

## Введение

### Актуальность и постановка проблемы

В условиях современного производства и высококонкурентного рынка эффективное управление материальными потоками является критическим фактором успеха предприятия. Центральным звеном в этой системе выступает складской учет готовой продукции, точность и оперативность которого напрямую влияют на выполнение заказов, оборачиваемость капитала, уровень логистических издержек и, в конечном итоге, на удовлетворенность клиентов. Несмотря на это, многие предприятия, особенно в сегменте малого и среднего бизнеса, продолжают использовать для учета устаревшие, неинтегрированные методы.

Опора на бумажные журналы, разрозненные электронные таблицы (чаще всего в Microsoft Excel) и ручные операции порождает комплекс системных проблем:

- Низкая точность данных и «человеческий фактор»: Ручной ввод информации о приемке, отгрузке и перемещениях ведет к неизбежным ошибкам, расхождениям между фактическими остатками и данными в учетной системе («слепые зоны» склада).
- Отсутствие оперативной информации: Менеджер по продажам не может в реальном времени проверить наличие товара, а логист — оперативно найти нужную партию продукции по конкретным характеристикам (серийный номер, срок годности, категория качества). Это приводит к простоям, срыву сроков отгрузки и конфликтам.
- Трудоемкость и низкая эффективность инвентаризации: Процедура полной пересортицы остатков требует остановки работы склада, привлечения значительных человеческих ресурсов и часто дает неточные результаты, требующие последующих корректировок.
- Сложность анализа и планирования: Отсутствие структурированной истории движений и внятной аналитики затрудняет выявление тенденций (например, сезонности спроса на определенные позиции), оптимизацию складских запасов и принятие обоснованных управленческих решений.

Последствиями становятся прямые финансовые потери: недовложение или пересортица товара при отгрузке, замороженный капитал в излишках неликвидной продукции, штрафы за срыв поставок и снижение общей операционной эффективности цепи поставок.

### Анализ существующих решений и обоснование разработки

Рынок предлагает спектр решений для автоматизации складского учета — от специализированных WMS (Warehouse Management System) до модулей в

составе комплексных ERP-систем (например, SAP, 1C, Oracle). Однако для многих организаций они оказываются недоступны или избыточны.

Мощные корпоративные WMS и ERP-системы предполагают высокие затраты на лицензии, длительное и дорогостоящее внедрение, а также постоянные расходы на поддержку и обучение. Их сложность часто не соответствует потребностям предприятия, которому требуется четкое и простое решение для базового, но безошибочного учета готовой продукции.

С другой стороны, использование универсальных табличных редакторов или самописных баз данных не обеспечивает необходимой надежности, многопользовательского доступа с разграничением прав, автоматизации документооборота (приемные акты, накладные) и интеграции со смежными системами (например, с программой для торговли или производства).

Таким образом, существует потребность в специализированной, сфокусированной информационной системе, которая заполняет нишу между избыточными корпоративными продуктами и ненадежными кустарными решениями. Такая система должна обеспечивать точный оперативный учет, формирование первичных документов, базовую аналитику по остаткам и оборотам, при этом оставаясь интуитивно понятной, относительно недорогой в разработке и внедрении и легко настраиваемой под специфику конкретного производства.

Цель, задачи, объект и предмет исследования

Целью данной курсовой работы является проектирование и разработка архитектуры информационной системы для автоматизированного складского учета готовой продукции, направленной на повышение точности, оперативности и эффективности управления складскими запасами.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ предметной области: изучить типовые бизнес-процессы склада готовой продукции, выявить ключевые проблемы и сформулировать функциональные требования к системе.
2. Спроектировать архитектуру системы: разработать концептуальную и логическую модель базы данных, определить структуру основных модулей (учет номенклатуры, приемка, отгрузка, инвентаризация, отчетность) и спроектировать пользовательский интерфейс.
3. Выбрать и обосновать стек технологий для реализации (языки программирования, фреймворки, СУБД).
4. Разработать ключевые функциональные модули системы, включая справочники продукции, учет партий и серийных номеров, операции приемки и отгрузки, формирование оборотных ведомостей и отчетов по остаткам.

5. Протестировать основные сценарии работы системы для подтверждения ее корректности и соответствия требованиям.

Объектом исследования выступает процесс складского учета готовой продукции на промышленном предприятии.

Предметом исследования являются методы, модели и программные решения для автоматизации данного процесса.

Практическая значимость работы

Практическая значимость заключается в создании проектного решения (архитектуры, моделей данных, спецификаций), которое может быть использовано для непосредственной разработки рабочего программного продукта. Внедрение подобной системы позволит предприятию:

- Ликвидировать расхождения между фактическими и учетными остатками.
- Сократить время обработки заказов и отгрузки продукции.
- Повысить дисциплину и подотчетность на складе за счет фиксации всех операций.
- Получать оперативные и достоверные данные для анализа оборачиваемости и планирования закупок сырья или производства.
- Снизить логистические издержки и минимизировать финансовые потери, связанные с ошибками в учете.

Таким образом, тема работы является высокоактуальной, а ее результаты имеют конкретную прикладную ценность для повышения эффективности управления запасами на современных производственных предприятиях.

## ГЛАВА 1. АНАЛИЗ И ПРОЕКТИРОВАНИЕ СИСТЕМЫ СКЛАДСКОГО УЧЕТА ГОТОВОЙ ПРОДУКЦИИ

### 1.1. Комплексный анализ предметной области и проблематики

#### 1.1.1. Современные вызовы в управлении складскими запасами

В эпоху цифровой трансформации и глобализации цепей поставок эффективное управление материальными потоками становится стратегическим преимуществом для производственных предприятий. Склад готовой продукции выполняет роль критического узла, связывающего производственные мощности с рыночным спросом. Однако статистические данные свидетельствуют о сохраняющихся системных проблемах: согласно исследованиям Supply Chain Digest, до 65% предприятий малого и среднего сегмента продолжают использовать неавтоматизированные методы учета, что приводит к средним годовым потерям в размере 3-8% от стоимости складских запасов.

Ключевые проблемы традиционного подхода к складскому учету носят комплексный характер:

1. Информационные искажения и их последствия:
  - Средняя погрешность ручного учета составляет 5-15%, достигая в отдельных случаях 30% для быстрооборачиваемой продукции
  - Расхождения между физическими и учетными остатками создают "слепые зоны", парализующие планирование отгрузок
  - Ложная информация о наличии товара приводит к потерям продаж и ухудшению репутации компании
2. Операционные неэффективности:
  - Время поиска нужной позиции на складе при бумажном учете увеличивается на 70-120%
  - Простои погрузочно-разгрузочного оборудования достигают 25% рабочего времени из-за неоптимального размещения товара
  - Коэффициент использования складской площади редко превышает 60-65% при отсутствии системы адресного хранения
3. Кадровые и организационные ограничения:
  - Зависимость от компетенций конкретных сотрудников создает операционные риски

- Подготовка нового кладовщика занимает 3-6 месяцев при отсутствии формализованных процедур
- Сопротивление изменениям и консервативность персонала усугубляют проблему внедрения современных методов

#### 1.1.2. Структура и взаимосвязи предметной области

Предметная область складского учета готовой продукции представляет собой многоуровневую систему со сложными взаимосвязями:

Первичный уровень (физические объекты):

- Готовая продукция: Классифицируется по категориям, техническим характеристикам, единицам измерения. Для каждой позиции определяется минимальный/максимальный запас, условия хранения, срок годности
- Складские мощности: Иерархическая структура от общего склада до конкретной ячейки. Каждое место имеет уникальный идентификатор, технические характеристики (грузоподъемность, габариты), температурный режим
- Технологическое оборудование: Погрузчики, штабелеры, системы автоматической идентификации, которые должны быть интегрированы в учетную систему

Операционный уровень (бизнес-процессы):

- Приемочные операции: Включают не только количественную приемку, но и качественный контроль, присвоение идентификаторов партиям, формирование транспортных единиц
- Процессы размещения: Определение оптимальных мест хранения на основе алгоритмов ABC-XYZ анализа, ротационных принципов (FIFO, LIFO, FEFO), совместимости продукции
- Комплектационные операции: Формирование оптимальных маршрутов отбора, минимизация холостых пробегов, контроль полноты собранных заказов
- Инвентаризационный цикл: От плановых ежегодных инвентаризаций до непрерывного циклического учета с минимальным влиянием на операционную деятельность

Информационный уровень (данные и документы):

- Документы первичного учета: Приходные ордера, требования-накладные, акты списания, инвентаризационные описи
- Аналитические данные: История движений, показатели оборачиваемости, статистика ошибок, коэффициент использования мощностей
- Нормативно-справочная информация: Ставки НДС, нормы естественной убыли, классификаторы продукции, стандарты упаковки

## 1.2. Глубокий анализ и обоснование технологического стека

### 1.2.1. Методология выбора технологий для промышленных систем учета

При выборе технологического стека для системы складского учета применялась многофакторная модель оценки, учитывающая 12 ключевых критериев:

1. Техническая адекватность: Способность технологии решать специфические задачи предметной области
2. Экономическая эффективность: Совокупная стоимость владения в течение 5-летнего жизненного цикла
3. Масштабируемость: Возможность роста системы пропорционально росту бизнеса
4. Интеграционный потенциал: Поддержка стандартных протоколов обмена данными
5. Надежность и отказоустойчивость: Гарантии бесперебойной работы в производственных условиях
6. Безопасность: Соответствие требованиям ФЗ-152 и отраслевым стандартам
7. Производительность: Время отклика на критически важные операции
8. Экосистема: Наличие готовых библиотек, фреймворков, инструментов разработки
9. Кадровый потенциал: Доступность специалистов на региональном рынке труда
10. Документация: Качество и полнота технической документации
11. Сообщество: Активность и отзывчивость пользовательского сообщества
12. Перспективы развития: Дорожная карта развития технологии на ближайшие 3-5 лет

### 1.2.2. Сравнительный анализ серверных технологий

PHP в контексте промышленных систем:

- Архитектурные преимущества: Поддержка многопоточности через расширение pthreads, асинхронное программирование через ReactPHP, обработка очередей через RabbitMQ/Redis
- Производительность: PHP 8.1 демонстрирует на 40-50% лучшую производительность по сравнению с версией 7.4 благодаря JIT-компиляции
- Безопасность: Встроенные механизмы защиты от 95% известных веб-уязвимостей при правильной конфигурации
- Интеграционные возможности: Нативная поддержка SOAP, REST, GraphQL, WebSocket протоколов

Конкурентные технологии и их ограничения для проекта:

1. Java/Spring: Избыточная сложность для типовых операций CRUD, требующая в 2-3 раза больше ресурсов сервера

2. Python/Django: Ограничения в обработке синхронных транзакций при высокой нагрузке, проблемы с многопоточностью из-за GIL
3. Node.js: Риски блокировки event loop при CPU-интенсивных операциях анализа данных
4. C#/.NET: Лицензионные ограничения и зависимость от экосистемы Microsoft

### 1.2.3. Детальный анализ систем управления базами данных

MySQL 8.0 как оптимальное решение:

Оптимизация для складских операций:

- Поддержка оконных функций для аналитических запросов
- Индексы FULLTEXT для быстрого поиска по описаниям продукции
- Механизм Materialized Views для предрасчета агрегированных показателей

Обеспечение целостности данных:

- Транзакционность уровня REPEATABLE READ
- Каскадные ограничения внешнего ключа
- Триггеры для автоматического аудита изменений

Масштабируемость:

- Репликация Master-Slave для распределения нагрузки
- Поддержка шардирования через MySQL Router
- Горизонтальное масштабирование через кластер InnoDB

Альтернативы и их недостатки:

- PostgreSQL: Более высокая стоимость администрирования, ограниченная поддержка на стандартных хостингах
- Microsoft SQL Server: Лицензионные платежи увеличивают ТСО в 3-4 раза
- MongoDB: Отсутствие транзакционности на ранних версиях, сложности с JOIN-запросами

### 1.3. Расширенное техническое задание на разработку системы

#### 1.3.1. Детализированные функциональные требования

Модуль управления номенклатурным справочником:

- Поддержка иерархической структуры категорий до 7 уровней вложенности
- Возможность прикрепления технической документации, сертификатов, изображений
- Управление единицами измерения с коэффициентами пересчета
- Ведение альтернативных артикулов и штрих-кодов

- История изменений цен и характеристик с версионированием

#### Модуль адресного хранения:

- 3D-визуализация складских помещений с размещением товара
- Алгоритмы оптимизации размещения на основе ротации, веса, габаритов, условий хранения
- Управление зонами: приемки, хранения, отбора, брака, карантина
- Планирование и оптимизация маршрутов перемещения техники

#### Модуль оперативного учета:

- Приемка с поддержкой группового сканирования штрих-кодов
- Комплектация заказов с формированием оптимальных маршрутов отбора
- Инвентаризация с возможностью выборочного пересчета
- Резервирование товара под подтвержденные заказы
- Управление просроченной и некондиционной продукцией

### 1.3.2. Нефункциональные требования

#### Требования к производительности:

- Время отклика на стандартные операции не более 2 секунд
- Поддержка одновременной работы 50+ пользователей
- Обработка не менее 1000 транзакций в час в пиковые периоды
- Формирование стандартных отчетов за 5-10 секунд

#### Требования к надежности:

- Время восстановления после сбоя не более 15 минут
- Доступность системы 99,5% в рабочее время
- Автоматическое резервное копирование данных ежедневно
- Ведение журнала аудита всех значимых действий

#### Требования к безопасности:

- Соответствие требованиям PCI DSS для финансовых операций
- Шифрование конфиденциальных данных в базе (AES-256)
- Двухфакторная аутентификация для административных функций
- Ролевая модель доступа с минимальными привилегиями

### 1.3.3. Требования к интеграции

#### Внутренние системы предприятия:

- 1С:Предприятие через COM-соединение или веб-сервисы
- ERP-системы по протоколу OData
- Системы электронного документооборота
- Бухгалтерские программы через выгрузку в форматах XML/JSON



## Внешние системы и сервисы:

- Маркировка товаров через ГИС МТ (Честный ЗНАК)
- Электронные торговые площадки (Wildberries, OZON, Яндекс.Маркет)
- Транспортные компании для отслеживания отгрузок
- SMS и email рассылки для уведомлений клиентов

## 1.4. Детализированный план разработки системы

### 1.4.1. Фаза предпроектного анализа

1. Исследование и моделирование
  - Анализ существующих бизнес-процессов на складе
  - Интервью с ключевыми стейкхолдерами (начальник склада, кладовщики, логисты)
  - Документирование текущих проблем и узких мест
  - Разработка концептуальной модели системы
2. Техническое проектирование
  - Разработка логической модели базы данных (ER-диаграммы)
  - Проектирование архитектуры системы (диаграммы компонентов)
  - Выбор и обоснование технологического стека
  - Оценка рисков и разработка митигирующих мероприятий
3. Планирование и подготовка
  - Составление детального календарного плана
  - Подготовка технического задания
  - Создание прототипов ключевых интерфейсов
  - Формирование команды проекта и распределение ролей

### 1.4.2. Фаза разработки

1. Базовый каркас
  - Настройка среды разработки и инфраструктуры
  - Реализация модуля аутентификации и авторизации
  - Разработка ядра системы и базовых сервисов
  - Создание основных административных интерфейсов
2. Функциональные модули
  - Модуль управления номенклатурой и справочниками
  - Система адресного хранения и управления складом
  - Базовый функционал приемки и отгрузки
  - Формирование первичных документов
3. Расширенная функциональность
  - Система резервирования и управления заказами
  - Модуль инвентаризации и переучета
  - Аналитические отчеты и дашборды
  - Интеграционные механизмы

#### 1.4.3. Фаза тестирования и внедрения

1. Предварительное тестирование
  - Модульное тестирование отдельных компонентов
  - Интеграционное тестирование взаимодействия модулей
  - Тестирование производительности под нагрузкой
  - Проверка безопасности и уязвимостей
2. Пользовательское тестирование
  - Формирование группы тестировщиков из сотрудников склада
  - Проведение прямо-сдаточных испытаний
  - Сбор обратной связи и выявление проблем
  - Корректировка системы по результатам тестирования
3. Подготовка к внедрению
  - Разработка пользовательской документации
  - Подготовка обучающих материалов и видеоуроков
  - Проведение тренингов для будущих пользователей
  - Миграция исторических данных из старых систем
4. Промышленное внедрение
  - Поэтапный запуск системы в эксплуатацию
  - Техническая поддержка в переходный период
  - Мониторинг работы и сбор метрик эффективности
  - Финальная корректировка и оптимизация

#### 1.4.4. Постпроектная фаза

Первые 3 месяца после внедрения:

- Ежедневный мониторинг работы системы
- Оперативная техническая поддержка пользователей
- Сбор и анализ статистики использования
- Планирование доработок и улучшений

Долгосрочное сопровождение:

- Регулярные обновления и исправления ошибок
- Расширение функциональности по запросам пользователей
- Масштабирование системы при росте бизнеса
- Обучение новых сотрудников работе с системой

#### 1.5. Критерии успеха и ожидаемые результаты

##### 1.5.1. Количественные показатели эффективности

Операционные улучшения:

- Сокращение времени обработки заказа на 40-60%
- Увеличение точности учета до 99,5-99,9%
- Снижение потерь от пересортицы на 70-80%

- Увеличение оборачиваемости запасов на 25-35%

Экономический эффект:

- Сокращение логистических издержек на 15-25%
- Высвобождение оборотных средств за счет оптимизации запасов
- Снижение затрат на инвентаризацию в 3-4 раза
- Минимизация штрафов за срыв поставок

#### 1.5.2. Качественные улучшения

Для бизнеса:

- Повышение удовлетворенности клиентов за счет своевременных поставок
- Улучшение управляемости складским хозяйством
- Формирование прозрачной и достоверной отчетности
- Создание основы для дальнейшей цифровизации

Для персонала:

- Снижение рутинной нагрузки на сотрудников
- Устранение конфликтов из-за ошибок учета
- Повышение квалификации и технологической грамотности
- Улучшение условий труда за счет автоматизации

Для ИТ-инфраструктуры:

- Создание масштабируемой и расширяемой платформы
- Обеспечение интеграции с другими корпоративными системами
- Формирование базы для внедрения Industry 4.0 технологий
- Развитие компетенций в области промышленной автоматизации

Данный комплексный подход к проектированию системы складского учета готовой продукции обеспечивает не только решение текущих операционных проблем, но и создает основу для долгосрочного развития складской логистики предприятия в условиях цифровой экономики.

# ТЕХНИЧЕСКОЕ ЗАДАНИЕ

## 1. Основание для разработки

Основанием для разработки является курсовой проект студента по теме: «Проектирование и разработка архитектуры информационной системы для автоматизированного складского учета готовой продукции».

## 2. Назначение системы

Разрабатываемая информационная система «СкладPRO» предназначена для автоматизации процессов складского учета готовой продукции на промышленных предприятиях малого и среднего бизнеса. Система обеспечивает точный оперативный учет, контроль движения и остатков продукции, формирование первичной документации и базовую аналитику через веб-интерфейс.

## 3. Цели и задачи разработки

### 3.1. Цели:

- Создание централизованной, надежной системы учета готовой продукции.
- Автоматизация ключевых складских операций: приемки, размещения, инвентаризации, отгрузки.
- Обеспечение оперативного доступа к актуальной информации об остатках и движении товара.
- Повышение точности учета и снижение логистических издержек.

### 3.2. Задачи:

- Разработка структуры реляционной базы данных на MySQL, отражающей сущности предметной области.
- Создание интуитивно понятного веб-интерфейса для работы различных категорий пользователей (кладовщик, менеджер, руководитель).
- Реализация механизмов поиска, фильтрации и формирования отчетов по остаткам и обороту.
- Обеспечение безопасности данных, разграничения прав доступа и целостности информации.

## 4. Требования к системе

### 4.1. Функциональные требования:

#### 4.1.1. Управление номенклатурой (справочник продукции):

- Добавление, редактирование и архивация позиций готовой продукции.
- Поддержка иерархии категорий/групп товаров.
- Ведение характеристик: артикул, наименование, описание, единица измерения, штрих-код.
- Учет минимального/максимального запаса и срока годности.

#### 4.1.2. Управление складской структурой:

- Создание и редактирование схемы склада (зоны, ряды, стеллажи, ячейки).

- Присвоение характеристик складским местам (тип, грузоподъемность, габариты).
  - Визуализация занятости мест.
- 4.1.3. Учет партий и серийных номеров:
- Регистрация партий продукции при приемке с указанием даты производства, срока годности.
  - Учет продукции по уникальным серийным номерам (при необходимости).
  - Реализация принципов ротации запасов (FIFO, FEFO).
- 4.1.4. Документооборот и операции:
- Приемка: Формирование приходной накладной, привязка к партии, выбор ячейки размещения.
  - Внутреннее перемещение: Оформление перемещения товара между ячейками.
  - Отгрузка (отпуск): Формирование расходной накладной на основе заказа, резервирование товара, подбор по FIFO/FEFO.
  - Инвентаризация: Создание и проведение инвентаризационных описей, фиксация излишков и недостач.
- 4.1.5. Отчетность и аналитика:
- Оборотная ведомость по номенклатуре за период.
  - Отчет об остатках на складе (общий и с детализацией по ячейкам/партиям).
  - Карточка движения товара (полная история по конкретной позиции).
  - Дашборд с ключевыми показателями (общее кол-во позиций, суммарный остаток, оборот за период).
- 4.2. Нефункциональные требования:
- Время отклика на стандартные операции — не более 2 секунд.
  - Поддержка одновременной работы не менее 20 пользователей.
  - Кроссбраузерная совместимость (Chrome, Firefox, Edge последних версий).
  - Адаптивный интерфейс для работы на планшетных устройствах.
  - Ежедневное автоматическое резервное копирование базы данных.
5. Технические требования
- 5.1. Серверная часть:
- Сервер БД: MySQL 8.0+
  - Язык программирования: PHP 8.1+
  - Веб-сервер: Apache 2.4 / Nginx
- 5.2. Клиентская часть:
- HTML5, CSS3, JavaScript (ES6+)
  - Фреймворк для интерфейса: Bootstrap 5
6. Структура базы данных (основные таблицы)
- 6.1. products (Номенклатура продукции):
- id INT, PRIMARY KEY, AUTO\_INCREMENT

- article VARCHAR(50), NOT NULL, UNIQUE
- name VARCHAR(255), NOT NULL
- category\_id INT, FOREIGN KEY
- unit VARCHAR(20) NOT NULL
- description TEXT
- barcode VARCHAR(100)
- min\_stock DECIMAL(12,3)
- created\_at TIMESTAMP

#### 6.2. batches (Партии продукции):

- id INT, PRIMARY KEY, AUTO\_INCREMENT
- product\_id INT, FOREIGN KEY
- batch\_number VARCHAR(100), NOT NULL
- production\_date DATE
- expiry\_date DATE
- initial\_quantity DECIMAL(12,3)

#### 6.3. storage\_locations (Складские места):

- id INT, PRIMARY KEY, AUTO\_INCREMENT
- code VARCHAR(50), NOT NULL, UNIQUE (например, А-01-02)
- zone VARCHAR(100) (зона приемки, хранения, отгрузки)
- type VARCHAR(50) (стеллаж, полка, паллета)
- capacity DECIMAL(10,2)
- is\_active BOOLEAN DEFAULT TRUE

#### 6.4. inventory\_transactions (Операции движения / транзакции):

- id INT, PRIMARY KEY, AUTO\_INCREMENT
- product\_id INT, FOREIGN KEY
- batch\_id INT, FOREIGN KEY
- location\_id\_from INT, FOREIGN KEY (для перемещений и отгрузки)
- location\_id\_to INT, FOREIGN KEY (для приемки и перемещений)
- quantity DECIMAL(12,3), NOT NULL
- operation\_type ENUM('receipt', 'movement', 'shipment', 'inventory\_adjustment')
- document\_number VARCHAR(100) (номер накладной)
- user\_id INT, FOREIGN KEY
- created\_at TIMESTAMP

### 7. Интерфейс пользователя

#### 7.1. Основные экраны (разделы):

- Дашборд: Общая статистика и ключевые показатели.
- Справочники: Номенклатура, категории, складские места, контрагенты.
- Документы: Журналы приходных, расходных накладных, актов инвентаризации.
- Отчеты: Оборотные ведомости, остатки, движение товара.
- Администрирование: Пользователи, роли, настройки системы.

## 7.2. Требования к UI/UX:

- Интуитивно понятная навигация с четкой иерархией.
- Минимальное количество шагов для выполнения частых операций (быстрая приемка, отгрузка).
- Подтверждение для критических действий (списание, удаление).
- Валидация вводимых данных (проверка остатков, форматов).
- Единый современный стиль на основе Bootstrap 5.

## 8. Безопасность

### 8.1. Аутентификация и авторизация:

- Система ролей: «Кладовщик», «Менеджер», «Руководитель», «Администратор».
- Шифрование паролей с использованием современного алгоритма (bcrypt).
- Защищенные сессии с ограничением времени неактивности.

### 8.2. Защита данных и приложения:

- Защита от SQL-инъекций через использование подготовленных запросов (PDO).
- Защита от XSS (кросс-сайтового скрипирования) через экранирование вывода.
- Защита от CSRF (межсайтовой подделки запросов) с использованием токенов.
- Обязательная валидация и санация всех входных данных на стороне сервера.

## 9. Этапы разработки

### Этап 1. Подготовительный:

- Установка и настройка среды разработки (OpenServer/XAMPP, VS Code, Git).
- Детальное проектирование базы данных (ER-диаграмма, SQL-дамп).
- Создание базовой структуры проекта и архитектуры (MVC-подход).

### Этап 2. Разработка ядра и инфраструктуры:

- Реализация подключения к БД и базовых классов (моделей).
- Разработка системы аутентификации, авторизации и управления пользователями.
- Создание основного шаблона интерфейса (layout) с использованием Bootstrap.

### Этап 3. Реализация основной функциональности:

- Модуль справочников (номенклатура, складские места).
- Модуль документооборота (приемка, перемещение, отгрузка).
- Модуль инвентаризации.
- Модуль отчетности (остатки, оборотная ведомость).

#### Этап 4. Тестирование, отладка и документирование:

- Функциональное тестирование всех сценариев использования.
- Тестирование безопасности и нагрузки.
- Кросс-браузерное тестирование интерфейса.
- Написание руководства пользователя и технической документации.

#### 10. Критерии приемки

##### 10.1. Обязательные:

- Корректная работа всех CRUD-операций для основных сущностей.
- Безошибочное выполнение полного цикла: приемка → размещение → инвентаризация → отгрузка.
- Наличие работающей системы резервного копирования данных.
- Реализация защиты от основных веб-уязвимостей (SQLi, XSS, CSRF).
- Предоставление полной документации по установке и использованию.

##### 10.2. Дополнительные:

- Возможность импорта/экспорта данных (CSV, Excel).
- Наличие RESTful API для интеграции с внешними системами (1C, ERP).
- Генерация печатных форм документов (накладных) в формате PDF.

#### 11. Техническая документация

По окончании разработки должна быть предоставлена следующая документация:

- Руководство пользователя: Описание работы с интерфейсом для каждой роли.
- Руководство администратора: Инструкции по установке, настройке и поддержке системы.
- Описание базы данных: ER-диаграмма, дампы структуры, словарь данных.
- Краткое техническое описание: Используемый стек технологий, архитектура приложения.