JATOS for UiT

A beginner's handbook

v1.0.0.

jspsych

Prolific

JATOS

Just Another Tool for Online Studies

What is this handbook?

This document (in ongoing development) is meant as a starting point for people at UiT who want to conduct behavioural experiments and surveys online. It offers a basic explanation of the data protection regulations that we, as researchers based in Norway, have to adhere to with regards to online research; an introduction to JATOS, our chosen tool for running studies from UiT's own servers; and a collection of tips and resources to help you on your way. Though it contains some "tech support" here and there, this is **not** a manual or tutorial guide for any of the software tools discussed herein: if you're a researcher at UiT who wants to run an online study, read this first, then e-mail me (the author) or chat with me in our Teams group (O365-Online Research) and we'll take it from there.

Written and maintained by Myrte Vos

myrte.vos@uit.no

Contents

1 What is JATOS?

JATOS[1] is short for "Just Another Tool for Online Studies". It's a free, open source piece of software that you can install on your own server, that makes it much easier to run studies on the internet. It's especially compatible with OpenSesame (OS and JATOS share a help forum!), but also works well with other Javascript libraries written for experiment-building, such as JsPsych and lab.js. Have a look on Cortex, the demo server for jatos.org, to see some examples of the kinds of studies you can run with JATOS. And here's the very accessible documentation site for JATOS.

JATOS is created and under active development by Kristian Lange and Elisa Filevich, formerly resp. currently of the Humboldt Universität Berlin. Don't forget to cite JATOS when you use it to run a study!

1.1 Why JATOS (and not, for instance, Gorilla)?

Good question. JATOS gives us a lot of freedom and flexibility, but platforms like Gorilla, FindingFive, and Pavlovia (here is a very recent overview paper by Sauter, Draschkow and Mack; and here is a spreadsheet admin'd by UiT's Psychology department, about available tools for running behavioural experiments online) are more structured and user-friendly, and also come with fast and expert tech support. They cost money, of course, but that's more than worth it for the time and effort saved. Why not just use those?

The long answer to this question is in Chapter 3 of this handbook (see NSD and GDPR compliance). The short answer is: we're not allowed. The main issue is whether your participants' data passes through the electronic hands of a third party, which makes the question of who is legally responsible for that data more complicated. **However,** it's worth double-checking whether platforms like Gorilla are really off-limits to you! Things may not be hopeless if one of the following applies:

- The 'third party' you want to run your study with is located in, or has a server located in, Germany, Norway, or another country with stringent data protection laws;
- Your study is part of a collaboration with a partner university outside Norway, and you can make your collaborator the data owner;
- Your study doesn't collect **any** identifying personal data.

If you work in Norway and get permission to use (something like) Gorilla, **let me know**. In the meantime, we're (slowly) working towards getting the UiT administration's support for our (responsible) use of third party data processors, which, if we get it, will hopefully give us a lot more options.

1.2 Setting up a local installation of JATOS on your own computer

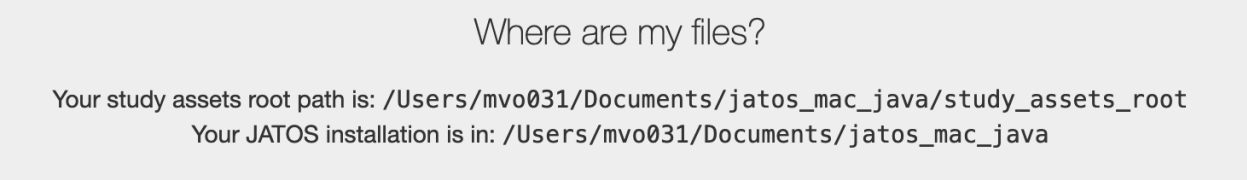
1.2.1 Why do I need a local installation of JATOS?

Isn't it already installed on UiT's own server? Yes, JATOS is all set up and ready here. But when you import a study you built to this server, all the files for that study go into a directory that you do not have direct access to.[2] That makes it annoying to make changes to your study (especially when you're still getting the hang of it): you keep having to delete and re-upload your study every time you need to change something.

[1] This section is derived almost entirely from *The idiot-proof guide to installing JATOS on your own computer*, also written by me.

[2] That may change soon! UiT's IT department is working on giving us access to individual accounts' asset files on Real JATOS. Needless to say you would NOT want to mess with those while your study is live.

With JATOS on your own computer, you know exactly where your files are, because it says so right on the home page:



So you can 'upload' your study to your local JATOS, test and edit it to your heart's content. Then when you're completely satisfied and certain that everything works, you can click the Export button on your study's overview page. That will package your study into a .jzip file, which you can upload to the "real" JATOS server, where it will be ready to go straightaway.

1.3 How to install JATOS locally

1.3.1 First steps

- Go to this webpage and read it.
- OPTIONAL BUT RECOMMENDED: watch this video of Kristian Lange talking you through the installation process.
- Click on this link (or the download link on the webpage), which will take you to JATOS's GitHub page.
- Download whichever zip file is appropriate for your operating system. [Note: only download *jatos.zip* if you're certain that you have the latest version of Java installed!]
- Move the zip file to a location that is convenient for you. (I put it in my Documents folder.) **Do not put it in a folder that syncs across devices, like Dropbox or OneDrive.**

1.3.2 Windows

Unzip the file, then go into it. You should now be able to just double-click on the `loader.bat` file. A command window should open and run your local JATOS installation. If you want to stop JATOS, just close this window.

1.3.3 Unix (Linux and Mac)

Open Terminal.[3] In the following steps, the code you're supposed to type is highlighted yellow.

1. Go into the folder where your JATOS zip file is. E.g., using my own Terminal as an example:
   `blia-hsl-m0023:~mvo031 cd Documents`
2. Unzip the file.[4] This should trigger an avalanche of code. (That's good.)

[3] If you don't know what that is or where to find it, or you do know but you don't dare/want to interact with it, please contact your nearest "computer person" for help. I recommend buying them a coffee first.

[4] Some people's browsers unzip files automatically at download. If that happens, just skip Step 2. You can also make your browser stop doing that in your browser's Preferences.

```
blia-hsl-m0023:Documents mvo031$ unzip jatos_mac_java
```

3. Go into your newly unzipped JATOS folder.
   ```
   blia-hsl-m0023:Documents mvo031$ cd jatos_mac_java
   ```

4. Start JATOS.
   ```
   blia-hsl-m0023:jatos_mac_java mvo031$ ./loader.sh start
   ```

5. You should be seeing something like this:
   ```
   JATOS uses local Java
   Starting JATOS...  started
   To use JATOS type 0.0.0.0:9000 in your browser's address bar
   ```

Your local JATOS installation will run in the background. If you want to stop it, just type `./loader.sh stop` in your terminal window. Go to your browser, copy-paste `0.0.0.0:9000` into the address bar and press Enter. You'll be asked for login details: the default username and password are both `admin`.

You should now be looking at your own JATOS homepage! I recommend you download any of the example studies and try uploading them to JATOS. When that's successful, take a look in the root assets folder in your JATOS folder: that's where you'll find all the files associated with that study. When you edit them, those changes will be directly implemented in the study on your local JATOS.

IMPORTANT NOTE: your local JATOS installation has a **black** toolbar at the top of the page, and the real JATOS has a grey toolbar! Don't be like me and send your long-suffering co- workers pilot links from the wrong JATOS. Your local JATOS is not connected to the internet, even though it runs in your browser. (You *can* hook up the JATOS on your personal computer to the web, which may be a useful option in some cases - but you have to be careful to keep your computer running for as long as your study is live, and other such matters.)

1.4 More resources

Sebastiaan Mathôt, a neuroscientist for the University of Groningen and the main developer of OpenSesame, is creating a series of tutorials on using JATOS (in combination with OSWeb). Have a look at the first three, and also at his other OpenSesame videos.

- Video Tutorial 1: using inline Javascript in OpenSesame
- Video Tutorial 2: distributing links
- Video tutorial 3: downloading data

There is also an active help forum, which bundles support for OpenSesame, its web extension OSWeb, and JATOS in one place. If you run into any issues, the solution can probably be found there, and if not, the developers are generally quick to respond.

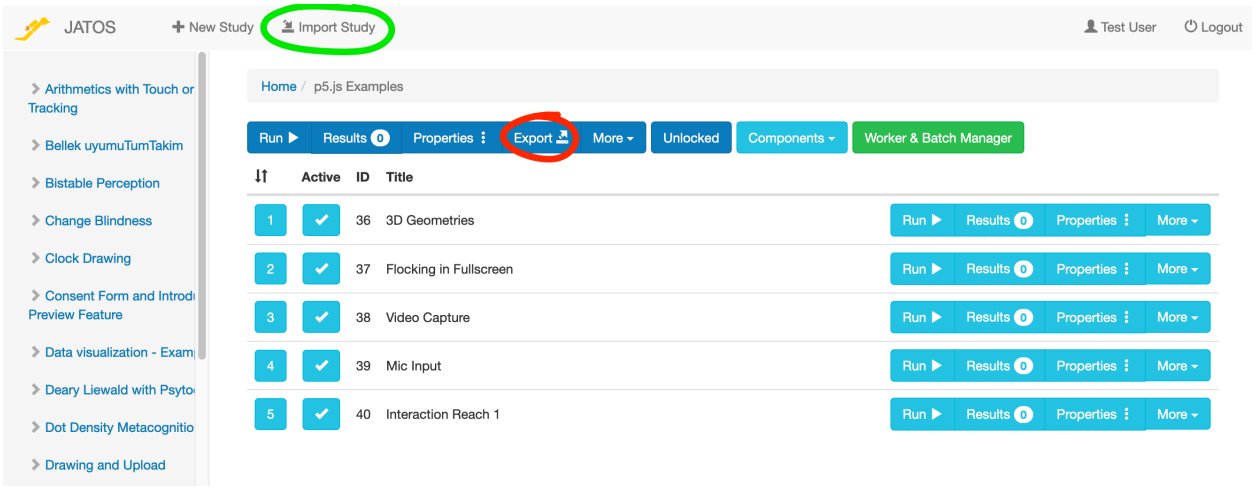
1.4.1 Donate to the developers!

If JATOS and OpenSesame are useful to you, please consider donating to the developers. You can donate to OpenSesame here, and you can "buy a coffee" for the developers of JATOS through the coffee cup icon in the bottom right corner of the JATOS introduction page.

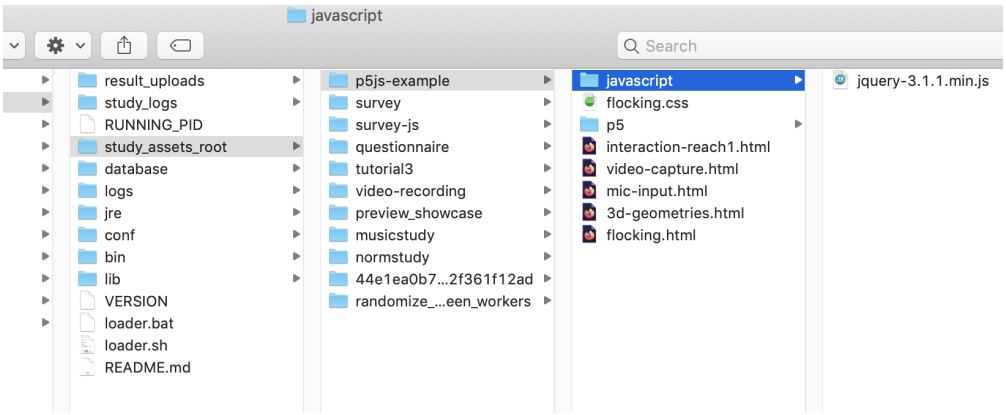
2 Getting started with JATOS

2.1 A quick tour

If you go to Cortex [5], the public server maintained by the developers of JATOS, you can click around and try different demo studies at your leisure. Download one of them by clicking "Export" (circled in red in the screenshot below), then go to your Downloads folder and change the file extension of the file you just downloaded from `.jzip` to `.zip` (don't worry, that won't break anything). Then, go to your local JATOS (which of course you've got open in your browser), click on "Import Study" (circled in green in the screenshot), and import the **zipped** file.



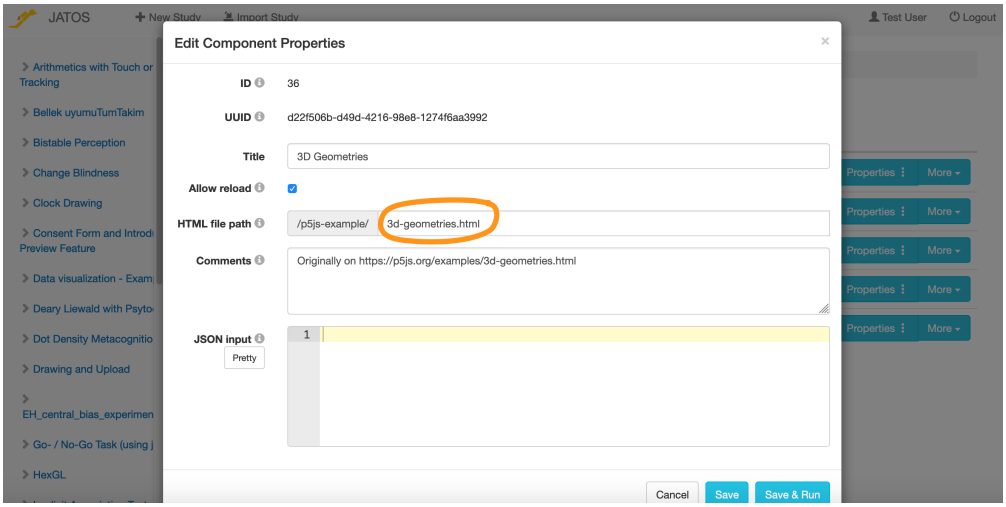
On your local JATOS homepage, you can see where are the study files are stored. They should be in a folder named `study_assets_root`. Go to that folder, and you'll see that it now contains one folder: the zipped file you just imported. Go into that folder and have a look around.



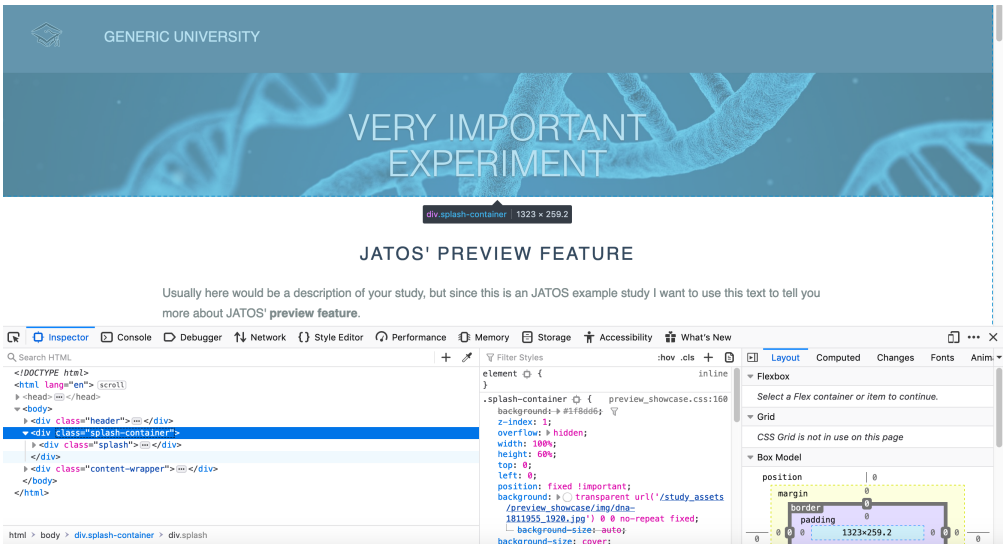
In this folder, you'll find everything the study needs to run on JATOS: the study's *assets*. (Except the very important file `jatos.js`, which is stored on JATOS itself.) The assets folder will look different for every

[5] username: test@jatos.org
password: abc1234

study, but it *must* contain a `.html` file for every individual component in the study. If you go back to this study's overview page on JATOS, you will see that there are multiple components, and that you can run them, view their results, and edit their properties individually. Click on the **Properties** button.

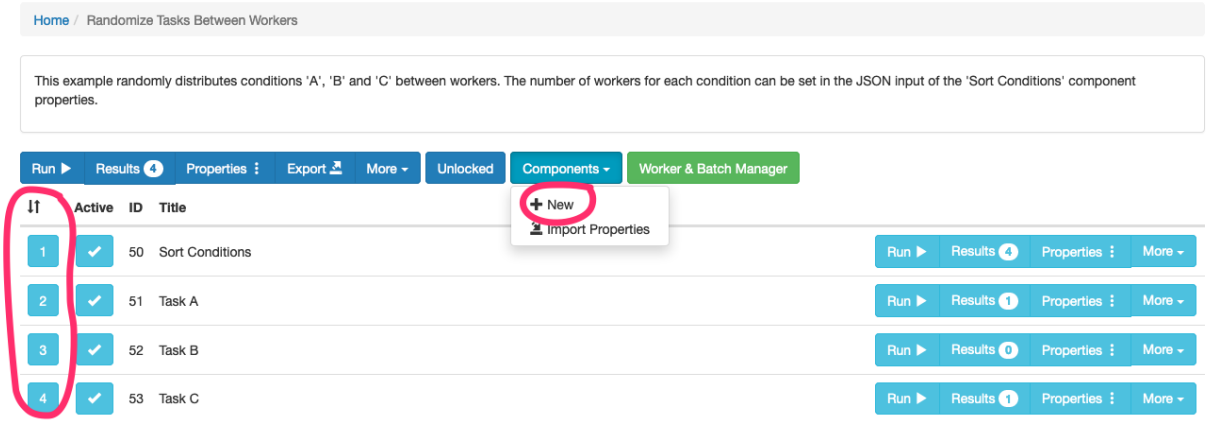


A component needs an HTML file to run. The HTML file contains instructions for how the web page should look, often in combination with one or more CSS files, which specify things like margins, fonts, button design, etc. You can inspect all this code by right-clicking on the web page (in fact, any web page!) and selecting 'Inspect element'.



This may seem like a lot to take in, but exploring an HTML file you want to adapt via the browser inspector is a nice way of identifying the lines of code you need to target. By clicking on or hovering over lines of code in the Inspector, the relevant elements in the web page are highlighted. You can edit the HTML code (and the linked CSS code, under {}`Style Editor`) directly in the Inspector, and view the changes straightaway. To make permanent edits to the source file, I recommend using a code editor - any text editor will do, as long

as it doesn't change your code in any way, but a code editor is nice because it makes code more readable through e.g. colour and indenting, and it flags syntax errors. There are many options, including online editors; I use the (free) desktop editor Atom.
The browser Inspector also comes in handy as a debugger, when you try to run your edited experiment in local JATOS and it inevitably doesn't. There will be many errors, especially in the beginning! Keep calm, copypaste the error message into Google, and eat something.



Build your study by adding new components, one by one. Does your study have an information and consent page, a demographic survey, and an OSWeb experiment? Then all three of those have their own HTML file in your assets folder. You can drag-and-drop them by the component numbers (circled in pink) to reorder them.

2.1.1 Uploading your study to "the real JATOS"

Once your study looks and behaves exactly the way you want it to (remember that you can't send working links from Local JATOS! You still need to pilot your study on the Real JATOS!), you do the same thing you did at the beginning of this tour: click **Export**, which creates a `.jzip` file. Go to the real JATOS, log in to your account (if you don't have (access to) one, e-mail me!), and use the **Import Study** button at the top of the page to import your `.jzip` file.

2.2 Tools for building basic experiments

2.2.1 OpenSesame

OpenSesame is an experiment-building program with a Graphical User Interface (GUI). PsychoPy is a different example of that kind of software; both are Python-based, both require very little programming knowledge.[6] To build basic experiments, use OpenSesame. If you're already a little familiar with Javascript you can also use the JsPsych library, see below.) We take 'basic' to mean experiments that present image and sound stimuli, and that collect keyboard and mouseclick/screen tap responses. To be very specific, 'basic' means the whole experiment can be built and run using functionalities that are supported by OSWeb, the web plugin for OpenSesame.

[6]The developers of PsychoPy (at the University of Nottingham) have created their own for-pay version of JATOS, 'Pavlovia'. In the same way OpenSesame has been made compatible with JATOS, PsychoPy can be run on Pavlovia - but not, so far, on JATOS. Pavlovia is hosted in the UK; see Why JATOS (and not, for instance, Gorilla)? for more information.

Björn Lundquist has written up a step-by-step tutorial for building a simple linguistic experiment in OpenSesame, available here, and Sebastiaan Mathôt himself has created a variety of written and video tutorials, including one for how to upload your OS experiment to JATOS.
OpenSesame is under constant development, so check for updates once every so often.

2.2.2 JsPsych

If you have fairly simple audio playback needs, you can get away with using OpenSesame. The `sampler` function in OpenSesame is compatible with OSWeb, but you can't do much with it: OSWeb can't fade in sound, or cut it off after some other event. OpenSesame's `media_player`, which plays video, is not compatible with OSWeb.

JsPsych is probably your best bet if you want to build an experiment with audio/video playback.[7] JsPsych is a good tool if you're a) already somewhat comfortable with programming (if you do all your data tidying and analysis in R, this is you), and b) growing frustrated with what OSWeb *can't* do.

JsPsych is a Javascript library for running behavioural experiments online. (A lot of the demo experiments on Cortex are written with JsPsych.) The chief developer of JsPsych, Josh de Leeuw, is currently (summer 2020) leading an online workshop on using JsPsych to build and run online studies. The archived tutorial streams, as well as a lot of other helpful resources, can be found here.
And here is a written tutorial for JsPsych by Hugh Rabagliati (published in 2017, so may be slightly outdated by now).

2.2.3 Audio and video recording

A recently developed plugin exists for JsPsych sound recording, though (as of July 1 2020) it isn't yet listed in JsPsych's plugin library. There are also several video and audio recording demos (based on WebRTC and web-dictaphone) on jatos.org, but we do not yet have a working experiment with which those tools are integrated. This is under development and will hopefully be available as a template in fall 2020.

2.2.4 Surveys

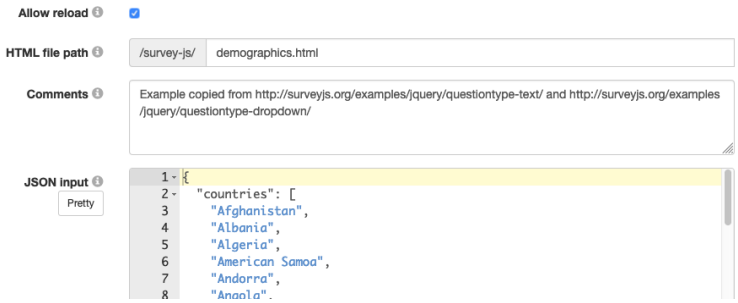
Use SurveyJS. SurveyJS, like JsPsych, is principally a Javascript library to make it easier to build something that will run in a browser - in this case, a survey. SurveyJS has a survey-building GUI on its website: you can make a free account and build and save your surveys there.[8] SurveyJS offers the option of running the surveys you build on its website, *via* its website, but since it's hosted in Estonia, we're not going to do that. Instead, copy the survey's code (available in the Survey Designer's JSON Editor; more on this in section 2.2.5), and insert it in the designated place in the Survey Template available through the AcqVA website or the "Online Research" Microsoft Teams group. **NOTE: THIS TEMPLATE ISN'T AVAILABLE YET, BUT IT WILL BE VERY SOON.**

2.2.5 A note on JSON

In the Survey Designer, there's a tab named **JSON Editor**. JSON stands for Javascript Object Notation - it's just a data format that's easy to read for humans and machines. JATOS permits us to define JSON

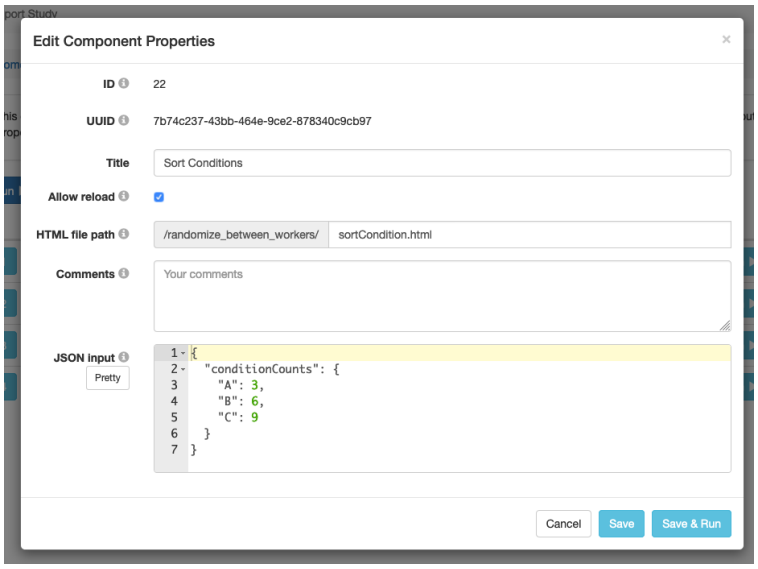
[7] Paid online research platforms with inbuilt experiment builders, such as Gorilla and FindingFive, also have great audio recording tools! See Why JATOS (and not, for instance, Gorilla)? for more information.
[8] The path to your surveys is **(Products)** > **Service** > **My Surveys**. Unfortunately, the whole website is organised in a slightly deranged way. Still worth it, but you've been warned.

variables via the **Properties** window of your study, your individual component, or your "Batch" (collection of workers; see here for more information). In the example below, one of the components is a short demographics survey. One of the questions is about the participants' nationality: it includes a drop-down menu with a list of possible choices.



Inside `demographics.html`, the variable "countries" is defined with an empty list; instead, the list of countries is supplied via the Component Properties' JSON Input. This is a really useful gimmick if there are aspects of your study that you would like to be able to adjust on the fly, without having to pause or reupload your study.

A slightly more complex example of what JSON Input can do is the Randomize Workers Across Tasks demo. This is a short script that randomly distributes workers over a number of components (specified by their position in the component list, and assigned letter names A, B and C). The script uses the numbers of participants specified for each component in the JSON Input to generate a list for the empty variable "conditions".



That variable together with its list (which looks something like ["A","A","A","B","B","B","B", etc.]) then gets placed in the study's Batch Session Data (which you can also see into and edit, under **Worker & Batch Manager**). The script then randomly selects an item off the list, starts the corresponding component, and deletes the item from the Batch Session Data. This set-up makes it possible for an experimenter to easily

adjust participant numbers between study runs, or even while a study is live via the Batch Session Data. This script, and a variant that distributes workers over components sequentially rather than randomly, is available in the template Information and Consent HTML file, on the AcqVA website and in the "Online Research" Microsoft Teams group.

2.3 Information and consent

You have a responsibility to obtain informed consent from your participants, which means your study should include a consent page with more or less the following information (copied from LSE's informed consent guidelines):

- name of the project and purpose of the research, including:
  - the researcher's institution
  - funding source (if appropriate)
- what participation will involve (e.g. if interviews are proposed, how many and how long they will be and where they will be held)
- risks to participants and benefits to participants and others
- that participants are free to withdraw at any time without prejudice and without providing a reason
- that if the participant withdraws, they should be given the option to have any information they have provided thus far to be removed from the study
- what usage will be made of the data: during the research itself (where data will be stored and who will have access to it); after the research (if/how the data will be published); whether the (anonymised) data will be used in possible future research or archived to meet funder requirements, and/or how and when the data will be destroyed
- strategies for maintaining confidentiality and anonymity
- contact details:
  - contact details for researchers
  - who to contact should they have a complaint
  - how to request a copy of the data about themselves

Don't forget to clearly warn people about whether they can or cannot reload a component! In the Properties window of each component is a tick box **Allow Reload**. JATOS will also show participants a warning window if they try to refresh or close an experiment tab that does not permit reloading.

2.3.1 Quit and Delete Everything button

With JATOS, it's easy to provide your participants with the option of quitting anytime during the study and deleting all data collected from them up to that point. Simply include the following code snippet in the Javascript portion of your HTML file: `jatos.addAbortButton()`. Here is an example of how you can customise your Quit button.

3 Getting NSD approval for your study

If all you want is instructions for how to fill out your NSD notification form (meldeskjema) such that they will approve your online study, skip ahead to What do I write in my NSD meldeskjema?. For some background on what we are and aren't allowed to do with personal data, and why, keep reading.

3.1 NSD and GDPR compliance

3.1.1 What is NSD?

NSD is the Norsk Senter for Forskningsdata, which you already know if you've ever tried to get ethical approval for a study involving the processing of personal data. You can check here whether you have to notify NSD of your study, but if you're reading this because you want to do web-based research with humans, you can safely assume that you have to.
Since the introduction of the General Data Protection Regulation (GDPR) throughout the EU and in many of its satellite nations including Norway, the rules around the processing of personal data have become much stricter.

NSD is the Data Protection Official for all universities and most hospitals and research institutes in Norway. Where, for example, the University of Edinburgh conducts its ethics and data protection evaluations in-house, UiT has commissioned NSD to evaluate projects *according to UiT's own regulations*. This distinction matters because NSD can *advise against* running a study (for instance, when you want to run it online using participants recruited from platforms such as Mechanical Turk or Prolific - more on that later), but they can't *forbid* it. When that happens, you can give NSD a letter from your institution's **personvernombud** giving you permission to override NSD's decision, and in fact that is what NSD will advise you to do.[9]
However, if UiT is then audited by the Norwegian Data Protection Authority, and it turns out you *did* breach data protection legislation, the consequences are harsh: a massive fine for UiT, the loss of all data collected under the faulty protocol, retracted papers, nullified PhD's, etc. We don't want that.

3.1.2 Data controller vs. data processor

Why is it so difficult to get NSD approval for an online study using crowdsourced participants? Because such studies very often make use of what is called a **third party data processor**. I'll explain this using my own study: a self-paced reading experiment for which I wanted to recruit participants via Prolific, a company in the UK. (I might have run the experiment via Ibex Farm, which is hosted in the US. Other popular options include Gorilla (hosted in Ireland), FindingFive (until very recently hosted only in the US), and Pavlovia (hosted in the UK.))
NSD considers participant recruitment platforms like Prolific, and experiment hosting platforms like Ibex Farm, to be **data processors**, as opposed to **data controllers**. These two examples illustrate the difference: UiT uses Azure Microsoft to deliver its cloud computing solutions; Microsoft is a data processor, processing data on UiT's behalf and not doing anything else with that data for their own nefarious purposes (they say). For this, UiT has a data processing agreement with Microsoft. On the other hand, when someone at UiT wanted to do a survey via YouGov, and NSD had to evaluate that study, it was decided that YouGov was a data controller: they 'owned' the panel that the survey was put to. Whatever personal data YouGov processed of its platform members, it did so *on its own behalf*. The survey was therefore a kind of transaction between two separate data controllers (UiT and YouGov).

[9] Our personvernombud will then say that he is merely an advisor to "the administration", and that he has no authority to give such permission. Only "the administration" has that authority. We are currently working with the administration to solve this once and for all.

In evaluating my study, NSD said that if I ran an experiment with participants recruited via Prolific, this platform would act as a data processor, and UiT would therefore have to draw up a data processing agreement with them. When asked for such an agreement, Prolific said that "there is no processing of personal data from Prolific to Researchers, as any data we share with you is anonymised." (Gorilla makes a similar claim on their website's data privacy page.) They, understandably, will not agree to drawing up a data processing agreement outside their Terms of Service.

3.1.3 A note on anonymised versus de-identified data

This is diving pretty deep, but in case you're interested: most people say data is 'anonymised' when they mean 'de-identified'. The International Association of Privacy Professionals (IAPP) defines anonymisation as "The act of permanently and completely removing personal identifiers from data, such as converting personally identifiable information into aggregated data. Anonymized data is data that can no longer be associated with an individual in any manner. Once this data is stripped of personally identifying elements, those elements can never be re-associated with the data or the underlying individual. [...] De-identification involves the removal of personally identifying information in order to protect personal privacy. This may mean that the personally identifying information may be able to be re-associated with the data at a later time."
Though this can feel like pedantry in a lot of cases, it comes into play when scrutinising claims from self-proclaimed GDPR-compliant companies such as Prolific that their data exchange with clients is anonymised.

3.1.4 What if I want to use crowdsourced participants?

As of now, you can recruit participants for your study on Prolific (the jury's still out on platforms outside Europe, such as Mechanical Turk in the US and OMI in Russia), *so long as you collect no personal data from your participants whatsoever.* That includes an e-mail address, an ID that can be used to find someone's name on a list, or a collection of triangulating background information. Examples of "safe" studies are reaction time studies, norming and rating studies, and surveys (provided the demographic questions don't get too specific). Double-check with me if you're unsure whether your study collects any personal data.

If in the past you have gotten NSD approval for studies that recruit participants via e.g. Facebook, and you want to recruit participants the same way for your online study, **nothing changes for you.** (Of course, you still need to report your online study to them.)

3.1.5 How do I reward participants of online studies?

We're not sure yet. All ideas are welcome! If you're not using a service like Prolific, it can be tricky to figure out how to reward your participants without collecting very sensitive data such as bank account numbers. Depending on the region your participants are from, electronic gift cards from Amazon or some large chain store may be appropriate.

3.2 What do I write in my NSD meldeskjema?

We suggest something like the following paragraph:

> "The study will be run over the internet through any type of browser on the participants own devices (PC, mobile, tablet etc.). The study is hosted on a server in UiT's cloud computing platform (delivered by Azure Microsoft, with which UiT has a data processing agreement). All participant and study data is

directly collected, processed, and stored on this secure server. The study will not collect HTTP header field data of any kind, including browser type and IP address. Studies are created from freely available, Creative Commons-licensed Javascript libraries and experiment-building software."

We recommend that you do **not** mention the software we use for running online studies on our UiT server by name. The one time we tried, NSD got confused and thought JATOS was a third party data processor. It is not: it is an open-source Javascript library with a pretty user interface that we downloaded off the internet and used to make our own server hospitable to non-programmers. In the unlikely event that you do need to explain more about data privacy and ethics with JATOS, I refer you to this webpage.