

Dzień dobry,

Z przyjemnością informujemy, że prace nad zleceniem "Rozbudowa funkcjonalności Learndash - generator szybkich testów" zostały w całości zakończone.

Wszystkie wprowadzone zmiany zostały już wdrożone na Państwa serwerze ('luxedu.ekursy.online') i są w pełni aktywne. Przeprowadziliśmy testy, które potwierdziły, że nowa funkcjonalność działa stabilnie i zgodnie z wszystkimi założeniami opisanymi w zleceniu.

Podsumowanie zrealizowanych prac:

- **Generator quizów:** Stworzyliśmy i wdrożyliśmy pełną logikę backendową (PHP/AJAX), która pobiera pytania na podstawie kryteriów wybranych w formularzu (poziom trudności, dział, limit czasowy, liczba i typy zadań).
- **Automatyczne tworzenie i przypisanie:** System dynamicznie generuje nowy quiz w strukturach LearnDash i przypisuje go do zalogowanego użytkownika.
- **Przekierowanie użytkownika:** Po pomyślnym wygenerowaniu quizu użytkownik jest **natychmiast i automatycznie przekierowywany** na stronę z nowo utworzonym testem, co było kluczowym wymogiem projektu.
- **Dodatkowe konfiguracje:** Zgodnie z ustaleniami, utworzyliśmy wymaganą taksonomię "Poziom trudności" oraz pole do zarządzania globalnym limitem pytań.
- **Obsługa przypadków brzegowych:** Generator poprawnie obsługuje sytuację, w której w bazie nie ma wystarczającej liczby pytań, informując o tym użytkownika i tworząc test z dostępnej puli.

Wszystkie elementy zlecenia zostały zrealizowane. Dla pełnej transparentności i Państwa archiwum, w załączniku do tej wiadomości przesyłamy również spakowany folder zawierający wszystkie pliki, które zostały przez nas utworzone lub zmodyfikowane w trakcie realizacji zadania.

Prosimy o przetestowanie nowej funkcjonalności i potwierdzenie, że wszystko działa zgodnie z Państwa oczekiwaniami.

Dziękujemy za owocną współpracę przy tym zadaniu. Mamy nadzieję, że jest to udany początek naszej dalszej, stałej współpracy.

Pozdrawiamy serdecznie,
Zespół Code Olympus Studio

Pytania 29.07

Natrafiliśmy jednak na jedną, standardową kwestię techniczną, którą musimy wspólnie rozwiązać, aby móc w pełni ruszyć z pracą. Chodzi o autoryzację zapytań do REST API WordPressa.

Podczas próby komunikacji z API LearnDash (w celu np. pobrania listy quizów) otrzymujemy błąd `401 Unauthorized`. Oznacza to, że serwer odrzuca nasze zapytanie z powodu braku uprawnień. Domyślnie WordPress nie pozwala na zewnętrzną komunikację z API przy użyciu standardowego hasła do panelu – jest to zabezpieczenie, które wymaga dodatkowej konfiguracji.

Sprawdziliśmy również, że w panelu edycji profilu użytkownika brakuje standardowej dla nowszych wersji WordPressa sekcji "Application Passwords" (Hasła aplikacji), która rozwiązałaby ten problem.

Abyśmy mogli kontynuować, potrzebujemy jednej z dwóch rzeczy:

1. Preferowana opcja: Dostęp do plików serwera (FTP/SFTP)

To dla nas najszybsze i najpełniejsze rozwiązanie. Dostęp do plików pozwoli nam na:

- Samodzielne i bezpieczne skonfigurowanie autoryzacji do REST API.
- Płynne pobranie ID zalogowanego użytkownika po stronie serwera.
- Co najważniejsze, dostęp do plików jest nam również niezbędny, aby zaimplementować końcowy etap zlecenia – czyli skrypt, który po wygenerowaniu testu automatycznie przekieruje na niego użytkownika. Bez możliwości umieszczenia tego skryptu w plikach motywu, wykonamy logikę tworzenia quizu, ale nie będziemy w stanie zrealizować samego przekierowania.

Posiadając dostęp, sprawnie poradzimy sobie z konfiguracją i będziemy mogli od razu przejść do właściwej części zadania.

2. Opcja alternatywna: Instalacja wtyczki po Państwa stronie

Jeśli nie chcą Państwo udostępniać dostępu do serwera, drugą opcją jest zainstalowanie przez Państwa wtyczki, która zarządza dostępem do REST API. Rekomendujemy tę wtyczkę, ponieważ oficjalnie wspiera LearnDash:

- Wtyczka: <https://wordpress.org/plugins/wp-rest-api-authentication/>

Po jej instalacji i aktywacji prosimy o wygenerowanie dla nas klucza API lub danych do autoryzacji (Basic Auth) i przesłanie ich do nas. Należy jednak pamiętać, że ta opcja nie rozwiąże problemu z implementacją skryptu przekierowującego, co jest kluczową częścią funkcjonalności.

Podsumowując, gorąco rekomendujemy opcję nr 1, ponieważ pozwoli nam ona zrealizować 100% zakresu zlecenia w najszybszy i najbardziej efektywny sposób.

Prosimy o informację, które rozwiązanie jest dla Państwa wygodniejsze. Jesteśmy gotowi do dalszego działania zaraz po uzyskaniu potrzebnych dostępuów.

Dane dostępowe SFTP i SSH

Host: web13.jdm.pl

Login: awrntbmp

Port: 2233

Hasło: 6W5dLVf2UFf1Jd33ULfb

dostęp

<https://luxedu.ekursy.online/wp-admin/>

k.kwasniewski@etechnologie.pl

GDgTVim\$Mdj)ZtWTrLe\$@6cE

1. Środowisko deweloperskie: Czy posiadają Państwo środowisko deweloperskie/stagingowe, na którym moglibyśmy bezpiecznie przeprowadzić prace? Praca na działającej stronie produkcyjnej jest ryzykowna, dlatego zawsze rekomendujemy prowadzenie developmentu na osobnej kopii.

Obecnie działamy na stronie "live", ponieważ i tak jest stagingiem. Nie działa oficjalnie.

2. Taksonomia "Poziom trudności": Czy nowa taksonomia dla pytań (`sfwd-question`) o nazwie "Poziom trudności" już istnieje, czy jej utworzenie również leży po naszej stronie?

Tak trzeba stworzyć najszybciej to zrobić przez ACF.

3. Pola ACF: Czy pole w ustawieniach do globalnego limitu liczby zadań (utworzone w ACF) już istnieje, czy mamy je również stworzyć?

Trzeba stworzyć.

4. Istniejąca baza pytań: Czy w systemie znajdują się już przykładowe pytania z przypisanymi kategoriami (działami), na których będziemy mogli testować działanie generatora? Pomoże to nam zweryfikować poprawność działania logiki.

Tak

5. Obsługa brzegowych przypadków: Co powinno się wydarzyć w sytuacji, gdy generator nie znajdzie wystarczającej liczby pytań pasujących do wybranych przez użytkownika kryteriów (np. użytkownik wybierze 20 pytań z działu, w którym jest tylko 5)? Czy system ma wyświetlić stosowny komunikat, czy może wygenerować test z mniejszą liczbą pytań?

Gdy w bazie nie będzie wystarczającej liczby pytań należy wygenerować quiz z mniejszą ilości i poinformować o tym użytkownika poprzez komunikat.

6. Wersje oprogramowania: Czy mogliby Państwo podać, jakiej wersji WordPressa, LearnDash oraz PHP używa Państwa serwer?

PHP 8.1.x

LearnDash 4.22.1

WordPress 6.8.1

Zadanie #1

Do wykonania jest Generator szybkich testów.

Generator szybkich testów

Tutaj możesz wygenerować każdy test

Wybierz poziom trudności Lista rozwijana

Wybierz dział Lista rozwijana

Liczba zadań w teście Wpisz liczbę od 1 do 30

Test ograniczony czasowo Nie Tak

Typy zadań Lista rozwijana

Generuj

760 x 522

Do wykonania jest back-end.

Widok front-endu jest tutaj: <https://luxedu.ekursy.online/szybkie-testy/> (dane do logowania).

Zadanie polega na tym, by w PHP oraz Ajax utworzyć generator pobierający zadania po wskazanych danych i generujący nowy quiz w LearnDashu, który zostanie przypisany do użytkownika. Po wygenerowaniu należy go od razu użytkownikowi wyświetlić, czyli przekierować na jego widok.

- Poziom trudności trzeba dodać w pytaniach. Można do tego stworzyć nową taksonomię do sfwd-question.
- Jako dział użyć kategorii pytań

Kategoria Pytanie
(opcjonalnie)

Możesz przypisać kategorię do pytania.
Kategorie są widoczne np. w statystykach.
Kategoriami można zarządzać w ustawieniach ogólnych.

--- Brak kategorii ---

--- Utwórz nową kategorię ----

--- Brak kategorii ---

<https://luxedu.ekursy.online/wp-admin/admin.php?page=questions-options>

- Liczba zadań w teście nich będzie ograniczona globalnym ustawieniem. Najlepiej dodać pole ustawień w ACF.
- Test ograniczony czasowo pozwoli wprowadzić to pole. Pole rozwijane tak pozwoli wpisać limit czasu między 15 - 999 minut.
- Typy zadań, domyślnie niech to jest typ "dowolne" czyli każdy możliwy typ zadań zostanie wylosowany. Typy pytań oznaczają [to](#):

Rodzaj odpowiedzi

Wybór jednokrotny

Wielokrotny wybór

"Własna" odpowiedź

Sortowanie

Dopasowanie

Uzupełnij puste pola

Ocena

Opracowanie / własna odpowiedź