

B. SC. (HONS) IN COMPUTER SCIENCE

**SEPTEMBER
2010**

**NUMBER
3**

Acknowledgement

With this the author wishes to thank the project supervisors: Mr. Hubert Tan Lai Chai and Ms. Deshinta Arrova Dewi. It was through their guidance, help, and supervision that this project may be completed and this document may be compiled. Credits should next be given to interviewees and other people who participated in the project by lending their voices and opinions; of particular note are Halls of Residence officers, student helpers, and resident fellows who cooperated admirably during the course of the project. Sincere thanks also go to colleagues and classmates who had given help and support, as well as numerous authors of books and web resources consulted to aid project decision making.

Very special thanks are reserved for the author's family, who, over the years, unconditionally offered their support in terms of financial, moral, and other dimensions in life. Even this insignificant feat would not have been possible otherwise.

About the Author



The author was born in the 30th of July, 1989 in Pekanbaru, Indonesia. He completed his primary education in 2001, graduating from SD 003 Pekanbaru, and spent the next two years in junior high school education at SLTP 4 Pekanbaru. He then enrolled in SMA 8 Pekanbaru and graduated from high school three years later in 2006.

Joining INTI International University (then INTI University College) in 2008, the author has always been greatly drawn to the subject of computer studies, which he sees as embodying both the meticulous, systematic nature of mathematics and the open-ended character of the arts. In INTI he enrolled in the 3+0 Honours Degree in Computer Science from Coventry University, currently in his final year.

Executive Summary

Major Findings

Developing this project, the author had acquired new knowledge in artificial intelligence, specifically the methods of simulating intelligent agents without significant natural language processing. This experience also deepened the author's command of Java, as well as algorithmic logic in general.

Aside from technical knowledge, the author had also learned valuable lessons in project management, working with people, eliciting information, and other soft skills demanded by the industry of today.

Project Selling Points

Virtual assistants are particularly interesting nowadays because users seem to be able to accept that they are conversing with an artificial entity with full awareness; they also do not seem to be demanding very high level AI for mundane tasks. It is starting to seem like the bots are fully accepted.

In addition to its quirky, interesting nature, virtual assistants can prove useful in a wide array of circumstances, and even at its worst it would still be considered as a lushly integrated help interface.

This kind of project would be of great interest to business establishments intending to secure a better bond with their customers, or as an aid in teaching young children a topic encompassing a large amount of knowledge. In its current form, it is surely helpful as a repository of knowledge in students residing in the college housing environment used as the case study.

Problem Definition

Problem Statement

Currently the college housing environment at INTI International University (INTI Halls of Residence) do not have an online repository of their regulations, events calendar, and other bulletin materials. Instead, they issue booklets to both many offices and student rooms in addition to placing notices in boards scattered around the campus.

Meanwhile, companies around globe are starting to implement chat bots—weak artificial intelligence modules used to converse with users and answer company-related questions. This way they effectively serve as helpdesks. The goal of the project is to demonstrate the processes of designing, developing, and deploying a chat bot-based virtual assistant when used in the context of a college housing environment, in this case the INTI Halls of Residence.

Scope of Research

It must be said that the focus of the project lies in the chat bot and less in the Halls of Residence aspect. That is to say, the context is merely used as a knowledge base material to be implemented by the chat bot.

The project revolves around programming, and had none to do with plug-ins, APIs, and other integration-related issues. The author had to build the AI from scratch in Java with Eclipse as the IDE. Once the core shell is completed, the system is being given a knowledge base and the package would later be packaged inside an applet tailored to fit the requirements.

Limitations of Research

The project is being carried out by an undergraduate student, and as such it doesn't have a reliable programmer when measured by industrial standards. This was compensated by reading, researching, and exhaustive trial and error, eventually succeeding in meeting completion.

The project is limited in the way that it doesn't cover updating and maintaining the knowledge base afterwards. In fact, the project doesn't cover actual product deployment. It is meant to construct a chat bot, and use a case study to fill in the requirements.

Research Methodology

Knowledge and skills necessary for the completion of the project are being procured by consulting with books, web resources, and lecturers. Books found in the university library were of great help in the project.

Also, ideas are being extracted by observing how similar projects are being constructed, particularly older, simpler chat bots which leave rooms for improvements.

Good and Bad Experiences

The difficulties of the project mainly lie in finding the right method/form to start developing. Eventually it was decided that it will be coded from scratch in Java, but before there were much doubt. Also, there was a need for the author to take leave in the critical time between Phase 4 and the final project submission date, which made a mess with the time management aspect. However, eventually these hardships were overcome.

Good experiences include the opportunity to design a virtual personality and generally work with algorithms while researching the history and development of artificial intelligence, expert systems, and natural language processing.

Conclusion

The project, while surely imperfect, was eventually completed according to plans. During the course of working on it the author derived a good amount of new knowledge and skills, ones which no doubt would be useful in his next projects and work activities. It was a project in which one can learn so much in so little time.

It is hoped that the project, its product, its ideas, or anything it contributes will be proved useful for whoever may eventually derive benefits from it in the future.

Preface

The project intends to create from scratch a chat bot (a soft AI implementation of artificial conversational entity) to serve as a virtual assistant in a college housing environment. The emphasis is in the bot, where the case study is there chiefly to provide context.

Efforts were made to document the processes properly, so that the project may be studied better in cases where knowledge can be derived from it.

Chapter 1: Literature Review

A chapter dedicated to the processes of reviewing and background study in general. Here what the chat bots are about, the history and development of disciplines related to its background, as well as their relevance to the project at hand are elaborated.

Chapter 2: Process Modelling

Projects usually follow a process model to structure its schedules and phases/stages properly. This project is no exception: the chapter contains a short recapitulation on what process models are; an introduction to a number of popular process models; and subsequently tried to deduce which process model would be most suitable given the purposes the project has at hand.

Chapter 3: Requirements Analysis

Eliciting and formulating requirements begins with identifying the stakeholders and then listing out more than a dozen techniques and why some are applicable and some others are not. Those which were deemed applicable were later used to elicit requirements. These resulting requirements are then listed so that they may be accessed easily.

Chapter 4: Systems Analysis

Systems analysis feature several UML diagrams to explain how the system is supposed to work under a number of given scenarios. Provided in this chapter are a use case diagram and a set of activity diagrams. This chapter is quickly followed up by systems design.

Chapter 5: Systems Design

Systems design is partitioned into three activities. The first would be UML diagram constructions, this time featuring the class diagram as well as a number of sequence diagrams. The second would be low-fidelity prototypes design. The third and final one would be to produce the virtual assistant sprites as well as the dummy webpage in which the applet will reside.

Chapter 6: Systems Development

Systems development is perhaps the most important stage in the project, but the documentation version is less busy, as it is only meant to document the processes and final structure and does not necessarily reflect the efforts put

inside the development. Here the classes are elaborated, and the applet is integrated.

Chapter 7: Systems Testing

Where the system is tested using three testing methods: unit testing, integration testing, and system testing. There were a few errors initially but they were corrected and debugged.

Chapter 8: Systems Evaluation

Now that the product is finished, the system will be evaluated. Three methods were utilised: by comparing the finished product with the list of requirements and see if it satisfies, by conducting a very small-scale questionnaire activity to a group of users, and by ensuring the integrity of the information used in the system.

Chapter 9: Concluding Statements

Conclusion and a list of future enhancements, just in case someone might want to improve on the system. Late additions are also covered in this chapter; these include final touches which were not assessed but provided anyway as a means of closure.

TABLE OF CONTENTS	Page
Acknowledgements	i
About the Author	ii
Executive Summary	iii
Major Findings	iii
Project Selling Points	iii
Problem Definition	iv
Problem Statement	iv
Scope of Research	iv
Limitation of Research	iv
Research Methodology	iv
Good and Bad Experiences	v
Conclusion	v
Preface	vi
Chapter 1: Literature Review	1
1.1. Introduction	1
1.2. Automated Helpdesk	1
1.3. Natural Language Processing	2
1.3.1. Natural Language Defined	2
1.3.2. Generational Categorisation of Programming Languages	2
1.3.3. Processing Natural Languages	5
1.3.3.1. Difficulties with NLP: Limitations of Machine Translations	5
1.3.3.2. Chat Bots Defined	7
1.3.3.3. The Turing Test	7
1.3.3.4. Chat Bots: Techniques and Famous Examples	9
1.3.3.5. Specific-Purpose Chat Bots	13
1.3.4. Technical Aspects	14
1.3.5. Relevance to the Proposed Project	16
1.4. Expert Systems	16
1.4.1. Expert Systems Defined	16
1.4.2. History and Development	16
1.4.3. Basic Operations	17
1.4.4. Benefits and Limitations	19
1.4.5. Relevance to the Proposed Project	20
1.5. Defining Project Scope	21
1.6. Chapter Summary	22
Chapter 2: Process Modelling	24
2.1. Introduction	24
2.2. Process Models	24
2.2.1. The Waterfall Model	25
2.2.2. Prototyping	29
2.2.3. The Incremental Model	31
2.2.4. Rapid Application Development	32
2.2.5. The Spiral Model	34
2.3. Choosing a Process Model	36
2.4. Chapter Summary	36
Chapter 3: Requirements Analysis	38
3.1. Stakeholder Identification	38

3.2. Requirements Elicitation	38
3.2.1. Overview	39
3.2.2. Elicitation Techniques Utilised	39
3.2.2.1. Reviewing the Applicability of Different Elicitation Techniques	39
3.2.2.2. Choosing Elicitation Techniques	43
3.2.3. Requirements Elicitation Activities	45
3.2.3.1. Domain Analysis	45
3.2.3.2. Interviewing	50
3.2.3.3. Scenario-Based Analysis	54
3.2.4. List of Requirements	57
3.3. Chapter Summary	58
Chapter 4: Systems Analysis	59
4.1. Introduction	59
4.2. Tools and Techniques	59
4.2.1. Tools	59
4.2.2. Techniques	60
4.3. UML Diagrams for Systems Analysis	61
4.3.1. Use Case Diagram	61
4.3.2. Activity Diagrams	65
4.4. Chapter Summary	74
Chapter 5: Systems Design	75
5.1. Introduction	75
5.2. Tools and Techniques	75
5.2.1. Tools	75
5.2.2. Techniques	76
5.3. Designing the System	77
5.3.1. Sequence Diagrams	77
5.3.2. Class Diagram	81
5.3.3. Low-Fidelity Prototypes	84
5.3.4. Sprites and Dummy Webpage	87
5.4. Chapter Summary	89
Chapter 6: Systems Development	91
6.1. Introduction	91
6.2. Tools and Techniques	91
6.2.1. Tools	91
6.2.2. Techniques	93
6.3. Structure and Packages	94
6.4. Classes	96
6.4.1. Prior Assumptions	96
6.4.2. Knowledge Base	96
6.4.2.1. Canned Text	96
6.4.2.2. IO	98
6.4.2.3. Keyword	99
6.4.2.4. Topic	101
6.4.2.5. Presubs and Presubs List	103
6.4.2.6. Synonyms and Synonyms List	105
6.4.2.7. Catalogue	107
6.4.3. Data Processing	109
6.4.3.1. Archivist	109
6.4.3.2. Collector	111
6.4.3.3. Keyword Stack	113

6.4.3.4. Session	113
6.4.3.5. Processor	114
6.4.3.6. Mood Control	118
6.4.4. User Interface	118
6.5. Chapter Summary	121
Chapter 7: Systems Testing	123
7.1. Introduction	123
7.2. Tools and Techniques	123
7.2.1. Tools	123
7.2.2. Techniques	124
7.3. Testing Activities	124
7.3.1. Unit Testing	124
7.3.2. Integration Testing	138
7.3.3. System Testing	138
7.4. Chapter Summary	139
Chapter 8: Systems Evaluation	140
8.1. Introduction	140
8.2. Evaluation Techniques and Results	140
8.2.1. Requirements Evaluation	140
8.2.2. User Evaluation	142
8.2.3. Knowledge Base Integrity Test	145
8.3. Chapter Summary	145
Chapter 9: Concluding Statements	147
9.1. Future Enhancements	147
9.2. Late Addition: Text-Based Database	147
9.3. Conclusion	148
References	149
List of Tables	
Table 1.1. Corruptions in Machine Translations	6
Table 1.2. Specialised Chat Bots	13
Table 1.3. Benefits and Limitations of Expert Systems	20
Table 1.4. Benefits and Limitations of Choosing a College Housing Environment as the Project's Designated Area of Specialisation	21
Table 2.1. Benefits and Pitfalls of Prototyping	30
Table 3.1. Some Requirements Elicitation Techniques	39
Table 3.2. Assessment on Requirements Elicitation Techniques	43
Table 3.3. Benefits and Pitfalls of Domain Analysis	46
Table 3.4. Virtual Assistants Examined in the Domain Analysis	47
Table 3.5. Benefits and Pitfalls of Interviewing	50
Table 3.6. Benefits and Pitfalls of Scenario-Based Analysis	55
Table 4.1. Actors and Use Cases: Virtual Agent as an Automated Helpdesk in College Housing Environment	62
Table 5.1. List of Classes	83
Table 6.1. The Pros and Cons of Exhaustive IDE Comparison	92
Table 6.2. Functions and Classes of the Three Main Components	95
Table 7.1. Archivist, 1 st Test	124
Table 7.2. Archivist, 2 nd Test	125
Table 7.3. Collector, 1 st Test	126
Table 7.4. Keyword Stack, 1 st Test	126

Table 7.5. Keyword Stack, 2 nd Test	127
Table 7.6. Mood Control, 1 st Test	128
Table 7.7. Session, 1 st Test	128
Table 7.8. Processor, 1 st Test	128
Table 7.9. Processor, 2 nd Test	129
Table 7.10. Processor, 3 rd Test	130
Table 7.11. Processor, 4 th Test	131
Table 7.12. Processor, 5 th Test	133
Table 7.13. Canned Text, 1 st Test	134
Table 7.14. Canned Text, 2 nd Test	135
Table 7.15. IO, 1 st Test	135
Table 7.16. Presubs List, 1 st Test	136
Table 7.17. Synonyms List, 1 st Test	136
Table 7.18. Synonyms List, 2 nd Test	136
Table 7.19. Applet, 1 st Test	137
Table 7.20. Integration Testing	138
Table 7.21. System Testing	138
Table 8.1. Requirements Evaluation	140

List of Figures

Figure 1.1. The Evolution of Programming Languages	3
Figure 1.2. The Standard Interpretation of the Turing Test	8
Figure 1.3. Natural Language Processing Architecture	15
Figure 1.4. Expert Systems Architecture	19
Figure 2.1. The Opportunistic Approach	25
Figure 2.2. The Waterfall Process Model	26
Figure 2.3. Feedback-Enabled Waterfall Process Model	27
Figure 2.4. The V-Model	28
Figure 2.5. Prototype Types: Throwaway and Revolutionary	30
Figure 2.6. The Incremental Model	32
Figure 2.7. An interpretation of the Rapid Application Development Model	33
Figure 2.8. The Spiral Model	35
Figure 4.1. Use Case Diagram: Virtual Agent as an Automated Helpdesk in College Housing Environment	64
Figure 4.2. Activity Diagram: Conversation	67
Figure 4.3. Activity Diagram: FAQs Display	69
Figure 4.4. Activity Diagram: Schedule Display	70
Figure 4.5. Activity Diagram: Map Display	71
Figure 4.6. Activity Diagram: Input Processing	73
Figure 5.1. Sequence Diagram: Conversation	78
Figure 5.2. Sequence Diagram: FAQs Display	80
Figure 5.3. Sequence Diagram: Schedule Display	80
Figure 5.4. Sequence Diagram: Map Display	81
Figure 5.5. Class Diagram: Virtual Agent as an Automated Helpdesk in College Housing Environment	82
Figure 5.6. The Chat Bot Screen	85
Figure 5.7. The FAQs Screen	86
Figure 5.8. The Schedule Screen	86
Figure 5.9. The Map Screen	87
Figure 5.10. Lucy Sprites	88
Figure 5.11. The Dummy Webpage	89
Figure 6.1. Expert Systems: Basic Architecture	94
Figure 6.2. The Chat Bot's Basic Architecture	95
Figure 6.3. Swing UI	119

Figure 6.4. The Lucy Applet	121
Figure 8.1. User Assessment: First Impressions	142
Figure 8.2. System Helpfulness Evaluation	143
Figure 8.3. User Assessment: Information Integrity	144
Figure 8.4. Additional Features Request	144

List of Displays

Display 6.1. Canned Text: Random Message Functions	97
Display 6.2. Canned Text: Sandwiching a Text Inside Another	98
Display 6.3. IO: Sample loadCannedText() Method	99
Display 6.4. Keyword: toString()	100
Display 6.5. Topic: getReply()	102
Display 6.6. Topic: toString()	103
Display 6.7. Presubs Class	104
Display 6.8. PresubsList: substitute()	105
Display 6.9. Synonyms Class	106
Display 6.10. SynonymsList: matches and getSynonyms Method	107
Display 6.11. Catalogue: Instance Variables	107
Display 6.12. Catalogue: Sorting Methods	108
Display 6.13. Catalogue: toString()	109
Display 6.14. Archivist: copy for String[]	110
Display 6.15. Archivist: add for String[]	110
Display 6.16. Archivist: Sorting Keywords	111
Display 6.17. Collector: Tokenising/Detokenising	112
Display 6.18. Collector: Supporting Methods	113
Display 6.19. Session Class	114
Display 6.20. Instance Variables	115
Display 6.21. Processor: process Method	116
Display 6.22. Processor: Supporting Methods	118
Display 6.23. SwingUI: Supporting Methods	120
Display 9.1. Database Sample	148

Appendices

Appendix A: Project Proposal	A1-A4
Appendix B: Gantt Chart	B1-B4
Appendix C: Project Monitoring Reports	C1-C12
Appendix D: Screen Captures	D1-D3
Appendix E: Knowledge Base File	E1-E4
Appendix F: Turnitin Report	F1-F2

Chapter 1: Literature Review

1.1. Introduction

The idea of implementing a form of artificial intelligence entity to offer an automated helpdesk support is, like a great many other ideas in modern computing, an offshoot of certain separate ideas and disciplines. This particular idea (henceforth referred to as simply 'automated helpdesk') was not merely inspired by these older areas, in which case one may divorce the links between them and proceed to treat it as a fundamentally independent implementation: these entities are still instances of those areas and must adhere to whatever principles, definitions, and heuristics applicable to any other instance. It should follow from there that, prior to investigating the current state of affairs regarding automated helpdesks, one is advised to cover the topics of those aforementioned ideas and disciplines, from which automated helpdesks sprung.

It may be easily observed that the most apparent one here would be chat bots; computer programs intended to simulate understanding and engage in conversations with human agents. Itself a more simplified, somewhat less ambitious form of natural language processing, chat bots (otherwise known as chatterbots or talk bots) are by their nature applicable to virtually all kinds of contexts. This approach, however, demands overly exhaustive efforts, and as a response chat bot authors occasionally frame their bots to a narrower scope or scenario. The famed ELIZA, specifically its DOCTOR script, for example, plays a role of a medical doctor, and the relatively more recent GoogleMinotaur bot specialised in the topic of Radiohead.

In this chapter the aim is to study the beginnings of the automated helpdesks, by looking at how the disciplines which brought it to life were growing and ultimately crossing ways to produce such a cross-discipline kind of ICT product.

1.2. Automated Helpdesks

Automated helpdesks are ultimately a form of specialisation, where the AI's focus is narrowed, and the programmers may invest greater time and effort in certain areas of simulated expertise. Using this setup, one may also provide conversational boundaries in a manner which is more acceptable to the common user: rather than vaguely attempting to avoid a given topic because they are not specifically programmed to converse about it, bots may simply refuse to go beyond these boundaries. This, in addition to being practical, may also (quite incidentally) be among the few ways chat bots can be remotely useful for practical affairs. That is to say, rather than being novelty toys or purely academic research outcomes, helpdesks actually have real-life benefits for the general public. Automated helpdesks are also ultimately a type of expert systems, since they are aimed to replace a kind of human expertise (to be precise, the expertise of an employee equipped with in-depth knowledge about a given product, service, or other parallels). A popular example would be PayPal's Sarah, which may be accessed at PayPal's contact page.

At this point the areas from which automated helpdesks sprung are apparent; chat bots (also, natural language processing in general) and expert systems. The initial phases of this discourse will be devoted for covering these topics: their history, characteristics, practices/techniques, weaknesses, and relevance to the project at hand.

This project in particular aims to design and implement automated helpdesk in a college housing environment. The choice is arbitrary; the project seeks to explore and evaluate the practice of implementing automated helpdesks as a form of assistance in commercial environment, and they are applicable to all sorts of products/services. A section would exclusively cover this subject, where the rationale behind the choosing of theme is explored, and where a real-life college housing environment is chosen to be the basis of the whole dissertation.

To better understand how chat bots work, one should be familiar with the basics of the two disciplines: natural language processing and expert systems. Ultimately the chat bot wouldn't actually be an instance of a hard NLP machine nor an expert system, but it uses the principles behind the two. It is hoped that, by understanding these topics, not only will the readers understand chat bots, but they will also be able to see how chat bots can be improved.

1.3. Natural Language Processing

1.3.1. Natural Language Defined

In computing terms, natural language may be defined as a type of query language which allows its users to communicate with it in conversational, spoken, or written language as opposed to computer language. To put it in another, more technical way, it refers to informal (i.e. potentially ambiguous) languages used in everyday speech, as opposed to the unambiguous and formal languages used in programming (Evans et al., 2010; French, 1996; Fuller and Manning, 1997). Examples include any conceivable language used in practical affairs, including English, Spanish, Russian, and Arabic.

1.3.2. Generational Categorisation of Programming Languages

Programming languages has been undergoing a steady evolution ever since their conception, and this progress is often represented in generations, with each generation more removed from machine language than its respective predecessors. Each are often coded with the postfix GL, ranging from 1GL to, as of 2010, 5GL. To approach the subject of natural language properly, below is a review for this process as a short digression.