Challenges to Upgrading a Mobile Web Application

Howard Mall
ECS, Inc.
Orlando, FL
HowardMall@ecsorl.com

Teresita Sotomayor, PhD.
Army Research Lab
Orlando, FL
teresita.sotomayor@us.army.mil

ABSTRACT

The Army Research Laboratory (ARL), Human Research and Engineering Directorate, (HRED), Simulation and Training Technology Center (STTC) developed a mobile web application for conducting research in applying emerging mobile capabilities to the U.S. Army's Combat Medic curriculum. The mobile application used early Apple iOS devices and their native web browsers to deliver highly interactive training content. It consisted of a web server that delivered the application to mobile devices via Wi-Fi wireless internet connections. Students would play group trivia games or answer questions about emerging medical scenarios that included a visual synthetic casualty. A training effectiveness evaluation was conducted to assess how introducing this system into a program of instruction would improve individual learning outcomes. Lessons learned identified the need to update the application. This paper outlines the challenges and solutions that were addressed in updating the mobile application to take advantage of the strides made in mobile web capabilities. Application server technologies and web client development libraries have matured and become highly capable in terms of visual fidelity and usability. The mobile devices themselves now support multi-core Central Processing Units, Graphic Processing Units for rendering, highly optimized web browsers, and greater resolution screens that sometimes eclipse their desktop counterparts. We will describe our strategy for upgrading the mobile application to take advantage of the new technologies especially with regard to the simulation and visualization of the synthetic casualties in the scenario exercise portion of the mobile application. We will walk through our decision process and describe the lessons learned during the upgrade. We conclude with a set of guidelines for other groups taking on the task of upgrading an older mobile web application to take advantage of the myriad and ever-expanding possibilities that mobile devices afford in delivering important simulation-based curriculum to our warfighter and to education in general.

ABOUT THE AUTHORS

Howard Mall is Vice President of Engineering at Engineering and Computer Simulations, Inc. He has spent the last nine years at ECS building various kinds of training systems. He has lead efforts for the Navy to develop training solutions deployed on cell phones and hand-held computers. For the Army, he delivered the Tactical Combat Casualty Care (TC3) Simulation used by combat medics to learn triage and medical decision-making on a virtual battlefield. He has led several virtual world and game-based simulation projects for a myriad of federal agencies and commercial concerns. He currently oversees multiple engineering efforts at ECS both mainstream and on the fringe of the state-of-the-art.

Dr. Teresita Sotomayor is a chief engineer and subject matter expert in the area of severe trauma simulation at the U.S. Army Research Laboratory (ARL), Human Research and Engineering Directorate (HRED), Simulation and Training Technology Center (STTC). Her expertise in user-centric design and technology effectiveness evaluations has been instrumental in the development and transition of modeling and simulation solutions in support of medical training. She is a graduate of the University of Puerto Rico (Mayaguez Campus) with a degree in Industrial Engineering. She holds a Master of Science degree in Operations Research Stochastic Simulation from The George Washington University and a Doctorate in Modeling and Simulation from the University of Central Florida. She is a member of the Army Acquisition Corps since 2003 and has over 26 years of experience in the modeling, simulation, and training domain.

Challenges to Upgrading a Mobile Web Application

Howard Mall
ECS, Inc.
Orlando, FL
HowardMall@ecsorl.com

Teresita Sotomayor, PhD.
Army Research Lab
Orlando, FL
teresita.sotomayor@us.army.mil

INTRODUCTION

Combat Lifesavers (CLS) face multiple, complex, and often interdependent tasks, including mission planning, command and control, communications, and coordination with other team members. They are required to quickly assess a situation, develop an appropriate course of action, and act decisively to save lives. This workload is multiplied as the complexity of administering battlefield medicine continues to evolve. Developing appropriate training and support tools is therefore critical due to limited opportunities to practice such skills.

There are many challenges associated with educating, training, and maintaining the knowledge and skills necessary to successfully perform battlefield medicine. At various locations around the world, Army CLS courses are offered disparately with little or no communication or standardization. Additionally, the student's immediate reach back capability is generally limited to his or her peers who attended the course with them. With the goal of providing CLS training to every soldier, new technologies promote communication and collaboration between students and instructors across training locations and throughout the life of the learner (Butler, 2009).

Technology trends have the potential to change how people work and could become the preferred tools for some users to obtain the specialized skills needed for effective job performance. It is important to bring together a framework of cutting-edge technologies that provide learner-centered experiences from a mobile computing device. Users can perceive and interact with multiple types of content including external simulations and/or other applications while performance is tracked, assessed, communicated and maintained.

The U.S. Army Research Laboratory-Human Research and Engineering Directorate, Simulation and Training Technology Center (ARL-HRED STTC) has been evaluating the evolving technologies in mobile learning to determine the feasibility of their use in a range of medical training applications and facilities. Mobile devices are ideal platforms to support web-based training applications. They provide a capability to improve warfighter's readiness (Stone, 2009). Mobile devices are accessible and have a smaller physical footprint, facilitating the ability to train anytime, anywhere (Ally, 2009).

The STTC developed TraumaCon, a prototype Medical Mobile Application (MMA) to support the training of the CLS course material. The prototype MMA was developed to provide students with an opportunity to augment the current program of instruction through the use of challenging, game-based opportunities to review course material and sustain important skills. This paper discusses the research and enhancements made to address known limitations and lessons learned from the initial effort. The aim is to improve game play dynamics for use by Army combat medics (also known as 68W) and CLS instructors as a part of the Army's Medical Simulation Training Centers (MSTC) Program of Instruction (POI).

To this end, the upgrade of the MMA from older web technologies to new was paramount to success. The upgrade had to address three major areas: visuals, data persistence and retrieval, and the use of standards to improve future maintenance and extensibility. Simulated casualties are greatly improved for greater immersion and performance during scenario exercises. The selection of an adaptive database technology allows for a natural and transparent representation of the data. Finally, the application of well-known and well-supported standards lowers the barrier of effectiveness for programmers and maintainers of the web applications. These upgrades make MMA look better, work faster, and make it much easier to extend and maintain.

BACKGROUND

The field of medical simulation has blossomed in the last decade, pushed by past successes of flight simulators and current successes in biomedical technologies. The STTC conducts research and develops new technologies to enhance capabilities of medical simulations. STTC has forged a partnership with the US Army Medical Department Center and School, Department of Combat Medic Training (DCMT) at Fort Sam Houston, TX, to identify gaps in training and research new technologies to support simulation-based training environments for Army combat medics (Wood & Woodill, 2008). The goal is to identify innovative technologies that could potentially provide combat medics with greater opportunities to develop their technical medical and critical thinking skills.

To help fill the downtime during training or deployment, STTC has been evaluating many of the evolving technologies associated with mobile learning to determine the feasibility of their use in a range of medical training applications and facilities (Naismith, et al., 2009). TraumaCon is a MMA for training developed to investigate if mobile application technologies could improve student retention of the CLS course material. The MMA was developed as a platform independent web application and is available on various mobile platforms such as the iPod Touch, iPad, iPhone, Blackberry, and Android smart phones as well as desktop PCs. It is hypothesized that these applications could allow students to independently learn or sustain skills.

The application was initially designed to be used during five, 10-minute intervals of down time typically available between exercises and classes within each of the MSTC. The initial MMA prototype provides students with an opportunity to augment the current program of instruction through the use of a challenging, game-based opportunity to review course material and sustain important skills.

In early 2012, the STTC conducted a Training Effectiveness Evaluation (TEE) of the MMA in an effort to determine the best methods for incorporating this technology into the current Program of Instruction at the MSTCs (Sotomayor, et al., 2012). During the TEE, users had the opportunity to play the MMA games and provide feedback via questionnaire and focus group discussions. Overall, the MMA was well received and changes for improvement were captured and documented for future consideration. Specific recommendations from the TEE showed areas where additional research was needed to improve the prototype MMA.

In 2013, the STTC conducted additional research to analyze and leverage the latest technologies in mobile learning and gaming to improve and enhance the usability and potential impact of the MMA on U.S. Army medical training for combat medic and CLS students. The outcome resulted in a robust training prototype that could be tailored by authorized instructors to provide the warfighter additional opportunities to prepare to make medical decisions within an operational environment more effectively.

The architecture and core technology prototyped could potentially benefit students at the MSTCs and the DCMT. The DCMT provides initial training to combat medics on the principles of Tactical Combat Casualty Care (TC3). The MSTC program supports training for medical and non-medical personnel including Active Duty, Reserves, and National Guard, with priority given to deploying units. Both the DCMT and the MSTCs aim to prepare the warfighter for the application of medical interventions under combat conditions (Army Field Manual 4-02.4, 2013).

MOBILE MEDICAL GAME

The MMA is a web application designed to work in most common mobile browsers. Due to their penetration in the marketplace, iOS and Android (many vendors) were the targeted platforms. iOS is found on all Apple mobile devices, including iPhones, iPads, and iPods. Android is a Linux-derived platform managed by Google that is found in many vendors' devices, including Motorola, Samsung, HTC, and many others. These are the most common devices to source and with which the public (and our target demographic) has already had significant experience.

The MMA leverages new web technologies (HTML5 and advanced JavaScript) so it looks and responds like a native application. It consists of a central web server that delivers code to a mobile device connected over a wireless network. The server also provides services that supply and record data as the user interacts with the application on their mobile device.

Figure 1 outlines this architecture. The central server is Node.js, which delivers data and interactive web pages through the HyperText Transfer Protocol (HTTP). Data management is provided using MongoDB, which contains the user, leaderboard, content and live session data. When users navigate to the MMA website over their local network, the web applications make HTTP requests to retrieve database queries, images, Cascading Style Sheets (CSS), Hypertext Markup Language (HTML) pages, and JavaScript code that together constitute a web application. Different web applications are served to students and instructors based on their login credentials.

USER EXPERIENCE

The MMA is composed of two games: trivia and medical scenarios. These games are designed to be quick to learn and provide short "gap-filling" training during lane training down time. The trivia game tests knowledge, while the medical scenarios are geared more toward decision-making.

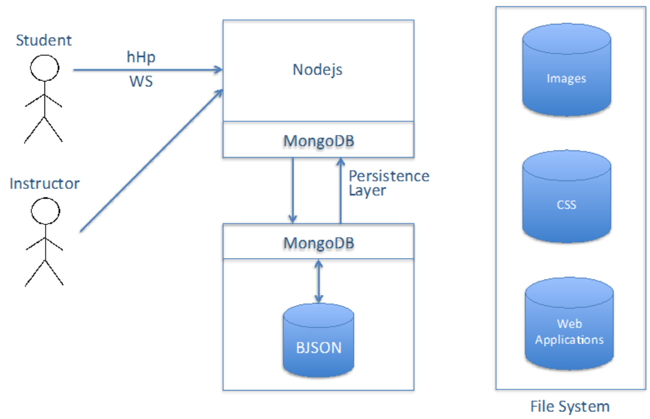

Figure 1. MMG Architecture

Trivia

The mechanics of the trivia game would be familiar to anyone who has competed in a bar trivia contest. The player competes against a cohort of their peers or can play alone. The game asks questions (related to knowledge of combat medicine) and provides four choices to select as answers. A timer counts down until the player answers. The amount of time spent to correctly answer a question contributes as a bonus to the player's score. During group play, the first player to answer correctly wins the points.

A web-based question-authoring interface was developed to allow instructors/administrators to create their own trivia questions. An instructor or administrator logs into the system from a web browser that has access to the mobile game's web server. Selecting the authoring interface, they fill out a form that is submitted to a hosted web service that stores that question in the relational database. The question includes the text of the question, the right answer, the wrong answers, and the topic or topics that categorize the question.

Scenarios

The medical scenarios are less trivial and are meant to be played independently. The player selects a particular subject such as "Bleeding Control" and is presented with a synthetic casualty rendered in a window on their screen. The player would then be asked a series of questions about what decisions to make. The interface is similar to the trivia game in that the decisions are multiple-choice. Rather than randomly selected questions, each question follows the progression of the casualty through the scenario. A casualty's medical state progresses at each step, and the player is asked to "make the call". Correct answers have a positive effect on the casualty's medical state while incorrect answers are detrimental.

The medical scenarios are also stored within the database and can be authored by an instructor or administrator in a similar fashion to the trivia questions. However, the next question or next state is defined within the interface so that the questions progress from an emergent medical situation through to the ultimate outcome based on the player's decisions. The medical state of a synthetic casualty along with its associated art is specified within the same interface to define what to display as the result of the answers the player will select.

TRAINING EFFECTIVENESS EVALUATION

The initial MMA was developed as a platform-independent web application and targeted mobile platforms such as the iPod Touch, iPad, iPhone, Blackberry, and Android smart phones as well as desktop PCs. It was hypothesized that by using the mobile application in conjunction with programs of instruction to complement classroom instruction, skills labs, and lane training, students could become more familiar with fundamental skills during the time they are not directly involved with instructors. The potential benefit of the MMA is that it allows the student to independently learn or sustain skills, and is designed to be used during five to ten minute intervals of downtime typically available between exercises and classes within the Army's MSTCs.

A usability evaluation was conducted in September 2010 at Fort Indiantown Gap with the participation of eight Subject Matter Experts (SMEs) (Nielsen, 2009). During the evaluation, the SMEs had an opportunity to play the MMA games and provide feedback via questionnaire and focus group discussions. Overall, the MMA was well-received and changes for improvement were captured and documented for future consideration. As a follow on, a user evaluation was conducted at the Fort Drum Medical Simulation Training Center from November 2011 to February 2012 (Sotomayor, et al., 2012). The objective of the evaluation was to assess the training effectiveness of the MMA using the Kirkpatrick model (Table 1). The main purpose was to investigate the use of the prototype mobile application as a complementary training aid to supplement the current CLS POI while also providing additional opportunities for sustainment training of CLS skills. A total of 140 soldiers volunteered across five separate test events split equally between the control group (70 subjects) and the MMA group (70 subjects).

Table 1. Kirkpatrick Model of Training Evaluation.

Level	Evaluation	Focus	Instruments
1. Reaction	Affective and attitudinal response	Self-report measures	Surveys, questionnaires, focus groups
2. Learning	Increase in knowledge and capability	Learning outcomes	Pre/post knowledge tests, interviews, assessments
3. Behavior	Changes in on-the-job performance	On-the-job performance measures	Self-assessments, on-the-job performance ratings
4. Results	Effect on the overall organization	Long-term productivity measures	Organizational performance reports, utility analysis, quality assurance reports

The first two levels were evaluated and data was collected on student reactions to the prototype MMA technology (Level 1) and on increase in knowledge test scores after receiving the training (Level 2). Results showed that the MMA group reacted positively to the technology. In terms of knowledge acquisition, the results were not statistically significant. It is important to note that the MMA was designed primarily for sustainment training, such that soldiers could utilize it outside the classroom to practice and improve their skills. It would be instructive to study the effects of the MMA on retention and skills transfer. Parallels between knowledge gain and battlefield performance cannot meaningfully be drawn without collection of Kirkpatrick Level 3 (longitudinal) data, which was beyond the scope of the effort but is a consideration for future work.

Specific recommendations from the usability study and the training effectiveness evaluation showed areas where additional research was needed to improve the prototype MMA. These areas included:

- Content management
- Database architecture
- Reporting and data management
- Account management
- Combat Medic (68W) curriculum content (e.g. scenarios and questions/answers)
- Graphical rendering across mobile platforms beyond Apple iPod/iPhone
- Validated CLS content

The intent of continuing research was to analyze and leverage the latest technologies in mobile learning and gaming to improve and enhance the usability and potential impact of the MMA on U.S. Army medical training for combat medic and CLS students. The outcome should result in a robust training prototype that could be tailored by authorized instructors to provide the warfighter additional opportunities to prepare to make medical decisions within an operational environment more effectively. Addressing the identified areas of improvement will make the prototype suitable to support a longitudinal study. The longitudinal study is planned to evaluate up to Level 3 of the Kirkpatrick model.

TECHNOLOGY IMPROVEMENTS

Node.js' offers a straightforward platform for architecting scalable network programs. Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js uses an event-driven, non-blocking input/output model making it simple and efficient, which is ideal for real-time applications that are data-intensive and run across distributed machines or mobile devices such as the iPod Touch.

One of the many advantages of Node.js is it handles many client connections simultaneously. Node.js notifies the operating system when a new connection is made, and then sleeps until a new connection is requested, at which time a callback is executed. This approach differs from more popular models where operating system threads are employed concurrently. This is especially important because thread-based networking is not the most efficient or easy to use. Node.js provides memory efficiency despite incorporating systems with intensive loads, allocating two-megabyte thread stacks for each connection. Additionally, Node.js does not deadlock a process because no function in Node.js directly performs input/output. This means the process never blocks, thus eliminating the possibility of a deadlock. Because of this attribute, Node.js proves to be a favorable option for less experienced software engineers, allowing them to design systems more efficiently.

Node.js is similar in design to and influenced by systems like Ruby's Event Machine or Python's Twisted. Node.js expands upon the event model—it presents the event loop as a language construct instead of as a library. In other systems there is always a blocking call to start the event-loop. Typically one defines behavior through callbacks at the beginning of a script and at the end starts a server through a blocking call like EventMachine::run(). In Node.js there is no such start-the-event-loop call. Node.js simply enters the event loop after executing the input script. Node.js exits the event loop when there are no more callbacks to perform. This behavior is like JavaScript running in a web browser—the event loop is hidden from the user.

HTTP is a first-class protocol in Node.js. Node.js' HTTP library has grown out of the author's experiences developing and working with web servers. For example, streaming data through most web frameworks is impossible. Node.js attempts to correct these problems in its HTTP parser and API. Coupled with Node.js' purely event-based infrastructure, it makes a good foundation for web libraries or frameworks.

For the MMA, we selected Node.js for the following reasons:

- Larger community supports Node.js when compared to other frameworks
- Modern and effective technology that supports the requirements
- Client/server is all the same language, alleviating some of the burden to the programmer
- Transparent code for new programmers to read and understand
- Easy to deploy and distribute
- Can use readily available and common development tools for editing and version control
- Supported by and used by major corporations and highly scaled web applications

In an attempt to enhance MMA's mobile technology, a considerable effort was made to improve the way in which graphics were implemented in the Casualty Challenge game. With this iteration of the MMA, a 3D casualty model was used to render the necessary poses and then the finished asset was assembled in Photoshop. In essence, this technique created a digitized "paper doll". This was a significant improvement from the previous approach that started with photographs and then led to a labor-intensive process of hand painting the rest of the layers for each casualty. The new approach also lends itself to support the state of the art high definition resolutions (1920x1080) versus the lower resolutions screens available during the previous iteration of the initial MMA (640x480). The ability to support higher resolution increases the number of mobile devices capable of running the MMA. The increase in resolution also provides higher fidelity casualty assets.

Additionally, the graphics rendered on the client side of the application versus the previous approach that rendered the graphics on the server. The new casualty states are assembled and rendered without the latency of server side communication and processing. These changes have already proven to provide better performance with a higher resolution display of the graphics across more platforms.

Upgrading the Server

At the conclusion of previous research and development for the MMA, a recommendation was made to replace the database architecture with a standard database such as SQL so that instructors could more easily edit the current games or add additional questions and scenarios. At that time, instructors from Fort Drum suggested that they would like to have the flexibility to add up to 500 questions. As a result, part of the research for this current effort involved determining the best type of database to use for implementation.

MongoDB is an open-source document database that addresses performance, availability, and scaling. Additionally, MongoDB is the leading NoSQL database, providing data storage and retrieval not architected around tabular relations used in relational databases. This approach enhances the simplicity of design, horizontal scaling, and finer control over availability. MongoDB is written in C++ and provides:

- Document-oriented storage
- Full index/key support
- Replication & high availability
- Horizontal scaling through automatic distribution of data records across multiple computers
- Robust document-based queries
- Flexible data processing and compression

A record in MongoDB is a document, which is a data structure composed of field and value pairs. MongoDB documents are similar to JavaScript Object Notation (JSON) objects. The values of fields may include other documents, arrays, and arrays of documents. Figure 2 depicts a sample MongoDB data object.

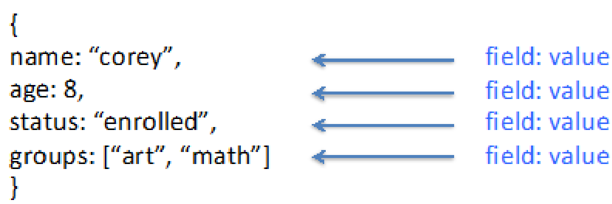

Figure 2. Sample MongeDB Data Object

A MongoDB document is similar to native data types in programming languages. MongoDB provides high performance data persistence. Specifically, embedded data models are supported, thus minimizing the transactional activity with the database system. Database indexes support faster queries and can include keys from embedded documents and arrays. For the MMA, we selected MongoDB because of the following reasons:

- Document database allows schema to grow and extend for multiple data persistent needs thus making it a good platform for research
- Mongo has good integration with Node.js
- Used extensively by large corporations and major web applications, proving its stability
- Strong developer community assures that problems get fixed and people are available to answer questions
- Developed and supported by a major corporation assures that it will be a viable tool for many years to come
- Very well-suited for mining data
- Well-suited for recording complex scenario data required by this application

Upgrading the Client

HTML5 has been leveraged extensively in providing a very responsive web application to the user. HTML5 is a host of new technologies being included in all modern browsers that provides device native capabilities. JQuery is also used in the MMA client application. JQuery is a Javascript library that makes it much easier to develop web applications and is becoming a *de facto* standard. This should make it much easier to upgrade the application in the future.

The MMA visuals were upgraded in this iteration. The styling of the application was given an overhaul making it cleaner and more visually appealing. The images for the medical scenarios were increased in resolution and fidelity. These new images were captured from high-resolution 3-D reference models, allowing easier upgrades in the future as the performance of mobile devices increases. The client also caches and downloads all of the image data in the background allowing for better performance as the images animate and switch states.

Expanding Reporting Capability

Figure 3. Reports Screen

Since MongoDB was specifically selected for its data mining and faster queries, providing expanded reporting capability was straightforward. Now that the database architecture supports a more robust data set, these reports will be even more valuable to instructors. Three reports are available for instructors to view and then download as a comma delimited (CSV) file, which can be viewed and reduced using Microsoft Excel or a similar application. Once instructors have downloaded and accessed these output files that record student performance, etc., they are free to incorporate them into any desired reports or presentations. Figure 4 depicts the "Reports" screen and the reports that are available to the Instructor as well as the "Leaderboard History" report.

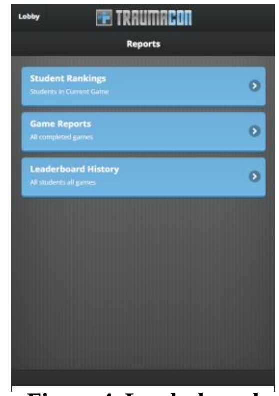

Figure 4. Leaderboard Report Screen

Redefining Student Account Management and Instruction

Once the database was re-architected, it not only made the implementation of the Instructor reports straightforward, it also made the credentialing and account management capability straightforward to implement. The individual account management capability is available to both students and instructors. Instructors also have access to account management of student accounts as depicted by the "Manage Users" screen depicted in Figure 4. An instructor can import a CSV file to enter multiple users into the game as depicted in Figure 5. Instructors can also create, edit, and delete one student account record at a time.

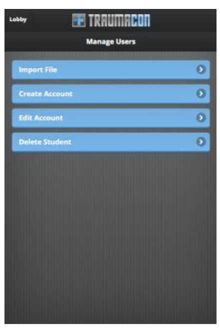

Figure 5. Manage Users Screen

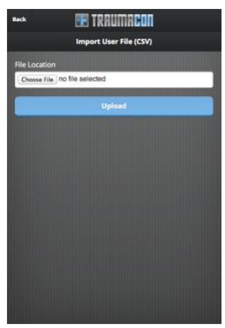

Figure 6. Import User File

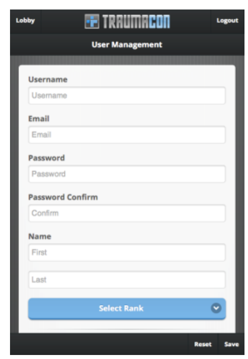

Figure 7. User Management Screen

The User Management Screen allows a student or instructor enter in the pertinent credentials to create or edit an account. Figure 6 depicts the "User Management" screen.

Adding New Administrative Tools

The Administrative Tools provided with the MMA include the User Account Management tools, as well as, the Game Editing tools provided to create new, edit existing, and delete existing Casualty and Knowledge Trivia games. Figure 7 depicts the Game Editor screen. Because of the structure of the Knowledge Trivia Game, an Import Game capability is also provided, which allows an instructor to upload a CSV file that contains questions and answers.

Figure 8 shows the "Select Lesson" screen. Lessons are composed of multiple games. The instructor selects which lesson from which to choose games to add, edit, or delete.

Figure 8. Game Editor Screen

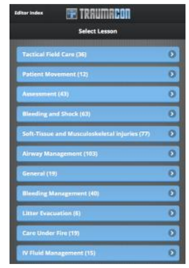

Figure 9. Select Lesson Screen

RECOMMENDATIONS AND GUIDELINES

The upgrades made to the MMA make it look better, work faster, and make it much easier to extend and maintain. When planning to upgrade a web application it is important to evaluate the technologies utilized at the various layers of the application: client, server, and persistence. In this case, these different layers all required an upgrade to use some modern web technologies that are emerging standards to increase application performance and programmer productivity. The process of selecting these technologies was based on the major criteria of extensibility, ease of development, and transparency.

Extensibility describes the ease with which new features or capabilities can be added to a software program. For a web application, the selection of the server technology stack becomes very important. The application server must be easy to install for deployment and development, utilize a well-known computer language, and have a structure that is easy to understand. Additionally, it should be well supported by the community, have good documentation, and an expectation of longevity.

Node.js is a newer web application server that meets these criteria. Web services are written in JavaScript which is the same language in which the client code is written lowering the learning curve for programmers. Debugging the server code in the client application is also made much easier. It is easy to install, manage, and run with many facilities for auto-updating and the inclusion of libraries and plug-ins for supporting one's functional service needs. It is popular and used in many new and large web applications. This has assured a large community with active development, longevity, many tutorials, and up-to-date documentation.

Ease of development was an important part of the MMA client upgrade. The client uses CSS for all of its styling, which is a well-established internet standard that allows the presentation of the application to be separated from the content. Digital artists are well-versed in CSS and can apply a technique called responsive design that (with the proper consideration) allows the look of the application to adapt to the screen size and platform on which the application is running.

Additionally, JQuery is the JavaScript library that makes it much easier to develop web client applications. The code becomes more readable and compact. It has also become a *de facto* standard for web development assuring that many programmers will be familiar with its users. It has a large set of community written plug-ins and customizations that can be leveraged to deliver some very advanced features with a modicum of coding. There are many tutorials available on the web to help a developer solve problems and increase their knowledge and technique using the library. It also has an integrated user interface library that supports many useful widgets and advanced effects.

Transparency helps developers to understand how the system performs and is especially important in the data persistence layer so that the data and the model are clear. MongoDB is a document database that delivers data in the JavaScript Object Notation (JSON) and is well integrated with Node.js. JSON is very easy to access and manipulate in JavaScript. It is also human-readable. For a programmer, it is very easy to populate a MongoDB database and to see what data is being stored. This transparency allows the programmer to quickly see the relationships between the data stored in the persistence layer, how it is being delivered as a service through Node.js, and how it is being utilized in the JQuery/JavaScript web client application.

Architecturally, the layers of the application are distinct and modular. The selection of technologies support the JavaScript language standard across the many layers. These technologies have also proven popular with well-established communities. All of these factors support extensibility, development ease, and transparency. The MMA as a platform is now (after this upgrade) easier to deploy, maintain, extend, and understand. When upgrading your own web application you should consider these factors to create a system that is efficient and will have longevity.

FUTURE WORK

The majority of the future work that remains for the MMA is conducting a thorough Training Effectiveness Evaluation and Usability Study to determine if it is ready for use in the classroom. It would also be very useful to devote time and resources to solely developing content for the MMA for different clinicians such as nurses, physician assistants, medical technicians, etc. Content development for the scenario based game, the Casualty Challenge, will require an investment in new artwork, which depending on how far in the future this would take place, might also merit investigation of new methodologies for improving the asset rendering, resolution, and realism.

CONCLUSIONS

The results of this effort have created an opportunity to explore new learning strategies for mobile computing platforms and have provided insight into how a mobile platform can continue to support medical training. The MMA is an ideal tool for filling gaps of downtime with an opportunity for learning. It also allows instructors to continue updating content and challenge students with a competitive forum for learning within and outside of the classroom.

Today's Army medical professionals face unprecedented challenges that are constantly evolving. Resources and capabilities are heavily taxed at all levels. Meeting these challenges requires a range of innovative solutions, to include a transformation in the way we prepare for deployment and combat trauma. Through the enhancements of the MMA, the research team demonstrated how innovative learning technologies could be blended and distributed to meet AMEDD's training and education requirements through instructional content that combines elements of persistent 3-D graphics, immersive learning environments, simulation-based training, and collaborative, online knowledge repositories.

The results of this research have increased our understanding of how to unify these separate components into medical training facilities to promote increased sustainment of perishable skills to a generation of trainees that learn through different instructional modalities not yet developed. This research will increase the U.S. Army's and the other services understanding of the impact technology has on learning and performance. Overall, this project has made an important and valuable contribution to the training and education possibilities for the AMEDD community. Perhaps more importantly, it represents a foundation upon which additional enhancements can be built and offers opportunities for extensive studies of training effectiveness involving a full spectrum of blended learning solutions.

REFERENCES

Ally, Mohamed (Ed.) (2009). *Mobile Learning: Transforming the Delivery of Education and Training*. Edmonton, Canada: Athabasca University Press.

Anttila, A. & Jung, Y. (2006). Discovering design drivers for mobile media solutions. *Proceedings from the Computer-Human Interaction Conference* (CHI 06), Montreal.

Army Field Manual 4-02.4 – Medical Platoon Leader's Handbook. Role of the Combat Medic. Retrieved from www.globalsecurity.org/military/library/policy/army/fm/4-02-4/appc.pdf.

Butler, F. (2009). Tactical Combat Casualty Care. Defense Health Board Upload.

Cooper, L. J. (2005). Customizing online information: How learning style, content delivery and pre-instructional strategy affect recall and satisfaction (Doctoral dissertation, Ohio State University).

Felder, R. M., & Soloman, B. A. (2000). Learning styles and strategies. Retrieved from http://www.engr.ncsu.edu/learningstyles/ilsweb.html.

Gadd, Robert (2010) mLearning content types – Overview and intro. mLearning Trends, January 19.

Kolb, D. A. (1984). *Experiential Learning: Experience as the Source of Learning and Development* .(Vol. 1). Englewood Cliffs, NJ: Prentice-Hall.

Naismith, L., Lonsdale, P., Vavoula, G., & Sharples, M. (2004). *Literature Review in Mobile Technologies and Learning*. Bristol: NESTA FutureLab.

Nielsen , J. (1993). *Usability Engineering*. Boston: Academic Press.

Paay, Jeni (2008). From ethnography to interface design. In Lumsden, Joanna (Ed.), Handbook of Research on User Interface Design and Evaluation for Mobile Technology.

Sotomayor, T. M., Salva, A. M., & York, B. W. (2012). Measuring the training effectiveness of combat lifesaver simulation training systems. *Proceedings from the Interservice/Industry Training, Simulation, and Education Conference (IITSEC 2012), Orlando, FL*

Stone, G. (2009). Hand-to-hand training: Detailed, interactive mission training, execution and review can be shared on mobile devices. Retrieved from http://www.tsjonline.com/story.php?F=4069327.

Wood, A. & Woodill, G. (2008). *Learning Technologies for Healthcare Education and Training*. Sunnyvale, CA: Brandon Hall Research.

"RDBMS dominate the database market, but NoSQL systems are catching up". DB-Engines.com. 21 Nov 2013. Retrieved 24 Nov 2013.

Harris, Amber (April 1, 2012). "The Birth of Node: Where Did it Come From? Creator Ryan Dahl Shares the History". Devops Angle. Retrieved 26 October 2013.

Fontaine, Timothy (January 16, 2014). "Node.js and the Road Ahead". Retrieved 21 January 2014.

"Here's why you should be happy that Microsoft is embracing Node.js". The Guardian. November 9, 2011. Retrieved May 10, 2012.

"Why Walmart is using Node.js". VentureBeat. January 24, 2012. Retrieved May 10, 2012.