

```

1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 import statistics
6 import scipy.stats as stats
7 import datetime
8 import openpyxl

```

```
1 !pip install pymongo
```

```

Collecting pymongo
  Downloading pymongo-4.15.5-cp312-cp312-manylinux2014_x86_64.manylinux_2_17_x86_64.manylinux_2_28_x86_64.whl.metadata (22 kB)
Collecting dnspython<3.0.0,>=1.16.0 (from pymongo)
  Downloading dnspython-2.8.0-py3-none-any.whl.metadata (5.7 kB)
Downloading pymongo-4.15.5-cp312-cp312-manylinux2014_x86_64.manylinux_2_17_x86_64.manylinux_2_28_x86_64.whl (1.7 MB)
----- 1.7/1.7 MB 23.5 MB/s eta 0:00:00
Downloading dnspython-2.8.0-py3-none-any.whl (331 kB)
----- 331.1/331.1 kB 24.7 MB/s eta 0:00:00
Installing collected packages: dnspython, pymongo
Successfully installed dnspython-2.8.0 pymongo-4.15.5

```

```

1 from urllib.parse import quote_plus
2 USERNAME_RAW = "Ninh_DoAn"
3 PASSWORD_RAW = "123456Ninh"
4 HOSTNAME_URI_PART = "ninhbeta.8kk4yvt.mongodb.net"
5 APP_NAME = "ninhbeta"
6 USERNAME_ENCODED = quote_plus(USERNAME_RAW)
7 PASSWORD_ENCODED = quote_plus(PASSWORD_RAW)
8 MONGO_URI = f"mongodb+srv://{USERNAME_ENCODED}:{PASSWORD_ENCODED}@{HOSTNAME_URI_PART}/?appName={APP_NAME}"
9 print("Chuỗi kết nối đã được mã hóa hoàn chỉnh.")

```

Chuỗi kết nối đã được mã hóa hoàn chỉnh.

```

1 import pymongo
2 from pymongo.mongo_client import MongoClient
3 try:
4     client = MongoClient(MONGO_URI)
5     client.admin.command('ping')
6     db = client["DoAn_Congtacnhom"]
7     print("KẾT NỐI THÀNH CÔNG VÀ SẴN SÀNG SỬ DỤNG!")
8 except Exception as e:
9     print(f"Lỗi kết nối: {e}")

```

KẾT NỐI THÀNH CÔNG VÀ SẴN SÀNG SỬ DỤNG!

## ▼ Thành viên

```

1 def import_excel_to_mongo(excel_file_path, Thanh_vien):
2     try:
3         df = pd.read_excel('/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/Thanh_vien.xlsx')
4         data = df.to_dict('records')
5         ngay_hien_tai = datetime.datetime.now()
6         for item in data:
7             item['NgayTG'] = ngay_hien_tai
8         col = db[Thanh_vien]
9         col.delete_many({})
10        if data:
11            col.insert_many(data)
12            print(f"Đã nhập thành công {len(data)} dòng từ '{excel_file_path}' vào collection '{Thanh_vien}'")
13        else:
14            print("File excel trống!")
15    except Exception as e:
16        print(f"Lỗi: {e}")
17 import_excel_to_mongo('Thanh_vien.csv', 'Thanh_vien')

```

Đã nhập thành công 15 dòng từ 'Thanh\_vien.csv' vào collection 'Thanh\_vien'

## ▼ Nhóm

```

1 def import_groups_excel(file_path, collection_name='groups'):
2     try:
3         df = pd.read_excel('/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/Group.xlsx', dtype={'Thanh_vien': str})
4         data_raw = df.to_dict('records')
5         data_clean = []
6         users_col = db['Thanh_vien']
7         for item in data_raw:
8             if 'NgayTao' in item:
9                 item['NgayTao'] = pd.to_datetime(item['NgayTao'], errors='coerce')
10            member_str = item.get('Thanh_vien', '')
11            detailed_members_list = []
12            if pd.isna(member_str) and str(member_str).strip() != '':
13                id_list = [x.strip() for x in str(member_str).split(',') ]
14
15            # --- TRA CỨU USER ---
16            for user_id in id_list:
17                # Tìm user trong database dựa theo ID
18                user_info = users_col.find_one({"_id": user_id})
19                if user_info:
20                    detailed_members_list.append({
21                        "id": user_id,
22                        "HoTen": user_info.get('HoTen', 'Chưa cập nhật tên'),
23                        "role_id": user_info.get('role_id', 'Member')
24                    })
25                else:
26                    detailed_members_list.append({
27                        "id": user_id,
28                        "HoTen": "Không tìm thấy",
29                        "role_id": "Unknown"
30                    })
31            item['Thanh_vien'] = detailed_members_list
32            data_clean.append(item)
33        col = db[collection_name]
34        col.delete_many({})
35        if data_clean:
36            col.insert_many(data_clean)
37            print(f"Nhóm Đã nhập {len(data_clean)} dòng từ '{file_path}'.")
38        else:
39            print(f"File '{file_path}' trống!")
40    except Exception as e:
41        print(f"Lỗi nhập Nhóm: {e}")
42 import_groups_excel('Group.xlsx', 'groups')
43

```

Nhóm Đã nhập 3 dòng từ 'Group.xlsx'.

## ▼ Dự án

```

1 def import_DuAn_excel(excel_file_path):
2     try:
3         df = pd.read_excel('/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/DuAn.xlsx', dtype={'group_id': str})
4         df['NgayBD'] = pd.to_datetime(df['NgayBD'], format='%Y-%m-%d', errors='coerce')
5         df['NgayKT'] = pd.to_datetime(df['NgayKT'], format='%Y-%m-%d', errors='coerce')
6         data = df.to_dict('records')
7         col = db['projects']
8         col.delete_many({})
9         if data:
10            col.insert_many(data)
11            print(f"Đã nhập thành công {len(data)} dự án từ '{excel_file_path}'")
12        else:
13            print("File CSV trống!")
14    except Exception as e:
15        print(f"Lỗi: {e}")
16 import_DuAn_excel('DuAn.xlsx')

```

Đã nhập thành công 3 dự án từ 'DuAn.xlsx'

## ▼ Kiểm tra NaN, Null .Xem trước dữ liệu, xóa dcm cũ để tránh lỗi

```

1 def simple_import_sheet(file_path, sheet_name, collection_name, column_dtypes=None):
2     try:

```

```

3     df = pd.read_excel(file_path, sheet_name=sheet_name, dtype=column_dtypes)
4     df = df.fillna(value=np.nan).replace({np.nan: None})
5     print(f"\n--- Kiểm tra 5 dòng đầu tiên của sheet '{sheet_name}' (Null = None): ---")
6     print(df.head())
7     print("-----")
8     data = df.to_dict('records')
9     ngay_hien_tai = datetime.datetime.now()
10    for item in data:
11        item['NgayImport'] = ngay_hien_tai
12        col = db[collection_name]
13        delete_result = col.delete_many({})
14        if data:
15            col.insert_many(data)
16            print(f"Đã xóa {delete_result.deleted_count} documents cũ.")
17            print(f"Đã nhập thành công {len(data)} dòng từ sheet '{sheet_name}' vào collection '{collection_name}'")
18        else:
19            print("Sheet excel trống!")
20    except Exception as e:
21        print(f"Lỗi khi xử lý sheet '{sheet_name}': {e}")

```

## ▼ Taks

```

1 excel_path = "/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/taks.xlsx"
2 file_name = "taks.xlsx"
3 taks_dtypes = {
4     "_id": str,
5     "TenTask": str,
6     "MoTa_Ta": str,
7     "ID_ThanhVien": str,
8     "ID_DuAn": str
9 }
10 print(f"--- Bắt đầu Import Sheet: Taks từ {file_name} ---")
11 simple_import_sheet(
12     file_path=excel_path,
13     sheet_name="Taks",
14     collection_name="Tasks",
15     column_dtypes=taks_dtypes
16 )

```

--- Bắt đầu Import Sheet: Taks từ taks.xlsx ---

--- Kiểm tra 5 dòng đầu tiên của sheet 'Taks' (Null = None): ---

	_id	TenTask \
0	T01	Khảo sát dự án
1	T02	Phân tích sơ đồ dữ liệu
2	T03	Xác định các Ràng buộc & Quy tắc
3	T04	Thu thập dữ liệu mẫu
4	T05	Làm sạch dữ liệu

  

	MoTa_Ta	Trangthai	NgayBD \
0	Phòng vấn, thu thập các quy trình quản lý ký t...	done	2025-11-08
1	Thiết kế các bảng, mối quan hệ giữa các thực t...	done	2025-11-09
2	Định nghĩa các quy tắc kinh doanh cụ thể. Ví d...	done	2025-11-09
3	Thu thập dữ liệu thực tế (hoặc giả lập) để phụ...	done	2025-11-10
4	Chuẩn hóa định dạng, xử lý dữ liệu thiếu hoặc ...	done	2025-11-22

	NgayKT	ID_ThanhVien	ID_DuAn
0	2025-11-20	A02,A05	C01
1	2025-11-20	A02	C01
2	2025-11-10	A02	C01
3	2025-11-21	A02,A05	C01
4	2025-12-02	A02	C01

-----

Đã xóa 49 documents cũ.

Đã nhập thành công 49 dòng từ sheet 'Taks' vào collection 'Tasks'

## ▼ File

```

1 excel_path = "/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/taks.xlsx"
2 file_dtypes = {
3     "_id": str,
4     "TenFile": str,
5     "ID_Taks": str,
6     "LoaiFile": str,

```

```

7     "DuongDan": str,
8     "ID_NguoiGui": str
9 }
10 print("\n--- Bắt đầu Import Sheet: File ---")
11 simple_import_sheet(
12     file_path=excel_path,
13     sheet_name="File",
14     collection_name="Files",
15     column_dtypes=file_dtypes
16 )

```

--- Bắt đầu Import Sheet: File ---

--- Kiểm tra 5 dòng đầu tiên của sheet 'File' (Null = None): ---

	_id	TenFile	NgayUp	ID_Taks	Loai_File	ID_NguoiGui	\
0	F01	File_test_dulieu	2025-12-06	T08	taks	A02	
1	F02	DuAn_Full	2026-01-01	T015	taks	A03	
2	F03	File_YeuCau	2025-11-08	T017	taks	A06,A010	
3	F04	File_CleanData	2025-11-24	T020	taks	A010	
4	F05	File_test_nhatro	2025-12-04	T023	taks	A010	

DuongDan

0 <https://docs.google.com/document/d/1gFePtyIw40...>

1 <https://docs.google.com/document/d/1gFePtyIw40...>

2 <https://docs.google.com/document/d/1gFePtyIw40...>

3 <https://docs.google.com/document/d/1gFePtyIw40...>

4 <https://docs.google.com/document/d/1gFePtyIw40...>

-----

Đã xóa 9 documents cũ.

Đã nhập thành công 9 dòng từ sheet 'File' vào collection 'Files'

## ▼ Comments

```

1 excel_path = "/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/taks.xlsx"
2 comments_dtypes = {
3     "_id": str,
4     "ID_Taks": str,
5     "NoiDung": str,
6     "ID_Thanhvien": str
7 }
8 print("\n--- Bắt đầu Import Sheet: Comments ---")
9 simple_import_sheet(
10     file_path=excel_path,
11     sheet_name="Comments",
12     collection_name="Comments",
13     column_dtypes=comments_dtypes
14 )

```

--- Bắt đầu Import Sheet: Comments ---

--- Kiểm tra 5 dòng đầu tiên của sheet 'Comments' (Null = None): ---

	_id	NoiDung	NgayCM	ID_Taks	\
0	CM01	Test dữ liệu lỗi quá lỗi rồi team	2025-12-03	T07	
1	CM02	Test dữ liệu thành công không phát hiện lỗi	2025-12-06	T07	
2	CM03	Chọn framework với ngôn ngữ nào đây	2025-12-07	T09	
3	CM04	Giờ chọn trợ ai hỏi đây?	2025-11-08	T017	
4	CM05	Lỗi kết nối database rồi team	2025-12-03	T013	

ID\_Thanhvien

0 A05

1 A05

2 A03

3 A03

4 A01

-----

Đã xóa 10 documents cũ.

Đã nhập thành công 14 dòng từ sheet 'Comments' vào collection 'Comments'

## ▼ Lịch sử hoạt động

```

1 excel_path = "/content/drive/MyDrive/DoAnCoSoNganh_Ninh_DA23TTA/taks.xlsx"
2 lichsu_dtypes = {
3     "_id": str,

```

```

4     "ID_Thanhvien": str,
5     "Mo_ta":str,
6     "TenHD": str
7 }
8 print("\n--- Bắt đầu Import Sheet: Lichsu_hd ---")
9 simple_import_sheet(
10     file_path=excel_path,
11     sheet_name="Lichsu_hd",
12     collection_name="ActivityHistory",
13     column_dtypes=lichsu_dtypes
14 )

```

--- Bắt đầu Import Sheet: Lichsu\_hd ---

--- Kiểm tra 5 dòng đầu tiên của sheet 'Lichsu\_hd' (Null = None): ---

```

      _id                               TenHD \
0 HD01  Kiểm tra độ chính xác của dữ liệu (Test Dữ liệu)
1 HD02                               User Acceptance Test (UAT)
2 HD03  Kiểm tra độ chính xác của dữ liệu (Data Test)
3 HD04  User Acceptance Test (UAT - Kiểm thử chấp nhận)
4 HD05  Kiểm tra độ chính xác của dữ liệu (Data Test)

```

```

      ThoiGian_HD                               Mo_ta \
0 2025-12-06 00:00:00  Kiểm tra tính toàn vẹn, tính duy nhất, và logi...
1 2026-01-04 00:00:00                               Khoa test hệ thống lần cuối.
2 2025-12-06 00:00:00  Kiểm tra ràng buộc dữ liệu (ví dụ: một phòng c...
3 2026-01-04 00:00:00  Thẳng thực hiện các quy trình nghiệp vụ thực t...
4 2025-12-06 00:00:00  Kiểm tra logic giả, tính nhất quán của danh mục...

```

```

ID_Thanhvien
0          A05
1          A02
2          A010
3          A010
4  A013,A014

```

Đã xóa 16 documents cũ.

Đã nhập thành công 16 dòng từ sheet 'Lichsu\_hd' vào collection 'ActivityHistory'

## ✧ Xem thông tin của một Database cụ thể và collection bên trong nó có gì

```

1 database_to_view = input("Nhập tên database: ")
2 print(f"\n--- {database_to_view} ---")
3 try:
4     db = client[database_to_view]
5     collection_names = db.list_collection_names()
6     if not collection_names:
7         print(f"Không có collection nào trong database '{database_to_view}'.")
8     else:
9         for col_name in collection_names:
10             print(f"Collection: {col_name}")
11             collection = db[col_name]
12             print("Sample Documents (first 5):")
13             sample_documents = collection.find().limit(5)
14             found_docs = False
15             for doc in sample_documents:
16                 print(f"        {doc}")
17                 found_docs = True
18             if not found_docs:
19                 print("Không tìm thấy tài liệu nào trong collection này.")
20 except Exception as e:
21     print(f"Lỗi khi truy cập database '{database_to_view}': {e}").pretty()

```

Show hidden output

```

1 # Tìm một gì đó (_id) có (A01)
2 doc = collection.find_one({"_id": "A01"})
3 print(doc)

```

```
{'_id': 'A01', 'HoTen': 'Nguyễn An Ninh', 'Email': 'ninhnguyen@gmail.com', 'role_id': 'member', 'NgàyTG': datetime.datetime(2025
```

```

1 def get_member_info(member_id):
2     try:
3         members_collection = db['Thanh_vien']

```

```

4         member_document = members_collection.find_one({"_id": member_id})
5         return member_document
6     except Exception as e:
7         print(f"Lỗi khi lấy thông tin thành viên: {e}")
8         return None
9
10 def get_member_groups(member_id):
11     try:
12         groups_collection = db['groups']
13         member_groups = list(groups_collection.find({"Thanh_vien": {"$elemMatch": {"id": member_id}}}))
14         return member_groups
15     except Exception as e:
16         print(f"Lỗi khi lấy thông tin nhóm của thành viên: {e}")
17         return []
18
19 def get_projects_for_groups(group_ids):
20     try:
21         projects_collection = db['projects']
22         related_projects = list(projects_collection.find({"group_id": {"$in": group_ids}}))
23         return related_projects
24     except Exception as e:
25         print(f"Lỗi khi lấy thông tin dự án cho các nhóm: {e}")
26         return []
27
28 def get_member_tasks(member_id):
29     try:
30         tasks_collection = db['Tasks']
31         regex_pattern = f".*{member_id}|$).*"
32         member_tasks = list(tasks_collection.find({"ID_ThanhVien": {"$regex": regex_pattern}}))
33         return member_tasks
34     except Exception as e:
35         print(f"Lỗi khi lấy thông tin nhiệm vụ của thành viên: {e}")
36         return []
37
38 def get_full_member_details(member_id):
39     print(f"\n--- Tổng hợp thông tin chi tiết cho thành viên: {member_id} ---")
40
41     member_info = get_member_info(member_id)
42     if member_info:
43         print("\n--- Thông tin thành viên --- ")
44         print(f"ID: {member_info.get('_id')}")
45         print(f"Họ Tên: {member_info.get('HoTen')}")
46         print(f>Email: {member_info.get('Email')}")
47         print(f>Vai trò: {member_info.get('role_id')}")
48         #print(f"MongoDB Query Equivalent: db.Thanh_vien.find_one({'_id': '{member_id}'})")
49     else:
50         print(f"Không tìm thấy thông tin thành viên với ID: {member_id}")
51         return
52
53     member_groups = get_member_groups(member_id)
54     if member_groups:
55         print("\n--- Các nhóm thuộc về --- ")
56         group_ids_for_projects = []
57         for group in member_groups:
58             print(f" - Tên nhóm: {group.get('TenGr')}, ID nhóm: {group.get('_id')}")
59             group_ids_for_projects.append(group.get('_id'))
60             mongodb_query_string_groups = f"db.groups.find({'Thanh_vien': {'$elemMatch': {'id': '{member_id}'}}}))"
61             #print(f"MongoDB Query Equivalent: {mongodb_query_string_groups}")
62     else:
63         print(f"Không tìm thấy nhóm nào cho thành viên với ID: {member_id}")
64         group_ids_for_projects = []
65
66     if group_ids_for_projects:
67         related_projects = get_projects_for_groups(group_ids_for_projects)
68         if related_projects:
69             print("\n--- Các dự án liên quan --- ")
70             for project in related_projects:
71                 print(f" - Tên dự án: {project.get('Name_pr')}, ID dự án: {project.get('_id')}")
72                 group_ids_str = ', '.join([f'{gid}' for gid in group_ids_for_projects])
73                 mongodb_query_string_projects = f"db.projects.find({'group_id': {'$in': [{group_ids_str}]}}))"
74                 #print(f"MongoDB Query Equivalent: {mongodb_query_string_projects}")
75             else:
76                 print(f"Không tìm thấy dự án nào cho các nhóm có ID: {group_ids_for_projects}")
77         else:
78             print("Không có nhóm nào, do đó không có dự án liên quan để hiển thị.")
79
80     member_tasks = get_member_tasks(member_id)

```

```

81     if member_tasks:
82         print("\n--- Các nhiệm vụ được giao --- ")
83         for task in member_tasks:
84             print(f"    - Tên nhiệm vụ: {task.get('TenTask')}, ID nhiệm vụ: {task.get('_id')}")
85             regex_pattern_str = f".*{member_id}.*"
86             mongodb_query_string_tasks = f"db.Tasks.find({'ID_ThanhVien': {'\\$regex': '{regex_pattern_str}'}})"
87             #print(f"MongoDB Query Equivalent: {mongodb_query_string_tasks}")
88         else:
89             print(f"Không tìm thấy nhiệm vụ nào cho thành viên với ID: {member_id}")
90
91 print("Tất cả các hàm đã được định nghĩa.")

```

Tất cả các hàm đã được định nghĩa.

```

1 import time
2 start_time = time.time()
3 get_full_member_details("A013")
4 end_time = time.time()
5 print(f"Thời gian thực thi truy vấn: {end_time - start_time:.4f} giây")

```

```

--- Tổng hợp thông tin chi tiết cho thành viên: A013 ---

--- Thông tin thành viên ---
ID: A013
Họ Tên: Lâm Minh Thiện
Email: minhthien@gmail.com
Vai trò: member

--- Các nhóm thuộc về ---
- Tên nhóm: Nhóm 3, ID nhóm: B03

--- Các dự án liên quan ---
- Tên dự án: Hệ thống quản lý nhà xe sinh viên, ID dự án: C03

--- Các nhiệm vụ được giao ---
- Tên nhiệm vụ: Khảo sát dự án & Phân tích yêu cầu, ID nhiệm vụ: T036
- Tên nhiệm vụ: Phân tích sơ đồ dữ liệu (ERD), ID nhiệm vụ: T037
- Tên nhiệm vụ: Thu thập dữ liệu mẫu, ID nhiệm vụ: T038
- Tên nhiệm vụ: Cài đặt sơ đồ dữ liệu vào hệ thống, ID nhiệm vụ: T040
- Tên nhiệm vụ: Nhập dữ liệu mẫu, ID nhiệm vụ: T041
- Tên nhiệm vụ: Kiểm tra độ chính xác của dữ liệu (Data Test), ID nhiệm vụ: T042
Thời gian thực thi truy vấn: 0.7065 giây

```

```
1 get_full_member_details("A013")
```

```

--- Tổng hợp thông tin chi tiết cho thành viên: A013 ---

--- Thông tin thành viên ---
ID: A013
Họ Tên: Lâm Minh Thiện
Email: minhthien@gmail.com
Vai trò: member

--- Các nhóm thuộc về ---
- Tên nhóm: Nhóm 3, ID nhóm: B03

--- Các dự án liên quan ---
- Tên dự án: Hệ thống quản lý nhà xe sinh viên, ID dự án: C03

--- Các nhiệm vụ được giao ---
- Tên nhiệm vụ: Khảo sát dự án & Phân tích yêu cầu, ID nhiệm vụ: T033
- Tên nhiệm vụ: Phân tích sơ đồ dữ liệu (ERD), ID nhiệm vụ: T034
- Tên nhiệm vụ: Thu thập dữ liệu mẫu, ID nhiệm vụ: T035
- Tên nhiệm vụ: Cài đặt sơ đồ dữ liệu vào hệ thống, ID nhiệm vụ: T037
- Tên nhiệm vụ: Nhập dữ liệu mẫu, ID nhiệm vụ: T038
- Tên nhiệm vụ: Kiểm tra độ chính xác của dữ liệu (Data Test), ID nhiệm vụ: T039
- Tên nhiệm vụ: Triển khai dữ liệu & Tài liệu cho Nhóm Lập trình, ID nhiệm vụ: T040

```

truy vấn: tên dự án , công việc của dự án, người tham gia dự

xem những người trong nhóm , công việc trên nhóm, nhưng trao đổi trên

dự án có nhiều công việc nhất ,ít nhất

nhóm nào có nhiều cv nhất, ít. Người đc giao cv nhiều nhất, ít nhất

.

Thống kê cv nào có nhiều trao đổi nhất, ít nhất

✓ Truy vấn: tên dự án , công việc của dự án, người tham gia dự

```
1 def truyvan_namePr(project_name):
2     print(f"\n--- Tổng hợp thông tin chi tiết cho dự án: {project_name} ---")
3     projects_collection = db['projects']
4     groups_collection = db['groups']
5     tasks_collection = db['Tasks']
6     project = projects_collection.find_one({"Name_pr": project_name})
7     if project:
8         print(f"\n--- Thông tin dự án ---")
9         print(f"ID Dự án: {project.get('_id')}")
10        print(f"Tên Dự án: {project.get('Name_pr')}")
11        print(f"Mô tả: {project.get('Mo_ta')}")
12        print(f"Ngày Bắt Đầu: {project.get('NgàyBD').strftime('%Y-%m-%d')} if project.get('NgàyBD') else 'N/A'}")
13        print(f"Ngày Kết Thúc: {project.get('NgàyKT').strftime('%Y-%m-%d')} if project.get('NgàyKT') else 'N/A'}")
14        group_id = project.get('group_id')
15        if group_id:
16            group = groups_collection.find_one({"_id": group_id})
17            if group and 'Thanh_vien' in group:
18                print(f"\n--- Thành viên tham gia dự án (nhóm {group.get('TenGr')} - ID: {group_id}) ---")
19                for member in group['Thanh_vien']:
20                    print(f" - ID: {member.get('id')}, Họ Tên: {member.get('HoTen')}, Vai trò: {member.get('role_id')}")
21                project_tasks = list(tasks_collection.find({"ID_DuAn": project.get('_id')}))
22                if project_tasks:
23                    print(f"\n--- Các nhiệm vụ của dự án ---")
24                    for task in project_tasks:
25                        print(f" - ID: {task.get('id')}, Tên nhiệm vụ: {task.get('TenTask')}, Trạng thái: {task.get('Trangthai')}")
26        truyvan_namePr("Hệ thống quản lý ký túc xá")
27        #truyvan_namePr("Hệ thống quản lý nhà trọ")
28        #truyvan_namePr("Hệ thống quản lý nhà xe sinh viên")
29
```

--- Tổng hợp thông tin chi tiết cho dự án: Hệ thống quản lý ký túc xá ---

--- Thông tin dự án ---

ID Dự án: C01

Tên Dự án: Hệ thống quản lý ký túc xá

Mô tả: Xây dựng phần mềm nhằm tự động hóa và tối ưu hóa các nghiệp vụ quản lý sinh viên, phòng ốc, thu chi và an ninh trong môi

Ngày Bắt Đầu: 2025-12-06

Ngày Kết Thúc: 2026-01-10

--- Thành viên tham gia dự án (nhóm Nhóm 1 - ID: B01) ---

- ID: A01, Họ Tên: Nguyễn An Ninh, Vai trò: member
- ID: A02, Họ Tên: Huỳnh Tân khoa, Vai trò: role\_leader
- ID: A03, Họ Tên: Từ Gia Đạt, Vai trò: member
- ID: A04, Họ Tên: Thạch Minh Nhật, Vai trò: member
- ID: A05, Họ Tên: Đinh Hữu Nhân, Vai trò: member

--- Các nhiệm vụ của dự án ---

- ID: T01, Tên nhiệm vụ: Khảo sát dự án, Trạng thái: done
- ID: T02, Tên nhiệm vụ: Phân tích sơ đồ dữ liệu, Trạng thái: done
- ID: T03, Tên nhiệm vụ: Xác định các Ràng buộc & Quy tắc, Trạng thái: done
- ID: T04, Tên nhiệm vụ: Thu thập dữ liệu mẫu, Trạng thái: done
- ID: T05, Tên nhiệm vụ: Làm sạch dữ liệu, Trạng thái: done
- ID: T06, Tên nhiệm vụ: Cài đặt sơ đồ dữ liệu vào hệ thống, Trạng thái: done
- ID: T07, Tên nhiệm vụ: Nhập dữ liệu mẫu, Trạng thái: done
- ID: T08, Tên nhiệm vụ: Kiểm tra độ chính xác của dữ liệu (Test Dữ liệu), Trạng thái: done
- ID: T09, Tên nhiệm vụ: Triển khai dữ liệu cho nhóm lập trình, Trạng thái: done
- ID: T010, Tên nhiệm vụ: Thiết kế Kiến trúc/Database, Trạng thái: done
- ID: T011, Tên nhiệm vụ: Phát triển Backend (API), Trạng thái: done
- ID: T012, Tên nhiệm vụ: Phát triển Frontend (UI/UX), Trạng thái: done
- ID: T013, Tên nhiệm vụ: Tích hợp Frontend & Backend, Trạng thái: done
- ID: T014, Tên nhiệm vụ: Unit Test & Integration Test, Trạng thái: done
- ID: T015, Tên nhiệm vụ: Kiểm thử hệ thống (System Test), Trạng thái: done
- ID: T016, Tên nhiệm vụ: Quản lý lỗi, Trạng thái: done
- ID: T017, Tên nhiệm vụ: Fix bug, Trạng thái: done
- ID: T018, Tên nhiệm vụ: Triển khai hệ thống, Trạng thái: done
- ID: T019, Tên nhiệm vụ: User Acceptance Test (UAT), Trạng thái: done



- ✧ Xem những người trong nhóm , công việc trên nhóm, những trao đổi

Thống kê cv nào có nhiều trao đổi nhất, ít nhất

```

1 def get_full_group_details(group_name):
2     print(f"\n--- Tổng hợp thông tin chi tiết cho nhóm: {group_name} ---")
3     groups_collection = db['groups']
4     projects_collection = db['projects']
5     tasks_collection = db['Tasks']
6     comments_collection = db['Comments']
7     group = groups_collection.find_one({"TenGr": group_name})
8     if group:
9         print("\n--- Thông tin nhóm ---")
10        print(f"ID Nhóm: {group.get('_id')}")
11        print(f"Tên Nhóm: {group.get('TenGr')}")
12        print(f"Mô tả: {group.get('Mota_Gr')}")
13        if 'Thanh_vien' in group and group['Thanh_vien']:
14            print("\n--- Thành viên của nhóm ---")
15            for member in group['Thanh_vien']:
16                print(f"    ID: {member.get('id')}, Họ Tên: {member.get('HoTen')}, Vai trò: {member.get('role_id')}")
17        group_id = group.get('_id')
18        if group_id:
19            related_projects = list(projects_collection.find({"group_id": group_id}))
20            if related_projects:
21                print("\n--- Các dự án của nhóm ---")
22                for project in related_projects:
23                    print(f"    - Tên dự án: {project.get('Name_pr')}, ID dự án: {project.get('_id')}")
24                    project_tasks = list(tasks_collection.find({"ID_DuAn": project.get('_id')}))
25                    if project_tasks:
26                        print(f"        --- Nhiệm vụ của dự án '{project.get('Name_pr')}' ---")
27                        task_with_most_comments = None
28                        max_comments = -1
29                        task_with_least_comments = None
30                        min_comments = float('inf')
31                        for task in project_tasks:
32                            comment_count = comments_collection.count_documents({"ID_Taks": task.get('_id')})
33                            print(f"            - Tên nhiệm vụ: {task.get('TenTask')}, ID nhiệm vụ: {task.get('_id')}, Số bình lu
34                            if comment_count > max_comments:
35                                max_comments = comment_count
36                                task_with_most_comments = task
37
38                            if comment_count < min_comments:
39                                min_comments = comment_count
40                                task_with_least_comments = task
41
42                        if task_with_most_comments:
43                            print(f"        Nhiệm vụ có nhiều bình luận nhất: '{task_with_most_comments.get('TenTask')}' (ID: {
44                        if task_with_least_comments:
45                            print(f"        Nhiệm vụ có ít bình luận nhất: '{task_with_least_comments.get('TenTask')}' (ID: {ta
46
47 get_full_group_details("Nhóm 1")
48 get_full_group_details("Nhóm 2")
49 get_full_group_details("Nhóm 3")

```

--- Tổng hợp thông tin chi tiết cho nhóm: Nhóm 1 ---

--- Thông tin nhóm ---

ID Nhóm: B01

Tên Nhóm: Nhóm 1

Mô tả: Xây dựng hệ thống quản lý ký túc xá sinh viên

--- Thành viên của nhóm ---

- ID: A01, Họ Tên: Nguyễn An Ninh, Vai trò: member
- ID: A02, Họ Tên: Huỳnh Tân khoa, Vai trò: role\_leader
- ID: A03, Họ Tên: Từ Gia Đạt, Vai trò: member
- ID: A04, Họ Tên: Thạch Minh Nhật, Vai trò: member
- ID: A05, Họ Tên: Đinh Hữu Nhân, Vai trò: member

--- Các dự án của nhóm ---

- Tên dự án: Hệ thống quản lý ký túc xá, ID dự án: C01
- Nhiệm vụ của dự án 'Hệ thống quản lý ký túc xá' ---
- Tên nhiệm vụ: Khảo sát dự án, ID nhiệm vụ: T01, Số bình luận: 0

```

- Tên nhiệm vụ: Phân tích sơ đồ dữ liệu, ID nhiệm vụ: T02, Số bình luận: 0
- Tên nhiệm vụ: Xác định các Ràng buộc & Quy tắc, ID nhiệm vụ: T03, Số bình luận: 0
- Tên nhiệm vụ: Thu thập dữ liệu mẫu, ID nhiệm vụ: T04, Số bình luận: 0
- Tên nhiệm vụ: Làm sạch dữ liệu, ID nhiệm vụ: T05, Số bình luận: 0
- Tên nhiệm vụ: Cài đặt sơ đồ dữ liệu vào hệ thống, ID nhiệm vụ: T06, Số bình luận: 0
- Tên nhiệm vụ: Nhập dữ liệu mẫu, ID nhiệm vụ: T07, Số bình luận: 2
- Tên nhiệm vụ: Kiểm tra độ chính xác của dữ liệu (Test Dữ liệu), ID nhiệm vụ: T08, Số bình luận: 0
- Tên nhiệm vụ: Triển khai dữ liệu cho nhóm lập trình, ID nhiệm vụ: T09, Số bình luận: 1
- Tên nhiệm vụ: Thiết kế Kiến trúc/Database, ID nhiệm vụ: T010, Số bình luận: 0
- Tên nhiệm vụ: Phát triển Backend (API), ID nhiệm vụ: T011, Số bình luận: 3
- Tên nhiệm vụ: Phát triển Frontend (UI/UX), ID nhiệm vụ: T012, Số bình luận: 1
- Tên nhiệm vụ: Tích hợp Frontend & Backend, ID nhiệm vụ: T013, Số bình luận: 1
- Tên nhiệm vụ: Unit Test & Integration Test, ID nhiệm vụ: T014, Số bình luận: 0
- Tên nhiệm vụ: Kiểm thử hệ thống (System Test), ID nhiệm vụ: T015, Số bình luận: 0
- Tên nhiệm vụ: Quản lý lỗi, ID nhiệm vụ: T016, Số bình luận: 0
- Tên nhiệm vụ: Fix bug, ID nhiệm vụ: T017, Số bình luận: 1
- Tên nhiệm vụ: Triển khai hệ thống, ID nhiệm vụ: T018, Số bình luận: 0
- Tên nhiệm vụ: User Acceptance Test (UAT), ID nhiệm vụ: T019, Số bình luận: 0
Nhiệm vụ có nhiều bình luận nhất: 'Phát triển Backend (API)' (ID: T011) với 3 bình luận.
Nhiệm vụ có ít bình luận nhất: 'Khảo sát dự án' (ID: T01) với 0 bình luận.

--- Tổng hợp thông tin chi tiết cho nhóm: Nhóm 2 ---

--- Thông tin nhóm ---
ID Nhóm: B02
Tên Nhóm: Nhóm 2
Mô tả: Xây dựng hệ thống quản lý nhà trọ

--- Thành viên của nhóm ---
- ID: A06, Họ Tên: Trần Hoàng Long, Vai trò: role_leader
- ID: A07, Họ Tên: Khuu Gia Bảo, Vai trò: member
- ID: A08, Họ Tên: Trần Phú Đình, Vai trò: member
- ID: A09, Họ Tên: Thang Kim Hoàng Quý, Vai trò: member
- ID: A010, Họ Tên: Trần Quốc Thắng, Vai trò: member

--- Các dự án của nhóm ---
- Tên dự án: Hệ thống quản lý nhà trọ, ID dự án: C02
--- Nhiệm vụ của dự án 'Hệ thống quản lý nhà trọ' ---
- Tên nhiệm vụ: Khảo sát dự án & Phân tích yêu cầu, ID nhiệm vụ: T01, Số bình luận: 0

```

## ✓ Dự án có nhiều công việc nhất ,ít nhất.

```

1 def phan_tich_nhiem_vu_du_an():
2     # In tiêu đề phân tích ra màn hình
3     print("\n--- Số lượng nhiệm vụ cho dự án ---")
4
5     # Kết nối các bộ sưu tập (collections) từ cơ sở dữ liệu
6     du_an_collection = db['projects']
7     nhiem_vu_collection = db['Tasks']
8
9     # Dùng Aggregation Pipeline để nhóm các nhiệm vụ theo ID dự án và đếm số lượng
10    pipeline = [
11        {
12            # Nhóm theo trường ID_DuAn và tính tổng số lượng nhiệm vụ
13            "$group": {
14                "_id": "$ID_DuAn",
15                "so_luong_nhiem_vu": {"$sum": 1}
16            }
17        },
18        {
19            # Sắp xếp theo số lượng nhiệm vụ giảm dần (-1)
20            "$sort": {"so_luong_nhiem_vu": -1}
21        }
22    ]
23
24    # Thực thi truy vấn thống kê
25    thong_ke_nhiem_vu = list(nhiem_vu_collection.aggregate(pipeline))
26
27    if not thong_ke_nhiem_vu:
28        print("Không có nhiệm vụ nào được tìm thấy trong cơ sở dữ liệu.")
29        return
30
31    # Chuyển đổi kết quả sang DataFrame (Pandas) để dễ dàng thao tác
32    df_thong_ke = pd.DataFrame(thong_ke_nhiem_vu)
33    df_thong_ke.rename(columns={'_id': 'id_du_an'}, inplace=True)
34
35    # Xác định dự án có nhiều nhiệm vụ nhất
36    du_an_nhiều_nhat = df_thong_ke.loc[df_thong_ke['so_luong_nhiem_vu'].idxmax()]

```

```

37 id_du_an_nhiều = du_an_nhiều_nhat['id_du_an']
38 dem_nhiều_nhat = du_an_nhiều_nhat['so_luong_nhiem_vu']
39
40 # Lay thông tin chi tiết (tên dự án) từ collection projects
41 info_du_an_nhiều = du_an_collection.find_one({"_id": id_du_an_nhiều})
42 ten_du_an_nhiều = info_du_an_nhiều.get('Name_pr', 'Tên dự án không xác định') if info_du_an_nhiều else 'Không tìm thấy'
43
44 # Xác định dự án có ít nhiệm vụ nhất
45 du_an_it_nhat = df_thong_ke.loc[df_thong_ke['so_luong_nhiem_vu'].idxmin()]
46 id_du_an_it = du_an_it_nhat['id_du_an']
47 dem_it_nhat = du_an_it_nhat['so_luong_nhiem_vu']
48
49 # Lay thông tin chi tiết từ collection projects
50 info_du_an_it = du_an_collection.find_one({"_id": id_du_an_it})
51 ten_du_an_it = info_du_an_it.get('Name_pr', 'Tên dự án không xác định') if info_du_an_it else 'Không tìm thấy dự án'
52
53 # In kết quả phân tích cuối cùng
54 print(f"\n--- Kết quả --- ")
55 print(f"Dự án có nhiều nhiệm vụ nhất: '{ten_du_an_nhiều}' (ID: {id_du_an_nhiều}) với {dem_nhiều_nhat} nhiệm vụ.")
56 print(f"Dự án có ít nhiệm vụ nhất: '{ten_du_an_it}' (ID: {id_du_an_it}) với {dem_it_nhat} nhiệm vụ.")
57
58 phan_tich_nhiem_vu_du_an()

```

--- Số lượng nhiệm vụ cho dự án ---

--- Kết quả phân tích ---

Dự án có nhiều nhiệm vụ nhất: 'Hệ thống quản lý ký túc xá' (ID: C01) với 19 nhiệm vụ.

Dự án có ít nhiệm vụ nhất: 'Hệ thống quản lý nhà xe sinh viên' (ID: C03) với 14 nhiệm vụ.

## ▼ Nhóm nào có nhiều cv nhất, ít nhất.

```

1 def phan_tich_nhiem_vu_nhom():
2
3     print("\n--- Số lượng nhiệm vụ cho mỗi nhóm ---")
4
5     # Kết nối các bộ sưu tập (collections) từ cơ sở dữ liệu
6     cong_viec_collection = db['Tasks']
7     du_an_collection = db['projects']
8     nhom_collection = db['groups']
9
10    # Thiết lập pipeline để truy vấn và thống kê dữ liệu từ nhiều bảng
11    pipeline = [
12        {
13            # Kết hợp dữ liệu với bảng 'projects' dựa trên ID dự án
14            "$lookup": {
15                "from": "projects",
16                "localField": "ID_DuAn",
17                "foreignField": "_id",
18                "as": "thong_tin_du_an"
19            }
20        },
21        {
22            # Trai phang mảng thông tin dự án để dễ dàng truy cập thuộc tính
23            "$unwind": "$thong_tin_du_an"
24        },
25        {
26            # Nhóm dữ liệu theo ID nhóm và đếm tổng số công việc trong mỗi nhóm
27            "$group": {
28                "_id": "$thong_tin_du_an.group_id",
29                "so_luong_nhiem_vu": {"$sum": 1}
30            }
31        },
32        {
33            # Sắp xếp danh sách theo số lượng công việc giảm dần
34            "$sort": {"so_luong_nhiem_vu": -1}
35        }
36    ]
37
38    # Thực thi truy vấn thống kê và chuyển kết quả thành danh sách
39    thong_ke_nhom = list(cong_viec_collection.aggregate(pipeline))
40
41    if not thong_ke_nhom:
42        print("Không có nhiệm vụ nào được tìm thấy hoặc không liên kết với nhóm.")
43    return

```

```

44
45 # Chuyen doi sang DataFrame de thao tac du lieu de dang hon
46 df_thong_ke_nhom = pd.DataFrame(thong_ke_nhom)
47 df_thong_ke_nhom.rename(columns={'_id': 'id_nhom'}, inplace=True)
48
49 # Xac dinh nhom co nhieu nhieu vu nhat
50 nhom_nhieu_nhat = df_thong_ke_nhom.loc[df_thong_ke_nhom['so_luong_nhiem_vu'].idxmax()]
51 id_nhom_nhieu = nhom_nhieu_nhat['id_nhom']
52 dem_nhieu = nhom_nhieu_nhat['so_luong_nhiem_vu']
53 # Lay ten nhom tu bang 'groups'
54 info_nhom_nhieu = nhom_collection.find_one({"_id": id_nhom_nhieu})
55 ten_nhom_nhieu = info_nhom_nhieu.get('TenGr', 'Ten nhom khong xac dinh') if info_nhom_nhieu else 'Khong tim thay nhom'
56
57 # Xac dinh nhom co it nhieu vu nhat
58 nhom_it_nhat = df_thong_ke_nhom.loc[df_thong_ke_nhom['so_luong_nhiem_vu'].idxmin()]
59 id_nhom_it = nhom_it_nhat['id_nhom']
60 dem_it = nhom_it_nhat['so_luong_nhiem_vu']
61 # Lay ten nhom tu bang 'groups'
62 info_nhom_it = nhom_collection.find_one({"_id": id_nhom_it})
63 ten_nhom_it = info_nhom_it.get('TenGr', 'Ten nhom khong xac dinh') if info_nhom_it else 'Khong tim thay nhom'
64
65 # In ket qua phan tich cuoi cung
66 print("\n--- Ket qua phan tich --- ")
67 print(f"Nhóm có nhiều nhiệm vụ nhất: '{ten_nhom_nhieu}' (ID: {id_nhom_nhieu}) với {dem_nhieu} nhiệm vụ.")
68 print(f"Nhóm có ít nhiệm vụ nhất: '{ten_nhom_it}' (ID: {id_nhom_it}) với {dem_it} nhiệm vụ.")
69
70 phan_tich_nhiem_vu_nhom()

```

--- So luong nhieu vu cho moi nhom ---

--- Ket qua phan tich ---

Nhom co nhieu nhieu vu nhat: 'Nhóm 1' (ID: B01) với 19 nhieu vu.

Nhom co it nhieu vu nhat: 'Nhóm 3' (ID: B03) với 14 nhieu vu.

```

1 def phan_tich_nhiem_vu_nhom():
2
3     print("\n--- Số lượng nhiệm vụ cho mỗi nhóm ---")
4
5     # Ket noi cac bo suu tap (collections) tu co so du lieu
6     cong_viec_collection = db['Tasks']
7     du_an_collection = db['projects']
8     nhom_collection = db['groups']
9
10    # Thiet lap pipeline de truy van va thong ke du lieu tu nhieu bang
11    pipeline = [
12        {
13            # Ket hop du lieu voi bang 'projects' dua tren ID du an
14            "$lookup": {
15                "from": "projects",
16                "localField": "ID_DuAn",
17                "foreignField": "_id",
18                "as": "thong_tin_du_an"
19            }
20        },
21        {
22            # Trai phang mang thong tin du an de de dang truy cap thuoc tinh
23            "$unwind": "$thong_tin_du_an"
24        },
25        {
26            # Nhom du lieu theo ID nhom va dem tong so cong viec trong moi nhom
27            "$group": {
28                "_id": "$thong_tin_du_an.group_id",
29                "so_luong_nhiem_vu": {"$sum": 1}
30            }
31        },
32        {
33            # Sap xep danh sach theo so luong cong viec giam dan
34            "$sort": {"so_luong_nhiem_vu": -1}
35        }
36    ]
37
38    # Thuc thi truy van thong ke va chuyen ket qua thanh danh sach
39    thong_ke_nhom = list(cong_viec_collection.aggregate(pipeline))
40
41    if not thong_ke_nhom:
42        print("Khong co nhieu vu nao duoc tim thay hoac khong lien ket voi nhom.")

```

```

43         return
44
45     # Chuyen doi sang DataFrame de thao tac du lieu de dang hon
46     df_thong_ke_nhom = pd.DataFrame(thong_ke_nhom)
47     df_thong_ke_nhom.rename(columns={'_id': 'id_nhom'}, inplace=True)
48
49     # Xac dinh nhom co nhieu nhien vu nhat
50     nhom_nhieu_nhat = df_thong_ke_nhom.loc[df_thong_ke_nhom['so_luong_nhiem_vu'].idxmax()]
51     id_nhom_nhieu = nhom_nhieu_nhat['id_nhom']
52     dem_nhieu = nhom_nhieu_nhat['so_luong_nhiem_vu']
53     # Lay ten nhom tu bang 'groups'
54     info_nhom_nhieu = nhom_collection.find_one({"_id": id_nhom_nhieu})
55     ten_nhom_nhieu = info_nhom_nhieu.get('TenGr', 'Ten nhom khong xac dinh') if info_nhom_nhieu else 'Khong tim thay nhom'
56
57     # Xac dinh nhom co it nhien vu nhat
58     nhom_it_nhat = df_thong_ke_nhom.loc[df_thong_ke_nhom['so_luong_nhiem_vu'].idxmin()]
59     id_nhom_it = nhom_it_nhat['id_nhom']
60     dem_it = nhom_it_nhat['so_luong_nhiem_vu']
61     # Lay ten nhom tu bang 'groups'
62     info_nhom_it = nhom_collection.find_one({"_id": id_nhom_it})
63     ten_nhom_it = info_nhom_it.get('TenGr', 'Ten nhom khong xac dinh') if info_nhom_it else 'Khong tim thay nhom'
64
65     # In ket qua phan tich cuoi cung
66     print("\n--- Ket qua phan tich --- ")
67     print(f"Nhóm có nhiều nhiệm vụ nhất: '{ten_nhom_nhieu}' (ID: {id_nhom_nhieu}) với {dem_nhieu} nhiệm vụ.")
68     print(f"Nhóm có ít nhiệm vụ nhất: '{ten_nhom_it}' (ID: {id_nhom_it}) với {dem_it} nhiệm vụ.")
69
70     phan_tich_nhiem_vu_nhom()

```

--- So luong nhien vu cho moi nhom ---

--- Ket qua phan tich ---

Nhom co nhieu nhien vu nhat: 'Nhóm 1' (ID: B01) với 19 nhiệm vụ.

Nhom co it nhien vu nhat: 'Nhóm 3' (ID: B03) với 14 nhiệm vụ.

## ✓ Người đc giao công việc nhiều nhất, ít nhất.

```

1 def phan_tich_nhiem_vu_thanh_vien_theo_nhom():
2     print("\n--- Phan tich so luong nhien vu duoc giao cho moi thanh vien trong tung nhom ---")
3
4     # Ket noi cac collections tu co so du lieu
5     nhom_collection = db['groups']
6     du_an_collection = db['projects']
7     cong_viec_collection = db['Tasks']
8     thanh_vien_collection = db['Thanh_vien']
9
10    # Lay danh sach tat ca cac nhom
11    tat_ca_nhom = list(nhom_collection.find({}))
12
13    if not tat_ca_nhom:
14        print("Khong tim thay nhom nao.")
15        return
16
17    for nhom in tat_ca_nhom:
18        id_nhom = nhom.get('_id')
19        ten_nhom = nhom.get('TenGr', 'Khong xac dinh')
20        # Lay tap hop ID cua cac thanh vien thuoc nhom hien tai
21        ds_thanh_vien_nhom = {tv.get('id') for tv in nhom.get('Thanh_vien', [])}
22
23        print(f"\n--- {ten_nhom} (ID: {id_nhom}) ---")
24
25        # 1. Tim cac du an thuoc ve nhom hien tai
26        du_an_cua_nhom = list(du_an_collection.find({"group_id": id_nhom}))
27        if not du_an_cua_nhom:
28            print(" Khong tim thay du an nao cho nhom nay.")
29            continue
30
31        id_cac_du_an = [da.get('_id') for da in du_an_cua_nhom]
32
33        # 2. Tim cac nhien vu thuoc cac du an nay
34        ds_cong_viec_nhom = list(cong_viec_collection.find({"ID_DuAn": {"$in": id_cac_du_an}}))
35
36        if not ds_cong_viec_nhom:
37            print(" Khong tim thay nhien vu nao cho cac du an cua nhom nay.")

```

```

38         continue
39
40     # Khởi tạo bộ đếm nhiệm vụ cho mỗi thành viên trong nhóm bằng 0
41     dem_nhiem_vu_thanh_vien = {id_tv: 0 for id_tv in ds_thanh_vien_nhom}
42
43     # 3. Đếm số nhiệm vụ cho từng thành viên trong nhóm
44     for cong_viec in ds_cong_viec_nhom:
45         chuoai_id_thanh_vien = cong_viec.get('ID_ThanhVien')
46         if chuoai_id_thanh_vien:
47             # Tách chuỗi ID bằng dấu phẩy, loại bỏ khoảng trắng và chỉ giữ lại thành viên thuộc nhóm này
48             thanh_vien_duoc_giao = [m.strip() for m in chuoai_id_thanh_vien.split(',') if m.strip() in ds_thanh_vien_nhom]
49             for id_tv in thanh_vien_duoc_giao:
50                 dem_nhiem_vu_thanh_vien[id_tv] += 1
51
52     # Liệt ra những thành viên có ít nhất 1 nhiệm vụ
53     thanh_vien_co_viec = {id_tv: count for id_tv, count in dem_nhiem_vu_thanh_vien.items() if count > 0}
54
55     if not thanh_vien_co_viec:
56         print(" Không có thành viên nào trong nhóm này được giao nhiệm vụ cụ thể.")
57         continue
58
59     # 4. Xác định thành viên có nhiều nhiệm vụ nhất và ít nhiệm vụ nhất
60     if thanh_vien_co_viec:
61         # Tìm người nhiều việc nhất
62         id_tv_nhieu_nhat = max(thanh_vien_co_viec, key=thanh_vien_co_viec.get)
63         so_luong_nhieu_nhat = thanh_vien_co_viec[id_tv_nhieu_nhat]
64
65         # Bao gồm cả những thành viên có 0 nhiệm vụ để tìm người ít việc nhất
66         tat_ca_thong_ke_nhom = {id_tv: dem_nhiem_vu_thanh_vien.get(id_tv, 0) for id_tv in ds_thanh_vien_nhom}
67
68         id_tv_it_nhat = min(tat_ca_thong_ke_nhom, key=tat_ca_thong_ke_nhom.get)
69         so_luong_it_nhat = tat_ca_thong_ke_nhom[id_tv_it_nhat]
70
71         # Lấy thông tin tên thành viên từ collection Thanh_vien
72         info_tv_nhieu = thanh_vien_collection.find_one({"_id": id_tv_nhieu_nhat})
73         ten_tv_nhieu = info_tv_nhieu.get('HoTen', 'Không xác định') if info_tv_nhieu else 'Không xác định'
74
75         info_tv_it = thanh_vien_collection.find_one({"_id": id_tv_it_nhat})
76         ten_tv_it = info_tv_it.get('HoTen', 'Không xác định') if info_tv_it else 'Không xác định'
77
78         print(f" Thành viên được giao nhiều nhiệm vụ nhất: '{ten_tv_nhieu}' (ID: {id_tv_nhieu_nhat}) với {so_luong_nhiem_vu} nhiệm vụ.")
79         print(f" Thành viên được giao ít nhiệm vụ nhất: '{ten_tv_it}' (ID: {id_tv_it_nhat}) với {so_luong_it_nhat} nhiệm vụ.")
80     else:
81         print(" Không có thành viên nào trong nhóm này được giao nhiệm vụ.")
82
83    phan_tich_nhiem_vu_thanh_vien_theo_nhom()

```

--- Phân tích số lượng nhiệm vụ được giao cho mỗi thành viên trong từng nhóm ---

--- Nhóm 1 (ID: B01) ---

Thành viên được giao nhiều nhiệm vụ nhất: 'Huỳnh Tân Khoa' (ID: A02) với 9 nhiệm vụ.  
 Thành viên được giao ít nhiệm vụ nhất: 'Thạch Minh Nhật' (ID: A04) với 4 nhiệm vụ.

--- Nhóm 2 (ID: B02) ---

Thành viên được giao nhiều nhiệm vụ nhất: 'Trần Hoàng Long' (ID: A06) với 6 nhiệm vụ.  
 Thành viên được giao ít nhiệm vụ nhất: 'Trần Phú Đình' (ID: A08) với 4 nhiệm vụ.

--- Nhóm 3 (ID: B03) ---

Thành viên được giao nhiều nhiệm vụ nhất: 'Đinh Minh Vẹn' (ID: A014) với 6 nhiệm vụ.  
 Thành viên được giao ít nhiệm vụ nhất: 'Huỳnh Thiên Sơn' (ID: A011) với 4 nhiệm vụ.

