This Starter Guide is continually updated. Download the latest version at: https://developer.ibm.com/code/community
Do you code with IBM?

Throughout its history, IBM has been a steadfast advocate for its clients. Now we are reinventing ourselves to ensure that we continue to meet our clients’ needs and help them solve today’s complex technology challenges.

At the center of that reinvention is the recognition that the developer is the single most important role in IT today. That’s why IBM is putting the developer at the heart of everything we do.

Reach and engage developers

The way we share information has changed; we now communicate and share in a fast-flowing stream of data using cloud technologies. Developers are no longer simply technology implementers, they’re thinkers, planners, architects and builders, providing real-world experience, submitting recommendations, and making decisions.

IBM is redefining our relationship with developers. We’re reaching out to the global development community, engaging, connecting, sharing, and listening, learning how to make development easier.

What’s Inside?

This field guide provides is divided into four main sections:

- Developer Journeys
- Deliver a journey
- Become an IBM Coder
Developer Journeys

Developer Journeys are at the core of IBM’s commitment to developers. They create a connection, both technical and emotional, by engaging the developer through shared experience and desire to innovate and improve.

Developer Journeys are 360-degree views of code. They provide access to code, as well as supporting content such as architecture diagrams, videos, live demonstrations, blog posts, and tutorials.

A Developer Journey engages developers and creates an attachment and “pull.” Developers who have followed a journey will want to use the technology they’ve seen demonstrated. A Developer Journey is much more than the sum of its parts – it provides context, substance, and a vision for technologies.

There are two types of Developer Journeys: **Demonstration** and **On Ramp**.

- **On Ramp Journeys**: These journeys are “art of the possible” examples. They show new features and functions, and introduce new ways to think of existing solutions. They’re targeted at developers who like being the first to try new technologies and approaches, the ones who love to “play with new toys.”
- **Solution Journeys**: These journeys are “run in production” examples that describe the real-world implications of deploying code publicly. Developers already understand the fundamental differences between production code and demos or proofs of concepts; these journeys give them need-to-know information and best practices to run their code in production.
Tell a story

People love stories. Through stories, we share experiences, connect to others, and come to understand the world. Stories help us build emotional bonds and form attachments.

Like any other quality writing, technical writing should tell a story. We need to guide our readers from the beginning (the concept), through a set of challenges (the tasks, or steps), to the desired outcome (accomplishing the goal).

When you create a Developer Journey, you’re telling a story. You’re going to include characters, a plot, challenges to overcome, and a resolution. You’ll also include a whole lot of technical information along the way. The seamless blending of that technical information with the other elements of the story is what makes a Developer Journey successful.

A good Developer Journey must connect emotionally with readers. You as an advocate need to gain a developer’s confidence and reassure them that they are on track to reaching their goal. Developers need to recognize themselves in the journeys that you create.

TAKE-BACK TIPS

- **Develop your characters.** Characters are at the heart of every story, and every Developer Journey.
- **Know your main character.** The person reading your Developer Journey is the main character of the story.
- **Know your audience.** Who are they? What motivates them? What are their main challenges? Remember, the audience should recognize themselves in your characters.
- **Pay attention to the plot.** Plot comprises the main events of your story. It’s the things that happen as the developer overcomes challenges and works toward a goal.
• **The code or demo is the plot.** The code and demo show the audience the challenges, the solutions, and ultimately, the achieving of goals. When you show these elements, you show your audience how they can be successful.

• **Details give the story color.** There’s more to developing than a concept, a challenge, and a release. Highlight interesting code features, offer tips, show off helpful little time-savers. The details make the Developer Journey realistic, useful, and fun.
Code as the foundation

A Developer Journey tells a story and provides context, describing, clarifying, and amplifying code. But the foundation of your Developer Journey is code, and code the central focus of your role as an IBM Developer Advocate. Without quality code, developers will not engage in a journey.

An advocate must be authentic. Being authentic means sharing the code and sharing the configurations. All your code development for journeys -- from commits, to issues, to automated test cases -- will happen in the open, and all code will be open sourced under the Apache License, Version 2.0. Advocates need to be familiar with and understand the IBM internal open source contribution guidelines.

TAKE-BACK TIPS

- All code will be hosted on GitHub under https://github.com/ibm.
- All code will be open sourced under the Apache License, Version 2.0.
- All code will contain a README.md file that describes what the code does, how to install it, and how to run it.
- The code will include automated testing.
- Wherever applicable, you will include a “Run in Bluemix” link.
- Think “technology first,” not “product first.” Developers care about technology solutions. If they think you’re selling a product, they’re less likely to engage.
Deliver the journey

Developer Journeys are not just static websites with a few pages and links. They are living and evolving content sets -- as underlying components and technologies change, the journey grows and adapts.

When you identify and solve issues, you will update your Developer Journey to reflect the new solution. As you create new content, your journey is enhanced.

When you launch a journey, it will contain at minimum the source code, an overview, an architecture map, a list of related technologies, and a blog post.

Then, as you demonstrate the journeys to developers and receive feedback, you’ll expand and add components: videos from conferences, presentations from meetups, new blog posts, practitioner tutorials – add to the content set anything that contributes to a developer’s understanding.
Developer Journeys

Developer Advocates

Developers
IBMCode

The IBMCode website (developer.ibm.com/code) provides a public home and repository for Developer Journeys. The site is easy to navigate and search to find relevant content.

When developers visit the site they will find all current journeys, plus a calendar of advocate events, blogs related to journeys and the advocates, and content related to various technical topics. The site will also provide contact information for advocates.

TAKE-BACK TIPS

- **Provide links.** Include a link to developer.ibm.com/code in all external communications, including presentations, blog posts, articles, videos, and other content.
- **Update your journey.** You must keep your journey current! Submit any new content related to journeys to the site.
- **Events matter.** Take every opportunity to meet and talk with developers, and make sure you let people know by updating the site with your upcoming events.
- **Write.** People want to know about updates. Write new blog posts as changes occur to with the code or journey to let your audience know about changes, challenges, and success.

---

**Accelerate your development – smarter, faster, together**

---

developer.ibm.com/code
IBM Code
Solve specific problems quickly

On IBM Code, everything you need to solve real problems is compiled into a Journey: architecture diagrams, one-click deployment cheat sheets, and pointers to essential docs.

Don’t see the problem you’re trying to solve?

Engage with us

Featured Journeys

Test drive Hyperledger Fabric Transactions
Take a Fabric-powered test drive for free.
- Read more
- Get the code

Run GitLab on Kubernetes
Deploy a distributed GitLab leveraging Kubernetes and Docker.
- Read more
- Get the code

Create a "cognitive" retail chatbot
Build a fully-functional, retail-ready chatbot.
- Read more
- Get the code

See all journeys

Featured Technologies and Topics

Cognitive
Gain business advantage by unlocking new insights from vast quantities of structured and unstructured data and by developing deep predictive models.
- Read more

Chatbots
Chatbots are conversational robots that interact with users through text and voice on the web or mobile.
- Read more

Serverless
Serverless is a cloud-first development environment that allows you to execute code in response to an event.
- Read more

See all technologies and topics

Featured Open Source Projects

Qiskit API
Qiskit is a Python library around the Quantum Experience that enables you to connect and execute IBM Q Api.
- Read more
- Get the code

TJBot programmable robot
TJBot is a do-it-yourself kit that enables you to build a programmable robotic controller powered by IBM Watson. It’s a part of IBM Watson Maker Kits, a collection of DIY open-source templates to connect to Watson services in a fun way.
- Read more
- Get the code

Application Metrics for Swift
Collect and visualize resource and performance monitoring data for Swift-based applications.
- Read more
- Get the code

See top Open Source projects

Recent blog posts

Analysis: streaming Message Hub records with OpenWire
To write - October 16, 2017
In many of today’s cloud-native applications, data is generated at huge volumes. In other cases, it is analyzed and stored. For this reason, several tools have been developed to manage the environment.

Continue reading

Analysis: Cloudant database changes with OpenWire
To write - October 16, 2017
In many of today’s cloud-native applications, data is generated at huge volumes. In other cases, it is analyzed and stored. For this reason, several tools have been developed to manage the environment.

Continue reading

Upcoming events

Apr 11
DockerCon Europe 2017
April 12 - April 13, 2017, Austin Convention Center

May 9
OSSUK 2017
May 9 - May 11, 2017, Austin Convention Center
OSSUK is the open source community gathering to celebrate achievements, share new ideas, and map the future of open computing through collaboration, education, and connecting people with technology.
Presentations

You’ve talked to developers, you’ve gathered feedback, you’ve built your Developer Journey. Now you need to spread the word.

A presentation to a group of developers is a simple, effective way to share the journey, build your network, and communicate your message. Whether you’re talking to a small meetup or presenting at a large technical conference, you control the message and wield influence. You need to take every opportunity to share IBM Developer Journeys and the tools and processes that will benefit developers.

How do you put together an effective presentation? There are all kinds of instructions on what makes a good talk and how to improve your speaking skills, but to engage with developers you have to first and foremost be authentic. If your audience senses that you don’t believe what you’re saying, that you don’t know your content, or that you’re trying to make a sale, they’ll disengage.

TAKE-BACK TIPS

- **Be useful.** Developers don’t want sales pitches. They want to know how to make their jobs easier. Show them.
- **Don’t just read your slides.** Everyone in your audience can read. Don’t simply put a few bullet points on a slide and then repeat them back. Use images, code snippets, and high-level bullet points to introduce your themes and then expand on them in your own words.
- **Be yourself.** We’re all different. Some of us are naturally outgoing while others are more reserved. Don’t try to be something you’re not. Let your own personality come through.
- **Point the way.** Make sure you give your audience a clear direction. Everyone listening to you should know who to contact, which sites to visit, and next steps to take to implement IBM tools and processes.
Blogs

The developer community around a particular technology or open source project relies on people who are willing to publish the lessons that they’ve learned and who create easy-to-discover content for their peers. A great way to share your experiences is by blogging.

Blogs fill the gap between formal technical articles and short responses on sites like Stack Overflow. A good post requires a descriptive title, provides context, and gets to the point quickly. It can also include relevant code snippets and compelling images such as technical illustrations, architecture diagrams, screenshots, or photos.

By blogging regularly, you build your reputation in the developer community as someone who is willing to help, and you establish yourself as a thought leader on a particular topic. You sharpen your written communication skills and can build up a portfolio that often leads to more formal publication opportunities (such as journals or books).

As a developer advocate, you’ll have access to IBMCode’s WordPress blogging platform, making it easy to get started. You can also publish on your personal blog and syndicate through aggregators to gain wider distribution and reach larger audiences.

TAKE-BACK TIPS

- **Be yourself.** Write with your own voice. Blogging as a developer advocate is less formal than writing for a technical journal or one of IBM’s official product sites.
- **Use the style guide.** To ensure your writing has a consistently high quality level, use the developer advocate style guide and submit your posts for review by IBM content editors.
- **Blog often.** Writing on a consistent basis, even if it’s only a paragraph per week, will help you communicate effectively and build an audience. Engage your readers by telling them about challenges, collaboration, recent events -- anything that connects them to your Developer Journey.
Tutorials

A tutorial is a step-by-step guide to complete a task. For developers, tutorials are an essential learning tool, providing hands-on technical instruction and experience before working on production-level code.

For IBM Developer Advocates, a tutorial provides the opportunity to show real-world benefits of a tool or process. It gives you a platform to engage developers with IBM and open source technologies.

An effective tutorial guides the developer towards a goal by providing clearly defined steps. Whenever possible, you’ll amplify these steps by including working code: samples, demos, snippets and “try me” examples. This code will demonstrate to developers what they’re trying to accomplish, and act as a signpost along the way to their ultimate destination.

Be sure to provide tips, best practices, and potential trouble spots as your audience completes the steps. Your tutorial should be realistic, not frustrating. Teach developers how to achieve a goal, show them the practical benefits of the technology, and highlight lessons learned along the way.

TAKE-BACK TIPS

Include code. Remember who your audience is. Developers live, breathe, and want code – give it to them!

Clarity is crucial. You can’t be informal or fuzzy when you’re providing detailed steps. Your instructions need to be clear and your language precise. Test your steps repeatedly, and have the tutorial reviewed by an editor.

Highlight achievements. Everyone wants to know they’ve been successful. Make sure you highlight what your audience has learned how to do. Tell them when they’ve implemented a new process. Applaud a new end-to-end implementation. Encourage them to keep learning. You’ll build trust and engagement.
Meetups

One of the best ways to engage with a developer community is to lead or participate in a local meetup event. Meetups provide the opportunity to share technical knowledge and meet other developers in a social setting.

Meetup participants can learn a new technology in a casual environment. Because meetings often repeat on a monthly schedule, developers can build their skills, try out solutions, come back with questions to challenges they’re facing, and get answers from local experts.

Participants can build their skill-set and professional network and overcome public speaking anxiety. These small, friendly gatherings are excellent springboards for developers looking to build their eminence and experience.

If you want to host a meetup, aim for a clear schedule of repeating events loosely structured around a 2-3 hour agenda. Your meeting can include a single presentation, several lightning talks, or a panel discussion. Make sure you provide adequate time for networking and discussion before and after the core presentation so attendees can engage speakers and learn more from each other. It’s a good idea to provide refreshments or host your meetup at an establishment where participants can eat and drink what they want.

Content is always the most important ingredient, but a truly great meetup group provides high quality speakers, a fun, modern venue, and a diverse crowd of participants.

TAKE-BACK TIPS

- **Pay attention to detail.** A great meetup needs compelling content, engaging speakers, a great venue, and committed participants. Start with content and pay attention to details to make the experience delightful for developers.

- **Be prepared to devote time.** It’s not easy to organize a meetup. It takes time, commitment, and enthusiasm. Before you create a new group, make sure you check to see if there are existing
groups in your area. It might be more effective to join and contribute to an established meetup.

- **Help is available.** There are all kinds of videos and presentations to help you organize your meetup. Start with this presentation and recording on how to find and organize a great meetup group: [http://bit.ly/cf-meetup](http://bit.ly/cf-meetup).
Conferences

Technical conferences are an ideal place to make a deep dive into a specific technical area and get exposure to new topics, expert speakers, and peer developers.

There are conferences on every conceivable technical topic, and IBM Developer Advocates are encouraged to attend these events whenever possible. However, you must plan carefully because of limited budgets for registration and travel.

Look for well-established conferences that have a track record of quality content. But don’t ignore smaller gatherings, or even a first conference focused on a project or technology with potential to grow in popularity. Be sure to look at a diverse set of events; when you attend a conference as an IBM Developer Advocate, you have a great opportunity to maximize your exposure to important established and emerging developer communities.

TAKE-BACK TIPS

- **Watch for “call for papers.”** The easiest way to get management approval to attend a conference is to have a talk accepted as part of the agenda. Practice your talks at meetups to help convince speaker selection committees to accept your talk.
- **Start big.** Put your initial focus on the bigger events in your general region and register as an attendee. You’ll get familiar with the conference experience and will learn how to make a bigger impact at subsequent smaller events.
- **Tell a friend.** Brainstorming with your colleagues on presentation proposal ideas will help you narrow your focus and improve your submission quality. If you have a co-speaker, you can share the workload and reduce the possibility of errors. It helps to have a backup if your demo doesn’t launch!
Become an IBM Coder

IBM Coders are passionate about building great software. The IBM Coder program is all about focusing that passion. When you become an IBM Coder, you’ll connect with technology pacesetters and coding groundbreakers. You’ll be at the forefront of new coding techniques and will help to build stronger technology communities. Simply put, you will be helping to lead the next wave of the technology revolution.

Become an IBM Coder!

It’s easy to become an IBM Coder. There’s no cost to join, and you get immediate practical benefits:

- **Access to developer resources.** IBM Coders are leaders in the development community. When you become an IBM Coder, you get access to educational resources, guides, and best practices that will help you optimize your code and establish yourself as a technology leader. Whether you’re looking to organize a meetup, fork an open source project, or complete a tutorial on a hot new programming language, you’ll find practical instructions and examples from coders just like you who are in the field every day.

- **Notifications about new developer journeys, open source projects, and more.** IBM Coders are problem-solvers and innovators. Our global team is busy producing developer journeys to show you how to achieve programming goals quickly. As an IBM Coder, you’ll be notified first whenever we publish a new journey. We’ll also let you know about exciting new open source projects that you can use as-is or as the starting point for your own programming effort.

- **Advice from a developer advocate.** IBM has a broad network of developer advocates around the world who can help you with your meetups, hackathons, workshops, or any other activity around technology that you are passionate about. As an IBM Coder, you get exclusive access to these technology leaders;
their sole focus is to engage with you and help you make your community successful.

- **Achieve recognition and rewards.** We’ll soon announce details about how you can gain recognition and win prizes for being an active IBM Coder and community leader. We want you to reap the rewards for helping to build the IBM Code Community. You’ll be recognized for such community-building activities as:
  - Participating in an IBM-hosted meetup
  - Hosting a meetup using IBM-provided materials
  - Meeting with an IBM Code developer advocate
  - Inviting an IBM Code developer advocate to present at your meetup
  - Contributing to an IBM Code open source project
  - Present journey content at a conference or meetup