Word Game

Your deliverable should be a distributed source code repository (prefer git or mercurial), some source code in said repo, build instructions, and deployment instructions if necessary. If it matters for your deployment instructions the target production platform is Ubuntu. We should be able to run your solution.

Return your response by email as a .zip or .tar.gz archive to SoftwareInterviewTest@ClearCorrect.com. All descriptions and comments should be in English. If you have questions about this assignment, please feel free to email SoftwareInterviewTest@ClearCorrect.com for clarification.

Requirements

You are responsible for implementing the word-checking part of a Scrabble-like board game. You will need to create a board structure as well as provide a mechanism by which a player can add letters to the board of 19x19 square spaces. Each time a word is added, your solution should check the validity of the placed letters using the following rules:

- Letters must be played in a straight line, up-down or left-right.
- The first word can be played anywhere on the board.
- All subsequent words must share at least one space with an existing word.
- Word direction can be left-to-right or top-to-bottom.
- All sets of adjacent letters must form valid words.

You may assume a word validating class is available with the following interface:

```
class WordValidator
{
public:
     virtual ~WordValidator() = default;
     virtual bool isValid(const std::string &word) const = 0;
};
```

You should write your code in c++ and it should compile. You are free to use whatever build system and unit testing framework you like. We suggest writing a main function that exercises your system. You may use external libraries (such as STL and boost), but our goal is to evaluate *your* ability. Expect that the problem may take you several hours to complete, once started.

Could Have

- A command line build via some popular build tool, e.g. Ant, Maven, Gradle, Pants, Buck, Bazel, etc. FYI, we currently use Gradle.
- A WordValidator using some word list such as CSW, OSPD, SOWPODS, or TWL
- Support for multiple board sizes
- Some kind of scoring system