

# Functional and Non-Functional Requirements

## UG Helper

### Functional requirements:

- Mapa może wykorzystywać GPS do śledzenia lokalizacji użytkownika i oferować nawigację w czasie rzeczywistym do wybranych punktów. Zdjęcie satelitarne kampusu, na którym widać pinezkę ze zdjęciem wydziału, po kliknięciu w nią przenosi nas do planu budynku, na którym widoczne są piętra i sale. Zdjęcia satelitarne można pobrać z Google Maps a zdjęcia wydziałów/sal zrobić samemu.
- Aplikacja musi wyświetlać mapę kampusu z oznaczeniami sal wykładowych, bibliotek, stołówek, miejsc do nauki, biur administracyjnych i obiektów rekreacyjnych. System powinien zawierać opcje wyszukiwania konkretnego miejsca na mapie (np. „Biblioteka”) lub kategorii (np. “Jedzenie”).
- Aplikacja umożliwia przeglądanie dostępnych kursów i kół naukowych wraz z ich opisami oraz daje możliwość zapisania się na nie. By zapewnić dostęp studentom do wszystkich zajęć dodatkowych, umożliwimy prowadzącym dodanie informacji na ich temat do systemu.

- Inteligentne rekomendacje: Aplikacja powinna automatycznie sugerować kursy i koła naukowe na podstawie danych użytkownika, takich jak kierunek studiów lub zainteresowania (można dodać je na swoim profilu).
- Aplikacja powinna automatycznie pobierać plan zajęć użytkownika z systemu uczelni i wyświetlać go w czytelny sposób, organizując zajęcia według dni tygodnia i godzin. Użytkownik może ręcznie dodać własne wydarzenia i edytować istniejące elementy, aby dostosować plan do swoich potrzeb.
- Planer powinien umożliwiać ustawienie przypomnień dla nadchodzących zajęć, terminów oddania prac lub testów. Użytkownik może dostosować czas otrzymywania przypomnień (np. 10 minut przed zajęciami, dzień przed terminem oddania projektu). Powiadomienia będą wyświetlane na ekranie głównym aplikacji lub w formie powiadomień push.

## Non-Functional requirements:

### 1. Bezpieczeństwo danych użytkownika

- **Opis:** Aplikacja musi chronić dane osobowe studentów, takie jak harmonogram zajęć i inne prywatne informacje. Wszystkie dane przesyłane między aplikacją a serwerem powinny być szyfrowane za pomocą protokołu HTTPS (SSL/TLS).
- **Cel:** Zapewnienie prywatności i bezpieczeństwa danych użytkownika zgodnie z wymaganiami RODO oraz politykami ochrony danych Uniwersytetu Gdańskiego.
- **Dodatkowe wymagania:** Dostęp do konta użytkownika powinien być chroniony uwierzytelnianiem. Dodatkowo, sesje użytkownika powinny wygasać automatycznie po określonym czasie nieaktywności.

### 2. Dostępność i niezawodność działania

- **Opis:** Aplikacja powinna działać niezawodnie i być dostępna przez większość czasu, z minimalną liczbą przerw i błędów. W razie problemów technicznych lub dużego obciążenia systemu, aplikacja powinna automatycznie próbować przywrócić swoje działanie.
- **Cel:** Zapewnienie studentom ciągłego dostępu do najważniejszych funkcji, szczególnie wtedy, gdy wielu użytkowników korzysta z aplikacji jednocześnie (np. na początku semestru). Przeprowadzenie testów obciążeniowych w celu sprawdzenia czy aplikacja jest w stanie obsłużyć większą liczbę osób bez opóźnień systemów.

- **Dodatkowe wymagania:** Aplikacja powinna informować o planowanych przerwach technicznych i umożliwiać przeglądanie ostatnio pobranych danych nawet bez połączenia z internetem.

### 3. Wydajność i responsywność interfejsu użytkownika

- **Opis:** Aplikacja powinna być responsywna, zapewniając czas ładowania poniżej 2 sekund niezależnie od systemu czy specyfikacji urządzenia. Wydajność powinna być testowana i zoptymalizowana, aby zminimalizować zużycie baterii i zasobów pamięci urządzenia.
- **Cel:** Zwiększenie komfortu użytkowania i wydajności działania aplikacji, szczególnie na urządzeniach mobilnych o niższych parametrach technicznych.
- **Dodatkowe wymagania:** Warto także uwzględnić różne rozdzielczości ekranów, aby zapewnić spójność wizualną i funkcjonalność.