SRE & Product Management

How to level up your team (and career!) by thinking like a product manager

Jen Wohlner, senior product manager at Livepeer
Linkedin: Jen Wohlner, IG: @jensface

October 3, 2019, SRECon EMEA, Dublin
Sometimes SREs must do product management.
You: Jen, what?
Jen: Yes.
# Product vs Program Management

Some key differences

<table>
<thead>
<tr>
<th>Product</th>
<th>Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prioritize <strong>ideation</strong> to impact business</td>
<td>Prioritize <strong>implementation</strong> to impact business</td>
</tr>
<tr>
<td>Closer to <strong>users</strong></td>
<td>Closer to <strong>engineers</strong></td>
</tr>
<tr>
<td>Often focus on <strong>growth of product</strong></td>
<td>Often focus on <strong>health of tech stack</strong></td>
</tr>
</tbody>
</table>

**What and why?**  
**When and how?**
When do SREs do product management?
All the time!
Common SRE product work

You do these things.

- proof of concept experiments
- creating alert plans
- rolling out a new deployment process
- stuff with UIs, like Prometheus dashboards
- internally negotiating and creating SLAs with your team and other teams
- and more!
Um, but why should you listen to me?
Listen to me.

Please.

- BuzzFeed's infrastructure/SRE TPM
  - Initially focused on products to escape toil
  - Struggled to create broadly useful data and platform infrastructure

- Failed. Then learned from lead product designer

- Took learnings to LinkedIn and Fastly and iterated

- Now at Livepeer as a product manager working on video and CDN infrastructure
Product management is about meeting needs.
Product management is about **EMPATHY**.
It doesn't matter WHO does product work as long as it gets done.

— me, but don’t tell my PM colleagues
Let your inner product manager shine.
1 Know your users and talk to them.
1. Know your users and talk to them.

- Most users of your SRE products are other engineers at your company.
- Just because you sit near them or have a similar job title … **do not assume** you know how they work or use your product or service.
User Interviews

*Do them!

- 30 mins
- Interview no more than 2 people at a time
- Interview team leads and managers separately from individual contributors (ICs).
  - They have different jobs and use tools differently.
- Feel free to bring one teammate to help you interview.
- Share user interview findings with your team.
  - presentations
  - video watching party
  - quotes
Ask non-leading questions.
2. Ask non-leading questions

**Examples**

<table>
<thead>
<tr>
<th>Leading</th>
<th>Non-leading</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Chef</strong></td>
<td>Do you have problems with Chef?</td>
</tr>
<tr>
<td><strong>A spreadsheet documenting a major compliance effort</strong></td>
<td>Do you wish correlating estimated resource weeks to priority happened automatically?</td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td>Why does your team do so few deploys? And why are they always huge?</td>
</tr>
</tbody>
</table>
3 Prototyping sprints
3. Prototyping sprints are wonderful!

- One week, one focus for all involved
- Narrow in on core users (some are more important than others)
- Limited time forces limited scope
- Team bonding
Add user-centric goals to roadmaps.
4. Add user-centric goals to roadmaps

**Simple example**

<table>
<thead>
<tr>
<th>Project</th>
<th>Stakeholders</th>
<th>User-centric Goal</th>
<th>Deliverables</th>
<th>Deliverable Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make tacos</td>
<td>family members</td>
<td>feed the family cheaply and quickly on a weeknight</td>
<td>buy tacos shells, ground beef and seasoning</td>
<td>Done</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>stretch goal: find affordable side, like refried beans</td>
<td>Won't do</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>cook food</td>
<td>In progress</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>serve food</td>
<td>Possible delay</td>
</tr>
<tr>
<td><strong>Higher priority projects above</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Lower priority projects below</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Make dessert</td>
<td>family members</td>
<td>feed the family a cheap post-dinner sweet</td>
<td>defrost cheesecake</td>
<td>At risk</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>serve cheesecake</td>
<td>At risk</td>
</tr>
</tbody>
</table>
Roadmaps

Do you even roadmap, bro?

- Including more than just technical deliverables helps define scope.
- User-centric goals answer what and why?
- Deliverables answer how and when?
- Shorter roadmapping intervals (ex. 6 weeks) allow you to pivot from bad investments and sunk costs
### Breaking down a big project, engineering example

<table>
<thead>
<tr>
<th>Project</th>
<th>Stakeholders</th>
<th>User-centric goal</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>New CI pilot</td>
<td>Eng leads at company</td>
<td>Get early feedback on the new CI pipeline before general rollout to all of eng</td>
<td>User interviews &amp; share findings</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Architecture design doc</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Select one eng team for pilot</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Technical deliverable 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Technical deliverable 2 ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Migrate team’s services</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SRE &lt;&gt; pilot eng team retro</td>
</tr>
<tr>
<td>New CI general availability</td>
<td>Eng leads at company</td>
<td>Improve all engineers’ CI workflows</td>
<td>TBD: Transition all non-legacy applications with owners to new CI</td>
</tr>
<tr>
<td>New CI final phase</td>
<td>Eng leads at company</td>
<td>Reduce risks to the engineering org</td>
<td>TBD: Transition legacy and ownerless apps to new CI</td>
</tr>
</tbody>
</table>
How to use a roadmap

- Strict start and stop dates
- Add updates to the roadmap weekly or bi-weekly in dedicated meeting
- Team lead shares screen
- Update statuses line by line for incomplete projects and deliverables
- Be willing to occasionally move high priority projects down to low priority section
- At the end of the quarter or 6 week interval, hold a retrospective on roadmapping process, estimation and project execution
Roadmap note

The roadmap spreadsheet is an internal tool.

- To share project status, do not just share the spreadsheet.
- Make it easy for your manager, other teams, and stakeholders to understand your progress:
  - bi-weekly emails
  - monthly status meetings
Follow-up with users regularly
5. Follow up with your users regularly.

- Users’ needs and pain points change
- Follow-up with users after releasing a product or feature
- And at major development milestones
Now get started!
To hire or not to hire a product manager?

Stuff to consider.

- Is this an experiment to you? Because it’s a career for the new hire.
- Product management mentorship?
- Who has product management potential on your existing team?
SRE product manager characteristics

- Senior individual contributor (IC)
- Must be technical
- Must be a strong leader
- Must have excellent soft skills
- Very valuable to have direct SWE or SRE programming experience
Recap time
SRE & Product Management

1. User relationships! Know your users and talk to them.
2. Non-leading questions! Ask non-leading questions.
3. Prototyping sprints! Prototyping sprints are wonderful.
4. User-centric goals! Add user-centric goals to roadmaps.
5. Follow-up with users! Follow-up with users regularly.
Thank you

LinkedIn: Jen Wohlner

Instagram: @jensface

livepeer.org