

Описание процессов жизненного цикла "Jay Data"

Версия 1.0: © ООО «Кросстех Солюшнс Групп», 2023.03

Содержание

Аннотация	1
Общая информация	2
Описание системы	2
Жизненный цикл системы	4
Поддержка версий и доработка	4
Процессы, обеспечивающие поддержание ЖЦ	5
Технические процессы	5
Разработка и совершенствование	5
Внедрение	6
Документирование	7
Управление развертыванием	7
Поддержка и сопровождение	8
Обеспечение гарантии качества	8
Тестирование	9
Аудит	9
Устранение неисправностей, выявленных в ходе эксплуатации	9
Устранение сбойных ситуаций	10
Эксплуатация	10
Приобретение и поставка	11
Информация о персонале	12
Обучение и квалификация персонала	12
Информация о необходимом персонале для обеспечения поддержки	12
Размещение персонала вендора	13

Аннотация

В данном документе содержится описание процессов, обеспечивающих поддержание жизненного цикла системы, в том числе устранение неисправностей, выявленных в ходе эксплуатации, совершенствование программного обеспечения, а также информацию о персонале, необходимом для обеспечения такой поддержки.

Настоящий документ носит описательный характер.

Общая информация

Система управления тестовыми данными (далее - Система, англ. Test Data Manager, TDM) предназначена для поиска, классификации и обезличивания чувствительных данных путем их маскирования, что помогает устранить риски передачи конфиденциальной информации третьим лицам. Система позволяет защитить конфиденциальные данные от несанкционированного доступа с помощью алгоритмов маскирования.

Система разработана ООО «Кросстех Солюшнс Групп» (далее — Вендор) в соответствии с требованиями и лучшими практиками разработки систем подобного класса.

Система работает на различных операционных системах через web-интерфейс в браузере. В состав системы входят только российские компоненты.

Описание системы

Система осуществляет следующие процедуры:

- Подключение к базам данных;
- Профилирование данных;
- Отключение/включение служебных объектов в базах данных;
- Маскирование всех необходимых данных в соответствии с настройками;
- Проверка корректности проведённого маскирования;
- Управление методами маскирования;
- Управление доменами и политиками данных;
- Управление справочниками.

Особенности и возможности системы:

- Современный быстрый Web-интерфейс;
- Поддержка популярных БД: MS SQL, Oracle, PostgreSQL и др.;
- Возможность авторизации под доменными учётными данными;
- Возможность работы в разных часовых поясах;
- Возможность отслеживать историю изменения любых атрибутов и действий пользователя;
- Модуль администрирования для управления приложением без перезапуска.

Система поддерживает операционные системы:

- Windows;
- Linux.

Система поддерживает следующие браузеры:

- Mozilla FireFox;
- Яндекс.Браузер;
- Microsoft Edge;
- Internet Explorer;
- Google Chrome;
- и др.

Жизненный цикл системы

Жизненный цикл системы — это процесс развития, начинающийся со стадии замысла и заканчивающийся прекращением применения (далее — ЖЦ).

Ценность внедрения системы заключается в следующем:

- Снижение рисков передачи конфиденциальной информации, благодаря обезличиванию чувствительных данных;
- Сохранение оригинальности первоначальных данных;
- Защита конфиденциальной информации за счет исключения возможности восстановления реальных данных из замаскированных.

Поддержка версий и доработка

Система является развивающейся, поэтому ее ЖЦ носит циклический характер, т.е. является последовательностью ЖЦ отдельных версий системы — ее релизов. Версия системы - цифровое или цифро-буквенное обозначение конкретного образа системы, получаемого в процессе её разработки. В рамках настоящего документа существенными в версии системы считаются два первых числа. Версии системы имеют следующий общий вид: X.X.X (XXXX.XX), где X - цифра в диапазоне 0-9.

Пример 1. Общее правило версионирования:

<Мажорная версия>.<Минорная версия>.<Хотфикс> (<Год выпуска>.<Месяц выпуска>),
например: 1.0.1 (2023.03)

- **Мажорная версия** Основной номер версии или поколение системы. Одно поколение от другого может существенно отличаться по функциональным и нефункциональным возможностям;
- **Минорная версия** Вспомогательный номер версии системы. Используется для различения версий системы внутри одного поколения при несущественном изменении функциональности или исправление несущественных недостатков;
- **Хотфикс** Вспомогательный номер версии системы. Используется для различения версий системы внутри одного поколения при исправлении выявленных дефектов (hotfix).
- **<Год выпуска>** — обязательное поле.
- **<Месяц выпуска>** — обязательное поле. Начинается с 0, т.е.: 01, 02, 03, 04 и др.

К каждому новому релизу в рамках доработки привязывается список историй (Story) и исправленных дефектов в аналитической системе Jira. При выпуске релиза к новой версии прикладывается Release Notes, в которых описан состав релиза с указанной версией.

Процессы, обеспечивающие поддержание ЖЦ

Жизненный цикл системы рассматривается с точки зрения ГОСТ Р ИСО/МЭК 12207-2010, Процессы ЖЦ программных средств (Information technology. System and software engineering. Software lifecycle processes). Процессы ЖЦ реализуются под управлением сторон, вовлеченных в ЖЦ. Под стороной понимают одну из тех организаций, которые инициируют или выполняют разработку, эксплуатацию или сопровождение системы.

Основными сторонами являются:

- Заказчик — он же конечный пользователь, эксплуатирующий систему;
- Вендор — он же Разработчик, обеспечивающий разработку, внедрение, модернизацию и сопровождение системы.

Технические процессы

Разработка и совершенствование

Процесс разработки и совершенствования — технический процесс, посредством которого потребности пользователей преобразуются в программный продукт.

Для системы определена долгосрочная стратегия развития (Roadmap), в рамках которой составляется план на год. Новые версии системы (релизы) выпускаются, как правило, каждые 3 месяца. Годовой план детализируется и корректируется для каждой выпускаемой версии.

Стратегическое видение развития системы формируется руководством **Дирекции по технической деятельности** Вендора и дополняется рекомендациями от службы эксплуатации, отдела развития продуктов и пожеланиями от конечных пользователей системы, которые фиксируются службой клиентской поддержки в специализированной аналитической системе Jira. Функциональные и нефункциональные требования к системе формируют план каждого релиза. Задачи записываются в бэклог, который ведётся в аналитической системе Jira и приоритезируются владельцем продукта.

Процесс разработки и совершенствования строится на базе методологии Scrum по принципам Agile с управлением задачами в учётной системе Jira. Требования к системе описываются в Confluence и декомпозируются таким образом, чтобы обеспечить возможность реализации каждой отдельной функции системы в виде отдельного Merge Request'a в Gitlab.

При реализации каждая разрабатываемая функция проходит полный жизненный цикл:

- Анализ;
- Проектирование;
- Разработка;
- Код ревью;
- Тестирование;

- Приемка.

Результат каждой итерации анализируется и корректирует план следующей итерации. Каждый компонент системы разрабатывается силами отдельных команд. Это означает, что команды укомплектованы специалистами, которые гарантируют полный цикл реализации требований к компоненту без привлечения внешних специалистов: владелец продукта как источник знания о бизнес-требованиях, разработчики, тестировщики и системные аналитики необходимой квалификации.

Бизнес-требования заводятся в системе Confluence в виде описания. Исходя из описания производится декомпозиция на истории (Story), а затем на отдельные задачи (Issues) в Jira для программирования и определяются сроки их выполнения.

Каждая задача разрабатывается в отдельной ветке Git. После проведения функционального тестирования и стабилизации, соответствующая ветка проходит pipeline со сборкой, Unit-тестами и т.д., и, в случае успеха, сливается с веткой основной разработки develop. В конце каждой итерации основная ветка разработки проходит регресс-тестирование для подтверждения общей работоспособности системы.

Перед релизом все компоненты системы развертываются совместно для релизного оценочного и интеграционного тестирования системы. На базе результатов тестирования в соответствии с критериями качества принимается решение о выпуске релиза или возвращении системы на доработку. В случае принятия решения о выпуске релиза в Git создается отдельная ветка с префиксом "-rc" (Release Candidate), после чего она специалистами отдела автоматизации сливается с веткой со стабильным кодом master.

Инсталляция системы осуществляется Вендором или Заказчиком самостоятельно. Инструкция по инсталляции и дистрибутив предоставляется Заказчику Вендором. По запросу Заказчика ему предоставляются сертификаты на переданные лицензии.

Внедрение

Заказчик, получив дистрибутивы и лицензионные сертификаты от Вендора, организует развёртывание системы в своей инфраструктуре и установку лицензий. Лицензионные права на систему передаются Заказчику в рамках договора на внедрение.

Процесс внедрения и доработки функционала под требования Заказчика, а также процесс функционирования, сопровождения и устранения возникающих ошибок производится в соответствии с договором на внедрение. При необходимости Вендор оказывает требуемую помощь и устраняет возникающие ошибки.

Основные цели и задачи внедрения:

- Развёртывание системы в инфраструктуре Заказчика;
- Подключение пользователей Заказчика к системе;
- Подключение системы к полигонам и базам данных Заказчика;
- Передача и разработка всей необходимой документации;
- Обучение персонала Заказчика использованию и поддержке системы.

В процессе внедрения также может быть выполнено:

- Настройка любой части решения под бизнес-процессы Заказчика;
- Изменение/расширение модели бизнес-данных необходимыми объектами, изменение атрибутивного состава;
- Изменение/расширение экранов, форм, интерфейса;
- Изменение/расширение любой логики, связанной с особенностями бизнес-процессов Заказчика;
- Проведение интеграции с любой существующей системой Заказчика для получения или передачи данных;
- Разработка необходимых отчётов, виджетов, дашбордов;
- и многое другое, в зависимости от пожеланий Заказчика.

Документирование

Процесс управления документацией - является неотъемлемой частью всех стадий и этапов ЖЦ системы. Документирование происходит одновременно с процессами проектирования и разработки каждого из релизов системы или после них.

Заказчику в составе дистрибутивов предоставляются документация в составе:

- Текущего документа;
- Документации на Jay Data.

Документация на систему написана с применением формата AsciiDoc.

После внедрения системы Заказчику предоставляется отчётная документация по проекту, в составе которой могут быть:

- Организационно-техническое архитектурное решение;
- Пояснительная записка к техническому заданию;
- Инструкция пользователя;
- Инструкция администратора;
- Программа и методика испытаний;
- и др.

Управление развертыванием

Развёртывание и сборка исходных кодов системы поддерживается системой управления версиями Gitlab. Система запускается как набор Docker-контейнеров, взаимодействующих между собой. Последовательность запуска Docker-контейнеров определяется специальным файлом-оркестратором, входящим в состав дистрибутива системы. Для сборки Компонентов на основе исходных кодов используется сервер непрерывной интеграции и доставки (CI/CD) Gitlab, который позволяет быстро выявлять проблемы интеграции, немедленно прогонять тесты для свежих изменений, а также вместе с текущей стабильной версией иметь другие

версии сборок — для тестирования, демонстрации, и др.

После каждого релиза система со всем окружением разработки развёртывается на независимом сервере, расположенном в devops-инфраструктуре.

Каждый релиз системы выпускается системой управления версиями Gitlab в виде отдельного дистрибутива, который представляет собой .zip-архив. В архиве содержится .rpm-пакет для развёртывания системы на серверах заказчика и лицензия в виде .lic-файла с публичным ключом public.key для проверки валидности лицензии.

Поддержка и сопровождение

Гарантийная поддержка системы осуществляется Вендором. Срок гарантийной поддержки определяется договором на внедрение.

Плановое **техническое сопровождение** внедрённой системы осуществляется специалистами Заказчика. При необходимости Заказчик может обратиться за поддержкой к Вендору.

Обеспечение гарантии качества

Обеспечение качества программного обеспечения (англ. Software quality assurance, SQA) — набор процедур мониторинга разработки программного обеспечения и методов, используемых для обеспечения его качества.

Руководство по качеству при разработке Вендора (далее — РК) основано на требованиях стандартов системы менеджмента качества ГОСТ Р ИСО 9000-2015, 9001-2015, 9004-2010, 19011-2003, 10005-2007.

Управление качеством разрабатываемой системы обеспечивается условиями договорных отношений с Заказчиками в соответствии с требованиями [ГОСТ Р ИСО 9001-2015](#) и контролируется соответствующими службами и структурными подразделениями Вендора.

Процесс проверки качества Системы осуществляется в соответствии с запланированными мероприятиями, чтобы удостовериться, что получаемые в результате разработки системы результаты соответствуют поставленным целям и задачам. Проверка завершается до официального выпуска релиза системы.

Процедуры обеспечения качества у Вендора охватывают весь цикл разработки системы, включая такие процессы как: определение требований, проектирование, разработка, контроль исходного кода, анализ кода, конфигурационное управление, тестирование, управление релизами и интеграция продуктов.

Процедуры обеспечения качества включают:

- Цели;
- Возможности;
- Процедуры;
- Измерения;

- Проверки.

Тестирование

Для определения полноты соответствия установленных функциональных требований и созданного релиза у Вендора организованы следующие процессы тестирования:

- Функциональное тестирование (functional testing);
- Системное тестирование (system testing);
- Тестирование производительности (performance testing);
- Регрессионное тестирование (regression testing);
- Модульное тестирование (unit testing);
- Тестирование безопасности (security testing).

В случае выявления на этапе тестирования дефектов или несоответствия реализованного функционала функциональным требованиям задача переводится обратно в разработку, а затем, после доработки, снова в тестирование. Цикл повторяется до тех пор, пока указанная функциональность не будет полностью соответствовать заявленным требованиям.

Перед слиянием кода, исправляющего дефект, проводится код ревью, а также функциональное и регресс-тестирование на отдельно развёрнутом под данный дефект стенде.

Аудит

Вендор проводит регулярные аудиты (проверки) своей системы менеджмента качества (далее — СМК), процессов ЖЦ и самих разрабатываемых систем.

Назначением **аудита** системы является независимая оценка системы и процессов на предмет их соответствия применимым регулирующим документам, стандартам, руководящим указаниям, планам и процедурам.

Аудит является формально организованной деятельностью, участники которой выполняют определенные роли, такие как главный аудитор (lead auditor), второй аудитор (another auditor), регистратор (recorder) и инициатор (initiator). В аудите принимает участие представитель Вендора. В результате аудита идентифицируются случаи несоответствия и формируется отчет, необходимый команде разработки для принятия корректирующих действий, а также формируется конкретный план мероприятий. После реализации плана осуществляется верификация выполненных работ. Проводится как внутренний аудит, так и аудит с привлечением сторонних организаций.

Устранение неисправностей, выявленных в ходе эксплуатации

Проблемы, возникающие в процессе эксплуатации регистрируются в специализированной аналитической системе вендора Jira. Данная система используется не только Службой поддержки, но и другими структурными подразделениями. В данную систему также поступают все типы замечаний и пожеланий независимо от их источника возникновения.

Собранные проблемы категоризируются по критичности для использования Системы и по распространенности данной проблемы. Категоризированные проблемы анализируются на предмет возможных действий. Вновь выявленные типовые проблемы, которые можно решить силами Службы поддержки Вендора, добавляются в Базу знаний. Для вновь выявленных проблем, которые не могут быть решены силами Службы поддержки, формируется заявка на разработку. Данная заявка на разработку привязывается к ошибке в Jira и контролируется до момента устранения в следующем релизе. В некоторых случаях может быть разработано временное решение, реализуемое Службой поддержки и смягчающее остроту проблемы (hotfix), но не решающее проблему в полном объеме, тогда это временное решение добавляется в Базу знаний Службы поддержки.

Устранение сбойных ситуаций

Сбойные ситуации регистрируются в специализированной аналитической системе вендора Jira. После фиксации сбойной ситуации вендором производится классификация и приоритезация обращения.

Устранение сбойной ситуации в работе, возникшей по причине ошибочного поведения системы, выполняется сотрудниками Вендора в соответствии с SLA, определённым на этапе подписания договора на вреднение Заказчику. Устранение сбоев может осуществляться как в удалённом формате, так и с выездом на объект, если удалённое подключение по каким-либо причинам невозможно. Особо критичные сбои держатся на контроле непосредственно у руководителя управления разработки и развития информационных систем со стороны Вендора. После устранения сбойной ситуации Вендором проводится работа по выявлению причин сбоя и минимизации рисков повторения подобной ситуации в будущем.

Эксплуатация

Эксплуатация системы заключается в исполнении, функционировании её на ЭВМ для обработки информации и в получении результатов, являющихся целью её создания, а также, в обеспечении достоверности и надежности выдаваемых данных.

Процесс эксплуатации состоит из работ и задач организации, эксплуатирующей систему. Процесс охватывает эксплуатацию системы и поддержку пользователей в процессе эксплуатации.

Оператор делает следующее:

- Управляет процессом эксплуатации на проектном уровне в соответствии с процессом управления;
- Определяет инфраструктуру для данного процесса в соответствии с процессом создания инфраструктуры;
- Адаптирует данный процесс к условиям проекта в соответствии с процессом адаптации;
- Управляет процессом эксплуатации на организационном уровне в соответствии с процессами усовершенствования и обучения.

Если оператор является поставщиком программной услуги, то оператор выполняет также процесс поставки.

Процесс эксплуатации системы состоит из следующих работ и задач:

- Подготовка процесса;
- Эксплуатационные испытания;
- Эксплуатация системы;
- Поддержка пользователя.

Приобретение и поставка

Приобретение Заказчиком программного обеспечения осуществляется в рамках договора на поставку или лицензионного договора. В поставке Заказчику передаётся экземпляр программного обеспечения в виде .zip-архива с дистрибутивом, лицензией и документацией в составе.

Дистрибутив представляет собой .tar-архив, который содержит в себе пакеты, файлы конфигурации и скрипты запуска установки.

Лицензия представляет собой файл самой лицензии файл публичного ключа для проверки валидности лицензии. Лицензия может быть выдана Заказчику с разным набором параметров, в зависимости от условий договора.

По договору поставки Заказчику передается в собственность экземпляр программы, который рассматривается в качестве вещи. Покупатель может свободно владеть и распоряжаться (в т.ч. передавать) таким носителем с экземпляром системы (ст.1270 ГК РФ), однако саму систему он может использовать только путем записи в память ЭВМ и применения пределах объявленных функциональных возможностей (ст.1280 ГК РФ). Поставка системы может осуществляться в коробочной версии на материальных носителях в комплекте с документацией. Возможна поставка системы путем передачи экземпляра по сети связи. Существуют варианты поставки с установкой на оборудовании Заказчика непосредственно сотрудником поставщика на объекте Заказчика или в удаленном режиме.

Условия договора поставки системы зависят от коммерческих и финансовых договоренностей сторон.

Предметом лицензионного договора на программное обеспечение выступает предоставление права использования системы. Экземпляр системы к моменту заключения лицензионного договора может находиться у Заказчика, он может получить его впоследствии как самостоятельно, так и от лицензиара. В лицензионном договоре определяется, какие права (или способы использования ПО) предоставляются Заказчику, на какой срок и территорию. Данные права в виде параметров вносятся в предоставляемую лицензию и передаются вместе с экземпляром системы. Активируя лицензию Заказчик получает сроки и возможности использования системы, определённые лицензионным договором.

Информация о персонале

Обучение и квалификация персонала

Вендор обладает высококвалифицированной командой с опытом разработки программных продуктов и утилит на языке Java. Все специалисты компании проходят обучение и внутреннюю сертификацию, а также постоянно совершенствуют свои навыки и квалификацию.

Вендор уделяет значительное внимание повышению квалификации своих сотрудников. Необходимый уровень знаний, умений и навыков, как требование к каждой должности, определяется моделью профессиональных и личных компетенций, принятой для соответствующей позиции в компании (hard/soft skills' model). Все сотрудники активно участвуют в реальных проектах разработки, получая опыт и нарабатывая практику.

Цель повышения квалификации персонала — поддержание и повышение уровня квалификации персонала с учетом требований компании к определенной должности, роли.

Информация о необходимом персонале для обеспечения поддержки

Для работы с системой необходимы следующие специалисты:

- **Пользователь Системы** — Специалист (служащий организации) Заказчика, использующий Систему для осуществления собственных ежедневных трудовых функций;
- **Администратор Системы** — Специалист ИТ-подразделения Заказчика, ответственный за корректную работу прикладных сервисов внутри организации, а также оказывающий внутреннюю поддержку пользователей при работе с Системой;
- **Инженер** — Инженер ИТ-подразделения Заказчика, осуществляющий работы по технической поддержке системы на территории Заказчика.
- **Администратор ОС (Системный администратор)** — Специалист ИТ-подразделения Заказчика ответственный за корректную работу базовых сервисов внутри организации (Операционные системы, сетевая инфраструктура, политики безопасности).

Общая численность персонала, относящегося к перечисленным категориям, зависит от специфики решаемых прикладных задач, масштаба системы и количества одновременно работающих пользователей. Обучение персонала Заказчика возможностям и методикам настройки Системы может выполняться Вендором в рамках проекта по внедрению системы на завершающих этапах и обычно занимает от одного до нескольких дней (в зависимости от категории пользователя).

Первичный инструктаж пользователей по работе с системой осуществляется в форме демонстрации ключевых особенностей системы и её использования на рабочем месте пользователя. Инструктаж может быть проведен в дистанционной (вебинар) или очной формах. Длительность инструктажа около 120 минут. Обучение Администраторов и

Инженеров работе с системой и её развёртыванию осуществляется в рамках проекта внедрения или на специальных Вендорских курсах.

В ходе обучения слушатели знакомятся с возможностями поддержки и настройки и выполняют практические (лабораторные) работы. Обучение может быть проведено в очной и дистанционной (вебинар) формах. В зависимости от уровня подготовки слушателя длительность курса может быть скорректирована в соответствии с программой обучения, утвержденной Вендором.

Размещение персонала вендора

Персонал Вендора, задействованный в разработке и совершенствовании ПО, устранении сбойных ситуаций и техподдержки, включает сотрудников следующих отделов:

- отдела разработки;
- отдела автоматизации;
- отдела развития продуктов;
- отдела технической поддержки.

Фактический адрес размещения инфраструктуры разработки: Российская Федерация, г.Москва, Ленинградский пр-т., 31А, 125284

Фактический адрес размещения разработчиков: Российская Федерация, г.Москва, Ленинградский пр-т., 31А, 125284

Фактический адрес размещения службы поддержки: Российская Федерация, г.Москва, Ленинградский пр-т., 31А, 125284