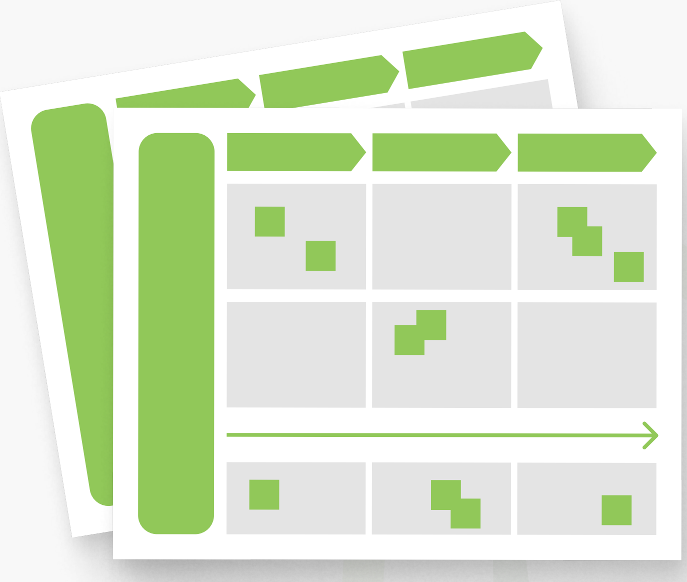
DevRel.Agency

Developer Journey

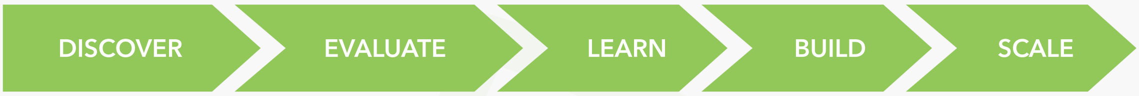


Like a customer journey, The Developer Journey Map is a visualization that identifies the path a developer follows and experiences.

Your goal is to progress your developer through their journey from discovery to scale quickly as possible to increase product adoption and revenue potential.

Developer Journey

Stages



DISCOVER EVALUATE LEARN BUILD SCALE

There are five stages on the Developer Journey which indicate significant changes in the developer's intent and actions. They do not imply elapsed time:

1. Discover
2. Evaluate
3. Learn
4. Build
5. Scale

For example, a developer may well complete the first three or four stages on the same day if they invest the time, and you have provided a friction-free experience.

Conversely, the adoption cycle to become a paying user could take a year if you piqued a developer's interest, but they haven't yet found the right use case or project to build with your product. Also, the type of developer product matters. For example, an API typically has a faster journey time than an IOT hardware board.

Note that each stage has a stated goal for the developer.

If the goal is achieved, the developer will move on to the next stage of adoption. Each goal has key questions that you need to answer for the developer to achieve the overall goal of the stage in question. The goals and associated questions may be different for your program. This is fine. What is critical is you have identified them, and can provide the answers necessary for the developer to progress.

Developer Journey

Touchpoints

As a Developer Relations leader, it's your job to develop and optimize each of the "touchpoints," ensuring you answer all of their questions and meet the goals for each stage of the journey.

Owned and Earned

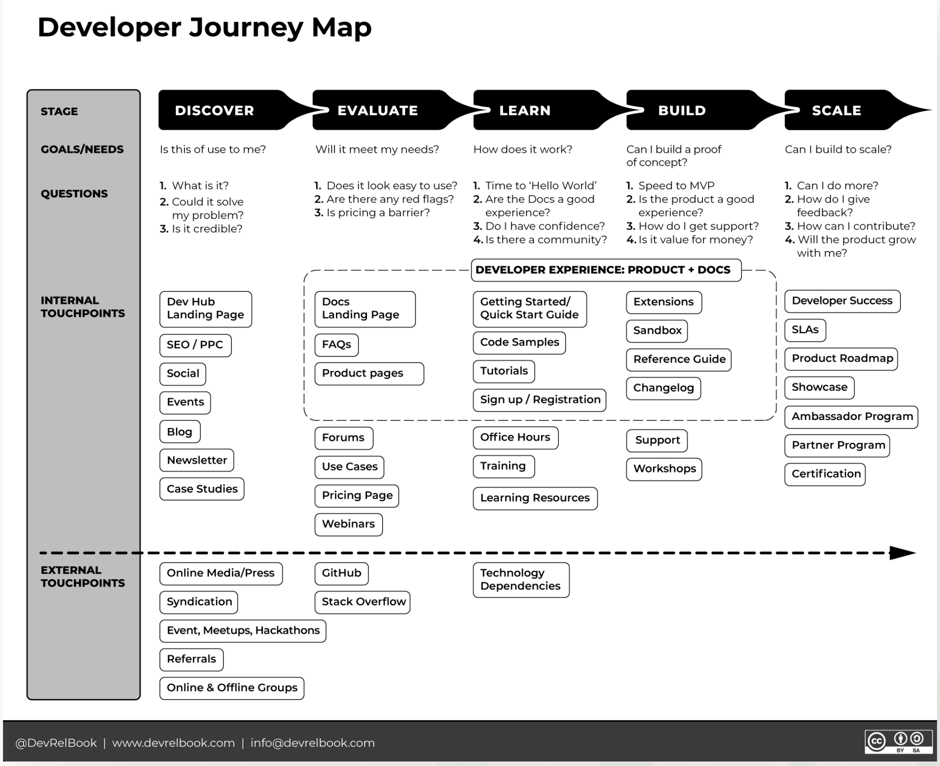
Owned touchpoints are the properties and content that your company owns and controls. You have complete control over what a touchpoint is, what it does, and how it contributes. Examples include your website, developer hub, documentation, your messaging on social media, your advertising, pricing information, code samples, support, etc.

External touchpoints are the opposite. You do not have direct control over them, however, they are key resources and channels to reach both your existing users and your potential future customers. Ensuring your brand and tools are visible, discussed, and supported on these properties is absolutely vital to ensure awareness grows, reputation is positive and adoption rises. Examples include developer focused media, industry analysts, 3rd party communities, external forums, etc.

Those interactions, which we call developer touchpoints, are used to map a developer's experience along the way - how they engage, what and whom they engage with, how it makes them feel, and how they react. Organizing these touchpoints into a map helps you identify shortcomings or friction, which you can then optimize to create a better overall experience.

Developer Journey

Example Developer Journey Map



Developer Journey Map

STAGE	DISCOVER	EVALUATE	LEARN	BUILD	SCALE
GOALS/NEEDS	Is this of use to me?	Will it meet my needs?	How does it work?	Can I build a proof of concept?	Can I build to scale?
QUESTIONS	1. What is it? 2. Could it solve my problem? 3. Is it credible?	1. Does it look easy to use? 2. Are there any red flags? 3. Is pricing a barrier?	1. Time to 'Hello World' 2. Are the Docs a good experience? 3. Do I have confidence? 4. Is there a community?	1. Speed to MVP 2. Is the product a good experience? 3. How do I get support? 4. Is it value for money?	1. Can I do more? 2. How do I give feedback? 3. How can I contribute? 4. Will the product grow with me?
INTERNAL TOUCHPOINTS (DEVELOPER EXPERIENCE: PRODUCT + DOCS)		Docs Landing Page FAQs Product pages	Getting Started/ Quick Start Guide Code Samples Tutorials Sign up / Registration	Extensions Sandbox Reference Guide Changelog	
INTERNAL TOUCHPOINTS	Dev Hub Landing Page SEO / PPC Social Events Blog Newsletter Case Studies	Forums Use Cases Pricing Page Webinars	Office Hours Training Learning Resources	Support Workshops	Developer Success SLAs Product Roadmap Showcase Ambassador Program Partner Program Certification
EXTERNAL TOUCHPOINTS	Online Media/Press Syndication Event, Meetups, Hackathons Referrals Online & Offline Groups	GitHub Stack Overflow	Technology Dependencies		

@DevRelBook | www.devrelbook.com | info@devrelbook.com

Here we present a sample Developer Journey Map which includes the goal of the Developer at each stage, the key questions for each stage, and forty-two example touchpoints for you to consider.

We have placed them in the spot where they are typically first encountered. Of course, touchpoints can be relevant in more than one stage of the journey. Don't feel constrained by what these are or where they are situated in our example - do what is right for you. We encourage you to iterate on our example.

Developer Journey

How To Use The Developer Journey Map

We strongly recommend you regularly audit your Developer Experience (DX) to identify and remove any friction that may stop or slow down the adoption of your products & services.

Getting your Developer Experience (DX) right is the most important factor of converting your engagement with developers into adoption of your product, a process often called onboarding.

Create your own Map

We recommend recreating the Developer Journey Map template on a whiteboard or on A3 paper. Vertically add headings for Stage, Goal/Needs, Questions, Internal and External Touchpoints. Horizontally add the five stage headings.

Write in the Goal/Needs and Questions - either use our examples or adapt them to fit your own context.

Now put yourself in the shoes of your Developer.

Plan as many scenarios as necessary, which take into account:

- The Developer's Persona
- Other relevant factors, like their familiarity level with your company and your product
- The end goal they are trying to achieve in this scenario e.g. complete an API call, build a Hello World style application, encode a video, or move some cryptocurrency between two accounts.

Developer Journey

Using the parameters of the scenario, walk through the Developer Experience of discovering, evaluating, using, and building with your technologies, from the Developers perspective.

Use post it notes to add touch points as they present themselves, and then objectly rate the experience with that touchpoint. A RAG (Red/Amber/Green) colour system is useful here as it will enable you and your team to quickly scan your end to end journey for problems.

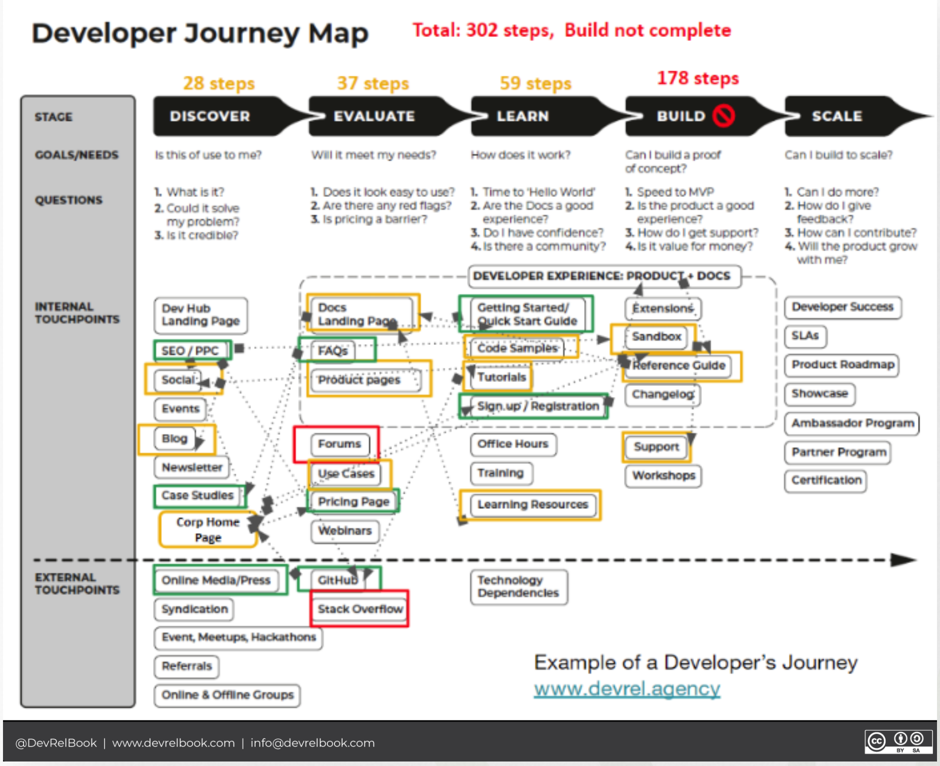
- Red indicates friction which stops or significantly hinders the Developer progressing on their journey
- Amber indicates friction that slows progression which may test Developers patients and undermine their confidence in your product, but doesn't actually lead to the Developer walking away
- Green indicates zero friction and a smooth interaction with that touchpoint

Draw directional arrows to indicate the path a Developer travelled along to help identify the relationships between your touchpoints, so you can ensure smooth entry and exits between them, consistent UI design, branding, etc.

Add notes to your touchpoints detailing what your audit uncovered for future reference. Remember the final results will likely be reviewed by people that may not have participated in the audit, so don't assume your Developer Journey Map conclusions and observations will be obvious to them.

Developer Journey

Your completed map may end up looking something like this:



Developer Journey Map

Total: 302 steps, Build not complete

	28 steps	37 steps	59 steps	178 steps	
STAGE	DISCOVER	EVALUATE	LEARN	BUILD 🚫	SCALE
GOALS/NEEDS	Is this of use to me?	Will it meet my needs?	How does it work?	Can I build a proof of concept?	Can I build to scale?
QUESTIONS	1. What is it? 2. Could it solve my problem? 3. Is it credible?	1. Does it look easy to use? 2. Are there any red flags? 3. Is pricing a barrier?	1. Time to 'Hello World' 2. Are the Docs a good experience? 3. Do I have confidence? 4. Is there a community?	1. Speed to MVP 2. Is the product a good experience? 3. How do I get support? 4. Is it value for money?	1. Can I do more? 2. How do I give feedback? 3. How can I contribute? 4. Will the product grow with me?
INTERNAL TOUCHPOINTS	Dev Hub Landing Page SEO / PPC Social Events Blog Newsletter Case Studies Corp Home Page	Docs Landing Page FAQs Product pages Forums Use Cases Pricing Page Webinars	Getting Started/ Quick Start Guide Code Samples Tutorials Sign up / Registration Office Hours Training Learning Resources	Extensions Sandbox Reference Guide Changelog Support Workshops	Developer Success SLAs Product Roadmap Showcase Ambassador Program Partner Program Certification
EXTERNAL TOUCHPOINTS	Online Media/Press Syndication Event, Meetups, Hackathons Referrals Online & Offline Groups	GitHub Stack Overflow	Technology Dependencies		

DEVELOPER EXPERIENCE: PRODUCT + DOCS

Example of a Developer's Journey
www.devrel.agency

@DevRelBook | www.devrelbook.com | info@devrelbook.com

Developer Journey

Act on your findings

Using the Developer Journey Map to audit your Developer Experience is an incredibly powerful and actionable exercise. It will:

- Indicate how different Developer Persona's interact with your offering
- Provide you with an 'at a glance' visual representation of where you need to focus your effort to improve adoption of your product.
- Illustrate the relationship between touchpoints
- Provide you with a list of actionable improvements required to change red and amber touchpoints to green

To maximize the impact and ensure cross functional buy in to the results, we strongly recommend a cross functional team collaborates on the journey mapping exercise.

Developer Journey

Support & Resources

- The most comprehensive resource covering The Developer Journey is Developer Relations - How To Build And Grow A Successful Developer Program available from Apress and all good bookstores. It contains six chapters totaling over 15,000 words dedicated to exploring The Developer Journey Map.
- We have a number of blog posts on our website exploring The Developer Journey and Developer Experience. Visit devrel.agency/blog to read more.

Even with the best will in the world, it can be difficult for your team to be completely free of bias when they are involved in reviewing something they are so close to and invested in.

People have a tendency to seek out information that supports something they already believe (confirmation bias), or believe fault lies in things outside of their control (self-serving bias), or attribute behavior to unfounded stereotypes (fundamental attribution error). Also, when something is seen many times, the steps along the way are often forgotten (generalization).

Getting a fresh perspective from experts that are detached from the product is imperative. DevRel.Agency specializes in Developer Experience Audits. Our external expertise is impartial and brings an “industry perspective” with a view on best practice and how you benchmark against others.

Learn more about our Developer Experience Audit service.

Developer Journey Map

Designed for:

Designed by:

Date:

Version:

STAGE	DISCOVER	EVALUATE	LEARN	BUILD	SCALE
GOALS/NEEDS					
QUESTIONS					
INTERNAL TOUCHPOINTS					
EXTERNAL TOUCHPOINTS					