

# Podstawowa Funkcjonalność Systemu

## 1. Dodanie nowego nawyku

**Given:** Użytkownik otworzył aplikację i znajduje się na głównym ekranie.

**When:** Użytkownik kliknie przycisk "Dodaj nawyk", wypełni pola (nazwa, częstotliwość, przypomnienia) i zapisze nawyk.

**Then:** Nowy nawyk pojawi się na liście nawyków na głównym ekranie.

---

## 2. Oznaczenie nawyku jako zrealizowanego

**Given:** Użytkownik ma dodany nawyk na liście i jest zalogowany w aplikacji.

**When:** Użytkownik kliknie w ikonę obok nawyku, aby oznaczyć go jako zrealizowany na dany dzień.

**Then:** Ikona zmieni kolor lub wygląd, a nawyk zostanie oznaczony jako zrealizowany w kalendarzu.

---

## 3. Usunięcie istniejącego nawyku

**Given:** Użytkownik ma co najmniej jeden nawyk na liście.

**When:** Użytkownik przytrzyma nawyk na liście i wybierze opcję "Usuń nawyk".

**Then:** Nawyk zostanie usunięty z listy, a dane związane z tym nawykiem znikną z historii.

---

## 4. Przypomnienie o nawyku

**Given:** Użytkownik ustawił przypomnienie dla nawyku w danym dniu i godzinie.

**When:** Nastąpi ustawiona godzina przypomnienia.

**Then:** Użytkownik otrzyma powiadomienie push przypominające o wykonaniu nawyku.

---

## 5. Wyświetlenie statystyk realizacji nawyków

**Given:** Użytkownik ma historię realizacji nawyków w aplikacji.

**When:** Użytkownik przejdzie do sekcji statystyk w menu aplikacji.

**Then:** Wyświetli się wykres lub inna wizualizacja z postęmem realizacji nawyków (np. ilość dni z rzędu, procent realizacji).

---

## 6. Edycja istniejącego nawyku

**Given:** Użytkownik ma dodany nawyk na liście.

**When:** Użytkownik wybierze nawyk, kliknie "Edytuj" i zmieni jego parametry (np. nazwę, częstotliwość).

**Then:** Zaktualizowane informacje o nawyku będą widoczne na liście i w szczegółach nawyku.

---

## 7. Rejestracja nowego użytkownika

**Given:** Nowy użytkownik otworzył aplikację po raz pierwszy.

**When:** Użytkownik wypełni formularz rejestracyjny i kliknie "Zarejestruj".

**Then:** Konto zostanie utworzone, a użytkownik zostanie przekierowany na ekran główny.

## Logowanie i Rejestracja

### 1. Udań utworzenie konta

- **Given:** Użytkownik nie ma jeszcze istniejącego konta.
- **When:** Użytkownik podaje unikalny adres e-mail i prosi o utworzenie konta.
- **Then:** System powinien utworzyć konto i potwierdzić jego utworzenie.

### 2. Adres e-mail już używany podczas tworzenia konta

- **Given:** Użytkownik ma już istniejące konto.
- **When:** Użytkownik podaje adres e-mail, który jest już używany.
- **Then:** System powinien poinformować, że adres e-mail jest już używany i nie pozwolić na utworzenie konta.

### 3. Udań logowanie

- **Given:** Użytkownik ma już istniejące konto.

- **When:** Użytkownik podaje poprawny adres e-mail i hasło.
- **Then:** System powinien przyznać dostęp użytkownikowi.

## 4. Nieprawidłowe hasło podczas logowania

- **Given:** Użytkownik ma już istniejące konto.
- **When:** Użytkownik podaje poprawny adres e-mail, ale nieprawidłowe hasło.
- **Then:** System powinien poprosić o odzyskanie hasła.

## 5. Proces odzyskiwania hasła

- **Given:** Użytkownik zapomniał hasła.
- **When:** Użytkownik prosi o odzyskanie hasła.
- **Then:** System powinien wysłać kod odzyskiwania, poprosić o podanie kodu odzyskiwania i przyznać dostęp, jeśli kod jest poprawny.

## 6. Nieprawidłowy kod odzyskiwania

- **Given:** Użytkownik poprosił o odzyskanie hasła.
- **When:** Użytkownik podaje nieprawidłowy kod odzyskiwania.
- **Then:** System nie powinien przyznać dostępu i poprosić użytkownika o ponowne podanie poprawnego kodu odzyskiwania.

## Operacje bazodanowe

### 1. Pobierz wszystkie aktywności dla konkretnego użytkownika

- **Given** użytkownik o ID user\_id istnieje w bazie danych
- **When** wykonuję zapytanie `SELECT * FROM Activity WHERE user_id = user_id;`
- **Then** system powinien zwrócić wszystkie rekordy w tabeli Activity dla user\_id

## 2. Pobierz wszystkie nawyki dla konkretnego użytkownika

- **Given** użytkownik o ID user\_id istnieje i ma przypisane nawyki w bazie danych
- **When** wykonuję zapytanie `SELECT Habit.* FROM Habit JOIN Activity ON Habit.id = Activity.habit_id WHERE Activity.user_id = user_id;`
- **Then** system powinien zwrócić wszystkie rekordy w tabeli Habit dla user\_id

## 3. Pobierz statystyki nawyków dla konkretnego użytkownika i nawyku

- **Given** użytkownik o ID user\_id oraz nawyk o ID habit\_id istnieją w bazie danych
- **When** wykonuję zapytanie `SELECT * FROM Habit_Statistics WHERE user_id = user_id AND habit_id = habit_id;`
- **Then** system powinien zwrócić rekord w tabeli Habit\_Statistics dla kombinacji user\_id i habit\_id

## 4. Pobierz łączną liczbę ukończonych nawyków dla konkretnego użytkownika

- **Given** użytkownik o ID user\_id istnieje w bazie danych
- **When** wykonuję zapytanie `SELECT SUM(total_completed) FROM Habit_Statistics WHERE user_id = user_id;`
- **Then** system powinien zwrócić łączną liczbę ukończonych nawyków dla user\_id

## 5. Pobierz najnowszą aktywność dla konkretnego użytkownika

- **Given** użytkownik o ID user\_id istnieje i posiada aktywności w bazie danych
- **When** wykonuję zapytanie `SELECT * FROM Activity WHERE user_id = user_id ORDER BY date DESC LIMIT 1;`
- **Then** system powinien zwrócić najnowszy rekord w tabeli Activity dla user\_id

