**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Технологии разработки программного обеспечения

Software Developing Technologies

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 3

Регистрационный номер рабочей программы: 057526

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Целью учебных занятий по дисциплине является формирование у учащихся знаний о процессах, практиках и стандартах, принятых в индустрии при работе над проектами по разработке программного обеспечения (ПО).

Задачами учебных занятий по дисциплине являются:

• сформировать у обучающихся систему знаний о жизненном цикле ПО и основных его этапах, об актуальных методологиях разработки ПО и применяемых в них практиках, о работе по выявлению и управлению требованиями, о выделении рисков проекта и построении стратегий противодействия им, о критериях качества ПО и методах повышения качества кода в проектах, о групповой разработке и правилах построения команд, о практиках эффективного управления проектами, о методах и средствах проектирования пользовательских интерфейсов и архитектуры ПО;

• сформировать умения эффективно пользоваться системами конфигурационного управления, инструментами сборки проекта, системами непрерывной интеграции кода;

• сформировать навыки декомпозиции проектов, оценки задач, построения графика проекта, написания читаемого и сопровождаемого кода, создания модульных и других тестов.

**1.2. Требования подготовленности обучающегося к освоению содержания учебных занятий (пререквизиты)**

Учебные занятия по дисциплине «Технологии разработки программного обеспечения» проводятся в 8-м семестре для обучающихся бакалавриата по направлению подготовки 02.03.01 «Математика и компьютерные науки».

Для успешного освоения дисциплины обучающийся должен иметь предварительную подготовку по дисциплинам информатики и программирования, изучаемых на I-II курсах математико-механического факультета СПбГУ.

**1.3. Перечень результатов обучения (learning outcomes)**

В результате освоения дисциплины обучающиеся должны:

• обладать представлением об общей структуре хода проекта по разработке ПО;

• уметь идентифицировать основные роли в команде и их области ответственности;

• обладать навыками декомпозиции и оценки задач, построения графика работ и оценки бюджета проекта;

• понимать важность работы с требованиями, знать основные виды деятельности при сборе и управлении требованиями;

• уметь обоснованно выбрать методологию разработки проекта и парадигму программирования в зависимости от его параметров и требований;

• иметь представление о подходах и инструментах проектирования макетов пользовательского интерфейса приложения;

• осознавать важность архитектуры проекта, понимать критерии оценки качества архитектуры и методов её проектирования;

• понимать особенности работы в команде, уметь использовать средства контроля версий для коллективной разработки ПО;

• уметь обнаруживать, классифицировать и отслеживать дефекты в ПО;

• понимать основные характеристики качества ПО и методы по его достижению.

**1.4. Перечень и объём активных и интерактивных форм учебных занятий**

В качестве основных интерактивных форм предполагается:

Проведение практических занятий, консультаций;

Проведение лекций, основная часть лекционного занятия проводится в интерактивных формах; при этом до 75% лекционных занятий проводится в активных формах;

Самостоятельная работа с использованием методических материалов.

**Раздел 2. Организация, структура и содержание учебных занятий**

**2.1. Организация учебных занятий**

**2.1.1 Основной курс**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Трудоёмкость, объёмы учебной работы и наполняемость групп обучающихся | | | | | | | | | | | | | | | | | | |
| Код модуля в составе дисциплины,  практики и т.п. | Контактная работа обучающихся с преподавателем | | | | | | | | | | | | Самостоятельная работа | | | | Объём активных и интерактивных  форм учебных занятий | Трудоёмкость |
| лекции | семинары | консультации | практические  занятия | лабораторные работы | контрольные работы | коллоквиумы | текущий контроль | промежуточная  аттестация | итоговая аттестация | под руководством преподавателя | в присутствии  преподавателя | сам. раб. с использованием  методических материалов | текущий контроль (сам.раб.) | промежуточная аттестация (сам.раб.) | итоговая аттестация  (сам.раб.) |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | | | | | | | | | | | | | |
| Форма обучения: очная | | | | | | | | | | | | | | | | | | |
| Семестр 8 | 24 |  | 2 | 24 |  |  |  |  | 2 |  |  |  | 40 |  | 16 |  | 20 | 3 |
|  | 2-100 |  | 2-100 | 10-25 |  |  |  |  | 2-100 |  |  |  | 1-1 |  | 1-1 |  |  |  |
| ИТОГО | 24 |  | 2 | 24 |  |  |  |  | 2 |  |  |  | 40 |  | 16 |  |  | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Виды, формы и сроки текущего контроля успеваемости и промежуточной аттестации | | | | | | |
| Код модуля в составе дисциплины, практики и т.п. | Формы текущего контроля успеваемости | | Виды промежуточной аттестации | | Виды итоговой аттестации  (только для программ итоговой аттестации и дополнительных образовательных программ) | |
| Формы | Сроки | Виды | Сроки | Виды | Сроки |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | |
| Форма обучения: очная | | | | | | |
| Семестр 8 |  |  | экзамен, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

1. Программа и программный продукт, понятие программной инженерии, состав команды, виды деятельности разработчика ПО.
2. Жизненный цикл ПО, модели ЖЦ. Rational Unified Process и eXtreme Programming как примеры методологий разработки.
3. Процессный фреймворк Scrum: роли, артефакты, основные практики и мероприятия.
4. Виды и характеристики требований. Инженерия требований: выявление, анализ, провека, спецификация, управление требованиями.
5. Проектирование GUI. Понятие User Experience, User-centered design, Activity-centred design. Персонажи и сценарии. Подходы к прототипированию пользовательских интерфейсов: storytelling, бумажные прототипы, bodystorming, макеты, дизайн-макеты, интерактивные прототипы. Подходы к исследованию удобства использования продукта, юзабилити-исследования.
6. Функции менеджера проекта: определение проекта, планирование, управление и контроль. Матрица ответственностей, план коммуникаций. Управление рисками. Декомпозиция проекта.
7. Построение графика работ: матрица зависимостей, сетевой график, диаграмма Гантта. Оценка задач и графика в целом. Типичные ошибки при оценке проектов. Треугольник равновесия проекта.
8. Балансирование равновесия проекта: основные подходы и практики. Отслеживание прогресса хода проекта и реагирование на отклонения от плана. Организация команды: типы команд, роли участников.
9. Расчёт бюджета проекта, варианты финансового взаимодействия с заказчиком. Инструментальная поддержка для управления проектами.
10. Основные парадигмы программирования, их особенности и применимость в реальных промышленных проектах.
11. Роль проектирования и архитектора в производстве ПО. Функции архитектора. Архитектура и жизненный цикл ПО. Декомпозиция. Базовые принципы проектирования.
12. Принципы объектно-ориентированного проектирования. Критерии хорошего кода.
13. Групповая разработка. Управление версиями: централизованные и распределённые системы. Удачные модели коллективной разработки ПО в распределённых системах контроля версий.
14. Понятие дефекта. Методы и системы управления дефектами в программных проектах. Характеристики дефектов.
15. Качество ПО: характеристики качества, стандарты. Оценка качества ПО: основные метрики. Методы обеспечения качества ПО.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Посещение учащимися лекций и практических занятий, работа с основной и дополнительной литературой, рекомендованной преподавателем. В рамках лекционного материала основное внимание уделяется изложению теоретических основ курса.

**3.1.2 Методическое обеспечение самостоятельной работы**

Основными видами занятий при изучении дисциплины являются самостоятельная работа обучающихся.

**3.1.3 Методика проведения текущего контроля успеваемости и промежуточной аттестации и критерии оценивания**

Поскольку дисциплина имеет практическую направленность и ориентирована, в том числе, на успешное применение полученных студентом навыков и знаний на практике при дальнейшей работе по специальности, подготовка ответов на экзамене проводится без ограничения доступа к материалам дисциплины, литературе и иным информационным источникам.

Устный экзамен проводится в соответствии с заранее опубликованным списком вопросов, покрывающих рассмотренные на лекционных занятиях темы. Студент должен в ходе экзамена ответить на два вопроса, на подготовку ответа на которые даётся не менее 40 минут, а также на несколько дополнительных вопросов, без подготовки. Количество и содержание дополнительных вопросов – на усмотрение преподавателя, принимающего экзамен.

Оценка «отлично» (A) ставится за полностью раскрытый теоретический материал и правильные ответы на дополнительные вопросы преподавателя.

Оценка «хорошо» (B) ставится за изложенный теоретический материал билета (возможно с помощью наводящих подсказок преподавателя) и правильные ответы на дополнительные вопросы преподавателя.

Оценка «хорошо» (C) ставится за полный ответ на вопросы билета с небольшими неточностями.

Оценка «удовлетворительно» (D) ставится за знание ответов на основные вопросы по каждой теме.

Оценка «удовлетворительно» (E) ставится за знание основных формулировок (по одной из каждого вопроса) с подсказками со стороны преподавателя.

Оценка «неудовлетворительно» (F) выставляется, если не выполняются условия для получения оценок «отлично», «хорошо» и «удовлетворительно».

**3.1.4 Методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации (контрольно-измерительные материалы, оценочные средства)**

Пример списка вопросов для устного экзамена:

1. Программирование и разработка коммерческих продуктов
2. Особенности разработки ПО от других инженерных областей
3. Компетенции профессионального программиста
4. Понятие жизненного цикла ПО
5. Водопадная модель жизненного цикла
6. Итеративная и спиральная модель жизненного цикла
7. Rational Unified Process
8. Agile подход к разработке
9. eXtreme Programming: общий подход, достоинства и недостатки
10. eXtreme Programming: практики “Короткий цикл разработки”
11. eXtreme Programming: практики “Непрерывность процесса”
12. eXtreme Programming: практики “Понимание, разделяемое всеми”
13. Scrum: назначение, особенности, общий подход
14. Scrum: роли в команде
15. Scrum: артефакты, используемые в планировании и разработке
16. Scrum: основные процессные мероприятия
17. Типы требований к программным системам
18. Основные характеристики требований
19. Основные действия, выполняемые при разработке требований
20. Основные действия, выполняемые при управлении требованиями
21. Основные документы, создаваемые при работе с требованиями
22. Функции менеджера проекта
23. Матрица ответственности и план коммуникаций
24. Основные действия по управлению рисками
25. Декомпозиция проекта
26. Построение графика работ: матрица зависимостей, сетевой график
27. Построение графика работ: сетевой график, диаграмма Гантта
28. Построение графика работ: оценка задач и графика в целом
29. Типичные ошибки при оценке проектов
30. Треугольник равновесия проекта
31. Основные практики сохранения равновесия проекта
32. Особенности добавления людей в проект
33. Практики по отслеживанию прогресса хода проекта
34. Типы проектных команд
35. Роли в проектной команде
36. Подходы к расчёту бюждета проекта
37. Особенности объектно-ориентированной парадигмы разработки ПО
38. Особенности процедурной парадигмы разработки ПО
39. Особенности функциональной парадигмы разработки ПО
40. Особенности рекурсивно-логической парадигмы разработки ПО
41. Понятие арихтектуры ПО
42. Функции и основные навыки архитектора ПО
43. Архитектура и жизненный цикл ПО
44. Основные подходы к декомпозиции задач
45. Базовые принципы проектирования ПО
46. Принципы объектно-ориентированного проектирования ПО
47. Критерии хорошего объектно-ориентированного дизайна
48. Системы контроля версий: назначение и типы.
49. Методы управления дефектами в программных проектах
50. Основные характеристики и жизненный цикл дефектов
51. Понятие качества ПО
52. Основные характеристики качества ПО
53. Основные метрики качества ПО
54. Методы обеспечения качества ПО

**3.1.5 Методические материалы для оценки обучающимися содержания и качества учебного процесса**

Оценка обучающимися содержания и качества учебного процесса по дисциплине осуществляется в установленном в СПбГУ порядке.

**3.2. Кадровое обеспечение**

**3.2.1 Образование и (или) квалификация штатных преподавателей и иных лиц, допущенных к проведению учебных занятий**

К преподаванию дисциплины могут быть допущены преподаватели, имеющие диплом о высшем образовании по соответствующему направлению и либо прошедшие обучение в аспирантуре не менее года, либо имеющие ученую степень.

**3.2.2 Обеспечение учебно-вспомогательным и (или) иным персоналом**

Специальных требований нет.

**3.3. Материально-техническое обеспечение**

**3.3.1 Характеристики аудиторий (помещений, мест) для проведения занятий**

В аудитории, где проводятся занятия, необходимо наличие досок и средств письма на них, а также технических средств, указанных в пункте 3.3.2.

**3.3.2 Характеристики аудиторного оборудования, в том числе неспециализированного компьютерного оборудования и программного обеспечения общего пользования**

В аудитории, где проводятся занятия, должен иметься проектор и экран.

**3.3.3 Характеристики специализированного оборудования**

Специальных требований нет.

**3.3.4 Характеристики специализированного программного обеспечения**

Специальных требований нет.

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Специальных требований нет.

**3.4. Информационное обеспечение**

**3.4.1 Список обязательной литературы**

1. Kerzner, Harold. Project management a systems approach to planning, scheduling, and controlling. Hoboken, N.J.: John Wiley & Sons, Inc., 2013. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/EBC1113482.

2. Charrel, Pierre-Jean, Galarreta, Daniel. Project Management and Risk Management in Complex Projects. Dordrecht: Springer Netherlands, 2007. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/978-1-4020-5837-0.

3. Mohapatra, Pratap K. J. Software engineering (a lifecycle approach). New Delhi: New Age International, 2010. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/EBC3017407

4. Tomayko, J. E. Human aspects of software engineering. Hingham, MA: Charles River Media, 2004. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/EBC3135679

5. Mistrík, Ivan, Grundy, John, Hoek, André, Whitehead, Jim. Collaborative Software Engineering. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/978-3-642-10294-3

6. Hazzan, Orit, Dubinsky, Yael. Agile Software Engineering. London: Springer London, 2008. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/978-1-84800-199-2.

7. Davis, Barbara. Mastering software project requirements: a framework for successful planning, development & alignment. Plantation, Florida: J. Ross Publishing, 2013. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/EBC3319548.

8.Chemuturi, Murali. Mastering software quality assurance best practices, tools and techniques for software developers. Fort Lauderdale, Fla: J. Ross Pub., 2011. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/EBC3319498.

**3.4.2 Список дополнительной литературы**

1. Lewis, James P. Fundamentals of project management. New York: American Management Association, 2007. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/EBC3001873.

2. Qin, Zheng. Zheng, Xiang,.Xing, Jiankuan. Software Architecture, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, 337pp. - ЭР по подписке СПбГУ: <https://find.library.spbu.ru/vufind/Record/978-3-540-74343-9>.

3. Aurum, Aybüke, Wohlin, Claes. Engineering and Managing Software Requirements. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. - ЭР по подписке СПбГУ: https://find.library.spbu.ru/vufind/Record/978-3-540-28244-0.

4. Evans, Isabel. Achieving software quality through teamwork. Boston: Artech House, 2004. - ЭР по подписке СПбГУ: <https://find.library.spbu.ru/vufind/Record/>.

**3.4.3 Перечень иных информационных источников**

Не предусмотрено.

**Раздел 4. Разработчики программы**

Брыксин Тимофей Александрович, к.ф.-м.н., доцент, t.bryksin@spbu.ru.