Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

A Customizable Game Engine for Mobile Game-Based Learning

S. Mininel, F. Vatta, S. Gaion, W. Ukovich
DEEI
University of Trieste
Italy
mininel@gnbts.units.it

M.P. Fanti
DEE
Polytechnic of Bari
Italy

***Abstract*— The use of computers in education has greatly increased during the last two decades. At the same time, technology advances have opened new spaces and possibilities for the field of computer-based edutainment—education in the form of entertainment—where learners can achieve their learning goals while having fun. Games on mobile phones have become a significant part of the contemporary culture experienced by young people. Research indicates the potential of mobile games to encourage learning in young adults. The 3-year EC-supported project mGBL (mobile Game-Based Learning) had the objective to prototype a platform for the development and deployment of mobile learning and guidance games, able to support the learning process and the support of decision making in critical situations not only in a cognitive but also in an emotional way. This paper describes key issues emerged in development phase of the Mogabal game engine within mGBL framework in both technical and pedagogical aspects, showing technologies, strategies and methodology adopted. A number of game prototypes based on such engine were devised and some were tested during the project user-trials. These games prototypes demonstrate the capabilities of the devised engine to cover a wide range of different games types and educational contents.**

***Keywords*— game-based learning, mobile devices, edutainment, simulation, training systems.**

I. INTRODUCTION: VIDEOGAMES, GAME-BASED LEARNING AND EDUTAINING

Videogames industry has been growing at impressive rate in the last years and according to several reports it even succeeded in historical surpasses, collecting an early revenue superior to those of film and music industries respectively. While periodically issues are raised, from some associations, regarding violent contents, it is a generally accepted evaluation that many successful mainstream videogames in recent years showed interesting educational contents, although sometimes that happened just as a side-effect of game mechanics. Examples that come to mind, regarding game genre, conferred educational content and famous game titles are:

- Historical simulation: history knowledge ("*Europa Universalis*").
- City managements: understanding some logistics problems ("*SimCity*™").
- Civilization management: history of science ("*Sid Meier's Civilization*").
- Social simulation: rules of social behaviour ("*The Sims*™").

Sometimes the "game engines" of such games could be adapted for versions fully devoted to education, but it's usually not an easy task. There is a consistent effort, in literature and in industry, to exploit the potentialities of videogames in a more focused learning effort: the so called Game-Based Learning. Although the word is relatively recent, the concept of "Edutainment" (education & entertainment) is quite old, since the first "educational games" for the youngest and since the "moral" fairy tales, aiming to give ethics lessons to pupils while entertaining them. The recent enormous development in videogames potentialities was revolutionary for the new possibilities given in educational games, yet one of the oldest "traps" of edutainment is always present, the risk of losing too much of the entertainment factor thus creating "dull" educational games. To create a game (either "old style" of for PC) that can confer useful educational contents and yet maintains the game qualities of involvement, interest and enjoyment is always a challenge for the developer. Not only does the integration of learning with gaming make science more fun; it also:

- Motivates students to learn
- Immerses them in the material so they learn more effectively
- Encourages them to learn from their mistakes

A wide body of research documents the pedagogical role of fun in learning [1]. For example strategic use of games in learning programs can contribute a 'flow' experience that is a characteristic of successful learning processes. Video games in fact demand far more from a player than traditional games like Monopoly. To experience the game, the player must first determine the objectives, as well as how to complete them. They must then learn the game controls and how the human-machine interface works. Beyond such skills video games are based upon the player navigating (and eventually mastering) a highly complex system with many variables. This requires a strong analytical ability, as well as flexibility and adaptability. The process of learning the boundaries, goals, and controls of a given game is often a highly demanding one that calls on many different areas of cognitive function. The obvious limit is that game-based learning cannot substitute "traditional" learning (books). But for older students it can be a precious aid for exercises, simulations and knowledge assessment, while for

younger pupils it can help to raise interest towards some topics or to help understanding which are the topics towards which the pupil shows more interest. An important and emerging aspect in videogames industry is the one related to the so-called mobile gaming, that is, gaming on personal and portable mobile technologies: mobile phones, portable gaming devices, Personal Digital Assistants (PDA's). Initially game use on this kind of devices was severely limited by technology issues (limited capabilities in computing power, screen resolution, etc.) but quickly evolving hardware allows development of more and more complex and interesting games for such devices.

II. MOBILE GAME-BASED LEARNING AND THE MGBL PROJECT

A growing body of research [2],[3] indicates that mobile technologies can be effective tools in catering for students in a digital age and there are signs of the motivating potential and possible learning gains of games played on mobile devices with young adult audiences. With the growing sophistication and affordability of mobile technologies and applications, their use as learning tools becomes increasingly viable, hence the growing interest in the field of m-learning. However, in seeking to cater for the learning needs of young audiences who in general have high relation to mobile technologies, merely trying to adapt e-learning approaches for use with mobile technologies will not be enough. Young adults in particular need m-learning opportunities that are not only cognitively accessible but that also engage them in affective learning.

mGBL [4] is a 3-year research and development project supported by the European Commission (EC) within the Information Society Technologies (IST) programme in the Sixth Framework. The project lasted from October 2005 to December 2008. Ten partner organizations from EC countries Austria, Italy, Slovenia and UK and from associate country Croatia formed the consortium, led by "evolaris next level Privatstiftung" in Graz, Austria. Target audiences were young adults aged 18–24, who regularly use mobile technologies. Project's challenge was to design exciting learning games that young adults could find fun to use. The focus was on supporting the development of decision-making skills for use in critical situations, a key area of concern in the EC. Three game models have been developed within the project, with example contents in the fields of e-commerce, e-health, career guidance and maritime distress, which were areas of strength within the consortium.

A. Pedagogical Framework

Specific theories informing the mGBL game design process included theory of the 'zone of proximal development' [5], i.e. the level of development that learners achieve when they engage in social behaviour is greater than they can achieve when working alone. We also based on experiential learning theory [6], with particular focus on learning phases: Planning, Doing, Feedback, Digesting [7]. To cater for different learner preferences, we use the Theory of Multiple Intelligences [8] in which Gardner proposes that human intelligence is a mixture of several intelligences. Finally, to support the development of decision-making skills and creative problem-solving strategies, we make use of the concept of 'single loop' and 'double loop' reflection [9] understood here as follows:

- Co-operation (following agreed rules, procedures and reflecting on efficiencies achieved thereby - 'single loop reflection').
- Collaboration (open sharing of ideas - challenging basic assumptions - 'double loop reflection').

This conceptual framework is seen to fit the benefits of mobile technologies, with their potential for making learning opportunities available anywhere, anytime and for social interaction and collaborative learning.

III. MOGABAL GAME ENGINE: MATERIALS AND METHODS

The mGBL game developed by the University of Trieste has been named "Mogabal" (extending the project acronym of MObile GAme-BAsed Learning). It is more correct to define it as a sort of game engine rather than a game, as its graphical aspects, rules, educational contents and many other elements can be fully configured and altered thus giving the potentiality for creation of widely different games and game styles. The Mogabal game engine was developed in Java ME and is then playable on any mobile device having a Java Virtual Machine compliant with Java Micro Edition (J2ME) [10], CLDC 1.0 and MIDP 1.1. Java development environments were Eclipse and NetBeans, with use of the J2ME Polish 2.0 suite of tools to manage portability of code towards different mobile devices and language localization.

IV. MOGABAL GAME ENGINE: TECHNICAL SOLUTIONS

The challenge, from the technical point of view, in programming a J2ME game in the mGBL framework, is that the objective is not to create "one" game, but the potentiality for wide range of games. Key issues in design were as follow:

- the game type and style may be highly variable;
- the game pedagogical content is highly variable;
- both game type and its pedagogical content are to be created, modified or customized by users (the teachers creating the game as support to their courses) in a relatively easy way.

In fact, the objective was to create something that could be modified with user-friendly tools without need of knowledge of Java programming, beyond some basic knowledge of steps necessary to modify the Java archive file to be installed on the mobile device. The solution, regarding the game mechanics, was found in the use of the relatively simple game concept of movement on a rectangular map, having different graphical layers: background graphics, active elements ("sprites") and possible "fog of war". The player's avatar roams the map. If "fog of war" is present, initially the map is obscured and revealed only during map exploration. The collision with one of the sprites launches an "event". Each sprite is tagged with a three characters code and a list of "1 to N" events is associated with that code. According to game construction rules, sprites can be programmed to "disappear" from the game after being collided, or they can be "permanent" and launch a random event taken from the list of events with the same code at each collision with players' avatars. The different kinds of "events" supported are:

- Quiz: a text and one or more options to choose from.
- Decision Tree: similar to Quiz, however the different possible choices have no immediate reward but instead they link to a subsequent event (which can be any event type). This allows construction of complex simulations of chains of choices or decisions.
- Conditional decision tree: is similar to Decision Tree, but some of the possible choices are available and visible to player only under particular conditions, typically a set of minimum points in one or more characteristics of the avatar (see later regarding characteristics).
- Simple: a text message that can be used as a "leaf" of a decision tree or as a simple random event.
- Multimedia: opens a multimedia resource and then links to a subsequent event. This can be used to enhance graphic aspect of the events or to insert audio/visual elements in the decision trees. Some basic file formats for images, audio and movie files are supported at the moment. It must be noted that current hardware limitations for mobile devices actually limit severely the quantity and size of the multimedia files that can be inserted in a whole game to a handful of low resolution images, but the game is ready to manage heavier duties in terms of multimedia content when mobile hardware will improve.
- Set internal variables value: the game holds an internal array. This can be used as a sort of "state machine" for complex events correlations. Up to 200 integer variables can be modified at run-time by use of this event. Putting this event at the end of a "decision tree" allows to keep memory of decisions taken by player.
- "Case of" tree: event structured just like the CASE instruction in programming languages. According to current value of one of the internal variables, different events can follow. This, in combination with the previously described event type, allows to set further events depending on previous decisions (maybe also decisions taken far earlier in game).
- Null event: game contents logic may require an "empty" event.
- Game Over: event overriding the normal game over rules.

Another important concept in the game, taken from the role-playing-games, is that the player's avatar is personalized by a set of 4 to 6 attributes, or "characteristics" [11]. The characteristics names are fully configurable: they could be, e.g., "Strength, Intelligence, Charisma, etc", if one wants to create a "typical" RPG avatar configurability. They may be "Linguistic, Logical-mathematical, Intrapersonal, Interpersonal" to follow Gardner's theory of Multiple Intelligences [8] in case of a career guidance game, where we're not interested in characterizing the physical aspects of the avatar but rather the facets of its intelligence. Players can choose between different "characters" with varied skills (Fig 1). During the game events, or better, choices made in "decision" events, can be set to

Figure 1. Game Screenshot. Choice of Player's Avatar at Begin of Game.

modify these values in terms of rewards or penalties, so player's choices will have impact on characteristics' values and also possibly on game evolution, since with use of the above described "Conditional decision tree" events some options in game will be available only under certain conditions in player's characteristics. The Java code runs as a "game engine" while everything else is contained in "resource files". The process of compilation of the application with Java creates a "jar" archive file and a "jad" descriptor file. The jar file is the final "application" that will be installed on the mobile phone, it contains all the compiled code and all the resource files included in the Java project and its contents can be opened on any PC by using one of the many archives/compression management applications capable of opening such files. In particular, the resource files contained in the jar file include all the graphic resources used for game map creation or in multimedia events and three text configuration files containing, respectively:

- game setup: defining characteristics' names, game over conditions, etc;
- graphic setup: defining how one or more game maps (one per game "stage") will be constructed from the resources containing the sprites and graphic elements needed;
- events' contents: the complete list of all the events that may happen in game.

Game setup can be edited with a simple text editor. For graphic setup, we developed for the task a Java tool called "Boardmaker" (figure 2) meant to be run on PC, presenting a graphic user interface that allows relatively easily to create game maps and the corresponding "Graphic setup" text file. Different "levels" of the game can be created, each with its own map. Obstacles and borders can be inserted on the map: visible or invisible, permanent or removable, one way or two-

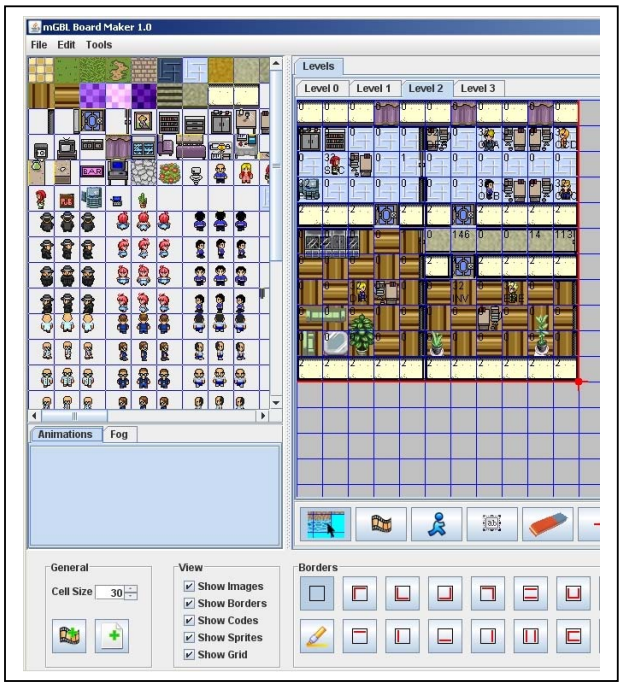

Figure 2. Boardmaker Tool Screenshot. Portion of the User Interface.

ways. Also this tools allows to put on the maps the active "sprites" that will be the core of the game. The sprites will be linked to launching events, and in this case they will be assigned the (non-unique) three digit code that will "mark" the event or events associated to avatar's collision with the sprite. Sprite may also be used to give "keys" that allow removal of some obstacles or to give or take points to player. The tool also allows creation of mobile sprites by defining zero or more finite-states machines that will be included into the configuration file. Sprites associated with such tables can be programmed for movement on the map on fixed or random trajectories. Future developments in the code could allow sprites programmed to approach the player's avatar ("enemies to avoid") or to avoid it ("preys to catch"), but this is not in the list of priorities as this could be an interesting option for creation of more "arcade-style" games but is not particularly necessary for "learning" games. For the list of events, XML is used [12]. An ad hoc XML-DTD describes all the possible events' structures, so that events following this Data Type Definition will be forcibly consistent with game code. An XML-XSL file allows automatic conversion into the text file that must be inserted within the game resources. In this way a game creator can develop his own events for customized games with no need to know game code, as long as he follows the structure defined, and the use of XML allows automatic check of correctness of the structure. Yet it was not opportune to use directly the XML file in the resources, as the XML tags create a substantial overhead. Currently on most mobile phones the size limit of the jar file for installing the application is about 500 KB. To give more space for use of multimedia resources or for longer events lists, we opted for conversion of the list of the events into a "pure text" file, with events and their fields' contents separated by vertical bar character. With all the above described elements, a wide range of game styles can be covered by the Mogabal engine. Some examples include:

- Quiz game: using "permanent" event-sprites, linked each to a long list of random quizzes regarding various topics.
- Exploration: use of fog of war and visible or hidden obstacles/borders can allow creation of labyrinth games of exploration.
- Arcade style: using mostly event-sprites with programmed or semi-random movement on a map, the game can be aimed to avoiding "baneful event" sprites while searching collision with "positive event" sprites.
- Simulation: an interactive map can "put" players avatar in a particular situation (e.g.: car accident or generally a crisis situation). Interaction with elements present may force player to try to make the right decision.
- Adventure: with more preliminary work of "plot creation", the simple "simulation" game above described can be evolved into a complex "adventure-game", with several stages (maps). By using the "Set internal variables value" event it is possible for the game to "keep memory" of players choices and have the "adventure world" react accordingly.

In all these different "game styles" contents are completely customizable. As an example: in a quiz game, as above described, the lists of quizzes could be substituted with others regarding completely different topics by simply changing the "events' content" configuration file while maintaining unchanged game logic and graphic appearance.

V. GAME PROTOTYPES

In mGBL project 3 cycles of User Trials were planned, directly related to the iterative process adopted for mGBL design and development. The second User trials took place in September 2007, finishing at end of year 2 of the project. They were conducted with "Focus Groups" methodology and primarily sought understanding and insights from members of target audiences concerning the 3 mGBL games in terms of "Fun and Playability" and "Content". Opinions from potential users (both students and teachers) were sought regarding style and design, relevance to topic, suitability, support to users in developing decision-making skills. Six different User Trials were conducted in four countries, evaluating the three game models developed within the project. The Mogabal game engine was tested in four of such trials and with different game templates, regarding the topics of e-health, e-guidance and maritime distress. Thus the above described instruments of "customization" were used to create three different games quite different in game style and completely different in educational contents.

A. e-Health: triage and first aid

The game prototype we developed for e-health contents is a good example of possible use of such mobile learning games for use in simulation of crisis situations. The game was developed, for what regards the "game-style", as a combination of Simulation, Quiz and Adventure. The "characteristics" of the avatar were of less importance in this context and this feature

Figure 3. Game Screenshot. Triage on Train Crash Site.

of the game was not used. In the first stage of game (Fig. 3), the player has to conduct a triage on the victims of a train accident. He has to approach the injured and follow procedures that allow him to discriminate and "tag" them with color codes that will help oncoming medical services to treat victims in order of urgency (Fig. 4). Only under particular circumstances in the triage also some medical assistance is given: objective is only quick decision on priority of treatments. The second stage of the game revolves instead on use of first aid techniques as the player has to assist the victims of a road accident. Touching the casualties, the system displays events and series of linked multiple answer questions forming decision trees. The game contents for first stage of game (Triage) are quite specialized and the prototype game

Figure 4. Game Screenshot. First Decisions in Triage.

was created as an example for a possible tool for knowledge assessment in combination with a triage course for emergency operators, to help in memorization of correct procedures. The second stage of the game can be an example of educational game: playing the game several times and receiving feedback (Fig. 5), after taking a decision, indicating what was the correct thing to do, means that a young player who will reach the "highest score" will probably have also fixed in memory the correct first aid procedures, paying to them more attention than if he had to memorize them in more "standard" learning ways. Reception from audience in the focus groups (which were composed by teachers and students) was quite positive. There was overall appreciation for game playability and fun by part of the students and for its potential developments and educational possibilities by part of the teachers.

B. Career guidance

Another game prototype was developed by other mGBL partners using Mogabal game engine and with our technical help for career guidance contents. This kind of contents cannot easily fit in more "simple" game styles, as can be quiz games or small simulations; in fact, in guidance, there are usually no "right" answers or "wrong" answers, every choice is possible. Moreover, more interaction is needed in order to address a "guidance simulation" and to face up the different possibilities. The most apt game style was chosen to be a combination of Role-Playing and Adventure game style. The storyboard sees a player who has to conduct a stage abroad and has to take series of decisions, sometimes easy, sometimes hard. For this kind of games creation much more preliminary work is needed, compared to e-health prototypes seen above, in order to create adequate adventure's plot and several sub-plots in storyboard development and then to "translate" them into series of game events. In preliminary storyboard development several critical situations were identified through real cases analysis, in collaboration with employment centers, young information centers and guidance counselors.

Figure 5. Game Screenshot. Feedback and Information for the Player.

Figure 6. Game Screenshot. Career: exploring the new workplace.

In the game events there is wide use of events linked to avatar's skills and of those that allow the game to "keep track" of previous player's choices and actions and react accordingly. The avatar's skills and the scoring system were based on Gardner's theory of multiple intelligence (Fig. 1). This game prototype is a good full example of how the Mogabal game engine can be used to create elaborate simulations in form of "adventure game" (Fig. 6), with exploration of possibilities an consequences of choices greatly adding to game "replayability". The game prototype was tested during other user trials conducted with "Focus Group" methodology, with positive reception.

C. *Maritime distress*

Yet another game prototype based on Mogabal game engine was developed by the mGBL partner PFRI (Faculty of Maritime Studies of Rijeka). In this case the game-style was basically that of a quiz-type game and game preparation (creation of events and graphical aspects) was relatively straightforward: a simple map where multiple answer questions appear during exploration. The player has to touch all elements in the map and must answer to all the questions. Time is limited and final score accounts for number of correct answers. This is an example of how, even with quite basic graphic and game layout and moderate need of preliminary work, the game can be used as a possible tool for knowledge assessment, with ease of customization towards different educational contents. This game prototype was tested during mGBL 2nd User Trials an was actually used as a tool for conducting intermediate course exams.

VI. FUTURE POSSIBILITIES AND DEVELOPMENTS

Although the "Mogabal" games developed for and tested during the mGBL 2nd user trials were sort of limited prototypes, acceptance received show a good potential of this kind of mobile game for learning. The game definitely doesn't rely on stunning 3D graphics and is based on relatively simple game concept, but our aim was mainly focused upon a wide possibility of customization of all aspects of game (contents, graphics, game style, rules, victory conditions) by a user who don't need to know anything about programming, by a relatively easy creation or modification of XML based resource files. Moreover the aim was that the game should be playable on most cell phones models and shouldn't be limited to more expensive smart phones. Yet the basic structure for game mechanics is such that, as "typical" mobile phones will allow more and more to use screens with better resolution and a larger memory footprint, this will allow to insert into the games larger maps to be explored and more multimedia elements like photos, videos and so on, so will become better the chance to create "interesting" games-simulations that can help in the training and in developing decision-making skills. Work will continue in creating new prototypes based on current version of the game engine and all the Mogabal source code and related material (documentation, the Boardmaker tool) was published as Open Source under European Union Public Licence (EUPL) v1.0 and is available at sourceforge.net. Hopefully this will lead to the birth of a larger community of developers, both for improvement of the code with further features and improved bug-tracking (thanks to a larger audience) and for creation of more other game examples exploring the full range of possibilities given by Mogabal game engine.

REFERENCES

[1] Fabricatore, A., 2000. Learning and Videogames: an unexploited synergy. Available from: http://www.learndev.org/dl/FabricatoreAECT2000.PDF Accessed 27th April 2008.

[2] Bellotti, F., Berta, R., De Gloria, A., Ferretti, E., Margarone, M., 2003. Designing mobile games for a challenging experience of the urban heritage. Proceedings of EURO-PAR 2003 Parallel Processing, Lecture Notes in Computer Science, Vol 2790 pp. 1129-1136. August 26-29, Klagenfurt (Austria).

[3] Ha, I., Yoon, Y., Choi, M., 2007. Determinants of adoption of mobile games under mobile broadband wireless access environment. Information & management 44 (3): 276-286.

[4] mGBL official web site at: http://www.mg-bl.com.

[5] Vygotsky, L.S., 1978. Mind in society: The development of higher psychological processes (Cambridge, MA: Harvard University Press.).

[6] Kolb, D.A., 1984. Experiential Learning, Prentice-Hall, New Jersey.

[7] Race, P., 1994. The Open Learning Handbook. London: Kogan Page.

[8] Gardner, H., 1983. Frames of Mind. Basic Books, New York.

[9] Argyris, C., 1976. Increasing Leadership Effectiveness, Wiley, New York.

[10] Lei, Y., Hui, L., 2006. Which One Should be Chosen for the Mobile Geographic Information Service Now, WAP vs. i-mode vs. J2ME. Mobile networks and applications : the journal of special issues on mobility of systems, users, data and computing 11 (6): 901-916.

[11] Kun, C. L., Byung, S. M., 2007. Enhanced Avatar Design Using Cognitive Map-Based Simulation. CyberPsychology & Behavior 10 (6): 757-766.

[12] Natchetoi, Y., Wu, H., Babin, G., Dagtas, S., 2007. EXEM : Efficient XML data exchange management for mobile applications. Information Systems Frontiers 9 (4): 439-448.