

# Mathify – Aplikacja do śledzenia rozwoju matematycznego.

## 1. Wstęp

### 1.1. Opis Aplikacji

Aplikacja, która będzie służyć jako element śledzenie rozwoju i postępu matematycznego uczniów danego nauczyciela. Przed wszystkim uczeń będzie mógł korzystać z materiałów, które będzie udostępniał nauczyciel a także, sam będzie mógł wysyłać własne pliki, to znaczy pliki rozwiązań zadań maturalnych. Każdy nauczyciel będzie mógł stworzyć swój kurs i dodawać do niego niezależnie uczniów. Uczniowie się w tym nie będą wiedzieć. Kurs będzie mógł mieć swoje rozdziały i podrozdziały. Każdy podrozdział i każdy plik w podrozdziale będzie mógł mieć ustawioną widoczność czy dany uczeń może to wiedzieć. Uczniowie do każdego podrozdziału będą mogli wysyłać dane pliki o ile nauczyciel odblokuje im dostęp wysyłania. Jednym z najbardziej ukrytych a także najważniejszym elementem aplikacji będzie funkcja automatycznego sprawdzania pracy domowej. Na czym to polega? Uczeń wysyłając tą pracę wysyła ją bezpośrednio do Gemine / lub innego AI, które sprawdza daną pracę w i zadanej formie wysyła odpowiedź o szczegółowych wynikach, które później będą weryfikowane przez nauczyciela i potwierdzone. Po potwierdzeniu wiadomość o wynikach pracy domowej będzie przekazywana do ucznia. Uczeń również będzie miał dostęp do takiego panelu, w którym będzie mógł śledzić własne statystki, dotyczące jego pracy.

### 1.2. Powód realizacji

Powodem realizacji jest zwiększenie optymalności pracy nauczyciela. Zamiast wysyłać prace przez Messenger nauczyciel może przez system odblokowywać lub nawet ustawiać czas, w którym dana praca się odblokuje. Jednocześnie uprawi to proces sprawdzania prac domowych.

### 1.3. Główny cel projektu.

Ułatwienie pracy nauczyciela, usprawnienie wymiany informacji pomiędzy uczeń – nauczyciel.

## 2. Cel projektu.

### 2.1. Główny cel.

Stworzenie systemu, który będzie częściowo obsługiwać relacje pomiędzy nauczyciel – uczeń. Uczeń będzie mógł korzystać z materiałów udostępnianych przez nauczyciela. Będzie mógł jednocześnie wysyłać prace domowe, które będą sprawdzane przez AI a następnie weryfikowane przez nauczyciela i gotowe do odesłania.

## 2.2. Cele szczegółowe.

### 2.2.1. Admin

Będzie istniała rola Admina, który będzie mógł tak naprawę robić wszystko. To znaczy tworzyć konta nauczycieli a także konta uczniów. Usuwać je a także edytować dane związane z nimi.

### 2.2.2. Nauczyciel

Tworzenie kursów, do których będzie mógł dodawać uczniów. Kursy będą jako materiał zewnętrzny to znaczy uczeń nie będzie miał wpływu na edycje. Tylko będzie mógł korzystać z danych oferowanych przez dany kurs. Będzie mógł również, tam gdzie będzie to odblokowane dodawać własne rozwiązania w formie plików. Uczniowie nie będą wiedzieć siebie ze sobą uczestnikami kursu. Nauczyciel będzie mógł weryfikować czy sprawdzenie pracy domowej przez dane AI jest poprawne, jeżeli tak będzie mógł potwierdzić i dany uczeń będzie mógł odczytać swój wynik a także szczegółowy opis błędów.

### 2.2.3. Uczeń.

Uczestnictwo w kursie. Korzystanie z materiałów udostępnianych przez dany kurs. Możliwość wysyłania prac domowych dla odblokowanych podrozdziałów. Bo to w tym miejscu będziemy mogli rozpatrywać prace domowe. Także śledzenie swoich postępów pracy na oddzielnym oknie.

## 3. Zakres projektu.

### 3.1. System logowania

System logowania za pomocą loginu i hasła. Aby móc dostać się do kursu. Użytkownik niezalogowany nie widzi nic więcej niż stronę tytułową.

### 3.2. System rejestracji

System rejestracji będzie mógł być dostępny dla wszystkich, lecz z pewnymi ograniczeniami. Można zarejestrować nowe konto, lecz jak zarejestruje je ktoś. Admin czyli rola admina będzie musiała potwierdzić że te dane konto, będzie mogło być uczniem lub nauczycielem. W zależności od tego co sobie wybierze. Oczywiście system rejestracji będzie mógł być dostępny z panelu admina. Admin będzie mógł tworzyć uczniów i nauczycieli. Wtedy już nie będzie trzeba żadnych elementów potwierdzenia. Dodatkowo nauczyciel będzie mógł tworzyć uczniów, i przypisywać ich od razu do swoich utworzonych kursów.

Podczas rejestracji będą wymagane takie informacje jak:

- Imię
- Nazwisko
- Login
- Hasło
- E-Mail
- Rola użytkownika (Admin będzie już stworzony i będzie jeden, więc można wybrać jedynie uczeń/nauczyciel)

### **3.3. Kursy.**

#### **3.3.1. Zarządzanie kursami**

Nauczyciel będzie mógł tworzyć kursy, w których będą mogły być rozdziały i podrozdziały o określonych nazwach ustalonych przez nauczyciela. Nauczyciel będzie mógł dodawać uczniów i dla OKREŚLONEGO ucznia będzie mógł zmieniać widoczność danego PODROZDZIAŁU lub ROZDZIAŁU. I będzie mógł ustawić dla określonego PODROZDZIAŁU widoczność do wstawienia pracy domowej. Wszystkie elementy domyślnie będą niewidoczne, a nauczyciel będzie mógł ustawiać widoczność manualnie lub automatycznie (czyli po jakieś godzinie). Praca domowa w kursie automatycznie będzie sprawdzana przez AI, lecz zanim zostanie wysłany wynik pracy domowej musi być zaakceptowana przez nauczyciela, który będzie mógł modyfikować treść komunikatów.

#### **3.3.2. Sprawdzanie pracy domowej przez AI**

Uczeń wysyłając pracę domową, będzie ona automatycznie sprawdzana przez AI. Sztuczna inteligencja, będzie wysyłała jako odpowiedź tej pracy domowej, dane w formie JSON, które będą wklejane do tabelek, które nauczyciel będzie mógł modyfikować. Po ewentualnej modyfikacji nauczyciel może potwierdzić poprawność sprawdzonej pracy i odesłać do ucznia do wglądu szczegółowego. Treść JSON to będzie taka:

```
{[{"id": id_pracy_domowej, "zadanie": zadanie, "ilość punktów zdobytych": ilosc_punkow, "maksymalna ilość punktów": maksymalna_ilosc_punkow, "komentarz": komentarz}]}
```

Jeżeli chodzi o szczegółowy request do AI. To należy przesłać plik, który będzie zawierał pracę domową, może być to PDF na początek. I na podstawie tego PDF będzie sprawdzać pracę. Tutaj trzeba zastosować specjalny tekst napisany do tego AI.

### **3.4. Śledzenie postępów.**

Po tym nauczyciel potwierdzi że dana praca domowa jest sprawdzona dobrze, lub po weryfikacji i poprawy pracy domowej uczeń dostaje powiadomienie o sprawdzonej pracy i wyniki automatycznie zapisują się do systemu. Po pierwszej sprawdzonej pracy uczeń ma dostęp do wykresu, w którym jest uwzględniony postęp ile procent udało się zdobyć z danego działu. To znaczy jest to prosty wykres gdzie na osi pionowej są procenty, a na osi poziomej temat. Generalnie będą to słupki. Uczeń może kliknąć w dany słupek, aby odczytać szczegóły dotyczące poszczególnych zadań. Każde zadania jak było to wspominane dział temu będzie zawierało dane takie jak ilość zdobnych punktów, maksymalna ilość zdobytych punktów, i komentarz, zawierający uwagi jeżeli dane zadanie było zrobione źle. Jeżeli uczeń zdobył max punktów komentarz nie jest obowiązkowy.

### **3.5. Profile**

Każde konto będzie miało dostęp do zakładki „Mój profil” w której będą opisane szczegóły konta. Jaki typ to jest konta i możliwość zmiany hasła lub zmiany danych, które będą należeć do konta. Można również zobaczyć do jakich kursów się należy. Generalnie takie podstawowe rzeczy związane z kontem.

## 4. Grupa docelowa

Aplikacja będzie przeznaczona dla osób, które prowadzą korepetycje lub inne zajęcia. Docelowo, póki co ja jestem taką osobą więc sam dla siebie będę grupą docelową. Lecz aby zadbać o łatwiejszą rozbudowę chciałbym aby wiele nauczycieli mogły tworzyć wiele kursów i mieć tam przypisanych uczniów. Więc to zakładam jako element początkowy. Rodzice nie muszą mieć kontroli nad tym, ponieważ uważam że uczniowie będą z tego korzystać w przyszłości może zostanie to oddane do rozbudowy.

## 5. Wymagania funkcjonalne

- Użytkownik może się zalogować i zarejestrować.
- Admin może zatwierdzić takie konto.
- Admin może utworzyć użytkownika danego typu.
- System zapisuje wyniki uczniów w bazie danych po sprawdzeniu.
- System za pomocą AI sprawdza wyniki.
- Aplikacja generuje wykresy.
- Nauczyciel tworzy kursy, w których są rozdziały i podrozdziały. A w podrozdziałach są pliki.
- Nauczyciel może kontrolować dostępem do danych plików czy rozdziałów lub podrozdziałów.

## 6. Wymagania niefunkcjonalne.

- Responsywny interfejs
- Intuicyjna obsługa
- Nowoczesny wygląd

## 7. Narzędzia i Technologie

### 7.1. Frontend

Frontend będzie pisany w React & Next.

### 7.2. Backend

Backend będzie pisany w Next.

### 7.3. Baza Danych

Baza danych to PostgreSQL

### 7.4. Narzędzia dodatkowe

Będzie używany Git do kontroli wersji. Na bieżąco pisane commity.