# Bonzah Insurance Website

## Final Report

**Project Title:** Bonzah Insurance Web Platform

**Client:** Bonzah Insurance

**Course:** COM S 402 – Senior Design

**Instructor: Simanta Mitra**

**Teaching Assistant: Scott Song**

**Semester:** Fall 2025

**Team Members:**

- Emanuel Moreno
- *Eddie Gong*
- *Anthony Campana*
- *Kenneth Gisler*

## Client Letter

# Chapter 1 – Requirements

## 1.1 Problem Statement

Bonzah Insurance required a modern, scalable, and user-friendly web platform to support its insurance offerings. The existing workflow for users to explore coverage options, request quotes, and navigate insurance resources lacked clarity, responsiveness, and integration with backend services. The goal of this project was to design and implement a full-stack web application that improves user experience, supports business growth, and provides a maintainable foundation for future development.

## 1.2 Stakeholders

- Bonzah Insurance (Client)
- Insurance Customers (End Users)
- Bonzah Administrators and Agents
- Development Team

## 1.3 Functional Requirements

- Users can browse insurance products and resources
- Users can request insurance quotes through guided forms
- Users can navigate the website intuitively across devices
- Administrators can manage content and resources
- The system supports secure communication with backend services

## 1.4 Non-Functional Requirements

- Responsive design for desktop and mobile devices
- Secure handling of user data
- High availability and reliability
- Maintainable and modular codebase
- Reasonable performance and page load times

## 1.5 Constraints

- Fixed academic semester timeline
- Required use of modern web technologies
- Integration with existing or planned backend systems
- Deployment and hosting limitations

# Chapter 2 – Design (Including API)

## 2.1 System Architecture Overview

The Bonzah Insurance platform follows a full-stack web architecture consisting of:

- **Frontend:** React with Next.js and TypeScript
- **Backend:** Node.js with Express
- **Database:** MySQL
- **Deployment:** Docker Compose and Netlify CI/CD

The frontend communicates with the backend through RESTful APIs, while the backend handles business logic, authentication, and data persistence.

## 2.2 Design Decisions

- **Next.js** was chosen for server-side rendering and performance benefits
- **TypeScript** improves type safety and maintainability
- **Express** provides a lightweight and flexible backend framework
- **MySQL** ensures structured and reliable data storage
- **Docker** simplifies environment consistency and deployment

## 2.3 API Design

### Endpoints

- `GET /api/resources` – Retrieve insurance resources

- `POST /api/quote` – Submit an insurance quote request
- `GET /api/blogs` – Fetch educational blog content
- `GET /api/chats/open` – Retrieve active support tickets (Polling)
- `POST /api/chats` – Send real-time messages

### Data Format

All API requests and responses use JSON format with clear validation and error handling.

## 2.4 Data Design

The database schema includes tables for:

- Users
- Quotes
- Insurance Products
- Blog Posts and Resources

Entity relationships were designed to support scalability and future expansion.

## 2.5 UI/UX Design

The UI emphasizes:

- Clear navigation
- Consistent branding
- Accessible forms
- Minimal clutter and readable typography

# Chapter 3 – Work Done (By Team Member)

## 3.1 Emanuel Moreno

- Developed major frontend components using React and Next.js
- Implemented navigation helpers and reusable UI components

- Assisted with backend API integration
- Learned and applied Docker, CI/CD workflows, and Netlify deployment

## 3.2 Eddie Gong

- Designed backend API routes and database schema
- Implemented authentication and data validation
- Assisted with deployment configuration

## 3.3 Anthony Campana

- Designed UI layouts and styling for marketing and admin interfaces.
- Implemented real-time polling chat architecture for serverless compatibility.
- Fixed critical SQL syntax and API integration issues for backend stability.
- Resolved Netlify build pipeline failures and module configuration errors.
- Conducted usability testing and created technical documentation.

## 3.4 *Kenneth Gisler*

- Managed project coordination and version control
- Integrated frontend and backend components
- Assisted with testing and bug fixes

# Chapter 4 – Results Achieved

The Bonzah Insurance web platform successfully delivers a modern, responsive, and scalable solution. Key achievements include:

- A fully functional insurance website with guided quote requests
- Clean, maintainable code architecture
- Successful deployment with CI/CD integration
- Positive client feedback on usability and design

Most functional and non-functional requirements were fully met, providing Bonzah Insurance with a strong foundation for future enhancements.

# Appendix – Demo Slides

Demo 1 Presentation
Demo 2 Presentation
Final Demo Presentation