

Modeling the Mobile Application Development Lifecycle

Tejas Vithani, Member, IAENG and Anand Kumar

Abstract—Software Development Lifecycle is crucial in Desktop or web application development. Mobile applications are being developed for deployment in smart phones. Looking at the rising need of mobile applications and the associated development complexity, it is imperative to have a dedicated framework lifecycle for mobile application - Mobile Application Development Lifecycle (MADLC). The proposed lifecycle includes the following phases: Identification, Design, Development, Prototyping, Testing and Maintenance. This lifecycle has been used for over a year in developing Android mobile applications. This lifecycle addresses some of the distinguishing characteristics of mobile applications like life span, complex functionalities, fewer physical interfaces, more number of screens for interaction, battery and memory usage, cross platform development and maintenance. The preliminary indicators are that this will benefit the mobile developers to follow the development process within MADLC to build robust and optimal control applications.

Index Terms—Mobile Application Development, Mobile Development Lifecycle, Mobile SDLC, MADLC.

I. INTRODUCTION

A Software development lifecycle process is a type of structure or framework used in the development of any software product. There are many different lifecycle models defined. Waterfall model, spiral model, prototyping model are a few such models. Each model is described by a sequence of activities. The development steps or the activities may vary in each and every model but all the models will include planning, requirement, analysis, design etc. The waterfall model emphasizes more on the step-by-step process. The spiral model emphasizes on risk assessment while the prototyping model takes an incremental approach in each and every phase of the development process. [1, 2]

Designing the product to fulfill the requirements of the user based on the different standard development lifecycle models is a crucial aspect of the development process. A *pre-defined* development process is of essence. The development lifecycle might be impacted if the development occurs incrementally. If the customer is engaged in the

Manuscript received December, 23 2013; revised February, 5 2014.

T. Vithani is a student pursuing M.E (Software System) with Computer Science Department, Birla Institute of Technology and Science Pilani, Dubai Campus, Dubai, U.A.E (email: tejas.vithani@gmail.com).

A. Kumar is Associate Professor with the Electrical & Electronics Department, Birla Institute of Technology and Science Pilani, Dubai Campus, U.A.E (email: akumar@dubai.bits-pilani.ac.in).

development process, customer requirements are better implemented by the end of the development process leading to better customer satisfaction. [1, 2]

Waterfall model, Spiral model, Iterative model and Prototyping model are some of the software development lifecycle models widely used in the industry. There is need to have a distinct mobile application development lifecycle model: mobile application development involves various complex functionality and services like telephony services, location based services and different connectivity modes [2].

II. LITRATURE SURVEY

A. Desktop Software Application Development Process

Development of any desktop application involves similar phases of any SDLC [2] irrespective of technology being used. The output of each and every phase in SDLC feeds into the next phase. For instance, the requirements gathered during the planning phase are analyzed in the next phase. These refined requirements are then designed in the *Design Phase*. The design is then implemented in the *Implementation Phase*. The code generated during the implementation phase is verified and tested. Customer support is provided for the designed and deployed application.

B. Mobile Application Development Process

Intel IT has created a mobile application development framework: this framework defines specific activities, tools and resources, to support the planning, development and deployment of mobile applications [3]. Intel has a two-step process defined for developing a mobile application.

- Use a pre defined decision matrix; to verify whether the application is suitable for mobile development.
- Determine the process to be used in delivering the application.

Slalom Consulting, have defined the mobile application development lifecycle as “Enterprise Mobile Application Lifecycle”: this process is for an end-to-end mobile application development [4]. They have defined the lifecycle in the following four different phases:

- Discovery
- Design

- Development/Testing
- Deployment

Initially the mobile applications were only developed to implement calculator, calendar, alarm and currency converter functionalities. With the advent of 2G and 3G mobile networks, web based *mobile* applications were implemented on a variety of platforms; many of the existing web based applications were ported to platforms on the mobile device. These web-based applications included social networking, blogging, sharing of multimedia such as music, photos, video over the web [5].

Mobile applications can be classified into five major categories [6]: Social Networking, Personal Productivity, Leisure-based, Transaction-Based, and Content Dissemination-Based.

Mobile application development has received a boost with two major developments. One development pertains to the availability of increased network bandwidth going from 2G to 3G and now to LTE. The second significant development is on the mobile device side: large screen, increased memory and high speed processing capability [6]. Developers are motivated by these technological advances to create more innovative applications and services.

The differences between PC application development and mobile application development are discussed below.

Life Span

The mobile application is developed to implement one or few but limited functionality. Java, Objective-C, .net and similar traditional development frameworks are used. Due to these factors, the number of mobile application developers is large. Given this intense competition in mobile application space, the life span of mobile application is less as compared to that of any desktop application.

Complex Functionalities

Mobile application development involves complex functionality to interface to telephony, camera, GPS etc. When desktop application are considered the development of the application is much more restricted to the desktop or the laptop device. On the other hand, mobile applications may require the implementation of telephony functionality to send/receive messages or handle voice calls; the application may also require the implementation of GPS functionality to track location by fetching latitude and longitude data.

Fewer physical interfaces

Desktop laptop have different physical interfaces such as keyboard, mouse, touch panel and other external devices but the mobile physical interfaces are strictly restricted to touch panel or the mobile keyboard. The size of the touch panel and keyboard is smaller than that of the laptop. These differences should be considered in the design of mobile applications to make them user friendly despite fewer physical interfaces.

More Number of screens for Interaction

The large screen on a desktop or laptop permits more functionality to be designed in one screen. With a smaller screen size being available, the mobile application requires more number of layered screens to support a similar functionality as a single large screen. Furthermore, the mobile application developer should optimize the functionality design to maximally utilize the screen available for display.

Battery & Memory usage by the application

Optimal usage of the battery and memory is a major design consideration in developing mobile applications: the mobile phone is on all the day round with many applications and services being constantly executed. The mobile applications should be designed to optimally utilize the memory available. The idle processes running on a desktop or laptop would not consume a significant percentage of the available battery or memory in a desktop or laptop as do the telephony applications on the mobile device.

Cross Platform Development

Many mobile operating system platforms share the market rather equally; developing a mobile application on one platform will not suffice in maximizing access to users. A key factor in choosing mobile application features is its portability to all major platforms. For example, the Android platform is open source making it developer-friendly. But other mobile platforms like iOS, Windows, BlackBerry are proprietary: they are closed and restrict developers' access to internals and thereby limiting features that can be implemented on the platform. The limitation does not exist in the same magnitude in desktop and laptop application development.

Updates

With desktop or laptop applications often sold to the user on a CD, the user registration is not tied to the sale and is optional. Mobile applications are more often than not downloaded from the application store; user registration goes hand in hand with the download. Hence, it is easier to provide updates for mobile applications than for desktop or laptop applications.

III. MOBILE APPLICATION DEVELOPMENT LIFECYCLE MODEL (MADLC)

As the mobile applications have complex functionality and are different from the desktop applications, the following Mobile Application Development Lifecycle model (MADLC) is proposed to enable a systematic approach in development.

A. Identification Phase

In the first phase, ideas are collected and categorized. The main objective of this phase is to come out with a

new idea or improvements to the existing application. The ideas can come from the customer or from the developers.

If the customer himself comes out with the idea, the idea is further detailed and analyzed. Developers can brainstorm to generate ideas for new applications. The filtered list of ideas is discussed by the mobile application idea team comprising of the business and IT representatives for the feasibility to launch a project around the idea.

The existing applications on any of the standards platforms are searched to establish the novelty of the idea. If a similar application exists in the market, the popularity of the application and the features supported are studied and compared. The differences with the existing application(s) are documented. If no similar application exists on any mobile platform, then the idea with its core functionality should be documented. The other important task in this phase is to define the time required to develop the application. The initial requirement gathering should also be completed. The work done by the mobile application idea team should then be documented and forwarded to the design team.

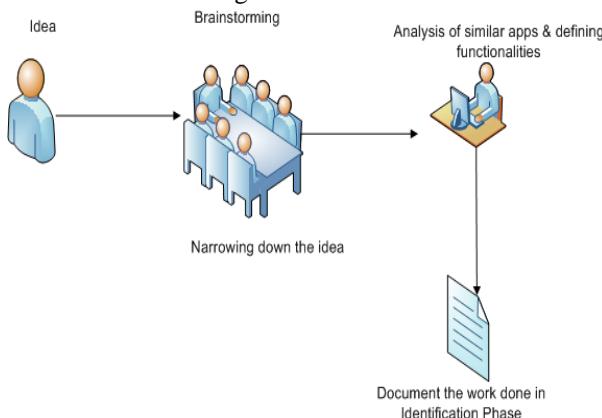


Fig 1. Identification Phase

B. Design Phase

In this phase, the idea from the mobile application team is developed into an initial design of the application.

The feasibility of developing the application on all mobile platform is determined. Alternatively, the specific target mobile platform is identified. A decision has to be made on whether the developed application is to be released as a free version or trial version with limited features or released only as a premium version. The application functionality is broken down into modules and into prototypes i.e., combination of modules which are to be released in the prototype fashion. The functional requirements are defined. The software architecture of the application is created. Then the prototypes and associated modules are defined. A very important part of the design phase is to create the storyboard for the user interface interaction: this storyboard describes the flow of the application. The design team's work is documented and forwarded to the development team for coding.

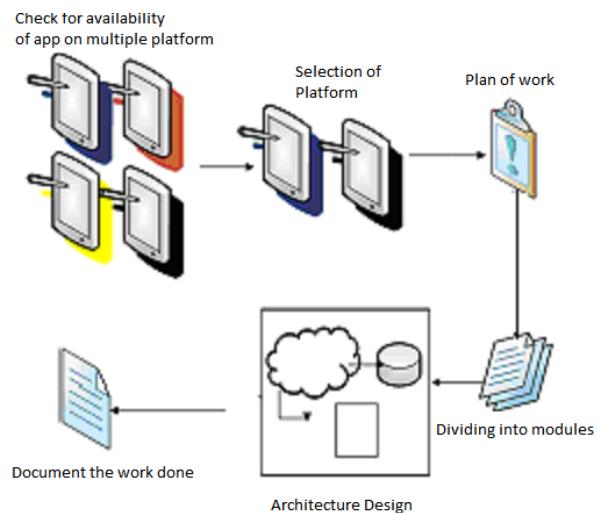


Fig 2. Design Phase

C. Development Phase

In this phase, the application is coded. Coding for different modules of the same prototype can proceed in parallel. The development process can be in two stages: Coding for Functional Requirement and Coding for UI requirements. The code is developed first for the core functionalities. Parallel development can be done for modules of the same prototype that are independent of each other. Subsequently, these modules can be integrated. In the second stage, user interface is designed so that it can be supported on as many mobile operating system platforms as possible; it is not good practice to have a different look and feel for the same application on different platforms. The minimum set of interface components present in all mobile OS platforms should be used in the design. Finally, the documentation of the development phase is then forwarded to the prototyping phase.

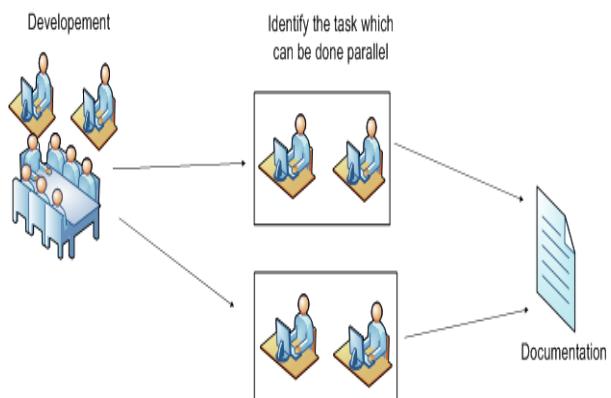
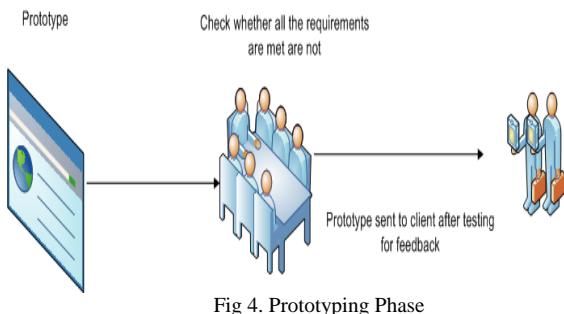


Fig 3. Development Phase

D. . Prototyping Phase

In this phase, the functional requirements of each prototype are analyzed; the prototypes are tested and sent to the client for feedback. After feedback is received from the client, the required changes are implemented through

the development phase. When the second prototype is ready, it is integrated with the first prototype, tested and then sent to the client. The development, prototyping and testing phases are repeated until the final prototype is ready. The final prototype is sent to the client for a final feedback. The work done in this prototyping phase is documented and then forwarded to the testing phase.



E. Testing Phase

Testing is one of the most important phases of any development lifecycle model. The testing of the prototype types is performed on an emulator/simulator followed by testing on the real device. The emulator/simulator is often provided in the SDK. The testing on the real device, for example in the case of Android operating system development, should be performed on multiple operating system versions, multiple models of handsets with variable screen size. The test cases are documented and forwarded to the client for feedback.



Fig 5. Testing Phase

F. Deployment Phase

Deployment is the final phase of the development process. After the testing is completed and the final feedback is obtained from the client, the application is ready for the deployment. The application is uploaded to the appropriate application store/market for user consumption. Before the application is deployed, the following steps are to be checked.

- Register as a developer on the respective application developer's website by paying the annual fee, if necessary, for respective OS.
- Check the rules and regulations of the application store for the deployment of an application.
- Refine the application, for example, by removing all the log files and comments.

- Design the icon and wallpaper to be used on the application store.
- Create the file format required on operating system platform.



Fig 6. Deployment Phase

G. Maintenance Phase

The maintenance is the final phase of this model and this maintenance is a continuous process. Feedback is collected from users and required changes are made in the form of bug fixes or improvements. Appropriate security patches, performances improvements, additional functionality, new user interfaces should be provided at regular intervals in the form of updates to the application. The maintenance phase also includes the marketing of the application: advertising and highlighting its unique features. If any application requires a backend server: this server and related operating system must be maintained as well.

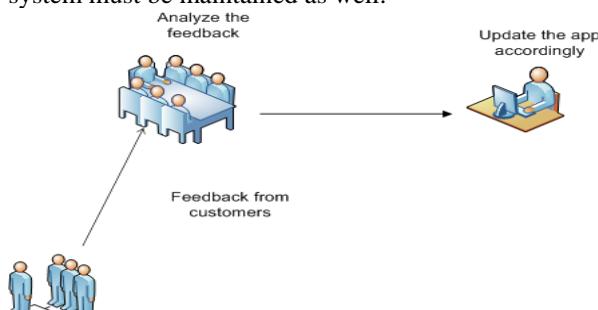


Fig 7. Maintenance Phase

IV. PRELIMINARY RESULTS OF APPLYING THE MADLC

An industry-oriented application was developed using MADLC model. This application includes use of GPS locating service, telephony services like sending SMS, configuring device profile. The development of this application development provided a test case for the proposed MADLC. All the functional requirements of the application were divided into various modules and delivered in a prototype fashion to the organization at various intervals. The outcome of the development process is that this model helped the developers to plan and execute the application involving various features in a more effective manner.

V. CONCLUSION

A Mobile Application Development Lifecycle (MADLC) has been proposed to bring out a formal lifecycle for mobile application development. MADLC will aid the mobile application developers in developing high-end apps. This lifecycle includes the following phases: Identification,

Design, Development, Prototyping, Testing and Maintenance. The different tasks and activities in the various phases of MADLC have been discussed. This lifecycle also addresses some of the distinguishing characteristics of mobile applications like life span, complex functionalities, fewer physical interfaces, more number of screens for interaction, battery and memory usage, cross platform development and maintenance. Preliminary testing of the MADLC indicates that this lifecycle will help developers and project managers in efficiently execute projects and deliver solutions on time.

REFERENCES

- [1] Dr. Drrpshikha Jamwal "Analysis of software Development Models", ISSN: 2229-4333 (print), ISSN:09768491 (online), vol. 1 ISSUE 2, December 2010.
- [2] Ian Sommerville, Software Engineering, 8th edition.
- [3] IT Intel White Paper, "Building a Mobile Application Development Framework", August 2012.
- [4] Daniel Mavcock, Slalom Consulting, "Enterprise Mobile Application Lifecycle."
- [5] State of the Mobile Web Opera, April 2008.
- [6] Gasimov, Anar, et al. "Visiting mobile application development: What, how and where." Mobile Business and 2010 Ninth Global Mobility Roundtable (ICMB-GMR), 2010 Ninth International Conference on. IEEE, 2010.