

*Министерство образования и науки Республики Татарстан
Государственное автономное профессиональное образовательное
учреждение
«Лениногорский нефтяной техникум»*

КУРСОВОЙ ПРОЕКТ

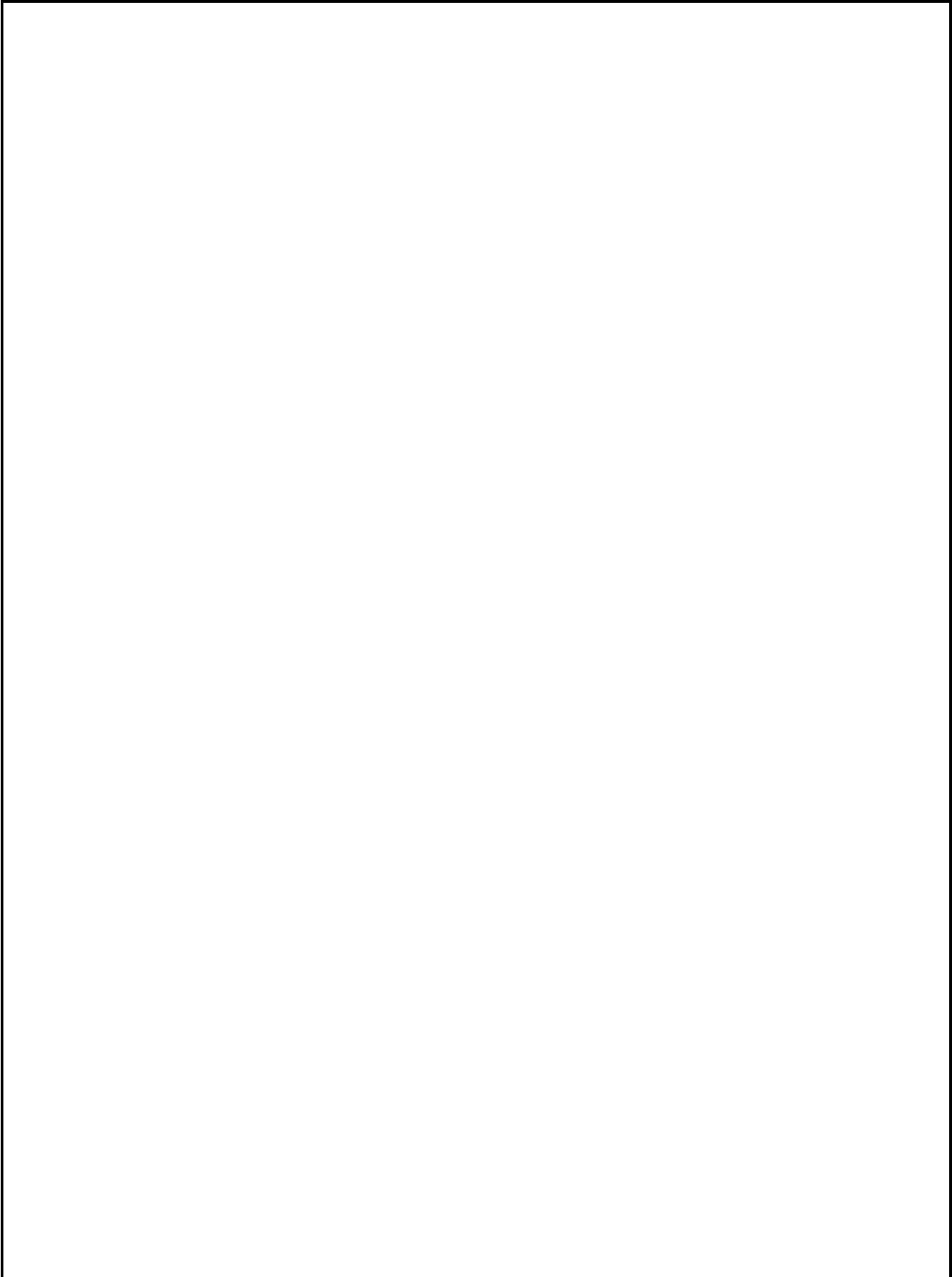
*Разработка системы работы аналитика и диспетчера задач в
отделе IT-аналитики*

ЛНТ О. 09.02.07 03 02. ИС-1-22

Разработал _____ / Бочкарева С.С. /

Руководитель _____ / Переверзова К.С. /

					ЛНТ 0. 09.02.07 03 02. ПЗ			
Изм	Лист	№ докум.	Подп.	Дата	Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики	Лит.	Лист	Листов
Разраб.	Бочкарева С.С.							
Пров.	Переверзова К.С.						2	54
						ИС-1-22		
Н. контр.	Переверзова К.С.							
Утв.	Переверзова К.С.							



					<i>ЛНТ 0. 09.02.07 03 02. ПЗ</i>			
<i>Изм</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>	<i>Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики</i>	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Разраб.</i>		<i>Бочкарева С.С.</i>					<i>3</i>	<i>54</i>
<i>Пров.</i>		<i>Переверзова К.С.</i>						
						<i>ИС-1-22</i>		
<i>Н. контр.</i>		<i>Переверзова К.С.</i>						
<i>Утв.</i>		<i>Переверзова К.С.</i>						

СОДЕРЖАНИЕ

Введение.....	5
1 Описание предметной области информационной системы.....	7
2 Анализ инструментальных средств разработки, используемых при реализации информационной системы.....	9
3 Техническое задание.....	12
4 Проектирование и разработка информационной системы.....	18
4.1 Структурно-функциональное проектирование информационной системы.....	18
4.1.1 Модель движения потоков данных в стандарте IDEFO с дополнительными уровнями декомпозиции.....	18
4.1.2 Моделирование потоков данных (процессов) в стандарте DFD.....	23
4.1.3 Диаграмма «сущность-связь» (ERD).....	24
4.2 Объектно-ориентированное проектирование информационной системы.....	25
4.2.1 Диаграмма вариантов использования.....	25
4.2.2 Диаграмма последовательности.....	28
4.2.3 Логическая модель информационной системы. Диаграмма классов.....	30
4.3 Проектирование и разработка интерфейса.....	31
4.4 Разработка кода информационной системы.....	35
5 Руководство пользователя информационной системы.....	41
Заключение.....	51
Список использованных источников.....	53

					ЛНТ 0. 09.02.07 03 02. ПЗ		
Изм	Лист	№ докум.	Подп.	Дата			
Разраб.		Бочкарева С.С.			Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики	Лит.	Лист
Пров.		Переверзова К.С.					4
						ИС-1-22	
Н. контр.		Переверзова К.С.					
Утв.		Переверзова К.С.					

ВВЕДЕНИЕ

В современном мире информационные технологии играют ключевую роль в развитии экономики, бизнеса и общества в целом. Быстрый технологический прогресс, цифровизация различных отраслей и возросшая потребность в автоматизации процессов создают высокий спрос на инновационные технологические решения. В таких условиях организации, предлагающие современные программные продукты и сервисы, занимают важное место на рынке, помогая компаниям эффективно адаптироваться к быстро меняющимся условиям.

В то же время, в условиях стремительного развития информационных технологий и увеличения объёмов данных, с которыми приходится работать специалистам, эффективность управления внутренними процессами становится ключевым фактором успеха компании. Особенно остро это проявляется в отделах IT-аналитики, который отвечает за обработку информации, взаимодействие с клиентами и координацию выполнения заказов, сталкиваясь с растущей нагрузкой и необходимостью оперативного реагирования на изменения. На данный момент процессы управления задачами и клиентскими запросами в отделе не автоматизированы, что приводит к увеличению времени обработки и риску ошибок.

Актуальность автоматизации работы аналитика и диспетчера задач обусловлена необходимостью повышения качества управления клиентскими запросами и заказами, оптимизации распределения задач и сокращения времени их обработки. Внедрение специализированных информационных систем позволит упростить и систематизировать рабочие процессы, снизить вероятность ошибок, повысить прозрачность выполнения заказов и улучшить коммуникацию внутри команды.

Разработка такой системы требует комплексного подхода и глубокого анализа текущих процессов внутри отдела IT-аналитики. В рамках данного курсового проекта будет проведено исследование существующих методов обработки клиентских запросов и управления задачами, выявлены основные проблемные участки и сформулированы требования к автоматизированной системе.

Особое внимание будет уделено проектированию архитектуры и функциональных возможностей системы, которая должна быть удобной и интуитивно понятной для сотрудников отдела, обеспечивать прозрачность процессов и способствовать оперативному принятию

					ЛНТ 0. 09.02.07 03 02. ПЗ								
Изм	Лист	№ докум.	Подп.	Дата	Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики				Лит.	Лист	Листов		
Разраб	Бочкарева С.С.										5	54	
Пров.	Переверзова К.С.								ИС-1-22				
Н. контр.	Переверзова К.С.												
Утв.	Переверзова К.С.												

решений. Для этого необходимо детально изучить рабочие процессы аналитиков и диспетчеров, определить их потребности и возможные сценарии взаимодействия с системой.

Предметом исследования является автоматизированная система управления клиентскими запросами и задачами в отделе IT-аналитики, направленная на оптимизацию процессов обработки информации и повышения качества обслуживания клиентов.

Объектом исследования выступает отдел IT-аналитики, где осуществляется координация выполнения заказов и взаимодействие с заказчиками.

Ключевым этапом жизненного цикла проекта является анализ текущих бизнес-процессов, выявление узких мест и слабых звеньев, а также определение требований пользователей. Это позволит сформировать чёткое представление о потребностях компании и создать эффективное решение, адаптированное к специфике работы отдела.

Целью проекта является разработка информационной системы, которая автоматизирует работу аналитика и диспетчера задач, минимизирует ручной труд и повысит скорость обработки клиентских заказов, а также улучшит коммуникацию внутри команды.

Для достижения поставленной цели необходимо решить следующие задачи:

1) изучить предметную область и организационную структуру отдела IT-аналитики для понимания особенностей работы и ключевых процессов;

2) выявить проблемы и требования к учёту и диспетчеризации задач, а также к обработке клиентских запросов;

3) определить основные требования к автоматизированной системе управления задачами и клиентскими запросами;

4) проанализировать и обосновать выбор инструментов разработки, включая фреймворк Django и средства реализации учёта и аналитики на Python;

5) выполнить проектирование информационной системы с использованием нотации UML, отражающих архитектуру и функциональность системы;

6) спроектировать и протестировать информационную систему, обеспечивающую удобство работы и прозрачность процессов;

7) оценить эффективность внедрения системы на основе анализа улучшений в работе отдела.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		6

1 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Общество с ограниченной ответственностью «Будущие Информационные Технологии» (далее ООО «БИТ») – динамично развивающаяся компания, специализирующаяся на разработке и внедрении современных IT-решений, направленных на поддержку и автоматизацию бизнес-процессов различных отраслей. Компания оказывает широкий спектр услуг в сфере информационных технологий, включая разработку программного обеспечения, внедрение корпоративных систем, интеграцию цифровых решений, сопровождение проектов и консультационные услуги.

Основными клиентами организации являются коммерческие предприятия, государственные учреждения и промышленные предприятия, нуждающиеся в надёжных и гибких IT-инструментах для повышения эффективности своей деятельности. Коллектив компании состоит из высококвалифицированных специалистов – разработчиков, аналитиков, инженеров и менеджеров, объединённых общей целью – создавать технологические решения, соответствующие современным требованиям бизнеса.

Особое значение в структуре организации имеет отдел IT-аналитики, выполняющий функцию связующего звена между бизнес-требованиями заказчиков и технической реализацией проектов. Именно этот отдел отвечает за систематизацию поступающих запросов, анализ потребностей бизнеса, формирование требований к автоматизации, а также за контроль соответствия создаваемых решений поставленным задачам.

Должность аналитика и диспетчера задач предполагает полный цикл управления проектами – от первичного приёма и анализа клиентских запросов, оценки технической возможности их реализации, разработки детального плана и распределения задач между исполнителями, до контроля выполнения работ и сдачи готового результата заказчику. Такой специалист является ключевым связующим звеном, обеспечивающим качество предоставляемых услуг и эффективное взаимодействие между заказчиками и технической командой.

Основные функции данной должности включают:

- 1) анализ и документирование бизнес-требований, поступающих от клиентов и внутренних подразделений;
- 2) оценка технических возможностей реализации проектов с учётом доступных ресурсов и существующих ограничений;

					ЛНТ О. 09.02.07 03 02. ПЗ			
Изм	Лист	№ докум.	Подп.	Дата				
Разраб.	Бочкарева С.С.				Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики		Лит.	Лист
Пров.	Переверзова К.С.							Листов
								7
								54
Н. контр.	Переверзова К.С.				ИС-1-22			
Утв.	Переверзова К.С.							

3) разработка планов и распределение задач между исполнителями с учётом компетенций и загрузки команды;

4) контроль исполнения задач, управление приоритетами и координация взаимодействия между сотрудниками;

5) ведение коммуникаций с клиентами и смежными подразделениями для уточнения требований и статусов работ;

6) подведение итогов проектов, оформление отчетности и сдача готовых решений заказчикам.

В настоящее время процессы управления задачами и обработки клиентских запросов в отделе IT-аналитики полностью не автоматизированы. Использование ручных списков, электронных таблиц и электронной почты снижает оперативность работы, увеличивает вероятность ошибок, усложняет контроль выполнения задач и создаёт недостаток прозрачности как в работе сотрудников, так и во взаимодействии с клиентами.

Для повышения эффективности работы отдела необходима автоматизированная система, которая позволит систематизировать управление задачами, оптимизировать распределение нагрузки, ускорить обработку запросов и улучшить коммуникацию как внутри команды, так и с клиентами. Такая система обеспечит единую платформу для фиксации всех данных, контроля прогресса выполнения, прозрачного отображения ответственных и сроков, а также автоматическое формирование отчетов и аналитики.

Особенности предметной области, которые необходимо учесть при разработке информационной системы:

1) приём и регистрация клиентских запросов, что обеспечивает фиксацию всех входящих заявок и их систематизацию для последующей обработки;

2) многоуровневое распределение задач с учётом иерархии ответственности, что позволяет создавать задачи и подзадачи, назначать исполнителей и контролировать выполнение;

3) контроль сроков и статусов выполнения, позволяющий отслеживать прогресс задач и своевременно информировать о выполнении или возможных задержках;

4) разграничение прав доступа пользователей, которое гарантирует возможность настройки доступа для разных ролей и обеспечивает безопасность информации;

5) автоматизация документооборота, обеспечивающая централизованное хранение и удобное управление документацией по задачам и проектам;

6) формирование отчетности и аналитики, что позволяет получать сводные данные о выполнении задач, загрузке сотрудников и эффективности работы отдела.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
						8
Изм.	Лист	№ докум.	Подп.	Дата		

2 АНАЛИЗ ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ РАЗРАБОТКИ, ИСПОЛЬЗУЕМЫХ ПРИ РЕАЛИЗАЦИИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

В рамках курсового проекта было принято решение о создании автоматизированной информационной системы для управления задачами и аналитикой, ориентированной на потребности отдела IT-аналитики ООО «БИТ». Для реализации проекта выбран современный стек технологий, обеспечивающий гибкость, масштабируемость и высокую надёжность. Основными инструментальными средствами разработки стали: фреймворк Django, язык программирования Python, система управления базами данных PostgreSQL, а также технологии веб-разметки HTML и каскадные таблицы стилей CSS для создания удобного и интуитивно понятного пользовательского интерфейса.

В качестве основных инструментов для работы с документацией и табличными данными были выбраны Microsoft Word и Microsoft Excel. Эти приложения предоставляют широкий функционал для создания текстовых документов, диаграмм, таблиц, анализа и визуализации данных. В сравнении с альтернативными решениями – такими как Google Docs, LibreOffice Writer или Calc – продукты Microsoft отличаются более высокой производительностью при работе с большими объемами информации, расширенными возможностями редактирования, а также лучшей совместимостью с корпоративными стандартами. В частности, Excel предоставляет мощные средства для построения сводных таблиц, написания формул, автоматизации расчётов с помощью макросов и интеграции с внешними источниками данных. Word, в свою очередь, предлагает богатый набор инструментов для форматирования, структурирования и совместного редактирования документов, что особенно важно при подготовке проектной и технической документации.

Фреймворк Django выбран для разработки серверной части системы благодаря своей высокой надёжности и широкому набору встроенных инструментов. Он обеспечивает удобное управление пользователями, маршрутизацию запросов, взаимодействие с базой данных, а также возможности администрирования и обеспечения безопасности. Django отличается высокой готовностью к промышленной эксплуатации и большим количеством встроенных модулей, что позволяет ускорить процесс разработки, снизить вероятность ошибок и сократить сроки создания и тестирования программного продукта по сравнению с другими фреймворками, такими как Flask или Laravel.

					ЛНТ 0. 09.02.07 03 02. ПЗ							
Изм	Лист	№ докум.	Подп.	Дата	Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики				Лит.	Лист	Листов	
Разраб	Бочкарева С.С.										9	54
Пров.	Переверзова К.С.											
Н. контр.	Переверзова К.С.											
Утв.	Переверзова К.С.								ИС-1-22			

Язык Python выступает основным средством реализации бизнес-логики системы. Он является универсальным языком программирования с простым и читаемым синтаксисом, большим набором встроенных библиотек и развитой экосистемой сторонних модулей. Python широко используется в веб-разработке, анализе данных, автоматизации и искусственном интеллекте, что делает его особенно подходящим для создания комплексных решений, объединяющих управление данными и аналитические функции. В сравнении с языками PHP и JavaScript, Python обеспечивает более высокую скорость разработки, удобство поддержки кода и возможность интеграции с библиотеками для статистического анализа и визуализации.

Система управления базами данных PostgreSQL выбрана для организации хранения и обработки информации благодаря своей надёжности, производительности и поддержке современных стандартов. PostgreSQL является объектно-реляционной СУБД, обеспечивающей поддержку транзакций, целостности данных, сложных SQL-запросов и расширяемых структур хранения. По сравнению с MySQL, PostgreSQL предоставляет более гибкие возможности для аналитических запросов и работы со сложными связями между таблицами. SQLite, напротив, проще в использовании, но не подходит для систем с высокой нагрузкой и многопользовательским доступом. Таким образом, PostgreSQL оптимально подходит для корпоративных систем, где важны надёжность, безопасность и масштабируемость.

HTML и CSS используются для реализации клиентской части системы, обеспечивающей визуальное представление и удобство взаимодействия пользователя с приложением. HTML отвечает за структурирование содержимого веб-страниц, тогда как CSS позволяет реализовать оформление, адаптивность интерфейса и единый визуальный стиль. Их сочетание с Django позволяет создавать интуитивно понятный и современный интерфейс, соответствующий требованиям корпоративного дизайна и пользовательского опыта.

Совместное использование Django, Python и PostgreSQL образует сбалансированный и эффективный технологический стек. Django обеспечивает организацию архитектуры и управление потоками данных, Python отвечает за бизнес-логику и реализацию аналитических функций, а PostgreSQL гарантирует надёжное хранение и обработку информации. В совокупности эти технологии позволяют создать масштабируемую, безопасную и устойчивую систему, способную адаптироваться к росту числа пользователей и объёма данных.

Следует отметить, что использование данного стека требует высокой квалификации разработчиков, обладающих знаниями в области Python-программирования, администрирования баз данных PostgreSQL и веб-технологий. В отличие от готовых платформ, таких как 1С:Предприятие или Bitrix24, разработка собственной системы требует большего времени на проектирование,

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
						10
Изм.	Лист	№ докум.	Подп.	Дата		

реализацию и тестирование, однако обеспечивает более широкие возможности адаптации под специфику бизнес-процессов и последующего расширения функционала.

Таким образом, анализ выбранных инструментальных средств показывает, что сочетание Django, Python и PostgreSQL является оптимальным решением для реализации автоматизированной информационной системы управления задачами и аналитикой. Этот стек технологий обеспечивает баланс между производительностью, безопасностью, гибкостью и удобством разработки, позволяя создать надёжный и функциональный программный продукт.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
						11
Изм.	Лист	№ докум.	Подп.	Дата		

3 ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Техническое задание – это ключевой документ, в котором детально прописаны все требования и задачи, подлежащие выполнению при разработке программного продукта или информационной системы. Он служит руководством для разработчиков, обеспечивает четкое понимание необходимого функционала и устанавливает критерии оценки конечного результата. Техническое задание является основой для планирования, разработки, тестирования и внедрения системы.

Настоящее техническое задание определяет требования к разработке программного продукта – системы аналитики и диспетчера задач, предназначенной для автоматизации управления задачами и анализа процессов в отделе IT-аналитики компании ООО «Будущие Информационные Технологии». Данная система направлена на улучшение контроля над рабочими процессами, повышение эффективности распределения ресурсов и обеспечение своевременного выполнения проектов.

Разработка информационной системы направлена на достижение следующих целей:

- 1) повысить эффективность управления персоналом и задачами;
 - 2) снизить количество ошибок при вводе данных и принятии управленческих решений;
 - 3) обеспечить проверку корректности вводимой информации и предотвратить дублирование данных;
 - 4) организовать мониторинг статусов задач и активности сотрудников в режиме реального времени;
 - 5) автоматизировать формирование аналитических отчетов и графиков;
 - 6) обеспечить прозрачность распределения и исполнения задач между сотрудниками;
 - 7) обеспечить безопасность данных и контроль доступа в системе;
 - 8) интегрировать систему с существующими корпоративными информационными системами и базами данных;
 - 9) автоматизировать уведомления и напоминания для своевременного выполнения задач и обработки заказов;
 - 10) создать удобный и интуитивно понятный интерфейс для пользователей разных уровней.
- Для достижения поставленных целей необходимо выполнить следующие задачи:

					ЛНТ О. 09.02.07 03 02. ПЗ			
Изм	Лист	№ докум.	Подп.	Дата				
Разраб.	Бочкарева С.С.				Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики		Лит.	Лист
Пров.	Переверзова К.С.							Листов
								12
								54
Н. контр.	Переверзова К.С.				ИС-1-22			
Утв.	Переверзова К.С.							

1) провести анализ текущих бизнес-процессов компании ООО «Будущие Информационные Технологии» и собрать требования к системе аналитики и диспетчера задач;

2) спроектировать архитектуру информационной системы с использованием фреймворка Django и базы данных PostgreSQL;

3) разработать модули управления задачами, включающие создание, назначение, мониторинг и завершение задач;

4) создать аналитические инструменты для формирования отчетов и визуальных дашбордов, позволяющих оценивать эффективность работы;

5) внедрить механизмы автоматической проверки вводимых данных на корректность и предотвращение дублирования;

6) обеспечить защиту данных и контроль доступа в соответствии с требованиями информационной безопасности;

7) провести комплексное тестирование системы на предмет функциональности, надежности и удобства пользовательского интерфейса;

8) подготовить полную документацию для пользователей и администрации системы.

Система будет использоваться следующими категориями пользователей:

1) аналитик и диспетчер задач – выполнение и распределение задач, регистрация статуса, контроль исполнения и формирование отчетов;

2) сотрудники – непосредственное выполнение поставленных задач, обновление статуса выполнения;

3) клиенты – возможность создавать обращения и оставлять сообщения в системе.

Для обеспечения эффективного функционирования автоматизированной информационной системы аналитики и диспетчера задач в ООО «Будущие Информационные Технологии» предъявляются следующие ключевые требования:

а) требования к функциональным характеристикам.

Программное решение должно функционировать в виде веб-приложения, доступного через современные браузеры без необходимости установки дополнительного ПО. Централизованная архитектура системы предусматривает единое хранилище всех сущностей: задач, пользователей, статусов, отчетных данных и журналов активности. Система обязана поддерживать разграничение прав доступа на уровне ролей, включая создание и настройку профилей сотрудников, а также разграничение видимости и возможности редактирования объектов.

В состав системы должны входить следующие функциональные и технологические подсистемы:

					ЛНТ О. 09.02.07 ОЗ 02. ПЗ	Лист
						13
Изм.	Лист	№ докум.	Подп.	Дата		

1) управление задачами: пользователи должны иметь возможность создавать задачи, редактировать их содержание, назначать исполнителей, отслеживать статусы выполнения, комментировать и закрывать задачи. Также необходима поддержка вложений и история изменений;

2) аналитика и отчётность: система должна обеспечивать построение графиков, таблиц и отчётов по ряду параметров: сотрудник, проект, срок исполнения, приоритет, статус. Желательно наличие визуализации прогресса в формате диаграмм ганта или круговых диаграмм;

3) экспорт данных: важно наличие функций выгрузки отчётной информации в офисные форматы: docx, xlsx;

4) уведомления: система должна генерировать автоматические уведомления об изменениях в задачах, приближении сроков и новых назначениях. Уведомления могут приходить как внутри системы, так и по e-mail. При этом важно предусмотреть возможность настройки частоты и типов уведомлений.

Также система должна поддерживать дополнительные функции, повышающие качество данных и удобство работы:

1) проверка корректности введённых данных: даты не могут быть в прошлом, приоритет должен быть из доступного списка, недопустимы пустые поля в обязательных параметрах;

2) исключение дублирующих задач: идентичные по названию, проекту и исполнителю задачи должны выявляться автоматически;

3) гибкий поиск и фильтрация: пользователь должен иметь возможность быстро находить информацию по ключевым параметрам – дате, автору, проекту, статусу, роли.

б) требования к надёжности.

Система должна демонстрировать высокую степень надёжности при стандартной эксплуатации. Предусматривается круглосуточная доступность, с минимальным временем отклика и без необходимости участия администратора при стандартных операциях.

Ключевые принципы надёжности:

1) сохранность данных: все действия пользователя, включая редактирование и удаление, должны автоматически фиксироваться в логах. При аварийных сбоях не допускается потеря информации – даже в случае прерывания сессии;

2) резервное копирование: создание резервных копий выполняются ежедневно автоматически, с хранением копий в течение как минимум 7 дней. Восстановление после сбоя должно занимать не более 1 часа, при этом данные возвращаются в последнюю корректную точку;

3) информирование об ошибках: при критических сбоях система отправляет автоматические уведомления администраторам или ответственным за сопровождение ИТ-инфраструктуры.

					ЛНТ О. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		14

Для обеспечения надежной информационной безопасности в системе реализован комплекс технических и программных мер, включающий следующие ключевые механизмы:

1) авторизация пользователей: все пользователи системы обязаны проходить процедуру авторизации с вводом индивидуального пароля. Ввод пароля осуществляется при каждом входе в систему, что исключает возможность несанкционированного доступа в случае ухода пользователя от рабочего места;

2) контроль уровня доступа: при выполнении любых действий, включая операции чтения, записи, удаления и изменения данных, система проводит проверку уровня доступа пользователя. Это обеспечивает разграничение полномочий и минимизацию рисков случайного или умышленного нарушения целостности данных;

3) шифрование данных: для защиты информации как при хранении, так и при передаче по сети применяется криптографическое шифрование. Это обеспечивает конфиденциальность и защищенность данных от несанкционированного доступа, даже в случае их перехвата или утечки.

Технические и программные средства системы должны соответствовать требованиям действующих стандартов и обеспечивать устойчивую работу в условиях воздействия различных факторов внешней среды:

1) электробезопасность и заземление: оборудование должно быть подключено с соблюдением требований ГОСТ Р 50571.22-2000, регламентирующего системы заземления и электробезопасности в информационно-технических комплексах;

2) пожарная безопасность: при установке и эксплуатации оборудования необходимо соблюдать нормы ГОСТ 12.1.004-91, определяющего требования к пожарной безопасности, включая использование негорючих материалов и наличие первичных средств пожаротушения.

Устойчивость оборудования к воздействиям окружающей среды обеспечивается в соответствии со следующими параметрами:

1) оборудование должно сохранять работоспособность при колебаниях напряжения в диапазоне от 160 до 250 В;

2) допустимый температурный режим эксплуатации составляет от +10 до +35 °С;

3) относительная влажность воздуха в помещении не должна превышать 75% при нормальных условиях;

4) устойчивость к вибрационным воздействиям должна соответствовать допустимым значениям, установленным производителем оборудования.

в) условия эксплуатации.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		15

Для корректной и стабильной работы системы пользователям необходимо обеспечить соответствие ряду технических, программных и организационных требований.

Аппаратные условия:

- 1) используются персональные компьютеры или ноутбуки под управлением Windows 10 или более новой версии;*
- 2) объем оперативной памяти – не менее 4 гб;*
- 3) свободное дисковое пространство – от 20 гб;*
- 4) наличие стабильного интернет-соединения (желательно с пропускной способностью от 5 мбит/с).*

Программное обеспечение:

- 1) современные веб-браузеры с актуальными обновлениями: Google Chrome, Microsoft Edge, Яндекс.Браузер;*
- 2) офисные программы (для открытия отчётов и документов);*
- 3) антивирус с актуальной базой сигнатур и регулярным сканированием.*

Организационные условия эксплуатации:

- 1) перед внедрением системы обязательно обучение сотрудников с предоставлением кратких инструкций;*
- 2) поддержка пользователей должна обеспечиваться через внутреннюю ИТ-службу или техническую поддержку разработчика;*
- 3) резервное копирование данных должно осуществляться по регламенту, доступ к архивам ограничен;*
- 4) все данные, включая чувствительную информацию, должны храниться в пределах локальной сети предприятия либо в сертифицированном защищённом облачном хранилище.*

Лицензирование: все программные компоненты, включая операционные системы, серверное ПО, базы данных, сторонние библиотеки и API-интерфейсы, должны иметь актуальные лицензии. Пиратские и устаревшие версии категорически не допускаются.

г) требования к составу и параметрам технических средств.

Минимальные клиентские устройства, подходящие для использования системы:

- 1) процессор не ниже по производительности Intel Core i3 или AMD Ryzen 3;*
- 2) оперативная память от 2 до 4 ГБ;*
- 3) минимум 5 ГБ свободного пространства на диске;*
- 4) возможность подключения по Ethernet или Wi-Fi;*
- 5) графическая подсистема – встроенная, без необходимости дискретной видеокарты.*

					<i>ЛНТ 0. 09.02.07 03 02. ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		<i>16</i>

При локальной серверной установке (если нет размещения в облаке) рекомендуются следующие параметры:

- 1) процессор: Intel Core i5 или выше;*
- 2) оперативная память: от 8 ГБ (оптимально – 16);*
- 3) твердотельный накопитель (SSD): от 256 ГБ;*
- 4) операционная система: Windows 10 или более новой версии либо дистрибутив Linux (Ubuntu LTS, CentOS, Debian).*

Система должна обеспечивать стабильную работу при одновременной нагрузке от нескольких десятков пользователей, без задержек и потери данных. При необходимости поддерживается масштабирование (например, горизонтальное – за счёт подключения дополнительных серверов).

д) специальные требования: отсутствуют.

					<i>ЛНТ 0. 09.02.07 03 02. ПЗ</i>	<i>Лист</i>
						<i>17</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		

4 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ

4.1 Структурно-функциональное проектирование информационной системы

4.1.1 Модель движения потоков данных в стандарте IDEF0 с дополнительными уровнями декомпозиции

Разработка информационной системы для автоматизации деятельности аналитика и диспетчера задач в ООО «БИТ» предполагает проведение детального анализа бизнес-процессов и их описание в структурированной форме. Для наглядного представления функциональных взаимосвязей и последовательности выполнения процессов используется методология IDEF0, которая является одним из наиболее эффективных и широко применяемых инструментов моделирования бизнес-процессов.

Метод IDEF0 позволяет отразить систему в виде иерархии функциональных блоков, каждый из которых описывает конкретный процесс и его взаимодействие с другими элементами через четыре типа связей: вход, выход, управление и механизм. Каждый блок можно рассматривать как логическую единицу, в которой входные данные преобразуются в выходные результаты под воздействием управляющих факторов и при использовании определённых ресурсов. Такой подход обеспечивает ясное и формализованное представление структуры работы системы.

Функциональная модель IDEF0 позволяет детально описать все ключевые процессы, выполняемые в отделе IT-аналитики, включая приём и регистрацию заявок, их анализ и распределение, выполнение задач сотрудниками, контроль их выполнения, а также формирование аналитических отчётов. Основные элементы модели, представленные на рисунке 1, включают:

1) входящие стрелки – исходные данные, поступающие в систему (заявки пользователей, сведения о проектах, данные о сотрудниках и параметрах задач);

2) исходящие стрелки – результаты работы системы (зарегистрированные и выполненные заявки, уведомления, аналитические отчёты);

3) управляющие стрелки – нормативно-методологическую основу процессов (регламенты компании, политику безопасности, правила обработки данных);

					ЛНТ 0. 09.02.07 03 02. ПЗ			
Изм	Лист	№ докум.	Подп.	Дата				
Разраб.	Бочкарева С.С.				Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики		Лит.	Лист
Пров.	Переверзова К.С.							18
Н. контр.	Переверзова К.С.						ИС-1-22	
Утв.	Переверзова К.С.							
							Листов	54

4) механизмы – ресурсы, обеспечивающие выполнение функций (сотрудники отдела IT-аналитики, программные модули, базы данных, вычислительные ресурсы).

Использование методологии IDEFO при проектировании информационной системы позволяет выявить взаимосвязи между процессами, определить роли участников и механизмы взаимодействия, а также обнаружить потенциальные узкие места и направления для оптимизации. Такой подход обеспечивает целостное понимание функционирования системы, повышает её прозрачность, управляемость и эффективность.

Основной блок верхнего уровня, представленный на рисунке 1, отражает информационную систему аналитика и диспетчера задач ООО «Будущие Информационные Технологии», объединяющую все ключевые процессы – от регистрации заявок клиентов до формирования аналитических отчётов и контроля выполнения задач.

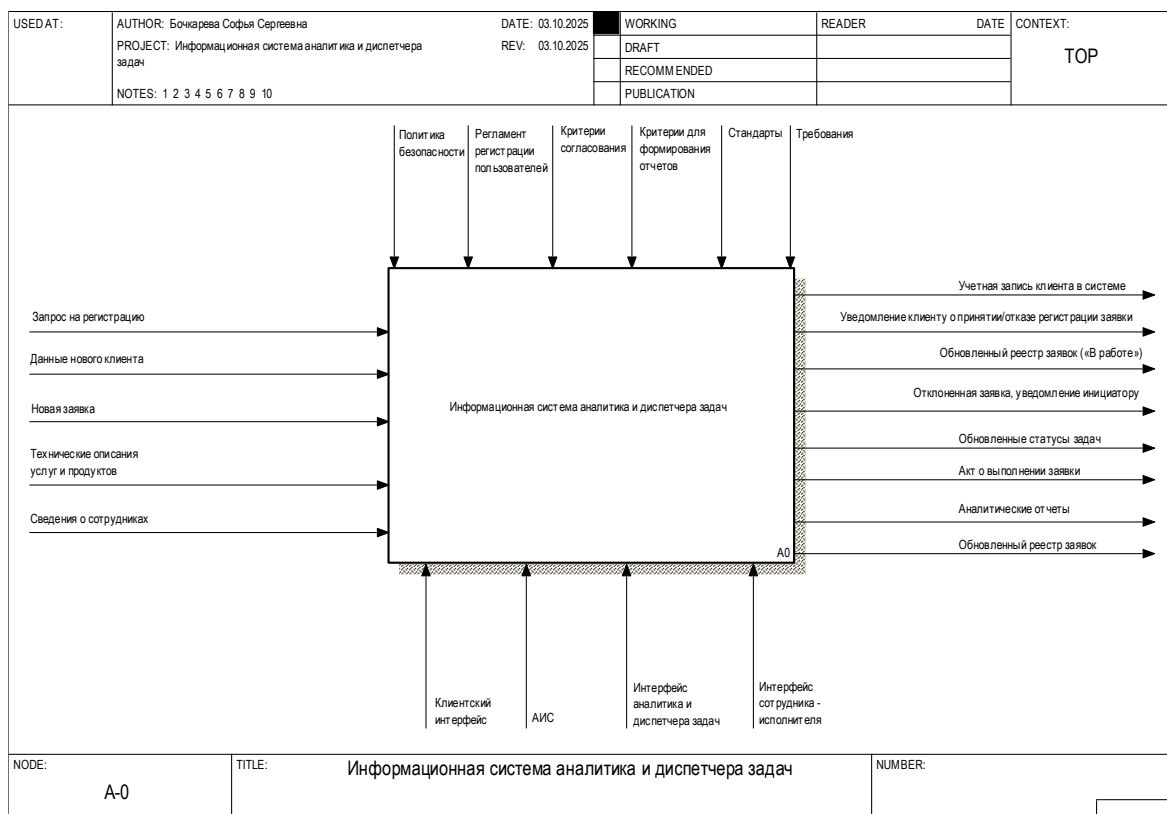


Рисунок 1 Модель движения потоков данных в нотации IDEFO. Контекстная диаграмма

На диаграмме представлена контекстная схема функционирования информационной системы аналитики и диспетчера задач, которая отражает её ключевые компоненты, информационные потоки, управляющие воздействия и механизмы реализации. Модель демонстрирует, как система взаимодействует с внешними и внутренними элементами предприятия, обеспечивая автоматизацию процессов постановки, распределения, контроля и анализа задач.

Входными данными для системы являются запросы на регистрацию новых пользователей, сведения о клиентах, вновь созданные заявки, технические описания услуг и продуктов, а также

информация о сотрудниках и их компетенциях. Эти данные поступают в центральный блок – информационную систему аналитики и диспетчера задач, где проходят обработку, анализ и преобразование в структурированные записи, необходимые для дальнейшего управления процессами.

В верхней части диаграммы указаны элементы управления, задающие правила функционирования системы: политика информационной безопасности, регламенты регистрации пользователей, внутренние стандарты и критерии согласования заявок, требования к отчётности и нормы обслуживания. Эти управляющие воздействия обеспечивают корректное выполнение процессов, контроль качества, защиту данных и соблюдение корпоративных процедур.

Механизмы реализации включают программно-технические средства и участников, обеспечивающих работу системы: клиентский интерфейс для подачи и отслеживания заявок, автоматизированную информационную систему (далее АИС), интерфейс аналитика и диспетчера для распределения задач и контроля их выполнения, а также интерфейс сотрудника-исполнителя, позволяющий фиксировать результаты работы. Совокупность этих компонентов обеспечивает интеграцию всех этапов жизненного цикла задачи в едином информационном пространстве.

Выходные данные системы представляют собой результаты обработки информации: созданные учётные записи клиентов, уведомления о результатах регистрации и рассмотрения заявок, обновлённый реестр заявок (включая статусы «в работе», «выполнено» или «отклонено»), акты выполненных работ, а также аналитические и статистические отчёты для руководства. Эти данные обеспечивают прозрачность процессов, обратную связь с клиентами и сотрудниками, а также формируют основу для принятия управленческих решений.

Таким образом, представленная контекстная диаграмма иллюстрирует комплексный подход к управлению задачами в ООО «БИТ», отражая взаимодействие всех ключевых участников и элементов системы. Она демонстрирует, как автоматизация процессов способствует повышению эффективности, сокращению ошибок, прозрачности и управляемости бизнес-процессов компании, а также улучшает качество обслуживания клиентов.

Для более детального представления логики функционирования информационной системы выполнена декомпозиция основного блока контекстной диаграммы, представленная на рисунке 2. На данном уровне уточняются ключевые процессы – от приёма и регистрации заявок до контроля их выполнения и формирования отчетности, что позволяет более полно отразить структуру и взаимосвязь функций системы. Такой подход позволяет более полно отразить структуру системы, взаимосвязь её функций и ответственных участников, а также выявить точки оптимизации процессов для дальнейшего совершенствования работы компании.

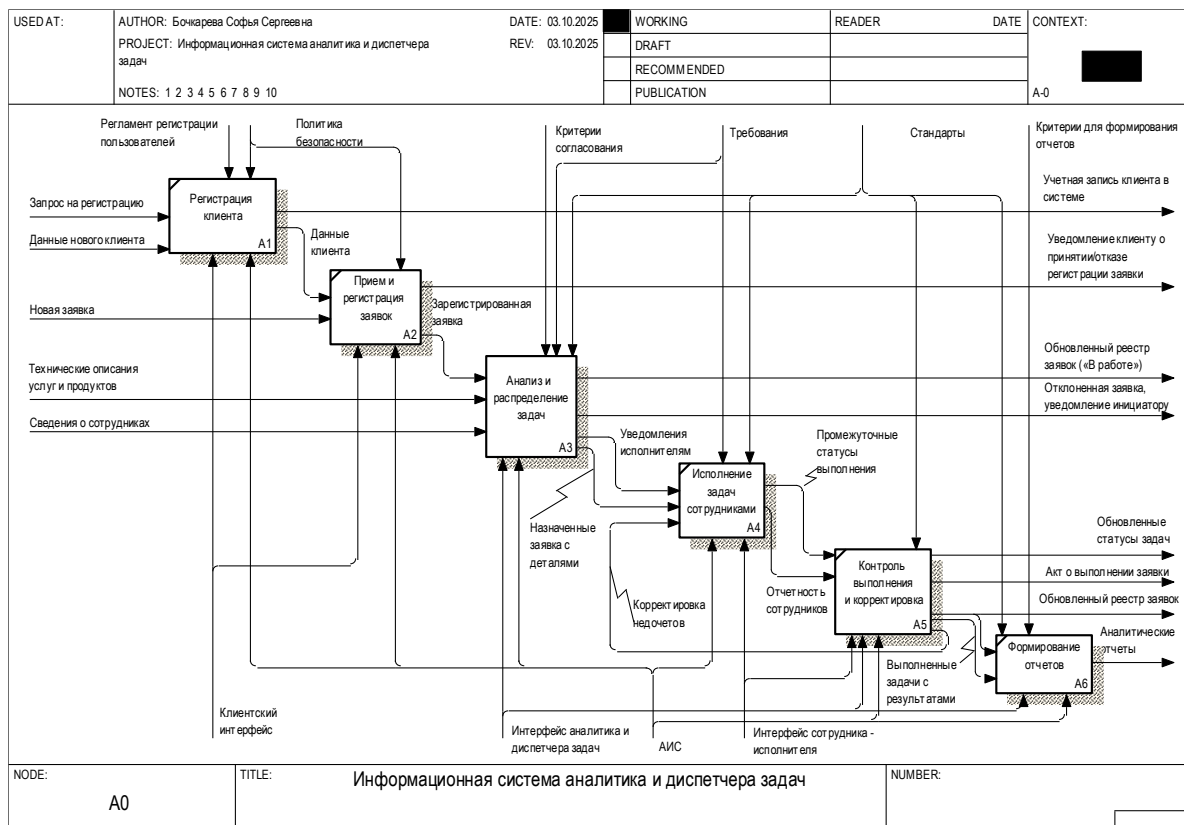


Рисунок 2 Диаграмма декомпозиции первого уровня в нотации IDEFO

Декомпозиция основной функции, представленная на рисунке 2, позволяет разложить общую задачу системы на логически завершённые процессы и наглядно показать последовательность её работы. Общая функция разбита на следующие ключевые процессы:

- 1) регистрация клиента – приём данных нового пользователя и подготовка информации для последующей обработки;
- 2) приём и регистрация заявок – обработка поступивших заявок, ввод технических описаний и сведений о сотрудниках;
- 3) анализ и распределение задач – оценка требований и стандартов, распределение задач между исполнителями, уведомление сотрудников;
- 4) исполнение задач сотрудниками – выполнение назначенных действий, корректировка ошибок, формирование отчётности;
- 5) контроль выполнения и корректировка – проверка результатов работы, внесение изменений, обновление реестра заявок;
- 6) формирование отчётов – создание аналитических и статистических отчётов на основе актуальных данных о выполнении задач.

Такое разбиение позволяет наглядно представить распределение функций, последовательность действий системы и взаимодействие с внешними объектами, включая регламенты, политику безопасности, технические описания и пользовательские интерфейсы.

Более детальная декомпозиция процесса «Анализ и распределение задач» показана на рисунке 3, где можно проследить этапы оценки требований, распределения задач и уведомления исполнителей, что обеспечивает эффективное управление выполнением каждой заявки.

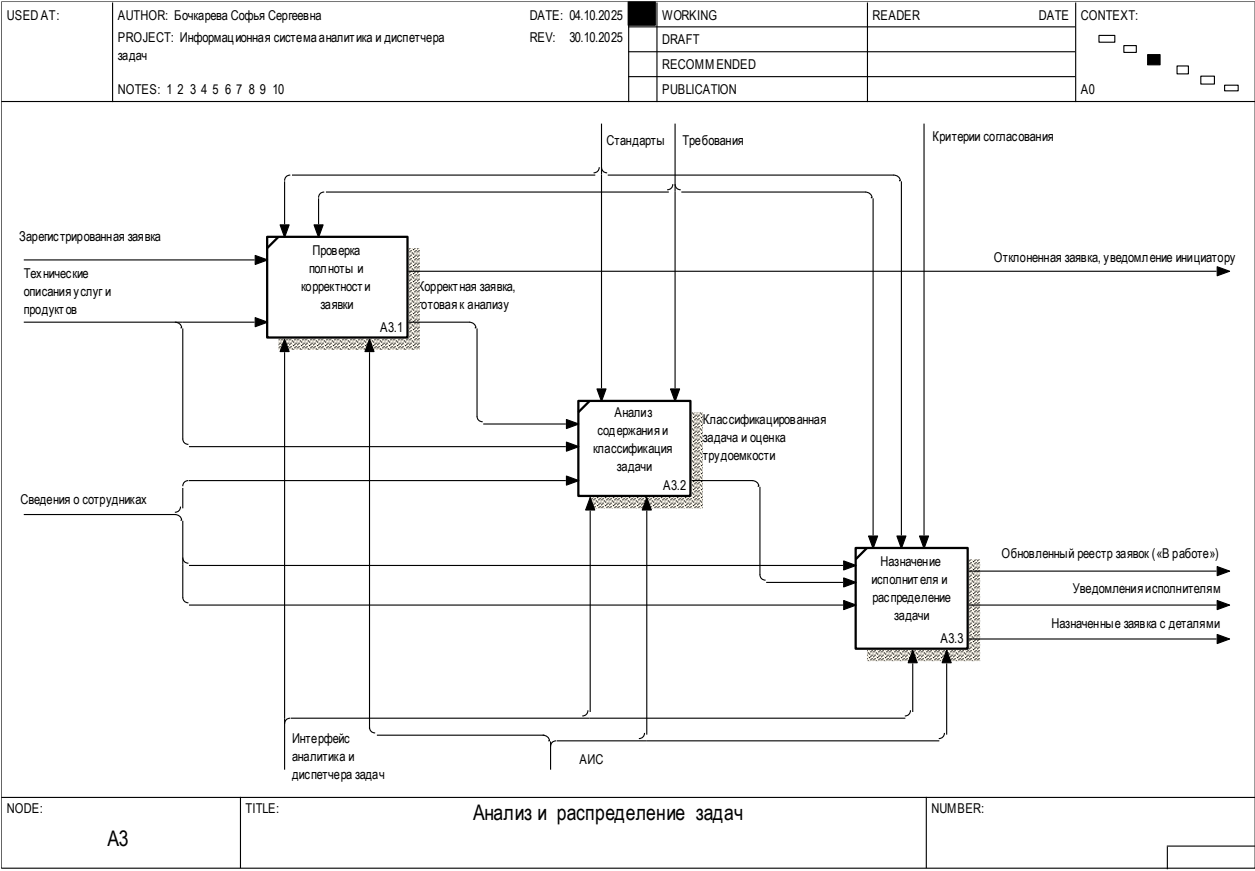


Рисунок 3 Декомпозиция блока «Анализ и распределение задач» в нотации IDEFO

Представленная на рисунке 3 декомпозиция процесса «Анализ и распределение задач» отражает последовательность обработки заявок в информационной системе аналитика и диспетчера задач. Процесс начинается с регистрации заявки, содержащей техническое описание услуги или продукта, а также сведения о сотрудниках, участвующих в выполнении работ. После регистрации заявка поступает в обработку, включающую несколько основных этапов:

- 1) проверка полноты и корректности заявки – поступившая заявка проверяется на полноту и корректность заполнения. В случае обнаружения ошибок или отсутствующих данных заявка может быть возвращена инициатору для доработки либо отклонена;
- 2) анализ содержания и классификация задачи – корректная заявка проходит детальный анализ содержания. Проводится классификация по типу задачи, определяется её срочность, приоритетность и сложность выполнения. Также осуществляется проверка соответствия установленным стандартам и требованиям предприятия;
- 3) назначение исполнителя и распределение задачи – классифицированная и описанная задача передаётся в центр назначения исполнителей. задачи распределяются между сотрудниками

Внешними сущностями являются:

- 1) клиент: инициирует процесс, предоставляя заявки и получая акты о выполненных работах;*
- 2) сотрудники: выполняют задачи, назначенные по заявкам;*
- 3) аналитик и диспетчер задач: читает заявки, согласовывает, создают отчеты, контролируют выполнение заявок.*

Хранилищами данных являются:

- 1) заявки: содержат информацию о клиентских запросах, ожидающих обработки или находящихся в процессе выполнения;*
- 2) отчеты: содержат итоговую информацию о выполненных задачах и аналитические данные, сформированные системой.*

4.1.3 Диаграмма «Сущность–связь» (ERD)

Для моделирования структуры данных в информационной системе используется графический инструмент – диаграмма «сущность–связь» (ERD, Entity–Relationship Diagram). Эта диаграмма позволяет наглядно представить и систематизировать ключевые элементы системы, включая основные сущности, такие как сотрудники, объекты, процессы, клиенты и задачи, а также связи между ними.

Основными компонентами ER-диаграммы являются:

- 1) сущности, которые отображают объекты или элементы системы;*
- 2) атрибуты, характеризующие свойства этих сущностей;*
- 3) связи, показывающие взаимодействие между сущностями.*

Такой подход обеспечивает четкое понимание организации данных, выявление взаимосвязей между различными элементами системы и упрощает проектирование базы данных.

Применение ER-диаграмм способствует значительному повышению качества и точности проектирования, поскольку позволяет заранее определить структуру информации, ключевые атрибуты сущностей, типы связей (один–к–одному, один–ко–многим, многие–ко–многим) и оптимальные способы хранения и обработки данных. Это, в свою очередь, обеспечивает эффективную автоматизацию рабочих процессов отдела, упрощает управление информацией и повышает согласованность работы всех компонентов системы.

На рисунке 5 представлена ER-диаграмма информационной системы, разработанной для автоматизации и оптимизации эффективной деятельности аналитика и диспетчера задач в отделе IT-аналитики.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		24

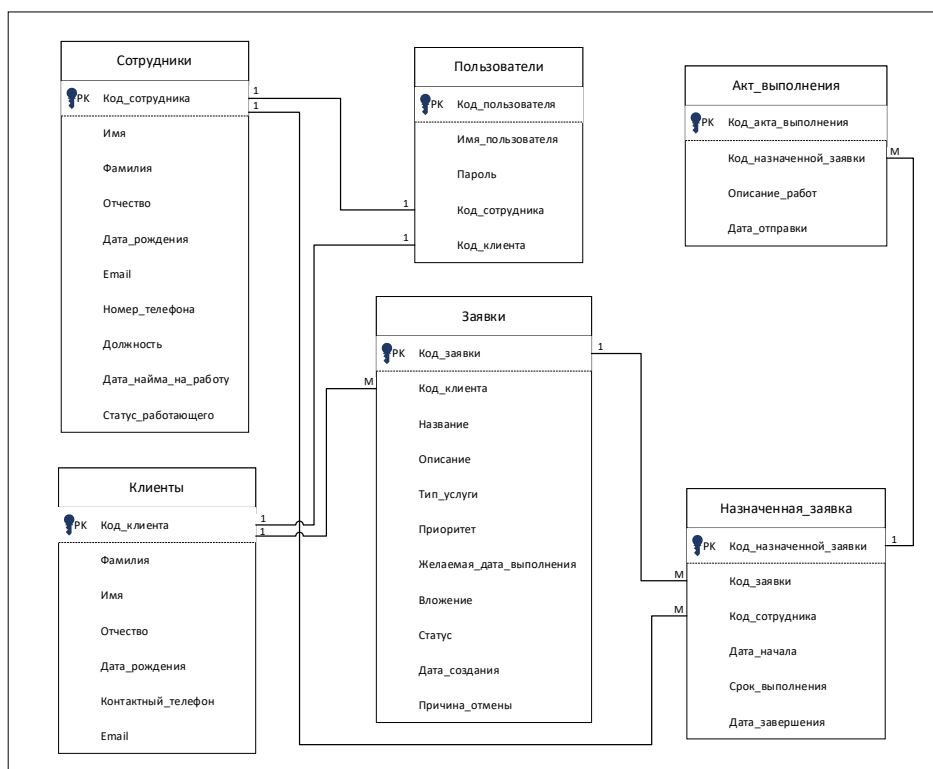


Рисунок 5 Диаграмма «сущность-связь»

На рисунке представлена ER-диаграмма информационной системы, показывающая основные сущности и их взаимосвязи:

- 1) пользователи – содержат данные о пользователях системы, включая логин, пароль и связь с сотрудником;
- 2) сотрудники – включают личную информацию, должность и дату найма каждого сотрудника;
- 3) клиенты – содержат сведения о клиентах системы, включая их личные данные;
- 4) заявки – информация о заявках клиентов на услуги;
- 5) назначенные заявки – фиксируют распределение заявок между сотрудниками с указанием даты начала, срока и фактической даты выполнения работ;
- 6) акты выполнения – содержат описание выполненных работ и дату отправки акта клиенту, связаны с назначенными заявками.

4.2 Объектно-ориентированное проектирование информационной системы

4.2.1 Диаграмма вариантов использования

Диаграмма вариантов использования (use case diagram) – это визуальное средство моделирования, которое показывает отношения между актерами (внешними пользователями или системами) и вариантами использования – функциональными действиями, которые система

выполняет от их имени. Она является базовой концептуальной моделью системы, применяемой на начальных этапах разработки для понимания её основных возможностей и взаимодействия с пользователями.

Цели создания диаграммы вариантов использования включают:

1) определение границ и контекста предметной области. Диаграмма помогает понять, что именно входит в сферу ответственности системы и какие внешние участники с ней взаимодействуют;

2) формулирование общих требований к функциональному поведению. Это позволяет представить, какие функции и сценарии использования должна поддерживать система, и как пользователи будут с ней взаимодействовать;

3) разработка исходной концептуальной модели, которая станет основой для дальнейшего проектирования, включая создание логических и физических моделей системы;

4) подготовка документации для взаимодействия разработчиков с заказчиками и пользователями. По диаграмме легко объяснить, как устроена система и какие задачи она решает, что облегчает согласование требований.

Диаграмма вариантов использования представлена на рисунке 6.

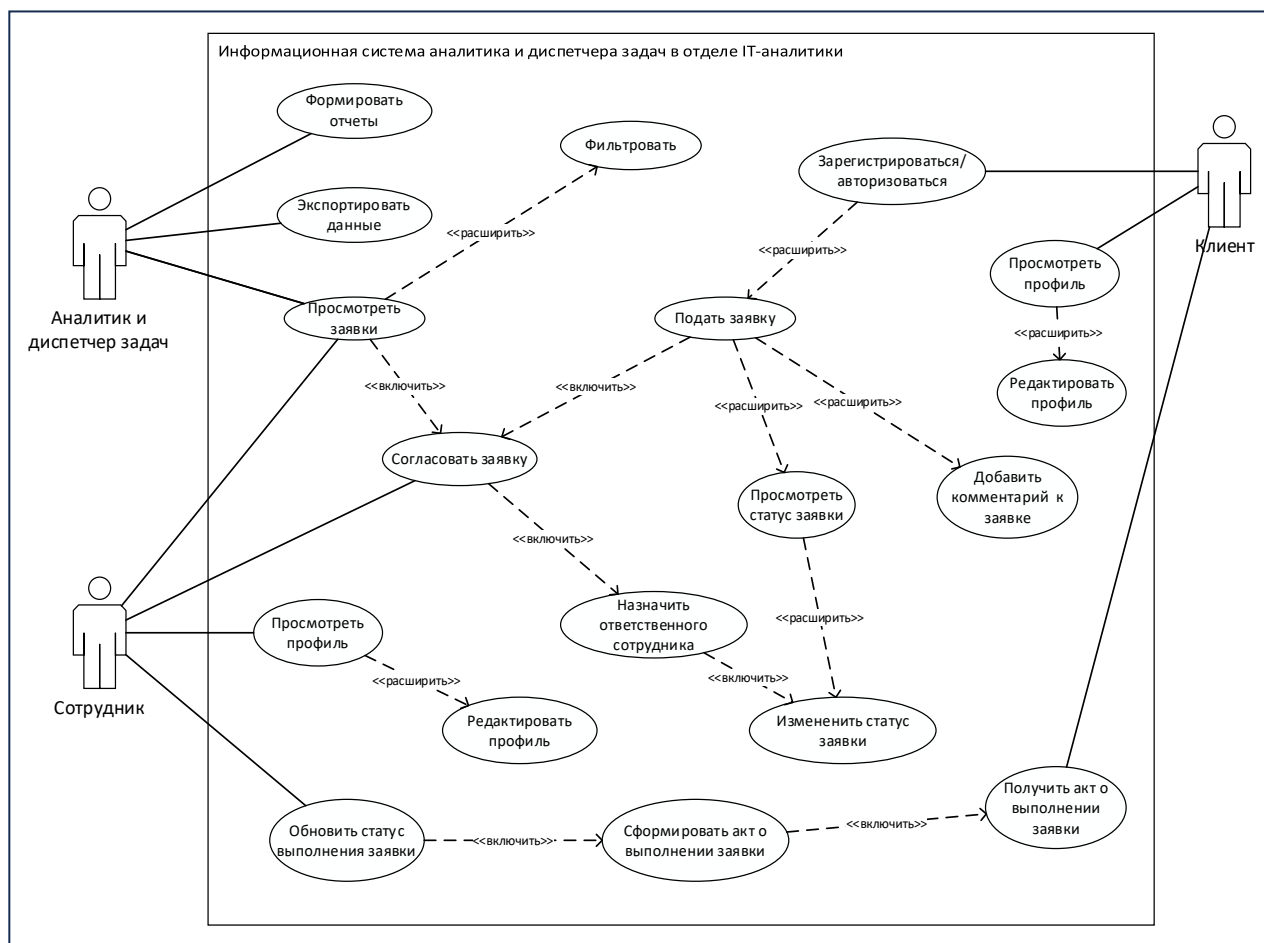


Рисунок 6 Диаграмма вариантов использования

В информационной системе четыре актера: аналитик и диспетчер задач, сотрудник и клиент. Как видно из диаграммы – все процессы автоматизированы, кроме непосредственно выполнения заявки сотрудником.

Описательная спецификация диаграммы вариантов использования информационной системы аналитика и диспетчера задач в отделе IT-аналитики представлена в таблице 1.

Таблица 1 Описательная спецификация диаграммы вариантов использования информационной системы аналитика и диспетчера задач в отделе IT-аналитики

Раздел	Описание
1	2
Краткое описание	Клиенты создают заявки и отслеживают их статус. Аналитик и диспетчер задач распределяет задачи между сотрудниками, контролирует выполнение и формирует отчёты для оценки эффективности работы отдела. Сотрудники выполняют назначенные задачи, фиксируют результаты в системе, после чего задачи считаются завершенными. Автоматизация процессов повышает эффективность работы, уменьшает вероятность ошибок и ускоряет обработку заявок.
Субъекты	Аналитик и диспетчер задач, сотрудник, клиент.
Предусловия	Клиент создаёт заявку через систему. Система имеет доступ к базе данных сотрудников, клиентов и задач. Аналитик авторизован в системе и имеет права на распределение задач и просмотр отчётов.
Основной поток	1) клиент создаёт и отправляет заявку через систему; 2) аналитик и диспетчер задач просматривает новые заявки и распределяет их между сотрудниками; 3) сотрудник выполняет назначенную задачу, фиксирует результаты выполнения в системе; 4) аналитик контролирует статус выполнения задач и может корректировать их распределение; 5) после завершения задачи система обновляет её статус, и клиент получает уведомление о выполнении; 6) аналитик формирует отчёты по выполненным задачам и эффективности сотрудников.

Продолжение таблицы 1

1	2
Альтернативный поток	<p>1) аналитик может редактировать или удалять задачи, если требуется изменить распределение или статус;</p> <p>2) сотрудник может запрашивать уточнение по задаче через комментарии в системе;</p> <p>3) система позволяет фильтровать задачи по дате, статусу, сотруднику или приоритету;</p> <p>4) клиент может отменить заявку до назначения задачи сотруднику.</p>
Постусловия	<p>Все выполненные задачи фиксируются в системе с указанием результатов. Статусы задач обновлены, клиент уведомлён о выполнении. Аналитик получает готовые отчёты по прогрессу и эффективности работы сотрудников, которые могут использоваться для анализа и планирования.</p>

4.2.2 Диаграмма последовательности

Диаграмма последовательности – это графический инструмент, используемый для описания динамики работы информационной системы и моделирования взаимодействий. Она демонстрирует, как объекты взаимодействуют между собой во времени, обмениваются сообщениями и выполняют определённые функции, что позволяет наглядно представить логику выполнения процессов и последовательность операций. Такая диаграмма особенно полезна на этапе проектирования, когда важно понять, каким образом происходит взаимодействие между пользователем, системой и её внутренними компонентами.

Ключевыми элементами диаграммы последовательности являются:

1) объекты – прямоугольники с названиями, обозначающие участников взаимодействия (например, пользователь, система, модуль базы данных, интерфейс администратора);

2) линии жизни – вертикальные пунктирные линии, отражающие существование объекта во времени и его участие в процессе;

3) активности – прямоугольники на линиях жизни, отображающие периоды, в течение которых объект выполняет определённые действия или обрабатывает запрос;

4) сообщения – стрелки между объектами, демонстрирующие передачу данных, вызовы функций, запросы и ответы между элементами системы.

Диаграмма последовательности для процесса обработки заявки представлена на рисунке 7.

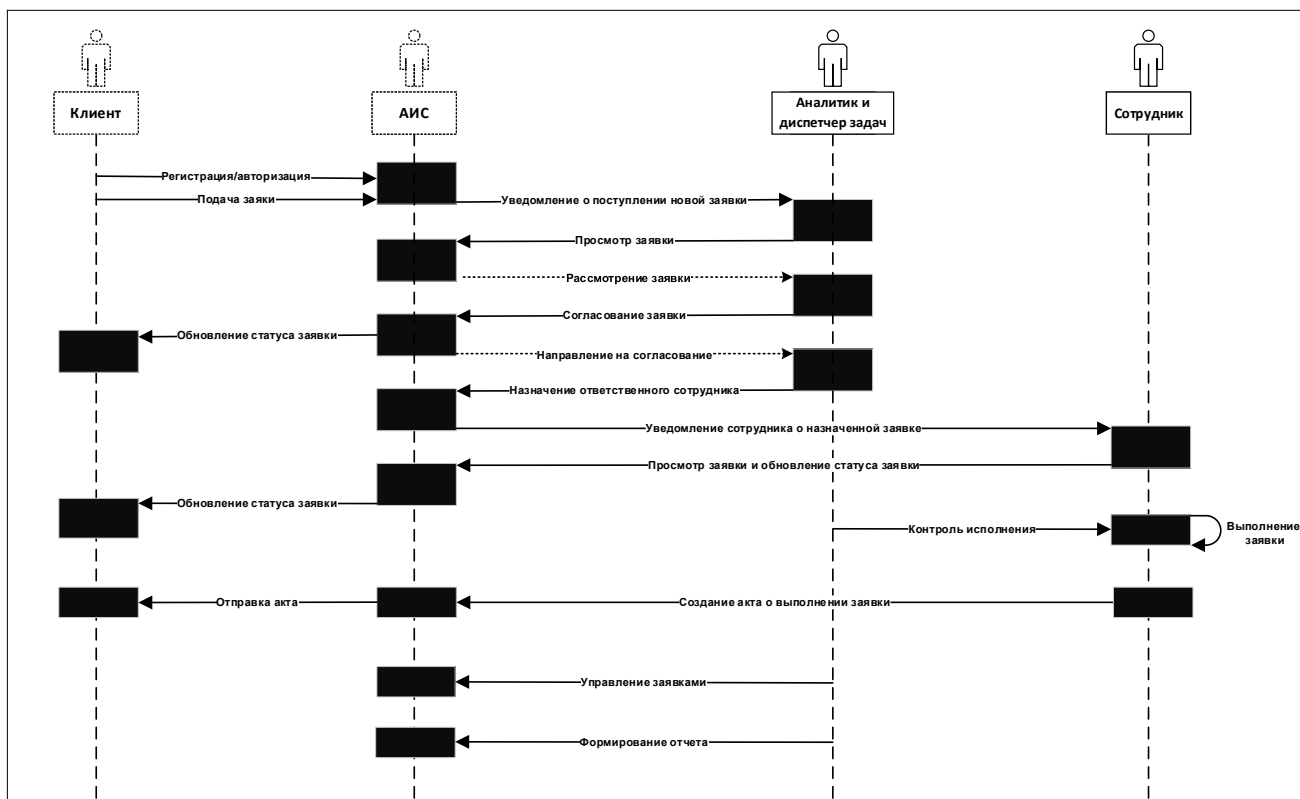


Рисунок 7 Диаграмма последовательности

Автоматизированная информационная система выполняет следующие функции:

- 1) принимает заявки от клиента;
- 2) уведомляет аналитика диспетчерского отдела о поступлении новой заявки;
- 3) обеспечивает возможность просмотра и рассмотрения заявок;
- 4) направляет заявку на согласование и назначает ответственного сотрудника;
- 5) уведомляет сотрудника о назначенной заявке;
- 6) обновляет статус заявки;
- 7) создает акт о выполнении заявки.

Действующее лицо «Клиент» выполняет следующие действия:

- 1) регистрирует или авторизуется в системе;
- 2) подает заявку на проведение работ;
- 3) отслеживает обновление статуса заявки.

Действующее лицо «Аналитик и диспетчер задач» выполняет следующие действия:

- 1) получает уведомления о новых заявках;
- 2) просматривает поступившие заявки;
- 3) рассматривает заявку и направляет её на согласование;
- 4) назначает ответственного сотрудника для выполнения заявки;
- 5) формирует отчетную ведомость.

Действующее лицо «Сотрудник» выполняет следующие действия:

- 1) получает уведомления о назначенных ему заявках;
- 2) просматривает свои заявки и обновляет их статус;
- 3) создает акт о выполнении заявки.

4.2.3 Логическая модель информационной системы. Диаграмма классов

Диаграмма классов является одним из ключевых инструментов объектно-ориентированного моделирования, позволяя отразить статическую структуру информационной системы. Она показывает, какие классы входят в состав системы, какие атрибуты и операции они содержат, а также каким образом эти классы взаимосвязаны между собой.

Данный тип диаграммы используется для построения логической модели, которая служит основой проектирования программного обеспечения. С её помощью можно наглядно описать основные сущности предметной области, выделить их свойства, определить ответственность каждого класса и зафиксировать отношения, такие как ассоциации, зависимости, обобщения и реализации.

Каждый класс на диаграмме изображается в виде прямоугольника с секциями, где указываются его название, характеристики (поля) и функции (методы). Интерфейсы позволяют формализовать общий набор операций, которые должны поддерживаться конкретными классами. Для организации структуры классы и интерфейсы могут объединяться в пакеты.

Диаграмма классов, отражающая логическую модель рассматриваемой системы, представлена на рисунке 8.

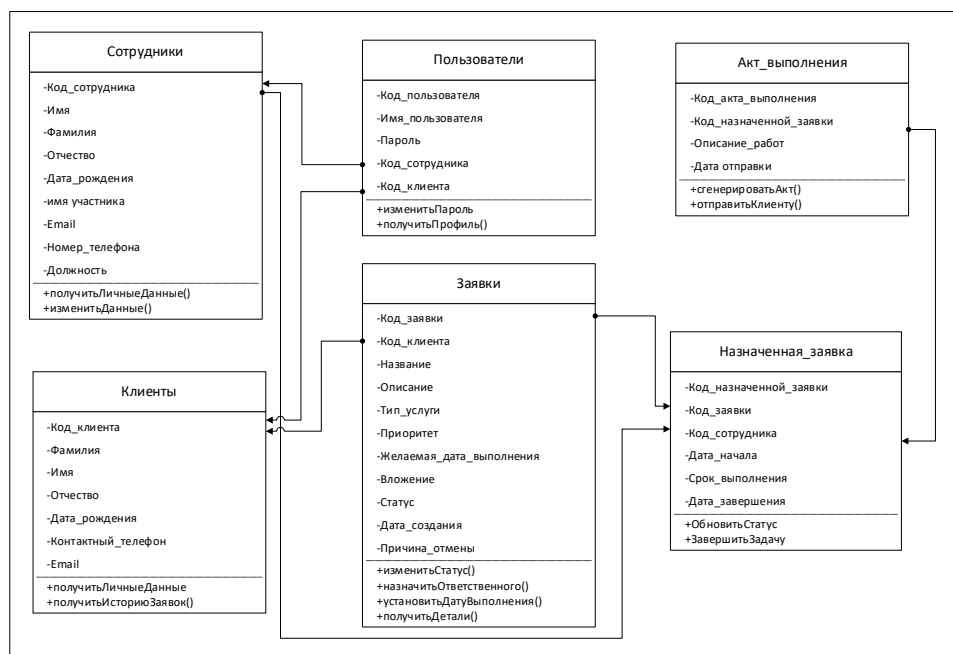


Рисунок 8 Диаграмма классов

4.3 Проектирование и разработка интерфейса

Проектирование и разработка интерфейса – это процесс создания пользовательского взаимодействия с системой, направленный на обеспечение удобства, понятности и эффективности работы пользователей. Он включает определение целей и функций приложения, анализ потребностей конечных пользователей, а также разработку структуры и визуального оформления интерфейса.

При проектировании и разработке интерфейса важно учитывать следующие принципы:

1) простота и интуитивность – интерфейс должен быть понятен без необходимости длительного обучения;

2) единообразие элементов – использование одинаковых стилей, кнопок и форм позволяет пользователю быстро освоить логику работы системы;

3) минимизация действий – для выполнения любой операции должно требоваться минимальное количество шагов;

4) обратная связь – система должна своевременно информировать пользователя о результатах действий, ошибках и изменениях состояния;

5) адаптивность – интерфейс должен корректно отображаться на различных устройствах и экранах.

Пользовательский интерфейс информационной системы также должен соответствовать следующим требованиям:

1) функциональность – наличие всех необходимых элементов для выполнения задач пользователя;

2) эргономичность – удобное расположение элементов управления, понятная структура меню и навигации;

3) информативность – четкое и лаконичное отображение данных, необходимых для принятия решений;

4) эстетичность – использование приятных цветовых схем, современных шрифтов и визуального баланса;

5) безопасность – предотвращение ошибок ввода, защита пользовательских данных и предотвращение несанкционированного доступа;

6) гибкость и масштабируемость – возможность расширения функционала без кардинальных изменений интерфейса.

На рисунке 9 представлена карта автоматизированной информационной системы.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		31

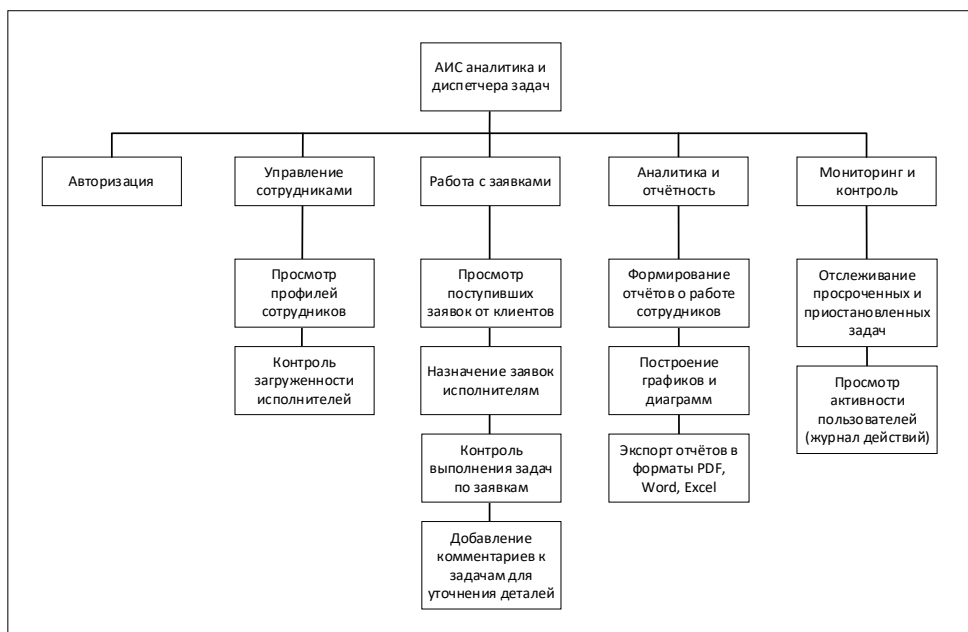


Рисунок 9 Карта АИС аналитика и диспетчера задач

На карте визуально отражено, какие модули и функции будут включены в АИС аналитика и диспетчера задач.

Главная страница разработана с целью обеспечить пользователю быстрый и удобный доступ к ключевым функциям и информации о компании. Дизайн построен на принципах простоты и интуитивности, чтобы сотрудники и клиенты могли легко ориентироваться в системе без предварительного обучения.

Эскиз интерфейса главной страницы показан на рисунке 10.



Рисунок 10 Эскиз интерфейса главной страницы

Целевые пользователи системы включают три основные категории: аналитик и диспетчер задач, сотрудник-исполнитель и клиент. Каждая из этих групп имеет свои уникальные потребности и доступ к различным функциональным возможностям системы.

Аналитик и диспетчер задач, как основной управляющий системой, обладает полным доступом ко всем функциям и расширенными административными правами. Это включает централизованное управление задачами, оперативное назначение исполнителям, постоянный контроль за их выполнением в режиме реального времени, а также автоматическую генерацию детализированных отчетов о продуктивности и загрузке. Данный уровень доступа необходим для обеспечения полного контроля над рабочими процессами, эффективного распределения ресурсов и стратегического планирования деятельности отдела.

Сотрудник-исполнитель имеет ограниченный доступ, сконцентрированный исключительно на выполнении поставленных задач и оперативной деятельности. Он может просматривать персональный список назначенных задач, детализацию по каждой из них, а также оперативно изменять их статус. Кроме того, исполнитель может прикреплять к задачам результаты работы и оставлять комментарии для диспетчера. Это позволяет поддерживать полную прозрачность процесса и информировать о ходе работ.

Клиент обладает наиболее ограниченным доступом, ориентированным исключительно на отслеживание прогресса по своим запросам и взаимодействию. Он может создавать новые заявки, которые автоматически преобразуются в задачи для исполнителей, просматривать статус выполнения своих задач и получать уведомления о их завершении. Данный функционал обеспечивает клиенту прозрачность и контроль над исполнением его требований без прямого контакта с исполнителями.

Для каждой категории пользователей система предусматривает индивидуальное отображение функционала, соответствующее его роли и уровню ответственности.

Интерфейс аналитика и диспетчера задач включает в себя основные элементы управления заявками, интуитивную панель распределения задач по сотрудникам, комплексный модуль аналитики и отчетности, а также доступ к профилям сотрудников. На главном экране отображаются активные заявки, их приоритет, статус выполнения и ответственные исполнители. Аналитик может принимать новые заявки, назначать ответственных сотрудников, а также контролировать выполнение задач в режиме реального времени.

Кроме того, в интерфейсе предусмотрены инструменты для анализа эффективности работы сотрудников и мониторинга ключевых показателей. Раздел с профилями сотрудников содержит сведения об их профессиональных компетенциях, контактных данных и текущих задачах, что позволяет точно распределять нагрузку и повышать продуктивность отдела.

Эскиз интерфейса для аналитика и диспетчера задач представлен на рисунке 11 и иллюстрирует организацию элементов управления.



Рисунок 11 Эскиз интерфейса аналитика и диспетчера задач

Интерфейс сотрудника предназначен для удобного выполнения назначенных задач. На главной странице отображается список активных задач с указанием приоритета, сроков выполнения и текущего статуса. Пользователь может открывать карточку каждой задачи, где представлено подробное описание, прикрепленные файлы, а также возможность изменять статус по мере выполнения. Дополнительным элементом интерфейса является профиль сотрудника, который содержит основные персональные данные, контактную информацию и сведения о должности и отделе. Сотрудник может редактировать часть информации в профиле, что обеспечивает актуальность данных и упрощает взаимодействие внутри команды. Эскиз интерфейса сотрудника представлен на рисунке 12.

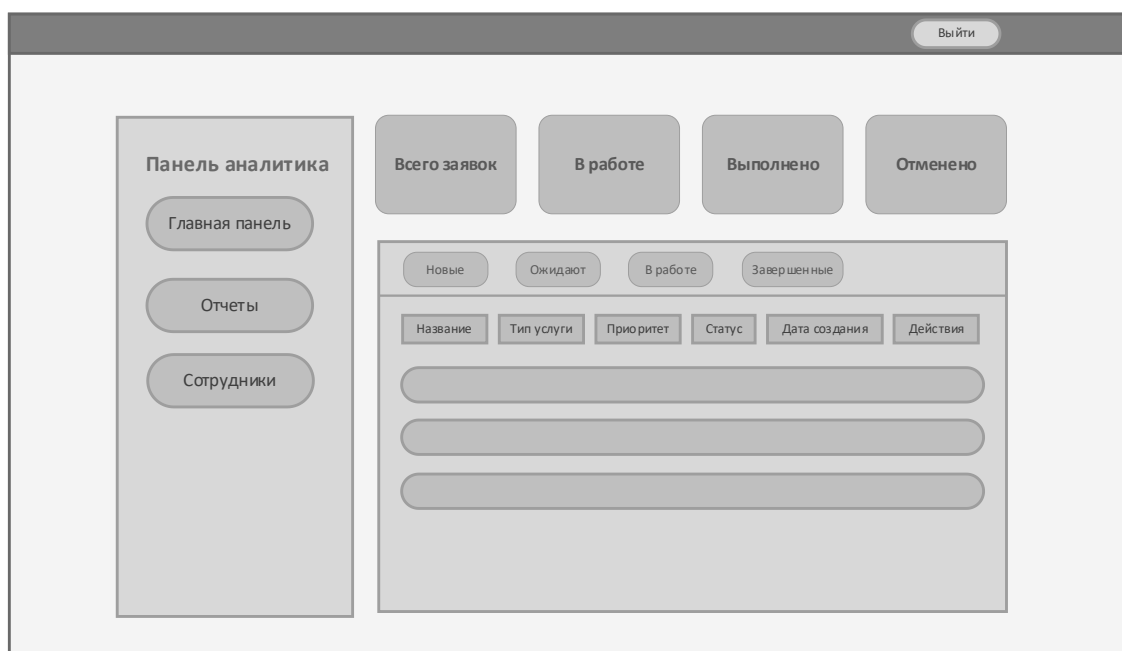


Рисунок 12 Эскиз интерфейса сотрудника

На главной странице пользователь может создавать новые заявки, заполняя форму с описанием необходимой работы, указанием сроков и дополнительными комментариями. Все поданные заявки автоматически поступают в систему и становятся доступными для обработки аналитиком и диспетчером задач. Клиент также имеет возможность отслеживать статус своих заявок в режиме реального времени.

Также в интерфейсе предусмотрен раздел для редактирования профиля, где клиент может изменять свои персональные данные и контактную информацию. Это обеспечивает актуальность сведений и упрощает взаимодействие пользователя с системой.

Эскиз интерфейса сотрудника представлен на рисунке 13.



Рисунок 13 Эскиз интерфейса клиента

4.4 Разработка кода информационной системы

Разработка кода информационной системы включает проектирование логики работы, реализацию функциональных модулей и интеграцию с базой данных, что позволяет обеспечить стабильную и эффективную работу всей системы. Качественно написанный код гарантирует корректное выполнение задач, удобство обслуживания и возможность дальнейшего расширения функционала в соответствии с требованиями пользователей и развитием бизнес-процессов.

На рисунке 14 представлен эскиз формы создания заявки, демонстрирующий основные поля для ввода информации о задаче, сроках выполнения, приоритете и дополнительных комментариях. Форма разработана для удобства пользователя, обеспечивая простоту заполнения и корректность передаваемых данных в систему.

Рисунок 14 Эскиз формы создания заявки

Для обеспечения корректной связи между учетными заявками и другими элементами системы было решено хранить данные заявок в отдельных таблицах базы данных. Для этого в системе создана модель *Ticket*, включающая основные сведения о заявке, её статус, приоритет и тип услуги, а также информацию о клиенте и сроках выполнения.

Каждая заявка может быть связана с одной или несколькими задачами через модель *Task*, где указывается исполнитель, описание задачи, сроки начала и дедлайны, а также статус выполнения. Код моделей представлен на рисунках 15–16. Такой подход позволяет централизованно управлять данными заявок и задач, обеспечивать их целостность, прозрачность выполнения, удобство фильтрации и формирования отчетности, а также гибко расширять функционал системы в будущем.

```

models.py

class Task(models.Model):
    ticket = models.ForeignKey(Ticket, on_delete=models.CASCADE,
related_name="tasks")
    employee = models.ForeignKey(Employee, on_delete=models.CASCADE,
related_name="tasks")
    description = models.TextField("Описание")
    created_at = models.DateTimeField("Дата создания", auto_now_add=True)
    start_time = models.DateTimeField("Начало задачи", blank=True, null=True) #
    новое поле
    deadline = models.DateTimeField("Дедлайн", blank=True, null=True)
    completed = models.BooleanField("Выполнено", default=False)
    accepted = models.BooleanField("Принята", default=False)
    canceled = models.BooleanField("Отменена", default=False) # ← новое поле

    def __str__(self):
        return f"Задача для {self.employee.user.username} по заявке {self.ticket.title}"

```

Рисунок 15 Код модели Task

```
models.py

class Ticket(models.Model):
    SERVICE_CHOICES = [
        ("software", "Разработка или настройка ПО"),
        ("web_dev", "Создание или поддержка веб-сайта"),
        ("mobile_app", "Разработка мобильного приложения"),
        ("integration", "Интеграция систем и API"),
        ("automation", "Автоматизация бизнес-процессов"),

        ("support", "Техническая поддержка"),
        ("maintenance", "Сопровождение и обновление систем"),
        ("bug_fix", "Исправление ошибок и доработки"),

        ("network", "Сетевые решения и безопасность"),
        ("server_admin", "Администрирование серверов"),
        ("backup", "Резервное копирование и восстановление данных"),

        ("training", "Обучение и документация"),
        ("other", "Другое (указать в описании)"),
    ]

    PRIORITY_CHOICES = [
        ("low", "Низкий"),
        ("medium", "Средний"),
        ("high", "Высокий"),
        ("critical", "Критический"),
    ]

    STATUS_CHOICES = [
        ("new", "Новая"),
        ("pending", "Ожидание подтверждения"),
        ("in_progress", "В работе"),
        ("done", "Завершена"),
        ("canceled", "Отменена"),
    ]

    client = models.ForeignKey(Client, on_delete=models.CASCADE,
related_name="tickets")
    title = models.CharField("Название заявки", max_length=200)
    description = models.TextField("Подробное описание задачи")
    service_type = models.CharField("Тип услуги", max_length=50,
choices=SERVICE_CHOICES, blank=True, null=True)
    priority = models.CharField("Приоритет", max_length=20,
choices=PRIORITY_CHOICES, default="medium")
    desired_date = models.DateField("Желаемая дата выполнения", blank=True,
null=True)
    attachment = models.FileField("Вложение (если нужно)", upload_to="tickets/",
blank=True, null=True)
    status = models.CharField("Статус", choices=STATUS_CHOICES, default="new",
max_length=20)
    created_at = models.DateTimeField("Дата создания", auto_now_add=True)
    cancel_reason = models.TextField(blank=True, null=True)

    def __str__(self):
        return f"{self.title} ({self.get_status_display()}"
```

Рисунок 16 Код модели Ticket

В системе реализован функционал, позволяющий уведомлять клиента о причинах отклонения заявки, а также автоматически отправлять акт о выполнении работ по завершении задачи. На рисунке 17 представлен фрагмент кода функции *change_status*, отвечающей за обработку изменения статуса заявки.

Если заявка отклоняется, система формирует и отправляет клиенту письмо с указанием причины отказа, указанной сотрудником. В случае успешного выполнения задачи, система автоматически создает акт о выполнении работ в формате docx, содержащий сведения о клиенте, исполнителе, типе услуги и дате завершения. Сформированный акт прикрепляется к электронному письму и направляется на email клиента для подтверждения выполнения заявки.

```

views.py

def change_status(request, ticket_id):
    ticket = get_object_or_404(Ticket, id=ticket_id)

    task = ticket.tasks.first()
    if not task:
        messages.error(request, "Задача для этой заявки не найдена.")
        return redirect('employee_tasks')

    if request.method == "POST":
        action = request.POST.get("action")
        if action == "complete":
            ticket.status = "done"
            ticket.save()
            doc = Document()

            style = doc.styles['Normal']
            style.font.name = 'Times New Roman'
            style.font.size = Pt(12)

            title = doc.add_heading('АКТ ВЫПОЛНЕННЫХ РАБОТ', 0)
            title.alignment = WD_ALIGN_PARAGRAPH.CENTER

            doc.add_paragraph(f"№ {ticket.id}-АБП от {timezone.now().strftime('%d.%m.%Y')}")
            last_paragraph = doc.paragraphs[-1]
            last_paragraph.alignment = WD_ALIGN_PARAGRAPH.CENTER
            last_paragraph.style.font.bold = True

            doc.add_paragraph()

```

Рисунок 17 Код для обработки изменения статуса заявок

На рисунке 18 представлена эскиз круговой диаграммы распределения задач между сотрудниками, которая наглядно демонстрирует загрузку каждого специалиста. Диаграмма отображает соотношение выполненных и активных задач в разрезе отдельных сотрудников. Визуальное представление включает цветовую кодировку: зеленым цветом обозначены завершённые задачи, оранжевым – находящиеся в работе. Такой подход позволяет быстро оценить эффективность работы каждого сотрудника, выявить потенциальные проблемы с распределением нагрузки и определить специалистов, требующих дополнительной поддержки или перераспределения задач.



Рисунок 18 Эскиз диаграммы распределения задач между сотрудниками

На рисунке 19 представлены фрагменты кода системы аналитической отчетности. Показана функция *tickets_detailed_report*, реализующая фильтрацию заявок по статусу, приоритету, типу услуги и датам. Также представлена функция *employee_detailed_report*, обеспечивающая фильтрацию сотрудников по ФИО, отделу и временному периоду с агрегацией данных о задачах.

Обе функции формируют детализированные отчеты для анализа эффективности работы компании. Отчет по заявкам содержит информацию о клиентах, исполнителях и сроках выполнения работ. Отчет по сотрудникам включает статистику выполнения задач, что позволяет оценить продуктивность каждого специалиста.

```
views.py

@login_required
def tickets_detailed_report(request):
    """Детальный отчет по заявкам"""
    status_filter = request.GET.get('status')
    priority_filter = request.GET.get('priority')
    service_type_filter = request.GET.get('service_type')
    date_from = request.GET.get('date_from')
    date_to = request.GET.get('date_to')

    tickets =
Ticket.objects.all().select_related('client').prefetch_related('tasks')

    if status_filter:
        tickets = tickets.filter(status=status_filter)
    if priority_filter:
        tickets = tickets.filter(priority=priority_filter)
    if service_type_filter:
        tickets = tickets.filter(service_type=service_type_filter)

# ... дальнейший код tickets_detailed_report

@login_required
def employee_detailed_report(request):
    """Детальный отчет по сотрудникам"""
    employee_id = request.GET.get('employee')
    date_from = request.GET.get('date_from')
    date_to = request.GET.get('date_to')
    department = request.GET.get('department')

    employees = Employee.objects.all()

    if employee_id:
        employees = employees.filter(id=employee_id)
    if department:
        employees = employees.filter(department=department)
    detailed_report = []
    for emp in employees:
        tasks = Task.objects.filter(employee=emp)
        if date_from:
            tasks = tasks.filter(created_at__date__gte=date_from)
        if date_to:
            tasks = tasks.filter(created_at__date__lte=date_to)

# ... дальнейший код employee_detailed_report
```

Рисунок 19 Реализация функций аналитических отчетов

На рисунке 20 представлена реализация функций экспорта аналитических отчетов в формате Microsoft Excel. Система предоставляет возможность загрузки как отчетов по сотрудникам, так и отчетов по заявкам в структурированном виде, готовом для дальнейшего анализа. Оба экспортируемых отчета автоматически применяют профессиональное форматирование: заголовки таблиц выделяются жирным шрифтом с синим фоном, обеспечивается

выравнивание данных и автоматическая подгонка ширины столбцов под содержимое. Это позволяет руководителям непосредственно использовать полученные файлы в корпоративной отчетности и презентациях без необходимости дополнительного редактирования.

```

views.py

@login_required
def export_employee_detailed_report(request):
    """Экспорт детального отчета по сотрудникам в Excel"""
    employee_id = request.GET.get('employee')
    date_from = request.GET.get('date_from')
    date_to = request.GET.get('date_to')
    department = request.GET.get('department')

    employees = Employee.objects.all()
    if employee_id:
        employees = employees.filter(id=employee_id)
    if department:
        employees = employees.filter(department=department)

    wb = Workbook()
    ws = wb.active
    ws.title = "Отчет по сотрудникам"

    # ... дальнейший код export_employee_detailed_report

@login_required
def export_tickets_detailed_report(request):
    """Экспорт детального отчета по заявкам в Excel"""
    status_filter = request.GET.get('status')
    priority_filter = request.GET.get('priority')
    service_type_filter = request.GET.get('service_type')
    date_from = request.GET.get('date_from')
    date_to = request.GET.get('date_to')

    tickets =
Ticket.objects.all().select_related('client').prefetch_related('tasks')

    if status_filter:
        tickets = tickets.filter(status=status_filter)
    if priority_filter:
        tickets = tickets.filter(priority=priority_filter)
    if service_type_filter:
        tickets = tickets.filter(service_type=service_type_filter)
    if date_from:
        tickets = tickets.filter(created_at__date__gte=date_from)
    if date_to:
        tickets = tickets.filter(created_at__date__lte=date_to)

    wb = Workbook()
    ws = wb.active
    ws.title = "Отчет по заявкам"

    # ... дальнейший код export_tickets_detailed_report

```

Рисунок 20 Реализация функций экспорта аналитических отчетов

5 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Информационная система аналитика и диспетчера задач в ООО «Будущие Информационные Технологии» разработана для упрощения и оптимизации работы сотрудников компании. Она обеспечивает полную автоматизацию процессов управления заявками, распределения задач между исполнителями, контроля их выполнения и формирования аналитических отчетов. Система объединяет все ключевые функции отдела в едином веб-интерфейсе, что позволяет диспетчеру задач и сотрудникам эффективно взаимодействовать между собой и с заказчиками.

Основная цель системы – повысить прозрачность и оперативность работы подразделений, сократить время на обработку заявок и улучшить контроль над качеством выполнения задач. Пользователи могут добавлять, редактировать, удалять, искать и фильтровать заявки, управлять сотрудниками и их нагрузкой, а также анализировать эффективность работы с помощью встроенных инструментов аналитики. Функция генерации отчетов позволяет формировать сводные данные по задачам, сотрудникам или заявкам за выбранный период и экспортировать их в формат Excel.

Работа с системой не требует специальных технических знаний. Для доступа к автоматизированной информационной системе аналитика и диспетчера задач в ООО «Будущие Информационные Технологии» достаточно иметь компьютер, ноутбук или мобильное устройство с подключением к интернету и установленным современным веб-браузером.

Для запуска и начала работы с системой необходимо выполнить следующие шаги:

1) открыть веб-браузер (google chrome, mozilla firefox, opera, microsoft edge или safari) на компьютере или мобильном устройстве;

2) в адресной строке браузера ввести адрес системы – <http://bit-tasks.ru> (или указанный корпоративный адрес);

3) на странице авторизации ввести предоставленные администратором логин и пароль;

4) после успешного входа система откроет главную страницу, где пользователь сможет работать с заявками, задачами и отчетами в соответствии со своей ролью;

5) при необходимости пользователь может перейти в раздел аналитики для формирования отчетов или в панель управления для распределения и контроля задач.

					ЛНТ О. 09.02.07 03 02. ПЗ							
Изм	Лист	№ докум.	Подп.	Дата	Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики			Лит.	Лист	Листов		
Разраб.	Бочкарева С.С.									41	54	
Пров.	Переверзова К.С.							ИС-1-22				
Н. контр.	Переверзова К.С.											
Утв.	Переверзова К.С.											

При запуске информационной системы открывается стартовая страница, представленная на рисунке 21, которая содержит информацию о компании и кнопку для входа в систему.

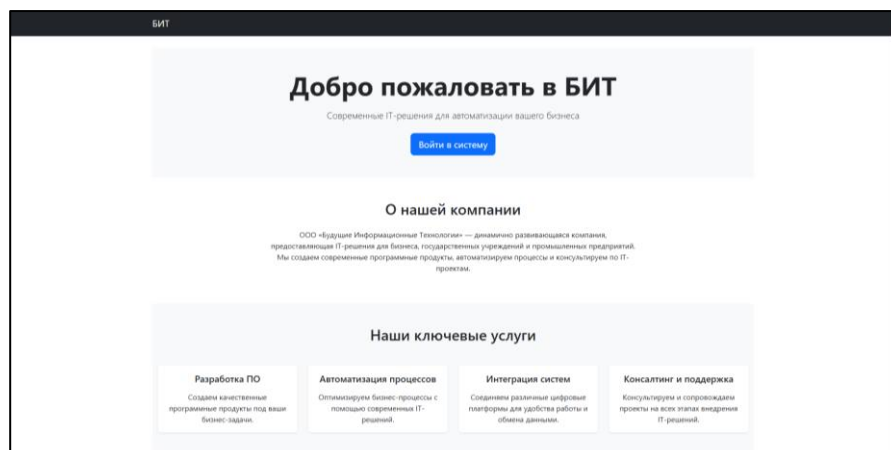


Рисунок 21 Стартовая страница

После перехода со стартовой страницы пользователь попадает на страницу аутентификации, представленную на рисунке 22, которая содержит форму входа в систему с полями для ввода логина и пароля, а также кнопку авторизации.

Рисунок 22 Страница аутентификации

Клиент может зарегистрироваться в системе для возможности оформления заявок на услуги. Для этого необходимо нажать на кнопку «Регистрация» на стартовой странице, после чего откроется интерфейс регистрации, представленный на рисунке 23.

Рисунок 23 Форма регистрации клиента

Войдя в систему под учетной записью клиента, пользователь попадает на страницу «Мои заявки», где отображается список всех его ранее созданных заявок с указанием текущего статуса выполнения, что представлено на рисунке 24.

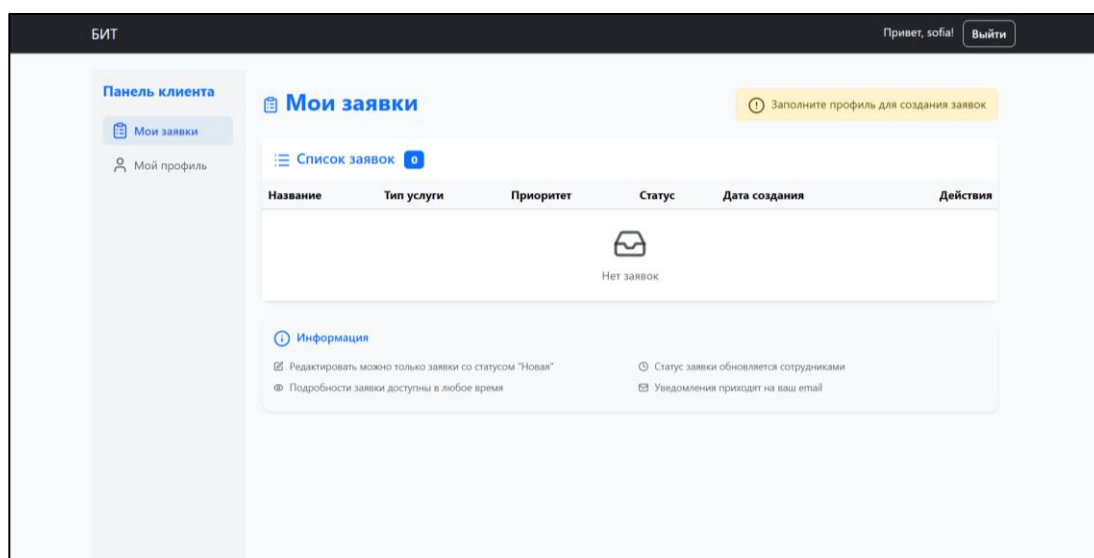


Рисунок 24 Панель управления клиента с списком заявок

Для возможности оформления заявки клиенту необходимо предварительно заполнить свой профиль на странице «Мой профиль», представленной на рисунке 25.

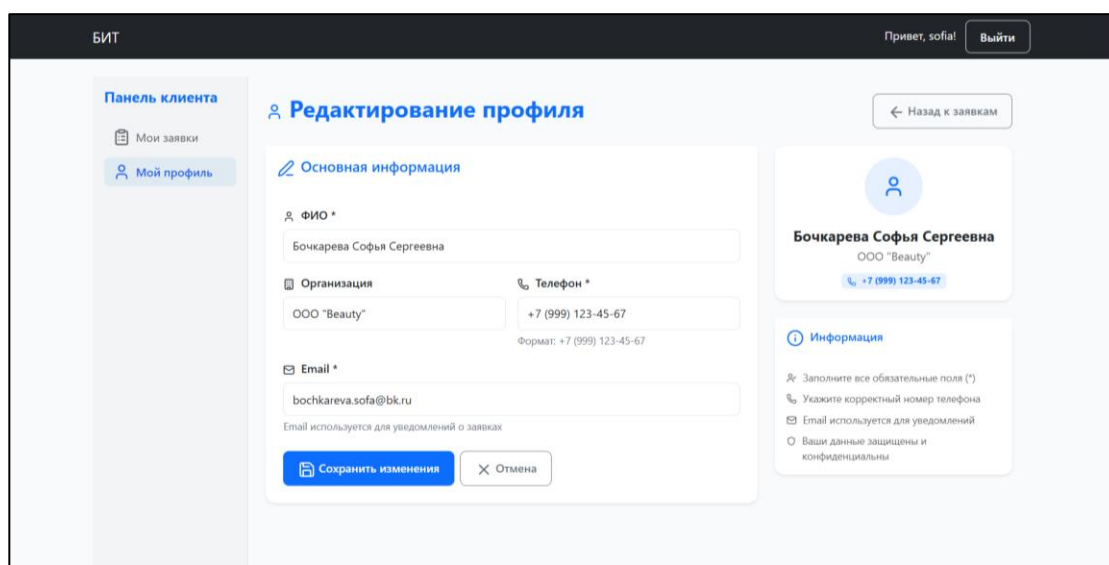


Рисунок 25 Страница редактирования профиля клиента

На данной странице пользователь должен ввести обязательные персональные данные: полное ФИО, контактный телефон, адрес электронной почты, а также при необходимости указать дополнительную информацию о своей организации. Только после сохранения корректно заполненного профиля система предоставляет доступ к функции создания новых заявок на услуги компании.

После заполнения профиля клиент получает доступ к странице создания новой заявки, представленной на рисунке 26.

Рисунок 26 Форма создания новой заявки

После заполнения всех необходимых полей и нажатия кнопки «Отправить заявку» система автоматически регистрирует обращение и присваивает ему статус «Новая». До момента назначения заявки аналитиком и диспетчером задач клиент имеет возможность редактировать созданную заявку. После принятия заявки в работу возможность редактирования становится недоступной, и все дальнейшие изменения осуществляются через систему комментариев и согласований с назначенным исполнителем.

Войдя в систему под учетной записью аналитика и диспетчера задач, пользователь попадает на панель управления аналитика, представленную на рисунке 27.

Рисунок 27 Панель управления аналитика

На данной панели отображается комплексная статистика по всем заявкам системы. Интерфейс содержит вкладки с группировкой задач по различным статусам, что позволяет быстро ориентироваться в текущей рабочей нагрузке. Аналитик и диспетчер задач имеет возможность просматривать детальную информацию по каждой заявке. При нажатии кнопки «Назначить» в

интерфейсе открывается диалоговое окно назначения исполнителя, представленное на рисунке 28.

Рисунок 28 Окно назначения исполнителя и установки дедлайна

После выбора сотрудника и установки сроков выполнения аналитик может подтвердить назначение. Система автоматически уведомляет назначенного сотрудника о новой задаче и устанавливает статус заявки «Ожидание подтверждения».

В системе реализован механизм уведомления клиентов при отклонении заявок аналитиком и диспетчером задач. При принятии решения об отказе в выполнении заявки по тем или иным причинам, система автоматически отправляет клиенту электронное письмо с информацией о принятом решении и указанием причины отказа. Как показано на рисунке 29, при нажатии кнопки «Отклонить» в интерфейсе аналитика открывается модальное окно с формой для указания причины отказа.

Рисунок 29 Интерфейс отклонения заявки с указанием причины

После указания причины отказа и подтверждения система изменяет статус заявки на «Отклонена», сохраняет причину отказа, направляет клиенту письмо с указанной причиной. Пример уведомления показан на рисунке 30.

Рисунок 30 Интерфейс отклонения заявки с указанием причины

В разделе «Сотрудники» панели аналитика отображается список всех сотрудников, представленный на рисунке 31.

ФИО	Должность	Отдел	Активные задачи	Действия
Виктор Романов	Специалист по резервному копированию	Отдел инфраструктуры и администрирования	Нет	Профиль
Василий Васильев	Ведущий разработчик ПО	Отдел разработки	1	Профиль
Сергей Егоров	Инженер по интеграции систем и API	Отдел разработки	1	Профиль
Павел Григорьев	Тестировщик (QA-инженер)	Отдел контроля качества	Нет	Профиль
Елена Федорова	Специалист по сетевой безопасности	Отдел инфраструктуры и администрирования	Нет	Профиль
Игорь Волков	Инженер по автоматизации процессов	Отдел разработки	Нет	Профиль
Наталья Орлова	Системный администратор	Отдел инфраструктуры и администрирования	Нет	Профиль
Дмитрий Кузнецов	Специалист технической поддержки	Отдел клиентской поддержки	Нет	Профиль

Рисунок 31 Страница управления сотрудниками

На данной странице аналитик может:

- 1) просматривать список всех сотрудников с указанием должности и отдела;
- 2) видеть текущую загрузку каждого исполнителя (количество активных задач);
- 3) оценивать процент выполнения задач за выбранный период.

На рисунке 32 представлена детальная карточка сотрудника, которая открывается при нажатии кнопки «Подробнее» в разделе управления персоналом. Карточка содержит комплексную информацию об исполнителе:

- 1) полные ФИО и контактные данные;
- 2) информацию о должности и отделе;
- 3) статистику выполнения задач (всего/выполнено/в работе);
- 4) историю назначенных заявок;
- 5) текущую рабочую нагрузку и доступность для новых заданий.

Профиль сотрудника	
← Назад к списку	
Основная информация	
ФИО: Васильев Василий Васильевич	Email: vasilyu@mail.ru
Должность: Ведущий разработчик ПО	Имя пользователя: vasilyev.employee
Отдел: Отдел разработки	Дата регистрации: 02.10.2025
	Последний вход: 17.10.2025 11:31
Статистика задач	
Всего задач: 2	В работе: 1
Выполнено: 1	Просрочено: 0
Активные задачи 1	
Разработка интернет-магазина косметики	
Задача по заявке: Разработка интернет-магазина косметики	
Высокий 10.11.2025	
Нагрузка	
Текущая нагрузка: 0	
Коэффициент занятости: 0%	
Действия	
Написать email	

Рисунок 32 Детальная карточка сотрудника

Данный функционал позволяет аналитику оптимально распределять задачи между исполнителями с учетом их специализации и текущей занятости.

В разделе «Отчеты» системы реализован комплексный аналитический модуль, обеспечивающий глубокую визуализацию и всесторонний анализ данных о работе компании, как показано на рисунке 33. Система генерирует интерактивные круговые диаграммы для каждого сотрудника, наглядно отображающие распределение задач по статусам выполнения. Цветовая кодировка позволяет мгновенно оценить эффективность работы и текущую загрузку исполнителей.

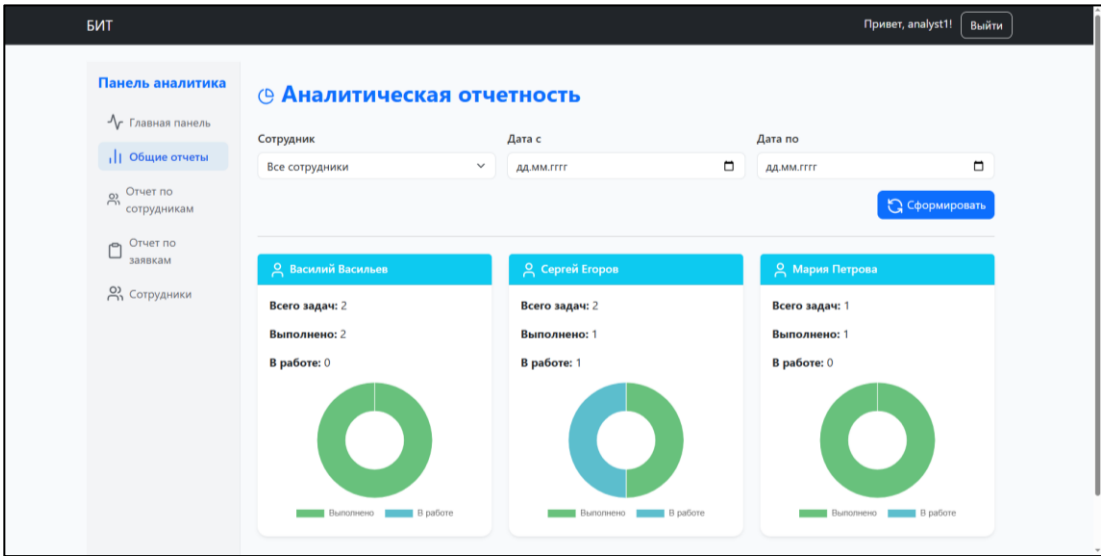


Рисунок 33 Интерфейс аналитических отчетов

Детализированная аналитика по сотрудникам, представленная на рисунке 34, обеспечивает полную статистику выполнения задач за выбранный временной период. Система автоматически также анализирует распределение рабочей нагрузки по структурным подразделениям и должностным позициям.

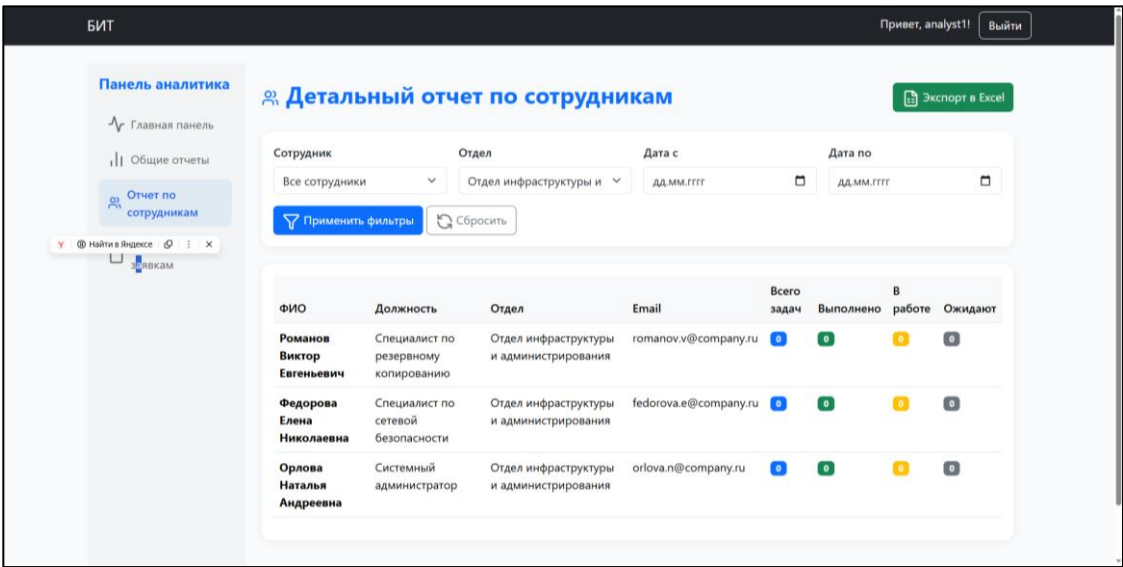


Рисунок 34 Детальный отчет по сотрудникам

Комплексный анализ заявок, показанный на рисунке 35, предоставляет возможности многоуровневой фильтрации данных по статусам обработки, приоритетам и категориям услуг. В отчете представлена детальная статистика обработки клиентских обращений за различные периоды, проводится анализ клиентской базы и оценивается востребованность услуг компании, что позволяет принимать обоснованные управленческие решения.

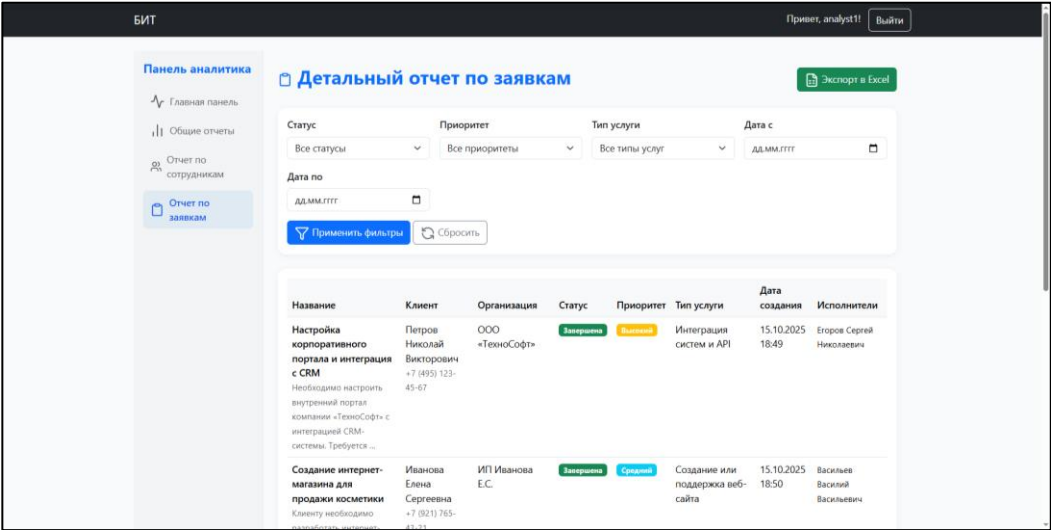


Рисунок 35 Детальный отчет по заявкам

Все виды отчетов поддерживают функцию экспорта в формат Microsoft Excel с автоматическим применением профессионального форматирования.

Войдя в систему под учетной записью сотрудника, пользователь попадает на персональную панель исполнителя, представленную на рисунке 36.

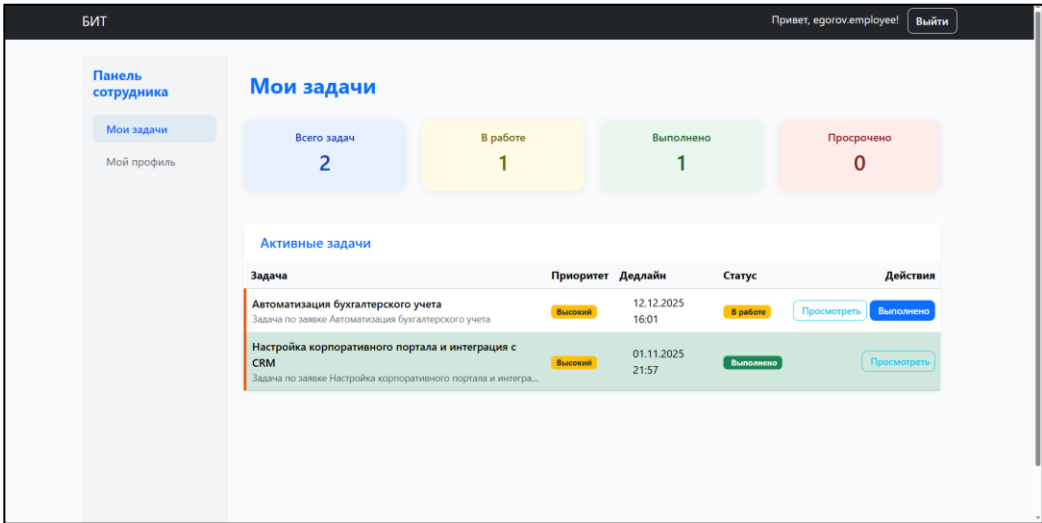


Рисунок 36 Панель управления сотрудника

На данной панели отображаются все задачи, назначенные конкретному сотруднику, с группировкой по статусам выполнения. Интерфейс включает следующие ключевые элементы:

- 1) список текущих задач с указанием сроков выполнения и приоритета;
- 2) статистику по выполненным и просроченным заданиям;

3) быстрый доступ к принятию новых задач и отправке отчетов;

4) информацию о текущей рабочей нагрузке и ближайших дедлайнах.

При нажатии на кнопку «Подробнее» сотрудник получает доступ к детальной информации о заявке, как показано на рисунке 37.

The screenshot shows a web interface for task management. The title is 'Автоматизация бухгалтерского учета'. It contains several sections: 'Информация о заявке' (Task Information) with a description of the task, a 'Желаемая дата' (Desired date), a 'Дата создания' (Creation date), and a 'Тип услуги' (Service type); 'Информация о клиенте' (Client Information) with fields for 'ФИО' (Full name), 'Email', 'Организация' (Organization), and 'Телефон' (Phone); 'Информация о задаче' (Task Information) with 'Дедлайн' (Deadline) and 'Статус задачи' (Task status); and a section for uploading a completion act with a file selection button and a 'Завершить задачу и отправить акт' (Complete task and send act) button.

Рисунок 37 Детальная информация о заявке

При выполнении заявки система автоматически отправляет клиенту уведомление на электронную почту с актом о выполненной работе, как показано на рисунке 38.

The screenshot shows an email notification. The subject is 'Акт выполненных работ по заявке №1077'. The sender is 'bochkareva.sofa@bk.ru'. The email contains a link to a document 'Акт_ВР_1077_19102025.docx'. Below the link is a preview of the document, which is a form titled 'АКТ ВЫПОЛНЕННЫХ РАБОТ'. The email also contains a message from 'Уважаемый(ая) Бочкарева Софья Сергеевна!' (Dear Ms. Sofya Sergeevna Bockareva!) stating that the work on the task 'Разработка интернет-магазина косметики' (Development of an internet cosmetics store) is completed. It lists details of the task: 'Номер заявки: 1077' (Task number: 1077), 'Наименование работы: Разработка интернет-магазина косметики' (Task name: Development of an internet cosmetics store), 'Дата выполнения: 19.10.2025 13:16' (Completion date: 19.10.2025 13:16), and 'Исполнитель: Василий Васильев' (Executor: Vasily Vasilyev). The email ends with a signature from 'ООО "Бугульма - Будущие Информационные Технологии"' (OOO 'Bugulma - Future Information Technologies').

Рисунок 38 Уведомление клиенту о выполнении заявки

					ЛНТ О. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		49

Процесс включает следующие этапы:

- 1) сотрудник отмечает заявку как выполненную в системе;
- 2) система автоматически генерирует акт выполненных работ в формате docx;
- 3) на электронную почту клиента отправляется письмо;
- 4) в теле письма содержится информация о выполненной работе и акт во вложении;
- 5) клиент получает полный отчет о выполненных работах с возможностью скачивания акта.

Акт содержит подробное описание выполненных работ, сроки выполнения, информацию об исполнителе и реквизиты сторон для подписания.

Сотрудник имеет доступ к редактированию персональной информации через раздел «Мой профиль», представленный на рисунке 39. Важно отметить, что служебная информация (должность, отдел) изменяется исключительно администратором системы. Это обеспечивает корректность организационной структуры и соблюдение политики компании. Все внесенные сотрудником изменения проходят автоматическую проверку и мгновенно отражаются в системе.

БИТ Привет, egorov.employee! Выйти

Редактирование профиля

Основная информация

Фамилия * Егоров Имя * Сергей Отчество * Николаевич

Email * egorov.s@company.ru
Email используется для уведомлений о задачах

Должность Инженер по интеграции систем и API Отдел Отдел разработки
Для изменения должности обратитесь к администратору Для изменения отдела обратитесь к администратору

Сохранить изменения Назад к задачам

Сергей Егоров
Инженер по интеграции систем и API
Отдел разработки

Статистика

Текущая нагрузка: 0
Активные задачи: 1
Завершено: 1

Информация

Вы можете редактировать только свои личные данные
Должность и отдел изменяются администратором

Рисунок 39 Интерфейс редактирования профиля сотрудника

Выход из системы осуществляется через нажатие кнопки «Выйти» в правом верхнем углу интерфейса, как показано на рисунке 40.

БИТ Привет, egorov.employee! Выйти

Рисунок 40 Интерфейс выхода из системы

При нажатии на кнопку система мгновенно завершает текущую пользовательскую сессию и выполняет перенаправление на стартовую страницу. Все несохраненные данные автоматически сохраняются перед выходом. Для последующего доступа к системе потребуются повторная авторизация с вводом логина и пароля.

ЗАКЛЮЧЕНИЕ

В условиях активного развития цифровых технологий и увеличения объёмов обрабатываемой информации автоматизация рабочих процессов является необходимым условием эффективного функционирования современных организаций. Особенно это актуально для подразделений, занимающихся аналитикой и управлением задачами, где требуется точная координация действий, оперативное взаимодействие между сотрудниками и высокий уровень контроля над выполнением работ. Применение автоматизированных информационных систем позволяет оптимизировать рутинные процессы, снизить вероятность ошибок и обеспечить прозрачность выполнения задач, что напрямую повышает производительность и качество работы отдела.

В рамках курсового проекта была разработана автоматизированная информационная система аналитика и диспетчера задач для ООО «Будущие Информационные Технологии». Основной целью проекта являлась оптимизация процессов управления заявками и задачами, сокращение времени обработки клиентских запросов, повышение прозрачности выполнения задач и улучшение взаимодействия между сотрудниками отдела.

В ходе выполнения проекта были достигнуты следующие результаты:

1) проведён детальный анализ предметной области и организационной структуры отдела IT-аналитики, что позволило выявить узкие места в текущих процессах и определить направления для автоматизации;

2) сформулированы функциональные и технические требования к системе, учитывающие потребности пользователей и специфику работы подразделения, включая удобный интерфейс, возможности настройки уведомлений и интеграцию с внешними системами;

3) обоснован выбор технологий разработки – фреймворк Django и язык Python, обеспечивающие надёжность, масштабируемость и гибкость решения, а также возможность быстрого внедрения новых функций;

4) выполнено структурное и объектно-ориентированное проектирование системы с применением UML-диаграмм;

5) разработан и протестирован прототип системы, демонстрирующий ключевые функциональные возможности и пользовательский интерфейс, что позволило выявить потенциальные проблемы на раннем этапе и внести корректировки перед финальной реализацией;

					ЛНТ 0. 09.02.07 03 02. ПЗ		
Изм	Лист	№ докум.	Подп.	Дата			
Разраб.	Бочкарева С.С.				Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики	Лит.	Лист
Пров.	Переверзова К.С.						Листов
							51
							54
Н. контр.	Переверзова К.С.				ИС-1-22		
Утв.	Переверзова К.С.						

6) реализованы основные модули для управления заявками, задачами, сотрудниками и аналитическими отчётами, каждый из которых прошёл тестирование на предмет функциональности и удобства использования;

7) создан интуитивно понятный пользовательский интерфейс, повышающий комфорт работы и сокращающий время выполнения рутинных операций;

8) проведено комплексное тестирование системы, подтвердившее её корректность, соответствие требованиям и готовность к внедрению в реальную эксплуатацию.

Структурно-функциональное проектирование включало следующие элементы:

1) диаграммы потоков данных (DFD) для визуализации последовательности передачи информации между объектами;

2) модель IDEFO с декомпозицией процессов для детального отображения структуры и функциональных блоков;

3) ERD-диаграмму для построения структуры базы данных и определения взаимосвязей между сущностями.

Объектно-ориентированное проектирование включало:

1) диаграмму вариантов использования, показывающую основные сценарии взаимодействия пользователей с системой и реализуемые функции;

2) диаграмму последовательности, отображающую порядок обмена сообщениями между объектами в процессе выполнения операций;

3) диаграмму классов, определяющую структуру объектов и их связи.

Реализованная система обеспечивает централизованное управление задачами и заявками, позволяет отслеживать прогресс выполнения, формировать отчётность и анализировать эффективность работы. Она повышает прозрачность процессов, снижает риски потери данных и способствует принятию управленческих решений. Прототип системы позволил оптимизировать пользовательский опыт и убедиться в корректности логики взаимодействия между модулями до реализации проекта.

Таким образом, разработанная автоматизированная информационная система аналитика и диспетчера задач соответствует современным требованиям к цифровизации бизнес-процессов. Она сокращает трудозатраты на выполнение операций, улучшает коммуникацию между сотрудниками и клиентами, повышает производительность отдела и создаёт платформу для совершенствования процессов управления. Результаты проекта подтверждают важность внедрения подобных систем как этапа цифровой трансформации, обеспечивающего повышение конкурентоспособности и устойчивое развитие организации.

					ЛНТ 0. 09.02.07 03 02. ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		52

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) ГОСТ Р 59853-2021 Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения. – Введ. 01.01.22. – М., 2022. – 16 с. – (Стандартинформ)
- 2) Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений / Г. Буч. – М.: Вильямс, 2021. – 720 с.
- 3) Зараменских Е.П. Информационные системы: управление жизненным циклом / Е. П. Зараменских. – М.: Юрайт, 2025. – 486 с.
- 4) Лапин В.И., Кузнецов А.А. Информационные технологии в управлении предприятиями / В.И. Лапин, А.А. Кузнецов– М.: Горячая линия – Телеком, 2023. – 412 с.
- 5) Чистов Д. В. Проектирование информационных систем / Д. В. Чистов. – М.: Юрайт, 2025. – 273 с.
- 6) Диаграмма вариантов использования (UseCase diagram) // URL: <https://docs.ensi.tech/analyst-guides/tools/diagrams/uml/use-case-diagram> (дата обращения 17.09.2025)
- 7) Диаграмма последовательности – Рувикс: Интернет-энциклопедия // URL: https://ru.ruwiki.ru/wiki/Диаграмма_последовательности (дата обращения 19.09.2025)
- 8) Диаграмма классов (Class diagram) // URL: <https://ensi-platform.gitlab.io/analyst-guides/tools/diagrams/uml/class-diagram/> (дата обращения 19.09.2025)
- 9) Основы веб-технологий: HTML, CSS, JavaScript и HTTP // URL: <https://sky.pro/wiki/javascript/osnovy-veb-tehnologij-html-css-javascript-i-http/> (дата обращения 05.10.2025)
- 10) Описание нотации IDEFO // URL: <https://micro-solution.ru/bp-know/regulation/IDEFO> (дата обращения 24.09.2025)
- 11) Что такое моделирование данных? // URL: <https://dzen.ru/a/ZQU-tKIUpDMdnD96> (дата обращения 29.09.2025)
- 12) Настройка подключения к базе данных в Django // URL: <https://sky.pro/wiki/python/nastrojka-podklyucheniya-k-baze-dannyh-v-django/> (дата обращения 06.10.2025)

					ЛНТ 0. 09.02.07 03 02. ПЗ		
Изм	Лист	№ докум.	Подп.	Дата			
Разраб.	Бочкарева С.С.				Разработка системы работы аналитика и диспетчера задач в отделе IT-аналитики	Лит.	Лист
Пров.	Переверзова К.С.						Листов
							53
							54
Н. контр.	Переверзова К.С.				ИС-1-22		
Утв.	Переверзова К.С.						

- 13) *Административный сайт Django – Python // URL: <https://pythonclass.ru/django/administrativnyj-sajt-django/> (дата обращения 03.10.2025)*
- 14) *Django II Создание моделей // URL: <https://metanit.com/python/django/5.1.php> (дата обращения 12.10.2025)*
- 15) *Карпенцева Н.А., Павлова Е.Н. Методические указания по выполнению и оформлению дипломных и курсовых проектов для преподавателей и студентов очной, заочной и очно – заочной формы обучения для всех специальностей. – Лениногорск: ГАПОУ «Лениногорский нефтяной техникум», 2025 г–50 с.*
- 16) *Переверзова К.С. Методические указания по выполнению и оформлению курсового проекта по МДК 05.02 «Разработка кода информационных систем» для студентов специальности 09.02.07 «Информационные системы и программирование». – Лениногорск: Лениногорский нефтяной техникум, 2023. – 22 с.*

					<i>ЛНТ 0. 09.02.07 03 02. ПЗ</i>	<i>Лист</i>
						<i>54</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		