MDN MOZILLA DEVELOPER NETWORK

Learning Area Survey 2014

— Results —

ABOUT THIS SURVEY 3

ABOUT MDN 3

WHAT'S MISSING? 3
MDN LACKS OF ORGANIZATION FOR BEGINNERS 3
MDN NEEDS A GLOSSARY AND EXPLANATION OF BASIC CONCEPTS 4
MDN NEEDS TO IMPROVE EXAMPLES (INTERACTIVE, VIDEOS, ETC.) 4
MDN'S EXISTING CONTENT NEEDS TO BE UPDATED AND ADAPTED 5
MDN NEEDS A "FROM ZERO" GUIDE 5
MDN'S CONTENT SHOULD FOCUS ON TASKS RATHER THAN TECHNOLOGIES 5
MISC. 5

ABOUT OTHER WEB SITES 6

CONTROVERSIAL RESOURCES 7
W3SCHOOLS 7
STACKOVERFLOW 7
COMMON BLESSINGS 7
COMMON ISSUES 8

LEARNING FEATURES 9

OTHER FEATURES TO CONSIDER 10

KNOWLEDGE 11

OTHER KNOWLEDGE TO CONSIDER 12
POINTS OF DIFFICULTY 12

ABOUT RESPONDERS 13

HOW DID THEY LEARN THE WEB? 13
WHAT WERE THEIR MOTIVATIONS TO LEARN THE WEB? 13

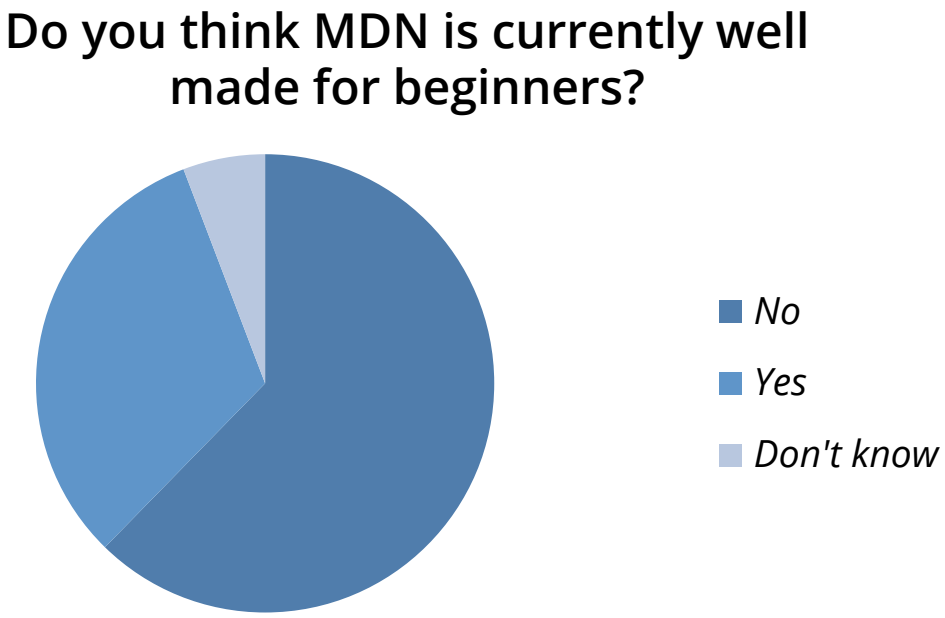
About this survey

This survey is part of the Learning Area project run in 2014 on MDN. The goal of this project is to make MDN a friendly place for people who want to learn about web technologies.

The survey was run in January and February 2014 and was filled by 70 participants.

About MDN

At first glance, we asked about MDN for beginners.



62% of participants think MDN is not well made for beginner, whereas 32% think the opposite and 6% don't know.

What's missing?

We asked for inputs about what's missing, here are the most important feedback and some associated verbatim:

MDN lacks of organization for beginners

- "*a Prerequisites section in the articles will help beginners blaze their own path thru the site.*"

- "*the pre-requisite are quite high and *not mentioned**."
- "*there are too many entries for beginners*"
- "*an organized structure that moves form one tutorial to the next for HTML, CSS, or JS would be great*"
- "*It's not clear where to start.*"
- "*you lack first-level welcome and basic directions*"
- "*beginners need a more step-by-step path to understand how all the different parts of web development fit together*"
- "*There is no direction of 'Start here'*"
- "*the documentation resources are missing a hierarchy*"
- "*we don't know where to start*"
- "*[Needs] Easy path to follow on each topics*"
- "*It is very confusing for a beginner to look for where to start*"
- "*More step by step approach*"

MDN Needs a glossary and explanation of basic concepts

- "*Some sort of a FAQ or a QA site for developers will also help.*"
- "*Wording in articles is often complex: we miss glossaries and definition of basic concepts*"
- "*Have sections describing common problems to be solved*"
- "*Many definitions are weak*"
- "*I suggest we create an 'introduction to' section that covers the basis of everything web development.*"
- "*A basic definition with diagram of how web works and where different languages and platforms are used should be mentioned first.*"
- "*There should be something for those who don't know about the basics of basic languages*"

MDN needs to improve examples (interactive, videos, etc.)

- "*a "try it" feature [...] or links to examples on codepen/jsfiddle*"
- "*it should be interactive like code.org*"
- "*More examples would be great. Plus some screencasts.*"
- "*il manque [...] un espace pour « pratiquer »*" (It miss a space to "try")
- "*More step-by-step code examples*"
- "*some step by step examples*"
- "*allow video tutorial*"
- "*interactive demo [...], it's very useful for test and try the example code.*"

MDN's existing content needs to be updated and adapted

- "*most of the tutorials are very dated*"
- "*The way in which the resources are written are not beginner-friendly.*"
- "*The formatting of reference documentation could be improved.*"
- "*many articles are not focused.*"
- "*Too much information at the hands of beginners.*"
- "*HTML elements descriptions specially adapted for beginners*"

MDN Needs a "from zero" guide

- "*Guides to walk a person thru the journey*"
- "*a guide to start from zero*"
- "*Clear instructions on how to start Web development*"

MDN's content should focus on tasks rather than technologies

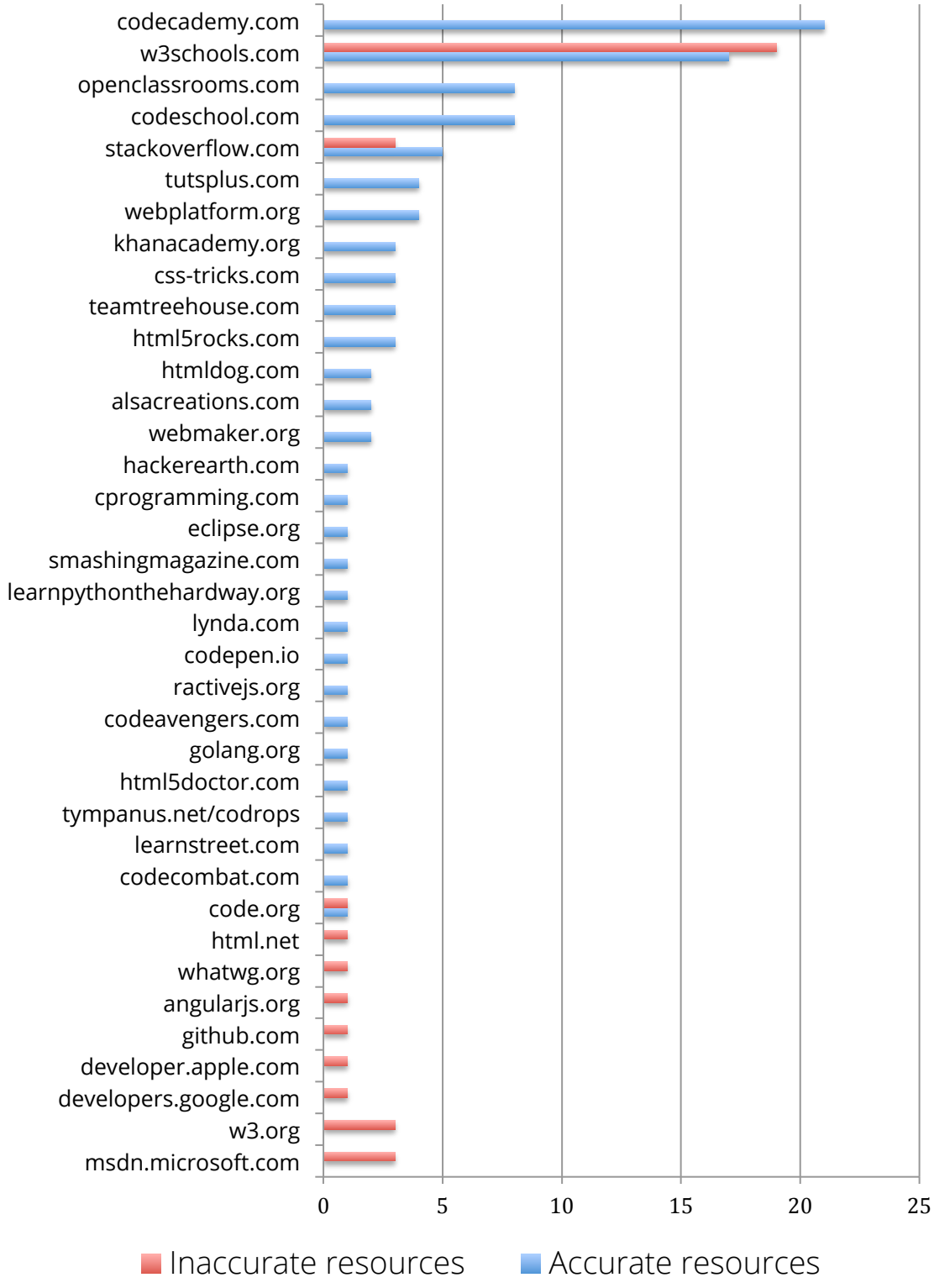
- "*Beginners often look for answer instead of articles.*"
- "*how to assemble [APIs and language features] into something useful.*"
- "*which language is the most useful to be learnt for me (Why?) and what for*"

Misc.

- "*Everything is made/explained for Linux/Mac users. [...] Windows [...] is a Nightmare to setup.*"
- "*a curated set of links*"
- "*make the home page more beginner friendly*"
- "*It's hard to understand translated articles, MDN should have articles that are mainly written in those languages by local experts*"

About other web sites

We asked for a list of third party resources in order to figure what is beginner-friendly and what's not. *Note: answers pointing to MDN were not taken into account.*



Controversial resources

It's worth noting that among all the resources provided by the responders, they have given two in particular where feedback was very controversial (both very positive and very negative feedback were provided).

W3Schools

W3Schools is in majority considered an unreliable resource for beginners, essentially because of its misleading content. It is considered outdated and promoting bad practices.

However, the way *W3Schools* is handling its structure and learning tools has quite some traction. The entire wording is clearly made for beginners with easy straight-to-the-point English, trying to avoid any jargon. The content structure is easy to understand and to follow right from the homepage to the whole sub-content. And last but not least, its live editable examples are definitely a top-notch feature for beginners.

StackOverflow

As a place to ask any question, *StackOverflow* is mostly considered a good resource for beginners who need efficient and immediately usable answers to their questions.

However, as the answers are essentially provided by average and expert developers, they lack context to understand the reasoning and requirements underneath. So it can mislead beginners and unintentionally promote bad practice.

Common blessings

Codecademy reached a clear consensus as the best resource to learn and was suggest by 30% of the responders. The various feedbacks regarding that resource and the others show the following points to be the most important things for a good learning web sites:

- Progressive, clearly structured courses
- Possibility to fiddle with live editable examples
- Interactive exercises

- Video contents
- Possibility to ask questions

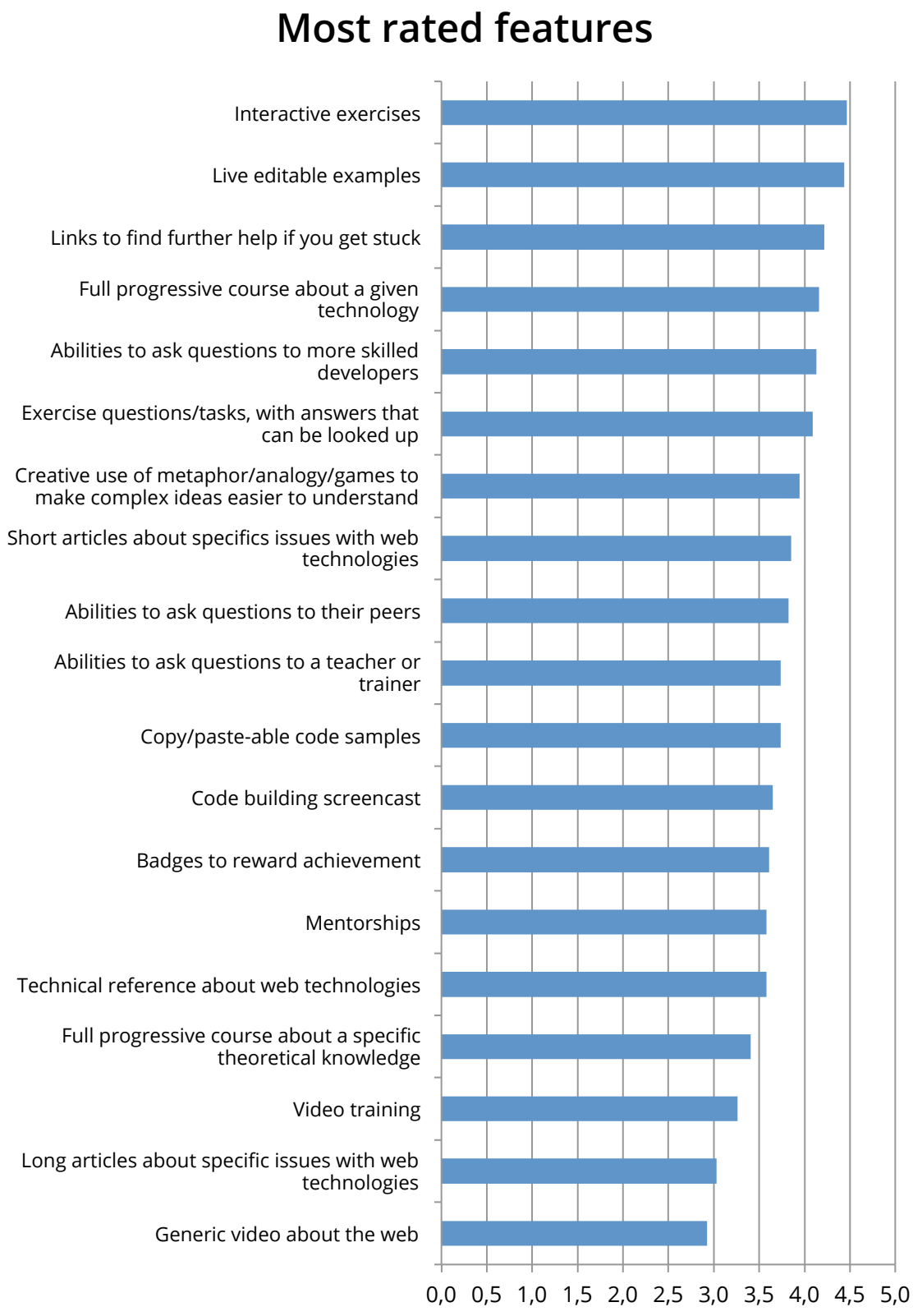
Common issues

On the other hand, there are some flaws that must be avoided at all cost. The points of attention raised in that survey was:

- Avoid outdated content
- Even if learning requires simplification, never promote bad practices
- Do not provide a solution without explaining why it’s a good solution
- Avoid unsorted, bloated content
- Avoid to be over-technical

Learning features

As we are considering some features to improve MDN, the responders were asked to rate them from 1 (not important for beginners) to 5 (absolutely needed for beginners).



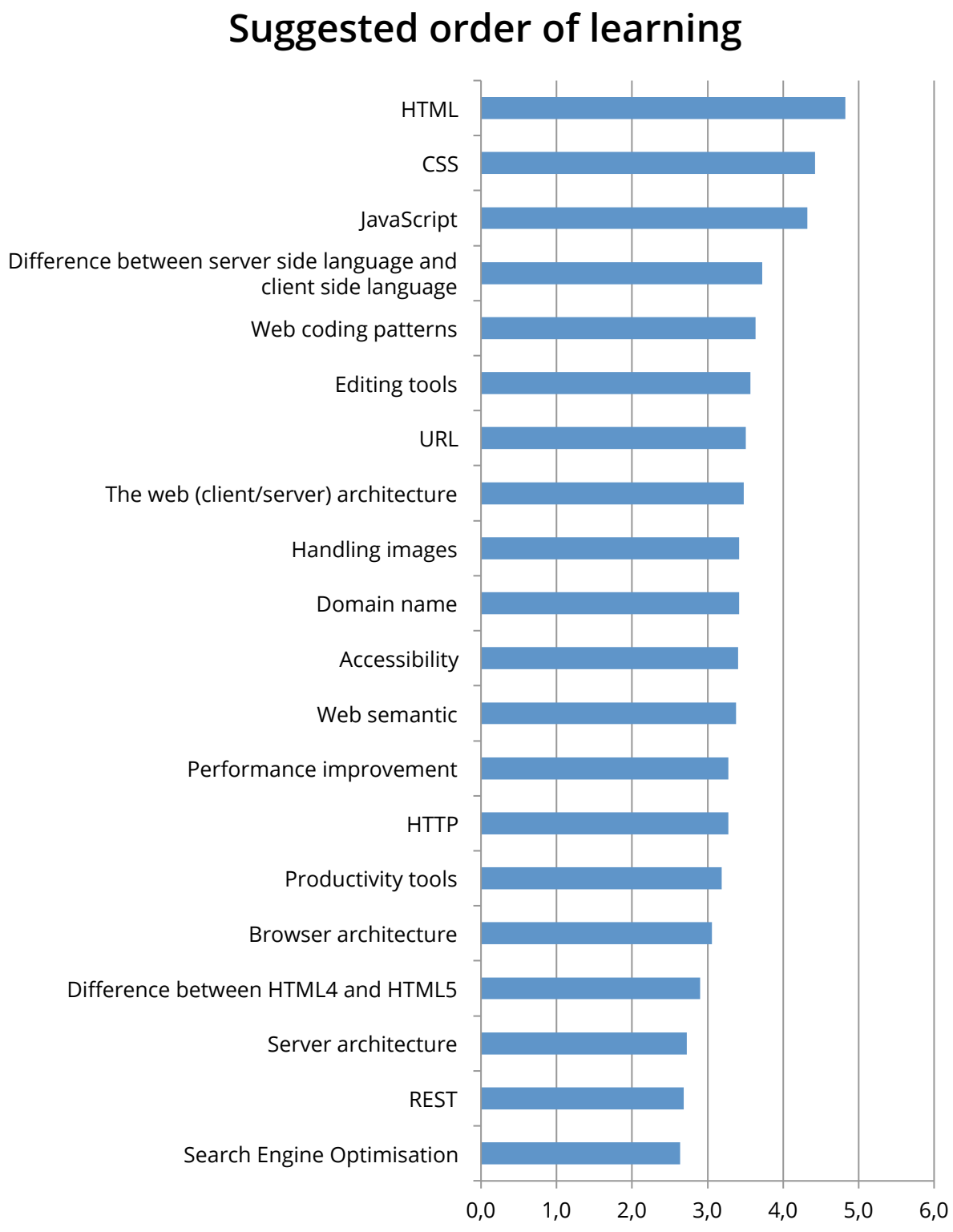
Other features to consider

When asked, the responders also suggested that we consider the following features:

- Gamification: Badges, Skills progression, etc.
- “How-To” and practical tutorials to solve real-world issues
- MCQs and testing questions with detailed and explained answers
- Content ranking/vote by users
- Organizing teaching/learning events

Knowledge

We also asked the responders to rate web-related knowledge from 1 (not important) to 5 (should be learn first).



Other knowledge to consider

When asked, the responders also suggested the following knowledge that was worth learning early (if not first):

- Network (TCP/IP)
- Third party “useful” library
- CSS Preprocessors
- Some server side languages (Ruby, Python, PHP, Java, etc.)
- Other standards (DOM, WebGL, etc.)
- How to work without an HTTP server
- Specificity of mobile development
- Responsive design
- How to push a web site online.

Points of difficulty

Responders were asked what were the hard parts they faced while learning the web technologies themselves. Those are points we must keep in mind while building our content for beginners:

- The amount of technologies and tools to learn
- Contradictory and outdated information
- The lack of straightforward process to build a web site
- Misleading practices
- Cross-browser issues
- The rate at which new technologies and concepts arise
- Overwhelming deep technical concepts
- English (for non-native English speakers)

About responders

In order to better understand the answers we asked a few questions about the responders.

To the question "*When did you start learning the web?*" we got two kinds of answers: a date or an age. The answers indicate that the population of responders was well balanced between seasoned web developers (since 1995) and real beginners (since 2013), and that the biggest part were quite young (below 25) when they started learning.

How did they learn the web?

They used various ways to learn:

- From books
- Through online resources
- By building a web site
- Mostly by themselves for the older ones,
- At schools or through online course for some of the younger
- Use of "view source" and from peers

But ultimately, their main way to learn was by "*getting their hands dirty*". It appears that learning the web has structured only very recently and it still needs to be improved.

What were their motivations to learn the web?

Note: *Even if we tried to spread the survey at large, it's worth considering that the answers to that question can be quite tainted by the way Mozilla is considering the web as many of the responders are Mozillans.*

- To communicate ideas
- To share information
- For professional reasons (Earning a living, running a business, etc.)
- To build something (web site, web app, game, etc.)
- Curiosity
- To have some fun
- Because it's the tool of the future
- To express themselves
- To help some friends
- To understand how it works