# Building quality WooCommerce extensions

skyver.ge/woosesh18

# Heya!

I'm the Head of Product at SkyVerge.

Our team has built over 70 eCommerce products for **WooCommerce**, **Shopify**, & **Easy Digital Downloads** over the past 5 years.

SkyVerge.com  |  Jilt.com

**0**

# Questions to answer

How will this help me build great eCommerce software?

" How can I build something
customers need?

How do I figure out customer
needs without any customers?

"

How can I determine what's in-scope or out-of-scope for my product?

What mistakes can I learn from or avoid?

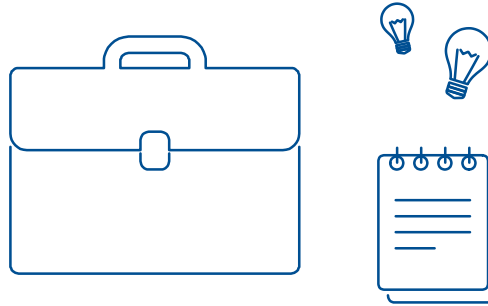" How do I gather actionable customer feedback?

How can I set my product up for long-term success?

# 1

# Your initial build

Validating ideas and creating a v1

# The job for your product

What pain points will this product solve?
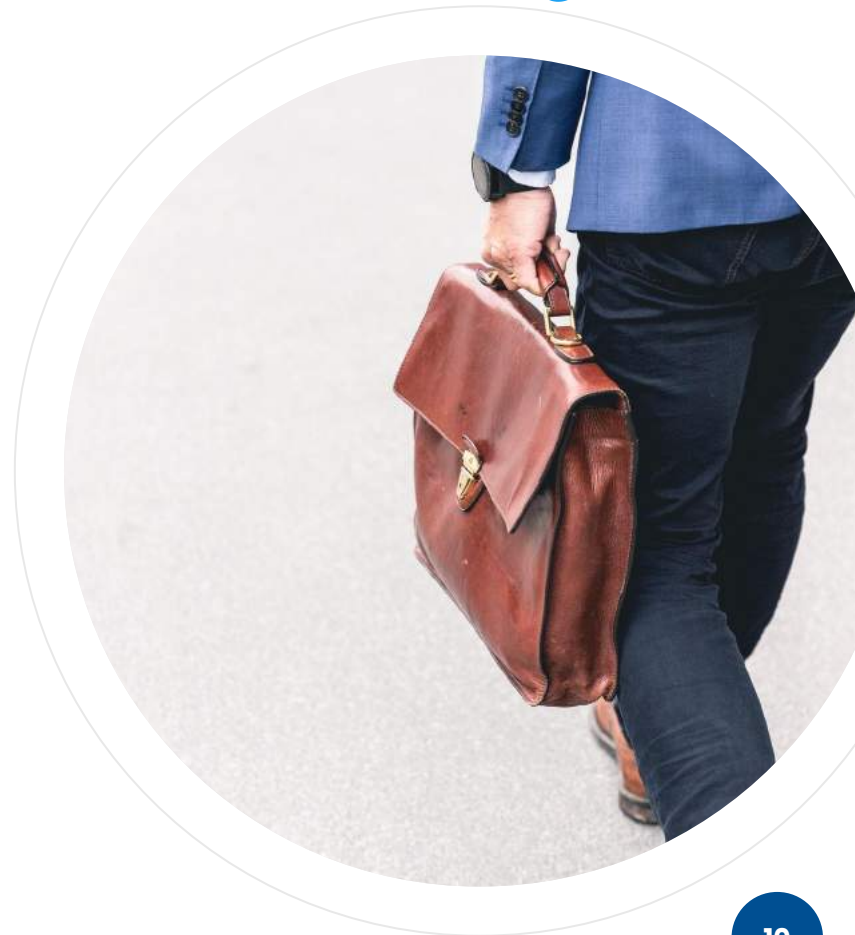
Why did Peter Buy The Snickers?

*Do we need to know about Peter, or what Peter wants from a Snickers?*

# Jobs to be done

Framework for understanding the needs of your customers. Rather than understanding **who** uses it – via *personas* or *user stories* – it focuses on **why** they use it, or what *job* the product was hired to do.
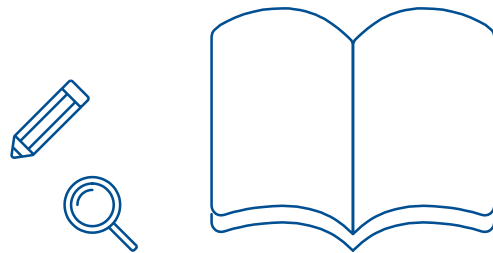
*Designing features using Job Stories* ➥

> " *Products don't match people; they match problems.*

*Des Traynor, co-founder, Intercom*

# Customer research

Understand your audience

# Interview your existing customers

Even if they use an unrelated product, dig deep into existing pain points. See if your product job aligns with current customers.

*A framework for JTBD interviews*

*Creating B2B personas*

# What if I don't have customers?

What if you're starting out?

- Read industry news and guides
- Read the comments!
- Drive traffic to landing pages + ask for surveys / interviews

Getting people to talk to you isn't easy, but it's worth it.
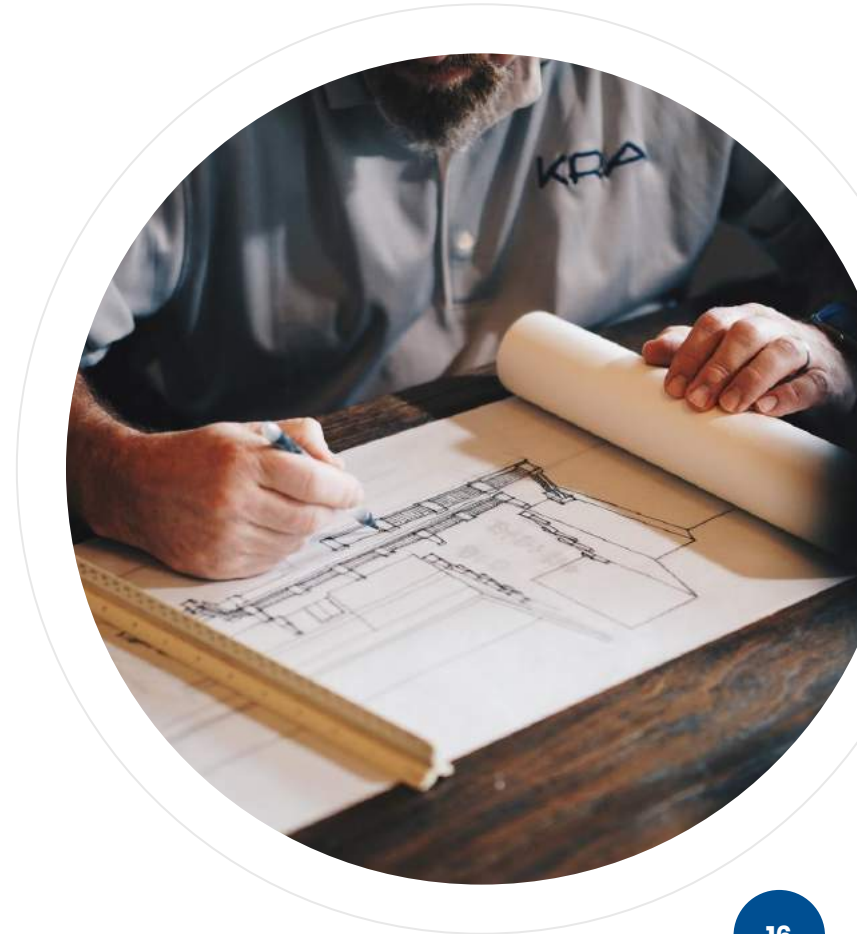
# 2

# To build or not to build

Making v1 product decisions

# Be ruthless about scope

*Feature complete* doesn't mean *feature rich*. You can deliver value to your customers with a project that is smaller in scope than you think.

*Path to a Minimum Lovable Product* ➡

Pick **one essential job** for your product.

This is your *North Star*. Use it to define "feature complete" for v1.

# Why cut scope?

Deliver value to customers as quickly as possible. Like compound interest, this will increase total value from your product over time vs waiting to deliver more value later.
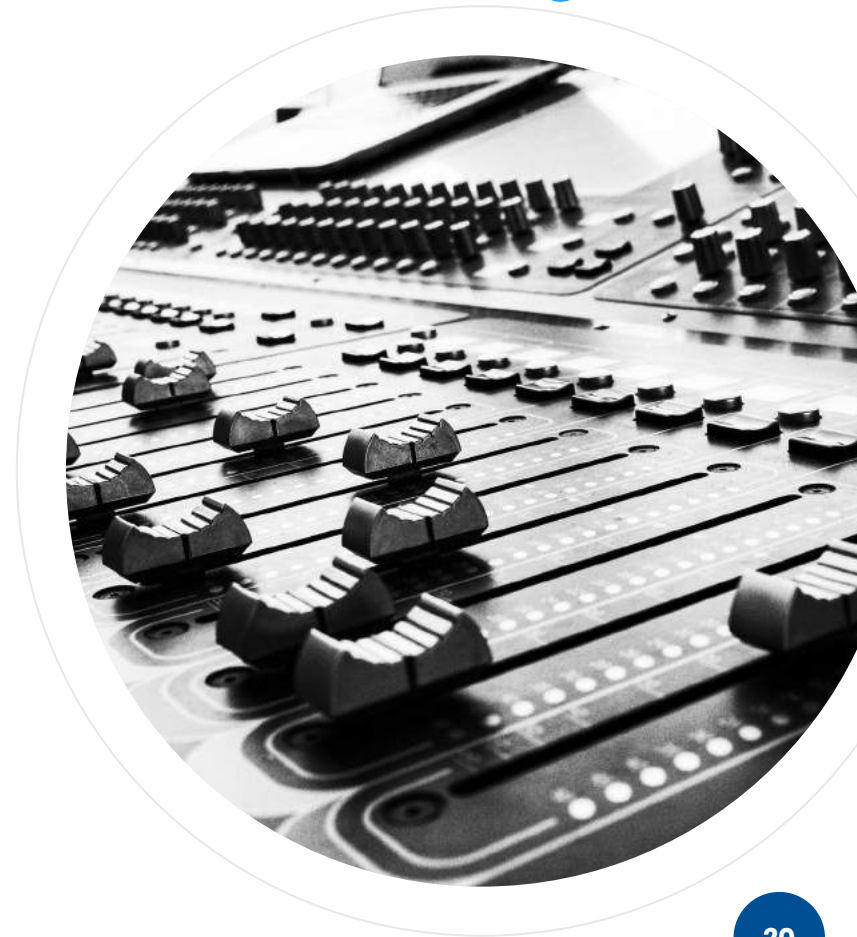
*The Time Value of Shipping* ➔

" *As with the time value of money, the time value of shipping is a simple idea: delivering customer value now is worth more than delivering value later.*

*Brandon Chu, Director of Product, Shopify*

19

# Be opinionated about design

Not simply *visual design*, but your **product design** and user experience. What does "decisions, not options" really mean?
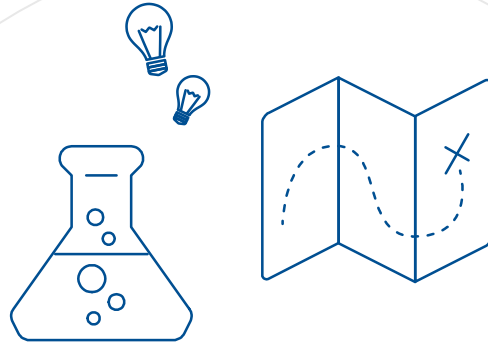
# 3

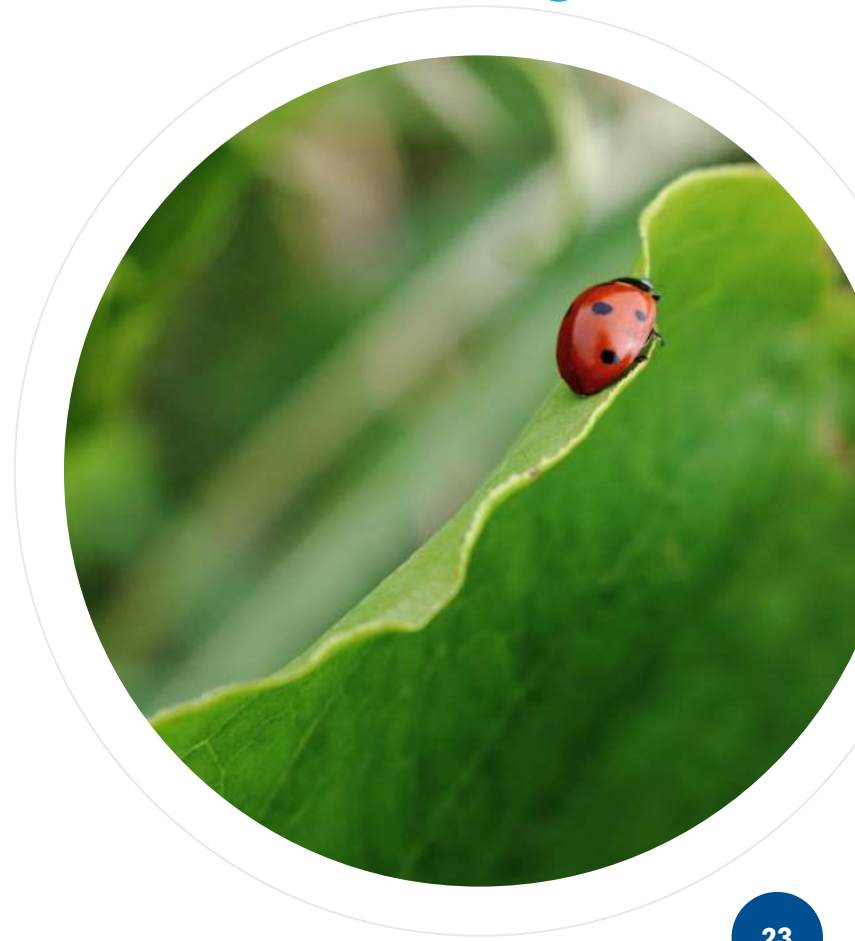# Maintaining products

Optimizing for longevity

# Product updates

Prioritizing fixes and features

# Triage bugs

With a new product, it's easy to fix every bug (and you should spend time here!). Over time, you need to develop a triage framework for what to fix, and what to backlog.

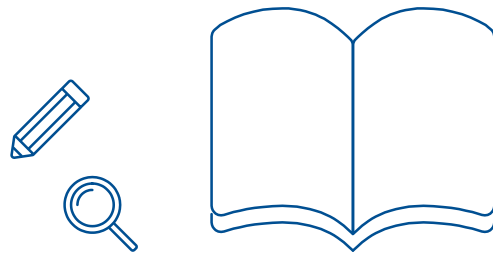*Software has bugs. This is normal.*

# Add features by customer ROI

Develop a formula for customer ROI – have multipliers for features that reduce churn or increase customer value – and assess new projects using this formula.

*Ruthless Prioritization*

# Customer development

Build great feedback loops

# Talk to customers

## Interviews

Survey existing customers and schedule as many interviews as you can. Uncover motivations (jobs) and friction.

*How to Structure Customer Development Interviews*

## Support

Support isn't separate from the software experience, it's part of it. Use support to inform UX improvements.

*Developing a Customer Service Tone Guide*

# Requests vs. needs

What customers *say* they want and what they *need* may be different.

- Ask: "I want to subscribe to a service that gives me updated tax rate CSVs as tax rates change."
- Need: An easy way to incorporate changes to tax rates in a store.
- Best solution: Tax service integration

27

# A cautionary tale

Listen to your **happy** customers as much as unhappy customers or sales leads.

People who already pay you can help you see existing value, and ensure you don't lose it.
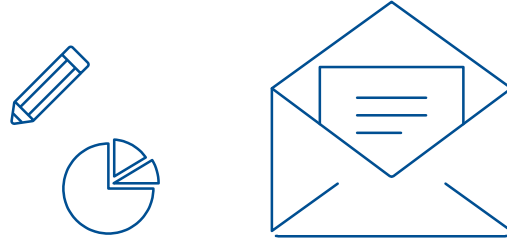
# Tracking feedback

Connecting customers to product

# Log everything

Track customer feedback, even if you're the only person doing support or building your product.
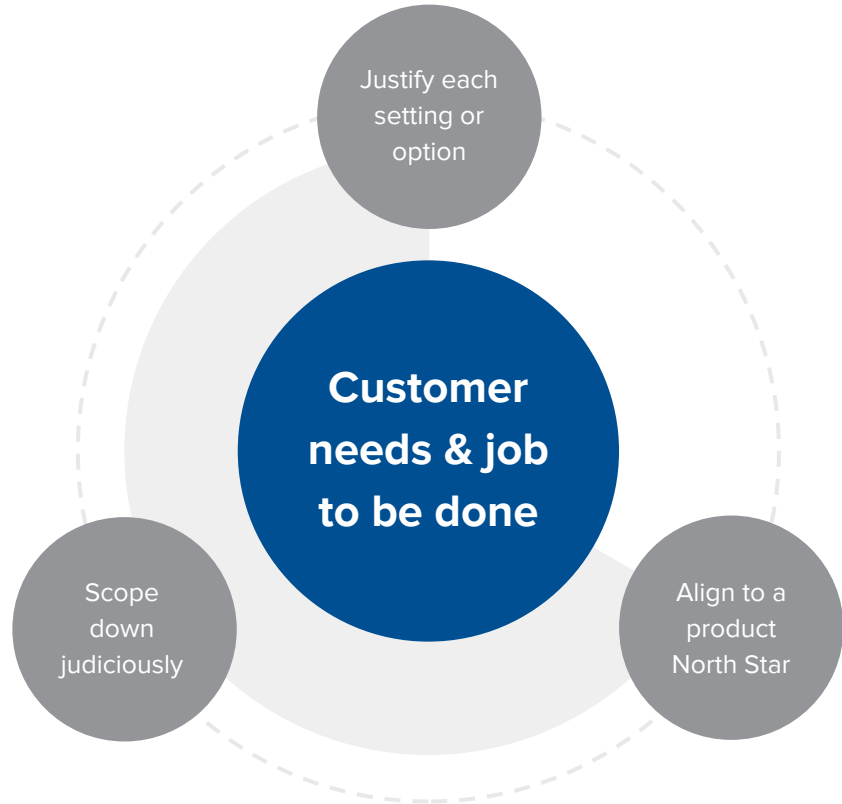
*Trello* → *Clubhouse* →

# Takeaways

# Build opinionated products

Justify each setting or option

Customer needs & job to be done

Scope down judiciously

Align to a product North Star

# Deliver value

**1** **Ship sooner to compound value**
The Time Value of Shipping

**2** **Deliver *feature* completeness**
Build a minimum lovable product before being feature-rich.

**3** **Understand needs vs requests**
Solve deep problems to be forward-thinking.

# Thanks.

## Let's dig into questions!

SkyVerge.com | Jilt.com

skyver.ge/woosesh18

# Credits

Thanks to all the people who made and released these awesome resources for free:

- ○ Presentation template by SlidesCarnival
- ○ Photographs by Unsplash