E-Health System

Team 2 K16T1

Project plan

For E- Health System

Version 1.1

Written by Le Ngoc Chau

Contents

Revision history

Version	Updated Date	Author	Description
1.0	26/05/2013	Le Ngoc Chau	Define project plan
1.1	09/06/2013	Le Ngoc Chau	Update project schedule

List of table

1. Introduction

1.1 Project overview

1.1.1 Purpose

Allows ABC hospitals to manage outpatient's information through the category management and cashier management to handle business processes of the cashier easily and conveniently, to avoid mistakes, losses information.

1.1.2 Project scope

- System used for the cashier department of ABC hospital.
- Staff of each department only have right to action on system with business processes of their department, they aren't known and influenced other departments.

1.2 Project deliverables

- Products will be completed within 10 weeks from the start of the project.
- Customers will be responsible for the hardware to be able to run the software and take responsibility for the software after it has been deployed and in use.

1.3 The implementation of project management

- All changes must be approved from the Change control board before going into practice.
- All changes must be documented and updated continuously.

1.4 References

Refer to from K14T graduation projects, technical reading software project management, final project of SPM course - FGS Logistic, records abcHospital of ABC hospital.

1.5 Definitions and Acronyms

No	Acronyms	Description
1	UC	Use case
2	ConOps	Concepts of Operation
3	URD	User requirement document
	SRS	System requirement specification
5	SDS	Software detail design specification
6	WBS	Work breakdown structure
7	PM	Project manager.

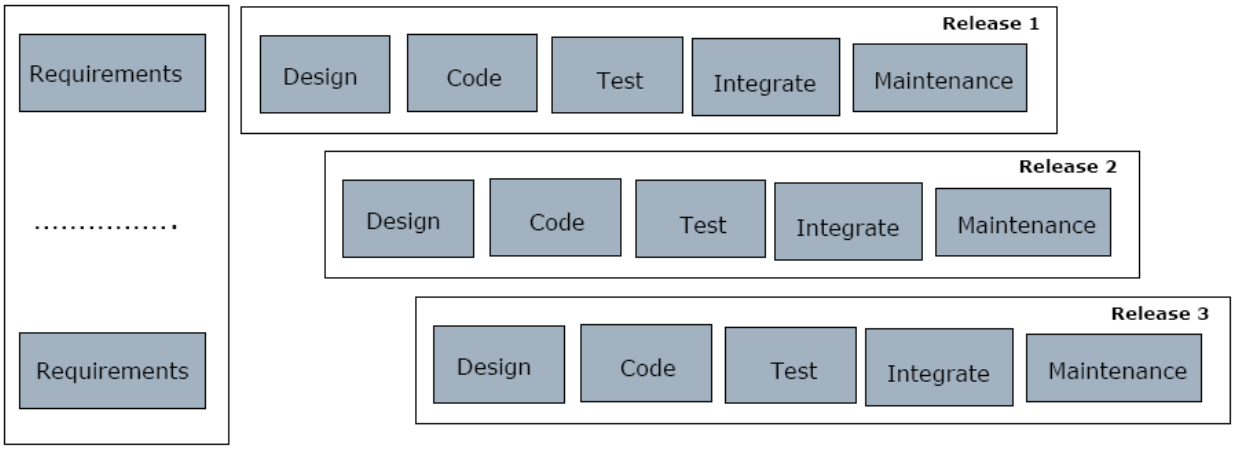
Table 1: Definitions and acronyms

2. Project Organization

2.1 Process model

2.1.1 Introduction

- Ascending Model is a model combining waterfall model with the philosophy of iterative prototyping model.
- It is a repeating pattern, which allows software engineers to develop the complete software version increases.



- Growth model combines the advantages of the waterfall model and prototyping model. The idea of this model is divided into software growth and development, turn handed them to customers by level of importance. The growth which in turn was developed corresponding requirements will be analyzed.changes from customers for new development has not been accepted.
- According to Lan Sommerville, each of the growth should not exceed 20 000 lines of code and must provide certain benefits to customers. According to Mike Cotterell and Bob Hughes in "Software Project Management", each of which should grow from 1% to 5% of the entire project and should not last more than a month.

2.1.2 Process phase

2.1.2.1 Description

No	Role	Description
1	*Requirement*	- This is the first stage of the process to determine customer requirements and define the scope of the project.After requirements analysis can identify the main functions, the high-risk function and the potential function in the future.
		- Through Requirement phase, the PM can make the business case include: evaluating the success of the project, risk assessment, identify resources needed, given the summary calendar process of the major milestones of the project, the risks of the request, the function should also be pointed out.
		- During this period also refers to the modification of the existing system, but still only summarize this stage focuses on two important of projects: worth doing or not, and the ability to perform
		- At the end of this process is to examine the objectives of the development process of the project and decide whether to continue development or not. The result of this phase is set to be the consensus among those who played a key role on the objectives of the project.
2	*Architecture*	- This stage creates preliminary sketches for the system as a basis for the detailed design process. This architecture is extended to the analysis of requirements (requirements have a major impact on the system and the main function) and risk assessment.
		- The objective of this phase is to analyze the business problem, define the logical architecture, provides a detailed plan for each increment.
		- End stage to check the detailed objectives and scope of the system, choice of architecture and how to deal with risks.
3	*Design*	- Based on sketches from System Architect phase will provide detail design for each specific function of each incremental phase.
		- The objective of this phase further analyzed each function in each increment and determine the steps of each function in a concrete
		- At the end of this stage will provide the detail design to switch to stage implements.

4	*Code*	- This implementation phase coding through the detail design and make unit tests and integration test.
5	*Test*	- This stage given the test plan and execute system test, integration test and writing test scripts
6	*Integrated*	- Integrate and handle the source code to build a complete product for each increment
		- When have the next increment, must integrate and monitoring and evaluate these errors occur in the process of integrating the increment
		- Make sure to follow the configuration management system which was launched
7	*Maintain*	- Make sure the system is running properly on request and can overcome the problems in terms of hardware for products.
8	*Release management*	- This is the main manager of the product at each increment, working with customers and users of the software.
		- Get feedback from customers to improve the increment after
		- Learning for the project and ensure the project continues to grow or not.

Table 2: Process description

2.1.2.2 Characteristics

- When Incremental models are used, the first version is the core product. It meets the basic needs, but also many other features, functions unresolved. Core products used by customers / users, reviews, and a new plan was developed for subsequent improvements. The plan to change the core product to better meet customer needs, adding new features and functions. This process is repeated with a product delivery improved each time, until the finished product is created.
- The requirements are identified and classified according to priority, a high priority for the key functions and the high-risk function.
- Only changes from the client side has not been done will be accepted.

2.1.3 Roles

No	Role	Responsibility
1	*Project manager*	- Creates and updates the plan - Recognize team's issues (personnel, project progress...) early - Track and control progress - Resolve team's issues - Help team allocate tasks and must ensure tasks are scheduled - Communicate with customer - Communicate with team - Track personal work of each of team member

2	*Requirement engineer*	- Communicate with customer to get requirement - Analysis and presentation about specific requirement for stakeholder who can understand. - Maintenance of the requirement specification documentation. - Manage change and evolution of requirement. - Accurately records task tracking data.
3	*Architecture*	- Design system and monitoring the development of architecture phase - Provide an overview of the entire project - Supervise the difference of increments.
4	*Designer*	- Detailed design, implementation of the architectural elements, and integration of the element into the system.
5	*Coder*	- Writes code - Write unit test script - Ensure function passed all units test script
6	*Tester*	- Generating test plans - Execute system test, integration test and record - Write Result into report - Defect analysis
7	*Document person*	- Document all minute of meeting team - Document the operational system and installation instructions

Table 3: Role

2.1.4 Pros and cons

a. Pros

- Shorten the waiting time of customers.Customers do not have to wait until the entire system is completed to be able to go into use. The most important component is delivery earlier and carries earlier benefits to customers.
- Increase the quality of software. The most important component of the system was developed and put into operation soon, so it is well tested.In addition, the opinions of our customers and previous experience developing components will be applied immediately for the following components.
- Reduce unnecessary requirements from customers. When a feature is not present in the system, they will think that it will be integrated into the next delivery.
- Increase in labor productivity. Many programmers do a better job in small projects and they can see the fruits of their labor sooner
- Reduce the risk of the entire project, the adverse risks are broken down in the entire project.

b. Cons

- Not all projects can be divided into the growth of small to sequential development and delivery.If the period of planning and analysis system is not good, the conflict between the components may arise.

- Must determine fully functional and complete before the next development functions.
- See core products, customers may think the work is simple, inexpensive, easy to done.
- Requires good planning and design, reasonable division of work, the employee must work well.

2.2 Team structure

No	Member	Roles
1	Trinh Thai Anh	Software engineer
2	Le Ngoc Chau	Project manager
3	Khau Thanh Dao	Software engineer
4	Pham Ngoc Hung	Software engineer
5	Ngo Quang Huy	Software engineer
6	Ta Ngoc Then Phu	Software engineer

Table 4: Team structure

2.3 Role and responsibilities

No	Roles	Responsibility
1	ABC Hospital project manager	Le Ngoc Chau
2	Requirement leader	Pham Ngoc Hung
3	Design leader	Ta Ngoc Thien Phu
4	Code leader	Khau Thanh Dao
5	Test leader	Ngo Quang Huy
6	Document person	Trinh Thai Anh

Table 5: Role and responsibilities

3. Managerial Process

3.1 Management Objectives & Priorities

- Provide product in the 10-week period and within the proposed budget.
- Identify priority to complete the module to meet the requirements of customers in each period.
- Members of the team must meet and report progress of work assigned with person who has responsibility and PM weekly
- Meet the customers to report the progress of work with customers at the same time control and determine if any changes from the customers side weekly.

3.2 Assumptions, Dependencies & Constraints

- Time and cost must be satisfied.
- Products must be reliable, satisfactory.
- Products must be consistent with functional and non-functional requirements which were identified
- Products must be friendly with user and easy to use.

3.3 Risk management plan

- Refer to the documentation *SEP_PM_RiskManagementPlan.pdf*

3.4 Change management plan

- Refer to the documentation *SEP_PM_ChangeManagementPlan.pdf*

3.5 Configuration management plan

- Refer to the documentation *SEP_PM_ConfigurationManagementPlan.pdf*

3.6 Measurement plan

- Refer to the documentation *SEP_PM_MeasurementPlan.pdf*

3.7 Communication plan

- Refer to the documentation *SEP_PM_CommunicationPlan.pdf*

3.8 Architecture and design plan

- Refer to the documentation *SEP_PM_ArchitecturePlan.pdf*

3.9 Implement plan

- Refer to the documentation *SEP_PM_ImplementPlan.pdf*

3.10 Test plan

- Refer to the documentation *SEP_PM_TestPlan.pdf*

3.11 Project charter

- Refer to the documentation *SEP_PM_ProjectCharter.pdf*

3.12 Team charter

- Refer to the documentation *SEP_PM_TeamCharter.pdf*

3.13 Monitoring and Controlling Mechanisms

No	Members	Work
1	Le Ngoc Chau	Responsible for all the evaluation and supervision of the work of the development team
2	Khau Thanh Dao Pham Ngoc Hung Ngo Quang Huy Ta Ngoc Thien Phu	• At each meeting progress reports, each member will have to present the progress of the work and problems encountered during the implementation of the work that his main responsibility. • Provide specific list of work and plan for their work.
3	All team	Any appearance issues that affect or impact the project must immediately report to the PM.
4	Le Ngoc Chau Trinh Thai Anh	Responsible record the relevant documents in the work of the group.Including meeting minute, customers meetings, presentations, progress for our customers.

Table 6: Monitoring and controlling mechanisms

3.14 Staffing Plans

No	Phases	Implement time	Responsibility
1	Management, to identify and update the process for management	10 weeks (27/5 - /2013)	Le Ngoc Chau
2	Get requirement and document the customers's requirements	2 weeks (10/6 - 23/6/2013)	Pham Ngoc Hung
3	Detail design for system	1 weeks (24/6 – 30/6/2013)	Ta Ngoc Thien Phu
4	Implementation of modules and integrated	2 weeks (1/7 – 14/7/2013)	Khau Thanh Dao
5	Test product	2 weeks (8/7 - 21/7/2013)	Ngo Quang Huy

Table 7: Staff plans

4. Technical Process

4.1 Methods, Tools and Techniques

- Incremental model of Agile and Plan driven approach.
- The development tools: Visual Studio Ultimate 2010 and SQL Server 2005.
- Code is written in C #, using PC Suite API and PC connectivity SDK 3.0.
- This application run under windows 7/Vista/XP personal computer and requires Microsoft NET Framework 04.
- Documentation and code will be made based on the desired customers expectations in combination with standard of the development team.

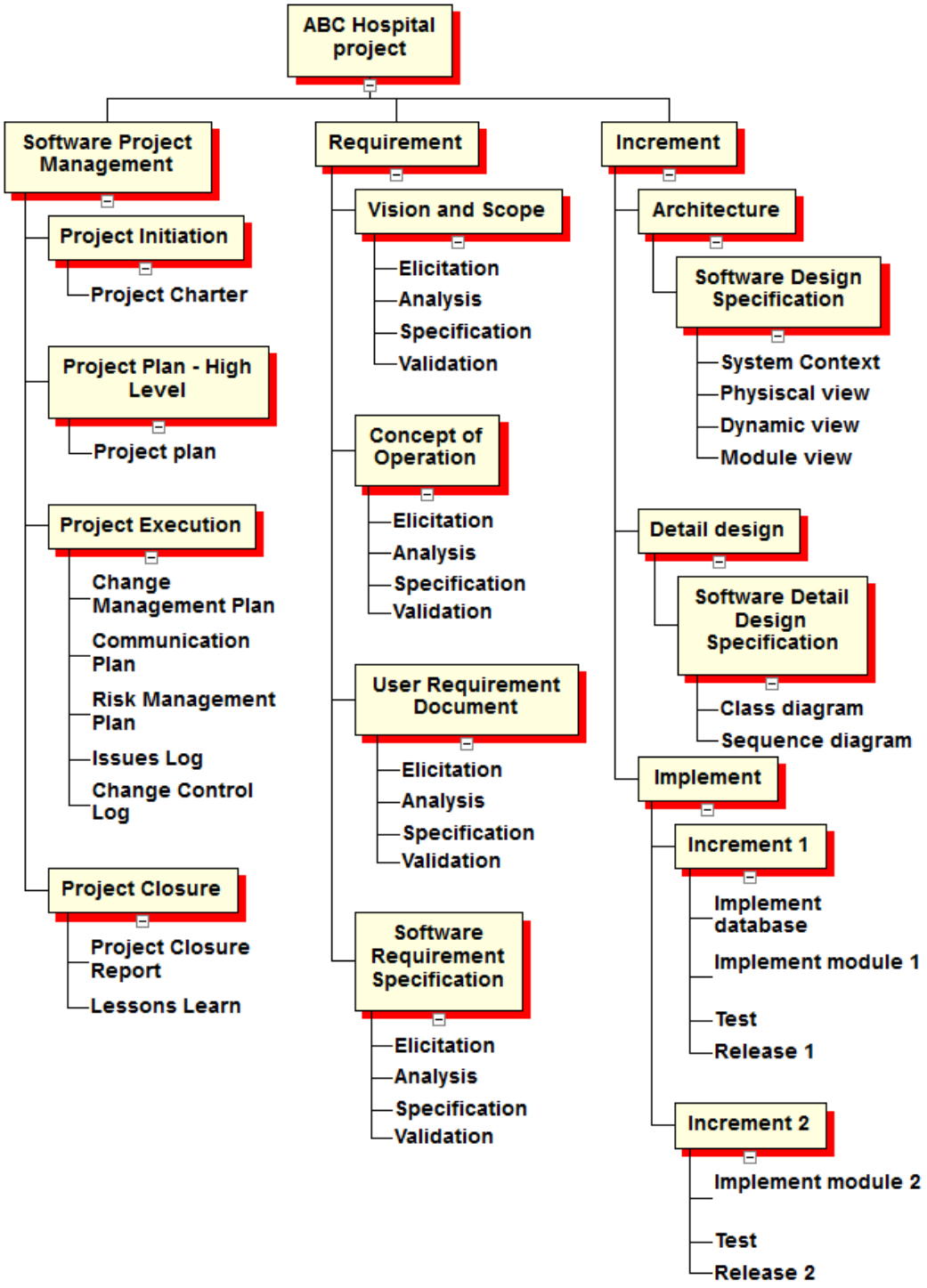
4.2 Software documentation

- Document software must comply with the standards of the development team.
- Review of documents will be carried out by the PM at each stage of the job done.

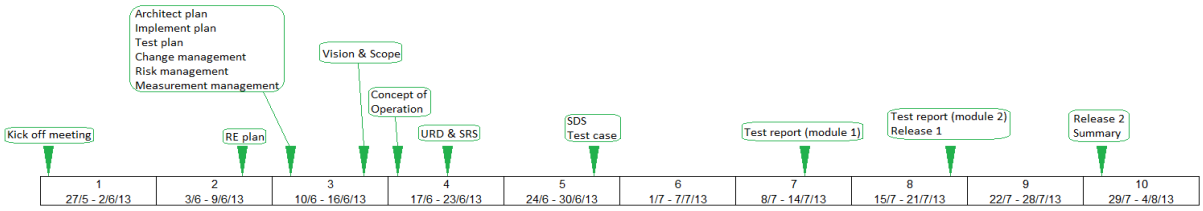
5. Work Packages, Schedule

5.1 Work Packages

WBS of E - Health System.



5.2 Milestone



5.3 Schedule

Week	Task Description
1	- Identify members - Define role and responsibility - Identify principles work for the group. - Define management processes - Training code for team members
2	- Define project plan - Define requirement plan, architecture plan, implement plan, test plan. - Identify risk mitigate managment plan, change management plan, measurement plan, configration management plan - Training code for team members
3	- Get requirement - Evaluation processes - Identify risk mitigate managment plan, change management plan, measurement plan, configration management plan - Training code for team members - Write vision and scope document
4	- Get requirement - Write Concept of operation - Write URD - Write SRS
5	- Identify architect drivers, perspective - Write Architectural specification - Write Detail design specification - Write ATAM report
6	- Design interface - Review document - Implement module 1
7	- Test module 1 by integration test and write test report - Implement module 2 - Fix defect in module 1
8	- Test module 2 by integration test and write test report - Fix defect in module 2
9	- Review document - Reivew code
10	- Present

Table 8: Schedule of project