Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет радиоэлектроники

Кафедра электронных вычислительных машин

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ОРГАНИЗАЦИИ БЭБИСИТЕРСТВА

Студент гр. 444601 О.А. Швед

Руководитель С. А. Кучук

МИНСК 2016

Содержание

[Введение 3](#_Toc402190757)

[1 Обзор литературы](#_Toc402190758) 4

[2 Структурное проектирование](#_Toc402190759) 7

[3 Функциональное проектирование](#_Toc402190761) 8

[4 Тестирование 1](#_Toc402190761)0

[5 Руководство пользователя 1](#_Toc402190762)2

[Заключение 1](#_Toc402190763)6

[Список используемых источников 1](#_Toc402190764)7

**вВедение**

В данной курсовой работе будет разработано программное средство для организации бэбиситерства. Пользователь получит возможность вести учет нянь и клиентов. Без всякого труда сможет добавлять или удалять записи о них, также он может сохранять все данные в базе данных.

Бэбиситтер - это практика временного ухода за ребёнком распространённая в США и странах Западной Европы.. Обычно в качестве бэбиситтеров нанимают подростков или студентов, как правило девушек. В обязанности бэбиситтеров могут включать не только наблюдение за маленькими детьми, но и игры с ними, чтение им книг, обучение девочек кулинарии или даже обучение вождению автомобиля.

Планируется реализовать такие функции как: список нянь, клиенты, отпуск няни, расписание для няни, заявки на услуги для няни. Данное приложение существенно упростить ведение бэбисетерства.

# **1 ОБЗОР ЛИТЕРАТУРЫ**

Согласно теме курсового проекта задачей проектирования является разработка прикладного приложения.

Построение удобного пользовательского интерфейса — это задача сама по себе достаточно сложная. Для создания пользовательского интерфейса существуют разные технологии. Рассмотрим некоторые из них.

Для выбора СУБД произведем анализ бесплатных популярных систем:

**MySQL** – вторая из самых используемая в мире open-source реляционная СУБД. Данная СУБД используется во многих крупных программных продуктах и на многих веб-сайтах, в том числе YouTube, Wikipedia, Facebook, и многих других. Система полностью бесплатна (в том числе и для коммерческого использования).MySQL создан с помощью языков C/C++, однако большое количество языков программирования включают в себя APIдля работы с MySQL (MySQLConnector / Netдля C# иVB, JDBC driver для Java, MyODBC для языков, поддерживающих интерфейс ODBC).

**Microsoft SQL Server** — система управления реляционными базами данных (РСУБД), разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

**PostgreSQL**– мощная и тяжелая система, отвечающая всем современным стандартам СУБД. Больше подходит для серьезных проектов, требующих сложных баз данных. По скорости работы **PostgreSQL** уступает MySQL. Также данная СУБД сложнее в администрировании, но является не чисто реляционной, а реляционно-объектной, что позволяет ей легко отражать иерархии объектов на реляционные таблицы.

В качестве источника данных будет использовать MSQL является надежной СУБД, не требующей высокопроизводительных компьютеров/

Для выбора GUIбиблиотеки, произведем анализ популярных продуктов:

**GTK+** (сокращение от **GIMP ToolKit**) — кроссплатформенная библиотека элементов интерфейса, имеет простой в использовании API, наряду с Qt является одной из двух наиболее популярных на сегодняшний день библиотек для X Window System. Будучи изначально частью графического редактора GIMP, она развилась в отдельный проект и приобрела заметную популярность. GTK+ — свободное ПО, распространяемое на условиях GNU LGPL, позволяющей создавать как свободное, так и проприетарное программное обеспечение с использованием библиотеки. GTK+ является официальной библиотекой для создания графического интерфейса проекта GNU.

**Qt**— кроссплатформенный инструментарий разработки ПО на языке программирования, также имеющий привязки ко многим другим языкам программирования. Позволяет запускать написанное с его помощью ПО в большинстве современных операционных систем путём простой компиляции программы для каждой ОС без изменения исходного кода. Включает в себя все основные классы, которые могут потребоваться при разработке прикладного программного обеспечения, начиная от элементов графического интерфейса и заканчивая классами для работы с сетью, базами данных и XML. Qt является полностью объектно-ориентированным, легко расширяемым и поддерживающим технику компонентного программирования.

**Windows Forms** – элемент Фреймворка .NET, построенный на основе устаревшего MFC и широко используемый в разработке графических интерфейсов многих приложений. Windows Forms предоставляет возможность разработки кроссплатформенного графического пользовательского интерфейса. С выходом .NET Framework 3.0 (в составе Windows Vista) Windows Forms была заменена каркасом Windows Presentation Foundation.

В данной работе будет использован Windows Forms, так как он прост в использовании и, в отличие от других библиотек, не требует установки дополнительных компонентов для разработки приложения (нужные компоненты поставляются вместе со средой разработки Visual Studio), а .NETфреймворк версии 3.0 или выше установлен на большей части используемых сейчас машин.

**2 Структурное проектирование**

**3 Функциональное проектирование**

**3 Тестирование**

Целью тестирования является проверка работоспособности разработанного программного обеспечения. На этой стадии необходимо проверить корректное функционирование разработанного программного обеспечения и соответствие его требованиям, выдвинутым в техническом задании. При выявлении несоответствий работы программы техническому заданию, либо ошибок, требуется доработка программного обеспечения и (или) документации. Разработанное программное обеспечение должно гарантировать устойчивое функционирование независимо от действий конечных пользователей. При возникновении отдельных ошибочных ситуаций разработанное программное обеспечение должно их успешно обрабатывать. Для проведения корректного тестирования сначала необходимо разработать порядок испытаний. Порядок испытаний - это список последовательности действий направленных на проверку корректности работы программы и (или) ее отдельных функциональных частей.

Существует два основных вида тестирования: функциональное и структурное. При функциональном тестировании программа рассматривается как “черный ящик” (то есть ее текст не используется). Происходит проверка соответствия поведения программы ее внешней спецификации. Возможно ли при этом полное тестирование программы? Очевидно, что критерием полноты тестирования в этом случае являлся бы перебор всех возможных значений входных данных, что невыполнимо.

Поскольку исчерпывающее функциональное тестирование невозможно, речь может идти о разработки методов, позволяющих подбирать тесты не “вслепую”, а с большой вероятностью обнаружения ошибок в программе. При структурном тестировании программа рассматривается как “белый ящик” (т.е. ее текст открыт для пользования). Происходит проверка логики программы. Полным тестированием в этом случае будет такое, которое приведет к перебору всех возможных путей на графе передач управления программы (ее управляющем графе). Даже для средних по сложности программ числом таких путей может достигать десятков тысяч.

Таким образом, ни структурное, ни функциональное тестирование не может быть исчерпывающим. Рассмотрим подробнее основные этапы тестирования программных комплексов. В тестирование многомодульных программных комплексов можно выделить четыре этапа:

Этапы тестирования приведены в таблице 4.1.

Таблица 4.1 – Этапы тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| Вид тестирования | Стадия, этап | Объект | Критерий |
| Структурное, надежности | Разработка | Компоненты | Покрытые ветвлений, функций |
| Сборочное | Разработка | Подсистемы | Функциональность, степень проверки компонентов |
| Функциональное | Разработка | Система в целом | Соответствие функциональным требованиям ТЗ |
| Регрессионное | Разработка, сопровождение | Система в целом | Проверка качества внесения изменений |
| Нагрузочное | Разработка, сопровождение | Система в целом | Оценка статистических характеристик системы, соответствие ТЗ, ТТХ, подбор конфигурация оборудования |
| Стрессовое | Разработка, сопровождение | Система в целом | Корректность работы системы при предельных нагрузках |

# **4** **Руководство пользователя**

Для работы компьютер должен удовлетворять следующим системным требованиям:

Операционная система: Window 7, Windows 8

* RAM: 128 Мб
* HDD: 10 Мб
* Устройства ввода: клавиатура, мышь.

Для того, чтобы подключить базу данных необходимо указать имя сервера, имя базы данных и, если есть, пароль (см.рисунок 5). Попадая на главное окно приложения, пользователь видит главную форму для работы с приложением. На ней расположены элементы работы с базой данных (см. рисунок 5.1 и рисунок 5.2).

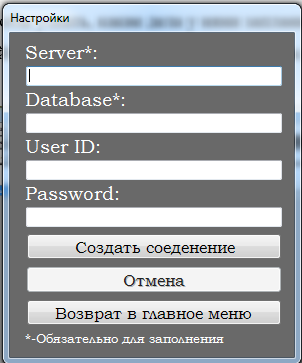


Рисунок 5 – Подключение базы данных

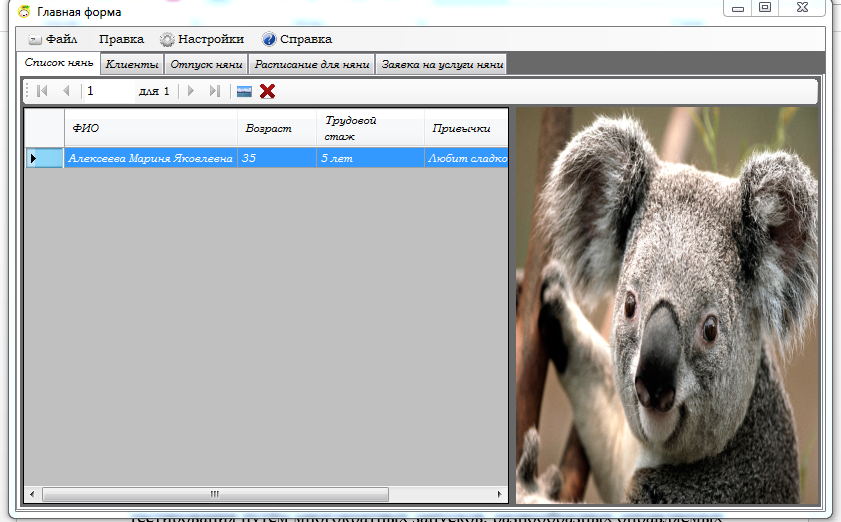


Рисунок 5.1 – Главная форма для работы с приложением

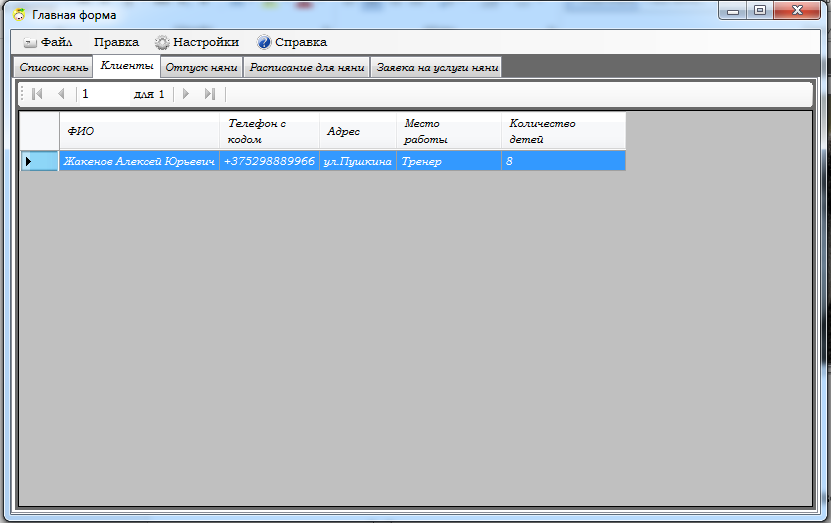


Рисунок 5.2-Форма данных о клиентах

Для того чтобы добавить клиента, необходимо перейти на вкладку “Правка “выбрать пункт добавить, после этого ввести данные (см.рисунок 5.3).

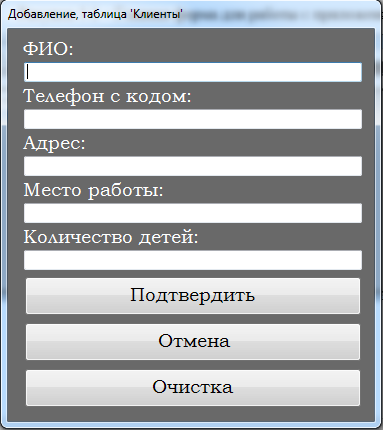


Рисунок 5.3 – Поля для заполнения данных о клиенте

Добавление данных в таблицу “Отпуск для няни” происходит ,как и добавление клиента, но заполняются другие данные, а так же причина ее ухода в отпуск(см.рисунок 5.4)

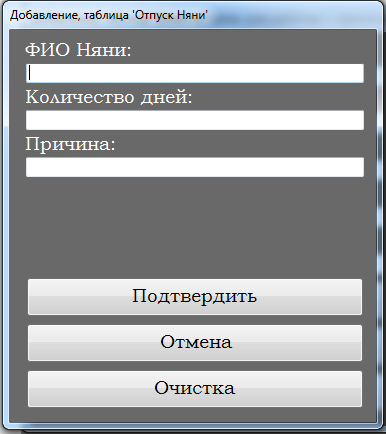


Рисунок 5.4- Уход няни в отпуск

Для того чтобы узнать, какие дела у няни запланированы на неделю, необходимо перейти на вкладку “Расписание няни”(см.рисунок 5.5)

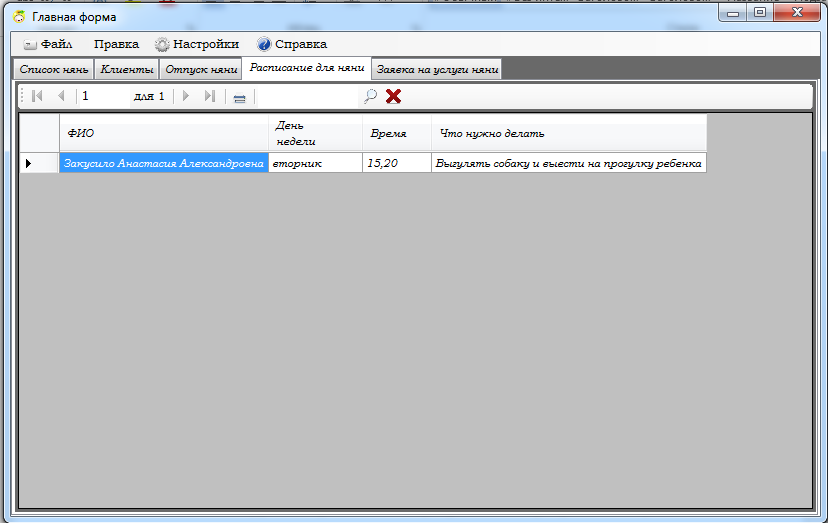


Рисунок 5.5 – Вкладка “Расписание для няни”

Для того чтобы добавить фото няни необходимо нажать на соответствующую иконку (см.рисунок 5.6).

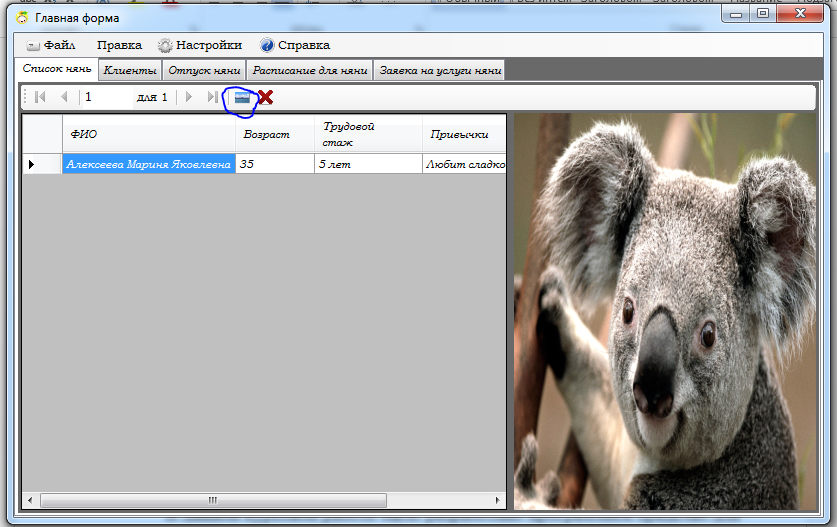


Рисунок 5.6 – добавление фотографии няни

**Заключение**

В данной курсовой работе было разработано программное средство для организации бэбиситерства.

Тестирование всего приложения проводилось методом ручного тестирования путём многократных запусков, разнообразных оправляемых событиях и моделируемых ошибочных ситуаций.

В ходе тестирования были выявлены ошибки в проектировании системы, места в программе, которые замедляют ее. Все данные недочеты были исправлены, что подтверждается в результате многочисленных тестовых запусках приложения.

# 

# 

# Список используемых источников

1. Начальный курс С и С++. Учебник. /Б. И. Березин. Москва:"ДИАЛОГ-МИФИ",1999г.
2. Язык программирования С++. Учебник. /. Страуструп. Киев:"ДиаСофт", 1993 г.
3. Структуры и алгоритмы обработки данных: Учебник. / Матьяш В.А., Путилов В.А., Фильчаков В.В. , Щёкин С.В. - Апатиты, КФ Петр ГУ, 2000.
4. С++ /Дэвис Стефан Р.,4-е издание : Пер. с англ.:- М.: Издательский дом «Вильямс»,2003.
5. Основы программирования: Учеб. Для сред. проф. образования /И.Г.Семакин, А.П.Шестаков. – М., 2006.
6. С++ экспресс курс: Учебник. /Лаптев В.В. – СПб.: БХВ- Петербург 2004.
7. С++ учебный курс: Учебник. /Франка П. – СПб.: Питер 2005.
8. МОДЕЛИ И CТРУКТУРЫ ДАННЫХ:/ Учебное пособие/
9. Д.Далека, А.С. Деревянко, О.Г.Кравец, Л.Е. Тимановская -Харьков: ХГПУ, 2000.