

Описание жизненного цикла, поддержки и обслуживания программного обеспечения «Коммерческая электронная торговая площадка (ЭТП)»

Аннотация

Данный документ содержит описание процессов, обеспечивающих поддержание жизненного цикла программного обеспечения «Коммерческая электронная торговая площадка (ЭТП)», устранение неисправностей, выявленных в ходе эксплуатации программного обеспечения, совершенствование программного обеспечения, а также информацию о персонале, необходимом для обеспечения такой поддержки. Документ предназначен для администраторов и пользователей программного обеспечения «Коммерческая электронная торговая площадка (ЭТП)».

Термины и определения

- **ПО** — программное обеспечение.
- **СУБД** — система управления базами данных.
- **QA** — контроль качества (Quality Assurance).
- **ТЗ** — техническое задание.

Процессы жизненного цикла программного обеспечения

1. Общие сведения

Жизненный цикл программного обеспечения «Коммерческая электронная торговая площадка (ЭТП)» обеспечивается в соответствии с требованиями ГОСТ Р ИСО/МЭК 12207-2010. Программное обеспечение представляет собой электронную торговую площадку, предназначенную для проведения закупочных процедур между заказчиками и участниками. Система обеспечивает полный цикл проведения закупки, включая аккредитацию поставщиков, проведение аукционов, конкурсов, запросов предложений, а также попозиционных аукционов.

Программное обеспечение реализовано в виде web-приложения, разработанного с использованием Symfony, Vue.js и PostgreSQL.

2. Процессы организационного обеспечения проекта

2.1. Процесс менеджмента модели жизненного цикла

Модель жизненного цикла разработки программного обеспечения основана на методологии Scrum. Организация определяет, поддерживает и совершенствует процессы жизненного цикла для обеспечения бесперебойной разработки и сопровождения системы. Ответственность за актуальность и корректность применяемых процессов несёт руководитель проектов.

В результате успешного осуществления данного процесса обеспечивается согласованность применяемых подходов к планированию, разработке и сопровождению программного обеспечения, а также единообразие процедур.

2.2. Процесс менеджмента инфраструктуры

Инфраструктура разработки размещена на собственных серверах организации. Для обеспечения процессов жизненного цикла используются следующие средства: баг-трекинг осуществляется в Jira (серверной версии), хранение исходного кода — в GitLab, процессы непрерывной интеграции и доставки обеспечиваются средствами CI\CD GitLab. Тестовые среды размещены на внутренних ресурсах компании и поддерживаются штатными специалистами.

В результате успешного осуществления данного процесса обеспечивается доступность ресурсов, необходимых для выполнения задач разработки, тестирования и сопровождения.

2.3. Процесс менеджмента портфеля проектов

Процесс менеджмента портфеля проектов направлен на обеспечение развития «Коммерческая электронная торговая площадка (ЭТП)» в соответствии со стратегией продукта. Решения о развитии принимаются продуктовой командой, которая формирует план работ, расставляет приоритеты и корректирует направление развития системы. Утверждение новых версий и планов развития осуществляется этой же командой.

В результате успешного осуществления процесса обеспечивается актуальность и обоснованность развития продукта, своевременное распределение ресурсов и поддержка жизненного цикла системы.

2.4. Процесс менеджмента людских ресурсов

В процессе разработки и сопровождения программного обеспечения участвуют системные аналитики, бизнес-аналитики, разработчики, тимлиды, специалисты по тестированию

(QA), DevOps-инженеры и техническая поддержка. Распределение ролей и ответственность определяются потребностями проекта.

В результате успешного осуществления процесса обеспечивается устойчивость проекта, соответствие компетенций требованиям задач и оперативное закрытие потребностей в ресурсах.

2.5. Процесс менеджмента качества

В рамках обеспечения качества применяются код-ревью, статический анализ кода, ручное функциональное тестирование, регрессионное тестирование, автоматизированные unit-тесты и UI-тесты. Регресс выполняется в каждом спринте, что обеспечивает контроль стабильности продукта.

В результате успешного осуществления процесса обеспечивается своевременное выявление и устранение дефектов, повышение стабильности и предсказуемости поведения системы.

3. Процессы проекта

3.1. Процесс планирования проекта

Планирование работ выполняется совместно с бизнес-подразделением. Оценка объёмов работ проводится на планингах. Разработка ведётся двухнедельными спринтами, по итогам которых формируются релизы.

В результате успешного осуществления процесса обеспечивается прогнозируемость разработки и контроль сроков выполнения задач.

3.2. Оценка проекта и процесс управления

Процесс управления проектом включает контроль выполнения задач и сроков ответственными лицами, руководителями проектов и проджект-менеджерами. Ежедневно проводятся дейли-встречи, по завершении каждого спринта — ретроспектива и демонстрация реализованного функционала.

В результате успешного осуществления процесса обеспечивается выполнение проектных обязательств и поддержание прозрачности разработки.

3.3. Процесс менеджмента решений

Технические решения принимаются архитектурным комитетом в рамках крупных доработок. В оперативной деятельности решения принимаются тимлидом.

В результате успешного осуществления процесса обеспечивается согласованность технических решений и соответствие архитектурной стратегии проекта.

3.4. Процесс менеджмента рисков

В рамках проекта рассматриваются риски, связанные с доступностью внешних интеграций, задержками со стороны заказчиков, ошибками постановки требований, техническими сбоями инфраструктуры или человеческим фактором. Управление рисками осуществляется посредством оперативного мониторинга, анализа возникающих инцидентов, корректировки планов и принятия соответствующих мер по их предотвращению.

В результате успешного осуществления процесса повышается устойчивость проекта и способность своевременно реагировать на внешние и внутренние угрозы.

3.5. Процесс измерений

В процессе разработки и сопровождения собираются данные, характеризующие состояние продукта: количество инцидентов, время реакции на обращения, скорость выполнения задач в спринтах, степень покрытия тестами, частота выпуска релизов. На основе этих данных проводится анализ и планирование улучшений.

В результате успешного осуществления процесса обеспечивается объективная оценка состояния проекта и возможность принятия обоснованных управленческих решений.

4. Технические процессы

4.1. Процесс определения требований правообладателей

Требования к системе поступают от заказчика, продуктовой команды и внутренних аналитиков. Первичные требования формируются в виде пользовательских историй, после чего оформляются в виде технического задания и декомпозируются на отдельные задачи.

В результате успешного осуществления процесса формируется согласованный перечень требований, на основании которого осуществляется разработка.

4.2. Процесс анализа системных требований

Анализ системных требований выполняется системным архитектором и системными аналитиками. Для уточнения требований применяются модели, диаграммы и схемы. Все требования проходят обязательное согласование.

В результате успешного осуществления процесса обеспечивается полнота, непротиворечивость и проверяемость требований.

4.3. Процесс квалификационного тестирования системы

Квалификационное тестирование осуществляется командой тестирования (QA). В каждой итерации выполняется функциональное, регрессионное и интеграционное тестирование. Нагрузочные испытания проводятся при необходимости. Результаты тестирования фиксируются и учитываются при подготовке релиза.

В результате успешного осуществления процесса подтверждается готовность системы к поставке.

4.4. Процесс сопровождения программных средств

Сопровождение системы осуществляется посредством выпусков релизов и установки хотфиксов при возникновении критических ситуаций. Развёртывание изменений поддерживается процессами автоматизации CI/CD. Релизные заметки оформляются и сохраняются в Confluence (серверной версии).

В результате успешного осуществления процесса обеспечивается контролируемое развитие продукта и поддержание его работоспособности.

5. Процессы реализации программных средств

5.1. Процесс реализации

Разработка программного обеспечения осуществляется с использованием Symfony, Vue.js и PostgreSQL. В проекте применяются стандарты кодирования и внутренние гайдлайны, обеспечивающие единый стиль разработки.

В результате успешного осуществления процесса формируются программные компоненты, соответствующие архитектурным требованиям.

5.2. Процесс анализа требований к программным средствам

Детализация требований к программным модулям осуществляется архитектором, который определяет технические ограничения и согласует их с аналитиками.

В результате успешного осуществления процесса обеспечивается целостность и однозначность требований к программным элементам.

5.3. Процесс проектирования архитектуры программных средств

Архитектура «Коммерческая электронная торговая площадка (ЭТП)» построена по клиент-серверной модели. Документация по архитектуре находится в процессе формирования и актуализируется по мере развития системы.

В результате успешного осуществления процесса обеспечивается структурированность программных компонентов и их взаимная согласованность.

5.4. Процесс квалификационного тестирования программных средств

Квалификационное тестирование модулей выполняется QA-специалистами. Каждый модуль проверяется на соответствие требованиям и корректность работы. Результаты тестирования фиксируются уполномоченными специалистами и учитываются при принятии решения о релизе.

6. Процессы поддержки программных средств

6.1. Процесс менеджмента документации программных средств

Ведутся эксплуатационная, техническая, архитектурная и API-документация. Все материалы размещаются в Confluence (серверной версии).

6.2. Процесс обеспечения гарантии качества программных средств

Контроль качества осуществляется посредством чек-листов, тест-кейсов, прогонов в Zephyr (серверной версии) и согласования результатов тестирования ответственными специалистами.

6.3. Процесс валидации программных средств

Готовность программного обеспечения подтверждается руководителем тестирования. После выполнения приёмочного тестирования результаты согласуются проджект-менеджером.

6.4. Процесс ревизии программных средств

Проводятся регулярные ревью кода, ревизии архитектурных решений и внутренние проверки процессов разработки с целью выявления возможных отклонений и обеспечения качества.

6.5. Процесс решения проблем в программных средствах

Инциденты фиксируются в Jira (серверной версии). Проблемы анализируются, классифицируются и устраняются в установленном порядке. Реакция на инциденты осуществляется оперативно. Формальный SLA находится в стадии формирования.

Порядок технической поддержки программного обеспечения

Техническая поддержка включает консультирование пользователей, анализ возникающих ситуаций и устранение неисправностей. Поддержка осуществляется в рамках текущих регламентов и обеспечивается силами специалистов проекта.

Техническая поддержка первого уровня

Первичный приём обращений осуществляется через чаты, телефонные звонки и электронную почту. На данном уровне поддержку оказывают специалисты QA, выполняющие диагностику инцидентов.

Техническая поддержка второго уровня

Устранение ошибок осуществляется разработчиками и техническими специалистами. Режим работы определяется организационными регламентами и зависит от текущих задач.

Техническая поддержка третьего уровня

При необходимости выполняется эскалация обращений к разработчикам продукта или архитектурной группе для решения сложных или критических проблем.

Требования к персоналу

К эксплуатации программного обеспечения допускаются пользователи, обладающие базовыми навыками работы с браузером. Интерфейс системы интуитивно понятен и не требует специализированных знаний. Администратору необходимы базовые навыки работы с информационными системами. Сотрудники технической поддержки и разработчики должны обладать базовыми профессиональными компетенциями, достаточными для выполнения своих обязанностей в рамках сопровождения и развития программного обеспечения.