

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC KINH TẾ - LUẬT**

oooooooo □ □ □ oooooooooo



**BÁO CÁO CUỐI KỲ**

**MÔN: TỰ ĐỘNG HÓA QUY TRÌNH BẰNG ROBOT TRONG TMĐT**

**ĐỀ TÀI: TỰ ĐỘNG HÓA QUY TRÌNH THU THẬP DỮ LIỆU  
TRÊN SÀN BITCOIN VÀ GỬI KẾT QUẢ PHÂN TÍCH**

**Giảng viên hướng dẫn:** Cô Lâm Hồng Thanh

**Mã lớp học phần:** 241MI5801

**Nhóm:** 8

**Sinh viên thực hiện:**

Lai Thị Bảo Trân	- K214111321
Nguyễn Ngọc Mai Chi	- K214111941
Hoàng Thanh Giang	- K214111942
Nguyễn Huỳnh Mai Hân	- K214111943
Nguyễn Thị Như Thảo	- K214111953

*Thành phố Hồ Chí Minh, ngày 29 tháng 10 năm 2024*

# MỤC LỤC

<b>MỤC LỤC</b>	<b>1</b>
<b>DANH MỤC HÌNH ẢNH</b>	<b>2</b>
<b>DANH MỤC BẢNG</b>	<b>3</b>
<b>LỜI MỞ ĐẦU</b>	<b>4</b>
<b>CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN ĐỀ TÀI</b>	<b>5</b>
1.1. Lý do lựa chọn đề tài	5
1.2. Mục tiêu của đề tài	5
1.3. Mục đích của tự động hóa quy trình	6
1.4. Công cụ đánh giá quy trình	6
1.5. Kế hoạch	9
<b>CHƯƠNG 2: THỰC HIỆN DỰ ÁN</b>	<b>20</b>
2.1. Tổng quan dự án	20
2.2. Chi tiết dự án	21
2.3. Demo	24
2.3.1. Python	24
2.3.2. Unipath	29
<b>CHƯƠNG 3. TỔNG KẾT</b>	<b>41</b>
3.1. Tóm tắt quá trình thực hiện dự án	41
3.2. Phát triển trong tương lai	41
3.3. Kết luận	42
<b>BẢNG PHÂN CÔNG VÀ ĐÁNH GIÁ THÀNH VIÊN</b>	<b>43</b>
<b>LỜI CẢM ƠN</b>	<b>44</b>

## DANH MỤC HÌNH ẢNH

<i>Hình 1.1: Kết quả đánh giá quy trình tự động hóa</i>	7
<i>Hình 2.1: Quy trình được ứng dụng tự động hoá</i>	21
<i>Hình 2.2: Sequence Diagrams trong UiPath Studio</i>	22
<i>Hình 2.3: Đoạn code import thư viện và chuẩn hóa dữ liệu</i>	24
<i>Hình 2.4: Đoạn code chuẩn bị dữ liệu cho việc huấn luyện mô hình LSTM</i>	25
<i>Hình 2.5: Đoạn code tạo và huấn luyện mô hình LSTM</i>	26
<i>Hình 2.6: Đoạn code vẽ biểu đồ để so sánh giá thực tế và giá dự đoán của Bitcoin</i>	27
<i>Hình 2.7: Đoạn code dự đoán giá Bitcoin trong 30 ngày</i>	28
<i>Hình 2.8: Khai báo các biến để lấy dữ liệu từ sàn Binance</i>	29
<i>Hình 2.9: Thực hiện lấy dữ liệu từ sàn Binance</i>	30
<i>Hình 2.10: Lưu dữ liệu đọc được từ sàn Binance</i>	31
<i>Hình 2.11: Click chạy code Python</i>	32
<i>Hình 2.12: Truy cập vào hệ thống Viindoo</i>	33
<i>Hình 2.13: Đăng nhập vào hệ thống Viindoo</i>	35
<i>Hình 2.14: Truy cập nơi lưu trữ thông tin nhân viên</i>	36
<i>Hình 2.15: Đọc và lưu dữ liệu từ hệ thống Viindoo</i>	37
<i>Hình 2.16: Thực hiện gửi kết quả phân tích đến email nhân viên</i>	39

## **DANH MỤC BẢNG**

<i>Bảng 1.1: Roles and responsibility</i>	<i>9</i>
<i>Bảng 1.2: Lý do sử dụng tài liệu</i>	<i>15</i>
<i>Bảng 1.3: Assessment plan</i>	<i>18</i>

## **LỜI MỞ ĐẦU**

Nhóm 8 thuộc chuyên ngành Thương mại điện tử của lớp K21411 đã thực hiện đề tài “Tự động hóa quy trình thu thập dữ liệu trên sàn Bitcoin và gửi kết quả phân tích” để phục vụ cho bài đồ án cuối kỳ của nhóm, được hoàn thành hơn 1 tháng kể từ ngày đầu xây dựng ý tưởng, với sự hướng dẫn tận tình từ cô Lâm Hồng Thanh, giảng viên bộ môn mà nhóm đang theo học môn học thuộc chương trình đào tạo “Tự động hóa quy trình bằng Robot trong TMĐT” - Trường Đại học Kinh tế - Luật, ĐHQG HCM. Nhóm đã tiến hành thực hiện đồ án một cách công khai và minh bạch, dựa trên sự cố gắng và nỗ lực của mỗi thành viên trong nhóm cũng như sự hỗ trợ và đóng góp từ giảng viên hướng dẫn.

Trong thời gian thực hiện đồ án, nhóm đã nỗ lực tối đa để phát triển ý tưởng và đáp ứng các yêu cầu ban đầu một cách tốt nhất có thể. Tuy nhiên, việc mắc phải một số sai sót là khó tránh khỏi. Nhóm rất mong nhận được ý kiến đóng góp từ giảng viên để có thể rút kinh nghiệm và hoàn thiện đề tài hơn trong tương lai.

# CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN ĐỀ TÀI

## 1.1. Lý do lựa chọn đề tài

Trong bối cảnh công nghệ phát triển không ngừng, nhu cầu tự động hóa các quy trình làm việc trở nên ngày càng cấp thiết để tối ưu hóa thời gian và tăng tính hiệu quả. Đối với các doanh nghiệp trong lĩnh vực tài chính, việc liên tục theo dõi, thu thập và phân tích dữ liệu từ sàn giao dịch Bitcoin là rất quan trọng. Việc tự động hóa quy trình thu thập dữ liệu và báo cáo phân tích mang lại nhiều lợi ích to lớn cho doanh nghiệp, đặc biệt trong việc giảm thiểu những tác vụ khó như thu thập dữ liệu qua api và mang lại tiện lợi cho các nhà phân tích nhận kết quả cuối cùng là báo cáo trực quan.

Hiện tại quá trình thu thập và xử lý dữ liệu trên sàn Bitcoin vẫn được thực hiện theo các bước thủ công thu thập, lọc dữ liệu, phân tích và lập báo cáo gửi đến các bộ phận liên quan. Nhận thấy việc thực hiện thủ công như thế đã tiêu tốn quá nhiều thời gian và dễ xảy ra sai sót làm giảm hiệu quả tổng thể và độ tin cậy của kết quả phân tích. Để khắc phục những thiếu sót trong quy trình cũ, hệ thống tự động hóa được xây dựng nhằm giảm thiểu tác vụ thủ công và tăng độ chính xác cho kết quả phân tích.

Với hệ thống tự động hóa quy trình thu thập dữ liệu và báo cáo kết quả phân tích từ sàn Bitcoin, doanh nghiệp không chỉ giảm thiểu các thao tác thủ công, nâng cao tính chính xác mà còn đảm bảo rằng thông tin quan trọng được cập nhật liên tục và kịp thời đến các bộ phận. Hệ thống giải quyết các hạn chế hiện tại đồng thời mở ra hướng đi mới cho việc ứng dụng tự động hóa trong việc thu thập dữ liệu và phân tích khác trong tương lai. Do đó, nhóm em đã chọn đề tài: “Tự động hóa quy trình thu thập dữ liệu trên sàn Bitcoin và gửi kết quả phân tích”.

## 1.2. Mục tiêu của đề tài

Để tự động hóa quy trình thu thập dữ liệu, nhóm sẽ phát triển một Bot có khả năng tự động lấy dữ liệu từ sàn Binance trong đúng 12 tháng. Sau khi dữ liệu được lấy về, Bot sẽ tự động chạy mã Python đã lập trình sẵn để phân tích và tính toán các chỉ số cần thiết,

với kết quả được lưu trữ trong thư mục Results. Hệ thống cũng sẽ tự động gửi email báo cáo đến các bộ phận liên quan như bộ phận giao dịch, quản lý rủi ro và chiến lược kinh doanh, đảm bảo thông tin quan trọng luôn được cập nhật kịp thời.

### **1.3. Mục đích của tự động hóa quy trình**

- Quy trình tự động hóa nhằm tiết kiệm thời gian và công sức cho nhân viên bằng cách sử dụng Bot để tự động thu thập và phân tích dữ liệu.
- Bot còn gửi báo cáo phân tích đến các bộ phận liên quan một cách kịp thời, tạo điều kiện cho sự phối hợp hiệu quả giữa các bộ phận như giao dịch, quản lý rủi ro và chiến lược kinh doanh.
- Quy trình cũng giúp quản lý dữ liệu liên tục và linh hoạt vì khi mỗi lần Bot chạy, dữ liệu cũ sẽ được thay thế bằng dữ liệu mới, đảm bảo thông tin phản ánh chính xác tình hình thị trường hiện tại. Doanh nghiệp cũng như người quản lý có thể theo dõi và điều chỉnh quy trình kịp thời khi cần thiết.

### **1.4. Công cụ đánh giá quy trình**

#### **1.4.1 Quy trình trước tự động hóa**

- **Bước 1:** Xác định nhu cầu thu thập dữ liệu: Nhân viên sẽ xác định các loại dữ liệu cần thu thập là giá Bitcoin, khối lượng giao dịch, và các chỉ số tài chính khác. Thời gian thu thập dữ liệu cũng được xác định, thường là từ 12 tháng gần nhất.
- **Bước 2:** Thực hiện thu thập dữ liệu: Nhân viên sẽ đăng nhập vào sàn Binance và sử dụng các công cụ hoặc qua API để truy xuất dữ liệu cần thiết.
- **Bước 3a:** Lọc và làm sạch dữ liệu: Sau khi thu thập, nhân viên sẽ lọc dữ liệu để loại bỏ các giá trị không hợp lệ hoặc không cần thiết.
- **Bước 3b:** Phân tích dữ liệu: Nhân viên sử dụng các công cụ phân tích như Python để tính toán các chỉ số quan trọng như giá trung bình, biến động giá và khối lượng giao dịch. Nhân viên sẽ tạo ra các biểu đồ và dự đoán giá trong 30 ngày tiếp theo.

- **Bước 3c:** Tạo báo cáo kết quả: Kết quả phân tích nhận được là hình dự đoán giá Bitcoin trong 30 ngày tiếp theo.
- **Bước 4:** Gửi kết quả: Nhân viên sẽ gửi kết quả dự đoán phân tích qua email cho các bộ phận liên quan như giao dịch, quản lý rủi ro và chiến lược kinh doanh.

#### 1.4.2 Process Assessment Tool - Partial unattended

RESULTS		
EASE OF IMPLEMENTATION	Low Effort	7%
BENEFIT/SUITABILITY	High Benefit	94%
EASE OF DEVELOPMENT	Automating this process requires professional expertise. Submit this idea to your CoE.	
BANDWIDTH FREED (man hours/year)	56	
ERROR REDUCTION	94%	

**Hình 1.1: Kết quả đánh giá quy trình tự động hóa**

- Ease of Implementation:
  - Được đánh giá là Low Effort nghĩa là quy trình này có thể được triển khai với mức độ phức tạp ít không khó để thực hiện.
  - Với chỉ số 7% cho thấy mức độ khả thi khi triển khai cao trong việc tự động hóa hoàn toàn do doanh nghiệp có sẵn đội ngũ CoE nhưng vẫn cần có sự giám sát và đánh giá về độ chính xác khi dữ liệu được lấy.
  - Nguyên nhân khiến chỉ số khá thấp (ở mức low) vì chỉ số input data có trọng số lớn hơn 80% do dữ liệu thu thập được đều từ sàn Bitcoin dẫn đến quy đổi điểm số nằm ở mức 0 nên hệ số Ease of Implementation thấp hơn.
- Benefit/Suitability:
  - Được đánh giá là High Benefit cho thấy việc tự động hóa quy trình mang lại giá trị cao đáng kể cho doanh nghiệp áp dụng.



- Điểm phần trăm 94%, khi tự động hóa sẽ có thể mang lại lợi ích cao.
- Ease of Development:
  - Phần này yêu cầu kỹ năng chuyên môn cao mới có thể tự động hóa quy trình. Dòng thông báo đề nghị ý doanh nghiệp cần phải thông qua bộ phận CoE để tiến hành tự động một cách chuyên nghiệp và đạt hiệu suất tốt.
  - Nguyên nhân: Dữ liệu đầu vào là Digital and Structured vì cấu trúc dữ liệu đầu vào là dạng bảng đã mặc định.
- Bandwidth Freed: Giá trị 56 giờ/năm thể hiện lượng giờ làm việc của nhân viên được tiết kiệm khi quy trình này được tự động hóa.
- Error Reduction: Mức giảm thiểu lỗi được ước tính là 94% có nghĩa là việc tự động hóa quy trình thu thập dữ liệu trên sàn Bitcoin có thể giúp giảm thiểu hoàn toàn lỗi trong quá trình xử lý thủ công.

Kết luận: Quy trình tự động hóa thu thập dữ liệu trên sàn Bitcoin và gửi kết quả phân tích mang lại nhiều lợi ích cho doanh nghiệp, đặc biệt là tiết kiệm thời gian và giảm thiểu lỗi. Ngoài ra cần có đội ngũ CoE chuyên môn sâu để giám sát quy trình chạy thử và tham khảo ý kiến. Qua quá trình đánh giá nhận thấy được quy trình này đáng để áp dụng vào thực tế để tối ưu hóa cho công ty nhưng cần phải đầu tư về công nghệ cao mới có thể thực hiện thành công.

## 1.5. Kế hoạch

**Bảng 1.1: Roles and responsibility**

Stages	Time	Team	Role	Documents	Key Tasks
Kick - off	1 week	Client Team	Client Project Manager	<ul style="list-style-type: none"> <li>- SOW</li> <li>- Issue Tracker</li> </ul>	<ul style="list-style-type: none"> <li>- Thảo luận về mục tiêu và phạm vi dự án.</li> <li>- Phân công nhiệm vụ cho các thành viên trong nhóm.</li> <li>- Xác định yêu cầu kỹ thuật.</li> <li>- Lập kế hoạch và chuẩn bị cho các giai đoạn sau.</li> </ul>
			IT Team		
		Implementation Team	Project Manager		
Business Case And Technical Validation	2 week	Client Team	Client Project Manager	<ul style="list-style-type: none"> <li>- Business Case Document</li> </ul>	<ul style="list-style-type: none"> <li>- Phân tích lợi ích và chi phí của dự án.</li> </ul>
			Process	<ul style="list-style-type: none"> <li>- Technical Requirements Document</li> </ul>	<ul style="list-style-type: none"> <li>- Đánh giá tính khả thi của giải pháp đề xuất.</li> </ul>

			Owner	<ul style="list-style-type: none"> <li>- Use case</li> <li>- Technology Checklist</li> </ul>	<ul style="list-style-type: none"> <li>- Xác nhận các yêu cầu kỹ thuật.</li> <li>- Đảm bảo và xác nhận các yêu cầu về use case đều được thực hiện đúng với hợp đồng.</li> <li>- Chuẩn bị báo cáo cho Client Team và những role tham gia vào giai đoạn này.</li> </ul>
		Implementation Team	Project Manager		
			Business Analyst		
Process Analysis	2 week	Client Team	Client Project Manager	<ul style="list-style-type: none"> <li>- PDD</li> <li>- As-Is Process Map</li> <li>- To-Be Map</li> <li>- Solution Design Document</li> <li>- UAT</li> </ul>	<ul style="list-style-type: none"> <li>- Phân tích quy trình hiện tại và xác định điểm cải tiến.</li> <li>- Lập kế hoạch cho quy trình mới.</li> <li>- Xác định các chỉ số thành công.</li> <li>- Tham gia vào</li> </ul>
			Process Owner		
			Client IT Team		
		Implementation	Project		

		Team	Manager		các cuộc họp phân tích.
			Business Analyst		- Chuẩn bị báo cáo phân tích.
Solution Design	2 week	Client Team	Client Project Manager	- SDD - Completing the Technical Testing Plan	- Thu thập và xem xét yêu cầu từ phía khách hàng.
		Implementation Team	Project Manager		- Phân tích quy trình hiện tại (As-Is).
			Automation Developers		- Thiết kế quy trình tự động hóa mới. - Lập sơ đồ quy trình (To-Be) và xác định bước triển khai. - Xác định công nghệ cần thiết thông qua Technology Checklist. - Xem xét logic

					tự động hóa, kiểm tra tính khả thi mã nguồn trước khi triển khai.
Develop ment And Testing	6 week	Client Team	Client Project Manager	<ul style="list-style-type: none"> <li>- Code Review Template</li> <li>- SDD</li> <li>- PDD</li> <li>- Technical Testing Plan</li> </ul>	<ul style="list-style-type: none"> <li>- Phát triển tính năng của giải pháp.</li> <li>- Thực hiện kiểm tra lại phần coding (Code Review).</li> <li>- Xác nhận các chức năng mới.</li> <li>- Đảm bảo tính năng đồng nhất với hệ thống.</li> <li>- Thực hiện kiểm tra kỹ thuật và đánh giá kết quả.</li> <li>Sửa lỗi và tối ưu hóa mã nguồn.</li> </ul>
			Process Owner		
			Client IT Team		
		Implementation Team	Project Manager		
			Automation Developers		

User Acceptance Testing (UAT)	2 week	Client Team	Client Project Manager	<ul style="list-style-type: none"> <li>- UAT Planner</li> <li>- Test Case</li> <li>- Issue Tracker</li> </ul>	<ul style="list-style-type: none"> <li>- Lập kế hoạch kiểm thử cho người dùng.</li> <li>- Xác định và thực hiện các trường hợp kiểm thử UAT.</li> <li>- Thu thập phản hồi từ người dùng.</li> <li>- Tham gia vào các buổi kiểm thử UAT và sửa đổi.</li> <li>- Đưa ra các cải tiến nếu cần.</li> <li>- Sửa chữa các lỗi được phát hiện.</li> </ul>
			Process Owner		
			Client IT Team		
		Implementation Team	Project Manager		
			Business Analyst		
			Automation Developers		
Deployment & Hypercare	2 week	Client Team	Client Project Manager	<ul style="list-style-type: none"> <li>-Deployment Plan</li> <li>- Runbook</li> <li>- Hypercare</li> </ul>	<ul style="list-style-type: none"> <li>- Lập kế hoạch triển khai và hướng dẫn người dùng.</li> </ul>

			Automation Operations Manager	Support Document	<ul style="list-style-type: none"> <li>- Đưa ra các giải pháp cho hệ thống.</li> <li>- Giám sát hoạt động và xử lý sự cố trong giai đoạn Hypercare.</li> <li>- Cung cấp hỗ trợ và giải đáp thắc mắc cho người dùng.</li> <li>- Tối ưu hóa quy trình sau triển khai.</li> </ul>
		Implementation Team	Project Manager		
			Automation Developers		
Project Closure	2 week	Client Team	All role	<ul style="list-style-type: none"> <li>- Project Closure Report</li> <li>- Lessons Learned Document</li> <li>- Final Acceptance</li> <li>- Document Archive</li> </ul>	<ul style="list-style-type: none"> <li>- Tổng kết dự án và đánh giá hiệu quả.</li> <li>- Ghi nhận bài học kinh nghiệm.</li> <li>- Hoàn thành các tài liệu cần thiết.</li> <li>- Đề xuất các cải tiến cho các dự án trong tương</li> </ul>
		Implementation Team			

				Document	lai.  - Đánh giá kết quả của quy trình tự động hóa.
--	--	--	--	----------	---

- Lý do sử dụng tài liệu:

**Bảng 1.2: Lý do sử dụng tài liệu**

<b>Tài liệu sử dụng</b>	<b>Lý do</b>
Issue Tracker	Quản lý và theo dõi các vấn đề phát sinh trong suốt quá trình triển khai dự án.
SOW (State of work)	Chứa các yêu cầu về các nhiệm vụ cần thực hiện, phạm vi của dự án, các tiêu chuẩn chất lượng, mục tiêu và các thời hạn cần thiết để hoàn thành.
Business Case Document	Mô tả lý do kinh doanh cho dự án bao gồm lợi ích, chi phí và rủi ro.
Technical Requirements Document	Xác định các yêu cầu kỹ thuật cụ thể cho dự án, đảm bảo sản phẩm cuối cùng đáp ứng yêu cầu người dùng.
Solution Design	Cung cấp chi tiết về thiết kế giải pháp, giúp đảm bảo rằng tất cả các team tham gia hiểu rõ cách giải pháp và



Document	hệ thống hoạt động.
Technology Checklist	Đảm bảo tất cả công nghệ cần thiết được xem xét và đáp ứng trước khi triển khai, giúp giảm thiểu rủi ro kỹ thuật.
PDD (Process Definition Document)	Xác định quy trình hiện tại và quy trình mới, cung cấp cái nhìn sâu sắc về cách mà quy trình sẽ hoạt động sau khi tự động hóa.
As-Is Process Map	Hiểu rõ quy trình hiện tại, từ đó xác định các điểm cần cải tiến và thiết lập quy trình mới.
To-Be Process Map	Mô tả trạng thái mong muốn của quy trình và các bước cần thiết để đạt được mục tiêu đó.
Development Plan	Phác thảo kế hoạch phát triển, giúp quản lý tài nguyên và thời gian hiệu quả trong giai đoạn phát triển.
Code Review Template	Hướng dẫn quá trình kiểm thử mã nguồn, đảm bảo chất lượng mã và phát hiện lỗi trước khi triển khai.
Technical Testing Plan	Xác định các bước kiểm thử kỹ thuật cần thực hiện trước khi triển khai để đảm bảo chất lượng sản phẩm.
UAT Planner and Test	Lập kế hoạch cho các bài kiểm thử chấp nhận của người

Case Template	dùng, đảm bảo rằng sản phẩm đáp ứng nhu cầu người dùng.
Test Cases	Cung cấp thông tin chi tiết về các tình huống kiểm thử cụ thể để đảm bảo tất cả các chức năng đều được kiểm thử.
UAT Results Report	Tóm tắt kết quả của các bài kiểm thử chấp nhận, giúp đánh giá sự hài lòng của người dùng và xác nhận sản phẩm.
Deployment Plan	Lập kế hoạch cho việc triển khai giải pháp, đảm bảo rằng mọi thứ đều được chuẩn bị sẵn sàng cho ngày triển khai.
Runbook	Cung cấp hướng dẫn chi tiết về cách vận hành và quản lý hệ thống sau khi triển khai.
Hypercare Support Document	Cung cấp thông tin hỗ trợ trong giai đoạn sau triển khai, đảm bảo rằng người dùng có thể nhận được sự hỗ trợ cần thiết.
Project Closure Report	Tóm tắt kết quả và kinh nghiệm của dự án, giúp rút ra bài học cho các dự án tương lai.
Lessons Learned Document	Ghi lại những bài học kinh nghiệm trong quá trình thực hiện dự án, giúp cải thiện quy trình trong tương lai.

Final Acceptance	Xác nhận sản phẩm đã được chấp nhận bởi tất cả các bên liên quan, đánh dấu hoàn thành dự án.
Document Archive Document	Cung cấp hướng dẫn cho việc lưu trữ các tài liệu dự án, đảm bảo thông tin có sẵn cho các dự án tương lai.

- Nhóm đã sử dụng các yêu cầu công cụ hỗ trợ sau để thực hiện đồ án:

- Python version 3.9.13
- PyCharm Community Edition version 2024.2.4
- Uipath Studio version 2024 10.1
- Outlook và Email.
- Phần mềm quản trị Viidoo.
- API Binance với API key:  
"q6ZVC0u7Av9tToFMrMSfq7osgunuOmXj3tNoMSkm2wQwWOZYnTDAChc6Y8h7Bcbo" symbol: BTCUSDT dữ liệu lấy theo ngày thời gian lấy là 1 năm tính tới ngày hiện tại.

## CHƯƠNG 2: THỰC HIỆN DỰ ÁN

### 2.1. Tổng quan dự án

Dự án "Tự động hóa quy trình thu thập dữ liệu từ sàn Bitcoin và báo cáo kết quả phân tích" hướng đến việc tự động hóa việc lấy trực tiếp dữ liệu đồng Bitcoin từ sàn giao dịch Binance, phân tích bằng công cụ Python và Machine Learning và báo cáo kết quả gửi cho những bên yêu cầu để hỗ trợ quyết định đầu tư. Với mô hình tự động hóa **Partial Unattended**, quy trình thu thập, phân tích và báo cáo dữ liệu sẽ diễn ra tự động một

phần, yêu cầu người dùng kích hoạt trigger tự động hoá. Dữ liệu Bitcoin từ sàn giao dịch sẽ được thu thập định kỳ mỗi tháng và mỗi lần thu thập sẽ là dữ liệu trong vòng 12 tháng gần nhất và đưa vào hệ thống phân tích nhằm cung cấp các báo cáo về xu hướng thị trường, biến động giá cả và các thông tin liên quan khác.

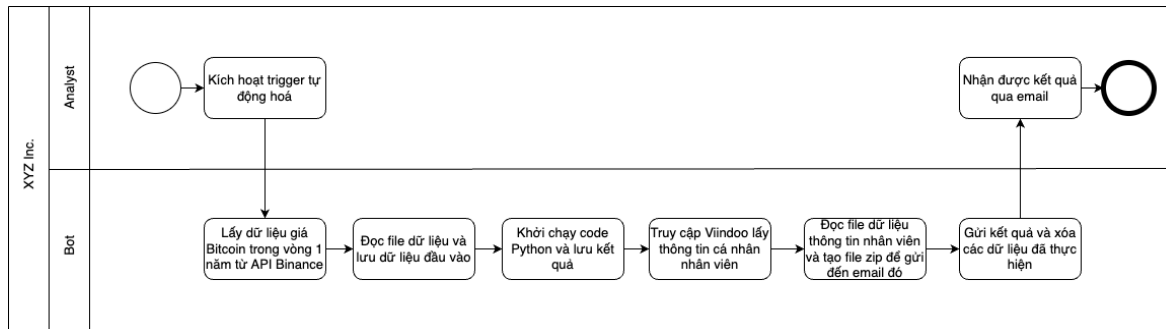
Quy trình tự động hoá trên có thể được sử dụng cho nhiều đối tượng cụ thể như: Nhân viên công ty tài chính chuyên phân tích tiền điện tử, Bộ phận giao dịch, quản lý rủi ro và chiến lược kinh doanh, Các nhà đầu tư cá nhân, tổ chức đầu tư, nhà phân tích tài chính, và bất kỳ ai quan tâm đến việc theo dõi và phân tích giá Bitcoin.

Việc tự động hóa quy trình giúp mang lại một số lợi ích như:

- Giảm thiểu thời gian và công sức cần thiết để thu thập và phân tích dữ liệu, giúp người dùng tập trung vào các quyết định đầu tư.
- Hệ thống tự động đảm bảo việc thu thập dữ liệu diễn ra liên tục và chính xác, giảm thiểu sai sót do con người.
- Người dùng cuối có thể nhận được các báo cáo trực quan và dễ hiểu về xu hướng giá Bitcoin.
- Tự động hóa quy trình giúp giảm thiểu nhu cầu về nguồn lực cho các tác vụ khó như thu thập, phân tích dữ liệu.

## **2.2. Chi tiết dự án**

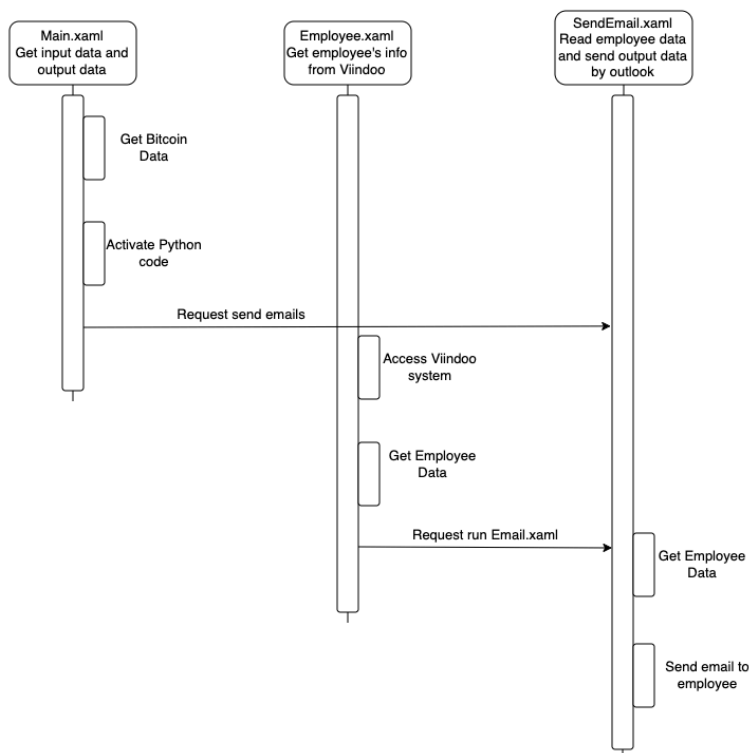
Qua đánh giá từ Process Assessment Tool, quy trình trước đây từ việc cào dữ liệu từ trang web Binance và thực hiện tiếp việc chạy dữ liệu trên hệ thống phân tích dữ liệu, từ đó trích xuất báo cáo thủ công và gửi nộp cho các bên yêu cầu, dẫn đến việc gây tốn thời gian vì gồm nhiều bước thủ công và nguy cơ giảm độ chính xác, nhóm đề xuất một quy trình tự động hoá Partial Unattended có workflow dưới đây:



*Hình 2.1: Quy trình được ứng dụng tự động hoá*

- **Bước 1:** Nhà phân tích (Analyst) sẽ kích hoạt một trigger để bắt đầu quy trình tự động hóa. Đây là bước đầu tiên trong quy trình.
- **Bước 2:** Bot thực hiện xuất các dữ liệu về ngày, khối lượng giao dịch, giá,... của đồng Bitcoin trực tiếp qua API từ sàn Binance trong vòng 12 tháng.
- **Bước 3:** Sau khi cào dữ liệu từ API, Bot thực hiện lưu dữ liệu lại thành một file để làm dữ liệu đầu vào cho các quá trình xử lý tiếp theo.
- **Bước 4:** Bot sẽ khởi chạy đoạn mã Python - trong đây là đoạn mã để phân tích dữ liệu đã thu thập, xử lý, và sau đó lưu lại kết quả phân tích.
- **Bước 5:** Bot truy cập vào hệ thống Viindoo (ERP) để lấy thông tin cá nhân của các nhân viên liên quan (xuất thành 1 file dữ liệu), để phục vụ cho việc gửi kết quả phân tích tới đúng người.
- **Bước 6:** Bot đọc file dữ liệu thông tin nhân viên, sau đó tạo một file zip chứa các kết quả Bitcoin phân tích và thông tin cần thiết để gửi qua email.
- **Bước 7:** Bot gửi file zip chứa kết quả phân tích đến email của người nhận, sau đó xóa các dữ liệu đã thực hiện nhằm đảm bảo tính bảo mật và tránh chiếm dụng bộ nhớ không cần thiết.
- **Bước 8:** Người phân tích/nhân viên liên quan nhận được email với kết quả phân tích trong file đính kèm.

Dưới đây là Sequence Diagrams mô tả các thành phần Sequence trong UIPath Studio tương tác với nhau theo trình tự thời gian



Hình 2.2: Sequence Diagrams trong UIPath Studio

Quy trình trên chia thành ba phần chính (Main.xaml, Employee.xaml, và SendEmail.xaml) và luồng xử lý được diễn giải như sau:

- Main.xaml - Lấy dữ liệu đầu vào và xử lý:
  - Get Bitcoin Data: Tập Main.xaml bắt đầu bằng việc lấy dữ liệu giá Bitcoin, lấy thông qua API của Binance, sau đó tổng hợp thành một file dữ liệu thô.
  - Activate Python code: Sau khi đã có dữ liệu thô, Main.xaml kích hoạt mã Python để xử lý và phân tích dữ liệu Bitcoin đã thu thập.
  - Request send emails: Sau khi phân tích xong, Main.xaml gửi yêu cầu đến quy trình tiếp theo để gửi email chứa kết quả phân tích.

- Employee.xaml - Lấy thông tin nhân viên từ Viindoo
  - Access Viindoo system: Employee.xaml truy cập vào hệ thống Viindoo (ERP - module quản lý nhân viên), để lấy thông tin của nhân viên.
  - Get Employee Data: Sau khi truy cập vào Viindoo, Employee.xaml lấy các thông tin cần thiết của nhân viên, bao gồm email và các thông tin liên lạc khác.
  - Request run Email.xaml: Sau khi có được dữ liệu của nhân viên, Employee.xaml gửi yêu cầu đến SendEmail.xaml để thực hiện gửi email đến nhân viên hoặc người nhận có liên quan.
- SendEmail.xaml - Gửi dữ liệu qua email bằng Outlook
  - Get Employee Data: SendEmail.xaml đọc thông tin nhân viên từ yêu cầu của Employee.xaml để chuẩn bị cho việc gửi email.
  - Send email to employee: SendEmail.xaml thực hiện gửi email chứa kết quả phân tích dữ liệu Bitcoin đến nhân viên hoặc người nhận qua Outlook.

## 2.3. Demo

### 2.3.1. Python

```
1  # Import các thư viện cần thiết
2  from datetime import timedelta
3  import pandas as pd
4  import numpy as np
5  import torch
6  import torch.nn as nn
7  import torch.optim as optim
8  from sklearn.preprocessing import MinMaxScaler
9  from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
10 import matplotlib.pyplot as plt
11 import os
12 # import matplotlib.dates as mdates
13
14 # Tải dữ liệu
15 file_path = r'C:\Users\Hi\Documents\UiPath\Project_1\DataFile\Data.xlsx'
16 df = pd.read_excel(file_path)
17
18 # Nhóm dữ liệu theo ngày và tính giá đóng cửa trung bình mỗi ngày
19 df_close = df.groupby('Open Time')['Close'].mean().reset_index()
20 df_close['Open Time'] = pd.to_datetime(df_close['Open Time'])
21 df_close.set_index('Open Time', inplace=True)
22
23 # Chuẩn hóa dữ liệu
24 scaler = MinMaxScaler(feature_range=(0, 1))
25 scaled_data = scaler.fit_transform(np.array(df_close['Close']).reshape(-1, 1))
26
27 # Chia dữ liệu thành tập huấn luyện và kiểm tra
28 train_size = int(len(scaled_data) * 0.75)
29 train_data = scaled_data[:train_size]
30 test_data = scaled_data[train_size - 60:]
```

Hình 2.3: Đoạn code import thư viện và chuẩn hóa dữ liệu

Đầu tiên sẽ nhập các thư viện cần thiết để xử lý dữ liệu và xây dựng mô hình học máy. Tiếp theo, nhóm sẽ tải dữ liệu từ file Excel và được nhóm theo cột "Open Time" để tính giá trị trung bình của giá đóng cửa trong ngày. Sau khi chuyển đổi cột "Open Time" sang định dạng datetime và đặt làm chỉ số của DataFrame, dữ liệu giá đóng cửa được chuẩn hóa bằng MinMaxScaler để các giá trị nằm trong khoảng từ 0 đến 1. Cuối cùng, dữ liệu được chia thành tập huấn luyện (75%) và tập kiểm tra (25%) để chuẩn bị cho quá trình huấn luyện mô hình.



```

32 # Chuẩn bị dữ liệu huấn luyện cho LSTM
33 def create_dataset(data, n_steps): 2 usages
34     x, y = [], []
35     for i in range(n_steps, len(data)):
36         x.append(data[i-n_steps:i])
37         y.append(data[i])
38     return np.array(x), np.array(y)
39
40 n_steps = 60
41 x_train, y_train = create_dataset(train_data, n_steps)
42
43 x_train = torch.tensor(x_train, dtype=torch.float32)
44 y_train = torch.tensor(y_train, dtype=torch.float32)
45
46 # Định nghĩa mô hình LSTM
47 class LSTMModel(nn.Module): 2 usages
48     def __init__(self):
49         super(LSTMModel, self).__init__()
50         self.lstm1 = nn.LSTM(input_size=1, hidden_size=100, num_layers=2, batch_first=True, dropout=0.2)
51         self.fc1 = nn.Linear(in_features: 100, out_features: 64)
52         self.fc2 = nn.Linear(in_features: 64, out_features: 32)
53         self.fc3 = nn.Linear(in_features: 32, out_features: 1)
54
55     def forward(self, x):
56         out, _ = self.lstm1(x)
57         out = out[:, -1, :] # Chỉ lấy đầu ra cuối cùng của chuỗi
58         out = self.fc1(out)
59         out = self.fc2(out)
60         out = self.fc3(out)
61         return out

```

Hình 2.4: Đoạn code chuẩn bị dữ liệu cho việc huấn luyện mô hình LSTM

Đoạn code trên nhóm chuẩn bị dữ liệu cho việc huấn luyện mô hình LSTM. Đầu tiên, nhóm sẽ tạo ra các tập dữ liệu đầu vào (x) và đầu ra (y) từ dữ liệu huấn luyện, trong đó mỗi đầu vào sẽ bao gồm 60 bước trước đó để dự đoán giá trị tiếp theo. Sau đó, dữ liệu đầu vào và đầu ra được chuyển đổi thành các tensor sử dụng PyTorch. Tiếp theo, một lớp mô hình LSTM được định nghĩa, bao gồm hai lớp LSTM với kích thước đầu vào là 1, kích thước ẩn là 100 và có hai lớp, một dropout 0.2 để tránh overfitting. Lớp này cũng bao gồm ba lớp fully connected để biến đổi đầu ra của LSTM thành một giá trị duy nhất. Cuối cùng, phương thức forward được định nghĩa để thực hiện tính toán qua các lớp này và trả về kết quả đầu ra cuối cùng.

```

63 # Tạo và huấn luyện mô hình
64 model_LSTM = LSTMModel()
65 criterion = nn.MSELoss()
66 optimizer = optim.Adam(model_LSTM.parameters(), lr=0.001)
67
68 # Huấn luyện mô hình và Lưu lịch sử lỗi
69 epochs = 200 # Tăng số lượng epochs
70 history = {'loss': [], 'mean_absolute_error': []}
71
72 for epoch in range(epochs):
73     model_LSTM.train()
74     optimizer.zero_grad()
75     outputs = model_LSTM(x_train.view(-1, n_steps, 1))
76     loss = criterion(outputs, y_train.view(-1, 1))
77     loss.backward()
78     optimizer.step()
79
80     # Lưu lỗi qua các epochs
81     history['loss'].append(loss.item())
82     mae = mean_absolute_error(y_train.detach().numpy(), outputs.detach().numpy())
83     history['mean_absolute_error'].append(mae)
84
85     if (epoch + 1) % 10 == 0:
86         print(f'Epoch [{epoch + 1}/{epochs}], Loss: {loss.item():.4f}, MAE: {mae:.4f}')
87
88 # Dự đoán trên tập huấn luyện
89 model_LSTM.eval()
90 y_pred_train = model_LSTM(x_train.view(-1, n_steps, 1)).detach().numpy()
91
92
93 # Tính toán chỉ số lỗi
94 mae = mean_absolute_error(y_train.numpy(), y_pred_train)
95 mse = mean_squared_error(y_train.numpy(), y_pred_train)
96 r2 = r2_score(y_train.numpy(), y_pred_train)
97
98 # Lưu kết quả dự đoán và lỗi vào file Excel
99 results = pd.DataFrame({
100     "True Price": y_train.numpy().flatten(),
101     "Predicted Price": y_pred_train.flatten()
102 })
103
104 # Đường dẫn tương đối tới tệp cần lưu
105 file_path1 = "C:/Users/Hi/Documents/UiPath/Project_1/Report/PredictedBitcoinPrices.xlsx"
106
107 os.makedirs(os.path.dirname(file_path1), exist_ok=True)
108
109 # Lưu tệp vào đường dẫn đã chỉ định
110 results.to_excel(file_path1, index=False)
111
112 # Lưu các chỉ số lỗi vào file khác
113 metrics = pd.DataFrame({
114     "Metric": ["Mean Absolute Error", "Mean Squared Error", "R^2 Score"],
115     "Value": [mae, mse, r2]
116 })
117
118 file_path1 = "C:/Users/Hi/Documents/UiPath/Project_1/Report/PredictionMetrics.xlsx"
119 metrics.to_excel(file_path1, index=False)

```

Hình 2.5: Đoạn code tạo và huấn luyện mô hình LSTM

Đoạn code trên thực hiện việc tạo và huấn luyện mô hình LSTM. Đầu tiên, khởi tạo mô hình LSTM. Trong quá trình huấn luyện mô hình LSTM, vòng lặp for được sử dụng để thực hiện huấn luyện mô hình qua 200 epochs. Lịch sử lỗi và chỉ số độ chính xác trung bình (MAE) được lưu trữ trong dictionary history. Sau mỗi 10 epochs, thông tin về mất mát và MAE sẽ được in ra màn hình. Sau đó nhóm sẽ tiến hành dự đoán trên tập huấn luyện, tính toán các chỉ số MAE, MSE và R<sup>2</sup> Score được tính toán và lưu vào các file Excel khác nhau bao gồm kết quả dự đoán và các chỉ số. Thư mục lưu trữ file sẽ được tạo ra nếu chưa tồn tại.

```
119 # Vẽ biểu đồ so sánh True vs Predicted Prices
120 plt.figure(figsize=(10, 5))
121 plt.plot(*args: y_train.numpy(), label='True Data')
122 plt.plot(*args: y_pred_train, label='Predicted Data')
123 plt.legend()
124 plt.title("True vs Predicted Prices")
125 plt.xlabel("Time Steps")
126 plt.ylabel("Price")
127 image_path = "C:/Users/Hi/Documents/UiPath/Project_1/Report/Training_Plot.png"
128 plt.savefig(image_path)
129 plt.close()
130
131 # Dự đoán giá trên tập kiểm tra
132 x_test, y_test = create_dataset(test_data, n_steps)
133 x_test = torch.tensor(x_test, dtype=torch.float32)
134
135 # Dự đoán giá trị sử dụng mô hình
136 model_LSTM.eval()
137 predictions = model_LSTM(x_test.view(-1, n_steps, 1)).detach().numpy()
138
139 # Đảo ngược tỉ lệ hóa của các dự đoán để thu được các giá trị thực tế
140 predictions = scaler.inverse_transform(predictions)
141
142 # Đảo ngược việc tỉ lệ hóa trên dữ liệu đầu ra của tập kiểm tra để thu được các giá trị thực tế
143 y_test = scaler.inverse_transform(y_test.reshape(-1, 1))
144
145 # Vẽ biểu đồ dự đoán
146 plt.figure(figsize=(16, 6))
147 plt.title(label: 'Dự đoán giá đóng cửa Bitcoin', fontsize=18)
148 plt.xlabel(xlabel: 'Ngày', fontsize=18)
149 plt.ylabel(ylabel: 'Giá đóng cửa', fontsize=18)
150 plt.plot(*args: df_close.index[train_size:], y_test, label='Giá thực tế', linewidth=2)
151 plt.plot(*args: df_close.index[train_size:], predictions, label='Giá dự đoán', linewidth=2)
152 plt.legend()
153 plt.grid()
154 image_path = "C:/Users/Hi/Documents/UiPath/Project_1/Report/Bitcoin_Price_Predictions.png"
155 plt.savefig(image_path)
156 plt.close()
```

Hình 2.6: Đoạn code vẽ biểu đồ để so sánh giá thực tế và giá dự đoán của Bitcoin

Đoạn code trên sẽ vẽ biểu đồ để so sánh giá thực tế và giá dự đoán của Bitcoin và lưu các biểu đồ này vào file hình ảnh. Đầu tiên là so sánh giá thực tế với giá dự đoán trên tập huấn luyện, sau đó là dự đoán giá trên tập kiểm tra và vẽ biểu đồ cho kết quả. Đoạn code trên sẽ trực quan hóa hơn để đánh giá hiệu suất của mô hình LSTM trong việc dự đoán giá Bitcoin, cho phép người dùng so sánh giá thực tế và giá dự đoán một cách dễ dàng.

```

158 # Dự đoán tương lai
159 n_future = 30
160 future_forecast = []
161
162 input_data = scaled_data[-n_steps:]
163
164 for _ in range(n_future):
165     input_tensor = torch.FloatTensor(input_data).view(1, n_steps, 1)
166     with torch.no_grad():
167         pred_price = model_LSTM(input_tensor)
168         future_forecast.append(pred_price.item())
169         input_data = np.append(input_data[1:], pred_price.numpy())
170
171 # Chuyển đổi kết quả dự đoán về dạng numpy
172 future_forecast = np.array(future_forecast).reshape(-1, 1)
173
174 # Đảo ngược chuẩn hóa để thu được giá thực
175 forecast_prices = scaler.inverse_transform(future_forecast)
176
177 # Tạo các ngày để gán cho dự đoán
178 last_date = df_close.index[-1]
179 future_dates = [last_date + timedelta(days=i) for i in range(1, n_future + 1)]
180
181 # Tạo DataFrame cho dự đoán tương lai
182 future_df = pd.DataFrame(forecast_prices, columns=['Predicted Price'], index=future_dates)
183
184 file_path1 = "C:/Users/Hi/Documents/UiPath/Project_1/Report/Future_Bitcoin_Prices.xlsx"
185 # Lưu kết quả vào file Excel
186 future_df.to_excel(file_path1)
187
188 # Vẽ biểu đồ dự đoán giá Bitcoin trong 30 ngày tới
189 plt.figure(figsize=(10, 5))
190 plt.plot(*args: df_close.index, df_close['Close'], label='Giá thực tế', color='blue', linewidth=2)
191 plt.plot(*args: future_df.index, future_df['Predicted Price'], label='Dự đoán trong 30 ngày tới', color='orange', linestyle='--', linewidth=2)
192 plt.title(label: 'Dự đoán giá Bitcoin trong 30 ngày tới', fontsize=18)
193 plt.xlabel(xlabel: 'Ngày', fontsize=14)
194 plt.ylabel(ylabel: 'Giá Bitcoin', fontsize=14)
195 plt.legend()
196 plt.xticks(rotation=45)
197 plt.grid()
198
199 image_path = "C:/Users/Hi/Documents/UiPath/Project_1/Report/Future_Bitcoin_Price_Predictions.png"
200 # Lưu biểu đồ vào file
201 plt.savefig(image_path)

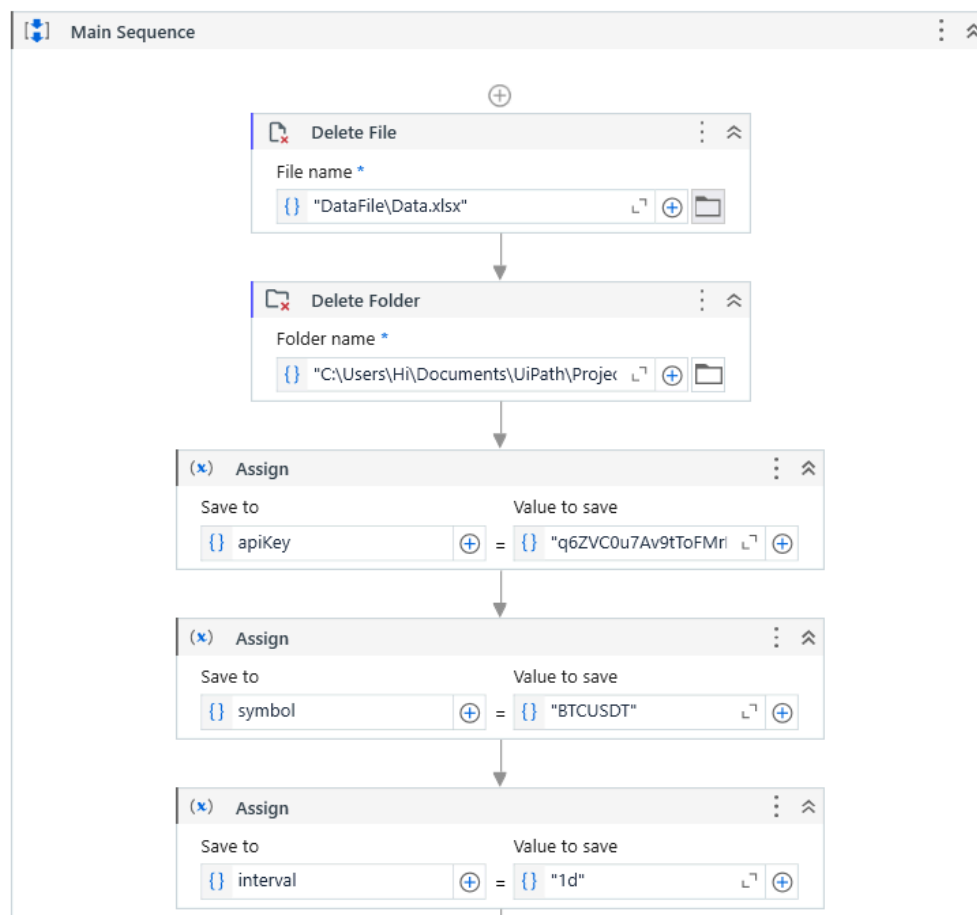
```

Hình 2.7: Đoạn code dự đoán giá Bitcoin trong 30 ngày

Đoạn code trên dự đoán giá Bitcoin trong 30 ngày tới dựa trên mô hình LSTM đã được huấn luyện. Đầu tiên nhóm sẽ chuẩn bị dữ liệu đầu vào, sau đó đưa vào vòng lặp for để dự đoán giá trong tương lai trong 30 lần. Tiếp theo giá dự đoán được chuyển đổi sang dạng numpy, sau đó đảo ngược chuẩn hóa để thu được giá thực tế và lấy ngày cuối cùng trong dữ liệu gốc để tạo một danh sách các ngày trong tương lai để gán cho giá dự đoán. Cuối cùng là lưu kết quả và vẽ biểu đồ.

### 2.3.2. Unipath

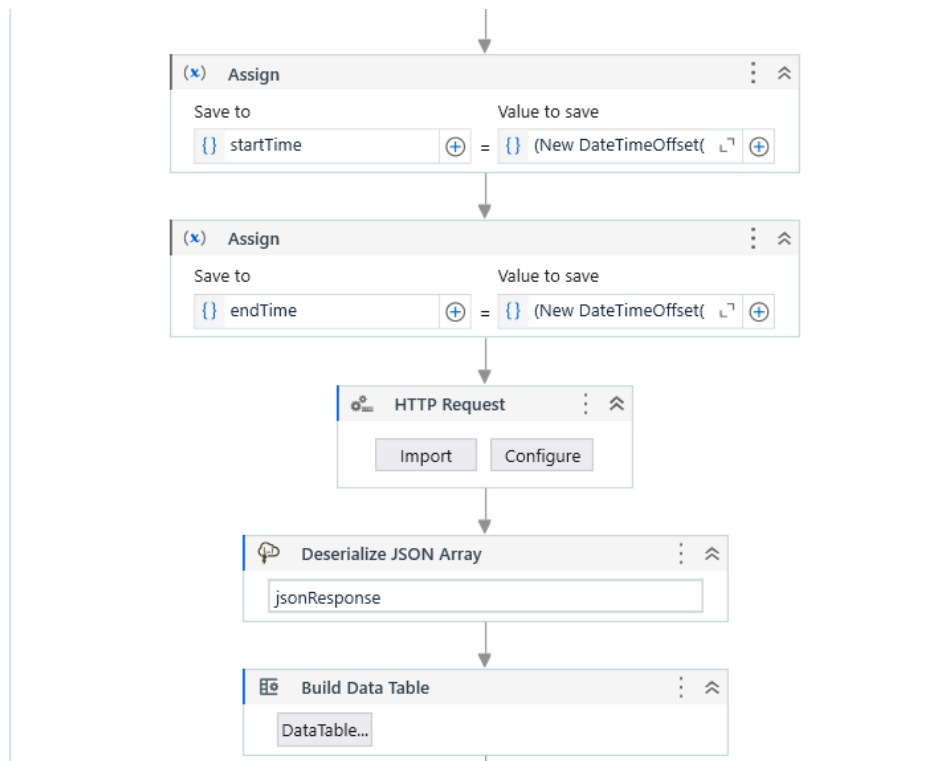
#### - Bước 1:



Hình 2.8: Khai báo các biến để lấy dữ liệu từ sàn Binance

Đầu tiên, nhóm sẽ xóa tệp “DataFile” để dọn dẹp tệp cũ, sau khi chạy thì hệ thống sẽ tự động xóa file cũ tương tự “Delete Folder” để xóa đi những thư mục cũ để tránh trùng lặp với các dữ liệu cũ, tăng độ chính xác đối với dữ liệu và giải phóng bộ nhớ. Tiếp theo là gán giá trị cho các biến “apiKey”, “symbol”, “interval” để lấy dữ liệu theo ngày, thời gian lấy sẽ là 1 năm tính tới ngày hiện tại.

## - Bước 2:



Hình 2.9: Thực hiện lấy dữ liệu từ sàn Binance

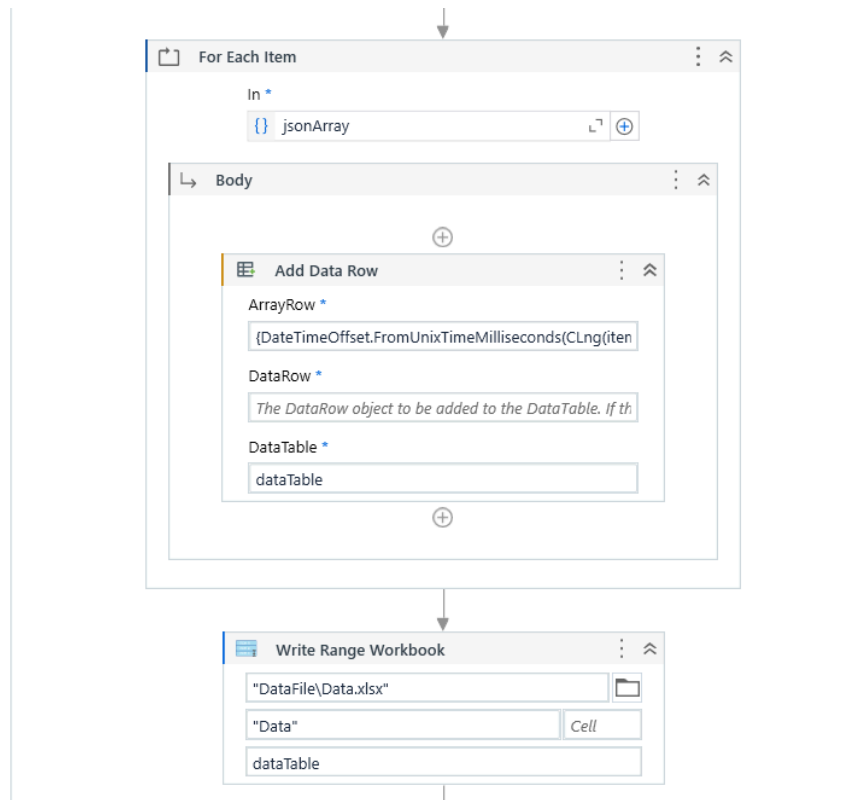
Đối với biến “starTime” và “endTime” thì nhóm sẽ dùng đoạn code sau để gán giá trị:

- `startTime = (New DateTimeOffset(DateTime.Now.AddYears(-1)).ToUnixTimeMilliseconds()).ToString`

- endTime = (New DateTimeOffset(DateTime.Now).ToUnixTimeMilliseconds()).ToString

Với `DateTime.Now.AddYears(-1)` sẽ lấy thời gian hiện tại lùi lại 1 năm tính giá trị cho “startTime”, còn `DateTime.Now` sẽ tính thời gian hiện tại cho biến “endTime”. Sau đó sử dụng HTTP Request để gửi yêu cầu HTTP tới trang API của Binance. Tiếp theo nhóm sử dụng “Deserialize JSON Array” để chuyển đổi chuỗi JSON trong `jsonResponse` thành Array để định dạng dữ liệu có cấu trúc và thuận tiện cho việc Build Table, sau đó dữ liệu sẽ được lưu trữ vào output là ‘jsonArray’ cho bước Build Table tiếp theo.

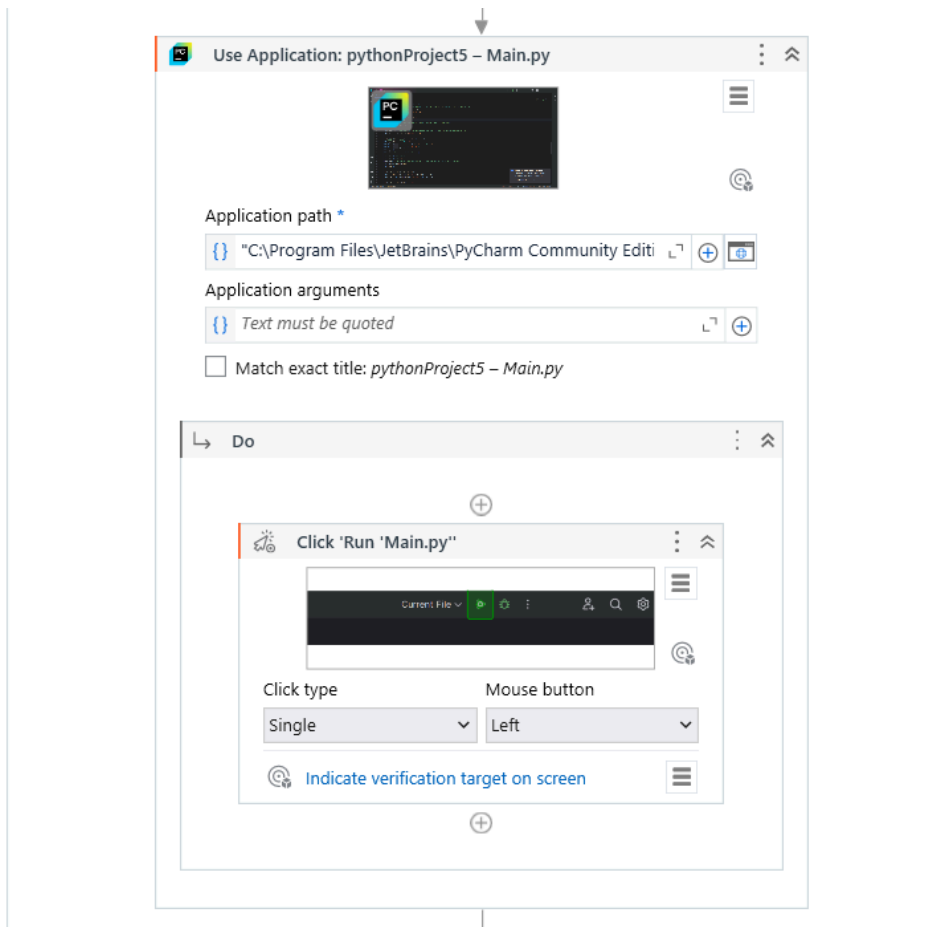
### - Bước 3:



Hình 2.10: Lưu dữ liệu đọc được từ sàn Binance

Tiếp theo hệ thống sẽ duyệt qua từng đối tượng trong “jsonArray” theo từng hàng Array và chèn dữ liệu này vào biến `dataTable`. Sau đó, ghi dữ liệu từ biến `dataTable` vào sheet tên “Data” của file `Data.xlsx`.

- **Bước 4:**



*Hình 2.11: Click chạy code Python*

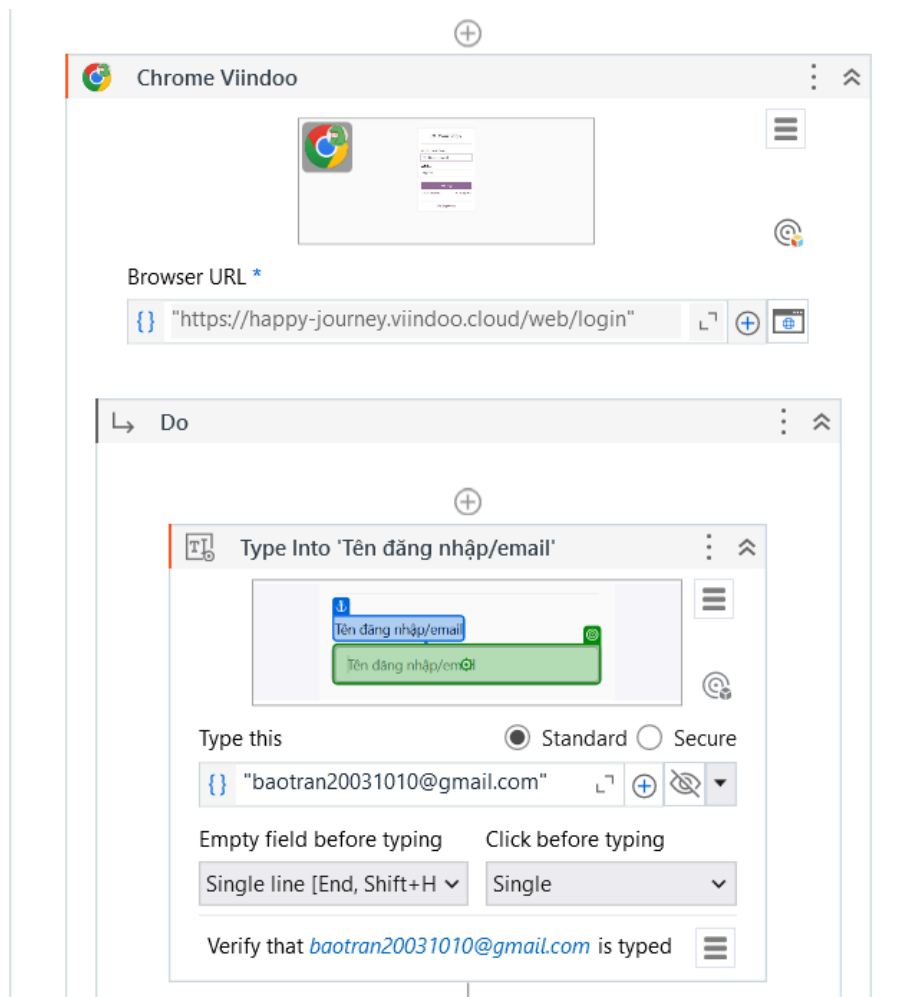
Sau khi đã có file dữ liệu thô của Bitcoin là Data.xlsx ở trên, nhóm tiến hành sử dụng Activity **Use Application/Browser** để sử dụng phần mềm PyCharm chứa đoạn code Python chạy phân tích dữ liệu Bitcoin, trong đó:

- **Application Path:** Chỉ định đường dẫn đến ứng dụng PyCharm Community Edition. Đây là nơi cài đặt PyCharm trên máy tính.
- **Application Arguments:** Dùng để thêm các tham số dòng lệnh khi chạy chương trình. Trong trường hợp này để trống, nghĩa là không có tham số nào được thêm vào.



- **Match exact title:** Tích vào ô này sẽ yêu cầu UiPath phải khớp chính xác tiêu đề của cửa sổ là “pythonProject5 – Main.py”.
- **Do (Click 'Run 'Main.py')):** Đây là một hoạt động để khi Sequence đến Activity này sẽ thực hiện lệnh nhấp vào nút "Run" trong PyCharm.
  - **Click type:** Loại nhấp chuột là "Single" (một lần nhấp chuột).
  - **Mouse button:** là chuột trái “Left”.

- **Bước 5:**

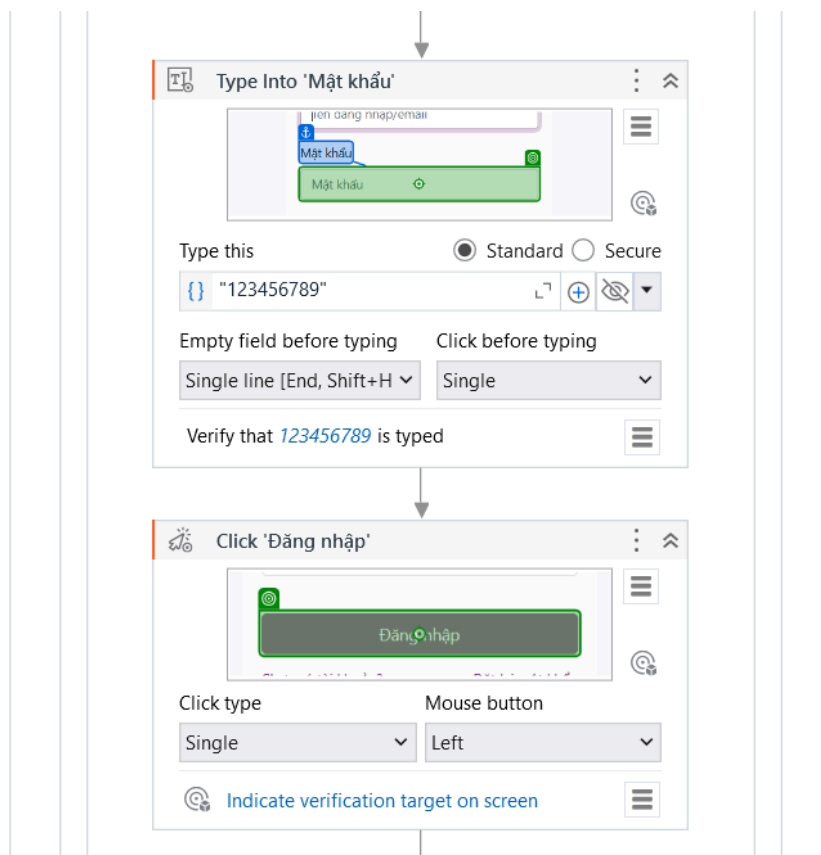


Hình 2.12: Truy cập vào hệ thống Viindoo

Tiếp theo, nhóm thực hiện truy cập vào hệ thống quản trị doanh nghiệp Viindoo để tự động đăng nhập vào tài khoản người dùng đã có trên đây, thực hiện Activity “Chrome Viindoo”:

- **Browser URL:** Chỉ định URL của trang web cần mở, trong trường hợp này là "https://happy-journey.viindoo.cloud/web/login". Đây là trang đăng nhập của hệ thống Viindoo.
- **Do (Type Into 'Tên đăng nhập/email'):** Sau khi truy cập vào đường dẫn trên, nhóm thực hiện chọn trường để tự động điền “Tên đăng nhập” và neo vào Label của nó để xác định vị trí.
  - **Type this:** Chuỗi ký tự sẽ được nhập vào là tài khoản đã có của người dùng "[baotran20031010@gmail.com](mailto:baotran20031010@gmail.com)".
  - **Empty field before typing:** Xóa nội dung cũ trước khi nhập nội dung mới.
  - **Click before typing:** Trước khi nhập liệu, hệ thống sẽ nhấp vào trường này để chắc chắn con trỏ chuột đã nằm ở đúng vị trí.
  - **Verify that [email] is typed:** UiPath sẽ kiểm tra lại để xác nhận rằng địa chỉ email đã được nhập đúng vào trường.

- **Bước 6:**



*Hình 2.13: Đăng nhập vào hệ thống Viindoo*

Tương tự với “Tên đăng nhập”, nhóm sử dụng Activity Type Into 'Mật khẩu' tương tự liên tiếp với Activity trên để tự động nhập vào mật khẩu:

- **Type this:** Chuỗi ký tự sẽ được nhập vào là "123456789".
- **Standard và Secure:** Nếu chọn Secure, mật khẩu sẽ được nhập vào một cách bảo mật, không hiển thị trực tiếp chuỗi ký tự. Trong trường hợp này, không có chọn Secure nên mật khẩu sẽ hiện ở dạng văn bản thường.
- **Empty field before typing:** Xóa nội dung cũ trước khi nhập nội dung mới.
- **Click before typing:** Nhấp vào trường "Mật khẩu" trước khi nhập liệu để đảm bảo rằng con trỏ đang ở đúng vị trí.

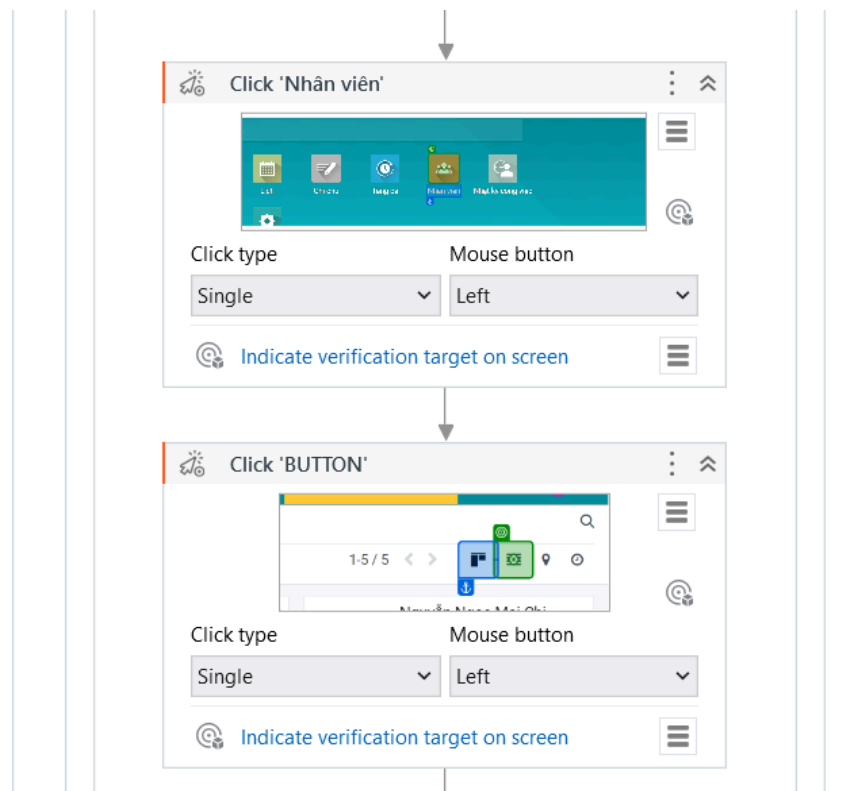
- **Verify that 123456789 is typed:** UiPath sẽ kiểm tra lại để xác nhận rằng mật khẩu đã được nhập đúng vào trường.

- **Bước 7:**

Tiếp theo, sau khi đã nhập thành công tài khoản, hệ thống tự động aim vào button “Đăng nhập” để đăng nhập vào hệ thống.

- **Click type:** Loại nhấp chuột là "Single" (một lần nhấp chuột).
- **Mouse button:** Chuột trái được chọn để nhấp vào nút "Đăng nhập".

→ Việc tự động hoá quá trình đăng nhập giúp người dùng giảm bớt đi thời gian và sai sót.



Hình 2.14: Truy cập nơi lưu trữ thông tin nhân viên

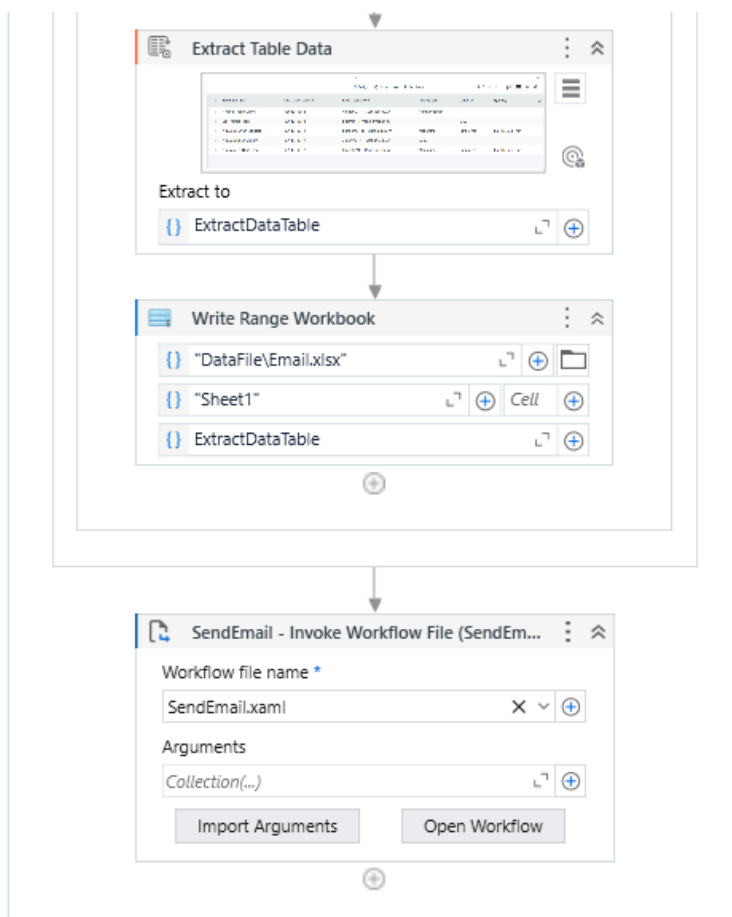
Màn hình tiếp theo hiển thị là màn hình của hệ thống ERP doanh nghiệp bao gồm các Module, trong đó có Module “Nhân viên” là module mà nhóm hướng đến đang lưu trữ thông tin của nhân viên công ty. Thực hiện tự động activity “Click 'Nhân viên’”

- **Click type:** Chọn Single để nhấn một lần vào đối tượng.
- **Mouse button:** Chọn Left để sử dụng nút chuột trái.

Activity Click 'BUTTON': dùng để click chuyển đổi dạng hiển thị trên màn hình module nhân viên thành dạng list.

- **Click type:** Chọn Single để nhấn một lần.
- **Mouse button:** Chọn Left để nhấn bằng chuột trái.

- **Bước 8:**



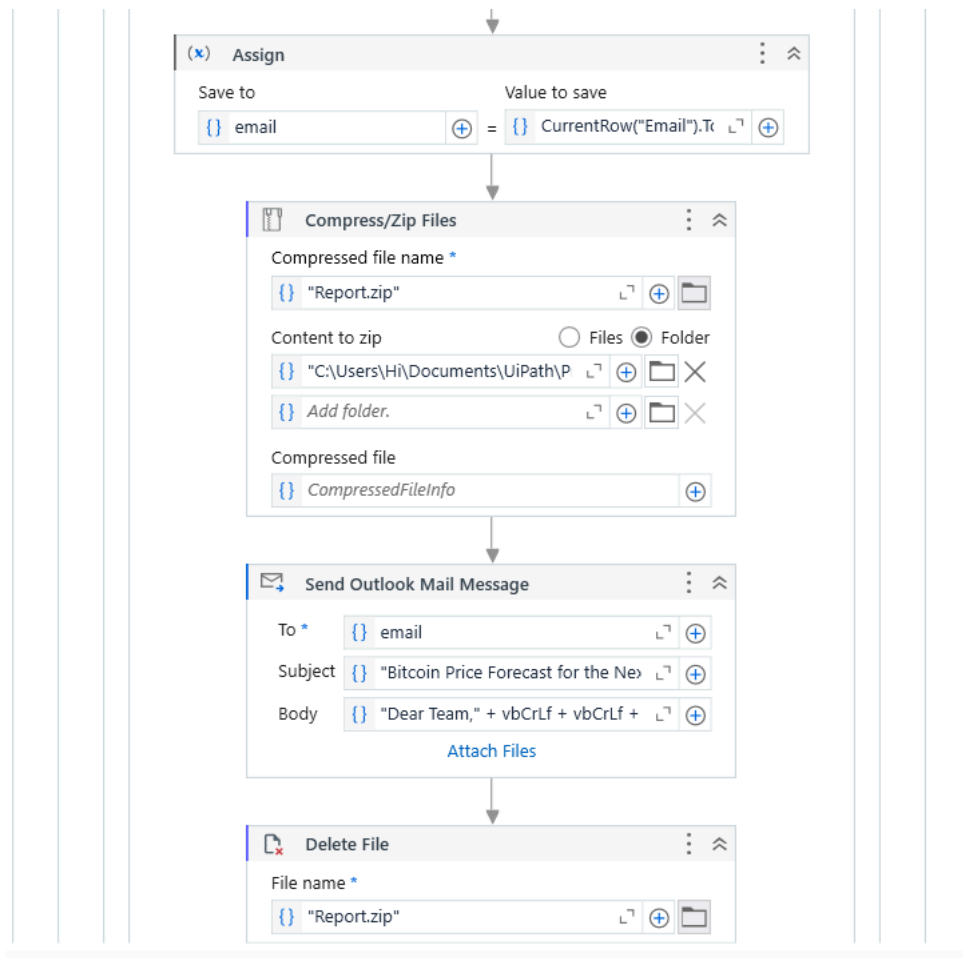
Hình 2.15: Đọc và lưu dữ liệu từ hệ thống Viindoo

Tiếp theo, nhóm định vị dữ liệu thông tin của nhân viên dạng list trên màn hình để trích xuất đúng dữ liệu thành một Data table và lưu thành file Email.xlsx, chạy **Activity ‘Extract Table Data’** và **‘Write Range Workbook’**, trong đó:

- **Activity ‘Extract Table Data’** **Extract to**: Trích xuất dữ liệu ra một biến có tên là ExtractDataTable. Dùng để lấy dữ liệu từ một bảng và lưu trữ trong một biến cho các bước xử lý tiếp theo.
- **Activity ‘Write Range Workbook’**: lệnh này dùng để ghi dữ liệu vào một file Excel.
  - **File path**: Đường dẫn tới file là DataFile\Email.xlsx.
  - **Sheet**: Chọn Sheet1 trong file Excel.
  - **Data**: Ghi dữ liệu từ biến ExtractDataTable vào file Excel.

- **Bước 9:**

Tiếp theo nhóm thực hiện gọi **SendEmail.xaml** để thực hiện tiếp tục workflow tiếp theo.



Hình 2.16: Thực hiện gửi kết quả phân tích đến email nhân viên

Cuối cùng, trong workflow tiếp theo này, nhóm thực hiện truy xuất cột thông tin email của nhân viên cần gửi đến, nén toàn bộ file kết quả sau khi phân tích với Python thành một file .zip và gửi sử dụng Outlook gửi email hàng loạt theo danh sách email đã trích xuất. Cuối cùng, xóa file nén cũ để đảm bảo dữ liệu không bị tồn đọng mỗi khi thực hiện trigger chạy hệ thống, chi tiết giải thích như sau:

- **Activity ‘Assign’:** Lệnh Assign dùng để gán giá trị cho biến email, biến này là một bảng Data.
  - **Value to save:** Giá trị để lưu vào biến email được lấy từ cột Email trong file Email.xlsx.

- **Activity ‘Compress/Zip Files’:** Lệnh Compress/Zip Files dùng để nén file hoặc thư mục thành file zip.
- **Activity ‘Send Outlook Mail Message’:** lệnh này dùng để gửi email qua Outlook.
  - **Compressed file name:** Đặt tên file zip là Report.zip.
  - **Content to zip:** Chọn các file hoặc thư mục cần nén.
- **Activity ‘Send Outlook Mail Message’:** lệnh này dùng để gửi email qua Outlook.
  - **To:** Các địa chỉ email lấy từ biến email.
  - **Subject:** Chủ đề email là "Bitcoin Price Forecast for the Next 30 days".
  - **Body:** Nội dung email được nhóm cố định theo một template.
  - **Attach Files:** Đính kèm file Report.zip vào email.
- **Activity ‘Delete File’:** Lệnh Delete File để xóa file đã chọn.
  - **File name:** file cần xóa là file đã nén ở trên - Report.zip.



## CHƯƠNG 3. TỔNG KẾT

### 3.1. Tóm tắt quá trình thực hiện dự án

Dự án “Tự động hóa quy trình thu thập dữ liệu trên sàn Bitcoin và gửi kết quả phân tích” tập trung vào việc xây dựng một hệ thống bot có khả năng tự động thu thập dữ liệu từ sàn Binance, phân tích thông tin và gửi báo cáo kịp thời đến các bộ phận liên quan. Quy trình bắt đầu với việc thiết kế bot sao cho dễ dàng khởi động sau đó bot sẽ tự động thực hiện các bước từ thu thập dữ liệu giao dịch trong 12 tháng đến lưu trữ kết quả và phân tích các chỉ số cần thiết bằng Python. Sau khi hoàn tất, kết quả phân tích sẽ được lưu vào thư mục Results và gửi tự động qua email đến các bộ phận liên quan sao cho đảm bảo mọi người nắm bắt được thông tin nhanh chóng. Việc tích hợp với hệ thống Viindoo còn giúp bot có thể truy cập thông tin nhân viên nhằm hỗ trợ trong các hoạt động trích xuất thông tin nhanh chóng.

Với sự tập trung vào tự động hóa, dự án này đã tạo ra một quy trình liền mạch và tối giản sự tham gia thủ công của nhân viên. Mỗi lần bot chạy, các dữ liệu cũ đều được làm mới hoàn toàn giúp duy trì độ chính xác và đảm bảo dữ liệu luôn phản ánh đúng tình hình thị trường hiện tại. Quy trình khi triển khai thực tế không chỉ cải thiện hiệu quả làm việc mà còn giảm thiểu rủi ro do lỗi nhập liệu và tiết kiệm thời gian cho các nhân viên tham gia vào các công việc có giá trị cao hơn.

### 3.2. Phát triển trong tương lai

Dự án có tiềm năng phát triển xa hơn với một số cải tiến quan trọng. Trong tương lai, hệ thống có thể mở rộng để thu thập và phân tích dữ liệu từ nhiều nguồn sàn giao dịch khác nhau. Việc tích hợp trí tuệ nhân tạo và machine learning vào quy trình phân tích cũng là một bước tiến đáng cân nhắc nhằm nâng cao độ chính xác của dự báo và cung cấp thông tin phong phú hơn về xu hướng thị trường.

Ngoài ra, một tính năng quản lý sự cố có thể được thêm vào để tự động cảnh báo và xử lý khi có bất kỳ trục trặc nào xảy ra trong quá trình thu thập hoặc phân tích dữ liệu.






Hệ thống cũng có thể phát triển để có khả năng đáp ứng theo nhu cầu tùy chỉnh của người dùng như các báo cáo phân tích chuyên sâu, phân tích nâng cao hoặc các chỉ số bổ sung.

### **3.3. Kết luận**

Dự án “Tự động hóa quy trình thu thập dữ liệu trên sàn Bitcoin và gửi kết quả phân tích” của nhóm đã chứng minh hiệu quả trong việc cải thiện độ chính xác và tốc độ của quy trình phân tích dữ liệu. Việc chuyển đổi từ các quy trình thủ công sang tự động hóa không chỉ giảm tải công việc cho nhân viên mà còn hạn chế sai sót và giúp duy trì dòng thông tin liên tục đảm bảo mọi quyết định đều dựa trên dữ liệu mới nhất. Hệ thống bot còn giúp nâng cao hiệu quả của tổ chức bằng cách tự động gửi báo cáo đến các bộ phận cần thiết và giảm bớt sự phụ thuộc vào thao tác thủ công.

Nhìn chung, hệ thống tự động hóa này là một bước tiến quan trọng cho doanh nghiệp trong việc tối ưu hóa quản lý dữ liệu và tận dụng dữ liệu thị trường. Trong tương lai, việc tích hợp AI và mở rộng quy mô hệ thống sẽ góp phần mang lại nhiều giá trị hơn giúp doanh nghiệp phản ứng nhanh hơn với tình hình thực tế và chuẩn bị tốt hơn cho các quyết định chiến lược.

## BẢNG PHÂN CÔNG VÀ ĐÁNH GIÁ THÀNH VIÊN

STT	MSSV - Họ và Tên	Nhiệm vụ	Mức độ hoàn thành (%)	Xác nhận
1	K214111321 - Lai Thị Bảo Trân	1. Xây dựng workflow 2. Triển khai UiPath	100%	
2	K214111941 - Nguyễn Ngọc Mai Chi	1. Nội dung Chapter 2 2. Tổng hợp nội dung	100%	
3	K214111942 - Hoàng Thanh Giang	1. Nội dung Chapter 1, 2 2. Báo cáo Slide	100%	
4	K214111943 - Nguyễn Huỳnh Mai Hân	1. Nội dung Chapter 1 2. Tổng hợp Word	100%	
5	K214111953 - Nguyễn Thị Như Thảo	1. Triển khai UiPath 2. Tổng hợp Code	100%	

## LỜI CẢM ƠN

Lời đầu tiên cho chúng em xin gửi lời cảm ơn chân thành nhất đến cô Lâm Hồng Thanh. Từ những kiến thức mà chúng em đã tìm hiểu hiểu được và áp dụng nó thông qua bài báo cáo này.

Có lẽ kiến thức là vô hạn mà sự tiếp nhận kiến thức của bản thân mỗi người luôn tồn tại ở những hạn chế. Do đó trong quá trình hoàn thành bài báo cáo, chắc chắn không tránh khỏi những điều thiếu sót. Bản thân chúng em rất mong nhận được những góp ý đến từ thầy để có thể hoàn thiện bản thân hơn.

Kính chúc cô có thật nhiều sức khỏe, hạnh phúc và thành công trên con đường sự nghiệp giảng dạy, mang đến tri thức cho nhiều thế hệ.