**Базовый план по качеству программного обеспечения**

* Введение

Это документ описывает разработку программного продукта для учета заявок на ремонт бытовой техники. Он предназначен для разработчиков, тестировщиков и других участников проекта, чтобы обеспечить единообразие подхода к разработке и интеграции системы.

* Цели разработки

Основная цель разработки данного программного продукта – автоматизация процесса учета заявок на ремонт бытовой техники, улучшение взаимодействия между клиентами и сервисными центрами, а также повышение эффективности работы сотрудников компании.

* Описание предметной области

Программный продукт будет использоваться для автоматизации процессов:

Регистрации новых заявок на ремонт от клиентов.

Назначения мастеров для выполнения ремонта.

Учет материалов и запчастей, необходимых для ремонта.

Контроля за выполнением работ и закрытием заявок.

Формирования отчетов о выполненных работах.

* Основные требования к системе
* Регистрация заявки:
* Клиент может подать заявку через веб-интерфейс или мобильное приложение.
* Заявка должна содержать информацию о клиенте, технике, характере неисправности и предпочтительном времени визита мастера.
* Назначение мастера:
* Система автоматически назначает мастера на основе доступности и специализации.
* Возможность ручного назначения мастером в случае необходимости.
* Учет материалов и запчастей:
* Ведется учет наличия запчастей и расходных материалов на складе.
* При назначении мастера система проверяет наличие необходимых материалов и уведомляет об их отсутствии.
* Контроль за выполнением работ:
* Мастер отмечает начало и завершение работ в системе.
* Возможен ввод комментариев и фотографий для подтверждения выполненных работ.
* Формирование отчетности:
* Генерация отчетов по выполненным заявкам за определенный период.
* Отчеты могут включать данные о количестве заявок, среднем времени выполнения, затратах на материалы и другие показатели.
* Безопасность данных:
* Обеспечение защиты персональных данных клиентов и сотрудников.
* Ограничение доступа к информации в зависимости от роли пользователя.
* Интеграция с другими системами:
* Возможность интеграции с бухгалтерскими программами для автоматического учета затрат на материалы и оплату труда.

Архитектура системы

Система будет построена на базе трехуровневой архитектуры:

* Презентационный уровень: Веб-интерфейсы и мобильные приложения для пользователей.
* Бизнес-логика: Серверная часть, отвечающая за обработку запросов и выполнение бизнес-правил.
* Хранилище данных: База данных для хранения информации о клиентах, заявках, материалах и других сущностях.

Технологический стек

* Для реализации системы будут использованы следующие технологии:
* Фронтенд: React/React Native для веб-приложений и мобильных приложений соответственно.
* Бэкенд: Node.js с использованием Express.js для серверной части.
* База данных: PostgreSQL для хранения данных.
* Контейнеризация: Docker для упрощения развертывания и масштабируемости.
* CI/CD: GitLab CI/CD для автоматической сборки и деплоя кода.

Процесс разработки

Процесс разработки будет организован по методологии Agile с использованием Scrum. Команда будет работать над задачами в рамках двухнедельных спринтов. По завершении каждого спринта проводится демонстрация результатов и ретроспектива для анализа проделанной работы и улучшения процесса.

Тестирование

* Тестирование будет проводиться на всех этапах разработки. Для этого используются следующие подходы:
* Unit-тестирование: Написание тестов для отдельных функций и модулей.
* Интеграционное тестирование: Проверка взаимодействия различных компонентов системы.
* UI-тестирование: Автоматизированное тестирование пользовательского интерфейса.
* Нагрузочное тестирование: Оценка производительности системы под нагрузкой.

Документация

Документирование кода будет осуществляться с помощью JSDoc для JavaScript и SQL-комментариев для базы данных. Также будет поддерживаться актуальная документация по API и руководства для пользователей.

**Руководство сопровождения программного обеспечения**

Введение

Настоящее руководство разработано для сопровождения программного продукта, предназначенного для учета заявок на ремонт бытовой техники. Документ включает в себя процедуры и рекомендации по техническому сопровождению, устранению неисправностей, модернизации и улучшению функциональности системы.

Цель документа

Целью данного руководства является обеспечение бесперебойной работы программного продукта, минимизация простоев и своевременное реагирование на возникающие проблемы. Руководство также направлено на поддержание высокого уровня качества обслуживания пользователей и удовлетворение их потребностей.

Сфера применения

Руководство применяется всеми сотрудниками, участвующими в процессе сопровождения программного продукта, включая системных администраторов, технических специалистов, разработчиков и менеджеров проектов.

Основные задачи сопровождения

* Мониторинг работоспособности системы
* Постоянный контроль за состоянием серверов и баз данных.
* Анализ логов и выявление потенциальных проблем.
* Устранение неисправностей
* Быстрая реакция на сообщения об ошибках и сбоях.
* Диагностика причин возникновения проблем и их устранение.

Обновление и модернизация

Внедрение новых версий программного обеспечения.

Добавление новых функций и улучшений.

Поддержка пользователей

Консультации и помощь пользователям при возникновении вопросов.

Обучение новых пользователей работе с системой.

Процедуры сопровождения

Мониторинг работоспособности системы

Настройка мониторинговых инструментов

Использование систем мониторинга (например, Zabbix, Nagios).

Настройка оповещений о критических событиях.

Анализ логов

Ежедневный анализ логов сервера и базы данных.

Выявление аномалий и потенциальных угроз.

Проверка резервных копий

Периодическая проверка целостности резервных копий.

Тестирование возможности восстановления данных из резервных копий.

Устранение неисправностей

Прием сообщений об ошибках

Создание тикетов в системе управления инцидентами (например, Jira).

Определение приоритетов и сроков решения задач.

Диагностика проблемы

Сбор дополнительной информации у пользователей.

Воспроизведение ошибки в тестовой среде.

Исправление ошибок

Разработка и внедрение исправлений.

Тестирование исправленных функций перед внедрением в рабочую среду.

Документирование решений

Описание шагов устранения проблемы в базе знаний.

Обновление документации по эксплуатации системы.

Обновление и модернизация

Планирование обновлений

Составление плана внедрения новых версий ПО.

Согласование сроков с пользователями и руководством.

Тестирование обновлений

Проведение тестирования на тестовом стенде.

Проверка совместимости с существующим оборудованием и программным обеспечением.

Внедрение обновлений

Выполнение миграции данных и настроек.

Переход на новую версию в рабочее время с минимальным прерыванием сервиса.

Оценка результатов

Анализ отзывов пользователей после внедрения новой версии.

Корректировка планов дальнейших обновлений на основании полученных данных.

Поддержка пользователей

Консультации и обучение

Организация регулярных обучающих семинаров и вебинаров.

Предоставление справочной документации и инструкций.

Техническая поддержка

Ответы на вопросы пользователей через службу поддержки.

Решение возникающих проблем в режиме реального времени.

Сбор обратной связи

* Проведение опросов среди пользователей для оценки удовлетворенности работой системы.
* Внесение предложений по улучшению функционала на основании полученной информации.

Заключение

Следование данному руководству позволит обеспечить стабильную работу программного продукта для учета заявок на ремонт бытовой техники, минимизировать риски возникновения сбоев и повысить качество предоставляемых услуг.

**Руководство системного программиста**

Введение

Данное руководство предназначено для системных программистов, участвующих в разработке и сопровождении программного продукта для учета заявок на ремонт бытовой техники. Оно охватывает технические аспекты проектирования, разработки, настройки и сопровождения системы, а также взаимодействие с другими компонентами ИТ-инфраструктуры организации.

Описание предметной области

Программный продукт предназначен для автоматизации процесса учета заявок на ремонт бытовой техники. Основные функции включают:

* Регистрацию новых заявок от клиентов.
* Назначение мастеров для выполнения ремонтных работ.
* Учет материалов и запчастей, используемых в ремонте.
* Контроль за выполнением работ и закрытии заявок.
* Формирование отчетов о выполненных работах.

Основные требования к системе

Регистрация заявки:

* Клиенты должны иметь возможность подавать заявки через веб-интерфейс или мобильное приложение.
* Заявки должны содержать информацию о клиенте, технике, характере неисправности и предпочтительных сроках ремонта.

Назначение мастера:

* Система должна автоматически назначать мастера на основе его доступности и специализации.
* Должна быть предусмотрена возможность ручного назначения мастера в случае необходимости.

Учет материалов и запчастей:

* Необходимо вести учет наличия запчастей и расходных материалов на складе.
* При назначении мастера система должна проверять доступность нужных материалов и уведомлять о недостатке.

Контроль за выполнением работ:

* Мастера должны отмечать начало и окончание работ в системе.
* Возможно добавление комментариев и фотографий для подтверждения выполненных работ.

Формирование отчетности:

* Система должна генерировать отчеты по выполненным заявкам за выбранный период.
* Отчеты могут включать количество заявок, среднее время выполнения, затраты на материалы и другие показатели.

Безопасность данных:

* Обеспечивать защиту персональных данных клиентов и сотрудников.
* Ограничивать доступ к информации в зависимости от ролей пользователей.

Интеграция с другими системами:

* Возможность интеграции с бухгалтерскими программами для учета затрат на материалы и оплату труда.
* Архитектура системы

Система строится на базе трехуровневой архитектуры:

* Презентация: Веб-интерфейсы и мобильные приложения для конечных пользователей.
* Логический слой: Серверная часть, реализующая бизнес-логику и взаимодействующая с хранилищем данных.
* Хранилище данных: Базы данных для хранения информации о клиентах, заявках, материалах и других сущностях.

Технологический стек

Для реализации системы применяются следующие технологии:

* Фронтэнд: React/React Native для веб-приложений и мобильных приложений соответственно.
* Бэкэнд: Node.js с использованием Express.js для серверной части.
* База данных: PostgreSQL для хранения данных.
* Контейнеризация: Docker для упрощения развертывания и масштабируемости.
* CI/CD: GitLab CI/CD для автоматической сборки и деплоя кода.

Процесс разработки

Процесс разработки организован по методике Agile с использованием Scrum. Работа ведется в рамках двухнедельных спринтов. По окончании каждого спринта проводятся демонстрации результатов и ретроспективы для анализа проделанного и улучшения процесса.

Тестирование

Тестирование проводится на всех этапах разработки:

* Unit-тестирование: Написание тестов для отдельных функций и модулей.
* Интеграционное тестирование: Проверка взаимодействия различных компонентов системы.
* UI-тестирование: Автоматическое тестирование пользовательских интерфейсов.
* Нагрузочное тестирование: Оценка производительности системы под высокой нагрузкой.

Документация

Код документируется с использованием JSDoc для JavaScript и SQL-комментариев для базы данных. Также поддерживается актуальная документация по API и руководства для пользователей.

Установка и настройка системы

Требования к оборудованию:

* Минимум 8 ГБ оперативной памяти.
* Процессор Intel Core i5 или аналогичный.
* 100 ГБ свободного пространства на жестком диске.
* Операционная система Ubuntu 20.04 LTS или новее.

Шаги установки:

* Установить необходимые пакеты
* Клонировать репозиторий с исходным кодом
* Запустить контейнеры с помощью Docker Compose
* Проверить работоспособность системы

Конфигурация системы:

* Настроить базу данных: создать базу данных и пользователя, предоставить ему права.
* Настроить подключение к базе данных
* Настроить серверную часть, изменив параметры в файле конфигурации.
* Поддержка и обслуживание системы

Резервное копирование:

* Создавать резервные копии базы данных регулярно
* Восстанавливать базу данных из резервной копии
* Мониторинг и логирование:
* Использовать Prometheus и Grafana для мониторинга состояния системы.

Обновление системы:

* Остановить текущую версию системы
* Обновить исходный код
* Запустить обновленную версию системы

Заключение

Данное руководство предоставляет подробные инструкции по установке, настройке и сопровождению программного продукта для учета заявок на ремонт бытовой техники. Следуя этим рекомендациям, вы сможете эффективно управлять системой и обеспечивать ее надежную работу.

|  |  |
| --- | --- |
| Оценочные элементы | Методы оценки |
| Заявка на ремонт | Анализ формы заявки, интервью с пользователями |
| Регистрация заявки | Протоколирование действий, логи системы |
| Обработка заявки | Временные метрики, автоматизированное тестирование |
| Исполнение заявки | Отслеживание статуса, обратная связь от исполнителей |
| Отчетность и информирование | Аудит отчетов, опросы пользователей |
| Мониторинг и анализ | Инструменты мониторинга, анализ логов |