

Memorandum of Understanding

"Unhosted/RemoteStorage" Project

Number: 2013-08-019

The parties:

Stichting NLnet, domiciled in Science Park 400, 1098 XH Amsterdam, The Netherlands, referred to as "NLnet" in this document, represented by Jos Alsters,

and

Unhosted Project, domiciled in De Wickelaan 29, 2265DG Leidschendam, referred to as "Unhosted" in this document,

given that:

- RemoteStorage is a draft internet protocol currently being standardised in the Internet Engineering Task Force
- **Unhosted** wants to empower users of cloud applications to control their own data, by building a suite of open source tools around the RemoteStorage (draft) protocol
- **NLnet** wishes to contribute to the development and maturation of RemoteStorage;
- **Unhosted** is committed to advance both the RemoteStorage draft protocol and Unhosted applications;

agree to the following:

1. **Unhosted** has written the proposal "*Unhosted/RemoteStorage*" which is attached to this document as Appendix 1.
2. **Unhosted** is understood to be responsible for all aspects of the project including coordinating the project and allocating funds - in line with the terms of this MoU. Appendix 1 forms an integral part of this Memorandum of Understanding. If and where statements in these Appendix or other appendices are in contradiction with one or more statements in the main memorandum text, the statement or statements in the main memorandum text will prevail.
3. **Unhosted** commits to publicly reporting its progress on at least a monthly basis (more often is encouraged) and to maintain a public status page for the project. When technical, legal or personal circumstances require so, **Unhosted** may

alternatively send non-public updates about the status of the project to **NLnet**.

4. Overall project management shall be directed by *Michiel B. de Jong*, acting as project manager and shall be run under the oversight of **Unhosted**.
5. NLnet commits to donate an amount of maximum Euro **25.900** to *Michiel B. de Jong* in order to support the "*Unhosted/RemoteStorage*" Project. Amounts will be paid when the specified milestones or partial deliveries have been completed. The payment will be made by wire transfer into a bank account to be designated by *Michiel B. de Jong*, where the account holder is **Unhosted Project** or *Michiel B. de Jong*.
6. The source code and documentation of the Project will be made publicly available as open source under the *GNU Affero General Public License version 3* or higher.
7. When appropriate, **NLnet** and **Unhosted** might issue one or more joint public statements announcing the project and NLnet's financial support. **Unhosted** is also encouraged to visibly and vocally acknowledge NLnet's contribution where possible - e.g. through its website, promotional materials, in presentations and in source code.
8. Funding to any particular group or person will be on the understanding that they coordinate their activities in agreement with **Unhosted**, in the spirit of cooperation, and in an effort to achieve the successful release of the results of the "*Unhosted/RemoteStorage*" Project.
9. This Memorandum of Understanding cannot be seen as any kind of employment agreement. All donation payments are made as a grant payments to the **Unhosted** project. **Unhosted** is responsible for paying any and all taxes with regard to this grant.

On behalf of NLnet

On behalf of Unhosted

Jos Alsters

Michiel B. de Jong

(Place)

(Date)

(Place)

(Date)

Annex I: Project plan

1 Steps towards RFC status

The remoteStorage spec is currently published as an IETF internet draft. At IETF 87 in Berlin there was a dedicated BoF which raised industry interest. Mark Nottingham (Chair of the IETF HTTPbis Working Group), Paul Hoffman and other senior internet technologists stated that an upgrade from Internet Draft status to Request For Comments status would not only be possible, but even desirable. They also indicated that market parties such as Akamai, Cisco and Rackspace would be potentially interested. We want to improve the specification based on the comments given to us. We need to test the interoperability of the various implementations of the remoteStorage spec, and make sure we resolve any issues that come up. It is obviously hard to predict whether we can reach this goal within the 6 month renewal requested here, but we will definitely be able to take the most important first steps on the path to RFC status.

Milestones

- a) write and submit a new version of the internet draft (draft-dejong-remotestorage-02 spec) to the IETF (2.500 euro)
- b) write -02 compliant server as an npm module (2.500 euro)

2 Adding extensible support for "legacy" and hosted accounts to remotestorage.js

Users need to work with both existing data, and with data from other people and data stored in resources they do not fully control themselves. Therefore it makes sense for Unhosted users to be able to access popular services such as Dropbox and GoogleDrive through Unhosted, utilising the (provider-specific) javascript API's. We want to make sure that the Unhosted framework is made extensible so that users and developers can add any resource that can be accessed cross-origin (with CORS headers). Given that a lot of users may have their personal data currently hosted on a Dropbox or GoogleDrive account, some developers may want to publish their unhosted web apps in such a way that it offers an alternative for people who do not (yet) have a remoteStorage account. For instance, the option could be given: "Connect to your remoteStorage" / "Connect to your Dropbox". Even if our goal is to have everybody on remoteStorage, providing an alternative Dropbox option may make the library more useful to application developers. Also, the option to register for a remoteStorage account from the application provider can make a lot of sense. Users could sign up for a hosted account at the app they use first, and then reuse that account at another app provider later if they wish. This is feedback we have received from app developers, and we are listening to their request.

Milestones:

- a) Design pluggable extensibility mechanism for remotestorage.js (1.250 euro)
- b) Add working examples (Google Drive, Dropbox) (1.250 euro)

3 Documentation for users and developers

In order to position the remotestorage.js library and unhosted web apps as viable alternatives to end users and developers, we need to better document the many features of the library and the example apps.

Milestones

- a) update remotestorage.io documentation (1.250 euro)
- b) create an app store for Unhosted apps (1.250 euro)

4 Community and marketing of remoteStorage/unhosted web apps

The remoteStorage spec is reaching an ever higher level of maturity. Many "bugs" have been fixed in it over the last 2 years, and it is now on its way to its 6th improved version. Incorporating the corrections from Mark Nottingham (the current chief editor of HTTP) will boost both its quality and its credibility when talking to personal data storage providers like Rackspace, Cisco, Dropbox, Google, Microsoft, as well as to players who are interested in disrupting the current market power. Orange, Huawei and PhoneGap have already approached us, but we will also pro-actively contact other prospective supporters and implementers of our spec. We will also contact key industry players, and offer to give a presentation at their respective offices (in Silicon Valley or elsewhere). End users do not want to scourge code repositories like github. We want to make it easy for end users to start using unhosted apps, and so we aim to create a 'store' for web apps where people can find out which applications have been deployed on 5apps. We will encourage authors of unhosted web apps to make their apps available there as well as in other ways that end users can easily find them.

We want to contact existing small storage providers about offering remoteStorage. for instance ownCloud providers would be good targets. In addition we want to talk with big industry partners about remoteStorage.

Milestones

- a) Talks at Leiden university, Complicity, QConsSF, 30C3
- b) Organize PDX, Unhos and Unhost meetup
- c) Write another 10 episodes of the Unhosted Adventures blog, and make the complete blog cycle available as one downloadable, CC-licensed html-book.

Total amount for these milestones: (2.500 euro)

5 Core remoteStorage modules

The current contacts module is not adequate, and needs to be upgraded.

Milestones

- a) get the contacts module working properly (2.000 euro)

6 New feature: shared notification

In order to make users switch to unhosted applications these should be able to give the same level of platform integration across separate applications that 'bundled' applications from monolithic origin can and do provide. Currently there is no canonical way for unhosted applications to 'talk' to the user unless they are opened in their own window and the user is actively logged in to the application. Adding a **shared notification mechanism** will greatly improve the integrated user experience of Unhosted web apps.

Milestones

- a) Design extensible shared notification mechanism (2.500 euro)
- b) Implement shared notification mechanism in remoteStorage.js (2.500 euro)

7 New feature: portability Von Munchausen-style

If the license of an Unhosted app allows it, the user might want to pick up code he or she likes and store that code with his or her remoteStorage provider and/or his local storage in the browser. That way he or she no longer is exposed to changes at the source, and no longer 'burdens' the original source of the code using expensive bandwidth. The user is free to run the application directly from the localStorage (for offline use) or remoteStorage location (with possible advantages like lower latency, availability of TLS). We call this **Portability Von Munchausen-style**.

There are advanced use cases for running apps from remoteStorage like sharing/provisioning the application and/or its helper scripts with other users, cloning applications and their configurations across different unhosted accounts from the same user (similar to what dotfiles do for unix applications), automated scanning for known malicious scripts by the Unhosted provider etc. In general, it would help to prevent unnecessary dependencies as well as privacy and security risks. Self stored apps also mean the user doesn't have to remember all the unique locations of his or her favorite unhosted apps -- logging into a single instance would be enough to gain a path to access all of them.

Milestones

- a) Portability Von Munchausen-style (2.000 euro)

8 New feature: Hood.ie integration

The "NoBackend" movement is rapidly gaining popularity among web developers, with **Hood.ie** currently being the most popular implementation of a so-called app-agnostic backend server. We will combine the power of unhosted web apps and Hoodie servers by defining a discovery and authentication procedure that allows the user of a Hoodie-based web app to connect to their *own* Hoodie instance instead of the default one.

Milestones

- a) Hoodie integration (1.250 euro)

9 New feature: SocketHub integration

In order for SocketHub and Unhosted apps to integrate easily, **SocketHub integration** with the core Unhosted platform needs to be investigated. SocketHub platforms are already good in quality, but their scope is still not complete. There are still a number of platforms we need to add, and verbs we need to add to existing platforms.

Milestones

- a) SocketHub integration (1.850 euro)

10 New feature: Inbox Functionality

In order for a user to receive messages while the user herself is not online, it is necessary to have something like an "inbox" on a publically addressable port. We will implement a working proof-of-concept of a server-side service that implements this functionality, publish its source code under an MIT/AGPL dual license, and write a report about the findings and conclusions encountered in the process.

Milestones

- a) Explore inbox functionality (1.300 euro)