МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ
КАФЕДРА ІНФОРМАТИКИ ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Звіт
до практичної роботи 3
«Аналіз функціональних точок»
з дисципліни «Економіка ІТ-індустрії та підприємництво»

Викладач:
професор кафедри ІПІ
Сидоров Микола Олександрович

Виконавець:
студент групи ІП-91
Кочев Геннадій Геннадійович
залікова книжка № ІП-9113

Прийняв:
старший викладач кафедри ІПІ
Родіонов Павло Юрійович

Київ - 2022

ЗМІСТ

ЗАВДАННЯ

Розробити модель для аналізу функціональних точок з переліком і докладним описом всіх внутрішніх логічних (ILF) і зовнішніх інтерфейсних (EIF) файлів, а також всіх транзакцій (EI, EO, EQ).

Виконати оцінки кількості RET і DET для внутрішніх логічних (ILF) і зовнішніх інтерфейсних (EIF) файлів, а також оцінки кількості FTR і DET для зовнішніх вводів (EI), виведнь (EO) і запитів (EQ).

Провести аналіз ступеня впливу основних характеристик системи.

Виконати розрахунки по моделі функціональних точок.

Всі розрахунки внести в робочий лист, що наведений в додатку 1.

Скласти звіт про виконану практичну роботу.

Захистити практичну роботу.

ОПИС ПРОГРАМИ, ДЛЯ ЯКОЇ ПРОВОДЯТЬСЯ РОЗРАХУНКИ

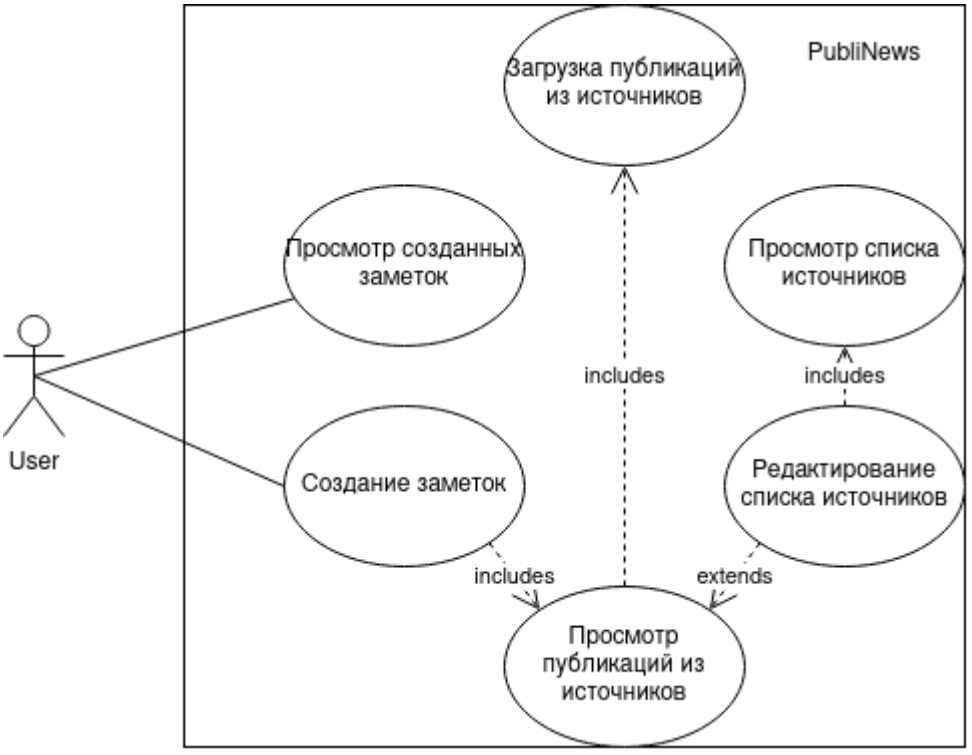
Програма, яку використано для підрахунку функціональних точок, є серверною частиною фулстек веб-застосунку для збереження та агрегації новин. Бекенд написаний мовою JavaScript на базі фреймворку Express.js та відповідає параметрам REST API, використовує ORM Knex.js для запитів до бази даних PostgreSQL, а також робить запити до зовнішніх API соціальних мереж, таких як Facebook, Twitter, Tumblr.

Готовий REST API дає можливість створювати джерела для новин - посилання на авторів контенту із вказаних вище соціальних мереж, стягувати останні пости від цих авторів, фільтрувати їх, та створювати нотатки з отриманих нових постів, зберігати їх у базі даних.

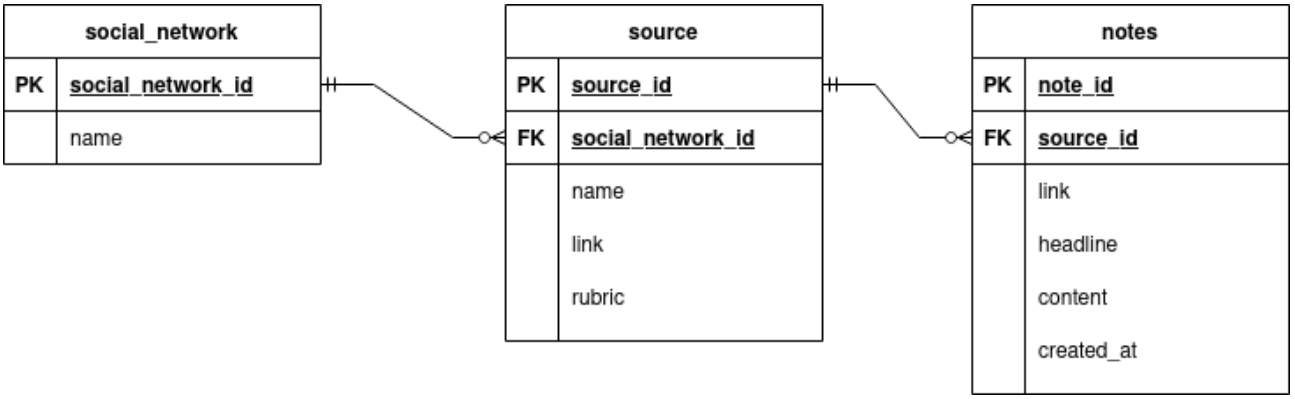
Посилання на вихідний код:

https://github.com/web-nodejs-kpi/publinews-backend

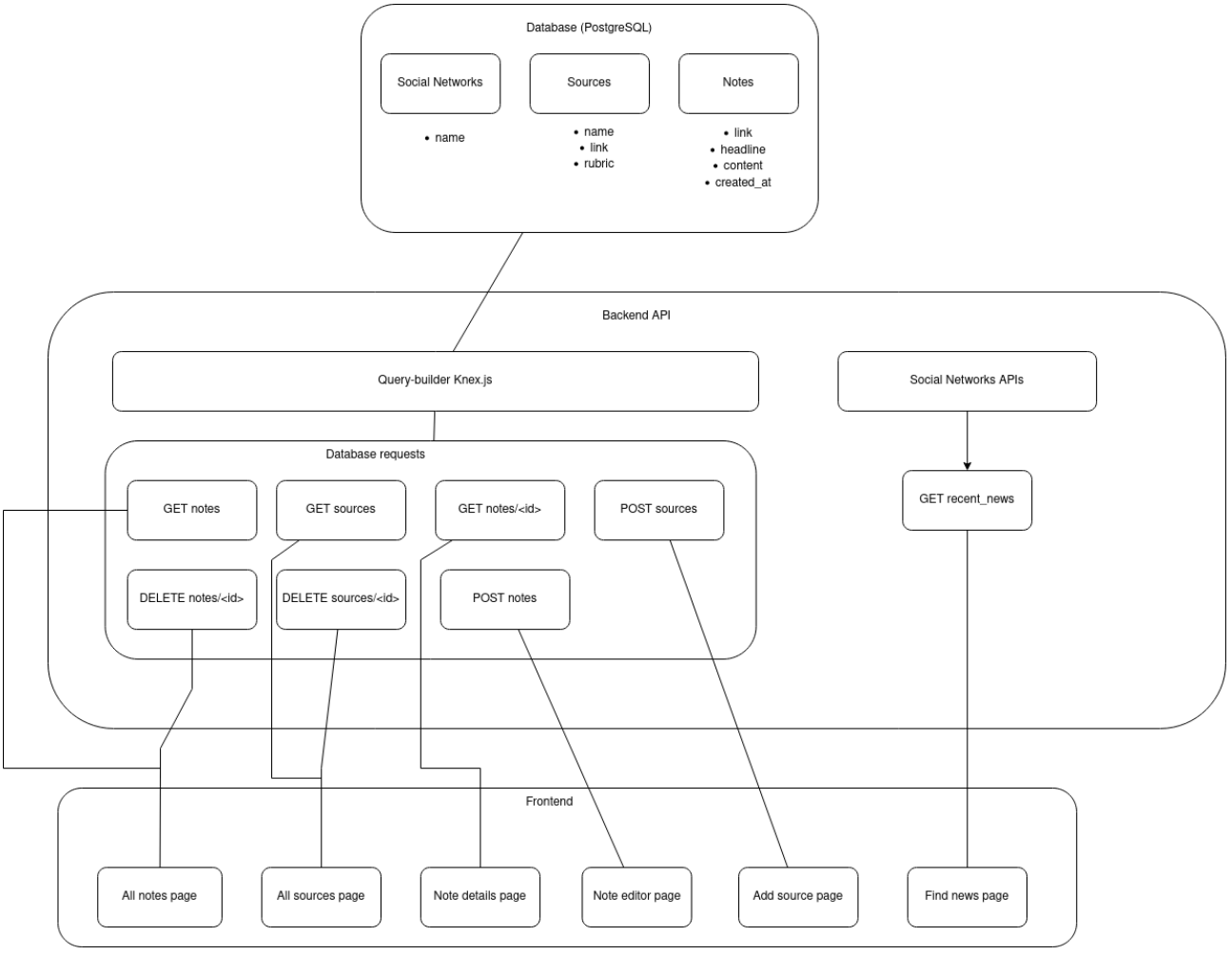
Діаграма Use Cases



Структура Бази Даних



Діаграма компонентів застосунку



РОБОЧИЙ ЛИСТ АНАЛІЗУ ЗА МЕТОДОМ ФУНКЦІОНАЛЬНИХ ТОЧОК

Визначення типу оцінки	Оцінка готового програмного продукту			
Визначення границі продукту	*Дані, що створюються продуктом, підтримуються ним у повній мірі, застосунок використовує зовнішні ресурси - API Facebook, Twitter, Tumblr для отримання нових даних для користувача, застосунок працює разом із окремо запущеною реляційною БД PostgreSQL.*			
Підрахунок кількість функцій в кожній категорії				
	Застосування вагових коефіцієнтів складності			
	Простий	Середній	Складний	Функціональні точки
Кількість вводів	1*3	_*4	_*6	3
Кількість виводів	1*4	_*5	_*7	4
Кількість запитів	_*3	_*4	_*6	0
Кількість файлів	2*7	_*10	_*15	14
Кількість інтерфейсів	_*5	_*7	_*10	0
Обчислення ненормованої кількості функціональних точок **UFPC**				**3+4+0+14+0 = 21**
Основні характеристики системи (GSC)				

Фактори середовища	Рейтинг (0,1,2,3,4,5) **з поясненням вибору**
Обмін даними	4 Програма являє REST API - використання HTTP-протоколу, робота з зовнішніми API соцмереж додає складності в обміні даних
Розподілена обробка даних	0 Бекенд представлений монолітною архітектурою - нема розбиття на мікросервіси

Вимоги до продуктивності	3 Пошук нових постів за соцмережами потребує деякої швидкості, створення нових нотаток, а також їх фільтрація та сортування відбувається тільки для одного користувача, тому середньої продуктивності цілком достатньо.
Обмеження по апаратним ресурсам	0 Обмежень ресурсів немає, для роботи потрібно запустити сервер БД та сервер бекенду, обидва сервери легкі для запуску та по ресурсах.
Транзакційне навантаження	0 Застосунок працює для одного конкретного користувача, тому виконати ефективно DDOS-атаку не вийде, навантаження транзакціями практично неможливе.
Інтенсивність взаємодії з користувачем	5 З усіма запитами REST API користувач може взаємодіяти через веб-інтерфейс або спеціальні утиліти (cURL, Postman, httpie)
Ергономіка(Ефективність роботи кінцевих користувачів)	0 Без запитів від користувача створене ПЗ не може відобразити свою функціональність.
Інтенсивність зміни даних(ILF) користувачами	2 Зміни даних відбувається у PostgreSQL при додаванні нових постів, нотаток, або їх редакції та видаленні.
Складність обробки	0 Додаток доволі простий, без аутентифікації, розрахований на одного користувача, що його запускає.
Повторне використання	4 За додавання нових нотаток корситувач може заново запустити додаток і уся введена інформація буде знов відтворена для

	подальшого використання.
Зручність інсталяції	2 Користувач повинен мати демон бази даних запущеним, а також мати встановленим Node.js та NPM.
Зручність адміністрування	3 Створення запитів можливе за допомогою будь-якого браузера або спеціальних утиліт для HTTP-запитів.
Використання декількох вузлів (портів)	0 Програма інсталюється на одному комп’ютері - комп’ютері кінцевого користувача
Гнучкість	3 Зміна даних проходить у реальному часі на очах у користувача, але працює лише на одному комп’ютері, дані не можна переносити, поки не реалізований відповідний функціонал.
Обчислення **TDI**	**4+0+3+0+0+5+0+2+0+4+2+3+0+3 = 26**

ЗАГАЛЬНІ ПІДСУМКИ ДЛЯ ПРОЕКТУ

Загальні розрахунки по проекту	
Обчислення нормуючого фактора **VAF**	**VAF = (TDI * 0.01) + 0.65 =** 26 * 0.01 + 0.65 = **0.91**
Обчислення нормованої кількості функціональних точок **AFPC**	**AFPC = UFPC * VAF =** 21 * 0.91 = **19.11**
Обчислення оцінки кількості рядків вихідного коду	**SLOC = AFPC * LM =** 19.11 * 31 = **592.41**
Обчислення **DFP**	**DFP = (UFP + CFP) * VAF =** (21 + 3) * 0.91 **= 21.84**
Обчислення **EFP**	**EFP = [(ADD + CHGA + CFP) * VAFA] + (DEL* VAFB) = 0**
Обчислення **AFP**	**AFP = [(UFPB + ADD + CHGA) - (CHGB + DEL)] * VAFA =** [(21+0+0) - (0+0)] * 0.91 **= 19.11**

ВИСНОВОК

У ході практичної роботи було проведено аналіз функціональних точок на прикладі написаного власноруч застосунку мовою JavaScript.

ER-діаграма застосунку показує три основні таблиці - соціальні мережі, джерела(автори постів), збережені нотатки. Усі три таблиці пов’язані первинними ключами. За методологією функціональних точок, у наявності 2 ILF. Перший має в собі 3 одиниці інформації про джерело постів (автора) і список відповідних джерел - отже, 3 DET та 1 RET. Другий має 4 одиниці інформації про нотатку та відповідний список нотаток - 4 DET та 1 RET. Обидва ILF належать до категорії ‘Low’ згідно з http://www.fredosaurus.com/notes-softeng/technology/fpa/fpa-data.html

Зіставимо метрику оцінювання кількості рядків коду з фактичним результатом. Мова проекту має LM коефіцієнт для маленьких проектів рівним 31. Результат оцінювання 592 рядки, насправді у програмі 529 рядків сумарно. Результат оцінювання доволі близький до реальних зусиль та об’ємів коду.

Роботу виконано успішно.