

$$Q_{n,max} = \frac{F_n}{2\zeta_n \omega_n^2}$$

3) Влияние положения двигателя. Амплитуда вынужденных колебаний пропорциональна  $\sin(n\pi a/L)$ . При установке двигателя в узле  $n$ -й моды ( $a = L/n, 2L/n, \dots$ ) возбуждение этой моды отсутствует.

4) Переменная частота вращения. При разгоне двигателя частота  $\omega(t)$  изменяется во времени. Уравнение для обобщённой координаты принимает вид:

$$\ddot{q}_n + 2\zeta_n \omega_n \dot{q}_n + \omega_n^2 q_n = \frac{2m_0 r}{\rho AL} \cdot \omega^2(t) \sin\left(\int_0^t \omega(\tau) d\tau\right) \cdot \sin\left(\frac{n\pi a}{L}\right)$$

При линейном законе разгона  $\omega(t) = \varepsilon t$ , где  $\varepsilon$  — угловое ускорение, получаем:

$$\phi(t) = \frac{1}{2} \varepsilon t^2$$

$$\ddot{q}_n + 2\zeta_n \omega_n \dot{q}_n + \omega_n^2 q_n = \frac{2m_0 r \varepsilon^2 t^2}{\rho AL} \sin\left(\frac{1}{2} \varepsilon t^2\right) \cdot \sin\left(\frac{n\pi a}{L}\right)$$

Данное уравнение не имеет аналитического решения в замкнутой форме и требует численного интегрирования.

5) Амплитудно-частотная характеристика. Безразмерная амплитуда колебаний определяется соотношением:

$$A(\Omega) = \frac{1}{\sqrt{(1-\Omega^2)^2 + (2\zeta\Omega)^2}}$$

где  $\Omega = \omega/\omega_n$  — безразмерная частота.

6) Фазовая характеристика. При  $\omega \ll \omega_n$  фаза  $\psi \approx 0$  (колебания в фазе с силой). При  $\omega \gg \omega_n$  фаза  $\psi \approx \pi$  (колебания в противофазе). В резонансе  $\psi = \pi/2$  независимо от величины демпфирования.

7) Добротность системы. Добротность  $Q$  характеризует остроту резонанса:

$$Q = \frac{1}{2\zeta} = \frac{\omega_n}{\Delta\omega}$$

где  $\Delta\omega$  — ширина резонансной кривой на уровне 0,707 от максимума.

**Заключение.** Разработана математическая модель вынужденных колебаний горизонтальной балки под действием неуравновешенной вращающейся массы с вязким демпфированием.

Установлено, что резонанс наступает при совпадении частоты вращения двигателя с одной из собственных частот балки  $\omega_n$ .

Для режима переменной частоты вращения получено дифференциальное уравнение, найдено численное решение при линейном разгоне.

Результаты моделирования могут быть использованы для расчёта допустимых режимов работы оборудования с вращающимися элементами, выбора параметров виброзащиты и оценки динамической прочности конструкций.

## СИСТЕМА МОНИТОРИНГА И ДОСТУПНОСТИ ВЕБ-ПРИЛОЖЕНИЙ

*Тын-Лай-Си А.С.*

*студентка 4 курса ВГУ имени П.М. Машерова, г. Витебск, Республика Беларусь*

*Научный руководитель – Никитин А.И., канд. физ.-мат. наук, доцент*

В условиях активного развития цифровых технологий и роста числа веб-приложений и API их стабильная работа становится критически важной для бизнеса, образовательных платформ и государственных сервисов. Сбои и простои веб-сервисов приводят к финансовым потерям, ухудшению пользовательского опыта и снижению доверия к платформам. Согласно аналитическим отчётам ведущих облачных провайдеров, среднегодовое время недоступности даже крупных сервисов может достигать нескольких

часов [1]. Традиционные системы мониторинга (Zabbix, Prometheus, Nagios) требуют сложной настройки и мощной инфраструктуры, тогда как облачные сервисы (UptimeRobot, Pingdom) ограничены в кастомизации интервалов проверок, каналов уведомлений и анализа данных. В связи с этим возникает потребность в разработке эффективной, масштабируемой и гибкой системы мониторинга доступности веб-приложений и API. Таким образом, целью работы является разработка веб-приложения для автоматизированного мониторинга сайтов и API, обеспечивающего сбор ключевых метрик производительности, оперативное уведомление о сбоях и простоту сопровождения.

**Материал и методы.** Система мониторинга реализована с использованием современного технологического стека. Серверная часть разработана на базе фреймворка NestJS с использованием TypeScript[2]. Для хранения данных используется реляционная база данных PostgreSQL. Работа с базой данных осуществляется посредством ORM-инструмента Prisma, обеспечивающего типобезопасное взаимодействие с данными. Обработка асинхронных задач мониторинга реализована с помощью системы очередей BullMQ, работающей на базе Redis. Такой подход позволяет распределять задачи проверки между воркерами и обеспечивать регулярное выполнение запросов к отслеживаемым сервисам. Система выполняет периодические HTTP-запросы к указанным веб-ресурсам и API [3]. В ходе проверки фиксируются следующие показатели:

- время отклика сервера;
- HTTP-код ответа;
- частота возникновения ошибок;
- история проверок доступности.

Проверки выполняются с интервалом 10–600 секунд, что позволяет масштабировать мониторинг от нескольких до десятков ресурсов одновременно. Тестирование проводилось в среде Docker Compose с моделированием сбоев и нагрузочным тестированием.

**Результаты и их обсуждение.** В ходе работы разработана система мониторинга доступности веб-сайтов и API-сервисов. Основные возможности системы включают:

- автоматическая проверка доступности сервисов с заданным интервалом времени;
- сбор метрик производительности, включая время отклика и коды HTTP-ответов;
- хранение исторических данных для анализа стабильности работы сервисов;
- оперативное уведомление пользователей о возникновении сбоев;
- модульная архитектура, обеспечивающая расширяемость системы.

По сравнению с существующими инструментами мониторинга, разработанное решение обеспечивает более гибкое управление задачами и интеграцию с современным стеком технологий (NestJS, Redis, BullMQ, PostgreSQL), что повышает эффективность и надежность мониторинга.

**Заключение.** Разработанная система мониторинга обеспечивает контроль доступности веб-приложений и API-сервисов, а также сбор ключевых метрик производительности. Использование современных технологий разработки обеспечивает масштабируемость и гибкость архитектуры системы. Перспективным направлением развития является расширение аналитических возможностей системы и внедрение инструментов визуализации статистики работы сервисов.

1 Atatus. Prometheus vs Zabbix: A Hands-On Technical Comparison [Электронный ресурс] // Atatus Blog. – URL: <https://www.atatus.com/blog/prometheus-vs-zabbix> (дата обращения: 01.03.2026).

2 NestJS. Official Documentation [Электронный ресурс] // NestJS. – URL: <https://docs.nestjs.com/> (дата обращения: 01.03.2026).

3 Новый В.В. Программирование сетевых приложений на основе TCP сокетов : методические рекомендации к выполнению лабораторных работ / В.В. Новый ; УО «Витебский государственный университет имени П.М. Машерова», Каф. прикладного и системного программирования. – Витебск : ВГУ имени П.М. Машерова, 2023. – 25 с. URL - <https://rep.vsu.by/bitstream/> (дата обращения: 01.03.2026). – Текст: электронный.