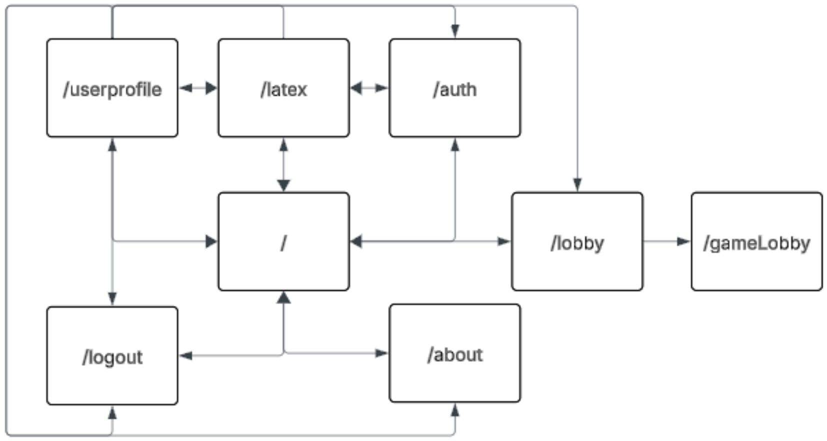
SoftDev P05
TPNG: Lorem ipsum
Roster: Chloe Wong, Princeden Hom, Sascha Gordon-Zolov, Aidan Wong
2025-05-06
Time spent: 2 hours
TARGET SHIP DATE: 2025-06-02

ProtoType

Typing is an extremely useful skill, one that is used daily in all of our lives. However, often the limiting factor with typing has to do with speed. Our website, ProtoType, is set on trying to fix that in a fun way. ProtoType is a typing game centered around improving one's typing speed. We will have two main game modes for typing: individual and multiplayer. In singleplayer users will have the ability to see how fast they are able to type in custom time intervals. They will still be able to compare against their friends, through leaderboard and user profile stats. Additionally, in multiplayer users will be able to race each other, with the results rewarding users who type both fast and accurately.

ProtoType also has a second life, one that lives in LaTeX. While some of us can type quickly (even fast enough to almost keep up with one's thoughts!), there is no doubt that typing in LaTeX is slow and hard. Typing math can be really painful, despite how beautiful the result is. In ProtoType, one will be able to switch to a "LaTeX" mode. Here, an equation will be rendered and displayed. Users will be asked to type the equation in LaTeX, under the same settings as can found in the standard game. Multiple different ways to write identical terms will be accepted, and any other tips/question answers will be displayed in the home LaTeX page.

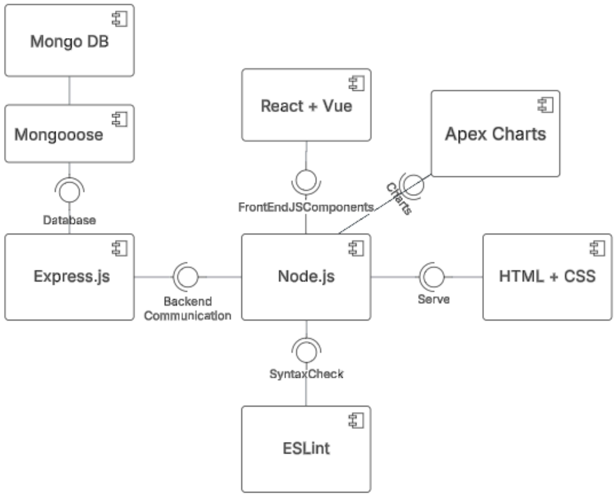
Site Map



Components and LMP *(libraries/modules/packages)*

Component	Role
React + Vite	Front-end framework to build dynamic and reactive UIs
Node.js	Back-end to handle server logic and routing
Express JS	A web application framework to manage middleware and make Node.js easier to use
MongoDB	Cloud-based database to store typing data
Chart.js	Used to render visuals for user performance over time during typing tests and account history
HTML + CSS	Defining site layout and aesthetic
ESLint	Enforce code style, detect syntax errors, and help with finding errors in code
Mongoose	Integrates with MongoDB, schemas to documents (which can compile to models), CRUD operations (Easier to perform MongoDB operations), and hooks (perform actions before/after MongoDB operations)
Websockets	Allows constant communication between a server and user for easy multiplayer functionality

Component Map



Database Organization

<u>Database Type</u>: MongoDB (Document-based)

Collections:

1. users: { username, email, password, joinDate, stats: [testResultIDs] }
2. testResults: { userID, testType, rawInput, wpm, accuracy, date, duration }
3. testTemplates: { type, content, difficulty, tags }—for LaTeX, these are made manually

Front End Framework: React (with Vite)

Why and How React?

1. Component-based, reusable UIs make site building easy
2. Easy integration with Chart.js for accurate and typing speed graphs
3. State management allows for real-time WPM and accuracy tracking
4. Hot Module Replacement that enables website updates without page reload (good for development)

Why and How Vite?

1. Starts the React app quickly
2. Bundle the app when ready for publishing
3. Handles confusing config and slow setup of React, allowing faster development

Datasets

- Popular/Common words

- The LaTeX dataset will consist of a list of common (or uncommon) interesting equations with a variety of complexity. Each equation will have a “conventional” depiction in LaTeX. This is how the equation will be stored. However, multiple different forms of the same equation will be accepted. KaTeX will be used to display this data visually.
- We may also include different settings. For example, users may be able to choose between a variety of datasets, such as poetry/literature/old english/ etc.

APIs

- Custom REST API created with Express: Will handle communication between backend and frontend of website (typing test submissions, fetching test content, user authentication, etc)

Data Visualization Library: Chart.js

Why?

1. Lightweight and React-compatible
2. Simple rendering of line graphs for typing performance
3. Easy to configure and animate for user results

Task Breakdown

Task	Description	Assigned Members
Typing Test Module	Core functionality for typing tests (includes timer, input tracking, real-time WPM and accuracy calculations)	Aidan, Princeden
Typing Race (Multiplayer feature)	Multiplayer lobby where multiple users can compete in real time	Aidan, Princeden
Word Bank (for typing test)	Word datasets for different difficulties and modes	Aidan
LaTeX typing test mode	Typing module where math equations are reproduced in LaTeX syntax	Sascha, Chloe

LaTeX parser	Display LaTeX in real time when typing	Sascha, Chloe
User Profile Page	Display user statistics, test history, WPM/accuracy graphs, etc	Chloe
Database Design + Integration	Schema for website and interaction logic (saving test results, etc)	Aidan
Graphical Data Visualization	Using Chart.js to display various user data	Sascha
Frontend Styling	Making the website look nice	Everyone
Error Handling	Client + server-side input validation, error messages for incorrect LaTeX, words, etc	Princeden