

– ОК Google.

Решение от Google самое «серьезное»: отвечает серьезно, без лишней информации и шуток». Алиса и Siri пытаются воссоздать человеческий разговор: манерой ответа, интонациями.

Голосовые ассистенты – полезная технология, которая может помочь пользователю в любой жизненной ситуации. С помощью голоса вы можете выполнять разные действия: выполнять звонки, строить маршруты, искать ответы на интересующие вопросы и многое другое. Выбор голосового интерфейса в большей степени зависит от используемой платформы и предпочтений пользователя.

#### ЛИТЕРАТУРА

1. Голосовой интерфейс [Электронный ресурс]. – Режим доступа:

[https://ru.wikipedia.org/wiki/%D0%93%D0%BE%D0%BB%D0%BE%D1%81%D0%BE%D0%B2%D0%BE%D0%B9\\_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81](https://ru.wikipedia.org/wiki/%D0%93%D0%BE%D0%BB%D0%BE%D1%81%D0%BE%D0%B2%D0%BE%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81). – Дата доступа: 26.03.2021.

УДК 557.114:616-006

Студ. Е.П. Шестовец, Е.В. Дурко  
Науч. рук. ст. преп. Т.В. Кишкурно  
(кафедра информатики и веб-дизайна, БГТУ)

#### **СРАВНЕНИЕ АРХИТЕКТУРНЫХ СТИЛЕЙ REST И SOAP**

Архитектурный стиль – это крупномасштабная заранее заданная структура решения. Использование архитектурных стилей помогает нам построить систему быстрее, чем строить все с нуля. Архитектурные стили похожи на шаблоны, но позволяют решить более серьезную задачу.

Приложения нуждаются в посреднике для взаимодействия друг с другом, поэтому разработчики часто наводят мосты – интерфейсы прикладного программирования или API. Это необходимо для того, чтобы предоставить одной системе доступ к информации из другой системы. Для того, чтобы облегчить быстрые и крупномасштабные интеграции приложений, интерфейсы реализованы с использованием протоколов и / или спецификаций, которые определяют семантику и синтаксис сообщений, передаваемыми. Эти спецификации составляют архитектуру API.

Для того, чтобы разработчик мог выбрать подходящую для него архитектуру проектирования необходимо сравнить их, а также выделить плюсы и минусы.

SOAP (простой протокол доступа к объектам) – это стандартный протокол доступа к веб-службам, который существует уже давно. Первоначально разработанный Microsoft. Этот протокол основан на XML, что упрощает чтение, даже если сообщения длиннее и, следовательно, значительно медленнее передаются. Это могут быть некоторые причины для использования SOAP вместо того, что предлагают службы REST.

REST (передача репрезентативного состояния) – это стиль архитектуры программного обеспечения, предназначенный для распределенных гипермедийных систем, таких как Интернет. Этот термин относится конкретно к совокупности принципов проектирования сетевых архитектур. Есть несколько проектов, которые могут выиграть от архитектуры REST. В частности, те, для которых основная идея заключается в том, как серверные запросы выполняются клиентом на основе соответствующего ресурса [1].

В таблице приведена сравнительная характеристика.

**Таблица – Сравнение RESTи SOAP**

Параметр	REST	SOAP
Стиль	Архитектурный стиль	Протокол
По количеству обрабатываемых ресурсов	Подходит для операции с большим количеством ресурсов	Использует операции с небольшими ресурсами
Специализация	Ориентирован на масштабируемость и крупномасштабную производительность для распределенных гипермедийных систем	Ориентирован на разработку распределенных приложений
Формат	HTML, XML и JSON	Только XML
Кэширование данных	Можно	Нельзя кэшировать данные
Безопасность	Поддерживает SSL и HTTPS	Поддерживает SSL и WS-security
Обработка полезной нагрузки	Нет необходимости знания API	Имеет строгий договор для общения и требует знания всего перед любым взаимодействием
Соответствие ACID	Имеет встроенное соответствие ACID	Отсутствует

У обоих есть свои преимущества и недостатки, поэтому при выборе протокола доступа следует учитывать языки программирования, используемые в организации, среду приложения и требования приложения. Обе информационные архитектуры имеют свои специфические

ниши, однако, по сравнительному анализу, который мы сделали можно предположить, что REST получит большее распространение в будущем.

#### ЛИТЕРАТУРА

1. SOAPvs. REST: ALookatTwoDifferentAPIStyles [Электронный ресурс]. – 2017. – Режим доступа: <https://www.upwork.com/resources/soap-vs-rest-a-look-at-two-different-api-styles>. – Дата доступа: 28.03.2021.

УДК 557.114:616-006

Студ. В.В. Акуленкова, студ. Е.В. Шевченко  
Науч. рук. ст. преп. Т.В. Кишкурно  
(кафедра информатики и веб-дизайна, БГТУ)

### **ОСНОВНЫЕ ПРИНЦИПЫ РАЗРАБОТКИ КАЧЕСТВЕННЫХ ИНТЕРФЕЙСОВ**

Создание качественного интерфейса в проектировании – это самый объемный и один из самых важных этапов, который требует глубокого исследования и тщательно спланированного тестирования.

Психология и дизайн тесно связаны между собой, а гештальт-принципы помогают понять и контролировать эти связи. Можно выделить следующие принципы [1]:

- завершение. Люди склонны подсознательно дорисовывать изображение, если оно им кажется незавершенным;

- симметрия и упорядоченность. Часто люди воспринимают объекты как упорядоченные группы, симметрично расположенные вокруг условного центра;

- фигура и фон. Сознание человека балансирует равновесием восприятия фигуры и фона, в результате чего одна из составляющих объекта будет восприниматься как элемент, несущий основную смысловую нагрузку, а второй — второстепенный;

- принцип связанности. Если два элемента визуально связаны третьим, они воспринимаются как единое целое;

- зонирование. Группа элементов будет восприниматься сознанием как единое целое, если она ограничена замкнутой областью;

- близость. Объекты, расположенные ближе друг к другу, вероятнее воспринимаются как единое целое, в сравнении с объектами, расположенными на большем расстоянии;

- продолжение. Элементы, расположенные на одной линии, воспринимаются как нечто целое, когда другие критерии группировки (цвет, форма, размер) не имеют значения.