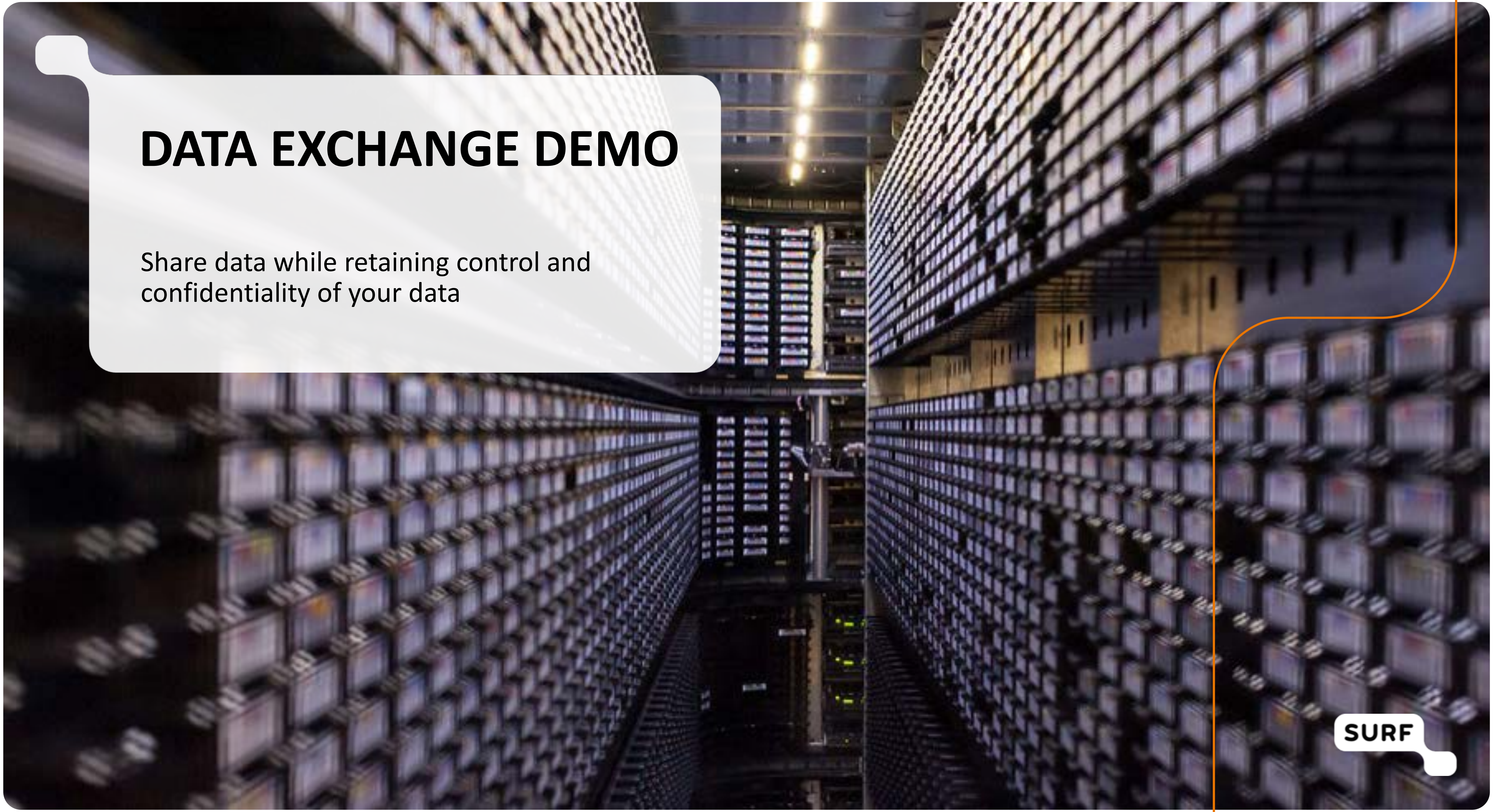


DATA EXCHANGE DEMO

Share data while retaining control and confidentiality of your data

SURF

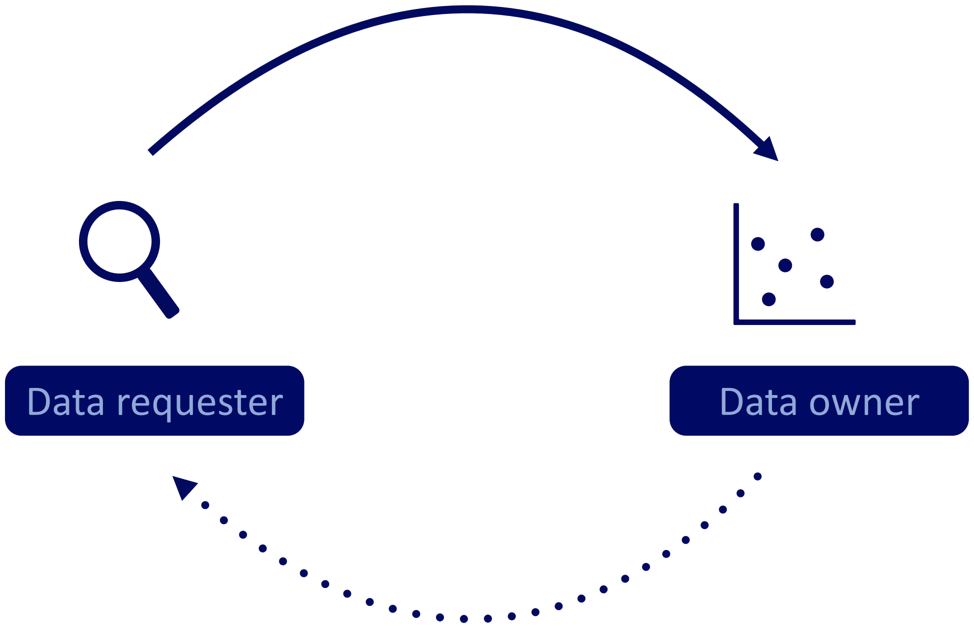
Version 2020-04-29

Gains and difficulties of sharing confidential data

+ Access to non-public data.

+ Potential new research and collaborations.

- More work to manage confidential data.



~ Possible to gain new insights.

- Risks on privacy and security.

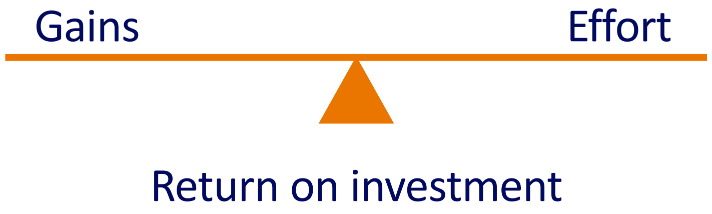
- Additional work without direct return on investments (ROI).

⚠ Gain is usually with the data requester, burden is with the data provider

Willingness to share data

ROI

Return on Investment (ROI) is determined by the balance between effort it takes to share data, and the gain received by sharing data



Gains

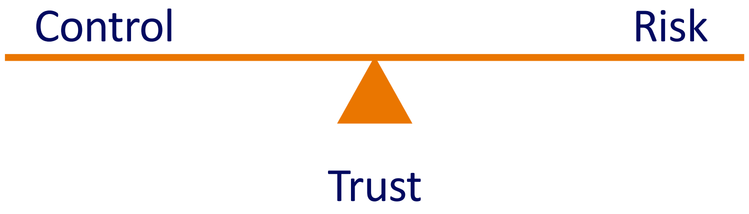
Effort

Return on investment

+

Trust

Trust is determined by the balance between the risks (due to privacy or competition), and the control (due to verification and security) of sharing and usage of data



Control

Risk

Trust

Type of Data Owners

Data aggregators

Health care
(Palga, NZa)
Social-economic (CBS, municipalities)

Hospitals
+ medical institutions

Hospital (AMC, vuMC, St. Antonius)
Insurance companies (Zilveren Kruis)

Onderzoekers
+ universiteiten

Universities (Twente, Wageningen, Groningen)
Researchers

Bedrijven

Friesland-Campina, Elsevier

Privacy sensitive

Competitive data

Example: Find the average income

Run #1

- 21 people
- Algorithm verified
- Outcome guaranteed not to be traceable to individual people

Run #2

- 22 people (same 21 and 1 other)
- Algorithm verified
- Outcome guaranteed not to be traceable to individual people

Even if individual runs are fine, combining two runs may reveal confidential data

Different Methods to Ease Data Sharing

Agreements

- Stipulation of what can/cannot be done
- Signing of contract or NDA
- Dispute resolution process

Registration

- Authentication
- Verification of credential
- Reputation score
- Policy framework
- Audit trails

Pseudonymization

- Filtering (on records)
- Pruning (on properties)
- Aggregation (combine records)
- Make coarse grained buckets
- Slight alteration of data
- One-way hashing
- One-time identifiers
- Synthetic data (mix records)

Data Vault

- Data source retains control
- Delegate permissions
- No central data lake
- Data marketplace

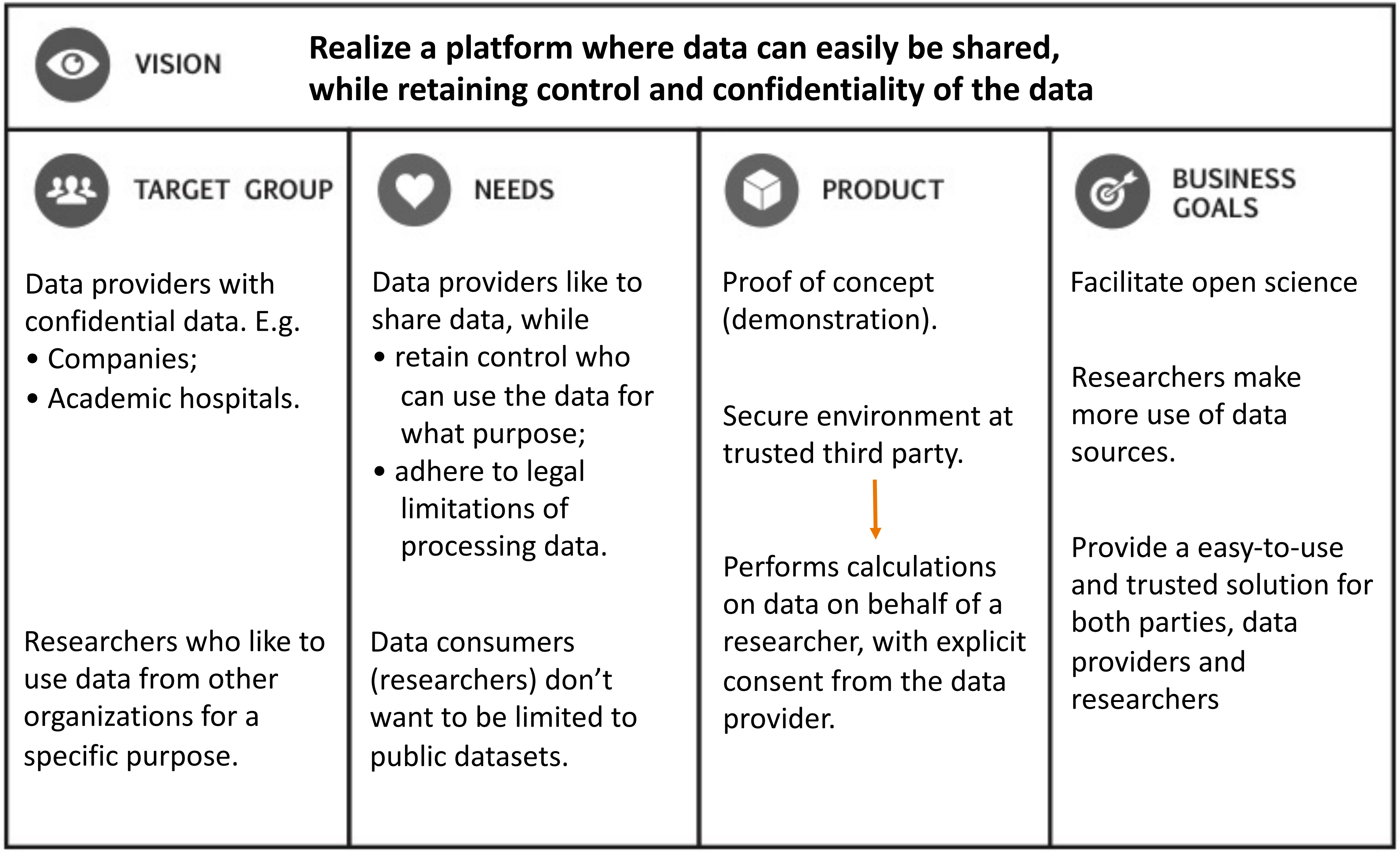
Secure Containers

- Bring algorithm to data
- At Trusted third party or at data provider
- Share output instead of data

Secure Computing

- Secure multi-party computation
- Homomorphic encryption
- Garbled Circuits
- Zero-knowledge proof

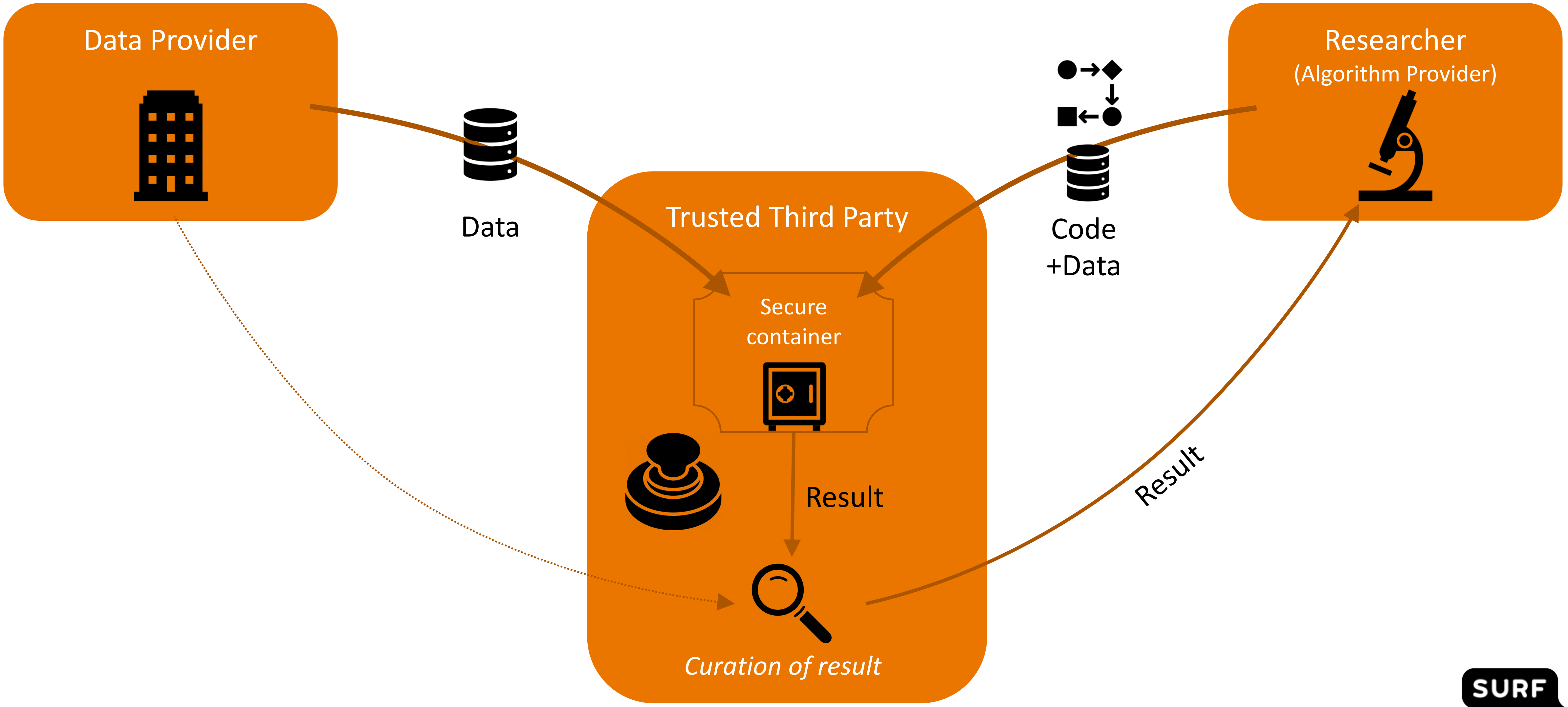
Data Exchange



VISION **Realize a platform where data can easily be shared, while retaining control and confidentiality of the data**

TARGET GROUP	NEEDS	PRODUCT	BUSINESS GOALS
Data providers with confidential data. E.g. • Companies; • Academic hospitals.	Data providers like to share data, while • retain control who can use the data for what purpose; • adhere to legal limitations of processing data.	Proof of concept (demonstration).	Facilitate open science
		Secure environment at trusted third party. ↓ Performs calculations on data on behalf of a researcher, with explicit consent from the data provider.	Researchers make more use of data sources.
Researchers who like to use data from other organizations for a specific purpose.	Data consumers (researchers) don’t want to be limited to public datasets.		Provide a easy-to-use and trusted solution for both parties, data providers and researchers

Collaborating without direct Sharing Data



Workflow

Share data	Data provider shares data with trusted third party; Researcher shares algorithm with trusted third party;
Request	Researcher makes request to data provider;
Verify algorithm	Data provider verifies requester and algorithm; ... and selects data set(s);
Run	Trusted third party creates secure container; ... mounts algorithm and data set; ... executes algorithm;
Curate output	Data provider verifies output and algorithm behaviour;
Release output	Once released, the researcher receives the output.

Permission Models

One-off permission	Trust a researcher	Run on a data stream
The data provider permits a researcher to **run** a specific algorithm **once** on a specific dataset.	The data provider permits a researcher to **run any algorithm on a specific dataset**. The permission can be revoked at any time. Example use cases: • the data provider trust the researcher to always write benevolent code • the researchers wants to tweak the algorithm, and run it on a sample dataset every time.	The data provider permits a researcher to **run a specific algorithm on any data set in a selected folder**. Every time a new dataset is added to the folder, the algorithm is automatically run. The permission can be revoked at any time, but is also automatically revoked as soon as a change to the shared algorithm is detected.

Currently supported permission models

Implementation (Proof of Concept)

- Working prototype
- Non-production (not scalable nor fast, not rigorously tested)
- Data stored at ResearchDrive (OwnCloud implementation at SURF for researchers)
- Data sharing: https://dataexchange.surfsara.nl/
  (simple password to emphasis it is a demonstration only: demo / dex)
- **Goal is to understand user requirements**

Mike Kotsur

Rienk Koenders

Sijmen Schoon

Tijs Teulings

Sander
van Wickeren

Axel Berg

Hylke Koers

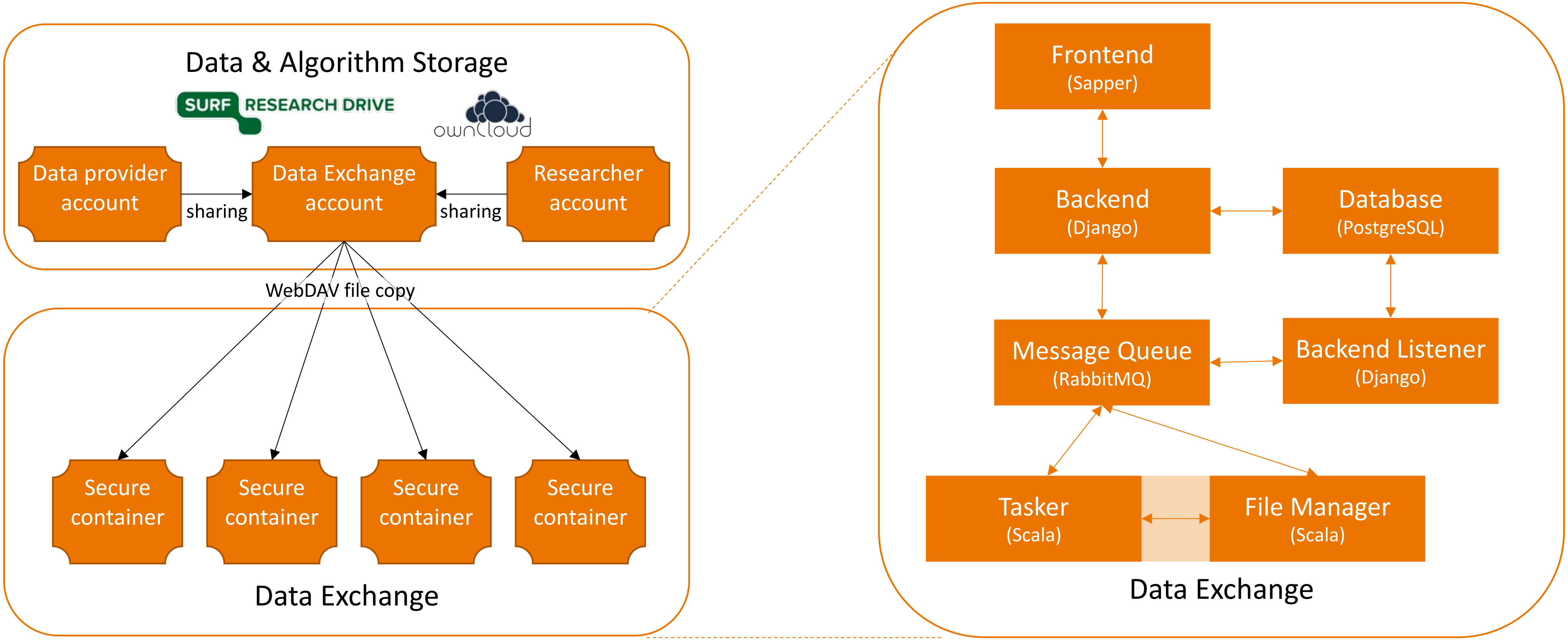
Gerben
van Malenstein

Freek Dijkstra

bit

SURF

Technical Implementation of the prototype



Data & Algorithm Storage

SURF RESEARCH DRIVE

ownCloud

Data provider account

sharing

Data Exchange account

sharing

Researcher account

WebDAV file copy

Secure container

Secure container

Secure container

Secure container

Data Exchange

Frontend (Sapper)

Backend (Django)

Database (PostgreSQL)

Message Queue (RabbitMQ)

Backend Listener (Django)

Tasker (Scala)

File Manager (Scala)

Data Exchange

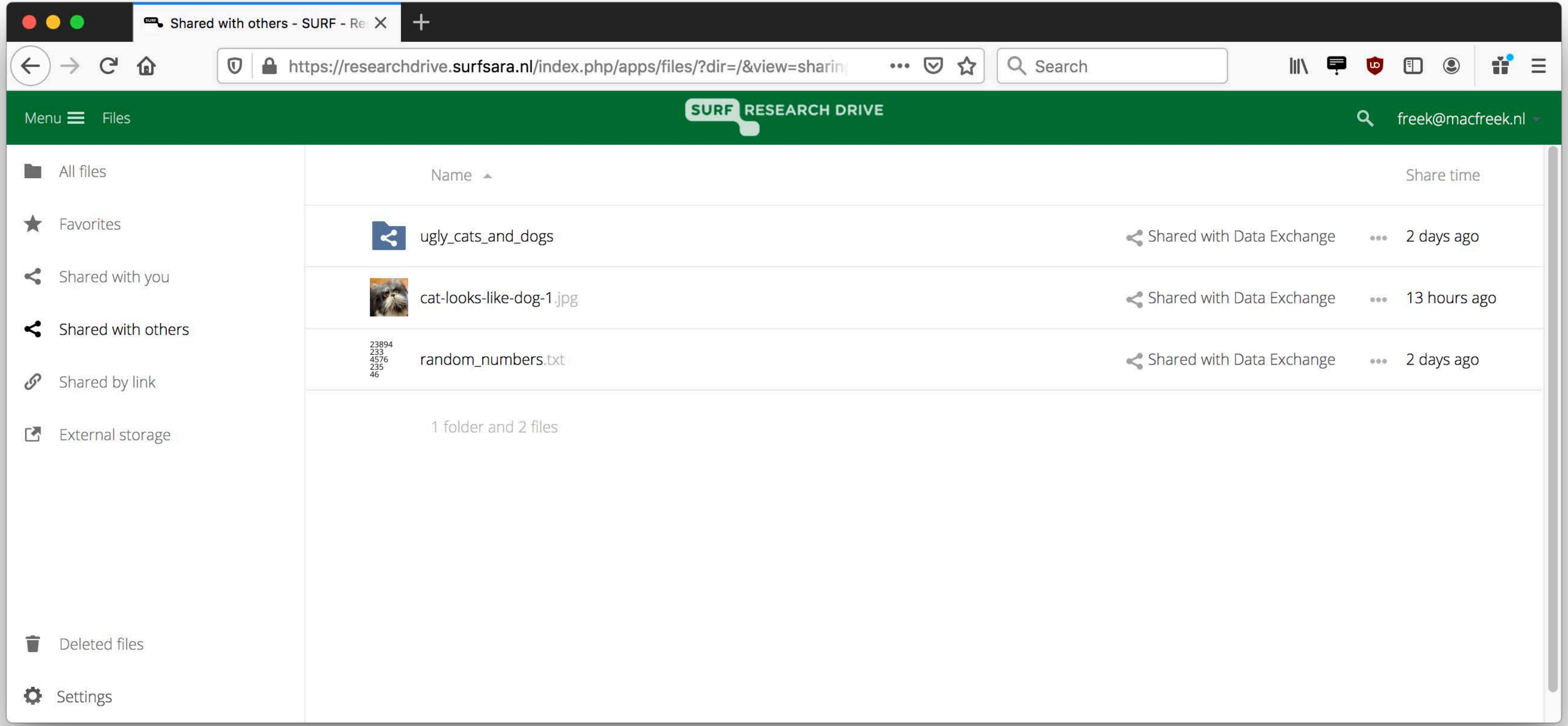
External integrations

Internal Components

Risks and Mitigations

Risk	Mitigation
Data is leaked to outside world	Researcher can never view the raw data, only the result
Data is used in other ways than intended	Data provider can review algorithm
Algorithm is leaked to outside world	Algorithm is not reviewed by data provider, researcher is trusted to write benevolent code only *
Output contains confidential information	Data provider curates output before releasing it to researcher
Malicious algorithm tries to copy data to remote server	No network access is allowed in secure container
Malicious algorithm tries to embed data in output	Data provider can review algorithm
Algorithm is altered after it is shared	Permissions involving this algorithm are automatic revoked
Researcher can no longer be trusted	Permission can be revoked by data provider at any time
Trusted third party can no longer be trusted	Sharing of data to trusted third party can be revoked at any time
Data is corrupt or data provider can no longer be trusted	Researcher should look for other data sources
Data can’t leave premises, not even to a trusted third party	Secure container can be run at premises of data provider *

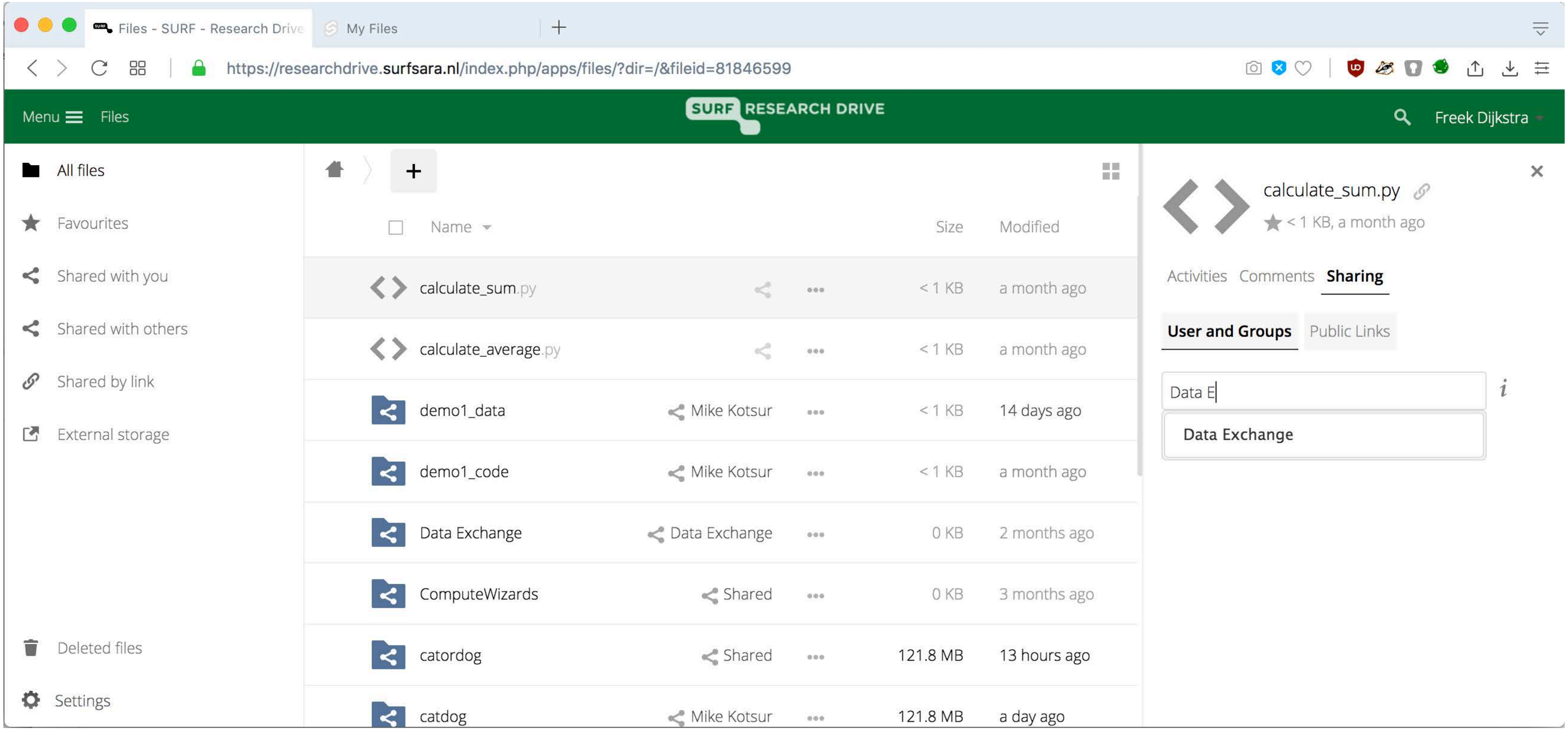
* Not yet implemented in the prototype



Name ▲			Share time
ugly_cats_and_dogs	Shared with Data Exchange	•••	2 days ago
cat-looks-like-dog-1.jpg	Shared with Data Exchange	•••	13 hours ago
random_numbers.txt	Shared with Data Exchange	•••	2 days ago

1 folder and 2 files

Data is shared with the Data exchange



Name		Size	Modified
calculate_sum.py		< 1 KB	a month ago
calculate_average.py		< 1 KB	a month ago
demo1_data	Mike Kotsur	< 1 KB	14 days ago
demo1_code	Mike Kotsur	< 1 KB	a month ago
Data Exchange	Data Exchange	0 KB	2 months ago
ComputeWizards	Shared	0 KB	3 months ago
catordog	Shared	121.8 MB	13 hours ago
catdog	Mike Kotsur	121.8 MB	a day ago

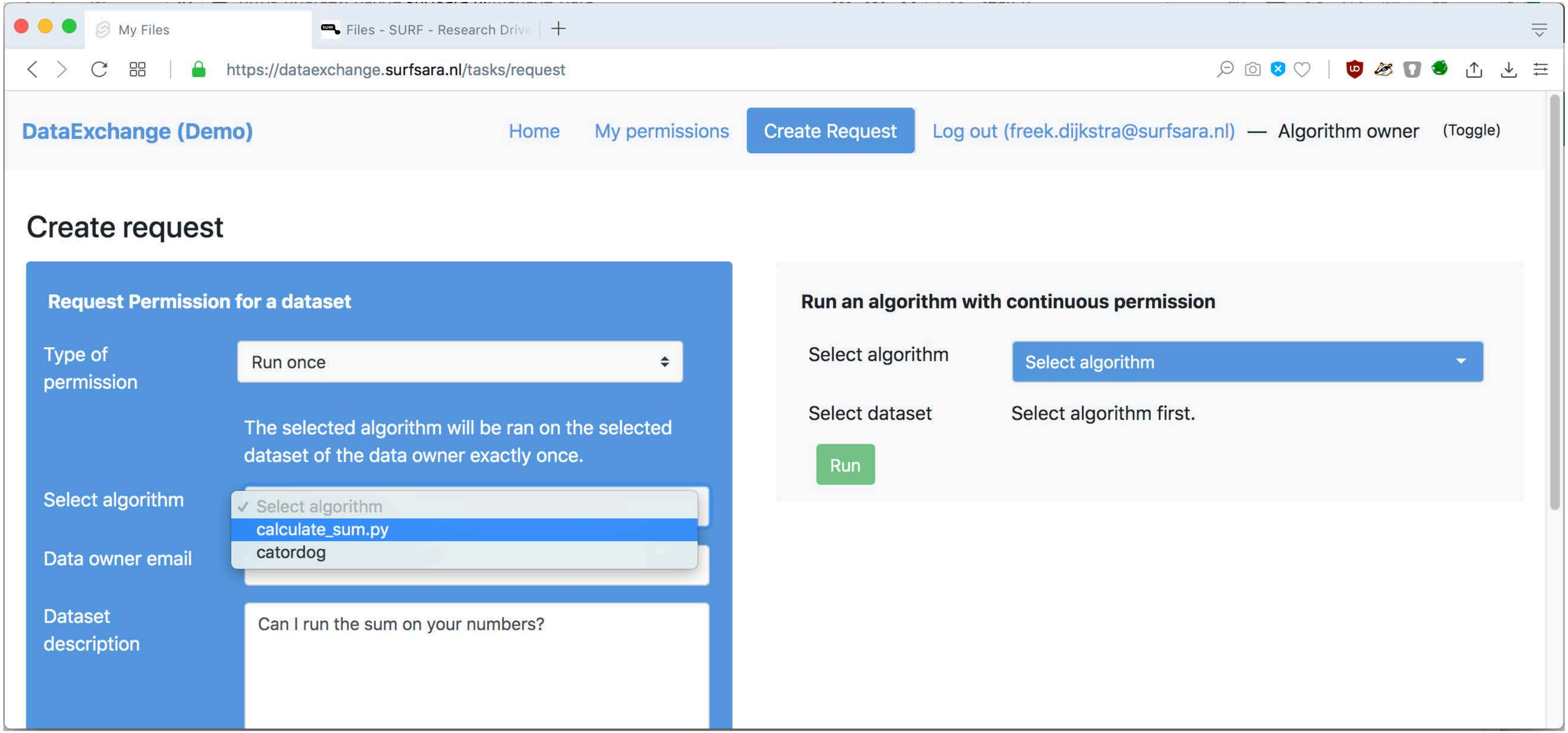
calculate_sum.py

< 1 KB, a month ago

Data E

Data Exchange

Algorithm is shared with the Data exchange by researcher



My Files
Files - SURF - Research Drive
https://dataexchange.surfsara.nl/tasks/request
DataExchange (Demo)
Home
My permissions
Create Request
Log out (freek.dijkstra@surfsara.nl) — Algorithm owner (Toggle)

Create request

Request Permission for a dataset

Type of permission: Run once

The selected algorithm will be ran on the selected dataset of the data owner exactly once.

Select algorithm: ✓ Select algorithm / calculate_sum.py / catordog

Data owner email

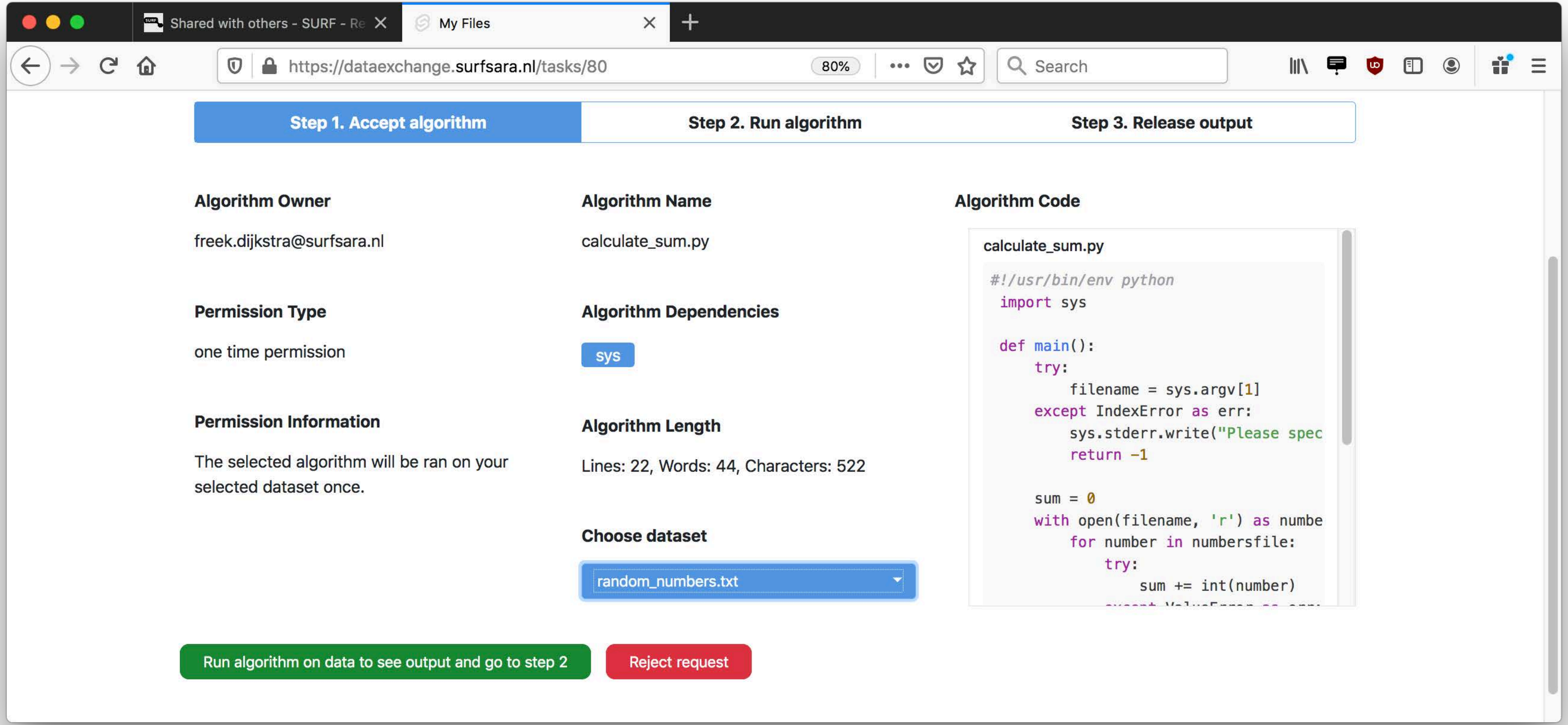
Dataset description: Can I run the sum on your numbers?

Run an algorithm with continuous permission

Select algorithm: Select algorithm

Select dataset: Select algorithm first.

Researcher makes a request to the data provider



Algorithm Owner

freek.dijkstra@surfsara.nl

Permission Type

one time permission

Permission Information

The selected algorithm will be ran on your selected dataset once.

Algorithm Name

calculate_sum.py

Algorithm Dependencies

sys

Algorithm Length

Lines: 22, Words: 44, Characters: 522

Choose dataset

random_numbers.txt

Algorithm Code

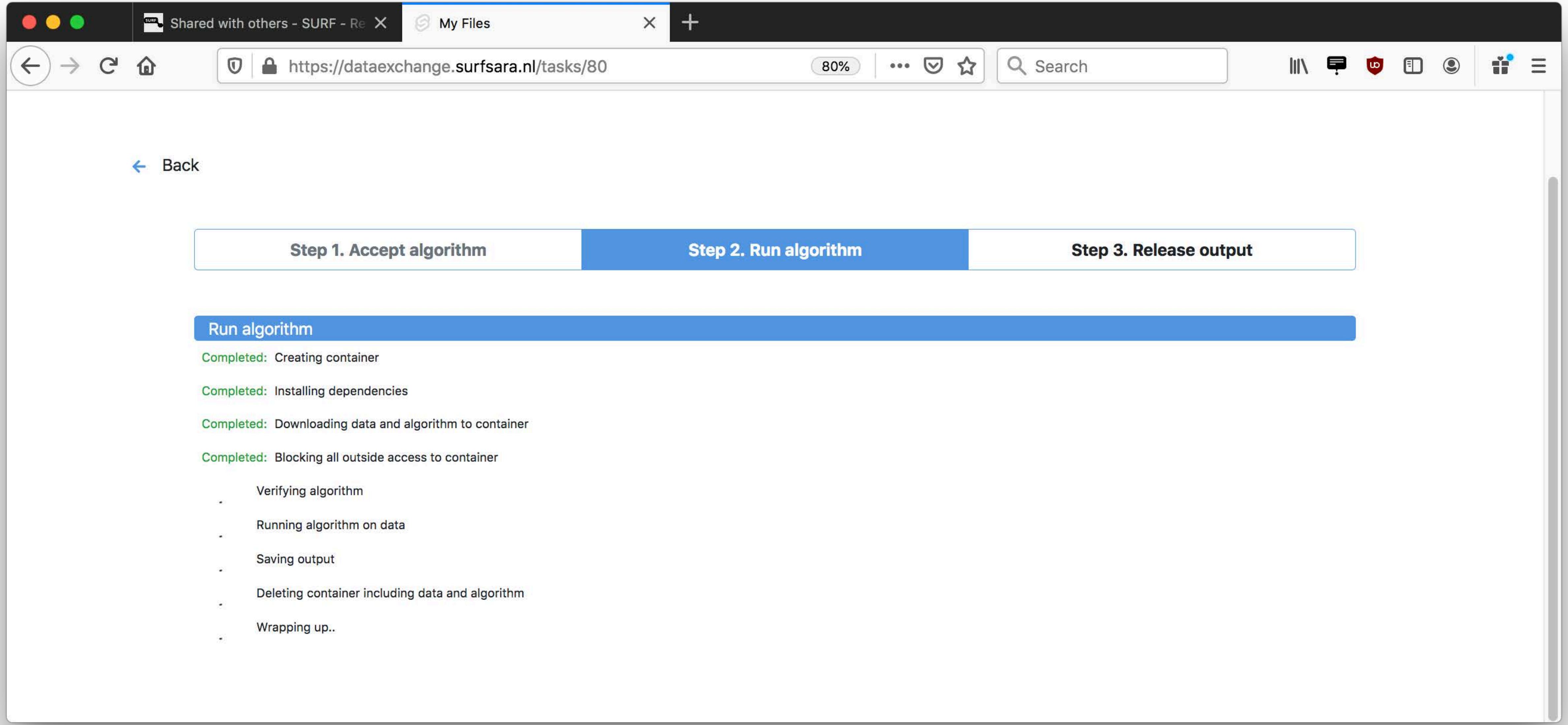
calculate_sum.py

```
#!/usr/bin/env python
 import sys

 def main():
     try:
         filename = sys.argv[1]
     except IndexError as err:
         sys.stderr.write("Please spec
         return -1

     sum = 0
     with open(filename, 'r') as numbe
         for number in numbersfile:
             try:
                 sum += int(number)
             except ValueError as err:
```

Data provider reviews request and selects dataset


Shared with others - SURF - Re
My Files
https://dataexchange.surfsara.nl/tasks/80
80%
Search
Back

Step 1. Accept algorithm | **Step 2. Run algorithm** | Step 3. Release output

Run algorithm

Completed: Creating container

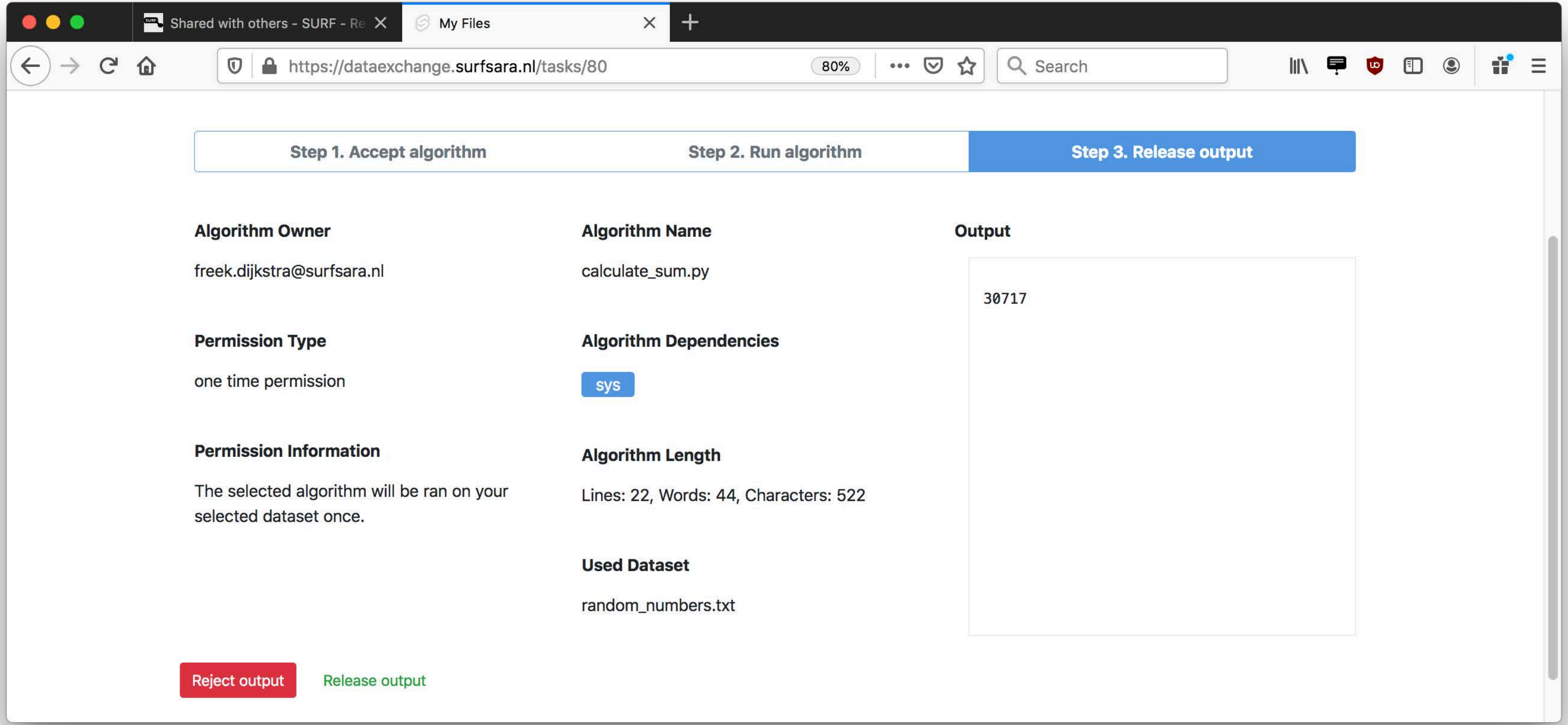
Completed: Installing dependencies

Completed: Downloading data and algorithm to container

Completed: Blocking all outside access to container

- Verifying algorithm
- Running algorithm on data
- Saving output
- Deleting container including data and algorithm
- Wrapping up..

Trusted Third Party runs algorithm on dataset


Shared with others - SURF - Re
My Files
https://dataexchange.surfsara.nl/tasks/80
80%
Search

Step 1. Accept algorithm | Step 2. Run algorithm | Step 3. Release output

Algorithm Owner

freek.dijkstra@surfsara.nl

Permission Type

one time permission

Permission Information

The selected algorithm will be ran on your selected dataset once.

Algorithm Name

calculate_sum.py

Algorithm Dependencies

sys

Algorithm Length

Lines: 22, Words: 44, Characters: 522

Used Dataset

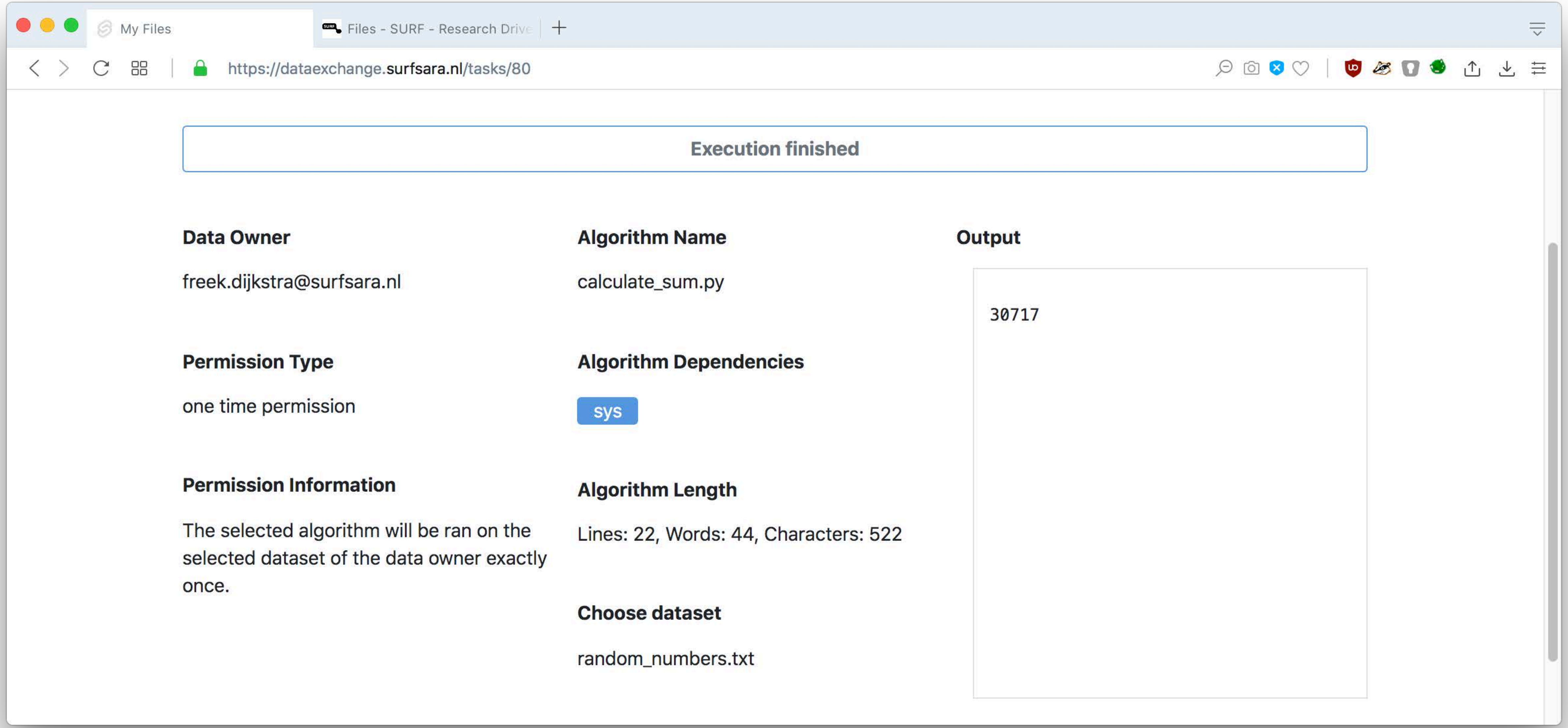
random_numbers.txt

Output

```
30717
```

Reject output | Release output

Data provider reviews output


My Files
Files - SURF - Research Drive
https://dataexchange.surfsara.nl/tasks/80

Execution finished

Data Owner

freek.dijkstra@surfsara.nl

Permission Type

one time permission

Permission Information

The selected algorithm will be ran on the selected dataset of the data owner exactly once.

Algorithm Name

calculate_sum.py

Algorithm Dependencies

sys

Algorithm Length

Lines: 22, Words: 44, Characters: 522

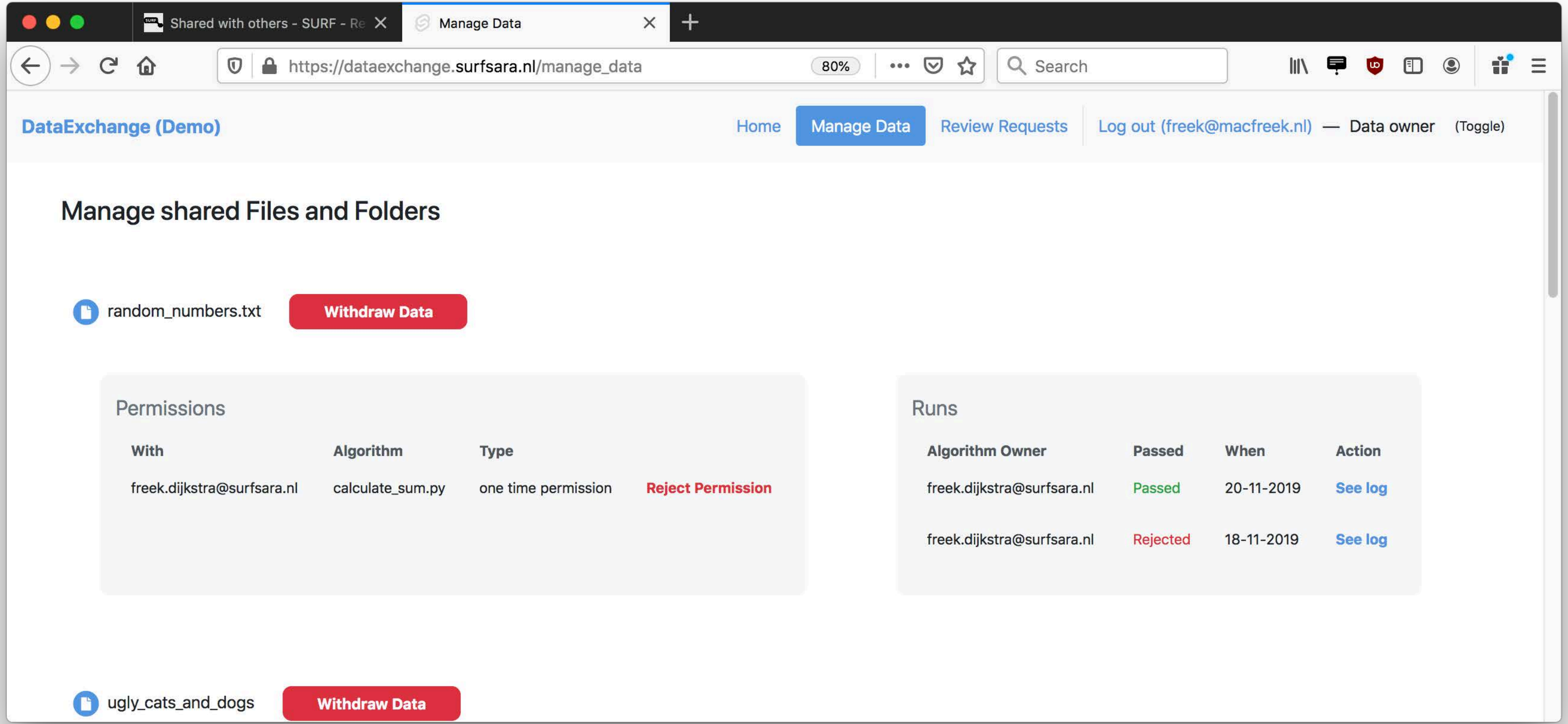
Choose dataset

random_numbers.txt

Output

```
30717
```

Researcher can see released output



Manage shared Files and Folders

random_numbers.txt

Permissions

With	Algorithm	Type	
freek.dijkstra@surfsara.nl	calculate_sum.py	one time permission	Reject Permission

Runs

Algorithm Owner	Passed	When	Action
freek.dijkstra@surfsara.nl	Passed	20-11-2019	See log
freek.dijkstra@surfsara.nl	Rejected	18-11-2019	See log

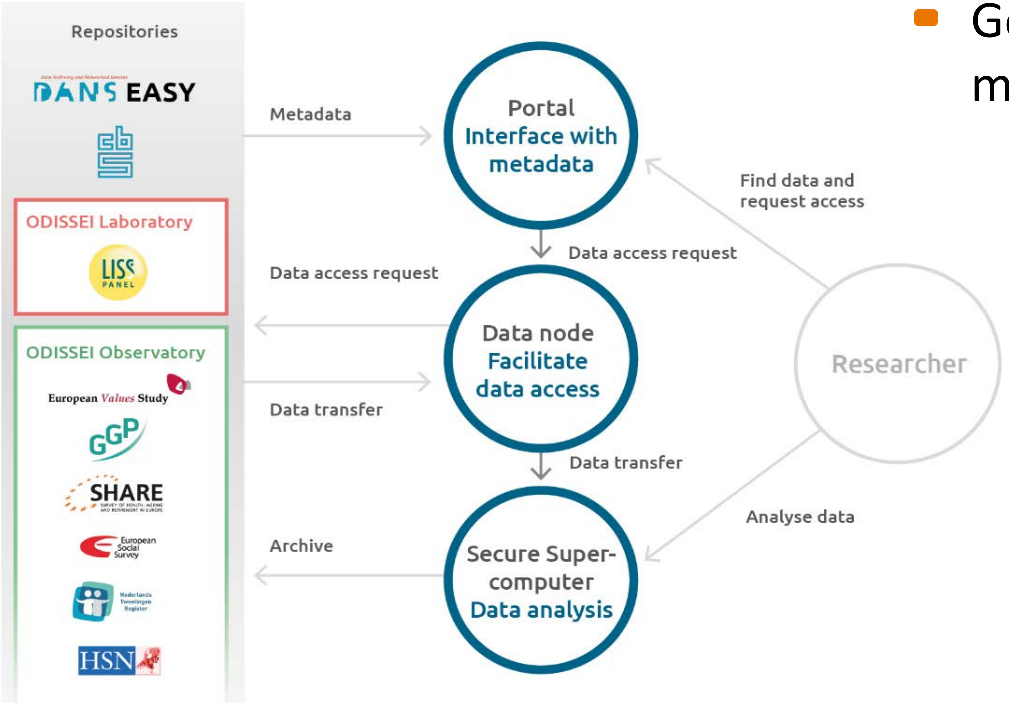
ugly_cats_and_dogs

Data provider can at any time withdraw permissions

Related Projects

ODISSEI Secure Supercomputer (OSSC)

- In production
- Processes CBS micro-data on Cartesius
- Does pseudonymization as well



AMdEX

- Collaboration of interested parties
- Initiated by Amsterdam Economic Board
- Goal is to build an infrastructure for multiple Data Marketplaces

Partnership Questions

- Who may benefit from a data exchange?
  - Are there researchers that want to use confidential data?
  - Who are the data providers in this case?
  - Under what conditions would these data providers release their data?
- What should the role of SURF?
  - Service provider; software developer; community manager; …
- Should SURF turn this prototype into a pilot?
- Are there other projects we should collaborate with?

Technical Questions

- Is a trusted third party the right approach?
- What is the trust relation?
  - Does the data provider trust the researcher?
  - Does the data provider trust the algorithm?
- More advances user scenarios (e.g. with 3 parties):
  - Patient trust a hospital with their data
  - Hospital trust a researcher with the patient data
  - What are the implications for the current demo with 2-part user-scenario?
  - Who gives what permissions, and is that a continuous permission? How to withdraw permissions?

COLLABORATION WITHOUT SHARING DATA

Freek Dijkstra

Freek.Dijkstra@surfsara.nl

www.surf.nl

Driving innovation together

SURF