

## Methodology Approaches Regarding Classic versus Mobile Enterprise Application Development

Vasile-Daniel PĂVĂLOAIA  
 Alexandru Ioan Cuza University of Iasi,  
 Faculty of Economics and Business Administration, Iasi, Romania  
 danpav@uaic.ro

*In the nowadays enterprise computerized context, there is a trend that shifts the business applications to the new mobile environments. In the light of this information, it is highly important to be knowledgeable about the software development methodologies available in order to make the right choice when it comes to developing a mobile application. The current research aims to presenting the methodological approaches regarding the development cycle of classic enterprise software versus mobile apps. In the first part of the paper a brief literature review regarding the mobile apps is made, for the purpose of justifying the current research theme. The most consistent part of the article puts face-to-face the "classical" and the new development methodologies adapted to the requirements of the new mobile environment trends. The paper also presents the challenges and limitations of mobile applications as well as few of the future trends in the researched domain.*

**Keywords:** Application's Lifecycle Models, Agile Models, Mobile Environment, Mobile Apps

### 1 Introduction to Mobile Applications

Mobile applications, also referred to as mobile apps, are software applications that are usually designed to be ran on smart phones and tablet computers [1]. The term "apps" had become extremely popular and, as a consequence, in 2010 was listed as "Word of the Year" by the American Dialect Society. Mobile applications are available through application distribution platforms, which are managed by mobile operating system developer.

There are several major trends that we believe it contributes to the boost of using the today mobile applications:

- the increased availability and options to Internet connection for mobile devices;
- the shift to the new mobile environments for the business applications;
- cloud application's development;
- the newest research findings that identifies new technologies which increased the performance of mobile device while the price remained at the same level, allowed mobile devices to became available for a wider variety of users.

Another important aspect, when it comes to mobile application development, is the num-

ber of users. Thus, the number of mobile worldwide workers in 2013 is over 1 billion and accounts for about 35% of the worldwide workforce. The structure of the above ratio shows that Asia/Pacific (including Japan) have the largest total number of mobile workers, 785 million, while the second place is taken by Western Europe whose mobile workforce reached 129.5 million in 2013 and accounts for about half of the Western European workforce population. The study [2] also highlights that US has the highest percentage of mobile workers in its workforce, with 75.5% by the end 2013 accounting to 119.7 million mobile workers. A quick glimpse to the future is displayed in the two most recent studies which predict that mobile worker population will reach 1.2 billion by 2014 (a VDC study [3]) and 1.3 billion by 2015 (an IDC study [4]).

Every day, hundred thousands of mobile devices are sold and about the same number of applications is downloaded. By 2015, International Data Corporation (IDC) predicts [5] that 182.7 billion mobile apps will be downloaded. That's a 1600 % increase from the 10.7 billion apps downloaded in 2010. This information highlights the mobile user's ap-

petite for a wide range of software applications.

Cloud application development brings innovation [6] to mobile computing as mobile devices use clouds for a wide range of purposes, from data processing and data storage to other intensive data and information transactions. So far, there are few mobile cloud applications (like Google Maps, Gmail for iPhone and Cisco's WebEx on iPad and several others that use Software as a Service model) but the trend is highly increasing.

The main purpose of the paper is to present the methodologies for developing classic enterprise application versus mobile applications. Also, the current research will present

several news and trends related to mobile applications.

A study [7] performed in 2012 by Keynote entitled Mobile User Survey investigates the Mobile User Preferences, Habits, and Expectations. The study was performed on 5,388 panelists and the study focused mobile user habits and preferences designed to understand usage patterns and expectations. The results are presented in Table 1. The study is important for our paper as it displays important information for a developer, namely the type of mobile device mostly used, the type of operating systems preferred and others.

**Table 1.** The main results of Mobile User Survey investigating the Mobile User Preferences, Habits, and Expectations

The type of mobile device	Smartphone	Tablet
Operating Systems preferred	Android 43% iOS 37%	iOS 43% Android 35%
Type of wireless data access	3G service 57%, 4G service 27%	Wi-Fi 77% 3G 12%, 4G 7%
Top activities	Accessing local information (88%), searching for information (82%), participating in social media/networking sites (76%)	News and entertainment, search for information, watch videos
Top mobile applications used	Map information, social media updates, e-mail and banking information	Shopping, banking, e-mail, Search for information, Access local information
Mobile websites preferred	News and food/entertainment, travel, shopping information and doing shopping	Shopping, banking, e-mail, Social media updates, Map information
The time spent per day browsing the Net for 1-2 hrs	29%	37%

A research [8] conducted in 2010 investigates, among other aspects, the number and

percentage of papers type published within the area of mobile applications.

**Table 2.** Percentage and number of papers by focus of research

Focus	% of papers
Context	8.46%
Development Methodology	6.15%
Interaction	29.77%
Task	46.92%
User	9.23%

Thus, in table no.2 the findings of the research are displayed. It can be easily observed that the majority of papers(47%) focus on allowing users to complete a specific task, while only 6,15% of the papers focus on development methodologies. In our opinion, this information justifies the necessity of the current study.

## 2 Enterprise Applications for Mobile Environment. A Brief Literature Review

According to the study developed by the UTest [9], entitled The Essential Guide to Mobile App Testing, in the last years there was a blast of software application for mobile user ranging from entertainment to business

environments. The more outstanding examples are for the domains listed below:

- *Retail applications* featuring location identification and used for intelligent mobile commerce;
- *Media industry*, the newspapers and magazines are going 100% digital version, without printed versions (paper-based);
- *Travel industry* features mobile ticketing, booking, check-ins, the use of maps, place deals, etc;
- *Education system* have accepted the use of tablets and mobile software applications in classrooms;
- *Healthcare system* is using mobile devices and applications for patient records, physician notes, etc.

Finance area uses specific software for real-time trading, portfolio analysis

Furthermore, the study analyzes more thoroughly the state of application in the B2B environment and declares that within this sector of activity, the mobile applications are running rampant as well, especially for area as:

- Business Applications throughout CRM, ERP, HR systems;
- Productivity by the use of documents, spreadsheets, presentations;
- Collaboration throughout the intensive use of Emailing, InstantMessaging, Document publishing.

Strictly related to the Enterprise mobile applications, the specialized literature reveals several researches that we refer to as milestones that set the framework for such specialized software. Accordingly, Al Bar, et al [10] elaborated in 2011 a comprehensive study regarding the preliminary review of implementing EMA (Enterprise Mobile Application) within the ERP environment. The research focused on the groundwork assessment of EMA in ERP environment by illustrating the implementation challenges to mobile application development framework. The authors present the five layers of enterprise taxonomies for mobile application, namely: mobile broadcast, information, transaction, operation and collaboration specifying that a

new more layer is required. The missing layer should offer mobile devices administration roles allowing it to pursue control and to monitor the users and the other mobile devices within the enterprise which are remotely monitored and controlled by administrators. The authors concluded that the mobile technologies and applications offer many new opportunities for the business environment, together with new development and implementation challenges.

Dospinescu et al [11], also foresee the inevitable entrance of ERP within the mobile application portfolio. The research highlights that ERP, as a significant step forward made by companies, will inevitably pursue to provisioning its main services through mobile technologies. The originality of the research mainly resides in the fact that, besides exploring the landscape of mobile applications, it proposes an architectural model for the mobile services beginning with the necessary functionalities required by a portal of mobile services. The article presents both the general architecture of the portal and the minimal set of functionalities required for implementation in order to ensure the promotion and use of services.

Nevertheless, Unhelkar and Murugesan [12] in 2010, perform a thoroughly study, starting from the idea that enterprises today are facing several challenges in terms of deploying mobile applications. The above mentioned challenges are generated by features like location independence, contextualization, and personalization. The authors propose MADF, a six-layer enterprise Mobile Applications Development Framework which offers a systematic and highly comprehensive solution to mobile application development and maintenance. Thus, within the research, the authors design a new taxonomy for enterprise mobile applications, based on an earlier work done at MethodScience.com.

The above articles illustrate few of the research endeavors related to enterprise mobile applications and justify the current study which will present next the software development methodologies in a comparison manner: classical versus new ones, designated for

particularities of the new mobile environments.

### 3 Software Development Methodologies

According to the particularities of the domain that is to be computerized, different methodologies and programming language can be used [13]. Therefore, according to the domain, different approaches will be taken as projects from different domains ask for different work efforts and risks. Thus, we consider important to review the most important methodologies so that a correct decision to be made when is necessary to computerize the field of financial analysis.

#### 3.1 Defining the Relevant Terms

A working methodology for a software product represents how the development process is structured, planned and controlled. The methodology to develop a system is a standard process, used by an enterprise in order to cover all the necessary stages for the analysis, design, implementation and maintenance of information systems.

In what follows we shall present some terms that are associated to the concept of working methodology:

- Methodology (or method) represents a collection of principles and/or practices;
- Family of methodologies constitutes the set of coexisting alternative methods;
- Framework can be likened to a frame (for methods) that must be developed and personalized before being used;
- Model is a description (for methods) that must be implemented by a method, family or framework.

Methodology can be likened to a class that can be instantiated for a certain project. A framework – working frame – can be considered an abstract class that must first be inherited and extended. The model can be compared to an interface that represents only a description which, later, must be implemented via classes.

#### 3.2 Classical Models Specific to Software Development

The models specific to the development of systems represent a set of specifications that are necessary for the management of the stages in the development of information systems. In general, these models have at least four stages: Planning and selection, Analysis, Design, Implementation and maintenance [14] to which others can be added, depending on the model. In our opinion, it is worth mentioning that: stages must not necessarily be sequential; each stage has certain results and deliverables, and each organization personalizes the model of the life cycle according to the specific profile of its operations.

Table no 2 features a comparative overview of the consecrated models intended for software product development, also known in literature as system development life cycle models [15]. Apart from this overview, the table captures the advantages and disadvantages of the models, according to their specific features mentioned in literature. We could add to the models in the Table 3 the evolutionary model, the three-dimensional model, the X model, the fountain model, the pinball model, the baseball ball Model (it facilitates concurrent development), and the pyramid model.

**Table 3.** Models of system life cycle

Model title	Model overview	Advantages	Disadvantages
Water-fall Model	This is a reference model that consists in a process of sequential implementation, frequently used in software product development. The model's stages are: Identification of requests, Design, Implementation, Integration and Maintenance.	<ul style="list-style-type: none"> <li>- The documentation and structure design represent an advantage when new members join the team;</li> <li>- A simple model that is easy to use by all team members;</li> <li>- Each stage has an expected outcome, as the model is rigid (it imposes total control over stages);</li> <li>- Stages are implemented individually, in order;</li> </ul>	<ul style="list-style-type: none"> <li>- Problems that emerged in one stage are completely solved in another stage;</li> <li>- It does not allow to partition the project according to stages;</li> <li>- When new requirements are formulated, they will be implemented in another version. Consequently, it imposes supplementary costs for their</li> </ul>

Model title	Model overview	Advantages	Disadvantages
		- It is recommended to be used for small projects, in which requirements are very clearly formulated.	implementation; - It is hard to give a correct estimate of the time and cost allotted for each stage; - The finite product is obtained quite late.
Proto-type Model	The model aims to counter certain limitations of the Waterfall model and it is developed starting from currently known requests. With the help of the prototype, the customer perceives more easily how the application functions because he or she can interact with it during the development cycle. The model is used in the case of large and complex systems.	- Users are directly involved in development and they can better understand how the application functions by means of the prototype; - Errors can be detected in time' - The user's feedback is fast, which leads to better solutions; - Less time and lower costs.	- The model leads to an increase in the system's complexity, and it goes beyond the conditions established at first; - The project's analysis is insufficient; - developers may become attached to a prototype, out of subjective reasons, running the risk to transform the prototype into a final product even though the basic architecture is not correct; - It takes an excessive amount of time to implement a prototype.
Incremental model	The model is an evolved form of the Waterfall model. With its help, the application can be delivered incrementally. The latter should have a small scale, so that there could be an interval ranging from a few weeks to maximum two months between launches. The model is recommended for projects that evolve in time.	- The highest priorities are the first to be delivered; - Deliveries are done once in a few weeks; - The feedback on delivered increments offers the specifications for subsequent increments; - The model ensures a low risk of total failure of the project; - Higher priority increments undergo an ampler testing process.	- it may be necessary to create temporary solutions so as to deliver an increment in time; - in some situations, a significant part of the code can be erased; - under the circumstances, planning is difficult.
V Model	This is a variant of the Waterfall model which introduces new concepts such as system and subsystems. The model highlights the demarcation between the user's participation, the architectural model and the implementation model.	- It can also be used in object-oriented programming as it favors the turning of higher structures into prototypes and their reuse; - It provides a strong control over the system, which allows it to be used in the case of complex systems as well; - It encourages one to approach the system according to its constitutive parts.	- The major reproach that has been brought to this model is connected to the fact that the validation stage is launched late, which makes the system suffer from the point of view of efficiency.
Spiral model	The model is based on the convictions that: - development and the need to plan are iterative in nature; - it eliminates the deficiency of the V model in which validation is done late; this model does it earlier, and it does it several times even.	- It allows one to assess risks at several moments; - The model is characterized by high flexibility (both in fund allocation and in defining activities).	- The allocated time and the costs involved are hard to estimate from the beginning.

### 3.3 Agile Models

Apart from the classical models used in software product development, we ought to mention the group of methods termed agile methods. They are based on incremental and iterative development and they are mostly applied in cases that require project specifications and solution be the product of the collaboration between teams organized individually but which aim towards the same common purpose.

Agile methods were developed starting from twelve principles included in the "Agile Manifesto", which we are going to list below:

- satisfying the customer requirements, through quick delivery of usable software solutions;
- possibility to change specifications, no matter how late the product would be in its implementation stage;
- software versions delivered very frequently;

- the main measure of progress is usable software;
- development is at a steady pace, it can maintain a steady rhythm;
- close collaboration between developer and customer;
- face-to-face conversation is the best way to communicate;
- projects are made by motivated individuals of high credibility;
- simplicity;
- teams are organized individually;
- adaptation to changing circumstances;
- constant attention to technical excellence and quality design.

Among the most important agile methods we could mention the Extreme Programming (XP) method and SCRUM. Table 4 gives an overview of the main features and presents the advantages and disadvantages of each method.

**Table 4.** Basic characteristic features, advantages and disadvantages of agile methods

Method title	Model overview	Advantages	Disadvantages
Extreme Programming (XP)	The method aims to lower costs imposed by changes in requirements by running several shorter implementation cycles instead of a long one. In the XP methodology changes are a natural aspect and they must be planned instead of trying to establish a fixed and stable set of requirements. XP supposes the use of the following four basic activities, completed during the software development process: Code writing, Testing, Implementation and Design.	<ul style="list-style-type: none"> <li>- The method offers quality projections and software, and it observes deadlines;</li> <li>- It fully tests all aspects, which provides a high level of software quality;</li> <li>- It encourages team-work;</li> <li>- It offers a high level of satisfaction of customer requirements, through the way in which the latter are gathered;</li> <li>- Test-cases are easy to understand;</li> <li>- The process of development can be completely visualized and measured.</li> </ul>	<ul style="list-style-type: none"> <li>- It is difficult to accomplish because it requires large teams of programmers and much discipline is needed for the project to be completed;</li> <li>- The incremental design proposed by XP does not favor the current software requirements;</li> <li>- XP emphasizes the refactoring, in time, of the implementation process, which can diminish the productivity of other aspects;</li> <li>- The method imposes development based on code and not on design;</li> <li>- Consequently, there is no design documentation.</li> </ul>
SCRUM	The method can be visualized as a skeleton that contains sets of practices and roles. The main SCRUM roles are: -“Scrum Master” - maintains processes; - “product holder” – represents investors and the business;- “Team”-7 employees who are in charge with analysis, design and implementation. During every “sprint” (which can last from 2 to 4 weeks), the team creates an increment that can be delivered to the beneficiary.	<ul style="list-style-type: none"> <li>- It allows time and money to be saved;</li> <li>- It favors quick error correction;</li> <li>- It offers good visibility of the project’s implementation;</li> <li>- It offers permanent feedback from partners;</li> <li>- Possible problems can be solved promptly because they are identified in the early stages;</li> <li>- It leads to the delivery of quality products within the planned delivery schedule.</li> </ul>	<ul style="list-style-type: none"> <li>- The lack of a deadline allows customers to ask for more functionalities;</li> <li>- If the team is not seriously self-involved in the project, the latter may fail;</li> <li>- Requests must be well defined in order to draw a fair estimate of the costs and time needed to complete the project;</li> <li>- The model is recommended to be used only for small and fast projects;</li> <li>- If a member leaves the team, this can massively damage the project.</li> </ul>

During the process of system development, enterprises must consider the various existing approaches and options. Thus, agile models will be opted for only when the project involves [16]:

- (1) requirements that are dynamic or hard to foresee;
  - (2) responsible and motivated developers;
  - (3) customers who are willing to get involved in the completion of the project.
- The advocates of the agile methodology characterize it by the fact that it is grounded

in three key principles of operation, as follows: (1) it is focused more on adaptive methodologies and not on predictive methodologies, (2) it complies with employees’ requirements and not with their roles, (3) it supposes a self-adjusting process. Table 5 provides a comparison between the agile methodology and traditional approaches. In this context, the comparison focuses on five factors: product size, information sensitivity, enterprise dynamics, employees and environment/culture.

**Table 5.** Factors that differentiate system development: agile models vs. traditional approaches

Analyzed factor	Agile Models	Traditional Methods
Dimension	Can be used with small-size products and teams. They are based on tacit knowledge; there are limitations at the level of scalability.	They make it possible to manage large teams and ample projects. They are difficult to use with small projects.
Critical projects	They have not been tested in the case of critical products. Difficulties can occur due to the simplicity of design and the lack of documentation.	They are sufficiently evolved to successfully face critical products. Yet, they are hard to use with small-scale critical projects.
Dynamism	Due to their simple design and multiple versions, they are frequently used in dynamic environments.	Their use in dynamic environments is very expensive but they are considered an excellent choice in stable environments.
Employees	They require the massive presence of a large number of experts, who are expensive and hard to find. It is risky to use employees who are not familiar with agile methods.	They require many experts only in the project definition stage; subsequently, the project can use few employees, on condition that the field should be stable.
Environment/ culture enterprise	They thrive in environments that offer comfort and freedom to their employees.	They are feasible in environments in which employees have comfort and freedom and roles are clearly defined through practices and procedures.

Source: adapted after [16]

### 3.4 Mobile Application Methodologies

The specialized literature is not abundant in presenting references related to the methodologies for the development of enterprise mobile applications. Yet, a study from 2008 performed by Shiratuddin, illustrates the mobile development methodology that were used at that point. The author used four of it

in order to develop  $m^d$ -Matrix, a decision making tool that mainly aims to assisting software developers in the process of choosing the appropriate development methodology for the development project of mobile applications. In Table 6 we present the list of methodologies as well other brief characteristics [17].

**Table 6.** Examples of mobile development methodologies used for the development of  $m^d$ -Matrix decision making tools (adapted after Shiratuddin [17])

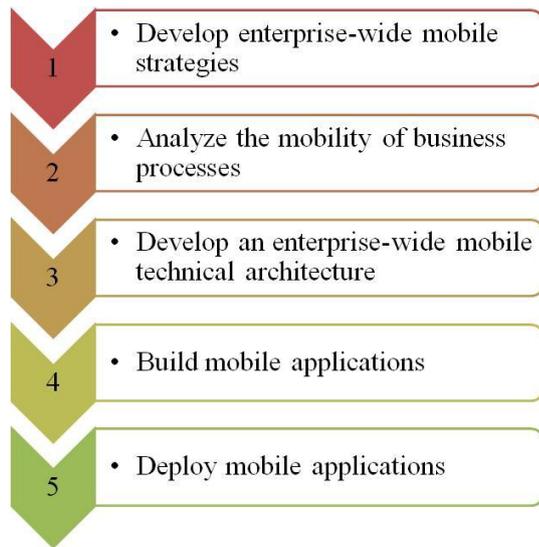
Mobile Development Methodology	Mobile Application	Approach
Mobile-D	Mobile Entertainment	Agile
Mobile RAD	Mobile Commerce	RAD
Dynamic Channel Mode	Location Based Information Systems	Object Oriented
Mobile Engineering (MobE)	Mobile Learning	Process Driven

In the following sections the methodologies identified will be concisely presented by using the guidelines and descriptions offered by its authors.

#### 3.4.1 Chen, M. Methodology for Building Enterprise-Wide Mobile Computing Applications

A pioneer methodology for developing enterprise wide mobile applications was created by Chen [18] in 2004 at George Mason University, School of Management and its main phases are displayed in Fig. 1.

The author specifies that the methodology should be considered as a suggestion and general guideline for the purpose of developing enterprise software for mobile devices. Therefore, the interested companies can plan and develop mobile applications by conducting activities in the various phases of the life cycle iteratively and concurrently in order to facilitate quick prototyping and to receiving feedbacks in due time. The methodology contains five important phases for building mobile enterprise applications are they are briefly presented below:



**Fig. 1.** A Methodology for Building Enterprise-Wide Mobile Applications  
Adaptation after Chen, M., 2004[18]

1. Develop enterprise wide mobile strategies. Enterprise employs initiatives in mobile applications as they are interested in taking advantages from the emerging mobile computing technologies and to improve their competitive advantages over the market by better serving their customers. The mobile strategy team should be formed out from the top level management and the process of creating enterprise-wide mobile strategies should be based on existing business strategies and objectives, and on high-level understanding of mobile technology's trends and impacts.
2. Analyze the mobility of business processes. Before the rise of mobile environments, enterprise communication and computing support performed exclusively throughout desktop computing tools. Now-a-days, the workers are no longer forced to be static due to mobile technologies that enable organizations to redesign their business processes. In this phase, the mobile analysis team both develops a business process map and identifies mobile projects that bring in more revenues and improve the quality of relationship with the business partners.
3. Develop an enterprise-wide mobile technical architecture. This phase facilitates the

development of comprehensive mobile technical architecture using mobile business strategies, mobility analysis of business process map, as well as mobile application portfolios developed in the previous two phases.

4. Build mobile applications. In this phase, there will be revised and considered the most obvious limitations in terms of building apps for mobile devices, such as smaller screen sizes, less efficient data entry methods, limited local CPU power. As a result, developers need to design new development tools and consider the above limitations in building mobile applications

5. Deploy mobile applications involves a proper user training and support. The management strategies should be applied together with mobile application deployment to ensure effective behavioral changes of people who are involved in this mobilizing phase.

The first three phases of the methodology proposed by Chen are important for enterprise-wide mobile computing effort while the last phases are more project-specific.

### 3.4.2 Mobile-D Methodology Overview

Some authors consider that agile methodologies (presented above in current research) offer a reasonably development solution for mobile applications. Yet, Mobile-D is considered more detailed and serves better [19] the purpose of development.

Abrahamsson, et al, [20] affirms that Mobile-D methodology should be used by a team of maximum ten developers working towards a product delivery within ten weeks. The Mobile-D methodology was first proposed by VTT Electronics in 2006 [21] and includes five phases presented in Fig. 2. Each phase has a number of associated stages, tasks and practices.

Spataru in 2010 [19] proposes an extent of the methodology by adding a new phase (called Evolve [19]) that deals with continuously integrating end-user feedback on the delivered product into future releases.



Fig. 2. Mobile-D with added Evolve phase. Adapted from [19], [21]

Since 2006, Mobile-D has been applied in various development projects and some advantages have been identified. For example, an increased progress visibility, earlier discovery and repair of technical issues, low defect density in the final product, and a constant progress in development [20]. Other applications of the method are presented in [22] and [23].

### 3.4.3 Other Development Methodologies For Mobile Applications

#### Mobinex's - The Mobile Application Development Methodology V3

Mobinex, a leading provider of mobile applications and on-device solutions, is committed to increasing revenues of mobile operators, content providers, media companies and enterprises with compelling user experiences [24]. The methodology was proposed in 2009 (updated in 2010) for the purpose of develop-

ing mobile applications. The main phases of the methodology are presented in Figure 3.

#### Mobile Development Method Proposed by Rahimian & Ramsin

The mobile development methodology proposed by Rahimian & Ramsin in 2008 [25] is created in four iterations and the beginning point is the generic software development lifecycle, as Analysis, Design, Implementation, Testing. The first iteration details the methodology by adding those practices that are found within the agile methodologies. The second iteration details the process concerned with introducing new product/service on the market. In the third iteration, concepts of Adaptive Software Development are integrated into the methodology and in the final iteration the prototyping was added to mitigate likely technology-related risks. The methodology as presented by the authors is illustrated in Figure 4.

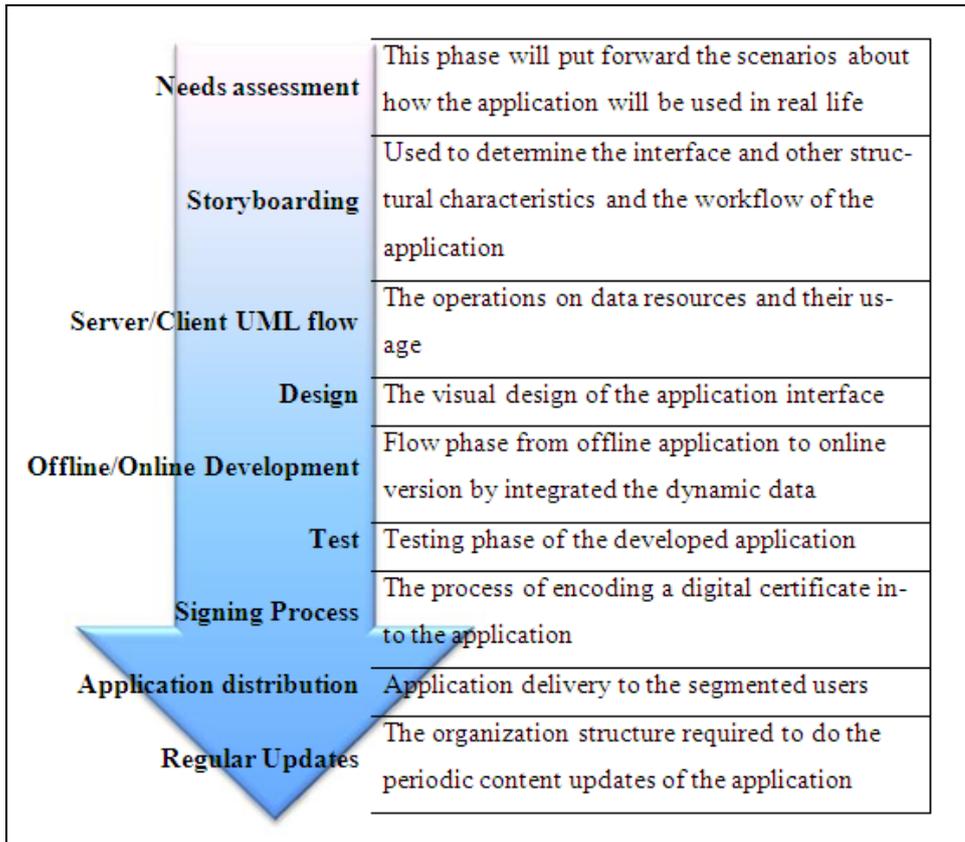


Fig. 3. The phases and explanations of Mobinex Mobile Application Development Methodology V3. Adapted from [24]

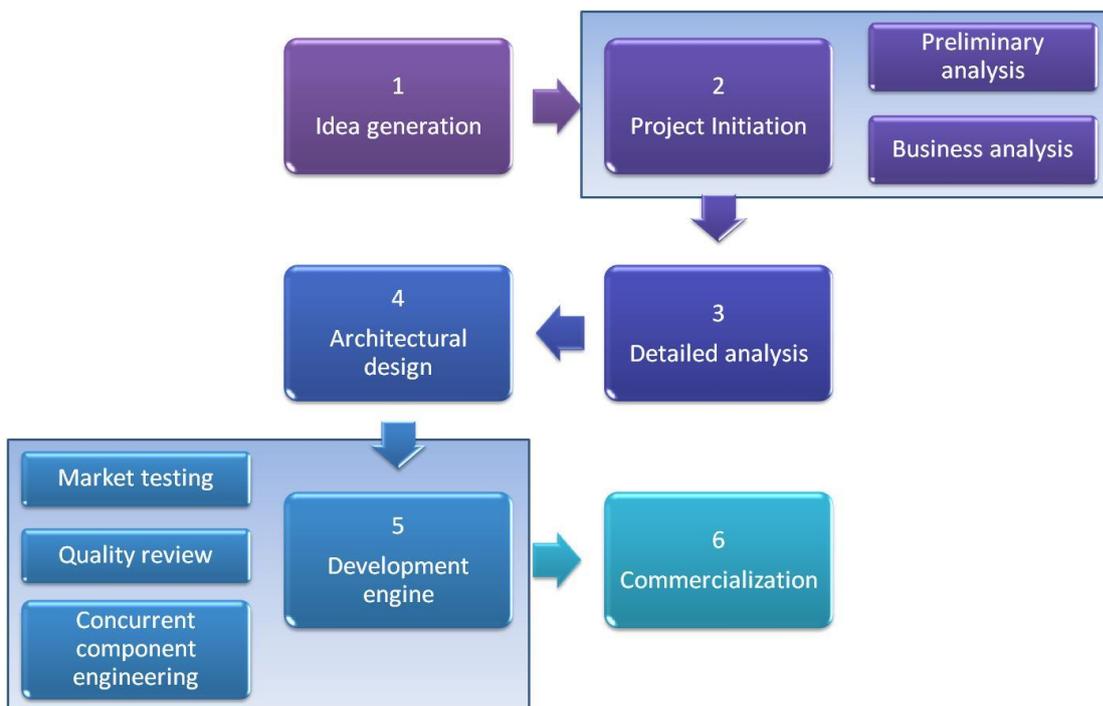


Fig. 4. Adaptation of the mobile development method proposed Rahimian & Ramsin [24]

#### 4 Limitations and Challenges of Mobile Applications

Though highly popular, the mobile applications have general limitations as the advent and development of new mobile devices present new usability challenges [26] that are difficult to model. Some researchers have highlighted certain limitations [27], [28] that are listed below (we have to mention that some of them already have or may be overcome as the technology evolves rapidly):

- *Geographic environment and meteorological conditions*: the user may access the application from different geographic location and even interact with objects from the environment that can be attention distractive. Also, weather conditions and the range from the nearest signal point can significantly interfere with signal transmission and reception.
- *Display resolution*: in comparison with tablets display, the mobile phone's display is reduced which results in lower quality images.
- *Connection issues*: the internet connection may be slow or interrupted from time to time. Reception in tunnels, inside buildings with massive walls and rural areas is often poor. Therefore, it will dramatically affect the performance of the mobile application unless it is developed in such way that covers such episodes (by working offline, for instance).
- *Power and processing capability*: for portability reasons, the mobile devices are less performing if compared with desktop computers. This will limit the type of application that can be installed and used on mobile devices.
- *Screen size (in the case of phone devices) and data entry*: describe the human interface with mobile device. The screens and keyboards are small making it difficult to use for repetitive and fast operations. Though alternate input methods such as speech or handwriting recognition exists, they require training and are expensive.
- *Security issues*: the mobile worker becomes highly dependent on public networks and, thus, it requires careful use of

VPN as it can easily be attack through a huge number of networks interconnected through the connecting line.

Additional challenges in mobile application development include [29]: customizing mobile functionality, according to the user profile, ensuring access and usability, facilitating optimum user interactions on mobile devices, and dealing with the bandwidth and poor reliability of wireless communication.

#### 5 Conclusions

The theoretical approach to computerized modeling shows that the enterprise model consists in the computerized representation of an enterprise's structure, activities, processes, information, resources, employees, behaviors, purposes and constraints. Thus, from the point of view of the design stage, the enterprise model must offer a specialized language meant to define the enterprise as explicitly as possible [30]. In general, the classical models of the software products life cycle contain four large stages: Planning and selection, Analysis, Design, Implementation and Maintenance, to which other stages can be added if the case may be. Agile models have been developed by programmers out of their wish to be as efficient as possible in successfully completing projects and recently, adapted model based on Agile are also used for designing mobile applications. The Agile models are opposed to the classical models that offer the beneficiary the possibility to test the product only at the end. Instead, agile methods launch new versions weekly and they incorporate the beneficiaries' specifications from one version to another. Under the circumstances, we consider that such an approach could be successful in the development of applications for the new mobile environments. Due to the advent of mobile computing, the way of doing business has changed dramatically. The mobile workers can conduct businesses at any time without been constrained by the availability of physical networking connections or specific computing platforms. This way, the information can be placed in the hands of mobile workers while they are away from their offices. The

mobile application developing methodologies that are presented within the current research are meant to assist enterprises in planning and developing enterprise-wide mobile strategies and applications. Voke Inc. analyst Theresa Lanowitz [31] affirms that “Mobile isn't a luxury, it's a necessity” and we can only agree with her and the business trend is for the enterprises to tackle mobile applications higher than ever. In terms of mobile technologies methodologies [32], [33], we may say that due to the particularities of mobile apps, the development cycle gets shorter and, unlike generic enterprise applications, the new mobile environment require more software updates for the mobile projects.

## References

- [1] T. Treeratanapon, “Design of the Usability Measurement Framework for Mobile Applications,” in *Proceedings of the international Conference on Computer and Information Technology (ICCIT'2012)* June 16-17, 2012, Bangkok, pp.71-75
- [2] S. Drake, et. all, *IDC's Worldwide Mobile Worker Population 2009-2013 Forecast*, June 2010, [http://www.gotomypc.com/remote\\_access/images/pdf/How\\_to\\_Equip\\_Your\\_Company\\_for\\_the\\_New\\_Mobile\\_Workforce.pdf](http://www.gotomypc.com/remote_access/images/pdf/How_to_Equip_Your_Company_for_the_New_Mobile_Workforce.pdf)
- [3] D. Krebs, “Enterprise Mobile Workers: Tracking Tomorrow's Mobility Opportunities,” *VDC Research*, March 2012 [http://vdcresearch.com/\\_Documents/researchnote/res-attachment-2667.pdf](http://vdcresearch.com/_Documents/researchnote/res-attachment-2667.pdf)
- [4] S. Crook, M. Shirer, *Mobile Worker Population to Reach 1.3 Billion by 2015*, IDC January 2012, <http://www.idc.com/getdoc.jsp?containerId=prUS23251912>
- [5] IDC, 2012, Available at: <http://www.idc.com/getdoc.jsp?containerId=prUS22917111>
- [6] S. S. Qureshi, T. Ahmad, K. Rafique, “Mobile cloud computing as future for mobile applications - Implementation methods and challenging issues,” in *Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on*, pp. 467 – 471.
- [7] Keynote, 2012 Mobile User Survey: Mobile User Preferences, Habits, and Expectations <http://www.keynote.com/docs/reports/Keynote-2012-Mobile-User-Survey.pdf>
- [8] R. Harrison, D. Flood, D. Duce, *Usability of Mobile Applications: Literature Review and Rationale for A New Usability Model*, Oxford Brookes University, date of last revision: 2013 [http://cct.brookes.ac.uk/staff/resources/rachelharrison/HarrisonFloodDuce\\_JOIS.pdf](http://cct.brookes.ac.uk/staff/resources/rachelharrison/HarrisonFloodDuce_JOIS.pdf) p.19
- [9] UTest, *The Essential Guide to Mobile App Testing*, Available: [http://c0954852.cdn.cloudfiles.rackspacecloud.com/uTest\\_eBook\\_Mobile\\_Testing.pdf](http://c0954852.cdn.cloudfiles.rackspacecloud.com/uTest_eBook_Mobile_Testing.pdf)
- [10] A. A. Bar, E. Mohamed, M. K. Akhtar, F. A. Abuhashish, “Preliminary review of implementing Enterprise Mobile Application in ERP environment,” in *International Journal of Engineering & Technology IJET-IJENS 2011*, Vol: 11, No. 04, 2011, pp. 77.
- [11] O. Dospinescu, D. Fotache, B. A. Munteanu, L. Hurbean, “Mobile Enterprise Resource Planning: New Technology Horizons,” *Communications of the IBIMA*, 1(11), 2008.
- [12] B. Unhelkar, S. Murugesan, “The enterprise mobile applications development framework,” *IT professional*, 12(3), pp. 33-39, 2010.
- [13] V. D. Păvăloaia, “Methodological Approaches to Computer Modelling Possibilities in Financial Analysis,” in *Annals of the Alexandru Ioan Cuza University-Economics*, 59(1), 2012, pp. 375-393.
- [14] J. Valacich, J. George, J. Hoffer, 2012. *Essentials of Systems Analysis and Design*, 5/E, Prentice Hall.
- [15] D. Oprea, G. Meșniță, F. Dumitriu, *Analiza sistemelor informaționale*, Editura Universității "Alexandru Ioan Cuza" Iași, 2005.
- [16] B. Boehm, R. Turner, “Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods,” *Proceedings of the 26th International*

- Conference on Software Engineering*, Washington, DC, USA, 2004.
- [17] N. Shiratuddin, S. M. Sarif, "The md-Matrix: a learning tool in the mobile application development course," *International Journal of Mobile Communications*, 7(4), 2009, pp. 494-514.
- [18] C. Minder, "A methodology for building mobile computing applications," *Int. J. of Electronic Business*, 2004, Vol. 2, No. 3, pp. 229 – 243, Available: <http://www.inderscience.com/info/inarticle.php?artid=5139>
- [19] A. C. Spataru, *Agile Development Methods for Mobile Applications*, School of Informatics, University of Edinburgh, 2010, <https://www.inf.ed.ac.uk/publications/thesis/online/IM100767.pdf>
- [20] P. Abrahamsson, A. Hanhineva, H. Hulkko, T. Ihme, J. Jääliñoja, M. Korkala, et al., "Mobile-D: an agile approach for mobile application development," *Conference on Object Oriented Programming Systems Languages and Application; Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications*, 2004, pp. 174-175.
- [21] VTT Electronics, *Portal of Agile Software Development Methodologies*, 2006, retrieved from Mobile-D Method: <http://virtual.vtt.fi/virtual/agile/mobile.html>
- [22] M. Pikkarainen, O. Salo, & J. Still, *Deploying Agile Practices in Organizations: A Case Study*, Springer Berlin / Heidelberg, 2005.
- [23] H. Hulkko, P. Abrahamsson, "A Multiple Case Study on the Impact of Pair Programming on Product Quality," *Proceedings of the 27th international conference on Software engineering*, St Louis, 2005, pp. 495-504.
- [24] Mobinex, *Mobile Application Development Methodology V3*, [http://developer.smartface.biz/documents/Application\\_Development\\_Methodology.pdf](http://developer.smartface.biz/documents/Application_Development_Methodology.pdf), updated 2010.
- [25] V. Rahimian, R. Ramsin, "Designing an Agile Methodology for Mobile Software Development: A Hybrid Method Engineering Approach," *Second International Conference on Research Challenges in Information Science. RCIS 2008*, Marrakech, 2008, pp. 337-342.
- [26] R. Harrison, et al., "Usability of Mobile Applications: Literature Review and Rationale for A New Usability Model," *Journal of Interaction Science*, Springer (in press) 2013, Available at: [http://cct.brookes.ac.uk/staff/resources/rachelharrison/HarrisonFloodDuce\\_JOIS.pdf](http://cct.brookes.ac.uk/staff/resources/rachelharrison/HarrisonFloodDuce_JOIS.pdf)
- [27] W. Zhang, "Architecturally Reconfigurable Development of Mobile Games," *Second International Conference on Embedded Software and Systems (ICESSE'05)*, Xian, 2005, pp. 66-72.
- [28] G. Deepak, B. S. Pradeep, "Challenging Issues and Limitations of Mobile Computing," in *Int. J. Computer Technology & Applications*, Vol. 3, No. 1, pp. 179.
- [29] B. Unhelkar, S. Murugesan, "The enterprise mobile applications development framework," *IT professional*, 12(3), 2010, pp. 33-39.
- [30] C. Strîmbei, "Issues Concerning the Impact of the Object-Relational Model on Analytical Data Processing," *Annals of the Alexandru Ioan Cuza University-Economics*, 59(1), 2012, pp. 395-401.
- [31] J. Lent, 'New normal' emerging as software teams go mobile, March 2013, <http://searchsoftwarequality.techtarget.com/news/2240180414/New-normal-emerging-as-software-teams-go-mobile>
- [32] K. Slattery, *Infographic: Trying to Decide Between a Native or Web App?*, April, 2013, <http://www.mobileapptesting.com/>
- [33] L. Păvăloaia, "Business Valuation in Terms of Indicators of Sustainable Development," *CES Working Papers*, (1), 2013, pp. 66-76.



**Vasile-Daniel W. PĂVĂLOAIA** has graduated the Faculty of Economics and Business Administrations in 1998. From 2008 he holds a Ph.D. diploma in Economics with Accounting Information Systems specialization and joined the staff of the “Alexandru Ioan Cuza” University of Iasi in 2002. Since graduation he has granted many professional awards and scholarships and took part in several academic trainings and summer schools. As a Fulbright fellow he attended a master program at Washington State University in 2003-2004. Since 2013 he is Associate Professor within the Department of Accounting, Management Information Systems and Statistics at the Faculty of Economics and Business Administration and teaches courses at both graduate and undergraduate level. He is the co-author of more than 16 books and the single author of one book. His research area is in the field of accounting information systems, business intelligent systems, SMEs architecture and enterprise modeling.