ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ

КАФЕДРА КИБЕРНЕТИКИ

ОДОБРЕНО  
  
протокол № 18 / 03   
  
от « 31 » мая 2018 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ КИБЕРНЕТИЧЕСКИХ СИСТЕМ (УПРАВЛЕНИЕ ПРОГРАММНЫМИ ПРОЕКТАМИ)

|  |  |
| --- | --- |
| Направление подготовки (специальность) | 09.03.04 Программная инженерия |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Семестр** | **Интерактив** | **Трудоемкость, кред.** | **Общий объем курса, час.** | **Лекции, час.** | **Практич. занятия, час.** | **Лаборат. работы, час.** | **СРС, час.** | **КСР, час.** | **Форма(ы) контроля, экз./зач./КР/КП** |
| 7 |  | 3 | 108 | 16 | 32 | 16 | 44 | 0 | З |
| 8 |  | 4 | 144 | 20 | 20 | 20 | 57 | 0 | Э |
| ИТОГО | 0 | 7 | 252 | 36 | 52 | 36 | 101 | 0 |  |

Группа: Б18-504, Б18-514

АННОТАЦИЯ

Дисциплина призвана обеспечить освоение студентами базовых теоретических знаний и практических приемов, необходимых для организации процесса коллективной разработки сложных программных систем. Основное внимание в курсе уделяется аспектам разработки, связанным с разработкой сертифицируемого программного обеспечения.

1. ЦЕЛИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целями освоения учебной дисциплины «Технология программирования кибернетических систем» являются:

• усвоение основных процессов Жизненного Цикла сертифицируемых программных разработок;

• получение практических навыков разработки и заполнения необходимых технологических документов.

2. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина «Технология программирования кибернетических систем» относится к факультативной дисциплиной для студента.

Для успешного обучения требуются знания в объеме первых трех курсов факультета кибернетики НИЯ МИФИ, в том числе:

• Информатика;

• Языки программирования и методы программирования; основы архитектуры параллельных вычислительных систем;

• Операционные системы;

3. КОМПЕТЕНЦИИ СТУДЕНТА, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ/ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ОБРАЗОВАНИЯ И КОМПЕТЕНЦИИ СТУДЕНТА ПО ЗАВЕРШЕНИИ ОСВОЕНИЯ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

ПК-1.1 – способность к проверке работоспособности и рефакторингу кода программного обеспечения

ПК-10 – владением методами контроля проекта и готовностью осуществлять контроль версий

ПК-11 – способность к формализации в своей предметной области с учетом ограничений используемых методов исследования

ПК-12 – готовность к использованию методов и инструментальных средств исследования объектов профессиональной деятельности

ПК-13 – готовность обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности

ПК-14 – способность готовить презентации, оформлять научно-технические отчеты по результатам выполненной работы, публиковать результаты исследований в виде статей и докладов на научно-технических конференциях

ПК-15 – владением навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения

ПК-16 – способность оценивать временную и емкостную сложность программного обеспечения

ПК-17 – владением навыками чтения, понимания и выделения главной идеи прочитанного исходного кода, документации

ПК-18 – способность создавать программные интерфейсы

ПК-2 – владением навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных

ПК-3 – владением навыками использования различных технологий разработки программного обеспечения

ПК-4 – владением концепциями и атрибутами качества программного обеспечения надежности, безопасности, удобства использования, в том числе роли людей, процессов, методов, инструментов и технологий обеспечения качества

ПК-5 – владением стандартами и моделями жизненного цикла

ПК-6 – владение современными методами проектирования, применения и обеспечения информационной безопасности баз данных

ПК-7 – владением классическими концепциями и моделями менеджмента в управлении проектами

ПК-8 – владением методами управления процессами разработки требований, оценки рисков, приобретения, проектирования, конструирования, тестирования, эволюции и сопровождения

ПК-9 – владением основами групповой динамики, психологии и профессионального поведения, специфичных для программной инженерии

УК-6 – Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни

4. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№ п.п** | **Наименование раздела учебной дисциплины** | **Недели** | **Лекции, час.** | **Практ. занятия / семинары, час.** | **Лабораторные работы, час.** | **Обязат. текущий контроль (форма\*, неделя)** | **Аттестация раздела (форма\*, неделя)** | **Максимальный балл за раздел\*\*** |
|  | *7 Семестр* |  |  |  |  |  |  |  |
| 1 | Раздел1 | 1-8 | 8 | 16 | 8 | КР-8 | КИ-8 | 25 |
| 2 | Раздел2 | 9-16 | 8 | 16 | 8 |  | КИ-16 | 25 |
|  | *Итого за 7 Семестр* |  | 16 | 32 | 16 |  |  | 50 |
|  | **Контрольные мероприятия за 7 Семестр** |  |  |  |  |  | З | 50 |
|  | *8 Семестр* |  |  |  |  |  |  |  |
| 1 | Раздел1 | 1-5 | 10 | 10 | 10 |  | КИ-8 | 25 |
| 2 | Раздел2 | 6-10 | 10 | 10 | 10 |  | КИ-10 | 25 |
|  | *Итого за 8 Семестр* |  | 20 | 20 | 20 |  |  | 50 |
|  | **Контрольные мероприятия за 8 Семестр** |  |  |  |  |  | Э | 50 |

\* – сокращенное наименование формы контроля

\*\* – сумма максимальных баллов должна быть равна 100 за семестр, включая зачет и (или) экзамен

Сокращение наименований форм текущего контроля и аттестации разделов:

|  |  |
| --- | --- |
| **Обозначение** | **Полное наименование** |
| КИ | Контроль по итогам |
| З | Зачет |
| Э | Экзамен |

КАЛЕНДАРНЫЙ ПЛАН

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Недели** | **Темы занятий / Содержание** | **Лек., час.** | **Пр./сем., час.** | **Лаб., час.** |
|  | *7 Семестр* | 16 | 32 | 16 |
| **1-8** | **Раздел1** | 8 | 16 | 8 |
| 1 | **Введение** История развития технологий и инструментов программирования. Проблемы качества программного обеспечения (ПО). Основные характеристики программных систем. Основные характеристики процесса разработки ПО. Проблемы разработки ПО. Взаимосвязь характеристик ПО и процесса его разработки. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 1 | 2 | 1 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 2 - 3 | **Технология программирования как инженерная дисциплина** Основные понятия технологии программирования. Свод знаний о программной инженерии (SWEBOK). Цели и история проекта SWEBOK. Области знаний SWEBOK. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 2 | 4 | 2 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 4 - 5 | **Жизненный цикл коллективной разработки ПО** Определение понятия жизненного цикла. Процессы жизненного цикла. Проблемы коллективной разработки ПО. Принципы коллективной разработки ПО и модели команд разработчиков. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 2 | 4 | 2 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 6 | **Процессы жизненного цикла ПО** Процессная модель разработки ПО. Основные процессы разработки ПО: планирование, определение требований, проектирование, кодирование, интеграция, верификация, управление конфигурацией. Процессы и проблемы сертификации ПО. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 1 | 2 | 1 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 7 - 8 | **Основы управления проектами** Концепции проекта и управления проектом. Проект как система. Цели, требования, среда, участники проекта. Жизненный цикл проекта. Планирование, оценка, управление рисками, обеспечение качества проекта | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 2 | 4 | 2 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| **9-16** | **Раздел2** | 8 | 16 | 8 |
| 9 - 11 | **Модели жизненного цикла ПО** Каскадные, эволюционные и итерационные версии моделей жизненного цикла. Agile технологии. Понятие промышленного программирования. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 3 | 6 | 3 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 12 - 13 | **Стандарты разработки ПО** Стандарты IEEE, ISO, ГОСТ. Использование стандартов в процессах разработки ПО. Стандарты предприятия и проекта. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 2 | 4 | 2 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 14 - 16 | **Требования к ПО** Виды требований в процессе разработки ПО. Бизнес-требования, пользовательские требования, системные требования, требования высокого уровня, требования низкого уровня. Трассировка требований различных уровнях. Процесс разработки требований. Проблемы разработки требований. Управление требованиями. Спецификации требований. Стандарты спецификации требований. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 3 | 6 | 3 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
|  | *8 Семестр* | 20 | 20 | 20 |
| **1-5** | **Раздел1** | 10 | 10 | 10 |
| 1 | **Процесс проектирования ПО** Цели и мероприятия процесса проектирования ПО. Методологии проектирования. Критерии проектирования. Стандарты проектирования ПО. Описание проекта ПО. Архитектура ПО. Требования низкого уровня. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 2 | 2 | 2 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 2 - 3 | **Стандарты разработки сертифицируемого ПО (DO-178B/C, КТ-178B)** Цели, содержание и структура документов DO-178B/C. Процессы жизненного цикла, определенные в DO-178B/C: процесс планирования ПО, процессы разработки ПО, интегральные процессы, процесс сертификации. Цели и мероприятия процессов жизненного цикла. Уровни ПО. Данные жизненного цикла ПО. Потоки данных жизненного цикла. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 4 | 4 | 4 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 4 - 5 | **Управление конфигурациями** Цели и мероприятия процесса управления конфигурациями. Идентификация конфигурации. Трассируемость элементов конфигурации. Категории контроля данных. Контроль версий. Базовые версии. Индексы конфигурации. Управление изменениями, сообщения о проблемах, запросы на изменение. Системы управления конфигурациями. Информационная модель процесса разработки ПО. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 4 | 4 | 4 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| **6-10** | **Раздел2** | 10 | 10 | 10 |
| 6 | **Процесс верификации ПО** Цели и мероприятия процесса верификации ПО. План верификации. Рассмотрения, анализы и испытания как виды деятельности в процессе верификации. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 2 | 2 | 2 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 7 - 8 | **Рассмотрения и анализы ПО** Рассмотрения и анализы в процессе верификации. Различие между рассмотрениями и анализами. Рассмотрения и анализы требований высокого уровня. Рассмотрения и анализы требований низкого уровня. Рассмотрения и анализы архитектуры ПО. Рассмотрения и анализы исходного кода ПО. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 4 | 4 | 4 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |
| 9 - 10 | **Тестирование ПО** Тестирование в процессе верификации. Способы тестирования. Тестирование черного ящика, белого ящика, серого ящика. Тестовые планы и тестовые процедуры. Среда тестирования. Тесты на основе требований. Робастные тесты. Интеграционные и низкоуровневые тесты. Анализ структурного покрытия ПО. Критерии структурного покрытия. Рассмотрения и анализы тестовых примеров, тестовых процедур и результатов тестов. | Всего аудиторных часов | Всего аудиторных часов | Всего аудиторных часов |
| 4 | 4 | 4 |
| Онлайн | Онлайн | Онлайн |
|  |  |  |

Сокращенные наименования онлайн опций:

|  |  |
| --- | --- |
| **Обозначение** | **Полное наименование** |
| ЭК | Электронный курс |
| ПМ | Полнотекстовый материал |
| ПЛ | Полнотекстовые лекции |
| ВМ | Видео-материалы |
| АМ | Аудио-материалы |
| Прз | Презентации |
| Т | Тесты |
| ЭСМ | Электронные справочные материалы |
| ИС | Интерактивный сайт |

ТЕМЫ ЛАБОРАТОРНЫХ РАБОТ

|  |  |
| --- | --- |
| **Недели** | **Темы занятий / Содержание** |
|  | *7 Семестр* |
| 1 - 8 | **Раздел 1** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |
| 9 - 16 | **Раздел 2** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |
|  | *8 Семестр* |
| 1 - 8 | **Раздел1** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |
| 9 - 16 | **Раздел2** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |

ТЕМЫ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

|  |  |
| --- | --- |
| **Недели** | **Темы занятий / Содержание** |
|  | *7 Семестр* |
| 1 - 8 | **Раздел1** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |
| 9 - 16 | **Раздел2** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |
|  | *8 Семестр* |
| 1 - 8 | **Раздел1** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |
| 9 - 16 | **Раздел2** На практических и лабораторных занятиях проводится коллективная разработка учебного проекта по выбору студентов (преподавателем предлагается на выбор несколько проектов). Реализуется модель промышленной разработки ПО. Из студентов создается коллектив разработчиков, между ними распределяются роли. Коллективу ставится общая задача, которая под контролем преподавателей детализируется на более мелкие. Проводится контроль и оценка выполнения частных и общей задач отдельными студентами и коллективом в целом. Занятия проводятся каждую неделю. |

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

1. Лекционные занятия:

a. комплект электронных презентаций/слайдов,

b. аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук)

2. Практические занятия:

a. компьютерный класс,

b. презентационная техника (проектор, экран, компьютер/ноутбук),

c. стандартный пакет программ Microsoft Office.

6. ТРЕБОВАНИЯ К ФОНДУ ОЦЕНОЧНЫХ СРЕДСТВ В РАМКАХ РЕАЛИЗУЕМОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Оценка формируется с учетом посещаемости студентами лекций, семинаров, их практической работы, включая выполнение лабораторных и домашних заданий, а также выполнение контрольных работ. Контрольные работы, каждая длительностью по 2 академических часа, включают в себя по 5 вопросов по материалам прочитанных лекций. Несданные контрольные работы пересдаются на зачете. Работы оцениваются по 4-балльной шкале (от 2 до 5).

При оценке используется 100-балльная шкала. Итоговый балл (КИ) по каждому разделу рассчитывается следующим образом:

посещаемость лекций не менее 80% +10 баллов

не менее 60% +5 баллов

не менее 40% 0 баллов

менее 40% - 5 баллов

посещаемость семинарских занятий не менее 80% +10 баллов

не менее 60% +5 баллов

не менее 40% 0 баллов

менее 40% - 5 баллов

контрольные работы: Оценка 5 +10 баллов

Оценка 4 +5 баллов

Оценка 3 0 баллов

Оценка 2 -5 баллов

Максимальная оценка за зачет и экзамен составляет 50 баллов.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОСНОВНАЯ ЛИТЕРАТУРА:

1. ЭИ S34 Agile Software Development Teams : , Cham: Springer International Publishing, 2016

2. ЭИ S70 Software Quality. The Future of Systems- and Software Development : 8th International Conference, SWQD 2016, Vienna, Austria, January 18-21, 2016, Proceedings, Cham: Springer International Publishing, 2016

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА:

1. 004 Б42 Экстремальное программирование. Разработка через тестирование : , Бек К., М.и др.: Питер, 2003

2. 681.3 Б89 Как проектируются и создаются программные комплексы : Мифический человеко-месяц. Очерки по системному программированию, Брукс Ф.П., М.: Наука, 1979

3. ЭИ С79 Методы объектно-ориентированного описания систем и моделирования на языке UML : учеб. пособие, Е. Б. Степанова, А. В. Тимофеев, Москва: МИФИ, 2006

4. 004 Ф28 UML. Основы : краткое рук-во по унифицированному языку моделирования, М. Фаулер, К. Скотт, СПб: Символ, 2002

5. 004 О-66 Технологии разработки программного обеспечения : Разработка сложных программных систем:Учеб.пособие для вузов, Орлов С.А., М.и др.: Питер, 2003

6. 004 С38 Верификация программного обеспечения : учебное пособие, С. В. Синицын, Н. Ю. Налютин, Москва: Интернет-Университет информационных технологий; Бином. Лаборатория знаний, 2008

7. 004 С38 Операционные системы : учеб. пособие, С. В. Синицын, Н. Ю. Налютин, Москва: МИФИ, 2006

8. 681.3 С38 Языки управления заданиями в операционных системах : Текст лекций, С.В. Синицын, М.: МИФИ, 1989

9. 681.5 Т47 Введение в проектирование систем управления : , Тищенко Н.М., М.: Энергоатомиздат, 1986

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ:

Специальное программное обеспечение не требуется

LMS И ИНТЕРНЕТ-РЕСУРСЫ:

-

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Специальное материально-техническое обеспечение не требуется

9. ##Definition not found: 'static\_section\_edu\_stud'##

Семинарские и лабораторные работы по курсу "Технология программирования" проходят в режиме имитации реальной программной разработки. Ведущий занятия преподаватель играет роль Заказчика.

Группа студентов образуют команду разработчиков. Роли в группе студенты выбирают и назначают исполнителей самостоятельно. Вмешательство в этот процесс преподавателей должно быть минимизировано.

Тема общего проекта выбирается студентами самостоятельно. При этом преподаватель должен помогать группе в этом путём участия в анализе предлагаемых тем и предсказания областей риска в каждой из них.

У выбираемых тем есть единственное ограничение: тема не должна "умещаться в одну голову", т.е. она должна быть настолько большой, чтобы все её спецификации не могли быть запомнены даже самым талантливым членом группы.

Возможным вариантом следует признать вариант, при котором группа продолжает проект, начатый другой группой ранее и не завершённый в связи с окончанием курса.

Следует приветствовать выбор студентами в качестве темы своего проекта одну из компонент среды сертифицируемых программных разработок (конфигурационное управление, тестирование, включая валидацию и верификацию,...).

Ещё более должно быть интересным, если один из "работающих" студентов приносит тему из места своей работы. В последнем случае этот студент автоматически становится "проводником" между заинтересованными лицами (своими работодателями) и исполнителями. При этом ведущий занятия преподаватель должен иметь право на включение собственных требований по совершенствованию разрабатываемого продукта в его спецификации.

Предложения об оплате деятельности группы следует сдержанно (без энтузиазма) приветствовать, но при этом жёстко предупредить заказчиков, что назначение этих занятий – не зарабатывание денег, а обучение студентов практике коллективной работы. Также следует поддерживать студентов в их намерении завершить свою разработку после завершения данного курса, превратив её результат в коробочный продукт. До настоящего времени было две попытки (предложения оплачивать работу группы), но обе они завершились "ничем" сразу после такого предупреждения. Была одна попытка доведения результатов проекта до уровня коробочного продукта, однако она ничем не завершилась.

Выбор студентами собственной технологии работы и используемого инструментария следует всячески поощрять и поддерживать, добиваясь реального опробования нескольких вариантов. На эту деятельность ведущему преподавателю не следует жалеть времени (ни своего, ни студенческого).

В самом начале курса следует жёстко потребовать быстрого создания сайта проекта, на котором хранить все материалы проекта, включая протоколы общих собраний и принятых решений. Следует приветствовать решения, связанные с использованием SVN, Subversion или иных подобных компонент для управления версиями создаваемых материалов. Этот сайт следует использовать в качестве общего хранилища данных проекта и быстрого обмена сведениями между участниками разработки.

В ходе разработки ведущий преподаватель должен, с одной стороны, всячески мотивировать студентов пробовать разные роли (меняться ими) в ходе проекта и вносить полезные (и не полезные) предложения по изменению применяемой технологии, включая используемые инструменты.

Поскольку главная цель курса – научить студентов работать командой в современных российских условиях – ведущий преподаватель должен время от времени обоснованно менять требования к разрабатываемому продукту, заставляя студентов переделывать уже принятые решения.

Рабочая программа дисциплины составлена в соответствии с ОС НИЯУ МИФИ (ФГОС) и учебным планом основной образовательной программы (программ).

Автор(ы):

|  |  |  |
| --- | --- | --- |
|  | Козырев Владимир Петрович, к.т.н. |  |