**Додаток 2**

**до тендерної документації**

**ІНФОРМАЦІЯ ПРО НЕОБХІДНІ ТЕХНІЧНІ, ЯКІСНІ ТА КІЛЬКІСНІ ХАРАКТЕРИСТИКИ ПРЕДМЕТА ЗАКУПІВЛІ, У ТОМУ ЧИСЛІ ВІДПОВІДНА**

**ТЕХНІЧНА СПЕЦИФІКАЦІЯ**

 **ТЕХНІЧНЕ ЗАВДАННЯ**

1. **Загальні відомості:**

Учасник надає послуги з розробки програмного забезпечення (блок програмних модулів для забезпечення освітнього процесу) для електронного сервісу «ХДУ24» за кодом ДК 021:2015: 48190000-6 Пакети освітнього програмного забезпечення, далі "Програми", вимоги до яких зазначено у Технічному завданні, при цьому виключні майнові права передаються користувачу (Замовнику) в повному обсязі та залишаються у нього.

1. **Опис предмету закупівлі:**

Розробка Програм включає в себе:

А - Створення окремого репозиторію Абітурієнт 24

В - Модуль вступна кампанія

С - Модуль цифрових відомостей і цифрових заліковок

D - Модуль Рейтингування

**А - Створення окремого репозиторію Абітурієнт 24**

Створення окремого репозиторію для сайту університету є важливим етапом у процесі розробки та підтримки безперебійної роботи сайту. Репозиторій повинен дозволяти зберігати весь код сайту та забезпечувати можливість співпраці над проектом з іншими розробниками. Наявність репозиторію повинна дозволяти зберігати код сайту у повному обсязі та забезпечувати можливість якісної співпраці з іншими розробниками проекту. Також репозиторій повинен зберігати всі версії сайту та повертатися до попередніх версій в разі потреби для забезпечення безпеки та стабільності сайту та зменшення ризику втрати даних чи зламу сайту. Створення окремого репозиторію повинно спрощувати процес співпраці зі сторонніми розробниками та агентствами, які можуть бути залучені до розробки та підтримки сайту та забезпечувати доступ до коду та контролювати роботу над проектом.

**Задачі**:

1. **Створення репозиторію на платформі GitHub**:

- Зареєструватися на GitHub, якщо потрібно

- Ввести назву репозиторію та опис, якщо потрібно

- Обрати тип доступу (приватний)

2. **Реалізація прав доступу для відповідних користувачів**:

- Запросити користувачів до репозиторію

- Надати права доступу до репозиторію (read або write)

- Встановити права доступу до окремих гілок, якщо потрібно

3. **Розгортання репозиторію на платформі серверу ХДУ 24**:

- Створити директорію для репозиторію на сервері

- Склонувати репозиторій з GitHub на сервер ХДУ 24

4. **Розгортання докер компонентів на платформі серверу ХДУ 24**:

- Створити Dockerfile для кожного компонента

- Створити docker-compose.yml файл для запуску компонентів

- Запустити докер контейнери за допомогою команди docker-compose up

5. **Розгортання ленс скрипту для забезпечення протоколів безпеки**:

- Встановити та налаштувати скрипт Lint для забезпечення стандартів кодування

- Додати скрипт в процес збірки проекту за допомогою CI/CD системи

6. **Клонування репозиторію і оновлення**:

- Клонувати репозиторій на локальну машину

- Оновлювати репозиторій щоденно, після відповідних комітів

7. **Розробка backend складової з відповідним API, розробка дизайну модулю, розробка frontend складової, яка взаємодіє з API**:

- Розробити REST API для взаємодії з даними в базі даних та з іншими компонентами

- Розробити backend складову з використанням популярного фреймворку, такого як Django, Flask, або Node.js

- Розробити дизайн модулю, що відповідає вимогам користувача та бізнес-логіки

- Розробити frontend складову з використанням популярного фреймворку, такого як React, Angular або Vue.js

- Взаємодія між frontend та backend складовими має бути здійснена за допомогою HTTP запитів до API.

**В - Модуль вступна кампанія**

Модуль вступної кампанії є важливим елементом для сайту університету, оскільки цей модуль повинен містити інформацію про умови вступу до різних факультетів, розклад вступних іспитів, список необхідних документів для подачі заявки, а також контактну інформацію для зв'язку з відповідальними особами університету.

Модуль вступної кампанії повинен містити функціонал для онлайн-заповнення заявки на вступ, яка міститься на сайті, для забезпечення можливості швидко та зручно подати свої дані, надати необхідні документи та зареєструватися на вступні іспити.

Додатково модуль вступної кампанії повинен мати функціонал для автоматичного відстеження статусу заявки, щоб студенти могли слідкувати за її станом та бути в курсі, коли будуть отримані результати вступної кампанії.

**Задачі**:

1. **Розробка backend складової модулю вступної кампанії відповідним API, розробка дизайну модулю, розробка frontend складової, яка взаємодіє з API**:

 - Розробити backend складову з відповідним API, яке надає доступ до функцій модуля вступної кампанії.

 - Розробити дизайн модулю відповідно до вимог дизайну ХДУ 24.

 - Розробити frontend складову, яка взаємодіє з API, і забезпечує можливість взаємодії з користувачем.

2. **Створення елементу "Роль" і функціонал особи в Абітурієнт**:

 - Розробити елемент "Роль" в системі, який визначає, що користувач є абітурієнтом.

 - Розробити функціональні можливості для абітурієнта в системі ХДУ 24.

3. **Реалізація відповідних полей, згідно з полями в системі ЄДЕБО**:

 - Список необхідних полів для абітурієнта та законного представника (особу вступника та законного представника (у разі, якщо вступник є неповнолітньою особою) щодо встановлення й фіксації повних реквізитів (ПІБ, дата народження, серія і номер паспорта, орган та дата видачі, РНОКПП/ІПН, місце реєстрації за відомостями з Єдиного державного демографічного реєстру або штампу в паспорті, номеру телефону та адреси особистої електронної пошти)) відповідно до вимог системи ЄДЕБО.

 - Розробити відповідні поля в системі ХДУ 24.

4. **Налаштування механізмів імпорту відповідних даних з ЄДЕБО в ХДУ 24**:

 - Налаштувати механізми імпорту даних в csv з вкладки Абітурієнти та Студенти з ЄДЕБО в систему ХДУ 24.

 - Визначити формати даних, які необхідно імпортувати, та перевірити їх на відповідність форматам, які підтримуються системою ХДУ 24.

 - Перевірити правильність заповнення полів та коректність збереження імпортованих даних.

5. **Реалізація можливості реєстрації засобами підтвердження введеної інформації з модуля ЄДЕБО для верифікації особи**:

 - Реалізувати можливість верифікації особи засобами, які підтверджують введену інформацію, такі як номер паспорту, тип паспорту, контактна інформація, контактний e-mail, ПІБ, дата народження.

 - Для цього можна використовувати інструменти для перевірки достовірності введеної інформації, які підтримуються системою ХДУ 24, або розробити власні механізми для перевірки інформації.

 - Необхідно забезпечити можливість реєстрації в системі ХДУ 24, зберігаючи інформацію про особу та засоби підтвердження пошти користувача.

6. **Додавання елементу підписання публічної угоди із вказаним набором даних, які потім можна буде імпортувати в систему ХДУ 24**:

- Розробка форми підписання публічної оферти з вказанням необхідних полів (наприклад: ПІБ, адреса тощо).

- Розробка механізму генерації PDF-документу з підписаною публічною угодою та вказаними даними.

- Додавання можливості завантаження та перегляду підписаних оферт у системі ХДУ 24.

7. **Розробка елементів скриптів, які необхідні для подальшого перенесення даних в ХДУ 24**:

- Розробка скрипту для експорту даних з системи ЄДЕБО.

- Розробка скрипту для імпорту даних з ЄДЕБО в систему ХДУ 24 з урахуванням відповідних полів та форматів даних.

- Налаштування автоматичного запуску скриптів для періодичного оновлення даних у системі ХДУ 24.

**С - Модуль цифрових відомостей і цифрових заліковок**

Модуль цифрових відомостей та цифрових заліковок є важливим компонентом сайту університету, оскільки він надає можливість студентам та викладачам взаємодіяти з інформацією про успішність студентів та їх академічну історію, електронний формат робить відомість та залікову книжку комфортними у використанні для всіх учасників освітнього процесу.

Цей модуль повинен дозволяти студентам переглядати свої оцінки, графіки занять, розклади і додаткові матеріали, пов'язані з їх курсами у будь-який час доби, що є значною перевагою електронного формату. Цей модуль повинен дозволяти викладачам перевіряти роботи студентів, виставляти оцінки та надавати додаткової інформації про курси, які вони ведуть у зручний для них час.

Модуль повинен забезпечувати збір та обробку інформації про академічну успішність студентів для аналізу ефективності курсів та для вдосконалення навчальних планів університету.

**Задачі**:

1. **Розробка backend складової**:

- Створення бази даних для зберігання інформації про студентів, курси, викладачів, оцінки тощо

- Розробка API для взаємодії з базою даних та забезпечення функцій модулю

- Забезпечення безпеки даних та обмеження доступу до конфіденційної інформації

2. **Розробка frontend складової**:

- Розробка інтерфейсу користувача для взаємодії з модулем

- Забезпечення взаємодії з API для здійснення операцій з цифровими відомостями та заліковками

3**. Створення цифрових відомостей**:

- Забезпечення можливості додавання інформації про курси та студентів до бази даних

- Забезпечення можливості заповнення цифрових відомостей студентами та викладачами

- Забезпечення можливості відображення цифрових відомостей в інтерфейсі користувача

4. **Надання прав певним користувачам для заповнення**:

- Забезпечення можливості створення користувачів з різними рівнями доступу

- Забезпечення можливості надання прав користувачам для заповнення цифрових відомостей

5. **Надання прав для підписання**:

- Забезпечення можливості надання прав на підписання цифрових відомостей викладачами та керівниками кафедр та факультетів

6. **Надання прав редагування**:

- Забезпечення можливості редагування цифрових відомостей користувачами з відповідними правами

7. **Надання прав для закриття відомостей**:

- Розробка механізму підтвердження прав на закриття відомостей для викладачів, завідуючих кафедрою та декана.

- Розробка функціоналу для забезпечення доступу до відомостей залежно від ролі користувача.

- Розробка алгоритму закриття відомостей, який передбачатиме підписання викладача, завідуючого кафедрою та декана.

- Розробка інтерфейсу для підписання відомостей користувачами з правами на закриття відомостей.

8. **Для створення елементу Заліковка: імпортування результатів із цифрових відомостей, які були заповнені**:

- Розробка механізму імпорту результатів з цифрових відомостей для створення елементу заліковки.

- Розробка функціоналу, який дозволяє зв'язати імпортовані результати заліковки з конкретним студентом.

- Розробка інтерфейсу для відображення результатів студентів в заліковці.

- Розробка функціоналу для підрахунку кількості кредитів та відображення загальної оцінки студента.

**Модуль Рейтингування**

Розробити модуль для системи управління освітою, який дозволить оцінювати та ранжувати науково-педагогічних працівників, кафедри та факультети на основі різних критеріїв.

**Задачі:**

1**. Рейтинг науково-педагогічного працівника:**

- Враховувати наукову діяльність (кількість публікацій, цитувань, грантів, конференцій тощо);

- Враховувати педагогічну діяльність ( дистанційні курси тощо);

- Враховувати адміністративну діяльність (керівництво кафедрою, факультетом, участь у комітетах тощо);

- Можливість налаштування ваги кожного критерію в загальному рейтингу;

- Автоматичне оновлення рейтингу на основі введених даних.

2. **Рейтинг кафедри:**

- Агрегування рейтингів науково-педагогічних працівників, що належать до кафедри;

- Враховувати загальну наукову активність кафедри (отримані гранти, публікації, конференції тощо);

- Враховувати педагогічну ефективність кафедри (задоволеність студентів, успішність випускників, залучення в проекти тощо);

- Можливість налаштування ваги кожного критерію в загальному рейтингу;

- Автоматичне оновлення рейтингу на основі введених даних.

3. **Рейтинг факультету:**

- Агрегування рейтингів кафедр, що належать до факультету;

- Враховувати загальну наукову активність факультету (отримані гранти, публікації, конференції тощо);

- Враховувати педагогічну ефективність факультету (задоволеність студентів, успішність випускників, залучення в проекти тощо);

- Враховувати роль факультету в міжнародному співробітництві та партнерствах;

- Можливість налаштування ваги кожного критерію в загальному рейтингу;

- Автоматичне оновлення рейтингу на основі введених даних.

4. **Додаткові функціональні вимоги:**

- Зручний інтерфейс для введення даних про діяльність науково-педагогічних працівників, кафедр та факультетів;

- Можливість порівняння рейтингів за різними критеріями (наприклад, наукова діяльність проти педагогічної діяльності);

- Можливість формування звітів за вказаний період (місяць, квартал, рік) з візуалізацією результатів;

- Можливість експорту рейтингів у формати CSV, Excel або PDF для зручності подальшої обробки та аналізу;

- Реалізація захисту даних та контролю доступу до інформації про рейтинги залежно від рівня користувача (адміністратор, викладач, студент).

5. **Технологічні вимоги:**

- Модуль повинен бути розроблений на основі сучасних веб-технологій з можливістю інтеграції з наявною системою управління освітою;

- Забезпечити сумісність з основними сучасними веб-браузерами (Google Chrome, Mozilla Firefox, Safari, Microsoft Edge тощо);

- Реалізувати адаптивний дизайн для зручного відображення інформації на різних пристроях (настільні комп'ютери, ноутбуки, планшети, смартфони);

- Забезпечити надійний та безпечний обмін даними між клієнтською та серверною частинами, використовуючи протокол HTTPS та відповідні методи шифрування;

- Продумати оптимальну структуру бази даних для зберігання та обробки інформації про рейтинги з урахуванням можливості масштабування та оптимізації.

6. **Тестування та впровадження:**

- Розробити план тестування модулю на різних етапах розробки, забезпечуючи функціональне, інтеграційне, навантаженче та безпекове тестування;

- Забезпечити створення тестових сценаріїв та використання автоматизованих тестів, де це можливо, для перевірки стабільності та надійності модулю;

- Розробити план впровадження модуля рейтингування в наявну систему управління освітою, передбачаючи створення резервних копій даних, тестування на тестовому середовищі та підготовку користувачів до використання нового модулю;

- Організувати підтримку користувачів після впровадження модуля, забезпечуючи швидке вирішення виниклих питань та проблем, а також збір зворотнього зв'язку для подальшого вдосконалення модуля.

 **Вимоги до знання компонентів та застосуванню їх під час реалізації**

 **технічного завдання:**

* python 3.11.3,
* alpine docker 3.17,
* PostgreSQL 11.15,
* Django 4.1.7,
* djangorestframework 3.14.0,
* graylog 5.0.2,
* mongo 5.0.3,
* elasticsearch-oss 7.10.2,
* nginx 1.19.0-alpine,
* certbot v2.4.0,
* Sphinx 4.2.0,
* JavaScript (version ES6),
* ReactJS (version 18.2.0),
* antd (version 5.4.2) - components library,
* react-redux (version 8.0.5),
* reduxjs/toolkit (version 1.9.3),
* tanstack/react-query (version 4.26.1)

**Для підтвердження відповідності тендерної пропозиції технічним, якісним та кількісним характеристикам до предмета закупівлі, учасник повинен надати:**

- довідку, в якій необхідно вказати інформацію про технічні, якісні, кількісні та інші характеристики комп’ютерних програм, майнові права на які буде надано у відповідності до вимог замовника, наведених у Технічному завданні.

*Примітки:*

*Будь-які посилання в технічних вимогах на конкретну торговельну марку або тип, передбачає надання еквіваленту (технічні вимоги еквіваленту не повинні бути гіршими).*

*У разі надання еквіваленту, Учасник в пропозиції повинен зазначити найменування та технічні характеристики запропонованого еквіваленту. Запропонований учасником еквівалент предмету закупівлі за своїми технічними характеристиками повинен бути не гіршим.*