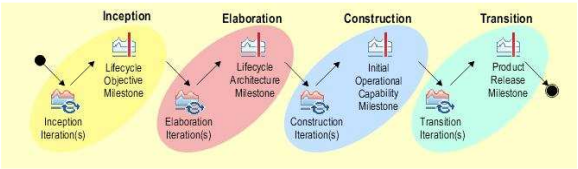
Alexander Ramsey  
SWEN 603  
August 02, 2016

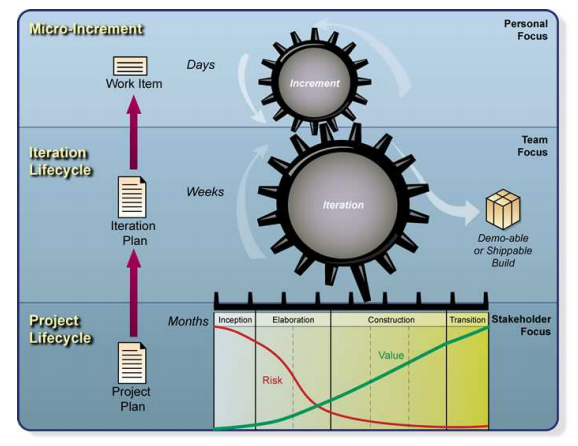
Rational Unified Process Programming Methodology

The Unified Process (UP) development methodology made its mark primarily as the Rational Unified Process in the late 1980s. It was developed by Rational Corporation, a company acquired by IBM in 2003 (Taft, 2002). The terms Unified Process and Rational Unified Process are often interchanged and generally indicate the use of the Rational suite of software planning tools. Similar to Rational Unified Process is the OpenUP methodology. This methodology is adapted from the RUP in that it provides similar guidelines proposed by IBM. OpenUP is generally used in the Eclipse IDE suite as an open source alternative.



It is essentially the same as RUP in that it favors small incremental iterations and a strong collaborative environment. OpenUP structures the lifecycle of the project into four phases: 1. Inception, 2. Elaboration, 3. Construction, and 4. Transition. Within each of these iteration steps are some common recommended tasks including planning, requirements gathering, architecture development, testing, and training.

OpenUP is considered to be a minimal methodology, in that much of the guidance for more complex development is found in the IBM processes. OpenUP is considered to be an agile methodology and is open for interpretation by a team. It can be an excellent starting point for a small team with limited concerns.



The diagram shows a typical lifecycle of OpenUP. A work item is a small part of the major project scope; it could be a smaller story if considered in that manner. The iteration plan and project plan serve as large planning documents that are composed of these smaller work items. “The work item can be a use case, a scenario, a supporting requirement or a change request. A context can be specified when a work item is assigned to be developed, thus specifying how broadly a work item is to be developed in that increment. Development may be focused, for example, on a layer (e.g., user-interface, business logic or database access), or on a component. Whether a context is specified or not, the developer's responsibility is to create a design and implementation for that work item, and to write and run developer tests against the implementation to make sure the implementation works as designed, both as a unit and integrated into the code base.” (Eclipse Software, 2007).

References:

Eclipse Software (2007, August). Introduction to OpenUP. Retrieved August 2, 2016, from https://eclipse.org/epf/general/OpenUP.pdf

Taft, D. K. (2002, December 06). IBM Acquires Rational. Retrieved June 23, 2016, from http://www.eweek.com/c/a/Desktops-and-Notebooks/IBM-Acquires-Rational