

Project Outline: CheMixer Pro – Precision Chemical Mixing Calculator

I. Introduction

Objective:

Develop a responsive, mobile-friendly web application that helps lawn care professionals, golf course managers, and sports field technicians quickly calculate precise chemical mixtures based on lawn size or spray volume.

Tools and Technologies:

HTML, CSS (Flexbox, Grid, Media Queries), JavaScript, Express.js, Node.js, and the EPA Pesticide Product Label System API.

Goals:

- Deliver an application that focuses on accuracy, responsiveness, and user-friendliness.
 - Provide real-time tank mix calculations based on user inputs.
 - Include direct access to chemical label information via API integration.
 - Lay groundwork for potential data persistence through local storage or a database.
-

II. Responsive Design

Implementation:

- Media Queries: Ensure layouts adapt for phones, tablets, and desktop users in the field.
 - CSS Grid and Flexbox: Create flexible panels for inputs, results, and preset buttons.
 - Mobile Optimization: Prioritize large, thumb-friendly buttons and clear text for outdoor use.
-

III. Feature Implementation

Selected Features:

1. Analyze data that is stored in arrays, objects, sets, or maps and display information about it in the app.
2. Create a function that accepts two or more input parameters and returns a value that is calculated or determined by the inputs.
3. Convert user input between two formats and display the result(e.g. Gallons to ounces)
4. Create a node.js web server using Express.js

Backup Feature:

Persist important data to the user to local storage and make it accessible in the app.

Integration of Third-Party API

The app will use the EPA Pesticide Product Label System(PPLS) API to allow users to search by product name or EPA registration number and retrieve direct links to official labels

IV. Data Handling and Persistence

Data Storage and Retrieval:

- Initially store preset data in JavaScript objects or arrays.
- Implement local storage to persist past mixes and reload them on refresh.
- (Stretch Goal) Add an Express route to handle storing and retrieving mix data from a simple JSON or SQLite database.

Data Visualization (Optional):

- If time allows, add a second page using Chart.js or similar to visualize application data (e.g., total product usage over time).
-

V. Advanced Features (Optional / Future)

- Persistent Mix Log: Store previous tank mixes with timestamps.
 - Database Integration: Experiment with saving data to SQLite3 or MongoDB.
 - User Profiles: Save custom presets per technician or location.
 - Interactive Features: Add autocomplete for chemical names, or smart mix warnings for incompatible products.
 - Add a Custom Mix page where users can add their own chemicals instead of presets
-

VI. Project Development

Node.js Web Server:

- Create an Express-based backend to serve the application.
- Implement one or more routes (e.g., `/mix`, `/labels`, `/presets`).

Frontend Interaction:

- Fetch data from the Express routes using `fetch()` in JavaScript.
- Update the DOM dynamically to show mix results in real time.

Framework (Optional):

- Integrate React for modular UI components (if time allows).
-

VII. Review Process

Internal Review:

- Test input validation, calculation accuracy, and responsiveness across screen sizes.
- Ensure label lookup and links function as expected.

External Feedback:

- Get feedback from coworkers or instructors who work in lawn care to verify mix logic and usability.
 - Incorporate feedback into UI tweaks or formula accuracy improvements.
-

VIII. Documentation and Final Submission

Code Annotation and Documentation:

- Include clear comments in all JavaScript and Express files.
- Write a complete [README.md](#) covering:
 - Project overview and purpose.
 - Setup instructions ([npm install](#), [node server.js](#), etc.).
 - Usage guide with screenshots or GIFs.
 - API integration details (EPA Label System).
 - Known issues or planned future features.

Final Submission:

Deliver a fully functional project showcasing:

- Real-time calculation engine.
 - Responsive layout.
 - At least one API integration.
 - Clean, commented codebase ready for demonstration or portfolio inclusion.
-

CheMixer Pro will be a professional-grade tool for lawn care technicians to accurately calculate chemical tank mixes, saving time and reducing errors. The project emphasizes functionality, mobile usability, and API integration while providing a foundation for future data persistence, analytics, and advanced features.