Effective ways to model requirements for successful software products



Cristiano Spadaro Product Manager and QA Manager

This presentation has been modified from its original version to protect IP-sensitive content





- You should leave the room with an understanding of:
 - The main challenges that a software (enterprise) product business typically has.
 - 2. Why requirements are so critical for the business.
 - 3. What are good ways to model requirements for products that are more likely to be successful than others.



The Market Problems. Buyers. Perception





Have you ever heard of...

- …"The customer said that the feature does not resolve his problem. But we did what he asked for!"
- What's the difference between a requirement and a specification?
- Who writes requirements?
- Who writes specifications?



Have you ever heard of...

• Developers complain: I cannot program to these requirements.

- Product Manager says: "Our developers wrote features that didn't do what the customer need!"
- Developer: "don't tell me how to do my job"



Have you ever heard of...

• Deev: This is NOT A REQUIREMENT!

 Many: How much detailed documents developers need to build successful products?

Developing tech products is hard



- From ideas to a shipped product there is a LOT of work
- Uncover the **Problem** (Need)
 - Problem must be pervasive AND urgent, AND there are people (buyers) willing to pay for getting it resolved
- Problem needs to be communicated and internalized by stakeholders (product teams)

• Then, we need to be able to **build** a solution.

- Developers need context, intent, and a level of detail so that they can analyze, feed back, provide estimates and build innovative solutions that can be tested.
- From Problem to a Built Product there is a LOT fo work and Requirements play the most important role

From research to shippable code







• Functional Requirement

 Specifies something that the delivered system must be able to do.

Non-functional requirements

Usability, availability, reliability, supportability, testability and maintainability,

•••



Requirements (as The Tradition tells us)

- Functional requirements describe the functionality that the system is to execute.
- Non-functional requirements describe characteristics of the system that the user cannot affect or (immediately) perceive.
- Constraint requirements impose limits upon the design alternatives or project/process operations.

Good requirements must be... learning solutions

- Unitary (Cohesive) The requirement addresses one and only one thing.
- **Complete** The requirement is fully stated in one place with no missing information.
- **Consistent** The requirement does not contradict any other requirement and is fully consistent with all authoritative external documentation.
- Non-Conjugated (Atomic) The requirement is *atomic*, i.e., it does not contain conjunctions.
- Traceable The requirement meets all or part of a business need as stated by stakeholders and authoritatively documented.
- **Current** The requirement has not been made obsolete by the passage of time.
- Feasible The requirement can be implemented within the constraints of the project.
- Unambiguous The requirement is concisely stated without recourse to technical jargon, acronyms. It expresses objective facts, not subjective opinions. It is subject to one and only one interpretation.
- Mandatory The requirement represents a stakeholder-defined characteristic the absence of which will result in a deficiency that cannot be ameliorated.
- Verifiable The implementation of the requirement can be determined through one of four possible methods: inspection, demonstration, test or analysis.

Nothing wrong here either, but this is not a comprehensive list! Source: wikipedia

We all use artifacts for communicating requirements



- Product Requirements Specification
- Product Requirement Document
- Product Specification Document
- Market Requirement Document
- Business Requirements Document

Here we start getting really wrong ...



Dilemma: How much detail?

- Artifacts are never enough
- There is simply NO way anybody can provide all details for a product that solves a problem end-to-end
- Discuss collaboratively (no PRD's thrown across the wall)
- Refer to Personas, Business, Market (not egos)

How much detail?



- PdM must transfer domain knowledge
- PdM must represents the customer
- Software factories need details (no domain knowledge)
- Effective dev teams *produce* details by using an iterative process using product managers domain knowledge
- Product manager represent the customer

Requirements & Design Specs



Requirement – PdM (The What)

• Vision

- Persona
- Problem
- Use scenario
- Market evidence
- Impact
- Priority

Plus: Constraints from Boundaries (e.g. Security, Deployment)

Specification – PD (The How)

- Overview
- Wireframes, Flow chart
- Use cases
- User interaction
- UI

- Prototypes / models
- Error conditions (what are the conditions)
- Document open issues
- Link to requirements

Requirements are implementation free!



Roles and deliverables

Product Manager	Finds and quantifies market problems , articulating them in the form of	Requirements Guidance	
Product Designer	Describes the approach to solving the problem in the form of	Conceptual design Mockups Design (Functional) Specifications	
Product Developer	Fully describes how the functional specification will be implemented in the form of (and then implements it)	Technical specifications Coded solutions	

Requirements...



• Describe a user (persona) interacting with a product to resolve a problem

As a *<Persona>* I want to *<Intent>* so that *<Valuable>* As a technical reviewer

- I want to access the information I am asked to review in an univocal manner. I don't want to fish for the files I need every time.
- So that I h to review submissio Suppliers for a prototype run. He

suppliers for a prototype run. He need to generate automatically specific file formats (PDF and IGES) from his many 3D CAD models so that he can save time and distraction from his core engineering tasks.



http://www.pragmaticmarketing.com/publications/magazine/5/4 /the-power-of-the-persona

PERSONAS!

© eXact learning solutions 2010 • Page 17

Whom to Please?



- Sales process (an easier to sell product)
- Users: who are they?
- Buyers (want an online option at least, demand no-installation tools for casual users etc)



- We have found several types, regardless of our solution.
- Looking at the customer we already have is misleading
- It's the ones that we haven't caught that matter (because we want to catch them with something that please them
- Our current tools do not please them (and the buyers)
- So, we have looked at a spectrum



A REAL EXAMPLE

© eXact learning solutions 2010 • Page 20



Primary User Persona: Jeff

0	

- I am a SME first.
- I like creating learning materials.
- I have no clue what a CMS or DITA are and I don't want to know. Just let me compose my contents quickly.
- Jeff is a SME and content author in a large company in the engineeringsector. His company has a small but dedicated documentation organization based on Instructional designers, tech writers, SME's and training managers. The quality of content they produce is not high, but they significantly value quantity, production speed and cost reduction.
- Jeff works as a SME and content author for most of this time.
- He is able to use rapid authoring such as Power Point or MS Word, which he has been using for years now.
- Sometimes he is required to create content "quickly" to chase emergencies or criticalities for a new financial program.
- Jeff is not a computer geek at all, but he has a decent understanding of MS Office Tools and Windows.
- He is able to learn the basics of a new online authoring tool in 2 hours and be able to create and package a course. He was required to do so during the evaluation of a new authoring tool

Personas









Jeff the SME Novice User

My job is creating contents, not fighting with software Michelle the ISD Power user

My job is creating quality and pedagogically-sound content Paul the demo guy eXact Power user

My job is make great product demos that create the perception that our product is the right one for solving the evaluator's problem

Primary Personas



My job is creating contents, not fighting with software



- SME
- Required to produce quick training jobs
- Part Time in the training dept
- No complex tools please (can learn simple ones)
- May or may not work with Michelle



- ISD and training Manager
- Full time in the training dept
- Authoring Tools Expert
- Tools: xxxx/YYYY

Jeff is the Primary User for the our new

Primary?



- Primary persona means that we can make happy secondary personas unless we go against the primary Persona
- It's where the focus is
- Helps prioritizing

Primary User Persona: Matt

Mechanical Design Engineer

His company produces Tablets (eBook-like)

This new product will get in my way! I don't want and need to use it.



I am an engineer and want to be an engineer!

Matt's Business Goals:

• Develop products that meet functional requirements, delivered on time, maximize part reuse, meet cost targets and regulatory compliance etc.

Matt's Personal Goals

- Jeff <u>thrives on designing parts</u>, resolving all sort of technical challenges and conflicting constraints, and he excels at that.
- Matt wants to get work done more easily. Matt has a family and even a life. He wants to get the job done during "normal hours" to spend time with his <u>family</u>.
- Matt <u>hates</u> spending time in activities that not only are <u>tedious</u>, <u>labor-intensive</u> and <u>error-prone</u>, but that also can pose risks in accomplishing his assigned objectives.
- Matt is not familiar and non-versed in data crushing and cleanup through spreadsheets

Environment

• Team of 15 Mech engineers on MCAD and XDOC and 3 locations around the world (US, NL and Philippines)

A perfect day

• I resolve a design problem with a cool solution. It gets the product out back on schedule and my boss praises my deep technical skills. I go back home on time



Buyers and User Personas

• Buyer Persona

- Has a urgent business problem to solve
- Has budget and buying power

User Persona

- Uses the product everyday
- Even if she does not want to
- May not be aware of buyer problems

Technical reviewer Persona

Reviews standards, compliance and provide recommendations

Personas?



- Well-described, archetype of a user group
- Must feel like a real person to everyone in your company.
- Coworkers will talk in terms of Matt, not like "the system must do this and that"



- When the company internalizes the personas and how they relate to the product, these personas help everybody in any function
 - Superfluous information and overspecifications can be eliminated
 - Features are prioritized according to the persona's values
 - Product messaging can focus on the product's value to the buyer persona

Power of Personas



- The dev team...
 - Understand requirements with less detail and specification
 - Make good, reasonable implementation decisions independently
 - Raise valid concerns and opportunities
 - Stay focused on the real requirements and avoid being sidetracked by edge cases



BACK TO REQUIREMENTS

© eXact learning solutions 2010 • Page 30

Requirements...



• Describe a user (persona) interacting with a product to resolve a problem

As a *<Persona>* I want to *<Intent>* so that *<Valuable>* As a technical reviewer

- I want to access the information I am asked to review in an univocal manner. I don't want to fish for the files I need every time.
- So that I h to review submissio Suppliers for a prototype run. He

suppliers for a prototype run. He need to generate automatically specific file formats (PDF and IGES) from his many 3D CAD models so that he can save time and distraction from his core engineering tasks.





- Requirements are described by the point of view of a persona
- Includes the business benefit by focusing on value
- Emphasize Intent and value over mandatory / to be met
- Provide enough detail for making a reasonably low risk effort estimate
- Includes acceptance criteria by which we agree that it is "done".
- Requirements are not work items (DB stories or UI stories).
 They are end-to-end and focus on intent and value

Requirement Example



• When publication fails the integration shall fetch an alert as soon as possible from the Auth tool and post it into an interim file. The product should automatically load the content of this file and send an email to the user. When clicking on the attachment, a browser window will open with a list of all alerts for that user with flexible options for sorting. The user can click on any of them. See attached for screen example

Bad example





• The PdM here is asking developer to guess

Better example





Too many alerts!

Throughout the day, Tina gets alerts on her console but she only needs to know about the ones requiring action on her part so that she does not miss important ones by browsing among hundreds of them.

Example



Alerts and Notifications



Todd needs to publish his designs into the repository to make them available for other departments in his company. This process is manual, error prone and time consuming

Whatever happens to the publication process, Todd needs to be immediately alerted if there is a something wrong, notified of what is wrong and where the problem might be so he can plan for mitigating actions to keep the delivery on track and complete the publication within 4 business hours

- 1. Todd is notified that the publication has started, as soon as this event occurs, within his Authoring System (MCAD)
- 2. Todd is notified that the publication has reached its destination in the target repository
- 3. Whatever happens to the publication, there is a human readable alert that tells Todd and Tina (1) What failed (2) What is the cause of that failure (3) Where is the problem (4) What is recommended for resolving these situations. In other words, there is a precise and forthcoming way to help Todd and Tina.
- 4. When Todd is notified of a problem, he is able to troubleshoot, fix and restart the publication without having to reconstruct the publication process from scratch
- 5. Todd does not need to rely on a log file to troubleshoot (Jeff can use a UI designed for that purpose)
- 6. Todd is notified when the publication is successful the package has been successfully processed

Clarify with context



- Developers need more information. Right
- Product managers develop more details. Wrong
- Providing context via persona, problem and use scenario goes a long way.
- A PdM will never be able to figure out all details

The Root Cause



- What is the root cause of the problem?
- Always try to spend time to identify the root cause
- The more you get to the root cause, the more innovative your solution will be
- If you discuss this and hear 5 why's then you are getting to the root

A User Story A Requirement



- Provide enough detail to make a reasonably low risk estimate of how long the requirement will take to implement.
- Describes a requirement by the point of view of a persona
- Includes the business benefit/context. Focus on value
- Includes a set of criteria by which we all agree that it is "done".
- Used instead of a large requirements document.
- Written <u>by the "customers</u>"
- About three sentences of text written by the customer in the customers terminology without techno-syntax.



DETAILS ?

Requirements and Details









- In any discussion, especially with customers, mockups are extremely effective
- Can be paper mockups, electronic ones etc
- You may get surprised to see how effective simple paper mockups are

Example: Description





Todd wants to automatically generate PDF and IGES files from YYYYY documents

Business context and usage scenario (may be "inherited" from top)

Todd wants to share his design (in form of a sub assembly) with suppliers for a prototype run. He need to generate automatically specific file formats (PDF and JPEG) from his MCAD documents (part, assemblies and drawings) so that he can save him time and distraction from his core engineering tasks.



Example: Discussion/Discovery





Automatic generation of PDF/IGES files from within XXXXCAD

Business context and usage scenario (may be "inherited" from top)

Todd wants to share his design (in form of a sub assembly) with suppliers for a prototype run. He need to generate automatically specific file formats (PDF and JPEG) from his MCAD documents (part, assemblies and drawings) so that he can save him time and distraction from his core engineering task

Questions

How about part configurations? How about Assembly configurations? Does he need to use other formats (like STEP and IGES) How about Native files? How does Todd initiate this generation?

Difficult use cases

Publishing drawings that include views belonging to different configurations Not sure how to handle assembly configurations



Description updated - more detailed



Example: Discussion/Discovery









Effort: **3-6** Priority: PO



Automatic generation of PDF/IGES files from within XXXX

Business context and usage scenario (may be "inherited" from top)

Todd wants to share his design (in form of a sub assembly) with suppliers for a prototype run. He need to generate automatically specific file formats (PDF and IGES) from his MCAD documents (part, assemblies and drawings) so that he can save him time and distraction from his core engineering tasks.

Todd can generate distribution file formats for each part configuration

Todd is able to designate specific distribution file formats , including native ones, as per the ones supported by the "Save As" functionality in MCAD

Design specs link



Difficult use cases

How about Assembly configurations ? A. are hard to support, but there is an alternative solution: by default, the "in-use" or "active" configuration will be used for generating the DF.

Publishing drawings that include views belonging to different configurations. Still Open.

Dev and PM and responsibilities learning solutions

Developers need:

- Personas (they may not be aware of this)
- Context
- The Intent (value)
- Scenarios
- Develpers will always have to make decisions.
- The more they know about the problem, the better the decisions
- The more innovative the solution
- They have the right to be helped on decisions for trade offs. They will make them anyway.
- Unambiguous and succint single statements of truth are the form they digest better.
- Solving problems. They excel at that
- Guidance, guidance, guidance on what?

Product Manager need:

- Timely Feedback
- Estimates (estimates are friends, not enemies)
- Risks (Risks are friends, not enemies)

The Product Manager



- Drives delivering value this is really the #1
- Clearly explain context, problem and value delivered by requirements (personas!). Include the "why important"
- Select requirement prioritization scheme that is adequate with the organization and the business
- Validate requirements timely and accurately by well defined (upfront) acceptance criteria and customer feedback
- Make decisions (make the call on behalf on the business)



Boundaries? Constraints?



Boundaries create constraints



Requirements and tasks

Requirement		Dev Task					
 Intent and Value (PM) Priority (PM) Design Specs (PD) Estimate (ENG) Right size for sprint (ENG) End to End? (PM) 			 Path to deliver that Story made of tasks Sets of tasks (ENG) Acceptance Criteria for each (ENG) 				
٠	 Acceptance Criteria (PM) 					S pecific	
		Independent				M easurable	
		Negotiable				R elevant	
		Valuable				Time boxed	
		Estimable					
		S mall					
		Testable					

Example of requirements (user stories



Suggested readings



Specs and requirements

 Pragmaticmarketing.com - On Reqs. and Specs.: The Roles and Behaviors for Effective Product Definition <u>http://tinyurl.com/y9r6omr</u>

Personas

- Pragmaticmarketing.com The power of personas <u>http://www.pragmaticmarketing.com/publications/magazine/5/4/the-power-of-the-persona</u>
- Alan Cooper The inmates are running the asylum http://tinyurl.com/yc8x2ja

References



- productpersonas.com
- joelonsoftware.com
- pragmaticmarketing.com
- Rally software
- Sinan Si Alhir
- Allan Cooper
- Ken Schwaber
- Mike Cohn
- Mary and Tom Poppendiecks
- Dean Leffingwell
- Top Peters
- Peter F Drucker
- Feature plan
- Jeff Patton