Galactic Rotation Curves Simulator
20-Week Roadmap with Dark-Matter Extension

Johnny Nguyen
B.S. Computer Science '27, Purdue University

April 20, 2025

Driven by a passion for mathematics, physics, and discovery in astronomy

GitHub Repository •

Contents

1 Overview

This document lays out a rigorous, 20-week plan to build an interactive 3D web app for simulating galactic rotation curves, including a research-grade Dark-Matter module. It covers fundamental derivations, algorithm implementations, custom shaders, UI controls, documentation, and outreach deliverables.

2 Week 0: Project Setup & Lab Notebook

Objectives:

- Initialize GitHub repo with CI (Vercel/GitHub Pages).
- Scaffold a lab-notebook in Markdown with MathJax/KaTeX.
- Draft Project Charter: scope, goals, timeline, key references.

Deliverables:

- Repo with `README.md`, `/notebook/week0.md`, and CI config.

3 Weeks 1–2: Vector Math & Rotations

Readings: Arfken & Weber §§1–3; Lengyel, *3D Math Primer* Ch. 1–3. **Exercises:**

- Derive dot/cross-product identities and Rodrigues' formula.
- Implement `Vector3`, `rotateAxis(v,axis,`θ`)`, and `Quaternion` classes in JS.

Deliverables:

- `vector3.js`, `rotation.js`, `quaternion.js` + unit tests.
- Lab notebook with LaTeX proofs (`notebook/week1.md`).

4 Week 3: Kepler Orbits & Conic Sections

Readings: Goldstein Ch. 3; Kleppner & Kolenkow App. A. **Exercises:**

- Derive polar conic equation $r(\theta) = \frac{p}{1+e\cos\theta}$.
- Code `keplerOrbit(a,e,steps)` and plot in 2D.

Deliverables:

- `keplerOrbit.js` + demo plot.
- Notebook derivation (`notebook/week2.md`).

5 Weeks 4–5: Numerical Integrators

5.1 Week 4: RK4 Integrator

Readings: Numerical Recipes Ch. 16; Hairer et al. Sec. II. **Exercises:**

- Derive RK4 update; implement `rk4Step()`.
- Simulate Sun–Earth; plot energy vs. time.

5.2 Week 5: Adaptive RKF45 & Leapfrog

Readings: Hairer I Sec. IV; *Geometric Numerical Integration* Ch. 6. **Exercises:**

- Implement RKF45 with error control.
- Derive and code velocity-Verlet (symplectic leapfrog).
- Benchmark energy conservation vs. step size.

Deliverables:

- `rk4.js`, `rkf45.js`, `leapfrog.js` + benchmark plots.
- Interactive error plot widget.

6 Weeks 6–8: Stability, WebGL Shaders

6.1 Week 6: Three-Body & Lagrange Points

Readings: Binney & Tremaine; Lagrange (1788). **Exercises:**

- Derive $L_{1–5}$ positions; implement `lagrange.js`.
- Build restricted 3-body solver (Sun–Earth–Mars).

6.2 Week 7: Raw WebGL Pipeline

Readings: OpenGL Red Book Ch. 2–3; Rost GLSL Ch. 1–2. **Exercises:**

- Write MVP vertex shader and fragment shader.
- Build helper functions (`initGL`, `compileShader`, `draw`) and render a cube.

6.3 Week 8: Bloom Custom Effects

Readings: OpenGL ES 3.0; GPU Gems 3 Ch. 36. **Exercises:**

- Implement off-screen FBO + blur for bloom glow.
- Prototype grid/overlay shaders.

7 Weeks 9–12: React-Three-Fiber UI

7.1 Week 9: R3F Core & Shaders

Exercises:

- Scaffold `<Canvas>` scene.
- Port raw shaders into `<shaderMaterial>`.
- Create `OrbitLine` component.

7.2 Week 10: PBR & Environment

Exercises:

- Use `meshPhysicalMaterial` with HDRI environment.
- Load textures for planets.

7.3 Week 11: Post-Processing

Exercises:

- Add `<EffectComposer>` with `<Bloom>` and fog.

7.4 Week 12: Controls & D3 Overlay

Exercises:

- Build React sliders for model parameters.
- Integrate D3 rotation-curve plot overlay.

8 Weeks 13–16: Dark-Matter Rotation Curves

8.1 Week 13: Disk Curve

Derivation:

$$v_{\text{disk}}(r) = \sqrt{\frac{G}{r}\int_0^r 2\pi\,\Sigma_0 e^{-r'/R_d} r'\,dr'}.$$

Implementation:

- Write `rotationCurveDisk(rArray,,Rd)` + tests.

8.2 Week 14: NFW Halo

Derivation:

$$\rho(r) = \frac{\rho_0}{(r/R_s)(1+r/R_s)^2}, \quad v_{\text{halo}}(r) = \sqrt{\frac{GM_{\text{halo}}(<r)}{r}}.$$

Implementation:

- Write `rotationCurveHalo(rArray,,Rs)`.
- Combine to get $v_{\text{tot}}(r)$.

8.3 Week 15: Data Integration & Plot

Exercises:

- Load ‘all$_galaxies.json$‘; $filterforselectedgalaxy.Plotv_{\mathrm{obs}}$, v_{disk}, and v_{tot} with D3.

8.4 Week 16: 3D Orbits

Exercises:

- Spawn particles at radii r_i with angular speed $\omega_i = v_{\mathrm{tot}}(r_i)/r_i$.
- Sync 3D animation with 2D plot updates.

9 Weeks 17–18: Documentation & Outreach

- Embed KaTeX derivations in “Scholar Mode.”
- Write a 12-page technical report.
- Publish a blog series and NPM packages.
- Record a screencast and prepare slides.

10 Week 19: Workshop & Community

- Host a campus/online workshop.
- Share on forums and collect feedback.

11 Week 20: Reflection & Extensions

- Analyze usage metrics.
- Write a “Lessons Learned” post.
- Plan future features (e.g. exoplanet ML, AR/VR).

12 Data Sources & References

- Lelli et al. (2016), “SPARC: Spitzer Photometry & Accurate Rotation Curves.”
- Navarro, Frenk & White (1996), “The Structure of Cold Dark Matter Halos.”
- Goldstein, H., *Classical Mechanics*.
- Press et al., *Numerical Recipes*.
- React-Three-Fiber, Drei, D3, KaTeX official docs.