

Strategies of Improvement in Requirements Administration

Ing. Beatriz Fernández Tamayo¹, M. en C. Jose Juan Hernández Mora², M. en C. Carolina Roció Sánchez³,
M. en C. María Guadalupe Medina Barrera⁴

¹Master in computational systems, Instituto tecnologico of Apizaco,
Apizaco, Tlaxcala, México
fernandezbeatriz1908[at]gmail.com

²Department of systems and computing, Instituto tecnologico of Apizaco,
Apizaco, Tlaxcala, México
jjhmora[at]itapizaco.edu.mx

³Innovation and talent development center MBN-Miracle Business Network S.A.,
Tlaxcala, México.
sanchez.carolina[at]mbn.com.mx

⁴Department of systems and computing, Instituto tecnologico of Apizaco,
Apizaco, Tlaxcala, México
medinabm0184[at]itapizaco.edu.mx

Abstract: *This article presents a model of developing and defining a series of strategies, for improvement in process of collection, analysis and specification of software project requirements, using agile develop methodologies like SCRUM reforced whit ideology of design UX and mind maps, exploiting artefacts designed standards low of quality CMMI and MoProsoft, this model support work team, for to prevent fails and changes what don't contribute evolution of software product.*

Keywords: Design UX, Improvement, Requirements, Software engineering

1. Introduction

IEEE define software engineering as application of a systematic, disciplined and quantifiable approach to software development, operation and maintenance (IEEE, 2008). The concept of software engineering it arose on 1986, after a conference in Garmischi (Germany) what have like objective solve the problems of software crisis.

The software engineering encompasses:

- ✓ Development software process (specification, implementation and design, etc.).
- ✓ Methodologies for software development (RUP, patterns, framework).
- ✓ Tools of development software.

In the software process we find:

Specifications: users and engineers define the software to produce and restrictions in function.

Development: phase of design and encode of software.

Validation: the software has must be tested for ensure quality.

Evolution: the software must be scalable for to accept changes future (TELLO, 2012).

One big part of the software crisis it's precisely the phase of requirements specification, in what found the problems like lack of know of environment of software work or sometimes exclude details about functions and features specific for user,

these failures result in costly losses of time, money and work, for the company, when we cope with bad practices of collection and validation of requirements.

In the development of software products its fundamental for little and big companies the evolution of requirements specification process of a new software products, using parameters more recent, for example include client in all phases of project, for achieve one system totally usable and intuitive.

This work searches the training of software factory team for good development of software products, focused in ideology UX (user experience), centralize all in the user perception.

2. Collection of requirements, whit focused in User experience

UX (user experience) it's a design philosophy whose objective is creation of products what resolve needs specifics of users, ensure higher satisfaction and best using possible, with minimum effort, it's a process in what using multidisciplinary techniques and all decisions it's based in needs, objectives, expectative and motivations of users.

All process of UX has the next structure:

- Thorough knowledge of users, using qualitative research or quantitative.
- Design a product that meets your needs and fits your capabilities, expectations and motivations.

Volume 7 Issue 5, May 2018

www.ijsr.net

Licensed Under Creative Commons Attribution CC BY

- Try the designed, with a usability test. (Clarenc, 2011)

Several tools of creativity they are useful to team of software factory for to land ideas, and after translate in business opportunities, one of this tools it's SCRUM what in easy terms it's set of practices good, for collaborative job of team with the end of get best results possible of the project, while in this context a prototype, one representation limited of a product, whose using resides in learning and exploration of functionality of initial idea, allowing evolution and correction of mistakes in early stages of project.

With base in analysis this models, tools and activities for process of requirements collection and specification that are established in several company's, we propose a new strategic methodology.

What contemplate reduction of mistakes in requirements knowledge, and that is adjusted considering the existing resources in the company, in order not to generate additional costs, this model can be learned by the current members of the software factory team. (Figure 1).

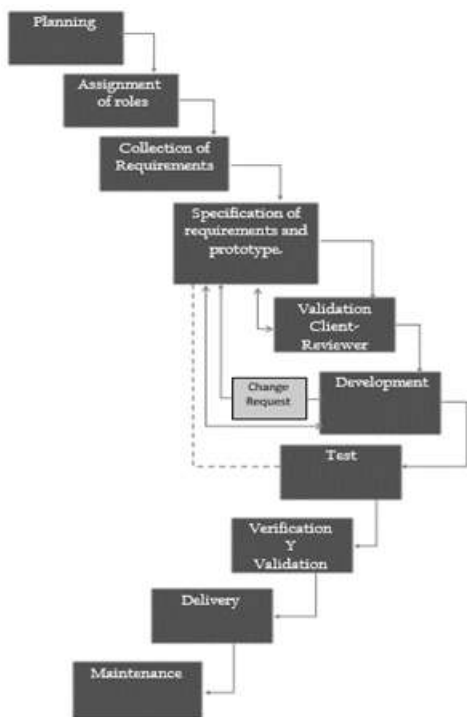


Figure 1 Requirements collection model with UX approach

Were:

Table 1: Symbology

Arrow	Relation
→	Sequence
- - - -	Dependency
↔	Feedback

The first box indicates a review of development plan elaborated of administration of projects department that provides information such as policies, strategies, objectives and goals to the members of the development team, once reviewed and commented on this plan, the next point leads to Software Factory to determine what will be the role or roles

of each member, specify their tasks and deliverable documents.

As soon as this is specified, the requirements analyst, software architect, responsible for testing and the client will be cited for an interview session in which the tests will be applied and the client will be asked to make his own sketch of how he visualizes at that moment product that is required, sound recording of the meeting is suggested (with authorization of those present) for further analysis, the purpose of making backups of all this information is to understand not only the system as such but also the environment in which it will be used.

2.1 Assignment and description of roles

One of first activities of factory software area its the assignment of roles, for stablish responsibilities and chores.

After of revision and according to this chore, is important understand of each role and responsibilities inside team of SF for best organisation of development.

Then a diagram with the roles suggested by Moprosoft for SF is shown (Figure 2).

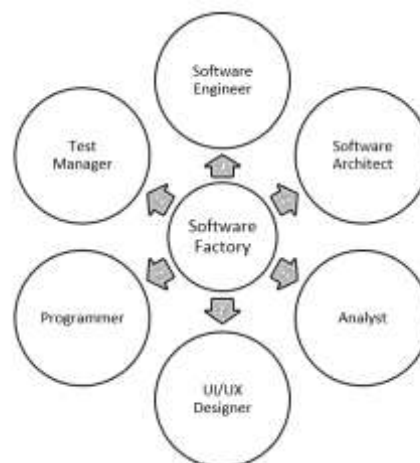


Figure 2 Roles

The adequate organization within the software factory for project development. it requires these five Roles understood and carried out correctly:

Software engineer knowledge's about principles and methodologies of software development, applying the scientific understanding to the logic design, construction and all documentation require for develop, operate and to give maintenance.

Software architect will possess knowledge about the organization and / or structure of the program, that is, diagrams, diagrams, language selection (if not specified by the client), total description to carry out the problem and in what form.

Analyst is one of the links between the user and the computer area, develops an analysis on the functionalities for the project, to detect and understand future problems, requires

skills to detect opportunities for improvement that are associated with the main problem, collaborate effectively with the team through work sessions, verbal and written skills, business knowledge and technology domains, ability to absorb and understand information quickly. Idea and project visual messages, contemplating diverse needs that vary according to the case: stylistic, informative, identifying, persuasive, code, technological, production, innovation, etc.

UX / UI Designer this role could be executed by two different people since the UX designer will possess skills in the design of the total system interaction the proposal of navigation and system responses based on mental maps, perceptions and even emotions that can present the user, while a UI designer has knowledge about design, but referring to the aesthetics of the interfaces.

Programmer develops computer programs analyzing requirements, designing logical solutions, using the appropriate computational tools, in order to satisfy the client's requirements.

Test Manager Identifies tests to be carried out, and the appropriate way to apply them, implements individual tests, prepares and executes tests, records results, performs an analysis and recovery of execution errors.

Figure 3 shows a distribution indicating the participation of the different roles for each phase.



Figure 3 Roles Distribution

It's very important that the analyst learn to understand to each user and help him establish ideas.

2.3 Material

The challenge for analyst it's get useful information, in the shortest possible time, for start the development of the customer-focused system, when start information collection, needs more than knowledge in communication and software developer, will using didactics materials as test, pencil, paper, colors, etc. that help translate ideas from imagination a written narrative on paper, during meeting the requirements team, will inquire various aspects of the work environment and user characteristics, for start design not only the logic, also interface of system, this will give a broader perspective about our problem to be solved or detect opportunities for improvement.

To perform this task, different techniques necessary to obtain a better knowledge of the client's requirements are described. Using the following tools:

- ✓ initial test
- ✓ white screen test
- ✓ Sketch
- ✓ Prototype

What gives us as a result an extensive knowledge base for the analysis and definition of the base requirements of our project.

2.3.1 Initial test

As mentioned previously it's very important have information about the users, its say, question what your job is? or What are the processes you perform in your work? for which we using the next preliminary test:

- ✓ What is the sector of the company to which this project is directed?
- ✓ What is the objective this project?
- ✓ What is the approximate number of users?
- ✓ Are you familiar with any programming language or environment?
- ✓ What is the operating system most used in the company?
- ✓ What is the technical requirement (standards, and procedures to be used and applied) of this project?
- ✓ Requirements special of users for project?
- ✓ Do you have an example that serves as a reference?
- ✓ Who will be our direct contact?
- ✓ What will be its availability?
- ✓ We would have access to company information?
- ✓ Describe in your words the project (Some sheets are provided, maybe pencils or elements to make strokes).

One narrative or sketch made by our interviewee that will be reviewed during a subsequent meeting, the test should previously send for reduce the time invested. this can have complemented whit an ideas storm between team, one of the most common ways to get concepts informally is very useful in many situations where creativity and cognitive thinking are required.

This list of questions is based on the analysis of the problems obtained from past experiences, in internal and external projects of a company.

2.3.2 White screen test

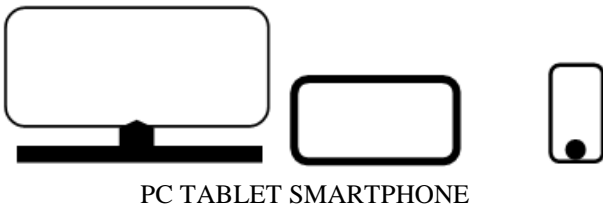
This test will implement together with the initial test, for establishing idea of appropriate mind map for choose the artefacts and position inside the interface.

The objective of this test is show a vision close to the developer of how it could be interaction of user with the interface.

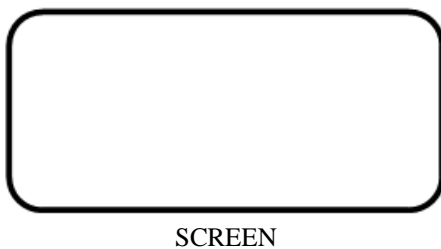
Start

The illustrations that show in the test are examples; the result of prototype could to vary.

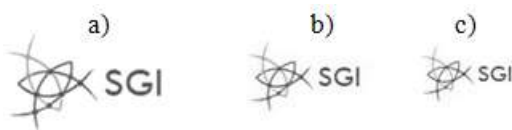
Please choose the dispositive or (s) for to work.



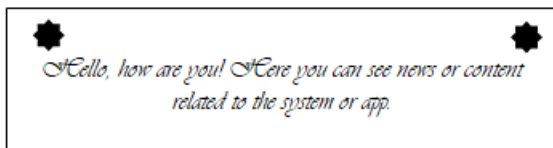
Each element has an index 1,2,3,4 etc. and options of selection as a), b), write the number the element and letter that chose, in a position in the screen.



1. Company's logo, choose size and write position in the screen of up.



2. Write number 2 in the position that choose in the screen.



3. Write number 3 and letter of your election in the position that choose in the screen.

a) Horizontal b) Vertical

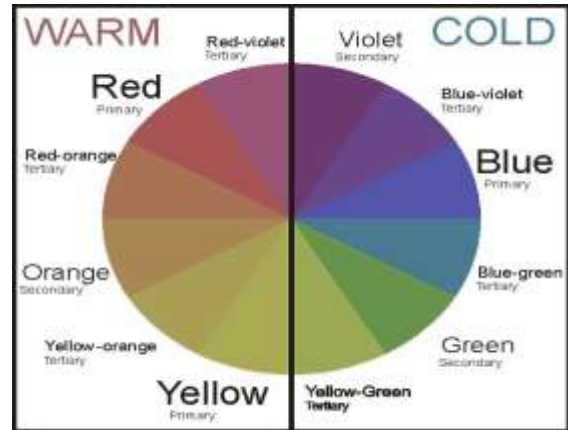


4. Write number 4 and letter of your election in the position that choose in the screen.

a) Icons b) Description



5. What colors do you prefer?



Write here your answer: _____

Thanks for your help!

2.3.3 Develop of deliverables documents and specification of requirements

In factory or development of software, makes several documents for collection information, the one it's "Collection of requirements" where to save a preliminary list of characteristics necessary of system, narrative, sketch, business rules and notes, in this portfolio to enclose the archives of audio and any other element of utility, whit meticulous analysis of all information, make the official requirements list, BPMN diagram, case of use diagrams and prototype that find in the document "Requirement Specification" that will verify and valid, the name of archives will are complemented whit name of project and portfolio will are called "Requirements".

3. Results

In test time of model "requirements based in user experience", we find several strengths and weaknesses they can improve in each project depending on the condition of this.

The model was proved in the project develop "Incidences Administration System".

All software development project has modifications but this not to be a indicative of failure or bad management of requirements, these changes to be considered as evolutions of system and the important is reduce them or reduce of impact this these can have in develop of project.

In this project, specifically the requirements are very specific, in the requirements collection it was easy for team analysis of requirement understands very well the functions you need, whit help of a prototype realised whit Balsamiq Mock-ups software.

The documents necessary for CMMI and MoProsoft verification were realised more speed whit help SpiraTest platform.

The results of tests of Requirements collection model with UX approach, they threw the following data:

In table 1 they have increment of time invested in relationship whit others projects.

Table 2: Estimated time vs. Real time inverted in hours

Estimated Time	Real Time Inverted
1	0.3
48	48
72	72
72	72
96	120
1	2
48	48

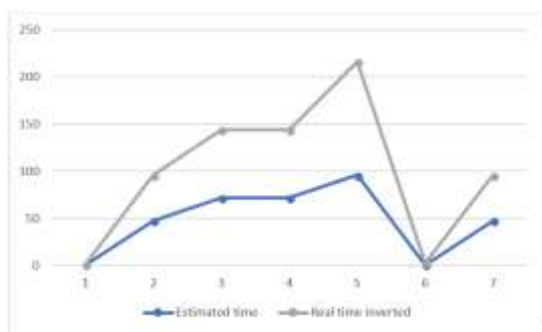


Figure 4 Time inverted in hours

Table 3 show corrections of project "Incidences Administration System". vs past projects, a reduction of corrections was obtained during the development of the project.

Table 3: Number corrections of project Administration requirements system vs past projects.

Activities	Others Projects	Corrections
Requirements collection	3	2
Analysis	4	2
Development prototype	4	3
Specification of requirements	3	1
Validation	3	1
Verification	1	0

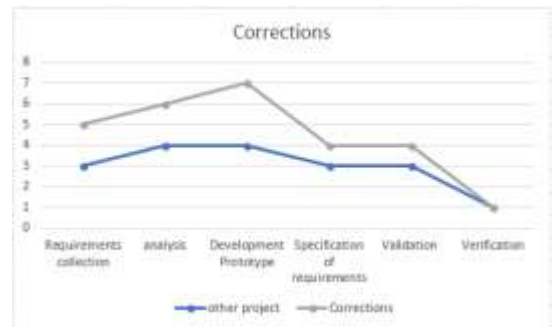


Figure 5 Time invested

4. Conclusion

Recognize interaction requirements as equal important that functional requirements and take in consideration, of minds maps and user experience provide more usability all systems of software and logically more quality.

During the meetings it was possible find the changes necessary and was easier for user and team understand what the desired result was in each module.

A low-level design prototype allowed user navigation, which generated more confidence in the project, it is necessary to specify that the visual design may vary if it is of the user's taste but there will be a time limited for said changes, but the iteration will be as faithful to the prototype as possible.

The tendency of all new software development models is to include the client in each step that the team takes, for this reason an approval is required from the requirements to continue to the development phase, without a written approval there is no progress to another phase.

References

- [1] Clarenc, C. A. (2011). Notions of cyberculture and journalism .
- [2] IEEE. (2008). Standart software engineering. IEEE 830-1993. IEEE.
- [3] TELLO, D. E. (2012). Basic concepts of Software engineering. Tamaulipas: CINVESTAV.

Author Profile



Ing. Beatriz Fernández Tamayo obtained the degree of Engineer in Information Technology and Communications from the Technological Institute of Apizaco. Student of the Masters in Computational Systems of the Technological Institute of Apizaco. Areas of interest in research: Software Engineering, Application Software Development Methodologies.



M. in C. Jose Juan Hernández Mora is a professor of the Master's Degree in Computational Systems of the Technological Institute of Apizaco. Area of Affiliation: Systems and Computing, is a Computer Engineer from the Autonomous University of Tlaxcala and Master in Computer Science from the National Center for Technological Research

and Development. Areas of interest in research: Software Engineering, Application Software Development Methodologies, Digital Image Processing, Neural Networks.



M. in C. Carolina Rocío Sánchez Pérez is a full-time associate professor of Computer Engineering at the Universidad Autonomy de Tlaxcala and academic coordinator of the Computer Engineering Education Program. Master in Computer Science from the National Institute of Astrophysics, Optics and Electronics (INAOE). He has the Recognition of Desirable Profile PRODEP and is a member of the Academic Body "Distributed Systems" of the UATx. He has participated in bonding projects funded by CONACYT.



M. in C. María Guadalupe Medina Barrera is a full-time professor of the Master's Degree in Computational Systems of the Technological Institute of Apizaco. Ascription area: Division of Postgraduate Studies and Research. Master in Computational Systems by the National Center for Technological Research and Development (cenidet). Professor with Recognition of the Desirable Profile PRODEP and member of the academic body PRODEP "Information Systems". Evaluator of the National Council of Accreditation in Computing and Computing A.C. (conaic)