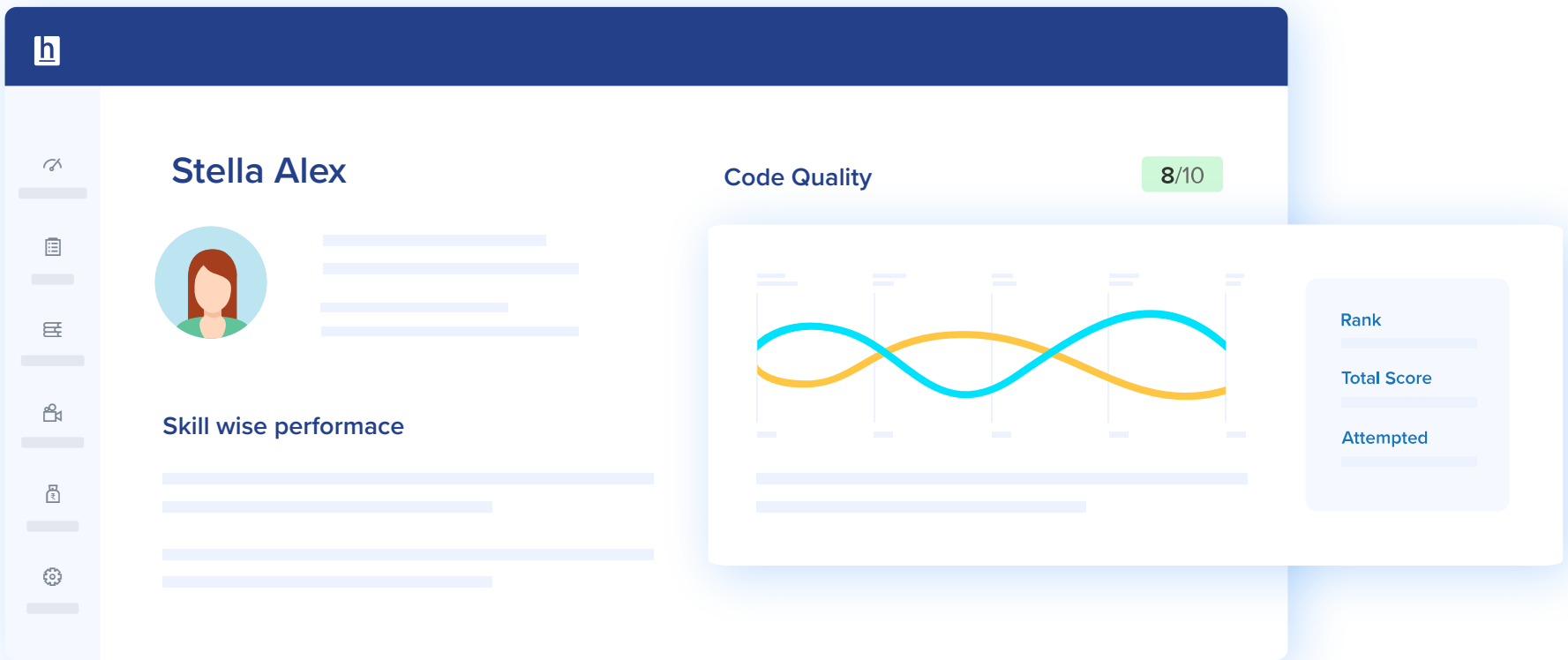
hackerearth

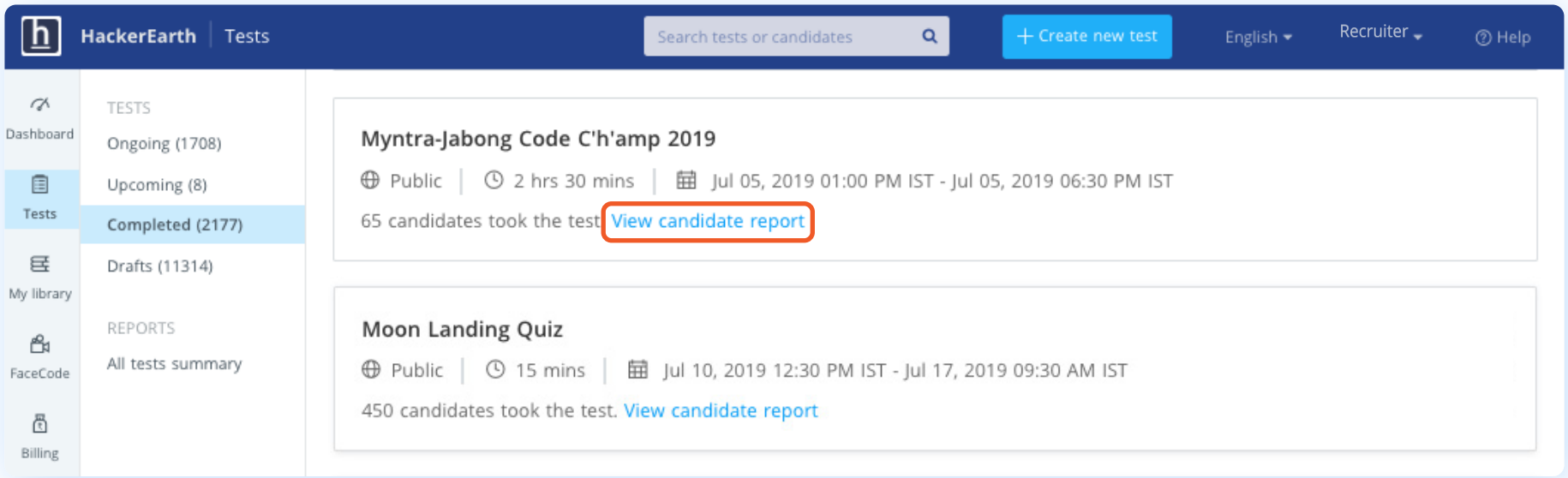


How to analyze a **candidate's performance** using HackerEarth

re

Derive performance insights using detailed reports

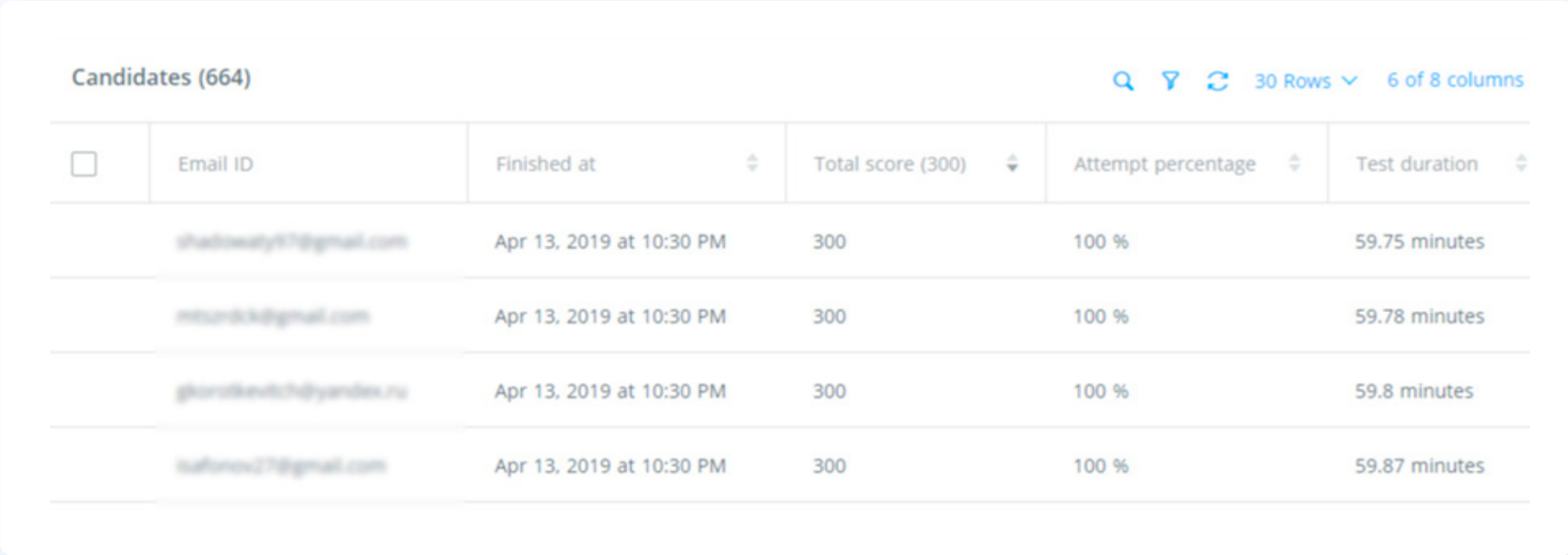
Once candidates have completed your test, it's now time to identify the best performers from the pool. The ideal way to do this is via performance reports. While candidate performance reports are the most convenient way to assess performance, yet, not all reports are intuitive or detailed enough to gather adequate insights.



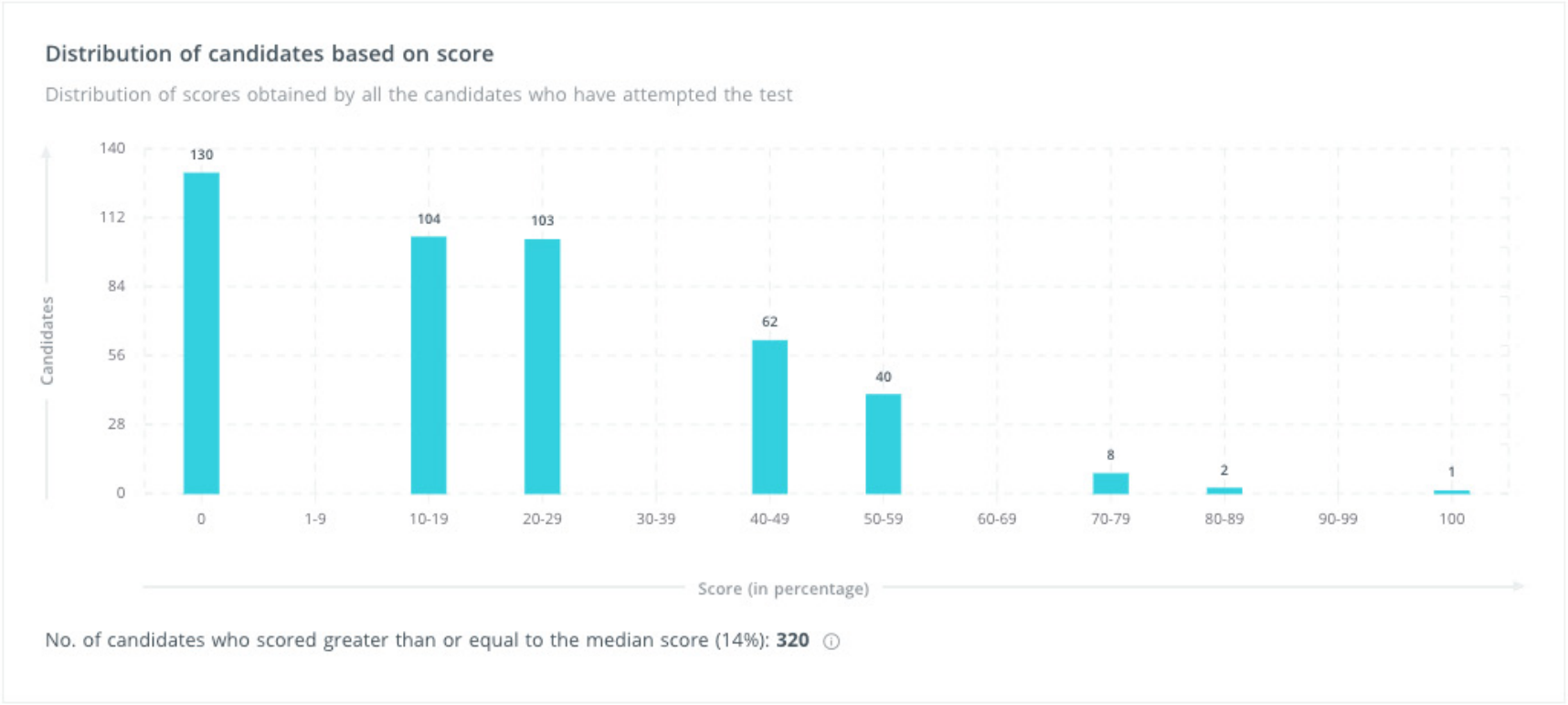
The reports created on the HackerEarth platform capture all the information necessary for you to precisely assess a candidate's performance in a matter of minutes. Here are a few reports to help you analyze a candidate's performance after they've completed a test.

1. Use the overall performance report to determine top scorers

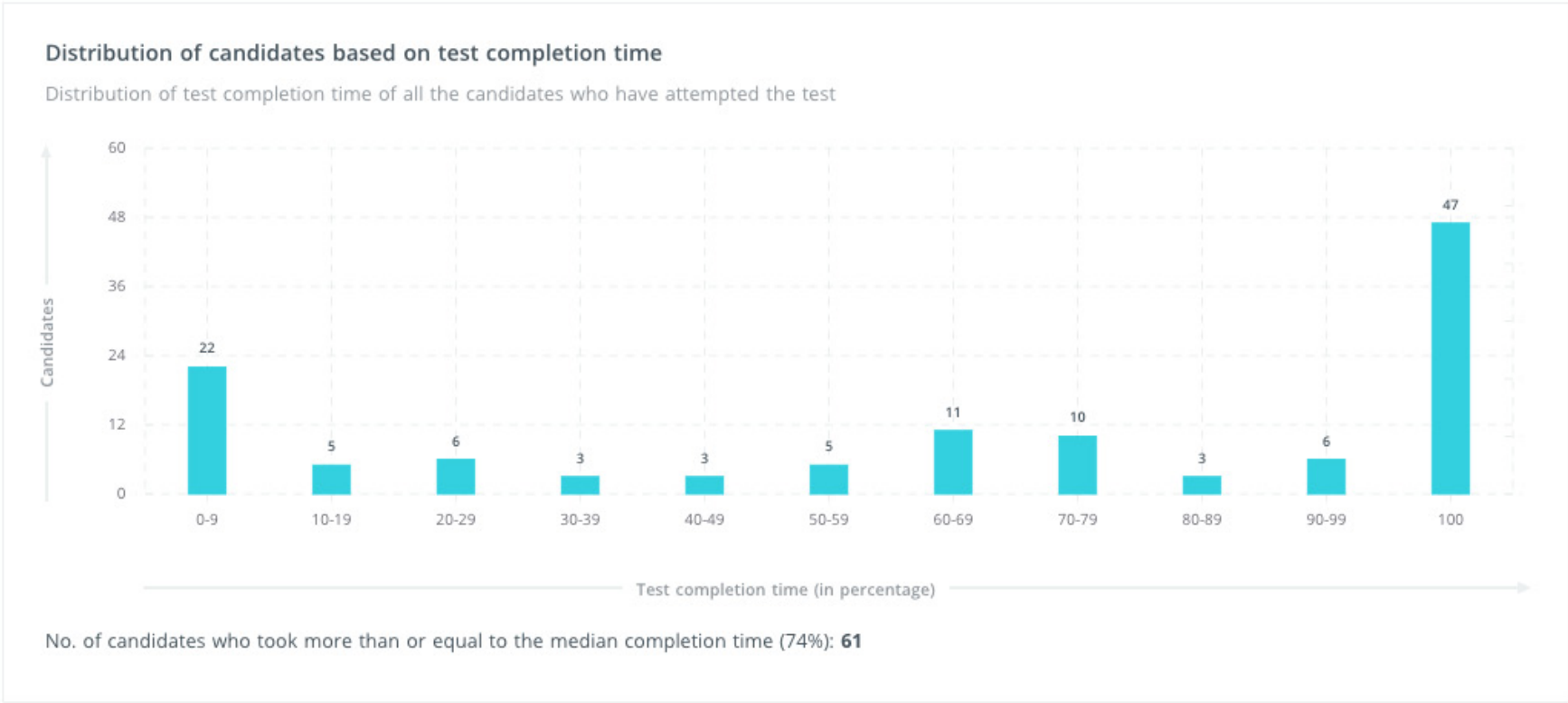
After candidates have completed a test, you can compare the overall performance of candidates by viewing their scores, attempt percentage, and test duration. In addition, you can filter candidates based on their experience or on factors such as total score, time taken to finish a test, or choice of programming language.



Pro tip: Additionally, you can view reports in the Analytics tab to get an overall distribution of candidates based on test scores and test completion time.



Here, 320 out of 450 (71%) candidates have scored marks higher than the median score, which indicates that the quality of candidates from this test was high.

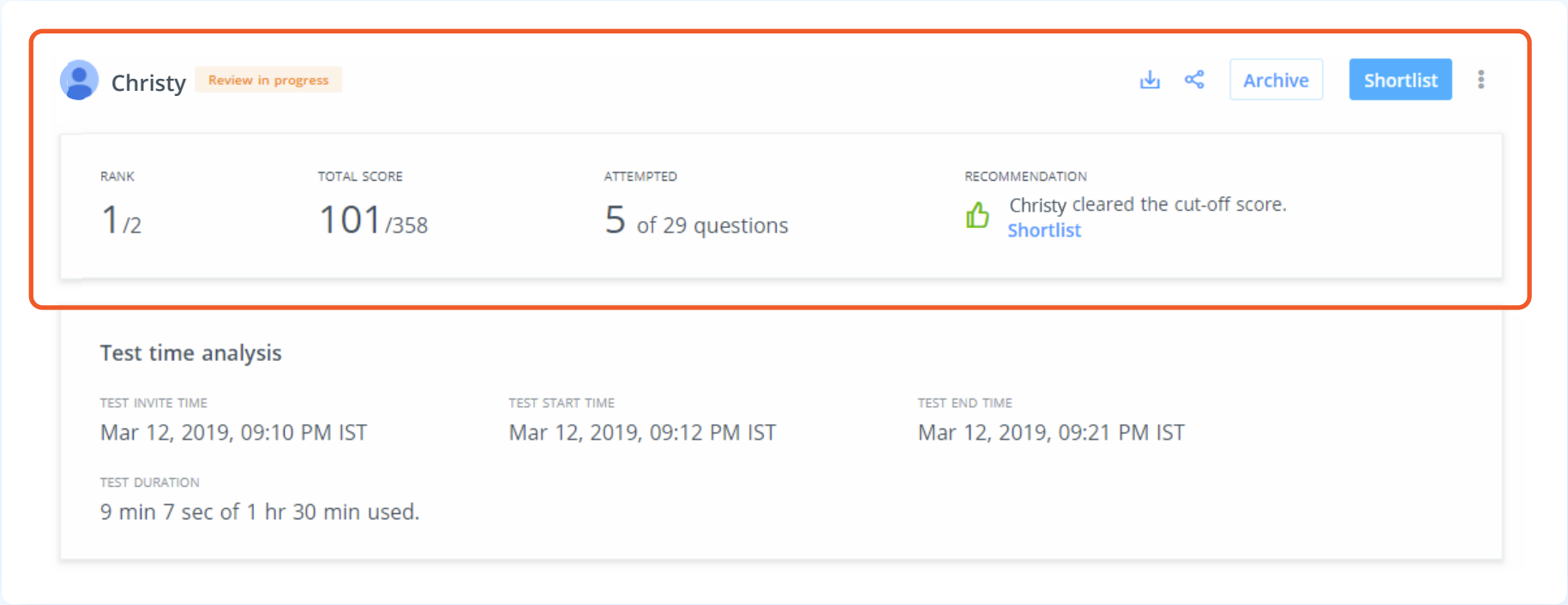


According to this chart, 61 out of 121 (50%) candidates have taken more than the median completion time to take the test.

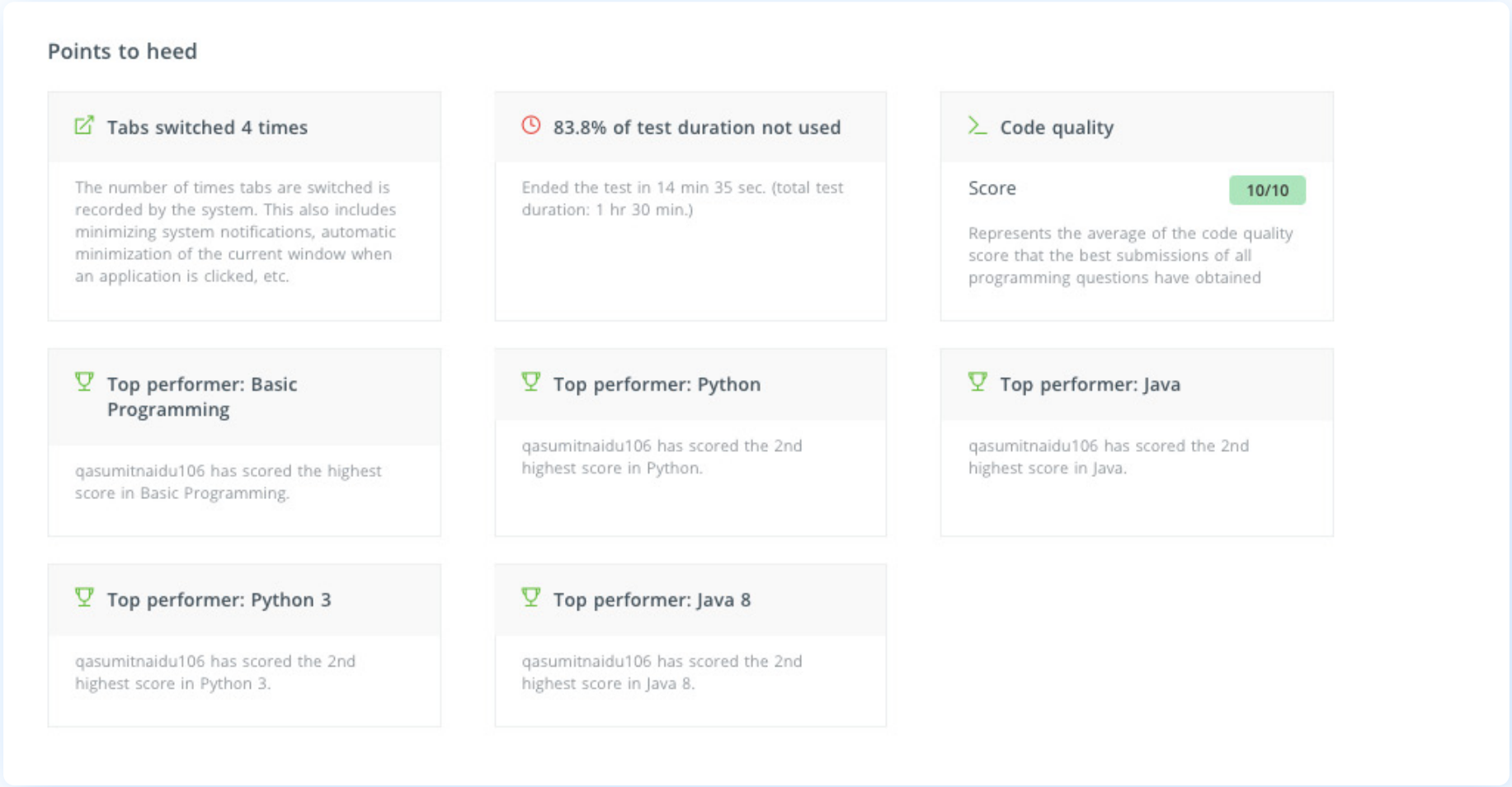
2. View individual reports to screen a candidate better

After completing an assessment, HackerEarth creates a performance report. Using this report, you can:

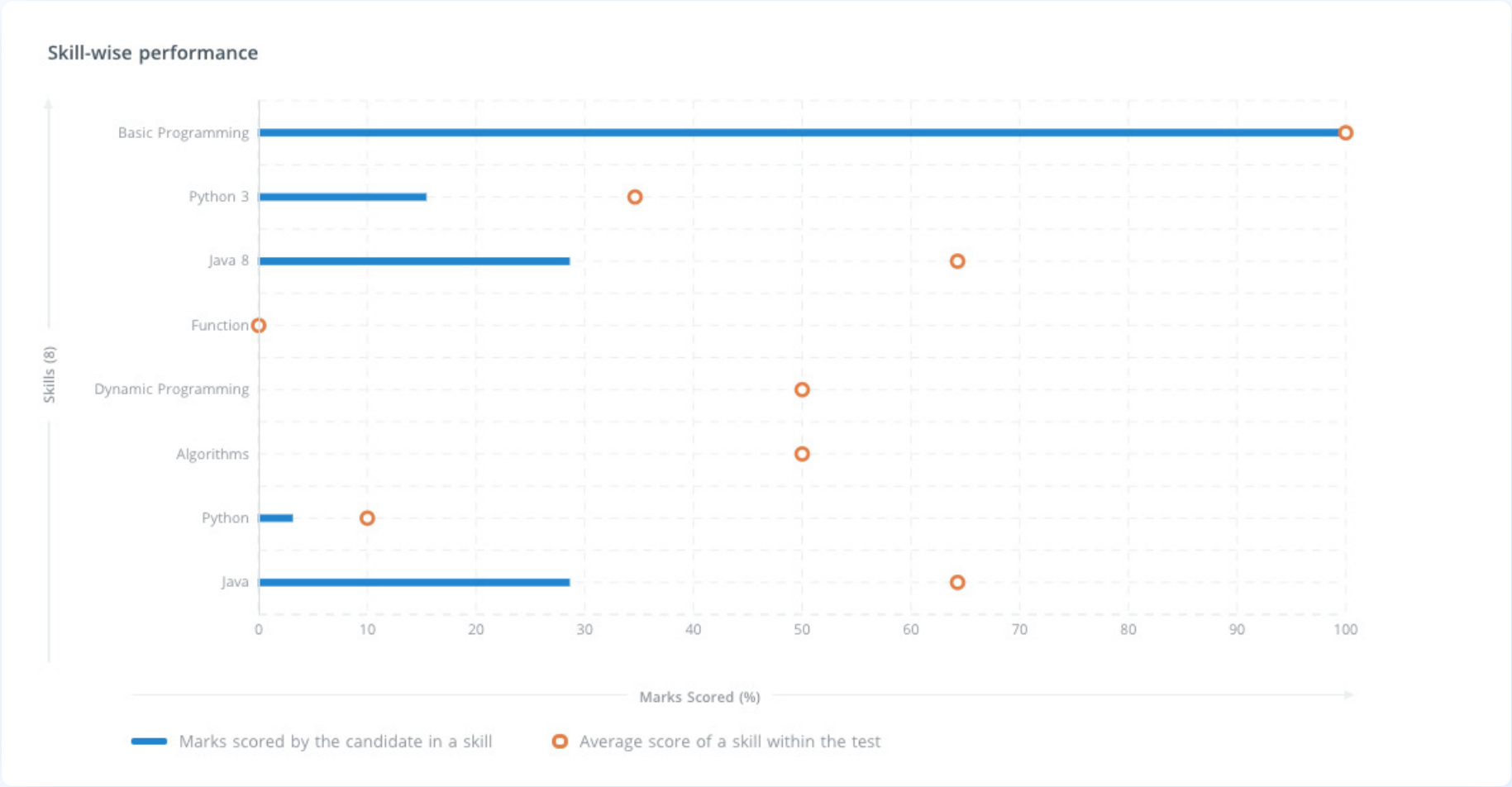
View a candidate's total score, rank, and number of questions attempted. Also, you can receive recommendations to shortlist or archive candidates based on their performance directly from the interface.



Track the number of tab switches, average code quality score (for all submissions in a test), time utilized, and the top skills of a candidate.



Check for compatibility by mapping the candidate's performance in each skill against the average score set for each skill, and determine if their skill sets match your requirement.



Get details on how a candidate has performed in each section of the test. You can also track code quality score for the best code submissions (for each programming question in the test).

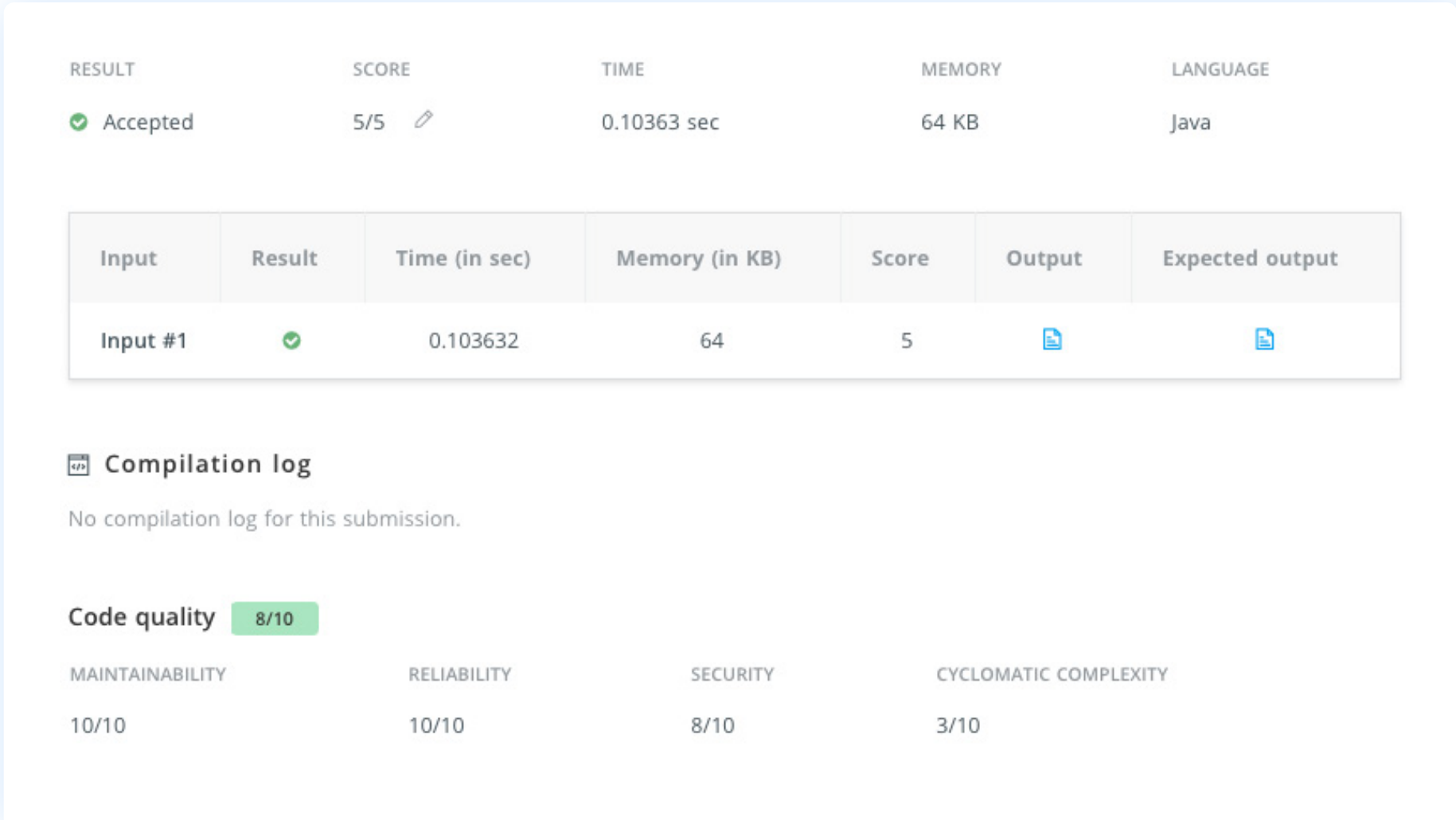
Programming Questions

Questions attempted: 5 of 5

#	Questions (5)	No. of attempts	Result	Code quality	Score (207/127)
>	1 Longest special subsequences	1	✓	8/10	5/5
>	2 Cube Square	1	✓	10/10	2/2
>	3 Test Problem	1	✗	—	0/10
>	4 Test Problem---Approx	1	✓	9/10	100/10
>	5 Test Problem -golf	1	✓	9/10	100/100

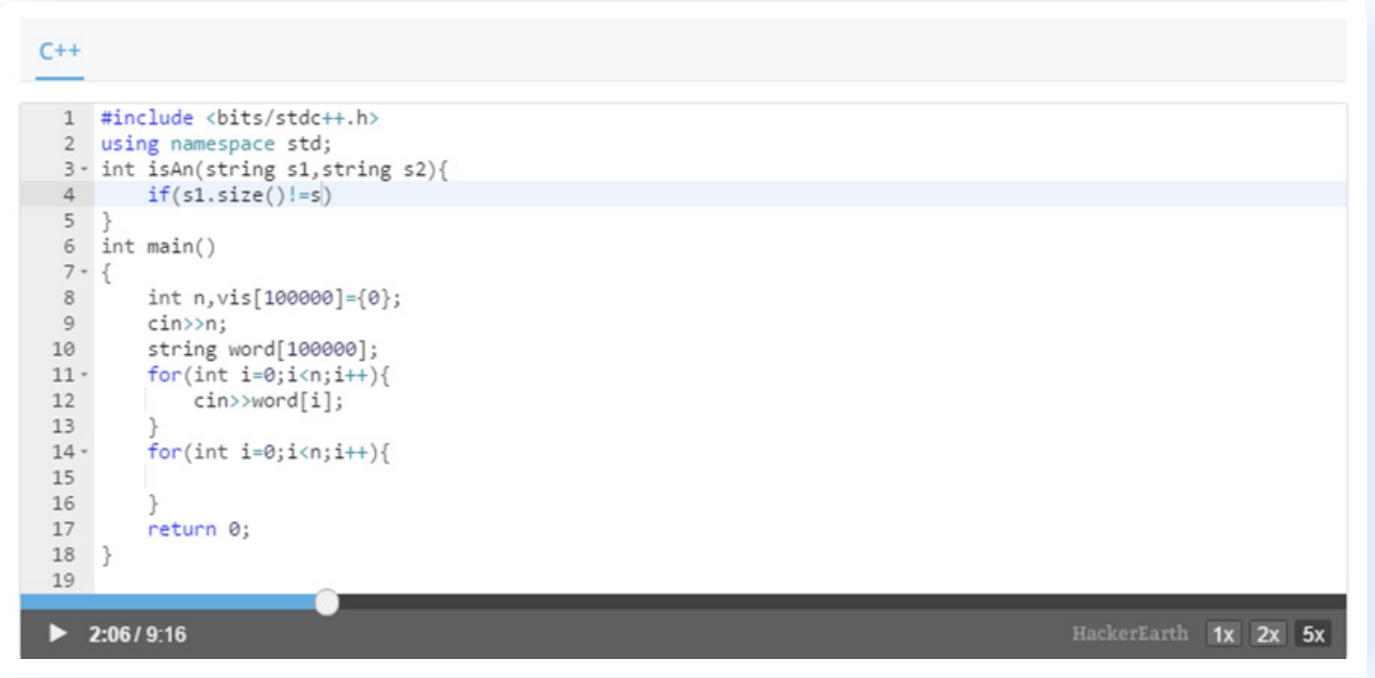
3. Review a candidate's code to analyze its quality

You can navigate to the 'details and code quality' section in the detailed submission report to analyze each code submission made by the candidate. Additionally, you can analyze the code on key parameters including readability, maintainability, security, and cyclomatic complexity. The code quality score which is measured on a scale of 0-10 (10 being the highest) is derived from the scores for each of these parameters.



4. Analyze a candidate's logical skills by replaying their code

Within the report, there's an option to replay a candidate's code as a video. This can help you analyze a candidate's logical and programming skills in detail and even use it as talking points during interviews.



Try using candidate analytics in your assessments and let us know how it works for you. If you need any help on using this feature, write back to us at **support@hackerearth.com**. If you're new to HackerEarth and want to create accurate skill-based developer assessments, **sign up for our 14-day free trial**.