High Level Design Document

Introduction

This High Level Design (HLD) document outlines the architecture and core components for **Emmet Explorer - HTML Snippet Generator**. The project is an educational, open-source tool that enables users to generate and preview HTML snippets using Emmet abbreviations, built with React and Tailwind CSS.

---

1. System Architecture Overview

Architecture Description:
The system is a single-page web application (SPA) structured with modular React components. It processes user input (Emmet abbreviations), generates HTML snippets, and displays live previews, all within the browser.

Module/Component	Role/Responsibility
UI Layer	User input, display output, and controls
Emmet Parser Module	Parses Emmet abbreviations to HTML
Preview Renderer	Renders generated HTML in a live preview area
State Management	Manages user input, output, and application state

---

2. Component Interactions

Sequence Step	Interaction Description
1	User enters Emmet abbreviation in the input field (UI Layer)
2	Input is passed to the Emmet Parser Module
3	Emmet Parser generates HTML snippet
4	Generated HTML is sent to the Preview Renderer
5	Preview Renderer updates the live preview area in the UI

---

3. Data Flow Overview

Data Source	Destination	Data Type	Purpose
User Input (UI)	Emmet Parser	String	Emmet abbreviation for processing
Emmet Parser	Preview Renderer	HTML String	Generated HTML snippet
State Management	All Components	State Object	Synchronize input, output, and preview

---

4. Technology Stack

Layer/Functionality	Technology/Framework
Frontend Framework	React
Styling/UI	Tailwind CSS
Emmet Parsing	JavaScript (Emmet library)
State Management	React useState/useReducer
Build Tooling	Vite or Create React App

---

5. Scalability & Reliability

- **Scalability:**
  The SPA architecture supports modular growth; additional features or components can be integrated with minimal impact.
- **Reliability:**
  All processing occurs client-side, ensuring high availability and responsiveness. No backend dependencies reduce points of failure.
- **Security:**
  User input is sandboxed; generated HTML is rendered safely to prevent XSS. No sensitive data is processed or stored.

End of Document