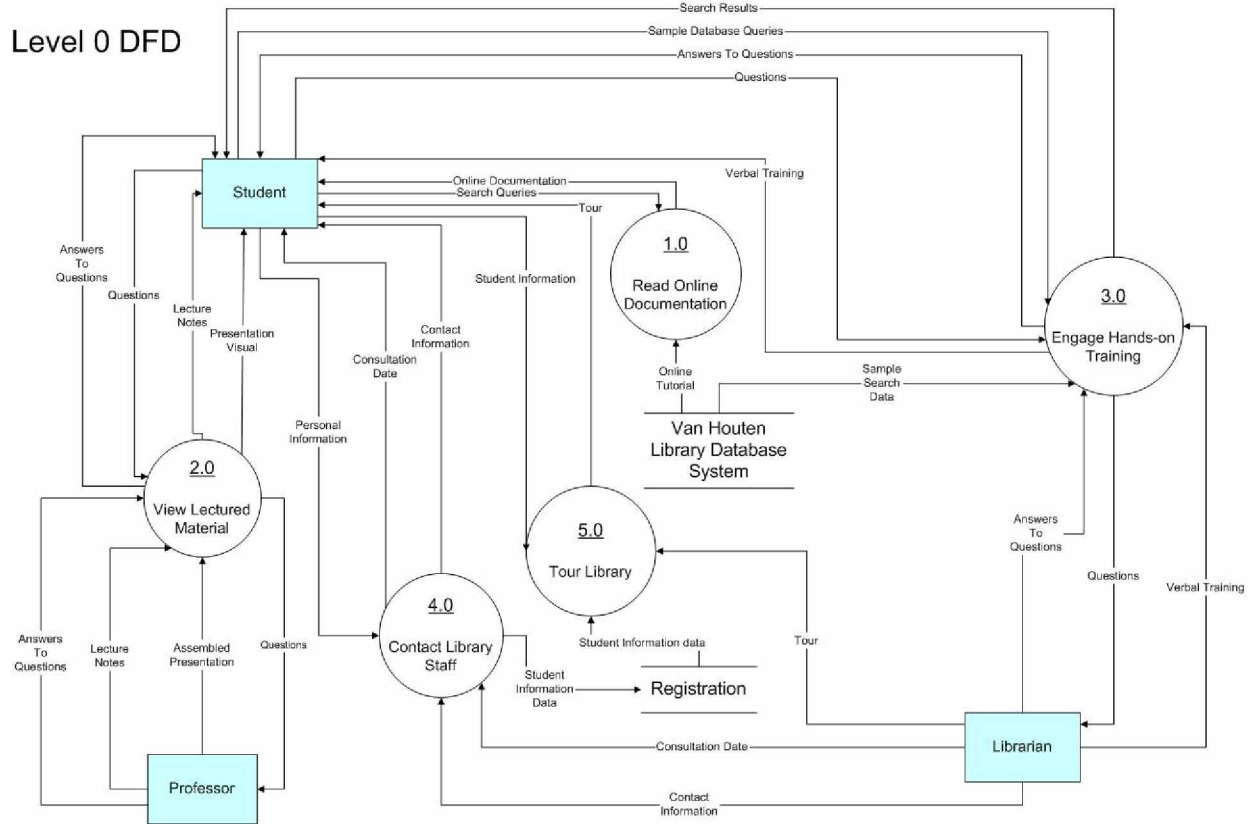
#### 17.1.1 Context Diagram

Context Level Diagram:  
As-Is System



# Zeek2Find

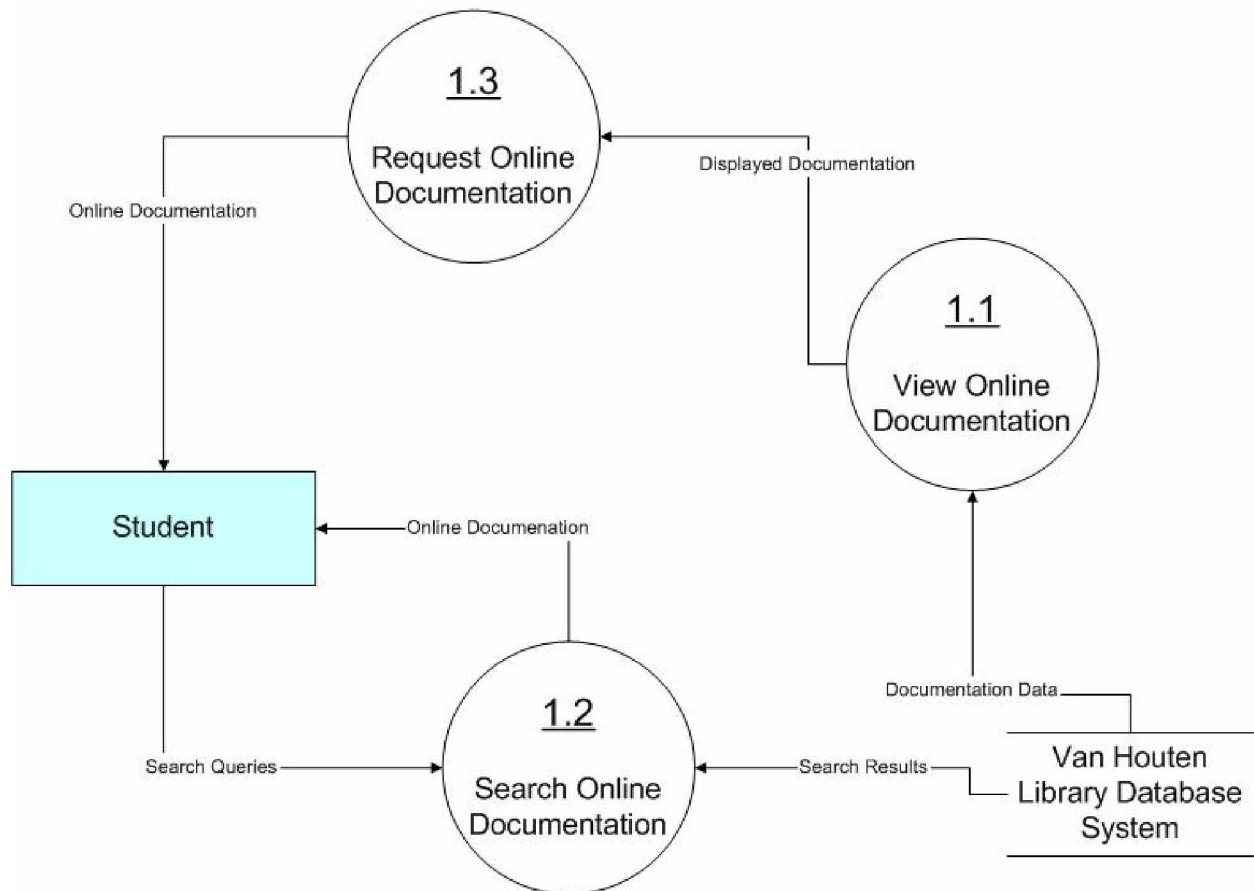
## 17.1.2 Level-0 Diagram



# Zeek2Find

## 17.1.3 Level-1 Decomposition: 1.0

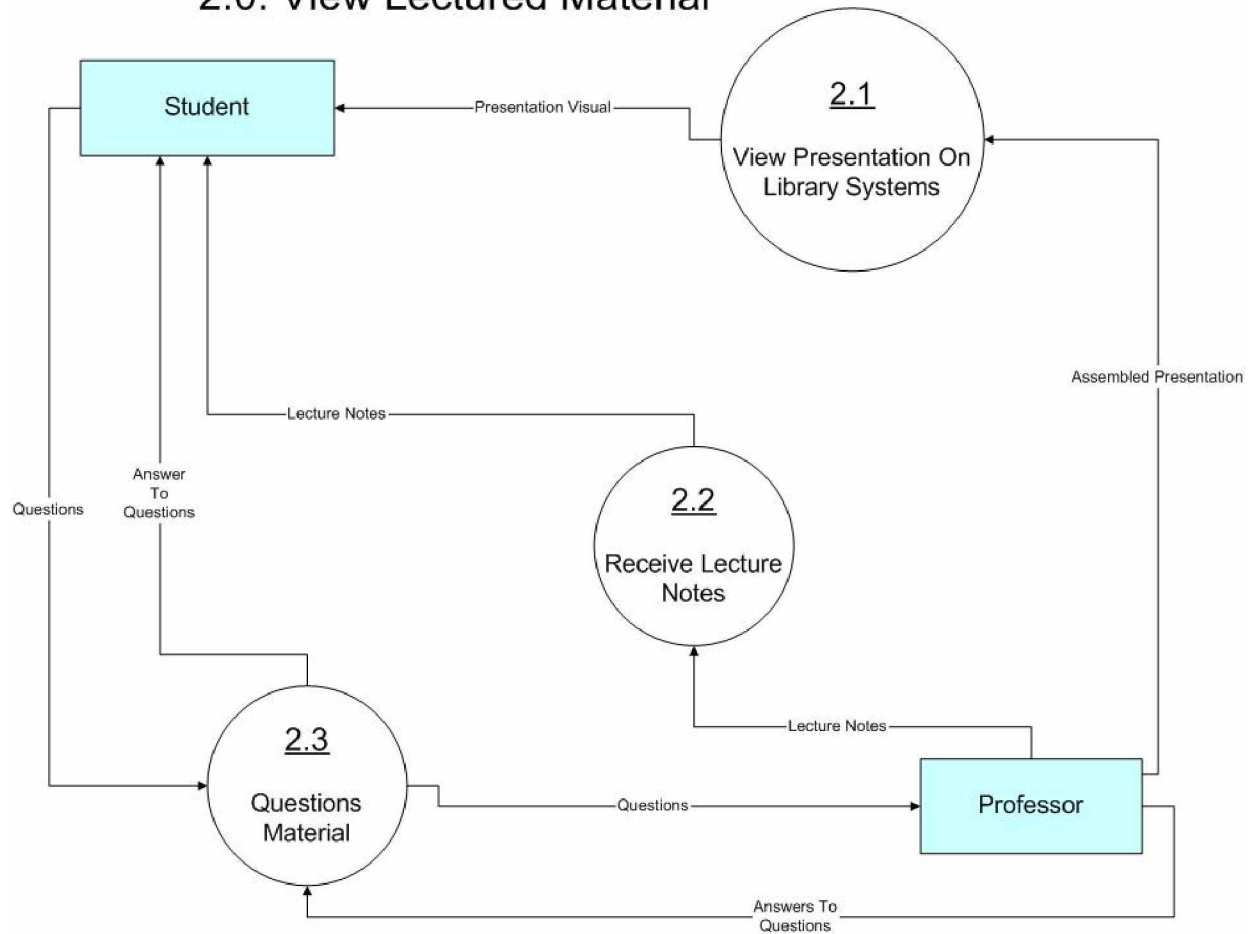
### 1.0: Read Online Documentation



# Zeek2Find

## 17.1.4 Level-1 Decomposition: 2.0

### 2.0: View Lectured Material

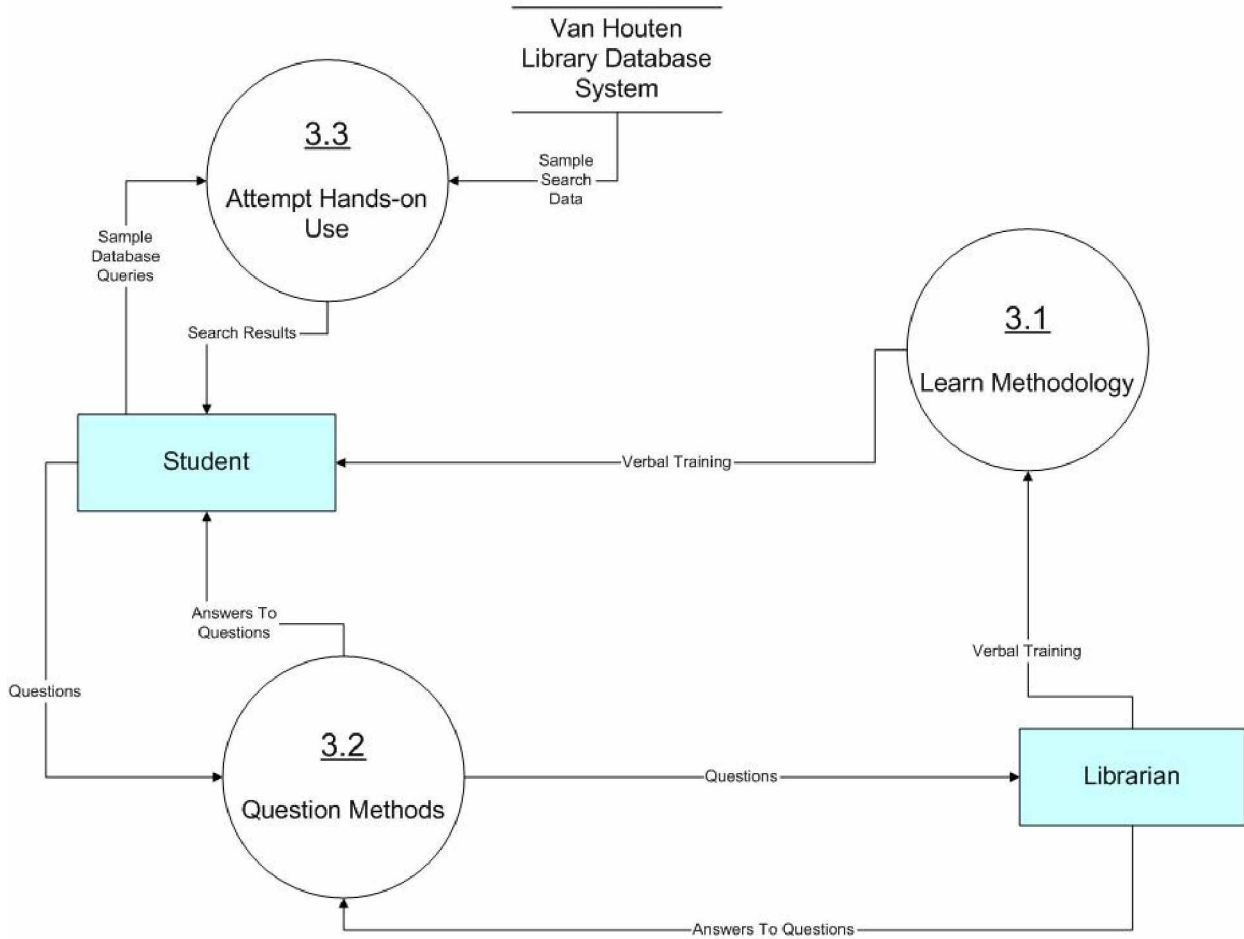




# Zeek2Find

## 17.1.5 Level-1 Decomposition: 3.0

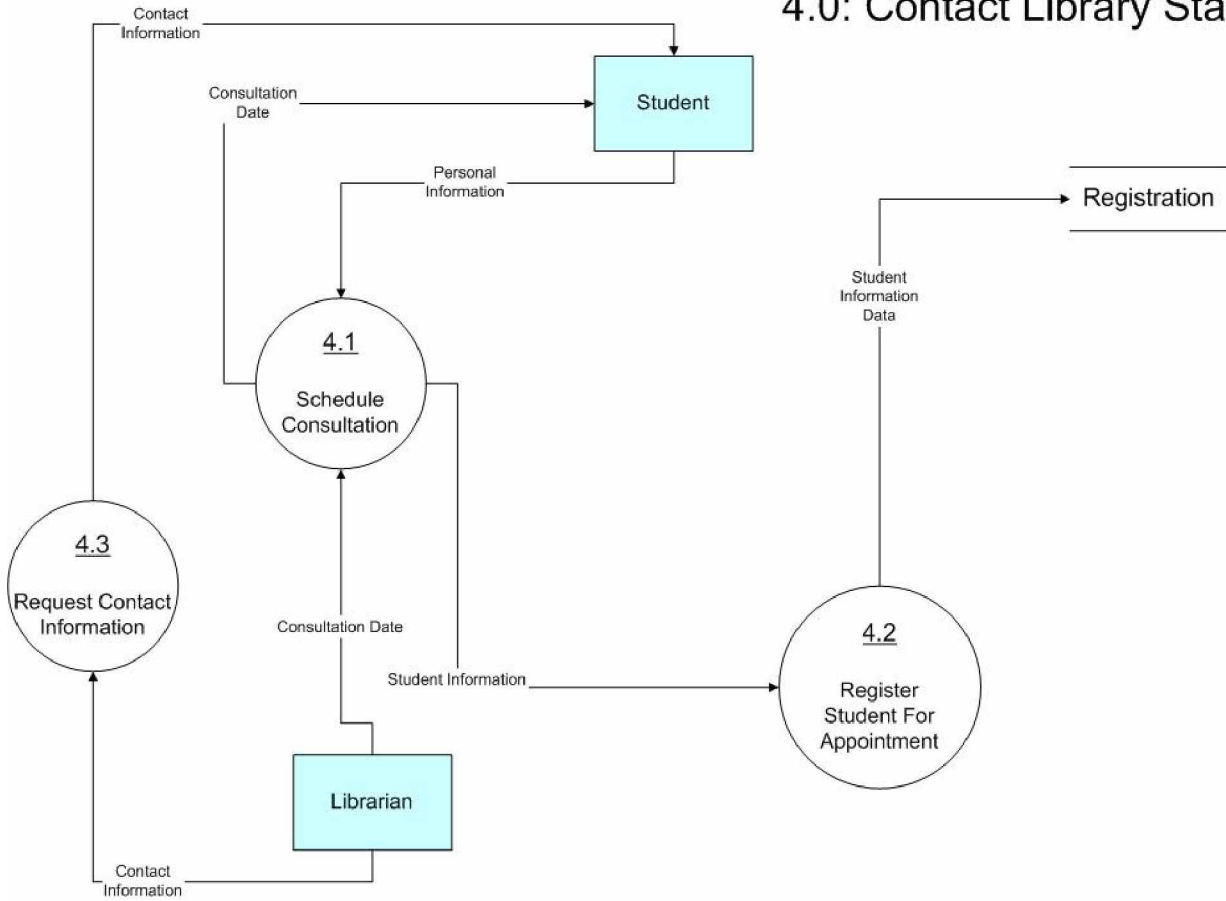
### 3.0: Engage Hands-On Training



# Zeek2Find

## 17.1.6 Level-1 Decomposition: 4.0

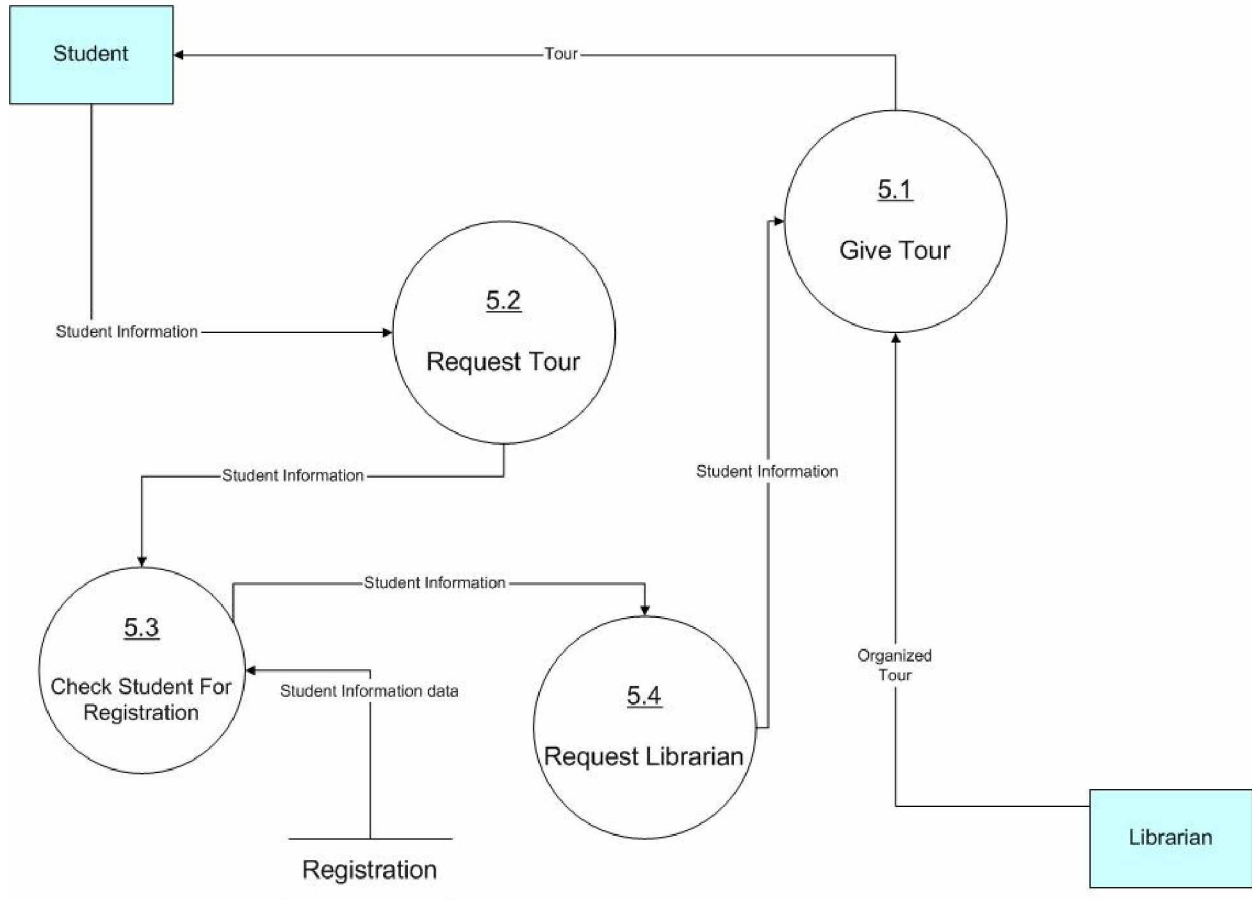
### 4.0: Contact Library Staff



# Zeek2Find

## 17.1.7 Level-1 Decomposition: 5.0

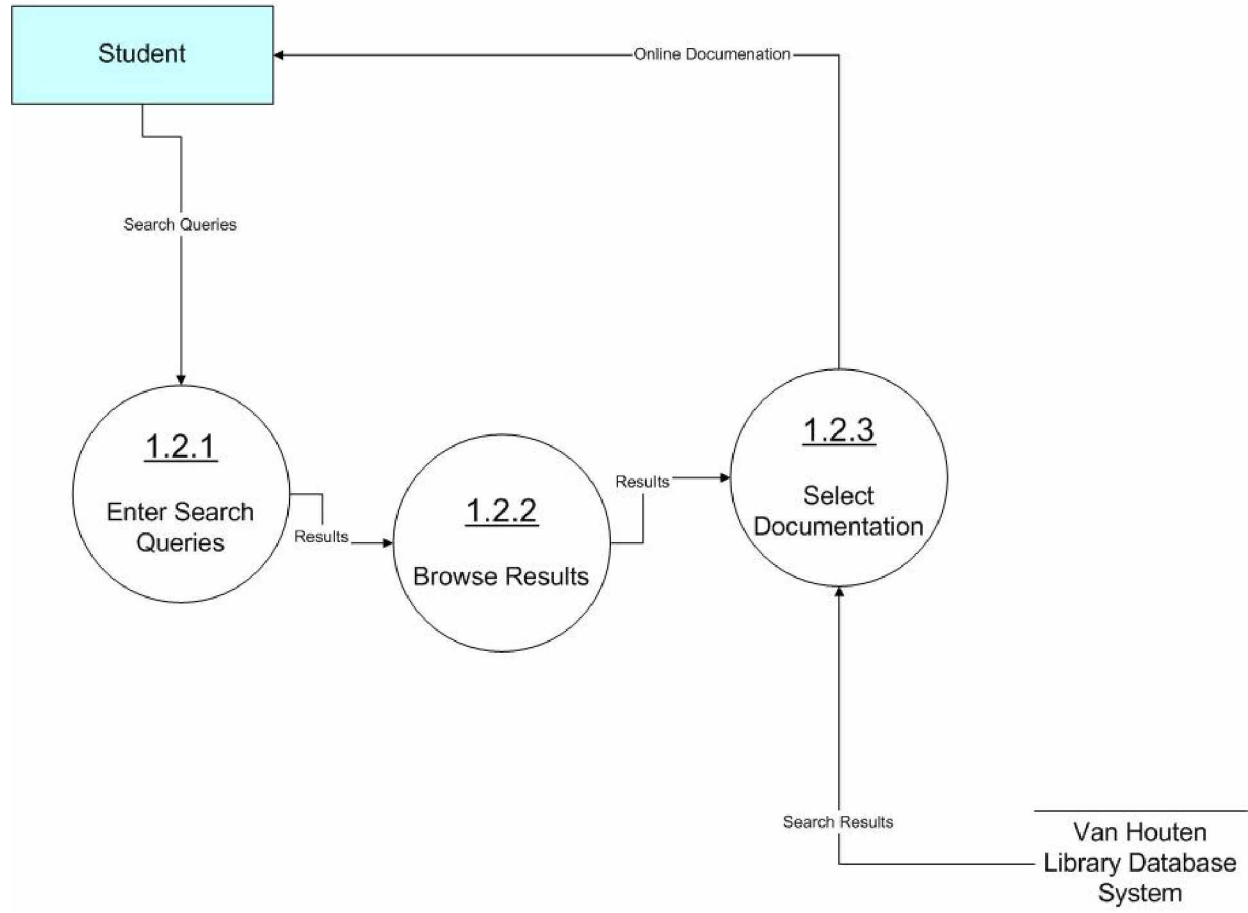
### 5.0: Tour Library



# Zeek2Find

## 17.1.8 Level-2 Decomposition: 1.2

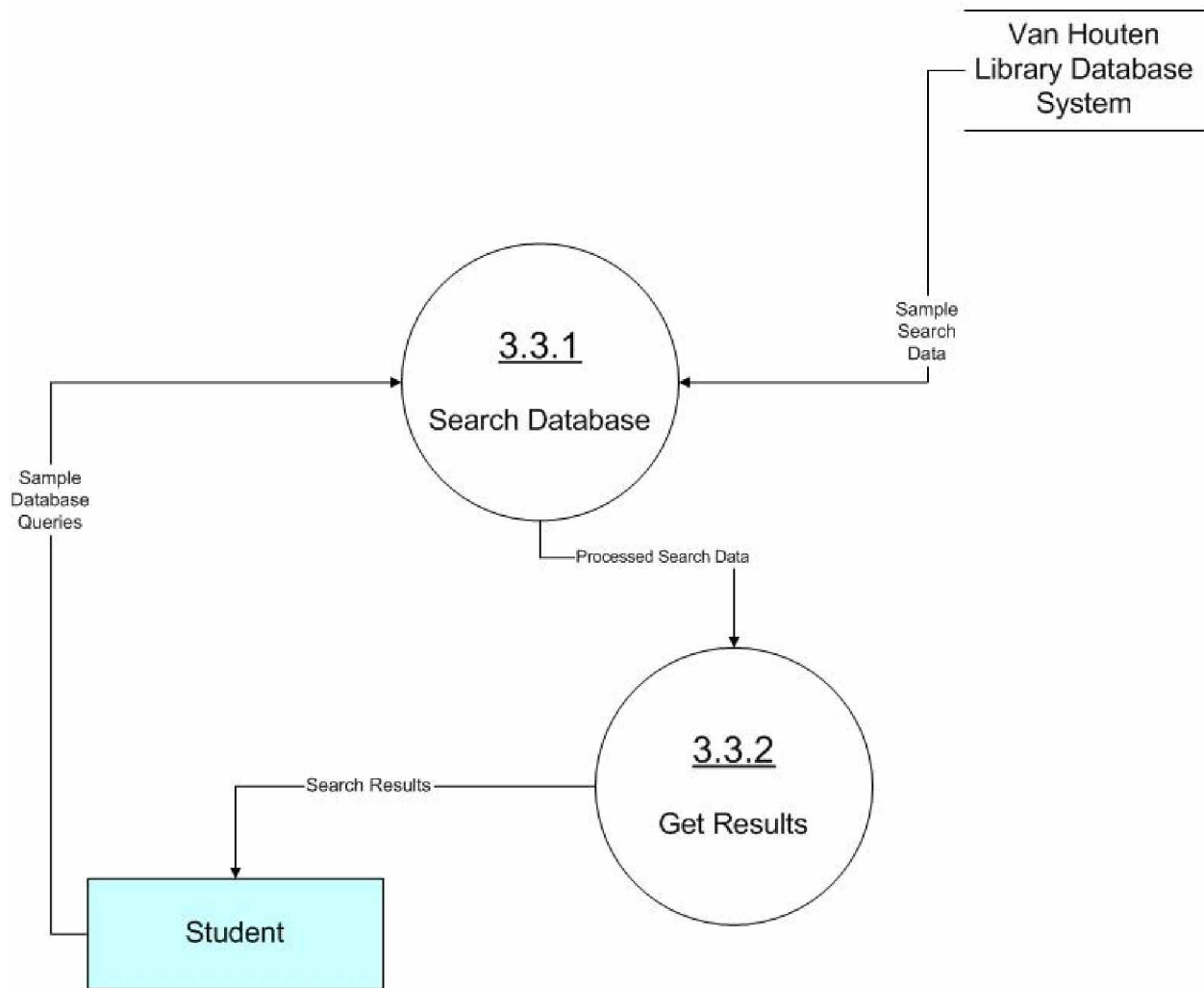
### 1.2: Search Online Documentation



# Zeek2Find

## 17.1.9 Level-2 Decomposition: 3.3

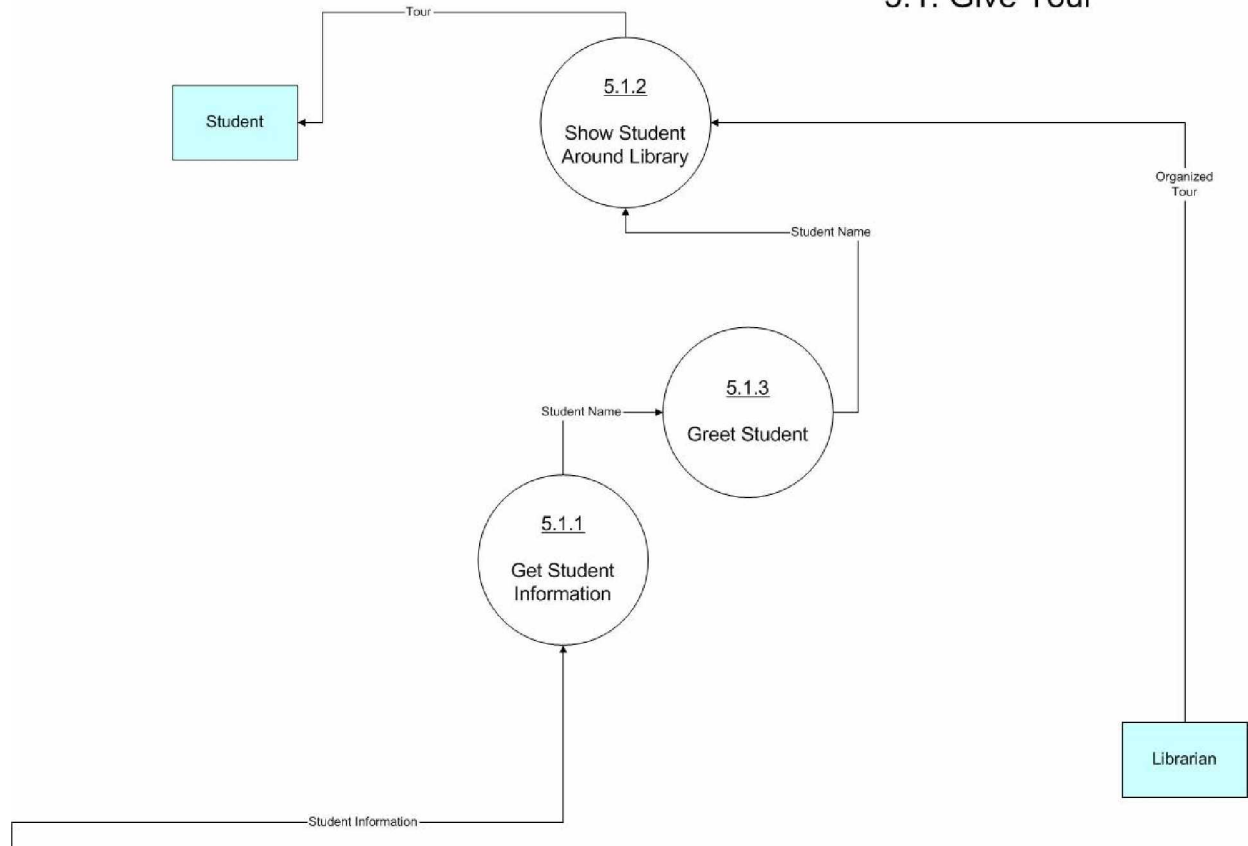
### 3.3: Attempt Hands-On Use



# Zeek2Find

## 17.1.10 Level-2 Decomposition: 5.1

### 5.1: Give Tour

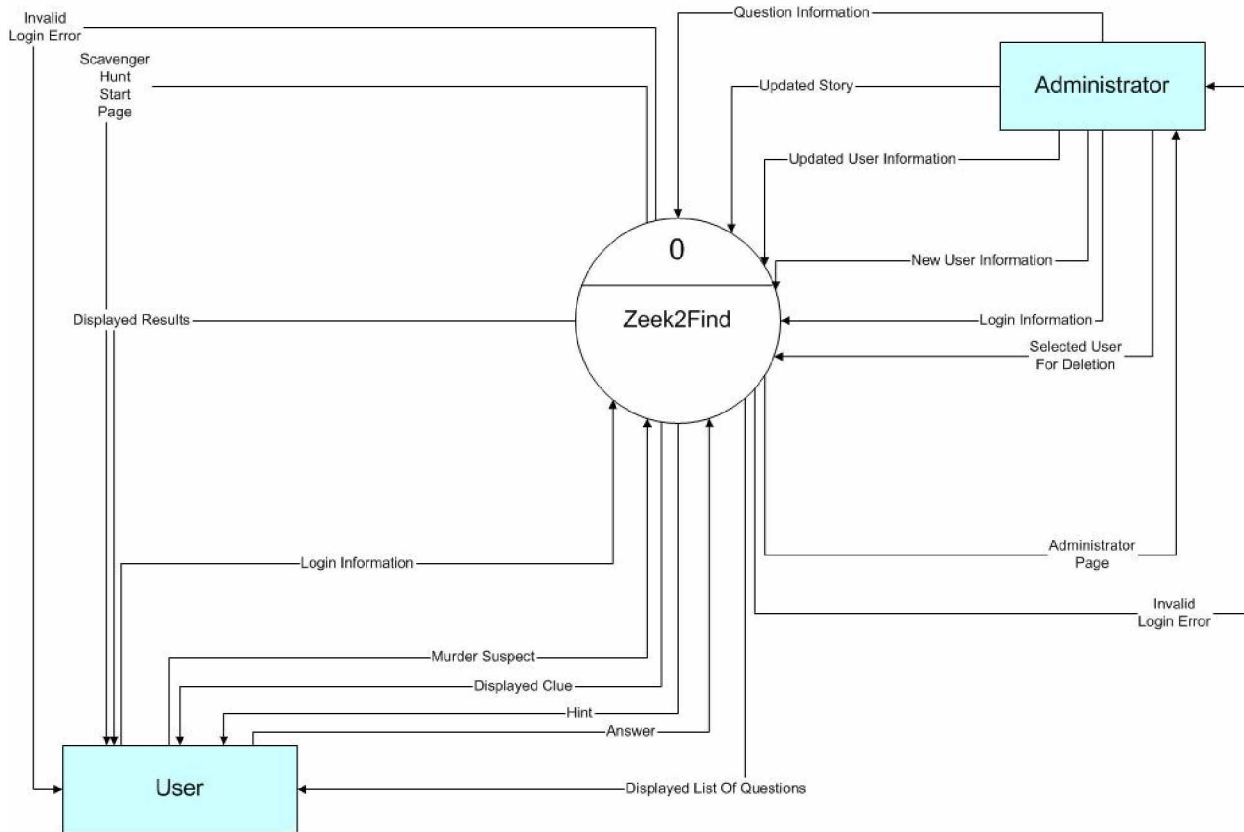


# Zeek2Find

## 17.2 TO-BE DFD

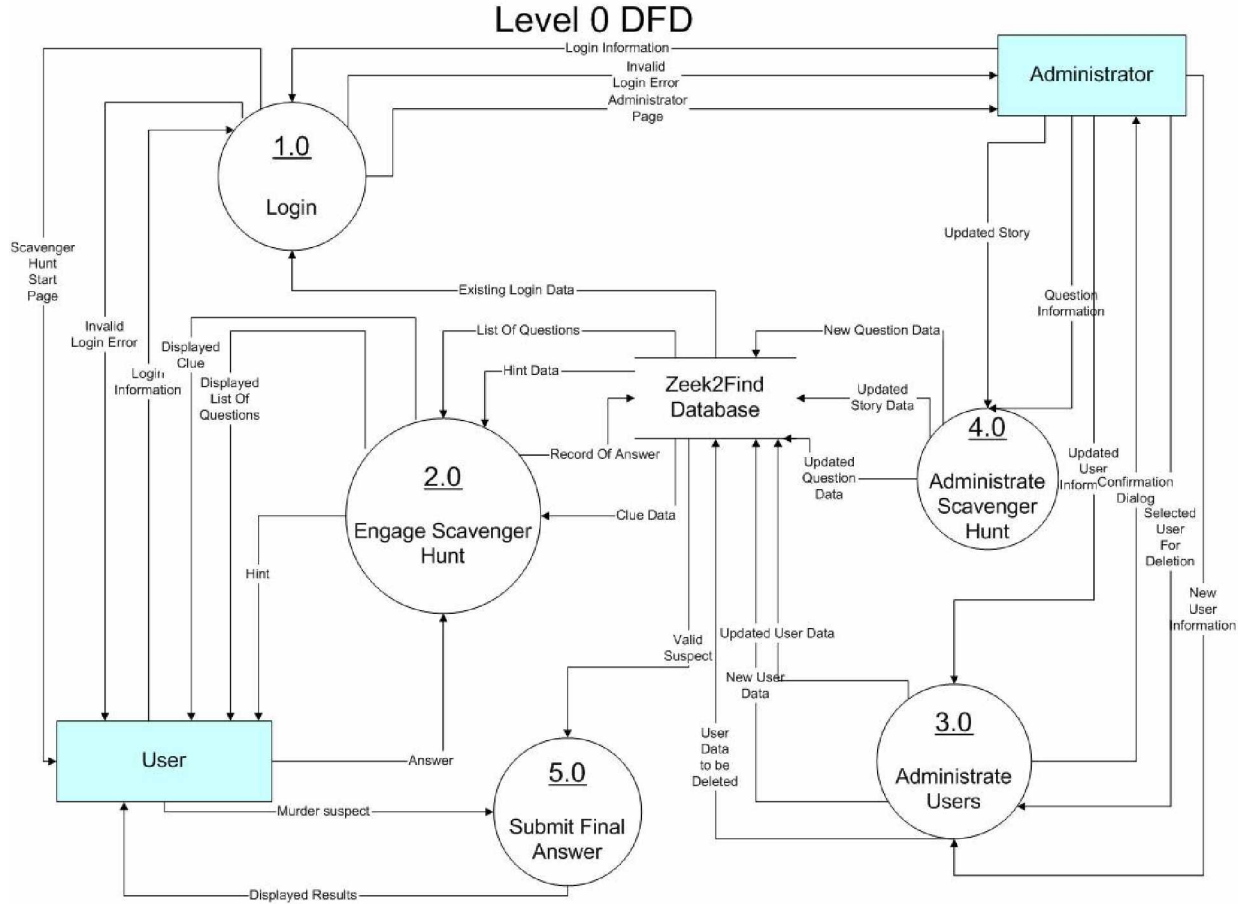
### 17.2.1 Context Diagram

Context Level Diagram:  
To Be System



# Zeek2Find

## 17.2.2 Level-0 Diagram

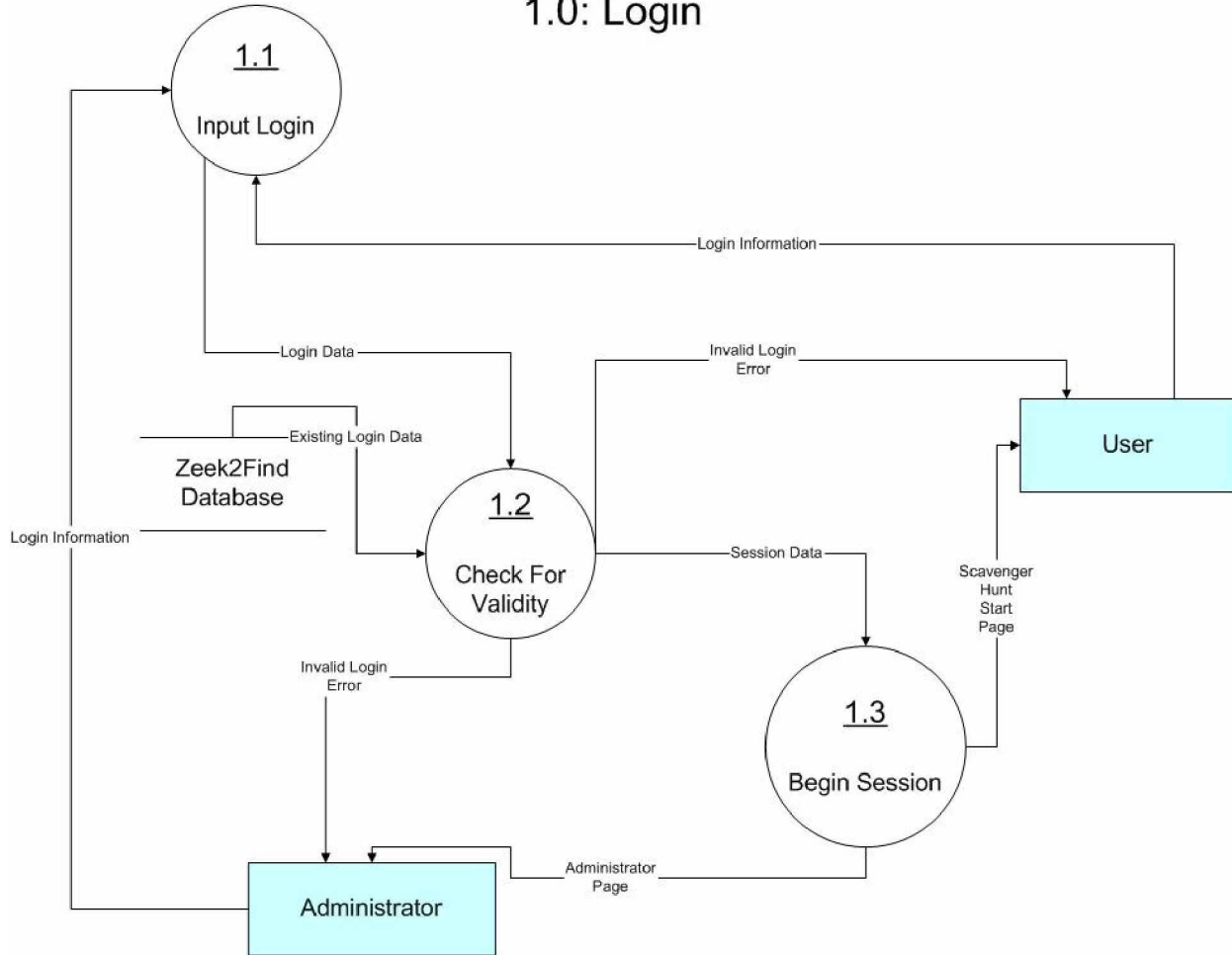




# Zeek2Find

## 17.2.3 Level-1 Decomposition: 1.0

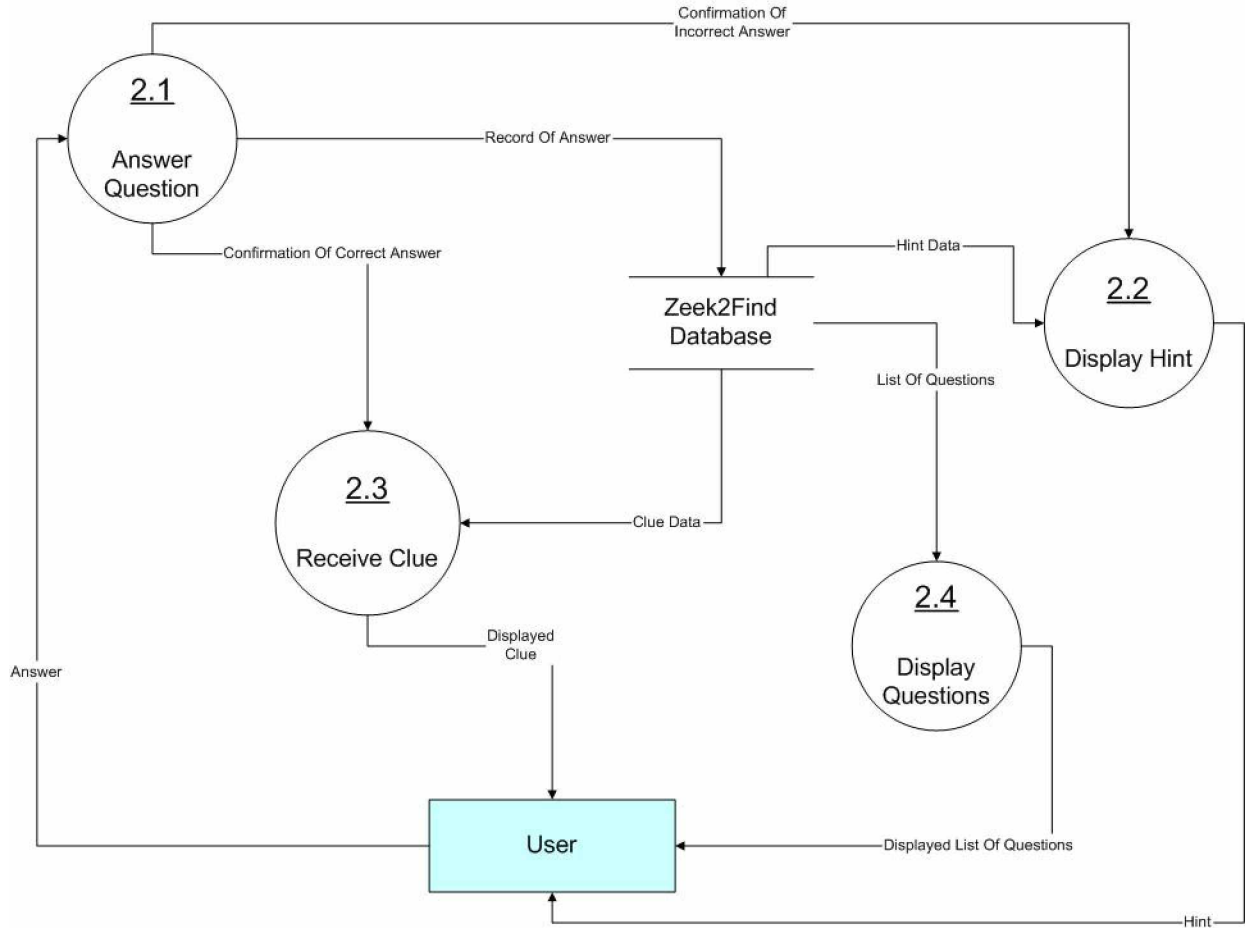
### 1.0: Login



# Zeek2Find

## 17.2.4 Level-1 Decomposition: 2.0

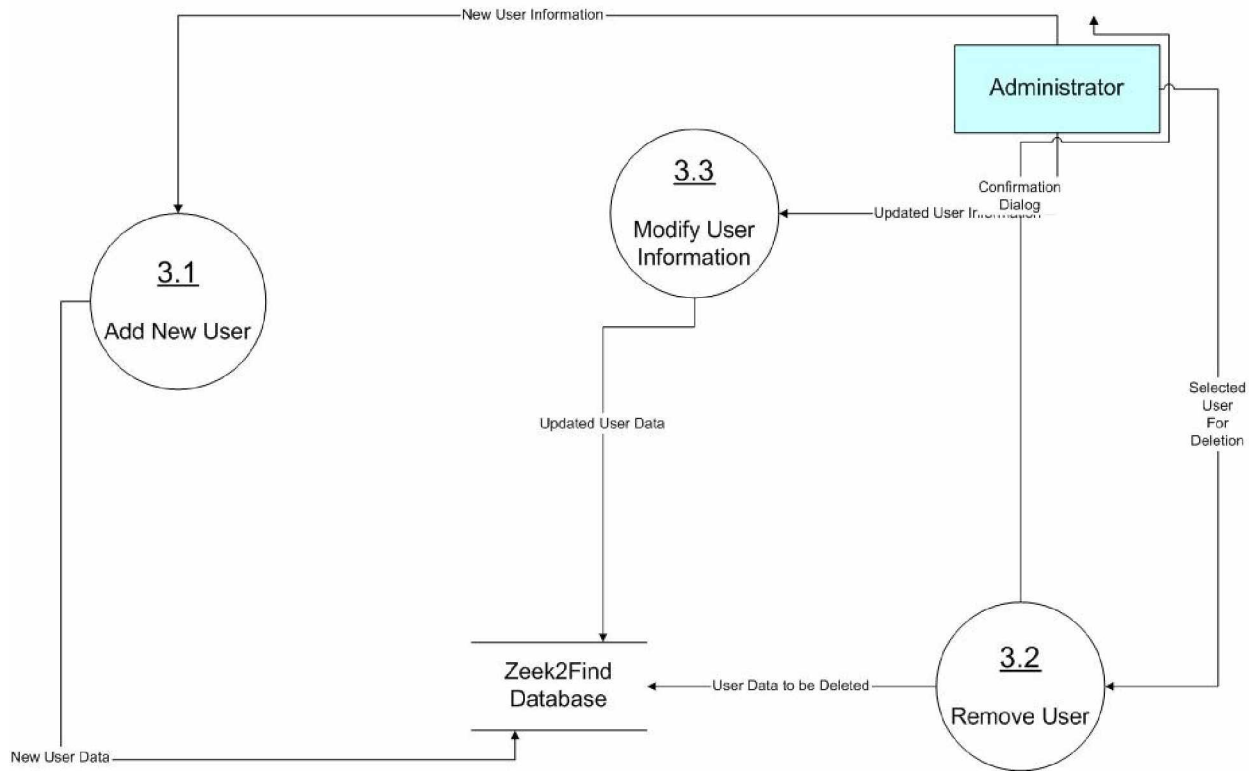
### 2.0: Engage Scavenger Hunt



# Zeek2Find

## 17.2.5 Level-1 Decomposition: 3.0

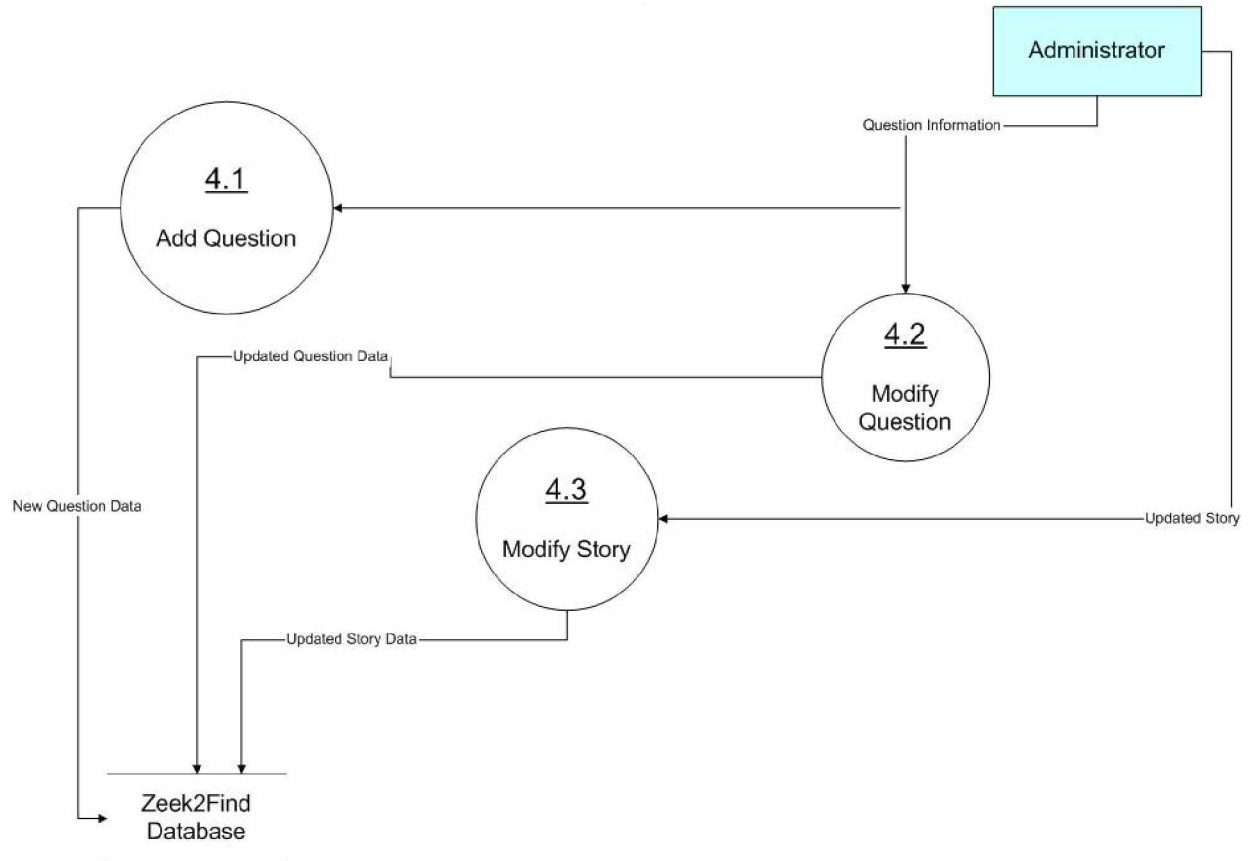
### 3.0: Adminstrate Users



# Zeek2Find

## 17.2.6 Level-1 Decomposition: 4.0

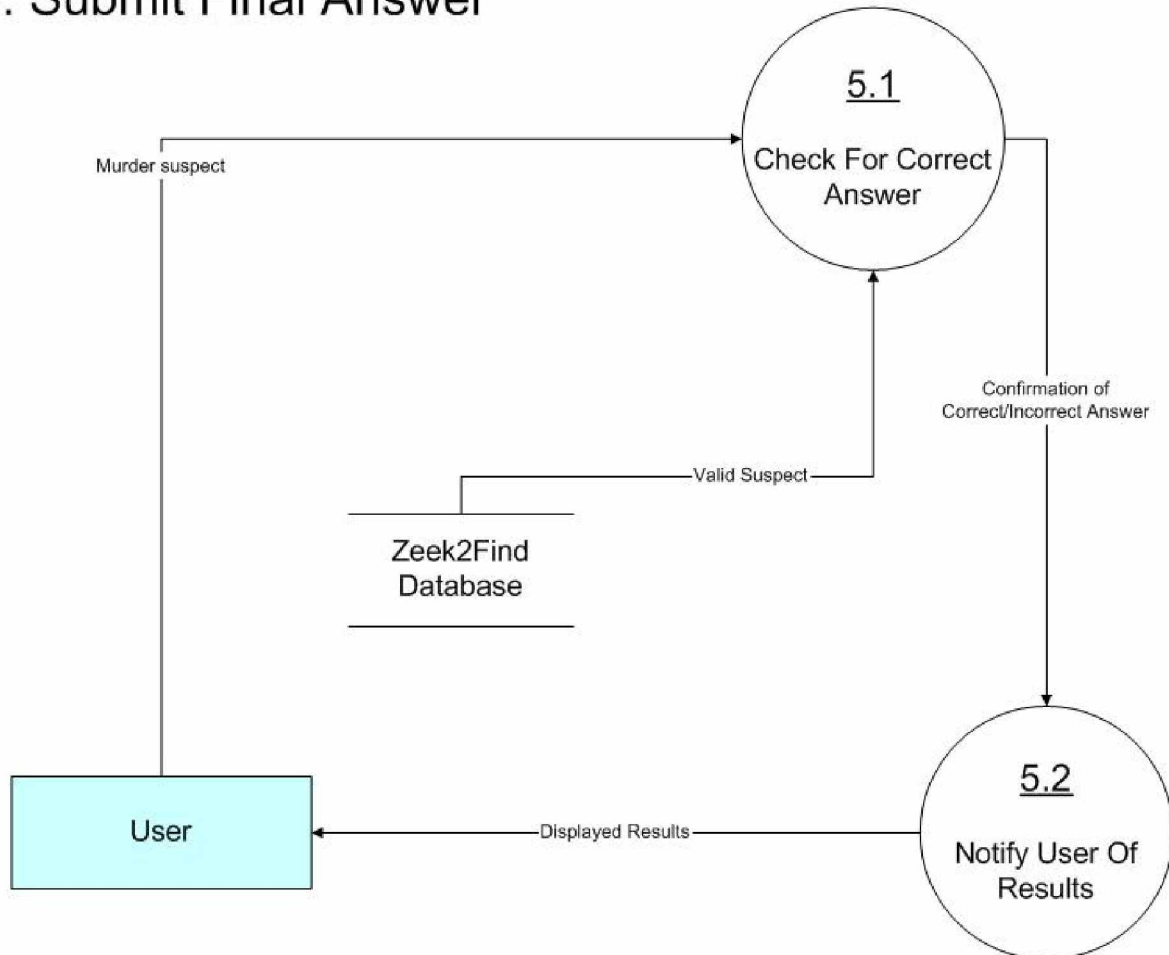
### 4.0: Adminstrate Scavenger Hunt



# Zeek2Find

## 17.2.7 Level-1 Decomposition: 5.0

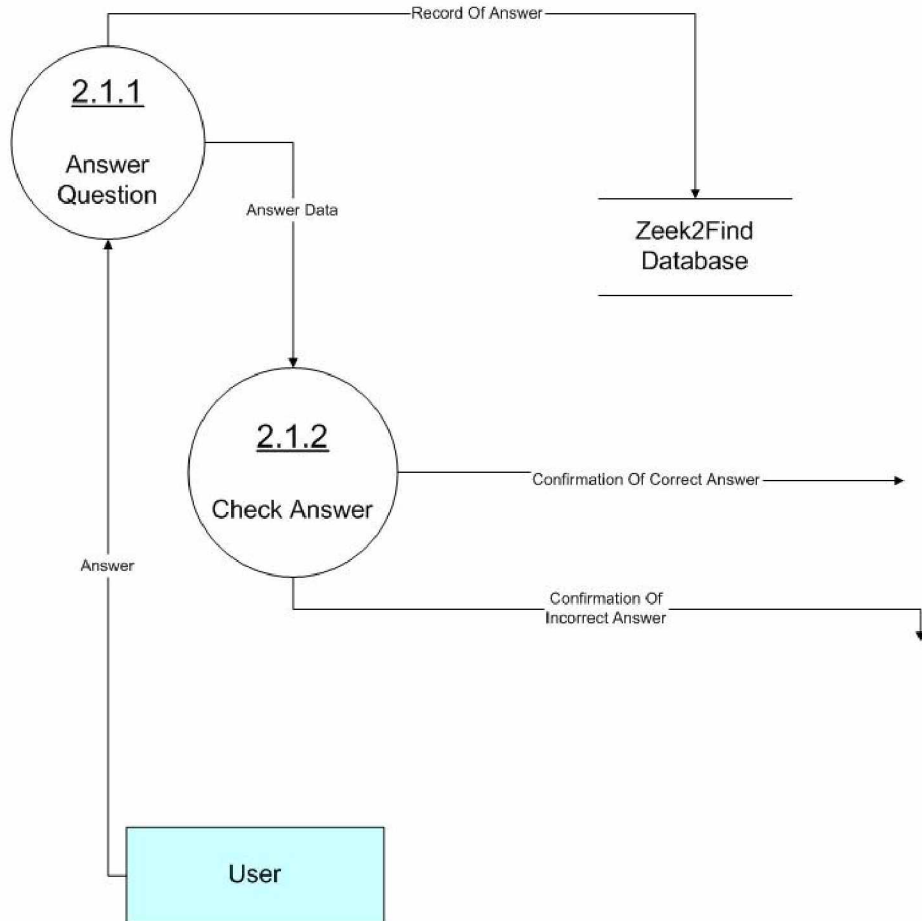
### 5.0: Submit Final Answer



# Zeek2Find

## 17.2.8 Level-2 Decomposition: 2.1

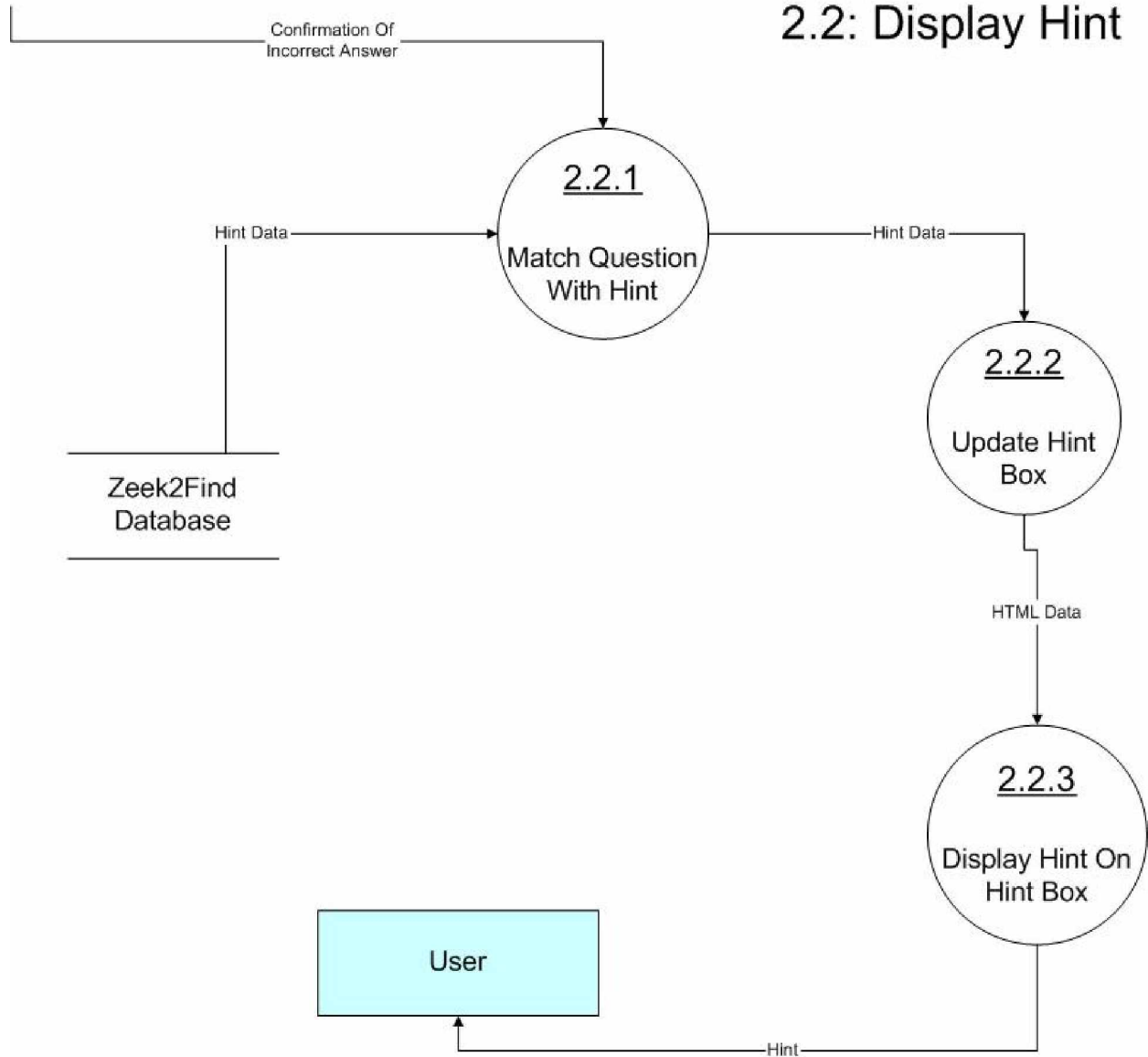
### 2.1: Answer Question



# Zeek2Find

## 17.2.9 Level-2 Decomposition: 2.2

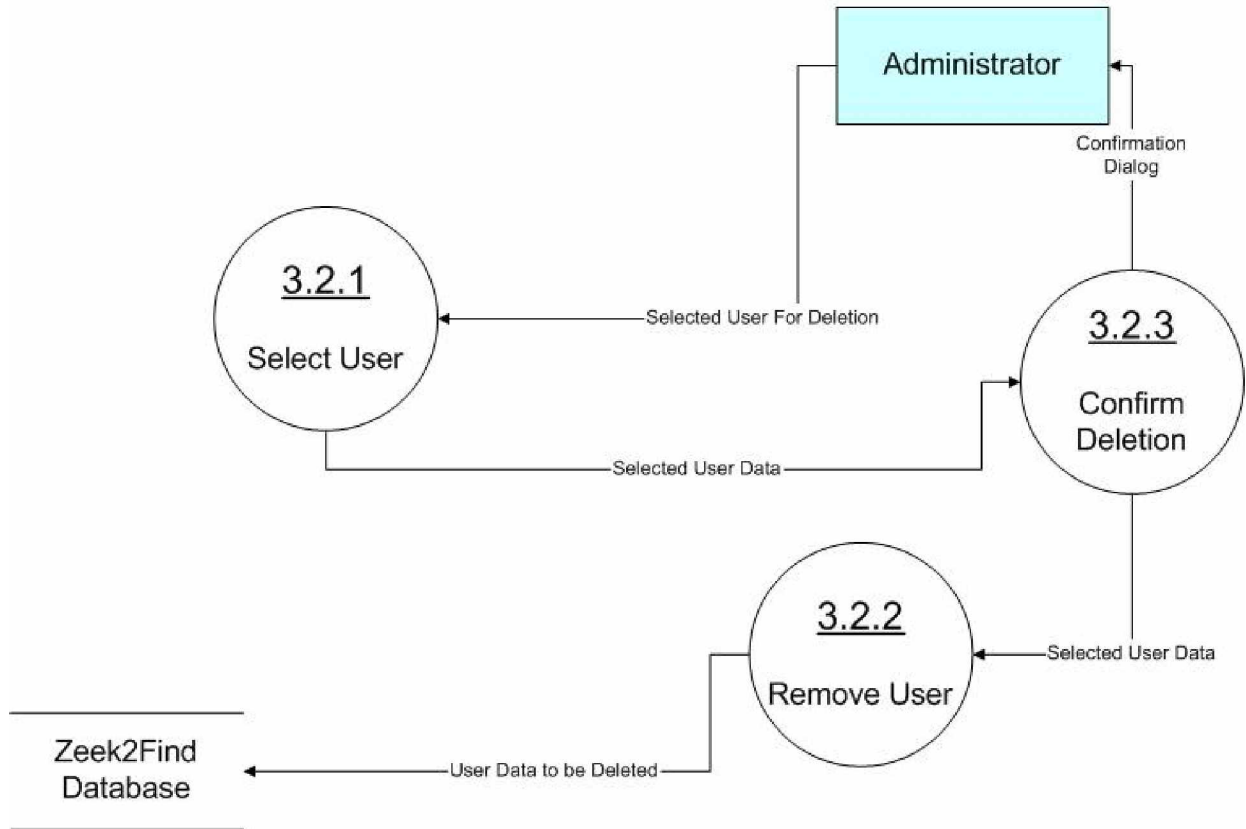
### 2.2: Display Hint



# Zeek2Find

## 17.2.10 Level-2 Decomposition: 3.2

### 3.2: Remove User

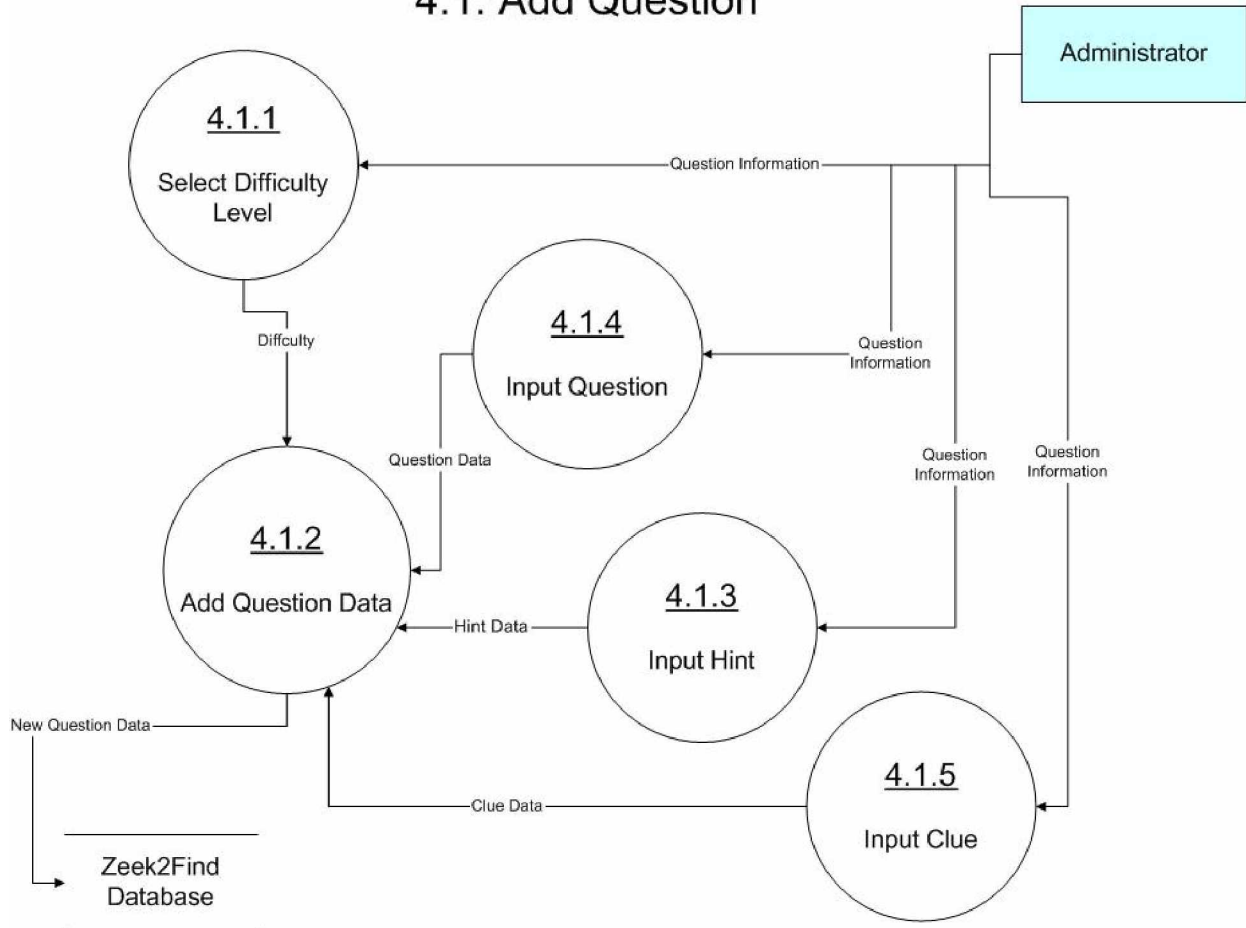




# Zeek2Find

## 17.2.11 Level-2 Decomposition: 4.1

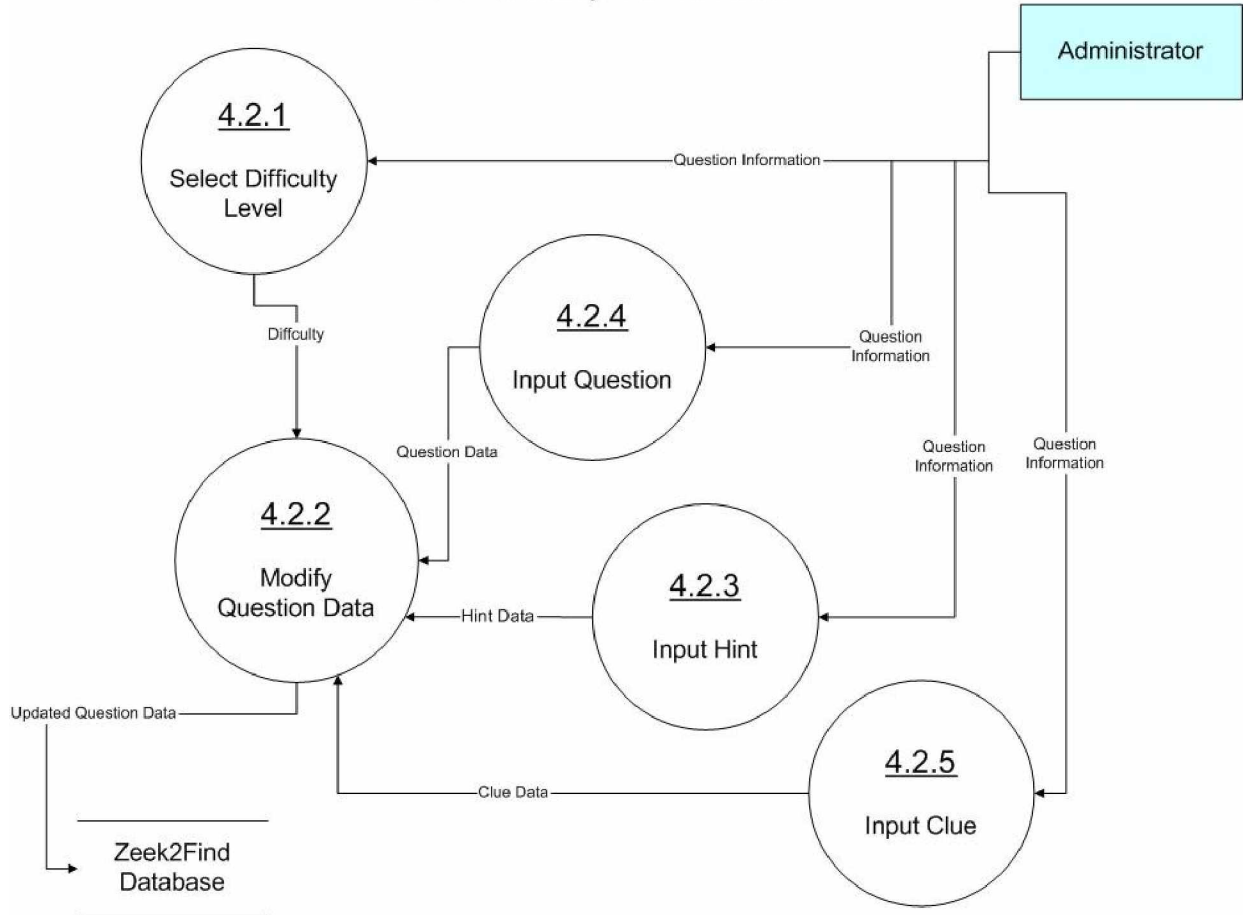
### 4.1: Add Question



# Zeek2Find

## 17.2.12 Level-2 Decomposition: 4.2

### 4.2: Modify Question



# Zeek2Find

## 18 PROCESS SPECIFICATIONS

### 18.1 Structured English

Process 1.0: Login

DO

    READ user's e-mail address

    IF e-mail address is formatted properly

        Engage Scavenger Hunt

UNTIL e-mail verified

Engage Scavenger Hunt

Process 2.0: Engage Scavenger Hunt

DO

    IF level is not chosen

        GET Level

    IF level is chosen

        get a story and clues from database

        get 10 questions from database

        grade any newly answered questions

    DO

        PRINT question

        IF question is answered correctly

            give the user a clue

        IF question is answered once incorrectly

            give the user a hint

        IF question is answered twice incorrectly

            give the user a walkthrough movie

    UNTIL no more questions

UNTIL all questions are correct

allow user to Submit Final Answer

Process 3.0: Administrate Users

DO

    show list of first 50 users by completion date

    for each user show time to completion

    for each level show average time to completion

# Zeek2Find

UNTIL logout

Process 4.0: Adminstrate Scavenger Hunt

DO

    get questions/hints/clues/levels/answers from database

    show questions/hints/clues/levels/answers

    for each question show percentage of users who used a hint for the question

    for each question show percentage of users who used a video for the question

    for each story show average guesses before correct murder is identified

    IF a change has been made

        IF Question is selected

            update question for that record

        IF Hint is selected

            update hint for that record

        IF Answer is selected

            update answer for that record

        IF Difficulty is selected

            update difficulty for that record

UNTIL logout

Process 5.0: Submit Final Answer

DO

    get murderer from database

    compare to suspected murderer

UNTIL murder suspect is correct

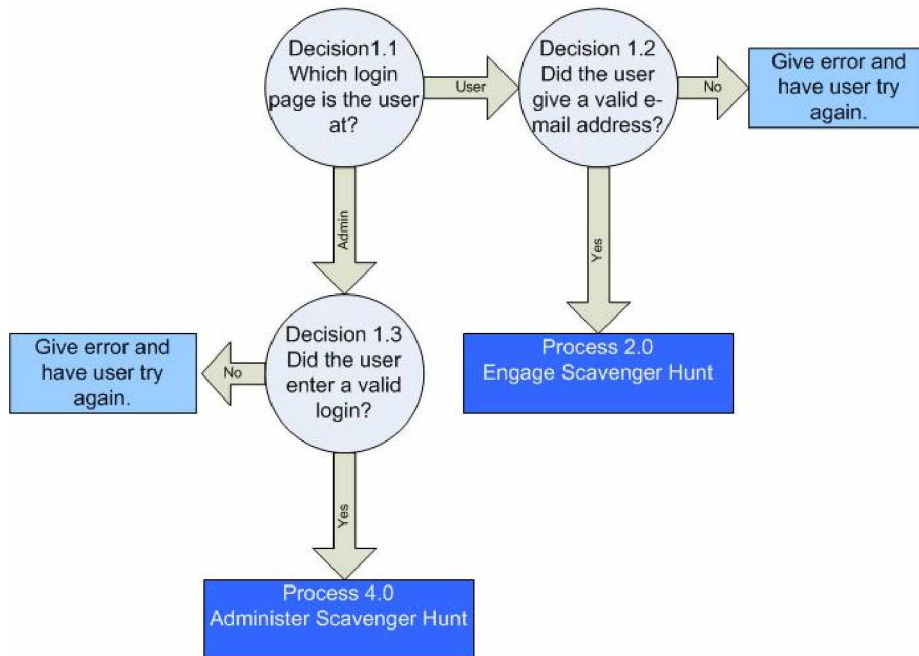
    user has won the game

# Zeek2Find

## 18.2 Decision Tree

### 18.2.1 Login

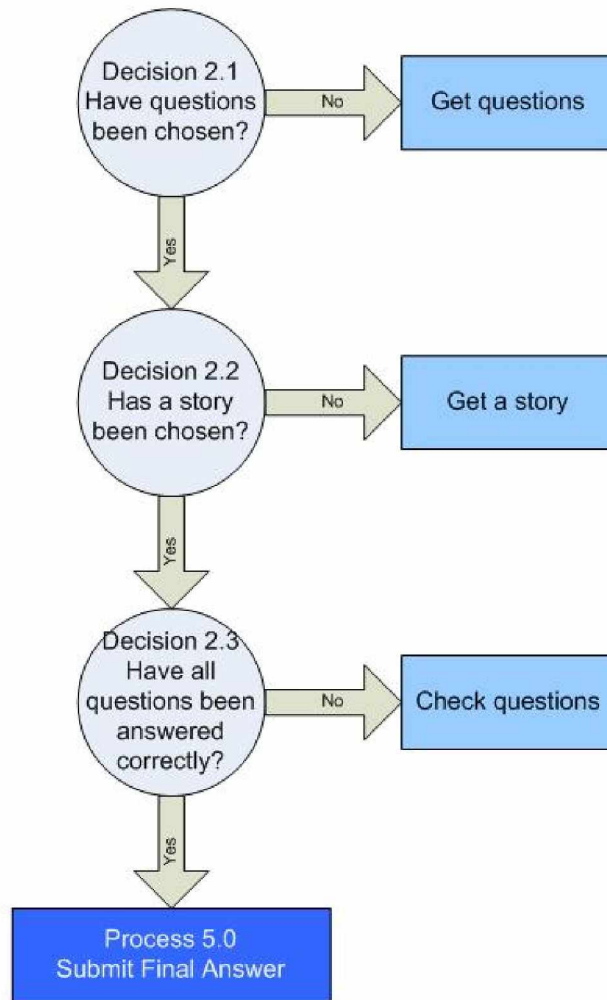
Decision Tree for Process 1.0:  
Login



# Zeek2Find

## 18.2.2 Engage Scavenger Hunt

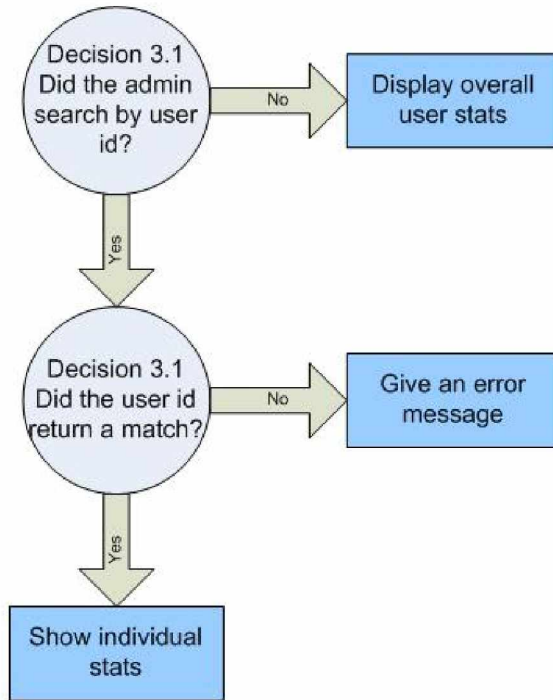
### Decision Tree for Process 2.0: Engage Scavenger Hunt



# Zeek2Find

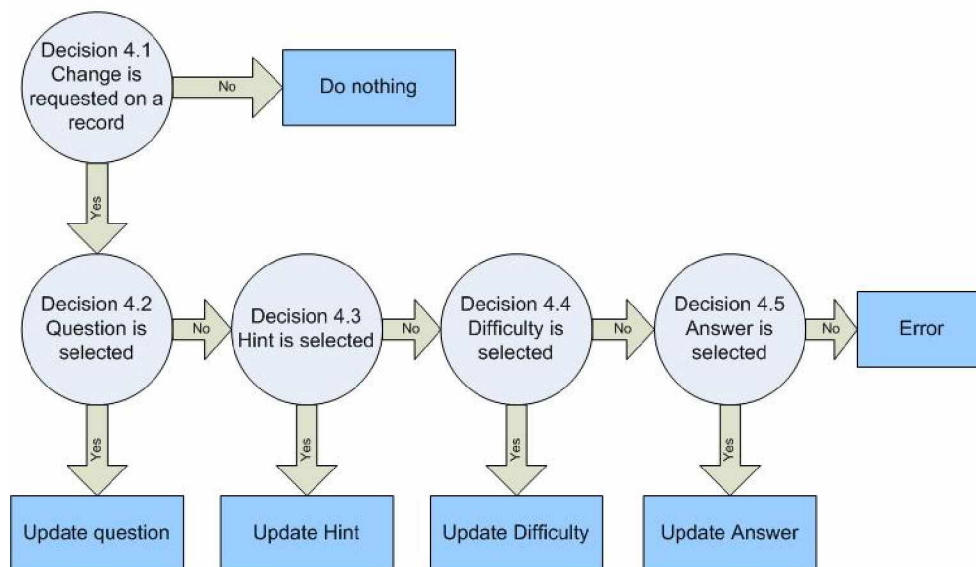
## 18.2.3 Administrate Users

### Decision Tree for Process 3.0: Administrate Users



## 18.2.4 Administrative Scavenger Hunt

### Decision Tree for Process 3.0: Administrative Scavenger Hunt

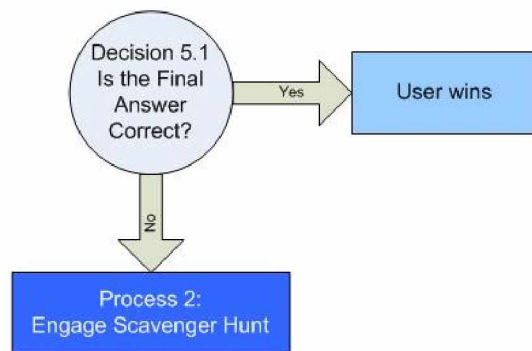




# Zeek2Find

## 18.2.5 Submit Final Answer

### Decision Tree for Process 5.0: Submit Final Answer



# Zeek2Find

## Decision Table

### 18.2.6 Process 1.0 - Login

Process 1.0  
 Login:

admin login page	Y	Y	N	N
valid login	Y	N	Y	N
enter game			X	
enter admin page	X			
give error message		X		X

### 18.2.7 Process 2.0 – Engage Scavenger Hunt

Process 2.0 Engage Scavenger Hunt:

questions are chosen	Y	Y	N	Y
story is chosen	Y	N	N	Y
all correct	Y	N	N	N
allow final answer	X			
get questions		X		
get a story			X	
check questions				X

### 18.2.8 Process 3.0 – Adminstrate Users

Process 3.0 Adminstrate Users:

did the admin search by userid?	Y	Y	N
individual found	Y	N	N
show individual stats	X		
give error message		X	
show overall stats			X

### 18.2.9 Process 4.0 – Adminstrate Scavenger Hunt

Process 4.0 Adminstrate Scavenger Hunt:

change is requested on a record	Y	Y	Y	Y	N	N	N	N
question is selected	Y	N	N	N	Y	N	N	N
hint is selected	N	Y	N	N	N	Y	N	N

# Zeek2Find

difficulty is selected	N	N	Y	N	N	N	Y	N
answer is selected	N	N	N	Y	N	N	N	Y
update question for record	X							
update hint for record		X						
update difficulty for record			X					
update answer for record				X				
do nothing					X	X	X	X

## 18.2.10 Process 5.0 – Submit Final Answer

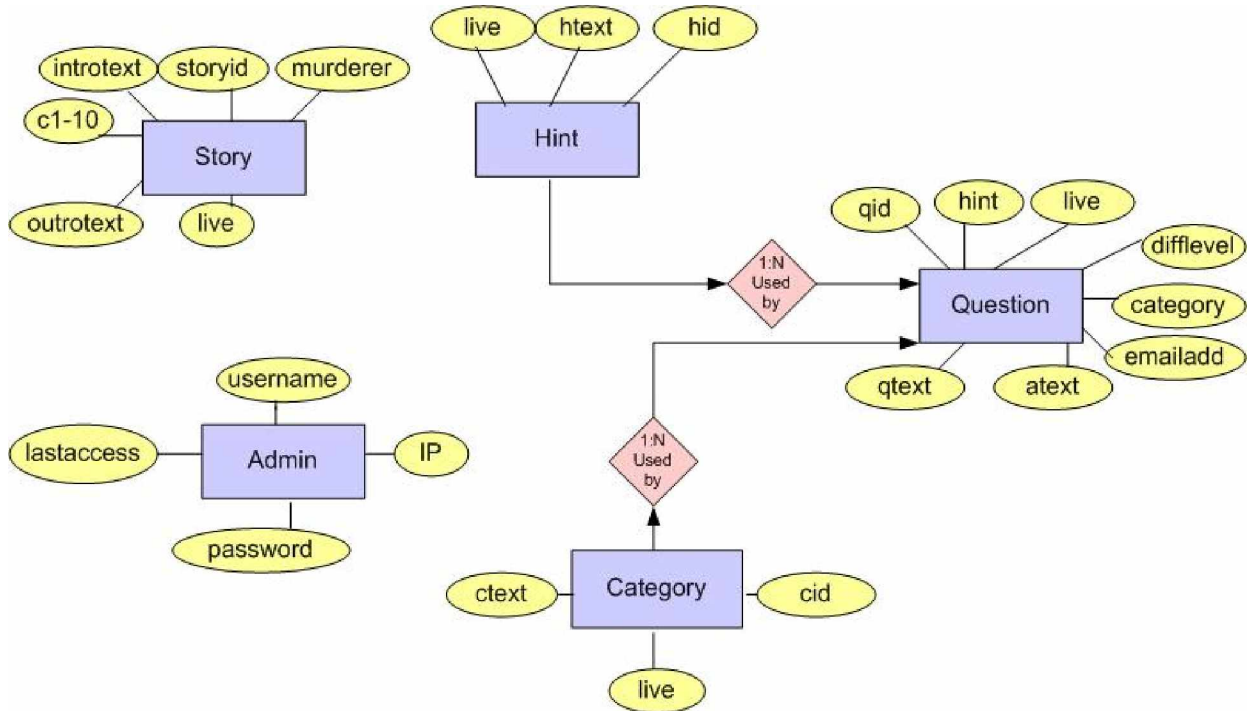
Process 5.0 Submit Final Answer:

murder suspect correct	Y	N
user wins	X	

# Zeek2Find

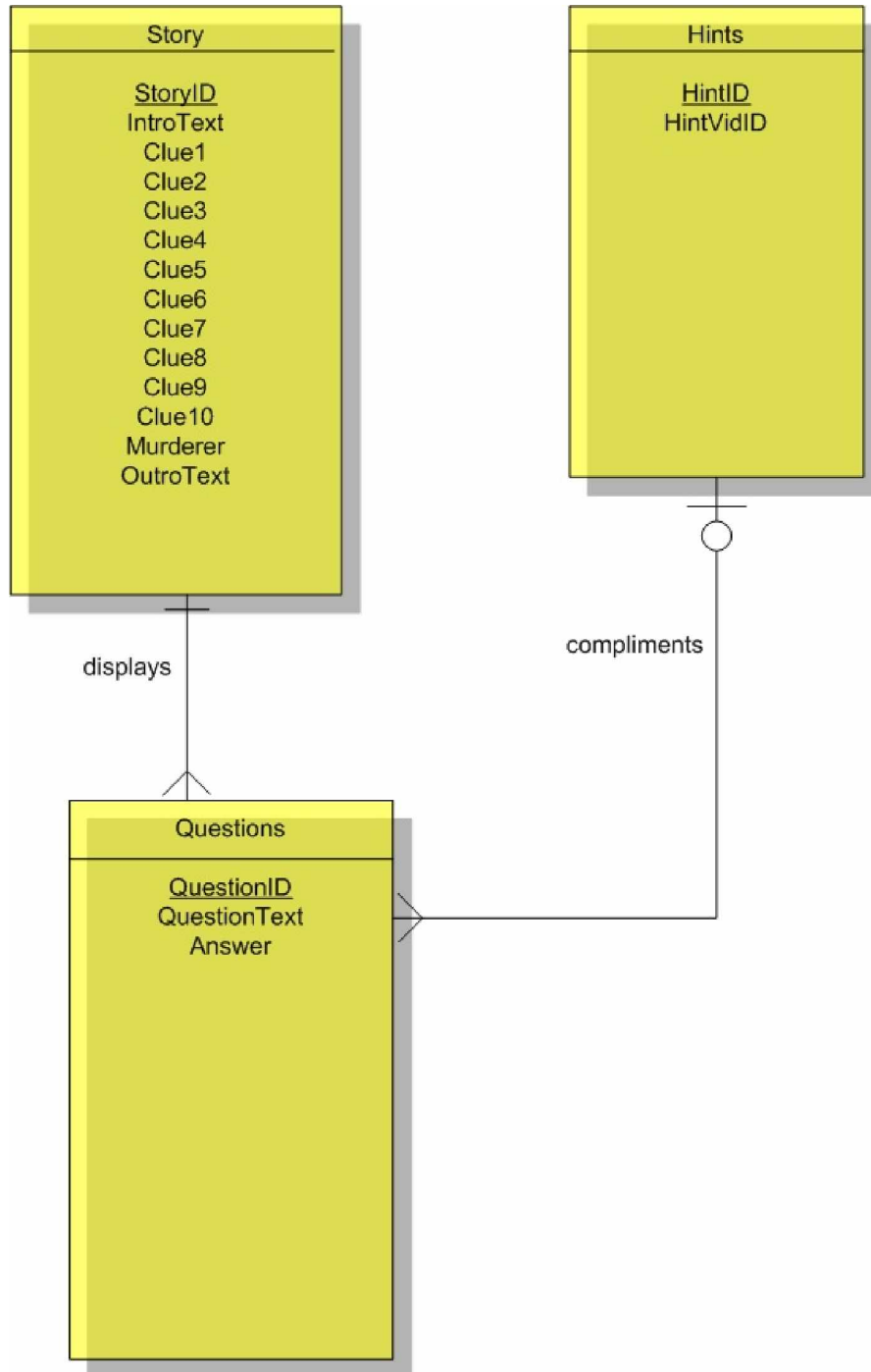
## 19 ENTITY RELATIONSHIP

### 19.1 Entity Relationship Model (ERM)



# Zeek2Find

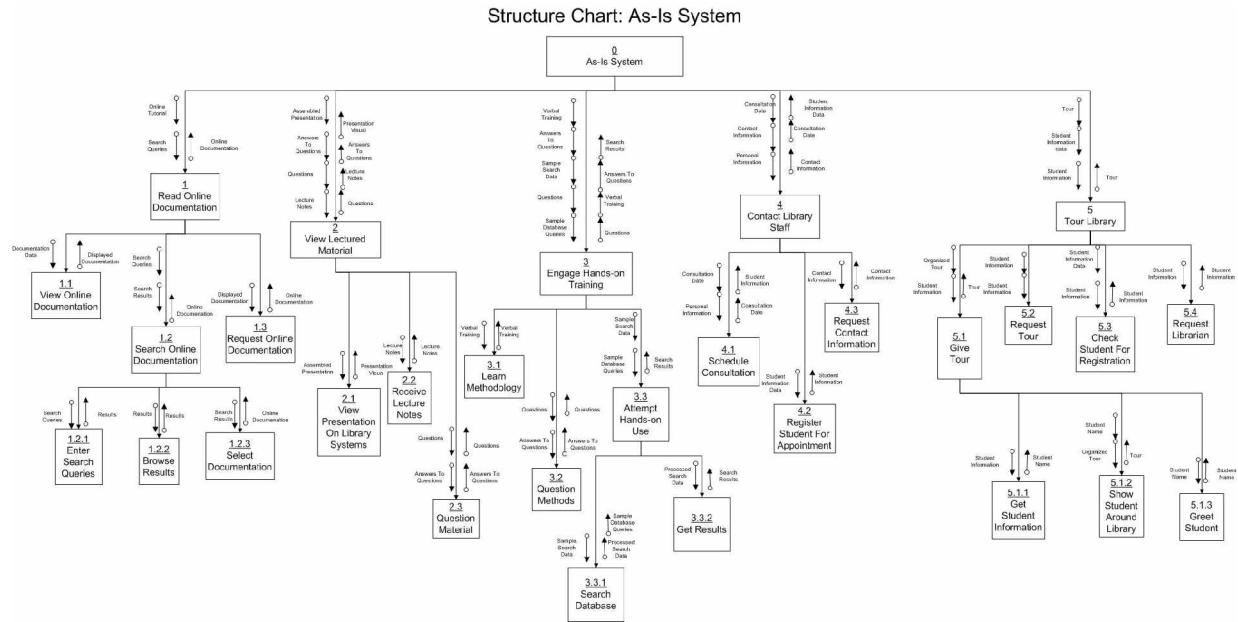
## 19.2 Entity Relationship Diagram (ERD)



# Zeek2Find

## 20 STRUCTURE CHART

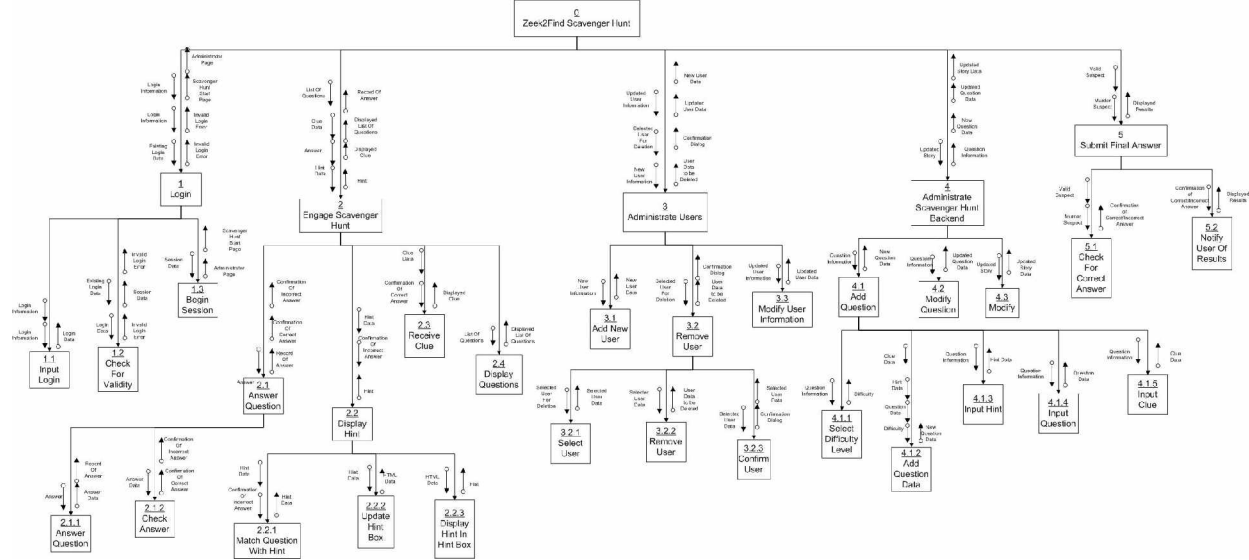
### 20.1 AS-IS Structure Chart



# Zeek2Find

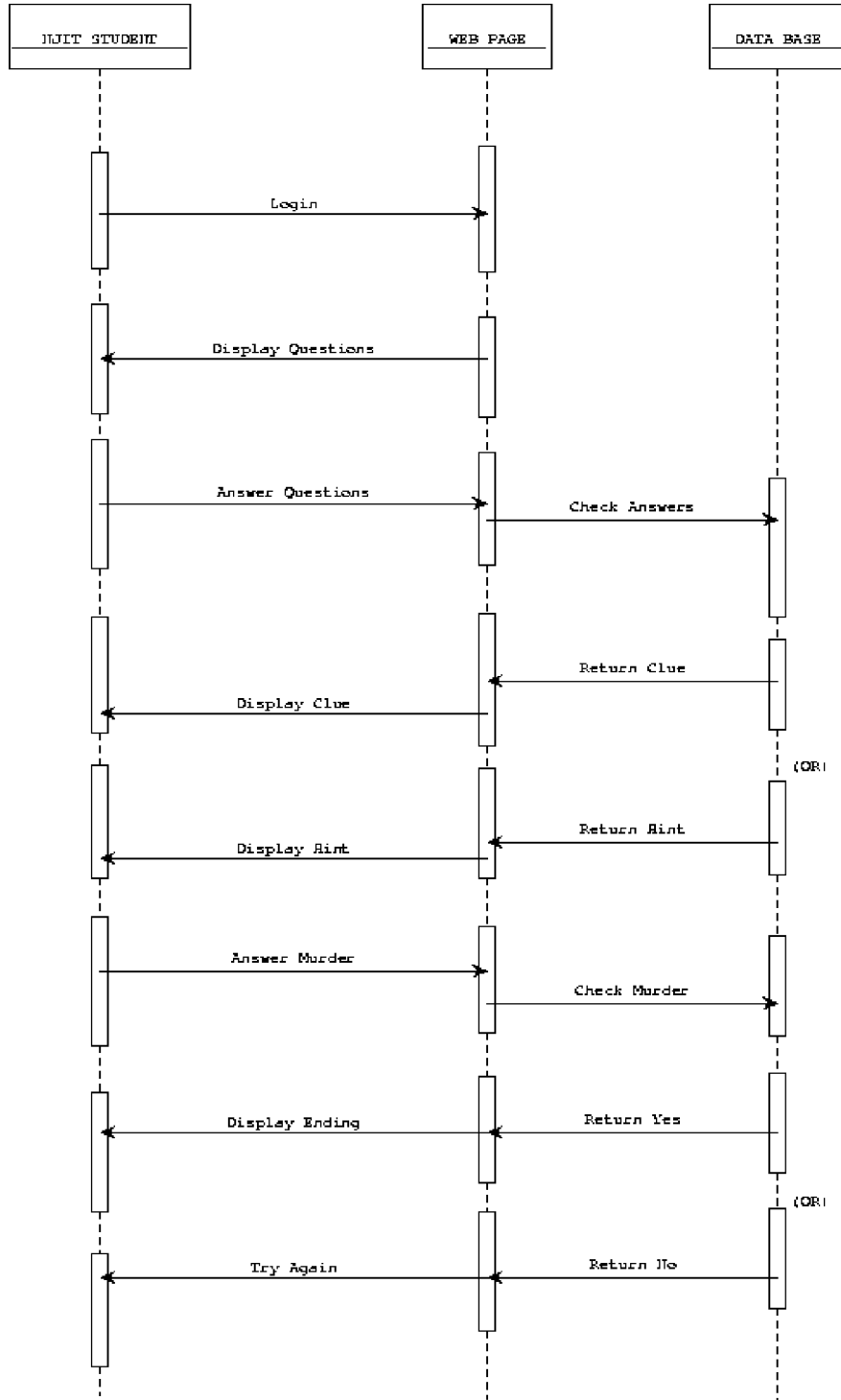
## 20.2 TO-BE Structure Chart

Structure Chart: To-Be System



# Zeek2Find

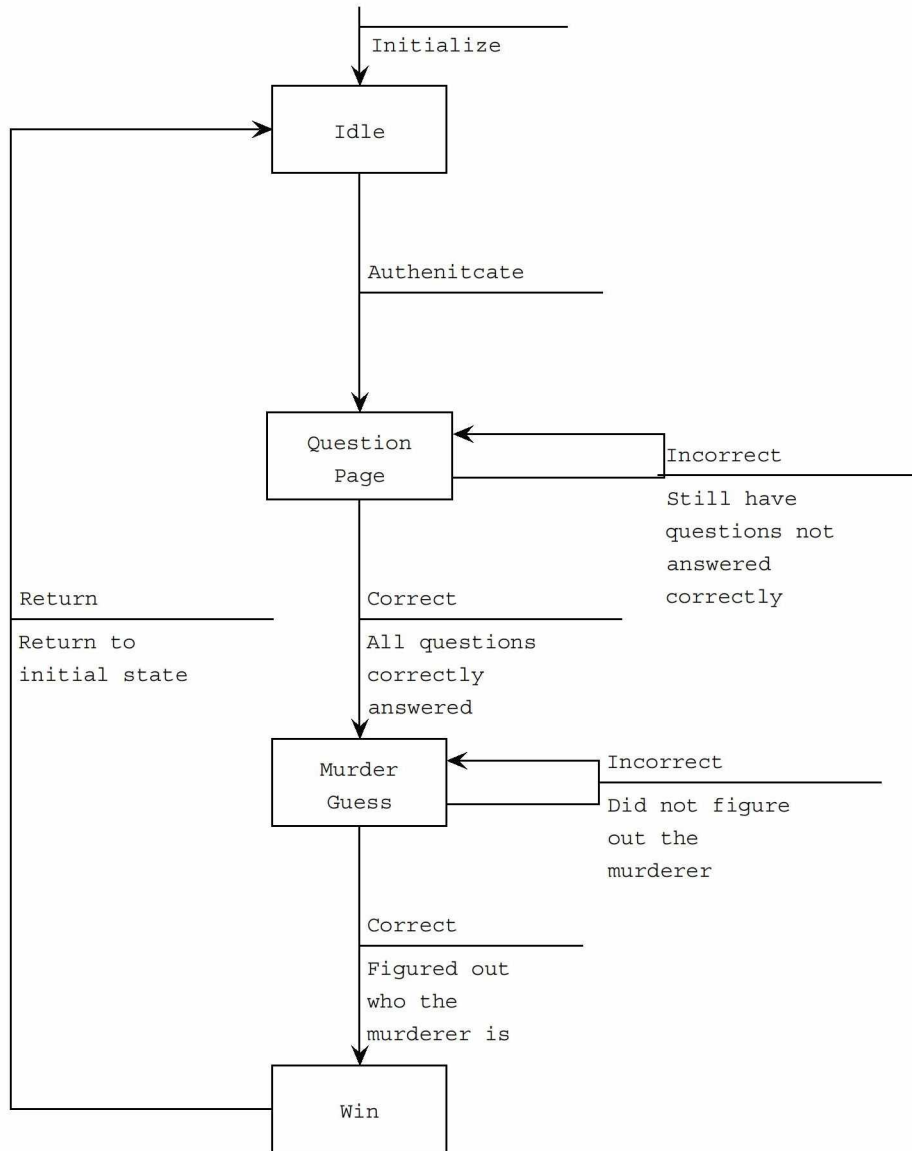
## 21 SEQUENCE DIAGRAM





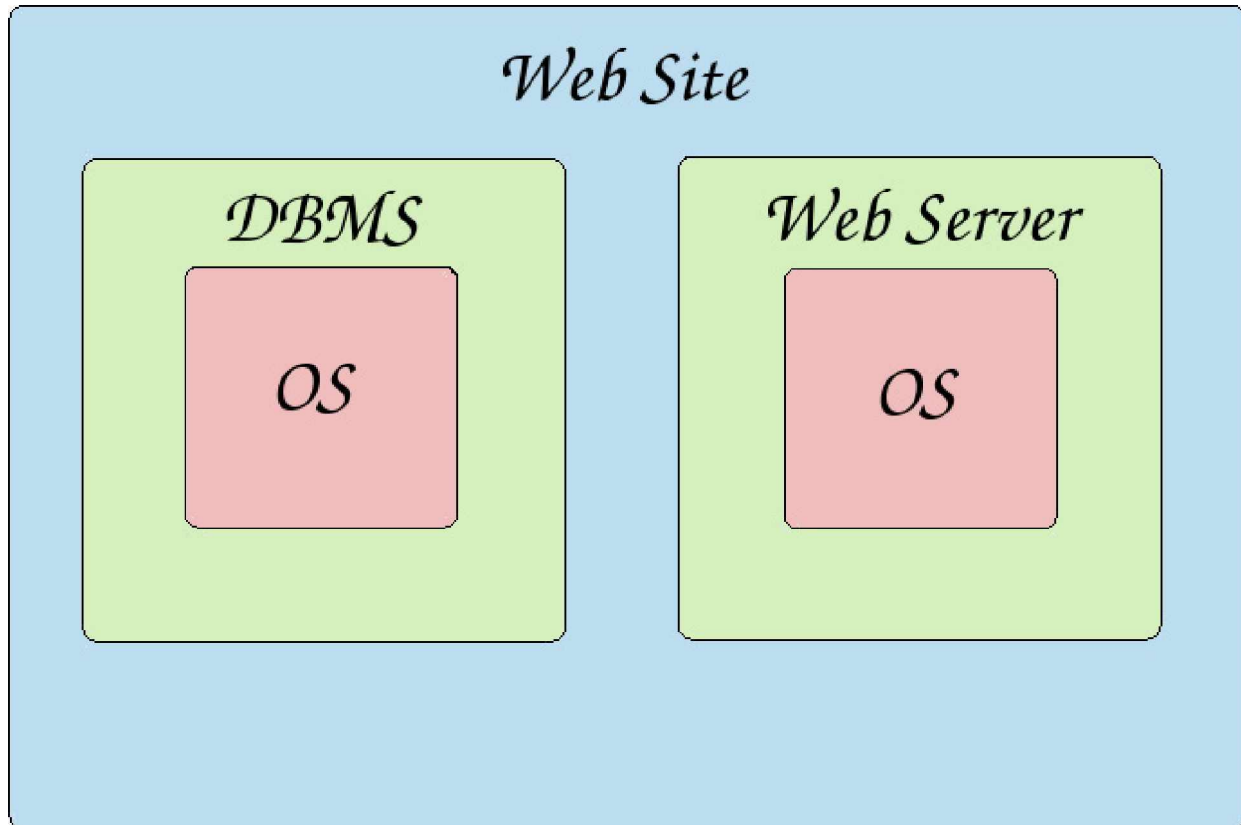
# Zeek2Find

## 22 STATE TRANSITION DIAGRAM



# Zeek2Find

## 23 ABSTRACT MACHINE MODEL



# Zeek2Find

## 24 DATA DICTIONARY

### 24.1 Data Dictionary

Name	BNF
Invalid Login Error(to admin entity)	[Invalid user   SQL Error"]
Administrator Page	menu
Scavenger Hunt Start Page	story + instructions
Invalid Login Error (to user entity)	[“invalid UCID”   “SQL error”]
Displayed Clue	text
Displayed List of Questions	{ question+(hint)+(answer) }
Hint	hintid+text
Record of Answer	text
User Data to be Deleted	userid
New User Data	UCID + level
Updated User Data	UCID + level
Confirmation Dialog	[“yes”   “no”]
Updated Question Data	[“live”   “not live”]+ questiontext+difficulty+answertext
Updated Story Data	[“live” “not live”]+questiontext+introttext+outrottext+killer
New Story Data	[“live”   “not live”]+questiontext+introttext+outrottext+killer
Displayed Results	[outrottext   “try again”]
Login Information	UCID + level + categoryAnswer => text

# Zeek2Find

<b>Murder Suspect</b>	text
<b>Login Information</b>	userid + password
<b>Updated Story</b>	introtext + outrotext + clues + killer
<b>Question information</b>	[live"   "not live"]+questiontext+difficulty+answertext+clue+hintid
<b>Updated user information</b>	ucid + level
<b>New user information</b>	ucid + level
<b>Existing Login Data</b>	UCID + level
<b>List of Questions</b>	{ questiontext + answertext + hintid }
<b>Hint data</b>	{hintid + text}
<b>Clue data</b>	{clueid + text}
<b>Valid suspect</b>	text

# Zeek2Find

## 25 USER INTERFACE DESIGN

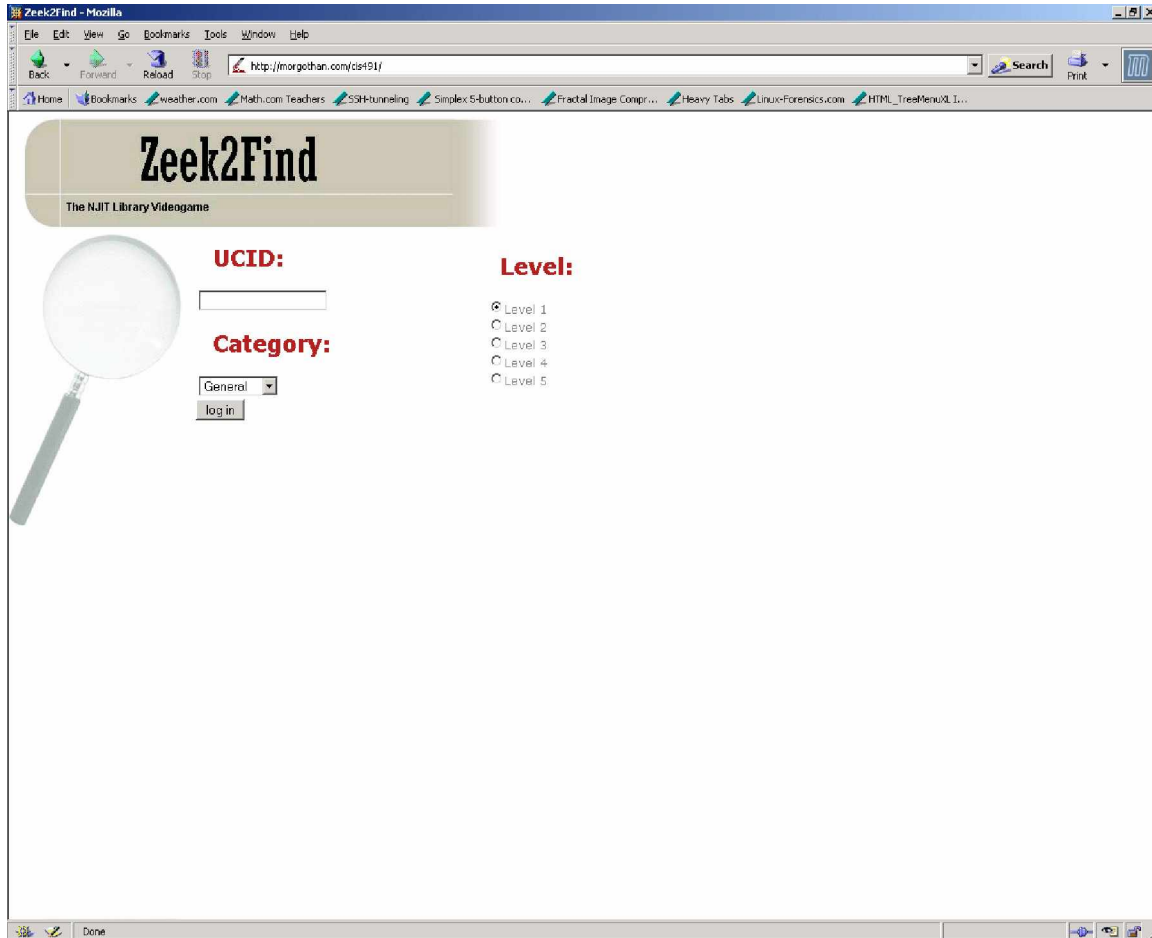
### 25.1 Users Page

Upon loading the main page, users are presented with the following.



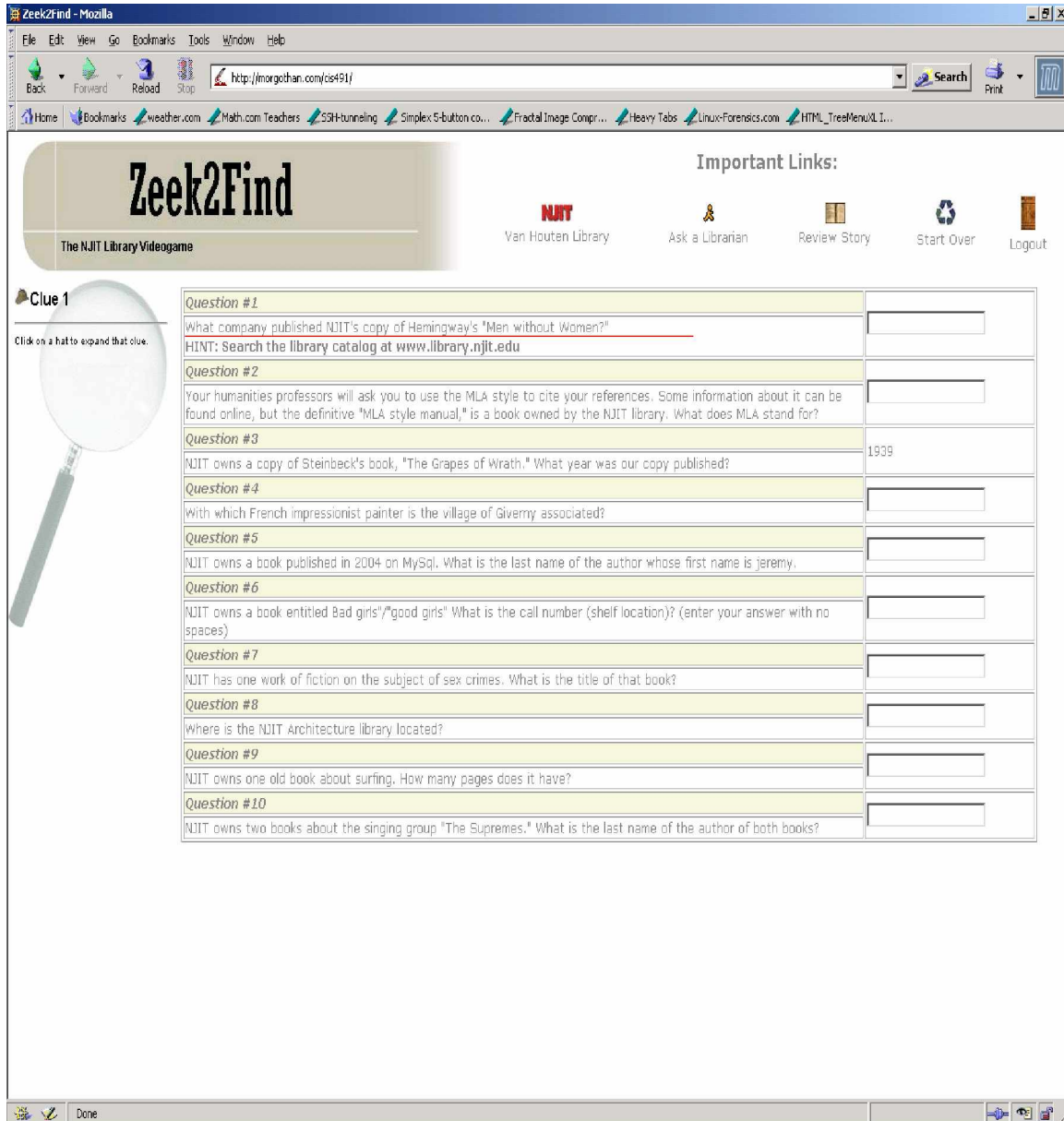
This main screen informs users what their goal is in the game, and presents them with their story. When the “Proceed” button is clicked, the next page loads.

# Zeek2Find



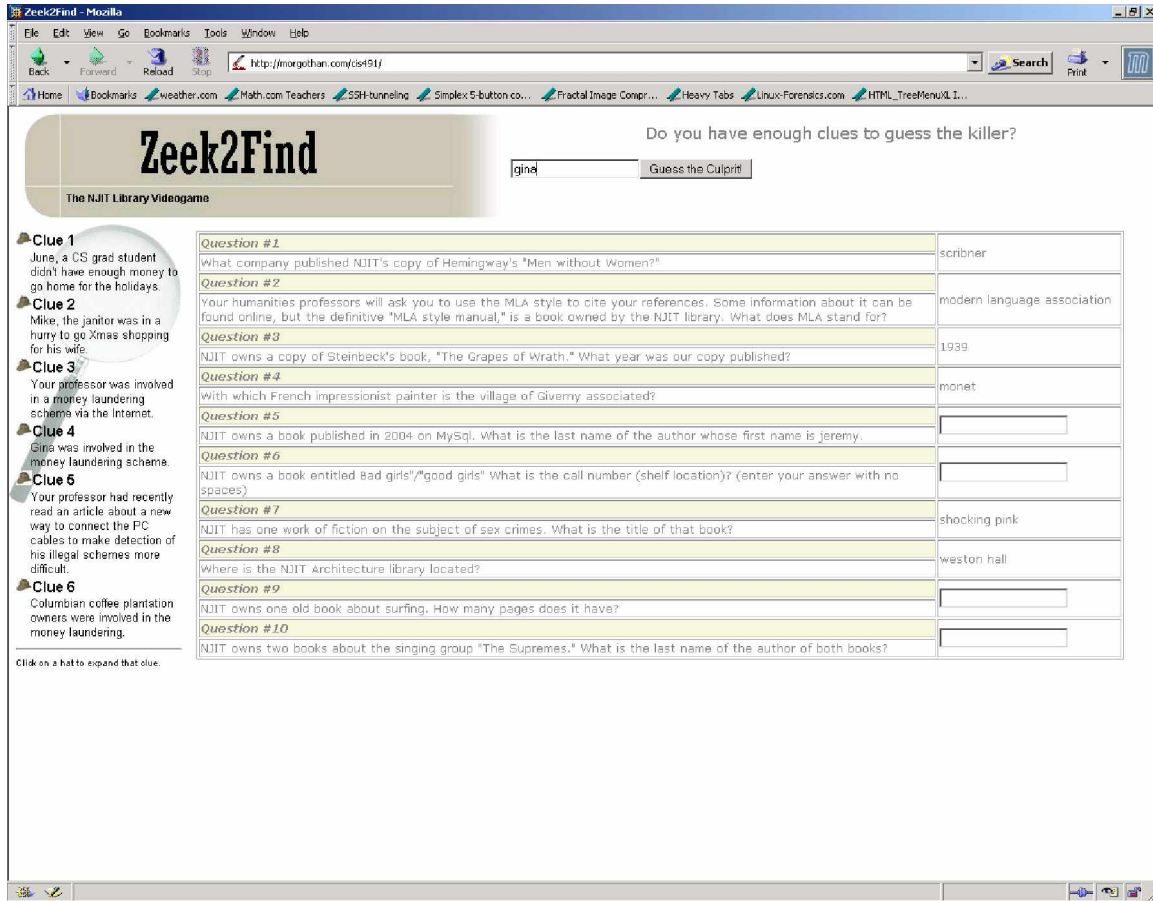
This page allows users to select the level of difficulty, and select the categories of the questions asked to the user. Next, questions are asked of the user, and the user has the chance to answer these questions. The user also has a chance to visit the library, ask a librarian, see the story again, start over, or logout.

# Zeek2Find



Notice that when a user gets a question correct, as in the third question, they are given a clue on the left frame pertaining to the mystery. However, if the user gets the questions wrong, they will be given a hint that tells them where to look to find the answer. After six questions are answered correctly, the user is given the chance to take a guess at who the killer is.

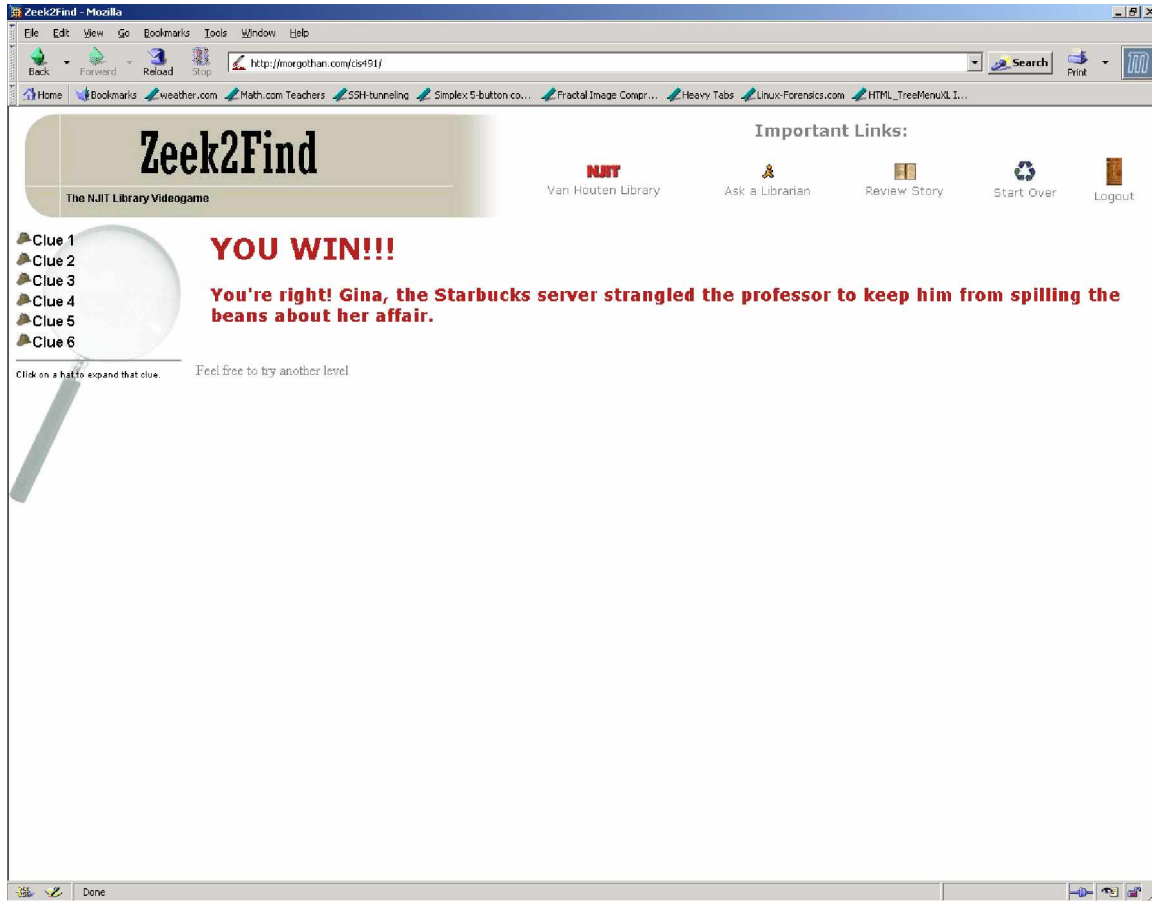
# Zeek2Find



If the guess is correct, they are then presented with the final screen.



# Zeek2Find

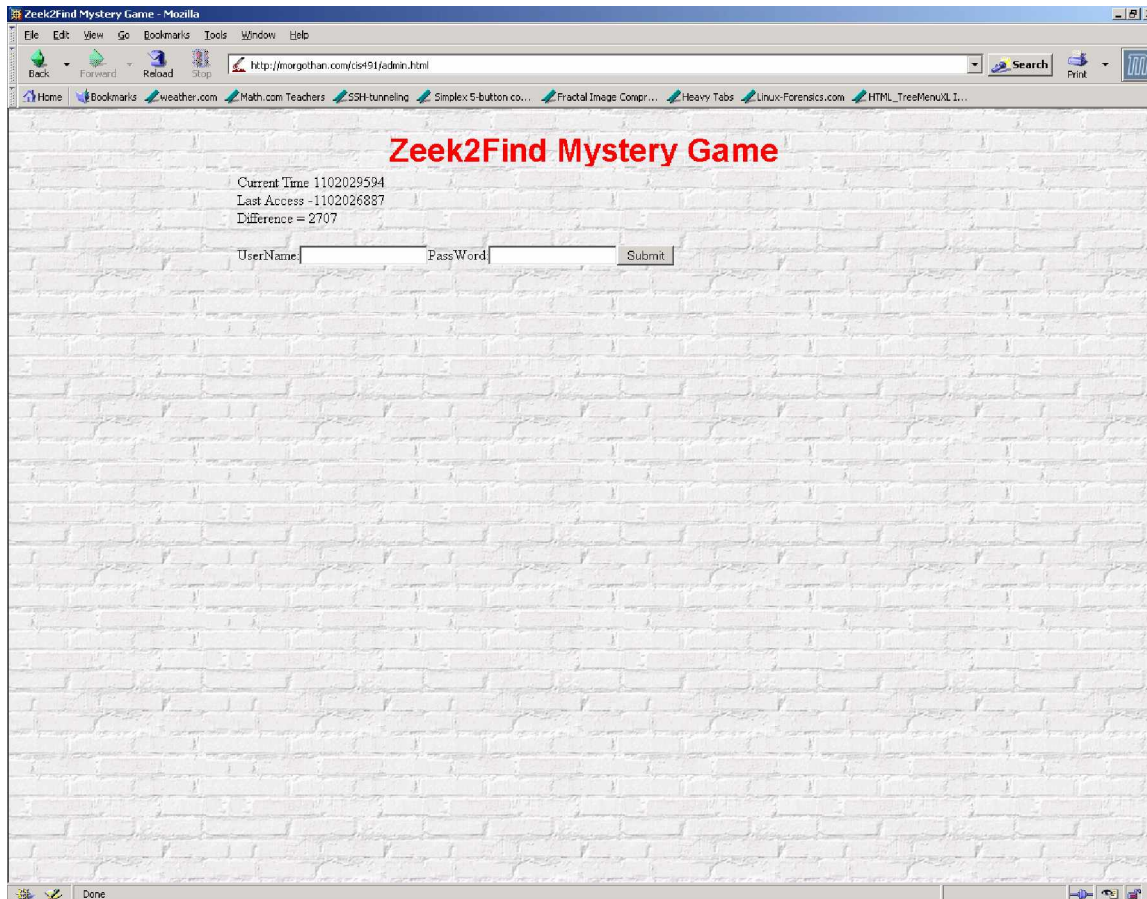


The user has now won the game, and can choose to try another level or logout and stop playing the game.

# Zeek2Find

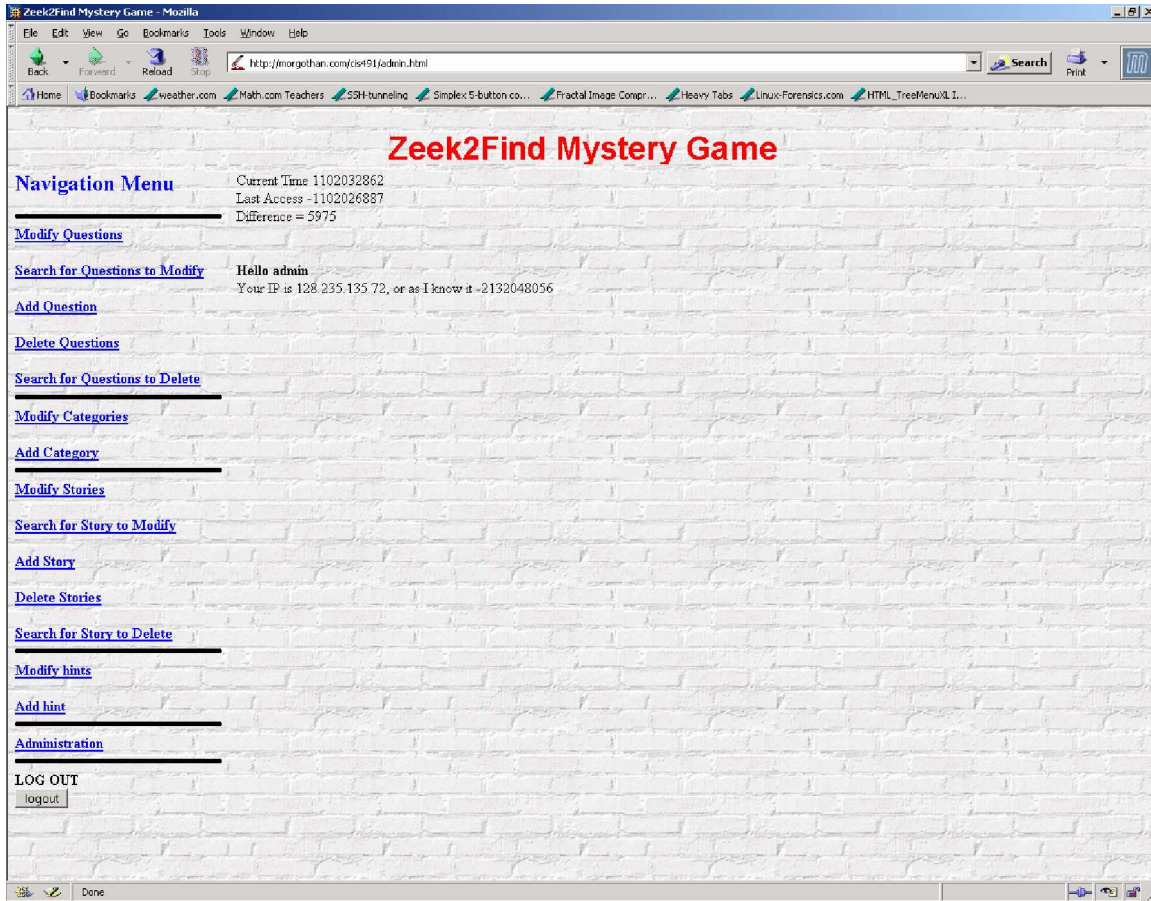
## 25.2 Administration Page

Upon loading the page, an administrator is greeted by the login page.



The administrator gives the correct credentials, and voila!

# Zeek2Find



As you can see, the administrator can then add, delete or modify stories, hints, and questions. The page is extremely intuitive, and by request from our sponsor, we've added a search option, that helps you locate your question very quickly.

**26 SURVEY SECTION**

For one of our research methods our team chose to do surveys. We surveyed almost a hundred people. They were current students, prospective students, administrators, and parents of prospective students. The surveys were very helpful even though some of the information was slightly skewed because of the high number of people who never were inside the Van Houten Library. However, it did give us valuable insight into what people thought about interactive learning. The next section is the survey that was distributed to the students, administrators, and parents. Following the survey are charts to show the results of the task.

# Zeek2Find

## 26.1 Survey

Survey

---

- What is your affiliation with NJIT? (Circle One)
 

Prospective student	Parent	Faculty
Administrator	Current Student	
  
- Have you ever been to the NJIT library? (Circle One)
 

Yes | No
  
- Have you ever used an interactive form of learning? (Circle One)
 

Yes | No
  
- Do you prefer an interactive learning technique as opposed to traditional instructive tutorials? (Circle One)
 

Yes | No
  
- Would you find an interactive instructional tour a good/better way of learning how to use the NJIT library? (Circle One)
 

Yes | No

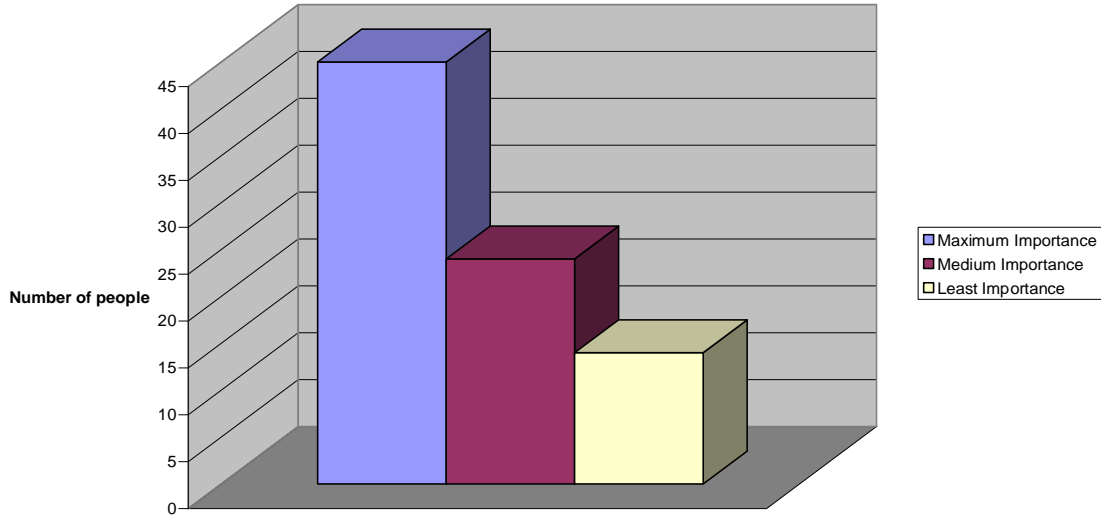
  - o Rate these traits in importance in an interactive system, 1 being most important and 3 being least important: (Circle One Per Line)
 

Easy	1	2	3
Entertaining	1	2	3
Informative	1	2	3
  
- Which interactive method would you prefer as an instructional tutorial? (Circle One)
 

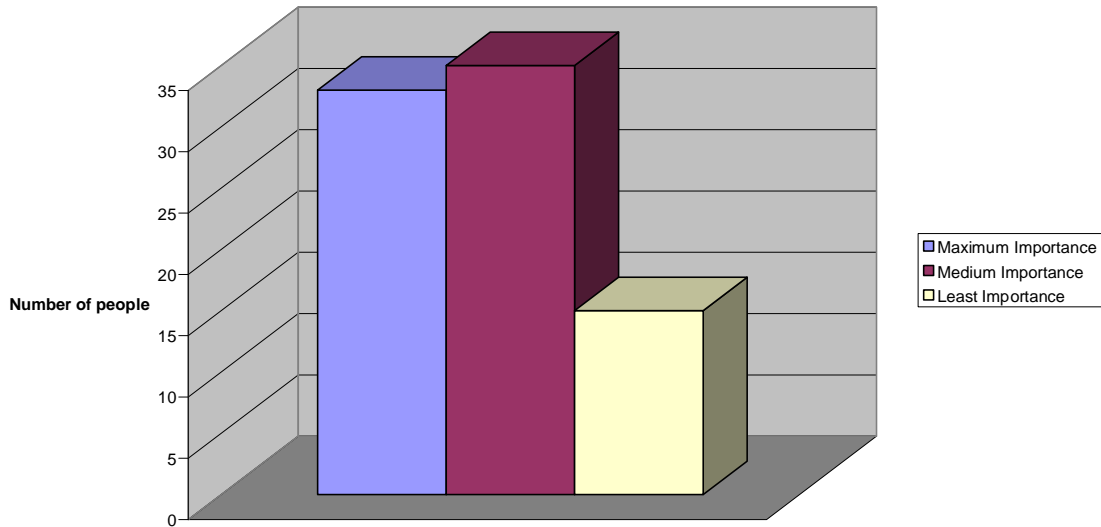
Scavenger Hunt	Murder/Mystery	Puzzles
Crosswords	Online Tutorials	Other

## 26.2 Survey Results

Levels of Importance for Easy System

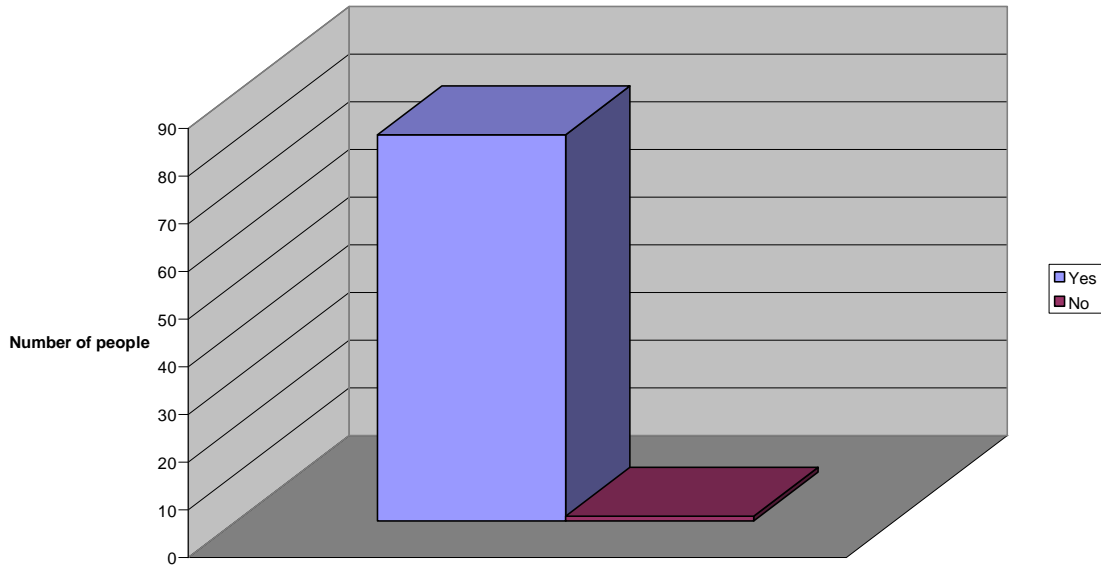


Levels of Importance for Entertaining System

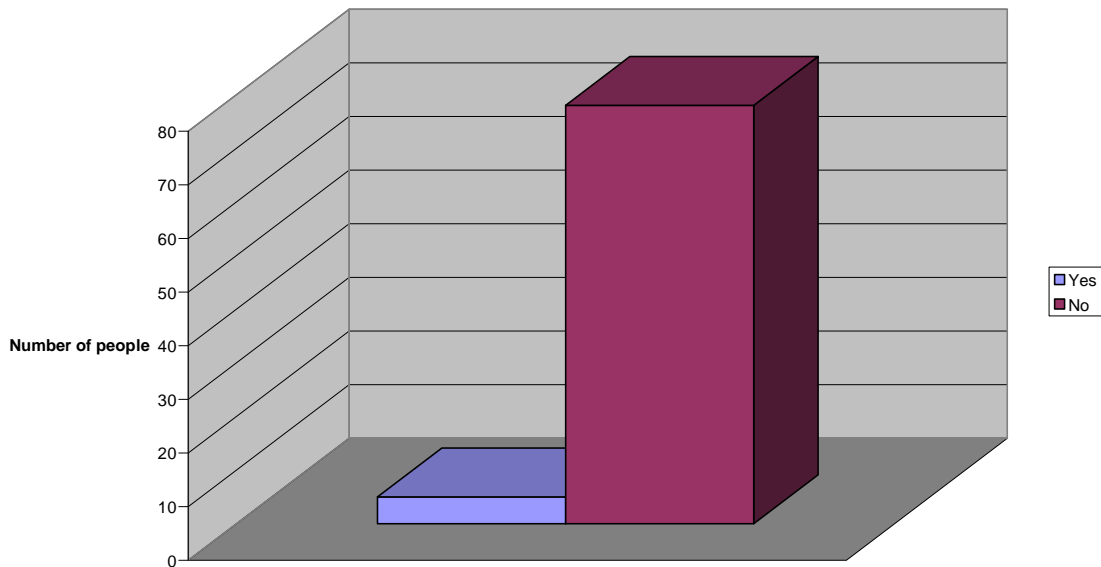


# Zeek2Find

People who would find Interactive Learning Helpful

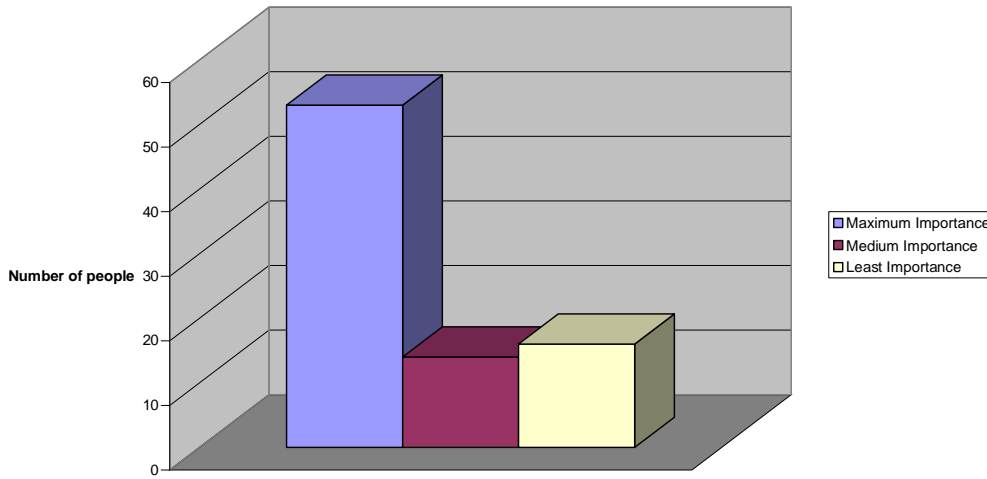


People who have been to the Library

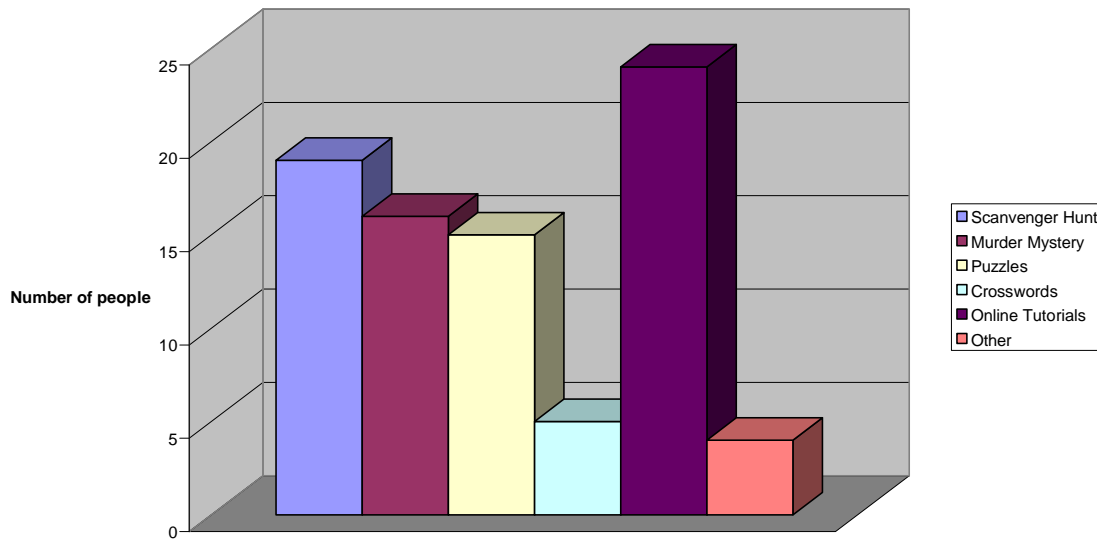


# Zeek2Find

Level of Importance for Informative System



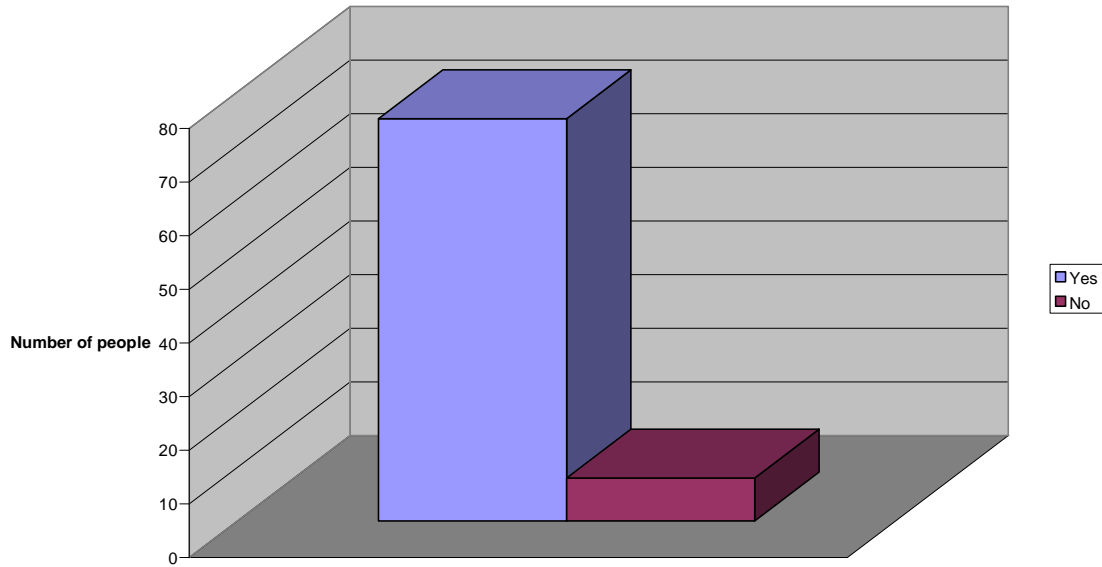
Interactive Methods Preferred



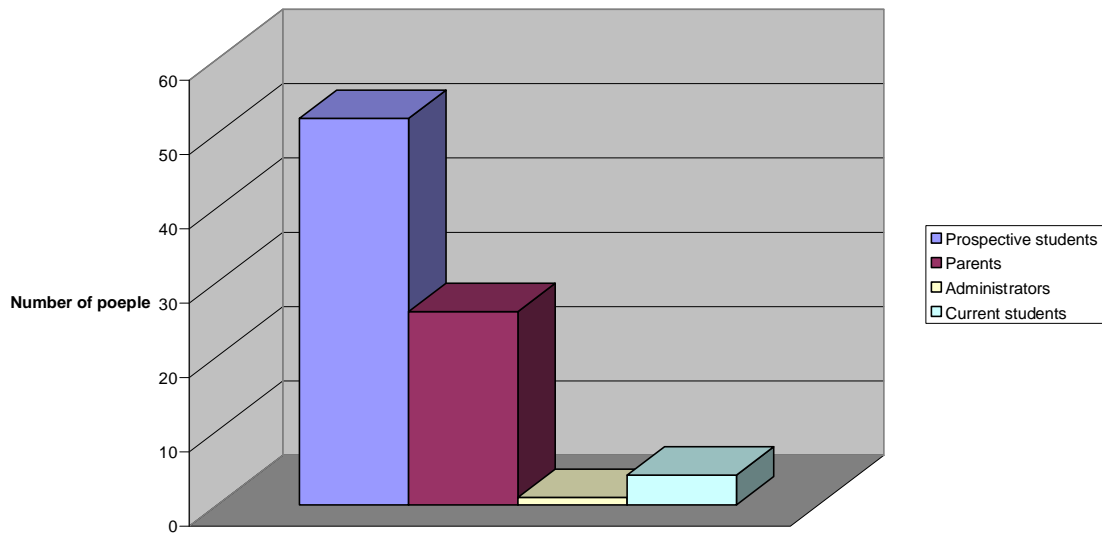


# Zeek2Find

People who prefer an Interactive form of Learning

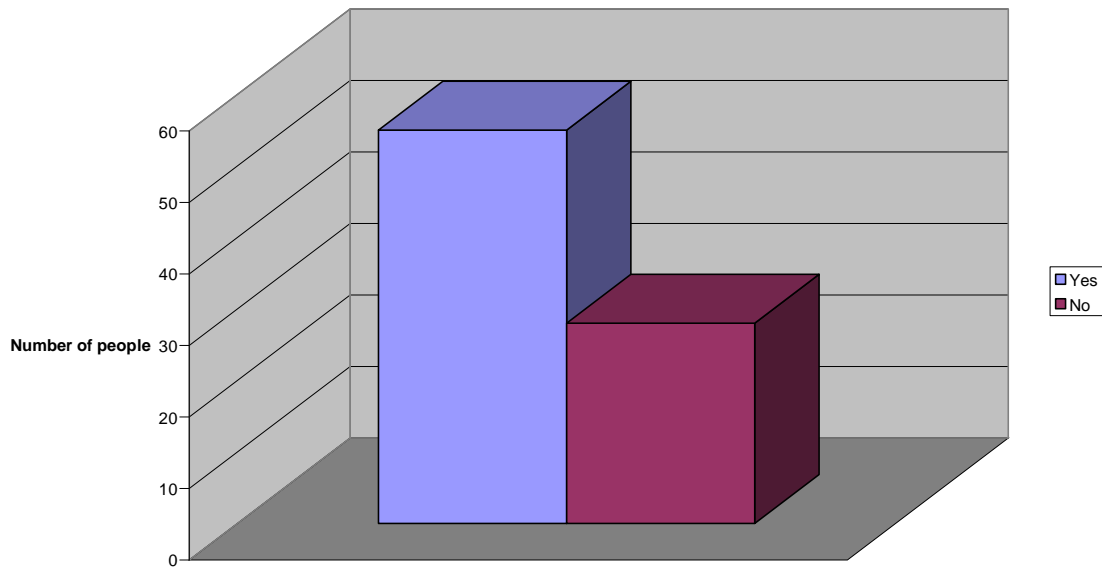


Types of People



# Zeek2Find

People who have used Interactive type of Learning



# Zeek2Find

**27 SOFTWARE TESTING**

A formal test case document was created for the testing phase of the project. This document contained comprehensive directions with benchmark questions to obtain a standard way to test the validity, functionality, and stability of our product. In the following sections you will find the introduction to our formal testing document, the known issues of our product, and a table of test cases, in which two users followed the formal test case and documented their results. Comparison of these test results with future versions of our software will give us a basis for debugging and regression testing.

**27.1 Introduction**

The following is a list of test cases to be followed to completely test our Zeek2Find application. Please read ALL of the following instructions before continuing. The procedure is as follows:

- 1.) Record the date of the test, your name, and the current version of the Zeek2Find software.
- 2.) Read each test case completely.
- 3.) Carry out the specified test.
- 4.) Record any problems in the specified blank. If you find no problems, just enter "OK" in the specified field.

Please record all errors, both small and large. Be as detailed as possible in your description. Please do not skip over any of the test cases, even if they become extremely boring or tiring. The validity of this benchmark test directly correlates to the accuracy in which you carry out and record the results of the test cases.

**27.2 Known Issues**

The following are the currently known issues with the Zeek2Find software:

- Questions are not anchored, so that when you answer a question, the page jumps back to the top, instead of remaining in the position where it had been before answering the question.
- When answering a question correctly, a clue pops up on the left-hand side. Currently, the clue pops up in retracted format. We eventually want the clues to pop up in expanded format.
- If there are no stories in the database, the game will not give an error message. Instead the game will go straight to the end, and say that you have won.

<i>Zeek2Find 2004</i>	<h1>Zeek2Find</h1>
<i>CIS 491 Final Project</i>	

## 27.3 Test Cases

ID	Description	Expected Outcome	Tester #1 Results	Tester #2 Results
1	Open the main page.	The main page should open without a problem or error.	OK	No Problem
2	From the main page, test out all of the categories and difficulty levels and make sure 10 questions appear when they are selected.	Each game should have 10 questions.	Some of the category and difficulty combinations lead to 0 questions.	Not all of the categories and difficulties have questions. Some only have a few questions.
3	On the game page, click all of the "Important Links" in the upper right hand corner.	The "library", "ask a librarian", and "review story" links should open in a new browser. The other links should redirect you to another appropriate page.	OK	No Problem
4	Answer a question incorrectly. What happens?	A hint should appear under the question that was answered incorrectly.	OK	One of the hints was chopped off. It appeared that only part of the hint was being shown.
5	Answer the same question incorrectly again. What happens?	A helpful link should appear next to the hint.	No link appeared. The same hint stayed there.	Nothing changed, the hint remained.
6	Answer the same question incorrectly again for a third time. What happens?	Nothing should happen.	OK	No Problem
7	Answer a question correctly. What happens?	The answer should stay next to the question, and a clue should popup in expanded form on the left side of the page.	Yes, but not in expanded form.	Not expanded form.
8	Click on the hat next to the clue. What happens?	The hat should expand or contract depending on what it was previously.	OK	No Problem
9	Click on the hat again.	The hat should expand or	OK	No Problem

Created: <09/08/04> Last Edited: <10/28/04>

# Zeek2Find

	What happens?	contract depending on what it was previously.		
10	Answer more than one question at a time, before pressing enter. What happens?	Each question should either give a hint or a clue respectively.	OK	No Problem
11	Press the back arrow. What happens?	Nothing should happen.	OK	No Problem
12	Press the refresh button. What happens?	Nothing should happen.	OK	No Problem
13	After answering 6 questions correctly, what happens?	The “whodunit” question should appear where the “Important Links” used to be located.	OK	No Problem
14	Answer the “whodunit” question incorrectly. What happens?	The answer box should be cleared, so you can try again.	OK	No Problem
15	Answer the “whodunit” question correctly. What happens?	A closing text should appear in the main frame telling you a short story about the murderer’s motive.	OK	No Problem
16	Do the clues appear to line up with the story, or do they seem to have nothing to do with each other?	The clues should always line up with the story.	OK	No Problem

## 28 REFERENCES

### 28.1 Publications

Hoffer, Jeffery, Joey George, and Joseph Valacich. Modern Systems Analysis and Design. 3<sup>rd</sup> ed. New Jersey: Addison Wesley, 2001.

### 28.2 Websites

Dr. Clue (2004). Scavenger Hunts and Treasure Hunts for Team Building: Retrieved October 10, 2004 from the World Wide Web: <http://www.drclue.com>

Ford, M. LC Call Number Quiz: October 17, 2004 from the World Wide Web: <http://www.lcsc.edu/library/SatchLCall>

Gowans, D. PHP/MYSQL Tutorial: Retrieved October 1, 2004 from the World Wide Web: <http://www.freewebmasterhelp.com/tutorials/phpmysql>

Houghton Mifflin Company. (2004). Fake Out!: Retrieved October 17, 2004 from the World Wide Web: <http://www.eduplace.com/fakeout/>

Infopoll Inc. (2004). How To Write A Good Survey: Retrieved October 10, 2004 from the World Wide Web: <http://www.infopoll.com/support/resources/howto/write.a.good.survey.htm>

NJIT (2004). Van Houten Library @ NJIT: Retrieved September 10, 2004 from the World Wide Web: <http://library.njit.edu>

PHP.net (2004). PHP: A simple tutorial: Retrieved September 26, 2004 from the World Wide Web: <http://www.php.net/tut.php>

Quia (2004). Quia: Retrieved September 18, 2004 from the World Wide Web: <http://www.quia.com/>

St. Joseph School (2004). Librarywork: Retrieved September 17, 2004 from the World Wide Web: <http://www.stjosephsea.org/library/libwkshts.htm>

Unitar.edu (2004). Creating Educational Games for Online Courses: Retrieved October 21, 2004 from the World Wide Web: <http://elc.unitar.edu.my/modules.php?name=News&file=article&sid=34>

<i>Zeek2Find 2004</i>	<h1>Zeek2Find</h1>
<i>CIS 491 Final Project</i>	

W3Schools (2004). PHP Tutorial: Retrieved October 1, 2004 from the World Wide Web:  
<http://www.w3schools.com/php/default.asp>