DBLab

Software Requirements Specification

(Специфікація Вимог до Програмного Забезпечення)

0.0.1

20.03.2025

Роман Геньбач

Business Analyst, Frontend Developer

Антон Шумейко

Project Manager, Lead Frontend Developer

Микола Галкін

Backend Developer, Database Engineer, DevOps

# Revision History (Історія Ревізій)

| **Дата** | **Опис** | **Автор** | **Коментарі** |
| --- | --- | --- | --- |
| 20.03.25 | Документ створено | Роман Геньбач |  |
| 10.04.25 | Документ перекладено і доповнено | Роман Геньбач |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval (Затвердження Документа)

Наступний документ Software Requirements Specification (Специфікація Вимог до Програмного Забезпечення) було прийнято і затверджено такими особами:

| **Підпис** | **Ім’я** | **Посада** | **Дата** |
| --- | --- | --- | --- |
|  | Оксана Мазурова | Project Leader |  |
|  |  |  |  |
|  |  |  |  |

**Зміст**

[**Revision History (Історія Ревізій)**](#_ot9cdi10l1l8) **3**

[**Document Approval (Затвердження Документа)**](#_kzzxzafxyild) **3**

[**1. Introduction (Вступ)**](#_1fob9te) **1**

[1.1 Purpose (Призначення)](#_8wbam712mutw) 1

[1.2 Scope (Рамки)](#_zh2lunze7xtq) 1

[1.3 Definitions, Acronyms, and Abbreviations (Визначення, Акроніми і Абревіатури)](#_d8dkxcfl5fkb) 1

[1.4 References (Посилання)](#_ds3b0jsihbco) 1

[1.5 Overview (Огляд)](#_o3013sctnyh) 1

[**2. General Description (Загальний Опис)**](#_6xv914ev7v1y) **2**

[2.1 Product Perspective (Перспектива Продукту)](#_5n1v5er4fijk) 2

[2.2 Product Functions (Функції Продукту)](#_l3cwcgw8t365) 2

[2.3 User Characteristics (Характеристики Користувачів)](#_xwjm4ov4n055) 2

[2.4 General Constraints (Загальні Обмеження)](#_a4xs1izfoytd) 2

[2.5 Assumptions and Dependencies (Припущення та Залежності)](#_fvg6dgdqq3wt) 2

[**3. Specific Requirements (Специфічні Вимоги)**](#_u7w50tkkagl) **3**

[3.1 External Interface Requirements (Вимоги до Зовнішніх Інтерфейсів)](#_hspegundmsmt) 3

[3.1.1 User Interfaces (Користувацькі інтерфейси)](#_44sinio) 3

[3.1.2 Hardware Interfaces (Апаратні інтерфейси)](#_2jxsxqh) 3

[3.1.3 Software Interfaces (Програмні Інтерфейси)](#_z337ya) 3

[3.1.4 Communications Interfaces (Інтерфейси Комунікації)](#_3j2qqm3) 3

[3.2 Functional Requirements (Функціональні вимоги)](#_1y810tw) 3

[3.2.1 Feature #01: Timetable Viewing (Перегляд Розкладу)](#_88jxclfrxyzt) 3

[3.2.2 Feature #02: Career Development Roadmap (Дорожня Мапа Кар’єрного Розвитку)](#_saffteavt90a) 4

[3.2.3 Feature #03: Timetable Altering (Редагування Розкладу)](#_s3c3o3og40i) 4

[3.2.4 Feature #04: Information Altering (Редагування Вмісту)](#_w5bt81fvjl55) 5

[3.3 Use Cases (Сценарії Використання)](#_iw3a38v9yhpc) 5

[3.3.1 Use Case #1: Timetable Viewing (Перегляд Розкладу)](#_3whwml4) 5

[3.3.2 Use Case #2: Timetable Altering (Зміна розкладу)](#_2bn6wsx) 5

[3.3.3 Use Case #3: Roadmap Viewing (Перегляд дорожньої мапи)](#_qsh70q) 6

[3.4 Classes / Objects (Класи / Об’єкти)](#_y1apjebs1s89) 6

[3.4.1 User (Користувач)](#_3as4poj) 6

[3.4.2 Teacher (Викладач)](#_4qunp7q1ij79) 6

[3.4.3 Discipline (Дисципліна)](#_3as4poj) 6

[3.4.4 Lesson (Заняття)](#_3as4poj) 6

[3.4.5 Event (Подія)](#_3as4poj) 7

[3.4.6 Material (Матеріал)](#_3as4poj) 7

[3.4.7 Development Direction (Напрямок Розвитку)](#_3as4poj) 7

[3.5 Non-Functional Requirements (Нефункціональні Вимоги)](#_1naq14rio291) 7

[3.5.1 Performance (Продуктивність)](#_c8t701whidg8) 7

[3.5.2 Reliability (Надійність)](#_147n2zr) 7

[3.5.3 Availability (Доступність)](#_3o7alnk) 7

[3.5.4 Security (Безпека)](#_23ckvvd) 7

[3.5.5 Maintainability (Підтримуваність)](#_ihv636) 7

[3.5.6 Portability (Портативність)](#_32hioqz) 8

[3.6 Inverse Requirements (Зворотні Вимоги)](#_1hmsyys) 8

[3.7 Design Constraints (Обмеження Дизайну)](#_ktuasxsu7fw5) 8

[3.8 Logical Database Requirements (Логічні Вимоги до Бази Даних)](#_ftu0gskanp2) 8

[3.9 Other Requirements (Інші Вимоги)](#_borhdr2vh50q) 8

[**4. Analysis Models (Моделі Аналізу)**](#_aznarwat8xl1) **9**

[4.1 BPMN Diagrams (BPMN-діаграми)](#_uwtjdhszwcje) 9

[4.2 User Flow Diagram (Діаграма Потоків Користувача)](#_4f1mdlm) 9

[4.3 Deployment Diagram (Діаграма Розгортання)](#_2u6wntf) 9

[**5. Change Management Process**](#_19c6y18) **9**

# 1. Introduction (Вступ)

## 1.1 Purpose (Призначення)

У цьому SRS описується веб-додаток DBLab для Лабораторії розробки баз даних Харківського національного університету радіоелектроніки.

## 1.2 Scope (Рамки)

Веб-додаток DBLab надасть корисну інформацію для студентів, які бажають вивчити створення, підтримку та оптимізацію баз даних. Крім того, це забезпечить зручний спосіб участі в роботі Лабораторії розробки баз даних, надаючи доступ до розкладу заходів і дозволяючи студентам обирати напрямок навчання та факультативні предмети, які вони хочуть вивчати.

## 1.3 Definitions, Acronyms, and Abbreviations (Визначення, Акроніми і Абревіатури)

ХНУРЕ – Харківський національний університет радіоелектроніки.

SRS – Software Requirements Specification (Специфікація Вимог до Програмного Забезпечення).

БД – база даних.

Лабораторія – Лабораторія розробки баз даних Харківського національного університету радіоелектроніки.

## 1.4 References (Посилання)

IEEE Guide to SRS (https://ieeexplore.ieee.org/document/278253)

## 1.5 Overview (Огляд)

Решта цього SRS-документу містить опис веб-додатку DBLab і структурована відповідно до Посібника IEEE з SRS (IEEE Guide to SRS).

# 2. General Description (Загальний Опис)

## 2.1 Product Perspective (Перспектива Продукту)

DBLab не є частиною жодної системи, однак містить посилання на опис предметів із сайту nure.ua. Усі користувачі, інформація та процеси реального життя, що стосуються DBLab, будуть міститися в екосистемі ХНУРЕ.

## 2.2 Product Functions (Функції Продукту)

DBLab надасть студентам інформацію про факультативні предмети та навички, які вони формують, керівників Лабораторії та запрошених експертів, а також розклад заходів Лабораторії.

Керівники Лабораторії матимуть можливість оновлення розкладу та зазначеної інформації.

## 2.3 User Characteristics (Характеристики Користувачів)

Усі користувачі DBLab будуть походити з академічного середовища ХНУРЕ: студенти та викладачі пов’язаних з БД дисциплін.

## 2.4 General Constraints (Загальні Обмеження)

Перша робоча версія програмного продукту з базовим функціоналом має бути представлена ​​до 30.05.2025.

Зустрічі з керівником проєкту необхідно проводити кожні 1-2 тижні.

Потрібно використовувати Git для керування версіями і Github, щоб зробити роботу видимою для всіх розробників і керівника проєкту.

Веб-додаток має відповідати бренду ХНУРЕ та використовувати його фірмові кольори.

Для проєктування дизайну необхідно використовувати Figma.

## 2.5 Assumptions and Dependencies (Припущення та Залежності)

Припущення:

– Студенти та керівники Лабораторії матимуть доступ до Інтернету та сучасний веб-браузер для використання веб-додатку DBLab;

– Користувачі матимуть базові знання про бази даних і веб-додатки для ефективної навігації системою;

– Керівники Лабораторії будуть активно оновлювати та підтримувати розклад та інформацію про предмети;

– Веб-сайт ХНУРЕ (nure.ua) залишатиметься доступним і надаватиме дійсні посилання на опис предметів;

– Команда проєкту матиме достатньо часу та ресурсів, щоб завершити розробку до встановленого терміну.

Залежності:

– GitHub використовуватиметься для контролю версій, тож всі учасники повинні мати облікові записи та права доступу;

– Figma буде використовуватися для проєктування дизайну, вимагаючи доступу всіх дизайнерів і зацікавлених сторін;

– Процес розробки залежить від регулярних зустрічей з керівником проєкту для забезпечення прогресу та узгодження з вимогами;

– Розміщення та розгортання веб-додатку залежатиме від наявності відповідного сервера або хмарної платформи.

# 3. Specific Requirements (Специфічні Вимоги)

## 3.1 External Interface Requirements (Вимоги до Зовнішніх Інтерфейсів)

### 3.1.1 User Interfaces (Користувацькі інтерфейси)

Веб-додаток DBLab забезпечить зручний та інтуїтивно зрозумілий інтерфейс як для керівників Лабораторії, так і для студентів, забезпечуючи легкий доступ до всіх функцій.

### 3.1.2 Hardware Interfaces (Апаратні інтерфейси)

Серверна частина (сервер університету або хмарний сервіс) повинна мати обчислювальну потужність, що дорівнює потужності принаймні двоядерного процесора, принаймні 4ГБ оперативної пам’яті і 40ГБ постійної пам’яті. Також необхідне стабільне підключення до інтернету.

Клієнтська частина працюватиме у браузері і не вимагає прямої взаємодії з апаратним забезпеченням.

### 3.1.3 Software Interfaces (Програмні Інтерфейси)

Серверна частина буде взаємодіяти з СУБД (системою управління базою даних) MySQL через ORM-засіб sequelize.

Клієнтська частина буде комунікувати з серверною частиною за допомогою протоколу HTTPS з використанням формату JSON. Веб-браузер буде інтерпретувати HTML/CSS/JS код клієнтської частини.

### 3.1.4 Communications Interfaces (Інтерфейси Комунікації)

Система буде комунікувати за допомогою стандартних веб-технологій і протоколів, зокрема:

– HTTPS: забезпечує безпечний зв’язок між користувачами та веб-додатком;

– REST API: використовується для отримання та оновлення даних, наприклад оновлення розкладу;

## 3.2 Functional Requirements (Функціональні вимоги)

### 3.2.1 Feature #01: Timetable Viewing (Перегляд Розкладу)

3.2.1.1 Introduction (Опис)

Ця функція дозволяє користувачам переглядати розклад із запланованими подіями, пов’язаними з Лабораторією. Таблиця розкладу забезпечує організований перегляд розкладу лекцій, практичних занять та інших важливих подій. Це допомагає студентам і викладачам ефективно розподіляти свій час і бути в курсі майбутніх заходів.

3.2.1.2 Inputs (Вхідні дані)

Хронологічні рамки: користувач обирає тиждень для перегляду.

3.2.1.3 Processing (Обробка)

Система враховує межі дат і отримує інформацію про події, що відбулися в діапазоні. Інформація береться або з бази даних, або з кеш-файлів на веб-сервері.

3.2.1.4 Outputs (Вихідні дані)

Структурований розклад з інформацією про відповідні події, зокрема:

– Назва події

– Дата і час проведення

– Тип події

– Формат (онлайн/офлайн)

– Розташування/посилання

– Додаткові посилання та/або файли

3.2.1.5 Error Handling (Обробка Помилок)

Події не знайдено: якщо жодна подія не відповідає фільтрам, відображається повідомлення, що пропонує ширші критерії пошуку.

Помилка підключення: якщо системі не вдається отримати дані, відображається повідомлення про помилку, і користувачеві пропонується повторити спробу пізніше.

### 3.2.2 Feature #02: Career Development Roadmap (Дорожня Мапа Кар’єрного Розвитку)

3.2.2.1 Introduction (Опис)

Ця функція надає студентам можливість переглядати дорожню мапу кар’єрного розвитку. Вона демонструє послідовність здобуття навичок і вмінь, які сприяють досягненню визначених кар’єрних цілей.

3.2.2.2 Inputs (Вхідні дані)

Обраний студентом напрямок кар’єрного розвитку.

3.2.2.3 Processing (Обробка)

Система отримує дані для дорожньої мапи і відображає їх користувачеві у зручній формі.

3.2.2.4 Outputs (Вихідні дані)

Візуалізована дорожня мапа, розділена на рівні, що включає групи навичок, що містять окремі навички, за деякими з яких можна побачити дисципліни, на яких можна отримати дану навичку.

3.2.2.5 Error Handling (Обробка помилок)

Проблема завантаження даних: повідомлення з пропозицією спробувати пізніше.

### 3.2.3 Feature #03: Timetable Altering (Редагування Розкладу)

3.2.3.1 Introduction (Опис)

Функція надає керівникам гуртка можливість змінювати або доповнювати розклад заходів Лабораторії. Це дозволяє підтримувати актуальність інформації про події, заняття, консультації тощо.

3.2.3.2 Inputs (Вхідні дані)

Дані події: назва, дата і час, тип, формат, посилання або місце проведення, додаткові матеріали.

Дія: додати / редагувати / видалити.

3.2.3.3 Processing (Обробка)

Система перевіряє авторизацію користувача, валідність введених даних і оновлює відповідні записи в базі даних.

3.2.3.4 Outputs (Вихідні дані)

Підтвердження успішного додавання/редагування/видалення.

3.2.3.5 Error Handling (Обробка помилок)

Недостатньо прав: повідомлення про відсутність доступу.

Невалідні дані: повідомлення з описом помилки.

Проблеми з базою даних: загальна помилка і пропозиція спробувати пізніше.

### 3.2.4 Feature #04: Information Altering (Редагування Вмісту)

3.2.4.1 Introduction (Опис)

Функція дозволяє адміністраторам змінювати текстовий та мультимедійний вміст сайту Лабораторії: дані про викладачів, напрямки розвитку, дорожні мапи тощо.

3.2.4.2 Inputs (Вхідні дані)

Керівник обирає, який саме вміст хоче змінити, і вводить нові дані.

3.2.4.3 Processing (Обробка)

Система перевіряє права користувача і зберігає оновлення до бази даних або файлової системи.

3.2.4.4 Outputs (Вихідні дані)

Повідомлення про успішне оновлення вмісту.

3.2.4.5 Error Handling (Обробка помилок)

Відсутність прав редагування: відмова у доступі.

Технічна помилка збереження: повідомлення з проханням спробувати пізніше.

## 3.3 Use Cases (Сценарії Використання)

### 3.3.1 Use Case #1: Timetable Viewing (Перегляд Розкладу)

Користувач переходить у розділ «Розклад». Він обирає тиждень, який хоче переглянути, або залишає поточний. Система отримує дані від серверу і формує таблицю з подіями, запланованими на обраний період. Користувач бачить назви подій, їхній тип, формат проведення (онлайн чи офлайн), розташування або посилання, а також додаткові матеріали, якщо вони є. Якщо на вибраний тиждень подій немає, система показує відповідне повідомлення.

### 3.3.2 Use Case #2: Timetable Altering (Зміна розкладу)

Керівник Лабораторії проходить автентифікацію і переходить у розділ редагування розкладу в застосунку для адміністрування. Він обирає дату, створює або редагує подію: вводить її назву, тип, формат, місце проведення/посилання, а також додає файли за потреби. Після цього натискає кнопку «Зберегти». Система зберігає подію в базу даних і оновлює інтерфейс користувача. У випадку помилки система повідомляє користувача про проблему.

### 3.3.3 Use Case #3: Roadmap Viewing (Перегляд дорожньої мапи)

Користувач зацікавлений розвиватися як професіонал у певному напрямку. Він обирає напрямок розвитку на головній сторінці. Він переглядає дорожню мапу, розгортає секції з групами навичок. Він обирає навичку і потрапляє на сторінку з дисциплінами для перегляду дисциплін, де можна засвоїти обрану навичку. Він обирає дисципліну і потрапляє на сторінку цієї дисципліни, звідки може перейти на силабус дисципліни, що знаходиться на сайті nure.ua. На кожному кроці користувач може вийти з системи без наслідків. У випадку помилок отримання даних система повідомить користувача про них.

## 3.4 Classes / Objects (Класи / Об’єкти)

### 3.4.1 User (Користувач)

3.4.1.1 Attributes (Атрибути)

nickname – публічне ім’я користувача.

email – електронна пошта, унікальна.

login – логін для входу, унікальний.

password – хешований пароль.

role – роль користувача в системі.

### 3.4.2 Teacher (Викладач)

3.4.1.1 Attributes (Атрибути)

full\_name – повне ім’я викладача.

place\_of\_Employment – місце роботи.

position – посада.

text – коротка біографія або опис.

level – академічний рівень.

teacher\_role – роль викладача.

photo – зображення (BLOB).

### 3.4.3 Discipline (Дисципліна)

3.4.3.1 Attributes (Атрибути)

discipline\_name – назва.

discipline\_Description – короткий опис.

discipline\_type – тип дисципліни.

volume – обсяг у кредитах ЄКТС.

syllabus\_link – посилання на силабус (nure.ua).

### 3.4.4 Lesson (Заняття)

3.4.4.1 Attributes (Атрибути)

name – назва заняття.

lesson\_date – дата проведення заняття.

lesson\_time – час проведення заняття.

link – посилання на онлайн-заняття.

### 3.4.5 Event (Подія)

3.4.5.1 Attributes (Атрибути)

event\_name – назва події.

type – тип події.

format – формат проведення (онлайн/офлайн).

begin\_date – дата початку події.

status – поточний статус події.

### 3.4.6 Material (Матеріал)

3.4.6.1 Attributes (Атрибути)

material\_name – назва матеріалу.

file – вкладений файл (BLOB).

material\_type – тип матеріалу.

### 3.4.7 Development Direction (Напрямок Розвитку)

3.4.7.1 Attributes (Атрибути)

development\_direction\_name – назва напрямку.

development\_direction\_Description – короткий опис напрямку розвитку.

## 3.5 Non-Functional Requirements (Нефункціональні Вимоги)

### 3.5.1 Performance (Продуктивність)

– 95% запитів користувачів до розкладу або дорожньої мапи повинні оброблятись менш ніж за 1 секунду.

– Завантаження вебінтерфейсу на клієнтській стороні повинно відбуватись менш ніж за 2 секунди при нормальному навантаженні.

### 3.5.2 Reliability (Надійність)

– Система повинна мати коефіцієнт успішної обробки запитів не менше 99.5%.

– Середній час між відмовами (MTBF) повинен перевищувати 30 днів.

### 3.5.3 Availability (Доступність)

– Система повинна бути доступна щонайменше 99% часу впродовж місяця.

– Максимальний допустимий час простою — не більше 10 хвилин на добу.

### 3.5.4 Security (Безпека)

– Усі з’єднання мають відбуватись через HTTPS.

– Доступ до функцій редагування обмежений авторизованими користувачами з відповідною роллю.

– Паролі зберігаються в хешованому вигляді за допомогою сучасного алгоритму.

### 3.5.5 Maintainability (Підтримуваність)

– Код повинен бути структурованим і модульним для полегшення підтримки.

– Система має дозволяти оновлення окремих компонентів без повної зупинки.

### 3.5.6 Portability (Портативність)

– Система має підтримувати розгортання на Linux-серверах та бути сумісною з Docker.

– Клієнтська частина має коректно відображатись у сучасних браузерах.

## 3.6 Inverse Requirements (Зворотні Вимоги)

– Система не повинна надавати доступ до редагування розкладу неавторизованим користувачам.

– Інтерфейс не повинен вимагати від користувача встановлення сторонніх плагінів або додатків.

– Чутливі дані користувача не повинні зберігатись у незашифрованому вигляді.

## 3.7 Design Constraints (Обмеження Дизайну)

– Застосунок має використовувати архітектуру клієнт-сервер з REST API.

– Інтерфейс користувача має бути реалізований за допомогою сучасного JavaScript-фреймворку (React).

## 3.8 Logical Database Requirements (Логічні Вимоги до Бази Даних)

– База даних повинна підтримувати зберігання структурованої інформації про розклад, події, навички, дисципліни, користувачів тощо.

– Всі зовнішні ключі мають бути цілісними для підтримки зв’язків між таблицями.

## 3.9 Other Requirements (Інші Вимоги)

# 4. Analysis Models (Моделі Аналізу)

Усі перераховані діаграми знаходяться за посиланням: <https://drive.google.com/file/d/1juXr84MdCtj-zjB505joB5UUlziDWKrP/view>.

## 4.1 BPMN Diagrams (BPMN-діаграми)

Було створено BPMN-діаграми для процесів “Планування і проведення подій” та “Робота студента з дорожньою мапою розвитку”.

## 4.2 User Flow Diagram (Діаграма Потоків Користувача)

Було створено Діаграму Потоків Користувача для візуалізації навігації сайтом.

## 4.3 Deployment Diagram (Діаграма Розгортання)

Було створено Діаграму Розгортання для чіткого розуміння фізичної будови системи і визначення необхідних характеристик пристроїв, з якими система буде взаємодіяти.

# 5. Change Management Process

Дані Вимоги підлягають постійному уточненню під час зустрічей команди з замовником і будуть змінюватися у процесі розробки продукту.