

Wymagania funkcjonalne MicroBlog

RF-01: Rejestracja użytkownika

Pole	Opis
Aktorzy	Gość (niezalogowany) System
Przypadki użycia	UC-01: Rejestracja użytkownika
Priorytet	Wysoki
Opis	Gość zakłada konto w systemie podając login, hasło i email. System zapisuje użytkownika oraz przechowuje hasło w postaci skrótu (hash) przy użyciu jBCrypt.
Warunki początkowe	Użytkownik nie jest zalogowany. System (aplikacja + baza) jest dostępny.
Scenariusz główny	<ol style="list-style-type: none">Gość otwiera formularz rejestracji.Gość wpisuje login, email i hasło.Gość zatwierdza formularz.System waliduje dane (wymagane pola, format email, unikalność login/email).System hashuje hasło z użyciem jBCrypt.System zapisuje użytkownika w bazie.System informuje o poprawnej rejestracji i przekierowuje do logowania lub loguje automatycznie (w zależności od implementacji).
Scenariusze alternatywne	<p>A1: Brak wymaganych danych</p> <ol style="list-style-type: none">W kroku 4 system wykrywa brak loginu/hasła/email.System odrzuca rejestrację i pokazuje komunikat o błędzie. <p>A2: Login lub email już istnieje</p> <ol style="list-style-type: none">W kroku 4 system wykrywa konflikt unikalności.System odrzuca rejestrację i pokazuje komunikat.

Pole	Opis
	<p>A3: Błąd bazy danych</p> <p>1) W kroku 6 występuje wyjątek zapisu.</p> <p>2) System nie tworzy konta i pokazuje komunikat o błędzie.</p>
Kryteria akceptacji	<p>System nie zapisuje hasła w postaci jawnej; w bazie istnieje tylko password_hash .</p> <p>Próba rejestracji z istniejącym loginem/email kończy się błędem.</p> <p>Po poprawnej rejestracji użytkownik może się zalogować utworzonym kontem.</p>

RF-02: Logowanie / Wylogowanie

Pole	Opis
Aktorzy	<p>Gość (logowanie)</p> <p>Użytkownik zalogowany (wylogowanie)</p> <p>System</p>
Przypadki użycia	<p>UC-02: Logowanie</p> <p>UC-03: Wylogowanie</p>
Priorytet	Wysoki
Opis	Użytkownik loguje się podając login i hasło. System weryfikuje hasło (jBCrypt) i tworzy sesję. Użytkownik może zakończyć sesję poprzez wylogowanie.
Warunki początkowe	<p>Logowanie: użytkownik nie jest zalogowany.</p> <p>Wylogowanie: użytkownik jest zalogowany (aktywna sesja).</p>
Scenariusz główny	<p>Logowanie:</p> <ol style="list-style-type: none"> Gość otwiera formularz logowania. Gość podaje login i hasło. Gość zatwierdza formularz. System pobiera użytkownika po loginie. System porównuje hasło z password_hash (BCrypt). System tworzy sesję i zapisuje identyfikator użytkownika (np. username lub userId). System przekierowuje użytkownika do panelu/dashboardu.

Pole	Opis
	<p>Wylogowanie:</p> <ol style="list-style-type: none"> 1. Użytkownik wybiera akcję „Wyloguj”. 2. System unieważnia sesję (invalidate). 3. System przekierowuje na stronę publiczną.
Scenariusze alternatywne	<p>A1: Niepoprawne dane logowania</p> <ol style="list-style-type: none"> 1) W kroku 4 lub 5 system wykrywa brak użytkownika albo niezgodne hasło. 2) System nie tworzy sesji i pokazuje komunikat. <p>A2: Sesja wygasła / brak sesji</p> <ol style="list-style-type: none"> 1) System traktuje użytkownika jak niezalogowanego i wymaga ponownego logowania.
Kryteria akceptacji	<p>Po poprawnym logowaniu sesja jest aktywna do momentu wylogowania.</p> <p>Po wylogowaniu dostęp do zasobów wymagających logowania jest zablokowany.</p> <p>Przy błędnym hasle/loginie system nie tworzy sesji.</p>

RF-03: Tworzenie postów

Pole	Opis
Aktorzy	Użytkownik zalogowany System
Przypadki użycia	UC-04: Dodanie posta
Priorytet	Wysoki
Opis	Zalogowany użytkownik publikuje post. System zapisuje treść oraz datę publikacji (created_at).
Warunki początkowe	Użytkownik jest zalogowany.
Scenariusz główny	<ol style="list-style-type: none"> 1. Użytkownik otwiera formularz dodawania posta. 2. Użytkownik wpisuje treść posta. 3. Użytkownik publikuje post.

Pole	Opis
	4. System waliduje treść (np. niepusta, maksymalna długość). 5. System zapisuje post wraz z <code>created_at</code> . 6. System wyświetla post na linii czasu.
Scenariusze alternatywne	A1: Użytkownik niezalogowany 1) System blokuje dostęp do publikacji i przekierowuje do logowania. A2: Treść posta niepoprawna 1) W kroku 4 system odrzuca post i pokazuje komunikat.
Kryteria akceptacji	Po dodaniu posta jest on widoczny na linii czasu autora. Każydy post ma ustawioną datę publikacji (<code>created_at</code>). System nie pozwala dodać posta o pustej treści.

RF-04: Publiczne posty

Pole	Opis
Aktorzy	Gość Użytkownik zalogowany System
Przypadki użycia	UC-05: Przeglądanie postów publicznych
Priorytet	Średni
Opis	System umożliwia przeglądanie publicznych postów bez logowania.
Warunki początkowe	System jest dostępny.
Scenariusz główny	1. Użytkownik/Gość otwiera widok postów publicznych. 2. System pobiera listę publicznych postów. 3. System wyświetla posty (np. od najnowszych).
Scenariusze alternatywne	A1: Brak postów 1) System wyświetla pustą listę i komunikat „Brak postów”.
Kryteria akceptacji	Posty publiczne są dostępne bez logowania. Lista postów jest poprawnie renderowana dla użytkownika.

RF-05: Follow / Unfollow

Pole	Opis
Aktorzy	Użytkownik zalogowany System
Przypadki użycia	UC-06: Obserwowanie użytkownika (Follow) UC-07: Cofnięcie obserwowania (Unfollow)
Priorytet	Średni
Opis	Użytkownik może obserwować innych użytkowników. Posty obserwowanych osób pojawiają się na jego dashboardzie (linii czasu). Po Unfollow posty tej osoby znikają z dashboardu.
Warunki początkowe	Użytkownik jest zalogowany. Użytkownik, którego chcemy obserwować, istnieje.
Scenariusz główny	<p>Follow:</p> <ol style="list-style-type: none">1. Użytkownik wybiera profil/innego użytkownika.2. Użytkownik klika „Follow”.3. System zapisuje relację obserwowania.4. System aktualizuje dashboard tak, aby zawierał posty obserwowanego użytkownika. <p>Unfollow:</p> <ol style="list-style-type: none">1. Użytkownik wybiera obserwowanego użytkownika.2. Użytkownik klika „Unfollow”.3. System usuwa relację obserwowania.4. System aktualizuje dashboard, usuwając z niego posty tej osoby.
Scenariusze alternatywne	<p>A1: Próba obserwowania samego siebie</p> <ol style="list-style-type: none">1) System blokuje akcję i informuje o błędzie. <p>A2: Relacja już istnieje</p> <ol style="list-style-type: none">1) System nie tworzy duplikatu (operacja jest idempotentna) i pozostawia stan bez zmian. <p>A3: Użytkownik niezalogowany</p> <ol style="list-style-type: none">1) System przekierowuje do logowania.

Pole	Opis
Kryteria akceptacji	Po Follow posty obserwowanego pojawiają się na dashboardzie. Po Unfollow posty obserwowanego nie są widoczne na dashboardzie. System nie pozwala obserwować samego siebie.