Measuring the Power of Learning.®

Research Report

ETS RR–17-49

EPCAL: ETS Platform for Collaborative Assessment and Learning

Jiangang Hao

Lei Liu

Alina A. von Davier

Nathan Lederer

Diego Zapata-Rivera

Peter Jakl

Michael Bakkenson

December 2017

Since its 1947 founding, ETS has conducted and disseminated scientific research to support its products and services, and to advance the measurement and education fields. In keeping with these goals, ETS is committed to making its research freely available to the professional community and to the general public. Published accounts of ETS research, including papers in the ETS Research Report series, undergo a formal peer-review process by ETS staff to ensure that they meet established scientific and professional standards. All such ETS-conducted peer reviews are in addition to any reviews that outside organizations may provide as part of their own publication processes. Peer review notwithstanding, the positions expressed in the ETS Research Report series and other published accounts of ETS research are those of the authors and not necessarily those of the Officers and Trustees of Educational Testing Service.

The Daniel Eignor Editorship is named in honor of Dr. Daniel R. Eignor, who from 2001 until 2011 served the Research and Development division as Editor for the ETS Research Report series. The Eignor Editorship has been created to recognize the pivotal leadership role that Dr. Eignor played in the research publication process at ETS.

ETS Research Report Series ISSN 2330-8516

RESEARCH REPORT

EPCAL: ETS Platform for Collaborative Assessment and Learning

Jiangang Hao,[1] Lei Liu,[1] Alina A. von Davier,[2] Nathan Lederer,[1] Diego Zapata-Rivera,[1] Peter Jakl,[3] & Michael Bakkenson[3]

1 Educational Testing Service, Princeton, NJ
2 ACT Inc., Iowa City, IA
3 Pragmatic Solution Inc., Westlake Village, CA

Most existing software tools for online collaboration are designed to support the collaboration itself instead of the study of collaboration with a systematic team and task management system. In this report, we identify six important features for a platform to facilitate the study of online collaboration. We then introduce the Educational Testing Service Platform for Collaborative Assessment and Learning (EPCAL), which was developed to encompass these features to enable both large-scale online collaboration itself and the research on these collaborations. The collaboration management features of the EPCAL also make it an ideal tool for teachers to carry out computer-supported collaborative learning and assessment activities in classroom settings.

Keywords Online platform for collaboration; collaborative learning; assessment of collaboration; intelligent facilitation

doi:10.1002/ets2.12181

Collaboration is widely considered a key 21st-century skill that is important for both career and academic success (Griffin, McGaw, & Care, 2012; Organisation for Economic Co-operation and Development, 2017; von Davier & Halpin, 2013). Most existing studies on collaboration have focused on revealing important patterns or outcomes of collaboration based on relatively small samples of participants and nonstandardized items or tasks (Cohen, Lotan, Scarloss, & Arellano, 1999; DeChurch & Mesmer-Magnus, 2010; O'Neil, 2014; von Davier & Halpin, 2013). As a result, there are often questions about possible bias and the reproducibility of the findings due to the convenience sample (Hao, Liu, von Davier, & Kyllonen, 2017). To scale up the studies to larger groups of participants, standardized instruments (e.g., standardized items or tasks) should also be used. A prerequisite is to have a certain technological infrastructure to allow many participants to work together and get their responses as well as turn-by-turn interactions being properly recorded. It is not difficult to realize that the most feasible way to do this is through Internet-based collaboration.

Online collaboration is commonplace in today's world. Numerous tools have been developed to enable communication and collaboration via the Internet. These tools can be roughly classified into three main categories based on their primary functionality: instant messaging tools, such as Skype and Google Hangouts; web-conferencing tools, such as WebEx and ZOOM; and web forum tools, such as Wikipedia. In practice, many online systems integrate parts or all of the aforementioned functionalities. For example, Facebook has both instant messaging and web forum features. However, if one looks for a solution for carrying out research on collaboration at large scale, surprisingly, none of these existing tools satisfactorily meets the needs. This is mainly because these tools are primarily designed to enable actual collaboration rather than studies of collaboration. This realization motivated us to design and develop the ETS Platform for Collaborative Assessment and Learning (EPCAL), which comes with integrated team and task management systems together with collaboration facilitation functionality. These features make it an ideal platform for studying Internet-based collaboration on a large scale.

This report is organized as follows. We first introduce important features that an online platform should have to support research on collaboration and why most existing tools do not meet the needs. The findings are primarily based on our research on computer-supported collaborative learning and assessment (Andrews et al., 2017; Flor, Yoon, Hao, Liu, & von Davier, 2016; Halpin, von Davier, Hao, & Liu, 2017; Hao et al., 2017; Hao, Liu, von Davier, & Kyllonen, 2015; Hao, Liu, von Davier, Kyllonen, & Kitchen, 2016; Liu, Hao, von Davier, Kyllonen, & Zapata-Rivera, 2015; von Davier & Halpin, 2013).

Corresponding author: J. Hao, E-mail: jhao@ets.org

Then, we introduce the design and development of the EPCAL and show how it encompasses these important features to facilitate research on large-scale Internet-based collaboration.

Important Features of a Platform for Studying Collaboration

Let us start with a realistic but hypothetical scenario to illustrate the major features needed for a platform to study collaboration. Suppose you are a researcher and try to investigate the collaboration of students on a large scale. An immediate challenge is to decide "how" to make the collaboration happen on a large scale and capture all the interactions for further analyses. You will quickly realize that the most logistically feasible approach is to have the collaboration take place through the Internet instead of face-to-face, as the latter method has numerous practical challenges, such as activity arrangement, scheduling, and data capture. After deciding on Internet-based collaboration, the next thing you may realize is that you need to have a certain software infrastructure to deploy the collaborative tasks, manage participants into teams, allow participants to communicate via different means (e.g., video, audio, and text chat), and capture all the response and communication data. These are the very basic requirements for administering almost any collaborative activities to a large number of participants. Delving into the process further, you may realize that for the collaboration to go smoothly, some flow-control functionalities are needed. For example, you would need all participants' progress in the task to be synchronized at certain points to ensure that they are collaborating on the same thing. Further thinking about a collaborative activity, you may also want to have some (automated) external interventions at different stages during the collaboration process based on certain criteria. For example, suppose that, in a collaborative learning task, participants do not talk to each other; an intervention to remind them to collaborate would be helpful to avoid a completely failed collaboration. Finally, you may realize that you need to be able to save all the communication and response data in a reasonably structured way so that you can easily access and analyze them to understand the collaboration outcome and process.

The preceding hypothetical scenario may have already given you a good sense of what features a platform should have to support research on collaboration. Though we mentioned that the scenario is hypothetical, it is actually a concise summary of our journey when we carried out a research project on collaboration, the ETS Collaborative Science Assessment Prototype (ECSAP). In the ECSAP project, we developed a simulation-based task in the domain of science and recruited 500 dyads formed from 1,000 participants randomly selected to complete the task collaboratively. The details of the ECSAP are beyond the scope of the current report; we refer readers to the relevant papers for more detail (Andrews et al., 2017; Flor et al., 2016; Halpin et al., 2017; Hao et al., 2015; Hao, Liu, et al., 2016; Hao et al., 2017; Liu et al., 2015; von Davier & Halpin, 2013). All the pains we experienced during the development of the ECSAP project led us to think of a platform with important features that would greatly ease the implementation of research on collaboration at scale. We summarize these important features as follows and use F1–F6 (e.g., Feature 1 to Feature 6) to index them:

- F1: real-time multimodal communications, for example, via video, audio, and text chat.
- F2: management of participants and team formation.
- F3: mechanism to control the progress of the tasks and synchronize team members' activities.
- F4: management of items or tasks used in the collaborative activities.
- F5: proper capture of the response and communication data for further analysis.
- F6: mechanism for automated intervention/feedback based on the communications.

Almost all the instant messaging tools, web conferencing tools, and web forum tools may have a subset of the preceding six features, but not all of them in convenient ways, as they are primarily developed for enabling collaboration rather than for studying collaboration. For example, instant messaging tools, such as Skype, generally do not allow an easy way to obtain communication data from many teams, nor do they allow an open channel for external intervention or feedback based on the communication. On the other hand, in the community of computer-supported collaborative learning, a number of online communication or collaboration tools have been developed for different research projects. In Table 1, we provide a comparison between EPCAL and several widely used tools in computer-supported collaborative research.

Since 2013, ETS has systematically explored the possibilities for assessing collaboration at large scale. A dedicated conference and its proceedings were produced to synergize the efforts in the community (von Davier, Zhu, & Patrick, 2017). As an important infrastructure investment along this line, the EPCAL was developed to encompass the six important features for studying collaboration. In the rest of this report, we introduce the details of the design and development of

Table 1 Comparison of Different Software Tools Used in Computer-Supported Collaborative Learning

Application name	F1	F2	F3	F4	F5	F6
EPCAL	Y	Y	Y	Y	Y	Y
Skype (http://www.skype.com/)	Y	N	N	N	N	N
Google Hangouts (https://hangouts.google.com/)	Y	N	N	N	N	N
Facebook (http://www.facebook.com/)	M	N	N	N	N	N
Wikipedia (http://www.wikipedia.com/)	N	M	N	M	M	N
WebEx (http://www.webex.com/)	Y	M	N	M	M	N
ZOOM (https://zoom.us/)	Y	M	N	M	M	N
Google Docs (https://docs.google.com/)	Y	M	M	Y	M	N
Yammer (https://www.yammer.com/)	M	N	N	N	N	N
Etherpad (https://etherpad.org/)	M	M	N	Y	Y	M
MOODLE (https://moodle.org/)	M	Y	N	Y	N	N
Padlet (https://padlet.com/)	M	M	N	N	N	N

Notes. F1 – F6 denote the six features. M = marginally yes. N = no. Y = yes.

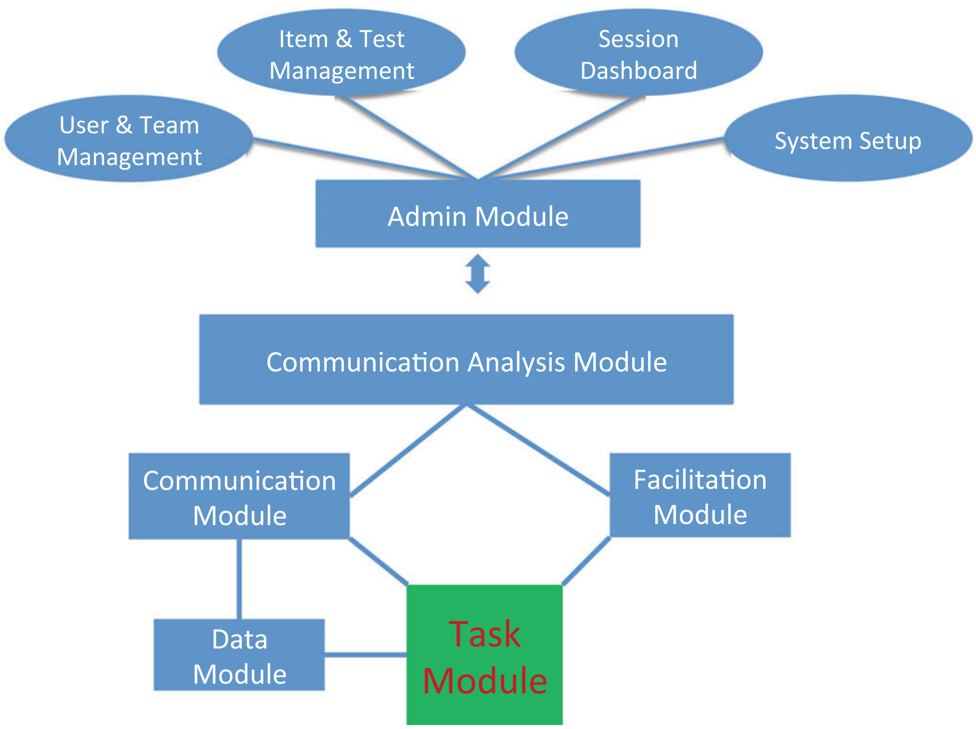

Figure 1 Design layout of the EPCAL.

EPCAL. It is worth noting that in addition to these features, EPCAL also comes with integrated analytics support as part of ETS's process-data analytics solution (Hao, Smith, Mislevy, von Davier, & Bauer, 2016).

The Design of EPCAL

Overview

The EPCAL consists of six major modules, the admin module, task module, communication module, facilitation module, communication analysis module, and data and analytics module, which are loosely mapped to the six features we listed in the previous section. Figure 1 is a schematic of the modules and their relations. The modules interact with each other via a set of application programming interfaces (APIs). A screenshot of the system as seen by users is shown in Figure 2. In the following subsections, we introduce the details of each module.

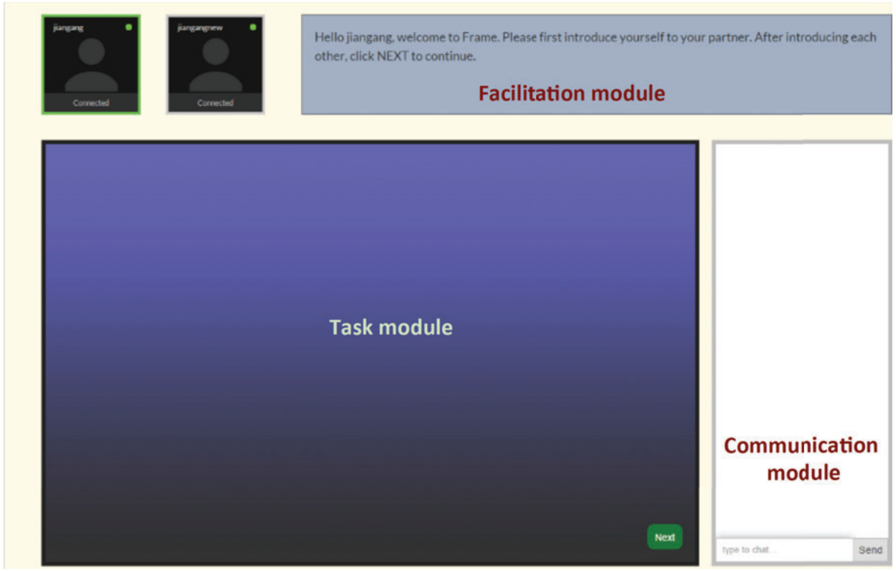

Figure 2 User interface of the EPCAL system.

Admin Module

The admin module is the overall control panel of the system. It has four components, as shown in Figure 1:

- user and team management
- item and test management
- session dashboard
- system setup

The user and team management component manages the user authentication and team formation. Each user is assigned a user ID, which will be authenticated when logging in to the system. Each member of the same team shares the same team ID. We currently allow two types of team formation. The first is random team formation, which means the members are grouped into teams of predefined sizes randomly based on the time they log in to the system. The second is predefined team formation, which means that members are grouped together based on some predefined rules. The predefined rules are implemented by assigning the same team ID to team members. On the EPCAL admin web interface, this component is covered under the "Users" tab, as shown in the screenshot in Figure 3. To implement a team formation rule, the administrator simply uploads a comma-separated values (csv) file with two columns: The first column specifies the user IDs, and the second specifies the team IDs.

The item and test management component allows the administrator to choose the item sets to form a test to administer. Though we call this a *test*, it is essentially a means of collecting items for people to collaborate on, so it can also be a set of learning activities. This is implemented under the "Problem sets" and "Tests" tabs of the admin web interface; see Figures 4 and 5 for the screenshots. Under the "Problem sets" tab, a set of XML-based templates are set forth; by following these, an administrator can create different item sets (problem sets). The currently supported item formats are multiple choice, constructed response, and numerical response. Some programming work is needed to integrate more complex items, such as simulations or games, into the EPCAL system. Under the "Tests" tab, the administrator can create different tests and assign different item sets to each. The administrator can then choose an active test to administer to the participants.

The session dashboard component allows the administrator to obtain a real-time overview of each team's progress. It also allows the administrator to view and download the corresponding log files for each session or a subset of sessions. It is implemented under the "Sessions" tab of the admin web interface. A screenshot is shown in Figure 6.

The system setup component is the core part of the admin module. It allows the administrator to configure several parameters to define the features of a particular session. Some of the key parameters include the number of members on each team, the modality of the communication (e.g., text mediated or video/audio mediated), whether team formation

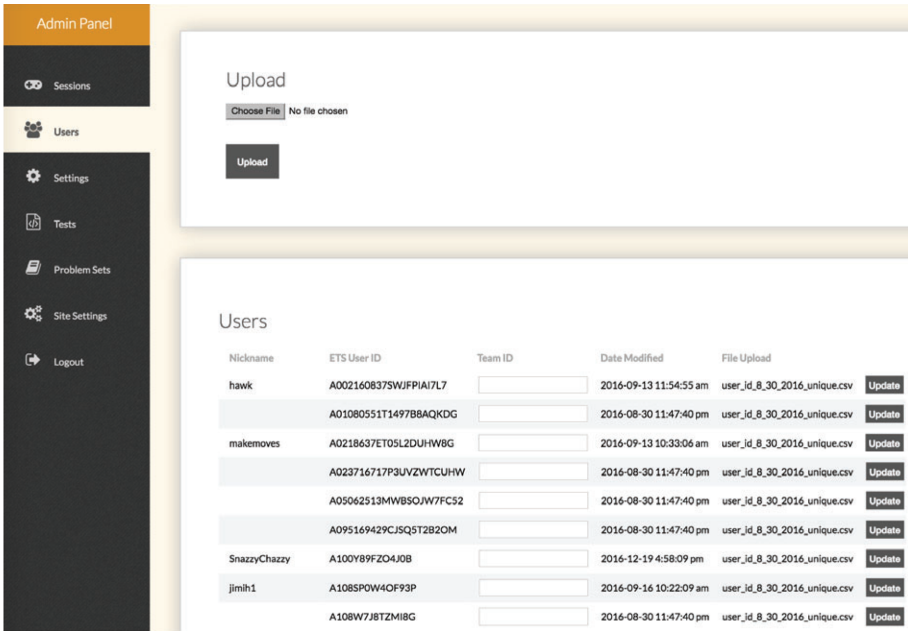

Figure 3 Screenshot of the user and team management component under the "User" tab in the admin web interface.

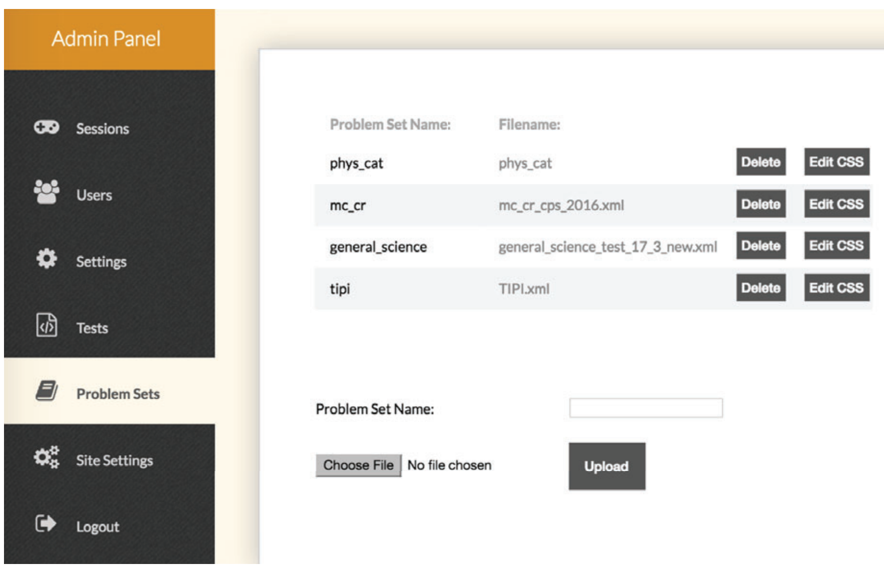

Figure 4 Screenshot of the item and test management component under the "Problem sets" tab in the admin web interface.

is random or predefined, and whether intelligent facilitation is enabled or disabled (see later sections for details). This component is implemented under the "Settings" tab of the admin web interface. Figure 7 shows a screenshot of the tab.

Task Module

The task module is essentially a placeholder for displaying the items or tasks to be used for collaborative activities. Figure 8 shows some simple examples: A multiple-choice item and a constructed-response item are displayed in the task module. Figure 9 shows a more complex item that involves a shared whiteboard for team members to draw or design collaboratively. The task module interacts with the communication module, facilitation module, and data and analytics module via a set of APIs that control the progress of the task based on the communication and facilitation. The APIs also capture the data from the task and save them to the data and analytics module.

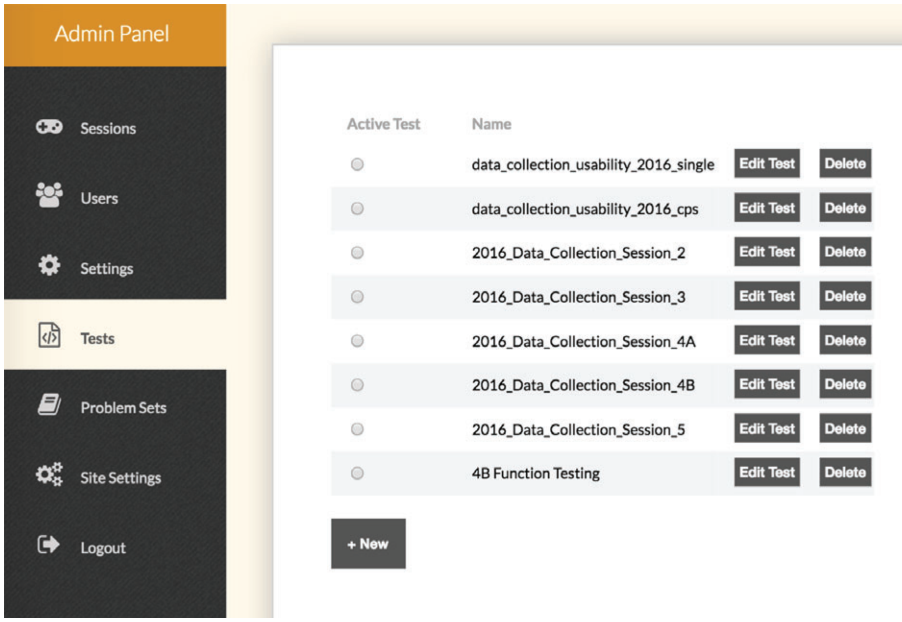

Figure 5 Screenshot of the item and test management component under the "Tests" tab in the admin web interface.

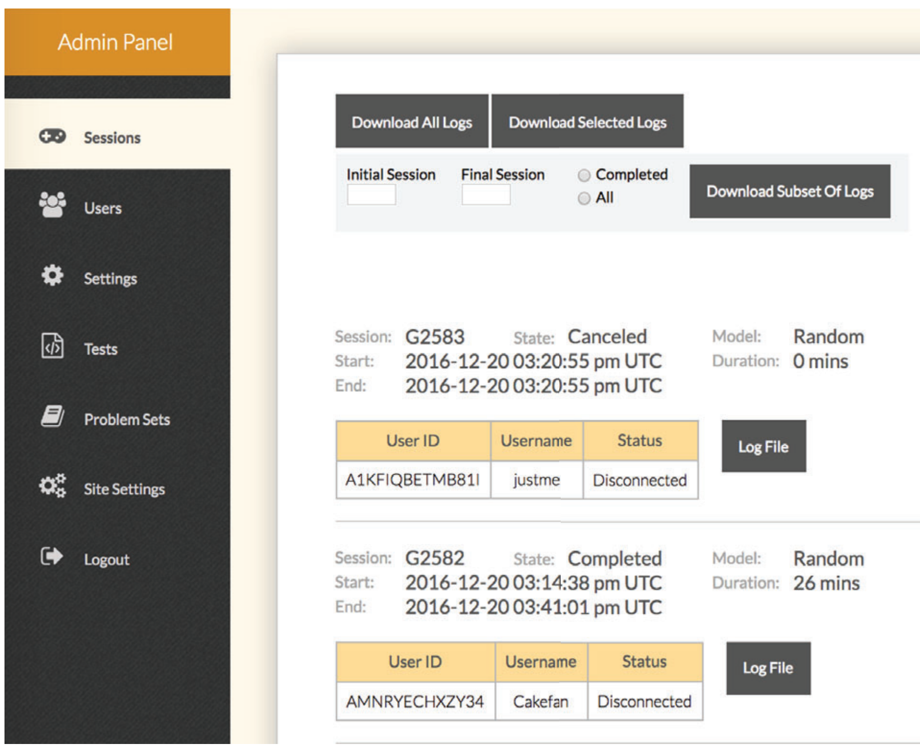

Figure 6 Screenshot of the dashboard panel in the admin module.

Communication Module

The communication module mediates the communications during the collaboration. It supports text chat and audio and video communication separately or in different combinations. The communications will be sent to the communication analysis module in real time (currently this is only implemented for text chat), which will receive and display any interventions or feedback.

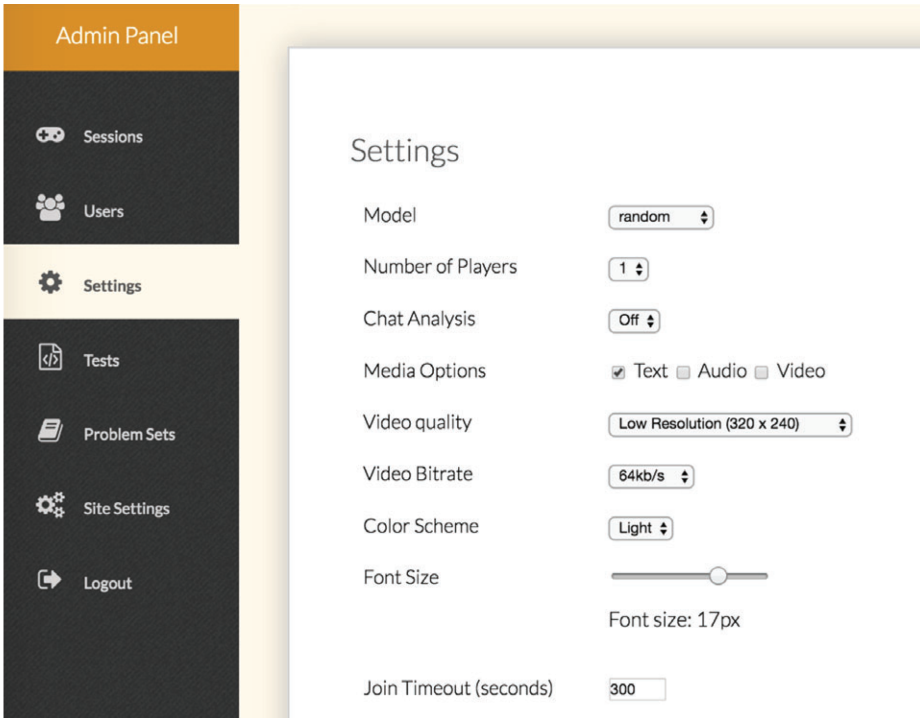

Figure 7 Screenshot of the system setup panel in the admin module.

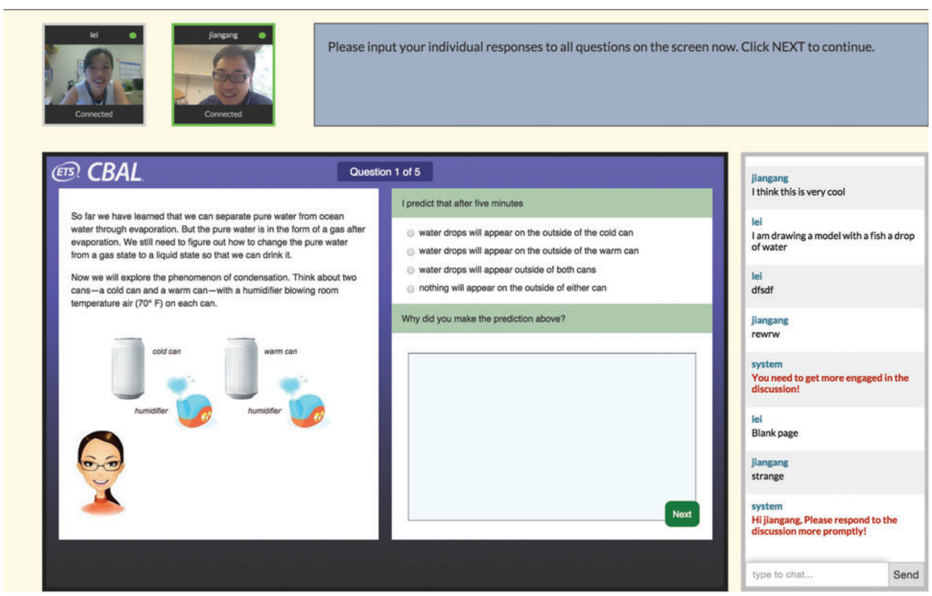

Figure 8 Screenshot of two simple items that are placed in the task module.

Communication Analysis Module

The communication analysis module is essentially a web service that analyzes the stream of communications output from the communication module. Though we consider only text communication in the current implementation, this can be extended to audio and video. The interaction between the communication analysis module and the communication module is governed by two parts: the message-transferring protocols and the message-processing rules implemented as an intelligent facilitator based on natural language processing technology.

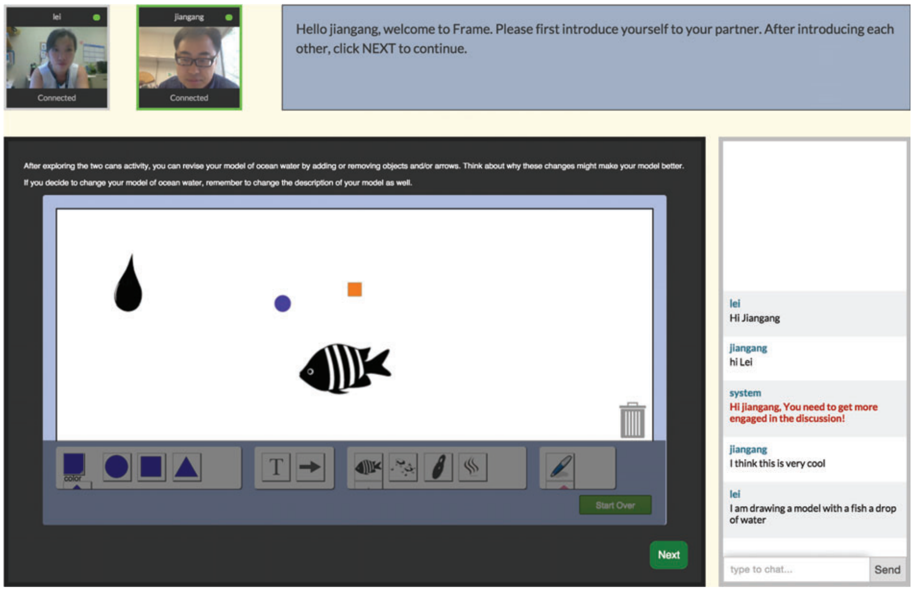

Figure 9 Screenshot of a complex item that involves a shared working space.

Message-Transferring Protocol

Chat messages transfer between the communication module and the communication analysis module according to the following protocols:

1. The communication module will send (via HTTP/HTTPS POST) an initialization (or destruction) message to the communication analysis module once a team is formed (or disbanded). This communication follows the following format:

   {“sessionID”:“789”, “teamID”:“user1_user2”,“target”:“_ALL_”, “time”:“2015-6-24T10:20:15Z”, “message”:“INIT”, “messageType”:“_INITIALIZATION_”}

   where the “sessionID” field is the unique identifier of the session. The “teamID” field is a concatenation of the user IDs, that is, user1, user2, and so on. The “target” field indicates for which user a communication is intended. The “message” field is the main body of the communicated information. The “messageType” field defines the function of this communication.
2. The communication module sends the real-time chats to the communication analysis module in the following format:

   {“sessionID”:“789”, “teamID”:“user1_user2”, “target”:“user1”, “time”:“2015-6-24T10:20:15Z”, “message”: “what is your name?”, “messageType”:“_CHAT_”}
3. Once the communication analysis module receives the message, it will process it and send intervention or feedback messages, when needed, in the following format:

   {“sessionID”:“789”, “teamID”:“user1_user2”, “target”:“user1”, “time”:“2015-6-24T10:20:15Z”, “message”:“Please be more engaged in the work!”, “messageType”:“_FACILITATION_”}

The “target” field specifies to whom (which member of the team) the feedback message should be delivered. When the communication module receives the string, it displays the message to the targeted members as specified by the “target” in the team, as further specified by the “teamID.” If the target field is _ALL_, the message will be displayed on every team member’s chat window. Figure 10 is a schematic of the communication flow between the communication module and the communication analysis module.

Figure 10 Schematic of the communication between EPCAL and communication analysis module.

Table 2 Some Sensors for the Intelligent Feedback

Facilitation name	Details	Detection	Action
Off topic	Member keep typing things irrelevant to the topic	Matching to a set of key words/phrases defined for the topic	Prompt to stay on topic
Inappropriate language	Member types inappropriate words (e.g., profanity words)	Matching to a set of inappropriate words	Prompt to be polite to each other
Less engagement	Member types very few words	Count the number of words	Prompt to be more engaged
Less responsive	Member has long latency before responding to the others	Count the time interval between messages	Prompt to be more responsive to the others

Intelligent Facilitator

The intelligent facilitator is a web service consisting of a set of feature sensors and a set of decision rules for feedback messages. The feature sensors detect some predefined features in the text chat message,[1] and each detection is registered. The decision rules specify the firing of a particular feedback message when the counts of the features reach certain threshold values. We list in Table 2 some examples of the features from the chat messages and the corresponding responses from the intelligent facilitator. This list can be easily expanded, when needed, with different research questions.

Facilitation Module

The facilitation module is used to display the system instructions for the collaboration. In our current design, we have two types of facilitation. The first type is "structure-free," that is, we allow team members to collaborate freely by themselves. The second type is "structured" and requires team members to follow specific steps in their collaboration (Hao et al., 2015). In the case of structured facilitation, the facilitation module prompts team members to follow four steps in working on each item:

1. Each of the members responds individually before any collaboration.
2. After the initial responses, team members are prompted to exchange their answers and discuss to find the best answer.
3. After the collaboration, each of the members is given an opportunity to revise his or her initial response.
4. Finally, one member is randomly chosen as team representative to submit a team answer.

Such a structured facilitation provides scaffolding to the collaboration to ensure that we get the most information from each member before and after collaboration. It is worth noting that we have implemented the preceding two types of

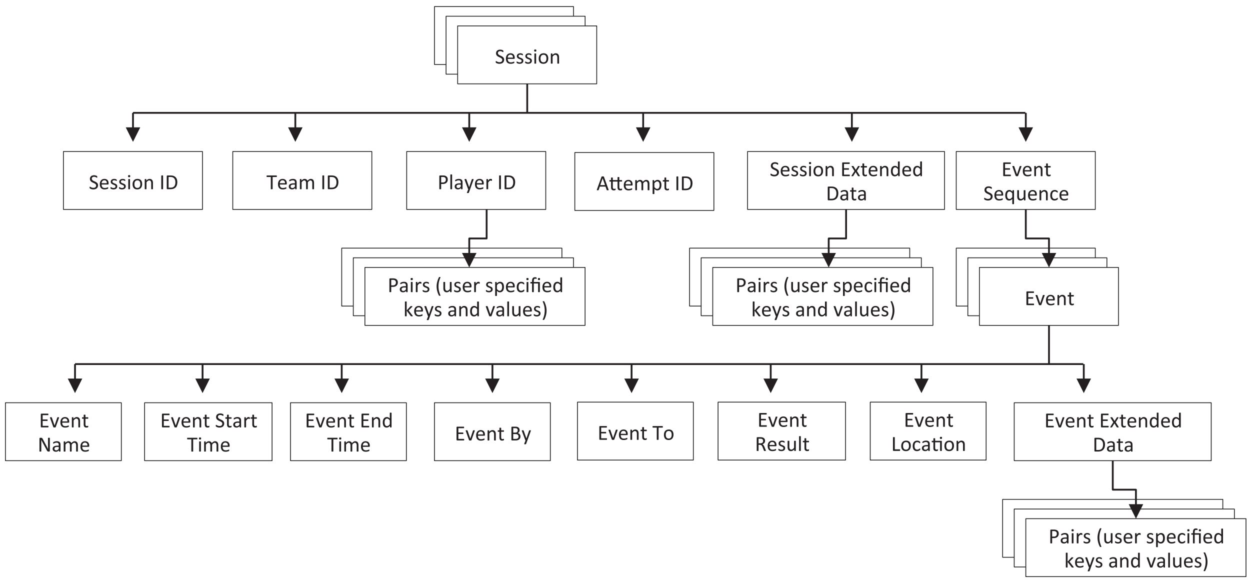

Figure 11 Schematic of the log file structure.

facilitation in the current version of the EPCAL. The EPCAL does allow implementation of other types of facilitation by adjusting the corresponding APIs.

Data and Analytics Module

The data and analytics module collects all the data generated in the task. If the communication medium is text chat, both responses to the items and text chat messages are directly saved into log files with a specifically designed structure in XML format. In Figure 11, we show a schematic of the data model for the log file, which is modified based on Hao, Smith, et al. (2016). In the log file, the basic unit of analysis is called a "session," which corresponds to one attempt at the task by one team. A minimal set of mandatory attributes characterizes a session, and an extended data placeholder stores additional information about the session. The main information holder in each session is a set of events with different attributes. Anything that happens in the task can be considered an event, with a set of attributes to describe it. Communications in audio and video formats are recorded separately for each member of the team. The names and location information of the audio and video files are saved in the log file, while the actual files are saved on a dedicated media server.

In addition to the structured log file, the EPCAL also comes with convenient and customizable analytics support. This analytics support is part of ETS's process-data analytics solution called game log analysis in Python (Hao, Smith, et al., 2016). A submodule named CPS analytics in Python (CAPy) was developed to provide support for data from collaborative tasks. In the still-growing CAPy module, we included a number of convenient functions, such as turn-taking counts, total number of words in a communication, and time latency for communication. All these functions are interfaced via an interactive analytics front end based on Jupyter Notebook (Ragan-Kelley et al., 2014). In Figure 12, we show a screenshot of the Jupyter notebook for the EPCAL analytics.

Software Implementation

The software implementation and art design of the EPCAL were carried out by Pragmatic Solutions Inc. The software implementation consists of modules that support user activity and the synchronization of that activity among two or more connected users. The user-facing portion of the EPCAL is executed in a web browser with modules that deliver the content and with support for the user to communicate via text in the form of chat and to view each connected user via webcam and microphone. Each module is isolated to provide the intended functionality while transmitting to a corresponding service located on a remote server (cloud-based). Figure 13 illustrates the functional client modules executing in the browser space and also indicates the server-side services and how they communicate with each other.

In the following subsections, we introduce the details of the major components.

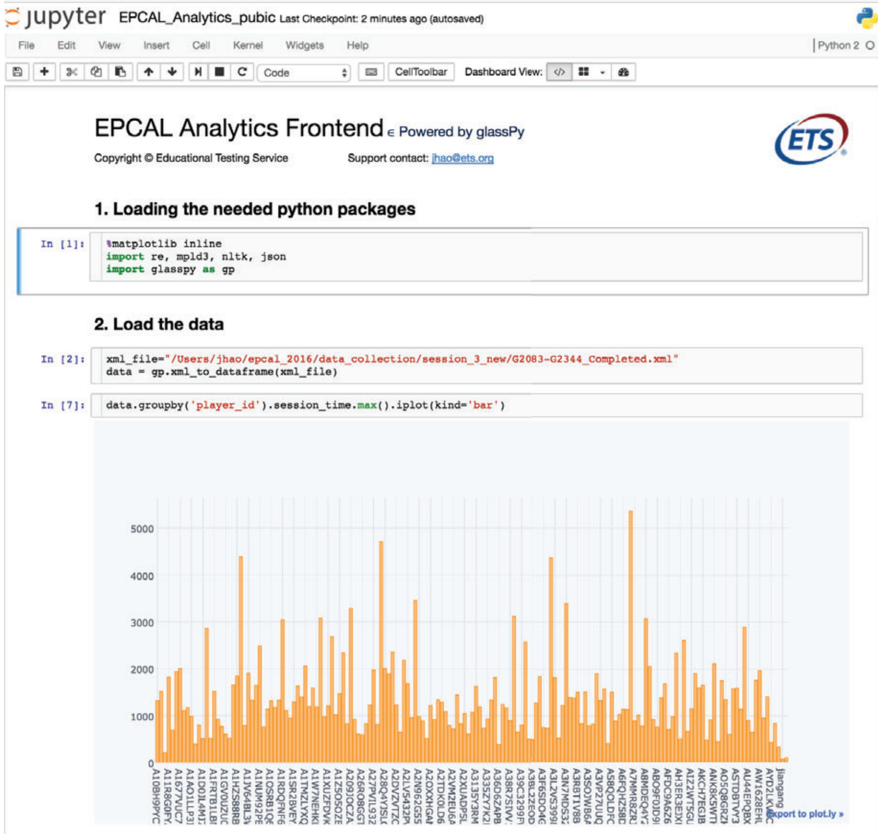

Figure 12 EPCAL analytics front end based on Jupyter Notebook.

Content Delivery

The EPCAL is agnostic to content with some guidelines. Content must be supported in a web browser and therefore be primarily stand-alone HTML (content can use CSS, JavaScript, jQuery, or any other supported language). To be fully supported by the EPCAL, an integration effort is required using the CPS Client or Content API.

CPS Content Application Programming Interface

The CPS Content API has a small set of functions that allow for synchronization of the content among the connected users in a collaboration session. The API supports control over enabling and disabling the chat service, controlling the flow of responses among team members, supplying a single team response for an item, and controlling access to an item by individuals on the team. The CPS Content API works with the EPCAL API to communicate with the EPCAL service to coordinate and synchronize all of the user sessions.

EPCAL Application Programming Interface

The EPCAL API is a closed system to EPCAL and coordinates the user session between the content, chat, and video. The EPCAL is responsible for notifying the EPCAL Service of the users' activity so that it can synchronize the actions among all team members. Figure 13 reflects this by showing the flow of responses and actions of User 1 and how the path to update User *n* is through those dedicated services. A chat message initiated by User 1 is only sent to the central chat service. It is the responsibility of the chat service to update the connected users with the message in the collaboration session.

CPS Chat Engine

The EPCAL has an embedded chat engine that functions like a typical instant messaging service. The user, under the control of the CPS Client API, can enter text as a means of communicating with other team members. The chat

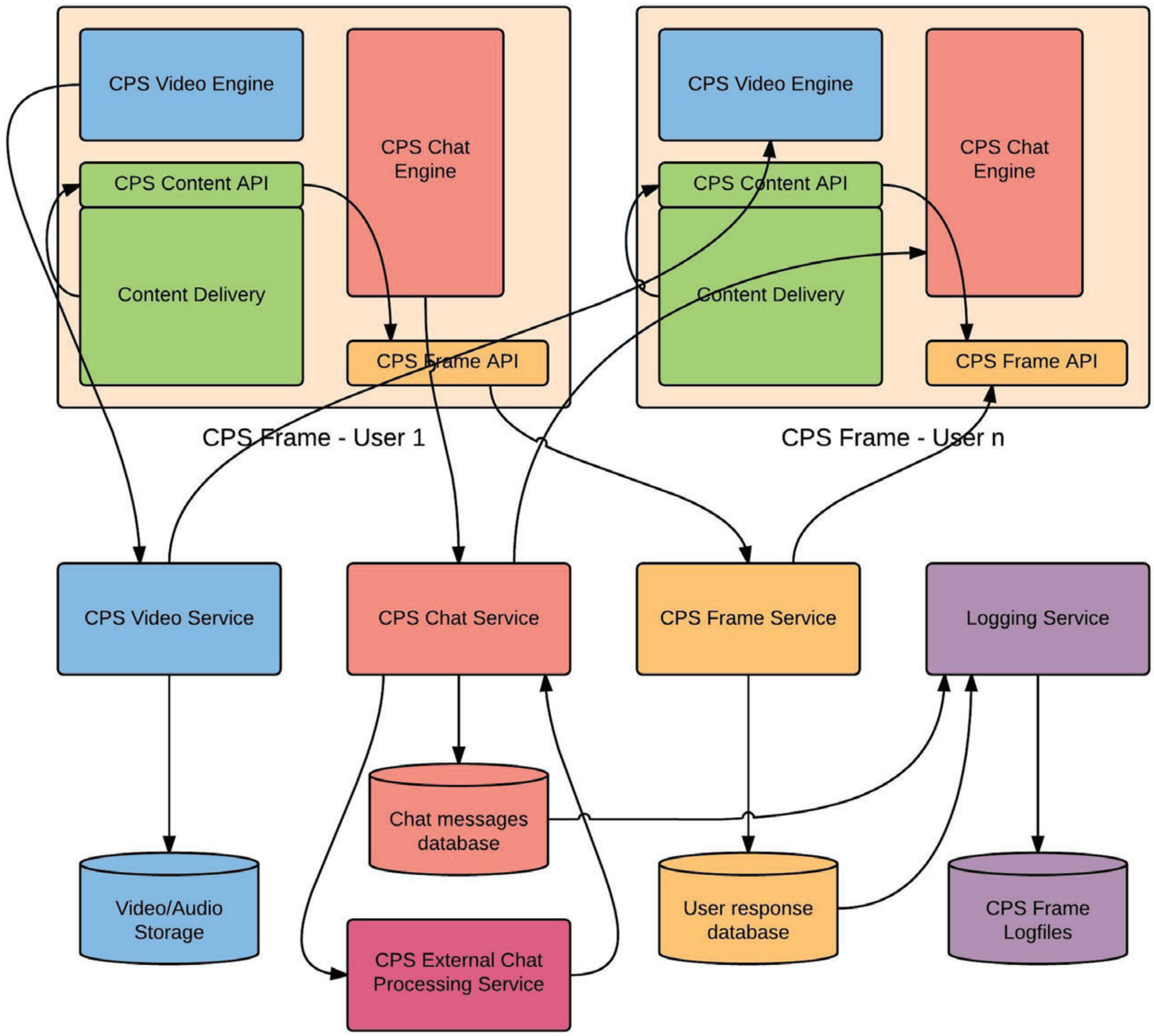

Figure 13 Diagram of the software implementation.

message is sent to the server-side chat service, which is responsible for sending the message to all of the connected team members.

CPS Audio and Video Engine

The CPS audio and video engine requests user access to the local microphone and camera device to stream voice and visual communications to a centralized audio and video service. This EPCAL embedded module is directly connected to the corresponding service in the cloud not only to stream audio and video to other connected users but also to record all of the voice and visual data on the server's storage for access by administrative users.

CPS Audio and Video Service

The CPS audio and video service is dedicated to receiving voice and visual streams from EPCAL users. Each user has a direct connection to the service to save each stream to the server's local storage. Administrative users are able to access these files for download. The service is responsible for synchronizing all of the voice and video streams to provide a seamless audio and video communication system much like a video conferencing system.

CPS Chat Service

The CPS chat service runs on a server in the cloud to facilitate instant messaging – type functionality. The service receives text messages, which are relayed to all of the members in a connected session. The service records each message to identify the user along with a time stamp of when the message was received. The chat service also relays each message to an external processing system that may provide a system message that is intended to be included in the chat stream of a connected session. The chat system supports this as another connected user with the name “SYSTEM” and therefore relays to all users logged in to the chat service database. The external system can analyze communication and provide direction or other prompts to facilitate the collaboration.

EPCAL Service

The EPCAL service primarily receives user responses triggered by content via the CPS Content API. As the user responds to items and clicks “Next” to continue, the Frame Service records all of the activity while also relaying all of the actions to the other connected users. The CPS Content API receives these notifications in order to synchronize the activity of the content. These notifications are how the client is able to control the flow of progress, such as waiting for all users to respond to items.

CPS Logging Service

As a result of the discrete interactions that are supported within the EPCAL environment, a separate server-side service facilitates logging all the activity in one cohesive log file on behalf of all of the users in a collaboration session. The logging service accesses the databases for chat, content activity, and audio and video activity and generates a single XML file representative of those actions to identify the user and time stamp of those actions. The XML file is made available to administrators for viewing and downloading.

Summary

In this report, we summarized six important features for an online platform to support the study of collaborative learning and assessment. We showed that the majority of the existing software tools for collaboration are designed to enable collaboration instead of supporting research on collaboration. As a result, most do not have all these desired features. We introduced the design and development of the EPCAL, which was developed to facilitate research on Internet-based collaboration. The EPCAL is currently the major platform for research on collaboration at ETS and is also available to the research community for a minimal fee to cover hosting and maintenance. In an ongoing research project, we collected more than 800 responses from individuals and teams using the EPCAL, which shows its great stability and scalability. The strength of the EPCAL in managing collaborative activities at large scale also makes it an ideal tool for teachers to carry out computer-supported collaboration in classroom settings.

Acknowledgments

The work reported in this paper was completed while Alina von Davier was employed by ETS.

Note

1 In our current implementation, we consider only text chat as a communication medium, though the same idea can be generalized to audio and video communication.

References

Andrews, J., Kerr, D., Mislevy, R., von Davier, A. A., Hao, J., & Liu, L. (2017). Using a simulation-based task to explore gender and cultural differences in collaboration. *Journal of Educational Measurement, 54*, 54 – 69. https://doi.org/10.1111/jedm.12132

Cohen, E. G., Lotan, R. A., Scarloss, B. A., & Arellano, A. R. (1999). Complex instruction: Equity in cooperative learning classrooms. *Theory Into Practice, 38*, 80 – 86. https://doi.org/10.1080/00405849909543836

DeChurch, L. A., & Mesmer-Magnus, J. R. (2010). The cognitive underpinnings of effective teamwork: A meta-analysis. *Journal of Applied Psychology, 95*, 32–53. https://doi.org/10.1037/a0017328

Flor, M., Yoon, S.-Y., Hao, J., Liu, L., & von Davier, A. (2016, June). *Automated classification of collaborative problem solving interactions in simulated science tasks*. Paper presented at the 11th Workshop on Innovative Use of NLP for Building Educational Applications, San Diego, CA.

Griffin, P., McGaw, B., & Care, E. (2012). *Assessment and teaching of 21st century skills*. New York, NY: Springer.

Halpin, P. F., von Davier, A. A., Hao, J., & Liu, L. (2017). Measuring student engagement during collaboration. *Journal of Educational Measurement, 54*, 70–84. https://doi.org/10.1111/jedm.12133

Hao, J., Liu, L., von Davier, A., & Kyllonen, P. (2015, June). *Assessing collaborative problem solving with simulation based tasks*. Paper presented at the 11th international conference on Computer-Supported Collaborative Learning, Gothenberg, Sweden.

Hao, J., Liu, L., von Davier, A., Kyllonen, P., & Kitchen, C. (2016, June). *Collaborative problem-solving skills versus collaboration outcomes: Findings from statistical analysis and data mining*. Paper presented at the 9th International Conference on Educational Data Mining, Raleigh, NC.

Hao, J., Liu, L., von Davier, A. A., & Kyllonen, P. (2017). Initial steps towards a standardized assessment for CPS: Practical challenges and strategies. In A. A. von Davier, M. Zhu, & P. C. Kyllonen (Eds.), *Innovative assessment of collaboration* (pp. 135–156). Cham, Switzerland: Springer.

Hao, J., Smith, L., Mislevy, R., von Davier, A., & Bauer, M. (2016). *Taming log files from game/simulation-based assessments: Data models and data analysis tools* (Research Report No. RR-16-10). Princeton, NJ: Educational Testing Service. https://doi.org/10.1002/ets2.12096

Liu, L., Hao, J., von Davier, A. A., Kyllonen, P., & Zapata-Rivera, D. (2015). A tough nut to crack: Measuring collaborative problem solving. In Y. Rosen, S. Ferrara, & M. Mosharraf (Eds.), *Handbook of research on technology tools for real-world skill development* (pp. 344–359). Hershey, PA: IGI Global.

O'Neil, H. F. (2014). *Workforce readiness: Competencies and assessment*. New York, NY: Psychology Press.

Organisation for Economic Co-operation and Development. (2017, July). *PISA 2015 draft collaborative problem solving assessment framework*. Retrieved from https://www.oecd.org/pisa/pisaproducts/Draft

Ragan-Kelley, M., Perez, F., Granger, B., Kluyver, T., Ivanov, P., Frederic, J., & Bussonier, M. (2014, December). *The Jupyter/iPython architecture: A unified view of computational research, from interactive exploration to communication and publication*. Paper presented at the fall meeting of the American Geophysical Union, San Francisco, CA. Abstract retrieved from http://adsabs.harvard.edu/abs/2014AGUFM.H44D..07R

von Davier, A. A., & Halpin, P. F. (2013). *Collaborative problem solving and the assessment of cognitive skills: Psychometric considerations* (Research Report No. RR-13-41). Princeton, NJ: Educational Testing Service. https://doi.org/10.1002/j.2333-8504.2013.tb02348.x

von Davier, A. A., Zhu, M., & Patrick, K. (2017). *Innovative assessment of collaboration*. New York, NY: Springer. https://doi.org/10.1007/978-3-319-33261-1

Suggested citation:

Hao, J., Liu, L., von Davier, A. A., Lederer, N., Zapata-Rivera, D., Jakl, P., & Bakkenson, M. (2017). *EPCAL: ETS Platform for Collaborative Assessment and Learning* (Research Report No. RR-17-49). Princeton, NJ: Educational Testing Service. https://doi.org/10.1002/ets2.12181

Action Editor: James Carlson

Reviewers: Eric Hansen