Scrum

Controlling Chaos

Scrum?!? What?!?

- **Scrum** is an **agile** project management framework for software development.
- Work is structured in cycles of work called sprints, iterations of work that are typically 2 - 4 weeks in duration.
- During each sprint, teams pull from a prioritized list of customer requirements (user stories) so that the features that are developed first are of the highest business value to the customer.
- At the end of each sprint, a **potentially shippable product** is delivered. This is very important!

Scrum for our purposes

- Throughout this lecture Scrum will be discussed as it is used in the workplace, but there will be points where this is incompatible with the structure of a class, so modifications to the framework are pointed out where appropriate.

What is this **Agile** thing you speak of?

A set of work methods and tools aimed at:

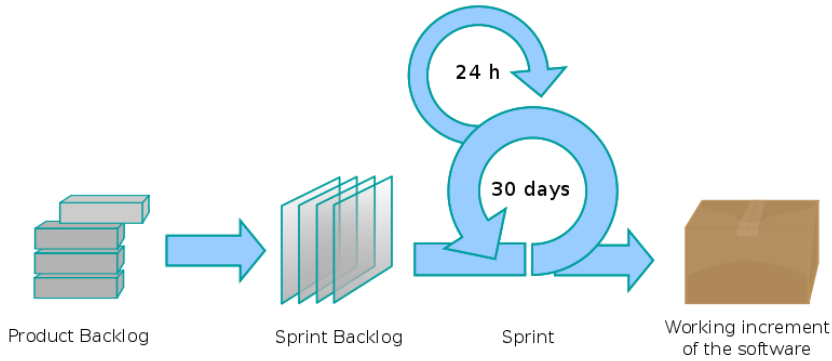
- Improving the ability to respond quickly to needs and requests from the market
- Cutting down waste and waiting periods
- Reducing employee stress while simultaneously increasing productivity

“Scrum”?

“Scrum”?

- A Scrum is a team pack in Rugby, where everybody in the pack acts together with everyone else to move the ball down the field.
- The reasons for this naming *cough*the meeting*cough* will become apparent as we progress

The Scrum Process



Scrum Roles

- Product Owner – represents the customers (stakeholders) and speaks from a business perspective
- Scrum Master – maintains scrum processes, main job is to remove impediments to the process
- Scrum Team – the developers, they organize the work themselves and lack a formal project manager, normally consists of 5 – 10 people, should be cross functional
- (Our teams will obviously be smaller, which is better)

Scrum Artifacts

- Artifact - a tangible byproduct produced during the development of software
  - Ex: class diagrams, uses cases, UML models
- The Product Backlog
- The Sprint Backlog
- The Burndown Chart

The Product Backlog

- The **product backlog** is a document prepared by the product owner that contains a list of customer requirements prioritized by business value.
- These requirements can and will frequently change.
- It should include all features visible to the customer, as well as the technical requirements needed to build the product

The Product Backlog (Example)

	Item #	Description	Est	By
Very High				
	1	**Finish database versioning**	16	KH
	2	**Get rid of unneeded shared Java in database**	8	KH
	-	**Add licensing**	-	-
	3	Concurrent user licensing	16	TG
	4	Demo / Eval licensing	16	TG
		Analysis Manager		
	5	File formats we support are out of date	160	TG
	6	Round-trip Analyses	250	MC
High				
	-	**Enforce unique names**	-	-
	7	In main application	24	KH
	8	In import	24	AM
	-	**Admin Program**	-	-
	9	Delete users	4	JM
	-	**Analysis Manager**	-	-
	10	When items are removed from an analysis, they should show up again in the pick list in lower 1/2 of the analysis tab	8	TG
	-	**Query**	-	-
	11	Support for wildcards when searching	16	T&A
	12	Sorting of number attributes to handle negative numbers	16	T&A
	13	Horizontal scrolling	12	T&A
	-	**Population Genetics**	-	-
	14	Frequency Manager	400	T&M
	15	Query Tool	400	T&M
	16	Additional Editors (which ones)	240	T&M
	17	Study Variable Manager	240	T&M
	18	Haplotypes	320	T&M
	19	**Add icons for v1.1 or 2.0**	-	-
	-	**Pedigree Manager**	-	-
	20	Validate Derived kindred	4	KH
Medium				
	-	**Explorer**	-	-
	21	Launch tab synchronization (only show queries/analyses for logged in users)	8	T&A
	22	Delete settings (?)	4	T&A

The Sprint Backlog

- The **sprint backlog** is a detailed document containing information about what requirements and *how* the team is going to implement these requirements for the upcoming sprint.
- Tasks are typically broken down into *hours* with no task being more than 16 hours. If a task is greater than 16 hours, it should be broken down further.

The Sprint Backlog (Example)

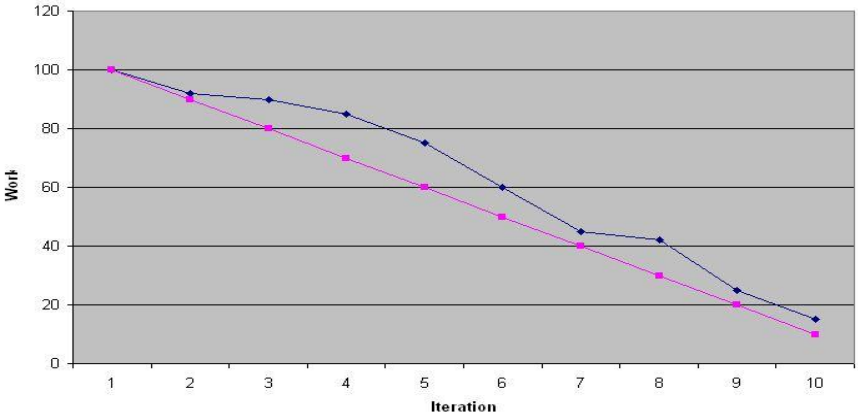
Tasks	Mon	Tues	Wed	Thurs	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

The Burndown Chart

- The Burndown Chart shows the cumulative work remaining in a Sprint and is updated on a day-to-day basis.
- This is used as a tool to guide the development team to successful completion of a Sprint on time with working final product.

The Burndown Chart (Diagram)

The straight line represents an ideal iteration where work is completed in a perfectly steady and evenly distributed manner. The more erratic line represents the work that is actually completed over time by the team.



Sprint Planning Meeting

- A meeting at the beginning of a sprint between the Product Owner, the Scrum Master, and the Team.
- Product Owner describes highest priorities and the Team decides what to move from the product backlog to the sprint backlog
- Typically takes 8 hours
- For our purposes this may entail a single class meeting

The Sprint

- A **Sprint** is a 15-30 day period, the actual length being decided by the team, during which the team creates an increment of potentially shippable software. Each day during the sprint a **Scrum Meeting** is held.
- During a sprint NO outside interference with the Team is allowed
- Our sprints will be considerable shorter, lasting only a week or two

The Scrum Meeting (1)

Each day during the sprint, a project status meeting run by the Scrum Master is arranged. This has specific guidelines:

- Anyone may attend and listen at the meeting, but only the Scrum Master and the team members may speak.
- The meeting is typically time-boxed at 15 minutes regardless of the team's size or the project
- The meeting should happen at the same location and same time every day
- Punishments for those who are late to meetings is decided by the Team

The Scrum Meeting (2)

During the meeting, each team member has to answer three very important questions:

- What have you done since yesterday?
- What are you planning to do by tomorrow?
- Do you have any problems preventing you from accomplishing your goal? (It is the role of the Scrum Master to remember and attempt to remove these impediments.)

The Scrum Meeting (3)

- Google Code should have a framework in place for you to regularly answer the 3 questions
- We may have an actual standup meeting with each team once a week

The Project At The End Of The Sprint

- At the end of the sprint the team should have created a **working, potentially shippable increment** of the project
- If there is not a **working, potentially shippable increment** of the project then the sprint has been wasted and time has been lost

Typical Impediments To The Process

- The meeting rules are not followed
- Product Vision and Sprint Goal are unclear
- The Product Owner is not available for questions
- The Product Backlog is not prioritized by business value
- Not everyone who contributes to the delivery is in the team
- The Scrum Master has to perform other tasks and is not able to focus on the team progress
- The team is too big (>10 members)
- The team has no room where they can work together
- The team has no dashboard to access the Sprint Backlog

The Sprint Retrospective

- After each sprint a brief meeting (3 – 4 hours), called a ***sprint* retrospective** is held, at which the Scrum Master and all team members reflect about the past sprint.
- The purpose of this is to review both what went well and what should be improved in the next sprint.
- Once again, for our purpose, this may entail a single class meeting

Scrum vs. Other Agile Methods

- **Lean Development** deals with which comprehensive principles should apply for the entire development organization
- **Scrum** deals with how the project is organized and planned
- **XP (Extreme Programming)** deals with how to work with programming

Benefits Of Scrum (1)

- Scrum is agile at its finest
  - A key principle of Scrum is its recognition that during a project the customers can (and will) change their minds about what they want and need and that unpredicted challenges cannot be easily addressed in a planned or predictive manner.
- Scrum adopts an empirical approach
  - It acceps that the problem cannot be fully understood or defined, focusing instead on maximizing the team's ability to deliver quickly and respond to emerging requirements.

Benefits Of Scrum (2)

- Quickly developed, potentially shippable product produced within 30 days
- Delivers the highest business value features first and will always try and avoid building unrealistic features.
- Frequent customer input
- Customers, through product owner, set development priorities

Scrum websites

- Scrum Alliance
- Control Chaos
- Scrum on Wikipedia

Questions?

Fin.