MVP on Github Pages (compile webpack, then host as normal site? Idk if possible, using node requires

Plugins to look at: matter-attractors by liabru

For springs: matter-springs by momentumworks

matter-collision-events by dxu

Cool examples (like airFriction, constraints, events, gravity, etc): <https://github.com/liabru/matter-js/tree/master/examples>

On looks:

smooth grass(fricCoeff = 0) & ground (visible rectangle/sprite of groudn (fricCoeff > 0);

Collision filtering to turn off/on objects?

Logistics:

Max user input (ex no higher than 400N, 15 objects, etc)

* Dynamic graph drawing of velocity and position
* Use an SVG to make a concave pit and other stuff

Objects (let user set w, h, r, etc): Box; Car; Disk (different sizes, radii, lengths, and hollow vs dense; since u can set mOinertia, u can have user set the 2 radii, u draw it w the 2 radii, then do the calculatiosn behind the scenes to get mOi), Sphere, pendulum, 2 masses w light cord on frictionless pulley

Free body diagram: Always displaying forces that will act on the object on “Play”; FBD is ON the object in the screen; how to display Fforce on FBD?

On play: Constant forces stay on there (color coded), instant forces disappear, but velocity vectors are shown (magnitude, direction)

Forces: Can select where on object to apply force (exactly what point, direction, and magnitude)

(torque applied by perp force)

Gravity, air resistance, restitution, friction (static, kinetic): Can toggle on and off as a force; can change value

Surface: Smooth or friction; flat, incline (adjustable angle), downward circular hill, flat cliff with edge to fall off of (projectile motion);

Create path of object (to visualize projectile motion; wouldn’t be hard if u use draw to canvas based on coord of the object)

General: x-axis and y-axis labeled for measurements

Strings and tension forces need to be available as well!!! (obviously pendulum strings will exist, but I’m referring to like two blocks connected by a string, but the heavier one is pulled down by gravity and hanging on a pulley, etc etc

**Dashboard:**

Options: Pause, play, reset (to customizations b4 pressing play), clear (empties whole board/window)

On play: can edit forces incrementally or enter a force to change to at any time (this can help show static friction for example); can drag objects manually around (will prob have to base force on mouse speed?)

Values to display on screen/dashboard (can toggle on/off these values from dashboard): ΣF (magnitude, direction); velocity (dynamically changing), angVelocity, momentum, work, etc

After world functions:

Note collisions = perfectly inelastic when restitution = 0 I believe

Contact form for feature requests, bugs, etc

Some type of interactive tutorial

Way for Kipp to know you did it? (name/ID submit? etc)

Interactive problems (Ask Kipp makes lesson planning easier…):

What force is necessary to get object from pt A to pt B (person does math on paper, then puts in force # into program)? Visual dotted object at point b to show destination kinda thing (and can use the detector or whatever in p5 or matterjs to indicate success)

What torque is necessary to get circle object from pt a to pt b? (so person would need to know where to put the force vector on the screen — perpendicular and some magnitude).

What coefficient of friction necessary to stop object exactly 5 m away from bottom of ramp?

Extra:

Formula sheet (organized by units)

Sharing Physics Simulator setups for others to use/interact w (passport auth for this) (teachers can use this)

? help icons to explain physics concepts perhaps (ex on inclines w a square, ability to show component forces, how to calculate, etc in ? bubble, things like that)

Add this as cool demonstration of forces (<https://www.youtube.com/watch?v=jsYwFizhncE>) (ask user to input how many digits of pi wants to approximate)

Magnetism stuff

Positive / negative particle stuff + electric fields

Etc:

* Three.js to make a 3d version (basically, option to select 2D or 3D, and to easily transfer objects, you’ll need to figure out how to rnder 3d based on the 2d objects matter.js already creates for u)