



Тема 1: Введение в разработку ПО

Занятие 1.1. Введение в курс, цели и ожидания.

Цель занятия:

- Ознакомиться со структурой курса "Технология разработки программного обеспечения" основными целями, ожиданиями, методами обучения и оценки успехов.



Роль программного обеспечения в современном мире



Разработка программного обеспечения имеет огромное значение в современном мире из-за своего влияния на практически все сферы жизни и бизнеса. Вот несколько аспектов, подчеркивающих важность разработки программного обеспечения:

Автоматизация и оптимизация

Программное обеспечение позволяет автоматизировать множество рутинных и повторяющихся задач, что снижает человеческий труд и увеличивает эффективность процессов. Это особенно актуально в бизнесе, где оптимизация операций может привести к сокращению издержек и повышению конкурентоспособности.

Информационная обработка

Программное обеспечение позволяет автоматизировать множество рутинных и повторяющихся задач, что снижает человеческий труд и увеличивает эффективность процессов. Это особенно актуально в бизнесе, где оптимизация операций может привести к сокращению издержек и повышению конкурентоспособности.

Коммуникация и связь

Системы связи, социальные сети, мессенджеры - все они являются программным обеспечением, которое связывает людей на глобальном уровне. Программы для видеосвязи и онлайн-коллаборации также помогают преодолевать географические и временные барьеры

Инновации и новые возможности

Разработка программного обеспечения способствует созданию новых технологий и продуктов, которые могут изменить способ взаимодействия общества. Это включает в себя искусственный интеллект, интернет вещей, виртуальную реальность, блокчейн и многое другое.

Научные исследования и разработки

Множество научных исследований зависит от специализированного программного обеспечения для моделирования, анализа данных и симуляций. Это применяется в астрономии, биологии, химии, физике и других дисциплинах.

Безопасность и защита

Программное обеспечение играет важную роль в обеспечении информационной безопасности. Разработка криптографических методов, антивирусов, средств аутентификации и защиты данных, становится критически важной в мире цифровых угроз.

Медицина и здравоохранение

Программное обеспечение применяется в медицинских исследованиях, диагностике, телемедицине и управлении медицинскими данными, что способствует улучшению здоровья и жизни людей.

И многое другое

- офисные приложения;
- приложения для управления проектами;
- приложения для работы с локальной сетью;
- приложения для Интернета;
- программы для научных исследований и расчетов;
- образовательные программы;
- программы для организации работы учебных заведений;
- программы для библиотек;
- программы для работы с мультимедиа;
- бухгалтерские программы;
- финансовые программы;
- программы для проектирования;
- программное обеспечения для бизнеса;
- программное обеспечение органов государственной власти;
- программы поддержания безопасности;
- программы для личного планирования;
- игры.

Цели курса:

1. Освоение основных концепций разработки ПО.
2. Понимание процесса анализа и проектирования:.
3. Овладение практическими навыками программирования.
4. Развитие навыков работы в команде.
5. Применение современных инструментов и технологий.
6. Развитие аналитического мышления.
7. Формирование уверенности в разработке ПО.
8. Стимулирование креативности и инноваций.
9. Подготовка к реальным сценариям работы.
10. Понимание этических и профессиональных стандартов

Что будет на курсе?

- теоретические занятия;



Что будет на курсе?

- 10 лабораторных работ;



Что будет на курсе?

- финальный проект.



Требования к проекту:

Тематика проекта: Проект должен соответствовать тематике курса "Проектирование и технология разработки программного обеспечения". Тема может быть связана с разработкой веб-приложения, мобильного приложения, программного инструмента или другого программного решения.

Требования к проекту:

Функциональность: Проект должен включать минимальный набор функций, который бы демонстрировал навыки, полученные студентами в процессе обучения. Функциональные требования могут быть определены на основе лекций, лабораторных работ и обсуждений.

Требования к проекту:

Архитектура и ООП: Проект должен следовать принципам ООП (или функционального программирования) и хорошей архитектуре. Это может включать в себя правильное разделение на модули, использование паттернов проектирования и создание чистого, масштабируемого кода.

Требования к проекту:

Интерфейс пользователя: Если проект имеет пользовательский интерфейс (например, веб-приложение или мобильное приложение), интерфейс должен быть интуитивно понятным, функциональным и приятным для использования.

Требования к проекту:

База данных: Если проект включает работу с данными, должна быть предусмотрена адекватная структура базы данных и механизмы взаимодействия с ней.

Требования к проекту:

Тестирование: Проект должен быть подвергнут тестированию. Это включает юнит-тестирование, интеграционное тестирование и, возможно, функциональное тестирование для обеспечения качества и стабильности.

Требования к проекту:

Документация: Проект должен быть документирован. Включать описание функций, инструкции по установке и использованию, описание архитектуры и другую необходимую документацию.

Требования к проекту:

Код-ревью: Проект может быть подвергнут код-ревью, чтобы проверить качество кода и соответствие принципам.

Требования к проекту:

Презентация: Студенты должны подготовить презентацию проекта, в которой они должны продемонстрировать его функциональность, архитектуру и принятые решения.

Требования к проекту:

Сроки: разные этапы разработки проекта, включая создание, тестирование, документирование и презентацию имеют свои сроки.

Требования к проекту:

Индивидуальность: Каждый проект должен быть уникальным и выполненным индивидуально или коллективно.

В случае коллективной разработки, у каждого участника должна быть своя роль в проекте.

Требования к проекту:

Масштаб проекта: Требования к масштабу проекта не устанавливаются жестко.

Это может быть небольшое приложение с несколькими функциями или более сложное решение с широким функционалом.

Этапы разработки финального проекта:

Определение Требований:

Понимание целей и функциональности приложения.

Составление списка основных требований и функций приложения.

Определение пользовательских сценариев и потребностей.

Этапы разработки финального проекта:

Проектирование:

Разработка архитектуры приложения.

Проектирование базы данных (ER-диаграммы, схемы таблиц).

Разработка диаграммы классов или других необходимых диаграмм.

Этапы разработки финального проекта:

Разработка Прототипа:

Создание начальной версии интерфейса пользователя (UI) или прототипа приложения.

Тестирование прототипа с пользователем для получения обратной связи.

Этапы разработки финального проекта:

Разработка Кодовой Базы:

Написание исходного кода приложения с учетом проектирования.

Разработка бэкэнда и фронтэнда (если применимо).

Внедрение функциональности, включая обработку данных, взаимодействие с базой данных и другие операции.

Этапы разработки финального проекта:

Документирование:

Создание документации пользователя, включая инструкции по использованию.

Документирование структуры базы данных и архитектуры приложения.

Этапы разработки финального проекта:

Интеграция и Деплой:

Интеграция всех компонентов приложения.

Подготовка приложения к деплою на сервер (если это веб-приложение).

Этапы разработки финального проекта:

Тестирование Продакшн Версии:

Тестирование окончательной версии приложения в реальных условиях.

Проверка производительности и надежности приложения.

Этапы разработки финального проекта:

Финальная подготовка и представление:

Подготовка окончательной версии приложения.

Подготовка презентации проекта для представления перед аудиторией.

Примерный перечень тем проектов:

<https://github.com/ShViktor72/softwaredevelopment.git>

Обратная связь:

colledge20education23@gmail.com

Список литературы: