

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.

8.06.2018 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.10 Использование фреймворков при разработке приложений

1. Код и наименование направления подготовки/специальности:

02.03.03. Математическое обеспечение и администрирование информационных систем

2. Профиль подготовки/специализация:

Информационные системы и базы данных

3. Квалификация (степень) выпускника: бакалавр

4. Форма обучения: очная

5. Кафедра, отвечающая за реализацию дисциплины: Программного обеспечения и администрирования информационных систем

6. Составители программы: Зиновьев С.В., преподаватель кафедры программного обеспечения и администрирования информационных систем

7. Рекомендована: НМС факультет ПММ № 10 от 18.06.2018

8. Учебный год: 2018/2019

Семестр(ы): 6

9.Цели и задачи учебной дисциплины: Целью изучения дисциплины является знакомство с принципами разработки современных приложений, проектирование их архитектуры, а также ускорение процесса разработки базовых модулей приложения ввиду использования фреймворков.

10. Место учебной дисциплины в структуре ООП: Дисциплина «Использование фреймворков при разработке приложений» входит в базовую часть программы бакалавриата, изучается в 6 семестре и является предшествующей для следующих дисциплин:

- Сервис-ориентированные архитектуры
- Шаблоны проектирования
- Проектирование пользовательских интерфейсов

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников):

Компетенция		Планируемые результаты обучения
Код	Название	
ОПК-7	Способность использовать знания основных концептуальных положений функционального, логического, объектно-ориентированного и визуального направлений программирования, методов, способов и средств разработки программ в рамках этих направлений	<p>знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения;</p> <p>уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git;</p> <p>владеть (иметь навык(и)): теоретическими и практическими навыками разработки приложений с использованием фреймворков.</p>
ОПК-8	Способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения	<p>знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения;</p> <p>уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git;</p> <p>владеть (иметь навык(и)): навыками практической разработки приложений с использованием фреймворков и систем контроля версий.</p>
ОПК-11	Готовность использовать навыки выбора,	<p>знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа</p>

	проектирования, реализации, оценки качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях	<p>разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения;</p> <p>уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git;</p> <p>владеть (иметь навык(и)): навыками практической разработки приложений с использованием фреймворков и систем контроля версий.</p>
ПК-2	Готовность к использованию основных моделей информационных технологий и способов их применения для решения задач в предметных областях	<p>знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения;</p> <p>уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git;</p> <p>владеть (иметь навык(и)): навыками практической разработки приложений с использованием фреймворков и систем контроля версий.</p>

12. Объем дисциплины в зачетных единицах/час.(в соответствии с учебным планом) — 2 / 72.

Форма промежуточной аттестации(зачет/экзамен) Зачет с оценкой

13. Виды учебной работы

Вид учебной работы	Трудоемкость	
	Всего	По семестрам
		Сем. 6
Аудиторные занятия	32	32
в том числе: лекции	0	0
практические	16	16
лабораторные	16	16
Самостоятельная работа	40	40
Форма промежуточной аттестации (зачет – 0 час. / экзамен – 36час.)	0	Зачет с оценкой
Итого:	72	72

13.1. Содержание дисциплины

п/п	Наименование раздела	Содержание раздела дисциплины
-----	----------------------	-------------------------------

дисциплины		
1. Практические занятия		
1.1	Введение. Синтаксис языка Ruby	Определения понятия фреймворков. Краткий обзор существующих фреймворков. Интерпретируемость и динамическая типизация языка Ruby. Классы, переменные, методы, массивы, хэши, перечисления, условные операторы, циклы.
1.2	Фреймворк Ruby on Rails.	Архитектура фреймворка, структура приложения, наследование классов. Обзор моделей, контроллеров, представлений, роутера, ассетов, миграций.
1.3	Моделирование приложения.	К Построение моделей разрабатываемого приложения, выбор используемых средств для его разработки, построение логической и физической ER-диаграммы.
1.4	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.
1.5	Модели.	Реализация сущностей в виде моделей Ruby on Rails, работа с миграциями, установление связей между моделями. Разработка валидаций для объектов моделей.
1.6	Автоматизированное тестирование.	Сущность автоматизированного тестирования. Unit-тесты, функциональные тесты, fixtures, фабрики. Реализация тестов моделей. Методология TDD.
1.7	Бизнес-логика.	Реализация бизнес-логики приложения, карты запросов, построение контроллеров, вёрстка представлений. Написание автоматизированных тестов. Идентификация, авторизация и аутентификация.
1.8	Внедрение приложения.	Подготовка приложения к процессу внедрения на production окружении. Средства автоматизации процесса внедрения.
2. Лабораторные работы		
2.1	Разработка приложения на скриптовом языке	Выбрать подход к решению задачи, структуры данных, используемые для её решения, реализовать скрипт, принимающий входные параметры и выводящий результат выполнения задачи на экран
2.2	Разработка web-приложения, использующего БД	Выбрать подход к решению поставленной задачи, выбрать используемые технологии, спроектировать БД, реализовать web-приложение, позволяющее просматривать, фильтровать, искать, добавлять, редактировать, удалять информацию из БД

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1	Введение. Синтаксис языка Ruby.	0	2	2	2	6
2	Фреймворк Ruby on Rails.	0	2	2	3	7
3	Моделирование приложения.	0	2	2	3	7
4	Среда разработки.	0	2	2	2	6
5	Модели.	0	2	2	6	10
6	Автоматизированное тестирование.	0	2	2	8	12
7	Бизнес-логика.	0	2	2	8	12
8	Внедрение приложения.	0	2	2	8	12
	Итого:	0	16	16	40	72

14. Методические указания для обучающихся по освоению дисциплины

(рекомендации обучающимся по освоению дисциплины: работа с конспектами лекций, презентационным материалом, выполнение практических заданий, тестов, заданий текущей аттестации и т.д.)

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

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

№ п/п	Ресурс
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	Hartl M. Ruby on Rails Tutorial. Learn Web Development with Rails [Электронный ресурс]: . — Режим доступа: https://www.railstutorial.org/book
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	Ресурс
4	Блюстайн М. Изучаем MonoTouch. Создание приложений на платформе iOS с помощью C# и .NET [Электронный ресурс] : . — Электрон. дан. — М. : ДМК Пресс, 2012. — 336 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=9129
5	Гаршина В.В., Сапегин С.В. Основы языка PL/SQL : учебно-методическое пособие для вузов / Воронеж. гос. ун-т; сост.: В.В. Гаршина, С.В. Сапегин .— Воронеж : ИПЦ ВГУ, 2008 .— 55 с.— Режим доступа: http://www.lib.vsu.ru/elib/texts/method/vsu/m08-97.pdf

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

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

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных) работ и др.)

17. Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-справочные системы (при необходимости)

ОС Linux, Ruby

18. Материально-техническое обеспечение дисциплины:

(при использовании лабораторного оборудования указывать полный перечень, при большом количестве оборудования можно вынести данный раздел в приложение к рабочей программе)

Аудитория, доска, проектор.

19. Фонд оценочных средств:

19.1. 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)	Этапы формирования компетенции	ФОС* (средства)
---	---	--------------------------------	-----------------

	компетенции посредством формирования знаний, умений, навыков)	(разделы (темы) дисциплины или модуля и их наименование)	оценивания)
<p>ОПК-7 способность использовать знания основных концептуальных положений функционального, логического, объектно-ориентированного и визуального направлений программирования, методов, способов и средств разработки программ в рамках этих направлений</p>	<p>Знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения</p>	<p>Раздел 5. Модели. Раздел 6. Автоматизированное тестирование. Раздел 7. Бизнес-логика. Раздел 8. Внедрение приложения.</p>	<p>Контрольные работы №1-2</p>
	<p>Уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git.</p>	<p>— « —</p>	<p>Контрольные работы №1-2</p>
<p>ОПК-8 способность использовать знания методов проектирования и производства программного продукта</p>	<p>Знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения</p>		<p>Контрольные работы №1-2</p>
	<p>Уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git.</p>		<p>Контрольные работы №1-2</p>
	<p>Владеть: навыками практической разработки приложений с использованием фреймворков и систем контроля версий</p>		<p>Контрольные работы №1-2</p>
<p>ОПК-11 готовность использовать навыки выбора, проектирования, реализации, оценки</p>	<p>Знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов</p>		<p>Контрольные работы №1-2</p>

качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях	проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения		
	Уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git.		Контрольные работы №1-2
	Владеть: навыками практической разработки приложений с использованием фреймворков		Контрольные работы №1-2
ПК-2 готовность к использованию основных моделей информационных технологий и способов их применения для решения задач в предметных областях	Знать: основные паттерны проектирования, используемые для построения архитектуры приложений; специфику методов проектирования в зависимости от типа разрабатываемого приложения; фреймворки, используемые для реализации спроектированного приложения		Контрольные работы №1-2
	Уметь: обосновать целесообразность использования конкретного шаблона проектирования; спроектировать и реализовать приложение с использованием фреймворка Ruby on Rails и системы контроля версий Git.		Контрольные работы №1-2
	Владеть: навыками практической разработки приложений с использованием фреймворков		Контрольные работы №1-2
Промежуточная аттестация			Комплект КИМ №1

* В графе «ФОС» в обязательном порядке перечисляются оценочные средства текущей и промежуточной аттестаций.

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

Для оценивания результатов обучения на зачете используются следующие показатели:

- 1) теоретические знания паттернов проектирования и фреймворков, используемых для разработки приложений;
- 2) практические навыки выбора и использования технологий для реализации лабораторных задач.

Для оценивания результатов обучения на зачете с оценкой используется 4-балльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Отличное знание теоретического материала, правильное и эффективное решение задачи, правильные ответы на тестовые вопросы. Должны быть выполнены ВСЕ лабораторные работы по дисциплине <i>Использование фреймворков при разработке приложений</i>	<i>Повышенный уровень</i>	<i>Отлично</i>
Хорошее знание теоретического материала, в целом правильное решение задачи. НО: допускает незначительные ошибки в решении задачи. Неправильный ответ на тест. ИЛИ: выполнены все показатели повышенного уровня, но не зачтена одна задача из дисциплины <i>Использование фреймворков при разработке приложений</i> , но студент продемонстрировал умение решать задачи по этой теме (это задача в КИМе)	<i>Базовый уровень</i>	<i>Хорошо</i>
Решение задачи не доведено до конца или недостаточное знание теоретического материала, ошибки в тестах, неоптимальное решение задачи и недостаточное владение теоретическим материалом. ИЛИ: выполнены все показатели базового уровня, но по дисциплине <i>Использование фреймворков при разработке приложений</i> не зачтены более одной задачи.	<i>Пороговый уровень</i>	<i>Удовлетворительно</i>
Задача не решена или серьезные пробелы в знании теоретического материала (с незнанием могут быть связаны и грубые ошибки в ответе на тестовые вопросы)	–	<i>Неудовлетворительно</i>

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 вопросов к зачету:

Иллюстрируется на примере КИМ1-3

19.3.2 Перечень практических заданий

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
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
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 теоретические вопросы, позволяющие оценить степень сформированности умений и

навыков. Практические навыки определяются в ходе проверки выполнения лабораторных работ.

При оценивании используются количественные шкалы оценок. Критерии оценивания приведены выше.