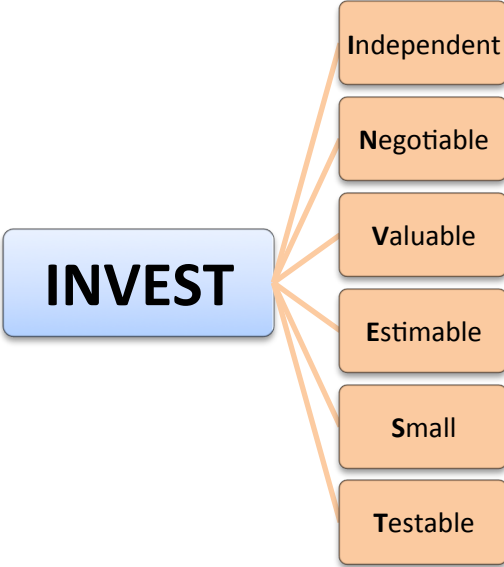
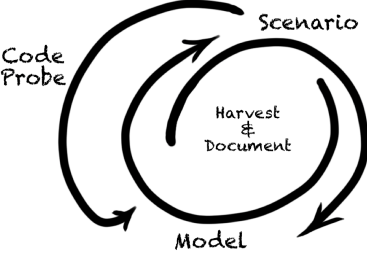

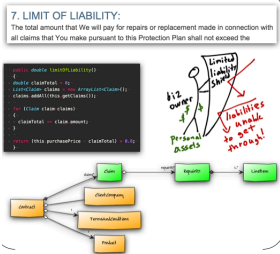
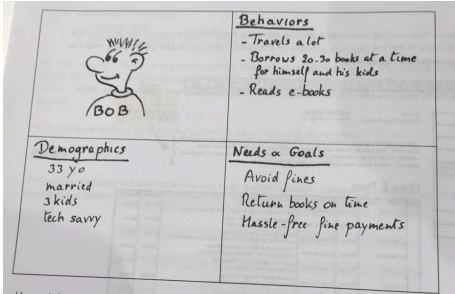


# User Stories Reference Guide

What is a story?	Characteristics of useful stories	Why stories?	
<p>A story describes <i>product functionality</i> from a <i>customer's perspective</i>. It is a <b>collaboration tool</b> - a <i>reminder to have a conversation</i> about what the customer needs so the team can design it well &amp; deliver it quickly.</p>		<ul style="list-style-type: none"> <li>• Support satisfying the customer through early and continuous delivery of valuable software</li> <li>• Shifts focus from writing to talking</li> <li>• Involve users, domain experts and stakeholders/customers in a creative, iterative, collaborative design process (using DDD &amp; UXD)</li> <li>• Describe concrete business reference scenarios, understandable by all in a common, shared language</li> <li>• Team designs iteratively and delivers incrementally</li> <li>• Level of detail based on implementation horizon</li> </ul> 	
<p><b>The goal of using stories</b></p>		<p>Talk together about concrete, realistic examples to <b>build shared understanding</b>. The focus should be on <i>telling stories, not writing them</i>. Look for risks and rewards in the stories being told.</p>	
<p><b>Discover and organize in a story map</b></p>		<p>Start with each persona's needs and goals, and build a story map to understand <i>what is needed and why, not how</i>. <i>Understand whether a goals &amp; tasks described by a story is core domain, or supporting or generic subdomain</i>.</p>	
<p><b>Combine with domain model exploration</b></p>	<p><b>Enrich with design artifacts</b></p>	<p><b>Build around personas</b></p>	
<ul style="list-style-type: none"> <li>• Avoid design fragmentation when splitting stories by doing model exploration when needed</li> <li>• Explore business domain with EventStorming</li> </ul>  <ul style="list-style-type: none"> <li>• Exercise and refine the ubiquitous language</li> <li>• Harvest and document key examples as reference scenarios</li> </ul>	<ul style="list-style-type: none"> <li>○ Design documents/sketches</li> <li>○ Proof of concepts, code probes</li> <li>○ Photos, screenshots, mockups</li> <li>○ Examples of inputs and expected results</li> <li>○ Business rules, data dictionaries, use cases, glossaries, diagrams, spreadsheets</li> </ul> 	<ul style="list-style-type: none"> <li>• Personas grow empathy and deeper understanding of your customers/users as <i>real people with real needs</i>.</li> <li>• Enable you to better prioritize stories</li> <li>• Used extensively in usage-centered design</li> </ul> 	



**virtualgenius**  
leading by design

Independent <sup>1</sup>	Estimable	Three common techniques for splitting stories <sup>2</sup>
<ul style="list-style-type: none"> <li>Identify dependencies – they make prioritizing and planning more difficult</li> <li>“Slice the cake” – each story must have a little from each system layer</li> </ul>	<ul style="list-style-type: none"> <li>Focus on <i>delivering</i>. Don’t get hung up on estimation or traceability</li> <li>Understand <i>relative effort/uncertainty/risk</i></li> <li>Factors that make estimation hard: <ul style="list-style-type: none"> <li>Developers lack domain knowledge</li> <li>Developers lack technical knowledge</li> <li>Story is too big</li> </ul> </li> </ul>	<p><b>1. Conjunction Slicing</b></p> <p><b>Operations (eg. CRUD) – supporting subdomain? (unlikely to be in core domain)</b>  As a content publisher, I can manage my account ...I can edit my account settings  ...I can sign up for my account</p> <p><b>Data Variation – supporting? maybe simple case is good enough</b>  As a content manager, I can create news stories ...in English  ...in Japanese</p> <p><b>Data Entry Methods – collaborate with UXD</b>  As a traveler, I can search for flights between two destinations ...using simple date input  ...with a fancy calendar UI</p> <p><b>Major Effort – supporting subdomain? maybe only the simple case is good enough</b>  As a account holder, I can pay for my flight with VISA, MasterCard, Diners Club, or American Express  ...I can pay with one credit card type (of VISA, MC, DC, AMEX)  ...I can pay with all four credit card types (VISA, MC, DC, AMEX)</p> <p><b>Workflow Steps – potential need for modeling here if core domain</b>  As a content manager, I can publish a news story to the corporate website  ...I can publish a news story directly to the corporate website  ...I can publish a news story with editor review  ...I can publish a news story with legal review</p> <p><b>2. Rule Relaxation</b></p> <p><b>Business Rule Variations – potential need for modeling &amp; UXD here</b>  As a traveler, I can search for flights with flexible dates ...as “n days between x and y”  ...as “a weekend in December”</p> <p><b>Simple/Complex – is it complex stuff within core domain? Otherwise avoid adding complexity</b>  As a user, I can search for flights between two destinations ...specifying a max number of stops  ...including nearby airports</p> <p><b>3. Design Probes</b></p> <p><b>Break Out a Code Probe or Spike/POC – in core domain (avoid otherwise)</b>  As a traveler, I can pay by credit card ...Investigate credit card processing  ...Implement credit card processing (as 1 or more stories)</p> <p><b>Defer Performance – pay close attention to aggregate boundaries</b>  As a user, I can search for flights between destinations ...(slow – just get it done, show “searching...”)  ...(in under 5 seconds)</p>
Negotiable	Small	
<ul style="list-style-type: none"> <li>Stories are not: <ul style="list-style-type: none"> <li>Written contracts</li> <li>Requirements</li> </ul> </li> <li>Don’t include all details, otherwise gives impression of: <ul style="list-style-type: none"> <li>false precision or,</li> <li>completeness</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Small stories for implementing in the near future, and higher-level (larger) stories for further out.</li> <li>Large stories (aka “epics”): <ul style="list-style-type: none"> <li>Separate what is core from what is supporting and generic</li> <li>Hide assumptions that should be made explicit</li> <li>Hard to estimate and to plan</li> </ul> </li> </ul>	
Valuable	Testable	
<ul style="list-style-type: none"> <li>Identify if story relates to core, supporting or generic subdomain</li> <li>Stories must be valuable either to users/domain experts</li> <li><i>Finish stories early and often</i></li> <li>Customer/Product Owner orders stories in story map according to relative business value</li> </ul>	<ul style="list-style-type: none"> <li><i>Use specific, concrete, actual business scenarios</i> for modeling and testing</li> <li>Drive important design decisions test-first with unit tests</li> <li>Specify <i>acceptance criteria</i>: demonstrate a story meets expectations</li> <li>Where possible, automate acceptance tests</li> </ul>	

<sup>1</sup> User stories content adapted from Mike Cohn, *User Stories Applied* (Addison Wesley: 2004).

<sup>2</sup> Adapted from *Story Splitting Cheat Sheet* by Richard Lawrence of Humanizing Work (<http://www.richardlawrence.info/2009/10/28/patterns-for-splitting-user-stories>)

<sup>3</sup> DDD Whirlpool adapted from <http://domainlanguage.com/ddd/whirlpool>