Psi (Ψ) Model For Handheld Application Development

Shrish Bajpai & Sanjana Tiwari

Abstract— **Growing demand of different electronics portable handheld devices, development of applications for these devices is expanding in manifolds. A large number of handheld applications are develop with the use of code or component of existing application. This will save application development cost & increase productivity of the application development team. To enhance the reusability features in the development handheld applications, a new process model is required which is useful for team of one member to several member team. Psi (Ψ) Model is proposed for the development of handheld application which lays emphasis on domain engineering, cost estimation, time management & risk analysis in every major phase to improve the quality and to reduce the time & cost. We can schedule test activities after the change detection & requirements has been freeze.**

Index Terms—**About four key words or phrases in alphabetical order, separated by commas.**

I. INTRODUCTION

Software industry in present scenario have a enormous pressure for meeting the deadline of the software products with minimum workforce, development cost & time. Reusability of software components or code or module is an important prerequisite for cost and time-optimized software development. A lot of software companies are adopting component based software development processes to meet the demands of customers to deliver the product with changing requirement, minimum time frame and at lowest cost[2,5,6]. Component based software development emerged in the late 1990 as a reuse based approach to software development**.** It's principle is to reuse already completed components or code instead of developing everything from the beginning of the development of a new software. But the main constraint in this approach is that all software engineers working in the team have domain knowledge about the application.

Growing demand of different handheld devices including portable electronics devices, a large number of software application has been develop with the component software development approach[1]. Now a day's people do not want to keep one handheld for a long time & changes their handheld with their requirements. With development new 3G or other more advance telecommunication networks, it generates a demand of compactable handheld device which is more focus on cellular data transfer. Since handheld manufacturing companies launching their products in market with a short spam of time having minor change in the hardware of handheld & importing new applications to the handheld[4,7]. Now software companies are focusing to make these handheld application with the desktop environment, so that it should be install with the more & more digital devices. Handheld application development teams are developing application with the help of existing software components. There are many software design models but they do not address the issues related to application development specifically & more over they have too many steps that makes the development of application complicated. Now there is demand to develop a new software design model based on component based software development for the developing the different application for handhelds, electronics equipments & mini computers software which is more focus on the development of application[3].

Present model is effective for small project or developing a handheld application which is develop by the individual or a very small team (upto seven member). It is also good for the short term projects or a independent module in software which need to be integrated to the software system and only have minor changes. The main advantage of using this software model is that it is simple, number of stages are reduce with respect to other component development models & easy to applicable on software or application development.

II. PSI (Ψ) MODEL

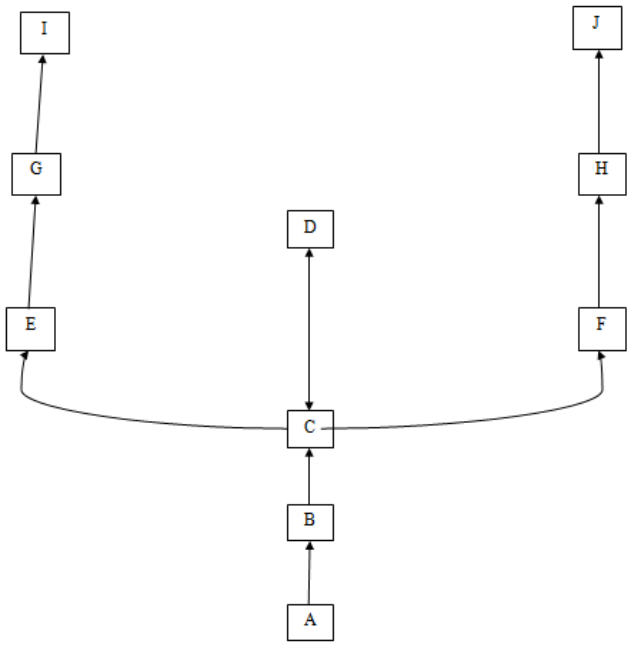
Market of handheld devices are growing very rapidly now a days, development or updation or extension or enhancement of any existing application with the component base development is advisable [6,7]. Present model is useful for both, a new application development, updation or enhancement or designing a new add on feature of any existing handheld application & change detection of any existing feature. This model is useful for independent software engineer to application development team. Proposed model have six stages in which three stages are common to development & testing process and last three stages of development activities are monitored by dedicated testing activities as shown in the FIG 1.

Initial two stages of the presented model are common to both process because they includes the documentation &

Manuscript received January 14, 2015

Shrish Bajpai, Electronics & Communication Engineering Department, Integral University, Lucknow, India.

Sanjana Tiwari, Electronics & Communication Engineering Department, Integral University, Lucknow, India.



A. Requirement & Change Detection Analysis
B. System & Test System Specification Analysis
C. Code & Test cases/scripts Development Outline
D. Repository or Database
E. Module Development
F. Integration Testing
G. System Development
H. System Testing
I. Operational System
J. User Interface Level Testing

FIG 1 : PSI (Ψ) model with different stages for handheld software development

Functional testing process, regression testing issues and how many different hardware configuration this application has to deploy is discuss in this current phase. This initial phase is generally handle by the high level team members & major outcomes will discuss by higher official to the joiner team members in the later stages according to their work role & situation of the application development project. Systems specification phase is handle by the software & test architecture who has a the domain knowledge. The system requirement phase demands the software & test architecture to study the whole application with it's constraints, understand the major features of the application, understand the requirements expected to be satisfied by the software system (operating system & hardware configuration of device), how much reusable features or component or test cases or test scripts can be develop & generate an abstract model of the application in which all requirements are met. The major outcome of this phase is a graphical or textual description of a abstract model of the application under development. At this phase, the services delivered by a software system help figure out its subsystems and major components. In this phase how to develop the application & how much testing (black box & white box testing ; functional & non functions ; manual & automation ; total number of regression cycles ; code coverage) is needed for the application queries are solve. Third stage is dedicated to the fetching the code or test cases or test scripts from connected repository. Reusable code or components of the application which develop from the existing code should be define in this phase by the development team as per requirement given by the software architecture. Similarly QA team will take the associated test cases or test scripts from the connected repository & modify it according to the line of the requirments.

After from here, application development has been divided into two paths, development path & QA path. In development path first phase is module development in which modules have been develop with the help of existing code or if code is not available then a new module will develop by the team. QA team in integration testing phase, use the modified exiting test cases or test scripts if module is develop with the existing code & also check that developed module it should be comfortable with the target hardware & operating system. In this phase it has to be check that the develop module should be functionally stable in the target hardware & operating system. In the next phase of development path, all modules or features get integrated & creates a complete system or application. This system or application should be compatible with the operating system & other present (install) application in the target hardware configuration. Functional & non functional issues such that performance, stress, load, reliability, usability, security, configuration, compatibility & recovery issues are covered in the system testing phase of the QA path. QA team have to perform black box testing to test the application in the system testing. For non functional testing, selection of appropriate tool is needed because if choice of appropriate tool or non functional testing does not perform well then certain issue such that device hanging or device not properly responding will may be occur. Now final phase of the development path in psi model is operational system in which development team have to develop the application for operating system or different languages (localization issues). This can be achieve with small change in the application code. QA team in the final phase of the QA path, have to test application on the shoe of the end users or customers. In this phase localization testing has to be done. If the application is associated with the network end then field testing is necessary which is done by specialist QA team for telecommunication network testing to test network related issues such that network coverage, mobility, signal strength etc issues.

III. AN EXAMPLE

Let take a handheld application "Birthday Alert". It is the application which is use to store the birthday of the persons whose number is store in the handheld memory. At the time of adding a new contact, it ask user to store the date of birth of contact by default. User can also go to edit feature & put the date of birth for that contact. This application works with the calendar & time of handheld, for giving the birthday alert alarm (default time set by user).

This application "Birthday Alert" has to redevelop with adding some new feature under the guidance of proposed Psi (Ψ) model with existing code. In the adding new features which are sending the automatically default message to the contact having birthday through short message services or any internet message services, time set by user. For development

of the new application "Birthday Wishes" from the existing "Birthday Alert" application. First all requirements, hardware configuration & operating system of the target handheld, programming environment use to develop application, size of the application, it is need to check that this application should be compatible with the other applications or features going to present in handheld & analysis and test requirement for testing of this application have to be gather. In system & test system specification analysis phase two different abstract model of whole development & testing process of application is designed by the respective architecture. After the clear requirements & project flow, reusable code, component, respective test case & test scripts have be fetch from the connected repository. After then each development activity is mirror by testing activity. Individual feature or module is develop in the module development phase. If feature or some component of feature is present in database then it should be reused. It's corresponding mirror activity is integration testing in which testing process has been more focused on one feature & functional testing should be focused more. Non functional testing issues generally not covered in this phase. All features or module of the Birthday Wishes get integrated in the system development phase. Other hardware, operating system & other install application compatibility issues should be resolved in this phase. In system testing phase, functional test cases associated to two or more features should be execute with non functional testing. Localization, documentation & internationalization issues which is more related to the language of the target region with the network (bandwidth) related issues if any, are the major focus in operational system phase. It's corresponding mirror activity is user interface level testing, α & β level testing has been done on the shoe of target customer with the field testing for network related test case execution.

IV. CONCLUSION

Present psi model can be use for the all type of handheld applications as well as development of any other product which is developing with the component base software development approach from reuse code of existing software. Psi model has a clear requirement specification or change detection, understanding requirement, low cost, effective recourse utilization & no overlapping issues in different phases. This model have many advantages. This models works from single member team to the many member team. This model has some shortcoming such that domain knowledge person is needed, code & test cases should be available for the modification or updation of existing application.

During the QA phases, the developer or development team is responsible for the removal of defects and the correction of the implementation in the next regression to move the lifecycle ahead. The early collaboration and the tight co-operation between the development & QA team can often in practice avoid conflict meetings. In application development the most important aspects may vary or change and so therefore the resource allocation is unlikely to be equal across all activity. For highly critical applications such that banking application or e-commerce application or security application, the test activities certainly have higher weighting or at least equal weighting with other activities.

REFERENCE

[1] Jon Peddie, "Handheld Market Enters a New Phase: Multimedia", "https://www.khronos.org/assets/uploads/developers/library//whitepaper_handheld_market_enters_new_phase.pdf", accessed on January 10, 2015.
[2] K.K. Aggarwal & Yogesh Singh, 'Software Engineering", New Age Publication, India, 2008.
[3] Sanjana Tiwari, Shrish Bajpai & M.Arshad, "Modified W Model for Handheld Application Development", International Journal of Engineering and Technical Research (IJETR), ISSN: 2321-0869, Volume-2, Issue-12, December 2014.
[4] Bryan Patten, Inmaculada Arnedillo Sanchez, Brendan Tangney, "Designing collaborative, constructionist and contextual applications for handheld devices", Computers & Education 46 (2006) 294–308.
[5] Richard Fairley, "Software Engineering Concepts", McGraw Hill Education (India) Private Limited, 2001.
[6] Jibitesh Mishra & Ashok Mohanty, "Software Engineering", Pearson Education, India, First edition, 2011.
[7] Xia Cai, Michael R. Lyu, and Kam-Fai Wong. Component-Based Embedded Software Engineering: Development Framework, Quality Assurance and A Generic Assessment Environment, International Journal of Software Engineering and Knowledge Engineering Vol. 12, No. 2 (2002) 107.