HOW TO BUILD AN ENGAGED DEVELOPER COMMUNITY

CONTENTS

INTRODUCTION

Software developers are driven by the passion to solve problems. They're eager to share their expertise and learn from the experience of their peers. If you unleash these like-minded folks in the right smartly-built, not overly-controlled community environment, amazing things start to happen. It's the power of shared knowledge.

It's this power that makes community the cornerstone of successful developer relations (devrel) programs. An online developer community is the ideal platform to facilitate your devrel goals. It's where you give developers the freedom to play with your company's product, to put its features and functionality to the test, and share their findings, questions, and comments with other developers and customers all over the world.

A community is the ideal place to quench developers' insatiable thirst for quality content, tools, tips, best practices, industry news and trends, social networking, education, and any other resources that help them grow professionally and do their jobs faster, better, and more efficiently.

Getting developers to your community is only the beginning. The trick is in spiriting their motivation and keeping them engaged, focused, and willing to contribute to your product.

KNOW YOUR AUDIENCE

Before outlining goals for your community, understand the needs and wants of your members and your management team.

Research how to position your organization and developer community as a strong online resource. Who's posting about your

company? What are they saying? How are competitors positioning themselves in the marketplace?

By analyzing the current situation, you'll have the knowledge necessary to start relevant conversations, build relationships, and establish a foundation that will foster developer engagement and community growth.
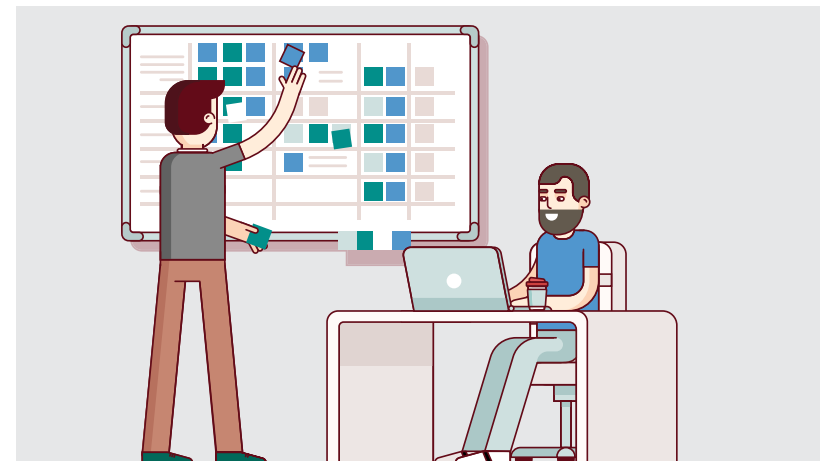
SET A CLEAR VISION AND GUIDELINES

The **objectives of your online community must be clear** in order to produce a measurable return on your investment. Guidelines must be set for community engagement. After clarity is established on the purpose of your community and the rules that guide it, successful engagement will follow.

Company stakeholders should understand the roles they play and the expectations for their participation. Your community manager should monitor engagement to ensure it aligns with your organization's objectives.

MAP STRENGTHS TO DIFFERENT ROLES

Successful communities look for members with strengths to fill different roles. Here's where outreach from your developer relations manager or developer evangelist can be of great help. Seek out developers whose skills align exceptionally well with the following roles:

- Writers to compose blogs, articles, and other community content.

- Speakers to present at industry events, webinars, and community-arranged meetups.

- Coders to write code samples and provide assistance at hackathons or other developer activities.

POSITION YOUR COMMUNITY AS PART OF THE BRAND AFFINITY

Your developer community should reinforce your company's overall value proposition and promote lasting customer relationships built on trust and shared common beliefs. It's a showcase for values such as transparency, communication, innovation, and user satisfaction.

Position your community as a central feature in new customer relationships. Also, create awareness for your community. Make it easy to find on your website and promote it in marketing emails and other collateral, customer communications, and internal newsletters.
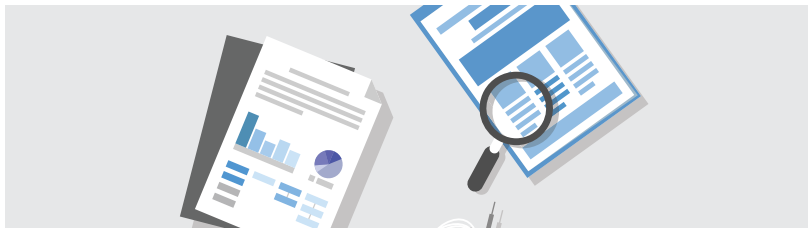
THE BUILDING BLOCKS OF COMMUNITY ENGAGEMENT

Most developers are do-it-yourselfers who are short on time. Don't make them work hard to use your community. Ensure that community resources are easily accessible and in formats developers know well. Your developer portal should be easy to access as well, with simple sign-on and unencumbered navigation.

Furnish your community with collaboration and information features developers expect: forums, blogs, a resource library, technical content, code snippets, SDKs, and APIs.

Now, let's look at the primary building blocks that drive developer engagement in the community:

- Knowledge sharing and transfer
- Communication and feedback
- Incentives and rewards
- User-generated content

CULTIVATE KNOWLEDGE TRANSFER

The mind of a software developer is filled with copious amounts of knowledge. Getting that valuable knowledge out of their heads and captured in a shared community can save both developers and customer service agents valuable time.

Knowledge transfer is replicating the expertise, wisdom, and skills possessed by software developers in the heads and hands of coworkers, developers, and community members. Achieving

knowledge transfer requires inserting collaborative technologies deep into processes while also providing incentives for collaborative behaviors—ultimately transforming the way developers turn knowledge into action.

DEVELOPER FORUMS

Developers enthusiastically spread knowledge on forums, making them terrific places for developers to share their expertise and skills.

Prune your developer forum often. You want to minimize the noise on forums so developers can find the information most relevant to them and locate other members with similar interests easily.

INSIDER GROUPS

Engagement depends on authentic relationships between your company and its community. An effective way to build relationships and foster engagement is an insider group. Insiders are influencers, and they possess insight into the needs and interests of your community.

An insider group enables community experts to have a sense of ownership. Listen to their advice and opinions to create a more beneficial experience for your community members. For example, in DZone.com we leverage our own experts, called Zone Leaders, to lead by example in the community.

FOSTER OPEN COMMUNICATION AND ONGOING FEEDBACK

Open communication between your organization and community experts is crucial. Incorporating user feedback will ensure a more beneficial experience and keep developers coming back to your community.

As a best practice, actively communicate with your experts to gauge user satisfaction. Regularly send out surveys soliciting feedback, or create a dedicated group within your community where experts can leave suggestions and share ideas on how to build a stronger community.

Thank experts for their feedback. Even if you can't make every suggested improvement, they want to be acknowledged and feel that their opinions matter.

Your community manager should actively engage with developers as well. The community manager can also generate engagement by promoting your community at speaking events, conferences, and other venues where developers congregate.

OFFER INCENTIVES AND REWARDS

Experts are the backbone of a community's success. It's important to identify and recognize developers who actively share their

insights, provide support, post content, and drive discussions with fellow members.

Gamification is a pivotal feature of developer relations software. Use gamification features such as badges, reputation points, and topic expertise to reward quality engagement and incentivize other community members to participate on a regular basis.

The ability to recognize and reward key community contributors through moderation privileges makes maintaining your community considerably easier. With AnswerHub, for instance, it's easy to identify the most active members in your community and automatically reward them.

AnswerHub's gamification features allow you to drive desired behaviors by rewarding community members for completing specific tasks. For example, if your goal for the month is to get more users voting on questions, you can design a badge that will encourage users to do just that. Perhaps you want more users to submit answers. You can simply customize a badge to encourage this action. This feature gives you precise control over the actions you want users to complete.

Developers are motivated by recognition and rewards. Choose a developer relations platform that makes it fun and rewarding for developers to provide and find answers.

POST QUALITY USER-GENERATED CONTENT

High-quality user-generated content is essential to a successful developer community. Developers want answers. They don't want to sift through useless information.

If you want developers to come back, you've got to get them to trust you. Developers need to have trust in the content they use. The content needs to solve their problems and continue to improve their skills. Fluff won't cut it with developers.

For example, a wildly successful type of content in the **DZone.com** community is more than 250 **Refcardz**, branded as the world's largest library of technical cheat sheets. Most Refcardz are 5 to 8 pages, written for experienced developers who want to learn about a particular technology without reading the entire manual or book. All Refcardz are created to high-quality standards and undergo a two-round peer review before being published in the community. Developers are solicited in the community to submit Refcardz topics.

DZone Guides dive deeper, providing insight and expert analysis of leading industry topics.

In addition to tapping developers for topics and content, use your management team or company's subject matter experts to come up with fresh content every day (ideally) or at least once a week.
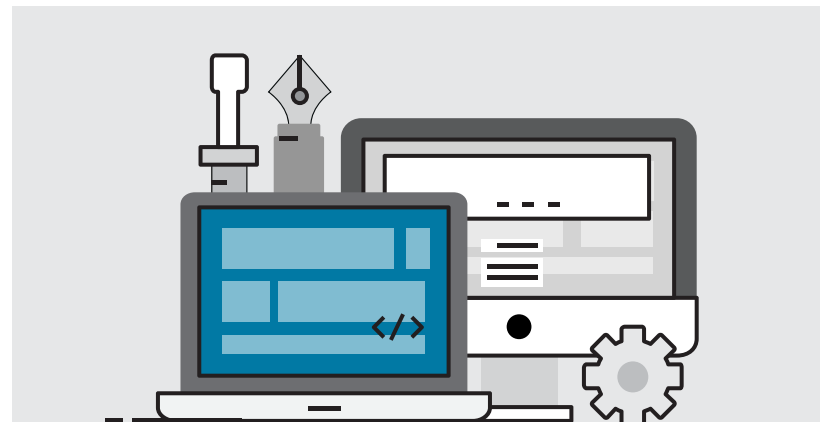
RICH MEDIA CONTENT

Images and videos are a great way to provide answers and get developers engaged in the community. AnswerHub allows users to embed rich media directly into the body of questions, answers, and comments.

BLOGS

The most useful blogs focus on knowledge sharing, not product speeds and feeds. Plant seeds for creativity and innovation with blogs about new technologies, thought leadership, and outcome-oriented product use cases.

Ask your developer relations manager to help tap the self-identified writers on your developer team, in addition to community members, to generate relevant, high-caliber content.

DOCUMENTATION

Likewise, documentation in your community needs to be up to date and top notch. The quality of documentation such as tutorials, product guides, reference cards, and application code can make or break a developer's experience.

For developers, the concept of writing (outside of coding) can seem daunting. Developers need to hone this skill because business is increasingly collaborative. Open source projects invite worldwide participation, and developers who take on the task of writing clear, concise findings of their products help move the industry forward as a whole.

When developers transfer their knowledge through documentation, they contribute to the growing **beneficial engine of self-service support**. Studies continually tell us that more and more people prefer self-service over contacting a support agent. Documentation produced by developers can help make your community a solid self-service resource:

- Coworkers have a place to go to find, share, and research knowledge that would otherwise be in the developer's head. Having clear guidelines for staff allows processes to be standardized across your business to achieve consistency.

- Customer Support has an alternative place to go for answers rather than interrupting a developer.

- External users have a place to go rather than dialing a toll-free number or sending an email.

Not only is writing important, but so is collecting, sharing, and storing the documentation in an accessible, collaborative space. Developers want the information they need to get their work done from anywhere. Community documentation is a great source of collective knowledge that can be accessed any time.

DEVELOPER ENGAGEMENT: KEY POINTS TO PONDER

Following is a summary of the **key ways to help promote developer engagement** and increase overall productivity in your community:

• Create an environment that's open, direct, and trusting. Make it easy for developers to find what they're looking for in a quick and easy way. The easier you make it, the more frequently they'll come back and encourage their colleagues to use the site as well.

• Embrace developers' natural tendency to share their knowledge and help others by enabling them to do so in an online community that's open and accessible 24/7 from any device.

• Encourage an ongoing dialogue among developers with a knowledge base of questions, answers, solutions, and code snippets in an environment that eliminates redundant questions and tedious, endless threads.

• Encourage collaboration across teams, projects, geographies, and management. Senior developers, engineers, and architects likely have more knowledge to share. They can lead the way in developing and expanding the community of knowledge and culture of sharing.

• Be open to feedback from developers and other community members. Encourage them to take ownership by moderating questions and answers in the areas where they have the most experience and knowledge.

• Ask developers to share what they've learned from their community engagement. These testimonials will be invaluable as you introduce the community to new developers.

• Use gamification to recognize topic experts and empower community members to upvote the best answers. People who are trusted in the community earn more power and responsibility.

• Listen intently and respond in a timely manner to enhance productivity and promote developer engagement. A satisfied developer will recruit others to the community.

• Facilitate real-time communication and sharing to enhance your community's reputation as a collaboration resource for knowledge, questions, and answers around technology development.
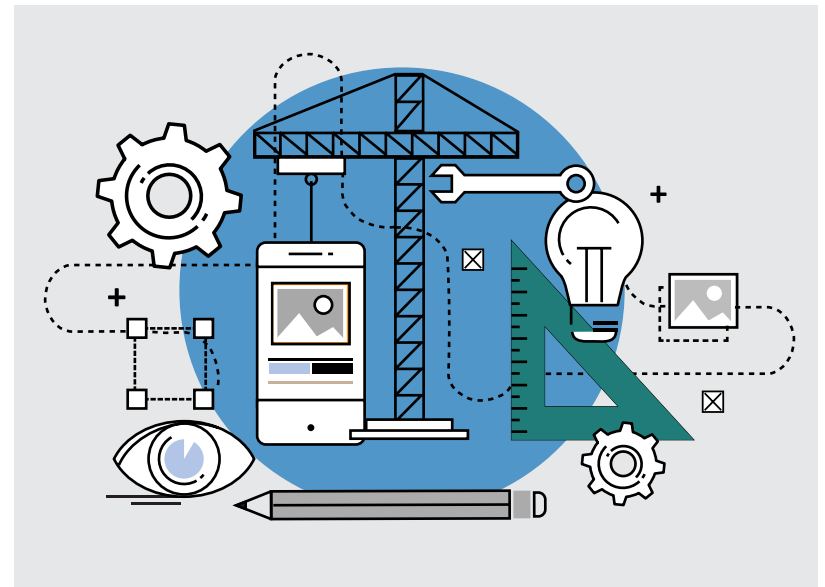
- Promote a culture of shared knowledge by incorporating it into your organization's vision, mission, and values. Developers should lead the way in how your community improves productivity, efficiency, effectiveness, and quality of life.

- Implement solid community tools because engagement can't be created in a community built on tools that are buggy or difficult to understand. The technology underlying your community must be solid. One bad user experience can result in a lost developer and ultimately create a detrimental impact on engagement.

The more you can do to get developers to share their knowledge and to use the community to answer their questions, the more productive, empowered, and engaged your developers will be.

MEASURE ENGAGEMENT, MEASURE SUCCESS

In **successful developer communities**, engagement and knowledge sharing are closely monitored. Community managers are responsible for monitoring the health of your community. Your manager should be proactive in collecting and analyzing data.

Ensure that your developer community platform allows for solid metrics tracking. **AnswerHub**, for example, has an advanced analytics system that enables community managers to find potential gaps in your organization's needs and take action to correct them. AnswerHub tracks metrics such as page views, number of questions asked, unanswered questions, new users, and more.

Track your **community success** and share it. Is the community gaining plenty of new visitors? Are you nailing search engine optimization

WEBINAR

MEASURE THE SUCCESS OF YOUR ONLINE COMMUNITY

Are business stakeholders asking you to quantify the value of having a developer community? Learn how customers have successfully made the business case for AnswerHub, our developer community platform. Get real-life examples of the metrics they rely on to measure the value and assess the health of their community. Hear from DZone Software COO Jesse Davis, who outlines the key metrics at each lifecycle phase, along with former community manager at Lenovo, Mark Hopkins, who shares his experience getting buy-in from senior management.

Register for this insightful on-demand webinar.

(SEO) with relevant, high-quality content? Are you gaining traffic from your own website? Is your community answering questions in less time? Have you seen an increase in member signups?

Take five minutes to look at your key metrics at least once a week, adjust your strategy, amplify and continue what's working, and take corrective actions sooner rather than later on what's not.

Remember, too, that your developer portal is where developers begin their journey. From the portal, you can start tracking conversion metrics such as the number of links from the Internet to your portal.

ANSWERHUB CHECKLIST

Thanks for your interest in our eBook. To help you get started on building an engaged developer community, here's a checklist of questions from AnswerHub.

☐ **Know your audience.**
  ▶ What do your developers want?
  ▶ What do they need?
  ▶ What drives them?

☐ **Align the wants and needs of your audience with the wants and needs of your management team.**
  ▶ Position yourself as a strong online resource.

☐ **What are people saying about your company?**
  ▶ How do your competitors position themselves?

☐ **Set a clear vision and guidelines for your community.**
  ▶ Do your goal and objectives map to measurable returns on investment?
  ▶ Do your company stakeholders understand their roles and expectations?
  ▶ Do you have guidelines for community engagement?

☐ **Put mechanisms in place to ensure that your community is promoted in external and internal marketing communications.**

☐ **Make knowledge transfer easy and pervasive throughout your community.**
  ▶ Are you putting features in your community that facilitate knowledge transfer such as forums, blogs, and insider groups?

☐ **Keep lines of communication open.**
  ▶ Do you have mechanisms in place to communicate with developers easily and often?
  ▶ Do you have mechanisms for requesting developer feedback?

☐ **Offer incentives and rewards.**
  ▶ Are you incorporating gamification within your community?
  ▶ Does your community have a system for automatically rewarding member contributions?

☐ **Ensure high-quality user-generated content.**
  ▶ Do you have a filter for stripping marketing fluff from your content?
  ▶ Are you proactively seeking members with writing aptitude and rewarding them appropriately?

☐ **Measure community engagement.**
  ▶ Is your community manager on board with consistently monitoring the online health of your community?
  ▶ Do you have mechanisms in place to track metrics such as page views, upvotes and downvotes, and response times?
  ▶ Are you ensuring that metrics are pulled and examined regularly?

REFERENCES:

DZone Software, eBook, "Success from the Ground Up: Building an Effective Online Community" landing.dzonesoftware.com/success-from-ground-up-building-an-effective-online-community

JWPlayer, "Building a Better Developer Community: DevRelCon Takeaways," April 22, 2016 jwplayer.com/blog/devrelcon-takeaways/

OpenChannel, "How to Build an Awesome Developer Community," August 19, 2014 openchannel.io/blog/how-to-build-an-awesome-developer-community/

AnswerHub

Helping Companies Harness

Developer Influence and Expertise

ANSWERHUB.COM

(919) 238 7100

INFO@DZONESOFTWARE.COM