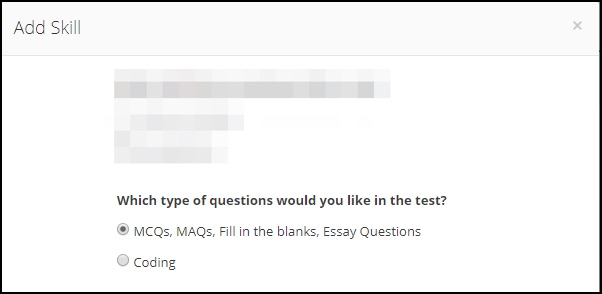
# Step 1: Select the Coding Test as a Type



No scoring will be done on Fill in Blanks or Essay Type Questions but the responses will be printed in the PDF as well as viewable online.

# Step 2: Select Configure Test Settings

* Enter Max Duration - 30 Mins - 3 Hours (in 30 Minutes Interval)
* Choose Difficulty Level - Easy, Medium, Hard

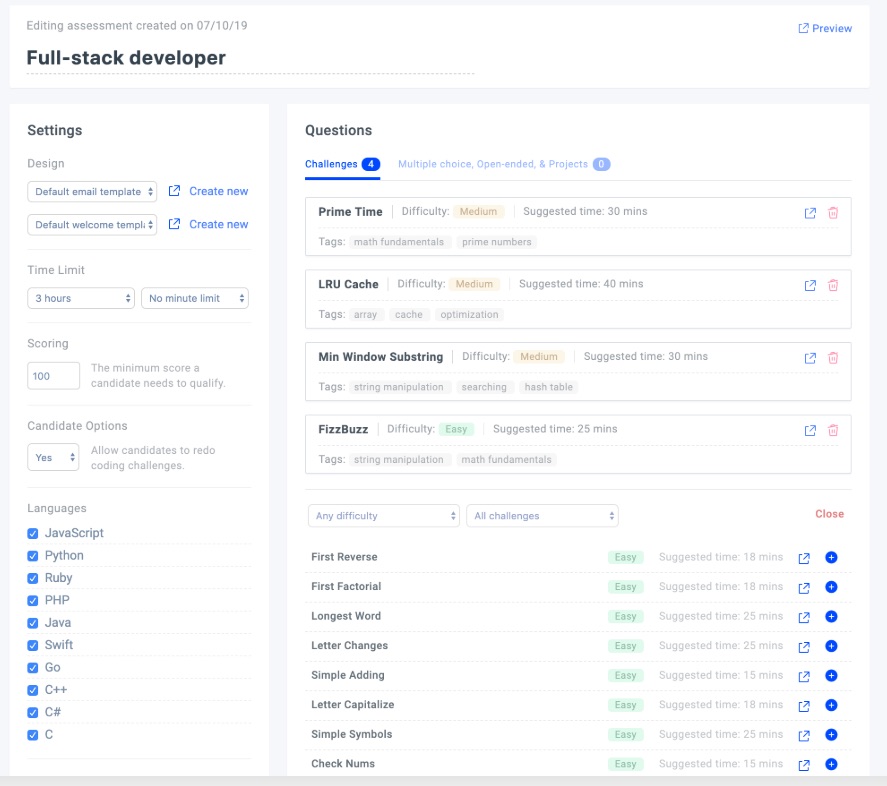
# Step 3: Select Programming Language

1. C
2. C++
3. C#
4. Java
5. JavaScript
6. Kotlin
7. Node.js
8. Objective-C
9. PHP
10. Perl
11. Python
12. Ruby
13. Swift
14. Go

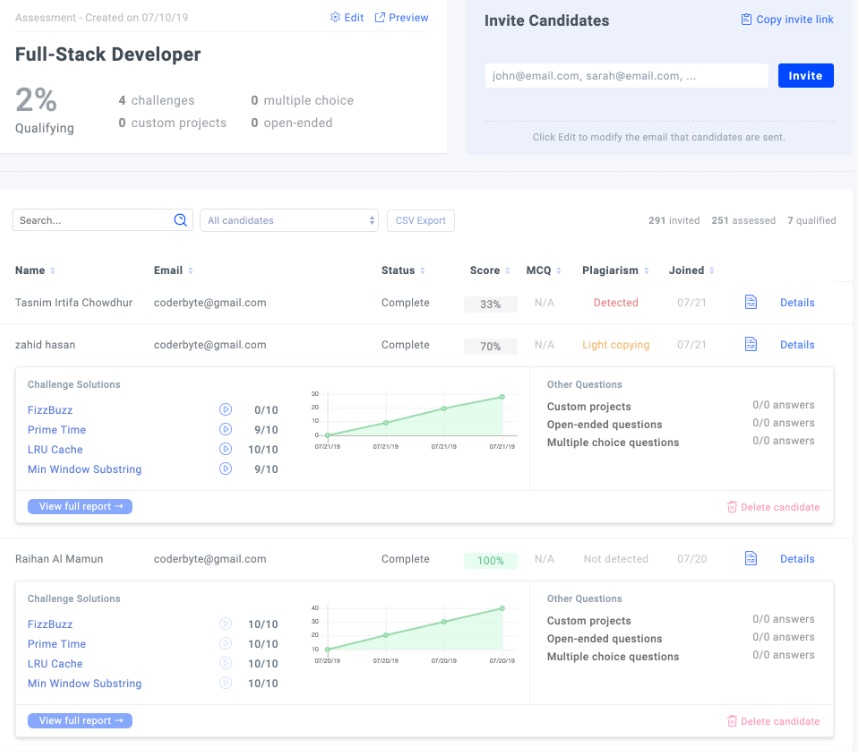
Step 4: Select Questions

Copy from <https://coderbyte.com/challenges> or other similar sources

### Sample Screen Design



# Step 4: Change in Reporting



The detailed reporting needs to be changed to <https://coderbyte.com/report/user07671111:intership-test-3-ag-qffbgrvpoq>

Features Needed:

1. Automatic scoring
2. Code Plagiarism check from <https://codequiry.com/>
3. Integration with Sphere Engine
4. The test interface for coding test should allow users to copy and paste code from their favorite IDE’S.

# Integration with Sphere Engine

## The solution to our requirements as offered by Sphere

● Sphere Engine Compilers module:

1. Allowing compilation and execution of 80+ programming languages
2. API allowing for highly customizable integration
3. Widgets allowing quick integration with dedicated UI

● Sphere Engine Problems module:

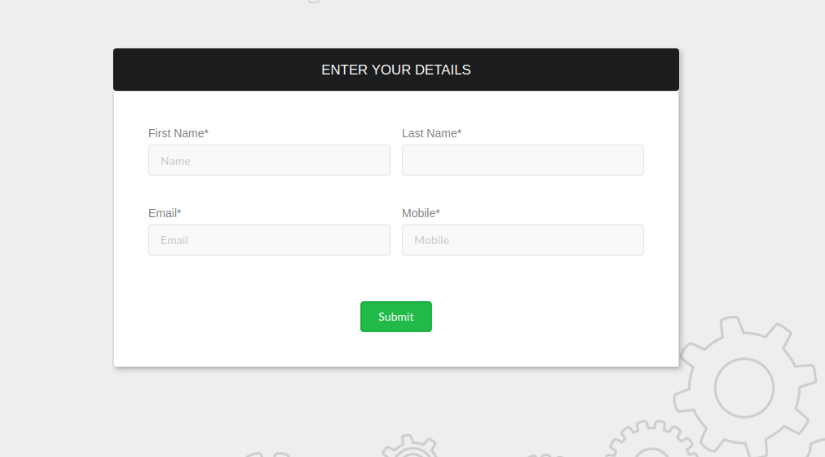
1. Allowing the assessment of the correctness and quality of the source code written in 80+ programming languages
2. Problem Editor for designing and managing programming challenges
3. API allowing for highly customizable integration
4. Widgets allowing quick integration with dedicated UI

## Next Steps

1. Please signup for an account over here at <https://sphere-engine.com/signup>
2. I would first recommend you to get familiar with the introduction to Sphere Engine, which also covers differences between the Compilers and the Problems modules: <https://developer.sphere-engine.com/introduction>
3. If you would like to test the Problems module and figure out how it works on your own, you can go to <https://sphere-engine.com/services/problems/editor#/problem/SETEST>, click "Clone" button and play with the cloned problem and its options. BTW, you will find a solution to this problem here: <http://ideone.com/samples#sample_lang_116>
4. You may also want to check out our demo <https://sphere-engine.com/demo> to get the feel of in what ways the Sphere Engine can be employed when it comes to IT skills assessment
5. If you plan to use the Sphere Engine Problems module then I also strongly recommend familiarizing yourself with our comprehensive handbook on creating programming challenges: <https://developer.sphere-engine.com/other/handbook> (FYI: English in the handbook is slightly rough around the edges.
6. You will find many more useful articles here in our documentation: <https://developer.sphere-engine.com/>

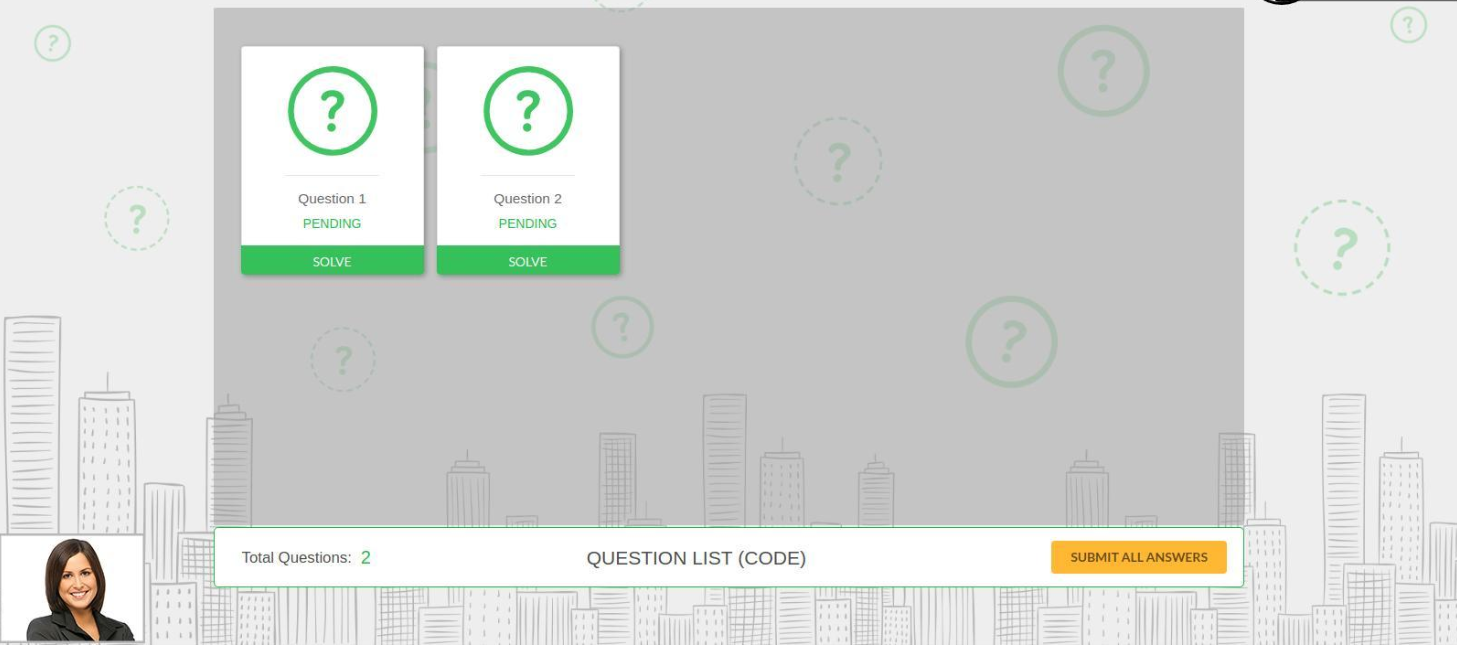
# Coding Test - Developer Perspective

**Step 1**: If the candidate is invited by Public URL otherwise this data is auto-filled

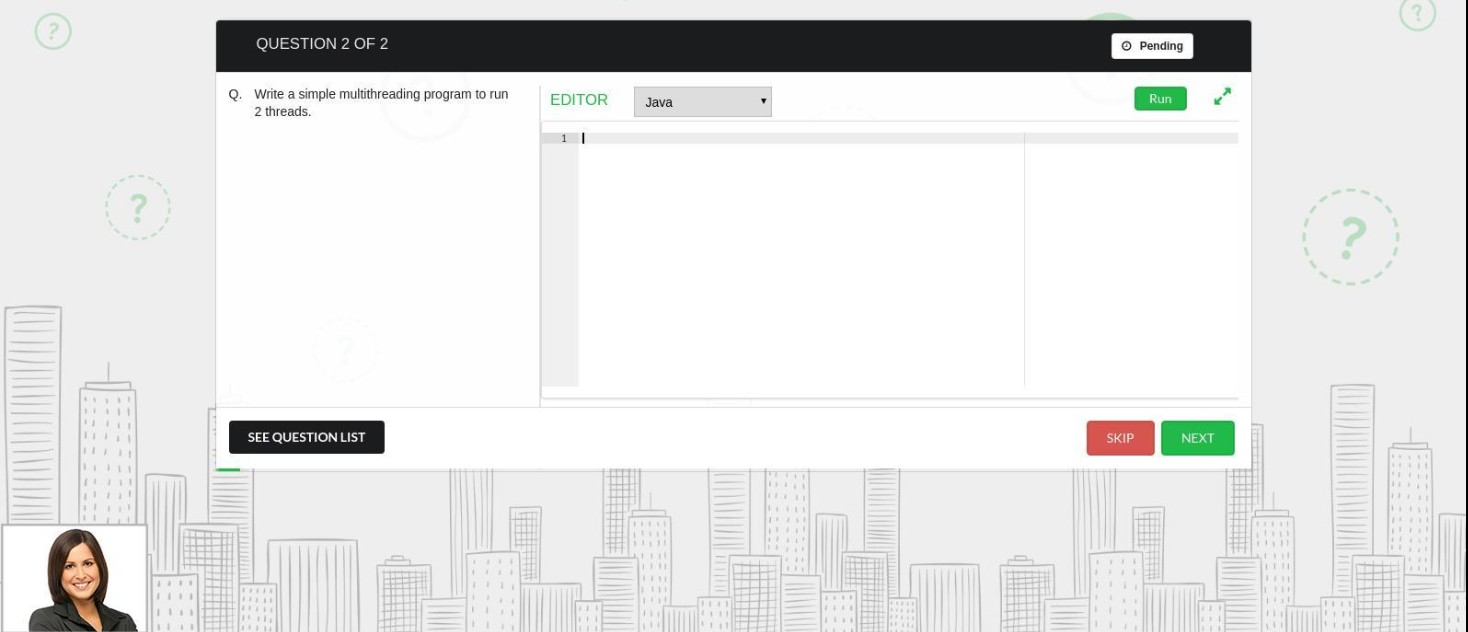


**Attending the Coding Assessment**

* Once they enter the assessment page, they can see the list of questions.
* You can click on 'Solve' and attend the question of your choice



* They will reach the screen below when you start answering a question. The question is visible on the left of the screen and the space to type down your answers is on the right. The question timer will be on the top right corner of the screen to help you keep track of time, with the 'Skip' and 'Submit' options in the bottom-right corner of the screen.
* They can run your program by clicking on 'Run' option in the top-right corner of the screen.



* Once they have answered all the questions, click on the 'Submit All Answers' button. Their responses are saved in the system successfully and they will see the screen shown below.

