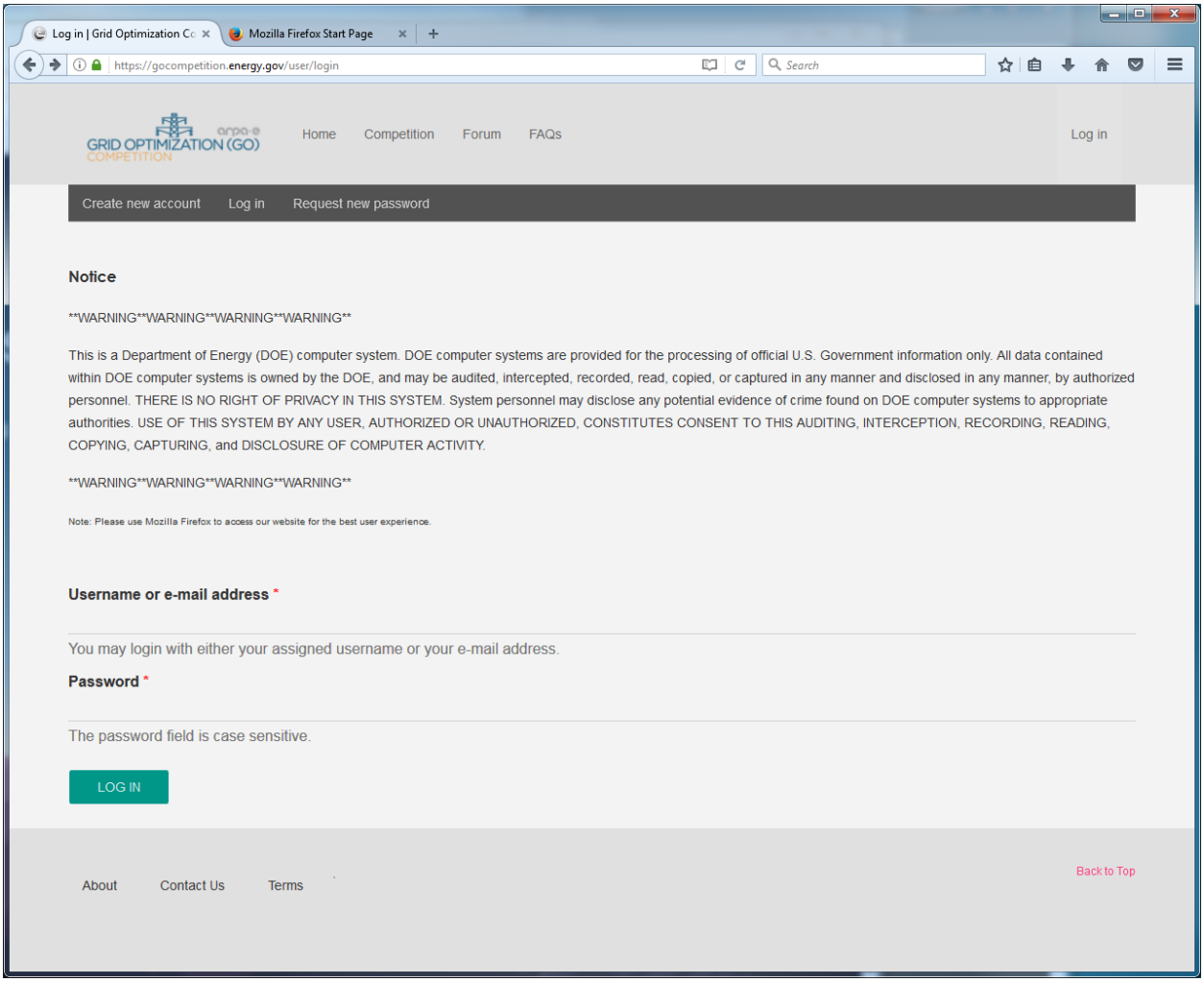
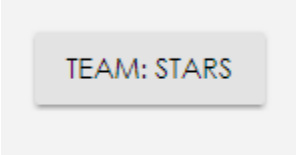
How to Add an SSH Public Key on GitHub and Make a Submission

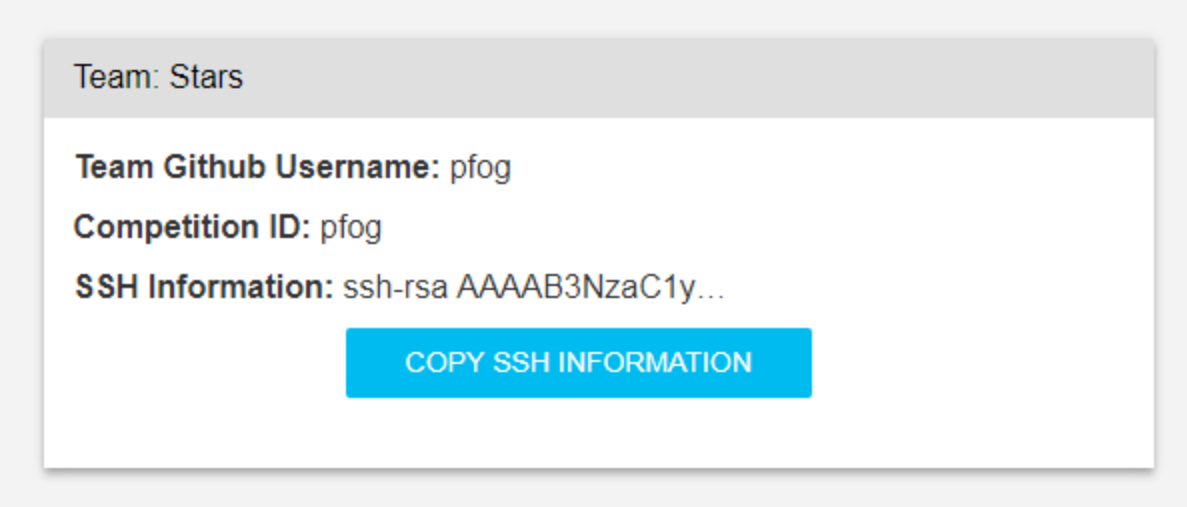
1. Go to the GO Competition login page https://gocompetition.energy.gov/user/login and log in using the username or e-mail and password associated with your account.



2. Upon login you will be in your "My account" page. The SSH information is on your Team page, which is accessible from the My account page or the Competition Overview by clicking the TEAM: xxx button (TEAM: STARS in this example).

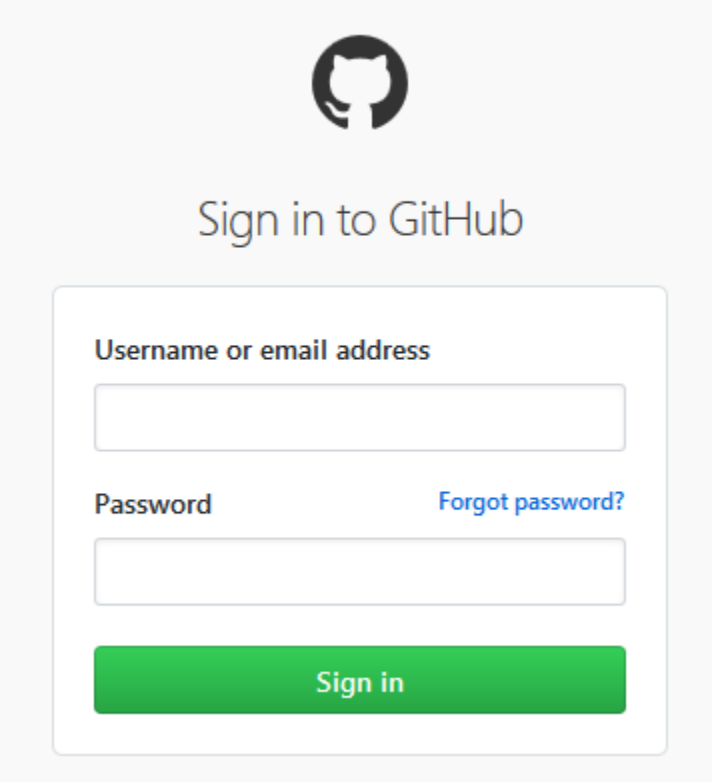


3. On the Team page you will see a blue box called COPY SSH INFORMATION

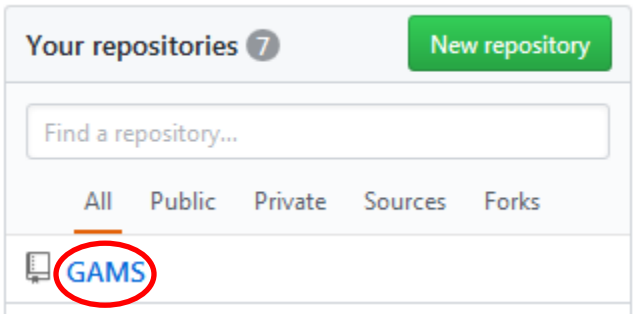


4. Click on the blue box to put the SSH information into your copy buffer (don't copy anything else until you complete the paste in step 10).

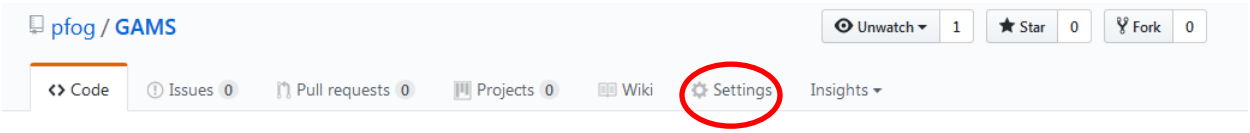
5. Go to the GitHub login page (https://github.com/login) and log in using the username/e-mail address and password associated with your account.



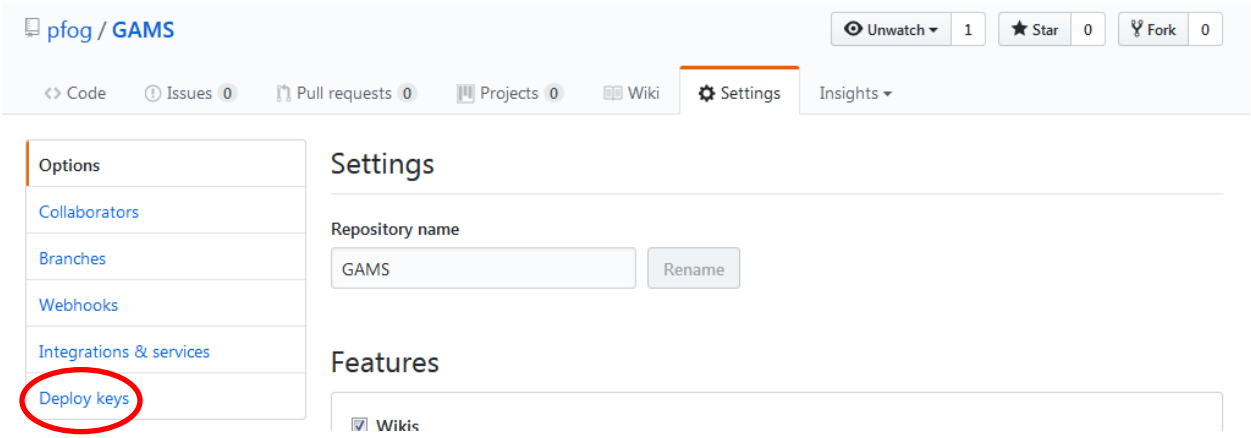
6. From the list of Your repositories, select the GO Competition related repository (GAMS in this example)



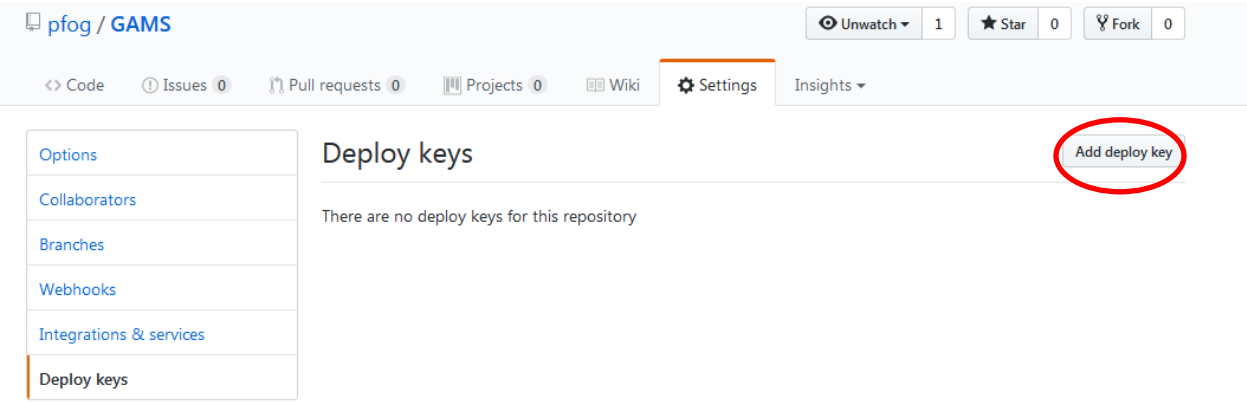
7. Click the "Settings" option from the tab bar along the top of the page.



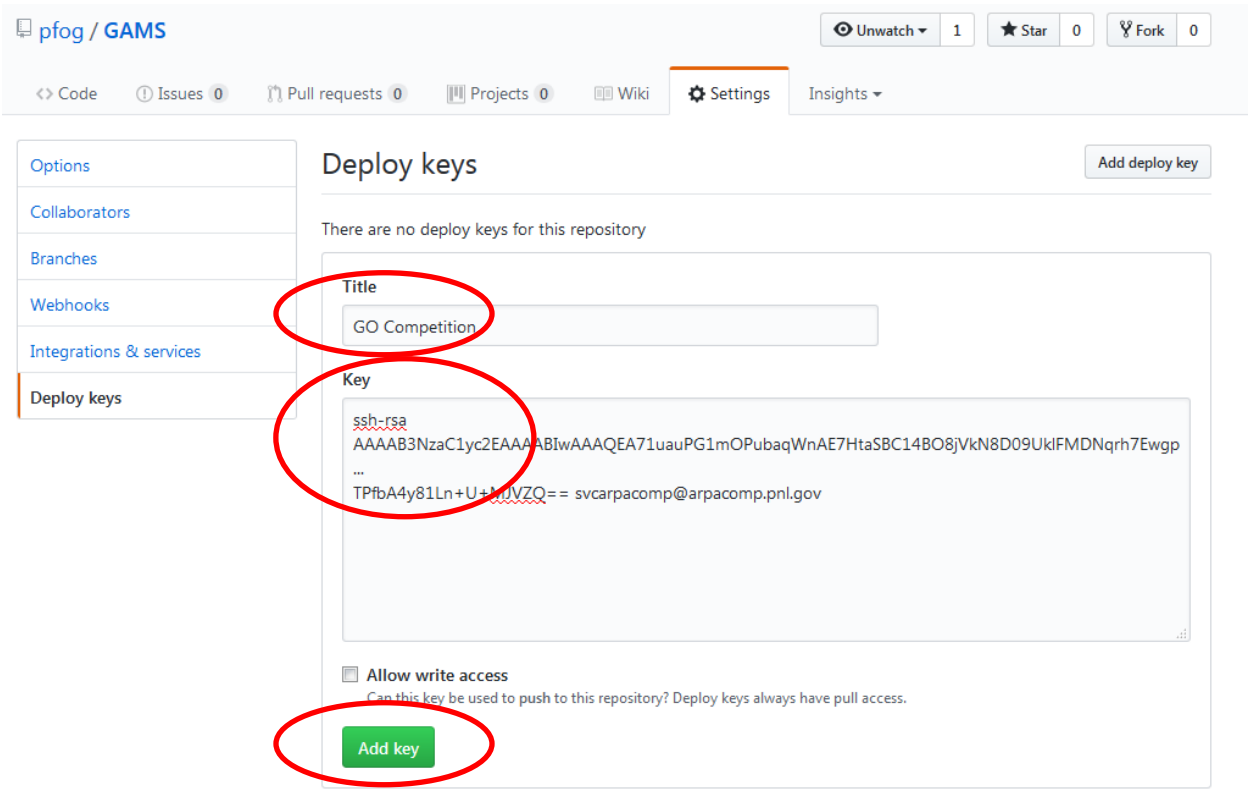
8. Click the "Deploy keys" option from the pane on the left hand side of the page.



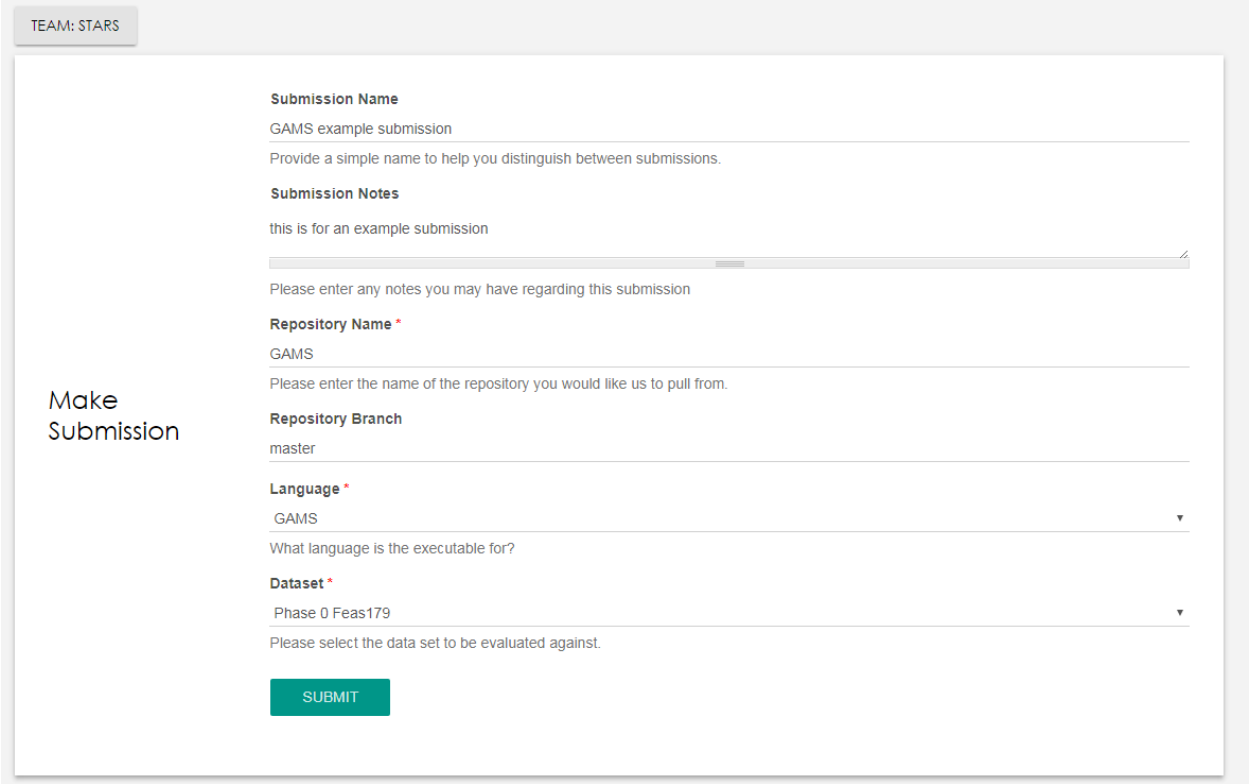
9. Click “Add deploy key” button on the right hand side of the page.



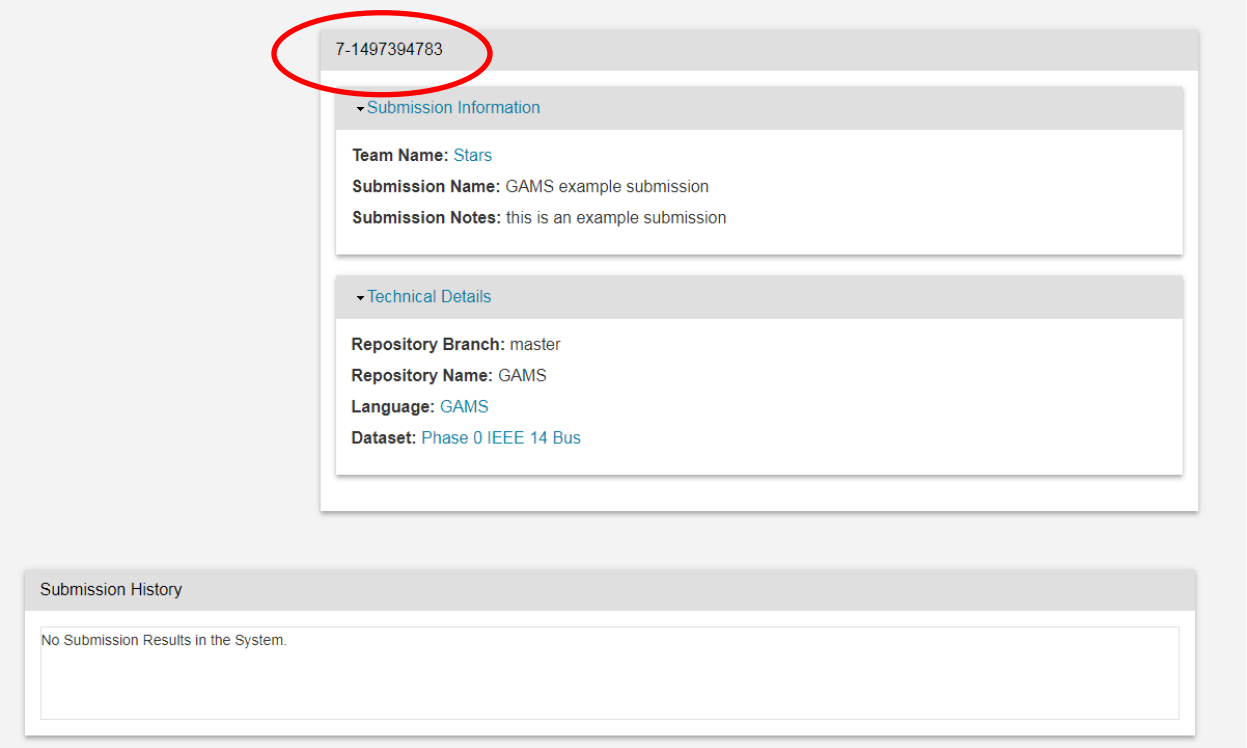
10. Enter a “Title” (e.g. GO Competition) and copy the SSH Public Key in the copy buffer (from step 3—no copying meanwhile) into the “Key” input field and click the green Add key button.



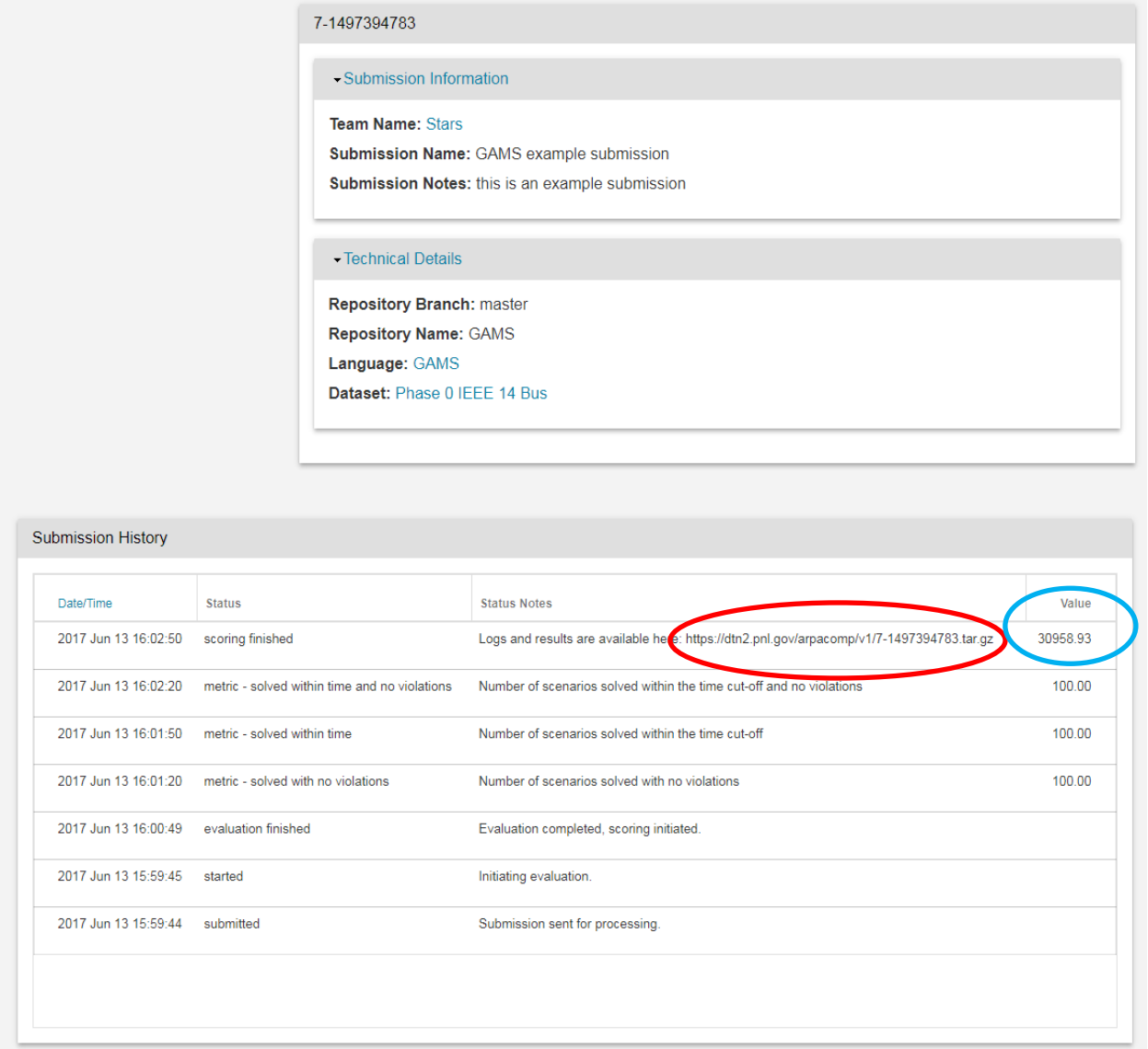
11. You may now return to the GO Competition Submission page, fill the information (Language and Dataset are selected from drop-down menus), and submit an algorithm for evaluation by clicking on the green SUBMIT button.



12. The resulting screen gives a unique SubmissionID (red oval) that is used to track the submission. The Submission History shows no results immediately after submission.



13. After evaluation and scoring have completed the submissionID page shows the various steps in the process. The final step gives the URL where the results may be downloaded (red oval) and the score received (blue oval).



14. The tar.gz file has a separate directory (output1…output*n*) for each scenario (*n* is100 for this example) plus scenario_results.csv and score.txt files.

15. The scenario_results.csv file contains the identification information, the parameters used in scoring, and a summary of results from each scenario: the scenario number and the official time in seconds. For GAMS runs that produced a solution0.txt file, the summary information from line 3 of solution0.txt for each scenario is copied to scenario_results.csv.

16. The score.txt file is also a csv file with the score, objective value (computed from the solution), the maximum feasibility violation, the associated Contingency ID (base case is 0), and the official time for each scenario. With the scoring parameters in scenario_results.csv and the values in score.txt, the computed score for each scenario can be verified. The score shown in the Submission History panel above is the geometric mean of each of the scenario scores.

17. Each of the output directories contains the solution1.txt and solution2.txt files used to evaluate the solution. Language dependent output and log files are also included. For the GAMS example used here these include log1.txt, submission.log, solution0.txt, and four GAMS lst files. See the Tutorial on submitting reference GAMS code for a complete explanation.