



Ведение DS- проектов

Лекция 2



Алексей Кузьмин

Директор
разработки
ДомКлик.ру

Работаю в ДомКлик.ру с 2016 года
Руководжу направлением Data Science и
работы с данными
До этого работал в компании АBBYY, где
занимался распознаванием языков со
сложной письменностью
Окончил мехмат МГУ

О чем поговорим?

Сегодня на лекции

01

Agile

Гибкая методика разработки

02

CrispDM

Методология ведения DS-проектов

03

Смотрим ноутбук

С небес на землю

Agile

Зачем оно?

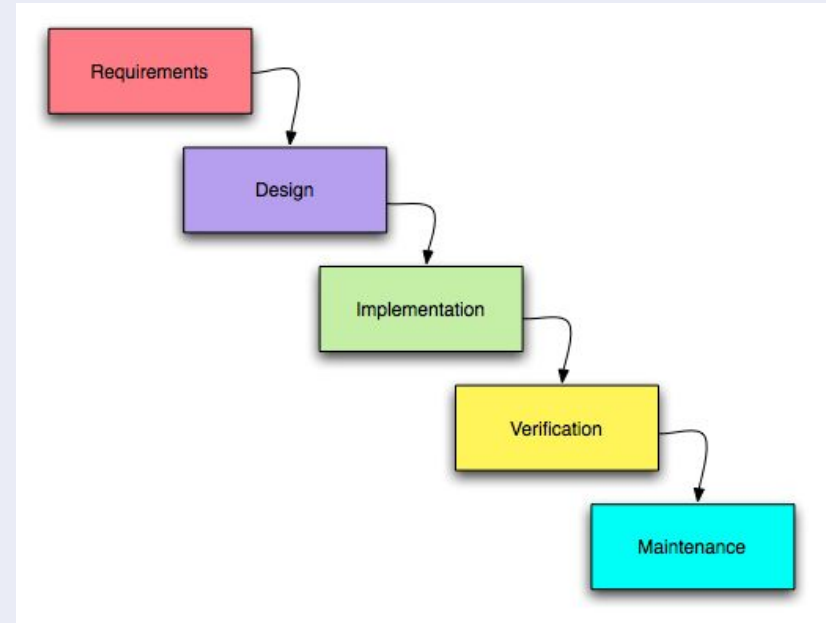
О причинах появления процессов

Сложная задача без процесса несет под собой много рисков:

- Человеческий фактор
- Отсутствие внятного контроля
- Сложности приемки-передачи результатов заказчику
- И тп

Старый подход – фиксировать все в ТЗ

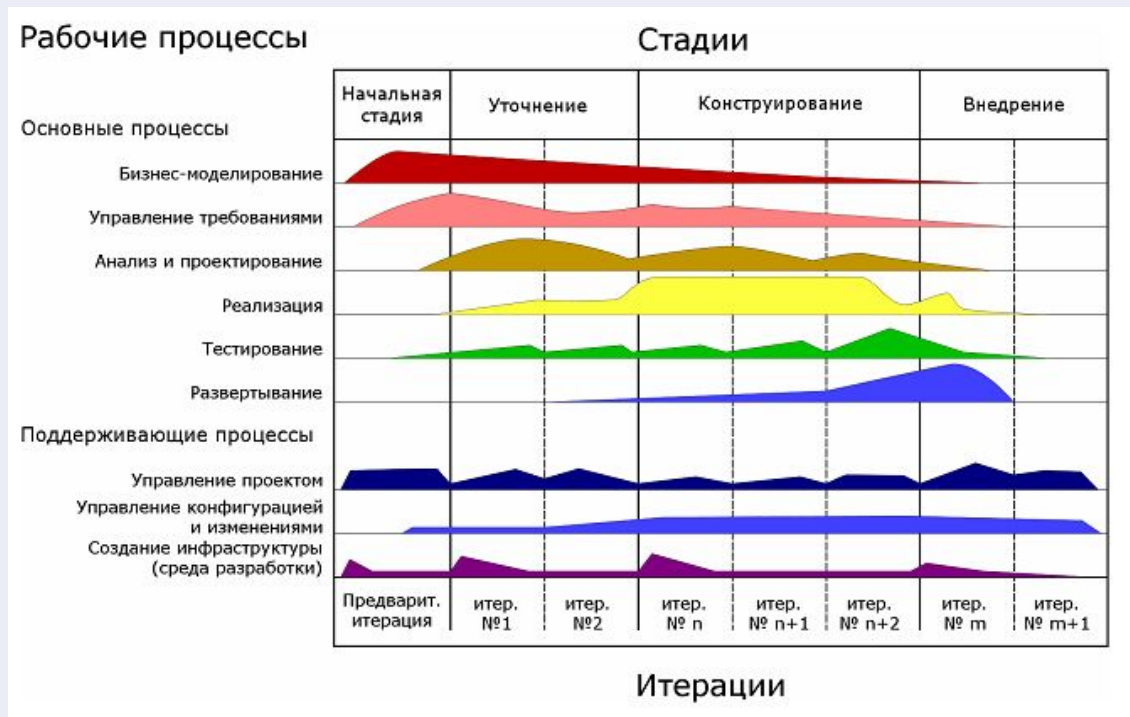
- Подробно все проектируем, рисуем интерфейсы, описываем в ТЗ
- Кодируем
- Тестируем
- Сдаем проект/этап Заказчику



Проблемы


- Заказчик выпадает из проекта на время его разработки - не видит результат и не может на него оперативно влиять
- Затянутые сроки согласования/сбора ТЗ
- Часто приводит к далеко не идеальному результату

Итеративный подход



Agile

- Гибкий подход к разработке ПО.
- Лучшие практики:
 - Scrum
 - XP
 - TDD, etc.



Гибкость – это не технология, наука или продукт, а – культура

Agility is not a technology, science, or
product but a culture

Philippe Kruchten

Agile Манифест

Люди и взаимодействие важнее процессов и инструментов

Работающий продукт важнее исчерпывающей документации

Сотрудничество с заказчиком важнее согласования условий контракта

Готовность к изменениям важнее следования первоначальному плану

То есть, не отрицая важности того, что справа, мы всё-таки больше ценим то, что слева.

Ценности Agile

- Гибкость и простота
- Частые релизы
- Самоорганизующаяся команда
- Больше общения

Гибкость и простота

- Agile-процессы готовы к изменениям требований даже на поздних этапах разработки.
- Важна простота - искусство увеличения объема работ, которых удалось избежать.

Частые релизы

- Наивысший приоритет - удовлетворенность заказчика:
 - ранние и периодические поставки ПО
 - ПО работающее и ценное для заказчика
- Продолжительность каждой итерации - от пары недель до пары месяцев.
- Предпочтение - коротким интервалам.

Самоорганизующаяся команда

- Над проектом работают мотивированные люди.
- Создаются все условия, поддержка и полное доверие.
- Самые лучшие архитектуры, требования и дизайны систем создаются самоорганизующимися командами.
- Команда сама организует оптимальный процесс.

Больше общения

- Потенциальные пользователи системы и разработчики должны работать вместе на протяжении всего проекта.
- Самый действенный и эффективный способ обмена информацией как внутри команды разработчиков, так и с внешним миром - непосредственное общение.

Agile не работает/плохо работает

- Garbage in - garbage out
- Есть строгий dead line
- Команда не самодостаточна для создания продукта (фронт в одной команде, бек - в другой и тп)

Scrum

Наиболее распространенная практика разработки в Agile.

Ключевые термины:

- Product backlog
- Sprint
- Daily scrum

Product Backlog

- Содержит список функциональных единиц системы (“user stories”), запланированных на след релиз

ID	Важн	Название	Описание	Как показать
248	75	Заставка (splash screen)	Как пользователь я хочу видеть заставку пока приложение открывается.	1. Запустить приложение – заставка показ. до появления главного окна

Product Backlog

- Product backlog один на весь релиз
- Им владеет менеджер продукта (“product owner”)
- Он не статичен – записи можно добавлять, удалять, менять им приоритет
- Общедоступен, но поддерживается одним человеком

Спринт (Sprint)

- Фаза разработки состоит из нескольких итераций – спринтов.
- Обычно спринт длится 1-2 недели.
- Этапы:
 - Планирование
 - Разработка
 - Демонстрация
 - Ретроспектива

Sprint Backlog

- Описывает задачи, запланированные командой на спринт
- Задачи – действия, необходимые для реализации запланированной на спринт функциональности
- В описание задачи входит ее оценка

Планирование (Sprint Planning)

- Проводится в начале спринта
- Участвует вся команда
- User stories разбиваются на задачи и оцениваются членами команды
- В результате команда подписывается на ту функциональность, на которую хватает времени спринта

Оценка

- Для оценки выбирается единица – идеальный человеко-день...или зеленый крокодил
- Следует оценить помехи (например focus factor между 0 и 1) перед каждым спринтом
- Результаты предыдущего спринта помогают лучше запланировать следующий






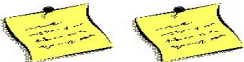
Ежедневный скрам (Daily Scrum)

- Проводится каждый день в фиксированное время
- Рекомендуется проводить стоя в течение 10-15 минут
- Если что-то нужно обсудить, назначается время после скрама

Вопросы

- Scrum Master спрашивает каждого:
 - Что ты делал?
 - Что ты собираешься делать?
 - Какие были проблемы?

Sprint whiteboard

PLANNED	IN PROGRESS	READY FOR TEST	DONE	BURN-DOWN
				
				UNPLANNED
				

Демонстрация (ревью)

- В конце каждого спринта проводится ревью
- Это демонстрация реализованной функциональности
- В ней может участвовать любой человек, задействованный в проекте
- В идеале после каждой демонстрации можно отправлять продукт заказчику

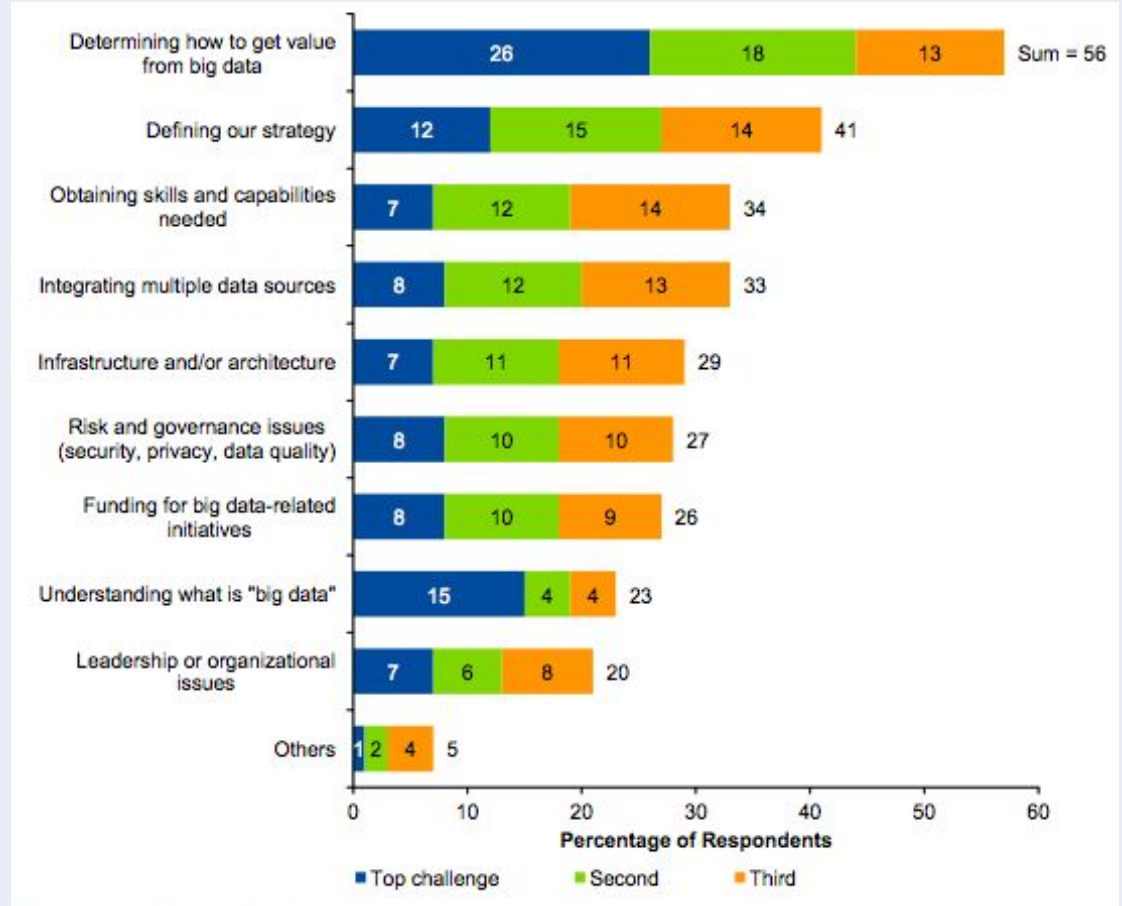


Ретроспектива спринта

- После каждого спринта (ревью)
- Участвуют все члены команды
- Цель - осознать:
 - Что было хорошо?
 - Что могло бы быть лучше
- Это обсуждение процесса, а не технических сложностей

CRISP DM

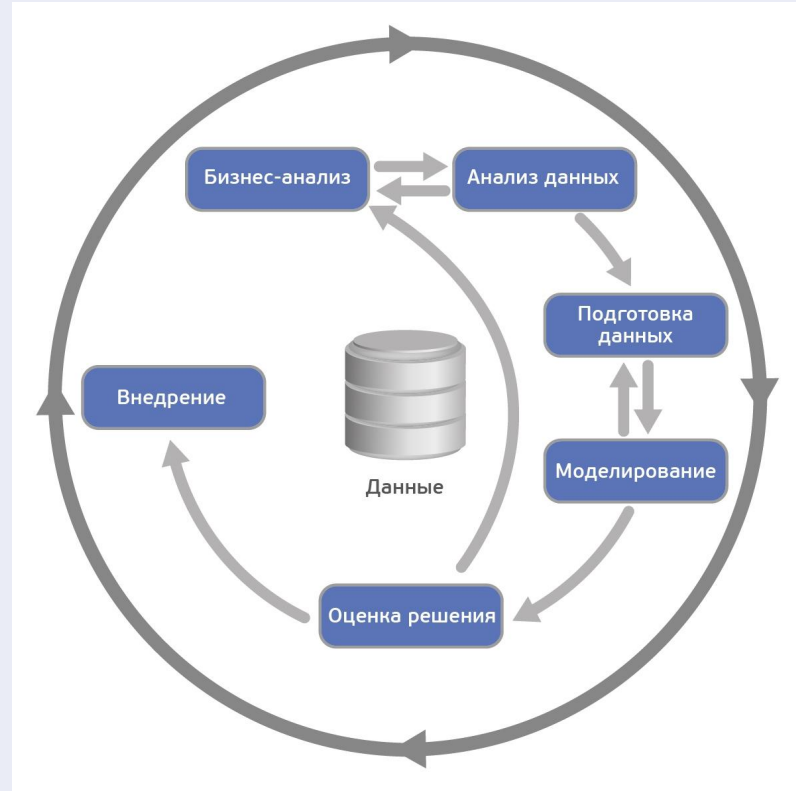
Top Big Data Challenges



Gartner, 2013

Кто помнит основные шаги Crisp DM? =)

Crisp DM



Business Understanding

1. Сбор справочной информации

- Составление бизнес-фона
- Определение бизнес-целей
- Критерии успеха проекта с точки зрения бизнеса

Business Understanding

2. Оценка ситуации

- Инвентаризация ресурсов
- Требования, предположения и ограничения
- Риски и непредвиденные обстоятельства
- Анализ затрат / выгод

Business Understanding

3. Определение целей ds

- Цели ds
- Критерии успеха ds

4. Создание плана проекта

Пример

Фаза	Время	Ресурсы	Риски
Business Understanding	1 неделя	Аналитики	Изменение условий
Data understanding	3 недели	Аналитики	Проблемы с данными / технологиями
Data preparation	4 недели	DS, DE	Проблемы с данными / технологиями
Modeling	2 недели	DS	Не удастся построить модель
Evaluation	1 неделя	Аналитики	Изменение условий, отсутствие результатов
Deployment	1 неделя	DS, Разработчик	Изменение условий, отсутствие результатов

Готовность к Data Understanding

С точки зрения бизнеса:

- Что бизнес надеется получить от этого проекта?
- Как будет определяться успех проекта?
- Есть бюджет и ресурсы?
- Есть ли доступ ко всем данным, необходимым для этого проекта?
- Обсуждали ли вы и ваша команда риски и непредвиденные обстоятельства, которые могут возникнуть?
- Оправдывают ли результаты вашего анализа затрат / выгод этот проект?

Готовность к Data Understanding

С точки зрения DS:

- Как конкретно анализ данных может помочь достичь целей бизнеса?
- Есть ли представление о том, какие методы DM могут дать наилучшие результаты?
- Как узнать, что результаты являются достаточными для нужд бизнеса?
- Как будут поставлены результаты моделирования? Отчет? Сервис?
- Включает ли план проекта все этапы CRISP-DM?
- Обозначены ли риски и зависимости в плане?

Data Understanding

1. Соберите имеющиеся данные (собственные, внешние и тп)
2. Опишите данные (количество, значения, связи и тп)
3. Исследуйте данные
4. Изучите качество данных (отсутствующие данные, ошибки в данных, плохие метаданные и тп)

Готовность к Data Preparation

- Все ли источники данных четко определены и доступны? Известно ли о каких-либо проблемах или ограничениях?
- Определены ключевые атрибуты? Эти атрибуты помогли вам сформулировать гипотезы?
- Определен ли размер всех источников данных? Можно ли использовать подмножество данных, где это уместно?
- Рассчитаны базовые статистики для каждого интересующего атрибута?
- Каковы проблемы качества данных для этого проекта?
- Четко ли определены этапы подготовки данных?

Data Preparation

Выберите данные (разделите на train/test, выберите признаки)

Очистите данные (Заполните пропуски, исправьте ошибки и тп)

Расширьте данные (Новые признаки и тп)

Сохраните данные (подготовьте data frame для обучения модели и сохраните его)

Готовность к Modeling

- На основании вашего первоначального исследования, смогли ли вы выбрать подходящие подмножества данных для моделирования?
- Эффективно ли вы очистили данные?
- Правильно ли соединены разные наборы данных?
- Задokumentировали ли Вы все сделанные шаги по подготовке данных?

Modeling

- Выберите метод моделирования
- Постройте модель
- Оцените модель
- Опишите результат

Готовность к Evaluation

- Дала ли модель понятные результаты? Есть ли очевидные несоответствия, которые требуют дальнейшего исследования?
- Исследовано более одного типа модели и результаты сравнены?
- Можно ли поставить модель заказчику?

Evaluation

1. Оцените результаты

- Четко ли представлены результаты?
- Есть ли какие-нибудь новые инсайты?
- Может ли модель и результаты быть применимыми к бизнесу?
- Какие дополнительные вопросы появились после моделирования?

2. Просмотрите процесс

- Что пошло не так и как это можно исправить?
- Есть ли альтернативные решения/действия, которые могли бы быть выполнены?

3. Определите следующие шаги

Deployment

1. Планирование внедрения
 - Для каждой модели создайте план внедрения.
 - Определите все проблемы внедрения и составьте план на случай непредвиденных обстоятельств.
2. Планирование мониторинга и технического обслуживания.
 - Определение моделей и результатов, которые требуют поддержки.
 - Как понять, что модель перестала быть актуальной?
 - Что делать в этом случае?
3. Провести итоговый обзор проекта

CRISP-DM

Business Understanding/ Бизнес-анализ	Data Understanding/ Анализ данных	Data Preparation/ Подготовка данных	Modeling/ Моделирование	Evaluation/ Оценка решения	Deployment/ Внедрение
Determine Business Objectives/ Определение бизнес-целей	Collect Initial Data/ Сбор данных	Select Data/ Выборка данных	Select Modeling Techniques/ Выбор алгоритмов	Evaluate Results/ Оценка результатов	Plan Deployment/ Внедрение
Assess Situation/ Оценка текущей ситуации	Describe Data/ Описание данных	Clean Data/ Очистка данных	Generate Test Design/ Подготовка плана тестирования	Review Process/ Оценка процесса	Plan Monitoring and Maintenance/ Планирование мониторинга и поддержки
Determine Data Mining Goals/ Определение целей аналитики	Explore Data/ Изучение данных	Construct Data/ Генерация данных	Build Model/ Обучение моделей	Determine Next Steps/ Определение следующих шагов	Produce Final Report/ Подготовка отчета
Produce Project Plan/ Подготовка плана проекта	Verify Data Quality/ Проверка качества данных	Integrate Data/ Интеграция данных	Assess Model/ Оценка качества моделей		Review Project/ Ревью проекта
		Format Data/ Форматирование данных			

Кейс

<https://colab.research.google.com/drive/16hTmqg1PIsQBsa9P8m7nr5zBfbKpL3pt?usp=sharing>

Итоги

Итоги

1. Узнали что такое методология Agile, и почему скорее всего Вы будете работать в Agile-команде
2. Познакомились с методологией CrispDM. Поговорили про основные этапы работы над ds-задачей
3. Посмотрели jupyter-ноутбук с анализом данных по методологии CrispDM

Домашнее задание

Домашнее задание

1. Возьмите задачу с винами (<https://www.kaggle.com/rajyellow46/wine-quality>) и решите ее, оформив в виде CrispDM-подхода
 - a. Решение - jupyter notebook на github или colab
2. * Только для тех у кого уже диплом
 - a. Оформите задачи по дипломной работе в виде этапов CrispDM (например, в trello)
 - b. Пришлите скриншот



**Спасибо за
внимание**