

# Technologie i narzędzia użyte w projekcie

Nasza praca nad projektem zarządzania aplikacją laboratorium analitycznego opierała się na wykorzystaniu różnorodnych technologii i rozwiązań, które umożliwiły nam efektywne tworzenie kompleksowego systemu. Użyliśmy technologii i narzędzi takich jak:

1. Visual Studio i C#: Do pisania kodu aplikacji desktopowej użyliśmy środowiska programistycznego Visual Studio oraz języka programowania C#. Visual Studio jest jednym z najpopularniejszych narzędzi programistycznych, które oferuje zaawansowane funkcje i narzędzia do tworzenia aplikacji. C# to język programowania zorientowany obiektowo, który jest często stosowany w aplikacjach desktopowych dla systemu Windows. Wybór Visual Studio i C# pozwolił nam na szybkie tworzenie aplikacji o solidnej architekturze, wydajności i skalowalności.
2. Windows Forms: Aplikację desktopową stworzyliśmy przy użyciu technologii Windows Forms, która jest częścią platformy .NET. Zapewnia ona framework i zestaw kontrolki, które umożliwiają tworzenie interfejsu użytkownika w aplikacjach desktopowych dla systemu Windows. Używając Windows Forms, mogliśmy stworzyć intuicyjny i responsywny interfejs użytkownika, umożliwiający łatwą nawigację i wygodne korzystanie z funkcji systemu zarządzania laboratorium analitycznego.
3. SQLite to lekka, bezserwerowa baza danych, która przechowuje dane w lokalnym pliku. W naszej aplikacji SQLite może być został wykorzystany jako lokalna baza danych do przechowywania danych klientów, informacji o próbkach, wynikach, rodzajach badań i zleconych zamówieniach. SQLite oferuje prosty interfejs, który można łatwo zintegrować z aplikacją. Zapytania SQL można wykonywać bezpośrednio do bazy danych SQLite, co umożliwia zarządzanie danymi i wykonywanie operacji, takich jak wstawianie, aktualizowanie, usuwanie i pobieranie danych.
4. GitHub to platforma hostingowa dla repozytoriów Git, która umożliwia zespołową pracę nad projektem, śledzenie zmian, zarządzanie wersjami kodu i udostępnianie kodu open source. Dzięki temu mieliśmy kontrolę nad historią i rozwojem naszego kodu, umożliwiając równoczesną pracę wielu członków zespołu nad projektem.
5. Jira: Aby skutecznie zarządzać projektem, użyliśmy oprogramowania Jira. Jira to popularne narzędzie do śledzenia zadań, zarządzania projektem i raportowania postępów. Dzięki Jirze mogliśmy tworzyć i przypisywać zadania, śledzić ich postęp, zarządzać priorytetami, planować sprinty, tworzyć raporty i wiele więcej. Jira zapewniła nam przejrzystość i kontrolę nad całym projektem, umożliwiając efektywne zarządzanie zespołem i harmonogramem prac.
6. StarUML: Do tworzenia diagramów użyliśmy narzędzia StarUML. Jest to profesjonalne oprogramowanie do modelowania, które umożliwia tworzenie różnych typów diagramów UML, takich jak diagramy klas, diagramy sekwencji, diagramy stanów, diagramy przypadków użycia itp. StarUML pozwolił nam lepiej zrozumieć strukturę naszego projektu, zidentyfikować zależności między komponentami, a także efektywniej komunikować się w zespole programistycznym. Tworzenie diagramów UML pomogło nam również w dokumentacji i analizie projektu.

7. Podczas pisania kodu, staraliśmy się stosować do zasad SOLID oraz KISS. Te zasady pomogły nam tworzyć modułowy, skalowalny i łatwo zrozumiały kod.

Dzięki wykorzystaniu tych różnorodnych technologii i narzędzi, byliśmy w stanie stworzyć prosty, a zarazem zaawansowany system zarządzania laboratorium analitycznego. Integracja Visual Studio, C#, Windows Forms, SQLite, GitHuba, Jiry i StarUML pozwoliła nam skutecznie planować i monitorować postępy projektu oraz zapewnić intuicyjny interfejs dla użytkowników, natomiast przestrzeganie zasad SOLID oraz KISS dało nam możliwość łatwej współpracy nad jednym kodem. To zestawienie technologii pozwoliło nam na efektywne i niezawodne tworzenie aplikacji, spełniając potrzeby naszego projektu.