

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Воронежский государственный технический университет»

УТВЕРЖДАЮ  
Декан факультета экономики менеджмента и  
информационных технологий  
С.А.Баркалов  
«30» августа 2017 г

**РАБОЧАЯ ПРОГРАММА**

дисциплины

«Методы разработки программных систем»

**Направление подготовки** 09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И  
ТЕХНОЛОГИИ

**Профиль** Информационные системы и технологии строительстве


**Квалификация выпускника** Бакалавр

**Нормативный период обучения** 4 года

**Форма обучения** очная

**Год начала подготовки** 2017

Автор программы

 /Ошивалов А.В./

Заведующий кафедрой  
Информационных технологий  
и автоматизированного  
проектирования в  
строительстве

 /Смолянинов А.В./

Руководитель ОПОП

 /Курипта О.В./

Воронеж 2017

## 1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

### 1.1. Цели дисциплины

дать специальную подготовку, формирующую способность правильно воспринимать, перерабатывать и воспроизводить информацию; показать место программирования в промышленности; познакомить с базовыми понятиями и приемами современных методов проектирования ИС и методами творческого решения конструкторских и инженерных задач.

### 1.2. Задачи освоения дисциплины

- раскрытие методов представления информации в условиях программирования;
- изучение способов и приемов представления информации на основе моделирования и программирования;
- знакомство с современной идеологией цифрового прототипирования будущих изделий.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Методы разработки программных систем» относится к дисциплинам базовой части блока Б1.

## 3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Методы разработки программных систем» направлен на формирование следующих компетенций:

ОПК-3 - способностью применять основные приемы и законы создания и чтения чертежей и документации по аппаратным и программным компонентам информационных систем

ПК-25 - готовность использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-3	Знать основы методов разработки программных систем
	Уметь описывать состояние архитектуры информационных систем
	Владеть навыками описания архитектуры информационных систем
ПК-25	Знать основы программирования и моделирования
	Уметь применять интерактивные системы для выполнения прикладных задач
	Владеть современными программными средствами моделирования и программирования, навыками подготовки информации.

## 4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Методы разработки программных систем»

систем» составляет 5 з.е.

**Распределение трудоемкости дисциплины по видам занятий  
очная форма обучения**

Виды учебной работы	Всего часов	Семестры
		3
<b>Аудиторные занятия (всего)</b>	54	54
В том числе:		
Лекции	18	18
Лабораторные работы (ЛР)	36	36
<b>Самостоятельная работа</b>	126	126
<b>Курсовой проект</b>	+	+
Виды промежуточной аттестации - зачет с оценкой	+	+
Общая трудоемкость: академические часы зач.ед.	180 5	180 5

**5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

**5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий  
очная форма обучения**

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Моделирование ИС	Структура однопользовательской и многопользовательской, малой и корпоративной ИС, локальной и распределенной ИС, состав и назначение подсистем. Методы программной инженерии в проектировании ИС. Требования к эффективности и надежности проектных решений	6	12	44	62
2	Проектирование ИС	Методы и средства проектирования ИС. Формализация технологии проектирования ИС. Выбор технологии проектирования	6	8	32	46
3	Основы программирования ИС	Парадигмы программирования. Технология создания программного кода. Технологические средства разработки программного обеспечения	6	16	50	72
<b>Итого</b>			<b>18</b>	<b>36</b>	<b>126</b>	<b>180</b>

**5.2 Перечень лабораторных работ**

Лабораторная работа №1. «Анализ объекта автоматизация. Написание технического задания на разработку ИС»

Лабораторная работа №2. «Анализ объекта автоматизация. Написание этапов рабочего проекта ИС»

Лабораторная работа №3. «Методы проектирования программного

обеспечения» Лабораторная работа №4. «Методы проектирования аппаратного обеспечения» Лабораторная работа №5. «Библиотеки стандартных компонентов» Лабораторная работа №6. «Библиотеки объектов» Лабораторная работа №7. «Инструментальная среда разработки» Лабораторная работа №8. «Библиотека VCL.»  
Лабораторная работа №9. «Средства поддержки проекта. Отладчики.»

## **6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ**

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 3 семестре для очной формы обучения.

Примерная тематика курсового проекта: «Проектирования и разработка программной системы по заданной предметной области»

Задачи, решаемые при выполнении курсового проекта:

1. Составить развёрнутое описание поставленной задачи;
2. Выполнить анализ задания, выбрать модель решения задачи (алгоритм, формулы), обосновать структуру входных и выходных данных;
3. Разработать программу, отладить ее и подготовить документацию к ней.

Варианты тем курсовой работы.

1. Разработка информационной системы медицинского кабинета.
2. Разработка автоматизированного рабочего места секретаря.
3. ИС торгового предприятия.
4. Разработка автоматизированной инвентаризационной книги.
5. Разработка автоматизированного рабочего места библиотекаря.
6. ИС "Кадровое агентство".
7. Разработка автоматизированной информационной системы «Билетные кассы».
8. Разработка автоматизированной информационной системы складского учета.
9. ИС "Инвентаризационный учет оргтехники и ПК"
10. Разработка электронного магазина.
11. Разработка автоматизированной информационной системы «Деловые бумаги».

Курсовой проект включает в себя графическую часть и расчетно-пояснительную записку.

## **7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

**7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания**

### **7.1.1 Этап текущего контроля**

Результаты текущего контроля знаний и межсессионной аттестации

оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ОПК-3	Знать основы методов разработки программных систем	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Уметь описывать состояние архитектуры информационных систем	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Владеть навыками описания архитектуры информационных систем	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ПК-25	Знать основы программирования и моделирования	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Уметь применять интерактивные системы для выполнения прикладных задач	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	Владеть современными программными средствами моделирования и программирования, навыками подготовки информации.	Выполненные лабораторные работы и курсовая работа	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

### 7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 3 семестре для очной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ОПК-3	Знать основы методов разработки программных систем	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	Уметь описывать состояние архитектуры информационных систем	Решение стандартных практических задач	Задачи решены в полном объеме и получены	Продемонстрирован верный ход решения всех, но не получен	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

			верные ответы	верный ответ во всех задачах		
	Владеть навыками описания архитектуры информационных систем	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ПК-25	Знать основы программирования и моделирования	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	Уметь применять интерактивные системы для выполнения прикладных задач	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	Владеть современными программными средствами моделирования и программирования, навыками подготовки информации.	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

## 7.2 Примерный перечень оценочных средств (типичные контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

### 7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Три принципа объектно-ориентированного программирования

- **Наследование;**
- Перегрузка методов
- **Инкапсуляция;**
- Рефакторинг;
- **Полиморфизм;**

2. У класса А есть свойство с модификатором private. Будет ли доступно данное свойство у наследника данного класса

- да
- **нет**

только если класс имеет модификатор protected

3. Какой модификатор для свойств и методов задает область видимости «только для данного класса»

- protected
- **private**
- final

4. Какой модификатор для свойств и методов задает область видимости «для

всех»

- protected
  - private
  - **public**
5. Какой модификатор для свойств и методов задает область видимости «только для наследников данного класса и классов в том же пакете»
- **protected**
  - private
  - static
6. Какой модификатор для свойств запрещает изменение значения после создания экземпляра класса
- **final**
  - private
  - static
7. Какой модификатор для методов запрещает изменение (перегрузку) методов у наследников
- **final**
  - private
  - static
8. Какой модификатор для свойств и методов позволяет обращаться к ним без создания экземпляра класса
- protected
  - private
  - **static**
9. К какому системному классу относится класс out
- **System**
  - Runtime
  - java.util
10. Встроенные классы Java для работы со строками
- **String**
  - **StringBuffer**
  - StringArray
  - **StringBuilder**
11. Класс, рекомендуемый Java для работы с датами и временем
- **Calendar**
  - Date
  - Time
  - DateTime
11. Что произойдет при добавлении элемента в ArrayList, в случае если он заполнен до конца
- ошибка
  - будет заменен последний элемент
  - **будет увеличен размер на несколько элементов и добавлен новый**

### после последнего заполненного

- будет увеличен размер на 1 элемент и добавлен новый на его место

12. Какие из отношений допустимы на диаграмме вариантов использования:

- наследования (Generalization) между акторами;
- расширения между вариантами использования; в) ассоциации между акторами;
- агрегации между акторами.

### 7.2.2 Примерный перечень заданий для решения стандартных задач

1. Постройте UML-диаграммы классов и последовательностей для приведенного ниже программного кода:

```
public class Product
{
    public string Name { get; set; }
    public string GetPrice(ShopKeeper keeper)
    {
        var name = keeper.GetName();
        return name + ", Price is low";
    }
}

public class ShopKeeper
{
    public bool Take(Product p)
    {
        p.GetPrice(this);
        return true;
    }

    public string GetName()
    {
        return "John";
    }
}

class Program
{
    static void Main(string[] args)
    {
        ShopKeeper sh = new ShopKeeper();
        Product pr = new Product();
        pr.Name = "Food";
        sh.Take(pr);
    }
}
```

2. Укажите назначение и содержание документа Product Backlog, используемого в методологии SCRUM
3. Постройте диаграмму DFD веб-приложения для ведения блогов.



На диаграмме должно быть не менее трех накопителей данных, не менее пяти процессов и не менее двух внешних сущностей.

4. Укажите название, назначение и участников встреч, которые проводятся при разработке программных приложений по методологии SCRUM.

5. Постройте граф потока управления для приведенного ниже программного кода. Создайте тест-кейсы с покрытием ветвей.

```
bool b1 = Convert.ToBoolean(Console.ReadLine());
bool b2 = Convert.ToBoolean(Console.ReadLine());
if (b1 == true)
{
    Console.WriteLine("true");
    if (b2 == false)
    {
        Console.WriteLine("not b2");
    }
    else
    {
        Console.WriteLine("b2");
    }
}
```

### 7.2.3 Примерный перечень заданий для решения прикладных задач

1. Создать базовый объект со свойствами — воздушный поток (куб.м./мин.), потери в трубопроводе (% / м) и методами для расчета объема подаваемого воздуха в час при заданной длине трубопровода. Создать объект на основе базового со свойствами — наименование, объем и методом для расчета времени заполнения.

2. Создать базовый объект со свойствами — баллы ЕГЭ по математике, русскому, информатике и методами для расчета среднего балла и суммарного балла. Создать объект на основе базового со свойствами — фио, профильный предмет и методами для поиска абитуриента с максимальным средним баллом.

3. Создать базовый объект со свойствами — баллы ЕГЭ по математике, русскому, информатике и методами для расчета среднего балла и суммарного балла. Создать объект на основе базового со свойствами — фио, профильный предмет и методами для поиска абитуриента с максимальным суммарным баллом.

4. Создать базовый объект со свойствами — баллы ЕГЭ по математике, русскому, информатике и методами для расчета среднего балла и суммарного балла. Создать объект на основе базового со свойствами — фио, профильный предмет и методами для поиска абитуриента с максимальным баллом по профильному предмету.

5. Создать базовый объект со свойствами — ширина, высота, толщина и методами для расчета объема. Создать объект на основе базового

со свойствами — наименование, объем, коэффициент заполнения (1.00 .. 3.00) и методами для расчета количества предметов, помещающихся в базовом объеме.

6 Создать объект с данными о сдаче студентами сессии. Есть информация об имени, фамилии, отчестве студента, названии предмета (математика, физика, черчение), оценке по предмету. Определить, средний балл по каждому предмету.

7. Создать объект с данными содержащими сведения о заработной плате: фамилия, имя, отчество, должность (инженер, конструктор, лаборант), оклад. Определить, средний оклад для каждой должности.

8. На выставке собак составляется анкета, содержащая сведения о породе (боксер, терьер, такса, пудель, доберман, овчарка), возрасте, высоте холки. Определить сколько собак и какой породы имеют одинаковую высоту холки.

#### **7.2.4 Примерный перечень вопросов для подготовки к зачету**

1. Что такое технология разработки ПО?
2. Что явилось предпосылкой становления дисциплины «Технология разработки ПО»? Что явилось причиной стремительного развития ПО?
3. Чем отличаются программа и программное обеспечение?
4. Достаточно ли при работе над проектом большой программной системы быть компетентным в области вычислительной техники и программировании. Почему?
5. Может ли большая программная система быть отлажена до конца и почему?
6. Что представляют собой утилитарные программы?
7. При каких условиях созданный программный комплекс может быть назван программным продуктом?
8. Чем отличаются shareware-программы от freeware-программ?
9. Что такое системное программное обеспечение?
10. Что такое инструментальный технологии программирования?
11. Что такое жизненный цикл ПО?
12. Каковы основные этапы разработки ПО (классический жизненный цикл)?
13. Какие мероприятия выполняются на этапе анализа? Что является результатом этого этапа?
14. Какие мероприятия выполняются на этапе проектирования? Что является результатом этого этапа?
15. Какие мероприятия выполняются на этапах кодирования, тестирования и отладки? Что является результатом этих этапов?
16. Какие мероприятия выполняются на этапе внедрения? Что является результатом этого этапа?
17. Какие мероприятия выполняются на этапе сопровождения? Что является результатом этого этапа?
18. Чем отличается классический жизненный цикл от реального

жизненного цикла ПО?

19. Каковы достоинства и недостатки классического жизненного цикла?

20. Каковы достоинства и недостатки макетирования?

21. Какие существуют стратегии разработки ПО? Чем они отличаются?

22. Что представляет собой инкрементная модель?

23. Что представляет собой спиральная модель?

24. Каковы достоинства и недостатки спиральной модели?

25. Чем отличаются спиральная и компонентно-ориентированная модели?

26. Каковы достоинства компонентно-ориентированной модели?

27. Что общего и чем отличаются модели: каскадная, спиральная, MSF?

28. Что такое вехи?

29. Одинакова ли длительность фаз в MSF и может ли деятельность фазы выходить за границы этой фазы? Почему?

30. Что такое единое видение проекта? Почему необходимо его достигать?

31. Допускает ли модель MSF изменения условий проекта? Почему?

32. Каковы достоинства свободного обмена информацией внутри команды и с заинтересованными лицами?

33. Какая фаза является нововведением модели MSF по сравнению со спиральной моделью?

34. Что представляет собой точка конвергенции?

35. Что представляет собой точка достижения нуля?

36. Чем отличаются понятия заказчик и потребитель (пользователь)?

37. Что представляют собой заинтересованные стороны в модели MSF?

38. Насколько важна вовлеченность заказчика в проект при использовании модели MSF и почему?

39. Чем отличаются единое видение проекта и рамки проекта?

40. Что представляет собой треугольник компромиссов?

41. Что представляет собой матрица компромиссов?

42. Что такое экстремальное программирование? Каковы его основные характеристики?

43. Что такое парное программирование? Каковы его достоинства?

44. Как происходит развитие компании в соответствии с моделью зрелости процесса конструирования ПО?

45. Что понимается под надежностью ПО?

46. В каком случае люди и месяцы полностью взаимозаменяемы?

47. В каком случае добавление дополнительных людей не влияет на срок выполнения задачи?

48. Что можно предпринять при отставании от графика?

49. Как достигается образование «семьи» в японском менеджменте?

50. К чему может привести недооценка стоимости программного проекта?
51. К чему может привести переоценка стоимости программного проекта?
52. Какие есть способы оценки затрат программного проекта?
53. Почему следует написать требования к программному обеспечению?
54. Какие есть способы выразить требования заказчика?
55. Чем отличаются структурный и объектно-ориентированный подходы к проектированию ПО?
56. Что представляет собой метод SADT?
57. Какие существуют виды дуг в методе IDEF0?
58. Что представляет собой метод DFD?
59. Каковы основные компоненты DFD?
60. Что представляет собой метод ERD?
61. Как разрешаются связи «многие ко многим»?
62. Чем отличаются диаграмма последовательности и кооперативная диаграмма?
63. Чем отличаются диаграммы взаимодействия и диаграммы состояний?
64. Что представляет собой связь обобщения на диаграмме классов?
65. Что представляет собой связь агрегации на диаграмме классов?
66. Что представляет собой диаграмма реализации?
67. Что такое основной и альтернативный поток событий?
68. В чем преимущество использования единого стиля программирования?
69. Каковы преимущества редактора, встроенного в интегрированную систему, по сравнению с обычным текстовым редактором?

### **7.2.5 Примерный перечень вопросов для подготовки к экзамену**

*Не предусмотрено учебным планом*

### **7.2.6. Методика выставления оценки при проведении промежуточной аттестации**

*Зачет проводится по билетам, каждый из которых содержит 2 вопроса и 2 задачи. Каждый правильный ответ на вопрос оценивается 5 баллами, задача оценивается в 5 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20. Для получения зачета по дисциплине необходимо набрать минимум 10 баллов.*

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.
2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 6 до 10 баллов
3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.
4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов.)

### 7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Моделирование ИС	ОПК-3, ПК-25	Тест, защита лабораторных работ, требования к курсовому проекту, зачет
2	Проектирование ИС	ОПК-3, ПК-25	Тест, защита лабораторных работ, требования к курсовому проекту, зачет
3	Основы программирования ИС	ОПК-3, ПК-25	Тест, защита лабораторных работ, требования к курсовому проекту, зачет

### 7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методики выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется на лабораторных работах под руководством преподавателя. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется самостоятельно в течении недели. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Защита курсового проекта или отчета по всем видам практик осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

## 8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 8.1 Перечень учебной литературы, необходимой для освоения дисциплины

Орлов С. А. Теория и практика языков программирования [Текст] : учебник. – Москва ; Санкт-Петербург ; Нижний Новгород [и др.] : Питер, 2014 (Чехов : Первая Образцовая тип., фил. "Чеховский Печатный Двор", 2014). - 688 с.

Кулямин, В. В. *Технологии программирования. Компонентный подход [Электронный ресурс]* / В. В. Кулямин. — 2-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 590 с. — 5-9556-0067-1. — Режим доступа: <http://www.iprbookshop.ru/73733.html>

Ковалевская Е.В. *Методы программирования [Электронный ресурс]: учебное пособие* / Ковалевская Е.В., Комлева Н.В.— Электрон.текстовые данные.— М.: Евразийский открытый институт, 2011.— 320 с.— Режим доступа: <http://www.iprbookshop.ru/10784>

Леоненков, А. В. *Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose. Курс лекций [Электронный ресурс]* : учебное пособие для студентов вузов, обучающихся по специальностям в области информационных технологий / А. В. Леоненков. — Электрон. текстовые данные. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 318 с. — 978-5-4487-0081-1. — Режим доступа: <http://www.iprbookshop.ru/67388.html>.

**8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:**

Программное обеспечение

Среда разработки IntelliJ IDEA 2016

Java SE SDK, Netbeans.

MICROSOFT VISUAL STUDIO

Информационно-справочные системы

1. Сайт ixbt.com [www.ixbt.com](http://www.ixbt.com) Полная оперативная и объективная информация о персональных компьютерах, их компонентах и периферийных устройствах

2. Сайт CITForum [www.citforum.ru](http://www.citforum.ru) Библиотека технических материалов по информационным технологиям

3. Сайты поддержки разработчиков ПО [www.eclipse.com](http://www.eclipse.com), [www.java.com](http://www.java.com) Справочная техническая документация по среде разработки Eclipse и поддержки языка программирования Java

4. Комитет по стандартизации в области радиоэлектроники и вычислительной техники [www.ieee.org](http://www.ieee.org) Нормативно-справочная документация по вычислительной технике

5. Программная инженерия <http://www.software-engin.com/>  
<http://www.cs.st-andrews.ac.uk> Авторские обзоры по современным тенденциям в инженерии ПО, обновление глав учебника «Программная инженерия»

<http://netbeans.org/>

<http://www.oracle.com/>

## **9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ**

## ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

*Компьютерный класс с мультимедийными оборудованием и компьютерами с выходом в Интернет и предустановленной средой разработки приложений*

### 10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Методы разработки программных систем» читаются лекции, проводятся лабораторные работы, выполняется курсовой проект.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на ЭВМ в соответствии с методиками, приведенными в указаниях к выполнению работ.

Методика выполнения курсового проекта изложена в учебно-методическом пособии. Выполнять этапы курсового проекта должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины производится проверкой курсового проекта, защитой курсового проекта.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к	Готовиться к промежуточной аттестации следует систематически, в

промежуточной аттестации	течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед зачетом с оценкой три дня эффективнее всего использовать для повторения и систематизации материала.
--------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------