Scrum

Lecturer: Ngo Huy Bien
Software Engineering Department
Faculty of Information Technology
VNUHCM - University of Science
Ho Chi Minh City, Vietnam
nhbien@fit.hcmus.edu.vn

Objectives

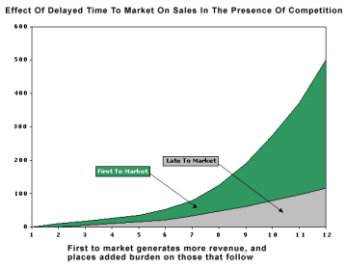
- To present *Agile development* concepts
- To present Scrum *roles*
- To present Scrum *activities*
- To present Scrum *products*
- To *apply* Scrum method to develop a system

Books And Reading

1. James Martin. Rapid application development. 1991.
2. Craig Larman, *Agile and Iterative Development: A Manager's Guide*. 2003.
3. Ken Schwaber. SCRUM Development Process. 1995.
4. Ken Schwaber, *Agile Project Management with Scrum*. 2004.
5. Jeff Sutherland and Ken Schwaber. The Scrum Papers -- Nuts, Bolts, and Origins of an Agile Process. 2007.

Time To Market

Time to market is the time until your product is sufficiently debugged that it can be shipped in volume production.



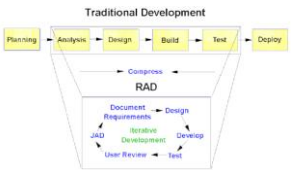
- Your Time To Market Determines The *Success* of Your Product
- Your Time To Market Determines Your *Rate of Return On Investment*

Rapid Application Development [1]

- *RAD* is an approach to building computer systems which combines Computer-Assisted Software Engineering (CASE) tools and techniques, user-driven prototyping, and stringent project delivery time limits into a potent, tested, reliable formula for top-notch quality and productivity.
- RAD takes advantage of *automated tools* and techniques to restructure the process of building information systems.
- RAD replaces hand-design and coding processes, which are dependent upon the skills of isolated individuals, with *automated design and coding*, which is an inherently more stable process.

The RAD Approach

- RAD *compresses* the step-by-step development of conventional methods into an iterative process.
- The RAD approach thus includes *developing and refining* the data models, process models, and prototype in parallel using an iterative process.
- User requirements are refined, a solution is designed, the solution is prototyped, the prototype is reviewed, user input is provided, and the process begins *again*.

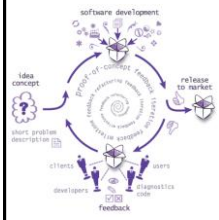


Essential Aspects of RAD

- Rapid Application Development has four essential aspects: methodology, people, management, and tools.
- If any one of these ingredients is *inadequate*, development will not be high speed.

Agile Software Development [2]

Agile development methods apply time-boxed iterative and evolutionary development, adaptive planning, promote evolutionary delivery, and include other values and practices that *encourage agility*—rapid and flexible response to change.



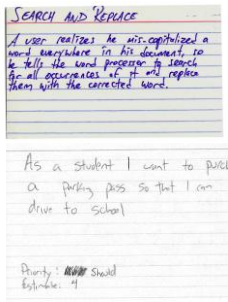
1. *Individuals and interactions* over processes and tools
2. *Working software* over comprehensive documentation (Produce no document unless its need is immediate and significant)
3. *Customer collaboration* over contract negotiation (A successful contract should govern the way the developing team and the customer collaborate rather than details of scope and schedule for a fixed cost.)
4. *Responding to change* over following a plan

Agile Principles

1) Early and continuous delivery of valuable software
2) Welcome changing requirements, even late in development
3) Deliver working software frequently with a preference to the shorter timescale
4) Business people and developers must work together daily
5) Trust individuals to get the job done
6) Face-to-face conversation
7) Working software is the primary measure of progress.
8) Sustainable development
9) Continuous attention to technical excellence and good design enhances agility
10) Simplicity is essential
11) The best architectures, requirements, and designs emerge from self-organizing teams
12) At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly

User Story

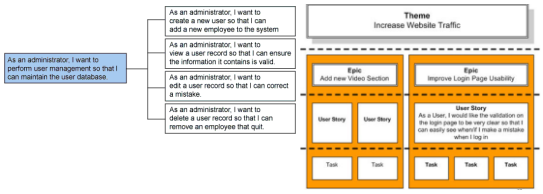
User stories are one-liners that state customer requirements.



I – Independent
N – Negotiable
V – Valuable
E – Estimable
S – Small (+ Screens)
T – Testable

Epics, Themes and Tasks

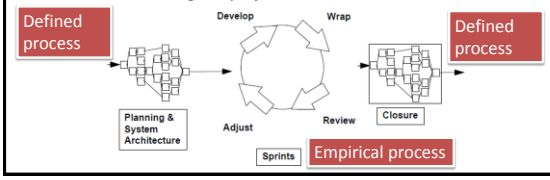
- *Epics* are large user stories that need to be disaggregated into smaller user stories at some point.
- *A theme* is a group of epics.



Scrum 101

What is Scrum? [3]

- SCRUM is a management, enhancement and maintenance methodology for *an existing system* or *production prototype*.
- Waterfall and Spiral methodologies set the context and *deliverable definition* at the start of a project.
- SCRUM and Iterative methodologies initially plan the context and *broad deliverable definition*, and then evolve the deliverable during the project based on the environment.



Kick - Off Meeting

Kick-off meeting is held to agree on the foundation objectives and requirements of the project.

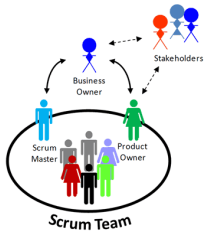
1. Project Vision
2. Feasibility Study
3. Statement of Work
4. Proof of Concept

The Product Owner

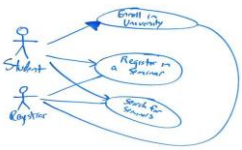
- Representing the interests of everyone with a stake in the project and its *resulting system*.
- Achieving initial and ongoing funding for the project by creating the project's initial overall requirements, return on investment (ROI) objectives, and *release plans*.
- Using the *Product Backlog* to ensure that the most valuable functionality is produced first and built upon.

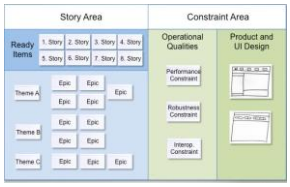
The Team

- *Developing* functionality
- *Self-managing*, self-organizing, and cross-functional
- The Scrum Master
  - Scrum *process*
  - *Teaching* Scrum to everyone involved in the project
  - Implementing Scrum so that it fits within an *organization's culture*
  - Ensuring that everyone follows Scrum *rules* and practices



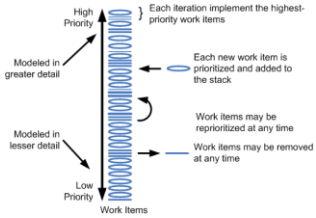
Requirements Gathering





Product Backlog

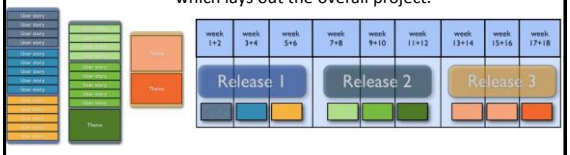
The Product Backlog is the master list of *all functionality* desired in the product, prioritized as an absolute ordering by business value, containing *rough estimates* of both *business value*, frequency of use and *development effort*.



We don't need a product backlog we just need to figure out what to do for the next Sprint!!!

Release Planning

Release planning is a meeting used to create a release plan, which lays out the overall project.



Release backlog are the user stories that are included in the next release.

We CAN'T come up with accurate estimates!!!

Other Planning Activities

- Assessment of risk and appropriate *risk controls*.
- Validation or reselection of *development tools* and infrastructure.
- Estimation of *release cost*, including development, collateral material, marketing, training, and rollout.
- Verification of *management approval* and funding.

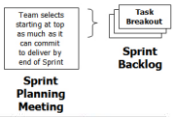
Architecture/High Level Design

- Review *assigned backlog items*.
- Identify changes necessary to *implement backlog items*.
- Perform *domain analysis* to the extent required to build, enhance, or update the domain models to reflect the new system context and requirements.
- Refine the *system architecture* to support the new context and requirements.
- *Identify any problems* or issues in developing or implementing the changes.
- Design *review meeting*, each team presenting approach and changes to implement each backlog item. Reassign changes as required.

Sprint Planning [4] [5]

The sprint backlog is the list of work the team must address during the next sprint. Features are broken down into tasks, which, as a best practice, should normally be between four and sixteen hours of work.

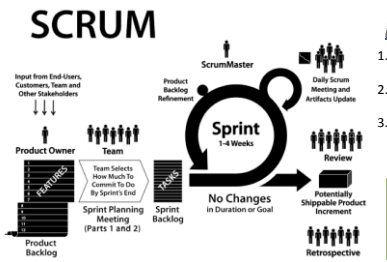
What Does "Done" Mean?



Backlog Item	Task	Owner	Initial Time Estimate
Enable all users to place book in shopping cart	Configure database and space IDs for Trac	Sanjay	4 hours
	Use test data to tune the learning and action model	Jing	2 hours
	Setup a cart server code to run as apache server	Philip	3 hours
	Implement pre-Login Handler	Tracy	3 hours
Upgrade transaction processing module (must be able to support 500 transactions /sec)	Merge DCP code and complete layer-level tests	Jing	5 hours
	Complete machine order for pRank	Jing	4 hours
	Change DCP and reader to use pRank http API	Tracy	3 hours

Sprints

With Scrum, projects progress via a *series of iterations* called sprints. Each sprint is typically *2-4 weeks* long.

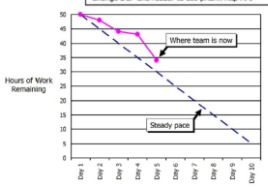


1. What did you do since last Scrum meeting?
2. Do you have any obstacles?
3. What will you do before next meeting?

The team's ability to tackle its problems and solve them is the heart of Scrum

Burn-down Chart

Task	Task Owner	Hours of Work Remaining on Each Day of the Sprint: Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
Configure database and space IDs for Trac	Sanjay	4	4	3	1	0					
Use test data to tune the learning and action model	Jing	2	2	2	2	1					
Setup a cart server code to run as apache server	Philip	3	3	5	2	0					
Implement pre-Login Handler	Tracy	3	3	3	3	3					
Merge DCP code and complete layer-level tests	Jing	5	5	2	2	2					
Complete machine order for pRank	Jing	4	4	3	3	3					
Change DCP and reader to use pRank http API	Tracy	3	3	0	0	0					
	Total	50	48	44	43	34					

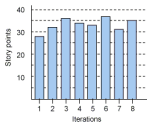


Burn-down chart shows, each day, *how much work* (measured in hours or days) *remains* until the team's commitment is completed.

Measuring Velocity

Iteration	Points
1	28
2	32
3	36
4	34
5	33
6	37
7	31
8	35

Velocity is the long-term tracking of how much work has been done by a team per iteration.

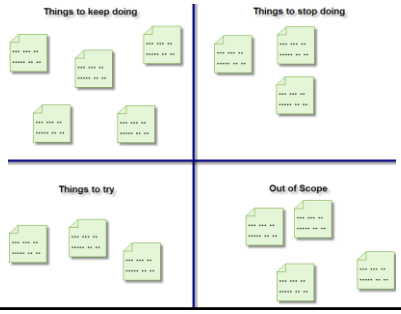


Sprint Review

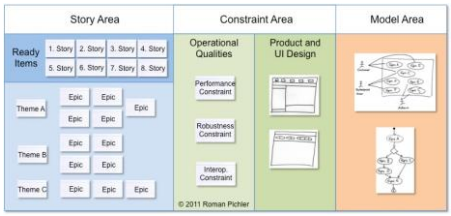
Sprint review is a meeting at after the Sprint ends, it's just a demo of what's been built, and anyone present is free to ask questions and give input.

Sprint Retrospective

Sprint retrospective is a meeting for the team to discuss what's working and what's not working, and agree on changes to try.

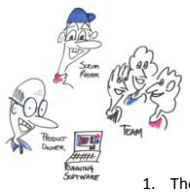


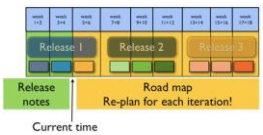
Product Backlog Board



Scrum Work Products

Re-planning





1. The schedule is *not changed*. The 4th iteration still ends with week number 8, and the releases are delivered in the weeks 6, 12 and 18.
2. The *scope* and content of the project *is adjusted empirically* as the developers and customers gain understanding of the solution.

Scrum Values

- Commitment
- Focus
- Openness
- Respect
- Courage

Methodology Comparison [3]

	Waterfall	Spiral	Iterative	SCRUM
Defined processes	Required	Required	Required	Planning & Closure only
Final product	Determined during planning	Determined during planning	Set during project	Set during project
Project cost	Determined during planning	Partially variable	Set during project	Set during project
Completion date	Determined during planning	Partially variable	Set during project	Set during project
Responsiveness to environment	Planning only	Planning primarily	At end of each iteration	**Throughout**
Team flexibility, creativity	Limited - cookbook approach	Limited - cookbook approach	Limited - cookbook approach	**Unlimited during iterations**
Knowledge transfer	Training prior to project	Training prior to project	Training prior to project	**Teamwork during project**
Probability of success	Low	Medium low	Medium	**High**

Fixed-Price, Fixed-Date Contracts

Capability Maturity Model Integration

Thank You For Your Time