

Практическая работа «Пользовательские теги» по курсу JavaEE.

В рамках данного задания необходимо разработать аналог Google Analytics для сбора статистики посещения страниц Вашего приложения с возможностью получения информации о клиенте и фиксации на уровне БД. В последующем, эта информация может оказаться полезной для сбора различных метрик и отображении информации в удобном представлении.

Работа будет включать следующие этапы:

1. Создание необходимых сущностей на уровне БД
2. Разработка пользовательского тега и сервиса для сбора статистики
3. Отображение статистической информации

В конечном счете, выполнение данной работы предполагает расширение возможностей web-приложения из предыдущих домашних работ.

Итак, в рамках задания предполагается:

- Создать сущность БД с ниже перечисленными статистическими атрибутами:
 1. Идентификатор
 2. Название маркера статистики (значение запрашивается из системных переменных окружения)
 3. Наименование «JSP-страницы» или модуля для SPA-приложения
 4. Клиентский IP-адрес
 5. Информация о браузере
 6. Клиентское время обращения
 7. Серверное время поступления запроса
 8. Прочая интересная на Ваш взгляд информация о клиенте (не менее двух дополнительных характеристик о пользовательском посещении – например, логин авторизованного пользователя, куки, геолокация (если включена) и т.д.).
 9. Предыдущий идентификатор статистики (для всех последующих входов – для возможности анализа «маршрута» посещения Вашего приложения).
- Разработать веб-сервлет, осуществляющий прием данных о клиенте и последующем сохранении данной информации в виде хранимой процедуры с возвращаемым идентификатором. В случае возникновения сложностей с PLSQL, можно обойтись стандартным insert-ом или выполнить добавление посредством JPA. Данную функциональность следует вынести в качестве отдельного сервиса для возможности повторного использования этой бизнес-логики. В качестве ответа данного сервлета должен быть возвращен json-объект с идентификатором созданной записи. Предусмотреть возможность гибкого отключения сбора статистики без пересборки приложения.
- Клиентский код сбора информации о пользователе обернуть в виде пользовательского тега, который следует разместить на всех jsp-страницах Вашего приложения.
- Предусмотреть дополнительную форму/страницу, сгенерированную средствами любого понравившегося шаблонизатора и доступную после авторизации в системе, в которой отображается собранная информация о посещениях пользователей с выводом всех статистических атрибутов. В простейшем варианте можно вывести информацию в табличном виде, либо воспользоваться визуальными диаграммами для отображения бизнес-метрик (опционально).
- Для возможности сбора статистики, в том числе и в сторонних приложениях, следует реализовать CORS для сервлета сбора статистической информации, для поддержки выполнения кросс-доменных JSONP-запросов (опциональное задание).

Общий комментарий: Если в предыдущих работах использовалась архитектура SPA, то для сбора статистики следует воспользоваться асинхронными запросами к сервлету фиксации статистики.