Proposal: SAGE Dashboard

Ruicong Xie and Tianci Zhong
Columbia University

Fall 2016

1. Introduction

Computational thinking is not just an essential skill for computer scientist. It also has been considered as one of the fundamental skill that is critical to every child's analytical ability(J. M. Wing 2006). Computational thinking involves solving abstract problems, decomposing large complex problem, approximating problems' difficulty, forecasting potential risks and issues, considering the worst case scenarios, controlling the uncertainty and thinking recursively(J. M. Wing 2006). Many instructors are motivated to incorporate computational thinking into their teaching methods, but they are lack of reliable ways to assess this kind of skill.

This project is motivated by the difficulty of quantitatively present the learning progression of students and quantify their degree of computational thinking. It is focused on the visualization of students' study progression and their improvement of computational thinking through the dashboard on SAGE platform to assist both students and instructors. With the dashboard, students can compare their learning progression with other students who are working on the same project or solving the same problem on the gameful learning platform. It also provides useful information of student programming activities that assist instructors by providing real-time help to the students. Instructors can conveniently track each student's progression and keep the student inside the Zone of Proximal Flow. The Zones of Proximal Flow is a concept combining Vygostsky's Zone of Proximal Development and Csikszentmihalyi's state of Flow. The concept of flow is focused on how one can have optimal experience by performing an activity at one's limit. The Zone of Proximal Development is concentrated on what one can achieve with external help. The Zones of Proximal Flow aims at the point to keep the learner continually have optimal experience while pushing one's limit with accessible external help (Murphy, C., Kaiser, G., Loveland, K., & Hasan, S. 2009). In this way, students can have the optimal experience with deep concentration, feeling of enjoyment, and the incentive to explore more.

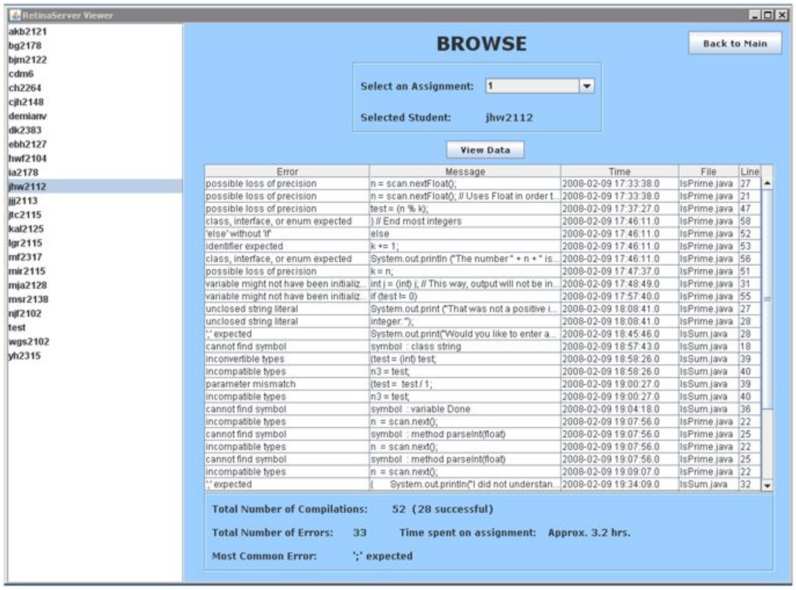
In order to help students reaching such preferable state of flow, instructors need a more automatic and direct visual presentation that reflect the changes in students' learning process. It should be able to detect when a student is stuck at a problem for a relative long period of time and inform the instructor to give real-time help to the student. The dashboard should also assist the instructor to adjust the problems' degrees of difficulty as student progress.

Through a project-first approach, students are more motivated compare to the traditional theory first approach and are able to gain knowledge inside Zones of Proximal Flow. It creates a more nature way for each student push their own limit. This project-first approach also increases the difficulty for instructors to evaluate each student's improvement. To control the assignment's difficult and individualize each assignment, instructors are given the role of game designers who create games with Scratch for students to play while improving their computational thinking skills. Scratch is a programming language with an online platform that enable everyone to create and share their own projects. Scratch opens the opportunity for instructors with no previous programming experience to create games. SAGE extends Scratch by incorporating tooling that collects and analyses data to visualize students learning progression. With these additional tools, instructors gain more control over the whole project and at the same time, students still have the freedom to collaborate and compete through the open online platform of Scratch.

2. Background and Related Work

2.1 Retina

Retina is a tool that collects students' programming activities and processes the collected data to provide helpful information for both students and instructors. It tries to solve the problems such as how much time students spend on each assignment, what errors students made during the assignment, how each students stand in the class compare to other students. Retina collects students' compilation time, compilation error, runtime errors, stack trace and so on. On the instructor side, Retina displays a list of information about each students and aggregate data such as total number of compilation error, most common compilation error, average times spent on the assignment, etc. Figure 1 gives an instructor example view of Retina.



akb2121
bg2178
bjm2122
cdm6
ch2264
cjh2148
demianv
dk2383
ebh2127
hwf2104
ia2178
jhw2112
jjj2113
jtc2115
kal2125
lgr2115
mf2317
mir2115
mja2128
msr2138
njf2102
test
wgs2102
yh2315

Select an Assignment: 1

Selected Student: jhw2112

Error	Message	Time	File	Line
possible loss of precision	n = scan.nextFloat();	2008-02-09 17:33:38.0	IsPrime.java	27
possible loss of precision	n = scan.nextFloat(); // Uses Float in order t...	2008-02-09 17:33:38.0	IsPrime.java	21
possible loss of precision	test = (n % k);	2008-02-09 17:37:27.0	IsPrime.java	47
class, interface, or enum expected	} // End most integers	2008-02-09 17:46:11.0	IsPrime.java	58
'else' without 'if'	else	2008-02-09 17:46:11.0	IsPrime.java	52
identifier expected	k += 1;	2008-02-09 17:46:11.0	IsPrime.java	53
class, interface, or enum expected	System.out.println ("The number " + n + " is...	2008-02-09 17:46:11.0	IsPrime.java	56
possible loss of precision	k = n;	2008-02-09 17:47:37.0	IsPrime.java	51
variable might not have been initializ...	int j = (int) j; // This way, output will not be in...	2008-02-09 17:48:49.0	IsPrime.java	31
variable might not have been initializ...	if (test != 0)	2008-02-09 17:57:40.0	IsPrime.java	55
unclosed string literal	System.out.print ("That was not a positive i...	2008-02-09 18:08:41.0	IsPrime.java	27
unclosed string literal	integer: ");	2008-02-09 18:08:41.0	IsPrime.java	28
';' expected	System.out.print("Would you like to enter a...	2008-02-09 18:45:46.0	IsSum.java	28
cannot find symbol	symbol : class string	2008-02-09 18:57:43.0	IsSum.java	18
inconvertible types	(test = (int) test;	2008-02-09 18:58:26.0	IsSum.java	39
incompatible types	n3 = test;	2008-02-09 18:58:26.0	IsSum.java	40
parameter mismatch	(test = test / 1;	2008-02-09 19:00:27.0	IsSum.java	39
incompatible types	n3 = test;	2008-02-09 19:00:27.0	IsSum.java	40
cannot find symbol	symbol : variable Done	2008-02-09 19:04:18.0	IsSum.java	36
incompatible types	n = scan.next();	2008-02-09 19:07:56.0	IsPrime.java	22
cannot find symbol	symbol : method parseInt(float)	2008-02-09 19:07:56.0	IsPrime.java	25
incompatible types	n = scan.next();	2008-02-09 19:07:56.0	IsPrime.java	22
cannot find symbol	symbol : method parseInt(float)	2008-02-09 19:07:56.0	IsPrime.java	25
incompatible types	n = scan.next();	2008-02-09 19:09:07.0	IsPrime.java	22
';' expected	{ System.out.println("I did not understan...	2008-02-09 19:34:09.0	IsSum.java	32

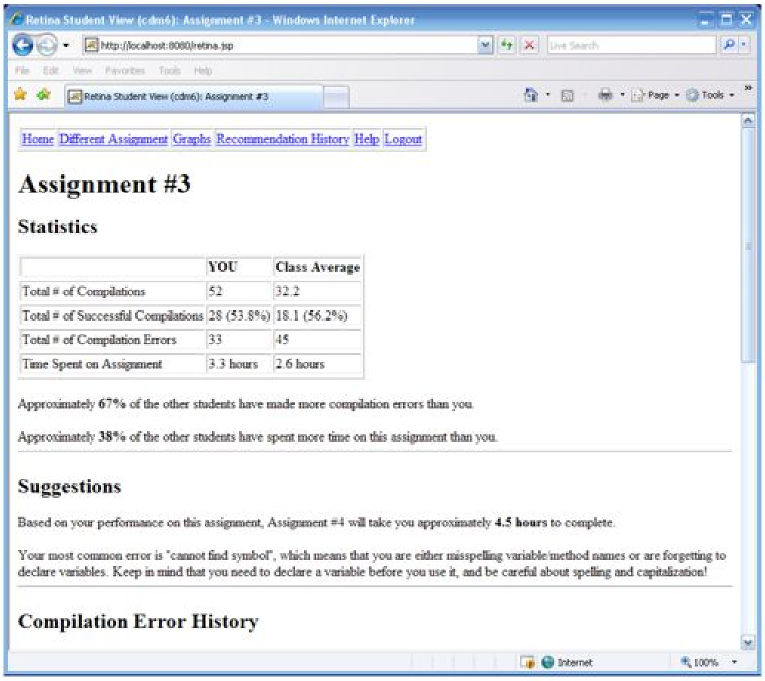
Total Number of Compilations: 52 (28 successful)

Total Number of Errors: 33 Time spent on assignment: Approx. 3.2 hrs.

Most Common Error: ';' expected

Figure 1. Retina Instructor View in Browse Mode

On the student side, Retina shows the number of compilation time and type of errors. It also gives suggestion on expected time consumption on upcoming assignment and give suggestion about the potential errors the students might make. Figure 2 gives a student example view of Retina.


Retina Student View (cdm6): Assignment #3 - Windows Internet Explorer
http://localhost:8080/retina.jsp
Live Search
File Edit View Favorites Tools Help
Retina Student View (cdm6): Assignment #3
Page
Tools
Internet
100%

Home Different Assignment Graphs Recommendation History Help Logout

Assignment #3

Statistics

	YOU	Class Average
Total # of Compilations	52	32.2
Total # of Successful Compilations	28 (53.8%)	18.1 (56.2%)
Total # of Compilation Errors	33	45
Time Spent on Assignment	3.3 hours	2.6 hours

Approximately **67%** of the other students have made more compilation errors than you.

Approximately **38%** of the other students have spent more time on this assignment than you.

Suggestions

Based on your performance on this assignment, Assignment #4 will take you approximately **4.5 hours** to complete.

Your most common error is "cannot find symbol", which means that you are either misspelling variable/method names or are forgetting to declare variables. Keep in mind that you need to declare a variable before you use it, and be careful about spelling and capitalization!

Compilation Error History

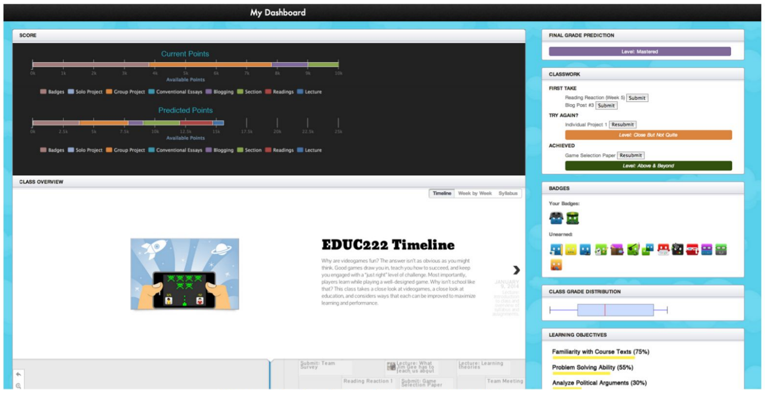
Figure 2. Retina Student View

Students are able to learn from their previous errors with those aggregate data and suggestion given for the upcoming assignment. Retina also displays how many students are currently working on the assignment and creates a committee feels to encourage students that they are not working alone. It is essential to keep students motivated and curious as they learn. We are adopting the similar concept that shows the attendance of the student. If the students logged into the system and tried to complete one of his or her assignment 7 days in a roll, he or she will get a badge to reward his or her effort. To motivate them and indicate that they are making progress, the dashboard will display a progression bar that shows how many subtasks are completed out of the total number of subtasks. By improving the interaction between instructors and students, Retina helps improve instructor's lecture and students' experience with the course. To improve the interaction between instructors and students, the dashboard in SAGE provides real-time information as students working on the assignment and instructors can respond accordingly as students get stuck on the assignment.

2.2 GradeCraft

GradeCraft (Holman, C., Fishman, B., Aguilar, S. 2013) is a custom learning management system that supports game-inspired courses and foreground the affordance of gameful learning environment. The developers of GradeCraft took an list of heavily researched techniques when designing their gameful learning platform. By deploying many simply gameful techniques, the developers established a solid foundation from which to build more nuanced gameful functionality in the future.

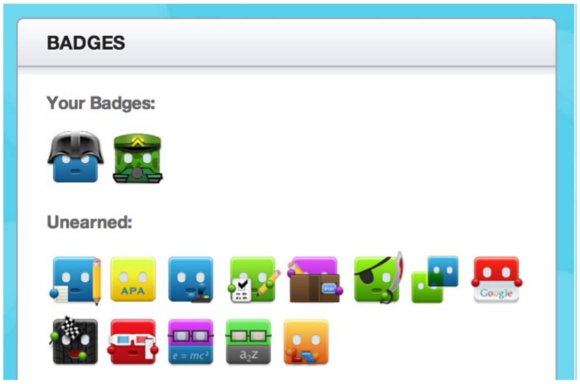
The GradeCraft platform includes two dashboard views, the Student Dashboard and Instructor Dashboard. The student dashboard is a single-page displaying of student's comprehensive course progress. Figure 3 shows an example of Student Dashboard.



Why are videogames fun? The answer isn't as obvious as you might think. Good games draw you in, teach you how to succeed, and keep you engaged with a "just right" level of challenge. Most importantly, players learn while playing a well-designed game. Why isn't school like that? This class takes a close look at videogames, a close look at education, and considers ways that each can be improved to maximize learning and performance.

Figure 3 GradeCraft Student Dashboard Overview

The top progress bar serves an informational purpose and also have a motivational effect that help boosts user motivation to complete tasks. The top portion includes visual chart of the point and badges they have earned paired with ones that are still available to work on. Figure 4 shows an example of Student Dashboard badges.



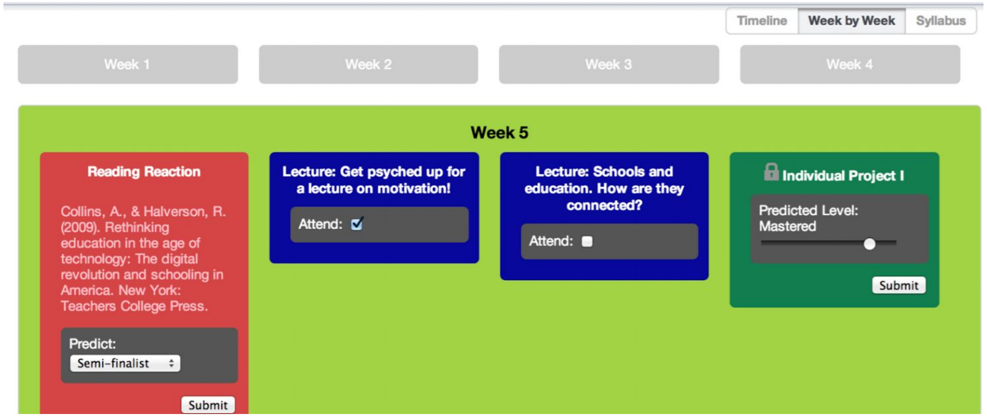
BADGES

Your Badges:

Unearned:

Figure 4 GradeCraft Student Dashboard Badges

It also shows a To Do list that highlights upcoming assignments, assignments that could be redone to show improvement or skill mastery, and, if possible, feedback on a recent successful assignment. It provides a convenient view for student to keep track of their upcoming assignments and course progress. Figure 5 shows an example of Student Dashboard To Do list.



Week 5

Reading Reaction

Collins, A., & Halverson, R. (2009). Rethinking education in the age of technology: The digital revolution and schooling in America. New York: Teachers College Press.

Predict: Semi-finalist

Submit

Lecture: Get psyched up for a lecture on motivation!

Attend: ☑

Lecture: Schools and education. How are they connected?

Attend: □

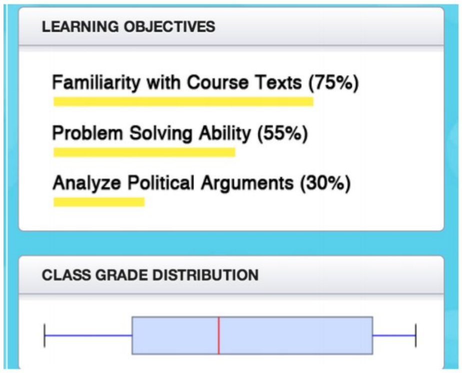
Individual Project I

Predicted Level: Mastered

Submit

Figure 5 GradeCraft Student Dashboard Badges TO DO List

Tabs shows their progress towards achieving the course learning objectives. Class grade distribution is charted to show the students their overall performance against the class. Figure 6 shows an example of Student Dashboard Learning Objectives Tab and Class Grade Distribution Tab.



LEARNING OBJECTIVES

Familiarity with Course Texts (75%)

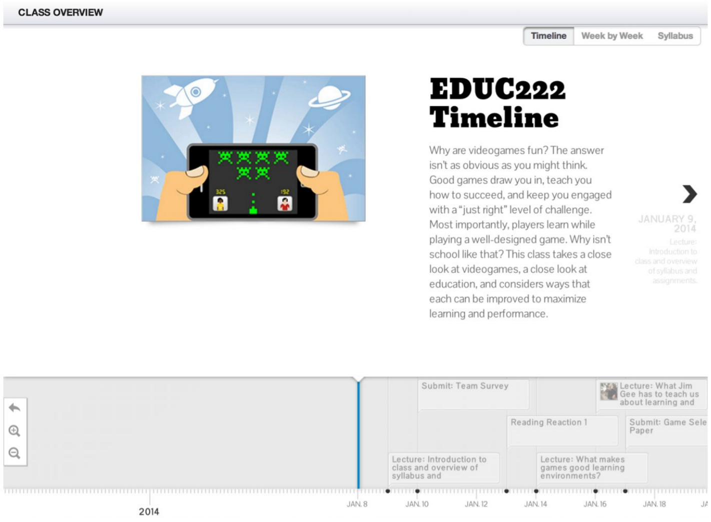
Problem Solving Ability (55%)

Analyze Political Arguments (30%)

CLASS GRADE DISTRIBUTION

Figure 6 GradeCraft Student Dashboard Objectives Tab and Class Grade Distribution Tab.

Beneath the tabs is a display of the semester plan that students can manipulate, selecting between a calendar view, a list view, a timeline, and a tech-tree display of the semester dates and assignments. These displays also operate as the portal through which students submit their work, identify self-selected groups, record participation, predict their scores, and receive feedback. Figure 7 shows an example of Student Dashboard Calendar.

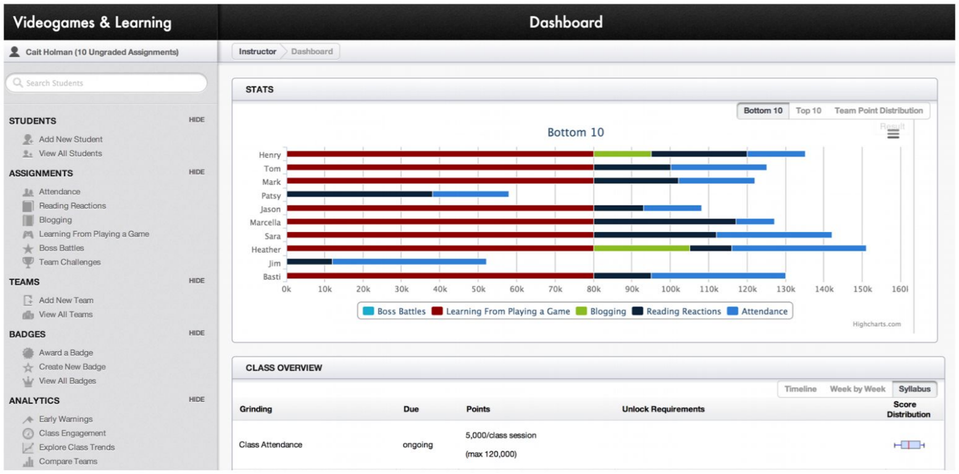

CLASS OVERVIEW
Timeline
Week by Week
Syllabus
325
192
JANUARY 9, 2014
Lecture: Introduction to class and overview of syllabus and assignments.
Submit: Team Survey
Lecture: What Jim Gee has to teach us about learning and
Reading Reaction 1
Submit: Game Sele Paper
Lecture: Introduction to class and overview of syllabus and
Lecture: What makes games good learning environments?
2014
JAN. 8
JAN. 10
JAN. 12
JAN. 14
JAN. 16
JAN. 18
JA

EDUC222 Timeline

Why are videogames fun? The answer isn't as obvious as you might think. Good games draw you in, teach you how to succeed, and keep you engaged with a "just right" level of challenge. Most importantly, players learn while playing a well-designed game. Why isn't school like that? This class takes a close look at videogames, a close look at education, and considers ways that each can be improved to maximize learning and performance.

Figure 7 GradeCraft Student Dashboard Calendar

The instructor dashboard shows teachers how their class is performing in a single view. The ten lowest and highest performing students' grades are each visualized with stacked bar charts, each color segment reflecting achievements within an assignment. Instructors can rapidly isolate which students may be in need of more help. A box and whisker plot is used to capture the overall class performance, displaying the range of achievement as well as situating how the majority of students are doing. Figure 8 shows an example of GradeCraft Instructor Dashboard.



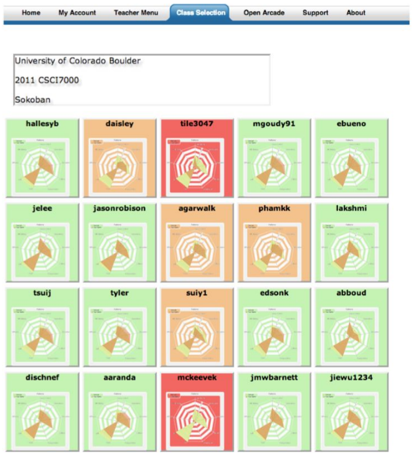
Grinding	Due	Points	Unlock Requirements	Score Distribution
Class Attendance	ongoing	5,000/class session (max 120,000)		

Figure 8 GradeCraft Instructor Dashboard

There is a list of techniques that can be immigrant from GradeCraft to the proposed project includes: using points and incremental levels instead of grades; awarding badges to recognize achievements and skill-acquisition; allowing students to redo assignments as many times as necessary to succeed; allowing students to determine how much assignments would count towards their final grade; selective view of top and bottom performancer in class; and displaying generalized information regarding classmates' performance.

2.3 REACT

REACT(Koh, K., Basawapatna, A., Nickerson H., & Repenning A. 2014) is a real time graphic assessment tool that quickly gives teachers insight into students' mastery of computational thinking construct as they creating games and simulations. It's an embedded assessment and web-based system that does not focus on end-user's programming tools providing real-time system alter for teacher on the student's mistakes. An example of this feedback is illustrated in Figure 9.



University of Colorado Boulder

2011 CSCI7000

Sokoban

Figure 9 REAL Assessment Feedback

Furthermore, it explicitly displays computational thinking pattern based assessment instead of lower-level programming construct. An example of this feedback is illustrated in Figure 10. It allows teacher to see which high-level computational thinking concepts student mastered and which ones they are struggling with on class and individual level, eventually allow them to make more effective instructional decisions.

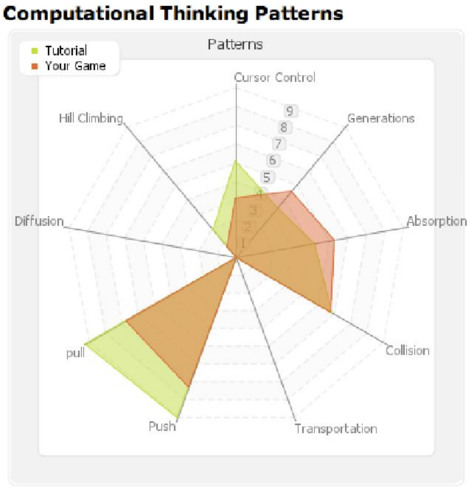
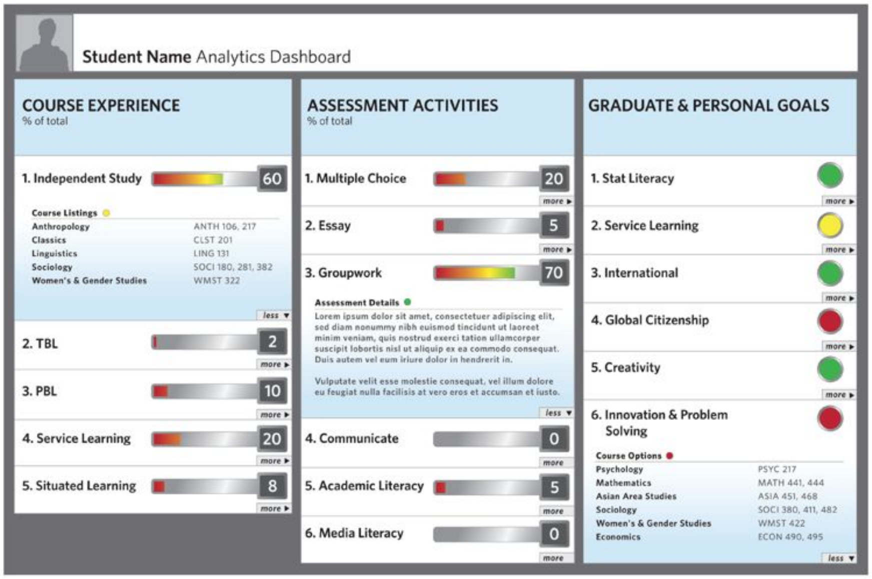


Figure 10 REAL Assessment Computational Thinking Pattern Feedback

In the proposed project, we can adopt a feedback system similar to the one developed by the REAL assessment team. For example, REAL generates personalized individual students feedback based on computational thinking pattern.The proposed project can also leverage its web-based natural and generate automated evaluation results in real time for teachers to view.

2.4 The Dashboard of Open Learning Analytics

The dashboard is implemented to assist individuals on making decisions about teaching and learning (Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Shum, S.B., Ferguson, R., Duval, E., Verbert, K., Baker, R.S.J.d. (2011)). It contains progression bars that indicates instructors' or students' self defined learning goals of the assignment. The decomposition of the assignment are lists for students control their own pace of learning. There are also categories for students to evaluate of the course and assignment and also serves as feedbacks for instructor to improve the course or assignment. Figure 11 provides a demo of the learner dashboard.



Student Name Analytics Dashboard

COURSE EXPERIENCE
% of total

1. Independent Study 60

Course Listings ●

Anthropology	ANTH 106, 217
Classics	CLST 201
Linguistics	LING 131
Sociology	SOCI 180, 281, 382
Women's & Gender Studies	WMST 322

2. TBL 2
3. PBL 10
4. Service Learning 20
5. Situated Learning 8

ASSESSMENT ACTIVITIES
% of total

1. Multiple Choice 20
2. Essay 5
3. Groupwork 70

Assessment Details ●

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in.

Vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto.

4. Communicate 0
5. Academic Literacy 5
6. Media Literacy 0

GRADUATE & PERSONAL GOALS

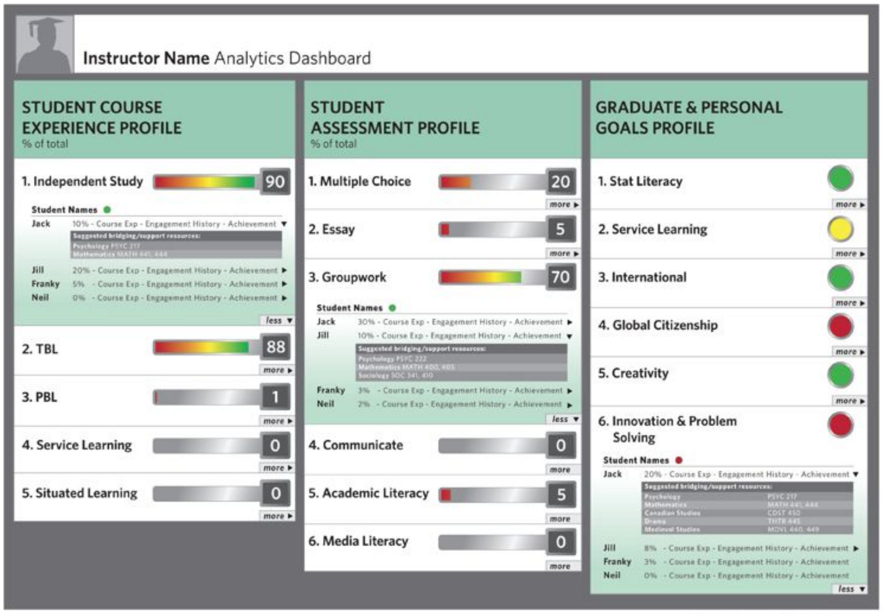
1. Stat Literacy ●
2. Service Learning ●
3. International ●
4. Global Citizenship ●
5. Creativity ●
6. Innovation & Problem Solving ●

Course Options ●

Psychology	PSYC 217
Mathematics	MATH 441, 444
Asian Area Studies	ASIA 451, 468
Sociology	SOCI 380, 411, 482
Women's & Gender Studies	WMST 422
Economics	ECON 490, 495

Figure 11 Learner DashBoard of Open LA

The instructor dashboard is similar to the learner dashboard that displays the course experience of each students and the goals that instructors aimed for each assignment. Based on the established researches, students' completion and satisfaction of a course can be impacted by students' attendances, connection of the rest learning committees, early assistance from instruction and etc. Figure 12 shows the view of instructor dashboard.



Instructor Name Analytics Dashboard

STUDENT COURSE EXPERIENCE PROFILE
% of total

1. Independent Study 90

Student Names ●
Jack 10% - Course Exp - Engagement History - Achievement ▼
Suggested bridging/support resources:
Psychology PSYC 217
Mathematics MATH 441, 444
Jill 20% - Course Exp - Engagement History - Achievement ▶
Franky 5% - Course Exp - Engagement History - Achievement ▶
Neil 0% - Course Exp - Engagement History - Achievement ▶

2. TBL 88
3. PBL 1
4. Service Learning 0
5. Situated Learning 0

STUDENT ASSESSMENT PROFILE
% of total

1. Multiple Choice 20
2. Essay 5
3. Groupwork 70

Student Names ●
Jack 30% - Course Exp - Engagement History - Achievement ▶
Jill 10% - Course Exp - Engagement History - Achievement ▼
Suggested bridging/support resources:
Psychology PSYC 222
Mathematics MATH 400, 405
Sociology SOC 341, 410
Franky 3% - Course Exp - Engagement History - Achievement ▶
Neil 2% - Course Exp - Engagement History - Achievement ▶

4. Communicate 0
5. Academic Literacy 5
6. Media Literacy 0

GRADUATE & PERSONAL GOALS PROFILE

1. Stat Literacy 🟢
2. Service Learning 🟡
3. International 🟢
4. Global Citizenship 🔴
5. Creativity 🟢
6. Innovation & Problem Solving 🔴

Student Names ●
Jack 20% - Course Exp - Engagement History - Achievement ▼

Suggested bridging/support resources:	
Psychology	PSYC 217
Mathematics	MATH 441, 444
Canadian Studies	CDST 450
Drama	THTR 445
Medieval Studies	MDVL 440, 449

Jill 8% - Course Exp - Engagement History - Achievement ▶
Franky 3% - Course Exp - Engagement History - Achievement
Neil 0% - Course Exp - Engagement History - Achievement

Figure 12 Instructor Dashboard of Open LA

With the dashboard, instructors' can identify which student need help at an early state. The students will get warning when attendance pattern changes or attendance drop substantially. These are some basic and important indicators that can also be adopted in SAGE's dashboard for instructors to provide assistance in time and for students to control their own learning pace.

3. Proposal

The proposed project can be break into three pieces, the backend mock data generation, students' view of dashboard and instructors' view of dashboard. The backend mock data will be in JSON format and served by the test server that written in Node.js. In student view, the dashboard will display the assignments' completeness and the number of drag/drop and delete mouse movement for the logged in student. It will also provide the cumulative time that this student has spent on this assignment and the number of students currently working on the same assignment. It will update with real-time information collected while students completing the assignment. In instructor view, the page will display the list of students and the detailed progression of each student. It will also provide aggregate data such as average completion time for each assignment, average length of students' working solution for each assignment to reflect the cleanness and simplicity of their solution.

3.1 Mock Data and Test Server

Since the focal point of the proposed project is to design student and instructor's view of the dashboard, which is part of the frontend of SAGE platform, we will first construct a couple of mock data set in JSON and build test server that serves the mock data set to dashboard pages. The structure of the project is shown in Figure 11.

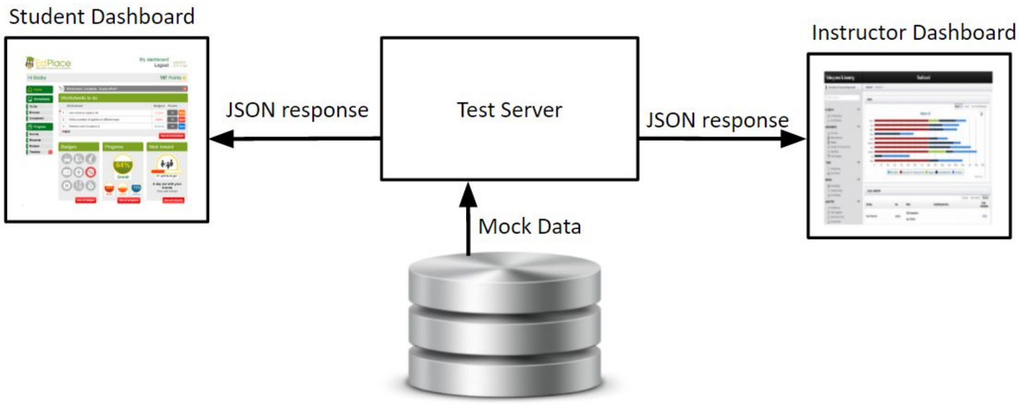


Figure 11 REAL Assessment Computational Thinking Pattern Feedback

We will only use mock data for now to display and test the dashboard that quantifies and visualizes the learning progression of students. The mock data will contain the potential information that can be collect through SAGE while students doing their assignment and the aggregate data that can be calculated from collected data. The test server will be written in Node.js since Ruicong and I are more familiar with and the choice of test server is not critical.

The Node.js will server the mock data through the html page with Angular.js. Angular.js will serve as the pipeline to connect all the component of Model-View-Controller and keep the implementation simple and clean. The mock data and test server are for development purpose to make testing simple and feasible without having a lot of modification of the actual main server.

3.2 Student Dashboard

There are a couple of mechanics we are proposing in order to encourage student to take ownership of their learning progress including displaying student's photo on their dashboard view; introducing a behavior based badges system that awards students behaviors like starting assignment early, adding comments and etc. The dashboard will also contain a progress bar for current assignment. We will separate current version of dashboard into two different sections under current assignment, implemented list and to do list. In the class overview section we will put student's current progress in context of the entire class by showing a histogram on how they are performance comparing with their fellow classmates.

3.3 Teacher Dashboard

Teacher dashboard, on the other hand, will have a more comprehensive view. All the class will be listed on the right. When clicked a specific class, all the assignment related to that class is presented in tabs on the right side of the page. Under each tab we will show a class progress bar at the top and a bar chart of top and bottom 10 performances for that assignment. Below the bar charts is a table view of each student with their id and aggregate data such as assignment starting time, completion time, and solution length that can be sorted in ascending or descending order. This design will help teachers identify struggling and overachieving students. All the students entry will also have their photo next to their id, so instructor can quickly identify their students. When instructors click on a particular student the server will redirect the browser to student's assignment page and show a list of error student made in time order, so instructors can more easily understand what students are struggling with and provide help accordingly.

4. Milestones

Table 1 outlines the all the milestone and their estimated completion dates. Milestones are ranked by their completion date.

Milestone	**Date**
Tech stack review	October 7
Backend	
Static mock dataset	October 14
Testing backend server	October 21
Student Dashboard	
Design	October 28
Implementation	November 11
Midterm written progress report	November 18
Instructor Dashboard	
Design	November 25
Implementation	December 9
Final written report and oral presentation	December 16

Table 1. Milestones and Deadlines

5. Future Work

More field research must be completed in order to evaluate and validate the effectiveness of the current design of the dashboard system on improving on encourage students to learn computational thinking and kept track of their learning progress, and also teachers' ability on tracking the progress of entire class and identifying struggling and overachieving students in class. Also further studies should also try to identify what additional sets of data can be collected that may be helpful for students and teacher in identifying problems and tracking progress, and which presentation of the dashboard data is most and least useful, so more comprehensive subsequent versions of the dashboard system can be implemented.

References

Holman, C., Fishman, B., Aguilar, S. (2013). *Designing a Game-Inspired Learning Management System*. Poster for Games+Learning+Society Conference 9.0. University of Wisconsin-Madison. Madison, WI.

J. M. Wing (2006). "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, p. 33, Mar. 2006.

Koh, K., Basawapatna, A., Nickerson H., & Repenning A.(2014). *Real Time Assessment of Computational Thinking.* 2014 IEEE Symposium on Visual Languages and Human-Centric Computing(VL/HHC)

Murphy, C., Kaiser, G., Loveland, K., & Hasan, S. (2009). Retina: Helping Students and Instructors Based on Observed Programming Activities. *SIGCSE* (pp. 178-182). Chattanooga: ACM.

Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Shum, S.B., Ferguson, R., Duval, E., Verbert, K., Baker, R.S.J.d. (2011) Open Learning Analytics: an integrated & modularized platform: Proposal to design, implement and evaluate an open platform to integrate heterogeneous learning analytics techniques. Athabasca, Alberta, Canada: Society for Learning Analytics Research.