

Product Management

---

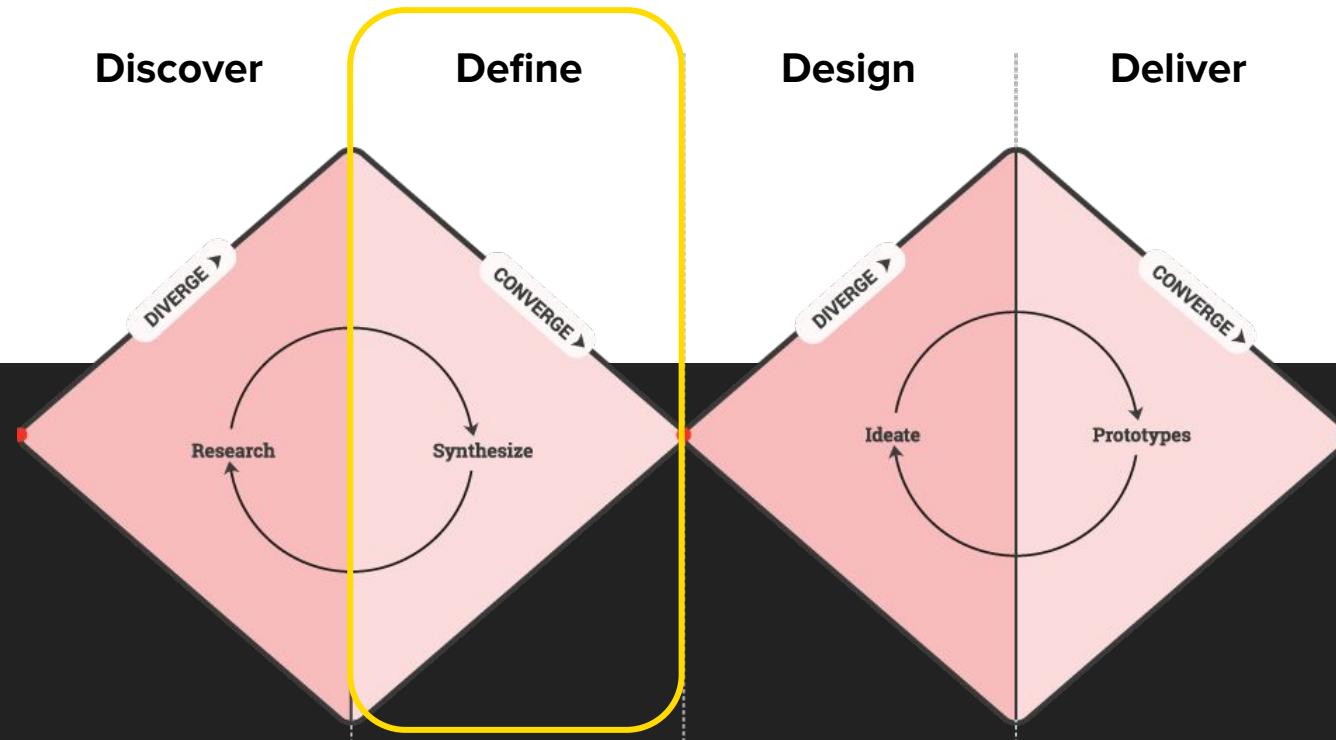
# Feature Prioritization

# Our Learning Goals

- **Estimate** the level of effort and customer value of a given feature in order to prioritize it.
- **Use** a product backlog to manage and track features and tasks.



# The Double Diamond





# Features: How Did We Get Here?

We've talked about features a few times already, so let's recap!

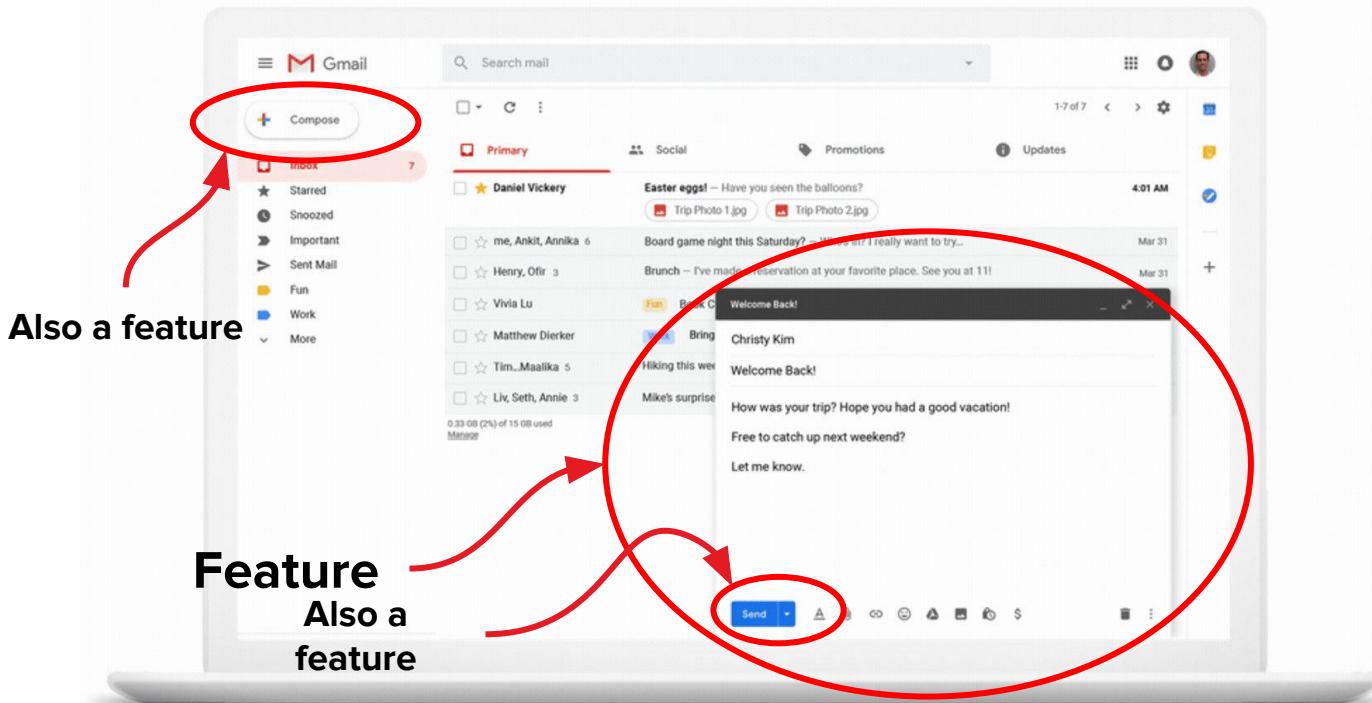
- How do features relate to your **problem statement and hypothesis?**
- How do features relate to **MVPs?**

# Features Are Jobs To Be Done





# Features Inside of Features Inside of Features



Product Management

---

# Prioritizing Features



# Why Prioritisation?

**Focus**

**Deep Understanding**

**Deep Value**



Your ability to **focus** your limited resources

... determines the level of **depth of quality** in your team's understanding and work

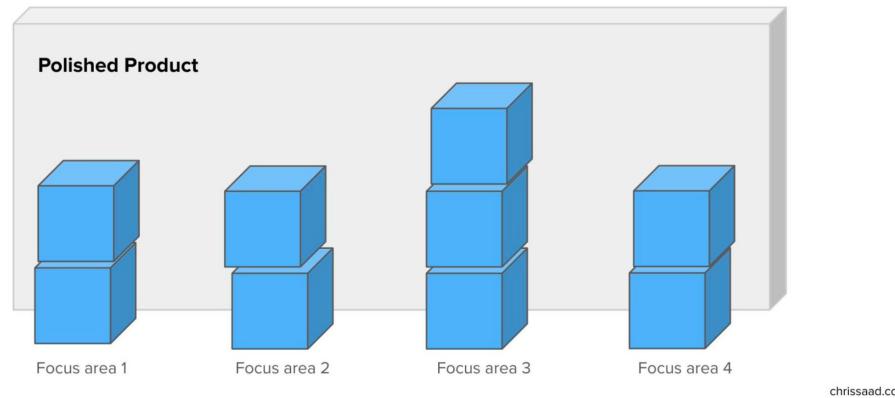
... which is the **critical foundation** to you delivering **deep value** to your customers and your business.

# Why Prioritisation?

POLISH REQUIRES FOCUS

## Why is focus important?

If the resources are stacked against too many focus areas, then you won't make it over the wall. It will take longer for **standardized, repeatable, scalable** market value to be created.



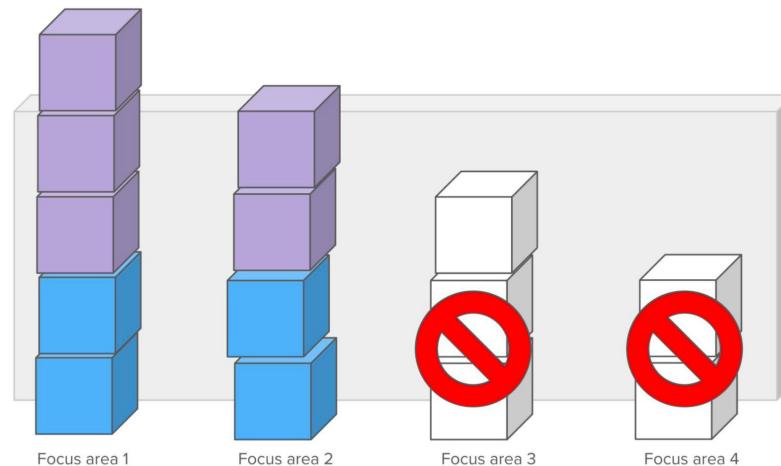
chrissaad.com

# Why Prioritisation?

PRIORITIZATION REQUIRES THAT YOU MAKE HARD CHOICES

## What are hard choices?

Hard choices means picking fewer focus areas and cutting others. This allows you to stack your resources to make it over the wall as quickly as possible.



chrissaad.com

# A thought from a smart, experienced PM



**Shreyas Doshi** • Following

Advisor to fast-growing startups. ex-Sripe, Twitter, Google, Yahoo.

1mo •

There are huge skill differences between

- A) understanding customer pain, and
- B) deciding if your product should even solve that pain, and if Yes,
- C) how to solve it in a way that creates major differentiation

A is easiest of these three. Many can do A well.

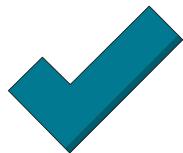
B should often be No. Most people cannot resist the temptation of saying Yes for B.

C is where fortunes are made. Few can do C consistently well.

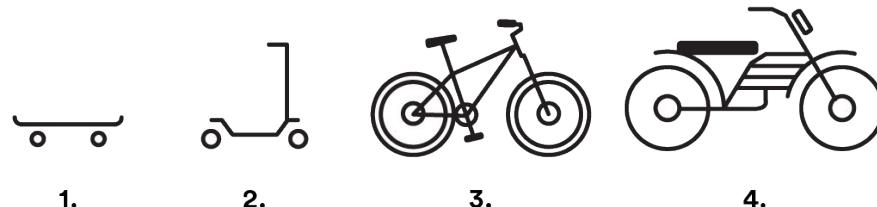
As a product person, it pays to get really good at A, B, \*and\* C.



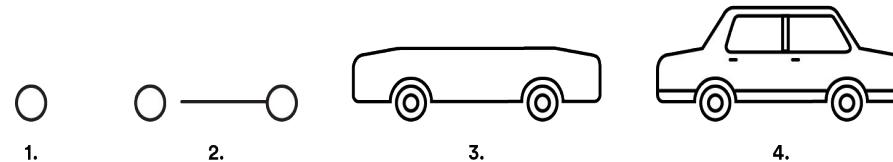
# How We MVP



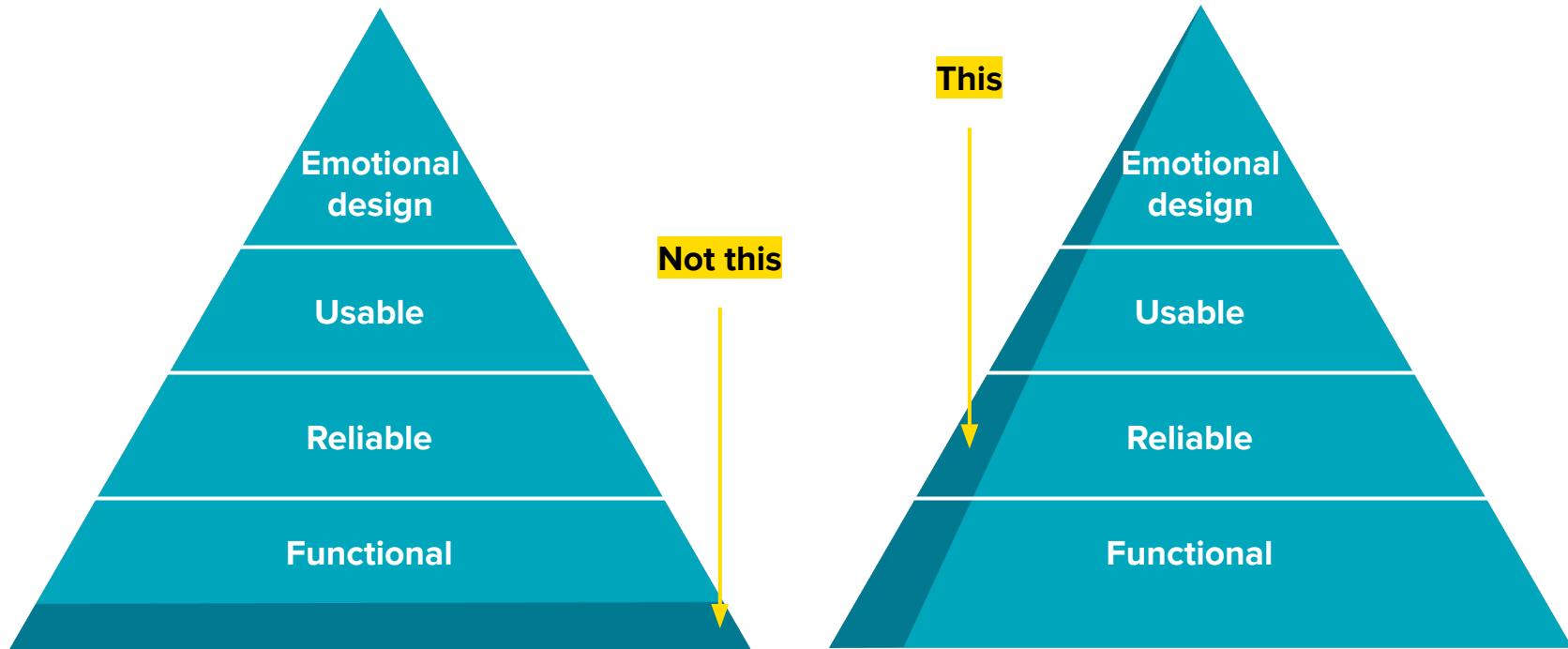
**Like this:**



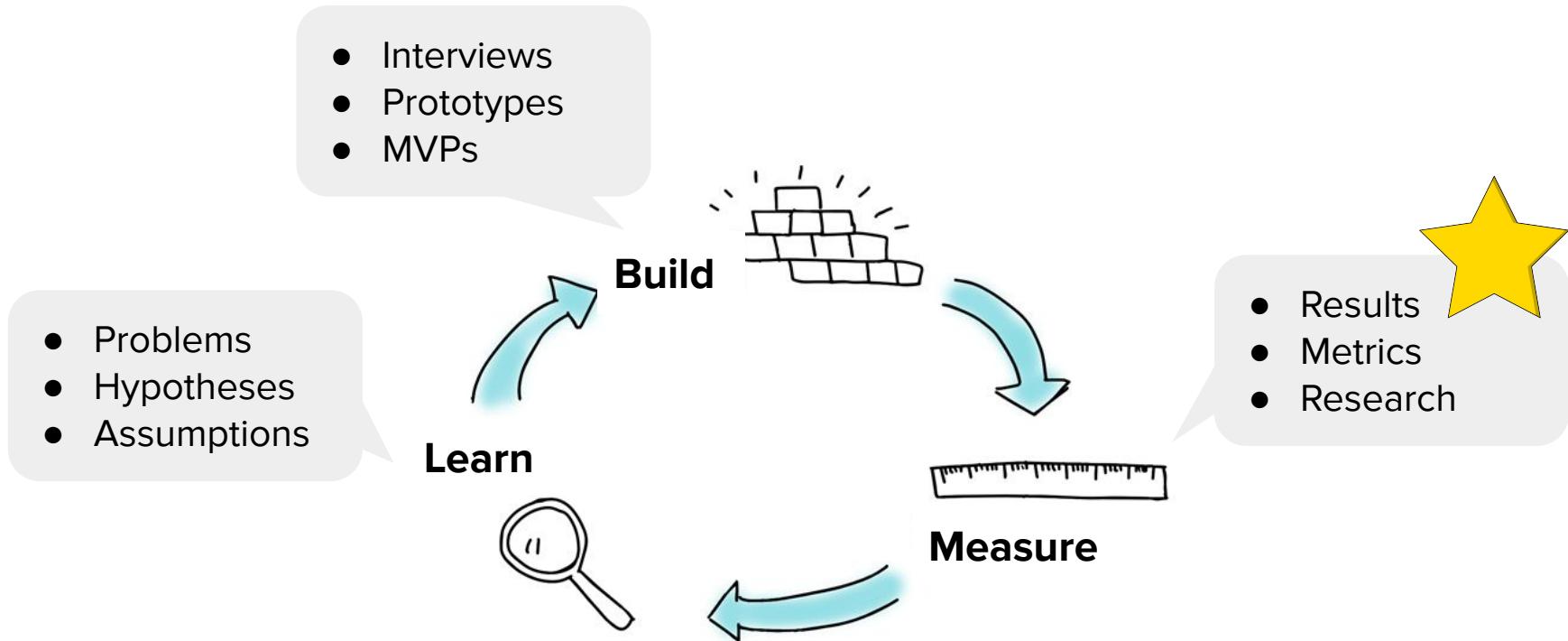
**Not this:**



# A Good MVP Is Still Well Designed



# Measuring and Learning

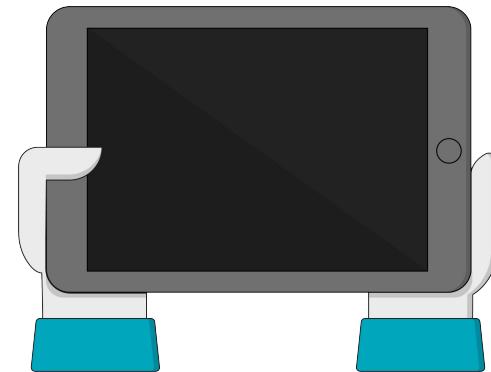


# More Is Not Always Better

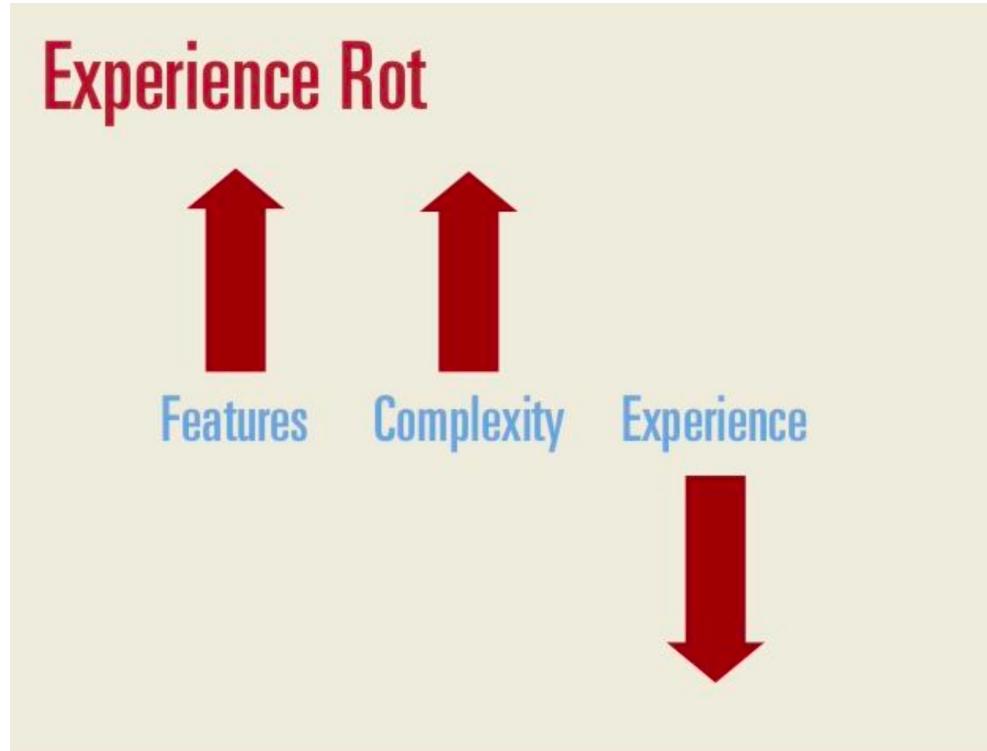
In designing a user interface, one of your key decisions is about **balancing features, functionality, and simplicity**.

The **more features**, the **more complicated** your system:

- Your screens look busier, leading to cognitive load. It will be difficult for users to find what they need.
- You need more extensive documentation to explain your features.
- Excessive feature interactions means more could go wrong. Focus on designing critical features first.

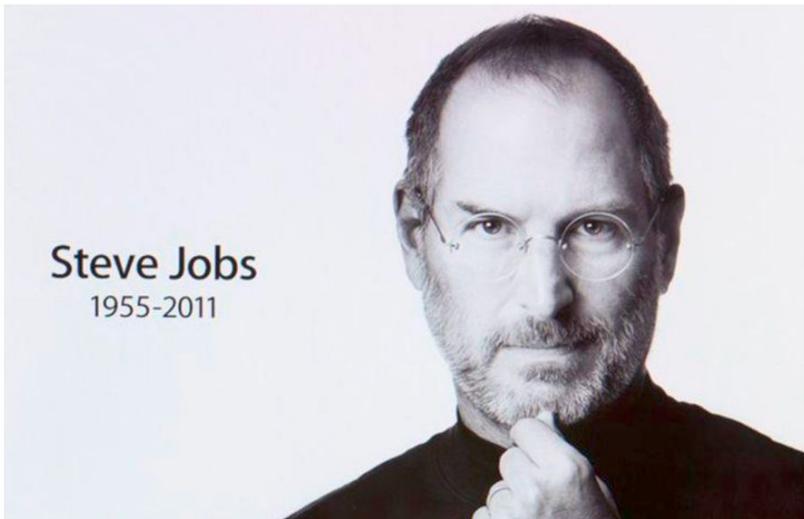


# More Is Not Always Better



# More Is *Not* Always Better

This



Steve Jobs  
1955-2011

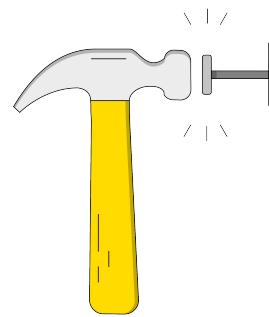
Not This





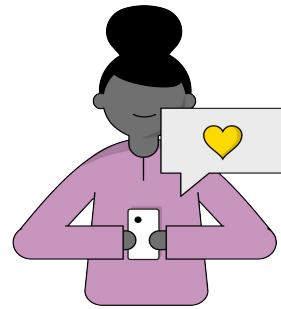
# Features and Prioritization

## Level of Effort



How difficult will it be to build?

## Customer Value



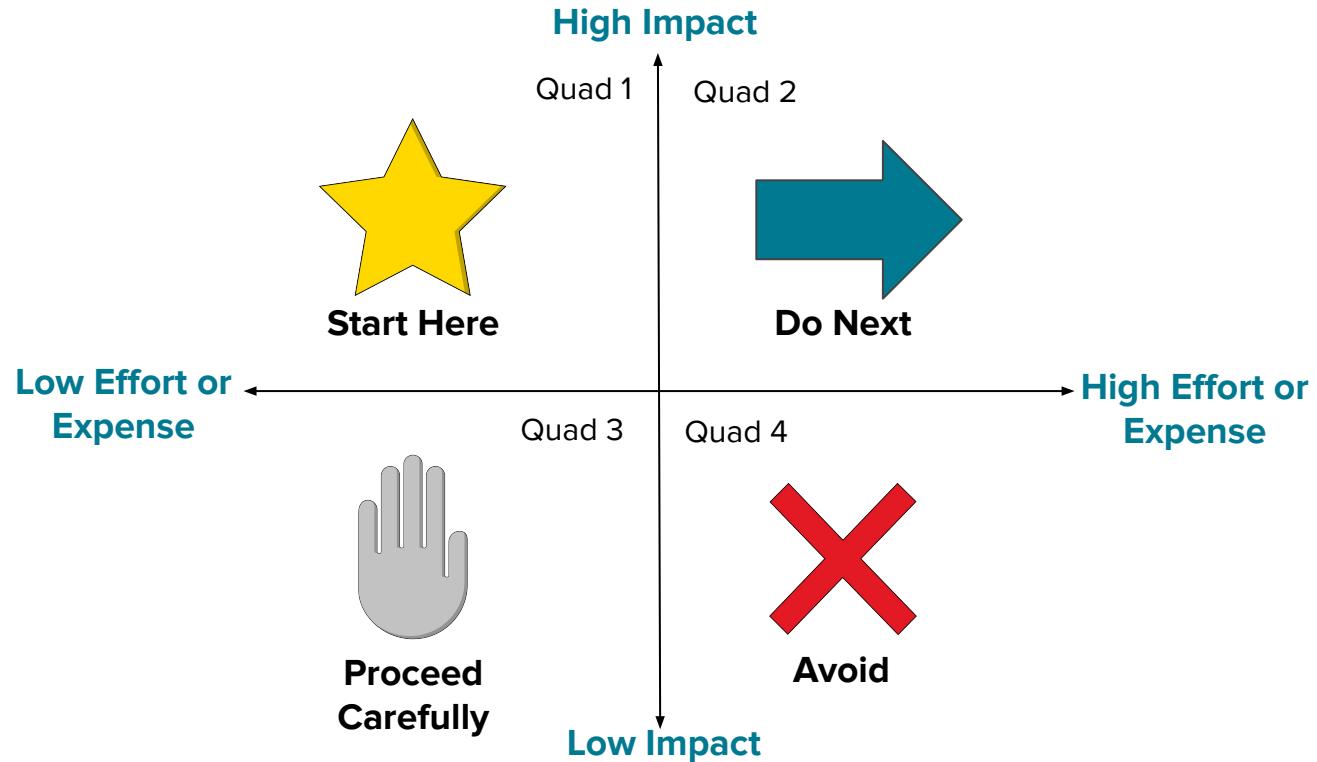
How much more satisfied will our customers be?

## Business Value



How much will it help our business?

# Features and Prioritization



# Determining LoE Is a Team Effort

We don't currently have those components in our style guide.

*Visual designer*

The feature itself is easy to test, but we'll need to do a regression.

*QA engineer*



If we only do this, it's fairly straightforward. Adding these components adds risk to the project.

*Developer*

# Determining LoE Is a Team Effort (Cont.)

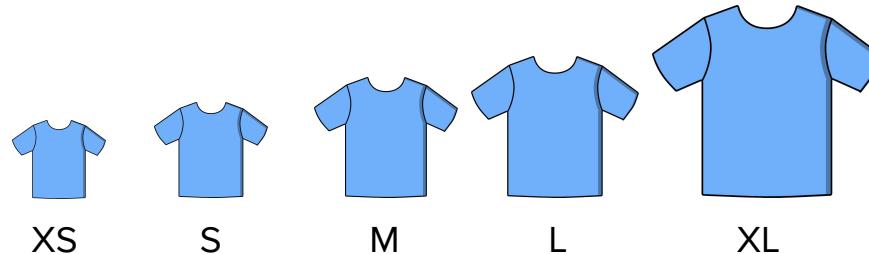
Let's get on the  
same page!

*Product manager*



# Determining LoE

## T-Shirt Sizing



## Story Points

1    2    3    5    8    13



# Determining Customer Value

M	Must
S	Should
C	Could
W	Won't

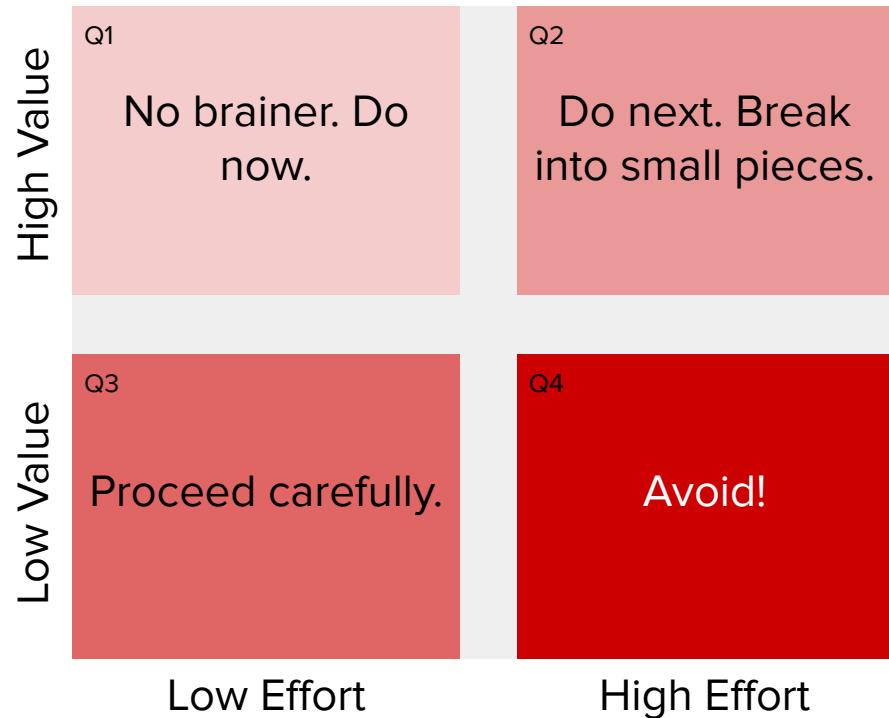
## The MoSCoW Method

Each feature is assigned a priority label based on its relative importance.

# Bringing It All Together

You'll quickly realize that you can't do everything on your list all at once. What now?!

Use a 2x2 matrix to prioritize what you'll build in terms of customer value and level of effort.



# Alternative Prioritisation Methods: Stack Ranking

Another method: **stack ranking**.

Feature	Customer Value (1 = low, 5 = high)	Level of Effort (1 = high, 5 = low)	Score	Rank
Compose Email	5	3	15	1
Create Draft	3	3	9	2



Group Exercise:

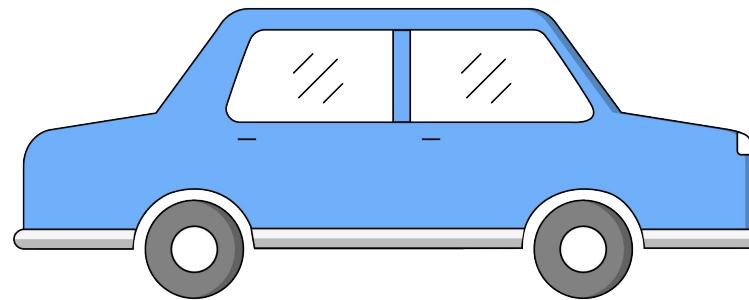
## The GA Motor Company

5 minutes



With your group, you're going to build a new luxury sedan model for next year.

Brainstorm **the features** you'll include for the car.





## Group Exercise: The GA Motor Company

5 minutes



Next, estimate the LoE for each feature.

Use **T-shirt sizing**.





Group Exercise:

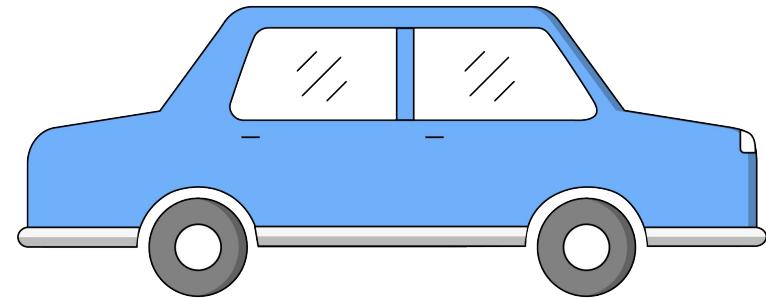
# The GA Motor Company

5 minutes



Rank your features in terms of customer value using the **MoSCoW** method.

M	Must
S	Should
C	Could
W	Won't





## Group Exercise:

# The GA Motor Company

5 minutes



Finally, prioritize features based on level of effort and customer value.

**Stack rank** them in order.

Feature	Customer Value (1 = low, 5 = high)	Level of Effort (1 = high, 5 = low)	Score	Rank
Compose Email	5	3	15	1
Create Draft	3	3	9	2

# Alternative Prioritisation Methods: RICE

Reach: How many customers will this help / reach?

Impact: How positively will this impact the average customer?

Confidence: How confident are we in our estimates (R,I,E)

Effort: How much effort will this take?



Project name	Reach	Impact	Confidence	Effort	RICE score
Project 1	450	3	100%	2	675
Project 2	2,000	1	80%	4	400
Project 3	800	2	50%	1	800

# Alternative Prioritisation Methods: RIFE (Liam's RICE edit)

Reach: How many customers will this help / reach?

Impact: How positively will this impact the average customer each time they use it?

Frequency: How frequently do we estimate our customers will use this?

Effort: How much effort will this take?

$$\frac{\text{Reach} \times \text{Impact} \times \text{Confidence}}{\text{Effort}} = \text{RICE SCORE}$$

*Note: Play with the above equation if you think Reach/Impact/Confidence are not equal weighting, or if dividing by Effort is too much.*

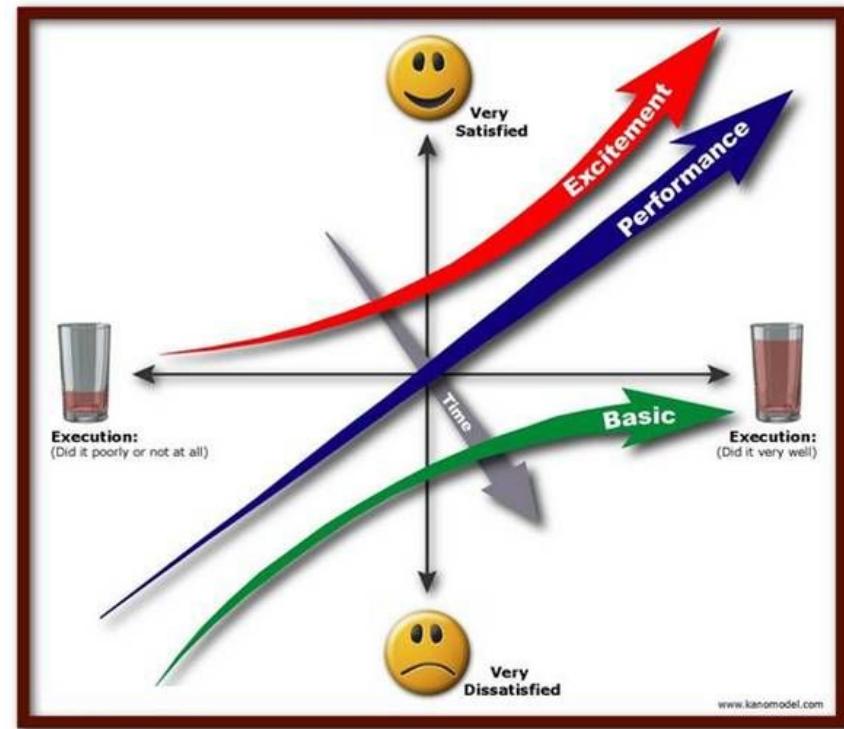
# Alternative Prioritisation Methods: Kano Model

Developed in the 80s by Prof Noriaki Kano during the Japanese Quality Movement which revolutionised manufacturing.

**“Basic”**: If you don’t have this, I will be unsatisfied with your product.

**“Performance”**: Satisfaction improves in a linear relationship.

**“Delighters”**: Features that customers did not suspect, that surprise and delight



Product Management

---

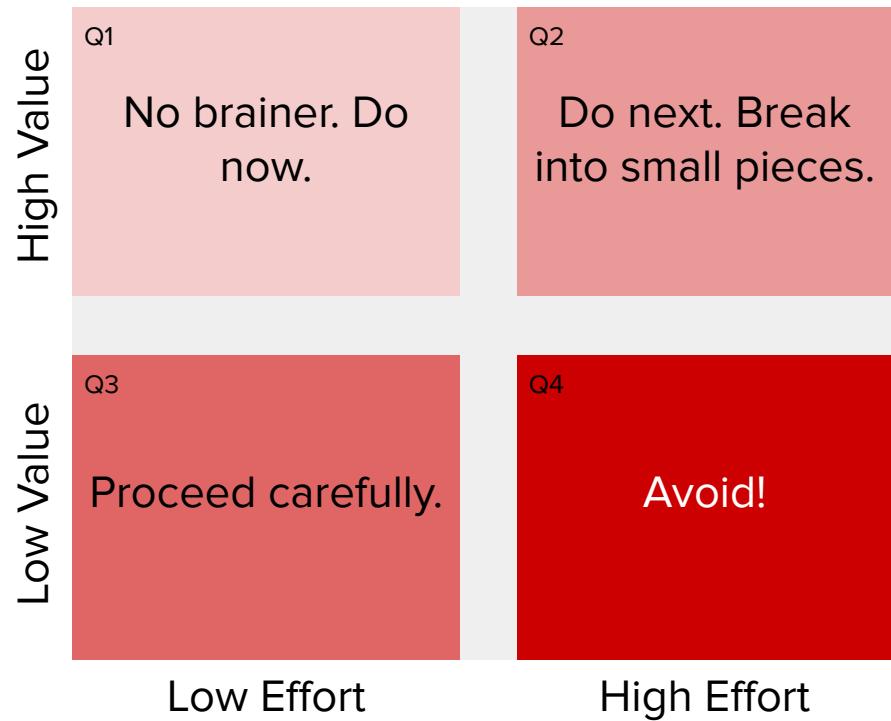
# The Backlog



# The Ones That Didn't Make It

What about all of the features in  
**Quadrants 2, 3, and 4?**

It's not that we don't love them —  
we just can't do them right now.



# The Product Backlog



Makes it **easy to plan** for future work cycles and releases.



A single source of **next steps** for the development team.



Captures **all of the work** you do, not just the big stuff.



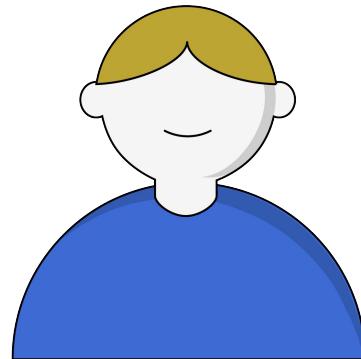
Items are **prioritized** and fully fleshed out.

# What's in the Backlog?

- List of prioritized items to build, including features, issues, and engineering tasks.
- Owned by product owner or product manager.
- Should be “groomed” regularly, keeping the most important items at the top and removing irrelevant items.

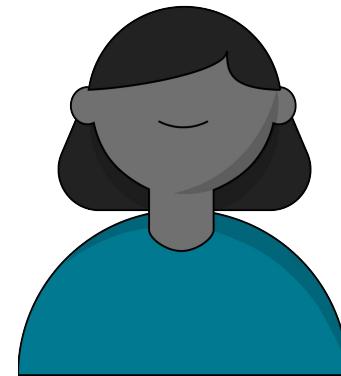
		TIS-30	Create Saturn Summer Siz	2.2	Summer Saturn Sale	2
		TIS-31	Create Teams in Space we	2.2	Summer Saturn Sale	2
		TIS-32	Create Video Assets for Sa	2.2	Summer Saturn Sale	3
		TIS-29	Create Banner Ads to use f	2.2	Summer Saturn Sale	2
		TIS-23	Engage JetShuttle Space	2.1	Space Travel Partners	3
		TIS-24	Sign Contract for SunSpo	3.0	Space Travel Partners	2
		TIS-21	Create Email Campaign for	2.2	Summer Saturn Sale	5
		TIS-35	Draft out analytics plan for	2.1	Summer Saturn Sale	5
		TIS-34	Create Launch Plan for the	2.1	Summer Saturn Sale	8
		TIS-46	Update LocalTransportCont	3.0	Large Team Support	5
		TIS-42	Extend booking experience	2.1	Large Team Support	5
		TIS-43	Extend booking experience	2.1	Large Team Support	13
12 issues				Estimate	55	

# Who Wants to Know?



**Development Team**

Find the most important thing to work on next.



**Stakeholders**

Understand your upcoming work and why you can't take on more.

# The DEEP Backlog Model

Your backlog should be...

- **Detailed**. Detailed enough that any member of the team can work on the items independently.
- **Emergent**. Flexible work that fluctuates based on the needs of the team, business, or customer.
- **Estimated**. If an item comes with a high LoE estimate, it is likely large enough to break apart into more than one item.
- **Prioritized**. If all items in the sprint are completed early, the team should pull in the next item in the backlog to continue work.

# The Product Backlog

TIS board

## Backlog

QUICK FILTERS: Only My Issues Recently Updated

VERSIONS  
EPICS

Backlog 7 issues Create Sprint

6 20 0 ...

▼ TIS Sprint 1 3 issues

Complete first pass  
07/Sep/16 4:49 AM • 21/Sep/16 4:49 AM

...  
TIS-1 Add support for teams larger than 20 people  
TIS-3 Make working with space travel partners easier  
TIS-4 500 Error when requesting reservations

Teams in Space / TIS-5

Requesting available flights takes > 5 seconds

Estimate: 4

Details

Status: TO DO (View workflow)

Priority: Medium

Component/s: None

Labels: None

Affects Version/s: None

Fix Version/s: None

Epic Link: None

People

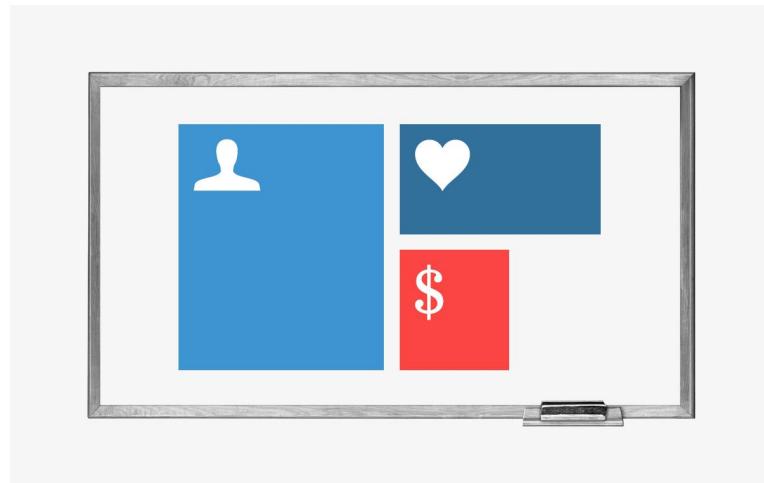
Reporter: Atlassian  
OnDemand  
[Administrator]

Assignee: Will  
Assign to me

This screenshot shows a Jira product backlog board titled 'TIS board'. The main area displays two sections: 'TIS Sprint 1' with 3 issues and 'Backlog' with 7 issues. On the left, there are vertical navigation bars for 'VERSIONS' and 'EPICS'. An epic titled 'Teams in Space / TIS-5' is currently selected. The epic details include an estimate of 4, a status of 'TO DO', and a priority of 'Medium'. The 'Details' section also lists components, labels, and version information as 'None'. The 'People' section shows the reporter as 'Atlassian OnDemand [Administrator]' and an assignee named 'Will', with an option to 'Assign to me'. The backlog issues include tasks like 'Add support for teams larger than 20 people', 'Make working with space travel partners easier', and '500 Error when requesting reservations'.

# Backlog Pro Tips

- Keep everything in one place: features, bugs, requirements, etc.
- Review it on a regular basis to make sure prioritization and requirements are correct.
- Group items into “near term” (more detail) and “long term” (more vague).
- Get rid of items that will never be worked on!



Product Management

---

# Wrapping Up



# Today's Learning Objectives

In this lesson, we:

- ✓ Estimated the level of effort and customer value of a given feature in order to prioritize it.
- ✓ Used a product backlog to manage and track features and tasks.

**What are your takeaways?  
What questions remain?**

# Additional Resources

Practice Again	Digging Deeper
<p><b>Project Scoping</b></p> <ul style="list-style-type: none"><li>• <a href="#"><u>Project Estimation Through T-Shirt Size — Radius-Engineering</u></a></li></ul> <p><b>Strategies for feature prioritization</b></p> <ul style="list-style-type: none"><li>• <a href="#"><u>7 Strategies to Choose the Best Features for Your Product</u></a></li><li>• <a href="#"><u>A Radical, and Simple, Approach to Product Prioritization</u></a></li><li>• <a href="#"><u>How PMs Defeat Feature Creep</u></a></li><li>• <a href="#"><u>Killing Features to Improve Your Product</u></a></li></ul>	<p><b>Jobs To Be Done</b></p> <ul style="list-style-type: none"><li>• <a href="#"><u>What is Jobs to Be Done (JTBD)?</u></a></li><li>• <a href="#"><u>8 Things to Use in “Jobs-To-Be-Done” Framework</u></a></li></ul> <p><b>How to Prioritize Product Features &amp; Improvements</b></p> <ul style="list-style-type: none"><li>• <a href="#"><u>How to Prioritize Product Features and Improvements</u></a></li><li>• <a href="#"><u>Product Prioritization Frameworks</u></a></li><li>• <a href="#"><u>20 Product Prioritization Techniques</u></a></li><li>• <a href="#"><u>Scope Creep: 5 Ways PMs Can Reduce It</u></a></li></ul>



# Don't Forget: Exit Tickets!



