Validating solution:

* The first rule of Hitori state that’s No number can appear in a row or column more than once, to make sure our solution does not violate this rule we can easily implement a predicate that check if a row or a column have at least two elements repeated and the two of them is not shaded. Then for each row and column we will call this predicate; If the predicate failed to find such two elements then the row or the column has distinct number and hence our solution is fulfilling the rule.
* Shaded cells do not touch each other vertically or horizontally.
* all un-shaded (white) cells create a single continuous area. Let be an arbitrary white cell then we can do a simple depth first search from it to visit all reachable white cells. Let be the count of visited cell, B is the count of shaded cell and N, M is the dimensions of the grid then it easy to see that if our solution fulfils the rule.

Solving the board:

We can easily implement a naïve backtracking approach by trying to shade every cell on the board. Such approach with overall time complexity is really bad.

Optimization and playing techniques:

Basic facts:

1. The second rule of Hitori says shaded squares must not touch each other vertically or horizontally. Implying that a shaded square is always surrounded by four un-shaded squares (or less, depending on whether it is located next to the edge or in a corner).
2. let cell x be un-shaded cell at row R and column C then to avoid conflict with the first rule we must shade all cells which have the same value of x in R and C
3. What it's true is that a cell cannot be black and white simultaneously, obviously.

Starting techniques:

let cells x, y, z be an adjacent triple(xyz) such that . it is not hard to see that we must shade two cells to make sure we don’t violate the first rule of the game. We have three possible ways of shading: x and y,  
y and z, x and z. the first two will cause a violation in the second rule therefore we have only one valid option: shading x, z and mark a circle around y to indicate it must remain un-shaded.

Let’s try to generalize this technique, let cells x, y, z be an adjacent triple(xyz) such that

. at this situation we must shade either x or z, then by induction and using the basic facts: y always will be un-shaded.

Corner techniques:

Let cell x be a corner cell on the board and let cells y, z be its neighbors such that .

if x is un-shaded then y and z must be shaded according to the first rule of Hitori. However, by shading cells y and z, a wall is formed in the corner which partitions the un-shaded cell x from the rest of the unshaded area. Therefore, x must be shaded and y, z must be un-shaded.

Advanced techniques:

After solving a few boards, we can easily observe that if the board is solvable then we can solve it by considering rows and columns which have a repeated element only.

Let’s return to our naïve approach, so the main idea is assumption and conflict, with the previous observation we have reduced the search space to where k is the count of repeated elements which still a bad approach.

Let us assume that we are trying to shade a cell x, so if we used the basic facts to infer other cells before we go down the backtracking search tree we can reduce the search space further if we had a conflict with the current game state (shading an un-shaded cell and vice versa, or when we have partitioned the board).

Since x is a repeated element, the following holds in the current game state:

If we had a conflict then cell x must be un-shaded.

So again, if we had a conflict with the current game state then the state is not valid and we must force backtracking, yet again we have reduced the search space.

Calculating the exact time complexity of such approach is extremely difficult but it will give a good result on the average case.