

THE IMPLEMENTATION, ANALYSIS, AND EVALUATION OF A
HUMANIZED INFORMATION RETRIEVAL CHAT-BOT

A THESIS`

Presented to

The Faculty of the Computer Science Department

Southern University and A&M College

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Kadrica Dimone Tillotson

July 2012

The Graduate School

UMI Number: 1516181

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent on the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 1516181

Copyright 2012 by ProQuest LLC.

All rights reserved. This edition of the work is protected against unauthorized copying under Title 17, United States Code.



ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

The Graduate School
Southern University and A&M College
Baton Rouge, Louisiana
CERTIFICATE OF APPROVAL

Thesis

This is to certify that the thesis of
Kadrica Dimone Tillotson
has been approved by the Examining Committee for the Project
requirement for the Master of Science degree in
Computer Science, July 2012

 7.11.2012

Chair, Thesis Committee:
Dr. Nigel Gwee

 7/11/12

Committee Member:
Dr. Ebrahim Khosravi

 07/11/2012

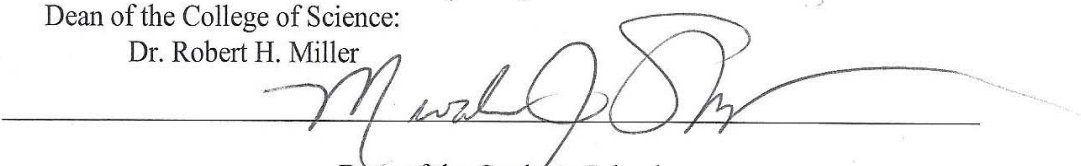
Committee Member:
Dr. Shuju Bai

 7/11/12

Chair, Department of Computer Science:
Dr. Ebrahim Khosravi

 7/23/12

Dean of the College of Science:
Dr. Robert H. Miller



Dean of the Graduate School:
Dr. Mwalimu Shujaa

THE IMPLEMENTATION, ANALYSIS, AND EVALUATION OF A HUMANIZED
INFORMATION RETRIEVAL CHAT-BOT

An Abstract of a Thesis

Presented to

The Faculty of the Computer Science Department
Southern University and A&M College

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Kadrica Dimone Tillotson

July 2012

ABSTRACT

The purpose of this study is to enhance human computer interaction by creating a chat-bot that has the capability to provide users with important information on the faculty and staff located in the Computer Science Department at Southern University. The chat-bot processes the users input and scans through its textual database to extract a proper response. The completed chat-bot is capable of providing the users with important information concerning the faculty. The chat-bot presents users with helpful information in various subjects within the department. The chat-bot is able to supply users with a range of greetings and responses each time the user runs the software. The chat-bot is an information retrieval system integrated with voice based output and operates with the above functions incorporated into it.

ACKNOWLEDGEMENTS

The author wishes to express her sincere appreciation to Dr. Nigel Gwee, whose persistent guidance, assistance, and true devotion was an influential part of the success and completion of this research project. Sincere thanks are also extended to Dr. Shuju Bai for her assistance and the Chairman of the Computer Science Department, Dr. Ebrahim Khosravi, for his encouragement, unselfish support, and true understanding. The author would also like to recognize and thank Mrs. Muriel Johnson for her hard work and dedication to the department.

The author also wishes to acknowledge the unconditional love and tremendous support given to her throughout the years by her mother Ms. Barbara Jean Tillotson and her grandmother Mrs. Mary Tillotson.

TABLE OF CONTENTS

CHAPTER	PAGE
ABSTRACT.....	iv
ACKNOWLEDGEMENTS.....	v
TABLE OF CONTENTS.....	vi
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
I. INTRODUCTION.....	1
1.1 Background.....	1
1.2 Significance of Study.....	4
1.3 Statement of the Problem.....	5
1.4 Research Questions.....	5
1.5 Hypothesis.....	6
1.6 Null Hypothesis.....	6
1.7 Objectives.....	6
1.8 Delimitations.....	7
II. REVIEW OF LITRATURE.....	8
2.1 Evolution of the Chat-Bot.....	8
2.2 Artificial Intelligence.....	12
III. METHODOLOGY.....	16
3.1 Design.....	16
3.2 Implementation.....	17

3.2.1 Regular Expressions in VBScript	17
3.2.2 VBScript RND Function.....	20
3.3 Resources/Instruments	20
IV. DATA ANALYSIS	22
V. CONCLUSION.....	29
5.1 Research Outcome	29
5.3 Future Work.....	30
REFERENCES	33
APPENDIX A.....	36
APPENDIX B.....	38
APPENDIX C	58
VITA	68

LIST OF TABLES

TABLE	PAGE
TABLE 4.1.....	23
TABLE 4.2.....	26
TABLE 4.3.....	27

LIST OF FIGURES

FIGURE	PAGE
Figure 3.1	17
Figure 3.2	19
Figure 4.1	25
Figure 4.2	28
Figure 5.1	29

CHAPTER I

INTRODUCTION

1.1 Background

The most essential component that will mainly be involved in the process of creating the chat-bot will involve the use of AI. In this section a brief introduction is given.

Thinking machines and artificial beings appear in Greek myths, such as Talos of Crete, the bronze robot of Hephaestus and Pygmalion's Galatea (McCorduck, 2004). Human likenesses believed to have intelligence were built in every major civilization: animated cult images were worshipped in Egypt and Greece (Crevier, 1993) and humanoid automatons were built by Yan Shi, Hero of Alexandria and Al-Jazari (McCorduck, 2004). It was also widely believed that artificial beings had been created by Jābir ibn Hayyān, Judah Loew and Paracelsus (O'Connor, 1994). By the 19th and 20th centuries, artificial beings had become a common feature in fiction, as in Mary Shelley's *Frankenstein* or Karel Čapek's *R.U.R. (Rossum's Universal Robots)* (McCorduck, 2004). Pamela McCorduck argues that all of these are examples of an ancient urge, as she describes it, “to forge the gods” (McCorduck, 2004). Stories of these creatures and their fates discuss many of the same hopes, fears and ethical concerns that are presented by artificial intelligence. Mechanical or “formal” reasoning has been developed by philosophers and mathematicians since antiquity. The study of logic led directly to the invention of the programmable digital electronic computer, based on the work of mathematician Alan

Turing and others. Turing's theory of computation suggested that a machine, by shuffling symbols as simple as “0” and “1”, could simulate any conceivable act of mathematical deduction (Berlinski, 2000). This, along with recent discoveries in neurology, information theory and cybernetics, inspired a small group of researchers to begin to seriously consider the possibility of building an electronic brain (McCorduck, 2004).

The field of AI research was founded at a conference on the campus of Dartmouth College in the summer of 1956 (McCorduck, 2004). The attendees, including John McCarthy, Marvin Minsky, Allen Newell and Herbert Simon, became the leaders of AI research for many decades (McCorduck, 2004). They and their students wrote programs that were, to most people, simply astonishing that computers were solving word problems in algebra, proving logical theorems and speaking English (Russell and Norvig, 2003). By the middle of the 1960s, research in the U.S. was heavily funded by the Department of Defense (McCorduck, 2004) and laboratories had been established around the world (Howe, 1994). AI's founders were profoundly optimistic about the future of the new field: Herbert Simon predicted that “machines will be capable, within 20 years, of doing any work a man can do” and Marvin Minsky agreed, writing that “within a generation ... the problem of creating 'artificial intelligence' will substantially be solved” (Crevier, 1993). They failed to recognize the difficulty of some of the problems they faced. In 1974, in response to the criticism of England's Sir James Lighthill and ongoing pressure from Congress to fund more productive projects, the U.S. and British governments cut off all undirected, exploratory research in AI. The next few years, when funding for projects was hard to find, would later be called an “AI winter” (Russell and Norvig, 2003).

In the early 1980s, AI research was revived by the commercial success of expert systems, (McCorduck, 2004) a form of AI program that simulated the knowledge and analytical skills of one or more human experts. By 1985 the market for AI had reached over a billion dollars. At the same time, Japan's fifth generation computer project inspired the U.S and British governments to restore funding for academic research in the field. However, beginning with the collapse of the Lisp Machine market in 1987, AI once again fell into disrepute, and a second, longer lasting AI winter began (McCorduck, 2004).

In the 1990s and early 21st century, AI achieved its greatest successes, albeit somewhat behind the scenes. Artificial intelligence is used for logistics, data mining, medical diagnosis and many other areas throughout the technology industry (McCorduck, 2004). The success was due to several factors: the increasing computational power of computers (see Moore's law), a greater emphasis on solving specific sub problems, the creation of new ties between AI and other fields working on similar problems, and a new commitment by researchers to solid mathematical methods and rigorous scientific standards (McCorduck, 2004). On 11 May 1997, Deep Blue became the first computer chess-playing system to beat a reigning world chess champion, Garry Kasparov (McCorduck, 2004). In 2005, a Stanford robot won the DARPA Grand Challenge by driving autonomously for 131 miles along an unrehearsed desert trail.

The first computer programs to use pattern matching were text editors. At Bell Labs, Ken Thompson extended the seeking and replacing features of the QED editor to accept regular expressions. Early programming languages with pattern matching constructs include SNOBOL from 1962, SASL from 1976, NPL from 1977, and KRC from 1981.

The first programming language with tree-based pattern matching features was Fred McBride's extension of LISP, in 1970 (McCorduck, 2004).

1.2 Significance of the Study

This study will be an effort in advancing the Department of Computer Science and developing a more composed environment for both students and faculty. This study is directed towards the specific purpose of enhancing HCI using a information retrieval based software. This chat-bot is designed to provide needed information to the students regarding the faculty and staff located in the department, in case a student encounters a problem and needs to contact a specific member for help. In case the wireless network on campus fails to operate and the faculty or staff member has left for the day, this software will give the students the needed resources to get in contact with that particular person without needing access to the internet. This software could be beneficial to a department that is short on staff and has only one secretary. This application will be particularly helpful in lessening the interruptions and workload for that specific employee.

The chat-bot has been designed to operate in a humanistic manner with voice based output and provide a number randomized ways to greet and interact with the user. The chat-bot starts off by greeting the user with a message asking him or her to enter the name of the faculty or staff member that he or she is trying to reach. The chat-bot will either output the correct information of that faculty member or give a randomized failed search message, depending on the input the student enters. Being that no human being is perfect, the chat-bot has been intentionally pre-programmed to make minor mistakes here and there to add to the humanistic design.

1.3 Statement of the Problem

The problem is to build a more humanized representation of a chat-bot based software with the necessary resources that have the ability to support the enhancement of human computer interaction. The chat-bot will have the ability to provide the students with important information about the faculty and staff working in the department in case of an emergency. The main focus during the design stage in this software will be centered on designing the chat-bot with a humanized based interaction for the users thus incorporating the voice activation to aid in this goal. In order for the chat-bot to be able to emulate a human, it will be programmed with a number of randomized responses to provide the user with a variety of replies each time the software is used. It is important for the responses that the chat-bot gives the user be of a wide variety in order to give it a humanistic design. Therefore, the voice based output feature will be beneficial to accomplishing that design.

1.4 Research Questions

How can a chat-bot that operates as a human be built? This chat-bot needs to be able to understand the input that the user gives. It needs to be able to analyze the input text before giving an appropriate response. In order for this chat-bot to operate in a more humanistic manner what type of techniques and resources can be helpful in accomplishing this? How can these techniques and resources be incorporated with the chat-bot?

For the chat-bot to be useful for information retrieval purposes it needs to be able to output the accurate data that the user needs. How can the chat-bot understand and analyze the input from the user efficiently?

As the chat-bot interacts with the user it collects information that it will have to analyze. The information in this case is the text that is inputted by the user, which is the sentence as a whole and the individual words used to form that sentence. How can this information it is given be translated by the chat-bot to allow it to give the user a proper response?

1.5 Hypothesis

Intelligent Information Retrieval System (I.R.I.S.) will pass the Turing Test and 60% of the participants testing the software will say the responses the chat-bot provided were humanistic.

1.6 Null Hypothesis

The results of the Turing test on I.R.I.S. will prove to be non-conclusive.

1.7 Objectives

The objective of the study is to enhance human computer interaction by creating a chat-bot that has the capability to provide users with important information. The aim is to create a chat-bot that can interact with users in a humanistic manner whilst enhancing the interaction between users and computers. At the end of the study we are to create a humanistic chat-bot that is capable of supplying users of the application with important information, analyze the humanistic level of the completed software application, provide users with important information regarding the departments' faculty and staff, and evaluate the overall usefulness of the finished application.

1.8 Delimitations

This software comes with some delimitations. To begin with, the voice based output feature that is incorporated in the chat-bot does not correspond with the text outputted by the chat-bot. The software will be limited to the Operating System it can run on. Only machines running Windows XP or higher will be compatible with this software. The information the chat-bot provides is only limited to the faculty and staff working in the Computer Science Department at Southern University. The information I.R.I.S. supplies will be limited to only faculty and staff in the CS department and, given the purpose it serves in accordance, the number of ways the chat-bot responds to the user will be limited. I.R.I.S does not operate with voice activation. Internet Explorer is the only browser that is compatible with the software. The only language that I.R.I.S. understands and can process is English.

CHAPTER II

REVIEW OF LITRATURE

2.1 EVOLUTION OF THE CHAT-BOT

This section gives a review of literature in relation to some of the prior work that has been on the subject of Artificial Intelligence which is similar to the idea of this thesis.

A chat-bot is a computer program designed to simulate an intelligent conversation with one or more human users via auditory or textual methods, primarily for engaging in small talk. The primary aim of such simulation has been to fool the user into thinking that the program's output has been produced by a human. Programs playing this role are sometimes referred to as Artificial Conversational Entities, talk bots or chatterboxes. In addition, however, chatterbots are often integrated into dialog systems for various practical purposes such as online help, personalized service, or information acquisition. Some chatterbots use sophisticated natural language processing systems, but many simply scan for keywords within the input and pull a reply with the most matching keywords, or the most similar wording pattern, from a textual database. The term “ChatterBot” was originally coined by Michael Mauldin (Creator of the first Verbot, Julia) in 1994 to describe these conversational programs (Mauldin, 1994).

In 1950, Alan Turing published his famous article “Computing Machinery and Intelligence” (Turing, 1950) which proposed what is now called the as a criterion of intelligence. This criterion depended on the ability of a computer program to impersonate

a human in a real-time written conversation with a human judge, sufficiently well that the judge is unable to distinguish reliably on the basis of the conversational content alone between the program and a real human. The notoriety of Turing's proposed test stimulated great interest in Joseph Weizenbaum's program , published in 1966, which seemed to be able to fool users into believing that they were conversing with a real human. Eliza's key method of operation involves the recognition of cue words or phrases in the input, and the output of corresponding pre-prepared or pre-programmed responses which can move the conversation forward in an apparently meaningful way (e.g. by responding to any input that contains the word 'MOTHER' with 'TELL ME MORE ABOUT YOUR FAMILY'). Thus an illusion of understanding is generated, even though the processing involved has been merely superficial (Turing, 1950).

ELIZA showed that such an illusion is surprisingly easy to generate, because human judges are so ready to give the benefit of the doubt when conversational responses are capable of being interpreted as “intelligent”. Thus, the key technique here, which characterizes a program as a chat-bot rather than as a serious natural language processing system, is the production of responses which are sufficiently vague and non-specific that they can be understood as “intelligent” in a wide range of conversational contexts. More recently, however, interface designers have come to appreciate that humans' readiness to interpret computer output as genuinely conversational, even when it is actually based on rather simple pattern-matching, can be exploited for useful purposes (Turing, 1950).. Most people prefer to engage with programs that are human-like, and this gives chat-bot-style techniques a potentially

useful role in interactive systems that need to elicit information from users, as long as that information is relatively straightforward and falls into predictable categories. Therefore, for example, online help systems can usefully employ chat-bot techniques to identify the area of help that users require, potentially providing a “friendlier” interface than a more formal search or menu system.

This sort of usage holds the prospect of moving chat-bot technology from Weizenbaum's “shelf ... reserved for curios” to that marked “genuinely useful computational methods”. “The ELIZA effect, in computer science, is the tendency to unconsciously assume computer behaviors are analogous to human behaviors. In its specific form, the ELIZA effect refers only to the susceptibility of people to read far more understanding than is warranted into strings of symbols, especially words, strung together by computers” (Hofstadter, 1996). A trivial example of the specific form of the Eliza effect, given by Douglas Hofstadter, involves an automated teller machine which displays the words “THANK YOU” at the end of a transaction. A casual observer might think that the machine is actually expressing gratitude; however, the machine is only printing a preprogrammed string of symbols (Hofstadter, 1996). More generally, the ELIZA effect describes any situation (Ekbia, 2008) where, based solely on a system's output, users perceive computer systems as having “intrinsic qualities and abilities which the software controlling the (output) cannot possibly achieve”(King, 1995) or “assume that [outputs] reflect a greater causality than they actually do” (Rouse, 2005). In both its specific and general forms, the ELIZA effect is

famous for occurring even when users of the system are conscious of the determinate environment of output formed by the system.

From a psychological standpoint, the ELIZA effect is the result of a subtle cognitive dissonance between the user's awareness of programming limitations and their behavior towards the output of the program. (Ekbia, 2008) The discovery of the ELIZA effect was an important development in artificial intelligence, demonstrating the principle of using social engineering rather than explicit programming to pass a Turing test (Trappl, 2002).

The effect is named for the 1966 chatterbot ELIZA, developed by MIT computer scientist Joseph Weizenbaum. When executing Weizenbaum's DOCTOR script, ELIZA parodied a Rogerian psychotherapist, largely by rephrasing the “patient’s” replies as questions:

Human: Well, my boyfriend made me come here.

ELIZA: Your boyfriend made you come here?

Human: He says I'm depressed much of the time.

ELIZA: I am sorry to hear you are depressed.

Human: It's true. I'm unhappy.

ELIZA: Do you think coming here will help you not to be unhappy? (Güzeldere, 1995).

Though designed strictly as a mechanism to support “natural language conversation” with a computer (Weizenbaum, 1966), ELIZA's DOCTOR script was found to be surprisingly successful in eliciting emotional responses from users who, in the course

of interacting with the program, began to ascribe understanding and motivation to the program's output. (Suchman, 1987) As Weizenbaum later wrote, "I had not realized ... that extremely short exposures to a relatively simple computer program could induce powerful delusional thinking in quite normal people" (Weizenbaum, 1976). Indeed, ELIZA's code had not been designed to evoke this reaction in the first place. Upon observation, researchers discovered users unconsciously assuming ELIZA's questions implied interest and emotional involvement in the topics discussed, even when they consciously knew that ELIZA did not simulate emotion (Billings, 2007).

More recent notable programs include A.L.I.C.E., Jabberwacky and D.U.D.E. While ELIZA and PARRY were used exclusively to simulate typed conversation, many chatterbots now include functional features such as games and web searching abilities. Chatterbots are often integrated into the dialog systems of automated online assistants, giving them the ability of, small talking or engaging in casual conversations unrelated to the scopes of their primary expert systems. Companies which include Lloyds Banking Group, Royal Bank of Scotland, Renault and Citroën are use automated online assistants instead of call centers to provide a first point of contact. (Agence Nationale pour la Recherche and CNRS, 2006).

2.1 ARTIFICIAL INTELLIGENCE

One pertinent field of AI research is natural language processing. Usually, weak AI fields employ specialized software or programming languages created specifically for the narrow function required. For example, A.L.I.C.E. utilizes a programming language called AIML which is specific to its function as a conversational agent, and

has since been adopted by various other developers of, so called, Alicebots. However, A.L.I.C.E. is still purely based on pattern matching techniques without any reasoning capabilities, the same technique ELIZA was using back in 1966. This is considered to not be strong AI, which would require sapience and logical reasoning abilities.

Artificial intelligence is the intelligence of machines and the branch of computer science that aims to create it. AI textbooks define the field as “the study and design of intelligent agents” (Poole, 1998) where an intelligent agent is a system that perceives its environment and takes actions that maximize its chances of success.(Russell, 2003) John McCarthy, who coined the term in 1956, (Skillings, 2006) defines it as “the science and engineering of making intelligent machines.” (McCarthy, 2007) AI research is highly technical and specialized, deeply divided into subfields that often fail to communicate with each other. (McCorduck, 2004) Subfields have grown up around particular institutions, the work of individual researchers, the solution of specific problems, longstanding differences of opinion about how AI should be done and the application of widely differing tools. The central problems of AI include such traits as reasoning, knowledge, planning, learning, communication, perception and the ability to move and manipulate objects (Russell and Norvig, 2003). General intelligence (or “strong AI”) is still among the field's long term goals (Kurzweil, 2005).

The purpose of this thesis is to create a chat-bot with the aim of providing users with more humanized responses that can be used in a practical manner and be beneficial at the same time. A similar research was done at the University of Huddersfield in 2001.

A Research group at this University started this research in August of 2001 that ran for a period of eleven months. The purpose of this project was to create a chat-bot that would enhance the learning of social theory. They were to develop four knowledge bases for key social and/or political theorists for use with an open source chat-bot via the WWW, test them in a teaching context using the chat-bot in Blackboard, disseminate experience in production and teaching, and make available all materials produced to the UK academic community (Moore and Gibbs, 2002).

Many students of sociology experience problems in learning social theory and philosophy because they find it very hard to get started with reading., need cartoon level introduction, leading on to more complex material, find it hard to know what is relevant discussion, so seminar discussion often wanders off topic, need the information provided to reflect their stage of learning, and want to talk with experts when they need help. This is not usually in seminars/tutorials but when essays are being written, which increases the burden on staff via remedial work (Moore and Gibbs, 2002).

In this context, an accessible chat-bot with a knowledge base reflecting key areas of social theory or philosophy could make an important educational contribution. Many universities are moving rapidly to the use of Virtual Learning Environments (VLEs) or Managed Learning Environments (MLEs) such as Blackboard to support student learning (Moore and Gibbs, 2002). Whilst it is fairly easy in these systems to make course documents and lecture and seminar notes available to students, there is still much work to be done, both developmentally and in staff training, to create more

innovative ways of using these systems in teaching. This is especially so in sociology, where it seems that the only additional use of MLEs is for online discussion. Many academics, rightly, are concerned that this will take up even more of their time in supporting students. One important response to this is to encourage students to converse with and learn from each other (Moore and Gibbs, 2002). This has been tried over the last few years by systems such as coMentor. Whilst this frees up staff time, one disadvantage is that the quality of discussion and hence of the information that others might vicariously experience may be questionable. The use of a chat-bot in such a context, whilst it cannot create the same kind of quality of conversation as human participants, can, at least, ensure high accuracy of the content of the conversation (Moore and Gibbs, 2002).

The project ran for 11 months. It started in August 2001. During the first semester 2001-2 staff will developed and checked the knowledge bases. In December 2001 the research assistant was appointed. During the second semester January to May 2002 the material was evaluated in a teaching context and the web site was established. In the end a functioning web-based chat-bot called Emile was able to be produced and in June of 2002 a final report was produced and a workshop was offered (Moore and Gibbs, 2002).

CHAPTER III

METHODOLOGY

The final goal of this thesis is to create a humanistic chat-bot and enhance human computer interaction whilst providing the users of the software with valuable information. The chat-bot should be able to operate in a humanistic form by providing the users with a variety of responses. In addition, the chat-bot should be able to communicate with the user both textually and verbally. Last but not least, the chat-bot should be able to provide the students with important information in regards to the faculty and staff located in the department. Having the chat-bot have a range of responses whilst including the voice based output are the two major components of this study that supports the humanistic build of the software. Each of the previously stated components are two separate functions but are interrelated in creating the finished application.

3.1 DESIGN

Figure 3.1 diagram shows the process of the software which starts by prompting the user for input. The software then scans the users input for keywords within its textual database to extract an appropriate response. Finally the software outputs a response to the user and this process is repeated until the user exits.

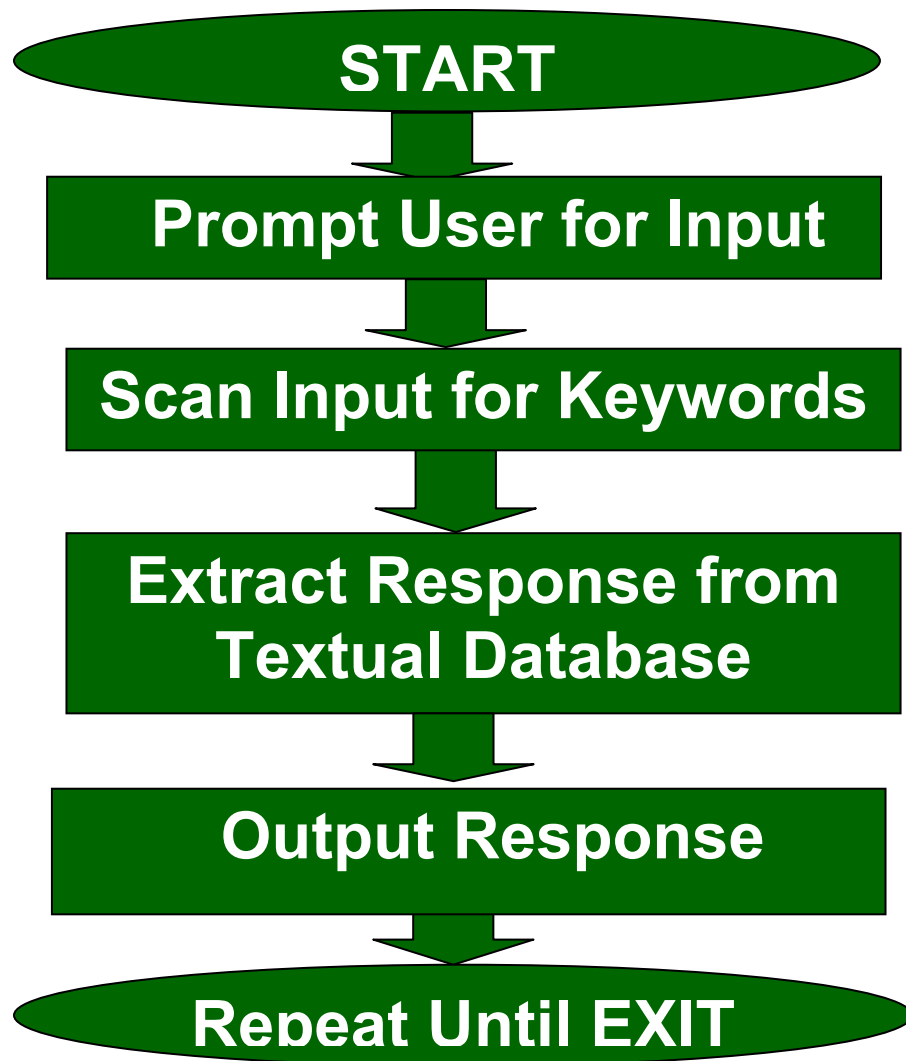


Figure 3.1: Diagram displaying I.R.I.S.'s software application process

3.2 IMPLEMENTATION

3.2.1 Regular Expressions in VBScript

Reading the users input and being able to extract an efficient response was a major factor in the implementation of I.R.I.S. To be able to do this effectively patterns had to be defined in order to efficiently read the users input. The following is a VBScript object used to implement the pattern matching technique used for I.R.I.S. There are three properties that are available for the RegExp object:

Global - if TRUE, find *all* matches, if FALSE find only the *first* match

IgnoreCase - if TRUE perform a case-*insensitive* search, if FALSE perform a case-sensitive search

Pattern - the RegExp pattern to search for

The following is a pattern implemented in I.R.I.S. to locate a specific course/subject:

EX :

```
Set objRegExpCO = New RegExp
```

```
objRegExpCO.IgnoreCase = True
```

```
objRegExpCO.Pattern = "COMPUTER ORGANIZATION|COMPUTER
```

```
ORG|ORGANIZATION|CO"
```

When executed, the above pattern is matched against the users input string searching for keywords and/or a combination of those keywords listed in the pattern define above.

“|” - This symbol matches a character are a set of characters against a string

EX: For example, “a|food” matches any character in a string beginning with “a” or “food”.

The following is a diagram showing the pattern matching technique I.R.I.S. implements to match keywords to its textual database.

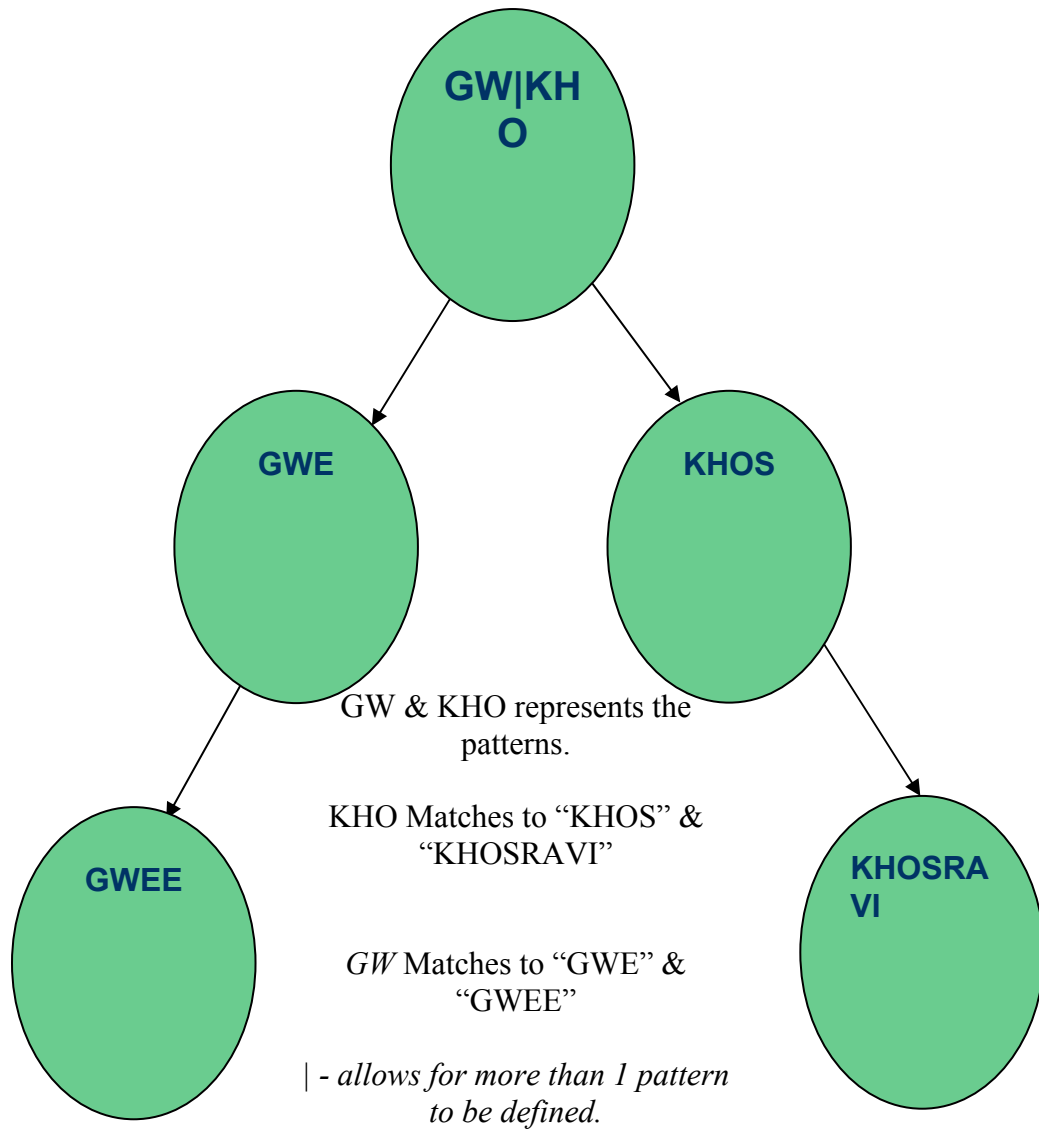


Figure 3.2 Pattern matching technique implemented by I.R.I.S.

3.2.2 VBScript Rnd Function

To add to the humanistic design of I.R.I.S. adding a range of responses and randomizing them was incorporated. In VBScript Randomize uses numbers to initialize the Rnd function random-number generator, giving it a new seed value. If you omit number, the value returned by the system timer is used as the new seed value. If Randomize is not used, the Rnd function (with no arguments) uses the same number as a seed the first time it is called, and thereafter uses the last generated number as a seed value.

The following is a block of code which illustrates the use of the Randomize statement implemented in I.R.I.S.:

```
Randomize //Must always be used ahead of the rnd function //
strTextstart = (welcomestatemnts (rnd)) // (rnd) Randomizes the pre-defined greetings
in the variable "welcomestatemnts"//
Set objvoice = CreateObject ("SAPI.SpVoice")
ObjVoice.Speak strTextstart
Welcome()
```

3.3 RESOURCES/INSTRUMENTS

The following are a list of the main software components that were used in this study to implement the finished information retrieval chat-bot application.

- 1) Adobe Dreamweaver - an application web designers and developers use to create websites and applications for use across multiple targets including browsers, devices, and tablets.

- 2) Adobe Photoshop - an application photographers, graphic designers, web designers, videographers, and 3D artists use to enhance and manipulate photos and create original digital art.
- 3) VbSEdit - VBScript editor that dramatically reduces the time you spend writing.
- 4) Notepad - a basic Windows utility used for creating and/or editing unformatted files.
- 5) Imagicon - Imagicon is a graphics utility designed to convert multiple image files from BMP, JPG or PNG image formats into any of the following image formats (24-bit): BMP, JPG, PNG and ICO.
- 6) Macromedia Flash - a powerful authoring application for creating animation and multimedia content. Design immersive interactive experiences that present consistently across desktops and multiple devices, including tablets, smartphones, and televisions.
- 7) IVONA TTS - IVONA Software develops and delivers speech synthesis solutions that lead the TTS market in natural voice quality, accuracy, and ease of use.

CHAPTER IV

DATA ANALYSIS

Data was gathered from all of the thirty participants to get their view on the software from a student's perspective. Study included total of thirty participants all of which are Adults/Students/Faculty/Staff ages ranging from 18 to 64. Only persons who had successfully completed Computer Literacy were qualified to participate in the study. The purpose for human participants in the research was to gain feedback regarding the software that was created. No sensitive data regarding the participants were collected and their names remained anonymous. Study participants were informed that he or she would not be identified by name in any type of presentation and instead be referred to as Tester 1, Tester 2, etc. All participants were informed that they could withdraw from the study at any time without penalty. Participants were told that they could be terminated from study for sudden uncivil behaviors towards researcher. The participants' failure to complete the study procedures or to answer all questions could result in the data not being used in the study. All thirty participants tested the software application and were given a brief survey regarding the chat-bot's overall performance. All of the thirty study participants were asked a total of five questions after testing the software. To determine the ratio of how humanistic the chat-bot's responses were the participants were given a rating scale and asked to rate how humanized the responses the chat-bot provided were. Secondly, the participants were asked if they preferred a chat-bot with humanized

responses or generic computerized responses. Thirdly, they were asked how efficient the software was at providing information on to them on a scale of one to five with five being very efficient. In Addition, they were asked if the chat-bot had stopped functioning at any point during use. Last but not least, the participants were asked if they felt the software would be useful in a department at a University. All of the ratings and survey results are based on each of the study participant's individual experience upon testing the software application. If a rating of 4 or 5 was selected by the participant, it counted as a vote towards HUMANISTIC. Anything below a rating of 4 counted as a vote against being humanistic.

The following Tables displays the results of each question asked on the survey regarding the I.R.I.S.

Question 1: How humanized were the responses the chat-bot provided?

Question 3: How efficient was the software at providing information on a scale of 1 to 5?

Question 1		Question 3	
Study Participate	Rating	Study Participate	Rating
P1	4	P1	4
P2	4	P2	5
P3	5	P3	5
P4	4	P4	5
P5	4	P5	5
P6	4	P6	5
P7	4	P7	5
P8	4	P8	4
P9	5	P9	5
P10	5	P10	5
P11	5	P11	5
P12	5	P12	5
P13	2	P13	4
P14	1	P14	4

P15	1	P15	4
P16	3	P16	5
P17	3	P17	4
P18	5	P18	5
P19	3	P19	5
P20	2	P20	5
P21	4	P21	5
P22	3	P22	4
P23	5	P23	5
P24	3	P24	4
P25	5	P25	5
P26	2	P26	4
P27	5	P27	5
P28	4	P28	5
P29	4	P29	5
P30	3	P30	4

TABLE 4.1 Study Participants Survey Results for Question 1 and 3

The first question the study participants asked was regarding the responses the chat-bot provided and how humanistic they were on a scale of one to five with five being the best. 2 of the 30 participants voted Not Humanistic, 3 of the 30 participants voted Not Very Humanistic, 6 of 30 participants voted Neutral, 10 of the 30 Participants voted Humanistic, and 9 of the 30 participants voted Very Humanistic. The third question the study participants were asked was regarding how efficient the software was at providing information on to them on a scale of one to five with five being the most efficient. 10 of the 30 participants gave a rating of 4, 20 of the 30 participants gave a rating of 5. The majority of the participants were in agreement that the software was efficient at providing information that was valuable.

How humanized were the responses the chat-bot provided?

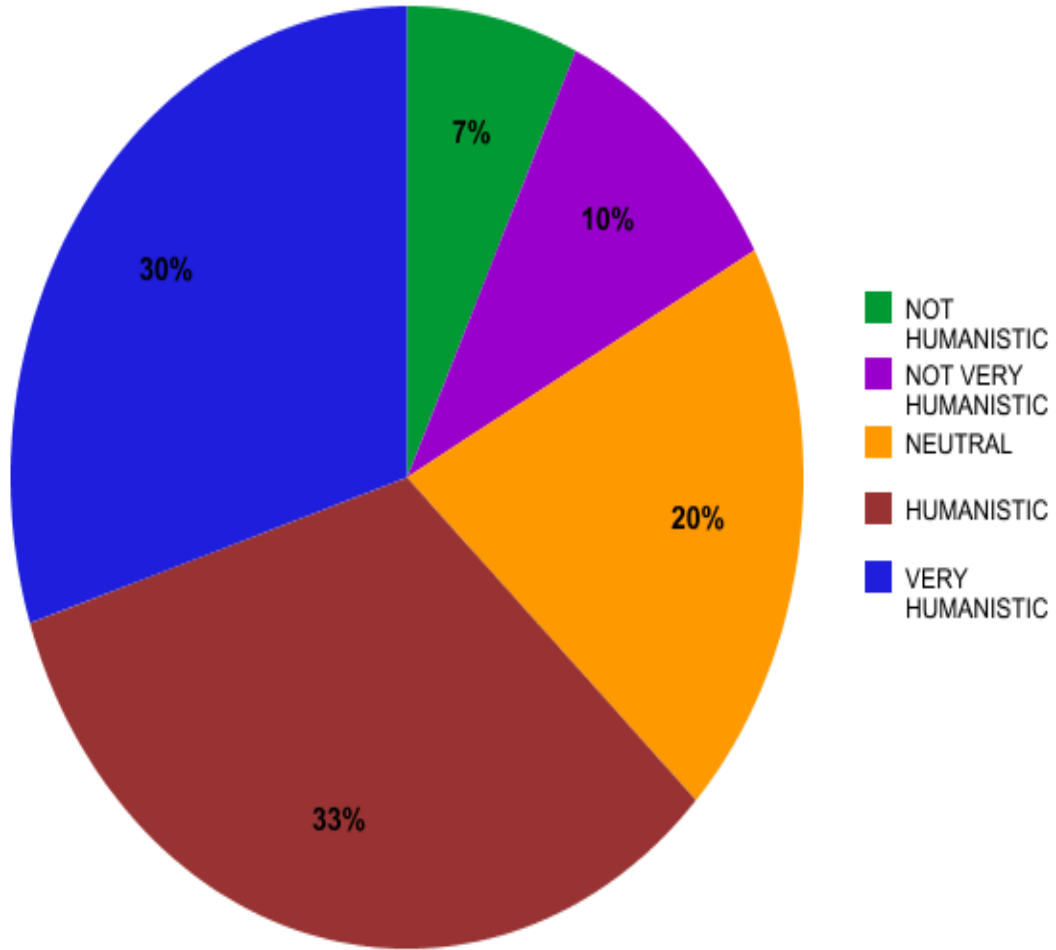


Figure 4.1 Pie Chart displaying the color shaded choices for question number one and the percentage of participants who voted in each category.

Question 2: Would you prefer a chat-bot with humanized responses or generic computerized responses?

Study Participate	Rating
Participate 1	Humanized Responses
Participate 2	Humanized Responses
Participate 3	Humanized Responses
Participate 4	Humanized Responses
Participate 5	Humanized Responses
Participate 6	Humanized Responses
Participate 7	Humanized Responses
Participate 8	Humanized Responses
Participate 9	Humanized Responses
Participate 10	Humanized Responses
Participate 11	Humanized Responses
Participate 12	Humanized Responses
Participate 13	Humanized Responses
Participate 14	Humanized Responses
Participate 15	Humanized Responses
Participate 16	Humanized Responses
Participate 17	Humanized Responses
Participate 18	Humanized Responses
Participate 19	Humanized Responses
Participate 20	Humanized Responses
Participate 21	Humanized Responses
Participate 22	Humanized Responses
Participate 23	Humanized Responses
Participate 24	Humanized Responses
Participate 25	Humanized Responses
Participate 26	Humanized Responses
Participate 27	Humanized Responses
Participate 28	Humanized Responses
Participate 29	Humanized Responses
Participate 30	Humanized Responses

TABLE 4.2 Study Participants Survey Results for Question 2

The second question the participants were asked was if they preferred a chat-bot with humanized responses or generic computerized responses. 30 of the 30 study participants

chose to have a chat-bot who provided humanized responses. Every one of the study participants were in agreement in that they preferred software with humanized responses over generic computerized responses.

Question 4: At any point did the chat-bot stop functioning?

Question 5: Do you think this software would be useful in a department at a University?

Question 4		Question 5	
Study Participate	Rating (NO/YES)	Study Participate	Rating (NO/YES)
P1	NO	P1	YES
P2	NO	P2	YES
P3	NO	P3	YES
P4	NO	P4	YES
P5	NO	P5	YES
P6	NO	P6	YES
P7	NO	P7	YES
P8	NO	P8	YES
P9	NO	P9	YES
P10	NO	P10	YES
P11	NO	P11	YES
P12	NO	P12	YES
P13	NO	P13	YES
P14	NO	P14	YES
P15	NO	P15	YES
P16	NO	P16	YES
P17	NO	P17	YES
P18	NO	P18	YES
P19	NO	P19	YES
P20	NO	P20	YES
P21	NO	P21	YES
P22	NO	P22	YES
P23	NO	P23	YES
P24	NO	P24	YES
P25	NO	P25	YES
P26	NO	P26	YES
P27	NO	P27	YES
P28	NO	P28	YES
P29	NO	P29	YES
P30	NO	P30	YES

TABLE 4.3 Study Participants Survey Results for Question 4 and Question 5

The fourth question the study participants were asked was if the chat-bot had stopped functioning at any time during use. 30 of the 30 study participants chose “NO”. The fifth question and final question the study participants were asked was if they felt the software would be useful in a department at an Educational Institution. The results were unanimous with 30 of the 30 participants choosing “YES”.

Survey/Feedback Form

<p><u>RATING SCALE</u> 1 - Not Humanistic 2 - Not very Humanistic 3 - Neutral 4 - Humanistic 5 - Very Humanistic</p>
--



- 1.) How humanized were the responses the chat-bot provided? (See Rating Scale Above)
- 2.) Would you prefer a chat-bot with humanized responses or generic computerized responses?
- 3.) How efficient was the software at providing information on a scale of 1 to 5? (5 being the VERY EFFICIENT)
- 4.) At any point did the chat-bot stop functioning? (YES/NO, if “YES” please provide the error.)
- 5.) Do you think this software would be useful in a department at a University?(YES/NO)

Figure 4.2: Screenshot of the survey form given to the study participants.

CHAPTER V

CONCLUSION

5.1 RESEARCH OUTCOME

Upon testing the software two of the thirty study participants voted Not Humanistic, three of the thirty participants voted Not Very Humanistic, six of the thirty participants voted Neutral, ten of the thirty participants voted Humanistic, and seven of the thirty participants voted Very Humanistic.

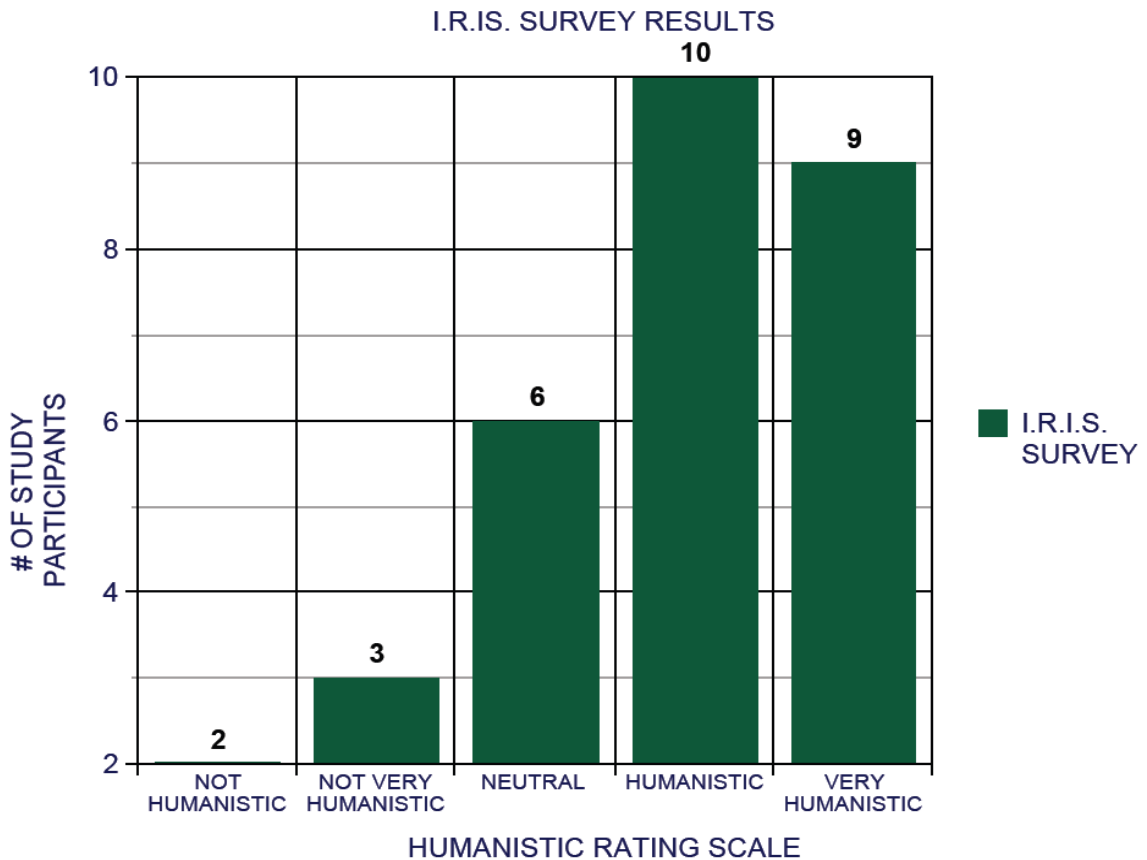


Figure 5.1 Survey Results for Determining Humanistic Level of I.R.I.S.

The rating scale for determining the humanistic level of I.R.I.S. responses is as follows: 1 – Not Humanistic, 2 – Not Very Humanistic/Slightly, 3 – Neutral, 4 Humanistic, and 5 – Very Humanistic. For every rating of four or higher counted as a vote for humanistic. Nine of the thirty study participants voted Very Humanistic and ten of the thirty study participants voted Humanistic accumulating a total of nineteen votes. The final percentage of participants who concluded that the responses I.R.I.S. provided were humanistic totaled 63%.

In summary, we have implemented a chat-bot that operates in a humanistic form by providing users with a range of responses presents the opportunity for enhancing the quality of human computer interaction. We were able to incorporate the two main components to create an Information Retrieval chat-bot. To begin, we created a chat-bot that was able to respond to the user using voice based output and in addition provide a variety of responses therefore ultimately passing the Turing Test. In other words, this study was successful in implementing, analyzing, and evaluating an Information Retrieval Chat-bot by integrating the previously stated main functions and performing the necessary tests and observations to obtain useful results.

5.2 FUTURE WORK

Following the investigations described in this thesis, many facilities can be explored and many enhancements can be added. There are numerous areas where new methods of implementing a chat-bot whilst enhancing HCI can be tried and the general design could be improved upon. Here is a list of some of the improvements to I.R.I.S. that could be

made in the future. I.R.I.S. could be further expanded by incorporating a voice activation feature into the software. Voice Recognition is an interface methodology wherein a system recognizes spoken language. The machine or software take the spoken language as input and translates it into written text depending on the function or command given it is to execute. This feature could be very beneficial and user friendly for the users who are physically unable to operate a computer and use their hands.

Secondly, a broader range of responses could be added to the chat-bot's textual database to advance the way in which it responds. This enhancement could be feasible to someone who wanted to create a chat-bot for an entire University versus a separate entity such as a department. In addition to the previously stated, a virtual TTS character with text-to-speech can be assimilated to improve upon human-computer interaction. Virtual characters can converse back with the users, answering questions in a human-like voice. The voice can be adjusted to various accents and expressions, and even company specific terminology, products and services. These are just a few of the many improvements that can be expanded upon to enhance human computer interaction.

REFERENCES

REFERENCES

1. Abu, Shawar. CHAT-BOTS: Can They Serve as Natural Language Interfaces to QA Corpus? Proceedings of the sixth IASTED international conference on Advances in Computer Science and Engineering. 2010.
2. Billings, Lee. "Rise of Roboethics". SEED. (2007).
3. Ekbia, Hamid. 2008. Artificial dreams: The quest for non-biological intelligence. 1st ed. Cambridge, UK: Cambridge University Press.
4. Güzeldere, Güven. "Dialogues with colorful personalities of early ai". SEHR, Vol. 4 no. 2, (1995).
5. Hofstadter, Douglas R. 1996. "Preface 4 The Ineradicable Eliza Effect and Its Dangers, Epilogue". Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought. Basic Books. p. 157.
6. King, David. 1995. Expert Systems: Artificial Intelligence in Business. Cambridge, UK: John Wiley and Sons Inc.
7. Kurzweil, Ray. 2005. The Singularity is Near When Humans Transcend Biology. London, England : Penguin.
8. Mauldin, L. Michael. ChatterBots, TinyMuds, and the Turing test: entering the Loebner Prize competition. Proceedings of the twelfth national conference on Artificial intelligence, Menlo Park, California, 1994. 16 – 21.

9. McCarthy, John. "WHAT IS ARTIFICIAL INTELLIGENCE?" Stanford University. (2007).
10. McCorduck, Pamela. 2004. *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence*. 2nd ed. NY, New York. W H Freeman & Co.
11. R. Moore and G. Gibbs, "Emile: Using a chatbot conversation to enhance the learning of social theory," Univ. of Huddersfield, Huddersfield, England (2002).
12. Rouse, William B.; Boff, Kenneth R. 2005. *Organizational Simulation*. Wiley-IEEE. pp. 308–309.
13. Russell, Stuart and Peter Norvig. 2003. *Artificial Intelligence: A Modern Approach*. 3rd ed. Saddle River, New Jersey. Prentice Hall.
14. Skillings, Jonathan. "Getting Machines to Think Like Us. (2006).
15. Suchman, Lucy A. 1987. *Plans and Situated Actions: The problem of human-machine communication*. Cambridge University Press. pp. 24.
16. Trappl, Robert; Petta, Paolo; Payr, Sabine. 2002. *Emotions in Humans and Artifacts*. Cambridge, Mass.: MIT Press. pp. 353.
17. Turing, Alan, "Computing machinery and intelligence." *Mind*, Vol. 1 no. 50, 433–460 (1950).
18. Weizenbaum, Joseph, "ELIZA--A Computer Program for the Study of Natural Language Communication between Man and Machine." *Comm. ACM*, Vol. 9 no. 1, 35–36 (1966).

APPENDIX A

Definition of Terms

- 1) Chat-bot/chatterbot – a computer program designed to simulate an intelligent conversation with one or more human users via auditory or textual methods, primarily for engaging in small talk.
- 2) NLP (Natural language processing) – a field of computer science and linguistics concerned with the interactions between computers and human (natural) languages.
- 3) AI (Artificial intelligence) – the intelligence of machines and the branch of computer science that aims to create it.
- 4) HCI (Human Computer Interaction) – the study, planning and design of the interaction between people (users) and computers.
- 5) AIML (Artificial Intelligence Markup Language) – is an XML dialect for creating natural language software agents.
- 6) VLE (Virtual Learning Environment) – a system designed to support teaching and learning in an educational setting.
- 7) MLE (Managed Learning Environment) – a system designed to support management.

APPENDIX B

The following is the final program for the chat-bot I created. This program can be copied into a VBScript editor and modified accordingly. When copied into an editor the files should be saved with .vbs extension. Once saved, program will be able to run as soon as the file is opened by the user. The known browser compatible with VBScript is internet explorer.

```
dim fname
dim welcomeinit
Dim user

dim goodbyestatemnts(5)
    goodbyestatemnts(0)="Is there anything else I can assist you with?"
    goodbyestatemnts(1)="Can I assist you with locating another faculty
or staff member?"
    goodbyestatemnts(2)="Do you need to access information on another
faculty member?"
    goodbyestatemnts(3)="Do you need to assistance locating another
faculty member?"
    goodbyestatemnts(4)="Can I help you access information on another
member of the department?"
    goodbyestatemnts(5)="Do you need help locating another member of the
department?"

dim facultysearchstatements(6)
    facultysearchstatements(0)="Please type the name of the professor you
wish to locate."
    facultysearchstatements(1)="Please type the name of the faculty or
staff member you want to locate."
    facultysearchstatements(2)="Please type the name of the professor you
want to locate."
    facultysearchstatements(3)="Okay you go ahead and enter the name of
the professor you want to locate."
    facultysearchstatements(4)="When you are ready you can enter the name
of the professor you want to locate below."
    facultysearchstatements(5)="Okay give me the name of the professor
you are trying to find."
    facultysearchstatements(6)="Okay, please enter the name of the
professor you want to locate."

dim thanksgbyestatemnts(6)
    thanksgbyestatemnts(0)="Im glad I could assist you " & user & " .
```



```

Goodbye."
    thanksgbyestatemnts(1)="Thank you " & user & " and goodbye."
    thanksgbyestatemnts(2)="I'm Happy I could assist you " & user & ".
Goodbye now."
    thanksgbyestatemnts(3)="Please come back " & user & "if you need
further assistance. Goodbye."
    thanksgbyestatemnts(4)="If you need further assistance " & user & "
relaunch IRIS. Goodbye now."
    thanksgbyestatemnts(5)="Iris will now exit. Goodbye."
    thanksgbyestatemnts(6)="No further assistance is needed. IRIS will
now exit."

user=InputBox("Please Enter your name: ")

dim welcomestatemnts(5)
welcomestatemnts(0)="Hello and Welcome " & user & ". I am IRIS. Do
you need assistance with locating one of the faculty members in the
department?"
welcomestatemnts(1)="Hi and Welcome " & user & ". My name is IRIS. Do
you need assistance with locating a faculty member in the department?"
welcomestatemnts(2)="Welcome " & user & ". My name is IRIS, Can I
assist you with locating a faculty member in the department?"
welcomestatemnts(3)="Greetings " & user & ". I am IRIS, Can I help
you with locating a member of the department?"
welcomestatemnts(4)="How are you today " & user & "? I am IRIS, Do
you need help locating a member of the department?"
welcomestatemnts(5)="Hi" & user & ". My name is IRIS, Can I assist
you with locating a faculty member in the department?"

Randomize
strTextstart = (welcomestatemnts(rnd))
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strTextstart
Welcome ()

Function Welcome ()

Set objRegExpyes1 = New RegExp
objRegExpyes1.IgnoreCase = True
objRegExpyes1.Pattern = "ye|yes|yep|yeah|yup|uh huh|I do|yea"

Set objRegExpno1 = New RegExp
objRegExpno1.IgnoreCase = True
objRegExpno1.Pattern = "NO|NOPE|NADA|I DON'T|NEGATIVE"

welcomeinit=inputbox(strTextstart)

If objRegExpyes1.Test(welcomeinit) Then
EnterProfessorname ()

```

```

ElseIf objRegExpno1.Test(welcomeinit) Then
Goodbye ()

ElseIf welcomeinit=vbNullString Then
wscript.quit(1)

Else
Yesno ()

End If

End Function

Public Function EnterProfessorname ()
Dim facultysearchinput

Set objRegExpkhos = New RegExp
objRegExpkhos.IgnoreCase = True
objRegExpkhos.Pattern = "kh|kho|khosravi|dr khosravi|dr khos|dr
kh|dr khosravi|brahim|dr ebrahim|dr kos"

Set objRegExpallen = New RegExp
objRegExpallen.IgnoreCase = True
objRegExpallen.Pattern = "allen|alvin|mr alvin allen|mr
allen|alvin allen"

Set objRegExpbai = New RegExp
objRegExpbai.IgnoreCase = True
objRegExpbai.Pattern = "bai|dr ba|dr shuju bai|dr
shuju|shuju|shuju bai|dr bai|dr bi|bi"

Set objRegExpfinley = New RegExp
objRegExpfinley.IgnoreCase = True
objRegExpfinley.Pattern = "finley|rachel|dr rachel finley|dr
finley|dr rachel|rachel finley"

Set objRegExphawkins = New RegExp
objRegExphawkins.IgnoreCase = True
objRegExphawkins.Pattern = "keisha|hawkins|ms hawkins|ms keisha
hawkins|keisha hawkins"

Set objRegExpjohnson = New RegExp
objRegExpjohnson.IgnoreCase = True
objRegExpjohnson.Pattern = "johnson|mr johnson|mr alonzo
johnson|alonzo|alonzo johnson"

Set objRegExpkandara = New RegExp
objRegExpkandara.IgnoreCase = True
objRegExpkandara.Pattern = "kandara|dr kandara|dr osman
kandara|osman kandara|dr ka"

```

```

Set objRegExpkourouma = New RegExp
objRegExpkourouma.IgnoreCase = True
objRegExpkourouma.Pattern = "kourouma|dr kourouma|dr mathieu
kourouma|mathieu kourouma"

Set objRegExpmarilyn = New RegExp
objRegExpmarilyn.IgnoreCase = True
objRegExpmarilyn.Pattern = "marilyn|antione|mrs marilyn
antione|mrs antione|marilyn antione"

Set objRegExpmoreman = New RegExp
objRegExpmoreman.IgnoreCase = True
objRegExpmoreman.Pattern = "moreman|dougous|dr moreman|dr
douglas moreman|douglas moreman"

Set objRegExpratana = New RegExp
objRegExpratana.IgnoreCase = True
objRegExpratana.Pattern = "gwee|gw|gwe|dr gwee|dr
nigel|nigel|nigel gwee"

Set objRegExprouquemore = New RegExp
objRegExprouquemore.IgnoreCase = True
objRegExprouquemore.Pattern = "roquemore|ms roquemore|ms deanna
roquemore|deanna roquemore|deanna"

Set objRegExpsalam = New RegExp
objRegExpsalam.IgnoreCase = True
objRegExpsalam.Pattern = "salam|dr salam|dr abdu salam|dr sal|dr
sa|abdu|abdu salam"

Set objRegExptrivedi = New RegExp
objRegExptrivedi.IgnoreCase = True
objRegExptrivedi.Pattern = "trivedi|trivedi|dr sudhir trivedi|dr
trivedi|dr trivedi|sudhir|sudhir trivedi"

Set objRegExpyang = New RegExp
objRegExpyang.IgnoreCase = True
objRegExpyang.Pattern = "yang|dr shizhong yang|dr yang|dr
shizhong|shizhong|shizhong yang"

Set objRegExpgge = New RegExp
objRegExpgge.IgnoreCase = True
objRegExpgge.Pattern = "gwee|gw|gwe|dr gwee|dr nigel|nigel|nigel
gwee|gee"

Randomize
strText = (facultysearchstatements(rnd))
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
facultysearchinput=inputbox(strText)

If objRegExpkhos.Test(facultysearchinput) Then

```

```

        Displaykhos ()
    ElseIf objRegExpgwee.Test (facultysearchinput) Then
        Displaygwee ()
    ElseIf objRegExpbai.Test (facultysearchinput) Then
        Displaybai ()
    ElseIf objRegExpfinley.Test (facultysearchinput) Then
        Displayfinley ()
    ElseIf objRegExphawkins.Test (facultysearchinput) Then
        Displayhawkins ()
    ElseIf objRegExpkandara.Test (facultysearchinput) Then
        Displaykandara ()
    ElseIf objRegExpjohnson.Test (facultysearchinput) Then
        Displayjohnson ()
    ElseIf objRegExpkourouma.Test (facultysearchinput) Then
        Displaykourouma ()
    ElseIf objRegExpmarilyn.Test (facultysearchinput) Then
        Displaymarilyn ()
    ElseIf objRegExpmoreman.Test (facultysearchinput) Then
        Displaymoreman ()
    ElseIf objRegExppratana.Test (facultysearchinput) Then
        Displaypratana ()
    ElseIf objRegExprouquemore.Test (facultysearchinput) Then
        Displayrouquemore ()
    ElseIf objRegExpssalam.Test (facultysearchinput) Then
        Displayssalam ()
    ElseIf objRegExptrivedi.Test (facultysearchinput) Then
        Displaytrivedi ()
    ElseIf objRegExpyang.Test (facultysearchinput) Then
        Displayyang ()
    ElseIf objRegExpallen.Test (facultysearchinput) Then
        Displayallen ()
    ElseIf facultysearchinput= vbNullString Then
        wscript.quit (1)
    Else

```

```

        Nofacultymatch()

    End If

End Function

Function Yesno()

    strText = ("Im sorry but I do not understand your entry.
Please respond with either YES or NO")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    Welcome()

End Function

Function Nofacultymatch()
    strText = ("Im sorry but the name you entered does not match
anyone in the department. Please Check your spelling and retry your
entry")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    EnterProfessorname()

End Function

Function Displaykhos()
    strText = ("The contact information for Dr. Khosravi, chairman of the
Computer Science department is as follows")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    MsgBox "Dr. Ebrahim Khosravi Professor Chairman" & vbCrLf & "Email:
khosravi@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 106"&
vbCrLf & "Phone: 225-771-2060",1, "CONTACT INFORMATION"
    Goodbye()

End Function

Function Displayallen()
    strText = ("The contact information for Mr. Allen, Assistant Professor
of the Computer Science department is as follows")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    MsgBox "Alvin Allen Assistant Professor" & vbCrLf & "Email:
allena@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 111F"& vbCrLf
& "Phone: 225-771-2454",1, "CONTACT INFORMATION"
    Goodbye()

```

End Function

```
Function Displaybai()  
    strText = ("The contact information for Dr. Bi, Associate Professor of  
the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Dr. Shuju Bai Associate Professor" & vbCrLf & "Email:  
bais@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 135"& vbCrLf &  
"Phone: 225-771-3021",1, "CONTACT INFORMATION"  
    Goodbye()  
End Function
```

End Function

```
Function Displaymarilyn()  
    strText = ("The contact information for Mrs. Antoine, Instructor of  
the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Marilyn Antoine Instructor" & vbCrLf & "Email:  
graym@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 111D"& vbCrLf  
& "Phone: 225-771-3223",1, "CONTACT INFORMATION"  
    Goodbye()  
End Function
```

End Function

```
Function Displayjohnson()  
    strText = ("The contact information for Mr. Johnson, Assistant  
Professor of the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Mr. Alonzo Johnson Assistant Professor" & vbCrLf & "Email:  
johnsona@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 112"&  
vbCrLf & "Phone: 225-771-4570",1, "CONTACT INFORMATION"  
    Goodbye()  
End Function
```

End Function

```
Function Displaykandara()  
    strText = ("The contact information for Dr. Kandara, Assistant  
Professor of the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Dr. Osman Kandara Assistant Professor" & vbCrLf & "Email:
```

```
kandara@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 142"& vbCrLf
& "Phone: 225-771-3556",1, "CONTACT INFORMATION"
    Goodbye ()
```

```
End Function
```

```
Function Displaykourouma()
    strText = ("The contact information for Dr. Kourouma, Assistant
Professor of the Computer Science department is as follows")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    MsgBox "Dr. Mathieu Kourouma Assistant Professor" & vbCrLf & "Email:
mkkourouma@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 138"&
vbCrLf & "Phone: 225-771-3652",1, "CONTACT INFORMATION"
    Goodbye ()
```

```
End Function
```

```
Function Displaymoreman()
    strText = ("The contact information for Dr. Moreman, Assistant
Professor of the Computer Science department is as follows")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    MsgBox "Dr. Douglas Moreman Assistant Professor" & vbCrLf & "Email:
moremand@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 137"&
vbCrLf & "Phone: 225-771-4382",1, "CONTACT INFORMATION"
    Goodbye ()
```

```
End Function
```

```
Function Displayroquemore()
    strText = ("The contact information for Mrs. Roquemore, Assistant
Professor of the Computer Science department is as follows")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    MsgBox "Mrs. Deanna Roquemore Assistant Professor" & vbCrLf & "Email:
dearoque@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 143"&
vbCrLf & "Phone: 225-771-3013",1, "CONTACT INFORMATION"
    Goodbye ()
```

```
End Function
```

```

Function Displaysalam()
    strText = ("The contact information for Dr. Salum, Associate Professor
of the Computer Science department is as follows")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
    MsgBox "Dr. Mohammad Abdus Salam Associate Professor" & vbCrLf &
"Email: md_salam@subr.edu" & vbCrLf & "Office: T. T. Allain Hall 136"&
vbCrLf & "Phone: 225-771-4383",1, "CONTACT INFORMATION"
    Goodbye ()

```

End Function

```

Function Displaygwee()
    strText = ("The contact information for Dr. Gwee, Associate Professor
of the Computer Science department is as follows")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
    MsgBox "Dr. Nigel Gwee Associate Professor" & vbCrLf & "Email:
nigel@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 134"& vbCrLf &
"Phone: 225-771-3431",1, "CONTACT INFORMATION"
    Goodbye ()

```

End Function

```

Function Displayfinley()
    strText = ("The contact information for Dr. Finley, Assistant
Professor of the Computer Science department is as follows")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
    MsgBox "Dr. Rachel Vincent-Finley Assistant Professor" & vbCrLf &
"Email: revf@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 139"&
vbCrLf & "Phone: 225-771-4484",1, "CONTACT INFORMATION"
    Goodbye ()

```

End Function

```

Function Displaytrivedi()
    strText = ("The contact information for Dr. Trivedi, Professor of the
Computer Science department is as follows")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
    MsgBox "Dr. Sudhir Trivedi Professor" & vbCrLf & "Email:
trivedi@cmps.subr.edu" & vbCrLf & "Office: T. T. Allain Hall 144"& vbCrLf

```



```
& "Phone: 225-771-4385",1, "CONTACT INFORMATION"  
    Goodbye ()
```

```
End Function
```

```
Function Displayyang()
```

```
    strText = ("The contact information for Dr. Yang, Computational  
Scientist of the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Dr. Shizhong Yang" & vbCrLf & "Email: yangsz@cmps.subr.edu" &  
vbCrLf & "Office: T. T. Allain Hall 140"& vbCrLf & "Phone: 225-771-  
5364",1, "CONTACT INFORMATION"  
    Goodbye ()
```

```
End Function
```

```
Function Displayratana()
```

```
    strText = ("The contact information for Ms. Warren, Laboratory  
Technician of the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Ms. Ratana Warren" & vbCrLf & "Email: ratana@cmps.subr.edu" &  
vbCrLf & "Office: T.T. Allain Hall 121B"& vbCrLf & "Phone: 225-771-  
4675",1, "CONTACT INFORMATION"  
    Goodbye ()
```

```
End Function
```

```
Function Displayhawkins()
```

```
    strText = ("The contact information for Ms. Hawkins, Resource  
Coordinator of the Computer Science department is as follows")  
    Set objvoice = CreateObject("SAPI.SpVoice")  
    ObjVoice.Speak strText  
    MsgBox "Ms. Kyesha Hawkins" & vbCrLf & "Email: kyesha@cmps.subr.edu" &  
vbCrLf & "Office: T. T. Allain Hall 140"& vbCrLf & "Phone: 225-771-  
3553",1, "CONTACT INFORMATION"  
    Goodbye ()
```

```
End Function
```

```
Function Goodbye()
```

```
    dim goodbyeinputbox
```

```

Randomize
strText = (goodbyestatements (rnd))
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
goodbyeinputbox=inputbox(strText)

Set objRegExpgoodbyeeyes = New RegExp
objRegExpgoodbyeeyes.IgnoreCase = True
objRegExpgoodbyeeyes.Pattern = "ye|yes|yep|yeah|yup|uh huh|I do|yea"

Set objRegExpgoodbyeno = New RegExp
objRegExpgoodbyeno.IgnoreCase = True
objRegExpgoodbyeno.Pattern = "NO|NOPE|NADA|I DON'T|NEGATIVE"

If objRegExpgoodbyeeyes.Test(goodbyeinputbox) Then
    EnterProfessorname()

ElseIf objRegExpgoodbyeno.Test(goodbyeinputbox) Then
    PromptforHelp()

ElseIf goodbyeinputbox= vbNullString Then
    wscript.quit(1)

Else
    strText = ("Im sorry but I do not understand your entry. Please
respond with enter YES or NO")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    Goodbye()

End If

End Function

Function PromptforHelp()

dim helppromptinput

Set objRegExpyes = New RegExp
objRegExpyes.IgnoreCase = True
objRegExpyes.Pattern = "yes|yea|yep|yeah|yup|uh huh|I do"

Set objRegExpno = New RegExp
objRegExpno.IgnoreCase = True
objRegExpno.Pattern = "NO|NOPE|NADA|I DON'T|NEGATIVE|DO NOT"

Randomize
strText = ("Do you need help in a particular subject in this

```

```

department?")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    helppromptinput=inputbox("Do you need help in a particular subject in
this department?")

    If objRegExpyes.Test(helppromptinput) Then
        HelpInResearch()

    ElseIf objRegExpNO.Test(helppromptinput) Then
        SubjectExit()

    ElseIf helppromptinput=vbNullString Then
        wscript.quit(1)

    Else
        Yesnohelp()

End If

End Function

Function HelpInResearch()

Dim subjectinput

Set objRegExpDDN = New RegExp
objRegExpDDN.IgnoreCase = True
objRegExpDDN.Pattern = "Digital Data Networks|Networks|Digit|DDN|DATA
NETWORKS"

Set objRegExpDATAS = New RegExp
objRegExpDATAS.IgnoreCase = True
objRegExpDATAS.Pattern = "DATA STRUCTURES|DATA|STURCTURES"

Set objRegExpDBM = New RegExp
objRegExpDBM.IgnoreCase = True
objRegExpDBM.Pattern = "DATAB|DATABASE MANAGEMENT|DBM|DATABASE"

Set objRegExpCL = New RegExp
objRegExpCL.IgnoreCase = True
objRegExpCL.Pattern = "COMPUTER LITERACY|COMPUTER LIT|LITERACY|CL"

Set objRegExpPL = New RegExp

```

```

objRegExpPL.IgnoreCase = True
objRegExpPL.Pattern = "PROGRAMMING L|PROGRAMMING LANGUAGES|LANGUAGES|PL"

Set objRegExpSE = New RegExp
objRegExpSE.IgnoreCase = True
objRegExpSE.Pattern = "SOFTWARE ENGINEERING|SOFTWARE|ENGINEERING|SE"

Set objRegExpCO = New RegExp
objRegExpCO.IgnoreCase = True
objRegExpCO.Pattern = "COMPUTER ORGANIZATION|COMPUTER O|ORGANIZATION|CO"

Set objRegExpIS = New RegExp
objRegExpIS.IgnoreCase = True
objRegExpIS.Pattern = "INFOR|INFORMATION SYSTEMS|INFORMATION|IS"

Set objRegExpNM = New RegExp
objRegExpNM.IgnoreCase = True
objRegExpNM.Pattern = "NUM|NUMERICAL METHODS|NUMERICAL|METHODS|NM"

Set objRegExpDISCRETES = New RegExp
objRegExpDISCRETES.IgnoreCase = True
objRegExpDISCRETES.Pattern = "DISC|DISCRETE STURCTURES|DISCRETE"

Set objRegExpOS = New RegExp
objRegExpOS.IgnoreCase = True
objRegExpOS.Pattern = "OPER|OPERATING SYSTEMS|OPERATING|OS"

Set objRegExpCA = New RegExp
objRegExpCA.IgnoreCase = True
objRegExpCA.Pattern = "Digital Data Networks|Networks|Digital"

Set objRegExpDP = New RegExp
objRegExpDP.IgnoreCase = True
objRegExpDP.Pattern = "DISTRIBUTED PROCESSING|PROCESSING|DiSTributed"

Set objRegExpPROGRAMM = New RegExp
objRegExpPROGRAMM.IgnoreCase = True
objRegExpPROGRAMM.Pattern = "PROGRAMMING|ALGORITHMS|C++|C PLUS PLUS"

Set objRegExpTHEORY = New RegExp
objRegExpTHEORY.IgnoreCase = True
objRegExpTHEORY.Pattern = "THEO|THEORY|THEORY OF COMPUTING"

Set objRegExpSEMINAR= New RegExp
objRegExpSEMINAR.IgnoreCase = True
objRegExpSEMINAR.Pattern = "COMPUTER SCIENCE|SEMINAR|COMPUTER
SEMINAR|CCS"

dim helpstatements(3)
helpstatements(0)="The following professors should be able to assist you
in that area."
helpstatements(1)="These are some of the professors qualified to assist

```

```

you."
helpstatements(2)="The professors listed are knowledgable in that
particular subject."
helpstatements(3)="Listed are some of the professors qualified to provide
you with help."

        strText = ("Please enter the name of the subject you need help
in.")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        subjectinput=inputbox("Please enter the name of the subject you
need help in.")

        If objRegExpDDN.Test(subjectinput) Then
            Randomize
            strText = ("Dr. Khosravi and Dr. Salummm should be able to
assist you in this subject area")
            Set objvoice = CreateObject("SAPI.SpVoice")
            ObjVoice.Speak strText
            MsgBox "Dr. Ebrahim Khosravi Professor Chairman: Office
106" & vbCrLf & "Dr. Salam Associate Professor: Office 136",1, "SUBJECT:
DIGITAL DATA NETWORKS"
            SubjectGoodbye ()

            ElseIf objRegExpDATAS.Test(subjectinput) Then
                Randomize
                strText = ("Mr. Johnson and Mr. Allen are knowledgable in
this particular subject")
                Set objvoice = CreateObject("SAPI.SpVoice")
                ObjVoice.Speak strText
                MsgBox "Mr. Johnson, Assistant Professor: Office 112" &
vbCrLf & "Mr. Allen Assistant Professor: Office 111F",1, "SUBJECT: DATA
STRUCTURE"
                SubjectGoodbye ()

            ElseIf objRegExpDBM.Test(subjectinput) Then
                Randomize
                strText = ("Dr. Bi And Dr. Kandara are qualified To assist
you in this particular area")
                Set objvoice = CreateObject("SAPI.SpVoice")
                ObjVoice.Speak strText
                MsgBox "Dr. Bai, Associate Professor: Office 135" & vbCrLf &
"Dr. Kandara Assistant Professor: Office 142",1, "SUBJECT: DATABASE
MANAGEMENT"
                SubjectGoodbye ()

            ElseIf objRegExpCL.Test(subjectinput) Then
                Randomize
                strText = ("Mrs. Antoine, Dr. Kandara, and Dr. Moreman would
be able to offer help in this subject")
                Set objvoice = CreateObject("SAPI.SpVoice")
                ObjVoice.Speak strText

```

```

Msgbox "Mrs. Antoine Instructor: Office 111D" & vbCrLf &
"Dr. Kandara Assistant Professor: Office 142" & vbCrLf & "Dr. Moreman
Assistant Professor: Office 137",1, "SUBJECT: COMPUTER LITERACY"
SubjectGoodbye()
ElseIf objRegExpPL.Test(subjectinput) Then
Randomize
strText = ("Dr. Bi, Dr. Gwee, and Dr. Kourouma are qualified
to provide you with help in this area")
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
Msgbox "Dr. Bai, Associate Professor: Office 135" & vbCrLf &
"Dr. Gwee Associate Professor: Office 134" & vbCrLf & "Dr. Kourouma
Assistant Professor: Office 138",1, "SUBJECT: PROGRAMMING LANGUAGES"
SubjectGoodbye()
ElseIf objRegExpSE.Test(subjectinput) Then
Randomize
strText = ("Dr. Gwee can offer you assistance in this
subject area")
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
Msgbox "Dr. Gwee Associate Professor: Office
134",1,"SUBJECT: SOFTWARE ENGINEERING"
SubjectGoodbye()
ElseIf objRegExpCO.Test(subjectinput) Then
Randomize
strText = ("Mr Johnson should be able to offer you help")
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
Msgbox "Mr. Johnson, Assistant Professor: Office 112",1,
"SUBJECT: COMPUTER ORGANIZATION"

ElseIf objRegExpIS.Test(subjectinput) Then
Randomize
strText = ("Mr. Johnson is qualified to help you in this
particular subject")
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
Msgbox "Mr. Johnson, Assistant Professor: Office 112",1,
"SUBJECT: INFORMATION SYSTEMS"
SubjectGoodbye()
ElseIf objRegExpNM.Test(subjectinput) Then
Randomize
strText = ("Dr. Moreman should be able to provide you with
assistance in this area")
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
Msgbox "Dr. Moreman Assistant Professor: Office 137",1,
"SUBJECT: NUMERICAL METHODS"
SubjectGoodbye()
ElseIf objRegExpDISCRETES.Test(subjectinput) Then
Randomize
strText = ("Dr. Moreman And Dr. Finley should be able assist

```

```

you in that subject")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    MsgBox "Dr. Moreman Assistant Professor: Office 137" &
vbCrLf & "Dr. Finley Assistant Professor: Office 139",1, "SUBJECT:
DISCRETE STRUCTURES"

    ElseIf objRegExpOS.Test(subjectinput) Then
        Randomize
        strText = ("Dr. Salummm, Dr. Kandara, and Dr. Kourouma
should are very knowlegable in this subject")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        MsgBox "Dr. Salam Associate Professor: Office 136" & vbCrLf
& "Dr. Kandara Assistant Professor: Office 142" & vbCrLf & "Dr. Kourouma
Assistant Professor: Office 138",1, "SUBJECT: OPERATING SYSTEMS"
        SubjectGoodbye ()
    ElseIf objRegExpCA.Test(subjectinput) Then
        Randomize
        strText = ("Dr. Salummm can help you in this subject area")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        MsgBox "Dr. Salam Associate Professor: Office 136",1,
"SUBJECT: COMPUTER ARCHITECTURE"

    ElseIf objRegExpDP.Test(subjectinput) Then
        Randomize
        strText = ("Dr. Trivedi can be helpful in this particular
subject")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        MsgBox "Dr. Trivedi Associate Professor: Office 144",1,
"SUBJECT: DISTRIBUTED PROCESSING"
        SubjectGoodbye ()

    ElseIf objRegExpPROGRAMM.Test(subjectinput) Then
        Randomize
        strText = ("Mr. Johnson, Mr. Allen, and Mrs Roquemore are
qualified to help you in this subject")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        MsgBox "Mr. Johnson, Assistant Professor: Office 112" &
vbCrLf & "Mrs. Roquemore Assistant Professor: Office 143" & vbCrLf & "Mr.
Allen Assistant Professor: Office 111F",1, "SUBJECT: PROGRAMMING I & II"
        SubjectGoodbye ()

    ElseIf objRegExpTHEORY.Test(subjectinput) Then
        Randomize
        strText = ("Dr. GWEE is qualified to help you in this
subject")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText

```

```

        MsgBox "Dr. Gwee Associate Professor: Office 111F",1,
"SUBJECT: THEORY OF COMPUTING"
        SubjectGoodbye()

        ElseIf objRegExpSEMINAR.Test(subjectinput) Then
Randomize
        strText = ("Mr. Johnson is qualified to help you in this
subject")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        MsgBox "Mr. Johnson, Assistant Professor: Office 112",1,
"SUBJECT: COMPUTER SCIENCE SEMINAR"
        SubjectGoodbye()

        Else
        Nosubjectmatch()
End If

End Function
Function Nosubjectmatch()
        strText = ("Im sorry but the subject you entered does not
match anything in the database. Please Check your spelling and retry your
entry")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        HelpInResearch()
End Function

Function Yesnohelp()

        strText = ("Im sorry but I do not understand your entry.
Please enter either YES or NO.")
        Set objvoice = CreateObject("SAPI.SpVoice")
        ObjVoice.Speak strText
        PromptforHelp()
End Function

Function SubjectGoodbye()

        dim subgoodbyestatemnts(2)
        subgoodbyestatemnts(0)="Do you need help in another subject area?"
        subgoodbyestatemnts(1)="Do you need to find help in another
subject?"
        subgoodbyestatemnts(2)="Would you like to find help in another
subject?"

        Set objRegExpsubyes = New RegExp
        objRegExpsubyes.IgnoreCase = True
        objRegExpsubyes.Pattern = "ye|yes|yep|yeah|yup|uh huh|I do|yea"

```



```

Set objRegExpsubno = New RegExp
objRegExpsubno.IgnoreCase = True
objRegExpsubno.Pattern = "NO|NOPE|NADA|I DON'T|NEGATIVE"

dim subgoodbyeinputbox
Randomize
strText = (subgoodbyestatemnts(rnd))
Set objvoice = CreateObject("SAPI.SpVoice")
ObjVoice.Speak strText
subgoodbyeinputbox=inputbox(strText)

If objRegExpsubyes.Test(subgoodbyeinputbox) Then
    HelpInResearch()

ElseIf objRegExpsubno.Test(subgoodbyeinputbox) Then
    SubjectExit()

ElseIf subgoodbyeinputbox=vbNullString Then
    wscript.quit(1)

Else
    strText = ("Im sorry but I do not understand your entry. Please
respond with enter YES or NO")
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    SubjectGoodbye()

End If

End Function

Function SubjectExit()

    dim subexitinputbox
    dim subexitstatements(3)
    subexitstatements(0)="Im glad I could assist you " & user & ".
Goodbye."
    subexitstatements(1)="Thank you " & user & " and goodbye."
    subexitstatements(2)="I'm Happy I could assist you " & user & ".
Goodbye now."
    subexitstatements(3)="Please come back " & user & "if you need further
assistance. Goodbye."

    Randomize
    strText = (subexitstatements(rnd))
    Set objvoice = CreateObject("SAPI.SpVoice")
    ObjVoice.Speak strText
    wscript.quit(1)

```

End Function

APPENDIX C

USER MANUAL

NOTE: SOFTWARE WILL ONLY RUN IN INTERNET EXPLORER

STEP 1: ONCE THE START PAGE IS ACCESSED, USER SHOULD CLICKS ON THE “CLICK TO LAUNCH” BUTTON IN ORDER TO RUN THE SOFTWARE.



STEP 2: ONCE THE LAUNCH BUTTON IS CLICKED, USER WILL BE PROMPTED TO ENTER THEIR NAME.



STEP 3: ONCE THE USER ENTERS THEIR NAME, HE/SHE WILL BE GREETED WITH A MESSAGE & PROMPTED TO ENTER “YES” OR “NO” FOR ASSISTANCE. NOTE: IF USER ENTERS NO THE APPLICATION WILL EXIT.



STEP 4: ONCE THE USER YES, HE/SHE WILL BE PROMPTED TO ENTER THE NAME OF THE PROFESSOR THEY WISH TO LOCATE IN THE DEPARTMENT.

NOTE: IF USER CLICKS “CANCEL” THE APPLICATION WILL EXIT.



STEP 5: ONCE THE USER ENTERS THE NAME OF THE PROFESSOR THEY WISH TO LOCATE IN THE DEPARTMENT, THE CONTACT INFORMATION FOR THAT SPECIFIC FACULTY OR STAFF MEMBER WILL THEN BE DISPLAYED.

The screenshot displays the I.R.I.S. (Intelligent Retrieval Information System) interface. On the left, a vertical banner reads "Department of Computer Science". The main content area features the I.R.I.S. logo, which includes the letters "IRIS" in a stylized font inside a blue circle, with "I.R.I.S." written below it. To the right of the logo is a portrait of a woman. Below the logo and portrait, the text "INTELLIGENT RETRIEVAL INFORMATION SYSTEM" is displayed. At the bottom right, a prominent button says "CLICK TO LAUNCH".

Department such as their office location and contact information

Department of Computer Science

VBScript: CONTACT INFORMATION...

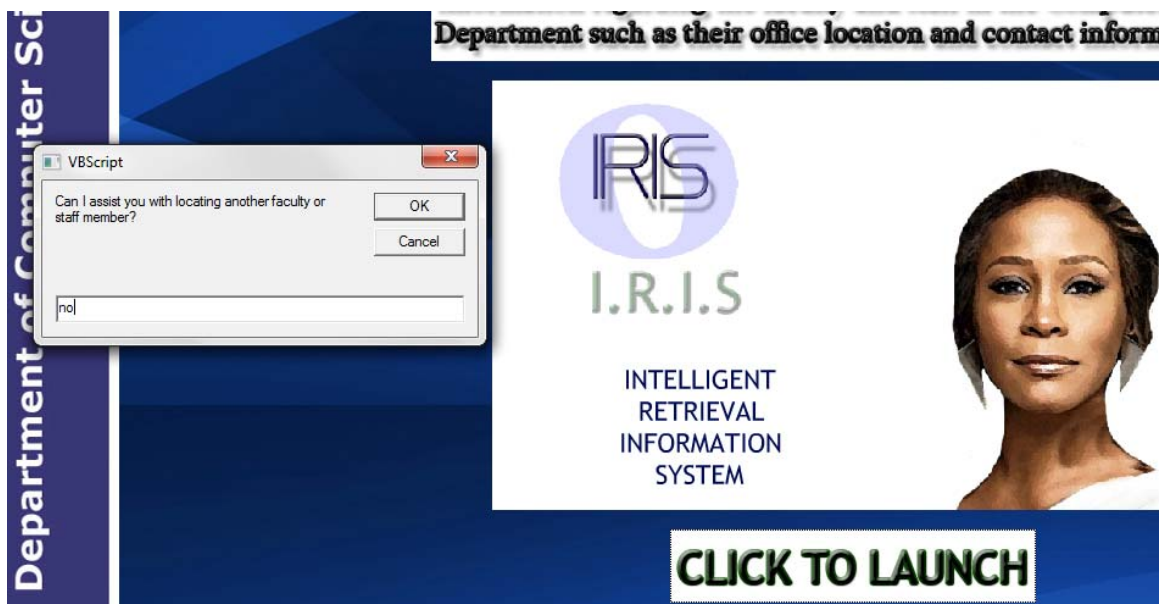
Dr. Nigel Gwee Associate Professor
Email: nigel@cmps.subr.edu
Office: T. T. Allain Hall 134
Phone: 225-771-3431

OK Cancel

IRIS
I.R.I.S.
INTELLIGENT
RETRIEVAL
INFORMATION
SYSTEM

CLICK TO LAUNCH

STEP 6: ONCE THE CONTACT INFORMATION FOR THAT SPECIFIC FACULTY OR STAFF MEMBER IS DISPLAYED, THE USER WILL BE ASKED IF THEY WOULD LIKE TO LOCATE ANOTHER PERSON. IF USER ENTERS “YES”, HE/SHE WILL BE PROMPTED TO ENTER THE NAME OF THE PERSON THEY WANT TO LOCATE.



STEP 7: IF USER ENTERS “NO”, HE/SHE WILL BE ASKED IF THEY NEED HELP IN A SUBJECT IN THE DEPARTMENT.

Department of Computer Science

Department such as their office location and contact inform

VBScript

Do you need help in a particular subject in this department?

OK

Cancel

yes

IRIS

I.R.I.S

INTELLIGENT
RETRIEVAL
INFORMATION
SYSTEM

CLICK TO LAUNCH

STEP 8: IF USER ENTERS “YES”, HE/SHE WILL BE PROMPTED TO ENTER THE NAME OF THE SUBJECT THEY NEED HELP IN.



STEP 9: ONCE USER ENTERS THE NAME OF THE SUBJECT THEY NEED HELP IN, A LIST OF PROFESSORS QUALIFIED TO ASSIST WILL BE DISPLAYED.

Department of Computer Sci

Department such as their office location and contact inform

VBScript: SUBJECT: SOFTWARE ENGINEERI...
Dr. Gwee Associate Professor: Office 134
OK Cancel

IRIS
I.R.I.S
INTELLIGENT
RETRIEVAL
INFORMATION
SYSTEM

CLICK TO LAUNCH

STEP 10: ONCE USER CLICKS “OK” HE/SHE WILL BE ASKED IF THEY NEED HELP IN ANOTHER SUBJECT THEY NEED HELP IN. IF USER ENTERS “YES” THE PROCESS WILL COMPLETE. IF USER ENTERS “NO” APPLICATION WILL EXIT.



VITA

Kadrica Tillotson was born in Baton Rouge, Louisiana on December 19, 1986. She attended elementary schools in the East Baton Rouge Parish and graduated from Istrouma High School & Technology Magnet with honors in May 2004. The following August the author entered Southern University A&M College in Louisiana and in August 2008 received the degree of Bachelor of Science in Computer Science with honors. Re-entering Southern University in January 2010 she is a candidate for the Master of Science Degree in Computer Science with a concentration in database management.

APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Southern University Library the right to reproduce, by appropriate methods, upon request, any or all portions of this project document.

It is understood that “request” consists of the agreement, on the part of the requesting party, that said reproductions is for his personal use and subsequent reproduction will not occur without written approval of the author of this project document.

The author of this project document reserves the right to publish freely, in the literature, at any time, any or all portions of this project document.

Author Kachia Linas
Date July 24, 2012