

## Audit Process during Projects for Development of New Mobile IT Applications

Marius POPA

Department of Computer Science in Economics  
Academy of Economic Studies, Bucharest, Romania  
marius.popa@ase.ro

*This paper presents characteristics of the computer audit process during software development life cycle focused on specific aspects of the mobile IT applications. There are highlighted specific features of the distributed informatics systems implemented in wireless environments as hardware components, wireless technologies, classes of wireless systems, specialized software for mobile IT applications, quality characteristics of the mobile IT applications, software development models and their specific stages and issues aspects of the computer audit during software development life cycle of the distributed informatics systems customized on mobile IT applications. In the computer audit process, tasks of the computer auditors and what controls they must implement are also presented.*

**Keywords:** *Audit Process, Mobile It Applications, Software Development Life Cycle, Project Management*

### 1 Characteristics of Mobile IT Applications

In [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [15] and [16], the most important characteristics of the distributed informatics systems were presented. Also, classes of distributed informatics systems were highlighted in those papers.

A mobile system has the main characteristic to implement communications among its components through radio frequencies. In addition, the locations of the components of such a system can be changed whenever needed.

For example, such hardware components are [18]:

- *Wireless Local Area Networks (LAN)* – are used by business computer users because they has quality characteristics like flexibility and reliability;
- *Cellular phones and pagers* – are used to connect portable and mobile applications;
- *Global Positioning System (GPS)* – allows to find a location on earth;
- *Cordless computer peripherals* – are used to link peripherals to a computer via wireless;
- *Satellite television* – allows to view television channels in almost any location;
- *Cordless telephone sets* – are not cell

phone; they are limited-range devices;

- *Home-entertainment-system control boxes* – TV channel control, VCR/DVD control, hi-fi sound systems and so forth;
- *Remote garage-door openers* – operates at radio frequencies to open or close doors;
- *Two-way radios* – includes communications through radio services like amateur radio service;
- *Baby monitors* – have limited range and they are simplified radio transmitter/receiver.

Some of the above components are operated and controlled using specific applications for each kind of device. This kind of software is called mobile application. In addition, if the mobile application is developed for a device class from IT&C systems then the software is called mobile IT application.

Wireless system evolved rapidly. As it is presented in [18], some specialized wireless technologies are:

- *Global System for Mobile Communication (GSM)* – wireless telephone standards in Europe;
- *General Packet Radio Service (GPRS)* – provides continuous Internet connection for mobile phone and computer users;
- *Enhanced Data GSM Environment (EDGE)* – is a faster GSM wireless ser-

vice;

- *Universal Mobile Telecommunications System (UMTS)* – is a packet-based system containing a set of services to mobile phone and computer users;
- *Wireless Application Protocol (WAP)* – is a set of communication protocols to standardize the way that wireless devices are used for Internet access;
- *i-Mode* – build for Web browsing.

Nowadays, it is working to built new technologies or to improve the existent ones to offer improved ways to Internet content, especially for multimedia content and business, using mobile devices, especially cell phones and smart phones.

3G or 3<sup>rd</sup> Generation is a generation of standards used by mobile phones and mobile telecommunications [19]. The standards branded as 3G are UMTS system, presented above, and CDMA2000 system.

CDMA2000 system uses channel access Code Division Multiple Access – CDMA for various radio communication technologies. It is used especially in North America and South Korea, and UMTS system is used in Europe, Japan and China.

3G devices have improved capabilities to execute applications as [19]:

- *Mobile TV* – a TV channel is redirected to subscriber's phone;
- *Video on demand* – a movie is sent to subscriber's phone;
- *Video conferencing* – subscribers can see and talk to each other;
- *Tele-medicine* – a medical provider helps an isolated subscriber;
- *Location-based services* – a provider sends localized services like weather, traffic conditions and so forth to the phone.

In addition to devices classes, the mobile applications are conditioned by wireless technologies that support communications among the components of a distributed system.

Depending on location and radio technology, the wireless systems are divided into [18]:

- *Fixed wireless* – connection to the Internet is made through specialized modems; wireless devices and systems are con-

nected to modems and they are used in homes and offices;

- *Mobile wireless* – wireless devices and systems work on moving vehicles;
- *Portable wireless* – wireless devices and systems are autonomous and battery-powered; they are used outside the homes, offices or vehicles;
- *IR wireless* – using the infrared radiation to transmit data; they are used for limited-range communications and control systems.

Mobile applications are designed to run on mobile devices like cell phones, smartphones, personal digital assistants (PDA) and enterprise digital assistants (EDA). Mobile devices like smartphones are controlled by operating systems called mobile operating systems. Mobile operating systems are simpler than computer operating systems and deal with Internet and local connectivity, mobile multimedia formats and different input methods.

Some examples of mobile operating systems are [20]:

- *Symbian OS* – one of Nokia's mobile operating systems for mobile devices and smartphones; it contains libraries, user interface, frameworks and reference implementations of common tools;
- *Research in Motion BlackBerry OS* – was designed for business and is focused on easy operation;
- *Apple iPhone OS* – has a GUI and multi-touch interface and does not run on third-party hardware;
- *Windows Mobile* – is similar to desktop version of Windows operating system; it has a non-optimize interface for touch input and third-party software development is allowed;
- *Google Android* – is an open-source operating system; it aims mobile phones, tablet computers and netbooks; it is based upon the Linux kernel;
- *Linux* – is a free and open source software and it can be installed on a wide variety of computer hardware from mobile phones to supercomputers;
- *Palm webOS* – is a proprietary mobile

operating system running on Linux kernel; it uses on devices with touchscreens;

- *Samsung bada* – is in development;
- *MeeGo* – combines the best of Moblin and the best of Maemo to create an open source operating system;
- *Maemo* – is based on Debian operating system; it is developed by Nokia for smartphones and Internet Tablets.

Mobile operating systems are a class of software that runs on mobile devices. Also, they control the code execution of software dedicated to a type of mobile device. Some of them permit the running of third-party software, but other operating systems allow only proprietary software. More information regarding the mobile operating systems are presented in [17].

The mobile software running under the control of operating software represents the second class of IT applications dedicated to mobile device. The mobile software intends to enhance the feature of a mobile device with additional functionalities and utilities.

In [2], there are presented some reasons and characteristics to invoke mobile applications as it follows:

- *Flexibility* – access to information anytime, anywhere; the Internet resources are accessed from anyplace when the user need them;
- *Collaboration* – users can offer feedback to the information provider; in such way, mobile applications contributes to increase the quality of Internet resources in a shorter time than using the fixed devices and IT&C technologies;
- *Motivation* – multimedia resources offer entertaining experience, so users are tempted to use them again;
- *Accessibility* – resource are available anywhere where there is access to the mobile technologies;
- *Portability* – mobile devices can be moved very easy from a location to another one.

The above characteristics of the mobile IT applications must be consider in addition to the quality characteristics specified in ISO 9126 to customize the quality evaluation of

this kind of software.

In [14], some important characteristics of mobile applications are depicted as it follows:

- *Startup time* – is an important characteristic of mobile applications because users tend to use mobile application frequently and for short periods of time; mobile applications have short sessions, so a quick startup time is imperative;
- *Responsiveness* – users have to receive prompt responses from mobile device after they perform an action on a device; it is possible that repeated actions to have implemented different actions in mobile application;
- *Focused purpose* – mobile application must execute specific tasks with a minimum number of actions performed by users; mobile application must have special interface controls to execute specific tasks; for example, buttons for access the contacts database or other data can be implemented even the code increases; the extra code provides focused purpose and enables users to achieve their goals quicker;
- *Customized interactions with off-device information sources* – some mobile IT applications interacts with information sources; to build a great mobile application, it must consider the information sources returning information in an appropriate way for mobile application; also, the information sources must be configured so that they provide information for a mobile user;
- *Consistency of experience* – mobile application must be develop in accordance to the mobile device features; the experience offered by mobile device to user must be consistent; this means that user do not need to adapt permanently to features outside the pattern provide by mobile device and its mobile operating system.

There are some differences between mobile IT application and desktop application. Some of these differences are presented in [14]:

- *Long-duration and short-duration activities* – desktop computers are used for long

sessions; laptop computers combines the characteristics of desktop computers and mobile devices: they can be used as desktop computer connected to a fixed communication infrastructure, but they also are used as portable devices, having hardware to connect to a wireless communication infrastructure; yet, the processing capabilities are over the mobile devices and very similar to desktop computers; the same activities are on short term when mobile devices, less laptop computers, are used;

- *Exploratory and focused activities* – the hardware resources of the desktop or laptop computers are very large to explore information; also, other computer peripherals offer the tools to navigate easily among information displayed on screen; the communication infrastructure permits to download large amount of information on local computer; also, application installed on desktop computers provide tools to access resources in different optimized ways; instead, activities on mobile devices are very focused; users know exactly what information they need and access them directly through mobile application available on their mobile devices; these applications cannot have all capabilities of the similar applications installed on a desktop computer, but offer the main functionalities to the user because the hardware constraints are very major.

Fitting in a pocket is the main advantage of the mobile devices. They can be carried anywhere and this aspect offers flexibility to the user. Some form-factor considerations are highlighted in [14]:

- *Use in crowded and noisy spaces* – the mobile device has not to disturb people around it; this consideration aims the possibility to use speech input; otherwise, the mobile application cannot be functional when all people try to use speech input in a small room;
- *Single or two-handed operation* – generally, a mobile device is operated by one hand; as exception, PDAs and smart-

phones require to be held in one hand and operated by another; laptops require two hands for efficient usage; mobile IT application must meet this requirement imposed by usage way of the mobile device; also, it must take into consideration the capabilities of the hardware: a laptop with touch screen is more similar to a PDA; mobile IT application must profit by all features of the hardware and it must adapt to them;

- *Lack of power cords or other cables* – mobile devices are effectiveness if they are not linked to a power source or other device for long periods of time; mobile application must run without linking the mobile device to another hardware or power source.

There are mobile devices that can be used as fixed devices or in a non-wireless system. Also, a complex distributed informatics system includes all kinds of devices and it must accord different technologies to offer services for which it was designed.

Development of mobile application must consider the type of mobile device, what mobile operating system is installed on it, hardware resources and their features. Minimizing the mobile application code is a very important optimization criterion, but extra code is necessary to provide a high quality mobile application.

## 2 Development Process of New Mobile IT Applications

Development of new mobile applications is a process of making applications for mobile devices like cell phones, personal digital assistants or enterprise digital assistants. There two types of mobile applications [21]:

- *Pre-installed applications* – are installed on mobile devices during manufacturing process;
- *User-installed applications* – are downloaded and installed by users from application stores or other distribution platforms.

There are multiple platforms to support devices and they are of two types depending on number of device manufacturers: multiple

devices from more manufacturers and multiple devices from one manufacturer [21].

The platforms for multiples devices from more manufacturers are [21]:

- *Java Platform, Micro Edition* – produces portable applications; provide simple applications on feature phones;
- *Symbian Platform* – is a real-time, multi-tasking operating system with optimization features on resource constraints of a mobile device;
- *Android* – is a Linux-based platform; the programming language used to develop applications is Java;
- *.NET Compact Framework* – is used to develop applications on Pocket PC and Windows Mobile;
- *Qt (framework)* – is a cross-platform application development platform;
- *BREW* – is used to develop application on CDMA devices;
- *Windows Mobile* – is an operating system for smart phones and mobile devices;
- *Palm OS* – is a mobile operating system for PDAs;
- *Flash Lite* – is used for devices that support Flash Lite player;
- *Microbrowser based* – is designed to display Web content optimized for mobile devices.

The platforms for multiples devices from one manufacturer are [21]:

- *BlackBerry* – is used to access Internet content;
- *Apple iOS* – is available only for one operating system; mobile applications must be hosted in the application store of the manufacturer.

Generally, the reasons to begin development of a new informatics system are [24]:

- Business needs;
- New legal requirements;
- Maintaining or enhance profitability;
- Improving efficiency;
- Reducing costs.

Developing new IT systems implies large investments what makes one of the most important areas of investment in an organization.

To start a mobile application development

process, it must be established the following issues:

- Target mobile devices;
- Functionality;
- Usability;
- Software security;
- Software performance.

Depending on the source of the IT application, an organization has the following options [24]:

- In-house IT development;
- Software package from a software house;
- IT development by a software house;
- Partnership for IT development by a software house and in-house IT team;
- End-user development.

There are two main headings considered during a development process of a new IT application [24]:

- Project management;
- Life cycle of the system development.

The project management aims solution delivering on time, within budget and expected level of quality.

There are a variety of project management methodologies such as PRINCE, PMW or MS-Project. A project management methodology includes typically the following components [24]:

- *Organisation* – involvement of the senior management to support the project; it is established a framework for direction and management of the project;
- *Planning* – activities are detailed at appropriate levels, resource requirements are identified and risks are properly evaluated; it is the key component for successful project management;
- *Control* – viability of the project is continuously monitored; controls aim financial controls (budget), time controls (milestones), subjective controls (quality assurance).

Life cycle of the system development must meet the following requirements for the new IT solution [24]:

- Controlled manner of the solution development;
- Documented solution;
- Solution maintainability;

- Efficient and secure manner of the solution development;
- Meeting the user's requirements.

There are different development processes and methodologies to build new IT systems. Each of them has particular stages to reach to requirements of new IT systems.

Typical stages of a life cycle of new IT system development are [24]:

- *Project initiation/Feasibility study* – formal definition of the project and determination of the costs;
- *Analysis and user requirements* – confirming project objectives and scope, identifying and classifying the required data, identifying and prioritizing business requirements;
- *Design* – completion logical and technical design of the system;
- *Build* – programming and testing the system; testing consists of unit testing, link testing, system testing and user acceptance testing;
- *Implementation* – moving the IT system into production with maximum benefits and minimum disruption of the client's activity;
- *Post implementation review* – reviewing the development process and learning lessons for the future.

There are some modern methods to develop IT applications as [24]:

- *Computer Aided Software Engineering – CASE* – software helping managers, system analysts, programmers and users to automate the design and implementation of IT applications; CASE offers to IT applications characteristics like high-quality, defect-free and maintainability;
- *Object Orientation* – software is considered a collection of objects containing data structures and routines implementing algorithms; this method provides software reusability, acceleration of the development process on integration/assembly operations;
- *Prototyping* – an iterative and interactive development process to produce an acceptable solution by the user; then, it passes to the full production; prototyping

provides error correction, improve solution on feedback, tracking the emerging technologies, clear development goals;

- *Rapid Application Development – RAD* – end to end development life cycle; it is based on the premise that the most part of the IT applications are obtained in less time than the rest of the software solutions; RAD offers early functionality, quick response to business needs, risk control, documentation of the end product, early conversations between client and developer, limited exposure to the changes, short development time.

Software development process is implemented on software development models like [22]:

- *Waterfall model* – developers must follow some phases; a new phase is started after ending the previous one; the stages of the model are (Figure 1):

1. Requirements specification;
2. Software design;
3. Implementation;
4. Testing;
5. Deployment;
6. Maintenance.

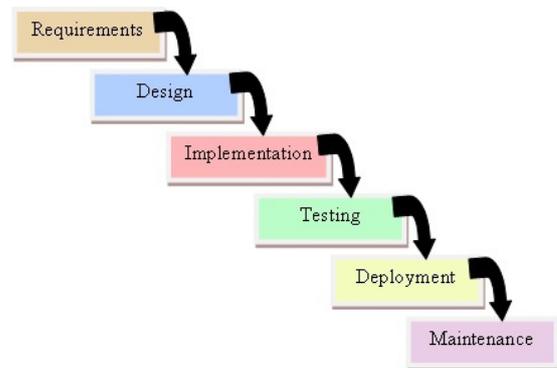


Fig. 1. Waterfall model

This model is inflexible and discourages revisiting and revising the prior phase once it is complete;

- *Spiral model* – combines key aspects of the waterfall model and rapid prototyping methodologies; the key characteristic is risk management; the steps in spiral model are generalized as follows (Figure 2):

  1. Defining the system requirements in a very detailed manner;

2. Creating a preliminary design for the new system; it is the most important stage of the model; all possible risks in project development are identified and resolved;
3. Building a first prototype from preliminary design; the prototype is an approximation of the final product;
4. A second prototype is evolved in ac-

- cordance to the following procedure:
- a. Evaluating the strengths, weakness and risks of the first prototype;
  - b. Defining the requirements for the second prototype;
  - c. Planning and designing the second prototype;
  - d. The second prototype is built and tested.

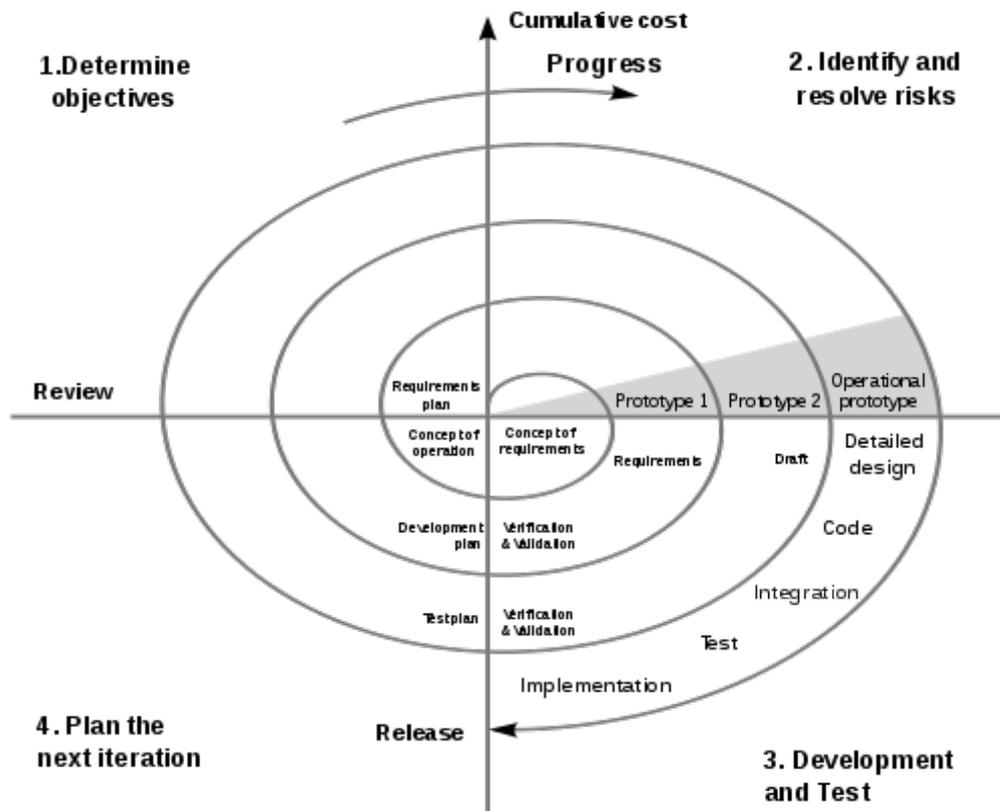
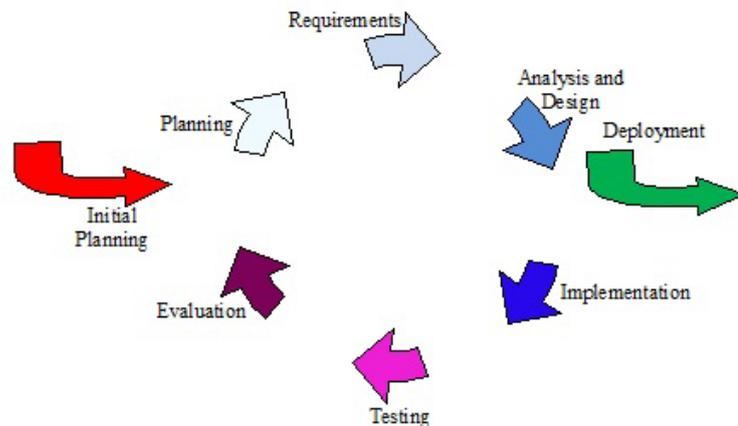


Fig. 2. Spiral model [1]

As application, this model is used for game development and large projects;

- *Iterative and incremental development* – is a cyclic software development process to reduce the weaknesses of the waterfall model; initial planning starts the development process and the deployment ends the process with cyclic interactions in between;
- *Agile development* – uses iterative development as basis; as primary control mechanism it is used the feedback; it is a group of software development methodologies; as agile software development methods, the following are included:
  - Agile Modeling;
  - Agile Unified Process – AUP;
  - Dynamic Systems Development Method – DSDM;
  - Essential Unified Process – EssUP;
  - Extreme Programming – XP;
  - Future Driven Development – FDD;
  - Open Unified Process – OpenUP;
  - Scrum;
  - *Extreme Programming – XP* – there is a very short time of software development process. The steps are: writing automated test, coding by a pair of programmers completed when all tests pass, design and architecture come after coding, deployment of the incomplete system and restarting of writing test for the most important

part of the system.



**Fig. 3.** Iterative and incremental development

- *Scrum* – is used for management of software development projects. It contains sets of practices and predefined roles. As applications, financial products, Internet products and medical products can be developed using Scrum.

As IT applications, the mobile IT applications can be developed on software development methodologies presented above. Generally, due to their characteristics, the mobile IT applications are developed in shorter time than complex desktop application. This kind of applications has a focused purpose and this characteristic provides a smaller requirements package than the desktop applications. In addition, the hardware constraints make a shorter development process of a mobile IT application. It is possible the mobile IT application development process to focus on some stages of the software development methodologies due to focused purpose and resource constraints.

If it considers that mobile applications are part of a very complex distributed informatics system then they must be developed in accordance with decisions taken in beginning of the system development process. This component of the distributed informatics system must be integrated in the larger system. The methodology chosen to develop this component must consider the development process of the larger system. All stages of the mobile component development must be correlated to what happens in larger system.

As portable devices, having wireless capabilities, the laptops can have role as mobile device. IT applications installed on such devices can be very complex, similar or identical to desktop software application. In this case, software development process respects the specifications from development methodologies for desktop IT applications.

### 3 Issues of Audit Process during Life Cycle of Mobile IT Applications

Definitions of the computer audit for IT systems and other characteristics of this relatively newness concept are presented in [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [15] and [16].

Computer audit of the distributed informatics systems is developed in permanent changing environment. New improved IT&C technologies challenge the computer auditors to be up-to-date with new evolutions in this field. The computer audit area must be a flexible and dynamic structure to face the new technology challenges.

Computer audit for IT systems is a part of the overall audit activity and requires a high-level of technical knowledge to implement the controls. Due to this requirement, it is necessary very specialized staff to implement audit process for distributed informatics system.

Computer audit adds value to the organization if it is correlated to business audit. Only controls in technical part of an organization

are not enough to increase the profitability. It is very important to use conclusions of IT audit processes to be implemented in new IT systems to make them cheaper and easier to develop. Also, it is very important to involve computer auditors in development process of new IT systems because their experience in this field adds improved and better features to the new IT product.

According to [24], the scope of the computer audit aims:

- Systems under development;
- Live applications;
- IT infrastructure;
- Audit automation.

Development process of a new IT system is organized and carried out under project management principles. Due to this management issue, to control how the activities are implemented requires a computer auditor should be involved.

Computer audit in project management of the IT system development aims [24]:

- Project team, responsibilities, senior management involvement;
- Plans and assessment regarding their achievement;
- Monitoring of project progress to implement actions when variations from the plans occur.

Early involvement of the computer audit in system development life cycle is reasoned by needs to develop new system in a secure and controlled manner. The most important phases of life cycle in which computer audit is carried out are design and testing. The following activities must be carried out in IT audit during the system development [24]:

- Understanding and confirming the user requirements;
- Covering requirements by IT system;
- Appropriate development approach and methodology, consideration of risks and inclusion of controls;
- Adequate documentation on how the system works.

Generally, computer audit for IT systems under development is focused on the following areas [24]:

- How a new IT application is developed;

- Security and control within an IT application.

Security and control features are implemented in the new IT application to meet the identified risks. According to the [24], there are two headings of applications controls:

- *Application specific controls* – controls within IT application:

- *Input controls* – aims operations on data like data handling, data quality, data checks, recording sensitive or critical activities to produce evidences of processing; they are combination of manual and automated controls; examples of input controls: authorizing the input document by adequate person, input validation through different checks and so forth;

- *Processing controls* – aims complete, accurate and timely processing of the transactions; examples of processing controls: controlled, tested and authorized changes in transactions, checking key calculations and so forth;

- *Output controls* – aims the quality characteristics which application output must meet, like: completeness, accuracy and availability; the application output can have a physical form or an electronic form; examples of output controls: secure holding and distribution of the output, logical access over the output, an output schedule is available and so forth;

- *Procedures* – aims procedure preparing to define controls and checks; examples of procedures: procedure manuals containing adequate controls and checks, procedure testing before system deployment and so forth;

- *General IT infrastructure controls* – supplements application specific controls; IT application runs in an IT environment which influences the level of security and control features of the application; there are various IT infrastructure controls depending on hardware nature where the IT application will be deployed; this kind of controls includes: physical security, contingency planning, logical access control,

program change control, operating system, telecommunications, storage media, databases, cryptography, computer operations. It is easier and cheaper to implement security and control features at the design time of the new IT system. During the development process of the new IT application, computer audit provides security and control features on documentation and discussions with relevant personnel.

The quality of the security and control features implemented in development process of the new application also depends on security and control mechanisms of the IT infrastructure where the IT application will run. Two situations are identified:

- *Existing hardware installation* – the quality of the security and control mechanism is evaluated on computer audit process of the existing infrastructure; implementation of the security and control features of the new application should consider the audit report of the IT infrastructure to ensure high level of quality for these features;
- *New hardware installation* – security and control features must be considered in more details because the new hardware infrastructure and its configuration is unknown for application developers; the quality of the features can be evaluated after the hardware installation together with the impact of the hardware configuration on security and control features of the IT application.

The audit automation is a component of the computer audit consisting of use of software products in audit process. These software products are increasingly user friendly, having the following major advantages:

- Increasing of the productivity for the audit process;
- Transferring the audit process carrying out from technical specialized auditors to business auditors; as result, the role of the computer auditor is to assist with specialized expertise the business auditor.

The ways in which the software products are used in audit processes are [24]:

- Audit tool;
- Administration tool.

As audit tools, audit automation software uses computer assisted audit techniques CAATS in audit process to increase efficiency and effectiveness. There are two categories of CAATS:

- *Data reviewing* – extraction, examination and manipulation of data by software programs; on this kind of CAATS, accuracy and integrity of data are reviewed, providing strength or weakness to the control;
- *Control reviewing* – verifying the existence of the controls and the functioning way of these ones.

In literature, the following classes of CAATS are referred [24]:

- *Interrogation software* – interrogates system or application data; it is the most used software as CAATS; the most known interrogation software: Interactive Data Extraction and Analysis, Auditor Command Language, Structured Query Language, Statistical Analysis Software, CA-Examine and Enterprise Security Manager for various IT platforms;
- *Embedded data collection* – is similar to interrogations software; the program logic is embedded within live applications;
- *Parallel simulation* – compares a rewritten system and processing data to the live system;
- *Other CAATS* – aim software such as code comparison.

As administration tools, audit automation software aims internal audit function. They simplify the internal audit, but also provide other functionalities for the organization. These products are used for planning, risk assessment, time recording, electronic working papers and presentations.

For an audit process of the software development life cycle, the most relevant information criteria are [23]:

- Primary: effectiveness and efficiency;
- Secondary: confidentiality, integrity, availability, compliance and reliability.

The seven information criteria are defined in COBIT to be met by application systems. During an audit process of an application system during software development life cycle, the computer auditor should evaluate

whether the application meets all these information criteria. Also, the computer auditor evaluates how the software development life cycle stages contribute to achieve the above criteria.

In accordance to audit standards of software development life cycle, the following types of reviews must carry out [23]:

- Studying the proposed model of software development life cycle to evaluate its appropriateness and risks to make recommendations for risk mitigation; it is made for pre-implementation review;
- Reviewing the relevant stages of software development life cycle as they are happening to highlight risks or other issues; it is made when the reviewing is a parallel or concurrent one;
- Studying the relevant stages of software development life cycle after their completion to provide corrections or learn lessons for the future; it is made in post-implementation reviews.

As aspects to be reviewed during software development life cycle, the ISACA standard provides the following [23]:

- Project charter (including the project plan, deliverables and their schedules) and business case (highlighting costs and benefits) for the application system;
- Project structure including any working groups, steering groups, and the related roles and responsibilities;
- The formal project management methodology adopted, if any, and the related process of creating the customized design of processes;
- The development or application development methodology, such as waterfall development, prototyping, rapid application development, CASE, and object-oriented development, and the associated tools chosen for the application system;
- Contracts with suppliers for purchased application systems;
- Contracts with suppliers for outsourced services, such as customization and/or development;
- Control processes within the software development life cycle model: particularly

reviews, validations, approvals and sign-offs for the software development life cycle stages under review;

- Structure of the deliverables for the software development life cycle stages under review;
- Minutes of relevant meetings, such as working group and steering group meetings;
- Actual deliverables, as well as the audit trails of their reviews and sign-off;
- Project reporting, progress tracking (efforts, time and cost) and escalation;
- Resource management;
- Ongoing risk management;
- Quality management/assurance;
- Change management;
- Performance and problem management including service level agreements;
- Configuration management;
- Data conversion/migration;
- Documentation relating to in-project reviews including testing;
- In-project and supplier communications;
- Reviews, if any, of earlier software development life cycle stages of the application system;
- Earlier software development life cycle reviews, if any, of similar applications;
- Relevant legal, regulatory and policy aspects to be complied with, if any.

Desktop IT applications installed on portable devices like laptop computers with wireless capabilities become mobile IT application if it has wireless functionalities implemented during development process. This kind of applications is developed in the same way like desktop applications following the same methodologies. The difference consists of use of wireless capabilities and completion with wireless functionalities.

A computer audit during life cycle of this kind of application is the same with the audit for desktop application.

Other kind of mobile IT application is given by applications installed on portable devices with large resource constraints than the laptop computers. A computer audit of mobile IT application seems to be much easier because the code is much smaller and it has less

implemented functionalities than a classical IT application.

Considering the focused purpose of a mobile IT application, a computer audit process can increase as complexity. The mobile IT application must be built in a secure manner for the focused purpose and without compromises regarding security features. More and more mobile IT applications are developed for various portable devices having different hardware features. Also, these applications aim critical fields like banking and military purposes. Therefore, a computer audit of the mobile IT applications reviews thoroughly the application features, especially the security implementations.

#### 4 Conclusions

Computer audit process is a very important tool for senior management to adjust the organization capabilities. Together with business audit, this process adds value to the organization and makes it more profitable.

As component of the computer audit of the IT systems, the mobile IT application audit has specific features due to the environment in which these applications run and the hardware constraints of the portable devices. Anyway, there are classical desktop applications that can be portable to devices having wireless capabilities. Thus, this application becomes a mobile application in a wireless environment.

#### Acknowledgements

This work was supported by CNCSIS – UEFISCSU, project number PNII – IDEI 1838/2008, contract no. 923/2009 and the title *Implementation of the Quantitative Methods in Distributed Informatics System Audit*, financed by The National University Research Council – Ministry of Education, Research, Youth and Sports from Romania.

#### References

- [1] B. Boehm, “A Spiral Model of Software Development and Enhancement,” *ACM SIGSOFT Software Engineering Notes*, ACM, 11(4), 1986, pp. 14 – 24
- [2] C. Boja and L. Bătăgan, “Software Characteristics of M-Learning Applications,” *Proceedings of the 10th WSEAS Int. Conference on Mathematics and Computers in Business and Economics*, Prague, Czech Republic, 23 – 25 March 2009, pp. 88 – 93
- [3] S. Capisizu, G. Noșca and M. Popa, “Informatics Audit,” *The 37th International Scientific Symposium of METRA*, Military Equipment and Technologies Research Agency, Bucharest, 25 – 26 May 2006, electronic format
- [4] I. Ivan, S. Capisizu and M. Popa, “Tehnici și metode de audit a programelor sursă,” *Tehnologia Informației și a comunicării – Științe exacte, științe ale naturii*, Defense National University „Carol I” Printing House, Bucharest, pp. 41 – 46, 10 – 11 March 2006.
- [5] M. Popa, “Characteristics for Development of an Assessment System for Security Audit Processes,” *Economy Informatics*, vol. 9, no. 1, pp. 55 – 62, 2009.
- [6] M. Popa, “Requirements for Development of an Assessment System for IT&C Security Audit,” *Journal of Information Technology & Communication Security*, Academy of Economic Studies, Bucharest, ASE Publishing House, Bucharest, Romania, pp. 221 – 230, November 2009.
- [7] M. Popa, “Framework for Compilers’ Security Audit,” *Journal of Information Technology & Communication Security*, Academy of Economic Studies, Bucharest, ASE Publishing House, Bucharest, Romania, pp. 93 – 99, November 2008.
- [8] M. Popa and F. Alecu, “ERP Informatics System Audit,” *2<sup>nd</sup> supplement Informatica Economică*, pg. 109 – 116, 2006.
- [9] M. Popa, F. Alecu and C. Amancei, “Characteristics of the Audit Process for Information Systems,” *Competitiveness and European Integration, The Proceedings of the International Conference Competitiveness and European Integration – Business Information Systems & Collaborative Support Systems in Business*, „Babeș-Bolyai” University, Cluj-Napoca, Romania, 26 – 27 October 2007

- [10] M. Popa and S. Capisizu, "Using Quantitative Methods as Support for Audit of the Distributed Informatics Systems," *Informatica Economică*, vol. 14, no. 1(53), pp. 103 – 112, 2010.
- [11] M. Popa and M. Doinea, "Audit Characteristics for Information System Security," *Informatica Economică*, vol. 11, no. 4, pp. 103 – 106, 2007.
- [12] M. Popa, M. Florescu and C. Bodea, "Information System Quality Evaluation Based on Audit Processes," *Proceedings of the 2008 International Conference of Information Engineering*, Imperial College London, London, Great Britain, 2 – 4 July 2008, Newswood Limited, International Association of Engineers, pp. 494 – 496, 2008.
- [13] M. Popa and A. Paraschiv, "Premises for Development of an Assessment System for Security Audit of Distributed Information Systems," *The Proceedings of the Ninth International Conference on Informatics in Economy – Section 7: Informatics Security*, Academy of Economic Studies, Bucharest, Romania, ASE Publishing House, pp. 827 – 832, 07 – 08 May 2009.
- [14] I. Salmre, *Writing Mobile Code: Essential Software Engineering for Building Mobile Applications*, Addison-Wesley Professional, 2005.
- [15] M. Popa and C. Toma, "Stages for Development the Audit Processes of the Distributed Informatics Systems," *Journal of Applied Quantitative Methods*, vol. 4, no. 3, pp. 359 – 371, 2009.
- [16] M. Popa, C. Toma and C. Amancei, "Characteristics of the Audit Processes for Distributed Informatics Systems," *Informatica Economică*, vol. 13, no. 3(51), pp. 165 – 178, 2009.
- [17] C. Toma, M. Popa and C. Boja, "Mobile Application Security Frameworks," Romanian Workshop on Mobile Business, *Annals of the Tiberiu Popoviciu Seminar – Supplement Romanian Workshop on Mobile Business*, „Babeş-Bolyai” University, Cluj-Napoca, Romania, vol. 6, Mediamira Science Publisher, Cluj-Napoca, pp. 79 – 93, 10 – 12 October 2008.
- [18] <http://searchmobilecomputing.techtarget.com/definition/wireless>
- [19] <http://en.wikipedia.org/wiki/3G>
- [20] [http://en.wikipedia.org/wiki/Mobile\\_operating\\_system](http://en.wikipedia.org/wiki/Mobile_operating_system)
- [21] [http://en.wikipedia.org/wiki/Mobile\\_development](http://en.wikipedia.org/wiki/Mobile_development)
- [22] [http://en.wikipedia.org/wiki/Software\\_development\\_process](http://en.wikipedia.org/wiki/Software_development_process)
- [23] Information Systems Audit and Control Association, *System Development Life Cycle (SDLC) Reviews*, IS Auditing Guideline, Document G23
- [24] Barclay Simpson, *An Introduction to Computer Auditing*, Available: <http://www.barclaysimpson.com>



**Marius POPA** has graduated the Faculty of Cybernetics, Statistics and Economic Informatics in 2002. He holds a PhD diploma in Economic Cybernetics and Statistics. He joined the staff of Academy of Economic Studies, teaching assistant in 2002 and lecturer in 2006. Currently, he is lecturer in Economic Informatics field and branches within Department of Computer Science in Economics at Faculty of Cybernetics, Statistics and Economic Informatics from Academy of Economic Studies. He is the author and co-

author of 6 books and over 100 articles in journals and proceedings of national and international conferences, symposiums, workshops in the fields of data quality, software quality, informatics security, collaborative information systems, IT project management, software engineering. From 2009, he is a member of the editorial team for the *Informatica Economică Journal* and between 2003 and 2008 he was a member of the editorial team for the journal *Economic Computation and Economic Cybernetics Studies and Research*.