DotDynamics
Transforming Physics Word Problems into Interactive Simulations

1. Project Description
    Problem Statement

Understanding physics through word problems requires students to mentally visualize abstract motion and relationships between physical quantities. Traditional textbooks and static diagrams often fail to provide dynamic, interactive insight into motion, forces, energy transformations, and optical behavior.

This leads to conceptual gaps, especially in mechanics and wave-based systems.

Proposed Solution

DotDynamics is an interactive physics visualization platform that converts natural language physics problems into real-time computational simulations.

The system:

Parses physics word problems

Extracts key parameters

Classifies the type of motion

Generates mathematically accurate simulations

Displays real-time graphs and energy transformations

Allows dynamic parameter manipulation

Instead of memorizing formulas, users can observe physics in motion.

2. Supported Simulation Engines

DotDynamics currently supports the following physics domains:

Vertical Motion

Projectile Motion

Simple Harmonic Motion (SHM)

Circular Motion

Inclined Plane Motion

Rotational Mechanics

Ray Optics (Mirrors & Lenses)

Each engine is independently modeled using analytical physics equations and rendered using dynamic visualization logic.

3. System Architecture

The platform follows a modular three-layer architecture:

User → React Frontend → Node.js Backend → Firebase Services

Frontend handles UI and real-time physics rendering.

Backend parses and classifies natural language problems.

Firebase manages authentication and cloud data storage.

Architecture Diagram:

4. Tech Stack Used
Frontend

React

HTML5 Canvas

Tailwind CSS

Backend

Node.js

Express.js

Database

Firebase Firestore

Authentication

Firebase Google Authentication

5. Key Features

Natural language physics problem parsing

Real-time simulation rendering

Live parameter manipulation

Analytical equation-based computation

Dynamic graph generation

Universal energy visualization system

Secure user authentication

Cloud-based simulation history

6. How to Run the Project
   Step 1: Clone Repository
   git clone
   cd DotDynamics

Step 2: Install Frontend Dependencies
npm install

Step 3: Start Frontend
npm run dev

Step 4: Start Backend
cd backend
npm install
npm run dev

Backend runs on:

http://localhost:5000

Frontend runs on:

http://localhost:8080 (or specified port)

7. Environment Setup

To run the project successfully:

Create a Firebase project.

Enable:

Firestore Database

Google Authentication

Add Firebase configuration in:

src/firebaseConfig.js

Ensure backend API URL matches frontend configuration.

8. Dependencies

Major dependencies include:

react

firebase

express

node

charting/graphing library (if applicable)

Refer to package.json for full dependency list.

9. Important Instructions

Do not commit .env files.

Ensure Firebase configuration is properly set before running.

Backend must be running for problem parsing to work.

Node.js version 16+ recommended.

For best performance, use latest Chrome or Edge browser.

Firestore rules must be configured to allow authenticated access.

10. Demo Video (MVP)

Demo Video Link:

Watch Demo Video

(Replace # with actual video link)

11. Demo Screenshots (MVP)
Projectile Simulation

SHM Simulation

Ray Optics Simulation

(Add screenshots in /assets folder)

12. Future Scope

Air resistance modeling

Damped oscillations

Advanced optical systems

Comparative multi-simulation mode

Exportable graph reports

Classroom collaboration mode

13. Conclusion

DotDynamics demonstrates how computational physics, real-time rendering, and structured system architecture can bridge the gap between theoretical equations and conceptual understanding.

It transforms static problems into dynamic experiences.