

Agile to the Bone

**Introduction to Agile
by Pietari Kettunen**

Agenda

- Problem with traditional software engineering
- Why Agile is the solution?
- Roots of Agile
- Values of Agile
- Common implementations
 - Scrum
 - Kanban
- Common pitfalls

Me

- Born in Finland
- M.Sc. on Software Engineering from Tampere University of Technology
- ~10 years of experience
- Employed by Swisscom Strategy & Innovation
- Contact:
 - Twitter: @Pietrotull
 - kettunen@gmail.com
- Blog: www.pietrotull.com

Traditional S.E.



- Modelled after construction engineering
 - Architect
 - Software Architecture
 - Build tools

Nou Mestalla



Shame of Scotland



The Problem

“Only thing that is constant is change”

- Heraclitus

Construction vs Software

low design cost & high build cost

VS

high design cost & minimal “build” cost

The Solution: Agile



- Adaptation
- Embrace change

The Essence of Agile

“Agile methods are adaptive rather than predictive”

- Martin Fowler

Roots of Agile Movement

- MIT Hacker Culture
- Organizational Patterns by James O. Coplien
- Toyota Manufacturing System
- The New Product Development Game -1986

Agile Manifesto

- February 2001

- Kent Beck
- Mike Beedle
- Arie van Bennekum
- Alistair Cockburn
- Ward Cunningham
- Martin Fowler
- James Grenning
- Jim Highsmith

- Andrew Hunt
- Ron Jeffries
- Jon Kern
- Brian Marick
- Robert C. Martin
- Steve Mellor
- Ken Schwaber
- Jeff Sutherland
- Dave Thomas

Agile Manifesto

Individuals and interactions over

processes and tools

Working software over

comprehensive documentation

Customer collaboration over

contract negotiation

Responding to change over

following a plan

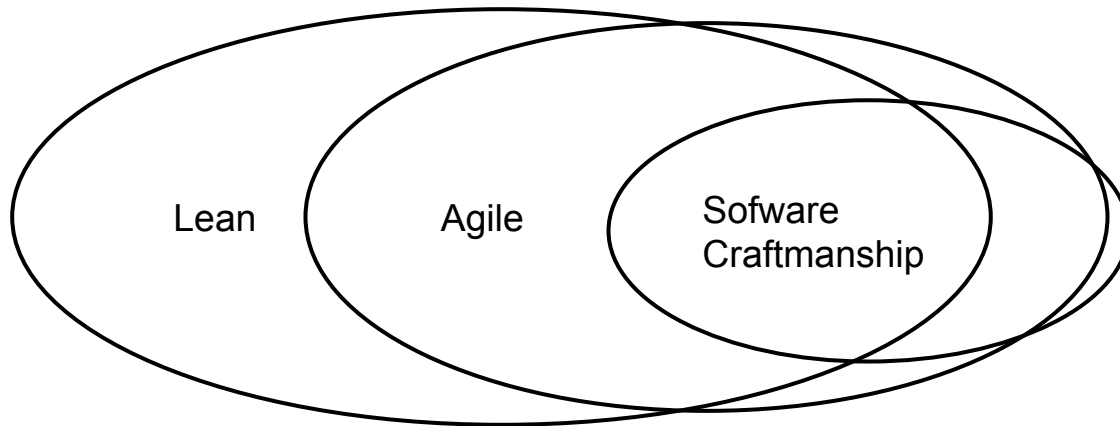
12 Principles of Agile

- customer satisfaction
- embrace change
- frequent delivery
- collaboration
- motivated individuals
- face to face
- working software
- sustainable development
- technical excellence
- simplicity
- self-organizing team
- retrospection

Communication



Holy Trinity of Software



- Lean Software Development
- Agile
- Software Craftmanship

Lean

Three enemies of effectiveness

- Muda
 - waste
- Mura
 - inconsistencies
- Muri
 - disturbances in the flow

Lean applied to software

- Eliminate Waste
 - Build Quality In
 - Create Knowledge
 - Defer Commitment
 - Deliver Fast
 - Respect People
 - Optimize the Whole
- by Mary Poppendieck



Software Craftmanship

Not only working software,

but also well-crafted software

Not only responding to change,

but also steadily adding value

Not only individuals and interactions,

but also a community of professionals

Not only customer collaboration,

but also productive partnerships

Why Agile Works

- Cynefin framework
- Nature of exploration
- do - evaluate - adjust

Yin & Yang of Agile

- Technical Practises
- Process Management

Technical practices

- Pair programming
- Test driven development
- Continuous integration
- Continuous deployment
- Behaviour driven development / specification by example

Process Methodologies

- Scrum
- Kanban
- XP

Process Methods

- Just In Time decisions
- Pull -mechanism
- Visualization
- Transparency
- Splitting work into smaller pieces
- Limiting work in process

Scrum



Scrum



Pioneering Scrum

Scrum (early 90's)

- Jeff Sutherland (Easel Corp)
- Ken Schwaber (Advanced Development Methods)

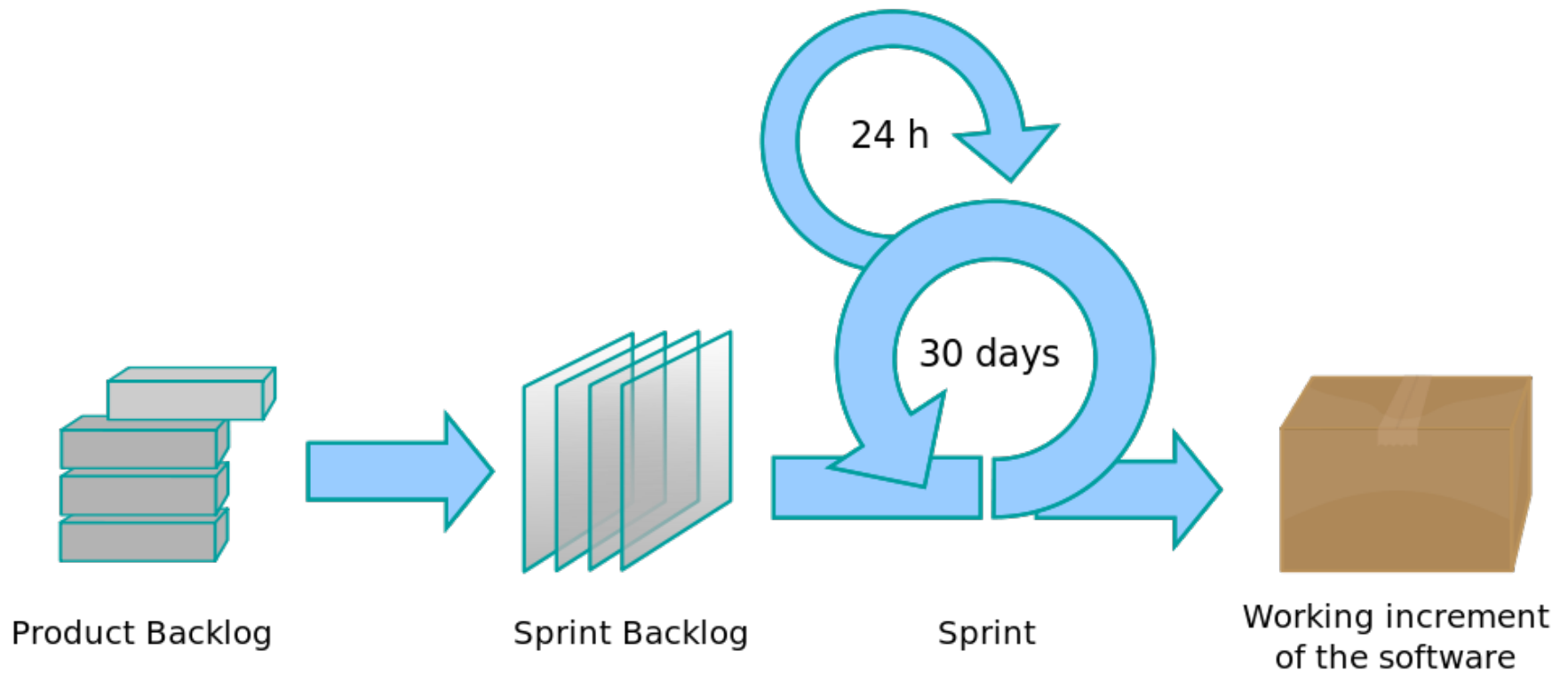
Jeff & Ken collaborated to present

- Scrum methodology at OOPSLA '95

Scrum

- Agile process for producing business value
- Iterative
- Produce value
- does NOT prescribe technical practises

Scrum in action



Scrum Roles

- Product owner
- Scrum master
- Team

Product Owner

- Decides priorities / order
- Vision of the product

ScrumMaster

- Takes care of the process
- Enables the team to do their work
- Owns the impediment list

Team

- Makes the magic happen
- Crossfunctional
 - includes all the skills to finish the product
- Self-organizing

Scrum Ceremonies

1. Sprint planning
2. Daily scrum meeting
3. Sprint review
4. Sprint retrospective

1. Sprint planning

Who

- Team, ScrumMaster & Product Owner

Agenda

- Discuss top priority backlog items
- Team selects which to do

Why

- Know what will be worked on
- Understand it enough to do it

2. The daily scrum

Parameters:

- Daily
- max 15 minutes
- standing

Not a problem solving meeting

- Whole world is invited
- only team, ScrumMaster, Product Owner can talk

2. The daily scrum

Questions

1. What did you do?
2. What will you do?
3. Is there anything stopping you?
(impediment)

3. Sprint review

- Inspect and adapt the product
- Team presents what was accomplished
- Typically involves a demo
- Informal

4. Retrospective

- Inspect and adapt the process
- Everyone can participate

Scrum Artifacts

- Product backlog
- Sprint goal
- Sprint backlog
- Burndown chart
- Impediment List

Product backlog

- List of desired work
- Ordered / prioritized by the product owner
- Reorganized at the start of each sprint

Sprint goal

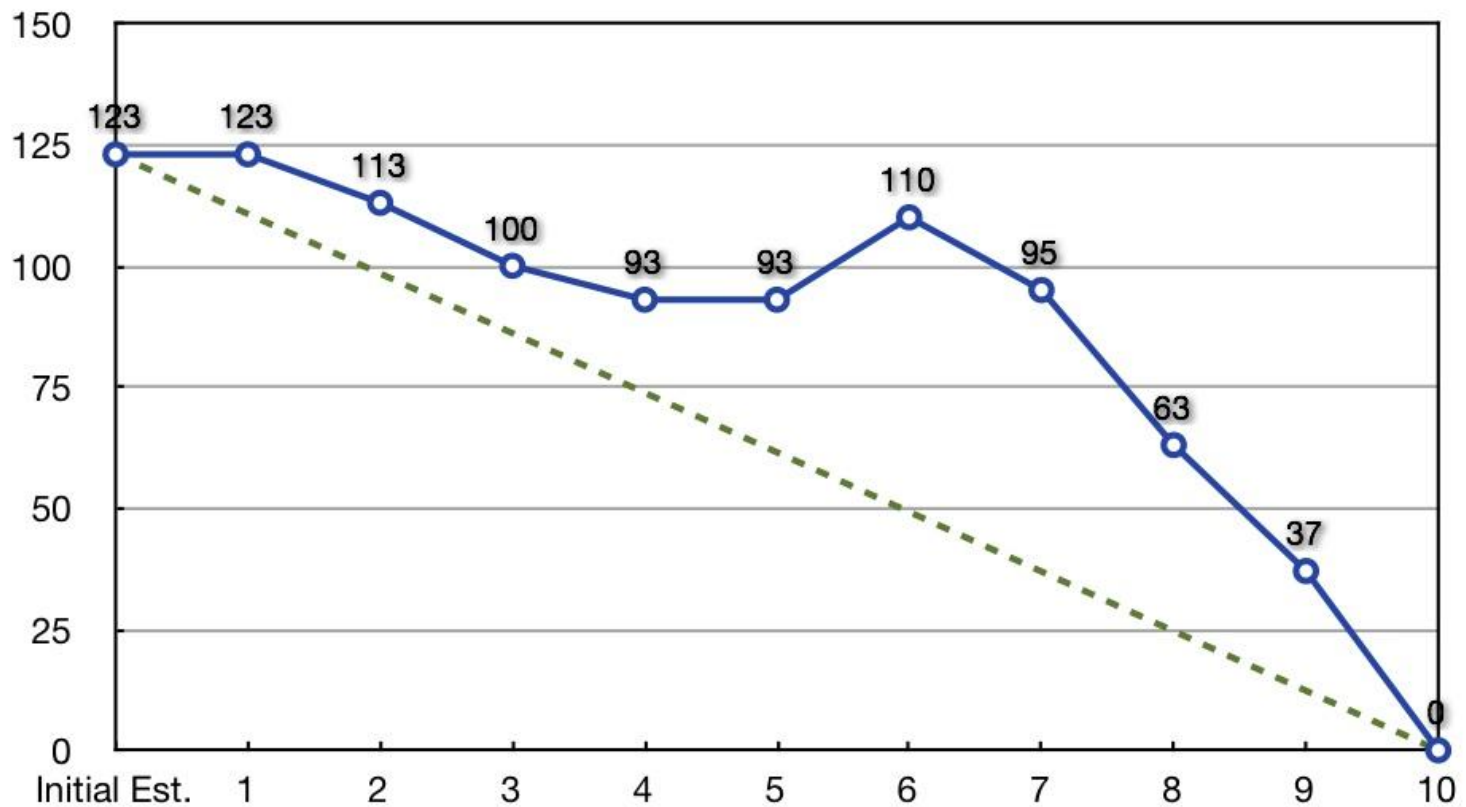
High level summary of where the focus is for given sprint

For the “high level” boss

Sprint backlog

- Evolves
- Team maintains
 - can add tasks
 - can remove tasks
 - re-estimate
- The team owns the sprint backlog
- “Best guess” what the team needs to do
- Progress visible in the task board

Burndown chart



Scaling Scrum

- Several Scrum teams
- Scrum of scrums
 - coordination over several Scrum teams

Kanban



Kanban

- Kan ban = “signal card”
- Originally by Taichi Ono (Toyota)
- Software Kanban by David Anderson
- Evolutionary approach

Kanban method

1. Start with what you have
2. Agree to pursue incremental, evolutionary change
3. Respect current process, roles & titles
4. Leadership at all levels

Kanban 6 practises

1. Visualize workflow
2. Limit work in progress
3. Manage flow
4. Explicit policies
5. Implement feedback loops
6. Improve collaboratively, evolve with experiments

1. Visualize workflow

- Analyze work states
- Define work item types
- Make problems visible

2. Limit work in progress



MULTITASKING

- Prevent multitasking / context switching
- Less work in progress = less waste

3. Manage flow



- Sustainable pace

4. Explicit policies



- No secrets
- Gew clear rules



5. Improve collaboratively

" To be termed scientific, a method of inquiry must be based on gathering empirical and measurable evidence subject to specific principles of reasoning"

-Isaac Newton

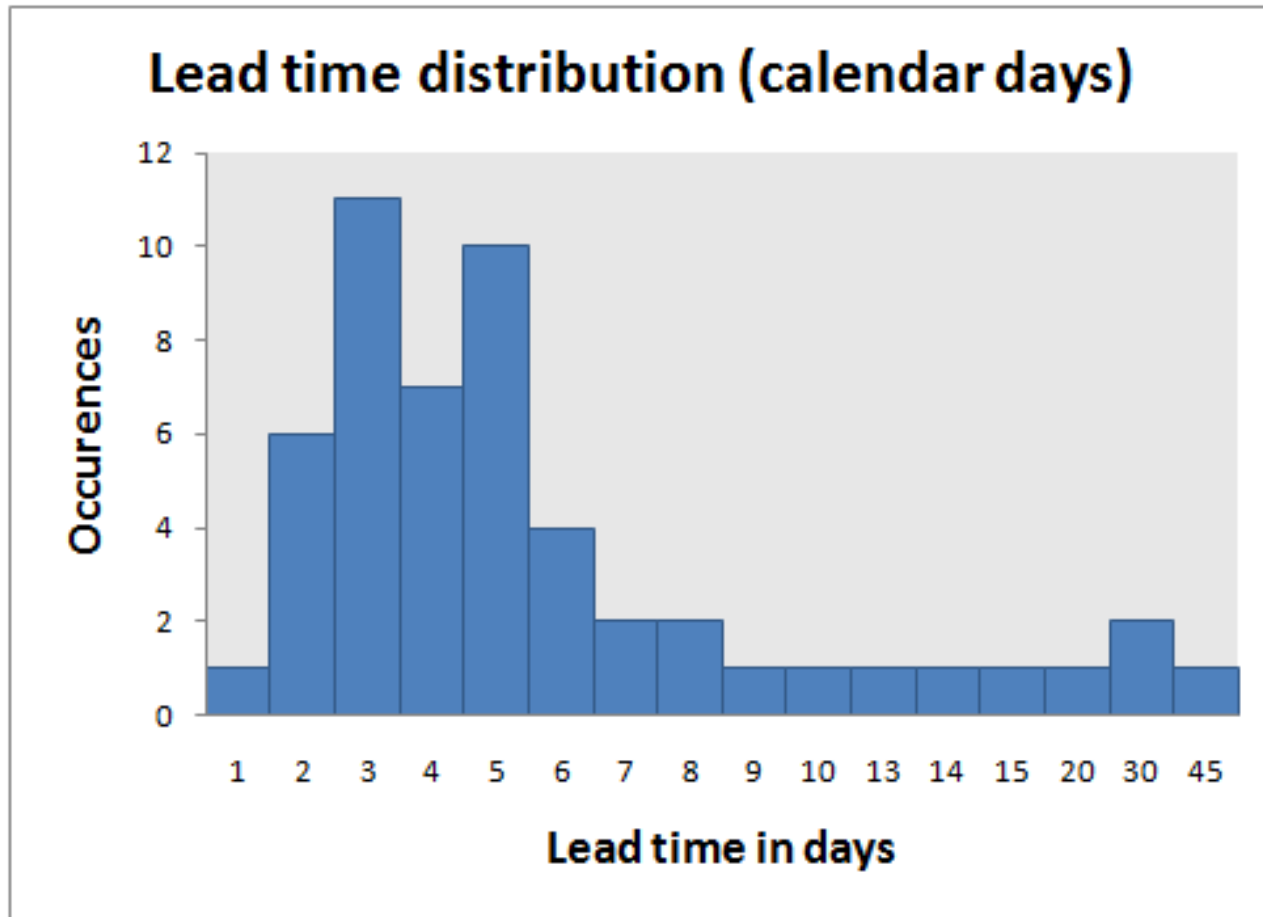
- Collaboration with all stakeholders
- avoid local optimizations

Cadence



- Everything has it's own rhythm

Estimation



Scrum vs Kanban

Scrum

- WIP per sprint
- Sprint content set
- Task size
- Crossfunctional teams
- Timeboxed
- Velocity
- Fair amount of rules

Kanban

- WIP per stage
- No untouchable tasklist
- Task size unlimited
- Allows specialist teams
- No time limits
- Lead time
- Very few rules

Tool for the job?



Agile Documentation

- Documentation is a poor substitute for conversation
- UI mockups
- Only code is up to date

Code example 1

```
public List<int[]> getThem() {  
    List<int[]> list1 = new ArrayList<int[]>();  
    for(int[] x : theList) {  
        if (x[0] == 4) {  
            list1.add(x);  
        }  
    }  
    return list1;  
}
```

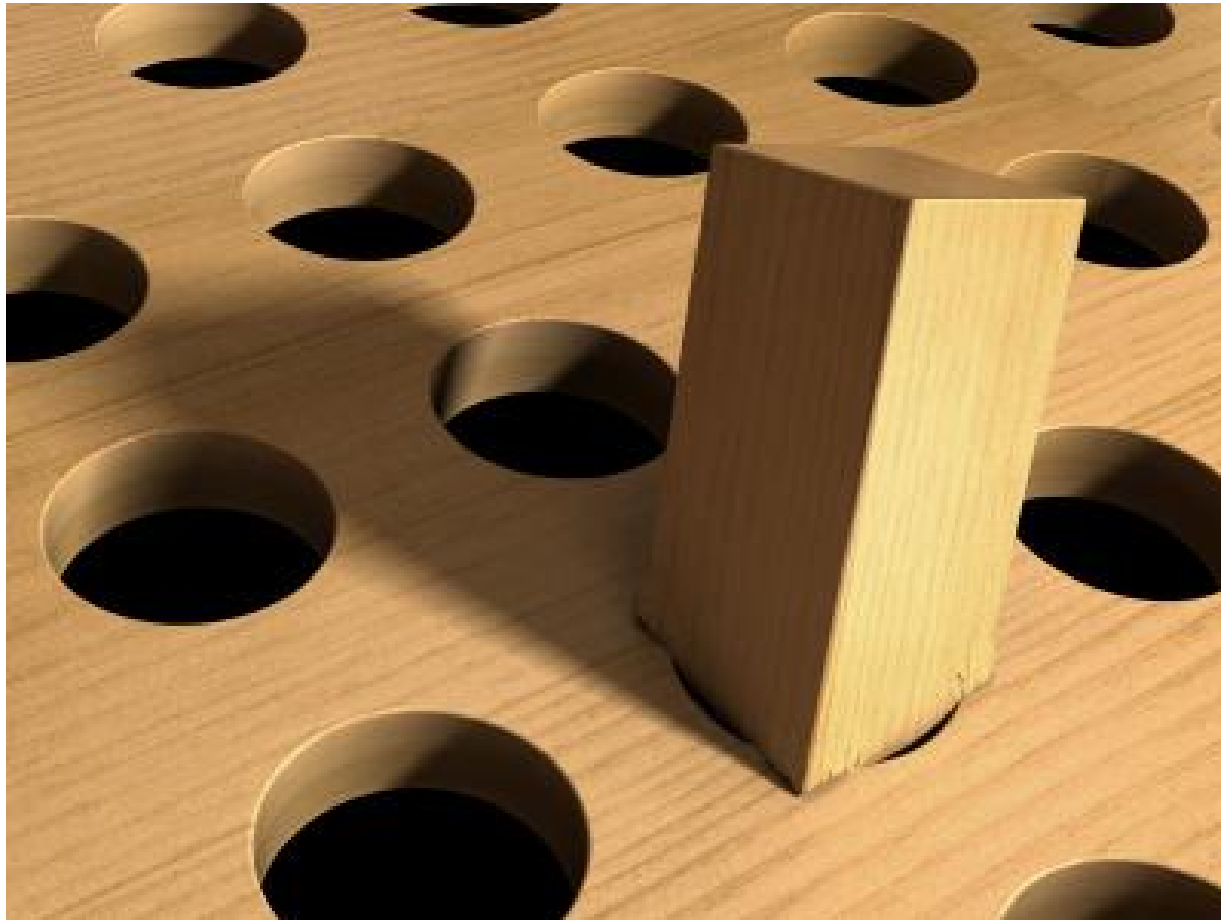
Code example 2

```
public List<int[]> getFlaggedCells() {  
    List<int[]> flaggedCells = new ArrayList<int[]>();  
    for(int[] cell : gameBoard) {  
        if(cell[STATUS_VALUE] == FLAGGED) {  
            flaggedCells.add(cell)  
        }  
    }  
    return flaggedCells;  
}
```

Do the right thing



Do the right thing



User Stories

- Communication problem
- Way to describe functionality
- Collaborative effort

User Story example

As a *<user>* I want to *<action>*

E-Bank example

- As an *account holder*, I want to *withdraw cash* from ATM

Add context

- So that I can get money when the bank is closed

Scenario

Given

The the account balance is 100CHF
and the card is valid

When

The account holder requests 20CHF

Then

The ATM should dispense 20CHF
and the account balance should be 80CHF
and the card should be returned

Minimum Viable Product

- pareto principle (80/20)
- Just In Time

Full of features (2007)

	Nokia N95	competitor
3G	yes	no
Camera	5mp	2mp
Memory card	microSD up to 32GB	no
MMS	yes	no
3rd party apps	yes	no
Video out	yes	no
VoIP	yes	no
Video calls	yes	no
Instant Messaging	yes	no
Bluetooth	yes	no

N95 vs iPhone



Some other stuff



Holy Physical Task Board

- Promotes team interaction
- Visibility



Kaizen

- Continuous Improvement

Slack



Feedback

- feedback is information
- every mistake is a change to learn

“I didn’t fail, I found 10 000 ways that didn’t work”

- Thomas Edison

Testing

“Only fully tested software is the one not yet implemented.”

“Verify functionality”

Common Pitfalls

- Estimation
- Illusion of Importance
- Find and Replace
- Detailed plans

The Black Art of Estimation



www.shutterstock.com - 36926263

REVERENDFUN.COM COPYRIGHT BIBLE GATEWAY



(See Matthew 25:21)

01-21-2011

HOLD UP ... WE ALSO FORGOT TO CARRY A ONE ON PAGE THREE

“predicting is very difficult, especially if it involves the future”

- Niels Bohr