

УДК 004.42

**РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПРОВЕДЕНИЯ
ОЛИМПИАД КАЛУЖСКИМ ГОСУДАРСТВЕННЫМ УНИВЕРСИТЕТОМ
ИМ. К.Э. ЦИОЛКОВСКОГО****Винокуров Игорь Викторович**

кандидат технических наук, доцент кафедры «Информатика и информационные технологии» Калужского Государственного Университета им. К.Э. Циолковского.

Дереглазов Кирилл Юрьевич

студент кафедры «Информатика и информационные технологии» Калужского Государственного Университета им. К.Э. Циолковского

Аннотация

В статье рассматривается реализация информационной системы проведения олимпиад среди школьников, учащихся и выпускников организаций среднего профессионального образования, разработанной в Калужском Государственном Университете им. К.Э. Циолковского. Описываются этапы его проектирования и реализации. Приводятся использованные при её разработке информационные технологии и инструментальные средства.

Ключевые слова: информационная система, контроллер, представление, MVC, Razor, REST.

**IMPLEMENTATION OF THE INFORMATION SYSTEM OF THE OLYMPIAD
OF KALUGA STATE UNIVERSITY NAMED AFTER K.E. TSIOLKOVSKI****Igor V. Vinokurov**

candidate of technical sciences, associate professor "Informatics and information technology" Kaluga State University named after K.E. Tsiolkovski

Kirill Yu. Dereglazov

student of the department "Informatics and information technology" Kaluga State University named after K.E. Tsiolkovski

ABSTRACT

The article discusses the implementation of the information system for conducting competitions among schoolchildren, students and graduates of secondary vocational education organizations, developed at Kaluga State University. K.E. Tsiolkovski. The stages of its design

and implementation are described. Information technologies and tools used in its development are considered.

Keywords: informational system, controller, view, MVC, Razor, REST.

Информационная система (ИС) для проведения олимпиад КГУ им. К.Э. Циолковского, как и любое *web*-приложение, состоит из двух основных составляющих – серверной и клиентской. Особенности функционирования каждой из этих составляющих определяются текущим режимом работы *web*-приложения – режимом пользователя или режимом администрирования. В режиме пользователя осуществляется ввод информации при регистрации нового участника олимпиад и прохождение им тестирования по выбранному предмету. В режиме администрирования реализуется описание предмета олимпиады, формирование списка вопросов и соответствующих им ответов, осуществляется проверка правильности вопросов олимпиады. В этом же режиме реализуется и формирование результатов проведения олимпиад. Логическая организация ИС приведена на рисунке 1.

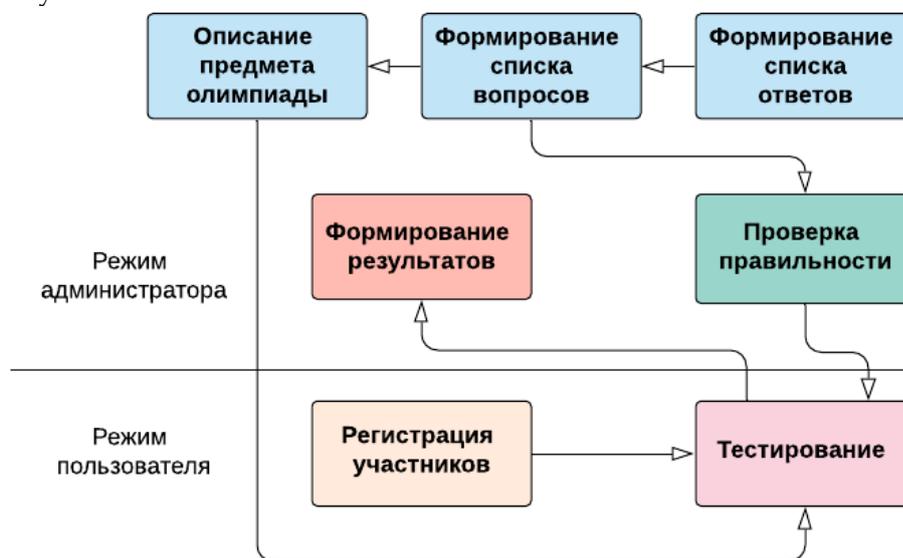


Рисунок 1. Логическая организация ИС для проведения олимпиад

Для реализации *web*-приложения была выбрана платформа .NET, являющаяся основной на серверах вуза и, как следствие, технология ASP.NET [1], предполагающая использование архитектурного паттерна MVC [1,2] и формирование динамических html страниц с использованием технологии Razor [3]. Такая организация предполагает отправку состояния модели данных контроллеру, обработку её соответствующими методами контроллера и отображение нового состояния модели на html-странице. Под моделью данных в данном случае понимается либо вычисляемая, либо проецируемая из источника данных ИС сущность.

Все классы серверной части *web*-приложения были объединены в пакеты с учетом шаблона MVC (рис. 2).

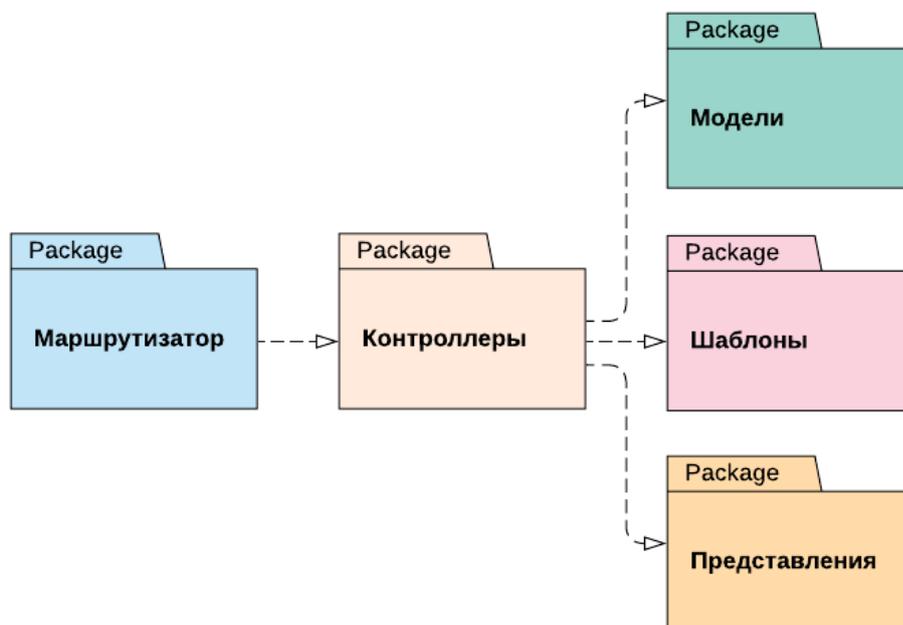


Рисунок 2. Диаграмма пакетов серверной части ИС

Классы пакета “Маршрутизатор” анализируют запрос и делегируют его дальнейшую обработку запросу управляющим объектам нижележащего уровня – классам пакета “Контроллеры”. Классы этого пакета отвечают за обработку запроса в результате вызова соответствующих методов (рис. 3). Все методы обработчики используют классы шаблона “Модели”, взаимодействующие с сущностями источника данных ИС. Классы пакетов “Представления” и “Шаблоны” реализуют отображение информации на html-страничках.

Основные контроллеры ИС, классы которых образуют пакет “Контроллеры”, предназначены для взаимодействия с основными сущностями источника данных ИС (рис. 3). На этом же рисунке указаны и реализуемые контроллерами действия.



Рисунок 3. Контроллеры ИС

В таблицах 1 и 2 приведены имена классов контроллеров ИС и их основные методы.

Таблица 1.
Имена контроллеров ИС

Имя класса контроллера	Назначение контроллера	Тип сущности модели данных
HomeController	Контроллер главной странички	-
RegInfosController	Контроллер сущности "Участник"	Проецируемая
SchoolSubjectsController	Контроллер сущности "Предмет"	Проецируемая
QuestionsController	Контроллер сущности "Вопрос"	Проецируемая
AnswersController	Контроллер сущности "Ответ на вопрос"	Проецируемая
ResultsController	Контроллер сущности "Результаты"	Проецируемая

Таблица 2.
Имена основных методов контроллеров для проецируемых сущностей

Имя метода	Назначение метода	Аргументы метода
Index()	Отображение списка экземпляров	-
Create()	Добавление нового экземпляра сущности	Экземпляр сущности
Edit()	Изменение экземпляра сущности	Идентификатор экземпляра
Delete()	Удаление экземпляра сущности	Идентификатор экземпляра
Details()	Отображение информации об экземпляре сущности	Идентификатор экземпляра

Методы контроллеров, приведённые в таблице 2, их имена и аргументы определяются используемой в *web*-приложении методологией REST. Остальные методы приведённых в таблице 1 контроллеров предназначены либо для вызова из основных методов, либо для вызова из клиентской части *web*-приложения посредством технологии Ajax [4].

В таблице 3 приведены классы пакета "Модели".

Таблица 3.
Имена моделей ИС

Имя модели	Назначение модели	Тип сущности модели данных
RegInfo	Информация об участнике	Проецируемая
SchoolSubject	Информация о предмете олимпиад	Проецируемая
Question	Информация о вопросе	Проецируемая
Answer	Информация об ответе	Проецируемая
Results	Информация о результатах проведения олимпиад	Проецируемая
Test	Результаты тестирования	Вычисляемая

Взаимодействие классов пакетов “Контроллеры” и “Модели” реализуют бизнес-логику работы ИС. На рисунках 4 и 5 приведены UML-диаграммы классов этих пакетов.

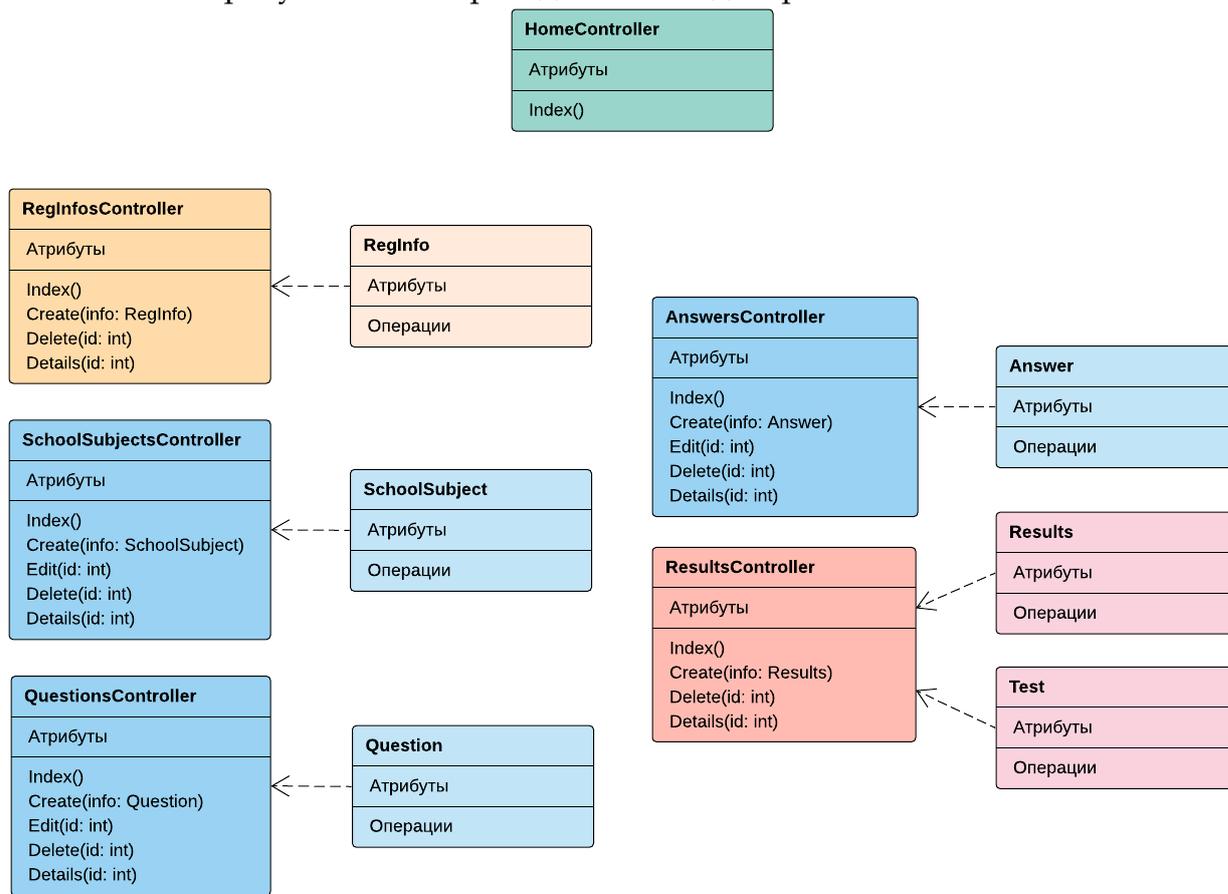


Рисунок 4. UML-диаграмма классов пакетов “Контроллеры”

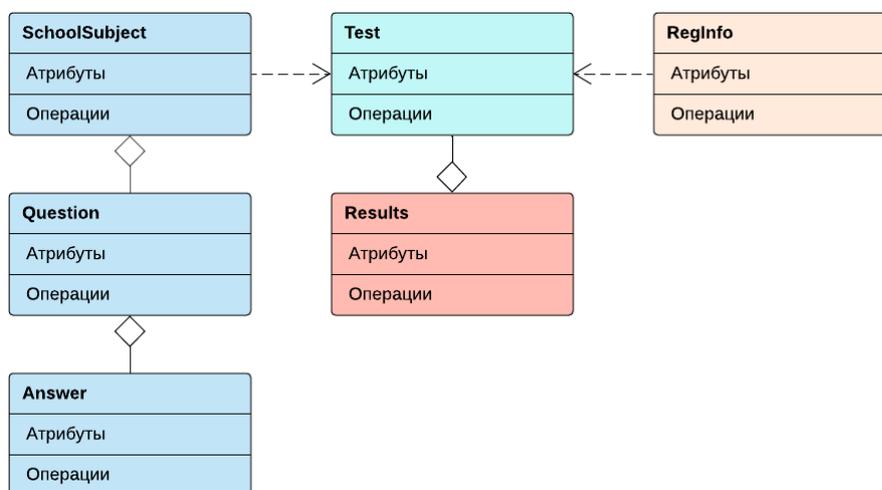


Рисунок 5. UML-диаграмма классов пакетов “Модели”

Клиентская часть *web*-приложения представляет собой архитектурное решение “Толстый клиент”, реализующее функциональный интерфейс с пользователем, в котором выполняется часть бизнес-логики ИС не требующая обращения к БД. Обмен данными между серверной и клиентской частями *web*-приложения приведен на рисунке 6.

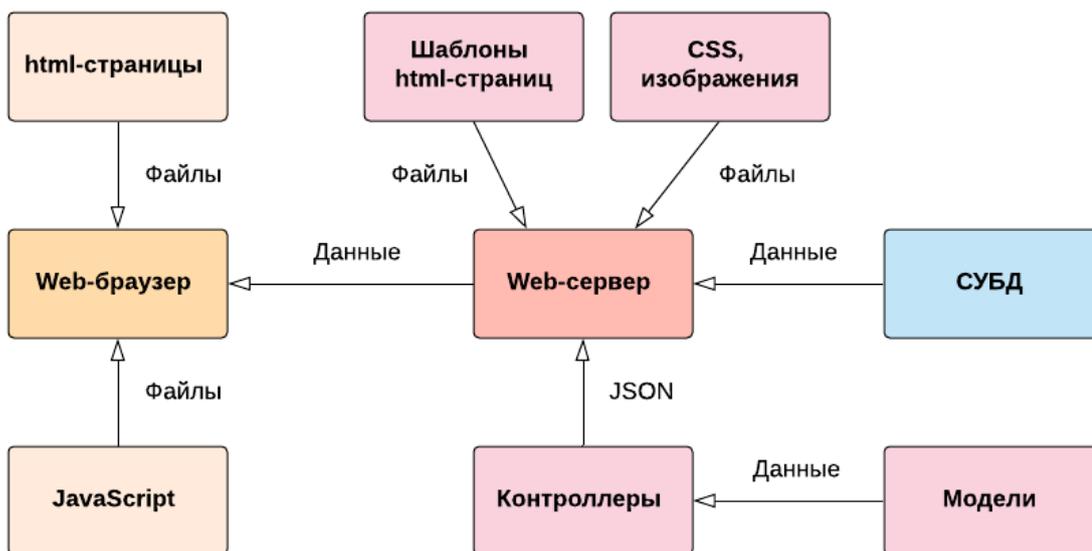


Рисунок 6. Трёхуровневая архитектура взаимодействия серверной и клиентской частей web-приложения

На рисунке 7 показана обобщенная UML-диаграмма последовательности взаимодействия основных частей web-приложения.

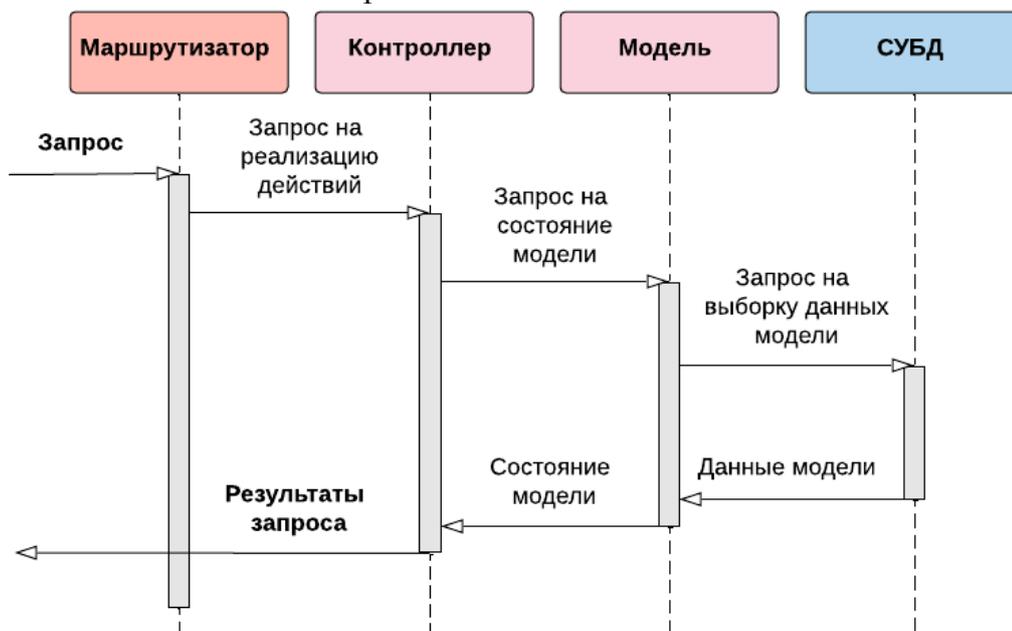


Рисунок 7. Основные этапы взаимодействия серверной и клиентской частей web-приложения

Внешний вид главной странички *web*-приложения приведен на рисунке 8.

Олимпиады Предметы Вопросы Ответы Участники Результаты Администрирование

АДМИНИСТРИРОВАНИЕ

ОЛИМПИАДЫ КГУ ИМ. К.Э. ЦИОЛКОВСКОГО
Для школьников, учащихся и выпускников организаций дополнительного профессионального образования

В 2020 г. КГУ им. К.Э. Циолковского проводит заочные олимпиады по следующим предметам:

Информатика	Основы процедурного программирования на языке С# Пройти тестирование » Время прохождения теста ограничено 45 мин.
История	История древней Руси Пройти тестирование » Время прохождения теста ограничено 120 мин.
Математика	Интегральное и дифференциальное исчисление Пройти тестирование » Время прохождения теста ограничено 100 мин.
Обществознание	Дореволюционная Калуга и Калужская губерния Пройти тестирование » Время прохождения теста ограничено 45 мин.
Химия	Органическая химия Пройти тестирование »

Для корректного отображения информации следует использовать **Microsoft Edge**, **Safari** - только под управлением macOS (Mac OS) и **Mozilla Firefox**

© 2020. Кафедра "Информатика и информационные технологии" © 2020. КГУ им. К.Э. Циолковского

Разработчик
Доцент, к.т.н. **Винокуров И.В.**

Рисунок 8. Главная страница web-приложения в режиме администратора

Web-приложение реализовано в среде Visual Studio 2019. Для реализации серверной части был выбран язык Visual С# [5], клиентской – JavaScript, jQuery [6] и фреймворк Bootstarp 3 [7], входящие в состав проекта, формируемого Visual Studio на начальном этапе разработки *web*-приложения.

Список литературы

1. Эспозито, Дино Программирование на основе Microsoft ASP.NET MVC [Текст] / Д. Эспозито. – М: БХВ-Петербург, 2012. – 464 с.
2. Фримен, Адам ASP.NET MVC 5 с примерами на С# 5.0 для профессионалов [Текст] / А. Фримен. – М: Диалектика-Вильямс, 2016. – 736 с.
3. ASP.NET MVC-Razor. [Электронный ресурс]. Режим доступа: https://www.tutorialspoint.com/asp.net_mvc/asp.net_mvc_razor.
4. Пауэлл, Thomas Ajax. Полное руководство [Текст] / П. Теренс. – М.: Эксмо, 2009. – 712 с.
5. Троелсен, Эндрю Язык программирования С# 7 и платформы .NET и .NET Core [Текст] / Э. Троелсен, Ф. Джепикс. – М: Вильямс, 2018. – 1328 с.
6. Макфарланд, Дэвид JavaScript и jQuery. Исчерпывающее руководство [Текст] / Д. Макфарланд. – М.: Эксмо, 2012. – 688 с.
7. Bootstarp 3. [Электронный ресурс]. Режим доступа: <http://bootstrap-3.ru>

References

1. Esposito, Dino Programming Microsoft ASP.NET MVC [Text] / D. Esposito. – Microsoft Press. – 2014. – 528 p. [in Russian].
2. Freeman, Adam Pro ASP.NET MVC 5 [Text] / A. Freeman. – Apress. – 2013. – 832 p. [in Russian].
3. ASP.NET MVC-Razor. [Electronic resource]. Access Mode: https://www.tutorialspoint.com/asp.net_mvc/asp.net_mvc_razor [in Russian].
4. Powell, Thomas Ajax. The Complete Reference [Text] / T. Powell. – McGraw-Hill Education. – 2008. – 654 p. [in Russian].
5. Troelsen, Andrew. Pro C # 7 With .NET and .NET Core [Text] / E. Troelsen, F. Jepiks. – Apress. – 2017. – 1372 p. [in Russian].
6. McFarland, David JavaScript & jQuery: The Missing Manual [Text] / D. McFarland. – O'Reilly Media. – 2011. – 538 p. [in Russian].
7. Bootstarp 3. [Electronic resource]. Access Mode: <http://bootstrap-3.ru> [in Russian].