



Архитектор 1С

Этапы и артефакты проекта(часть 2)



Проверить, идет ли запись

Меня хорошо видно && слышно?



Ставим "+", если все хорошо
"-", если есть проблемы

Давайте знакомиться!

Напишите свою должность в чат, чтобы я понимал уровень группы и смогу адаптироваться в процессе презентации



Кузин Роман

Должность преподавателя:

«Ведущий архитектор ИТ-систем» в компании «МТС Диджитал»,
СТО продукта по автоматизации финансовой и хозяйственной деятельности компании.

Об опыте:

Общий стаж работы в 1С с 2015 года
Из них разработчиком и ведущим разработчиком – 4 года
Архитектором, ведущим архитектором и team lead-ом – 4 года

Профессиональные интересы:

Повышение надежности и производительности конфигураций 1С
Повышение прозрачности разработки и управления изменениями в компании
Внедрение DEVOPS и SCRUM-практик в командах

Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в чате группы телеграм
Архитектор1С 2023-07 или пишите
в мой ТГ **@Kuzin_RV**



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Условные обозначения



Индивидуально



Время, необходимое
на активность



Пишем в чат



Говорим голосом



Документ



Ответьте себе или
задайте вопрос

Маршрут вебинара

Знакомство

Виды требований при старте разработки

Формирование технического задания разработчику

Прохождение процесса разработки и его моделирование

Виды собраний

Рефлексия

Цели вебинара

К концу занятия вы сможете

1. Познакомиться с основными шагами разработки на проекте
2. Пройдем основные этапы проекта, а также детализируем каждый этап проекта
3. Познакомимся с понятием артефакта, поймем основную цель каждого артефакта, рожденного на каждом этапе.

Смысл

Зачем вам это уметь

1. Гибко управлять процессом разработки и адаптировать его под реалии вашей компании

2. Управлять потерями при прохождении каждого шага процесса разработки и устранять их причины

3. Повысить качество и прозрачность разработки для каждого стейкхолдера и команды проекта

Старт разработки продукта и виды требований

Бизнес-требование

Кем составляется:

- Руководителем продукта, как начальной точкой, получения запроса от бизнеса в AGILE-проектах. Также бизнес-требование может приходиться от заказчика, напрямую бизнес-аналитику.

Где моделируется:

- IDEF0(СППР) в разделе «Функции системы».

Для чего используется:

- Для дальнейшей трансформации в артефакты типа EPIC (Эпик).

Бизнес-требование

Пример:

Хочу обеспечить интеграцию по командировкам между системой планирования командировок и системами 1С для отражения фактических затрат в рамках бюджета подразделения, а также автоматическом отражении в таблице рабочего времени и расчетном листе.

.

Функциональное требование

Кем составляется:

- Аналитиком, функциональным архитектором.

Где моделируется:

- СППР(Технический проект)
- PlantUml

Для чего используется:

- Для дальнейшей трансформации в артефакт User Story (Пользовательская история)

Задача архитектора на данном этапе:

- Проверить полноту входящих данных
- Смоделировать процесс, сформировать основные вехи реализации, нарисовать интеграционную схему
- Обозначить узкие места в разработке
- Сформировать задачу на разработку или помочь с ее постановкой аналитику/функциональному архитектору

Функциональное требование

Пример:

AS IS:

Сейчас для отражения командировки, бухгалтер выгружает реестр командировок в Excel из программы бронирования, далее обогащает его данными по бюджетам и проводит командировки в программе 1С ЗУП и 1С Бухгалтерия. Если лимит по командировкам превышен, то передает номер командировки администратору в программе бронирования для отклонения заявки

TO BE:

Реализовать интеграцию между программами 1С Бухгалтерия и 1С ЗУП. Предусмотреть механизм автоматической передачи номера командировки в программу бронирования для отклонения заявки.

Задача и подзадача на разработку

- Задачи – это то, на что дробится пользовательская история. Так, для разработки интеграции «Система бронирования командировки и 1С» могут понадобиться десятки задач, в которых будут вовлечены дизайнеры, программисты и тестировщики.
- Примеры задач по данному эпику со стороны 1С:
 - Реализовать API на стороне «1С:ЗУП» по приему командировки.
 - Реализовать API на стороне «1С:Бухгалтерия» по приему командировки.
 - Реализовать процесс по согласованию командировки в программе 1С Документооборот.
 - Реализовать передачу пакета «Командировка» через RabbitMQ во все корпоративные системы.
 - Реализовать отчет по командировкам в «1С:ЗУП», пришедшим из сторонней программы.

Шаблон описания интеграционной задачи

API	КД2	Бесшовная интеграция	Web-сервис	RMQ	Таблицы
-----	-----	----------------------	------------	-----	---------

Система получатель	Система отправитель	Объект системы приемник	Объект системы отправителя	Сущность	Когда происходит отправка?
ЗУП	Система командировок	Физическое лицо	документ "Заявка на командировку"	«Командировка», пакет RMQ	
ЗУП	Система командировок	Договор авторского заказа / Договор ГПХ	документ "Заявка на командировку"		

Документ «Командировка» в системе бронирования	Реквизит уже существует?	Документ «Командировка» (1С ЗУП)	Реквизит уже существует?	Поле JSON	Тип _ Значение поля	Что сделать, как искать	Используются ли еще таблицы для заполнения?	Комментарий
Поле «Employee»	+	Реквизит «Сотрудник»	+	employee	Строка	Передаем строку ищем по коду	Да, регистр в ЗУП "Гражданство физических лиц"	
Поле «Сумма»	+	Реквизит «Результат»	+	Result	Число	Заполняем из JSON	Нет	

Вопросы?



Ставим "+",
если вопросы есть



Ставим "-",
если вопросов нет



Задаем вопросы
голосом

Разработка

Артефакты:

- Код. Версия в хранилище или ветка в гите.
- Тесты. Feature-файл теста в СППР или в гите.
- Обработки или фичи, необходимые для реализации задачи, ее проверки. JSON, XML, обработка, выгруженная в файлы в гите(или выложенная на общей шаре).
- Настроечные файлы или автозаполняемые константы, без которых задача на проде не сможет функционировать.

Обязанность архитектора:

- Проконтролировать код.
- Проконтролировать, что тест сделан верно и проходит проверку.
- Проконтролировать, что все необходимые фичи лежат в репозитории или в шаре, из которой затем попадают на прод.

Рекомендации по адаптации кода систем 1С

- Каждая доработка обрамляется комментарием кода. Каждый объект системы имеет префиксацию.
- Добавление объектов на форму происходит программно. На каждый объект конфигурации выделяется отдельный общий модуль для добавления кнопок и реквизитов.
- Типовые роли не дорабатываются. Для новых объектов создаются новые роли.
- Модуль объекта при записи, как правило, не дорабатывается. Дополнительные функции при записи и перепроведении документа выносятся в отдельные подписки.
- Ссылки на элементы хранятся в отдельном объекте системы «Дополнительные настройки» и заполняются автоматически при обновлении релиза.
- Доработка систем, с помощью расширений возможна, но не рекомендуема на больших проектах от 5+ разработчиков без git-a.

Рекомендации по написанию запросов в системе

- Несоответствие условий запроса индексам таблиц
- Соединение с подзапросами
- Соединение с виртуальными таблицами
- Подзапросы в условиях соединений
- Использование условия ИЛИ в запросах
- Неоптимальное использование RLS
- Условия в запросе за скобками виртуальных таблиц
- Обращение через точку к полям составного типа
- Сложные запросы, использующие большое количество соединений
- Расчет остатков, оборотов по таблицам документов и таблицам движений регистров
- Выполнение преобразований над индексированным столбцом
- Поиск по произвольной подстроке
- Запросы вида ВЫБРАТЬ * ИЗ
- Использование ОБЪЕДИНИТЬ для объединения наборов строк, которые заведомо не могут содержать дубли

Вопросы?



Ставим "+",
если вопросы есть



Ставим "-",
если вопросов нет



Задаем вопросы
голосом

Виды собраний на проекте

- Poker планирования
- Sprint Planning
- Ежедневное Scrum-совещание
- Code-review
- Sprint review
- Sprint Retrospective

Практика

Посмотрим процесс движения задачи по доске

1. Откроем яндекс трекер
2. Сопоставим состояние каждой задачи с состоянием в колонке и наполним ее артефактами
3. Сделаем эмуляцию процесса от постановки задачи до попадания ее в продуктив

Вопросы?



Ставим "+",
если вопросы есть



Ставим "-",
если вопросов нет



Задаем вопросы
голосом

Рефлексия

Цели вебинара

Смогли ли мы достичь данных целей?

1. Познакомиться с основными шагами разработки на проекте
2. Пройдем основные этапы проекта, а также детализируем каждый этап проекта
3. Познакомимся с понятием артефакта, поймем основную цель каждого артефакта, рожденного на каждом этапе.

Ключевые тезисы

1. На каждом этапе проекта рождается артефакт, повышающий его прозрачность.
2. Назначение каждого этапа процесса – повышение производительности проекта и его оптимизация.
3. Правильно используя методы гибкой разработки и DEVOPS можно значительно увеличить производительность проекта и команды

Рефлексия



С какими впечатлениями уходите с вебинара?
Что в прошедшем занятии вам показалось наиболее полезным?
Насколько тема была для вас сложной?
По какому разделу вам не хватило информации и примеров?



Как будете применять на практике то, что узнали на вебинаре?

**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате**

Спасибо за внимание!

Приходите на следующие вебинары



Кузин Роман

Должность преподавателя:

«Ведущий архитектор ИТ-систем» в компании «МТС Диджитал»,
СТО продукта по автоматизации финансовой и хозяйственной деятельности
компании.

Об опыте:

Общий стаж работы в 1С с 2015 года
Из них разработчиком и ведущим разработчиком – 4 года
Архитектором, ведущим архитектором и team lead-ом – 4 года

Профессиональные интересы:

Повышение надежности и производительности конфигураций 1С
Повышение прозрачности разработки и управления изменениями в компании
Внедрение DEVOPS и SCRUM-практик в командах