# Табло для самбо: Версия 3.0 (Финальная)

### Полный комплект файлов для сборки .exe

Все ваши требования реализованы в финальной версии 3.0 приложения.

### Реализованные изменения

### 1. Увеличенные фамилии спортсменов ✅

**Зрительское окно (spectator_window.py):**

```
# Фамилия борца (УВЕЛИЧЕННЫЙ ШРИФТ - 30px)
name_label.setStyleSheet(f"""
    QLabel {{
        font-size: 30px;
        font-weight: bold;
        color: white;
        background-color: {color};
        padding: 10px;
        border-radius: 5px;
    }}
""")
```

- Размер фамилий: **30px** (было 22px)
- Жирный шрифт
- Хорошо видно даже на большом расстоянии

### 2. Полная масштабируемость окон ✅

**Оба окна полностью масштабируемые:**

```
# Зрительское окно
self.setGeometry(100, 100, 1200, 700)
self.setMinimumSize(800, 500)  # Минимальный размер

# Судейское окно
self.setGeometry(50, 50, 1150, 800)
self.setMinimumSize(900, 600)  # Минимальный размер
```

**Возможности:**

- Можно перемещать мышью
- Можно растягивать/сжимать за края
- Установлен минимальный размер для читаемости

- Можно перенести на второй монитор/проектор
- Размер сохраняется при изменении

### 3. Функция отмены последнего действия ✅

**Реализация системы Undo в match_data.py:**

```
class MatchAction:
    """Класс для хранения действия (для отмены)"""
    def __init__(self, action_type, athlete_num, value, prev_state):
        self.action_type = action_type
        self.athlete_num = athlete_num
        self.value = value
        self.prev_state = prev_state  # Для восстановления

class MatchData(QObject):
    # История действий
    self.action_history = []

    def add_action(self, action_type, athlete_num, value):
        """Добавить действие в историю"""
        prev_state = self.save_state()
        action = MatchAction(action_type, athlete_num, value, prev_state)
        self.action_history.append(action)

        # Ограничиваем историю 20 последними действиями
        if len(self.action_history) &gt; 20:
            self.action_history.pop(0)

    def undo_last_action(self):
        """Отменить последнее действие"""
        if not self.action_history:
            return False

        last_action = self.action_history.pop()
        self.restore_state(last_action.prev_state)

        # Испускаем сигналы обновления
        self.score_changed.emit(1, self.athlete1_score)
        self.score_changed.emit(2, self.athlete2_score)
        self.warning_added.emit(1, self.athlete1_warnings)
        self.warning_added.emit(2, self.athlete2_warnings)
        self.action_undone.emit()

        return True
```

**Кнопка отмены в судейском окне:**

```
# КНОПКА ОТМЕНЫ (UNDO) - фиолетовая
undo_btn = QPushButton("↶ ОТМЕНИТЬ ПОСЛЕДНЕЕ ДЕЙСТВИЕ")
undo_btn.clicked.connect(self.undo_last_action)
undo_btn.setStyleSheet("""
    background-color: #9b59b6;
```

```
    color: white;
    font-size: 15px;
    min-height: 50px;
""")
```

**Что отменяется:**

- Начисление очков (+1, +2, +4)
- Добавление предупреждений
- Автоматическое начисление очков за предупреждения
- История до 20 последних действий

**Что НЕ отменяется:**

- Ввод имен и университетов
- Запуск/остановка таймеров
- Ручное объявление победы

### 4. Убраны всплывающие уведомления ✅

**ДО (версия 2.0):**

```
# Показывалось уведомление
QMessageBox.information(
    self,
    "Предупреждение добавлено",
    f"Борцу {athlete_num} выдано {warnings}-е предупреждение\n{points_msg}"
)
```

**ПОСЛЕ (версия 3.0):**

```
def add_warning(self, athlete_num):
    # Добавляем предупреждение
    result = self.match_data.add_warning(athlete_num)

    # Обновляем UI
    warning_display.setText(f"Предупреждения: {warnings}/3")
    score_display.setText(f"Счет: {score}")

    # БЕЗ ВСПЛЫВАЮЩИХ УВЕДОМЛЕНИЙ - только обновляем UI
```

**Что убрано:**

- ✅ Уведомления при добавлении предупреждений
- ✅ Уведомления при начислении очков за удержание (10 сек)
- ✅ Уведомления при начислении очков за удержание (20 сек)
- ✅ Уведомления о блокировке после окончания матча

**Что осталось (критичные подтверждения):**

- ⚠ Подтверждение дисквалификации (4-е предупреждение)
- ⚠ Подтверждение объявления победы вручную
- ⚠ Подтверждение сброса всего матча
- ⚠ Диалог выбора победителя при ничьей

## Структура проекта

```
sambo_scoreboard/
│
├── ОСНОВНЫЕ ФАЙЛЫ:
│   ├── main.py                  # Точка входа приложения
│   ├── match_data.py            # Модель данных + правила FIAS + Undo
│   ├── spectator_window.py      # Окно для зрителей
│   └── judge_window.py          # Окно для судей
│
├── ФАЙЛЫ ДЛЯ СБОРКИ .EXE:
│   ├── requirements.txt         # PyQt6&gt;=6.4.0, pyinstaller&gt;=5.13.0
│   ├── build.bat                # Автосборка для Windows
│   ├── build.sh                 # Автосборка для Linux/Mac
│   └── SamboScoreboard.spec     # Конфигурация PyInstaller
│
├── ДОКУМЕНТАЦИЯ:
│   ├── README.txt               # Полная инструкция (детальная)
│   └── START_HERE.txt           # Быстрый старт (краткая)
│
└── ОПЦИОНАЛЬНО:
    └── icon.ico                 # Иконка для .exe файла
```

## Инструкция по сборке .exe

## Способ 1: Автоматическая сборка (рекомендуется)

**Windows:**

```
# Дважды кликните на build.bat
# Или запустите в терминале:
build.bat
```

**Linux/Mac:**

```
chmod +x build.sh
./build.sh
```

**Результат:**

- Готовый .exe (или исполняемый файл) в папке `dist/`

- Размер: ~50-80 МБ
- Включает Python runtime и все зависимости
- Не требует установки Python на целевом компьютере

### Способ 2: Ручная сборка

```
# 1. Установите зависимости
pip install -r requirements.txt

# 2. Создайте .exe
pyinstaller SamboScoreboard.spec

# Или базовая команда:
pyinstaller --onefile --windowed --name="SamboScoreboard" main.py
```

### Способ 3: С пользовательской иконкой

```
# Если у вас есть icon.ico
pyinstaller --onefile --windowed \
    --icon=icon.ico \
    --name="SamboScoreboard" \
    main.py
```

### Параметры PyInstaller

Используемые параметры в SamboScoreboard.spec:

| Параметр | Значение | Описание |
|---|---|---|
| `--onefile` | Да | Один .exe файл вместо папки |
| `--windowed` | Да | Без консольного окна |
| `--name` | "SamboScoreboard" | Имя .exe файла |
| `--icon` | icon.ico | Иконка приложения (опционально) |
| `upx` | True | Сжатие для уменьшения размера |
| `console` | False | Графическое приложение |

### Использование готового .exe

1. **Скопируйте** `SamboScoreboard.exe` на любой компьютер
2. **Запустите** двойным кликом
3. **Готово!** Оба окна откроются автоматически

**Системные требования:**

- Windows 7/8/10/11 (64-bit)

- 4 ГБ RAM (рекомендуется)
- Разрешение экрана: минимум 1024x768

## Краткое руководство пользователя

## Судейское окно

**Основные действия:**

1. **Начало матча:**
   - Введите фамилии борцов
   - Нажмите [▶ Старт]
2. **Начисление очков:**
   - [+1] [+2] [+4] для каждого борца
   - Автоматическая проверка тотальной победы (12+ очков)
3. **Предупреждения:**
   - [⚠ Добавить предупреждение]
   - Очки начисляются автоматически: 1, 2, 1
   - БЕЗ всплывающих уведомлений
4. **Удержание:**
   - [ Начать удержание]
   - 10 сек → +2 очка (автоматически, без уведомления)
   - 20 сек → +4 очка + победа (автоматически)
5. **Отмена действия:** ⭐
   - [↶ ОТМЕНИТЬ ПОСЛЕДНЕЕ ДЕЙСТВИЕ]
   - Отменяет до 20 последних действий
   - Полезно при ошибке
6. **Завершение:**
   - [ Завершить матч] - определение победителя по FIAS
   - [ ПОБЕДА] - ручное объявление
   - [ СБРОС ВСЕГО] - новый матч

## Зрительское окно

- **Фамилии:** Крупные (30px)
- **Предупреждения:** 3 желтых квадрата ( )
- **Масштабирование:** Полная свобода
- **Второй монитор:** Автоматически определяется

## Правила FIAS

Все правила встроены и применяются автоматически:

### Предупреждения

| № | Очки сопернику | Визуализация |
|---|---|---|
| 1 | +1 очко | □□□ |
| 2 | +2 очка | □□□ |
| 3 | +1 очко | □□□ |
| 4 | Дисквалификация | Матч завершен |

### Удержание

- **10 секунд:** +2 очка
- **20 секунд:** +4 очка + тотальная победа

### Тотальная победа

- Разница **≥ 12 очков**
- Удержание **20 секунд**
- Дисквалификация (4 предупреждения)
- Болевой прием (кнопка "Победа")

### Победа по времени

- **8-11 очков разницы:** Победа по преимуществу
- **1-7 очков:** Победа по очкам
- **Равенство очков:** По количеству предупреждений
- **Полная ничья:** Решение судей (диалог выбора)

### Технические особенности

### Архитектура

**Model-View-Controller (MVC):**

- **Model:** match_data.py (данные + логика + история)
- **View:** spectator_window.py, judge_window.py
- **Controller:** Встроен в окна через обработчики кнопок

### Синхронизация данных

**Система сигналов PyQt6:**

```
# Model испускает сигналы
self.score_changed.emit(athlete_num, new_score)
self.warning_added.emit(athlete_num, warnings_count)
self.action_undone.emit()

# Views слушают сигналы
self.match_data.score_changed.connect(self.update_score)
self.match_data.warning_added.connect(self.update_warnings)
self.match_data.action_undone.connect(self.on_action_undone)
```

### История действий (Undo)

**Реализация:**

1. Каждое действие сохраняет предыдущее состояние
2. История хранит до 20 последних действий
3. При отмене восстанавливается полное состояние
4. Оба окна обновляются синхронно

**Что сохраняется:**

- Счета обоих борцов
- Количество предупреждений
- Флаг окончания матча
- Время удержания

### Преимущества версии 3.0

✓ **Увеличенные фамилии (30px)** - лучше видно на проекторе
✓ **Полная масштабируемость** - оба окна можно изменять
✓ **Функция отмены (Undo)** - исправление ошибок
✓ **Без уведомлений** - плавная работа без прерываний
✓ **Правила FIAS** - автоматическое применение
✓ **Один .exe файл** - легко распространять
✓ **Кросс-платформенность** - Windows/Linux/Mac

### Отличия от версии 2.0

| Функция | v2.0 | v3.0 |
|---|---|---|
| Размер фамилий | 22px | **30px** |
| Масштабирование окон | Частичное | **Полное** |
| Отмена действий | ✗ | **✓ До 20 действий** |

| Функция | v2.0 | v3.0 |
|---|---|---|
| Всплывающие уведомления | Много | **Только критичные** |
| Уведомления о предупреждениях | ✅ Есть | ❌ **Убраны** |
| Уведомления об удержании | ✅ Есть | ❌ **Убраны** |

## Заключение

Версия 3.0 - это финальная, полностью готовая к использованию версия табло для самбо с:

1. **Всеми запрошенными изменениями**
2. **Полным комплектом файлов для сборки .exe**
3. **Подробной документацией**
4. **Автоматическими скриптами сборки**

Приложение готово к использованию на соревнованиях по самбо любого уровня!

⁂