Mobile Learning Applications Audit

Paul POCATILU[1], Adrian POCOVNICU[2]
[1]Academy of Economic Studies, Bucharest, Romania
[2]ISA Consulting, New York, USA
ppaul@ase.ro, pocovnicu@gmail.com

While mobile learning (m-learning) applications have proven their value in educational activities, there is a need to measure their reliability, accessibility and further more their trustworthiness. Mobile devices are far more vulnerable then classic computers and present inconvenient interfaces due to their size, hardware limitations and their mobile connectivity. Mobile learning applications should be audited to determine if they should be trusted or not, while multimedia contents like automatic speech recognition (ASR) can improve their accessibility. This article will start with a brief introduction on m-learning applications, then it will present the audit process for m-learning applications, it will iterate their specific security threats, it will define the ASR process, and it will elaborate how ASR can enhance accessibility of these types of applications.
Keywords: *IT Audit, Software Testing, Penetration Testing, Mobile Applications, Multimedia, Automatic Speech Recognition*

1 Introduction

According to International Telecommunication Union (ITU) March, 2009 press release there were an estimated 4.1 billion mobile cellular subscriptions worldwide as of end of 2008 [7], figure 1.

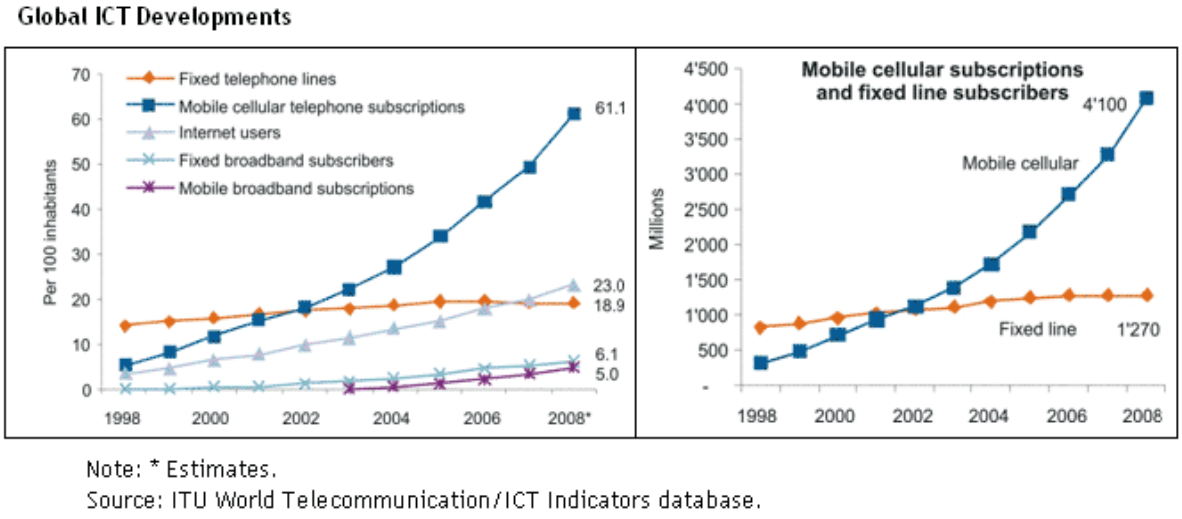

Fig. 1. Global ICT Developments [7]

A statistic from multiple sources of information, published on Wikipedia [8], shows that in US there were over 276 million mobile phones in use as of June 2009 while in Romania there were almost 23 million mobile phones in use as of March 2008. It is interesting to see that while Romania ranks 32nd in the world by number of mobile phones in use, it ranks 13th by the number of mobile phones in use per 100 people with 108.5 mobile phones in use per 100 people, figure 2.

The numerous mobile phones are not used only to make calls but they are also used as digital assistants, as communication tools (other than calls), as educational tools and as source for entertainment.

Rank	Country or region	Number of mobile phones	Population	% of population	Last updated
—	World	4,100,000,000	6,797,100,000	60.6	Dec 2008[1]
1	China	747,380,000	1,335,330,000	56.0	Dec 2009[2]
2	India	545,048,136	1,175,430,000	46.37	Jan 2010[3]
3	United States	276,610,580	308,505,000	89.0	June. 2009[4] [5]
4	Russia	208,330,000	141,927,297	146.8	Jan. 2010[6] [7]
5	Brazil	175,600,000	191,480,630	91.70	Jan. 2010 [8]
...	...	...	...	...	...
32	Romania	22,800,000	21,438,000	108.5	March 2008

Fig. 2. List of countries by number of mobile phones in use [8]

While the number of mobile phones has dramatically increased over the past years, researches show that the market for mobile learning applications is also growing at a fast pace.

As mobile devices have become very small yet the purposes they serve have greatly diversified, the use of speech-centric interfaces has become more frequent. Deploying computational intensive automatic speech recognition (ASR) software on mobile devices that are still characterized as having limited computational power, limited memory and limited battery life has to overcome these challenges. The proceedings in this direction were to implement efficient ASR on embedded platforms primarily by using fixed-point arithmetic and low computational complexity and low memory footprint optimized algorithms. Another approach to overcome mobile devices limitations is the use of distributed speech recognition (DSR), which means that the process of automatic speech recognition is broken into two components: a client component which extracts the features and a serve component that does the heavy processing this alleviating this way the burden on the mobile devices.

The use of DSR introduces a new parameter in the equation, that of the network availability and performance.

The traditional user interfaces like keypad, stylus and small screens are sometime inconvenient and speech-centric interface seems like a good solution to overcome these usability challenges.

There are several aspects that influence the audit of m-learning applications that will be discussed in the following paragraphs.

M-learning applications are developed as standalone applications or based on distributed applications. Distributed applications can be Web based or based on a dedicated architecture.

The provided course content can be free or paid. User registration could optional or mandatory when the content is free. If the course content is not free then the user will need to register for an account.

The course content can be online of offline. The online content is accessed through a network connection. The offline content is installed or downloaded on the mobile device.

Regarding the course assessments, the m-learning application could be:

- Test based, scores recorded for academic situations
- Test based, scores recorded for personal information
- Without partial or final tests

The security audit of every m-learning system is very important, especially when confidential information about students is recorded. The following unwanted issues can show up in every organization that has m-learning and/or e-learning implementations:

- The unauthorized access to course content.
- Unauthorized access to student's personal data (marks, scores)
- Unauthorized access to exam tests

These potential security issues can be prevented and discovered by a specialized audit.

In order to be proactive, the testing process has to include the audit requirements for mobile learning applications. To be confident that an application is secure, it has to be tested using specific methods and techniques. Secure software is good quality software. This paper highlights the main characteristics of the mobile applications testing process, how security testing is done and specific audit challenges.

2 IT Audit

The IT audit focuses on availability, confidentiality and integrity [1], [11]. IT audit will highlight

potential problems and risks regarding its processes and structure availability, data confidentiality against unauthorized access and data integrity.

Fig. 3. The audit process stages

The audit can be external or internal. It has to be a specialized department in order to have an internal audit. The auditor independence is very important for an objective audit process [2]. It doesn't have to be involved in the organization activities that are audited.

Figure 3 depicts the general steps of the audit process.

The audit process needs to be planned before it starts. The planning deliverables depend on the audit requirements. During the audit process the analysis is done using different means, from SWOT analysis to running elaborated tools. After the on-site audit process ends, reports are written and sent to the organization. After the reports were received, detected problems and risks are to be fixed or mitigated.

In today's world, the security of software applications is one of the most important aspects that have to be considered. Almost all organizations are connected to a computer network. Auditing the security of any organization is crucial especially when they manage sensitive data. All security problems need to be solved in order to be confident in these organizations.

During the IT audit dedicated tools are used. These tools are both standardized and custom, depending on the organization specific.

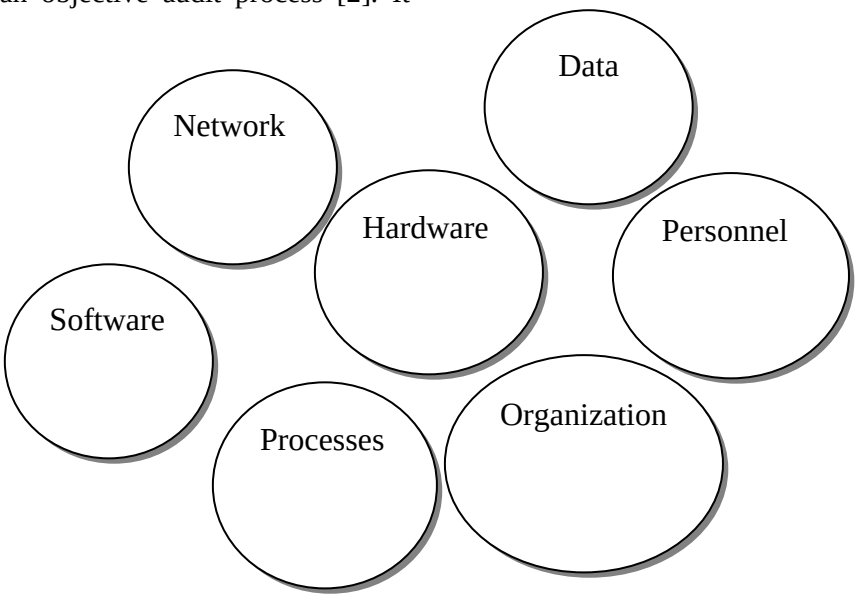

Fig. 4. The audit process areas

The IT audit is done on one or more of areas depicted in figure 4. For m-learning systems, these areas can be identified as:

- M-learning software, Web servers,
- Mobile devices, servers, personal computers
- Educational process, m-learning procedures
- Network infrastructure used for m-learning process and
- M-learning organization
- The personnel involved in the m-learning process: students, content providers, administrators
- M-learning content, course content, student records databases

Every organization that provide educational services need to demonstrate that its processes comply with all required standards in the field in order to be trusted. This is necessary for certification organizations, academic organization and continuous learning providers. The virtual educa-

tional organization can be seen as collaborative systems [6].

3 Enhanced accessibility using automatic speech recognition

Speech is the most natural human form of communication. Extending this form of communication from man-man to man-machine has been a major breakthrough. It is called automatic speech recognition (ASR), it saves time and it is also more convenient than the traditional ways of data input.

There are two major paradigms in ASR, the first one which is the most widely used is the phonetic content of the speech signal, which varies from language to language, but there are no more than 30 different phonemes without some variations, such as accentuation, duration, and the concatenation [10]. The second one is the use of syllables instead of phonemes as an alternative for development of the ASR systems, because in general, the natural way to understand the language by the human brain is to store and recognize syllables not phonemes. [10]

An outline of the anatomy of the human speech production system is shown in figure 5. It consists of the lungs, larynx, vocal tract cavity, nasal cavity, teeth, lips, and the connecting tubes. The combined voice production mechanism produces the variety of vibrations and spectral–temporal compositions that form different speech sounds.

The act of production of speech begins with exhaling (inhaled) air from the lungs. The information is first modulated onto the passing air by the manner and the frequency of closing and opening of the glottal folds. The output of the glottal folds is the excitation signal to the vocal tract which is further shaped by the resonances of the vocal tract and the effects of the nasal cavities and the teeth and lips.

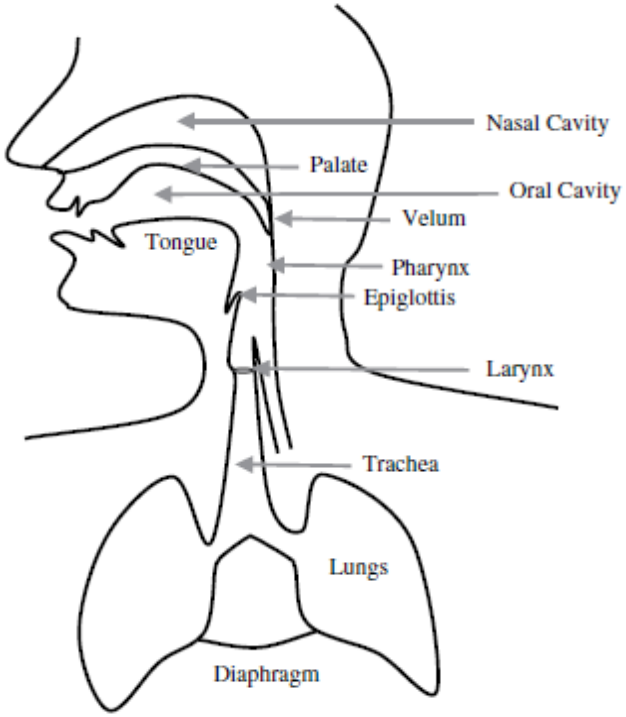

Fig. 5. Illustration of anatomy of speech production [9]

The tongue, lips, jaw and velum are the primary articulators; movement of these articulators appear to account for most of the variations in the vocal tract shape associated with making different sounds when speaking. [9]

The automatic speech recognition is a complex pattern recognition task in which the speech signal is analyzed to extract the most significant features, once the speech signal has been digitalized with a sampling rate between 8 and 16 kHz.

The number of applications that are using mobile devices in educational environments is growing. Mobile devices limitations such as screen size and small keypad, if any, are impairing the accessibility of the applications.

Speech input interfaces play an important role in improving the usability of mobile learning applications. The following are examples of ASR uses in mobile learning applications:

- Speech to text
- Speech to speech translation
- Voice recognition

- IVR

Speech to text (conversion process from human speech to digital text) applied to m-learning applications improves searching and indexing while reducing storage. M-learning applications can be used to record real time speeches and then convert these to text. Digital text requires less storage space than an audio file. Also searching and indexing processes have less complexity and offer better performance when they are executed against text files than when they are executed against audio files. This enables students to search through their courses for keywords and to retrieve those courses that contain the specific keywords.

Speech to speech translation is the process that captures the speech input from one source, digitizes it, transforms the information and outputs it as a speech with the same meaning but in a different language. While speech to speech translation algorithms are very complex and heavy resource consumers, mobile device are not yet suitable for such complex processing. The required architecture to accomplish speech to text translation is client/server architecture. The mobile device act as an input interface capturing the speech coding it in digital content. This digital content is then transferred over the network (3G, WiFi, etc) to the server, which is a powerful machine capable to execute high complexity translation algorithms. Once translation is done it is sent back to the mobile device which decodes it and plays it as audio to the receiving user. Such client server mobile application allows foreign students to listen to the courses in their native language.

Interactive voice response (IVR) is a technology that uses voice input or keypad input to browse through a hierarchical structure of options. Such technology incorporated in m-learning allows students to easily locate the desired content by simply answering a set of questions.

Voice recognition technology enhances m-learning applications security while making it more appealing to the users. It uses the features from the user's voice for authentication. When user authentication is required prior to access specific educational content, the user needs to recite a passphrase to the mobile device which captures it, codes it, extracts specific features, compares them against an existing voice features database and correctly authenticates the user providing him access to the content. A similar approach can be done to verify the identity of the student that is taking a test remotely. In this case test answers have to be input verbally and the m-learning application will validate both the answer and the identity of the user that recited the answer.

4 Mobile Applications Testing

Many software issues can be avoided earlier from the development stages through a well planned management. One important area is software testing, the process of identifying errors in software. The mobile application testing is application type dependent. If the mobile application is standalone, the testing process is similar to the classical one, having in mind the limited resources of the mobile devices and the unexpected events while using the application, like incoming phone calls, text or multimedia messages etc.

It is very important that code review will take place during the software development process. These reviews could uncover potential security issues earlier in the software life cycle (e.g. buffer overflows).

The testing process is different for distributed mobile applications. Testing web based mobile applications breaks down in the following components [5]:

- functional testing
- compatibility testing
- content testing
- performance testing
- load testing
- security testing
- web server testing
- application server testing
- and database testing.

Testing web based mobile applications, requires testing of the server applications and testing of the content that is send to the mobile device.

The automation of software testing consists of a series of processes, activities and tools brought together in order to execute the software under test and to record the result of the tests. Tools for automated software testing of mobile applications are various, and they can be used in different areas of testing. There are many tools to assist software testing: capture/playback tools, tools for automated execution of tests, coverage analyzers, test case generators, logical and complexity analyzers, code instrumentation tools, defect tracking tools and test management tools.

Testing mobile application is a very complex process, depending on the application type. For web based applications, not only the specific response sent to the mobile device has to be tested, but the whole application from the server side. That includes the testing of JSP, ASP.NET and

PHP scripts.
There are many aspects that influence the testing process and make it difficult. When a failure occurs, it can caused by one of the following:

- insufficient memory;
- hardware incompatibilities (screen characteristics, input methods etc.);
- poor user interface design;
- network problems;
- bugs in application;
- the web server is poorly configured;
- the database management system is not working properly;
- the database scripts contains errors.

For each type of device and platform there are emulators, so until the application is running on the device, it is implemented and tested on those emulators. Using mobile devices emulators enables solving any applications compatibility issues before the application is deployed to the mobile devices. Before using the application on the mobile device, it will be tested using an emulator and the environment where is developed.
When using various integrated development environments (IDE) that allow: developing, running and debugging mobile applications, the testing process becomes easier. Every programming language used in developing mobile applications has specific characteristics that can influence the testing process.

5 Security issues

Computer applications security is one of the most important problems to be solved in the context of information society. Computer applications receive, process, store and transmit a wealth of information and data, using a variety of transport technologies and interface (memory areas, sockets, environment variables, files, communication protocols etc.). Identifying weaknesses and theirs correction during the application development is still done through a testing process specific, well planned and organized, leading to confidence that the application has a high security level.
Based on attack aims, these are classified in attacks for [4]:

- obtaining access to confidential information;
- data modifications;
- breaking the functionality of the system.

Computer applications security testing is the identification of existing security holes in applications submitted for testing. Testing the applications security is done both in the development phase and after the application was released.
Application security testing is done both at the source code if available and desired, and running executable applications [3].

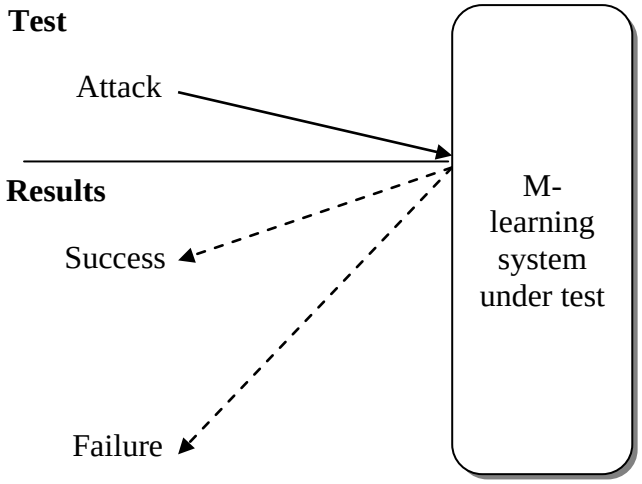

Fig. 6. IT systems security test results

Figure 6 depicts a system under security test. If the attack used in a test is successfully completed (the application allows the access, although it wouldn't have) it is considered that the application is uncertain at such attacks. Otherwise, the application is considered safe, taking account of the completion of the attack. The success of an attack and is considered a failure of system security and, conversely, a failure of the attack is considered a success in terms of system security test.
Among the errors introduced by programmers in source code, intentional or not, stand the following categories:

- logical errors;
- errors related to boundary checking (areas of memory, arrays, strings etc.) ;

- memory allocation errors;
- debugging and testing code note removed before the application release ;
- use of backdoors to access the application after the release ;

Testing of computer applications designed to identify security issues:

- unauthorized access to a system or application;
- unauthorized data modification;
- viewing and accessing confidential information;
- obtain privileges;
- prevent other users to use application.

Another aspect that has to be considered is the logging subsystem of an application. In case of an attempted unauthorized access, monitoring and logging will help to eliminate the problems and identify further unauthorized source.

Testing security computer applications such strategies will be white-box and black-box. In addition to these strategies, with their specific methods are recommended and source code analysis and browsing.

Penetration testing leads to the identification of vulnerabilities of systems such as vulnerable services, ineffective security policies, configuration issues. It is used for different aspects of networks and different network topologies, firewalls, operating systems and applications.

External penetration testing focuses on identifying weaknesses in networks from outside attacks. It is performed with or without the knowledge about the network configuration. The *internal penetration testing* is trying to identify network weaknesses based on the internal configuration using various tests.

Penetration testing is done manually or automatically. The following tools are used in penetration testing:

- DNS query (*nslookup*);
- Network scanners for address and open ports;
- Passwords breakers;
- Network packet filtering;
- navigation and remote debugging programs;
- other network tools (*whois*, *tracert*, *ping*);

For better results, penetration testing of mobile learning systems based on Web applications is recommended to be done both manually and automatically, each type of testing detecting specific types of problems. Also, it is recommended to use penetration testing both internal and external.

Denial of service attack (DoS) is another cause of distributed systems failure. The m-learning systems based on distributed architecture have to be audited in order to identify such kind of risks.

6 Conclusions

The audit of the m-learning applications makes use of both general and specialized methods. An audit process on mobile learning applications helps identify security vulnerabilities, reliability issues and accessibility limitation. The audit process is both application specific and architecture specific, meaning that different mobile learning application types or architectures: stand alone or client/server require different auditing approaches, methods and technologies Many of the security issues can be solved during the application development through a planned verification and validation phase. While is impossible to foreseen all possible scenarios, it is always better to prevent first than fix later.

The diverse set of mobile devices set of future allows for non-traditional ways to improve both security and usability of mobile learning application by using automatic speech recognition and voice recognition.

Acknowledgements

This paper presents some results of the research project IDEI 2637 (920/2009): *Project management methodologies for the development of mobile applications in the educational system*, financed within the framework of IDEI research program by the National University Research Council – Ministry of Education, Research and Innovation from Romania.

References

[1] R. Cascarino, *Auditor's guide to information systems auditing*, John Wiley & Sons, Inc., 2007

[2] G. Hinson, "Frequently Avoided Questions about Computer Auditing", [Online]. Available: http://www.isect.com/html/ca_faq.html

[3] P. Pocatilu "Software Security Testing," *Informatica Economica*, vol. 9, no. 4/2009, pp. 78-82

[4] L. Gong, G. Ellison and Mary Dageforde, *Inside Java™ 2 Platform Security: Architecture, API Design, and Implementation, Second Edition*, Addison Wesley, 2003

[5] P. Pocatilu and M. Popa, "Internet Applications Testing," *Proceedings of 6th International Conference on Economic Informatics*, IE'2003, Digital Economy, Bucharest, 8-11 May 2003, pp 1028-1032

[6] C. Ciurea, "A Metrics Approach for Collaborative Systems," *Informatica Economica*,

vol. 13, no. 2/2009, pp. 41-49
[7] *** New ITU ICT Development Index compares 154 countries, [Online], Available: http://www.itu.int/newsroom/press_releases/2009/07.html
[8] *** List of countries by number of mobile phones in use, [Online], Available: http://en.wikipedia.org/wiki/List_of_countries_by_number_of_mobile_phones_in_use#cite_note-1
[9] S. V. Vaseghi, *Multimedia Signal Processing*, John Wiley & Sons, Inc., 2007
[10] B. Verma, M. Blumenstein, *Pattern Recognition Technologies and Applications: Recent Advances*, Information Science Reference, 2007
[11] C. Davis and M. Schillerand and K. Wheeler, *IT Auditing: Using Controls to Protect Information Assets*, McGraw-Hill, 2007

Paul POCATILU graduated the Faculty of Cybernetics, Statistics and Economic Informatics in 1998. He achieved the PhD in Economics in 2003 with thesis on Software Testing Cost Assessment Models. He has published as author and co-author over 45 articles in journals and over 40 articles on national and international conferences. He is author and co-author of 10 books, (Software Testing Costs, and Object Oriented Software Testing are two of them). He is associate professor in the Department of Economic Informatics of the Academy of Economic Studies, Bucharest. He teaches courses, seminars and laboratories on Mobile Devices Programming, Economic Informatics, Computer Programming and Project Management to graduate and postgraduate students. His current research areas are software testing, software quality, project management, mobile application development and mobile learning.

Adrian POCOVNICU is a PhD Candidate at Academy of Economic Studies. He graduated the Faculty of Cybernetics, Statistics and Economic Informatics and holds a masters degree in Computer Science for Management since 2004. His main research areas are: Multimedia Databases, Information Retrieval, Multimedia Compression Algorithms and Data Integration. He is a Data Integration Consultant for ISA Consulting, USA providing expertise to leading US companies.