THE NEXT GENERATION ELEARNING PLATFORM FOR COMPUTING EDUCATION

Final Report

APRIL 14, 2017
WONG CHING QUEN 3035098161
1. Abstract

Concerning the deep relationship between e-learning and computing education, this final report aims to provide a detailed information on our e-learning platform which fit the uses of students who involved in computing learning.

Although there are various choices of e-learning platform available on the market now, current platforms do not focus on the learning progress of students, especially support students in their learning. Therefore, this project would like to set up a new e-learning platform which made improvements in supporting tools on learning progress. With collaborating tools and data analytics, the platform can provide a CS specialized environment, and data analytics can reduce the time of students on breaking the bottleneck in computing learning. Students can receive automatic and relevant feedback so they can no longer rely on others when they face problem in computing learning.

C++ is used as the coding language of the code editor, while Laravel is used as the web framework. Implementation detail will be mentioned in Project Methodology part.

2. Acknowledgement

The platform would not be built without the help of many individuals. Our FYP supervisor, Dr. CK Chui, provided general direction and scope of the platform, and many suggestions on the implementing process, especially in the data analytics.

The platform relies on machine learning techniques for data analytics. We would like to thank Dr. Andrew Ng, Dr. Li Yi Wei and Professor N. Mamoulis, who are the instructor of the Coursera machine learning course, HKU machine learning courses and Advanced Database Systems respectively.
3. Table of Content

1. Abstract ..............................................................................................................................................1
2. Acknowledgement .................................................................................................................................1
3. Table of Content .....................................................................................................................................2
4. List of Figures .........................................................................................................................................3
5. List of Tables ..........................................................................................................................................3
6. Introduction ...........................................................................................................................................4
7. Project Background .................................................................................................................................6
  7.1 AI Teacher .........................................................................................................................................6
    7.1.1 Test cases classification .................................................................................................................6
    7.1.2 Ranking Test cases ..........................................................................................................................6
    7.1.3 Suggesting Test cases group ..........................................................................................................7
    7.1.4 Show code changes or Provide code change directions .................................................................7
  7.2 Web Interface ......................................................................................................................................7
8. Project Methodology ...............................................................................................................................15
  8.1 Overview ..........................................................................................................................................15
  8.2 Front End ..........................................................................................................................................16
    8.2.1 Web Framework .............................................................................................................................16
    8.2.2 User Interface ................................................................................................................................17
    8.2.3 Code Editor ..................................................................................................................................18
    8.2.4 Compiler .......................................................................................................................................19
    8.2.5 Discussion Forum .............................................................................................................................20
  8.3 Back End ..........................................................................................................................................20
    8.3.1 Database .......................................................................................................................................20
    8.3.2 Firebase ..........................................................................................................................................24
    8.3.3 Data analytics ................................................................................................................................24
9. Conclusion ...............................................................................................................................................32
10. Future Works .........................................................................................................................................32
11. Reference .............................................................................................................................................33
12. Appendix .............................................................................................................................................34
4. List of Figures

Figure 1 Dashboard interface of web system ........................................7
Figure 2 Course list interface of web system .......................................8
Figure 3 Forum interface of web system ............................................8
Figure 4 Course page of the web system .........................................9
Figure 5 Assignment page of the web system ..................................10
Figure 6 File uploader in assignment page ....................................10
Figure 7 Test case management page of web system ......................11
Figure 8 Student performance page of web system .........................11
Figure 9 Code editor of web system ...........................................12
Figure 10 Renaming interface in code editor .................................12
Figure 11 Test input interface in code editor ................................13
Figure 12 Console in code editor ..................................................13
Figure 13 Result of test cases shown in console ............................14
Figure 14 Hint on coding ............................................................14
Figure 15 Architectural diagram of the platform ............................16
Figure 16 General data analytics workflow ................................25
Figure 17 The elbow method ..........................................................31
Figure 18 Classification results of K-means, DBSCAN and Agglomerative
   clustering on sample data .....................................................36
Figure 19 Computation of silhouette coefficient ............................36

5. List of Tables

Table 1 Requirement of project ......................................................5
Table 2 Contribution of project ....................................................6
Table 3 Small plugins used in project ..........................................18
Table 4 Database schema of web system back-end .......................23
Table 5 Comparison between ML candidates .................................34
6. Introduction

In the past few years, the uses of e-learning platforms increase such that it is unavoidable for students nowadays to use e-learning platform within their learning progress. E-learning platforms provide a good medium for students to gather course materials, involve in course assignment, as well as catching up the latest news of the course. Assignment submissions, notice board, discussion forum are some of the corresponding components in nowadays e-learning platform.

The e-learning platforms provide the above functions to students, but it is not enough for computing education. Unlike traditional learning, computing education emphasize the importance of learning progress, such as the methodology to create an algorithm. Students can learn a programming language quickly, while this new language has a similar use or purpose when comparing to the languages that students have been learnt. Students can pick up the new programming language to solve problem without putting much effort to get familiar with the syntax as they own experiences.

The relationship between e-learning platform and programming is not strong currently. There are no supporting tools for students when they due with programming assignments. Students would use other code editor or even Integrated Development Environment (IDE) to handle the coding. The platform has no use in helping the students, but only as a storage for assignment documents and submission.

Therefore, our project aims to provide a platform to fill this position. By providing collaborative tools and coding support, students can stick to the e-learning platform in their learning progress, which are expected to learn programming in a better way and efficient way.
The objective of this project is to allow students to learn programming by themselves with minimal efforts from teachers using the coding support and real-time feedback provided by the platform. This project aims to provide a collaborative and CS-specialized environment for computing students to communicate and share their ideas using collaborative tools provided by the system. At last, this project aims to provide timely coding feedback to students by constructing an AI teacher.

The requirement for this project is as below:

<table>
<thead>
<tr>
<th>Medium</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server, virtual machine (with high computational power)</td>
<td>1. To compile the codes submitted by students</td>
</tr>
<tr>
<td></td>
<td>2. To carry out data analytics</td>
</tr>
<tr>
<td>PHP framework</td>
<td>To act as a base of the e-learning platform</td>
</tr>
<tr>
<td>Data analytics’ tools or libraries (Scikit-learn(^1), Numpy(^2))</td>
<td>To facilitate the process of machine learning</td>
</tr>
</tbody>
</table>

Table 1 Requirement of project

The contribution of this project is as below:

<table>
<thead>
<tr>
<th>Member</th>
<th>Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cai Tung San</td>
<td>Data analytics</td>
</tr>
<tr>
<td>Chan Wai Lun</td>
<td>Data analytics</td>
</tr>
<tr>
<td>Cheung Siu Wai</td>
<td>Collaborative tools of web systems</td>
</tr>
<tr>
<td></td>
<td>UI design</td>
</tr>
<tr>
<td></td>
<td>Documentation</td>
</tr>
<tr>
<td>Wong Ching Quen</td>
<td>Framework of web system</td>
</tr>
<tr>
<td></td>
<td>Code editor integration</td>
</tr>
</tbody>
</table>

---


7. Project Background

The platform owns a role of assisting students in learning coding. We will first provide an overview on the concept of AI teacher, and move on to the actual interface of the web system.

7.1 AI Teacher

To assist students on their coding as well as helping teachers to see students’ progress, an AI teacher is provided to provide coding support and feedbacks to students and calculating results for teachers. The AI teacher has four features provided and are listed below.

7.1.1 Test cases classification

The AI teacher can classify different test cases into different clusters, which corresponds to different programming concepts. When new test cases are input by teachers, the AI teacher tries to classify them to determine whether this is a test case to test new concepts in the code or just another test case which test the concepts like some other test cases already stored.

Having test cases testing new concepts allow teachers to test students’ codes much comprehensively and ensure students’ codes have less bugs.

7.1.2 Ranking Test cases

Test cases are divided into different clusters, the AI teacher can also rank the clusters in different difficulties. Scoring are done by the passing of the number of clusters instead of the number of test cases that are passed. Higher...
weighting can also be added to cluster with higher difficulties.

7.1.3 Suggesting Test cases group

Classifying test cases into different groups and ranking different groups allows the AI teacher to know which test cases groups the students have not passed and the AI teacher could also know the level the students are currently in. Therefore, the AI teacher can suggest test cases groups that are in the same level with students, so that students can deal with test cases that are easier for them to deal with in their current levels.

7.1.4 Show code changes or Provide code change directions

Students may still struggle on how they can pass the test cases groups; the AI teacher can also use other students’ code as a reference and show to the students about the code changes that leads to the passing of those test cases groups.

7.2 Web Interface

The project consists of a webpage of e-learning platform. The following will show all the interfaces of the web system:

Figure 1 Dashboard interface of web system

Figure 1 shows the dashboard page of web system. There is only a welcome
message in this stage.

Figure 2 Course list interface of web system

Figure 2 shows the course list which shows all the available course on the e-learning platform. By clicking the course name, students can go into the course page, which will be showed in the latter part.

Figure 3 Forum interface of web system

Figure 3 shows the discussion forum provided by the e-learning platform. Students can create and reply to posts to discuss with peers.
Figure 4 Course page of the web system

Figure 4 shows the course page of the course. As this e-learning platform mainly focus on programming assignments, the lecture notes part is eliminated. Tutors and professor can use the add button to add new assignments. The number in bubble indicates the number of files of that assignment. The trash icon represents the deletion of assignment. The add button and trash icon will be hided when the logged in user is a student. By clicking the assignment number, user will be redirected to the assignment page.
Figure 5 Assignment page of the web system

Figure 5 shows the assignment page. Users can view the assignment files in this page as well as download the file, by clicking the file name. Tutors and professors can use this page to manage the assignment files, by deleting the files or upload files in the uploader, as in figure 6 below:

Figure 6 File uploader in assignment page

Professors and tutors can manage the test cases of this assignment by clicking the Test Case button in management panel shown in figure 5. Users will be redirected to the test case management page. They can also click on the student performance button shown in figure 5 for viewing the current performance of students. When Coding button shown in figure 5 is clicked, user will be directed to
the code editor.

Figure 7 Test case management page of web system

Figure 7 shows the test case management page. Tutors and professors can modify the test cases, as well as the test case group title in this page.

Figure 8 Student performance page of web system

Figure 8 shows the student performance page. Tutors and professors can view
the student performance, included score and test cases details in this page.

Figure 9 Code editor of web system

Figure 9 shows the code editor of the assignment. User can type in the code with this code editor. Syntax highlighting as well as block collapse and expand are provided as general function of code editor.

Figure 10 Renaming interface in code editor

User can rename their source code by clicking the file name at the left bottom corner. User can also download the source code file by clicking the download button.
User can define the default input for testing the program by clicking the input button at the bottom.

When user click compile and run, the console will be pop out and output will be generated.
When the assignment is done, user can click the submit and check button. Then the output of the code will be tested against test cases which inputted by tutors or professors.

When the students encounter troubles, they may click on the hint button for getting suggestions. The hint function is based on the previous submission of the source, by showing two pieces of code, which corresponded to code before passing.
the test case and code which passed the test case. Our project team engage in this project since team members all study in computer science, which mean we experienced the similar difficulties and bottleneck like other students in computing learning. Therefore, we know the need of the students in computing learning. We hope this platform can support the students in the future in learning programming.

8. Project Methodology

8.1 Overview

The two major components of this project are collaborative tools and data analytics. Collaborative tools used to provide a CS-specialized environment for students. Data analytics generate auto feedback to students in the code editor. The structure of the platform is illustrated as below:
8.2 Front End

The front-end part is to set up a web system such that user can interact with the platform through user interfaces. For example, students can use the code editor to edit, compile and view the source code as well as the program output. Professors or tutors can view, edit, and delete the test cases of an assignment.

8.2.1 Web Framework

This project uses Laravel as the web framework. Laravel\(^3\) is the PHP framework chosen due to its popularity and extendibility. Due to community

---

efforts, solutions to problems encountered in the development stage are found easily in the community. The problems can be solved quickly and can secure the development process. Laravel community provide a huge amount of interactive and timely support for developers. Therefore, the difficulty and development time of the platform are reduced with the aids of Laravel community.

Furthermore, the extendibility of Laravel reduced the difficulty and development time of the platform. Laravel provides more than 9000 packages, which are the fast solutions for desired functions. One of the example is the forum plugin, which will be mentioned in the latter part.

8.2.2 User Interface

The web system heavily relies on Bootstrap to utilize the style of user interface. Bootstrap provides a clean and fast solution for components positioning and overall styling. This reduced the development time as our team member can focus on the functionality of the platform, instead of concerning about the style of buttons, menu bar or even forms.

Furthermore, the blade template is useful when handling data presentation. One of the example is the test cases page. Test cases page provide the list of test cases of an assignment while the test case group groups the test cases, at the same time provided forms for updating and deletion function of the test cases, as well as the test cases group information.

In addition, several small plugins are imported to the platform for facilitating better user experiences. The details are as below:

<table>
<thead>
<tr>
<th>Name</th>
<th>Author</th>
<th>Reference</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>bootstrap-notify.js</td>
<td>mouse0270</td>
<td><a href="http://bootstrap-notify.remabledesigns.com/">http://bootstrap-notify.remabledesigns.com/</a></td>
<td>Use as the notification pop</td>
</tr>
<tr>
<td>Small plugins used in project</td>
<td>Code Editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>jquery-pop-up-overlay.js</strong></td>
<td><strong>Firepad</strong> and <strong>Ace</strong> together. Firepad provides an interface to link with firebase, a cloud data management system. Therefore, all the input of the students can be recorded. These records then can be accessed and manipulated for data analytics. This</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>vast-engineering</strong></td>
<td><strong>official Firepad Website.</strong> (n.d.). Retrieved April 14, 2017, from <a href="https://firepad.io/">https://firepad.io/</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong><a href="http://dev.vast.com/jquery-pop-up-overlay/">http://dev.vast.com/jquery-pop-up-overlay/</a></strong></td>
<td><strong>Use as the pop up window in code editor</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>johnny</strong></td>
<td><strong>Use as the table formatting and styling with integrated sortable functions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong><a href="https://johnny.github.io/jquery-sortable/">https://johnny.github.io/jquery-sortable/</a></strong></td>
<td><strong>FileSaver.js</strong></td>
<td><strong>eligrey</strong></td>
<td><strong><a href="https://eligrey.com/blog/saving-generated-files-on-the-client-side/">https://eligrey.com/blog/saving-generated-files-on-the-client-side/</a></strong></td>
</tr>
</tbody>
</table>

8.2.3 Code Editor

The code editor is developed by integrating Firepad\(^4\) and Ace\(^5\) together.


native data logging system reduced the workload of design and deployment of data logging system. In addition, Firepad serves as a real-time collaborative code editor, which means several students can edit the same piece of code at the same time. Detailed information of Firepad will be mentioned in the back-end part.

Ace serves as a general code editor which provides useful functions, such as syntax highlighting, block collapse and expend, code completion, etc. With the aids of online tutorial and demo, the integration of the code editor components was carried out quickly.

8.2.4 Compiler

The compiler of code editor mainly focus on compiling C++ code. As the web server runs on top of a Linux machine, it is convenience to compile the source code by local gcc compiler in the Linux machine. When the user clicks compile button, a AJAX request will be sent to the web server to trigger compiling of source code. In corresponding Laravel controller class, the Process library from Symfony will be used to execute the gcc compiler like Linux terminal. Furthermore, as Linux bash command supports redirection of standard input and output, our group integrated this feature into our code editor. Students can view their program output base on the input.

Moreover, the function of testing the source code against test cases strongly rely on the above feature. When user clicks on the submit button, the web system will receive a request for submission. A new submission of current version of the source code will be created, and the source code will be compiled.

---

and test against all the test cases. All the results will be recorded and to be stored in database. In addition, results of the submission will be show in the console of the code editor. Students can then know whether the source code is sufficient for handling all situations.

8.2.5 Discussion Forum

The discussion forum of the platform is an integrated plugin imported from Riari/Laravel-forum\(^7\). This plugin provides the template on forum user interface as well as the structured data model used by the forum.

8.3 Back End

The back-end part of the platform is mainly to set up a database for storing all the necessary data. These necessary data included: course information, assignment details, assignment files, test cases of assignments, code submission of students and submission results. These data will be used in either data presentation to users through the web system or carry out data analytics in the backend. The data analytics process will be mentioned in the latter part.

8.3.1 Database

The database of the platform is essential to store all the information required. Plenty of changes in the database schema are involved in the development progress. Therefore, the migration system of Laravel together with MySQL provides a good solution for version control. Changes in database can be done by migration and each migration can be viewed as an atomic operation. This ensures the databases in local development environment and deployment are synchronized. The detailed database schema and explanation are as below:

<table>
<thead>
<tr>
<th>Database Schema</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>assignments (assignment_id, course_id, order, timestamp)</td>
<td>Table to store the information of assignments; assignment_id as primary key; course_id reference course (course_id); order as the indicating number</td>
</tr>
<tr>
<td>coding (coding_id, course_id, assignment_id, user_id, filename, default_input, timestamp)</td>
<td>Table to store the information of coding of students; coding_id as primary key; course_id, assignment_id reference assignments (course_id, assignment_id); user_id reference user(user_id); default_input as the input that pass to the output executable after compiling</td>
</tr>
<tr>
<td>courses (course_id, fullName, name, courseCode, timestamp)</td>
<td>Table to store the information of courses; course_id as primary key; fullName as the full name of course; name as the abbreviation of course name; courseCode as the course code</td>
</tr>
<tr>
<td>files (file_id, assignment_id, filename_original, filename_storage, size, timestamp)</td>
<td>Table to store the information of assignment files; file_id as primary key; assignment_id reference assignments (assignment_id); filename_original as the file name of file</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>submissions (submission_id, course_id, assignment_id, user_id, content, timestamp)</td>
<td>Table to store the information of submissions; submission_id as primary key; course_id, assignment_id reference assignments (course_id, assignment_id); user_id reference user(user_id); content as the code content</td>
</tr>
<tr>
<td>testcases (testcase_id, course_id, assignment_id, input, output, group_id)</td>
<td>Table to store the information of test cases; testcase_id as primary key; course_id, assignment_id reference assignments (course_id, assignment_id); input as the input of the testcase; output as the output of the testcase; group_id as the group id of test case group</td>
</tr>
<tr>
<td>testcase_group (id, course_id, assignment_id, avg_rate, title)</td>
<td>Table to store the information of test case groups; id as primary key; course_id, assignment_id reference assignments (course_id, assignment_id); avg_rate as the rate for calculating the score of students;</td>
</tr>
</tbody>
</table>

when user upload;
filename_storage as the file name of file in the storage disk in the server;
size as the file size
| test_by (testcase_id, submission_id, user_output, result) | Table to store the testing result when a submission is triggered, the source code of the submission is tested against to all test cases and stored in this table; testcase_id reference testcases (testcase_id); submission_id reference submissions (submission_id); user_output as the output of the executable of source code correspond to the test case; result as the Boolean value that represents the output is correct or not when compare to the output of test case |
| users (id, name, email, password, group) | Table to store the user information, generated by Laravel Middleware; id as primary key; name as the name of user; email as the email address of user; password as the password of user; group as the user group of users |

Table 4 Database schema of web system back-end

The above database schema constructs all the essential information of the platform, including test case management, submission management, testing
of source code against test cases, etc.

8.3.2 Firebase

Firebase\textsuperscript{8} serves as a native data logging system. In our project, firebase provides a quick and efficient solution for solving the storage of source code of students. Without firebase, the development of code editor may involve the implementation of file input and output method for every single saving operation of source code, as well as the synchronization problem when providing a real-time collaborative code editing environment. Now all the source code on firebase are separated by user id and assignment id, which means all the assignments are individual assignment now. Therefore, the real time collaborative feature is not used in the current version of the platform, but it provides a flexibility to extend assignments to group assignments easily.

8.3.3 Data analytics

The data analytics is carried out to provide the features of the AI teacher mentioned in Section 12. The passing of test cases by students are used for the data.

Machine learning is used to group the test cases into different categories. Figure 16 below illustrates the flow of data analytics from data being inputted to auto feedback being produced eventually.

8.3.3.1 Preprocessing and getting correlation matrices

The percentage of test cases grades the students result they pass for that assignments. A test case contains both input and output. When students submit a program code, input is used to test the code and try to generate output using the code. The student’s output is then compared with the test case output, if they are the same, then that test case is passed. Usually the students cannot pass all the test cases by the first time they hand in a submission, they will try to modify their code and then submit again. Therefore, one students may have many submissions regarding just one assignment.

These submissions leading to the changes in passing of test cases are used to calculate correlation matrices and for data analysis.

These data will be split into training data, validation data and test data in about 50-25-25 proportion. The training data is responsible for setting parameters in the classification algorithm. The validation data is responsible for fine-tuning to prevent underfit or overfit of models as well as comparing performance between algorithm candidates. The test data is responsible for...
verification and evaluating the accuracy of the algorithm.

8.3.3.1 Submission Vectors and Delta Vectors

The submissions of students are modelled as submission vectors. For example, if student passed test cases #3, #5, #6 out of the 10 provided test cases, the submission vector will be [0 0 1 0 1 1 0 0 0 0]. The submission of students will then be combined as submission matrix by concatenating submission vectors. The difference between submission vectors, delta-vectors, are computed using XOR operator. To illustrate, for submission #1 [0 1 1 0] and submission #2 [1 1 0 0], the delta vector will be [1 0 1 0].

8.3.3.1.2 Correlation tuples

The delta vectors calculated are used to generate correlation tuples. Correlation tuples are generated between “1” and “1” entry in each delta vectors. For examples, for delta vector [1 1 1 0 0], test cases #1, #2, #3 are the 1s in the delta vector, correlation tuples are the combination between these 3: (#1, #2), (#1, #3), (#2, #3).

Correlation tuples are used to update the correlation matrices by a function of 0.8x-2, where x is the number of “1” entries in the delta vectors, in the case [1 1 1 0 0], x is 3, value will be 0.83-2=0.8, the value is used to update the correlation matrices by the correlation tuples. (#1, #2) represents update row 1 and column 2 in the correlation matrix.

8.3.3.1.3 Penalty function

To model the degree of correlation between test cases more accurately, penalty function is introduced. The penalty function used is 0.95x-2, where x is the number of non-0 entries in the delta vector.
Consider a simple example consists of only 2 submissions, submission #1 [0 1 1 0 0] and submission #2 [1 1 0 0 1]. The delta vector obtained according to above section is [1 0 1 0 1]. In the delta vector, there are three non-zero entries. The output of the penalty function would be 0.953-2=0.95, which is named penalty coefficient. All the entries of the delta vector are multiplied with the penalty coefficient. Following the example, the final delta vector would be [0.95 0 0.95 0 0.95]. The penalty function is a measure for regulating correlation. By imposing higher penalty, the correlation between test cases are reduced.

8.3.3.1.4 Correlation Matrices

After obtaining delta vectors from submissions, correlation matrix can be calculated. The entries in the matrix range from 0 to 1, implying low degree of correlation and high degree of correlation correspondingly. Each row in the correlation matrix is initialized to identity matrix with dimension equal to number of test cases. The choice of identity matrix assumes that each test cases are correlated with themselves and different test cases are not related at first. The test cases will have correlation only after being analyzed by the system. The update of the correlation matrix will be based on the clustering result.

Data first passing in the process will undergo preprocessing before being analyses to provide much relevant data for analysis. For example, code history must undergo DIFF function to get code changes. The function works like the “diff” command in Bash⁹, which sends back the

---

difference between two files. Other data inputted will also undergo normalization or standardization for better comparison and visualization of output.

8.3.3.2 Learning

After receiving data for input, learning algorithms should be decided for classification. Currently, we consider clustering algorithms only since they are unsupervised learning method, which suits our unlabeled data. There are multiple possible algorithm candidates, they include K-means, DBSCAN, Agglomerative clustering, which will be explained below.

8.3.3.2.1 K-means

Among the clustering algorithms, K-means has a relatively efficient time complexity, O(tkn), where t is number of iterations; k is number of clusters; and n is number of data points. However, K-means algorithm requires the parameter k(number of clusters) to be specified in advance. Extra experiments are required to be conducted to find optimal set of parameters. The algorithm also has the disadvantage of being unable to handle outliers. Dataset with outliers can have classification result with low accuracy since the algorithm attempts to put outliers in clusters.

8.3.3.2.2 DBSCAN

DBSCAN stands for Density-based spatial clustering of applications with noise. DBSCAN is associated with two parameters, and minPts. For an unvisited point p, if there are at least minPts points with distance not greater than, p is concluded as core point. In other words, p will become a core point if there are minPts points inside the circle area with radius and centered at p. After all points have been processed and all
core points have been found, the core points and the points within their radii will form groups. If two groups’ core points are within the radius of their counterparts, the two groups will be merged. Hence, Clusters are formed and outliers are separated since outliers will not be able to form cluster. The advantage of DBSCAN is the number of clusters is not required as parameters and robust to outliers. It can also form arbitrary shape of cluster, which may be favored depends on situation. However, it required large amount of experiments to find out suitable values for the two parameters for good clustering result.

8.3.3.2.3 Agglomerative clustering

Agglomerative clustering assigns each data point to its own cluster first. By using an evaluation metrics like single link, clusters are merged together. The process continues until termination condition is reached. The evaluation metrics determines whether clusters should be merged together. Take single link as example, if there is two data points in two clusters which has distance smaller than a certain threshold, the two clusters are merged. Eventually all clusters will be merged to a single cluster which contains all data points. For desired clustering result, termination condition is introduced. Common termination condition includes number of clusters in current result smaller than a threshold. A variant to this clustering algorithm is divisive clustering, which reverse the process. Instead of merging clusters, A single big cluster is divided until termination condition is reached. This clustering algorithm can determine number of clusters with the help of termination condition. However, the algorithm requires $O(n^2)$ time and hence does not scales up well.
These three algorithms both have their merits and demerits. Table 5 in appendix summarizes the features of the ML candidates and compares them.

The project team will try to implement them and the algorithm with higher prediction accuracy will be the major selection criterion. To understand the difference between classification result of different classifiers, sample data set was used to plot graphs and the characteristics of the resulting graphs were studied. The sample data used is Iris flower data set, which is included in the Scikit-learn library\(^\text{10}\). Figure 18 in appendix shows the difference between results of K-means, DBSCAN, Agglomerative clustering.

### 8.3.3.3 Evaluation

The number of clusters is a parameter required in several clustering algorithms. It plays a prominent role in affecting the clustering result. To obtain accurate clustering result, several methods are used to determine the number of clusters.

#### 8.3.3.3.1 Elbow method

The elbow method plot a graph of error against number of clusters. The optimal number of clusters will be the elbow point.

Figure 17 The elbow method

From figure 17, we can observe that the elbow point is $n=4$. This implies the optimal number of clusters is 4. This can be verified by computing second derivative for every point and find the maximum absolute value.

8.3.3.3.2 Silhouette analysis

The silhouette analysis mainly measure 2 quantities, coherence in between cluster and separation between clusters. High coherence and high separation implies good clustering. After computing silhouette coefficient for every data point as Figure 19 in appendix, average silhouette value in clusters can be found and plotted as silhouette plot. Similar silhouette distribution is a sign for good clustering result.
8.3.3.4 Prediction

The final model after evaluated could then be used for prediction. New data including test cases’ rates comes from students could be inputted to the model. The analysis will attempt to classify the test cases and offering feedback back to the code editor. These data can also be used for further tuning of the model and algorithms in the evaluation part.

9. Conclusion

To create a better environment for computing education, we presented the new platform which may greatly facilitate the learning experience of coding. Coding support and feedbacks help students to learn more efficiently without the need of tutors or lecturers.

This report described the architectural diagram of our new e-learning platform in the final stage.

This report also describes the whole picture of the platform with detailed implementation details, included all the libraries and algorithms integrated in the platform.

10. Future Works

One of our future work is to implement the full features of collaborative tools. Example is the real-time collaborative feature, which allows students writing scripts of a same source code at the same time. Group project can be used as a type of assignment and the real-time collaborative feature of code editor can be used. Furthermore, video presentation and drawing board are not developed or integrated due to time limit.
Another future work is to provide more accurate recommendation to students. This involves investigation of find code differences, which two piece of code are retrieved, which are the code before passing a test case and the code passing the test case respectively.

11. Reference

### 12. Appendix

<table>
<thead>
<tr>
<th>Advantage(s)</th>
<th>K-means</th>
<th>DBSCAN</th>
<th>Agglomerative clustering</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Relatively efficient</td>
<td>Can determine number of clusters</td>
<td>Manual tagging is not required relatively efficient</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Robust to outliers</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Disadvantage(s)</th>
<th>K-means</th>
<th>DBSCAN</th>
<th>Agglomerative clustering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensitive to outliers</td>
<td>Require manual tuning of parameters</td>
<td>Do not scale up well due to high time complexity</td>
<td></td>
</tr>
<tr>
<td>Need to specify number of clusters manually</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5 Comparison between ML candidates
Coherence

\( a(i) \): the average distance (dis-similarity) to all other samples within the same cluster:

\[
a(i) = \frac{1}{|C|} \sum_{x^{(i)} \in C} d(x^{(i)}, C)
\]

Separation

\( b(i) \): the average distance to the nearest cluster that \( x^{(i)} \) does not belong to:

\[
b(i) = \min_{C \subset \{C_1, \ldots, C_k\}; x^{(i)} \notin C} d(x^{(i)}, C)
\]

Silhouette

\( s(i) \): the silhouette value of \( x^{(i)} \):

\[
s(i) = \frac{b(i) - a(i)}{ \max(b(i), a(i)) }
\]

Which can be spelled out as:

\[
s(i) = \begin{cases} 
1 - \frac{a(i)}{b(i)}, & a(i) < b(i) \\
0, & a(i) = b(i) \\
\frac{b(i)}{a(i)} - 1, & a(i) > b(i)
\end{cases}
\]

So \(-1 \leq s(i) \leq 1\)

- \( s(i) \) is close to 1 if we have \( a(i) \ll b(i) \) → good clustering
- \( s(i) \) is close to \(-1\) → bad clustering

Figure 18 Classification results of K-means, DBSCAN and Agglomerative clustering on sample data

Figure 19 Computation of silhouette coefficient