CORE SYSTEMS

WINDOW
Initialize, resize, close; frame pacing / FPS cap.

GAME LOOP
Input → update → render; fixed/semi-fixed Δt.

STATE
Menu / playing / ended; new game / reset.

BOARD & RENDERING

MODEL
8×8 grid; square colors; coords; (opt.) turn/half-move counters.

BOARD
Draw light/dark; last-move trace; selection/highlight.

PIECES
Sprite/SVG; clean layering; hover/selection affordances.

MOVEMENT RULES (PER PIECE)

- **King**: one square any direction; never into/through check.
- **Queen**: slider (rook + bishop vectors).
- **Rook**: orthogonal slider; blocked by occupants.
- **Bishop**: diagonal slider; blocked by occupants.
- **Knight**: L-jump; ignores blockers.
- **Pawn**: one step; optional two-step from start; diagonal capture.

TURN & INPUT

FLOW
Select piece → show legal targets → commit/cancel.

CHANGE TURN
White ↔ Black after a legal move.

ILLEGAL MOVE
Reject with concise feedback; preserve state.

SPECIAL RULES

- **Castling**: king/rook unmoved; clear path; not in/through/into check.
- **En passant**: track EP square after pawn double; capture on immediate reply.
- **Promotion**: on last rank to $Q/R/B/N$; deterministic default (e.g., Q).

LEGALITY & THREATS

CHECK DETECTION
Is side-to-move king attacked?

SELF-CHECK FILTER
Remove pseudo-legal moves that expose king.

GUARDS
Off-board; friendly-occupied; path-blocked.

END STATES

CHECKMATE
In check *and* no legal moves.

STALEMATE
Not in check *and* no legal moves.

DRAWS (OPT.)
Insufficient material / 50-move / repetition.

QUALITY OF LIFE (OPT.)

- Move list (SAN/LAN), undo/redo, quick reset; themes.
- Test hooks; load/save positions (e.g., FEN).