1. **Анализ задачи**
   1. **Постановка задачи**

Организационно-экономическая сущность задачи: Наименование задачи - разработка программного обеспечения «BdayCalendar» с целью обеспечения эффективного управления информацией о днях рождения сотрудников предприятия ЧТУП «ЭтикетСервис». Продукт предназначен для сотрудников по работе с персоналом, используется ежедневно, источниками данных являются самые сотрудники предприятия. Проводится обзор аналогичных программных продуктов для выделения уникальных особенностей «BdayCalendar».

Функциональные требования: Программа должна обеспечивать отображение списка сотрудников с указанием дней и месяцев их рождения, фото сотрудника, возможность добавления, редактирования и удаления информации о днях рождения, оповещение пользователей о предстоящих днях рождения.

Описание исходной (входной) информации: Исходной информацией являются имена, даты рождения и фотографии сотрудников, вводимые в интерфейсе программы. Пользователями исходной информации выступают сотрудники подразделения по работе с персоналом.

Описание результатной (выходной) информации: Программа формирует списки сотрудников, уведомления о предстоящих событиях, предоставляет печатные сводки и уведомления в интерфейсе. Информация обновляется ежедневно.

Описание используемой условно-постоянной информации: Используется условно-постоянная информация в виде классификаторов (должности, подразделения) и таблиц с данными о сотрудниках, представленных в интерфейсе программы.

Нефункциональные (эксплуатационные) требования: Программа должна обеспечивать интуитивно понятный пользовательский интерфейс, сопровождаться подробной документацией и обучающими материалами.

Требования к производительности включают отзывчивость интерфейса и минимальное время реакции на запросы. Реализация должна соответствовать стандартам программирования и обеспечивать совместимость с операционными системами предприятия. Надежность программы проявляется в возможности восстановления после сбоев и минимизации воздействия сбоев на ее работу.

Требования к интерфейсу включают четкое взаимодействие с пользователем и поддержку мобильных устройств.

* 1. **Диаграмма вариантов использования**

Диаграмма вариантов использования представляет собой графическое изображение взаимодействия между актерами (пользователями) и системой «BdayCalendar». Эта диаграмма служит для иллюстрации различных сценариев использования программного продукта, выявления основных функциональных возможностей и понимания того, как система взаимодействует с внешними сущностями.

Данная диаграмма направлена на визуализацию ключевых вариантов взаимодействия пользователей с программой «BdayCalendar». Она помогает идентифицировать функциональные возможности системы, определить потребности пользователей и обеспечить более четкое понимание работы приложения на уровне использования.

Элементы диаграммы вариантов использования:

1. Добавление информации о дне рождения:

* Актер: Сотрудник по работе с персоналом.
* Система позволяет сотруднику вводить данные о днях рождения сотрудников, указывая их имена, даты рождения и фотографии сотрудников.

1. Редактирование информации:

* Актер: Сотрудник по работе с персоналом.
* Пользователь может вносить изменения в существующую информацию о днях рождения.

1. Удаление информации:

* Актер: Сотрудник по работе с персоналом.
* Система предоставляет возможность удалять устаревшие или ошибочные записи о днях рождения сотрудников.

1. Просмотр списка дней рождения:

* Актер: Сотрудник по работе с персоналом.
* Пользователь может просматривать список сотрудников с указанием имен сотрудников.

1. Оповещение о предстоящих событиях:

* Актер: Система.
* Система автоматически уведомляет пользователей о предстоящих днях рождения сотрудников.

Смотреть в Приложение А, Рисунок 1.

**1.3 Инструменты разработки:**

Для разработки программного продукта «BdayCalendar» с использованием Xamarin выбраны следующие инструменты. В качестве среды разработки (IDE) будет использоваться Visual Studio, предоставляющая широкие возможности для создания, отладки и сопровождения кода, а также интегрированная с Xamarin. Язык программирования C# для разработки кроссплатформенных мобильных приложений, благодаря своей читаемости и обширной поддержке библиотек.

Фреймворк Xamarin.Forms обеспечивающий возможность создания кроссплатформенных интерфейсов пользователя для Android и iOS с использованием единого кода. Для управления версиями кода и совместной разработки выбран Git, предоставляющий удобные инструменты отслеживания изменений.

Для управления базой данных в приложении «BdayCalendar» выбрана система управления базами данных SQLite. SQLite является легковесным и встроенным встроенным решением, что делает его отличным выбором для мобильных приложений, особенно в контексте кроссплатформенной разработки с использованием Xamarin. SQLite предоставляет надежный и эффективный механизм хранения данных, а также хорошо интегрируется с платформой Xamarin.

Конфигурация ПК, на котором ведется разработка, включает в себя процессор AMD Ryzen 5, 16 GB оперативной памяти, SSD-накопитель объемом 512 GB, а также графические процессоры NVIDIA 3050ti Laptop GPU и AMD Radion (TM) Graphics. Операционная система Windows 11. Эти инструменты и конфигурация обеспечат эффективное и удобное развитие программного продукта.

**1.4 Стратегия разработки и модель жизненного цикла**

Таблица 1 - Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | Являются ли требования к проекту легко определимыми и реализуемыми? | **Да** | **Да** | **Да** | Нет | Нет | Нет |
| 2 | Могут ли требования быть сформулированы в начале ЖЦ? | **Да** | **Да** | **Да** | Да | Нет | Нет |
| 3 | Часто ли будут изменяться требования на протяжении ЖЦ? | **Нет** | **Нет** | **Нет** | **Нет** | Да | Да |
| 4 | Нужно ли демонстрировать требования с целью их определения? | **Нет** | **Нет** | Да | **Нет** | Да | Да |
| 5 | Требуется ли проверка концепции программного средства или системы? | **Нет** | **Нет** | Да | **Нет** | Да | Да |
| 6 | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | **Нет** | **Нет** | **Нет** | Да | Да | Да |
| 7 | Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | **Да** | **Да** | **Да** | **Да** |

Вычисления:Каскадная – 6,V-образная – 6,RAD – 4**,** Инкрементная – 4**,** Быстрого прототипирования – 1**,** Эволюционная – 1.

Итог: На основе результатов заполнения табл. 1 подходящей является Каскадная или V-Образная.

Таблица 2 - Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории команды разработчиков  проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | **Да** | **Да** |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | **Да** | **Да** | Нет | Нет | Нет | **Да** |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | **Да** | **Да** | **Да** |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | **Нет** | Да | **Нет** | **Нет** |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | **Нет** | **Нет** |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | **Да** | **Да** | Нет | **Да** | **Да** | **Да** |

Вычисления:Каскадная – 2,V-образная – 2,RAD – 3**,** Инкрементная – 2**,** Быстрого прототипирования – 5**,** Эволюционная – 6.

Итог: На основе результатов заполнения табл. 2 подходящей является Эволюционная.

Таблица 3 - Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории коллектива пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | **Да** | **Да** | Нет | **Да** | Нет | **Да** |
| 2 | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | **Нет** | **Нет** | **Нет** | Да | Да | Да |
| 3 | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | **Нет** | **Нет** | Да | **Нет** | Да | **Нет** |
| 4 | Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | **Да** | **Да** |

Вычисления:Каскадная – 3,V-образная – 3,RAD – 1, Инкрементная – 2**,** Быстрого прототипирования – 1**,** Эволюционная – 3.

Итог: На основе результатов заполнения табл. 3 подходящей является Каскадная, V-Образная или Эволюционная.

Таблица 4 - Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

| № критерия | Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Разрабатывается ли в проекте продукт нового для организации направления? | **Нет** | **Нет** | **Нет** | Да | Да | Да |
| 2 | Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | **Нет** | **Нет** |
| 3 | Будет ли проект крупно- или среднемасштабным? | **Нет** | **Нет** | **Нет** | Да | Да | Да |

Продолжение таблицы 4

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 4 | Ожидается ли длительная эксплуатация продукта? | **Да** | **Да** | Нет | **Да** | Нет | **Да** |
| 5 | Необходим ли высокий уровень надежности продукта проекта? | **Нет** | Да | **Нет** | Да | **Нет** | Да |
| 6 | Предполагается ли эволюция продукта проекта в течение ЖЦ? | **Нет** | **Нет** | **Нет** | Да | Да | Да |
| 7 | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | **Нет** | **Нет** | **Нет** | Да | Да | Да |
| 8 | Является ли график сжатым? | Нет | Нет | **Да** | **Да** | **Да** | **Да** |
| 9 | Предполагается ли повторное использование компонентов? | **Нет** | **Нет** | Да | Да | Да | Да |
| 10 | Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? | **Нет** | **Нет** | **Нет** | **Нет** | Да | Да |

Вычисления:Каскадная – 8,V-образная – 7,RAD – 7, Инкрементная – 3**,** Быстрого прототипирования – 3**,** Эволюционная – 3.

Итог: На основе результатов заполнения табл. 4 подходящей является Каскадная

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является и каскадная модель, набравшая 19 баллов V-образная – 18, RAR – 15, Инкриментная – 11, Быстро прототипированная – 10, Эволюционная - 13.

**1.5 Стратегия разработки и модель жизненного цикла**

Первые две недели будут посвящены изучению xamarin и C#. С 6 по 20 октября вы планируете фокусироваться на освоении этих технологий. Затем следующие две недели (с 20 по 27 октября) также будут выделены для изучения xamarin и C#.

На третьей неделе, с 27 октября по 3 ноября, начнется написание проекта. За этот период вы сосредоточитесь на создании основных элементов вашего проекта. На следующей неделе, с 3 по 10 ноября, вы планируете завершить написание и приступить к тестированию проекта.

Завершающая, шестая неделя (с 10 по 15 ноября), будет посвящена составлению документации. Вы планируете документировать весь процесс разработки, включая создание, тестирование и внесение изменений.

Рисунок 1 - Диаграмма Гранта