

Wikifunctions mental model

This research focused on Wikimedians who had demonstrated an interest in the Wikifunctions project by voting in the recent naming contest. 18 applicants were asked to self-assess their coding abilities on a six point scale. Four claimed no coding experience. Most were beginner or intermediate-level. Only three rated themselves as advanced or professional.

It's helpful to think about the potential research cohort as a 2x2. Some people care about Wikimedia while others care about code. A few care about both. This study focused on those who were not technically inclined but were invested in Wikimedia as an ideal.

Cares about code	Programmers outside the Wikimedia ecosystem	Programmers active on Wikimedia projects
	General non-technical internet users	Wikimedia users who are non-technical
		Cares about Wikimedia

In retrospect, seven of the 18 research participants were more technical than the rest. These seven are indicated by the underlined name accompanying their quotes. They offer contrasting perspectives throughout the study.

Research findings

- Target audience
- Confusion about the UI
- Goals for the project
- Alternate contributions for non-programmers
- Templates as entry level code
- Form based programming
- Documentation
- Naming conventions
- Coding in English
- Mobile coding editors
- Next steps

Target audience

The participants reviewed screenshots of the [Wikifunctions main page](#), the [create a new object](#) flow and examples such as [conjunction](#) and [succession](#).

Regardless of their individual coding abilities, participants overwhelmingly felt that the function builder interface was meant for programmers. Those who were not technically inclined recognized that it was not for them while the more experienced participants correctly predicted confusion for everyone else.

Programming logic can be an obstacle

One beginning-level coder theorized that programming logic was the key factor that separated programmers from non-programmers.

Others have articulated this idea in more detail. Are there habits of mind that are required for programming which are not found in the general population? If so, this could be an obstacle for non-technical users, regardless of the Wikifunctions interface.

Maciej: The difficult thing for beginners is understanding the logic process. If this happens, then do that. I think that people may have the same problem here at Wikifunctions because of this lack of understanding. If something happens, then the program should do this.

Maciej: It's the same thing in Excel. Most people don't like Microsoft Excel because they cannot follow the logic principle.

This project is for programmers

Many non-technical users were open about their lack of understanding, explaining that they were not a "techie" or not a technical person. A few referred to friends or acquaintances who were more technical by way of comparison.

Embracing the identity of a non-technical person makes technology easy to evaluate. Anything too complex can be categorically dismissed.

Marc: This speaks to the coder more than the user. The coder should be able to read all the different parts and understand how they connect. And then, it's already assumed that the coder will code. To a certain extent, we cannot hide what we are doing.

Nicolas: Well, I guess it depends on who you're catering to. I think a person who works with programming would probably make sense out of this. It would probably make sense to them, but to me it doesn't. Maybe if they made it very, very simple to contribute. But I have the feeling it's going to be mostly programmers working on this.

Frederick: I have an open mind and I think many of the youngsters are much more clued in than I would be and I think they would see a lot of potential in these kind of things. Maybe we should rope in some technical guys to feedback on things like this, because they are the guys who would be more clued in than us.

Mithun: From the interface, I can't say what type of script I'm building. But if a technical user would try, I guess they can know better than me. I'm not much in the technical side.

Non-technical users wouldn't understand

Those who didn't know how to code believed that Wikifunctions would be impossible for other non-programmers to understand. None of the non-technical participants from this study displayed any real understanding of the interface.

Deror: I understand what they are trying to do here but for a non-programmer it would be impossible to understand. Most people who edit Wikipedia would not understand anything. Would run away. And if they're not an English speaker it's going to be even faster.

Krishna: Maybe that is a problem for people who are not full-time developers, or expert developers, but only beginner to intermediate, with some little experience.

Nicolas: What do they mean by functions? Because it could at least have two different meanings, right? It can be the general meaning of the word function, or the programming sense, which I guess is what they're using here, but that's not super clear just from looking at the intro. I don't get a clear sense of what the project is for.

Nicolas: I don't understand what this is doing, essentially.

Marc: I think that for a non-programmer or someone non-technically savvy, this is not helping. I see that it's difficult to provide tools to non-programmers without a technical minimal qualification. I don't know how to say it.

Confusion about the UI

Many of the participants were initially confused about the Wikifunctions interface but the main distinction between programmers and non-programmers was that the programmers actually tried to figure it out. They believed that the interface could be understood with enough time and perseverance and they eventually articulated a mental model.

Those who didn't know how to write code or weren't technically inclined made more of a token effort. A few had polite feedback like "that's interesting" or "looks good" while others failed to demonstrate much understanding.

Everyone will get it eventually

One advanced programmer made the argument that coding was becoming more ubiquitous and would eventually become common knowledge. A few of the participants had learned to code in high school or even earlier but this will likely remain a niche skill similar to the population that learns advanced mathematics (which has been around for much longer).

Marc: There will be a point. I don't know if this is going to be in 10 years or in 15 or maybe in seven years, but a certain degree of knowledge of what programming is just to understand what the function is, I think that it's going to become common knowledge.

A few Wikidata users invoked the success of that project as a defense for incomprehensible interfaces. This is probably not an argument for Wikifunctions. Those with the ability and the inclination to decipher complex systems are outliers.

Katherine: I've watched other people who are brand new, like day one to Wikidata, try to edit, and they tell me that the interface is mystifying. I feel like all of this stuff is something you have to just learn and get used to. And I'm not saying it's not worth it to do a good design. Obviously it's really nice to have these designed as well as possible.

Frederick: I'm sure it'll take a bit of time. It may take a bit of time, but people will get the hang of it soon enough, because even we saw it with Wikidata initially, like no one had an open mind on it, but once they opened up even in India, I mean, there were guys who actually gave up Wikipedia and just went to Wikidata because they said, this is the future.

Programmer feedback

The technical participants commented on the usability of the interface. They understood the subject matter and tried to make it fit what they were seeing.

The static screenshots were not interactive and demonstrated only limited functionality. Participants would probably have overcome their initial confusion more quickly with access to a functional prototype for exploration by trial and error.

See also: **form based programming**.

Christian: So when I look at the first screen, it's unclear to me whether I'm supposed to write something in there, or whether I can just click on one of the buttons, links to fill in the thing. It gives me some impression that I might click on tests and then it fills in tests, and then I get to the next page. I'm not really sure.

Katherine: If I'm at "create a new object" and I see type there, I would have no idea what to type, but obviously that would be my first time ever looking at it. Once I get used to it, I would probably have a sense of what to type there. I do not yet know what all of the links do unless... I can't click on them, so I don't know what those do.

Katherine: Function call and result condition definitely clarify what I should be typing there. I don't know how the interface knew to present left-hand side and right-hand side.

Maciej: I can kind of understand what is happening, what to click next, what to expect. I can read what I can read while I try to figure out what this function does. So, it's a function. No, I cannot really tell you what it does. Oh, no, now I can see, that it takes a list of the words and it can add an "and" between the last two.

Non-programmer feedback

The non-technical participants who evaluated the UI had a hard time grasping the mental model behind building functions. Some non-technical participants didn't even try to interpret the UI beyond the main landing page.

Frederick: What's a function? Excuse my ignorance.

Krishna: Conjunctions, arguments, implementation, I'm not able to understand the arguments sections. What is the argument section trying to do? The whole section? I do understand the individual terms and list English known phrase up for commons, all those terms. But what, as a whole, what is it trying to do?

Nicolas: I guess it's creating a function, but yeah, that's about it.

Deror: I just don't understand this page. Okay. Type. Function call is old condition. Okay, that's already programming. I don't know. Let me try and figure out the code for conjunction. No. I don't understand the code. I think you know what they're trying to do, but...

Mohamed: But this function of the day, you see the place. Yes. It's better it stays on front of the date. I see. But Earth, Neowise and time. So what was that saying? What does this section explain. Test? Okay. It's fine.

Erkin: I think it's difficult because I don't know. I didn't understand. It's very strange.

Syntax is not clear

A few participants questioned the language for building new functions. Programmers looked for a declaration of syntax while those with less experience assumed that this was simply a programming language they didn't understand.

Christian: Yes. And also I need to say which language I'm writing the thing is, and then it does not appear to me what result condition happens.

Katherine: I'm a little wary of the magic input box because I would want to know what syntax is being used and how I can learn the rules of that syntax. And so a magic box sounds like it's going to be easier, but if you don't know what to type in there, just like if a person doesn't know Wikitext, they're not going to know what to type in for an article. So I'm a little wary of this one but maybe it will be helpful.

Nahum: It's a new syntax you need to learn. But it isn't easy to learn.

Deror: I'm guessing it's C++, the code behind it. You need to learn to understand it.

Mithun: Actually, the language used here is not recognized by me.

Goals for the project

When asked to articulate the purpose of Wikifunctions in their own words, most participants focused on how the project could benefit other sister projects or the Wikimedia ecosystem more broadly. None of the participants voiced any support for third party use of the functions. Future research should validate this finding with external participants.

Some users focused on how Wikifunctions will support Abstract Wikipedia. Wikifunctions has been overshadowed by Abstract Wikipedia since its inception. It will be important for research to decouple these projects to the extent Wikifunctions is meant to stand alone.

Formalizing the bot environment

Several of the more technical users wanted to address the sprawl of bots across Wikipedia. They imagined that an officially sanctioned project for programmatically generating articles would be better than the current landscape of random bots with no oversight producing stub articles of unknown quality. This oversight argument also applies to Wikidata bots.

One of the less technical participants saw the potential for a bot service that would allow users to create and run bots by selecting from options without necessarily knowing how to write code.

Samuel: The projects are a little bit shy of complete automation that outputs anything. But one of the nice things about Wikifunctions is just getting over that anxiety. The anxiety of having anything be fully automated.

Marc: The first goal of Wikifunctions is to substitute the bots or somehow provide a better space in a structure or a framework for it to reduce the different pieces of code. People are already coding bots. There are a couple of crazy guys creating a million articles for Cebuano and this is already happening.

Houce: It can be important, mainly if we will get rid of many, many bots that are used now on Wikidata and Wikipedia, in general. Mainly algorithms are using now to do automated stuff, and if the functions are imported in Wikifunctions, we can just make one, but we do all the stuff instead of coding many, many bots.

Mithun: If you create a script and know about a lot of this, how do we make calls to an API and change the text of the Wikipedia piece, which we can edit with this script, so if those process can be simplified. I enter all of those requisites and build a code so that I can use.

Documenting historically important functions

Wikipedia has many articles on computer programming and functions. These articles offer detailed descriptions but cannot demonstrate usage other than with conceptual diagrams. For example: articles on [sorting algorithms](#).

Several of the more technical participants saw Wikifunctions as a way to precisely document the behavior of functions for these articles.

Katherine: There's a set of functions that are already widely in use that would be super useful, and I anticipate those would go in there first, because many people who are likely to contribute to this will want something to start from. It wouldn't really be writing a new function. It's just structuring for functions that have been in use for a long time.

Sandra: It will be good for the Arabic Wikipedia, because when you are writing, like when you are working on Python, we do recently we have, we have set quite an article as it's one of the good articles. Then we can mention for the readers you can start working on part on how you can learn on site. And from here, we can function, it will help students to know better.

Samuel: We need a place to be able to unambiguously reference and describe and annotate functions. It's one of those things that we don't have a cross-wiki solution for. There hasn't really been a place for people to capture those things.

Samuel: Even at the encyclopedic level for historically important ones, it's frustrating because you often want to reference a particular instance of a gradually changing function.

Houce: Some items on Wikidata are about formulas. For example, the body mass index in Wikidata is a formula itself, but it has a Wikidata item. The point is that we can represent that using Wikifunctions.

Houce: I'm talking not only about formulas, but also about classifications. For example, for cancer, there is something called DNN, which classify the cancer according to several features from anatomic location and size. This will be complex stuff and if you can represent it on Wikifunctions this will be good, because we cannot represent it on Wikidata.

Wikipedia global scripts

Less technical participants saw Wikifunctions through the lens of their past experience with user scripts. They didn't write scripts themselves but they did have experience searching for scripts and using them.

They recognized the problems with scripts scattered across language wikis and the advantage a global hub like Wikifunctions might provide.

Luis: Maybe this finally having somewhere to store scripts and work with them in a sense that it's not so easy right now. It give me this feeling from my community. Like I talk to that guy and the guy sends an email to that other guy because he knows them from that day. It's from the basics. We need to go further and have a library or something like that that we can use many things from other people from other languages.

Nicolas: I think that would make sense if the user scripts were global, in a sense, because a lot of the user scripts are for specific things on specific sites, right? But a lot of user scripts, they are meant to be used across Wiki. So yeah, in that case, I think it would be very useful.

Mithun: Actually, there are many great scripts on other Wikipedias that English Wiki people also don't know about. I think it'd be really great if you moved those into a central place where people from other wikis can also find out.

Wikipedia global templates

Templates were another common touchpoint for less technical participants. Global templates have been a highly requested feature across Wikipedia for years.

See also: **templates as entry level code**.

Samuel: I have the sense people are avoiding talking about templates as an example for Wikifunctions because they don't want to just open a particular can of worms.

Nicolas: I know that a lot of people are talking about global templates. That was actually one of the highest voted features on this year's wishlist. Every year they have one, and I think this year the most voted one was related to that.

Nicolas: A lot of modules and templates, they are virtually identical in a lot of Wikipedias, but maybe there is a change or maybe there's an update. Someone changes the code a little bit on the English Wikipedia, for example, and that doesn't necessarily cascade down the other Wikis. So, you might have some Wikis with outdated templates, and these things, they rely on each other a lot. There are dependencies, right?

Mithun: I think templates and user script both as a goal, so in my opinion both should be at one place so all the community can use them. Every Wikipedia requires them, so in the end they have to import them. If we had all of those at one place, we can easily use them and improve the quality of the article.

Krishna: I think a template would be helpful to put in Wikifunctions, definitely. Because templates, on English Wikipedia at least, I think on Commons, the base templates have been created based on Lua modules.

Wikidata query service

SPARQL queries were an obvious use case for Wikidata users. One participant was intrigued by the idea of a query service that could be developed within Wikilambda. Others saw the potential to share queries that might have broad appeal across the community.

Luis: It could be really fancy and interesting to have a Wiki of this type of functions, but just that it's a thought that came to my mind when I saw about Wikifunctions. That is a query service of Wikidata. A library of Wikidata queries.

Katherine: Some SPARQL queries are highly reusable for other people. One thing particularly that I think would be awesome would be queries that describe a subset of Wikidata that are likely to be really important to people. So something like all human genes, or all human genes, human proteins, and disease information.

Samuel: A bunch of the functions are probably going to just look like a SPARQL query. So anyone who's played around with the Wikidata Query Service.

Stand-alone benefits

Two of the more technical participants felt that Wikifunctions was valuable in its own right because it could enable new possibilities that are not yet obvious.

Houce: Wikifunctions can be a good Wiki by itself. This means that you can have people that contribute to Wikifunctions just because they think Wikifunction can be a good and can be queried using a language that is similar to SPARQL and used for the generating many, many stuff with combination with Wikidata.

Katherine: But then I think that the set of all functions being available to everyone, and labels and descriptions for them across as many languages as the system can support is a super cool thing to work on, and will be really valuable to me personally, and will enable a lot of people internationally to do even more cool creative projects.

Alternate contributions for non-programmers

Many other open source projects have explored how to integrate non-technical users into the development process. Typical approaches include:

- Bug testing and reports
- Knowledge base (docs, tutorials, Q&As, etc.)
- UX, artwork, design
- Translation and localization
- Evangelization

Participants from this study voiced a few of these ideas when asked to imagine how non-technical users might contribute. Most suggestions were centered around organization and localization, two activities which already happen with Wikipedia user scripts.

When it comes to writing functions, no one could articulate a use case for a non-programmer beyond basic editing. Some were optimistic about redefining Wikifunctions to include less complex types of code that would appeal to a more general audience.

Naming, descriptions, categorization and search

Non-technical users already engage in the organization of content across Wikipedia and many other sister projects. Wikimedia Commons is analogous to Wikifunctions in that the content is highly dependent on metadata in order to be searchable.

Katherine: Well, I guess all of the functions will need to be accessed in some way. And so naming functions in a way that is meaningful, but not too verbose, and makes sense for different communities, where certain programming languages call certain functions something specific where other programming languages might refer to it differently. I think everybody who contributes to Wikifunctions is going to need to work to find meaningful names.

Katherine: People are going to need to search, and they're going to need to type something in, in order to search. And so I think finding meaningful access points for the different functions will be one area where you don't need programming experience to contribute.

Samuel: Inline documentation, describing, copying, and modifying the function descriptions. What it's for, how it will be used, and anything in that sequence.

Translation and localization support

Users without technical knowledge can contribute to code by translating any text strings into their language. This already happens with user scripts imported from one wiki to another.

Nicolas: I use scripts that other people have written. My contributions are mostly translation once in a while. I'll translate something into the languages I know. Translating strings in the scripts, but not contributing with code, just localization.

Christian: When it comes to a new user who is not a programmer, it's important to have a task that's small enough for the user to be able to start with. And I imagine that adding translations, if the function is language dependent, can be in many cases the first task that the user without a programming background could do, and set could familiarize the user with the thing.

Mithun: If people from a different language wanted to contribute to the project and the documentation is entirely in English, that might be a problem because we are building a project that has script from all over Wikipedia. It makes more sense to make it multilingual than being in English. Definitely I can participate in translating the interface.

Editing existing functions

None of the non-programmers could imagine editing actual code but two of the more technical users thought this might be a valid entry point. In general, editing templates is seen as a much lower bar for non-programmers.

Samuel: Copying and modifying is often the best way to contribute. And once you do that, once you know how to write code, but making it easy to clone something and play around with it. You don't have to know very much as long as you can visualize whether what you're making still does something.

Christian: I think especially as the first action that a non-programmer takes, making a small change in function, it's more likely to be something that a non-programmer can do, and then they can kind of learn how the things work.

Importing code

If Wikifunctions were to adopt templates or user scripts then non-technical users could help to find and import those elements.

Nicolas: Oh yeah, for sure. I would gladly do that. Actually finding and relocating the templates into Wikifunctions.

Krishna: I'm not really aware of how this is going to function, but I do publish my codes on GitHub, so something that we use for that, I'll definitely be happy to have people put them into Wikifunctions as well.

Templates as entry-level code

Templates vary greatly in complexity but for many users they offer a basic introduction to code. A project that incorporated the existing template ecosystem could attract those who do not think of themselves as programmers.

Templates should qualify as functions

Two participants who advocated for templates in Wikifunctions argued that they qualify as functions due to their internal logic and use of inputs and outputs.

Samuel: Simple templates are just switch statements. It could just be, do you or do you not include each of these rows in the table on the info box? But then they started adding parameters that would change which style you applied. And now that's looking more like a piece of code.

Samuel: When the templates were switched over to Lua, it was specifically so you could do more meaningful variable referencing right inside your templates.

Samuel: Everyone has a template that they care about. Most people who write templates would say that they don't know how, or they're not good at it because they can be very complicated, but they still write simple ones.

Samuel: I think most of the functions for the first couple of years, you're going to be very low complexity. They're template functions probably, with a much easier to parse interface. A major question here is to what extent templates can all be moved into Wikifunctions.

Krishna: Most of the people are accustomed to templates on Wikimedia projects. For example, I can take an example of an Infobox on English Wikipedia. You insert name and it appears in a specific way on the Infobox and you insert a place of date of birth, fields like that.

Krishna: We call them *template fields*. There is a common word we use. So maybe there's a function fields, like, what are the input to the function.

Need for cross-wiki templates

Adopting templates as an aspect of Wikifunctions would also address a longstanding request from the community for global templates.

See also: **Wikipedia global templates**.

Samuel: We desperately need cross-wiki templates, and people have avoided it for various reasons because they've felt somehow it was an extra vandalism risk, or it wasn't clear enough to people across languages what a template was doing. But that is one of the hardest things when you're sitting up a new wiki. It breaks a lot of things. It takes a lot of time.

Mithun: I can write a few basic templates, but if there are some complicated code in it, then I can request other people. Basically, when I needed templates mostly was in the small Wikis. At that moment, I can ask them local users to import it from English sites that they are already available so they can do that job.

Template inheritance

Templates sometimes have multiple levels of inheritance traced back to an original template written by a Lua expert.

Krishna: I think a template would be helpful to put in Wikifunctions, definitely. Because templates, on English Wikipedia at least, I think on Commons, maybe what happened is the base templates have been created based on Lua modules. So the template is based on another template, another template is based on another template. So when you trace back to the origin, it is based on a role model which has been written by some expert in Lua.

Form based programming

Many participants from this study recognized the theoretical benefits of an abstracted coding environment for non-technical users. A few mentioned Scratch as an example. But in practice the prototype design for the function builder interface was incomprehensible to the non-technical users and limiting for the programmers.

Expecting to import functions

Two of the more advanced participants framed the function builder interface as a way to import functions. In their mind it was not an expressive tool for generating new code.

Katherine: No. I don't think this is a programming environment. I see it as a form-based entry of programming snippets or something like that. I mean, if you told me this was form-supported code writing, I guess I could believe you, but that's not how I had been thinking of it.

Richard: Of course it isn't the usual way of doing it. The usual way is to type in text and some of the examples are with just concatenation of functions with texts. So I would expect that also to be possible, then at least you can prepare things locally and you don't have to have this interface and do everything interactively.

Expecting to type code

Other participants were confused by the word "type" and interpreted it as an affordance for literally typing code with the keyboard rather than selecting an object type. In general they expected to input multiple lines of code.

Krishna: I'm still a little confused about where I should put the code here. I'm not able to see a field where I can... Where is the code to be inserted. So it says type...

Christian: It's not clear to me what function call and result condition are here. So if I'm kind of supposed to write normal programming code and language in the function call, then I'm not sure why this is only one line.

Richard: And if you need to do a lot of these things, then it might be easier to prepare it using a text editor than it would be to go through doing everything interactively. Yes. So I think you would want another way of doing it.

Non-programmer feedback

Some participants recognized similarities between the function builder interface and other block-building systems they had encountered. This recognition didn't help the less technical users to actually understand the new interface.

Deror: When I was in the Navy, they developed something called the rest machine, which you program like this. You just choose. So, I choose the name for the function and then I choose from another function and it opens. You need to put in this parameter, this parameter and this parameter, and this is the output. It was like programming with buttons.

Deror: If you have to type a function, then you need programming. If you choose from a combo box, maybe somebody who doesn't program can do it. This limits your options, but it can increase the usability.

Luis: Never seen something like this. It could be very easy to use. Block type coding is nice. It recalls me of something that gets used to start with code like Scratch. Programming with blocks is something that people that are not experienced with programming find useful.

Krishna: And also all these options seem to be some sort of a dropdown, which I would prefer what function to be called. I think it is already defined, multiply. And again, I'm not sure of what this input fields as of now. And the result condition.

Programmer feedback

The highly technical participants were able to articulate a mental model for the function builder interface. Their learning process would have benefited from a functional prototype to test their hypotheses through trial and error.

Katherine: I would probably just browse all of those lists so I could get a better sense of how this whole thing is structured. And for result condition, I would want to know what the full inventory of result conditions are. So I would probably explore that to try to understand more of what is possible to express.

Christian: Function call means that I'm calling wiki functions here. So this is why I don't have to write a programming language beforehand. This suggests that the function can be made of WikiLambda functions. This would be a composer for built-in functions, probably, in the previous syntax.

Richard: Yes. This is what I was expecting. So yes, I go through the process. I can make myself objects. So a function called and then has its arguments. I suppose that I'd click on the symbol at the right and that offers me either that I put in a number or it can be a new function which carries on. So I suppose I can see what types of objects I can add there.

Documentation

Some of the confusion encountered by participants was due to a lack of documentation. The more technical users would have been satisfied with the ability to explore the interface and learn by trial and error. Less experienced users wanted inline help and clear instructions.

Inputs and outputs

Participants reasoned that knowing the inputs and outputs for a function was the minimum level of information necessary to understand its usage.

Krishna: I'm also a little bit confused, what is the input and what is the output that we're trying to achieve? So that is also not clear.

Krishna: When I write a function on... Again, I'm not a professional programmer. I don't know how things are done. So for example, if I don't know how to function on Python script, I'd just say what parameters this function should have? It should have a... Basically what the functions should have as an input to execute the function.

Marc: Sometimes, as a programmer even you don't know what you're doing, but you know that the inputs and the outputs, how they look like. And then, you just try luck and see if it works or it doesn't work. So making very clear the inputs and the outputs help non-programmers.

Houce: The Wikifunction is a project about functions. This means that you have an input, and you will try to represent the way that leads to the outputs.

Christian: So I see that when it comes to the type, it's not immediately clear. The output type or the entry type? So this case, probably output type is the most likely thing, but this was kind of what I was thinking. And maybe you could also sort it by input time somewhere.

Descriptions as documentation

Meaningful function names and descriptions are examples of user-generated documentation. They also represent a way for non-programmers to contribute to the project by writing and translating these descriptions.

See also: **naming functions**.

See also: **alternate contributions for non-programmers**.

Katherine: Okay. And then when I'm looking at the one that finally has the ZID, I'm wondering if where it says no text, if that is the description. And I'm wondering then, why wasn't I presented a spot to put a description? And if I was, I totally missed that. And we certainly want the description to be prominent because we want people to add descriptions. The kind of minimal version of what is stored under function call and result condition, this makes sense to me.

Katherine: So now there is a description, but I didn't see any options for where this description was input or anything. Descriptions should be part of that creation process and not something that takes place afterwards.

Error messages

Meaningful error messages are a type of documentation that can help users to learn from mistakes by experimentation.

Sandra: Yeah, multiply. Because I find it pretty easy because I know about integer, but I'm thinking about if somebody who don't know what is integer on that, there's a float and like that. The user should have the right to like an error code. If you put this, it will have an error.

Krishna: Wikidata properties are property constrained to the field, which guides the property, how it should behave. For example, if it is an ER property and you have a property constraint and if someone tries to insert a text, it rejects the input. So when for example, like ISBN property only allows the user to input a form of ISBN number.

Inline help

A few participants specifically asked for inline help such as tooltips or other explanations regarding the interface options.

Eduardo: Like all these functions here, like the key function, the call, the multiplier, there should be a way to explain each of these functions and what they're for. It's very similar to code, the way this would be written. I think there's some documentation missing for each of the parameters of each code. It could give some sort of examples of the function.

Deror: How should I put it? Maybe you need some help, because I don't know what's in... Oh, it's links. No, it's not links. The function in parentheses, is it linked to some help, some information page? Or some kind of, like a question mark, an explanation. What is the implementation link? Or why is it in parentheses?

Sandra: When you are working on Wiki function and you are not a programmer, it's safe to read the menu or the basics of it, of this switch, encyclopedia to write better, we can add help for defining the type and the function and the class and so on. When you are not a programmer. You would ask these questions.

Naming functions

Meaningful function names act as a minimum bar for documentation. Some participants recognized the potential for naming conflicts and understood the importance of naming for search support.

Function naming conventions

The more technical participants discussed whether it was actually important to have function names. For example: many of the saved queries at quarry.wmflabs.org are untitled.

Marc: Maybe it's unavoidable to come up with a name. All the libraries have names for the functions. Most of them, when they do the same thing, they have the same name even though they are libraries from different languages. It seems unavoidable to come up with names.

Katherine: Well, I guess all of the functions will need to be accessed in some way. And so naming functions in a way that is meaningful, but not too verbose, and makes sense for different communities, where certain programming languages call certain functions something specific where other programming languages might refer to it differently.

Christian: If the headline tells me what the subject is, then I don't have to look up in the documentation what a Lambda function's called, something like this.

Maciej: There is an equation at the top. So, I wouldn't be able to describe to others what I am seeing, but the process of creating it looks kind of easy.

Duplicate functions

Some participants recognized the potential for duplicate function names or identical functions with different names. This was seen as something that should be addressed by search.

Deror: When you start a new item, you give it a name? Then other people can start the same program by giving it another name, without knowing that somebody is already working on it. There needs to be a way to find if there is overlapping and combine functions.

Houce: So, you can get stuck with two functions having the same name, and this the main issue here, but as a form it is good as it is now.

Marc: It would be good to be able to find functions that they are doing the same, and it would be good to see the number of times a function is used by other functions.

Function versions and inheritance

More advanced users raised the issue of versioning and inheritance. They felt that Wikifunctions should support this type of dependency.

Samuel: I think it's going to be very important for people to be able to maintain a number of different active versions of a function. There'll just be one most useful version that gets better over time. You may well have a function like how to describe baseball, or how to summarize a soccer game. Soccer feels like it might be played in many more cultures, and there'll just be variations in how people talk about it, or what details they include.

Houce: What I see is that you are making a function based on another function. You can do elastic search here, this means that the function does not have a name only, but a name with a description. That would be good because in medicine, there are many DNNs. There are many DNN classification for each cancer. So, if you write the DNN, you can, for example, have the DNN for breast cancer, but not for, for example, bladder cancer.

Coding in English

Wikifunctions may be different from other Wikimedia projects due to the priority of the English language in programming. This doesn't mean the interface shouldn't be localized but it might not be straightforward since some programming terms are expected to be in English.

English is the language of programming

Most keywords of a given programming language are written in English but comments, variables and user-defined classes and methods can be written in the programmer's own language.

Sandra: I speak Arabic but I use English in programming. I use functions and that's why, I don't know... But there is a word for function in Arabic, but I don't know if all use it or they all use English instead. So it should be an option too, to put it in your language, but it's not mandatory because I think a lot of programmers use English. I am sure that English is the main language for them, for working on a program or programming.

Mohamed: PHP is in English.

Eduardo: The codes are in English for Java and C++.

Translating the interface

The interface screenshots used for this study were in English. This was an obstacle for some participants with a lower level of fluency and will be addressed for future research.

Deror: Keep in mind that even English speakers sometimes, English is difficult so there is the translation tools usually, somewhere in the background, so people can translate all the messages and the main page to make it easier to use. This page is simple, but it should display in other languages besides English.

Sandra: To help users to work bit better and find it easy for them if we can write the page in our own language, because we are connected with Wikidata. But I first translated it to Arabic. Maybe the user can put the work in his own language and help to work better.

Christian: Having the key data items as input from Wikidata would be useful there to translate names of those objects over multiple languages.

Mobile coding editors

Participants who mentioned programming languages during their interview were asked about their experience with mobile coding platforms. None had any positive examples to share. A few expressed cautious optimism for future advancements but most agreed that coding on a mobile phone was not currently possible.

Cultural trends toward mobile

One of the non-technical participants was involved with a local user group and mentioned the trend toward mobile adoption. Other Wikimedia studies have confirmed this trend.

Frederick: In India mobile is huge. Many people are using their mobiles to actually do their main communication. So you find, if you go to the recent changes on the Konkani Wikipedia, you'll find people editing on mobile. And I keep wondering how are they managing to do it in that sense? I mean, you have script issues and all kinds of things there but so it's going to be tougher, I guess.

Frederick: They've been warning us that people in India are just going to shift away from the desktop to the mobile and the bulk of the users here are going to actually shift or even just completely leapfrog this full desktop technology and go straight to mobile.

Optimism for mobile interfaces

Two of the more technical participants hoped that future advances in technology might support mobile coding in the same way that predictive text and auto complete had helped with mobile messaging.

Sandra: Learning and working on mobile phones nowadays it's improving and people with auto complete it will help students and ensure to help people to reach the language and to find it near to their intentions. So yeah, I think it would be a good idea. It will be hard to write on mobile phone, but with auto-complete and having... helping out, it will be good for people who use it without having the laptops or desktop computers.

Katherine: People are super smart and somebody might come up with an awesome interface and a really awesome software keyboard so that this is less painful in the future.

Linear vs non-linear coding styles

The more advanced programmers also explained that writing code was different than writing messages because of the need to move back and forth between many different parts of the document and between supporting documents.

The closest Wikipedia analogy might be to the section translation tool for mobile.

Marc: Wow. I have no idea how other people do it. Actually, I would really miss a real keyboard because typing with a phone even though is easier because the sensitivity of the current phones is quite good compared to five years ago, it just feels difficult to type. I don't know if the jumps between the lines and the tabs, I don't know how that could work. I think it's an ambitious goal. It's not a goal to forget about, but I'm really curious if this is possible.

Katherine: The SPARQL query editor interface, it's hard to see stuff there and on a mobile device. And so if I was trying to do that on a phone, I would need options for seeing query fragments. I mean, some of the SPARQL queries that people write can be longer than a SPARQL query service screen on a desktop computer, so it would be very difficult to move through that query on a mobile device.

Interface bottlenecks

The more technical participants articulated their concerns about using a mobile interface to write code. The screen can display less information, the keyboard covers much of the screen and it also requires switching between different inputs for numbers, letters and symbols.

Many programmers use two or more monitors to display code and some have specialized keyboards. This is simply faster for writing code and better for understanding complex systems.

Katherine: So I think the challenge is it can already be fiddly and annoying to write a SPARQL query, because I mean, I can write SPARQL queries in my head much faster than I can type them. And so that's already annoying on a desktop computer. I would be way too frustrated trying to do that on mobile currently.

Maciej: The main problem is that you have a smaller screen and a smaller keyboard, and you will write much slower than on the computer, but it is possible.

Christian: I don't think I have actually written code on a mobile phone before. Having a bigger monitor is something that's quite useful if you want to understand a lot of code.

Mobile for quick tasks

Mobile interfaces are commonly used for simple tasks. Examples for Wikifunctions might include organization, documentation or string translation. The mobile UI might not need to support writing an entire function from scratch.

Houce: But the mobile editor itself is good as an idea because it is a quick, you can fix things in a nutshell, but the technical part is mostly difficult.

Maciej: If it is a short answer, it's quicker to take your phone and write something, but if you have to work on something long and a few pages long, it will be easier to open your computer and write on there.

Next steps

Future research will use the same 2x2 matrix as a guide but focus on a different quadrant. What can we learn from those who care more about code than they do about Wikimedia? Half the participants would ideally be recruited from outside the Wikimedia ecosystem and all should identify as programmers.