Министерство образования и науки Кыргызской Республики

Кыргызский Государственный Технический университет имени И. Раззакова

Факультет информационных технологий

Кафедра: ПОКС

Дисциплина «Проектирование программного обеспечения 2»

ОТЧЕТ

По выполнению курсового проекта по дисциплине

«Проектирование программного обеспечения 2»

**Выполнил:** студент группы ПИ(б)2-19

**Улан у. Нурдин**

**Проверила:** ст.преподаватель каф. ПОКС

**Мусина Индира Рафиковна**

**Бишкек 2021**

***Содержание***

[***Введение 3***](#_Toc72108830)

[*Цели: 3*](#_Toc72108831)

[*Назначение: 3*](#_Toc72108832)

[***Анализ и разработка требований. 4***](#_Toc72108833)

[*Функциональные требования: 4*](#_Toc72108834)

[*Нефункциональные требования: 4*](#_Toc72108835)

[*Требования к видам обеспечения: 4*](#_Toc72108836)

[***Стадия проектирования 7***](#_Toc72108837)

[*Диаграмма вариантов использования 7*](#_Toc72108838)

[*Не авторизованный клиент 8*](#_Toc72108839)

[*Авторизованный клиент 11*](#_Toc72108840)

[*Модератор 13*](#_Toc72108841)

[*Менеджер 14*](#_Toc72108842)

[*Курьер 14*](#_Toc72108843)

[*Диаграмма деятельности 16*](#_Toc72108844)

[*Модель базы данных 21*](#_Toc72108845)

[*Диаграмма классов 22*](#_Toc72108846)

[*Диаграмма последовательности 23*](#_Toc72108847)

[*Диаграмма состояния 25*](#_Toc72108848)

[***Стадия реализации 26***](#_Toc72108849)

[*Выбор средств реализации 26*](#_Toc72108850)

[*Прототип программы 26*](#_Toc72108851)

[***Заключение 31***](#_Toc72108852)

# Введение

ОАО «Пекарня» хочет создать ресурс для продвижения своего продукта в массы предположительно для стран бывшего СССР: Россия, Украина, Беларусь, Кыргызстан, Казахстан и т.д.

Для пользователей предусмотрена система регистрации и авторизации, а также снятие и пополнение баланса на сайте через различные системы оплаты: VISA, Master Card и т.д.

# Цели:

Помочь организовать все процессы, направленные на:

* приём;
* обработку;
* управление;
* отслеживание заказов.

Среди основных задач, которые позволяет решать такое приложение, стоит выделить:

* получение заказа от потенциального клиента с детальным описанием запроса;
* быстрый вывод на рынок нового товара;
* существенное снижение затрат, необходимых для заключения/завершения сделки;
* существенно сокращение «дороги», которую продукт вынужден пройти от поставщика к клиенту.

# Назначение:

Веб приложение предназначено для:

* Упрощения взаимоотношений с клиентами;
* Возможность производителям и/или поставщикам предлагать свои товары потенциальной аудитории непосредственно в сети;
* Быстрого просмотра каталога.

# Анализ и разработка требований.

# Функциональные требования:

Веб приложение должно иметь следующие функциональные требования:

* Ввод / вывод данных
* Запросы на осуществление транзакций на карту
* Поиск и фильтрация данных
* Регистрация и авторизация пользователя
* Редактирование данных

# Нефункциональные требования:

Веб приложение должно иметь следующие нефункциональные требования:

* Иметь систему рекомендаций;
* Иметь личную корзину пользователя;
* Иметь облачное хранилище для хранения информации в формате .txt и фото;
* Иметь доступный и простой интерфейс пользователя.

# Требования к видам обеспечения:

Требования к информационному обеспечению системы:

Состав, структура и способы организации данных в системе должны быть определены на этапе технического проектирования.

Уровень хранения данных в системе должен быть построен на основе современных реляционных или объектно-реляционных СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Средства СУБД, а также средства используемой операционной системы должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Требования к лингвистическому обеспечению веб приложения:

Все прикладное программное обеспечение веб приложения для организации взаимодействия с пользователем должно использовать не только русский язык.

Требования к информационной системе:

1. Используемое при разработке программное обеспечение, библиотеки программных кодов, СУБД должны иметь широкое распространение, быть общедоступными и использоваться в промышленных масштабах.
2. В качестве основы для разработки будет использоваться платформа Sublime Text 3.
3. В качестве серверной операционной системы будет использоваться ОС семейства Windows NT или Unix(GNU/Linux).
4. В качестве СУБД должна использоваться MySQL версии не ниже 5.

Требования к клиентской части

Клиент должен использовать один из следующих веб браузеров:

* Google Chrome, Mozilla Firefox, Yandex Browser, Opera – 25 версии и выше;
* Internet Explorer 10 версии или выше;
* Safari 5 или выше;

Требования к серверной части

Сервер должен удовлетворять следующим критериям:

* Операционная система семейства Windows NT или Unix (GNU/Linux);
* Кроссплатформенный веб-сервер Apache
* СУБД MySQL 5 и выше.
* Ввод и вывод средств для покупки/продажи внутри системы с помощью популярных систем оплаты VISA, Master card и т.д.

Требования к организационному обеспечению

Организационное обеспечение Веб приложения должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей при осуществлении автоматизированных и связанных с ними неавтоматизированных функций веб приложения.

**Заказчиком** должны быть определены должностные лица, ответственные за:

* Обработку и контроль вводимой информации в систему;
* Администрирование информационной системой;
* Управление работы сотрудников информационной веб приложения.

К работе с системой должны допускаться сотрудники, имеющие навыки работы на персональном компьютере, ознакомленные с правилами эксплуатации и прошедшие обучение работе с системой.

# Стадия проектирования

## ***Диаграмма вариантов использования***

На рис.1 в виде диаграммы прецедентов использования представлена концептуальная модель системы.



Рис 1. Концептуальная модель системы

На диаграмме представлены 5 внешних сущности(актера):

* Не авторизованный клиент (просмотр товаров, поиск и фильтрация товаров)
* Авторизованный клиент (просмотр товаров, поиск и фильтрация товаров, заполнение собственной корзины, оформление заказа)
* Модератор (добавление товаров, изменение товаров)
* Менеджер (формирование отчета, расчет общей прибыли)
* Курьер (получение счет-фактуры)

Ниже в виде таблиц приведено описание вариантов использования.

# Не авторизованный клиент

**Таблица 1. Регистрация**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 1. Регистрация |
| **Основное действующее лицо:** | Не авторизованный пользователь, авторизованный пользователь, модератор |
| **Описание:** | Не авторизованный пользователь, посетивший сайт, отправляет запрос на регистрацию. |
| **Триггер:** | Не авторизованный пользователь желает зарегистрироваться. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Заполнение формы  PRE-3. Отправка формы |
| **Выходные условия:** | POST-1. Сообщение с подтверждением нового аккаунта на номер телефона. |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Регистрация  1. Не авторизованный пользователь заходит на форму регистрации  2. Не авторизованный пользователь заполняет и отправляет форму регистрации  3. Не авторизованный пользователь вводит код на сайте и подтверждает регистрацию.  4. Система сохраняет данные пользователя в БД и отправляет сообщение с кодом, для завершения регистрации. |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Регистрация  A-1. Если не подтверждает регистрацию на сайте в течении 5 минут, то остается не авторизованный пользователь и возвращается в меню регистрации  A-2. Не авторизованный пользователь ввел некорректные данные, и остается не авторизованный пользователь и возвращается в основное меню |
| **Исключения:** | 4.1 E1. Если регистрируемый номер телефона или ник уже занят другими пользователями, то следует отказ регистрации |

**Таблица 2. Авторизация**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 2. Авторизация |
| **Основное действующее лицо:** | Не авторизованный пользователь, авторизованный пользователь, модератор |
| **Описание:** | Не авторизованный пользователь заходит на сайт, отправляет запрос на вход в аккаунт. |
| **Триггер:** | Не авторизованный пользователь желает авторизоваться. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Не авторизованный пользователь должен быть зарегистрированным |
| **Выходные условия:** | POST-1. Сайт перенаправляет пользователя в его личный кабинет. |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Авторизация  1. Пользователь заходит на форму авторизации  2. Пользователь вводит логин и пароль  3. Система проверяет наличие этой записи в базе данных  4. При условии, что данные введены, верно, перенаправляет пользователя в его личный кабинет. |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Авторизация  1. Пользователь заходит на форму авторизации  2. Пользователь вводит логин и пароль  3. Если введенные данные неверные, то сайт выдаёт сообщение об ошибке и предлагает восстановить пароль  4. Перенаправляет на меню авторизации |
| **Исключения:** | 4.1 E1. Если введен неправильный логин, то повторный ввод.  E2. Если введен неправильный пароль, то повторный ввод. |

**Таблица 3. Просмотр каталога**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 3. Просмотр каталога |
| **Основное действующее лицо:** | Не авторизованный клиент, Авторизованный клиент, Модератор |
| **Описание:** | Клиент, посетивший сайт, отправляет запрос на просмотр каталога. |
| **Триггер:** | Клиент запрашивает просмотр каталога |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Проверка остатка на наличие |
| **Выходные условия:** | POST-1. Каталог для просмотра |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Просмотр каталога  1. Клиент запрашивает просмотр каталога  2. Система формирует каталог товаров  2.1 Система проверяет наличие в списке товаров на складе  2.2 Система формирует форму со списком товаров имеющихся на складе  3. Пользователь просматривает каталог |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Просмотр каталога  A-1. 1.Клиент не запрашивает просмотр каталога  2. Система не выполняет никаких действий и находится в состоянии ожидания |
| **Исключения:** | 4.1 E1. Товары отсутствуют на складе  Система выводит сообщение что товаров в настоящее время нет на складе |

**Таблица 4. Поиск товара**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 4. Поиск товара |
| **Основное действующее лицо:** | Не авторизованный клиент, Авторизованный клиент, Модератор |
| **Описание:** | Клиент, посетивший сайт, отправляет запрос на поиск товара. |
| **Триггер:** | Клиент вводит ключевые слова для поиска. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Ввод ключевых слов |
| **Выходные условия:** | POST-1. Товары доступные для просмотра |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Поиск товара  1. На главной странице клиент вводит ключевые слова в текстовое поле (категории, название товара)  2. Система осуществляет поиск в списке товаров и формирует список товаров, соответствующих ключевым словам  3. Система выводит список пользователю |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Поиск товара  1. Клиент не вводит ключевые слова для поиска товара  2. Система не выполняет никаких действий и находится в состоянии ожидания |
| **Исключения:** | 4.1 E1. Клиент вводит ключевое слово в текстовое поле, но товар отсутствует на складе и выводиться сообщение об отсутствии |

**Таблица 5. Фильтрация товара**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 5. Фильтрация товара |
| **Основное действующее лицо:** | Система |
| **Описание:** | Система формирует список согласно заданному пользователем ключу поиска (см. UC4) |
| **Триггер:** | Клиент выбирает фильтр для просмотра. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Выбор фильтра |
| **Выходные условия:** | POST-1. Товары доступные для просмотра |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Фильтрация товара  1. Клиент выбирает нужные фильтры  2. Система формирует список товаров  3. Пользователь просматривает список |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Фильтрация товара  1. Клиент не выбирает фильтр просмотра каталога  2. Система не выполняет никаких действий и находится в состоянии ожидания |
| **Исключения:** | 4.1 E1. Товара нету на складе |

**Таблица 6. Проверка остатка на наличие**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 6. Проверка остатка на наличие |
| **Основное действующее лицо:** | Система |
| **Описание:** | Система проверяет есть ли указанный товар из списка на складе. |
| **Триггер:** | - |
| **Предварительные условия:** | PRE-1. Выбор опции Просмотр каталога  PRE-2. Выбор опции Поиск товара  PRE-3. Выбор опции Добавление в корзину |
| **Выходные условия:** | POST-1. True  POST-2. False |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Проверка на наличие  1. Система сортирует список  2. По ключу находит соответствующий товар  3. Поле количества проверяет на наличие  4. Система вызывает проверку остатка на наличие  5. Проверка выдает True или False |

# Авторизованный клиент

**Таблица 7. Добавление в корзину**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 7. Добавление в корзину |
| **Основное действующее лицо:** | Авторизованный клиент, модератор |
| **Описание:** | Добавление в корзину товара. |
| **Триггер:** | Клиент желает добавить товар в собственную корзину. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Клиента  PRE-3. Выбор товара (кнопка добавления в корзину находится рядом)  PRE-4. Нажатие на кнопку |
| **Выходные условия:** | POST-1. Список товаров собственной корзины |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Добавление в корзину  1. Клиент нажимает на кнопку  2. Система добавляет товар в корзину клиента  3. Клиенту выводиться список товаров из корзины |
| **Альтернативное**  **направление развития**  **варианта использования:** | - |
| **Исключения:** | 4.1 E1. Товар отсутствует на складе и выдается соответствующее сообщение |

**Таблица 8. Подтверждение покупки**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 8. Подтверждение покупки |
| **Основное действующее лицо:** | Авторизованный клиент, модератор |
| **Описание:** | Клиент хочет приобрести товар из корзины товаров. |
| **Триггер:** | Клиент нажимает кнопку для подтверждения покупки из корзины товаров. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Клиента  PRE-3. Добавление товара в корзину |
| **Выходные условия:** | POST-1. Формирование счет – фактуры о приобретении покупки |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Подтверждение покупки  1. Клиент подтверждает покупку нажимая на соответствующую кнопку  2. Система формирует счет – фактуру и отправляет ее курьеру и обновляет количество товаров на складе  3. Клиент просматривает счет – фактуру |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Подтверждение покупки  1. Клиент не подтверждает покупку  2. Система не выполняет никаких действий и находится в состоянии ожидания  3. Через неделю удаляет товар из корзины |
| **Исключения:** | 4.1 E1. Товара нет на складе |

**Таблица 9. Обновление количества товара на складе**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 9. Обновление количества товара на складе |
| **Основное действующее лицо:** | Система |
| **Описание:** | Клиент хочет приобрести товар из корзины товаров. |
| **Триггер:** | Клиент нажимает кнопку для подтверждения покупки из корзины товаров. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Клиента  PRE-3. Добавление товара в корзину  PRE-4. Подтверждение покупки |
| **Выходные условия:** | POST-1. Изменение количества товара на складе |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Обновление количества товара на складе  1. Клиент подтверждает покупку  2. Система обновляет количество товаров на складе |

**Таблица 10. Формирование счет – фактуры**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 10. Формирование счет – фактуры |
| **Основное действующее лицо:** | Система |
| **Описание:** | Клиент хочет приобрести товар из корзины товаров. |
| **Триггер:** | Клиент нажимает кнопку для подтверждения покупки из корзины товаров. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Клиента  PRE-3. Добавление товара в корзину  PRE-4. Подтверждение покупки |
| **Выходные условия:** | POST-1. Формируется счет – фактура и отправляется курьеру |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Формирование счет – фактуры  1. Клиент подтверждает покупку  2. Система формирует счет – фактуру и отправляет ее курьеру |

# Модератор

**Таблица 11. Добавление товара**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 11. Добавление товара |
| **Основное действующее лицо:** | Модератор |
| **Описание:** | Модератор сайта добавляет новый (готовый к продаже) товар на сайт. |
| **Триггер:** | Модератор желает добавить товар для дальнейших продаж. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Модератора  PRE-3. Заполнение формы |
| **Выходные условия:** | POST-1. Товар добавлен на сайте, и готов к продаже. |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Добавление товара на сайте  1. Модератор заходит на личный аккаунт  2. Модератор заходит в форму добавления товара  3. Добавляет товар, готовый к продаже |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Добавление товара на сайте  1. Модератор заходит в личный кабинет  2. Модератор заходит в форму добавления товара  3. Вводит не корректные данные |
| **Исключения:** | 4.1 E1. Новый товар уже есть в списке товаров |

**Таблица 12. Изменение цены товара**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 12. Изменение цены товара |
| **Основное действующее лицо:** | Модератор |
| **Описание:** | Модератор сайта изменяет цену товара. |
| **Триггер:** | Модератор хочет изменить цену товара. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Модератора  PRE-3. Выбрать товар для изменения |
| **Выходные условия:** | POST-1. Товар, с новой изменённой ценой. |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Изменение цены товара  1. Модератор заходит на личный аккаунт  2. Открывает форму для изменения  3. Изменяет цену на товар. |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Изменение цены товара  1. Модератор заходит в личный кабинет  2. Открывает форму для изменения  3. Вводит некорректные данные. |

# Менеджер

**Таблица 13. Формирование отчета по прибыли**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 13. Формирование отчета по прибыли |
| **Основное действующее лицо:** | Менеджер |
| **Описание:** | Менеджер магазина хочет получить отчет по прибыли от проданных товаров. |
| **Триггер:** | Менеджер желает сформировать отчет по прибыли. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Менеджера  PRE-3. Нажатия на кнопку. |
| **Выходные условия:** | POST-1. Отчет по продажам. |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Формирование отчета по прибыли  1. Менеджер нажимает на кнопку.  2. Система формирует отчет по продажам от прибыли.  3. Менеджер просматривает отчет. |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Формирование отчета по прибыли  1. Менеджер не нажимает на кнопку  2. Система не выполняет никаких действий и находится в состоянии ожидания |
| **Исключения:** | 4.1 E1. Продаж нет |

**Таблица 14. Расчет общей прибыли**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 14. Расчет общей прибыли |
| **Основное действующее лицо:** | Система |
| **Описание:** | Система производит расчет общей прибыли по продажам. |
| **Триггер:** | По нажатию кнопки, система производит расчет. |
| **Предварительные условия:** | PRE-1. Вход на сайт под аккаунтом Менеджера  PRE-2. Выбор функции (формирование отчета по продажам) |
| **Выходные условия:** | POST-1. Общая прибыль по продажам |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Расчет общей прибыли  1. Менеджер заходит в личный кабинет  2. Нажимает на кнопку формирование отчета по продажам  3. Система производит расчет общей прибыли по продажам |

# Курьер

**Таблица 15. Получение счет – фактуры**

|  |  |
| --- | --- |
| **Идентификатор и название:** | UC 15. Получение счет – фактуры |
| **Основное действующее лицо:** | Курьер |
| **Описание:** | Курьер, хочет получить счет – фактуру. |
| **Триггер:** | Курьер желает получить счет – фактуру. |
| **Предварительные условия:** | PRE-1. Вход на сайт  PRE-2. Авторизация в виде Курьера  PRE-3. Нажатие на кнопку |
| **Выходные условия:** | POST-1. Счет – фактура |
| **Нормальное направление**  **развития варианта**  **использования:** | 4.0 Получение счет – фактуры  1. Курьер нажимает на кнопку  2. Система распечатывает счет – фактуру  3. Курьер просматривает счет – фактуру |
| **Альтернативное**  **направление развития**  **варианта использования:** | 4.1 Получение счет – фактуры  1. Курьер не нажимает на кнопку  2. Система не выполняет никаких действий и находится в состоянии ожидания |
| **Исключения:** | 4.1 E1. Заказов нет |

## ***Диаграмма деятельности***

На рисунках:

* 2 – регистрация пользователя;
* 3 – авторизация пользователя;
* 4 – просмотр каталога;
* 5 – добавление в корзину;
* 6 – подтверждение покупки;
* 7 – добавление товара;
* 8 – изменение данных о товаре;
* 9 – расчет общей прибыли;
* 10 – просмотр счет – фактуры.

   
 Рис 2. Регистрация Рис 3. Авторизация

 

Рис 4. Просмотр каталога Рис 5. Добавление в корзину

## 

 

Рис 6. Подтверждение покупки Рис 7. Добавление товара

 Рис 8. Изменение данных о товаре Рис 9. Расчет общей прибыли



Рис 10. Просмотр счет – фактуры

## ***Модель базы данных***

На рис. 11 при ведена модель базы данных разрабатываемой системы, которая состоит из следующих таблиц:



Рис 11. Структура БД

## ***Диаграмма классов***

На рис. 12 представлена статическая структура системы в диаграмме классов.



Рис 12. Диаграмма классов

## ***Диаграмма последовательности***

На рисунках:

* 13 – формирование отчета;
* 14 – просмотр каталога;
* 15 – сортировка товаров;
* 16 – поиск товаров;
* 17 – добавление в корзину;



Рис 13. Формирование отчета



Рис 14. Просмотр каталога

 

Рис 15. Сортировка товаров Рис 16. Поиск товаров



Рис 17. добавление в корзину

# Диаграмма состояния



Рис 18. Смена состояния системы в ходе работы

# Стадия реализации

## **Выбор средств реализации**

* Среда разработки – phpStorm (Ultimate version)
* Язык программирования - php, HTML/CSS, JavaScript
* Операционная система серверного компьютера – Windows Server 2015
* Язык интерфейса – русский

## **Прототип программы**

Ссылка на GitHub: [Bakery](https://github.com/Smile-Bonchichi/ComputerHardwareStore-Bakery)

Main Controller

<?php  
  
namespace App\Http\Controllers;  
  
use App\Category;  
use App\Product;  
use Illuminate\Http\Request;  
  
class MainController extends Controller  
{  
 public function index() {  
 $products = Product::*get*();  
 return view('index', compact('products'));  
 }  
  
 public function categories() {  
 $categories = Category::get();  
 return view('categories', compact('categories'));  
 }  
  
 public function category($code) {  
 $category = Category::where('code', $code)->first();  
 return view('category', compact('category'));  
 }  
  
 public function product($category, $product = null) {  
 return view('product', ['product' => $product]);  
 }  
}

Basket Controller

<?php  
  
namespace App\Http\Controllers;  
  
use App\Order;  
use Illuminate\Http\Request;  
  
class BasketController extends Controller  
{  
 public function basket()  
 {  
 $orderId = session('orderId');  
 if (!is\_null($orderId)) {  
 $order = Order::*findOrFail*($orderId);  
 }  
 return view('basket', compact('order'));  
 }  
  
 public function basketPlace()  
 {  
 return view('order');  
 }  
  
 public function basketAdd($productId)  
 {  
 $orderId = session('orderId');  
 if (is\_null($orderId)) {  
 $order = Order::*create*();  
 session(['orderId' => $order->id]);  
 } else {  
 $order = Order::*find*($orderId);  
 }  
 $order->products()->attach($productId);  
  
 return view('basket', compact('order'));  
 }  
}

User Table

<?php  
  
use Illuminate\Support\Facades\Schema;  
use Illuminate\Database\Schema\Blueprint;  
use Illuminate\Database\Migrations\Migration;  
  
class CreateUsersTable extends Migration  
{public function up()  
 {  
 Schema::create('users', function (Blueprint $table) {  
 $table->bigIncrements('id');  
 $table->string('name');  
 $table->string('email')->unique();  
 $table->timestamp('email\_verified\_at')->nullable();  
 $table->string('password');  
 $table->rememberToken();  
 $table->timestamps();  
 });  
 }  
public function down()  
 {  
 Schema::dropIfExists('users');  
 }  
}

Product Table

<?php  
  
use Illuminate\Support\Facades\Schema;  
use Illuminate\Database\Schema\Blueprint;  
use Illuminate\Database\Migrations\Migration;  
  
class CreateProductsTable extends Migration  
{public function up()  
 {  
 Schema::create('products', function (Blueprint $table) {  
 $table->bigIncrements('id');  
 $table->integer('category\_id');  
 $table->string('name');  
 $table->string('code');  
 $table->text('discription')->nullable();  
 $table->text('image')->nullable();  
 $table->double('price')->default(0);  
 $table->timestamps();  
 });  
 }public function down()  
 {  
 Schema::dropIfExists('products');  
 }  
}

Category Table

<?php  
  
use Illuminate\Support\Facades\Schema;  
use Illuminate\Database\Schema\Blueprint;  
use Illuminate\Database\Migrations\Migration;  
  
class CreateCategoriesTable extends Migration  
{public function up()  
 {  
 Schema::create('categories', function (Blueprint $table) {  
 $table->bigIncrements('id');  
 $table->string('name');  
 $table->string('code');  
 $table->text('discription')->nullable();  
 $table->text('image')->nullable();  
 $table->timestamps();  
 });  
 }public function down()  
 {  
 Schema::dropIfExists('categories');  
 }  
}

Order Table

<?php  
  
use Illuminate\Database\Migrations\Migration;  
use Illuminate\Database\Schema\Blueprint;  
use Illuminate\Support\Facades\Schema;  
  
class CreateOrdersTable extends Migration  
{public function up()  
 {  
 Schema::create('orders', function (Blueprint $table) {  
 $table->bigIncrements('id');  
 $table->tinyInteger('status')->default(0);  
 $table->string('name')->nullable();  
 $table->string('phone')->nullable();  
 $table->timestamps();  
 });  
 }public function down()  
 {  
 Schema::dropIfExists('orders');  
 }  
}

OrderProduct Table

<?php  
  
use Illuminate\Database\Migrations\Migration;  
use Illuminate\Database\Schema\Blueprint;  
use Illuminate\Support\Facades\Schema;  
  
class CreateOrderProductTable extends Migration  
{public function up()  
 {  
 Schema::create('order\_product', function (Blueprint $table) {  
 $table->bigIncrements('id');  
 $table->integer('order\_id');  
 $table->integer('product\_id');  
 $table->timestamps();  
 });  
 }public function down()  
 {  
 Schema::dropIfExists('order\_product');  
 }  
}

Master Template

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="utf-8">  
 <meta http-equiv="X-UA-Compatible" content="IE=edge">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
  
 <title>@lang('main.online\_shop'): @yield('title')</title>  
  
 <link href="https://fonts.googleapis.com/css?family=Raleway:100,600" rel="stylesheet" type="text/css">  
 <script src="/js/jquery.min.js"></script>  
 <script src="/js/bootstrap.min.js"></script>  
 <link href="/css/bootstrap.min.css" rel="stylesheet">  
 <link href="/css/starter-template.css" rel="stylesheet">  
</head>  
<body>  
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
 <a class="navbar-brand" href="{{ route('index') }}">Магазин техники: Пекарня</a>  
 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarText"  
 aria-controls="navbarText" aria-expanded="false" aria-label="Toggle navigation"><span  
 class="navbar-toggler-icon"></span></button>  
 <div class="collapse navbar-collapse" id="navbarText">  
 <ul class="navbar-nav mr-auto">  
 <li class="nav-item active"><a class="nav-link" href="{{ route('index') }}">Все товары <span  
 class="sr-only">(current)</span></a></li>  
 <li class="nav-item"><a class="nav-link" href="{{ route('categories') }}">Категории</a></li>  
 <li class="nav-item"><a class="nav-link" href="{{ route('basket') }}">В корзину</a></li>  
 </ul>  
 <span class="navbar-text"> Улан у. Нурдин и Асанов Руслан </span>  
 </div>  
</nav>  
  
<div class="container">  
 @yield('content')  
</div>  
  
</body>  
</html>

Product Template

@extends('master')  
  
@section('content')  
  
 <div class="starter-template">  
 <h1>iPhone X 64GB</h1>  
 <h2>{{ $product }}</h2>  
 <p>Цена: <b>71990 Сом</b></p>  
 <img src="http://internet-shop.tmweb.ru/storage/products/iphone\_x.jpg">  
 <p>Отличный продвинутый телефон с памятью на 64 gb</p>  
 <form action="" method="POST">  
 <button type="submit" class="btn btn-success" role="button">Добавить в корзину</button>  
 <input type="hidden" name="\_token" value="olbqlmFBdMvBczBNUUFBgUAtFSkKaz8yXLUWAVWy">  
 </form>  
 </div>  
  
@endsection

Card

<div class="col-sm-6 col-md-4">  
 <div class="thumbnail">  
 <img src="http://internet-shop.tmweb.ru/storage/products/iphone\_x.jpg" alt="iPhone X 64GB">  
 <div class="caption">  
 <h3>{{ $product->name }}</h3>  
 <p>{{ $product->price }} сом.</p>  
 <p>  
 <form action="{{ route('basket-add', $product->id) }}" method="POST">  
 <button type="submit" class="btn btn-primary" role="button">В корзину</button>  
 <a href="{{ route('product', [$product->category->code, $product->code]) }}" class="btn btn-default"  
 role="button">Подробнее</a>  
 @csrf  
 </form>  
 </p>  
 </div>  
 </div>  
</div>

# Заключение

В ходе работы над проектом достигнуты следующие результаты:

* Изучена предметная область;
* Выявлены проблемы;
* Проведено проектирование;
* Выбраны средства реализации;
* Разработан прототип программы.