eSpace Development Process

Document Version: 2.0

Date: 19/07/09

Document Versions

Version	Date	Modifier	Modifications
<1.0>	09/16/08	Heba Hosny	
<2.0>	07/19/09	Ali Maher	

Highlighted Colors Key

Color	Description
Yellow	The highlighted words represent the measures that are covered in the measures report. Each of these measures is evaluated in the project; according to the evaluation an idea can be given concerning the progress of each element in the project and, thus, the progress of the project overall.

I-Introduction

This document describes eSpace Development Process. This description includes:
a- Development Team.
b- Development practices.
c- Development work flow.
d- Communication with the other teams.
c- Process management tools.

II-Objectives

While designing this process our main goal was to define a process that achieve:

1. Customer's satisfaction: deliver the right product on time and on budget with high quality.
2. Keeping eSpace spirit and environment.
3. Minimizing process overhead.
4. Collecting measurements.
5. Smoothing the work flow and reducing the chances of problems.

1-Team and Roles

Team <u>must not</u> be less than **4** members; a **UI designer**, a **tester** and 2 **developers** one of them is a **team leader**. Note that there are some other roles that are important for the success of the project but they are not fully dedicated to the project and not under the control of the team leader. These roles are: **Business expert**, **Solutions admin** and **Graphic designer**.

2-Contract writing

1- Customer send his description of the required system in a form of requirement document.

2- Operations (or services) manager decide the best team to be assigned to the new project.

3- Operations (or services) manager sends the requirement document to the team leader.

4- Team leader converts this document into a set of undetailed user stories.

5- Team leader gives a “size” estimate for each story in story points. This size estimate <u>must</u> take into consideration the estimate of design, testing, risks and learning <u>without</u> passing by every team member. Only one size estimate is generated per story.

6- He passes the same stories to one of his team members or to any other developer to give his estimates.

7- The combined estimates after discussing the large differences and converting the size into time (man-days) are passed back to the operations manager.

8- This estimation must be reflected in contract in the following articles:

a) Deliverables: stated in the contract in the form of themes. **Theme** is the deliverable of a set of consecutive iterations that add a new value to customer's business.

b) Schedule (time line): stated in the contract as Release plan. Expected calender date for each theme.

c) Changes: Contract must clearly state that we only accept changes within the Deliverables scope.

d) Communication: Contract must clearly state a limited review and feedback period for the customer after each delivery.

e) Payments: every theme should have its corresponding payment. Theme may be equivalent to one iteration only based on the nature of the project.

3-Project kickoff meeting

Kickoff meeting is the first gathering for all team members and the operations manager. The output of this meeting is:

1. **Team** is assigned to the project.
2. **Scope** (list of project features) is determined.
3. **Goals** (examples: customer trust, building experience in new tech, ...) are stated clearly.
4. **Final product** is described.
5. **Investigations , learnings and risks** are determined.
6. **Release plan** is done. (Iteration's calender deadlines (start, end, estimated story point, derived estimated man-days. Note that man-day is a normal business day for single developer. He is expected to pay 8 hours per day which will result in 6.8 hours net), cost and resources are defined.
7. **Themes** are stated (as written in the contract).
8. The start and end day (of the week) for the project iterations.

Important meeting points must be documented in the *Wiki* of the project

4-Project Setup

All tools are setup and working on servers and developers' machines.
Tools are for:

1. A revision control system tool, *i.e. SVN* .
2. A project management tool, *i.e. Redmine.*
3. A UI automatic testing tool, *i.e. Selenium.*
4. An IDE with all Plugins, *i.e. eclipse,.*
5. A web UI for managing deployments, *i.e. Webistrano.*
6. A testing environment server; *staging server.*
7. An integration testing tool, i.e. a recipe for testing integration in *Webistrano.*
8. A code coverage test tool, *i.e. rcov.*
9. Defects tracking tool, if it's not embedded in the project management tool.

5-SVN Structure

Any project at eSpace (service, product, library, component,...) must have the following SVN structure: <Project name>/trunk

/branches

/tags

-trunk contains directly the project version that is currently under development. It may be unstable (but must be running and passing the unit tests) or finished.

-Branches are used for project versions that may need major, risky, complex,.. changes.

-Each iteration (in release 1,2,.. or in support) has a deliverable at its end. A "tag" must be created with the following naming convention: <release>.<iteration>.<build>

If the deliverable is ready to deploy on the staging server, another tag called "testing" should be created. If it is ready for production this tag should be "stable".

6-Learning and Investigations

If the project requires investigating new tools or technologies, or requires acquiring new knowledge, then this must be done as follows:

a) Estimate the time they will take in the investigation or learning.
b) State the output of the investigation or learning in terms of working software.
c) Add this to the project tasks.

7-Architecture Design

1) Designing the system **architecture** and **main modules**.
2) Defining the **reusable components** that can be reused (from open source projects or eSpace's previous projects) in this project.
3) Documenting the architecture on the Wiki.

8-Iteration 0

Iteration 0 is the first iteration in the project. It is predefined to be only one business week length. It should be taken into consideration when estimating and planning. It is dedicated for the following tasks:

1- Project setup (see point 4 above)

2- Learning and investigations (see point 5 above)

3- Architecture design (see point 6 above)

4- Technical pre-analysis

5- Domain pre-analysis and design documentation.

6- Story prioritizing and planning for Iteration 1.

9-Iteration Work flow

Iteration must start and finish on one of Mon, Tue, Wed. (any extra days should be in the first iteration)

1. Start a new **iteration** in the project management tool, and fill it with the **stories** that have the highest instant priority. Each story has to have (beside its previously estimated points):

   a) Title.

   b) Description.

   c) Time estimate.

   d) Due time.

This is done by a planning meeting with the customer or by sending him the final iteration plan to review.

2. Assign each story to a member. The assigned member must divide the story into **tasks** and give each task an estimate.
3. Hold a wire frames meeting between the developers and the designer. Wire frames have to contain the main work flow design and privileges for each of the system users. The output must be accepted by the customer.
4. Beside the stories, iteration plan must include **defects fixing** for pending defects based on their **severity.** Iteration plan also may include any remaining tasks from a previous iteration.

For every story, tasks has to be done as follows:

1. Designers should deliver the interface design (HTML, CSS, Flash and effects JS in the case of building web applications). Developers then fill those interfaces with dynamic code. It's not allowed for developers to change those GUI files.
2. Developers then implement both main code and its required unit/functional tests individually or on pairs. Code must at least be covered by 80% by unit tests.
3. Do code Refactoring whenever you need. This will enhance the design by time.
4. Ensure that the code passes the unit tests.
5. A code review has to be done by any one in the team. It is not allowed to commit code without review.
6. **Commit** the code in the version control system, *SVN,* (based on the previous two steps this is an approved commit).
7. Team leader has to run both integration tests and coverage tests to be sure of their pass, QA should run them as well.
8. **Build, test and deploy** with project deployment tool. Never to leave code with failure in building, testing or deployment. Note: Deployment will be on the *staging* server which may be accessed by the customer.
9. Tester has to write the GUI **functional** tests using UI automatic testing tool (*Selenium*) and add them to the version control tool (*SVN*).
10. Task is finished if it is **accepted** by the team leader and the tester who's responsible of

closing it before reviewing it by the customer. Note that the task life cycle is:

new (created by developer) --> resolved (finished by developer) -->closed (accepted by tester) / reopened (rejected by tester or customer)

11. The task may not accepted. This may lead to one of the following states:
    a) *Sever*: task is not completed and developer must fix it immediately (before moving to the next task) because his task is not finished yet.
    b) *defects*: task is completed and defects will be opened in *defects tracking tool* (*Bugzilla*).
    c) *New feature*: task is completed and new Feature(s) is added as a task in *project management tool* (*Redmine*).
12. Introducing new tasks may lead to one of the following **decisions:** Refactoring, logic changes, integration issues, change layout design or workflow design, ... Any decision must be a team decision. This must be reflected in **Documentation** and **Tests** (if needed).
13. Iteration closing
    a) At the end of the iteration the planned deliverable should be tested, accepted and deployed on the staging server. A tag must be created at SVN as described in section 5.
    b) If the iteration's due date is detained, new due date has to be specified in project management tool, and that must be reflected in the project plan. This update has to be announced early and clearly to all parties.
    c) Iterations should be closed in the middle of the working week; it's not recommended that iterations be due for delivery on days that immediately precede official vacations.
    d) Iteration must be closed. A decision may be taken to extend the due date of the iteration. This extension must not to exceed double the time estimated for it.

10-Customer handling

Customer is free to change his requirements. We have to respond to his changes based on the nature of each change (Ordered by their difficulty):

1-Small changes: small change is the change that requires around one or two hours and will not affect our architecture or add new stories to the system. We should quickly accept these changes. This helps in building the trust.

2-Non-implemented details changes: Changes in the details of a non-implemented story. We should accept these changes as long as it will not change the old story into a totally new (more difficult) one.

3-Implemented details changes: After implementing a story, the customer is asked to review it and give us his feedback. If he send his feedback within the period stated in the contract then it may be duo to one of the following reasons:

a) Our nonstandard implementation: Then it is his right to have the changes he needs.

b) Our misunderstanding: Then it is his right to have the changes he needs.

c) He found that it is better: We should evaluate the request and accept or reject.

4-Scheduling of future story in the current iteration: Sometimes customer shows his urgent need for a future (scheduled in one of the coming iterations) story to be done in the current iteration. Customer has to choose between one of the following:

a) Wait for the current iteration to finish and schedule the story in the next iteration plan

b) Replace the story with one or more equivalent (in size) stories.

5-Introducing new story or scenario: Customer may try to add new stories and business scenarios to the previously estimated and scheduled stories. Customer has to choose between one of the following:

a) Replace the story with one or more equivalent (in size) stories and drop the replaced stories from the release plan.

b) Open a story list for the next release plan (if not already opened) and add the new story to it.

6-Customer communication is too rapid and causes too much overhead: This needs careful handling by the team leader. Our suggestion is to fix two times per week for customer meeting. This will give him time for preparing his requirements and limits his interruptions.

11-Team work

1. Team must have at least **5** hours of intersection of their time at eSpace and leader has to dedicate not less than quarter of his day time and not more than half of his day time for process following up activities.
2. Team members should increase the communication to achieve the following:
   a) **Knowledge** and news sharing and updating.
   b) **Keep** their eyes on the **delivery time**. If there is a possibility for missing any deadline, the new schedule must be **communicated** early and clearly to all parties.
   c) **Solve** problems and take **decisions**.
   d) **Respond** to customer changes.

3. Tester checks the code unit tests coverage using the code coverage tool (*RCov*) and check the integrity of the system using integration tests at the end of each iteration.
4. Team is highly encouraged to focus on producing reusable components while working in their project. If They found some code that may be re-factored to be reusable they are encouraged to do this.
5. CTO has to assign a fixed time for each project to:
   a) Monitor and enhance the technical level of team members.
   b) Help in hard problems.
   c) Make sure that company goals are achieved.
   d) Encourage reusable components extraction and enriching our components library, code base, knowledge base and open source contributions.
   e) Recommend blog posts and sessions to write.
6. Project Measurement report has to be sent weekly, on Wednesdays, to services manager to follow up projects progresses.

12-Education

If the team has some valuable achievement they can publish this achievement via:

1.Blog posts.

2.Wiki entries.
3.Sessions.

13-Project Documentation

Documentation is part of our quality. Developers have to deliver 4 documents for every project:

1- Basic scenarios and business logic.

2- Environment and setup: Describes server configurations, database, versions, system deployments, 3rd party packages and plugins.

3- Architecture design

1. Patterns, frameworks, models, important data-structures, ...
2. Main modules: used components, their interaction, main algorithms, assumption,...

Note: Skip architecture design If it's a pure normal MVC.

4- An auto-generated code documentation: by *rdoc.* Developers have to describe the code by comments.

14-Handover to support

Handover process is critical for smooth and successful support. The following are required from the development team that has just finished a project and it will be handovered to the support team:

a) Code coverage: not less than 80%

b) Documentation: Complete documentation for business logic, architecture, environment setup and rdocs.

c) Handover walk through.

d) 4 hours/week for 1 month after handover.

e) SVN contains the latest code.

f) SVN follows eSpace standard structure.

15-Project States

Any project may be in one of the following states:

1-**development** (using *staging* server)

2-**production only** (using *production* server)

3-**production/support** (using *production* server and *amazon* server for testing)

4-**production/development** (using *production* server and *staging* server for new features)

5-**production/support/development** (using *production* server and *staging* server for new features and testing).

The only exception is the pure CMS projects. For these projects Production = Staging. Testing take place on developer's machines.

16- Project Sign Off

Signing off is an important step that helps in:

1.Clear Announcement of the closure of the current phase
2.Putting the project in production state
3.Starting support (if needed)
4.Collecting customer feedback
5.Writing down the Lessons learned

This may be done by sign off meeting with the customer or at least by writing the sign off report.