**Министерство образования и науки Российской Федерации**

**Федеральное государственное бюджетное образовательное**

**учреждение высшего образования**

**«Московский государственный университет технологий и управления**

**имени К.Г. Разумовского (Первый казачий университет)»**

**Университетский колледж информационных технологий**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**(ДИПЛОМНЫЙ ПРОЕКТ)**

**тема: Разработка прикладного приложения «Консультант скейтбордиста»**

**Студент:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Строков С.А.

(подпись студента) (Ф.И.О. студента)

**Группа:** **П-403**

**Специальность: 09.02.03 «Программирование в компьютерных**

**системах»**

**Руководитель:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Маликов Е.В.

(подпись руководителя) (Ф.И.О.руководителя)

**Консультанты:**

**Экономическая часть** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Блесткина О.В.

(подпись консультанта) (Ф.И.О.консультанта)

**Нормоконтроль** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гусева Е.Л.

(подпись консультанта) (Ф.И.О.консультанта)

Москва

2017

Введение

В настоящее время получило широкое распространение такое течение, как «скейтбординг» и с каждым годом все больше и больше как подростков, так и взрослых встают в их ряды.

Скейтбординг — это спорт современных людей, чьей неотъемлемой частью жизни являются информационные технологии. Живя в век высоких технологий людям стали приходить на помощь специальные программы-помощники, которые являются высокотехнологичными системами, включающими в себя множество элементов.

Прогресс не стоит на месте и каждый год можно наблюдать все новые и новые виды досок, что вводит новичков в растерянность при выборе своего первого скейтборда. Технологии не обошли стороной и эту область – теперь люди идут выбирать скейты в специализированные интернет-магазины, обращаясь за помощью к онлайн-помощникам.

Актуальность выбранной темы обусловлена тем, что спрос на программы-помощники как на рынке информационных технологий, так и в повседневной жизни неуклонно растет. Так как большинство людей в наше время привыкли обращаться за помощью к различным компьютерным программам, то и необходимость в создании подобного рода продукта стоит остро, особенно связанного с таким прогрессивным и современным течением, как скейтбординг.

Актуальность создания данного приложения заключается в том, что на данный момент не существует специальных программ-помощников для выбора скейтборда, поэтому ставится цель – создать актуальное на сегодняшний день программное решение, предназначенное для людей, желающих стать скейтбордистами. Программа должна не только помогать с выбором, но и снабжать актуальной информацией о различных моделях. Так пользователи смогут получить достаточно информации для ориентирования в предметной области.

Хочется отметить, что данная тема актуальна и для специализированных магазинов, где продаются скейтборды. Порой покупатели в растерянности от разнообразия досок и не могут остановиться на чем-то конкретном, а консультантам трудно объяснить все тонкости каждой отдельной модели, не прибегая к терминологии. Если дать пользователю воспользоваться данным приложением прежде чем обратиться за помощью к консультанту, то это даст покупателю готовую подборку подходящих ему моделей и необходимую информационную базу, что сильно облегчит и сократит время покупки.

Целью выпускной квалификационной работы является разработка прикладного приложения «консультант скейтбордиста», изучение принципов работы систем поддержки принятия решений и их реализации, закрепление знаний работы с базами данных, освоение новых методов разработки систем.

В соответствии с поставленной целью необходимо решить следующие задачи:

1. Изучить и составить описание параметров основных видов скейтбордов;
2. Разработать базу данных для хранения информации о скейтбордах;
3. Разработать ветку вопросов и ответов и внести все взаимоотношения в таблицы в базе данных;
4. Разработать приложение для пользователя и администратора;
5. Создать запросы на выборку для каждого вида скейтбордов;
6. Разработать интуитивно понятный дизайн приложения;
7. Проанализировать имеющиеся разработки в этой области.

Объектом исследования будут являться компьютерные автоматизированные системы, так как приложение напрямую связано с данным видом информационных систем.

Предметом исследования в свою очередь являются системы поддержки принятия решений.

Работа состоит из введения, четырех частей, заключения, списка использованной литературы, приложений, программы и базы данных.

1. Теоретическая часть
   1. **Анализ предметной области**

Данную программу можно выделить в класс программ-помощников. Программы такого рода служат для частичной замены экспертов-специалистов конкретной области и помогают пользователю со сложившейся проблемой.

Система поддержки решений СППР решает две основные задачи:

1. Выбор наилучшего решения из множества возможных (оптимизация);
2. Упорядочение возможных решений по предпочтительности (ранжирование).

Рассмотрим структуру СППР:

1. Информационные хранилища данных;
2. Средства и методы извлечения, обработки и загрузки данных;
3. Многомерная база данных и средства анализа;
4. Средства Data Mining.

По взаимодействию с пользователем выделяют три вида СППР:

1. Пассивные помогают в процессе принятия решений, но не могут выдвинуть конкретного предложения;
2. Активные непосредственно участвуют в разработке правильного решения;
3. Кооперативные предполагают взаимодействие СППР с пользователем. Выдвинутое системой предложение пользователь может доработать, усовершенствовать, а затем отправить обратно в систему для проверки. После этого предложение вновь представляется пользователю, и так до тех пор, пока он не одобрит решение.

В рассматриваемой теме дипломной работы подобный помощник-консультант реализован в качестве прикладного приложения с серверной частью. В зависимости от выбираемого пользователем варианта отсылается запрос на серверную часть и по установленным правилам приходят результаты на клиентскую часть.

Приложение «консультант скейтбордиста», является прикладной СППР, так как служит для анализа небольших объемов данных, и предназначена для одновременной работы одного пользователя. Так же это активная СППР, так как результатом работы программы является выдача наилучших вариантов из всех существующих.

* 1. **Анализ существующих разработок**

На сегодняшний день это является уникальной разработкой, так как прямых аналогов не существует, но имеются интернет-магазины, обладающие частичным функционалом. Анализ производится в таблице 1.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 1 – сравнительный анализ разработок | | | |
| Критерии | Allaboardshop | X3mboardshop | «Консультант скейтбордиста» |
| Вид приложения | Интернет - магазин и школа  по сноубордам и лонгбордам | Спортивный интернет-магазин | Прикладное приложение |
| Целевая аудитория | Клиенты, желающие приобрести экипировку для катания на скейтбордах и сноубордах, сами доски и комплектующие и аксессуары, а так же желающие записаться на обучение с инструктором по данным дисциплинам. | Клиенты, которые подбирают скейтборды, лонгборды, сноуборды, лыжи, специализированную экипировку и аксессуары. Так же предоставляет спонсорскую поддержку молодым райдерам. | Частные пользователи или клиенты специализированных спортивных магазинов, которые желают подобрать себе скейтборд или лонгборд, а также узнать подробную информацию о каждом. |
| Способ оказания помощи в | На сайте реализован фильтр товаров | Так же реализован фильтр товаров | Алгоритм подбора подходящих вариантов, через |
| подборе скейтборда |  |  | систему помощи принятия решений |
|  |  |  |  |
| Продолжение таблицы 1 | | | |
| Количество критериев, учитываемых при отборе | У фильтра стоит 6 критериев (рис. 1) | У фильтра 6 групп критериев, в каждой группе доступно несколько параметров с возможностью множественного выбора (рис. 4) | Пользователь отвечает на вопросы одним из трех предложенных вариантов, и в зависимости от варианта ответа ветка вопросов может либо дальше развиваться, либо предоставить готовый результат |
| Скорость выдачи результатов пользователю | Результаты выдаются сразу по окончанию заполнения пользователем фильтра | После выбора каждого нового критерия сортировки происходит автоматический подбор досок по параметрам, из-за чего сайт переходит в режим ожидания | Когда пользователь ответит на все предложенные вопросы – выдаются результаты |
| Способ выдачи результатов | Фильтр выдает ряд скейтбордов, заключенных в блоки (рис. 2) | Выводится отсортированный список скейтбордов, к тому же можно указывать сколько досок выводить сразу на | Результаты выдаются списком по три элемента на страницу |
| Продолжение таблицы 1 | | | |
|  |  | странице и по каким критериям производить сортировку результатов (рис. 5) |  |
| Просмотр одного выбранного варианта | После нажатия правой кнопкой мыши на блок раскрывается подробная информация о выбранном варианте | Если кликнуть правой кнопкой по одному из предложенных вариантов, то откроется экран с доской | Чтобы рассмотреть один вариант – достаточно нажать на кнопку «подробнее», находящейся в нижней части блока |
| Описание  Одной модели скейтборда | В данном окне можно рассмотреть подробную информацию о скейтборде, включая описание, основные характеристики, стиль катания и отзывы, оставленные покупателями этой модели. Так | Здесь очень подробно выведена вся необходимая информация о скейтборде и его параметрах. Это оптимальное количество сведений, в котором новичку будет трудно запутаться, но и достаточное, чтобы | В появившемся окне выводится вся доступная в базе данных информация о конкретном скейтборде, его характеристики, описание и компоненты. Эти сведения являются достаточными для пользователя, чтобы появилось понимание основ скейтбординга и было проще определиться с выбором. |
| Продолжение таблицы 1 | | | |
|  | же можно заметить те критерии, по которым проводилась фильтрация (рис. 3) | определиться с выбором (рис. 6) |  |
| Различия способов фильтрации | Тут используется обычная выборка, по немногочисленным параметрам, используется всего один пункт, где можно выбрать сразу несколько вариантов | В данном случае используется сложная множественная выборка, здесь в каждом блоке доступны для пользователя выбор от одного до всех возможных вариантов. | Алгоритм на каждом этапе добавляет, либо наоборот отсекает множество вариантов из всех доступных вариантов, так что дойдя до последнего вопроса будет сделана точная выборка по пожеланиям пользователя |
| Недостатки данного способа фильтрации | В данном случае разработчики отталкивались от комплекции покупателей, а не делали упор на комплектующие доски, это в итоге привело к тому, что | После выбора каждого нового критерия сортировки происходит автоматический подбор досок по параметрам, из-за чего сайт переходит в | Пользователь не знает точно по каким критериям идет фильтрация, а видит только уже результат. |
| Продолжение таблицы 1 | | | |
|  | единственный важный критерий сортировки – стиль катания, ибо по всем остальным критериям будет выдавать идентичные результаты | режим ожидания, это некомфортно пользователю, поэтому лучше дать сперва выбрать все критерии, а потом по нажатию на кнопку проводить фильтрацию. |  |
| Интуитивность интерфейса | Сам фильтр находится не на стартовой странице, чтобы перейти на него – необходимо кликнуть на закладку в верхней части панели. Сам фильтр реализован до банального понятно- пользователю лишь надо кликнуть по подходящему варианту (или | На данном сайте фильтр присутствует на каждой странице, где имеются скейтборды. Он реализован на боковой панели, в виде выпадающих блоков-групп с возможностью выбора многих вариантов, после выбора одного критерия сразу выводятся отобранные варианты | На начальном экране сразу предлагают начать и нажать на кнопку «старт», далее предлагается ответить на определенное количество вопросов, после чего выводятся результаты фильтрации, где пользователь может ознакомиться с каждым отдельно. |
| Продолжение таблицы 1 | | | |
|  | нескольким), а в конце подтвердить, нажав на кнопку «Подобрать борд» |  |  |

Рассмотрим пункт «Вид приложения». Можно заметить, что все аналоги представлены в виде веб-приложений, а именно - интернет-магазинов. Это является лучшим способом помочь в выборе товара, имеющегося в магазине, но когда встает необходимость лишь разобраться в предметной области, то ценники и другая излишняя информация сбивают с толку. К тому же, если отсутствует подключение к сети Интернет, то доступ к ресурсам веб-приложений теряется, а прикладное приложение всегда можно запустить, единожды установив. Поэтому стоит ставить именное такой вид программ на терминалы в специализированных магазинах, чтобы клиенты всегда могли получить оперативную помощь в подборе скейтбордов.

Следующий пункт «Целевая аудитория». К первым двум ресурсам обращаются клиенты с целью что-то купить или получить услугу, предоставляемую организацией. Следовательно, упор делается именно на покупателей и во многом весь функционал несет цель продать товар или услугу. Приложение «консультант скейтбордиста», в свою очередь, является приложением-помощником, и пользователи получают оперативные советы по выбору скейтборда, именно то, в чем и заключался основной смысл программы.

«Способ оказания помощи в подборе скейтборда» в данном пункте рассматриваются программные решения в каждом приложении. В онлайн-магазинах реализованы фильтры товаров, что является стандартным и практичным решением для большинства подобных веб-приложений. При создании программы «консультант скейтбордиста» была разработана система вопросов-ответов, где после каждого выбранного варианта идет фильтрация из общего количества скейтбордов по запросам к серверу. Данный способ позволяет быстро и достаточно точно определить, что необходимо пользователю, ведь один вопрос может затрагивать сразу несколько критериев отбора.

Рассматривая пункт «количество критериев, учитываемых при отборе», обратим внимание, что пускай на обоих онлайн-магазинах и реализованы фильтры, но подход совсем разный – на Allaboardshop учитываются всего 6 критериев, которых явно недостаточно для точного подбора скейтборда. На X3mboardshop уже используется сложный фильтр, где все критерии разбиты по одной тематике на 6 групп, но в данном случае пользователю легко запутаться в большом количестве терминологии и потратить гораздо больше времени. Приложение «консультант скейтбордиста» свело подбор доски к автоматизму и не требует от пользователя широких познаний в предметной области, требуя от него только выбирать один из предложенных вариантов. Всего в базе данных предусмотрено 14 вопросов и 42 ответа, которые перебирают все возможные комбинации и отбирают то, что наиболее подходит пользователю.

«Скорость выдачи результатов пользователю» - в данном пункте идет речь о том, когда пользователь сможет увидеть результат фильтрации. После сравнения всех трех способов, можно сделать вывод, что на X3mboardshop выбранный способ является не удачным, ибо в среднем ожидание длится около пяти секунд, а, учитывая, что критериев 6 групп и в каждой около десяти вариантов, то ожидание становится раздражающим фактором. Поэтому в «консультанте скейтбордиста» алгоритм вывода результата активируется после прохождения всех вопросов.

Следующим пунктом является «Способ выдачи результатов». Здесь рассматривается в каком виде пользователю предоставляется выводимые результаты фильтрации. Основным отличаем в этом пункте приложения «консультант скейтбордиста» от веб-приложений является вывод группами по три варианта, а не целым списком. Плюсом данного способа является возможность более детально изучить каждый предложенный вариант, заставляя задержаться на каждом более внимательно, а не пролистывать список сразу десятками. Такой способ подходит именно такому роду приложений, где вариантов не столь много и можно просмотреть каждый.

Далее идут два пункта «Просмотр одного выбранного варианта» и «Описание одной модели скейтборда», тут расписано как пользователь может ознакомиться с одним предложенным вариантом. Особых различий во всех трех рассматриваемых приложений нет, за исключением Allaboardshop. В данном онлайн-магазине слишком много лишней информации, которая не является ключевой при выборе скейтборда и лишь затрудняет понимание пользователя.

В следующих пунктах «Различия способов фильтрации» и «Недостатки данного способа фильтрации» разбираются основные отличительные особенности каждого подхода. После анализа, можно сделать вывод, что в приложении «консультант скейтбордиста» реализован наиболее совершенный алгоритм, так как в сравнении с Allaboarshop совершает гораздо более сложную выборку по большему количеству критериев, а по сравнению с X3mboardshop скорость выборки и практичность несколько выше.

После анализа пункта «Интуитивность интерфейса» приходим к выводу, что прикладное приложение удобнее использовать, так как все элементы на форме легко отличимые и пользователю интуитивно понятно какую функцию выполняет тот или иной компонент. В веб-приложении страницы сильно перегружены и часто приходится «гулять» по ним, пока не найдется нужное.

1. **Выбор инструментальных средств**

Работа с базой данных состоит из нескольких основных этапов и для каждого используется свое инструментальное средство:

* + 1. **Этап проектирования**

На этапе проектирования выбор пал на Sybase PowerDesigner. Это мощный и современный комплекс с возможностью концептуального и физического проектирования баз данных с и редактированием UML диаграмм. Можно выделить особенности, отличающие его от аналогового ряда:

1. Наличие единого репозитория, с возможностью работы сразу нескольких специалистов над одной моделью;
2. Возможность устанавливать только необходимые модули программы, предназначенные для конкретной задачи;
3. Улучшенное управление моделями, включая синхронизацию объектов, моделей и баз данных.

Для целей дипломного проекта использовалась Trial версия программы с установленными модулями :

1. PhysicalArchitect (PDM) – обладает всей необходимой конфигурацией для создания физической модели базы данных, а так же генерирования Sql-кода.
2. ObjectArchitect (PDM, CDM, OOM) – предназначено для визуального моделирования баз данных, разработчикам доступно 9 видов UMLдиаграмм.
   * 1. **Реализация базы данных**

После проектирования базы данных – необходимо составить скрипт создания на языке Transact-SQL и выполнить его в СУБД Microsoft Sql Server R8. Выбор данного программного продукта обусловлен широким спектром возможностей, которые необходимы для реализации выбранной темы. Удобный и интуитивно понятный графический интерфейс данной СУБД позволяет быстро переключаться между доступным функционалом и не затормаживать процесс разработки.

Между Sybase PowerDesigner и Microsoft Sql Server R8 просто установить связь и в случае, если понадобится оперативно изменить физическую модель базы данных, то такая возможность предусмотрена, без потери уже внесенных данных.

Очень удобная внутренняя файловая система, благодаря которой всегда можно быстро найти сохраненные хранимые процедуры, функции и запросы. Так же наличие встроенного динамического помощника очень упрощает написание запросов, посредством появляющегося списка объектов.

Еще одним большим плюсом является возможность изменения в структуре данных и создание резервных копий базы данных прямо во время работы сервера. А разграничение прав доступа, организованное на уровне пользователя, не даст ему совершить несанкционированные действия, поскольку есть возможность создать пользовательские привилегии.

* + 1. **Заполнение базы данных**

В ходе разработки системы необходимо было продумать дерево вопросов-ответов и его визуализировать. Самым удобным инструментальным средством, посредством чего создавались диаграммы является «Mind Manager».

Благодаря обширному функционалу данного продукта, не составляет большого труда составить необходимую схему, провести связи и сделать пометки. Имеется интеграция в более чем 700 других приложений, сохранение диаграмм во многих форматах, а также доступно множество инструментов для бизнес-планирования, анализа рисков и соблюдения крайних сроков.

В составленной диаграмме интуитивно понятная структура вопросов-ответов, благодаря чему не возникает труда заполнить саму базу данных.

* + 1. **Разработка приложения**

При разработке приложения выбор пал на среду разработки Lazarus - Delphi совместимый кроссплатформенной IDE. Языком программирования является Free Pascal, который используется специально для работы с синтаксисом Delphi, построенном на ООП. Lazarus является открытым программным продуктом (open source) и любой пользователь может создать свой модуль, а так же использовать уже имеющийся в свободном доступе. Особенностью Lazarus является совместимость продукта со многими платформами, например, создавая приложение под Windows, может возникнуть необходимость перенести приложение на Linux, но никаких трудностей не возникнет, ведь встроенное расширение упрощает данный процесс, не прибегая к написанию дополнительных строк кода.

Наличие большого количества компонентов позволяют создавать приложения, предназначенные для самых разнообразных областей. В целях дипломного проекта использовались следующие компоненты:

1. Tpanel – это основной класс, предназначенный для хранения других компонентов в себе;
2. Timage – компонент, предназначенный для вывода графического изображения на форму;
3. TGroupbox – предназначен для группировки схожих компонентов;
4. TLabel – компонент для вывода текста, который пользователь не может изменить в ходе работы программы;
5. TEdit – компонент для ручного ввода данных пользователем;
6. TDBtext – предназначен для вывода текста из присоединенной к программе базы данных;
7. TSqlQuery – служит для работы с Sql запросами, выдает данные в виде таблицы;
8. TSqlTransaction – предназначен для транзакции с базой данных;
9. MSSqlConnection – компонент для присоединения к базе данных MsSql server и поддержание соединения в ходе работы приложения.

Lazarus оснащен очень удобным и простым осуществлением перехода с формы на форму, что сильно ускоряет процесс разработки.

При написании кода очень помогает встроенные функции рефакторинга и автоформатировния.

Основным критерием, повлиявшим на выбор в качестве среды разработки Lararus – стала кроссплатформенность, так как приложение «консультант скейтбордиста» может использоваться пользователями разных операционных систем.

* + 1. **Графическое оформление приложение**

Каждому пользователю приятно использовать приложение не только с интуитивным графическим интерфейсом, но и визуально проработанным. В качестве графического редактора использовался продукт компании Adobe Systems, а именно - Photoshop CS6. Это на данный момент самый популярный и совершенный инструмент для работы с растровыми изображениями.

При разработке интерфейса Photoshop CS6 использовался для создания изображения, подходящего под размеры формы программы. В ходе разработки использовались различные встроенные инструментальные средства, позволяющие как редактировать изображение, так и создавать что-то новое. Например, самым используемым инструментом была «кисть», благодаря которой создавались основные элементы интерфейса.

После того, как все необходимые элементы интерфейса были созданы – изображение сохраняется в формате Png, поддерживаемый средой разработки Lazarus, и уже непосредственно используется для оформления приложения.

## Практическая часть

* 1. **Реализация ресурса**

Рассмотрим процесс разработки приложения поэтапно:

* + 1. **Этап проектирования**

На этапе проектирования была разработана структура базы данных. Итогом этого этапа стали две XML-схемы, описывающие основные таблицы и связи между ними. На рисунке 7 изображена физическая модель базы данных, на которой идет полное описание скейтборда, его характеристик и составляющих.

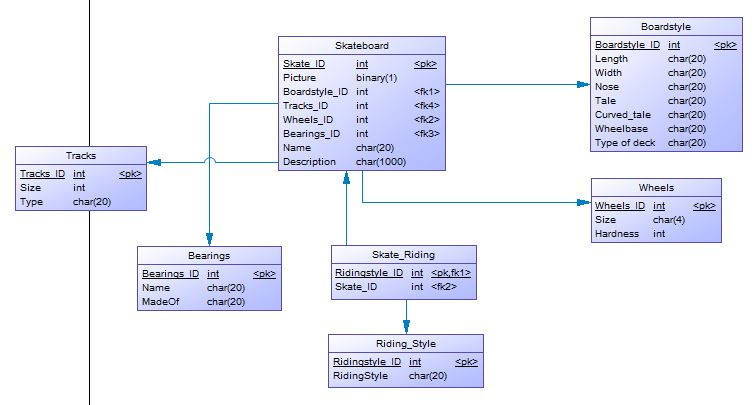


Рисунок 7 – Модель таблиц скейтборда.

Вторая физическая модель, изображенная на рисунке 8, описывает структуру, предназначенную для хранения вопросов и ответов, а также все взаимосвязи между ними.

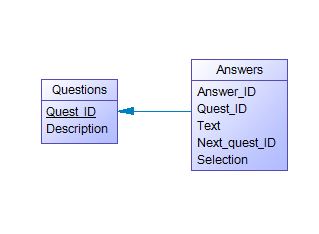


Рисунок 8 – Модель структуры вопросов-ответов.

* + 1. **Реализация базы данных**

Рассмотрим каждую таблицу, ее столбцы и предназначение.

**Таблица Skateboard**

Данная таблица является составной и содержит в себе полную информацию об одной модели скейтборда.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 2 – Описание таблицы Skateboard | | | |
| Название | Описание | Тип данных | Дополнительно |
| Skate\_ID | Уникальный идентификатор скейтборда. | int | Первичный ключ |
| Picture | Содержит в себе путь до изображения на диске | Nvarchar(max) |  |
| Boardstyle\_ID | Уникальный идентификатор вида деки. | Int | Внешний ключ |
| Tracks\_ID | Уникальный идентификатор подвесок. | Int | Внешний ключ |
| Продолжение таблицы 2 | | | |
| Wheels\_ID | Уникальный идентификатор колес. | Int | Внешний ключ |
| Bearings\_ID | Уникальный идентификатор подшипников. | Int | Внешний ключ |
| Name | Название скейтборда | Char(20) |  |
| Description | Описание скейтборда | Char(1000) |  |

**Таблица Riding\_Style**

Данная таблица содержит все виды стилей катания на скейтборде.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 3 – Описание таблицы Riding\_Style | | | |
| Название | Описание | Тип данных | Дополнительно |
| Ridingstyle\_ID | Уникальный идентификатор стиля катания. | Char(10) | Первичный ключ |
| RidingStyle | Название стиля катания | char(20) |  |

**Таблица Boardstyle**

Данная таблица содержит в себе характеристики деки.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 4 – Описание таблицы Boardstyle | | | |
| Название | Описание | Тип данных | Дополнительно |
| Boardstyle\_ID | Уникальный идентификатор деки. | Int | Первичный ключ |
| Length | Параметр, отвечающий за длину. | Char(20) |  |
| Nose | Наличие у доски переднего конкейва. | Char(20) |  |
| Tale | Наличие у доски заднего конкейва. | Char(20) |  |
| Curved\_tale | Определяет загнут ли задний конкейв у деки. | Char(20) |  |
| Wheelbase | Расстояние между задними траками и передними | Char(20) |  |
| Type\_of\_deck | Тип деки | Char(20) |  |
| Description | Описание деки | varchar(max) |  |

**Таблица Bearings**

Данная таблица описывает виды подшипников.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 5 – Описание таблицы Bearings | | | |
| Название | Описание | Тип данных | Дополнительно |
| Bearings\_ID | Уникальный идентификатор подшипников. | int | Первичный ключ |
| Name | Название подшипников | char(20) |  |
| MadeOf | Материал изготовления |  |  |

**Таблица Skate\_Riding**

Данная таблица служит для присвоения сразу нескольких стилей катания одной доске.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 6 – Описание таблицы Skate\_Riding | | | |
| Название | Описание | Тип данных | Дополнительно |
| Ridingstyle\_ID | Уникальный идентификатор стиля катания. | char(20) | Внешний ключ |
| Skate\_ID | Уникальный идентификатор скейтборда. | int | Внешний ключ |

**Таблица Tracks**

Данная таблица служит для хранения информации о подвесках.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 7 – Описание таблицы Tracks | | | |
| Название | Описание | Тип данных | Дополнительно |
| Tracks\_ID | Уникальный идентификатор подвесок. | int | Первичный ключ |
| Size | Параметр, определяющий размер подвесок | int |  |
| Type | Тип строения подвесок | Char(20) |  |
| Description | Описание модели подвесок | Varchar(max) |  |

**Таблица Wheels**

Данная таблица служит для хранения информации о колесах.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 8 – Описание таблицы Wheels | | | |
| Название | Описание | Тип данных | Дополнительно |
| Wheels\_ID | Уникальный идентификатор колес. | int | Первичный ключ |
| Size | Параметр, определяющий размер колес | Char(4) |  |
| Hardness | Жесткость колес | int |  |
| Description | Описание модели колес | Varchar(max) |  |

**Таблица Questions**

Данная таблица служит для описания вопросов.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 9 – Описание таблицы Questions | | | |
| Название | Описание | Тип данных | Дополнительно |
| Quest\_ID | Уникальный идентификатор вопроса. | char(10) | Первичный ключ |
| Description | Текст вопроса. | Char(1000) |  |

**Таблица Answers**

В данной таблице описываются ответы, а также указывается к какому вопросу они относятся. Также информация на какой следующий вопрос переходить в случае выбора и SQL-запрос на выборку скейтбордов.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 10 – Описание таблицы Wheels | | | |
| Название | Описание | Тип данных | Дополнительно |
| Answer\_ID | Уникальный идентификатор вопроса. | Char(10) | Первичный ключ |
| Quest\_ID | Уникальный идентификатор ответа. | Char(10) | Внешний ключ |
| Text | Текст ответа. | Char(1000) |  |
| Next\_quest\_ID | Здесь указывается ID вопроса, на который надо переходить после выбора данного варианта ответа. | Char(10) |  |
|  |  |  |  |
|  |  |  |  |
| Продолжение таблицы 10 | | | |
| Selection | Sql запрос на выборку скейтборда. | nvarchar(max) |  |

**Хранимые процедуры и представления**

На рисунке 9 изображена хранимая процедура Skates. Функция заключается в возвращении всех параметров доски и компонентов по ее уникальному идентификатору.

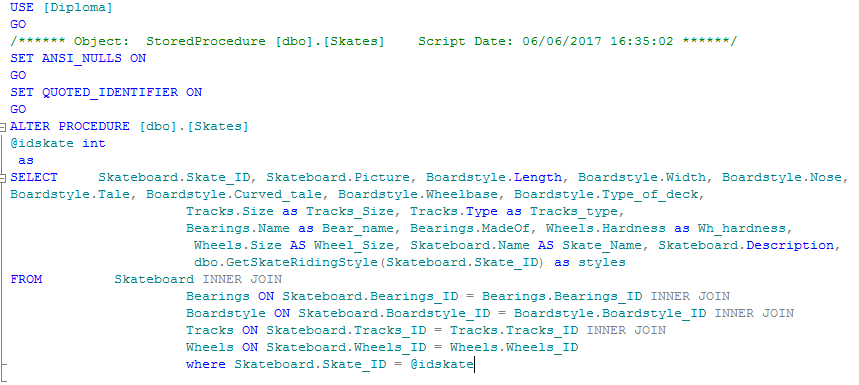


Рисунок 9 – Хранимая процедура Skates.

На рисунке 10 представлена хранимая процедура AddRdStyles, которая используется при добавлении нового скейтборда и предназначена для присваивания ему стилей катания.

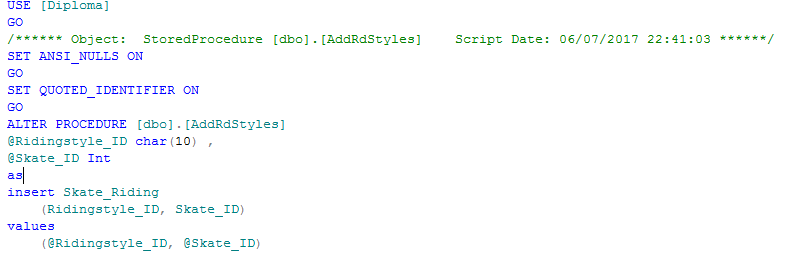


Рисунок 10 – Хранимая процедура AddRdStyles.

На рисунке 11 изображена хранимая процедура AddSkate, которая активируется при добавлении новой доски из программы. Сама определяет максимальный id и добавляет всю внесенную информацию о скейтборде.

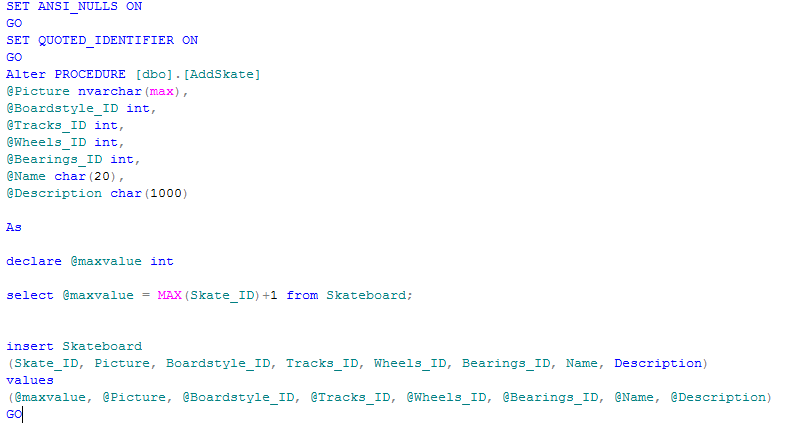


Рисунок 11 - Хранимая процедура AddSkate.

* + 1. **Заполнение базы данных**

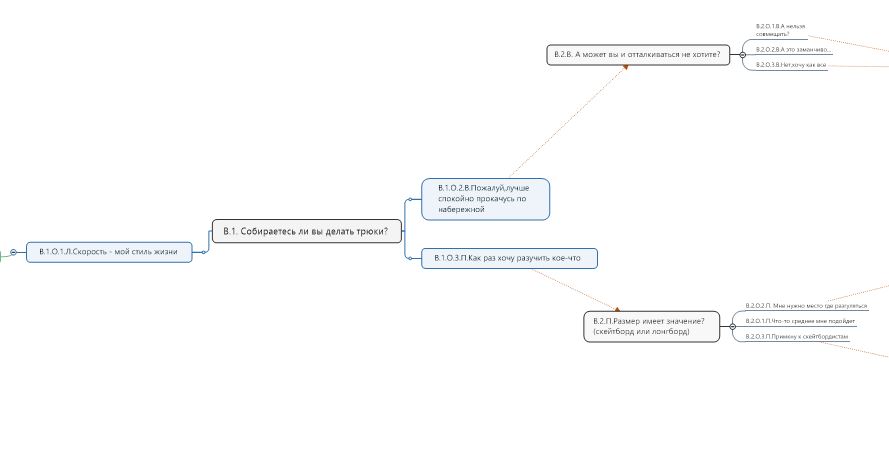
На рисунке 12 изображена UML-диаграмма структуры вопросов-ответов. Здесь представлен небольшой кусок дерева, где у каждого вопроса представлено по три варианта ответа, связанного со следующим вопросом, если он не является конечным.

Рисунок 12 – структура вопросов ответов.

* + 1. **Разработка приложения**

В таблице 11 рассматриваются основные компоненты TSqlQuery, использовавшиеся в ходе разработки приложения, так как через них осуществляются логика программы и связь с базой данных.

|  |  |  |
| --- | --- | --- |
| Таблица 11 – описание компонентов TSqlQuery. | | |
| Название | Описание | Код Sql (забитый в «свойствах» компонента) |
| Qryforid | Через данный компонент получают sql-запрос, предназначенный для дальнейшей выборки скейтбордов из общего количества. Он активируется после того как пользователь нажал на один из вариантов ответа и в запрос передался его id. | Select Selection from dbo.Answers where Answer\_id = |
| Qryquest | Предназначен для загрузки из базы данных вопросов. |  |
| Qrytestskate | Служит для загрузки ответов из базы данных, соответствующих текущему вопросу. |  |
| Продолжение таблицы 11 | | |
| Qryinfo | В данном компоненте содержится основное условие выборки скейтбордов, к которому после выбора пользователем варианта ответа появляются добавочные условия, посредством конструкции «where». Результатом является список досок, подходящих под составленный запрос. | SELECT distinct s1.Name as Skate\_Name, s1.Skate\_ID, s1.Picture, s1.Tracks\_ID, s1.Bearings\_ID, s1.[Description],  s1.Name, s1.Wheels\_ID, s1.Boardstyle\_ID,  b1.Type\_of\_deck  FROM Skateboard s1  INNER JOIN Boardstyle b1 ON s1.Boardstyle\_ID = b1.Boardstyle\_ID  INNER JOIN Skate\_Riding sr1 ON s1.Skate\_ID = sr1.Skate\_ID  INNER JOIN  Riding\_Style rs1 ON rs1.Ridingstyle\_ID = sr1.Ridingstyle\_ID |
| qryonepage | Данный компонент служит для получения данных об одной доске, используя хранимую процедуру «Skates», описанную ранее. |  |

Рассмотрим блок-схемы нескольких алгоритмов работы основных процедур в программе. На рисунке 13 изображен принцип работы процедуры Choosevar, которая в зависимости от выбранного пользователем варианта ответа возвращает выборку скейтбордов.

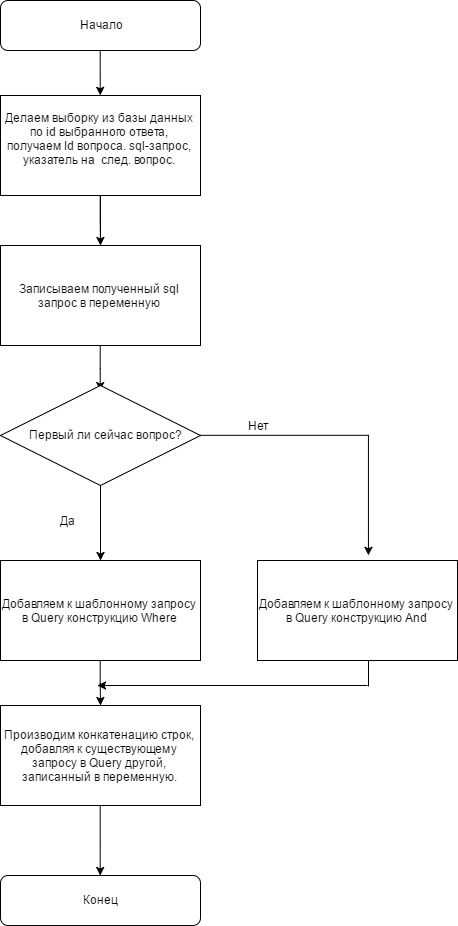


Рисунок 13 – Блок-схема работы процедуры Choosevar.

На рисунке 14 изображена работа функции printnextquest, принцип которой заключается в выводе ответов, соответствующих текущему вопросу.

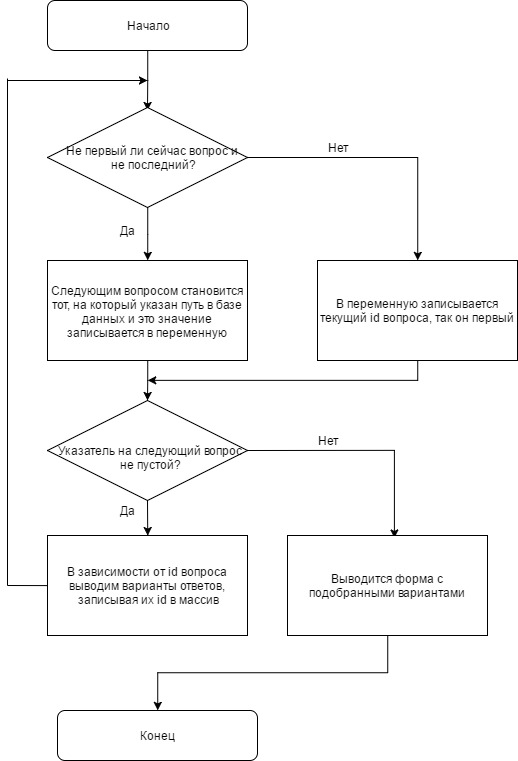


Рисунок 14 – Блок-схема работы функции printnextquest.

* + 1. **Интерфейс приложения**

Основным критериями, при разработке интерфейса были интуитивность и визуальная привлекательность. Так как приложение ориентировано на тех, кто планирует приобрести скейтборд, то это в основном продвинутые современные пользователи, которых можно привлечь интересной графикой или дизайном, поэтому было решено создать, используя инструментальные средства, свое индивидуальное оформление, соответствующее заданным критериям.

Приложение «консультант скейтбордиста» состоит из четырех основных форм. На рисунке 15 изображена первая форма, которая открывается пользователю после запуска приложения. На ней находятся кнопки «Старт», «Wiki» и «Admin».



Рисунок 15 – Стартовая форма программы.

При нажатии на кнопку «Старт» появляются вопросы и ответы, как изображено на рисунке 16.

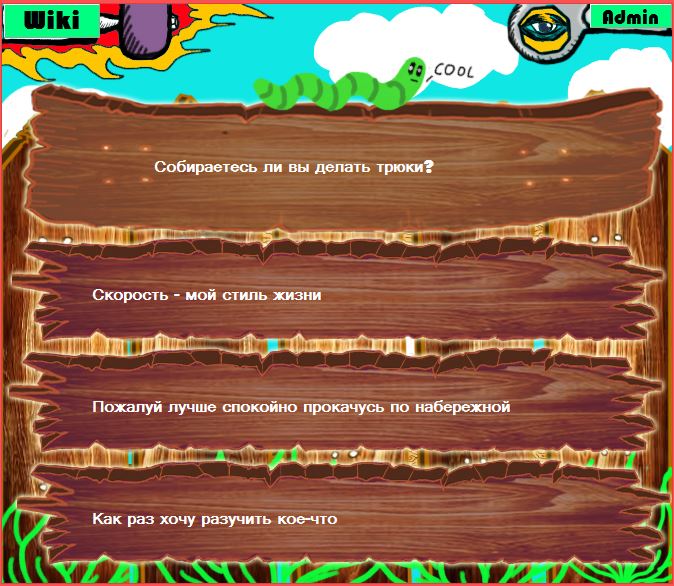


Рисунок 16 – Вывод вопросов и ответов.

Ответив на последний вопрос пользователю откроется вторая форма, как на рисунке 17, на которой будут выведены все результаты выборки. Так же эта форма доступна после нажатия на кнопку «Wiki» на стартовом экране.



Рисунок 17 – Форма вывода результатов выборки.

Если пользователя заинтересовала конкретная модель, то, нажав на кнопку «подробнее», есть возможность рассмотреть данную модель, как представлено на рисунке 18.



Рисунок 18 – Форма вывода одной модели скейтборда.

Так же в программе реализована не только пользовательская часть, но и администратора. Нажав на кнопку «Admin» на стартовом экране и правильно указав пару логин/пароль можно добавить новый скейтборд.

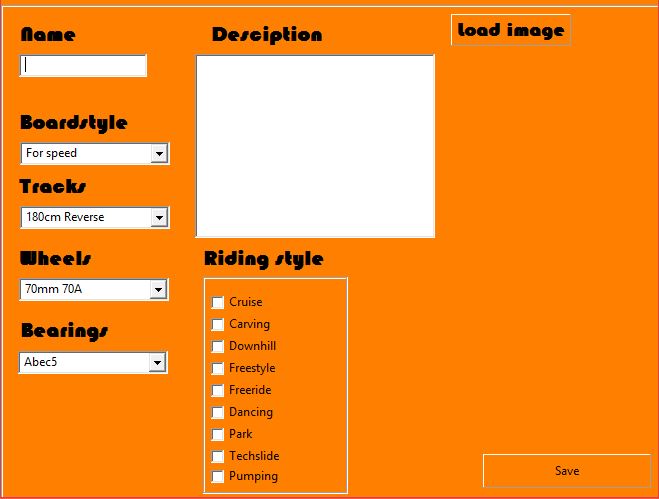


Рисунок 19 – Администраторская часть программы.

**Приложение 1. Иллюстрации к таблице 1**

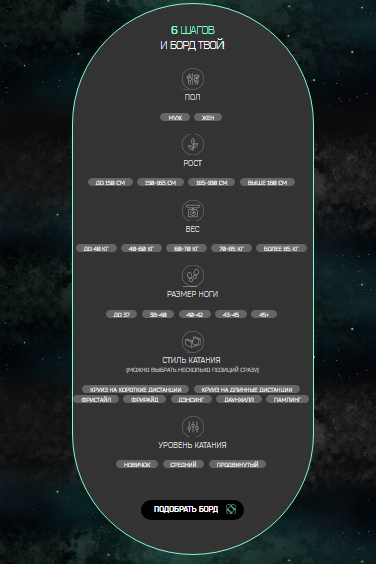


Рисунок 1 – Фильтр товаров на Allaboardshop.

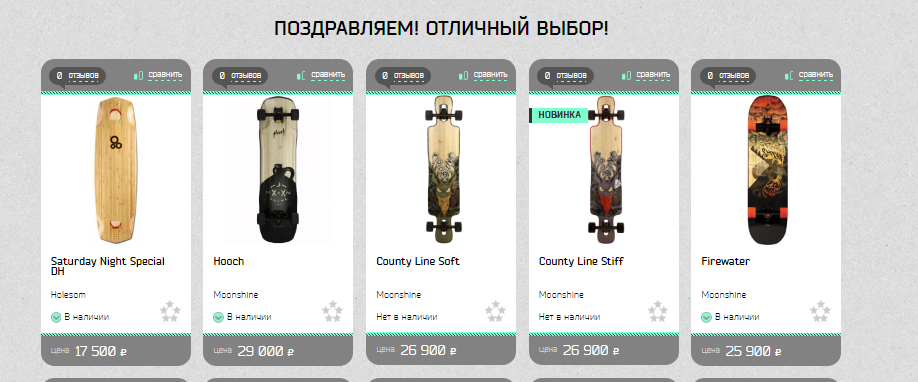


Рисунок 2 – Вывод скейтбордов после фильтрации на Allaboardshop.

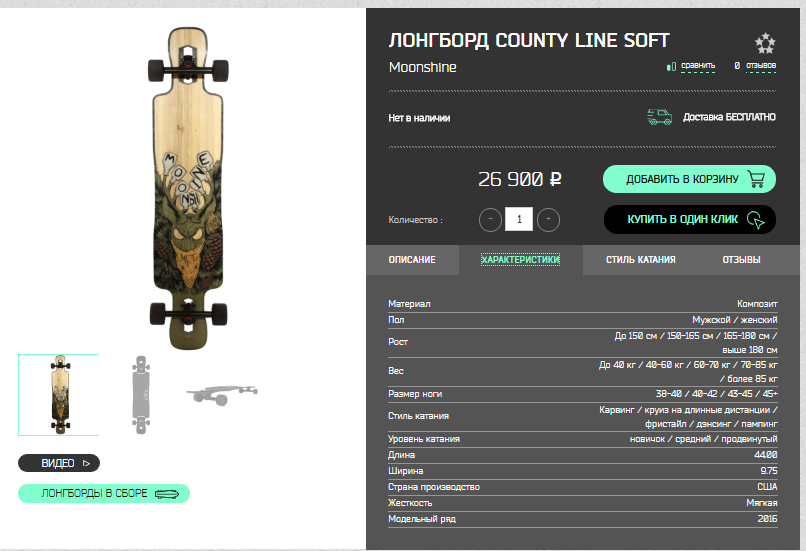


Рисунок 3 – Выбранный скейтборд на Allaboardshop.

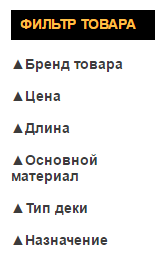
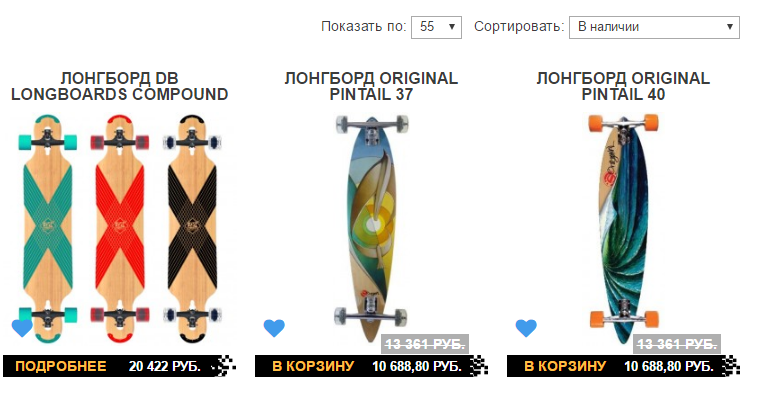


Рисунок 4 – Фильтр товаров на X3mboardshop.

Рисунок 5 – Результат фильтрации на сайте X3mboardshop.

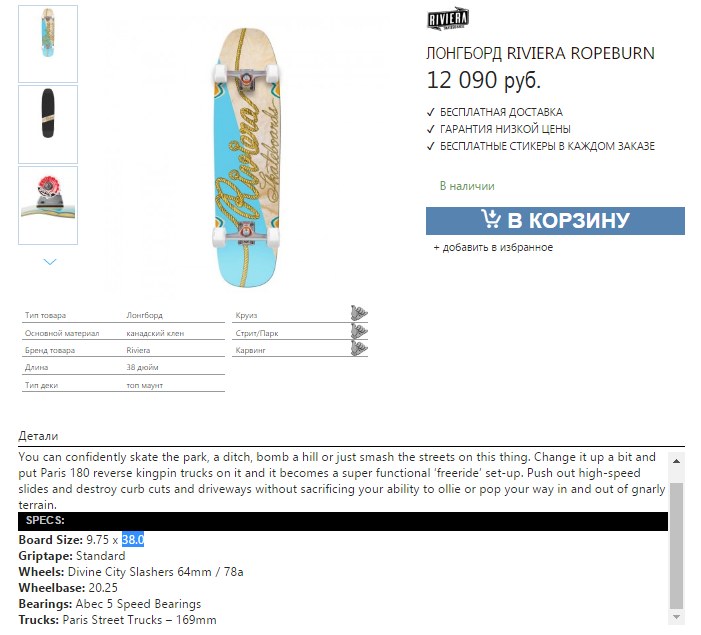


Рисунок 6 – Экран отображения скейтборда на X3mboardshop.

**Приложение 2. Структура вопросов-ответов**