14-Day JavaScript + Full Stack Bootcamp

14-Day JavaScript + Full Stack Bootcamp
Designed by a Senior JavaScript Instructor for Aspiring Full Stack Developers

Target:
Build and deploy a real full stack project (game feedback system) with clean architecture, testing, and frontend/backend integration. This plan is suited for someone with strong programming fundamentals who wants to level up in JavaScript for real-world job use.

Time per day: ~3-4 hours

Final Project: Game Feedback Management System
- A customizable feedback/bug report form for players
- An admin dashboard for game developers to manage reports
- Full REST API, authentication, database integration
- Optional features: image upload, search, filters, tags
- A foundation to rewrite the frontend later in React

Day 1: JavaScript Essentials I - The Foundation
Topics:
- let, const, scope (block vs global)
- Primitive types: strings, numbers, booleans, null, undefined
- Functions (declaration, expression, arrow functions)
- Conditionals (if, else, ternary), loops (for, while, for...of)
- Basic array and object usage

Challenge:
Write a mini utility library of functions:
- removeDuplicates(arr): Accepts an array and returns a new array with all duplicate values removed.
- flatten(arr): Takes an array that contains nested arrays and returns a flat array.
- deepClone(obj): Returns a deep copy of an object, including all nested objects and arrays.

Day 2: JavaScript Essentials II - Data Mastery
Topics:
- Destructuring arrays and objects
- Spread/rest operators for copying and gathering

- Array methods: map, filter, reduce, some, every
- String methods: split, slice, includes, replace
- Closures & higher-order functions

Challenge:
Add to your utility library:
- groupBy(arr, key): group array of objects by key
- getNested(obj, path): safely access deep keys like a.b.c in an object

---

Day 3: DOM & Events - The Interactive Browser
Topics:
- DOM navigation: querySelector, parentNode, children
- Events: click, submit, event listeners, bubbling
- Input fields, buttons, form data handling
- Using localStorage for persistence

Challenge:
Build a comment board app:
- Add/edit/delete comments using buttons
- Store and retrieve comments using localStorage
- Render comments dynamically in the DOM

Day 4: Async JS + Fetch - Real Data Handling
Topics:
- Callbacks, Promises, async/await
- Using the Fetch API for HTTP requests
- try/catch, error handling patterns

Challenge:
Build a Pokmon viewer:
- Fetch data from pokeapi.co
- Show images and stats
- Handle errors (e.g. invalid names)

Day 5: Project Kickoff - Structure & Setup
Topics:

14-Day JavaScript + Full Stack Bootcamp

- Intro to Node.js
- npm init, creating your project
- Installing Express
- Creating file/folder structure
- Serving static HTML and JS

Challenge:
Set up a basic server:
- GET / serves a basic HTML form
- GET /admin serves a placeholder dashboard

Day 6: Express - Routing & Middleware
Topics:
- Express routes: GET, POST, PUT, DELETE
- Using express.json() to parse body data
- Sending JSON responses
- Postman for testing APIs

Challenge:
Create /api/reports routes:
- GET returns mock reports
- POST accepts and stores reports in memory

Day 7: MongoDB - Persistent Storage
Topics:
- What is MongoDB and why it works well with JS
- Mongoose setup and models
- Connecting to a database
- Writing and querying documents

Challenge:
- Define a Report schema
- Replace memory storage with MongoDB
- Add query filtering by tag or status

Day 8: Connecting Frontend to Backend

14-Day JavaScript + Full Stack Bootcamp

Topics:
- fetch() from frontend to backend routes
- Handling POST submissions from form
- Showing dynamic messages
- Loading existing data on page load

Challenge:
- Submit bug reports via form to your Express backend
- Show loading / success state
- Fetch and display existing reports

Day 9: File Uploads - Sending Screenshots
Topics:
- Handling file uploads with multer
- Frontend file input logic and preview
- Saving file paths in MongoDB
- Serving uploaded files back to users

Challenge:
- Add image upload to your form
- Store image file
- Show thumbnail in the admin dashboard

Day 10: Admin Dashboard Functionality
Topics:
- Admin features: toggle report status, filtering, search
- Query parameters in URLs
- Sorting, timestamps, and display logic

Challenge:
- Add toggle status button (resolved/in progress)
- Add search input and filter buttons
- Sort reports by timestamp

Day 11: Authentication - Securing the Admin Side
Topics:

- JWT or session-based authentication
- Simple login route
- Protecting routes via middleware
- Secure cookie handling or token storage

Challenge:
- Build login form
- Store and verify token
- Hide /admin content if not logged in

Day 12: Testing & Validation
Topics:
- Unit testing with Jest
- API testing with Supertest
- Validating input with express-validator
- Writing assertions and test runners

Challenge:
- Add validation to API endpoints
- Write unit tests for utilities
- Write API tests for /api/reports

Day 13: Deployment & Polish
Topics:
- Hosting backend (Railway, Render)
- Hosting frontend (Netlify, Vercel)
- Environment variables (.env)
- CORS, base URLs, HTTPS

Challenge:
- Deploy full app (frontend + backend)
- Test live app from start to finish

Day 14: Documentation + Devlog Draft
Topics:
- README formatting

14-Day JavaScript + Full Stack Bootcamp

- Screenshots, live demo link
- Markdown skills, repo structure
- Optional: begin drafting a devlog

Challenge:
- Create a great README
- Include instructions, feature list, screenshots, deployment link

---

Optional: Week 3+ - Rewrite Frontend in React
Topics:
- React basics: JSX, props, state
- useEffect, controlled forms
- Routing with React Router
- Component structure

Stretch Goals:
- Rewrite the frontend in React
- Keep backend and database the same
- Use hooks, context, and modular components