

<p style="text-align: center;">Politechnika Świętokrzyska w Kielcach Wydział Elektrotechniki, Automatyki i Informatyki</p>	
<p>Laboratorium Technologie IoT rozproszone sieci sensoryczne</p>	<p>Lab 1 Packet Tracer</p>
<p>Data wykonania: 30.10.2018r</p>	<p>Autor: Karol Zuba Grupa: 3ID15A</p>

1. Cel laboratorium

Zapoznanie z IoT przy stosowaniu symulacji na Packet Tracer. Zapoznanie z systemem kontroli wersji Git.

2. Opisać zastosowanie i opis systemu kontroli wersji oraz podstawowe komendy w GitHub

System kontroli wersji śledzi wszystkie zmiany dokonywane na pliku (lub plikach) i umożliwia przywołanie dowolnej wcześniejszej wersji. Przykłady w tej książce będą śledziły zmiany w kodzie źródłowym, niemniej w ten sam sposób można kontrolować praktycznie dowolny typ plików.

Pozwala on przywrócić plik(i) do wcześniejszej wersji, odtworzyć stan całego projektu, porównać wprowadzone zmiany, dowiedzieć się kto jako ostatnio zmodyfikował część projektu powodującą problemy, kto i kiedy wprowadził daną modyfikację. Oprócz tego używanie VCS oznacza, że nawet jeśli popełnisz błąd lub stracisz część danych, naprawa i odzyskanie ich powinno być łatwe. Co więcej, wszystko to można uzyskać całkiem niewielkim kosztem.

git init - Inicjalizuje repozytorium GIT w danym katalogu

git add [nazwa_pliku] - Dodaje zmiany we wskazanym pliku do commita

git add - Dodaje wszystkie zmienione pliki do commita

git add -p [nazwa_pliku] - Udostępnia możliwość dodania wybranych linii w zmodyfikowanym pliku do commita

git commit -m "[treść_commita]" - Dodaje opis do commita. Dobrym zwyczajem jest opisanie co ta zmiana wprowadza do kodu w zakresie funkcjonalnym

git add origin [adres_repozytorium] - Ustawia konkretny adres zdalnego repozytorium jako główne repozytorium

git push origin master - Wysłanie zmian do branacha zdalnego

git push -f - Wysłanie zmian do zdalnego repozytorium ignorując konflikty, to znaczy, że jeśli wystąpią konflikty to pliki zostaną nadpisane właśnie wysłaną wersją. Trzeba stosować to bardzo ostrożnie.

git checkout [nazwa_brancha] - Zmienia aktywny branch na wybrany przez użytkownika

git checkout [nazwa_pliku] - Usuwa zmiany w wybranym pliku

git checkout . - Usuwa zmiany we wszystkich zmienionych plikach

git checkout -b [nazwa_brancha] - Tworzenie nowego brancha z aktywnego brancha i przełączenie się na niego

git rebase master - Zaciągnięcie zmian z brancha głównego do brancha aktywnego

git push origin :[nazwa_brancha] - Usunięcie zdalnego brancha

git branch -d [nazwa_brancha] - Usuwanie brancha lokalnie. Nie można usunąć w ten sposób aktywnego brancha

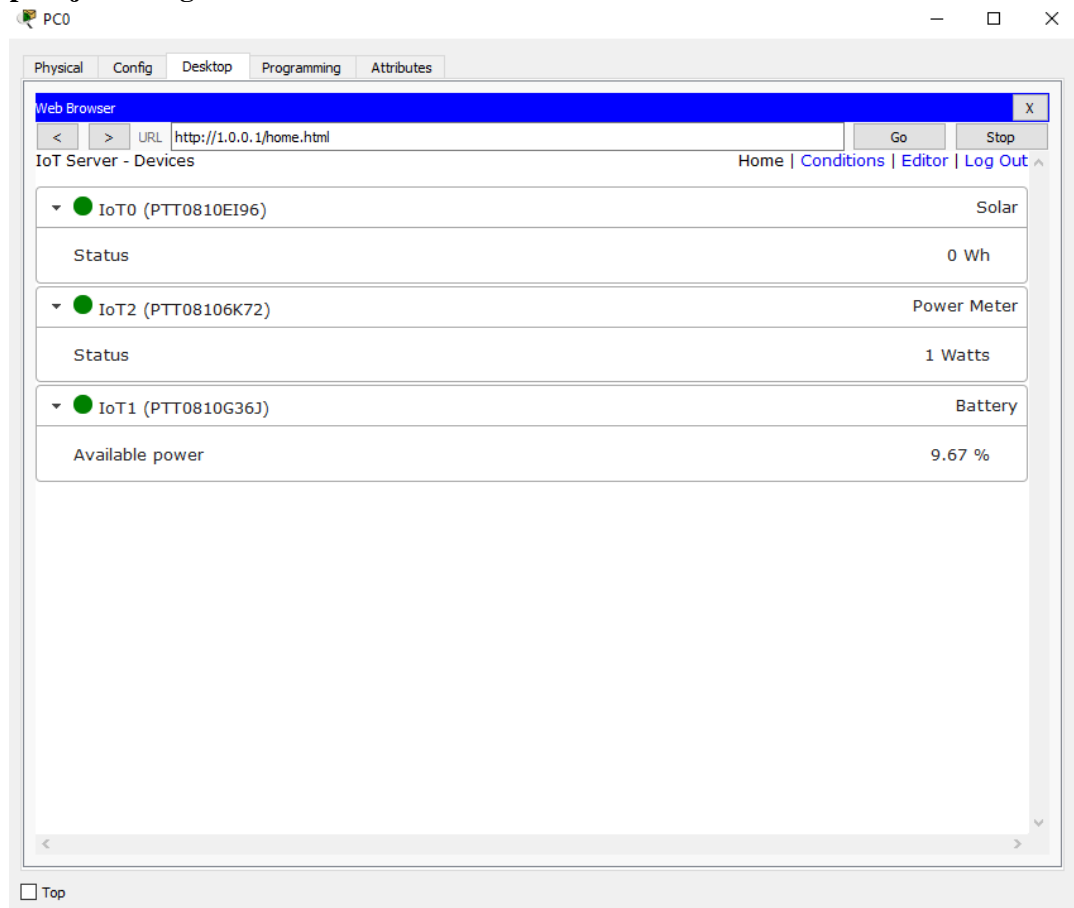
git stash - Dodanie zmienionych plików do pamięci/stosu i usunięcie ich z aktywnego brancha

git pull --rebase - Pobranie najnowszych zmian z aktywnego brancha zdalnego
git stash pop - Przywrócenie zmodyfikowanych plików z pamięci/stosu
git stash clear - Czyszczenie pamięci/stosu
git remote prune origin - Pobranie aktualizacji o usuniętych branchach zdalnych
git fetch --all - Pobranie listy zdalnych branchy
git branch - Wyświetlenie listy lokalnych branchy
git branch -r - Wyświetlenie listy zdalnych branchy
git status - Wyświetlenie listy zmienionych plików
git diff [nazwa_pliku] - Szczegółowe wyświetlenie zmian w wybranym pliku
git reset HEAD - Resetowanie przygotowanych commitów (przed wysłaniem).
 Zmodyfikowane pliki są dostępne do ponownego dodania.
git reset HEAD --hard - usuwanie wszystkich zmian z brancha lokalnego i przywrócenie zmian z brancha zdalnego
git reset HEAD^ --hard - Usuwanie ostatniego commita z brancha>
git reset HEAD^^
git reset HEAD~2 - Obydwie komendy usuwają ostatnie 2 zmiany z brancha. Im więcej daszków (^) tym więcej commitów zostanie usuniętych.
git rebase -i HEAD~3 - Interaktywne zmienianie zawartości, opisów commitów. Commity można łączyć wtedy w jeden duży, zmienić jego opis, itd.

3. Część obserwacyjna

- Zaobserwuj w jaki sposób bateria ładowana jest za pomocą ogniwa fotowoltaicznego.
 Bateria ładowana jest za pomocą kabla IoT (IoT custom cable) bezpośrednio z panelu słonecznego

- Połącz się z serwerem za pomocą PC. Desktop -> Web Browser. Wpisz IP serwera i podaj dane logowania.



Mamy dostęp do wszystkich skonfigurowanych przez nas komponentów dzięki czemu możemy kontrolować ich parametry w czasie rzeczywistym.

- Odłącz panel słoneczny i zaobserwuj działanie systemu przy pomocy PC.

Po odłączeniu panelu słonecznego bateria przestała się ładować, miernik nie ma już co mierzyć i dodatkowo zgasły wszystkie LEDy. PC0 pokazuje że urządzenie Solar Panel jest aktualnie offline, po ponownym podłączeniu panelu solarnego wszystko powinno wrócić do normy.

- Opisz działanie urządzeń. Jakie możliwości rozbudowy posiada symulowany system?

Panel solarny generuje prąd wykorzystując do tego światło słoneczne. Nagromadzony prąd może zasilac urządzenia, m.in. takie jak właśnie bateria.

Bateria przesyła prąd do urządzeń takich jak np. LEDY lub innych urządzeń wymagających zewnętrznego zasilania.

Miernik mierzy aktualną moc w Wh (wato godzinach) tam gdzie się znajduje. W tym przypadku na linii pomiędzy panelem słonecznym, a baterią

System ma wiele możliwości rozbudowy, oprócz diod led można zamontować do tego układu wiele pożytecznych czujników takich jak np. czujnik tlenku węgla powietrza lub czujnik gazu. Dzięki temu że układ korzysta z odnawialnego źródła energii nie musimy się martwić że przy braku prądu z sieci elektrycznej coś złego stanie się w naszym mieszkaniu. Inną możliwością jest również dodanie do układu czujnika ruchu który by automatycznie zapalał diody(oczywiście mocniejsze od tych 10 Watowych) np. pod bramą wjazdową do garażu.

