UNT Web-Based Grade Book

Software Requirements Document
October 3, 2006

Justin Blaydes
Cameron Palmer
Robert Thompson
Davud Turkdill

Table of Contents

1. Introduction

The web-based grade book will allow instructors to manage their students' grades through a web browser. The system will allow instructors to create a profile for each of the classes they are teaching, including a customizable category creation. The grade book will allow the instructors to import or export students' profiles to or from a **CSV file**, as well as allow them to handle the entry of individual students manually.

Management of each class's grade book will occur through a grid-based layout where the instructor can see all the grades for all the students. The grid will calculate each student's current score in the class, as well as calculate the average scores in the class in each category (for example, the average score on a certain exam or assignment). The instructor can also hide categories of grades to better manage the grid.

This document describes the requirements and specification for the web-based grade book, categorized by system requirements and software requirements. The last page of the document contains a glossary that explains terms made bold-faced in the text.

2. System Requirements

2.1. *Server Requirements*

In order to run the web-based grade book program, the user needs access to a server that is Intel-compatible and has the following programs installed:

a. **Ubuntu Linux 6.06 LTS**
b. **Apache 2**
c. **PHP5**
d. **MySQL**

2.2. *Client Requirements*

Instructors that are using the web-based grade book needs to use a computer with the following requirements in order to run it:

a. **Standards-compliant web browser**
b. Network access to server

Standards-compliant web browsers include all versions of Mozilla Firefox, Opera, and Safari, as well as Internet Explorer 7. (Earlier versions of Internet Explorer are not standards-compliant.)

3. Software Requirements

3.1. *Design Considerations*

When it comes to design considerations, the grade book adheres to the following requirements:

a. The grade book should follow the University North Texas (UNT) color and web design conventions as much as possible.

b. Records should be preserved as much as possible. When an instructor first deletes a record, it is actually just hidden. The secondary deletion (similar to emptying the trash can in most operating systems) will be required to actually delete the record.

3.2. *System Startup*

The grade book has the following requirement for booting the server:

a. System startup, including all grade book-required components on the server side, should start automatically on boot.

3.3. *Course Management*

3.3.1. Course Creation and Editing

The following items describe the requirements regarding how the instructor should be able to add a new course to the grade book and how to edit the input for the course or delete it once it is created:

a. Adding a course (see Figure 1) requires the following information (includes examples):

1. Title – Software Development I
2. Department – CSCE
3. Course – 4410
4. Section – 002
5. Term – Fall
6. Year – 2006

b. The course add and edit pages will allow **letter grade cut-offs** to be applied to a class.

c. To select a course to edit, a list of the currently active courses should be displayed and selectable (see Figure 2).

d. The user should be able to delete a course within the edit screen.

e. All information that was assigned during course creation should be available to edit within the edit screen.

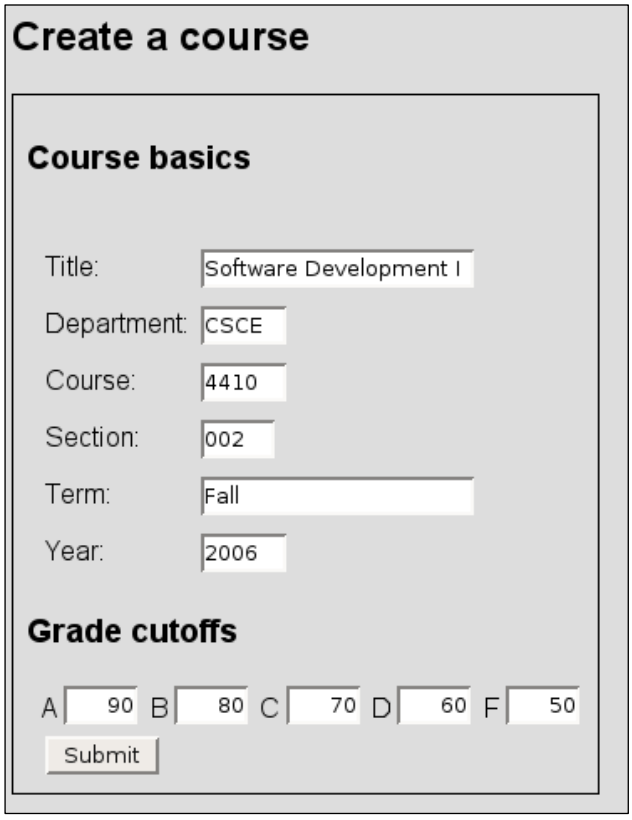


Fig. 1. How to create a new course for the grade book.

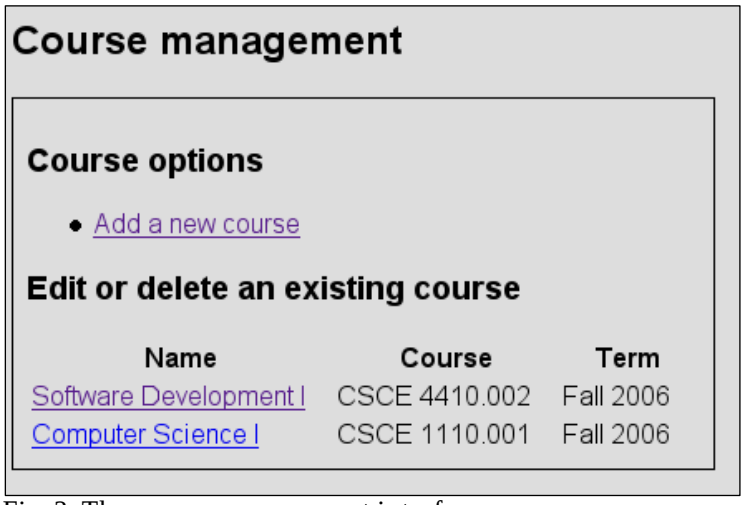


Fig. 2. The course management interface.

3.3.2. Course Category Management

A **course category** defines major headings under which assignments and exams will be created, for example “Homework” or “Exams.” The following requirements apply to the management of these categories:

a. The grade book categories will handle only whole percentages, 1-100.

b. Adding a **course category** will require the following information (includes examples):

   1. Name – Exams
   2. Percentage – 20
   3. Rank – 1

c. Rank will be used to determine the order in which to display the categories in the grade book (see Figure 3).

d. When creating a category, the rank should insert the new item into that position in the list.

e. The order of the categories should be editable.

f. **Course category** percentages should always add up to 100%.

g. Inserting a new category into a course that is at 100% should be allowed if percentage set to 0%.

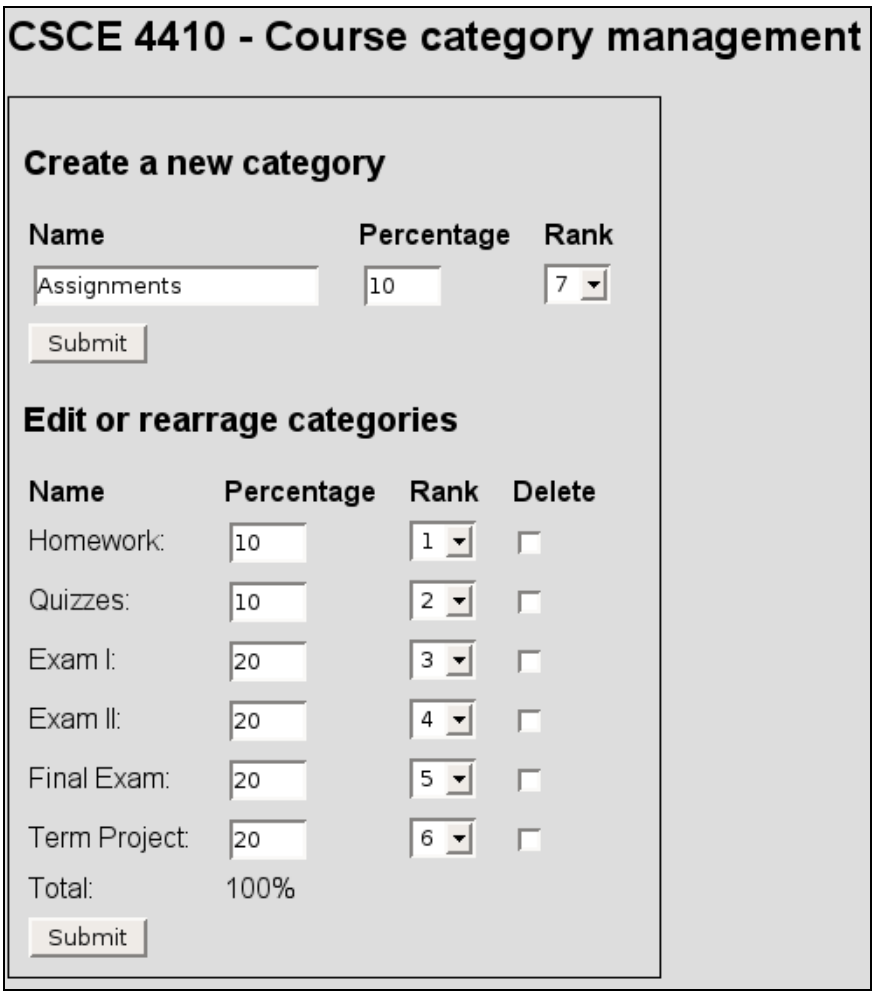


Fig. 3. The instructor can assign categories' weight.

3.4. *Student Management*

Student management will allow the user to edit all aspects of the user profile. This interface will also allow for import and export of the students to a file.

a. Adding a student requires (see Figure 4) the following information (includes examples):

    1. Name (Last, First Middle) – Palmer, Cameron L

2. EUID – clp0147
3. EmplID – 10436511
4. Email address – clp0147@unt.edu
5. Web address – http://cameronpalmer.com
6. Phone – 9723751441
7. Comments – nice guy.
8. Picture

b. Mass import of the student portion of the database should be allowed via **CSV** file. The **CSV** file format is defined at the end of this document.

c. All students should be exportable to a **CSV** file format that is defined at the end of this document.

d. All elements of the student should be editable after creation (see Figure 5).

e. The student image format will be a jpeg not to exceed 1MB in size.

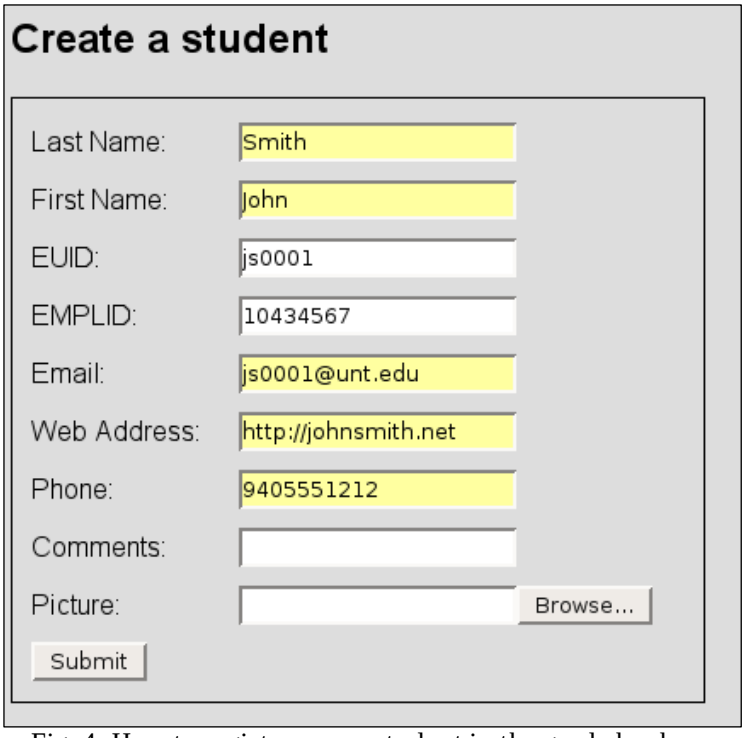


Fig. 4. How to register a new student in the grade book.

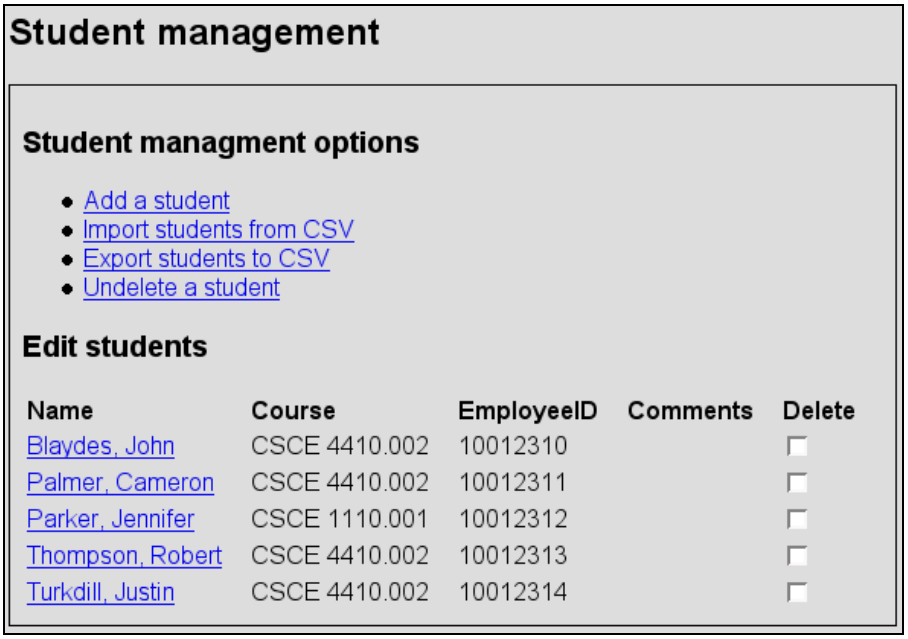


Fig. 5. Access student edits from the student management interface.

3.5. *Assignment Management*

The user will be able to create new assignments (see Figure 6) in the grade book based on the following requirements:

a. Assignments can be created from the grade book interface or the standalone interface.

b. Adding a new assignment requires the following information (includes examples):

1. Title
2. Maximum Points
3. Due Date
4. Comments
5. Course
6. Category
7. Rank

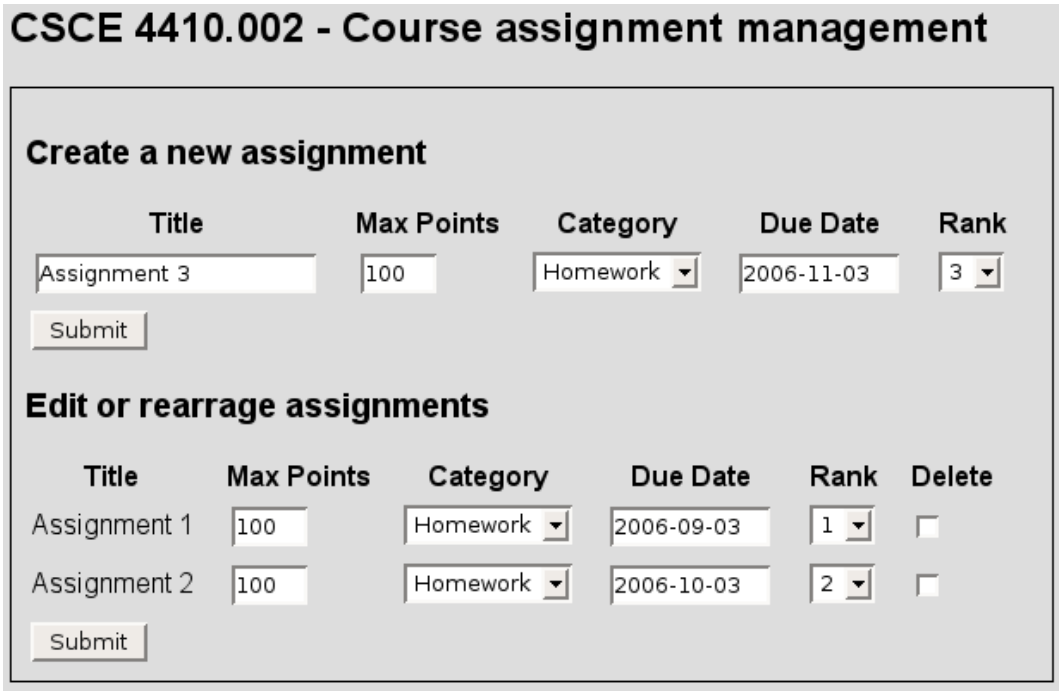


Fig. 6. Edit the assignments and their order from the assignment management page.

3.6. *The Grade Book Specifications*

The grade book interface is a multi-level grid (see Figure 7) organizing the students and categories for the class and averaging the various scores for each student and each category. The following requirements apply to the grade book:

a. Blank grades and comments on columns are not averaged.

b. Text should be allowed in the grade field.

c. Comments on rows are averaged and considered zero.

d. An arbitrary number of grades are allowed within each category.

e. Students with a **populated comments field** should provide a visual indicator.

f. Categories should be seen if set to viewable.

g. The instructor should be able to set any category as hidden.

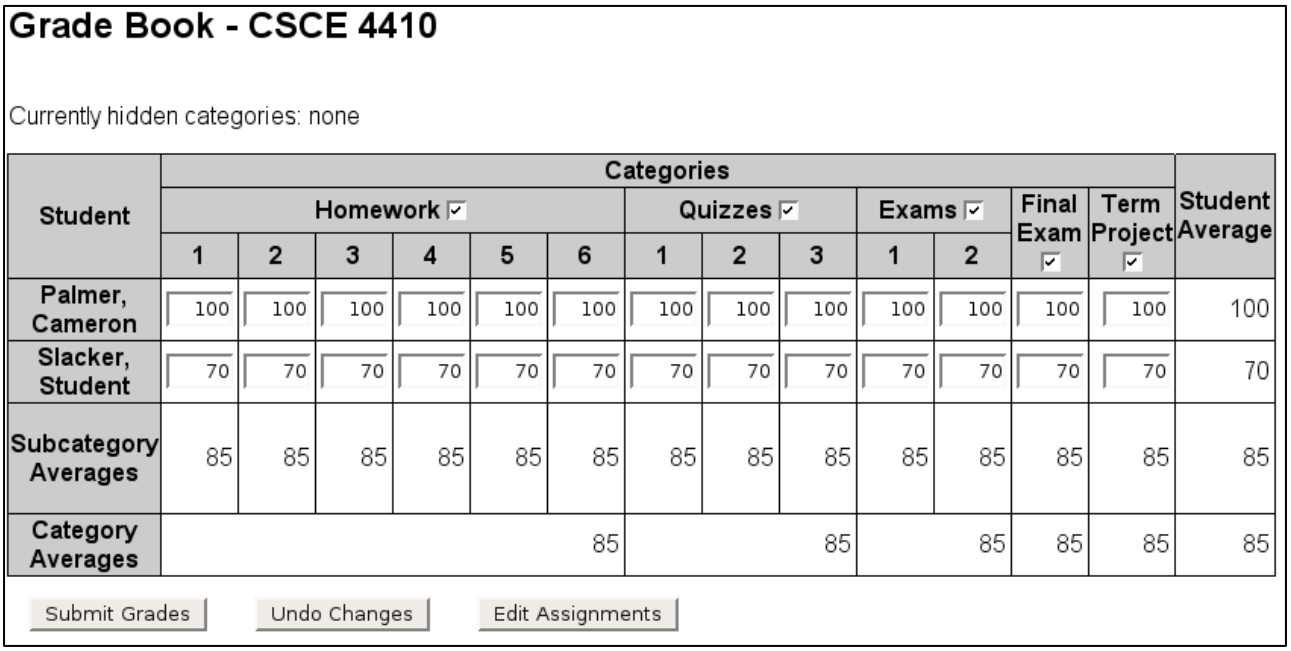
h. When a category is hidden it will appear as a link at the top of the grade book as a link that will re-enable it.

i. When the user hovers over the hide-a-category checkbox context sensitive help will appear reminding the user what the checkbox does.

j. Assignments should be able to be created and edited with grade book.

k. Access to the student's information should be available from grade book.

l. **Validation of input** should occur so that values entered in grade book outside the maximum points should indicate a warning.



Grade Book - CSCE 4410

Currently hidden categories: none

Student	Categories													Student Average
	Homework ☑						Quizzes ☑			Exams ☑		Final Exam ☑	Term Project ☑	
	1	2	3	4	5	6	1	2	3	1	2			
Palmer, Cameron	100	100	100	100	100	100	100	100	100	100	100	100	100	100
Slacker, Student	70	70	70	70	70	70	70	70	70	70	70	70	70	70
Subcategory Averages	85	85	85	85	85	85	85	85	85	85	85	85	85	85
Category Averages						85			85		85	85	85	85

Submit Grades Undo Changes Edit Assignments

Fig. 7. The grade book grid automatically averages each student's grades, overall and in different categories. By unchecking the checked boxes, instructors can hide the categories they want.

3.7. *Student Import/Export File Format*

In order to be able to import and export a student registry, the file must be in the following format:

#emplid,first_name,middle_name,last_name,euid,email,phone

An example of this format would look like this:

```
10235567,John,Randall,Smith,jrs0147,jrs@example.com,9405551
212
```

4. Glossary

CSV file – a common text file format that contains comma-delimited values (comma separated values).

X86-compatible system – refers to a computer system that is compatible with Intel's x86 CPU family.
http://en.wikipedia.org/wiki/X86_compatible

Standards-compliant web browser – Web standards are a set of rules and recommendations on how to use XHTML, CSS, and the document object model to make websites readable by a variety of browsers.
http://www.w3.org/

Letter grade cut-offs – the point at which a numerical grade will be rounded to the next highest letter grade. Ex. 89.5 = A, 89.2 = B.

Course category – Assessment types as defined by the instructor in the gradebook. Ex. Quiz, homework, tests, or other.

Populated comments field – the status of a field when it contains a value (i.e. not empty). In this case the comments field has a value.

Validation of input – a check to see that the value entered falls within a specified range.

Ubuntu Linux 6.06 LTS – a Linux operating system distribution targeting personal computers. It concentrates on usability, and ease of installation.
http://en.wikipedia.org/wiki/Ubuntu_(Linux_distribution)

Apache 2 – an open-source HTTP server. It is secure, efficient, and runs about 70% of web sites on the internet.
http://httpd.apache.org/

PHP5 – an open-source programming language used for developing server-side applications and dynamic web content.
http://www.php.net/

MySQL – an open source relational database management system that uses Structured Query Language (SQL), the most popular language for processing data in a database.
http://www.mysql.org/