**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

**----------------🙢🕮🙠----------------**



**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**MÃ SỐ: 7480201**

**HÀ NỘI - 2025**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

**----------------🙢🕮🙠----------------**

**ĐỒ ÁN TỐT NGHIỆP**

**CHUYÊN NGÀNH KHOA HỌC MÁY TÍNH**

**ĐỀ TÀI**

**XÂY DỰNG PHÂN HỆ QUẢN TRỊ CHO PHẦN MỀM QUẢN LÝ DỮ LIỆU BIM**

|  |  |
| --- | --- |
| **SINH VIÊN THỰC HIỆN**  **NGUYỄN VĂN SÔNG**  **LỚP DCCTCT66\_07A** | **CÁN BỘ HƯỚNG DẪN**  **PGS.TS LÊ HỒNG ANH**  **BỘ MÔN KHOA HỌC MÁY TÍNH** |

**HÀ NỘI – 2025**

MỤC LỤC

[DANH MỤC CÁC CỤM TỪ VIẾT TẮT 10](#_Toc198123702)

[LỜI CẢM ƠN 11](#_Toc198123703)

[MỞ ĐẦU 12](#_Toc198123704)

[CHƯƠNG 1. GIỚI THIỆU CHUNG VỀ ĐỀ TÀI 13](#_Toc198123705)

[1.1. Lý do chọn đề tài 13](#_Toc198123706)

[1.2. Mục tiêu của đề tài 14](#_Toc198123707)

[1.3. Phương pháp nghiên cứu 15](#_Toc198123708)

[1.4. Phạm vi nghiên cứu 16](#_Toc198123709)

[1.5. Cấu trúc của đồ án 17](#_Toc198123710)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ 18](#_Toc198123711)

[2.1. Ngôn ngữ lập trình và công nghệ sử dụng 18](#_Toc198123712)

[2.1.1. Ngôn ngữ HTML 18](#_Toc198123713)

[2.1.2. Ngôn ngữ CSS 19](#_Toc198123714)

[2.1.3. Ngôn ngữ JavaScript 20](#_Toc198123715)

[2.1.4. Framework ReactJS 22](#_Toc198123716)

[2.1.5. NodeJS 23](#_Toc198123717)

[2.1.6. ExpressJS 24](#_Toc198123718)

[2.2. Công cụ hỗ trợ triển khai 33](#_Toc198123719)

[2.2.1. Visual Studio Code 33](#_Toc198123720)

[2.2.2. Postman 35](#_Toc198123721)

[2.2.3. Draw.io 37](#_Toc198123722)

[2.2.4. Figma 39](#_Toc198123723)

[2.3. Speckle platform 41](#_Toc198123724)

[2.3.1. Tổng quan về speckle 41](#_Toc198123725)

[2.3.2. Speckle Viewer: Xem mô hình 3D trực tuyến 43](#_Toc198123726)

[2.3.3. Ứng dụng của Speckle Viewer trong quản lý BIM 45](#_Toc198123727)

[CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG 48](#_Toc198123728)

[3.1. Khảo sát bài toán 48](#_Toc198123729)

[3.1.1. Quản lý và xem dữ liệu 3D trực tuyến trong BIM là gì? 48](#_Toc198123730)

[3.1.2. Lợi ích của quản lý và đặt bàn trực tuyến 49](#_Toc198123731)

[3.1.2. Các chức năng chính của hệ thống 49](#_Toc198123732)

[3.2. Giới thiệu bài toán 51](#_Toc198123733)

[3.2.1. Xác định các tác nhân 51](#_Toc198123734)

[3.2.2. Yêu cầu các chức năng 51](#_Toc198123735)

[3.3. Đặc tả yêu cầu chức năng 52](#_Toc198123736)

[3.4. Đặc tả yêu cầu phi chức năng 53](#_Toc198123737)

[3.4.1. Hiệu năng và khả năng mở rộng: 53](#_Toc198123738)

[3.4.2. Bảo mật: 53](#_Toc198123739)

[3.4.3. Độ tin cậy và khả năng phục hồi: 54](#_Toc198123740)

[3.4.4. Giao diện và trải nghiệm người dùng: 54](#_Toc198123741)

[3.4.5. Bảo trì và nâng cấp: 54](#_Toc198123742)

[3.5. Phân tích biểu đồ ca sử dụng 54](#_Toc198123743)

[3.5.1. Biểu đồ ca sử dụng tổng quát 54](#_Toc198123744)

[3.5.2. Sơ đồ hoạt động 56](#_Toc198123745)

[a. Sơ đồ hoạt động đăng nhập 56](#_Toc198123746)

[b. Sơ đồ hoạt động đăng ký 56](#_Toc198123747)

[c. Sơ đồ hoạt động tạo dự án 57](#_Toc198123748)

[e. Sơ đồ hoạt động thêm thành viên vào dự án 58](#_Toc198123749)

[f. Sơ đồ hoạt động quản lý tài khoản người dùng 58](#_Toc198123750)

[g. Sơ đồ chức năng quản lý dự án 59](#_Toc198123751)

[3.5.4. Biểu đồ tuần tự 60](#_Toc198123752)

[a. Sơ đồ tuần tự đăng nhập và đăng ký 60](#_Toc198123753)

[c. Sơ đồ tuần tự tạo dự án 61](#_Toc198123754)

[d. Sơ đồ tuần tự quản lý dự án 61](#_Toc198123755)

[e. Sơ đồ tuần tự quản lý account 62](#_Toc198123756)

[f. Sơ đồ tuần tự upload file 63](#_Toc198123757)

[3.5.5. Biểu đồ thành phần 64](#_Toc198123758)

[CHƯƠNG 4. THIẾT KẾ HỆ THỐNG 65](#_Toc198123759)

[4.1. Biểu đồ lớp chi tiết 65](#_Toc198123760)

[4.2. Thiết kế cơ sở dữ liệu 66](#_Toc198123761)

[4.2.1. Sơ đồ quan hệ CSDL 66](#_Toc198123762)

[4.2.2. Thông tin các bảng 67](#_Toc198123763)

[4.2.3. Thiết kế giao diện người dùng 70](#_Toc198123764)

[4.3. Kiểm thử 73](#_Toc198123765)

[4.3.1. Kịch bản cho các usecase 73](#_Toc198123766)

[4.3.1.1. Đăng ký tài khoản: 73](#_Toc198123767)

[4.3.1.2. Đăng nhập: 73](#_Toc198123768)

[4.3.1.3. Xem danh sách dự án: 74](#_Toc198123769)

[4.3.1.4. Xem mô hình 3D trực tuyến: 75](#_Toc198123770)

[4.3.1.5. Thêm chú thích trên mô hình: 75](#_Toc198123771)

[4.3.2. Kịch Bản Use Case cho Quản Trị Viên: 76](#_Toc198123772)

[4.3.2.1. Quản lý người dùng: 76](#_Toc198123773)

[4.3.2.2. Quản lý dự án: 76](#_Toc198123774)

[4.3.3.3. Quản lý quyền truy cập: 77](#_Toc198123775)

[CHƯƠNG 5. KẾT QUẢ THỰC NGHIỆM 78](#_Toc198123776)

[5.1. Môi trường cài đặt 78](#_Toc198123777)

[5.1.1. Phần Cứng (Hardware): 78](#_Toc198123778)

[5.1.2. Phần Mềm (Software): 79](#_Toc198123779)

[5.1.3. Yêu cầu Mạng: 79](#_Toc198123780)

[5.2. Môi trường phát triển (Development Environment): 79](#_Toc198123781)

[5.2.1. Cấu hình máy tính cá nhân phát triển 79](#_Toc198123782)

[5.2.2. Cấu trúc thư mục backend 80](#_Toc198123783)

[5.2.3. Cấu trúc thư mục Frontend 87](#_Toc198123784)

[5.3. Kết quả thực nghiệm 89](#_Toc198123786)

[5.3.1. Các trang lớn của dự án 89](#_Toc198123787)

[5.3.2. Các chức năng của admin 90](#_Toc198123788)

[KẾT LUẬN 91](#_Toc198123789)

[HƯỚNG PHÁT TRIỂN ĐỀ TÀI TRONG TƯƠNG LAI 92](#_Toc198123790)

[TÀI LIỆU THAM KHẢO 94](#_Toc198123791)

**DANH MỤC HÌNH ẢNH**

[Hình 2‑1. Ngôn ngữ HTML 17](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960004)

[Hình 2‑2 CSS 18](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960005)

[Hình 2‑3 JavaScript 19](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960006)

[Hình 2‑4 Framework ReactJS 21](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960007)

[Hình 2‑5 NodeJS 22](#_Toc197960008)

[Hình 2‑6 ExpressJS 24](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960009)

[Hình 2‑7 Tailwindcss 25](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960010)

[Hình 2‑8 Vite 27](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960011)

[Hình 2‑9 PostgreSQL 29](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960012)

[Hình 2‑10 Visual Studio Code 33](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960013)

[Hình 2‑11 Postman 35](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960014)

[Hình 2‑12 Drawio 38](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960015)

[Hình 2‑13 figma 39](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960016)

[Hình 2‑14 Các công cụ plugin của figma 40](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960017)

[Hình 2‑15 Speckle server 41](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960018)

[Hình 3‑1 Mô hình 3D với speckle viewer 49](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960019)

[Hình 3‑2 Trực quan hóa mô hình 3D 50](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960020)

[Hình 3‑3 Các chức năng chính với viewer 3D 51](file:///D:\nodejs\bimcde\DATN_NguyenVanSong.docx#_Toc197960021)

[Hình 4‑1 Biểu mẫu đầu vào Admin board với figma 72](#_Toc197960022)

[Hình 5‑1 Token Based Authentication 75](#_Toc197960023)

[Hình 5‑2 Flow for Signup & Login with JWT Authentication 76](#_Toc197960024)

[Hình 5‑3 Flow for Login with JWT refreshToken 76](#_Toc197960025)

[Hình 5‑4 Kiến trúc NodeJS và ExpressJS với Authentication & Authorization 77](#_Toc197960026)

[Hình 5‑5 Cấu trúc thư mục backend 78](#_Toc197960027)

[Hình 5‑6 Cấu trúc thư mục frontend 81](#_Toc197960028)

[Hình 5‑7 Chức năng đăng nhập 82](#_Toc197960029)

[Hình 5‑8 Chức năng đăng ký 82](#_Toc197960030)

[Hình 5‑9 Home page 83](#_Toc197960031)

**DANH MỤC BẢNG**

[Bảng 3. 1. Bảng đặc tả chức năng quản lý tin tức 28](#_Toc191885817)

[Bảng 3. 2. Bảng đặc tả chức năng quản lý món ăn 29](#_Toc191885818)

[Bảng 3. 3. Bảng đặc tả chức năng chặn tài khoản khách hàng 29](#_Toc191885819)

[Bảng 3. 4. Bảng đặc tả chức năng quản lý bàn ăn 30](#_Toc191885820)

[Bảng 3. 5. Đặc tả chức năng quản lý hóa đơn 31](#_Toc191885821)

[Bảng 4. 1. Bảng categories 47](#_Toc191885822)

[Bảng 4. 2. Bảng configs 47](#_Toc191885823)

[Bảng 4. 3. Bảng contacts 47](#_Toc191885824)

[Bảng 4. 4. Bảng detail\_orders 48](#_Toc191885825)

[Bảng 4. 5. Bảng food 48](#_Toc191885826)

[Bảng 4. 6. Bảng news 48](#_Toc191885827)

[Bảng 4. 7 . Bảng orders 49](#_Toc191885828)

[Bảng 4. 8 . Bảng reviews 49](#_Toc191885829)

[Bảng 4. 9. Bảng tables 49](#_Toc191885830)

[Bảng 4. 10. Bảng users 50](#_Toc191885831)

[Bảng 4. 11. Bảng wishlists 50](#_Toc191885832)

[Bảng 3.1. Bảng đặc tả chức năng quản lý tin tức 32](#_Toc191501205)

[Bảng 3.2. Bảng đặc tả chức năng quản lý món ăn 33](#_Toc191501206)

[Bảng 3.3. Bảng đặc tả chức năng chặn tài khoản khách hàng 33](#_Toc191501207)

[Bảng 3.4. Bảng đặc tả chức năng quản lý bàn ăn 34](#_Toc191501208)

[Bảng 3.5. Đặc tả chức năng quản lý hóa đơn 35](#_Toc191501209)

[Bảng 4.1. Bảng categories 50](#_Toc191501210)

[Bảng 4.2. Bảng configs 50](#_Toc191501211)

[Bảng 4.3. Bảng contacts 50](#_Toc191501212)

[Bảng 4.4. Bảng detail\_orders 51](#_Toc191501213)

[Bảng 4.5. Bảng food 51](#_Toc191501214)

[Bảng 4.6. Bảng news 51](#_Toc191501215)

[Bảng 4.7 . Bảng orders 52](#_Toc191501216)

[Bảng 4.8 . Bảng reviews 52](#_Toc191501217)

[Bảng 4.9. Bảng tables 52](#_Toc191501218)

[Bảng 4.10. Bảng users 53](#_Toc191501219)

[Bảng 4.11. Bảng wishlists 53](#_Toc191501220)

# DANH MỤC CÁC CỤM TỪ VIẾT TẮT

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ viết tắt** | **Từ tiếng Anh** | **Từ và nghĩa tiếng Việt** |
| 1 | DB | Database | Cơ sở dữ liệu |
| 2 | PK | Primary Key | Khóa chính |
| 3 | FK | Foreign Key | Khóa ngoại |
| 4 | GVHD |  | Giảng viên hướng dẫn |
| 5 | HTML | Hypertext Markup Language | Ngôn ngữ đánh dấu siêu văn bản |
| 6 | CSS | Cascading Style Sheets | Ngôn ngữ nhằm tìm và định dạng lại các phần tử được tạo ra bởi ngôn ngữ đánh dấu HTML |
| 7 | FE | Front end | Phần phía trước – giao diện của trang web |
| 8 | BE | Back end | Phần phía sau – xử lý logic cho trang web |
| 9 | API | Application Programming interface | Các giao thức kết nối |
| 10 | RDBMS | Relational Database Management System | Hệ quản trị cơ sở dữ liệu quan hệ |
| 11 | BIM | Building Information Modeling | Mô hình thông tin công trình |
| 12 | CDE | Common Data Environment | Môi trường dữ liệu chung |
| 13 | AEC | Architecture, Engineering, and Construction | viết tắt của Kiến trúc, Kỹ thuật và Xây dựng |
| 14 | DOM | Document Object Model | Mo hình các đối tượng tài liệu |

# LỜI CẢM ƠN

Trước hết, tôi xin gửi lời cảm ơn chân thành và sâu sắc đến Trường Đại học Mỏ - Địa chất, khoa công nghệ thông tin và Bộ môn Khoa học máy tính đã tạo điều kiện thuận lợi để tôi thực hiện và hoàn thành đồ án tốt nghiệp với đề tài: *“xây dựng phân hệ quản trị cho phần mềm quản lý dữ liệu BIM”*

Tôi cũng xin được gửi những lời tri ân chân thành đến các thầy cô trong Trường, trong Khoa và Bộ môn Khoa học máy tính và ứng dụng, những người đã tận tâm dìu dắt, truyền đạt kiến thức và kinh nghiệm quý báu trong suốt quá trình học tập và thực hiện dự án. Sự hướng dẫn nhiệt tình và tinh thần cống hiến của thầy cô đã tiếp thêm niềm tin và động lực cho tôi vượt qua mọi khó khăn, thử thách trong quá trình nghiên cứu và phát triển.

Đặc biệt, tôi xin gửi lời cảm ơn sâu sắc đến PGS. Lê Hồng Anh, giảng viên hướng dẫn, người đã tận tâm chỉ dẫn, cung cấp tài liệu và chia sẻ những kinh nghiệm quý báu trong suốt quá trình thực hiện đồ án. Sự hướng dẫn tận tình và định hướng khoa học từ thầy không chỉ giúp tôi hoàn thành tốt nhiệm vụ đề ra mà còn mang lại cho tôi những bài học quan trọng, mở ra định hướng rõ ràng cho con đường nghiên cứu và phát triển sự nghiệp tương lai.

Tôi cũng xin bày tỏ lòng biết ơn đến gia đình, những người luôn ủng hộ, động viên và tin tưởng vào khả năng của em. Sự hy sinh, tình thương và sự khích lệ của mọi người là nguồn động viên vô giá giúp em tự tin tiến bước trên con đường học vấn và sự nghiệp. Không thể không nhắc đến sự đồng hành, chia sẻ của các bạn bè, và tập thể lớp Khoa học máy tính và ứng dụng K66A những người đã cùng em vượt qua những giờ phút căng thẳng, cùng nhau trao đổi ý kiến, góp ý và hỗ trợ nhau trong mọi hoàn cảnh. Tình bạn chân thành và sự giúp đỡ của các bạn chính là nguồn động lực lớn lao giúp em hoàn thành dự án thành công.

Mặc dù đã cố gắng hết sức, nhưng do thời gian và kinh nghiệm còn hạn chế, chắc chắn đồ án này không tranh khỏi những thiếu sót. Tôi hi vọng sẽ nhận được sự đóng góp ý quý báu từ các thầy cô và những người có chuyên môn để tôi có thể tiếp tục hoàn thiện hơn trong những nghiên cứu và dự án sắp tới

Tôi xin chân thành cảm ơn!

Sinh viên thực hiện

Nguyễn Văn Sông

# MỞ ĐẦU

Trong bối cảnh chuyển đổi số mạnh mẽ và sự phát triển vượt bậc của công nghệ thông tin, việc ứng dụng các giải pháp kỹ thuật số vào hoạt động quản lý dữ liệu đã trở thành yếu tố then chốt, mang tính quyết định đến hiệu quả và sự chuyên nghiệp trong nhiều lĩnh vực. Đặc biệt, trong ngành kiến trúc, kỹ thuật và xây dựng (AEC), với sự phức tạp và khối lượng thông tin ngày càng gia tăng, việc quản lý dữ liệu một cách hiệu quả, nhất là dữ liệu mô hình 3D trong môi trường BIM (Building Information Modeling), trở nên vô cùng quan trọng. Từ đó, ý tưởng xây dựng một phân hệ quản trị chuyên biệt cho phần mềm quản lý dữ liệu BIMCDE ra đời, nhằm tối ưu hóa quy trình quản lý, chia sẻ và khai thác dữ liệu 3D cho người dùng

Trong quá trình nghiên cứu và phát triển đề tài, chúng em đã nỗ lực tìm hiểu các xu hướng công nghệ tiên tiến trong lĩnh vực BIM và quản lý dữ liệu, thu thập thông tin từ các chuyên gia và phân tích yêu cầu thực tế của người dùng trong việc làm việc với dữ liệu 3D. Qua đó, đề tài không chỉ hướng đến tính khả thi về mặt kỹ thuật mà còn tập trung vào việc đáp ứng nhu cầu ngày càng cao về một môi trường quản lý dữ liệu BIM trực quan, hiệu quả và dễ dàng sử dụng.

Tôi xin gửi lời cảm ơn sâu sắc đến PGS.TS Lê Hồng Anh và các giảng viên, chuyên gia và những người đã tận tình hướng dẫn, chia sẻ kiến thức và hỗ trợ chúng em trong suốt quá trình nghiên cứu và phát triển đề tài này. Sự động viên và những ý kiến đóng góp quý báu của quý thầy cô, bạn bè và gia đình đã là nguồn động lực to lớn giúp chúng em vượt qua những khó khăn và hoàn thành công trình này.

Hy vọng rằng, “*phân hệ quản trị cho phần mềm quản lý dữ liệu BIM”* được xây dựng sẽ góp phần mang lại những giá trị thiết thực cho người dùng trong ngành AEC, nâng cao hiệu quả quản lý dữ liệu 3D, tối ưu hóa quy trình làm việc và thúc đẩy sự ứng dụng rộng rãi của công nghệ BIM trong các dự án xây dựng.

# GIỚI THIỆU CHUNG VỀ ĐỀ TÀI

## Lý do chọn đề tài

Trong bối cảnh chuyển đổi số mạnh mẽ và sự phát triển không ngừng của công nghệ thông tin, việc ứng dụng các giải pháp kỹ thuật số vào quản lý dữ liệu đã trở thành một xu hướng tất yếu, mang lại những lợi ích to lớn cho nhiều ngành công nghiệp. Đặc biệt, trong lĩnh vực kiến trúc, kỹ thuật và xây dựng (AEC – Architecture Engineering Construction), sự phức tạp của các dự án hiện đại đòi hỏi việc quản lý một khối lượng lớn thông tin đa dạng, trong đó dữ liệu mô hình 3D theo công nghệ BIM (Building Information Modeling) đóng vai trò then chốt. Việc quản lý hiệu quả dữ liệu BIM không chỉ giúp tối ưu hóa quy trình thiết kế, thi công mà còn nâng cao khả năng cộng tác và giảm thiểu rủi ro trong suốt vòng đời dự án.

Tuy nhiên, thực tế cho thấy, nhiều tổ chức và cá nhân trong ngành AEC vẫn đang đối mặt với những thách thức đáng kể trong việc quản lý và khai thác hiệu quả dữ liệu BIM 3D. Sự thiếu hụt các công cụ quản lý chuyên biệt, khả năng tương tác và chia sẻ dữ liệu hạn chế, cùng với các vấn đề về kiểm soát phiên bản và bảo mật thông tin, đã và đang gây ra những ảnh hưởng tiêu cực đến hiệu suất làm việc và chất lượng dự án. Điều này đặt ra một nhu cầu cấp thiết về việc phát triển các giải pháp phần mềm mạnh mẽ và thân thiện với người dùng, nhằm giải quyết những bất cập hiện tại.

Xuất phát từ nhận thức về tầm quan trọng của việc quản lý dữ liệu BIM 3D hiệu quả và những nhu cầu thực tế chưa được đáp ứng đầy đủ, chúng tôi đã quyết định lựa chọn đề tài "*Xây dựng phân hệ quản trị cho phần mềm quản lý dữ liệu BIM*". Đề tài này hướng đến việc nghiên cứu và phát triển một phân hệ quản trị toàn diện, cung cấp các công cụ và chức năng cần thiết để người dùng có thể dễ dàng quản lý, truy cập, chia sẻ và cộng tác trên dữ liệu 3D một cách hiệu quả. Việc xây dựng một môi trường dữ liệu chung (Common Data Environment - CDE) được quản lý tốt sẽ giúp tăng cường khả năng phối hợp giữa các bên liên quan, nâng cao tính minh bạch của thông tin dự án và tối ưu hóa quy trình làm việc tổng thể.

Ngoài ra, việc phát triển một phân hệ quản trị tập trung và trực quan còn góp phần thúc đẩy quá trình chuyển đổi số trong ngành AEC, tạo tiền đề cho việc ứng dụng rộng rãi hơn nữa công nghệ BIM và nâng cao năng lực cạnh tranh của các doanh nghiệp xây dựng Việt Nam trong bối cảnh hội nhập quốc tế.

## Mục tiêu của đề tài

Mục tiêu của đề tài là xây dựng một phân hệ quản trị hiệu quả trên nền tảng web cho phần mềm quản lý dữ liệu BIMCDE, giúp người dùng quản lý, chia sẻ và khai thác dữ liệu mô hình 3D một cách trực quan, tập trung và hiệu quả hơn. Các mục tiêu cụ thể bao gồm:

* Xây dựng một giao diện quản trị trực quan và dễ sử dụng: Phân hệ quản trị sẽ được thiết kế với giao diện thân thiện, các chức năng được bố trí logic, giúp người dùng dễ dàng thao tác và quản lý dữ liệu 3D mà không đòi hỏi nhiều kiến thức chuyên sâu về công nghệ.
* Cung cấp khả năng quản lý người dùng và phân quyền chi tiết: Hệ thống sẽ cho phép quản trị viên tạo, chỉnh sửa và quản lý tài khoản người dùng, đồng thời thiết lập các quyền truy cập khác nhau đối với dữ liệu và các chức năng của phân hệ, đảm bảo tính bảo mật và kiểm soát thông tin.
* Hỗ trợ quản lý dự án và tổ chức dữ liệu BIM: Phân hệ quản trị sẽ cung cấp các công cụ để tạo, quản lý và theo dõi các dự án BIM, cho phép người dùng tổ chức dữ liệu 3D theo cấu trúc dự án, giúp dễ dàng tìm kiếm, lọc và quản lý các tệp tin liên quan.
* Triển khai chức năng quản lý dữ liệu 3D toàn diện: Hệ thống sẽ hỗ trợ các thao tác cơ bản như tải lên, xem trước, tải xuống và quản lý các phiên bản khác nhau của mô hình 3D. Điều này giúp người dùng dễ dàng theo dõi lịch sử phát triển của mô hình và đảm bảo sử dụng phiên bản mới nhất.
* Tích hợp các chức năng hỗ trợ cộng tác: Phân hệ quản trị có thể bao gồm các tính năng như chia sẻ dữ liệu với các thành viên trong dự án, theo dõi hoạt động và lịch sử truy cập, tạo điều kiện cho sự phối hợp hiệu quả giữa các bên liên quan.
* Tối ưu hóa quy trình làm việc với dữ liệu BIM: Bằng cách cung cấp một môi trường quản lý tập trung và các công cụ hỗ trợ, phân hệ quản trị sẽ giúp người dùng tiết kiệm thời gian, nâng cao hiệu suất làm việc và giảm thiểu các sai sót trong quá trình quản lý và sử dụng dữ liệu 3D.
* Tạo nền tảng cho việc mở rộng và tích hợp trong tương lai: Việc xây dựng một phân hệ quản trị vững chắc sẽ tạo tiền đề cho việc tích hợp thêm các chức năng nâng cao khác, chẳng hạn như liên kết với các công cụ phân tích BIM, quản lý tài liệu liên quan hoặc tự động hóa một số quy trình làm việc.

## Phương pháp nghiên cứu

Để thực hiện thành công các mục tiêu đã đề ra, đề tài sẽ áp dụng kết hợp các phương pháp nghiên cứu sau:

***Phương pháp nghiên cứu lý thuyết:***

+ Tìm hiểu sâu về các khái niệm và nguyên lý cơ bản của Mô hình hóa Thông tin Công trình (BIM) và môi trường dữ liệu chung (Common Data Environment - CDE).

+ Nghiên cứu các tiêu chuẩn và quy trình quản lý dữ liệu BIM đang được áp dụng trong ngành AEC.

+ Khảo sát các công nghệ và nền tảng phát triển web hiện đại phù hợp cho việc xây dựng phân hệ quản trị, ví dụ như ReactJS, Tailwindcss cho giao diện người dùng và Node.js, Express, cho phần backend.

+ Nghiên cứu các hệ quản trị cơ sở dữ liệu phù hợp để lưu trữ và quản lý dữ liệu BIM, ví dụ như PostgreSQL.

+ Tìm hiểu về các giao thức và API liên quan đến việc truy cập và quản lý dữ liệu 3D, có thể bao gồm các định dạng tệp BIM (IFC, Revit API...), các dịch vụ đám mây cho BIM, Speckle và các công nghệ visualization 3D trên web.

***Khảo sát và phân tích nghiệp vụ:***

+ Tiến hành thu thập thông tin từ những người dùng tiềm năng (ví dụ: kiến trúc sư, kỹ sư, nhà quản lý dự án) để hiểu rõ nhu cầu, quy trình làm việc hiện tại và những khó khăn họ gặp phải trong việc quản lý dữ liệu BIM 3D.

+ Phân tích các yêu cầu chức năng và phi chức năng cần thiết cho phân hệ quản trị BIMCDE, đảm bảo rằng hệ thống được xây dựng sẽ đáp ứng được nhu cầu thực tế của người dùng.

***Nghiên cứu các giải pháp tương tự:***

+ Nghiên cứu các phần mềm và hệ thống quản lý dữ liệu BIM hiện có trên thị trường, phân tích các tính năng, ưu nhược điểm và cách tiếp cận của chúng trong việc giải quyết các bài toán quản lý dữ liệu 3D.

+ Tham khảo các bài báo khoa học, sách chuyên ngành và tài liệu nghiên cứu liên quan đến quản lý thông tin công trình và phát triển các ứng dụng web cho ngành AEC.

Việc kết hợp các phương pháp nghiên cứu trên sẽ đảm bảo rằng đề tài *"Xây dựng phân hệ quản trị cho phần mềm quản lý dữ liệu BIM”,* được triển khai một cách bài bản, dựa trên cơ sở lý thuyết vững chắc, hiểu biết sâu sắc về nhu cầu thực tế của người dùng và học hỏi kinh nghiệm từ các giải pháp đã có, từ đó mang lại một sản phẩm có giá trị ứng dụng cao trong ngành AEC.

## Phạm vi nghiên cứu

Phạm vi nghiên cứu của đề tài tập trung vào việc thiết kế, xây dựng và triển khai một phân hệ quản trị trên nền tảng web, hỗ trợ quá trình quản lý dữ liệu mô hình 3D cho người dùng trong môi trường BIM CDE. Cụ thể:

* Quản lý người dùng và phân quyền: Phân hệ sẽ cung cấp các chức năng quản lý tài khoản người dùng (tạo, sửa, xóa) và cho phép thiết lập các cấp độ quyền truy cập khác nhau đối với dữ liệu và các chức năng của hệ thống.
* Quản lý dự án: Hệ thống cho phép người dùng tạo, quản lý và theo dõi các dự án BIM, tạo không gian làm việc riêng biệt cho từng dự án.
* Quản lý dữ liệu 3D: Tập trung vào các chức năng cơ bản như tải lên, lưu trữ, tổ chức và quản lý các tệp tin mô hình 3D (ví dụ: IFC, RVT...). Hệ thống có thể hỗ trợ xem trước mô hình 3D cơ bản (nếu khả thi về mặt công nghệ trong phạm vi đề tài).
* Kiểm soát phiên bản: Phân hệ sẽ cung cấp khả năng theo dõi và quản lý các phiên bản khác nhau của mô hình 3D, giúp người dùng dễ dàng truy cập và khôi phục các phiên bản trước đó.
* Giao diện người dùng: Xây dựng giao diện web trực quan, thân thiện và dễ sử dụng, tập trung vào việc quản lý và tương tác với dữ liệu 3D.
* Công nghệ: Nghiên cứu và áp dụng các công nghệ web phù hợp như HTML, CSS, JavaScript và một framework frontend (ReactJS, Tailwindcss) cùng một ngôn ngữ backend và framework (Node.js/Express) và hệ quản trị cơ sở dữ liệu (PostgreSQL).

Với phạm vi nghiên cứu này, đề tài tập trung vào việc xây dựng một nền tảng quản lý dữ liệu 3D cốt lõi cho môi trường BIM CDE, mang lại giải pháp thiết thực cho người dùng trong việc tổ chức, quản lý và chia sẻ hiệu quả các mô hình 3D trong quá trình làm việc cộng tác.

## Cấu trúc của đồ án

CHƯƠNG 1. GIỚI THIỆU CHUNG VỀ ĐỀ TÀI

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG

CHƯƠNG 4. THIẾT KẾ HỆ THỐNG

CHƯƠNG 5. KẾT QUẢ THỰC NGHIỆM

# CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

## Ngôn ngữ lập trình và công nghệ sử dụng

### Ngôn ngữ HTML

.

A logo of a software company

Description automatically generated

Hình 2‑1. Ngôn ngữ HTML

HTML (HyperText Markup Language) là công nghệ nền tảng trong phát triển giao diện người dùng cho các trang web. Chúng đóng vai trò quan trọng trong việc xây dựng cấu trúc và định dạng của website, mang đến trải nghiệm trực quan và dễ sử dụng cho người học. HTML là ngôn ngữ đánh dấu tiêu chuẩn dùng để tạo nên cấu trúc cơ bản của các trang web. Mỗi trang web được thiết kế dựa trên các thẻ HTML, sắp xếp theo cấu trúc cây DOM (Document Object Model). HTML chịu trách nhiệm xác định các thành phần của trang, như tiêu đề, đoạn văn, hình ảnh, liên kết, biểu mẫu và các phần tử giao diện khác.

HTML được tạo ra bởi Tim Berners-Lee, một nhà vật lý học của trung tâm nghiên cứu CERN ở Thụy Sĩ. Hiện nay, HTML đã trở thành một chuẩn Internet được tổ chức W3C (World Wide Web Consortium) vận hành và phát triển. Bạn có thể tự tìm kiếm tình trạng mới nhất của HTML tại bất kỳ thời điểm nào trên Website của W3C.

Phiên bản đầu tiên của HTML xuất hiện năm 1991, gồm 18 tag HTML. Phiên bản HTML 4.01 được xuất bản năm 1999. Sau đó, các nhà phát triển đã thay thế HTML bằng XHTML vào năm 2000.

Đến năm 2014, HTML được nâng cấp lên chuẩn HTML5 với nhiều tag được thêm vào markup, mục đích là để xác định rõ nội dung thuộc loại là gì (ví dụ như: <article>, <header>, <footer>,…).

Theo Mozilla Developer Network thì HTML Element Reference hiện nay có khoảng hơn 140 tag. Tuy nhiên một vài tag trong số đó đã bị tạm ngưng (do không được hỗ trợ bởi các trình duyệt hiện hành).

HTML là một loại ngôn ngữ đánh dấu siêu văn bản. HTML giúp cấu thành các cấu trúc cơ bản của một Website, làm cho trang Web trở thành một hệ thống hoàn chỉnh. Cụ thể, ngôn ngữ đánh dấu siêu văn bản này giúp bố cục, chia khung sườn các thành phần trang Web. Đồng thời, nó còn hỗ trợ khai báo các File kỹ thuật số như nhạc, Video, hình ảnh,…

Nếu muốn Website có cấu trúc tốt, sử dụng nhiều loại yếu tố trong văn bản, bạn sẽ cần đến HTML. Để trở thành lập trình viên website có thể lựa chọn ngôn ngữ lập trình riêng cho Website tùy vào mục đích sử dụng. Nhưng HTML chứa những yếu tố cần thiết cho mọi thể loại Website. Trang Web của bạn sẽ cần đến ngôn ngữ HTML để hiển thị nội dung cho người truy cập. Điều này đúng dù trang của bạn xây dựng trên bất kỳ nền tảng nào, giao tiếp với bất kỳ ngôn ngữ lập trình nào để xử lý dữ liệu, trước khi bắt tay vào thiết kế, làm việc trên một trang Web thì nhà phát triển Web, lập trình viên đều phải học HTML như một loại ngôn ngữ cơ bản.

### Ngôn ngữ CSS



Hình 2‑2 CSS

CSS là viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu ([HTML](https://topdev.vn/blog/html-la-gi/)). Nói ngắn gọn hơn là ngôn ngữ tạo phong cách cho trang web. Bạn có thể hiểu đơn giản rằng, nếu HTML đóng vai trò định dạng các phần tử trên website như việc tạo ra các đoạn văn bản, các tiêu đề, bảng,…thì CSS sẽ giúp chúng ta có thể thêm style vào các phần tử HTML đó như b CSS được phát triển bởi W3C (World Wide Web Consortium) vào năm 1996, vì HTML không được thiết kế để gắn tag để giúp định dạng trang web.

Phương thức hoạt động của CSS là nó sẽ tìm dựa vào các vùng chọn, vùng chọn có thể là tên một thẻ HTML, tên một ID, class hay nhiều kiểu khác. Sau đó là nó sẽ áp dụng các thuộc tính cần thay đổi lên vùng chọn đó.

Mối tương quan giữa HTML và CSS rất mật thiết. HTML là ngôn ngữ markup (nền tảng của site) và CSS định hình phong cách (tất cả những gì tạo nên giao diện website), chúng là không thể tách rời.ố cục, màu sắc trang, đổi màu chữ, font chữ, thay đổi cấu trúc.

### Ngôn ngữ JavaScript



Hình 2‑3 JavaScript

Javascript là một ngôn ngữ lập trình website, được tích hợp và nhúng trong HTML và giúp cho website trở nên sống động hơn. Đây là dạng ngôn ngữ theo kịch bản, được hình thành dựa trên chính đối tượng phát triển có sẵn hoặc đơn giản là tự định nghĩa ra. Javascript cho phép bạn kiểm soát các hành vi của trang web tốt hơn so với việc chỉ sử dụng mỗi HTML. Có thể kể đến một vài ứng dụng thực tiễn cực kỳ quen thuộc của Javascript như slideshow, pop-up quảng cáo hay tính năng autocomplete của Google, … chúng đều được viết bằng ngôn ngữ Javascript.

JavaScript có thể được viết ở phía máy khách (client-side) và cả ở phía máy chủ (serverside) trong phát triển website. Được tạo ra vào những năm 1990, JavaScript ban đầu được thiết kế để cung cấp các tính năng tương tác động cho các trang web.

Javascript được biết đến lần đầu tiên vào tháng 9 năm 1995, được tạo vỏn vẹn trong vòng 10 ngày bởi Brendan Eich – một nhân viên Netscape. Thực chất tên gọi đầu tiên người ta dùng cho ngôn ngữ lập trình này là Mocha, sau đó đổi thành Mona và tiếp tục biến đổi thành Livescript trước khi trở thành ngôn ngữ lập trình Javascript phổ biến như bây giờ. Phiên bản đầu tiên của ngôn ngữ lập trình này bị Netspace giới hạn độc quyền và chỉ có các tính năng hạn chế. Tuy nhiên nó vẫn tiếp tục phát triển theo thời gian và hoàn thiện đến thời điểm hiện tại nhờ sự làm việc liên tục của cộng đồng các lập trình viên.

***Một số đặc điểm quan trọng của JavaScript:***

* Đa nền tảng: JavaScript có thể chạy trên mọi trình duyệt web hiện đại, bao gồm Chrome, Firefox, Safari và Edge. Ngoài ra, JavaScript cũng có thể được sử dụng trên phía máy chủ thông qua các môi trường như Node.js.
* Đối tượng và kiểu dữ liệu: JavaScript là một ngôn ngữ dựa trên đối tượng, điều đó có nghĩa là mọi thứ trong JavaScript đều là một đối tượng hoặc có thể được chuyển đổi thành đối tượng. JavaScript cũng hỗ trợ nhiều kiểu dữ liệu như số, chuỗi, mảng, đối tượng, v.v.
* Tương tác với HTML và CSS: JavaScript cho phép tương tác với các phần tử HTML và CSS trên trang web. Bằng cách sử dụng JavaScript, ta có thể thay đổi nội dung của trang, thao tác với sự kiện như bấm nút, di chuột, v.v., và áp dụng các hiệu ứng hoặc thay đổi giao diện của trang.
* Xử lý sự kiện: JavaScript cho phép xử lý các sự kiện người dùng như bấm nút, gửi biểu mẫu, di chuột, v.v. Ta có thể gắn kết các hàm xử lý sự kiện để đáp ứng các hành động người dùng và thay đổi trạng thái của trang web.
* Thư viện và Framework: JavaScript có nhiều thư viện và framework phổ biến như React, Angular và Vue.js. Những công cụ này giúp tạo ra các ứng dụng web phức tạp, quản lý trạng thái ứng dụng, tối ưu hóa hiệu suất và xây dựng *giao diện người dùng.*

Cho đến thời điểm hiện tại, Javascript đã phát triển và hoạt động trên hầu khắp mọi trình duyệt và trên các thiết bị di động đến máy tính bàn. Chỉ trong hơn 20 năm, Javascript từ một ngôn ngữ lập trình riêng trở thành một công cụ quan trọng nhất trên bộ công cụ của các chuyên viên lập trình web

### Framework ReactJS



Hình 2‑4 Framework ReactJS

React là một thư viện JavaScript phổ biến được sử dụng để xây dựng giao diện người dùng cho các ứng dụng web đơn trang (single-page applications) và ứng dụng di động. React được phát triển bởi Facebook và được phát hành công khai vào năm 2013. Nó đã nhanh chóng trở thành một trong những công nghệ phổ biến nhất trong lĩnh vực phát triển web.

React sử dụng một nguyên tắc gọi là "component-based architecture" (kiến trúc dựa trên thành phần). Điều này có nghĩa là ứng dụng React được xây dựng bằng cách tách thành phần thành các phần nhỏ hơn có thể tái sử dụng. Mỗi thành phần đại diện cho một phần của giao diện người dùng và có thể chứa mã JavaScript và HTML.

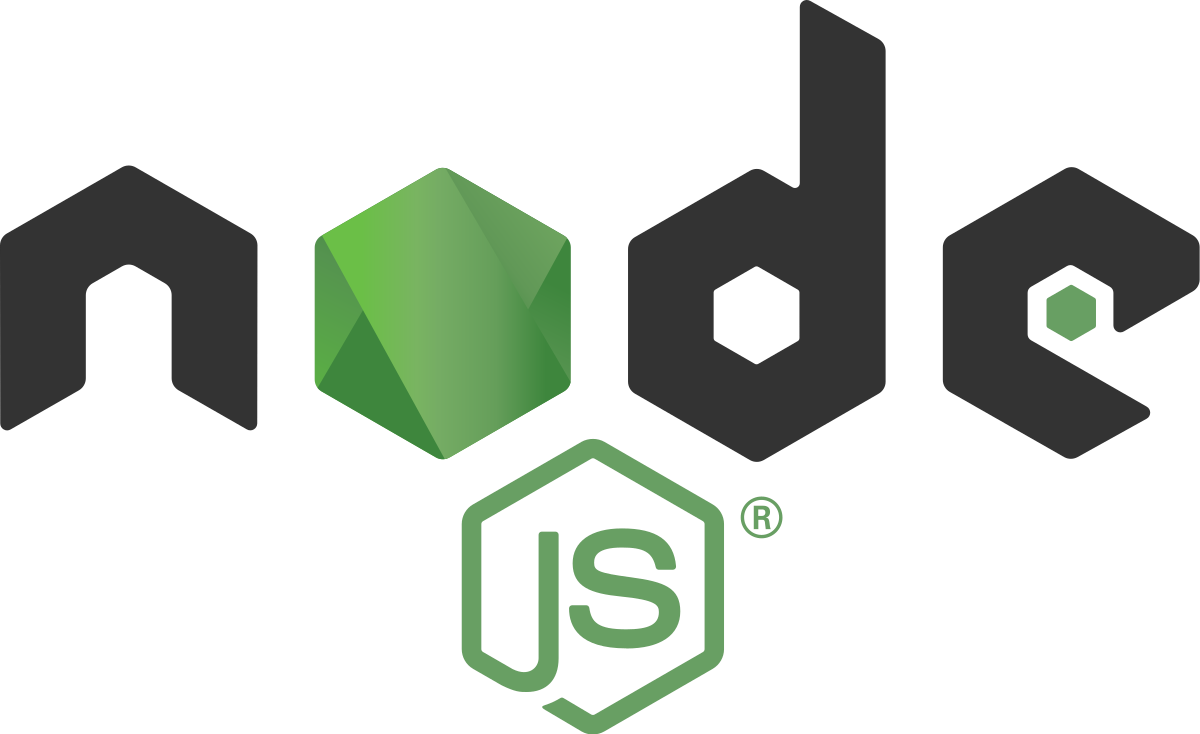
Một trong những điểm mạnh của React là Virtual DOM (DOM ảo). React sử dụng một cấu trúc dữ liệu tương tự DOM để duy trì trạng thái của giao diện người dùng. Khi có sự thay đổi trong trạng thái, React sẽ cập nhật chỉ các thành phần cần thiết trong DOM thực tế, giúp cải thiện hiệu suất.

React cũng có hệ sinh thái phong phú của các thư viện và công cụ hỗ trợ. Các thư viện như React Router giúp quản lý định tuyến trong ứng dụng React, Redux cung cấp quản lý trạng thái toàn cục, và Axios cho phép gửi và nhận dữ liệu từ các API.

Tóm lại, React là một thư viện phát triển giao diện người dùng mạnh mẽ và linh hoạt. Với cấu trúc dựa trên thành phần và Virtual DOM, nó giúp xây dựng ứng dụng web hiệu suất cao, dễ bảo trì và dễ mở rộng.

Với thư viện JavaScrip này, lập trình viên có thể nhập liệu bằng phương pháp mới và render website nhanh chóng và dễ dàng. Tốc độ phản hồi của React được đánh giá rất cao. Hiện nay, nó được sử dụng rộng rãi bởi hơn 1300 nhà phát triển và hơn 94000 website.

### NodeJS



Hình 2‑5 NodeJS

Node.js là một môi trường chạy mã JavaScript phía máy chủ được xây dựng dựa trên Chrome V8 JavaScript engine. Nó cho phép ta thực thi mã JavaScript bên ngoài trình duyệt web, giúp xây dựng các ứng dụng mạng nhanh chóng và mở rộng.

Đặc điểm chính của Node.js bao gồm:

Mã JavaScript phía máy chủ: Node.js cho phép bạn viết mã JavaScript không chỉ cho phía máy khách (trình duyệt web), mà còn cho phía máy chủ. Điều này giúp bạn xây dựng ứng dụng web toàn diện sử dụng cùng một ngôn ngữ lập trình.

Kiến trúc không đồng bộ: Node.js sử dụng mô hình không đồng bộ (asynchronous) với cơ chế xử lý sự kiện (event-driven), cho phép xử lý hàng ngàn kết nối đồng thời mà không gây ra hiện tượng chặn (blocking). Điều này làm cho Node.js rất hiệu quả trong việc xử lý các ứng dụng mạng thời gian thực, đặc biệt là ứng dụng phía máy chủ.

Hệ sinh thái mạnh mẽ: Node.js có một hệ sinh thái mạnh mẽ với hàng ngàn các module mã nguồn mở được phát triển bởi cộng đồng. Các module này giúp bạn xây dựng ứng dụng nhanh chóng bằng cách sử dụng các thư viện và công cụ đã có sẵn.

Scalability (khả năng mở rộng): Với kiến trúc không đồng bộ và hiệu suất cao, Node.js rất phù hợp cho việc xây dựng các ứng dụng có khả năng mở rộng cao. Nó có thể xử lý hàng nghìn yêu cầu đồng thời mà không gặp vấn đề về hiệu suất.

Phát triển đa nền tảng: Node.js có thể chạy trên nhiều nền tảng khác nhau như Windows, macOS và Linux. Điều này cho phép bạn phát triển ứng dụng trên môi trường mà bạn thích và triển khai chúng trên các máy chủ khác nhau.

Với các ưu điểm trên, Node.js đã trở thành một công cụ phổ biến cho việc xây dựng ứng dụng web và dịch vụ mạng. Nó được sử dụng vô cùng rộng rãi trong các công ty ngày nay.

### ExpressJS

ExpressJS là một framework ứng dụng web có mã nguồn mở và miễn phí được xây dựng trên nền tảng Node.js. ExpressJS được sử dụng để thiết kế và phát triển các ứng dụng web và API một cách nhanh chóng. Để hiểu ExpressJS, người dùng chỉ cần phải biết JavaScript, do đó nên việc xây dựng các ứng dụng web và API trở nên đơn giản hơn đối với các lập trình viên và nhà phát triển đã thành thạo JavaScript trước đó.



Hình 2‑6 ExpressJS

Express giúp giảm độ phức tạp trong việc xử lý yêu cầu và phản hồi HTTP bằng cách cung cấp các công cụ và tính năng hỗ trợ như định tuyến (routing), middleware và quản lý độ trễ (middleware pipeline). Điều này giúp phát triển viên dễ dàng xây dựng các ứng dụng web mạnh mẽ mà không cần phải viết lại nhiều mã nguồn.

Express cũng hỗ trợ sử dụng middleware, là các hàm được chạy tuần tự trước khi yêu cầu đến các định tuyến. Middleware cho phép bạn thực hiện các tác vụ như xác thực, ghi nhật ký, nén dữ liệu và xử lý lỗi. Ngoài ra, Express cũng cho phép ta tạo ra các middleware tùy chỉnh để phục vụ các nhu cầu cụ thể của ứng dụng.

Vì ExpressJS là một framework của Node.js nên hầu hết các mã đã được viết sẵn cho các lập trình viên làm việc. Ta có thể tạo các ứng dụng web cho một trang, nhiều trang hoặc kết hợp lại bằng cách sử dụng ExpressJS. framework này khá nhẹ, giúp tổ chức các ứng dụng web ở phía máy chủ thành một kiến trúc MVC hoàn hảo hơn.

Express có một cộng đồng lớn và phong phú của các module và plugin, cho phép bạn mở rộng và mở rộng chức năng của ứng dụng. Bạn có thể tìm thấy các module hỗ trợ cho việc xác thực người dùng, xử lý file tải lên, gửi email, kết nối với cơ sở dữ liệu và nhiều hơn nữa. Tóm lại, Express là một framework phát triển ứng dụng web đơn giản và mạnh mẽ cho Node.js. Với cú pháp dễ hiểu, hỗ trợ middleware và cộng đồng đa dạng, Express là một công cụ tuyệt vời cho việc xây dựng các ứng dụng web và API hiệu quả.

2.1.7. Tailwind CSS



Hình 2‑7 Tailwindcss

Tailwind CSS là một framework CSS utility-first, tương tự như Bootstrap, cung cấp một tập hợp các lớp CSS được tích hợp sẵn mà chúng ta có thể sử dụng trong ứng dụng của mình. Tuy nhiên, điểm đặc biệt của Tailwind CSS là nó cung cấp một số lượng lớn các lớp CSS với các thuộc tính và quy tắc CSS khác nhau. Thay vì sử dụng các lớp CSS được định nghĩa sẵn cho các thành phần cụ thể, Tailwind CSS tập trung vào việc cung cấp các lớp CSS cơ bản mà chúng ta có thể kết hợp để xây dựng giao diện.

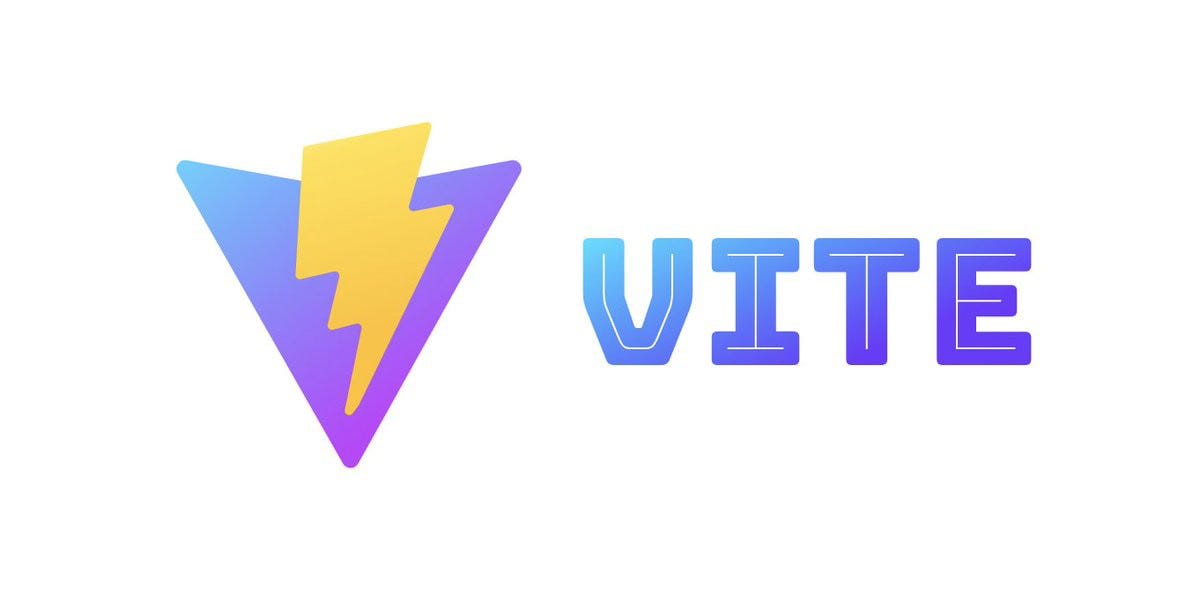
Một trong những ưu điểm của Tailwind CSS là khả năng mở rộng. Bằng cách sử dụng các lớp CSS có sẵn và kết hợp chúng, chúng ta có thể tạo ra những lớp CSS mới phù hợp với yêu cầu cụ thể của dự án. Điều này cho phép chúng ta linh hoạt trong việc tạo ra giao diện độc đáo và tùy chỉnh theo ý muốn.

Với Tailwind CSS, chúng ta không cần phải viết CSS tùy chỉnh nhiều như trước đây. Thay vào đó, chúng ta sử dụng các lớp CSS có sẵn để áp dụng các quy tắc và kiểu dáng cho các phần tử trong giao diện. Điều này giúp tăng hiệu suất phát triển, giảm thời gian viết CSS từ đầu và tạo ra giao diện một cách nhanh chóng.

Những điểm nổi bật của Tailwindcss:

* Tăng tốc độ code với Extension thông minh trên VS Code.Xây dựng giao diện chỉ bằng việc khai báo class trên file HTML: Với Tailwind CSS, chúng ta có thể xây dựng giao diện bằng cách sử dụng các lớp CSS được khai báo trực tiếp trên các phần tử HTML. Điều này giúp giảm đáng kể việc viết CSS tùy chỉnh và tăng tốc độ phát triển.
* Xuất ra chỉ những file CSS chứa các class đã sử dụng: Tailwind CSS có khả năng tạo ra bản build chỉ chứa các class CSS đã sử dụng trong giao diện, giúp giảm kích thước file CSS và tăng hiệu suất tải trang.
* Thiết lập Responsive ngay trên file HTML: Tailwind CSS cung cấp các lớp CSS định nghĩa sẵn cho Responsive Design. Chúng ta có thể dễ dàng áp dụng các lớp này trực tiếp trên file HTML để điều chỉnh giao diện theo các kích thước màn hình khác nhau.
* Hỗ trợ tương tác hover, focus... ngay trên class HTML: Tailwind CSS cung cấp các lớp CSS cho các tương tác như hover, focus, active. Chúng ta có thể sử dụng các lớp này trực tiếp trên các phần tử HTML để tạo hiệu ứng tương tác một cách dễ dàng.
* Tối ưu cấu trúc HTML và CSS có tính lặp lại: Tailwind CSS khuyến khích việc sử dụng các lớp CSS tái sử dụng để tối ưu hóa cấu trúc HTML và CSS. Thay vì viết lại các đoạn mã CSS lặp lại, chúng ta có thể sử dụng lại các lớp có sẵn để tái sử dụng và giảm thiểu mã lặp.
* Định nghĩa thêm class mới phối hợp với các class có sẵn trong Tailwind: Ngoài việc sử dụng các lớp CSS có sẵn, chúng ta có thể định nghĩa thêm các class mới phối hợp với các class có sẵn trong Tailwind. Điều này giúp tạo ra các kiểu dáng và tùy chỉnh giao diện độc đáo cho dự án của chúng ta.
* Hỗ trợ chế độ tối Dark Mode: Tailwind CSS cung cấp hỗ trợ cho chế độ tối Dark Mode bằng cách sử dụng các lớp CSS định nghĩa sẵn. Chúng ta có thể dễ dàng chuyển đổi giữa chế độ sáng và tối trong giao diện của ứng dụng.
* Dễ dàng mở rộng, chỉnh sửa và biến đổi: Tailwind CSS được thiết kế để dễ dàng mở rộng và tùy chỉnh. Chúng ta có thể thêm, chỉnh sửa và thay đổi các tùy chỉnh trong tệp cấu hình của Tailwind để đáp ứng yêu cầu cụ thể của dự án.
* Tăng tốc độ code với Extension thông minh trên VS Code: Tailwind CSS cung cấp các extension thông minh cho các trình biên tập mã phổ biến như VS Code. Các extension này giúp tăng tốc độ code và cung cấpgợi ý các gợi ý, hoàn thiện câu cuối:
* Tăng tốc độ code với Extension thông minh trên VS Code: Tailwind CSS cung cấp các extension thông minh cho các trình biên tập mã phổ biến như VS Code. Các extension này giúp tăng tốc độ code và cung cấp tính năng gợi ý, kiểm tra lỗi và xem trực quan các lớp CSS có sẵn trong Tailwind. Điều này giúp chúng ta viết code hiệu quả và tiết kiệm thời gian trong quá trình phát triển.

2.1.8. Vite

Vitejs là một công cụ build và phát triển ứng dụng web được tạo ra bởi Evan You, người đã phát triển Vue.js, Vite ra đời mục đích khắc phục những hạn chế về tốc độ và hiệu suất.

Hình 2‑8 Vite

Vite mang đến cách tiếp cận mới trong việc xây dựng các dự án web bằng cách tận dụng các module ES gốc trong trình duyệt và cung cấp tính năng Hot Module Replacement (HMR).

Giả sử bạn đang xây dựng một ứng dụng web đơn giản với React. Trước khi có Vite, công cụ phổ biến như Webpackhoặc Create React App (CRA) được sử dụng nhiều. Tuy nhiên, các công cụ này có thể chậm khi khởi động hoặc cập nhật code.

Trước đây, Webpack phải thực hiện quá trình "đóng gói" (bundle) toàn bộ mã nguồn thành một hoặc nhiều tệp lớn trước khi trình duyệt có thể tải và hiển thị trang web, làm cho việc khởi động dự án mất nhiều thời gian, đặc biệt khi mã nguồn lớn và phức tạp.

Vite khác biệt ở chỗ nó tận dụng Rollup để thực hiện việc đóng gói mã cuối cùng, tạo ra các file nhỏ gọn và tối ưu, từ đó cải thiện hiệu suất tổng thể của ứng dụng.

Điều đáng chú ý hơn là cách Vite xử lý mã nguồn trong quá trình phát triển. Thay vì gộp tất cả mã lại từ đầu, Vite sử dụng ES Modules – một tiêu chuẩn JavaScript được browser hỗ trợ – để tải từng module riêng lẻ khi cần thiết. Nói cách khác, nếu trang web của bạn sử dụng nhiều file JavaScript, Vite sẽ chỉ phân phát những file nào mà trình duyệt yêu cầu vào đúng thời điểm đó, thay vì tải toàn bộ ngay từ đầu.

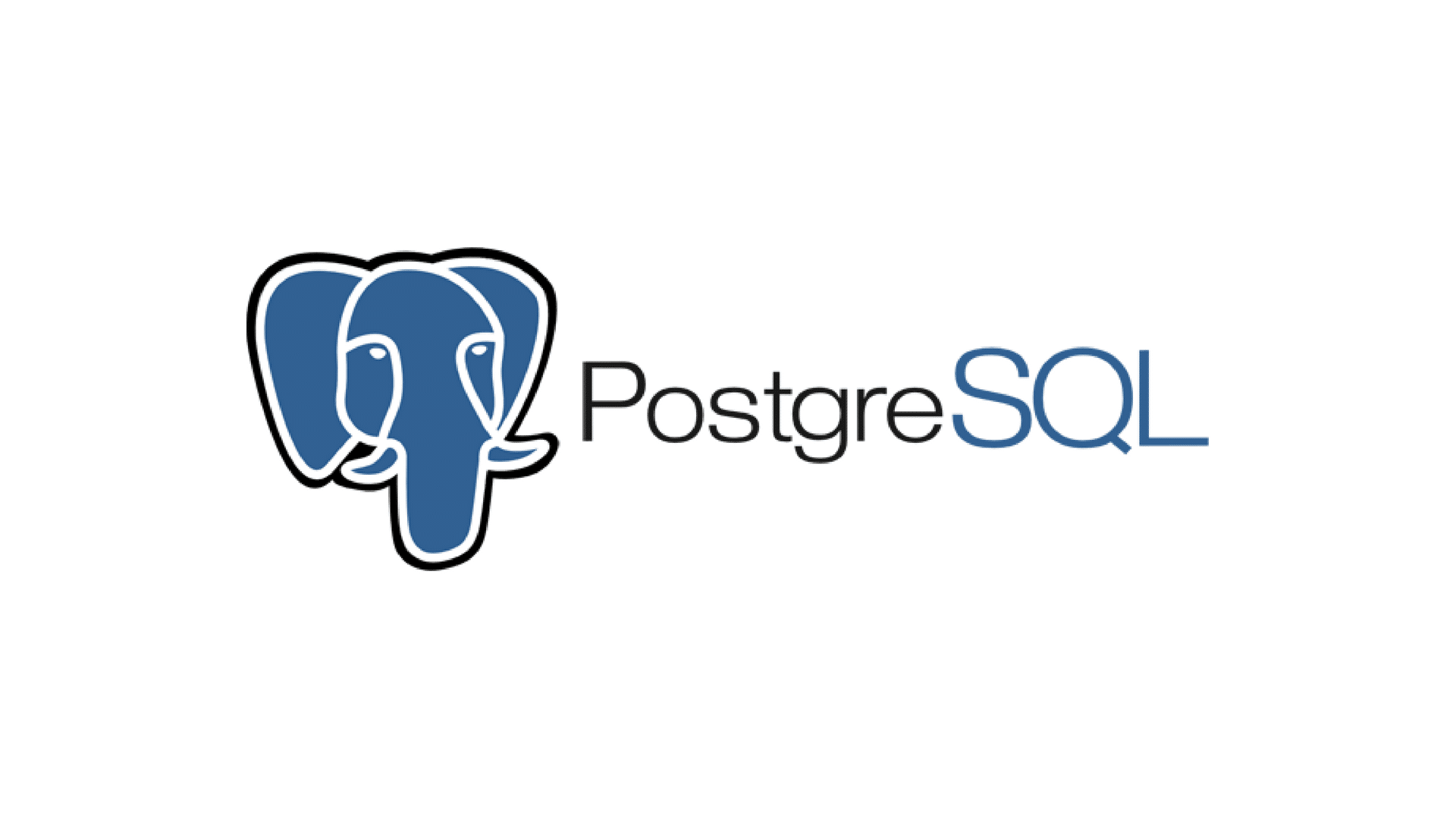
Khi bạn sửa đổi mã, Vite chỉ cập nhật phần liên quan mà không cần phải đóng gói lại toàn bộ ứng dụng. Điều này không chỉ giúp tốc độ phản hồi nhanh hơn mà còn mang lại trải nghiệm phát triển mượt mà và liên tục, giúp bạn thấy ngay kết quả thay đổi mà khôngBạn có thể hiểu đơn giản là, Vite "chia nhỏ" công việc và chỉ làm những gì cần thiết ngay lúc đó, thay vì gộp mọi thứ vào một bước, giúp quá trình phát triển nhanh và linh hoạt hơn nhiều.

Tại sao nên sử dụng Vite:

* Khởi động nhanh chóng: Vite cho phép bạn bắt đầu dự án gần như ngay lập tức, giúp bạn tiết kiệm thời gian.
* Cấu hình đơn giản: Vite được thiết kế để dễ dàng thiết lập, đặc biệt thân thiện với người mới bắt đầu, giúp bạn khởi tạo dự án mà không cần phải loay hoay với các cấu hình phức tạp.
* Hỗ trợ TypeScript: Vite tích hợp tốt với TypeScript, giúp bạn viết mã nguồn an toàn và dễ bảo trì hơn, đồng thời tận dụng đầy đủ các tính năng mạnh mẽ của TypeScript.
* Hiệu suất cao: nhờ sử dụng ES Modules gốc của trình duyệt và tính năng Hot Module Replacement (HMR), Vite mang đến trải nghiệm phát triển mượt mà, với phản hồi tức thì khi bạn thay đổi mã.

Đặc điểm và chức năng của Vite

* Phân phát tệp theo yêu cầu qua ESM gốc: thay vì đóng gói toàn bộ mã nguồn thành một khối lớn, Vite tận dụng các modules ECMAScript (ESM) gốc của browser để tải từng tệp riêng lẻ khi cần. Cách tiếp cận này giúp loại bỏ quá trình đóng gói phức tạp, tăng tốc và tối ưu hiệu suất.
* Hot Module Replacement (HMR): Vite cung cấp tính năng cho bạn thấy ngay lập tức những thay đổi trong code mà không cần phải reload trang hoặc mất đi trạng thái ứng dụng. Điều này giúp trải nghiệm phát triển trở nên mượt mà và liền mạch hơn.
* API plugin và API JavaScript của Vite: bạn có thể thêm các tính năng mới, thay đổi cách Vite hoạt động, hoặc tích hợp với các công cụ khác để phù hợp với yêu cầu của dự án từ đơn giản đến phức tạp.
* Hỗ trợ nhiều framework và công nghệ: Vite hoạt động tốt với React, Vue. Ngoài ra, còn tích hợp được với Sass và Tailwindcss, giúp bạn xây dựng ứng dụng một cách dễ dàng và hiệu quả. làm gián đoạn quá trình làm việc.

2.1.9. Hệ quản trị cơ sở dữ liệu: PostgreSQL

Hình 2‑9 PostgreSQL

PostgreSQL là một hệ thống quản trị cơ sở dữ liệu quan hệ và đối tượng (object-relational database management system) miễn phí và nguồn mở (RDBMS) tiên tiến nhất hiện nay. khả năng mở rộng cao và tuân thủ các tiêu chuẩn kỹ thuật. Nó được thiết kế để xử lý một loạt các khối lượng công việc lớn, từ các máy tính cá nhân đến kho dữ liệu hoặc dịch vụ Web có nhiều người dùng đồng thời.

* PostgreSQL được phát triển bởi PostgreSQL Global Development Group, Phát hành lần đầu: 08/07/1996
* PostgreSQL linh động có thể chạy được trên nhiều nền tảng khác nhau như Mac OS X, Solaris và Windows.
* PostgreSQL là một phần mềm mã nguồn mở miễn phí bởi vậy PostgreSQL có thể được dùng, sửa đổi và phổ biến bởi bất kỳ ai cho bất kỳ mục đích nào.
* PostgreSQL có tính ổn định cao.
* PostgreSQL là hệ thống quản lý cơ sở dữ liệu đầu tiên triển khai tính năng kiểm soát đồng thời nhiều phiên bản (MVCC).

PostgreSQL là một hệ quản trị cơ sở dữ liệu có nhiều tính năng đa dạng, hỗ trợ nhà phát triển xây dựng ứng dụng, quản trị viên bảo vệ tính toàn vẹn dữ liệu và cung cấp một môi trường chịu lỗi (fault-tolerant) để quản lý dữ liệu bất kể kích thước lớn hay nhỏ. PostgreSQL không chỉ là một hệ thống mã nguồn mở miễn phí mà còn có khả năng mở rộng tuyệt vời. Dưới đây là một số tính năng quan trọng của PostgreSQL:

* Kiểu dữ liệu đa dạng: PostgreSQL hỗ trợ nhiều kiểu dữ liệu như số nguyên, số thực, chuỗi, Boolean, ngày/giờ, mảng, phạm vi, UUID, JSON/JSONB, XML, key-value (Hstore), hình học và các kiểu dữ liệu tùy chỉnh.
* Tính toàn vẹn dữ liệu: PostgreSQL cung cấp các cơ chế để đảm bảo tính toàn vẹn dữ liệu như ràng buộc UNIQUE, NOT NULL, Primary Keys, Foreign Keys và ràng buộc loại trừ. Ngoài ra, nó còn hỗ trợ khóa hàm số/explicit locks và khóa khuyến nghị/advisory locks.
* Đồng quy và hiệu suất: PostgreSQL cung cấp các phương pháp lập danh mục (indexing) như B-tree, multicolumn, expressions, partial và các phương pháp lập danh mục nâng cao như GiST, SP-Gist, KNN Gist, GIN, BRIN, Bloom filters. Nó cũng có trình lập kế hoạch/trình tối ưu hóa truy vấn phức tạp, hỗ trợ quét chỉ trên danh mục (index-only scan) và thống kê đa cột.
* Giao tác và đồng thời: PostgreSQL hỗ trợ giao tác ACID (Atomicity, Consistency, Isolation, Durability) và giao tác dạng nest thông qua lưu điểm (savepoints). Nó cũng hỗ trợ điều khiển đồng thời nhiều phiên bản (MVCC) và truy vấn đọc song song.
* Phân vùng bảng: PostgreSQL cho phép phân vùng bảng để tăng hiệu suất truy vấn và quản lý dữ liệu lớn.
* Độ tin cậy và phục hồi: PostgreSQL sử dụng ghi nhật ký ghi trước (Write-ahead Logging - WAL) để đảm bảo độ tin cậy và hỗ trợ khôi phục điểm-theo-thời gian (Point-in-time Recovery - PITR) và active standbys.
* Bảo mật: PostgreSQL cung cấp các phương pháp xác thực như GSSAPI, SSPI, LDAP, SCRAM-SHA-256, Certificate và hệ thống kiểm soát truy cập mạnh mẽ. Nó cũng hỗ trợ bảo mật cấp độ cột và hàng.
* Khả năng mở rộng: PostgreSQL có khả năng mở rộng thông qua các phương pháp lưu trữ và hỗ trợ nhiều ngôn ngữ thủ tục như PL/PGSQL, Perl, Python và nhiều ngôn ngữ khác. Ngoài ra, nó hỗ trợ tích hợp với các cơ sở dữ liệu và ứng dụng khác thông qua giao diện SQL chuẩn và cung cấp nhiều tiện ích mở rộng bổ sung như PostGIS.
* Tìm kiếm văn bản: PostgreSQL hỗ trợ các bộ ký tự quốc tế thông qua ICU collations và cung cấp khả năng tìm kiếm văn bản toàn diện.

PostgreSQL đã được chứng minh là có khả năng mở rộng cao với khả năng quản lý cả terabyte và petabyte dữ liệu trong các môi trường sản xuất. Nó là một hệ quản trị cơ sở dữ liệu mạnh mẽ và linh hoạt, hỗ trợ nhiều tính năng và khả năng tùy chỉnh, đã được áp dụng phổ biến trong các ứng dụng có tính phức tạp.

PostgreSQL có nhiều ưu điểm nổi bật:

* Dễ sử dụng: PostgreSQL cung cấp một cú pháp SQL trực quan và dễ hiểu, giúp người dùng dễ dàng tạo, truy vấn và quản lý cơ sở dữ liệu một cách hiệu quả.
* Hỗ trợ cho ứng dụng web động: PostgreSQL hoạt động tốt với môi trường LAMP (Linux, Apache, MySQL, PHP/Python/Perl), cho phép chạy các trang web và ứng dụng web động một cách linh hoạt và có hiệu suất cao.
* Khả năng lưu trữ nhật ký và khôi phục: PostgreSQL sử dụng ghi nhật ký ghi trước (WAL) để lưu trữ các thay đổi dữ liệu, giúp đảm bảo tính toàn vẹn và khả năng khôi phục dữ liệu khi có lỗi xảy ra.
* Mã nguồn mở miễn phí: PostgreSQL được phát triển dưới giấy phép nguồn mở, cho phép người dùng sửa đổi, tùy chỉnh và triển khai theo nhu cầu của họ mà không phải trả phí cho bản quyền phần mềm.
* Đối tượng hóa dữ liệu: PostgreSQL hỗ trợ đối tượng hóa dữ liệu, cho phép lưu trữ và truy vấn các đối tượng phức tạp như bảng, quan hệ, hình học, mảng và JSON.
* Cộng đồng hỗ trợ đông đảo: PostgreSQL có một cộng đồng người dùng và nhà phát triển rộng lớn, với nhiều kinh nghiệm và kiến thức. Điều này đảm bảo rằng người dùng có sự hỗ trợ và tài liệu phong phú khi gặp vấn đề hoặc cần tư vấn.
* Giảm thiểu công việc bảo trì: PostgreSQL được thiết kế để giảm thiểu công việc bảo trì hệ thống. Nó cung cấp các tính năng như sao lưu và khôi phục dễ dàng, cơ chế tự động tăng cường (autovacuum) để quản lý không gian lưu trữ, cùng nhiều công cụ hỗ trợ quản lý hiệu suất và tối ưu hóa.

PostgreSQL không chỉ dễ sử dụng, mà còn hỗ trợ ứng dụng web động, có khả năng lưu trữ và khôi phục dữ liệu, mã nguồn mở, khả năng đối tượng hóa, hỗ trợ từ cộng đồng đông đảo và giảm thiểu công việc bảo trì hệ thống.

## Công cụ hỗ trợ triển khai

Trong quá trình phát triển website quản lý và đặt bàn cho nhà hàng, việc lựa chọn các công cụ hỗ trợ triển khai là rất quan trọng để đảm bảo hiệu quả công việc, tiết kiệm thời gian và nâng cao chất lượng dự án. Dưới đây là ba công cụ chính được sử dụng trong dự án này:

### Visual Studio Code

Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn miễn phí, đa nền tảng, được phát triển bởi Microsoft. Nó đã nhanh chóng trở thành một trong những trình soạn thảo phổ biến nhất dành cho các nhà phát triển, nhờ giao diện thân thiện, tính năng mạnh mẽ và khả năng mở rộng cao.



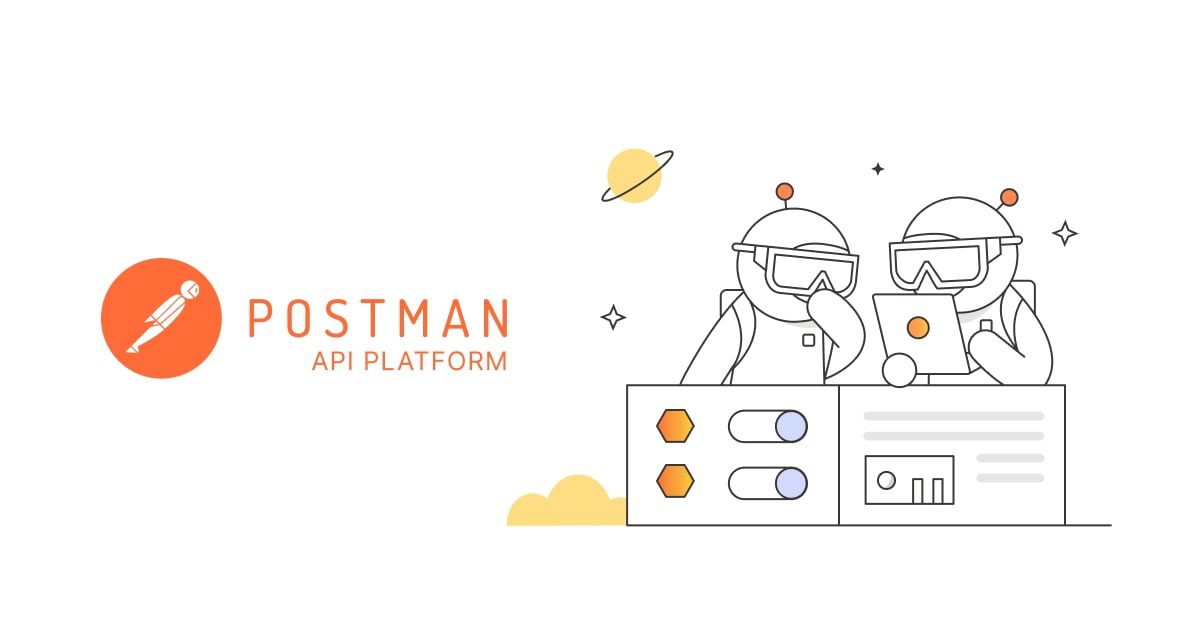
Hình 2‑10 Visual Studio Code

**Tính năng nổi bật:**

* Miễn phí và mã nguồn mở: VS Code hoàn toàn miễn phí và được cấp phép theo giấy phép MIT. Điều này nghĩa là bạn có thể sử dụng và sửa đổi nó tự do mà không phải trả bất kỳ khoản phí nào. Bạn thậm chí có thể đóng góp vào dự án bằng cách sửa lỗi, thêm tính năng mới hoặc dịch thuật. Điều này đã thu hút rất nhiều lập trình viên và đóng góp vào sự phát triển mạnh mẽ của VS Code.
* Đa nền tảng: VS Code có sẵn trên Windows, macOS và Linux, cho phép bạn sử dụng nó trên bất kỳ hệ điều hành nào. Điều này mang lại sự thuận tiện cho các lập trình viên khi họ có thể làm việc trên nhiều thiết bị khác nhau mà không cần thay đổi môi trường phát triển.
* Giao diện thân thiện: VS Code có giao diện đơn giản, dễ sử dụng và trực quan, phù hợp cho cả người mới bắt đầu và các nhà phát triển có kinh nghiệm.
* Hỗ trợ nhiều ngôn ngữ lập trình: VS Code hỗ trợ hơn 30 ngôn ngữ lập trình phổ biến, bao gồm JavaScript, Python, Java, C++, C#, Golang, PHP và nhiều hơn nữa. Bạn có thể sử dụng VS Code để viết mã cho bất kỳ dự án nào, bất kể ngôn ngữ lập trình nào bạn đang sử dụng. VS Code cung cấp hỗ trợ cú pháp, gợi ý mã thông minh và các tính năng gỡ lỗi cho mỗi ngôn ngữ, giúp bạn viết mã hiệu quả và dễ dàng hơn.Ví dụ: Đối với JavaScript: VS Code cung cấp hỗ trợ hoàn chỉnh cho JavaScript, bao gồm IntelliSense (gợi ý mã thông minh), hỗ trợ gỡ lỗi tích hợp, hỗ trợ TypeScript.
* Đa dạng Extension hỗ trợ coder: VS Code có hệ thống extensions rộng lớn, cho phép bạn thêm các tính năng mới và tùy chỉnh trình soạn thảo theo nhu cầu của mình. Đây là một số extensions phổ biến cho người mới bắt đầu sử dụng.
* Hỗ trợ gỡ lỗi hiệu quả: VS Code cung cấp công cụ gỡ lỗi dễ sử dụng, hỗ trợ nhiều ngôn ngữ (Node.js, Python, C++, Java). Điểm nổi bật là thiết lập điểm dừng (Breakpoint), kiểm tra biến (Variable Inspection), điều khiển luồng mã (Control Flow) và console tích hợp.
* So với Sublime Text, VS Code vượt trội về tích hợp gỡ lỗi, tiện ích mở rộng và hỗ trợ Git. Trong khi Sublime Text tập trung vào hiệu suất nhanh và nhẹ nhưng cần phải trả phí. So với Atom, với khả năng tùy chỉnh cao và hỗ trợ cộng tác trực tiếp qua Teletype, lại không bằng VS Code về hiệu suất và ổn định. So với IntelliJ IDEA, mặc dù đắt tiền hơn, cung cấp công cụ phát triển mạnh mẽ và tích hợp sâu cho Java và các ngôn ngữ JVM, nhưng không đa năng như VS Code trong việc hỗ trợ nhiều ngôn ngữ lập trình.
* Kết nối với các dịch vụ phổ biến: VS Code tích hợp tốt với các dịch vụ phổ biến như Git, Azure, GitHub và Docker. Bạn có thể dễ dàng quản lý kho mã Git, triển khai ứng dụng Azure, sử dụng GitHub để cộng tác với các nhà phát triển khác và sử dụng Docker để tạo và chạy các container. Tính năng tích hợp này giúp bạn dễ dàng quản lý và triển khai mã của mình, giúp bạn tập trung vào việc viết mã và phát triển sản phẩm.

### Postman

Postman là một ứng dụng mã nguồn mở dùng để phát triển và kiểm thử các API (Application Programming Interface). Công nghệ cung cấp môi trường cho các nhà phát triển thực hiện hoạt động tạo, chia sẻ, kiểm thử và quản lý các API của họ.



Hình 2‑11 Postman

Postman hiện là một trong những công cụ phổ biến nhất được sử dụng trong thử nghiệm các API.

Như đã biết, API chịu trách nhiệm kết nối các ứng dụng với nhau, có Postman sẽ giúp cho thao tác với API này trở nên dễ dàng hơn. Thông thường, Postman sẽ được dùng cho API kiểu REST. Với Postman, ta có thể gọi Rest API mà không cần viết dòng code nào.

Postman hỗ trợ tất cả các phương thức HTTP (GET, POST, PUT, PATCH, DELETE, …). Bên cạnh đó, nó còn cho phép lưu lại lịch sử các lần request, rất tiện cho việc sử dụng lại khi cần.

Postman là một công cụ mạnh mẽ giúp các nhà phát triển và tester làm việc với API một cách hiệu quả. Dưới đây là một số tính năng nổi bật của Postman:

Postman’s Visualizer cho phép bạn tăng cường việc trình bày dữ liệu phản hồi API bằng cách sử dụng HTML, CSS và JavaScript. Bạn có thể tạo các hình ảnh tùy chỉnh như biểu đồ, bảng hoặc các phần tử khác để làm cho dữ liệu phản hồi dễ đọc và hiểu hơn.

Thư viện tích hợp sẵn (Built-in Libraries): Postman cung cấp nhiều thư viện bên ngoài mà bạn có thể sử dụng trong các tab Pre-request Script và Test Script để thêm các chức năng bổ sung mà JavaScript không có sẵn. Một số thư viện thông dụng trong Postman:

* Moment.js: Một thư viện JavaScript phổ biến để phân tích, thao tác và định dạng ngày giờ.
* Lodash: Thư viện tiện ích cung cấp nhiều hàm hỗ trợ xử lý mảng, đối tượng và chuỗi.
* Faker.js: Thư viện để tạo dữ liệu giả lập thực tế như tên, địa chỉ, số điện thoại, email, v.v.

Kiểm soát luồng công việc (Workflow Control): Postman cho phép bạn kiểm soát thứ tự thực hiện các yêu cầu trong một collection bằng phương thức postman.setNextRequest. Bằng cách sử dụng logic điều kiện trong tab Pre-request Script hoặc Tests, bạn có thể thiết lập yêu cầu tiếp theo được thực hiện dựa trên các điều kiện cụ thể.

Hỗ trợ GraphQL tích hợp sẵn (Built-in GraphQL Support): Postman cung cấp hỗ trợ mạnh mẽ cho GraphQL, giúp các nhà phát triển làm việc với các API GraphQL dễ dàng hơn. Các tính năng và chức năng chính cho GraphQL trong Postman:

* GraphQL Variables: Cho phép bạn định nghĩa và sử dụng các biến GraphQL trong các yêu cầu.
* GraphQL Request Body: Gửi các truy vấn và mutation GraphQL dưới dạng yêu cầu POST trong phần thân yêu cầu.
* GraphQL Autocompletion: Hỗ trợ tự động hoàn thành các trường, kiểu và tham số khi bạn gõ.
* User-Defined GraphQL Schemas: Cho phép nhập và sử dụng các schema GraphQL của riêng bạn.
* GraphQL Query History: Giữ lịch sử các truy vấn GraphQL, giúp dễ dàng truy cập và tái sử dụng.

Collections: Postman cung cấp tính năng “Collections” để giúp bạn tổ chức các yêu cầu API của mình. Với collections, bạn có thể nhóm các yêu cầu liên quan lại với nhau, giúp quản lý và thực hiện nhiều yêu cầu như một đơn vị đồng nhất. Các tính năng và lợi ích của Collections trong Postman:

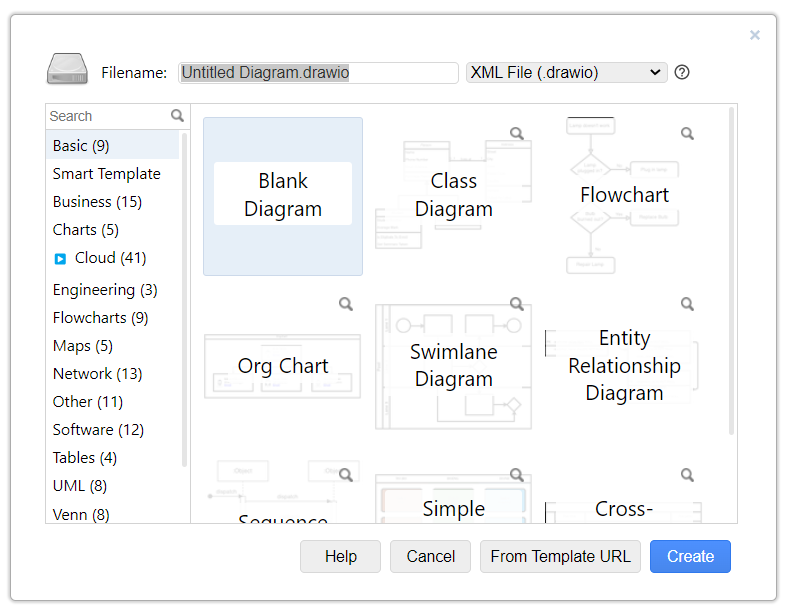
* Organization: Tổ chức các yêu cầu API theo cách phân cấp.
* Execution: Dễ dàng thực hiện tất cả các yêu cầu trong một collection chỉ với một lần nhấp.
* Variables and Environments: Hỗ trợ sử dụng các biến và thiết lập các môi trường khác nhau.
* Sharing and Collaboration: Chia sẻ các collections với các thành viên trong nhóm để cộng tác.
* Documentation: Tạo tài liệu cho các collections để người khác hiểu và sử dụng API.

Kiểm thử và xác nhận (Tests and Assertions): Postman bao gồm một khung kiểm thử mạnh mẽ, cho phép bạn viết các bài kiểm tra bằng JavaScript để xác thực các khía cạnh khác nhau của phản hồi API. Các xác nhận phổ biến trong Postman tests:

* Status code assertion: Xác minh mã trạng thái phản hồi khớp với giá trị mong đợi.
* Response body assertion: Kiểm tra dữ liệu trong phần thân phản hồi.
* Header assertion: Xác thực sự hiện diện và giá trị của các tiêu đề cụ thể trong phản hồi.
* Response time assertion: Đảm bảo thời gian phản hồi nằm trong phạm vi chấp nhận được.

### Draw.io

draw.io là một ứng dụng vẽ biểu đồ miễn phí cho phép người dùng tạo và chia sẻ các sơ đồ trên trình duyệt web. Công cụ trực tuyến này hoạt động với G Suite / Google Drive và Dropbox. Người dùng cũng có thể làm việc trên sơ đồ ngoại tuyến và lưu trữ cục bộ bằng ứng dụng draw.io dành cho macOS, Windows và Linux.

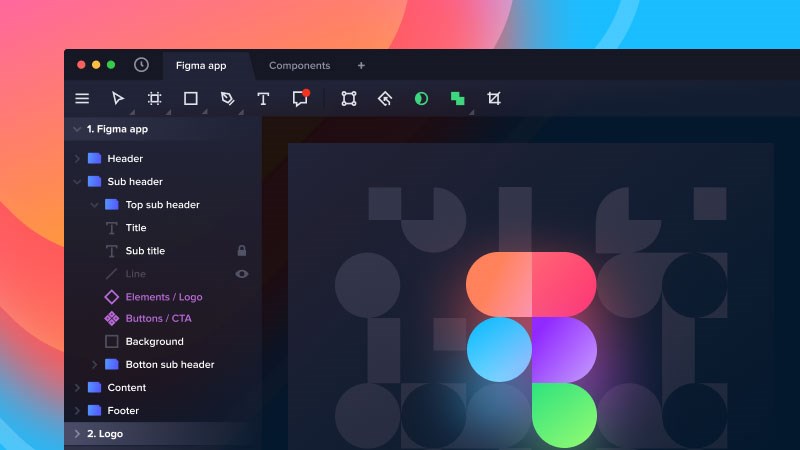


Hình 2‑12 Drawio

draw.io cung cấp giao diện trực quan với tính năng kéo và thả, các mẫu sơ đồ có thể tùy chỉnh và thư viện hình khối mở rộng. Người dùng có thể tạo và chỉnh sửa một loạt các sơ đồ, biểu đồ bao gồm biểu đồ luồng, biểu đồ quy trình, sơ đồ tổ chức, sơ đồ ER, UML, sơ đồ mạng, ... Hệ thống chức năng phong phú của draw.io cho phép người dùng theo dõi và khôi phục các thay đổi, nhập và xuất các định dạng khác nhau, đồng thời tự động xuất bản và chia sẻ công việc.

### Figma

*Figma là một ứng dụng gồm nhiều công cụ thiết kế mạnh mẽ trên nền tảng website. Với Figma, bạn có thể sáng tạo không giới hạn giao diện người dùng (UI/UX), thiết kế tạo mẫu, tạo bài đăng trên các mạng xã hội và nhiều dự án thiết kế khác*



Hình 2‑13 figma

Với thành công của phiên bản dùng thử vào cuối năm 2015, Figma đã chính thức ra mắt vào tháng 09 năm 2016 và tạo được vị trí riêng của mình trong cộng đồng thiết kế. Đến tháng 10 năm 2019, Figma Community ra đời. Từ đây, các nhà thiết kế có thể xuất bản tác phẩm của mình để người khác xem và phối lại.

Tháng 04 năm 2021, Figma phát triển FigJam, một công cụ cho phép người dùng sử dụng các công cụ ghi chú, vẽ hay thêm các biểu tượng cảm xúc vào các dự án cộng tác.

***Khả năng tương thích cao***

Là một phần mềm trên nền tảng website, Figma có thể được truy cập và sử dụng trên hầu hết các hệ điều hành phổ biến hiện nay như Windows, MacOS hay thậm chí là Linux, chỉ cần thiết bị của bạn được kết nối internet.

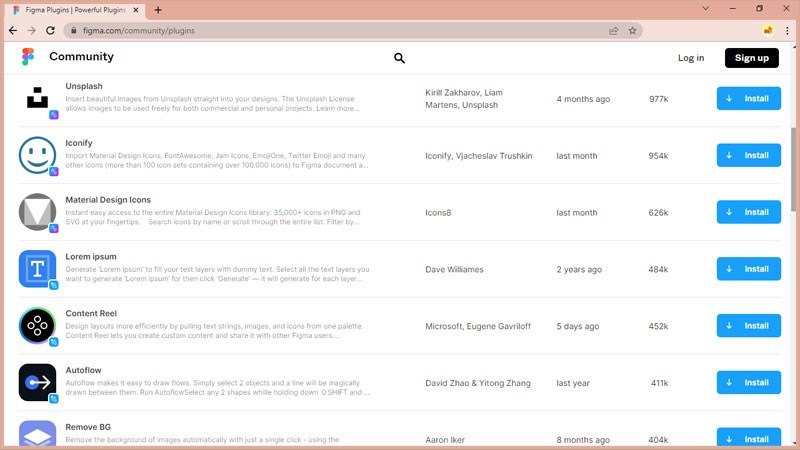
Hiện tại, Figma còn có các phiên bản ứng dụng trên máy tính và điện thoại để người dùng có thể sáng tạo hay theo dõi các dự án ở bất cứ nơi đâu*.*

***Tính cộng tác cao***

Ưu điểm hàng đầu của Figma phải kể đến tính cộng tác trong thời gian thực. Với Figma, làm việc nhóm từ xa trở nên dễ dàng khi nó cho phép những người khác trong dự án có thể cộng tác về thiết kế, cung cấp phản hồi, kiểm tra tiến độ công việc hoặc gắn cờ các vấn đề có thể xảy ra theo thời gian thực.

***Trang bị nhiều công cụ plugin mạnh mẽ***

Kho plugin khổng lồ và cực kỳ hữu ích chỉ có ở Figma sẽ giúp bạn khắc phục được những lỗi nhỏ và tăng hiệu suất làm việc. Việc quản lý màu sắc, nội dung hình ảnh hay hoạt ảnh đều trở nên dễ dàng hơn bao giờ hết.



Hình 2‑14 Các công cụ plugin của figma

***Thiết kế nhiều layout trong một sản phẩm***

Tính năng quản lý nhiều artboard cùng lúc cũng được tích hợp trên Figma. Với tính năng này, bạn có thể sáng tạo nhiều layout với các kích thước khác nhau trên cùng một sản phẩm mà không cần phải tạo thêm tệp mới.

***Xuất được đa dạng file ảnh cực sắc nét***

Tương tự như Illustrator, Figma thiết kế hình ảnh dưới dạng vector nên chất lượng của hình ảnh sẽ không bị giảm khi bạn thay đổi kích thước của chúng. Hình ảnh xuất ra cũng sẽ giữ được độ sắc nét với nhiều định dạng khác nhau như PNG, SVG, JPG hay thậm chí là PDF.

Hỗ trợ lưu trữ đám mây

Figma sử dụng dịch vụ đám mây để lưu trữ và chỉnh sửa dữ liệu, điều này tương tự như khi bạn thay đổi nội dung trên Google Docs. Nên sẽ giảm thiểu việc ổ cứng đầy dung lượng hay quên sao lưu vì Figma sẽ thực hiện tự động.

Với các lợi thế này thì có thể nói rằng Figma là một phần mềm hoàn hảo cho designer, nhất là trong lĩnh vực UI/UX có thể thiết kế trước giao diện của một trang web trước khi bắt tay vào code. Ngoài ra Figma có tất cả các công cụ hỗ trợ và tính năng mạnh mẽ nhất giúp designer có thể thiết kế và quản lý các dự án một cách hiệu quả. Đặc biệt, Figma nó có phiên bản miễn phí dành cho tất cả người dùng.

## Speckle platform

### Tổng quan về speckle



Hình 2‑15 Speckle server

Speckle là một nền tảng dữ liệu mã nguồn mở tiên phong, được thiết kế đặc biệt cho ngành Kiến trúc, Kỹ thuật và Xây dựng (AEC). Mục tiêu cốt lõi của Speckle là vượt qua những rào cản về khả năng tương tác dữ liệu giữa các phần mềm BIM (Building Information Modeling) khác nhau và các công cụ làm việc chuyên ngành khác.

Trong một ngành mà thông tin dự án thường bị "giam cầm" trong các phần mềm riêng biệt, Speckle mang đến một giải pháp trung tâm để chia sẻ, quản lý và truy cập dữ liệu dự án một cách linh hoạt và hiệu quả. Điều này tạo điều kiện thuận lợi cho sự cộng tác liền mạch giữa các đội ngũ thiết kế, kỹ sư, nhà thầu và các bên liên quan khác trong suốt vòng đời dự án.

***2.3.1.1. Các Thành Phần Chính của Speckle:***

Speckle bao gồm các thành phần chính sau, hoạt động cùng nhau để tạo ra một hệ sinh thái dữ liệu mạnh mẽ:

* Speckle Server: Đây là "trái tim" của nền tảng, một ứng dụng web mạnh mẽ (xây dựng trên Node.js) đóng vai trò là trung tâm quản lý dữ liệu. Nó chịu trách nhiệm lưu trữ, quản lý phiên bản, kiểm soát truy cập và cung cấp API (giao diện lập trình ứng dụng) để các ứng dụng khác có thể tương tác với dữ liệu Speckle.
* Speckle Connectors: Đây là các "cầu nối" được phát triển cho các phần mềm AEC phổ biến như Revit, Rhino, Grasshopper, AutoCAD, Blender, Excel, Power BI và nhiều ứng dụng khác. Các Connectors cho phép người dùng gửi (send) dữ liệu từ phần mềm họ đang sử dụng lên Speckle Server và nhận (receive) dữ liệu từ Speckle Server vào phần mềm khác.
* Speckle Viewer: Một trình xem 3D trực tuyến mạnh mẽ, tích hợp trực tiếp trên web. Viewer cho phép người dùng xem, điều hướng, kiểm tra và phân tích các mô hình 3D được chia sẻ qua Speckle mà không cần cài đặt bất kỳ phần mềm chuyên dụng nào. Đây là một công cụ quan trọng để giao tiếp trực quan và đưa ra quyết định dựa trên mô hình.
* Speckle SDKs (Software Development Kits): Speckle cung cấp các bộ công cụ phát triển phần mềm (SDK) cho các ngôn ngữ lập trình phổ biến như .NET (C#) và Python. SDK cho phép các nhà phát triển xây dựng các ứng dụng tùy chỉnh hoặc tích hợp Speckle vào các quy trình làm việc hiện có của họ.

***b. Giá Trị Cốt Lõi của Speckle:***

Khả năng Tương Tác Dữ Liệu (Interoperability): Giải quyết vấn đề "tường lửa" dữ liệu giữa các phần mềm khác nhau, cho phép thông tin dự án lưu chuyển một cách trơn tru.

Cộng tác Hiệu Quả: Tạo một nền tảng chung để các bên liên quan có thể cùng nhau xem, chia sẻ và làm việc trên dữ liệu dự án, bất kể họ đang sử dụng phần mềm nào.

Quản lý Dữ liệu Tập Trung: Cung cấp một nơi duy nhất để lưu trữ và quản lý tất cả dữ liệu dự án BIM, đảm bảo tính nhất quán và dễ dàng truy cập.

Tính Linh Hoạt và Mở Rộng: Kiến trúc mã nguồn mở cho phép tùy chỉnh và mở rộng nền tảng để đáp ứng các nhu cầu cụ thể của từng tổ chức và dự án.

Speckle là một nền tảng dữ liệu đột phá cho ngành AEC, mang lại khả năng tương tác, cộng tác và quản lý dữ liệu BIM một cách hiệu quả hơn. Với khả năng xem mô hình 3D trực tuyến mạnh mẽ thông qua Speckle Viewer và khả năng kết nối với nhiều công cụ khác nhau, Speckle giúp các đội ngũ dự án làm việc thông minh hơn, giảm thiểu lỗi và đưa ra các quyết định dựa trên thông tin chính xác và dễ dàng tiếp cận.

### Speckle Viewer: Xem mô hình 3D trực tuyến

Speckle Viewer 3D là một trình xem mô hình ba chiều mạnh mẽ, trực quan và hoàn toàn dựa trên nền tảng web, được tích hợp chặt chẽ trong hệ sinh thái Speckle. Nó cho phép người dùng xem, khám phá, phân tích và cộng tác trên các mô hình BIM và dữ liệu AEC khác một cách dễ dàng, mà không cần cài đặt bất kỳ phần mềm chuyên dụng nào.

***a. Chức Năng Chính của Speckle Viewer 3D:***

***Truy Cập Dễ Dàng và Phổ Biến:***

* Dựa trên trình duyệt: Hoạt động trực tiếp trên mọi trình duyệt web hiện đại (Chrome, Firefox, Safari, Edge), loại bỏ rào cản về hệ điều hành và yêu cầu phần cứng cao.
* Truy cập mọi lúc, mọi nơi: Người dùng có thể xem mô hình từ bất kỳ thiết bị nào có kết nối internet (máy tính để bàn, laptop, máy tính bảng, điện thoại thông minh).
* Chia sẻ đơn giản: Mô hình có thể được chia sẻ dễ dàng thông qua liên kết (URL), cho phép nhiều bên liên quan xem cùng một dữ liệu.

***Hiển Thị Mô Hình 3D Chất Lượng Cao:***

* Hỗ trợ nhiều định dạng: Có khả năng hiển thị dữ liệu từ nhiều nguồn và định dạng khác nhau, bao gồm các định dạng BIM tiêu chuẩn như IFC, RVT (Revit), DWG (AutoCAD), cũng như dữ liệu từ các ứng dụng thiết kế khác như Rhino, Grasshopper, Blender, v.v. (thông qua Speckle Connectors).
* Hiệu suất tối ưu: Được xây dựng để xử lý các mô hình 3D phức tạp một cách mượt mà và hiệu quả.
* Tùy chỉnh hiển thị: Cung cấp các tùy chọn để điều chỉnh màu sắc, độ trong suốt, chế độ xem (ví dụ: khung dây, tô bóng), và ẩn/hiện các đối tượng.

***Công Cụ Điều Hướng và Khám Phá Mô Hình:***

* Điều khiển trực quan: Các thao tác xoay, thu phóng, di chuyển (pan) được thực hiện một cách dễ dàng bằng chuột hoặc cảm ứng.
* Chế độ xem phối cảnh và trực giao: Cho phép xem mô hình từ nhiều góc độ khác nhau.
* Công cụ cắt mặt phẳng (section planes): Tạo các mặt cắt ngang và dọc để xem chi tiết bên trong mô hình.
* Đo lường: Cung cấp các công cụ để đo khoảng cách, diện tích và thể tích trực tiếp trên mô hình.
* Đi bộ ảo (walkthrough): Cho phép di chuyển qua mô hình như đang ở trong không gian thực.

***Truy Cập Thông Tin Thuộc Tính (Properties):***

* Xem chi tiết đối tượng: Khi chọn một đối tượng trong mô hình, người dùng có thể xem tất cả các thông tin thuộc tính (metadata) được liên kết với nó (ví dụ: vật liệu, kích thước, ID, thông tin kỹ thuật).
* Tìm kiếm và lọc: Cho phép tìm kiếm các đối tượng dựa trên thuộc tính và lọc hiển thị theo các tiêu chí cụ thể.

***Hỗ Trợ Cộng Tác:***

* Xem đồng thời: Nhiều người dùng có thể xem cùng một mô hình cùng một lúc.
* Chú thích và đánh dấu (annotations): Cho phép thêm các ghi chú, hình vẽ và đánh dấu trực tiếp trên mô hình để chia sẻ ý kiến và thông tin.
* Bình luận (comments): Tạo các cuộc thảo luận theo ngữ cảnh liên quan đến các đối tượng cụ thể trong mô hình.

***Tích Hợp và Mở Rộng:***

* API (Application Programming Interface): Speckle Viewer cung cấp API cho phép các nhà phát triển tích hợp trình xem vào các ứng dụng web khác hoặc tùy chỉnh chức năng của nó.
* Webhooks và sự kiện: Hỗ trợ các cơ chế để phản ứng với các sự kiện xảy ra trong trình xem.

***b. Lợi Ích của Speckle Viewer 3D trong Quy Trình BIM:***

Cải thiện giao tiếp và hiểu biết chung: Tất cả các bên liên quan có thể xem và thảo luận về cùng một mô hình 3D, giảm thiểu sự hiểu lầm và đảm bảo thông tin được truyền đạt rõ ràng.

Tăng cường khả năng tiếp cận dữ liệu: Loại bỏ sự phụ thuộc vào phần mềm BIM chuyên dụng để xem mô hình, mở rộng khả năng tiếp cận thông tin dự án cho nhiều đối tượng hơn (ví dụ: quản lý dự án, chủ đầu tư, khách hàng).

Hỗ trợ ra quyết định hiệu quả hơn: Khả năng xem xét mô hình trực quan và truy cập thông tin thuộc tính giúp các nhà quản lý và các thành viên khác đưa ra các quyết định sáng suốt hơn dựa trên dữ liệu thực tế.

Tiết kiệm thời gian và chi phí: Giảm thiểu thời gian và chi phí liên quan đến việc xuất nhập khẩu dữ liệu giữa các phần mềm khác nhau và việc chia sẻ tệp tin lớn.

Nâng cao hiệu quả cộng tác: Các công cụ chú thích và bình luận trực tiếp trên mô hình tạo điều kiện cho việc trao đổi thông tin và giải quyết vấn đề một cách nhanh chóng và hiệu quả.

Tính linh hoạt và khả năng mở rộng: Dễ dàng tích hợp vào các quy trình làm việc hiện có và có khả năng được tùy chỉnh để đáp ứng các nhu cầu cụ thể của dự án.

Speckle Viewer 3D không chỉ là một công cụ để xem mô hình BIM, mà còn là một nền tảng mạnh mẽ để giao tiếp, cộng tác và đưa ra quyết định dựa trên dữ liệu trực quan. Khả năng truy cập dễ dàng, các công cụ khám phá mạnh mẽ và khả năng tích hợp cao làm cho nó trở thành một thành phần không thể thiếu trong quy trình làm việc BIM hiện đại.

### Ứng dụng của Speckle Viewer trong quản lý BIM

Speckle Viewer 3D không chỉ là một công cụ hiển thị mô hình BIM mà còn là một nền tảng tương tác trực quan, đóng vai trò trung tâm trong việc tối ưu hóa quy trình làm việc, tăng cường giao tiếp và nâng cao hiệu quả quản lý dự án trong suốt vòng đời của một công trình xây dựng. Dưới đây là các ứng dụng toàn diện của Speckle Viewer 3D trong phần mềm BIM:

***a. Truy cập và phổ cập dữ liệu mô hình:***

* Đơn giản hóa việc xem mô hình: Loại bỏ rào cản về phần mềm chuyên dụng, cho phép mọi thành viên dự án, từ kiến trúc sư, kỹ sư, nhà thầu, đến chủ đầu tư và các bên liên quan không chuyên, dễ dàng truy cập và xem mô hình 3D thông qua trình duyệt web trên mọi thiết bị.
* Tiếp cận dữ liệu mọi lúc mọi nơi: Hỗ trợ làm việc từ xa, tại công trường hoặc trong các cuộc họp, đảm bảo mọi người đều có thể tham khảo mô hình mới nhất để đưa ra quyết định kịp thời.
* Chia sẻ tức thì: Việc chia sẻ mô hình thông qua liên kết đơn giản hóa quá trình truyền đạt thông tin, loại bỏ sự phức tạp của việc gửi và quản lý các tệp tin lớn.

***b. Khám phá và phân tích chi tiết mô hình:***

* Điều hướng trực quan và linh hoạt: Cung cấp các công cụ điều khiển 3D mạnh mẽ (xoay, thu phóng, di chuyển) giúp người dùng khám phá mọi ngóc ngách của mô hình.
* Xem xét cấu kiện chuyên sâu: Cho phép lựa chọn và xem xét chi tiết từng cấu kiện, hệ thống (kiến trúc, kết cấu, MEP), và các mối nối, giúp hiểu rõ về thiết kế và thi công.
* Sử dụng mặt cắt động: Tạo và điều chỉnh các mặt cắt ngang, dọc một cách linh hoạt để xem xét cấu trúc bên trong và sự phối hợp giữa các hệ thống.
* Đo lường chính xác: Tích hợp các công cụ đo khoảng cách, diện tích và thể tích trực tiếp trên mô hình, hỗ trợ việc kiểm tra kích thước và số lượng.
* Cho phép đánh dấu điểm cần xác định trên mô hình
* Mô thành viên trong dự án có thể comment các chi tiết của mô hình
* Gắn trục tọa độ Oxyz vào vật thể, phân tích các chi tiết của

***c. Khai thác thông tin thuộc tính:***

* Truy cập dữ liệu phi hình học: Khi chọn một đối tượng, người dùng có thể xem ngay lập tức các thông tin thuộc tính (metadata) được nhúng trong mô hình từ phần mềm BIM gốc, bao gồm vật liệu, thông số kỹ thuật, nhà sản xuất, mã hiệu, v.v.
* Lọc và phân tích dữ liệu: Một số triển khai nâng cao cho phép lọc và hiển thị các đối tượng dựa trên các thuộc tính cụ thể, hỗ trợ các hoạt động phân tích như kiểm tra tuân thủ tiêu chuẩn hoặc quản lý tài sản.

**d. Tăng cường cộng tác và giao tiếp:**

* Chú thích và đánh dấu trực tiếp: Cho phép người dùng thêm các ghi chú văn bản, hình vẽ và đánh dấu trực tiếp trên mô hình 3D để chỉ ra các vấn đề, đặt câu hỏi hoặc đề xuất thay đổi một cách trực quan.
* Bình luận theo ngữ cảnh: Tạo các cuộc thảo luận được liên kết trực tiếp với các đối tượng cụ thể trong mô hình, đảm bảo mọi người đều hiểu rõ ngữ cảnh của vấn đề đang được thảo luận.
* Xem đồng thời và theo dõi thay đổi: Nhiều người dùng có thể xem cùng một mô hình và theo dõi các thay đổi được thực hiện theo thời gian, tạo ra một không gian làm việc cộng tác hiệu quả.

**e. Hỗ trợ quản lý dự án và ra quyết định:**

* Theo dõi tiến độ trực quan: So sánh các phiên bản mô hình theo thời gian để theo dõi tiến độ xây dựng và xác định các khu vực chậm trễ hoặc có vấn đề.
* Phân tích xung đột: Mặc dù không phải là công cụ phát hiện xung đột chuyên dụng, Speckle Viewer có thể giúp người dùng trực quan hóa và xác định các xung đột tiềm ẩn giữa các hệ thống.
* Trình bày dự án hiệu quả: Sử dụng khả năng điều hướng mượt mà và hiển thị chất lượng cao để trình bày ý tưởng thiết kế và tiến độ dự án với khách hàng và các bên liên quan.
* Thu thập phản hồi dễ dàng: Các công cụ chú thích và bình luận tạo điều kiện thuận lợi cho việc thu thập phản hồi từ các bên liên quan và tích hợp chúng vào quá trình thiết kế và xây dựng.

***f. Tích Hợp và Mở Rộng:***

* Khả năng tích hợp cao: Thông qua API, Speckle Viewer có thể được nhúng vào các nền tảng quản lý dự án, các ứng dụng web nội bộ và các giải pháp phần mềm khác, tạo ra một luồng thông tin liền mạch.
* Tùy chỉnh và mở rộng chức năng: Nền tảng Speckle mở cho phép các nhà phát triển tùy chỉnh và mở rộng chức năng của Viewer để đáp ứng các nhu cầu cụ thể của từng dự án hoặc tổ chức.

# PHÂN TÍCH HỆ THỐNG

## Khảo sát bài toán

Trong quá trình khảo sát thực tế việc quản lý và chia sẻ dữ liệu mô hình 3D trong các dự án BIM, chúng tôi nhận thấy nhiều khó khăn và hạn chế trong các quy trình làm việc truyền thống.

***Đối với người dùng (Kiến trúc sư, Kỹ sư, Nhà quản lý dự án):***

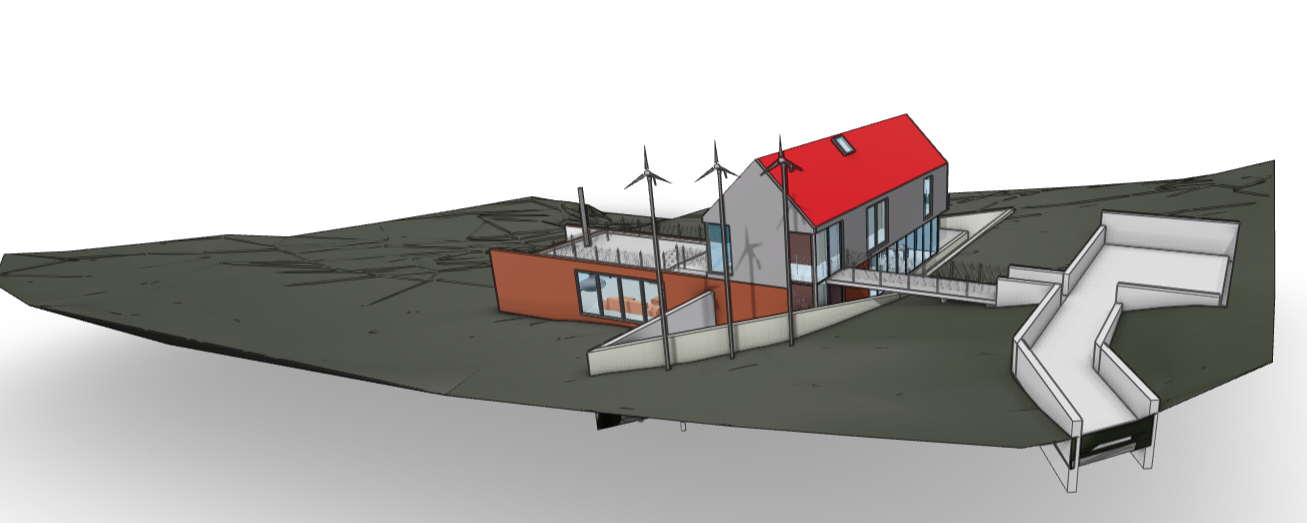
* Khó khăn trong việc chia sẻ dữ liệu: Việc chia sẻ mô hình 3D giữa các thành viên dự án sử dụng các phần mềm BIM khác nhau thường phức tạp, đòi hỏi nhiều bước chuyển đổi định dạng, dễ gây mất mát thông tin và tốn thời gian.
* Thiếu một nền tảng quản lý tập trung: Dữ liệu mô hình 3D và các tài liệu liên quan thường nằm rải rác ở nhiều nơi (máy tính cá nhân, email, các hệ thống lưu trữ đám mây khác nhau), gây khó khăn trong việc tìm kiếm, truy cập và quản lý phiên bản.
* Khó khăn trong việc theo dõi lịch sử thay đổi: Việc theo dõi và quản lý các phiên bản khác nhau của mô hình 3D trong quá trình phát triển dự án là một thách thức, dễ dẫn đến nhầm lẫn và sử dụng thông tin không chính xác.
* Hạn chế trong việc cộng tác trực quan: Việc cùng nhau xem xét và đưa ra phản hồi trên mô hình 3D thường gặp khó khăn, đặc biệt khi các thành viên sử dụng các công cụ khác nhau hoặc không có quyền truy cập vào phần mềm BIM chuyên dụng.

**Đối với nhà quản lý dự án:**

* Khó khăn trong việc theo dõi tiến độ dựa trên mô hình: Việc nắm bắt tiến độ dự án và các vấn đề phát sinh dựa trên dữ liệu mô hình 3D thường không trực quan và đòi hỏi nhiều thời gian tổng hợp thông tin.
* Khó khăn trong việc kiểm soát quyền truy cập: Việc đảm bảo an toàn và bảo mật cho dữ liệu mô hình 3D, đặc biệt là khi chia sẻ với nhiều bên liên quan, là một thách thức.

### Quản lý và xem dữ liệu 3D trực tuyến trong BIM là gì?

Quản lý và xem dữ liệu 3D trực tuyến trong BIM là hệ thống cho phép người dùng tải lên, lưu trữ, quản lý, xem và chia sẻ các mô hình 3D và thông tin liên quan của dự án BIM trên nền tảng web. Hệ thống này cung cấp một môi trường tập trung để truy cập dữ liệu từ mọi nơi, kiểm soát phiên bản, quản lý quyền truy cập và hỗ trợ các công cụ xem trực quan để khám phá mô hình mà không cần phần mềm chuyên dụng.



Hình 3‑1 Mô hình 3D với speckle viewer

### 3.1.2. Lợi ích của quản lý và đặt bàn trực tuyến

Cải thiện khả năng tương tác dữ liệu: Dễ dàng chia sẻ và xem mô hình 3D giữa các thành viên dự án sử dụng các phần mềm BIM khác nhau, giảm thiểu các bước chuyển đổi phức tạp và nguy cơ mất mát thông tin.

Tăng cường cộng tác hiệu quả: Cung cấp một nền tảng chung để các bên liên quan cùng nhau xem xét, chú thích và thảo luận trực tiếp trên mô hình 3D.

Quản lý dữ liệu tập trung và có tổ chức: Lưu trữ tất cả dữ liệu mô hình 3D và thông tin liên quan ở một nơi duy nhất, giúp dễ dàng tìm kiếm, truy cập và quản lý phiên bản.

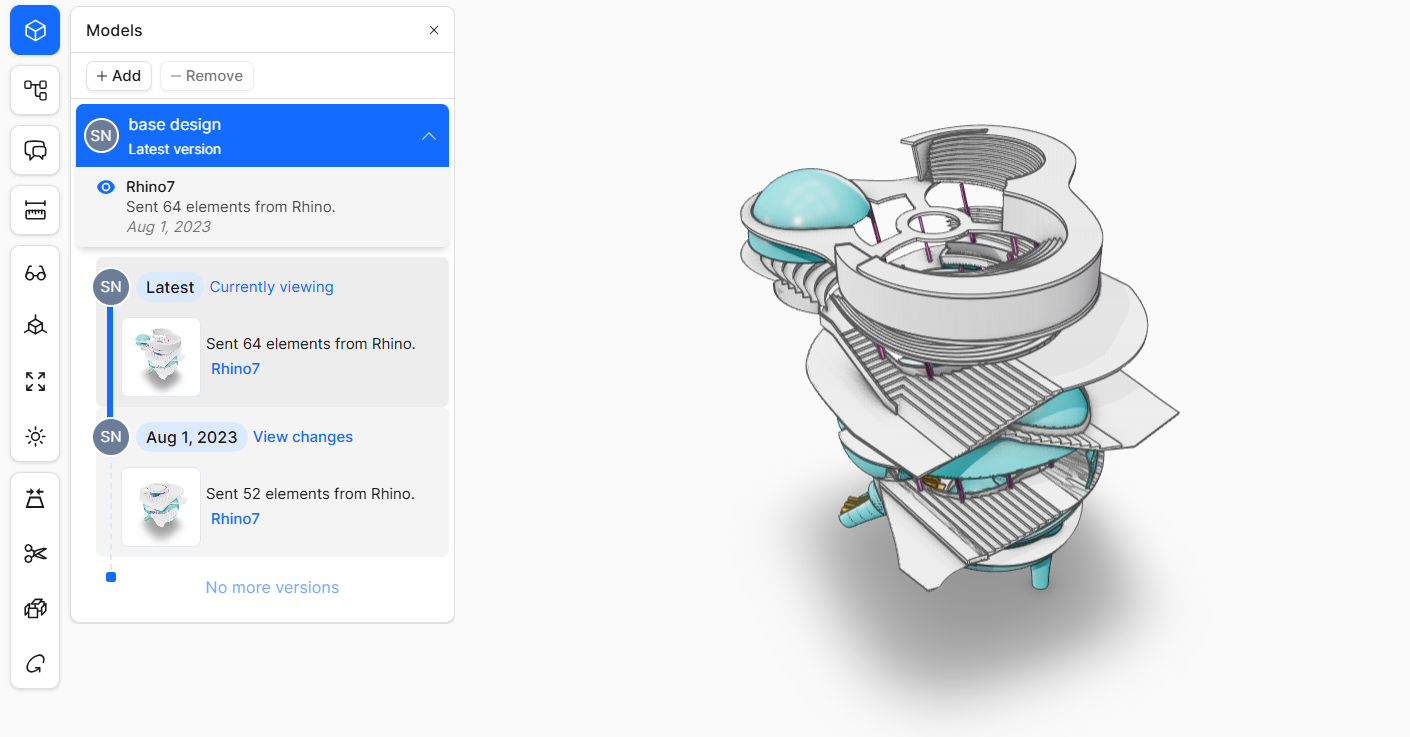
Nâng cao khả năng tiếp cận dữ liệu: Cho phép những người không có phần mềm BIM chuyên dụng vẫn có thể xem và hiểu được mô hình 3D thông qua trình duyệt web.

Hỗ trợ ra quyết định dựa trên thông tin trực quan: Khả năng xem mô hình 3D trực quan giúp các nhà quản lý và thành viên dự án đưa ra các quyết định chính xác hơn dựa trên dữ liệu thực tế.

### Các chức năng chính của hệ thống

Tải lên và quản lý mô hình 3D: Cho phép người dùng tải lên các tệp tin mô hình 3D từ nhiều phần mềm BIM khác nhau và quản lý chúng trong hệ thống.

Xem mô hình 3D trực tuyến: Cung cấp một trình xem 3D trực quan trên nền web để người dùng khám phá và kiểm tra mô hình mà không cần cài đặt phần mềm.

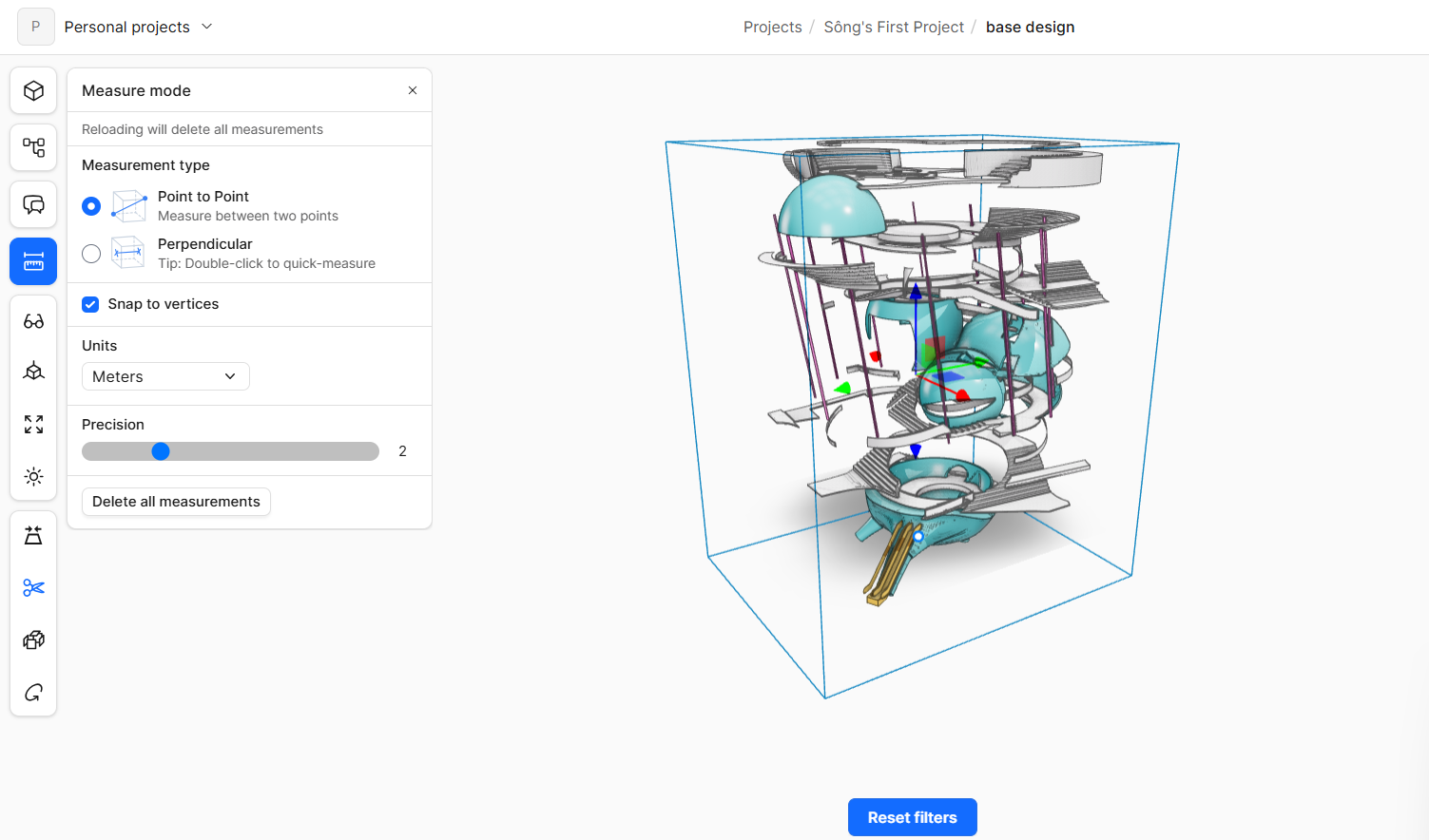


Hình 3‑2 Trực quan hóa mô hình 3D

Quản lý phiên bản: Theo dõi và quản lý các phiên bản khác nhau của mô hình 3D theo thời gian.

Quản lý người dùng và phân quyền: Cho phép quản trị viên tạo, chỉnh sửa và quản lý tài khoản người dùng, đồng thời thiết lập các quyền truy cập khác nhau đối với dữ liệu và chức năng của hệ thống.

Chú thích và bình luận: Cho phép người dùng thêm các ghi chú và bình luận trực tiếp trên mô hình 3D để giao tiếp và làm việc cộng tác.



Hình 3‑3 Các chức năng chính với viewer 3D

## Giới thiệu bài toán

### 3.2.1. Xác định các tác nhân

Hiện nay, nhu cầu quản lý hiệu quả và chia sẻ trực quan dữ liệu mô hình 3D trong các dự án BIM ngày càng trở nên cấp thiết nhằm nâng cao hiệu quả cộng tác và tối ưu hóa quy trình làm việc. Đề tài "*Xây dựng phân hệ quản trị cho phần mềm quản lý dữ liệu BIM"* hướng đến việc tạo ra một nền tảng trực tuyến cho phép:

* User (Kiến trúc sư, Kỹ sư, khách hàng): Dễ dàng tải lên, xem, quản lý và chia sẻ mô hình 3D từ nhiều nguồn khác nhau trên một giao diện web trực quan.
* Moderator: Quản lý người dùng, dự án và quyền truy cập vào dữ liệu, đảm bảo an toàn và bảo mật thông tin.
* Admin: có quyền cao nhất về quản lý người dùng, quản lý dữ liệu, quản lý phân quyền

Mục tiêu của hệ thống là tạo ra một môi trường dữ liệu chung (CDE) hiệu quả, giúp các bên liên quan dễ dàng truy cập, tương tác và cộng tác trên dữ liệu mô hình 3D, từ đó tối ưu hóa quy trình làm việc BIM và nâng cao chất lượng dự án.

### 3.2.2. Yêu cầu các chức năng

#### **3.2.2.1. User**

* Đăng ký / đăng nhập
* Khởi tạo dự án
* Upload file
* Quản lý dự án (Thêm, sửa, xóa, update)
* Cấp quyền cho các người dùng khác vào quản lý dự án

#### **3.2.2.2. Moderator**

* Đăng nhập/ đăng xuất
* Quản lý người dùng
* Quản lý các dự án

#### **3.2.2.3. Admin**

* Đăng nhập/ đăng xuất
* Quản lý người dùng
* Quản lý dự án
* Cấp quyền quản lý (Admin/Moderator/User)

## Đặc tả yêu cầu chức năng

### Quản lý người dùng:

Đăng ký/Đăng nhập: Cho phép người dùng (kiến trúc sư, kỹ sư, nhà quản lý dự án) và quản trị viên đăng ký tài khoản và đăng nhập vào hệ thống.

Quản lý hồ sơ: Cho phép người dùng cập nhật thông tin cá nhân, mật khẩu và các thông tin liên quan đến tài khoản.

Quản lý người dùng (Admin): Cho phép quản trị viên tạo, chỉnh sửa, xóa và quản lý tài khoản của người dùng.

### Quản lý dự án:

Tạo/Chỉnh sửa/Xóa dự án: Cho phép người dùng và quản trị viên tạo, chỉnh sửa thông tin và xóa các dự án BIM.

Xem danh sách dự án: Hiển thị danh sách tất cả các dự án có trong hệ thống.

### Quản lý dữ liệu mô hình 3D:

Tải lên mô hình 3D: Cho phép người dùng tải lên các tệp tin mô hình 3D từ các phần mềm BIM khác nhau (ví dụ: IFC, RVT).

Xem mô hình 3D trực tuyến: Cung cấp một trình xem 3D tích hợp để người dùng xem và điều hướng mô hình trực tiếp trên web.

Quản lý phiên bản mô hình: Theo dõi và hiển thị lịch sử các phiên bản của từng mô hình 3D, cho phép người dùng xem và khôi phục các phiên bản trước đó.

Tải xuống mô hình 3D: Cho phép người dùng tải xuống các tệp tin mô hình 3D.

### Quản lý quyền truy cập:

Phân quyền theo dự án: Cho phép quản trị viên thiết lập các quyền truy cập khác nhau (ví dụ: xem, tải xuống, chỉnh sửa, quản lý) cho người dùng đối với từng dự án cụ thể.

Phân quyền theo vai trò: Định nghĩa các vai trò khác nhau (Admin, Moderator, User) và gán quyền hạn tương ứng cho từng vai trò.

### Công cụ hỗ trợ cộng tác:

Chú thích trên mô hình: Cho phép người dùng thêm các chú thích văn bản hoặc hình vẽ trực tiếp trên mô hình 3D để chia sẻ thông tin và đặt câu hỏi.

Bình luận: Cho phép người dùng tạo và theo dõi các cuộc thảo luận liên quan đến các mô hình hoặc dự án cụ thể.

## 3.4. Đặc tả yêu cầu phi chức năng

### 3.4.1. Hiệu năng và khả năng mở rộng:

Tốc độ tải mô hình: Hệ thống phải tải và hiển thị mô hình 3D một cách nhanh chóng và hiệu quả, đặc biệt đối với các mô hình có kích thước lớn.

Khả năng xử lý đồng thời: Hỗ trợ nhiều người dùng truy cập và thao tác đồng thời mà không làm giảm đáng kể hiệu năng hệ thống.

Khả năng mở rộng: Kiến trúc hệ thống phải cho phép dễ dàng mở rộng tài nguyên (ví dụ: bộ nhớ, băng thông, dung lượng lưu trữ) để đáp ứng sự gia tăng về lượng dữ liệu và người dùng.

### 3.4.2. Bảo mật:

Bảo vệ dữ liệu mô hình: Đảm bảo dữ liệu mô hình 3D và thông tin liên quan được bảo mật, tránh truy cập trái phép và rò rỉ thông tin.

Xác thực và phân quyền: Sử dụng các cơ chế xác thực mạnh mẽ và phân quyền chi tiết để kiểm soát quyền truy cập của người dùng vào các chức năng và dữ liệu.

Mã hóa dữ liệu: Áp dụng các phương pháp mã hóa phù hợp cho dữ liệu nhạy cảm trong quá trình truyền tải và lưu trữ.

### 3.4.3. Độ tin cậy và khả năng phục hồi:

Tính ổn định: Hệ thống phải hoạt động ổn định và có khả năng xử lý lỗi một cách hiệu quả.

Khả năng phục hồi: Có cơ chế tự động phục hồi hoặc quy trình rõ ràng để khôi phục hệ thống và dữ liệu sau sự cố.

Sao lưu dữ liệu: Thực hiện sao lưu dữ liệu định kỳ để đảm bảo an toàn thông tin trong trường hợp xảy ra lỗi hệ thống hoặc sự cố không mong muốn.

### 3.4.4. Giao diện và trải nghiệm người dùng:

Thân thiện với người dùng: Giao diện web trực quan, dễ hiểu và dễ sử dụng cho người dùng với các trình độ kỹ năng khác nhau.

Khả năng tương thích đa trình duyệt: Hệ thống phải tương thích tốt với các trình duyệt web phổ biến như Chrome, Firefox, Safari và Edge.

Thiết kế đáp ứng (Responsive Design): Giao diện phải được tối ưu hóa để hiển thị tốt trên nhiều loại thiết bị với kích thước màn hình khác nhau (máy tính để bàn, laptop, máy tính bảng).

### 3.4.5. Bảo trì và nâng cấp:

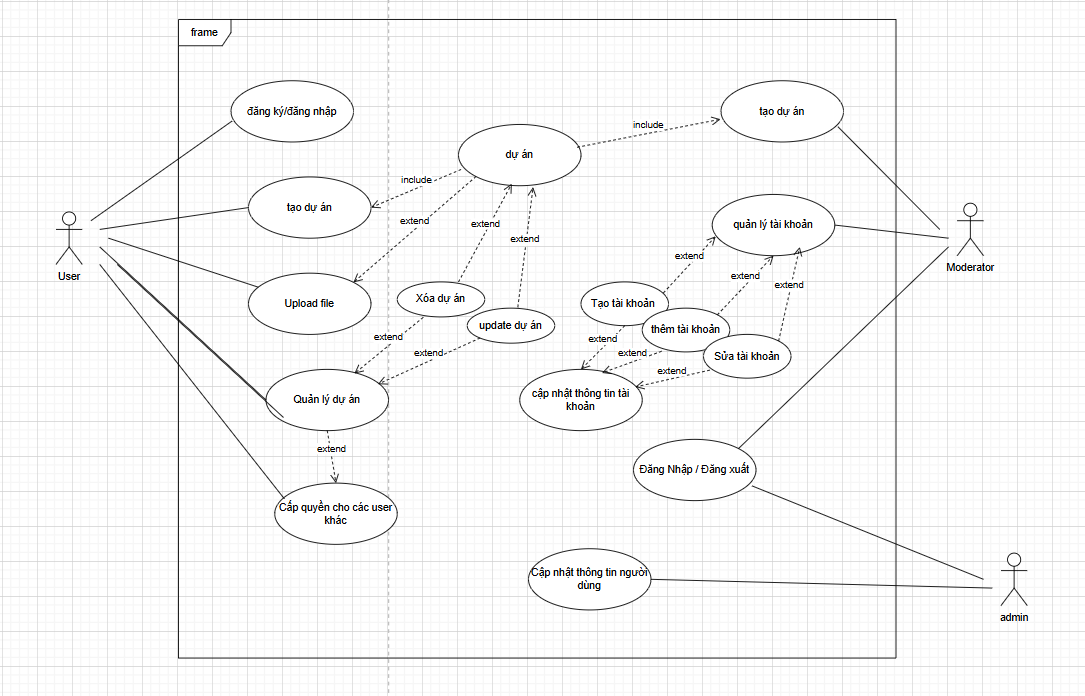
Dễ bảo trì: Cấu trúc mã nguồn rõ ràng, dễ hiểu và có tài liệu đầy đủ để thuận tiện cho việc bảo trì và sửa lỗi.

Khả năng nâng cấp: Thiết kế hệ thống phải linh hoạt, cho phép dễ dàng tích hợp các tính năng mới và nâng cấp các thành phần hiện có mà không gây ảnh hưởng lớn đến hệ thống đang hoạt động.

Ghi nhật ký hoạt động: Ghi lại các hoạt động quan trọng của người dùng và hệ thống để phục vụ cho việc theo dõi, gỡ lỗi và kiểm soát an ninh.

## 3.5. Phân tích biểu đồ ca sử dụng

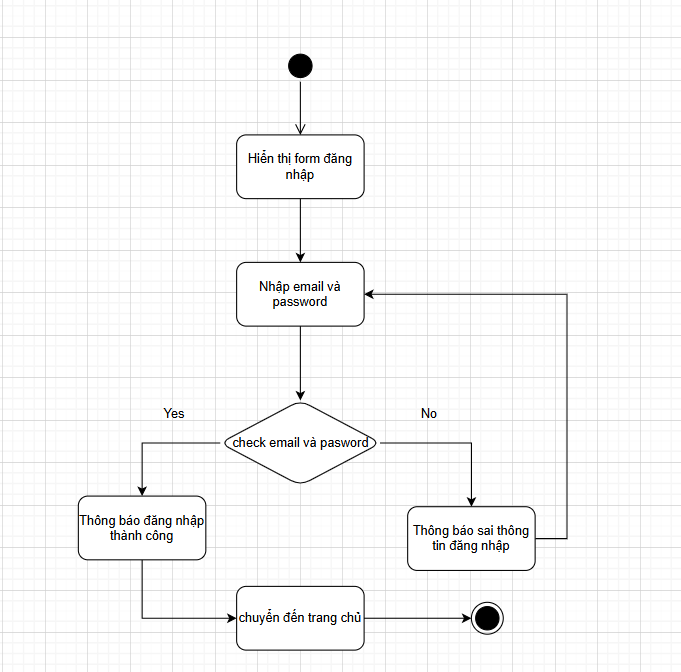
### 3.5.1. Biểu đồ ca sử dụng tổng quát



Hình 3. 1. Biểu đồ use case tổng quát

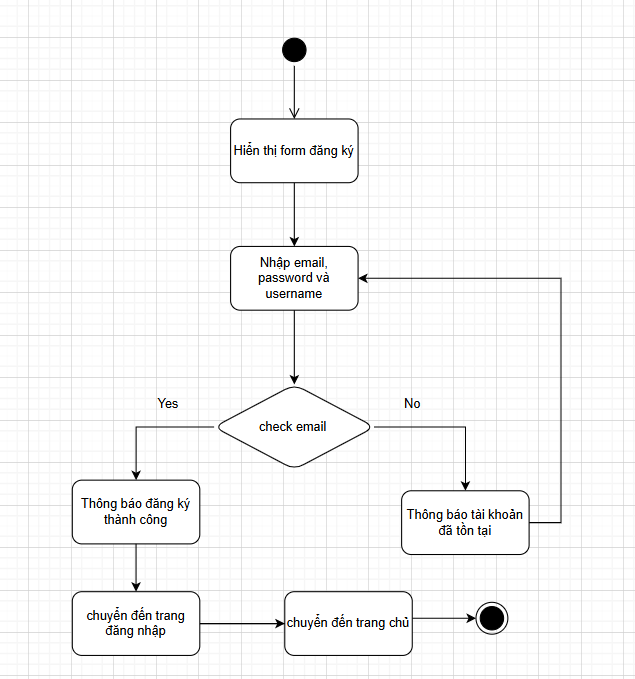
### 3.5.2. Sơ đồ hoạt động

#### **3.5.2.1. Sơ đồ hoạt động đăng nhập**



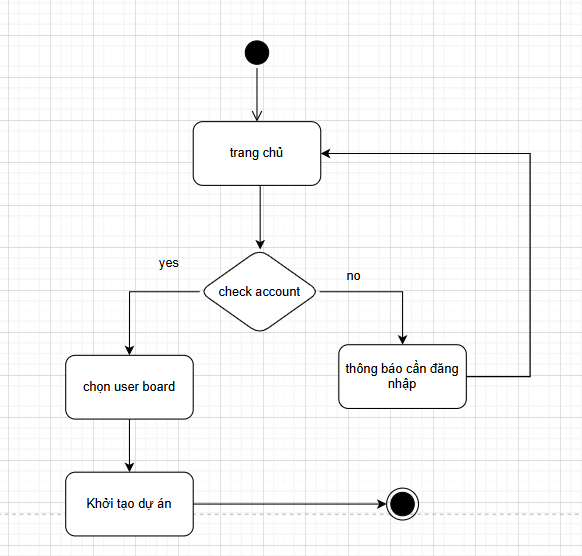
Hình 3. 2. Sơ đồ hoạt động đăng nhập

#### **3.5.2.2. Sơ đồ hoạt động đăng ký**



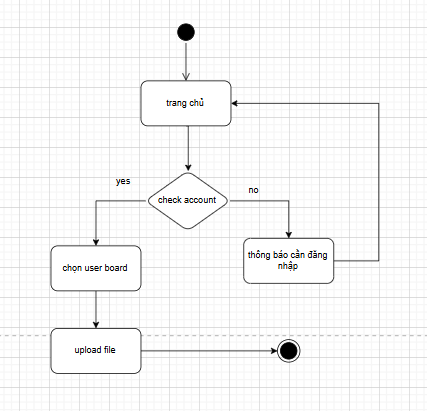
Hình 3. 3. Sơ đồ hoạt động đăng ký

#### **3.5.2.3. Sơ đồ hoạt động tạo dự án**



Hình 3. 4. Sơ đồ hoạt động tạo dự án

**3.5.2.4. Sơ đồ hoạt động upload file**



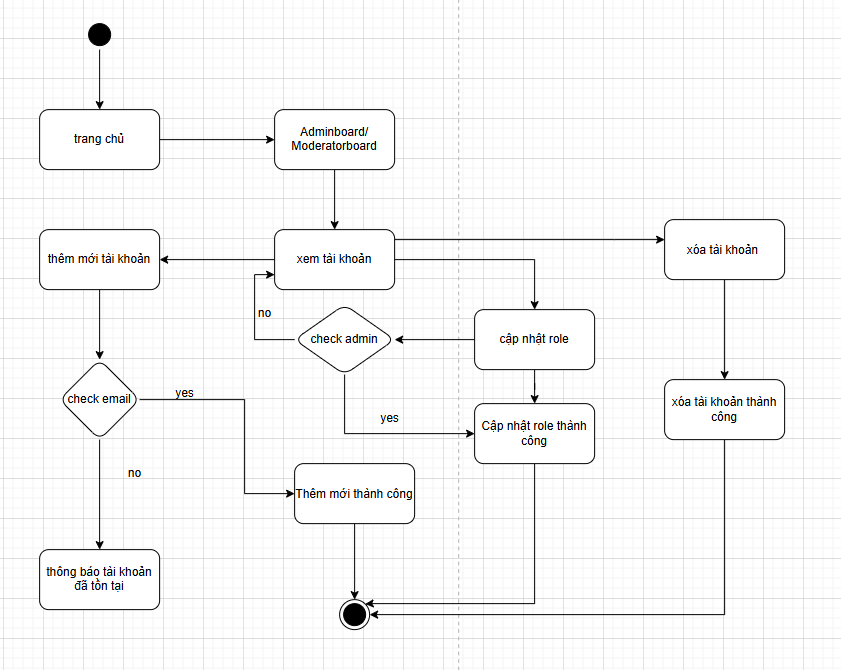
Hình 3. 5. Biểu đồ use case chức năng lịch sử đặt bàn

#### **3.5.2.5. Sơ đồ hoạt động thêm thành viên vào dự án**



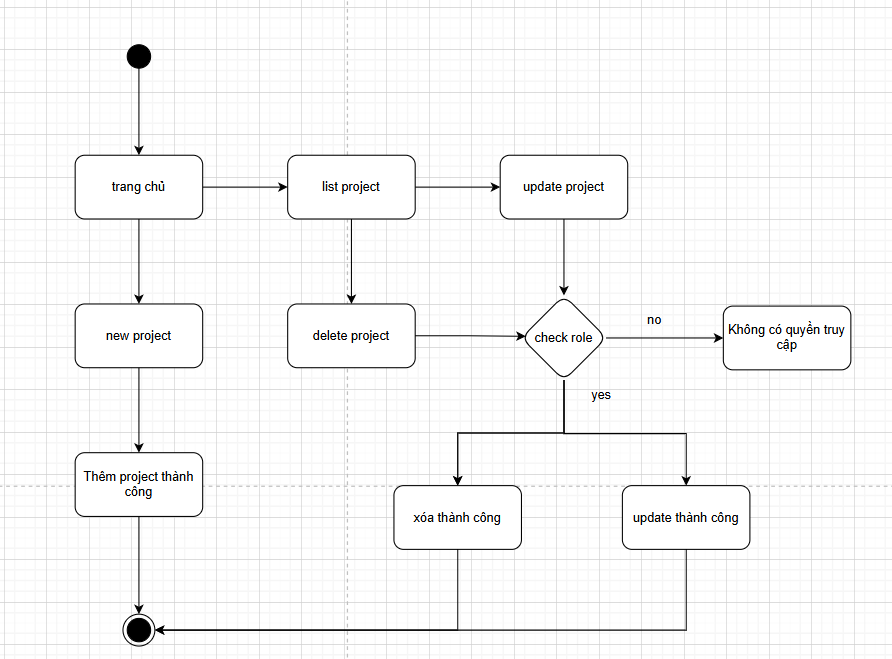
Hình 3. 6. Sơ đồ hoạt động thêm thành viên vào dự án

#### **3.5.2.6. Sơ đồ hoạt động quản lý tài khoản người dùng**



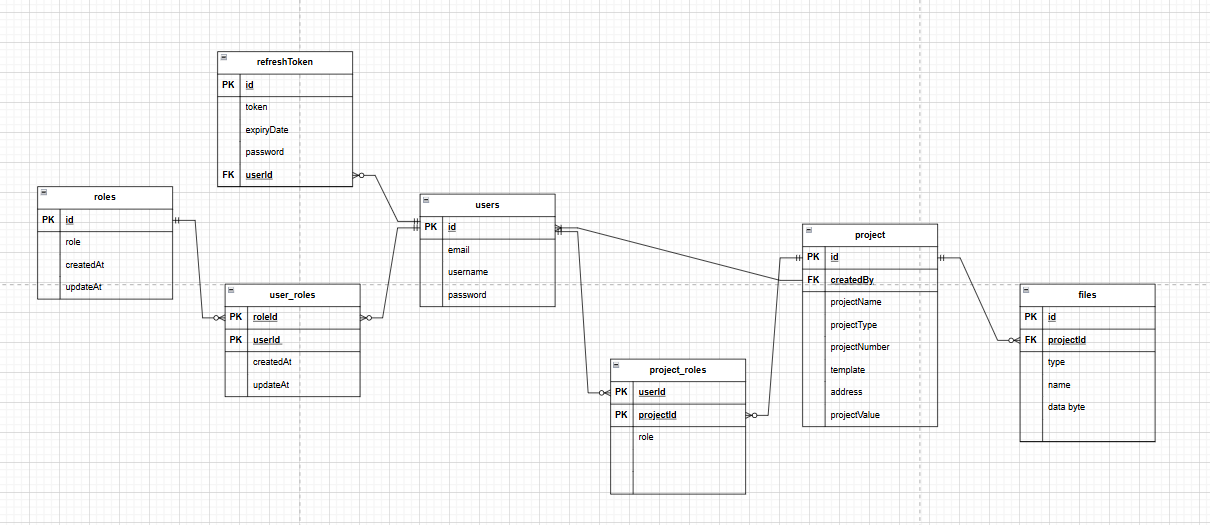
*Hình 3.7. Sơ đồ hoạt động quản lý tài khoản người dùng*

#### **3.5.2.7. Sơ đồ chức năng quản lý dự án**



Hình 3. 8. Sơ đồ chức năng quản lý dự án

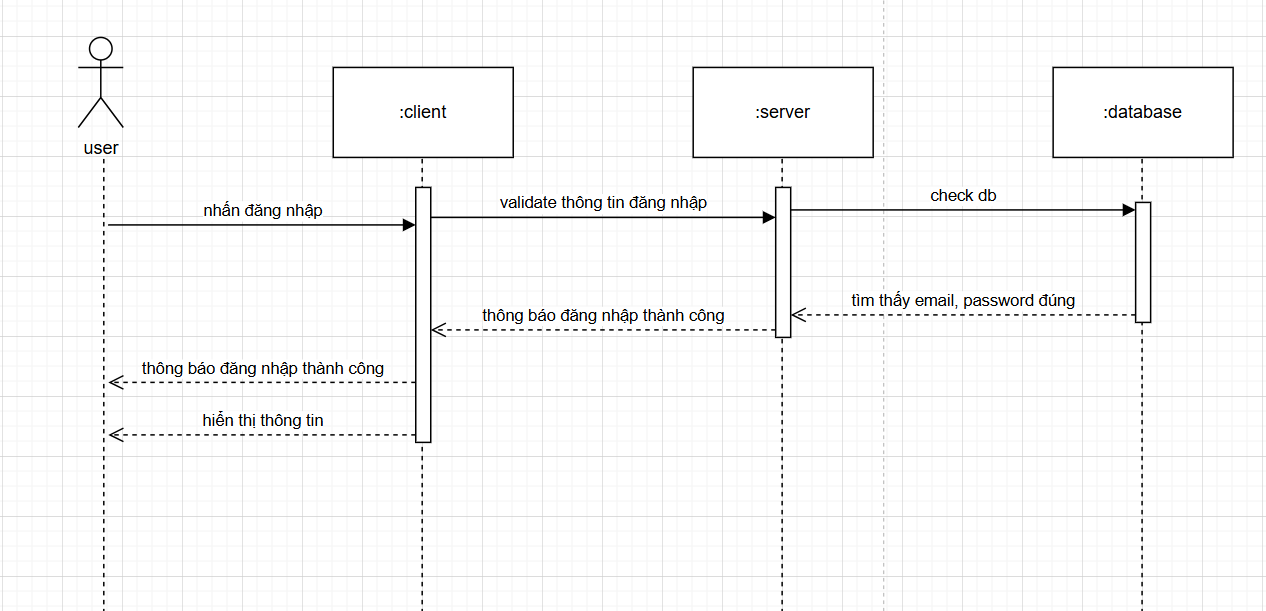
3.5.3. Sơ đồ thực thể quan hệ (ERD)



Hình 3.9. Sơ đồ thực thể quan hệ (ERD)

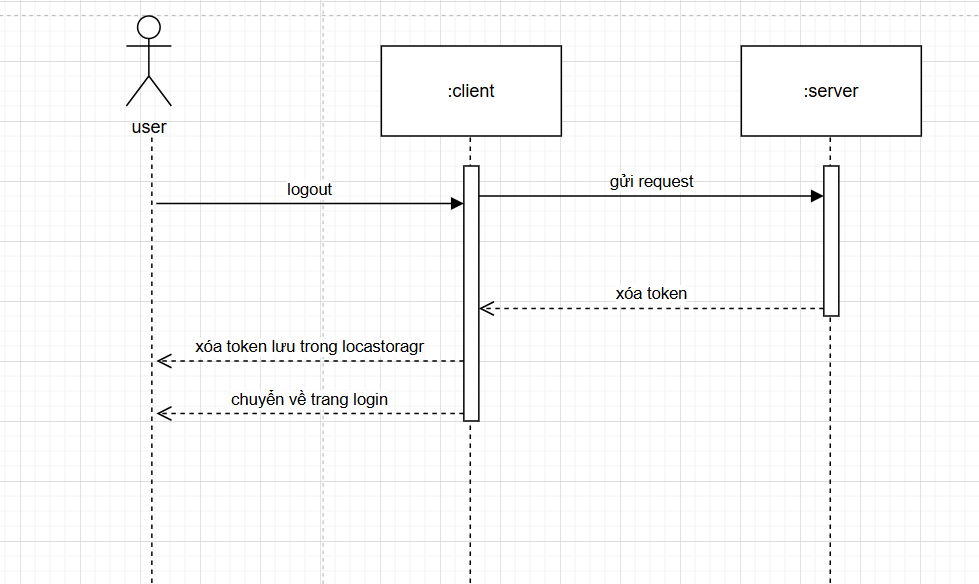
### 3.5.4. Biểu đồ tuần tự

#### **3.5.4.1 Sơ đồ tuần tự đăng nhập và đăng ký**



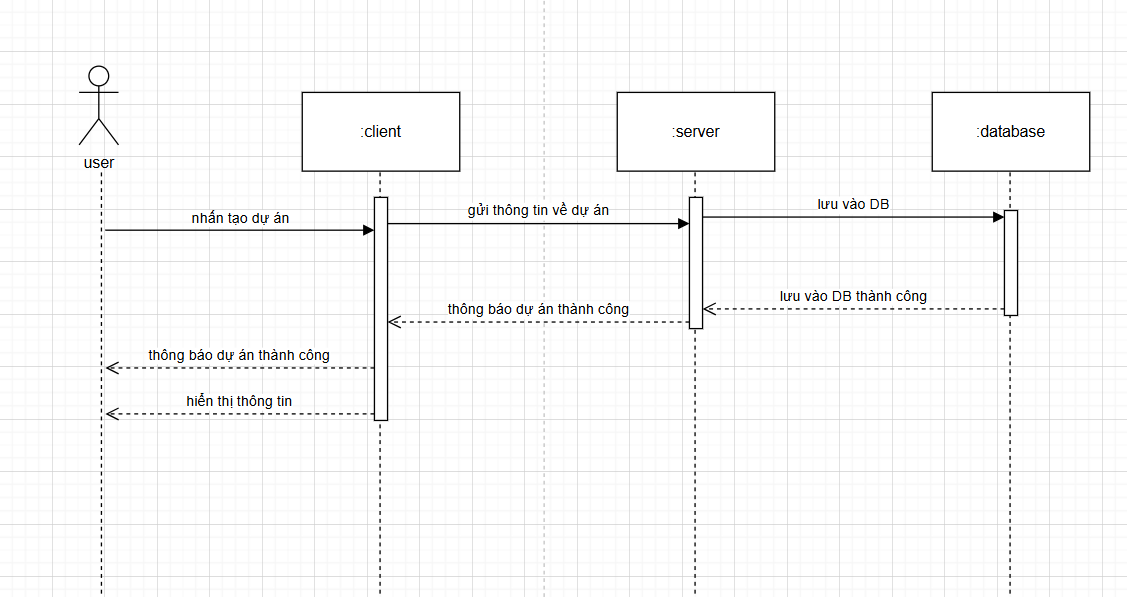
Hình 3. 11.Sơ đồ tuần tự đăng nhập

3.5.4.2. Sơ đồ tuần tự đăng xuất



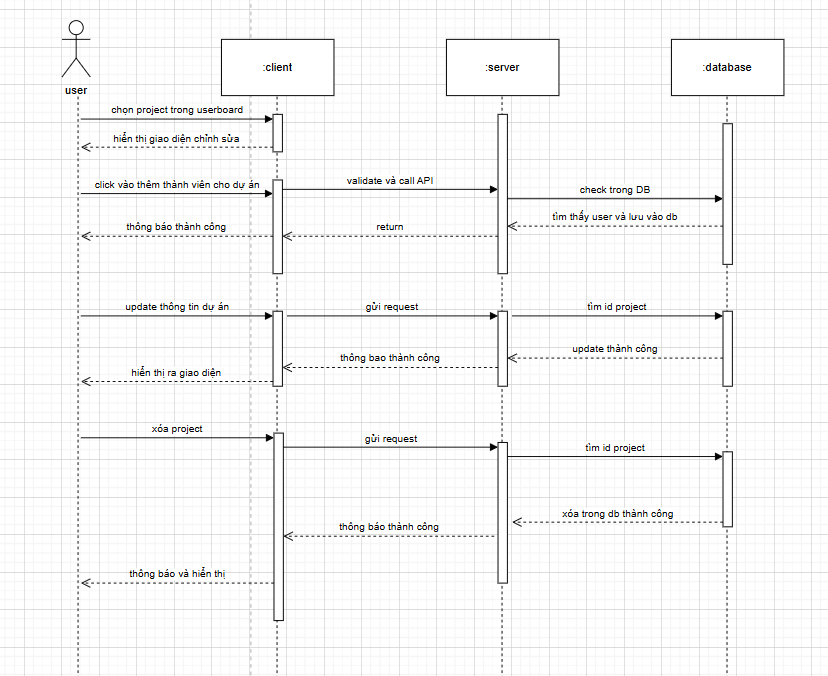
Hình 3.12. Sơ đồ tuần tự đăng xuất

#### **3.5.4.3. Sơ đồ tuần tự tạo dự án**



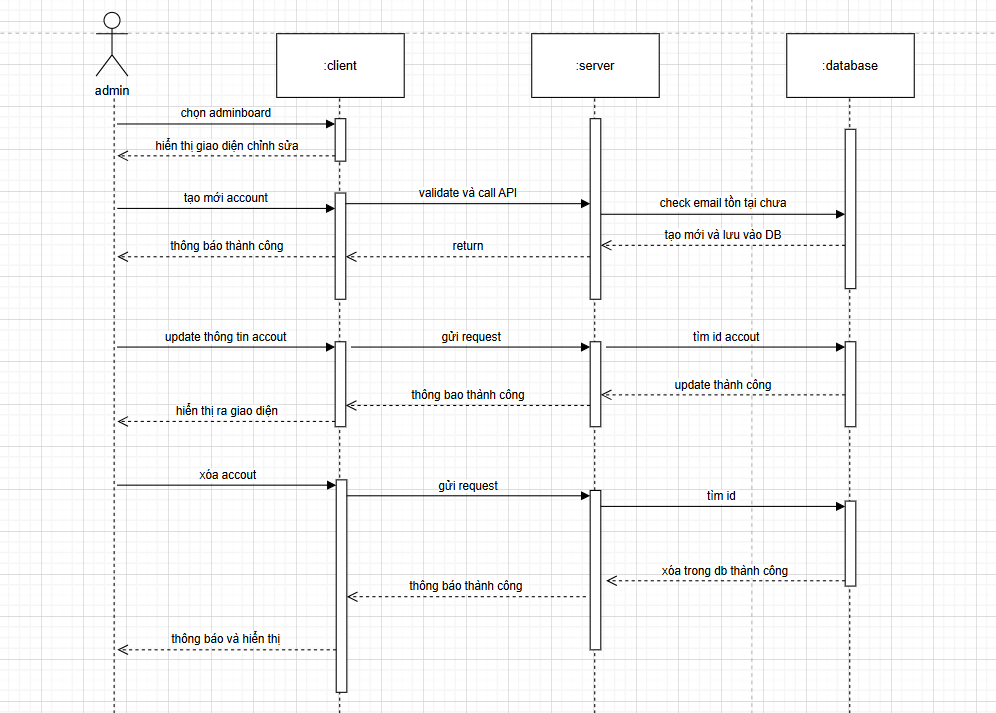
Hình 3. 12. Sơ đồ tuần tự tạo dự án

#### **3.5.4.4. Sơ đồ tuần tự quản lý dự án**



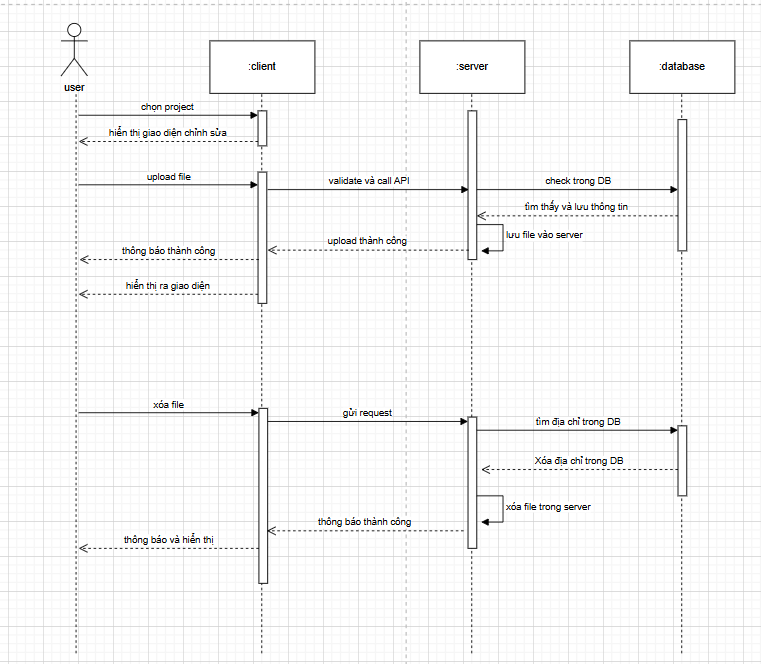
Hình 3. 13. Biểu đồ tuần tự quản lý dự án

#### **3.5.4.5. Sơ đồ tuần tự quản lý account**



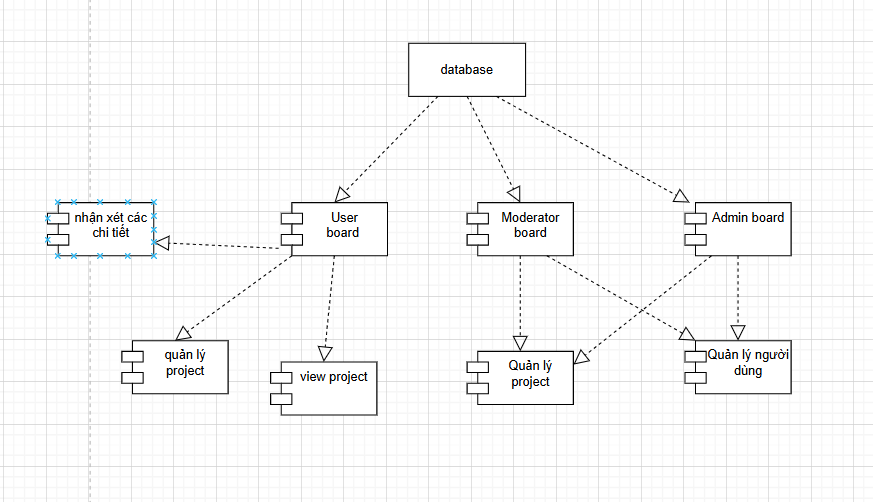
Hình 3. 14. Sơ đồ tuần tự quản lý account

#### **3.5.4.6. Sơ đồ tuần tự upload file**



Hình 3. 15. Sơ đồ tuần tự upload file

### 3.5.5. Biểu đồ thành phần



Hình 3. 28. Biểu đồ thành phần

# THIẾT KẾ HỆ THỐNG

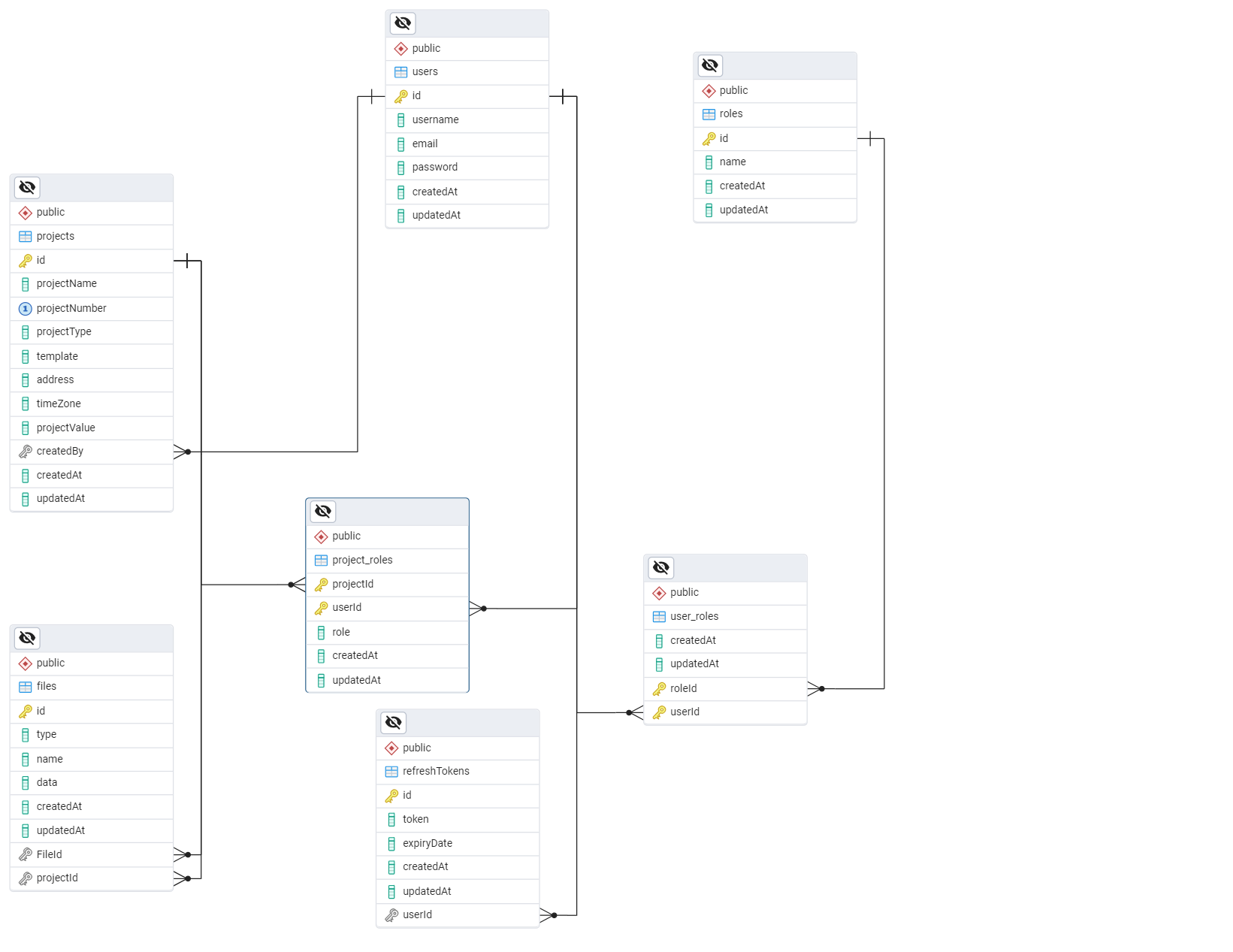
## Biểu đồ lớp chi tiết



Hình 4. 1. Biểu đồ lớp chi tiết của hệ thống

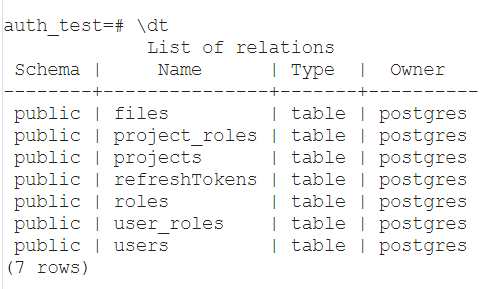
## Thiết kế cơ sở dữ liệu

### Sơ đồ quan hệ CSDL

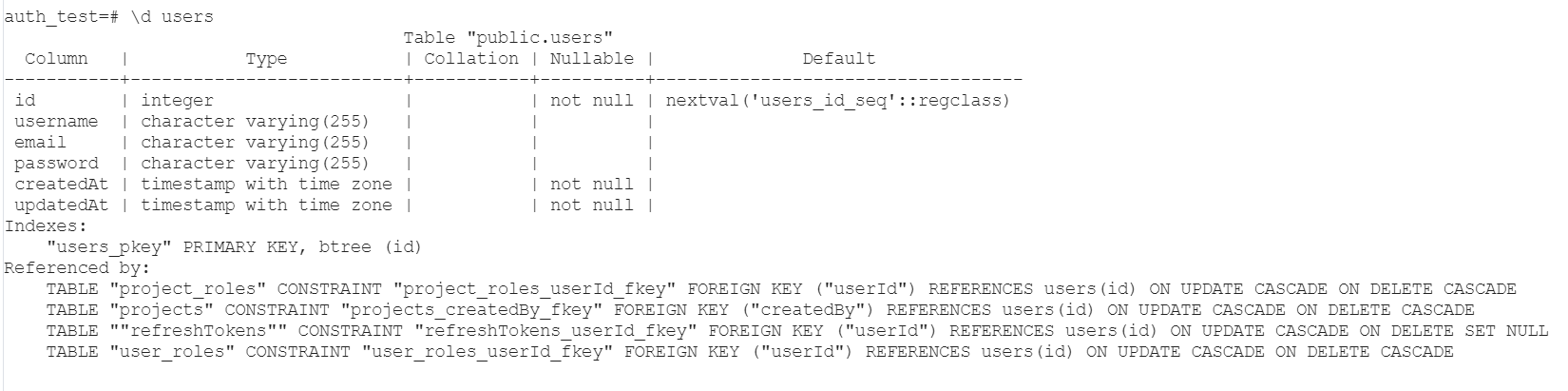


Hình 4. 1. Sơ đồ quan hệ trong CSDL

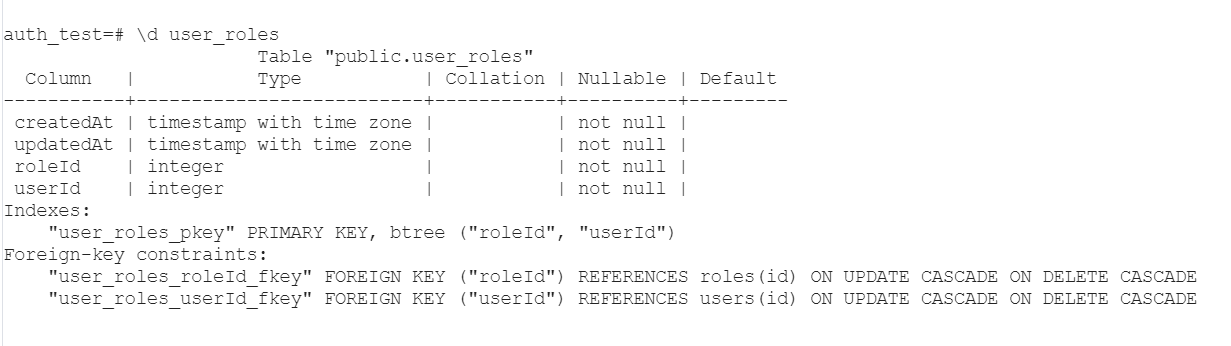
### Thông tin các bảng



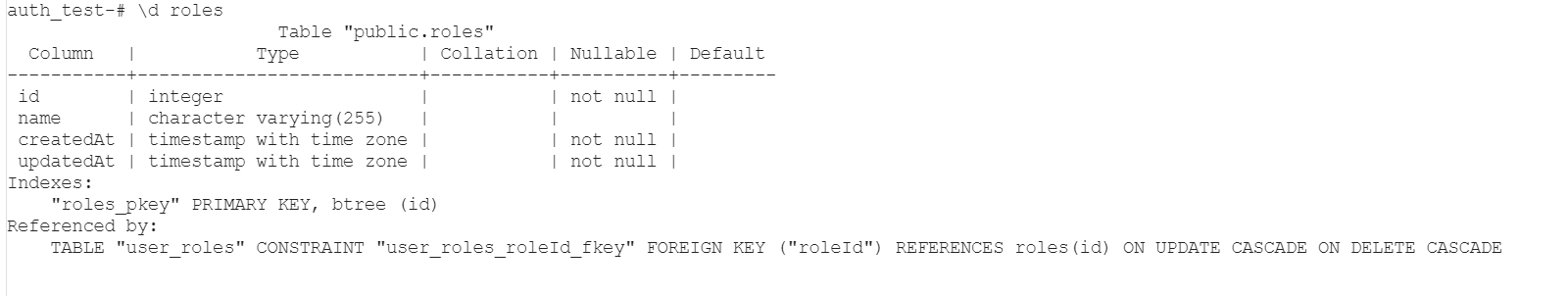
Bảng 4. 2. Danh sách các bảng



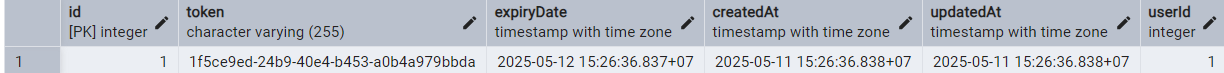
Bảng 4. 3. Bảng users



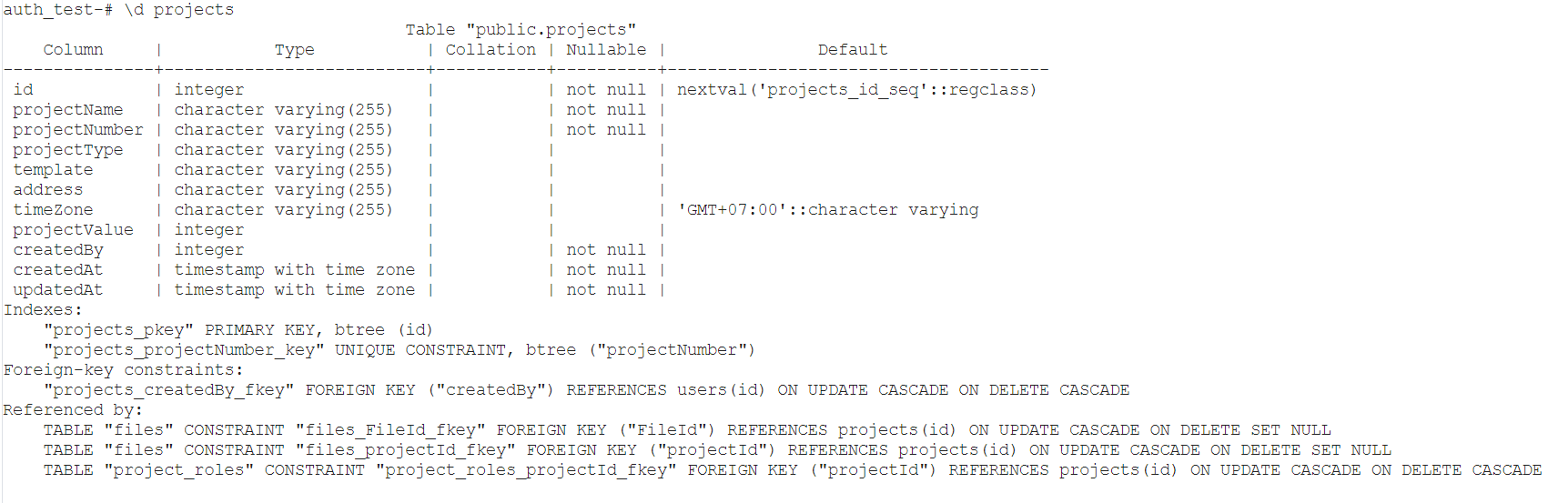
Bảng 4. 4. Bảng user\_roles



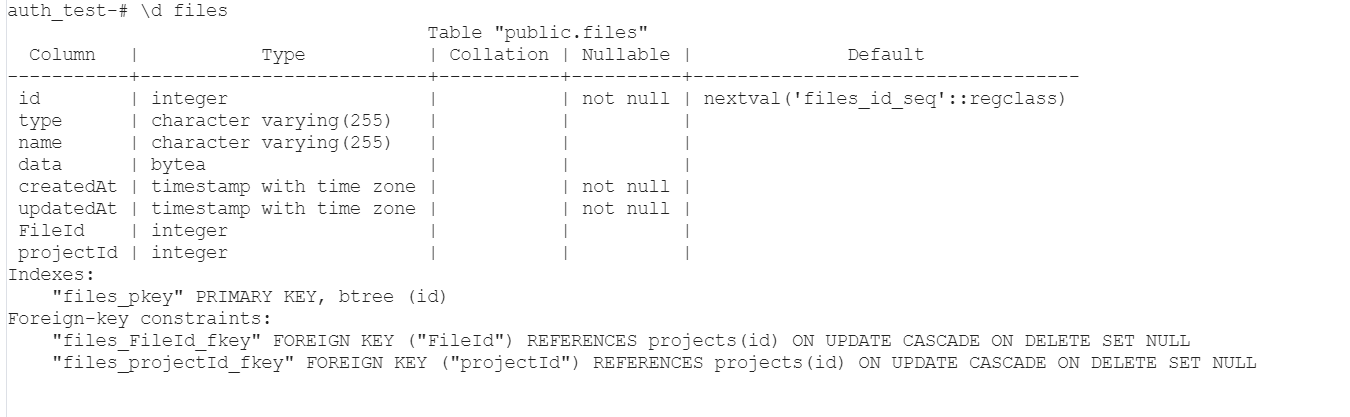
Bảng 4. 5. Bảng roles



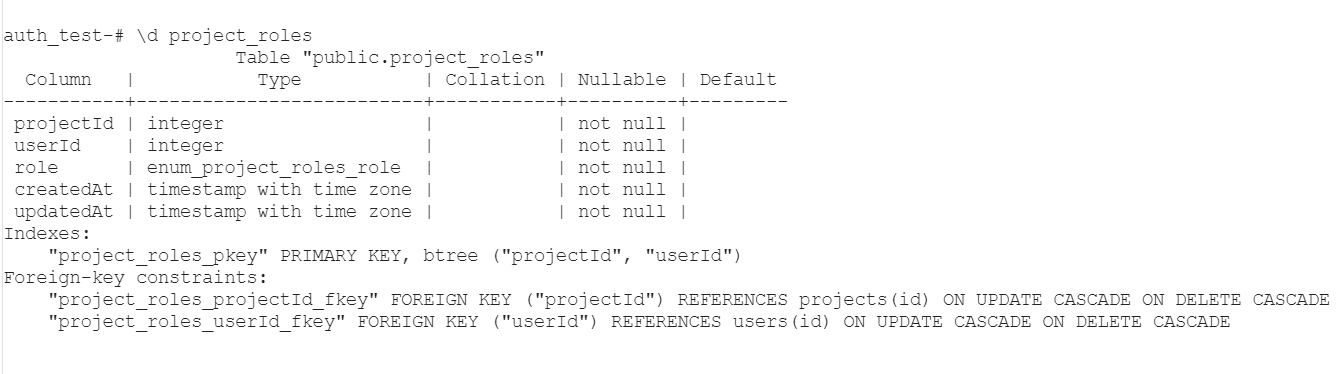
Bảng 4. 6. Bảng refreshTokens



Bảng 4. 6. Bảng projects



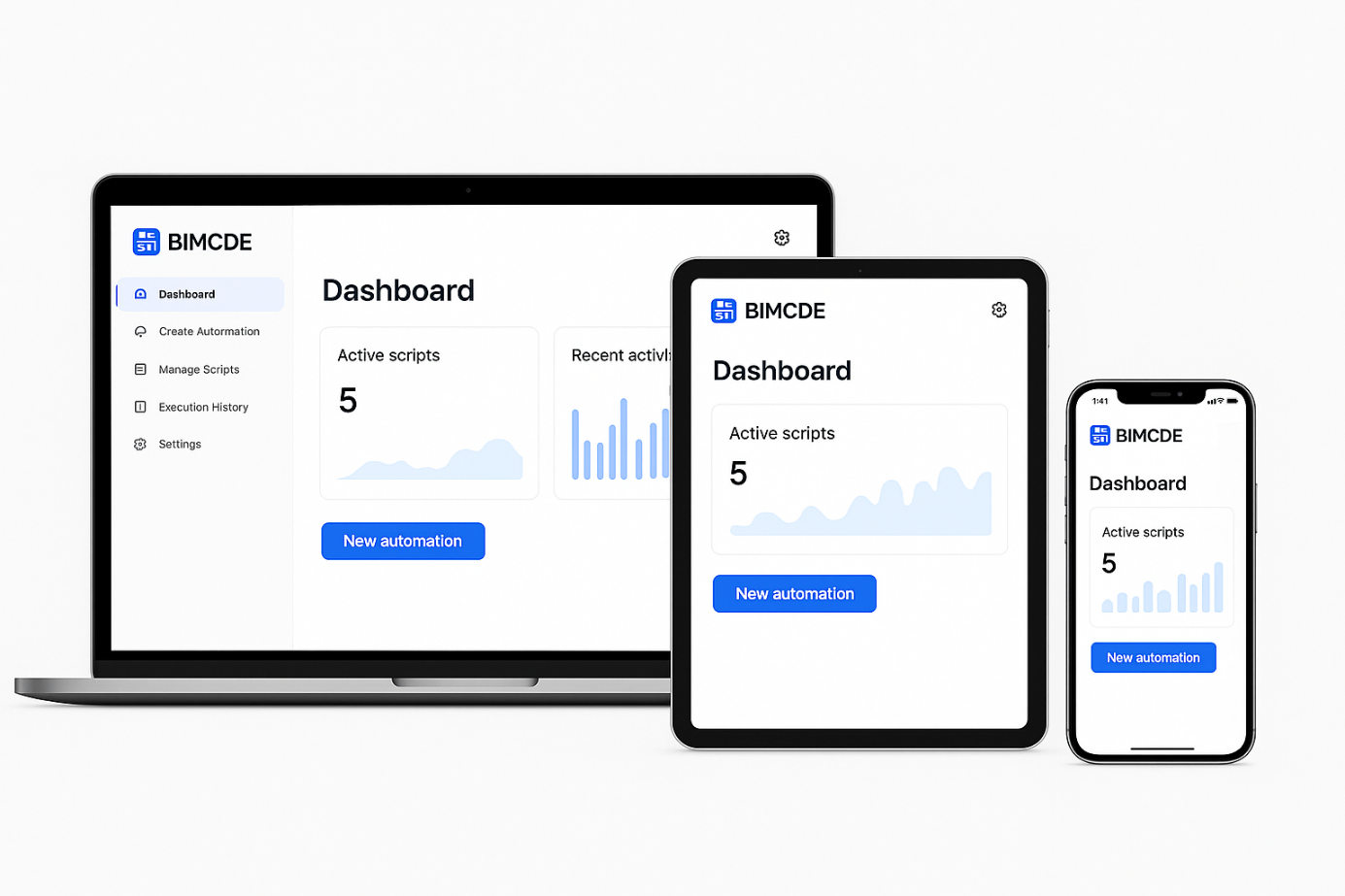
Bảng 4. 7 . Bảng files



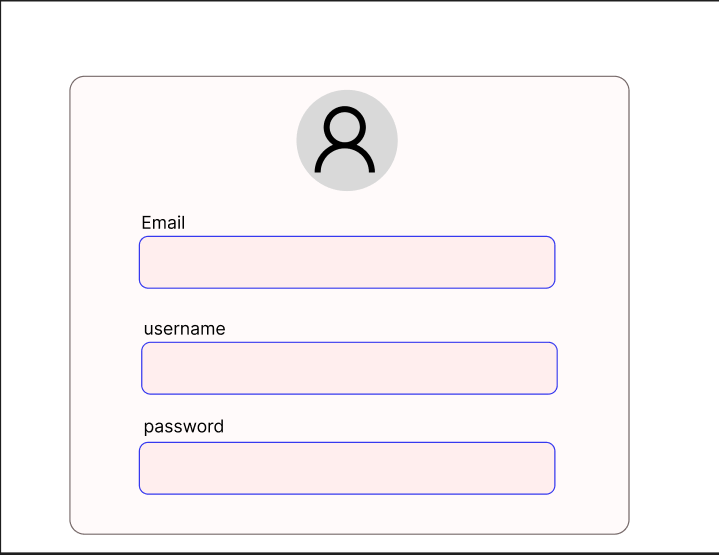
Bảng 4.8 . Bảng project\_roles

### Thiết kế giao diện người dùng

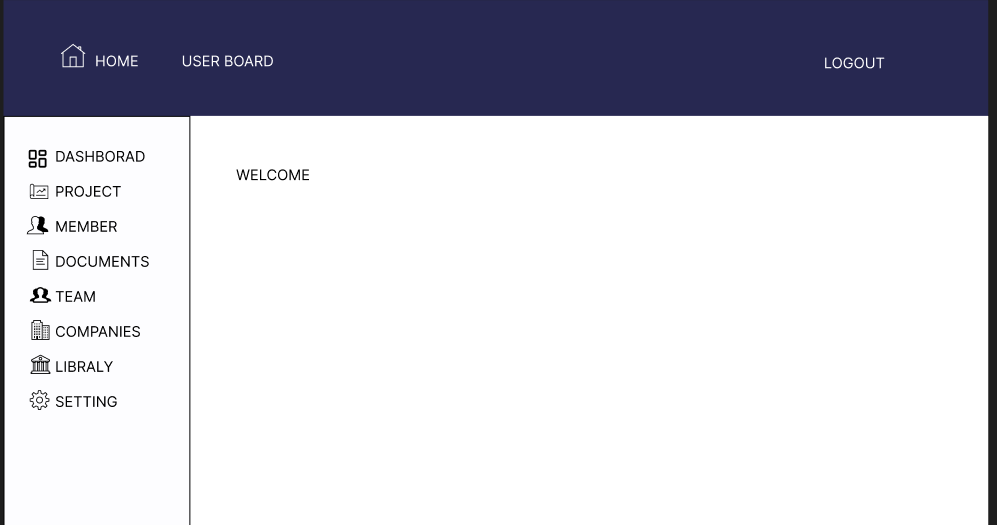
Hình 4‑2 biểu mẫu đầu vào login với figma



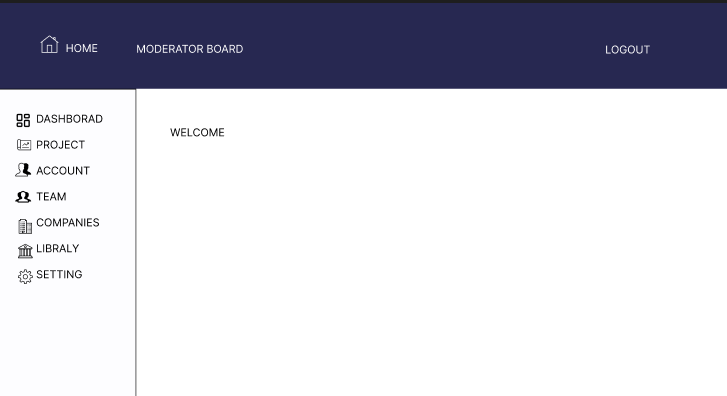
Hình 4‑1 Mockup UI



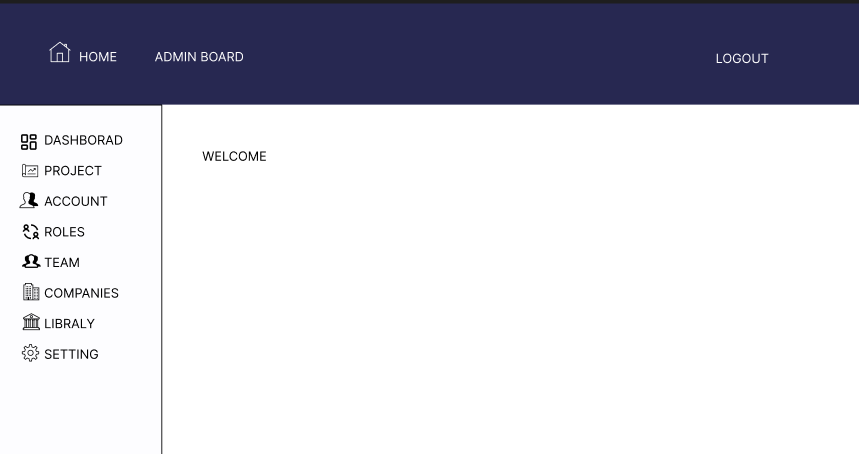
Hình 4‑3 Biểu mẫu đầu vào register với figma



Hình 4‑4 Biểu mẫu đầ vào user board với figma



Hình 4‑5 Biểu mẫu đầu vào Moderator board với figma



Hình 4‑1 Biểu mẫu đầu vào Admin board với figma

## Kiểm thử

### Kịch bản cho các usecase

#### **4.3.1.1. Đăng ký tài khoản:**

* **Tên Use Case:** Đăng ký tài khoản
* **Tác nhân chính:** Người dùng
* **Mô tả:** Người dùng mới truy cập hệ thống và tạo một tài khoản để có thể sử dụng các chức năng.
* **Luồng sự kiện chính:**

1. Người dùng truy cập trang đăng ký.
2. Hệ thống hiển thị biểu mẫu đăng ký (tên người dùng, email, mật khẩu, xác nhận mật khẩu, họ và tên).
3. Người dùng điền đầy đủ thông tin.
4. Người dùng nhấn nút "Đăng ký".
5. Hệ thống kiểm tra tính hợp lệ của dữ liệu (ví dụ: định dạng email, mật khẩu trùng khớp, tên người dùng không trùng).
6. Nếu dữ liệu hợp lệ, hệ thống tạo tài khoản mới và thông báo đăng ký thành công.
7. Hệ thống có thể gửi email xác thực tài khoản (tùy chọn).
8. Người dùng được chuyển hướng đến trang đăng nhập hoặc trang chủ (sau khi xác thực nếu có).

* **Luồng sự kiện thay thế:**
* Dữ liệu không hợp lệ: Hệ thống hiển thị thông báo lỗi cụ thể cho từng trường.
* Tên người dùng hoặc email đã tồn tại: Hệ thống thông báo tên người dùng hoặc email đã được sử dụng

#### **4.3.1.2. Đăng nhập:**

**Tên Use Case:** Đăng nhập

**Tác nhân chính:** Người dùng

**Mô tả:** Người dùng đã có tài khoản truy cập vào hệ thống.

**Luồng sự kiện chính:**

1. Người dùng truy cập trang đăng nhập.
2. Hệ thống hiển thị biểu mẫu đăng nhập (tên người dùng/email, mật khẩu).
3. Người dùng nhập thông tin đăng nhập.
4. Người dùng nhấn nút "Đăng nhập".
5. Hệ thống kiểm tra thông tin đăng nhập với cơ sở dữ liệu.
6. Nếu thông tin chính xác, hệ thống tạo phiên làm việc và chuyển hướng người dùng đến trang chủ hoặc trang trước đó.

**Luồng sự kiện thay thế:**

* Thông tin đăng nhập không chính xác: Hệ thống hiển thị thông báo lỗi "Tên người dùng/email hoặc mật khẩu không đúng".
* Tài khoản chưa được xác thực (nếu có): Hệ thống yêu cầu người dùng xác thực tài khoản.
* Tài khoản bị khóa (nếu có): Hệ thống thông báo tài khoản đã bị khóa và hướng dẫn liên hệ quản trị viên.

#### **4.3.1.3. Xem danh sách dự án:**

**Tên Use Case:** Xem danh sách dự án

**Tác nhân chính:** Người dùng

**Mô tả:** Người dùng xem danh sách các dự án mà họ có quyền truy cập.

**Luồng sự kiện chính:**

1. Người dùng đăng nhập vào hệ thống.
2. Người dùng điều hướng đến trang "Dự án".
3. Hệ thống truy vấn cơ sở dữ liệu và hiển thị danh sách các dự án mà người dùng được phép xem (dựa trên vai trò và quyền hạn).
4. Danh sách hiển thị các thông tin cơ bản của dự án (tên, mô tả ngắn, ngày bắt đầu, người tạo).

**Luồng sự kiện thay thế:**

* Không có dự án nào: Hệ thống hiển thị thông báo "Không có dự án nào".

#### **4.3.1.4. Xem mô hình 3D trực tuyến:**

**Tên Use Case:** Xem mô hình 3D trực tuyến

**Tác nhân chính:** Người dùng

**Mô tả:** Người dùng xem mô hình 3D của một dự án cụ thể.

**Luồng sự kiện chính:**

1. Người dùng đăng nhập vào hệ thống.
2. Người dùng chọn một dự án từ danh sách.
3. Người dùng chọn một tệp mô hình 3D trong dự án.
4. Hệ thống tải và hiển thị mô hình 3D bằng trình xem trực tuyến.
5. Người dùng có thể sử dụng các công cụ của trình xem để điều hướng, thu phóng, xoay mô hình, xem thuộc tính đối tượng (nếu có quyền).

**Luồng sự kiện thay thế:**

* Không có tệp mô hình 3D trong dự án: Hệ thống hiển thị thông báo.
* Lỗi tải mô hình: Hệ thống hiển thị thông báo lỗi.
* Người dùng không có quyền xem mô hình: Hệ thống hiển thị thông báo lỗi truy cập.

#### **4.3.1.5. Thêm chú thích trên mô hình:**

**Tên Use Case:** Thêm chú thích trên mô hình

**Tác nhân chính:** Người dùng

**Mô tả:** Người dùng thêm ghi chú văn bản hoặc hình vẽ trực tiếp lên mô hình 3D để chia sẻ thông tin.

**Luồng sự kiện chính:**

1. Người dùng đã xem mô hình 3D trực tuyến.
2. Người dùng chọn công cụ chú thích.
3. Người dùng chỉ định vị trí trên mô hình.
4. Người dùng nhập nội dung chú thích (văn bản) hoặc vẽ.
5. Người dùng lưu chú thích.
6. Hệ thống lưu chú thích và hiển thị nó trên mô hình cho những người dùng khác có quyền xem.

**Luồng sự kiện thay thế:**

* Người dùng không có quyền thêm chú thích: Hệ thống vô hiệu hóa chức năng hoặc hiển thị thông báo lỗi.

### Kịch Bản Use Case cho Quản Trị Viên:

#### **4.3.2.1. Quản lý người dùng:**

* **Tên Use Case:** Quản lý người dùng (Xem, Thêm, Sửa, Xóa)
* **Tác nhân chính:** Quản trị viên
* **Mô tả:** Quản trị viên thực hiện các thao tác quản lý tài khoản người dùng.
* **Luồng sự kiện chính:**
  1. Quản trị viên đăng nhập vào hệ thống.
  2. Quản trị viên điều hướng đến trang "Quản lý người dùng".
  3. **Xem:** Hệ thống hiển thị danh sách tất cả người dùng và thông tin cơ bản của họ.
  4. **Thêm:** Quản trị viên chọn "Thêm người dùng", điền thông tin người dùng mới (tên người dùng, email, mật khẩu, vai trò) và lưu. Hệ thống kiểm tra tính hợp lệ và tạo tài khoản mới.
  5. **Sửa:** Quản trị viên chọn một người dùng từ danh sách, chỉnh sửa thông tin (vai trò, trạng thái, v.v.) và lưu. Hệ thống cập nhật thông tin người dùng.
  6. **Xóa:** Quản trị viên chọn một người dùng từ danh sách và chọn "Xóa". Hệ thống yêu cầu xác nhận và sau đó xóa tài khoản người dùng.
* **Luồng sự kiện thay thế:**
  1. Dữ liệu không hợp lệ khi thêm/sửa: Hệ thống hiển thị thông báo lỗi.
  2. Lỗi cơ sở dữ liệu: Hệ thống hiển thị thông báo lỗi.

#### **4.3.2.2. Quản lý dự án:**

* **Tên Use Case:** Quản lý dự án (Xem, Thêm, Sửa, Xóa)
* **Tác nhân chính:** Quản trị viên
* **Mô tả:** Quản trị viên thực hiện các thao tác quản lý dự án.
* **Luồng sự kiện chính:** Tương tự như quản lý người dùng, với các thao tác xem, thêm, sửa, xóa thông tin dự án (tên, mô tả, ngày bắt đầu, ngày kết thúc, người tạo).

#### **4.3.3.3. Quản lý quyền truy cập:**

* **Tên Use Case:** Phân quyền theo dự án/vai trò
* **Tác nhân chính:** Quản trị viên
* **Mô tả:** Quản trị viên thiết lập quyền truy cập của người dùng đối với các dự án và các chức năng khác của hệ thống.
* **Luồng sự kiện chính:**
  1. Quản trị viên điều hướng đến trang "Quản lý quyền truy cập".
  2. **Theo dự án:** Quản trị viên chọn một dự án và chỉ định quyền (xem, tải xuống, chỉnh sửa, quản lý, v.v.) cho từng người dùng hoặc nhóm người dùng trong dự án đó.
  3. **Theo vai trò:** Quản trị viên định nghĩa các vai trò (ví dụ: "Người xem", "Biên tập viên") và gán các quyền hạn chung cho từng vai trò. Sau đó, gán vai trò cho người dùng.
  4. Hệ thống lưu các thiết lập quyền.
* **Luồng sự kiện thay thế:**
  1. Lỗi nhập dữ liệu quyền: Hệ thống hiển thị thông báo lỗi.

# KẾT QUẢ THỰC NGHIỆM

## Môi trường cài đặt

### Phần Cứng (Hardware):

* CPU (Bộ xử lý): Số lượng Core: Tối thiểu 16 cores (32 threads) trở lên. Các tác vụ BIM, đặc biệt là xử lý mô hình 3D và các dịch vụ backend, có thể tận dụng lợi thế của nhiều core. Tốc độ xung nhịp: Ưu tiên tốc độ xung nhịp cao (ví dụ: 3.0 GHz trở lên) cho các tác vụ đơn luồng và độ trễ thấp. (Intel Xeon Gold hoặc AMD EPYC series).
* RAM (Bộ nhớ đệm): Dung lượng: Tối thiểu 64 GB ECC RAM, khuyến nghị 128 GB hoặc cao hơn tùy thuộc vào kích thước mô hình và số lượng người dùng. ECC RAM (Error-Correcting Code RAM) quan trọng cho tính ổn định và độ tin cậy của hệ thống server.
* Tốc độ: Chọn RAM có tốc độ phù hợp với CPU để đảm bảo hiệu suất tốt nhất.
* Lưu trữ (Storage): Ổ cứng hệ điều hành và ứng dụng: Ít nhất 2 x SSD NVMe dung lượng 500GB trở lên, cấu hình RAID 1 (mirroring) để đảm bảo tính dự phòng và chống lỗi.
* Lưu trữ dữ liệu BIM: Tùy chọn 1: Một hoặc nhiều SSD NVMe dung lượng lớn (ví dụ: vài TB trở lên) cấu hình RAID 5 hoặc RAID 6 để cân bằng giữa hiệu suất, dung lượng và khả năng chịu lỗi. Tùy chọn 2: Sử dụng hệ thống lưu trữ mạng (NAS hoặc SAN) với dung lượng và hiệu suất phù hợp, có tính năng sao lưu và bảo vệ dữ liệu. Lưu trữ tạm và cache: Một SSD NVMe riêng biệt dung lượng vừa phải có thể cải thiện hiệu suất cho các tác vụ cache của hệ thống và ứng dụng.
* Card Mạng (Network Interface Card - NIC): Tối thiểu 2 x 1 Gbps Ethernet, khuyến nghị 10 Gbps nếu có nhu cầu truyền tải dữ liệu lớn thường xuyên và hệ thống mạng hỗ trợ. Hỗ trợ Teaming hoặc Link Aggregation để tăng băng thông và tính dự phòng.
* Card Đồ Họa (GPU - Tùy chọn): Nếu hệ thống có các dịch vụ render 3D nặng hoặc hỗ trợ các tính năng visualization phức tạp trực tiếp trên server, một card đồ họa chuyên dụng cấp độ workstation (ví dụ: NVIDIA Quadro hoặc AMD Radeon Pro) có thể cần thiết. Tuy nhiên, đối với việc xem mô hình thông thường trên trình duyệt (như Speckle Viewer), yêu cầu GPU trên server có thể không cao.
* Nguồn Điện (Power Supply Unit - PSU): Công suất phù hợp với tổng công suất tiêu thụ của các thành phần, có chứng nhận 80+ Gold hoặc cao hơn để đảm bảo hiệu suất và độ ổn định. Nên có dự phòng công suất để tránh quá tải.
* Hệ thống làm mát: Hệ thống tản nhiệt hiệu quả (ví dụ: tản nhiệt khí hoặc tản nhiệt chất lỏng) để đảm bảo các thành phần hoạt động ổn định dưới tải cao.

### Phần Mềm (Software):

* Hệ điều hành: Linux (CentOS, Ubuntu Server) thường được ưu tiên cho sự ổn định, bảo mật và hiệu suất của server. Windows Server cũng là một lựa chọn phổ biến.
* Hệ thống quản lý cơ sở dữ liệu (DBMS): PostgreSQL thường là lựa chọn tốt cho các ứng dụng BIM do tính ổn định, tuân thủ ACID và hỗ trợ các kiểu dữ liệu phức tạp.
* Web Server: Nginx hoặc Apache là các web server phổ biến và ổn định.
* Các ứng dụng và dịch vụ BIM: Phần mềm server của nền tảng BIM đang sử dụng (Speckle Server). Các dịch vụ phụ trợ khác (dịch vụ webhook, dịch vụ tạo preview).
* Phần mềm giám sát và quản lý server: Prometheus, Grafana, Zabbix để theo dõi hiệu suất và trạng thái của server.
* Phần mềm bảo mật: Firewall, Intrusion Detection/Prevention Systems (IDS/IPS).
* Phần mềm sao lưu và phục hồi: Các công cụ để thực hiện sao lưu định kỳ và phục hồi dữ liệu khi cần thiết.

### Yêu cầu Mạng:

* Băng thông internet: Đủ lớn để đáp ứng lưu lượng truy cập đồng thời và việc truyền tải dữ liệu mô hình (upload/download).
* Kết nối mạng nội bộ: Mạng LAN ổn định và tốc độ cao để đảm bảo giao tiếp nhanh chóng giữa server và người dùng trong mạng nội bộ.

## Môi trường phát triển (Development Environment):

### Cấu hình máy tính cá nhân phát triển

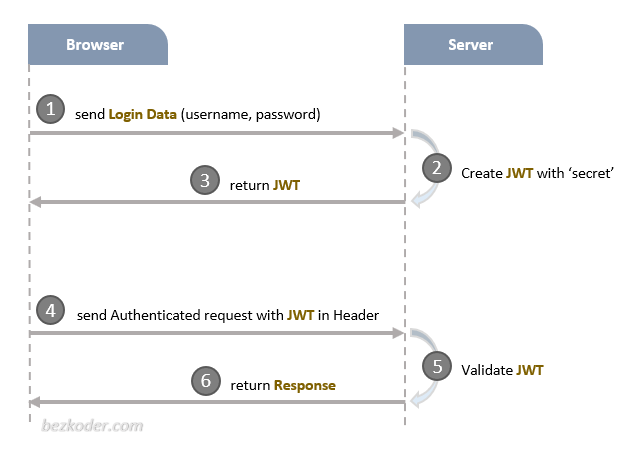
* Máy tính cá nhân: Dell vostro 5510
* CPU: Intel Core i5 11th Gen, 4 nhân (8 luồng)
* RAM: 8GB
* SSD (Solid State Drive)
* IDE/Trình soạn thảo mã nguồn: Visual Studio Code,
* Trình duyệt kiểm thử: Chrome: Phiên bản mới nhất
* Thông số mạng: Băng thông: Kết nối Internet với tốc độ tối thiểu 50 Mbps để đảm bảo truyền tải dữ liệu mượt mà. Bảo mật: Cài đặt tường lửa (Firewall) và chứng chỉ SSL/TLS để bảo mật thông tin người dùng
* Phần mềm phát triển: Node.js 20, React.js 17, postgreSQL 16.

Các cấu hình trên được chọn dựa trên yêu cầu vận hành của hệ thống, đảm bảo website có thể chạy ổn định, xử lý đồng thời nhiều yêu cầu đặt bàn, đồng thời hỗ trợ phát triển và mở rộng các tính năng trong tương lai.

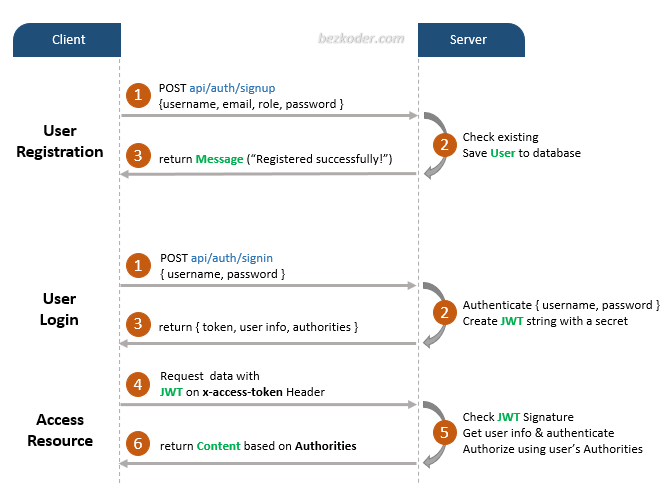
### Cấu trúc thư mục backend

Công nghệ xây dựng các chức năng của backend là

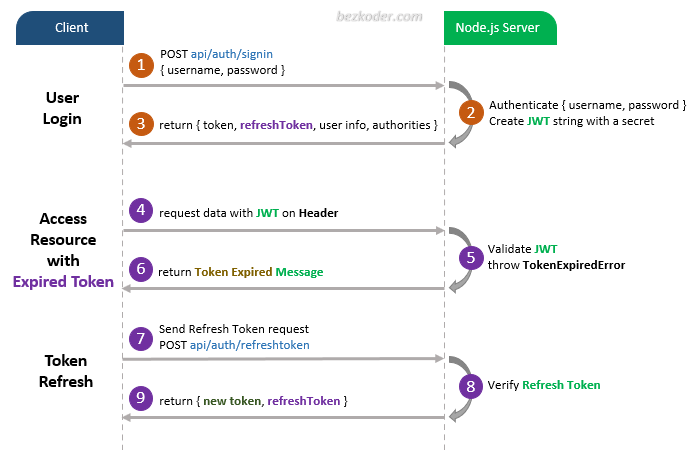
* NodeJS:
* Express.js
* PostgreSQL
* Npm



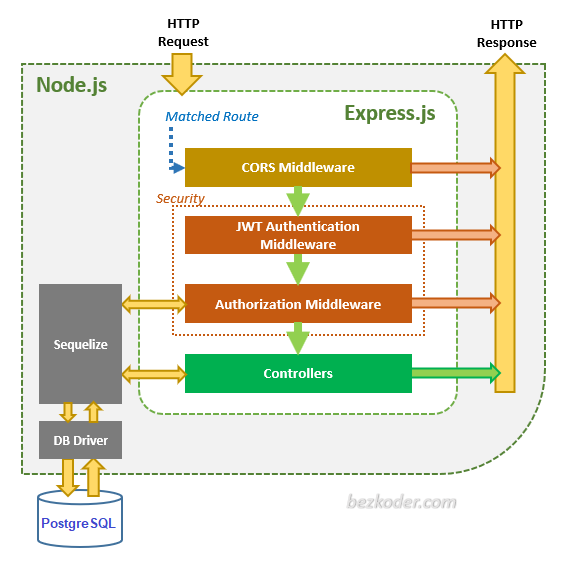
Hình 5‑1 Token Based Authentication



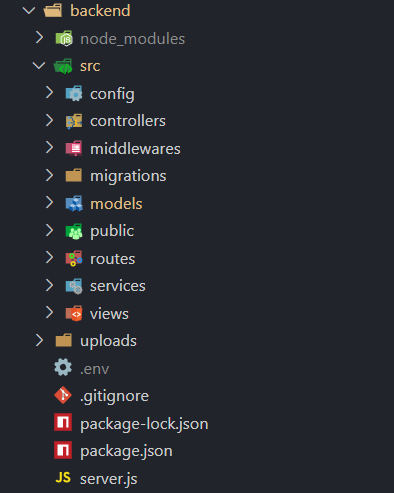
Hình 5‑2 Flow for Signup & Login with JWT Authentication



*Hình 5‑3 Flow for Login with JWT refreshToken*



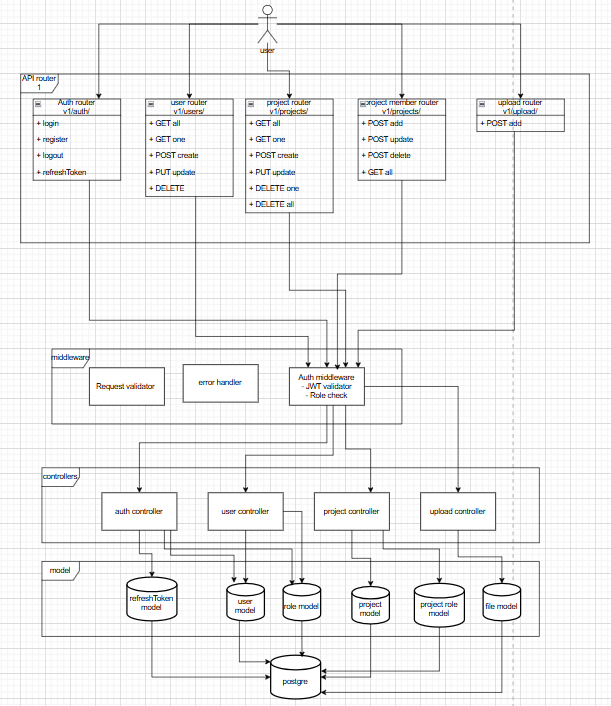
*Hình 5‑4 Kiến trúc NodeJS và ExpressJS với Authentication & Authorization*



Hình 5‑5 Cấu trúc thư mục backend

***Chức năng chính của các Module***

* Node\_modules: chứa tất cả các thư viện/phụ thuộc (dependencies) mà bạn khai báo trong package.json.
* Config: chứa các file cấu hình: kết nối database, secret key, cài đặt app
* Controllers: Chứa các hàm xử lý logic cho API, nhận request từ routes và xử lý dữ liệu (thường gọi service hoặc truy xuất model).
* Models: Định nghĩa các bảng, quan hệ (với Sequelize).
* Routes: Định nghĩa các endpoint và ánh xạ chúng với controller tương ứng.
* Middlewares: Chứa các chức năng trung gian như xác thực JWT, phân quyền, validate input,...
* Services: Xử lý logic nghiệp vụ phức tạp, tách khỏi controller để dễ kiểm thử và bảo trì.
* Views: kiểm thử giao diện với các chức năng
* Server.js: khởi tạo express app, đăng ký middleware, routes và lăng nghe server trên cổng chính cấu hình
* .env: lưu các biến môi trường quan trọng để cấu hình ứng dụng



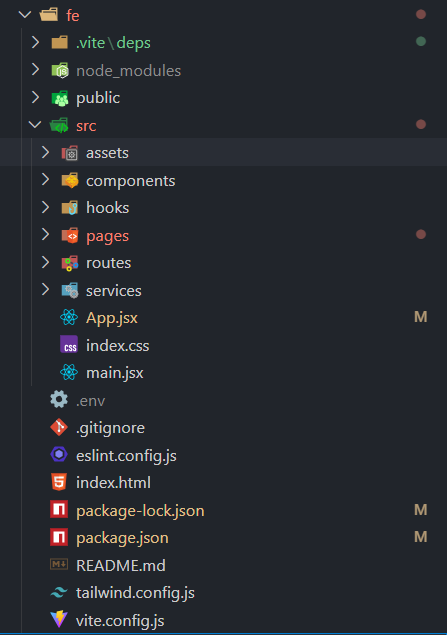
***Các API endpoint***

|  |  |  |
| --- | --- | --- |
| **Phương thức HTTP** | **Endpoint** | **Mô tả** |
| POST | /v1/auth/register | Đăng ký người dùng mới |
| POST | /v1/auth/login | Đăng nhập |
| POST | /v1/auth/logout | Đăng xuất |
| **POST** | /v1/auth/refreshTokens | Làm mới token xác thực |
| Post | /v1/users/create-user | Tạo tài khoản người dùng mới |
| Get | /v1/users/ | Lấy toàn bộ danh sách các tài khoản |
| Get | /v1/users/:id | Lấy thông tin tài theo id |
| Put | /v1/users/:id | Cập nhật tài khoản |
| Delete | /v1/users/:id | Xóa một tài khoản |
| Get | /v1/projects/ | Hiển thị toàn bộ danh sách dự án |
| Post | /v1/projects/create | Tạo dự án |
| Get | /v1/projects/:id | Tìm một dự án |
| Put | /v1/projects/:id | Cập nhật thông tin dự án |
| Delete | /v1/projects/:id | Xóa một dự án |
| Delete | /v1/projects/ | Xóa toàn bộ dự án |
| Post | /v1/projects/:id/add-member | thêm một thành viên vào dự án |
| Post | /v1/projects/update-member | Cập nhật lại các thành viên trong dự án |
| Post | /v1/projects/:id/remove-member | Xóa một thành viên trong dự án |
| Get | /v1/projects/getAllMember | Lấy tất cả các thành viên trong dự án |
| Post | /v1/uploads/ | Upload file |
| Delete | /v1/uploads/ | Xóa file |

**Mã Lỗi HTTP**

* 400: Bad Request - Yêu cầu không hợp lệ
* 401: Unauthorized - Chưa đăng nhập
* 403: Forbidden - Không có quyền truy cập
* 500: Internal Server Error - Lỗi phân loại

### react-express-authentication-jwt-example-react-componentsCấu trúc thư mục Frontend

****

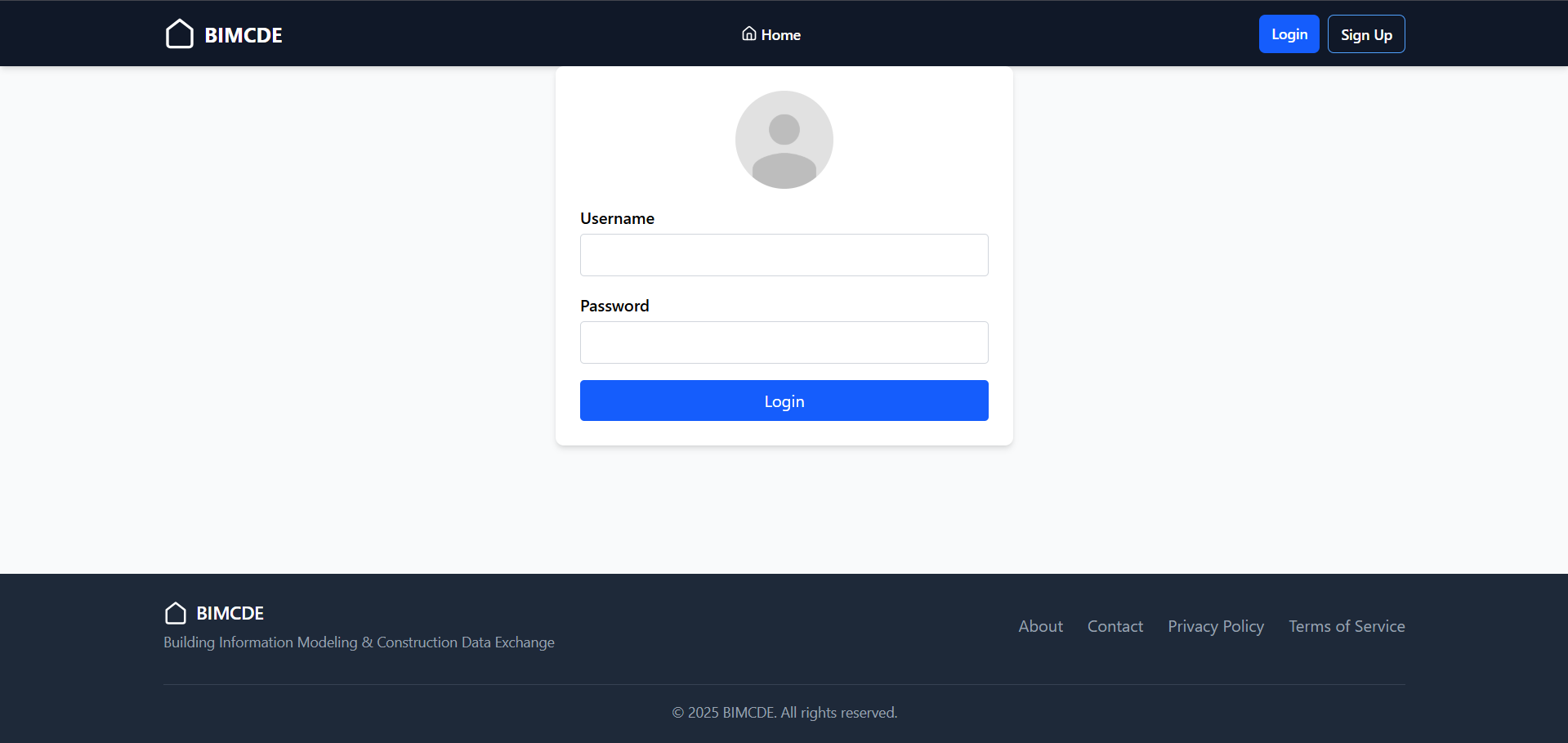
Hình 5‑6 Cấu trúc thư mục frontend

***Cấu trúc thư mục frontend :***

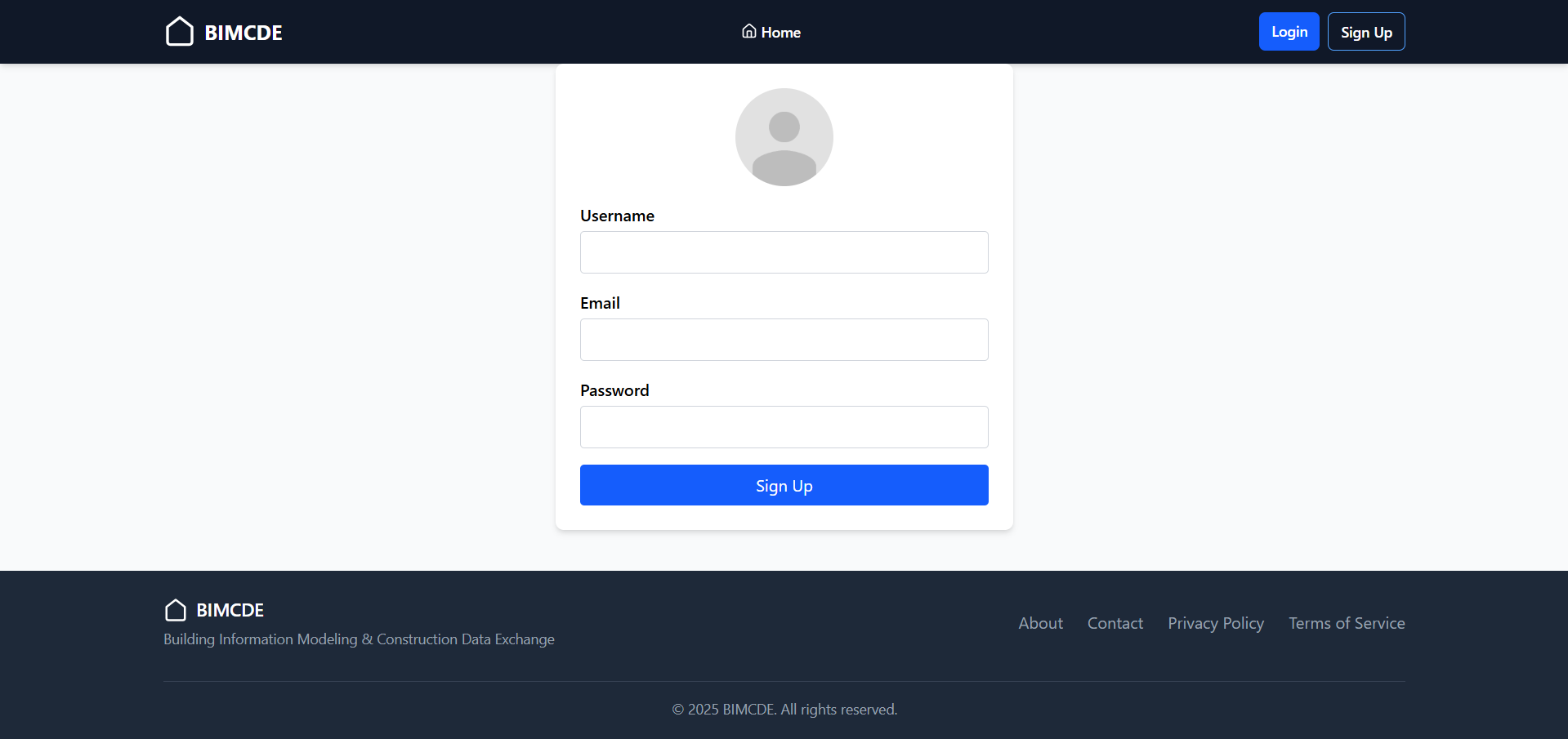
* Public : chứa các file tĩnh, favicon, ảnh không xử lý
* Src : chứa tất cả các source code React
* Assets : chứa tất cả các tài nguyên tĩnh như ảnh, fronts,…
* Components : chứa các thành phần UI nhỏ, dùng lại nhiều nơi
* Pages : các trang lớn, Login, Reagister, Home,…
* Services : chứa các file gọi API
* Routes : khai báo và định tuyến các trang với react-router-dom
* App.jsx : thành phần gốc của app
* Index.jsx : điểm khởi chạy app và reader vào DOM

## Kết quả thực nghiệm

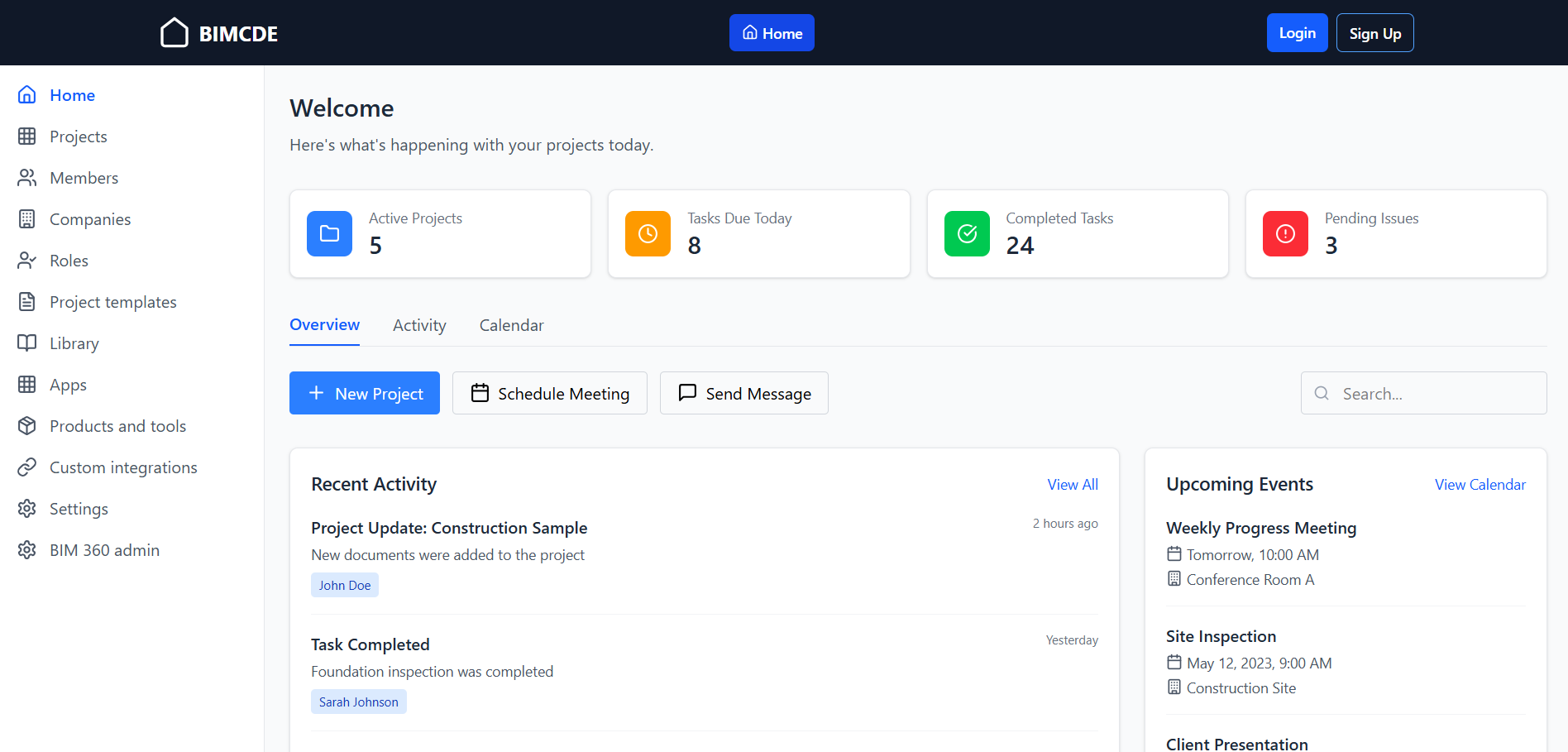
### Các trang lớn của dự án



Hình 5‑7 Chức năng đăng nhập



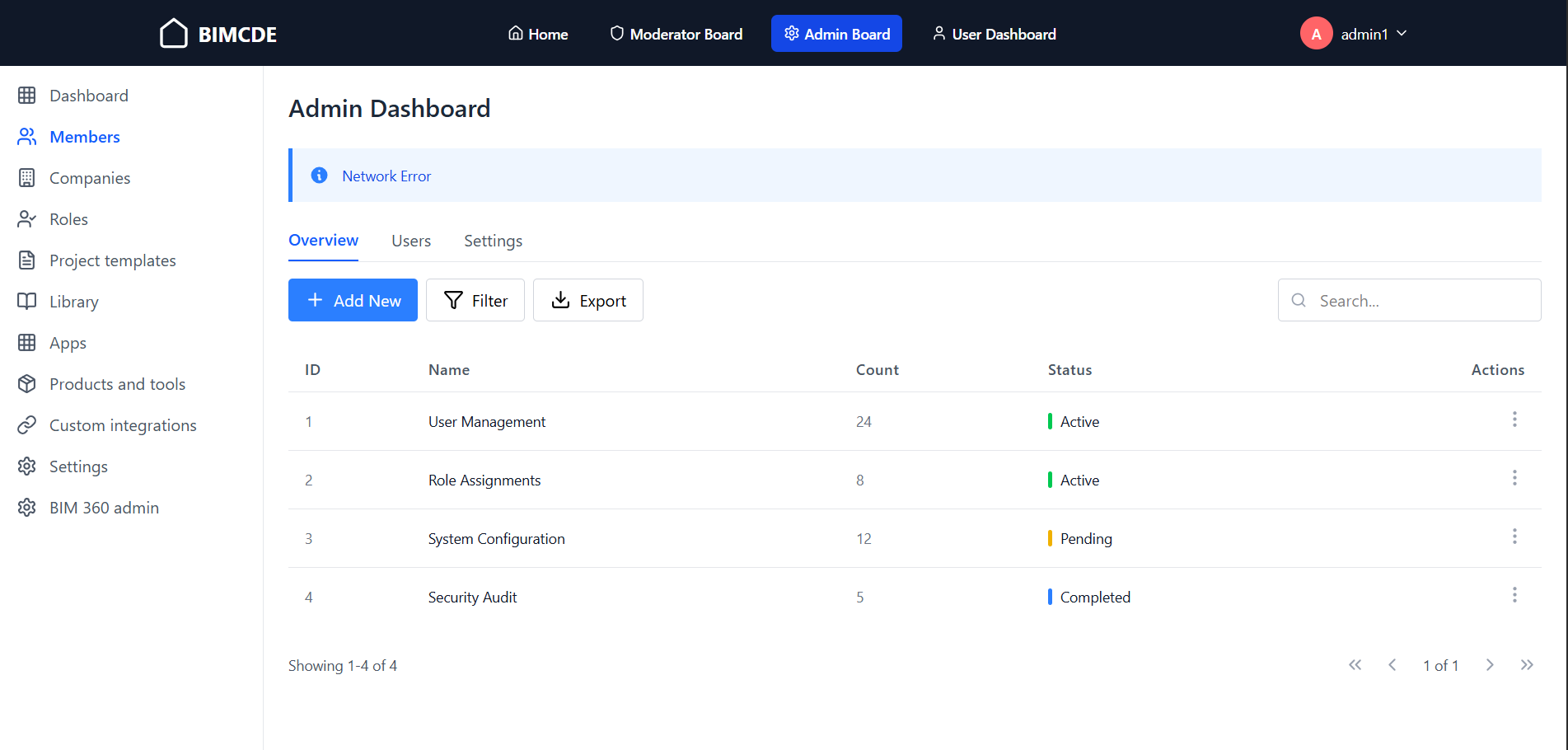
Hình 5‑8 Chức năng đăng ký



Hình 5‑9 Home page

### Các chức năng của admin

Chức năng quản lý tài khoản người dùng vào phân quyền role cho mỗi tài khoan



# KẾT LUẬN

Đề tài "*Xây dựng phân hệ quản trị cho phần mềm quản lý dữ liệu BIM*" đã tập trung vào việc giải quyết các thách thức trong quản lý và chia sẻ dữ liệu mô hình 3D trong môi trường BIM. Thông qua việc khảo sát bài toán, đã làm rõ những khó khăn mà người dùng và nhà quản lý dự án thường gặp phải khi làm việc với dữ liệu BIM phân tán và thiếu một nền tảng quản lý tập trung.

Phân hệ quản trị được đề xuất mang đến một giải pháp toàn diện, cung cấp các chức năng cốt lõi như quản lý người dùng và vai trò, quản lý dự án, tải lên, xem và quản lý phiên bản mô hình 3D trực tuyến, cũng như quản lý quyền truy cập và hỗ trợ cộng tác thông qua chú thích và bình luận. Các yêu cầu phi chức năng về hiệu năng, bảo mật, độ tin cậy, giao diện người dùng và khả năng bảo trì cũng đã được xem xét để đảm bảo hệ thống hoạt động ổn định và mang lại trải nghiệm tốt cho người dùng.

Việc xây dựng một phân hệ quản trị dữ liệu BIMCDE hiệu quả sẽ mang lại nhiều lợi ích thiết thực, bao gồm cải thiện khả năng tương tác dữ liệu giữa các bên liên quan, tăng cường hiệu quả cộng tác, quản lý dữ liệu tập trung và dễ dàng truy cập, nâng cao khả năng tiếp cận thông tin mô hình cho cả người dùng có và không có phần mềm BIM chuyên dụng, và hỗ trợ quá trình ra quyết định dựa trên thông tin trực quan.

# HƯỚNG PHÁT TRIỂN ĐỀ TÀI TRONG TƯƠNG LAI

Để phân hệ quản trị dữ liệu BIMCDE này tiếp tục phát triển và đáp ứng tốt hơn nữa nhu cầu ngày càng cao của ngành AEC, có một số hướng phát triển tiềm năng có thể được xem xét trong tương lai:

1. **Tích hợp sâu hơn với các nền tảng và công cụ BIM khác:** Mở rộng khả năng kết nối với nhiều phần mềm BIM hơn thông qua các API hoặc connector, bao gồm cả khả năng chuyển đổi dữ liệu giữa các định dạng khác nhau một cách thông minh và tự động.
2. **Phát triển các công cụ phân tích và báo cáo dữ liệu BIM:** Bổ sung các tính năng phân tích mô hình (ví dụ: kiểm tra xung đột nâng cao, phân tích hiệu suất năng lượng sơ bộ) và tạo ra các báo cáo trực quan về dữ liệu dự án, tiến độ và các thông số kỹ thuật.
3. **Tăng cường khả năng cộng tác thời gian thực:** Nghiên cứu và triển khai các tính năng cho phép nhiều người dùng cùng nhau tương tác và chỉnh sửa mô hình hoặc thông tin dự án trong thời gian thực.
4. **Ứng dụng công nghệ tiên tiến:** Khám phá việc tích hợp các công nghệ mới như Machine Learning (ví dụ: để dự đoán các vấn đề tiềm ẩn trong mô hình), Augmented Reality (AR) hoặc Virtual Reality (VR) để cải thiện trải nghiệm xem và tương tác với mô hình.
5. **Mở rộng khả năng quản lý thông tin phi hình học:** Bên cạnh dữ liệu 3D, tích hợp khả năng quản lý các loại thông tin khác liên quan đến dự án như tài liệu 2D, tiến độ, chi phí và thông tin liên lạc.
6. **Tối ưu hóa hiệu suất và khả năng mở rộng:** Tiếp tục cải thiện hiệu suất của hệ thống, đặc biệt là đối với các mô hình lớn và số lượng người dùng đồng thời cao, đồng thời đảm bảo khả năng mở rộng linh hoạt để đáp ứng sự tăng trưởng trong tương lai.
7. **Nâng cao tính bảo mật và quản lý quyền truy cập chi tiết:** Triển khai các biện pháp bảo mật tiên tiến hơn và cung cấp khả năng quản lý quyền truy cập ở mức độ chi tiết hơn cho từng đối tượng hoặc thuộc tính trong mô hình.
8. **Phát triển thêm ứng dụng desktop:** giup người dùng tăng hiệu quả
9. **Phát triển API mở và SDK:** Cung cấp API mạnh mẽ và bộ công cụ phát triển phần mềm (SDK) để cho phép các nhà phát triển bên thứ ba tích hợp hệ thống BIMCDE với các ứng dụng và quy trình làm việc khác của họ.

Việc tiếp tục nghiên cứu và phát triển theo các hướng này sẽ giúp phân hệ quản trị dữ liệu BIMCDE trở thành một công cụ ngày càng mạnh mẽ và hữu ích, đóng góp vào sự hiệu quả và đổi mới trong ngành xây dựng.

# TÀI LIỆU THAM KHẢO

[1] **Lê Văn Chương** (2017). *HTML & CSS: Thiết kế và xây dựng website*. Nhà xuất bản Công nghệ.

[2] **TS. Lê Văn Phùng** (2010). Phân tích thiết kế hướng đối tượng: Các mô hình cơ bản trong phân tích và thiết kế hướng đối tượng. Nhà xuất bản thông tin và truyền thông

[3] **Đoàn Văn Ban** (2006). Xây dựng các mô hình UML: Phân tích thiết kế hướng đối tượng bằng UML. NXB Đại học sư phạm

[4] **freeCodeCamp** (2020) “ Khóa học Fullstack PERN”.https://www.youtube.com/watch?v=ldYcgPKEZC8&t=4787s

[5] **bezkoder** (2023) “*React + Node.js + Express + PostgreSQL example: Build a CRUD App*”. https://www.bezkoder.com/react-node-express-postgresql/

[6] **Speckle** (2024). “Developer Docs”. <https://speckle.guide/dev/>

[7] Speckle 3D viewer (2024). “Speckle 3D viewer”.

https://speckle.guide/viewer/