Backlog Confessions:
Technical User Stories

Candase Hokanson, PMI-ACP

Senior Product Manager, Seilevel
# OUR AGENDA

<table>
<thead>
<tr>
<th>TOPIC</th>
<th>YOU WILL LEARN</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>What is a technical user story?</strong></td>
<td>The difference between “normal” user stories and technical user stories</td>
</tr>
<tr>
<td>How do we find technical user stories – System to System?</td>
<td>How to derive technical user stories from Ecosystem Maps</td>
</tr>
<tr>
<td>How do we find technical user stories – within one system?</td>
<td>How to derive technical user stories from System Flows</td>
</tr>
<tr>
<td>Conclusions/Q&amp;A</td>
<td></td>
</tr>
</tbody>
</table>
What backlog confessions does your team have?
Common Confessions

- Making the system the user in a typical user story
- Using placeholder stories or spikes to account for unexpected work
- Adjusting story points at the end of the sprint to match actual hours worked
Typical User Stories

Stories describes the **functionality** that will be delivered by the project

**Format**

As a [role]
I want to [do something]
So I can [achieve outcome]

**Example**

As an Amazon customer
I want to be able to search Amazon by keyword
So I can quickly find what I want to buy
Technical Story Format

In order to [deliver some benefit]
[these people] **Will need**
[this function]

In order to [deliver some benefit]
[this system] **Must**
[deliver this function]
Why does the difference matter?

Sprint carry over

Losing sight of end user value

Systems as Users?
A Different Structure for Technical Products

EPIC

FEATURE

Still crossing multiple products/feature teams

STORY

Typically contained within one feature team-written as normal user stories for the high level functionality to enable

Still written small enough to fit in one sprint, but now as technical user stories fulfilling a feature
# OUR AGENDA

<table>
<thead>
<tr>
<th>TOPIC</th>
<th>YOU WILL LEARN</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is a technical user story?</td>
<td>The difference between “normal” user stories and technical user stories</td>
</tr>
<tr>
<td><strong>How do we find technical user stories – System to System?</strong></td>
<td>How to derive technical user stories from Ecosystem Maps</td>
</tr>
<tr>
<td>How do we find technical user stories – within one system?</td>
<td>How to derive technical user stories from System Flows</td>
</tr>
<tr>
<td>Conclusions/Q&amp;A</td>
<td></td>
</tr>
</tbody>
</table>
Meet Blue, the SeiCars Credit Product Owner
SeiCars Credit Process Flow

1. Salesperson collects info from customer and enters in UI

2. Customer found at bureau?
   - Yes: System searches for the customer at the bureau
   - No: 3. Salesperson gathers rest of info from customer

3. Salesperson gathers rest of info from customer

4. Decision?
   - Approved: 5. Salesperson tells customer he is approved
   - Declined: 9. Salesperson tells customer he is not eligible

5. Salesperson tells customer he is approved

6. Customer finishes loan paperwork for car

7. Customer drives off lot happy

8. Salesperson verifies info from customer

10. Salesperson offers other financing options
Each interface is typically a technical story
Flows across systems are typically features
In order to ensure terrorists do not get on the plane, the TSA passenger system MUST check the passenger manifest with the known persons list.

In order to give up to date status for a customer’s luggage to the customer, the luggage tracking system MUST associate the scanned luggage with the flight reservation for the customer.

In order to reward our frequent fliers, the Frequent Flyer system MUST associate each flight reservation with the flier account.
Derive technical user stories from the ecosystem map provided
Technical Story Format

**In order to** [deliver some benefit]

[these people] **Will need**

[this function]

**In order to** [deliver some benefit]

[this system] **Must**

[deliver this function]
As a **SeiCars Customer,**
I want to be able to apply for a credit loan from the dealership,
So that I can get a better rate than with a 3rd party lender.

In order to manage the customer’s loan, the loan system **MUST** receive the accepted/approved loan terms from the credit system.

In order to determine if the customer is credit worthy, the credit system **MUST** be able to connect with the credit bureau for credit reports.

In order to meet compliance regulations with regards to consumers, the credit system must store all credit application data.
Lessons Learned:
• Think beyond just connections to the data underlying it
• Always ask why we need to move the data between the systems- what value is it bringing?
<table>
<thead>
<tr>
<th>TOPIC</th>
<th>YOU WILL LEARN</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is a technical user story?</td>
<td>The difference between “normal” user stories and technical user stories</td>
</tr>
<tr>
<td>How do we find technical user stories – System to System?</td>
<td>How to derive technical user stories from Ecosystem Maps</td>
</tr>
<tr>
<td>How do we find technical user stories – within one system?</td>
<td>How to derive technical user stories from System Flows</td>
</tr>
<tr>
<td>Conclusions/Q&amp;A</td>
<td></td>
</tr>
</tbody>
</table>
Shows the processes and decisions within a specific system

Depending on the level, the flow is a feature and each step becomes a technical user story
System Flows Have Levels: L1, L2, and L3

Level 1 (L1)  
ID = 1  
L1 Step 1 → L1 Step 2 → L1 Step 3 → L1 Step 4 → L1 Step 5 → L1 Step 6 → L1 Step 7

Level 2 (L2)  
ID = 1.1  
L2 Step 1 → L2 Step 2 → L2 Step 3 → L2 Step 4 → L2 Step 5 →  
L2 Process Flow

Level 3 (L3)  
ID = 1.1.5  
L3 Step 1 → L3 Step 2 → L3 Step 3 → L3 Step 4 →  
Step 1.1.5.2

Step 1.1.5
As a paying Airline Customer, I want to be able to retrieve a past flight reservation, so that I can plan for transportation to the airport.

In order to not waste the customer’s time, The flight system MUST validate the confirmation code matches the database.

In order to show the flight reservation, The flight system MUST search by confirmation code in the reservation database.
Derive technical user stories from the System Flow provided
Technical Story Format

In order to [deliver some benefit]
[these people] **Will need**
[this function]

In order to [deliver some benefit]
[this system] **Must**
[deliver this function]
As a **SeiCars Risk Auditor**, I want to force a search for the customer at the bureau prior to pulling the credit report, so that I can know the right credit is being used. In order to show the search results to the salesperson, the credit system MUST extract data from the bureau response and format in a way the UI can display.

In order to not waste the customer’s time, the credit system MUST validate that the request will meet the bureau minimum search requirements. In order to show the search results to the salesperson, the credit system MUST send the search parameters to the bureau and receive the response.
Lessons Learned:

- Depending on level, each step becomes a technical user story
- Think about decisions and combinations—those become Acceptance Criteria
## Our Agenda

<table>
<thead>
<tr>
<th>Topic</th>
<th>You Will Learn</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is a technical user story?</td>
<td>The difference between “normal” user stories and technical user stories</td>
</tr>
<tr>
<td>How do we find technical user stories – System to System?</td>
<td>How to derive technical user stories from Ecosystem Maps</td>
</tr>
<tr>
<td>How do we find technical user stories – within one system?</td>
<td>How to derive technical user stories from System Flows</td>
</tr>
<tr>
<td>Conclusions/Q&amp;A</td>
<td></td>
</tr>
</tbody>
</table>
Technical products are different

- Sometimes using a different type of user story to keep the team focused on delivering value is a good thing.

Use what you already have!

- Visual models such as Ecosystem Maps and System Flows can help you find these technical user stories.

Don’t forget about your other confessions!

- Understanding what your team is doing ‘differently’ and why is important to being agile.
Questions?

Candase Hokanson
Seilevel

Candase.hokanson@seilevel.com

www.seilevel.com
@Seilevel
www.facebook.com/Seilevel