



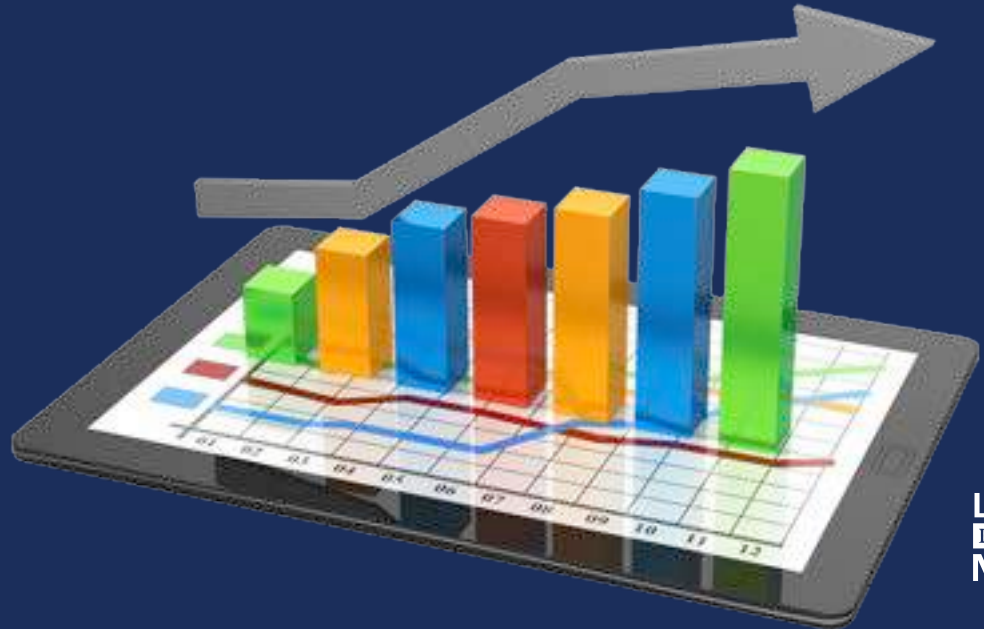
2015

CEE-SEC(R)

Разработка ПО

Метрики, которые приносят пользу

Светлана Мухина



Знакомимся



Светлана Мухина

ICAgile ICP, ICP-ATF, ICP-BVA, PSM I, CSPO

Agile and Career Coach

Опыт работы: 13+ лет в IT, руководитель проектов, компьютерный лингвист, тех. писатель, бизнес-аналитик

Интересы: управление проектами, процессные трансформации, коучинг, психология

Хобби: верховая езда, музыка, покер, путешествия

LinkedIn  <https://www.linkedin.com/in/svetlanamukhina>

Какие метрики мы собираем на проектах?

- ✓ **Capacity** – количество идеальных часов, доступное в следующем спринте
- ✓ **Velocity** – количество стори поинтов (work units), выполненных за прошлый спринт(ы)
- ✓ **Requirements stability index** – процент измененных требований в текущем спринте
- ✓ **Burn-down chart** – визуальная презентация сделанной и оставшейся работы
- ✓ **Time-tracking** - Зачем заполнять систему учета времени?



Capacity

Capacity *прогноз* - количество идеальных часов, доступное в следующем спринте

- ✓ Понять, сколько часов у нас есть на работу: на написание кода, тестирование, т.д.
 - Как правило участник проекта работает не более 5ти часов в день
- ✓ Эффективно распределить задачи
 - Нет смысла планировать задачи на тех, кто будет в отпуске или занят другими активностями
 - Мало пользы принесет технический анализ задачи, выполненный участником проекта, который в следующем спринте будет отсутствовать
- ✓ Аккуратно и точно планировать
 - Мы оцениваем задачи в часах и берем в спринт столько, сколько соответствует нашей capacity

Green Team	Capacity	R14.4 Sprint 1										General Quantity of hours	Meetings time, hours	General quantity of hours - Meeting time	Load factor	Quantity of hours *Load Factor	Minus General Buffer for team, 15%	Average time spent per person
Name	22-May	23-May	####	####	####	####	####	2-Jun	3-Jun	4-Jun	General Quantity of hours	Meetings time, hours	General quantity of hours - Meeting time	Load factor	Quantity of hours *Load Factor	Minus General Buffer for team, 15%	Average time spent per person	
Dev1	1	1	1	1	1	1	1	1	1	X	72	7.5	64.5	0.4	25.8	21.93	2.44	
Dev2	1	1	1	1	1	1	1	1	1	X	72	7.5	64.5	0.75	48.375	41.11875	4.57	
Dev3	1	1	1	1	1	1	1	1	1	X	72	7.5	64.5	0.75	48.375	41.11875	4.57	
Dev4	1	1	1	1	1	1	1	1	1	X	72	7.5	64.5	0.1	6.45	5.4825	0.61	
Dev5	1	1	1	1	1	1	1	1	1	X	72	7.5	64.5	0.1	6.45	115.1325	0.61	
Dev6	1	1	1	1	1	1	1	1	1	X	72	7.5	64.5	0.4	25.8	21.93	2.44	
											432	45	387	2.5	161.25	137.0625	Developers capacity	
QA1	0	0	1	1	1	1	1	1	1	1	64	8	56	0.65	36.4	30.94	3.87	
QA2	1	1	1	1	1	1	1	1	1	1	80	10	70	0.5	35	29.75	2.98	
											144	18	126	1.15	71.4	60.69	QA capacity	
											576	63	513	3.65	232.65	197.7525		
Sum of burnt time per day, the whole team	18.20	18.20	22.07	22.07	22.07	22.07	22.07	22.07	22.07	6.84								
Sum of burnt time per day, the dev team	15.23	15.23	15.23	15.23	15.23	15.23	15.23	15.23	15.23	0.00								

1 is present at work&usual day of sprint

0 absent at work (Day-off or whole day training)

X last day of sprint - is not included in capacity for developers, for auto-test writing

Идеальный час и load factor

- изучаю требования
- обсуждаю требования с бизнес-аналитиком
- планирую работу
- вношу изменения в БД
- занимался серверной частью
- занимаюсь интерфейсом
- проверяю возможность интеграции с другой системой
- запускаю сборку проекта
- тестирую
- исправляю дефекты

Включено в идеальный час

- создаю юнит тесты
- запускаю юнит тесты
- исправляю дефекты после юнит тестов
- готовлю тестовые данные
- объединяю изменения кода (merge)
- создаю и проверяю процедуры деплоймента
- деплою на сервер
- обновляю информацию Jira
- показываю функциональность тестировщику
- исправляю дефекты
- делюсь знаниями о приложении
- занимаюсь наставничеством

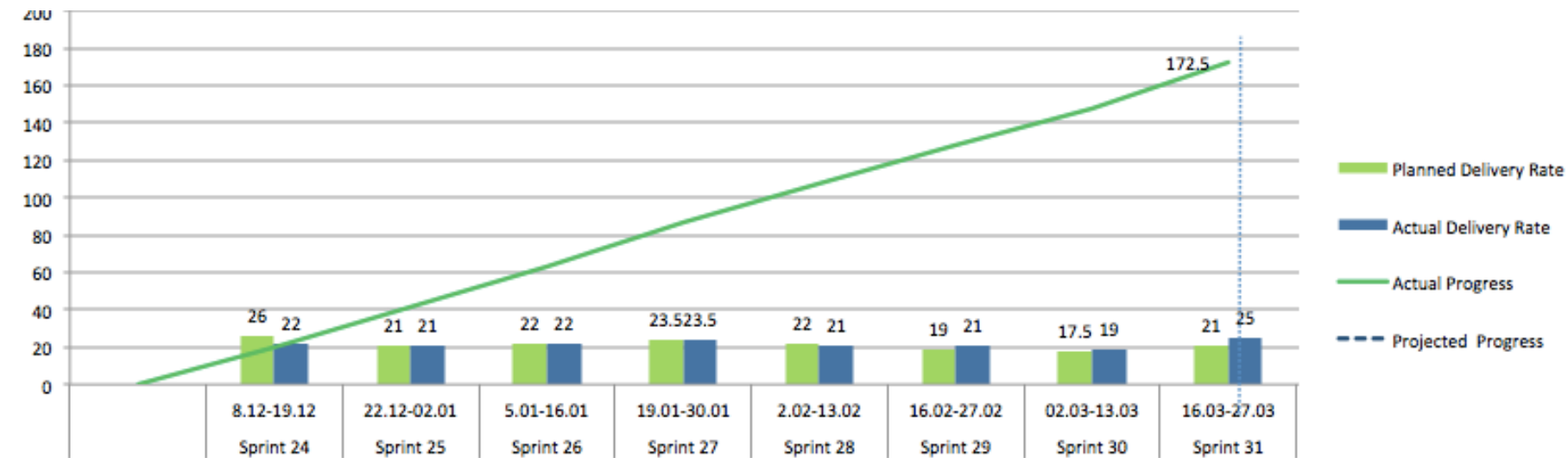
Включено в load factor

Velocity

Velocity *Опыт* — количество стори поинтов, выполненное в прошлый спринт(ы)

- ✓ Знать производительность, чтобы понимать возможности ее улучшения на уровне команды и индивидуально
- ✓ Формировать объем задач спринта основываясь на исторических данных прошлых спринтов
- ✓ Уменьшать давление менеджмента или/и владельца продукта в случае включения в спринт дополнительной работы
- ✓ Осознать наличие технического долга;
 - Технический долг не включается в задачи, но на него тратится время
- ✧ Совместное использование velocity и capacity позволяет более точно планировать исходя из предыдущего опыта и предположений о распределении времени в будущем

Визуализация velocity



Индекс стабильности требований

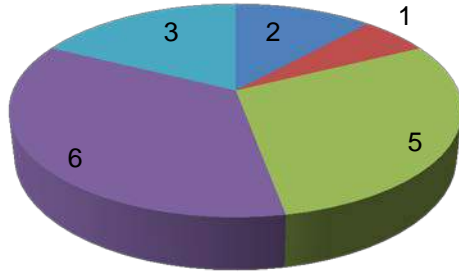
ИСТ – процент измененных требований во время спринта

- ✓ Понять сколько времени мы тратим на переделывание задач (re-work)
- ✓ Показать эти данные по времени менеджменту, или владельцу продукта, бизнес аналитику
 - Это может помочь стабилизировать требования в спринте
 - Это может убедить владельца продукта продумывать требования заранее

Учет рабочего времени

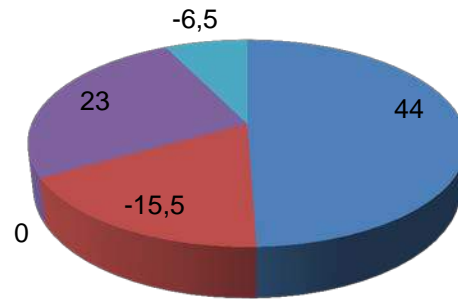
- ✓ Понять какие задачи чаще всего недооцениваются
 - Принести эту информацию на ретро или сессию работы над ошибками
 - Используя этот подход мы обнаружили, что постоянно недооцениваем UI задачи
 - Мы так же увидели, что работа на виртуальных машинах увеличивает время выполнения задачи от 20 до 30 %
 - Мы смогли показать, что нам не хватает времени на мануальное тестирование
- ✓ Определить как часто задачи переоткрываются и подумать, почему так происходит
 - Это помогло нам узнать о пользе sanity тестирования
- ✓ Узнать производительность каждого участника команды
 - Настольный теннис – это не тоже самое, что написание кода или тестирование

Count



- Underestimate (delta >= 10 h)
- Overestimate (delta <= -10 h)
- Perfect estimate
- Small underestimate (0 < delta < 10)
- Small overestimate (-10 < delta < 0)

Hours



- Underestimate (delta >= 10 h)
- Overestimate (delta <= -10 h)
- Perfect estimate
- Small underestimate (0 < delta < 10)
- Small overestimate (-10 < delta < 0)

Key	Issue Type	Status	Priority	Original Estimate, hours	Remaining Estimate, hours	Time Spent, hours	Original Estimate vs Spent
WL-530	Task	Resolved	Medium	0	0	34	34
LLP-418	Sub-task	Resolved	Medium	8	5	3	-5
LLP-417	Sub-task	Resolved	Medium	4	0	6	2
LLP-415	Sub-task	Resolved	Major	8	0	8	0
LLP-414	Defect	Resolved	Major	2.6666667	0	6.666667	4
LLP-412	Story	Resolved	Medium	4	0	4	0
LLP-408	Story	Resolved	Medium	7	0	17	10
LLP-407	Story	Resolved	Medium	9	1	8	-1
LLP-406	Story	Open	Medium	24	17	7	
LLP-405	Story	Resolved	Medium	10	0	13	3
LLP-404	Defect	Resolved	Critical	8.5	0	14.5	6
LLP-403	Story	Resolved	Medium	16	1	16	0
LLP-402	Story	Resolved	Medium	7	0	10	3
LLP-401	Defect	Resolved	Major	32	0	16.5	-15.5
LLP-397	Story	Resolved	Minor	2	0	2	0
LLP-396	Defect	Resolved	Medium	2	0	2	0
LLP-409	Story	Resolved	Medium	3.5	0.5	3	-0.5
LLP-410	Story	Resolved	Medium	5	0	10	5

	original estimate equals to 0 or is empty; time spent equals to 0 or is empty;
	difference between Original estimate and Spent which shows that issue was completed in time or earlier (based on estimate)
	time spent on issues exceeds original estimate value
	still not finished issues

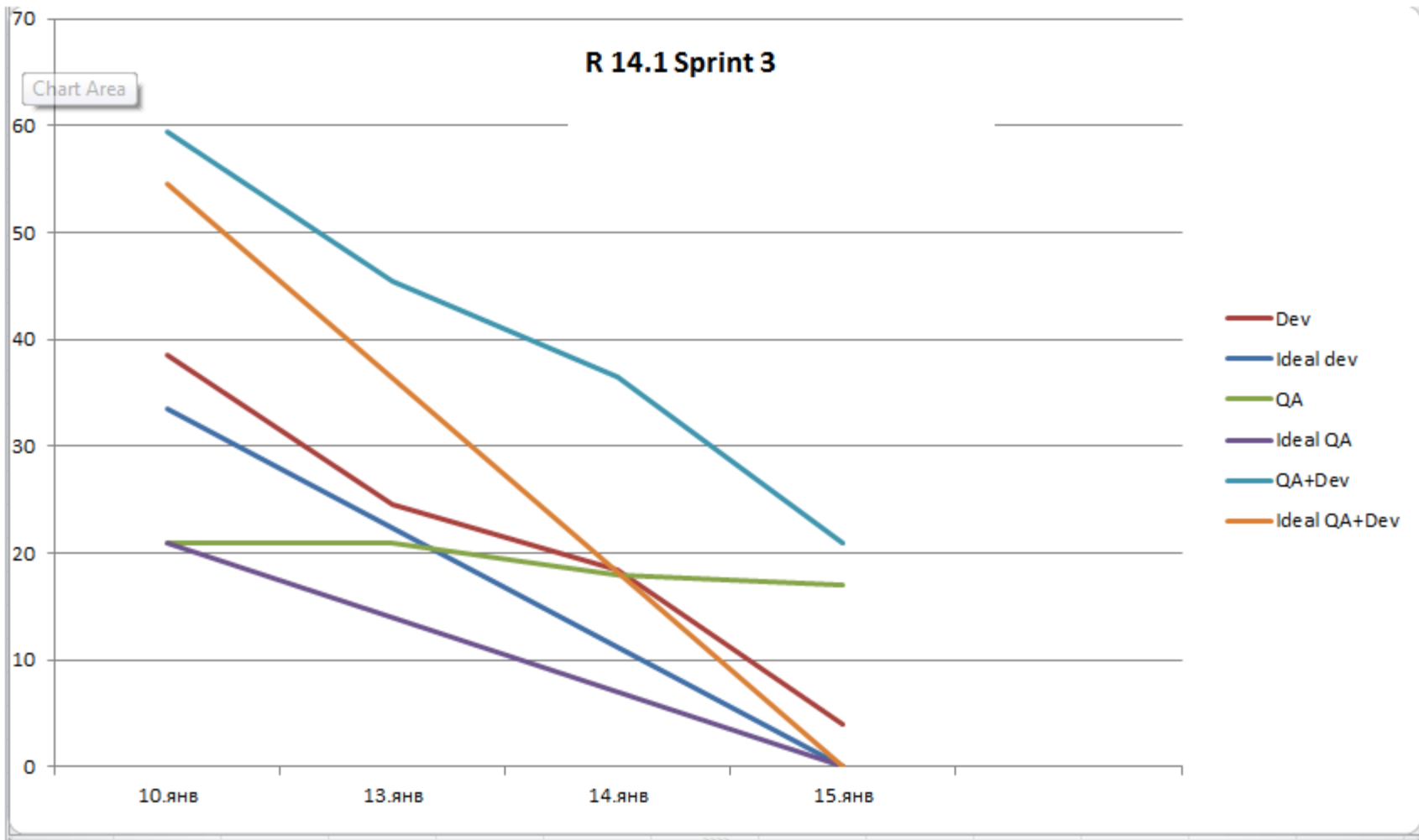
График burndown

Burn-down – это визуальное представление на линии времени проделанной работы

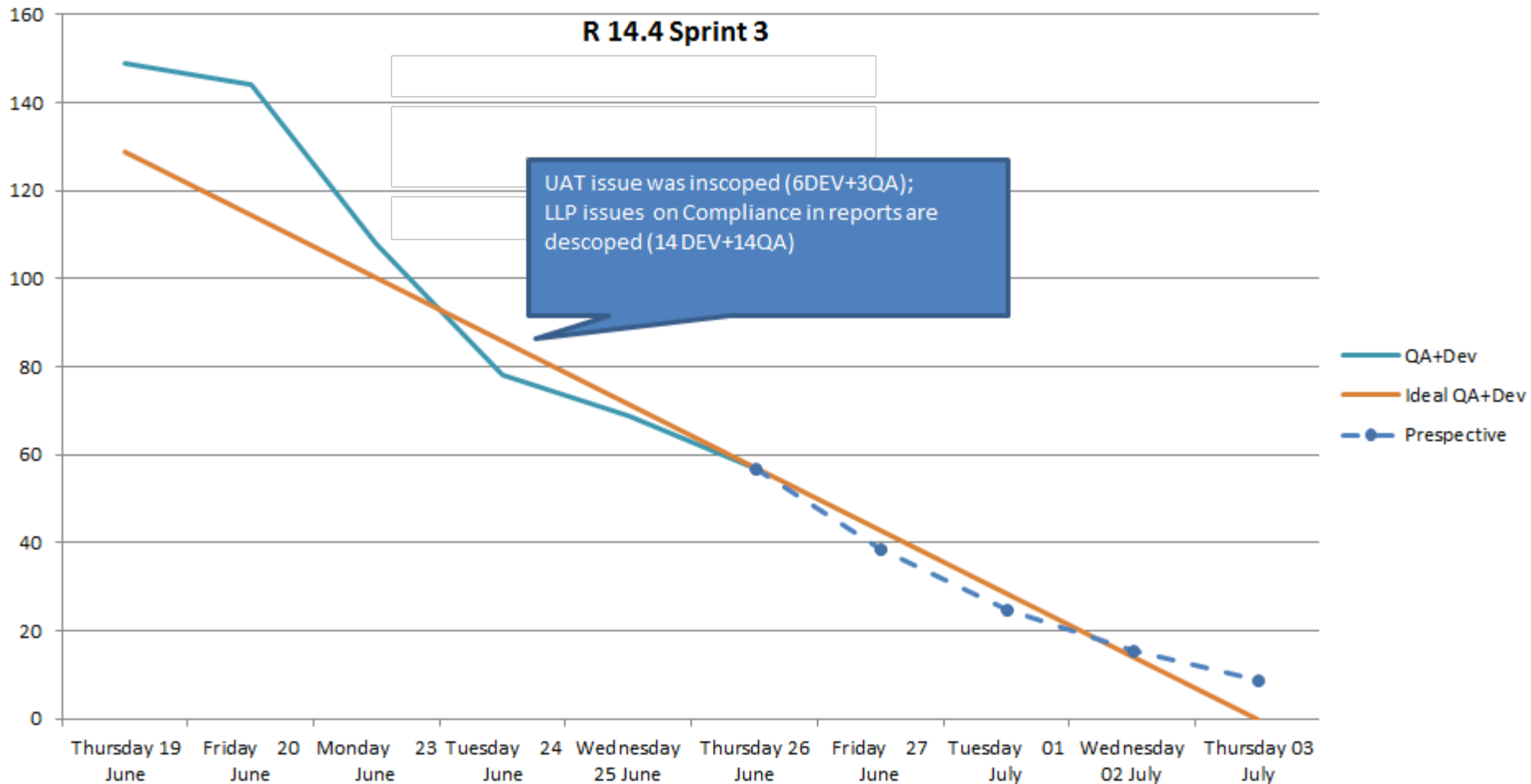
- ✓ Прогнозировать выполнение задач спринта в оговоренные сроки
- ✓ Визуализировать добавление дополнительной работы в спринт
- ✓ Сфокусировать команду на выполнение всего объема работ спринта, а не только индивидуальных задач
 - Рисуйте burn-down всей командой, это повышает уровень вовлеченности и ответственность
- ✓ Определить и уменьшить риски, когда на это еще есть время

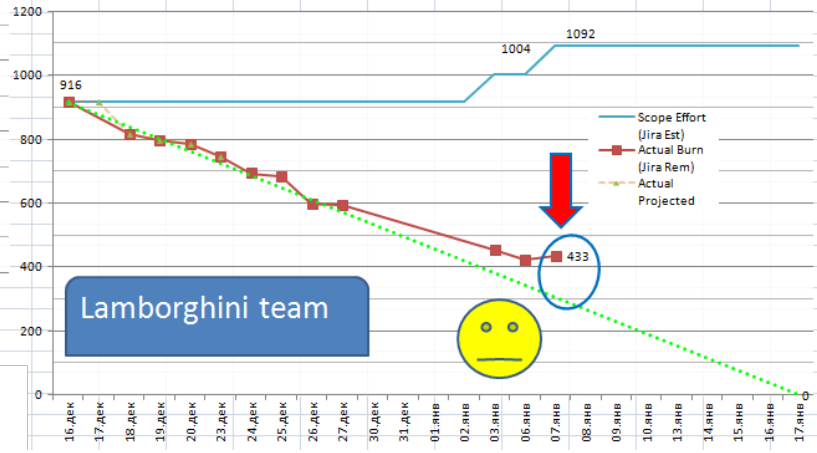
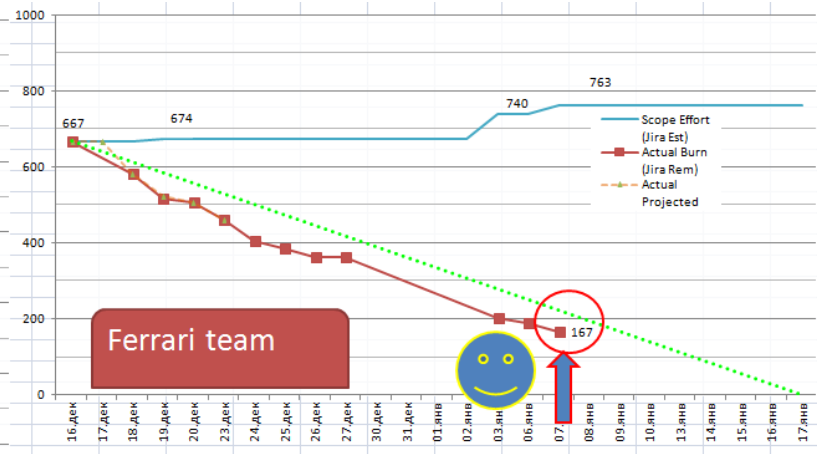
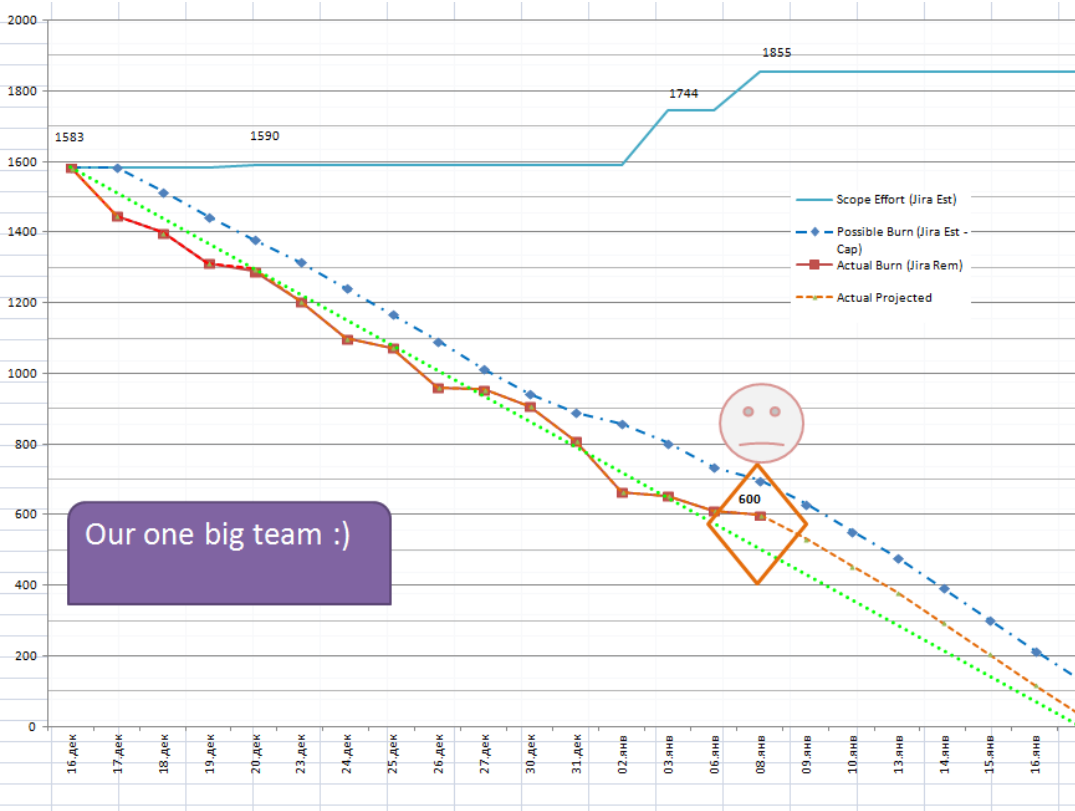
R 14.1 Sprint 3

Chart Area



R 14.4 Sprint 3



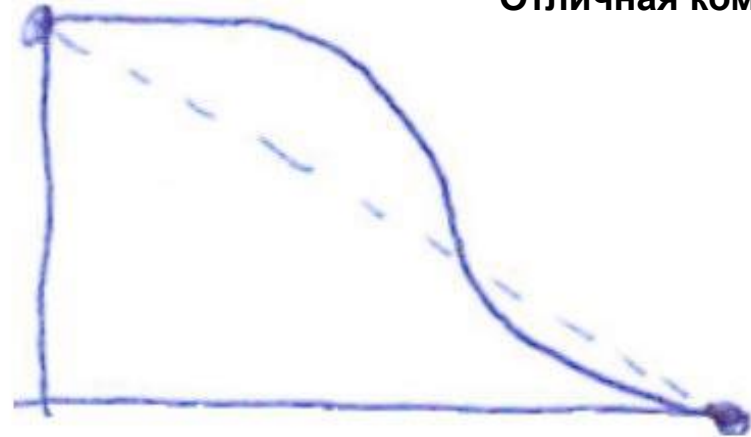


Идеальная команда



- ✓ Берем в спринт ровно столько, сколько сможем сделать
- ✓ Завершаем вовремя
- ✓ Вкладываемся в свои оценки
- Улучшений не требуется

Отличная команда



- ✓ Завершаем вовремя
- ✓ Адаптируемся в процессе работы
- ✓ В конце спринта есть время на доп. работу
- Обсудить причины медленного прогресса в начале спринта
- Поэкспериментировать с capacity

Обычная команда



- ✓ Вкладываемся в сроки
- ✓ Сумели “поднажать” и успели завершить работу
- ✓ Анализируем свой прогресс

- Договорится незамедлительно обсуждать снижение производительности
- Выработать процедуру удаления из скоупа спринта низкоприоритетных задач

Давайте отдохнем



- ✓ Взяли задач меньше, чем можем сделать
- ✓ Владелец продукта не предоставил готовых для работы требований
- ✓ Переоценили сложность задач

- Определить заранее способ реагирования на недостаточное кол-во задач
- Попросить предоставить больше задач
- Заняться исследованием требований на следующие спринты

Упс, опоздали



- ✓ Не успели выполнить запланированную работу
- ✓ Не завершили спринт
- ✓ В течение спринта не соотносили объем оставшихся задач с производительностью
- Перенести незавершенные задачи в список общих требований
- Снизить capacity на следующий спринт
- Предпринять какие-то действия после 2-3 дней медленного прогресса

Ай да молодцы

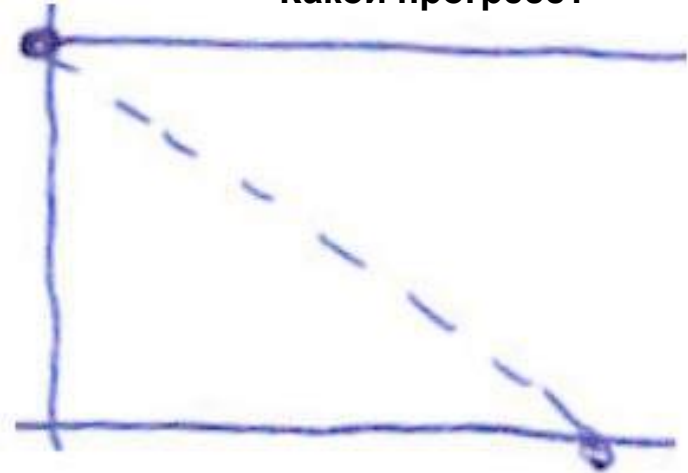


- ✓ Завершили работу раньше, чем планировали
- ✓ Не добавили дополнительных задач в спринт
- ✓ Задачи были переоценены
- ✓ Неверные данные по capacity или/и velocity
- Добавить в спринт дополнительных готовых задач

Ой, менеджмент идет



Какой прогресс?



- ✓ Прогресс не измерялся
- ✓ Добавлялись новые задачи в таком же объеме, в котором команда выполняла уже запланированные
- ✓ Были сложности с определением сроков завершения работ по спринту
- Объяснить команде для чего необходимо ежедневно измерять прогресс
- Остановить работу после 2-3 дней спринта и обсудить с командой причины отсутствия прогресса

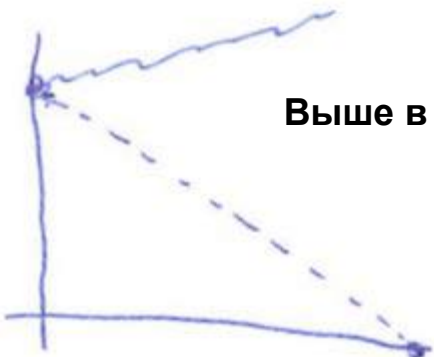
- ✓ Серьезные проблемы в команде
- ✓ Никто не занимается обучением команды
- ✓ Владелец продукта / менеджер не беспокоится о прогрессе и производительности
- Завершить спринт
- Объяснить команде цель проекта
- Обучайте команду

Нулевой эффект



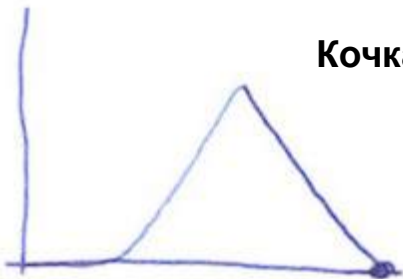
- ✓ Работа на спринт не была оценена
- ✓ Спринт еще не начался
- Организовать встречу по планированию
- Оценить работу на спринт
- Создать план работ на спринт
- Начать разработку функциональности

Выше в небо



- ✓ Обычно так выглядит первый спринт
- ✓ В спринт добавлялись новые задачи, а прогресс не измерялся
- ✓ Задачи постоянно переоценивались в большую сторону
- Еще раз оцените все задачи спринта
- Пригласите фасилитатора на это встречу
- Расскажите команду как оценивать задачи

Кочка на дороге



- ✓ Задачи были добавлены в спринт после начала спринта
- Начать спринт заново, уменьшив время спринта и используя метрики на сессии планирования спринта

Время для рефлексии

Что вы начнете делать из того, что услышали

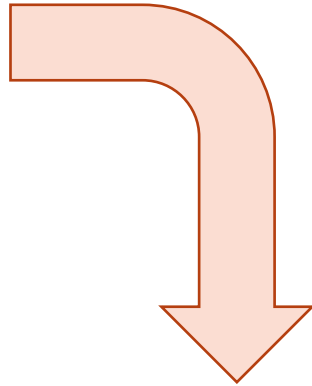
Что вы станете делать по-другому?

Что вы продолжите делать?

Что перестанете делать?



Презентация тут



<http://www.slideshare.net/SvetlanaMukhinalCAgi/ss-53946740>

Почитать и Посмотреть

- ✓ Алексей Жеглов – “Метрики для улучшения работы организации” - видео, презентация (рус)
- ✓ Как манипулировать диаграммами - статья (рус) <https://www.linkedin.com/pulse/как-манипулировать-диаграммами-svetlana>
- ✓ The Power on Visualization - вебинар (инг) <https://www.linkedin.com/pulse/20141202230246-48371619-follow-up-on-the-webinar-the-power-on-visualization>
- ✓ ICAgile Certified Professional - Agile Fundamentals –тренинг (рус) <https://icagile.com/icagile-certified-professional>
- ✓ ICAgile Certified Professional - Business Value Analysis – тренинг (рус) <https://icagile.com/icp-business-value-analysis>

Тренинги от Luxoft Agile Practice

- ✓ ICAgile Certified Professional - Agile Fundamentals
- ✓ ICAgile Certified Professional - Business Value Analysis
- ✓ ICAgile Certified Professional - Agile Team Facilitation
- ✓ Professional Scrum Master
- ✓ Management 3.0
- ✓ Workshops:
 - Coaching and self-coaching;
 - Agile project management;
 - Agile project metrics;
 - Motivation and team-building;
 - Effective communication;
 - etc.

❖ **Регистрации и информация**

<http://www.luxoft.com/agile/certified-trainings>





Спасибо!



LXFT
LISTED
NYSE