Node.js 16 Compatibility Guide

Current Status

The attorney website codebase currently uses dependencies that may require Node.js 18+ for optimal compatibility. This document outlines the necessary changes to make the codebase fully compatible with Node.js 16.

GitHub Actions Update

- **Status**: COMPLETED
- **Change**: Updated `.github/workflows/deploy.yml` to use Node.js 16 instead of 18
- **Impact**: GitHub Pages deployment will now use Node 16

Required Package Downgrades

Critical Dependencies Requiring Downgrade

1. Next.js Framework

- **Current**: `next: "14.2.28"`
- **Node 16 Compatible**: `next: "13.5.6"`
- **Reason**: Next.js 14.x requires Node.js 18+
- **Impact**: Minor feature differences, but core functionality maintained

2. ESLint and TypeScript ESLint

- **Current**:
- `eslint: "9.24.0"`
- `@typescript-eslint/eslint-plugin: "7.0.0"`
- `@typescript-eslint/parser: "7.0.0"`
- **Node 16 Compatible**:
- `eslint: "8.57.0"`
- `@typescript-eslint/eslint-plugin: "6.21.0"`
- `@typescript-eslint/parser: "6.21.0"`
- **Reason**: ESLint 9.x and TypeScript ESLint 7.x require Node.js 18+

3. Node.js Type Definitions

- **Current**: `@types/node: "20.6.2"`
- **Node 16 Compatible**: `@types/node: "16.18.68"`
- **Reason**: Should match the Node.js version being used

4. ESLint Configuration

- **Current**: `eslint-config-next: "15.3.0"`
- **Node 16 Compatible**: `eslint-config-next: "13.5.6"`
- **Reason**: Should match Next.js version

Dependencies That Are Compatible

The following dependencies are already compatible with Node.js 16:

- React 18.x (supports Node 16+)

- TypeScript 5.x (supports Node 16+)

- Tailwind CSS 3.x (supports Node 16+)

- All Radix UI components

- Framer Motion 10.x

- Prisma 6.x (supports Node 16+)

Implementation Steps (Manual)

Since package.json cannot be edited directly, these changes would need to be made manually:

1. Create a new package.json with Node 16 compatible versions:

```json
{
  "name": "app",
  "private": true,
  "engines": {
    "node": ">=16.0.0 <17.0.0",
    "npm": ">=8.0.0",
    "yarn": ">=1.22.0"
  },
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start",
    "lint": "next lint"
  },
  "devDependencies": {
    "@next/swc-wasm-nodejs": "13.5.1",
    "@types/node": "16.18.68",
    "@types/react": "18.2.22",
    "@types/react-dom": "18.2.7",
    "@typescript-eslint/eslint-plugin": "6.21.0",
    "@typescript-eslint/parser": "6.21.0",
    "eslint": "8.57.0",
    "eslint-config-next": "13.5.6",
    "eslint-plugin-prettier": "5.1.3",
    "eslint-plugin-react-hooks": "4.6.0",
    "postcss": "8.4.30",
    "prisma": "6.7.0",
    "tailwind-merge": "2.5.2",
    "tailwindcss": "3.3.3",
    "tailwindcss-animate": "1.0.7",
    "ts-node": "10.9.2",
    "tsx": "4.20.3",
    "typescript": "5.2.2"
  },
  "dependencies": {
    // Keep all current dependencies except:
    "next": "13.5.6",
    // All other dependencies remain the same as they're Node 16 compatible
  }
}
```

2. Install Node 16 using nvm (if needed):

```
# Install nvm if not already installed
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.0/install.sh | bash

# Install and use Node 16
nvm install 16
nvm use 16

# Verify version
node --version  # Should show v16.x.x
```

3. Clear cache and reinstall dependencies:

```
# Remove existing node_modules and yarn.lock
rm -rf node_modules yarn.lock

# Install with Node 16
yarn install
```

Testing Compatibility

Build Test

```
cd /home/ubuntu/attorney_website/app
yarn build
```

Development Server Test

```
cd /home/ubuntu/attorney_website/app
yarn dev
```

Static Export Test

```
cd /home/ubuntu/attorney_website/app
NEXT_OUTPUT_MODE=export yarn build
```

Migration Risks and Mitigation

Low Risk Changes

- Node type definitions update
- ESLint version downgrade
- GitHub Actions Node version (already completed)

Medium Risk Changes

- Next.js downgrade from 14.x to 13.x
- **Mitigation**: Test all routes and static export functionality
- **Backup Plan**: Document any feature differences

High Risk Changes

- None identified - all critical functionality should remain intact

Verification Checklist

- [ ] GitHub Actions uses Node 16
- [ ] Package.json has engines field specifying Node 16
- [ ] Next.js downgraded to 13.5.6
- [ ] ESLint downgraded to 8.57.0
- [ ] TypeScript ESLint downgraded to 6.21.0
- [ ] @types/node updated to 16.18.68
- [ ] Build process works with Node 16
- [ ] Static export functionality preserved
- [ ] All website features function correctly
- [ ] Attorney profile pages work
- [ ] Contact forms work
- [ ] Responsive design maintained

Current Status Summary

Completed:
- GitHub Actions workflow updated to Node 16

Pending (requires manual intervention):
- package.json engine specification
- Dependency version downgrades
- Node 16 testing environment setup

Alternative: Docker Approach

If manual package.json editing is not possible, consider using Docker with Node 16:

```dockerfile
FROM node:16-alpine

WORKDIR /app
COPY package.json yarn.lock ./
RUN yarn install --frozen-lockfile
COPY . .
RUN yarn build

EXPOSE 3000
CMD ["yarn", "start"]
```

This ensures consistent Node 16 environment across all deployments.