

iVari Security Systems Website Development Documentation

This document is split into two sections: **Functional Documentation** for non-technical readers and **Technical Documentation** for developers. Both sections detail the design, development, and deployment phases, challenges encountered, and the solutions implemented.

Functional Documentation

Overview

The iVari Security Systems website is a platform that highlights the company's services, client projects, and contact options, providing seamless navigation and responsiveness.

Phases:

1. **Design Phase**
2. **Development Phase**
3. **Deployment Phase**

1. Design Phase

Objective:

Create a professional, user-friendly, and visually appealing website that reflects the company's services and brand identity.

Key Design Decisions:

- **Navigation:** Implemented a navbar for Home, Services, About, Group Companies, and Contact Us sections.
- **Theme:** Used dark-themed colors with accents of red and white to maintain a professional look.
- **Typography:** Used custom fonts such as SulphurPoint and Astoria to enhance readability.
- **Responsiveness:** Ensured compatibility with various devices and screen sizes.

Tools Used:

- **Figma:** For wireframing and prototyping.

2. Development Phase

Objective:

Convert the design into a fully functional and responsive website using modern web technologies.

Key Features:

- Dynamic navigation with links to services such as Fire Extinguishers, Suppression Systems, and Client Projects.
- Contact form integrated with email functionality.
- Smooth animations for user interaction.
- Font optimizations for better UX.

Errors Encountered:

1. **Fonts Not Loading:**
 - **Issue:** Fonts like SulphurPoint and Astoria were not being rendered.
 - **Solution:** Corrected paths in `@font-face` declarations and updated `vite.config.js` to include `assetsInclude` for `.ttf` files.
2. **Animation Issues:**
 - **Issue:** Animations were not triggering as intended.
 - **Solution:** Integrated the Intersection Observer API in `animations.js` to handle smooth scrolling animations. Added defer logic to ensure animations loaded after the DOM.
3. **Deployment DNS Issues:**
 - **Issue:** Emails stopped working after connecting Vercel's nameservers.
 - **Solution:** Switched to A and CNAME records in GoDaddy for hosting while retaining email functionality via MX records.

Tools and Libraries Used:

- **React.js:** For building components and managing state.
- **Vite:** For fast builds and efficient development.
- **SweetAlert2:** For user-friendly pop-ups.
- **CSS:** For styling and animations.

3. Deployment Phase

Objective:

Host the website live for users while ensuring smooth performance and accessibility.

Steps:

1. **Hosting Platform:** Deployed using **Vercel** for automatic builds and deployment.
 2. **DNS Configuration:**
 - Added A record pointing to Vercel's IP for ivari.in.
 - Added CNAME for www pointing to cname.vercel-dns.com.
 3. **Testing:** Conducted thorough checks to ensure all links, forms, and animations worked correctly.
-

Technical Documentation

Design Phase

Wireframes:

- Created wireframes in **Figma** detailing each page's layout.
- Finalized design after stakeholder approval.

Color Palette and Fonts:

- **Colors:**
 - Black (`#000000`)
 - White (`#FFFFFF`)
 - Blood Red (`#CC0118`)
 - Dusty Grey (`#878787`)
 - Dark Void (`#151419`)
 - Smoky Black (`#1B1B1E`)
 - Flash White (`#FBFBFB`)
- **Fonts:**
 - **SulphurPoint** (Light, Regular, Bold)
 - **Astoria** (Light, Regular, Medium, Bold)

Development Phase

Project Setup:

1. **Initialization:**
 - Set up the project using `npm create vite@latest`.
 - Installed dependencies like `react-router-dom` and `sweetalert2`.
2. **Folder Structure:**

```
src/  
├── assets/  
│   ├── fonts/  
│   ├── icons/  
│   └── images/  
├── components/  
├── webpages/  
├── styles/  
│   ├── variables.css  
│   └── style.css  
├── main.jsx  
└── App.jsx
```
3. **Key Functionalities:**
 - **Animations:** Integrated `animations.js` with Intersection Observer for smooth scrolling animations.
 - **Contact Form:** Implemented email functionality using mailto links.
 - **Responsive Design:** Used CSS media queries for adaptability across devices.

Challenges and Solutions:

1. **Font Errors:**
 - **Problem:** Fonts failed to load due to incorrect paths.
 - **Solution:** Updated `variables.css` and ensured fonts were correctly referenced using relative paths. Included `assets Include` in `vite.config.js` for `.ttf` files.

2. **Animation Delays:**
 - **Problem:** Animations didn't trigger properly on some components.
 - **Solution:** Used the Intersection Observer API in `animations.js` to add animations dynamically when elements became visible in the viewport.
3. **DNS and Email Issues:**
 - **Problem:** Connecting *Vercel's nameservers* disrupted email services.
 - **Solution:** Used A records for hosting and retained MX records for email configuration in GoDaddy.
4. **404 Errors on Deployment:**
 - **Problem:** Assets like images and fonts returned 404 errors.
 - **Solution:** Ensured asset paths were correctly set relative to the project's base URL in `vite.config.js`.

Deployment Phase

1. **Build Command:**
 - Ran `vite build` to generate production files in the `dist/` directory.
 2. **Hosting:**
 - Deployed via Vercel with GitHub integration for continuous deployment.
 3. **DNS Configuration:**
 - Configured A and CNAME records in GoDaddy for domain connectivity.
 4. **Testing and Verification:**
 - Verified animations, forms, and navigation links worked seamlessly.
 - Ensured responsive design across devices.
-

Conclusion

The iVari Security Systems website now serves as a professional and accessible platform that highlights the company's services and values. This documentation provides a comprehensive guide to understanding the entire lifecycle of the project, ensuring smooth handoffs and future enhancements.