## Revisions

<table>
<thead>
<tr>
<th>Version</th>
<th>Primary Author(s)</th>
<th>Description of Version</th>
<th>Date Completed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Draft 01</td>
<td>Chad Hansen, Lisa Jordan, Daniel Osiecki, Chase Peeler</td>
<td>Preliminary version, all sections complete.</td>
<td>7 June 2005</td>
</tr>
<tr>
<td>Draft 02</td>
<td>Daniel Osiecki</td>
<td>Update after environment choice.</td>
<td>20 June 2005</td>
</tr>
<tr>
<td>Final 03</td>
<td>Lisa Jordan, Daniel Osiecki</td>
<td>Final updates and revision after draft grading.</td>
<td>26 July 2005</td>
</tr>
</tbody>
</table>
# Table of Contents

1. Introduction .......................................................................................................................... 5  
   1.1 Purpose .............................................................................................................................. 5  
   1.2 Scope ............................................................................................................................... 5  
   1.3 Definitions, Acronyms, and Abbreviations ...................................................................... 5  
   1.4 References ....................................................................................................................... 6  
   1.5 Document Organization ................................................................................................. 6  

2. Overview Requirements Description ....................................................................................... 6  
   2.1 Product Perspective ........................................................................................................ 7  
   2.1.1 System Interfaces ......................................................................................................... 7  
   2.1.2 User Interfaces ........................................................................................................... 7  
   2.1.3 Hardware Interfaces ..................................................................................................... 8  
   2.1.4 Internal Software Interfaces ....................................................................................... 8  
   2.1.5 Communications Interfaces ....................................................................................... 8  
   2.1.6 Memory Constraints ................................................................................................... 8  
   2.1.7 Operations .................................................................................................................. 8  
   2.1.8 Site Adaptation Requirements ................................................................................... 8  
   2.2 User Characteristics ......................................................................................................... 8  
   2.3 Apportioning of Requirements ........................................................................................ 9  
   2.4 Product Functions .......................................................................................................... 9  
   2.4.1 Student Functionality .................................................................................................. 9  
   2.4.2 Teacher Functionality ............................................................................................... 10  
   2.4.3 Administrator Functionality ....................................................................................... 10  
   2.5 Constraints .................................................................................................................... 10  
   2.6 Assumptions and Dependencies ...................................................................................... 11  

3. Specific Requirements Specification ....................................................................................... 11  
   3.1 External Interface Requirements ................................................................................... 11  
   3.1.1 User Interface ............................................................................................................. 11  
   3.1.2 Communications Interface ......................................................................................... 11  
   3.2 Functional Requirements ............................................................................................... 12  
   3.2.1 Exam Generation ....................................................................................................... 12  
   3.2.2 Data Filling ................................................................................................................ 12  
   3.2.3 User Accounts .......................................................................................................... 12  
   3.2.4 Exam Result Tracking ............................................................................................... 13  
   3.2.5 Question Categorization ........................................................................................... 13  
   3.2.6 Extended Functionality .............................................................................................. 13  
   3.3 Performance Requirements ............................................................................................ 14  
   3.4 Data Requirements ........................................................................................................ 14  
   3.4.1 Test Data .................................................................................................................. 14  
   3.4.2 User Data .................................................................................................................. 15  
   3.4.3 Exam Result Data ...................................................................................................... 15  
   3.4.4 Exam Statistics ......................................................................................................... 15  
   3.5 Software System Attributes ........................................................................................... 16  
   3.5.1 Reliability/Availability ............................................................................................... 16  
   3.5.2 Security ..................................................................................................................... 16  
   3.5.3 Extensibility/Maintainability ..................................................................................... 16
1. Introduction

1.1 Purpose

This Software Requirements Specification is intended to establish the functionality of the ICE @ Georgia Tech AP Practice Exam. It is intended for reference by the sponsor and the team of developers who will be extending the existing system and developing additional features. The purpose of this document is to precisely communicate the requirements for the new features.

1.2 Scope

The purpose of this project is to extend upon a currently existing system - the AP practice Exam developed by the ICE @ Georgia Tech. The existing software allows for users to take practice AP exams of a certain number of questions that have previously been entered into the system by teachers or developers. The additions to the existing system will facilitate greater functionality, including the ability to save sessions, track scores and display score statistics.

The existing software and its decedents are intended to help high school teachers and students prepare for the AP Computer Science Exam. The goal of the software is to provide an informational resource to teachers and give students a location to practice and apply their knowledge and receive detailed feedback. Students and teachers benefit from the system through raised awareness and higher exam scores.

The development of the AP Practice Exam project is expected to be continued by other parties after the end of this semester, and as such the list of requirements will be prioritized and divided into groups, some of which will be developed others that will only be specified and/or designed. All work done on the system will be well documented and extensible, as development may continue for the foreseeable future.

1.3 Definitions, Acronyms, and Abbreviations

AP – Advanced Placement
DBA – Database Administrator
Free Response Question - A type of question on the AP Exam that involves writing a complete answer, rather than choosing from a multiple choice set.
HTML – Hypertext Markup Language, a language used for writing webpages
HTTP – Hypertext Transfer Protocol
MD5 – Message Digest v 5, a hashing algorithm
1.4 References

All reference material provided by the sponsor are located at the sponsor's web site at the time of project commencement:

http://coweb.cc.gatech.edu/ice-dev/47

1.5 Document Organization

All stakeholders for this project should review all requirements contained within this document. However, sections one and two will be of primary interest to customers, while sections three and four will be of primary interest to developers.

Section one of this document is a general overview of the document. Section two contains a general description of the project, problem summary, and high-level customer requirements. Section three contains the specific requirements for the extension of the AP Practice Exam. These requirements will create a foundation to ensure that the design and implementation of the project will satisfy the problem requirements and customer needs. Developers will use these requirements throughout design, implementation, and testing. Section four is a list of supplemental material that may help in understanding how these requirements were formed.

2. Overview Requirements Description

The Ice @ Georgia Tech AP Practice Exam Database is a web based system created for three types of users. The primary user is students studying AP computer science. The system will store a set of questions and answers, present questions to students in an exam format, grade the student's responses, provide feedback and save statistics. The second type of user is teachers, who are able to manage classes, add questions and view their student's progress. The final user type is administrators, who can add schools, counties and questions, as well as edit questions and approve questions.
In addition to basic functionality, the system will keep track of various sets of statistics about exams taken and the grades received with respect to individual students, schools, counties, stats, and/or teachers. Access to AP exam questions will thus be tracked, with the data stored for later analysis.

Advanced features such as timed exams, the ability to pause and resume exams, and peer-graded free response questions may be made available.

When a teacher or administrator adds a question, they may specify multiple categories. This categorization is used when exams are taken to produce a balanced questions set that covers all categories - or the student may choose to create a custom question set and specify questions only from certain categories.

An advanced format for organizing categories in a hierarchy may be made available, with generalized parent categories and specific child sub-categories.

All data will be stored in a database, which will interface with a webserver on which the system is hosted. The webserver will facilitate a standard web-page based interface allowing users to access all system features via their web browsers.

2.1 Product Perspective

The AP Practice Test Database is a fully self-contained system. Though similar to other products on the market, the AP Practice Exam Database will be web based rather than a standalone program installed on one's computer. The system can be accessed on any system that has standard web browsing capabilities by students, teachers and administrators.

The AP Practice Exam Database is a unique product unlike others on the market because of the fact that it made for both the students and the teachers. While most products just focus on preparing the student for the AP exam, the AP Practice Exam Database also allows teachers to track the progress of their students, as well as manage their classes and add exam questions.

2.1.1 System Interfaces

The system will not interface with any other software systems, excluding user client access.

2.1.2 User Interfaces

The AP Practice Exam Database will feature a web based graphical user interface. As with all web systems, the interface is expected to be intuitive
and easy to learn; however, the students and teachers will have some
domain knowledge, so this will be taken into account in the design.

2.1.3 Hardware Interfaces

The system has no hardware interface requirements

2.1.4 Internal Software Interfaces

The system will depend on the associated functionality of a MySQL
database. The designers and developers must provide SQL data table
definitions and data to be provided to the system DBA for setup, as the
actual binary database application may not be distributed as a part of
secondary software.

Users of the system will not need to install any special browser software
(Java/javascript/flash etc) as the system will be designed to function with
only the basic browsing abilities provided by HTML and HTTP.

2.1.5 Communications Interfaces

The system will only interact using currently documented RFC protocols
transparently through HTTP.

2.1.6 Memory Constraints

The system itself will have minimal memory constraints. Satisfaction of
the minimum hardware requirements of the server environment will allow
full functionality of the system. Like all web applications, more hardware
resources may be required to handle increasing load on the system, but it is
expected that the minimal requirements will be sufficient for use in the
near future.

2.1.7 Operations

There are no processes of significant complexity that must be undertaken
by the customer to operate and maintain the system outside of maintenance
of the web services platform.

2.1.8 Site Adaptation Requirements

It is not expected that sponsor hardware or installation site must be
adapted in any way to support the system.

2.2 User Characteristics
There are three basic types of users of this system. These are students, teachers, and administrators. All user classes are assumed to have computer proficiency.

2.2.1 The students are high school students preparing to take the AP Computer Science exam.

2.2.2 The teachers are high school AP Computer Science teachers, using the system to help their students prepare for the exam.

2.2.3 Administrators are the users with high expertise of the system. They are in charge of system maintenance.

2.3 Apportioning of Requirements

In the following section, functions that are expected to appear in the final system are described with the word "will", while functions that appear only for the sake of documentation and future design are described with the word "may".

2.4 Product Functions

This section provides a simple overview of the specific functions the system will perform. Functions are organized by the user type that will generally perform them. For specific technical and programmatic requirements, see section 3.

2.4.1 Student Functionality

- Students will be able to login to the system, where they are presented with a menu which will allow them navigate to all features they have access to.
- Students will be able to request a set number of questions be presented to them in the form of an un-timed practice exam. Students can specify if they want a balanced set of questions from numerous categories, or all questions from specific categories. Students can choose whether to take an A or AB level exam.
- Students will be able to answer each question with the multiple-choice answer of their choice, or abstain from answering. Students will be able to view the answer to each question.
- Students will be able to view the results of any exam they have completed, including reviewing each question, their answers and percent of questions complete/incomplete.
- Students will be able to view a graph representing their scores on exams over time.
- Students may be able to adjust the time scale over which they view score statistics.
- Students will be able to change their information, including their password and email address.
• Students will manage this information for a single account.
• Students may be able to answer free response questions, as well as grade other students free response questions.
• Students may be able to take a timed test requires that all questions be answered in a given period of time.
• Students may be able to save their progress for a non-timed test and return later to complete it.
• Students will be able to search for a specific question, or browse through all questions, outside of a testing interface. Searching will be done based on author, keyword, and/or category.

2.4.2 Teacher Functionality

• Teachers will be able to login to the system, where they are presented with a menu which will allow them to navigate to all features they have access to.
• Teachers will be able to update their account information such as password and email address.
• Teachers will be able to setup accounts for their students.
• Teachers will be able to add new questions to the Exam Database, and modify or delete questions they originally authored.
• Teachers will be able to format their questions and answers using basic HTML tags. Teachers may be able to include images in their questions.
• Teachers will be able to view which of their students have taken which tests, as well as the results for the tests. They will also be able to view statistical, score-based data for different questions, over time.
• Teachers may be able to adjust the time scale over which they view statistical score data.
• Teachers will be classified based on their location, such as county, city, and state for the development of higher-level statistical data.
• Teachers may be able to grade free response questions.
• Teachers may be able to access printable versions of exams with an answer key.

2.4.3 Administrator Functionality

• Administrators will have access to all features that teachers have access to.
• Administrators will be able to approve modifications to existing questions, approve new questions, edit questions, and delete questions - regardless of author.
• Administrators will be able to create teacher accounts and new administrator accounts, and will be able to elevate teacher accounts to administrator status.

2.5 Constraints
2.5.1 MySQL must be used as the database system.

2.5.2 System must maintain a level of security that protects individual accounts from being used by non-authorized people. Users must also have their access restricted based on their security level.

2.5.3 Information Sharing: Teachers may view statistics for each of their students. Teachers may also view overall class/county/state statistics for students not in their classes - however, the teacher may not view statistics for single students who are not in their classes.

2.6 Assumptions and Dependencies

2.6.1 All initial data provided for the system is assumed to be correct and free of errors.

3. Specific Requirements Specification

The following language conventions are used in this section:
(1) "shall" indicates a required feature
(2) "should" indicates a desired feature
(3) "may" indicates a potential feature

3.1 External Interface Requirements

3.1.1 User Interface

3.1.1.1 The system shall present a user interface that takes the form of a W3C HTML 4.01 and W3C CSS 2.1 compliant web application.

3.1.1.2 The system shall provide a page or set of pages for each of the functions listed in section 3.2, organized in the manner described

3.1.1.3 The system shall support all inputs and outputs as listed in section 3.4.

3.1.1.4 The system shall preserve the graphical format established by the existing version of the software.

3.1.1.5 The system should provide consistently available navigation to all main functions from any location in the interface.

3.1.2 Communications Interface

3.1.2.1 The system shall take the form of an RFC 2616 compliant webservice providing the user interface pages described in section 3.1.1.
3.2 Functional Requirements

3.2.1 Exam Generation

3.2.1.1 The system shall store a set of questions and answers as described in 3.4.1.

3.2.1.2 The system shall generate practice exams of a user-specified (student or teacher) size and difficulty level. These exams shall be a random subset of the data stored by the system covering all categories, except in the case of customized question sets, in which case the categories will be user-specified.

3.2.1.3 The system shall present questions one at a time to be answered or skipped, and shall subsequently grade each question as it is answered, providing the correct answer and explanation material as described in 3.4.1.

3.2.2 Data Filling

3.2.2.1 The system shall accept new question data from sufficiently privileged users (see 3.2.3).

3.2.2.2 The system shall allow sufficiently privileged users to modify or remove existing questions.

3.2.2.3 The system shall require the review and approval of a sufficiently qualified user on all newly added questions before making use of questions in test generation. The system shall also require approval of modification or removal of questions that were approved in the past.

3.2.2.4 The system shall allow for the formatting of question data using basic HTML tags, limited to <ul>, <ol>, <p>, <b>, <i>, <br> and <table> (and table subtags).

3.2.3 User Accounts

3.2.3.1 The system shall support specialized access by known individuals with an associated user account (herein users).

3.2.3.2 The system shall, for each user, store profile information as described in 3.4.2.

3.2.3.3 The system shall allow and restrict access to all functions as dictated by the privilege level of the accessing user.

3.2.3.4 The system shall allow all users to update the information stored about themselves (see 3.2.3.2) and shall allow sufficiently privileged users to update information about all users. This shall include privilege level information.

3.2.3.4 The system shall allow access to all features regardless of login status, unless otherwise noted.
3.2.3.5 The system shall allow the creation of new user accounts by sufficiently privileged users. The system shall not, at this juncture, allow for account deletion.

3.2.4 Exam Result Tracking

3.2.4.1 The system shall store a log of access to generated exams as described in 3.4.3 when generated exams are accessed by a known user (see 3.2.3).

3.2.4.2 The system shall allow the user that took a generated exam to review view that exam and their results at any later time.

3.2.4.3 The system shall generate summary exam statistics as described in 3.4.4 for any subset of the results stored (see 3.2.4.1). These statistics may be generated over any categorization of users (see 3.2.4.2) and any time frame.

3.2.4.4 The system should allow the viewing of historical exam statistics (see 3.2.4.3) with respect to time.

3.2.5 Question Categorization

3.2.5.1 The system should support the creation of a list of question categories that may be applied to any question in the database.

3.2.5.2 The system should use categorization data when generating exams (see 3.2.1) to balance the number of questions used from each category.

3.2.5.3 The system should support user specification of the number of questions from each category to appear on a generated exam.

3.2.5.3 The system may support hierarchical organization of categories, involving general parent categories with many specific children categories.

3.2.6 Extended Functionality

3.2.6.1 The system may support questions without any supporting answers. The system should allow for a simple textual response to such questions and should store this response as in 3.2.4.

3.2.6.2 The system may allow students to view one another's answers to questions of a free-response nature (see 3.2.6.1) and allow a single other student to assign a grade to that answer. This grade should be used for such questions when generating exam statistics (see 3.2.4).

3.2.6.2 The system may support an alternate format for presenting generated exams in which all questions are asked sequentially without answer grading and the system should limit the time available to answer questions to a predetermined duration.
The system should grade such exams after this time limit expires, according to normal rules.

3.2.6.3 The system may allow identified users (see 3.2.3) to halt a generated exam that is in progress, saving their current state to the system. The system should then allow the same user to restart this exam session at a later time.

3.2.6.4 The system shall allow questions and associated answer information to be viewed outside of a generated exam. The system shall support the generation of a full list of questions, organized by category (see 3.2.5) or by author (see 3.2.2). The system may also support keyword searches of questions.

3.3 Performance Requirements

3.3.1 The systems shall support at least 100 concurrent users

3.3.2 The system may accept up to 1000 concurrent users

3.3.2 The system shall present a question load time of no more than 30 seconds in 99% of cases

3.4 Data Requirements

3.4.1 Test Data

The following data shall be stored for a set of questions that may appear on generated tests:

<table>
<thead>
<tr>
<th>Datum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question Text</td>
<td>HTML formatted text representing an exam question.</td>
</tr>
<tr>
<td>Answer Text</td>
<td>HTML formatted text representing an answer to a specific question.</td>
</tr>
<tr>
<td>Answer Explanations</td>
<td>HTML formatted text corresponding to a specific answer describing the (in)correctness of that answer.</td>
</tr>
<tr>
<td>References</td>
<td>Zero or more URLs that link to information associated with a specific question.</td>
</tr>
<tr>
<td>Categorization</td>
<td>One or more of an n enumerated list of taxons or keywords (TBD) that categorize a specific question.</td>
</tr>
<tr>
<td>Exam Form</td>
<td>One of an enumerated list of exam forms, either “A” or “AB” representing which exam form a question appears on.</td>
</tr>
<tr>
<td>Author</td>
<td>The name of the user that created this question (see section 3.4.2)</td>
</tr>
<tr>
<td>Approval Status</td>
<td>A boolean value indicating whether this question is approved for use in generated exams.</td>
</tr>
</tbody>
</table>
### 3.4.2 User Data

The following data shall be stored for each user of the system:

<table>
<thead>
<tr>
<th>Datum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>A unique, non-empty string of limited size that uniquely identifies this user.</td>
</tr>
<tr>
<td>Name</td>
<td>The user's first name.</td>
</tr>
<tr>
<td>Password Hash</td>
<td>An MD5 hash of this user's password.</td>
</tr>
<tr>
<td>Authority Level</td>
<td>One of an enumerated list of authority levels, currently “Student”, “Teacher”, or “Administrator”.</td>
</tr>
<tr>
<td>E-Mail</td>
<td>A current e-mail address for this user.</td>
</tr>
<tr>
<td>Teacher</td>
<td>A reference to a valid user account of a teacher, if this user is a student.</td>
</tr>
<tr>
<td>State</td>
<td>One of the 50 enumerated US states, in which this user teaches, if this user is a teacher. (students are assumed to be in the same school as their teacher)</td>
</tr>
<tr>
<td>County</td>
<td>The county in which this user teaches, if this user is a teacher.</td>
</tr>
<tr>
<td>School</td>
<td>The name of the school at which this user teaches, if this user is a teacher.</td>
</tr>
</tbody>
</table>

### 3.4.3 Exam Result Data

The following data shall be stored each time a generated exam is taken:

<table>
<thead>
<tr>
<th>Datum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>A reference to the user (see 3.4.2) that took this exam.</td>
</tr>
<tr>
<td>Date/Time</td>
<td>The date and time this exam was generated.</td>
</tr>
<tr>
<td>Answer Set</td>
<td>The set of answers given for each question on this generated exam.</td>
</tr>
</tbody>
</table>

### 3.4.4 Exam Statistics

The following data shall be generated upon request for a given scope of test results, with respect to time where appropriate:

<table>
<thead>
<tr>
<th>Datum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question Difficulty</td>
<td>For each question, the percentage of time that the question was answered incorrectly.</td>
</tr>
<tr>
<td>Question Popularity</td>
<td>The number of times a given question has appeared on a generated exam.</td>
</tr>
<tr>
<td>Exams Taken</td>
<td>The number of exams generated and completed, total.</td>
</tr>
<tr>
<td>Exam Size</td>
<td>The average number of questions requested to be on a generated exam.</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td>Average Exam Score</td>
<td>The average of all exam results.</td>
</tr>
</tbody>
</table>

### 3.5 Software System Attributes

#### 3.5.1 Reliability/Availability

As a web service, the system is expected to be available at all times. The system will be considered reliable if there are no failures due to programmatic error in 10 interleaved test runs of each use case specified in the eventual design. In the case of a failure, no sensitive information shall ever be lost, though individual student exam answers for incomplete exams may be unrecoverable in cases where results have not been expressly saved.

#### 3.5.2 Security

Because of the sensitive nature of some information included in the system, it is required that all sensitive data be transferred in a secure fashion. Sensitive information will not be made publicly available in any fashion outside the consent of the users.

#### 3.5.3 Extensibility/Maintainability

As this system will be used for the foreseeable future, it is required that all changes made be well documented and complete, while allowing for additions to be made in the future. All components used will be freely available. The backend for the system will be written in a format understandable by the customer, though the system should not need any internal maintenance.

#### 3.5.4 Usability

As a web based application, the system should be immediately intuitive and understandable. However, because the system is aimed at Computer Science students and teachers, the users will have some domain knowledge, allowing for a design that caters to an educated audience, rather than a novice user group.

### 4. Change Management Process

Changes to the requirements may be broached through email, written or verbal communications. The changes will then be discussed within the group and a decision will be made as to the feasibility of including the new requirements. The group leader has the final say in whether the change will be included. No new features will be added to the requirements after July 8th, 2005 – three
weeks from the final project deadline—in order to ensure sufficient time for completion. The SRS will be updated as changes are approved and the updates will be logged by date in Appendix A.

5. Document Approvals

All stakeholders in the project should sign off on this document to show that they have read and approve all of the requirements contained within.

Barb Ericson Date

Chad Hansen Date

Lisa Jordan Date

Daniel Osiecki Date

Chase Peeler Date

6. Supporting Information

Appendix A: Log of Requirements Changes

07/05 -
The system may provide printable versions of exams with answer key.
The system may allow images to be included in a question's text.
The system shall allow basic HTML tags to be included in a question's text.