April 2010

Automatic Reassessment and Relearning System

Samuel InHo Song
Worcester Polytechnic Institute

Automatic Reassessment and Relearning System

Submitted to

Worcester Polytechnic Institute

April 29, 2010

by Samuel Song

Advised by Neil Heffernan

This report explains the project of creating an Automatic Reassessment and Relearning System as a new feature to the ASSISTment tutoring system. The retesting system uses a schedule of increasing time intervals to help students retain knowledge. Research in cognitive science supports the hypothesis that student learning can be improved by practicing skills over a longer period of time instead of only requiring short term retention.

Table of Contents

List of Figures 2

Abstract 3

1. Introduction 3

2. The Problem 4

3. ASSISTment Tutor 5

4. Project Overview 7

5. Project Design 8

5.1 Overview........ 8

5.2 Settings 9

5.3 Reassessment Tests........ 11

5.4 Relearning Assignments 15

5.5 Reports 16

6. Technical Implementation 19

6.1 Overview........ 19

6.2 Existing Modifications 19

6.3 New Models........ 21

6.4 ARRS Tutor 22

6.5 ARRS Teacher 23

7. Future Work 25

8. Conclusions 26

9. References 27

List of Figures

Figure 1: ASSSTment Tutor 7

Figure 2: Class Settings Page 9

Figure 3: ARRS Settings Page 11

Figure 4: Show Assignments Page 12

Figure 5: Reassassment Test 13

Figure 6: Submitted Reassassment Test 14

Figure 7: Show Assignments Page after Retest 14

Figure 8: Relearning Assignment 15

Figure 9: Un-filtered Mastery Reportt 16

Figure 10: Reassessment and Relearning Report 17

Figure 11: Student ARRS Report 18

Figure 12: Problem Based Report 18

Figure 13: Cumulative Problem Based Report 19

Automatic Reassessment and Relearning System

By Samuel Song

Advised by Neil Heffernan

Abstract. Research in cognitive science supports the hypothesis that student learning can be improved by returning to past topics to test if students have retained knowledge. An Automatic Reassessment and Relearning System was implemented to an existing tutoring system to retest if students have retained knowledge using a schedule of increasing time intervals. The goal of the project is to improve student learning by helping students learn over an extended period of time.

1. Introduction

The Automatic Reassessment and Relearning System is a feature added to the ASSISTment Tutoring System to help improve student learning. Research in cognitive science supports the idea that student learning can be improved by returning to past topics periodically. Student that practice skills over a longer period of time can retain knowledge better and can possibly accelerate future learning. The Automatic Reassessment and Relearning System (ARRS) is a technological solution to help teachers implement this idea to improve learning.

The 2007 Practice Guide by the Institute for Education Sciences titled "Organizing Instruction and Study to Improve Student Learning" concludes "that cognitive science research supports the idea that teachers can improve student learning by making sure they come back to topics periodically to ensure students are retaining knowledge." The report also claims that the idea is supported by the well known *spacing effect* that shows students will learn more if practice is spaced out of time instead of in a single large practice session.

ARRS uses this idea of *spacing* in combination with the idea of *mastery learning*. In mastery learning, students practice skills until they achieve some threshold to gain mastery status instead of completing a preset number of problems. Research has shown mastery learning to be effective, however has not been widely used until computers were employed to manage the tedious bookkeeping. Current implementations of computer tutor systems that employ the idea of mastery learning use a criterion that allows students to gain mastery status without having to demonstrate retention over a period of time. The criterion for mastery status in ARRS will become more robust by requiring students to retain knowledge for a longer period of time. ARRS attempts to improve student learning by changing the mastery criterion to use this more robust measure of learning.

Practically, this means that ARRS will first require students to demonstrate the short term retention of the old mastery criterion. Students then have the knowledge reassessed based on a given schedule that progressively lengths the time between each practice. Students who fail the reassessment test are offered support in re-mastering the topics through the NSF funded intelligent tutoring system ASSISTment.

ARRS is built to extend the current *mastery learning* system in the ASSISTment tutor. Mastery learning in ASSISTments would previously require students to practice skills such as absolute value and equation solving for short periods of time before reaching a criteria threshold to achieve a mastery status with the associated skill. The ASSISTment mastery learning system only requires students to demonstrate short term retention of the knowledge. The Automatic Reassessment System is a new feature to ASSISTments that attempts to improve the learning of students using ASSISTments by altering the criteria for mastery to demonstrating understanding over time.

The Automatic Reassessment and Relearning System schedules mixed review tests for students with increasing intervals of time. The mixed review tests includes past skills in which students have already shown short term retention in the mastery learning system. Students are retested on these past skills to ensure that they have retained the knowledge. The reassessment tests are scheduled with increasing periods of time between them. For example, after a student completes the mastery learning for a skill, the student may be rested on the skill one day later, then one week later, etc. Students using ARRS will be practicing over an extended period of time instead of a shorter condensed learning time. Student who successfully demonstrate retention are advanced in the reassessment schedule to a longer spacing of time. Students who fail to show retention are helped to relearning the skill through the existing ASSISTment tutor.

2. The Problem

The idea of spaced learning from cognitive science literature is not being found in the classroom. The 2007 *IES Practice Guide* recommends the instructional practice of spacing learning over time and quizzing students to re-expose information already learned. Research in cognitive psychology has shown that learners will retain more when exposed to facts, concepts, or knowledge at multiple points in time (Cepeda, Pashler et al., 2006) which is known as the *spacing effect*. Rohrer's research presented identical question in two manners: all at one or spread over time. The subjects learned more when the exact same questions were given over a period of time (Rohrer & Pashler, 2007; Rohrer & Taylor, 2007). Roediger & Karpicke have found a "testing effect" that simply quizzing students causes more learning (2006).

Although the *spacing effect* is widely known to be effective in helping improve student learning, why is it not used much in the classroom? The benefits of spaced learning have been known for awhile in literature on human learning (Underwood, 1961). Even through laboratory results have been impressive, not much of these works have been put to use in the classroom. Rohrer (2009) writes "the spacing of practice is being grossly underused in mathematics instruction, as other authors have concluded" The results of research have been promising so why is there a lack of transfer to the classroom?

The reason may possibly be with the difficult bookkeeping required. The recommendation of re-quizzing students to ensure they retain knowledge is difficult to fully implement. It may not be practical to schedule reassessment tests and track each student's retention. For example if a teacher required students to answer problems that reflect on past lessons, the teacher would need to investigate the students who fail to show retention to determine if the students did not retain the knowledge or if it was only a careless error. The teacher would need to check how the student preformed on past problems with this skill to get a better picture. Students that do not retain understanding of a skill would then need additional practice for the individuals to relearn it which then would also require review from the teacher to know if the students attempt to relearn was successful. This creation of additional good problems for practice and grading would continue for each student's attempt until the skill is relearned. The teacher would need to do this for all of the students and for all of the skills that the students learn. It would require tedious bookkeeping for teachers to manually retest students for retention without computer aid.

Another reason spaced learning may not have been adopted is that the performance of students will lower in the short term. The situation is called "desirable difficulties" by Kornell and Bjork because student performance is decreased during the acquisition phase, but will cause benefits in the retention phase and will be "desirable" in the long term (Korneel & Bjork, 2008). Rohrer's "spacing" studies (2009) showed that students lowered performance during training than students who were given all training amassed together, but the spaced learners had a superior performance on the post-test. It may be hard to convinced student and even adults who do not value delayed rewards as much as immediate ones to trade worse results now for better outcomes later (Green, Fry et al., 1994).

The Automatic Reassessment and Relearning System was created to help teachers overcome the logistical difficulties of implementation with computer assistance. The ARRS does the scheduling and information bookkeeping for all of the students. ARRS tracks each student's progress through the scheduled reassessment and assigns relearning problems for students to practice skills again. The feature was added to the ASSISTment tutor assist teachers with an automated system to implement the idea from cognitive science of learning over longer periods of time to increase student retention of knowledge and to potentially accelerate future learning.

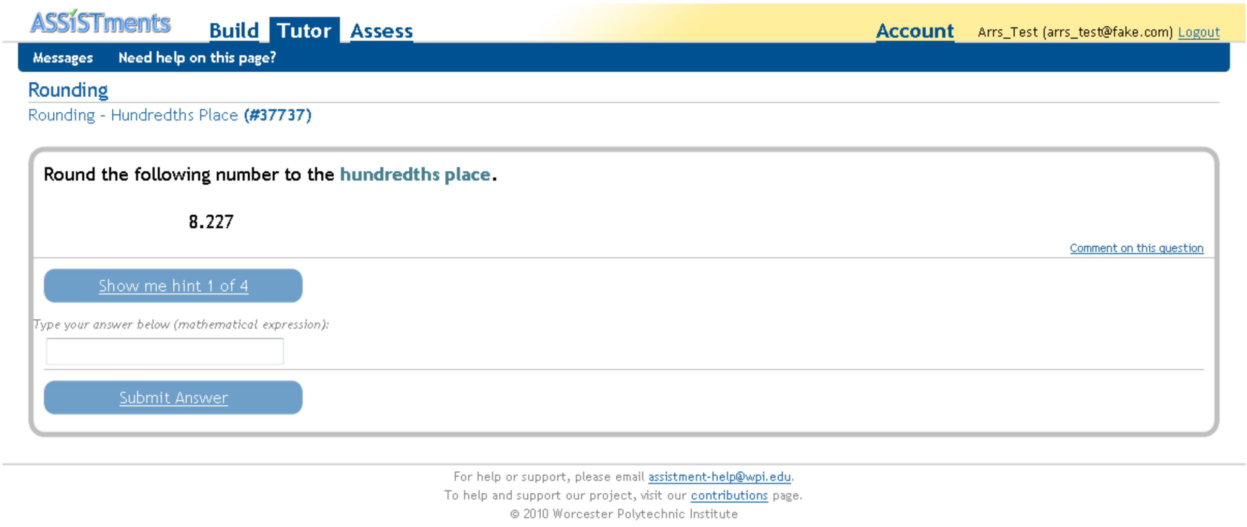
3. ASSISTment Tutor

The Automatic Reassessment and Relearning System feature is an extension to the mastery learning system in the ASSISTment tutor. There are other tutor systems that implement mastery learning, but none of them use criterion for mastery that requires students to demonstrate retention of the knowledge over an extended period of time. In mastery learning students practice until a mastery threshold is reached such as high performance on recent questions instead of completing a set number of problems to learn from. Other tutors are not taking advantage of the spacing effect for learning. The successful Cognitive Mathematic Tutor from Koedinger et al. (1997) allows students to reach mastery in a single day and then students are then never asked to return to check back on that mastery. Corbett and Bhatnagar found that the Cognitive Tutors would over predict paper and pencil test performance (1997). This may be due to the false assumption that students will never forget any topics that have mastered without testing for retention over time. The mastery criterion assumes that answering problems correct is independent of when the skills were practiced, but students will perform better on skills more recently practiced.

The ASSISTment tutor also uses a mastery learning system with only short term retention. The ARRS is an attempt to improve the mastery criterion by using a more robust measurement in which students will periodically be retested to maintain mastery. Researchers at the NSF funded Pittsburgh Science of Learning Center define "robust" learning to mean that learning is retained over a period of time, transfers to new areas, and aid in future learning (VanLehn & PSLC, 2006). Short term evaluations of learning are not robust measure of learning.

The mastery learning in the ASSISTment tutor uses special assignments called mastery learning problem sets. The mastery learning problems sets are associated with various skills such as Absolute Values or Equation Solving. There are over 130 mastery problem sets in the ASSISTment tutor that correspond to skills for middle school mathematics. Worcester public schools have been using ASSISTments for over six years and it now part of the school curriculum.

Students practice the skills until a criterion is fulfilled and the student achieves mastery status for the skill. Students in the ASSISTment tutor can be granted mastery status by either successfully testing out by answering correctly the first n problems, a variable setting chosen by teachers, or by demonstrating understanding by answering n problems consecutively correctly. The typical requirement in ASSISTment tutor is three correct answers. For answers to count towards the consecutive number correct the student may not take any tutor actions such as hints. The ASSISTment tutor helps students learn with features such as hint messages or breaking down problems for them.



Round the following number to the **hundredths place**.

8.227

Type your answer below (mathematical expression):

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 1: ASSSTment Tutor

Once a student has mastery status for a problem set in the ASSISTment tutor, the problem set becomes eligible for the Automatic Reassessment and Relearning system. If the feature is enabled, students are automatically given reassessments tests to test these skills in a mixed review format.

4. Project Overview

The Automatic Reassessment and Relearning System was three term project split into three distinct phases. The first term was spent studying large existing code base of ASSISTment, understanding the requirements of the system, and designing an implementation that would extend the mastery learning of the ASSISTment tutor. During the second term, the ARRS was implemented. The last term was spend testing, fixing errors, and deploying the code first on test servers and then production.

The first step of the design phase was to understand the idea of spaced learning and study the ASSISTment codebase. The project idea was explained at weekly meetings with the project advisor, Professor Heffernan. Technical implementation and how to best design the project for the ASSISTment tutor was discussed during these weekly advisor meetings in addition to meetings with members of the ASSISTment lab. The lab members were helpful in understanding the requirements from the teacher for the project and how teacher might practically use the system. Insight was given on what features teachers would and would not want such as the recommendation to remove an option that forces students to complete tests first. The ASSISTment lab members also gave direction to which code files I should begin to look into first and what files I would need to create or modify for the project. There were some meetings with various lab members to help understand portions of the ASSISTment codebase.

Part of the design was to understand what requirements were important to the project. The end users of the project are teachers, students, and researchers. Teachers needed the ability to easily review student progress so multiple types of reports were created to attempt to highlight different aspects of the system. The mixed review test needed to be displayed to students in similar manner that the ASSISTment tutor did so it would be familiar. However presenting multiple problems from multiple different assignments was not a feature supported yet in the ASSISTment tutor. Researchers needed the ability to control parameters the system so various settings were designed from them to control for future experiments using the ARRS.

It was important to the design that the project cause minimal impact to the existing codebase. It was advised to separate the new code as much as possible to ease debugging and potential complications. The system required additional information to be stored so multiple new models and database tables were designed. New controllers for the scheduling logic used these database entries were needed. Mock pictures of how the views should look for all of the new pages were also created before implementation. The v iews needed to maintain the look and feel of the ASSISTment website while also providing clear information about the functionality of the new web pages to new users.

The implementation of the Automatic Reassessment and Relearning System was completed during the second term of the project. At the end of the term, twenty-four files were added and fifteen existing files were modified to the ASSISTment codebase. The implementation included a major change to the existing codebase as a new column was added to the very important class_assignments table. The column contains a reference to another table and was used to distinguish between normal assignments and the new relearning assignments.

The ASSISTment lab members aided in solving problems implementing the project. Advice on what difficulties were worth my time trying to understand such as a confusing progress table of hashs and suggestions of alternate solutions on implementation.

The final phase of the project was concerned with code quality. Code reviews were conducted in group fashion with the ASSISTment lab. Multiple tests were conducted to discover and fix errors that were encountered. There was difficulty in testing due to the time component of the project, but a rubygem was used to control time. Feedback was given from the ASSISTment lab on changes to the user interface for clarity. The project was committed first to a test server then production after a week of testing.

5. Project Design

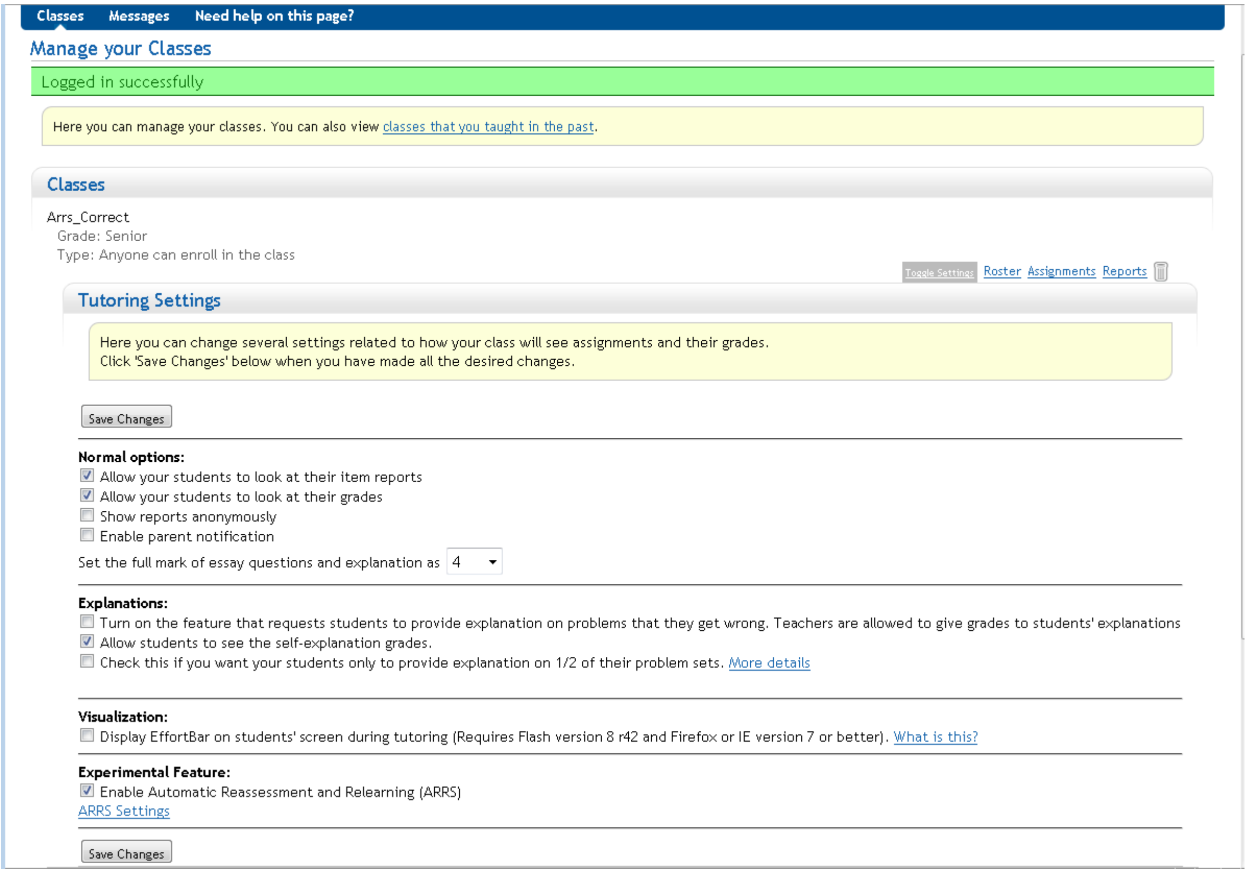
5.1 Overview

The Automatic Reassessment and Relearning System was created for use by the teachers and students using the ASSISTment tutor. The technology supports teachers to assist them by doing the work of scheduling reassessment tests and aid students in relearning. The feature is designed to extend the ASSISTment mastery learning. There are four parts described here of the Automatic Reassessment and Relearning System, the settings chosen by the teacher or researcher, the reassessment tests taken by the students, relearning assignments given to students who fail the reassessment tests, and new reports created to show teachers the progress of students.

Teachers or researchers are able to set parameters for the system such as controlling the dates of the reassessment schedule. The scheduling logic uses these dates to determine when to create reassessment tests for the students. The reassessment tests contain problems from past mastery learning problem sets that are eligible based off of a variety of conditions. Students then may have to complete relearning assignments which require students to re-master forgotten skills using the existing ASSISTment tutor. Reports are generated for teachers to view the progress of students.

5.2 ARRS Settings

The Automatic Reassessment and Relearning System is a class based feature, which means it requires teachers to enable it in each class's settings to turn on. The ARRS has an additional settings page for the parameters of the system to be set.



Manage your Classes

Logged in successfully

Here you can manage your classes. You can also view classes that you taught in the past.

Classes

Arrs_Correct
Grade: Senior
Type: Anyone can enroll in the class

Tutoring Settings

Here you can change several settings related to how your class will see assignments and their grades.
Click 'Save Changes' below when you have made all the desired changes.

Normal options:
☑ Allow your students to look at their item reports
☑ Allow your students to look at their grades
□ Show reports anonymously
□ Enable parent notification
Set the full mark of essay questions and explanation as 4

Explanations:
□ Turn on the feature that requests students to provide explanation on problems that they get wrong. Teachers are allowed to give grades to students' explanations
☑ Allow students to see the self-explanation grades.
□ Check this if you want your students only to provide explanation on 1/2 of their problem sets. More details

Visualization:
□ Display EffortBar on students' screen during tutoring (Requires Flash version 8 r42 and Firefox or IE version 7 or better). What is this?

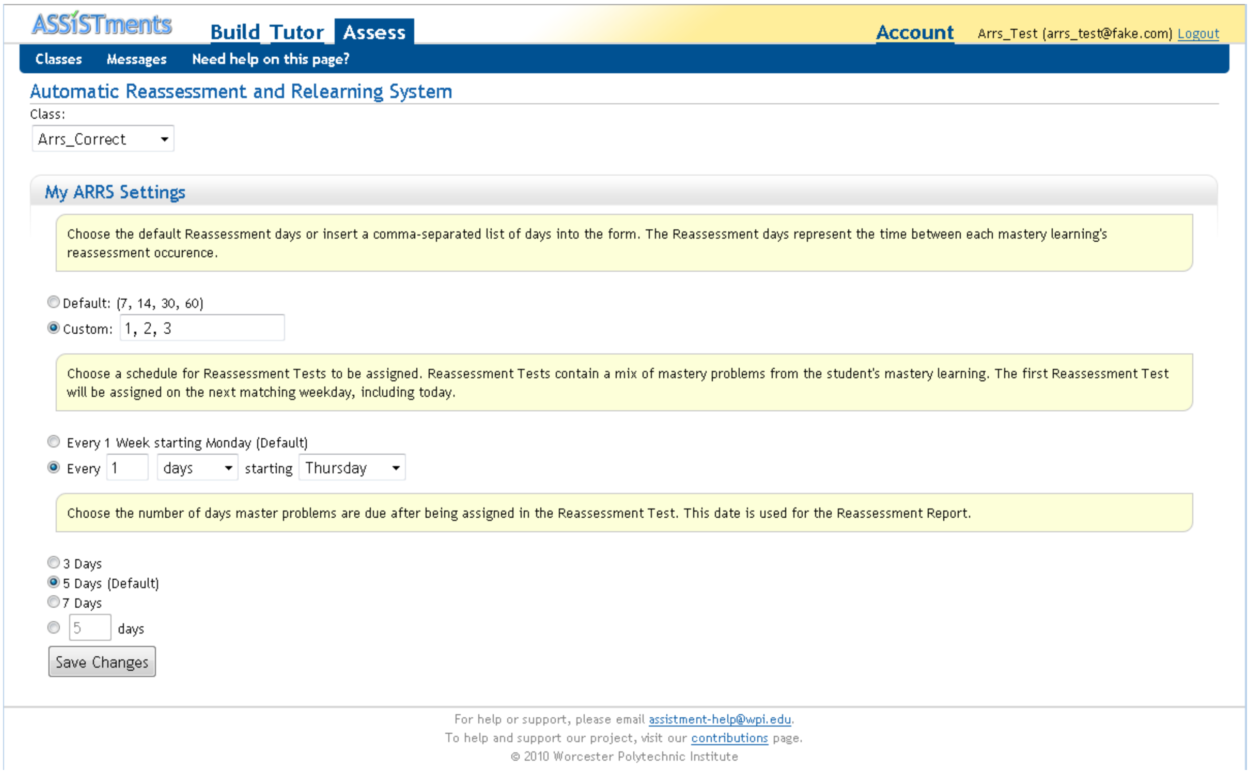
Experimental Feature:
☑ Enable Automatic Reassessment and Relearning (ARRS)
ARRS Settings

Figure 2: Class Settings Page

The first setting is the reassessment days, meaning the number of days after a mastery learning skills are tested before it will be eligible to be retested again. This allows teachers to choose the spaced retention intervals. The default values for this setting are “7, 14, 30, 60”. When students answer question correctly on the reassessment test, they advance to the next time intervals of days in this setting. For example, if a student gains initial mastery of a skill on day 1, that skill will be eligible for retesting 7 days later. On that test if the student answers correctly, he will not be retested for another 14 days. However, if the student answers incorrectly, the skill can be retested again 7 days later because the student did not advance in the schedule. Teachers can input any list of integer values to control the period of time between testing skills. A student completes the ARRS when the student advances past the last spacing level and the student not retested again on the skill.

Teachers can also set the schedule for reassessment tests. The schedule is determined by a frequency and period starting on some weekday. For example, it may be set to every 1 week starting on Monday. Reassessment tests will then be created once per week on Monday. The reassessment test schedule was created to allow teachers to know when students need to log into the ASSISTment tutor to be tested.

Additionally, the number of days students are given to complete the relearning assignments is settable by the teachers. Relearning assignments are created when the student completes the reassessment tests for problems which the student has failed. The due dates for these relearning assignments are set to n number of days later which is determined by this setting. The due dates are used for the reports that are viewable by the teachers and in the progress shown in the student’s show assignments page.



Arrs_Test (arrs_test@fake.com) Logout

Automatic Reassessment and Relearning System

Class: Arrs_Correct

My ARRS Settings

Choose the default Reassessment days or insert a comma-separated list of days into the form. The Reassessment days represent the time between each mastery learning's reassessment occurence.

○ Default: (7, 14, 30, 60)
● Custom: 1, 2, 3

Choose a schedule for Reassessment Tests to be assigned. Reassessment Tests contain a mix of mastery problems from the student's mastery learning. The first Reassessment Test will be assigned on the next matching weekday, including today.

○ Every 1 Week starting Monday (Default)
● Every 1 days starting Thursday

Choose the number of days master problems are due after being assigned in the Reassessment Test. This date is used for the Reassessment Report.

○ 3 Days
● 5 Days (Default)
○ 7 Days
○ 5 days

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

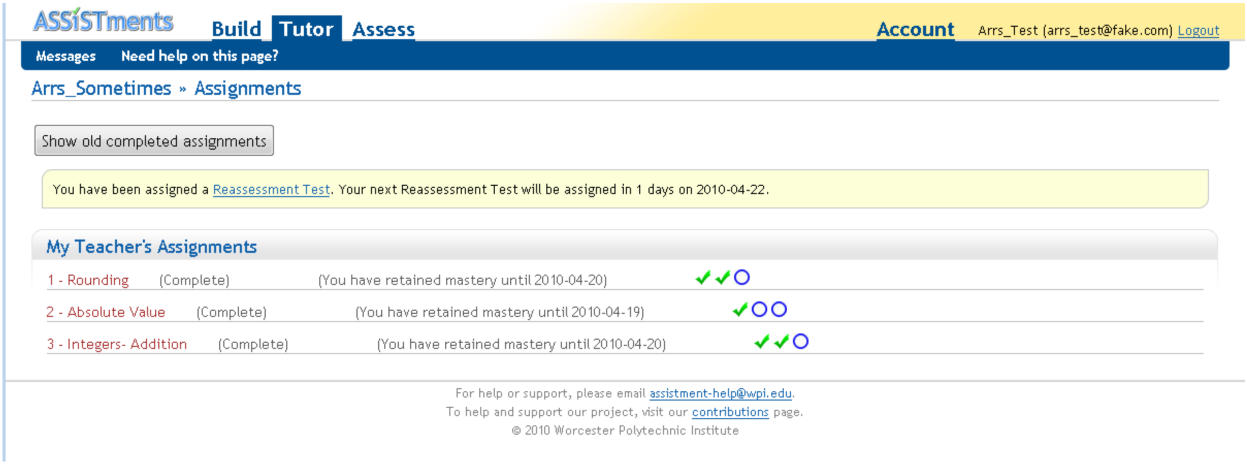
Figure 3: ARRS Settings Page

There were a few settings that have been removed from the initial design. One option was to force student to complete reassessment tests before any other assignments. Discussion with members of the ASSISTment lab who work closely with teachers advised that this would be an unwanted feature. Teachers may want students to complete a particular assignment during class and would not want the ARRS system from preventing students to work on it. Another feature was to set a number of days a student would have to complete the reassessment test before it would be considered late. The system has now been designed such that reassessment tests would be taken the day they are created and students would be aware of when tests are assigned by the teacher.

5.3 Reassessment Test

The reassessment test is a mixed review made up of problems eligible to be retested. Problems are eligible to be on the test based from various criteria. Problem sets must be of the mastery learning type and not a relearning assignment, the student must have initially mastered the skill, and it must be eligible based on the reassessment spacing schedule. The student receives no tutoring help such as hints or feedback until the test is submitted. The reassessment test is the automated periodic retesting to ensure that students have retained of past knowledge.

The scheduling logic is performed when a student views his show assignments page. The system uses the class's schedule for reassessment tests to determine if students of the class have a reassessment test to complete. Students are shown links to take the reassessment test if it has not yet been completed.


ASSISTments
Build
Tutor
Assess
Account
Arrs_Test (arrs_test@fake.com) Logout
Messages
Need help on this page?
Arrs_Sometimes » Assignments
Show old completed assignments

You have been assigned a Reassessment Test. Your next Reassessment Test will be assigned in 1 days on 2010-04-22.

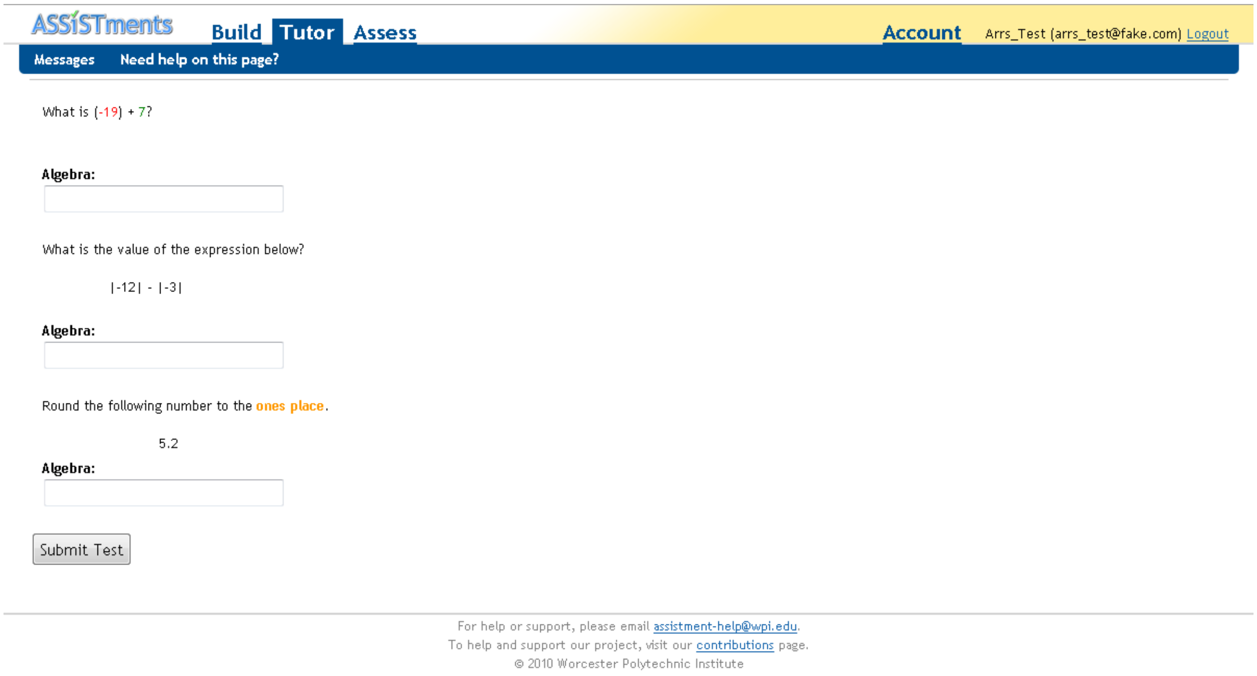
My Teacher's Assignments

1 - Rounding	(Complete)	(You have retained mastery until 2010-04-20)	✓✓○
2 - Absolute Value	(Complete)	(You have retained mastery until 2010-04-19)	✓○○
3 - Integers- Addition	(Complete)	(You have retained mastery until 2010-04-20)	✓✓○

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 4: Show Assignments Page

The reassessment test is a mixed review, meaning that the problems are presented to student in a random order so they may not know which skills are being retested. The test needed to be displayed in a similar format to the ASSISTment tutor, but the project could not use it for the reassessment test. The ASSISTment tutor was not able to support multiple problems from multiple different assignments at once. In addition, the reassessment test needed to offer no tutoring aid, not inform students what skill is being test and not show the correctness of answers until all of the problems were completed. A new page was designed to mimic the actions of the ASSISTment tutor to simulate the completion of multiple problems from multiple assignments simultaneously. This was done by mimicking the Ajax calls of the ASSISTment tutor. The reassessment page only uses the tutor files from the normal ASSISTment tutor to evaluate the correctness of answers.



What is (-19) + 7?

Algebra:

What is the value of the expression below?

|-12| - |-3|

Algebra:

Round the following number to the ones place.

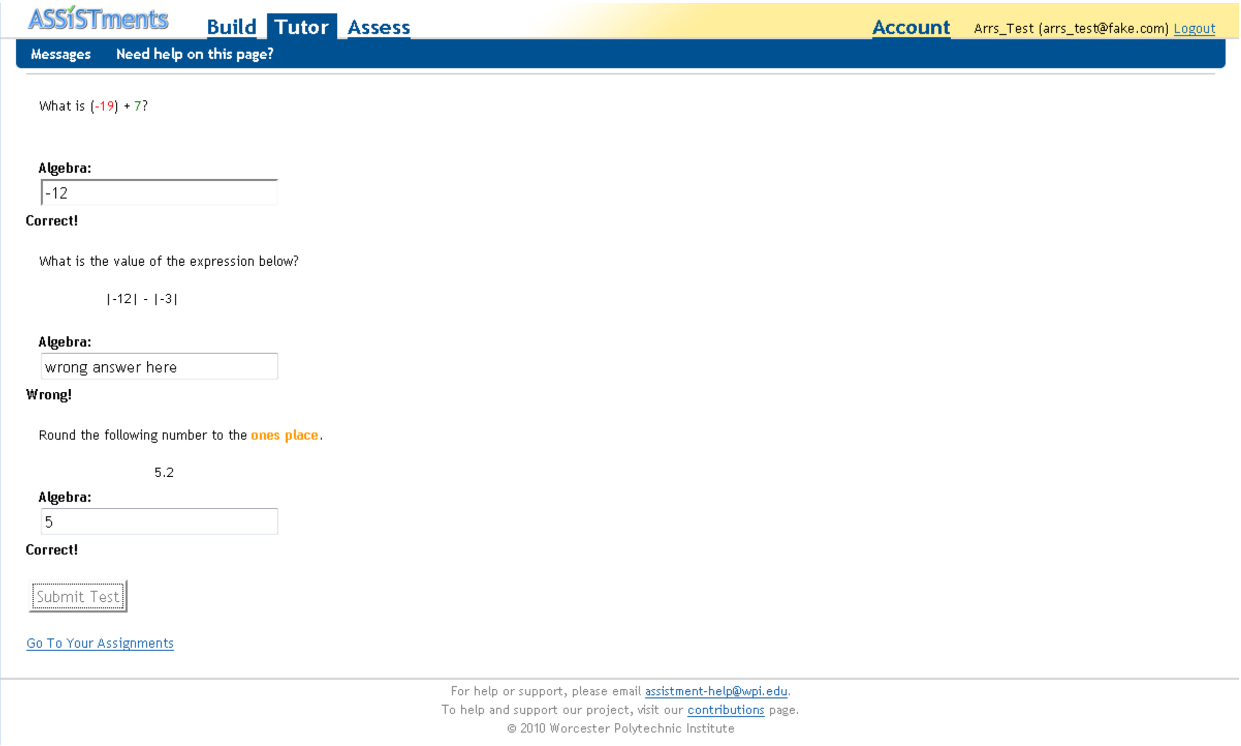
5.2

Algebra:

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 5: Reassassment Test

After submitting the tests, the student is told which problems were answered correctly and shows a link to the show assignments page. In the show assignments page, the student can view how far the student is in the reassessment schedule through visual, until what date the student has retained mastery for problems answered correctly, and the relearning assignments for problems answered incorrectly on the reassessment test.



What is (-19) + 7?

Algebra:
-12

Correct!

What is the value of the expression below?

|-12| - |-3|

Algebra:
wrong answer here

Wrong!

Round the following number to the ones place.

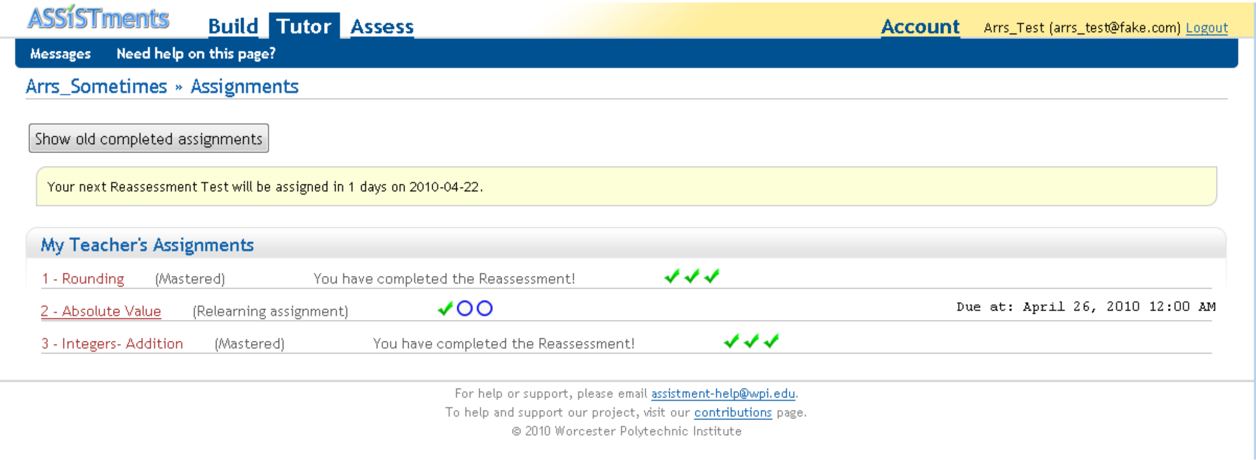
5.2

Algebra:
5

Correct!

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 6: Submitted Reassessment Test



ASSISTments
Build
Tutor
Assess
Account
Arrs_Test (arrs_test@fake.com) Logout
Messages
Need help on this page?
Arrs_Sometimes » Assignments
Show old completed assignments

Your next Reassessment Test will be assigned in 1 days on 2010-04-22.

My Teacher's Assignments

1 - Rounding	(Mastered)	You have completed the Reassessment!	✓✓✓	
2 - Absolute Value	(Relearning assignment)		✓○○	Due at: April 26, 2010 12:00 AM
3 - Integers- Addition	(Mastered)	You have completed the Reassessment!	✓✓✓	

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

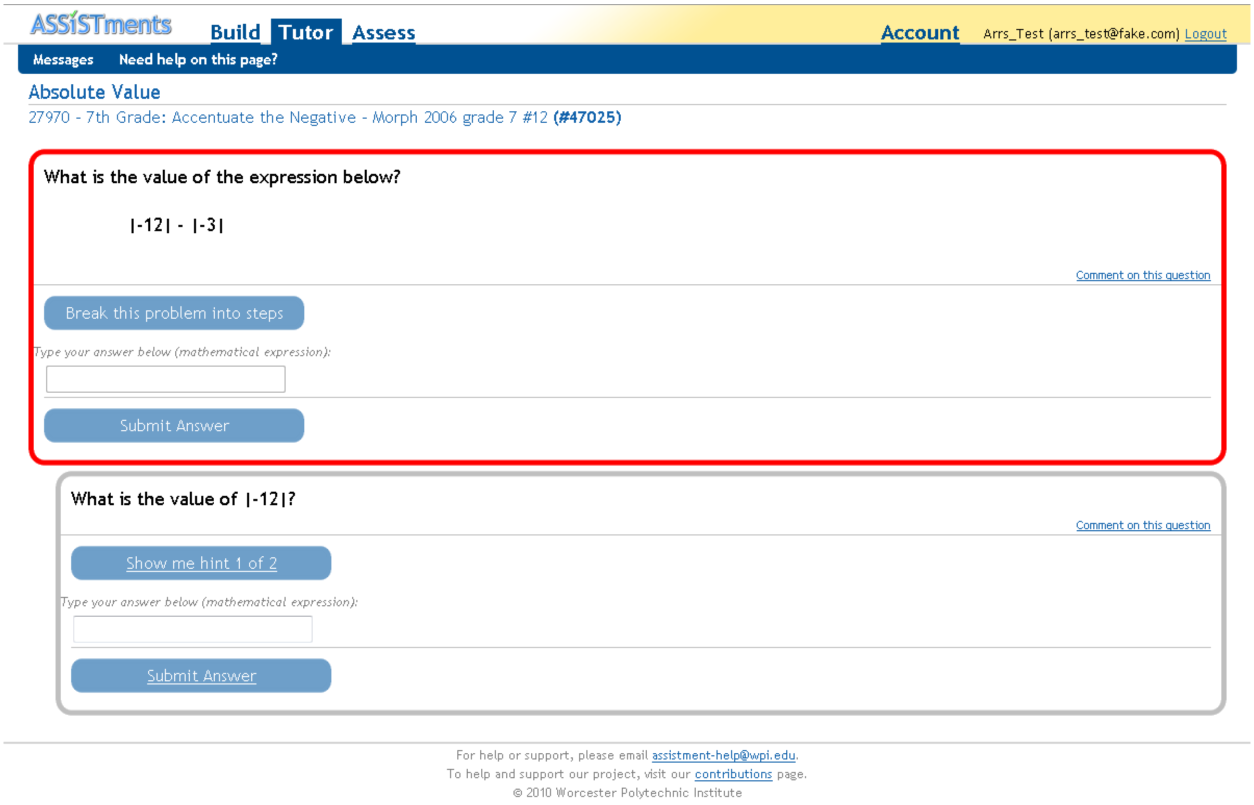
Figure 7: Show Assignments Page after Retest

The reassessment test was designed with the practicality of how teachers would use the system. It would be unreasonable for students to be required to log into ASSISTment to check if they have a retest on a skill every day. The reassessment test amasses various skills together to be retested on a known date. Teachers can set up weekly retest schedules and inform students to do the reassessment test on particular dates. It will also be easier for teachers to determine if students are keeping up with the reassessment tests because reports are able to display who has completed a particular test that everyone in the class has been assigned rather than checking many retests of skills on various calendar dates for each student.

5.4 Relearning Assignments

Relearning assignments are created for students that have failed to show retention on the reassessment test. Relearning assignments help student to practice skills until mastery status is re-obtained. New assignments are created for each reassessment test's mastery learning problem sets that some student in the class answers incorrectly. Since assignments are class wide, students that answer correctly on the retest are considered to have already completed the assignment.

Students complete the relearning assignment using the normal ASSISTment tutor. The relearning assignments contain the same problem given on the reassessment test as the first problem in the assignment. This allows the student to retry the problem with the help of the ASSISTment tutor. When a student completes the relearning assignment as the student would normally to gain initial mastery status, the mastery learning problem set become eligible for retesting again n days later with n determined by the scheduled reassessment spacing interval. Students who complete relearning assignments do not advance to the next level of the reassessment schedule.



Arrs_Test (arrs_test@fake.com)

Absolute Value

27970 - 7th Grade: Accentuate the Negative - Morph 2006 grade 7 #12 **(#47025)**

What is the value of the expression below?

|-12| - |-3|

Type your answer below (mathematical expression):

What is the value of |-12|?

Type your answer below (mathematical expression):

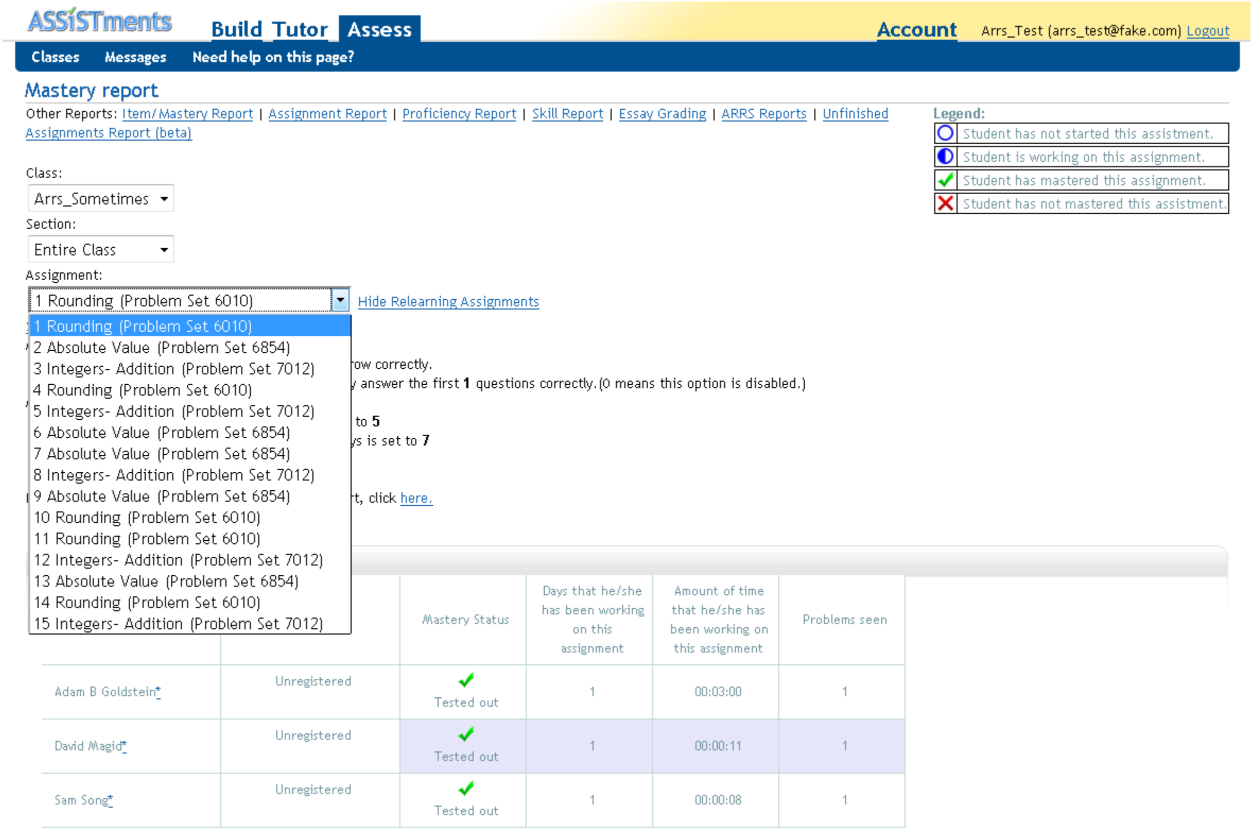
For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 8: Relearning Assignment

A new column was added to the class_assignments table , a reference to another relearning assignment table that creates a relation between the reassessment test the relearning assignment. The column is also used to easily distinguish between relearning assignments and normal assignments. Normal assignments will have a nil value for this column. The addition of the column was a major design change because now relearning assignments must be filtered in various sections of the website so lists are not flooded with repeated assignments.

5.5 Reports

The current teacher reports in ASSISTments are not good enough to report the performance of the ARRS. The old reports show no relation to the teacher of progression between multiple reassessment tests and relearning assignments associated with the same skill. The reports only show individual reports without a broader picture of if students are retaining knowledge through multiple retests being shown. By default the old reports filter the relearning assignments to avoid teachers having to deal with excessively long lists filled with unwanted assignments. There is an option to un-filter the relearning assignments in the item report so teachers may still see a history of actions taken by students on the relearning assignments. However the main views for teachers using the ARRS to gain an understanding of what is happening are the new ARRS reports.


ASSISTments
Build
Tutor
Assess
Account
Arrs_Test (arrs_test@fake.com) Logout
Classes
Messages
Need help on this page?
Hide Relearning Assignments
1 Rounding (Problem Set 6010)
2 Absolute Value (Problem Set 6854)
3 Integers- Addition (Problem Set 7012)
4 Rounding (Problem Set 6010)
5 Integers- Addition (Problem Set 7012)
6 Absolute Value (Problem Set 6854)
7 Absolute Value (Problem Set 6854)
8 Integers- Addition (Problem Set 7012)
9 Absolute Value (Problem Set 6854)
10 Rounding (Problem Set 6010)
11 Rounding (Problem Set 6010)
12 Integers- Addition (Problem Set 7012)
13 Absolute Value (Problem Set 6854)
14 Rounding (Problem Set 6010)
15 Integers- Addition (Problem Set 7012)

Mastery report

Other Reports: Item/ Mastery Report | Assignment Report | Proficiency Report | Skill Report | Essay Grading | ARRS Reports | Unfinished Assignments Report (beta)

Legend:
○ Student has not started this assistment.
◐ Student is working on this assignment.
✓ Student has mastered this assignment.
✗ Student has not mastered this assistment.

Class: Arrs_Sometimes

Section: Entire Class

Assignment: 1 Rounding (Problem Set 6010)

ow correctly.
answer the first **1** questions correctly.(0 means this option is disabled.)

to **5**
s is set to **7**

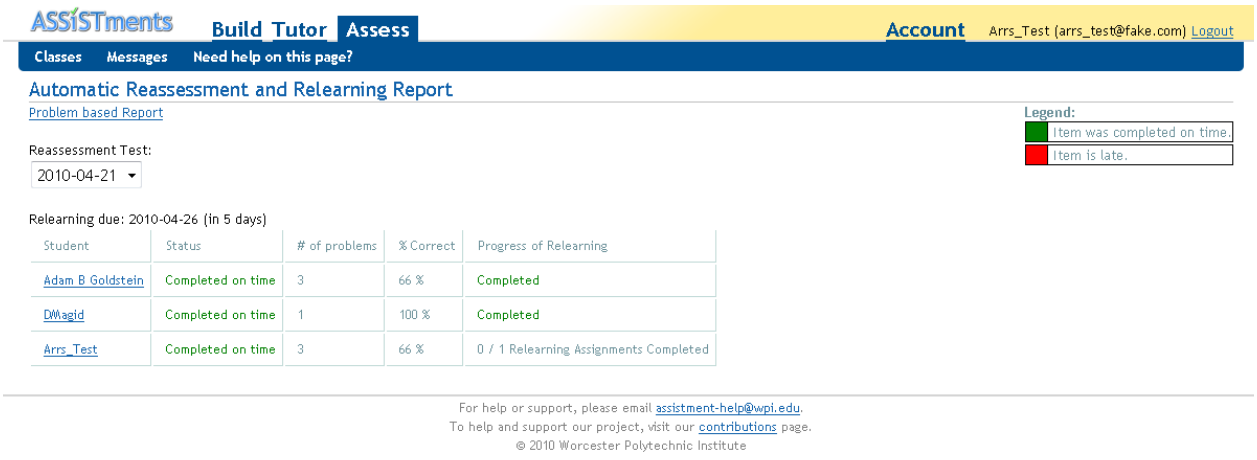
t, click here.

		Mastery Status	Days that he/she has been working on this assignment	Amount of time that he/she has been working on this assignment	Problems seen
Adam B Goldstein*	Unregistered	✓ Tested out	1	00:03:00	1
David Magid*	Unregistered	✓ Tested out	1	00:00:11	1
Sam Song*	Unregistered	✓ Tested out	1	00:00:08	1

Figure 9: Un-filtered Mastery Report

New reports viewable only by the teacher were created to show student's progress and performance in the Automatic Relearning and Reassessment System. The reports are meant to highlight students that have difficulty retaining knowledge to teachers so they may investigate further and offer addition help to the student. Also the reports attempt to display mastery learning problem sets which may be difficult or have low percentages of retention. There are three report pages which are the following, the reassessment test report, individual student reports, and the problem based report.

The reassessment test report is the default report viewable by the teacher. The report shows the results for each of the students on a given reassessment test with the page defaulting to the most recent. Teachers can select page reassessment test reports to view. The students are listed first those who have not completed the retest. The status of student can be "not completed" or "completed". The report page displays the number of problem given to each student, the percentage of problems answered correctly, and the progress of the student completing relearning assignments. The report is designed to help teacher identify students that may be having difficulties retaining knowledge and need extra help learning.



Automatic Reassessment and Relearning Report

Problem based Report

Legend:
🟩 Item was completed on time.
🟥 Item is late.

Reassessment Test:
2010-04-21

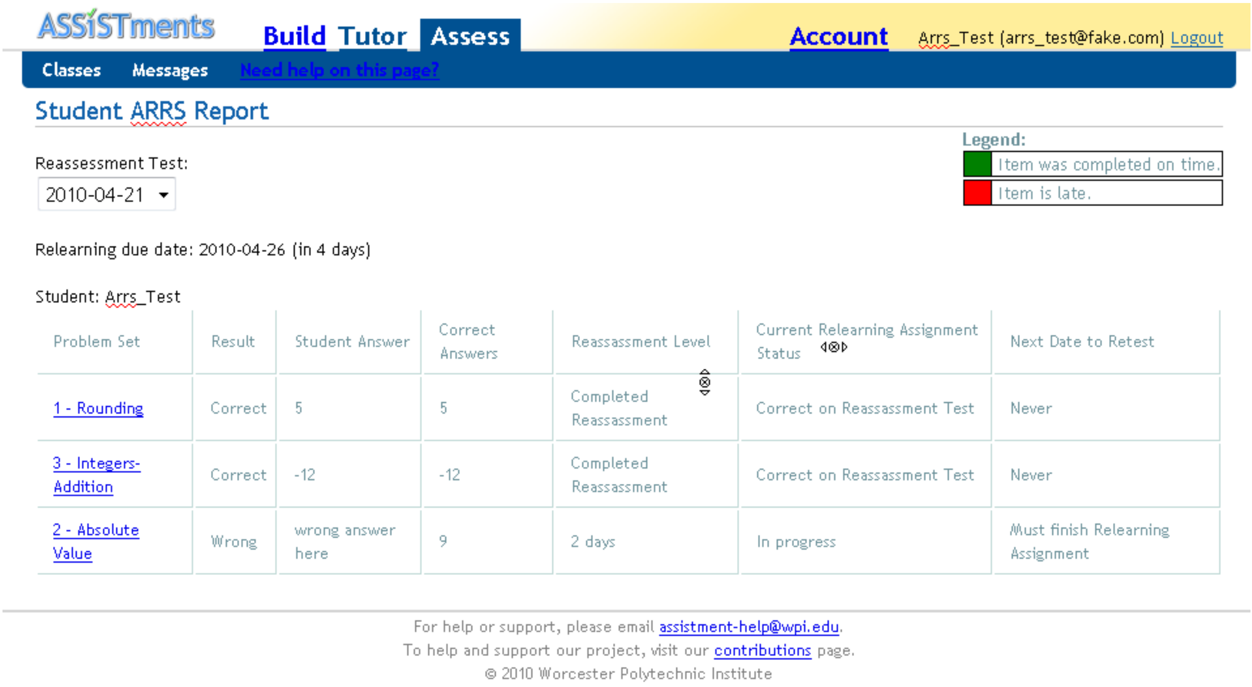
Relearning due: 2010-04-26 (in 5 days)

Student	Status	# of problems	% Correct	Progress of Relearning
Adam B Goldstein	Completed on time	3	66 %	Completed
DMagid	Completed on time	1	100 %	Completed
Arrs_Test	Completed on time	3	66 %	0 / 1 Relearning Assignments Completed

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 10: Reassessment and Relearning Report

The student ARRS report allows teacher to look at the details of a particular student's performance. The report displays a table showing each of the problems tested on a given reassessment test. Additionally the report displays the student's answer, the correct answer, the level in the reassessment schedule, current relearning assignment status, and date that a particular mastery learning problem set may be retested. The teacher is also able to view the tested problems using a preview feature in the ASSISTment tutor. The student ARRS report allows teachers to investigate the performance of specific students which may help in determine exactly where the students are having difficulties.



Student ARRS Report

Legend:
🟩 Item was completed on time.
🟥 Item is late.

Reassessment Test:
2010-04-21

Relearning due date: 2010-04-26 (in 4 days)

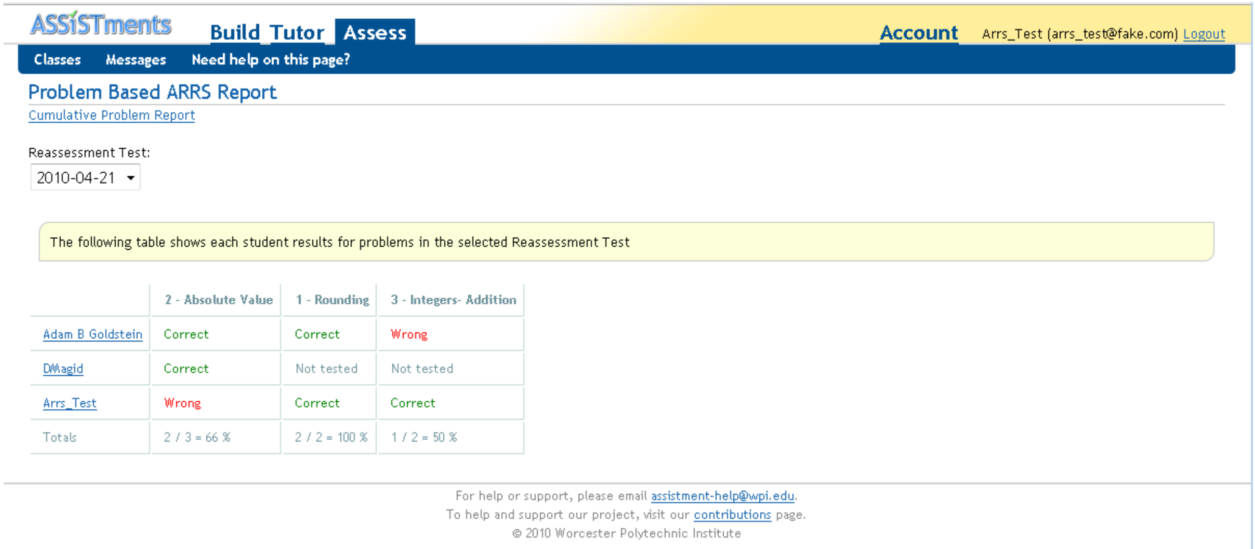
Student: Arrs_Test

Problem Set	Result	Student Answer	Correct Answers	Reassessment Level	Current Relearning Assignment Status	Next Date to Retest
1 - Rounding	Correct	5	5	Completed Reassessment	Correct on Reassessment Test	Never
3 - Integers- Addition	Correct	-12	-12	Completed Reassessment	Correct on Reassessment Test	Never
2 - Absolute Value	Wrong	wrong answer here	9	2 days	In progress	Must finish Relearning Assignment

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 11: Student ARRS Report

The problem based report was created to aid in identify mastery learning problem sets that may be more difficult for student to retain. For a given reassessment test, the report displays which students answered each problem tested correctly and total percentages of correctness for each problem.



Problem Based ARRS Report

Cumulative Problem Report

Reassessment Test:
2010-04-21

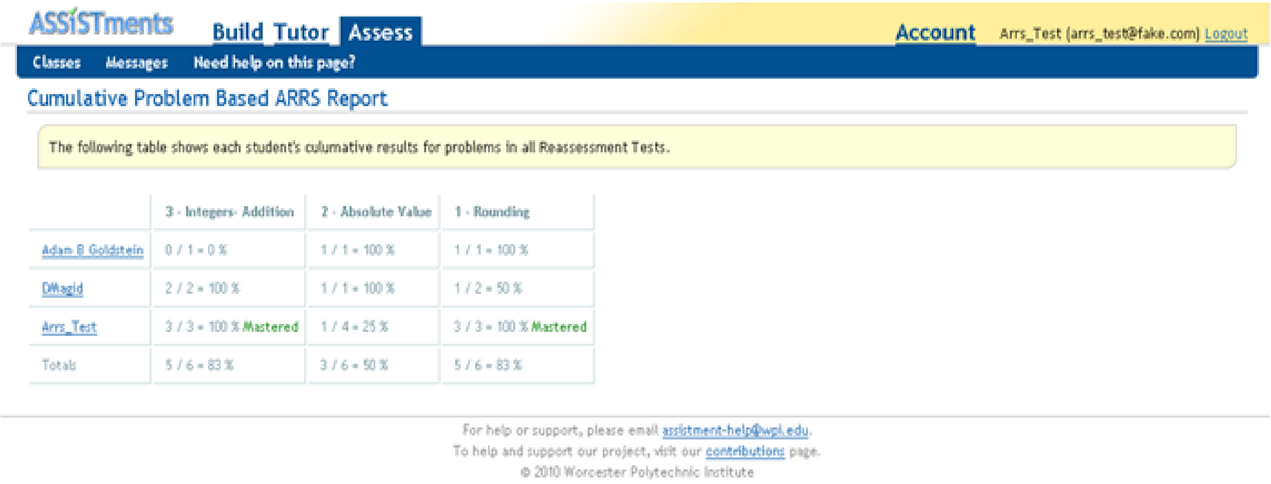
The following table shows each student results for problems in the selected Reassessment Test

	2 - Absolute Value	1 - Rounding	3 - Integers- Addition
Adam B Goldstein	Correct	Correct	Wrong
DMagid	Correct	Not tested	Not tested
Arrs_Test	Wrong	Correct	Correct
Totals	2 / 3 = 66 %	2 / 2 = 100 %	1 / 2 = 50 %

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 12: Problem Based Report

There is also a cumulative view of this report which displays each student history for each problem set. The cumulative report shows percentages for each student for answer each mastery learning problem set and total percentages of all reassessment tests. Teachers can view what the overall percentage of correctness is for each of the problems.



Cumulative Problem Based ARRS Report

The following table shows each student's culumative results for problems in all Reassessment Tests.

	3 - Integers- Addition	2 - Absolute Value	1 - Rounding
Adam B Goldstein	0 / 1 = 0 %	1 / 1 = 100 %	1 / 1 = 100 %
DMagid	2 / 2 = 100 %	1 / 1 = 100 %	1 / 2 = 50 %
Arrs_Test	3 / 3 = 100 % Mastered	1 / 4 = 25 %	3 / 3 = 100 % Mastered
Totals	5 / 6 = 83 %	3 / 6 = 50 %	5 / 6 = 83 %

For help or support, please email assistment-help@wpi.edu.
To help and support our project, visit our contributions page.
© 2010 Worcester Polytechnic Institute

Figure 13: Cumulative Problem Based Report

6. Technical Implementation

6.1 Overview

This section is intended for the members of the ASSISTment lab who will have to maintain and possibly extend the Automatic Reassessment and Relearning System. The ARRS was mostly implemented using Ruby on Rails, Ajax, and rhtml. The project was originally named spiraling, but was later changed to ARRS. The view displayed to the users has been renamed to ARRS, but spiral is still used in the source code. Spiral and ARRS have the same meaning.

The implementation follows the model, view, controller architecture of Ruby on Rails. I attempted to separate the ARRS code from the existing codebase as much as possible to avoid technical complications in debugging. In addition, merging onto the production will be easier with separated code as ASSISTment is in active development. The result is that the majority of the project files are new with minimally changes to the existing code. Other technologies used were a local postgres database and svn to the ASSISTment repository. File and method names are shown in italic for clarity.

6.2 Existing Modifications

There is a total of fifteen file modifications to the existing source. The following section will describe the changes. The *open_response_controller*, *report_controller*, *report_controller*, and *report_helper* were edited to filter relearning assignments. The filtering of relearning assignments was not encapsulated into one method because of the difference in database queries used in the files. The *student_class_controller* filters relearning assignments and adds the enable_spiral option which turn on ARRS for a class. When enable_spiral is true, the *spiralsettings* for the class is created with default settings if it does not exist. It is create here because a user may go to the ARRS report page, show assignments page, or ARRS settings which all require this table entry.

The existing tutor settings page *_tutor_settings.rhtml* was slightly modified with the addition of a check box to enable the ARRS system for each individual class under a new section labeled 'Experimental Feature'. An *enable_spiral?* method was added to the *student_class.rb*. e*nable_spiral* is a new parameter to the student_class options. The default for this setting is set to false. The *_report_menu.rhtml* was modified to create a link to the ARRS Reports if the class has the *enable_spiral* set to true.

The *class_assignment_controller* was edited to merge the original assignment and relearning assignments onto one line. The controller creates an array of the original assignments which are the assignments that would be shown if ARRS was disabled. Another array of relearning assignments is created by filtering non-relearning assignments, incomplete assignments. Also relearning assignments that were created for students who have completed the reassessment test which have been assigned to the whole class are hidden to those who have not yet completed the reassessment test. The array is reversed because the relearning assignments will always be assigned in order. The array of relearning assignments is used to replace the assignment in the original assignment array. A hash is used with the key being the relearning assignment id and value the original assignment id. This hash is used to retain the naming of the original assignments so the original numbering of the assignments is displayed.

The *_class_assignment.rhtml* was modified to display the changes made in the *class_assignment_controller*. The show assignments page was changed to only display the most recent relearning assignment for mastery learning problem set in the ARRS. The show assignments page also displays additional information for relearning assignments such as dates that it will be eligible to be retested and visual check marks to show progress in the schedule. The link the reassessment test is displayed on the show assignment page.

The *_class_assignments.rhtml* was modified to show an information message if the class has *enable_spiral*. The help message includes the date of the next reassessment test and a link to take the test if one has already been assigned. In addition the code calls the *hasSpiralTest?* method which calculates the test schedule and create new *spiral_test* entries in the database. The *_class_assignment_list.rhtml* was modified to be more efficient with database queries.

The most significant change the existing ASSISTment tutor codebase is the addition of a column named spiral_assignment_id to the *class assignment* model in *class_assignment.rb*. This column will refer to an entry in the spiral_assignments table if the class assignment is a relearning assignment. The value in this column is nil for normal assignments. The reason the column was needed is to create a relation from the relearning assignments to the reassessment tests and original class assignment that was being retested.

There effect of this change to existing code the addition of filtering relearning assignments in existing areas such as reports views. If the relearning assignments were not filtered these report pages may contain an excess number of assignments. *mastery_status.rthml* was revised to allow teachers the option to un-filter the relearning assignments in the item report. Teacher can also use the new ARRS reports to get a better understanding of student performance.

The *tutor.js* page was edited for a condition check in case assignments are undefined. The reassessment test uses the tutor.js to evaluate the correctness of answers. However the test is not assignment based so javascript assignment objects are not defined when the tutor attempt to log problems. The tutor automatically attempts to log problems when it evaluates the correctness of an answer. The ARRS view *start* for the test does this logging the tutor.js would.

6.2 New Models

There are several new tables added the ASSISTments for the ARRS. Three migrate files 135-137 add the following tables:

> spiral_settings: There is an entry for each student class with ARRS enabled. The table stores the settings for each student_class. The table stores the spacing days as text containing comma separated integers in the column *days*. The reassessment test schedule is stored in three separate columns containing integers for the weekday, period, and frequency in *schedule_weekday, schedule_period,* and *schedule_frequency*. The number of day until relearning assignments are due is also stored as an integer in *num_days_due*.
>
> spiral_tests: There is an entry for each reassessment test for each student_class. The table stores dates in which reassessment tests are scheduled in the *spiral_date* column. There is a column to containing a *student_class_id* to associate classes with the spiral test.
>
> spiral_logs: There is an entry for each student's answer for each reassessment test identified by the *user_id, spiral_test_id* and *class_assignment_id*. The table contains the results for each student's answer to each reassessment test problem in the *result* column and the student's current progress in the reassessment schedule for the given problem in the *mastery_level* column. This table also contains the *next_date* that this problem will be eligible to be retested. The next_date column is nil for failed tests until the relearning assignment is completed.
>
> spiral_student_test: There is an entry for each time a student completes a reassessment test. The existence of an entry in this table indicates that the student has completed the reassessment test. The entries in this table are identified by the *spiral_test_id, user_id,* and *spiral_date.*

spiral_assignments: There is an entry for each different mastery problem set that a student answers for each reassessment test identified in *spiral_test_id*. This table associates a relearning assignment the original assignment and the reassessment test with the columns *class_assignment_id*. *class_assignment_id* will refer to the id of the original mastery learning assignment. The new column in *class_assignment* table named *spiral_assignment_id* references this table. The *spiral_assignment_id* will be nil for non spiral_assignments.

Model files were created as ActiveRecords to maintain persistence in the database for each of the new database tables. The following is a description of additional methods added to the model files.

SpiralSettings: The important methods in spiralsettings to clarify are the *radio_spiral_days?*, *radio_num_days_due?,* and *radio_schedule?*. These methods are used for the radio buttons on the spiral settings web page to select the default or custom input. *getDefaultSpiralDays* returns the default settings for spiral days.

SpiralTest: This model is the ActiveRecord for the *spiral_test* table.

Spirallog: *next_date?* method is used to set the *next_date* column of spiral logs that have been set to nil because students answered the reassessment test incorrectly. The *next_date* column for students that fail the reassessment test need to be the date in which the relearning assignment is completed plus the reassessment period. Whenever a student uses the *show_assignments* page, *next_date?* is called to check if the relearning assignment has been completed yet and if so replace the nil value in the *next_date* column with the appropriate value. The *get_relearning_assignment* is a helper function for *next_date?* that returns the relearning assignment associated with this spirallog.

SpiralStudentTest: This model is the ActiveRecord for the *spiral_student_test* table.

SpiralAssignment: This model is the ActiveRecord for the *spiral_assignment* table.

6.3 ARRS Tutor

The following section describes the controller and view files for the tutor side of the Automatic Reassessment and Relearning System. The majority of the logic for the ARRS for the tutor is in the *arrs_test_controller.rb* file.

The *hasSpiralTest?* method in *arrs _test_controller* determines if the link to the reassessment test should be displayed on the student's show assignments page. The method checks if a new spiral_test needs to be created based on the class's ARRS settings. *hasSpiralTest?* will continually create *spiral_tests* entries for past tests until a *spiral_test* with a date greater than the current date is created. This is to maintain the correct schedule in case none of the students log in for extended periods of time. The method will return true if the student has a *spiral_test* and has not completed it yet (which is determined by the existence of a corresponding *spiral_student_test* database entry). This method is executed whenever a student loads the show assignments page.

The *start* method of the *arrs_test_*controller creates an array of assignments that will be on a given reassessment test. First all of the class assignments are collected and then filtered based on a variety of criteria such as assignments not yet mastered or not eligible based on schedule. The controller then randomizes the order of the assignments for the mixed review test. Retention assignments are then created as the first problems from each of these assignments are used for the reassessment test. Students who fail a problem in the retest can retry the same problem with the help of the ASSISTment tutor in the relearning assignment.

The *start.rhtml* file is very important for the reassessment test. This file uses Ajax requests to evaluate the answers for the test. The javascript in this file creates problem logs that normally the tutor.js would. The reassessment test does not use the tutor.js the same as the normal ASSISTment tutor because the javascript assignment objects do not exist. The problems are displayed in a random order without labeling of the skill tested. *_problem.rhtml* displays each of the problems because the normal ASSISTment tutor cannot be used. This is because the reassessment test offers no tutoring help for the students and must display one problem from multiple different assignments which is not possible in the normal tutor. The *log_spiral.rjs* shows the students which answers are correct after the test is submited. The *log_student_test* and *log_spiral* methods are called with ajax to create the necessary table entries to record the test results. A count variable is used to know when the Ajax has completed evaluating problems.

If the result of a problem on the test is correct, the *spiral_log* method will create an AssignmentLog so the relearning assignment will be complete for the student. The *spiral_*log will contain the *next_*date if the student answered correctly. Students who answer incorrectly will have to complete the relearning assignment using the normal ASSISTment tutor. Relearning assignments were created by the *start* method so that the first problem will be the same that was tested. After a relearning assignment is completed, the associated *spiral_log* will have the *next_date* column set by the *date?* method in the *spiral_log* model.

It was attempted to fill in the student’s answer into the relearning assignment’s ASSISTment tutor, however it did not work for all problem types and was removed.

6.4 ARRS Teacher

The following section describes the controller and view files for the teacher side of the Automatic Reassessment and Relearning System. The two major components of the ARRS for the teacher are the ARRS settings and the ARRS reports. The *arrs_controller* file contains the logic for both of these parts.

The *arrs_settings* method in *arrs_controller* is used to find the current ARRS settings to pre-fill out in the settings form. *arrs_settings.rhtml* and*_settings.rhtml* renders the *arrs_form* to maintain the look of the ASSISTment website. *arrs_settings.rhtml* ensures that the class has ARRS enabled before editing any of the settings. Teachers can also quickly skip to other classes they have that have ARRS enabled from this page using the selector. The *_arrs_form.rhtml* contains the form for the ARRS settings page. The form contains radio buttons to select default options and text fields to choose custom settings. The submission of the form uses an Ajax call to *set_arrs_settings* to modify the database entry for the class in the *spiral_settings* table. If the *set_arrs_settings* uses default values if the default radio button is selected for any of the options. Otherwise it will use the values entered into the form's text fields. *set_arrs_settings.rjs* redirects teachers back to the main class settings page after submitting settings. If the settings are changed midway through the class, it will not take effect until after the next reassessment test.

There are various ARRS reports each with an associated controller method: *arrs_reports*, *student_arrs_report*, *problem_report*, and *culumative_problem_report*. Each of these controller methods are similar in that they collect information from the database tables created for ARRS. The reports are simply displaying information stored in the database for the teacher. The views for these reports display them in different ways to help highlight different aspects, such as which students having difficulty in the *arrs_reports* or difficult problem sets in the *problem_report*.

arrs_reports.rhtml is the view for the main report. This report displays results for a given reassessment test with the default being the most recent. From this page, teachers can view the student report by clicking the link on each student's name or the problem based report with the link near the top of the page. This report is sorted first by students who have not completed the reassessment test by first loop through and displaying students that are not completed. This report should give a general overview of student performance by seeing each student's status, percentage of correct answers on the test, and their progress towards completing the relearning assignments.

student_arrs_report.html is the view for the student based report. Teachers can investigate individual students to see what may be causing problems. Teachers can see the specific problems asked on the reassessment test by using the preview function of ASSISTment. In addition the correct answer and student's answer are displayed. The student answer is assumed to be the first answer submitted by the student for this problem which is submitted during the reassessment test.

problem_report.rhtml is the view for problem based report. All of the problem sets tested on a given reassessment test are shown with which students correctly answered them. There is a totals value at the bottom to see a percentage of how many students answer the problem correctly. In this report there is a link to other reassessment tests in a selector and to the cumulative problem report. The *cumulative_problem_report.rhtml* is a similar report which uses all of the history for the problem sets across all of the reassessment tests for a broader picture of how successful students are in each problem set. Student's individual percentages for each of the problem sets is also shown in the table.

7. Future Work

The Automatic Reassessment and Relearning System can be built upon for future work to extend the system. The ARRS can be improved by running usability tests in schools to gain feedback from the teachers and students. ARRS could possibility be extended to allow customized schedules to each individuals.

In addition student's performance on the relearning materials could be used to affect the schedule. Students that re-master skills with the minimum number of problems might be handled differently than students that struggle awhile or never re-master the skill after dozens of problem attempts. ARRS currently takes no action if a student takes exceptionally long to complete the relearning assignment other than display it in the reports.

It is not only how much we practice but when we practice that has tremendous impact on our ability to learn and retain information (Pashler, Zarow et al., 2003). Learners that spread their practice across an extended period of time are able to better retain knowledge (Dempster, 1989). Pavlik & Anderson have done work to find how often students should review problems to optimize the rate of learning vocabulary and found that students learning to read receive almost no benefit from reading a word more than once in a single day (Beck & Mostow, 2008). Future work can be done as research experiments using the ARRS to determine if periodically testing students on knowledge that has been mastered will help them retain it.

Although there is literature about the benefits of spaced learning, there is insufficient knowledge to know what will be good time intervals for retention. Experimental questions such as what time intervals will best promote long lasting memory could be investigated. Intervals that are too short will waste time and perhaps even hurt retention while too long will results in students forgetting.

There are numerous research questions that can be investigated by using the ARRS for research experiments such as the following:

How much of an affect does ARRS improve retention of knowledge over no reassassment?

Is there an additional time cost or savings using ARRS? Does periodically retesting save time by retaining knowledge instead of relearning it?

What the most effective time intervals of spacing?

Is some knowledge easier to forget?

Does ARRS accelerate future learning in other skills? Students trained on simpler skills have resulted in improved performance eon more complex skills. (Royer, Tronsky et al., 1999)

Does the ARRS reports help teachers aid students that have trouble retaining knowledge?

8. Conclusions

The Automatic Reassessment and Relearning System can help teachers improve student learning. ARRS allows teachers to implement simple ideas from cognitive science of spaced learning. The ARRS supports the idea that the amount student learn may be fixed, but not the amount of time it takes for students to learn. ARRS will help to advance how and more importantly *when* student learn.

The feature extends the ASSISTment tutoring system and can potentially affect the learning of its user base of thousands of students. The ARRS created in this project also creates the foundation for future work to improve the system and support research experiments to improve learning.

ARRS is a technological solution for teachers, students, and researchers by automatically performing the tedious bookkeeping of a reassessment schedule. The project will hopefully help teachers improve student learning and open possibilities for researchers to gain knowledge on better techniques for learning.

9. References

Beck, J. E. and J. Mostow (2008). How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students. Ninth International Conference on Intelligent Tutoring System: 355-362.

Cepeda, N. J., H. Pashler, et al. (2206). "Distributed practice in verbal recall tasks: A review and quantitative synthesis." Psychological Bulletin **132**: 354-380.

Corbett, A. T. and A. Bhatnagar (1997). Student Modeling in the ACT Programming Tutor: Adjusting a Procedural Learning Model With Declarative Knowledge. Sixth International Conference on User Modeling, Springer.

Dempster, F. N. (1989). "Spacing effects and their implications for theory and practice" Educational Psychology Review **1**(4): 309-330.

Green, L., A. F. Fry, et al. (1994). "Discounting of delayed rewards: A life-span comparison." Psychological Science **5**(1): 33-36.

Koedinger, K. R., J. R. Anderson, et al. (1197). "intelligent tutoring goes to school in the big city." Internation Journal of Artifical Intelligence in Education **8**: 30-43.

Kornell, N. and R. A. Bjork (2008). "Learning concepts and categories: Is spacing the "Enemy of induction"? ." Psychological Science **19**: 585-592.

Pashler, H., G. Zarow, et al. (2003). "Is temporal spacing of tests helpful even when it inflaes error rates?" Jounral of Experimental Psychology: Learning, Memory, & Cognition **29**: 1051-1057

Roediger, H. L. and J. D. Karpike (2006). "Test-enhanced learning – Taking memory tests improves long-term retention." PYSCHOL SCI **17**(3): 249-255.

Rohrer, D. (2009). "The effects of spacing and mixing practice problems." Journal for Research in Mathematics Education **40**: 4-17.

Rohrer, D. and H. Pashler (2007), "Increasing retention without increasing study time." Current Directions in Psychological Science **16**: 183-186.

Rohrer, D. and K. Taylor (2007). "The shuffling of mathematics practice problems improves learning." Instructional Science **35**: 481-498.

Royer, J. M., L. N. Tronsky, et al. (1999). "Math-fact Retrieval as the Cognitive Mechanism Underlying Gender Differences in Math Test Performance." Contemporary Educational Psychology **24**: 181-266.

Underwood, B. J. (1961). "Ten years of massed practice on distributed practice." Psychological Review **68**(4): 229-247.

VanLehn, K. and PSLC (2006). The PSLC Theoretical Framework. www.learnlab.org/clusters/PSLC_Theory_Frame_june_15_2006.pdf.