# Nabu

# Architecture Design

## 1. Introduction

The purpose of this document is to give an overview of Nabu's architecture. It first explains the motivation behind the network structure, then shows that structure through a local-level network diagram. It then explains what technologies were used to develop this application, as well as recommended IDEs and extensions for readers looking to build similar projects. The document lists the npm packages necessary for the application's function, as well as those recommended for development. The architecture of Nabu is then discussed and represented visually through an Entity-Relation (ER) diagram and an iconographic diagram that shows the internal file structure of the project.

The goal of Nabu is to allow users to create and join classrooms in which they can create and use learning materials to stimulate gameified learning. As such, Nabu must be able to save some of its users' data, such as their credentials, classrooms, and learning materials. It must also allow interactions from multiple users. The optimal way to implement these desired features is through a website that can update and control its own internal database.
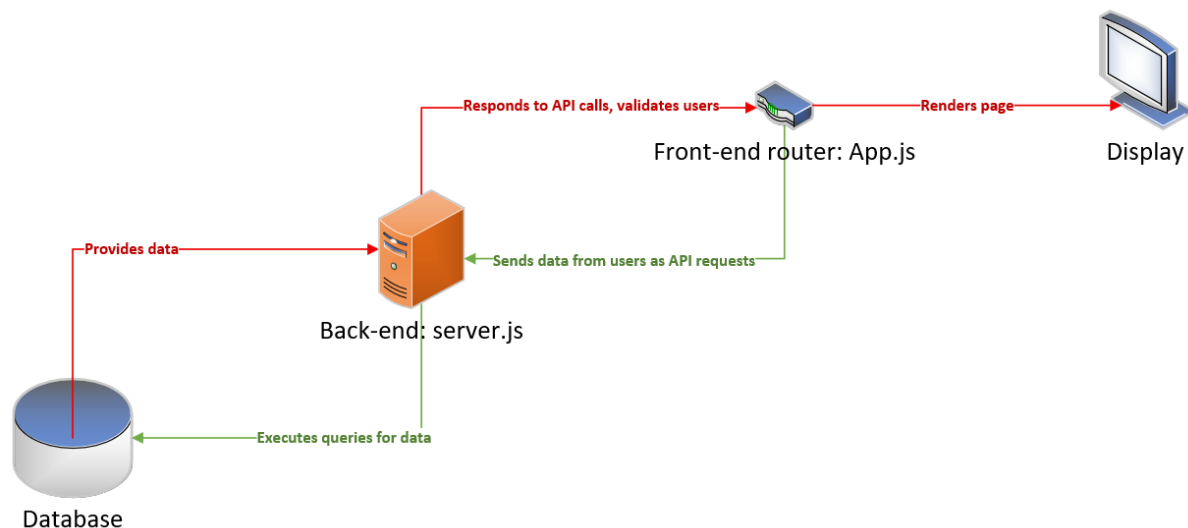
## 2. Network Diagram



Figure 1: The network diagram displays the general flow that the application goes through in order to render a page in the browser.

## 3. Technologies and Purposes

The following table describes the technologies used in this application and their purposes.

| Technology | Purpose |
|---|---|
| **Visual Studio Code** | IDE used to develop the application and synchronize work through GitHub |
| **React** | JavaScript framework used to develop the front-end of the application. |
| **Node.js** | JavaScript runtime environment used to develop the back-end of the application. |
| **MySQL** | SQL database editor and serving environment used to build and host the database. |

### 3.1. Recommended IDEs and extensions

Visual Studio Code (VSCode) was used to develop this application. The following VSCode extensions are recommended for people aiming to develop a similar application:

| Extension | Author |
|---|---|
| **Auto Rename Tag** | Jun Huan |
| **CSS Peek** | Pranay Prakash |
| **ES7+ React/Redux/React-Native Snippets** | dsznajder |
| **JavaScript (ES6) code snippets** | charalampos karypidis |
| **Markdown PDF** | yzane |
| **open in browser** | TechER |
| **React Extension Pack** | Rajbir Jawanda |
| **React PropTypes Generate** | suming |
| **React Refactor** | AppWorks Team |
| **Search node_modules** | Jason Nutter |

## 3.2. List of NPM Packages

This list of npm packages is incomplete, but provides an overview of the packages needed for this project to function. The following packages are necessary for the project to function:

- jsonwebtoken
- body-parser
- bdcrypt
- dotenv
- path
- bootstrap
- react-router-dom
- mysql

The following packages came with React and are necessary for its function:

- react
- react-dom
- react-script
- web-vitals

The following packages are not necessary for the application, but were used in development:

- nodemon

## 3.2.1. Standardized Syntax

A unified coding style is paramount to ensuring code legibility and better cooperation. For the purposes of the Nabu project the W3Schools' JavaScript Coding Conventions and Best Practices will be used. Developers should refer to the Coding Conventions and Best Practices to ensure their code is stylistically standardized before applying changes. The Coding Conventions and Best Practices are extensive and as such a source is provided below to avoid expending too much space describing all applicable rules in detail.

- https://www.w3schools.com/js/js_conventions.asp
- https://www.w3schools.com/js/js_best_practices.asp

## 4. Architecture

Nabu is a website built in React and hosted by Node.js. It is connected to a custom-built API server, which is connected to a database. This structure allows Nabu to take advantage of React's fast loading speed to quickly render its pages and make a smooth, responsive application. In addition, Node.js is one of the most popular hosting frameworks, allowing a lot of flexibility through the npm package library. The custom-built API server ensures security by allowing the developers to carefully curate what data is saved by the application and how that data is used to better protect its users. The database is also custom-built for a similar purpose. Nabu's overall design is such that it is easy to maintain and clearly partitions each function necessary for its operation in a logical manner.
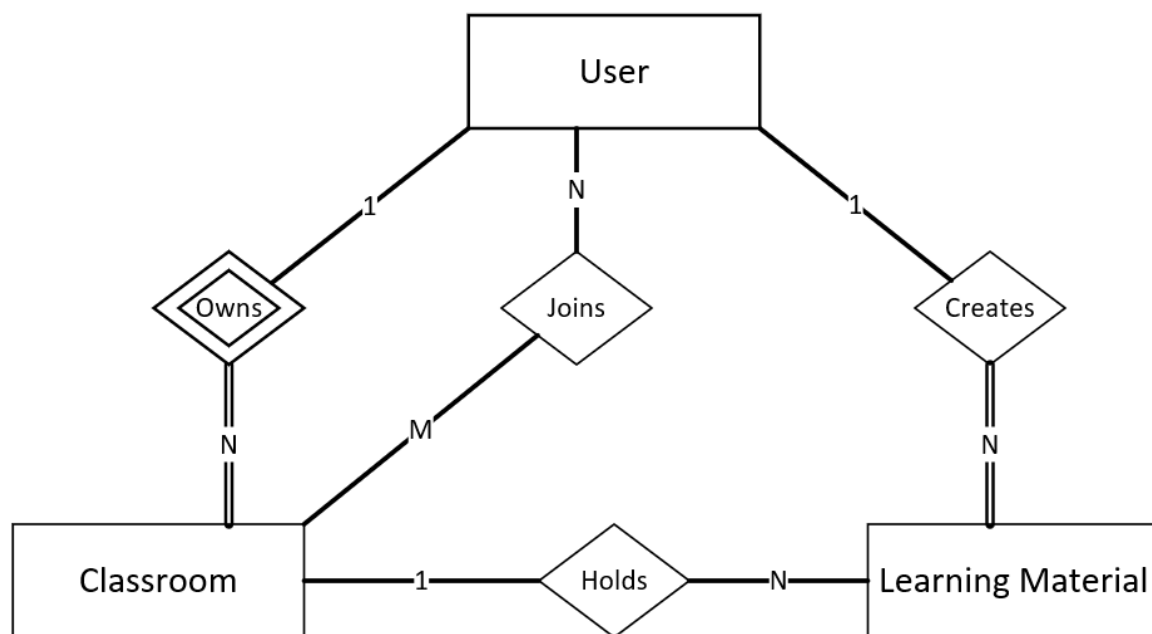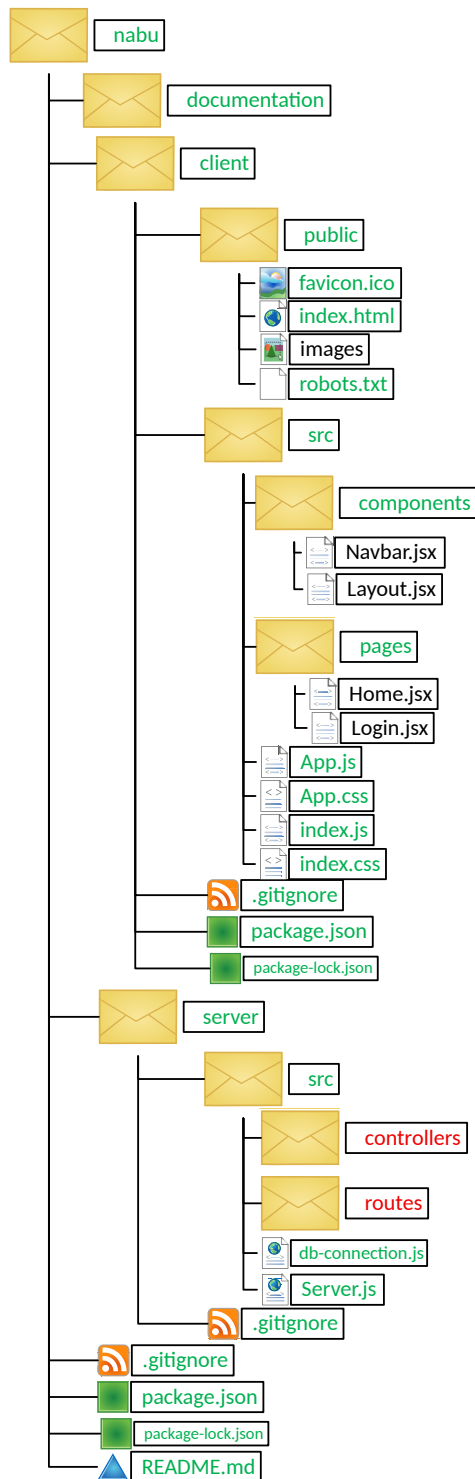
## 5. ER Diagram



Figure 2: The ER diagram shows only the entities and relations present in the database. Attributes have been omitted for readability.

## 5.1. File Structure

Figure 3: The diagram shows an overview of the structure of the GitHub project. Some paths highlighted in red (such as \nabu\server\routes) may be deprecated during development. Paths highlighted in green cannot be deprecated.