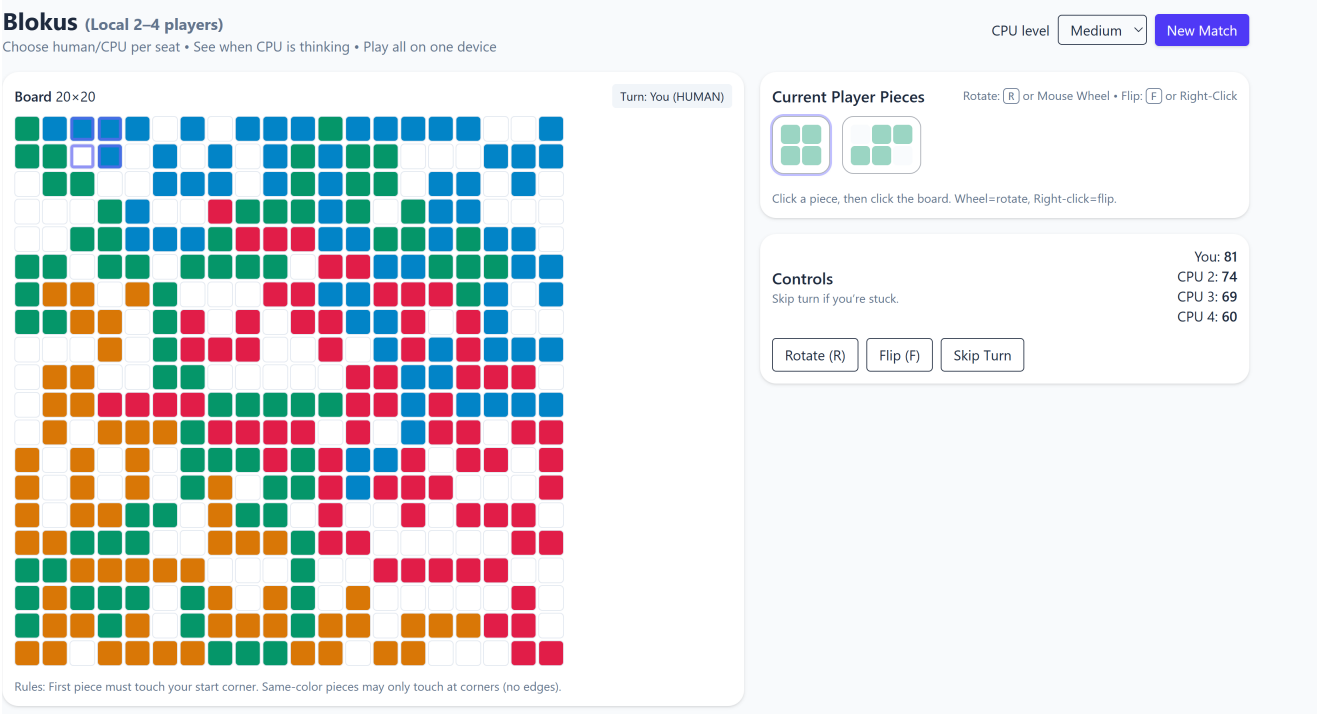
Blokus Game

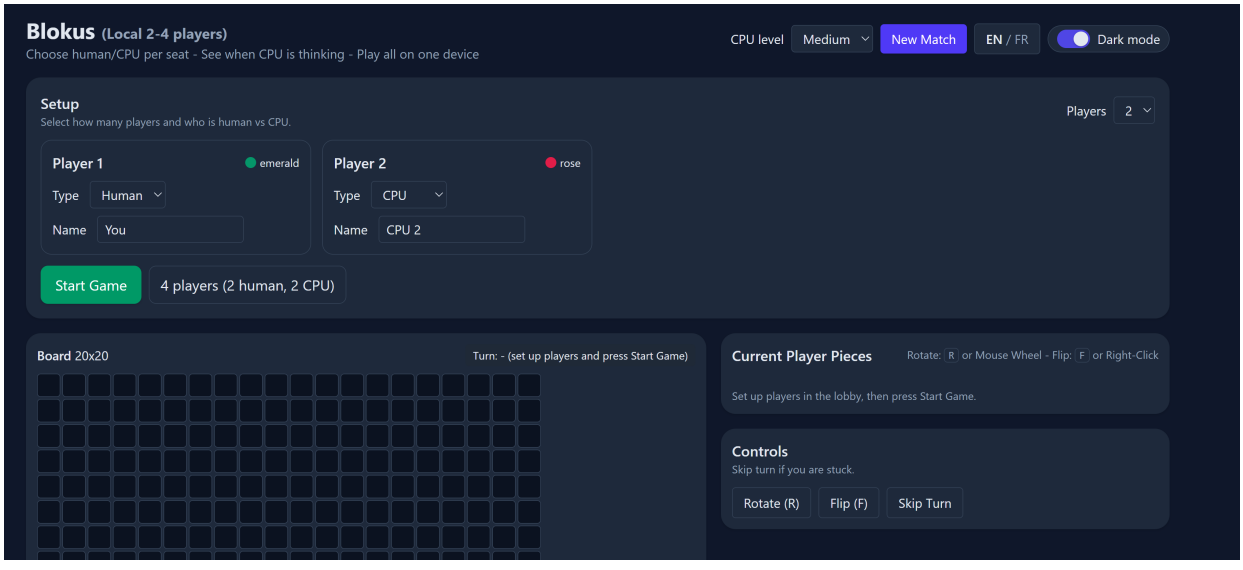
A lightweight, browser-only Blokus-style game. Play locally with up to 4 players, mixing humans and CPUs, directly from a single HTML file.

> **Work in progress:** This project is actively developed and may change without notice.



What you can do

- **2–4 players** on one device.
- Mix **Human** and **CPU** seats.
- **CPU difficulty**: Easy / Medium / Hard.
- **Full 21-piece set per player** (standard Blokus).
- **Dark mode** toggle available in-game .



- **Mouse & keyboard controls**:
  - Rotate piece: **R** or **mouse wheel** (trackpad scroll).
  - Flip piece: **F** or **right-click**.
- **Scoring** displayed live.
- **End-of-game overlay**: “Good! You win / You lose / Tie”.
- Works entirely **offline**—no server required.

How to start the game

```
git clone https://github.com/PiWebswiss/Blokus-Game.git
cd Blokus-Game
```

Docker (recommended)

```
docker compose up -d
```

Open `http://localhost:8080`.

To stop the container:

```
docker compose down
```

Without Docker

Double-click `index.html` to run the game.

Quick start a game

- Choose **Players** (2–4).
- For each seat, pick **Human** or **CPU** and set a name.
- Optionally pick the **CPU level** in the top-right.
- Click **Start Game**.

How to play

Goal

Place as many of your pieces on the 20×20 board as you can. Your **score** is the number of squares you’ve placed.

Turn order

Players take turns. The current player is shown next to **Turn**.

Rules (Blokus-like)

- **First move**: must cover **your start corner**:
  - Player 1 → top-left (0,0)
  - Player 2 → bottom-right (19,19)
  - Player 3 → bottom-left (0,19)
  - Player 4 → top-right (19,0)
- **After the first move**:
  - Your new piece must **touch at least one of your existing pieces by a corner**.
  - **Edge-to-edge contact with your own pieces is NOT allowed.**
- You can **rotate** or **flip** a piece before placing it.
- If you can't move, click **Skip Turn**.

Controls

- **Select a piece**: click it from your palette on the right.
- **Preview on board**: move your cursor over the board; valid cells are highlighted.
- **Place on board**: click a cell.
- **Rotate**: press **R** or use **mouse wheel / trackpad scroll**.
- **Flip**: press **F** or **right-click**.
- **Skip**: click **Skip Turn**.
- **New match**: click **New Match** (top-right).

> CPU turns show a short "thinking…" overlay.

CPU levels

- **Easy**: random legal move.
- **Medium**: prefers larger pieces.
- **Hard**: heuristic mix: piece size, board centrality, and future mobility.

Project files

- `index.html`
  Main page and UI layout.
- `src/js/app.js`
  Single JavaScript entrypoint: translations, game logic, CPU, rendering, and input handling.
- `output.css`
  Tailwind build CSS.

Notes

- This project keeps both files on purpose: `Dockerfile` defines how the app image is built.
- `docker-compose.yml` defines how the container is run (ports, name, restart policy).
- If you prefer using only `Dockerfile` (no Compose), you can run manually:

```
docker build -t blokus-game .
docker run -d --name blokus-game -p 8080:80 blokus-game
```

- Stop/remove that manual container with:

```
docker stop blokus-game
docker rm blokus-game
```

- Rebuild the Docker image only when needed (for example after changing `Dockerfile` or static files):

```
docker compose up -d --build
```

ChatGPT links

- chat 1
- chat 2