МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО «ВГУ»)**

**УТВЕРЖДАЮ**

Заведующий кафедрой программного обеспечения и администрирования информационных систем



Артемов М. А.

31.08.2019 г.

# РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.ДВ.02.01 Концепция MVC и фреймворки

1. **Шифр и наименование направления подготовки:**

02.04.03 Математическое обеспечение и администрирование информационных систем

1. **Магистерская программа:** Информационные технологии
2. **Квалификация (степень) выпускника:** магистр
3. **Форма обучения:** очная
4. **Кафедра, отвечающая за реализацию дисциплины:**

Программного обеспечения и администрирования информационных систем

1. **Составители программы:**

Зиновьев С.В., преп.

1. **Рекомендована:** НМС факультета ПММ протокол № 10 от 15.06.2019 г.

1. **Учебный год:** 2019/2020 **Семестр(ы):** 3

1. **Цели и задачи учебной дисциплины:**

Целью изучения дисциплины является знакомство с концепцией архитектуры проектирования MVC и её применением при проектировании приложений, а также получение навыков разработки веб-приложений с использованием MVCфреймворков.

1. **Место учебной дисциплины в структуре ООП:** В результате изучения дисциплины студент должен:

**Знать:** основные понятия и концепцию шаблона проектирования MVC, специфику реализации данного шаблона в зависимости от выбранного для разработки фреймворка.

**Уметь:** спроектировать и реализовать веб-приложение с использованием MVCфреймворка Ruby on Rails, используя знания об особенностях реализации MVC архитектуры в данном фреймворке.

**Владеть:** навыками практической разработки веб-приложений с использованием MVC-фреймворков.

1. **Компетенции обучающегося, формируемые в результате освоения дисциплины:**

|  |  |  |
| --- | --- | --- |
| Компетенция | | Планируемые результаты обучения |
| Код | Название |
| ОПК-6 | владение основными концептуальными положениями функционального, рекурсивного, логического, объектно-ориентированного и визуального направлений программирования, методами и средствами разработки программ в рамках этих направлений | Знать: проблемы и тенденции развития  рынка программного обеспечения,  структуру программного обеспечения,  основные виды офисных программ, методы  работы с офисными приложениями  Уметь: работать в качестве пользователя  персонального компьютера с  программными средствами общего  назначения, использовать языки и системы  программирования для решения  профессиональных задач  Владеть: навыками выбора и установки  программного обеспечения, решения  вопросов на форумах ИТ-специалистов  используя ресурсы интернет, поиска  информации в документальных системах, в  глобальных сетях |
| ОПК-10 | владение навыками использования основных моделей информационных технологий и способов их применения для решения задач в предметных областях | Знать: основные понятия методов управления компанией, разрабатывающей ПО, выделять  место проблем управления созданием качественного программного обеспечения в общих  проблемах управления,  Уметь: использовать знания о современной методологии управления качеством для формирования реальных предпосылок разработки качественного продукта;  ориентироваться в инфраструктуре проекта по разработке программных средств,  Владеть: методами формирования и применения метрик для эффективного управления  процессами, работами и процедурами разработки ПО. |
| ПК-3 | готовность применять современные методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения | Знать: основные положения различных парадигм программирования (функционального,  рекурсивного, логического, объектно-ориентированного и визуального).  Уметь: разрабатывать программы для решения задач прикладного характера из  различных  разделов  прикладной  поставленной задаче моделей,  Владеть: технологии  разработки программ в рамках этих направлений, возможные сферы их приложений при  решении практических задач, основы построения программ для систем с общей и  распределенной памятью, в том числе и для систем реального времени. |

.

1. **Объем дисциплины в зачетных единицах/часах в соответствии с учебным планом —** 3/108.

**Форма промежуточной аттестации:** зачет.

1. **Виды учебной работы**

|  |  |  |
| --- | --- | --- |
| Вид учебной работы | Трудоемкость (часы) | |
| Всего | Сем. 3 |
| Аудиторные занятия | 34 | 34 |
| в том числе: лекции | 0 | 0 |
| лабораторные | 34 | 34 |
| практические | 0 | 0 |
| Самостоятельная работа | 74 | 74 |
| Итого | 108 | 108 |

* 1. **Содержание дисциплины**

|  |  |  |
| --- | --- | --- |
| №  п/п | Наименование раздела дисциплины | Содержание раздела дисциплины |
| 1. **Лекции** | | |
| 1. **Практические занятия** | | |
| 1. **Лабораторные работы** | | |
| 3.1 | Концепция MVC. | Model, View, Controller, Router. Взаимодействие  компонентов. Различия в реализации MVC. Толстая модель, тонкий контроллер. View как шаблон и как представление. |
| 3.2 | Синтаксис языка Ruby. | Интерпретируемость и динамическая типизация языка Ruby. Классы, переменные, методы, массивы, хэши, перечисления, условные операторы, циклы. |
| 3.3 | Фреймворк Ruby on Rails. | Архитектура фреймворка, структура приложения, наследование классов. Обзор моделей, контроллеров, |
|  |  | представлений, роутера, ассетов, миграций. Построение моделей разрабатываемого приложения, выбор используемых средств для его разработки, построение логической и физической ER-диаграммы. |
| 3.4 | Среда разработки. | Настройка Unix-среды разработки с использованием Ruby on Rails. Использование систем контроля версий при проектировании и разработке приложения. Создание структуры простого приложения и его базовая конфигурация. |
| 3.5 | Модели. | Реализация сущностей в виде моделей Ruby on Rails, работа с миграциями, установление связей между моделями. Разработка валидаций для объектов моделей. |
| 3.6 | Автоматизированное тестирование. | Сущность автоматизированного тестирования. Unit-тесты, функциональные тесты, fixtures, фабрики. Реализация тестов моделей. Методология TDD. |
| 3.7 | Бизнес-логика. | Реализация бизнес-логики приложения, карты запросов, построение контроллеров, вёрстка представлений.  Написание автоматизированных тестов. |
| 3.8 | Внедрение приложения. | Подготовка приложения к процессу внедрения на production окружении. Средства автоматизации процесса внедрения. |

* 1. **Темы (разделы) дисциплины и виды занятий**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Наименование раздела дисциплины |  | Виды занятий (часов) | | |  |
| Лекции | Практичес кие | Лаборат орные | Самостоятел ьная работа | Всего |
| 1 | Концепция MVC. | 0 | 4 | 0 | 8 | 12 |
| 2 | Синтаксис языка Ruby. | 0 | 4 | 0 | 8 | 12 |
| 3 | Фреймворк Ruby on Rails. | 0 | 4 | 0 | 8 | 12 |
| 4 | Среда разработки. | 0 | 4 | 0 | 8 | 12 |
| 5 | Модели. | 0 | 4 | 0 | 8 | 12 |
| 6 | Автоматизированное тестирование. | 0 | 6 | 0 | 18 | 24 |
| 7 | Бизнес-логика. | 0 | 4 | 0 | 8 | 12 |
| 8 | Внедрение приложения. | 0 | 4 | 0 | 8 | 12 |
|  | Итого: | 0 | 34 | 0 | 74 | 108 |

1. **Методические указания для обучающихся по освоению дисциплины**

Выполнение практических заданий для самостоятельной работы, выполнение лабораторных работ, использование рекомендованной литературы и методических материалов.

1. **Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины** ) ) основная литература:

|  |  |
| --- | --- |
| № п/п | Источник |
| 1 | Фултон Х. Программирование на языке Ruby [Электронный ресурс] : справочник. — Электрон. дан. — М. : ДМК Пресс, 2009. — 689 с. — Режим доступа:  <http://e.lanbook.com/books/element.php?pl1_id=1250> |
| 2 | Астахова И.Ф. СУБД: язык SQL в примерах и задачах : учебное пособие для студ. вузов, обуч. по направлению подготовки и по специальности "Приклад. математика и информатика" .— М. : Физматлит, 2007 .— 165 с. |
| 3 | Скрипченко Ю.С. Объектно-ориентированное программирование в примерах и задачах :  учебное пособие.— Воронеж : Воронеж. гос. ун-т, 2006 .— 160 с. |

б) дополнительная литература:

|  |  |
| --- | --- |
| № п/п | Источник |
| 4 | Hartl M. Ruby on Rails Tutorial. Learn Web Development with Rails [Электронный ресурс]: . — Режим доступа: <https://www.railstutorial.org/book> |

в)информационные электронно-образовательные ресурсы**:**

|  |  |
| --- | --- |
| № п/п | Источник |
| 6 | Электронный каталог Научной библиотеки Воронежского государственного университета.  – [(http://www.lib.vsu.ru)](http://www.lib.vsu.ru/) |
| 7 | ЭБС «Издательство Лань» [http://e.lanbook.com](http://e.lanbook.com/) |
| 8 | Rails Guides [http://guides.rubyonrails.org](http://guides.rubyonrails.org/) |

1. **Перечень учебно-методического обеспечения для самостоятельной работы**

|  |  |
| --- | --- |
| № п/п | Источник |

1. **Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-**

**справочные системы (при необходимости)**

ОС Linux, Ruby

1. **Материально-техническое обеспечение дисциплины:**

Аудитория, проектор, доска

1. **Фонд оценочных средств:** 
   1. **Перечень компетенций с указанием этапов формирования и планируемых результатов обучения**

|  |  |  |  |
| --- | --- | --- | --- |
| Код и содержание компетенции (или ее  части) | Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенции посредством  формирования знаний, умений,  навыков) | Этапы формирования компетенции  (разделы (темы) дисциплины или модуля и их наименование) | ФОС\*  (средства оценивания) |
| ОПК-6 владение основными концептуальными положениями функционального, | Знать:  основные понятия и концепцию шаблона проектирования MVC; специфику реализации данного шаблона в зависимости от выбранного для разработки фреймворка | Раздел 5. Модели.  Раздел 6.  Автоматизированное тестирование. Раздел 7. Бизнеслогика.  Раздел 8. | Контрольные работы №1-2 |
| рекурсивного, логического, объектноориентированного и визуального направлений программирования, методами и средствами разработки программ в рамках этих направлений |  | Внедрение приложения. |  |
| Уметь:  спроектировать и реализовать вебприложение с использованием MVC-фреймворка Ruby on Rails, используя знания об особенностях реализации MVC архитектуры в данном фреймворке. | — « — | Контрольные работы №1-2 |
| ОПК-10 владение навыками использования основных моделей информационных технологий и способов их применения для решения задач в предметных областях | Знать: основные понятия и концепцию шаблона проектирования MVC;  специфику реализации данного шаблона в зависимости от выбранного для разработки фреймворка |  | Контрольные работы №1-2 |
| Уметь: спроектировать и реализовать вебприложение с использованием MVC-фреймворка Ruby on Rails, используя знания об особенностях реализации MVC архитектуры в данном фреймворке. |  | Контрольные работы №1-2 |
| Владеть: навыками практической разработки веб-приложений с использованием MVC-фреймворков |  | Контрольные работы №1-2 |
| ПК-3  готовность применять современные методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения | Знать: основные понятия и концепцию шаблона проектирования MVC;  специфику реализации данного шаблона в зависимости от выбранного для разработки фреймворка |  | Контрольные работы №1-2 |
| Уметь: спроектировать и реализовать вебприложение с использованием MVC-фреймворка Ruby on Rails, используя знания об особенностях реализации MVC архитектуры в данном фреймворке. |  | Контрольные работы №1-2 |
| Владеть: навыками практической разработки веб-приложений с использованием MVC-фреймворков |  | Контрольные работы №1-2 |
| **Промежуточная аттестация** | |  | Комплект КИМ №1 |

.

## 19.2 Описание критериев и шкалы оценивания компетенций (результатов обучения) при промежуточной аттестации

**Пример:**

Для оценивания результатов обучения на экзамене/зачете используются следующие показатели:

1. владение навыками реализации программ;
2. знание методологии структурного программирования и применение ее на практике;
3. знание и умение использовать при решении задач базовые структуры данных;
4. умение выбирать и применять при реализации оптимальную структуру данных;
5. умение разрабатывать оптимальный алгоритм решения задачи и выполнять его реализацию;

4) знание теоретического материала

Для оценивания результатов обучения на зачете используется 2-балльная шкала: «зачтено», «не

зачтено».

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

|  |  |  |
| --- | --- | --- |
| Критерии оценивания компетенций | Уровень сформирован ности компетенций | Шкала оценок |
| Знание основного учебно-программного материала в объёме, необходимом для дальнейшей учёбы и предстоящей работы по специальности, выполнение заданий, предусмотренных программой. | *Пороговый уровень* | *Зачтено* |
| Имеются пробелы в знаниях основного учебно-программного материала, принципиальные ошибки в выполнении предусмотренных программой заданий, наличие которых препятствует дальнейшему обучению студента. | *-* | *Не зачтено* |

## 19.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы

### 19.3.1 Перечень практических заданий

Иллюстрируется на примере КИМ1-3 **19.3.2 Тестовые задания**

Иллюстрируется на примере КИМ1-3

#### ПРИМЕРЫ КИМ 1

|  |  |
| --- | --- |
| **Задача** | **Блог.** Все зарегистрированные пользователи могут создать себе блог и постить в него блогпосты. Есть Admin пользователь, который может модерировать посты и блоги, давать админские права другим пользователям. |
|  | **КИМ2** |
| **Задача** | **Калькулятор.** Реализовать функции + - /, извлечение квадратного корня и несколько тригонометрических функций на своё усмотрение. Нет модели, базы, сессий. |

#### КИМ3

|  |  |
| --- | --- |
| **Задача** | **Интернет-магазин.** Администратор может редактировать, добавлять товары. У каждого товара есть описание, цена, название, категория, количество на складе. Администратор может редактировать, добавлять категории. Зарегистрированные пользователи могут совершать покупки, добавляя их себе в корзину (способ оплаты нас не интересует). Корзина хранится в базе. Должна быть предусмотрена возможность очистки корзины. |

**19.3.4 Перечень заданий для контрольных работ**

Иллюстрируется на примерах заданий для контрольных работ 1-2

## Примеры заданий для контрольной работы № 1

### Вариант № 1

В одном массиве записан рост некоторых студентов, а в другом (с тем же числом элементов) - их фамилии в том же порядке, в котором указан рост. Известно, что все студенты разного роста. Напечатайте фамилию самого высокого студента.

### Вариант № 2

Даны две строки. Определите, сколько начальных символов первой строки совпадает с начальными символами второй. Рассмотрите два случая: а) известно, что строки разные; б) строки могут совпасть.

**Критерии оценки:**

* оценка «отлично» выставляется студенту, если правильно решена задача и реализованы автоматизированные тесты (возможно с небольшими недочетами);
* оценка «хорошо» выставляется студенту, если задача решена правильно (возможно с небольшими недочетами); в реализации половины автоматизированных тестов допущены ошибки;
* оценка «удовлетворительно» выставляется студенту, если задача решена с ошибками реализации; в реализации половины автоматизированных тестов допущены ошибки
* оценка «неудовлетворительно» выставляется студенту, если не выполняются вышеуказанные критерии оценки.

## Примеры заданий для контрольной работы № 2

### Вариант № 1

Реализовать API, удовлетворяющий следующим тестам:

describe "GET /users" do

describe "for not logged in users" do it "should not show Users page" do get users\_path

response.status.should == 403 end end

describe "for logged in regular users" do before { login\_user FactoryGirl.create(:user) } it "should not show Users page" do get users\_path

response.status.should == 403 end end

describe "for logged in users as admins" do before { login\_user FactoryGirl.create(:admin) } it "should show Users page" do get users\_path

response.status.should == 200 end end end def login\_user(user) visit sign\_in\_path

fill\_in "Email", with: user.email fill\_in "Password", with: user.password click\_button 'Sign in' end

### Вариант № 2

Реализовать API, удовлетворяющий следующим тестам:

describe "DELETE /users/:id" do

 before(:each) { @user = FactoryGirl.create(:user) } it "not logged in users should not delete user" do delete "/users/#{@user.id}" response.status.should == 403 end

it "logged in as regular users should not delete user" do delete "/users/#{@user.id}"  response.status.should == 403 end

it "admins can delete user" do login\_user FactoryGirl.create(:admin) delete "/users/#{@user.id}" response.status.should == 302 end end def login\_user(user) visit sign\_in\_path

fill\_in "Email", with: user.email fill\_in "Password", with: user.password click\_button 'Sign in' end

**Критерии оценки:**

* оценка «отлично» выставляется студенту, если правильно решена задача;
* оценка «хорошо» выставляется студенту, если задача решена правильно

(возможно с небольшими недочетами);

* оценка «удовлетворительно» выставляется студенту, если при решении задачи правильно составлен алгоритм, а в реализации допущены ошибки;
* оценка «неудовлетворительно» выставляется студенту, если не выполняются вышеуказанные критерии оценки.

## 19.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в форме контрольных работ.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Контрольно-измерительные материалы промежуточной аттестации включают в себя практические задания, позволяющие оценить степень сформированности умений и навыков. При оценивании используются количественные шкалы оценок. Критерии оценивания приведены выше.