

Design proposal

“Narzędzie CLI do porównywania wersji dokumentów Typst”

Zespół 4 w składzie: Dominika Ferfecka, Sara Fojt, Małgorzata Kozłowska

1. Harmonogram

11.03 - 17.03	<ul style="list-style-type: none"><input type="checkbox"/> Przygotowanie design proposal<input type="checkbox"/> Spotkanie organizacyjne w celu zaplanowania harmonogramu<input type="checkbox"/> Wstępna analiza repozytorium narzędzia Typst oraz możliwych rozwiązań zadania<input type="checkbox"/> Utworzenie repozytorium projektu (Gitlab), oraz issue trackera (Jira)<input type="checkbox"/> Dołączenie do społeczności Typst (Discord)<input type="checkbox"/> Zaplanowanie konsultacji na najbliższy tydzień
18.03 - 24.03	<ul style="list-style-type: none"><input type="checkbox"/> Research pod kątem dwóch możliwych rozwiązań projektu:<ul style="list-style-type: none">o dodanie kolejnego kroku kompilacji (ekstrahowanie sekwencji kontentów po fazie ewaluacji, porównanie ich i przekazanie do faxy layout)o dokonanie restartu kompilacji po wcześniejszym wydobyciu zmian w kontentach<input type="checkbox"/> Konfiguracja:<ul style="list-style-type: none">o IDE z autoformatoremo wirtualnego środowiskao platformy do tworzenia dokumentacjio narzędzia do automatyzacji toxo narzędzia do budowania paczek<input type="checkbox"/> Ustalenie konwencji/standardu kodowania<input type="checkbox"/> Odbycie konsultacji z mentorem projektu
25.03 - 31.03	<ul style="list-style-type: none"><input type="checkbox"/> Wyekstrahowanie struktury kontentów z procesu kompilacji typsta
1.04 - 7.04	<ul style="list-style-type: none"><input type="checkbox"/> Zaimplementowanie struktur w Pythonie do edycji dokumentów formatu typst<input type="checkbox"/> Zaimplementowanie funkcjonalności porównywania sekwencji kontentów
8.04 - 14.04	<ul style="list-style-type: none"><input type="checkbox"/> Dodanie funkcjonalności oznaczania różnic między kontentami dwóch wersji dokumentów (za pomocą stylów)<input type="checkbox"/> Oddanie szkicu/dotychczasowej implementacji rozwiązania projektu
15.04 - 21.04	<ul style="list-style-type: none"><input type="checkbox"/> Konfiguracja budowy pakietu rozwiązania projektu

	<input type="checkbox"/> Research pod kątem kompilacji dokumentu z zaznaczonymi różnicami
22.04 - 28.04	<input type="checkbox"/> Zaimplementowanie kompilacji dokumentu z zaznaczonymi różnicami <input type="checkbox"/> Praca nad dokumentacją rozwiązania
29.04 - 5.05	<input type="checkbox"/> Stworzenie i praca nad instrukcją użytkownika <input type="checkbox"/> Implementacja testów rozwiązania <input type="checkbox"/> Dalsza praca nad dokumentacją
6.05 - 12.05	<input type="checkbox"/> Zakończenie implementacji wszystkich niezbędnych testów do obecnej wersji rozwiązania <input type="checkbox"/> Początek pracy nad refaktoryzacją kodu
13.05 - 19.05	<input type="checkbox"/> Dodanie funkcjonalności wyboru formatu pliku zwracanego przez proces kompilacji
20.05 - 26.05	<input type="checkbox"/> Dodanie funkcjonalności wyboru jaki typ zmian powinien zostać wyświetlony (dodany/usunięty kontent)
27.05 - 2.06	<input type="checkbox"/> Opcjonalnie: Customizacja kolorów zmian usuniętych i dodanych <input type="checkbox"/> Końcowa refaktoryzacja obecnego rozwiązania <input type="checkbox"/> Koniec prac nad dokumentacją

2. Bibliografia

- Python dokumentacja <https://docs.python.org/3/>
- Typst dokumentacja <https://typst.app/docs/>
- Rust dokumentacja <https://www.rust-lang.org/learn>
- Typst repo <https://github.com/orgs/typst/repositories>
 - Zwłaszcza plik z contentem
<https://github.com/typst/typst/blob/main/crates/typst/src/foundations/content.rs>
- Architektura Typst dokumentacja
<https://github.com/typst/typst/blob/main/docs/dev/architecture.md>
- Narzędzie latexdiff
[Using Latexdiff For Marking Changes To Tex Documents - Overleaf, Online LaTeX Editor](https://www.latexdiff.com/)
- MkDocs dokumentacja <https://www.mkdocs.org/>
- Dobre praktyki w korzystaniu z repozytorium [Conventional Commits](https://www.conventionalcommits.org/en/v1.0.0/)
- Zasady stylowania <https://peps.python.org/pep-0008/>
- Wersjonowanie semantyczne <https://peps.python.org/pep-0008/>
- Pathlib <https://docs.python.org/3/library/pathlib.html>
- Poetry <https://python-poetry.org/>
- Tox https://tox.wiki/en/4.14.1/user_guide.html

3. Planowany zakres eksperymentów

W celu przetestowania naszego narzędzia będziemy porównywać przygotowane wcześniej dokumenty Typst. Testy przeprowadzimy w sposób stopniowy, zwiększając złożoność dokumentów.

Przykładowe eksperymenty:

- Analiza dokumentów z różnymi typami zawartości:
 - tylko zwykły tekst (Markup)
 - zarówno tekst (Markup), jak i funkcje (Code)
 - zarówno tekst (Markup), jak i matematyczne wyrażenie (Math)
 - zarówno tekst (Markup), funkcje (Code) i matematyczne wyrażenia (Math)
- Analiza dokumentu o różnej liczbie linii:
 - maks 50 linii
 - 100 - 200 linii
 - ponad 300 linii
- Narzędzie zwraca dokument z zaznaczonymi:
 - tylko nowymi elementami
 - tylko usuniętymi elementami
 - zarówno usuniętymi i nowymi elementami

4. Planowana funkcjonalność programu

Narzędzie umożliwi porównywanie dwóch wersji dokumentów napisanych w formacie Typst wygenerowanych do postaci pdf.

Domyślnie program będzie zwracał przeanalizowany dokument pdf z zaznaczonymi zmianami:

- zmiany usunięte będą skreślane i zaznaczone na czerwono
- zmiany dodane będą zaznaczone na niebiesko

Dodatkowo program będzie umożliwiał podanie flagi pozwalającej wybrać dodatkową opcję:

- **--only_added** - zaznaczenie tylko zmian dodanych
- **--only_deleted** - zaznaczenie tylko zmian usuniętych
- **--color_added** - wybór koloru na który zaznaczane są zmiany dodane (funkcjonalność Nice To Have)
- **--color_deleted** - wybór koloru na który zaznaczane są zmiany usunięte (funkcjonalność Nice To Have)
- **--typst** - oprócz zwracanego pliku pdf, narzędzie zwróci również dokument w formacie typst, który po wygenerowaniu przez użytkownika utworzy plik w formacie pdf z zaznaczonymi zmianami - w ten sposób w razie potrzeby użytkownik będzie mógł zmodyfikować plik przed wygenerowaniem (funkcjonalność Nice To Have)

Wywołanie programu:

TypstDiff [ścieżka_do_pliku_typst_starego] [ścieżka_do_pliku_typst_nowego] >
[ścieżka_do_pliku_wyjściowego]

5. Planowany stack technologiczny

- Python - język programowania
 - biblioteki i moduły
 - argparse - parsowanie argumentów wywołania
 - pathlib - praca na plikach
- Rust - język programowania
- PyO3 - framework do integracji kodu w Pythonie i Rust
- Typst <https://github.com/typst/typst>
- Tox - testy automatyczne środowiska, paczek, systemów itp
- Pytest - framework do testowania kodu, łatwo konfigurowalny z Tox
- Visual Studio Code - środowisko programistyczne
- black - autoformatter
- flake8 - linter
- poetry - środowisko wirtualne, budowanie paczki, struktura projektu
- Gitlab - <https://gitlab-stud.elka.pw.edu.pl/dferfeck/zprp-typstdiff>
- MkDocs - dokumentacja oraz instrukcja użytkownika
- Discord - serwer Typst community do konsultacji oraz aktualnych informacjach o nowych wersjach Typst
- Jira - issue tracker