**SPRING BOOT:**

Jest to framework oparty na Java, który jest rozszerzeniem najpopularniejszego frameworka Java – Springa. Spring boot wykorzystuje najlepsze praktyki Springa, a dodatkowo zapewnia konfigurację oraz kontener aplikacji. Dzięki konfiguracji programista może pisać bardziej zwięzły kod, bez generowania często powtarzalnych skryptów konfiguracyjnych. Natomiast dzięki kontenerowi aplikacji, można uruchomić aplikację jednym kliknięciem. Pozwala na szybkie tworzenie aplikacji internetowych i mikrousług.

Wybraliśmy Spring boot ze względu na:

* małą lub nawet brak potrzeby konfiguracji(w porównaniu do Springa).
* Łatwą komunikację z bazą danych - Hibernate
* Łatwość dodawania potrzebnych bibliotek – Maven
* Szybkość pisania kodu – adnotacje
* Szeroką gamę bibliotek Javowych ułatwiających pracę – np. Lombok

**HTML5:**

Jest to język znaczników, służący do tworzenia dokumentów hipertekstowych oraz opisywania struktury informacji zawartych wewnątrz strony. Za pomocą HTML nadajemy określone znaczenie semantyczne stronie dla elementów takich jak nagłówki, nawigacja, hiperłącza, formularze itp. Mimo iż jest nazywany językiem, nie należy do języków programowania, gdyż w jego składni nie przewidziano wyrażeń warunkowych, obliczeniowych czy iteracyjnych. Obecnie jest najpopularniejszą technologia odpowiedzialna za opis treści znajdującej się na stronach internetowych WWW.

Wybraliśmy HTML5 ze względu na:

* Standaryzacje (strona wygląda tak samo na każdej przeglądarce)
* Semantykę kodu(wykorzystanie znaczników zamiast samych div’ów pozwala łatwiej określić ich rolę)

**CSS:**

Są to tzw. kaskadowe arkusze stylów. Służą one do opisania wyglądu elementów witryny, zdefiniowanych wcześniej w HTML. Jest to zbiór reguł opisujących, jak dany element powinien wyglądać. Odseparowuje on strukturę dokumentu od formy jego prezentacji, przez co kod jest bardziej przejrzysty i łatwiej jest wprowadzić zmiany w strukturze dokumentu.

Wybraliśmy CSS ze względu na:

* Krótszy i bardziej przejrzysty kod
* Łatwiejszą modyfikację kodu
* Możliwość stosowania zmiennych

**SASS**

Jest to preprocesor CSS. Wykorzystuję się go do tworzenia wydajniejszych i bardziej przejrzystych arkuszy stylów. Składnia jest podobna do tej z CSS, jednak różni się kilkoma funkcjami które nie są jeszcze obsługiwane w CSS. Jedną z takich funkcji są mixiny, dzięki którym nie musimy powtarzać wiele razy tego samego wycinka kodu. Kolejną funkcją są zagnieżdżenia selektorów CSS w taki sam sposób jak robi się to w HTML.

Wybraliśmy SASS ze względu na:

* Możliwość zagnieżdżania kodu
* Możliwość tworzenia mixinów

**JAVASCRIPT**

Jest to dynamicznie typowany język wysokiego poziomu, najczęściej wykorzystywany do tworzenia stron WWW / aplikacji webowych. Zapewnia on interaktywność stron oraz obsługę zdarzeń, walidację formularzy czy budowanie elementów nawigacyjnych. Jest on prostym językiem do rozpoczęcia programowania. JavaScript to język skryptowy, co oznacza, że kod jest interpretowany w czasie rzeczywistym przez przeglądarkę lub środowisko uruchomieniowe. Javascript jest jednym z najpopularniejszych języków programowania na świecie i jest obsługiwany przez większość przeglądarek internetowych.

Wybraliśmy JavaScript ze względu na:

* Prostą i intuicyjną składnie
* Popularność języka (duża baza wiedzy)
* Uniwersalność (pozwala na wykonanie niemalże dowolnego działania na stronie)

**FIGMA**

Jest to aplikacja do projektowania stron internetowych(np. interfejsu użytkownika). Jest darmowym oprogramowaniem graficznym do profesjonalnych zastosowań. Łączy w sobie najlepsze narzędzia i rozwiązania znane z komercyjnych pakietów graficznych. Umożliwia tworzenie i edycję zarówno rastrowej, jak i wektorowej grafiki. Ma przejrzysty i intuicyjny interfejs, więc nie ma problemu z użytkowaniem tej aplikacji.

Wybraliśmy Figmę ze względu na:

* Łatwość współpracy
* Brak konieczności instalowania specjalnego oprogramowania
* Możliwość używania aplikacji w dowolnym miejscu, na dowolnym urządzeniu

**POSTMAN**

Postman oferuje kompleksowy zestaw narzędzi do projektowania, testowania i dokumentacji. Dzięki niemu można usprawnić pracę z interfejsem API wykorzystując zaawansowane funkcje wyszukiwania czy powiadomień. Umożliwia on wykonywanie zapytań i przechowywanie ich w historii. Przy użyciu Postmana można łatwo tworzyć testować i udostępniać żądania http oraz przetwarzać odpowiedzi. Przyspiesza on proces tworzenia oprogramowania, a także ułatwia jego utrzymanie i rozwój.

Wybraliśmy Postmana ze względu na:

* Możliwość tworzenia i udostępniania dokumentacji interfejsów API
* Przyjazny interfejs użytkownika
* Możliwość monitorowania zmian w API
* Bezpłatną wersję
* Dużą bazę wiedzy