

УДК 004.4:338.2

5.2.2. Математические, статистические и инструментальные методы в экономике

**МОДЕЛИРОВАНИЕ И ЭКОНОМИКО-СТАТИСТИЧЕСКАЯ ОЦЕНКА НАДЕЖНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ ЭЛЕКТРОННОЙ ОЧЕРЕДИ ЗАЯВОК**

Великанова Лариса Олеговна  
канд. эк. наук, доцент  
e-mail: [velikanova.l@kubsau.ru](mailto:velikanova.l@kubsau.ru)

Аванесян Карина Зармиковна  
Магистр 2 курса курса факультета прикладной информатики  
e-mail: [karina.a.4101@yandex.ru](mailto:karina.a.4101@yandex.ru)  
*Кубанский государственный аграрный университет имени И.Т. Трубилина, Краснодар, Россия*

Цель данной работы – разработать и обосновать архитектурные, технические и организационные меры для создания высоконадежной информационной системы электронной очереди заявок, удовлетворяющей строгим требованиям дистрибуторской компании, с применением математико-статистических методов анализа надежности и инструментальных средств мониторинга, обеспечивающих экономически обоснованное управление IT-инфраструктурой как ресурсом предприятия

Ключевые слова: НАДЕЖНОСТЬ ИС, ТИКЕТ-СИСТЕМА, ОТКАЗОУСТОЙЧИВОСТЬ, МОНИТОРИНГ, ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ, ДИСТРИБУЦИЯ, УПРАВЛЕНИЕ IT-СЕРВИСАМИ

<http://dx.doi.org/10.21515/1990-4665-217-013>

## Введение

В современных условиях цифровой трансформации дистрибуторский бизнес характеризуется высокой зависимостью от бесперебойной работы IT-инфраструктуры. Простой даже одного элемента, например, системы управления заявками (тикет-системы) IT-отдела, способен парализовать логистические операции, ведущие к прямым финансовым потерям и репутационным рискам. Как отмечает Великанова Л.О., «надежность

<http://ej.kubagro.ru/2026/03/pdf/13.pdf>

UDC 004.4:338.2

5.2.2. Mathematical, statistical and instrumental methods in economics

**MODELING AND ECONOMIC AND STATISTICAL ASSESSMENT OF THE RELIABILITY OF THE INFORMATION SYSTEM OF THE ELECTRONIC QUEUE OF APPLICATIONS**

Velikanova Larisa Olegovna  
Candidate of Economic Sciences. Associate Professor  
e-mail: [velikanova.l@kubsau.ru](mailto:velikanova.l@kubsau.ru)

Avanesyan Karina Zarmikovna  
2d year Master's Degree from the Faculty of Applied Informatics  
e-mail: [karina.a.4101@yandex.ru](mailto:karina.a.4101@yandex.ru)  
*Kuban State Agrarian University named after I.T. Trubilin, Krasnodar, Russia*

The purpose of this work is to develop and substantiate architectural, technical and organizational measures for creating a highly reliable electronic application queue information system that meets the strict requirements of a distribution company, using mathematical and statistical methods of reliability analysis and monitoring tools that provide cost-effective management of IT infrastructure as a resource of an enterprise

Keywords: IS RELIABILITY, TICKET SYSTEM, FAULT TOLERANCE, MONITORING, ECONOMIC JUSTIFICATION, DISTRIBUTION, IT SERVICE MANAGEMENT

информационной системы является комплексным свойством, которое определяет способность системы выполнять требуемые функции в заданных условиях в течение определенного времени» [1]. Таким образом, проектирование тикет-системы должно с самого начала рассматриваться через призму обеспечения надежности как её неотъемлемого атрибута, а не как опциональной надстройки. В контексте экономики эффективность подобных проектов требует строгого финансового обоснования, поскольку «бизнес-план инвестиционного проекта представляет собой документ, который дает развернутое обоснование перспективности проекта и возможности его эффективной реализации» [5]. А также экономическое обоснование подобных проектов приобретает критическую важность, так как «любое проектное решение требует тщательной экономической оценки, поскольку определяет направление инвестиций и их будущую отдачу» [6].

#### **Анализ требований и критерии надежности**

Надежность системы оценивается через триаду ключевых атрибутов: доступность (Availability), целостность (Integrity) и конфиденциальность (Confidentiality), расширенную для бизнес-задач трибутом производительности (Performance). Экономическая целесообразность принимаемых решений должна быть тщательно просчитана, так как «основной задачей бизнес-планирования является построение эффективной системы достижения стратегических целей компании» [5].

#### **Целевые показатели (SLA) и анализ рисков:**

- **Доступность:**  $\geq 99.9\%$  в рабочее время (08:00–20:00). Это допускает простой не более ~1.7 часа в квартал. Для бизнес-критичных систем, согласно [1], допустимый простой составляет 8–10 часов в год, что делает задачу достижимой.
- **Целостность данных:** Гарантия отсутствия потерь или несанкционированных изменений данных заявки на всех этапах её

жизненного цикла. RPO (Recovery Point Objective) = 0, RTO (Recovery Time Objective) < 15 минут для критичных операций.

- **Производительность:** Время отклика 95-го перцентиля (p95) < 3 с для операций создания и обновления заявки; < 1 с для чтения.

#### **Угрозы надежности:**

1. **Аппаратные сбои:** отказ сервера, диска, сетевого оборудования.
2. **Программные ошибки:** баги в коде, утечки памяти, deadlock в БД.
3. **Перегрузка:** резкий рост числа заявок (всплеск нагрузки).
4. **Человеческий фактор:** ошибочные конфигурации, некорректные операции с данными.
5. **Внешние угрозы:** DDoS-атаки, попытки несанкционированного доступа.

#### **Архитектурные решения**

##### **Выбор и эволюция архитектуры:**

Для начального этапа с прогнозируемой нагрузкой до 1000 заявок/день оправдан модульный монолит. Это упрощает обеспечение транзакционной целостности данных, развертывание и отладку [3]. Однако, для критичных к доступности компонентов (сервис уведомлений, обработка вложений) предлагается использовать микросервисный подход, вынося их в отдельные, изолированно масштабируемые сервисы. Это создает основу для будущей эволюции в сторону полноценной микросервисной архитектуры при росте нагрузки.

##### **Схема обеспечения отказоустойчивости:**

- Кластеризация приложения: Развертывание 2+ экземпляров монолита за балансировщиком нагрузки (например, Nginx с алгоритмом round-robin и проверкой здоровья).
- Репликация базы данных: Схема Master-Slave репликации с использованием СУБД PostgreSQL. Асинхронная репликация на

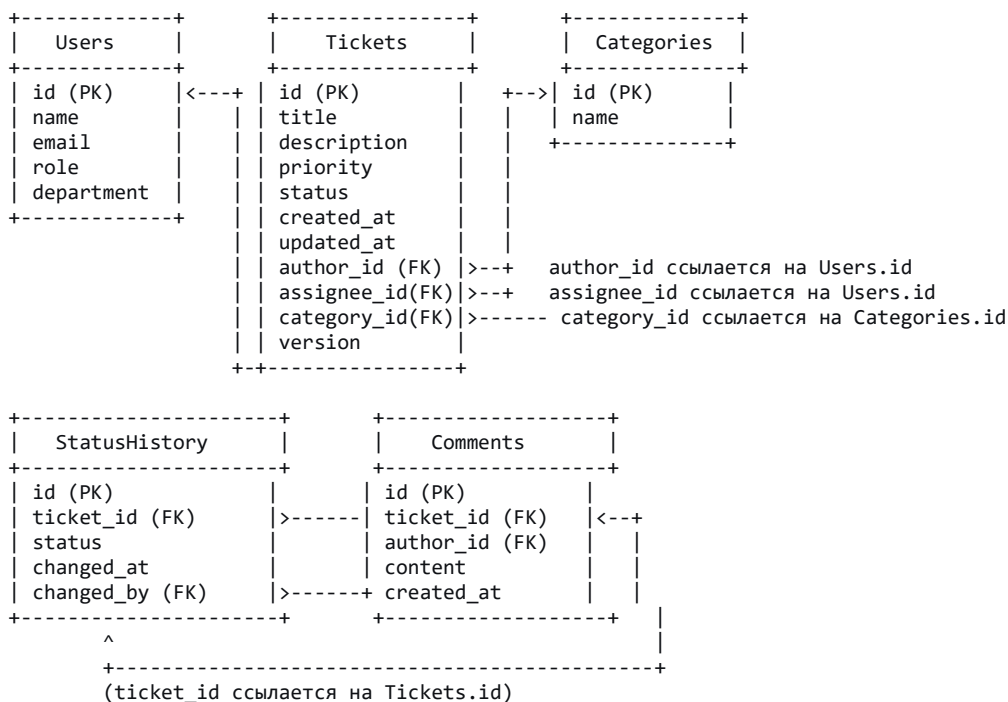
«горячий» standby-сервер обеспечивает  $RPO \approx 0$ . Возможность переключения (failover) при падении мастера.

- Резервное копирование: Ежедневное полное бэкапы + инкрементальные каждые 2 часа. Хранение на геораспределенном объектном хранилище (например, S3-совместимом) с политикой жизненного цикла.

### Обеспечение надежности на уровне данных и кода

#### Проектирование базы данных:

Нормализованная схема (рис.1) включает основные сущности: Users, Tickets, Categories, StatusHistory, Comments. Для обеспечения производительности вводятся индексы по полям ticket.status, ticket.created\_at, ticket.priority.



#### Транзакции и конкурентный доступ:

Все операции изменения состояния заявки (CREATE, UPDATE STATUS, ADD COMMENT) оборачиваются в ACID-транзакции. Для предотвращения «гонок» при обновлении одного тикета несколькими

операторами используется оптимистичная блокировка (поле version в таблице Tickets).

```
sql
BEGIN;
SELECT version FROM tickets WHERE id = 123 FOR UPDATE;
-- Проверка актуальности версии, бизнес-логика
UPDATE tickets SET status = 'IN_PROGRESS', version = version + 1 WHERE id =
123 AND version = :expected_version;
COMMIT;
```

### **Обработка ошибок и устойчивость API:**

Пример улучшенного обработчика создания заявки с валидацией, транзакцией и структурированным логированием:

```
javascript
const { body, validationResult } = require('express-validator');

router.post('/api/tickets',
  [
    body('description').isLength({ min: 10, max: 1000 }),
    body('priority').isIn(['LOW', 'MEDIUM', 'HIGH', 'CRITICAL'])
  ],
  async (req, res) => {
    // 1. Валидация входных данных
    const errors = validationResult(req);
    if (!errors.isEmpty()) {
      return res.status(400).json({ errors: errors.array() });
    }

    const { authorId, category, description, priority } = req.body;
    const transaction = await sequelize.transaction(); // Начало транзакции

    try {
      // 2. Проверка существования категории и пользователя
      const [categoryExists, userExists] = await Promise.all([
        Category.findByPk(category, { transaction }),
        User.findByPk(authorId, { transaction })
      ]);

      if (!categoryExists || !userExists) {
        await transaction.rollback();
        return res.status(404).json({ error: 'Ресурс не найден' });
      }

      // 3. Создание заявки в транзакции
      const newTicket = await Ticket.create({
        authorId, category, description, priority,
        status: 'OPEN',
        version: 1
      }, { transaction });

      await StatusHistory.create({
        ticketId: newTicket.id,
        status: 'OPEN',

```

```
        changedBy: authorId
    }, { transaction });

    await transaction.commit(); // Фиксация изменений

    // 4. Асинхронная отправка уведомления (вне транзакции)
    notificationService.notify(newTicket).catch(err =>
console.error('Ошибка уведомления:', err));

    res.status(201).json(newTicket);

    } catch (err) {
        await transaction.rollback(); // Откат при любой ошибке
        console.error(`[${new Date().toISOString()}] Ошибка создания заявки:`,
{ error: err.message, stack: err.stack, userId: authorId });
        // Интеграция с Sentry для трейсинга
        Sentry.captureException(err);
        res.status(500).json({ error: 'Внутренняя ошибка сервера', requestId:
req.id });
    }
    }
    );
```

### Система мониторинга и оповещений

Для оперативного контроля и профилактики инцидентов развертывается стек мониторинга:

- **Prometheus:** Сбор метрик с приложения (через `express-prom-bundle`), БД (PostgreSQL Exporter), ОС (Node Exporter), балансировщика.
- **Grafana:** Визуализация дашбордов. Ключевые дашборды: «Общее здоровье системы», «Производительность БД», «Бизнес-метрики (кол-во открытых/закрытых заявок)».
- **Loki:** Централизованный сбор и анализ структурированных логов приложения.
- **Alertmanager:** Управление алертами с маршрутизацией в Telegram/Email.

### Ключевые метрики для алертинга:

- `http_request_duration_seconds{p95 > 3s}` – деградация производительности.
- `up{job="app"} == 0` – недоступность экземпляра приложения.
- `postgres_up == 0` – недоступность БД.
- `rate(http_requests_total{status=~"5.."}[5m]) > 0.1` – высокий уровень ошибок сервера.

### **Экономическое обоснование**

Обеспечение надежности требует инвестиций, что требует проведения тщательного экономического обоснования. Как справедливо отмечают Медведева В.Н. и Нарожная Г.А., «расчет экономической эффективности проекта является обязательным разделом бизнес-плана и позволяет оценить целесообразность вложения средств» [5]. Необходимо рассчитать баланс между стоимостью простоя и затратами на инфраструктуру.

#### **Оценка стоимости простоя (Downtime Cost):**

Для дистрибуторской компании средней руки (оборот ~500 млн руб./год) простой IT-системы может приводить к потерям:

- Прямые потери: Простой логистов/складских работников = 20 чел. \* 1000 руб./час \* 2 часа = 40 000 руб.
- Косвенные потери: Штрафы за срыв поставок, репутационный ущерб (оценка экспертно) = ~100 000 руб.
- Итого потенциальный ущерб от инцидента: ~140 000 руб.

#### **Затраты на инфраструктуру надежности:**

- Дополнительный сервер для реплики БД: ~80 000 руб./год.
- Лицензии/поддержка ПО: ~30 000 руб./год.
- Трудозатраты на настройку и поддержку: ~50 чел./час \* 2000 руб./час = 100 000 руб.(разово).
- Итого годовые дополнительные затраты: ~110 000 руб. + разовые 100 000 руб.

#### **ROI (Return on Investment):**

Предотвращение даже одного серьезного инцидента в год (ущерб 140 000 руб.) покрывает значительную часть эксплуатационных затрат. Повышение доступности также увеличивает производительность сотрудников и удовлетворенность внутренних клиентов, что дает долгосрочный стратегический эффект. Данный подход соответствует

принципам эффективного планирования, поскольку «ключевым аспектом бизнес-планирования является не только расчет текущих затрат, но и оценка долгосрочных экономических последствий принимаемых решений» [5]. Как отмечают Великанова и Ткаченко, «информационные технологии в экономике должны обеспечивать не только функциональность, но и экономическую эффективность» [2].

### **Организационные меры**

Надежность обеспечивается не только технологиями, но и процессами:

1. Регламент резервного копирования и восстановления: Ежедневное тестирование процедуры восстановления из бэкапа.
2. План аварийного восстановления (DRP): Четкий пошаговый алгоритм действий при отказе основных компонентов с назначенными ответственными.
3. Процесс управления изменениями: Любое обновление кода или конфигурации проходит стадию тестирования в staging-окружении.
4. Регулярный аудит и пересмотр метрик: Ежеквартальный пересмотр порогов срабатывания алертов и ключевых показателей надежности.

«Успех реализации проекта во многом зависит от качества планирования организационных мероприятий и управления ресурсами» [6].

### **Заключение**

В работе предложен комплексный проект построения надежной информационной системы электронной очереди заявок для IT-отдела дистрибуторской компании. Ключевыми результатами являются:

1. Определены целевые показатели надежности (SLA), основанные на анализе бизнес-рисков.

2. Предложена гибридная архитектура (модульный монолит с микросервисными компонентами), обеспечивающая баланс между простотой и отказоустойчивостью.
3. Детально описаны технические реализации механизмов обеспечения целостности данных (ACID-транзакции, оптимистичная блокировка) и доступности (кластеризация, репликация).
4. Разработана схема мониторинга на основе современного стека инструментов с выделением бизнес- и инфраструктурных метрик.
5. Приведено экономическое обоснование, показывающее целесообразность инвестиций в надежность.

Последовательная реализация предложенных мер позволит создать систему, которая не только выполняет свои функциональные требования, но и выступает надежным активом, минимизирующим операционные риски бизнеса. Эффективное бизнес-планирование в сфере информационных технологий должно обеспечивать синергию между техническими решениями и экономической эффективностью. Таким образом, предложенный подход интегрирует математико-статистические методы анализа с практикой проектирования информационных систем, обеспечивая их экономическую эффективность и соответствие стратегическим целям предприятия.

#### **Литература:**

1. Великанова, Л. О. Надежность информационных систем: учебное пособие. – Краснодар: КубГАУ, 2020.
2. Великанова, Л. О. Информационные технологии в экономике: учебное пособие / Л. О. Великанова, В. В. Ткаченко. – Краснодар: КубГАУ, 2021.
3. Экономическая информатика: учебное пособие / Л. О. Великанова, С. А. Курносов, Е. В. Попова [и др.]. – 2-е изд. перераб. и доп. – Краснодар: КубГАУ, 2018.
4. Будович Л. С., Старцева Ю. В. Бизнес-планирование в предпринимательской деятельности: Учебно-методическое пособие. – МИРЭА - Российский технологический университет, 2021.

5. Медведева, В. Н. Бизнес-планирование : учебное пособие / В. Н. Медведева, Г. А. Нарожная. – Москва : РТУ МИРЭА, 2025.

6. Голубева, Е. А. Экономическое обоснование проектных решений: практикум : учебное пособие / Е. А. Голубева. – Омск : СибАДИ, 2023.

### References

1. Velikanova, L. O. Nadezhnost' informacionnyh sistem: uchebnoe posobie. – Krasnodar: KubGAU, 2020.

2. Velikanova, L. O. Informacionnye tehnologii v jekonomike: uchebnoe posobie / L. O. Velikanova, V. V. Tkachenko. – Krasnodar: KubGAU, 2021.

3. Jekonomicheskaja informatika: uchebnoe posobie / L. O. Velikanova, S. A. Kurnosov, E. V. Popova [i dr.]. – 2-e izd. pererab. i dop. – Krasnodar: KubGAU, 2018.

4. Budovich L. S., Starceva Ju. V. Biznes-planirovanie v predprinimatel'skoj dejatel'nosti: Uchebno-metodicheskoe posobie. – MIRJeA - Rossijskij tehnologicheskij universitet, 2021.

5. Medvedeva, V. N. Biznes-planirovanie : uchebnoe posobie / V. N. Medvedeva, G. A. Narozhnaja. – Moskva : RTU MIRJeA, 2025.

6. Golubeva, E. A. Jekonomicheskoe obosnovanie proektnyh reshenij: praktikum : uchebnoe posobie / E. A. Golubeva. – Omsk : SibADI, 2023.