Tytuł szkolenia: Techniczna strona Agile dla programistów

Kod szkolenia: AGILE-PROGR

Wprowadzenie

W ostatnich latach bardzo zyskały na popularności metodyki zwinne wytwarzania oprogramowania (Agile software development). Zamiast Team Leadera możemy spotkać Scrum Mastera, poranne spotkanie nazywane jest Scrumem, a wymagania zebrane są w postaci “historyjek” (ang. User Stories). Metodyki zwinne usprawniły produkcje oprogramowania na wielu płaszczyznach. Niestety jednak nadal często nie spełniają pokładanych w nich nadziei. Wytwarzane iteracyjnie w sposób zwinny oprogramowanie dalej zawiera dużo błędów i trudno je rozwijać. Dodawanie nowych funkcjonalności zajmuje coraz więcej czasu i powoduje powstawanie nowych błędów. Powodem takiego stanu rzeczy jest częste pomijanie technicznych aspektów podczas transformacji do metodyk zwinnych. Należy pamiętać, że ostatecznym produktem, który otrzymuje klient jest oprogramowanie. Aby zapewnić mu odpowiednią jakość potrzebne są odpowiednie narzędzia i techniki.

Adresaci szkolenia

Adresatami szkolenia są programiści Java pragnący usprawnić proces wytwarzania oprogramowania, w którym biorą udział. Mile widziana przynajmniej podstawowa znajomość programowania w języku Java.

Cel szkolenia

Szkolenie obejmuje przedstawienie procesu tworzenia oprogramowania z naciskiem na zapewnienie wysokiej jakości oprogramowania i minimalizację ryzyk w projekcie tworzonym przy użyciu metodyk zwinnych. Przedstawione zostaną zasady tworzenia czystego, łatwego w utrzymaniu kodu i testowania go z wykorzystaniem zasad TDD. Stworzony kod zostanie umieszczony w repozytorium kodu źródłowego Git. Skąd będzie pobierany i testowany z wykorzystaniem Jenkinsa, aby zaprezentować ideę Continus Integration i Continous Delivery. Przedstawione zostaną typowe scenariusze wykorzystania przedstawionych narzędzi, aby umożliwić członkom zespołu sprawną integrację swoich zmian w kodzie z wykorzystaniem wsparcia Gita dla tworzenia i merdżowania branchy (gałęzi). Proces obejmuje także inspekcję kodu oraz automatyczne testowanie zmodyfikowanej wersji aplikacji przed zmergowaniem jej do mastera.

Czas i forma szkolenia

- 14 godzin (2 dni x 7 godzin), w tym wykłady i warsztaty praktyczne.

Plan szkolenia

1. Krótkie wprowadzenie do Agile
    a. główne założenia
    b. przedstawiciele: Scrum, Kanban, XP ..
    c. dlaczego Agile wymaga nowych narzędzi, praktyk i zasad
2. Czystość kodu
    a. czytelność (formatowanie, nazwy, długość metod i klas, komentarze)
    b. łatwość pielęgnacji (zasady SOLID, DRY, KISS)
    c. narzędzia do analizy (CheckStyle, PMD, FindBug ...)
    d. inspekcja kodu (ang. code review), pair programming
3. Testowanie
    a. testy automatyczne,
    b. dlaczego pokrycie testami nie wystarcza
    c. zasady TDD
    d. wykorzystanie narzędzi (JUnit, Mockito)
4. Repozytorium kodu
    a. dlaczego nie SVN
    b. GIT
    c. typowy workflow
    d. GitHub
5. Git - przydatne funkcjonalności
    a. zdalne i lokalne branche
    b. rebase
    c. reset
    d. cherry-pick
6. Budowanie aplikacji
    a. zarządzanie zależnościami
    b. uruchamianie testów i dodatkowych narzędzi
    c. deployment
    d. przykładowe rozwiązania: Maven lub Gradle
7. Ciągła integracja
    a. ciągłe budowanie i testowanie aplikacji
    b. integracja z systemem kontroli wersji
    c. automatyzacja procesów (deployment, promocja)
    d. Jenkins