**2 Проектирование задачи**

# 2.1 Организация данных

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения.

Важным является он вследствие того, что методы, по средствам которых пользователь управляет формами, построены на высокой степени специализации каждого из компонентов.

Необходимым условием при разработке данного приложения является описание организации данных, т.е. логическая и физическая структура данных в контексте среды разработки. В разрабатываемой программе будут использоваться три вида данных.

Первым видом являются данные, которые будут введены разработчиком на этапе реализации задачи. Сюда можно отнести изображения (иконки, кнопки), описание, исходные коды теории в разделах.

Вторым видом данных, используемых в программе, является вводимая пользователем информация. К входной информации можно отнести вводимые пользователем значения, например ответы на вопросы при прохождения теста, ввод параметров автомобиля во вкладке «Тюнинг».

Третьим видом данных является результат программы – например, элементы автомобиля во вкладке «Тюнинг», которые пользователь не вводит, а выбирает посредством щелчка по нужному элементу. Их также относят к отдельному виду, так как ни пользователь, ни разработчик эти данные не вводят, а программа сама получает его в результате выполнения определенных действий.

Таким образом, организация данных является важной задачей при разработке данной и любой программы.

# 2.2 Процессы

Согласно всем перечисленным требованиям и указаниям, которые были рассмотрены в разделе «Анализ задачи», было определено, чем конкретно должна заниматься разрабатываемая курсовая программа. Главной ее задачей будет являться обучение пользователя и введение его в тему автомобилей.

Для реализации задач будут использоваться процедуры. С помощью процедур, например, будет осуществляться загрузка компонентов на форму, создание тестов.

Тест будет генерироваться на основе выбора пункта из трёх компонентов Image. Вопросы теста будут подгружаться исходя из выбора пользователя с помощью отключения или включения видимости компонентов Image. Далее после того, как пользователь прочитает материал (которым может управлять кнопками, сделанными с компонентом TImage), он сможет пройти тест для того, чтобы перейти к новой теме, в случае неправильного ответа программа ему порекомендует пройти тест заново (с начала уровня который он провалил).

Было бы важным на этапе проектирования рассмотреть еще состав и внешний вид используемой формы. Хотя проектирование формы заранее не так принципиально. Форму в любой момент времени достаточно просто можно изменить и отредактировать.

Однако хорошо организованная форма улучшает внешний вид программы. Поэтому эти вопросы более подробно будут рассмотрены ниже.

**2.3 Описание внешнего пользовательского интерфейса**

Важным при выполнении курсового проекта является организация диалога между пользователем и самой программой. Во многом это зависит от того, как программист разработает данную программу, какие компоненты будут использованы и какие методы будут автоматизированы.

Особое внимание следует уделить интерфейсу. Разработчик должен так организовать внешний вид своей программы, что бы пользователь понял, что от него требуется. При решении задачи будут использованы компоненты для ввода информации и её вывода, для вставки изображений, текста, и др.

Для организации эффектной работы пользователя нужно создать целостное приложение данной предметной области, в которой все компоненты приложения будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя.

Таким образом, для успешной работы всего проекта в целом следует обеспечить интуитивно понятный интерфейс с приятной гаммой цветов и шрифтами.

Структура навигации по проекту представлена на рисунке 2:

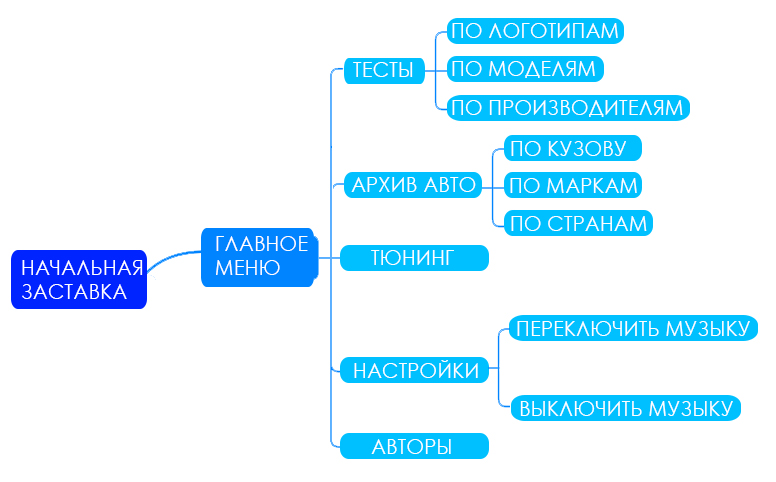


Рисунок 2 – Навигация между окнами программы