

UNIWERSYTET RZESZOWSKI
WYDZIAŁ NAUK ŚCISŁYCH I TECHNICZNYCH



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

Informatyka

Przykładowa praca inżynierska wykonana na seminariu

Praca inżynierska

Praca wykonana pod kierunkiem
dr Krzysztof Balicki

Rzeszów 2025

*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. Intefrejs użytkownika	10
3.4. Print screeny aplikacji	11
Bibliografia	12
Spis rysunków	13
Spis tabel.....	14
Spis listingów	15
Streszczenie pracy.....	16
Abstract.....	17
Oświadczenie studenta o samodzielności pracy	18
Oświadczenie studenta o zgodności wersji papierowej i elektronicznej pracy	19

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 kolokwiów 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.

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

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 tworzenie testów z automatycznym przydzielaniem zadań, monitorowaniem postępów studentów i generowaniem statystyk. Ich główną wadą jest koszt oraz konieczność instalacji i obsługi dedykowanego oprogramowania.

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

44 2. Opis technologii i algorytmów

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

48 2.1. Technologie użyte w systemie

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

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

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

63 2.2. Struktura danych

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

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

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

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

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

76 **2.3. Algorytm Losowania Zestawów**

77 Proces losowania opiera się na kilku krokach:

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

90 **2.4. Bezpieczeństwo i ograniczenia algorytmu**

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

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

101 3. Prezentacja wyników

102 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.

106 3.1. Opis aplikacji

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

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

117 3.2. Prezentacja wyników losowania

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

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

```
121 Osoba: [numer studenta]  
122 [zadanie Java]  
123 [zadanie Excel]
```

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

126 Dzięki temu każdy student otrzymuje unikalny zestaw zadań, a nauczyciel może w prosty sposób śledzić 127 przebieg losowań oraz sprawdzić, jakie zadania zostały już przydzielone.

128 3.3. Intefrejs użytkownika

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

- 130 • Minimalistyczny design – wszystkie elementy rozmieszczone w centralnym oknie o stałej szerokości 131 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 aktywowy tylek 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 – wyświetcone 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 **Bibliografia**

¹⁴⁹ **Spis rysunków**

150 **Spis tabel**

¹⁵¹ **Spis listingów**

¹⁵² **Streszczenie pracy**

¹⁵³ **Przykładowa praca inżynierska wykonana na seminarium**

¹⁵⁴ Streszczenie pracy w języku polskim.

¹⁵⁵ **Abstract**

156

¹⁵⁷ Abstract thesis ...

159 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

160 **OŚWIADCZENIE STUDENTA O SAMODZIELNOŚCI PRACY**

161 Bar Dawid, Dobranowski Bartłomiej, Piotr Prezydent

162 Imię (imiona) i nazwisko studenta

163

164 Wydział Nauk Ścisłych i Technicznych

165

166 Informatyka

167 Nazwa kierunku

168

169 12345,12345,12345

170 Numer albumu

171 1. Oświadczam, że moja praca dyplomowa pt.: Przykładowa praca inżynierska wykonana na semina-
172 rum

173 1) została przygotowana przeze mnie samodzielnie*,

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

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

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

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

182

183 _____ (miejscowość, data)

_____ (czytelny podpis studenta)

184 * Uwzględniając merytoryczny wkład promotora pracy

185 ** – niepotrzebne skreślić

¹⁸⁷ 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

¹⁸⁸ **OŚWIADCZENIE STUDENTA O ZGODNOŚCI WERSJI PAPIEROWEJ I ELEKTRONICZNEJ
PRACY**

¹⁸⁹

¹⁹⁰ Bar Dawid, Dobranowski Bartłomiej, Piotr Prezydent

¹⁹¹ Imię (imiona) i nazwisko studenta

¹⁹²

¹⁹³ Wydział Nauk Ścisłych i Technicznych

¹⁹⁴

¹⁹⁵ Informatyka

¹⁹⁶ Nazwa kierunku

¹⁹⁷

¹⁹⁸ 12345,12345,12345

¹⁹⁹ Numer albumu

²⁰⁰ Oświadczam, że treść pracy zamieszczonej przeze mnie w Systemie Wirtualna Uczelnia i zatwierdzonej
²⁰¹ przez promotora, jest identyczna z wersją drukowaną oraz zawartą na nośniku elektronicznym.

²⁰²

²⁰³ _____
(miejscowość, data)

(czytelny podpis studenta)