

Inżynieria Oprogramowania 2 - Kalendarz								
Metodologia: Unified Process								
Tydzień	Michał Mierzyński		Michał Padzik		Łukasz Napora		Kamil Żak	
	Plan	Komentarz	Plan	Komentarz	Plan	Komentarz	Plan	Komentarz
1	Zapoznanie się z dokumentacją (3h)		Zapoznanie się z dokumentacją (3h)		Zapoznanie się z dokumentacją (3h)		Zapoznanie się z dokumentacją (3h)	
2	Wstępna implementacja i projektowanie komunikacji - Task Solver (3h)		Stworzenie szkieletu projektu serwera i klas odpowiadających za rejestrację komponentów (3h)		Wstępna implementacja i projektowanie komunikacji - Computational Node (3h)		Wstępna implementacja i projektowanie komunikacji - Client (3h)	
3	Dopracowywanie komunikacji - Task Solver (3h)		Dopracowanie komunikacji - Server (3h)		Dopracowanie komunikacji - Computational Node (3h)		Dopracowanie komunikacji - Client (3h)	
4	Dopracowywanie komunikacji - Task Solver (3h)		Dopracowanie komunikacji - Server (3h)		Dopracowanie komunikacji - Computational Node (3h)		Dopracowanie komunikacji - Client (3h)	
5	Testowanie komunikacji - Task Solver (3h)		Testowanie komunikacji – Server (3h)		Testowanie komunikacji - Computational Node (3h)		Testowanie komunikacji - Client (3h)	
6	Działająca komunikacja pomiędzy innymi komponentami aplikacji		Oddanie etapu komunikacji (1h), szkielet klas odpowiadających za algorytmy rozwiązywania problemów (2h)		Oddanie działającej komunikacji pomiędzy innymi komponentami aplikacji (3h)		Oddanie działającej komunikacji pomiędzy innymi komponentami aplikacji (1h), przygotowanie do implementacji - Client (2h)	
7	Implementacja i integracja algorytmu - Task Solver (3h)		Dopracowanie algorytmów rozwiązywania problemów (3h)		Implementacja i integracja algorytmu - Computational Node (3h)		Implementacja i integracja algorytmu - Client (3h)	
8	Implementacja i integracja algorytmu - Task Solver (3h)		Testowanie działania algorytmów rozwiązywania problemów (3h)		Dopracowanie algorytmu - Computational Node (3h)		Poprawki w algorytmie - Client (3h)	
9	Działający algorytm		Oddanie etapu algorytmów (1h), dopracowanie poprzednich etapów (2h)		Oddanie działającego algorytm (3h)		Oddanie działającego algorytmu (1h), ewentualne poprawki w kodzie (2h)	
10	Testowanie i implementacja pozostałych funkcjonalności - Task Solver (3h)		Dopracowanie poprzednich etapów (3h)		Testowanie i implementacja pozostałych funkcjonalności - Computational Node (3h)		Testowanie i implementacja pozostałych funkcjonalności - Client (3h)	
11	Testowanie - Task Solver (3h)		Testowanie działania całego programu serwera (3h)		Testowanie gotowego Computational Node (3h)		Testowanie całego modułu - Client (3h)	
12	Testowanie - Task Solver (3h)		Testowanie całego projektu (3h)		Testowanie całego projektu (3h)		Testowanie całego projektu (3h)	
13	Oddanie projektu (3h)		Oddanie projektu (3h)		Oddanie projektu (3h)		Oddanie projektu (3h)	
14	Championship (3h)		Championship (3h)		Championship (3h)		Championship (3h)	