



Bar Dawid, Dobranowski Bartłomiej, Piotr Prezydent
12345,12345,12345

Informatyka

Przykładowa praca inżynierska wykonana na seminarium

Praca inżynierska

Praca wykonana pod kierunkiem
dr Krzysztof Balicki

*Serdecznie dziękuję . . . tu ciąg dalszych podziękowań np. dla promotora,
żony, męża, sąsiada/ki, psa, kota itp.*

Spis treści

1. Wprowadzenie	6
1.1. Opis Problemu	6
1.2. Analiza i opis istniejących rozwiązań	6
2. Opis technologii i algorytmów	8
2.1. Technologie użyte w systemie	8
2.2. Struktura danych	8
2.3. Algorytm Losowania Zestawów	9
2.4. Bezpieczeństwo i ograniczenia algorytmu	9
3. Prezentacja wyników	10
3.1. Opis aplikacji	10
3.2. Prezentacja wyników losowania	10
3.3. Interfejs użytkownika	10
3.4. Print screeny aplikacji	11
3.5. Wnioski	11
3.6. Możliwości rozbudowy systemu	11
Bibliografia	13
Spis rysunków	14
Spis tabel	15
Spis listingów	16
Streszczenie pracy	17
Abstract	18
Oświadczenie studenta o samodzielności pracy	19
Oświadczenie studenta o zgodności wersji papierowej i elektronicznej pracy	20

1. Wprowadzenie

Współczesne nauczanie przedmiotów technicznych wymaga od prowadzących zajęcia efektywnego zarządzania procesem oceniania oraz sprawdzania wiedzy studentów. W szczególności w przypadku przedmiotów praktycznych, takich jak programowanie czy analiza danych, istotne jest zapewnienie studentom różnorodnych zestawów zadań egzaminacyjnych, co pozwala uniknąć powtarzalności i zwiększa obiektywność ocen.

Celem niniejszej pracy jest opracowanie narzędzia informatycznego, które automatyzuje proces losowania zestawów zadań dla studentów. Narzędzie zostało zaimplementowane w języku PHP i udostępnia prosty interfejs webowy, umożliwiający przydzielanie zadań z dwóch głównych kategorii: programowania w języku Java oraz tworzenia arkuszy i analiz w Excelu. System ten pozwala na łatwe generowanie unikalnych zestawów dla kolejnych studentów oraz przechowywanie historii przydzielonych zadań w sesji użytkownika.

1.1. Opis Problemu

Podczas przygotowywania kolokwίων i sprawdzianów praktycznych nauczyciele napotykają problem losowego przydzielania tematów studentom w taki sposób, aby każdy student otrzymał unikalny zestaw zadań z różnych kategorii tematycznych. W warunkach tradycyjnych, ręczne losowanie lub przydzielanie zestawów może być czasochłonne i podatne na błędy. Ponadto istnieje ryzyko powtarzalności przydzielanych zadań oraz trudności w utrzymaniu spójnej dokumentacji wyników.

Problem ten wymaga zatem opracowania systemu, który:

1. Automatycznie przydziela zadania z określonych kategorii (np. Java i Excel).
2. Gwarantuje unikalność zestawów dla każdego studenta.
3. Rejestruje historię przydzielonych zadań w celu późniejszej analizy i kontroli.
4. Udostępnia proste, przyjazne dla użytkownika środowisko webowe.

Rozwiązanie tego problemu zwiększa efektywność pracy nauczyciela, minimalizuje ryzyko powtarzalności zadań oraz ułatwia organizację egzaminów i kolokwίων.

1.2. Analiza i opis istniejących rozwiązań

W literaturze i praktyce edukacyjnej można wyróżnić kilka podejść do przydzielania zadań studentom:

1. Ręczne losowanie i przydzielanie zestawów – najprostsze podejście polegające na fizycznym losowaniu kart z zadaniami lub przygotowaniu arkuszy z przypisanymi zadaniami. Metoda ta jest czasochłonna i nie gwarantuje pełnej losowości ani łatwego zapisu historii wyników.
2. Arkusze kalkulacyjne (Excel, Google Sheets) – istnieją rozwiązania wykorzystujące funkcje arkuszy kalkulacyjnych do losowania zadań przy użyciu wbudowanych funkcji losowych i filtrowania danych. Chociaż metoda ta może być w pełni automatyczna, wymaga znajomości formuł i jest ograniczona przy pracy wieloosobowej lub przy dużej liczbie zadań.

35 3. Specjalistyczne oprogramowanie do egzaminów i testów – komercyjne systemy pozwalają na two-
36 rzenie testów z automatycznym przydzielaniem zadań, monitorowaniem postępów studentów i gene-
37 rowaniem statystyk. Ich główną wadą jest koszt oraz konieczność instalacji i obsługi dedykowanego
38 oprogramowania.

39 W tym kontekście opracowane narzędzie w PHP stanowi rozwiązanie lekkie, elastyczne i łatwo do-
40 stępne w środowisku webowym. Dzięki wykorzystaniu sesji, losowania elementów z tablicy oraz prostego
41 interfejsu użytkownika, system pozwala szybko przydzielać unikalne zestawy zadań, jednocześnie rejestru-
42 jąc historię wyników bez konieczności stosowania zewnętrznych aplikacji czy skomplikowanych arkuszy
43 kalkulacyjnych.

2. Opis technologii i algorytmów

Niniejszy rozdział przedstawia technologie oraz algorytmy wykorzystane w opracowanym systemie do losowania zestawów zadań na kolokwium. Omówione zostaną zarówno narzędzia programistyczne, jak i logika działania aplikacji.

2.1. Technologie użyte w systemie

System został zaimplementowany jako aplikacja webowa w języku PHP, co umożliwia łatwe uruchamianie w przeglądarce internetowej bez konieczności instalacji dodatkowego oprogramowania. Wybrane technologie i narzędzia to:

1. PHP – język skryptowy po stronie serwera, wykorzystywany do logiki biznesowej, zarządzania sesjami i generowania dynamicznego HTML.
2. HTML i CSS – służą do tworzenia interfejsu użytkownika oraz stylizacji strony, w tym przycisków, formularzy i obszarów wyświetlania historii losowań.
3. Mechanizm sesji w PHP `session_start()` – pozwala na przechowywanie danych między kolejnymi wywołaniami strony, co umożliwia zachowanie stanu aplikacji, np. listy pozostałych zadań i historii losowań.
4. Przeglądarka internetowa – umożliwia interakcję użytkownika z aplikacją oraz wyświetlanie wyników losowań w formie czytelnej dla nauczyciela lub administratora.

Zastosowanie tych technologii pozwala na prostą, a jednocześnie funkcjonalną implementację systemu, który jest dostępny w dowolnym środowisku z obsługą PHP i przeglądarki internetowej.

2.2. Struktura danych

Podstawowym elementem systemu jest tablica wielowymiarowa `$tasks`, która przechowuje wszystkie dostępne zadania wraz z ich typami. Każdy element tablicy zawiera:

- nazwę zadania (ciąg tekstowy),
- typy zadań (np. EX – Excel, SO – Excel, BS, DZ – Java).

Podczas inicjalizacji aplikacji tablica `$tasks` jest przetwarzana na dwie listy: `$tasksExcel` i `$tasksJava`, które przechowują zadania dostępne do losowania dla każdej kategorii. Ponadto w sesji PHP przechowywane są:

- numer kolejnego studenta `$_SESSION['student']`,
- historia losowań `$_SESSION['records']`,
- aktualne listy zadań do wylosowania `$_SESSION['tasksExcel']` i `$_SESSION['tasksJava']`.

Taka struktura danych pozwala na szybkie losowanie zadań, uniknięcie powtarzalności oraz łatwe wyświetlanie wyników w interfejsie użytkownika.

2.3. Algorytm Losowania Zestawow

Proces losowania opiera się na kilku krokach:

1. Sprawdzenie, czy sesja została zainicjalizowana – jeśli nie, system tworzy listy zadań Excel i Java oraz inicjalizuje licznik studentów i historię losowań.
2. Losowanie zadania – z list `$tasksExcel` i `$tasksJava` losowane są po jednym zadaniu dla kolejnego studenta przy użyciu funkcji `rand()`.
3. Usunięcie wylosowanego zadania – `array_splice()` usuwa wybrane zadania z list, aby uniknąć powtórzeń w przyszłych losowaniach.
4. Rejestracja wyniku losowania – wylosowany zestaw zapisywany jest w tablicy `$_SESSION['records']` i numer student jest inkrementowany.
5. Wyświetlenie wyników – w interfejsie użytkownika prezentowana jest lista wszystkich wylosowanych zestawów.
6. Algorytm jest prosty, ale skuteczny – zapewnia losowy, unikalny przydział zadań przy minimalnym nakładzie obliczeniowym.

2.4. Bezpieczeństwo i ograniczenia algorytmu

Chociaż algorytm losowania działa poprawnie w środowisku lokalnym lub dla jednego użytkownika, istnieją pewne ograniczenia:

- Brak obsługi wielu użytkowników jednocześnie – w przypadku równoległego dostępu do systemu, sesje mogą się nadpisywać.
- Brak walidacji danych wejściowych – formularz nie zabezpiecza przed niepoprawnym wywołaniem metody POST.
- Ograniczona skalowalność – lista zadań przechowywana w pamięci sesji może być niewystarczająca dla dużej liczby studentów lub bardzo rozbudowanego zestawu zadań.
- Pomimo tych ograniczeń, system jest wystarczający do zastosowań edukacyjnych w małej skali i stanowi dobrą podstawę do dalszej rozbudowy.

3. Prezentacja wyników

Celem niniejszego rozdziału jest przedstawienie działania opracowanego systemu, jego interfejsu użytkownika oraz efektów losowania zestawów zadań dla studentów. W rozdziale pokazano, w jaki sposób aplikacja prezentuje wyniki, w jaki sposób użytkownik wchodzi w interakcję z systemem oraz przykładowe print screeny ilustrujące funkcjonalność aplikacji.

3.1. Opis aplikacji

Aplikacja została zaprojektowana jako prosty system webowy, umożliwiający losowanie zestawów zadań w dwóch kategoriach: Java i Excel. Interfejs użytkownika składa się z następujących elementów:

- Nagłówek strony – zawiera nazwę systemu „IRZ” i pełni funkcję informacyjną.
- Obszar historii losowań – wyświetla wylosowane zestawy dla kolejnych studentów w postaci listy tekstowej. Każdy wpis zawiera numer studenta oraz przydzielone zadania w obu kategoriach.
- Przycisk losowania zestawu – po kliknięciu generuje nowy zestaw zadań dla kolejnego studenta. Przycisk jest dezaktywowany, jeśli nie ma już dostępnych zadań w którejkolwiek kategorii.
- Aplikacja wykorzystuje sesje PHP do przechowywania stanu między kolejnymi wywołaniami strony, dzięki czemu wszystkie wylosowane zestawy pozostają widoczne w historii, nawet po odświeżeniu strony.

3.2. Prezentacja wyników losowania

Po naciśnięciu przycisku „Losuj zestaw”, aplikacja wykonuje następujące kroki:

1. Losuje po jednym zadaniu z list `tasksExcel` i `tasksJava`.
2. Tworzy rekord w formacie:

```
Osoba: [numer studenta]
       [zadanie Java]
       [zadanie Excel]
```

3. Dodaje rekord do historii `$_SESSION['records']` i wyświetla go w obszarze wyników.
4. Aktualizuje stan list zadań, usuwając już wylosowane elementy, aby zapewnić unikalność zestawów.

Dzięki temu każdy student otrzymuje unikalny zestaw zadań, a nauczyciel może w prosty sposób śledzić przebieg losowań oraz sprawdzić, jakie zadania zostały już przydzielone.

3.3. Intefrejs użytkownika

Interfejs użytkownika został zaprojektowany z myślą o czytelności i prostocie obsługi. Główne cechy:

- Minimalistyczny design – wszystkie elementy rozmieszczone w centralnym oknie o stałej szerokości i wysokości.

- 132 • Wyraźne oznaczenie kategorii zadań – w nagłówku okna podawana jest liczba pozostałych zadań w
133 każdej kategorii.
- 134 • Historia losowań w formie listy – każde losowanie jest wyświetlane w osobnym polu <p> z wyróż-
135 nionym numerem studenta.
- 136 • Przycisk aktywowany tylko przy dostępnych zadaniach – uniemożliwia losowanie, jeśli któraś z ka-
137 tegorii jest już pusta.

138 3.4. Print screeny aplikacji

139 W celu lepszego zobrazowania działania systemu, poniżej przedstawiono przykładowe zrzuty ekranu:

- 140 1. Ekran początkowy aplikacji – pokazuje nagłówek, liczby pozostałych zadań w każdej kategorii oraz
141 przycisk losowania. [Tutaj dodać zdj]
- 142 2. Historia losowań po kilku losowaniach – wyświetlone zostały wylosowane zestawy dla kilku studen-
143 tów. [Tutaj dodać zdj]
- 144 3. Przycisk losowania dezaktywowany – widok po wyczerpaniu list zadań, pokazujący, że system unie-
145 możliwia dalsze losowania. [Tutaj dodać zdj]

146 Print screeny umożliwiają wizualne przedstawienie działania aplikacji, potwierdzając poprawność im-
147 plementacji algorytmu losowania i funkcjonalności interfejsu użytkownika.

148 3.5. Wnioski

149 Na podstawie przeprowadzonych testów oraz analizy działania aplikacji można stwierdzić, że system
150 spełnia swoje podstawowe założenia funkcjonalne. Aplikacja poprawnie losuje zestawy zadań z dwóch
151 kategorii, zapewniając ich unikalność oraz umożliwiając użytkownikowi śledzenie pełnej historii przydzie-
152 lonych zestawów. Interfejs użytkownika jest czytelny i intuicyjny, co pozwala na wygodne korzystanie z
153 systemu bez konieczności posiadania wiedzy technicznej.

154 Ważnym elementem implementacji jest wykorzystanie mechanizmu sesji, który umożliwia przecho-
155 wywanie bieżącego stanu aplikacji, w tym numeru studenta oraz list pozostałych zadań. Rozwiązanie to
156 okazuje się wystarczające w środowisku jednostanowiskowym i przy niewielkiej liczbie użytkowników.

157 Pomimo swojej prostoty system pokazuje, że nawet niewielkie narzędzie webowe może skutecznie
158 wspierać proces dydaktyczny, automatyzując czynności, które tradycyjnie wykonywano ręcznie. Eliminacja
159 powtarzalnych zadań oraz ułatwiona kontrola nad przebiegiem losowania to główne zalety opracowanego
160 rozwiązania.

161 3.6. Możliwości rozbudowy systemu

162 Aplikacja, pomimo poprawnego działania w obecnej formie, może zostać rozszerzona o dodatkowe
163 funkcjonalności, które zwiększyłyby jej użyteczność oraz skalowalność. Do najważniejszych możliwości
164 rozbudowy należą:

- 165 1. Zapis wyników do bazy danych Zamiast przechowywania wyników w sesji, możliwe byłoby zapisa-
166 nie historii losowań w relacyjnej bazie danych (np. MySQL). Pozwoliłoby to na trwałe archiwizowa-
167 nie danych, analizę wyników, filtrowanie i generowanie raportów.

2. Obsługa wielu użytkowników Aktualna wersja zakłada pracę jednego użytkownika na jednej instancji sesji. Rozbudowa o system logowania lub identyfikacji nauczycieli umożliwiłaby korzystanie z aplikacji w środowisku wielodostępnym.
3. Dynamiczne zarządzanie listą zadań System można rozbudować o formularz umożliwiający dodawanie nowych zadań, usuwanie istniejących lub zmianę kategorii zadań bez konieczności edycji kodu źródłowego.
4. Eksport wyników Możliwość eksportu historii losowań do pliku CSV lub PDF ułatwiłaby dokumentowanie pracy nauczyciela oraz przechowywanie wyników w innych systemach.
5. Rozbudowa interfejsu użytkownika Można zastosować framework CSS (np. Bootstrap) lub bibliotekę JavaScript, aby zwiększyć ergonomię, dodać animacje, licznik pozostałych zadań czy bardziej rozbudowane komunikaty dla użytkownika.
6. Wsparcie dla różnych grup i przedmiotów System można rozbudować tak, aby losowane zadania nie powtarzały się między różnymi grupami studentów, jednocześnie dopuszczając powtarzalność, jeśli studenci piszą w tym samym dniu lub tej samej godzinie. Dodatkowo możliwe jest wprowadzenie obsługi różnych przedmiotów, dla których tworzona byłaby osobna pula zadań, co zwiększyłoby elastyczność systemu i umożliwiłoby stosowanie go w różnych kursach.

Wprowadzenie powyższych usprawnień mogłoby znacząco zwiększyć możliwości aplikacji i dostosować ją do bardziej zaawansowanych lub komercyjnych potrzeb.

187 **Spis rysunków**

Spis tabel

188

190 **Streszczenie pracy**

191 **Przykładowa praca inżynierska wykonana na seminarium**

192 Streszczenie pracy w języku polskim.

193 **Abstract**

194

195 Abstract thesis ...

196

197

Załącznik nr 2 do Zarządzenia nr 228/2021 Rektora Uniwersytetu Rzeszowskiego z dnia 1 grudnia 2021 roku w sprawie ustalenia procedury antyplagiatowej w Uniwersytecie Rzeszowskim

198

OŚWIADCZENIE STUDENTA O SAMODZIELNOŚCI PRACY

199

..... Bar Dawid, Dobranowski Bartłomiej, Piotr Prezydent

200

Imię (imiona) i nazwisko studenta

201

202

Wydział Nauk Ścisłych i Technicznych

203

204

..... Informatyka

205

Nazwa kierunku

206

207

..... 12345,12345,12345

208

Numer albumu

209

1. Oświadczam, że moja praca dyplomowa pt.: Przykładowa praca inżynierska wykonana na seminarium

210

211

1) została przygotowana przeze mnie samodzielnie*,

212

2) nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2021 r., poz. 1062) oraz dóbr osobistych chronionych prawem cywilnym,

213

214

215

3) nie zawiera danych i informacji, które uzyskałem/am w sposób niedozwolony,

216

4) nie była podstawą nadania dyplomu uczelni wyższej ani mnie, ani innej osobie.

217

2. Jednocześnie wyrażam zgodę/ nie wyrażam zgody** na udostępnienie mojej pracy dyplomowej do celów naukowo-badawczych z poszanowaniem przepisów ustawy o prawie autorskim i prawach pokrewnych.

218

219

220

221

(miejscowość, data)

(czytelny podpis studenta)

222

* Uwzględniając merytoryczny wkład promotora pracy

223

* – niepotrzebne skreślić

224

225

Załącznik nr 3 do Zarządzenia nr 228/2021 Rektora Uniwersytetu Rzeszowskiego z dnia 1 grudnia 2021 roku w sprawie ustalenia procedury antyplagiatowej w Uniwersytecie Rzeszowskim

226

OŚWIADCZENIE STUDENTA O ZGODNOŚCI WERSJI PAPIEROWEJ I ELEKTRONICZNEJ PRACY

227

228

..... Bar Dawid, Dobranowski Bartłomiej, Piotr Prezydent

229

Imię (imiona) i nazwisko studenta

230

231

Wydział Nauk Ścisłych i Technicznych

232

233

..... Informatyka

234

Nazwa kierunku

235

236

..... 12345,12345,12345

237

Numer albumu

238

Oświadczam, że treść pracy zamieszczonej przeze mnie w Systemie Wirtualna Uczelnia i zatwierdzonej przez promotora, jest identyczna z wersją drukowaną oraz zawartą na nośniku elektronicznym.

239

240

241

(miejscowość, data)

(czytelny podpis studenta)