

# Specyfikacja

## Aplikacja Wadokei

### 1. Wprowadzenie

#### 1.1 Cel dokumentu

Celem niniejszej specyfikacji jest zebranie, usystematyzowanie i uszczegółowienie wszystkich wymagań funkcjonalnych i нефunkcjonalnych aplikacji Wadokei. Dokument ma stanowić podstawę do projektowania, implementacji, testowania oraz ewaluacji aplikacji.

#### 1.2 Zakres dokumentu

Specyfikacja obejmuje opis funkcjonalności aplikacji, wymagania techniczne, нефunkcjonalne, harmonogram i zarządzanie zmianami.

### 2. Cel systemu

Aplikacja Wadokei to zegar cyfrowy inspirowany tradycyjnym japońskim systemem pomiaru czasu, w którym długość godzin dnia i nocy zmienia się w zależności od pory roku. Jej celem jest edukacja kulturowa, wizualizacja działania systemu Wadokei oraz umożliwienie jego obserwacji we współczesnym kontekście.

### 3. Interesariusze

Rola	Odpowiedzialność
szeф projektu	Zarządzanie priorytetami, kontakt z prowadzącym
zespół analityków	Dokumentacja wymagań, przygotowanie specyfikacji
zespół developerów	Implementacja backendu (Flask/Python) i frontendu (HTML)
zespół testerów	Testy funkcjonalne, wydajnościowe, użytecznościowe

### 4. Funkcjonalności

#### 4.1 Wyświetlanie czasu

- Dynamiczna animacja wskazówek godzinowej i minutowej, dostosowana do zmiennej długości dnia i nocy.
- Czas podzielony na 12 jednostek (6 dziennych, 6 nocnych) o zmiennej długości.

- Format godzin: znaki Kanji jako oznaczenia godzin.
- Stylizowana tarcza typowa dla Wadokei
- Wyświetlanie bieżącego czasu w predefiniowanych miastach (standardowy format HH:MM:SS) pod tarczą zegara
- Licznik pokazujący, ile standardowych sekund mieści się w jednej minucie systemu Wadokei

#### **4.2 Lokalizacja i obliczenia**

- Automatyczne ustalanie lokalizacji (GPS)
- Predefiniowane miasta: Nowy Jork, Londyn, Tokio, Sydney, Kioto, Nara
- Możliwość wprowadzania nazwy lub współrzędnych własnej lokalizacji
- Zapisywanie i zarządzanie wieloma lokalizacjami z opcją szybkiego przetaczania między nimi.
- Nie uwzględniamy wysokości względem poziomu morza.
- Obliczanie czasu trwania dnia i nocy na podstawie danych astronomicznych.

#### **4.3 Tryb offline**

- Aplikacja działa w trybie offline po pierwszym wczytaniu danych.

### **5. Wymagania Techniczne**

#### **5.1 Oprogramowanie**

- Języki programowania: Python, HTML i CSS

#### **5.2 Sprzęt**

- Kompatybilność z nowoczesnymi przeglądarkami (Chrome, Firefox, Safari, Edge).
- Działanie na systemach: Windows, macOS, Linux, Android, iOS.

### **6. Wymagania niefunkcjonalne**

#### **6.1 Wydajność**

- Renderowanie animacji nie powinno przekraczać 100ms na klatkę.
- System ticków (20 Hz) powinien działać stabilnie, bez opóźnień w przetwarzaniu logiki.

- Aplikacja powinna ładować się w czasie krótszym niż 2 sekundy na typowym połączeniu (LTE / Wi-Fi).
- Obsługa wielu lokalizacji nie może istotnie wpływać na szybkość działania aplikacji.

## 6.2 Bezpieczeństwo

- Dane lokalizacyjne użytkownika wykorzystywane są wyłącznie lokalnie w przeglądarce, nie są przysyłane na zewnętrzne serwery.
- Aplikacja nie wymaga rejestracji konta ani podawania danych osobowych.
- Rozwiązania zastosowane w aplikacji spełniają wymagania RODO.

## 6.3 Dostępność

- Responsywność UI: poprawne wyświetlanie i działanie na urządzeniach mobilnych, tabletach i komputerach.
- Możliwość korzystania bez dostępu do internetu (po pierwszym wczytaniu).

## 6.4 Skalowalność

- Architektura aplikacji umożliwia łatwe dodawanie nowych funkcji (np. więcej lokalizacji, dodatkowe tryby wyświetlania).
- W przyszłości możliwa integracja z API map (np. Google Maps, OSM) bez konieczności przebudowy podstawowej logiki.
- Aplikacja może zostać rozszerzona o treści edukacyjne, takie jak interaktywne lekcje dotyczące historii i kultury Japonii.

# 7. Harmonogram

## 7.1 Etapy realizacji

<b>Etap</b>	<b>Start</b>	<b>Koniec</b>	<b>Odpowiedzialny</b>
Analiza i specyfikacja	25-03-2025	08-04-2025	zespół analityków
Prototyp Backend	08-04-2025	29-04-2025	zespół developerów
Prototyp Frontend	29-04-2025	13-05-2025	zespół developerów
Testy i integracja	29-04-2025	03-06-2025	zespół testerów
Wdrożenie i prezentacja	03-06-2025	10-06-2025	szef projektu

## **8. Zarządzanie zmianami**

Zmiany w dokumencie zgłaszane są przez system Issues (GitHub), podlegają przeglądowi i zatwierdzeniu przez szefa projektu. Każda zmiana musi być oznaczona numerem wersji i datą.