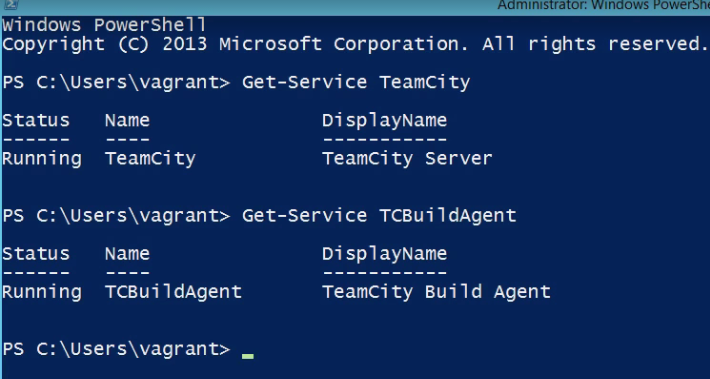
1. W win sciagam, instaluje default, odpalam agenty, po wejsciu konfig, jaka baza, moge na agentspr czy chodzi iz konsoli



1. Linux

- Tar zxf

- bin/runAll.sh start– startuje serwer i agenta, jest też stop, wypisuje o startach serwer agent i gdzie logi

- 8111 jest default to na localhost odpala

- jak padnie to znowu start robimy

- na prod wazna inna baza

- a Agents u gury mozna ich status obadac

1. Docker

- na hub.docker.com/jetbrains są konfiguracje

- potrzebujemy takiego yml jak tam, u nas odpalic, co tworzy image dockerowe

- odpalamy z odpowiednimi opcjami by wstał serwer i image dla agentów

1. Na view jest panel gorny i po prawej administrations, tam konfiguracje i np. Users – dodawanie usera

- wykly user nie moze dodawac projektow i nie ma admin linka

- w authetication mozna zmienic sposob autentykacji , default trzymane przez teamcity, ale moze byc LDAP czy widdows account

- groups – mozna tworzyc, by dawac up=rawnienia dla grup uzytkownikow nie pojedynczych

1. Projekt

- na github przycisk clone repository

- mozna miec sonar-project.properties

- też mvnw i mvnw.cmd

- mozna zainstalowac z konsoli jako admin chockolatey do pobierania

- tree –L 3

- obadac ten proj jego, strukture

- teamcity-course-spring-petclicic

https://github.com/g0t4/teamcity-course-spring-petclinic

1. Projekt

- klikamy create project

-- mozna repo url, github i bitbucketcloud, bierzemy pierwsze

- projekty sa jak foldery w filesystem, potrzebuja build konfiguracji w srodku

- pobieramy url z github, jest publiczny, to bez kredek

- dalej project name i build configuration name

- czyli w folderze parent (project root) mamy folder nasz\_proj a w nim file zwany build

- potem wykrywa build stepy, maven, gradle, ...

- można autodetect again potem

- mozna dodawac stepy, edytowac, zmieniac polecenia

1. Konfigi

- teamcity odnajduje zainstalowaną wersję jdk na maszynie, mozna wybierac na build

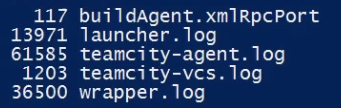
- team city agent, to tylko częśc team city co wykonuje buildy

- restart-service TCBuildAgent

- po restarcie agenta pojawiaja się wersje zainstalowane w iędzyczasie do wyboru (po wczesniejszym uruchomieniu)

- w logach mozna znaleźć na starcie, że znalazl jave





Nas interesuje teamcity-agent.log

<https://msdn.microsoft.com/en-us/powershell/scripting/setup/installing-windows-powershell>

windows power shell

1. Run

- w build queue widzimy ile czeka , a w agents ile zajętych agentow, jaki poziom, niebieskie cale to wszyscy

- przenosi po kliknieciu do „build configuration home”, możemy tam sami, na niej wyniki buildow

- klikamy na konkretny build by zobaczyc jego wyniki, z tad mozna szybko przejsc do settings – linki pomiedzy nimi, klikając kolo wyniku build, rozwija kombo i build log mozemy zobaczyc

- checkout directory to teamcity workspace

- mozemy isc na serwer do checkou directory, tam są normalne pliki projektu

- jak sie nic nie zmienilo w projekcie i odpalamy ponownie, to pliki sie nie odswierzaja, bo jest cache

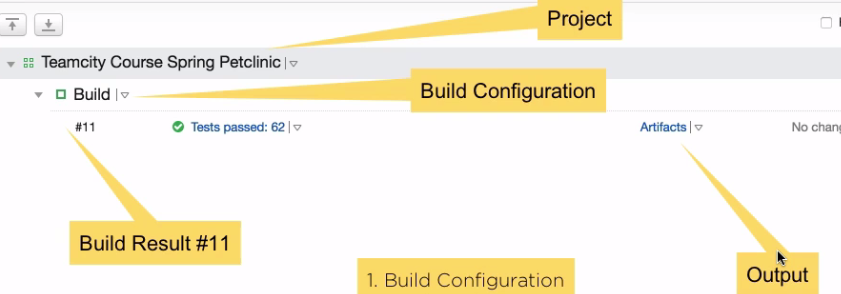
- dobra praktyka to przed compile dac clean ,żeby nowe budowanie bylo

- na general settings, w settings mamy artifact paths, definiuje jakie artefaky outputem z naszego builda – np. War file

- w wielu miejscach mamy przy labelkach ?, po kliknieciu do dokumentacji odnosza

- w historii build, w kolumnie artifacts, po zbudowaniu mogę pobrac artefakt

- kolejne buildy maja kolejne artefakty



1. Triggery

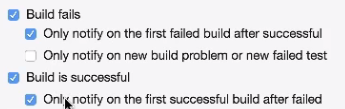
- mamy w konfiguracji taką pozycje, odpala budowania na jakieś zdarzenie

- mamy tam tez ssh keys, pewnie do stasha

- tez na przykład sonarqube servers

- dodatkowo mozemy ogladac testy integracyjne i konfigurowac notyfikacje

1. Moze sie zdarzyc, ze w agent requirements nie bedzie czego, co potrzebujemy do zainstalowania, jest to wykryte i musimy to zainstalowac na agencie
2. Przy uruchamianiu testow mozna wybrac by najpierw wykonywalo ostatnio co nie przeszly – szybkie sprawdzenie
3. W konfiguracji mamy trigger, schedule trigger, maven dependency – rozne typy, vcs trigger po zmianie w kodzie
4. Notyfikacje gdy cos padnie, mozna na poziomie projektu, ale można globalnie na poziomie usera, wchodze na usera i klikam notification rules, moze być emaij, jabber, windows tray, określamy ze sle notyfikacje kidy np. Build padnie, testy nie przejda, mozemy okreslic nap. Ze po build which containc my changes



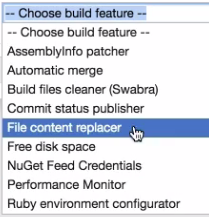
1. Ustawienie wysylania emaili

- mozna uruchomić email server na dokerze, np. mailhog

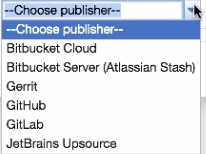
- wchodze na administration, email notifier i tam ustawiam

1. Github integration

- na projekcie w build settings jest build features

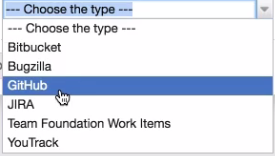


* Wybieram commit status publisher



* Tu wybieram z czy mintegruje

1. Pod projektem mam integracje z issue tracker i moge wybrac z jakim trackerem chce



1. Pod connections mozna 3mac dane polaczenia do wszystkiego