



**Silesian University  
of Technology**

# **Tabelkomistrz**

## **Dokumentacja projektu**

Mateusz Białecki, Alicja Chmielewska, Paweł Czyż,  
Kamila Gendasz

18 stycznia 2022

# Spis treści

<b>1</b>	<b>Wstęp, założenia projektu</b>	<b>2</b>
<b>2</b>	<b>Praca nad projektem</b>	<b>3</b>
2.1	Dobór narzędzi . . . . .	3
2.2	Wyszczególnienie zadań i przypisanie ról w projekcie . . . . .	4
<b>3</b>	<b>Efekt końcowy</b>	<b>4</b>
<b>4</b>	<b>Instrukcja obsługi</b>	<b>4</b>
<b>5</b>	<b>Wnioski</b>	<b>4</b>

# 1 Wstęp, założenia projektu

Tworzenie tabel w systemie LaTeX przez niektórych może być uznane za nieco nieintuicyjne, trudne w szybkim, sprawnym tworzeniu, żmudne przy formatowaniu. Gdy tabel takich nie tworzy się na co dzień łatwo zapomnieć skomplikowanej składni i zasad scalania poszczególnych komórek.

Nasz projekt „Tabelkomistrz” zakłada stworzenie programu pomocnego przy tworzeniu tabel w LaTeX. Użytkownik, za pomocą przyjaznego środowiska graficznego, ma być w stanie w łatwy, szybki i przyjemny sposób stworzyć tabelę w trybie WYSIWYG, której LaTeX-owy kod zostanie mu zwrócony.

Program posiada zaimplementowane funkcjonalności takie jak:

- Tworzenie tabeli w trybie WYSIWYG
- Scalanie i rozdzielanie komórek
- Eksportowanie do kodu LaTeX
- Stylizowanie treści komórek (pogrubienie, justowanie itp.)
- Zmiana rozmiaru tabeli, ilości kolumn i wierszy

## 2 Praca nad projektem

### 2.1 Dobór narzędzi

Sam program stworzony został w języku python - jest to język, z którym większość zespołu miała styczność już wcześniej i znała jego podstawy, co pozwoliło do sprawnego przejścia do części właściwej projektu.

W projekcie wykorzystane zostały dwie dodatkowe biblioteki:

- Tkinter - biblioteka ułatwiająca tworzenie Graficznego Interfejsu Użytkownika.
- Pytest - biblioteka pomagająca w tworzeniu automatycznych testów jednostkowych i tworzeniu raportów na temat procentowego pokrycia kodu testami.

Poza tymi ściśle programistycznymi narzędziami, zespół wybrał kilka rozwiązań dla ułatwienia pracy zespołowej, organizacji pracy, spisywania godzin. Są to kolejno:

- GitHub - na tej platformie stworzone zostało repozytorium, które następnie każdy członek zespołu sklonował, a następnie umieszczane były na nim kolejne poprawki, testy, usprawnienia i nowe funkcjonalności programu.
- Microsoft Teams - zespół starał się spotykać regularnie co tydzień by omówić bieżące sprawy, problemy oraz zaproponować i przeprowadzić dyskusję i/lub głosowanie na temat rozwiązań.
- Arkusz kalkulacyjny - wykorzystywany do zapisywania godzin poświęconych na poszczególne zadania w projekcie przez każdego z członków. Umieszczony został w zespole na platformie Teams

## **2.2 Wyszczególnienie zadań i przypisanie ról w projekcie**

Na początku wspólnej pracy zespół dokonał rozpoznania zadań, które należałoby wykonać, aby dostarczyć działający i spełniający założenia końcowy produkt. Zadania te to między innymi:

1. Spełnienie wymagań wstępnych
  - Rozpoznanie sposobu tworzenia tabel w LaTeX
  - Dobór narzędzi i bibliotek
  - Nauka korzystania z wybranych bibliotek i narzędzi
2. Tworzenie oprogramowania
  - Stworzenie warstwy widoku
  - Stworzenie modelu
  - Stworzenie kontrolera
  - Napisanie modułu eksportującego tabelę do kodu LaTeX
  - Napisanie testów jednostkowych
  - Testowanie manualne
3. Spełnienie wymogów projektowych
  - Tworzenie prezentacji
  - Skomponowanie dokumentacji projektu
  - Nagranie filmu przedstawiającego działanie programu

## **3 Efekt końcowy**

## **4 Instrukcja obsługi**

## **5 Wnioski**

## Literatura