Министерство образования Российской Федерации

Тверской государственный технический университет

Кафедра Программного обеспечения

Программа для работы с личными делами студентов

Техническое задание на лабораторную работу

Выполнил: Бакеев А.В.

Проверил: Мальков А.А

Тверь, 2024

**Содержание**

Введение 3

Постановка задачи 3

Модель waterfall 3

Фазы модели 3

Техническое задание 3

Плюсы и минусы модели 5

V-образная модель 6

Фазы модели 6

Техническое задание 6

Плюсы и минусы модели 8

# Введение

У любого программного обеспечения есть жизненный цикл — этапы, через которые оно проходит с начала создания до конца разработки и внедрения. Чаще всего это подготовка, проектирование, создание и поддержка. Этапы могут называться по-разному и дробиться на более мелкие стадии.

Модель разработки программного обеспечения описывает, какие стадии жизненного цикла оно проходит и что происходит на каждой из них.

Цель использования модели заключается в оптимизации различных бизнес процессов, сокращении временных и финансовых затрат для компании. А также структурировании работы специалистов.

# Постановка задачи

**Вариант 2.** Разработать программный модуль «Личные дела студентов». Программный модуль предназначен для получения сведений о студентах сотрудниками деканата, профкома и отдела кадров. Сведения должны храниться в течение всего срока обучения студентов и использоваться при составлении справок и отчетов.

Входные данные: ФИО, год окончания школы, город рождения, средний бал за последний семестр, наличие работ

Выходные данные: Справки нескольких видов, отчеты об успеваемости

# Модель waterfall

Техническое задание:

1. **Введение**

Настоящее техническое задание распространяется на разработку программы личных дел студентов, предназначенной для использования деканатом, отделом кадров и профкомом при работе, в моменты, когда необходимо получить некоторые личные данные того или иного студента (для заполнения отчетов, справок и т.д.)

1. **Основание для разработки**
   1. Программа разрабатывается на основе учебного плана кафедры «Программного обеспечения»
   2. Программный модуль «Личные дела студентов»
   3. Исполнитель: компания Бакеев & co.
   4. Соисполнители: нет.

**3. Назначение**

Программа предназначена для работы сотрудников отдела кадров, деканата и профком, для оперативного получения доступа к личным данным студентов.

**4. Требования к программе или программному изделию**

Функциональные требования:

1. Возможность получить все имеющиеся в базе данные о конкретном студенте (Фамилия, имя, отчество, город рождения, средний балл за предыдущий семестр, рабочий статус (работает, не работает, самозанятый, имеет Юр. Лицо))
2. Получить выборочно данные (которые, по мнению заказчика, чаще всего нужны) чаще всего необходимые для заполнения справок и документов
3. Возможность заполнять базу данных и редактировать существующие данные сотрудникам отдела кадров, остальные пользователи должны иметь возможность только на чтение данных

Нефункциональные требования:

1. Безопасность системы. Человек, не являющийся нашим сотрудником не сможет вводить данные студентов, дабы не фальсифицировать их.
2. Понятный интерфейс для пользователя любого уровня (отсутствие кликабельных картинок, очень маленького шрифта или надписей/кнопок, сливающихся с фоном окна приложения)
3. Возможность масштабирования (дополнительные справки, дополнительные данные в системе)

Требования к составу и параметрам технических средств:

* Процессор: Intel Core 2 Duo
* Оперативная память: 2 GB ОЗУ
* Видеокарта: DirectX 11 compatible video card (integrated or dedicated with min 512MB memory)
* DirectX: версии 11
* Место на диске: 1 GB

**5. Требования к программной документации**

1. Программа должна иметь исчерпывающие комментарии во всех существующих и будущих блоках кода.
2. Пояснительная записка для разработчиков с описанием технических решений.
3. Документация для пользователя

**6. Сроки разработки продукта**

Разработка приложения должна начаться в период с 1.10.2024 по 1.11.2024 и продлится не больше 3 недель

**#убрать Плюсы и минусы waterfall модели:**

+ Четкость и структура: Этапы проекта четко определены (требования, проектирование, разработка, тестирование, внедрение). Это упрощает управление проектом.

+ Легкость планирования: В связи четкой структуры проще планировать и прогнозировать сроки выполнения.

+ Документирование: Каждый этап располагает к написанию документации, что обеспечивает хорошую запись требований и решений.

+ Управляемость: Изменения в проекте контролируются на каждом этапе, что упрощает отслеживание процесса

- Негибкость: Трудно вносить изменения в требования после начала разработки, это может быть проблематично при изменении условий после начала работы над проектом.

- Риски: Возможность выявления серьезных проблем или несоответствий только на поздних этапах разработки и тестирования, как следствие это может увеличить затраты на исправление, как временные, так и финансовые.

- Задержки: Если один этап задерживается, это может задержать весь проект, так как этапы следуют друг за другом «паровозиком».

- Проблемы с требованиями: Если требования изменяются по ходу проекта, это может привести к необходимости переделки уже завершенных этапов, что увеличивает затраты и время. А в худшем варианте данного сценария, элементы проекта, шедшие за переделанными становятся не работоспособными в принципе.

# V-образная модель SDLC

Техническое задание

* 1. **Введение**

Настоящее техническое задание распространяется на разработку программы личных дел студентов, предназначенной для использования деканатом, отделом кадров и профкомом при работе, в моменты, когда необходимо получить некоторые личные данные того или иного студента (для заполнения отчетов, справок и т.д.)

**2. Основание для разработки**

* 1. Программа разрабатывается на основе учебного плана кафедры «Программного обеспечения»
  2. Программный модуль «Личные дела студентов»
  3. Исполнитель: компания Бакеев & co.
  4. Соисполнители: нет.

**3. Назначение**

Программа предназначена для работы сотрудников отдела кадров, деканата и профком, для оперативного получения доступа к личным данным студентов.

**4. Требования к программе или программному изделию**

Функциональные требования

1. Возможность получить все имеющиеся в базе данные о конкретном студенте (Фамилия, имя, отчество, город рождения, средний балл за предыдущий семестр, рабочий статус (работает, не работает, самозанятый, имеет Юр. Лицо))
2. Получить выборочно данные (которые, по мнению заказчика, чаще всего нужны) чаще всего необходимые для заполнения справок и документов
3. Возможность заполнять базу данных и редактировать существующие данные сотрудникам отдела кадров, остальные пользователи должны иметь возможность только на чтение данных

Нефункциональные требования:

1. Безопасность системы. Человек, не являющийся нашим сотрудником не сможет вводить данные студентов, дабы не фальсифицировать их.
2. Понятный интерфейс для пользователя любого уровня (отсутствие кликабельных картинок, очень маленького шрифта или надписей/кнопок, сливающихся с фоном окна приложения)
3. Возможность масштабирования (дополнительные справки, дополнительные данные в системе)

Требование к составу и параметрам технических средств:

* Процессор: Intel Core 2 Duo
* Оперативная память: 2 GB ОЗУ
* Видеокарта: DirectX 11 compatible video card (integrated or dedicated with min 512MB memory)
* DirectX: версии 11
* Место на диске: 1 GB

**5. Требования к программной документации:**

1. Программа должна иметь исчерпывающие комментарии во всех существующих и будущих блоках кода.
2. Пояснительная записка для разработчиков с описанием технических решений.
3. Документация для пользователя

**6. Сроки разработки продукта**

Разработка приложения должна начаться в период с 1.10.2024 по 1.11.2024 и продлится не больше 3 недель

**#Убрать Плюсы и минусы V-образной модели:**

+ Четкость и структура: Как и в водопадной модели, этапы проекта четко определены заранее и следуют друг за другом, что обеспечивает четко структурированный подход к разработке

+ Улучшенное тестирование: Тестирование начинается на ранних этапах разработки, параллельно совершенствуется вместе с разработкой, а каждый новый этап начинается только после успешного завершения старого, что позволяет выявлять ошибки сразу и улучшать качество продукта

+ Более гибкий: В отличие от водопадной модели, V-образная модель позволяет вносить изменения в требования и исправлять ошибки на более ранних этапах процесса разработки из-за постоянного тестирования продукта

+ Надежнее результат: Благодаря улучшенному тестированию и проверке на каждом этапе, вероятность ошибок и проблем в конечном продукте уменьшается

- Сложность управления: Поскольку каждый этап имеет свою под-стадию тестирования, управление процессом разработки может быть сложным и требовать больше усилий, т.к. придется интегрировать отдел с тестировщиками

- Затраты на тестирование: Тестирование проводится на каждом этапе, это может увеличить стоимость проекта и время разработки, а увеличение времени приведет к оплате сверхурочных для сотрудников

- Не всегда подходит для малых проектов: Из-за своей структурированности и длительности процесса, модель может быть излишне сложной для небольших проектов.