



# **WYBÓR NARZĘDZI I ICH UZASADNIENIE**

Amelia Hajkowska

Alicja Szulc

# JĘZYK PROGRAMOWANIA

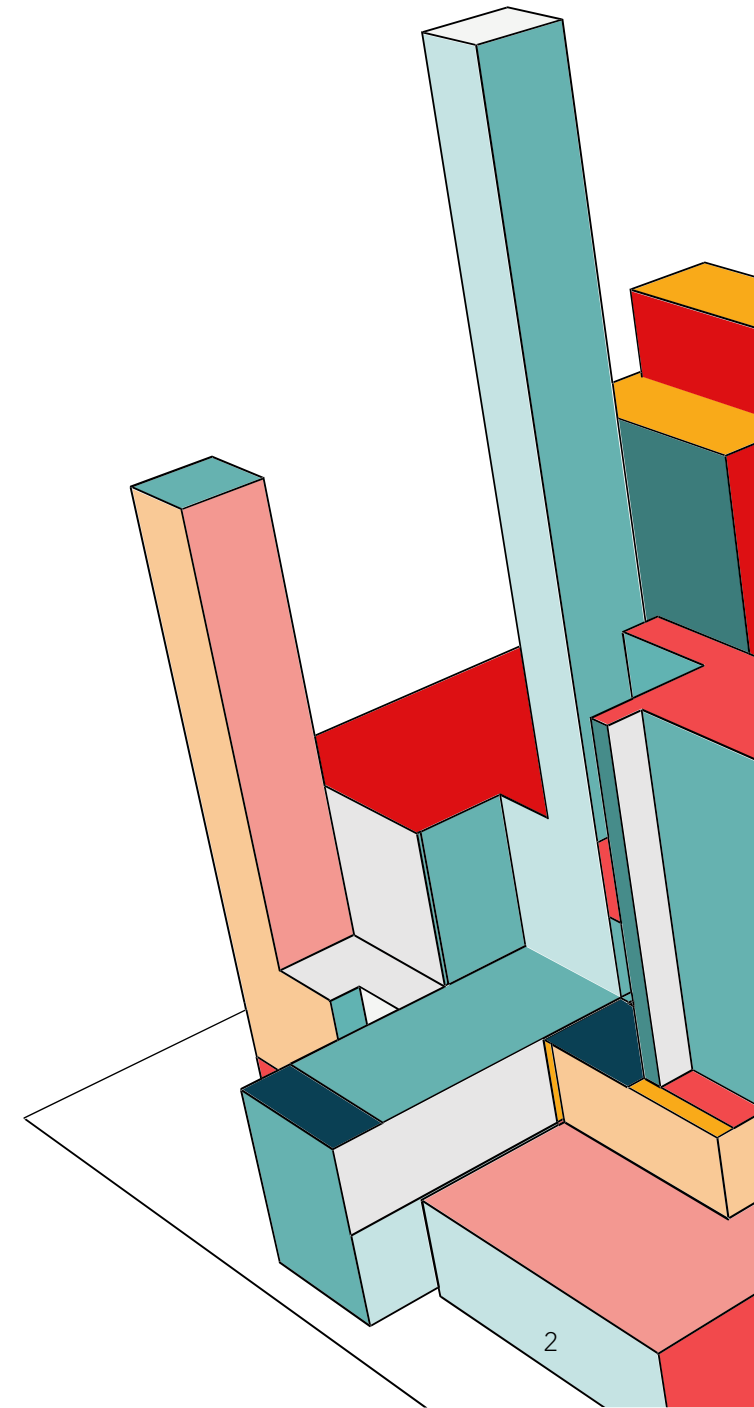
Kotlin

---

**Przeznaczenie:** Tworzenie aplikacji mobilnych na Androida.

**Uzasadnienie:**

- **Oficjalne wsparcie Google dla Androida**
- **Nowoczesna i czytelna składnia** - W porównaniu do Javy, Kotlin oferuje bardziej zwięzłą składnię. Kod w Kotlinie jest łatwiejszy do zrozumienia, co poprawia jego czytelność oraz ułatwia pracę programistom. Dzięki temu praca nad kodem jest szybsza, a liczba błędów zmniejsza się.
- **Wsparcie dla zaawansowanych funkcji asynchronicznych** - zapewnia wsparcie dla *coroutines*, które ułatwiają obsługę operacji asynchronicznych bez komplikowania kodu. W naszym projekcie umożliwi to płynne pobieranie rekomendacji mebli oraz przetwarzanie zdjęć przesyłanych przez użytkowników, bez blokowania głównego wątku interfejsu użytkownika.
- **Ścisła integracja z Android Studio** - Kotlin świetnie współpracuje z Android Studio - oficjalnym środowiskiem programistycznym dla aplikacji Android. Oznacza to dostęp do pełnego zestawu narzędzi deweloperskich, które usprawniają proces tworzenia i testowania aplikacji oraz pomagają w rozwiązywaniu problemów.





# ŚRODOWISKO IDE

## Android Studio

---

**Przeznaczenie:** Środowisko pracy do tworzenia, testowania i wdrażania aplikacji.

**Uzasadnienie:**

- Oficjalne wsparcie Google
- Zaawansowane narzędzia do zarządzania interfejsem użytkownika - Nasza aplikacja wymaga intuicyjnego i estetycznego interfejsu,
- Emulatory i testowanie na różnych wersjach Androida - Android Studio oferuje rozbudowany zestaw emulatorów, co pozwala na symulowanie działania aplikacji na różnych wersjach Androida i różnych urządzeniach.
- Łatwa integracja z bazą danych i innymi narzędziami - Android Studio wspiera integrację z narzędziami do zarządzania bazami danych (np. Room lub Firebase), co ułatwi nam wdrożenie funkcji, takich jak przechowywanie zdjęć i obsługa rekomendacji.

# FRAMEWORKI

## Android Jetpack

---

**Opis:** zestaw bibliotek i narzędzi stworzony przez Google, który wspiera tworzenie aplikacji na Androida. Zawiera komponenty takie jak LiveData, ViewModel, Room, Navigation i inne.

### Zastosowanie:

- **Room** - lokalne przechowywanie danych - zachowanie ulubionych stylów architektonicznych użytkownika nawet bez dostępu do internetu
- **Navigation Component** - usprawni tworzenie i zarządzanie nawigacją między ekranami aplikacji, co ułatwi poruszanie się po aplikacji, np. między ekranem wyboru stylu a rekomendacjami architektów.

## Glide

---

**Opis:** biblioteka do ładowania i buforowania obrazów w aplikacjach androidowych. Umożliwia szybkie i efektywne pobieranie oraz wyświetlanie obrazów z internetu

**Zastosowanie:** wyświetlanie obrazów w aplikacji. Aplikacja będzie płynnie ładować obrazy bez obciążania pamięci co poprawi wrażenia użytkowników



## Retrofit

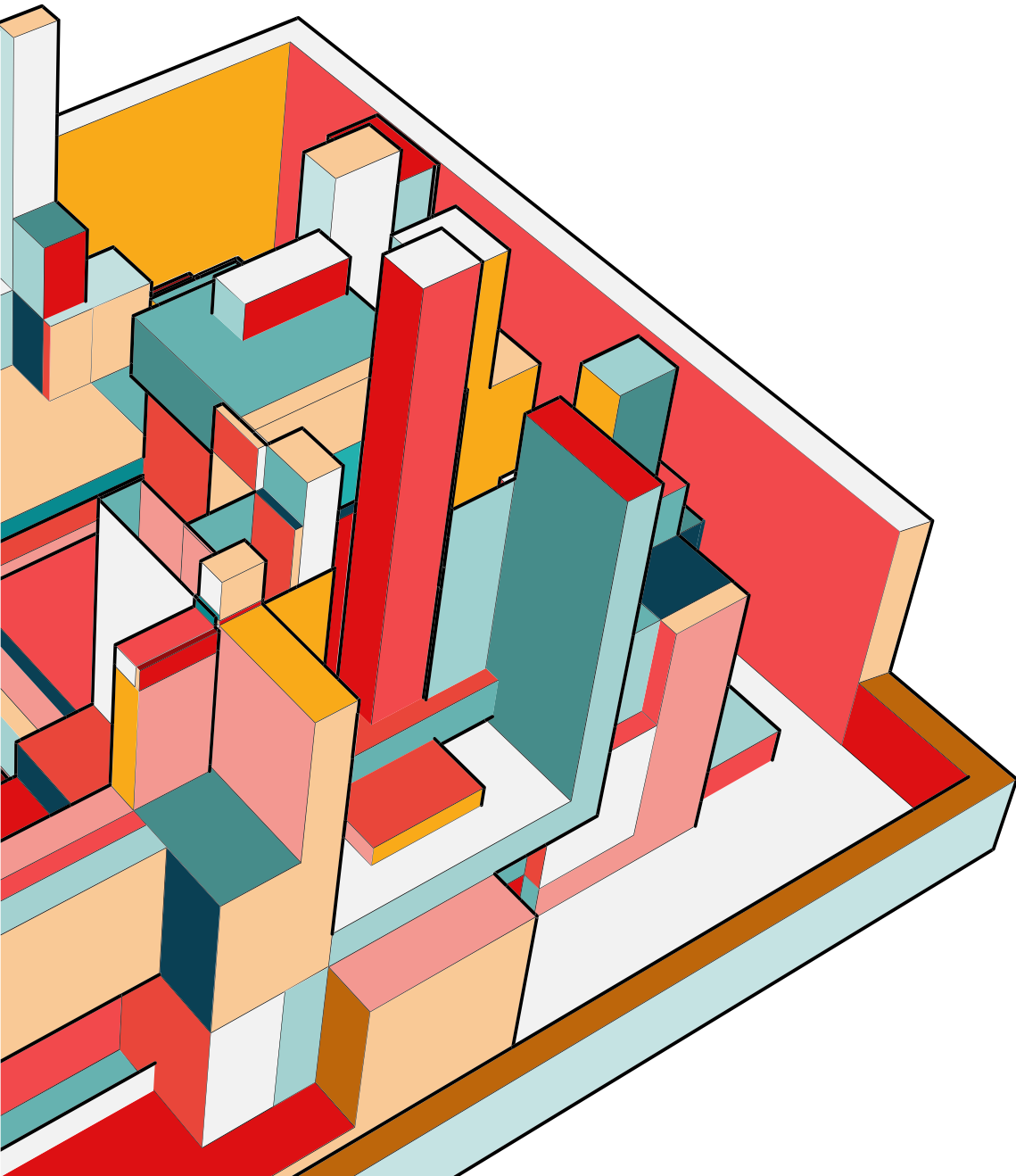
**Opis:** biblioteka do komunikacji z interfejsami API. Umożliwia łatwe wysyłanie żądań HTTP, przetwarzanie odpowiedzi oraz obsługę danych JSON.

**Zastosowanie:** do analizy zdjęć mebli przesłanych przez użytkowników (wysyłanie zdjęć do usługi rozpoznawania obrazu)

## TensorFlow Lite

**Opis:** TensorFlow Lite to wersja frameworku TensorFlow przeznaczona do implementacji modeli uczenia maszynowego na urządzeniach mobilnych.

**Zastosowanie:** Rozpoznawanie przedmiotów na zdjęciach. TensorFlow Lite umożliwi uruchomienie modeli AI na urządzeniu mobilnym



# SYSTEMY KONTROLI WERSJI

## Git

---

**Przeznaczenie:** Każdy członek zespołu może pracować nad kodem aplikacji w sposób bezpieczny i uporządkowany.

**Uzasadnienie:**

- Śledzenie zmian w kodzie - monitorowanie na bieżąco rozwoju aplikacji
- Praca zespołowa - umożliwienie pracy nad kodem kilku osobom jednocześnie



# SYSTEMY CI/CD

## GitHub Actions

---

**Przeznaczenie:** Automatyzacja testów i budowy aplikacji.

**Uzasadnienie:** Pozwoli to na automatyczne budowanie i testowanie aplikacji przy każdej zmianie w kodzie, co zmniejszy ryzyko błędów.

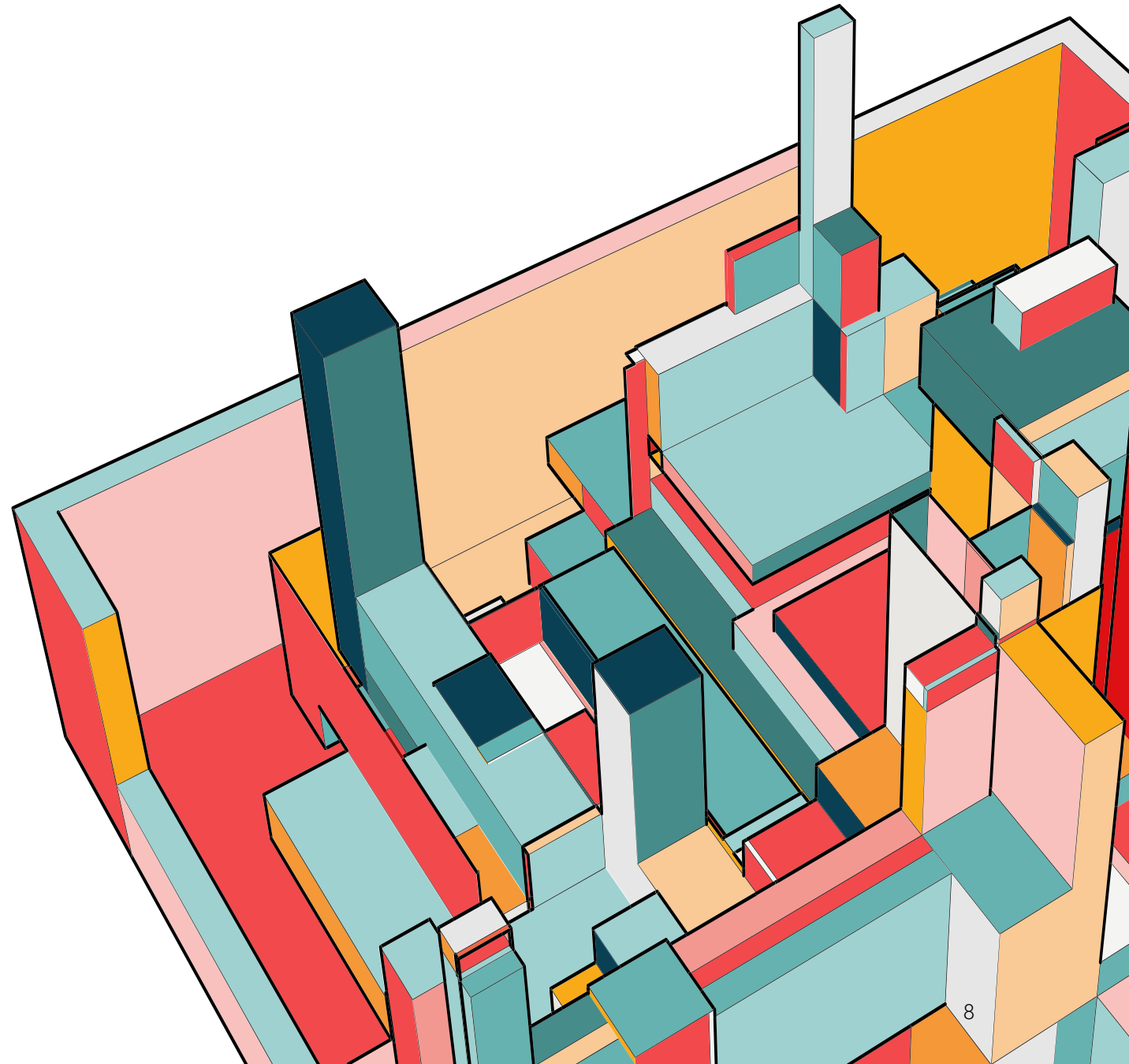
# NARZĘDZIA DO PROJEKTOWANIA

## Figma

---

**Przeznaczenie:** Używana do zaprojektowania interfejsu aplikacji

**Uzasadnienie:** Jest to popularne i łatwe w obsłudze narzędzie do projektowania, które ułatwi współpracę zespołową przy tworzeniu intuicyjnego i estetycznego interfejsu.





# INNE PLATFORMY DO WSPÓŁPRACY



**Jira**

**Cel:** Zarządzanie zadaniami i śledzenie postępów w projekcie.

**Uzasadnienie:** Umożliwiają organizowanie pracy zespołu, planowanie sprintów i monitorowanie postępu prac nad aplikacją.



**GitHub**

**Cel:** Przechowywanie kodu

**Uzasadnienie:** Umożliwia wspólne rozwijanie projektu, wersjonowanie kodu i automatyzację procesów, takich jak testowanie czy publikacja aplikacji. Umożliwia także korzystanie z funkcji „pull request”, która pomaga przeprowadzać przeglądy kodu.



**Discord**

**Cel:** Komunikacja zespołowa (tekstowa i głosowa) oraz organizacja projektowa

**Uzasadnienie:** Omawianie kwestii technicznych oraz wizualnych w prosty sposób.

**DZIĘKUJEMY  
ZA UWAGĘ**

