Chess

Vague, high-level problem statement, as delivered in an interview: Design/Model the game of Chess

What that typically means: Design a library that implements Chess. Developers using the library (i.e. your main() method) will initialize the chess-board with pieces and players, and call makeMove() function on the relevant entities. Design it such that the developer cannot make an invalid move and the library lets them know if a given move wins the game.

Deliverables:

1. A set of classes, showing relationships with each other where appropriate. Classes should show state and methods. Use any convenient notation.

2. Main() method, showing how you'll initialize your system and use it.

3. (Optional, only if it helps bolster understanding): A flow chart of main use-cases and a state-diagram.

Please put everything in one single file and upload it. It could be a Doc file, online GDoc (converted to PDF), a powerpoint, a spreadsheet, a Visio file, whatever. Just have it all in there.

Possible directions for further questions:

1. What assumptions are you making, and how will your design change, if those assumptions change?

2. If you provide this library as a service, how will you scale it?