ACSIJ Advances in Computer Science: an International Journal, Vol. 4, Issue 5, No.17 , September 2015
ISSN : 2322-5157
www.ACSIJ.org

Approaches and methodologies for mobile software engineering

Serena Pastore[1]

[1]**Astronomical Observatory of Padova, National Institute of Astrophysics (INAF)**
Padova, 35122, ITALY
serena.pastore@oapd.inaf.it

Abstract

The development of software for mobile devices takes place in a dynamic environment where constraints, technologies and user needs change very frequently requiring enhanced approaches and methodologies in software engineering essential to deal with the concept of bring your own device (BYOD). Mobile apps are developed for various purposes in different categories. This paper discusses about the challenges that affect mobile software engineering, specifically for a science educational and outreach aim as regards technologies, approaches and methodologies that could be applied to unlock the full potential of mobility. In particular, it analyzes the main challenges to development in this field, such as dealing with enhanced connectivity and networking protocols (e.g., from Bluetooth Low Energy, IEEE 802.15.4-based protocols and new Wi-Fi versions) and the fragmented ecosystem of mobile platforms. Moreover, it addresses the possible application of specific development methodologies such as Agile software methods.

Keywords: *Mobile software engineering, agile methodologies, mobile web software development, PaaS cloud services.*

1. Introduction

In recent years, there has been exponential growth in both the development and use of mobile applications, thus presenting new challenges to software engineering [1]. The concept of "bring your own device (BYOD)" has apparently shifted the computing paradigm from a desktop or even laptop computing to a fully mobile computing, in which users have access to computing wherever they are, and the data transfer takes place through the cloud computing platforms [2].

The development of software for mobile devices takes place in a dynamic environment where technologies, approaches and methodologies have to be improved to meet the changing needs and expectations of the user and to keep up with the enhanced capabilities and new applications of mobile devices. The same constraints and challenges apply to all category of apps, whether educational or outreach in different sectors (e.g., scientific culture, health, communication) or otherwise.

Mobile platforms are rapidly changing, with the enhancement of processing capabilities and the addition of features through both internal and external hardware modules (Fig. 1) [3]. Modern mobile devices are equipped with a Global Positioning System (GPS) module that identifies the location of the device and different kind of internal sensors such as proximity or gyro sensors, the accelerometer and gesture sensor. Improvements in this area can be seen in the integration of humidity and temperature sensors, and motion sensors with external sensors interacting with the mobile devices, such as 3D sensors or the skin stickers [4]. The grow of mobile sensor technology has transformed mobile devices into self-aware devices with continuous data logging. The trend has been to add more sensors, for example medical sensing or 3D/stereo cameras, in order to enable the device to understand emotions as well. The other major aspect of wireless network communication technologies, is that a mobile device includes short, medium and large range technologies, since their respective applications are different (e.g., data transfer with other devices like the smart applications related to activity logger, as well as micro-payments and broadband access to the Internet).

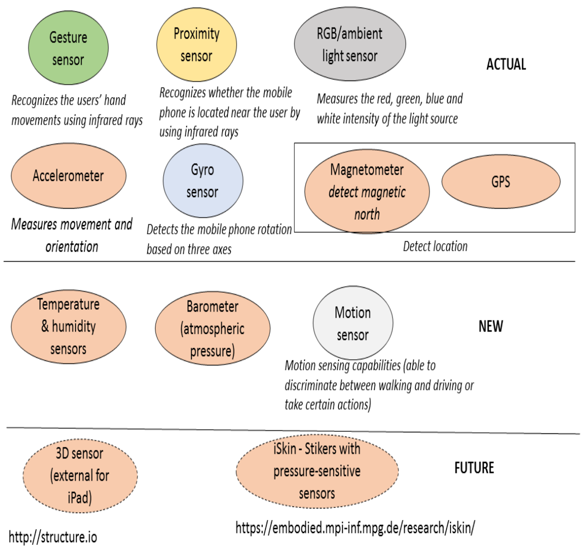

Fig. 1 Type of sensors related to mobile devices

When running on mobile platforms, modern applications need to scale on demand according to the hardware abilities. Moreover, other challenges affecting the development include the fragmentation of the hardware and software architectures and thus interoperability issues, the security implications and the authorization process for dataflow in a network environment, together with design features of customized user interface that are mainly based on touch interaction.

Mobile phones and devices are also becoming more powerful in terms of network connection and in the exploitation of web infrastructure as the preferred platform thanks to the availability of different cloud services. Such services allow for saving documents or using online applications without any local installation and synchronization of features in any device. In this context, as consumer expectations evolve, developers are faced with an array of challenges, both technical and business, that affect how they should systematically design, build and deploy new applications and systems.

Design applications involve the improvement of methodologies to follow the development lifecycle of mobile applications that manifest differences with desktop-based applications. Agile software development [5], is a group of software development values, principles and practices based on incremental development, in which requirements and solutions necessitate the capacity of building self-organizing and cross-functional teams. There are different approaches that could be used in this lightweight software development. Agile, however, is applied in the wider context of project management (PM), and Agile PM [6] is an innovative approach to project management that integrates with more formal approaches such as PRINCE2 [7]. Conceptually, Agile is also considered as a subset of lean principles and practices [8] that are born during manufacturing processes, but have been modified for application in the domain of software development.

This paper thus discusses about the challenges facing mobile software engineering, which even if independent by the target context, are considered for science dissemination or educational purposes.

It provides an overview of the main issues affecting mobile ecosystem, such as hardware and software features of devices architectures (i.e., plethora of devices, platforms, operators, languages and app stores) and the wireless network protocols that are the bases of mobile data transfer. Challenges related to the increasing fragmentation of the mobile ecosystem involve important decisions on the choice of a specific platform or framework that affects development strategy. Moreover, the paper focuses on methodologies and approaches needed in mobile development. Techniques should consider the specificity of applications in every phases of the development lifecycle (i.e., from planning and analysis, designing, building, testing, deploying, and reviewing steps) in order to understand if traditional or new software engineering could be adopted or if there is a need for other methods.

The diversity lies both in hardware and software features and in different approaches to development related to the deployment and distribution of an app, since basing the approach on the store's business model leads to lower revenues, shorter development cycle and more user interface (UI) complexity.

2. Addressing challenges to the mobile development

Developing robust mobile applications [9] is now the new frontier for software engineering. Therefore, realizing mobile software requires advanced practices, tools and methodologies, and involvement of computer engineers, similar to the development of desktop-based software development.

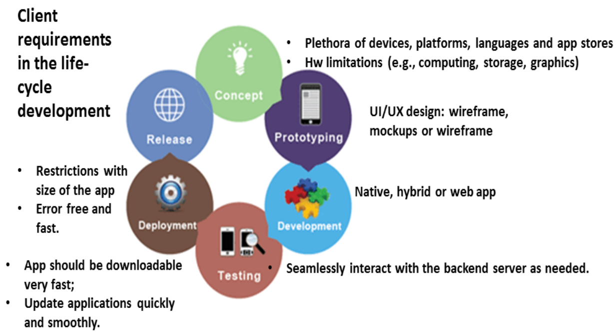

Fig. 2 Challenges of mobile development in the several lifecycle development steps

Techniques should take into consideration the platform complexity, fragmentation and constraints caused by working on multiple hardware and software platforms with different computational and storage limitations. This affects the strategy to choose in developing apps.

An application designed for a desktop operating system, does not create a mobile user experience, since the reduced capacities of a mobile devices (e.g., graphics and memory) might not handle various media as efficiently as the website itself. Approaches using responsive design [10], such as a methodology that enables websites to recognize the type of device on which the app is visualized so that it can change the layout and the image size accordingly, may not fit the scope due to the limitation on exploiting the full capabilities of the mobile device.

Whether it is better to host mobile applications natively on the device itself, or include a mobile version of their website, or both, depends on the context and purpose. A mobile site usually aims to promote something (e.g., a company or an event) or to share information or content. Standalone applications are useful as connections between the data collected by the devices and a cloud service (e.g., apps in the healthcare market).

UI design [11] is greatly limited due to the complexities of interaction with the mobile device and its sensors. At the same time, the developer has to provide a great user experience despite reduced capabilities. Constraints in the development phase include power management, security and privacy models and policies, and dependence with external services. Many development tools using multiple dynamic languages are available through various frameworks for providing native, web or hybrid application [12]. Testing and verification techniques should validate the execution of an app on a multitude of different devices. All development phase lead anyway to have very short development cycles, especially considering the business models of the app stores.

Based on such assumptions, we focus on the key factors that affect mobile development in its ecosystem as recently highlighted by Gartner's research [13], namely multi-platform and multi-architecture implications and the wireless connectivity and networking perspective.

2.1 Multi-platform and multi-architecture framework

The design and development of apps involve different decisions, depending on their purpose and intended scope. The mobile ecosystem is highly fragmented. It appears in a number of hardware and software platforms requiring different programming languages, tools, and deployment methods.

	iOS (Apple)	Android	Blackberry OS	Windows Phone
Languages	Obj-C, C, C++	Java, (some C, C++)	Java	C#, VB NET, etc
Tools	Xcode	Android SDK	BB Java Eclipse Plugin	Wisual Studio, Windows Phone Dev Tools
Executable files	.app	.apk	.cod	.xap
Application stores	Apple Store	Google Play	Blackberry Store	Windows Phone Store

Fig. 3 Fragmentation of mobile operating systems, hardware, languages and tools

The market is oriented on three key platforms as well as three key applications architectures (i.e., native, web and hybrid). The combination of these two variables (platform and architecture) requires the support of development tools to minimize software development costs (Fig.4).

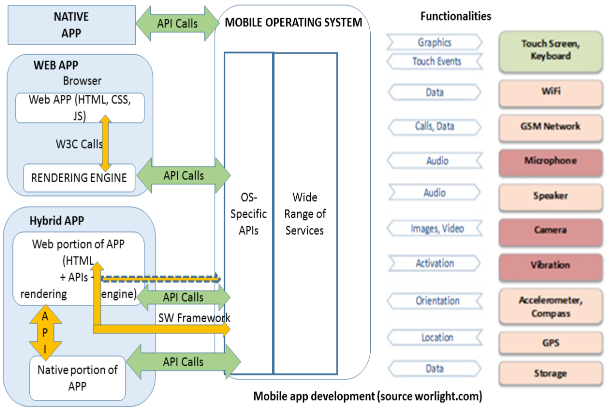

Fig. 4 The three approaches and vision in mobile app development

Tool selection is a complex balancing act, trading off many technical (especially in its capability of exploiting all hardware mobile features and graphics) and nontechnical issues (such as productivity versus vendor stability), that needs a portfolio of tools to deliver for the required architectures and platforms.

Another aspect to consider is relate to the apps' portability to different mobile platforms. The use of a standard language such as HTML5 [14] could guarantee a multi-platform portability, thus reducing the need for duplication and avoiding an increase in development time and cost. Considering the two strategies (i.e., web or native apps), advantages and disadvantages needs to be balanced, but developers should consider that, according statistics [15], mobile users spend 80% of their time using apps and only 20% of their time on mobile browsing. Mobile sites are typically less expensive to develop (e.g., its realization implies the application of HTML5 tags), and might be automatically included in the website developed services. Instead, native applications that generate a smoother experience usually require a team to make separate adjustments for changes in alignment with website version, which can lead to a higher cost of maintenance. Unfortunately, native apps are strictly related to the distribution channel and can go to undiscovered unless properly promoted. Web solutions have cost advantages and are easy to use; however, native applications offers their own batch of benefits and are more suited to address users' needs through easy-to-use programs for niche businesses and organizations. They enable customer engagement (e.g., by offering deals or specific information), ensure safer data transfers, and provide a more personalized experience. At the same time, hybrid apps, i.e., a mix of web and native approaches, are a trade-off solution, since they are based on frameworks working with HTML5-based standards that are able to convert a web page into a native app. Unfortunately, HTML5, is still fragmented and immature, therefore posing many implementation and security risks. However, when its development tools will be sufficiently stable, the popularity of the mobile web and hybrid applications will increase, because, despite many challenges, HTML5 is able to deliver applications across multiple platforms through Application Programming Interfaces (APIs) for each feature (e.g., graphics and video).

Another aspect delivering an advanced user experience. Designers should consider a variety of techniques and methodologies (e.g., motivational design, use cases, storytelling) to deliver apps that are easy to use, equipped with enhanced features and graphical appeal and able to take advantage of mobile hardware add-ons (e.g., camera and sensors), resulting in an augmented reality experience [16].

There are high standards for UI design that can only be met with skills and hard work, since the specificity of both the mobile device and its BYOD use. An app's execution cannot follow a linear cycle. An app could be interrupted and put it in background, while the user is taking a call. User attention in interacting with an app could be only partial, since unlike with a desktop, the user could be in any situation while using the app, such as walking, running, or sitting on a bus or a train. These are aspects that affect the design and should be included in the analysis. Today's mobile devices contain faster chips and better graphics processors that greatly boost rendering, but engineers should be careful not to compromise an application's functionality for the sake of excessive processor-heavy animations.

2.2 Wireless network protocols: connectivity vs. networking

Network is key factor on mobile apps. Mobile apps needs for their mobile nature wireless connections for data transfer, but several are the wireless protocols. They can be divided into two distinct groups: those providing long-range connectivity (e.g. networking through the IEEE 802.11 standard better known as Wi-Fi [17] or cellular networks), and those providing short-range connectivity (e.g., Bluetooth [18], Near Field Communication (NFC) [19] or infrared). Most apps, mainly interacting with other devices, sensors or objects, make a great use of short-range protocols (e.g., NFC for mobile payments [20] or Bluetooth in the interaction with on-body healthcare systems).

Other apps exploiting precise indoor location currently use technologies such as Wi-Fi or GPS to find out the individual's location for enabling location-based services. In many apps, the two protocols are related. Short-range protocols are used in interactions with other devices (singles or acting in a sensor networks), while collected data are sent through long-range connection to be saved on Internet-based databases or cloud service.

Another case is that of Internet-based apps, used for multimedia streaming (e.g. also TV programs) and thus exploiting the full potential of a broadband connectivity, such as the services offered by the new generation of cellular networks like Long term Evolution (LTE) networks [21].

2.2.1 Connectivity protocols

Mobile devices are currently part of personal area network (PAN) connecting not only smartphones or tablets, but other devices as well (e.g., sensors). Such networks are standardized (Fig. 5) by IEEE 802.15 standard [22], that with its various specifications and have a great role in the development of smart homes and cities [23]. Wireless networks are used for connectivity rather than networking, for the diffusion of such technology on mobile devices. Apps are becoming the method by the information collected by the various wearable devices (e.g., on-body healthcare sensors, smart watches, hand mounted devices (HMD) [24] display devices like Google Glass and sensors embedded in clothes and shoes), interact. These devices will communicate with mobile apps to deliver information, collect and transmit this information over the Internet in order to provide some services. The developed apps provide a wide range of products and services, especially in areas such as fitness or healthcare. This is also the idea behind the Internet of Things [25] wireless sensor technologies.

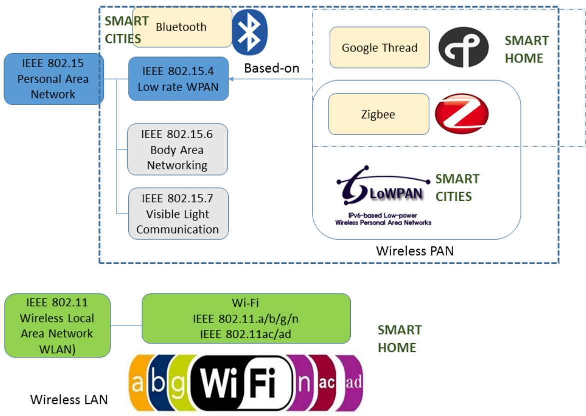

Fig. 5 Personal Area Networks standards and Wireless Local Area Network standards

Smart objects, as part of the IoT, communicate in some way with a mobile device and its apps. Such apps perform many functions like acting as remote controls, displaying and analysing information, interfacing with social networks to monitor "things" that can tweet or post, paying for subscription services, ordering replacement consumables and updating object firmware. The physical connectivity medium is wireless, and there is a necessity of low-energy network standards at a short/medium range. This is due to the battery management issues, which implies limited consumption by each individual app.

In the past, Bluetooth and Wi-Fi have been widely used, but at expense of battery life. Now the industry is working on low power versions of such protocols. Currently the reference protocols are the enhanced versions of the Bluetooth protocol with comes with different names (e.g., Bluetooth Smart or Bluetooth Low Energy or BLE, Bluetooth 4) [26]), and other protocols based on a sub-specification of the IEEE 802.15 standard that is the IEEE 802.15.4 [27] as implemented by different organizations. Bluetooth is a proprietary standard managed by the Bluetooth Special Interest Group (SIG); it was initially standardized by IEEE as IEEE 802.15.1, but this is no longer maintained. The IEEE 802.15.4 standard is a communication standard for wireless PANs optimized for low power devices and operation on, in, or around the human body, but not limited to humans. It defines data communication low-data-rate, low-power and low-complexity short-range radio frequency (RF) transmissions. Another sub-specification of the same IEEE 802.15 standard is the IEEE 802.15.6 [28]. This specification is applied to the wireless body area networks (WBAN) also known with the name of mobile body area (MBAN) networking [29]. It defines a short-range wireless communication near or inside a human body, using existing industrial scientific medical (ISM) bands or other approved bands. It support quality of service (QoS), low power and data rate up to 10 Mbps, and takes into account the effects of radiations on the human body.

The applications in the MBAN are in a state of constant growth to match the development of wearable networking devices. A standard case is a wearable heart-rate monitor device enabled with wireless protocol that transmits a signal to a gateway device that in turn can instantly relay that information over the Internet to an off-site cloud service. Connectivity protocols used in these contexts are Bluetooth's new version (i.e., Bluetooth 4.0, Bluetooth Smart, BLE) and other IEEE 802.15.4-based protocols such as Zigbee [30] and Google Tread [31]. Their applicability in modern mobile devices has close relationship to IoT applications. Such protocols are defined as connectivity protocols for the things such as sensors, while letting Wi-Fi manage the networking aspect. Apart from the wearables market, there is also the smart home market, where the adoption of such protocol, and thus the development of mobile app interacting with them, could have a great scope. Probably, an integration [32] these technologies in IoT is required rather that a competition between them, since essential building blocks of the things (e.g., sensors, actuators, controllers), are based on PAN connectivity.

Bluetooth connectivity is enabled in many peripheral devices (e.g., most wearables devices like fitness bands and step-counters, as well as many medical devices). The major problem of the high consumption of battery has been addressed with BLE. Bluetooth Smart (formerly known as Bluetooth Low Energy or BLE) was further specified in Bluetooth Smart Ready, that integrating classic and smart Bluetooth is used for backward compatibility.

Zigbee is a global wireless protocol supported by the Zigbee Alliance, based on IEEE 802.15.4 specification, in order to provide a reliable, robust, low power, scalable, and secure connection. Currently at the 3.0 version, it is used for machine-to-machine (M2M) communication [32], i.e., for small volume of data transfer over a short distance, but consuming little power. Zigbee-based chips are connected in a mesh network (rather than the star network of Wi-Fi) and uses ZigBee Pro networking to enable reliable communication.

Its applications are mainly in the smart homes, connected lighting and the utility industry. The Zigbee Alliance announced that is working to create a standard for another special type of network that are the neighborhood area network (NAN) [33]. Such networks allows connecting devices outside the home (e.g., smart meters or data aggregators). Currently, only few of such communications are standards based, but analysts [34] estimate that, by 2020, 85% of NAN connectivity will be standards-based. Transformation of connectivity into networking requires solving the challenges of system-level wide security, ease of installation, low latency, and standardization of the application languages for lighting, climate control, security, care, retail, etc.

Another implemented low power IEEE 802.15.4 based networking protocol is Google's Thread. Respect to BLE, such protocol is IPv6-based, that is has IPv6-based low power wireless PAN (6LoPAN) [35] as the foundation, thus allowing to exploit the features of IPv6 protocol. Google Thread was designed to connect and control products in the home, but with application in other context.

The development of these wireless standards is connected to the market of smart cities and home as well as that of wearable devices, especially in the healthcare sector, or the cultural and creative industries. A research of ABI on mobile health [36], forecasts that Bluetooth, Wi-Fi, and other standardized wireless protocols are set to trump proprietary wireless protocols in the field of mobile medical devices by 2018, because of the need to integrate remote health-reporting systems with familiar electronics. The same research forecasts the use of standardized alternatives (projected to grow by 72% in the next five years), especially those based on IEEE 802.15.4 standard with respect to proprietary wireless solutions. The healthcare device integrated circuit (IC) market, after adopting BLE and Wi-Fi, is expected to exceed $100 million. Moreover, the adoption of standardized wireless protocols in the medical industry represents another step towards the protocols' convergence. In this area, devices are also equipped with IEEE 802.15.6 connectivity that are for MBAN, though with a slower market penetration. The recent IEEE 802.15.6 and IEEE 802.15.4 enabled devices that can operate in the dedicated wireless spectrum (2360-2400 MHz) allocated for MBAN, could improve a further development, especially in addition to with the other protocols (e.g., Bluetooth). Connection protocols can enable MBAN wireless sensor data and provide connectivity through gateway devices to cloud-based healthcare services, by using customized mobile apps using such protocols.

Similar statistics could be applied in the field of cultural and creative industries or organization, where a more engagement of short-range connectivity in museum for example with the use of the beacon technology [37] based on Bluetooth is shown.

As regards the networking connectivity, emerging Wi-Fi standards such as 802.11ac (Waves 1 and 2), 11ad, 11aq and 11ah will increase Wi-Fi performance, and make Wi-Fi more relevant to apps in providing new services. Over the next three years, demands on Wi-Fi infrastructure will increase as more Wi-Fi-enabled devices begin to appear in organizations, with specific apps using these protocols to provide services. Finally, the networking functionality, like the broadband access, is related to the efficiency of cellular network.

Long Term Evolution (LTE) and its successor LTE Advanced (LTE-A) are cellular technologies that improve spectral efficiency in order to reach peak downlink speeds of up to 1 Gbps, while reducing latency. In this way, cellular networks could gain the same throughput of wired networks that are based on fibre optics. Hiding users, the past network latency, the Internet connection is guarantee and mobile web app can provide the same experience in terms of speed and interaction as locally installed app. The use of such protocols both for connectivity and networking along with their enhanced performance and improved bandwidth, will enable network operators to offer new services.

3. Agile methodologies on mobile software developments

Advances in mobile computer technology as regard the hardware, the layered software ecosystem, and the rapid growth of quality and quantity of wireless network, has introduced new software development methodologies and concerns. In addition, the requirements and constraints associated with mobile systems bring new challenges to software development that could need other development methodologies to fulfill the special needs. Traditional software development methods, that are today not suited to desktop-based software development, are inflexible and fail to respond on aggressive customer requests, and the velocity in providing software releases. Agile software methodologies provide a set of practices, frameworks and approaches, that, even if declined with different names (e.g., Scrum [38], eXtreme Programming [39]) introduce a different software cycle development and management. Usually, small, collocated teams are responsible for the release of the product that is developed in a number of incremental and interactive iterations requiring the involvement of all teams. These approaches allow for quick adaptations, and are applied in the different life-cycle phases through several methodologies (Fig. 6). In a mobile environment, developers face the challenge of a dynamic environment, with frequent modifications in customer needs and expectations together with technological
constraints

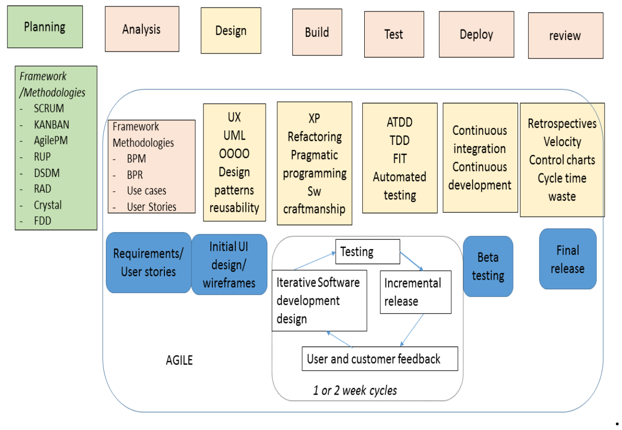

Fig. 6 Phases in software development and example of agile frameworks and approaches

These are related to hardware features (e.g., limited physical resources, rapidly changing specifications, and a variety of devices, firmware and operating systems), evolving capabilities (bandwidth, coverage and security), and inherent, permanent constraints (e.g., limited screen real estate, reduced data entry capability, memory capacity, processing power, and limited power reserve). Accordingly, the apps are small in size, are not safety-critical, and do not have to satisfy interoperability or reliability constraints. They are delivered in rapid releases, in order to meet market demands and are targeted at a large number of end-user.

The development environment and the supporting technologies should guarantee high-level of competitiveness, short time-to-delivery, identification of stakeholders and their requirements, and extreme usability, in spite of lower revenues compared to traditional software development, since the distribution in the app stores must follow the policies of these business models. These are the variables to be considered in the development and deployment lifecycle. Analysts [40] state that traditional practices used to define and develop desktop apps will not work for mobile application development due to device diversity, network connectivity and the specificity of mobile devices. The Agile methodology is defined in literature [41] as a method that could help developers to overcome these challenges. However, there is little evidence in literature about its successfully adoption [42].

3.1 The Agile approach proposed for mobile

Fundamental concepts to agile development are [43] include simple design principles, a large number of releases in a short timeframe, extensive use of refactoring, pair programming, test-driven development, and seeing change as an advantage. Among the several definition, an Agile development method is incremental (i.e., multiple releases), cooperative (i.e., a strong cooperation between developer and client), straightforward (i.e., easy to understand and modify) and adaptive (i.e., allowing for frequent changes).

The use of agile methods, in the different approaches as shown in Fig. 7, has received both support and opposition. Anyhow, an Agile approach that follows one of the different methodologies, of which Scrum seems to be the more used, should be customized in the context. As stated in the Fig 7, Scrum or Scrum variants are more than two-thirds of the methodologies being used.

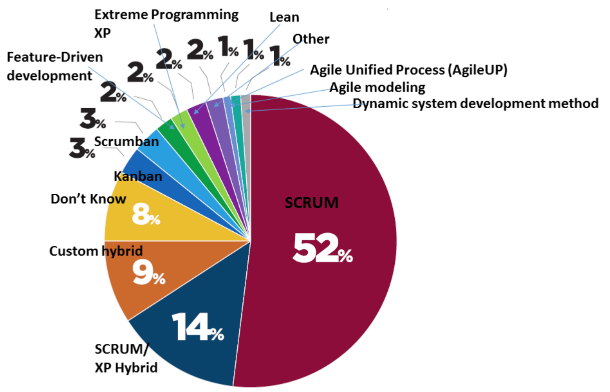

Fig. 7 Statistics on agile methodology used

Mobile app development requires frequent changes and need to be revised often to meet end-user expectations. Another important aspect is the design of an optimal users interface that meet users' expectations in terms of usability, interaction and simplicity, tanking into consideration the lack of a pointing device such a mouse.

The experience associated with mobile devices is significantly different from that with desktop devices, even if the user's requirements are the same, such as shorter session lengths and limited presentation. Most complaints about mobile apps have to do with poor user experience. This can be due to poor user interface design, poor application workflow or poor responsiveness. In literature, there are some modified Agile approaches applied to mobile development (Fig. 8). The Mobile-D approach [44] has been the first methodology proposed. It defines nine major elements involved in the different practices throughout the development cycle (i.e., Phasing and Placing; architecture Line, mobile Test-Driven Development, continuous Integration, pair programming, metrics; agile software Process Improvement; Off-Site Customer; User-Centred Focus). According the authors, its application in development projects has some advantages such as increased progress visibility, earlier discovery and repair of technical issues, low defect density in the final product, and a constant progress in development [45].

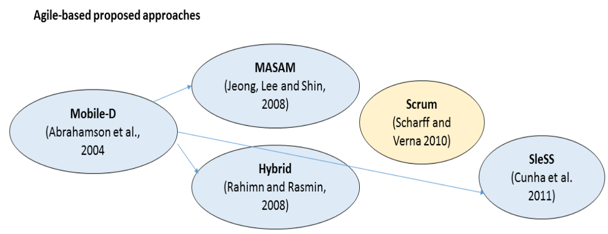

Fig. 8 Agile mobile approaches developed

From this approach, some other methods have been derived (e.g, the Hybrid Methodology Design (HMD) approach [45]), or it is reported the application of the classical Scrum framework. More research should be undertaken in order to understand the applicability of these approaches.

4. Conclusions

Mobile apps are a new category of software that are nowadays developed to meet user expectations. However, this kind of software manifest constraints and challenges typical of the mobile context and related to hardware, software and networks implications. Users' requirement both from a usability point of view and related to the business model underpinning the delivery, impose other consideration that affect all the life-cycle of software development. For example, mobile apps need to be frequently revised to meet end-user expectations, and interaction with users is made through a touch interface that affects input requirements. The number of mobile device types further complicates mobile app development and operations efforts, because the range of device screen sizes, resolutions, hardware API access and performance is fragmented and changes rapidly in technology. Software developers should in some way change their usual approach in development methodology, for example by embracing Agile approach that could help to address some of the features related to mobile environments. The issue is that there are no currently approaches that are considered universally valid for mobile environment. On the other end, from a commercially point of view, app development is not so profitably, because distribution takes place through app stores especially in the case of non-specialized apps. Moreover, the short cycle of development does not help in choosing a methodology that however implies knowledge and effort in its implementation.

References

[1] I. Sommerville, Software Engineering, 9th edition, Addison-Wesley, 2011.
[2] M.J. Kavis, Architecting the cloud: design decisions for cloud computing service models (SaaS, PaaS and IaaS), Wiley,2014.
[3] D. Nield, “Making sense of sensors: what you don’t know your phone knows about you,”Techradar.com, April, 2014. Available at url: http://www.techradar.com/news/phone-and-communications/mobile-phones/sensory-overload-how-your-smartphone-is-becoming-part-of-you-1210244
[4] M. Weigel, T. Lu, G. Bailly, A. Oulasvirta, C. Majidi and J. Steimle, “iSkin: Flexible, stretchable and visually customizable on-body touch sensors for mobile computing, ACM CHI 2015, Seul Korea, April 18-23, 2015.
[5] R.C. Martin, Agile Software Development, Principles, Patterns and Practices, 1st edition, Pearson, 2002
[6] J. Highsmith, Agile Project management: creating innovative products, Pearson education, 2009.
[7] s. Matos, E. Lopes, Prince2 or PMBOK – a question of choice, Procedia Technology 9, Elsevier, 2013, pp.787-794
[8] J. Pernstal, R. Feldt, T. Gorscheck, The lean gat: a review of lean approaches to large-scale software system development, Jorunal of System and Software 86, Elsevier, 2013, pp. 2797-2821
[9] S. Pastore, Website development and web standards in the ubiquitous world: where are we going?, WSEAS Transactions on Computers, Issue 9, Volume 11, September 2012. pp.309-318. E-ISSN: 224-2872
[10] B. S., Gardner, Responsive web design. Enriching the user experience”. Sigma Journal: Inside the Digital Ecosystem 11.1, 2011, pp.13-19.
[11] A. Kostick, The digital reading experience: learning from interaction design and UX-Usability experts, Publishing Research Quaterly, Vol. 27, Issue 2, Springer 2011, pp.135-140
[12] A. Charland and B. Leroux, Mobile application development: web vs. native. Communications of the ACM 54.5, 2011, pp. 49-53
[13] "Top 10 Mobile Technologies and Capabilities for 2015 and 2016," available on Gartner's website at http://www.gartner.com/doc/2665315
[14] M. Pilgrim, HTML5-Up and Running: Dive Into the Future of Web Development, 2010 - O'Reilly
[15] Saleforce, 2014 Mobile behavior report, http://www.exacttarget.com/sites/exacttarget/files/deliverables/etmc-2014mobilebehaviorreport.pdf
[16]
[17] IEEE website, IEEE 802.11: Wireless LANs, available at url: http://standards.ieee.org/about/get/802/802.11.html
[18] Bluetooth Special Interest Group website, https://www.bluetooth.org
[19] Near Field Communication, NFC forum, http://nfc-forum.org.
[20] M. Massoth, T. Bingel, Performance of different mobile payment service concepts compared with a NFC-based solution, Proc of 4th Inter. Conference on Internet and web applications and service, IEEE publishing, 2009, pp.205-210.
[21] R. Nibarger and M. Teubner, An Introduction to Long term Evolution (LTE), Issue brief, COPS, www.cops.usdoj.gov, Aug. 2012.
[22] IEEE 802.15: Wireless Personal Area Networks (PANs), IEEE website, https://standards.ieee.org/about/get/802/802.15.html
[23] H. Chourabi, Taewoo Nam, S. Walker, J.R. Gil-Garcia et al. Understanding smart cities: an integrative framework, Int. Conf. on System Science (HICSS), 2012, pp.2289-2297
[24] T. Bradshaw, Apple wins patent for wireless head-mounted display using iPhone, FT.com, technologies, 2014, http://www.ft.com/intl/cms/s/0/6943ebce-b6e2-11e4-a33b-00144feab7de.html#axzz3hkCDecZU
[25] M. Loukides & J. Bruner, What is the Internet of things? O’reilly 2015
[26] C. Gomez, J. Oller and J. Paradells, Overview and evaluation of Bluetooth low energy: an emerging low-power wireless technology, Sensors, 2012, 12(9), 11734-11753 doi:10.3390/s120911734
[27] IEEE 802.15.4 -2011, IEEE standard for local and metropolitan area networks, Part 15.4: Low-rate wireless personal area networks (LR-WPANs), https://standards.ieee.org/getieee802/download/802.15.4-2011.pdf
[28] IEEE 802.15.6 IEEE Standard for Local and metropolitan area networks - Part 15.6: Wireless Body Area Networkshttps://standards.ieee.org/getieee802/download/802
[29] G.V. Crosby, T. Ghosh, R. Murimi, C.A. Chin, Wireless body area networks for healthcare: a survey, International Journal of ad hoc, sensor and ubiquitous computing (IJASUC), Vol. 3, N. 3, June 2012
[30] M. Alleven, Zigbee Alliance aims to unify with Zigbee 3.0 for easier product development, adoption, FierceWirelessTech, Nov., 2014, at url: http://www.fiercewireless.com/tech/story/zigbee-alliance-aims-unify-zigbee-30-easier-product-development-adoption/2014-11-20
[31] Google Thread, at url: threadgroup.org
[32] D. Yang, Y. Xu, and M. Gidlind, Wireless coexistence between IEEE 802.11 and IEEE-802.15.4-based networks: a survey, International Journal of Distributed Sensor networks,

Vol. 2011, Article ID 912152, 17 pages, 2011. Doi:10.11.55/2011/912152
[33] M. Kuzlu, M. Pipattanasomporn and S. Rahman, Communication network requirements for major smart grid applications in HAN, NAN and WAN, Computer Networks 67, Elsevier, 2014, pp. 74-88.
[34] B. Gohn and C. Wheelock, , Executive summary: smart grid networking and communications. WAN, NAN and HAN communications, PIKE research report, 2010.
[35] B. M. Dorge, and T. Scheffler. "Using IPv6 and 6LoWPAN for home automation networks." *Consumer Electronics-Berlin (ICCE-Berlin), 2011 IEEE International Conference on*. IEEE, 2011.
[36] ABI research, mHealth wearables, platforms and services, 2015, at url: https://www.abiresearch.com/market-research/service/mhealth/
[37] Ho, Tsung-Yuan, et al. "BeaconPass: A Location Based APP Game for Traveler." *Universal Access in Human-Computer Interaction. Access to Today's Technologies*. Springer International Publishing, 2015. 288-297.
[38] J. Sutherland, et al. "The scrum papers: Nuts, bolts, and origins of an agile process.", 2007.
[39] Beck, Kent. Extreme programming explained: embrace change. Addison-Wesley Professional, 2000.
[40] C. Lee, Gartner says traditional development practices will fail for mobile apps, Gartner, News, 2014, http://www.gartner.com/newsroom/id/2823619
[21] A. Krishnan ABI research Demand for smart home technology to skyrocket, 2014 SecurityWacthInfo, at url: http://www.securityinfowatch.com/news/11338064/abi-research-predicts-that-more-than-500-billion-wireless-smart-home-monitoring-devices-will-be-installed-worldwide-by-2018
[41] Ilieva, Sylvia, Penko Ivanov, and Eliza Stefanova. "Analyses of an agile methodology implementation." *Euromicro Conference, 2004. Proceedings. 30th*. IEEE, 2004.
[42] L.Corral, A.Silitti, G. Succi, Software development processes for mobile systems: is Agile really taking over the business?, 1st Workshop on Engineering of Mobile-enabled systems MOBS, May 25, 2013
[43] B. Boehm, R. Turner. Balancing Agility and discipline: a guide for the perplexed. Elsevier, 2003)
[44] Abrahamsson, Pekka, et al. "Mobile-D: an agile approach for mobile application development." *Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications*. ACM, 2004.
[45] Rahimian, Vahid, and Raman Ramsin. "Designing an agile methodology for mobile software development: a hybrid method engineering approach."*Research Challenges in Information Science, 2008. RCIS 2008. Second International Conference on*. IEEE, 2008.

Serena Pastore is an electronic engineer working since 1998 as ICT researcher at the National Institute of Astrophysics (INAF). Her interests are distributing computing, web and Internet standards and the open source ecosystem from hardware to software. Moreover, she is involved as EU project consultant.