CUSTOMER CENTRIC PROCESS MANAGEMENT

Table of Contents

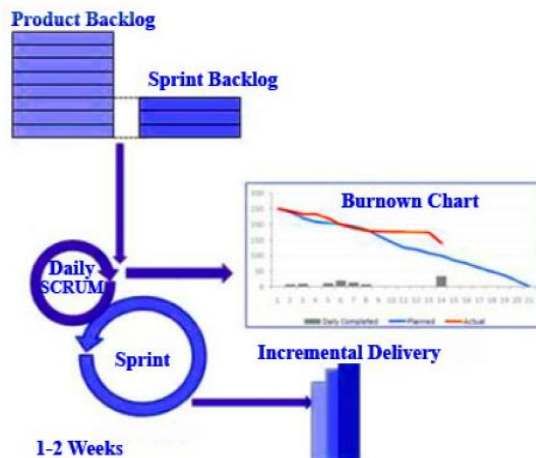# White Paper | Customer Centric Process Management

1. Compvue's process

Our customer-centric process' main objective is to get more CSAT by way of welcoming requirement changes and managing scope efficiently to create competitive advantages for our customers. Our process collaborates with product owners, business people, and development teams to keep them seamlessly updated and synched. We give continuous attention to quality and pride ourselves on our technical excellence. Our efficient task force's expertise lies in our ability to stay on top of all processes and meet all deliverables successfully and in a timely manner.

2. Process Architecture

Our adaptive, lightweight process architecture helps our Scrum team to meet deliverables as planned. This team is comprised of a Product owner, Scrum master, and execution team. The execution team includes BAS (if needed), Team Leads, developers, and Testers. The product owner collects the requirements and updates the product Backlog. Before the start of a spring, Spring Backlogs are updated and any line items in the Sprint Backlog are not allowed to change after the sprint is started.

In most projects, the account manager and product owners are available onsite and closely coordinate with the client to obtain and meet requirements. The Scrum Master collaborates with stakeholders and is available along with the development and testing team offshore. The Scrum Master facilitates resources and meeting and most of the collaboration happens through the Scrum master. Product owners and account managers take care of delivering a product walkthrough/demo to the client at the end of every sprint.

A sprint is a fixed period allocated for development and testing which lasts from one-to-two weeks and is an incremental delivery model. Our team has a daily Scrum meeting with project stakeholders, during which we evaluate tasks executed on the previous day, clarify questions for planned development, and schedule plans for the current day. This meeting is usually short and effective, and does not require much time from the stakeholders.

3. Process Checklist

3.1 Initiation

We have a clear and comprehensive set of processes during the initiation of a project. Before we kick off any projects, we first identify our resources. These include people, special hardware, documents related to business, process, standards, lessons learned, and best practices.

Practise
On-boarding of People for the project.
If any special hardware is required, we ensure it is available before kick-off development. Ensuring appropriate members in the team have studied pre-project documents related to business, process, standards, lessons learned, and best practices.
Ensure that everyone on the team is available in a mailing group. Also ensure whether they know each other, their contact information, and best times to call.

3.2 Requirements

We mandate documents detailing requirements for any projects requiring more than a two-man week effort. This document is provided by the product owner and followed by any elaborations and clarifications that the dev team has. If there is need for a wireframe/mockup, we either get these from our clients or we'll prepare them in-house (with client approval). We also capture non-functional requirements during this phase and ensure the environment availability for Dev and the QA team.

Practise
Preparing requirement documents for practise.
Clarification from client for all project-related questions.
Client approval for wire-frame/mock-up for GUI.
Receiving all non-functional requirements.
Defining of assumptions with related action items.
Defining of risks with associated action items.
Validation of environment and source code availability in source control.

3.3 Planning

Planning for iterations is made based on requirements (for specific projects iterations can be further broken down into user stories). Tasks are created based on planning and will be allocated to the execution team. During this phase we create/update system architecture, subsystem design, integration points, data models, and data definition.

Practise
Iteration Plan –Does the project have a plan with resource estimation?
Wherever there is complexity, are there detailed stories being written for the features and reviewed as per the iteration plan?
Requirements –Stories are broken down into specific requirements as per plan?
Are weekly tasks being created based on the iteration plan, stories, and requirements?
Design System Architecture: Is it documented before iteration one?
Test Plan: Is the test scope checklist with all specified details for iteration one?
Sub-system Design –Class Diagram: Is it being created and reviewed as per the iteration plan?
Sequence –As applicable: Is it being created and reviewed as per the iteration plan?
Is existing architecture enough to meet requirements or are updates required?

3.4 Execution

During our execution phase, we can start deliver output to clients. To ensure quality deliverables, we have mandated the below practice for quality assurance. Deliverables that are not designed according to this process are not accepted.

Practise
Are daily calls and stand-up meetings happening regularly?
Is the build released from the source control?
Each build release is accompanied by a manifest, release notes, and known issues?
Is every build release accompanies by a Unit test case output and Code review with fixes?
Is the demo and walkthrough delivered to the client by an on-site coordinator/released to UAT?
Are test cases being created and reviewed for all features in advance of feature delivery?
Is a QA Defect report being generated & shared to review every day from the third day of development?
Is a weekly status report being generated and shared to review every day?

3.5 Monitoring and Control Execution

Daily internal calls and standup meetings help us review and plan our daily activities. During these meetings, we track action items, calls, and build and review the approach documents for the project.

Practise
Are daily calls and stand-up meetings happening regularly?
Are all walkthrough defects raised by the client captured and categorized accordingly in the internal QA defect tracker?
Are metrics being generated for defects ratio and the escape defects ratio?
Is the number of test cases more than or equal to the requirements?
Is there a test coverage report being prepared by test engineers daily?
Are CRs being created, sent to approval, and controlled? Is the scope updated against CRs?
Are risks being monitored and controlled?
Is traceability maintained from Features? Stories? Requirements?

3.6 Project Closures

We only close projects after receiving a customer satisfaction report, feedback, and closure approval from the client. We use this feedback to learn for future executions.

Practise
Has the client signed off?
Have we collected all monies for the project which are not part of account management?
Is the lessons learned document updated with all relevant information?
Have we off-boarded and allocated resources to other projects?

4. Employee Satisfaction

Everyone in our team adheres to this process, which allows them to remain organized and efficient. This allows us to maintain a wide level of employee satisfaction as well as offering our workers the opportunity to learn and practice with new technologies on an industry-grade level.