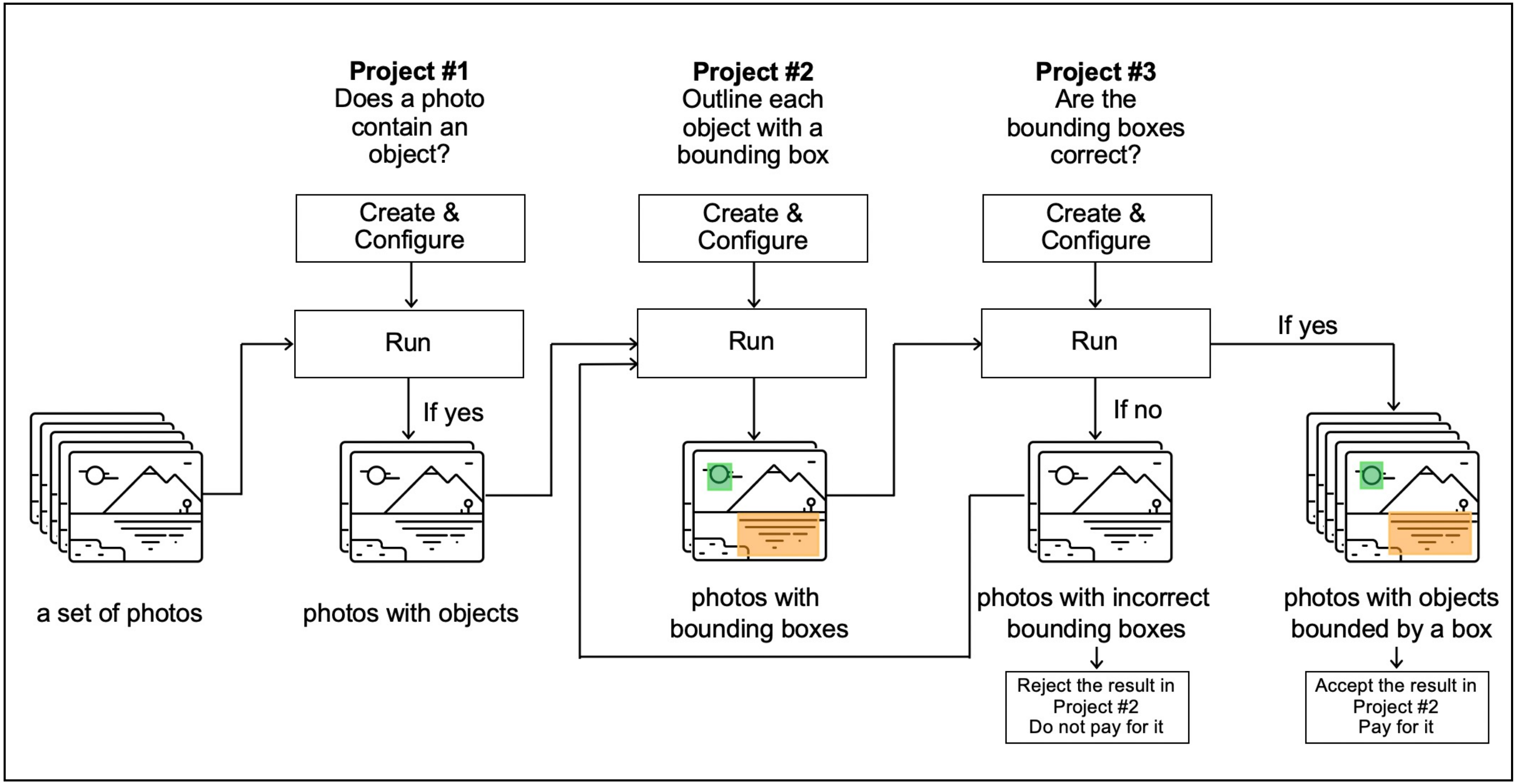
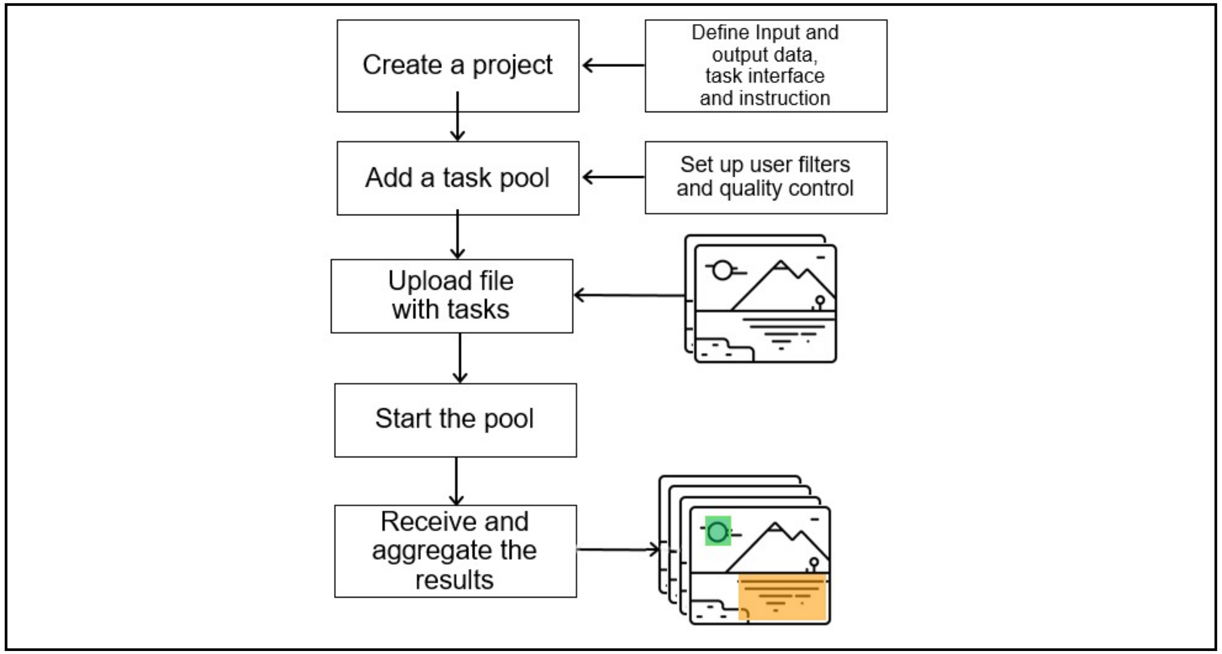
Efficient Data Annotation for Self-Driving Cars via Crowdsourcing on a Large-Scale

Suggested pipeline



Project #1
Does a photo contain an object?
Project #2
Outline each object with a bounding box
Project #3
Are the bounding boxes correct?
Create & Configure
Create & Configure
Create & Configure
Run
Run
Run
If yes
If no
If yes
a set of photos
photos with objects
photos with bounding boxes
photos with incorrect bounding boxes
photos with objects bounded by a box
Reject the result in Project #2 Do not pay for it
Accept the result in Project #2 Pay for it

Project creation. Main steps



Create a project

Define Input and output data, task interface and instruction

Add a task pool

Set up user filters and quality control

Upload file with tasks

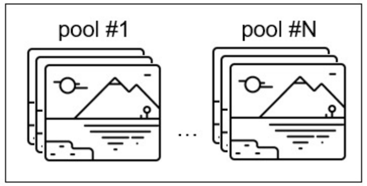
Start the pool

Receive and aggregate the results

Key types of instances in Yandex.Toloka

Project

- Defines the structure of tasks
- Defines how to perform them

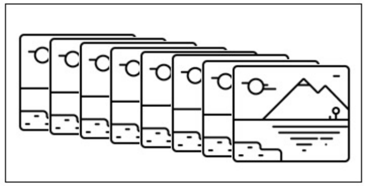


Configure in a project:

- Input and output data types
- Task interface
- Task instruction

Pool

- Is a batch of tasks
- Defines access of performers



Configure in a pool:

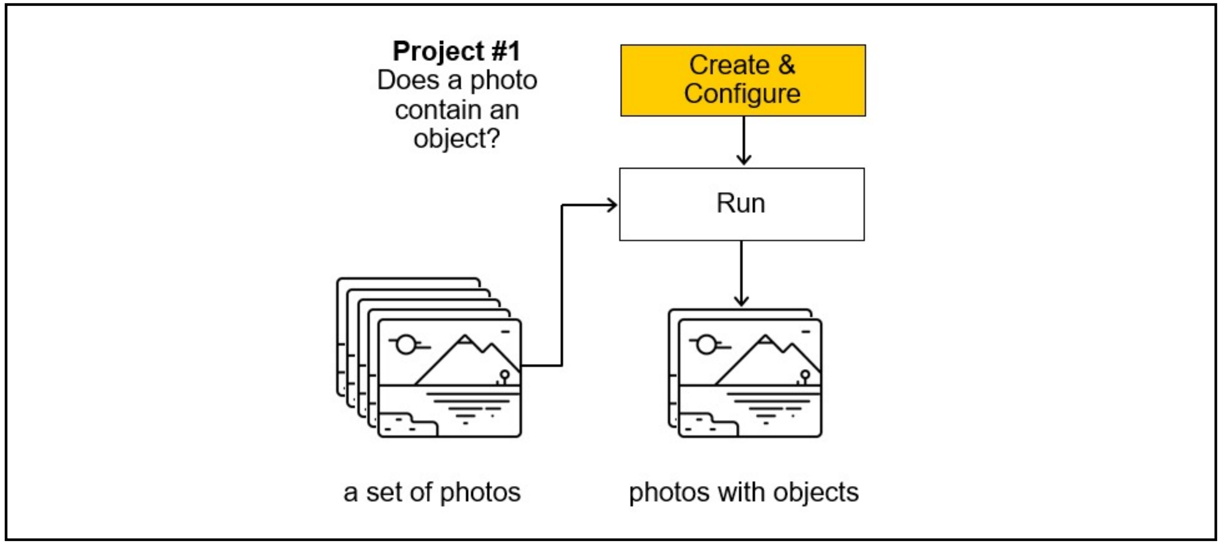
- Performer filters
- Quality control mechanisms
- Overlap settings

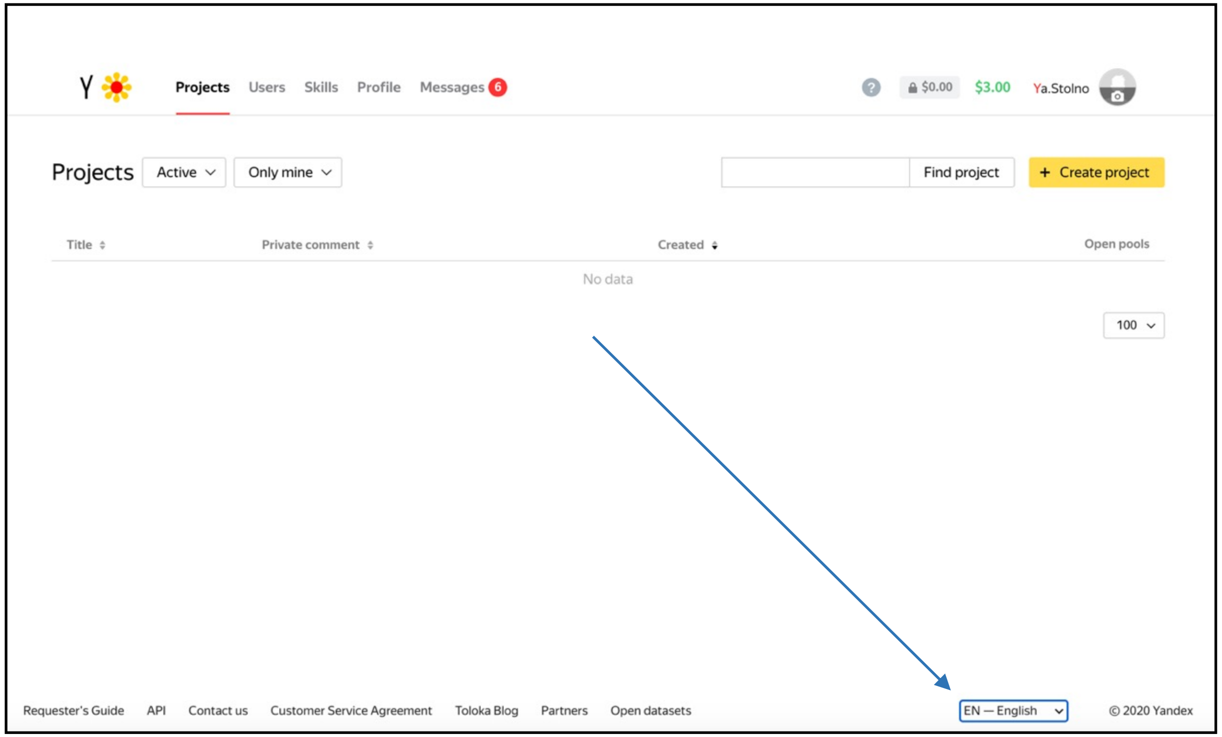
Task

- A particular input data
- Results for it from performers

Project #1

Does a photo contain an object?



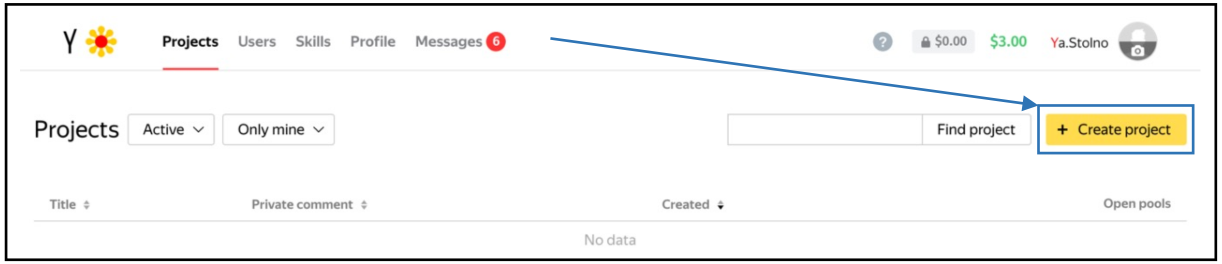


Important: Before you start using Toloka, make sure that the English language is selected

Project creation

1. Go to https://toloka.ai/

2. Click the button **+ Create project**



3. Choose the **Image classification** template

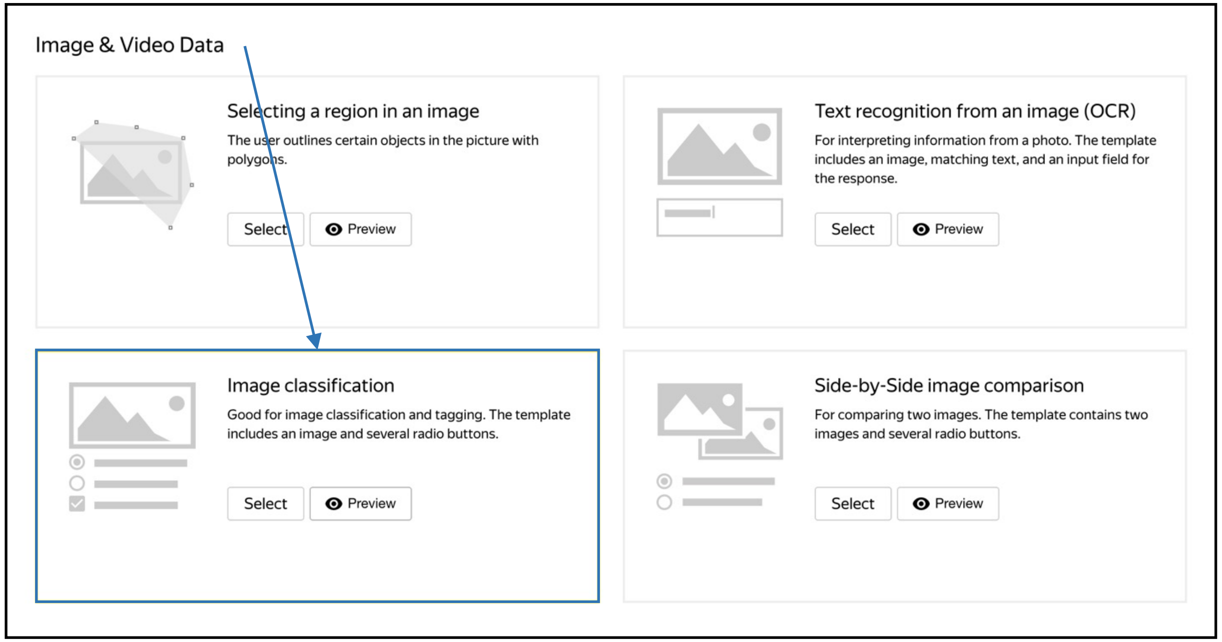


Image & Video Data

Selecting a region in an image

The user outlines certain objects in the picture with polygons.

Text recognition from an image (OCR)

For interpreting information from a photo. The template includes an image, matching text, and an input field for the response.

Image classification

Good for image classification and tagging. The template includes an image and several radio buttons.

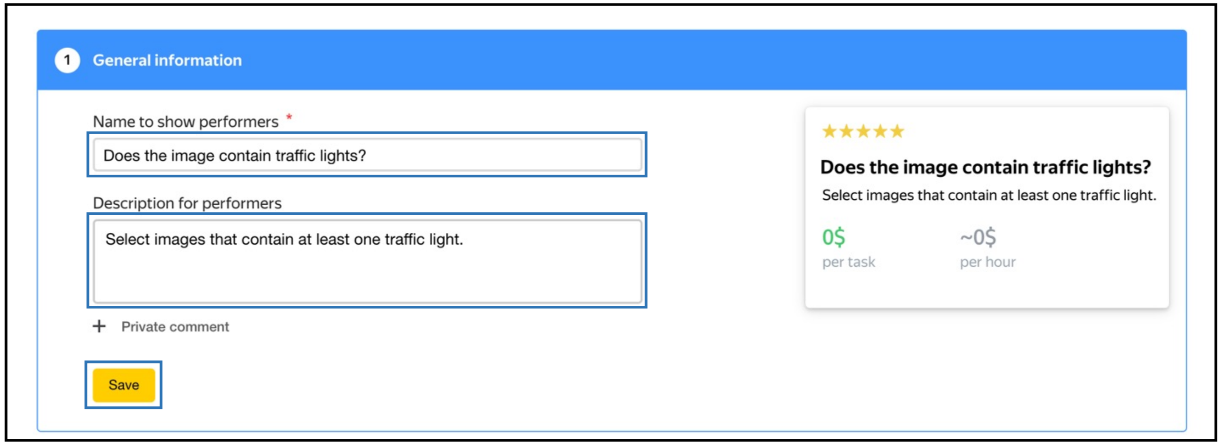
Side-by-Side image comparison

For comparing two images. The template contains two images and several radio buttons.

4. Enter a clear project name and description

Important: It will be visible for real people

5. Click **Save** button to save the general information



Name to show performers *

Does the image contain traffic lights?

Description for performers

Select images that contain at least one traffic light.

★★★★★

Does the image contain traffic lights?

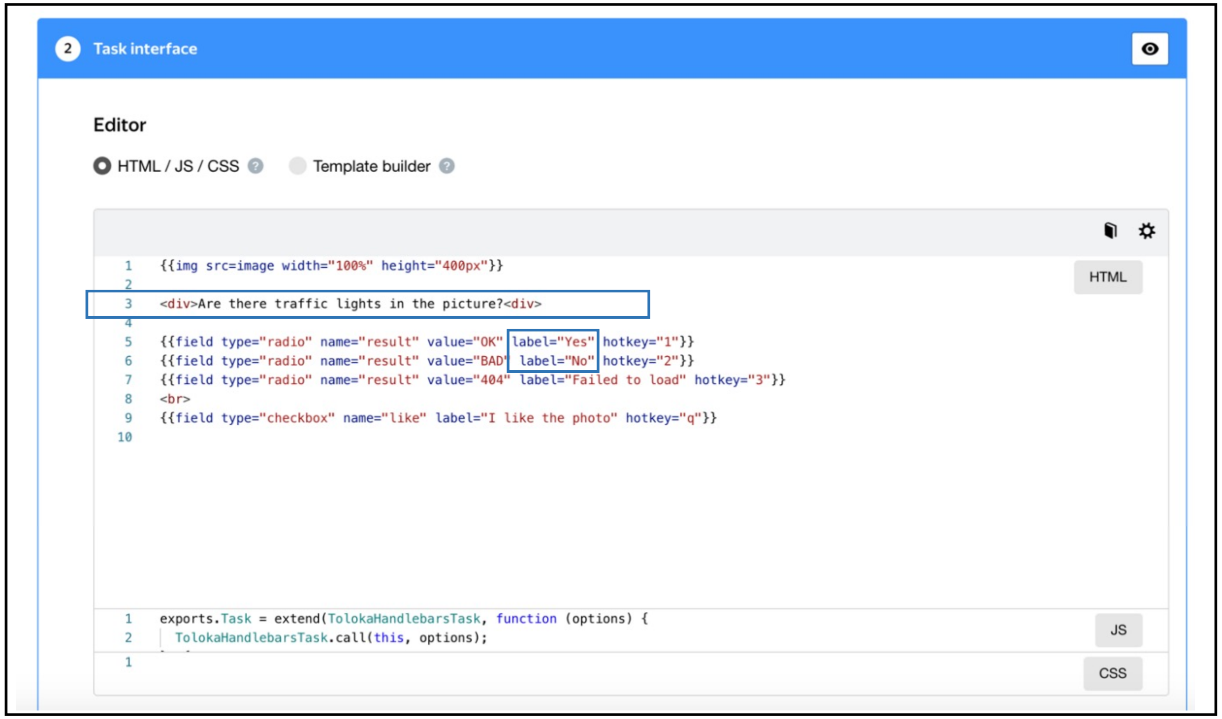
Select images that contain at least one traffic light.

0$ per task

~0$ per hour

6. Create the task interface in the HTML block

- Delete the line with the checkbox component:
  {{field type="checkbox" name="like" label="I like the photo" hotkey="q"}}
- Add a question: does the image include a certain object? Example:
  <div>Are there traffic lights in the picture?<div>
- Replace **"label"** with your response options. Example:
  {{field type="radio" name="result" value="OK" label="Yes" hotkey="1"}}
  {{field type="radio" name="result" value="BAD" label="No" hotkey="2"}}
  {{field type="radio" name="result" value="404" label="Failed to load" hotkey="3"}}



● HTML / JS / CSS ○ Template builder

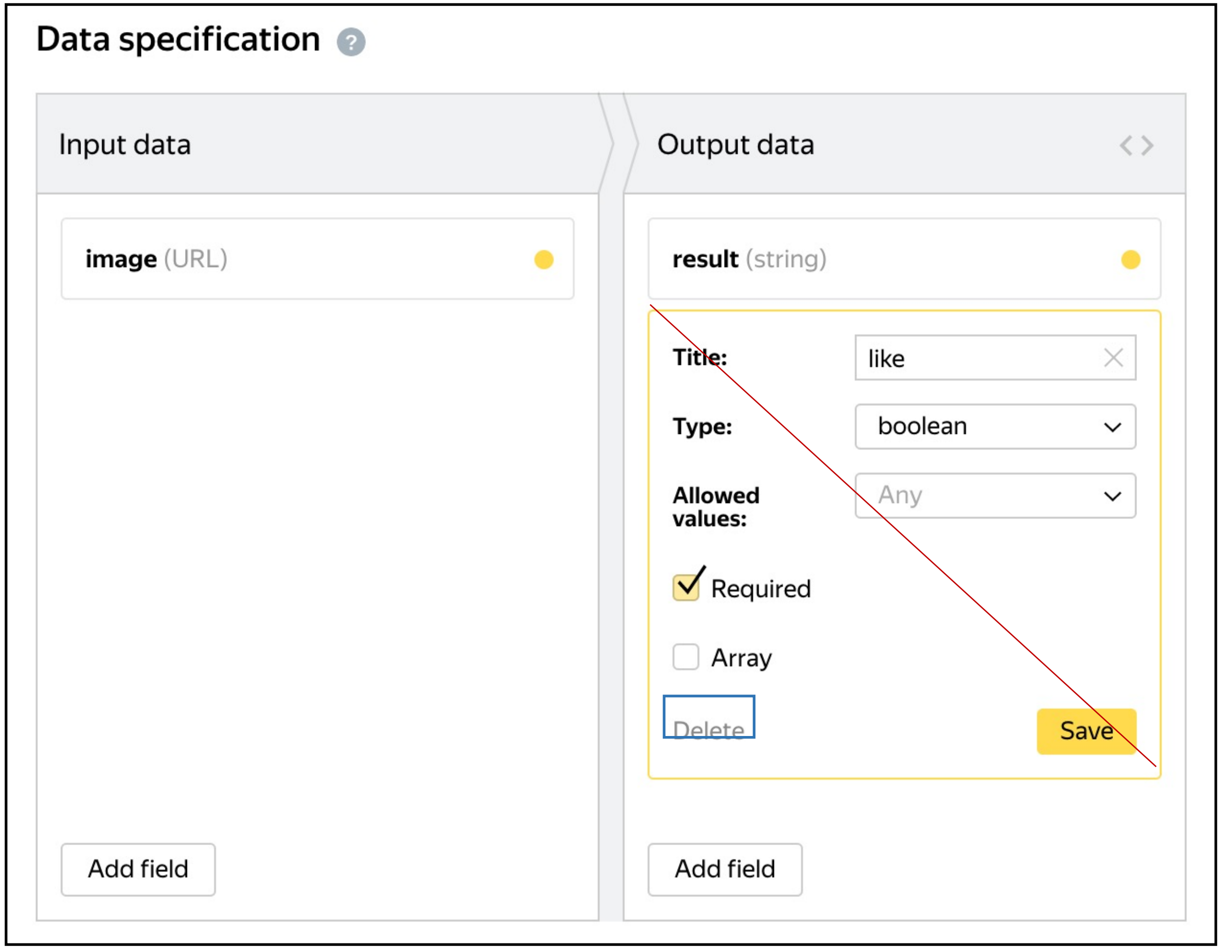
```
{{img src=image width="100%" height="400px"}}

<div>Are there traffic lights in the picture?<div>

{{field type="radio" name="result" value="OK" label="Yes" hotkey="1"}}
{{field type="radio" name="result" value="BAD" label="No" hotkey="2"}}
{{field type="radio" name="result" value="404" label="Failed to load" hotkey="3"}}
<br>
{{field type="checkbox" name="like" label="I like the photo" hotkey="q"}}
```

```
exports.Task = extend(TolokaHandlebarsTask, function (options) {
  TolokaHandlebarsTask.call(this, options);
```

7. Leave the CSS and JavaScript blocks unchanged

8. Define parameters for the input and output data:

- The **“image”** input data field with the link type will be used to pass the image links to the performers. You will be able to upload the file with links to the pool later.
- The **"result"** field will be used to receive performer's responses.
- The **"like"** field in the template is used to pass the response to the question "Do you like the photo?". Our project doesn't require this checkbox, so you don't need an output field for it. Let's remove it



Data specification

Input data

image (URL)

Add field

Output data

result (string)

Title: like

Type: boolean

Allowed values: Any

☑ Required

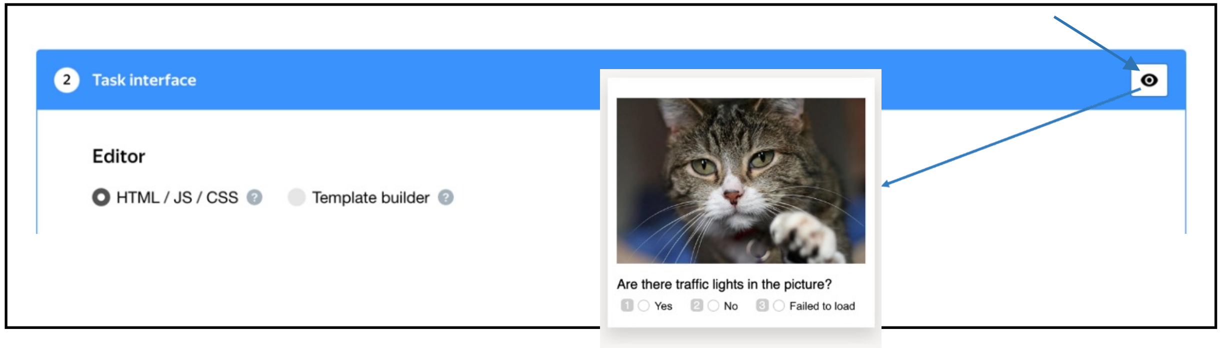
□ Array

Delete

Save

Add field

9. Click the **Preview** button to see the performer's view of the task. You will see one task with standard picture on the page. You will set the number of tasks per page when configuring a pool



2 Task interface

Editor

● HTML / JS / CSS ○ Template builder

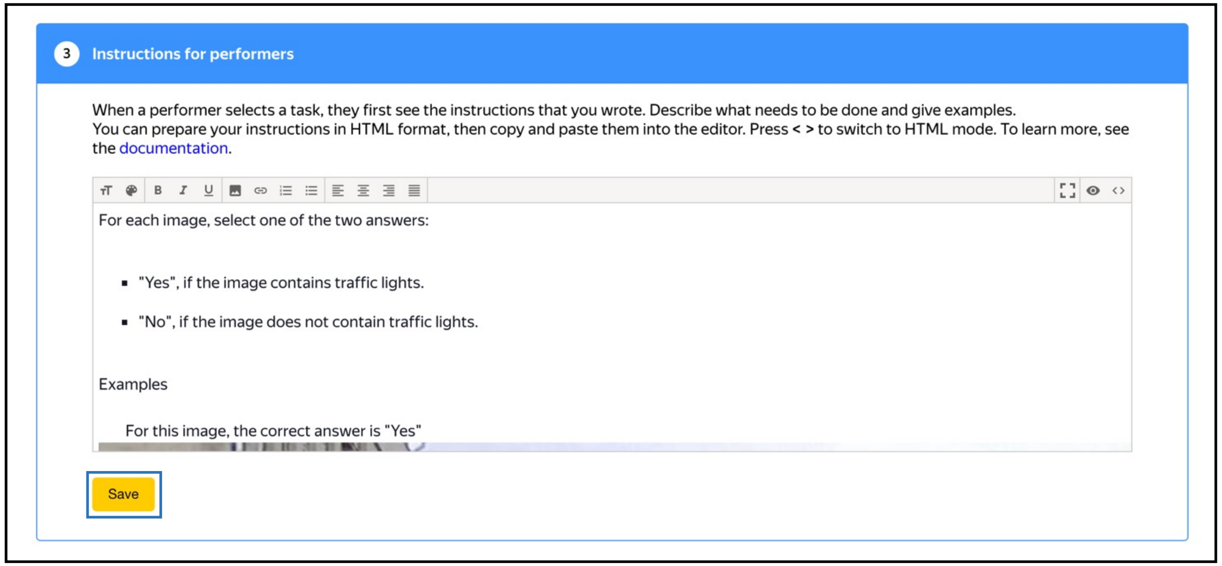
Are there traffic lights in the picture?

1 ○ Yes 2 ○ No 3 ○ Failed to load

10. Select the radio buttons in the preview and make sure that the task can be completed

11. Click **Save** button to save the task interface

12. Write short and simple instructions. **To include an image in the instruction just paste the link from the dataset provided by pressing button.** Example:



3 Instructions for performers

When a performer selects a task, they first see the instructions that you wrote. Describe what needs to be done and give examples.
You can prepare your instructions in HTML format, then copy and paste them into the editor. Press < > to switch to HTML mode. To learn more, see the documentation.

For each image, select one of the two answers:

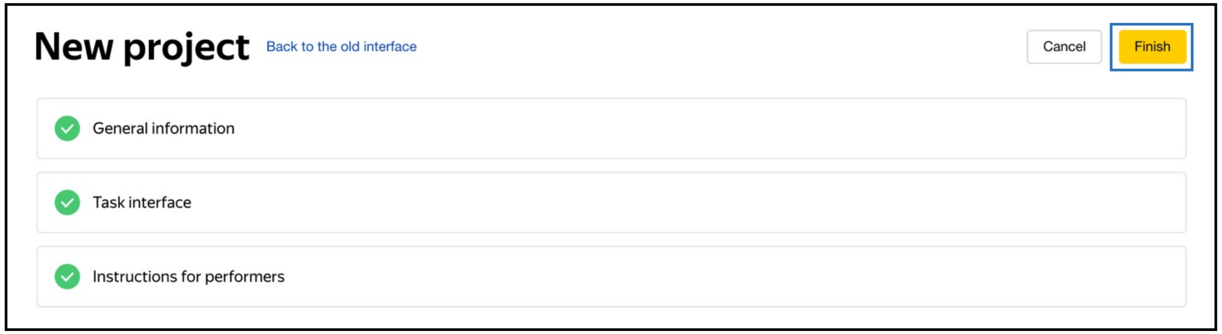
- "Yes", if the image contains traffic lights.
- "No", if the image does not contain traffic lights.

Examples

For this image, the correct answer is "Yes"

13. Click **Save** button to save the instruction

14. Click **Finish** button to save the project



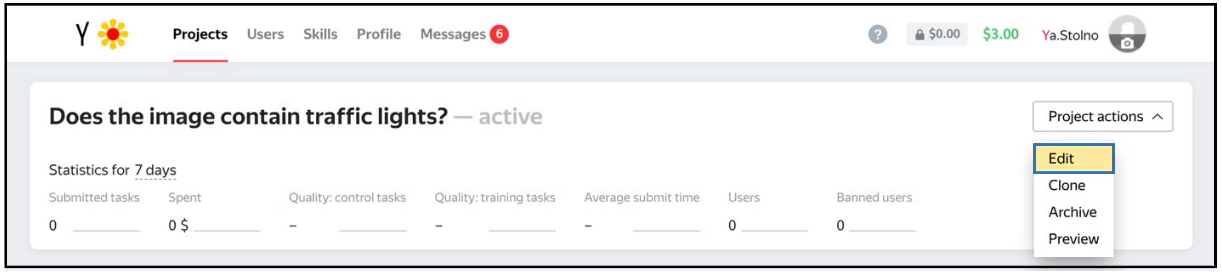
New project

✓ General information

✓ Task interface

✓ Instructions for performers

Note. *To edit project parameters, click the button in the list of projects or **Project actions** → **Edit** on the project page*



Does the image contain traffic lights? — active

Statistics for 7 days

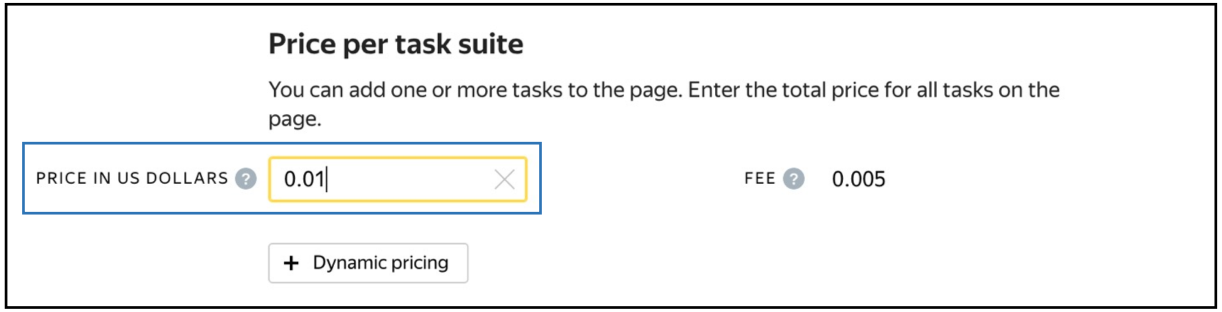
Submitted tasks	Spent	Quality: control tasks	Quality: training tasks	Average submit time	Users	Banned users
0	0 $	–	–	–	0	0

Pool creation

1. Click **Add pool**

2. Give the pool any convenient name and description. You are the only one who can see them

3. Specify the pool parameters:
   - Set the price per task page (for example, $0.01)



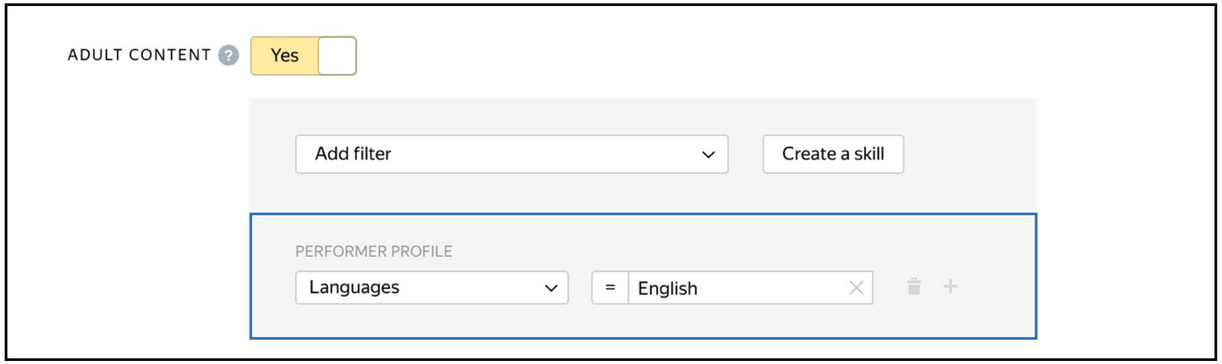
Price per task suite

You can add one or more tasks to the page. Enter the total price for all tasks on the page.

PRICE IN US DOLLARS 0.01

FEE 0.005

4. Set up user filters.
   - Select English-speaking performers using the **Language = English** filter



ADULT CONTENT Yes

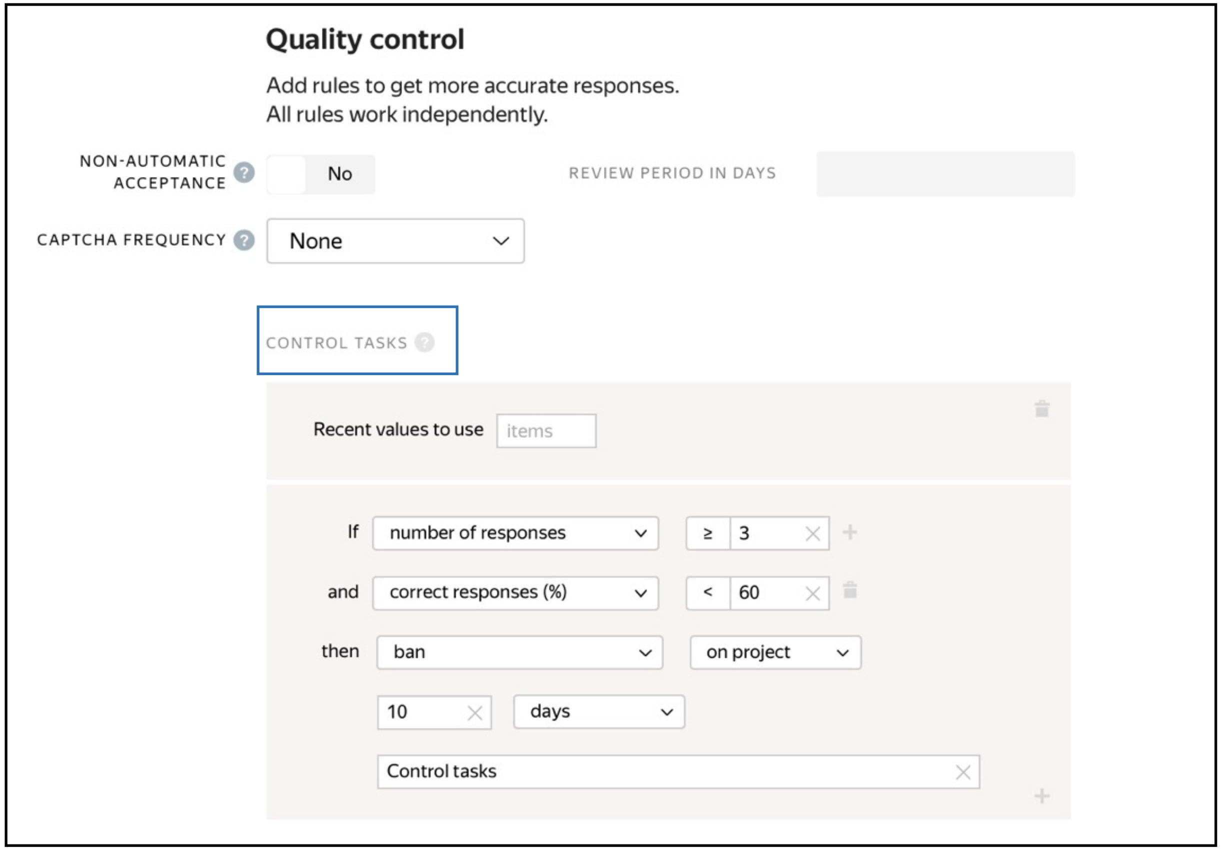
PERFORMER PROFILE

Languages = English

5. Set up quality control: Control tasks. Ban performers who give incorrect responses to control tasks. Example:

   This rule will be triggered when the performer completes 3 control tasks in the pool. If the performer gives at least 3 responses to the control tasks and the percentage of correct responses is less than 60%, they lose access to the project for 10 days. If the percentage of correct responses is over 60%, the performer can pass to the next task page. The rule will be triggered after the next control task.

   Optionally, add other quality control rules



Quality control

Add rules to get more accurate responses.
All rules work independently.

NON-AUTOMATIC ACCEPTANCE: No

REVIEW PERIOD IN DAYS

CAPTCHA FREQUENCY: None

CONTROL TASKS

Recent values to use: items

If number of responses ≥ 3

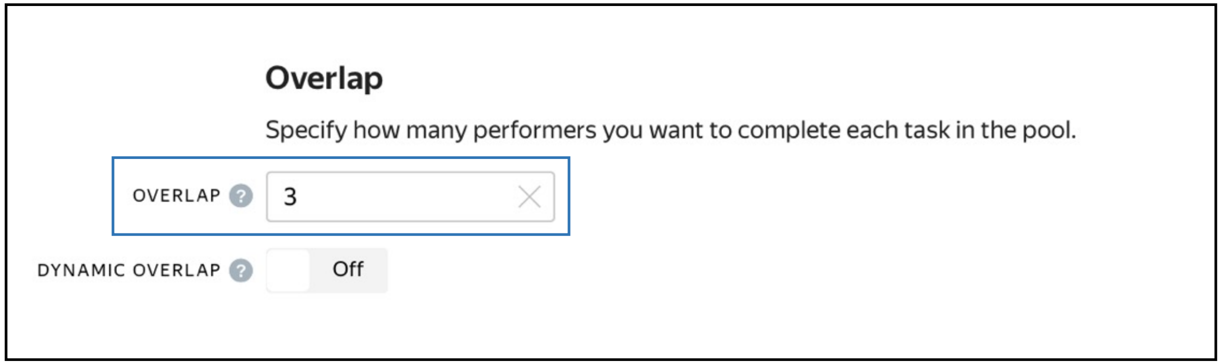
and correct responses (%) < 60

then ban on project

10 days

Control tasks

6. Overlap. This is the number of users who will complete the same task. For example, 3



Overlap

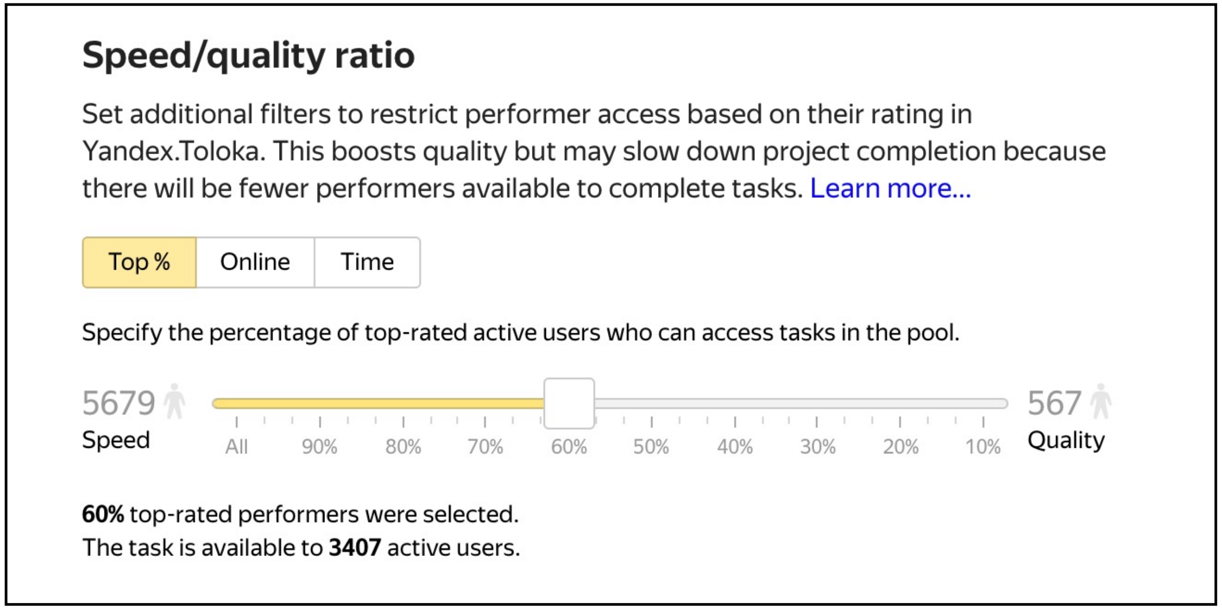
Specify how many performers you want to complete each task in the pool.

OVERLAP: 3

DYNAMIC OVERLAP: Off

7. Optionally, specify the percentage of top-rated performers in the Speed / Quality ratio

Important: This can slow down pool completion



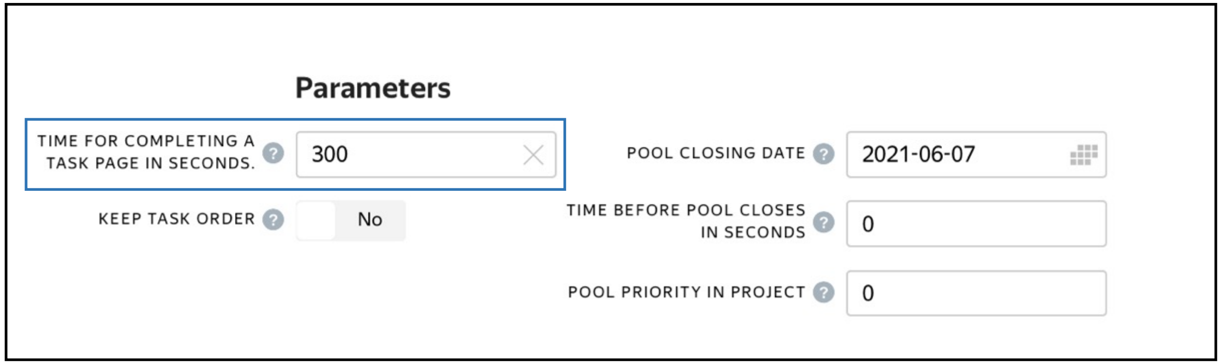
Speed/quality ratio

Set additional filters to restrict performer access based on their rating in Yandex.Toloka. This boosts quality but may slow down project completion because there will be fewer performers available to complete tasks. Learn more...

Specify the percentage of top-rated active users who can access tasks in the pool.

60% top-rated performers were selected.
The task is available to **3407** active users.

8. Time allowed for completing a task page (for example, 300 seconds)



Parameters

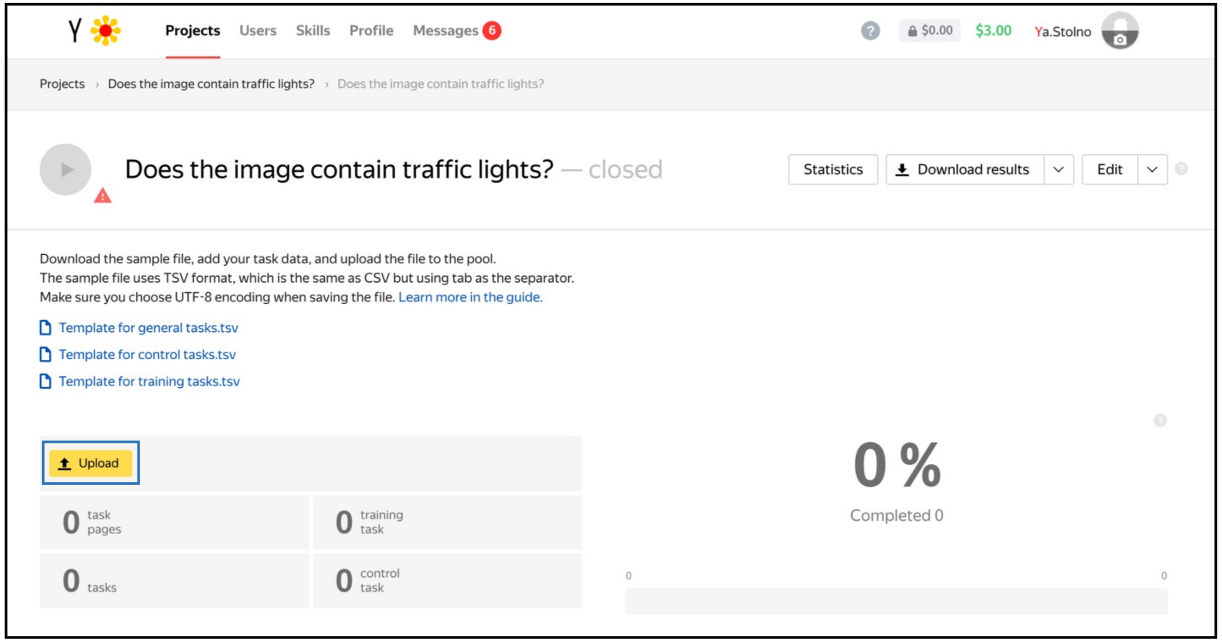
TIME FOR COMPLETING A TASK PAGE IN SECONDS.	300
KEEP TASK ORDER	No
POOL CLOSING DATE	2021-06-07
TIME BEFORE POOL CLOSES IN SECONDS	0
POOL PRIORITY IN PROJECT	0

9. Save the pool

Preparing and uploading a file with tasks

1. Download TSV-file with images through the link that you were provided with at the beginning of the practice session

2. Upload pool tasks from this file



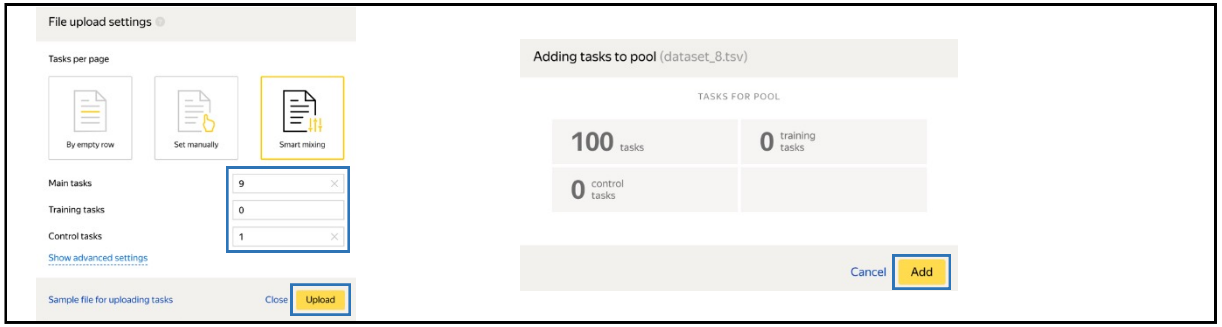
Does the image contain traffic lights? — closed

Download the sample file, add your task data, and upload the file to the pool.
The sample file uses TSV format, which is the same as CSV but using tab as the separator.
Make sure you choose UTF-8 encoding when saving the file. Learn more in the guide.

- Template for general tasks.tsv
- Template for control tasks.tsv
- Template for training tasks.tsv

Important: If you changed the name of the input field, change it in the file as well

Select Smart mixing and specify the number of tasks per page.
For example: 9 main tasks and 1 control task



Main tasks 9
Training tasks 0
Control tasks 1

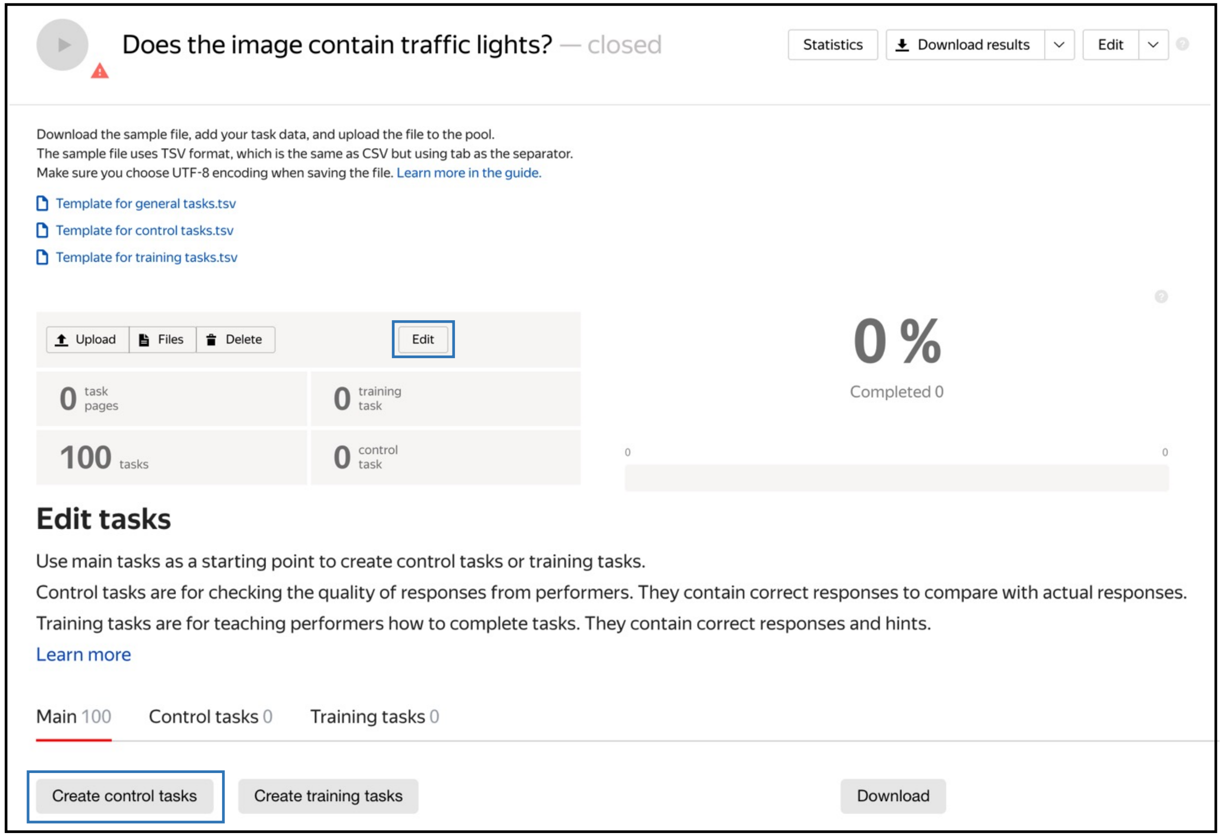
Adding tasks to pool (dataset_8.tsv)

100 tasks
0 training tasks
0 control tasks

3. Create control tasks

Note. *Control tasks are tasks with the correct response known in advance. They are used to track the performer's quality of responses. The performer's response is compared to the response you provided. If they match, it means the performer answered correctly.*

- Click **Edit → Create control tasks**



Download the sample file, add your task data, and upload the file to the pool.
The sample file uses TSV format, which is the same as CSV but using tab as the separator.
Make sure you choose UTF-8 encoding when saving the file. Learn more in the guide.

Template for general tasks.tsv
Template for control tasks.tsv
Template for training tasks.tsv

Edit tasks

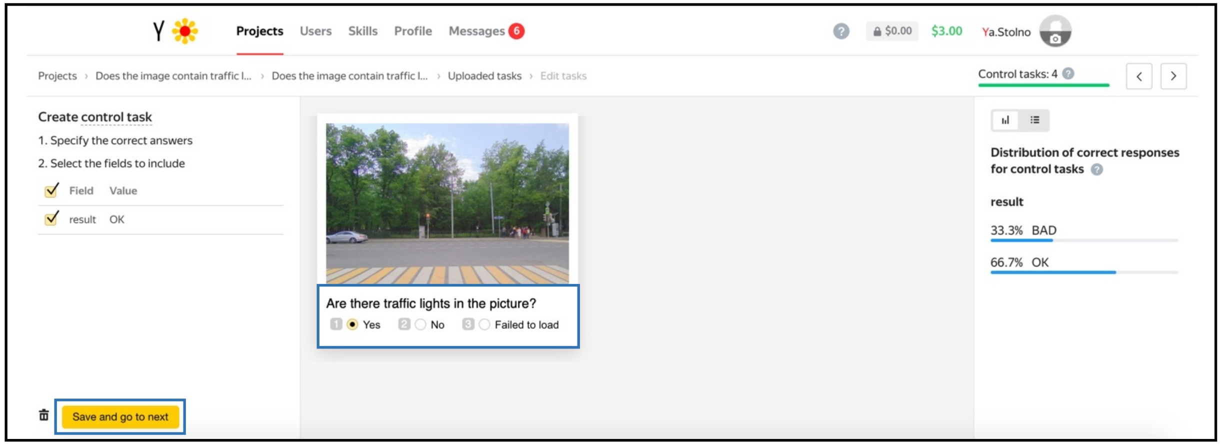
Use main tasks as a starting point to create control tasks or training tasks.

Control tasks are for checking the quality of responses from performers. They contain correct responses to compare with actual responses.

Training tasks are for teaching performers how to complete tasks. They contain correct responses and hints.

Learn more

- Check the "result" output field that is used to match the user response to the control one, select the response and click **Save and go to next**



Projects
Users
Skills
Profile
Messages 6
$0.00
$3.00
Ya.Stolno
Projects › Does the image contain traffic l... › Does the image contain traffic l... › Uploaded tasks › Edit tasks
Control tasks: 4

Create control task

1. Specify the correct answers
2. Select the fields to include

☑	Field	Value
☑	result	OK

Are there traffic lights in the picture?

1 ● Yes 2 ○ No 3 ○ Failed to load

Distribution of correct responses for control tasks

result

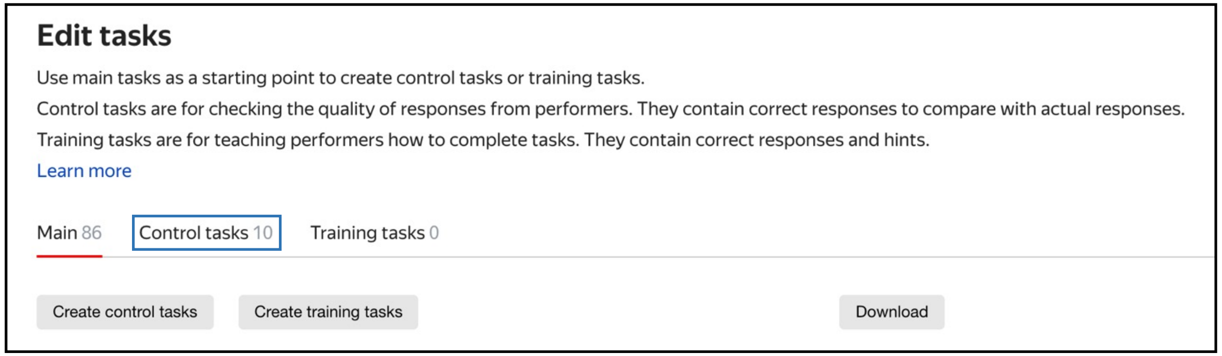
33.3% BAD

66.7% OK

Important: This can slow down pool completion

Enter correct responses for 10 tasks. In small pools, control tasks should account for less than 10% of all tasks.

Tip. *Make sure to include different variations of correct responses in equal amounts. Open* ***the Control → Distribution of known answers for control tasks*** *tab*



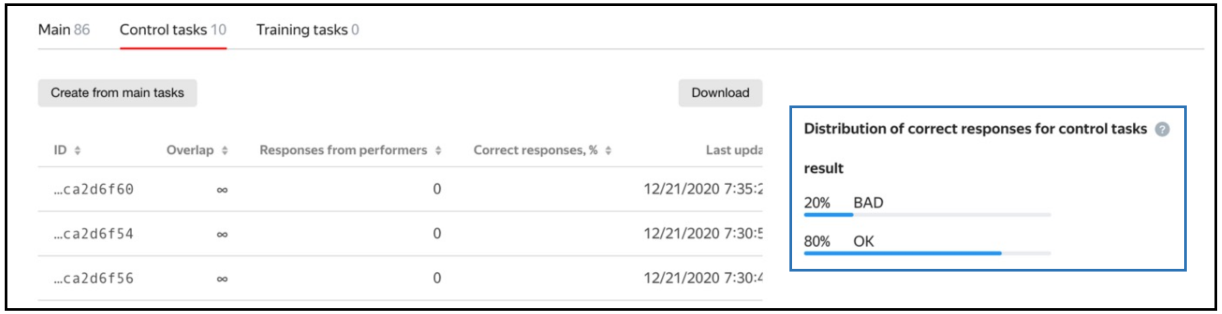
Edit tasks

Use main tasks as a starting point to create control tasks or training tasks.

Control tasks are for checking the quality of responses from performers. They contain correct responses to compare with actual responses.

Training tasks are for teaching performers how to complete tasks. They contain correct responses and hints.

Learn more



ID	Overlap	Responses from performers	Correct responses, %	Last upda
…ca2d6f60	∞	0		12/21/2020 7:35:2
…ca2d6f54	∞	0		12/21/2020 7:30:5
…ca2d6f56	∞	0		12/21/2020 7:30:4

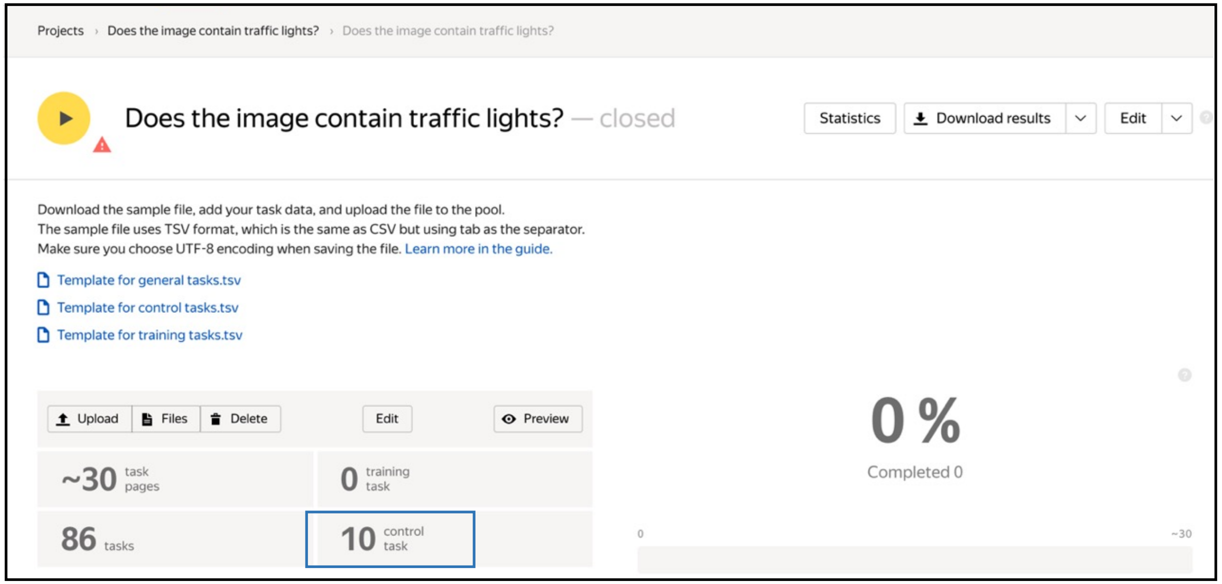
Distribution of correct responses for control tasks

result

20%	BAD
80%	OK

- Save the markup and check the number of control tasks

4. Start the pool



Projects › Does the image contain traffic lights? › Does the image contain traffic lights?

Does the image contain traffic lights? — closed

Statistics
Download results
Edit

Download the sample file, add your task data, and upload the file to the pool.
The sample file uses TSV format, which is the same as CSV but using tab as the separator.
Make sure you choose UTF-8 encoding when saving the file. Learn more in the guide.

Template for general tasks.tsv
Template for control tasks.tsv
Template for training tasks.tsv

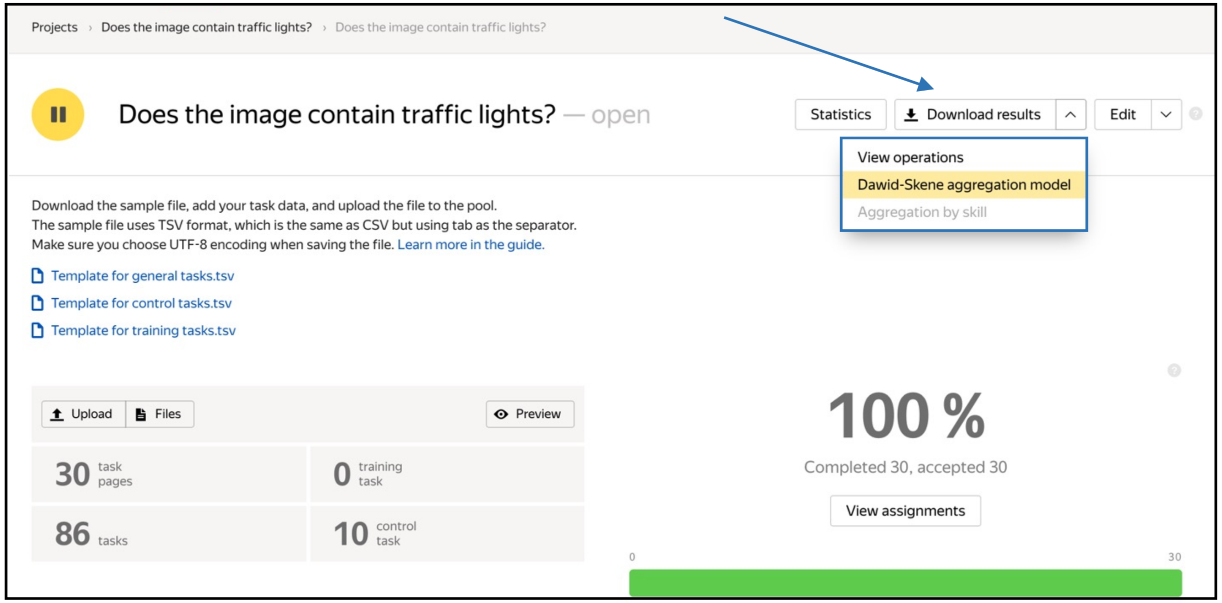
Important: Remember that real Toloka performers will complete the tasks. Double check that everything is correct with configuration of your project before you start the pool

Receiving responses

Disclaimer: Aggregation takes from 5 to 20 minutes. During this time, you can start configuring your next project. Refresh the Operations page to check progress

1. Wait until the pool is completed. Refresh the pool page to check progress

2. Click the arrow next to **the Download results** button and run aggregation using the Dawid-Skene model



Projects › Does the image contain traffic lights? › Does the image contain traffic lights?

Does the image contain traffic lights? — open

Download the sample file, add your task data, and upload the file to the pool.
The sample file uses TSV format, which is the same as CSV but using tab as the separator.
Make sure you choose UTF-8 encoding when saving the file. Learn more in the guide.

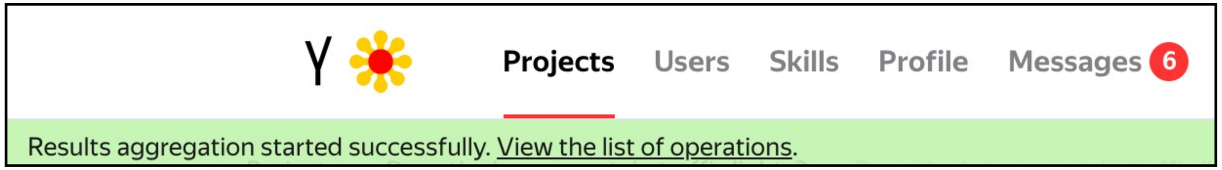
- Template for general tasks.tsv
- Template for control tasks.tsv
- Template for training tasks.tsv

30 task pages	0 training task
86 tasks	10 control task

100 %

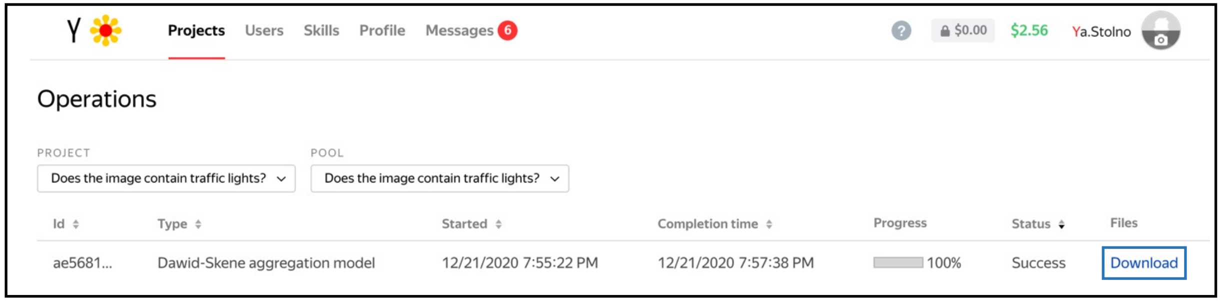
Completed 30, accepted 30

3. Go to the operations list and wait until aggregation finishes.
Note. *Aggregation takes from 5 to 20 minutes. During this time, you can start working on your next project. Refresh the Operations page to check progress*



Results aggregation started successfully. View the list of operations.

4. When aggregation is complete, download the TSV file with the results



Operations

PROJECT: Does the image contain traffic lights?

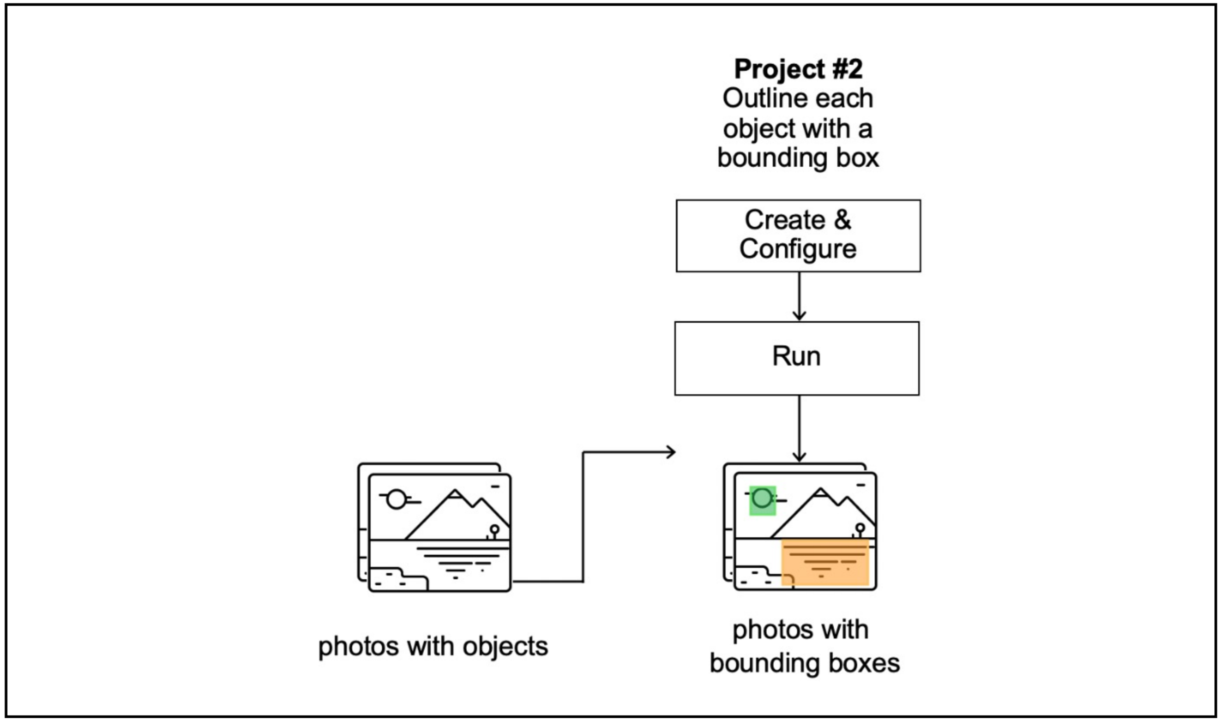
POOL: Does the image contain traffic lights?

Id	Type	Started	Completion time	Progress	Status	Files
ae5681...	Dawid-Skene aggregation model	12/21/2020 7:55:22 PM	12/21/2020 7:57:38 PM	100%	Success	Download

5. Use this file to prepare data for Project #2

Project #2

Outline each object with a bounding box



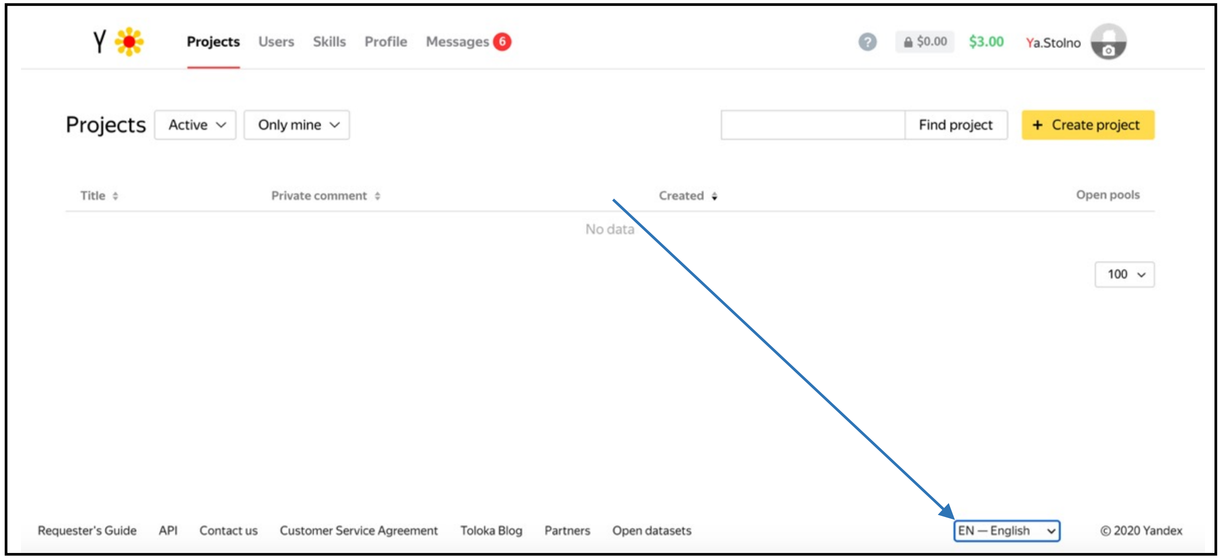
Project #2
Outline each object with a bounding box

Create & Configure

Run

photos with objects

photos with bounding boxes



Important: If you just start using Toloka, make sure that the English language is selected

Project creation

1. Click the **+ Create project** button and choose the **Selecting a region in an image** template

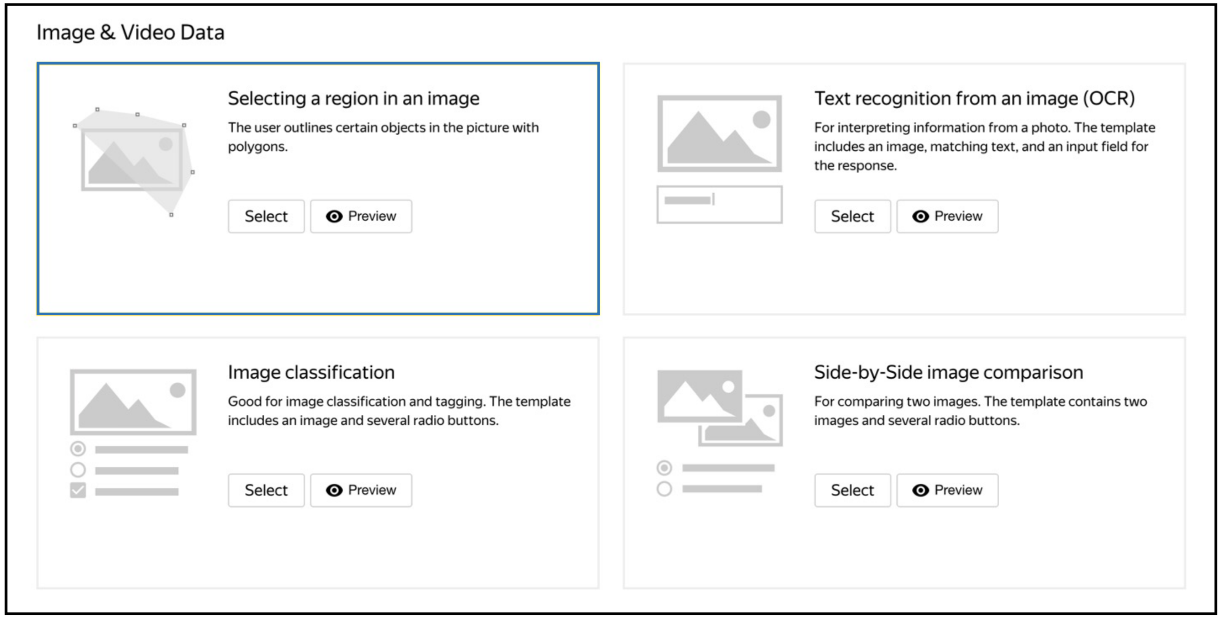


Image & Video Data

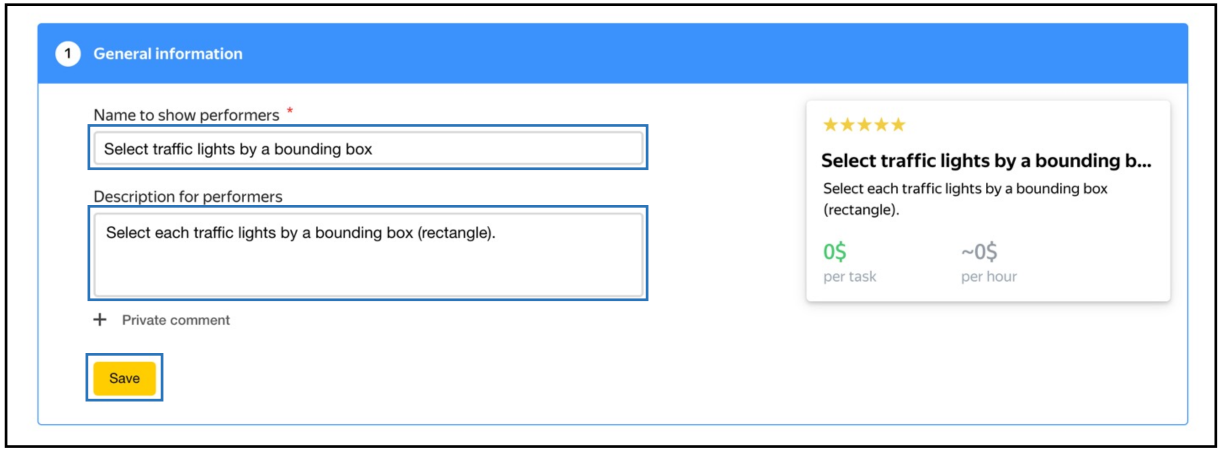
Selecting a region in an image
The user outlines certain objects in the picture with polygons.

Text recognition from an image (OCR)
For interpreting information from a photo. The template includes an image, matching text, and an input field for the response.

Image classification
Good for image classification and tagging. The template includes an image and several radio buttons.

Side-by-Side image comparison
For comparing two images. The template contains two images and several radio buttons.

2. Enter a clear project name and description. It will be shown to performers



1 General information

Name to show performers *
Select traffic lights by a bounding box

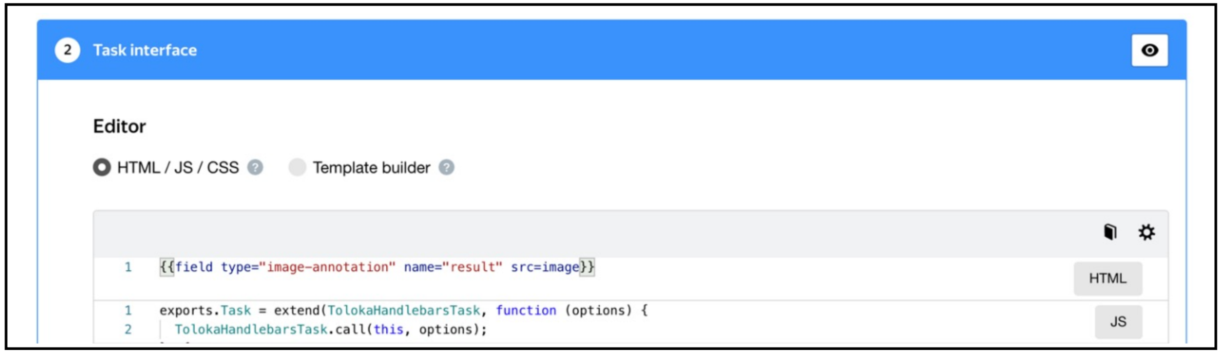
Description for performers
Select each traffic lights by a bounding box (rectangle).

Select traffic lights by a bounding b...
Select each traffic lights by a bounding box (rectangle).
0$ per task
~0$ per hour

3. Click **Save** button to save the general information

4. Create the task interface

   The image area selection editor is automatically included in the template. This means that the performer will see the image and can select an area with the editor tools. You don't have to change anything in the interface. Leave HTML and JS box unchanged


2 Task interface

Editor

● HTML / JS / CSS ○ Template builder

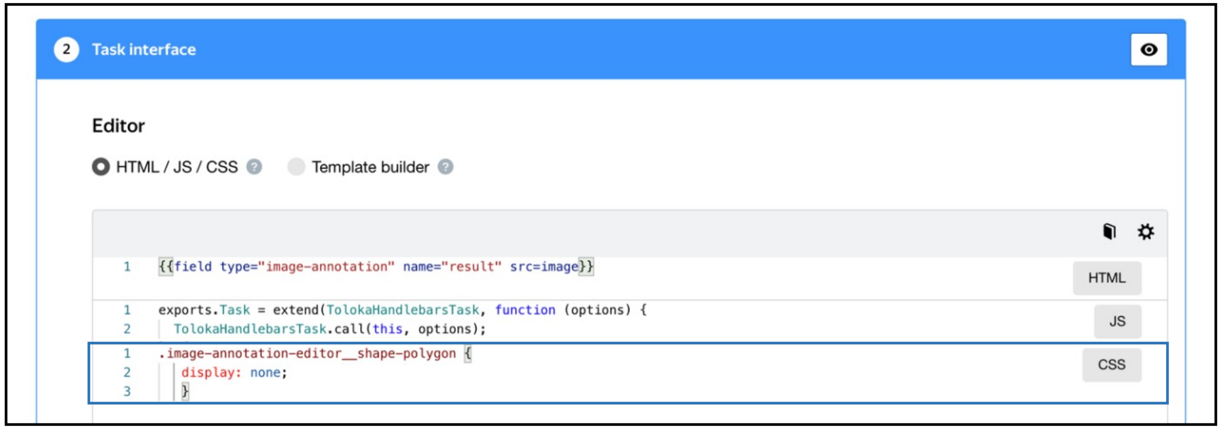
```
{{field type="image-annotation" name="result" src=image}}
```

```
exports.Task = extend(TolokaHandlebarsTask, function (options) {
  TolokaHandlebarsTask.call(this, options);
```

---

5. By default, rectangle tool is hidden in the task interface. To show it, remove the existing code from the CSS field. To hide polygon tool, add the following code:

   .image-annotation-editor shape-polygon { display: none;}


2 Task interface

Editor

● HTML / JS / CSS ○ Template builder

```
{{field type="image-annotation" name="result" src=image}}
```

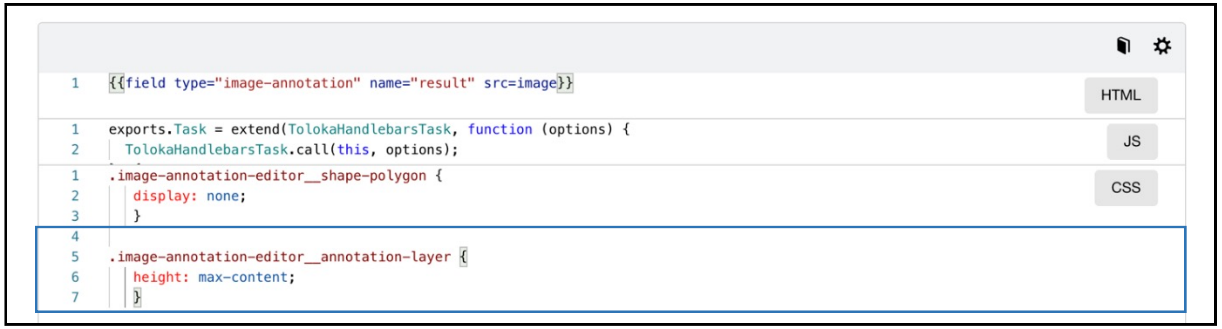
```
exports.Task = extend(TolokaHandlebarsTask, function (options) {
  TolokaHandlebarsTask.call(this, options);
```

```
.image-annotation-editor__shape-polygon {
  display: none;
}
```

---

6. Add the following code to set the height of the image markup interface to match the image size:

   .image-annotation-editor annotation-layer { height: max-content;}


```
{{field type="image-annotation" name="result" src=image}}
```

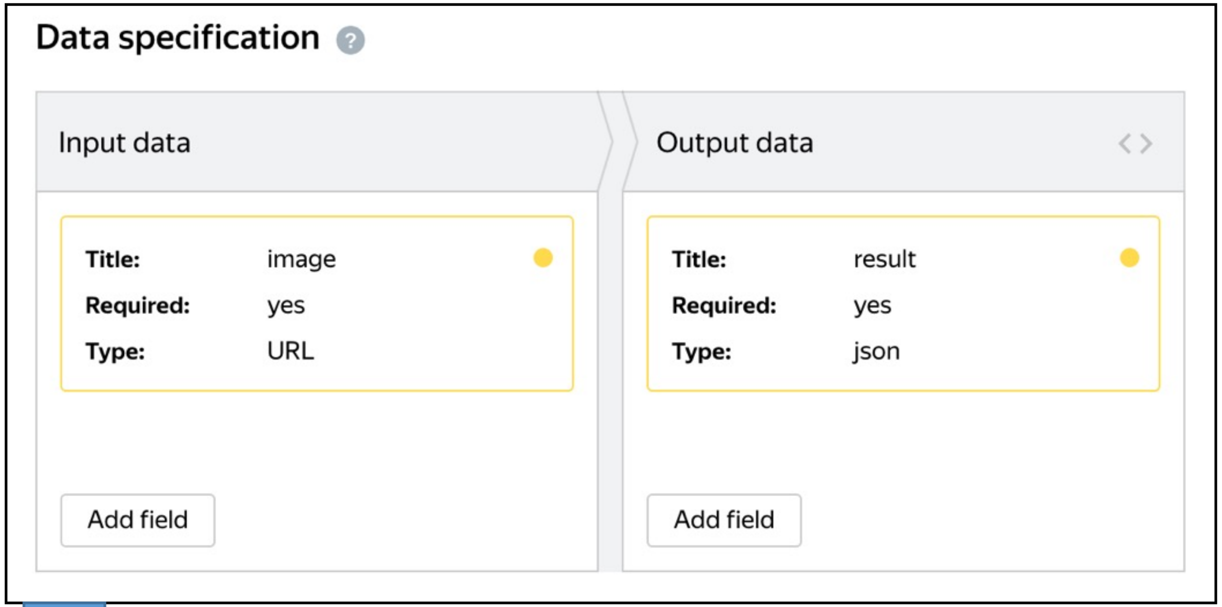
```
exports.Task = extend(TolokaHandlebarsTask, function (options) {
  TolokaHandlebarsTask.call(this, options);
```

```
.image-annotation-editor__shape-polygon {
  display: none;
}

.image-annotation-editor__annotation-layer {
  height: max-content;
}
```

7. Define parameters for the input and output data:

- The **“image”** input data field with the link type will be used to pass the image links to the performers.

The **"result"** field with the "json" type is used to record the coordinates of the points marked by the performer



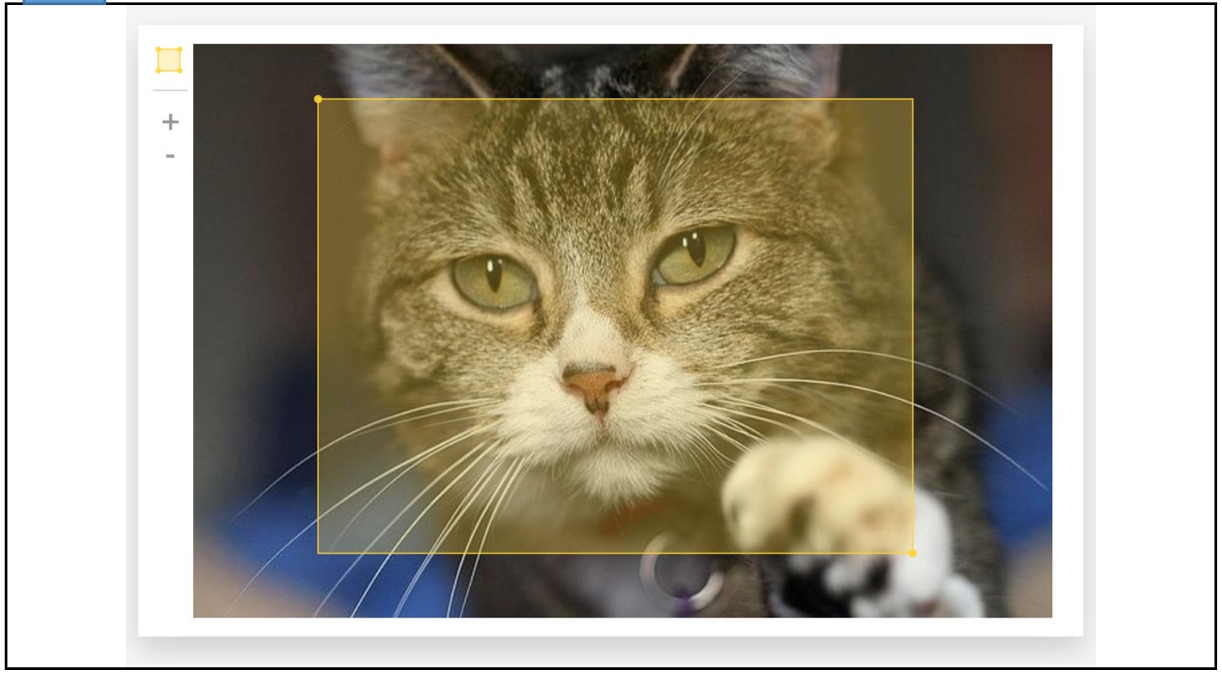
Data specification

Input data	
Title:	image
Required:	yes
Type:	URL

Output data	
Title:	result
Required:	yes
Type:	json

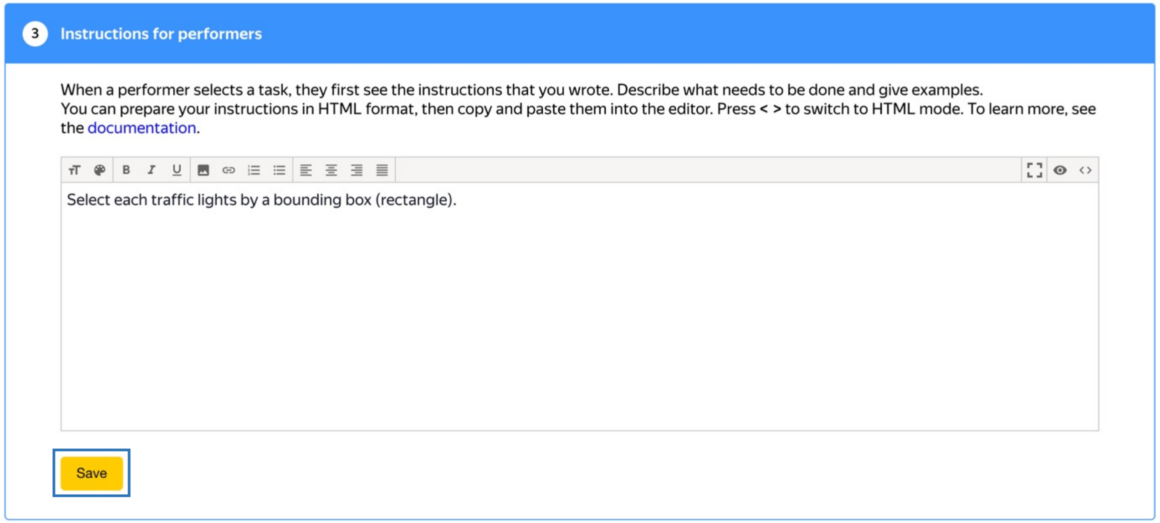
8. Click the Preview button to see the performer's view of the task. Make sure it can be submitted

You will see four tasks with standard pictures on the page. You will set the number of tasks per page when configuring a pool



9. Click **Save** button to save the task interface

10. Write short and simple instructions



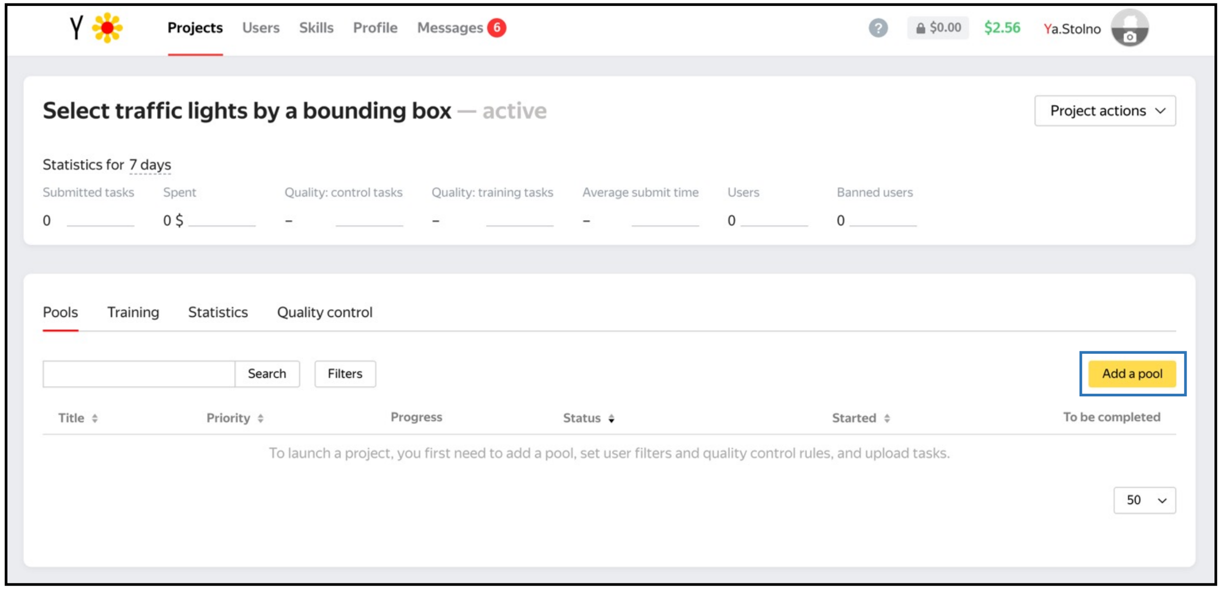
When a performer selects a task, they first see the instructions that you wrote. Describe what needs to be done and give examples.
You can prepare your instructions in HTML format, then copy and paste them into the editor. Press **< >** to switch to HTML mode. To learn more, see the documentation.

Select each traffic lights by a bounding box (rectangle).

11. Click Finish button to save the project

Pool creation

1. Click **Add pool**


Select traffic lights by a bounding box — active

Statistics for 7 days

Submitted tasks	Spent	Quality: control tasks	Quality: training tasks	Average submit time	Users	Banned users
0	0 $	–	–	–	0	0

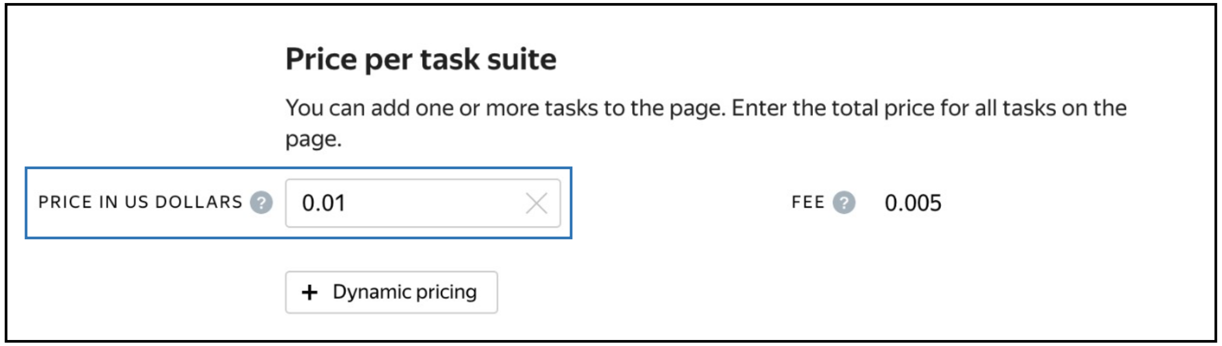
Title	Priority	Progress	Status	Started	To be completed

To launch a project, you first need to add a pool, set user filters and quality control rules, and upload tasks.

2. Give the pool any convenient name and description. You are the only one who can see them

3. Specify the pool parameters:

   - Price per task page (for example, $0.01)


Price per task suite

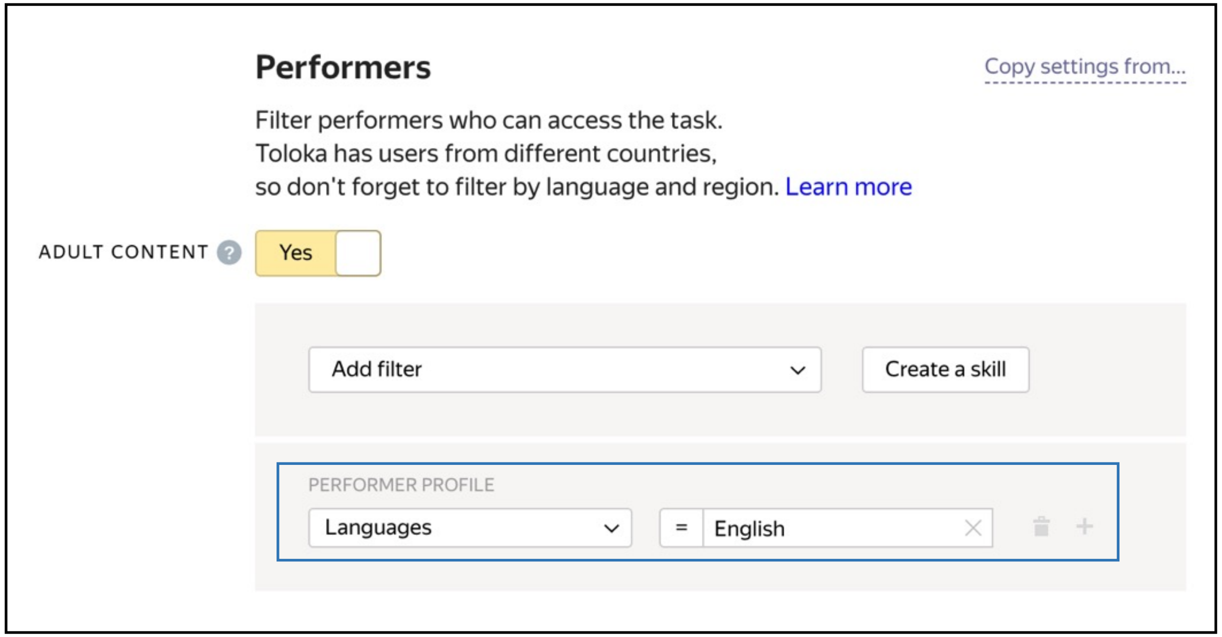
You can add one or more tasks to the page. Enter the total price for all tasks on the page.

PRICE IN US DOLLARS: 0.01

FEE: 0.005

4. Set up user filters

   - Select English-speaking performers using the Language = English filter



Performers

Copy settings from...

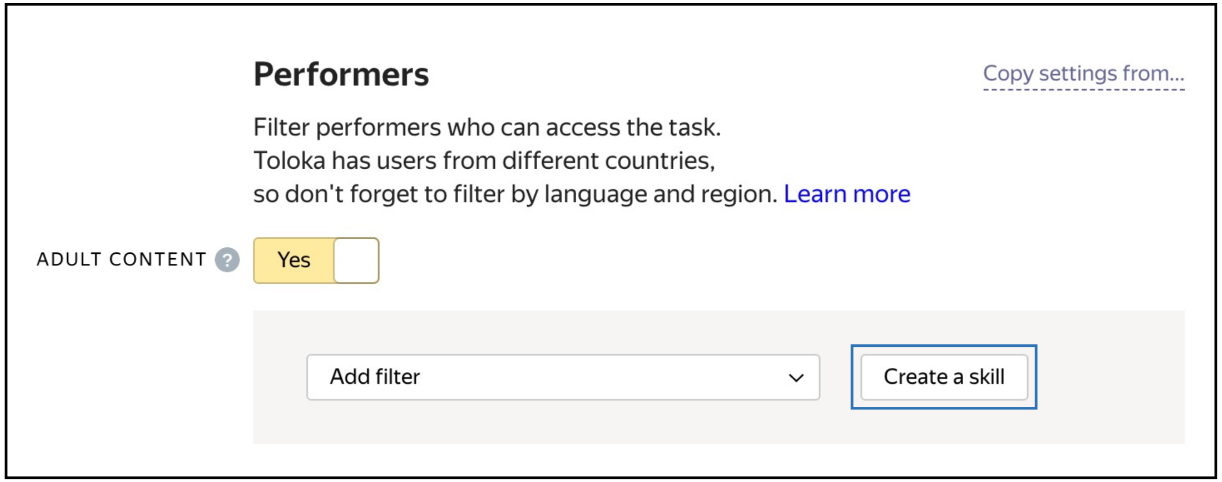
Filter performers who can access the task.
Toloka has users from different countries,
so don't forget to filter by language and region. Learn more

ADULT CONTENT ? Yes

PERFORMER PROFILE

Languages = English

- Create the **"worked on_bounding"** skill that will be assigned to users after they complete the pool tasks. You will use this skill to prevent these users from checking tasks in the next project. Click **Create skill**



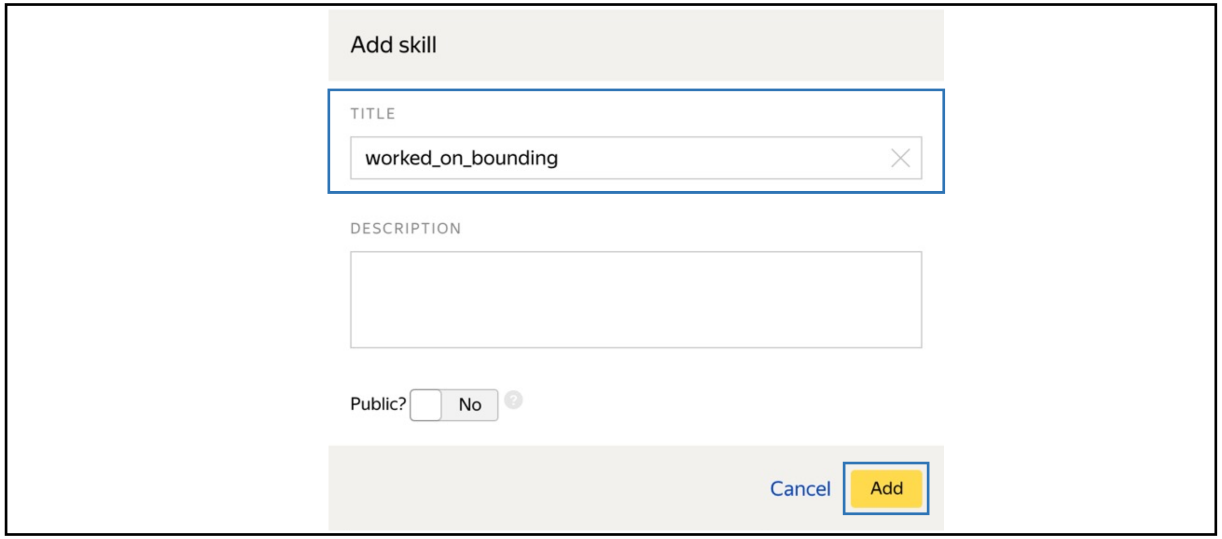
Performers

Copy settings from...

Filter performers who can access the task.
Toloka has users from different countries,
so don't forget to filter by language and region. Learn more

ADULT CONTENT ? Yes

- Enter the skill name and add a description if desired. You are the only one who will see it. Leave the skill **private**, as it is by default and click **Add**



Add skill

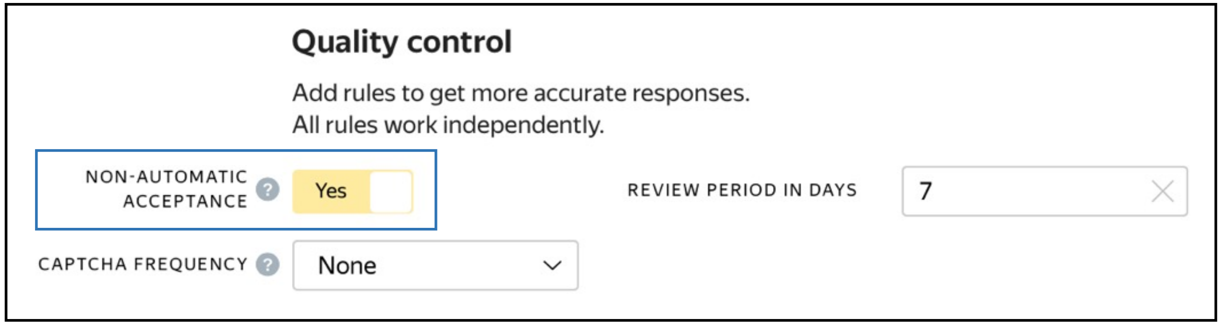
TITLE

worked_on_bounding

DESCRIPTION

Public? No

5. Turn on the **Non-automatic acceptance** option and enter the number of days for checking in the **Deadline** field (for example, 7)



Quality control

Add rules to get more accurate responses.
All rules work independently.

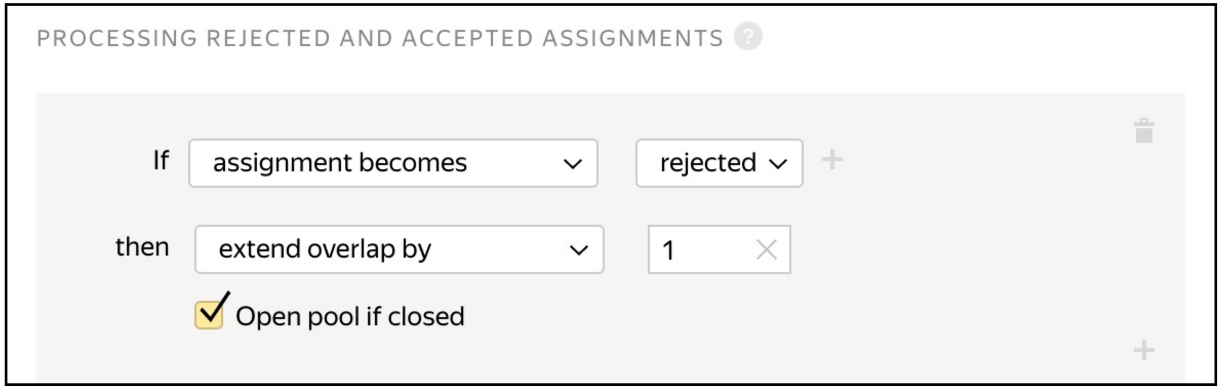
NON-AUTOMATIC ACCEPTANCE: Yes

REVIEW PERIOD IN DAYS: 7

CAPTCHA FREQUENCY: None

6. Set up quality control:

   - Resend the rejected tasks for completion. Add the Recompletion of rejected assignments:



PROCESSING REJECTED AND ACCEPTED ASSIGNMENTS

If assignment becomes rejected

then extend overlap by 1

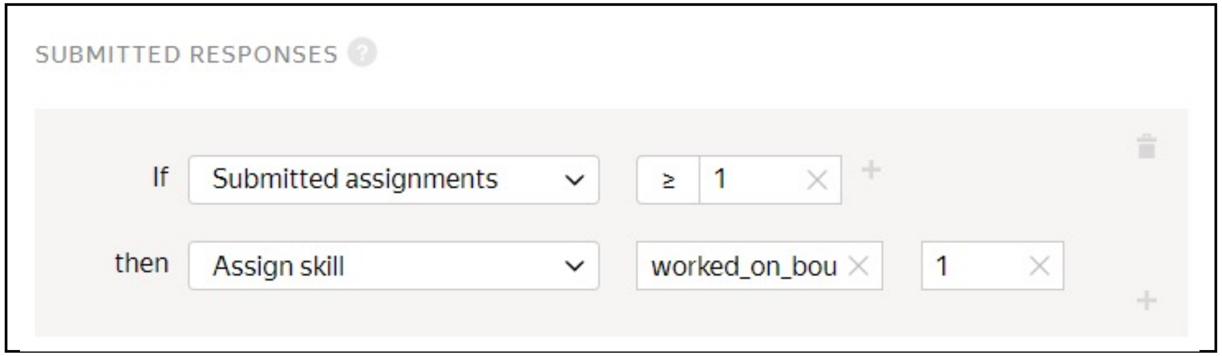
☑ Open pool if closed

- Submitted responses. Add a rule to mark users who completed at least one task in the pool

Tip. *If the skill you created doesn't appear in the drop-down list, save the pool, and then open it for editing again*

Optionally, add other quality control rules

Tip. *Control tasks and majority vote are not used in this type of project, because user highlighting must exactly match the reference, which is practically impossible*

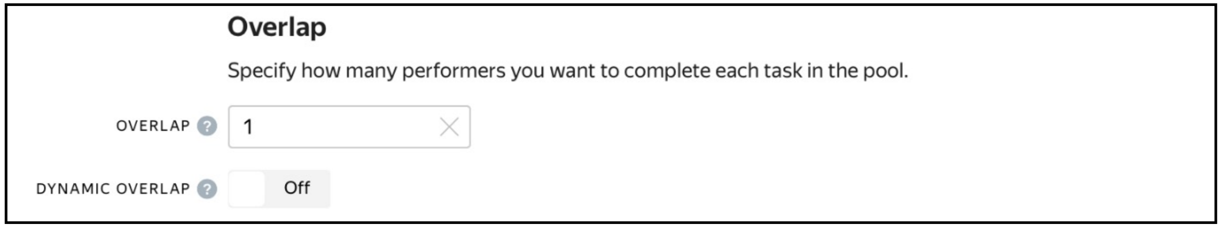


SUBMITTED RESPONSES

If Submitted assignments ≥ 1

then Assign skill worked_on_bou 1

7. Overlap. This is the number of users who will complete the same task. Because everyone will be outlining your traffic lights in their own unique way we will show each photo to only one performer



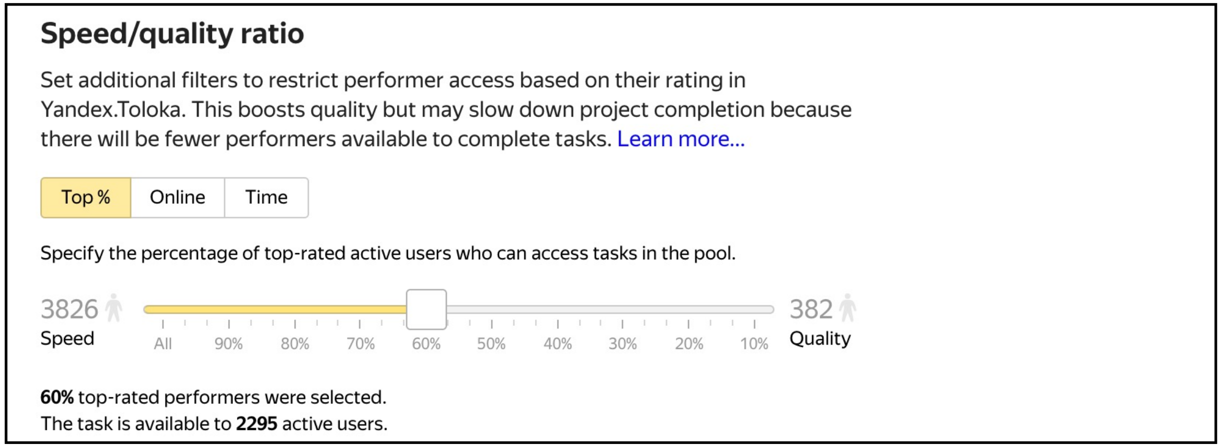
Overlap

Specify how many performers you want to complete each task in the pool.

OVERLAP 1

DYNAMIC OVERLAP Off

8. Optionally, specify the percentage of top-rated performers in the Speed / Quality ratio



Speed/quality ratio

Set additional filters to restrict performer access based on their rating in Yandex.Toloka. This boosts quality but may slow down project completion because there will be fewer performers available to complete tasks. Learn more...

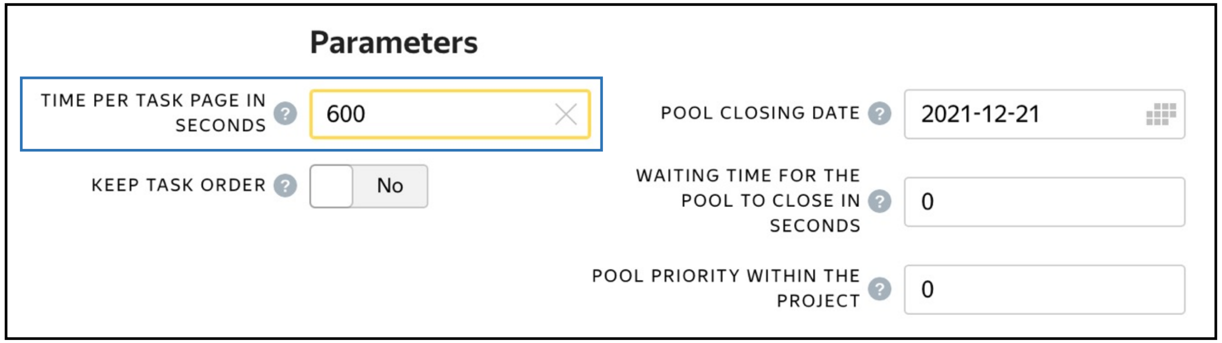
Top % | Online | Time

Specify the percentage of top-rated active users who can access tasks in the pool.

60% top-rated performers were selected.
The task is available to **2295** active users.

Important: This can slow down pool completion.

9. Time allowed for completing a task page (for example, 600 seconds)



Parameters

TIME PER TASK PAGE IN SECONDS: 600

KEEP TASK ORDER: No

POOL CLOSING DATE: 2021-12-21

WAITING TIME FOR THE POOL TO CLOSE IN SECONDS: 0

POOL PRIORITY WITHIN THE PROJECT: 0

10. Save the pool

Preparing and uploading a file with tasks

1. Open the file with aggregated results from the project #1

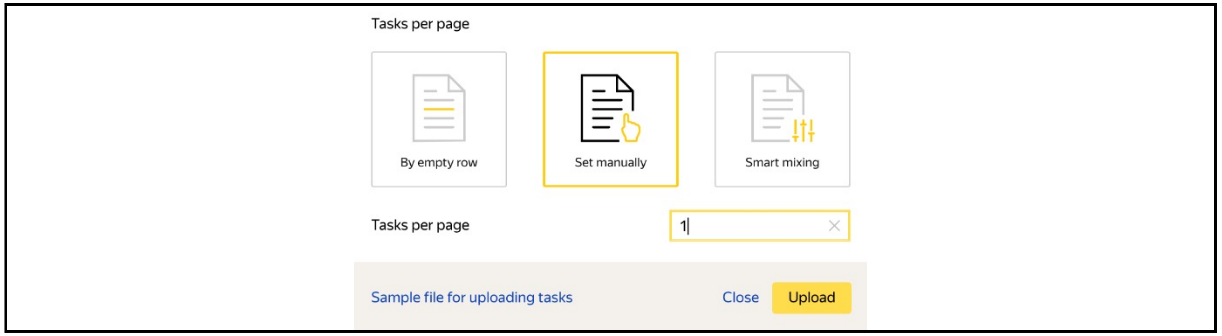
2. Select only images suitable for highlighting (**OK** answers or another value if you have changed it in the **"result"** field). Use a text editor or a spreadsheet editor

3. Copy the column with the selected links to a new page or document and give a name to the INPUT:image column

Important: If you changed the input field name in the project, change the name in the file as well: INPUT:<your field name>

4. Save the file in TSV format

5. [Upload the file]() to the pool by selecting **Set manually**. Set 1 task per page



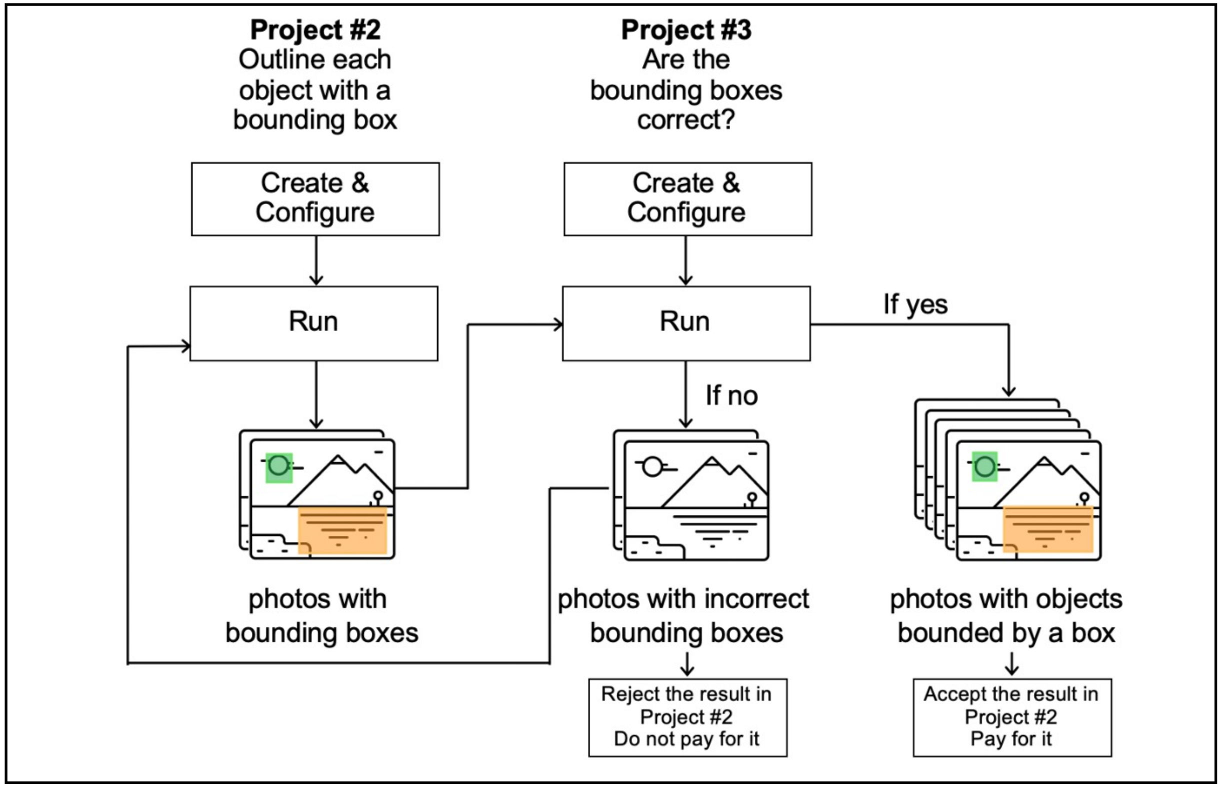
Tasks per page

Tasks per page: 1

6. Start the pool

Project #3

Are the bounding boxes correct?



Project #2
Outline each object with a bounding box

Project #3
Are the bounding boxes correct?

Create & Configure

Run

If yes

If no

photos with bounding boxes

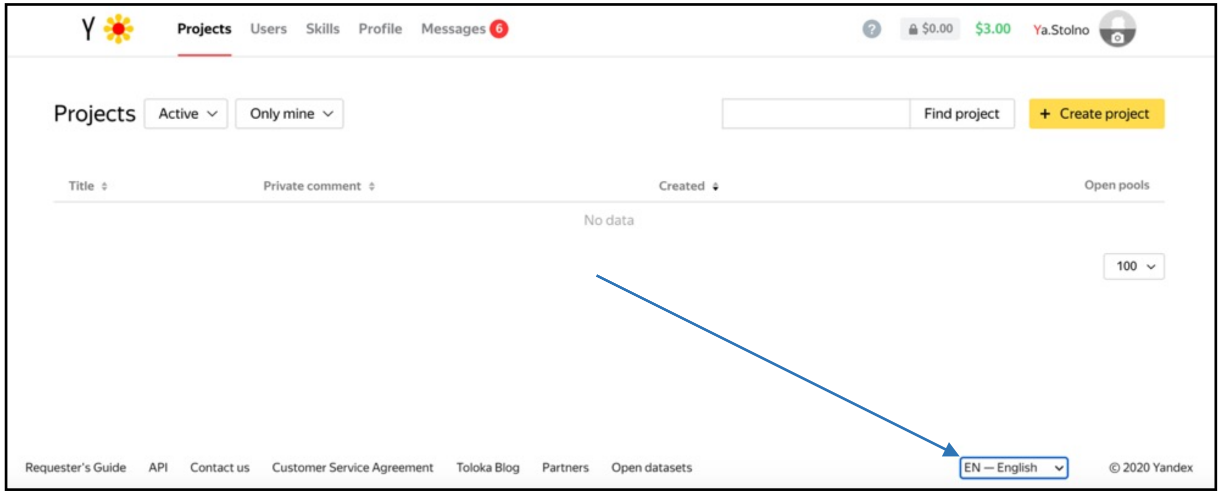
photos with incorrect bounding boxes

photos with objects bounded by a box

Reject the result in Project #2
Do not pay for it

Accept the result in Project #2
Pay for it

Important: If you just start using Toloka, make sure that the English language is selected



Project creation

1. Click the **+ Create project** button and choose **the Selecting a region in an image** template. This template contains the editor for image area selection, which help you to show performers images with bounding boxes

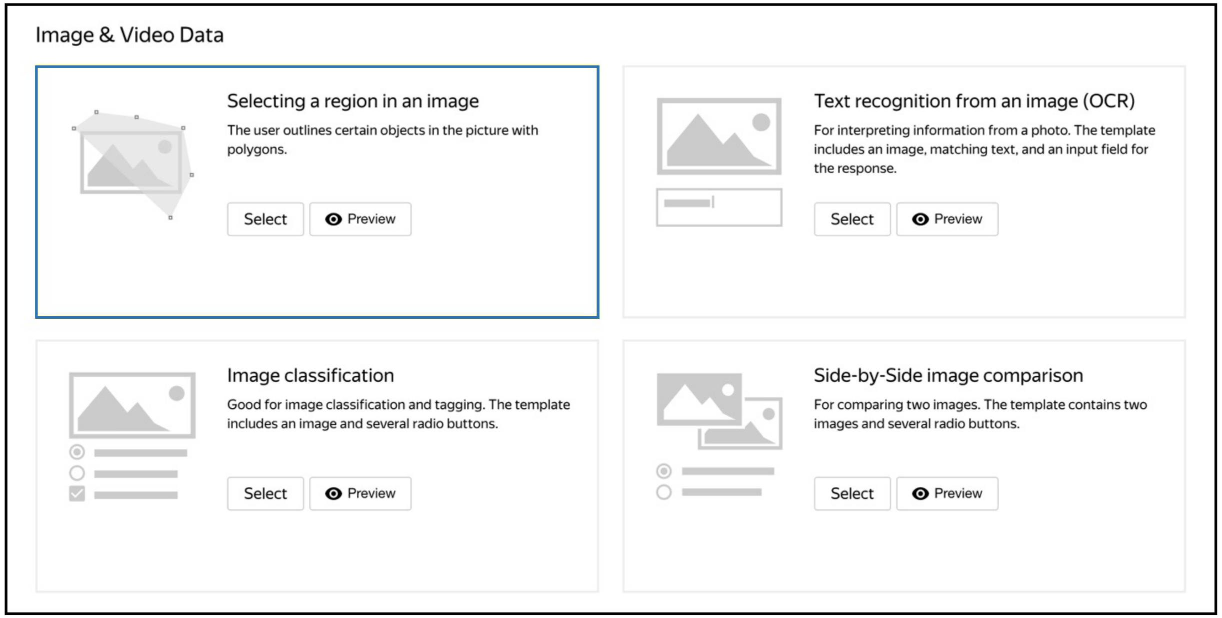


Image & Video Data

Selecting a region in an image

The user outlines certain objects in the picture with polygons.

Text recognition from an image (OCR)

For interpreting information from a photo. The template includes an image, matching text, and an input field for the response.

Image classification

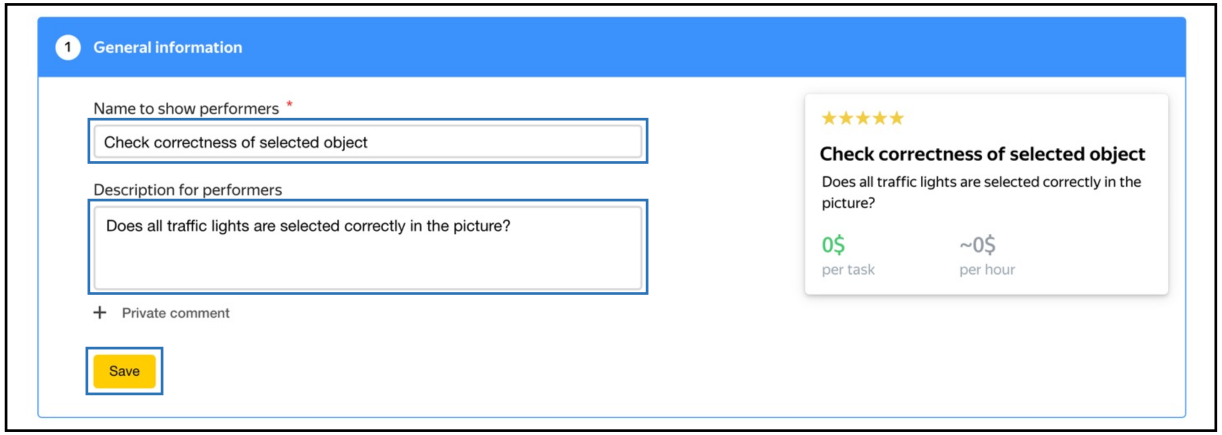
Good for image classification and tagging. The template includes an image and several radio buttons.

Side-by-Side image comparison

For comparing two images. The template contains two images and several radio buttons.

2. Enter a clear project name and description. It will be visible for real people

3. Click **Save** button to save the general information



1 General information

Name to show performers *

Check correctness of selected object

Description for performers

Does all traffic lights are selected correctly in the picture?

★★★★★

Check correctness of selected object

Does all traffic lights are selected correctly in the picture?

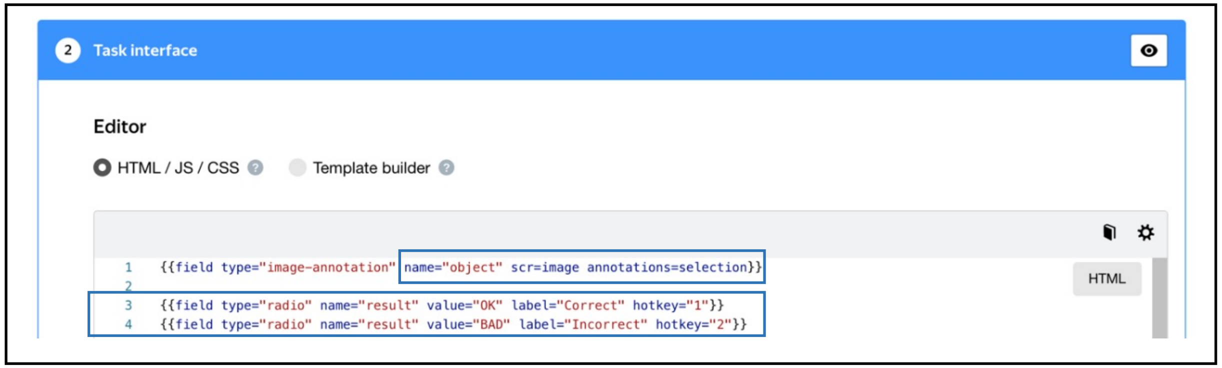
0$ per task

~0$ per hour

4. Create the task interface

   - Add the **"annotations=selection"** parameter to the area selection editor to show the selected object to the performer.
   - Chose another name for image area selection editor. For example, **"object"**.
   - Add two radio buttons:

     {{field type="radio" name="result" value="OK" label="Correct" hotkey="1"}}
     {{field type="radio" name="result" value="BAD" label="Incorrect" hotkey="2"}}



Editor

● HTML / JS / CSS ○ Template builder

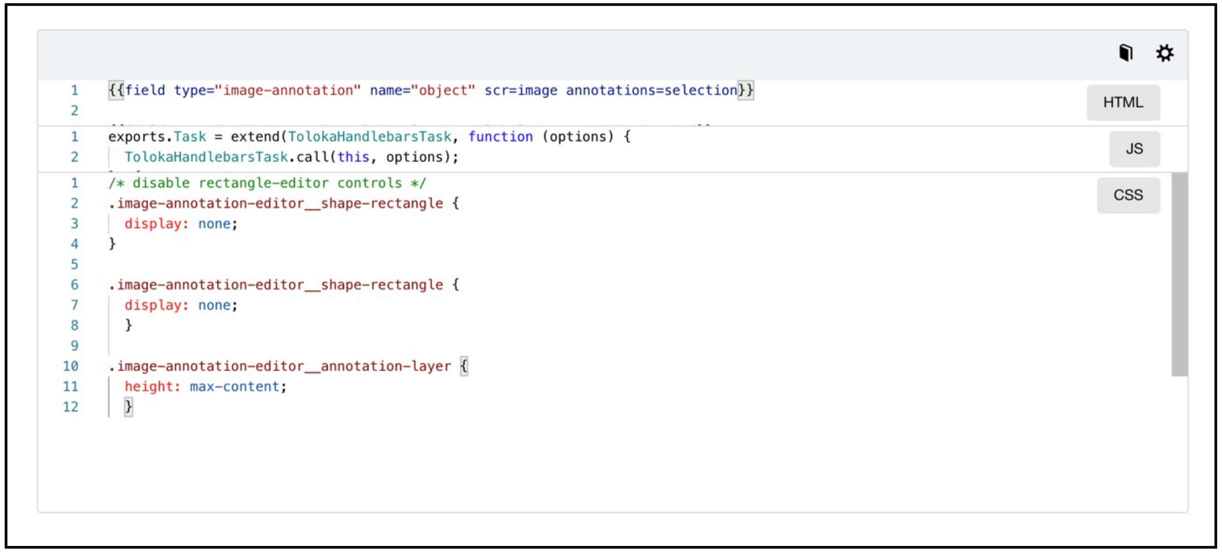
```
{{field type="image-annotation" name="object" scr=image annotations=selection}}

{{field type="radio" name="result" value="OK" label="Correct" hotkey="1"}}
{{field type="radio" name="result" value="BAD" label="Incorrect" hotkey="2"}}
```

   - Remove the area selection tool from the interface: in this task, you don't need to correct or add markup.
   - Add code to the CSS block:

     .image-annotation-editor__shape-rectangle {display: none;}

   - Add code to set the height of the image markup interface that matches the image size:

     .image-annotation-editor__annotation-layer {height: max-content;}

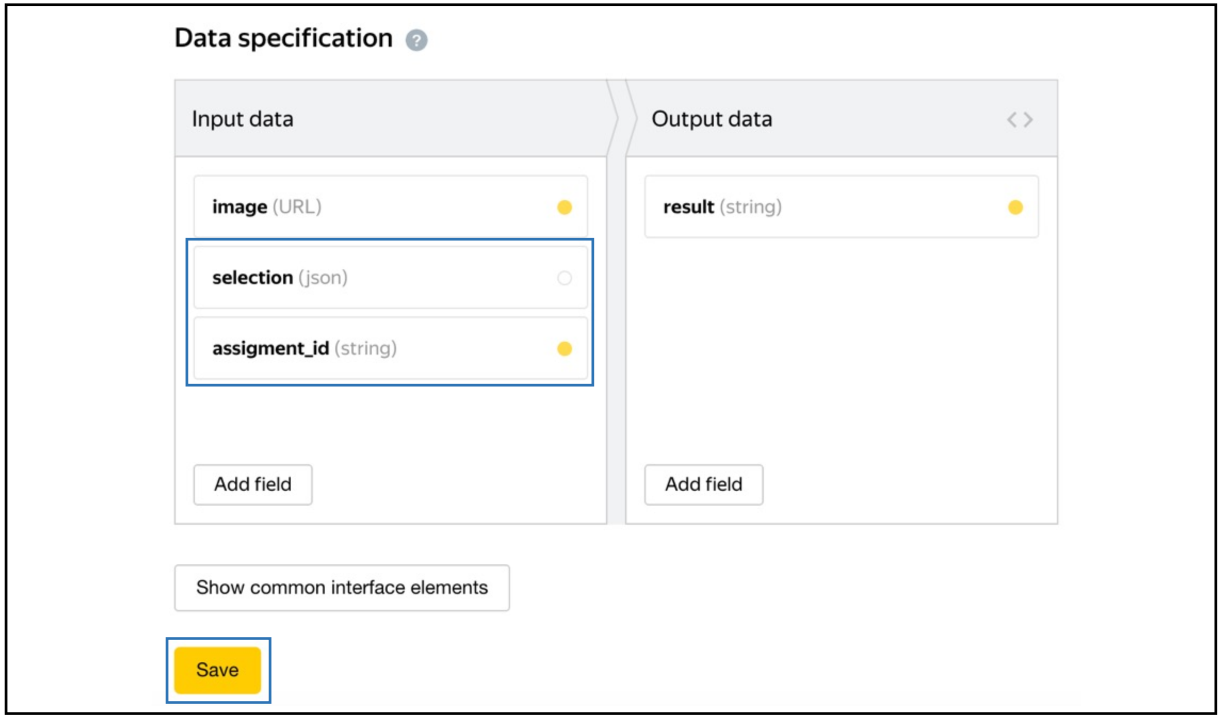


```
{{field type="image-annotation" name="object" scr=image annotations=selection}}

```

```
exports.Task = extend(TolokaHandlebarsTask, function (options) {
  TolokaHandlebarsTask.call(this, options);
```

```
/* disable rectangle-editor controls */
.image-annotation-editor__shape-rectangle {
  display: none;
}

.image-annotation-editor__shape-rectangle {
  display: none;
  }

.image-annotation-editor__annotation-layer {
  height: max-content;
  }
```

5. Define parameters for the input and output data:
    - The **“image”** input data field with the link type will be used to pass the image links to the performers.
    - The **"selection"** field with the "json" type, will be used to pass the coordinates of the objects selected in the previous task. Don't make this field required.
    - The **"assignment_id"** field with the "line" type, will be used to pass the number of the completed task.
    - The **"result"** field with the "string" type will be used to write the result of performer's selection: correct or incorrect

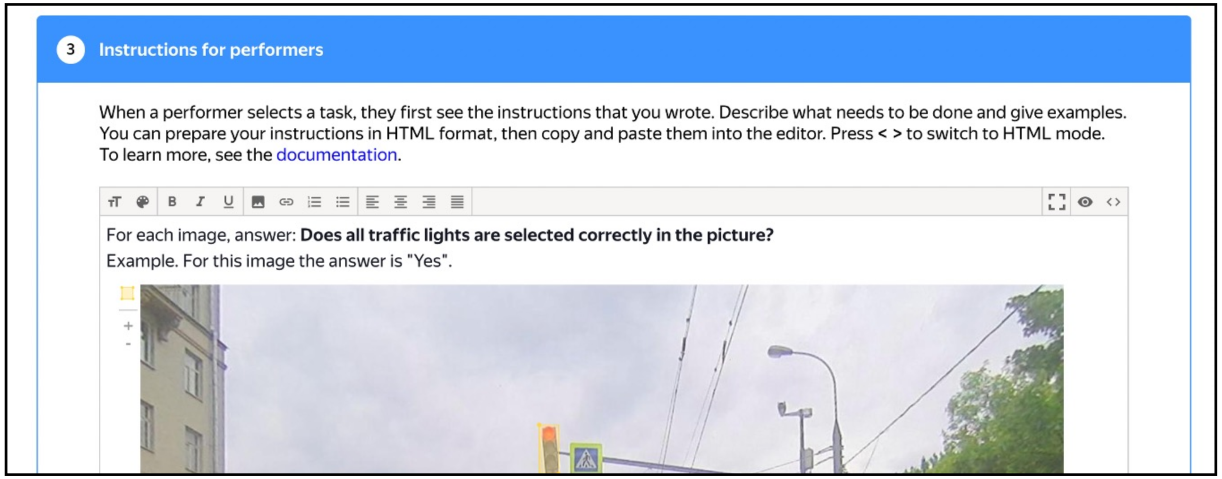


Data specification

Input data	Output data
image (URL) ●	result (string) ●
selection (json) ○	
assigment_id (string) ●	

6. Don’t forget to save

7. Write short and simple instructions



3 Instructions for performers

When a performer selects a task, they first see the instructions that you wrote. Describe what needs to be done and give examples. You can prepare your instructions in HTML format, then copy and paste them into the editor. Press < > to switch to HTML mode. To learn more, see the documentation.

For each image, answer: **Does all traffic lights are selected correctly in the picture?**
Example. For this image the answer is "Yes".

8. Click **Save** button to save the instruction

9. Click the **Preview** button to see the performer's view of the task.

   You will see tasks with standard pictures on the page. You can set the number of tasks per page in the pool

10. Select the radio buttons in the preview and make sure that the task can be completed

11. Click **Finish** button to save the project

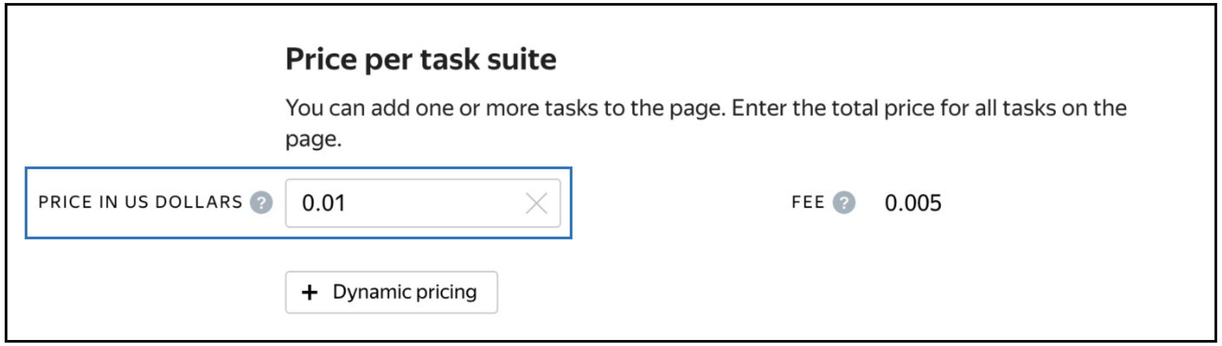
Pool creation

1. Click **Add pool**

2. Give the pool any convenient name and description. You are the only one who can see them

3. Specify the pool parameters:

   - Set the price per task page (for example, $0.01)



Price per task suite

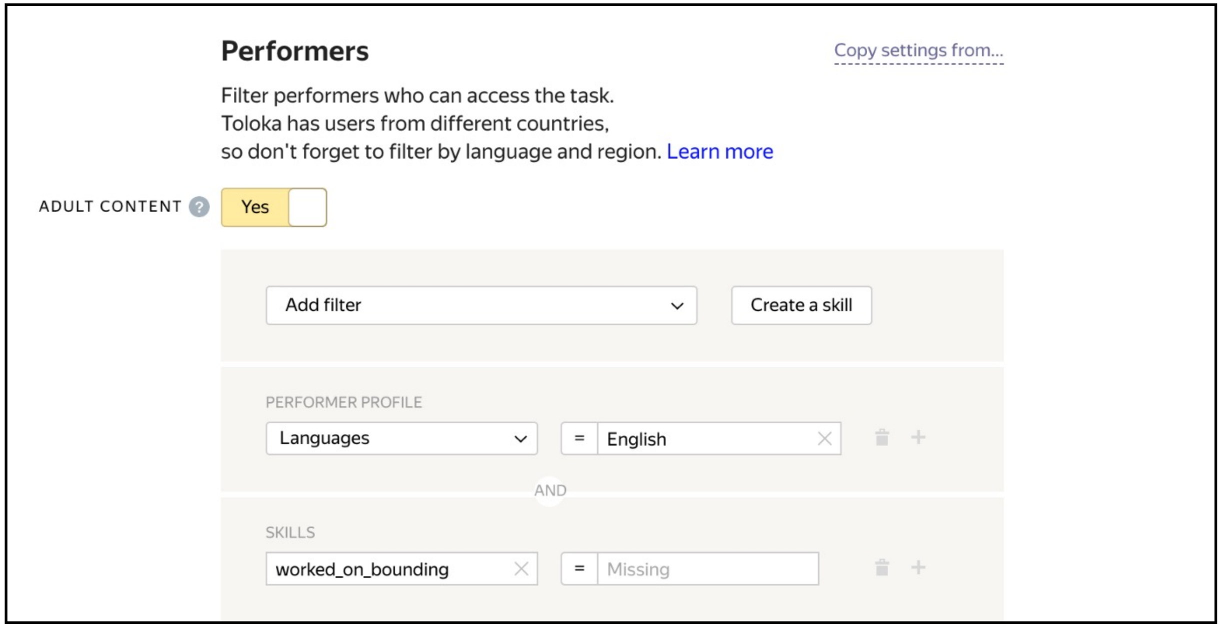
You can add one or more tasks to the page. Enter the total price for all tasks on the page.

PRICE IN US DOLLARS ? 0.01

FEE ? 0.005

4. Set up user filters.

   - Select English-speaking performers using the "Language = English" filter. Prevent performers who completed previous tasks from checking this one. To do this, set a filter with the "worked_on_bounding" skill: *The **"worked_on_bounding"** skill = missing (empty field)*



Performers

Filter performers who can access the task.
Toloka has users from different countries,
so don't forget to filter by language and region. Learn more

ADULT CONTENT ? Yes

PERFORMER PROFILE

Languages = English

AND

SKILLS

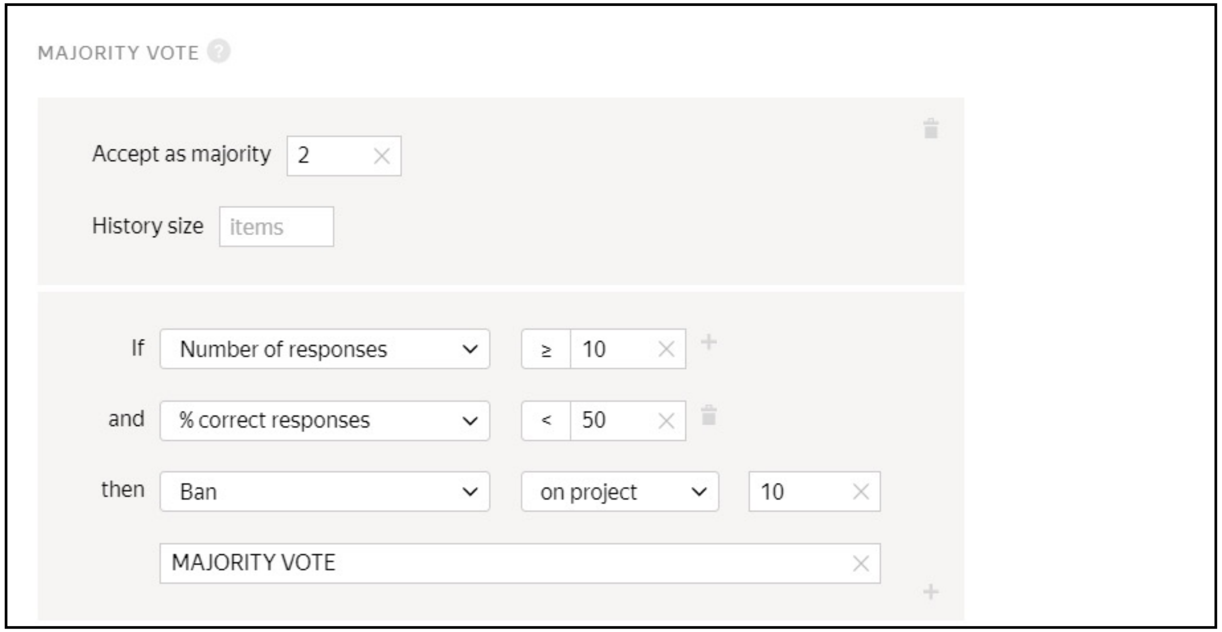
worked_on_bounding = Missing

5. Set up quality control:

   - Majority vote. Add the majority vote rule. Specify how many responses you accept as majority. For example, 3 out of 5 or 2

Important: The rule takes effect when the number of responses for the task is equal to the overlap. To get the required number of responses faster, turn on the Keep task order option in the pool parameters

Optionally, add other quality control rules



MAJORITY VOTE

Accept as majority: 2

History size: items

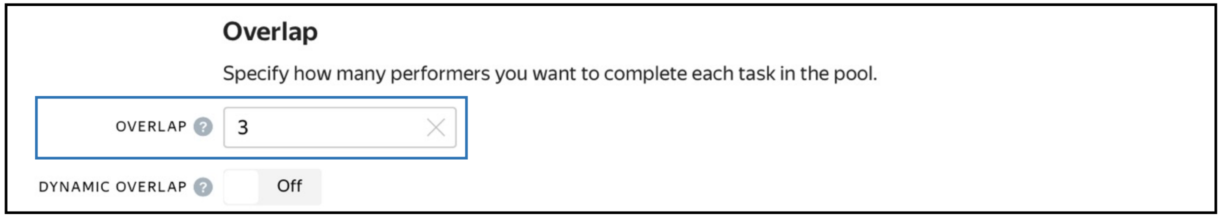
If Number of responses ≥ 10

and % correct responses < 50

then Ban on project 10

MAJORITY VOTE

6. Overlap. This is the number of users who will complete the same task. For example, 3 is enough for Majority Vote in this case



Overlap

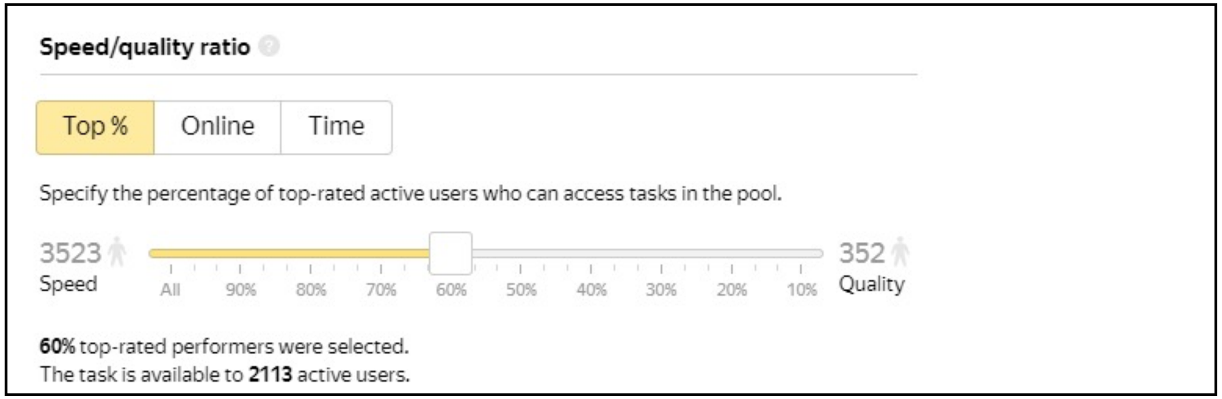
Specify how many performers you want to complete each task in the pool.

OVERLAP: 3

DYNAMIC OVERLAP: Off

7. Optionally, specify the percentage of top-rated performers in the Speed / Quality ratio. Important: This can slow down pool completion

Important: This can slow down pool completion



Speed/quality ratio

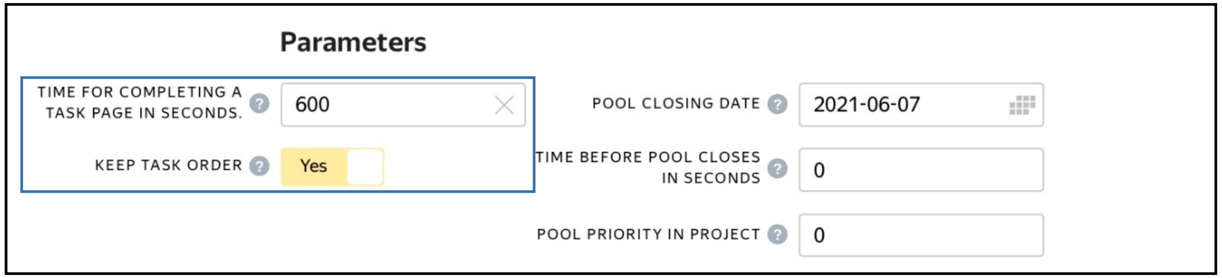
Top % | Online | Time

Specify the percentage of top-rated active users who can access tasks in the pool.

60% top-rated performers were selected.
The task is available to **2113** active users.

8. Time allowed for completing a task page (for example, 600 seconds)

9. Keep task order. (Activate the radio button to get the number of responses considered the "majority vote" faster.)



Parameters

TIME FOR COMPLETING A TASK PAGE IN SECONDS. 600

KEEP TASK ORDER Yes

POOL CLOSING DATE 2021-06-07

TIME BEFORE POOL CLOSES IN SECONDS 0

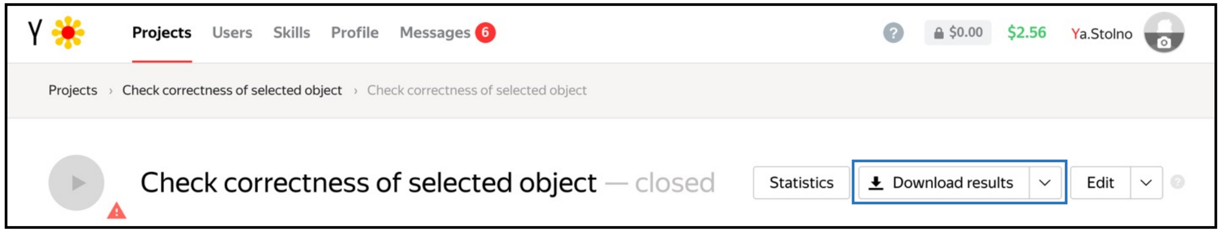
POOL PRIORITY IN PROJECT 0

10. Save the pool

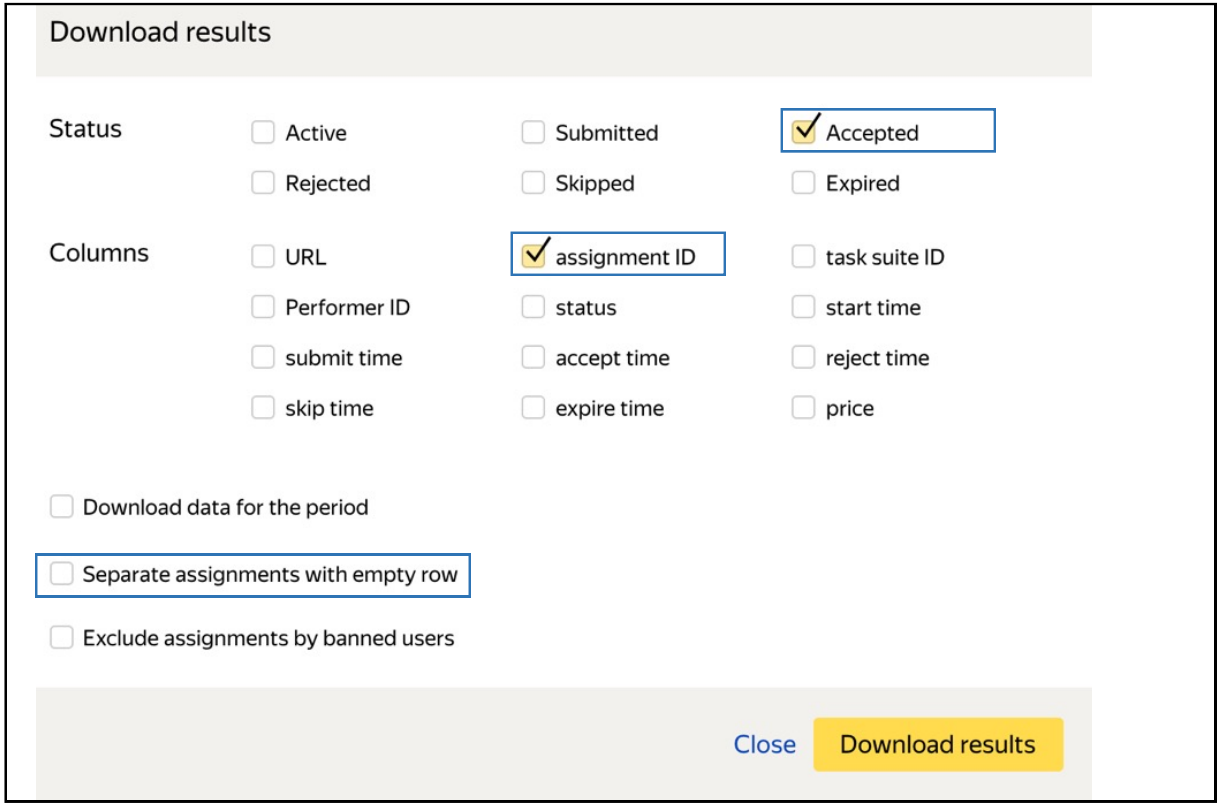
Preparing and uploading a file with tasks

1. Wait until the pool of project #2 on "object highlighting" is completed

2. Open the pool page in Project #2 and click the **Download results** button


Projects
Users
Skills
Profile
Messages 6
$0.00
$2.56
Ya.Stolno
Projects › Check correctness of selected object › Check correctness of selected object
Check correctness of selected object — closed
Statistics
Download results
Edit

- Clear the **Accepted** checkbox and select **Submitted**.
- Clear **link, user ID, status, start time and Separate assignments with empty row** checkboxes. This will give you a list of unreviewed tasks



Download results

Status
□ Active □ Submitted ☑ Accepted
□ Rejected □ Skipped □ Expired

Columns
□ URL ☑ assignment ID □ task suite ID
□ Performer ID □ status □ start time
□ submit time □ accept time □ reject time
□ skip time □ expire time □ price

□ Download data for the period

□ Separate assignments with empty row

□ Exclude assignments by banned users

3. Once you download the file just **change the headings in the file**.
    - Change the name of the **"OUTPUT:result"** column to **"INPUT:selection"**.
    - Change the **"ASSIGNMENT:assignment_id"** column name to **INPUT:assignment_id"**.
    - Leave the **"INPUT:image"** column unchanged.

    - You can keep **GOLDEN:result, HINT:text, ACCEPT:verdict** and **ACCEPT:comment** columns unchanged

---

4. Save the file in TSV format

Important: You can use a simple notepad for this action

---

5. Open the pool page in Project #3

---

6. Upload the file to the pool by selecting **Set manually**. Set the number of tasks per page (for example, 10)

---

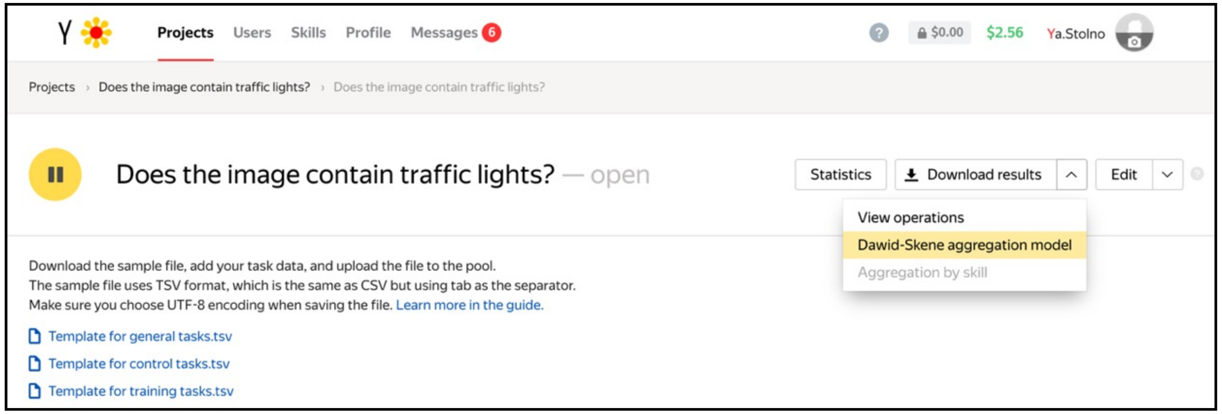
7. Start the pool

Receiving responses

Disclaimer: Aggregation takes from 5 to 20 minutes. During this time, you can start configuring your next project. Refresh the Operations page to check progress.

1. Wait until the pool is completed

2. Click the arrow next to **the Download results** button and run aggregation using the [Dawid-Skene model](#)

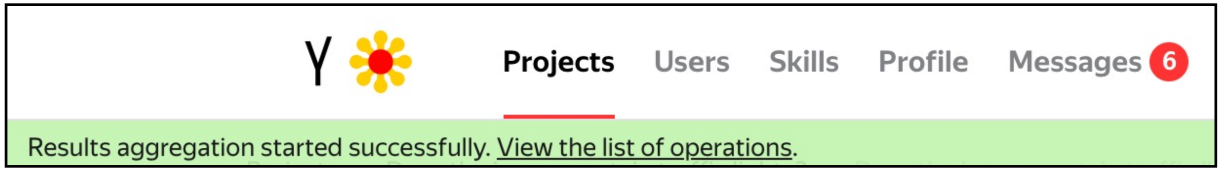


Does the image contain traffic lights? — open

Download the sample file, add your task data, and upload the file to the pool.
The sample file uses TSV format, which is the same as CSV but using tab as the separator.
Make sure you choose UTF-8 encoding when saving the file. Learn more in the guide.

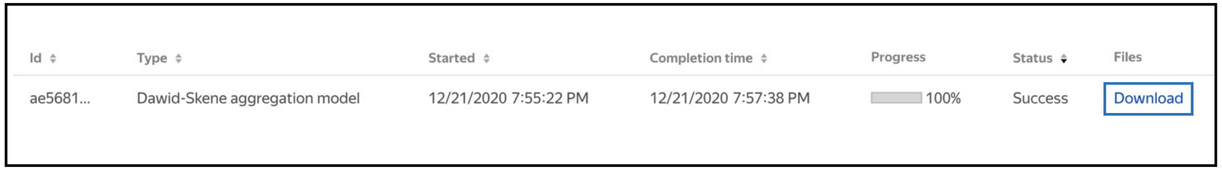
- Template for general tasks.tsv
- Template for control tasks.tsv
- Template for training tasks.tsv

3. Go to the operations list and wait until aggregation finishes



Results aggregation started successfully. View the list of operations.

4. Download the responses



Id	Type	Started	Completion time	Progress	Status	Files
ae5681...	Dawid-Skene aggregation model	12/21/2020 7:55:22 PM	12/21/2020 7:57:38 PM	100%	Success	Download

Upload review results

As you set **post acceptance** in the pool settings in Project #2, you need to check the performers' responses within the time limit set in the **Deadline** field

1. Open the file with aggregated results in a spreadsheet editor

2. Add the following columns:
   - **"ACCEPT:verdict"** — The result of verification.
   - **"ACCEPT:comment"** — Comments for performers if responses were rejected (for example, which part of the instructions wasn't followed)

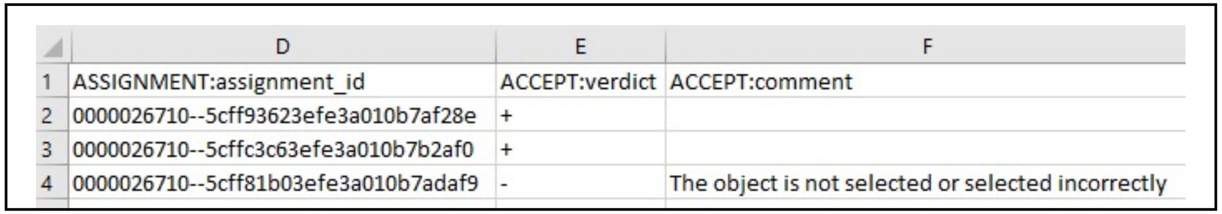
3. Change the name of the **"INPUT:assignment_id"** column to **"ASSIGNMENT:assignment_id"**

4. Fill in the **"ACCEPT:verdict"** and **"ACCEPT:comment"** columns:

   - If the aggregated result for the task is OK, put **"+"** then the task will be accepted.

   - If the result is BAD, put "-" then the task will be rejected. Enter the reason for rejection in the **"ACCEPT:comment"** field . For example: *The object is not selected or selected incorrectly*

5. Now you can delete the other columns. Save the edited TSV file

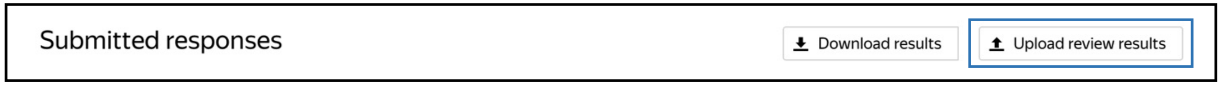


	D	E	F
1	ASSIGNMENT:assignment_id	ACCEPT:verdict	ACCEPT:comment
2	0000026710--5cff93623efe3a010b7af28e	+	
3	0000026710--5cffc3c63efe3a010b7b2af0	+	
4	0000026710--5cff81b03efe3a010b7adaf9	-	The object is not selected or selected incorrectly

Open the pool page in Project #2

6. Click [Review assignments](#) on the pool page above the progress bar

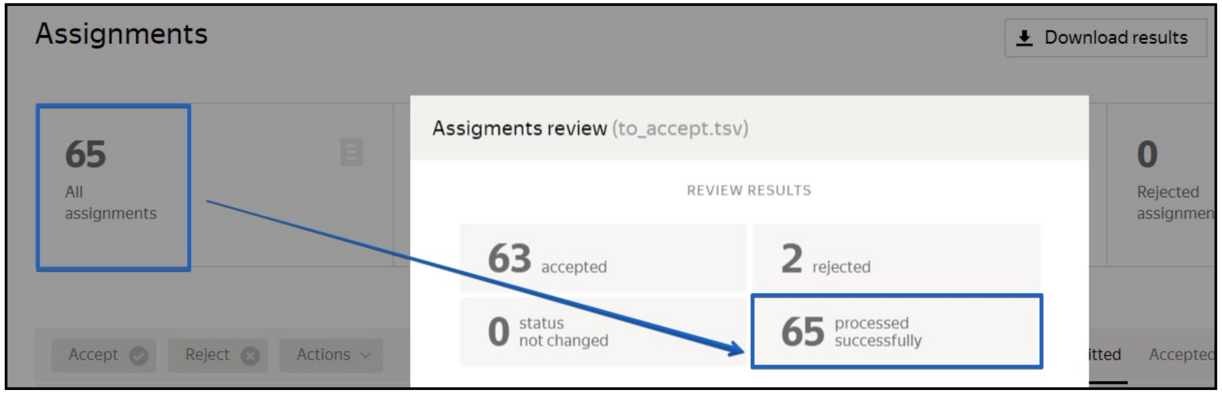
7. Click **Upload review results**



Submitted responses

8. Select the file and upload it to Toloka

9. Check that all tasks have changed their status to accepted or rejected



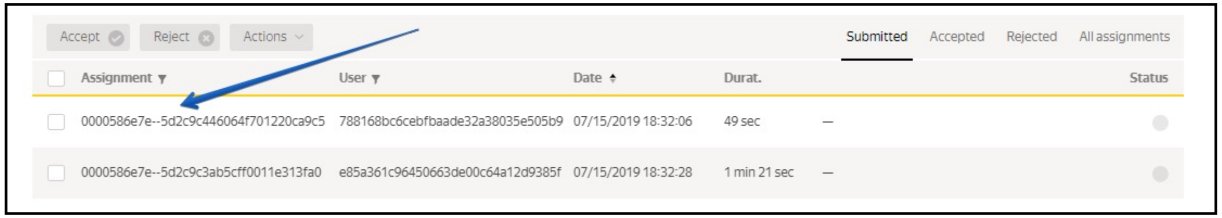
10. If you rejected tasks and set up the rule to send them for re-completion, the pool will open again, and these tasks will be resent to other performers. After the pool is marked up, download the new results and submit them for review. Download the results. Repeat these steps until all the images from the second project are correctly marked up

Review assignments online (the way to see the final results of your pipeline)

You can also review assignments by yourself and see the results of the crowdsourcing pipeline that you have created.

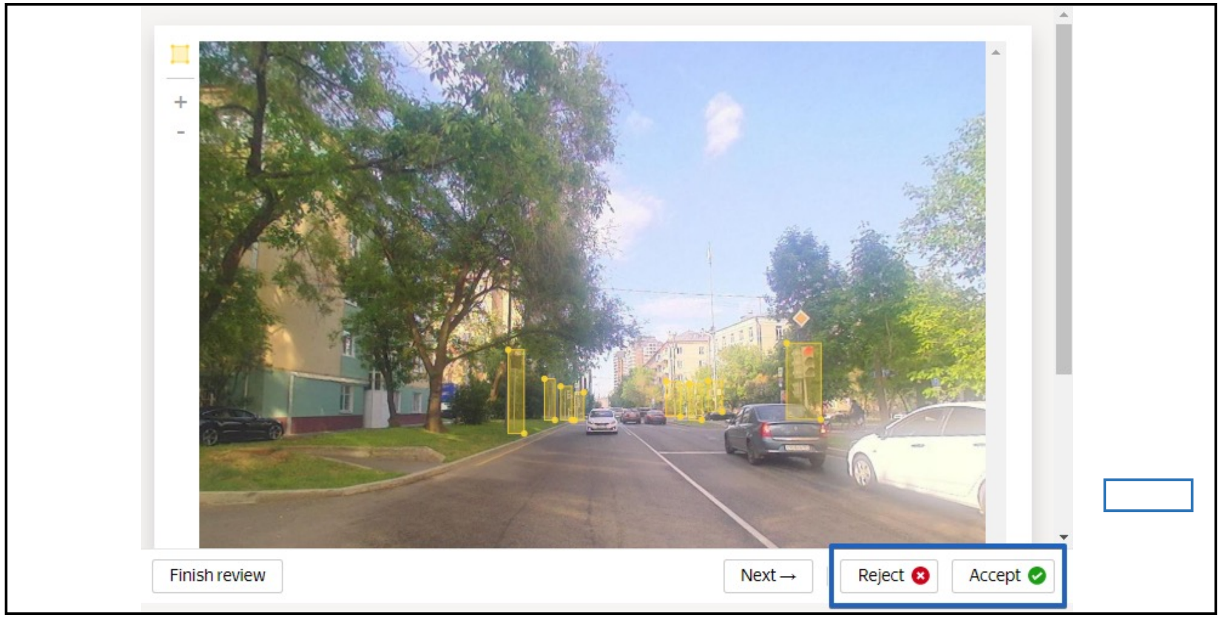
1. Wait until the pool is completed

2. Click the **Review assignments** button on the pool page

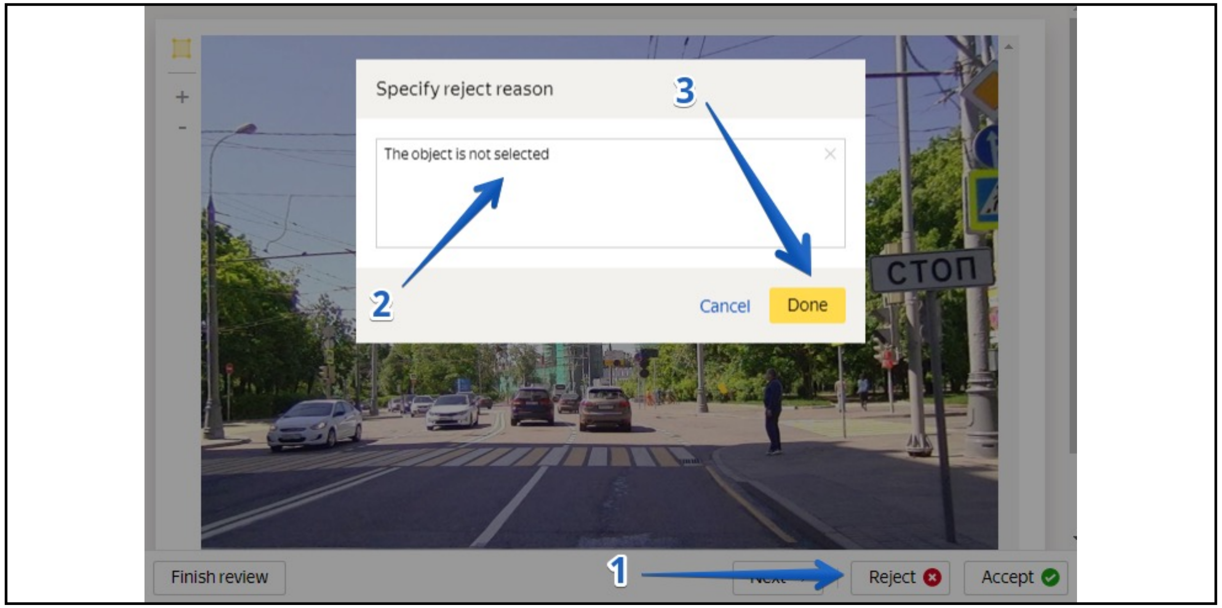


□	Assignment	User	Date	Durat.		Status
□	0000586e7e--5d2c9c446064f701220ca9c5	788168bc6cebfbaade32a38035e505b9	07/15/2019 18:32:06	49 sec	—	●
□	0000586e7e--5d2c9c3ab5cff0011e313fa0	e85a361c96450663de00c64a12d9385f	07/15/2019 18:32:28	1 min 21 sec	—	●

- Choose an assignment then click **Accept** or **Reject**



- For rejected assignments, enter a comment (explain why you decline it)



Specify reject reason

The object is not selected

3. Click the **Download results** button and select **Accepted**

4. Here you are photos with traffic lights bounded by a box