***К.Ю. Кожевников, В.С. Иванов, Э.Е. Ярдыков,***

***ст. преп. Н.В. Первова***

**ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ БРОНИРОВАНИЯ МЕСТ   
В КОВОРКИНГЕ**

**Аннотация**. Показан процесс разработки веб-платформы для бронирования рабочих мест в коворкинге университета. Описаны функциональные возможности сайта, их отображение для пользователя.

**Ключевые слова:** коворкинг, университет, веб-платформа, Laravel, MySQL, бронирование, отзывы.

***K.Y. Kozhevnikov, V.S. Ivanov, E.E. Yardykov,***

***Senior Lecturer N.V.Pervova***

**A WEB APPLICATION FOR BOOKING SPACES  
IN A CO-WORKING SPACE**

**Annotation.** The process of developing a web platform for booking workplaces in the university co-working center is shown. The functionality of the site, their display for the user are described.

**Keywords:** coworking, university, web platform, Laravel, MySQL, booking, reviews.

В современном мире концепция коворкинга становится все более популярной, особенно среди студентов и молодых специалистов, стремящихся к продуктивной и вдохновляющей рабочей среде. Коворкинг представляет собой совместное использование рабочего пространства, где люди с различными профессиями и областями интересов могут собираться в одном месте для работы, обмена идеями и совместного развития.

Основные особенности коворкинга:

1. Удобное расположение: коворкинг располагается в университетских зданиях или их непосредственной близости, что делает его легко доступным для студентов и преподавателей.
2. Разнообразие рабочих зон: проект включает в себя разнообразные зоны для работы, включая индивидуальные столы, общие рабочие зоны, комфортные зоны отдыха и общественные зоны для общения.

Однако, несмотря на все удобства и преимущества коворкинга возникает проблема с доступом к информации о свободных местах в коворкинге.

Для обеспечения удобства и эффективного использования пространства коворкинга разработана веб-платформа, позволяющая студентам бронировать рабочие места заранее.

В данной работе описывается веб-платформа для бронирования рабочих мест в коворкинге университета. Этот проект нацелен на обеспечение удобства и эффективности использования рабочих пространств в университете, а также на создание инструмента, который способствует развитию совместного взаимодействия и творчества среди студентов.

Главной целью является создание атмосферы для самостоятельного труда, обмена знаниями, совместной работы и общения между студентами из различных факультетов и специальностей.

Для создания веб-платформы бронирования рабочих мест в университетском коворкинге используются современные технологии и инструменты. Основой разработки является фреймворк Laravel, который обеспечивает удобную структуру и инструменты для создания веб-приложений. HTML, CSS и JavaScript применяются для создания интерфейса пользователя, обеспечивая удобство и привлекательный дизайн веб-страниц. Для хранения данных о бронированиях и другой информации используется база данных MySQL, обеспечивающая эффективное хранение и быстрый доступ к данным. В качестве редактора кода используется Visual Studio Code, который обеспечивает удобную среду разработки. Разработка ведется на языке программирования PHP с использованием Blade шаблонизатора для работы с шаблонами HTML.

Платформа для бронирования рабочих мест в университетском коворкинге состоит из нескольких основных страниц, каждая из которых выполняет определенные функции:

* Главная страница: здесь пользователи получают общую информацию о коворкинге, его целях и возможностях. Здесь предоставляется краткое описание услуг, которые предоставляет платформа, а также инструкции о том, как начать использовать её. На главной странице также размещены ссылки на другие разделы сайта, такие как страница бронирования и страница отзывов.
* Страница бронирования: эта страница предоставляет пользователям возможность выбрать дату бронирования, место работы и временной интервал, а также ввести свои студенческие данные. Здесь пользователи могут увидеть доступные места для бронирования на выбранную дату и время, а также получить информацию о том, сколько мест доступно в каждом рабочем пространстве.
* Страница отзывов: на этой странице пользователи могут просматривать отзывы других пользователей о коворкинге и оставлять свои собственные отзывы. Здесь они могут поделиться своими впечатлениями о качестве услуг, оставить рекомендации для других пользователей и выразить свои предложения по улучшению сервиса.

Реализация функционала сайта с использованием Laravel начинается с разработки контроллеров, которые содержат логику обработки запросов и взаимодействия с моделями.

Например, для отображения сайта бронирования и передачи данных с базы данных на сайт используется данная функция контроллера, которая возвращает переменную, содержащую нужную информацию из базы данных:

public function reserve()

{

$reserves = ReserveModel::all();

return view('reserve',['reserves' => $reserves]);

}

Для отправки данных о бронировании уже используется другая функция контроллера:

public function reserve\_check\_2(Request $request)

{

$valid = $request->validate([

'userdate' => 'required',

'userids' => 'required',

'userplace' => 'required'

]);

$userdate = $request->input('userdate');

$userids = $request->input('userids');

$userplace = $request->input('userplace');

$usertimes = $request->input('usertime');

if ($usertimes == 0) {

return redirect()->route('reserve');

}

foreach ($usertimes as $usertime) {

$reserve = new ReserveModel();

$reserve->data = $userdate;

$reserve->reserve\_time = $usertime;

$reserve->slot\_id = $userplace;

$reserve->UIDs = $userids;

$reserve->save();

}

return redirect()->route('reserve');

}

Также необходимы маршруты, которые определяют, какие действия выполнять при обращении к определенным URL-адресам.

Например:

* для отображения страницы сайта бронирования:

Route::get('/reserve', [MainController::class, 'reserve'])->name('reserve');

* для проверки валидности данных:

Route::post('/reserve/check2',[MainController:: class,'reserve\_check\_2'])>name('reserve\_check\_2');

После того как контроллеры и маршруты определены, создаются представления – HTML-шаблоны, которые отображают информацию пользователю. В представлениях используются переменные, передаваемые из контроллеров, для динамического отображения данных.

Для отправки данных с клиента на сервер используется тег <form>с методом «post».

<form action="{{ route('reserve\_check\_2') }}" method="POST">

<input type="date" id="selected\_date"

name="selected\_date">

<select name="selected\_place"

id="selected\_place"> </select>

<input type="text" name="userdate"

id="userdate">

<input type="text" name="userplace"

id="userplace">

<input type="submit" value="Забронировать">

</form>

В ходе разработки было создано веб-приложение, которое значительно упрощает процесс бронирования места на коворкинге для пользователей. С помощью этого приложения пользователи могут легко просматривать доступные места на площадке, выбирать наиболее подходящее себе и забронировать его в удобное время. Интерфейс приложения интуитивно понятен и удобен в использовании, что делает процесс бронирования максимально простым и быстрым. Кроме того, веб-приложение предоставляет пользователю информацию о доступных временных интервалах, позволяя выбрать оптимальный для своих потребностей. Таким образом, созданное приложение значительно повышает удобство и эффективность процесса бронирования места на коворкинге для пользователей.

Литература

1. Документация Laravel 10.x // Установка Laravel 10.x: Шаг за Шагом Руководство - Laravel Docs URL: https://laravel-docs.com/ru/docs/10.x (дата обращения: 8.03.2024).
2. jQuery метод .submit() // Метод .submit() | jQuery справочник URL: https://basicweb.ru/jquery/jquery\_event\_submit.php (дата обращения: 8.03.2024).
3. Руководство по PHP // PHP: PHP Manual - Manual URL: https://www.php.net/manual/en/index.php (дата обращения: 8.03.2024).
4. HTML: язык разметки гипертекста // HTML: HyperText Markup Language | MDN URL: https://developer.mozilla.org/en-US/docs/Web/HTML (дата обращения: 8.03.2024).
5. CSS: каскадные таблицы стилей // CSS: каскадные таблицы стилей | MDN URL: https://developer.mozilla.org/ru/docs/Web/CSS (дата обращения: 8.03.2024).