

Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный университет промышленных технологий и дизайна»
(СПбГУПТД)

УТВЕРЖДАЮ
Первый проректор, проректор
по УР

_____ А.Е. Рудин

Рабочая программа дисциплины

ФТД.01

Историко-методологические основы программной инженерии

Учебный план: 2026-2027 09.03.04 ИИТА Программ инженерии ОО №1-1-195.plx

Кафедра: **33** Цифровых и аддитивных технологий

Направление подготовки:
(специальность) 09.03.04 Программная инженерия

Профиль подготовки:
(специализация) Программная инженерия

Уровень образования: бакалавриат

Форма обучения: очная

План учебного процесса

Семестр (курс для ЗАО)		Контактная работа обучающихся		Сам. работа	Контроль, час.	Трудоёмкость, ЗЕТ	Форма промежуточной аттестации
		Лекции	Практ. занятия				
5	УП	16	16	39,75	0,25	2	Зачет
	РПД	16	16	39,75	0,25	2	
Итого	УП	16	16	39,75	0,25	2	
	РПД	16	16	39,75	0,25	2	

Санкт-Петербург
2026

Рабочая программа дисциплины составлена в соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.04 Программная инженерия, утверждённым приказом Минобрнауки России от 19.09.2017 г. № 920

Составитель (и):

к.т.н., доцент

Якуничева Елена
Николаевна

От кафедры составителя:

Заведующий кафедрой цифровых и аддитивных технологий

Сошников Антон
Владимирович

От выпускающей кафедры:

Заведующий кафедрой

Сошников Антон
Владимирович

Методический отдел:

1 ВВЕДЕНИЕ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

1.1 Цель дисциплины: Формирование у обучающихся целостного научно-методологического фундамента, основанного на понимании исторического развития, ключевых концепций и парадигм программной инженерии, необходимого для осознанного выбора инженерных решений, критического анализа современных подходов и прогнозирования технологических трендов.

1.2 Задачи дисциплины:

Познакомить студентов с историческими этапами становления программной инженерии как научной и профессиональной дисциплины.

Сформировать умение критически анализировать и синтезировать историко-методологическую информацию.

Развить навыки применения системного подхода к интерпретации эволюции методологий разработки ПО.

Научить использовать исторический опыт для обоснования современных инженерных практик и принятия решений в условиях неопределённости.

Обеспечить понимание взаимосвязи между технологическими, экономическими, социальными и методологическими факторами в развитии ПО.

1.3 Требования к предварительной подготовке обучающегося:

Предварительная подготовка предполагает создание основы для формирования компетенций, указанных в п. 2, при изучении дисциплин:

Алгоритмизация и программирование

Системное программирование

Математика

Теория систем и системный анализ

Прикладная статистика

2 КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

УК-1: Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	
Знать:	Историко-эволюционные этапы формирования и развития программной инженерии как научной и профессиональной дисциплины.
Уметь:	Интерпретировать исторические и методологические материалы в целях обоснования современных инженерных решений и прогнозирования их развития.
Владеть:	Навыками формирования научно-методологического фундамента для принятия обоснованных инженерных решений в условиях неопределённости, изменчивости требований и технологической эволюции.

3 РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Наименование и содержание разделов, тем и учебных занятий	Семестр (курс для ЗАО)	Контактная работа		СР (часы)	Инновац. формы занятий	Форма текущего контроля
		Лек. (часы)	Пр. (часы)			
Раздел 1. Историческое становление программной инженерии	5					О
Тема 1. Предыстория программирования: от механических вычислителей к первым ЭВМ. Практические занятия: Анализ биографий и вклада пионеров вычислительной техники (Бэббидж, Лавлейс, Тьюринг).		2	2	4		
Тема 2. Кризис программного обеспечения 1960–1970-х гг. и рождение термина «программная инженерия». Практические занятия: Сравнительный анализ докладов конференции НАТО 1968 и 1969 гг.		2	2	5		
Тема 3. Классические методологии: водопадная модель, структурное проектирование, верификация. Практические занятия: Реконструкция проектной документации по классической модели разработки.		2	2	5		

Тема 4. Роль стандартов (ISO, IEEE, ГОСТ) в институционализации программной инженерии. Практические занятия: Анализ соответствия исторических проектов стандартам своего времени.		2	2	5		
Раздел 2. Методологическая эволюция и современные парадигмы						
Тема 5. Переход к гибким методологиям: причины, критика классики, Agile-манифест. Практические занятия: Анализ кейсов провалов «водопадных» проектов и перехода на Agile.		2	2	5		
Тема 6. Эволюция качества ПО: от верификации к непрерывной доставке и observability. Практические занятия: Сравнение подходов к обеспечению качества в 1980-х, 2000-х и 2020-х гг.		2	2	5		
Тема 7. Влияние аппаратных и сетевых технологий на методологии разработки. Практические занятия: Построение хронологической карты «технология → методология → практика».		2	2	5		
Тема 8. Программная инженерия в условиях неопределённости: прогнозирование, адаптация, этика. Практические занятия: Разработка рекомендаций по выбору методологии для инновационного стартапа с учётом исторического опыта.		2	2	5,75		
Итого в семестре (на курсе для ЗАО)		16	16	39,75		
Консультации и промежуточная аттестация (Зачет)		0,25				
Всего контактная работа и СР по дисциплине		32,25		39,75		

4 КУРСОВОЕ ПРОЕКТИРОВАНИЕ

Курсовое проектирование учебным планом не предусмотрено

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

5.1 Описание показателей, критериев и системы оценивания результатов обучения

5.1.1 Показатели оценивания

Код компетенции	Показатели оценивания результатов обучения	Наименование оценочного средства
УК-1	Раскрывает основные принципы нахождения, систематизации и критического оценивания исторических источников по ПО. Выявляет причинно-следственные связи между технологическими, методологическими и социальными факторами. Представляет обоснование современных решений через призму исторического опыта.	

5.1.2 Система и критерии оценивания

Шкала оценивания	Критерии оценивания сформированности компетенций	
	Устное собеседование	Письменная работа
Зачтено	Обучающийся своевременно выполнил практические задания и представил результаты в форме презентации; отвечает на теоретический вопрос по материалам лекций, возможно допуская несущественные ошибки. Учитываются баллы, накопленные в течение семестра.	

Не зачтено	Обучающийся своевременно не выполнил (выполнил частично) практические задания и не представил результаты в форме презентации; при ответе на вопрос преподавателя допустил существенные ошибки. Не учитываются баллы, накопленные в течение семестра.	
------------	--	--

5.2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

5.2.1 Перечень контрольных вопросов

№ п/п	Формулировки вопросов
Семестр 5	
1	Какие предпосылки привели к появлению программирования как деятельности?
2	Кто считается первым программистом и почему?
3	В чём суть «кризиса программного обеспечения»?
4	Почему именно в 1968 году был введён термин «программная инженерия»?
5	Какие ключевые проблемы обсуждались на конференции НАТО по ПО?
6	Какие характеристики отличали ранние программные проекты от инженерных дисциплин?
7	Какие принципы лежали в основе водопадной модели?
8	Что такое структурное проектирование и как оно влияло на качество ПО?
9	Какие методы верификации использовались в 1970–1980-х гг.?
10	Какие ограничения имела классическая модель разработки?
11	Какие организации способствовали стандартизации ПО?
12	Какую роль сыграли стандарты IEEE в развитии ПО?
13	Как ГОСТы регулировали разработку ПО в СССР и РФ?
14	Что означает «институционализация» программной инженерии?
15	Как менялось понимание «инженерии» применительно к ПО?
16	Какие социальные и экономические факторы влияли на развитие ПО в XX веке?
17	Какие ошибки в управлении проектами способствовали кризису ПО?
18	Почему программирование долго не признавалось инженерной дисциплиной?
19	Какие технологии стимулировали переход от «хакерства» к инженерии?
20	Какие уроки современная инженерия извлекла из раннего опыта?
21	Какие недостатки классических методологий привели к появлению Agile?
22	Что провозглашает Agile-манифест и как он изменил подход к разработке?
23	Какие экономические и рыночные факторы способствовали гибким методологиям?
24	В чём различие между итеративной и инкрементальной разработкой?
25	Как менялось понимание «качества ПО» с течением времени?
26	Что такое непрерывная интеграция и доставка (CI/CD) и как она связана с историей ПО?
27	Как развитие облачных технологий повлияло на архитектурные подходы?
28	Почему микросервисы стали возможны только в определённый исторический период?
29	Как IoT и edge computing ставят новые методологические вызовы?
30	Какие этические дилеммы возникли с развитием ПО?
31	Как исторический опыт помогает прогнозировать технологические тренды?
32	Что такое «технический долг» и как он связан с методологическим выбором?
33	Как менялись роли в команде разработки с течением времени?
34	Какие факторы определяют выбор методологии в условиях неопределённости?
35	Какие ошибки повторяются в инновационных проектах несмотря на исторический опыт?
36	Какую роль играет культура в выборе методологии разработки?
37	Как глобализация и распределённые команды повлияли на методологии?
38	Какие современные подходы к управлению знаниями в ПО имеют исторические корни?
39	Какие методологические «моды» оказались временными, а какие — устойчивыми?
40	Как использовать исторический контекст для обоснования инженерного решения сегодня?

5.2.2 Типовые тестовые задания

Не предусмотрено.

5.2.3 Типовые практико-ориентированные задания (задачи, кейсы)

1. Написать аналитическую справку о вкладе одного из пионеров вычислительной техники в развитие ПО.
2. Проанализировать эволюцию подходов к тестированию ПО за 50 лет и выявить устойчивые принципы.

5.3 Методические материалы, определяющие процедуры оценивания знаний, умений, владений (навыков и (или) практического опыта деятельности)

5.3.1 Условия допуска обучающегося к промежуточной аттестации и порядок ликвидации академической задолженности

Проведение промежуточной аттестации регламентировано локальным нормативным актом СПбГУПТД «Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся»

5.3.2 Форма проведения промежуточной аттестации по дисциплине

Устная ☒ Письменная ☐ Компьютерное тестирование ☐ Иная ☐

5.3.3 Особенности проведения промежуточной аттестации по дисциплине

Время на подготовку к устному собеседованию составляет 30 минут;
Выполнение тестовых заданий осуществляется за 30 минут.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Учебная литература

Автор	Заглавие	Издательство	Год издания	Ссылка
6.1.1 Основная учебная литература				
Кознов, Д. В.	Введение в программную инженерию	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2025	https://www.iprbooks.hop.ru/133932.html
Доррер, Г. А.	Методология программной инженерии	Красноярск: Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева	2021	https://www.iprbooks.hop.ru/116641.html
6.1.2 Дополнительная учебная литература				
Носова, Л. С.	Основы программной инженерии	Челябинск, Саратов: Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа	2019	https://www.iprbooks.hop.ru/81488.html

6.2 Перечень профессиональных баз данных и информационно-справочных систем

Информационная система «Единое окно доступа к образовательным ресурсам. Раздел. Информатика и информационные технологии» [Электронный ресурс]. URL: http://window.edu.ru/catalog/?p_rubr=2.2.75.6
Электронно-библиотечная система IPRbooks [Электронный ресурс]. URL: <http://www.iprbookshop.ru/>

6.3 Перечень лицензионного и свободно распространяемого программного обеспечения

MicrosoftOfficeProfessional
Microsoft Windows

6.4 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Аудитория	Оснащение
Компьютерный класс	Мультимедийное оборудование, компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду
Лекционная аудитория	Мультимедийное оборудование, специализированная мебель, доска