

10 Ways to Go from Good to Great Scrum Master

Benjamin Day

- Brookline, MA
- Consultant, Coach, & Trainer
- Microsoft MVP for Visual Studio ALM
- Team Foundation Server, Software Testing, Scrum, Software Architecture
- Scrum.org Classes
 - Professional Scrum Developer (PSD)
 - Professional Scrum Foundations (PSF)
- www.benday.com,
benday@benday.com, @benday




Benjamin Day Consulting





Online courses at Pluralsight.com





Relevance Newest Popular Rating Level

[Load Testing with Visual Studio 2013](#)
by Benjamin Day  Intermediate 3h 7m 06 Dec 2014

[Real World Scrum With Team Foundation Server 2013](#)
by Benjamin Day  Intermediate 5h 2m 10 Nov 2014

[Load Testing with Visual Studio 2012 !\[\]\(b4572a044582c68c9e6e6b6b9b95c325_img.jpg\)](#)
by Benjamin Day  Intermediate 4h 3m 26 Aug 2013

[ALM for Developers with Visual Studio 2012 !\[\]\(c1b926fcee63cdc7f15c603966ea3d76_img.jpg\)](#)
by Benjamin Day  Intermediate 4h 31m 24 Sep 2012

[ALM with TFS 2012 Fundamentals !\[\]\(3e3e4303b0632a7f1c9ba6d71d06a491_img.jpg\)](#)
by Benjamin Day  Intermediate 5h 34m 05 Sep 2012

On with the show.

Assumptions

- You're technical. I'm technical.
- For today, we're talking about a fairly technical Scrum Master
 - (Don't forget the needs of the business though.)
- This talk =
Scrum Mastering + "Getting your project started right"

What does the Scrum Master do?

- Keeper of the process
- Help the team deliver
- Advocate for the team
- Help the Product Owner

Top 10 Ways: Things to Think About

- What can you do to help your team?
- What can you do to help your product owner?
- What can you do to help yourself?
- What can you do to help your organization?
- Change your Daily Scrum

Here are all the things I wanted to put in this talk.

- What can you do to help your team?
 - Written DoD
 - Encourage self-organization
 - Try to help minimize the work in progress.
 - Remind them to keep talking. One major downside of TFS is that people can sometimes think of it as an excuse to not talk.
 - Remind them about how much and how creatively you can decompose a PBI
 - Emergent Architecture.
 - Remind them they don't report to you
- What can you do to help your product owner?
 - [Product owner checklist <http://productownerchecklist.org>](http://productownerchecklist.org) by Lare Lekman
 - Remind them that they need to stay engaged and that their engagement is critical for the success of the team and the effort.
 - Remind them to plan ahead a bit and share their vision
 - Help with credibility by planning 2 to 3 sprints out
- What can you do to help yourself?
 - You can't fix everything.
 - Watch for burnout.
 - Do whatever it takes to get you to believe in self-organization
- What can you do to help your organization?
 - Get them cozy with the fact that change will be coming.
 - Changing of plans is a "feature" not a "bug."
 - Inoculate them so that they won't panic when change inevitably happens.
 - Question how products align with teams. Bring work to the teams rather than bringing Teams to the work. Let the Teams stay together and focused on a limited amount of stuff and they'll go faster.
 - Help them think about sprint lengths. Sprints help scope risk. Sprints force you to pause from time to time to see where you actually are. It helps keep you honest. Have you *actually* delivered anything? Why and why not?
 - Practice explaining Story Points.
 - Explain that Scrum doesn't replace the existing org structure. It is a layer on top of the existing org structure. While it probably won't change anyone's job, it might change how they perform their job.
 - Think like a data scientist. <http://blogs.hbr.org/2013/11/how-to-start-thinking-like-a-data-scientist/>
 - Watch out for fear. Fear is *everywhere*.
- Address the QA problems.
 - What does QA do at the front side of the sprint?
 - Answer collaborate with fellow team members to ensure that everyone knows how stuff is going to be tested.
 - Basically, front-load your quality focus.
- Change your daily standup
 - Consider dumping "3 questions"
 - Consider switching who "runs" the standup
 - Watch for people being bored and/or tuning out
 - Bored people implies doing too much work and/or a wobbly sprint goal
 - Bored people implies multiple teams of 1 rather than a team of X. You have people who sit near each other rather than actual functional team.
- What is your Velocity?
 - If you don't have estimates, you can't have a velocity.
 - If you don't have a DoD, velocity is problematic.
 - If you're wobbly on DoD, your velocity is suspect.
 - Be firm on DoD. No partial credit.
- Better Retrospectives
 - Write them down. Review the results.
 - Try to walk through the sprint day by day and ask people how they felt, what was going on on this day, why is the burndown going up or down? This helps them to "remember" what was happening.
- Common Objections to Scrum
- Leading vs Coaching
- Coaching organizations
- Coaching teams
- Common objections

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

Top 10 Ways

1. ***What is your DoD?***
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

Definition of Done (DoD) =
Everything it takes to say something is
completely done.

What is your DoD?

- Closest thing to a “silver bullet” in Scrum
- Technical Debt will ruin you.
- Write it down.
- Review and discuss it regularly.

“Done vs. Done Done”

Do not relax your DoD.

No partial credit...ever.

Partial credit usually means
Technical Debt.

Partial Credit & Technical Debt →
“Wobbly” Velocity

Top 10 Ways

1. What is your DoD?
2. ***People Skills***
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

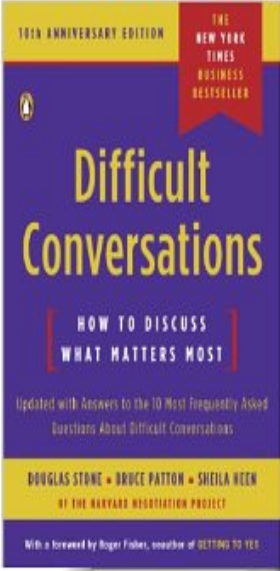
99% of the time,
it's a people problem.

Watch for interpersonal problems.

Here's a trick:
"Trust your gut."

Consider reading this book.

Look inside ↓



10th ANNIVERSARY EDITION THE NEW YORK TIMES BUSINESS BESTSELLER

Difficult Conversations

HOW TO DISCUSS WHAT MATTERS MOST

Updated with Answers to the 10 Most Frequently Asked Questions About Difficult Conversations

DOUGLAS STONE • BRUCE PATTON • SHEILA HEEN
OF THE HARVARD NEGOTIATION PROJECT

With a foreword by Roger Fisher, coauthor of GETTING TO YES

kindle edition

Click to open expanded view

Difficult Conversations: How to Discuss What Matters Most [Kindle Edition]

[Douglas Stone](#) (Author), [Bruce Patton](#) (Author), [Sheila Heen](#) (Author), [Roger Fisher](#) (Foreword)

★★★★★ (88 customer reviews)

Print List Price: ~~\$17.00~~
Kindle Price: **\$8.85**
You Save: **\$8.15 (48%)**
Sold by: Penguin Group (USA) LLC

- Length: 352 pages
- Don't have a Kindle? [Get your Kindle here.](#)

Formats	Amazon Price	New from	Used from
Kindle Edition	\$8.85	--	--
Hardcover	\$13.32	\$7.81	\$4.43

Want to be a super hero?
Go see a therapist.
(I'm not kidding.)

Top 10 Ways

1. What is your DoD?
2. People Skills
3. ***Credibility***
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

2 to 3 Sprints of Product Backlog

- Helps give the PO “room”
- Helps everyone to know what’s going on.
- Be ready to answer where a PBI (aka. “feature”) is on the backlog.
 - How many sprints out?
- Discuss your *written* DoD with the PO, executives, and stakeholders
 - Why is it in their best interest?

Fit for purpose.

- Remember that not everything is, needs to be, or should be a work of art.
- Balance “Time to Market” with “Long-term Maintenance”
- Communicate with the business in terms they understand
 - (Hint: this is probably money and resources.)

Train everyone to say “forecast” rather than “commitment”

The screenshot shows a weather website interface for Boston, MA. At the top, there are navigation tabs for Weather, Lifestyle, Social, Maps, Video, News, and TV. Below these are sub-tabs for Local, National Forecast, Severe Weather, Alerts, Hurricane Central, and Safety & Security. The main content area is titled "Boston Weather" and includes a "Local Pollen Alert" section. The "My 10 Day Forecast" section is the primary focus, displaying weather conditions for the next several days. The forecast data is as follows:

Day	Weather	High/Low	Chance of Rain	Wind
Today (May 25)	Few Showers	73° / 64°	30%	SSE at 11 mph
Sat (May 26)	Partly Cloudy	87° / 59°	20%	W at 10 mph
Sun (May 27)	Showers	72° / 59°	50%	ESE at 9 mph
Mon (May 28)	Partly Cloudy	67° / 59°	20%	E at 7 mph

Additional elements include a "Right Now" section on the left, a "TRUPOINT BETA" logo, and a "Recommendations" sidebar on the right.

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. ***What is your Sprint Goal?***
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

Do you have a Sprint Goal?

Is it easily understood and stated?

Hint:

Your sprint goal should not be

```
select * from  
SprintBacklog
```

Review your Sprint Goal
in the Daily Scrum.

And while we're talking about
the Daily Scrum...

...consider dumping the
“3 questions” format.

Remind your team that they
don't report to you.

Consider switching up who
“leads” that meeting.

Watch for people being bored
and/or tuning out.

Bored implies a lack of focus.

Bored implies that people aren't on the same team.

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. ***Do less.***
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

Minimize work in progress.

Put another way...
don't try to do everything at once.

If everyone on your team is working on
separate PBIs,
is the team really a team?

Finish one thing. Then do the next thing.

- Try to craft the work so that multiple people are working on related things
- Complete that thing. Move on to the next thing.
- Try to drive stuff to DoD early.
- This ensures that you're delivering *something* in the sprint.

Anti-pattern:
2 days from the end of the sprint
and nothing's DoD yet.

Top 10 Ways

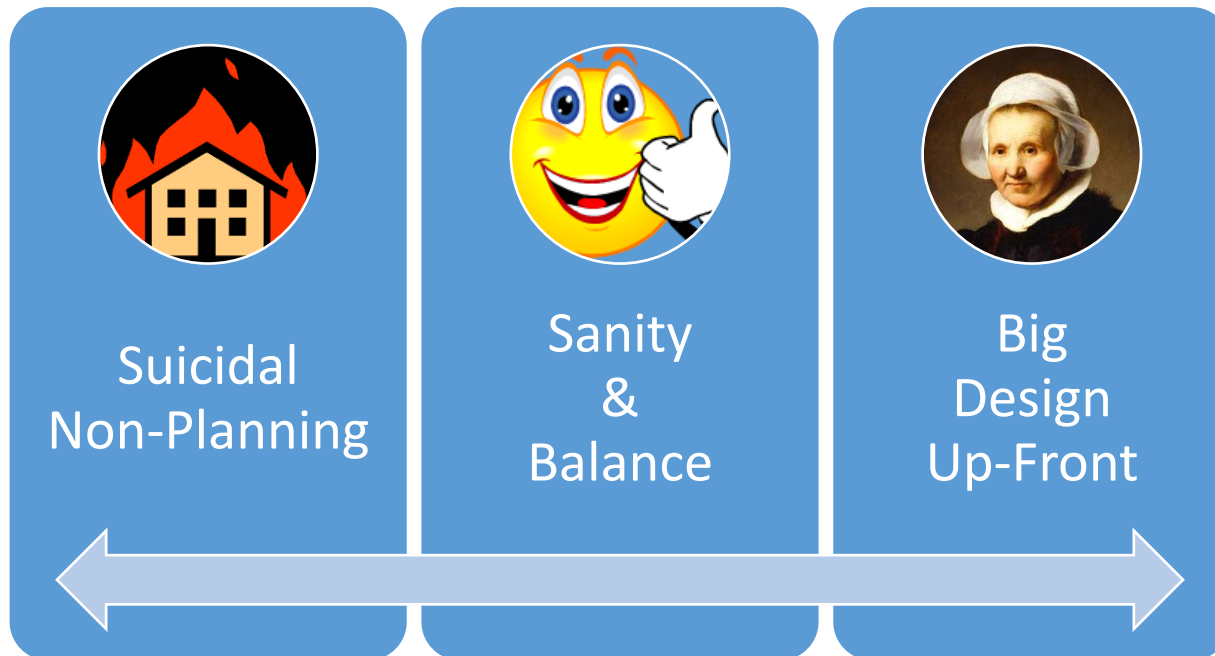
1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. ***Emergent Architecture***
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

Avoid BDUF.
(Big design up-front.)

YAGNI.
(You ain't gunna need it.)

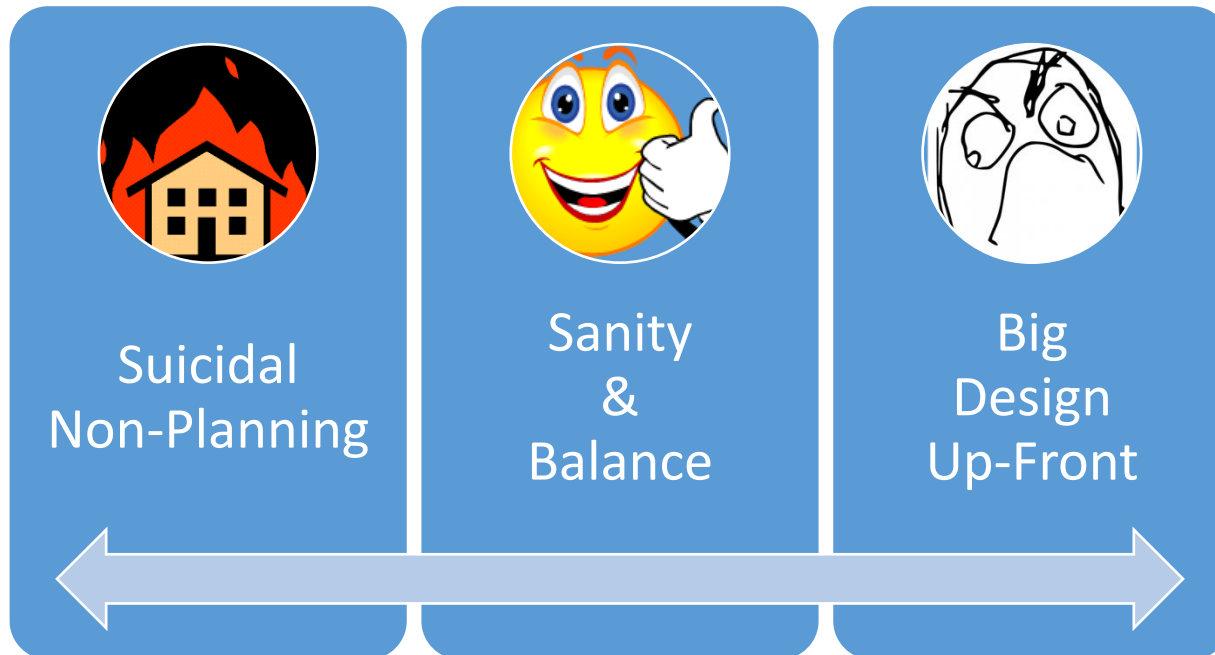
Emergent Architecture.

- Build what you need.
- It's a spectrum.



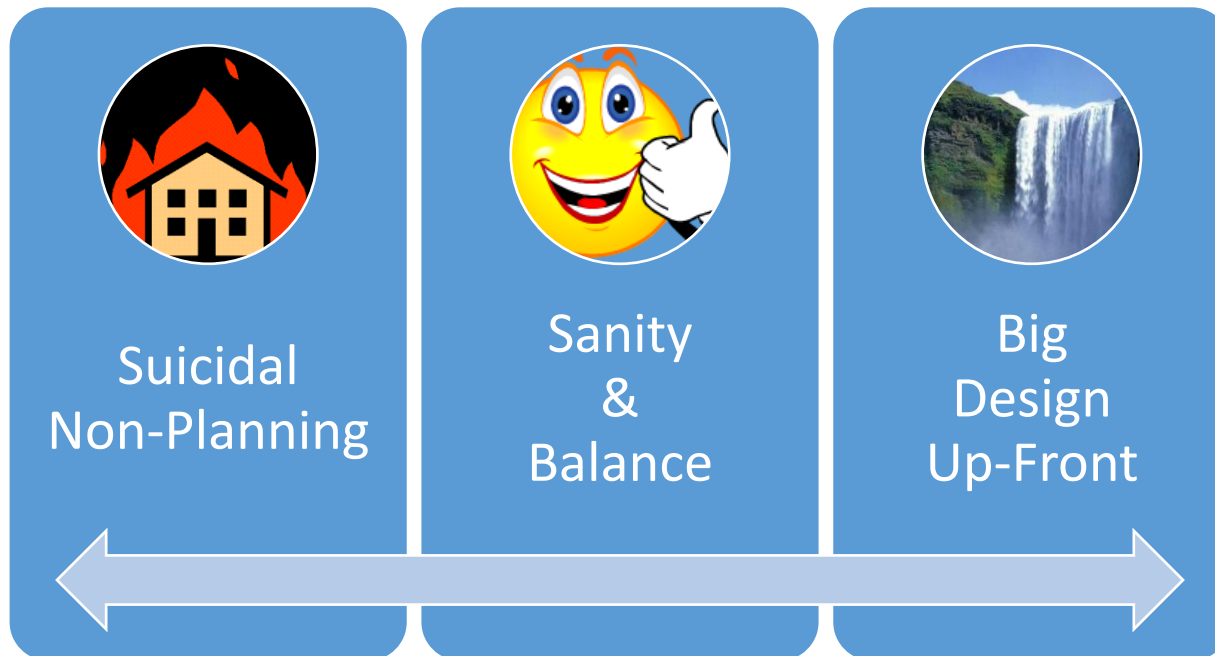
Emergent Architecture.

- Build what you need.
- It's a spectrum.



Emergent Architecture.

- Build what you need.
- It's a spectrum.



Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. ***Good programming & testing practices***
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

It's a metaphysical *certainty*
that you'll have to change stuff.

You *won't* get your
“requirements” right.

Your customers
will
change their minds.

Accept that
you'll have to change.

The Goal:
Make refactoring painless.

Loose coupling.

- Code to interfaces.
- Use the Dependency Injection Pattern
 - (Pass dependencies in on the constructor.)
 - Consider an IoC Framework
- Use the Repository Pattern
- Remember Single Responsibility Principle

Build for Testability

- Unit test, unit test, unit test
- Unit test != Integration Test
- Test one layer at a time in isolation
- No database connections from a unit test
- Integration tests in a separate project
 - Keep yourself honest

For the love of all things
precious & beautiful...



Biohazard

...DON'T USE A SHARED DEVELOPMENT DATABASE!!!!!!!!!!



Biohazard





How do I think about unit testing Silverlight? What's my process? (Part 1)
Beantown .NET Meeting, 2/3/2011: Bob German, "SharePoint 2010 Development"



Top 4 Reasons Why a Shared Development Database is Evil.

BY BEN DAY ON JANUARY 25, 2011 · 1 COMMENT · IN APPLICATION LIFECYCLE MANAGEMENT (ALM), BEST PRACTICES WITH VISUAL STUDIO ULTIMATE 2010, SQL SERVER, TEAM FOUNDATION SERVER 2010, UNIT TESTING / TEST-DRIVEN DEVELOPMENT (TDD), VSTS DBPRO

It's pretty common. A development team will frequently have one instance of SQL Server — filled with data from production — that all the devs use to develop and test the next version of the application. It's one of those things in software development that feels like "common sense". Of *course* you're going to have a single instance of the database that everyone uses. How else would you keep all the data synchronized across everyone's development workstation?

RECENT

- How to get numbers Test.
- Call to Visual Studio
- Slides & Studio
- Scrum vs
- Predict Windows

<http://tinyurl.com/bqextsa>

Eliminate the "works on my box" problem *early*.

- Set up automated builds from the very beginning.
- If you're using TFS, use Gated Check-in builds
- Deploy your database as part of your builds.
- Run your unit tests from the builds

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. ***“Kick it over the wall to QA”***
9. Teams : Products
10. Retrospective

Never say
“Kick it over the wall to QA” again.

- Us vs. Them
- QA is part of the team
- QA’s work should be part of the DoD

What does QA do at the start of a sprint?

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
- 9. *Teams : Products***
10. Retrospective

How do Teams align to Products / Projects?

- Watch out for partial allocations
- Do team members have more than one Daily Scrum?
- Rather than 1 team per 1 product, try 1 team that supports multiple products

Teams to Products

- Bad
 - Billing System has a Billing Team
 - Accounting System has an Accounting Team
 - CRM system has a CRM Team
 - Website has a Website team
 - People are 50% allocated to multiple teams.

Teams to Products

- Better
 - Team A
 - Billing
 - Accounting
 - Team B
 - CRM
 - Website

Teams to Products

- Best
 - Team A and Team B are cross-functional
 - Either team can do tasks from Billing, Accounting, CRM, or Website

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
- 10. Retrospective**

Retrospectives are the curer of all ills.

Retrospectives help keep problems contained to (hopefully) a single Sprint.

Tip:
Watch for unscheduled work.

Tip:

Get people REALLY thinking by trying to recreate what happened day by day.

Tip:
If it gets heated, learn to say
“Ok...and what else?”

Tip:
Record the findings from the
Retrospective

Tip:
Review the notes from
previous Retrospectives.

Top 10 Ways

1. What is your DoD?
2. People Skills
3. Credibility
4. What is your Sprint Goal?
5. Do less.
6. Emergent Architecture
7. Good programming & testing practices
8. “Kick it over the wall to QA”
9. Teams : Products
10. Retrospective

What can you do to help yourself?

Watch for burnout.

You can't fix everything.

Do whatever it takes to get yourself
to believe in self-organization.

But don't let your team walk all over
you.

Some teams will use
self-organization against you.

“...but let’s remember here.
This isn’t a * & ^ % \$ # democracy.”
-David Starr

How do you help the
Product Owner?

Remind them of how important they are to the success of the project.

Remind them that they need
to stay engaged.



LARElekman

Ketterän kehityksen kouluttaja ja valmentaja

Product Owner Checklist – November 2013

“This brief checklist helps you remember the most important things to become a good Product Owner.”

As the Product Owner you are responsible for maximizing the value of the product and the work of the Development Team. Your way of doing this may vary across organizations, Scrum Teams, and individuals. This checklist is therefore only an example.



<http://productownerchecklist.org>

An Example Checklist for Product Owners

Lare Lekman | Revised in November 2013 | productownerchecklist.org

Product Vision

- I have a product vision (created with customers, end users, and investors, when possible)
- I can answer to questions about the product vision and business model in a concise and meaningful way
- I have a short tagline for the product vision, for example "1,000 songs in your pocket" (that is easy to remember and use) to communicate the essence and value of the product release.

Stakeholders

- I understand the needs of my stakeholders (for example customers, end users, and investors)
- I communicate regularly with my stakeholders to understand their needs and to manage their expectations
- I can answer to questions about how each product backlog item will generate value for the stakeholders
- I am motivated to work as a Product Owner, and make sure I have the stakeholders' mandate
- My forecasts to stakeholders are based on development team's measured velocity or throughput

Product Backlog

- I have a product backlog
- I have a mandate to make decisions about the product backlog
- I update the product backlog at least before each sprint planning meeting
- The product backlog items are ordered (based on value, risk, work estimates, dependencies)
- The product backlog items are clearly expressed and more detailed towards the top
- The product backlog is accessible to all scrum team members

Any last questions?

Thanks.

benday@benday.com | www.benday.com

