Social Ogmented: JavaMon - Exploring Social Engineering

Aditi Baraskar
School of Computing, Informatics and Decision Systems Engineering, Arizona State University
anbarask@asu.edu

Ananta Soneji
School of Computing, Informatics and Decision Systems Engineering, Arizona State University
asoneji@asu.edu

Saurabh Abhale
School of Computing, Informatics and Decision Systems Engineering, Arizona State University
sabhale@asu.edu

Siva Kongara
School of Computing, Informatics and Decision Systems Engineering, Arizona State University
sskongar@asu.edu

Shaishavkumar Jogani
School of Computing, Informatics and Decision Systems Engineering, Arizona State University
sjogani@asu.edu

ABSTRACT

In this project we have implemented an augmented reality application to help users learn Java programming. We believe, that the idea of representing Java objects by augmented entities may boost the learning curve of any user. Coupled with social engineering, where users are able to interact with each other to learn is always a stepping stone for better understanding. We have implemented collaborative learning tools like interactive augmented environment and a chat box to help users interact and ease their Java learning. Not only a user is evaluated on the basis of his/her individual Java programming quiz scores, but also based on peer ratings with which he/she collaborated. Other parameters of evaluation are overall rank among users, levels completed, average time taken to complete, as well as ratings on basis of knowledge, encouragement and responsiveness. A dashboard has been modelled keeping in mind the Open Social Student Model. The levels completed, average self and peer times, and personal statistics are displayed to the user using interactive visualization techniques. This social augmented system does not only assist a user on basis social engineering model, but also system/peer feedback.

KEYWORDS

Augmented Reality, java programming, learning curve, social engineering, collaborative learning, interactive augmented environment, evaluation, interactive visualization, system/peer feedback, Open Social Student Model

1 INTRODUCTION

Web-based learning is nowadays used as an alternative to face to face learning. The use of web-based learning has increased with the advent of internet and with the increasing proportion of students. This has made both the academic and the research community to develop platforms of interactive content to ease the learning process. The use of multimedia has considerably proven the increasing usage of such platforms. Web and adaptive educational hypermedia are a great in influencing learning experience with technology. With the plethora of information available. It is crucial that the users are able to find the information that is most relevant and beneficial to them.

E-learning has grown and is growing significantly as an educational tool [1]. There has been many research works in this field to understand the needs and learnings of individuals. These works have made efforts in advancing the tool of e-learning. In an e-learning tool, one size does not fit all. Hence, it is very critical to make these systems adaptive and they can even be extended to include social aspects. Among the various design approaches of these tools, two of the most significant are Adaptive e-learning and Social e-learning.

The term Adaptive e-learning is defined by [2] as “a process where learning contents are delivered to learners adaptively, namely, the appropriate contents are delivered to the learners in an appropriate way at an appropriate time based on the learners’ needs, knowledge, preferences and other characteristics”. They define Social e-learning as “a process where connections are made among like-minded learners”. In this type of system, peers interact with each other to share knowledge, skills, abilities and materials.

Personalization facilitates getting the most relevant information. These systems emphasize on increased visibility into the learners state of knowledge. In this work, we present with an augmented application which is adopts the Social learning model. We propose a system which includes major social features based on the open social student model (OSSM) [5]. The proposed system is able to initialize the student model for determining the knowledge level of a student when the student registers for learning the concepts of Java such as polymorphism, inheritance, abstraction and encapsulation.

Once a student learns a java concept together with his peers by playing around with the augmented objects and chatting with his peers to clear any doubts, he/she is then presented with quiz to test the knowledge gained while the system keeps track of all the user activities. The student’s knowledge level, based on the quiz scores, is then updates into the system for use in the social process.

Augmented Reality (AR) and Virtual Reality (VR) are revolutionizing the classroom learning experience by providing immersion in the subject matter. AR is a technology that provides a composite view by superimposing a computer generated image on the user’s view of the real world. Learning through gamified AR can have a significant positive impact on students. It keeps them engaged throughout the lesson and makes learning fun and effortless. Learning applications that use AR can offer vast opportunities to diversify the learning experience. Interactive lessons involving multiple students help improve teamwork skills. We have integrated AR in our learning system. The examples for various concepts explained in the system are represented through various AR objects to simplify learning.

The rest of the report presents the technologies used in section 2; implementation of the system in section 3; results of the system in section 4; system evaluation in section 5; and conclusion in section 6.

2 TECHNOLOGIES USED

This section explains in detail which technologies are used to build the project. Each subsection shows the use of technologies in particular to the development of the system. This section gives a brief idea to the reader on which technologies to refer when considering building augmented systems. Learning was these technologies was essential in building the desired system.

2.1 Unity

Unity is a cross-platform game engine with a built-in IDE developed by Unity Technologies. It is used to develop video games for web plugins, desktop platforms, consoles and mobile devices. Unity provides a game engine that allows the game objects to run in a different environment. It also has a graphical viewer to preview the objects and control their behaviour by attaching the game controller scripts to them.

2.2 Vuforia Augmented Reality SDK

Vuforia Engine provides a platform to build and run augmented reality application in a mobile environment. It uses computer vision to identify the image target and render the augmented reality object in it. We used the Vuforia unity SDK in our project, and deploy and test it in Android devices.

2.3 Unity Multiplayer Network

Unity multiplayer network supports a High level network API that provides server deployment, game state management, scene management, content sharing, match making etc. We used the Unity NetworkManager API for chatting and content sharing.

2.4 Persistent Storage - Device file system

User's login credentials i.e. username and password are stored in the persistent storage for allowing only authorized users access to the system. The Application.persistentDataPath variable defined by unity is used as the file location in which data is stored. As this is the high level API defined by unity, the persistent data path varies according to different platforms (Eg: Android, Windows).

3 APPROACH

This section highlights the various stages of the application. The first section mentions the various flows present in the application. Subsequent sections provide an overview of each stage in the application.

3.1 System Design

The graphical representation of the application flow can be seen in Figure [1] and the interaction between different units of the system is explained as follows:

- The landing page for the user has the option for login as well as register.
- Once the user logs in, he/she is redirected to the dashboard.
- From the dashboard the user can visit the user stats page or start some game level.
- When a game level is launched the Augmented environment is rendered. The user can connect to his peer in this level. He/She would have to enter the chatroom between the participants.
- All the study material is accessible to the user in the augmented environment. Once the user has finished reading the material, he/she can start the quiz.
- All quizzes have three questions. After the user completes the quiz or if the time runs out. The user is required to complete a feedback form.
- Once the user completes the feedback he/she is taken back to the dashboard where the user can view the updated OSSM.

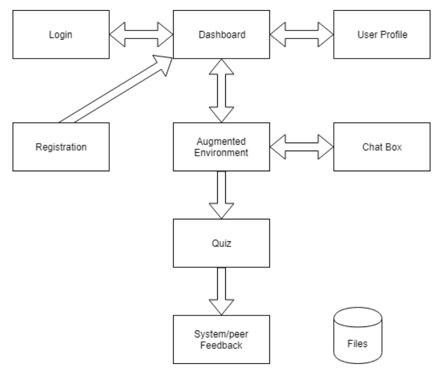

Figure 1: System Flow Diagram

3.2 System Architecture

For multiplayer social augmented collaboration between users, a Network Manager is used to manage connections. Using the High-Level API defined by unity, a host server is used for all the clients to connect to. Each client in our system could be a client, a dedicated server, or a combination of server and client at the same time. The server hosting the multi-client system manages all the connection traffic and interaction between multiple clients. All other clients use the same Unity Scenes and GameObjects.

In our system, different clusters of clients are formed on the basis of levels each user is learning. Level wise clustering of clients is done for effective communication and seamless collaboration for effortless learning. Augmented learning along with chat clients are shared environments between users studying the same level. This helps users collaborate effectively and learn Java programming effectively. Also, 'n' number of clients could be connected to a server and interacted over the Network manager, depending on the hardware processing capabilities of the server.

3.3 Login/Registration

Our social augmented system consists of a login and registration page as shown in Figure [3]. A new user needs to register with the system first before logging in. A user typically registers with a suitable username, password, age and his/her skill set. These parameters are used as explicit feedback from users for Open Social Student

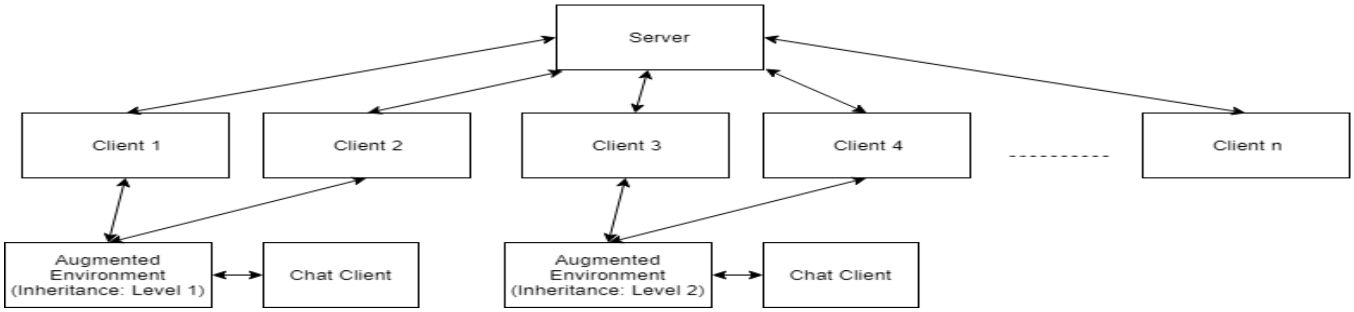

Figure 2: System Architecture Diagram

Modeling. Once, a user registers with the system, a persistent storage is used to store user credentials. This persistent storage (device file system) is used for verifying user login credentials, thus allowing authorized users to access the system's dashboard. After successful login/registration of a user, he/she is redirected to the system's interactive dashboard.

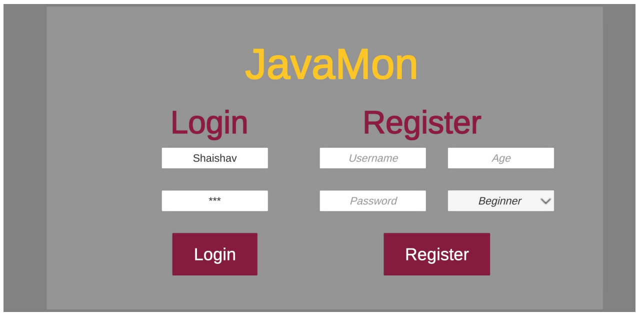

Figure 3: Login/Registration Screen

3.4 Dashboard

The Open Social Student Model has been implemented in our application through the dashboard page as shown in Figure [4]. It explores how visual approaches for open student modeling are provided to and used by students with an easy-to-grasp and holistic view of their progress. Students could use this interface to visualize their own progress, progress of their peers and the cumulative model of the entire class or group.

The page lists the different topics that are available for the user to study from and the various levels associated with it. The complexity of each level is different. Common statistics like the percentage of levels that the user has completed as well as the average completion percentage of the peers is displayed on the page. The average time spent by the user and his/her peers per level is also available.

The distinguishing factor of the dashboard is that the user can identify his performance in each level with respect to his peers. The performance indicator is relative and not absolute. The user can quantify how good or bad is he performing. This has been implemented using color coding. The darker the color for the levels indicated for the user, the better is his/her performance with respect to the average for that level. Similarly, lighter shade of the green color indicates that the time that the user has spent on that level is higher than the average and thus a lower performance. All the unattempted levels are indicated using red color.

The user also has the visibility to the most attempted levels in the system till date. The darker the shade of purple represents that higher percentage of users have successfully attempted the levels.

Availability of these user and peer statistics can aid the user to identify how is he/she performing with respect to their peers and motivate them to do better or help others in the system.

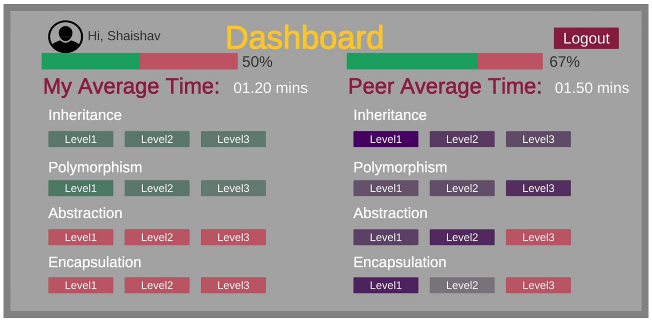

Figure 4: Dashboard Screen

3.5 User Stats

While it is useful to have a social model with peer statistics available. A lot of information can be overwhelming. It is useful to have a page designated to the user's personal statistics. Hence, we introduced a user statistics page (Figure [5]), which lists the statistics like user's rank in the system, average score, average time. The peer feedback received for the user is also listed in the same page. The rating with respect to knowledge of the student, encouragement and responsiveness that he/she provides while working on a level with another student is also presented.

3.6 Augmented Environment

Whenever user selects any level from the dashboard to learn the concept, the camera screen appears as displayed in Figure [7]. The screen has 3 components: The content of level, camera, and chat window. User can read the content from the slide panel and learn the concept. Each level has its own content related the concept and

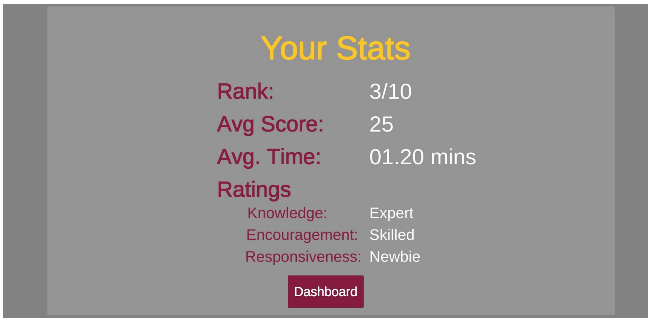

Figure 5: User Stats Screen

Figure 6: Image Target

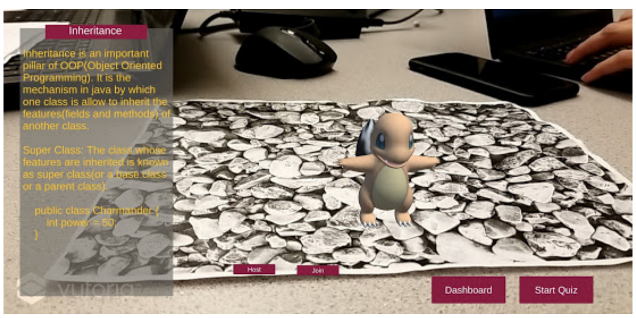

Figure 7: Chapter 1 of inheritance

difficulty level. The scrollable content provides the detail understanding to the user on the topic. Further, camera screen checks continuously for the Image Target to appear. Image Targets represent images that Vuforia Engine can detect and track. Unlike traditional fiducial markers, data matrix codes, and QR codes, Image Targets do not need special black and white regions or codes to be recognized. The Engine detects and tracks the features that are naturally found in the image itself by comparing these natural features against a known target resource database. Once the Image Target is detected, Vuforia Engine will track the image as long as it is at least partially in the camera's field of view. We used the "stones" image as display in Figure [6] for our image target. First, we uploaded it in the Vuforia

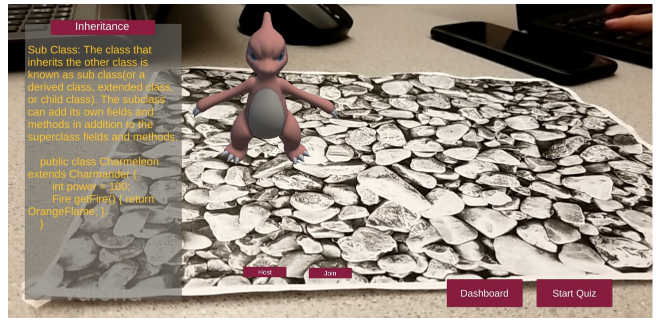

Figure 8: Chapter 2 of inheritance

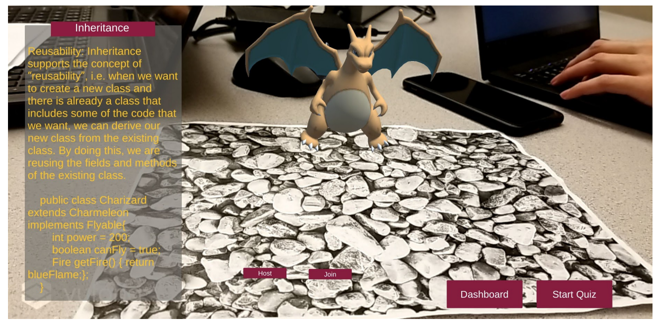

Figure 9: Chapter 3 of inheritance

image target database and then we downloaded it and sync it with our unity engine.

Once the camera captures the image target, it renders the AR object on the image. The AR objects are interactive and make the learning more interesting. The content of each level is divided into the chapters and distributed among several AR objects in that level. Each AR object has its own content that is displayed on the content panel. User can interact with virtual objects by clicking on them. When clicking on any AR objects, the system moves into the next chapter by changing the content and AR object accordingly. AR object selection is dependent on the content it is associated with.

Illustration on how the learning is done in the level is explained in the following example of Inheritance Level 1. First AR object in this level the popular Pokemon Charmander (Figure [7]). The content for that object explains the inheritance and super class definitions and defines the Charmander class as a superclass. On clicking of the Charmander AR object, the topic chapter changes. The new pokemon character Chameleon appears and content changes (Figure [8]). Here, the content defines the sub-class. In this chapter we defined the class Charmeleon which extends the Charmander class from previous chapter inheriting all its properties. On interacting with Charmeleon, next pokemon character Charizard appears and the content defines the Re-usability of class (Figure [9]). Here, the Charizard class extends the Charmeleon class and implements the Flyable interface. The topic has these three chapters and on clicking of Charizard, system will go into its initial Charmander chapter. Here you can see that we try to make this application interesting that is especially appealing to the kids with the use of Pokemon objects. In the pokemon, Charizard evolves from the Charmeleon which evolves from the Charmander. We used this fact in our system that helps the user in learning the superclass, subclass, etc. inheritance features.

As you can see in Figure [7] that there is a button to join the collaboration room of that level. Once user clicks on that level and joins the room the chat panel appears. We explained the Chat functionality in detail in next section.

Once the user joins the collaboration room of that level, apart from chatting another feature we provided is the social learning. That is now all the AR objects and content is in sync with the peers in that room. So, not only they can collaborate through chat, but also they can see the same content on the screen at any moment of time. Any interaction done by one user, its changes are also reflected to other users. This feature is really important when peer wants to guide the user by navigating it through different chapters and users can discuss and share knowledge while learning and interacting with the augmented objects. We used Multiplayer network functionality of Unity to implement this feature, which is explain in detail in the next section.

3.7 Chat

In social learning, it is crucial for the students/users to communicate with each other. This plays a key role if it is provided at the time of learning. Users can share their ideas and understandings, and thus help each other in collectively progressing as a group. In our project, we used this feature extensively so that the students can join chat groups (Figure [10])and see all the students in the same level. This encourages the student to ask his/her peers for more explanations that the system has provided.

More learning is achieved through this manner as the user receives more information than the system contains through the social aspect. We have provided an option to chat when the camera is running and when the users see the augmented objects. By providing the chat window in the same scene as the augmented level, we provide users with a pleasant experience of not having to navigate between scenes while trying to understand/clear the level and at the same time communicating with other users.

We have used UNet Multiplayer Chat - Unity Asset to achieve the chat feature in our project. The asset was imported into the project through the unity asset store. This asset uses Unity Networking as its base layer for network protocol. We have three options in this asset: the first one being able to run only as the chat server, the next one being able to run as a host i.e. a chat server and a client at the same time, and the final one being able to connect as a client. All the clients are registered in the chat manager and all the clients send the message to the server, while the server is responsible for sending out the messages to all the registered clients.

In our project, since it was tricky to get the network address on mobile, we decided to run the chat server on a PC. We have used the PC's network address in the app configuration and deployed the app to the mobiles. So, now when the users connect to the chat application as clients, they get connected to the server which is running on the PC. We also have the flexibility to choose on which port the chat server is running, but we left it to run on the default port of 7777.

We used this network setup for both chatting and screen sharing space explained in previous section. We created our own message transferring format to differentiate the chat and screen sharing messages. We processed them once the client receives any message from the server and operated them accordingly.

Initially the user can start the level on his own, but when he/she feels that he needs help, he can join the chat. We have provided the user with a button in the scene to join. Once he joins the chat server, we prompt the user to enter his chat name. This is done so that he can remain anonymous for the rest of the users if he/she wishes to. Through this, we have achieved total privacy for the user. Now, the user can send and receive chat messages from the users in the same level.

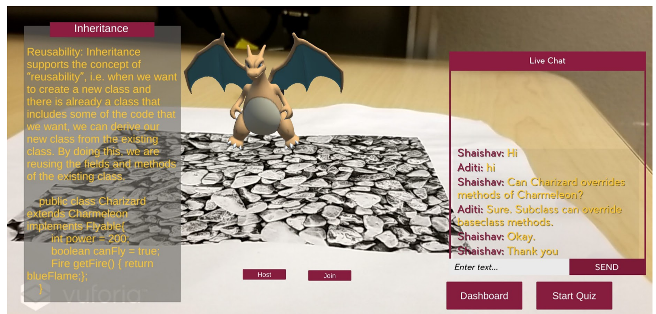

Figure 10: Content, object and chat sharing

3.8 Quiz

We used the assessment tool in the form of multiple choice question to measure user's knowledge, abilities, and skill in a particular topic. Each level has a quiz related to the concept learnt in that level (Figure [11]). User can read the content in the Augmented reality game and learn the concept while collaborating with peers and interacting with the AR objects. Each quiz has around three questions which test how much has the user grasped from the study session with the peers. In addition to the correctness of each user's answer in the question, we are also considering the time taken to finish the quiz to evaluate the user's efficiency and effectiveness in that level.

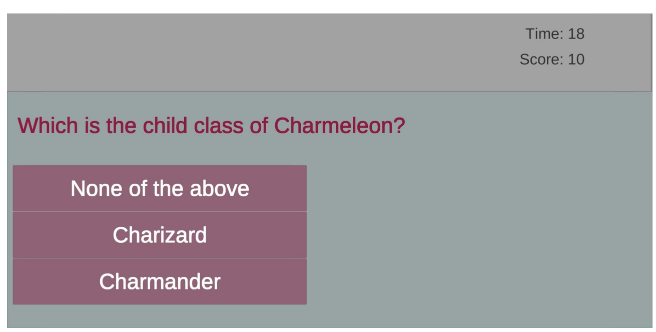

Figure 11: Quiz Screen

3.9 Peer and System Evaluation

Feedback is an important measure for improvement. In our system we concentrate on two feedbacks from the user; one for each peer that the user was paired with for the game and one for the level of the game (Figure [12]).

The idea of collecting the feedback for the peer is bifold. By collecting the feedback for the user from their peers, we can inform the user about the specifics of his interaction during the game. The peer ratings are evaluated and the user gets the results as one of the three categories; Newbie, Skilled, Expert. The feedback is gathered with respect to knowledge, encouragement and responsiveness.

The rating for knowledge indicates how well-versed is the user with the particular topic. Knowledge is an important aspect in a learning system but in a social learning system, encouragement and responsiveness towards the peer is equally crucial. A user with encouragement rating newbie can then focus on providing encouragement in the subsequent levels to achieve a better rating. This type of user rating would help improve the quality of social interaction in a learning system.

We also collect the feedback for the level from the users. The feedback for the level is with respect to three attributes; Content, Navigation and Presentation. The motivation behind this was to ensure that the user expectations regarding the content of the system are being met. Ratings are on the scale of one to three. The navigation aspect concentrates on whether the relevant content was easily accessible for the user. Whereas the presentation focuses on the layout and aesthetics of the level.

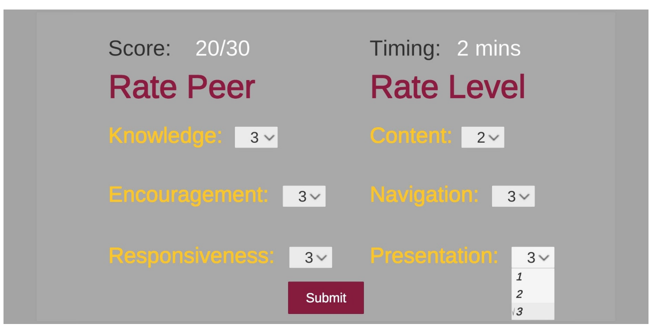

Figure 12: Peer and System Evaluation

4 RESULTS

In order to upgrade the system and handle the situation of *cold-start*, we have taken some of explicit and implicit feedback from the users of the system. The explicit feedback has helped us handle the *cold-start* problem and the implicit feedback will help us update the system for better performance.

Explicit Feedback

- Age
- Java Experience (Beginner, Intermediate, Experienced)
- Peer Rating (Knowledge, Encouragement, Responsiveness)
- Level Feedback (Content, Navigation, Presentation)

Implicit Feedback

- Time spent per level
- Performance in Quizzes
- Learning Curve
- Peer collaboration
- Difficulty level per topic

5 EVALUATION

The developed system provides a social learning platform for students to understand java concepts such as polymorphism, inheritance, abstraction, and encapsulation. In order to make this system more usable for future, we need to evaluate. By going this, we want to make sure that the quality of the developed system is retained.

To do so, we need to make both the interactions with the peers and the system should be upgraded with time. The peer evaluation discussed in section 3.9 helps us to gather the data related to each and every individual in terms of their knowledge, collaboration and responsiveness.

Moreover, as and when the users interact with the system, they might have some suggestions on how they want the system to do. In the lieu of accomplishing this task, we are taking the feedback from our users on the system as well with parameters for rating such as navigation, content, and presentation.

These system related parameters are not enough for us to evaluate how the system should be presented to its users. In order to evaluate the designed and developed system, we consider the following aspects.

5.1 Quantitative Metrics

- Evaluation of user tasks (quizzes in our project).
- Analysis and use of data for producing results
- Measurement of recommendation quality
- Measurement of all other possible system interactions by users
- User-based evaluation of the system (level feedback in our project after quiz completion)

5.2 Qualitative Metrics

- Evaluation of the system in terms of non-accuracy metrics
- How novel is the recommendation algorithm? (recommendation of other users to pair with)
- How much satisfied are the users with our system?
- How does our system behave in different scenarios?

6 CONCLUSION

Social learning is not something which is new in general. People learn by observing the behavior of the people surrounding them, trying to imitate them, trying to learn from them by asking for help. The main goal of the project was to take this general idea and incorporate into an application that facilitates programming language.

Open social student model and the ability to communicate between the users of the application played a key role in achieving this goal. The collective augmentation aspect of the project was used to enhance the learning as the users could visualize while communicating with their peers. The application proved to be an interesting way to learn new languages with the help of other students and their profiles.

REFERENCES

[1] Li, H., and Masters, J. (2009). E-Learning and Knowledge Management in the Early Years: Where are we and where should we go. Knowledge Management and E-Learning, 1(4), 245.

[2] Shi, L., Cristea, A. I., Foss, J. G., Al Qudah, D., and Qaffas, A. (2013). A social personalized adaptive e-learning environment: a case study in Topolor. IADIS Int. J. WWWInternet, 11(2).

[3] Unity Documentation, https://docs.unity3d.com/Manual/index.html

[4] Vuforia Documentation, https://library.vuforia.com/getting-started/overview.html

[5] Hsiao, I. H., Bakalov, F., Brusilovsky, P., and König-Ries, B. (2011, July). Open social student modeling: visualizing student models with parallel introspectiveviews. In International Conference on User Modeling, Adaptation, and Personalization (pp. 171-182). Springer, Berlin, Heidelberg.

[6] Bakalov, F., Hsiao, I. H., Brusilovsky, P., and König-Ries, B. (2011). Visualizing student models for social learning with Parallel IntrospectiveViews. In Proc. of the Workshop on Visual Interfaces to the Social and Semantic Web at the International Conference on Intelligent User Interfaces.

[7] Dimitrova, V., Self, J., and Brna, P. (2001, July). Applying interactive open learner models to learning technical terminology. In International Conference on User Modeling (pp. 148-157). Springer, Berlin, Heidelberg.