# Physics Sandbox: Interactive 2D Physics Simulation

Hemant Godve

2025

**Abstract**

This report presents the development and features of the **Physics Sandbox** project, an interactive browser-based 2D physics simulation built using the Matter.js JavaScript library. The sandbox allows users to experiment with basic physics concepts by adding, dragging, and manipulating geometric shapes in real-time.

## 1 Introduction

The Physics Sandbox aims to provide an accessible platform for visualizing and experimenting with fundamental concepts of classical mechanics. By leveraging web technologies and a modern physics engine, the project enables interactive and engaging learning experiences for users of all ages.

## 2 Objectives

- Create an intuitive and interactive environment for physics simulation in the browser.
- Demonstrate the use of the Matter.js library for simulating gravity, collisions, and object manipulation.
- Offer a responsive, visually appealing, and user-friendly interface.

## 3 Technologies Used

- **Matter.js**: For 2D physics simulation and rendering.
- **TailwindCSS**: For responsive, modern styling.
- **Custom CSS**: For further enhancements and dark-themed backgrounds.

## 4 Features

- **Interactive Canvas**: Add, remove, and drag shapes (circles, boxes, rectangles, triangles, pentagons).

- **Realistic Physics**: Simulated gravity, restitution, friction, and collision detection.
- **Modern UI**: Button controls for adding shapes and clearing the canvas.
- **Responsive Design**: Adapts smoothly to window resizing.
- **Randomized Visuals**: Shapes rendered in distinctive random colors.

## 5 Implementation

The main simulation is powered by Matter.js modules for engine, rendering, world, and input constraints. Shapes are instantiated and added to the simulation with randomized colors and slight randomization in positioning.

A summary of the files included:

- `sandbox.html`: Provides UI controls and the canvas container.
- `sandbox.js`: Initializes physics modules, sets up event handlers, and manages simulation updates.
- `sandbox.css`: With TailwindCSS for dark-themed, responsive styling.

Canvas and boundary walls update dynamically on browser resize, keeping all physics interactions contained.

## 6 Results

- **Successful Simulation**: Users can intuitively add shapes, observe interactions, and manipulate them in real-time.
- **Physics Effects**: Realistic gravity, collisions, and movement, as expected from a physics sandbox.
- **Accessible Design**: Web-based, no installation required, works across modern browsers.

## 7 Limitations

- Limited to 2D with basic control; no advanced tuning (mass, angular momentum, custom forces).
- No export or save/load options for sandbox states.
- Basic visual effects, with scope for richer graphics or analytics.

## 8 Future Improvements

- Add advanced physical properties, shape rotation, and simulation settings (sliders for gravity, friction).
- Add data export, state saving, and session restoration.
- Enhance UI (tooltips, object selection, delete single shape).
- Add visual analytics (graphs, charts for velocity, collisions).

## 9 Conclusion

The Physics Sandbox demonstrates the power of web-based physics simulation using modern JavaScript libraries. It is an effective tool for education, demonstration, and further extension into more complex simulations or game mechanics.

## Appendix

Files included: `sandbox.html`, `sandbox.js`, `sandbox.css`
**How to Use:** Open `sandbox.html` in your browser.

**Author:** Hemant Godve
**Date:** 2025