# Table of Contents

1. [Host Challenge Using GitHub](#host-challenge-using-github)
   * [Step 1: Use the Starter Template](#step-1-use-the-starter-template)
   * [Step 2: Generate a GitHub Token](#step-2-generate-a-github-token)
   * [Step 3: Configure Host Credentials](#step-3-configure-host-credentials)
   * [Step 4: Set Up Automated Update Push](#step-4-set-up-automated-update-push)
   * [Step 5: Update Challenge Details](#step-5-update-challenge-details)
   * [Step 6: Push and Monitor Changes](#step-6-push-and-monitor-changes)
   * [Step 7: Verify the Challenge](#step-7-verify-the-challenge)
2. [Host Prediction Upload Based Challenge](#host-prediction-upload-based-challenge)
   * [Step 1: Set Up the Challenge Configuration](#X21e423003bf643a29239a84e1dce9cf0220d1aa)
   * [Step 2: Edit the Challenge Configuration File](#X28db9524fa94b1ea399f02994c2192e58e5b598)
   * [Step 3: Customize the Evaluation Script](#step-3-customize-the-evaluation-script)
   * [Step 4: Update Challenge HTML Templates](#step-4-update-challenge-html-templates)

Host Challenge Using GitHub

This section describes how to host a challenge by leveraging GitHub’s workflow integration.

### Step 1: Use the Starter Template

* **Action:** Fork or use the [EvalAI-Starters](https://github.com) repository template.
* **Tip:** Refer to [GitHub’s documentation](https://docs.github.com) on how to use a repository as a template.
* **Outcome:** A new repository is created, pre-populated with the necessary structure for hosting a challenge.

### Step 2: Generate a GitHub Token

* **Action:** Create a [GitHub personal acccess token](https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/creating-a-personal-access-token) with appropriate permissions.
* **Tip:** Once generated, copy the token to your clipboard.
* **Integration:** Add the token to your forked repository’s **GitHub secrets** under the key **AUTH\_TOKEN.**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

### Step 3: Configure Host Credentials

* **Action:** Log in to EvalAI and collect the following details:
  + **evalai\_user\_auth\_token**: Available from your profile page (click on “Get your Auth Token”).
  + **host\_team\_pk**: Found on the host team page (copy the team ID you wish to use).
  + **evalai\_host\_url**: Use https://eval.ai for production (or https://staging.eval.ai for staging).
* **Tip:** These credentials link your GitHub repository to your EvalAI account.

A screenshot of a computer

AI-generated content may be incorrect.

### 

### Step 4: Set Up Automated Update Push

* **Action:** In your forked repository, **create a branch named challenge** (this branch will be synced with EvalAI). [Git Branch](https://www.w3schools.com/git/git_branch.asp?remote=github)
* **Configuration:** Add the evalai\_user\_auth\_token and host\_team\_pk details in the file github/host\_config.json.
* **Outcome:** Your repository is now configured to push updates automatically to EvalAI when changes occur on the challenge branch.

A screenshot of a computer

AI-generated content may be incorrect.

### Step 5: Update Challenge Details

* **Action:** Modify the **YAML configuration, HTML templates, and evaluation script** as needed.
* **Reference:** Consult the [EvalAI challenge creation documentation](https://evalai.readthedocs.io/en/latest/configuration.html) (or your internal reference) for field descriptions and further details.
* **Tip:** Ensure that every detail reflects your challenge’s requirements (e.g., start/end dates, evaluation metrics, submission limits).
* Note links to documents:
  + [Challenge configuration — EvalAI 1.1 documentation](https://evalai.readthedocs.io/en/latest/configuration.html)
  + [Writing Evaluation Script — EvalAI 1.1 documentation](https://evalai.readthedocs.io/en/latest/evaluation_scripts.html)

A screenshot of a computer program

AI-generated content may be incorrect.

### Step 6: Push and Monitor Changes

* **Action:** **Commit your changes and push the challenge branch**.
* **Monitoring:** Watch the build logs (available via GitHub) for any errors.
  + If errors exist, an issue is automatically created in the repository.
  + If there are no errors, the challenge will be created on EvalAI.
* **Tip:** **Always double-check the logs to ensure the build is successful**.

A screenshot of a computer

AI-generated content may be incorrect.

### Step 7: Verify the Challenge

* **Action:** Visit the [Hosted Challenges](https://pecha.services/web/hosted-challenges) page on EvalAI.
* **Outcome:** Your challenge will be publicly visible once the EvalAI admin approves it.
* **Tip:** For subsequent updates, simply push new changes to the challenge branch and wait for the build to complete.

A screenshot of a computer

AI-generated content may be incorrect.

Host Prediction Upload Based Challenge

This section covers the setup for challenges where participants upload prediction files (e.g., JSON, CSV, TXT).

### Step 1: Set Up the Challenge Configuration

* **Action:** Use the same [EvalAI-Starters](https://github.com) repository template to initialize your challenge.
* **Tip:** Follow the GitHub template instructions if you are unfamiliar with repository templates.

### Step 2: Edit the Challenge Configuration File

* **File:** Open the challenge\_config.yml file from your repository.

A screen shot of a computer program

AI-generated content may be incorrect.

* **Action:** Customize settings such as:
  + Start and end dates
  + Number of phases
  + Submission limits
  + Doc link: [Challenge configuration — EvalAI 1.1 documentation](https://evalai.readthedocs.io/en/latest/configuration.html)
* **Reference:** For field definitions and examples, refer to the EvalAI challenge configuration reference section.

### Step 3: Customize the Evaluation Script

* **Action:** Modify the evaluation script to define the metrics for submissions.
* **Tip:** Ensure the script aligns with the format of the uploaded predictions.
* **Reference:** Check the [Writing Evaluation Script](https://evalai.readthedocs.io/en/latest/evaluation_scripts.html) section for best practices and code examples.
* Doc Link: [Writing Evaluation Script — EvalAI 1.1 documentation](https://evalai.readthedocs.io/en/latest/evaluation_scripts.html)

### Step 4: Update Challenge HTML Templates

* **Action:** Edit the HTML templates in the templates/ directory.
* **Purpose:** Customize the **challenge’s landing page** with images, videos, tables, and inline CSS for custom styling.
* **Tip:** Include clear instructions for participants regarding:
  + Expected input/output file formats and names
  + Submission commands (e.g., using evalai-cli push <image>:<tag> --phase <phase\_name>)
  + Any additional setup or processing tips.
* **Outcome:** Your challenge is now fully documented and ready for review by the EvalAI team.

# Final Notes

* **Review Process:** Once submitted, the EvalAI team will review your challenge configuration and notify you upon approval.
  + Note: for now, we have removed the Submit step to prevent any conflicts of interest. So, any challenge will be addressed without approval.
* **Support:** If you encounter any issues during setup, contact the @uchihatashi GitHub.
* **Updates:** For any subsequent changes, update your repository (specifically the **challenge branch for GitHub-based challenges**) and let the automated build process update your challenge accordingly.