**SLAJD 1:**

Praca dyplomowa inżynierska pod tytułem „Opracowanie projektu komputerowego systemu zarządzania przepływem informacji w środowisku rozproszonym sortowni koksu”.

**SLAJD 2:**

Motywacją do podjęcia się tematu pracy, była potrzeba stworzenia na rzecz sortowni koksu systemu komputerowego, który będzie stabilny a jednocześnie elastyczny. Pierwotny system sortowni posiadał odrębne bazy danych w obrębie przedsiębiorstwa, takie rozwiązanie stwarzało ryzyko dublowania informacji oraz powstawanie zbędnych błędów.

Ten problem otworzył możliwość wykorzystania metodyki SAP ERP, która wprowadza zarządzanie przedsiębiorstwem z poziomu jednej bazy danych, która skupia w sobie zarządzanie wszystkimi dziedzinami danego przedsiębiorstwa. Działanie tej metodyki polega na zbieraniu, analizie i wykorzystaniu zdobytych informacji do podejmowania optymalnych decyzji, które wpływają na szybszy rozwój przedsiębiorstwa. Korzystanie z jednej bazy danych daje możliwości wglądu w dane wszystkich działów niezależnie od miejsca, co ma duże znaczenie dla płynności pracy.

Ponadto w pierwotnym systemie proces poboru próbki wykonywany był przez pracownika sortowni. W odpowiednim momencie podchodził i pobierał określoną część produktu, która następnie została kierowana na badania laboratoryjne. To generowało dużo większy nakład czasu, stąd pomysł, by proces ten zastąpić zautomatyzowanym ramieniem, obsługiwanym przez pracownika. W losowej chwili miał on uruchomić ramię, by kilkakrotnie pobrało składowe próbki z różnych wagonów, które finalnie składają się na próbkę wysyłaną na badania.

Dodatkowo motywacyjne było to, że nie istnieją udokumentowane przypadki podobnych systemów w tej gałęzi przemysłu. Po wielokrotnych poszukiwaniach nie zdołałam znaleźć informacji o podobnych istniejących systemach.

**SLAJD 3:**

Sam proces sortowania koksu można zamknąć w czterech głównych punktach. (1) Koks zostaje na początku wypalony w specjalnych piecach koksowniczych. (2) Następnie kopy wypalonego koksu zostają odpowiednio rozdzielane do wagonów, ważny jest również wybór sposobu zasypu, ponieważ w zależności od wielkości ziaren koksu istnieją różne metody zasypywania wagonów. (3) Podczas wypełniania wagonów zostają pobierane próbki, które następnie są wysyłane na (4) badania laboratoryjne, które mają za zadanie określić ich szczegółowy skład chemiczny, dokładną wielkość oraz szczegółowe właściwości.

Na podstawie informacji o funkcjonowaniu sortowni koksu przeszłam do projektu systemu. Na slajdzie znajduje się rysunek przedstawiający diagram przypadków użycia powstałego na rzecz sortowni koksu. Opisuje on funkcjonalność systemu i jego otoczenia. Aktorem w tym przypadku jest użytkownik systemu, czyli pracownik sortowni. Może on logować się oraz wylogowywać z systemu, wybierać zamówienie do realizacji, następnie na tej podstawie wybierać ilość, typ wagonów oraz sposób zasypu. Ma również możliwość odczytania etykiety wagonowej, która stanowi wejście do systemu, każdy wagon posiada indywidualną etykietę, która zawiera informacje dotyczące wagonu. Etykieta odczytywana jest za pomocą kamery umieszczonej na spodziewanej wysokości, a w momencie nieprawidłowego odczytu użytkownik może wprowadzić zmiany ręcznie. Decyduje on również o momencie rozpoczęcia zasypywania, kontroluje cały proces. Później wybiera z ilu wagonów mają zostać pobrane składowe próbki oraz decyduje o momencie wysłania próbki na badania. Ostatecznie może przygotować raport oraz zamknąć zamówienie.

**SLAJD 4:**

Na slajdzie znajduje się rysunek przedstawiający diagram sekwencji. Służy on do prezentowania interakcji pomiędzy obiektami wraz z uwzględnieniem w czasie komunikatów przesyłanych pomiędzy nimi. Diagram stworzony na rzecz projektu zawiera siedem obiektów. Reprezentują one, od lewej: wewnętrzną bazę danych systemu, dostarczaną z zewnątrz bazę danych zamówień, również dostarczaną z zewnątrz bazę danych dotyczących wagonów, użytkownika systemu, etykietę wagonową, zasypywanie oraz laboratorium.

Na diagramie zostały wyszczególnione wszystkie użycia systemu w przypadku pomyślnej realizacji zamówienia, od momentu pobrania zamówienia z zewnętrznej bazy danych przez użytkownika, po zamknięcie zamówienia oraz archiwizację. Diagram ten jest uzupełnieniem diagramu przypadków użycia, zasadniczym działaniem diagramu sekwencji jest modelownie zachowania systemu w kontekście scenariuszy przypadków użycia.

**SLAJD 5:**

Wzorując się na utworzonych diagramach przystąpiłam do wstępnej implementacji projektu. Powstała aplikacja jest aplikacją REST-ową. Wybrałam taką formę implementacji ze względu na to, iż REST ignoruje szczegóły implementacji komponentów i składni protokołu, by bardziej skupić się na roli komponentów, ich ograniczeniach, interpretacji istotnych elementów danych. Wiodącą zaletą tej metody są również zasoby (ang. *resources*), gdzie każdy posiada unikalny adres. Dzięki adresom oraz metodom protokołu HTTP możliwe jest sterowanie aplikacją. poniższy zrzut ekranu na ciemnym tle przedstawia właśnie wywołanie metod aplikacji, są to:

– GET, czyli pobranie (w tym przypadku zamówienia),

– POST, czyli żądanie dodania nowego elementu systemu (w tym przypadku próbki koksu),

– PATCH, aktualizacja elementu próbki (w tym przypadku pola reprezentującego datę poboru próbki) PATCH, a nie PUT, bo PATCH pozwala na aktualizację jednego pola, a nie całego elementu,

– DELETE, czyli żądanie wykasowania z systemu (w tym przypadku wcześniej utworzonej próbki).

Jak widać serwer zwraca odpowiedzi 200, świadczące o tym, że aplikacja działa poprawnie, 201 w momencie utworzenia oraz 404 (nie znaleziono) w przypadku wywoływania metody get dla usuniętej wcześniej próbki.

Po prawej stronie slajdu znajduje się natomiast zrzut ekranu reprezentujący strukturę aplikacji systemu, główne pliki aplikacji zostały zaznaczone zielonym prostokątem i są to pliki w folderach routes oraz models, są to implementacje odpowiednio obsługi wywołań metod HTTP oraz modele opisujące pola poszczególnych elementów systemów.

**SLAJD 6:**

Testy aplikacji odbywały się za pośrednictwem aplikacji Postman, gdzie po wprowadzeniu odpowiedniego adresu oraz wyboru metody (zaznaczone strzałką) można było monitorować działanie aplikacji. Zrzuty ekranu przedstawiają odpowiednio zrzut z aplikacji Postman, a dokładnie sprawdzenie metody pobierania utworzonych próbek koksu oraz ten sam przypadek dla monitorującej działania nierelacyjnej bazy danych MongoDB. Połączenie aplikacji z MongoDB automatyzuje działania administracyjne, ponieważ wiele zabezpieczeń oraz operacji są wbudowane. Tworząc aplikację dzięki temu rozwiązaniu można bardziej skupić się na budowaniu jej, a mniej na zarządzaniu bazą danych.

Wybrałam próbkę w celu przedstawienia działania aplikacji, ze względu na to, iż na ten moment stanowi ona serce całego systemu.

**SLAJD 7:**

Podsumowując. Należało stworzyć system, który będzie uniwersalny, jednocześnie przejrzysty dla klientów jak i pracowników sortowni koksu. Uniwersalność jest szczególnie ważnym aspektem w tym przypadku, ze względu na ciągły rozwój projektu. Dlatego też projekt systemu powstał w oparciu o wzorzec architektoniczny Model View Controller, który dzięki swoim niezależnym warstwom pozwala na przebudowę wybranego elementu bez konieczności przebudowy całego systemu. W skład projektu wchodzą diagramy UML przedstawiające funkcjonowanie systemu, elastyczny model danych MongoDB oznacza również, że schemat bazy danych może ewoluować wraz z wymaganiami projektu.

Dalsze prace nad tematem powinny koncentrować się na doskonaleniu implementacji, poprawie autoryzacji użytkowników, zwiększeniu poziomu zabezpieczeń oraz stworzeniu interfejsu graficznego.

**SLAJD 8:**

Dziękuję za uwagę.