

который должен будет облегчить поиск попутчиков со следующими функциональными возможностями:

- регистрация пользователя;
- проверка данных пользователя;
- поиск маршрута по заданным пользователем точкам;
- ввод водителем стоимости маршрута (при отсутствии стоимости будет происходить автоматический расчет, не превышающий стоимости бензина за половину пути);
- отображение стоимости маршрута;
- поиск срочных маршрутов;
- чат между водителем и пассажиром конкретной поездки;
- просмотр и написание отзывов.

Разработанное веб-приложение позволит облегчить поиск попутчиков, обеспечив при этом общение между водителем и пассажиром в рамках сайта и облегчить использование сервиса создание его интуитивно понятным, удобным для работы, как пользователям, так и администратору.

**К. Е. Карпекина**

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. А. Дробышевский**, ст. преподаватель

## **АРХИТЕКТУРА WEB-ПРИЛОЖЕНИЯ ДЛЯ СОВМЕСТНОГО ИСПОЛЬЗОВАНИЯ ЧАСТНОГО АВТОМОБИЛЯ**

Сервис для совместного использования частного автомобиля реализован с использованием Spring MVC.

MVC паттерн разделяет приложение на три основных части:

- 1 Модель.
- 2 Представление.
- 3 Контроллер.

Модели сервиса представляют собой простые Java классы. Для работы с базой данных используется специальный интерфейс JpaRepository, предоставляющий стандартные методы для работы с базой данных. Напрямую использовать Repository нежелательно, поэтому для получения данных созданы соответствующие сервис классы, в которых переопределяются методы для запросов в базу данных.

Представления для просмотра информации в Web-приложении реализованы в виде html страниц с использованием thymeleaf для реализации java-кода на SpEL (spring expression language) внутри html страниц, как представлено ниже.

```
<div th:if="{param.success}">
  <div class="alert alert-info">You've successfully
    registered!</div>
</div>
```

Для обработки пользовательских запросов созданы соответствующие контроллеры. Контроллеры определены с помощью специальных аннотаций: @Controller – указывает, что класс является контроллером; @RequestMapping(...). Основными параметрами аннотации являются:

- параметр value, предназначен для указания URL-адреса;
- параметр method, предназначен для указания метода доступа (Get, Post, Put, Delete, Patch);
- параметр consumes, предназначен для определения типа содержимого тела запроса.

Рассмотренная архитектура проекта на основе Spring MVC обеспечивает гибкость и легкость в тестировании. Это обусловлено разделением приложения на составные компоненты. Благодаря данной архитектуре можно вносить изменения и добавлять новую функциональность на последних этапах разработки.

**Ф. В. Карпман**

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

## **РАЗРАБОТКА ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА ДЛЯ ПРОВАЙДЕРА ОБРАЗОВАТЕЛЬНЫХ УСЛУГ**

Проект представляет собой веб-приложение для предоставления образовательных услуг в режиме онлайн. Приложение разделено на несколько ролей - ученик и преподаватель. В приложении доступно несколько модулей в зависимости от роли пользователя. Учитель имеет доступ к тематическому планированию, журналу, видеоконференцсвязи в качестве создателя, планированию урока, дневнику, библиотеке и домашнему заданию в качестве рецензента. В свою очередь, студент имеет доступ к видеоконференцсвязи в качестве участника, дневнику, библиотеке и домашнему заданию.