



МЕТОДЫ И СРЕДСТВА ИСИТ 2024



ЛЕКЦИЯ 10

Инструменты реализации проекта ИС и ИТ. Гибкая модель

ФРЕЙМВОРК AGILE МОДЕЛИ - SCRUM



ПРИНЦИП ФУНКЦИОНИРОВАНИЯ SCRUM-ФРЕЙМВОРКА

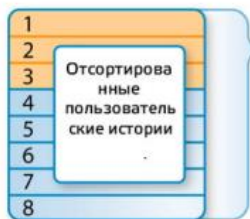
Требования от стейкхолдеров
(заинтересованных лиц)
заказчиков, пользователей...



Владелец продукта



Команда



Список требований



Команда выбирает столько задач, сколько сможет выполнить за Спринт

Встреча по планированию Спринта



Разложение на задачи
Формирование списка задач



Скрам Мастер

1-4 нед Спринты

Каждые 24 часа



Daily Scrum Meeting



Презентация спринта



Законченная работа



Ретроспектива Спринта

ДЕЙСТВУЮЩИЕ ЛИЦА SCRUM. ВЛАДЕЛЕЦ ПРОДУКТА

Единолично несет ответственность за принятие решений:

- какой функционал разрабатывать
- в каком порядке его разрабатывать

Обеспечивает четкое виденье продукта.

Отвечает за общий успех разрабатываемого или поддерживаемого решения.

ДЕЙСТВУЮЩИЕ ЛИЦА SCRUM. SCRUM МАСТЕР

Помогает понять и использовать по максимуму ценности, принципы и практики SCRUM.

Выступает в качестве тренера, лидера (но не менеджера)

- помогает организовать процесс управления изменениями
- помогает решать возникающие проблемы
- защищает команду от внешнего вмешательства

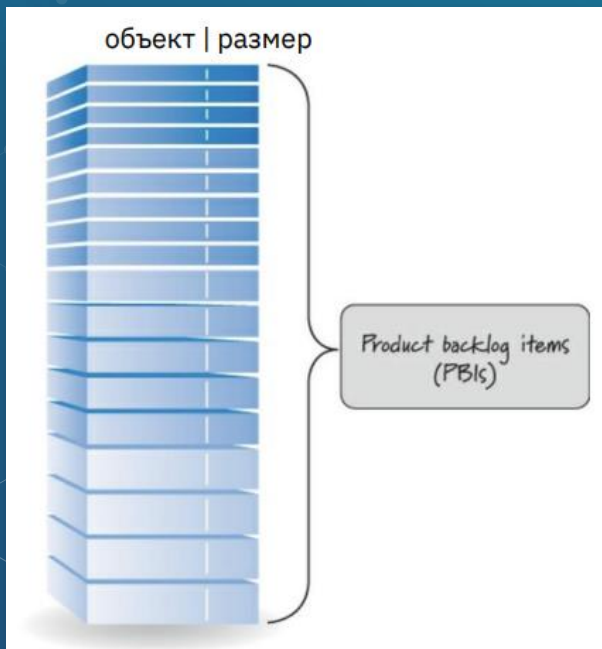
Не имеет полномочий контроля над командой.

ДЕЙСТВУЮЩИЕ ЛИЦА SCRUM. КОМАНДА РАЗРАБОТКИ

Состоящая из 5-9 человек, является самоорганизованной и кросс-функциональной. Разработчики объединяют роли архитектора, программиста, тестировщика, администратора баз данных, дизайнера пользовательских интерфейсов и т.д..

Отвечают за проектирование, создание и тестирование программного продукта.

ИНСТРУМЕНТ SCRUM - BACKLOG



- функциональное требование
- изменение
- дефект
- техническое улучшение
- приобретение знаний

Функциональные требования в SCRUM оформляются как ***Пользовательские истории (user stories)***

СТРУКТУРА ПОЛЬЗОВАТЕЛЬСКОЙ ИСТОРИИ

Как... (кто заинтересован в функционале)

Я хочу... (что представляет функционал)

Для того,... (почему это важно сделать)

ПОЛЬЗОВАТЕЛЬСКАЯ ИСТОРИЯ: ХОРОШАЯ И НЕ ОЧЕНЬ

Как начальник, я хочу установить умные датчики на узлы переработки, для того чтобы компания использовала умные датчики

Как инженер, я хочу автоматический сбор показаний на узлах переработки, для того чтобы исключить человеческий фактор

РВИ: ИЗМЕНЕНИЕ

Изменение: как представитель службы поддержки клиентов, я хочу, чтобы по умолчанию результаты поиска были по фамилии, а не по номерам заявок, чтобы было легче найти заявку адресованную к техподдержке.



РВИ: ДЕФЕКТ

Дефект: исправить дефект #UF256 в системе отслеживания дефектов, для того, чтобы спец-символы в поисковых запросах пользователей не приводили к системному сбою.

РВИ: ТЕХНИЧЕСКОЕ УЛУЧШЕНИЕ

Техническое улучшение: как разработчик, я хочу произвести миграцию системы для поддержки работы с последней версией базы данных Oracle, таким образом, мы избежим последствий работы с версией БД, которая в ближайшем времени будет снята с поддержки.

РВИ: ПРИОБРЕТЕНИЕ ЗНАНИЙ

Приобретение знаний: как архитектор, я хочу создать прототип системы для проверки двух архитектурных решений, а также запустить нагрузочное тестирование, для того чтобы понять, какое решение будет справляться лучше.

ХОРОШИЙ ВАСКЛОГ: ПРАВИЛЬНАЯ ДЕТАЛИЗАЦИЯ

Не все объекты в списке требований находятся одновременно на одном уровне детализации



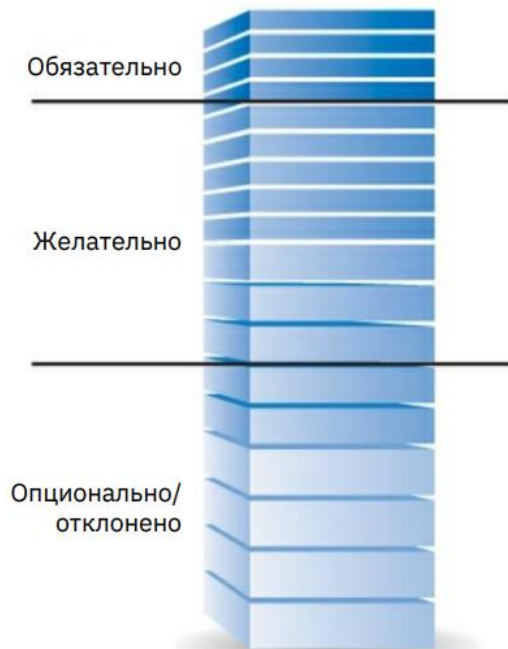
ХОРОШИЙ BACKLOG: РАССТАНОВКА ПРИОРИТЕТОВ

Обязательно – критично для разработки

Желательно – важно, но не критично для разработки

Опционально – наименее критичные объекты

Отклонено – объекты которые не будут разрабатываться

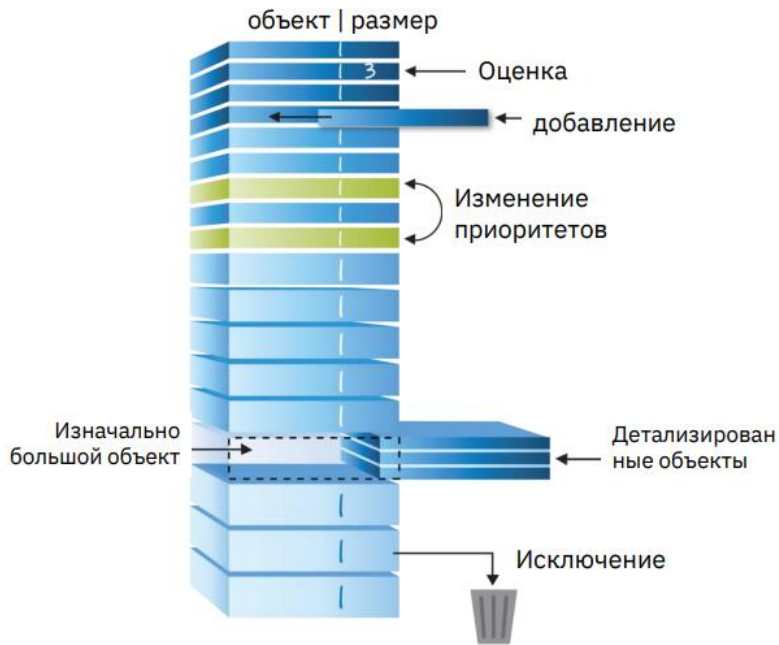


ХОРОШИЙ ВАСКЛОГ: ПОСТОЯННОЕ РАЗВИТИЕ

Список требований должен быть всегда актуальным.

Необходимо планомерно проводить:

- Переоценку объектов
- Обновление приоритетов
- Добавление новых объектов
- Детализацию объектов подходящих к выполнению



ЕДИНИЦЫ ИЗМЕРЕНИЯ РВИ

Story Points (стори поинты) – единица измерения, которая комбинирует фактор сложности и размер элемента списка требований. Стори поинты комбинируют фактор сложности и размер элемента списка требования. Позволяют абстрагироваться от профессионализма разработчика

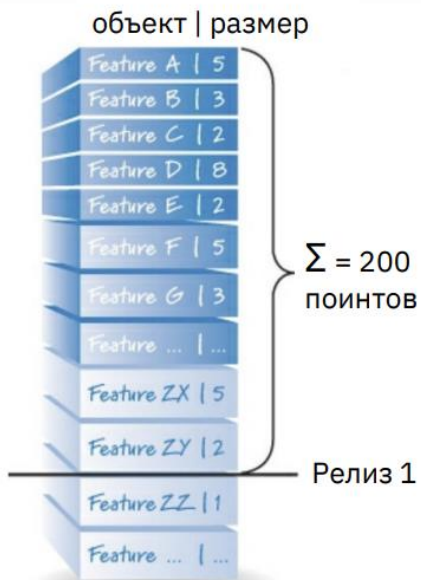
Большой ≠ сложный и маленький ≠ легкий

Ideal Days (человеко-часы) – отображают количество часов необходимых для выполнения задачи. Нет абстрагирования от профессионализма разработчика.

Человеко - часы ≠ затраченное время

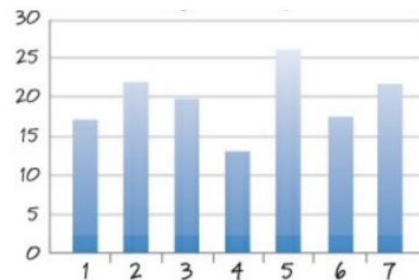
КЛЮЧЕВАЯ МЕТРИКА SCRUM: TEAM VELOCITY

Спрогнозированный размер ÷ измеренная продуктивность команды = количество итераций



200 поинтов ÷ 20 поинтов/итерация = 10 итераций

Средняя продуктивность* = 20



*team velocity – разгон команды

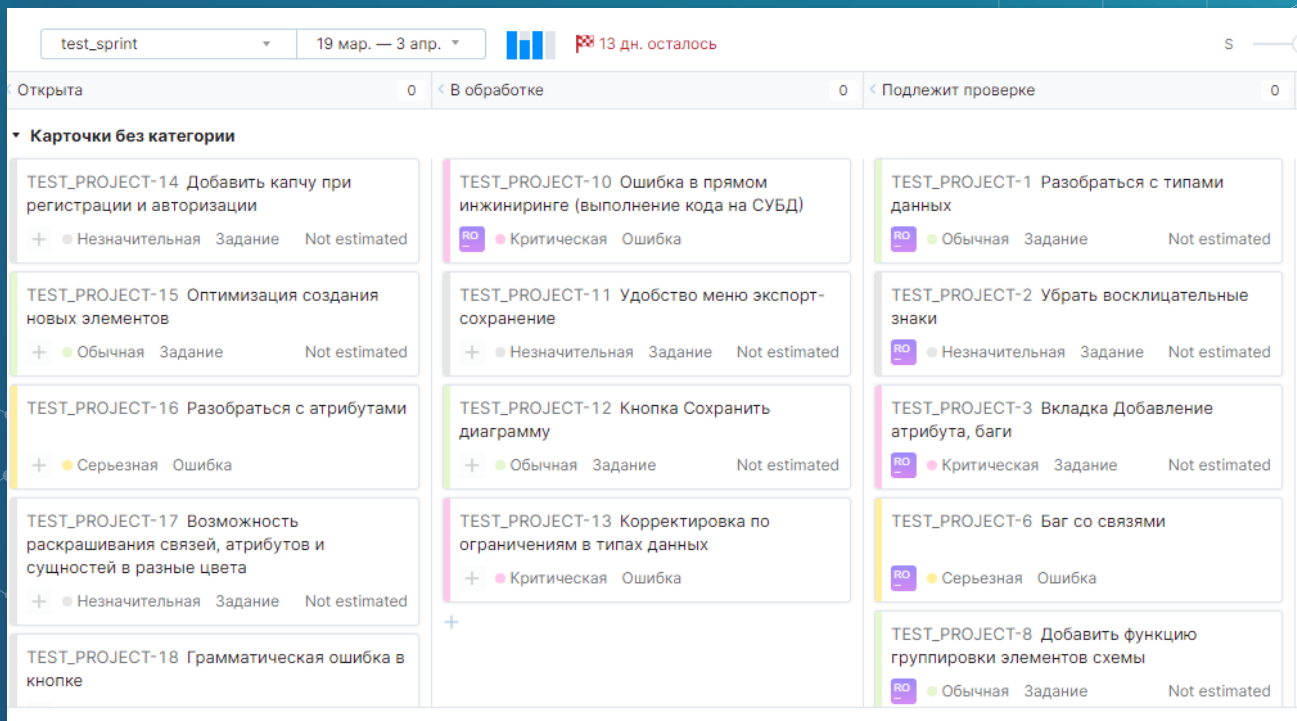
ИНСТРУМЕНТ SCRUM – ДОСКА SCRUM. ДЭШБОРД

My Dashboard ▾

#issues sort by: updated ¹⁹

- O TEST_PROJECT-19 Настройки шрифтов и цвета
- H TEST_PROJECT-18 Грамматическая ошибка в кнопке
- H TEST_PROJECT-17 Возможность раскрашивания связей, атрибутов и сущностей в разные цвета
- C TEST_PROJECT-16 Разобраться с атрибутами
- O TEST_PROJECT-15 Оптимизация создания новых элементов
- H TEST_PROJECT-14 Добавить капчу при регистрации и авторизации
- K TEST_PROJECT-13 Корректировка по ограничениям в типах данных
- O TEST_PROJECT-12 Кнопка Сохранить диаграмму
- H TEST_PROJECT-11 Удобство меню экспорт-сохранение
- K TEST_PROJECT-10 Ошибка в прямом инжиниринге (выполнение кода на СУБД)
- K TEST_PROJECT-9 Баги сущностей
- O TEST_PROJECT-8 Добавить функцию группировки элементов схемы
- C TEST_PROJECT-6 Баг со связями
- K TEST_PROJECT-3 Вкладка Добавление атрибута, баги

ИНСТРУМЕНТ SCRUM – ДОСКА SCRUM. СПРИНТ



ИНСТРУМЕНТ SCRUM – ДОСКА KANBAN

- группы поддержки программного обеспечения, где не важен «план», но важна скорость реагирования на изменения;
- группы тестирования, работающие отдельно от групп разработки;
- службы поддержки;
- стартапы без четкого плана результата проекта.

ИНСТРУМЕНТ SCRUM – ДОСКА KANBAN. СПРИНТ

0.01 ... Доработка ... 1.01 ...

+

- ✓ Масштабирование рабочей области
- ✓ Переписать обработчик на проверку названия атрибута
- ✓ Разобраться с типами данных 2
- ✓ Убрать восклицательные знаки 2

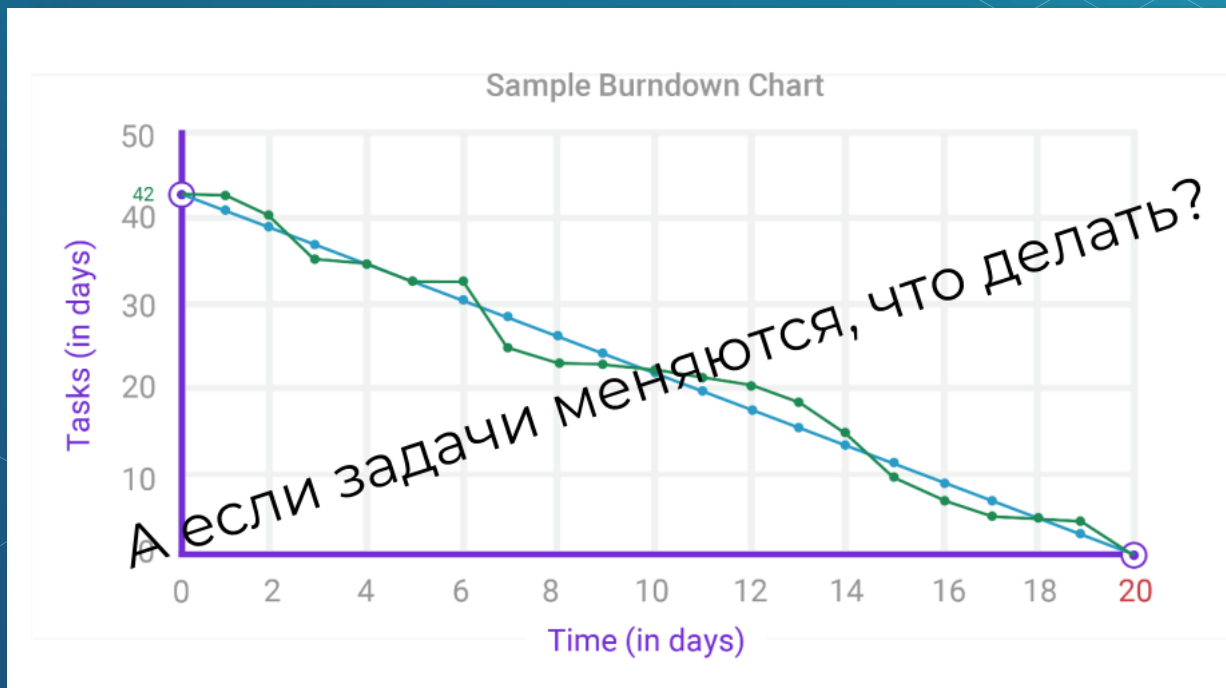
+

- ✓ Удобство меню экспорт-сохранение 6
- ✓ Кнопка Сохранить диаграмму 1
- ✓ Корректировка по ограничениям в типах данных

+

- Добавить капчу при регистрации и авторизации
- Оптимизация создания новых элементов
- Разобраться с атрибутами 2
- Возможность раскрашивания связей, атрибутов и сущностей в разные цвета

ИНСТРУМЕНТ SCRUM – ДИАГРАММА СГОРАНИЯ



УНИВЕРСАЛЬНЫЙ ИНСТРУМЕНТАРИЙ ПРОЕКТА РАЗРАБОТКИ - GIT

Набор консольных утилит, которые отслеживают изменения в файлах (репозитории, коллекции патчей и т.д.).

Благодаря git можно выбрать любую сохраненную версию проекта ПО, осуществлять слияние и разделение версий файлов, визуализировать историю разработки.

Проекты, использующие git: ядро Linux, Swift, Android, jQuery, PHP, QT и другие.

git существует для LINUX, Windows, OS X

ЛОКАЛЬНЫЙ РЕПОЗИТОРИЙ GIT

1. Локально, в папке проекта создается репозиторий git.
2. В репозиторий выкладываются файлы, части файлов или даже отдельные строки файлов, которые необходимо отслеживать.
3. Размещенные в репозитории файлы подлежат коммиту (commit) – снимок или состояние файла в определенном моменте времени. Очень желательно при коммите оставлять подробные комментарии по существу.

ЛОКАЛЬНЫЙ РЕПОЗИТОРИЙ GIT

```
git config --global user.name "My Name"  
git config --global user.email myEmail@example.com
```

```
$ mkdir Desktop/git_exercise/  
$ cd Desktop/git_exercise/  
$ git init
```

```
$ git add -A
```

```
$ git commit -m "Initial commit."
```



УДАЛЕННЫЙ РЕПОЗИТОРИЙ GIT

1. Подключение к удаленному репозиторию (возможно применение ssh и других методов обеспечения безопасного подключения).
2. Отправка изменений файлов в репозиторий.
3. Запрос изменений с сервера.

УДАЛЕННЫЙ РЕПОЗИТОРИЙ GIT

```
$ git remote add origin https://github.com/tutorialzine/awesome-  
project.git
```

```
$ git push origin master Counting objects: 3, done. Writing objects: 100%  
(3/3), 212 bytes | 0 bytes/s, done.
```

```
$ git pull origin master
```

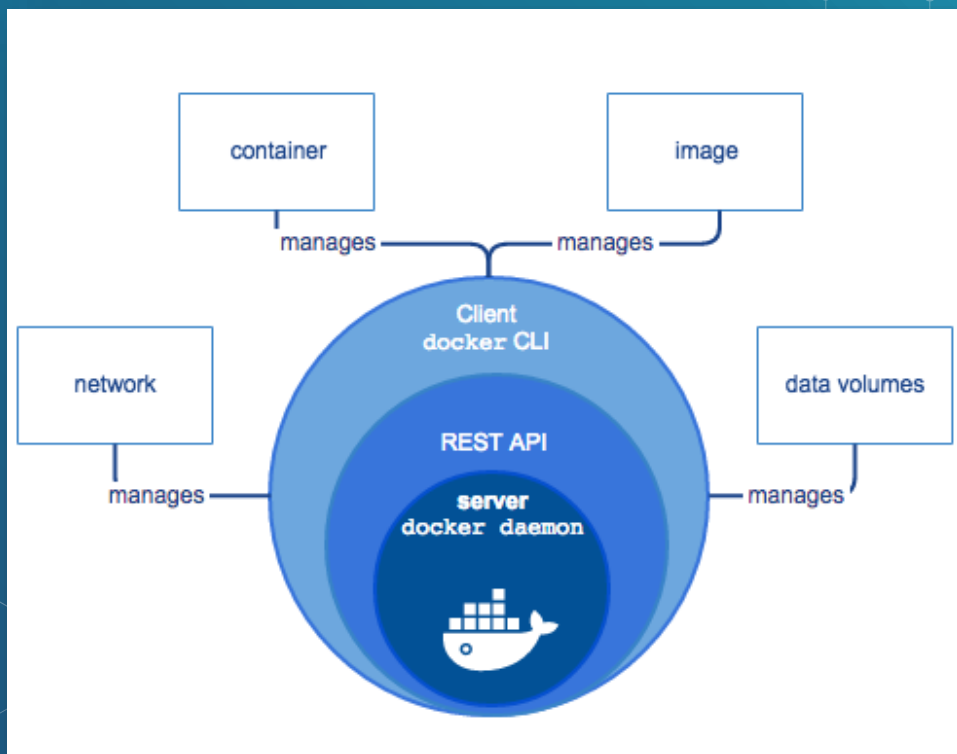
УНИВЕРСАЛЬНЫЙ ИНСТРУМЕНТАРИЙ ПРОЕКТА РАЗРАБОТКИ - DOCKER

Программное обеспечение для автоматизации процессов развертывания и управления приложениями в различных средах с поддержкой контейнеризации.

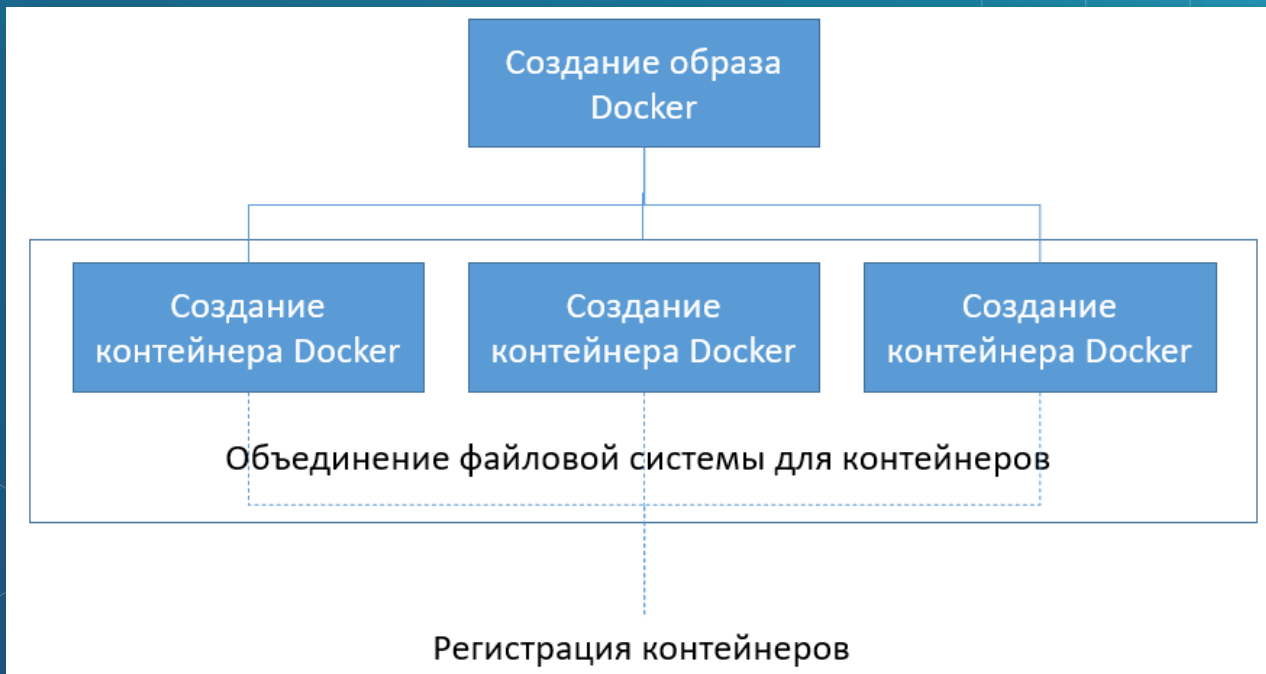
Приложение, со всем окружением “упаковывается” в контейнер, который может быть перенесен в любую LINUX-систему.

Docker существует для LINUX, Windows, OS X

КОМПОНЕНТЫ DOCKER



ПРИНЦИП РАБОТЫ DOCKER



САМОСТОЯТЕЛЬНОЕ ЗАДАНИЕ (ТЕСТЫ)

- ◆ Выполнить Тест 5. Дедлайн – 30 апреля.
- ◆ Google Classroom для групп БСБО-04,05,06,07,08,09: enkfxui.
- ◆ Google Classroom для групп БСБО-10,11,12: wdqcim2.
- ◆ Google Classroom для групп БСБО-1,2,3,13,14: zrhmuks.
- ◆ Обратите внимание на то, чтобы в Google Classroom ваш псевдоним содержал вашу фамилию кириллицей (в крайнем случае - латиницей).

САМОСТОЯТЕЛЬНОЕ ЗАДАНИЕ (КНИГИ И ЗАДАНИЕ)

- ◆ По учебнику Технологии разработки ПО: Глава 2, Руководство программным проектом, стр. 49-72, <http://msuniversity.ru/d/13>
- ◆ Придумайте и кратко опишите кейс использования виртуальной машины в процессе разработки-развертывания ПО.
- ◆ Ответьте письменно на следующий вопрос: Какие еще, кроме диаграммы сгорания задач, графические инструменты используются в SCRUM-досках?

СПАСИБО!

ВАШИ ВОПРОСЫ,
ПОЖАЛУЙСТА?

