**Giới thiệu về công nghệ đã sử dụng để xây dựng chương trình minh họa**

* Các công nghệ cụ thể:  Java và sử dụng ngôn ngữ SQL Server,ngôn ngữ UML, cầu nối JDBC để kết nối java và cơ sở dữ liệu, quản lý dự án với GitHub
* Tính năng, ưu điểm của các công nghệ này:
  + Java: ngôn ngữ lập trình phổ biến, dễ sử dụng, có thể chạy trên nhiều máy khác nhau( có cài máy ảo java)là công nghệ hướng đối tượng thuần giúp chương trình có khả năng thích nghi cao (khả năng kế thừa tái sử dụng) với môi trường .
  + Ngôn ngữ SQL Server: lưu trữ dữ liệu dễ dàng, truy xuất nhanh chóng , nhất quán , xây dựng cấu trúc đơn giản
  + Ngôn ngữ UML:
    - UML thể hiện phương pháp phân tích hướng đối tượng nên không lệ thuộc ngôn ngữ LT.
    - Hiện nay UML được sử dụng rất phổ biến trong các dự án phần mềm.
    - xây dựng cấu trúc mô tả mang tính thống nhât cho người lập trình trong quá trình mô tả và xây dựng phần mềm , nhất là trong môi trường làm việc nhóm
  + Quản lý dự án với GitHub:
    - GitHub dễ sử dụng (đơn giản), an toàn (quản lý các phiên bản) và nhanh chóng
    - Có thể giúp quy trình làm việc code theo nhóm đơn giản theo nhiều cách nhờ chia nhánh, kết hợp các nhánh( trộn )
    - Có thể làm việc ở nhiều địa điểm khác nhau( không cần gặp nhóm trực tiếp) mà chỉ cần clone mã nguồn từ kho hoặc clone phiên bản từ kho
    - Dễ dàng trong phát triển, quản lý các version
  + Kết nối JDBC:
    - Sử dụng tiếp kiểu dữ liệu hiện có JDBC cho phép làm việc với nhiều cơ sở dữ liệu, ngay cả khi cơ sở dữ liệu được lưu trên nhiều hệ quản trị cơ sở dữ liệu khác nhau
    - Cung cấp độc lập: Việc chuyển đổi giữa các nhà cung cấp cơ sở dữ liệu không có tác động đến chương trình.
* Lý do lại lựa chọn/sử dụng các công nghệ đó: các công nghệ phổ biến, đơn giản, thân thuộc với người sử dụng, được ưa thích cho với các lập trình viên do các ưu điểm nổi bật trên
* Kiến trúc phần mềm sử dụng để xây dựng chương trình:
  + Mô hình MVC (Model - View - Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Nó giúp cho các developer tách ứng dụng của họ ra 3 thành phần khác nhau Model, View và Controller. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.
* Mô tả về kiến trúc MVC:
  + Như đã giới thiệu ở trên , mô hình MVC gồm 3 thành phần  : Model , View, Control :
  + Model : Đây là thành phần chứa tất cả các nghiệp vụ logic, phương thức    xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…
  + View : Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images...Hiểu một cách đơn giản, nó là tập hợp các form ,......
  + Control : Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng... Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.
* Cách đơn giản để mô tả lại luồng sự kiện được xử lý trong MVC:
  + User tương tác với View, bằng cách click vào button, user gửi yêu cầu đi.
  + Controller nhận và điều hướng chúng đến đúng phương thức xử lý ở Model.
  + Model nhận thông tin và thực thi các yêu cầu.
  + Khi Model hoàn tất việc xử lý, View sẽ nhận kết quả từ Model và hiển thị lại cho người dùng.
* Lý do lựa chọn phương pháp :
  + Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế. Do được chia thành các thành phần độc lập nên giúp phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì…
* Các kết quả chương trình đã xây dựng trên kiến trúc này: Dựa vào quy mô chương trình và kết quả áp dụng mô hình này vào thực tiễn , nhóm em nhận thấy chương trình của nhóm đang ở quy mô nhỏ , mức độ tương tác người dùng với chương trình (View) và điều khiển chương trình (control) còn ít nên chương trình sẽ được cải tiến một chút  , giảm sự cồn kềnh không cần thiết , tích hợp phần control vào View  , làm cho quá trình thực hiện thao tác của người dùng linh hoạt hơn .
* Môi trường lập trình: công cụ hỗ trợ lập trình ngôn ngữ java  Eclipse. Do Eclipse là công cụ phổ biến, hỗ trợ nhiều công cụ trong thiết kế, xây dựng coade( hỗ trợ gợi ý, thư viện đa dang…),hỗ trợ một tập hợp không hạn chế các nhà cung cấp công cụ, bao gồm cả các nhà cung cấp phần mềm độc lập (ISV), hỗ trợ các công cụ thao tác các kiểu nội dung bất kỳ (bao gồm cả HTML, Java, C, JSP, EJB, XML, và GIF), ỗ trợ cả môi trường phát triển ứng dụng GUI lẫn không dựa trên GUI, chạy trên nhiều hệ điều hành, bao gồm cả Windows và Linux