



From DevEx Disaster to Delight

How to Champion a DevEx Revolution in Your Organization

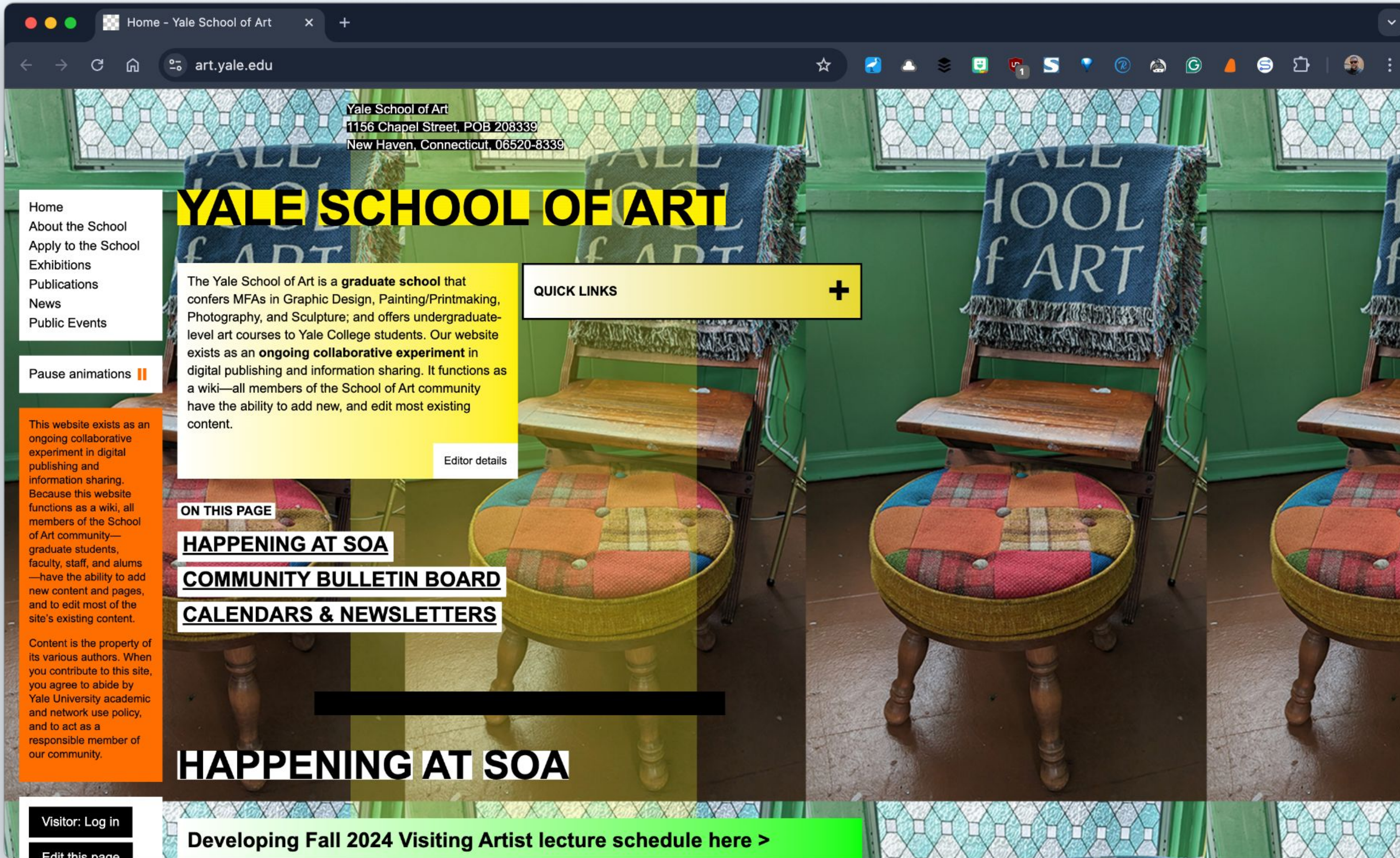
Jeremy Meiss

Director, Developer Relations

What we're going to cover

- What is Developer Experience (DevEx)?
- The impact of Good (and Bad) DevEx
- Strategies for improving DevEx
- Becoming the DevEx Champion you always dreamed of being

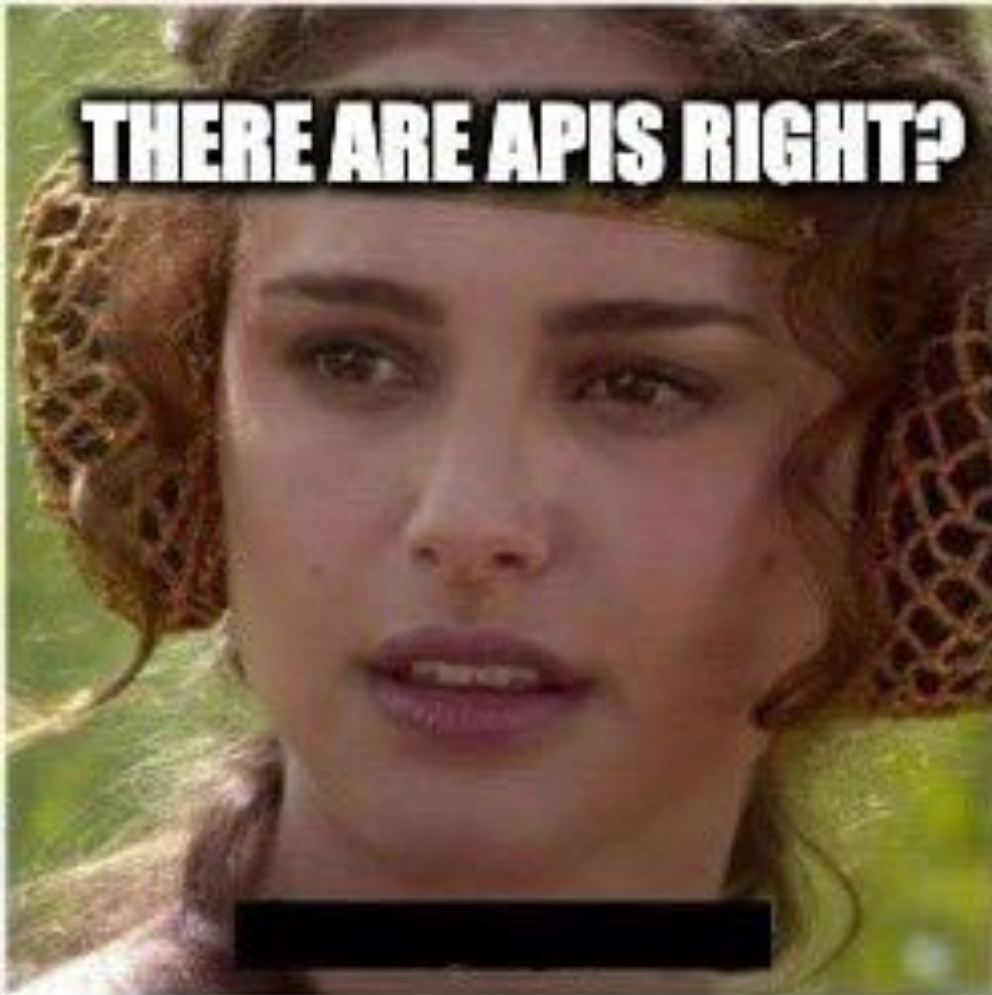
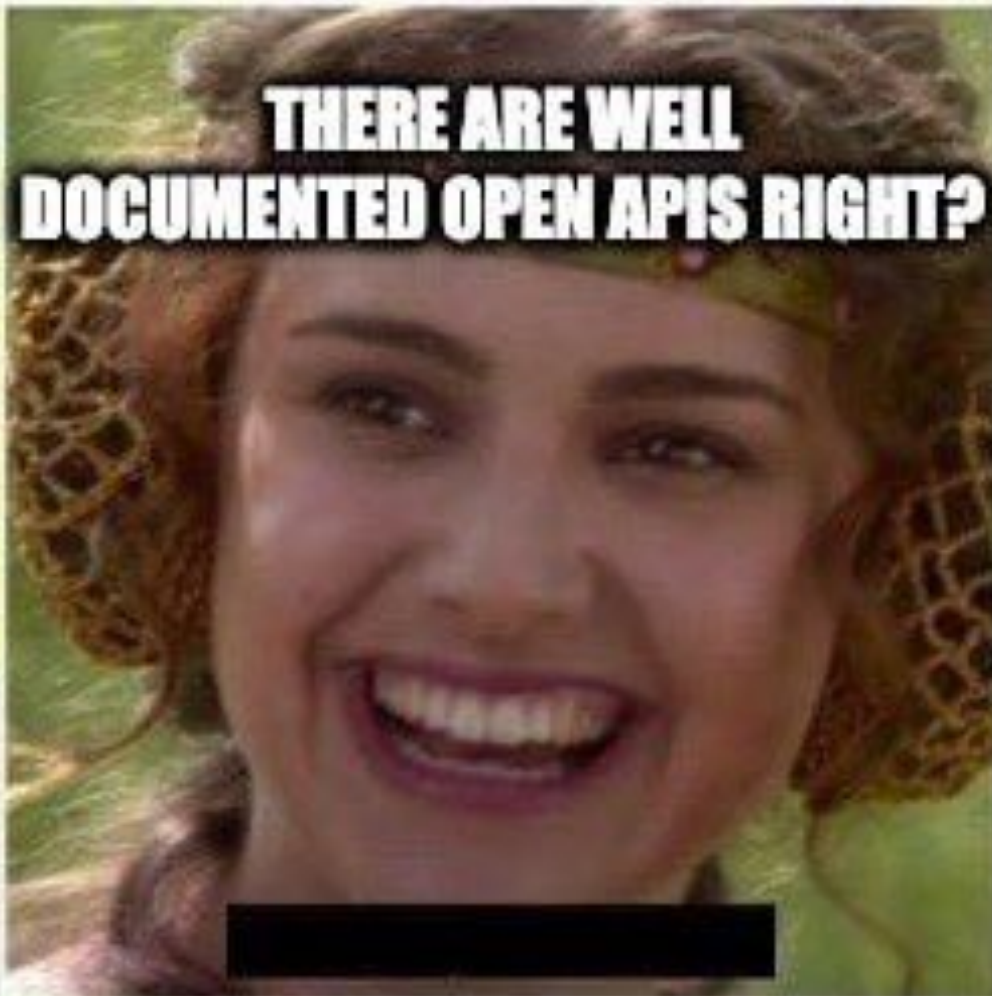
DevEx disasters...





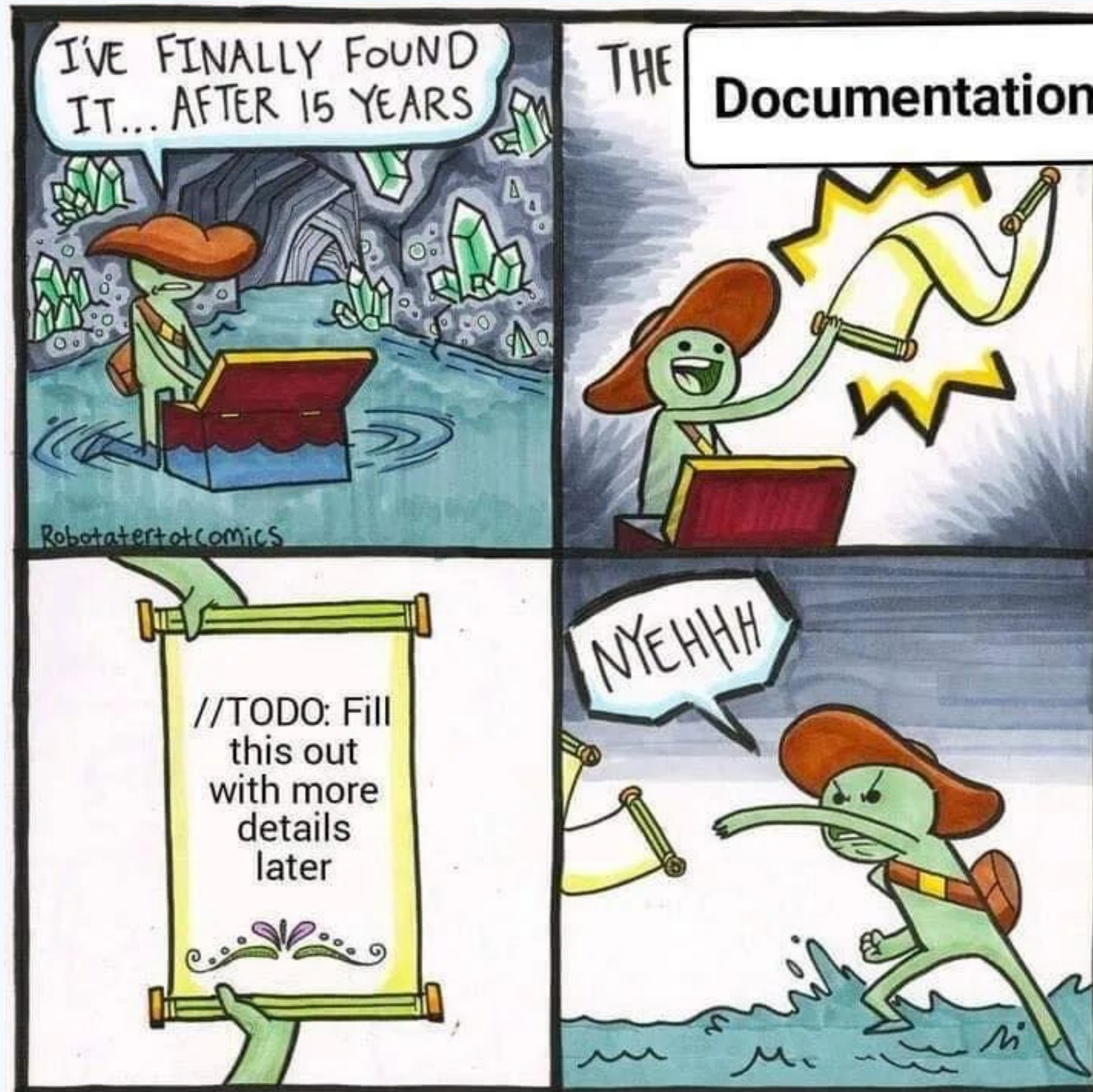
DevEx misses...

- Poorly documented features (or bugs)



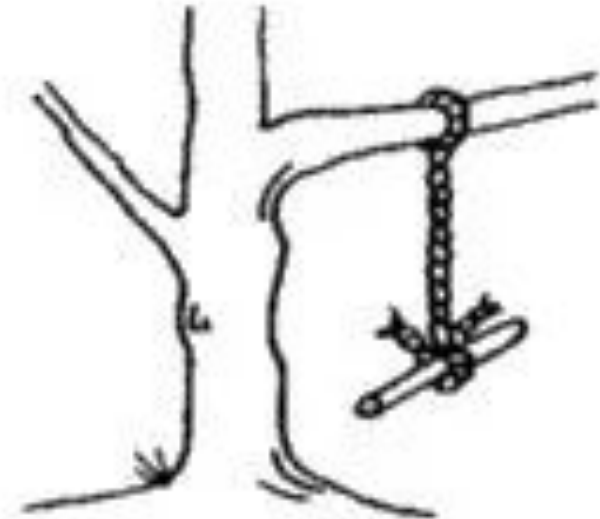
DevEx misses...

- Poorly documented features (or bugs)
- Missing OpenAPI spec, or even an API

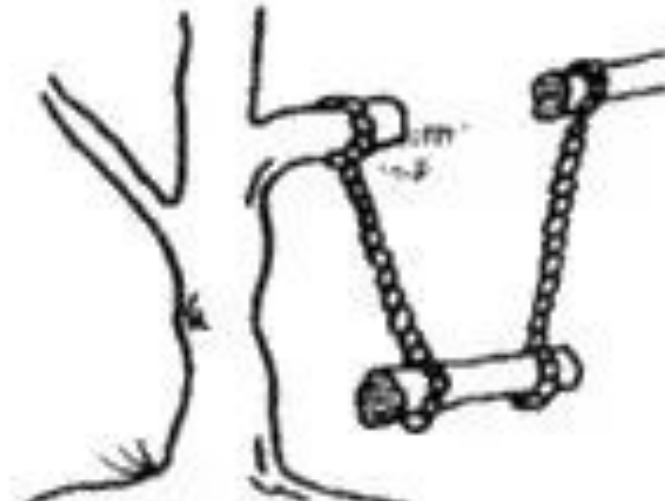


DevEx misses...

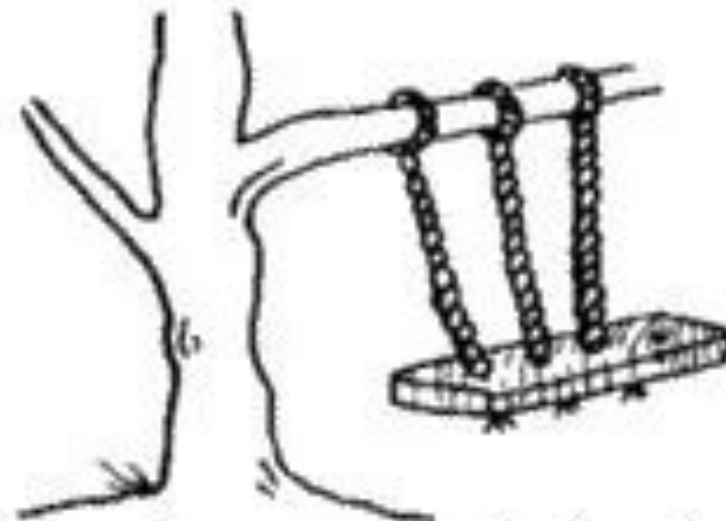
- Poorly documented features (or bugs)
- Missing OpenAPI spec, or even an API
- Downloading documentation... as a PDF, or access-gated



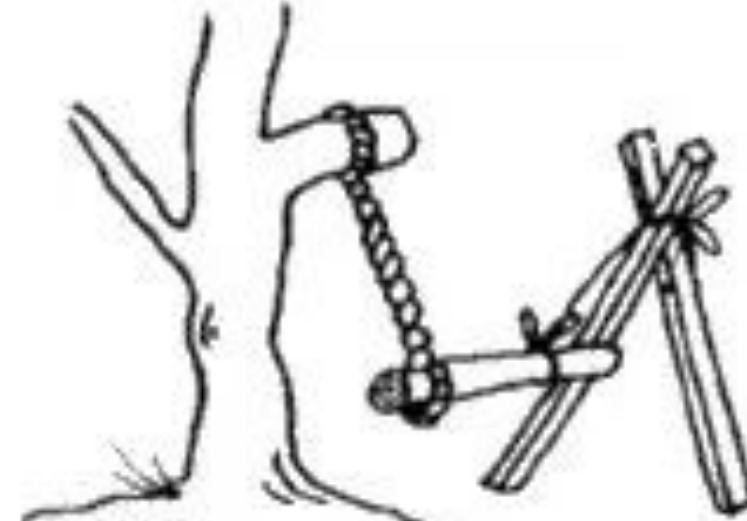
What the user asked for



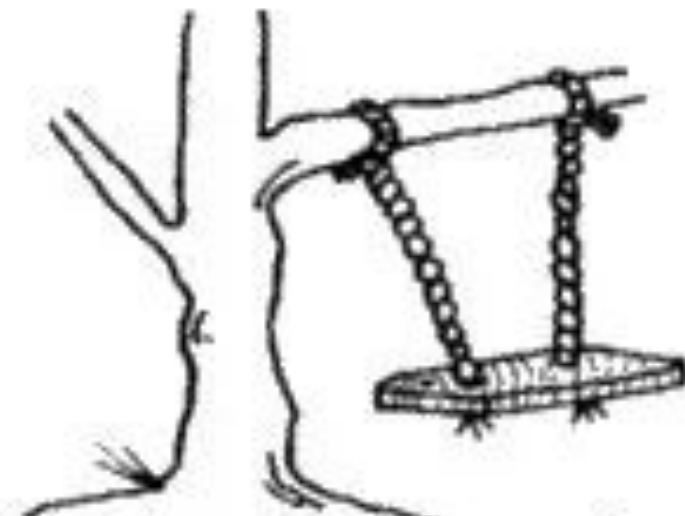
How the analyst saw it



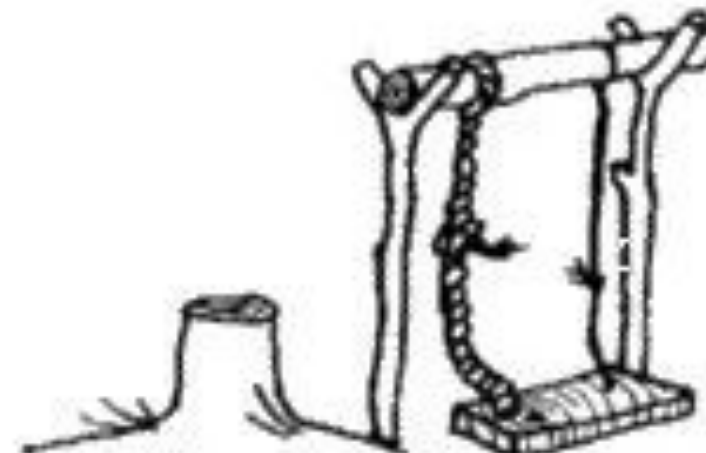
How the system was designed



As the programmer wrote it



What the user really wanted



How it actually works

DevEx misses...

- Poorly documented features (or bugs)
- Missing OpenAPI spec, or even an API
- Downloading documentation... as a PDF, or access-gated
- Missing examples... of anything



Ramiro Berrelleza 

@rberrelleza

Long time ago, in a galaxy far away, I worked at a team where our CI environment was so different from local or production, that the only realistic option way to validate a change was in prod. So we would commit the change, rerun CI jobs until they were green, deploy to prod, and then monitor the logs for about 1 hour. If no major errors were logged after that you were good to go 😄

12:39 AM · Aug 3, 2024

DevEx misses...

- Poorly documented features (or bugs)
- Missing OpenAPI spec, or even an API
- Downloading documentation... as a PDF, or access-gated
- Missing examples... of anything
- “CI as Magic 8-Ball”

What is Developer Experience (DevEx)?

DevEx is *more* than just your parent's SDLC



Developer Experience != Developer Productivity

DevEx also isn't new



arXiv > cs > arXiv:1312.1452

Computer Science > Software Engineering

[Submitted on 5 Dec 2013]

Developer Experience: Concept and Definition

Fabian Fagerholm, Jürgen Münch

New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments. User experience is a concept that captures how persons feel about products, systems and services. It evolved from disciplines such as interaction design and usability to a much richer scope that includes feelings, motivations, and satisfaction. Similarly, developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance. This article motivates the importance of developer experience, sketches related approaches from other domains, proposes a definition of developer experience that is derived from similar concepts in other domains, describes an ongoing empirical study to better understand developer experience, and finally gives an outlook on planned future research activities.

Comments: 5 pages. The final publication is available at [this http URL](#)

Subjects: **Software Engineering (cs.SE)**

Cite as: arXiv:1312.1452 [cs.SE]

(or arXiv:1312.1452v1 [cs.SE] for this version)

<https://doi.org/10.48550/arXiv.1312.1452>

Journal reference: Proceedings of the International Conference on Software and System Process (ICSSP 2012), pages 73–77, Zurich, Switzerland, June 2–3 2012

REF: F. Fagerholm and J. Münch, "[Developer experience: Concept and definition](#)," 2012 International Conference on Software and System Process (ICSSP), Zurich, Switzerland, 2012.



DevEx also isn't new



arXiv > cs > arXiv:1312.1452

Computer Science > Software Engineering

[Submitted on 5 Dec 2013]

Developer Experience: Concept and Definition

Fabian Fagerholm, Jürgen Münch

New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments. User experience is a concept that captures how persons feel about products, systems and services. It evolved from disciplines such as interaction design and usability to a much richer scope that includes feelings, motivations, and satisfaction. Similarly, developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance. This article motivates the importance of developer experience, sketches related approaches from other domains, proposes a definition of developer experience that is derived from similar concepts in other domains, describes an ongoing empirical study to better understand developer experience, and finally gives an outlook on planned future research activities.

Comments: 5 pages. The final publication is available at [this http URL](http://this.url)

Subjects: **Software Engineering (cs.SE)**

Cite as: arXiv:1312.1452 [cs.SE]

(or arXiv:1312.1452v1 [cs.SE] for this version)

<https://doi.org/10.48550/arXiv.1312.1452>

Journal reference: Proceedings of the International Conference on Software and System Process (ICSSP 2012), pages 73–77, Zurich, Switzerland, June 2–3 2012

REF: F. Fagerholm and J. Münch, "[Developer experience: Concept and definition](#)," 2012 International Conference on Software and System Process (ICSSP), Zurich, Switzerland, 2012.

“New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments.”



DevEx also isn't new



arXiv > cs > arXiv:1312.1452

Computer Science > Software Engineering

[Submitted on 5 Dec 2013]

Developer Experience: Concept and Definition

Fabian Fagerholm, Jürgen Münch

New ways of working such as globally distributed development or the integration of self-motivated external developers into software ecosystems will require a better and more comprehensive understanding of developers' feelings, perceptions, motivations and identification with their tasks in their respective project environments. User experience is a concept that captures how persons feel about products, systems and services. It evolved from disciplines such as interaction design and usability to a much richer scope that includes feelings, motivations, and satisfaction. Similarly, developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance. This article motivates the importance of developer experience, sketches related approaches from other domains, proposes a definition of developer experience that is derived from similar concepts in other domains, describes an ongoing empirical study to better understand developer experience, and finally gives an outlook on planned future research activities.

Comments: 5 pages. The final publication is available at [this http URL](#)

Subjects: **Software Engineering (cs.SE)**

Cite as: arXiv:1312.1452 [cs.SE]

(or arXiv:1312.1452v1 [cs.SE] for this version)

<https://doi.org/10.48550/arXiv.1312.1452>

Journal reference: Proceedings of the International Conference on Software and System Process (ICSSP 2012), pages 73–77, Zurich, Switzerland, June 2–3 2012

REF: F. Fagerholm and J. Münch, "[Developer experience: Concept and definition](#)," 2012 International Conference on Software and System Process (ICSSP), Zurich, Switzerland, 2012.

"...developer experience could be defined as a means for capturing how developers think and feel about their activities within their working environments, with the assumption that an improvement of the developer experience has positive impacts on characteristics such as sustained team and project performance."



A working definition of DevEx

*"...the **journey** of developers as they learn and deploy technology, which if successful, focuses on eliminating obstacles that hinder a developer or practitioner from achieving success in their endeavors."*

-Jessica West, Co-Founder, DevEx Institute

DevEx includes *every* interaction a developer/ops practitioner has with systems, tools, and processes

Key aspects of DevEx

1. Tools & Automation

- Code editors
- Version control system
- Deployment pipelines

Key aspects of DevEx

1. Tools & Automation
2. Development Environment Setup
 - Streamlined onboarding (i.e. IDPs)
 - Consistent configurations

Key aspects of DevEx

1. Tools & Automation
2. Development Environment Setup
3. Documentation, documentation
 - Clear, up-to-date documentation
 - Easy access to resources, trainings
 - Regular team audits

Key aspects of DevEx

1. Tools & Automation
2. Development Environment Setup
3. Documentation, documentation
4. Collaboration & Communication
 - Efficient communication channels
 - Knowledge-sharing platforms
 - Code reviews

Key aspects of DevEx

1. Tools & Automation
2. Development Environment Setup
3. Documentation, documentation
4. Collaboration & Communication
5. Culture & Feedback
 - Positive work environment
 - People and culture before tools
 - Opportunities for feedback and growth



Jeremy (#DevRel & #DevEx) 🇺🇸 🇺🇦

@IAmJerdog



If your company does not already have a process for gathering feedback (internal & external) on your product and/or the tools you use, you will not have a good Developer Experience, and I seriously question your commitment to it.

10:57 AM · Jan 25, 2024

The Impact of DevEx

Impact of poor DevEx

1. Increased development time and costs
 - Repetitive tasks
 - Inefficient tools
 - Constant firefighting

Impact of poor DevEx

1. Increased development time and costs
2. Lower code quality and frequent bugs
 - Lack of proper testing
 - Poor collaboration
 - Rushed development cycles
 - Delayed releases

Impact of poor DevEx

1. Increased development time and costs
2. Lower code quality and frequent bugs
3. High developer turnover
 - Frustration with development environment
 - Lack of growth opportunities
 - Burnout

Impact of poor DevEx

1. Increased development time and costs
2. Lower code quality and frequent bugs
3. High developer turnover
4. Stagnant innovation and missed opportunities
 - Focus on fixing issues rather than building new
 - Loss of competitive edge

Benefits of good DevEx

1. Increased productivity and faster development cycles
2. Improved code quality and fewer bugs
3. Reduced developer burnout and higher morale
4. Attracting + retaining top talent
5. Foster a sense of ownership and accountability
6. Improved collaboration and communication
7. Accelerated time-to-market + stronger competitive advantage + increased customer satisfaction

Strategies for improving DevEx

Improving DevEx in your organization

1. Foster a positive culture
2. Streamline the workflow(s)

Improving DevEx: Fostering a positive culture

Clear and concise documentation

- Encourage knowledge sharing
- Create easily accessible resources to reduce toil + empower

Promote collaboration and communication

- Facilitate code reviews
- Implement comms to foster teamwork + problem solving

Champion well-being and growth

- Encourage feedback, up and down
- Recognize achievements
- Create a sense of belonging

Improving DevEx: Streamline the workflow

Tools and Automation

- Explore tools which are highly regarded in your field
- Automate repetitive tasks wherever possible

Improving DevEx: Streamline the workflow

Tools and Automation

- Explore tools which are highly regarded in your field
- Automate repetitive tasks wherever possible

Standardize environment setup

- Use config management tools
- Streamline onboarding for all team members

Examples:



Becoming the DevEx Champion you've always dreamed of

Championing DevEx in your organization

- **Start small, build momentum**
 - Identify low-hanging fruit
 - Focus on areas of greatest pain
 - Celebrate small wins

Championing DevEx in your organization

- **Start small, build momentum**
- **Get buy-in from key stakeholders**
 - Communicate the benefits
 - Involve key decision-makers in the solution
 - Build a coalition of allies



Championing DevEx in your organization

- **Start small, build momentum**
- **Get buy-in from key stakeholders**
- **Develop a DevEx champion network**
 - Identify passionate individuals
 - Create a community of practice
 - Foster a culture of continuous improvement

Championing DevEx in your organization

- **Start small, build momentum**
- **Get buy-in from key stakeholders**
- **Develop a DevEx champion network**
- **Be prepared to advocate for DevEx improvements**
 - Data driven
 - Address concerns and objections
 - Be persistent, persuasive
 - Offer solutions, not just problems

"DevEx is ruthlessly eliminating whatever prevents your developers (and practitioners) from being successful"

-Me



THANK YOU

 **@IAmJerdog**

jmeiss.me
hachyderm.io/@jerdog
linkedin.com/in/jeremymeiss