



## Литература

- 1 Ковалев, М.А. Самолетный ответчик СО-69 [Текст]: учебное пособие / Самарс. Гос. Аэрокосмический ун-т. М.А. Ковалев. – Самара, 2003. – 48 с.
- 2 Методология объектно-ориентированного анализа и проектирования [Электронный ресурс.] URL: <https://intuit.ru/studies/courses/32/32/lecture/1000?page=2> (дата обращения 08.04.2022).

Г.С. Малахов, Е.В. Симонова

## РАЗРАБОТКА ВЕБ ПРИЛОЖЕНИЯ ДЛЯ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ

(Самарский университет)

### Введение

Процесс обучения программированию представляет собой последовательное прохождение множества теоретических и практических блоков. В связи с этим возникает проблема систематизации и запоминания полученных знаний.

В настоящее время обучение программированию основано на использовании различных подходов и методик: книг, разрозненных статей, учебных пособий, познавательного-развлекательного контента и т.п. Вследствие этого многообразия обучающийся может получить противоречивую или даже неверную информацию. Систематизация и наглядная визуализация учебных материалов обеспечивают однозначное и согласованное представление знаний, что облегчает усвоение материала.

### Постановка задачи

Образовательная платформа должна предоставлять интуитивно понятную структуру излагаемого материала, комфортный графический интерфейс, практические задания для каждой темы, обеспечивать должный уровень безопасности, возможность накопления частной и глобальной статистики, визуализировать прогресс пользователя, предоставлять возможности для социального взаимодействия и иметь динамический интерфейс.

### Предлагаемые решения

Для создания прочного каркаса, на основе которого будет происходить запоминание информации, необходимо ее структурирование с возможностью визуализации [1], т.е. графическое представление информации с четко выраженной структурой. Этим требованиям отвечает диаграмма «Ментальная карта», используемая для визуальной организации информации. Ментальная карта иерархична и показывает отношения между частями целого [2]. На рисунке 1 представлен пример ментальной карты, описывающий процесс создания Web приложения по обучению программированию.



Комфортный графический стиль возможно обеспечить, используя таблицы каскадных стилей, различные языки разметки, а также фреймворки графических компонентов: Bootstrap, ApexChart, Vuetify и другие.

Для обеспечения высокого уровня знаний и проверки усвоения материала необходимо обратиться к признанным работам известных программистов. Например, для web-приложения по обучению программированию на языке Java следует использовать книгу «Java. Библиотека профессионала» Кея Хорстманна [3]. Это позволит гарантировать качество и доступность материала в теоретических блоках.



Рис. 1. Ментальная карта процесса создания Web-приложения по обучению программированию

Любое приложение, работающее с пользовательскими данными, должно гарантировать безопасность информации, переданной пользователем. Этим требованиям отвечает открытый протокол авторизации OAuth 2.0, обеспечивающий предоставление третьей стороне ограниченного доступа к защищённым ресурсам пользователя без передачи логина и пароля [4]. В качестве системы, предоставляющей услуги аутентификации, возможно использование сервиса Google OAuth 2.0 с последующей регистрацией приложения в Google Cloud Platform, что позволит автоматизировать процесс аутентификации, используя любой из имеющихся у пользователя Google-аккаунтов.

Для хранения собранных статистических данных предлагается объектно-реляционная СУБД PostgreSQL, обладающая рядом преимуществ по сравнению с другими SQL базами данных с открытым исходным кодом (MySQL, MariaDB и Firebird): поддержка пользовательских объектов и их поведения, включая типы данных, функции, операции, домены и индексы, создание, хранение и извлечение сложных структур данных, что обеспечивает гибкость и надежность СУБД PostgreSQL [5].

Важным моментом в процессе любого обучения является прогрессивная оценка. Для этой цели была разработана система достижений, отображаемая на персональной странице пользователя. Достижения содержат информацию о том, что пользователь завершил изучение определенной главы или правильно ответил на большинство вопросов с первой попытки. Кроме достижений имеется система визуальной индикации прогресса пользователя на ментальной карте,



где зеленым цветом обозначаются вершины диаграммы, соответствующие изученным главам учебных материалов, а желтым цветом – вершины, соответствующие главам, изучение которых не завершено.

Для обеспечения социального взаимодействия была выбрана концепция социальной сети. Социальная сеть – онлайн-платформа, которая используется для общения, знакомств, создания социальных отношений между людьми, имеющими схожие интересы или офлайн-связи, а также для развлечения и работы. Были разработаны классические элементы социальных сетей: страницы пользователя, друзей, бесед и диалогов.

### **Архитектура приложения**

Современное веб-приложение должно быть динамичным и отвечать требованиям к одностраничным приложениям. Одностраничное приложение (Single Page Application (SPA)) – это веб-приложение или веб-сайт, использующий единственный HTML-документ как оболочку для всех веб-страниц и организующий взаимодействие с пользователем через динамически подгружаемые HTML, CSS, JavaScript, обычно посредством AJAX. Для этого следует использовать соответствующие фреймворки.

Для структурирования приложения подходит фреймворк Vue.js, который является только слоем представления, поэтому его можно использовать непосредственно на странице приложения вместо полномасштабного фреймворка. Это дает больше возможностей для сочетания Vue с другими библиотеками, но за архитектурные решения отвечает программист, например, ядро Vue.js не включает маршрутизацию или ajax-функции, и, как правило, предполагает, что приложение реализуется с использованием внешнего модуля [6].

Серверная сторона должна обеспечивать декларативное, производительное и легко настраиваемое функционирование со всеми вышеперечисленными блоками системы. С этими задачами может справиться Spring, рассматриваемый как коллекция меньших фреймворков или фреймворков во фреймворке. Большинство этих фреймворков может работать независимо друг от друга, однако они обеспечивают большую функциональность при совместном использовании.

Эти фреймворки являются структурными элементами типовых комплексных приложений:

- Контейнер Inversion of Control – обеспечивает делегированное (фреймворку) конфигурирование компонентов приложений и управление жизненным циклом Java-объектов.
- Фреймворк доступа к данным – работает с системами управления реляционными базами данных на Java-платформе, используя JDBC- и ORM-средства и обеспечивая решение задач, которые повторяются в большом числе Java-based environments.
- Фреймворк управления транзакциями – обеспечивает координацию различных API управления транзакциями и инструментарий настраиваемого управления транзакциями для объектов Java.



- Фреймворк MVC – предоставляет каркас, основанный на HTTP и сервлетах, дающий множество возможностей для расширения и настройки (customization) [7].

### Реализация

На рисунках 2-5 представлена последовательность работы с веб приложением для обучения программированию.

На рисунке 2 показана динамическая ментальная карта с визуальной индикацией пройденных тем:

- Введение в язык Java.
- Основные языковые конструкции Java.
- Объекты и классы.
- Наследование.
- Интерфейсы, лямбда-выражения и внутренние классы.
- Исключения, утверждения и протоколирование.
- Обобщенное программирование.
- Коллекции.
- Параллелизм.
- Поток данных.
- Ввод и вывод.
- XML.
- Модульная система на платформе Java.



Рис. 2. Динамическая ментальная карта с визуальной индикацией пройденных тем



При переходе с динамической ментальной карты открывается окно, в котором отображается тема, выбранная для изучения (рисунок 3).

После изучения темы обучаемому предлагается выполнить задания для проверки полученных знаний (рисунок 4).

По результатам обучения пользователю предоставляется статистика (рисунок 5).

Типы предоставляемой статистики:

- Количество попыток решения и количество тем в главе.
- Процент правильных ответов при выполнении заданий во всех главах.
- Время, затраченное на изучение каждой главы обучаемым и другими пользователями.
- Процентное соотношение правильных решений обучаемого и среднего значения результатов других пользователей.

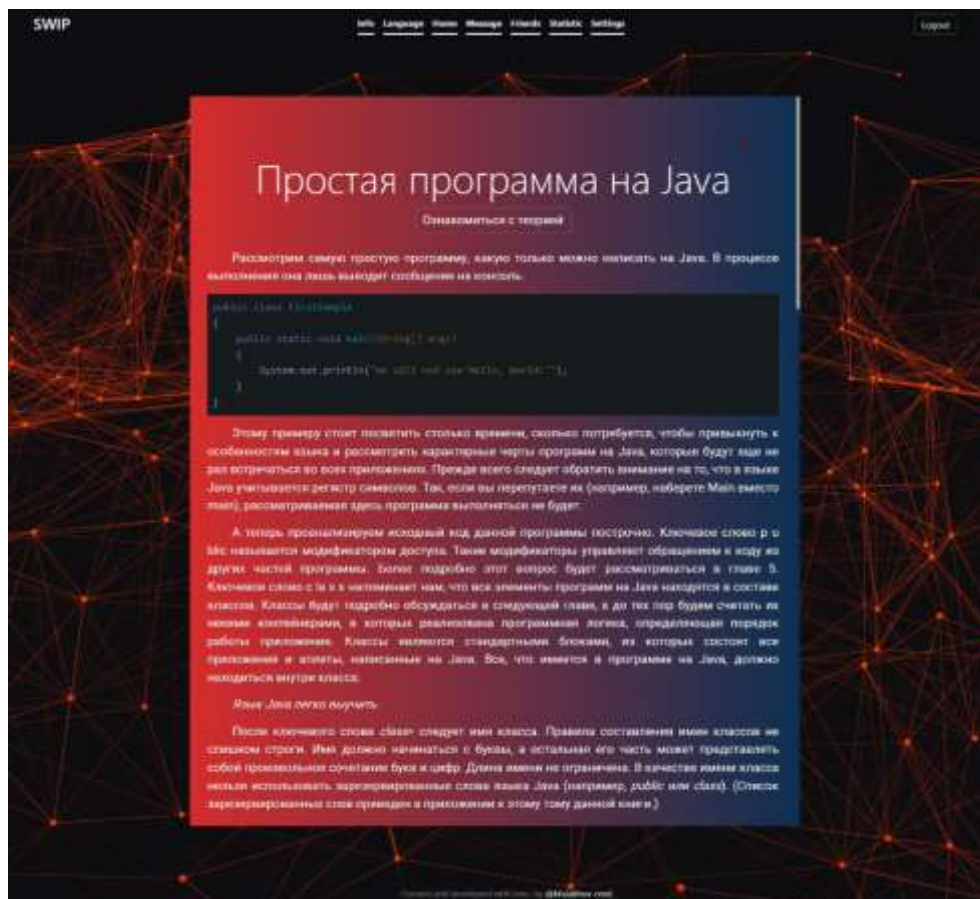


Рис. 3. Тема «Пример простой программы на Java»

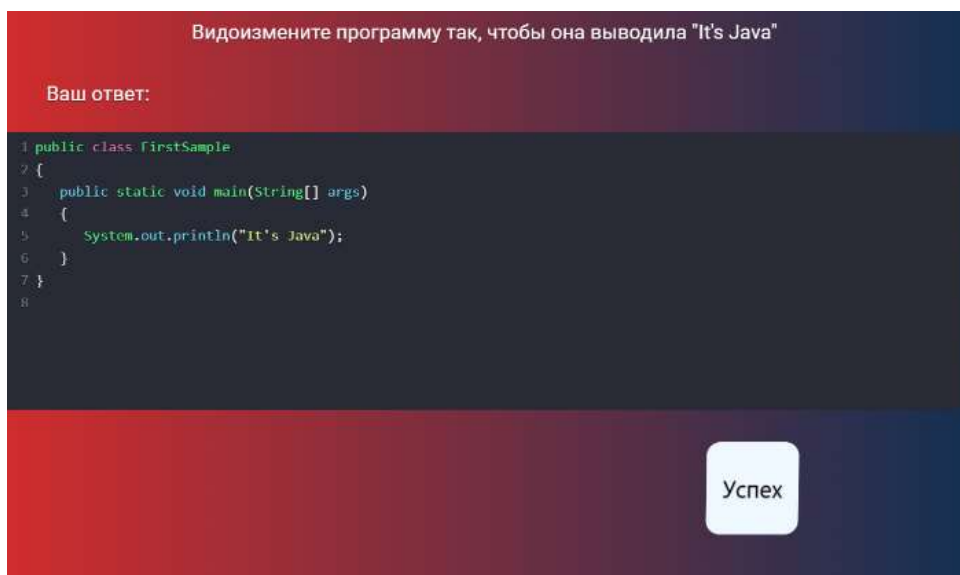


Рис. 4. Результаты правильного выполнения задания

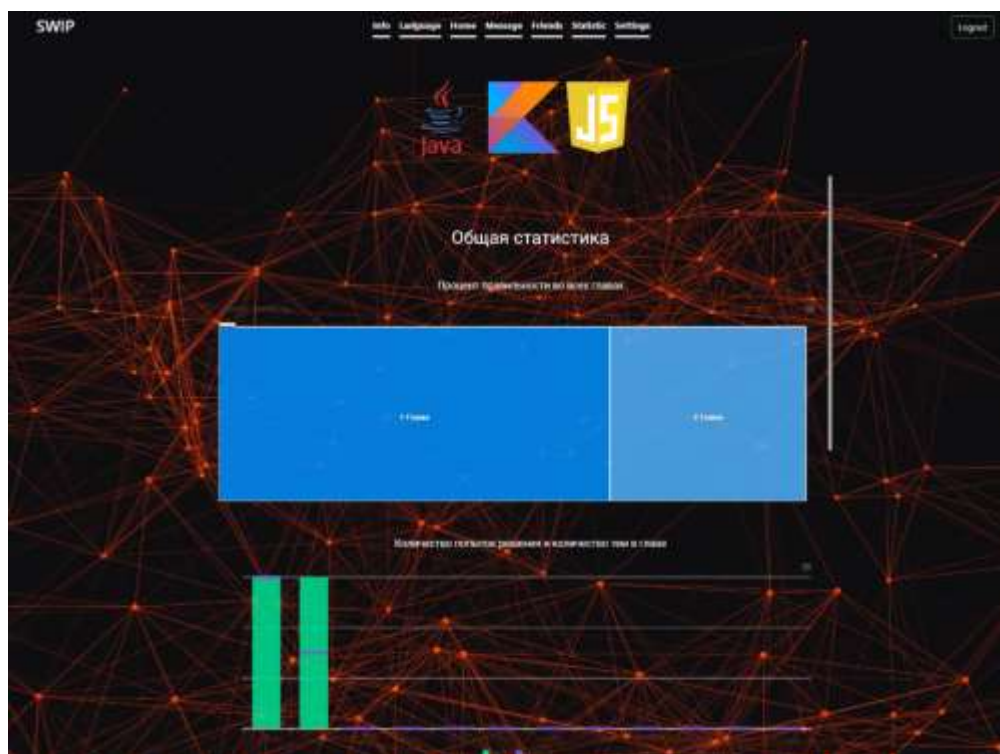


Рис. 6. Статистика, предоставляемая пользователю

У каждого пользователя имеется своя страница-блог для взаимодействия с другими пользователями и написания статей. Обучаемые могут организовать чат для обсуждения заданий и консультации с другими пользователями по возникшим вопросам.

### Заключение

Таким образом, разработанное веб-приложение предоставляет необходимую функциональность для обучения программированию.



## Литература

1. Структурирование информации: понятие и виды, модели и примеры [Электронный ресурс]. – Режим доступа: [Структурирование информации: понятие и виды, модели и примеры](#)
2. Ментальная карта [Электронный ресурс]. – Режим доступа: [Ментальная карта](#)
3. Core Java Volume I Volume II Fundamentals Advanced Topics 12th Edition [Электронный ресурс]. – Режим доступа: [Core Java Volume I Volume II](#)
4. OAuth [Электронный ресурс]. – Режим доступа: [OAuth](#)
5. Чем PostgreSQL лучше других SQL БД с открытым исходным кодом [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/282764/>
6. Vue [Электрон. ресурс]. – Режим доступа: [Vue](#)
7. Spring Framework [Электрон. ресурс]. – Режим доступа: [Spring Framework](#)

А.Д. Черкасов, А.А. Лобанков

## РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ДЛЯ ОБСУЖДЕНИЯ И ОЦЕНИВАНИЯ СТУДЕНЧЕСКИХ РАБОТ

(Самарский университет)

Качество студенческой работы зависит от многих показателей, главными из которых являются своевременное выполнение студентом этапов работы, а также обратная связь от преподавателя [1]. Также, поэтапное выполнение работы помогает студенту правильно распоряжаться своим временем и не терять мотивацию, так как он постоянно будет думать о том, что ему необходимо сдать следующую «контрольную точку». При использовании такого подхода, можно свести к минимуму такие ситуации, при которых выпускник откладывает выполнение работы на последний момент, а потом не успевает ее сделать. Особенно это актуально при выполнении объемной работы, такой как выпускная квалификационная работа. Таким образом, последовательное написание работы студентом и своевременная проверка каждого этапа преподавателем позволяет получить качественную работу.

Целью данной работы является разработка автоматизированной системы для обсуждения и оценивания студенческих работ, с помощью которой можно согласовывать темы работ, а также поэтапно обсуждать и оценивать прогресс выполнения работы. Гибкость данной системы позволит использовать ее для обсуждения и оценивания выпускных квалификационных работ, лабораторных и курсовых работ, а также других видов работ.

Разрабатываемая система представляет собой веб-приложение. Она предоставляет функции для того, чтобы разбить большую задачу на этапы, каждый из которых имеет дату крайнего срока сдачи, а также, функции для по-