**Dokument roboczy projekt ZSI Moduł 3**

**Arkadiusz Kałuża, Adam Kierat**

Przygotowanie dokumentu roboczego zawierającego zapisy dotyczące punktów wyróżnionych we wzorze dokumentu roboczego dostęnego na platformie zdalnej edukacji.

* **A1. Ustanowienie współdzielonego repozytorium**  
  Ustanowienie współdzielonego repozytorium na platformie github.com Link: <https://github.com/arcziko9/ZarzadzanieSystemamiInformatycznymiPOLSL>
* **B1. Ustalenie z prowadzącym tematu projektu.**  
  Sekcja wybrała projekt numer 3.8:  
   a. Przedstaw możliwości wybranego rozwiązania wspomagającego zarządzanie systemami informatycznymi (docker)  
   b. Przedstaw możliwości tworzenia dokumentacji technicznej (podręcznika użytkownika)  
   c. Opracuj film demonstrujący silne i słabe strony zadanego systemu informatycznego będący alternatywą dla Asana
* **B3. Przygotowanie notatki z ASD: 3 pozycje, 3 najważniejsze wyróżniki, linki do witryn zawierających szczegóły.**
  + a. Platforma Docker jest projektem typu open source do automatyzacji wdrażania aplikacji jako przenośnych, samowystarczalnych kontenerów, które można udostępniać w chmurze lub lokalnie. Doker jest również firmą, która promuje i rozwija tę technologię współpracując z dostawcami chmury, systemu Linux i Windows.
    - https://ujeb.se/DfGJR
    - https://ujeb.se/hZb38
    - https://ujeb.se/xX5Nx
  + b. Dokumentacja techniczna powinna być jak instrukcja obsługi – konkretna, czytelna, rozwiewająca wątpliwości, pokazująca użytkownikowi krok po kroku przebieg danej czynności. Podstawowym krokiemi w tworzeniu dokumentacji jest zdefiniowanie kto ma być odbiorcą dokumentacji, żeby odpowiednio dostosować styl pisania jej. Drugim ważnym krokiem jest stworzenie spisu treści w której łatwo będzie można znaleźć odpowiedź na problem, który się natkneliśmy.
    - <https://ujeb.se/GQ8Hg>
    - <https://ujeb.se/jg0pv>
    - <https://ujeb.se/XLuSo>
* **C1.Opracowanie koncepcji realizacji projektu.**
  + Docker to tak naprawdę zbiór funkcjonalnych narzędzi, który pozwala na uruchamianie wielu różnych procesów w tzw. kontenerach, czyli w odizolowanych środowiskach. To coraz popularniejsze rozwiązanie, które często można spotkać w chmurze. Jego główną zaletą jest to, że może zastąpić wirtualną maszyne. Wygoda jego użytkownika wynika z tego, że kontener pozwala na uruchomienie w sposób prostszy niż wirtualna maszyna – nie ma tu żadnego emulowania warstwy sprzętowej w dodatku nie potrzebujemy systemu operacyjnego.
    - **Dystrybucja aplikacji**Docker jest tak sprytny, że pozwala wykorzystać obrazy nie tylko do łatwego tworzenia środowiska deweloperskiego. Można to narzędzie użyć także do dystrybucji aplikacji. O ile w przypadku instalacji na serwerze, administrator lub użytkownik aplikacji musi sam poradzić sobie z tym procesem i ze wszystkimi związanymi z tym aspektami (np. inaczej instaluje się oprogramowanie w Javie i inaczej w Pythonie lub Ruby), o tyle wykorzystanie Dockera ułatwia całą sprawę.  
        
      Programista może łatwo przygotować paczkę dystrybucyjną. Wystarczy, że wykona obraz kontenera z aplikacją, która została ukończona. Już nie administrator czy użytkownik dokonuje instalacji, ale deweloper. Natomiast ci pierwsi dostają kontener, który jest gotowy do uruchomienia i nie muszą zastanawiać się, jaka to technologia, jak ją zainstalować i czy posiadają do tego odpowiednią wiedzę.
    - **Proces zamknięty w kontenerze**Uruchomienie aplikacji w sposób opisany powyżej jest możliwe, ponieważ Docker pozwala na funkcjonowanie samego procesu aplikacji. W ten sposób można inicjować działanie dowolnej liczby procesów w różnych kontenerach, a każdy z nich posiada przydzielony specjalnie dla niego obszar pamięci, własny adres prywatny IP i interfejs oraz ma własny obszar na dysku, gdzie wcześniej zainstalowano obraz systemu operacyjnego i wszystkich bibliotek, które konieczne są do uruchomienia aplikacji. Ważne jest także to, że każdy z tych kontenerów działa zupełnie niezależnie, ale możemy również tworzyć między nimi połączenia sieciowe. Warto jednak pamiętać, że Docker działa pod kontrolą GNU/Linux, natomiast użytkownicy Windows lub Mac muszą skorzystać z menedżera maszyn wirtualnych VirtualBox. Inne rozwiązanie to używanie maszyny wirtualnej z preinstalowanym Dockerem. To oferuje już coraz więcej dostawców cloud computing (np. Azure, Amazon AWS itd.).
    - **Wydajność**Z powodu izolowanych kontenerów Docker pozwala administatorom i programistom dostarać szybko działające, centralnie zarządzane i bezpieczne aplikacje rozproszone. Wydajność ta jest jeszcze większa gdy zastosujemy chmure.
  + Dokumentacja jest potrzebna. Jest **częścią produktu, ma wartość biznesową** - pozwala ograniczyć koszty i pozyskiwać nowych klientów. W zależności od zastosowania, odbiorcy czy skali projektu, może ona **przybierać różne formy**. Zazwyczaj pisanie dokumentacji nie wymaga tworzenia tysięcy dokumentów i skomplikowanego języka - treści powinny być proste i zrozumiałe dla jak największej grupy odbiorców. W idealnej sytuacji, dokumentacja może być częścią procesu tworzenia oprogramowania, co pozwala ograniczyć czas potrzebny na jej przygotowanie i publikację.  
    powinniśmy zasięgnąć trochę wiedzy na temat wypracowanych **metod skutecznej komunikacji**. Dobrym przykładem będzie **plain language** - czyli zestaw wytycznych, które mówią wprost, co to znaczy, że tekst jest prosty. Celem jest tworzenie dokumentów w taki sposób, aby były zrozumiałe dla możliwie szerokiego spektrum odbiorców.
    - **Przykładowe narzędzia do tworzenia dokumentacji technicznej:**
      * **Natural Docs (** [**https://www.naturaldocs.org/**](https://www.naturaldocs.org/) **)**pozwala dokumentować kod napisany w jednym z 21 języków programowania, a ponadto można go łatwo rozszerzyć o więcej, więc bez względu na to, czego używasz, może również. A jeśli twój projekt używa wielu języków, nie ma problemu! Wszystko to będzie zawarte w tym samym zestawie dokumentacji.
      * **DoxyGen (** [**http://www.doxygen.nl/**](http://www.doxygen.nl/) **)**Ndo generowania dokumentacji ze źródeł C ++ z adnotacjami, ale obsługuje także inne popularne języki programowania, takie jak C, Objective-C, C #, PHP, Java, Python, IDL (smaki Corba, Microsoft i UNO / OpenOffice) ),