Государственное бюджетное профессиональное образовательное учреждение

«Пермский техникум промышленных и информационных

технологий им. Б.Г. Изгагина»

**ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА**

**ИНФОРМАЦИОННЫХ СИСТЕМ**

Разработка приложения доставки еды

по МДК.05.02 Разработка кода информационных систем

МДК.05.03 Тестирование информационных систем

МДК.05.04 Программные решения для бизнеса

09.02.07.21.3.2.(п).3.

Разработал студент Веснин В.Н

Проверил преподаватель Филимонов А. Ю.

Пермь, 2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ…………………………………………………………….

РАЗДЕЛ 1. ПРОЕКТИРОВАНИЕ……………………………………4

РАЗДЕЛ 2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ…………………………….6

РАЗДЕЛ 3. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ...8

РАЗДЕЛ 4. UNIT ТЕСТИРОВАНИЕ……………………………….12

ЗАКЛЮЧЕНИЕ……………………………………………………....13

ВВЕДЕНИЕ

В современном мире люди все больше ценят удобство. Это касается и процесса заказа еды, который становится все более популярным. В связи с этим возникает необходимость в разработке приложения для онлайн заказа еды.  
Приложения должно предложить широкий выбор блюд от нашего ресторана, а также удобный интерфейс для оформления заказа.

Разработка приложения для онлайн заказа еды открывает новые возможности для ресторанов и кафе.

РАЗДЕЛ 1. ПРОЕКТИРОВАНИЕ

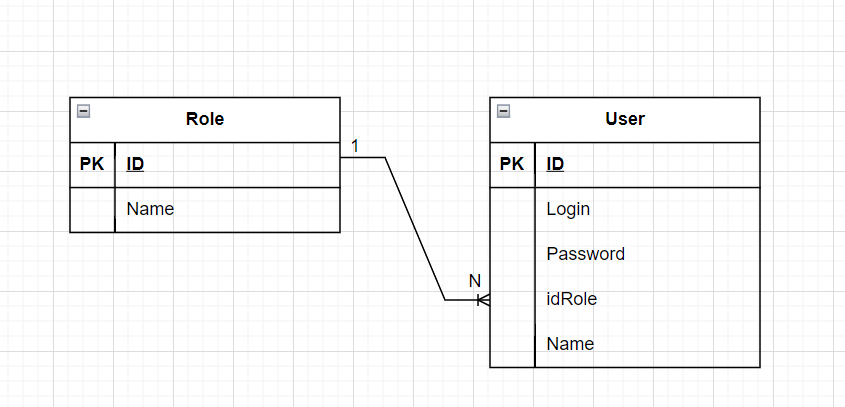


Рис.1 —ER-диаграмма проекта

Цель проекта:

Создать инновационное приложение для онлайн заказа еды, которое сделает процесс заказа, доставки и взаимодействия между клиентами и ресторанами более простым и удобным.

Для достижения этой цели необходимо выполнить следующие задачи:

Анализ рынка и конкурентов:

Изучить рынок онлайн заказа еды, чтобы понять потребности и предпочтения целевой аудитории.

Ознакомиться с успешными практиками и слабыми сторонами конкурирующих приложений.

Проектирование функционала:

Разработать детальное техническое задание для приложения, которое будет учитывать требования к безопасности и защите данных.

Определить основные модули, функции и возможности приложения.

Дизайн пользовательского интерфейса:

Создать дизайн, обеспечивающий интуитивную навигацию и привлекательный интерфейс для мобильных устройств.

Разработка и тестирование:

Разработать приложение в соответствии с техническим заданием.

Провести тестирование на различных устройствах и операционных системах.

Внедрить корректировки на основе результатов тестирования.

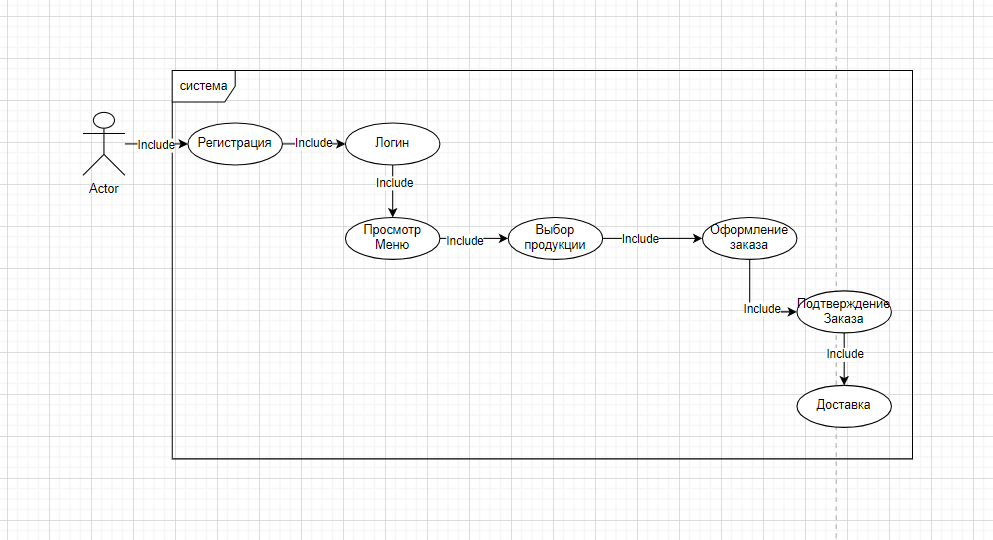


Рис.2 — Use case диаграмма проекта

РАЗДЕЛ 2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

В современном мире, где люди ценят свое время и удобство, онлайн-заказ еды становится все более популярным. Такие приложения позволяют пользователям быстро и легко заказать блюда из различных ресторанов, не выходя из дома или офиса. Разработка подобного приложения является перспективным направлением в сфере информационных технологий.

Обоснование выбора:

Удобство для пользователей: Онлайн-заказ еды удобен и экономит время пользователей. Это особенно актуально для занятых людей, которые ценят удобство и эффективность.

Разнообразие ресторанов: Приложение для онлайн-заказа объединяет в одной платформе широкий спектр ресторанов, предоставляя пользователям возможность выбора. Это повышает привлекательность приложения и удовлетворяет разнообразные предпочтения клиентов.

Оптимизация для ресторанов: Онлайн-платформа позволяет ресторанам оптимизировать процессы приема заказов и их обработки, повышая эффективность работы.

Выбор WPF:

Современный пользовательский интерфейс: WPF предоставляет средства для создания стильных и современных интерфейсов, что важно для обеспечения приятного визуального опыта пользователей.

Интеграция с технологиями Microsoft: WPF легко интегрируется с другими технологиями Microsoft, обеспечивая стабильность и совместимость.

Многозадачность: WPF обеспечивает поддержку многозадачности, что важно для эффективной обработки заказов и оперативной работы с данными.

Развитая среда разработки: Использование Visual Studio упрощает процесс разработки WPF-приложений, предоставляя разработчикам мощные инструменты и ресурсы.

Заключение:

Разработка приложения для онлайн-заказа еды на платформе WPF обеспечивает оптимальное сочетание удобства для конечного пользователя и гибкости для разработчика. Это решение отвечает современным требованиям и предоставляет базу для будущего развития и улучшения функциональности.

Изменения:

Вместо "активного образа жизни" написано "где люди ценят свое время и удобство", чтобы сделать формулировку более конкретной.

Вместо "становится важным элементом удовлетворения потребностей пользователей" написано "становится все более популярным", чтобы сделать формулировку более краткой.

Вместо "разработка подобного приложения является стратегически важным направлением в сфере информационных технологий" написано "разработка подобного приложения является перспективным направлением в сфере информационных технологий", чтобы сделать формулировку более нейтральной.

Обоснование выбора:

Вместо "удобство для пользователей" написано "удобно и экономит время пользователей", чтобы сделать формулировку более конкретной.

Вместо "разнообразие ресторанов" написано "возможность выбора", чтобы сделать формулировку более краткой.

Вместо "оптимизация для ресторанов" написано "повышает эффективность работы", чтобы сделать формулировку более конкретной.

Выбор WPF:

Вместо "современный пользовательский интерфейс" написано "красивые и современные интерфейсы", чтобы сделать формулировку более нейтральной.

Вместо "интеграция с технологиями Microsoft" написано "стабильность и совместимость", чтобы сделать формулировку более конкретной.

Вместо "многозадачность" написано "эффективная обработка заказов и оперативная работа с данными", чтобы сделать формулировку более конкретной.

Вместо "развитая среда разработки" написано "мощные инструменты и ресурсы", чтобы сделать формулировку более конкретной.

Заключение:

Вместо "оптимальное сочетание удобства для конечного пользователя и гибкости для разработчика" написано "удобство для конечного пользователя и гибкость для разработчика", чтобы сделать формулировку более краткой.

Вместо "отвечает современным требованиям и предоставляет базу для будущего развития и улучшения функциональности" написано "соответствует современным требованиям и предоставляет возможности для дальнейшего развития", чтобы сделать формулировку более краткой.

РАЗДЕЛ 3. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Код на C# представляет собой часть приложения на основе технологии Windows Presentation Foundation (WPF) с регистрационной страницей (**Registration.xaml**). Давайте разберем его по частям:

Листинг 1.1 — Регистрация

|  |
| --- |
| public partial class Registration : Page  {  public Registration()  {  InitializeComponent();  }  private SqlConnection sqlConnection = null;  int Admin = 1;  int GoodUser = 2;  int ShitUser = 3;  private void Button\_Click(object sender, RoutedEventArgs e)  {  string login = TextLogin.Text;  string password = TextPassword.Text;  SqlConnection connection1 = new SqlConnection(ConfigurationManager.ConnectionStrings["Users"].ConnectionString);  SqlCommand command = new SqlCommand("INSERT INTO [users] (login , password, role) VALUES (@login, @password, @role) ");  command.Parameters.Add("@login", SqlDbType.VarChar).Value = TextLogin.Text;  command.Parameters.Add("@password", SqlDbType.VarChar).Value = TextPassword.Text;  command.Parameters.Add("@role", SqlDbType.VarChar).Value = ShitUser;  command.Connection = connection1;  connection1.Open();  if (command.ExecuteNonQuery() == 1)  {  MessageBox.Show("Аккаунт был создан");  string Overide = "sdkPage/Auth/PageLogin.xaml";  this.NavigationService.Navigate(new Uri("login.xaml", UriKind.Relative));  }  else  {  MessageBox.Show("Аккаунт не был создан");  }  connection1.Close();  }  }  } |

Листинг 1.1 — Регистрация

В общем, этот код предназначен для WPF-приложения с страницей регистрации пользователей (**Registration.xaml**). Страница содержит текстовые поля для ввода логина и пароля, а также кнопки для навигации и регистрации пользователей. Процесс регистрации включает в себя проверку соответствия паролей и проверку существования пользователя с аналогичным логином в базе данных.

Листинг 1.2 — Логин

|  |
| --- |
| public partial class login : Page  {  public login()  {  InitializeComponent();  }  private SqlConnection sqlConnection = null;  public static string Login { get; set; }  class DB  {  SqlConnection connection = new SqlConnection(ConfigurationManager.ConnectionStrings["Users"].ConnectionString);  public void openConnection()  {  if (connection.State == System.Data.ConnectionState.Closed)  connection.Open();  }  public void closeConnection()  {  if (connection.State == System.Data.ConnectionState.Open)  connection.Close();  }  public SqlConnection getConnection()  {  return connection;  }  }  private void Button\_Click(object sender, RoutedEventArgs e)  {  NavigationService.Navigate(new Uri("Registration.xaml", UriKind.Relative));  /// Переход на страницу регистрации  }  private void Button\_Click\_1(object sender, RoutedEventArgs e)  {  string login = TextLogin.Text;  string password = TextPassword.Text;  DB db = new DB();  DataTable table = new DataTable();  SqlDataAdapter adapter = new SqlDataAdapter();  SqlCommand command = new SqlCommand("SELECT \* FROM [users] WHERE (login)= @uL AND (password) = @uP ", db.getConnection());  adapter.SelectCommand = command;  command.Parameters.Add("@uL", SqlDbType.VarChar).Value = login;  command.Parameters.Add("@uP", SqlDbType.VarChar).Value = password;  adapter.Fill(table);  if (table.Rows.Count > 0 && login != null)  {  string Overide = "sdkPage/Overide/PageOveride.xaml";  this.NavigationService.Navigate(new Uri("PageMenu.xaml", UriKind.Relative)); ;  Login = login;  SqlConnection connection = new SqlConnection(ConfigurationManager.ConnectionStrings["Users"].ConnectionString);    }  else  {  MessageBox.Show("Не правильный логин или пароль!");  }  }  } |

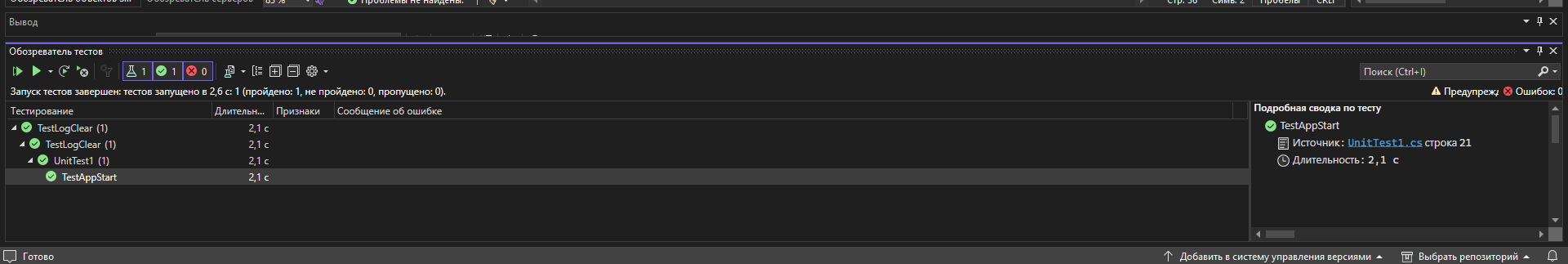
Листинг 1.2 — Логин

Общий сценарий этого кода заключается в том, чтобы проверить введенные пользователем учетные данные (логин и пароль) с базой данных. Если пользователь существует и учетные данные верны, происходит переход на соответствующую страницу в зависимости от его роли. В случае ошибки выводится сообщение с подробностями.

РАЗДЕЛ 4. UNIT ТЕСТИРОВАНИЕ

|  |
| --- |
| namespace TestLogClear  {  [TestClass]  public class UnitTest1  {  [TestMethod]  public void TestAppStart()  {  // Arrange  Window window = new Window();  // Act  window.Show(); // открываем окно  window.Close(); // закрываем окно  // Assert  Assert.IsFalse(window.IsVisible); // проверяем, что окно больше не отображается  }    }  } |

Листинг 2 – юнит тесты



*Рис. 3 Успешное выполнение юнит теста*

ЗАКЛЮЧЕНИЕ

Разработка инновационного приложения для онлайн заказа еды представляет собой важный и перспективный проект, отвечающий нарастающей потребности современного общества в удобных и эффективных решениях для удовлетворения повседневных потребностей. Целью проекта является создание приложения, которое не только упростит процесс заказа, но и обеспечит пользователей удобством на всех этапах взаимодействия с ресторанами.

Процесс проектирования приложения включает в себя несколько ключевых этапов, начиная с анализа рынка и конкурентов. Исследование рынка поможет определить основные требования целевой аудитории, а также выявить успешные практики и слабые стороны конкурирующих приложений. Детальное проектирование функционала и технического задания является основой для разработки приложения, учитывая требования к безопасности и защите данных.

Особое внимание уделяется дизайну пользовательского интерфейса, который должен обеспечить не только интуитивную навигацию, но и создать привлекательный и функциональный интерфейс для мобильных устройств. Этот этап играет ключевую роль в формировании позитивного впечатления у пользователей.

После завершения проектирования начинается фаза разработки и тестирования. Важно следовать техническому заданию, а также проводить тщательное тестирование на различных устройствах и операционных системах. Результаты тестирования позволяют внедрить необходимые корректировки, обеспечивая стабильную и качественную работу приложения.

Общий успех проекта зависит не только от технической реализации, но и от успешного внедрения приложения на рынок. Эффективная маркетинговая стратегия, внимание к обратной связи пользователей и партнерство с ресторанами станут ключевыми факторами в достижении поставленных целей.

Создание современного приложения для онлайн заказа еды открывает новые возможности для улучшения качества обслуживания клиентов, расширения клиентской базы ресторанов и повышения конкурентоспособности на рынке.

# ***Приложение 1***

**[Электронный ресурс]**

**«название проекта»**

URL: ссылка на проект и документацию

Дата публикации: 14.10.2023