

...

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="theme-color" content="#1a365d">
  <title>Security Business Website PRD</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
      line-height: 1.6;
      color: #333;
    }
    h1, h2, h3 {
      color: #1a365d;
    }
    pre {
      background-color: #f4f4f4;
      padding: 10px;
      border-radius: 5px;
      overflow-x: auto;
    }
    code {
      color: #d63384;
    }
    ul {
      list-style-type: disc;
      padding-left: 20px;
    }
    .section {
      margin-bottom: 30px;
    }
  </style>
</head>
<body>
  <h1>Product Requirements Document (PRD) for Security Business Website</h1>

  <div class="section">
    <h2>1. Project Overview</h2>
    <p><strong>Objective:</strong> Build a mobile-first security business website with
modern tooling.</p>
    <p><strong>Tech Stack:</strong></p>
    <ul>
      <li>Frontend: React + TypeScript + Tailwind CSS</li>
      <li>Build Tool: Vite</li>
    </ul>
  </div>
</body>
</html>
```

- <li>Testing: Vitest + Testing Library</li>
- <li>Deployment: GitHub Pages</li>

</ul>

<p><strong>Design Reference:</strong> <a href="https://4d212b8c-2c6a-4849-86ac-402c15600ce4-00-whhbi2lg9v6b.riker.replit.dev">Styling & Layout Guide</a></p>

</div>

<div class="section">

## <h2>2. Development Environment Setup</h2>

### <h3>2.1 Prerequisites</h3>

- <li>Node.js v18+ (for npm scripts and tooling)</li>
- <li>Git</li>
- <li>GitHub Account</li>

</ul>

### <h3>2.2 Initialize Project</h3>

```
<pre><code>
```

```
mkdir security-business-website
```

```
cd security-business-website
```

```
git init
```

```
npm init -y
```

```
</code></pre>
```

</div>

<div class="section">

## <h2>3. Core Dependencies</h2>

### <h3>3.1 Install Dependencies</h3>

```
<pre><code>
```

```
# Frontend
```

```
npm install react react-dom react-router-dom @emailjs/browser axios framer-motion
```

```
react-helmet react-hook-form @hookform/resolvers yup tailwindcss @headlessui/react
```

```
react-icons
```

```
# TypeScript
```

```
npm install typescript @types/node @types/react @types/react-dom @types/react-
```

```
router-dom @types/axios @types/react-helmet --save-dev
```

```
# Build & Tooling
```

```
npm install vite @vitejs/plugin-react prettier eslint eslint-config-prettier husky lint-
```

```
staged --save-dev
```

```
# Testing
```

```
npm install @testing-library/react @testing-library/jest-dom @testing-library/user-
```

```
event vitest --save-dev
```

```
# Deployment
```

```
npm install gh-pages --save-dev
```

```
</code></pre>
</div>
```

```
<div class="section">
  <h2>4. Project Configuration</h2>
  <h3>4.1 TypeScript Config</h3>
  <pre><code>
```

```
{
  "compilerOptions": {
    "target": "ES2020",
    "lib": ["ES2020", "DOM", "DOM.Iterable"],
    "module": "ESNext",
    "strict": true,
    "jsx": "react-jsx",
    "baseUrl": ".",
    "paths": { "@/*": ["src/*"] }
  },
  "include": ["src"]
}
```

```
</code></pre>
<h3>4.2 Tailwind Config</h3>
<pre><code>
```

```
module.exports = {
  content: ["/src/**/*.{ts,tsx}"],
  theme: {
    screens: { sm: "640px", md: "768px", lg: "1024px" },
    colors: { primary: "#1a365d", secondary: "#2d3748" }
  },
  plugins: [require("@tailwindcss/aspect-ratio")]
};
```

```
</code></pre>
<h3>4.3 Vite Config</h3>
<pre><code>
```

```
import { defineConfig } from 'vite';
import react from '@vitejs/plugin-react';

export default defineConfig({
  plugins: [react()],
  base: "/security-business-website/" // Match GitHub repo name
});
```

```
</code></pre>
</div>
```

```
<div class="section">
  <h2>5. Project Structure</h2>
  <pre><code>
```

```
security-business-website/
```

```

├── src/
│   ├── assets/      # Images, fonts, global CSS
│   ├── components/  # Reusable components
│   ├── pages/       # Page components
│   ├── services/    # API/EmailJS integration
│   ├── App.tsx      # Root component
│   └── main.tsx     # Entry point
├── public/          # Static assets
├── .github/workflows/ # CI/CD (optional)
├── .env.example     # Environment variables
├── package.json
└── tsconfig.json

```

<div class="section">

<h2>6. Implementation Steps</h2>

<h3>6.1 Setup Mobile-First Meta Tags</h3>

<pre><code>

&lt;meta name="viewport" content="width=device-width, initial-scale=1.0"&gt;

&lt;meta name="theme-color" content="#1a365d"&gt;

</code></pre>

<h3>6.2 Create Core Components</h3>

<p>Follow the component structure from the <a href="https://4d212b8c-2c6a-4849-86ac-402c15600ce4-00-whhbi2lg9v6b.riker.replit.dev">design reference</a>, including:</p>

<ul>

<li>Responsive navigation (<code>MobileNav.tsx</code>)</li>

<li>Contact form with validation (<code>ContactForm.tsx</code>)</li>

<li>Image gallery with lazy loading (<code>ImageGallery.tsx</code>)</li>

</ul>

<h3>6.3 Routing</h3>

<pre><code>

import { BrowserRouter, Routes, Route } from 'react-router-dom';

import { HomePage, AboutPage, ServicesPage } from './pages';

const App = () => (

&lt;BrowserRouter&gt;

&lt;Routes&gt;

&lt;Route path="/" element={&lt;HomePage /&gt;} /&gt;

&lt;Route path="/about" element={&lt;AboutPage /&gt;} /&gt;

&lt;Route path="/services" element={&lt;ServicesPage /&gt;} /&gt;

&lt;/Routes&gt;

&lt;/BrowserRouter&gt;



</code></pre>

</div>

```
<div class="section">
  <h2>7. EmailJS Integration</h2>
  <h3>7.1 Environment Variables</h3>
  <pre><code>
```

```
# .env
VITE_EMAIL_SERVICE_ID=your_service_id
VITE_EMAIL_TEMPLATE_ID=your_template_id
VITE_EMAIL_PUBLIC_KEY=your_public_key
</code></pre>
```

```
<h3>7.2 Email Service</h3>
```

```
<pre><code>
```

```
import emailjs from '@emailjs/browser';
```

```
export const sendEmail = async (params: EmailParams) => {
  await emailjs.send(
    import.meta.env.VITE_EMAIL_SERVICE_ID,
    import.meta.env.VITE_EMAIL_TEMPLATE_ID,
    params,
    import.meta.env.VITE_EMAIL_PUBLIC_KEY
  );
};
```

```
</code></pre>
```

```
</div>
```

```
<div class="section">
  <h2>8. Testing</h2>
  <h3>8.1 Example Test</h3>
  <pre><code>
```

```
import { render, screen } from '@testing-library/react';
import Button from '../common/Button';
```

```
test('renders button', () => {
  render(<Button variant="primary">Click Me</Button>);
  expect(screen.getByText(/click me/i)).toBeInTheDocument();
});
```

```
</code></pre>
```

```
</div>
```

```
<div class="section">
  <h2>9. Deployment to GitHub Pages</h2>
  <h3>9.1 Configure package.json</h3>
  <pre><code>
```

```
{
  "scripts": {
    "dev": "vite",
    "build": "tsc && vite build",
```

```
"predeploy": "npm run build",
"deploy": "gh-pages -d dist"
},
"homepage": "https://[your-github-username].github.io/security-business-website"
}
```

</code></pre>

### <h3>9.2 Deployment Steps</h3>

<ol>

<li>Push code to GitHub repository.</li>

<li>Run:

<pre><code>npm run deploy</code></pre>

</li>

<li>Enable GitHub Pages in repo settings (<code>Settings > Pages > Branch: gh-pages</code>).</li>

</ol>

</div>

<div class="section">

## <h2>10. Post-Deployment Checklist</h2>

<ul>

<li>Test responsive design on mobile/desktop.</li>

<li>Validate SEO meta tags using <a href="https://search.google.com/test/rich-results">Google Rich Results Test</a>.</li>

<li>Ensure EmailJS integration works in production.</li>

<li>Add analytics (e.g., Google Analytics).</li>

</ul>

</div>

<div class="section">

## <h2>11. Maintenance</h2>

<ul>

<li>Use GitHub Issues for bug tracking.</li>

<li>Enable Dependabot for dependency updates.</li>

<li>Add CI/CD pipeline for automated testing/deployment.</li>

</ul>

</div>

<script>

```
window.onload = function() {
  const { jsPDF } = window.jspdf;
  const doc = new jsPDF();
```

```
  doc.autoTable({ html: '#content' });
```

```
  doc.save('PRD.pdf');
```

```
};
```

</script>

```
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>  
</body>  
</html>  
` ``
```