

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Факультет прикладної математики

Кафедра програмного забезпечення комп'ютерних систем

«До захисту допущено»  
Науковий керівник кафедри

\_\_\_\_\_ І.А. Дичка

(підпис)

“ \_\_\_\_\_ ” \_\_\_\_\_ 2017 р.

## Дипломний проект

на здобуття ступеня бакалавра

з напрямку підготовки 6.050103 “Програмна інженерія”

на тему ПРОГРАМНІ ЗАСОБИ ДЛЯ АВТОМАТИЧНОЇ ПОБУДОВИ  
ДОКУМЕНТАЦІЇ ДО REACT-КОМПОНЕНТІВ ІНТЕРФЕЙСУ  
КОРИСТУВАЧА

Виконав: студент 4 курсу, групи КП-32

Малявка Євгеній Анатолійович

\_\_\_\_\_ (підпис)

Керівник к.т.н., доцент Заболотня Т.М.

\_\_\_\_\_ (підпис)

Консультант з нормоконтролю старший викладач Онай М.В.

\_\_\_\_\_ (підпис)

Рецензент доц. каф. ММСА ІПСА, доц., к.т.н. Дідковська М.В.

\_\_\_\_\_ (підпис)

Засвідчую, що у цьому дипломному  
проекті немає запозичень з праць інших  
авторів без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

## АНОТАЦІЯ

Даний дипломний проект присвячений розробці програмних засобів для автоматичної побудови документації до React-компонентів інтерфейсу користувача.

Автоматична побудова документації базується на поєднанні функцій консольної утиліти та односторінкового веб-додатку. Синхронізація роботи програмних засобів забезпечується використанням інструментів безперервної інтеграції та функцій веб-сервісів на базі систем керування версіями. Завданнями CLI-утиліти є надання дружнього консольного інтерфейсу з описом доступних функцій, забезпечення можливості її параметризованого запуску з різною кількістю аргументів, форматоване відображення інформації про перебіг виконання програми, генерування даних на основі проекту React-компонента. Односторінковий додаток відповідає за структурування та відображення репрезентацій згенерованих CLI-утилітою ресурсів у вигляді документації.

Реалізація автоматичного генерування даних, розподілення даних та їх репрезентацій, підтримка версіонування документації з використанням функцій систем контролю версій, забезпечення ефективного відображення згенерованих даних, налаштування взаємодії програмних засобів та синхронізація їх роботи є тими цілями, шляхи досягнення яких описано в даному дипломному проекті.

## ABSTRACT

This diploma project deals with the development of the software tools for automated building of documentation for the React-UI components.

The automated building of the documentation is based on combining the functions of the CLI-tool and the single-page application. The synchronisation of both software tools is achieved by the usage of the continuous integration software and the web-services of the version control system. The aim of the CLI-tool is to provide user-friendly console interface with the description of the available commands, support the parametrised launch of the tool with diverse number of arguments, display formatted output during the process of execution and generate data based on the React.js project. Single-page application is responsible for structuring and rendering the representations of the generated data by the CLI-tool in the form of documentation.

The implementation of the automated process of the data being generated, separating data and its representations, support of the documentation versioning on the base of the SCM, effective representation of the generated data, adjustments in the interaction of the software tools and synchronising their functionality are the desired goals. The ways they have been achieved are described in this diploma project.

## СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. React Styleguidist [Електронний ресурс]. — Режим доступу :  
<https://github.com/styleguidist/react-styleguidist>
2. React Styleguide Generator [Електронний ресурс]. — Режим доступу :  
<https://github.com/pocotan001/react-styleguide-generator>
3. SourceJS [Електронний ресурс]. — Режим доступу :  
<https://github.com/sourcejs/Source>
4. React Styleguidist Integration Plugin for SourceJS [Електронний ресурс]. —  
Режим доступу : <https://github.com/sourcejs/sourcejs-react-styleguidist+>
5. React Styleguide [Електронний ресурс]. — Режим доступу :  
<https://github.com/jmfurlott/react-styleguide>
6. Python [Електронний ресурс]. — Режим доступу :  
<https://uk.wikipedia.org/wiki/Python>
7. GitPython [Електронний ресурс]. — Режим доступу :  
<http://gitpython.readthedocs.io/en/stable/intro.html>
8. Go [Електронний ресурс]. — Режим доступу :  
<https://uk.wikipedia.org/wiki/Go>
9. Go-git [Електронний ресурс]. — Режим доступу :  
<https://github.com/srcd/go-git>
10. Perl [Електронний ресурс]. — Режим доступу :  
<https://uk.wikipedia.org/wiki/Perl>
11. Git Repository [Електронний ресурс]. — Режим доступу :  
<http://search.cpan.org/~book/Git-Repository-1.320/lib/Git/Repository.pm>
12. JavaScript [Електронний ресурс]. — Режим доступу :  
<https://uk.wikipedia.org/wiki/JavaScript>
13. NodeGit [Електронний ресурс]. — Режим доступу :  
<https://github.com/nodegit/nodegit>
14. Libgit2 [Електронний ресурс]. — Режим доступу :  
<https://libgit2.github.com/>
15. JavaScript vs Python vs GO [Електронний ресурс]. — Режим доступу :  
<https://hackernoon.com/javascript-vs-python-vs-go-the-deal-breaker-e54328ffd550>

16. Choosing Frontend framework [Электронный ресурс]. — Режим доступа :  
<https://smashingboxes.com/blog/choosing-a-front-end-framework-angular-ember-react/>
17. TypeScript [Электронный ресурс]. — Режим доступа :  
<https://uk.wikipedia.org/wiki/TypeScript>
18. WS-Tool [Электронный ресурс]. — Режим доступа :  
<https://github.com/Mercateo/ws>
19. Dive into MobX and React [Электронный ресурс]. — Режим доступа :  
<https://mobx.js.org/getting-started.html>