OPIS WYMAGAŃ DOTYCZĄCYCH ETAPU IV

# Wymagania

W ramach Etapu IV zaplanowano prace dla obydwu grup projektowych.  
Pierwsza i druga grupa powinna zmigrować kody do jednego, wspólnego prywatnego repozytorium github. W tym etapie obydwie grupy połączą wypracowane dotychczas funkcjonalności tak by oprogramowanie odpowiadało na zadane mu funkcjonalności.

# Zarys Prac

W ramach Etapu należy przygotować dokumentacje oprogramowania w wersji 0.4 tj realizującej funkcjonalności:  
**Dla obydwu grup** – uzgodnienie architektury rozwiązania. Realizacja funcjonalnosci:

1. Utworzenie pliku CSV z taskami i informacjami ich dotyczącymi
2. Utworzenie pliku CSV z relacjami pomiędzy taskami
3. Utworzenie pliku CSV z podsumowaniem **n** ostatnich uruchomień planu
4. Utworzenie raportów szczegółowych z poszczególnych uruchomień
5. Utworzenie pliku z raportem błędów z **n** ostatnich uruchomień

**Dla obydwu grup** – przygotowanie prezentacji dotyczącej działania programu oraz przeprowadzenie testów oprogramowania na „żywych przykładach”

# Terminy

**2018-04-170 18:00:00 –** podesłanie prezentacji i dokumentacji **2018-04-19 08:30:00** – spotkanie w celu prezentacji działającego prototypu, testy oprogramowania, dokumentem opisu rozwiązania informatycznego I mail z prezentacja dotycząca działania prototypu i architektura

# Wytyczne

**Ad 1.** Plik CSV z kolumnami zawierającymi wszystkie pozyskane informacje dotyczącymi tasku ,tj : *NAZWA\_TASKU, NAZWA\_PODPLANU, SCIEZKA\_DO\_PLIKU, FLAGA\_CZY\_DISABLE, WEKTOR\_ZASOBOW[(PULA, ZASOB, JEDNOSTKI)]***Ad 2.** Plik CSV z kolumnami dotyczącymi relacji , tj: *FROM, TO, TYPE(AT\_SUCCES/AT\_FAIL)***Ad 3.** Plik CSV z kolumnami dotyczącymi podsumowania n ostatnich uruchomień, tj: *NAZWA\_PLANU, TIMESTAMP\_STARTU, TIMESTAMP \_STOPU, CZAS\_PRZETWARZANIA, SUMA(CZAS\_OCZEKIWANIA\_NA\_ZASOBY), ILE\_BYLO\_BLEDOW***Ad 4.** N plików CSV z podsumowaniem uruchomień. Pliki powinny nazywać się : *NAZWA\_PLANU\_DATAGODZINASTARTU*. Pliki powinny zawierac kolumny : *NAZWA\_TASKU, TIMESTAMP\_START, TIMESTAMP\_STOP, CZAS\_TRWANIA, CZAS\_OCZEKIWANIA\_NA\_ZASOBY***Ad 5.** Plik CSV z podsumowaniem błędów z ostatnich n uruchomień planu(może być pusty). Plik powinien zawierać kolumny *NAZWA\_PLANU, TIMESTAMP\_STARTU\_PLANU, TIMESTAMP\_WYSTAPIENIA\_BLEDU, TASK\_KTORY\_ZGLOSIL\_BLAD, KOD\_BLEDU*

W celu pobrania plików logów do parsowania należy przeszukać katalog. Zarówno katalog w którym program będzie szukał logów jak i liczba przeszukiwanych plików powinny być parametryzowane. Katalog będzie parametrem zaszytym w programie tak by można było używać oprogramowania na różnych maszynach i środowiska dev/test/prod bez większej ingerencji w kod.   
Liczba plików do przeszukiwania będzie podawana jako parametr wejściowy razem z nazwa planu.  
  
**Przykład** : użytkownik wywoła program **./PROGRAM.sh PLAN\_TEST.plan 20** a oprogramowanie zwróci:   
-jeden plik z taskami,   
-jeden plik z relacjami,   
-jeden plik z podsumowaniem 20 ostatnich uruchomień,   
-20 plików ze szczegółowym podsumowaniem poszczególnych uruchomień  
-jeden plik z wyświetlonymi bledami z ostatnich 20 uruchomień  
  
**Wskazówka** – oprogramowanie w celu znalezienia plików z N ostatnimi logami wykonania PLAN\_INPUT\_PARAM w katalogu KATALOG\_PARAM wystarczy wywołać:  
**Ls –latr KATALOG\_PARAM/PLAN\_INPUT\_PARAM\* | tail –N**Na potrzeby testu należy utworzyć sobie swój zdalny katalog w którym należy umieścić pliki z logami a ścieżkę do niego   
umieścić w wyżej wspomnianym parametrze