

‘MyQuestion’ Inquiry System for UTP Students and Staff

by

Malika Seidova

14669

Dissertation submitted in partial fulfilment of

the requirements for the

Bachelor of Technology (Hons)

(Business Information Systems)

MAY 2014

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

‘MyQuestion’ Inquiry System for UTP Students and Staff

by

Malika Seidova

14669

A project dissertation submitted to the
Business Information Systems Programme
University Teknologi PETRONAS
in partial fulfilment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(BUSINESS INFORMATION SYSTEMS)

Approved by,

(Saipunidzam B Mahamad)

UNIVERSITY TEKNOLOGI PETRONAS

TRONOH, PERAK

May 2014

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

MALIKA SEIDOVA

ABSTRACT

Due to the lack of single comprehensive system, both students and staff are not satisfied with the current state of handling inquiries within administrative departments which is time-consuming and human-relied. As students struggle to get timely response to their inquiries in administrative departments, the necessity for automated question answering system becomes more important. There is a need in a system that allows a student to ask a question in everyday language and receive an answer quickly and succinctly, with sufficient context to validate the answer.

Hence, the main objective of the project is to develop system prototype based on the concepts of Automated Email Response and Question-Answering Systems using Iterative Development approach.

The given paper further elaborates the problem statement and scope of the project. In-depth analyses and development requirements have been carried out in order to better facilitate to the progress of building the proposed system.

ACKNOWLEDGEMENTS

At the very outset, I would like to take this opportunity and express my deepest gratitude to my supervisors Mr Saipunidzam B Mahamad and Ms Roselind Wan for their coordination and priceless guidance that they have given me throughout the entire period of the project. I express my heart-felt gratitude to all other lecturers of CIS department who would unhesitatingly assist and advice on necessary aspects of my project whenever I address them.

Next I would like to extend my gratitude to ITMS technicians for all the help they had given me.

It would have been impossible for me to accomplish all these without the blessings and a support of my parents and my friend's direct/indirect help whom I address my faithful thanks.

Last but not least, I am very much thankful to the administration of my beloved Universiti Teknologi PETRONAS for the opportunity to undertake individually and independently Final Year Project that granted a great learning exposure and its practical application.

TABLE OF CONTENTS

CERTIFICATION OF APPROVAL.		i
CERTIFICATION OF ORIGINALITY.		ii
ABSTRACT.		iii
ACKNOWLEDGEMENTS.		iv
CHAPTER 1: INTRODUCTION.		1
1.1. Background of Study.		1
1.2. Problem Statement		2
1.3. Project Objectives.		3
1.4. Project Scope.		4
1.5. Project Relevancy and Feasibility.		5
CHAPTER 2: LITERATURE REVIEW.		6
2.1. Related and Similar Works Done by Others.		6
2.2. Overview on Text Mining.		13
CHAPTER 3: METHODOLOGY.		17
3.1. System Development Approach.		17
3.2. Iteration 1 – Analysis phase.		17
3.3. Iteration 2 – Design phase.		19
3.3.1. Flow-chart.		19
3.3.2. System architecture.		20
CHAPTER 4: RESULTS AND DISCUSSION.		21
4.1. Iteration 2 – Analysis phase.		21
4.2. Development Requirements.		21
4.2.1. Tools used.		21
4.2.2. Network and Communication Architecture of the System.		23
4.3. Iteration 2 – Develop.		25
4.4. Iteration 3 – Test.		27
CHAPTER 5: CONCLUSION AND RECOMMENDATION.		29
REFERENCES.		30

APPENDICES.	33
Appendix 1: Queue in front of SSSD counter.	33
Appendix 2: Sample interview questions.	33
Appendix 3: Average number of walk-in students.	34
Appendix 4: Number of inquiries though email or phone.	35
Appendix 5: Average waiting time in front of administrative counters.	35
Appendix 6: Sample questions and answers for repository.	36

LIST OF FIGURES

Figure 1:	Architecture of Sandesh System	8
Figure 2:	Definition section	9
Figure 3:	Command section	9
Figure 4:	Snippet of QRR command section	10
Figure 5:	Text-mining process	13
Figure 6:	Sample Architecture layout of Question answering system	15
Figure 7:	RAD using Iterative Prototyping	17
Figure 8:	Flow-chart of 'MyQuestion' Inquiry System	19
Figure 9:	Architecture of 'MyQuestion' Inquiry System	20
Figure 10:	OSI and TCP/IP models	23
Figure 11:	Application layer protocols	24
Figure 12:	Domains and Accounts created	25
Figure 13:	Users under domain: utp.edu.com	25
Figure 14:	Roundcube login page	26
Figure 15:	phpMyAdmin page	26
Figure 16:	Sample RapidMiner page	27
Figure 17:	Sample email sent by student on visa problem	27
Figure 18:	Confirmation email sent to students' inquiry by server	28
Figure 19:	Provided answer by server to students' inquiry	28

CHAPTER 1

INTRODUCTION

1.1. Background of Study

As students struggle to get timely response to their inquiries in administrative departments, the necessity for automated question answering system becomes more important. There is a need in a system that allows a student to ask a question in everyday language and receive an answer quickly and succinctly, with sufficient context to validate the answer.

In search for best communication tool that would allow achieving this goal, Email has been seen as most relevant, mainly due to its efficiency, low cost, and compatibility of diversified types of information. Studies show that email allows better communication and provides students with the opportunity to ask more questions (Sadat & Rahman).

In order to facilitate better usages of emails and explore academic and business potential in emailing, email mining, which applies data mining techniques on emails, has been conducted extensively and achieved remarkable progress in both research and practice. Particularly, emails can be regarded as a mixed information cabinet containing both textual data and human social, organizational relations (Tang, Pei & Luk, 2013).

1.2. Problem Statement

As previously mentioned, there has been seen substantial gap in the process of responding to students' inquiries.

- I. Process inefficiency that comes from long queue standings for students in front of the administrative counters to address their issues (See Appendix 1)
- II. Many students fail to recognize which department or unit they should refer to with their questions. This, in turn, leads to the situations where students have to line up in more than one counter until they find right personnel who can assist with regards to their question
- III. Inefficiency of this practice for staff results in necessity to answer same questions or respond to same problems that students come with countless times every day
- IV. Telephone, as a mode of communication is highly ineffective, since the calls are mostly not answered because of the need to serve walk-in students first

Eventually, due to the lack of single comprehensive system, both students and staff are not satisfied with the current state of handling inquiries which is time-consuming and human-relied.

1.3. Project Objectives

The main objective of the project is to *iteratively* develop system prototype following *Rapid Application Development* approach.

- To conduct research and carry out in-depth analysis in order to gather relevant information and knowledge essential for project success
- To design system prototype at initial iteration of the project phase
- Following the analysis and design, examine necessary tools required for prototype development
- Based on the analysis, design, and development requirements build 'MyQuestion' Inquiry Server to manage daily questions of students regarding any administrative matters. It will allow students to get timely response to their inquiries without the need to go to the counter and eliminate the dependency on working hours of administrative staff.
- Test Automated E-mail Response functionality of the system between server and a student. This way, majority of inquiries can be handled by server, leaving more productive hours for staff and students.

1.4. Project Scope

Defining project scope is critical toward successful delivery of the system that targeted to be developed. Hence, below described the detailed outline of project range which lies within its scope:

- Intended users for this system are UTP students and staff of specified administrative departments
- System functionality includes automatic email responding from the server side; first that sends a confirmation of inquiry received and second corresponding answer to user's inquiry
- It is aimed to conduct thorough research and interview with administrative staff with the purpose of acquiring necessary information on most commonly addressed issues by students and relevant answers to that inquiries by staff (which currently is one of the factors that resulted in process inefficiency due to the need of repeating same responses to different students by staff) that will be stored on server database
- Great amount of project effort lies on in-depth study of text-mining techniques and methods which is essential part of the system (the ability of server to extract the text and analyse it in order to provide suitable answer to the end user)
- One of the aspects of system is to apply specific algorithms that empower the server to recognize certain '*key words*' or *patterns* within the email content (or text) and ability to acknowledge to which department the certain inquiry should be redirected

In order to complete the given project within specified time frame, following system boundaries will be put ahead to establish a sound project scope:

- The project, as at its initial stage, will consider to handle the inquiries for only three departments – Registration, Exam Unit, and a Student Support Services Department (SSSD)
- Server performance features such as workload (number of simultaneous inquiries) and scalability will be overlooked. Moderate level of workload that server is able to handle is assumed at this level of development

1.5. Project Relevancy and Feasibility

By successful completion of the project, end system is expected to aid UTP students (foundation, undergraduate, postgraduate, local, international, and exchange) and administrative staff in handling day-to-day inquiries. The proposed system will enhance overall process efficiency by eliminating reliance on human factor to certain rationale degree.

By the end of the project completion, it is expected to demonstrate how certain system is able to overcome described problem. Moreover, the single email address to which students can send any inquiries will help to avoid confusion on which department handles what kind of administrative issues.

In addition to that, system will assist administrative staff of UTP from Exam Unit, Registration and SSSD (*Student Support Services Department*) to concentrate on their daily work activities since the server will tackle most of students' questions automatically without human interference.

Having this system is pertinent to University's administration performance, since not all information is available on the main website, and reaching the staff through email is easier than through phone.

CHAPTER 2

LITERATURE REVIEW

2.1. Related and Similar Works Done by Others

Ouni & Neji (2012) suggested great application of *Email Response Automation* in the field of E-learning to ease the burden of tutors in the interaction with students while trying to provide timely reply to their questions. Their proposed system is aimed to solve the problem of increased number of e-mails received by the tutor, who must ensure students get corresponding answers to their inquiries and all within short period of time, or in other words as quickly as possible. So a system of automatic reply to emails is a solid method that serves to minimise the processing time of pool of emails in order to attain the satisfaction of the learners.

The easiest form of automatic reply systems is an *Auto-response system*. Most of the times also referred as on-demand-emails, infobots or mailbots. This type of system disregards the content of the email; it is rather used to manage mailing or sending lists. Auto-responders return a static document in response to an e-mail according to the presence of keywords in the subject or body of the message. Alternatively, the user can fill a set of predefined fields in a web form to customize the response (Lapalme & Kusseim, 2003). An evident disadvantage of such systems is their inability to analyse the content of free-form messages. The content of the text is reduced to a small set of keywords with no regards to the true meaning of the text. Despite that, it is useful in sending acknowledgments of receipt, until responder is available to work on the content of email (Ouni & Neji, 2012).

There is another system which has several functionalities – *System of Email Management*. It is able to:

- Classify contents of e-mails received
- Send automatic confirmation emails
- Suggest possible responses
- Integrate the system according to the working context of the responder
- Options enabling Response Pattern Customization

Likewise, some other systems are capable of responding to email content without human interference. These systems are – *Auto-Response Email by Keywords in the Subject*, which is able to analyse the message, select the response pattern and send it to the requester automatically (Ouni & Neji, 2012).

More sophisticated types of e-mail responders are included in e-mail management systems, and can provide pre-written response templates for frequently asked questions. Slots are usually filled in with information extracted manually from the incoming mail, although some systems seem to perform the extraction automatically (Lallement & Fox, 2000).

One of the progressive email management systems – *Auto-Response Web* can analyse the body of the e-mail. These systems execute on a condition that the email sent by completing a web form rather than writing a free text in natural language. Since the data is well structured within the form, it eliminates the necessity to interpret the natural language (which presents in unstructured e-mails) making it relatively easier for the system to automate the reply (Ouni & Neji, 2012).

Automatic Email Response System *Sandesh* functions as an agent between students and tutor. Such system is executable with the help of Natural Language Processing (NLP) tools and techniques, information retrieval structures and web technologies. Sandesh uses features that facilitate queries with fixed formats to ease the job of searching for suitable data source. Fixed format queries are essential since they give high reliability in response however they can be quite limiting in their scope (Aparna & Sasikumar, 2007).

The flow of process is as follows:

- I. When new email is received by tutor, the system checks if it is a fresh mail or follow up mail. This segregates two types of emails and indicates that follow up email is received due to the unsatisfactory answer sent automatically by system to the learner at the first stage and that student wants to consult directly with the tutor. The logic of system's architecture is described in the figure below.

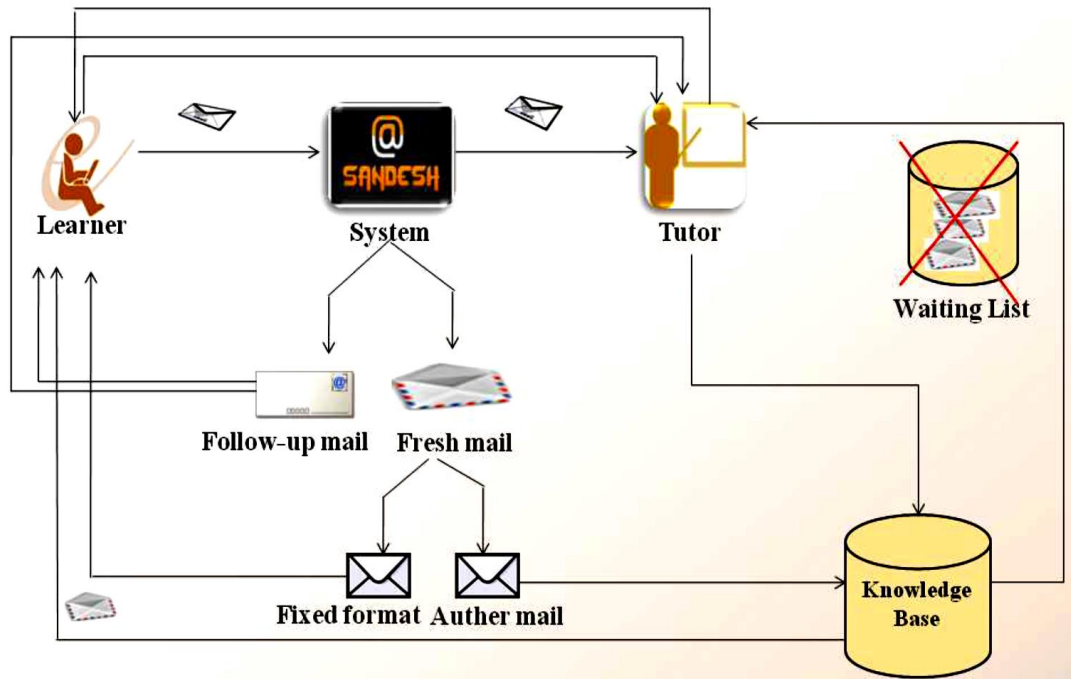


FIGURE 1: Architecture of Sandesh System

- II. If the mail received is a fresh mail, system verifies if it is fixed format mail or any other mail.

- III. In case if the email is in fixed format, Sandesh carries out respective command and sends out the answer to the learner (*Emails with set subject patterns have standard results to generate responses*). Command Structures are designated in Command File that represented in XML file (structure of XML is seen as practical in representing data and allowing for custom-defined tags). Following figures describe two main sections of this command.

```
<DEFINITIONS>
  <ROLLNO match=" [[: alpha:]] [[: digit
:]]{7}" />
  <MODULE
MATCH="(java|c|webprogramming|networks)" />
  <TIME match="(time|date)" />
</DEFINITIONS>
```

FIGURE 2: Definition section

```
<COMMAND>
  <VAR value="MODULE">
  <WORD value="RESULT">
  <VAR value="ROLLNO">
  <REPLY>
    <URL method="POST" value="http://202.141.152.1/Rollno">
    <FORMVAR name="moduleid" value="MODULE"/>
    <FORMVAR name="rollno" value="ROLLNO"/>
    </URL>
  </REPLY>
</COMMAND>
```

FIGURE 3: Command section

‘Command’ section determines keywords and variables composing command structure. For example, as seen in second section, RESULT is a keyword and MODULE and ROLLNO are predefined variables. This command is matched by the system against the subject line of the email, if it contains one of these three words. REPLY tag carries out the action that needs to be performed if above condition is satisfied and query matches given structure.

IV. In the condition of new email to be other than fixed format mail, the following sub-tasks take place within the system (Aparna & Sasikumar, 2007):

- Look up the *Query Response Repository* (QRR – storage of previously answered questions, regularly updated with new responses and queries) for a conforming solution which is also represented in XML file. If the answer is found, send it to the student with a notification that this is an automatically generated response, as in figure below:

```
<REPLY>
  <WORD val="FINAL"/>
  <WORD val="YEAR"/>
  <WORD val="STUDENT"/>
  <WORD val="WHICH"/>
  <WORD val="CST"/>
  <WORD val="LEVELS"/>
  <WORD val="ELIGIBLE"/>
  <QUESTION>
    <![CDATA[ I am a final year student. Which levels
of CST am I eligible for?]]>
  </QUESTION>
  <MIN-MATCH val="25"/>
  <BODY>
    <![CDATA[Students must be in the final
year of the respective degree/diploma in order to be
eligible, except for students of MCA, who can write the E,
I or D-level examination. Post-graduate students may apply
for any level for which they are eligible.]]>
  </BODY>
</REPLY>
```

FIGURE 4: Snippet of QRR command section

Everything within the QUESTION tag specifies inquiry sent by the student and BODY tag specifies given response by the tutor.

- If there is no suitable response found within the QRR, system identifies the tutor who will be able to handle the query by forwarding the mail to him and marking the mail request as pending.

- V. For the mail reply by tutor, system Sandesh performs the tasks as below:
- Identifies the type of mail – whether it is a fresh mail or the follow up mail.
 - If it is a reply to a follow-up email, system eliminates the respective query-mail from the pending list and sends off the response to the student.
 - When it is a reply to newly received (fresh) email, Sandesh removes the corresponding mail from the pending list and renews the QRR with latest query and the tutor's relative response to it. After that forwarding given answer to the learner.
- VI. For the follow up emails, system marks it pending and forwards the letter to respective tutor.

In order to execute certain system, many design issues are required to be performed. Some of them are described below.

A. How system recognizes similar queries

This feature is at the core of the Sandesh as it determines how well the system can perform. System must be able to identify how many useful words in a new received email are coming across in any other email.

To determine which words are best for matching, the domain needs to be specified. For instance, in the domain of E-learning key words are 'exam', 'course', 'grade', and etc. this technique enhances the quality of results generated. The same way, a match should not be one-to-one, i.e. a word 'when' in inquiry of one email can conform to specified date or time in some other email, just like in the example below (Aparna & Sasikumar, 2007):

*When will be the exam for our Management course?

*Is our exam for Management on May 12th?

Basically, both of the questions ask about the same thing. Yet there is no 'when' in the second question and no 'May 12th' in the first one. However, we know that both queries are looking for similar information and therefore must be matched to assure effective output. Thereby, simple word-to-word match won't be much of a help if synonyms and other related words are not considered in the domain for better conformation of the result (Aparna & Sasikumar, 2007).

B. Finding the tutor to whom the system needs to redirect learner's inquiry

Generally, course may include more than just one tutor/lecturer. System of Sandesh is required to determine to which tutor the query needs to be sent to answer the corresponding inquiry or question. Different matching algorithms are applied to look for similar inquiries.

2.1.1. Application of Automated Email Answering in Corporate World

Increasing number of companies are benefitting from the use of email as one of their customer service channels, which answer customer inquiries about products or services. Such companies mostly hire employees to answer questions from customers manually. Though, same or similar questions are often asked again and again by different customers. An automatic email answering system can help those companies to save some labour force in email customer service.

Bickel and Scheffer (2004) developed an automatic email answering system. An SVM classifier is used to find the right cluster for a newly incoming inquiry. An email customer service data set provided by a large online store was used to evaluate the performance of the classifier. The experiment results show that the classifier has precisions about 40-90% and recalls about 5-40%.

Many of the e-mails sent to corporations are asking for information and can be considered as questions from customers to which representatives should answer in the best possible way. Automatic E-mail Follow-up System ***Mercure*** applies a technique which is based on Question-Answering (QA) technology: the task of finding an exact answer to a natural language question in a large set of documents (Lapalme & Kusseim, 2003). The question type is determined by the presence of trigger phrases (e.g. where, how many, how much), which indicate the type of the answer required (e.g. location, number, money). Information retrieval is typically performed to identify a subset of the documents and a set of passages that may contain the answer.

2.2. Overview on Text Mining: *Techniques and Applications*

Gupta and Lehal (2009) described Text Mining as an emerging field that engages information retrieval (IR), data mining, machine learning, statistics and computational linguistics. This is due to the fact that more than 80% of the corporate, academic, and commercial information is stored as text, and also because unstructured texts still represent biggest available source of knowledge. Figure 5 outlines overall process of text mining:

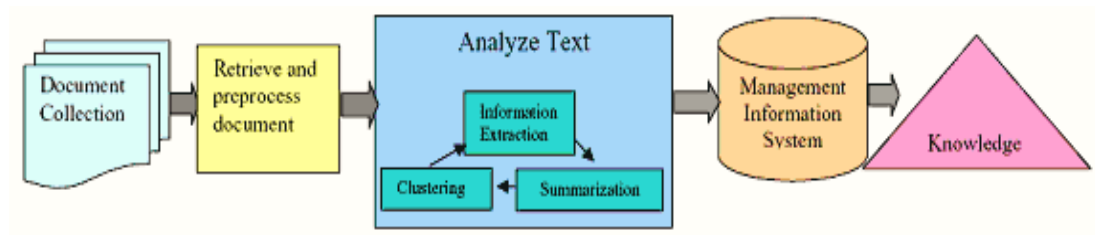


FIGURE 5: Text-mining process

Although terms *text mining* and *data mining* are used interchangeably, the difference between two is that data mining tools are intended to work with structured data from databases, whereas text mining is able to manage unstructured and/or semi-structured data sets such as email contents, any type of full-text office documents and HTML files. Consequently, many companies opt for text mining because of its scope and capabilities to manipulate the text that comes from different sources (Navathe & Ramez, 2000).

One of its aspects – natural language processing has generated technologies that is able to instruct computers and machines natural language by which they can analyse, understand, and process text (Fan, Wallace, Rich & Zhang, 2006). Fan et al (2006) have described some of the technologies currently applied in text mining process.

1) *Information Extraction*

Gupta et al (2009) state that in order to empower computers with capabilities to analyse unstructured data one should start from making use of information extraction (IE). This software recognizes key terms and relationships in textual data by searching for predetermined sequences within that text. This procedure is referred to as *pattern matching*. Software encloses relationships between time, places and people to produce meaningful pattern of information. This mechanism is genuinely helpful in dealing with text data in greater volumes (Gupta & Lehal, 2009).

Information Retrieval (IR), which, following convention, we take to be the retrieval of relevant documents in response to a user query, has been an active research area for the past few decades. It is related to question answering in the sense that users form queries because they wish to find answers to questions. However, beyond this the similarity largely ends (Hirschman & Gaizauskas, 2001). IR systems return documents, not answers, and users are left to extract answers from the documents themselves.

IR is, however, relevant to question answering for two reasons. First, IR techniques have been extended to return not just relevant documents, but relevant passages within documents. The size of these passages can be steadily reduced, at least in theory, so that in the limiting case, what is extracted is, effectively, just the answer to a question. Thus, question answering can be thought of as passage retrieval in the limit. Second, the IR community has, over the years, developed an extremely thorough methodology for evaluation, the most well-known current exemplars of which are the annual Text REtrieval Conferences, or TREC's, run by the US National Institute of Standards and Technology. It is from this methodology and community that the recent question answering evaluation developed, which in turn has stimulated much of the current interest in question answering

2) Question Answering

Other areas of application for NLP is *natural language query* or as it's usually called *question answering*, that mainly concerns with how to find a best solution for specific query or question. It can employ numerous text mining techniques, like extracting information from entities such as events, time and places mentioned above, or categorize questions in order to appoint them into predefined types like 'who', 'when' or 'where' and etc.

Natural language question answering (NLQA) is currently an active area of research. Though surprisingly few NLQA systems are available for use by the general public, research is plentiful and several systems appear to be forthcoming.

Apart from its common use in web applications, it is also widely known for internal uses to address the questions of employees looking for the answers regarding many common practices and procedures within the company. Moreover, it can come across in the fields of education and medical too to serve the same purpose. Figure 6 below shows how this system works:

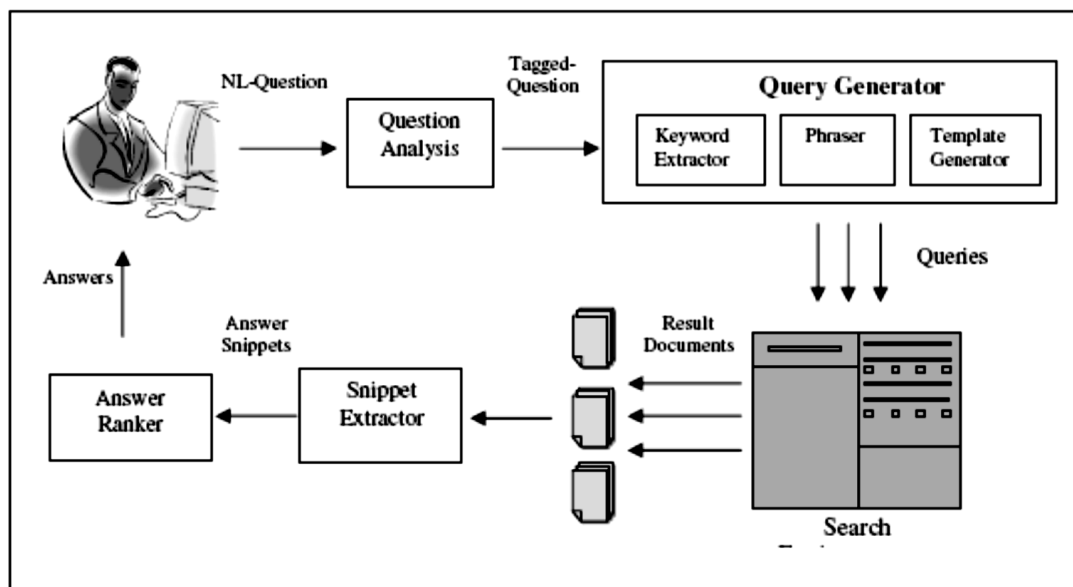


FIGURE 6: Sample Architecture layout of Question answering system

The number of free-form electronic documents available and needing to be processed has reached a level that makes the automatic manipulation of natural language a necessity. Manual manipulation is both time-consuming and expensive, making Natural Language Processing (NLP) techniques very attractive. E-mail messages make up a large portion of the free-form documents available today and as e-mail becomes more and more popular, an automated e-mail answering service will become as necessary as an automated telephone service is today.

QA differs from e-mail answering in several aspects. Generally speaking, e-mail answering involves *analysing* a longer text and *formulating* a linguistically-motivated answer, while QA takes a short and explicit question as input and focuses on locating the answer. Issues in discourse analysis and generation must therefore be addressed in e-mail answering, but not in QA. In addition, questions, at least in systems participating to the TREC evaluations, are restricted to specific types such as who, why, where, ... but pertain to an unrestricted discourse domain. On the other hand, in e-mail answering, the questions are of unrestricted type, but the discourse domain is typically restricted. E-mail answering thus involves finding passages from the textual knowledge base that best relate to the incoming message and sending the passages as is to the user.

CHAPTER 3

METHODOLOGY

3.1. System Development Approach

Rapid Application Development (RAD) using Iterative Prototyping method is selected for successful designing and completion of ‘MyQuestion’ Inquiry System. The processes involved in given development approach are shown in the Figure 7:

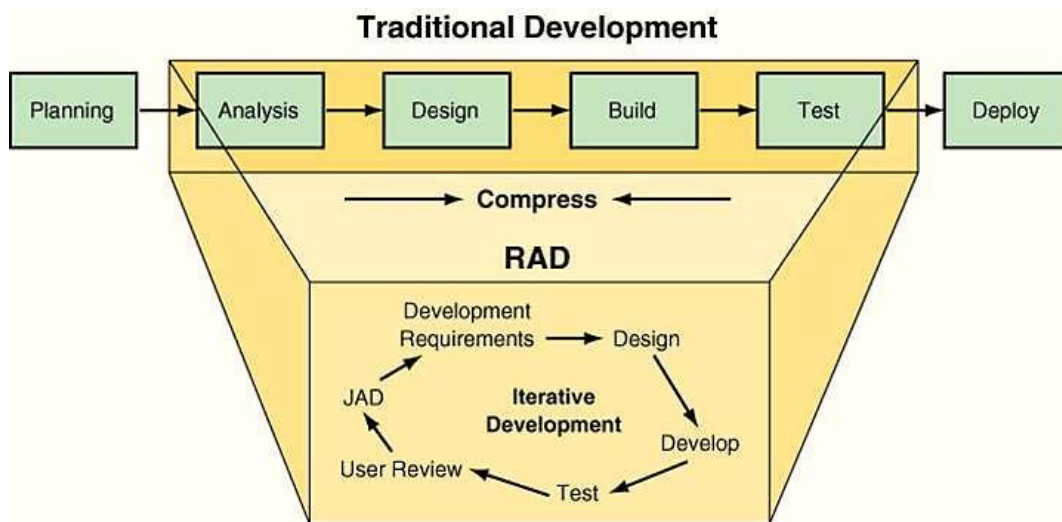


FIGURE 7: RAD using Iterative Prototyping

This approach is preferred over others due to its applicability for developing highly interactive systems and applications with clearly identified user groups. Iterative Prototyping helps to focus on building the system in a short period of time, by overlooking additional features and concentrating only on main system functionalities.

3.2. Iteration 1 – Analysis phase

Qualitative approach has been taken at preliminary stage of research which is helpful and relevant in assessment of opinions, attitudes and behaviour. In-depth interview session technique of this approach has been applied with an eye to analyse the insights of the current situation within inquiry management of administrative departments, and the scale of the problem from the perspective of students and corresponding staff.

Findings from conducted study confirmed the initial hypothesis on the scale of current problem. Interviewees comprised of random students, and respective permanent administrative staff of Exam Unit, Registry and SSSD. Each group has been interviewed separately (See appendix 2); to be more precise since the structure of the questions were open-ended; it was decided to interview them individually for getting better insights. Following results has been gained from the study.

Average number of walk-in students is from 40 to 80 in Exam Unit and Registration department respectively (See appendix 3). This by itself already creates a favourable condition to form a queue and keep personnel busy in front of the counter while having other office work to carry out as well. The situation is much worse in Student Support Services, as the number of students almost double than in other two departments – 150.

Another question posed was the number of phone and email inquiries received from students on daily basis. The aim of the question was to analyse if students try to resolve their issues before or without physically coming to the counter. Findings indicated that staff gets almost equal number of requests through email and phone simultaneously having other students waiting at the counter to be served, hence making it difficult to respond to them on timely manner (See appendix 4).

The problem of recognizing respective department to address their issues among students has proven to be correct. 100 percent of staff interviewed has confirmed the hypothesis. This is due to the fact that university practices, rules and regulations are constantly keep changing, therefore confusing students with regards to this question.

The interview revealed that staff preferred email as a mode of communication to be more convenient and effective than others. 100 percent of staff respondents favoured email over other communication medium to handle the student inquiries. And once explained how the proposed system will work like, the feedback was positive and encouraging, saying that it would have been very helpful to have such system.

When looking at the problem from students' perspective, 55 percent of the respondents stated that average waiting time at the administrative counters are from 15 to 30 minutes (See appendix 5). 20 percent complained that it can take up to 45 minutes at times. 10 percent of interviewed students expressed their frustration as the queuing took 1 hour until they reached their turn. And only 15 percent of respondents said that they have been served within 5 to 15 minutes.

3.3. Iteration 1 – Design phase

3.3.1. Flow-chart

Following flow-chart is outlined to depict the logic and process of how the system's work will look like to give the overall view on system's functionality.

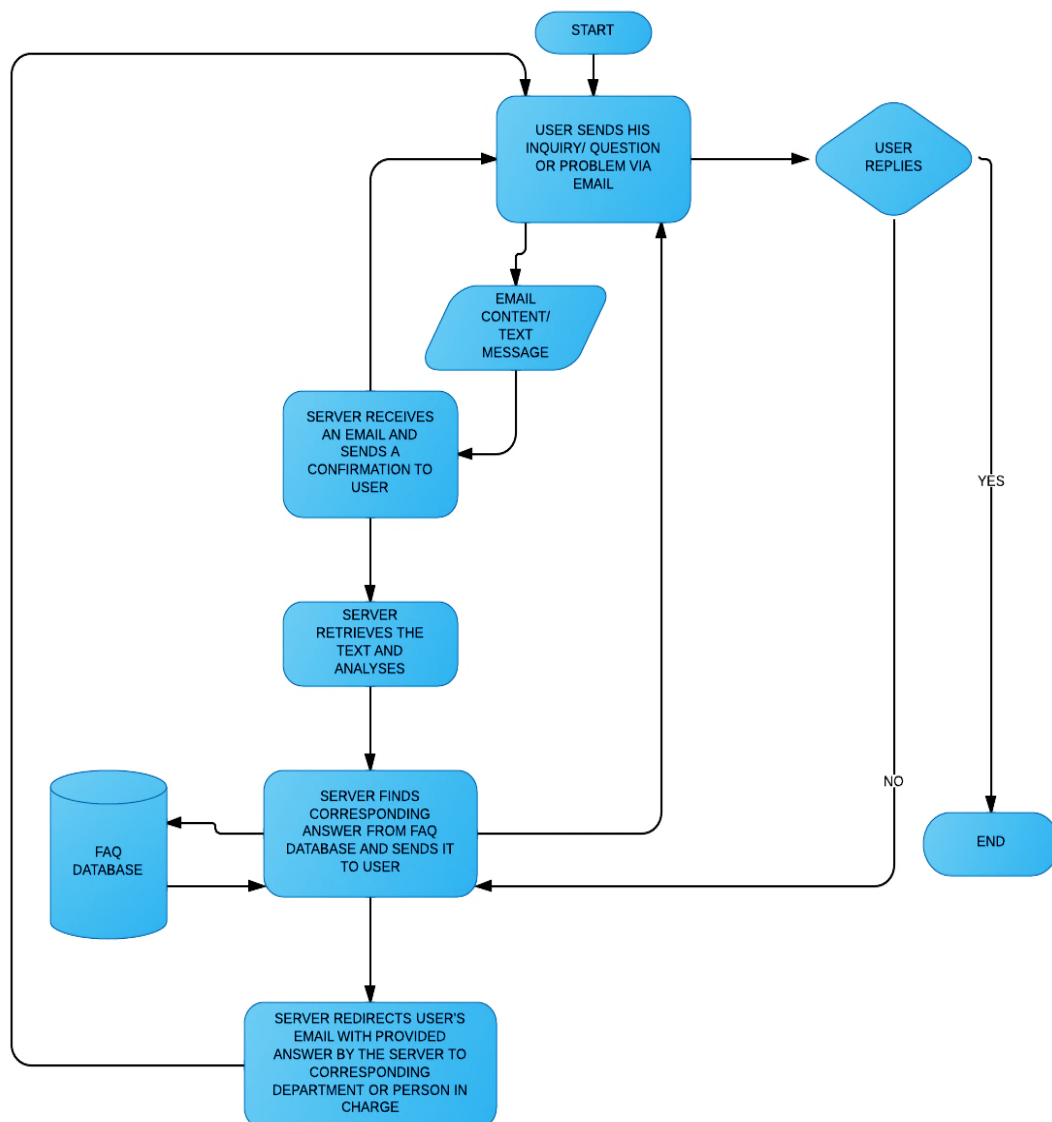


FIGURE 8: Flow-chart of 'MyQuestion' Inquiry System

3.3.2. System architecture

The architecture layout of proposed system has been developed during design stage of the project lifecycle to demonstrate the overview of ‘MyQuestion’ Inquiry System.

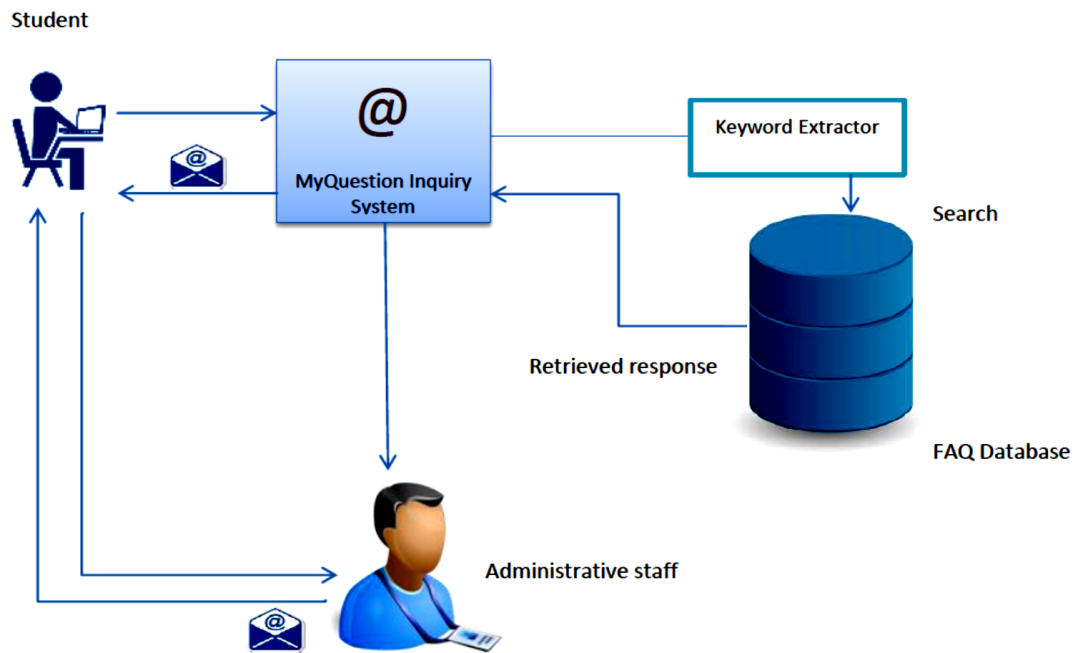


FIGURE 9: Architecture of ‘MyQuestion’ Inquiry System

CHAPTER 4

RESULTS AND DISCUSSION

4.1. Iteration 2 – Analysis phase

Another session with administrative staff has been conducted in order to gather sufficient and relevant information on mostly addressed issues, questions and inquiries by students. The main focus was to capture the information which mostly not available on university website or any other reference. Frequently asked questions and respective answers to them have been gathered in order to store in system's server (See appendix 6).

4.2. Development requirements

4.2.1. Tools used

VMware Player – virtualization software package

VMware Player can run existing virtual appliances and create its own **virtual machines** (which require an operating system to be installed to be functional).

Being that a system is to be developed as a prototype, it has been found feasible to build a testing environment where all clients and a server is created and managed locally. For this purpose, installing a virtual machine has proven to be an implementable solution.

iRedMail – open source Email Server Solution

Comprises major open source softwares:

- **Postfix (SMTP service)** – default mail transfer agent (MTA) that routes and delivers electronic mail, intended as an alternative to the widely used Sendmail MTA.
- **Dovecot (POP, IMAP)** – an open source IMAP and POP3 email server. Works well for both small and large installations. Genuinely faster, easier to set up, requires no special administration and uses very little memory. Dovecot can work with standard **mbox**, **Maildir**, and its own native high-performance **dbx** formats.

It also includes a **Mail delivery agent (MDA)**. It maintains highly secure design and implementation features.

As an IMAP and POP3 server, Dovecot provides a way for **Mail User Agents (MUAs)** to access their mail. So when a user's MUA contacts the mail server, the software which answers that request is an IMAP or POP3 server. IMAP and POP3 servers take requests from MUAs and answer those requests by accessing e-mail messages stored on the server and feeding them out to the MUA using IMAP or POP3. Dovecot is one program which can provide that IMAP and POP3 server functionality.

- **Apache (Web server)** – open-source HTTP server that provides a secure, efficient and extensible server supporting HTTP services in sync with the current HTTP standards.
- **MySQL/PostgreSQL** – storing application data and/or mail accounts. As a database server, its primary function is to store data, securely and supporting best practices, and retrieve it later, as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). It can handle workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.
- **OpenLDAP** – storing mail accounts.
- **Roundcube (Webmail)** – is a free and open source webmail solution with a desktop-like user interface which is easy to install/configure and that runs on a standard LAMPP server. It is a browser-based multilingual IMAP client with an application-like user interface.

Roundcube is used by a number of universities to provide email services to students and staff. Roundcube Webmail is designed to run on standard web servers such as Apache in conjunction with a relational database engine. Supported databases are MySQL and PostgreSQL. Roundcube connects to any IMAPv4 server.

RapidMiner – software platform

Provides an integrated environment for machine learning, data mining, text mining, predictive analytics and business analytics; It is used for business and industrial applications as well as for research, education, training, rapid prototyping, and application development and supports all steps of the data mining process including results visualization, validation and optimization. RapidMiner uses a client/server model with the server offered as Software as a Service or on cloud infrastructures.

4.2.2. Network and Communication Architecture of the System

Applications such as e-mail clients incorporate functionality of the OSI layers 5, 6 and 7 in the figure shown below. Hence, the project is planned to work mainly on application layers.

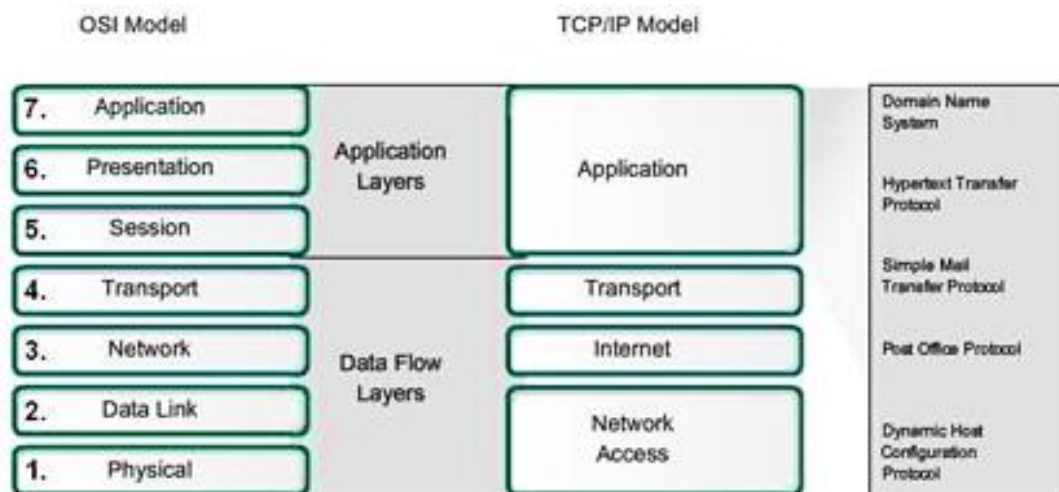


FIGURE 10: OSI and TCP/IP models

Some of the widely-used TCP/IP **Application layer ISO OSI** protocols such as:

- Domain Name Service Protocol (DNS) to resolve Internet names to IP addresses and
- Simple Mail Transfer Protocol (SMTP) for the transfer of mail messages

will be implemented in developing the project prototype. E-mail applications are able to execute certain protocols and communicate directly with the lower layers. Examples of such Application layer protocols are Post Office Protocol (POP) and Simple Mail Transfer Protocol (SMTP), shown in the following figure that defines client/server processes:

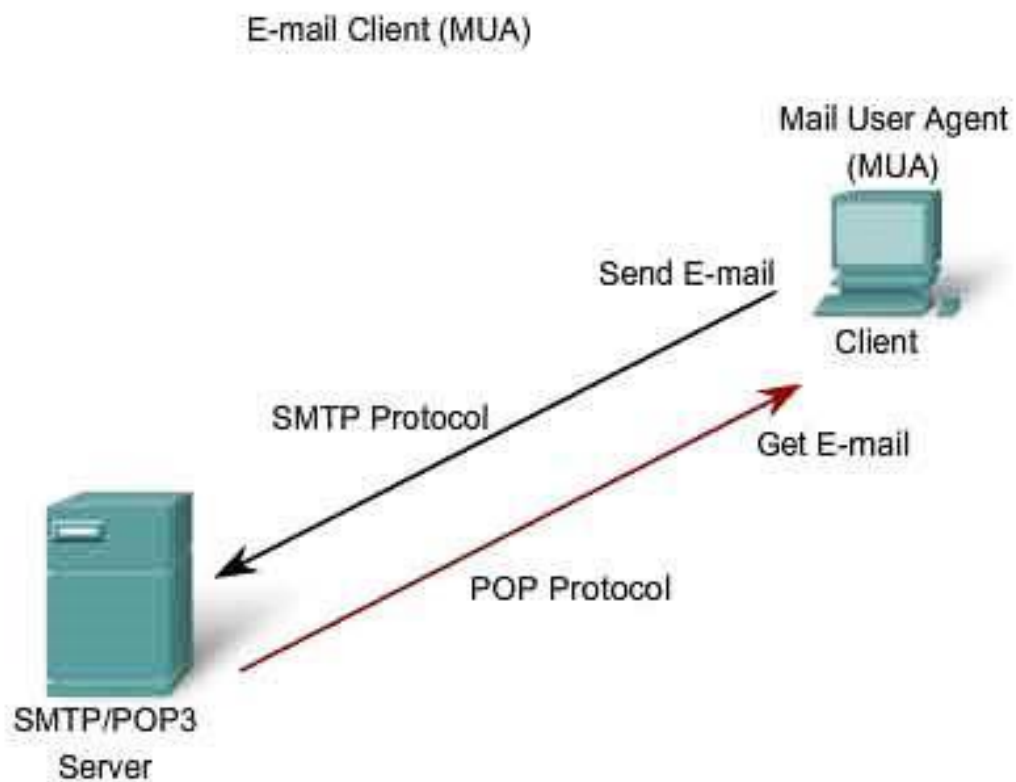


FIGURE 11: Application layer protocols

When e-mail messages are composed, the application called Mail User Agent (MUA) or e-mail client is used. To receive e-mail from e-mail server, the e-mail client implement POP. Sending e-mail from either a client or a server uses message formats and command strings defined by the SMTP protocol.

4.3. Iteration 2 – Develop

The following **domains** and **accounts** have been created. Under each account multiple **users** representing students, staff and a system (*server*) have also been created. To test the multi-domain functionalities different users have been created under various domains and a sample emails have been sent to test if the system is able to recognize and receive emails from different domains and email accounts.

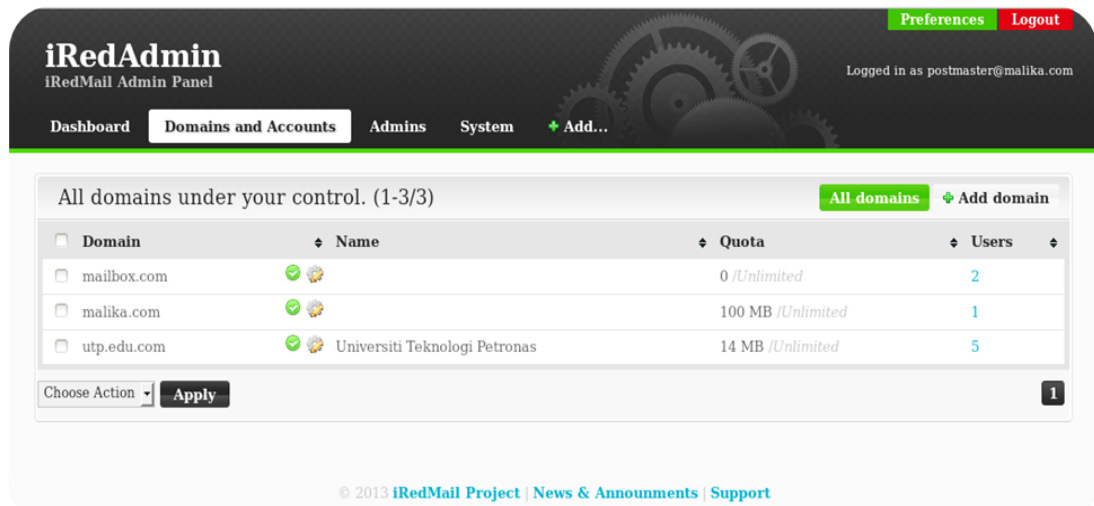


FIGURE 12: Domains and Accounts created

For students to send their inquiries an email account myquestion@utp.edu.com under the domain utp.edu.com has been created. Alongside, other email accounts allocated for each department have been created for further email redirection purposes. The accounts under the domain utp.edu.com are shown in the figure below:

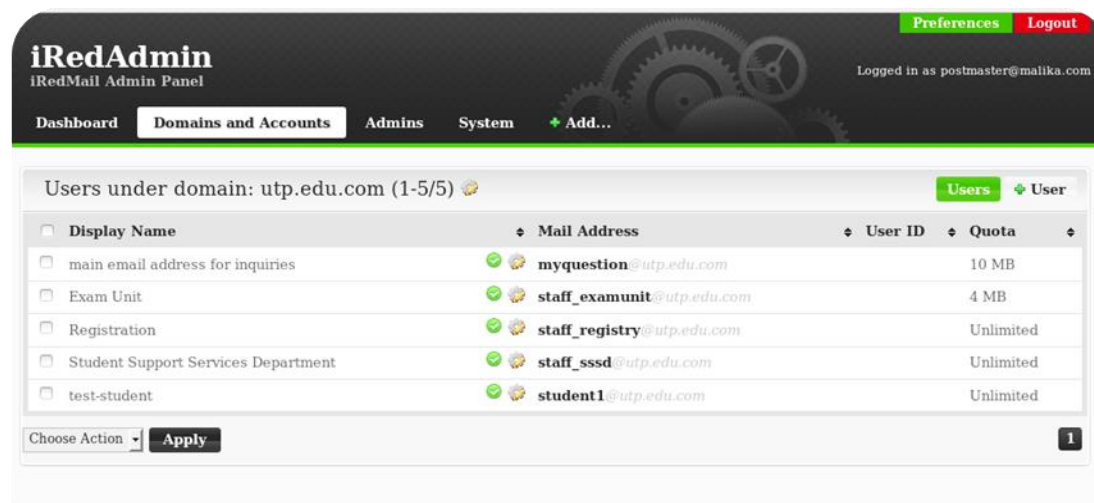


FIGURE 13: Users under domain: utp.edu.com

Since the system is being built and tested locally (*on localhost*) the email client *Roundcube* is used in order to login as different clients and users, i.e. students, system, staff to check if the emails sent have been received successfully among clients and a server. The figure below shows the login page:

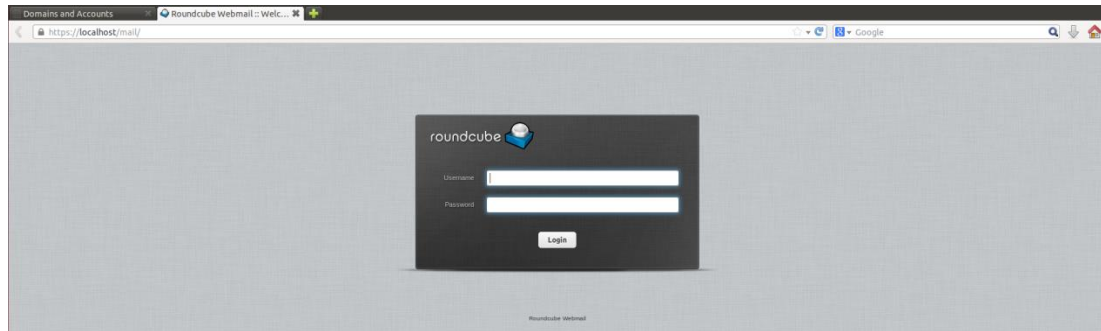


FIGURE 14: Roundcube login page

phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc.) can be performed via the user interface, while still having the ability to directly execute any SQL statement. Following figure displays phpMyAdmin page for managing the server settings.

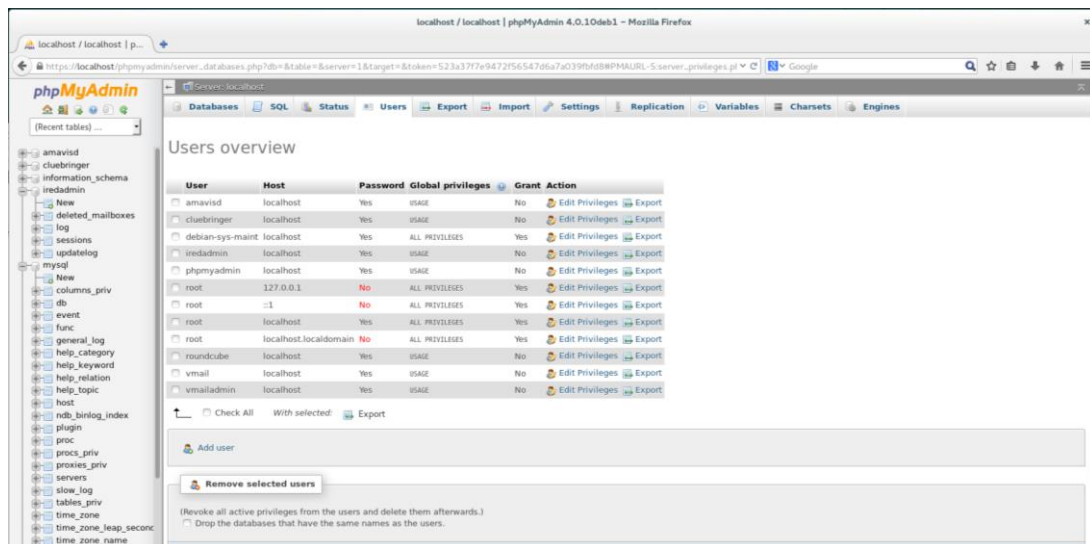


FIGURE 15: phpMyAdmin page

Sample page of RapidMiner – our selected tool for mining techniques is shown in the figure below.

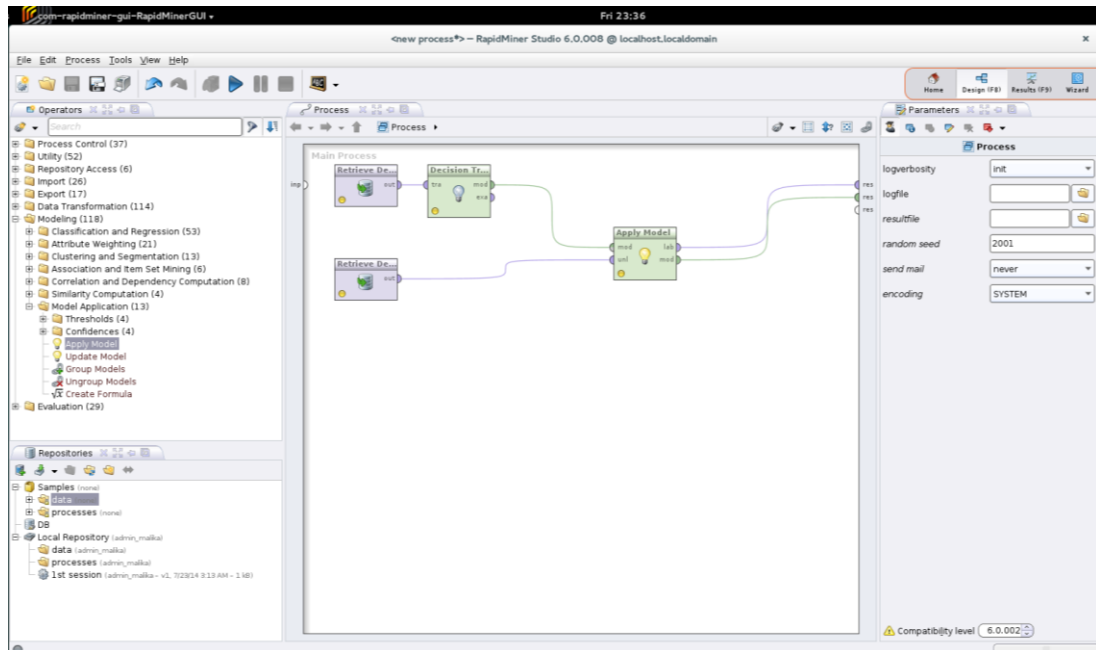


FIGURE 16: Sample RapidMiner page

4.4. Iteration 3 – Test

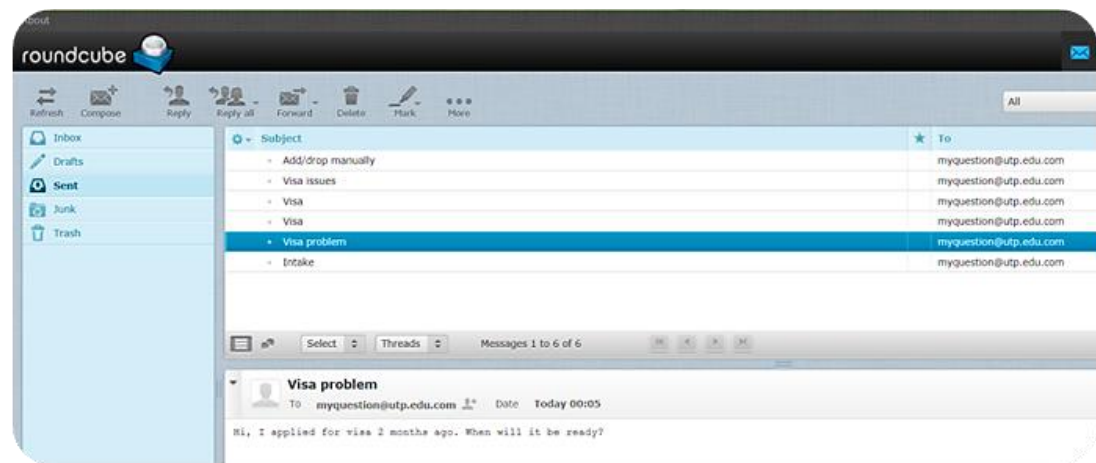


FIGURE 17: Sample email sent by student on visa problem

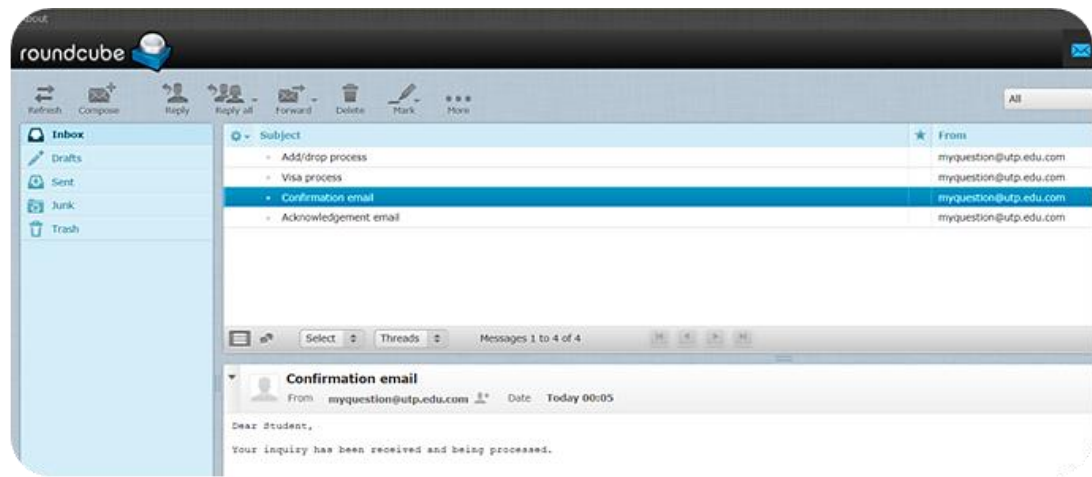


FIGURE 18: Confirmation email sent to students' inquiry by server

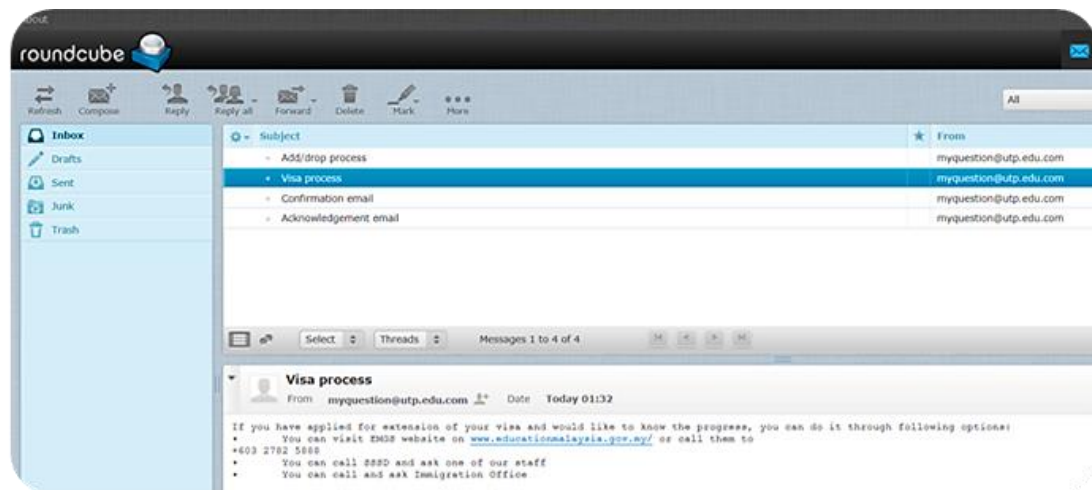


FIGURE 19: Provided answer by server to students' inquiry

CHAPTER 5

CONCLUSION AND RECOMMENDATION

In conclusion, it is believed that proposed system of automating email response between student and server is able to sufficiently decrease the staff workload and improve the overall process of handling student inquiries in administrative departments. One of the main aspects of project was to eliminate the dependency on working days and hours of administrative personnel and necessity for students to physically walk to administrative counters just to ask a question. The single email address helped to solve the issue of students' confusion on whom to address with their inquiries.

Performed critical analyses have contributed feasibly to further understanding and improving the system work throughout the project period. Furthermore, selected tools for building the system prototype has proven to be effective and appropriate for given project.

REFERENCES

- [1] Sadat, A., & Rahman, K. M. R. (n.d.). *Prospect of Email Communication as an Educational Tool for Distance Education in Bangladesh*. (Bangladesh Open University) Retrieved from http://wikieducator.org/images/7/73/PID_266.pdf
- [2] Tang, G., Pei, J., & Luk, W. S. (2013). Email mining: tasks, common techniques, and tools. *Knowledge and Information Systems*, 1-31.
- [3] Aloui, A., & Neji, M. (2010). Automatic Classification and Response of E-mails. *International Journal of Digital Society (IJDS)*.
- [4] Jangid, P., Gupta, R. H., & Sasikumar, M. (2002). Sandesh: Response Management System.
- [5] Ouni, S., & Neji, M. (2012). Automatic Response to E-mail Learner (pp. 631-636).
- [6] Aparna, R., & Sasikumar, M. (2007). Sandesh—An Automatic Email Response System. In *Innovations in E-learning, Instruction Technology, Assessment, and Engineering Education* (pp. 67-72). Springer Netherlands.
- [7] Abiteboul, S., Buneman, P., & Suciu, D. (2000). *Data on the Web: from relations to semistructured data and XML*. Morgan Kaufmann.
- [8] Gupta, V., & Lehal, G. S. (2009). A survey of text mining techniques and applications. *Journal of Emerging Technologies in Web Intelligence*, 1(1), 60-76.
- [9] Berry M. W., (2004), “Automatic Discovery of Similar Words”, in “Survey of Text Mining: Clustering, Classification and Retrieval”, Springer Verlag, New York, LLC, 24-43.
- [10] Navathe, S. B., and Ramez, E. (2000), “Data Warehousing And Data Mining”, in “Fundamentals of Database Systems”, Pearson Education pvt Inc, Singapore, 841-872.
- [11] Fan, W., Wallace, L., Rich, S., & Zhang, Z. (2006). Tapping the power of text mining. *Communications of the ACM*, 49(9), 76-82.

- [12] Berry, M. W., & Kogan, J. (2010). *Text mining: Applications and theory*. John Wiley & Sons, Ltd.
- [13] Aggarwal, C. C., & Zhai, C. (2012). *Mining text data*. Springer.
- [14] Kao, A., & Poteet, S. R. (2007). *Natural language processing and text mining*. Springer.
- [15] Highteck.net, (2014). Application Layer ISO OSI protocols functionality. [online] Available at: http://www.highteck.net/EN/Application/Application_Layer_Functionality_and_Protocols.html [Accessed 19 Jun. 2014].
- [16] Bickel, S., Scheffer, T. (2004). Learning from message pairs from automatic email answering. In *Machine Learning: ECML 2004* (pp. 87-98). Springer Berlin Heidelberg.
- [17] McGowan, K. (2005). Emma: A Natural Language Question Answering System for umich. edu. *Ann Arbor, 1001*, 48109.
- [18] Highteck.net, (2014). Application Layer ISO OSI protocols functionality. [online] Available at: http://www.highteck.net/EN/Application/Application_Layer_Functionality_and_Protocols.html [Accessed 19 Jun. 2014].
- [19] Wiki2.dovecot.org, (2012). *MailServerOverview - Dovecot Wiki*. [online] Available at: <http://wiki2.dovecot.org/MailServerOverview> [Accessed 31 Jul. 2014].
- [20] Group, D. (2014). *Apache HTTP Server Project*. [online] Httpd.apache.org. Available at: <https://httpd.apache.org/> [Accessed 31 Jul. 2014].
- [21] Lapalme, G., & Kosseim, L. (2003). Mercure: Towards an automatic e-mail follow-up system. *IEEE Computational Intelligence Bulletin*, 2(1), 14-18.
- [22] Y. Lallement and M. Fox. Interact: A Staged Approach to Customer Service Automation. In H. Hamilton and Q. Yang, editors, *Canadian AI 2000*, LNAI 1822, pages 164–175, Berlin, 2000. Springer-Verlag.

- [23] *Proceedings of the Tenth Text REtrieval Conference (TREC-X)*, Gaithersburg, Maryland, 2001.
- [24] Hirschman, L., & Gaizauskas, R. (2001). Natural language question answering: the view from here. *Natural Language Engineering*, 7(04), 275-300.
- [25] Katakis, I., Tsoumakas, G., & Vlahavas, I. (2006). Email mining: emerging techniques for email management. *Web Data Management Practices: Emerging Techniques and Technologies*, 219-240.
- [26] Busemann, S., Schmeier, S., & Arens, R. G. (2000, April). Message classification in the call center. In *Proceedings of the sixth conference on Applied natural language processing* (pp. 158-165). Association for Computational Linguistics.
- [27] Kosseim, L., Beauregard, S., & Lapalme, G. (2001). Using information extraction and natural language generation to answer e-mail. *Data & Knowledge Engineering*, 38(1), 85-100.
- [28] Sneiders, E. (2010). Automated email answering by text pattern matching. In *Advances in Natural Language Processing* (pp. 381-392). Springer Berlin Heidelberg.
- [29] Burke, R. D., Hammond, K. J., Kulyukin, V., Lytinen, S. L., Tomuro, N., & Schoenberg, S. (1997). Question answering from frequently asked question files: Experiences with the faq finder system. *AI magazine*, 18(2), 57.

APPENDICES

Appendix 1: *Queue in front of SSSD counter*



Appendix 2: *Sample interview questions*

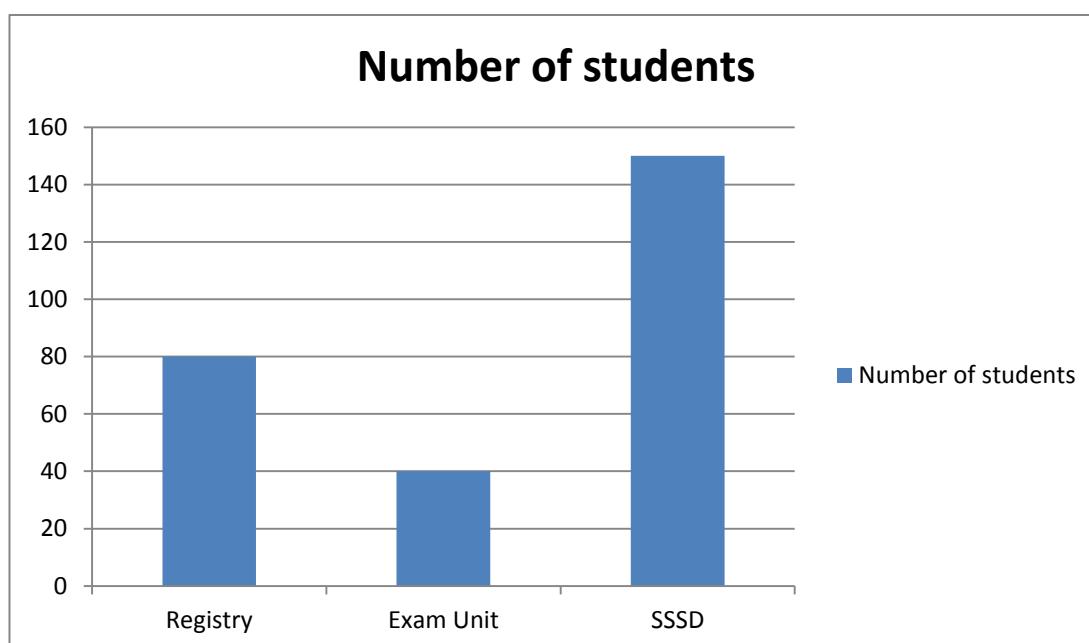
Interview questions for staff:

1. How many students do you serve daily?
2. What are their most common issues?
3. Is this information available on the website or any other place?
4. How many phone call or email requests do you get a day?
5. How often students mistakenly come to your desk with the questions that are dealt by other departments?
6. Which medium of communication do you prefer the most?
7. Have there been any such systems before?
8. If yes, how is it real-life implementation, or why it was not practiced?
9. How do you like the idea of this system?

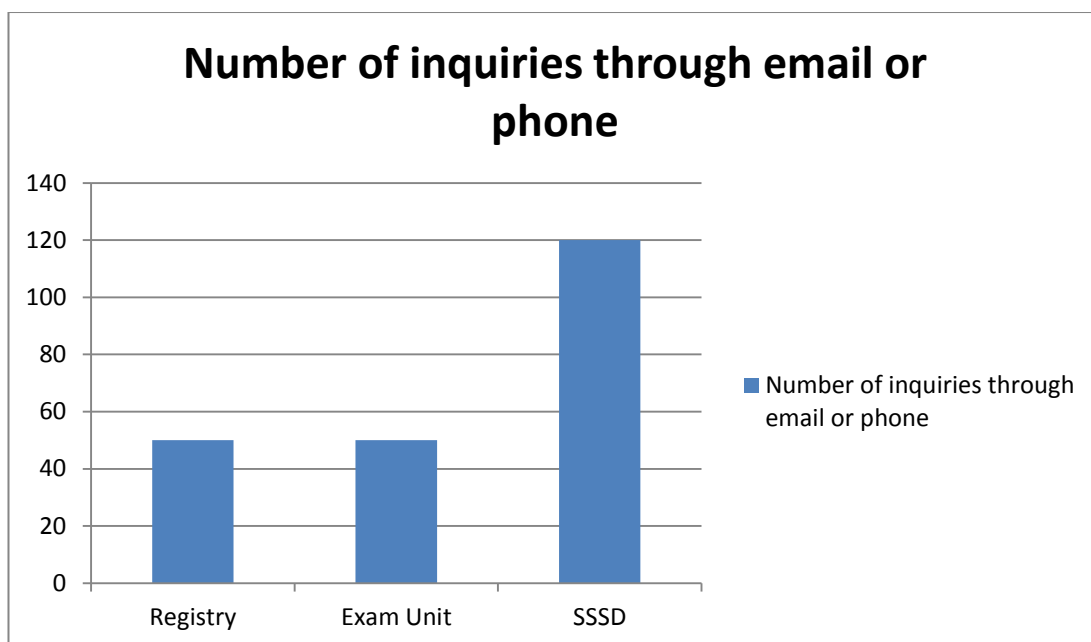
Interview questions for students:

1. Are you a local, international or exchange student?
2. How do you evaluate the current service of your requests from Exam Unit, Registry or SSSD departments?
3. Do you try to find information regarding your issue first through the website or E-learning prior to going or asking the staff?
4. Are you able to find the information?
5. Do you try to call or email the person in charge before you go to their desk?
6. What are your chances of getting timely respond?
7. Does it happen to you that sometimes you don't know which unit to go to with your issue because you don't know who is handling these problems?
8. When you go to desk, usually how long do you have to wait in a queue?
9. What do you think of such system?

Appendix 3: *Average number of walk-in students*



Appendix 4: *Number of inquiries through email or phone*



Appendix 5: *Average waiting time in front of administrative counters*



Appendix 6: Sample questions and answers for repository

Repository # 1 – Registry department

Q1:

- What is the minimum entry requirement?
- What is the minimal required CGPA level for applying to Postgraduate in UTP?
- What my CGPA should be to pass for Master's?

Answer:

The minimum required CGPA is 3.00 for applying to both Undergraduate and Postgraduate studies in UTP

Key words to search: requirement/required, minimum/minimal, CGPA, pass/apply, Postgraduate/Master's

Q2:

- When is the nearest intake?
- When is the next intake for Foundation/Undergraduate/Postgraduate?
- Do we have intake in September/January/May?

Answer:

UTP has intakes only in January and May for all Foundation, Undergraduate and Postgraduate studies. There is currently no intake in September

Key words to search: intake, nearest/next, May/January/September, when

Repository #2 – Exam Unit

Q1:

- How can I manually register if add/drop period has already expired?
- What should I do to if I want to add or drop one of my subjects after the add/drop period has been closed?
- Is it possible to add or drop some of the courses even after the period for it is over?

Answer:

If you would like to add or drop some of the courses you are taking after the add/drop period has expired, you have the option to do so, however you will have to do it manually. For that please follow the below steps:

- Take the Form from Exam Unit
- Fill up all the required fields
- Get the approval from the Lecturer(s)
- And finally submit everything to Exam Unit to process your request

Key words to search: add/drop, period, expired/finished/over, courses/subjects, manual, register, how, can

Q2:

- When is the registration date for next semester?
- When can I enrol to my courses for upcoming semester?
- Can I register for next semester subjects during my Exam week?

Answer:

The registration period for students to enrol to their courses for next semester and all the details required will be emailed to them in advance. Please make sure to check the email address you have provided to Exam Unit for administrative purposes.

Key words to search: registration/enrol, next/upcoming, semester, when, date

Repository #3 – SSSD

Q1:

- When will my visa be ready?
- How long the visa process is going to take?
- Why my visa is still not ready?

Answer:

If you have applied for extension of your visa and would like to know the progress, you can do it through following options:

- You can visit EMGS website on www.educationmalaysia.gov.my/ or call them to +603 2782 5888
- You can call SSSD and ask one of our staff
- You can call and ask Immigration Office

Key words to search: when/how long, visa, ready/not ready