

Demo

Demo



Praca z TFS

Demo

Pytania?



Wprowadzenie

Team Foundation Server

Wprowadzenie

Agenda

Wprowadzenie

Wprowadzenie

Wprowadzenie

Wprowadzenie



Wprowadzenie

Wprowadzenie

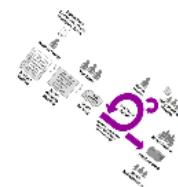


Wprowadzenie

Wprowadzenie

Wprowadzenie

Wprowadzenie



Team Foundation Server

Dawid Zarzycki

Agenda

Metodyka Scrum

- ogólny opis
- zespół
- zdarzenia
- artefakty
- przykład

Wstęp TFS

- ogólny opis
- sposoby dostępu
- repozytorium kodu
- ...

Praca z TFS

- wymagania
- instalacja
- podłączanie
- podpięcie projektu
- praca z repozytorium
- ...

Scrum

Scrum

Ogólny opis

iteracyjna metodyka
prowadzenia projektów

zaliczana do metodyk
zwinnych (ang. agile)

pełna metodyka oraz definicja
została sformalizowana przez
Kena Schwabera w 1986

Scrum

Zespół

product owner

osoba będąca łącznikiem
pomiędzy zespołem a klientem

scrum master

odpowiada za to, żeby proces
wytwarzania i oddawania produktu
przebiegał płynnie i bez zakłóceń

team members

od 5 do 9 osób, konkretnie nie
określonych, zależy od scrum mastera

Scrum

Zdarzenia

cykl wytwarzania produktu
podzielony jest na sprinty

sprint

najmniejsza jednostka
czasowa w scrumie

planowanie, realizacja i
podsumowanie

powinny trwać taki
sam przedział czasu

sugerowane: od 2
do 5 tygodni

Scrum

Zdarzenia

sprint

- nie można dokonywać zmian mających wpływ na realizację celów sprintu
- nie można zmieniać składu Zespołu w trakcie sprintu
- nie można obniżać celów jakościowych
- czas zadań nie powinien się zmieniać

Scrum

Zdarzenia

codzienny scrum

spotkanie zespołu, 15 min na stojąco,
scrum master jako moderator

planowanie sprintu

początek sprintu, 4h-8h, z puli zadań dla
całego projektu wybierane są te, które
zostaną wykonane w tym sprincie

przegląd sprintu

podsumowanie dokonywane za zakończenie
sprintu, co się udało, a co nie udało się zrobić,
omówienie problemów, od 2h do 4h

retrospektywa sprintu

zespół analizuje co dobrego/złego wydarzyło
się w ostatnim sprincie, propozycję popraw,
co trzeba dalej kontynuować, scrum master
pilnuje, aby każdy się udzielał, maks 3h

Scrum

Artefakty

product backlog

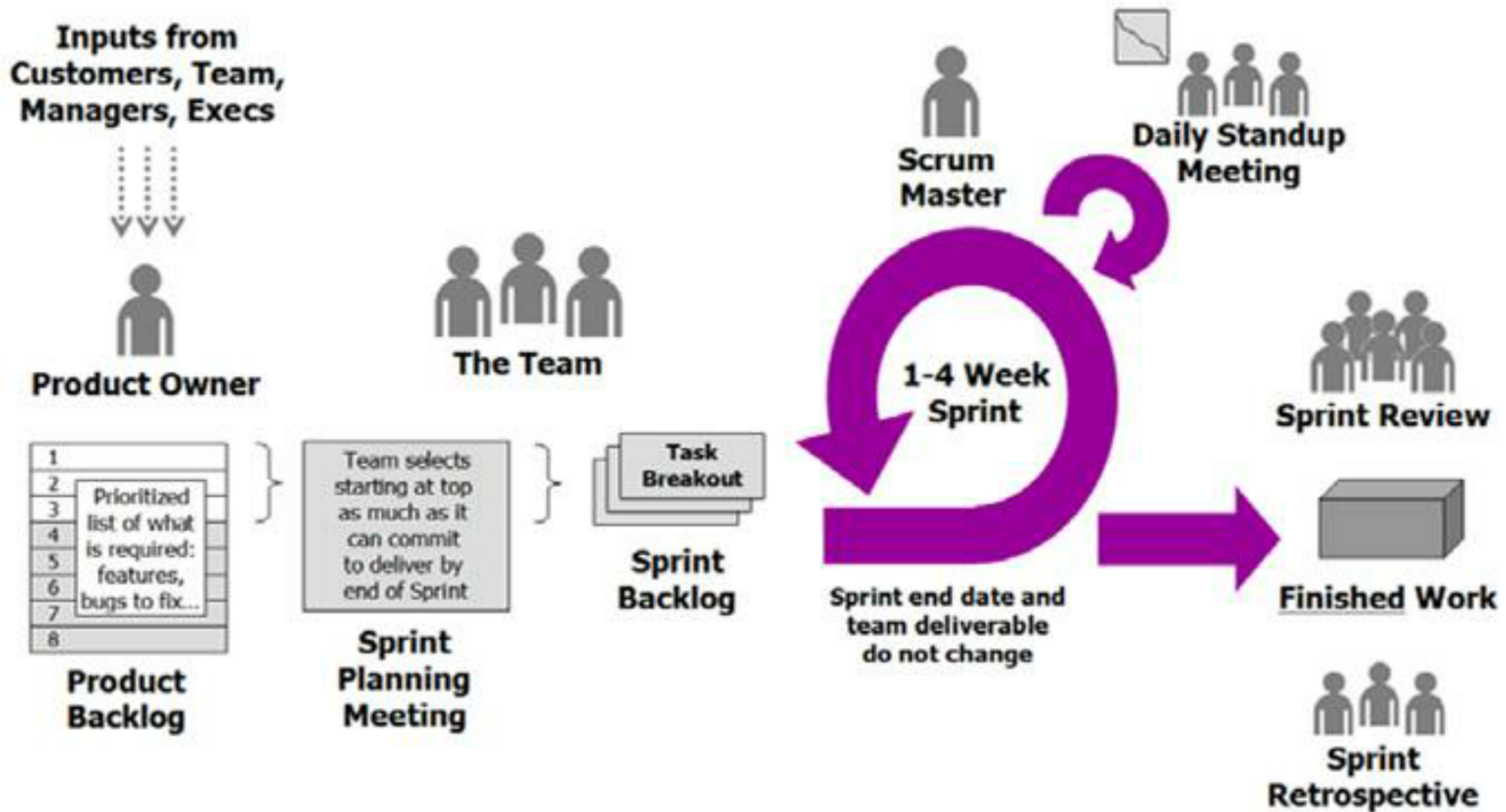
rejestr produktu, zwykle zawiera zadania
w postaci historyjek (User Stories),
odpowiada za niego product owner

sprint backlog

rejestr sprintu, elementy / zadania wchodzące
w skład zdefiniowanego celu dla danego
sprintu, decyduję team i scrum master

effort

wskaźnik, mówiący o tym co do tej pory udało się
zrealizować, ilość zrealizowanych funkcjonalności



Wprowadzenie do TFS

Ewolucja narzędzi

Edytory
Kompilatory

Zintegrowane środowiska
programistyczne

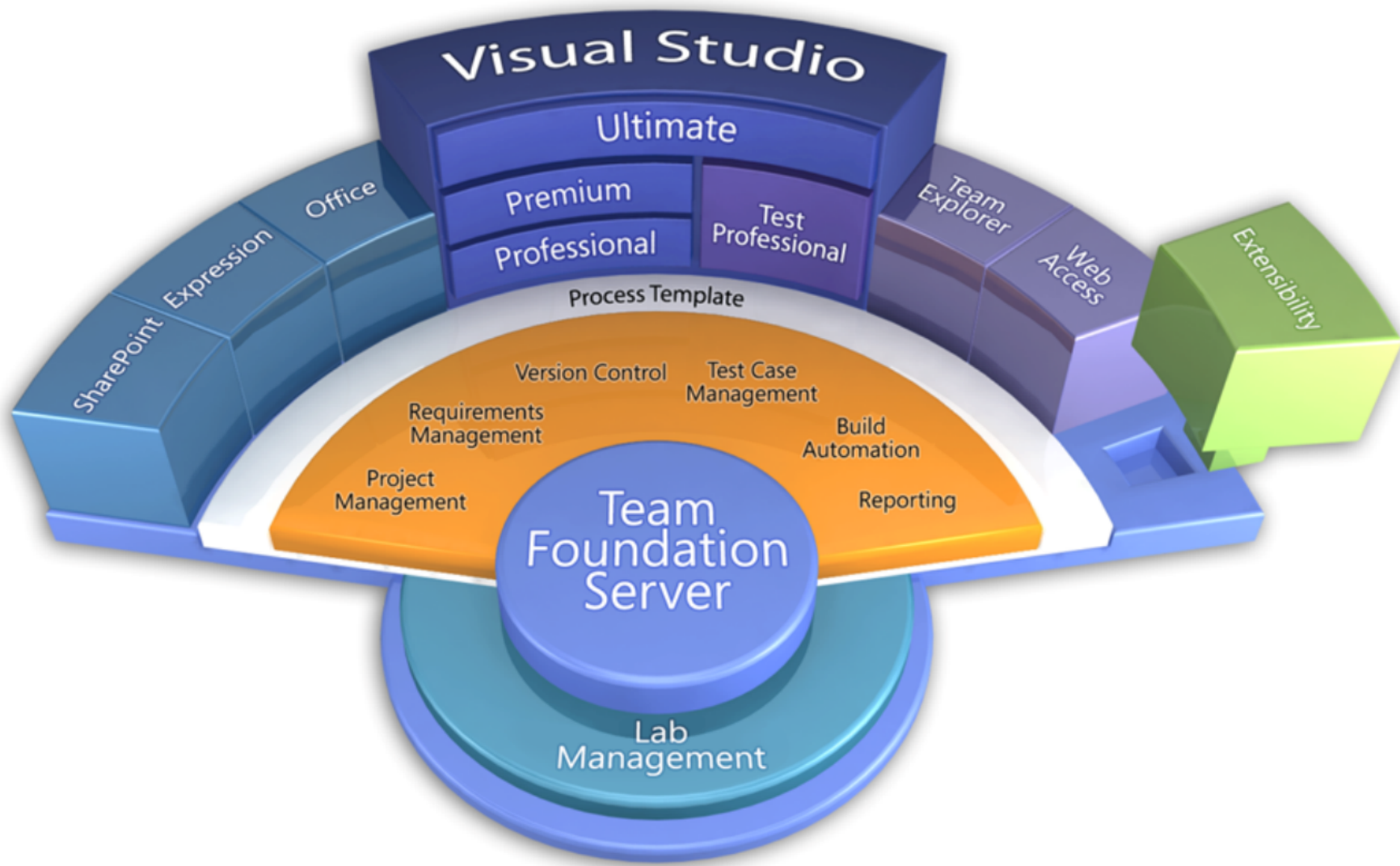
+

Skrypty,
rozszerzenia, etc.

Zintegrowane platformy
programistyczne



Platforma Visual Studio



Dlaczego warto poznać TFS?

- większy porządek w kodzie
- nie marnujemy czasu na organizowanie pracy
- skupiamy się na celach projektu
- nasze oprogramowania będą wyższej jakości
- łatwo włączyć do zespołu nowe osoby

Większe szanse na powodzenie projektu!

Sposoby dostępu do TFS

Developer

Visual Studio

Office

Excel, Project

Tester

Test Manager

Wszyscy

Web Acess
własne narzędzia

TFS Everywhere
Eclipse

Repozytorium kodu

Changeset

punkt w czasie życia
repozytorium, 'revision'

Check-In

"wrzutka" do
repozytorium, 'commit'

Workspace

zamapowana ścieżka serwerowa na
lokalną, zmiany lokalne nie wpływają na
repo, dopiero po check-in

Repozytorium kodu

Work Item

"jednostka robocza"

różne typy: Task, Bug, Issue, User
Story, Shared Step, Test Case

każdy typ posiada swój workflow

możemy tworzyć własne typy

Repozytorium kodu

Shelving

wstępne umieszczenie kodu w
repozytorium

bez wykonania check-in

odkładamy aktualną pracę na półkę

np. współdzielenie postępów prac,
recenzowanie kodu

Repozytorium kodu

Branching & Merging

scalanie gałęzi kodu

stabilne release

rozgałęzianie kodu

oznaczanie milestone

wersje zaakceptowane przez klienta

filozofie branchowania

Repozytorium kodu

Labels

coś jak branch, ale nie tworzymy nowej gałęzi

nie możemy rozwijać jako oddzielnego projektu

stabilne release

wersje zaakceptowane przez klienta

referencje do plików

Automatyzacja

Buildy

kompilacja kodu na różne maszyny

wirtualne środowiska uruchomieniowe
konfigurowalne np. po check-in, raz na tydzień

Testy

po udanej kompilacji uruchamiane są
testy (unit test, web test, load test)

powiadomienia np. email,
generowanie raportów, publikacja

Raportowanie

Automatyzacja

wykresy przyrostu, poprawionych
bugów, wykrytych bugów etc.

szczegółowe statystyki

statystyki poszczególnych osób w teamie

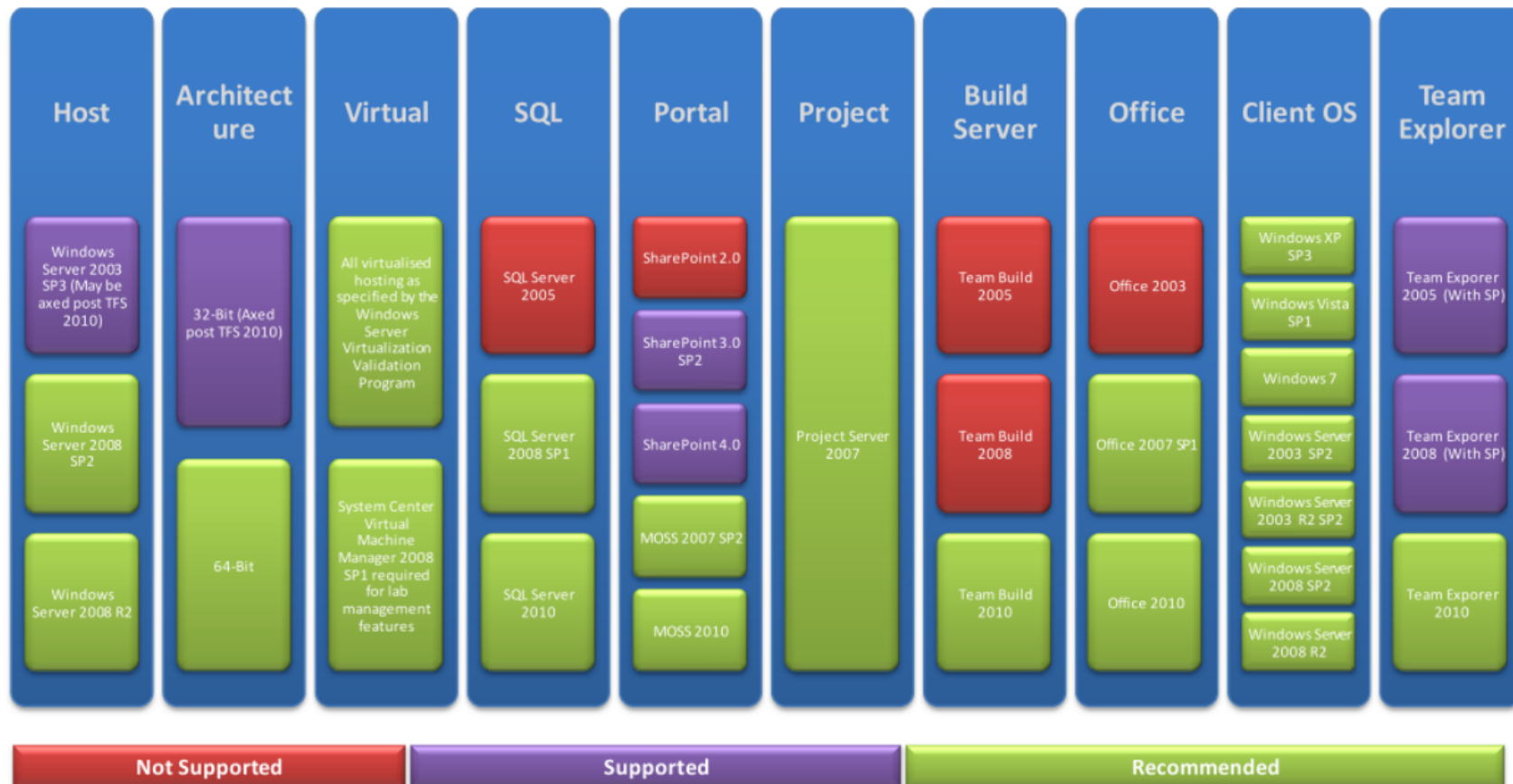
raportowanie o nieudanych testach - powiązania
z check-inami etc.

SharePoint

Praca z TFS

Planowanie przed instalacją

Planning for TFS 2010 --- System Requirements



Wymagania

Wielkość zespołu	Konfiguracja	CPU	Pamięć	Dysk
Mniej niż 250 osób	Jednoserwerowa (Team Foundation Server i baza danych na tym samym serwerze).	1. jednordzeniowy procesor taktowany zegarem 2.13 GHz	2 GB	1 dysk – 7200 obr./min (125 GB)
250 do 500 osób	Jednoserwerowa	1. dwurdzeniowy procesor taktowany zegarem 2.13 GHz	4 GB	1 dysk – 10000 obr./min (300 GB)
500 do 2200 osób	Dwuserwerowa (Team Foundation Server i baza danych na osobnych serwerach). Zalecenia dla serwera przeznaczonego na Team Foundation Server:	1. czterordzeniowy procesor Intel Xeon taktowany zegarem 2.13 GHz	4 GB	1 dysk – 7200 obr./min (500 GB)
	Zalecenia dla serwera bazy danych (500 do 2200 użytkowników):	1. czterordzeniowy procesor Intel Xeon taktowany zegarem 2.33 GHz	8 GB	Macierz dyskowa SAS – 10000 obr./min (2 TB)
2200 do 3600 osób	Dwuserwerowa Zalecenia dla serwera przeznaczonego na Team Foundation Server:	1. czterordzeniowy procesor Intel Xeon taktowany zegarem 2.13 GHz	8 GB	1 dysk – 7200 obr./min (500 GB)
	Zalecenia dla serwera bazy danych (2200 do 3600 użytkowników):	2.czterordzeniowe procesory Intel Xeon taktowane zegarem 2.33 GHz	16 GB	Macierz dyskowa SAS – 10000 obr./min (3 TB)

Instalacja

- Google: Team Foundation Server
 - 90 dni triala
- MSDNAA: Visual Studio 2010 : Team Foundation Server
 - Pełna wersja - Volume License
- Od wersji 2010 mamy do dyspozycji Wizard: Next, Next, Next, Thank You :)

Demo

- widok administratora
- widok developera
- podłączanie
- tworzenie team project
- wersjonowanie solution

Demo

- pobieranie / umieszczanie zmian
- przeglądanie historii
- porównywanie wersji
- konflikty
- cofanie zmian
- adnotacje

Demo

- shelving
- labels
- check-in policies
- branching & merging

Pytania?

Dziękuję za uwagę :)