

ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ»
ФИЗИКО-ТЕХНИЧЕСКИЙ ФАКУЛЬТЕТ
Кафедра Компьютерных технологий

УТВЕРЖДАЮ:

проректор по научно-методической

учебной работе

Е.И. Скафа

«22» апреля 2020 г.



РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
«ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ»

Направление подготовки: **09.03.01 Информатика и вычислительная техника**

Профиль подготовки: **Информатика и вычислительная техника**

Образовательная программа: **бакалавриат**

Квалификация: **академический бакалавр**

Форма обучения: **очная, очно-заочная, заочная, в том числе с ускоренным сроком обучения**

Донецк 2020



УТВЕРЖДАЮ:

Декан физико-технического факультета
 _____ Фоменко С.А.

«17» апреля 2020 г.

М.П.

Программа учебной дисциплины «Технологии разработки программного обеспечения» составлена на основе Государственного образовательного стандарта высшего профессионального образования (ГОС ВПО) по направлению подготовки 09.03.01 Информатика и вычислительная техника, утверждённого приказом Министерства образования и науки ДНР от «21» января 2016 г. №31»; «Порядка организации учебного процесса в образовательных организациях высшего профессионального образования Донецкой Народной Республики», утверждённого приказом Министерства образования и науки ДНР №1171 от «10» ноября 2017 г.»; учебного плана и основной образовательной программы высшего профессионального образования направления подготовки 09.03.01 Информатика и вычислительная техника, разработанных в ГОУ ВПО «Донецкий национальный университет».

Разработчик:

Доцент кафедры компьютерных технологий

Шарий Т.В.

Программа учебной дисциплины утверждена на заседании кафедры компьютерных технологий

Протокол № 12 от «2» апреля 2020 г.

Зав. кафедрой компьютерных технологий

Ермоленко Т.В.

Программа учебной дисциплины одобрена учебно-методической комиссией физико-технического факультета

Протокол № 5 от «15» апреля 2020 г.

Председатель учебно-методической комиссии
 физико-технического факультета

Котенко В.Н.

1. ОБЛАСТЬ ПРИМЕНЕНИЯ И МЕСТО ДИСЦИПЛИНЫ В УЧЕБНОМ ПРОЦЕССЕ

Учебная дисциплина «Технологии разработки программного обеспечения» относится к вариативной части профессионального блока и состоит из двух содержательных модулей: модуль 1 – «Анализ требований и проектирование программного обеспечения», модуль 2 – «Инструментарий разработки и документирования программного обеспечения».

Основывается на базе дисциплин: «Основы программирования», «Информатика и информационно-коммуникационные технологии». Является основой для изучения следующих дисциплин: «Тестирование и внедрение программного обеспечения», «WEB-программирование», «Объектно-ориентированное программирование».

2. СТРУКТУРА ДИСЦИПЛИНЫ

<i>Характеристика учебной дисциплины</i>				
Направление подготовки	09.03.01 Информатика и вычислительная техника			
Профиль	Информатика и вычислительная техника			
Образовательная программа	Бакалавриат			
Квалификация	Академический бакалавр			
Количество содержательных модулей	2			
Дисциплина базовой / вариативной части образовательной программы	Профессиональный блок. Вариативная часть			
Формы контроля (МК, экзамен, зачет)	Модульный контроль, зачет			
Показатели	очная форма обучения		заочная форма обучения	
	нормат. срок	ускор. срок	нормат. срок	ускор. срок
Количество зачётных единиц (кредитов)	3	3	3	3
Год подготовки	2	1	2	1
Семестр	3	3	1	1
Количество часов	108	108	108	108
- лекционных	18	18	4	4
- практических, семинарских				
- лабораторных	36	36	8	8
- самостоятельной работы	54	54	96	96
в т. ч. индивидуальное задание				
Недельное количество часов, т. ч.	6	6	6	6
аудиторных	3	3	0,75	0,75

3. ОПИСАНИЕ ДИСЦИПЛИНЫ

Цель – формирование у студентов базовых знаний и навыков, связанных с современным процессом разработки программного обеспечения.

Задачи – изучение процессов и моделей жизненного цикла разработки программного обеспечения; изучение основ составления документации на всех этапах разработки; изучение и применение на практике инструментария: систем контроля версий, багтрекеров, документирования кода; изучение основ рефакторинга кода.

Требования к результатам освоения дисциплины:

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ГОС ВПО по данному направлению подготовки (профилю):

а) общекультурных (ОК):

способность работать в коллективе, толерантно воспринимать социальные, этнические, конфессиональные и культурные различия (ОК-6);

способность к самоорганизации и самообразованию (ОК-7).

б) общепрофессиональных (ОПК):

знание современных методов построения и анализа алгоритмов, основ численных методов и умение их использовать на практике (ОПК-4).

в) профессиональных (ПК):**проектно-конструкторская деятельность:**

пользоваться методиками использования программных средств для решения практических задач (ПК-2);

использовать и самостоятельно разрабатывать интерфейсы взаимодействия человека и ЭВМ (ПК-3);

знание принципов программирования, средств современных языков программирования, структур данных (ПК-5).

Проектно-технологическая деятельность:

знание современных технологий и инструментальных способов разработки сложных программных систем (инженерии программного обеспечения), умение их использовать на всех этапах жизненного цикла программ (ПК-14).

Научно-исследовательская деятельность:

базовые знания научно-методических основ и стандартов в области компьютерной инженерии, проводить эксперимент по проверке корректности решений, рассчитывать экономическую эффективность (ПК-15);

умение готовить и проводить доклады с использованием современных компьютерных средств, писать научно-технические отчеты, оформлять результаты исследований в виде статей (ПК-16).

Научно-педагогическая деятельность:

готовить конспекты лекций, проводить повышение квалификации сотрудников (ПК-17).

Сервисно-эксплуатационная деятельность:

инсталлировать, настраивать и сопровождать программное и аппаратное обеспечение информационных и автоматизированных систем (ПК-21).

В результате изучения учебной дисциплины студент должен**Знать:**

- этапы, процессы и модели жизненного цикла программного обеспечения;
- роли и функции разработчиков в современной IT-разработке;
- стандарты документирования программного обеспечения на этапах анализа требований, планирования, проектирования и кодирования программного продукта;
- принципы работы систем контроля версий и багтрекеров;
- основы рефакторинга кода.

Уметь:

- анализировать и составлять спецификации требований заказчика (техническое задание, SRS);
- готовить макет продукта; составлять техническую документацию (SDD);
- выявлять «запахи» кода и осуществлять рефакторинг кода;
- составлять справочную документацию продукта для конечных пользователей.

Владеть:

- навыками разработки программного обеспечения с помощью систем контроля версий, багтрекеров и возможностей инструментальных сред разработчика, касающихся рефакторинга кода.

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ И ФОРМЫ ОРГАНИЗАЦИИ УЧЕБНОГО ПРОЦЕССА

Порядковый номер и тема	Краткое содержание темы
	Содержательный модуль 1. Анализ требований и проектирование ПО
Тема 1. Этапы, процессы и модели жизненного цикла программного обеспечения.	Четыре «П» (The 4P's) разработки ПО (Процесс, Продукт, Персонал, Проект). Этапы разработки ПО. Классические стратегии разработки ПО: каскадная, инкрементная, эволюционная, V-образная, RAD.
Тема 2. Введение в инженерии требований.	Функциональные и нефункциональные, концептуальные и детальные требования к ПО. Методы и инструменты сбора и анализа требований. Составление спецификаций требований заказчика (SRS). Функциональные схемы IDEF, диаграммы вариантов использования UML.
Тема 3. Планирование разработки программного обеспечения	Диаграммы Ганта и инструментарий для работы с ними. Виды рисков, оценка рисков.
Тема 4. Макетирование программного продукта.	Понятие и виды макета продукта. Средства макетирования программного продукта.
Тема 5. Этап проектирования программного обеспечения	Основные этапы, принципы и паттерны проектирования ПО. Модульность ПО. Связность и сцепление модулей. Методологические антипаттерны в разработке ПО.
Тема 6. Техническая документация	Техническая документация ПО (SDD). Диаграммы UML классов, компонентов.
	Содержательный модуль 2. Инструментарий разработки и документирования ПО
Тема 7. Инструментарий поддержки разработки программного обеспечения	Централизованные системы контроля версий. Работа с системой Subversion и программой-клиентом TortoiseSVN. Организация репозитория. Основные команды и принципы работы с Subversion. Распределенные системы контроля версий. Программа Git. Организация хранения данных. Основные команды и принципы работы с Git. Удаленные репозитории. Ресурсы GitHub, BitBucket. Багтрекеры. Системы инспектирования кода.
Тема 8. Рефакторинг кода	«Запахи» кода. Приемы рефакторинга кода: реорганизация методов и данных; реорганизация условных выражений; обобщения.
Тема 9. Инструменты и принципы автодокументирования кода	Автодокументируемый код. Инструменты автодокументирования кода (doxygen, SandCastle).
Тема 10. Пользовательская документация ПО	Создание справочной документации. Файлы README, F.A.Q.

Курс дисциплины «Технологии разработки программного обеспечения» предусматривает следующие **формы организации учебного процесса**:

1. лекции;
2. лабораторные занятия;
3. самостоятельная работа студента.

По источнику передачи и восприятия учебной информации используются словесные (лекция, беседа), наглядные (слайды, иллюстрации, коды программ), практические (исследования, упражнения, лабораторные работы) методы.

По характеру познавательной деятельности студентов используются объяснительно-иллюстративные и репродуктивные методы, проблемное преподавание, частично-поисковый и исследовательский методы.

В зависимости от основной дидактической цели и задач используются методы устного изложения знаний, закрепление учебного материала, самостоятельной работы студентов по осмыслению и усвоению нового материала, работы по применению знаний на практике и выработке умений и навыков, проверки и оценки знаний, умений и навыков.

Используются следующие методы контроля:

1. устный контроль (экспресс-опрос на лекциях);
2. проверка конспектов;
3. защита лабораторных работ;
4. проверка самостоятельных работ;
5. модульная контрольная работа.

Тематический план

	Содержательный модуль 1																						
Названия содержательных модулей и тем	Количество часов																						
	Очная форма обучения											Заочная форма обучения											
	Нормативный срок обучения						Ускоренный срок обучения					Нормативный срок обучения					Ускоренный срок обучения						
	всего	в т. ч.					всего	в т. ч.				всего	в т. ч.				всего	в т. ч.					
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа		индивидуальная работа	лекции	практические	лабораторные		самостоятельная работа	индивидуальная работа	лекции	практические	лабораторные работы	самостоятельная работа
Тема 1. Этапы, процессы и модели жизненного цикла программного обеспечения.	12	2		2	8		12	2		2	8		12				12				12		
Тема 2. Введение в инженерии требований.	16	2		6	8		16	2		6	8		13	1		2	10		13	1		2	10
Тема 3. Планирование разработки программного обеспечения	7	1		2	4		7	1		2	4		6				6		6				6
Тема 4. Макетирование программного продукта.	7	1		2	4		7	1		2	4		6				6		6				6
Тема 5. Этап проектирования программного обеспечения	14	2		4	8		14	2		4	8		13	1		2	10		13	1		2	10
Тема 6. Техническая документация	10	2		4	4		10	2		4	4		8				8		8				8
Итого по 1-му содержательному модулю	66	10		20	36		66	10		20	36		58	2		4	52		58	2		4	52

	Содержательный модуль 2																							
Названия содержательных модулей и тем	Количество часов																							
	Очная форма обучения										Заочная форма обучения													
	Нормативный срок обучения					Ускоренный срок обучения					Нормативный срок обучения					Ускоренный срок обучения								
	всего	В Т. Ч.					всего	В Т. Ч.					всего	В Т. Ч.					всего	В Т. Ч.				
		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные	самостоятельная работа	индивидуальная работа		лекции	практические	лабораторные работы	самостоятельная работа	
Тема 7. Инструментарий поддержки разработки программного обеспечения	16	4		6	6		16	4		6	6		19	1		2	16		19	1		2	16	
Тема 8. Рефакторинг кода	12	2		6	4		12	2		6	4		19	1		2	16		19	1		2	16	
Тема 9. Инструменты и принципы автодокументирования кода	7	1		2	4		7	1		2	4		8				8		8				8	
Тема 10. Пользовательская документация ПО	7	1		2	4		7	1		2	4		4				4		4				4	
Итого по 2-му содержательному модулю	42	8		16	18		42	8		16	18		50	2		4	44		50	2		4	44	
Всего часов	108	18		36	54		108	18		36	54		108	4		8	96		108	4		8	96	

5. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРОВЕДЕНИЯ ЛЕКЦИОННЫХ, ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ ЗАНЯТИЙ

Темы лекционных занятий

№ п/п	Название темы	Количество часов
1.	Этапы, процессы и модели жизненного цикла программного обеспечения	2
2.	Введение в инженерии требований	2
3.	Планирование разработки программного обеспечения. Макетирование программного продукта	2
4.	Этап проектирования программного обеспечения	2
5.	Техническая документация	2
6.	Инструментарий поддержки разработки программного обеспечения	4
7.	Рефакторинг кода	2
8.	Инструменты и принципы автодокументирования кода. Пользовательская документация программного обеспечения	2
	ВСЕГО	18

Темы лабораторных занятий

№ п/п	Название темы	Количество часов
1.	Составление спецификаций требований заказчика программного продукта.	10
2.	Проектирование и макетирование программного продукта.	10
3.	Инструментарий разработки программного продукта.	6
4.	Рефакторинг.	4
5.	Подготовка программного продукта к эксплуатации	8
	ВСЕГО	36

6. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Организация самостоятельной работы студентов.

Самостоятельная работа студентов по курсу «Технологии разработки программного обеспечения» предусматривает изучение дополнительной технической литературы и интернет-источников, рекомендуемые этой программой; самостоятельную разработку алгоритмов и текстов программ лабораторных работ, изучение дополнительного инструментария. При желании студент может подготовить реферат или доклад по одной из приведенных ниже тем:

1. Стратегии разработки программного обеспечения.
2. Инженерия требований. Анализ инструментов на рынке.
3. Техническая документация. Особенности профессии «Технический писатель».
4. Инструментарий макетирования программного обеспечения.
5. Объектно-ориентированные паттерны и анти-паттерны проектирования.

6. Современные системы контроля версий.
7. Современные багтрекеры.
8. Системы инспектирования кода.
9. Рефакторинг кода. Инструмент ReSharper.
10. Автодокументирование кода.

№ n/n	Название темы	Количество часов
1.	Этапы, процессы и модели жизненного цикла программного обеспечения.	8
2.	Введение в инженерии требований.	8
3.	Планирование разработки программного обеспечения	4
4.	Макетирование программного продукта.	4
5.	Этап проектирования программного обеспечения	8
6.	Техническая документация	4
7.	Инструментарий поддержки разработки программного обеспечения	6
8.	Рефакторинг кода	4
9.	Инструменты и принципы автодокументирования кода	4
10.	Пользовательская документация ПО	4
	ВСЕГО	54

7. ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ

Индивидуальные задания предусмотрены к каждой лабораторной работе.
Полный перечень всех заданий приведен в УМКД дисциплины.

Образец индивидуального задания

(Из лабораторной работы №1). Добавьте к разрабатываемой информационной системе дополнительный функционал в соответствии с вариантом:

Вариант 1. Генерация и сканирование штрих-кодов пропусков.

Вариант 2. Возможность менять в окне опций настройки графического интерфейса.

Вариант 3. Возможность менять в окне опций настройки адрес служебного имейла.

При окончании любого товара высылать на служебный имейл сообщение.

Вариант 4. Возможность переключаться между языками (русский, украинский и английский).

Вариант 5. Возможность конвертирования файлов с данными в формат Excel и XML.

Вариант 6. Автоматическое создание бэкапов информации с определенной периодичностью. Периодичность задается в окне настроек.

Вариант 7. При трех подряд неудачных попытках авторизации закрывать программу.

Вариант 8. Следить, чтобы не было запущено одновременно более одного экземпляра программы.

Вариант 9. Автоматический ресайз фотографий сотрудников при загрузке до размера 200x300.

Вариант 10. Автоматический запуск браузера и переход на специальный сайт по нажатию кнопки.

8. КОНТРОЛЬНЫЕ ВОПРОСЫ К ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

1. Этапы жизненного цикла программного обеспечения.
2. Процессы и модели жизненного цикла программного обеспечения
3. Функциональные и нефункциональные, концептуальные и детальные требования к программному продукту.
4. Составление спецификаций требований (SRS).
5. Планирование разработки программного обеспечения.
6. Модульность. Связность и сцепление модулей.
7. Техническая документация ПО (SDD).
8. Архитектурные паттерны и паттерны управления.

9. ОБРАЗЕЦ МОДУЛЬНОГО КОНТРОЛЯ

ГОУ ВПО «Донецкий национальный университет»

Физико-технический факультет

Направление подготовки	09.03.01 - Информатика и вычислительная техника
Профиль	Информатика и вычислительная техника
Программа подготовки	бакалавриат
Семестр	3
Учебная дисциплина	Технологии разработки программного обеспечения

МОДУЛЬНАЯ КОНТРОЛЬНАЯ РАБОТА №1 ВАРИАНТ №1

1. Созданные в процессе разработки, наряду с исполняемыми файлами программы, составляющие ПО, называются:

- а) варианты использования б) паттерны в) модули г) артефакты д) компоненты

2. Какая связность модуля из перечисленных наихудшая?

- а) логическая б) временная в) коммуникативная г) процедурная д) функциональная

3. Какое сцепление модулей из перечисленных наихудшее?

- а) по образцу б) по внешним ссылкам в) по данным г) по содержанию д) по управлению

4. Выберите верное утверждение о характеристиках модуля:

- а) связность необходимо уменьшать, сцепление – увеличивать
 б) сцепление необходимо уменьшать, связность – увеличивать
 в) и связность, и сцепление необходимо уменьшать
 г) и связность, и сцепление необходимо увеличивать

5. Межклассовое отношение типа "часть/целое" ("has-a"), в котором часть может существовать отдельно от целого, называется:

- а) наследование б) композиция в) агрегирование г) делегирование д) полиморфизм

6. Что такое PSP?

- а) персональный процесс разработки б) персональный продукт
 в) персональное управление проектом г) командный персонал

7. Организацией и контролем деятельности команды разработчиков занимается:

- а) бизнес-аналитик б) менеджер проекта в) QA г) архитектор д) тестировщик

8. Какая из следующих моделей разработки ПО используется для коротких проектов (2-3 месяца) с небольшими командами разработчиков?

- а) V-образная б) каскадная в) RAD г) инкрементальная д) спиральная

9. Выберите верное утверждение:

- а) требования к пользовательскому интерфейсу не могут иметь ни тип С, ни тип D
 б) требования к пользовательскому интерфейсу могут быть только типа С
 в) требования к пользовательскому интерфейсу могут быть только типа D
 г) требования к пользовательскому интерфейсу могут быть и типа С, и типа D

10. В каком документе оформляются результаты планирования проекта ПО?

- а) SPMP б) SRS в) SDD г) STD д) TZ

11. Для предварительной оценки объема продукта используют параметр (метрику):

- а) SRS б) SPMP в) CMM г) PSize д) LOC

12. «4П» разработки программных продуктов включает:

- а) прогресс, программа, персонал, процесс
 б) прогресс, персонал, продукт, проект
 в) проект, продукт, персонал, процесс
 г) программа, проект, персонал, процесс

Утверждено на заседании кафедры компьютерных технологий,
 протокол № 12 от «2» апреля 2020 г.

Заведующий кафедрой

Преподаватель

Критерии оценивания модульного контроля

<i>Номер задания</i>	<i>Количество баллов</i>
1-12	24 (1 вопрос = 2 балла)
Всего	24

10. ОБРАЗЕЦ ЭКЗАМЕНАЦИОННОГО БИЛЕТА

Экзамен не предусмотрен.

11. ОБРАЗЕЦ ТЕСТОВОГО ЗАДАНИЯ

1. Что такое СММ?

- а) модель управления клиентами
- б) модель требований клиента
- в) модель возможностей клиента
- г) модель зрелости возможностей

2. В какой из следующих моделей разработки ПО выделен отдельный этап анализа рисков?

- а) V-образная б) каскадная в) RAD г) инкрементальная д) спиральная

3. Составлением спецификаций требований заказчика занимается:

- а) бизнес-аналитик
- б) менеджер проекта
- в) тимлид
- г) архитектор
- д) тестировщик

12. КРИТЕРИИ ОЦЕНИВАНИЯ

По курсу предполагается проведение промежуточной аттестации в виде модульного контроля, выполнение лабораторных работ.

*Распределение баллов, которые могут получить студенты
в процессе изучения дисциплины*

	Содержательный модуль №1				Содержательный модуль №2				Всего
	Лабораторные работы		Мод. контр. работа	Всего С.М. №1	Лабораторные работы			Всего С.М. №2	
	№1	№2			№3	№4	№5		
Макс. балл	20	11	24	55	15	15	15	45	100

Шкала соответствия баллов национальной шкале

Оценка по шкале ECTS	Оценка по 100-балльной шкале	Оценка по государственной шкале (экзамен, дифференцированный зачет)	Оценка по государственной шкале (зачет)
A	90-100	5 (отлично)	зачтено
B	80-89	4 (хорошо)	зачтено
C	75-79	4 (хорошо)	зачтено
D	70-74	3 (удовлетворительно)	зачтено
E	60-69	3 (удовлетворительно)	зачтено
FX	35-59	2 (неудовлетворительно) с возможностью повторной сдачи	не зачтено
F	0-34	2 (неудовлетворительно) с возможностью повторной сдачи при условии обязательного набора дополнительных баллов	не зачтено

Оценка за овладение курса выставляется по следующим критериям:

– Оценку «отлично» заслуживает студент, который обнаружил глубокие знания при ответах на теоретические вопросы по темам курса, а также выполнил практические задания в полном объёме и набрал более 90 баллов.

– Оценку «хорошо» заслуживает студент, сделавший ошибки в теоретических или практических ответах, которые могут быть интерпретированы как малосущественные для вопросов, которые рассматривались. Студент должен набрать более 75 баллов.

– Оценку «удовлетворительно» заслуживает студент, который выполнил задания неполно и с ошибками, но при этом набрал более 60 баллов.

– Оценку «неудовлетворительно» заслуживает студент, который не выполнил большинства теоретических и практических задач и набрал менее 60 баллов.

13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

Для проведения лекционных занятий требуется аудитория на поток, оборудованная мультимедийным проектором и экраном, или интерактивной доской. Для проведения лабораторных занятий по дисциплине необходим оборудованный ПЭВМ или ноутбуками компьютерный класс с возможностью выхода в Интернет.

14. РЕКОМЕНДОВАННАЯ ЛИТЕРАТУРА

№ п/п	Наименование	Кол-во экземпляров в библиотеке ДонНУ	Наличие электронной версии в ЭБС
<i>Основная литература</i>			
1.	Чейкон С. Pro Git. Современная разработка приложений [Текст] / С. Чейкон, Б. Страуб. – Apress, 2014. – 564 с.	-	Да
<i>Дополнительная литература</i>			
2.	Брауде Э. Технология разработки программного обеспечения / Э. Брауде. – СПб.: Питер, 2004. – 655с.	1	Нет
3.	Макконелл С. Совершенный код. Мастер-класс / С. Макконелл. – М.: Русская редакция, 2014. – 896 с.	1	Нет
4.	Константайн Л. Разработка программного обеспечения / Л. Константайн, Л. Локвуд. – СПб.: Питер, 2004. – 592 с.	1	Нет
5.	Мартин Р. Чистый код: Создание, анализ и рефакторинг. Библиотека программиста/ Р. Мартин. – СПб.: Питер, 2010. – 464 с.	-	Нет

15. ИНФОРМАЦИОННЫЕ РЕСУРСЫ

1. Рефакторинг и паттерны проектирования. URL: <https://refactoring.guru/ru/> (дата обращения 18.03.2020 г.).
2. Материалы курса «Технологии разработки программного обеспечения» (автор – Шарий Т.В.). URL: <https://github.com/ar1st0crat/SoftDevCourse/> (дата обращения 18.03.2020 г.).

16. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

1. IDE Visual Studio Community версии 2017 или 2019.
2. Клиент TortoiseSVN системы контроля версий Subversion.
3. Система контроля версий Git.
4. Утилита для генерации документации Doxygen.

Рабочая программа рассмотрена и переутверждена на заседании кафедры компьютерных технологий с изменениями (без изменений) на 2020 год.

Протокол № 12 от «2» апреля 2020 г.

Заведующий кафедрой

Ермоленко Т.В.

Рабочая программа рассмотрена и переутверждена на заседании кафедры компьютерных технологий с изменениями (без изменений) на 2021 год.

Протокол № ____ от «____» _____ 2021 г.

Заведующий кафедрой

Рабочая программа рассмотрена и переутверждена на заседании кафедры компьютерных технологий с изменениями (без изменений) на 2022 год.

Протокол № ____ от «____» _____ 2022 г.

Заведующий кафедрой

Рабочая программа рассмотрена и переутверждена на заседании кафедры компьютерных технологий с изменениями (без изменений) на 2023 год.

Протокол № ____ от «____» _____ 2023 г.

Заведующий кафедрой