

*Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования*

*«Московский государственный технический университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)*

Система публикации блогов

Студент: Попутников И.В. гр. ИУ7-22М

Глоссарий

Термин	Определение
Блог	(англ. blog, от web log — интернет-журнал событий, интернет-дневник, онлайн-дневник) — веб-сайт, основное содержимое которого — регулярно добавляемые записи, содержащие текст, изображения или мультимедиа.
Модерирование	Модификация или удаление опубликованной на портале информации с целью соблюдения установленных правил
Валидация данных	Проверка на корректность, полноту и непротиворечивость входных, выходных и обрабатываемых данных
WEB-сервис	идентифицируемая веб-адресом программная система со стандартизированными интерфейсами.
Авторизация	предоставление определённому лицу или группе лиц прав на выполнение определённых действий
Аутентификация	процедура проверки легальности пользователя
Валидация данных	Проверка на корректность, полноту и непротиворечивость входных, выходных и обрабатываемых данных
COA (SOA)	Сервис-ориентированная архитектура (Service Oriented Architecture), модульный подход к разработке программного обеспечения, основанный на использовании распределённых, слабо связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам.
Регистрация	Процедура создания записи пользователя, необходимая для последующей его аутентификации и авторизации
WEB-интерфейс	Интерфейс пользователя, предоставляемой системой через Web-браузер

Принятые сокращения

Сокращение	Расшифровка
ОС	Операционная система
ПО	Программное обеспечение

Введение

Данное техническое задание составляется для проектирования ПО «Система публикации блогов». Техническое задание выполняется в соответствии со стандартом ГОСТ 19.201-78 «ЕСПД. Техническое задание, требования к содержанию и оформлению».

Краткое описание предметной области

Появление сети Интернет позволило рядовым пользователям публиковать свои заметки и мнения обо всём, что происходит вокруг: выход нового фильма, обзор

прочитанной книги или вовсе размышления о жизни. Также большой интерес вызывает получение обратной связи через комментирование публикаций и выставление оценок. Таким образом большой поток записей требует механизмов хранения как самих статей, так и комментариев к ним, а также способы отсеивания бесполезных, плохих и даже оскорбительных заметок.

Существующие аналоги

В качестве аналогов проектируемого программного обеспечения можно выделить следующие программные решения «Живой Журнал», блоги «StopGame». В большинстве систем, в частности в системе онлайн-дневников «Живой Журнал», оценивание весьма примитивно: можно пометить материал как хороший или не пометить. На других платформах оценивание происходит по некоторой шкале (пятибалльной, например). Несколько сложнее система рейтинга устроена на портале «StopGame». Пользователь может дать материалу оценку +1 или -1 (все оценки складываются, формируя общий рейтинг публикации). При этом оценка влияет на рейтинг самого автора с учётом веса оценивающего пользователя. Тем самым оценки признанных участников сообщества весомее, а авторы плохих публикаций получают отрицательный рейтинг и лишаются права оставлять заметки в дальнейшем. Однако и эта система не лишена недостатков. Зачастую новички быстро получают плохие отметки от старожилов и теряют как возможность создавать новые материалы, так и желание.

Описание системы

Главное назначение ПО «Система публикации блогов» – предоставление пользователю возможности публиковать свои блоги и читать публикации других авторов. Также требуется разработать и реализовать систему оценивания материалов и их авторов, с целью пресечения появления плохих, бесполезных и оскорбительных публикаций.

Назначение разработки

ПО «Система публикации блогов» предназначена публикации заметок пользователей, их оценивания и комментирования.

Требования к системе

1. Разрабатываемое программное обеспечение должно обеспечивать функционирование системы в режиме 24/7/365 со среднегодовым временем доступности не менее 99.9%.
2. Время восстановления системы после сбоя не должно превышать 20 минут.
3. Система должна поддерживать возможность «горячего» переконфигурирования системы. Необходимо поддержать возможность добавления нового узла во время работы системы без рестарта.

Требования к функциональным характеристикам

1. По результатам работы модуля сбора статистики медиана времени отклика системы на запросы, добавляющие или изменяющие информацию на портале не должна превышать 10 секунд без учета латентности географического расположения узла.
2. Портал должен обеспечивать возможность запуска в современных настольных браузерах: не менее 75% пользователей Интернета должны иметь возможность пользоваться порталом без какой-либо деградации функционала.

Требования по реализации

1. Пароли учётных записей должны подвергаться хешированию.
2. Система должна предоставлять как минимум два интерфейса – интерфейс пользователя и администратора.
3. Для хранения данных о пользователях, публикациях, комментариях использовать СУБД MySQL.

Функциональные требования к системе с точки зрения пользователя

ПО должна обеспечивать реализацию следующих функций:

1. Система должна обеспечивать регистрацию и авторизацию пользователей с учётной записью google.com.
2. Система должна обеспечивать аутентификацию пользователей.
3. Система должна обеспечивать разделение пользователей на три роли:
 - администратор;
 - пользователь;

- гость;
4. Система должна предоставлять **администратору** следующие функции:
 - изменение публикаций;
 - удаление публикаций;
 - изменение информации о пользователе;
 - блокировка учетной записи пользователя за строгие нарушения;
 - удаления пользователей;
 - просмотр отчетов статистики.
 5. Система должна предоставлять **пользователю** следующие функции:
 - просмотр публикаций;
 - создание публикаций;
 - изменение своих публикаций;
 - удаление своих публикаций;
 - комментирование и оценка публикаций.
 6. Система должна предоставлять **гостю** следующие функции:
 - просмотр публикаций;

Входные параметры системы

1. Система должна содержать следующую информацию о пользователях:
 - учётное имя пользователя в системе (необходимо для идентификации);
 - пароль в хешированном виде по алгоритму SHA256;
 - тип пользователя, который может быть следующим: администратор, пользователь;
 - имя пользователя;
 - фамилия пользователя;
 - краткая информация о пользователе;
 - рейтинг пользователя;
2. Система должна содержать следующую информацию о публикациях:
 - название публикации;
 - текст публикации;
 - ключевые слова для поиска;
 - логотип публикации (титольное изображение);
 - дата публикации;

- рейтинг публикации;
3. Система должна содержать следующую информацию о комментариях:
- идентификатор комментируемого блога;
 - текст комментария;
 - идентификатор автора комментария;
 - дата публикации комментария;
 - рейтинг комментария;

Выходные параметры системы

Выходными параметрами системы являются веб-страницы. Доступность веб-страниц определяется ролью пользователя:

Администратор имеет доступ к веб-страницам:

- модерирования публикаций;
- модерирования комментариев;
- модерирования пользователей;
- отчетов статистики;

Гость получает доступ к веб-страницам:

- со списком публикаций;
- просмотра публикации;
- просмотра информации о пользователе;

Зарегистрированный пользователь имеет доступ к веб-страницам:

- со списком публикаций;
- просмотра публикации;
- просмотра информации о пользователе;
- публикации собственных блогов;

Требования к составу и параметрам технических средств

Минимальные требования к компьютерам для развертывания:

- тактовая частота не менее 1.7 ГГц;
- оперативная память не менее 4 Гб;
- ОС версии не ниже Linux ubuntu 14.04 lts;
- свободное пространство на жестком диске не менее 40 Гб для ОС;

Требования к документации

Документация должна включать:

- руководство по развертыванию системы;
- руководство по использованию системы для администратора;

Требования по реализации со стороны заказчика.

1. Требуется использовать СОА (сервис-ориентированную архитектуру).
2. Система состоит из микросервисов, отвечающих за свой элемент бизнес-логики.
3. Взаимодействие между сервисам осуществляется посредством HTTP-запросов и/или очереди сообщений.
4. Требуется обеспечить возможность регистрации и авторизации пользователей с использованием аккаунта google.com
5. Необходимо реализовать один веб-интерфейс для фронтенда. Интерфейс должен быть доступен через тонкий клиент – браузер.
4. Запрещается непосредственное обращение одного сервиса к базам данных других сервисов.
5. Сервисы (кроме сервиса агрегации) недоступны пользователю, это реализуется их расположением во внутренней сети.
6. Валидация входных данных должна производиться и на стороне пользователя с помощью JavaScript скриптов, и на стороне сервиса агрегации. Остальные сервисы не должны валидировать входные данные.

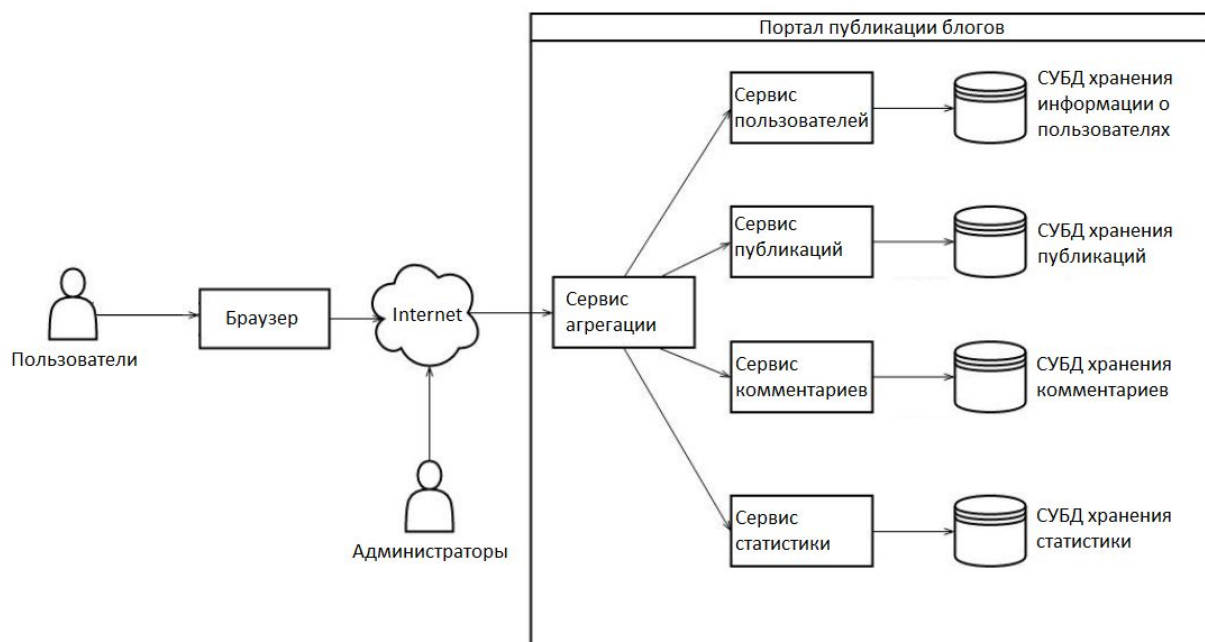


Рис. 1. Топология ПО «Система публикации блогов».

На рисунке 1 приведена топология ПО «Система публикации блогов». ПО состоит из следующих частей:

- Сервис агрегации
- Сервис пользователей
- Сервис публикаций
- Сервис комментариев
- Сервис статистики

Сервис агрегации

Основное назначение сервис агрегации заключается в сборе и обработке информации, собираемой от других сервисов. Также им предоставляется WEB-интерфейс для удобства работы с порталом.

Сервис пользователей

Предназначен для выполнения регистрации, аутентификации и авторизации пользователей.

Сервис публикаций

Предназначен для создания, изменения и выдачи публикаций.

Сервис комментариев

Предназначен для создания, изменения и выдачи комментариев к публикациям.

Сервис статистики

Предназначенная для формирования статистических отчетов о действиях пользователей.

Функциональные требования к сервисам.

1. **Сервис агрегации** – это серверное приложение, при разработке которого необходимо учитывать следующие факторы:
 - фронтенд должен принимать запросы по протоколу HTTP и формировать ответы пользователям портала в формате HTML или JSON;
 - в зависимости от типа запроса фронтенд должен отправлять последовательные запросы в соответствующие сервисы;
 - запросы к бекендам необходимо осуществлять по протоколу HTTP. Данные необходимо передавать в формате JSON. Данный текстовый формат обмена данными удобен для чтения;
2. **Сервис пользователей** – это серверное приложение, которое должно отвечать следующим требованиям по разработке.
 - Сервис пользователей должен принимать и возвращать данные в формате JSON по протоколу HTTP.
 - Выполнять авторизацию пользователей, проверять и удалять сессию, а также регистрировать пользователей.
 - Поддерживать авторизацию и регистрацию пользователей через google.com.
 - База данных, содержащая информацию о сессиях, должна находиться на этом же сервере.
3. **Сервис публикаций** должен быть серверным приложением, которое:
 - Должно принимать и отвечать на запросы в формате JSON по протоколу HTTP;
 - Поддерживать функции создания, просмотра, обновления и удаления публикаций.
 - Предоставлять список публикаций в формате JSON с пагинацией.
4. **Сервис комментариев** должен быть серверным приложением, которое:

- Должно принимать и отвечать на запросы в формате JSON по протоколу HTTP;
 - Поддерживать функции создания, просмотра, обновления и удаления комментариев.
 - Предоставлять список публикаций в формате JSON с пагинацией.
5. *Сервис статистики* должен быть серверным приложением, которое:
- принимает и отвечает на запросы в формате JSON по протоколу HTTP;
 - обрабатывает запросы на создание записи о действиях пользователя;
 - предоставляет список действий по фильтру только администратору;