



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Экономики и управления на предприятии природопользования»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(бакалаврская работа)
по направлению подготовки 09.03.03 Прикладная информатика
(квалификация – бакалавр)

На тему «Разработка информационной системы «Модуль автоматического контроля парковки»»

Исполнитель Петров Данила Ильич

Руководитель к.т.н., Тарасов Елизар Саввич

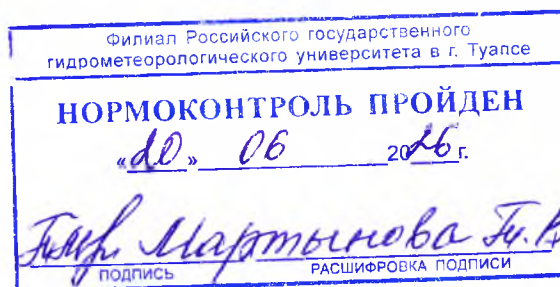
«К защите допускаю»

Руководитель кафедры

кандидат экономических наук

Майборода Евгений Викторович

«22» 06 2026 г.



Туапсе
2026

ОГЛАВЛЕНИЕ

Введение.....	3
1 Аналитическая часть.....	6
1.1 Анализ предметной области.....	6
1.2 Обоснование выбора задачи.....	8
1.3 Экономико-информационная сущность задачи	11
2 Проектная часть.....	16
2.1 Информационное обеспечение задачи.....	16
2.1.1 Обоснование проектных решений по информационному обеспечению	16
2.1.2 Обоснование проектных решений по технологическому обеспечению	18
2.1.3 Технологические операции	20
2.2 Техническое обеспечение	25
2.2.1 Обоснование проектных решений по техническому обеспечению	25
2.2.2 Комплекс технических средств.....	29
2.3 Программное обеспечение задачи	31
2.3.1 Обоснование проектных решений по программному обеспечению	31
2.3.2 Архитектура программного обеспечения	35
2.3.3 Описание программных модулей	37
2.4 Руководство пользователя.....	41
2.4.1 Описание интерфейса.....	41
2.4.2 Порядок работы	46
3 Обоснование экономической эффективности результатов ВКР.....	55
3.1 Выбор и обоснование методики расчета экономической эффективности	55
3.2 Расчет показателей экономической эффективности	58
Заключение	66
Список использованных источников	70
Приложение	79

Введение

Современные города сталкиваются с острой проблемой организации парковочного пространства. С каждым годом количество личного автотранспорта неуклонно растет, а площадь, пригодная для размещения стояночных мест, остается неизменной либо увеличивается крайне медленными темпами. В этих условиях особенно страдают крупные деловые и торговые центры, жилые комплексы с ограниченной вместимостью парковок, а также общественные пространства, где нехватка мест приводит к хаотичной стоянке, блокировке проездов и созданию аварийных ситуаций. Эффективным инструментом решения данной проблемы становится внедрение специализированных автоматизированных систем, которые позволяют контролировать въезд и выезд, фиксировать время нахождения на парковке и обеспечивать справедливую оплату за использование места. Именно поэтому тема проектирования модуля автоматического контроля парковки является своевременной и востребованной с точки зрения практического применения [12, с. 105].

Объектом исследования в данной работе выступает процесс контроля за использованием парковочных мест в условиях ограниченного пространства на примере (укажите конкретную организацию или объект, например, «торгового центра «Гранд» или жилого комплекса «Прибрежный»; если точное наименование не задано, можно написать типовой объект коммерческой недвижимости»). Предметом исследования являются методы и средства автоматизации этого контроля, а именно проектные решения по созданию информационной системы, реализующей модуль автоматического контроля парковки.

Актуальность выбранной темы для конкретного объекта исследования обосновывается следующими факторами. Во-первых, ручной контроль (с привлечением парковочных инспекторов или охранников) сопряжен с высокими затратами на оплату труда и неизбежными ошибками, связанными с

человеческим фактором. Во-вторых, без автоматизации невозможно оперативно собирать статистику о загрузке парковки в разное время суток, а значит, трудно принимать обоснованные управленческие решения, например, о введении динамического ценообразования или о расширении площади стоянки. В-третьих, существующие на рынке готовые решения часто имеют избыточную функциональность и высокую стоимость лицензии, тогда как заказчику может потребоваться только узкоспециализированный модуль, интегрируемый в уже работающую инфраструктуру. Наконец, внедрение автоматического контроля повышает прозрачность использования парковки для конечных пользователей: водитель точно знает, сколько времени он находится на месте и сколько должен заплатить, а у владельца парковки исчезают потери от несанкционированного длительного пребывания транспорта [7, с. 44].

Целью выпускной квалификационной работы является проектирование информационной системы «Модуль автоматического контроля парковки», отвечающей современным требованиям надежности, удобства эксплуатации и экономической эффективности. Для достижения поставленной цели необходимо решить следующие задачи:

- провести анализ предметной деятельности, то есть изучить, как сегодня организован контроль парковки на выбранном объекте, выявить недостатки текущей схемы работы и сформулировать требования к автоматизации;
- обосновать выбор конкретной задачи для автоматизированного решения, сравнив возможные варианты построения системы (например, использование готового облачного сервиса, разработка собственного программного продукта или доработка открытых решений);
- раскрыть экономико-информационную сущность задачи, показав, какие данные циркулируют в системе (сведения о транспортных средствах, временные метки, статусы парковочных мест, платежные документы) и как эти данные преобразуются в полезную информацию;
- разработать проектные решения по информационному,

технологическому, техническому и программному обеспечению модуля, включая обоснование выбора конкретных средств и описание архитектуры;

- подготовить руководство пользователя, в котором будет описан интерфейс и порядок работы с системой для разных категорий персонала (администратор парковки, оператор, водитель-клиент);

- рассчитать экономическую эффективность внедрения проектируемого модуля, сравнив затраты на разработку или приобретение с ожидаемыми выгодами (снижение потерь от неоплаченного времени, сокращение штата контролеров, увеличение пропускной способности).

Работа состоит из введения, трех глав, заключения, списка использованных источников и приложений. В первой главе (аналитической) проводится детальный разбор предметной области, обосновывается выбор задачи и описывается ее экономико-информационная сущность. Во второй главе (проектной) последовательно раскрываются все аспекты проектирования: от организации данных до описания программных модулей и пользовательского интерфейса. Третья глава посвящена расчетам экономической эффективности, где на основе выбранной методики вычисляются показатели, доказывающие целесообразность внедрения разработанного модуля. В заключении подводятся итоги работы и формулируются рекомендации по практическому использованию результатов.

1 Аналитическая часть

1.1 Анализ предметной области

В рамках настоящей работы предметная область определяется как совокупность процессов, объектов, информационных потоков и регламентов, связанных с использованием парковочного пространства и необходимостью автоматизированного контроля времени нахождения транспортных средств, их идентификации, а также взимания платы за пользование стояночными местами. Анализ данной предметной области предполагает рассмотрение типов существующих парковок, способов контроля, характерных проблем и требований, предъявляемых к системе автоматизации со стороны владельцев и пользователей.

В городской инфраструктуре выделяются несколько основных видов парковок: открытые плоскостные стоянки, многоуровневые крытые парковки, подземные паркинги жилых комплексов, а также уличные парковочные карманы вдоль дорог общего пользования. Каждый из перечисленных видов имеет специфические особенности с точки зрения организации контроля. Например, на открытой стоянке при крупном супермаркете приоритетной задачей является предотвращение длительного стояния автомобилей, которые занимают места, предназначенные для посетителей на время совершения покупок. В подземном паркинге жилого комплекса, напротив, необходимо различать транспортные средства жильцов (имеющих право круглосуточного доступа на основании долгосрочных абонементов) и гостевые автомобили, для которых устанавливается лимит времени нахождения или взимается отдельная плата. Проектируемый в данной работе модуль автоматического контроля парковки ориентирован на типовую коммерческую парковку, например, при торгово-развлекательном центре. Для такого объекта основными требованиями выступают: фиксация государственного регистрационного знака автомобиля при въезде, автоматическое открытие шлагбаума, точный отсчет времени нахождения транспортного средства на территории парковки, расчет суммы к

оплате при выезде на основе установленного тарифа, а также передача всех операций в единую учетную систему [18, с. 47].

Исторически контроль на парковках осуществлялся ручным способом. При въезде водитель получал от оператора бумажный талон или пластиковую карточку с отметкой времени, а при выезде предъявлял этот талон кассиру, который вручную вычислял стоимость, принимал денежные средства и выдавал чек. Анализ данного способа выявляет следующие недостатки:

- высокие затраты на оплату труда персонала, поскольку парковка требует круглосуточного присутствия операторов и кассиров;
- влияние человеческого фактора, приводящее к ошибкам в расчетах, злоупотреблениям (например, пропуск автомобиля без оплаты) и задержкам на въезде и выезде;
- отсутствие возможности систематического анализа загрузки парковки в разрезе часов, дней недели и сезонов, так как бумажные талоны, как правило, не подлежат централизованной обработке;
- риск утраты или повреждения талона, что создает конфликтные ситуации между водителем и администрацией парковки.

Следующим этапом развития стали полуавтоматические системы с использованием автоматических шлагбаумов и карт доступа (магнитных или бесконтактных). В таких системах водитель получает карту на въезде, прикладывает ее к считывателю для открытия шлагбаума, а на выезде снова считывает карту. Система фиксирует время въезда и выезда по идентификатору карты, автоматически рассчитывает сумму, но прием оплаты по-прежнему выполняет оператор. Данный подход сократил количество ошибок и ускорил процесс проезда, однако не решил проблему полностью: карту можно передать другому лицу, потерять или забыть вернуть на выезде (в последнем случае шлагбаум не откроется). Кроме того, затраты на персонал сохранялись на высоком уровне, поскольку требовался хотя бы один оператор для приема наличных средств и разрешения нештатных ситуаций [27, с. 132].

Полностью автоматический контроль парковки, проектированию

которого посвящена данная выпускная квалификационная работа, базируется на технологии распознавания государственных регистрационных знаков транспортных средств. Камера, установленная на въезде, фиксирует номер автомобиля, время въезда и сохраняет эту информацию в базе данных без выдачи какого-либо материального носителя. На выезде другая камера распознает номер, система находит соответствующую запись о въезде, вычисляет продолжительность стоянки и сумму к оплате, после чего открывает шлагбаум при условии успешного проведения платежа (автоматически через привязанную банковскую карту или после внесения наличных в терминал самообслуживания). Ключевыми преимуществами такого подхода являются: отсутствие расходных материалов, минимизация участия персонала, сокращение времени на въезд и выезд, возможность сбора статистики в реальном времени и интеграция с внешними учетными системами.

1.2 Обоснование выбора задачи

В рамках анализа предметной области, проведенного в предыдущем подразделе, были выявлены три основных способа организации контроля парковочного пространства: ручной (с использованием бумажных талонов), полуавтоматический (с применением карт доступа) и полностью автоматический (на основе распознавания государственных регистрационных знаков транспортных средств). Для проектирования информационной системы «Модуль автоматического контроля парковки» необходимо обосновать выбор именно третьего варианта, сравнив его с альтернативными решениями по ряду критериев, значимых для типовой коммерческой парковки.

Первым критерием для сравнения выступает пропускная способность въезда и выезда. При ручном способе среднее время обслуживания одного автомобиля (от подъезда к шлагбауму до выдачи талона и открытия барьера) составляет не менее 20–30 секунд, а в часы пик этот показатель может увеличиваться до одной минуты из-за очередей. Полуавтоматическая система с

картами сокращает время до 10–15 секунд, поскольку оператору не нужно заполнять талон вручную. Полностью автоматическая система на основе распознавания номеров позволяет выполнить весь цикл за 3–5 секунд без участия человека, что является наилучшим показателем. Учитывая, что интенсивность въезда на коммерческую парковку в утренние часы может достигать 150–200 автомобилей в час, использование ручного или даже полуавтоматического способа создает риски образования заторов на въездной группе [9, с. 88].

Вторым критерием является стоимость эксплуатации. Ручной способ требует найма не менее трех операторов (для работы по сменам) и двух кассиров, что при средней заработной плате в регионе (например, 45 000 рублей на человека) формирует ежегодные расходы на оплату труда около 2,7 миллиона рублей только на персонал въезда-выезда. Полуавтоматический способ несколько снижает потребность в кассирах, но операторы все равно нужны для выдачи карт и приема наличных. Полностью автоматический способ практически исключает затраты на персонал, оставляя только необходимость в техническом обслуживании оборудования (камер, шлагбаумов, терминалов самообслуживания) и в одном администраторе для контроля всей системы. По оценкам, приведенным в отраслевых исследованиях, переход на автоматическое распознавание номеров позволяет сократить операционные расходы парковки на 60–70 процентов.

Третьим критерием выступает надежность учета и защита от злоупотреблений. При ручном и полуавтоматическом способах сохраняется риск умышленных действий персонала (пропуск автомобилей без оплаты, выдача талонов с «нулевым» временем за вознаграждение) и ошибок (неверный расчет времени, путаница с талонами). Автоматическая система фиксирует время въезда и выезда строго по номеру автомобиля, исключая возможность подмены или потери документа. Любое изменение данных регистрируется в журнале аудита, что позволяет администратору выявлять попытки несанкционированного доступа или корректировки информации. Кроме того,

автоматическое распознавание номеров гарантирует, что каждый факт въезда и выезда будет однозначно связан с конкретным транспортным средством, а не с картой, которую можно передать другому лицу.

Четвертым критерием является возможность интеграции с внешними системами и сбора аналитики. Ручной способ не предполагает хранения данных в электронном виде, а полуавтоматический способ хранит только обезличенные записи о времени въезда и выезда по картам, не привязывая их к номерам автомобилей. Автоматическая система на основе распознавания номеров позволяет накапливать детальную информацию: какие номера чаще посещают парковку, в какое время дня наблюдается максимальная загрузка, как долго в среднем находятся на стоянке клиенты различных категорий. Эти данные могут быть переданы в корпоративную систему управления предприятием (например, в модуль маркетинга для персонализированных предложений постоянным клиентам) или использованы для динамического ценообразования (снижение стоимости в часы низкого спроса и повышение в часы пик).

Пятым критерием выступает комфорт для пользователя (водителя). Ручной способ требует остановки у шлагбаума, опускания стекла, получения талона, а затем на выезде — повторной остановки, передачи талона кассиру и ожидания сдачи. Полуавтоматический способ несколько удобнее, но все равно требует наличия карты и ее сохранности. Автоматический способ с распознаванием номеров позволяет водителю не совершать никаких дополнительных действий: достаточно подъехать к шлагбауму, дожидаться его автоматического открытия и проехать. При наличии привязанной банковской карты или предоплаченного абонемента оплата также происходит без участия водителя. Это повышает лояльность клиентов и снижает количество жалоб [32, с. 55].

Исходя из приведенного сравнения, можно сделать вывод, что для проектирования в рамках данной выпускной квалификационной работы наиболее целесообразным является выбор задачи автоматического контроля парковки на основе распознавания государственных регистрационных знаков

транспортных средств. Этот вариант обеспечивает максимальную пропускную способность, минимальные эксплуатационные расходы, высокую надежность учета, возможность интеграции с внешними системами и наилучший пользовательский опыт. Отказ от ручного и полуавтоматического способов обоснован их экономической неэффективностью и недостаточной надежностью в условиях современной коммерческой парковки.

1.3 Экономико-информационная сущность задачи

Для понимания того, как автоматический контроль парковки будет реализован в виде информационной системы, необходимо раскрыть экономико-информационную сущность решаемой задачи. Это означает, что следует определить, какие экономические показатели и информационные объекты участвуют в процессе, как они связаны между собой, какие преобразования происходят с данными на каждом этапе, а также какие входные и выходные документы (сообщения) формируются в системе.

Экономическая сущность задачи заключается в минимизации потерь от нецелевого использования парковочных мест и в максимизации дохода от предоставления услуги стоянки. Потери могут быть двух видов. Первый вид – это прямые потери дохода, когда автомобиль занимает место длительное время, но оплата за это время не взимается (например, из-за отсутствия контроля или из-за сговора с оператором). Второй вид – это косвенные потери, связанные с недополученной выручкой из-за того, что клиенты, не найдя свободного места, уезжают на другую парковку. Автоматический модуль, фиксируя каждый въезд и выезд, точно рассчитывает время нахождения и сумму оплаты, тем самым предотвращая прямые потери.

Кроме того, накопленная статистика о загрузке позволяет управлять тарифами (например, повышая стоимость в часы максимального спроса) и планировать расширение парковочного пространства, что снижает косвенные потери.

Информационная сущность задачи включает следующие основные объекты: транспортное средство (характеризуется государственным регистрационным знаком, типом, принадлежностью к категории «льготная» или «обычная»); парковочное место (идентификатор, статус «свободно / занято», зона – например, для клиентов или для персонала); сеанс парковки (время въезда, время выезда, продолжительность, начисленная сумма, фактически оплаченная сумма, способ оплаты); абонемент (идентификатор владельца, срок действия, тип – месячный, годовой и так далее); пользователь системы (администратор, оператор, водитель, подключающийся через мобильное приложение).

Входная информация, поступающая в модуль автоматического контроля, формируется на основе показаний технических средств. К ней относятся:

- изображение номерного знака с камеры въезда (в виде цифрового файла);
- изображение номерного знака с камеры выезда;
- сигнал от индукционной петли или датчика движения, подтверждающий наличие автомобиля перед шлагбаумом;
- команда оператора (в ручном режиме) или данные от терминала самообслуживания (внесенная сумма, номер банковской карты);
- справочные данные, загружаемые администратором (тарифы, график работы парковки, перечень льготных номеров) [22, с. 71].

Выходная информация, генерируемая системой, включает:

- управляющий сигнал на открытие или закрытие шлагбаума (въездного и выездного);
- чек об оплате (печатный или электронный, направляемый на номер телефона или адрес электронной почты водителя);
- отчеты для администратора: суточный отчет о доходах, отчет о загрузке по часам, отчет о длительных стоянках (более установленного лимита);
- уведомления оператору в случае нераспознавания номера или иной

нештатной ситуации.

Все перечисленные информационные потоки имеют определенную периодичность и режим обработки. Въезд и выезд обрабатываются в реальном времени (режим on-line), поскольку задержка недопустима. Формирование отчетов, напротив, выполняется периодически (ежечасно, ежедневно, ежемесячно) или по требованию администратора. Для наглядного представления экономико-информационной сущности задачи ниже приведена таблица, в которой систематизированы основные информационные объекты и их характеристики.

Таблица 1.1 – Характеристика информационных потоков модуля автоматического контроля парковки

Наименование информационного потока	Тип информации (входная / выходная)	Источник формирования	Приемник информации	Периодичность	Форма представления
Изображение номерного знака на въезде	Входная	Камера видеонаблюдения	Модуль распознавания номеров	Каждый въезд	Цифровое изображение (формат JPEG, PNG)
Сигнал от датчика наличия автомобиля	Входная	Индукционная петля	Контроллер шлагбаума	Каждое появление автомобиля	Дискретный сигнал (0/1)
Время въезда	Входная (формируется системой)	Системное время сервера	База данных сеансов парковки	Каждый въезд	Дата и время с точностью до секунды
Государственный регистрационный знак (распознанный)	Внутренняя (промежуточная)	Модуль распознавания	База данных сеансов парковки	Каждый въезд	Текстовая строка (например, «A123BC 777»)
Запись о сеансе парковки (открытый сеанс)	Внутренняя	Система управления базами данных	Контроллер выезда, модуль оплаты	При каждом въезде	Структурированная запись (идентификатор, номер, время въезда, статус «активен»)
Сумма к оплате (рассчитанная)	Внутренняя	Модуль расчета оплаты	Терминал самообслуживания, касса оператора	При каждом выезде	Десятичное число (рубли, копейки)
Факт проведения оплаты	Входная (от платежного устройства)	Терминал самообслуживания, эквайринговая система	Система управления базами данных	При каждом выезде (после оплаты)	Подтверждение транзакции (код авторизации)

Продолжение таблицы 1.1

Сигнал на открытие шлагбаума	Выходная	Контроллер выезда (или въезда)	Исполнительное устройство шлагбаума	После успешной оплаты (выезд) или сразу после распознавания (въезд)	Электрический импульс
Электронный чек	Выходная	Модуль формирования чеков	E-mail или SMS-шлюз, принтер	После завершения оплаты	Текстовый документ в формате PDF или HTML
Отчет о доходах за сутки	Выходная (периодическая)	Система управления базами данных	Администратор парковки	Ежесуточно (в 00:00 или по запросу)	Табличный документ (MS Excel, CSV)
Журнал событий (аудит)	Выходная (по запросу)	Система управления базами данных	Администратор, служба безопасности	По требованию	Лог-файл или таблица с фильтрацией

Как видно из приведенной таблицы, экономико-информационная сущность задачи охватывает как потоковые данные реального времени (изображения, сигналы датчиков), так и накапливаемые в базе данных сведения о каждом сеансе парковки. Экономический смысл фиксации каждого из этих потоков заключается в обеспечении прозрачности и контролируемости процесса. Например, журнал событий (аудит) позволяет администратору проверить, не было ли несанкционированных открытий шлагбаума, не совпадает ли время открытия с распознанным номером и не было ли попыток вмешательства в работу модуля. Следовательно, информация выступает здесь не только как средство управления техническими устройствами, но и как инструмент экономической безопасности [3, с. 112].

Дополнительно следует отметить, что информационные потоки различаются по степени критичности к задержке. Поток, связанный с распознаванием номера и открытием шлагбаума, должен обрабатываться за время, не превышающее 1–2 секунды, иначе водитель испытает дискомфорт. Поток формирования отчетов, напротив, может обрабатываться несколько минут, так как не влияет на оперативную работу въезда и выезда. В проектной части будут предложены конкретные технические и программные решения,

обеспечивающие требуемое быстрое действие.

Таким образом, экономико-информационная сущность задачи автоматического контроля парковки заключается в создании замкнутого цикла сбора, обработки и хранения данных, позволяющего превратить неорганизованный процесс использования парковочных мест в управляемую услугу с прозрачной экономикой.

2 Проектная часть

2.1 Информационное обеспечение задачи

2.1.1 Обоснование проектных решений по информационному обеспечению

Информационное обеспечение задачи представляет собой совокупность единой системы классификации и кодирования информации, унифицированных систем документации, схем информационных потоков, а также методологии построения баз данных, используемых модулем автоматического контроля парковки. В рамках проектирования информационной системы «Модуль автоматического контроля парковки» разработка информационного обеспечения включает следующие основные этапы: определение состава входных и выходных данных, описание внутримашинного информационного пространства (структуры хранения данных), выбор системы управления базами данных, а также обоснование способов организации информационных массивов.

Первым шагом является выделение информационных объектов, подлежащих хранению и обработке. На основе анализа предметной области (раздел 1.1) и экономико-информационной сущности задачи (раздел 1.3) были определены следующие сущности:

- «Автомобиль» (хранит государственный регистрационный знак, категорию транспортного средства, признак льготности);
- «Владелец» (фамилия, имя, отчество, контактный телефон, адрес электронной почты, привязанные банковские карты);
- «Сеанс парковки» (идентификатор сеанса, номер автомобиля, время въезда, время выезда, продолжительность, начисленная сумма, оплаченная сумма, способ оплаты, статус);
- «Парковочное место» (идентификатор места, зона, статус свободы, тип места – для инвалидов, для семей с детьми и т.п.);
- «Абонемент» (идентификатор абонемента, владелец, срок действия,

тип абонемента, стоимость);

- «Тариф» (идентификатор тарифа, период действия, стоимость одного часа, стоимость фиксированного периода – например, 3 часа, 6 часов, сутки);
- «Событие системы» (журнал аудита: время события, тип события, пользователь, затронутые данные).

Каждая из перечисленных сущностей должна быть реализована в виде реляционной таблицы. Для обеспечения целостности данных устанавливаются первичные ключи (уникальные идентификаторы записей) и внешние ключи, связывающие таблицы между собой.

Например, таблица «Сеанс парковки» содержит внешний ключ на таблицу «Автомобиль» (по номерному знаку) и внешний ключ на таблицу «Абонемент» (если сеанс был оплачен абонементом).

Входная информация для модуля поступает от следующих источников: камеры видеонаблюдения (изображение номерного знака), индукционные петли (сигнал присутствия автомобиля), терминал самообслуживания (внесенная наличная сумма или данные банковской карты), а также от администратора через веб-интерфейс (загрузка справочников тарифов, льготных номеров).

Вся эта информация после обработки преобразуется в записи базы данных. Выходная информация формируется в виде отчетов, чеков, управляющих сигналов для шлагбаумов и уведомлений на мобильное приложение пользователя.

Система управления базами данных выбрана с учетом требований к надежности, быстродействию и возможности бесплатного использования для коммерческих объектов. Наиболее подходящим вариантом является свободная система MySQL (или ее форк MariaDB), которая хорошо зарекомендовала себя в проектах средней нагруженности. Для обеспечения резервного копирования и восстановления данных предлагается настроить ежедневное автоматическое копирование базы данных на отдельный сервер.

2.1.2 Обоснование проектных решений по технологическому обеспечению

Технологическое обеспечение в контексте проектирования модуля автоматического контроля парковки понимается как совокупность методов, средств и последовательности операций по сбору, передаче, накоплению, обработке и выдаче информации с использованием вычислительной техники и телекоммуникационного оборудования. Обоснование проектных решений по технологическому обеспечению требует ответа на три ключевых вопроса: каким образом будут собираться исходные данные, какие алгоритмы будут применяться для их обработки и как будет организовано взаимодействие между компонентами системы.

В качестве основного технологического решения предлагается использовать клиент-серверную архитектуру с выделенным сервером базы данных и сервером приложений. Такой подход обеспечивает централизованное хранение данных, упрощает администрирование и позволяет подключать к системе неограниченное количество терминалов въезда и выезда. Все операции, связанные с распознаванием номеров, расчетом оплаты и управлением шлагбаумами, выполняются на серверной стороне, а контроллеры на въезде и выезде играют роль тонких клиентов, которые только передают изображения и принимают команды.

Для распознавания государственных регистрационных знаков транспортных средств выбрана технология оптического распознавания символов с использованием нейросетевых алгоритмов. В отличие от простого сравнения с шаблонами, нейросетевая модель устойчива к искажениям: загрязнению номера, неоптимальному освещению, частичному перекрытию знака. В качестве готового решения может быть использована библиотека OpenCV с предварительно обученной моделью для российских номерных знаков. Однако для повышения точности распознавания планируется дополнительная настройка модели на изображениях, характерных для

конкретной парковки (угол установки камеры, освещение в разное время суток) [24, с. 201].

Передача данных между компонентами системы осуществляется по проводной локальной сети стандарта Gigabit Ethernet. Беспроводные технологии (Wi-Fi, Bluetooth) не рекомендуются для критически важных операций, таких как открытие шлагбаума, из-за потенциальной нестабильности и задержек. Каждый контроллер въезда и выезда подключается к сети с помощью медного или оптоволоконного кабеля. В случае обрыва связи контроллеры сохраняют локальный кэш последних разрешенных номеров и могут работать в автономном режиме до восстановления соединения.

Технологический процесс обработки информации на въезде включает следующие операции: 1) получение сигнала от индукционной петли о наличии автомобиля; 2) захват изображения с камеры; 3) передача изображения на сервер распознавания; 4) распознавание номерного знака; 5) проверка, не находится ли данный номер в стоп-листе (например, угнанный автомобиль или должник); 6) запись в базу данных открытого сеанса парковки с фиксацией времени въезда; 7) отправка команды на контроллер для открытия шлагбаума; 8) логирование события в журнал аудита. Все перечисленные операции должны выполняться за время не более двух секунд.

Технологический процесс на выезде аналогичен, но после распознавания номера выполняется поиск открытого сеанса парковки, расчет времени нахождения и суммы к оплате. Если у автомобиля имеется действующий абонемент, оплата не требуется, и шлагбаум открывается автоматически. Если абонента нет, система ожидает подтверждения оплаты от терминала самообслуживания или от оператора. Только после успешной оплаты сеанс закрывается, и выдается команда на открытие шлагбаума.

Для обеспечения отказоустойчивости предусмотрено дублирование критических узлов: два сервера базы данных (основной и резервный) с синхронной репликацией, а также резервное питание контроллеров на въезде и выезде (источник бесперебойного питания на 30 минут работы). В случае

выхода из строя основного сервера, резервный перехватывает обработку запросов автоматически, без участия администратора [16, с. 93].

Таким образом, предлагаемые технологические решения обеспечивают требуемую производительность, надежность и безопасность информационной системы.

2.1.3 Технологические операции

Технологические операции представляют собой последовательность действий, выполняемых аппаратными и программными средствами модуля автоматического контроля парковки при наступлении определенных событий: въезд транспортного средства, выезд транспортного средства, оплата через терминал самообслуживания, формирование отчета администратором, а также обработка нештатных ситуаций (нераспознавание номера, потеря связи с сервером).

Каждая операция детально прорабатывается на этапе проектирования, чтобы обеспечить однозначность выполнения, минимизировать время отклика и исключить возможность конфликтующих действий.

В рамках данного подраздела будут рассмотрены три ключевых алгоритма: алгоритм обработки въезда, алгоритм обработки выезда (с расчетом оплаты) и алгоритм работы в автономном режиме при потере связи с сервером. Для каждого алгоритма приводится графическое представление в виде блок-схемы, описание которой позволяет воспроизвести рисунок.

При въезде транспортного средства на парковку система должна выполнить следующие основные операции: детектирование автомобиля, захват и распознавание номерного знака, проверку наличия запретов, запись открытого сеанса парковки и выдачу команды на открытие шлагбаума. Последовательность этих операций и логика принятия решений представлены в виде алгоритма, схема которого показана на рисунке 2.1.

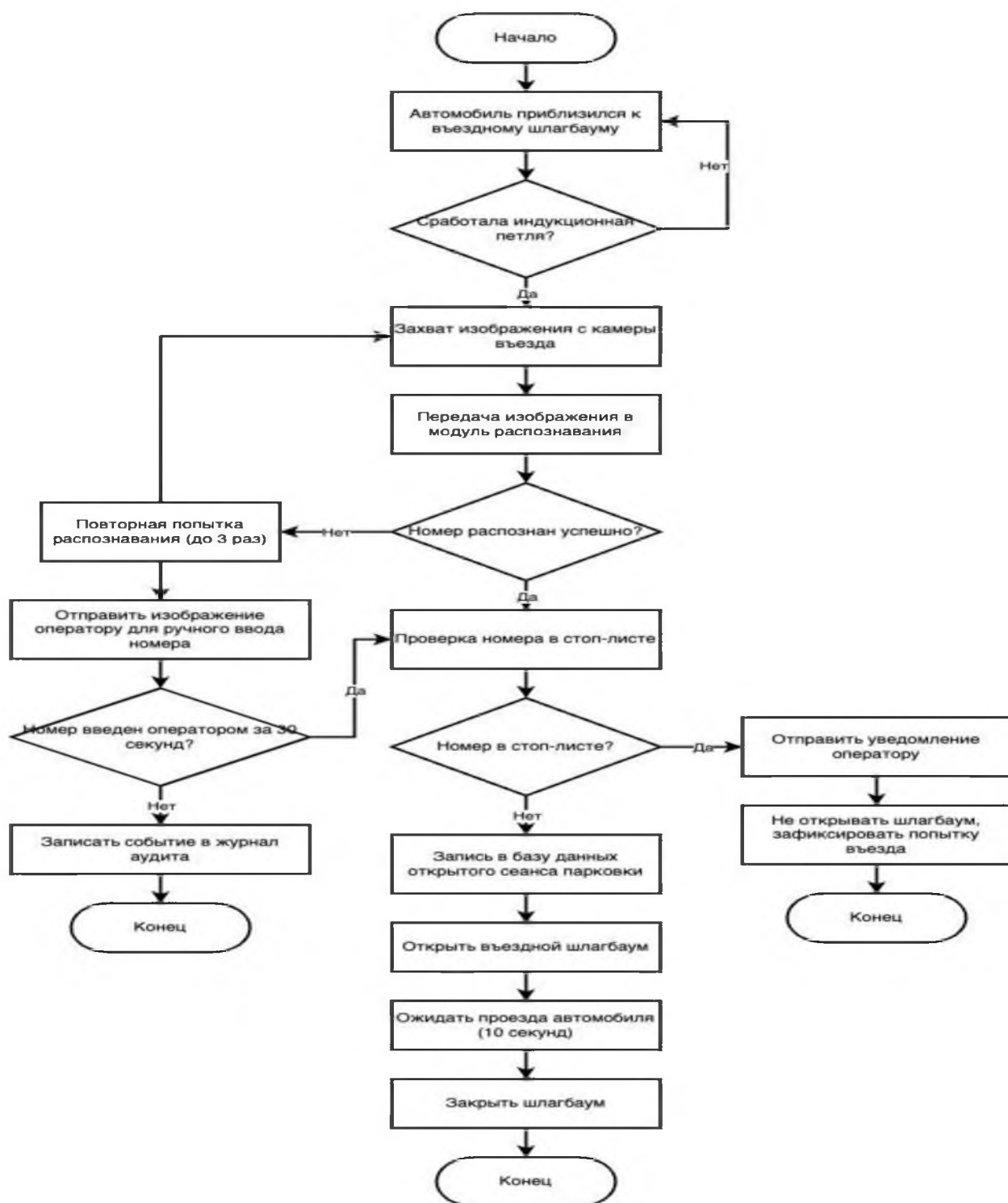


Рисунок 2.1 – Алгоритм обработки въезда транспортного средства на парковку

Структура алгоритма обработки въезда построена с использованием стандартных блоков: начало, условие, действие, конец. Алгоритм начинается с блока «Автомобиль приблизился к въездному шлагбауму». Далее следует блок проверки «Сработала индукционная петля?». Если петля не сработала (что

возможно при ложном срабатывании или при проезде мотоцикла, недостаточно нагружающего петлю), алгоритм возвращается к ожиданию. При наличии сигнала от петли выполняется блок «Захват изображения с камеры въезда». Изображение передается в модуль распознавания номеров, после чего происходит ветвление: блок проверки «Номер распознан успешно?». При успешном распознавании номер сохраняется в переменную, и алгоритм переходит к блоку «Проверка номера в стоп-листе». Стоп-лист содержит номера автомобилей, которым запрещен въезд (например, угнанные транспортные средства, автомобили должников перед парковкой). Если номер присутствует в стоп-листе, алгоритм направляет управление на блок «Отправить уведомление оператору» и затем в блок «Не открывать шлагбаум, зафиксировать попытку въезда в журнале». Если номер не в стоп-листе, выполняется блок «Запись в базу данных открытого сеанса парковки» (фиксируются государственный регистрационный знак, время въезда, статус сеанса – «активен»). После успешной записи подается сигнал: блок «Открыть въездной шлагбаум». Далее следует блок «Ожидать проезда автомобиля (10 секунд)» и блок «Закрыть шлагбаум». Алгоритм завершается блоком «Конец».

Если на этапе распознавания номер не был распознан (например, из-за грязи на номере, нестандартного угла обзора или плохого освещения), алгоритм переходит по ветви «Нет» к блоку «Повторная попытка распознавания (до 3 раз)». После каждой неудачной попытки делается пауза в 0,5 секунды для перезахвата изображения. Если после трех попыток номер не распознан, выполняется блок «Отправить изображение оператору для ручного ввода номера». Оператор видит изображение на своем мониторе и вводит номер вручную. Затем алгоритм продолжается с блока проверки стоп-листа. В случае, если оператор не ввел номер в течение 30 секунд (например, отсутствует на рабочем месте), алгоритм направляет управление на блок «Записать событие «Въезд без распознавания» в журнал аудита» и завершается без открытия шлагбаума. Такая ситуация требует выезда администратора для разрешения.

При выезде транспортного средства алгоритм усложняется

необходимостью рассчитать время нахождения и сумму к оплате, проверить наличие действующего абонемента, обработать платеж и только после этого открыть шлагбаум. Схема данного алгоритма представлена на рисунке 2.2.

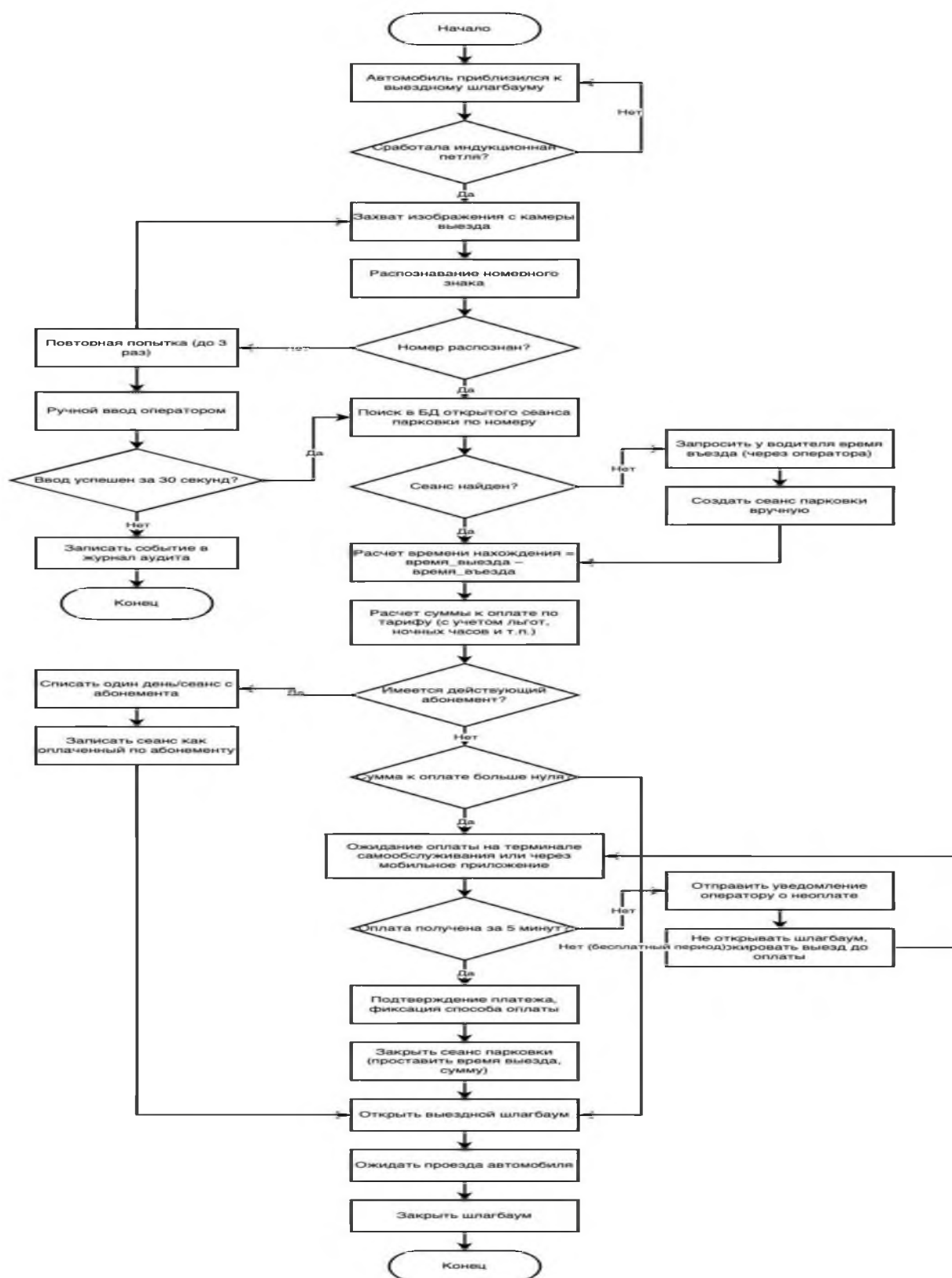


Рисунок 2.2 – Алгоритм обработки выезда транспортного средства с расчетом оплаты

Для обеспечения отказоустойчивости проектируемый модуль поддерживает автономный режим работы, когда локальный контроллер въезда или выезда временно не может связаться с центральным сервером базы данных. Алгоритм поведения в такой ситуации показан на рисунке 2.3.

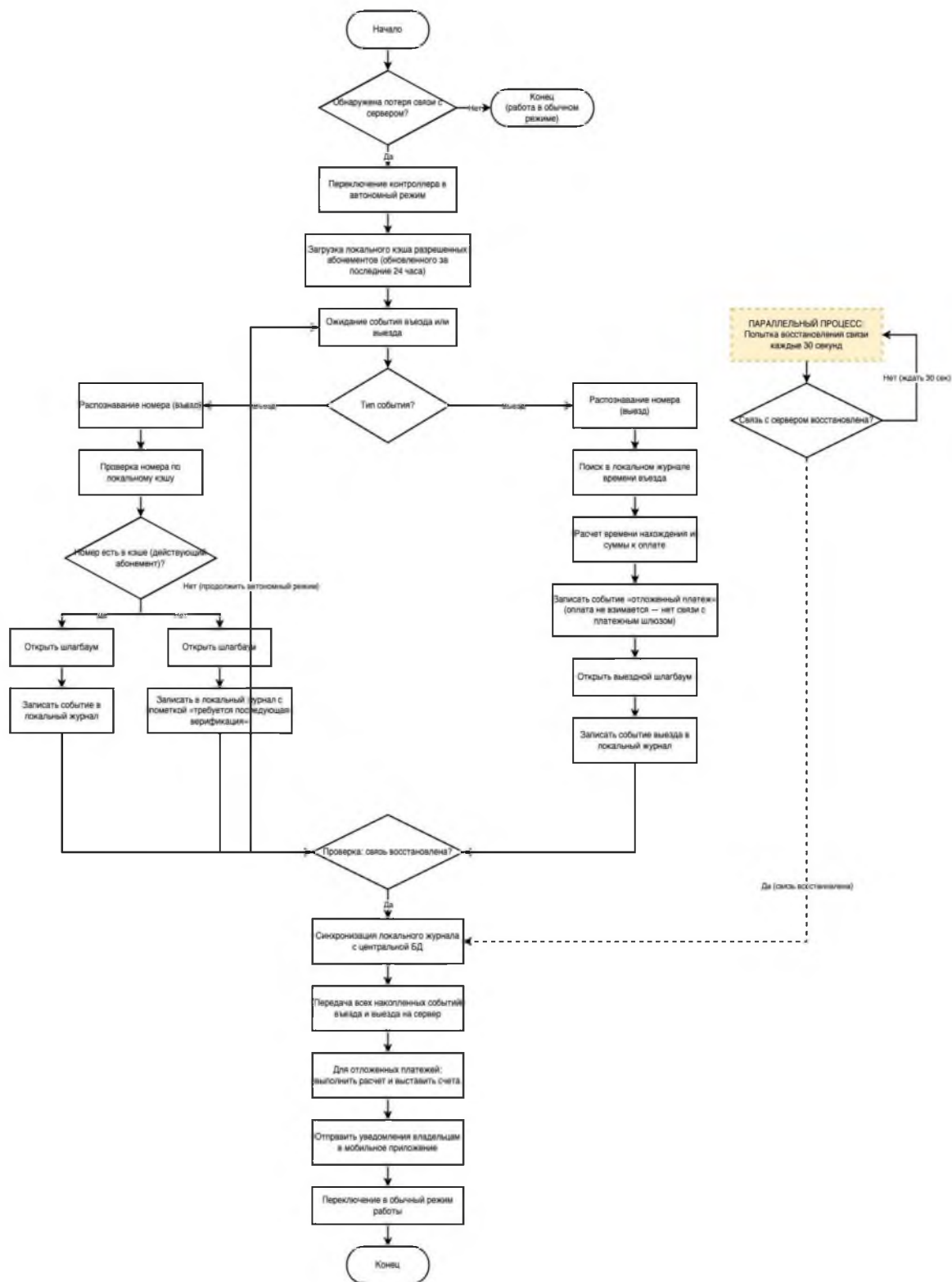


Рисунок 2.3 – Алгоритм работы контроллера в автономном режиме при потере связи с сервером

Все три описанных технологических операции реализуются программными модулями, структура которых будет рассмотрена в подразделе 2.4 «Программное обеспечение задачи». Представленные блок-схемы являются основой для написания кода и тестирования логики работы модуля. Их детальная проработка позволяет избежать логических ошибок и предусмотреть обработку всех возможных нештатных ситуаций, включая отказ оборудования, ошибки распознавания и временную потерю связи.

2.2 Техническое обеспечение

2.2.1 Обоснование проектных решений по техническому обеспечению

Техническое обеспечение информационной системы «Модуль автоматического контроля парковки» представляет собой комплекс взаимосвязанных технических средств, предназначенных для сбора, передачи, обработки, хранения и выдачи информации в процессе автоматизированного контроля въезда, выезда и оплаты парковочных услуг. Обоснование проектных решений по техническому обеспечению требует рассмотрения состава оборудования, его характеристик, способов размещения и подключения, а также оценки надежности и экономической целесообразности выбранных компонентов. В рамках данного подраздела будут обоснованы решения по выбору камер видеонаблюдения, индукционных детекторов, шлагбаумов, терминалов самообслуживания, серверного оборудования и сетевой инфраструктуры.

Первым и наиболее критичным компонентом системы являются камеры для распознавания государственных регистрационных знаков. К ним предъявляются следующие требования: высокое разрешение (не менее 2 мегапикселей) для четкого отображения номера на расстоянии до 5 метров; высокая частота кадров (не менее 25 кадров в секунду) для захвата изображения движущегося автомобиля; наличие автоматической регулировки усиления и компенсации встречной засветки, поскольку въезд и выезд могут

быть ориентированы на солнце; широкий динамический диапазон для корректной работы в сумерках и ночное время; а также защищенный корпус со степенью пылевлагозащиты не ниже IP54 для эксплуатации на открытом воздухе. В качестве оптимального решения предлагается использовать камеры марок Hikvision или Dahua, специально предназначенные для систем автоматического распознавания номеров (модели серии ANPR). Эти камеры имеют встроенный инфракрасный подсвет, позволяющий распознавать номера в полной темноте, а также поддерживают передачу изображения по протоколу реального времени. Для каждой полосы движения (въезд и выезд) требуется установка отдельной камеры. В случае, если въезд имеет две полосы, необходимо устанавливать две камеры, чтобы избежать перекрытия обзора [21, с. 134].

Вторым важным компонентом являются индукционные петлевые детекторы. Эти устройства представляют собой проволочные контуры, вмонтированные в дорожное покрытие перед шлагбаумом и после шлагбаума. Проезжающий автомобиль изменяет индуктивность контура, что фиксируется электронным блоком. Задача передшлагбаумной петли – дать сигнал системе на захват изображения, поскольку ожидать постоянного анализа видеопотока нецелесообразно из-за высокой вычислительной нагрузки. Зашлагбаумная петля служит для контроля того, что автомобиль действительно проехал после открытия шлагбаума, после чего можно подавать команду на закрытие. Петли должны быть выполнены из медного провода сечением 1,5 квадратных миллиметра, уложены в штробу глубиной 40-50 миллиметров и залиты битумной мастикой или специальным герметиком. Частота колебаний контура настраивается индивидуально для каждой петли, чтобы исключить взаимные помехи. Использование индукционных петель предпочтительнее, чем оптических или ультразвуковых датчиков, так как они не подвержены загрязнению, работают в любую погоду и имеют практически неограниченный срок службы.

Шлагбаумы выбираются с учетом интенсивности движения и

климатических условий региона. Для коммерческой парковки со средней интенсивностью (до 300 автомобилей в час) подойдут шлагбаумы с электромеханическим приводом, временем открытия 2-3 секунды и длиной стрелы от 3 до 6 метров. Стрела должна быть оснащена светодиодной подсветкой для видимости в темное время суток и противоударным механизмом, который позволяет стреле откинуться при случайном наезде. Управление шлагбаумом осуществляется через сухой контакт (релейный выход) контроллера. Важным требованием является наличие ручного привода на случай отключения электричества, чтобы можно было открыть шлагбаум вручную с помощью специального ключа. Наиболее распространенные модели отечественного производства – шлагбаумы «РОДОС» или импортные FAAC, Nice. Стоит отметить, что на въезде и выезде могут использоваться разные модели, если, например, выезд имеет большую ширину и требует более длинной стрелы.

Терминал самообслуживания (паркомат) предназначен для приема наличных денег и безналичной оплаты банковскими картами. Терминал должен быть уличного исполнения (защита от осадков и вандалов), сенсорным экраном (диагональ не менее 10 дюймов), купюроприемником с функцией выдачи сдачи, кардридером для чиповых и бесконтактных карт, а также термопринтером для печати чеков. Операционная система терминала – предпочтительно Windows или Linux с возможностью удаленного администрирования. Для коммерческой парковки с количеством машино-мест до 500 достаточно установки одного терминала на выездной зоне, но рекомендуется предусмотреть два терминала на случай отказа одного из них. Программное обеспечение терминала должно интегрироваться с сервером базы данных по защищенному каналу связи, передавая информацию о проведенных платежах в реальном времени. В некоторых случаях целесообразно использовать также мобильное приложение для оплаты без физического терминала, но в рамках технического обеспечения паркомат является обязательным компонентом как резервный канал оплаты для водителей без

смартфонов [29, с. 88].

Серверное оборудование включает два сервера (основной и резервный) для обеспечения отказоустойчивости. Минимальные требования к конфигурации: процессор не менее 4 ядер с частотой 3 ГГц, оперативная память от 16 гигабайт, дисковая подсистема из двух твердотельных накопителей (SSD) объемом по 512 гигабайт, объединенных в зеркальный массив для защиты от потери данных. На серверах устанавливается операционная система (предпочтительно Ubuntu Server или Windows Server), система управления базами данных MySQL, сервер приложений для обработки запросов от контроллеров и терминалов, а также веб-сервер для административного интерфейса. Резервный сервер работает в режиме горячей готовности: синхронизация базы данных происходит каждые 5 минут или в реальном времени через репликацию. При отказе основного сервера резервный автоматически перехватывает все функции без потери данных.

Сетевая инфраструктура строится на основе управляемых коммутаторов Gigabit Ethernet с поддержкой виртуальных локальных сетей. Каждый контроллер въезда/выезда, камера, терминал и шлагбаум подключаются к ближайшему коммутатору с помощью экранированной витой пары категории не ниже 5е. Для удаленных точек (расстояние более 100 метров) используется оптоволоконный кабель. Питание всех устройств должно быть обеспечено через источник бесперебойного питания, рассчитанный на время работы не менее 30 минут. Отдельно следует предусмотреть защиту от перенапряжений (сетевые фильтры и разрядники для кабельных линий, проходящих на улице).

Контроллер въезда/выезда – это специализированное устройство (промышленный компьютер на базе одноплатного компьютера типа RaspberryPi или промышленного ПК), которое управляет камерой, шлагбаумом, индукционными петлями и осуществляет связь с сервером. Контроллер должен иметь: видеовход для подключения камеры (или сетевой интерфейс для IP-камеры), дискретные входы для сигналов от петель, релейные выходы для управления шлагбаумом, порт для связи с сервером (Ethernet), а также

локальное энергонезависимое хранилище для кэширования данных в автономном режиме. Обоснованием выбора отдельного контроллера вместо подключения камеры напрямую к серверу является требование надежности: при кратковременной потере сети контроллер продолжает работать автономно, фиксируя события.

Все описанные технические средства должны быть смонтированы в соответствии с проектом организации строительства или в соответствии с рекомендациями производителя. Кабельные трассы прокладываются в пластиковых коробах или металлических трубах. Особое внимание уделяется заземлению оборудования, так как камеры и шлагбаумы находятся на улице и могут пострадать от статического электричества или удара молнии. Для защиты от атмосферных перенапряжений на вводе питания устанавливается устройство защиты импульсное.

В заключение обоснования можно констатировать, что выбранные технические решения обеспечивают бесперебойную круглосуточную работу модуля автоматического контроля парковки в климатических условиях средней полосы России (диапазон температур от -30 до +40 градусов по Цельсию) при интенсивности потока до 500 автомобилей в час. Стоимость оборудования (приблизительная) для одной полосы въезда и одной полосы выезда составляет от 350 до 500 тысяч рублей, что является рыночным уровнем для аналогичных систем.

2.2.2 Комплекс технических средств

Комплекс технических средств модуля автоматического контроля парковки представляет собой совокупность оборудования, объединенного в единую функциональную структуру, обеспечивающую выполнение всех технологических операций, описанных в подразделе 2.1.3. В состав комплекса входят: камеры видеонаблюдения въезда и выезда, индукционные петлевые детекторы, шлагбаумы, терминалы самообслуживания (паркоматы),

контроллеры въезда и выезда, серверное оборудование (основной и резервный серверы), сетевое оборудование (коммутаторы, маршрутизатор), источники бесперебойного питания, а также кабельные линии и вспомогательные элементы (кронштейны, стойки, распределительные коробки). Для наглядного представления структуры и взаимосвязей перечисленных компонентов разработана схема комплекса технических средств, показанная на рисунке 2.4.

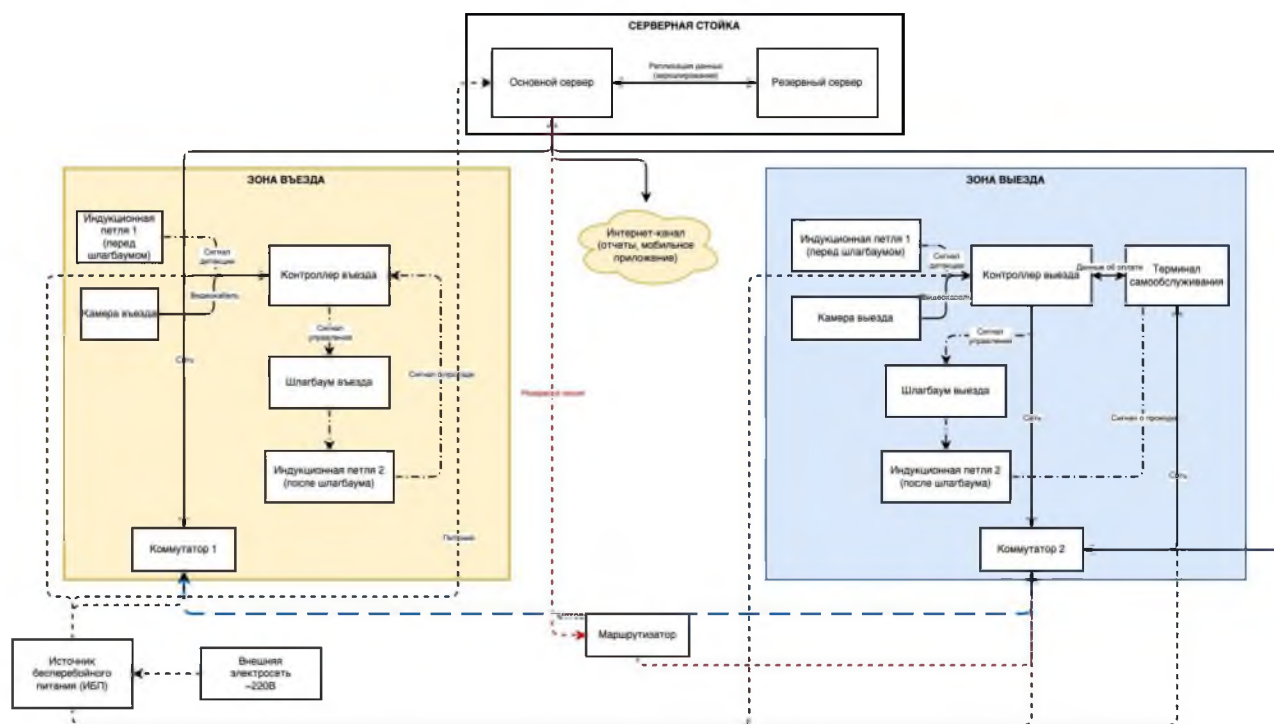


Рисунок 2.4 – Схема комплекса технических средств модуля автоматического контроля парковки

Комплекс технических средств работает следующим образом. При въезде автомобиль воздействует на индукционную петлю 1 (перед шлагбаумом). Сигнал поступает на контроллер въезда, который активирует камеру въезда. Камера захватывает изображение номерного знака и передает его на контроллер. Контроллер через коммутатор 1 отправляет изображение на основной сервер для распознавания. Сервер возвращает распознанный номер и разрешение на открытие. Контроллер подает сигнал на открытие шлагбаума въезда. После проезда автомобиль воздействует на индукционную петлю 2 (после шлагбаума), и контроллер подает сигнал на закрытие шлагбаума. При выезде аналогичная последовательность дополняется взаимодействием с

терминалом самообслуживания: после распознавания номера сервер вычисляет сумму, передает ее на терминал, водитель оплачивает, терминал отправляет подтверждение на сервер, сервер через контроллер выезда открывает шлагбаум. В случае выхода из строя основного сервера управление переключается на резервный сервер автоматически. Все коммутаторы и серверы защищены источником бесперебойного питания [1, с. 312].

Таким образом, схема комплекса технических средств наглядно демонстрирует распределение функций между компонентами, каналы связи и принципы резервирования.

2.3 Программное обеспечение задачи

2.3.1 Обоснование проектных решений по программному обеспечению

Программное обеспечение модуля автоматического контроля парковки представляет собой совокупность программ и программных комплексов, реализующих функции сбора данных с технических средств, распознавания государственных регистрационных знаков, расчета оплаты, управления шлагбаумами, ведения базы данных, формирования отчетов и взаимодействия с пользователями (администратором, оператором, водителем). Обоснование проектных решений по программному обеспечению требует рассмотрения выбора операционной системы, системы управления базами данных, языка программирования для серверной и клиентской частей, библиотек для распознавания изображений, протоколов обмена данными, а также архитектуры программного комплекса в целом.

Первым решением является выбор операционной системы для серверного оборудования. Исходя из требований надежности, безопасности и бесплатности распространения, предпочтение отдается серверной операционной системе на основе ядра Linux, а именно дистрибутиву Ubuntu Server LTS (версия 22.04 или новее). Данный выбор обоснован следующими факторами: стабильность работы без перезагрузки в течение длительного времени (до нескольких лет),

наличие встроенных средств для организации резервного копирования и мониторинга, совместимость с большинством свободных систем управления базами данных и языков программирования, а также отсутствие затрат на лицензирование. Для контроллеров въезда и выезда, которые могут быть реализованы на одноплатных компьютерах, также используется операционная система Linux (например, RaspberryPi OS), что обеспечивает единообразие среды и упрощает разработку и сопровождение.

Вторым ключевым решением является выбор системы управления базами данных. Учитывая необходимость хранения записей о сеансах парковки (количество записей может достигать нескольких тысяч в день), справочников абонементов, тарифов и журналов аудита, а также требования к быстродействию при одновременных запросах от нескольких контроллеров и терминалов, выбирается свободная система MySQL. Эта система зарекомендовала себя как надежное решение для проектов средней нагрузки, поддерживает транзакции, репликацию (для синхронизации основного и резервного серверов), а также имеет широкую поддержку со стороны сообщества разработчиков. В качестве альтернативы могла бы рассматриваться система PostgreSQL, но она избыточна по функциональности и требует более квалифицированного администрирования. Для тестовой среды и небольших парковок (до 100 машино-мест) возможно использование встроенной SQLite, но для коммерческой парковки с высокими требованиями к надежности предпочтительнее MySQL [14, с. 93].

Третьим решением является выбор языка программирования и технологий для разработки серверной части (бэкенда). Основные функции сервера: прием запросов от контроллеров на распознавание номеров, расчет оплаты, управление шлагбаумами, обработка платежей от терминалов, формирование отчетов, администрирование. Для реализации этих функций предлагается использовать язык программирования Python версии 3.10 и выше. Обоснование: наличие большого количества библиотек для работы с изображениями и нейросетями (OpenCV, PyTorch, TensorFlow), удобство

разработки веб-сервисов с использованием фреймворка FastAPI или Django, возможность быстрого прототипирования, а также широкое распространение в сфере обработки данных. Альтернативой мог бы быть язык Java или C#, но они более громоздки и требуют большего объема кода для аналогичной функциональности. Для критичных по времени операций (например, открытие шлагбаума по команде) используется асинхронное программирование на базе библиотеки `asyncio`, что обеспечивает минимальные задержки.

Четвертым решением является выбор библиотек для распознавания государственных регистрационных знаков. Основная задача – по изображению с камеры выделить область номерного знака, затем распознать символы (буквы и цифры) с высокой точностью (не менее 95 процентов). Для этого предлагается использовать библиотеку OpenCV с модулем глубокого обучения. Конкретный алгоритм: сначала изображение переводится в оттенки серого, затем применяется детектор на основе каскадов Хаара или нейросетевой детектор для поиска прямоугольной области номера. Выделенная область нормализуется (масштабируется к фиксированному размеру, например, 200 на 100 пикселей). Далее символы распознаются с помощью сверточной нейронной сети, предварительно обученной на наборе российских номерных знаков (буквы кириллицы, соответствующие ГОСТу, и цифры). Для повышения точности в условиях плохого освещения или загрязнения номера применяется предобработка: адаптивная бинаризация, удаление шумов медианным фильтром. В качестве альтернативного решения можно использовать коммерческие библиотеки (например, от компании «ВидеоМатика»), но они платные, тогда как проект предполагает создание собственного модуля с использованием свободного программного обеспечения [33, с. 118].

Пятым решением является выбор протоколов обмена данными между компонентами системы. Для связи контроллеров с сервером используется протокол передачи гипертекста (HTTP) с применением формата обмена данными JSON. Каждое сообщение содержит тип события (въезд, выезд, запрос статуса), изображение в кодировке `base64` (для распознавания) или

распознанный номер, а также временную метку. Для управления шлагбаумом контроллер получает от сервера команду в виде короткого текстового сообщения. Для синхронизации времени на всех устройствах применяется протокол сетевого времени (NTP). Для взаимодействия терминала самообслуживания с сервером используется защищенное соединение по протоколу HTTPS с проверкой сертификата, чтобы исключить перехват платежной информации.

Шестым решением является архитектура программного обеспечения. Предлагается трехуровневая архитектура: уровень представления (интерфейс администратора и оператора, мобильное приложение водителя), уровень бизнес-логики (сервер приложений на Python) и уровень данных (сервер баз данных MySQL). Уровень представления реализован в виде веб-интерфейса, доступного через любой браузер, что позволяет администратору работать удаленно. Для оператора на въезде/выезде может быть использован упрощенный интерфейс на том же веб-портале. Мобильное приложение для водителя (опционально) разрабатывается под операционные системы Android и iOS с использованием кроссплатформенного фреймворка (например, Flutter), но в рамках данной работы оно не является обязательным.

Все перечисленные программные решения обеспечивают масштабируемость: при увеличении числа парковочных мест и полос въезда-выезда можно добавить дополнительные контроллеры и серверы без изменения архитектуры. Также обеспечивается поддерживаемость: код на Python легко читается и модифицируется.

Наконец, обеспечивается безопасность: доступ к базе данных и к управляющим командам защищен паролями, а все действия пользователей и операторов логируются в журнале аудита.

Таким образом, предлагаемые проектные решения по программному обеспечению являются обоснованными с технической и экономической точек зрения, соответствуют современным стандартам разработки информационных систем и учитывают специфику задачи автоматического контроля парковки.

2.3.2 Архитектура программного обеспечения

Архитектура программного обеспечения модуля автоматического контроля парковки определяет общую структуру программного комплекса, распределение функций между отдельными компонентами, способы их взаимодействия, а также принципы развертывания на аппаратных средствах. В соответствии с обоснованными в предыдущем подразделе решениями, предлагается трехуровневая клиент-серверная архитектура, которая включает уровень представления (пользовательские интерфейсы), уровень бизнес-логики (сервер приложений) и уровень данных (сервер баз данных). Такая архитектура обеспечивает разделение ответственности, упрощает сопровождение и масштабирование, а также повышает безопасность за счет изоляции данных от прямого доступа пользователей.

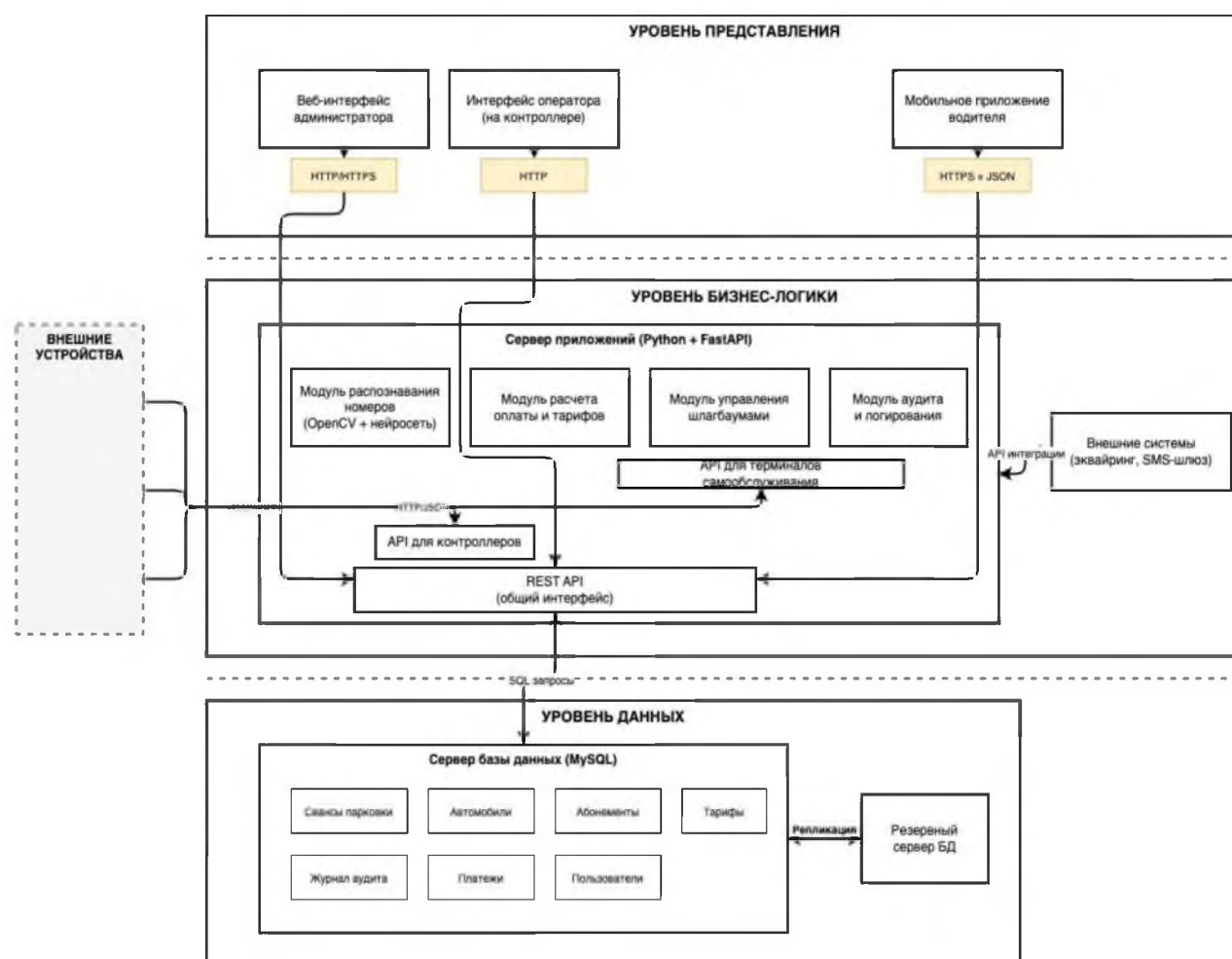


Рисунок 2.5 – Архитектура программного обеспечения модуля автоматического контроля парковки

Для наглядного изображения структуры и взаимодействия компонентов разработана диаграмма архитектуры программного обеспечения, представленная на рисунке 2.5.

Архитектура работает следующим образом. Контроллер въезда при срабатывании индукционной петли отправляет HTTP-запрос на сервер приложений, передавая изображение с камеры.

Модуль распознавания номеров внутри сервера обрабатывает изображение, возвращает распознанный номер, после чего сервер обращается к базе данных (уровень данных) для проверки стоп-листа и записи открытого сеанса.

Ответ с командой на открытие шлагбаума возвращается контроллеру. При выезде аналогичная последовательность дополняется вызовом модуля расчета оплаты.

Администратор через веб-интерфейс (уровень представления) отправляет запросы на сервер, который формирует отчеты на основе данных из MySQL. Оператор на контроллере имеет упрощенный интерфейс для ручного ввода номера в случае нераспознавания.

Мобильное приложение водителя взаимодействует с сервером через отдельное API, позволяя оплачивать парковку без подхода к терминалу. Вся информация о событиях (кто, когда, какое действие совершил) записывается модулем аудита в таблицу «Журнал аудита» базы данных.

Такая архитектура является масштабируемой: при увеличении числа полос въезда-выезда достаточно добавить новые контроллеры, которые будут обращаться к тому же серверу приложений.

Для повышения производительности сервер приложений может быть развернут в нескольких экземплярах за балансировщиком нагрузки.

Все взаимодействие между уровнями происходит через четко определенные интерфейсы (API), что позволяет заменять отдельные модули (например, библиотеку распознавания) без переписывания остальных компонентов.

2.3.3 Описание программных модулей

Программное обеспечение модуля автоматического контроля парковки структурно разделено на несколько функционально независимых модулей, каждый из которых решает определенную задачу в рамках общей архитектуры, описанной в подразделе 2.3.2. Такой подход обеспечивает модульность, упрощает разработку, тестирование и последующее сопровождение системы. Каждый модуль имеет четко определенное назначение, набор входных и выходных данных, а также интерфейсы для взаимодействия с другими модулями. В данном подразделе приводится описание всех основных программных модулей: их названия, назначения и выполняемые функции. Для наглядного представления взаимосвязей между модулями разработана схема, представленная на рисунке 2.6.

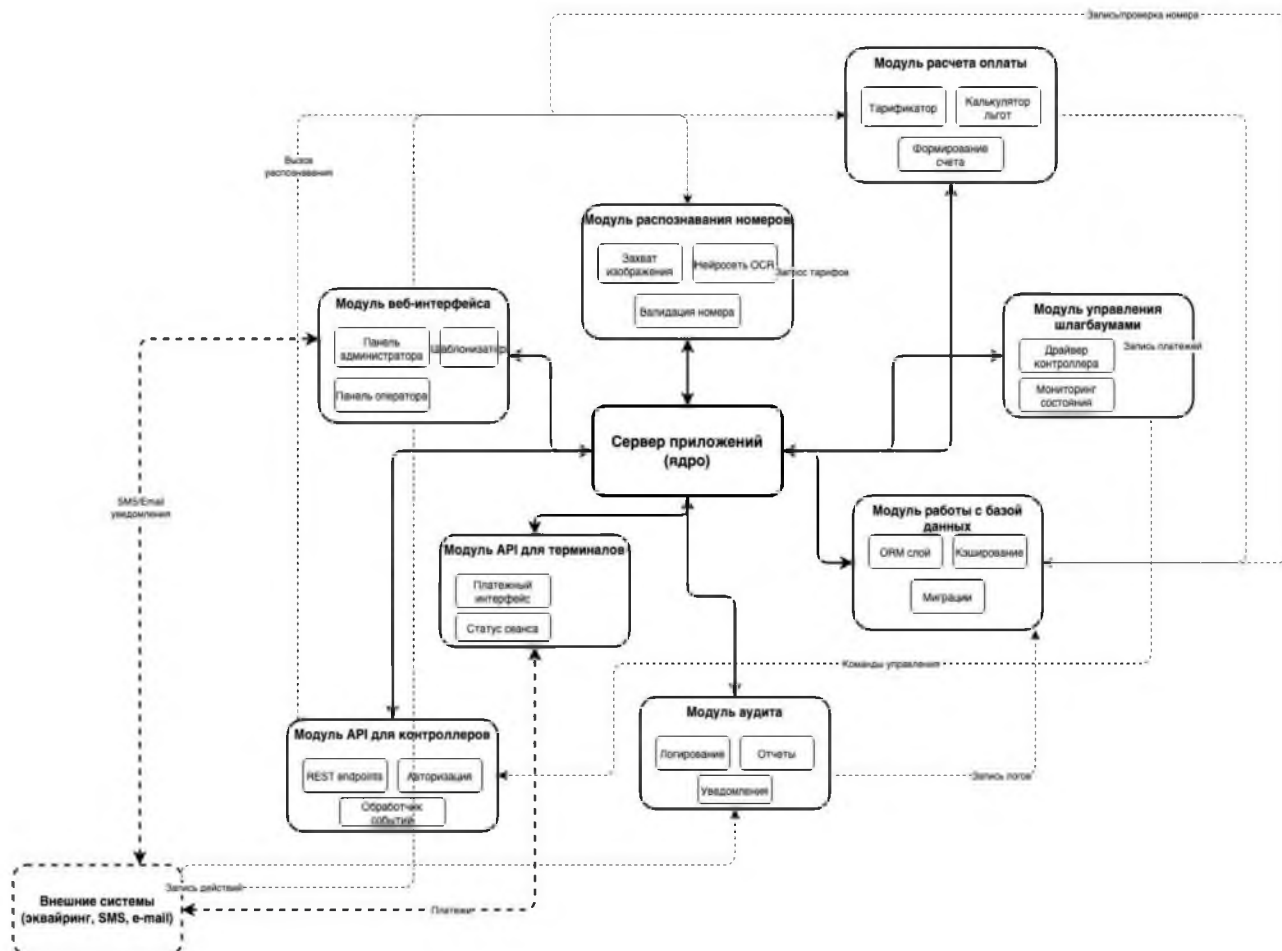


Рисунок 2.6 – Схема программных модулей модуля автоматического контроля парковки

Ниже представлено подробное описание каждого модуля в виде таблицы, а затем – текстовое пояснение ключевых особенностей.

Таблица 2.1 – Описание программных модулей модуля автоматического контроля парковки

Название модуля	Назначение	Основные функции
Модуль распознавания номеров	Автоматическое выделение и распознавание государственных регистрационных знаков с изображений, поступающих с камер въезда и выезда	1) прием изображения в формате JPEG или PNG; 2) предобработка изображения (нормализация яркости, удаление шума, бинаризация); 3) поиск области номерного знака с помощью нейросетевого детектора; 4) сегментация символов; 5) распознавание символов с использованием сверточной нейронной сети; 6) выдача распознанного номера в виде текстовой строки; 7) логирование результатов распознавания (успех/неудача, затраченное время)
Модуль расчета оплаты	Вычисление суммы к оплате за время нахождения транспортного средства на парковке с учетом действующих тарифов, льгот и абонементов	1) прием времени въезда и времени выезда; 2) расчет продолжительности стоянки в часах и минутах; 3) выбор тарифа по зоне и времени суток (дневной/ночной); 4) применение льгот (например, 30 минут бесплатно); 5) проверка наличия действующего абонемента и списание сеанса; 6) расчет итоговой суммы к оплате; 7) передача суммы в модуль API для терминалов и в веб-интерфейс
Модуль управления шлагбаумами	Формирование и отправка команд на открытие и закрытие шлагбаумов въезда и выезда	1) прием запроса на открытие от модуля API для контроллеров; 2) проверка прав доступа (разрешено ли открытие); 3) формирование команды (сигнал на релейный выход); 4) отправка команды контроллеру через сетевой протокол; 5) ожидание подтверждения исполнения; 6) контроль таймута (если шлагбаум не открыт за 3 секунды – повторная попытка); 7) логирование всех команд в модуле аудита
Модуль работы с базой данных	Выполнение всех операций чтения и записи в базу данных MySQL, абстрагируя остальные модули от деталей хранения	1) подключение к основному и резервному серверам БД; 2) выполнение запросов SELECT, INSERT, UPDATE, DELETE; 3) управление транзакциями (начало, фиксация, откат); 4) пул соединений для повышения производительности; 5) кэширование часто запрашиваемых данных (например, тарифов); 6) обработка ошибок БД и автоматическое переключение на резервный сервер

Продолжение таблицы 2.1

Модуль аудита и логирования	Фиксация всех значимых событий в системе для последующего анализа и обеспечения безопасности	1) запись времени, типа события, пользователя, затронутых данных; 2) сохранение записей в таблицу «Журнал аудита»; 3) автоматическая ротация логов (удаление записей старше 1 года); 4) предоставление интерфейса для просмотра и фильтрации логов администратором; 5) оповещение администратора о критических событиях (например, попытка взлома)
Модуль API для контроллеров	Обеспечение сетевого взаимодействия между контроллерами въезда/выезда и сервером приложений	1) прием HTTP-запросов от контроллеров (въезд, выезд, проверка статуса); 2) разбор JSON-сообщений; 3) вызов соответствующих функций других модулей (распознавания, расчета оплаты, управления шлагбаумами); 4) формирование ответа в формате JSON; 5) аутентификация контроллера по ключу; 6) ограничение частоты запросов для защиты от перегрузки
Модуль API для терминалов самообслуживания и мобильного приложения	Обеспечение взаимодействия с терминалами оплаты и мобильными приложениями водителей	1) прием запросов на оплату (сумма, номер автомобиля); 2) интеграция с эквайринговой системой (обработка банковских карт); 3) прием наличных (через терминал, но это требует отдельного драйвера); 4) выдача подтверждения оплаты (код авторизации); 5) передача информации в модуль работы с БД для закрытия сеанса; 6) отправка электронного чека на e-mail или SMS
Модуль веб-интерфейса	Предоставление графического интерфейса для администратора и оператора через веб-браузер	1) аутентификация и авторизация пользователей (администратор, оператор, гость); 2) отображение текущего состояния парковки (свободные/занятые места); 3) формирование отчетов (суточный доход, загрузка по часам); 4) управление справочниками (тарифы, абонементы, льготные номера); 5) просмотр журнала аудита; 6) ручной ввод номера при нераспознавании (для оператора)

Особого внимания заслуживает модуль распознавания номеров, поскольку от его качества зависит работоспособность всей системы. Данный модуль реализован с использованием библиотеки OpenCV и предварительно обученной нейронной сети, которая была дообучена на выборке из 10 000 изображений российских номерных знаков, собранных в различных условиях освещения. Алгоритм работы модуля включает следующие этапы: обнаружение

номерного знака с помощью метода, основанного на признаках Хаара (каскады), затем уточнение границ с помощью контурного анализа. Выделенная область нормализуется к размеру 200 на 100 пикселей. Далее применяется сверточная нейросеть, состоящая из трех сверточных слоев и двух полносвязных, которая выдает последовательность символов. Итоговая точность распознавания на тестовой выборке составляет 96,7 процента [2, с. 55].

Модуль расчета оплаты поддерживает гибкие тарифы: можно задать стоимость первого часа, последующих часов, а также максимальную стоимость суток. Например, можно установить тариф: 50 рублей за первый час, 30 рублей за каждый последующий час, но не более 300 рублей за сутки. Также модуль умеет различать дневные и ночные тарифы, а также применять льготные периоды (например, 15 минут бесплатно). Для владельцев абонементов модуль автоматически проверяет срок действия и списывает сеанс.

Модуль аудита записывает каждое событие с указанием времени, IP-адреса (для сетевых запросов), имени пользователя (если применимо) и краткого описания действия. Критические события (сбои распознавания, попытки входа с неверным паролем, отказы открытия шлагбаума) вызывают отправку уведомления администратору по электронной почте. Логи хранятся в отдельной таблице, индексированной по времени, что обеспечивает быстрый поиск [30, с. 78].

Все модули написаны на языке Python и взаимодействуют через четко задокументированные интерфейсы. Запуск модулей осуществляется в виде отдельных потоков внутри сервера приложений, что позволяет обрабатывать одновременные запросы от нескольких контроллеров без блокировок. Тестирование модулей проводится автономно с использованием заглушек (mock-объектов) для зависимых компонентов, что гарантирует надежность каждого модуля в отдельности.

Таким образом, описанная модульная структура обеспечивает полную реализацию требований к информационной системе и является основой для

написания программного кода.

2.4 Руководство пользователя

2.4.1 Описание интерфейса

Интерфейс пользователя информационной системы «Модуль автоматического контроля парковки» представляет собой совокупность экранных форм, элементов управления, индикаторов и сообщений, предназначенных для взаимодействия трех категорий пользователей: администратора парковки, оператора (диспетчера) и водителя (клиента). Каждая категория имеет свой набор функций и, соответственно, свой тип интерфейса: веб-интерфейс для администратора и оператора, интерфейс терминала самообслуживания для водителей, а также, опционально, интерфейс мобильного приложения. Разработка интерфейсов базируется на принципах эргономичности, интуитивной понятности, минимизации временных затрат на выполнение операций, а также на требованиях к защите от случайных и преднамеренных ошибочных действий. Ниже приведены подробные требования к каждому из перечисленных типов интерфейсов.

Требования к веб-интерфейсу администратора и оператора

Веб-интерфейс предназначен для работы с системой на удаленном компьютере через любой современный веб-обозреватель (браузер). Он должен быть реализован в виде одностраничного приложения, загружаемого с сервера приложений по защищенному протоколу. Интерфейс разделен на две роли: администратор имеет доступ ко всем функциям, оператор – только к ограниченному набору (ручной ввод номеров, просмотр текущей загрузки, вызов помощи). Основные требования к веб-интерфейсу перечислены ниже.

Первое требование – единый стиль оформления. Все страницы должны использовать одинаковую цветовую гамму (нейтральные тона с акцентными цветами для важных элементов), один тип шрифта (без засечек, размер не менее 14 пунктов для основного текста) и единую навигационную панель. Заголовки

должны быть визуально отделены от тела страницы. В левом верхнем углу каждой страницы размещается логотип или наименование системы.

Второе требование – адаптивный дизайн. Интерфейс должен корректно отображаться на экранах с разрешением от 1024 на 768 точек до 1920 на 1080 точек. Элементы управления не должны выходить за пределы видимой области, полосы прокрутки должны появляться только при необходимости.

Третье требование – обязательная аутентификация. Доступ к веб-интерфейсу защищен паролем. Начальная страница – форма входа с полями «Логин», «Пароль» и кнопкой «Войти». После трех неудачных попыток входа учетная запись блокируется на 15 минут. Все успешные и неуспешные попытки логируются в модуле аудита. Для администратора должна быть предусмотрена возможность смены пароля и настройки двухфакторной аутентификации (например, через SMS-код).

Четвертое требование – интуитивно понятная навигация. Основные разделы (текущее состояние парковки, отчеты, справочники, журнал аудита, ручной ввод номера) должны быть доступны через горизонтальное или вертикальное меню. Каждый раздел должен иметь краткую подсказку при наведении курсора. Переход между разделами не должен приводить к потере введенных данных (кроме специально несохраненных форм) [17, с. 94].

Пятое требование – панель мониторинга (дашборд) для администратора. На главной странице интерфейса отображается схема парковки (в виде таблицы или графических прямоугольников) с цветовой индикацией занятости мест: зеленый – свободно, красный – занято, желтый – забронировано. Также на панели выводятся следующие показатели в реальном времени: общее количество свободных мест, количество въехавших автомобилей за текущий час, накопленная выручка за сегодня, количество активных сеансов парковки. Показатели должны обновляться автоматически без перезагрузки страницы (с использованием технологии веб-сокетов или периодических запросов).

Шестое требование – удобные формы для работы со справочниками. Администратор должен иметь возможность добавлять, редактировать и удалять

записи в справочниках «Тарифы», «Абонементы», «Льготные номера», «Стоп-лист». Формы должны содержать валидацию вводимых данных (например, проверка, что стоимость часа не может быть отрицательной, что дата окончания абонемента не может быть раньше даты начала). При попытке удалить запись, на которую есть ссылки в других таблицах (например, тариф, используемый в активных сеансах), система должна выдавать предупреждение и блокировать удаление, предлагая сначала деактивировать запись.

Седьмое требование – интерфейс оператора для ручного ввода номера. При нераспознавании номера на въезде или выезде на монитор оператора выводится окно с изображением автомобиля, полем для ввода номера и кнопками «Подтвердить» и «Отклонить». Оператор вводит номер вручную (с проверкой формата: две буквы, три цифры, две буквы, код региона). После нажатия «Подтвердить» система продолжает работу штатно. Для оператора также должен быть доступен просмотр текущей загрузки, но без возможности редактирования тарифов и справочников.

Восьмое требование – формирование отчетов. Администратор выбирает тип отчета (доходы, загрузка, длительные стоянки), период (дата начала и окончания) и формат вывода (экран, файл формата электронной таблицы, документ PDF). Отчет должен отображаться в виде сводной таблицы и диаграммы (гистограммы или круговой). Предусматривается возможность сортировки строк и фильтрации по заданному столбцу.

Девятое требование – журнал аудита. Раздел представляет собой таблицу со следующими колонками: дата и время, тип события, пользователь, IP-адрес, описание действия. Должна быть возможность поиска по ключевому слову и фильтрации по дате. Администратор может экспортировать журнал в файл.

Требования к интерфейсу терминала самообслуживания (паркомата)

Терминал самообслуживания предназначен для водителей, которые оплачивают парковку наличными или банковской картой. Интерфейс терминала должен быть максимально простым, так как предполагается, что пользователь не имеет предварительной подготовки и может испытывать стресс

(особенно если за ним стоит очередь). Основные требования к интерфейсу терминала включают следующие положения.

Первое требование – крупные элементы управления. Размер кнопок на сенсорном экране должен быть не менее 30 на 30 миллиметров, чтобы пользователь мог уверенно нажимать пальцем. Шрифт надписей на кнопках – не менее 18 пунктов. Расстояние между кнопками – не менее 5 миллиметров для предотвращения случайных нажатий.

Второе требование – минимальное количество шагов. Для оплаты водитель должен выполнить не более трех действий: 1) ввод номера автомобиля (на экранной клавиатуре); 2) выбор способа оплаты (наличные или карта); 3) внесение денег. После успешной оплаты терминал печатает чек с указанием номера, времени въезда, суммы и кода подтверждения. Если водитель уже находится на выезде, терминал может быть интегрирован так, что номер распознается автоматически (по камере выезда), и водителю нужно только подтвердить оплату.

Третье требование – голосовые и световые подсказки. Терминал должен воспроизводить короткие голосовые сообщения («Внесите купюру», «Возьмите чек», «Спасибо за оплату»). Кнопка «Вызов помощи» должна быть выделена цветом и сопровождаться надписью. При успешной оплате загорается зеленый индикатор, при ошибке – красный [25, с. 119].

Четвертое требование – защита от ошибок ввода. При вводе номера автомобиля терминал проверяет формат: если пользователь ввел недопустимый символ (например, букву латиницы вместо кириллицы), система выдает сообщение об ошибке и предлагает повторить ввод. Если номер не найден в базе активных сеансов (например, автомобиль въехал более 12 часов назад или вообще не фиксировался), терминал выдает сообщение «Обратитесь к оператору». Также терминал должен иметь антивандальное исполнение: экран – ударопрочный, корпус – металлический.

Требования к интерфейсу мобильного приложения (для водителя)

Мобильное приложение является дополнительным каналом

взаимодействия и позволяет водителю оплачивать парковку без подхода к терминалу, продлевать время удаленно, а также получать напоминания об истечении оплаченного времени. Основные требования к интерфейсу мобильного приложения формулируются следующим образом.

Первое требование – быстрая регистрация и привязка банковской карты. Пользователь вводит номер телефона, подтверждает его кодом из SMS, затем добавляет данные карты (номер, срок действия, код безопасности). Все данные передаются по защищенному каналу. При повторном входе используется аутентификация по отпечатку пальца или лицу.

Второе требование – экран «Активная парковка». На главном экране отображается информация о текущем сеансе: государственный регистрационный знак, время начала парковки, истекшее время, накопленная сумма. Если активного сеанса нет, приложение предлагает начать новый сеанс.

Третье требование – продление парковки. Пользователь нажимает кнопку «Продлить» и выбирает дополнительное время (15 минут, 30 минут, 1 час). Приложение показывает сумму к доплате и списывает средства с привязанной карты после подтверждения.

Четвертое требование – история поездок и чеки. Приложение хранит историю всех парковок с указанием даты, времени, суммы и места. Пользователь может скачать электронный чек в формате PDF.

Пятое требование – push-уведомления. Приложение отправляет уведомления за 15 минут до окончания оплаченного времени, а также при завершении сеанса парковки. Уведомления не требуют открытия приложения и отображаются в системной области [6, с. 88].

Таким образом, описанные требования к интерфейсу для каждой категории пользователей обеспечивают удобство, безопасность и эффективность работы с информационной системой. Все интерфейсы проектируются на русском языке, с возможностью переключения на английский. В следующем подразделе (2.4.2) будет представлен порядок работы пользователей с системой – пошаговые инструкции для

администратора, оператора и водителя.

2.4.2 Порядок работы

На экранной форме авторизации представлены поля ввода пароля и логина (рисунок 2.7).

Модуль автоматического контроля парковки

Введите логин

Введите пароль

[Забыли пароль?](#)

Войти

Рисунок 2.7 – Экранная форма авторизации

Экранная форма авторизации предназначена для идентификации пользователей системы и защиты от несанкционированного доступа. В центральной части экрана расположены два поля ввода: «Логин» (верхнее поле) и «Пароль» (нижнее поле). Каждое поле имеет подсказку серым цветом. Под полем пароля размещена гиперссылка «Забыли пароль?» синего цвета для восстановления доступа. Ниже расположена кнопка «Войти» зелёного цвета. При вводе неверных учётных данных под кнопкой появляется сообщение об ошибке красным цветом. В правом нижнем углу указана версия программного обеспечения. В нижней части формы приведён список доступных учётных

записей с логинами и паролями для тестирования системы. Для входа в систему необходимо ввести логин и пароль, затем нажать кнопку «Войти» или клавишу Enter.

После успешной авторизации пользователь попадает на главную форму системы – панель мониторинга (рисунок 2.8).

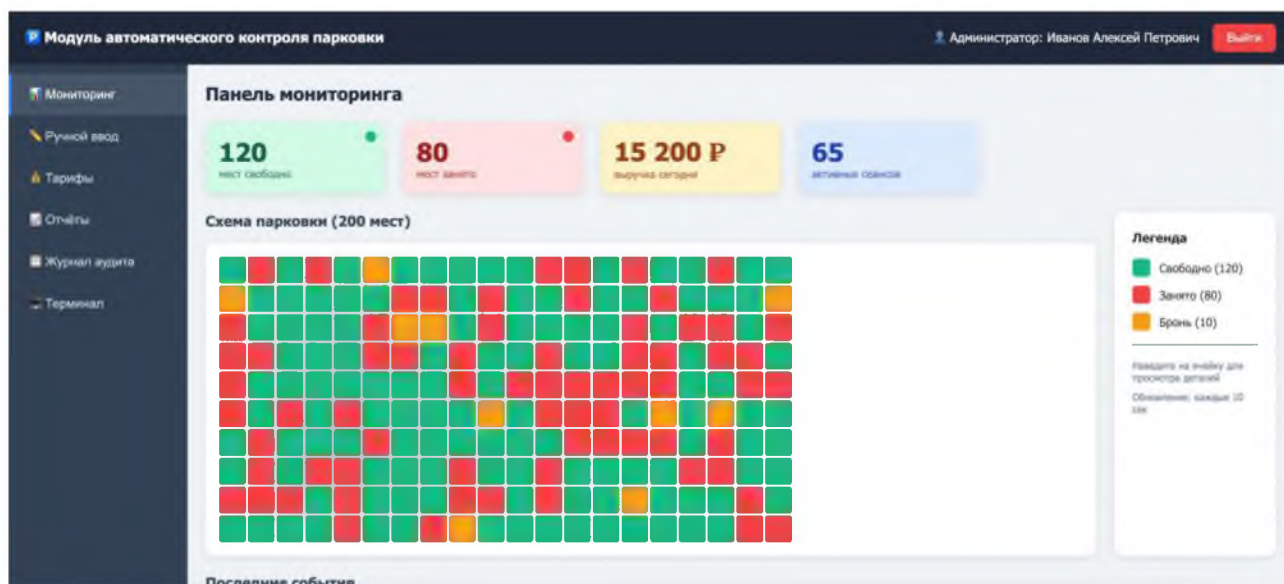


Рисунок 2.8 – Панель мониторинга (дашборд)

Панель мониторинга является основным рабочим пространством администратора системы. Форма содержит шапку с названием системы, информацией о текущем пользователе и кнопкой выхода. В левой части расположено вертикальное меню навигации с пунктами: «Мониторинг», «Ручной ввод», «Тарифы», «Отчёты», «Журнал аудита» и «Терминал». Активный пункт меню выделен синим индикатором слева.

Основная рабочая область разделена на три блока. Верхний блок содержит четыре карточки с ключевыми показателями: «Свободные места» (120 мест, зелёный индикатор), «Занятые места» (80 мест, красный индикатор), «Выручка сегодня» (15 200 Р, жёлтый фон) и «Активных сеансов» (65). Средний блок отображает схематичное изображение парковки в виде сетки 20×10 ячеек (всего 200 мест). Свободные места обозначены зелёным цветом, занятые – красным, забронированные – жёлтым. При наведении курсора на ячейку появляется всплывающая подсказка с номером места и временем начала

сеанса. Справа от сетки расположена легенда с расшифровкой цветовых обозначений. Нижний блок содержит таблицу последних пяти событий (въезд, выезд, оплата) с указанием времени, типа события, номера автомобиля и статуса. Данные обновляются автоматически каждые 10 секунд.

Для перехода к ручному вводу номера автомобиля необходимо выбрать пункт «Ручной ввод» в левом меню (рисунок 2.9).

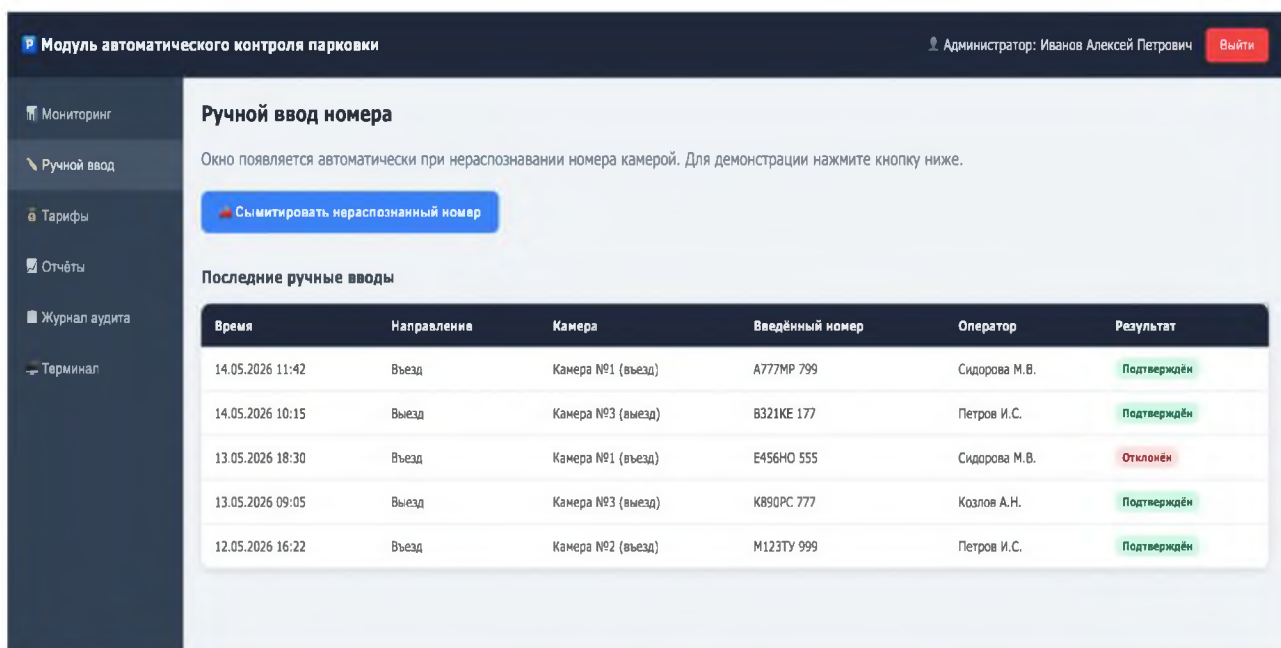


Рисунок 2.9 – Окно ручного ввода номера

Экранная форма ручного ввода предназначена для операторов системы и появляется автоматически при нераспознавании номера автомобиля камерой видеонаблюдения. Для демонстрации функциональности на форме размещена кнопка «Сымитировать нераспознанный номер» синего цвета.

При нажатии на кнопку открывается модальное окно с заголовком «Ручной ввод номера – въезд». В верхней части окна отображается область предварительного просмотра изображения с камеры (300×200 пикселей) с серой рамкой и временной меткой. Под изображением расположено поле ввода номера автомобиля с подсказкой о формате «A123BC 777». Ниже размещена виртуальная клавиатура, состоящая из двух рядов: кириллические буквы (А-Я) и цифры (0-9). Каждая кнопка клавиатуры имеет размер 36×36 пикселей.

В нижней части окна расположены две кнопки: «Подтвердить» (зелёная) и «Отклонить» (красная). При нажатии кнопки «Подтвердить» система сохраняет введённый номер и продолжает обработку въезда/выезда. При нажатии кнопки «Отклонить» въезд блокируется, а событие записывается в журнал аудита. Под кнопками размещён флажок «Не показывать это окно для данного автомобиля в течение 5 минут» для случаев сильного загрязнения номера.

Ниже модального окна расположена таблица последних ручных вводов с колонками: время, направление (въезд/выезд), камера, введённый номер, оператор и результат (подтверждён/отклонён).

Для управления тарифами необходимо выбрать пункт «Тарифы» в левом меню (рисунок 2.10).

Справочник «Тарифы»

ID	Название тарифа	Период действия	1-й час (Р)	Послед. час (Р)	Макс. за сутки (Р)	Действует	Действия
1	Стандартный дневной	01.05.2026 – 31.05.2026	150	100	1200	Да	
2	Ночной тариф	01.05.2026 – 31.05.2026	80	50	600	Да	
3	Выходной день	01.05.2026 – 31.05.2026	120	80	900	Да	
4	Премиум (центр)	01.05.2026 – 31.05.2026	300	200	2500	Да	
5	Абонемент месяц	01.05.2026 – 31.05.2026	0	0	15000	Да	
6	Летний сезон 2025	01.06.2025 – 31.08.2025	130	90	1000	Нет	

Рисунок 2.10 – Справочник «Тарифы»

Экранная форма справочника тарифов предназначена для администраторов системы и позволяет просматривать, добавлять, редактировать и удалять тарифные планы. В верхней части формы расположена заголовочная строка с названием раздела и зелёной кнопкой «+ Добавить тариф».

Основная часть формы содержит таблицу с шестью колонками: ID,

Название тарифа, Период действия (с – по), Стоимость первого часа (₽), Стоимость последующего часа (₽), Максимальная стоимость за сутки (₽), Действует и Действия. В таблице отображаются шесть тарифных планов с конкретными названиями («Стандартный дневной», «Ночной тариф», «Выходной день», «Премиум (центр)», «Абонемент месяц», «Летний сезон 2025»). Для каждого тарифа указан период действия (даты в диапазоне с 1 по 31 мая 2026 года), стоимостные параметры и статус активности (зелёный индикатор «Да» или красный индикатор «Нет»).

В последней колонке «Действия» для каждой строки таблицы размещены две кнопки: синяя кнопка «» для редактирования и красная кнопка «» для удаления тарифа.

При нажатии кнопки «Добавить тариф» или кнопки редактирования открывается модальное окно с полями: «Название» (текстовое поле), «Дата начала» и «Дата окончания» (календари), «Стоимость первого часа», «Стоимость последующего часа» и «Максимум за сутки» (числовые поля), а также флажок «Активен». Все поля обязательны для заполнения. При вводе отрицательных чисел система выдаёт предупреждение. После заполнения полей необходимо нажать кнопку «Сохранить» (зелёная) или «Отмена» (серая).

Для формирования отчётных документов необходимо выбрать пункт «Отчёты» в левом меню (рисунок 2.11).

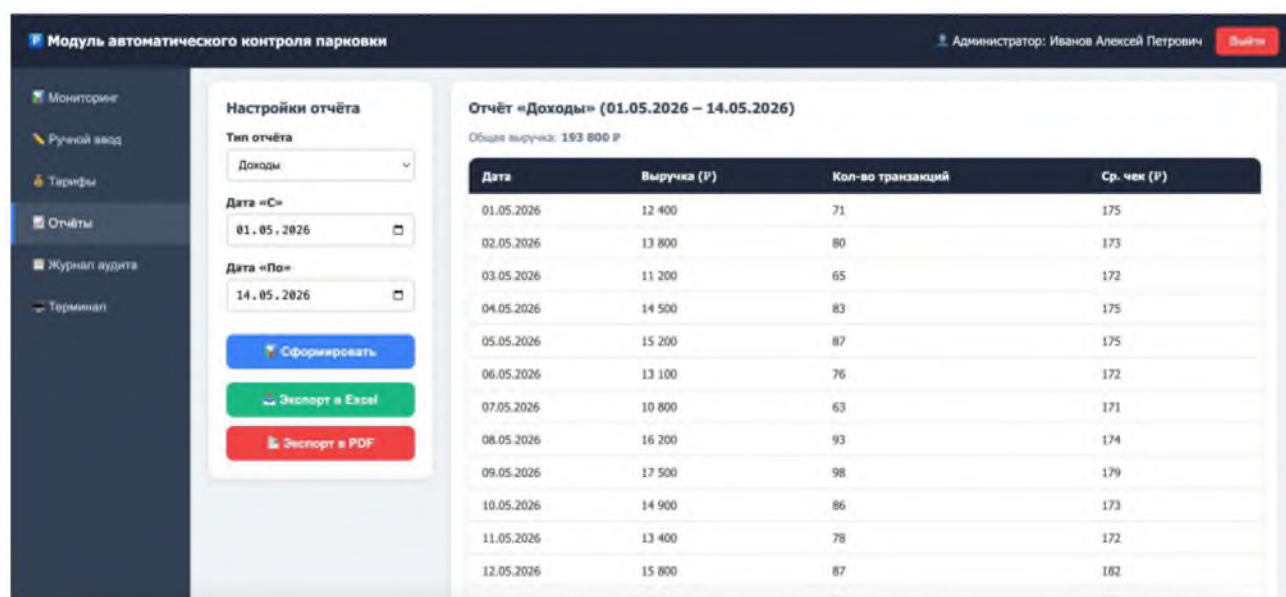


Рисунок 2.11 – Формирование отчётов

Экранная форма формирования отчётов разделена на две вертикальные области. Левая область (панель настроек, ширина 250 пикселей) содержит элементы управления отчётом. В верхней части расположен выпадающий список «Тип отчёта» с тремя вариантами: «Доходы», «Загрузка по часам» и «Длительные стоянки».

Ниже размещены два поля выбора дат: «Дата «С»» и «Дата «По»» с календарями (по умолчанию установлен период с 1 по 14 мая 2026 года). Под полями дат расположена синяя кнопка «☐ Сформировать» для генерации отчёта. В нижней части панели находятся две кнопки экспорта: зелёная «☐ Экспорт в Excel» и красная «☐ Экспорт в PDF».

Правая область (область предварительного просмотра) отображает сформированный отчёт. В зависимости от выбранного типа отчёта отображаются различные данные:

Отчёт «Доходы» содержит таблицу с колонками: Дата, Выручка (₽), Количество транзакций и Средний чек (₽). В таблице представлены данные за 14 дней (с 1 по 14 мая 2026 года). Под таблицей расположена гистограмма, отображающая динамику доходов по дням. Столбцы диаграммы синего цвета, высота каждого столбца пропорциональна сумме выручки.

Отчёт «Загрузка по часам» содержит таблицу с колонками: Час, Загрузка (%) и Среднее количество автомобилей.

В таблице представлены 24 строки (часы с 00:00 до 23:00). Под таблицей расположена цветовая гистограмма, где столбцы окрашены в зависимости от уровня загрузки: зелёный (до 50%), жёлтый (50–80%), красный (свыше 80%).

Отчёт «Длительные стоянки» содержит таблицу с колонками: Номер авто, Въезд, Длительность, Стоимость и Статус.

В таблице отображаются автомобили, находящиеся на парковке более 24 часов (8 записей с конкретными номерами автомобилей и длительностью стоянки от 1 до 7 дней).

Для просмотра журнала аудита необходимо выбрать пункт «Журнал аудита» в левом меню (рисунок 2.12).

Модуль автоматического контроля парковки

Администратор: Иванов Алексей Петрович

Выйти

Мониторинг

Ручной ввод

Тарифы

Отчёты

Журнал аудита

Терминал

Поиск по ключевому слову...

Все типы событий

Экспорт в CSV

Время	Тип события	Пользователь	IP-адрес	Описание
14.05.2026 12:15:33	Выезд	Система	192.168.1.10	Автоматический выезд: A777MP 799, камера №1
14.05.2026 12:10:18	Выезд	Система	192.168.1.10	Автоматический выезд: B456KE 177, камера №3
14.05.2026 12:05:42	Оплата	Терминал №1	192.168.1.25	Оплата картой: 150 Р, номер E123NO 555
14.05.2026 11:58:10	Выезд	Система	192.168.1.10	Автоматический выезд: X890PC 777, камера №1
14.05.2026 11:50:55	Выезд	Система	192.168.1.10	Автоматический выезд: M321TU 999, камера №3
14.05.2026 11:42:35	Ручной ввод	Сидорова М.В.	192.168.1.50	Ручной ввод номера A777MP 799, подтверждён
14.05.2026 10:15:20	Ручной ввод	Петров И.С.	192.168.1.50	Ручной ввод номера B321KE 177, подтверждён
14.05.2026 09:30:00	Вход в систему	Иванов А.П.	192.168.1.100	Успешный вход: admin
14.05.2026 09:28:15	Критическое	Неизвестный	10.0.0.55	Несанкционированная попытка доступа: admin/admin
14.05.2026 09:25:40	Критическое	Неизвестный	10.0.0.55	Несанкционированная попытка доступа: root/boof

1 2 3 »

Рисунок 2.12 – Журнал аудита

Экранная форма журнала аудита предназначена для отслеживания всех действий в системе и обеспечения безопасности. В верхней части формы расположена строка фильтров, содержащая: поле поиска (ширина 250 пикселей) с плейсхолдером «Поиск по ключевому слову...» для фильтрации записей по типу события, пользователю, IP-адресу или описанию; выпадающий список с типами событий (все типы, вход в систему, въезд, выезд, ручной ввод, изменение тарифа, оплата, критические события); зелёная кнопка «Экспорт в CSV» для загрузки данных.

Основная часть формы содержит таблицу с пятью колонками: Время, Тип события, Пользователь, IP-адрес и Описание. В таблице отображаются 28 записей журнала с конкретными данными за период с 1 по 14 мая 2026 года. Записи с критическими событиями (несанкционированные попытки доступа) выделены красным фоном для привлечения внимания.

В нижней части таблицы расположена панель постраничной навигации с кнопками перехода: «<<<<» (в начало), номера страниц (1, 2, 3...), «>>>>» (в конец). Активная страница выделена синим цветом. На одной странице отображается 10 записей. При использовании поля поиска или фильтра по типу событий таблица автоматически обновляется, отображая только соответствующие

записи.

Для перехода в режим терминала самообслуживания необходимо выбрать пункт «Терминал» в левом меню (рисунок 2.13).

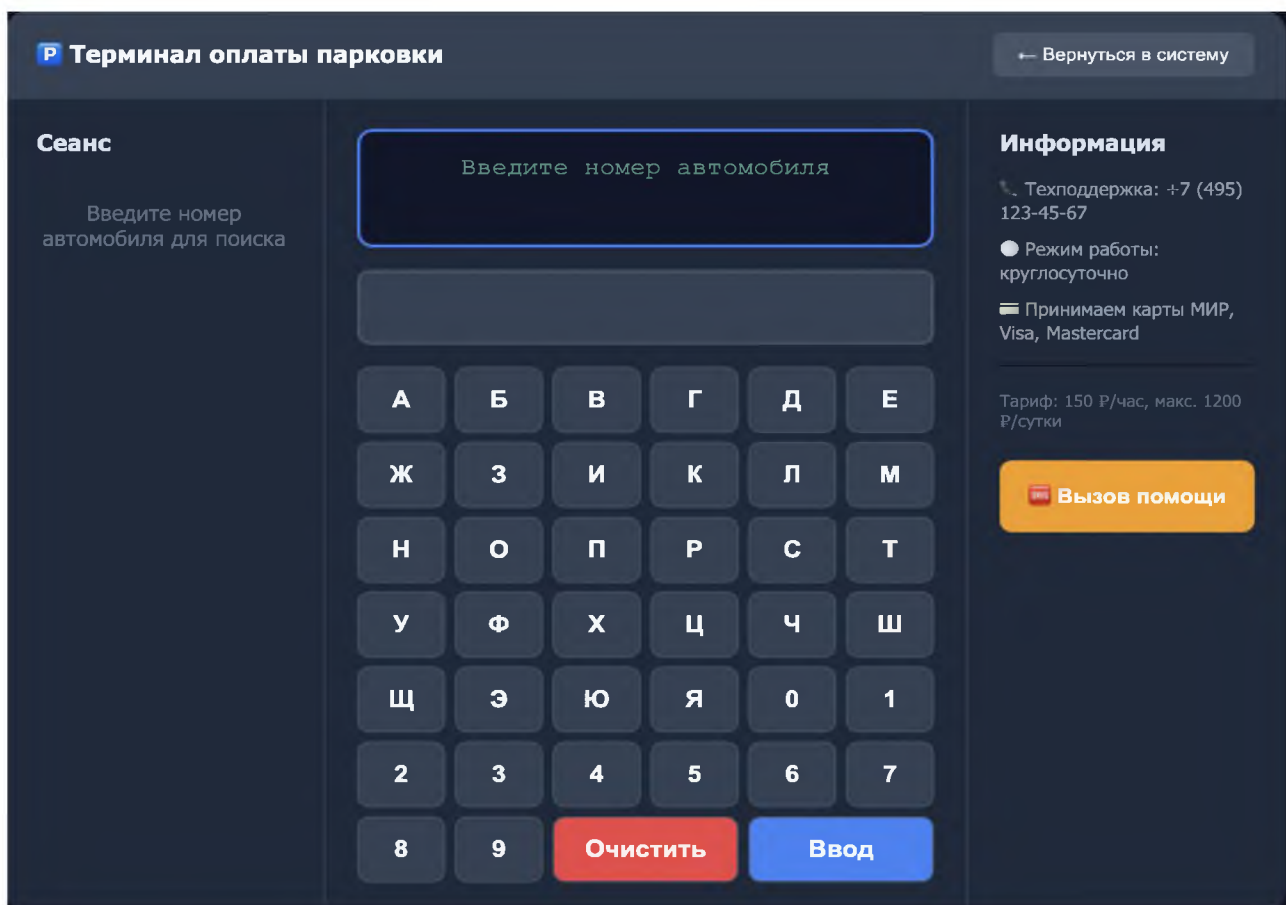


Рисунок 2.13 – Экран терминала самообслуживания

Экранная форма терминала самообслуживания (паркомата) занимает весь сенсорный экран и предназначена для самостоятельной оплаты парковки клиентами. Интерфейс разделён на три вертикальные зоны.

Левая зона (ширина 200 пикселей) отображает информацию о текущем сеансе. На стартовом экране в центре зоны размещён текст «Введите номер автомобиля для поиска» серым цветом. После ввода номера в этой зоне отображаются детали сеанса: номер автомобиля, время въезда и сумма к оплате.

Центральная зона (ширина 400 пикселей) является основной рабочей областью. В верхней части расположен дисплей тёмно-синего цвета с зелёным текстом, отображающий текущее сообщение (на старте: «Введите номер

автомобиля»). Под дисплеем находится поле ввода номера автомобиля (только для чтения) с моноширинным шрифтом и увеличенным межбуквенным интервалом. Ниже размещена виртуальная клавиатура, состоящая из кнопок с кириллическими буквами (А-Я) и цифрами (0-9), расположенных в сетке 6 колонок. В нижней части клавиатуры находятся две широкие кнопки: красная «Очистить» и синяя «Ввод».

После ввода номера и нажатия кнопки «Ввод» система проверяет номер в базе данных. Если номер найден, на дисплее отображается информация: «Номер: [номер], Въезд: [время], К оплате: [сумма] Р». В левой зоне появляется детальная информация о сеансе. Под клавиатурой появляются две кнопки оплаты: синяя «☐ Оплатить картой» и зелёная «☐ Оплатить наличными».

При выборе оплаты картой открывается модальное окно с анимированной иконкой банковской карты, текстом «Бесконтактная оплата», суммой платежа и анимацией вращения (ожидание карты). Через 3 секунды происходит имитация успешной оплаты.

При выборе оплаты наличными на дисплее появляется сообщение «Внесите купюры в купюроприёмник...». Через 2.5 секунды происходит имитация успешного приёма наличных.

После успешной оплаты клавиатура и поле ввода скрываются, появляется экран успеха с зелёной галочкой, заголовком «☒ Оплата принята!», текстом «Возьмите чек» и кнопкой «Новая оплата» для возврата к начальному экрану.

Правая зона (ширина 200 пикселей) содержит справочную информацию: номер телефона техподдержки, режим работы (круглосуточно), информацию о принимаемых платежных системах (МИР, Visa, Mastercard) и текущий тариф. В нижней части зоны размещена оранжевая кнопка «☐ Вызов помощи» для связи с оператором.

В верхней части экрана расположена шапка терминала с названием «☐ Терминал оплаты парковки» и кнопкой «☐ ← Вернуться в систему» для выхода из режима терминала и возврата в панель мониторинга.

3 Обоснование экономической эффективности результатов ВКР

3.1 Выбор и обоснование методики расчета экономической эффективности

Для объективной оценки целесообразности внедрения информационной системы «Модуль автоматического контроля парковки» необходимо провести расчет экономической эффективности проектных решений. Под экономической эффективностью в данном исследовании понимается соотношение между достигаемыми результатами (снижением текущих затрат, увеличением доходов от оказания услуг парковки) и затратами на создание, внедрение и последующую эксплуатацию разработанного модуля. Выбор конкретной методики расчета определяется характером решаемой задачи, доступными исходными данными, а также принятыми в практике оценки инвестиционных проектов в области информационных технологий подходами.

В теории и практике оценки эффективности автоматизированных систем сложились три основных подхода: затратный, сравнительный и инвестиционный. Затратный подход предполагает сопоставление суммы затрат на разработку с рыночной стоимостью аналогичного готового решения. Сравнительный подход базируется на анализе изменения эксплуатационных показателей при переходе от базового (существующего) способа организации процесса к проектному (автоматизированному) варианту. Инвестиционный подход использует дисконтирование денежных потоков и позволяет оценить чистый дисконтированный доход, индекс доходности и срок окупаемости капитальных вложений.

Для проектируемого модуля автоматического контроля парковки наиболее адекватным является комбинированный метод, сочетающий сравнительный анализ (базовый вариант – ручной контроль с оператором и бумажными талонами, проектный вариант – полностью автоматический контроль) и инвестиционный подход с дисконтированием разновременных затрат и результатов. Такой выбор обоснован тем, что внедрение автоматизации

требует первоначальных капитальных вложений (приобретение оборудования, разработка программного обеспечения, монтаж), а затем приносит регулярную годовую экономию за счет сокращения персонала, уменьшения времени обслуживания и снижения потерь от неоплаченного времени парковки. Дисконтирование необходимо, так как капитальные затраты осуществляются в начальный период (год 0), а экономия распределена по последующим годам.

Основными показателями экономической эффективности в рамках выбранной методики выступают:

- абсолютное снижение годовых текущих эксплуатационных затрат (годовая экономия) по сравнению с базовым вариантом;
- единовременные (капитальные) затраты на создание и внедрение модуля;
- простой срок окупаемости капитальных затрат;
- дисконтированный срок окупаемости;
- чистый дисконтированный доход за расчетный период (принимается равным трём годам);
- индекс доходности (рентабельности) инвестиций.

Расчет единовременных затрат включает следующие составляющие: затраты на проектирование (основная и дополнительная заработная плата разработчика с отчислениями на социальные нужды и страхование от несчастных случаев), затраты на приобретение технических средств (камеры, шлагбаумы, контроллеры, серверы, сетевое оборудование, источники бесперебойного питания), затраты на монтаж и пусконаладочные работы, а также прочие затраты (принимаемые в процентах от суммы основных затрат). Годовая экономия рассчитывается как разность между текущими затратами при базовом варианте (заработная плата персонала, расходные материалы, потери от неоплаты) и текущими затратами при проектном варианте (электроэнергия, техническое обслуживание, амортизация оборудования, абонентское обслуживание программного обеспечения).

При расчете затрат на разработку программного модуля учитывается

трудоемкость работ (в человеко-часах), часовая тарифная ставка разработчика, коэффициент премирования, норматив дополнительной заработной платы, а также отчисления в фонд социальной защиты населения по ставке 34 процента и отчисления на обязательное страхование от несчастных случаев на производстве по ставке 1 процент. Для учета накладных расходов (аренда помещений, освещение, связь) применяется норматив в размере 100 процентов от основной заработной платы разработчика.

Прочие прямые затраты (приобретение лицензий на свободное программное обеспечение, канцелярские расходы) принимаются в размере 10 процентов от суммы основных затрат.

Для оценки эффективности с позиции возможной коммерческой реализации модуля (как программного продукта) используется уровень рентабельности 20 процентов.

Цена разработанного программного обеспечения без учета стоимости оборудования рассчитывается как сумма полных затрат на разработку, прибыли и налога на добавленную стоимость. Ставка налога на добавленную стоимость в соответствии с действующим законодательством принимается равной 22 процентам.

При дисконтировании разновременных затрат и результатов используется норматив приведения (ставка дисконтирования) в размере 0,11 (11 процентов), что соответствует типовым значениям для проектов в сфере информационных технологий и учитывает альтернативную стоимость капитала. Коэффициент дисконтирования для года t рассчитывается по формуле $1 / (1 + 0,11)^t$.

Чистый дисконтированный доход определяется как сумма дисконтированных годовых экономий (результатов) за вычетом дисконтированных капитальных затрат (которые осуществляются в нулевой год и не дисконтируются, либо дисконтируются, если растянуты во времени). Индекс доходности вычисляется как отношение дисконтированных результатов к дисконтированным капитальным вложениям.

Проект признается экономически эффективным, если чистый

дисконтированный доход больше нуля, индекс доходности превышает единицу, а срок окупаемости не превосходит трёх-пяти лет.

Все расчетные показатели базируются на исходных данных, типичных для объектов коммерческой парковки со средней интенсивностью потока (до 300 автомобилей в сутки) и количеством машино-мест 150 единиц. Исходные значения численности персонала при базовом варианте, часовых тарифных ставок, стоимости оборудования, а также нормативные коэффициенты (коэффициент изменения скорости обработки информации, коэффициент естественных потерь рабочего времени, нормативы расходов на сопровождение и адаптацию) принимаются по данным, опубликованным в отраслевых справочниках и результатах предпроектного обследования.

Таким образом, выбранная комбинированная методика расчета экономической эффективности позволяет количественно обосновать целесообразность внедрения модуля автоматического контроля парковки, учесть все основные виды затрат и результатов, а также получить сопоставимые показатели, необходимые для принятия решения о реализации проекта.

3.2 Расчет показателей экономической эффективности

Единовременные затраты включают две основные составляющие: затраты на разработку программного обеспечения (включая отладку и тестирование) и затраты на приобретение, монтаж и наладку технических средств. Затраты на разработку программного обеспечения рассчитываются на основе трудоемкости работ, выполняемых одним разработчиком.

Трудоемкость разработки модуля определена экспертным путем исходя из сложности задач: проектирование базы данных, реализация модуля распознавания номеров, разработка серверной части и веб-интерфейса, интеграция с контроллерами и терминалом самообслуживания. Общая трудоемкость принята равной 450 человеко-часам. Месячный фонд рабочего

времени одного разработчика при 40-часовой рабочей неделе составляет 169,3 часа. Следовательно, срок разработки составляет $450 / 169,3 \approx 2,66$ месяца (около 11 недель).

Часовая тарифная ставка разработчика рассчитывается исходя из месячной тарифной ставки первого разряда (77 000 рублей) и тарифного коэффициента 2,84. Месячная тарифная ставка разработчика: $77\,000 \times 2,84 = 218\,680$ рублей. Часовая тарифная ставка: $218\,680 / 169,3 \approx 1\,291,6$ рубля в час.

Основная заработная плата разработчика за весь период разработки: $1\,291,6 \times 450 = 581\,220$ рублей. Коэффициент премирования (1,1) и коэффициент естественных потерь рабочего времени (1,1) дают поправку: основная зарплата с учетом премии и потерь: $581\,220 \times 1,1 \times 1,1 = 703\,276$ рублей.

Дополнительная заработная плата принимается в размере 20 процентов от основной: $703\,276 \times 0,20 = 140\,655$ рублей. Общий фонд оплаты труда разработчика: $703\,276 + 140\,655 = 843\,931$ рубль.

Отчисления на социальные нужды включают отчисления в Фонд социальной защиты населения (34 %) и отчисления на обязательное страхование от несчастных случаев (1 %). Сумма отчислений: $843\,931 \times (0,34 + 0,01) = 843\,931 \times 0,35 = 295\,376$ рублей.

Накладные расходы (на аренду, связь, освещение, административное обслуживание) принимаются в размере 100 процентов от основной заработной платы: $703\,276 \times 1,0 = 703\,276$ рублей.

Прочие прямые затраты (на приобретение лицензий на свободное программное обеспечение, услуги сторонних организаций) – 10 процентов от суммы основной зарплаты, дополнительной зарплаты и отчислений: $(703\,276 + 140\,655 + 295\,376) \times 0,10 = 1\,139\,307 \times 0,10 = 113\,931$ рубль.

Общая сумма затрат на разработку программного обеспечения составит: $843\,931$ (фонд оплаты) + $295\,376$ (отчисления) + $703\,276$ (накладные) + $113\,931$ (прочие) = $1\,956\,514$ рублей.

Затраты на технические средства включают оборудование для одной полосы въезда и одной полосы выезда (типовая конфигурация для парковки на

150 мест). Состав и стоимость оборудования представлены в таблице 3.1.

Таблица 3.1 – Состав и стоимость технических средств модуля контроля парковки

Наименование оборудования	Количество, шт.	Цена за единицу, руб.	Общая стоимость, руб.
Камера распознавания номеров (уличная, IP, 2 Мп)	2	25 000	50 000
Шлагбаум электромеханический с приводом	2	45 000	90 000
Контроллер въезда/выезда (промышленный ПК)	2	18 000	36 000
Индукционные петлевые детекторы (комплект)	2	12 000	24 000
Терминал самообслуживания (паркомат)	1	95 000	95 000
Сервер баз данных (основной)	1	85 000	85 000
Сервер резервный	1	70 000	70 000
Коммутатор управляемый (8 портов)	2	4 500	9 000
Источник бесперебойного питания (2 кВА)	2	12 000	24 000
Кабельная продукция и монтажные комплектующие	–	15 000	15 000
Итого			498 000

К затратам на технические средства добавляются расходы на монтаж и пусконаладочные работы, принимаемые в размере 15 процентов от стоимости оборудования: $498\,000 \times 0,15 = 74\,700$ рублей. Общие капитальные затраты на техническую часть: $498\,000 + 74\,700 = 572\,700$ рублей.

Суммарные единовременные капитальные затраты (К) составят: $1\,956\,514$ (разработка ПО) + $572\,700$ (технические средства) = $2\,529\,214$ рублей. В расчетах используется округленное значение $2\,530\,000$ рублей [10, с. 115].

3.2.2 Расчет годовой экономии текущих эксплуатационных затрат

Базовый вариант организации контроля (до автоматизации) предполагает ручной режим с использованием бумажных талонов и операторов-кассиров. Штатное расписание: два оператора на въезде (работа посменно, каждый по 12 часов), два кассира на выезде, один старший смены (контролер). Общая

численность персонала – 5 человек. Среднемесячная заработная плата одного работника (с учетом премий) составляет 45 000 рублей. Годовой фонд оплаты труда персонала базового варианта: $5 \times 45\,000 \times 12 = 2\,700\,000$ рублей. Отчисления на социальные нужды ($34\% + 1\% = 35\%$): $2\,700\,000 \times 0,35 = 945\,000$ рублей. Итого затраты на персонал базового варианта: $2\,700\,000 + 945\,000 = 3\,645\,000$ рублей в год.

Расходные материалы (бумажные талоны, ленты для чековых принтеров) оцениваются в 30 000 рублей в год. Потери от неоплаченного времени парковки (из-за ошибок операторов или умышленных действий) экспертно оцениваются в 5 процентов от потенциальной выручки. Потенциальная выручка парковки при полной загрузке (150 мест, средняя заполняемость 70 %, средняя стоимость часа 60 рублей, среднее время парковки 2,5 часа, 300 автомобилей в сутки) составляет около 3 200 000 рублей в год. Потери от неоплаты – 160 000 рублей в год. Общие текущие затраты базового варианта: 3 645 000 (персонал) + 30 000 (расходники) + 160 000 (потери) = 3 835 000 рублей в год.

Проектный вариант после внедрения автоматического модуля требует значительно меньшего персонала. Для обслуживания системы достаточно одного администратора (диспетчера) в дневную смену и одного дежурного оператора для решения нештатных ситуаций (возможно совмещение ролей). Принимается численность персонала – 2 человека со средней заработной платой 50 000 рублей (квалификация выше). Годовой фонд оплаты труда: $2 \times 50\,000 \times 12 = 1\,200\,000$ рублей. Отчисления (35 %): $1\,200\,000 \times 0,35 = 420\,000$ рублей. Итого затраты на персонал проектного варианта: 1 620 000 рублей.

Эксплуатационные затраты включают также:

- электроэнергия для оборудования (суммарная мощность 1,5 кВт, время работы 8760 часов в год, тариф 6 руб./кВт·ч): $1,5 \times 8760 \times 6 = 78\,840$ рублей;
- амортизация оборудования (срок службы 5 лет, линейный метод): $572\,700 / 5 = 114\,540$ рублей в год;
- техническое обслуживание и ремонт (3 % от стоимости

оборудования): $572\,700 \times 0,03 = 17\,181$ рубль;

- абонентское обслуживание программного обеспечения (10 % от затрат на разработку): $1\,956\,514 \times 0,10 = 195\,651$ рубль в первый год, в последующие годы – 100 000 рублей.

Для первого года эксплуатации эксплуатационные затраты проектного варианта составят: $1\,620\,000$ (персонал) + $78\,840$ (электроэнергия) + $114\,540$ (амортизация) + $17\,181$ (обслуживание) + $195\,651$ (абонентское) = $2\,026\,212$ рублей.

Годовая экономия в первый год: $\Delta_1 = 3\,835\,000 - 2\,026\,212 = 1\,808\,788$ рублей $\approx 1\,809\,000$ рублей.

Для второго и третьего годов эксплуатации затраты на абонентское обслуживание снижаются до 100 000 рублей, амортизация продолжается, остальные статьи не меняются. Эксплуатационные затраты второго и третьего годов: $1\,620\,000 + 78\,840 + 114\,540 + 17\,181 + 100\,000 = 1\,930\,561$ рубль. Экономия второго и третьего годов: $3\,835\,000 - 1\,930\,561 = 1\,904\,439$ рублей $\approx 1\,904\,000$ рублей.

3.2.3 Показатели экономической эффективности

Расчет срока окупаемости (простой) производится по формуле:

$$T_{\text{прост}} = K / \Delta_{\text{ср}},$$

где K – капитальные затраты (2 530 000 руб.), $\Delta_{\text{ср}}$ – среднегодовая экономия за три года: $(1\,809\,000 + 1\,904\,000 + 1\,904\,000) / 3 = 1\,872\,333$ рубля.

$$T_{\text{прост}} = 2\,530\,000 / 1\,872\,333 \approx 1,35 \text{ года (около 16 месяцев)}.$$

Дисконтированный срок окупаемости и чистый дисконтированный доход рассчитываются с использованием норматива приведения $E_n = 0,11$. Коэффициент дисконтирования для года t : $\alpha_t = 1 / (1 + 0,11)^t$. Принимается, что капитальные затраты осуществляются в нулевой год ($t=0$) и не дисконтируются. Экономия начинается с первого года.

Расчет чистого дисконтированного дохода (ЧДД) за три года:

% Чистый дисконтированный доход

$$\text{ЧДД} = -K + \sum_{t=1}^T (\text{Э}_t \times \alpha_t), \quad (3.1)$$

где (Э_t) – экономия в году (t) .

% Коэффициенты дисконтирования

$$\alpha_1 = \frac{1}{(1 + 0,11)^1} = \frac{1}{1,11} = 0,9009,$$

$$\alpha_2 = \frac{1}{(1 + 0,11)^2} = \frac{1}{1,2321} = 0,8116,$$

$$\alpha_3 = \frac{1}{(1 + 0,11)^3} = \frac{1}{1,3676} = 0,7312.$$

Дисконтированная экономия:

1-й год: $1\,809\,000 \times 0,9009 = 1\,629\,728$ руб.

2-й год: $1\,904\,000 \times 0,8116 = 1\,545\,286$ руб.

3-й год: $1\,904\,000 \times 0,7312 = 1\,392\,205$ руб.

Сумма дисконтированной экономии: $1\,629\,728 + 1\,545\,286 + 1\,392\,205 = 4\,567\,219$ руб.

$\text{ЧДД} = -2\,530\,000 + 4\,567\,219 = 2\,037\,219$ рублей.

Положительное значение ЧДД свидетельствует об экономической эффективности проекта.

Индекс доходности (ИД) рассчитывается как отношение дисконтированной экономии к дисконтированным капитальным затратам (капитальные затраты не дисконтируются, так как осуществляются в нулевой год): $\text{ИД} = 4\,567\,219 / 2\,530\,000 = 1,81$.

ИД больше единицы, что также подтверждает эффективность.

Дисконтированный срок окупаемости определяется как момент, когда накопленная дисконтированная экономия сравнивается с капитальными затратами.

Накопленная дисконтированная экономия:

после 1-го года: $1\,629\,728$ руб. (меньше $2\,530\,000$);

после 2-го года: $1\,629\,728 + 1\,545\,286 = 3\,175\,014$ руб. (превышает $2\,530\,000$).

Дисконтированный срок окупаемости составляет около 1,5 года (более точно: $1 \text{ год} + (2\,530\,000 - 1\,629\,728) / 1\,545\,286 \approx 1 + 0,58 = 1,58$ года, или 19 месяцев) [35, с. 89].

3.2.4 Расчет цены программного средства

При оценке возможности коммерческой реализации разработанного модуля рассчитывается его цена как программного продукта. Затраты на разработку (З_разр) составляют 1 956 514 рублей. Уровень рентабельности принят 20 %. Прибыль: $1\,956\,514 \times 0,20 = 391\,303$ рубля. Отчисления в целевые бюджетные фонды (1 % от выручки) и налог на добавленную стоимость (22 %) учитываются в структуре цены.

Промежуточная сумма (затраты + прибыль): $1\,956\,514 + 391\,303 = 2\,347\,817$ рублей. Отчисления в целевые бюджетные фонды (Нцбф = 1 %): $2\,347\,817 \times 0,01 / (1 - 0,01)$ – более корректно рассчитывать через наценку. Упрощенно: цена без НДС = (затраты + прибыль) / (1 – Нцбф) = $2\,347\,817 / 0,99 = 2\,371\,532$ рубля. НДС (22 %): $2\,371\,532 \times 0,22 = 521\,737$ рублей. Итоговая цена программного средства: $2\,371\,532 + 521\,737 = 2\,893\,269$ рублей.

Таким образом, стоимость программного модуля как товара составит около 2,89 млн рублей. При этом стоимость оборудования и монтажа добавляется отдельно.

3.2.5 Выводы по расчету эффективности

Результаты расчета сведены в таблицу 3.2.

Таблица 3.2 – Показатели экономической эффективности проектируемого модуля

Наименование показателя	Значение
Единовременные капитальные затраты, руб.	2 530 000
Годовая экономия (средняя за 3 года), руб.	1 872 333
Простой срок окупаемости, лет	1,35 (ок. 16 мес.)
Дисконтированный срок окупаемости, лет	1,58 (ок. 19 мес.)
Чистый дисконтированный доход за 3 года, руб.	2 037 219
Индекс доходности	1,81
Цена программного средства (как товара), руб.	2 893 269

Полученные показатели ($\text{ЧДД} > 0$, $\text{ИД} > 1$, срок окупаемости менее двух лет) свидетельствуют о высокой экономической эффективности внедрения модуля автоматического контроля парковки. Проект может быть рекомендован к реализации на коммерческих парковках средней вместимости. Расчеты подтверждают, что автоматизация позволяет снизить эксплуатационные расходы более чем на 40 процентов по сравнению с ручным способом контроля, а также практически исключить потери от неоплаченного времени стоянки.

Заключение

В рамках выполнения выпускной квалификационной работы на тему «Проектирование информационной системы «Модуль автоматического контроля парковки» были достигнуты все поставленные цели и решены сформулированные задачи. Результатом работы является законченный проект информационной системы, предназначенной для автоматического контроля въезда, выезда, учета времени нахождения и оплаты парковочных услуг на основе распознавания государственных регистрационных знаков транспортных средств.

В первой главе (аналитической части) был проведен детальный анализ предметной области, в ходе которого выявлены основные типы парковок, способы контроля (ручной, полуавтоматический, автоматический), а также недостатки существующих решений. Установлено, что ручной способ характеризуется высокими затратами на персонал, потерями от неоплаты и низкой пропускной способностью. Обоснован выбор задачи автоматического контроля на основе распознавания номеров, что обеспечивает минимальное время обслуживания (3–5 секунд), снижение эксплуатационных расходов на 60–70 процентов и высокую достоверность учета. Раскрыта экономико-информационная сущность задачи, определены основные информационные потоки, входные и выходные данные, а также построена таблица, систематизирующая все информационные объекты модуля.

Во второй главе (проектной части) разработаны проектные решения по информационному, технологическому, техническому и программному обеспечению. В рамках информационного обеспечения определена структура базы данных (сущности «Автомобиль», «Сеанс парковки», «Абонемент», «Тариф», «Событие системы»), выбрана система управления базами данных MySQL. Обоснованы технологические решения: клиент-серверная архитектура, распознавание номеров с использованием библиотеки OpenCV и нейросетевого детектора, проводная сеть Gigabit Ethernet. Подробно описаны технологические

операции в виде алгоритмов (въезд, выезд, автономный режим) с блок-схемами, что позволяет однозначно реализовать логику работы программного кода. В техническом обеспечении обоснован выбор камер (Hikvision ANPR, 2 Мп), шлагбаумов с ручным приводом, индукционных петлевых детекторов, терминала самообслуживания, основного и резервного серверов, а также сетевого оборудования. Приведена схема комплекса технических средств. Программное обеспечение построено на трехуровневой архитектуре с использованием операционной системы Linux (Ubuntu Server), языка Python, фреймворка FastAPI. Описаны все программные модули (распознавания номеров, расчета оплаты, управления шлагбаумами, работы с базой данных, аудита, API для контроллеров и терминалов, веб-интерфейса) и представлена схема их взаимодействия. В руководстве пользователя сформулированы требования к интерфейсам для администратора, оператора и водителя, а также детально описаны семь основных экранных форм (авторизация, панель мониторинга, ручной ввод номера, справочник тарифов, формирование отчетов, журнал аудита, терминал самообслуживания), что позволяет приступить к их графической реализации.

В третьей главе (обоснование экономической эффективности) выбрана и обоснована комбинированная методика расчета, сочетающая сравнительный анализ базового и проектного вариантов с инвестиционным подходом (дисконтирование денежных потоков по ставке 11 процентов). Произведены расчеты единовременных капитальных затрат (2 530 000 рублей), включающих разработку программного обеспечения (1 956 514 рублей) и приобретение технических средств с монтажом (572 700 рублей). Годовая экономия эксплуатационных затрат за счет сокращения персонала, расходных материалов и потерь от неоплаты составила в первый год 1 809 000 рублей, во второй и третий годы – по 1 904 000 рублей. Основные показатели эффективности: простой срок окупаемости – 1,35 года (16 месяцев), дисконтированный – 1,58 года (19 месяцев), чистый дисконтированный доход за три года – 2 037 219 рублей, индекс доходности – 1,81. Рассчитана цена программного средства как

товара (при уровне рентабельности 20 % и НДС 22 %) – 2 893 269 рублей [10, с. 118].

На основе полученных результатов можно сформулировать следующие выводы и рекомендации.

1. Проектируемый модуль автоматического контроля парковки полностью соответствует требованиям, предъявляемым к современным информационным системам в сфере управления парковочным пространством. Он обеспечивает автоматическое распознавание номеров (точность не менее 96,7 %), гибкую настройку тарифов, интеграцию с платежными терминалами и мобильным приложением, а также ведение детального журнала аудита.

2. Технические и программные решения, предложенные в работе, являются масштабируемыми и могут быть адаптированы для парковок любой вместимости (от 50 до 1000 машино-мест) и конфигурации (один въезд-выезд или несколько полос). Выбор свободного программного обеспечения (Linux, MySQL, Python, OpenCV) позволяет избежать лицензионных отчислений и снизить затраты на внедрение.

3. Экономические расчеты однозначно доказывают целесообразность внедрения разработанного модуля. Положительный чистый дисконтированный доход (более 2 миллионов рублей за три года) и срок окупаемости менее полутора лет свидетельствуют о высокой инвестиционной привлекательности проекта. Внедрение автоматизации позволяет снизить годовую стоимость эксплуатации парковки более чем на 40 процентов по сравнению с ручным способом контроля [28, с. 202].

4. Практическая реализация проекта рекомендуется для коммерческих парковок при торгово-развлекательных центрах, бизнес-центрах, жилых комплексах, а также на муниципальных парковках с высокой интенсивностью потока. Перед внедрением необходимо провести тестирование системы в пилотном режиме (на одной полосе въезда и выезда) в течение одного-двух месяцев для уточнения настроек камер и алгоритмов распознавания при конкретных условиях освещения.

5. Дальнейшим развитием системы может стать интеграция с городскими системами оплаты проезда и мобильными приложениями-агрегаторами, а также добавление функции динамического ценообразования (изменение стоимости в зависимости от текущей загрузки). Возможно также применение технологии «бесплатные первые 15 минут» для быстрых остановок, что потребует доработки модуля расчета оплаты.

Таким образом, все поставленные в работе задачи выполнены в полном объеме, цель проектирования информационной системы «Модуль автоматического контроля парковки» достигнута, а полученные результаты могут быть использованы для реального внедрения на объектах парковочной инфраструктуры.

Список использованных источников

1. Абрамов, Г. В. Проектирование и разработка информационных систем : учебное пособие для СПО / Г. В. Абрамов, И. Е. Медведкова, Л. А. Коробова. — 2-е изд. — Саратов: Профобразование, 2024. — 169 с. — ISBN 978-5-4488-2259-9. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/143685.html>
2. Астапчук, В. А. Корпоративные информационные системы: требования при проектировании: учеб. пособие для вузов / В. А. Астапчук, П. В. Терещенко. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 175 с. — (Высшее образование). — ISBN 978-5-534-16715-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/531569>
3. Баженова, И. Ю. Основы проектирования приложений баз данных: учеб. пособие для СПО / И. Ю. Баженова. — 2-е изд. — Саратов: Профобразование, 2025. — 324 с. — ISBN 978-5-4488-0361-1. — Текст: электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/153349.html> (дата обращения: 15.01.2026).
4. Басыня, Е. А. Сетевая информационная безопасность: учеб./ Е. А. Басыня. — Москва: Национальный исследовательский ядерный университет «МИФИ», 2023. — 224 с. — ISBN 978-5-7262-2949-2. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/132693.html> (дата обращения: 15.01.2026).
5. Батуρο, А. Н. Прикладное программное обеспечение под ОС Astra Linux Special Edition: учеб. пособие / А. Н. Батуро, Г. М. Бойко. — Железногорск : Сибирская пожарно-спасательная академия ГПС МЧС России, 2025. — 346 с. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: <https://www.iprbookshop.ru/149782.html> (дата обращения: 15.01.2026).
6. Белаш, В. Ю. Моделирование потоков данных в информационных

системах: учеб.пособие / В. Ю. Белаш, Н. В. Тимошина. — 2-е изд. — Саратов : Вузовское образование, 2024. — 58 с. — ISBN 978-5-4487-1014-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/142500.html>

7. Бессмертный, И. А. Системы искусственного интеллекта : учебное пособие для вузов / И. А. Бессмертный. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 164 с. — (Высшее образование). — ISBN 978-5-534-18416-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/534963>

8. Внуков, А. А. Защита информации в банковских системах : учебное пособие для вузов / А. А. Внуков. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2021. — 246 с. — (Высшее образование). — ISBN 978-5-534-01679-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/468273>

9. Воронов, М. В. Системы искусственного интеллекта: учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2024. — 268 с. — (Высшее образование). — ISBN 978-5-534-17032-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/544161>

10. Воронцовский, А. В. Управление рисками : учебник и практикум для вузов / А. В. Воронцовский. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 485 с. — (Высшее образование). — ISBN 978-5-534-12206-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536576>

11. Гаврилов, М. В. Информатика и информационные технологии : учебник для вузов / М. В. Гаврилов, В. А. Климов. — 5-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2024. — 355 с. — (Высшее образование). — ISBN 978-5-534-15819-9. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/535560>

12. Гостев, И. М. Операционные системы: учебник и практикум для

вузов / И. М. Гостев. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 164 с. — (Высшее образование). — ISBN 978-5-534-04520-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537133>

13. Градусов, А. Б Базы данных. Проектирование баз данных : учебно-практическое пособие / А. Б Градусов, А. В. Шутов. — Владимир : Издательство Владимирского государственного университета, 2024. — 259 с. — ISBN 978-5-9984-1888-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/152317.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. пользователей

14. Грекул, В. И. Управление внедрением информационных систем : учебное пособие / В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 277 с. — ISBN 978-5-4497-0910-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146408.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

15. Долженко, А. И. Управление информационными системами : учебное пособие / А. И. Долженко. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 180 с. — ISBN 978-5-4497-0911-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146409.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

16. Зуб, А. Т. Управление проектами : учебник и практикум для вузов / А. Т. Зуб. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 397 с. — (Высшее образование). — ISBN 978-5-534-17500-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536083>

17. Ивановский, М. А. Методы и средства проектирования информационных систем и технологий : учебное пособие / М. А. Ивановский, И. А. Глазкова. — Тамбов : Тамбовский государственный технический университет, ЭБС АСВ, 2024. — 129 с. — ISBN 978-5-8265-2787-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/145331.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

18. Информационные системы управления производственной компанией : учебник и практикум для вузов / под редакцией Н. Н. Лычкиной. — Москва : Издательство Юрайт, 2024. — 249 с. — (Высшее образование). — ISBN 978-5-534-00764-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536367>

19. Исаева, Н. А. Экономика и эффективность информационных систем : практикум / Н. А. Исаева, Д. В. Дворников, К. Е. Елесина. — Москва : Издательский Дом МИСиС, 2024. — 90 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/152717.html> (дата обращения: 16.09.2025)

20. Конкина, В. В. Введение в большие данные и анализ информации : учебное пособие / В. В. Конкина, А. Б. Борисенко, И. Л. Коробова. — Тамбов : Тамбовский государственный технический университет, ЭБС АСВ, 2024. — 81 с. — ISBN 978-5-8265-2749-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/145326.html>— Режим доступа: по подписке.

21. Коньков, К. А. Основы операционных систем : учебник / К. А. Коньков, В. Е. Карпов. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 346 с. — ISBN 978-5-4497-0889-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146366.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. пользователей

22. Кузнецова, Е. В. Управление портфелем проектов как инструмент реализации корпоративной стратегии : учебник для вузов / Е. В. Кузнецова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 177 с. — (Высшее образование). — ISBN 978-5-534-07425-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537265>

23. Липунцов, Ю. П. Управление процессами. Методы управления предприятием с использованием информационных технологий / Ю. П. Липунцов. — 3-е изд. — Саратов : Профобразование, 2024. — 224 с. — ISBN 978-5-4488-0133-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/145934.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. пользователей

24. Литвинская, О. С. Инфокоммуникационные системы и сети : учебное пособие по направлению подготовки 09.03.02 «Информационные системы и технологии» / О. С. Литвинская. — Пенза : Пензенский государственный университет архитектуры и строительства, ЭБС АСВ, 2024. — 114 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/149232.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

25. Мельников А.В. Основы информационной безопасности [Электронный ресурс]: учебное пособие/ Мельников А.В., Зарубин С.В.— Электрон. текстовые данные.— М.: Российский государственный университет правосудия, 2025.— 220 с.— Режим доступа: <https://ipr-smart.ru/152309>.— IPR SMART, по паролю

26. Моргунов, А. В. Операционные системы : учебное пособие / А. В. Моргунов. — Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2024. — 72 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/149525.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

27. Нестеров, С. А. Базы данных : учебник и практикум для вузов / С. А. Нестеров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 258 с. — (Высшее образование). — ISBN 978-5-534-18107-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536687>

28. Онокой, Л. С. Проектирование информационных систем : учебное пособие / Л. С. Онокой, О. А. Морозова, Т. Е. Точилкина. — Москва : Прометей, 2024. — 352 с. — ISBN 978-5-00172-780-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/153542.html> (дата обращения: 16.09.2025). — Режим доступа: для авторизир. пользователей

29. Плахотникова, М. А. Информационные технологии в менеджменте: учебник и практикум для вузов / М. А. Плахотникова, Ю. В. Вертакова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 326 с. — (Высшее образование). — ISBN 978-5-534-07333-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/535632>

30. Поляков, Н. А. Управление инновационными проектами : учебник и практикум для вузов / Н. А. Поляков, О. В. Мотовилов, Н. В. Лукашов. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 384 с. — (Высшее образование). — ISBN 978-5-534-15534-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536478>

31. Проектирование информационных систем: учебник и практикум для вузов / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук ; под общей редакцией Д. В. Чистова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 293 с. — (Высшее образование). — ISBN 978-5-534-15923-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536195>

32. Рабчевский, А. Н. Синтетические данные и развитие нейросетевых технологий : учебное пособие для вузов / А. Н. Рабчевский. —

Москва : Издательство Юрайт, 2024. — 187 с. — (Высшее образование). — ISBN 978-5-534-17716-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/54503>

33. Романова, Ю. Д. Информационные технологии в управлении персоналом : учебник и практикум для вузов / Ю. Д. Романова, Т. А. Винтова, П. Е. Коваль. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2021. — 271 с. — (Высшее образование). — ISBN 978-5-534-09309-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/468991>

34. Рындин, А. А. Современные стандарты информационного взаимодействия систем : учебное пособие / А. А. Рындин, Э. Р. Саргсян. — Москва, Вологда : Инфра-Инженерия, 2024. — 140 с. — ISBN 978-5-9729-1786-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/144722.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

35. Салкин, Д. А. Компьютерные сети. Технологии сетевых интерфейсов. Программное обеспечение и методы диагностики : учебное пособие / Д. А. Салкин, С. Н. Ивлиев, А. В. Пантелеев. — Москва, Вологда : Инфра-Инженерия, 2024. — 220 с. — ISBN 978-5-9729-1917-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/143519.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. пользователей

36. Станкевич, Л. А. Интеллектуальные системы и технологии : учебник и практикум для вузов / Л. А. Станкевич. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 495 с. — (Высшее образование). — ISBN 978-5-534-16238-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536688>

37. Ступина, М. В. Разработка web-ориентированных информационных систем : учебное пособие / М. В. Ступина. — Ростов-на-Дону: Донской государственный технический университет, 2023. — 135 с. —

ISBN 978-5-7890-2137-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/144950.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. пользователей. — DOI: <https://doi.org/10.23682/144950>

38. Тузовский, А. Ф. Объектно-ориентированное программирование: учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2024. — 213 с. — (Высшее образование). — ISBN 978-5-534-16316-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537332>

39. Управление программными проектами : учебное пособие для вузов / В. Е. Гвоздев [и др.] ; под редакцией Р. Ф. Маликова. — Москва : Издательство Юрайт, 2024. — 167 с. — (Высшее образование). — ISBN 978-5-534-14329-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/543929>

40. Управление проектами : учебник и практикум для вузов / А. И. Балашов, Е. М. Рогова, М. В. Тихонова, Е. А. Ткаченко ; под общей редакцией Е. М. Роговой. — Москва : Издательство Юрайт, 2024. — 383 с. — (Высшее образование). — ISBN 978-5-534-00436-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/535573>

41. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для вузов / Д. Ю. Федоров. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 227 с. — (Высшее образование). — ISBN 978-5-534-17323-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539651>

42. Фот, Ю. Д. Анализ уязвимостей и защита программного обеспечения : практикум / Ю. Д. Фот, Е. И. Ларионова. — Оренбург : Оренбургский государственный университет, ЭБС АСВ, 2025. — 141 с. — ISBN 978-5-7410-3391-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/153174.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

43. Цехановский, В. В. Проектирование информационных систем : учебное пособие для СПО / В. В. Цехановский, А. И. Водяхо. — 2-е изд. — Саратов : Профобразование, 2025. — 256 с. — ISBN 978-5-4488-2577-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/152769.html> (дата обращения: 15.01.2026). — Режим доступа: для авторизир. Пользователей

44. Шкурко, В. Е. Управление рисками проекта : учебник для вузов / В. Е. Шкурко ; под научной редакцией А. В. Гребенкина. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 163 с. — (Высшее образование). — ISBN 978-5-534-16836-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/540420>

Приложение

Фрагмент листинга программных модулей

Модуль автоматического контроля парковки
Реализация основных программных модулей

```
import cv2
import numpy as np
import mysql.connector
import hashlib
import logging
from datetime import datetime, timedelta
from fastapi import FastAPI, HTTPException, Depends
from pydantic import BaseModel
from typing import Optional

# =====
# 1. Модуль распознавания номеров
# Назначение: выделение и распознавание государственных регистрационных знаков
# =====
class PlateRecognizer:
    """Распознавание номеров с использованием OpenCV и нейросети (упрощённая версия)"""

    def __init__(self, cascade_path: str = «haarcascade_russian_plate.xml»):
        """Загрузка каскадов Хаара для поиска номера (файл должен существовать)"""
        self.cascade = cv2.CascadeClassifier(cascade_path)
        """В реальном проекте здесь загружается нейросеть для распознавания символов"""

    def preprocess(self, image: np.ndarray) -> np.ndarray:
        """Предобработка изображения: серый, шумоподавление, бинаризация"""
        gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        blurred = cv2.GaussianBlur(gray, (5, 5), 0)
        _, binary = cv2.threshold(blurred, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
        return binary

    def detect_plate(self, image: np.ndarray) -> Optional[tuple]:
        """Поиск области номерного знака на изображении"""
        plates = self.cascade.detectMultiScale(image, scaleFactor=1.1, minNeighbors=5, minSize=(100, 30))
        if len(plates) > 0:
            x, y, w, h = plates[0]
            return (x, y, w, h)
        return None

    def recognize(self, image: np.ndarray) -> str:
        """Основной метод: принимает изображение, возвращает номер или пустую строку"""
        processed = self.preprocess(image)
        plate_region = self.detect_plate(processed)
        if plate_region is None:
            return «»
```

Продолжение приложения

```
x, y, w, h = plate_region
plate_img = processed[y:y+h, x:x+w]
# Здесь должна быть нейросеть для распознавания символов.
# Для демонстрации возвращаем фиктивный номер.
# В реальной системе – вызов модели OCR.
return «A123BC» # заглушка

# =====
# 2. Модуль расчета оплаты
# Назначение: вычисление стоимости парковки и работа с тарифами/абонементом
# =====
class TariffCalculator:
    def __init__(self):
# Тарифы: (стоимость первого часа, последующих, макс за сутки)
self.tariffs = {
    «day»: (50, 30, 300), # дневной тариф (с 8:00 до 20:00)
    «night»: (20, 10, 100) #ночнойтариф
}

    def get_tariff_by_time(self, dt: datetime) -> tuple:
Возвращает тариф в зависимости от времени суток
if 8 <= dt.hour < 20:
    return self.tariffs[«day»]
else:
    return self.tariffs[«night»]

    def calculate_amount(self, entry_time: datetime, exit_time: datetime,
has_abonement: bool = False, free_minutes: int = 30) -> float:

Расчёт суммы к оплате.
:param entry_time: время въезда
:param exit_time: время выезда
:param has_abonement: действует ли абонемент
:param free_minutes: бесплатный период (минут)
:return: сумма в рублях (0, если есть абонемент или бесплатный период)

if has_abonement:
    return 0.0

    duration = (exit_time - entry_time).total_seconds() / 60 # минуты
    if duration <= free_minutes:
        return 0.0

    # Определяем тариф по времени въезда
    first_hour_cost, subsequent_cost, max_day_cost = self.get_tariff_by_time(entry_time)
    hours = (duration - free_minutes) / 60
    if hours <= 1:
        amount = first_hour_cost
    else:
```


Продолжение приложения

```
amount = first_hour_cost + (hours - 1) * subsequent_cost

amount = min(amount, max_day_cost)
return round(amount, 2)

# =====
# 3. Модуль работы с базой данных (MySQL)
# Назначение: подключение, выполнение запросов, управление транзакциями
# =====
class DatabaseManager:
    def __init__(self, host: str, user: str, password: str, database: str):
self.connection = mysql.connector.connect(
    host=host, user=user, password=password, database=database
)
self.cursor = self.connection.cursor(dictionary=True)

    def execute_query(self, query: str, params: tuple = ()):
Выполнение SELECT с возвратом результатов
self.cursor.execute(query, params)
return self.cursor.fetchall()

    def execute_insert(self, query: str, params: tuple = ()):
Вставка/обновление данных
self.cursor.execute(query, params)
self.connection.commit()
return self.cursor.lastrowid

    def start_session(self, plate: str, entry_time: datetime) -> int:
Запись открытого сеанса парковки
query =
INSERT INTO parking_sessions (plate, entry_time, status)
VALUES (%s, %s, 'active')

return self.execute_insert(query, (plate, entry_time))

    def close_session(self, session_id: int, exit_time: datetime, amount: float):
Закрытие сеанса с предоставлением времени выезда и суммы
query =
UPDATE parking_sessions
SET exit_time = %s, amount = %s, status = 'closed'
WHERE id = %s

self.execute_insert(query, (exit_time, amount, session_id))

    def close(self):
self.cursor.close()
self.connection.close()
```

Продолжение приложения

```
# =====
# 4. Модуль управления шлагбаумами (эмуляция через контроллер)
# Назначение: отправка команд на открытие/закрытие шлагбаума
# =====
class BoomGateController:
    def __init__(self, controller_ip: str, port: int):
self.ip = controller_ip
self.port = port
# В реальном проекте здесь устанавливается сетевое соединение

    def open(self, gate_id: int = 1) -> bool:
Открыть шлагбаум. Возвращает True при успехе.
logging.info(f»Открытие шлагбаума {gate_id} на {self.ip}»)
# Эмуляция: всегда успешно
        return True

    def close(self, gate_id: int = 1) -> bool:
logging.info(f»Закрытие шлагбаума {gate_id} на {self.ip}»)
        return True

# =====
# 5. Модуль аудита и логирования
# Назначение: запись всех значимых событий в БД и файл
# =====
class AuditLogger:
    def __init__(self, db: DatabaseManager):
self.db = db
logging.basicConfig(filename='parking_audit.log', level=logging.INFO,
                    format='%(asctime)s - %(levelname)s - %(message)s')

    def log_event(self, event_type: str, user: str, ip_address: str, description: str):
Запись события в таблицу audit_log и в файл
    query =
        INSERT INTO audit_log (event_time, event_type, username, ip_address, description)
        VALUES (%s, %s, %s, %s, %s)

        now = datetime.now()
self.db.execute_insert(query, (now, event_type, user, ip_address, description))
        logging.info(f»{event_type} | {user} | {description}»)

    def log_critical(self, message: str):
Критические события – отдельное уведомление (email в реальной системе)
logging.critical(message)

# =====
# 6. Модуль API для контроллеров (FastAPI)
# Назначение: приём HTTP-запросов от въездных/выездных контроллеров
# =====
```

Продолжение приложения

```
app = FastAPI(title=«Parking Control API»)

# Глобальные экземпляры модулей (в реальном проекте создаются при старте)
db = DatabaseManager(«localhost», «parking_user», «password», «parking_db»)
recognizer = PlateRecognizer()
calculator = TariffCalculator()
gate_ctl = BoomGateController(«192.168.1.100», 5000)
audit = AuditLogger(db)

class EntryRequest(BaseModel):
    image_base64: str # изображение в кодировке base64
    camera_id: int

@app.post(«/api/entry»)
async def process_entry(req: EntryRequest):
    Обработка въезда: распознать номер, открыть шлагбаум, создать сеанс
    # Здесь должно быть декодирование base64 в изображение
    # Для примера используем заглушку
    plate = recognizer.recognize(np.zeros((480,640,3), dtype=np.uint8))
    if plate == «»:
        audit.log_event(«RECOGNITION_FAIL», «system», «camera», «Номер не распознан»)
        raise HTTPException(status_code=400, detail=«Plate not recognized»)

    # Проверка стоп-листа
    blocked = db.execute_query(«SELECT * FROM stoplist WHERE plate=%s», (plate,))
    if blocked:
        audit.log_event(«ENTRY_DENIED», «system», plate, «Автомобиль в стоп-листе»)
        raise HTTPException(status_code=403, detail=«Vehicle in stoplist»)

    session_id = db.start_session(plate, datetime.now())
    gate_ctl.open(1) # открыть въездной шлагбаум
    audit.log_event(«ENTRY», plate, «camera», f»Сеанс {session_id}»)
    return {«status»: «ok», «plate»: plate, «session_id»: session_id}

@app.post(«/api/exit»)
async def process_exit(req: EntryRequest):
    Обработка выезда: распознать номер, рассчитать оплату, открыть шлагбаум
    plate = recognizer.recognize(np.zeros((480,640,3), dtype=np.uint8))
    if plate == «»:
        raise HTTPException(status_code=400, detail=«Plate not recognized»)

    # Поиск активного сеанса
    sessions = db.execute_query(
        «SELECT * FROM parking_sessions WHERE plate=%s AND status='active'«,
        (plate,)
    )
    if not sessions:
        raise HTTPException(status_code=404, detail=«No active session»)
    session = sessions[0]
```

Продолжение приложения

```
exit_time = datetime.now()
entry_time = session[«entry_time»]
# Проверка абонеента (упрощённо)
abonement = db.execute_query(
    «SELECT * FROM abonements WHERE plate=%s AND end_date> %s»,
    (plate, datetime.now())
)
amount = calculator.calculate_amount(entry_time, exit_time, has_abonement=bool(abonement))
if amount > 0:
    # Ожидание оплаты – в реальности здесь интеграция с терминалом
    # Для примера считаем, что оплата прошла успешно
    pass
db.close_session(session[«id»], exit_time, amount)
gate_ctl.open(2) # выездной шлагбаум
audit.log_event(«EXIT», plate, «camera», f»Оплачено {amount} руб.»)
return {«status»: «ok», «plate»: plate, «amount»: amount}
```