**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_bookmark0)

1. [Описание задачи](#_bookmark1) 3
2. [Техническое задание](#_bookmark2) 3
   1. [Общие требования](#_bookmark3) 3
   2. [Главное меню](#_bookmark4) 3
   3. [Выбор истории](#_bookmark5) 4
   4. [Интерфейсное окно выбранной истории](#_bookmark6) 5
3. [Пользовательское описание](#_bookmark7) 5
   1. [Главное меню](#_bookmark8) 5
   2. [Выбор истории](#_bookmark9) 6
   3. [Интерфейсное окно истории](#_bookmark10) 7
4. [Техническое описание деталей реализации](#_bookmark11) 8
   1. [Общие сведения](#_bookmark12) 8
   2. [Структура проекта](#_bookmark13) 8
   3. Комментирование кода 10
   4. Структура данных 11
5. [Техническое описание сборки и развертывания ПО](#_bookmark16) 12
   1. [Требования к ПО](#_bookmark17) 12
   2. [Сборка и развертывание ПО](#_bookmark18) 12

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ](#_bookmark20) 13

## ВВЕДЕНИЕ

В рамках данной курсовой работы применяются технологии программирования при разработке программы «Текстовые квесты». Программа представляет из себя развлекательный контент и предназначена для пользователей, которые хотели бы читать истории и влиять на их сюжет.

## Описание задачи

Разработать кроссплатформенное приложение «Текстовые квесты».

Для приложения разрабатывается пользовательский интерфейс, который должен быть интуитивно понятным пользователям приложения.

Пользователю предоставляется возможность прохождения выбранного текстового квеста. В главном меню есть возможность открыть окно с выбором истории. Пользователь осуществляет выбор и переходит в окно, где отображаются основной текст и две кнопки выбора. Когда квест будет завершен, пользователь может вернуться в главное меню.

## Техническое задание

## Общие требования

Разработать кроссплатформенное приложение «Квестовые тексты» на языке программирования C++ с использованием графической библиотеки Qt.

## Главное меню

Главное меню (его макет изображен на Рисунке 1) – стартовое интерфейсное окно, содержащее кнопки «Выбор истории», «Помощь» и «Выход».

После нажатия кнопки «Выбор истории» выполняется закрытие главного меню и открытие окна с выбором историй.

Также добавляются кнопки «Помощь» и «Выход». После нажатия кнопки «Помощь» открывается окно с сообщением. С помощью кнопки «Выход» осуществляется выход из программы.

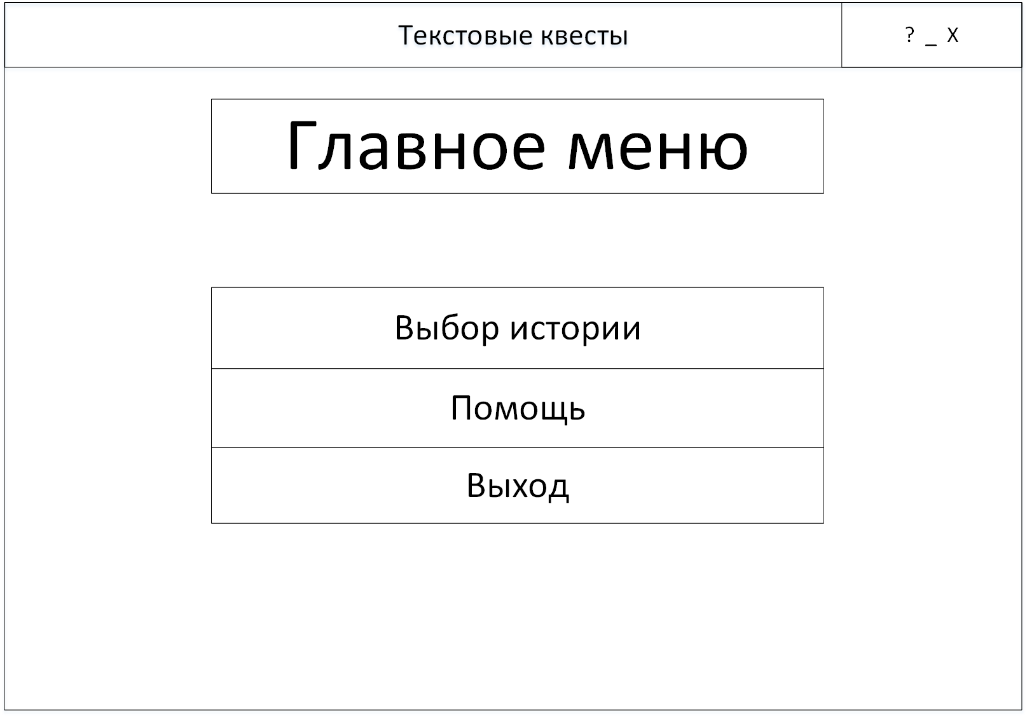


Рисунок 1. Макет главного меню

## Выбор истории

Интерфейсное окно выбора историй (его макет на Рисунке 2) можно открыть через главное меню. Добавляются кнопки двух историй и кнопка перехода в главное меню. При нажатии на одну из кнопок историй («Traum» или «Служба поддержки молящихся») закрывается окно выбора историй и открывается окно выбранной истории. После нажатия кнопки «Главное меню» пользователь возвращается в главное меню.

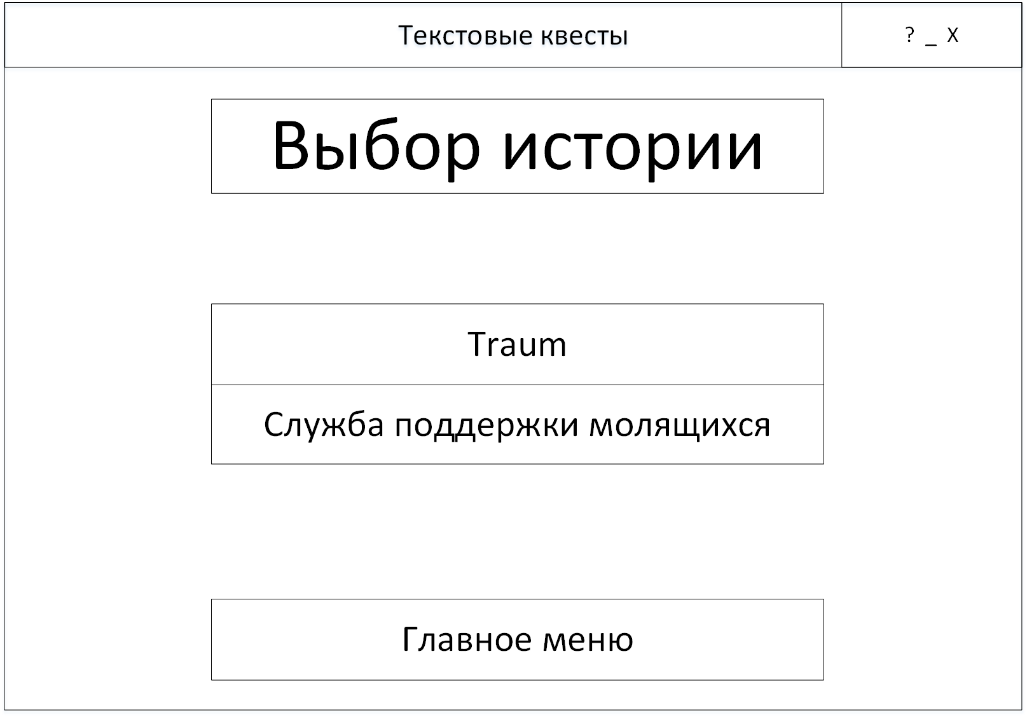


Рисунок 2. Макет окна выбора истории

## Интерфейсное окно выбранной истории

В подкаталоге «Data» находятся заранее подготовленные текстовые документы, пронумерованные в необходимом порядке. В зависимости от выбранной истории отображается нужный текст истории и текст вариантов ответов на кнопках. Пользователь делает выбор нажатием одной из кнопки выбора. После нажатия кнопки, в зависимости от сделанного выбора, отображаются новый текст истории и текст выборов. Также добавляется кнопка «Главное меню» для выхода в главное меню.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 3. Макет выбранной истории

## 3. Пользовательское описание

## Главное меню

После запуска программы открывается интерфейсное окно главного меню. Чтобы выбрать историю, необходимо нажать на кнопку «Выбор истории», после чего откроется соответствующее окно. Кнопка «Выход» позволяет выйти из программы. Она находится снизу в главном меню. Кнопку выхода можно также найти, открыв панель управления.

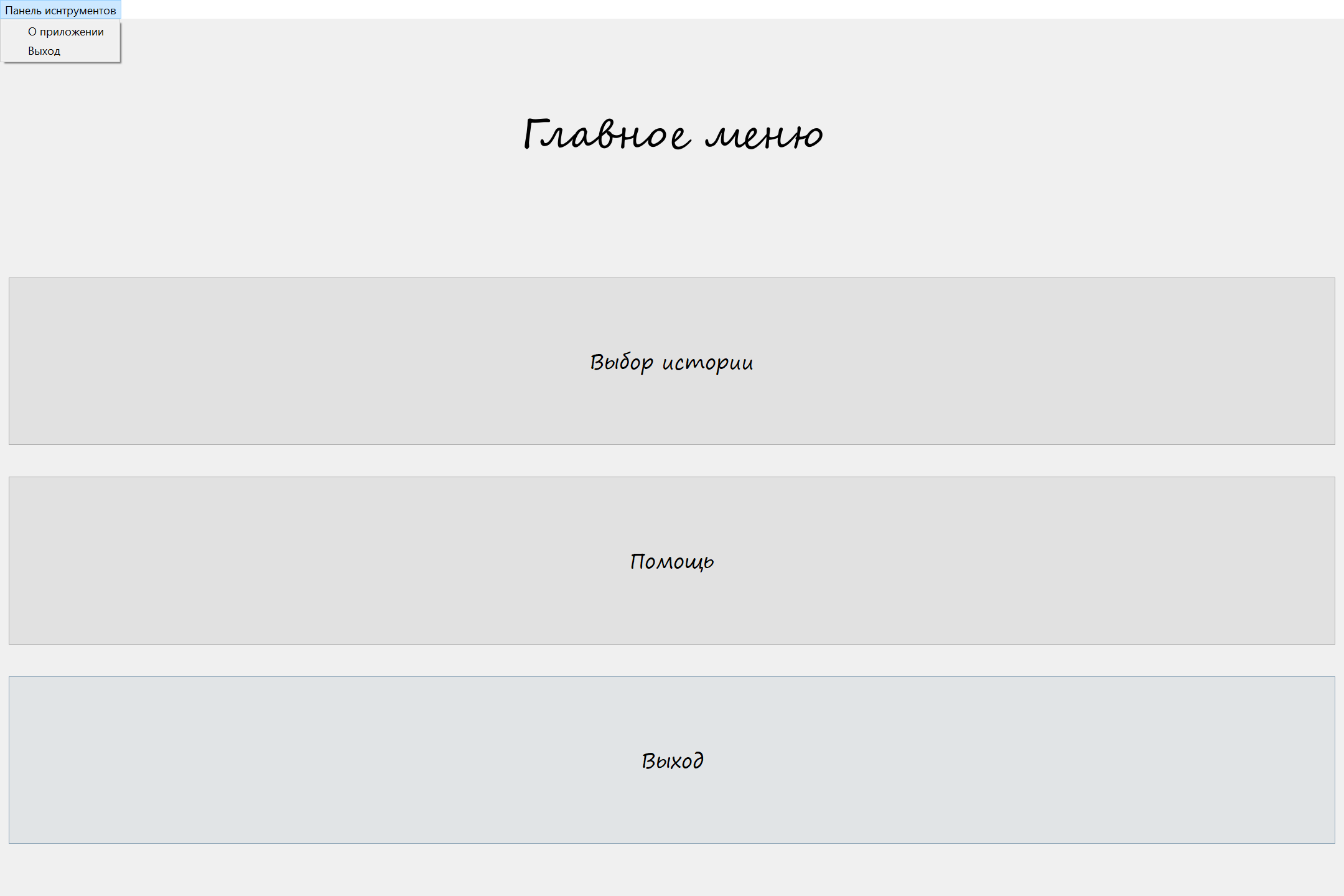


Рисунок 4. Интерфейсное окно главного меню

Помимо этого, открыв «панель управления» и выбрав соответствующую кнопку, можно узнать информацию о приложении. Если нажать на кнопку «Помощь» откроется окно с описанием приложения (Рисунок 5).

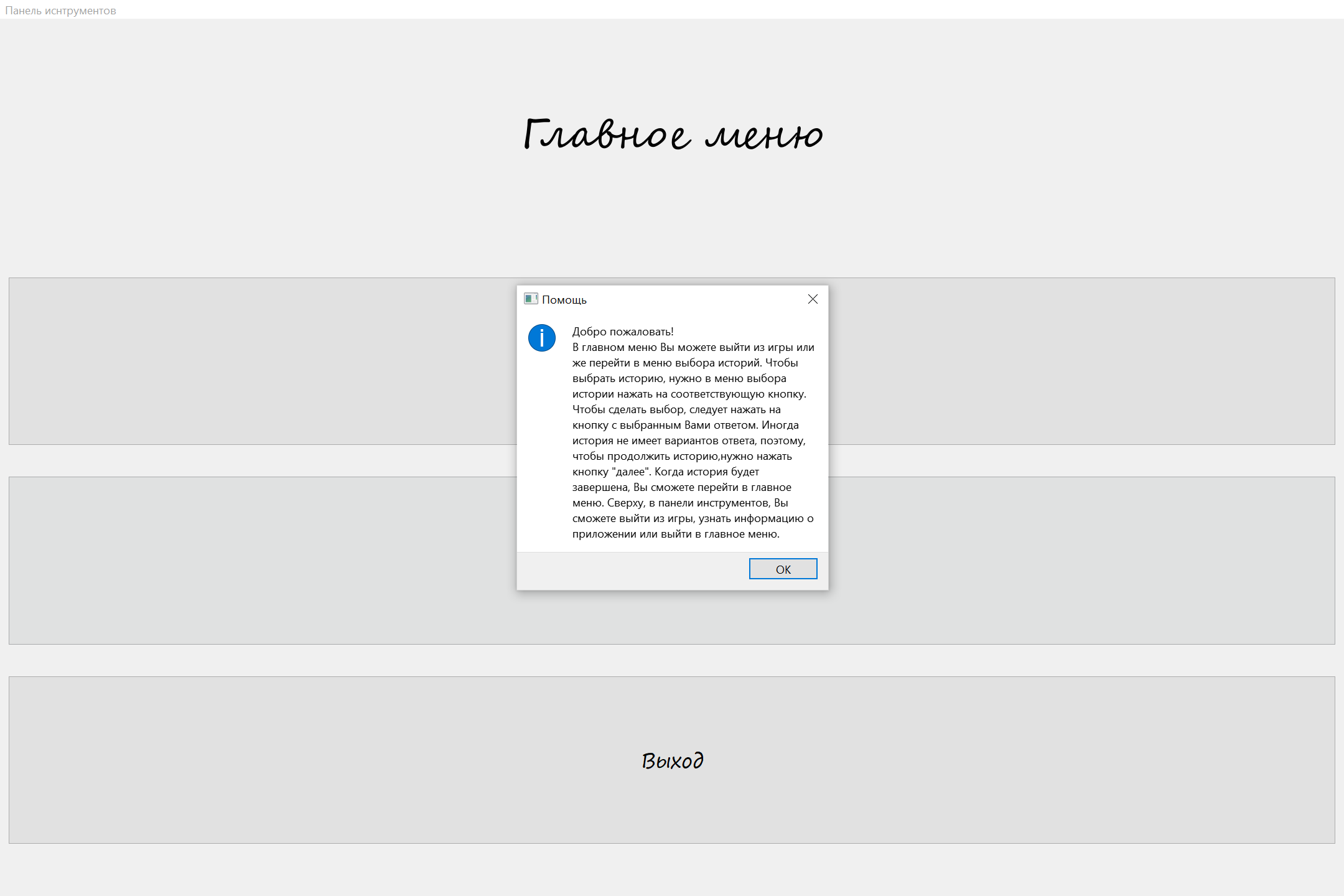


Рисунок 5. Описание приложения, открывающееся после нажатия кнопки «Помощь»

## Выбор истории

Из главного меню можно перейти в интерфейсное окно выбора истории, где будет предложено на выбор две истории. Чтобы выбрать историю, необходимо нажать на соответствующую кнопку с названием. С помощью кнопки «Главное меню» можно перейти в главное меню. «Панель инструментов» содержит следующие кнопки: «Помощь», «О приложении», «Выход».

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 6. Интерфейсное окно выбора истории

## Интерфейсное окно истории

После нажатия на кнопку выбранной истории откроется интерфейсное окно с текстом и выбором ответа. Чтобы осуществить выбор, нужно нажать на соответствующую кнопку. В «панели инструментов» находятся такие кнопки, как: «Главное меню», «Помощь», «О приложении», «Выход».

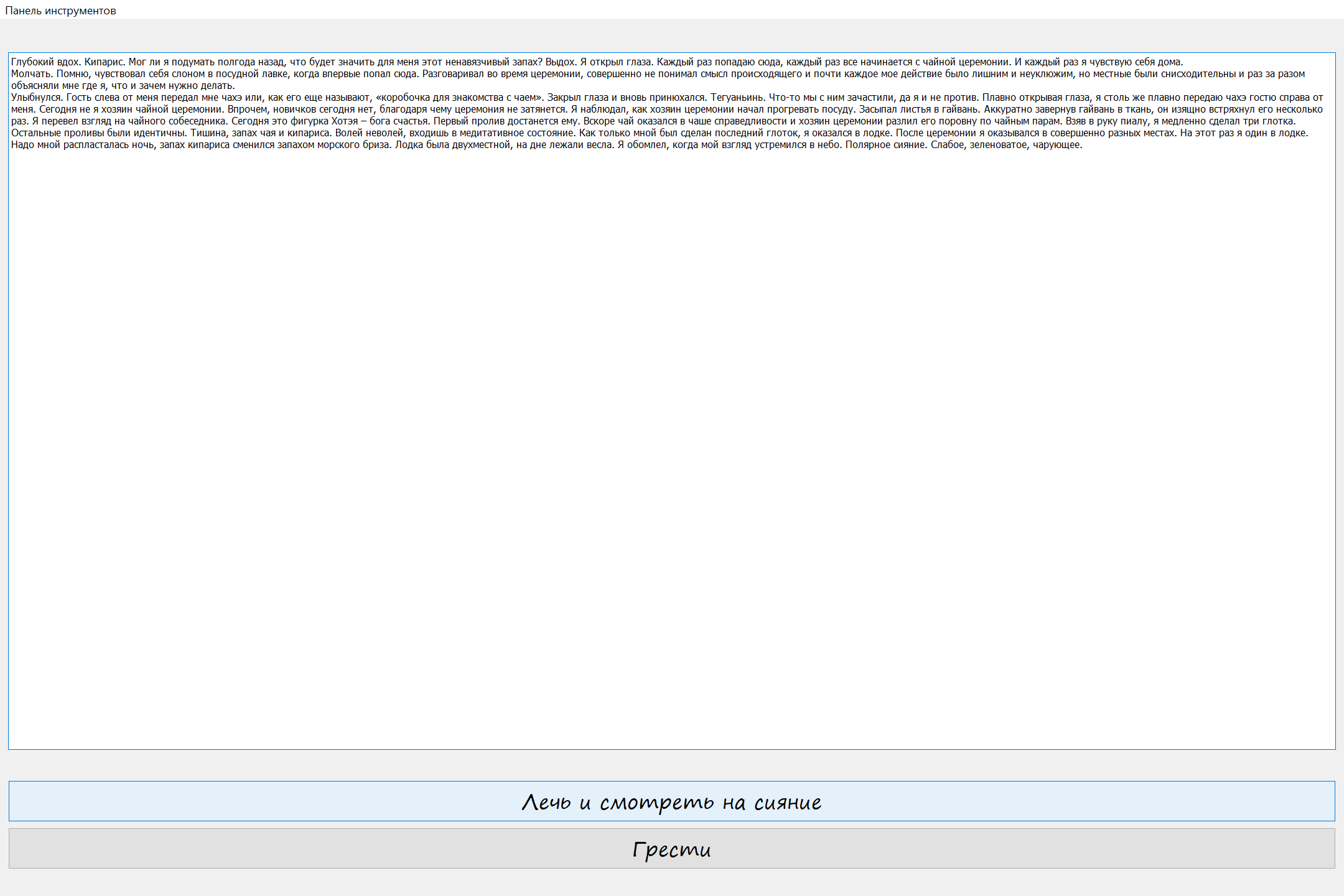


Рисунок 7. Интерфейсное окно истории с выбором ответа

Существуют интерфейсные окна истории без выбора ответов. Чтобы проходить историю дальше, нужно нажать на кнопку «Далее». «Панель инструментов» содержит те же кнопки, что и окно с выбором ответов.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 8. Интерфейсное окно истории без выбора ответа

Конец истории – интерфейсное окно с текстом и кнопкой перехода в «Главное меню». В «панели управления» можно найти кнопки: «Помощь», «О приложении», «Выход».

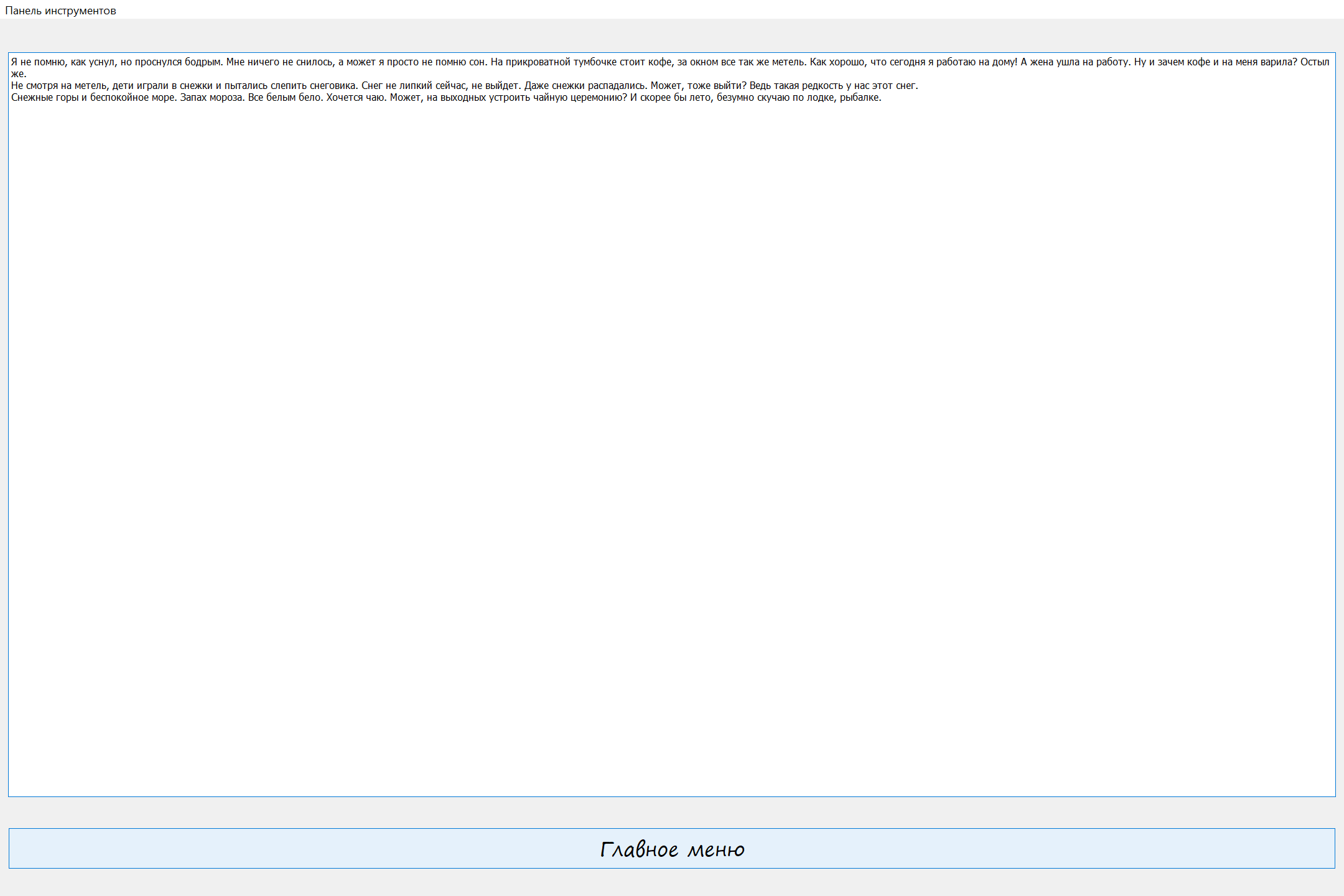


Рисунок 9. Интерфейсное окно конца истории.

## Техническое описание деталей реализации

## Общие сведения

Разработано кроссплатформенное приложение «Текстовые квесты» на языке программирования C++ версии 17 стандарта ISO/IEC.

## Структура проекта

Проект создается как приложение Qt Widgets. Структура проекта:

CMakeLists.txt – скрипт сборки проекта.

main.cpp - основной исходный файл, с которого запускается приложение.

mainwindow.h - заголовочный файл функционала главного меню (интерфейс главного меню).

mainwindow.cpp - исходный файл функционала главного меню (интерфейс главного меню).

mainwindow.ui – форма интерфейсного окна главного меню.

choosingstory.h - заголовочный файл функционала выбора истории (интерфейс выбора истории).

choosingstory.cpp - исходный файл функционала выбора истории (интерфейс выбора истории).

choosingstory.ui - форма интерфейсного окна выбора истории.

cpm.h - заголовочный файл функционала окна истории «Служба поддержки молящихся» с выбором ответа (интерфейс истории).

cpm.cpp - исходный файл функционала окна истории «Служба поддержки молящихся» с выбором ответа (интерфейс истории).

cpm.ui - форма интерфейсного окна истории «Служба поддержки молящихся» с выбором ответа.

сpm\_follow.h - заголовочный файл функционала окна истории «Служба поддержки молящихся» без выбора ответа (интерфейс истории).

cpm\_follow.cpp - исходный файл функционала окна истории «Служба поддержки молящихся» без выбора ответа (интерфейс истории).

cpm\_follow.ui - форма интерфейсного окна истории «Служба поддержки молящихся» без выбора ответа.

cpm\_end.h - заголовочный файл функционала окна конца истории «Служба поддержки молящихся» (интерфейс истории).

cpm\_end.cpp - исходный файл функционала окна конца истории «Служба поддержки молящихся» (интерфейс истории).

cpm\_end.ui - форма интерфейсного окна конца истории «Служба поддержки молящихся».

traum.h - заголовочный файл функционала окна истории «Traum» с выбором ответа (интерфейс истории).

traum.cpp - исходный файл функционала окна истории «Traum» с выбором ответа (интерфейс истории).

traum.ui - форма интерфейсного окна истории «Traum» с выбором ответа.

traum \_follow.h - заголовочный файл функционала окна истории «Traum без выбора ответа (интерфейс истории).

traum \_follow.cpp - исходный файл функционала окна истории «Traum» без выбора ответа (интерфейс истории).

traum \_follow.ui - форма интерфейсного окна истории «Traum» без выбора ответа.

traum \_end.h - заголовочный файл функционала окна конца истории «Traum» (интерфейс истории).

traum \_end.cpp - исходный файл функционала окна конца истории «Traum» (интерфейс истории).

traum \_end.ui - форма интерфейсного окна конца истории «Traum».

Техническое описание деталей реализации отражено в комментариях исходного кода. Для его получения формируется документация с использованием кроссплатформенной системы документирования исходных текстов Doxygen версии 1.9.1.

## Комментирование кода

Структура проекта описана с помощью комментариев и программы Doxygen. Ниже приведет фрагмент из собранного rtf файла.

На рисунке 10 мы видим, как выглядит структура проекта, описанная с помощью специальных комментариев в коде, которая при запуске Doxygen собирается в виде html-файла. Данный файл можно найти в подкаталоге Documentation.

**4.4 Структура данных**

Данные с текстами и вариантами ответов расположены в подкаталоге Data. Файлы формата txt пронумерованы и помещены в подкаталог в соответствии с названием истории. Нумерация файла зависит от развития истории. На схеме 1 приведен пример логической схемы обращения к данным в зависимости от развития сюжета истории Traum.

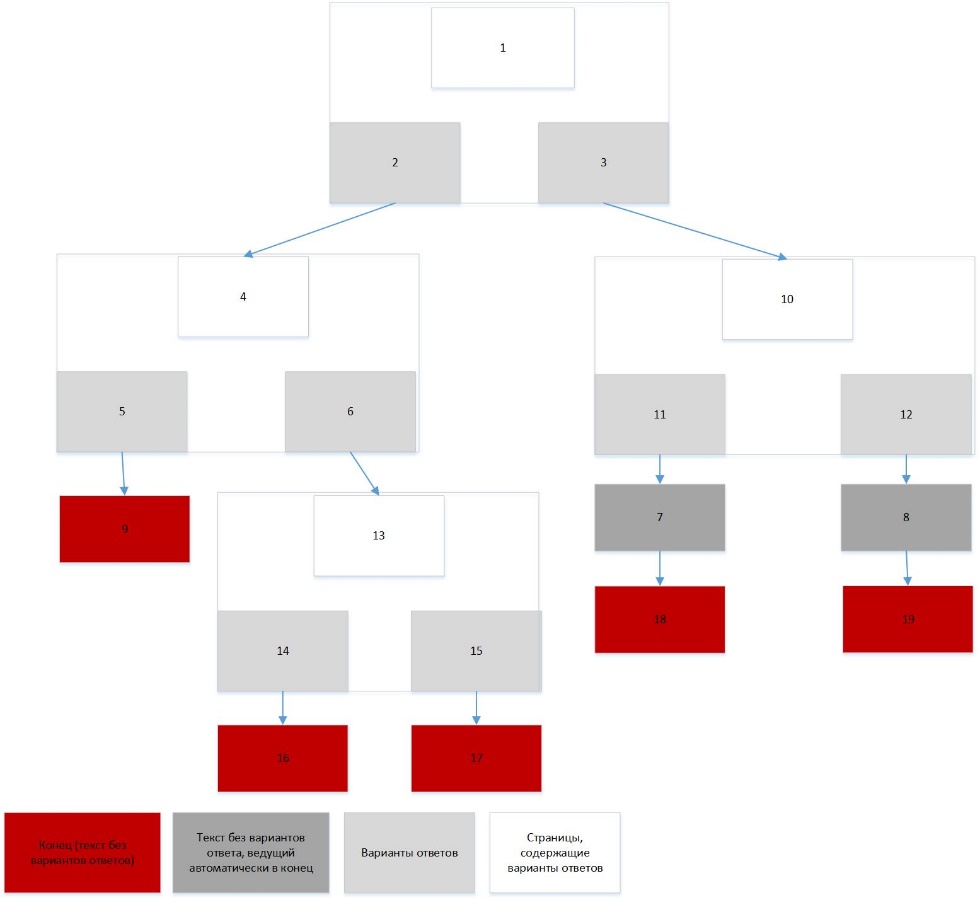


Схема 1. Логическая схема обращения к данным

## 5 Техническое описание сборки и развертывания ПО

* 1. **Требования к ПО**

Устанавливаются требования к программному обеспечению - IDE Microsoft Visual Studio 2019, кроссплатформенный фреймворк Qt версии 5.14.1 и CMake версии 3.19.2.

## Сборка и развертывание ПО

Для сборки приложения необходимо скачать исходный код по ссылке:

<https://github.com/Daria2001/Course_Work_TP/tree/master>

Сборка выполняется запуском файла build.bat в подкаталоге build.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Полевой Д.В. Уроки С++ (с Qt Creator).

url: <https://www.youtube.com/playlist?list=PL1D07918BD1371EED>

1. Documentation for Qt 5 releases.

url: <https://doc.qt.io/qt-5.12>

3. Легоцкой Евгений. Статьи C++ Qt.

url: <https://evileg.com/ru/knowledge/qt>