

15 puzzle

The goal of this task is to create a 15 puzzle. A 15 puzzle is a sliding puzzle that consists of a frame of numbered square tiles in random order with one tile missing.

We expect this task will take around 6 hours.

The puzzle must be implemented the following way:

1. On page load tiles order should be randomised.
2. The board size (number of columns and rows) should be configurable, and support a rectangular configuration (M x N board size)
3. The tile size should be configurable.
4. Must be performant for large grid (e.g above 30x30).
5. Should support adding multiple boards.
6. Validity checks. Every board should be solvable.
7. A success message should be displayed when all the elements are reordered properly.
8. Tiles should be animated and slide to the designated area (Nice to have).

UI Requirements

Please implement the simple UI wireframes outlined below. Minimal styling is acceptable.

Constraints

You may:

- Develop only for Chrome.
- Use any feature available in the latest stable release of Chrome.
- If you don't have time to implement the optional animation component, your code should still be written in a way that it could potentially support it without major refactoring.
- Use CSS animation for tiles animation.
- Class-based architecture (ES6 classes preferred).
- Assume that your code will be served from a simple web server (e.g. `serve` from npm), to avoid CORS errors that may otherwise occur when loading scripts/assets from the filesystem

You must not:

- Use any build tools or preprocessors (npm, webpack, PostCSS, etc.)
- Use any frameworks or libraries (Angular, jQuery, React, etc.)

Marking Criteria

Your code should be clear and easy to understand:

- Avoids unnecessary complexity / over-engineering

- Brief comments are added where appropriate
- Broken into logical chunks
- Follows a module pattern

Your code should be performant:

- Gives feedback to the user as soon as possible (perceived performance)
- UI remains responsive

Example:

4	2	7	1
6	8		3
10	5	15	12
14	11	13	9