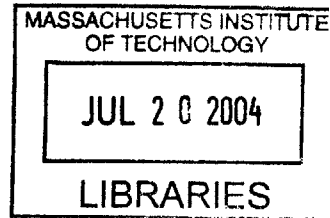


**FREiMOAT: The Free Research Extension to the  
iCampus-MIT Online Assessment Tool**

by

Jordan Michael Alperin



Submitted to the Department of Electrical Engineering and Computer  
Science

in partial fulfillment of the requirements for the degree of

Masters of Engineering in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

August 2003 [September 2003]

© Massachusetts Institute of Technology 2003. All rights reserved.

Author .....  
Department of Electrical Engineering and Computer Science  
August 22, 2003

Certified by .....  
Leslie Cooper Perelman  
Director, Writing Requirement  
Thesis Supervisor

Certified by .....  
Rafael Reif  
Associate Head, Department of Electrical Engineering and Computer  
Science  
Thesis Supervisor

Accepted by ..  
Arthur C. Smith  
Chairman, Department Committee on Graduate Students

BARKER



# **FREiMOAT: The Free Research Extension to the iCampus-MIT Online Assessment Tool**

by

Jordan Michael Alperin

Submitted to the Department of Electrical Engineering and Computer Science  
on August 22, 2003, in partial fulfillment of the  
requirements for the degree of  
Masters of Engineering in Computer Science and Engineering

## **Abstract**

MIT is currently developing a web-based service for the large-scale assessment of student writing (iMOAT.net). This service contains a database of useful data, particularly texts of student essays that should be available for research and collaboration purposes. In this thesis, I propose a high level design for an interface to the iMOAT system called FREiMOAT that will control access to this research data. This information has the potential to be used by both independent researchers as well as current users of the iMOAT system for self evaluation and collaboration purposes. Current users of the iMOAT system (administrators at a number of schools around the country), have requested the ability to view each others materials so they might improve upon their own assessments. (e.g. SMALL UNIVERSITY would like to see how STATE COLLEGE is able to use the service on larger bodies of students) Independent researchers, on the other hand, might want access to the site for purposes of determining if students from different states perform differently on the same assessments. This interface is responsible for two main tasks; access control and maintaining data anonymity.

Thesis Supervisor: Leslie Cooper Perelman  
Title: Director, Writing Requirement

Thesis Supervisor: Rafael Reif  
Title: Associate Head, Department of Electrical Engineering and Computer Science



## Acknowledgments

I have a number of people to thank for making this thesis possible.

First I would like to thank Dr. Leslie Perelman, not only for the opportunity to aid him in the creation of iMOAT, but also for taking on the additional responsibility of being my thesis advisor. I also want to thank Dr. Perelman for his exceptional taste in restaurants, and for inviting me along.

I would like to thank Rafael Reif for being the best advisor that I ever could have asked for. He has helped me through the tough times at MIT and has always supported me in my decisions. I owe you a lunch!

I fear that I never would have gotten this far without the aid of Shannon Larkin. Shannon saved me from direct exposure to feedback about iMOAT and was a great source of motivation during my less enthusiastic times. Keep singing!

I would also like to acknowledge Chris Felknor, my successor on iMOAT.net. Chris was there looking over my shoulder to make sure that I didn't make any stupid mistakes in iMOAT's design. Thanks for catching those doozies.

I would like to thank my friends, who helped me through a very difficult year - especially Galen, Dean, Rogers, Pravin, Alyse, and Ariel (who took the brunt of my frustrations but deserved the most of my love).

Lastly I would like to thank my parents, Scott and Leslie. Without their support - financial and emotional - none of this would have been possible. I love you both very very much. I made it!



# Contents

<b>1</b>	<b>Introduction, Purpose and Background Material</b>	<b>13</b>
1.1	Introduction to FREiMOAT . . . . .	13
1.2	Purpose of FREiMOAT . . . . .	14
1.3	Background of iMOAT.net . . . . .	15
1.3.1	Version 1.0 iMOAT.net architecture overview . . . . .	17
1.3.2	Issues with version 1 architecture/implementation . . . . .	21
1.3.3	Version 2.0 Architecture . . . . .	23
<b>2</b>	<b>Design Goals and Research</b>	<b>27</b>
2.1	Design Goals . . . . .	27
2.2	Anonymity . . . . .	28
2.2.1	Linkability . . . . .	29
2.2.2	Traceability . . . . .	30
2.3	Authentication . . . . .	31
<b>3</b>	<b>Design Decisions</b>	<b>33</b>
3.1	Current iMOAT Environment . . . . .	33
3.2	Design Strategy . . . . .	34
3.2.1	Interface . . . . .	34
3.2.2	Authentication and Access Management . . . . .	35
3.2.3	Query Tool . . . . .	38
<b>4</b>	<b>FREiMOAT</b>	<b>41</b>

4.1	Interface Design . . . . .	41
4.1.1	FREiMOAT Homepage . . . . .	42
4.1.2	Extended Information Page . . . . .	42
4.1.3	Request Access Page . . . . .	43
4.1.4	User Home Page . . . . .	43
4.1.5	Request School Access Page . . . . .	43
4.1.6	Update User Contact Info Page . . . . .	44
4.1.7	Query by School Page . . . . .	44
4.1.8	General Query Page . . . . .	45
4.1.9	Create New User Page . . . . .	46
4.1.10	Grant User Access Page . . . . .	46
4.1.11	Materials Access Page . . . . .	47
4.2	Authentication & Access Management Design . . . . .	47
4.2.1	Authentication Design . . . . .	47
4.2.2	Access Management Design . . . . .	48
4.3	Query Tool Design . . . . .	50
<b>5</b>	<b>Conclusion</b>	<b>53</b>
5.1	Potential Issues . . . . .	53
5.1.1	Scaling . . . . .	53
5.1.2	Interface Design . . . . .	54
5.1.3	Evolution of Use . . . . .	55
5.2	Future Work . . . . .	55
5.3	Final Thoughts . . . . .	56



# List of Figures

1-1	Version 1 iMOAT Architecture Diagram . . . . .	17
1-2	LSU version of Student Login Page . . . . .	18
1-3	MIT version of Student Login Page . . . . .	19
1-4	Version 1.0 iMOAT Database Schema . . . . .	21
1-5	Version 1 iMOAT Architecture Diagram . . . . .	23
1-6	Version 2.0: Test Group Table Interaction . . . . .	24
1-7	Version 2.0: Exam Student - Student Table Interaction . . . . .	25
3-1	iMOAT.net three-tier Architecture . . . . .	34
3-2	FREiMOAT Database Schema . . . . .	38
4-1	FREiMOAT Page Flow Diagram . . . . .	42
4-2	Additional Database Tables for FREiMOAT . . . . .	49



# List of Tables

1.1	Current host strings recognized by iMOAT.net . . . . .	20
4.1	Access Levels in iMOAT.net . . . . .	48
4.2	Access Levels in FREiMOAT . . . . .	49



# Chapter 1

## Introduction, Purpose and Background Material

### 1.1 Introduction to FREiMOAT

In this document, I present FREiMOAT: The Free Research Extension to the iCampus-MIT Online Assessment Tool. This project was developed under the coordination of iCampus (the collaborative educational partnership between MIT and Microsoft Research), and the MIT Office of Writing Across the Curriculum.

FREiMOAT has been designed as an extension of the iCampus-MIT Online Assessment Tool (iMOAT.net). iMOAT.net was developed under the supervision of Dr. Leslie Perelman, Director of Writing Across the Curriculum, and constructed over the past two years by Kenneth Walker (M.Eng MIT 02), Chis Felknor (B.M. University of Southern California) and myself. iMOAT is an assessment tool used by a consortium of schools across the country.

First, I will discuss my motivations for creating FREiMOAT as well as present necessary background material on iMOAT.net. In chapter 2, I will present the design goals that govern the design of FREiMOAT as well as present a summary of my research into anonymity. In chapter 3 the current iMOAT environment will be discussed as it pertains to FREiMOAT's design. Finally, in chapter 4, I present my design for FREiMOAT.

## 1.2 Purpose of FREiMOAT

It is important to develop a research interface for iMOAT. As the iMOAT project continues, the database will grow to contain tens of thousands of essays from students from around the country. It would be of great benefit for educators all around the world to be able to access this database for research. Queries into the dataset could aid in determining the average writing levels of all first year college students, vocabulary ranges, or location biases in results. The data contained in iMOAT could be used to judge the validity of the college boards writing exams (some of which are now largely computer-graded). The research possibilities are enormous.

The development of a project such as FREiMOAT has been foreseen for some time now and iMOAT.net has been constructed with FREiMOAT in mind. For example, iMOAT.net already has an informed consent question. This question asks students if they would allow their essays to be used anonymously for research. The specific wording is...

I do hereby give my university and its partner institutions in online essay evaluations permission to use anonymous versions of my writing for purposes of test development and teacher training. Furthermore, I give permission for anonymous versions of my writing to be used for scholarly research and for short excerpts of my text to be quoted in research publications.

This release is sufficient so that the data obtained can be used for research purposes.

Another reason to develop FREiMOAT is that educators at universities working with iMOAT have also requested to have access to the questions/readings from other schools in the consortium. Access to each other's materials is important so that schools might evaluate and improve their own assessments. Schools could allow access to reading materials and questions. They could also learn best practices from each other's successes and failures. Furthermore, schools could compare the writing levels of their incoming students to those of other schools as well as the writing level required by other schools in their assessment.

The complexity of the access and rights management that must be associated with the data make FREiMOAT an interesting and challenging project. The idea of access management conflicts with the goal of open or free use of this data. On the one hand, the purpose of FREiMOAT is to share, as openly and flexibly as possible, all the information contained in the database. On the other hand, it is the duty of the site to restrict access to the information and to allow only people who require that data, have authenticated themselves with the system, and have been granted access to access that data. The iMOAT database has significant potential for research and collaboration, which, right now, is inaccessible.

### **1.3 Background of iMOAT.net**

iMOAT was conceived in 2000 as a replacement for the Online Freshman Essay Evaluation web site. The Online Freshman Essay Evaluation web site was launched in 1998 in response to criticism of the traditional writing placement exam. Students complained that the traditional exam, which was administered to every incoming freshman during orientation week, did not accurately gauge their writing levels. They claimed that orientation was enormously distracting, and also that a two hour hand-written writing assessment was nothing like students' "normal" method of writing. Normally, writing tests allowed more time for critical thought and were mainly composed using word-processors. Accordingly, MIT set out to create a method for assessment that more accurately emulated real world writing environments.

In 1998, the Freshman Essay Evaluation web site (built by Roberta Crumrine of the Student Services Information Technology group) was launched by the Registrar's office and was an immediate success. Within two years, over 75 percent of incoming students elected to take the assessment on-line. Student feedback from the program was overwhelmingly positive. They really appreciated the ability to take time to work on their writing and complemented the program highly. The number of contested placements fell dramatically; from around 30 complaints about scoring, to the current level of one or two a year. Advisors also praised the system since it allowed them

access to the actual students essays. The success of the program pushed for expansion and in 2001, Dr. Leslie Perelman began the groundwork for iMOAT.net.

The ability to assess accurately the writing levels of incoming students has been a very hot topic among educators for some time. Dr. Perelman saw this as an opportunity to change the way that these types of assessments were done at our nation's colleges and universities.

iMOAT.net, as a collaborative project between MIT iCampus and the Office of Writing Across the Curriculum, was to be designed as a web service that could be shared with other universities. The service was very distinctive as other on-line testing and assessment tools were not geared towards longer essays and few, if any, allowed for human grading. While Kenneth Walker (M. Engineering MIT '02) spent his Research Associateship at MIT designing and coding the service, Dr. Perelman spoke at education symposiums across the country in his attempt to create an iMOAT consortium.

The iMOAT consortium was created in early 2002 and five schools beta tested the service over the summer. California Institute of Technology, Depaul University, Louisiana State University, the University of Cincinnati, and MIT made up the consortium in 2002. The initial system, though it has gone through some growing pains since its inception, has processed over 3,500 essays during the first summer. This year the consortium has expanded to include Cornell University, Oakland University, Clemson University, the Olin College of Engineering, and the State University of New York at Stony Brook. Already this summer, iMOAT has processed over 6,000 essays. While the project has been a success, there have been a number of enhancements that have been requested to the service.

In October 2002, the directors of the writing requirement of the five initial members of iMOAT met at MIT. During the three-days consortium, the members discussed the strengths and weaknesses of the current site. Among the long list of requested enhancements was the ability to share material between schools. However, due to time constraints and the limited number of changes that could be enacted within those constraints, this ability was not developed in version 2.0 of iMOAT.net. It was



this request that sparked the idea for this thesis.

As a result, FREiMOAT, the Free Research Extension to the iCampus MIT Online Assessment Tool, was born. If the members of the consortium wanted a method to exchange information on their assessments, then why wouldn't other researchers want access to this data as well? iMOAT.net, from the very beginning, was designed to be a large repository of student essays and has become a valuable tool for research.

Over the last few months, I have completed the code for version 2.0 of iMOAT and have been researching designs for FREiMOAT. However, to understand fully the design of FREiMOAT, it is necessary to understand the design of iMOAT.net.

### 1.3.1 Version 1.0 iMOAT.net architecture overview

The Version 1.0 iMOAT site can be best described with the architecture diagram found in Figure 1-1.

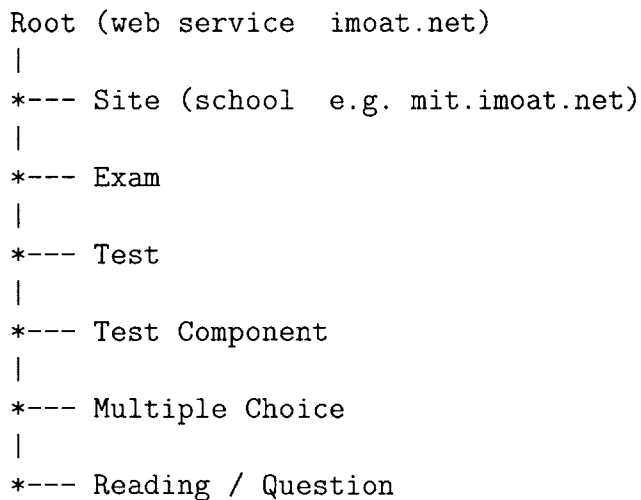


Figure 1-1: Version 1 iMOAT Architecture Diagram

As depicted in Figure 1-1, the web service (imoat.net) allows for the creation of any number of sites. It is a site that is assigned to a given school or university. Due to the Domain Name Server (DNS) configuration, all host names that end in "imoat.net" are directed to the web server. This enables the dynamic code on the server to look at the host string in a given request and format the presentation of the

site accordingly. Figures 1-2 and 1-3 are two examples of how iMOAT can present the same information on different sites.

Home > Department > Programs > Undergraduate > Firstyear Writing

**LSU ENGLISH**

**Essay Evaluation - Student Login**

*Please Log-in first!*

Login:

Password:

[Forgot Your Password?](#)

Login

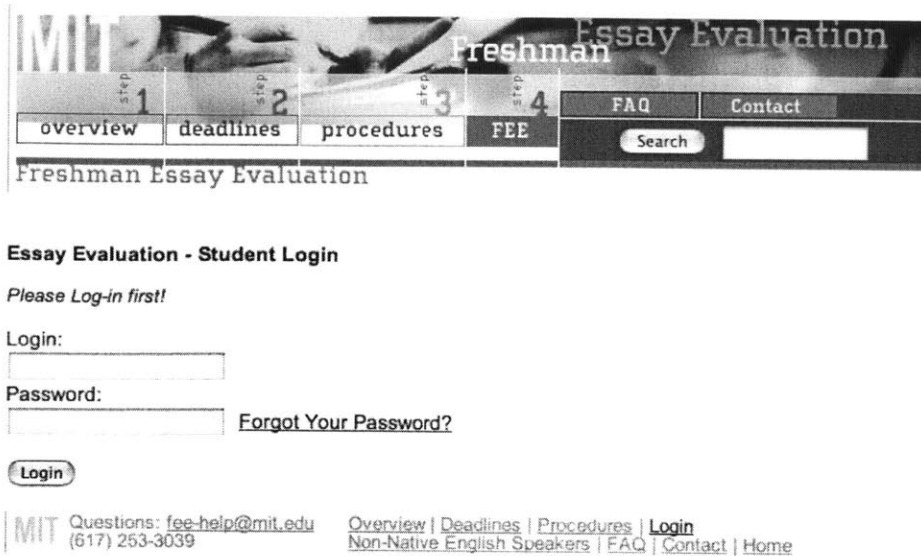
For questions  
Phone: 225.578.5923 or 225.802.3244  
or email: On-Line Challenge  
Firstyear Writing  
260 Allen Hall  
Louisiana State University  
Baton Rouge, 70803

iMOAT: iCampus-MIT Online Assessment Tool developed by MIT iCampus, Microsoft, & Jordan Alperin

Figure 1-2: LSU version of Student Login Page

At the Site level in the architecture, the SiteAdministrator can change the site's name as well as alter the appearance of iMOAT by either specifying the font and a few colors in the "Quick Style" section or by specifying their own style sheet. A site's appearance can also be customized by adding custom headers and footers. Also managed at the site level in the architecture is the system audit. Each site keeps a specific log of all noteworthy events that occur, enabling the SiteAdministrators to monitor all activity on their own sites. This audit information has been very helpful in diagnosing problems that students have had using iMOAT.

At the Exam level, the nomenclature that describes the functions of the site tends to stray a bit. An Exam is really what is traditionally thought of as a Class (the naming has been corrected in version 2.0, to be clearer). Thus, an Exam is an assessment name (what the students see when they log in), a collection of eligible students, a collection of Tests, and a Results Set. The specific definition of an Exam



IMOAT: iCampus-MIT Online Assessment Tool developed by MIT iCampus, Microsoft, & Jordan Alperin

Figure 1-3: MIT version of Student Login Page

will be made clearer when a Test is defined.

A Test is a collection of Test Components and is where Readers (Readers score student essays) and Commenters (Commenters leave comments on a student's Test performance) are assigned. Tests are defined by the date that the registration becomes open, the date that registration closes, the date that the questions become visible to students and by the date by when the submissions must be received. Other optional dates include when readings are available and when multiple-choice questions become available. It is at the Test level that Readers and Commenters are assigned to the Test Component that they are responsible for grading/commenting.

Because iMOAT.net was originally designed to emulate the online essay evaluation at MIT, a Test is really an instance of an Exam. This means that a student eligible for any Test under a given Exam may only participate in one Test, and that that Test will fulfill the student's Exam level requirement. This point is best illustrated by example.

At MIT, the Exam (The 2003 Freshman Essay Evaluation), contains three Tests: June, July, and Makeup. Students (the incoming 2003 freshman class), are eligible for

<i>Site Description</i>	<i>Host String</i>
<i>MIT</i>	<i>mit.imoat.net</i>
<i>TestWrite (Test Site)</i>	<i>www.imoat.net</i>
<i>CIT</i>	<i>caltech.imoat.net</i>
<i>Louisiana State University</i>	<i>lsu.imoat.net</i>
<i>Depaul University</i>	<i>depaul.imoat.net</i>
<i>University of Cincinnati</i>	<i>uc.imoat.net</i>
<i>iMOAT Demo</i>	<i>demo.imoat.net</i>
<i>MIT Demo</i>	<i>demo.mit.imoat.net</i>
<i>MIT - Graduate Site</i>	<i>grad.mit.imoat.net</i>
<i>Cornell University</i>	<i>cornell.imoat.net</i>
<i>Oakland University</i>	<i>oakland.imoat.net</i>
<i>Clemson University</i>	<i>clemson.imoat.net</i>
<i>Olin College of Engineering</i>	<i>olin.imoat.net</i>
<i>SUNY at Stony Brook</i>	<i>sunysb.imoat.net</i>

Table 1.1: Current host strings recognized by iMOAT.net

all three Tests. However, they are only allowed to register for, and therefore take, one of these Tests. Thus a student, who takes the June Test, can fulfill the same Exam level requirement as a student who decides to take the July Test. It is important to note that, although the June and July Tests satisfy the same Exam level requirement, they are separate Tests, which may or may not contain different readings, questions, and/or multiple-choice sections.

With a definition of a Test in mind, we now can define an Exam more accurately. An Exam is a collection of Tests that can be taken only exclusively, and that can satisfy the same Exam requirement.

The next level in the architecture is Test Components. Test Components are either Multiple-Choice problem sets or an essay question. A Test can contain any number of test components, allowing for a wide variety in testing options.

Multiple-Choice components can contain any number of questions, and up to six possible choices for the answer to each question. While the option to include a multiple choice test component is available, this feature has not been used extensively in iMOAT. Multiple-Choice Test Components were envisioned as complements to the essay portion of the assessment that could be used to gauge reading comprehension, or to collect more specific student data.

An Essay Test Component contains the questions/prompts that the student will be given and any readings (if desired) that are associated. Essay Test Components are very flexible in their use. These components have an optional or suggested minimum and maximum word count (the site will only warn a student that they are outside the given range and will still accept any essay length) that allows the Essay Test Components to be used for both short answer and essay length responses.

On paper, the iMOAT version 1.0 architecture is very strong, allowing for a wide range of customizations. However, the implementation of this architecture in the Database limited the growth potential of iMOAT.

### 1.3.2 Issues with version 1 architecture/implementation

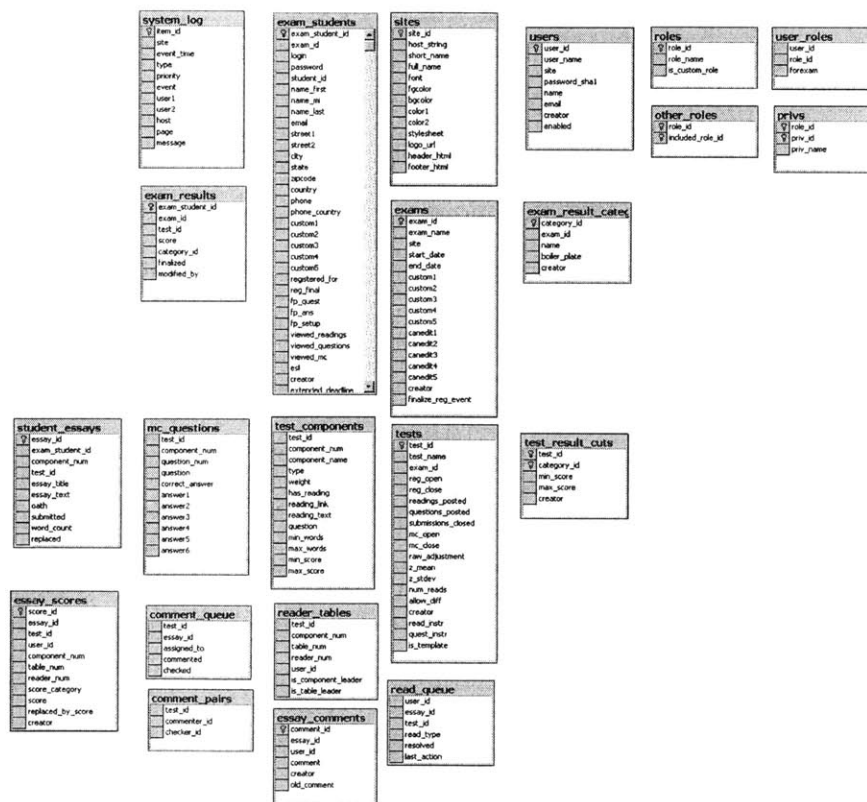


Figure 1-4: Version 1.0 iMOAT Database Schema

As can be seen from the Database Diagram (see figure 1-4), iMOAT stores quite a bit of information about a student's state in the exam\_students table. It is this table

that had the greatest effect on the growth of iMOAT. The exam\_students table contains all the data about a specific student, as well as all the data about that student's progress through an exam. The columns that will limit the growth of the iMOAT system are the viewed\_questions, viewed\_readings, viewed\_mc, extended\_deadline, release\_essays, and the registered\_for fields, since these fields describe the state of a student's progress in a specific test and are not "general information" about a student.

The consequence of joining student data and test state data into one table is that a given student can participate only in one iMOAT Test, ever. One method for getting around the one test per student limit would be to create a new, yet identical student, and allow the new student to participate in another test. This method is confusing, however, because each student in a given site must have a unique login. Each instance of a given student, to allow one student to take multiple tests, has to have a different login. Imagine if a school wanted to give five exams to the same student body over the period of their undergraduate schooling. Each student would need five unique logins for iMOAT. While creating five unique logins per student might be tolerable at a school with a very small student body, it creates an administrative nightmare for larger schools. Obviously, this solution is not desired, and the work-around can only be used temporarily.

Another issue with version 1 of iMOAT was that dates and times were fixed in the server time zone. While having only one time zone represented on the service eased its administration, it created unnecessary difficulty for those schools that wished to use the service in different time zones. The most common difficulty expressed by schools using version 1 of iMOAT was that specific times for the Test level events could not be set. The consequence of this was that events that happened at midnight in the Eastern time zone, would occur at 9pm (3 hours earlier) in the Pacific time zone.

Version 2.0 of the service has been designed to address these constraints.

```

Root (web service imoat.net)
|
*--- Site (school e.g. mit.imoat.net)
|
*--- Class
|
*--- Test Group
|
*--- Test
|
*--- Test Component
|
*--- Multiple Choice
|
*--- Reading / Question

```

Figure 1-5: Version 1 iMOAT Architecture Diagram

### 1.3.3 Version 2.0 Architecture

The version 2.0 iMOAT architecture departs significantly from the old architecture. As can be seen in Figure 1-5, almost an entirely new naming convention has been implemented.

First let us look at what has not changed. Tests, Test Components, Multiple-Choice Components, and Essay Components largely have remained the same. The ability to specify times as well as dates at the Test level has been added. Also, a test emails object has been created to allow for the administration of emails sent by a given test. These emails are triggered on specific test events: registration open, readings available, questions available, multiple-choice available, submission received, and results available.

The Site level has also been slightly modified to enable the setting of the time zone for a given site. The time zone setting is reflected at every location in the service that presents a date or time.

Now we get to the major changes. As alluded to in *Version 1.0 iMOAT.net architecture overview*, the old Exam level was a misleading name that could be easily confused with the Test level. To correct the nomenclature, the Exam Level has been

renamed Class. The change to Class should be very easy for educators to understand, as it represents a student group, a collection of Test Groups, and a Results Set, which resembles a real world class. The word "class" is also flexible as it can represent either a specific subject and term (e.g. Technical Writing Spring 2003), or a class of students (e.g. the class of 2002). The Class level has the ability to represent both of these interpretations.

To enable iMOAT to track a student over the course of many tests, the Test Group level has been added to the architecture. This change necessitated modifications to the version 1.0 database schema depicted in Figure 1-4. These changes are illustrated in Figures 1-6 and 1-7

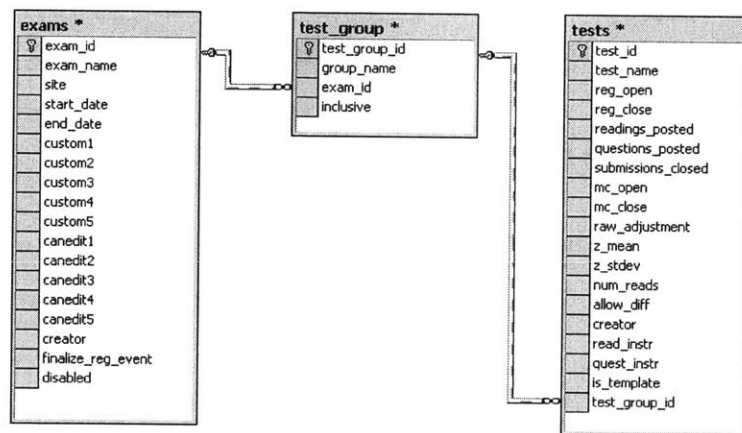


Figure 1-6: Version 2.0: Test Group Table Interaction

The test\_group table has been added in version 2.0, in between the old exams (now Class Level in the architecture) and tests tables. Previously, the completion of any single Test was sufficient to satisfy an Exam level requirement. With the addition of the new test\_group table now a student must satisfy all Test Groups to complete a Class Level requirement. As in version 1.0, any Test will satisfy a Test Group Level requirement. For example, for a student to complete the Class "Technical Writing Spring 2002", which contains three Test Groups ("Memo", "Proposal", and "Final Design Document"), the student would need to complete one Test in each separate test group. One reason multiple Tests might be available under each Test Group is to accommodate student's schedules. The "Memo" Test Group might contain Tests



”first week of February” and ”second week of February” for this purpose. The new design allows for a wide range of configurations that can represent almost any real-world testing schedule.

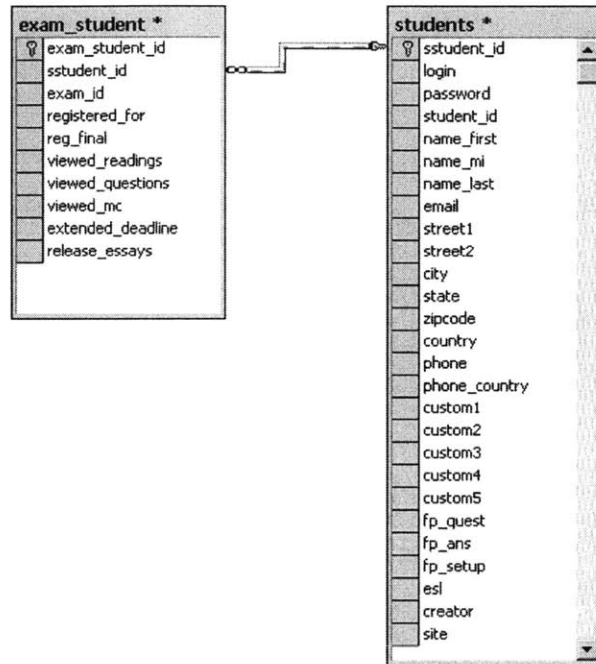


Figure 1-7: Version 2.0: Exam Student - Student Table Interaction

The exam\_students table also needed to be modified so that students could participate in more than one Test. As can be seen in Figure 1-7, the version 1.0 exam\_students table has been split into two separate tables: exam\_student and students. The new exam\_student table now contains all of the information that is specific to a student’s interaction with a given Test while the students table contains all information that is static describing a given student. The addition of the sstudent\_id field (a.k.a. Site Student id) provides the interaction between these two tables, making it possible for a single entry in the students table to link to many entries in the exam\_student table.

While there are many more changes between version 1.0 of iMOAT.net and version 2.0, it is only necessary to understand these architecture changes so that FREiMOAT can be presented.



# Chapter 2

## Design Goals and Research

It is important for any project to have a set of goals to govern its design and development, since having goals enables developers to gauge the success or failure of their design. In this chapter, I present the goals that I have laid out for the design of FREiMOAT, explain the motivations for those goals, and present a summary of the research that was done to support these goals.

### 2.1 Design Goals

1. Maintain data anonymity

It is very important that we do not compromise the rights of the authors of the essays in the database. In the release to which students can agree, (see section 1.2), it is specifically stated that all of their material will be presented anonymously if used for research purposes. Therefore, all data that is shared through FREiMOAT must be stripped of all identifiable, linkable, and traceable information.

2. Allow only authenticated access

Federal law prohibits (the Federal Educational Rights and Privacy Act of 1972) the unauthorized or public distribution of individual student data. The data contained in FREiMOAT's database is sensitive and should not be freely avail-

able to the public. It will be necessary to authenticate users of FREiMOAT to restrict and monitor access. Authentication is a very difficult task when the people who are being authenticated have never been met personally. Rendezvous, the necessary component in any authentication system, may or may not be possible with FREiMOAT (depending on the user). Requiring our users to authenticate themselves for access to the system will aid FREiMOAT so that it might grow and evolve with its use. Feedback from users and a system audit will be the best methods for re-evaluating the design of FREiMOAT.

### 3. Not interfere with the use of the iMOAT service

While FREiMOAT is a very important component of the iMOAT service, it is secondary to the main site's operations. Especially in the early stages of iMOAT.net when the entire service runs on a single server, computer resources will be very scarce. FREiMOAT must never interfere with the operation of the main site.

### 4. Keep data access as open and flexible as possible while maintaining other goals

From the above goals it appears as if the data in iMOAT should be locked down as tightly as possible. However, it is important to remember that the final goal of this service is to share information, and it must be designed to do that as openly and flexibly as possible.

## 2.2 Anonymity

Anonymity in the context of data sharing is a very grey term. Dictionary.com's definition of anonymity is as follows.

anonymity *noun*

1. The quality or state of being unknown or unacknowledged.
2. One that is unknown or unacknowledged.

This definition is very clear: you are either known or unknown. It does not take into account degrees of anonymity. Data that the FREiMOAT system will present to researchers and administrators will never be fully anonymous. That is to say that every essay that is retrieved from iMOAT's database is known to have been written by someone affiliated with iMOAT. While it is a large number of people, it is still a knowable size group.

Anonymity is really a scale from 0 to 1 of how "identifiable" an individual is. If absolutely nothing is known, this represents an anonymity level of 0; perfect anonymity. Once something has been identified uniquely, the level is 1; providing no anonymity. Levels of anonymity between 0 and 1 are expressed as a ratio from 1 (e.g. 1/2 or 1/512) with the denominator being the size of the group that the data could represent. This is a very important representation as much of FREiMOAT's design depends on it.

While Dictionary.com's black-and-white definition is easily applicable to individual pieces of data, such as a birth date, once data pieces are grouped the anonymity of the whole is almost always less than that of the individual pieces. A birth date or a zip code alone might be considered highly anonymous, but the combination of only these two data might identify an individual. Even if birth date and zip code are not enough, these data do describe a set of individuals. Depending on the zip code involved, this set could be very large ( $> 1000$ ) or intimately small. The size of the set described by data is the level of anonymity that is provided to a given individual. Throughout the rest of this paper, the size of a described data set will be called the "bin" size.

### **2.2.1 Linkability**

Older anonymity systems worked by stripping or replacing all data that was flagged as "identifying". In a patient record for a hospital visit, the data collected might have been Name, Address, Age, Sex, Insurance Provider, Date of Visit, Name of Physician, Observations, and Diagnosis. In this simplified system, stripping the Name and Address fields might make the rest of the data seem anonymous because it does

not directly reference a specific individual. What this system of anonymity neglects to acknowledge is the linkability of this data.

Linkability is the ability for one data set to be combined with other data sets to lower the level of anonymity. Returning to our previous example, what if the Age, Sex, Insurance provider, Date of Visit, Name of Physician, Observations, and Diagnosis data were linked to the address of that physician's office, linked again to a map of local zip codes, and further linked to a list of birth records for those zip codes? When combined with other data sets, the "anonymous" data suddenly provides very little anonymity.

Linkability is very difficult to prevent. The simplest way to address it is to present only data that it known not to exist elsewhere. If our repository is the only location where such data exist, then there is nothing else to link them to. However, doing so will limit severely the research potential for the data in iMOAT. To deny any access to fields such as "zip code" or "state" makes it impossible to do any regression based on location. What also makes linkability hard to protect against is that it is impossible for FREiMOAT to know the data sets to which its users have access. For the purpose of FREiMOAT's design, we must assume that users have access to all publicly available data sets. This includes data from government censuses, the Social Security Database, the phone book, and other repositories.

We must also consider the benefits of linkability. The information contained within iMOAT would be quite limited if it could not be linked to other data. Linkability in itself is not a bad thing. Only when linkability compromises the anonymity of the data should we limit the data that iMOAT can produce.

### **2.2.2 Traceability**

Traceability is another method by which anonymity can be compromised. Traceability is the ability to watch where data is collected from before it is sent to the user, and use the information about data location to gain more insight into the data received.

Traceability is normally a problem for distributed systems. If the computer system that receives a request needs to contact another computer system to satisfy that

request, then the possibility for a trace to be used exists. This risk is intensified even more if the secondary request is made to a computer that is vulnerable to hacking.

Due to the fact that the current incarnation of iMOAT.net resides on a single computer, and that all of the information that iMOAT uses resides in a single database, traceability is of little concern in the design of FREiMOAT. However, as iMOAT.net grows to span multiple computers, possibly on different networks, traceability may become a larger issue.

## 2.3 Authentication

Authentication is any method by which a person's identity is confirmed. Normally, authentication requires some sort of a rendezvous of information between two parties. Examples of this rendezvous in normal life are items like drivers' licenses and passports. These are documents that most authorities trust as a proper method of identification because they are widely known and standardized. While this paper type of authentication works well in the real world, computer-based authentication is more complicated.

The digital revolution has simplified the replication and distribution of any given document. The simplicity and accuracy of copying documents makes it difficult to verify a person's identity based on the sharing of a common document, as there is very little to prevent impersonation. In order to authenticate digitally, encryption based digital signing is the standard way to provide authenticity. Long digital key pairs are used to both encrypt and sign media.

The use of digital key pairs allows for secure and authenticated communication over the internet. Key pairs are issued to an individual from a common trusted authority so that the receipt of a signed message can contact that common authority to verify the identity of the sender. In this way a rendezvous is achieved in the digital world. The digital certificate authority is able to verify the identity that is the owner of that particular digital key, and since the authority is a "trusted" source, the recipient can be confident that the identity is real. However, the system is not

flawless and identity theft is still possible.

Current methods of digital authentication, just as in real-world authentication, are subject to a number of attacks. While the current algorithms that are used for encryption and authentication are believed to be secure, on occasion a vulnerability is discovered that compromises huge numbers of digital identities. Also, as with most any authentication mechanism, the greatest danger is human error. Authentication schemes are based on the fundamental idea that a given individual wants to keep their identity private. However, sometimes, this is not always the case.

One method of giving more than one person access to items that require authentication is simply for a single previously-authenticated user to distribute their digital keys to others, effectively allowing them to impersonate the authenticated user. This is the same as giving your passport to someone else to let them impersonate you. While this type of impersonation might not work well in the real world (due to photographs on ID cards, new biometric retinal scans and fingerprint-based authentication devices), there is virtually no way to prevent this type of an attack in the virtual world.

If a person chooses to share their digital identity, impersonation is very simple. It is also quite important to note that more often than not, the "sharing" of a digital identity is not done willingly. Identity theft is one of the fastest-growing crimes in America today and the fact that more and more critical systems use digital authentication is only making this crime easier.

Authentication in the digital world is only as trustworthy as the users that support the system. If there is great trust in users to keep their digital keys secure, then the system works. If users are sloppy and allow their keys to disseminate, the system fails. This shifts the truly important step in creating an authentication scheme to the initial trust established when keys are issued. Repeat authentication (that is, after the first) is simple once a high level of trust has been achieved.



# Chapter 3

## Design Decisions

Even before FREiMOAT can be designed, it is important to make certain design decisions that will govern the tool's actual development. In this chapter, I will discuss the current iMOAT environment and how it affects FREiMOATs development, as well as frame my method for designing FREiMOAT.

### 3.1 Current iMOAT Environment

Since FREiMOAT is an extension of iMOAT, the current iMOAT system architecture has a large effect on FREiMOATs design.

Because iMOAT was built as a collaboration between Microsoft and MIT, the system has been constructed on top of Microsoft's .NET architecture. Accordingly, the application is of a three tier architecture (see Figure 3-1). HTML and ASP scripts make up the presentation layer. Compiled C# code runs the application logic and a SQL server 2000 database stores the system data. This is a typical Microsoft three-tier design.

Currently the computer that runs iMOAT is hosted in Somerville, Massachusetts by Hosting.com. Hosting.com is the Internet division of Allegiance Telecom. The machine used to run the application is an IBM Netfinity Pentium 3 1266mhz with one gigabyte of RAM. While the specifications for this machine are impressive compared to a current desktop machine, it is important to note that all three layers of the

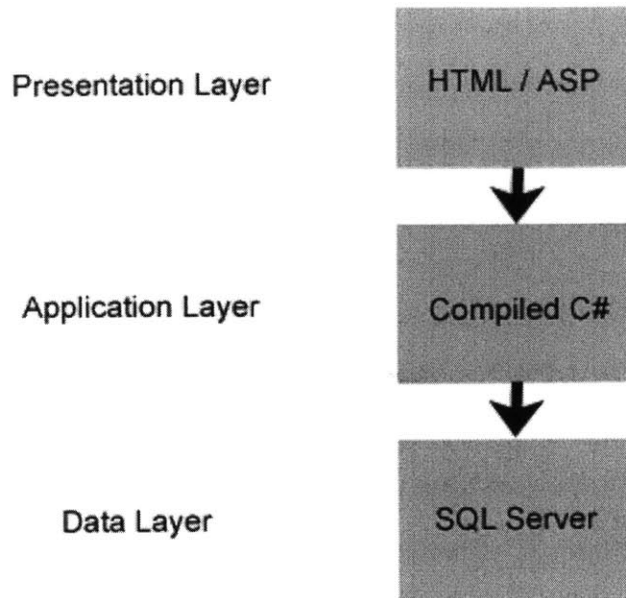


Figure 3-1: iMOAT.net three-tier Architecture

iMOAT.net application are served from the same machine. FREiMOAT must be designed for these limited resources.

## 3.2 Design Strategy

The design of FREiMOAT can be broken down into three main areas of inquiry: the design of the interface, the authentication and access management section, and the query tool.

### 3.2.1 Interface

First is the actual interface of the project. This interface will be responsible for the presentation of information about the service and the essays that it collects. One thing that the interface will display will be current statistics about the size of the iMOAT database in order to promote FREiMOATs use. The decision to make the interface web-based (not necessarily a web page) was made to allow for the greatest accessibility

(and automated access) of the data. However, in what form this web-based service will be available still needs to be considered. Should the interface be made up of web pages to allow for ease of presentation, or will an interface structured as pre-written web pages prove too restricting? Is a web service, like a .NET application, a better choice? To answer these, other questions must be answered first: how will the service be used and who will use it?

FREiMOAT has two main constituents: Administrators of iMOAT sites and independent researchers. Administrators of current iMOAT sites, as I have said, are interested in using FREiMOAT for collaborative purposes. Their main interest is in sharing assessment readings and questions so that they might learn from other schools what types of questions work best. The best feedback to determine how "good" a question is will be the student essays themselves, which will demonstrate whether or not most of the students understood the assignment.

Because I have no feedback from them, I will assume that independent researchers have equal desires to access all of the data. These data include readings, questions, essays, and demographical information.

Current iMOAT administrators are used to working with iMOAT. Since iMOAT has been built as a web site, the logical choice for FREiMOAT's design will also be a webpage. While a web service is much more flexible and faster to develop, the learning curve associated with using such a service is far too steep. We cannot expect administrative staff and humanities teachers to learn to program in order to access FREiMOAT. While there is no data about independent researchers' abilities to access FREiMOAT as a web service, we do not want to discourage FREiMOAT's use because of the difficulty of the interface. While this decision will limit the flexibility by which FREiMOAT can present information, accessibility is more important than flexibility.

### **3.2.2 Authentication and Access Management**

Second, and probably most important, is the access management section of the site. This section requires answering the most design questions.

First, what information is required of researchers and school administrators in

order to request access to the database? Once this information is submitted, who (or what) decides to grant access? Once access is allowed, what information can be accessed? Are there different levels of access for different researchers or is everything open to all? To help to answer these questions it will be necessary to determine who owns the data in the database. Clearly the students that use the site have consented to the use of their data but what about the schools themselves? Who owns the test questions and reading materials?

Over the past year, iMOAT has been used mainly for its essay components. The majority of these components have had readings along with the questions. These readings have come from a variety of sources, including news articles, famous writings, and even a few that were written specifically for the purpose of the assessment. Material that has been released for public use will be able to be shown on FREiMOAT without consent. But what about readings that have been written specifically for the assessment or unpublished texts from a given school's professors? Does FREiMOAT have the right to allow access to this information?

In order for access to be granted by FREiMOATs authentication system, there must be some sort of rendezvous between the FREiMOAT administrator and the person requesting access. In the case of current iMOAT administrators, this rendezvous has already occurred, either in a physical meeting or through telephone conversations during the iMOAT registration process. Other administrators at a school already associated with iMOAT can rendezvous by way of a person at that school who has already authenticated. This is what might be called the "grapevine method" of authentication. I know and trust someone who knows someone else who vouches for him or her - so I trust this new person. While this method works well for schools that already have access to iMOAT, what does this mean for independent researchers?

For an independent researcher to be granted access to FREiMOAT, it will be necessary for them to provide enough information that the FREiMOAT administrator can confirm so that they can be assumed to be trustworthy. This kind of authentication happens all the time on the Internet. Every time you type in your address, phone number, name, and email address you are providing information so that you

can be authenticated. While there is no rendezvous in this type of authentication, it is assumed that if the web site wanted to, they could attempt to verify that the information that you supplied is true. The information collected should also include a brief description as to why the independent researcher is interested in FREiMOAT. While this description is not important for authentication reasons, it will help the FREiMOAT administrator understand what researchers are interested in. This information can then be used in the design of the next version of FREiMOAT.

Now that users are authenticated, reauthentication can be as simple as a username and password, or as complex as certificates. MIT uses Kerberos, a certificate-based authentication mechanism to verify the identity of students and faculty members. While certificate-based authentication is very useful in MIT's case, the sensitivity of the data to which authentication grants access (registration, financial aid information, grade reports, etc.) makes certificates overkill for a project such as FREiMOAT. MIT spends a considerable amount of resources educating users about certificates, administering the use of these certificates, and controlling access to data based on these certificates. The resource expenditure needed to educate FREiMOAT users about certificates is prohibitory and certificate-based authentication makes it difficult to access a web site from multiple locations. Also noteworthy is that iMOAT uses username and password-based authentication, so requiring that FREiMOAT use certificates will simply create confusion. For these reasons FREiMOAT should use username and password authentication.

The next issue to consider is levels of access within FREiMOAT. Once a user has been authenticated into the system, to what should they have access? As already discussed, certain reading material is in the public domain, while some could require permission for access. One simple solution would be to require that all material posted by schools be placed in the public domain. However, this violates one of my design goals - not to interfere with the operation of iMOAT. It is necessary for the school to decide whether the information that they use should be accessible, so there must be an authentication level between each schools information.

### 3.2.3 Query Tool

Lastly, a query management tool will allow researchers to run a limited set of SQL queries on the database. Because unrestricted SQL access to the database is a large security risk, this tool must be designed very carefully - the tool will be allowed only to run SELECT queries on the iMOAT services section of the DB to avoid any changes to the iMOAT data. But should researches be allowed some space to store previous queries, or space to store temporary results? The tool also must be able to recognize queries that have the potential to disrupt the main site's use. This could include queries that use too much of the server's memory or simply take too long to process. Perhaps the management tool will have to schedule long queries during off-peak hours.

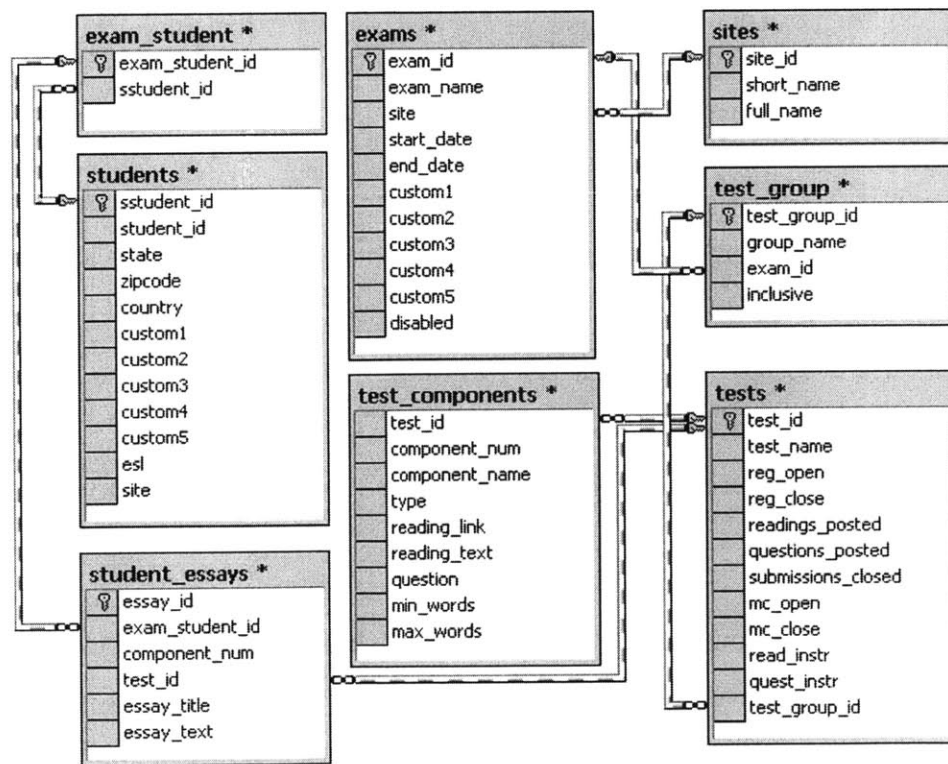


Figure 3-2: FREiMOAT Database Schema

The query tool is also responsible for presenting the data in iMOAT's database anonymously. This tool will have to restrict access to all database fields that contain identifying information, as well as lower the granularity of some data fields so that

some location information can be displayed. Only a limited set of data fields from the schema in figure 1-4 should be available on FREiMOAT. Figure 3-2 shows the portion of the schema that should be available through FREiMOAT.

As can be seen in Figure 3-2, almost all of the student data has been restricted. The only fields that are left are the state, zip code, country, the custom fields, and the esl fields. The zip code field should be modified further so that it only returns the first three or four digits of the code with the rest set to zeros. This will enable FREiMOAT to preserve some of the locality data while still providing anonymity.

Also important will be the ability to only return data that have a bin size large enough to be considered anonymous. This means that for all queries that are run on the site there must be some number of results (larger than 1) that could represent a given individual. Bin size will have to be an experimentally determined number. At the beginning of FREiMOAT, while there are a small number of essays in the database, it may be difficult to produce unlinkable results, but, as the database grows, this will get easier and easier.





# Chapter 4

## FREiMOAT

The Query tool, the access management component, and the user interface will all come together to make FREiMOAT possible. In this chapter, I propose a design for each component.

### 4.1 Interface Design

From the previous chapter, we know that the interface for FREiMOAT will be made up of web pages using the .NET architecture. We also know that the following services must be provided on these web pages.

1. Current database size statistics
2. Username and password reauthentication
3. Request for Authorization form
4. About FREiMOAT text

Based on these requirements, I propose Figure 4-1 for the page flow diagram for FREiMOAT.

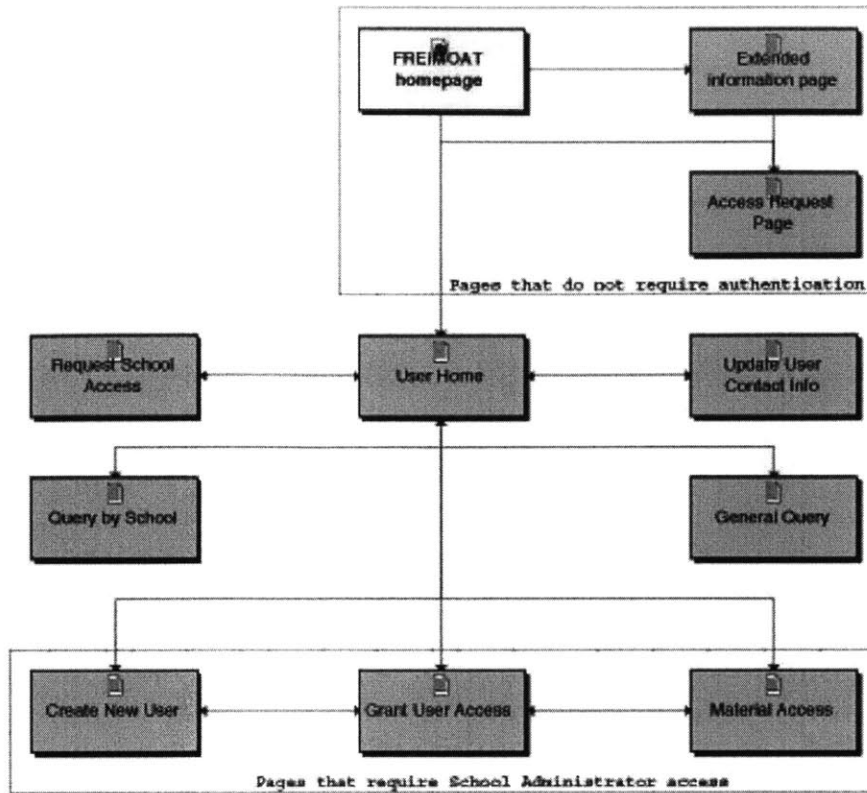


Figure 4-1: FREiMOAT Page Flow Diagram

### 4.1.1 FREiMOAT Homepage

The FREiMOAT homepage will be responsible for a number of tasks. First this page will give a brief introduction to FREiMOAT - its purpose and connection to iMOAT. There also should be a link to iMOAT and iMOAT's "about" page for more information on iMOAT. The homepage will also contain the login section of the site that will direct users to the user home page. This page will also have a link to email the FREiMOAT administrator to answer any questions not already answered in the online material.

### 4.1.2 Extended Information Page

The Extended Information Page is responsible for answering most questions about FREiMOAT. It is this page that will present current statistics about the number of essays and number of schools using iMOAT. These statistics should include general

data about student essays, such as the average word count. This page should also serve as a frequently-asked-questions-about-FREiMOAT page. It should describe, briefly, the procedure used to request access to FREiMOAT, as well as, link to any research that has been done using information contained in FREiMOAT.

### **4.1.3 Request Access Page**

The Access Request Page is where independent researchers start their FREiMOAT experience. This page is a web form where researchers request access to FREiMOAT. Later, in section 4.2.1 I will discuss the choice of fields that researchers have to fill out. For now I can say that they will have to write a brief description as to why they want access to FREiMOAT. This page will also describe the authentication process - stating that their request will be read by a FREiMOAT administrator and then they will receive a reply.

### **4.1.4 User Home Page**

The User Home page is the center of the FREiMOAT experience. It is from this page that users can submit feedback on their use of FREiMOAT. This feedback could be as simple as a thank you note or an interface complaint or as complicated as requesting a query for which the web page interface does not allow. On this page users will be able to link to other user FREiMOAT pages. These links should be in a recognizable navigation section (such as a header and footer or a navigation bar). If the user is a School Administrator (explained in section 4.2.2) then they will have access to another navigation element that will link them to their special user pages. The School Administrator pages can be seen in Figure 4-1.

### **4.1.5 Request School Access Page**

Because FREiMOAT cannot tell automatically if the materials that a school uses for their assessments are in the public domain, FREiMOAT will control access to a given school's materials. The Request Access Page is where FREiMOAT users will

be able to request access to schools' materials. This page will contain a group select box (a select box that allows for more than one selection) that will allow users to request access to many schools at once. There will also be a text box where users can write a brief description about their research and state why they have requested this access. This page will then send an email to the school administrators of the requested materials, requesting access.

#### **4.1.6 Update User Contact Info Page**

The Update User Contact Info page will allow users to keep their contact information current, should it change during their use of FREiMOAT. The contact information used by FREiMOAT is very important because if there are any questions about why a level of access should be granted, a human administrator will need to contact the requesting party to clarify.

#### **4.1.7 Query by School Page**

The Query by School page will be the interface to collect student essays and assessment materials by school. Once a FREiMOAT user has been authorized to access a specific school's assessment materials, they will be able to collect data on a per-school level. This page will have two purposes for users: first, to download all materials that are allowed for distribution by a given school and second, to query the database for student essays and be able to retrieve the associated school's material that goes along with each essay.

Because this page performs a larger number of functions than other pages on the site, this page will have the most complex interface. In fact, it might be better to split into two or more pages to simplify the interface. If the user simply wants to collect data on school materials, they will first click a series of checkboxes indicating from which schools they want materials. Then the user will click a "get assessments" button. Now the page will display a list of assessments, test groups, and tests that fall under the above request. Once again a series of check boxes will be used to select

the desired materials; one box for each reading or question in each test component and one for each available custom field. Once the requested materials have been highlighted, the user can click the "download" button and FREiMOAT will collect the desired material into a text file and present the user a link to download the requested materials.

If the user wishes to collect student essays from this page, then FREiMOAT will present the same interface used on the General Query Page (described in section 4.1.8) after the top-level select-schools-checkboxes have been selected. Once the query is entered, FREiMOAT will once again collect the requested materials into a text file and direct the user to download the file.

#### **4.1.8 General Query Page**

The General Query page is where FREiMOAT users can download student essays, irrespective of school. The interface of this page is rather extensive because FREiMOAT has to try to allow for a large variance of queries with the limited interface of a web page. This page also needs to contain a long disclaimer about the use of the materials that can be retrieved. In the same way that FREiMOAT cannot determine if a school's materials should be kept private, the site cannot determine if the student has provided enough information to identify themselves within a given student essay. (There is nothing to prevent a student from saying "My name is John Smith and I live at 2222 Main St. Somewhere, USA" in their essays.) The disclaimer on this page should state that if any such essay is retrieved then the researcher has the responsibility not to use that material.

The interface will be broken up into sections. The first will be a series of checkboxes allowing the user to select if they want zipcode, state, country, and esl data to be included in their query. Due to the query tool, the more information that is requested in a single query, the lower the probability that there is enough data in the database to meet the required bin size to allow for anonymity. This page will have to describe this process to the user. The next section will allow for the query to have minimum and maximum word limits for the essays. The final section will be an open

text box to search for certain text in the essays or state, zipcode, and country fields. Once this information has been filled out, the user can click the "download results" button and FREiMOAT will collect the data into a text file and present a link to download it.

#### **4.1.9 Create New User Page**

The Create New User page can be used by either specific school administrators or by the FREiMOAT administrator. If the page is used by a School Administrator, the new user created will, by default, be granted access to that school's materials. This allows iMOAT users who wish to create a new assessment to look over older assessment materials. If the page is used by the FREiMOAT administrator then the page will give the option to give the user rights to access any number of schools. The FREiMOAT administrator can also create another FREiMOAT administrator by giving the user access to site "0" in the site\_access table (shown in Figure 4-2). To create a new user, the administrator needs to create a unique username and password along with a name and email address for the user.

#### **4.1.10 Grant User Access Page**

The Grant User Access page allows school administrators to view a list of researchers that are requesting access to their schools materials. This page will list the users name, email, phone, and written request so that the administrator can judge if the user deserves access. Each user will have a "grant" and "deny" button next to their information, as well as an expiration date that can be used to control access. If no date is specified in the "expires" field, then the user will have access for all time. If the user is granted access, FREiMOAT will send the user an email notifying them. If they are denied access, the school administrator has the option to write a short note as to why they were denied, possibly requesting more information.

### **4.1.11 Materials Access Page**

The Materials Access Page is where a school administrator can select which materials are in the public domain. These materials include the readings, questions, and the custom fields stored about a student. This page will list all materials for all assessments in a given school's site. Check boxes will allow the administrator to select which materials should be presented. The Page should also have check-all and a uncheck-all buttons.

## **4.2 Authentication & Access Management Design**

### **4.2.1 Authentication Design**

As discussed in chapter 3, current iMOAT administrators automatically will be given access to FREiMOAT because they are already authenticated. Independent Researchers will need to request access to FREiMOAT through the access request page. The information required for access will be name, address, phone, email, and a written request, describing the purpose of their research. FREiMOAT will then send an email to the FREiMOAT administrator who can look over the information submitted and contact the requesting researcher, if necessary, for more information. The FREiMOAT administrator will then be able to create a username and temporary password for the researcher and FREiMOAT will send that user an email with their login information.

The next step is for the researcher to login to the site and decide if they want to request to access data from specific schools. If they do not need school-specific data then they can proceed to download essays that match their search criteria. If they decide that they do want access to school specific data, they can request access from the request access page.

The Request Access page will contain a list of schools that currently use iMOAT as well as a list of the classes that these schools have completed. From this page, the researcher can check a checkbox next to a school that he or she wants access to,

and FREiMOAT will send a request notification email to that school’s administrator. When the administrator receives this email, which will contain a copy of the researchers requested text, they have to log in to FREiMOAT and grant access to this user. The access will have the option to expire at a specific date.

## 4.2.2 Access Management Design

Current users of FREiMOAT have many levels of access as can be seen in Figure 4.1.

<i>Access Level</i>	<i>Description</i>
SystemAdministrstor	<i>Read and write all sites (partitions); read write all code. Read all data. Assign permissions. Change passwords.</i>
SiteAdministrstor	<i>Ability to view site audits and configure site headers/footers/names/time zone. Create administrative users and set passwords</i>
ExamAdministrstor	<i>Define classes, groups, and tests. Add eligible students. Download student data.</i>
RegAdministrstor	<i>Add eligible students. Download student data.</i>
Supervisor	
CommentWriter	<i>Write student essay comments.</i>
Grader	<i>Grade student essays.</i>
ClericalStaff	<i>Input Student Essay grades and comments in Data Entry Section.</i>
Faculty1	<i>Custom Role</i>
Faculty2	<i>Custom Role</i>
Faculty3	<i>Custom Role</i>
Admin1	<i>Custom Role</i>
Admin2	<i>Custom Role</i>
Admin3	<i>Custom Role</i>
Student	<i>Edit biographical information. Register for classes. Submit essays.</i>

Table 4.1: Access Levels in iMOAT.net

All users with SiteAdministration access and above should be given automatic logins into FREiMOAT. This makes sense because each school will need a user that has the ability to create new FREiMOAT users and the SiteAdministrator has already been granted this power in iMOAT.

The levels of permissions under FREiMOAT will be as follows in Figure 4.2.



<i>Access Level</i>	<i>Description</i>
SchoolAdministrstor	<i>View all of own School Material. Create new users and set passwords. Control access to school material. View all material with granted rights</i>
SchoolUser	<i>View all of own school material. View all material with granted rights.</i>
User	<i>View all material with granted rights.</i>

Table 4.2: Access Levels in FREiMOAT

School Administrators will be responsible for controlling access to the readings and questions for their school because FREiMOAT cannot determine automatically if this information is in the public domain or not. The default setting will be not to share any of this material. The School Administrator user will have to enable the sharing of material though the material access page. On the material access page, this user simply will check a check box next to each test that they wish to grant access to the reading or question material. There should be two check boxes: one for the reading material and one for the question material. There will also be a checkbox to grant access to each of the custom fields that are available.

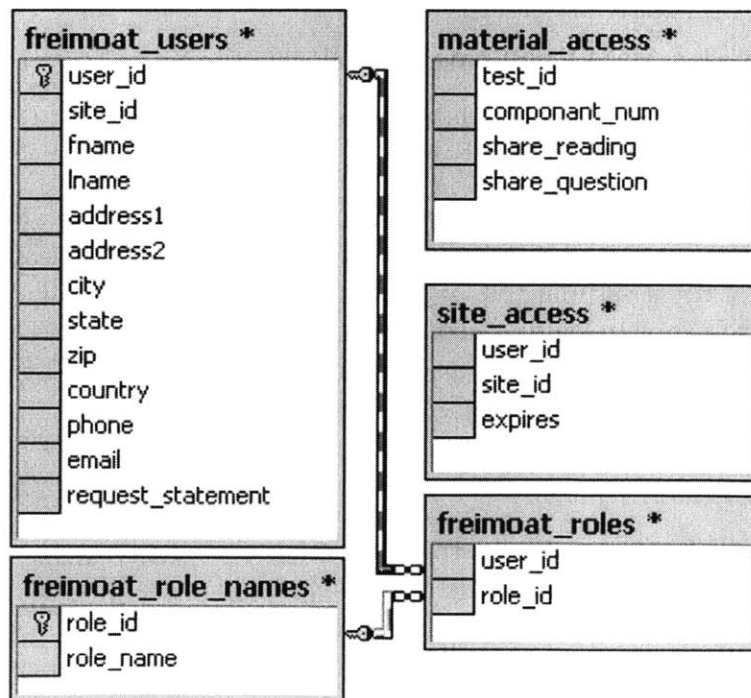


Figure 4-2: Additional Database Tables for FREiMOAT

The levels of access depicted in Table 4.2 will be stored in the `freimoat_role_names` table. Another option for storing this information could be to add a field called `role` into the `freimoat_users` table, but keeping the information separate allows for greater flexibility for the next version of FREiMOAT.

### 4.3 Query Tool Design

The query tool is responsible for taking the input from the Query by School Page and the General Query Page and outputting a text file. This section of FREiMOAT has the potential to disrupt service on iMOAT.net due to high resource demands. For this reason it is necessary for the query tool to attempt to use the minimum amount of system resources.

When a query is handed down to the query tool from the presentation layer, a new thread should be spawned to handle the request. This thread should set itself to run at a low or lowest priority level so that the threads running iMOAT will take precedence. Lowering the thread priority will help to lower FREiMOAT's stress on the CPU, but processing large queries will also tax the systems memory space.

Queries from FREiMOAT will have the following form.

For queries that collect student data and essays.

```
SELECT zipcode, state, country, esl, custom1, custom2, custom3, cus-
tom4, custom5, essay_title, essay_text FROM students INNER JOIN exam_student
ON exam_student.sstudent_id = students.sstudent_id INNER JOIN stu-
dent_essays ON exam_studnet.exam_student_id = student_essays.exam_student_id
WHERE students.site = ? AND student_essays.essay_test IS LIKE '%?%'
```

For queries that collect school data

```
SELECT read_instr, quest_instr, test_name, reading_link, reading_text,
question, min_words, max_words, componant_name, group_name, exam_name,
custom1, custom2, custom3, custom4, custom5 FROM test_components
INNER JOIN tests ON test_components.test_id = tests.test_id INNER
JOIN test_group ON test_group.test_group_id = tests.test_group_id IN-
NER JOIN exams ON exams.exam_id = test_group.exam_id INNER JOIN
sites ON sites.site_id = exams.site_id WHERE site_id = ?
```

Depending on how many schools or essays are involved, these queries could return very large results (on the order of hundreds of megabytes). For this reason, the query tool should have a limit to the amount of information that it can return in a single file. The first query that the tool should execute is one to estimate the size of the real query. This can be achieved by using a "SELECT COUNT(\*)..." query and then multiplying the count result by the average size of a student essay. If this estimated result size is greater than ~100mb, then the query should be broken into sections using a "SELECT TOP ? PERCENT" query. This is the procedure to collect data from the database.

The next step in processing a query is to anonymize data by replacing specific data with general data. The last two digits of every zip code returned should be replaced with zeros to limit the granularity in this data field. Next, each record returned needs to be checked so that it fits into a specific bin size. This means that there must be at least 3-5 other records in the same query that have the same location information - that is to say, having the same de-granulated zip code, state, and country information. All essays that do not fit into a large enough bin size should be removed from the result set. Once these steps are complete, then the query thread can output a text file, and return a link to the file to the presentation layer. While the query tool is processing, the presentation layer should use Javascript to periodically check the status of the query and report the status to the user.



# Chapter 5

## Conclusion

In this chapter, I discuss the potential issues with FREiMOATs design. Then layout the future work to be done for FREiMOATs development. Finally, I leave with a few final thoughts.

### 5.1 Potential Issues

During any design project, it is important to keep in mind as many potential problems as possible and to attempt to design them out. However, it is not always possible to out-design an issue. In this section, I look forward to issues that may arise in the development and testing of FREiMOAT.

#### 5.1.1 Scaling

There are a number of issues to look for as the use of FREiMOAT grows. Initially, FREiMOAT will be mainly used by members of the iMOAT consortium. However, hopefully the number of independent researchers will rise steadily as the service gains more popularity. Eventually the number of requests, by independent researchers, for access to FREiMOAT or access to a given school's materials, might become overwhelming. That is to say, SiteAdministrators may find all their time consumed by granting access for FREiMOAT. At this stage in FREiMOATs development it may

become necessary to eliminate the human gatekeepers altogether.

Creating automated FREiMOAT gatekeepers would be a difficult task indeed. First, an automated gatekeeper would need to scour the internet to verify the identity of a requesting researcher. This task could be simplified by using a service such as the google.com API. Second, these gatekeepers would have to be able to learn what kinds of requests for school materials are valid requests. This second task is quite a challenge for current artificial intelligence systems. Automated gatekeepers, though convenient, would be very difficult to implement. However, they may become necessary in the future.

More direct scaling issues could arise with very large queries. The SQL statements that are used by FREiMOAT are very complex and involve multiple tables. Once the iMOAT database reaches a large enough size, running a query could take hours, and use all of the computer's resources (e.g. CPU, hard drive space, memory) in the process. In order to prevent this from occurring, it will be very important to monitor large queries.

Monitoring can be done by hand or a monitoring tool can be created to keep statistics on all FREiMOAT queries. This tool would be a wonderful addition to FREiMOAT since it might, eventually, dynamically stall (or, if necessary, halt) a query that is interfering with the use of iMOAT.net.

### **5.1.2 Interface Design**

The success of FREiMOAT is largely tied to its interface. If users find that the Query by School page and the General Query page are too hard to use or not suited for their purposes, then FREiMOAT might prove useless. Multiple interfaces to the system should be developed to suit the needs of all users. Perhaps the correct decision for the first interface was a web service after all. Maybe some of the pages need to be broken down into smaller tasks. These questions will have to be answered by FREiMOAT's feedback.

### 5.1.3 Evolution of Use

Over time, the use of both FREiMOAT and iMOAT will change. However, it is difficult to predict how these changes will manifest themselves. It is possible that the multiple-choice components of iMOAT.net will be used more and that FREiMOAT will have to be modified to reflect that. Schools that use iMOAT might change the types of questions/prompts that they use in their assessments. If these changes lead students to compose more and more personal essays (i.e. containing more identifying material), then all of the anonymizing software in FREiMOAT could be totally circumvented. For this reason it will be important to constantly re-evaluate FREiMOAT's use and purpose.

## 5.2 Future Work

FREiMOAT has only been given a high level design, and now it must be implemented. The first step in this implementation should be to create screen mock-ups. Certain pages for the site have rather static content and will only require a single mock-up. Other pages (e.g. Query by School, General Query, Material Access, and Grant User Access) will need multiple mock-ups each to capture all of the possible states that can be achieved. Once these mockups have been checked for simple usability, development of FREiMOAT can begin.

During the development phase, FREiMOAT can be tested easily to create an interactive development process. Testing for usability and scale should be done during development to find any issues early on. Human testing can be done with the current consortium and their feedback will be the best resource for FREiMOAT's evolution.

Once FREiMOAT has been developed, tested, and patched, it will be ready for use by those outside the consortium.

## 5.3 Final Thoughts

The addition of FREiMOAT will place iMOAT.net at the forefront of online collaboration as well as aptitude testing services. The potential for this web service to become the de facto solution for educational research is quite high. The proposed design for FREiMOAT takes into account the operating environment under which iMOAT was developed, and has been designed to limit system impact. I look forward to FREiMOATs implementation and testing in the future.



# Bibliography

- [1] Iyengar, Vijay S. "Transforming Data to Satisfy Privacy Constraints" Copyright 2002
- [2] Sweeney, Latanya "Guarenteeing Anonymity when Sharing Medical Data, the Datafly System"
- [3] Seys, Stefaan, Claudia Díaz, et. all. "APES: Anonymity and Privacy in Electronic Services" 11/05/2001 <https://www.cosic.esat.kuleuven.ac.be/apes/>
- [4] Claessens, Joris, Claudia Díaz, et. all. "APES: Applications requirements for controlled anonymity" 05/12/2003 <https://www.cosic.esat.kuleuven.ac.be/apes/>
- [5] Kohl, John T., B. Clifford Neuman, Theodore Y. Ts'o "The Evolution of the Kerberos Authentication Service"