АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

«Сервис-ориентированные технологии разработки программных систем»

по основной профессиональной образовательной программе по направлению подготовки 09.04.04 «Программная инженерия» (уровень магистратуры)

Направленность (профиль): Разработка программно-информационных систем
Трудоемкость дисциплины – 3 з.е. (108 часов)
Форма промежуточной аттестации – Зачет.
В результате освоения дисциплины обучающийся должен обладать следующими компетенциями:

- ОПК-2 Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач;
- ОПК-5 Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем.

Содержание дисциплины:
Дисциплина «**Сервис-ориентированные технологии разработки программных систем**» включает в себя следующие разделы:
Форма обучения очная. Семестр 1.

1. **Введение.** Достоинства и недостатки .net. Цель разработки .net. Основные достоинства и недостатки .net. Приложение на основе .net
2. **Высокоуровневая архитектура бизнес -ПО.** Типовая высоко-уровневая архитектура бизнес-ПО. Некоторые вариации. Детализированная архитектура приложения, изучаемого в курсе.
3. **Отложенное сохранение данных.** Рассмотрение подхода при разработке распределенных приложений с условным названием "Отложенное сохранение данных". Его достоинства и недостатки. Оптимистичная и пессимистичная блокировка. Особенности реализации "Отложенного сохранения данных.
4. **Разработка модуля DataAccessor. DataAccessor**. Различные варианты DataAccessor-ов. Практические советы по реализации DataAccessor-ов. Вариант DataAccessor для выполнения лабораторных работ.
5. **Юнит-тестирование.** Определение юнит-тестирования. Его достоинства и недостатки. Использование юнит-тестирования при разработке ПО. Виды юнит-тестов, которые должны обязательно присутствовать. Возможности юнит-тестирования на базе фреймворка nunit.
6. **Бизнес-логика**. Бизнес-логика, её роль в программе. Практические советы по реализации бизнес-логики.
7. **Dependency Injection.** Dependency Injection. Достоинства и недостатки такого способа сбора различных модулей системы. Практическое использование Dependency Injection на примере Spring framework.
8. **Разработка презентационного слоя на стороне сервера**. Веб-сервис как презентационный слой. Разработка веб-сервиса на .net. Использование Spring framework для реализация веб-сервиса путем оборачивания компонента системы с помощью Spring.
9. **Data Binding**. Основные принципы. Собственные источники данных. Технология DataBinding. Принципы работы технологии. Практические советы по использованию технологии. Создание собственных источников данных для технологии DataBinding.
10. **Презентационный слой на стороне клиентского приложения**. Разработка клиентской части приложения. Особенности разработки клиентской части при реализации схемы "отложенное сохранение". Обработка конфликтов при сохранении. Работа с первичными ключами на стороне клиента.

Разработал:
Старший преподаватель
кафедры ПМ — А.А. Шальнев
Проверил:
Декан ФИТ — А.С. Авдеев

ФГБОУ ВО «Алтайский государственный технический университет им. И.И. Ползунова» (АлтГТУ)
ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ