

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
аэрокосмических технологий
С.С. Негодяев**

	Рабочая программа дисциплины (модуля)
по дисциплине:	Практика работы над большими проектами по разработке ПО
по направлению:	Информатика и вычислительная техника
профиль подготовки:	Программная инженерия передовая инженерная школа радиолокации, радионавигации и программной инженерии кафедра технологий проектирования сложных технических систем
курс:	4
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 75 час.

Всего часов: 135, всего зач. ед.: 3

Программу составил: Ю.М. Аношин

Программа обсуждена на заседании кафедры технологий проектирования сложных технических систем
06.02.2020

Аннотация

Дисциплина «Практика работы над большими проектами по разработке ПО» призвана познакомить обучающихся с современными практиками коллективной работы, являющимися стандартами дефакто в области разработки программного кода и дать им опыт самостоятельной работы с использованием этих подходов в разработке ПО. Курс включает в себя освоение систем контроля версии, систем коллективной разработки (таск-трекеров), юнит-тесты, рефакторинг кода, углубление знаний принципов объектно-ориентированного программирования и написание дипломного проекта с использованием этих знаний и подходов.

1. Цели и задачи

Цель дисциплины

Дать представление и навыки работы над большими проектами по разработке ПО

Задачи дисциплины

1. Познакомить со структурой проекта, системами сборки, внешними зависимостями.
2. Дать навыки работы с системой контроля версий, трекером задач и понимание организации работы в команде.
3. Обучить проведению ревью кода.
4. Дать основы и опыт в тестировании кода.
5. Научить оборачивать код в удобные абстракции.
6. Познакомить с отладкой и профилированием кода.
7. Дать опыт практического использования концепций ООП и понимание того, как они помогают в работе с большими проектами
8. Дать опыт работы со ""старым"" кодом".

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-3 Способен составлять и оформлять научные и (или) технические (технологические, инновационные) отчеты (публикации, проекты)	ОПК-3.1 Знает основные правила оформления научных публикаций и научно-технической документации, в том числе с использованием прикладного программного обеспечения
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- ☐ принципы программирования (DIE, KISS, SOLID);
- ☐ системы сборки;
- ☐ основы тестирования ПО.

уметь:

- ☐ производить отладку и профилирование ПО;
- ☐ писать юнит тесты;
- ☐ разбираться в старом коде.

владеть:

- ☐ системой контроля версий git;
- ☐ трекером задач jira.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Системы сборки и внешние зависимости	2	2		2
2	Системы контроля версий	2	2		2
3	Трекер задач	2	2		2
4	Тестирование ПО и профилирование	8	4		4
5	Принципы ООП	16	8		8
6	Написание проекта		12		57
Итого часов		30	30		75
Подготовка к экзамену		0 час.			
Общая трудоёмкость		135 час., 3 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 7 (Осенний)

1. Системы сборки и внешние зависимости

Структура проекта. Зависимости внутри проекта и внешние зависимости.

Зачем нужны системы сборки.

Системы сборки: make, MSBuild, Cmake. Утилита qmake.

Написание и подключение статических и динамически библиотек.

2. Системы контроля версий

Зачем нужны системы контроля версий. Принципы работы систем контроля версий на примере git и svn. Основные операции в git. Создание веток и способы работы с ними в git.

3. Трекер задач

Зачем нужен трекер задач. Трекер задач jira и работа с ним.

4. Тестирование ПО и профилирование

Зачем нужно тестирование. Виды тестирования: юнит, performance, интеграционное и функциональное. Основы написания юнит тестов. Профилирование ПО.

5. Принципы ООП

Принципы программирования: DIE, KISS, SOLID. Знакомство с паттернами проектирования.

6. Написание проекта

Постановка цели. Формирование требований. Декомпозиция задачи. Составление графика, управление ресурсами.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Необходимое оборудование для лекций: компьютер (ноутбук) и мультимедийное оборудование (проектор, звуковая система).

6. Перечень рекомендуемой литературы

Основная литература

1. Программное обеспечение и его разработка [Текст] : [учеб. пособие для вузов] / Дж. Фокс ; пер. с англ. Л. Е. Карпова под ред. Д. Б. Подшивалова .— М. : Мир, 1985 .— 368 с.

Дополнительная литература

1. Алгоритмы и программы. Решение олимпиадных задач [Текст] : [учеб. пособие для студентов младших курсов] / И. Н. Порублѣв, А. Б. Ставровский .— М. : Вильямс, 2007 .— 480 с.

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Не используются

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

Для контроля и коррекции знаний, обучающиеся могут использовать компьютерное тестирование.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Студент, изучающий курс «Практика работы над большими проектами по разработке ПО», должен с одной стороны, овладеть общим понятийным аппаратом, а с другой стороны, должен научиться применять теоретические знания в эксперименте.

В результате изучения дисциплины студент должен иметь представление об измерительных системах, применяемых в аэродинамических трубах, а также должен знать методику проведения экспериментальных исследований в аэродинамических трубах.

Успешное освоение курса требует напряжённой самостоятельной работы студента. В программе курса приведено минимально необходимое время для работы студента над темой.

Самостоятельная работа включает в себя:

- чтение и конспектирование рекомендованной литературы;
- проработку учебного материала (по учебной и научной литературе), подготовку ответов на вопросы, предназначенных для самостоятельного изучения;
- подготовку к дифференцированному зачету.

Руководство и контроль за самостоятельной работой студента осуществляется в форме индивидуальных консультаций.

ПРИЛОЖЕНИЕ

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению: Информатика и вычислительная техника
профиль подготовки: Программная инженерия
передовая инженерная школа радиолокации, радионавигации и программной инженерии
кафедра технологий проектирования сложных технических систем
курс: 4
квалификация: бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Дифференцированный зачет

Разработчик: Ю.М. Аношин

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-3 Способен составлять и оформлять научные и (или) технические (технологические, инновационные) отчеты (публикации, проекты)	ОПК-3.1 Знает основные правила оформления научных публикаций и научно-технической документации, в том числе с использованием прикладного программного обеспечения
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива

2. Показатели оценивания компетенций

В результате изучения дисциплины «Практика работы над большими проектами по разработке ПО» обучающийся должен:

знать:

- ☐ принципы программирования (DIE, KISS, SOLID);
- ☐ системы сборки;
- ☐ основы тестирования ПО.

уметь:

- ☐ производить отладку и профилирование ПО;
- ☐ писать юнит тесты;
- ☐ разбираться в старом коде.

владеть:

- ☐ системой контроля версий git;
- ☐ трекером задач jira.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

- 1 Задание на семестр. Написать проект, пройдя ревью и тестирование
- 2 Найти ошибку в коде
- 3 Покрыть функцию юнит-тестами
- 4 Выполнить рефакторинг кода
- 5 Найти узкое место в коде при помощи профайлера
- 6 Написанного файла добавить под систему контроля версий, создать pull request, закрыть его и влить ветку в master
- 7 Разрешить ошибки сборки проекта
- 8 Завести задачу в трекере задач и поменять ей статус

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Рассказать про методологии разработки (scrum, waterfall)
2. Рассказать про принципы программирования
3. Рассказать про системы контроля версий и их различия
4. Рассказать про системы сборки
5. Рассказать про профилировщик
6. Рассказать про методы тестирования программного кода
7. Рассказать про паттерны проектирования

Критерии оценивания

Оценка «отлично (10)» выставляется обучающемуся, если показавшему всесторонние, систематизированные, глубокие знания предмета и в ходе беседы он верно и детально ответил на четыре (4) произвольных вопроса из выше приведенного перечня. Детальный ответ предполагает верные ответы на все уточняющие вопросы. Подготовка и защита инициативной курсовой работы является преимуществом.

Оценка «отлично (9)» выставляется обучающемуся, если в ходе беседы он верно, но не исчерпывающее детально ответил на четыре (4) произвольных вопроса из выше приведенного перечня (мог не ответить на некоторые уточняющие вопросы). Подготовка и защита инициативной курсовой работы является преимуществом.

Оценка «отлично (8)» выставляется обучающемуся, если в ходе беседы он верно, но не исчерпывающее детально ответил на четыре (4) произвольных вопроса из выше приведенного перечня (не ответил на уточняющие вопросы).

Оценка «хорошо (7)» выставляется обучающемуся, если в ходе беседы он верно и достаточно детально ответил на три (3) произвольных вопроса из выше приведенного перечня. Детальный ответ предполагает верные ответы на все уточняющие вопросы. Подготовка и защита инициативной курсовой работы является преимуществом.

Оценка «хорошо (6)» выставляется обучающемуся, если в ходе беседы он верно, но не исчерпывающее детально ответил на три (3) произвольных вопроса из выше приведенного перечня (не ответил на некоторые уточняющие вопросы). Подготовка и защита инициативной курсовой работы является преимуществом.

Оценка «хорошо (5)» выставляется обучающемуся, если в ходе беседы он верно и достаточно детально ответил на два (2) произвольных вопроса из выше приведенного перечня. Детальный ответ предполагает верные ответы на все уточняющие вопросы.

Оценка «удовлетворительно (4)» выставляется обучающемуся, если в ходе беседы он верно и достаточно детально ответил на один (1) произвольный вопрос из выше приведенного перечня. Детальный ответ предполагает верные ответы на все уточняющие вопросы. Подготовка и защита инициативной курсовой работы является преимуществом.

Оценка «удовлетворительно (3)» выставляется обучающемуся, если в ходе беседы он верно, но не исчерпывающее детально ответил на один (1) произвольный вопрос из выше приведенного перечня (не ответил на уточняющие вопросы).

Оценка «неудовлетворительно (2)» выставляется обучающемуся, если в ходе беседы он не смог ответить ни на один произвольный вопрос из выше приведенного перечня, но смог ответить на наводящие вопросы и вопросы с «подсказками».

Оценка «неудовлетворительно (1)» выставляется обучающемуся, если в ходе беседы он не смог ответить ни на один произвольный вопрос из выше приведенного перечня, а так же ни на один наводящий вопрос.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Во время проведения дифференциального зачета обучающиеся могут пользоваться программой дисциплины, а также собственными конспектами занятий по предмету.

Дифференциальный ачет проводится по итогам текущей активности в ходе занятий, защиты инициативной курсовой работы, и путем организации специального опроса, проводимого в простой устной форме, в виде беседы преподавателя и студента.