

Министерство науки и высшего образования  
Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Донецкий государственный университет»

Физико-технический факультет  
Кафедра компьютерных технологий



УТВЕРЖДАЮ  
проректор

«29» марта 2024 г.

П.А. Машаров

МП

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

### ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Укрупненная группа направлений  
подготовки  
Программа высшего образования  
Направление подготовки

Профиль подготовки  
Квалификация  
Форма обучения

09.00.00 Информатика и вычислительная  
техника  
Программа бакалавриата  
09.03.01 Информатика и вычислительная  
техника  
Информатика и вычислительная техника  
Бакалавр  
Очная, заочная

Рабочая программа адаптирована для лиц  
с ограниченными возможностями здоровья и инвалидов

Донецк 2024

Рабочая программа дисциплины «Технологии разработки программного обеспечения» для обучающихся по направлению подготовки 09.03.01 Информатика и вычислительная техника (Профиль подготовки: Информатика и вычислительная техника), составлена на основании Федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденного приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 929 (с изм. и доп.), Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры, утвержденного приказом Министерства науки и высшего образования Российской Федерации от 06 апреля 2021 г. № 245 (с изм. и доп.), в соответствии с учебным планом, утвержденным Ученым советом ФГБОУ ВО «ДонГУ» для набора 2024 года.

Разработчики:

Старший преподаватель кафедры  
компьютерных технологий



И.И. Максименко

Рабочая программа одобрена на заседании кафедры компьютерных технологий.  
Протокол от 26.03.2024 г. № 12

Заведующий кафедрой



Г.В. Аверин

СОГЛАСОВАНО:

Декан физико-технического факультета  
28.03.2024 г.



С.А. Фоменко

Учебно-методическая комиссия физико-технического факультета.  
Протокол от 27.03.2024 г. № 2

Председатель



В. Н. Котенко

Руководитель основной профессиональной  
образовательной программы,  
д-р технических наук, проф.  
26.03.2024 г.



Г.В. Аверин

# 1. МЕСТО ДИСЦИПЛИНЫ / ПРАКТИКИ / КУРСОВОЙ РАБОТЫ / ГОСУДАРСТВЕННОЙ ИТОГОВОЙ АТТЕСТАЦИИ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

- 1.1. Требования к предварительной подготовке обучающихся, предшествующие и сопутствующие дисциплины, на которых основывается изучение данной: «Основы программирования», «Информатика и информационно-коммуникационные технологии».
- 1.2. Дисциплины, курсовые работы и практики, для которых освоение данной дисциплины необходимо как предшествующее: «Тестирование и внедрение программного обеспечения», «WEB-программирование», «Объектно-ориентированное программирование».

## 2. ОПИСАНИЕ ДИСЦИПЛИНЫ / ПРАКТИКИ / КУРСОВОЙ РАБОТЫ / ГОСУДАРСТВЕННОЙ ИТОГОВОЙ АТТЕСТАЦИИ

### 2.1. Общая характеристика

Наименование показателя	Значение показателя
Название образовательной программы	09.03.01 Информатика и вычислительная техника
Шифр и название в соответствии с учебным планом	Б1.Б.М4 Технологии разработки программного обеспечения
Часть образовательной программы	Базовая часть Вариативная часть: выбор вуза Вариативная часть: выбор обучающегося
Количество зачетных единиц / всего часов	4 / 128

### 2.2. Распределение часов по периодам обучения

Форма обучения	курс	семестр	Общее количество часов					Форма контроля
			лекционных	лабораторных	практических	самостоятельной работы	всего	
Очная	1	1	18	108	-	-	126	зачет
Очная, всего								
Заочная	1	1	0,8	1,2	-	-	2	зачет

## 3. ЦЕЛИ ДИСЦИПЛИНЫ / ПРАКТИКИ / КУРСОВОЙ РАБОТЫ / ГОСУДАРСТВЕННОЙ ИТОГОВОЙ АТТЕСТАЦИИ

Изучение процессов и моделей жизненного цикла разработки программного обеспечения, основ составления документации на всех этапах разработки, изучение и применение на практике инструментария, систем контроля версий, багтрекеров, документирования кода, изучение основ рефакторинга кода для формирования у студентов базовых знаний и навыков современных процессов разработки программного обеспечения.

#### 4. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ КОМПОНЕНТА ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ, ИХ ИНДИКАТОРЫ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

##### 4.1. Компетенции

ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью.

ОПК-6. Способен разрабатывать бизнес- планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.

##### 4.2. Индикаторы компетенций

ОПК-4.3. Применяет навыки составления технической документации на различных этапах жизненного цикла информационной системы.

ОПК-6.3. Обладает навыками составления технической документации на различных этапах жизненного цикла информационной системы.

##### 4.3. Результаты обучения

ОПК-4.3.1. Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.

ОПК-4.3.2. Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.

ОПК-4.3.3. Аргументированно выбирает метод составления технической документации на различных этапах жизненного цикла информационной системы.

ОПК-6.3.1. Знает основные принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.

ОПК-6.3.2. Умеет анализировать цели и ресурсы организации, разрабатывать бизнес-планы развития ИТ, составлять технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием.

ОПК-6.3.3. Аргументированно выбирает методы составления технических заданий.

Компетенции	Индикаторы	Результаты обучения
ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью.	ОПК-4.3. Применяет навыки составления технической документации на различных этапах жизненного цикла информационной системы.	ОПК-4.3.1. Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы. ОПК-4.3.2. Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы. ОПК-4.3.3. Аргументированно выбирает метод составления технической документации на различных этапах жизненного цикла информационной системы.
ОПК-6. Способен разрабатывать бизнес- планы и технические задания на оснащение отделов, лабораторий,	ОПК-6.3. Обладает навыками составления технической документации на различных этапах жизненного цикла	ОПК-6.3.1. Знает основные принципы формирования и структуру бизнес-планов и технических заданий на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3.2. Умеет анализировать цели и ресурсы организации, разрабатывать бизнес- планы развития ИТ, составлять технические задания на

офисов компьютерным и сетевым оборудованием.	информационной системы.	оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием. ОПК-6.3.3. Аргументированно выбирает методы составления технических заданий.
--	-------------------------	---

## 5. ПРОГРАММА ДИСЦИПЛИНЫ

Порядковый номер и тема	Краткое содержание темы
	<b><i>Содержательный модуль 1.</i></b> <b>Анализ требований и проектирование ПО</b>
<b>Тема 1.</b> 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.
<b>Тема 2.</b> 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.
<b>Тема 3.</b> Планирование разработки программного обеспечения	Диаграммы Ганта и инструментарий для работы с ними. Виды рисков, оценка рисков.
<b>Тема 4.</b> Макетирование программного продукта.	Понятие и виды макета продукта. Средства макетирования программного продукта.
<b>Тема 5.</b> Этап проектирования программного обеспечения	Основные этапы, принципы и паттерны проектирования ПО. Модульность ПО. Связность и сцепление модулей. Методологические антипаттерны в разработке ПО.
<b>Тема 6.</b> Техническая документация	Техническая документация ПО (SDD). Диаграммы UML классов, компонентов.
	<b><i>Содержательный модуль 2.</i></b> <b>Инструментарий разработки и документирования ПО</b>
<b>Тема 7.</b> 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.
<b>Тема 8.</b> Рефакторинг кода	«Запахи» кода. Приемы рефакторинга кода: реорганизация методов и данных; реорганизация условных выражений; обобщения.

<b>Тема 9.</b> Инструменты и принципы автодокументирования кода	Автодокументируемый код. Инструменты автодокументирования кода (doxygen, SandCastle).
<b>Тема 10.</b> 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.

## 6. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРОВЕДЕНИЯ ЛЕКЦИОННЫХ, ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ ЗАНЯТИЙ

### Темы лекционных занятий

<b>№ п/п</b>	<b>Название темы</b>	<b>Количество часов</b>
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
	<b>ВСЕГО</b>	<b>18</b>

### Темы лабораторных занятий

<i>№ п/п</i>	<i>Название темы</i>	<i>Количество часов</i>
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
	<b>ВСЕГО</b>	<b>36</b>

## 7. 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.

<b>№ п/п</b>	<b>Название темы</b>	<b>Количество часов</b>
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
	<b>ВСЕГО</b>	<b>54</b>

## 8. 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 бекапов 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.

## 9. 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.

## 10. ОБРАЗЕЦ МОДУЛЬНОГО КОНТРОЛЯ

ФГБОУ ВПО «Донецкий государственный 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)
<i>Всего</i>	<b>24</b>

## 11. ОБРАЗЕЦ ЭКЗАМЕНАЦИОННОГО БИЛЕТА

Экзамен не предусмотрен.

## 12. 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

## 13. 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)
<b>A</b>	90-100	5 (отлично)	зачтено
<b>B</b>	80-89	4 (хорошо)	зачтено
<b>C</b>	75-79	4 (хорошо)	зачтено

<b>D</b>	70-74	3 (удовлетворительно)	зачтено
<b>E</b>	60-69	3 (удовлетворительно)	зачтено
<b>FX</b>	35-59	2 (неудовлетворительно) с возможностью повторной сдачи	не зачтено
<b>F</b>	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 объеме 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.

#### 14. 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.

#### 15. РЕКОМЕНДОВАННАЯ ЛИТЕРАТУРА

№ п/п	Наименование	Кол-во экземпляров в библиотеке ДонНУ	Наличие электронной версии в ЭБС
<b><i>Основная литература</i></b>			
1.	Чейкон С. Pro Git. Современная разработка приложений [Текст] / С. Чейкон, Б. Страуб. – Apress, 2014. – 564 с.	-	Да
<b><i>Дополнительная литература</i></b>			
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

## 16. 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.).

## 17. 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.