

Учреждение образования  
«Полоцкий государственный университет имени Евфросинии Полоцкой»

**УТВЕРЖДАЮ**

Ректор учреждения образования  
«Полоцкий государственный  
университет имени  
Евфросинии Полоцкой»

 Ю.Я. Романовский

« 23 » 2025 г.

Регистрационный №УД-469/25/уч

## **ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Учебная программа учреждения образования  
по учебной дисциплине для специальности  
**6-05-0533-12 «Кибербезопасность»**

2025 г.

Учебная программа составлена на основе образовательного стандарта по специальности высшего образования ОСВО 6-05-0533-12-2023 и учебного плана специальности 6-05-0533-12 «Кибербезопасность». Регистрационный 14-23/уч. ФКНЭ от 04.04.2023 для дневной формы получения высшего образования

**СОСТАВИТЕЛЬ:**

Сергей Васильевич Кухта, старший преподаватель кафедры математики и компьютерной безопасности учреждения образования «Полоцкий государственный университет имени Евфросинии Полоцкой»

**РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:**

Кафедрой математики и компьютерной безопасности учреждения образования «Полоцкий государственный университет имени Евфросинии Полоцкой»  
(протокол № 11 от «21» 11 2025 г.)

Научно-методическим советом учреждения образования «Полоцкий государственный университет имени Евфросинии Полоцкой»  
(протокол № 3 от «15» 12 2025 г.)

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная дисциплина «Технология разработки программного обеспечения» знакомит студентов с основами программной инженерии, архитектурой и проектированием программных систем, управлением программными проектами и обеспечением качества, методологиями проектирования программных систем.

**Цели** изучения учебной дисциплины «Технология разработки программного обеспечения»:

- образовательная: дать систему знаний по теоретическим и прикладным основам проектирования программных систем;

- развивающая: развитие способности к совершенствованию своей деятельности, повышению своего профессионального уровня; развитие интереса обучающихся к проектированию программных средств для решения производственно-технологических задач;

- воспитательная: воспитание способности к социальному взаимодействию, межличностным коммуникациям; умение работать в коллективе, решать проблемные вопросы, принимать самостоятельные решения.

**Задачи**, решаемые при изучении учебной дисциплины «Технология разработки программного обеспечения»:

- обучение основам моделирования, проектирования и разработки программных систем;

- изучение основ управления программными проектами;

- освоение работы с различными методологиями, технологиями и платформами разработки программного обеспечения (ПО).

При изучении дисциплины «Технология разработки программного обеспечения» у студентов специальности 6-05-0533-12 «Кибербезопасность» должен сформироваться набор компетенций, соответствующих присваиваемой по завершению высшего образования квалификации «Специалист по кибербезопасности», обеспечивающих выпускникам успешность применения полученных знаний и умений в дальнейшей профессиональной деятельности:

***специализированные компетенции:***

- Владеть классическими и современными методами численного решения оптимизационных задач, навыками их практической реализации, определять возможности применения изученных методов к задачам, возникающим в машинном обучении.

- Разрабатывать программное обеспечение в интегрированных средах разработки. Использовать автоматизированные средства разработки программных средств. Разрабатывать необходимую для этого документацию.

***универсальные компетенции:***

- Владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации.

- Решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий.

- Быть способным к саморазвитию и совершенствованию в профессиональной деятельности.

- Проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности.

В результате освоения учебной дисциплины студент должен:

***знать:***

- современные технологии разработки ПО и перспективы их развития;

- архитектурное моделирование;

- возможности и основные принципы разработки программного обеспечения;

- использование шаблонов проектирования;

**уметь:**

- проводить исследование предметной области решаемой задачи и анализировать полученный результат;
- разрабатывать техническое задание на проектируемую программную среду;
- строить модели будущей системы на фазе бизнес-моделирования и проектирования;
- моделировать предметную область ПО с использованием современных средств проектирования;
- разрабатывать интерфейс пользователя;
- использовать шаблоны проектирования для создания ПО;

**владеть:**

- технологиями моделирования программных систем;
- технологиями проектирования и разработки программных систем;
- CASE-технологиями проектирования автоматизированных систем.

**Связи с другими учебными дисциплинами.**

Базой для изучения учебной дисциплины «Технология разработки программного обеспечения» являются учебные дисциплины «Программирование на C++», «Программирование на Python», «Разработка кросс-платформенных приложений», «Технологии программирования», «Базы данных», «Надежность программного обеспечения», «Стандартизация программного обеспечения».

Знания, полученные при изучении учебной дисциплины «Технология разработки программного обеспечения», являются основой для дипломного проектирования и учебной дисциплины «Системы управления базами данных».

**Форма получения образования – дневная.**

В соответствии с учебным планом специальности 6-05-0533-12 «Кибербезопасность» на изучение учебной дисциплины «Технология разработки программного обеспечения» отводится:

Курс	3
Семестр	6
Всего часов по дисциплине	108
Всего аудиторных часов по дисциплине	68
В том числе:	
Лекций, часов	34
Практические занятия, часов	34
Самостоятельная работа, часов	40
Форма промежуточной аттестации	экзамен
Трудоемкость дисциплины, зачетные единицы	3
Курсовой проект, часов	40
Трудоемкость курсового проекта, зачетные единицы	1

## СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

### Раздел 1. АРХИТЕКТУРА ПРОГРАММНЫХ СИСТЕМ

#### *Тема 1.1. Основы программной инженерии.*

Особенности развития компьютерной индустрии. Классификации программного обеспечения и программных проектов. Особенности современных программных проектов. Системный подход как основа проектирования программного обеспечения. Понятие методологии разработки ПО.

#### *Тема 1.2. Архитектура программных систем и моделирование.*

Понятие архитектуры ПО. Критерии хорошей архитектуры. Модульная архитектура. Декомпозиция. Архитектурные представления. Сервисно-ориентированная архитектура. Моделирование в разработке ПО. Системное моделирование. Цели моделирования. Объектно-ориентированное моделирование.

### Раздел 2. ПРОЕКТИРОВАНИЕ ПРОГРАММНЫХ СИСТЕМ

#### *Тема 2.1. Унифицированный язык моделирования UML.*

Унифицированный язык моделирования UML, спецификации, история развития. Классификация диаграмм (структурные и поведенческие). SysML как расширение языка UML. Общая иерархия моделей. Назначение метамодели UML ее структура. Механизмы расширения UML (стереотипы, метасвойства, ограничения, комментарии).

#### *Тема 2.2. Диаграмма прецедентов.*

Диаграмма прецедентов, ее элементы и связи между элементами. Связи ассоциации, включения, расширения, обобщения. Правила составления спецификаций и сценариев. Дополнительные обозначения языка UML для бизнес-моделирования. Область применения диаграммы прецедентов.

#### *Тема 2.3. Диаграмма классов.*

Диаграмма классов, ее элементы и связи между элементами. Статические операции и атрибуты. Интерфейсы и абстрактные классы. Шаблон класса (параметризованный класс). Область применения диаграммы классов.

#### *Тема 2.4. Диаграммы взаимодействия.*

Виды диаграмм взаимодействия и область их применения. Диаграммы взаимодействия, их элементы и связи между элементами. Диаграммы последовательности. Примеры диаграмм последовательности. Диаграммы кооперации. Примеры диаграмм кооперации Сравнение диаграмм последовательности и кооперативных диаграмм.

#### *Тема 2.5. Диаграммы деятельности и состояний.*

Диаграмма деятельности, ее элементы и связи. Дорожки. Область применения диаграмм деятельности. Примеры диаграмм деятельности. Диаграмма состояний, ее элементы и связи. Примеры диаграмм состояний.

#### *Тема 2.6. Шаблоны (паттерны) проектирования.*

Концепция шаблонов проектирования. Типы шаблонов проектирования: делегирования, функционального дизайна, интерфейса, адаптер. Шаблоны: абстрактная фабрика, строитель, фабричный метод, фасад, прототип, наблюдатель.

### Раздел 3. РЕАЛИЗАЦИЯ И СОПРОВОЖДЕНИЕ ПО

#### *Тема 3.1. Принципы проектирования пользовательского интерфейса.*

Виды интерфейсов ПО. Классификация интерфейсов пользователя. Психологическая основа пользовательского интерфейса. Этапы проектирования графического пользовательского интерфейса. Основные компоненты графического пользовательского интерфейса и рекомендации по их применению. Свойства качественного программного интерфейса.

*Тема 3.2. Диаграммы реализации.*

Виды диаграмм реализации и область их применения. диаграмм реализации, Элементы диаграмм и связи между элементами. Примеры диаграмм компонентов. Примеры диаграмм развертывания.

*Тема 3.3. Сопровождение ПО. Управление конфигурацией ПО.*

Процесс сопровождения ПО. Элементы процесса сопровождения ПО. Элементы конфигурации программного средства. Управление конфигурацией и версиями. Документирование в процессах жизненного цикла ПО.

## ТРЕБОВАНИЯ К КУРСОВОМУ ПРОЕКТУ

Курсовой проект по дисциплине «Технология разработки программного обеспечения» - работа студента, предназначенная для объективного контроля: степени освоения знаний, полученных студентами при изучении теоретического материала дисциплины; умений и навыков, полученных при выполнении практических работ; применения знаний, умений и навыков проектирования программного обеспечения и разработки прототипа программного средства для заданной предметной области; развитие навыков инженерного проектирования и навыков работы со специальной литературой согласно требованиям задания на курсовое проектирование.

Целью курсового проекта является:

- систематизация знаний и накопление практического опыта в области проектирования и реализации программного средства;
- формирование навыков самостоятельной работы по разработке программного обеспечения.

В состав курсового проекта входят:

- пояснительная записка;
- графическая часть;
- работающий прототип программного средства.

Пояснительная записка должна отражать основные этапы разработки курсового проекта. Графическая часть представляет собой диаграммы в нотации UML, отражающие результаты объектно-ориентированного анализа, проектирования и реализации программного средства. Прототип программного средства может разрабатываться на любом языке программирования.

На выполнение курсового проекта отводится 40 часов.

### Перечень тем курсовых проектов

1. Информационная система «Детский дом»
2. Информационная система театра
3. Информационная система деканата факультета
4. Информационная система «Распределение выпускников»
5. Информационная система водоканала
6. Информационная система «Расчет квартплаты»
7. Информационная система «Адвокатская контора»
8. Информационная система «Комбинат ЖБИ»
9. Информационная система приемной комиссии университета
10. Информационная система «Управление переработкой отходов и вторсырья»
11. Информационная система «Энергосбыт»
12. Информационная система страховой компании
13. Информационная система киностудии
14. Информационная система отдела кадров завода
15. Информационная система магазина одежды
16. Информационная система сети автозаправочных станций
17. Информационная система «Продажа стройматериалов»
18. Информационная система «Производство потолков»
19. Информационная система «Производство мебели по индивидуальным заказам»
20. Информационная система пилорамы
21. Информационная система «Лизинг автомобилей»
22. Информационная система фирмы по ремонту квартир
23. Информационная система предприятия по ремонту автомобильных дорог
24. Информационная система учета заявок населения в службе 115
25. Информационная система «Многоуровневый паркинг»
26. Информационная система «Учет оборудования в университете»

27. Информационная система «Агентство недвижимости»
28. Информационная система типографии
29. Информационная система «Ателье проката»
30. Информационная система «Курьерская служба»
31. Информационная система охранного предприятия
32. Информационная система «Клининговая компания»
33. Информационная система Event-агентства
34. Информационная система спортивной школы
35. Информационная система клиники пластической хирургии
36. Информационная система ветеринарной клиники
37. Информационная система фармацевтического предприятия
38. Информационная система «Репетитор»
39. Информационная система «База охоты и рыбалки»
40. Информационная система «Семья и дети»
41. Информационная система «Железнодорожные грузоперевозки»
42. Информационная система предприятия по установке и обслуживанию лифтов
43. Информационная система «Тепличное хозяйство»
44. Информационная система учета техосмотра транспортных средств
45. Информационная система транспортной инспекции
46. Информационная система мониторинга загрязнений окружающей среды города
47. Информационная система службы работы с арендаторами офисного центра
48. Информационная система организатора гастролей музыкантов
49. Информационная система по координации действий волонтеров региона
50. Автоматизированная система электронного документооборота предприятия
51. Автоматизированная система контроля доступа на предприятие
52. Мобильное приложение «Управление зарядкой электромобиля»
53. Мобильное приложение «Электронный справочник селекционера»
54. Мобильное приложение «Помощник путешественника»
55. Электронный школьный журнал

**Учебно-методическая карта учебной дисциплины «Технология разработки программного обеспечения»  
Дневная форма получения высшего образования**

Номер раздела, темы	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество аудиторных часов				Литература	Формы контроля знаний
		Лекции	Лабораторные занятия	Практические занятия	Управляемая самостоятельная работа студента		
1	2	3	4	5	6	7	8
Раздел 1	<b>Архитектура программных систем</b>	6					
Тема 1.1	<i>Основы программной инженерии.</i> Особенности развития компьютерной индустрии. Классификации программного обеспечения и программных проектов. Особенности современных программных проектов. Системный подход как основа проектирования программного обеспечения. Понятие методологии разработки ПО.	2				Осн. лит.: [1], [2]. Доп. лит.: [1], [2].	Блиц-опрос
Тема 1.2	<i>Архитектура программных систем и моделирование.</i> Понятие архитектуры ПО. Критерии хорошей архитектуры. Модульная архитектура. Декомпозиция. Архитектурные представления. Сервисно-ориентированная архитектура.	2				Осн. лит.: [2], [3]. Доп. лит.: [1].	Блиц-опрос
	Моделирование в разработке ПО. Системное моделирование. Цели моделирования. Объектно-ориентированное моделирование.	2				Осн. лит.: [2], [3]. Доп. лит.: [1].	Блиц-опрос
Раздел 2	<b>Проектирование программных систем</b>	20		22			
Тема 2.1	<i>Унифицированный язык моделирования UML.</i> Унифицированный язык моделирования UML, спецификации, история развития. Классификация диаграмм (структурные и поведенческие). SysML как расширение языка UML. Общая иерархия моделей. Назначение метамодели UML ее структура. Механизмы расширения UML (стереотипы, метасвойства, ограничения, комментарии).	2				Осн. лит.: [1], [2], [3]. Доп. лит.: [1], [2].	Контрольное тестирование №1

1	2	3	4	5	6	7	8
Тема 2.2	<b>Диаграмма прецедентов.</b> Диаграмма прецедентов, ее элементы и связи между элементами. Связи ассоциации, включения, расширения, обобщения. Правила составления спецификаций и сценариев. Дополнительные обозначения языка UML для бизнес-моделирования. Область применения диаграммы прецедентов.	2				Осн. лит.: [2], [3]. Доп. лит.: [1].	Блиц-опрос
	<b>Практическая работа №1.</b> Разработка диаграммы прецедентов.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №2.</b> Спецификация требований к программному средству.			2		Методические указания	Защита отчета по практической работе
Тема 2.3	<b>Диаграмма классов.</b> Диаграмма классов, ее элементы и связи между элементами. Статические операции и атрибуты. Интерфейсы и абстрактные классы.	2				Осн. лит.: [1], [2], [5]. Доп. лит.: [1].	Блиц-опрос
	Шаблон класса (параметризованный класс). Область применения диаграммы классов.	2				Осн. лит.: [1], [2], [5]. Доп. лит.: [1].	Блиц-опрос
	<b>Практическая работа №3.</b> Проектирование объектов базы данных.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №4.</b> Разработка диаграммы классов.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №4.</b> Разработка диаграммы классов.			2			
Тема 2.4	<b>Диаграммы взаимодействия.</b> Виды диаграмм взаимодействия и область их применения. Диаграммы взаимодействия, их элементы и связи между элементами.	2				Осн. лит.: [1], [2], [5]. Доп. лит.: [1].	Блиц-опрос
	Диаграммы последовательности. Примеры диаграмм последовательности. Диаграммы кооперации. Примеры диаграмм кооперации Сравнение диаграмм последовательности и кооперативных диаграмм.	2				Осн. лит.: [1], [2], [5]. Доп. лит.: [1].	Блиц-опрос
	<b>Практическая работа №5.</b> Разработка диаграммы последовательности.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №5.</b> Разработка диаграммы последовательности.			2			

1	2	3	4	5	6	7	8
Тема 2.5	<b>Диаграммы деятельности и состояний.</b> Диаграмма деятельности, ее элементы и связи. Дорожки. Область применения диаграмм деятельности. Примеры диаграмм деятельности.	2				Осн. лит.: [3], [4], [5]. Доп. лит.: [1].	Блиц-опрос
Тема 2.5	Диаграмма состояний, ее элементы и связи. Примеры диаграмм состояний.	2				Осн. лит.: [3], [4], [5]. Доп. лит.: [1].	Блиц-опрос
	<b>Практическая работа №6.</b> Разработка диаграммы деятельности.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №6.</b> Разработка диаграммы деятельности.			2			
	<b>Практическая работа №7.</b> Разработка диаграммы состояний.			2		Методические указания	Защита отчета по практической работе
<b>Практическая работа №7.</b> Разработка диаграммы состояний.			2				
Тема 2.7	<b>Шаблоны (паттерны) проектирования.</b> Концепция шаблонов проектирования. Типы шаблонов проектирования: делегирования, функционального дизайна, интерфейса, адаптер.	2				Осн. лит.: [1], [2]. Доп. лит.: [2].	Блиц-опрос
	Шаблоны: абстрактная фабрика, строитель, фабричный метод, фасад, прототип, наблюдатель.	2				Осн. лит.: [2], [4]. Доп. лит.: [2].	Блиц-опрос
Раздел 3	<b>Реализация и сопровождение ПО</b>	<b>8</b>		<b>12</b>			
Тема 3.1	<b>Принципы проектирования пользовательского интерфейса.</b> Виды интерфейсов ПО. Классификация интерфейсов пользователя. Психологическая основа пользовательского интерфейса. Этапы проектирования графического пользовательского интерфейса. программного интерфейса.	2				Осн. лит.: [1], [4]. Доп. лит.: [2].	Контрольное тестирование №2
	Основные компоненты графического пользовательского интерфейса и рекомендации по их применению. Свойства качественного программного интерфейса.	2					
	<b>Практическая работа №8.</b> Проектирование интерфейса пользователя ПО.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №8.</b> Проектирование интерфейса пользователя ПО.			2			
<b>Практическая работа №9.</b> Разработка прототипа объектно-ориентированного ПО.			2		Методические указания	Защита отчета по практической работе	

1	2	3	4	5	6	7	8
	<b>Практическая работа №9.</b> Разработка прототипа объектно-ориентированного ПО.			2			
Тема 3.2	<b>Диаграммы реализации.</b> Виды диаграмм реализации и область их применения. Элементы диаграмм и связи между элементами. Примеры диаграмм компонентов. Примеры диаграмм развертывания.	2				Осн. лит.: [3], [4]. Доп. лит.: [2].	Блиц-опрос
	<b>Практическая работа №10.</b> Разработка диаграммы компонентов.			2		Методические указания	Защита отчета по практической работе
	<b>Практическая работа №11.</b> Разработка диаграммы развертывания.			2		Методические указания	Защита отчета по практической работе
Тема 3.3.	<b>Сопровождение ПО. Управление конфигурацией ПО.</b> Процесс сопровождения ПО. Элементы процесса сопровождения ПО. Элементы конфигурации программного средства. Управление конфигурацией и версиями. Документирование в процессах жизненного цикла ПО.	2				Осн. лит.: [2], [4]. Доп. лит.: [2].	Контрольное тестирование №3
	<b>Всего (68 часов)</b>	<b>34</b>		<b>34</b>			

Примечание: в соответствии с рейтинговой системой для определения результата текущего контроля за семестр в виде отметки в баллах по десятибалльной шкале используются отметки, полученные за мероприятия текущего контроля в течение семестра, обозначенные в графе «Форма контроля знаний»

**ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ****ЛИТЕРАТУРА****Основная:**

1. Белов, В.В. Проектирование информационных систем : учебник / В. В. Белов, В. И. Чистякова. – Москва: КУРС, 2021. – 394 с. – Рекомендовано Научно-методическим советом ФГБОУ ВО "РГРТУ" в качестве учебника для студентов высших учебных заведений, обучающихся по направлению подготовки 2.09.03.03 "Прикладная информатика".

2. Лаврищева, Е.М. Программная инженерия. Парадигмы, технологии и CASE-средства: учебник для вузов / Е. М. Лаврищева. – Москва: Юрайт, 2024. – 280 с.

3. Волк, В.К. Практическое введение в программную инженерию: учебное пособие для вузов – СПб.: Лань, 2022. – 100 с. – Текст: электронный // Лань : электронно-библиотечная система. URL: <https://e.lanbook.com/book/249848> (дата обращения: 19.09.2023).

4. Заботина, Н. Н. Методы и средства проектирования информационных систем: учебное пособие / Н.Н. Заботина. – Москва : ИНФРА-М, 2023. – 331 с. – Текст : электронный. - URL: <https://znanium.ru/catalog/product/1902833> (дата обращения: 19.09.2023).

5. Шибут, М.С. Проектирование информационных систем: пособие / М. С. Шибут; Академия управления при Президенте Республики Беларусь. - Минск: Акад. упр. при Президенте РБ, 2021. – 174 с. - Рекомендовано Учебно-методическим объединением по образованию в области управления для студентов учреждений высшего образования, осваивающих образовательную программу высшего образования I ступени по специальности 1-26 03 01 Управление информационными системами в качестве пособия.

**Дополнительная:**

1. Вейцман, В.М. Проектирование информационных систем: учебное пособие для вузов – СПб.: Лань, 2022. – 316 с. – Текст: электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/208946> (дата обращения: 19.09.2023).

2. Гринченко, Н.Н. Проектирование информационных систем : учебник / Н. Н. Гринченко, А. Ю. Громов. – Москва : КУРС, 2021. – 175 с. – Рекомендовано Научно-методическим советом РГРТУ в качестве учебника для студентов, обучающихся по направлению подготовки 2.09.03.03 "Математическое обеспечение и администрирование информационных систем" 1.02.03.03, 1.02.04.03 (уровни 3,4) (квалификация "Бакалавр").

*Елена Турнова Е. В.*

**ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАНЯТИЙ**

- Практическая работа №1* Разработка диаграммы прецедентов.
- Практическая работа № 2* Спецификация требований к программному средству.
- Практическая работа №3* Проектирование объектов базы данных.
- Практическая работа №4* Разработка диаграммы классов.
- Практическая работа №5* Разработка диаграммы последовательности.
- Практическая работа №6* Разработка диаграммы деятельности.
- Практическая работа №7* Разработка диаграммы состояний.
- Практическая работа №8* Проектирование интерфейса пользователя ПО.
- Практическая работа №9* Разработка прототипа объектно-ориентированного ПО.
- Практическая работа №10* Разработка диаграммы компонентов.
- Практическая работа №11* Разработка диаграммы развертывания.

## ПЕРЕЧЕНЬ ВОПРОСОВ ДЛЯ ПРОВЕДЕНИЯ ЭКЗАМЕНА

1. Особенности развития компьютерной индустрии. Классификации программного обеспечения и программных проектов.
2. Особенности современных программных проектов.
3. Системный подход как основа проектирования программного обеспечения.
4. Понятие архитектуры ПО. Критерии хорошей архитектуры.
5. Модульная архитектура. Декомпозиция.
6. Архитектурные представления. Сервисно-ориентированная архитектура.
7. Моделирование в разработке ПО. Системное моделирование. Цели моделирования. Объектно-ориентированное моделирование.
8. Унифицированный язык моделирования UML, спецификации, история развития. Классификация диаграмм.
9. Общая иерархия моделей. Назначение метамодели UML ее структура.
10. Механизмы расширения UML (стереотипы, метасвойства, ограничения, комментарии).
11. Диаграмма прецедентов, ее элементы и связи между элементами. Область применения диаграммы прецедентов.
12. Правила составления спецификаций и сценариев. Дополнительные обозначения языка UML для бизнес-моделирования.
13. Диаграмма классов, ее элементы и связи между элементами. Область применения диаграммы классов.
14. Статические операции и атрибуты.
15. Интерфейсы и абстрактные классы. Шаблон класса (параметризованный класс).
16. Виды диаграмм взаимодействия и область их применения. Диаграммы взаимодействия, их элементы и связи между элементами.
17. Диаграммы последовательности. Диаграммы кооперации. Сравнение диаграмм последовательности и кооперативных диаграмм.
18. Диаграмма деятельности, ее элементы и связи. Дорожки. Область применения диаграмм деятельности.
19. Диаграмма состояний, ее элементы и связи. Диаграмма состояний.
20. Виды диаграмм реализации и область их применения. Диаграммы реализации, их элементы и связи между элементами.
21. Концепция шаблонов проектирования.
22. Тип шаблонов проектирования делегирования.
23. Тип шаблонов проектирования функционального дизайна.
24. Тип шаблонов проектирования интерфейса.
25. Тип шаблонов проектирования адаптер.
26. Шаблон абстрактная фабрика.
27. Шаблон строитель.
28. Шаблон фабричный метод.
29. Шаблон фасад.
30. Шаблон прототип.
31. Шаблон наблюдатель.
32. Виды интерфейсов ПО. Классификация интерфейсов пользователя.
33. Психологическая основа пользовательского интерфейса.
34. Этапы проектирования графического пользовательского интерфейса.
35. Основные компоненты графического пользовательского интерфейса и рекомендации по их применению.
36. Свойства качественного программного интерфейса.
37. Процесс сопровождения ПО. Элементы процесса сопровождения ПО.
38. Элементы конфигурации программного средства.
39. Управление конфигурацией и версиями.
40. Документирование в процессах жизненного цикла ПО.

## **ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ**

Обучение дисциплине «Технология разработки программного обеспечения» предполагает реализацию следующих форм самостоятельной работы студентов:

- проработка конспекта лекций и учебной литературы;
- изучение печатных источников по теме дисциплины;
- изучение профессиональных электронных ресурсов по теме дисциплины;
- изучение вопросов для самоконтроля;
- подготовка к аудиторному выполнению практических работ (предварительное знакомство с методическими указаниями, вариантом индивидуального задания по работе);
- решение индивидуальных задач при подготовке к практическим занятиям;
- подготовка к защите практических работ (оформление отчёта по индивидуальному варианту задания, защита результатов работы и демонстрации степени освоения навыков и умений по конкретной теме);
- углублённое изучение отдельных тем учебной дисциплины для подготовки к устным опросам;
- изучение основной и дополнительной литературы в процессе подготовки к анализу и решению проблемных задач, реализации элементов исследовательской деятельности;
- систематизация полученных знаний при подготовке к экзамену.

Условия для самостоятельной работы студентов, в частности, для развития навыков самоконтроля, способствующих интенсификации образовательного процесса, обеспечиваются:

- наличием и использованием в образовательном процессе учебного курса по дисциплине «Технология разработки программного обеспечения» в системе дистанционного обучения Moodle для доступа студентов к электронным вариантам курса лекций и учебно-методических указаний по основным разделам дисциплины, для организации учебной деятельности студентов и контроля ее результативности.

### **Дополнительное учебно-методическое обеспечение самостоятельной работы студентов очной формы обучения**

1. Учебный курс по дисциплине «Технология разработки программного обеспечения» в системе дистанционного обучения Moodle (ссылка <https://moodle.psu.by/course/view.php?id=263>).

2. Методические указания к выполнению практических работ по дисциплине «Технология разработки программного обеспечения» для студентов специальности 6-05-0533-12 «Кибербезопасность».

3. Методические указания к выполнению курсового проекта по дисциплине «Технология разработки программного обеспечения» для студентов специальности 6-05-0533-12 «Кибербезопасность».

## Содержание самостоятельной работы студентов

Вид самостоятельной работы	Тематическое содержание и используемые источники	Количество часов
1	2	3
Самостоятельное изучение отдельных вопросов по темам дисциплины	<p><i>Тема 1.1. Основы программной инженерии.</i> Особенности развития компьютерной индустрии. Классификации программного обеспечения и программных проектов. Особенности современных программных проектов. Системный подход как основа проектирования программного обеспечения. Понятие методологии разработки ПО.</p> <p>Осн. лит.: [1], [2]. Доп. лит.: [1], [2].</p>	1
	<p><i>Тема 1.2. Архитектура программных систем и моделирование.</i> Понятие архитектуры ПО. Критерии хорошей архитектуры. Модульная архитектура. Декомпозиция. Архитектурные представления. Сервисно-ориентированная архитектура. Моделирование в разработке ПО. Системное моделирование. Цели моделирования. Объектно-ориентированное моделирование.</p> <p>Осн. лит.: [2], [3]. Доп. лит.: [1].</p>	1
	<p><i>Тема 2.1. Унифицированный язык моделирования UML.</i> Унифицированный язык моделирования UML, спецификации, история развития. Классификация диаграмм (структурные и поведенческие). SysML как расширение языка UML. Общая иерархия моделей. Назначение метамодели UML ее структура. Механизмы расширения UML (стереотипы, метасвойства, ограничения, комментарии).</p> <p>Осн. лит.: [1], [2], [3]. Доп. лит.: [1], [2].</p>	1
	<p><i>Тема 2.2. Диаграмма прецедентов.</i> Диаграмма прецедентов, ее элементы и связи между элементами. Связи ассоциации, включения, расширения, обобщения. Правила составления спецификаций и сценариев. Дополнительные обозначения языка UML для бизнес-моделирования. Область применения диаграммы прецедентов.</p> <p>Осн. лит.: [2], [3]. Доп. лит.: [1].</p>	1
	<p><i>Тема 2.3. Диаграмма классов.</i> Диаграмма классов, ее элементы и связи между элементами. Статические операции и атрибуты. Интерфейсы и абстрактные классы. Шаблон класса (параметризованный класс). Область применения диаграммы классов.</p> <p>Осн. лит.: [1], [2], [5]. Доп. лит.: [1].</p>	1

1	2	3
	<p><i>Тема 2.4. Диаграммы взаимодействия.</i>            Виды диаграмм взаимодействия и область их применения. Диаграммы взаимодействия, их элементы и связи между элементами. Диаграммы последовательности. Примеры диаграмм последовательности. Диаграммы кооперации. Примеры диаграмм кооперации Сравнение диаграмм последовательности и кооперативных диаграмм.</p> <p>Осн. лит.: [1], [2], [5]. Доп. лит.: [1].</p>	1
	<p><i>Тема 2.5. Диаграммы деятельности и состояний.</i>            Диаграмма деятельности, ее элементы и связи. Дорожки. Область применения диаграмм деятельности. Примеры диаграмм деятельности. Диаграмма состояний, ее элементы и связи. Примеры диаграмм состояний.</p> <p>Осн. лит.: [3], [4], [5]. Доп. лит.: [1].</p>	1
	<p><i>Тема 2.6. Шаблоны (паттерны) проектирования.</i>            Концепция шаблонов проектирования. Типы шаблонов проектирования: делегирования, функционального дизайна, интерфейса, адаптер. Шаблоны: абстрактная фабрика, строитель, фабричный метод, фасад, прототип, наблюдатель.</p> <p>Осн. лит.: [1], [2], [4]. Доп. лит.: [2].</p>	1
	<p><i>Тема 3.1. Принципы проектирования пользовательского интерфейса.</i>            Виды интерфейсов ПО. Классификация интерфейсов пользователя. Психологическая основа пользовательского интерфейса. Этапы проектирования графического пользовательского интерфейса. Основные компоненты графического пользовательского интерфейса и рекомендации по их применению. Свойства качественного программного интерфейса.</p> <p>Осн. лит.: [1], [2], [4]. Доп. лит.: [2].</p>	1
	<p><i>Тема 3.2. Диаграммы реализации.</i>            Виды диаграмм реализации и область их применения. диаграмм реализации, Элементы диаграмм и связи между элементами. Примеры диаграмм компонентов. Примеры диаграмм развертывания.</p> <p>Осн. лит.: [3], [4]. Доп. лит.: [2].</p>	1
	<p><i>Тема 3.3. Сопровождение ПО. Управление конфигурацией ПО.</i>            Процесс сопровождения ПО. Элементы процесса сопровождения ПО. Элементы конфигурации программного средства. Управление конфигурацией и версиями. Документирование в процессах жизненного цикла ПО.</p> <p>Осн. лит.: [1], [2], [4]. Доп. лит.: [2].</p>	1
Подготовка к защите отчетов по практическим работам	<p><i>Практическая работа №1</i> Разработка диаграммы прецедентов.</p>	2
	<p><i>Практическая работа № 2</i> Спецификация требований к программному средству.</p>	2
	<p><i>Практическая работа №3</i> Проектирование объектов базы данных.</p>	2

1	2	3
	<i>Практическая работа №4</i> Разработка диаграммы классов.	2
	<i>Практическая работа №5</i> Разработка диаграммы последовательности.	2
	<i>Практическая работа №6</i> Разработка диаграммы деятельности.	2
	<i>Практическая работа №7</i> Разработка диаграммы состояний.	2
	<i>Практическая работа №8</i> Проектирование интерфейса пользователя ПО.	2
	<i>Практическая работа №9</i> Разработка прототипа объектно-ориентированного ПО.	3
	<i>Практическая работа №10</i> Разработка диаграммы компонентов.	1
	<i>Практическая работа №11</i> Разработка диаграммы развертывания.	1
	Подготовка к экзамену	8
		<b>40</b>

## КОНТРОЛЬ КАЧЕСТВА УСВОЕНИЯ ЗНАНИЙ

Контроль качества усвоения знаний проводится в форме текущего контроля и промежуточной аттестации.

Мероприятия текущего контроля проводятся в течение семестра и включают в себя следующие **формы контроля**:

- устная форма (блиц-опрос на лекциях);
- письменная форма (тесты, письменные отчёты по практическим работам);
- устно-письменная форма (отчёты по практическим работам с их устной защитой);
- техническая форма (электронные тесты).

Практические работы предполагают выполнение и защиту. При их выполнении выдаётся индивидуальное задание. Отчёт по практической работе представляется в электронном виде. Содержание отчёта: название работы, вариант задания, анализ задания, ход выполнения работы, основные и промежуточные результаты, выводы по работе. Защита работ проводится индивидуально и оценивается в соответствии с установленными правилами.

Результат текущего контроля за семестр оценивается отметкой в баллах по десятибалльной шкале и выводится, исходя из отметок, выставленных в ходе проведения мероприятий текущего контроля в течение семестра по следующей формуле:

$$T = \frac{(KT_1 + \dots + KT_{n1}) + (PP_1 + \dots + PP_{n2}) + (YO_1 + \dots + YO_{n3})}{(n1 + n2 + n3)},$$

где  $KT_1, \dots, KT_{n1}$  – отметки, выставленные по результатам контрольного тестирования;  $n1$  – количество тестов;  $PP_1, \dots, PP_{n2}$  – отметки, выставленные по результатам защит практических работ;  $n2$  – количество работ;  $YO_1, \dots, YO_{n3}$  – отметки, выставленные по результатам устных опросов на лекциях;  $n3$  – количество устных опросов.

Результат текущего контроля рассчитывается как округлённое среднее значение.

Для обучающего, пропустившего мероприятие текущего контроля по уважительной причине, кафедрой устанавливаются дополнительные сроки.

Обучающемуся, пропустившему мероприятие текущего контроля без уважительной причины, выставляется 1 (один) балл за данное мероприятие.

Результат текущего контроля может быть повышен:

- за участие обучающего в научно-практических мероприятиях, учебно-исследовательской, научно-исследовательской работе студентов (конференциях, семинарах, олимпиадах, конкурсах, научных кружках и т.п.) по профилю учебной дисциплины (модуля) и может быть повышен до 10 баллов при достижении значимых результатов в этой работе;

- обучающийся в целях повышения отметки по любому мероприятию текущего контроля может воспользоваться правом на дополнительные образовательные услуги (платные консультации, платные дополнительные занятия). Количество и сроки пересдач с целью повышения отметки определяет кафедра.

Промежуточная аттестация проводится в форме экзамена.

Итоговая экзаменационная отметка по дисциплине рассчитывается по формуле:

$$ИЭ = k \cdot T + (1 - k) \cdot O,$$

где  $k$  – весовой коэффициент текущего контроля;  $T$  – результат текущего контроля за семестр;  $O$  – отметка, полученная студентом на экзамене за ответ по билету.

Весовой коэффициент принимается равным  $k = 0.5$ .

Результат итоговой экзаменационной отметки округляется до целого значения.

Информация о весовом коэффициенте доводится до студентов на первом занятии в семестре. Положительной является отметка не ниже 4 баллов.

## ХАРАКТЕРИСТИКА ИННОВАЦИОННЫХ ПОДХОДОВ К ПРЕПОДАВАНИЮ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основные методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

- проблемное обучение (проблемное изложение, вариативное изложение), реализуемое на лекционных занятиях;
- учебно-исследовательская деятельность, реализация творческого подхода, реализуемые на практических занятиях.

Используемые технологии обучения и диагностики компетенций в преподавании дисциплины «Технология разработки программного обеспечения» реализуют подход, основанный на максимально возможном использовании внутренней и учебной мотивации студента, проявляющейся в чётком понимании им значимости всех видов выполняемых работ, как с точки зрения важности для профессиональной подготовки, так и с точки зрения оценивания. Подход предполагает использование элементов проблемного обучения и элементов исследовательской деятельности студентов в процессе аудиторной работы, а также при выполнении самостоятельных работ при постоянном рейтинговом контроле.

На лекционных занятиях по дисциплине «Технология разработки программного обеспечения» возможно использование элементов проблемного обучения: проблемное изложение некоторых аспектов, использование частично-поискового метода.

Изучение учебной дисциплины осуществляется на лекционных и практических занятиях.

На лекционных занятиях студенты овладевают системой теоретических знаний в области программной инженерии. В ходе лекционного изложения теоретических сведений используются традиционные словесные приёмы и методы, которые активизируются постановкой проблемных вопросов и заданий, организацией учебных дискуссий с опорой на имеющуюся начальную подготовку студентов и их математический кругозор, использованием интерактивных методов обучения.

На практических занятиях развиваются и формируются необходимые практические умения и навыки проектирования архитектуры программных средств, моделирования и разработки программных систем; CASE-системами проектирования программного обеспечения.

Применяется индивидуальный, творческий подход. Также во время проведения практических работ особое внимание уделяется формированию у студентов умения планировать свою работу и определять эффективную последовательность её выполнения.

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ  
С ДРУГИМИ УЧЕБНЫМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

Название дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу
Дипломное проектирование	Кафедра математики и компьютерной безопасности	<i>нет</i>	
Системы управления базами данных		<i>нет</i>	

Заведующий кафедрой математики и компьютерной безопасности, к.т.н., доцент



И. Б. Бураченко