

# CaFE Play: A Customizable Mobile Phone Game Framework for Enhancing English Literacy

Senior Honors Research Thesis Extended Abstract

Jonathan Harbuck

Advisor: M. Bernardine Dias

English proficiency is an invaluable skill in many parts of the world. It often helps individuals improve their quality of life and contribute more to society. However, for various reasons, many individuals have difficulty learning English, often simply due to a lack of guided practice. To help address this issue, the TechBridgeWorld research group ([www.techbridgeworld.org](http://www.techbridgeworld.org)) through their TechCaFE program (<http://www.techbridgeworld.org/techcafe/>) has developed the CaFE Phone tool: a prototype mobile-phone-based game for enhancing English literacy. The purpose of this thesis is to improve and build upon the game by creating a game customization framework, CaFE Play, which can be used to not only customize the existing game, but also to create other such customizable games. These games should increase the user's motivation to practice English literacy, and thereby better improve the user's English proficiency, a hypothesis that this thesis also aims to test.

## English Proficiency

Many people, both within and without the United States, have difficulty with English for various reasons. For example, because of the differences in the grammar and structure of English compared to American Sign Language, many Americans who are deaf or hard of hearing lack proficiency in English. The median reading level of deaf high school graduates is that of a fourth grader (1).

Another group that tends to have difficulty with English is the refugee population within the United States. Many refugees arrive in the United States with nothing more than the clothes they are wearing. They receive help to meet their immediate needs, but are usually given one year to become self-sufficient (2). During this period, they are extremely busy and many lack the time to receive formal English language training, but knowledge of the English language would be one of the most valuable skills for helping them adjust and become self-sufficient in their host country.

Improved proficiency in English would be a major benefit to these and other groups. Perhaps most importantly, it would allow them to find and secure better paying and more enjoyable employment. For example, immigrants who are literate in English are more likely to have continuous employment and earn more than those who are literate only in a language other than English (3). English literacy would also help them socially, by allowing them to make more friends and acquaintances, thereby also providing networking opportunities. They would have the foundation necessary to learn and acquire other skills, both in and out of formal education, and they would be better able to benefit from help offered by others. In general, they would be better able to prosper in society, and, in turn, contribute even more to society.

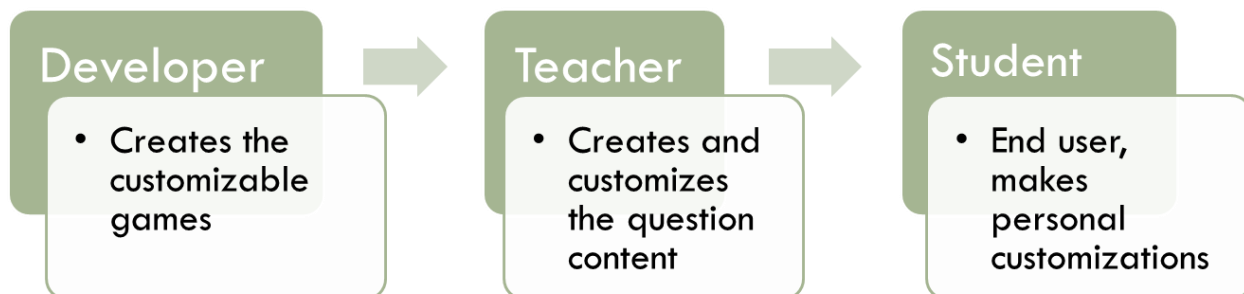
In order to help such groups, the purpose of this thesis has been to enhance the CaFE Phone mobile-phone-based tool designed for enhancing English Literacy. Mobile phones provide an ideal platform for such an educational tool. They are widely used, and, for many people, they are the only accessible computing technology. Many groups that might be thought not to use mobile phones, in fact, can and do use them. For example, many mobile phone service providers have data-only plans, specifically designed for the deaf and hard-of-hearing (4; 5). Also, many low-income Americans are eligible for a free wireless phone plan and a free phone (6).

## **Customization, Motivation, and Learning**

The central way this thesis aims to improve the existing CaFE Phone tool is to introduce more customization and control. Customization and control play an important role in games and learning, as they have a significant impact on effectiveness and motivation. While research has shown that giving learners control over parts of instructional programs has had mixed results on achievement, such control has given consistent, positive results on motivation and user reactions (7). One study by Cordova and Lepper found that giving students control over instructionally irrelevant aspects of a learning activity increases their motivation and learning (7). By creating a game framework that gives students control over various aspects of the game, including their avatar, and allowing games to offer their own customizations based on user interaction, it is our hypothesis that students will show more interest and motivation, and increase their learning. This will also provide another test of Cordova and Lepper's findings, specific to an educational game. Similarly, teacher customization should increase effectiveness by allowing the teacher to target the game for a specific learning activity, and by allowing the teacher to customize the content for the students. The teacher is the education professional, and knows what is best for his or her students. With teacher customizations, a game can be targeted to the level of knowledge of the students, and can be used to practice whatever the teacher is covering in class at the time.

## **CaFE Play: Framework Design**

To enhance the degree of customization in the CaFE Phone tool, we designed and implemented a game customization framework named CaFE Play. CaFE Play makes it possible for developers to create games that give teachers and students customization and control options to increase motivation and improve the effectiveness of the game. The framework is divided into three categories, based on a user's role in the process: Developer, Teacher, and Student.

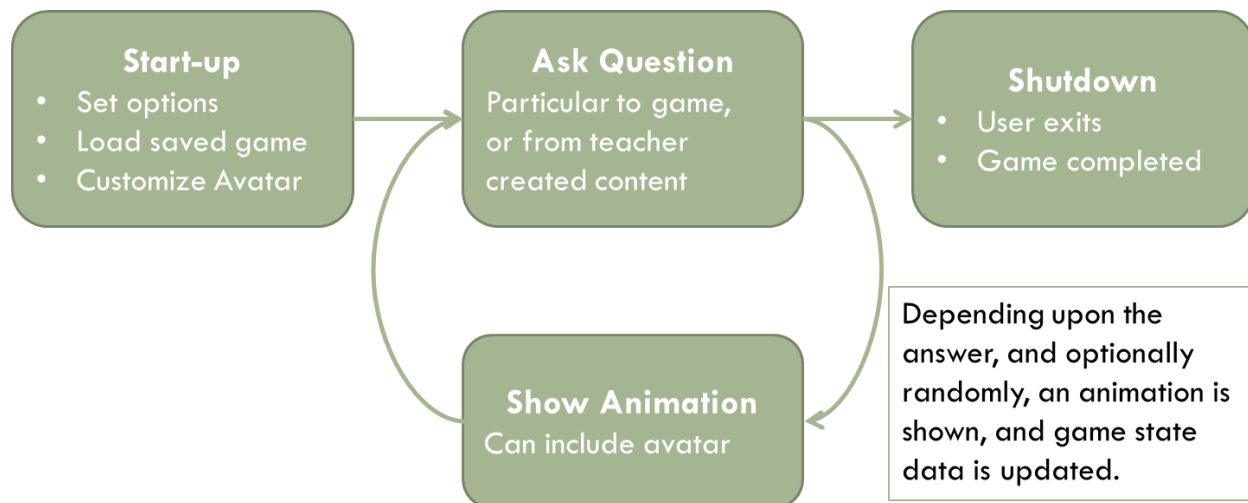


**FIGURE 1: ROLES WITHIN THE FRAMEWORK**

Each role is distinct, and a person in one role can be completely separated from those in other roles. It is possible, for example, for those who are not students, but want to practice English independently, to fulfill the role of student. They should be able to download a game that a developer created, and download question content that a teacher created, in order to practice English, but without having any personal contact with the developer or the teacher. While this scenario is both feasible and likely, it is not optimal. Developers working in conjunction with teachers and students would be more effective than working independently. Teachers and students know much more about the students' specific needs, interests, and abilities than a developer would. Needs, interests, and abilities of one group of students are also likely to be shared by other students. In the likely case that a developer is creating a game for general use, working with a teacher and a set of students would provide a useful window into the needs, interests, and abilities of a much larger group of students. More important than the developer's relationship with the teacher and students is the students' relationship with the teacher. The teacher's ability to assist the students can have a significant impact on the students' learning. It has also been shown that being debriefed by a teacher, after playing an educational game is a crucial aspect of a gaming or simulation experience, since it helps in reflection, understanding, and application (7).

### **The Role of the Developer**

The developer uses our Application Programming Interface (API) to create games that are based on the model shown below. The developer does need to be able to program in Java, but the API documentation and examples should allow someone with somewhat limited Java experience to create a new game for the framework. More experienced Java developers, however, would be able to create more advanced and complex games.



**FIGURE 2: GAME DESIGN OVERVIEW**

The API includes a standard main menu and standard options. From the main menu, the user can customize an avatar, using a screen that the developer can create using the API. The developer creates question stages, and their ordering, which are shown to the user when the game is started. There are two standard types of question stages that the developer can create: Game question stages and instructional question stages. Game question stages present questions that are particular to the game and are created by the game developer. Depending on the game, this type of question stage could ask questions such as to which player the avatar should pass the ball, or down which path the avatar should walk. Instructional question stages ask questions created by a teacher and are loaded into the instance of the game. The developer has no control over the specific questions asked in instructional question stages. The developer uses the API to create animations that are shown after a question is answered, and depend on the answer the user gave. For an instructional question stage, the developer can create one or more animations for a correct answer, and one or more animations for an incorrect answer. For a game question stage, the developer can create one or more animations for each of the answers. If more than one animation is given for an answer, an animation will be played randomly. After an animation is played, the next question stage is shown. The ordering of question stages is also chosen by the developer, and, similarly to the animations, can depend on the answer given and can be chosen randomly from options given by the developer. In this way, the developer can create a flow of question stages, which could be represented by a directed graph or a flow chart. This gives the developer the flexibility to include other good educational game practices and give the user more control in the way the game unfolds. The game can also be made to update and retrieve game data, such as the current score, the current level, or data specific to the game. Questions and animations can be updated based on the game data.

### **The Role of the Teacher**

The teacher’s main role is to create and customize sets of questions to be asked in a game, by using the CaFE Teach content authoring tool. CaFE Teach also lets the teacher specify a question’s difficulty level and optionally provide a hint and explanation for a question. The presentation of the questions can be

customized, including ordering (random ordering is also possible), repetition, and whether or not all questions must be asked or only a subset of the questions should be chosen. These options also give the teacher the opportunity to use the game as a testing mechanism, with every student being asked the exact same questions, in the order the teacher specifies.

### The Role of the Student

Students not only play the games created by developers with the questions created by teachers, but also exert their own control over the game. At the beginning of the game, the student customizes the avatar, either by selecting a premade avatar, or creating an avatar from composite parts. Depending on the game, the student can also customize the opponent. For example, in a soccer game, the user can select for the avatar a uniform with the colors of the student's favorite team, and select for the opponent a uniform with the colors of the corresponding rival team. Students can also customize game options, including whether or not sounds and animations should be played. Beyond these customizations possible in all games, students will be able to exert control and make customizations unique to a specific game. Developers can make this possible using game question stages and through tracking game data, as explained in the section about the [role of the developer](#).

### Implementation Details

Changes were made to the existing game code, which was also split into a general API that developers can use and code specific to the soccer game implementation. The API consists of three main parts: graphical objects, question processing, and game stages.

#### Graphical Objects

In the previous version of the game, the animations consisted solely of animated GIF files. These were easy to create externally, but any customizations to the animations, including any customized avatar, would require entirely new animations as animated GIF files. In order to allow for graphical customizations to the animations and the avatar, a new implementation was chosen, which includes customizable animations and sprites. Animations are now created using new classes that use LCDUI, a graphics package native to Java ME. Developers can create an Animation object by specifying a background image, and adding moving sprites to the animation, which appear and move based on which step the animation is on. The MovingSprite class extends the LCDUI Sprite class, and allows for pre-programmed movement and appearance of the sprite on the screen, in addition to the standard sprite options, such as creating a sprite with multiple frames from a single image.

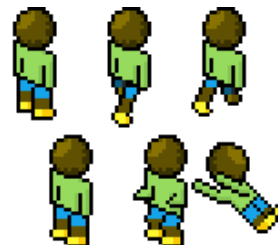
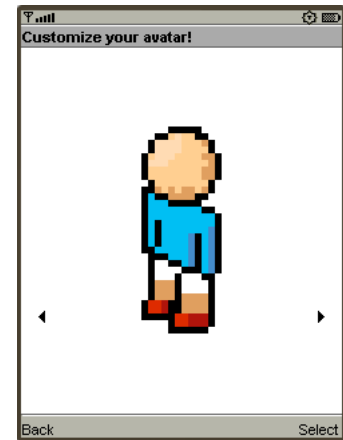


FIGURE 3: CHARACTER SPRITE IMAGE

The MovingSprite class extends the LCDUI Sprite class, and allows for pre-programmed movement and appearance of the sprite on the screen, in addition to the standard sprite options, such as creating a sprite with multiple frames from a single image. Developers can have the user select an avatar and an opponent, if desired, in one of two ways: Whole avatar selection and composite avatar selection. The developer can offer one or both types of avatar selection. With whole avatar selection, the user selects a complete pre-made avatar, such as the one shown in Figure 3. This is done with a screen similar to the one in Figure 4. With composite avatar selection, the user creates an avatar by selecting from various parts. In the example shown in Figure 4,

the user can choose a specific head, shirt, shorts, legs, and shoes. The composite parts chosen by the user are added to a CompositeAvatarBuilder object, which keeps track of each part as a separate MovingSprite, but allows for their movement and appearance properties to be modified all together.

Instances of Animation objects can be created by the developer by extending the AnimationBuilder class and specifying the various parameters of the animation objects. Before each animation is played, it is created as a new instance by the AnimationBuilder class, and can differ based on the selected avatar and opponent, and any current game state, such as the level or the number of points.



**FIGURE 4: COMPOSITE AVATAR SELECTION SCREEN**

### Question Processing

The question processing that was part of the previous version of the game has been simplified and updated to accommodate the desired customizations by the teacher. Questions and customizations are exported by CaFE Teach as XML files, which are read by the QuestionManager class of the game and stored in QuestionAnswer objects, which are for questions with a written response by the user, and MultipleChoiceQuestionAnswer objects, which are for questions with multiple answer choices. A MultipleChoiceQuestionAnswer object can hold data for various multiple choice question types, including questions that have an image as part of the question, questions that have images as answers, and regular text multiple choice questions. The QuestionManager class now processes the various XML tags and text that specify the customizations chosen by the teacher, as explained in [The Role of the Teacher](#) section. QuestionManager can also still process XML files that were created before CaFE Teach was updated for this thesis, allowing for backward compatibility, which assumes default values for unspecified customizations.

### Game Stages

The API also includes classes related to stages of the game, including MainMenu, InstructionalQuestionStage, and GameQuestionStage. MainMenu provides a standard main menu that can be shown at the beginning of the game, and when the user returns back to the main menu while playing the game. The developer needs to extend the MainMenu class to allow for any desired avatar and opponent selection and to create the stages of the game and specify their flow. The two types of stages of the game are InstructionalQuestionStage and GameQuestionStage, which are explained in the section [The Role of the Developer](#).

### Testing and Evaluation

In order to test the effectiveness of the customization framework and the specific games created, data collection in the game also needs to improve. In the previous version of the game, the only way to collect information on student achievement was to examine the student's score, which was displayed at the end of the game. To test for motivation and learning, other data should be collected by the game,

such as how long and often students play the game, and which questions are answered correctly and which are answered incorrectly.

Using the improved data collection, the framework will be tested at the Western Pennsylvania School for the Deaf. The students participating in the study will be divided randomly into two groups. The students in one group will be able to choose their avatars for the game, while the students in the other group will be assigned an avatar for the game. After a pre-test and a chance to play the game in class on mobile phones provided by TechBridgeWorld, the students will take the phones home with them over their Spring Break. After Spring Break, the phones will be gathered, and a post-test will be given. Usage data will be collected from the phones, which will allow us to examine what effect the avatar customization had on the desire of the students to play the game in their free time. This can also be compared with the results of the post-test, to see if the customization had an effect on student performance. After Spring Break, we will also conduct some in-class testing, which will allow us to gather results and collect feedback on the framework in a more controlled environment.

## Works Cited

1. *How Do Profoundly Deaf Children Learn to Read?* **Goldin-Meadow, Susan and Mayberry, Rachel I.** 4, *Learning Disabilities Research & Practice*, Vol. 16, pp. 222-229. <http://idiom.ucsd.edu/~rmayberry/pubs/GoldinMeadow-Mayberry.pdf>.
2. **U.S. Committee for Refugees and Immigrants.** Refugee Resettlement. [Online] [Cited: March 15, 2011.] <http://www.refugees.org/our-work/refugee-resettlement/>.
3. **Burt, Miriam.** Issues in Improving Immigrant Workers' English Language Skills. *Center for Adult English Language Acquisition*. [Online] December 2003. [Cited: March 15, 2011.] [http://www.cal.org/caela/esl\\_resources/digests/workplaceissues.html](http://www.cal.org/caela/esl_resources/digests/workplaceissues.html).
4. **Sprint.** *Sprint Relay Store*. [Online] [Cited: March 15, 2011.] <http://www.sprintrelaystore.com/>.
5. **AT&T.** Text Accessibility Plans for iPhone®. *Wireless from AT&T*. [Online] [Cited: March 15, 2011.] <http://www.wireless.att.com/learn/articles-resources/disability-resources/text-accessibility-plan-for-iphone.jsp>.
6. **SafeLink Wireless.** Lifeline/SafeLink Fact Sheet. [Online] [Cited: March 15, 2011.] <https://www.safelinkwireless.com/EnrollmentPublic/benefits.aspx>.
7. *Games, Motivation, and Learning: A Research and Practice Model.* **Garris, R., Ahlers, R. and Driskell, J. E.** 4, 2002, *Simulation and Gaming*, Vol. 33, pp. 441-467.