BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC XÂY DỰNG HÀ NỘI**

**ĐỒ ÁN**

**LẬP TRÌNH ỨNG DỤNG TRONG XÂY DỰNG**

**Add-in Vẽ nhanh thép cột tương tác với giao diện người dùng**

**<Lớp: XXXX>**

**<Số nhóm: XXXX>**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| NGUYỄN VĂN A | LỚP | MÃ SV |

**Giáo viên hướng dẫn:**

|  |
| --- |
|  |

**Hà Nội, 07/2025**

Lời cảm ơn, em tự viết cho đỡ trùng nhau

**CHƯƠNG 1. TỔNG QUAN**

Trong những năm gần đây, việc ứng dụng Building Information Modeling (BIM) ngày càng phổ biến trong thiết kế và thi công xây dựng. Autodesk Revit là một trong những phần mềm BIM được sử dụng rộng rãi nhất tại Việt Nam và thế giới. Tuy nhiên, Revit mặc định vẫn còn nhiều hạn chế trong việc hỗ trợ người dùng triển khai nhanh các chi tiết kỹ thuật – đặc biệt là cốt thép.

* 1. Các nghiên cứu và công cụ hiện có
     1. Các nghiên cứu và add-in về triển khai cốt thép

Một số add-in thương mại như **SOFiSTiK Reinforcement Detailing**, **Graitec PowerPack for Revit**, hay **IDEA StatiCa Detail** đã cung cấp các công cụ hỗ trợ vẽ và thống kê cốt thép tự động trong Revit. Tuy nhiên, các công cụ này chủ yếu phục vụ thị trường nước ngoài, có chi phí cao và đôi khi không phù hợp hoàn toàn với tiêu chuẩn thiết kế Việt Nam (TCVN 5574:2018, TCVN 5575:2012…).

Ngoài ra, một số nghiên cứu học thuật đã thử nghiệm viết add-in hỗ trợ vẽ thép bằng Revit API. Tuy nhiên, phần lớn các nghiên cứu chỉ dừng lại ở mức cơ bản (vẽ từng thanh), chưa có giải pháp tối ưu cho việc vẽ nhanh, đặc biệt là với cấu kiện bê tông có nhiều biến thể như cột.

* + 1. Ưu và nhược điểm của các tài liệu và phần mềm hiện có

|  |  |  |
| --- | --- | --- |
| **Công cụ / Tài liệu** | Ưu điểm | Nhược điểm |
| SOFiSTiK, Graitec, IDEA | |  | | --- | |  |   Mạnh về tính tự động, tích hợp nhiều tính năng | Chi phí cao, không hỗ trợ tiêu chuẩn Việt Nam |
| Add-in mã nguồn mở (trên GitHub, blog kỹ thuật) | Có thể tùy chỉnh, học hỏi từ mã nguồn | Giao diện kém thân thiện, thiếu tính năng cho thép cột |
| Tài liệu Autodesk Revit API | Hướng dẫn đầy đủ API | |  | | --- | |  |  |  | | --- | | Còn phức tạp, khó tiếp cận với sinh viên mới học lập trình | |

* 1. Nhận xét và khoảng trống nghiên cứu

Qua phân tích, có thể thấy rằng hầu hết các phần mềm hiện có đều chưa giải quyết tốt bài toán **tự động hóa và tối ưu quy trình vẽ thép cột theo tiêu chuẩn Việt Nam** trong Revit. Ngoài ra, các tài liệu hướng dẫn về Revit API tuy có đầy đủ về mặt kỹ thuật nhưng vẫn chưa có một giải pháp cụ thể được thiết kế riêng cho nhu cầu trong nước.

* 1. Tên đề tài và lý do chọn đề tài

Từ các đánh giá trên, tôi chọn đề tài:  
**“Lập trình add-in Revit API hỗ trợ vẽ nhanh thép cột theo tiêu chuẩn Việt Nam”**.

**Lý do chọn đề tài** là nhằm tạo ra một công cụ add-in đơn giản, dễ sử dụng, có khả năng tự động hóa quy trình vẽ cốt thép cột trong Revit – từ đó giảm thời gian triển khai, đảm bảo tính chính xác, đồng thời phù hợp với tiêu chuẩn và thói quen thiết kế trong nước.

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

* 1. Khái quát về cấu kiện cột bê tông cốt thép

Cột bê tông cốt thép là cấu kiện chịu lực thẳng đứng chủ yếu trong công trình, có nhiệm vụ truyền tải trọng từ dầm, sàn xuống móng. Cột có thể có tiết diện vuông, chữ nhật, tròn hoặc các dạng đặc biệt, trong đó phổ biến nhất là vuông và chữ nhật.

Tùy vào nội lực tính toán (moment, lực dọc) và yêu cầu thiết kế, thép cột thường bao gồm:

* **Thép dọc (thép chủ):** chịu lực nén hoặc kéo, thường bố trí dọc theo chiều cao cột.
* **Thép đai:** có chức năng giữ các thanh thép dọc, chống phình, ngăn chặn sự mất ổn định cục bộ và tăng khả năng dẻo của cột.
  1. Các quy định thiết kế liên quan (TCVN 5574:2018)

Theo tiêu chuẩn Việt Nam hiện hành:

* Số lượng và đường kính thép chủ được xác định theo yêu cầu chịu lực, nhưng tối thiểu không ít hơn 4 thanh.
* Khoảng cách giữa các đai tuân theo quy định, thường từ 100–200 mm tùy chiều cao và vị trí của cột (đầu, giữa, chân).
* Đường kính đai tối thiểu là 6 mm, và phải đảm bảo neo buộc đủ chiều dài (thường có dạng móc 135° hoặc 90°).

Các quy định này cần được phản ánh đúng trong quá trình lập trình tự động vẽ cốt thép.

* 1. Cấu kiện cột trong Revit

Cột trong Revit thuộc nhóm Structural Column (BuiltInCategory.OST\_StructuralColumns). Cột có thể là Family dạng Loadable Family (thường là Family cột chữ nhật hoặc vuông).

* 1. XYZ

XYZ là kiểu dữ liệu định nghĩa một điểm hoặc một vector chỉ phương trong không gian mô hình Revit. Kiểu dữ liệu XYZ dùng để định nghĩa các thông tin tọa độ, vector chỉ phương của các đối tượng hình học trong Revit.

Định nghĩa XYZ trong thư viện Revit API:

namespace Autodesk.Revit.DB

{

public class XYZ

{

public XYZ();

public XYZ(double x, double y, double z);

public double this[int idx] { get; }

public static XYZ Zero { get; }

public static XYZ BasisX { get; }

public static XYZ BasisY { get; }

public static XYZ BasisZ { get; }

public double Z { get; }

public double Y { get; }

public double X { get; }

}

}

Các thuộc tính tĩnh đại diện cho kiễu dữ liệu XYZ có thể truy xuất để lấy thông tin hệ tọa độ gốc của mô hình dự án.

XYZ origin = XYZ.Zero; // Gốc tọa độ, có giá trị là {0; 0; 0}

XYZ vecX = XYZ.BasisX; // Vector chỉ phương trục X, có giá trị là {1; 0; 0}

XYZ vecY = XYZ.BasisY; // Vector chỉ phương trục Y, có giá trị là {0; 1; 0}

XYZ vecZ = XYZ.BasisZ; // Vector chỉ phương trục Z, có giá trị là {0; 1; 0}

Ngoài ra, thư viện Revit API còn cung cấp các toán tử và phương thức để hỗ trợ việc tính toán các đối tượng XYZ như:

*Truy xuất các giá trị tọa độ X, Y, Z theo các phương trong hệ trục XYZ:*

double x = point.X;

double y = point.Y;

double z = point.Z;

*Truy xuất chiều dài Vector:*

double length = vec.GetLength();

*Di chuyển một điểm theo (hoặc ngược chiều) vector cho trước:*

XYZ endPnt1 = startPnt + vector;

XYZ endPnt2 = startPnt - vector;

*Phóng to, thu nhỏ độ lớn của một vector với một số:*

XYZ multiVector = vector \* factor;

XYZ divideVector = vector / factor;

*Tích vô hướng, tích có hướng giữa hai vector:*

double d = vector1.DotProduct(vector2);

XYZ vector3 = vector1.CrossProduct(vector2));

* 1. Các thông số (parameters) và cách lấy giá trị của cột trong Revit API
     1. Tên loại cột (Family Name)
* **Ý nghĩa:** Xác định loại family cột mà người dùng đang triển khai, giúp phân biệt kết cấu và hình học của cột.
* **Parameter trong Revit:** Family.Name
* **Cách lấy giá trị bằng API:**

string familyName = column.Symbol.Family.Name;

* + 1. Kích thước tiết diện (b x h)
* **Ý nghĩa:** Xác định hình dạng tiết diện của cột để bố trí thép chủ và thép đai hợp lý.
* **Parameter trong Revit:**
  + Width (chiều rộng)
  + Depth (chiều sâu)
* **Cách lấy giá trị bằng API:**

Parameter widthParam = column.get\_Parameter(“b”);

Parameter depthParam = column.get\_Parameter(“h”);

double width = widthParam.AsDouble(); // đơn vị feet

double depth = depthParam.AsDouble();

* + 1. Chiều cao cột
* **Ý nghĩa:** Dùng để tính toán số đai và chiều dài thép chủ.
* **Parameter trong Revit:**
  + Unconnected Height (nếu cột không gắn level trên)
  + hoặc tính từ Top Offset – Base Offset
* **Cách lấy giá trị bằng API:**

Parameter heightParam = column.get\_Parameter(BuiltInParameter.INSTANCE\_LENGTH);

double height = heightParam.AsDouble();

* + 1. Vị trí đáy / đỉnh cột (Base Level, Top Level)
* **Ý nghĩa:** Xác định cao độ vị trí cột trong công trình.
* **Parameter trong Revit:**
  + Base Level
  + Top Level
* **Cách lấy giá trị bằng API:**

Parameter baseLevelParam = column.get\_Parameter(BuiltInParameter.FAMILY\_BASE\_LEVEL\_PARAM);

Parameter topLevelParam = column.get\_Parameter(BuiltInParameter.FAMILY\_TOP\_LEVEL\_PARAM);

ElementId baseLevelId = baseLevelParam.AsElementId();

ElementId topLevelId = topLevelParam.AsElementId();

Sau đó có thể lấy tên level:

Level baseLevel = doc.GetElement(baseLevelId) as Level;

string baseLevelName = baseLevel.Name;

* + 1. Offset đáy / đỉnh (Base Offset, Top Offset)
* **Ý nghĩa:** hiệu chỉnh cao độ gốc khi đặt cột.
* **Parameter trong Revit:**
  + Base Offset
  + Top Offset
* **Cách lấy giá trị bằng API:**

double baseOffset = column.get\_Parameter(BuiltInParameter.FAMILY\_BASE\_LEVEL\_OFFSET\_PARAM).AsDouble();

double topOffset = column.get\_Parameter(BuiltInParameter.FAMILY\_TOP\_LEVEL\_OFFSET\_PARAM).AsDouble();

* + 1. Vị trí trong không gian (Location)
* **Ý nghĩa:** Lấy tọa độ điểm đặt cột.
* **Cách lấy giá trị bằng API:**

LocationPoint loc = column.Location as LocationPoint;

XYZ point = loc.Point;

double x = point.X;

double y = point.Y;

double z = point.Z;

* 1. Các thông số (Parameter) của đối tượng Thép – Rebar trong Revit

Trong Revit, một đối tượng Rebar đại diện cho một thanh cốt thép đặt trong mô hình. Việc lấy các parameter của Rebar giúp lập trình viên điều khiển chính xác hình dáng, vị trí, số lượng và cấu tạo của thép trong mô hình.

* + 1. Rebar Number
* **Ý nghĩa**: Mã đánh số thanh thép, dùng để phân biệt giữa các nhóm thép khác nhau.
* **Cách lấy**:

string number = rebar.get\_Parameter(BuiltInParameter.REBAR\_NUMBER).AsString();

* **Công dụng**: Phục vụ cho việc gán mã hiệu, phân nhóm trong bảng thống kê.
  + 1. Bar Diameter
* **Ý nghĩa**: Đường kính danh nghĩa của thanh thép (ví dụ: 12mm, 16mm).
* **Cách lấy**:

double diameter = rebar.get\_Parameter(BuiltInParameter.REBAR\_BAR\_DIAMETER).AsDouble(); // feet

* **Công dụng** Dùng để xác định kích thước thép, tính diện tích thép As.
  + 1. Rebar Shape
* **Ý nghĩa**: Hình dạng thanh thép (L, U, thẳng, móc, …).
* **Cách lấy**:

ElementId shapeId = rebar.get\_Parameter(BuiltInParameter.REBAR\_SHAPE).AsElementId();

Element shape = doc.GetElement(shapeId);

string shapeName = shape.Name;

* **Công dụng** Dùng để xác định kích thước thép, tính diện tích thép As.
  1. Tham số thiết kế bổ sung do người dùng nhập

Ngoài ra, phần mềm nên cho phép người dùng khai báo thêm:

* Số lượng thép chủ
* Đường kính thép chủ
* Đường kính và khoảng cách thép đai
* Lớp bảo vệ bê tông
* Kiểu móc neo đai

Các giá trị này có thể lưu vào file cấu hình (JSON) hoặc đặt trong form khi chạy add-in, để chủ động thay đổi mà không phải sửa code.