

Task category / Task	Estimated time in 1-man-weeks	Comment
----------------------	----------------------------------	---------

Instructions

The goal of this stage is to estimate how much time the development of individual tasks would cost us. We will measure in how many weeks a single developer would have to spend.

The goal is **NOT** to vote how much time you think we should *limit* spending on a task to!
This stage is only about *total* developer time cost estimation as if we had infinite money.

The cells for the week estimate should provide you with a dropdown, please only use the values there.
For tasks which you estimate to be below or above the min/max choices in the dropdown, just choose the min/max ones.
You do not need to provide an estimate for tasks whose codebase you don't feel familiar enough with.

Non-developers were allowed to add tasks to the list so some tasks may be vague, unclear, or under-specified ("I want a pony!"). To be fair we cannot just remove them - but you may consider increasing your cost estimate accordingly to account for this uncertainty.

For tasks which depend on other tasks, do not add the cost of the dependency if the dependency has a separate entry in the spreadsheet as well. Estimate the cost of the dependency separately then.

Speed - Improve performance of Freenet or Freenet applications

Web of Trust: Finish first iteration of most critical speed fixes (1 bugtracker entry). Was subject of previous 2 years of paid work. Ensures this work is not left unfinished. Needed for Sone / Freetalk / filesharing / ... (see all other sections except Technical Debt)	8	Been volunteering on this for a month (not full time). Depending on whether the planning period extends my volunteering this may progress further.
Freetalk: Make Freetalk usable again by using the new WoT event-notifications API to fix its most severe performance issues. The API had been implemented as previous paid work to specifically allow repairing Freetalk. Thus we'd reap what we sowed there.	16	Depends on making Freetalk itself event-driven instead of polling-based which is a bit of work. Adding 4 weeks of headroom to account for the fact that the codebase may have rusted due to not being in use for a long time.
WoT: Second iteration of less critical performance fixes: Reduce transaction size from O(512) to O(1) : Split up "import trustlist" transaction into 1 transaction for each trust.	12	Depends on amending event-notifications to support operating by callbacks instead of by FCP - which is likely a month of work.

Task category / Task	Estimated time in 1-man-weeks	Comment
WoT: Second iteration of less critical performance fixes: Identity garbage collection : Stats show that less than 1000 of the 14000 WoT identities are still active (thanks to ArneBab). Thus, we need to GC abandoned identities.	12	Depends on both aggressive and not-too-aggressive Trust-value decay algorithm which could take some time to invent. Also depends on code to gather statistics for estimating age of Trusts (we're no blockchain so this is difficult).
WoT: Second iteration of less critical performance fixes: Process results of peer review of xor's Web of Trust bachelor's thesis (which was about last year's performance improvements).	8	Depends on very through testing and benchmarking of the resulting algorithmic changes.
Fred: does it provide sufficient benefits to re-add native acceleration for FEC, or crypto? Especially JVMs without gmp acceleration - Oracle on Windows.		IMHO bogus task: We're too small of a project to have the resources for doing micro-optimizations such as replacing parts of standard Java or even JRE code with more custom one

Task category / Task	Estimated time in 1-man-weeks	Comment
User Experience - make things easier and more enjoyable to use		
Single use node references with authentication token: Currently, two users need to add the node reference of each other to create a connection. With this, only one of the two would have to add the other. Might benefit from FOAF (see 3 tasks below) for connectivity if Opennet is off.	4	Might be more difficult than anticipated: Needs low level changes of network code to accept packets from peers we don't know yet.
Darknet invitation bundles: Feature for adding a single use node reference to an installer executable. People could hand out the installer executable to their friends to allow them to connect by Darknet instantly.	2	0.5 weeks for each of our 3 installers + 0.5 week headroom: Depends on figuring out a way to put this into binaries signed with Windows' (and Mac's?) own signing mechanism without damaging the signature.
Short node references: Currently, references fill almost half a sheet of paper. Improve this by uploading node references to Freenet itself as a random KSK, with for example 128 bit entropy to be ~ 25 letters. (Requires having peers already, but without the UI could just show the long one)	1	
Friend-of-a-friend connection suggestions ("FOAF"): Like the Facebook friend finder, Freenet could be improved to tell you about Darknet peers of your peers. You could then chose to add them as your peers. Part of the code exists.	2	
Friend requests, like in Facebook: With primitive FOAF, both peers would still have to add each other. With friend requests, peers of your peers could just request to connect to you. Could use similar UI to Facebook friend requests.	2	
Something like lcicle to ease setting up Darknet connections - perhaps add support for connecting nodes over text message or Signal?		IMHO bogus – this would be a usability nightmare: No average user will understand that installing the “Freenet app” on their smartphone wouldn't give them Freenet but merely allow them to install Freenet elsewhere and then use their phone to connect the “elsewhere-Freenet” with each other. They'd expect the app to provide Freenet itself and uninstall and give us a bad rating if it does not. We either need to support running a full node on android or don't.
Darknet chat improvements: Freenet allows sending messages to Darknet peers. The UI for that is very primitive. It could be improved, for example to be similar to Facebook chat. Needs fixes for high message loss probability.	4	Extra careful estimation since this will be a lot of JavaScript and web development can be difficult due to browser compat issues.

Task category / Task	Estimated time in 1-man-weeks	Comment
Maybe FLIRCP can add darknet N2NTMs as a backend?	48	Underspecified task. Likely a full rewrite, and a complex one because we'd be building a new P2P network (how do we communicate with peers of our peers, peers of their peers, etc.?)
FlogHelper (WoT-based blogging app): Accumulate flogs of other users . List them nicely. Add a search function. Use that to replace or amend the current concept of FProxy's default bookmarks as entry point for Freenet. Hundreds of blogs instead of ~10 bookmarks.	2	
Better feedback on progress: currently in FProxy progress is displayed in terms of completed blocks. Information on how many fetch / insert requests are running for each one, and the progress on them, could help make the default mode of operation seem like hanging.	4	May be rather difficult: When I added the "Last activity" column to downloads, I learned that fred UI does NOT obtain progress info by merely calling getters on the low level code. Instead, progress is "pushed" from the lower layers to the upper ones by events. IIRC the information subject to this task is NOT pushed yet.
Make robust search functionality part of the default installation [task is possibly not specific enough: Complexity depends a lot on search target: Freesites? FlogHelper flogs? Freetalk messages? WoT identities? Sones? Filesharing?]	48	Under-specified task. Assuming the worst case of a real distributed, spam-proof freesite crawler - which is an unsolved research topic.
Add ability to comment on Freesites (fulfilled by babcom (is opt-in) / Freetalk / Sone?)	4	May be difficult to figure out potential security issues of freesite HTML potentially interacting with the inlined code.
Frost-alike communication system supporting easy usability, encrypted and read-only boards and hidden instant messaging between 2 or more ID's, integrated file transfer agent, mixed bulletproof ID'd and unsigned messaging. [= e.g. Freetalk. Likely requires the WoT speed tasks.]	48	Under-specified task. Assuming the worst case of recycling Frost to support Freetalk as a backend for the Frost UI. That'll be almost a full rewrite.
Improve the vulnerability and functionality of Flip [One anonymous user commented that you may as well re-invent FLIP]	48	Underspecified. Changing KSK-based and thus spam-able chat to secure WoT-based is likely a full rewrite and an unresolved research problem: How do we avoid having to poll all other users on the world? If we don't poll all of them, how do we keep the conversation readable? You cannot talk with people in a room if you only hear half of them.
Improve FProxy CSS3 support to allow better Freesite UI		Underspecified. Improve what?

Task category / Task	Estimated time in 1-man-weeks	Comment
Document the content filter		IMHO this is a bogus task: Why do this, with which goal?
Improve Shoeshop/blob handling [how?]		Underspecified.
p2p file exchange by time-set expiring keys [Imprecise task? Also violates main goal of Freenet of preventing censorship!?]		IMHO this is a bogus task. Besides “expiring keys” being censorship, I don't even understand why they would be related to providing filesharing.
Freenet key links directed to a dedicated Freenet browser, opennet [= non-Freenet Internet?] links to the system default like in the AutoHotKey script "Freenet Browser Director"	12	Browsers have degraded into being almost full operating systems, maintaining our own copy of one will likely be a lot of work.
Warn users not to use Chrome, but instead an open source/free software browser	0.5	
FLIP or FLIRCP could use work to become official plugins, maybe with a webclient instead of having to install a potentially privacy leaking 3rd party one. [Likely requires the WoT performance work in the “Speed” section.]	48	Requires solving the spam problem, which is very difficult. See above task “Improve the vulnerability and functionality of Flip”

Task category / Task	Estimated time in 1-man-weeks	Comment
Security - Make Freenet more secure against attacks		
Fix plugin data leak : None of our official plugins deletes its database when the user uninstalls it. They also do not obey the "PANIC" button of Freenet which should delete all client data.	2	While adding the actual code is very little work, setting up development environments for each of them is a hassle which might consume more time than anticipated.
Fix lack of plugin data encryption : WoT, Freetalk, and potentially other plugins do not encrypt their databases when the user has configured Freenet to encrypt user data. Should be changed to obtain an encryption key from Freenet to encrypt their stuff.	4	Same as above except that the actual code is more work.
Add simple tunnels. The easiest approach here may be to have some automatically added temporary time limited or data throughput limited FOAF connections that form simple tunnels. Tunnels should have a probability to use for high HTL inserts and requests based on security settings for the node.	48	The core network is 75 000 lines of code. As toad is currently not available for hire, a new developer will have to spend a lot of time in understanding it before being able to do this.

Task category / Task	Estimated time in 1-man-weeks	Comment
Technical debt - Stuff that will make future development faster		
fred: Documentation is said to be poor. Improve it. Would help people other than Matthew to understand the code. Good to do this now while Matthew is still available as a volunteer and we can thus maybe ask him questions.	12	Fred is 264 000 lines of code. As toad is currently not available for hire, a new developer will have to spend a lot of time in understanding it before being able to do this.
fred unit test coverage is 20%. Improve it. Would help people other than Matthew to understand the code. Good to do this now while Matthew is still available as a volunteer and we can thus maybe ask him questions.	12	Same as above.
WoT unit test coverage is 45%. Given the lower total line count, it would be possible to complete it to 100%. That would be a preparation of the major "refactoring" of replacing db4o with a different database. May become necessary: db4o's vendor has abandoned it.	12	Been volunteering on this for half a month (not full time). Depending on whether the planning period extends my volunteering this may progress further.
Alternate task instead of the above tests/doc: Permanently use 1 day per workweek (= 20% of total time) to deal with any kind of technical debt whatsoever. Allow the developer to choose freely which issues to deal with so even unimportant technical debt will be dealt with eventually.	(constant cost per week, no estimate needed)	
Alternate task instead of the above technical debt day: 1 day/week as "do whatever you want to"-day like Google does. Example activities: Generic education (new Java language features, programming patterns, algorithms, papers); Wiki; Drafting new apps; Tests; Doc; Mailing lists	(constant cost per week, no estimate needed)	
Fred: Migrate to the Gradle build tool - Gradle can provide dependency fetching and verification support and then defer to the existing ant build. (Or we can move the project to fit standard directory structure and then the Gradle build becomes a lot easier to write.)	2	nextgens has provided Gradle scripts but we're still on Ant: Doesn't work w/ Eclipse, needs bugfixes; developers need training.

Task category / Task	Estimated time in 1-man-weeks	Comment
Outreach - Stuff that will help attract users, developers, and donors to Freenet		
Language: Lower threshold of entry by working on how to install Freenet	1	Will cause a bit of team discussion.
Language: Add these strings for install to translation resource. Bonus for looking at this .	4	
Language: Simplify website language. Boil this into this for all pages.	4	Will cause a bit of team discussion.
Language: Fix current low-hanging problems with UI language, as per issues on transifex.	8	There are 377 issues Transifex, each will probably take half an hour to resolve because they're often imprecise, that's 5 weeks, rounding up to 8.
Language: Add Transifex issues to main bugtracker	2	Assuming 10 minutes per issue to account for the fact that they're often very imprecise.
Language: Establish overview of, and do away with needlessly difficult words	4	We have 54 000 (non-unique) words in all our resources on Transifex, this might take some time. Also this task is under-specified: What is this about, only fred or also all plugins & installers?
Language: Document how Freenet works, with a infographic, on the Wiki, detailing the remainder of speciality words.		Underspecified: There is a lot of things about Freenet. Which of them shall be documented?
Website redesign	4	While the redesign is finished, the content still needs to be migrated.
Website redesign: Migration to static Github Pages		
Website redesign: New CSS redesign		
Website redesign: Migration of more detailed documentation to Github Wiki, allowing website to focus on onboarding new users [Notice: we have 3 non-Github Wikis currently, 2 of those have already been waiting for migration into the main one for years.]		
'How to' YouTube video series: A couple of (maximal)-5-minute videos, each explaining one important practical aspect of using Freenet. Do this because currently we demand users to read far too much (FAQ, wiki, etc.). Example of ZeroNet .	4	Underspecified task, assuming worst case of also needing to do one for every app such as WoT/Sone/FMS/FlogHelper.

Task category / Task	Estimated time in 1-man-weeks	Comment
Update FSNG: Show web interface usage rather than Thunderbird to match modern user expectations		
Update FSNG: Use a modern Windows installation with the default theme for screenshots	0.5	
Legal docs: Have resources akin to the Tor legal FAQ for exit node operators, created by a legal expert on the official project page, that can be shown to LEA or users. Explain what it is used for, how to use it safely, best practices, plausible deniability, legal status, etc.	24	Lawyers are very expensive, I'd be surprised if this didn't suck up half a year of funds (while probably producing the same output we already have on the site as common-sense explanations anyway).
Legal docs: Sample response letters for the potential arrests or raids, especially for the recent events as some users were completely innocent. We need to educate LEA, more than ever, that it in itself is not a crime nor suspicious to run Freenet.	24	Same as above.
Legal docs: Instead of the above suggestions, just link the Tor legal FAQ. Tor is pretty similar and thus we shouldn't need to create our own one. Notice especially that there are 200 countries in the world and 200 lawyers would be pretty expensive.	0.5	
Fundraising: While volunteers did help with fundraising, they don't seem to enjoy it very much – we've contacted much fewer entities than those listed in the Wiki . So also have Freenet's employee do it. Important as Matthew will require much higher level of funding to come back.	8	This is estimated for how much effort it would be during our 1 year of funds we have available.

Task category / Task	Estimated time in 1-man-weeks	Comment
New core features - Enable using Freenet for new kinds of tasks/usecases		
Filesharing (#1 request on uservoice with 700 votes, not yet implemented but could easily be built upon Freetalk)	12	2 weeks for attachment support on Freetalk posts, 4 weeks for directory-uploader with notifications, doing posts, and asking user to categorize files, 2 weeks for search UI, 4 weeks for file ratings
Keepalive: This is very popular on uservoice. It could be improved significantly with some integration into the web interface and minor backend changes.		
Social networking (Sone, not enabled by default due to WoT performance issues and issues of its own)	36	Biggest issue of Sone is that it keeps everything in memory instead of using a database. This must be fixed to prevent OOM and for performance reasons (= to avoid it from re-downloading everything upon restart). Non-transactional code is very different from transactional code, so this will be a lot of work unfortunately. And then months of chasing bugs will follow ...
Blogging (FlogHelper, not enabled by default due to WoT performance issues)	0.5	While the UI of FlogHelper is not beautiful, it does at least work well. So if we fix WoT, we probably can just enable FlogHelper by default without any changes to it.
Forums (Freetalk, not enabled by default due to WoT performance issues and issues of its own: Freetalk itself needs its performance fixes in the "Speed" section. Also it needs general usability / security / bugfixing / polishing work which this entry shall be about.)	16	4 weeks for generic bug-fixing plus: Besides the event-notifications task at the "Speed" section, the bare minimum of scalability fixes to Freetalk is: 1) Limiting of days to download backwards = 1 week; 2) GC-ing of messages older than the limit = 1 week; 3) Relaying messages of other people to not have everyone subscribe to everyone else's USK = similar $O(N^2)$ fix as the first entry of WoT at the "Speed" section = 10 weeks.

Task category / Task	Estimated time in 1-man-weeks	Comment
Mail (Freemail, not enabled by default due to WoT performance issue)	12	Freemail is not spam-proof currently. So this depends on amending WoT to support providing CAPTCHAs by FCP. And that depends on: 1) Changing the code which uploads captchas to provide more of them on demand 2) Changing event-notifications to support captchas.
Distributed version control (various tools for Git / Mercurial over Freenet, not finished yet and lack code review)	48	Under-specified: Do we want bare usability or do we want a full-blown UI which looks like GitHub? Assuming the worst-case of inventing a decentralized GitHub, which is a lot of work.
Transport plugins are a very popular feature request. There is code but it needs a lot of work, and it may be better to start from scratch. In any case it will involve breaking up a lot of monster classes (good for code quality), and a lot of knock-on effects on concurrency (less good!).	72	The core network is 75 000 lines of code. As toad is currently not available for hire, a new developer will have to spend a lot of time in understanding it before being able to do this.
Make dynamic web pages possible [likely bogus suggestion: We must not allow full JavaScript for the security issue of preventing tracking / de-anonymization. We usually use plugins instead such as Sone, FlogHelper, Freetalk, Freemail, etc. So you might vote for those above instead]	96	Under-specified: Does dynamic mean “inline some output of an officially reviewed plugin” or does it mean “allow a safe subset of JavaScript”. Assuming the worst-case of JavaScript: As soon as we allow freesites to run code which could access the network (= even only merely requesting more files from Freenet) an insane lot of security issues will arise which could allow de-anonymizing users. Given the complexity of modern browsers it would be a major research project to determine a safe subset of JavaScript which does not allow these attacks.
Add FCP calls to Freemail to setup a user and to change user passwords	0.5	
Do not reject Freemail messages when an ID is not known, queue them for a period and attempt to find the identity	2	

Task category / Task	Estimated time in 1-man-weeks	Comment
New keytype: PSK (programmable subkeys - shared ownership)	96	<p>Giant piece of work:</p> <p>1) PSKs just existing doesn't help, we afterwards need to change all client apps to use them as well. As the PSK workflow is very different from the SSK workflow, this would throw away years of work on SSK-based apps such as the whole WoT/Sone/Freetalk/etc. ecosystem and cause many more years of work there. And for which benefit? The solutions for making SSK apps scalable sounds reasonable, no need to avoid them before we did try them at least. We haven't finished trying them at all!</p> <p>2) This would touch a lot of the fred codebase. As toad is not for hire currently a new developer would have to spend much time in training himself to be able to do this task.</p>
New keytype: RSK (revocable)	24	<p>1) Without support from apps, this is not useful. And it would be lots of work to wire it in to all apps which use SSKs. And most of those apps are WoT-based so they probably wouldn't even need it: Revocation could be emulated easily on top of WoT, see WoT issue 6770.</p> <p>2) This would touch a lot of the fred codebase. As toad is not for hire currently a new developer would have to spend much time in training himself to be able to do this task.</p>