

# **ОПИСАНИЕ ОСОБЕННОСТЕЙ РЕАЛИЗАЦИИ КЛИЕНТСКОГО МОДУЛЯ СИСТЕМЫ АВТОМАТИЗИРОВАННОГО СБОРА ДАННЫХ**

**Версия 1.0**

## **СОДЕРЖАНИЕ**

<b>1. СТРУКТУРА ПРОЕКТА КЛИЕНТСКОГО МОДУЛЯ.....</b>	<b>1</b>
<b>2. ОСОБЕННОСТИ РЕАЛИЗАЦИИ И ИСПОЛЬЗОВАНИЯ НЕКОТОРЫХ КЛАССОВ.....</b>	<b>2</b>
<b>3. ФОРМАТЫ БИНАРНЫХ ФАЙЛОВ.....</b>	<b>4</b>
<b>4. ИНТЕРФЕЙС ПРИЛОЖЕНИЯ.....</b>	<b>6</b>

## Структура проекта клиентского модуля

**-логика приложения (классы и соответствующие файлы .h и .cpp):**

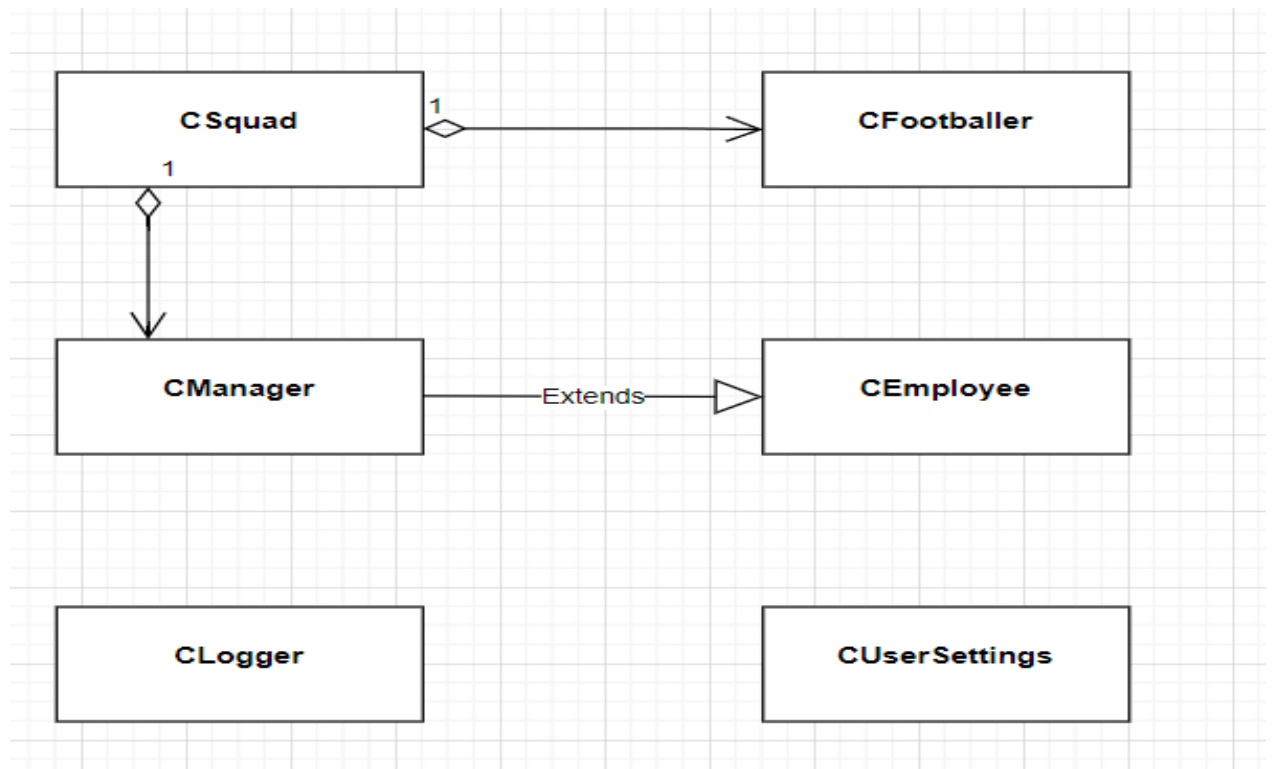
CFootballer	Класс футболиста
CSquad	Класс состава команды
CEmployee	Класс сотрудника персонала клуба
CManager	Класс тренера состава
CUserSettings	Класс настроек текущего пользователя
CLogger	Логгер (класс, работающий с логом событий и действий пользователя)

**-окна и GUI-элементы (классы и соответствующие файлы .h и .cpp):**

CAuthorizeDialog	Окно авторизации пользователя; агрегирует класс списка пользователей CUserList; обращается к свойствам CUserSettings
CRequestTasksDialog	Окно получения заданий
CLogDialog	Окно с логом событий
CImageTextButton	Класс кнопки с картинкой и текстом
CCloseButton	Класс кнопки для закрытия окна
CAddFootballerDialog	Окно добавления футболиста
CAddEmployeeDialog	Окно добавления сотрудника
CSquadDataDialog	Окно информации о составе команды
CEmployeeDataDialog	Окно информации о сотрудниках клуба

**-служебные файлы приложения:**

resource.h
stdafx.h и stdafx.cpp



(UML диаграмма основных классов приложения)

## ОСОБЕННОСТИ РЕАЛИЗАЦИИ И ИСПОЛЬЗОВАНИЯ НЕКОТОРЫХ КЛАССОВ

### **-CFootballer**

Данный класс содержит информацию о футболисте. В нём будут иметься следующие поля:

- имя
- фамилия
- возраст (int)
- зарплата (float)
- дата рождения (string)
- позиция на поле (string)
- дата окончания контракта (string)
- ид.номер (int)

### **-CSquad**

Класс содержит в себе информацию о составе игроков. Он агрегирует класс CFootballer и делит их на разные составы (основной состав, запас, резерв, молодежная команда), а также содержит поле id.

### **-CEmployee**

Класс содержит информацию о сотруднике клуба. Поля класса:

- имя
- фамилия
- дата рождения
- должность
- зарплата

### **-CManager**

Наследует класс CEmployee и расширяет его следующими полями: специализация и состав, с которым тренер работает

Для каждого из классов CFootballer, CEmployee и CSquad будет реализован метод ReadDataFromFile(string filename), который будет считывать информацию об объекте класса из соответствующего бинарного файла (см. форматы бинарных файлов)

### **-CLogger**

Класс для логирования всех событий и действий пользователя. Главная функция логгера – пишущая, т.е запись действий пользователя в соответствующий .dat файл с определённой структурой.

У каждого действия есть свой уникальный номер:

Добавить – 00

Изменить – 01

Удалить – 02

При этом соответствующие действия мы можем производить с футболистом/составом/сотрудником

Футболист – 00

Состав – 01

Сотрудник – 02

Таким образом мы можем закодировать соответствующее действие в виде двух двузначных чисел.

## ФОРМАТЫ БИНАРНЫХ ФАЙЛОВ

### Файл `footballer.data`

- 4 байта : ID футболиста (unsigned int)
- 4 байта : количество символов в имени футболиста M (unsigned int)
- M\*2 байт : M символов в имени футболиста (M\*sizeof(TCHAR))
- 4 байта : количество символов в фамилии футболиста M (unsigned int)
- M\*2 байт : M символов в фамилии футболиста (M\*sizeof(TCHAR))
- 4 байта : возраст (int)
- 4 байта : зарплата (float)
- 2 байта\*10 : дата рождения футболиста в формате DD.MM.YYYY (string)
- 2 байта\*2 : позиция футболиста на поле в формате из двух символов (CB,RB,ST и т.д) (string)
- 2 байта\*10 : дата окончания контракта футболиста в формате DD.MM.YYYY (string)

### Файл `employee.data`

- 4 байта : ID футболиста (unsigned int)
- 4 байта : количество символов в имени сотрудника M (unsigned int)
- M\*2 байт : M символов в имени сотрудника (M\*sizeof(TCHAR))
- 4 байта : количество символов в фамилии сотрудника M (unsigned int)
- M\*2 байт : M символов в фамилии сотрудника (M\*sizeof(TCHAR))
- 2 байта\*10 : дата рождения сотрудника в формате DD.MM.YYYY (string)
- 4 байта : количество символов в должности сотрудника M (unsigned int)
- M\*2 байт : M символов в должности сотрудника (M\*sizeof(TCHAR))
- 4 байта : зарплата (float)

### Файл `squad.dat`

- 4 байта : ID состава (unsigned int)

**-4 байта** : количество игроков в составе M (unsigned int)

**-4 байта \* M** : M подряд идущих id игроков в составе, каждый имеет тип (int)

#### Файл log.dat

**-8 байт** : Дата / время события (FILETIME)

**-4 байта** : тип операции (unsigned int)

**-4 байта** : к какому объекту системы она относится (игрок, состав, сотрудник)  
(unsigned int)

**-4 байта** : id объекта, к которому применена операция

## ИНТЕРФЕЙС ПРИЛОЖЕНИЯ

На данный момент дизайн приложения находится на начальной стадии разработки. Приложенные скриншоты являются лишь **наработками** интерфейса программы, и вполне могут видоизменяться до её конца разработки.

Окно входа пользователя (первым появляется при запуске приложения):

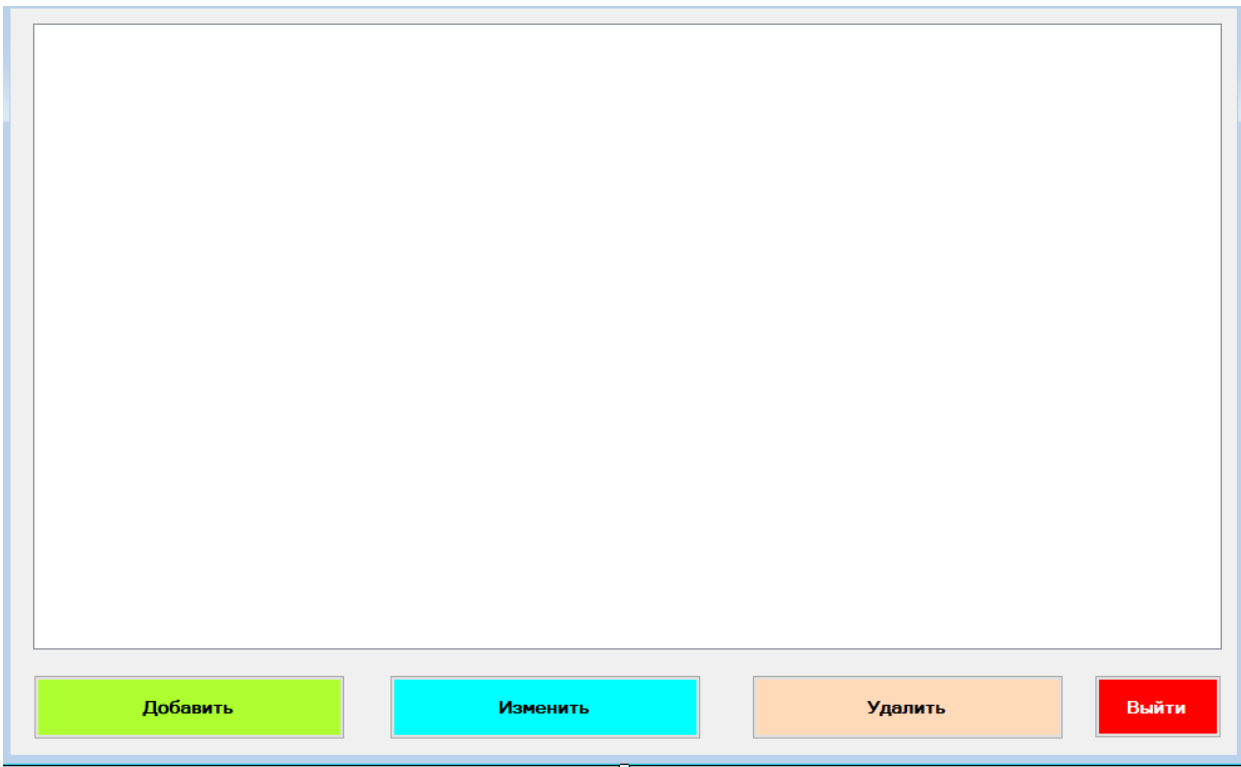
The screenshot shows a window titled "Form1" with a light blue title bar and standard Windows window controls (minimize, maximize, close). The main area has a light orange background. It contains the text "Логин" (Login) above a white text input field, followed by "Пароль" (Password) above another white text input field. At the bottom, there is a large green button with the text "Войти" (Login).

Окно выбора таблицы:

The screenshot shows a window titled "Form2" with a light blue title bar and standard Windows window controls. The main area has a light green background. It contains the text "Выберите желаемую информацию" (Select the desired information) in the center. Below this text, there are two buttons: an orange button on the left labeled "Футболисты" (Footballers) and a yellow button on the right labeled "Сотрудники" (Employees).



Таблица с данными:



The image shows a graphical user interface window. It features a large, empty rectangular area at the top, which is intended for a data table. Below this area, there is a horizontal bar containing four buttons. From left to right, the buttons are: a green button labeled 'Добавить' (Add), a cyan button labeled 'Изменить' (Edit), an orange button labeled 'Удалить' (Delete), and a red button labeled 'Выйти' (Exit).

Сама таблица находится в компоненте ListView. Ожидается, что она будет иметь стандартный вид (по типу таблиц в Access)