**Техническое задание к продукту**

**Автоматизированная система реализации билетов и абонементов в кинотеатр**

Куличок Никита

Дата создания: 3 декабря 2019

Версия 0.1, 3 декабря 2019

1. **Общие положения**
   1. **Команда разработчиков и роли**

Куличок Никита – full-stack developer

* 1. **Плановые сроки окончания работ**

Проект планируется сдать до 20 декабря 2019 года.

1. **Предназначения и формулировка целей**
   1. **Целевая аудитория**

Любые пользователи сети Интернет

* 1. **Предназначение приложения**

Создать информационное приложение, представляющее собой блогхост и помогающее в распространении различного рода информации в виде блогов, статей и комментариев к ним.

* 1. **Цели**

Главные цели данного документа следующие:

1. Установка плана работы
2. Распределение ролей внутри команды
3. Постановка задач
   1. **Риски**

Создание данного приложения может быть связано со следующими рисками:

1. Риск произвести ошибочную оценку сложности проекта и превысить ожидаемые сроки выполнения задач
2. Риск не учесть особенности предметной области
3. Риск создать неконкурентноспособный продукт
4. **Функциональные требования**
   1. **Общие требования к системе**

Приложение должно соответствовать следующим требованиям:

* создание блогов, статей блога и их последующее редактирование;
* возможность регистрации и авторизации пользователя, при этом нельзя зарегистрировать несколько аккаунтов на один email;
* иметь дружественный, интуитивно понятный интерфейс;
* редактирование личных пользовательских данных;
* управление списком пользователей через интерфейс администратора;
* модерирование статей и комментариев;
* поиск по тэгам, тексту;
* обмен сообщениями между пользователями;
* возможность оставить оценку «мне нравится» для статей;
* комментирование статей (добавление новых комментариев и их обновление должно происходить без перезагрузки страницы);

ограничение действий для не аутентифицированных пользователей.

* 1. **Доступные функции**
     1. **Поиск блога по названию**

Пользователь должен иметь возможность поиска блогов по набору слов, входящих в название.

Действия:

* Нажать на пункт «Search», находящийся в шапке сайта;
* В появившемся списке выбрать пункт «Blogs»;
* На открывшейся странице сайта ввести поисковый запрос в соответствующую форму ввода.
* Нажать кнопку «Search»;
* Просмотреть результаты поиска и выбрать нужный блог.
  + 1. **Поиск статьи по названию**

Пользователь должен иметь возможность поиска статей по набору слов, входящих в название.

Действия:

* Нажать на пункт «Search», находящийся в шапке сайта;
* В появившемся списке выбрать пункт «Posts»;
* На открывшейся странице сайта ввести поисковый запрос в соответствующую форму ввода.
* Нажать кнопку «Search»;
* Просмотреть результаты поиска и выбрать нужную статью.
  + 1. **Открытие статьи для просмотра**

Пользователь должен иметь возможность просмотра как собственных статей, так и статей других пользователей.

Действия:

* Нажать на заголовок выбранной статьи;
* Система отображает статью с необходимой информацией поста;
  + 1. **Открытие блогов для просмотра**

Пользователь должен иметь возможность просмотра как собственных блогов, так и блогов других пользователей.

Действия:

* Получить список блогов с помощью поиска;
* Нажать на заголовок выбранного блога;
* Система отображает список статей блога;
  + 1. **Создание статьи**

Пользователь должен иметь возможность создавать новые статьи для их последующего просмотра другими пользователями.

Действия:

* Нажать на кнопку «Add new post» у соответствующего

блога;

* На появившейся форме заполнить данные данные нового поста;
* Нажать кнопку «Save»;
* Система сохраняет всю введенную информацию в базе данных.
  + 1. **Редактирование статьи**

Пользователь должен иметь возможность редактировать основную информацию статей.

Действия:

* Нажать на кнопку «Edit» у соответствующей статьи;
* На появившейся форме отредактировать данные существующего поста;
* Нажать кнопку «Save»;
* Система сохраняет всю введенную информацию в базе данных.
  + 1. **Удаление статьи**

Пользователь должен иметь возможность удаления статей.

Действия:

* Нажать на кнопку «Delete» у соответствующей статьи;
* Система удаляет информацию о статье из базы данных.
  + 1. **Создание блога**

Пользователь должен иметь возможность создавать новые блоги, для их последующего просмотра другими пользователями.

Действия:

* Нажать на пункт «My blogs», находящийся в шапке

сайта;

* В появившемся списке выбрать пункт «Add new»;
* На появившейся форме заполнить данные нового блога;
* Нажать кнопку «Save»;
* Система сохраняет всю введенную информацию в базе данных.
  + 1. **Редактирование блога**

Пользователь должен иметь возможность редактировать основную информацию блогов.

Действия:

* Нажать на кнопку «Edit» у соответствующего блога;
* На появившейся форме отредактировать данные существующего блога;
* Нажать кнопку «Save»;
* Система сохраняет всю введенную информацию в базе данных.
  + 1. **Удаление блога**

Пользователь должен иметь возможность удаления статей.

Основной поток действий:

* Нажать на кнопку «Delete» у соответствующего блога;
* Система удаляет информацию о блоге из базы данных.
  + 1. **Поставить лайк**

Пользователь должен иметь возможность оценить понравившуюся статью с помощью «лайка».

Действия:

* Нажать на значок в форме сердца;
* Система отображает поставленный лайк путем изменения цвета значка с серого на красный и сохраняет информацию в базе данных (сохранение и отображение происходит без перезагрузки страницы).
  + 1. **Написание комменариев**

Пользователь должен иметь возможность прокомментировать любую из статей.

Действия:

* Ввести текст комментария в форму ввода находящуюся под основной информацией статьи;
* Нажать на кнопку «Comment»;
* Система отображает новый комментарий и сохраняет информацию о нем в базу данных (сохранение и отображение происходит без перезагрузки страницы).
  1. **Система ролей**

Рассмотрим подробнее возможности пользователей веб-приложения. Всех пользователей можно разделить на 4 категории, дополняющие друг друга:

* Неавторизованный пользователь – пользователь с наименьшим количеством возможностей, однако имеющий возможность пользоваться достаточным набором информации. Доступные операции: просмотр всех блогов, постов и комментариев к ним.
* Обычный аторизованный пользователь – пользователь, имеющий доступ ко всем операциям неавторизованного пользователя, плюс следующие операции: возможность создания, редактирования и удаления своих постов, блогов и комментариев, возможность поставить и убрать оценку «Мне нравится» для поста.
* Модератор – пользователь, имеющий доступ ко всем операциям обычного авторизованного пользователя, плюс следующие операции: редактирование и удаление своих постов, блогов и комментариев любых пользователей. Стоит также отметить, что модератор не может снять с себя предоставленные ему «особые» полномочия. Он может сделать это лишь обратившись к администротору.
* Администратор – пользователь, имеющий доступ ко всем операциям обычного модератора, плюс следующие операции: просмотр, редактирование
* удаление личных данных любых пользователей, назначение и удаление прав «модератор» и «администратор».

1. **Нефункциональные требования**
   1. **Интерфейс пользователя**

Требования к пользовательскому интерфейсу:

* + Интерфейс пользователя должен быть мультиязычным, в начале два языка: русский и английский.
  + Система должна отображать корректно интерфейс пользователя
* разрешением от 1024x600 пикселей.
  1. **Отображение на различных ОС**

Приложение должно быть кроссплатформенным, в том числе, с возможностью запуска серверной части на различных ОС.

* 1. **Документация**

Документация должна соответствовать следующим требованиям:

 Разрабатываемые программные модули должны быть самодокументированы, т. е. тексты программ должны содержать все необходимые комментарии;

* В состав сопровождающей документации должна входить UML-диаграммы.
  1. **Порядок разработки**

Требования к процессу разработки программного продукта:

 Вся разработка должна сопровождаться гранулярными коммитами (на английском языке) в системе GIT. Коммиты должны быть понятны и содержать лишь основную суть изменений кода программы. При разработке должны быть активно использованы соответствующие ветки (весь процесс разработки не может производиться только в ветке master);

 Разрабатываемая программа должна соответствовать руководству по стилю кода.

1. **Перспективы развития**

* дальнейшем планируется улучшение интерфейса web-приложения, увеличение скорости передачи данных на серверную часть, повышение удобства использования, улучшение системы ролей, повышение вовлеченности пользователей и привлекательности сайта.