РОБОТА ПЛАНУВАЛЬНКА ХНЕУ ІМЕНІ СЕМЕНА КУЗНЕЦЯ. ІНСТРУКЦІЯ ДЛЯ КОРИСТУВАЧА ІЗ РОЛЛЮ

«ГОЛОВА НАУКОВОГО СЕКТОРА»

МЕТА СТВОРЕННЯ ДОДАТКУ ТА СПИСОК ВИКОРИСТАНИХ ТЕХНОЛОГІЙ

Створення додатку «Планувальник ХНЄУ ім. С. Кузнеця» велося на основі web-технології ASP.NET MVC 5, з використанням технології для роботи з базою даних Entity Framework 6 Code First, через те, що веб-ресурси дуже універсальні і практичні в користуванні. Веб-додатки полегшують організацію зберігання даних. Для користування веб-додатками необхідні лише комп'ютер з браузером і з'єднання з Інтернет. Для оновлення веб-додатку, його необхідно оновити тільки на сервері і усі відразу зможуть працювати з новою версією. ASP.NET MVC пропонує швидку розробку та управління веб-додатками, дозволяє легко вирішувати рутинні завдання веб-програміста та має підтримку баз даних MS SQL. Таким чином розробка додатку «Планувальник ХНЄУ ім. С. Кузнеця» дозволить підвищити ефективність роботи та зменшити трудовитрати викладачів, та забезпечить вирішення задач модуля за допомогою без паперової технології.

ASP.NET MVC Framework — фреймворк для створення веб-застосунків, який реалізує шаблон Model-view-controller. Цей фреймворк доданий Microsoft в ASP.NET. Концепція шаблону MVC (model - view - controller) передбачає поділ додатка на три компоненти:

Контролер (controller) представляє клас, що забезпечує зв'язок між користувачем і системою, поданням і сховищем даних. Він отримує вводяться користувачем дані і обробляє їх. І в залежності від результатів обробки відправляє користувачеві певний висновок, наприклад, у вигляді подання.

Подання (view) - це власне візуальна частина або призначений для користувача інтерфейс програми. Як правило, html-сторінка, яку користувач бачить, зайшовши на сайт.

Модель (model) представляє клас, що описує логіку використовуваних даних.

Загальну схему взаємодії цих компонентів представлено на рис. 1.

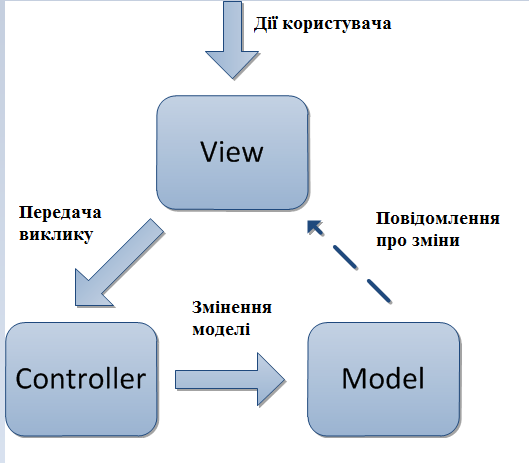


Рис. 1 Концепція шаблону MVC

У цій схемі модель є незалежним компонентом - будь-які зміни контролера або подання не зачіпають модель. Контролер і уявлення є відносно незалежними компонентами, і нерідко їх можна змінювати незалежно один від одного. Завдяки цьому реалізується концепція поділ відповідальності, в зв’язку з чим легше побудувати роботу над окремими компонентами. Крім того, внаслідок цього додаток має кращу тестованих. І якщо нам, припустимо, важлива візуальна частина або фронтенд, то ми можемо тестувати уявлення незалежно від контролера. Або ми можемо зосередитися на бекенда і тестувати контролер.

Для роботи з базою даних була використана технологія Entity Framework, яка являє спеціальну об'єктно-орієнтовану технологію на базі фреймворка .NET для роботи з даними. Якщо традиційні засоби ADO.NET дозволяють створювати підключення, команди та інші об'єкти для взаємодії з базами даних, то Entity Framework являє собою більш високий рівень абстракції, який дозволяє абстрагуватися від самої бази даних і працювати з даними незалежно від типу сховища. Якщо на фізичному рівні ми оперуємо таблицями, індексами, первинними і зовнішніми ключами, але на концептуальному рівні, який нам пропонує Entity Framework, ми вже працюємо з об'єктами.

Entity Framework передбачає три можливі способи взаємодії з базою даних:

Database first: Entity Framework створює набір класів, які відображають модель конкретної бази даних

Model first: спочатку розробник створює модель бази даних, по якій потім Entity Framework створює реальну базу даних на сервері.

Code first: розробник створює клас моделі даних, які будуть зберігатися в бд, а потім Entity Framework за цією моделлю генерує базу даних і її таблиці.

Проте, для розробки додатку «Планувальник ХНЄУ ім. С. Кузнеця» був використаний Code first, як спосіб взаємодії з базою даних.

Для створення користувальницького інтерфейсу, були використані наступні технології:

HTML — стандартна мова розмітки веб-сторінок в Інтернеті. Більшість веб-сторінок створюються за допомогою мови HTML. Документ HTML оброблюється браузером та відтворюється на екрані у звичному для людини вигляді.

Каскадні таблиці стилів (CSS) — спеціальна мова, що використовується для опису сторінок, написаних мовами розмітки даних. Найчастіше CSS використовують для візуальної презентації сторінок, написаних HTML.

JavaScript (JS) — динамічна, об'єктно-орієнтована мова програмування. Найчастіше використовується як частина браузера, що надає можливість коду на стороні клієнта (такому, що виконується на пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд веб-сторінки. Мова JavaScript також використовується для програмування на стороні сервера.

Для роботи сторони клієнта із базою даних, а саме для роботою за таблицями використовували jQuery - бібліотеку JavaScript, що фокусується на взаємодії JavaScript і HTML. Бібліотека jQuery допомагає легко отримувати доступ до будь-якого елементу DOM, звертатися до атрибутів і вмісту елементів DOM, маніпулювати ними.

При розробці валідації усіх форм, був використаний AngularJS є opensource JavaScript-фреймворк, який використовує шаблон MVC. Власне використання MVC є його однією з відмінних рис.

Для опису інтерфейсу використовується декларативне програмування, а бізнес-логіка відокремлена від коду інтерфейсу, що дозволяє поліпшити тестованого і розширюваність додатків.

Іншою відмінною рисою фреймворка є двостороннє зв'язування, що дозволяє динамічно змінювати дані в одному місці інтерфейсу при зміні даних моделі в іншому. Таким чином, AngularJS синхронізує модель і уявлення.

ГЛОСАРІЙ

Словник основних використовуваних термінів, у вигляді глосарію, наведений в табл. 1.

Таблиця 1

Глосарій

|  |  |
| --- | --- |
| Термін | Опис терміну |
| 1 | 2 |
| 1. Основні поняття та категорії предметної області та проекту | |
| Планувальник | Система, що розроблена як особистий кабінет викладачів університету |
| Кафедра | Підрозділ вищого навчального закладу, що проводить навчально – виховну, методичну і наукову діяльність. |
| Індивідуальний план | Основний документ викладача, на підставі якого оцінюється його робота, подається щомісячний рапорт у бухгалтерію про нарахування зарплати, ведеться перевірка КРУ |
| Розподілення навантаження | Основний документ навчальної частини, в якому відображений обсяг роботи у годинах за визначеними законом видами діяльності (навчальна, методична, наукова, організаційна), який дорівнює встановленої кількості годин. Навчальне навантаження обумовлює розмір заробітної плати, яку викладач отримає: частка ставки обчислюється поділом встановленої для однієї ставки норми навантаження на фактичне навантаження працівника. |
| Google Scholar | Вільна доступна пошукова система, яка індексує повний текст наукових публікацій всіх форматів і дисциплін |
| Open Researcher and Contributor ID (ORCID) | Некомерційний проект, мета якого створення єдиного, міжнародного реєстру вчених ORCID |
| Науково – дослідна робота (НДР) | Важливий засіб підвищення якості підготовки спеціалістів для народного господарства, здатних творчо застосовувати у практичній діяльності новітні досягнення науково – технічного прогресу |
| 2. Користувачі системи | |
| Викладач модератор | Користувач, котрий має повний доступ до всіх функцій сайту |
| Викладач | Користувач, котрий має доступ майже до всіх функцій сайту, крім адміністративних |
| Голова методичного відділу | Користувач, котрий має доступ до методичної роботи |
| Голова наукового сектора | Користувач, котрий має доступ до наукової роботи |
| Голова навчальної частини | Користувач, котрий має повний доступ до всіх функцій сайту |
| 3. Вхідні та вихідні документи | |
| Довідники індивідуального плану | Вхідний документ. Довідники, в яких вказані години роботи викладачів |

Закінчення табл. 1

|  |  |
| --- | --- |
| 1 | 2 |
| Відомості про викладачів | Вхідний документ. Інформація про дані викладачів |
| Файли публікацій | Вхідний документ. Файл публікації викладача |
| Плани розподілу годин | Вхідний документ. Плани із даними навантаження викладачів, для розподілення навантаження |
| Список кафедр | Вхідний документ. Список кафедр університету |
| Список груп | Вхідний документ. Список груп університету |
| Список факультетів | Вхідний документ. Список факультетів університету |
| Список предметів | Вхідний документ. Список предметів університету |
| Результати НДР студентів | Вхідний документ. Результати науково – дослідних робіт студентів |
| Публікації | Вихідний документ. Оформленні та завантажені до системи публікації викладачів |
| Звіт за кафедрою | Вихідний документ. Звіт за період по кафедрі |
| Звіт за півріччя | Вихідний документ. Звіт за півріччя по кафедрі |
| Індивідуальний план | Вихідний документ. Заповнений та сформований індивідуальний план |
| Розподілене навантаження | Вихідний документ. Сформоване та розподілене навантаження викладача |

Розгортання ТА функціонал програмного продукту

Розгортання додатку можливе на локальному та зовнішньому сервері.

Для розгортання додатку на локальному сервері необхідно:

частота процесора 2Ггц або більше;

ОЗП не менше 1Гб;

наявність підключення до мережі Internet;

IIS Internet Information Server;

Windows Server 2008 і вище;

SQL Server 2008 і вище;

наявність встановленої операційної системи Windows 7 та вище;

будь – який браузер, бажано Google Chrome.

Для розгортання додатку на зовнішньому сервері:

частота процесора 2Ггц або більше;

ОЗП не менше 2Гб;

наявність підключення до мережі Internet;

IIS Internet Information Server;

Windows Server 2008 і вище;

SQL Server 2008 і вище;

наявність встановленої операційної системи Windows 7 та вище;

будь – який браузер, бажано Google Chrome.

Для розгортання програмного продукту необхідно виконати запуск проекту у Microsoft Visual Studio та у запиті в браузері прописати URL локального хосту.

Це дозволяє динамічно формувати автоматично оновлювані веб – сторінки з боку веб – сервера. Для доступу до додатку на зовнішньому сервері необхідно у запиті в браузері ввести адресу зовнішнього хостингу.

При реалізації розроблюваного додатку всі функції були розподілені на два модулі.

Модуль перший:

авторизація користувача;

створення та редагування профілю викладача;

інтеграція з розкладом університету.

Модуль другий:

редагування даних викладачів;

формування індивідуального плану;

розподілення навантаження;

формування звітності (індивідуальний план викладача, звіт за кафедрою, звіт за півріччя);

створення публікацій;

формування результатів НДР студентів.

Розроблюваний сайт викладений в мережу Інтернет за назвою «Планувальник ХНЕУ ім. С.Кузнеця». Для переходу на сайт можна використати наступне посилання:

nmr.hneu.edu.ua – для входу поза університетом.

nmr.hneu.edu.ua:81 – для входу в університеті.

На головній сторінці сайту, на якій відображається логотип університету, головне меню сайту, новини щодо оновлення сайту й контактна інформація.

Поки користувач не увійшов в систему, йому буде доступна тільки сторінка авторизації (рис. 2).

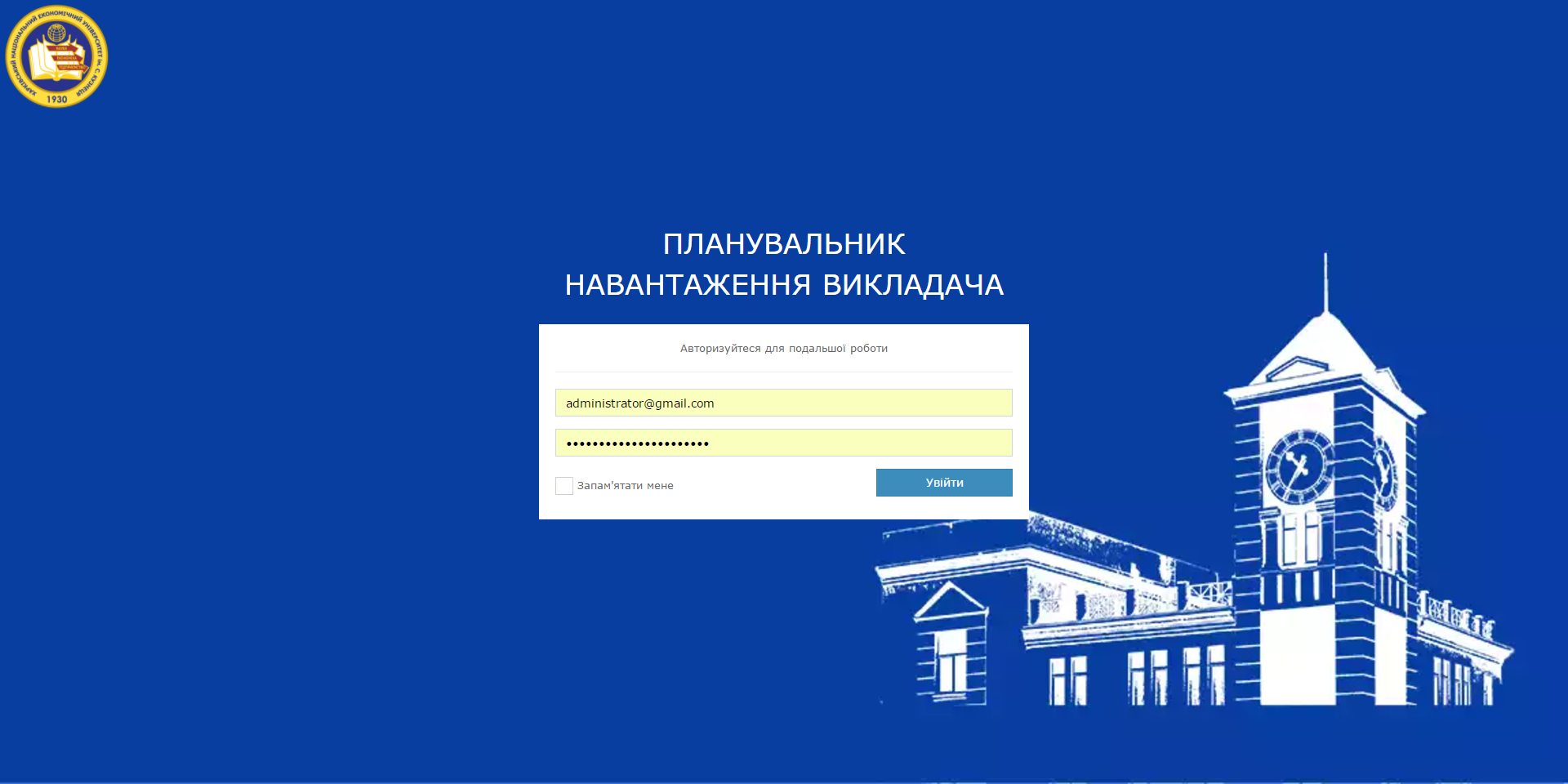


Рис. 2 Форма входу у систему

Для входу в систему, передбачено 7 ролей: Викладач модератор, викладач, голова методичного відділу, голова наукового сектора, голова навчальної частини. Якщо увійти під роллю «Адміністратор», то відкривається сторінка, що зображена на рис. 3. На цій формі користувач має доступ до усіх вкладок системи.

Перейшовши на сторінку «головна», голова наукового сектора зможе побачити власну інформацію (ПІБ, посада, кафедра тощо). Також, на цій сторінці, користувач зможе натиснути кнопку «редагувати», після чого матиме можливість редагувати власні дані (рис. 4).

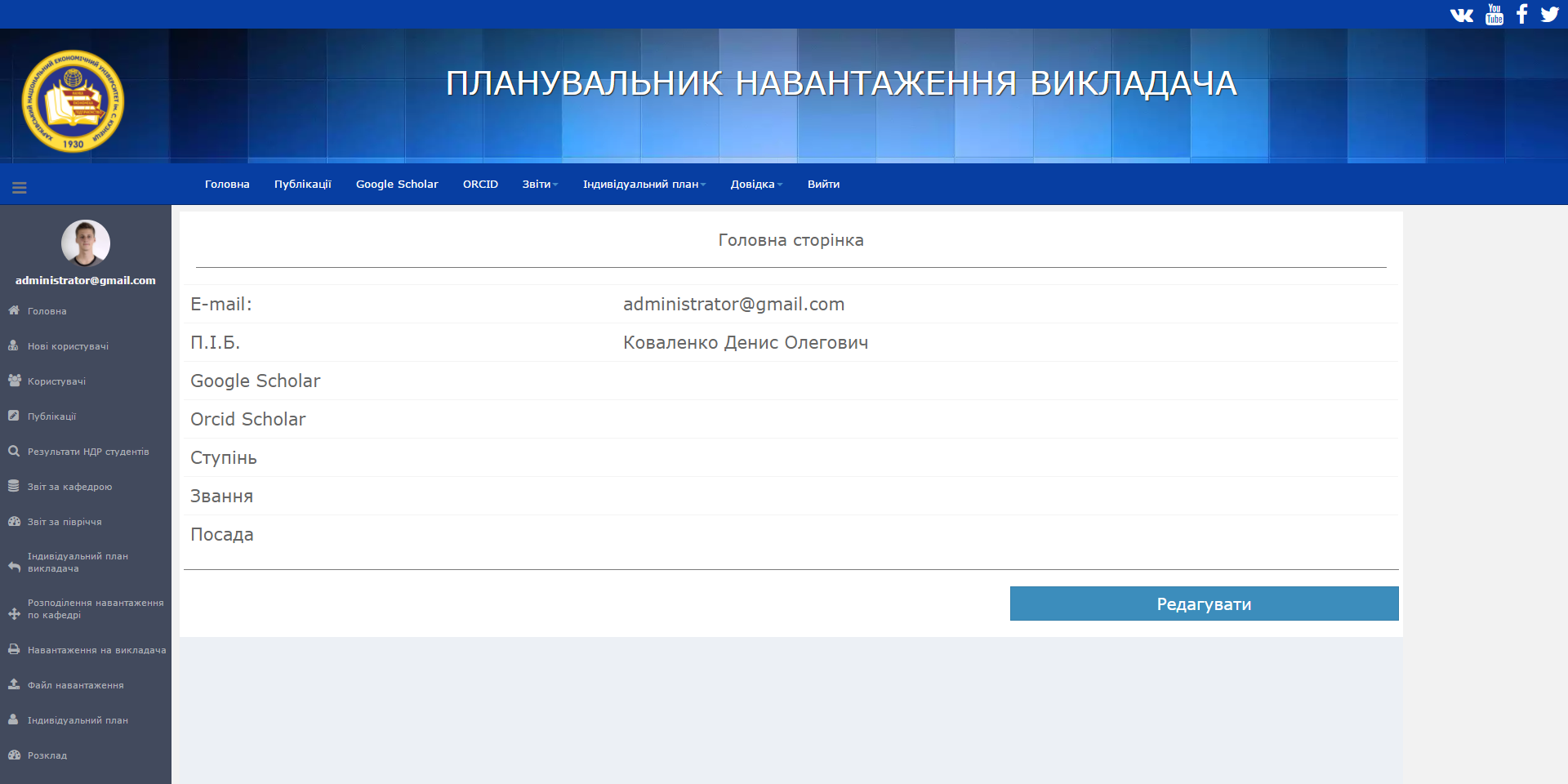


Рис. 3 Сторінка з правами «Голова наукового сектора»

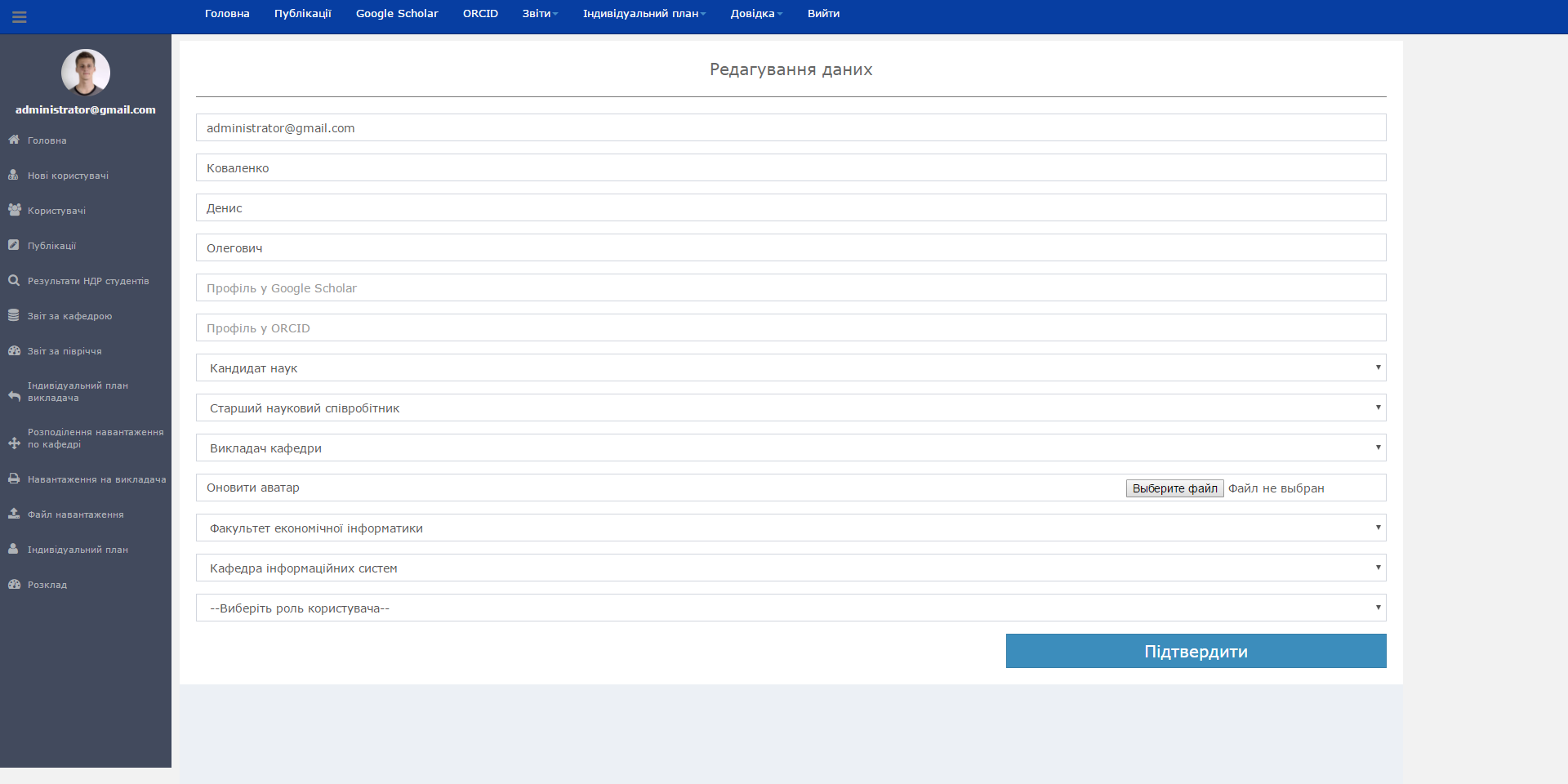


Рис. 4 Сторінка «Головна сторінка»

Натиснувши на випадаючий список у меню «індивідуальний план», голова наукового сектора має можливість формувати індивідуальний план по вкладці: «Наукова робота» (рис. 5).

На сторінці «наукова робота», що наведена нижче, голова наукового сектора може переходити між табами, та додавати кількість годин по плану та по факту, в кожній із робіт, а саме по натисненню кнопки підтвердити до індивідуального плану додається новий запис.

Примітка: при невірному введені полів – з’явиться помилка, так як на кожному полі стоїть обмеження на коректність введених даних. Після некоректного введення, поле, де було введено дані, буде підсвічено червоним кольором, повідомляючи про те, що потрібно ввести коректні дані.

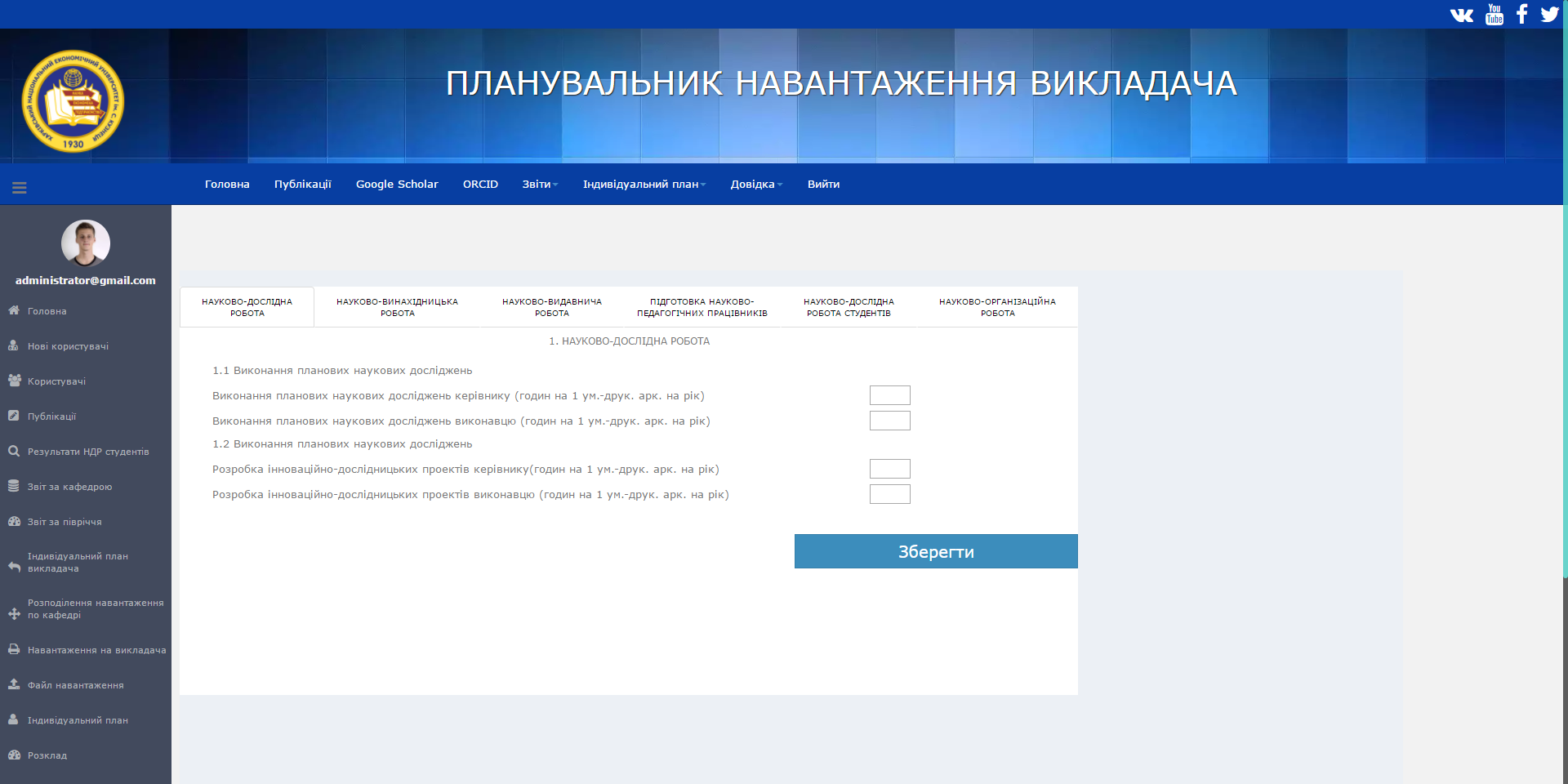


Рис. 5 Сторінка «Наукова робота»

На даній сторінці користувач завжди має можливість переглянути потрібний йому розклад (рис. 6).

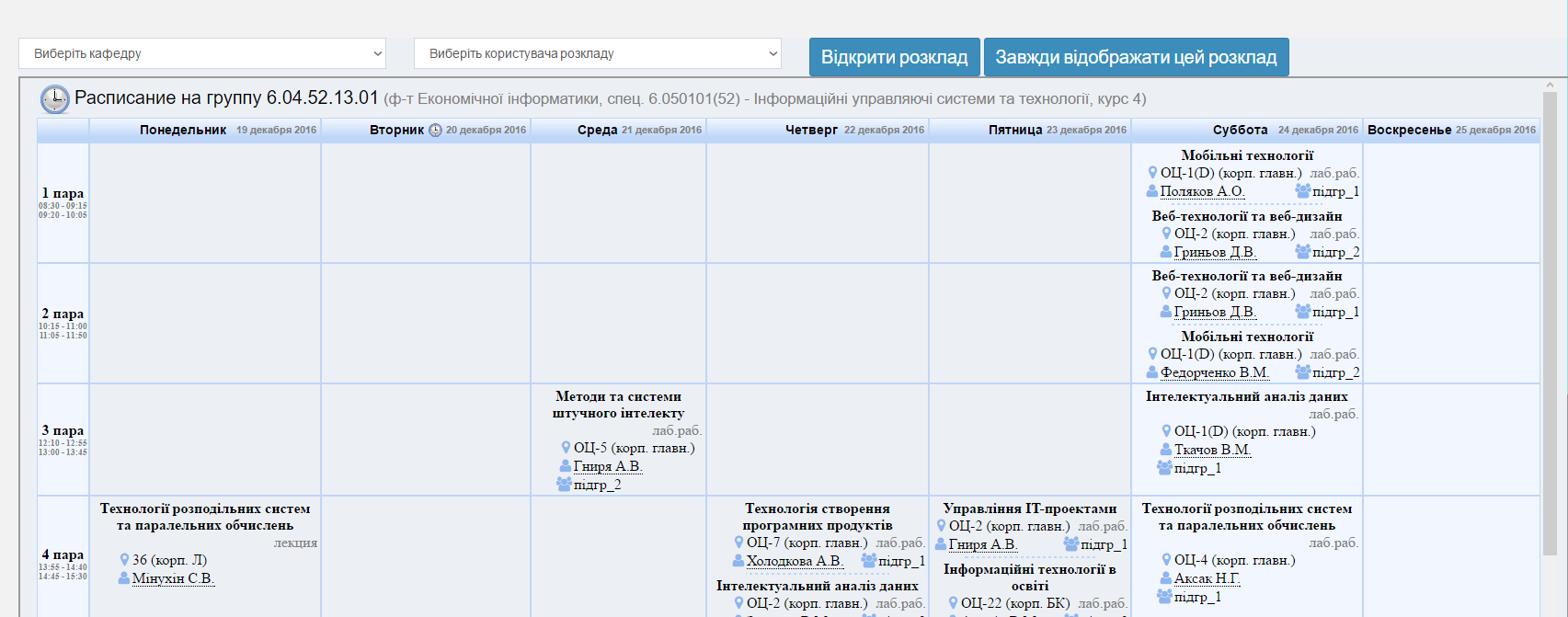


Рис. 6 Сторінка «Розклад»