Management of Software Development Projects

Felician ALECU, PhD, University Lecturer
Department of Economic Informatics
Academy of Economic Studies, Bucharest, Romania
E-mail: alecu[at]ase[dot]ro; Web Page: http://alecu.ase.ro

Abstract: Any major software development starts with the Initiating process group. Once the charter document is approved, the Planning and then to the Executing stages will follow. Monitoring and Controlling is measuring the potential performance deviation of the project in terms of schedule and costs and performs the related Integrated Change Control activities. At the end, during the Closing, the program/project manager will check the entire work is completed and the objectives are met.


Project Management Process Groups

According to PMBOK, there are five Project Management Process Groups (also known as Project Management Processes) required for any type of project, including the software development ones (Figure 1). These five Project Management Process Groups are totally different than the Project Phases (or Project Life Cycles) that are industry dependent, so projects from various fields will have completely different
phases.

Figure 1 – Project Management Process Groups

Any project starts with the **Initiating Process Group** where it is decided if the project will be selected and accepted based on high-level planning efforts performed at this stage. If the project is approved, it will move further to the **Planning Process Group**, so detailed **Project Management Plans** are prepared. The **Executing Process Group** will follow, so the work will be completed according to the plans and the results will be supplied to the **Monitoring and Controlling Process Group** that will make sure the project is on the track in terms of scope, time, cost, risk and quality. In the case when variations to the plan are encountered, depending on the severity of the identified issues the project will return back to **Initiating** (the issues are critical so a decision point is needed in order to find out if the project would continue or not), **Planning** (major issues are discovered or significant changes are approved) or **Executing** (when there are no baselines affected by the changes).
**Initiating**

Any major software development starts with the **Initiating** process group that usually means an iterative clarification of the high level customer needs, including but not limited to the project justification, objectives, high level requirements, summary budget, milestones, risks and so on. All these items become part of the **project charter**.

Typically such changes are actually affecting several components that could be managed as distinct projects since they have separate requirements, budgets and cost reporting schemes. According to this viewpoint, the software development initiative becomes a **program** made by several interdependent projects.

Even if the **collect requirements** usually belongs to the **planning** phase, the requirements clarification stage usually is started here in order to make available for the charter the needed functional requirements, packages, delivery dates and so on.

**Planning**

Once the charter document is approved, the program will progress from **Initiating** to the **Planning** stage. At this point the analysis part starts by identifying the scope of the individual projects and the actions to be performed in order to meet the requirements, so the approved **project management plans** will be available including requirements refinement, scope definition, WBS creation, schedule, cost and budget estimations, quality requirements, risk identifications, response planning and so on. The WBS will be detailed at the component level, so a typical WBS would look like the one presented below (Figure 2).
Based on the project charter, the analysis part should first generate a unique Integrated SRS document (at the program level) that will describe the use cases. This document will be used as a basis for the test cases generation.

The WBS creation will start right after and it will be first detailed at feature level. The analysis part will be continued with a complete clarification of requirements. For each delivery, a dedicated SRS
document will be created and it will be used to design the test cases and to split the features to be delivered into more manageable components by generating HLD documents so, finally, the latest WBS levels will be filled in.

**Executing**

The executing activities will be performed at the Project & Delivery level, so, for example, the Project #1 work will begin with the Delivery #1, followed by Delivery #2 and so on. Once a component is completely implemented (all the corresponding features are developed and individually tested), it will be moved further to the inspection and validation activities. The Quality Control can perform intermediate tests in order to validate some features, but these tests cannot replace the full delivery package inspection, as described in the next section.

**Controlling**

Quality Control will be FIRST performed at Project & Delivery level, so a dedicated quality check will take place for each delivery of a project, since these deliveries are managed at project level. There will be a separate role at the Quality Control Department level having the responsibility of assembling the individual components into a single package (installation kit and/or fix/service pack and/or ...) corresponding to the entire delivery for a project.

The testing activities will generate Test Results, Test Evaluation Reports and other documents, too, if needed.

A second Quality Control step will be performed at the PROGRAM level, in order to confirm the validated packages of each project are actually working together as a whole. Some packages may be rejected, so they will return to Executing or Planning stages.
Finally, a dedicated deliverer role (at the project level) will perform the delivery of the packages that are considered as passing BOTH STEPS of validations.

All the related Integrated Change Control activities are performed here in order to make sure that only the approved changes will become part of the program and its corresponding projects.

**Monitoring**

In order to measure the potential performance deviation of the program (and its corresponding projects) in terms of schedule and costs, the EVM (*Earned Value Management*) method is applied by starting from the following values:

- **EV** (*Earned Value* – estimated value of the work already accomplished) – measured by collecting the progress percents from the project team and applying them to the total estimated efforts

- **PV** (*Planned Value* – estimated value of the work planned to be done) – generated directly from the project plan by using a dedicated tool

- **AC** (*Actual Cost* – the cost of the work accomplished) – taken from the WIP (*Work In Progress*) system, the value is based on the daily reporting of the project team

A sample EVM chart is presented below (Figure 3).
The Schedule Variance (SV), Schedule Performance Index (SPI), Cost Variance (CV) and Cost Performance Index (CPI) are measuring the project deviations in terms of schedule and costs, so these indicators can be used to point out how the project is going.

EAC (Estimate At Completion), ETC (Estimate To Complete) and VAC (Variance At Completion) are additional useful EVM indicators.

The Monitoring process group could indicate some plans should be updated.
Closing

Project closing is assisting the program and project managers to make sure all the program work is completed and the projects have met their objectives stated into the charter and project management plans.

This means the entire program documentation, including the plans, should be updated in order to reflect the currently performed changes. Also, the lessons learned are supposed be documented, historical information updated (project records, statistics ...) and the files for program and projects archived for future use.

The customer acceptance should be included here, too, as well as the formal documentation indicating the completion of the project.

Finally, the project/process improvement ideas should be collected from the stakeholders, the program teams have to be informed about the end of the project and released for future new assignments.

Conclusions

This article tries to illustrate a methodology to be used to complement the PMI’s standard of management for software development projects and programs. The Project Management Institute standard is actually considered as being a project management reference, a framework that can be applied most of the time to most of the projects so that success chances could be greatly increased. The five Project Management Process Groups highlight the integrative nature of projects and project management.
References


