

- **API Gateway:** Отвечает за обработку всех входящих запросов от пользователей и маршрутизацию их к соответствующим сервисам.
- **User Service:** Управляет данными пользователей, такими как учетные записи, профили и аутентификация.
- **Tweet Service:** Обрабатывает публикацию новых твитов, их хранение и распределение по пользователям.
- **Feed Service:** Обеспечивает сбор и отображение твитов в ленте пользователя.
- **Cache:** Используется для ускорения доступа к часто запрашиваемым данным, таких как лента пользователя.
- **Database:** Хранит все данные системы, включая профили пользователей, твиты, связи между пользователями и т. д.
- **Load Balancer:** Распределяет входящие запросы между различными сервисами, обеспечивая масштабируемость и отказоустойчивость системы.

Функциональные требования

- **Пост твита:**
 - Пользователь отправляет твит через интерфейс веб-приложения или мобильного приложения.
 - Запрос поступает в **API Gateway**, который перенаправляет его в **Tweet Service**.
 - **Tweet Service** сохраняет твит в **Database** и обновляет кэшированные данные в **Cache** для немедленного отображения в ленте.
 - **Feed Service** автоматически обновляет ленту соответствующих пользователей.
- **Просмотр ленты:**
 - Пользователь запрашивает свою ленту через веб-приложение или мобильное приложение.
 - **API Gateway** направляет запрос в **Feed Service**.
 - **Feed Service** проверяет **Cache** для получения обновленных твитов. Если данные не найдены в кэше, сервис запрашивает их из **Database**.
 - Лента отображается пользователю с минимальной задержкой.

Нефункциональные требования

- **Масштабирование до 1 миллиарда пользователей:**
 - **Load Balancer** распределяет запросы между несколькими экземплярами сервисов, обеспечивая возможность обработки большого числа пользователей.
 - **Database** и **Cache** настроены на использование кластеров для обеспечения высокой доступности и быстрого доступа к данным.
 - Использование распределенных сервисов позволяет системе масштабироваться горизонтально.

- **Задержка обновления ленты до 300 мс:**

- **Feed Service** использует кэширование для быстрого доступа к часто запрашиваемым данным, что значительно снижает задержку.
- Оптимизация запросов в базу данных и использование асинхронной обработки запросов позволяет сократить время обработки до минимально возможного уровня.

Объяснения для каждого компонента и их ролей в системе:

- **API Gateway:** Центральный узел, принимающий все входящие запросы и направляющий их к соответствующим сервисам. Обеспечивает безопасность, контроль доступа и балансировку нагрузки.
- **User Service:** Обеспечивает управление учетными записями пользователей, включая регистрацию, авторизацию и хранение профилей.
- **Tweet Service:** Основной компонент, отвечающий за создание и сохранение твитов. Обновляет базу данных и кэш после каждого нового твита.
- **Feed Service:** Собирает и отображает ленту пользователя, используя кэш для повышения производительности. Обновляется в реальном времени при появлении новых твитов.
- **Cache:** Используется для хранения часто запрашиваемых данных, таких как лента пользователей, для уменьшения задержки при запросах.
- **Database:** Основное хранилище данных системы. Хранит все твиты, профили пользователей, связи между пользователями и т. д.
- **Load Balancer:** Обеспечивает равномерное распределение нагрузки между всеми компонентами системы, предотвращая перегрузку отдельных сервисов.