Большакова В.А. ИСТ-111

**Этап №3**

1. **Слой пользовательского интерфейса**

Веб-интерфейс — это совокупность средств, при помощи которых пользователь взаимодействует с веб-сайтом.

Схема страниц сайта с переходами представлена на рисунке 1.



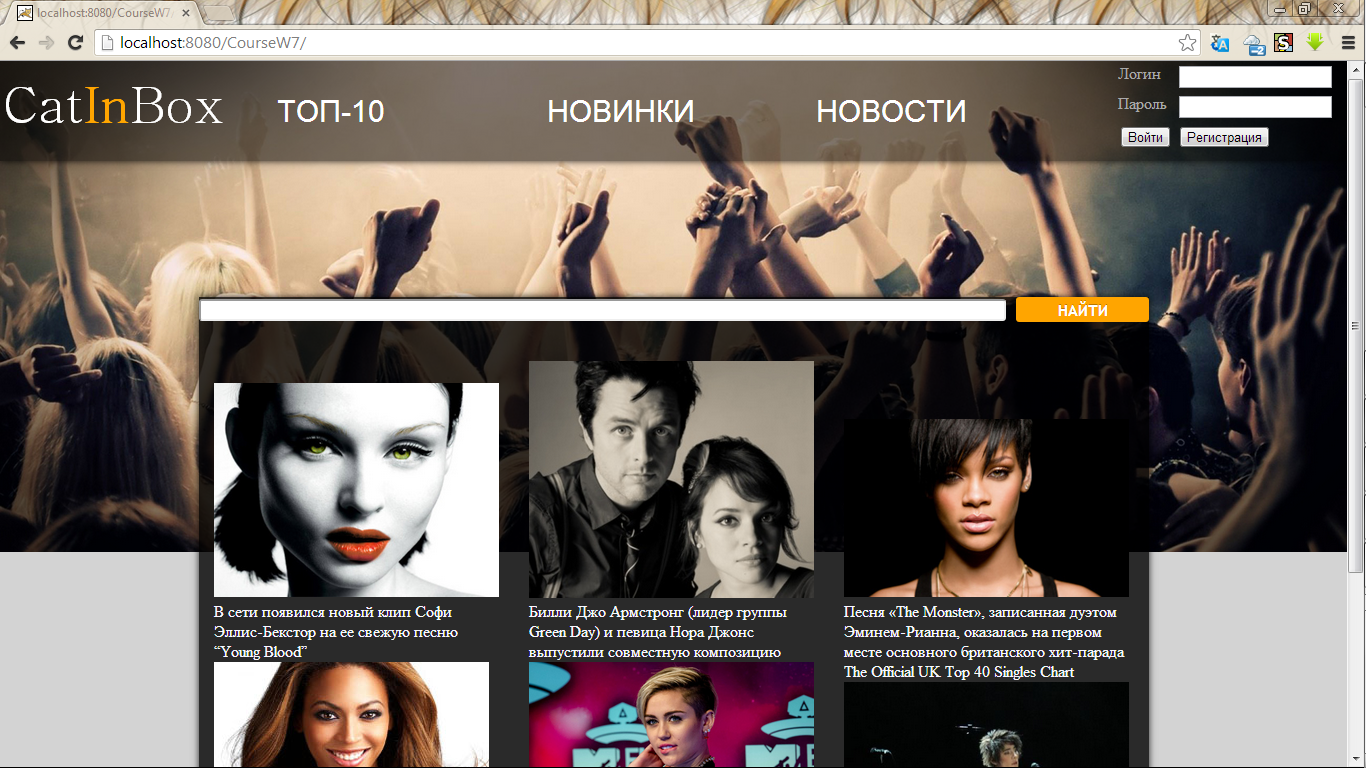
*Рисунок 1. Схема сайта*

В моей системе могут работать несколько видов пользователей, которые имеют разные права доступа: администратор, зарегистрированный пользователь, незарегистрированный пользователь. Переход на ту или иную страницу зависит от пользователя, так, например, администратор имеет доступ ко всем существующим страницам, а незарегистрированный пользователь не имеет доступ к странице платежа.

Каждая страница имеет одинаковую шапку, содержащую форму входа (если пользователь еще не авторизован), ссылки на основные разделы сайта и на форму регистрации.

Примерное содержание каждой страницы:

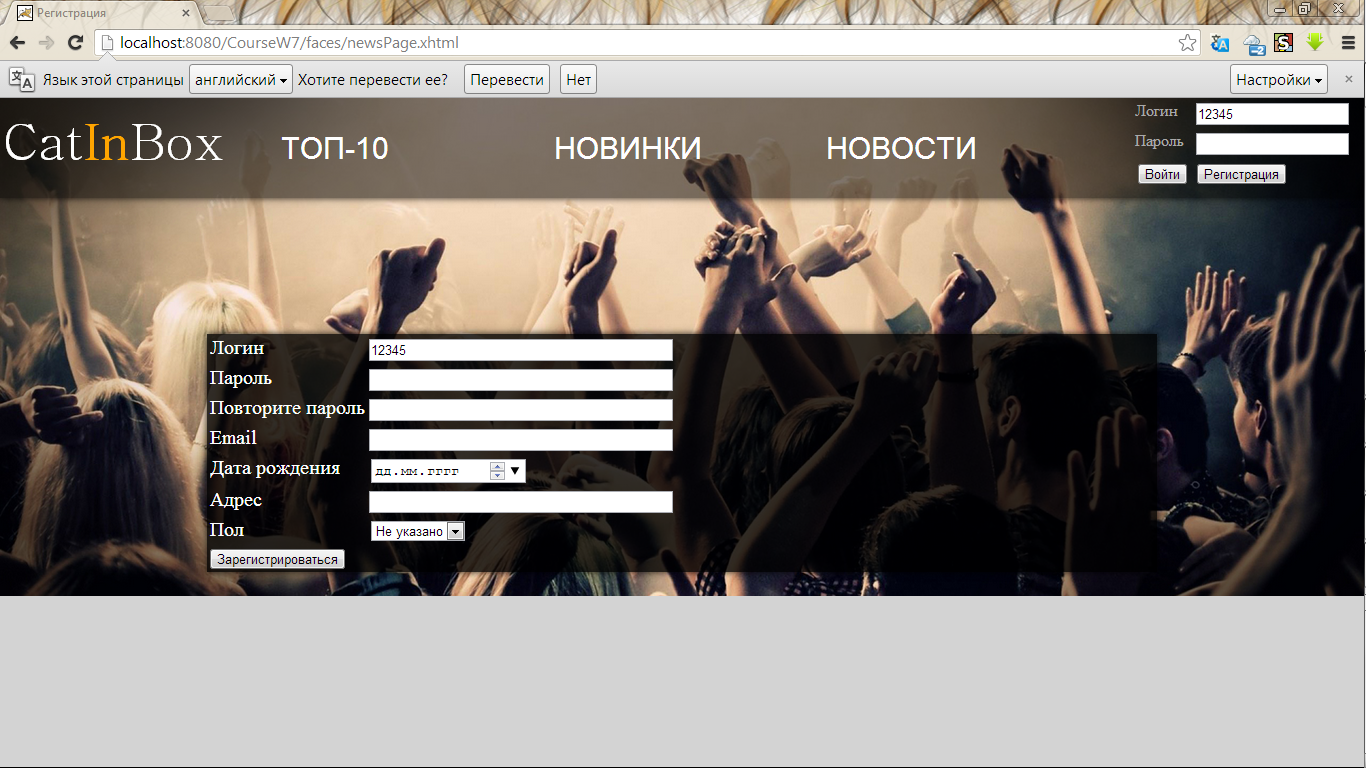
**index.xhtml** – начальная страница, на которой отображаются свежие новости. Доступна всем пользователям. Предположительно, будет содержать строку поиска (если не будет найдено более удачного расположения для нее). Первоначальный вариант index.html представлен на рисунке 2.



*Рисунок 2. Макет страницы index.html*

**registration.xhtml –** страница регистрации, доступна всем пользователям. Содержит обязательные и необязательные поля, которые заполняет пользователь: Логин\*, Пароль\*, Подтверждение пароля\*, Email\*, Имя, Город, Пол, Дополнительная информация, Номер карты, CVC-код(если есть номер карты). Регистрация происходит после нажатия кнопки Зарегистрироваться. Если одно или несколько обязательных полей не были заполнены или были заполнены неправильно, пользователю предлагается еще раз заполнить необходимые поля.

Макет страницы регистрации представлен на рисунке 3.



*Рисунок 3. Макет страницы registration.xhtml*

**profile.xhtml** – страница зарегистрированного пользователя. На ней отображаются регистрационные данные пользователя, которые он же или администратор может поменять. Если данные пользователя были изменены, необходимо подтвердить изменения, нажав на кнопку ниже.

**news.xhtml –** страница, на которых отображены все новости сайта. Отсюда администратор может попасть на страницу добавления новости. Обычный пользователь может посмотреть подробнее о любой новости.

**addnews.xhmtl** – страница добавления новости. Содержит поля: Заголовок, Новость, загрузка картинки, загрузка видео. Создание новости происходит после нажатия кнопки. Только что созданная новость автоматически добавляется на главную страницу

**newsname.xhtml** – страница конкретной новости. Название страницы будет соответствовать заголовку новости. Здесь также можно оставлять комментарии администраторам и зарегистрированным пользователям.

**upload.xtml** – страница добавления новой песни. Имеет поля Название, Исполнитель, Альбом, загрузка песни, Жанр, Год. Добавление песни происходит после нажатия кнопки. Если одно из полей было заполнено неверно, то пользователю предлагается подкорректировать данные. Только зарегистрированные пользователи и администраторы имеют доступ к этой странице.

**songname,xht**ml – страница песни. На ней располагается обложка песни, а рядом информация о песне: Название, Группа, Альбом, Жанр, Битрейт, имя добавившего пользователя, рейтинг. Ниже располагаются кнопки Скачать и Купить. Ниже располагается музыкальный плеер и youtube плеер с клипом (если есть). Внизу располагается форма для добавления комментария, а еще ниже список комментариев. Комментарии могут оставлять только администраторы и авторизованные пользователи.

**groupname.xhtml** – страница с информацией о группе. Содержит фотографию группы (если загружена), название группы, жанр, список участников и краткую информацию о группе.

**payment.xhtml** – страница покупки песни. Доступна зарегистрированным пользователям и администраторам. Здесь необходимо ввести пароль пользователя. На данной странице содержится информация о покупках. Ниже располагается кнопка Отправить.

**confirm.xhtml** – подтверждение прохождения/отклонения платежа. Содержит надпись «Оплата успешно произведена» или «Извините, платеж отклонен». Ниже имеется кнопка Ок.

1. **Слой бизнес-логики**

Слой бизнес-логики содержит все классы, ответственные за функционирование сайта. Они являются посредниками между пользовательским интерфейсом и бд.

Ниже представлены основные классы и их методы

**UserBean**

*logIn* – метод, необходимый для входа пользователя в систему. Проверяет совпадение логина и пароля

*logout* – метод для выхода из системы

*registrate* – метод для регистрации пользователя. После того, как пользователь успешно заполнил на форме все поля, данные передаются в этот метод, который заносит их в бд

*editPrifile* – метод для изменения регистрационных данных пользователя. Пользователь может редактировать свои регистрационные данные на странице profile.xhtml Эти данные передаются в этот метод, а потом передаются в бд

*addnews* – метод для добавления новости. После того, как администратор заполнил все поля на странице addnews.xhtml, они передаются в этот метод, а потом заносятся в бд

*deletenews* – метод для удаления новости

**AdminBean**

*banUser* – метод для блокирования пользователя. Если у пользователя есть нарушения, он блокируется.

*unbanUser* – разблокировка пользователя

**SongBean**

*uploadSong* – метод для добавления пользователя. Данные из формы для добавления новой песни передаются в этот метод, который вносит соответствующие записи в бд

*downloadSong* – метод для скачивания песни

*addComment* – метод для добавления песни

*deleteComment* – метод для удаления своего комментария. Удаляется соответствующая строчка из бд.

*buySong* – метод для покупки песни. Этот метод использует распределенные транзакции

*checkBankAccount* – метод для проверки состояния банковского счета пользователя.

Диаграмма последовательностей метода buySong представлена на рисунке 4



*Рисунок 4. Диаграмма последовательностей*

Диаграмма деятельности для бизнес-процесса «Покупка песни» представлена на рисунке 5



*Рисунок 5. Диаграмма деятельности*

1. **Слой доступа к данным**

**Песня - Song**

Данный класс представляет сущность, содержащая поля, в которые будут заноситься данные о песне, после чего они будут записаны в соответствующую таблицу Song. Имеет поля:

* Id:int – уникальный идентификационный номер песни
* name:String – название песни
* year:String – год выпуска песни
* genre:String – жанр песни
* bitrait:int - битрейт
* length:long – длина песни (сек)
* size:float – размер песни
* rait:float – рейтинг песни
* Url:String в бд хранятся только ссылки на mp3

**Группа**

Данный класс представляет сущность содержит данные об исполнителе, которые будут записаны в таблицу Band. Имеет поля:

* Id:int
* name:String – название группы
* genre:String - жанр
* discription:String – описание группы
* members:String – члены группы

**Альбом**

Данный класс представляет сущность, которая содержит информацию об альбомах выбранной группы. Информация оттуда будет записана в таблицу Album. Имеет поля:

* id
* name:String – название альбома
* genre:String - жанр
* length:long – длительность всего альбома (сек)
* year:String – год выпуска альбома

**Пользователь**

Данный класс представляет сущность содержащую поля, в которые будут заноситься данные о зарегистрированных пользователях, а затем записываться в БД. Имеет поля:

* id:int
* login:String
* email:String
* password:String
* address:String - адрес
* regDate:Date – дата регистрации
* birthDate:Date – дата рожден**я**
* name:String – имя пользователя

**Комментарий**

Данный класс представляет сущность, которая содержит все комментарии к песням. Имеет поля:

* id:int
* postText:String – текст комментария
* postDate:Date – дата, когда был оставлен комментарий

**Новость**

Данный класс представляет сущность, которая содержит новости и видеоматериалы к ним. Имеет поля:

* id:int
* newsText:String – текст новости
* Urlvideo:String – ссылка на видео

**Карты**

Содержит информацию о банковских картах, привязанных к аккаунту. Имеет поля:

* Id:int
* cardnum:String – номер карты
* valDate:Date - срок действия карты
* address:String
* CVC:String – CVC-код

**Транзакции**

Содержит информацию обо всех покупках, совершенных данным пользователем. Имеет поля:

* Id:int
* trDate:Date – дата транзакции
* status:boolean – статус платежа (Выполнен/Не выполнен)
* amount:String – сумма покупки
* name:String – название транзакции

**Бонусы**

Некоторые сайты предоставляют различные скидки, которые потом можно использовать. Содержит поля:

* Id:int
* name:String – название бонуса
* discount:int – размер скидки
* discountSite:String – источник бонуса