



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
ТЕХНОЛОГИЧЕСКОГО ОБРАЗОВАНИЯ**

Кафедра информационных технологий и электронного обучения

Основная профессиональная образовательная программа

Направление подготовки 09.03.01 Информатика и вычислительная техника

Направленность (профиль) «Технологии разработки программного обеспечения»
форма обучения – очная

Аналитический отчёт

«Web-технологии (Web service design)»

Обучающегося 4 курса
Нахатакяна Артура Романовича

Научный руководитель:
кандидат физико-математических наук,
доцент кафедры ИТиЭО
Государев Илья Борисович

Санкт-Петербург
2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
ОСНОВНЫЕ ПОДХОДЫ К ПРОЕКТИРОВАНИЮ ВЕБ-СЕРВИСОВ.....	3
АРХИТЕКТУРНЫЕ СТИЛИ И ТЕХНОЛОГИИ WEB SERVICE DESIGN.....	4
ВОПРОСЫ НАДЁЖНОСТИ, МАСШТАБИРУЕМОСТИ И БЕЗОПАСНОСТИ ВЕБ-СЕРВИСОВ	5
ЗАКЛЮЧЕНИЕ.....	5
ЛИТЕРАТУРА	7

ВВЕДЕНИЕ

Web-технологии играют ключевую роль в развитии современных информационных систем и цифровых сервисов. Практически все прикладные программные решения — от мобильных приложений до корпоративных платформ и облачных экосистем — используют веб-сервисы как основу для взаимодействия между компонентами. В этих условиях особую значимость приобретает проектирование веб-сервисов, которое определяет надёжность, расширяемость и удобство сопровождения программных систем.

Под web-сервисом понимается программный компонент, доступный по сети и взаимодействующий с другими компонентами посредством стандартных протоколов и форматов данных. Web Service Design охватывает архитектурные принципы, методы проектирования интерфейсов, способы обмена данными и подходы к обеспечению качества сервиса. Грамотно спроектированный веб-сервис снижает связанность компонентов, упрощает развитие системы и повышает её устойчивость к изменениям требований.

Целью данной работы является анализ основных подходов и технологий проектирования веб-сервисов на основе научной и учебной литературы, а также материалов профессиональных источников. В работе рассматриваются архитектурные стили, ключевые принципы проектирования и актуальные проблемы Web Service Design.

ОСНОВНЫЕ ПОДХОДЫ К ПРОЕКТИРОВАНИЮ ВЕБ-СЕРВИСОВ

Проектирование веб-сервисов базируется на общих принципах инженерии программного обеспечения, таких как модульность, слабая связанность и высокая связность компонентов. Одним из ключевых подходов является контракт-ориентированное проектирование, при котором интерфейс сервиса описывается формально и не зависит от его внутренней реализации. Это позволяет клиентам и серверам развиваться независимо друг от друга.

Важным принципом Web Service Design является ориентация на интерфейсы. Веб-сервис предоставляет чётко определённый набор операций и форматов данных, которые описываются с помощью спецификаций (например, OpenAPI или WSDL). Такой подход повышает прозрачность взаимодействия и облегчает интеграцию сторонних систем.

Широкое распространение получил принцип stateless-взаимодействия, согласно которому каждый запрос клиента должен содержать всю необходимую информацию для его обработки. Отсутствие состояния на стороне сервера упрощает масштабирование и повышает отказоустойчивость системы. Этот принцип особенно характерен для REST-ориентированных веб-сервисов.

Также важную роль играет версионирование API. Поскольку веб-сервисы часто используются внешними клиентами, изменения интерфейсов должны выполняться контролируемо. Использование версий позволяет поддерживать обратную совместимость и снижает риски при обновлении сервисов.

АРХИТЕКТУРНЫЕ СТИЛИ И ТЕХНОЛОГИИ WEB SERVICE DESIGN

В современной практике выделяют несколько основных архитектурных стилей веб-сервисов. Классическим подходом является SOAP-ориентированная архитектура, основанная на использовании XML-сообщений и строгих формальных контрактов. SOAP-сервисы обеспечивают высокий уровень формализации и безопасности, однако отличаются сложностью и избыточностью.

Наибольшее распространение в настоящее время получил архитектурный стиль REST (Representational State Transfer). REST-сервисы используют стандартные HTTP-методы и ресурсо-ориентированную модель. Простота, гибкость и хорошая совместимость с веб-инфраструктурой сделали REST де-факто стандартом для проектирования веб-API.

В последние годы активно развиваются альтернативные подходы, такие как GraphQL и gRPC. GraphQL предоставляет клиенту возможность запрашивать только необходимые данные, что снижает сетевую нагрузку и повышает гибкость API. gRPC, в свою очередь, ориентирован на высокопроизводительное взаимодействие между сервисами и широко применяется в микросервисных архитектурах.

С точки зрения архитектуры приложений Web Service Design тесно связан с микросервисным подходом. Микросервисы предполагают разбиение системы на небольшие независимые сервисы, каждый из которых выполняет ограниченный набор функций и взаимодействует с другими

сервисами через веб-интерфейсы. Такой подход упрощает масштабирование и развитие сложных систем, но требует строгого соблюдения принципов проектирования API.

ВОПРОСЫ НАДЁЖНОСТИ, МАСШТАБИРУЕМОСТИ И БЕЗОПАСНОСТИ ВЕБ-СЕРВИСОВ

Одной из ключевых задач Web Service Design является обеспечение надёжности и доступности сервисов. Для этого применяются механизмы балансировки нагрузки, репликации и автоматического восстановления. Архитектура веб-сервисов должна учитывать возможность частичных отказов и предусматривать корректную обработку ошибок.

Масштабируемость достигается за счёт горизонтального масштабирования и отказа от хранения состояния. Использование контейнеризации и облачных платформ позволяет динамически изменять количество экземпляров веб-сервисов в зависимости от нагрузки. Проектирование сервисов с учётом этих возможностей является важной частью современного Web Service Design.

Безопасность веб-сервисов также напрямую связана с их архитектурой. На этапе проектирования необходимо учитывать вопросы аутентификации, авторизации и защиты передаваемых данных. Широко применяются такие механизмы, как OAuth 2.0, JWT-токены и HTTPS-шифрование. Ошибки в проектировании API могут привести к утечкам данных и уязвимостям, даже при использовании современных средств защиты.

Таким образом, Web Service Design представляет собой комплексную задачу, объединяющую архитектурные, технологические и организационные аспекты разработки.

ЗАКЛЮЧЕНИЕ

В ходе анализа было установлено, что web-технологии и проектирование веб-сервисов являются фундаментальной основой современных программных систем. Эффективный Web Service

Design позволяет создавать гибкие, масштабируемые и надёжные приложения, способные адаптироваться к изменениям требований и росту нагрузки.

Рассмотренные архитектурные стили и подходы показывают, что выбор технологии и принципов проектирования должен определяться задачами конкретной системы. REST, GraphQL и gRPC решают разные классы проблем, а микросервисная архитектура усиливает требования к качеству проектирования API.

Таким образом, Web Service Design можно рассматривать как одно из ключевых направлений современной web-разработки, от которого напрямую зависит качество, безопасность и жизнеспособность информационных систем.

ЛИТЕРАТУРА

1. Сычев, А. В. Web-технологии : учебное пособие / А. В. Сычев. — Воронеж : ВГУ, 2021. — 163 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/455018> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
2. Кочер, П. С. Микросервисы и контейнеры Docker : руководство / П. С. Кочер ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2019. — 240 с. — ISBN 978-5-97060-739-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/123710> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
3. Смоленцева, Т. Е. Проектирование и разработка WEB-приложений: Практикум : учебное пособие / Т. Е. Смоленцева. — Москва : РТУ МИРЭА, 2023. — 68 с. — ISBN 978-5-7339-1759-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/368954> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
4. Государев, И. Б. Введение в веб-разработку на языке JavaScript : учебное пособие для вузов / И. Б. Государев. — 2-е изд., стер. — Санкт-Петербург : Лань, 2024. — 144 с. — ISBN 978-5-507-49407-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/388679> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.
5. Золкин, А. Л. Проектирование и конфигурация компьютерных сетей с внедрением микросервисной архитектуры : учебное пособие для СПО / А. Л. Золкин, В. Д. Мунистер. — 2-е изд., стер. — Санкт-Петербург : Лань, 2026. — 120 с. — ISBN 978-5-507-51389-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/511507> (дата обращения: 29.12.2025). — Режим доступа: для авториз. пользователей.