

# Zarządzanie Projektami Informatycznymi

dr hab. inż. Radosław Wajman

dr inż. Paweł Kapusta

dr hab. inż. Robert Banasiak

---

## Zadanie projektowe

### Cel ćwiczenia:

Celem ćwiczenia jest wytworzenie systemu informatycznego respektując ramy postępowania rozwoju produktu wg metody SCRUM oraz technikę wdrażania systemu opartą na ciągłej integracji. W ramach zadań projektowych studenci zobowiązani są do stosowania systemu kontroli wersji kodu źródłowego.

### Przygotowanie środowiska developerskiego (pipeline):

W pierwszej kolejności studenci zostaną dobrani w 4-ro osobowe grupy projektowe. Prowadzący losowo wybiera osoby, które będą pełniły rolę „SCRUM Master’a” w grupie. Każda z tych osób dobierze do swojej grupy kolejne trzy. W każdej z grup jej członkowie ustalą między sobą następujące role:

- operations,
- developer (x2),
- tester,

przy czym osoba pełniąca rolę „SCRUM Master’a” jest zobowiązana do pełnienia dodatkowej roli z powyżej wymienionych. Prowadzący zajęcia pełnić będzie rolę Product Owner.

Każda z grup:

- ma przypisane repozytorium w organizacji <https://github.com/IIS-ZPI> o nazwie ustalonej w ramach zadania ćwiczeniowego,
- ma utworzony zespół projektowy i przypisany do repozytorium, (w przypadku, gdy powyższe nie jest zrealizowane, osoba SCRUM master’a zbiera wszystkie ID GitHub’a od członków swojej grupy, dopisuje ustalone role i podaje je prowadzącemu wraz z proponowaną nazwą swojej grupy),
- archiwizuje w repozytorium poprzednie zadanie.

Po potwierdzeniu od prowadzącego zrealizowania powyższych operacji, grupa przystępuje do etapu projektowania. Studenci mają prawo do tworzenia własnych branch’y (featerów), definiowania issues itp.

Kolejnym zadaniem jest skonfigurowanie systemu ciągłej integracji (CI). Grupa samodzielnie decyduje się na jedno z licznych rozwiązań w zależności od wybranej platformy programistycznej. Należy jedynie pamiętać, że niezależnie od wyboru systemu, należy dać pełny dostęp prowadzącemu do systemu CI. W tym celu SCRUM Master grupy umieszcza informacje o konfiguracji systemu CI w pliku README.md w repozytorium. Tam również powinien być odsyłacz do Backlog (np. Trello, GitHub). Jeżeli grupa korzysta z systemu CI na lokalnym serwerze, w repozytorium powinien znaleźć się katalog CI\_CFG, w którym wgrany zostanie plik konfiguracyjny z ostatecznymi ustawieniami systemu lub raport z wykonanych ustawień.

### Polecenie ćwiczeniowe:

Do zaliczenia przedmiotu wymagane jest opracowanie systemu informatycznego, który będzie spełniał następujące funkcjonalności:

- dane, na bazie których system będzie realizował analizy statystyczne i obliczenia, będą pochodzić z platformy API NBP dostępnej pod adresem <http://api.nbp.pl/>,
- w ramach swojej funkcjonalności system powinien oferować następujące analizy statystyczne:
  - wyznaczenie ilości sesji wzrostowych, spadkowych i bez zmian w okresach ostatniego 1 tygodnia, 2 tygodni, 1 miesiąca, 1 kwartału, pół roku oraz 1 roku dla wybranej przez użytkownika waluty,

- miary statystyczne: mediana, dominanta, odchylenie standardowe i współczynnik zmienności za okres ostatniego 1 tygodnia, 2 tygodni, 1 miesiąca, 1 kwartału, pół roku oraz 1 roku dla wybranej przez użytkownika waluty,
- rozkład zmian miesięcznych i kwartalnych w dowolnych wybranych przez użytkownika parach walutowych np. EUR/USD. Analiza polega na wyliczeniu w kolejnym dniu sesji zmiany w stosunku do sesji poprzedniej.

#### Informacje uzupełniające:

- język implementacji systemu oraz środowisko developerskie dowolne,
- komunikacja systemu z użytkownikiem może odbywać się w dowolny sposób (tj. konsola, graficzny interfejs) jednakże ma udostępniać użytkownikowi możliwość sprecyzowania szczegółowych parametrów analizy statystycznej (np. rodzaj analizy, okres rozliczenia, rodzaj walut(y) itp.), której oczekuje,
- wynik przeprowadzonej analizy program może zaprezentować w dowolnej formie (np. tabeli, wykresu, eksportu do pliku CSV itp.),
- przypadku, gdy **grupa jest 5-cio osobowa** aplikacja musi posiadać interfejs graficzny oraz prezentować wynik przeprowadzonej analizy w formie wykresów.

#### Uwagi do etapów projektowania

1. Program powinien spełniać wszystkie kryteria opisane powyżej,
2. Udział poszczególnych osób w grupie w etapach projektowania,
3. Definiowanie issues, raporty z odbytych sprintów -> Product Backlog, BurnDown Chart (tu uzależniony od wybranego narzędzia wspierającego projektowanie w SCRUM)
4. Odbycie przynajmniej dwóch sprintów przy współudziale Product Owner,
5. Utworzenie pipeline'u oraz implementacja testów (jednostkowych, funkcjonalnych) i ich wykonanie,
6. Raporty z przeprowadzonych testów wraz z dokumentacją wykrycia i naprawy błędów (zasadniczo, co commit i build w CI to wykonanie testów i oddzielny raport) – szablon raportu w załączniku,
7. Czytelny i przejrzysty kod.

#### Odwołania

1. SCRUM
  - a. <https://mamopracuj.pl/jak-zostac-scrum-masterem-6-kluczowych-podpowiedzi>
  - b. <https://www.qagile.pl/scrum/scrum-opis/>
  - c. <https://kierownikprojektu.com/2017/05/31/scrum-w-pigulce-cz-1/>
2. w kontekście statystyk kursów walut:
  - a. [http://wyborcza.biz/Waluty/0,111976,,,Statystyki\\_waluty,USD,EBC.html](http://wyborcza.biz/Waluty/0,111976,,,Statystyki_waluty,USD,EBC.html)
  - b. <https://kantorlista.pl/porady-najwazniejsze-pojecia-dotyczace-wymiany-walut-453>
  - c. <https://analizy.investio.pl/analiza-statystyczna-zmienności-na-ryнку-forex/>
3. GitHub tutorial: <https://guides.github.com/activities/hello-world/>
4. Continuous Integration:
  - a. <https://medium.com/jumperiot/how-to-build-a-continuous-integration-and-delivery-flow-for-embedded-software-b0b5bf220a2>
  - b. <https://medium.com/quick-code/top-tutorials-to-learn-jenkins-ci-for-testing-automation-93c7ac068f66>
  - c. <http://namiekkko.pl/2015/05/30/ciagla-integracja-aniol-stroz-dobrego-programisty/>
  - d. <https://productvision.pl/2016/continuous-integration/>
5. Testowanie oprogramowania
  - a. <http://testpro.pl/poradnik-pocztakujacego-testera-jak-dobrze-zglaszac-bledy/>
  - b. Przykładowe raporty: <http://mrbuggy.pl/mrbuggy4/docs/Raporty-z-testowania-Mr-Buggy-4.pdf>