**Введение**

На учебную практику была поставлена задачаразработать программный продукт «учёт решений по очереди на жильё» для сотрудников администрации Октябрьского района г. Гродно.

Цель данного проекта заключается в создании ПП для сотрудников администрации, работающих в отделе по учёту и распределению жилья. С помощью данного программного продукта пользователь сможет ставить человека на учет в базу данных.

Создаваемый ПП рассчитан на сотрудников администрации. Далее приведем краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». В нем можно ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, инструменты разработки, диаграмму Ганта и модель жизненого цикла программного обеспечания. Также в этом разделе можно узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе. Во втором разделе «Инструменты разработки» будет рассмотрена среда, в которой создается данный проект.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки программного продукта. В данном разделе будет описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации.

«Реализация задачи» – это третий раздел отчета, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения. В этом разделе описаны функции пользователя и их структура. Описано руководство программиста и диаграмма последовательности.

Четвертый раздел – «Тестирование». В нем будет описано функциональное и нефункциональное тестирование данной программы, т.е. будет протестирован каждый пункт меню, операция, которая выполняется программным продуктом. Будут смоделированы все возможные действия пользователя при работе с программным продуктом, начиная от входа на сайт заканчивая закрытием вкладки.

В разделе «Руководство пользователя» ПП предоставляет подробное руководство по использованию приложения для пользователей.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств.

«Список используемых источников» будет содержать список используемых при разработке источников.

В приложениях к пояснительной записке будут приведены диаграммы.

**1 Анализ задачи**

**1.1 Постановка задачи**

**1.1.1 Организационно-экономическая сущность задачи**

Темой данного проекта является разработка программного обеспечения «Учёт решений по очереди на жильё» для администрации Октябрьского района г. Гродно.

Целью проекта является разработка программного обеспечения (ПО) для учёта решений по очереди на жилье с упрощённым дизайном и интуитивно понятным интерфейсом.

Данный программный продукт (ПП) Разрабатывается для сотрудников администрации, работающих в отделе по учёту и распределению жилья.

Информация была выдана на месте прохождения практики – администрацией Октябрьского района г. Гродно.

В качестве примера можно рассмотреть аналогичный ПП по данной теме, которым на данный момент используется в администрации.

ПП, используемый в настоящее время, предоставляет такие возможности как внесение в базу информации о человеке, её редактирование и удаление; постановка и снятие человека на очередь на жильё. При создании программного продукта для учёта решений по очереди на жилье будут реализованы все эти возможности.

**1.1.2 Функциональные требования**

К поставленной задаче была заявлены следующие функциональные требования, который может выполнять гость:

− ввод данных;

− редактирование данных;

− удаление данных;

− поиск данных;

− сортировка данных;

− фильтрация данных;

− сохранение данных в базе данных(БД) ;

− печать отчётов.

Функциональные требования администратора:

− просмотр БД;

– создание запросов;

– редактирование запросов;

– удаление запросов;

– создание форм;

– редактирование форм;

– удаление форм;

– создание таблиц;

– редактирование таблиц;

– удаление таблиц;

**1.1.3 Описание процессов с входной, выходной и условно-постоянной информацией**

Вся информация, которой оперирует пользователь в процессе решения задачи подразделяется на:

– входную информацию;

– выходную информацию;

– условно-постоянную информацию.

Входной информацией выступают личные данные о человеке.

Выходной информацией выступают: отредактированный контент; данные о человеке.

Условно-постоянной информацией выступают: контент формы, данные о человеке.

**1.1.4 Эксплуатационные требования**

Для реализации ПП будет использоваться Microsoft Access.

При разработке программного продукта должны быть использованы преимущественного белые оттенки. Основные формы должны быть доступны с первой страницы. Грамотный пользовательский интерфейс.

**1.2 Диаграмма вариантов использования**

Диаграмма вариантов использования – диаграмма, отражающая отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Суть данной диаграммы состоит в следующем: проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью так называемых вариантов использования.

Актером или действующим лицом является любая сущность, взаимодействующая с системой извне. Это может быть как живое существо, так и любая друга система, которая может служить источником воздействия на моделируемую систему так, как определяет сам разработчик.

Вариант использования является стандартным языком UML и применяется для спецификаций общих особенностей системы и любой другой сущности. Отдельные варианты использования обозначаются на диаграмме эллипсом, в котором содержится его краткое название.

Отношение ассоциации является главным понятием языка UML и используется при построении всех графических моделей. Оно служит для обозначения роли актера и отдельном варианте использования. На диаграмме отношение ассоциации обозначается сплошной линией между актером и вариантом использования.

Для отображения взаимосвязи экземпляра отдельного варианта использования с более общим вариантом, используется отношение расширения, обозначаемое направленной пунктирной линией со стрелкой от исходного варианта. Данная линия помечается ключевым словом «extend».

Отношение включения между двумя вариантами использования указывает, что некоторое заданное поведение для одного варианта использования включается в качестве составного компонента в последовательность поведения другого варианта использования. Данная линия помечается ключевым словом «include».

Определяя для выбранного актера варианты использования и устанавливая отношения между вариантами использования, получается полная диаграмма вариантов использования. Её можно увидеть в Приложении А.

**2 Проектирование**

**2.1 Выбор стратегии разработки и модели жизненного цикла**

Для разработки программного продукта следует выбрать стратегию разработки и модель жизненного цикла. Выбрается посредством составления таблиц:

Таблица 1 – Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да | Нет | Нет | Нет |
| Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Да |
| Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |
| Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |

Вычисления: 4 за каскадную, 4 за V- образную, 5 за RAD, 5 за инкрементную, 3 за быстрого прототипирования и 3 за эволюционную.

Итог: на основе результатов заполнения таблицы 1 подходящей является RAD модель и инкрементная модель.

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории команды разработчиков проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | Да | Да |
| Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | Да |
| Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | Нет | Да | Нет | Нет |
| Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Нет | Да | Да | Да |

Вычисления: 5 за каскадную, 5 за V-образную, 4 за RAD, 5 за инкрементную, 2 за быстрого прототипирования и 1 за эволюционную.

Итог: на основе результатов заполнения таблицы 2 подходящими являются каскадная, V-образная и нкрементная модели.

Таблица 3 **–** Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории коллектива пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да | Нет | Да | Нет | Да |
| Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет | Да | Да | Да |
| Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет | Да | Нет | Да | Нет |
| Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | Да | Да |

Вычисления: 0 за каскадную, 0 за V-образную, 2 за RAD, 1 за инкрементную, 4 за быстрого прототипирования и 2 за эволюционную.

Итог: на основе результатов заполнения таблицы 3 подходящей является модель быстрого проектирования.

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



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Разрабатывается ли в проекте продукт нового для организации направления? | Нет | Нет | Нет | Да | Да | Да |
| Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | Нет | Нет |
| Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет | Да | Да | Да |
| Ожидается ли длительная эксплуатация продукта? | Да | Да | Нет | Да | Нет | Да |
| Необходим ли высокий уровень надежности продукта проекта? | Нет | Да | Нет | Да | Нет | Да |
| Предполагается ли эволюция продукта проекта в течение ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет | Да | Да | Да |
| Является ли график сжатым? | Нет | Нет | Да | Да | Да | Да |
| Предполагается ли повторное использование компонентов? | Нет | Нет | Да | Да | Да | Да |
| Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? | Нет | Нет | Нет | Нет | Да | Да |

Вычисления: 5 за каскадную, 6 за V-образную, 6 за RAD, 8 за инкрементную, 4 за быстрого прототипирования и 6 за эволюционную.

Итог: на основе результатов заполнения таблицы 4 подходящей является инкрементная модель.

Общий итог: в итоге заполнения таблиц – наиболее подходящей является инкрементная модель.

**2.2 Инструменты разработки**

Для разработки UML-диаграмм использовалось приложение «Draw.io», так как данное приложение имеет удобный и понятный интерфейс для составления диаграмм.

Для UX/UI дизайна использовалась «Figma», так как в данном приложении удобно работать с компонентами — заранее подготовленными элементами дизайна (кнопки, формы и т. д.), изменения которых распространяются на все его копии.

Для составления документации использовался «Microsoft Word», так как эта программа наиболее подходит для выполнения данной задачи.

Для разработки данного проекта будет выбран Microsoft Access. Microsoft Excel — для диаграммы Ганта

Разработка проекта будет происходить на компьютере со следующими

параметрами:

– процессор Ryzen 5;

– объем ОЗУ 8 GB;

– объем места на HDD: 256 GB;

– ОС: Windows 10 Домашняя.

**2.3 Разработка UML-диаграмм**

В первом разделе «Постановка задачи» были описаны функциональные требования к проекту. На их основе построена диаграмму последовательности.

Диаграмма последовательности – это диаграмма, предназначенная для моделирования взаимодействия объектов системы во времени, а также обмена сообщениями между ними.

Графические диаграммы последовательности имеют два представления. Одно – слева направо – в виде вертикальных линий, которые соответствуют линии жизни отдельного участка взаимодействия. Второе – вертикальная временная ось, направленная сверху вниз.

Диаграмма последовательности представлена в Приложении Б.

**2.4 Разработка пользовательского интерфейса**

Важным элементом проектирования данного программного продукта является описание внешнего интерфейса разрабатываемого программного продукта. Для разработки визуального дизайн использовались сдержанные цвета для удобства использования программного продукта. Разработанная структура приложения расположена в Приложении Г.

Для организации эффективной работы пользователя нужно создать целостный программный продукт данной предметной области, в котором все компоненты будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя. ПП должен позволить пользователю решать задачи, затрачивая значительно меньше усилий, чем при работе с разрозненными объектами. Все исходные данные будут разделены на несколько групп.

Диаграмма деятельности представлена в Приложении В.

UX/UI макеты представлены в Приложении Ж.

**2.5 Тесты на использование**

В ходе разработки ПП были составлены тесты, которые необходимо выполнить в дальнейшем. Тесты составлены таким образом, чтобы предусмотреть максимальное количество возможных действий. Тесты описаны в таблице в Приложении Е.

**2.6 Разработка плана работы над проектом**

Для разработки плана над проектом был использован Microsoft Excel. Microsoft Excel помог в создании плана разработки проекта, который позволил контролировать стадию выполнения. Это должно было повысить эффективность разработки. План разработки, составленный в Microsoft Excel, представлен в Приложении Д.

**3 Реализация**

Данный программный продукт был разработан с помощью Microsoft Access. (рисунок 1).

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 1 – Microsoft Access

Создаем пустую базу данных. Далее создаем необходимые таблицы, для этого нужно выбирать в панели пункт «Создание», затем в области «таблицы» выбрать «Таблица» (рисунок 2,3).

Изображение выглядит как текст, снимок экрана, Шрифт, Прямоугольник

Автоматически созданное описание

Рисунок 2 – Область “Таблицы”

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 3 – Таблица создана

После создания таблицы необходимо заполнить ее (рисунок 4).

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 4 – Таблица заполнена

После этого нужно дать название таблице и сохранить ее (рисунок 5).

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 5 – Сохранение таблицы

Чтобы создать форму нужно выбирать в панели пункт «Создание», затем в области «формы», выбрать «Мастер форм», (рисунок 6).

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 6.1 – Мастер форм

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 6.2 – Мастер форм

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 6.3 – Мастер форм

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 6.4 – Мастер форм

Также были добавлены связи (рисунок 7).

Изображение выглядит как текст, диаграмма, План, Шрифт

Автоматически созданное описание

Рисунок 7– Связи

Чтобы добавить компоненты на форме необходимо воспользоваться элементами управления (рисунок 8).

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Рисунок 8 – Элементы управления

**4 Тестирование**

Тестирование программного обеспечения было осуществлено в соответствии с тестами на использование, описаными в разделе «Тесты на использование».

|  |  |
| --- | --- |
| Таблица 8 – Тестирование функций | |
| № | Статус |
| Т1 | Выполнено успешно |
| Т2 | Выполнено успешно |
| Т3 | Выполнено успешно |
| Т4 | Выполнено успешно |
| Т5 | Выполнено успешно |
| Т6 | Выполнено успешно |

В результате проведения тестирования программного продукта выяснилось, что все раннее оговоренные функции были разработаны, а также протестированы. Тесты показали, что все функции работают правильно.

1. **Руководство пользователя**

**5.1 Общие сведения**

Наименованием программного продукта является приложение «Учет решений по очереди на жилье» для сотрудников администрации Октябрьского района г. Гродно. Этот ПП предназначен для сотрудников, работающих в отделе по учёту и распределению жилья

**5.2 Выполнение программы**

При выборе формы появляется появляется форма с полями (рисунок 9).

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 9 – Главная страница

Кнопки (рисунки 10,11) — предыдущая и следующая запись соответственно (рисунки 12,13).



Рисунок 10 — Предыдущая кнопка



Рисунок 11 — Следующая кнопка

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 12 — Предыдущая запись

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 13 — Следующая запись

Так же гость может, после заполнения всех полей, нажать на кнопку «Поставить на учет» для сохранения в таблице «гражданин» (рисунок 14), а также в «гражданство» (рисунок 15).



Рисунок 14 — Таблица «гражданин»



Рисунок 15 — Таблица «гражданство»

Пользователь может сохранить, редактировать и удалить запись (рисунок 16,17).

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 16 — Запись сохранена

Изображение выглядит как текст, снимок экрана, Шрифт, логотип

Автоматически созданное описание

Рисунок 17 — Запись удалена

**Заключение**

Целью данного учебного проекта являлась разработка программного продукта «Учет решений по очереди на жилье» для сотрудников администрации Октябрьского района г. Гродно, работающих в отделе по учёту и распределению жилья.

В ходе реализации поставленной задачи были закреплены знания по использованию Microsoft Access, а также курс дисциплины ТРПО.

Кроме того, разработанный программный продукт имеет простой интерфейс, который дает возможность пользоваться сайтом людям без специальных знаний в области компьютеров.

После тщательного тестирования приложения были выявлены некоторые недоработки, которые были полностью исправлены после на стадии тестирования. В целом при реализации программы, были выполнены все условия, перечисленные в предыдущих разделах пояснительной записки. Таким образом, можно сказать, что программа была реализована вполне успешно.

**Список использованных источников**

1 Видеохостинг YouTube: [Электронный ресурс] – Режим доступа: https://www.youtube.com/@AccesshelpRu /03.01.2023

Изображение выглядит как текст, диаграмма, рисунок, линия

Автоматически созданное описание

Рисунок А.1 – Диаграмма вариантов использования

Изображение выглядит как текст, диаграмма, линия, Параллельный

Автоматически созданное описание

Рисунок Б.1 – Диаграмма последовательности

Изображение выглядит как текст, диаграмма, линия, Параллельный

Автоматически созданное описание

Рисунок Б.2 – Диаграмма последовательности

**Изображение выглядит как диаграмма, текст, План, Параллельный

Автоматически созданное описание**

Рисунок В.1 – Диаграмма деятельности

**Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание**

Рисунок Г.1 – Структура приложения

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Параллельный

Автоматически созданное описание**

Рисунок Д.1 – Диаграмма Ганта

Таблица Е – Тесты на использование

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название теста | Действие | Исходная информация | Ожидаемый результат | Фактический результат | Статус |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Ввод информации | 1 Ввод личных данных в поля |  | Все поля заполняются | Все поля заполняются | Выполнено успешно |
| Запись человека в БД | 1 На форме заполнить данные о человеке  2 нажать «Поставить на учет» |  | 1 В таблицах «Гражданин» и «Гражданство» записываются данные о человеке | 1 В таблицах «Гражданин» и «Гражданство» записываются данные о человеке | Выполнено успешно |
| Кнопка «Предыдущая запись» | Нажать на кнопку «Предыдущая запись» |  | Отображаются данные о предыдущем человеке | Отображаются данные о предыдущем человеке | Выполнено успешно |
| Кнопка «Следующая запись» | Нажать на кнопку «Следующая запись» |  | Отображаются данные о следующем человеке | Отображаются данные о следующем человеке | Выполнено успешно |
| Кнопка «Удалить» | Нажать на кнопку «Удалить» |  | В таблицах «Гражданин» и «Гражданство» удаляются данные о человеке | В таблицах «Гражданин» и «Гражданство» удаляются данные о человеке | Выполнено успешно |
| Кнопка «Редактировать» | Нажать на кнопку «Редактировать» |  | Информацию можно редактировать | Информацию можно редактировать | Выполнено успешно |

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Автоматически созданное описание

Рисунок Ж.1 – UI макет формы

Изображение выглядит как текст, чек, диаграмма, снимок экрана

Автоматически созданное описаниеРисунок Ж.2 – UX макет формы