

ĐỀ KIỂM TRA THỰC HÀNH

Học phần: Công nghệ Phần mềm - Đề #6

Thời gian làm bài: 60 phút

1. Tình huống

Bạn đang tham gia phát triển backend cho một ứng dụng nghe nhạc trực tuyến. Nhiệm vụ đầu tiên của bạn là xây dựng module quản lý danh sách phát (Playlist) cá nhân. Ứng dụng chạy trên giao diện dòng lệnh (CLI).

Để đảm bảo tính nhất quán và dễ bảo trì, bạn chọn lập trình thủ tục (procedural programming) với Python. Yêu cầu quan trọng nhất là bạn phải tuân thủ quy trình **Git Flow**: Không code dồn cục, mà phải tách từng tính năng ra nhánh riêng, review qua Pull Request rồi mới merge vào nhánh chính.

2. Yêu cầu chung về kỹ thuật

- **Ngôn ngữ:** Python.
- **Công cụ:** Git, GitHub, VS Code.
- **Repository:** Công khai (Public).

3. Nội dung thực hiện

Sinh viên thực hiện tuần tự các bước sau:

Bước 1: Khởi tạo Dự án (Setup)

- Tạo Repository trên GitHub tên: music-playlist-[MSSV] (Ví dụ: music-playlist-B1809999).
- Clone về máy tính.

Bước 2: Khởi tạo khung chương trình (Base Structure)

- Thực hiện trên nhánh main.
- Tạo file playlist.py.
- Khai báo biến danh sách toàn cục songs = [].
- Viết hàm main() hiển thị menu chọn chức năng cơ bản.
- Commit và Push lên nhánh main.

Bước 3: Tính năng "Thêm bài hát" (Feature 1)

- **Quy trình:** Tạo nhánh -> Code -> Push -> PR -> Merge.
- Từ main, tạo nhánh mới: feature/add-song.
- Viết hàm add_song(title, artist, duration): Nhập tên bài hát, tên ca sĩ và thời lượng (giây). Lưu vào danh sách songs dưới dạng Dictionary (ví dụ: {'title': 'Lạc Trôi', 'artist': 'Sơn Tùng MTP', 'duration': 240}).
- Commit và Push nhánh này lên GitHub.
- **Tạo Pull Request vào main, tự Review và Merge.**

Bước 4: Tính năng "Xem danh sách phát" (Feature 2)

- **Lưu ý quan trọng:** Cập nhật code mới nhất từ GitHub về máy trước khi làm tiếp.
- Tại local, chuyển về main, thực hiện git pull để lấy code tính năng 1 về.
- Tạo nhánh mới: feature/view-playlist.
- Viết hàm view_playlist(): Duyệt danh sách và in ra thông tin tất cả bài hát đang có trong playlist.
- Commit và Push nhánh này lên GitHub.
- Tạo Pull Request vào main, tự Review và Merge.

Bước 5: Tính năng "Tìm bài hát theo ca sĩ" (Feature 3)

- **Lưu ý:** Tiếp tục quy trình đồng bộ code.
- Tại local, chuyển về main, thực hiện git pull.
- Tạo nhánh mới: feature/search-artist.
- Viết hàm search_by_artist(artist_name): Cho phép nhập tên ca sĩ. Chương trình sẽ tìm và in ra tất cả các bài hát do ca sĩ đó thể hiện.
- Commit và Push nhánh này lên GitHub.
- Tạo Pull Request vào main, tự Review và Merge.

4. Gợi ý Code (Python - Non OOP)

Sinh viên có thể tham khảo cấu trúc dưới đây:

```
# Biến lưu trữ dữ liệu: Mỗi bài hát là một dict {'title': '...', 'artist': '...', 'duration': 0}
songs = []
```

```
def add_song():
    # Nhập tên bài, ca sĩ, thời lượng -> append vào songs
    print("Đã thêm bài hát vào playlist.")
```

```
def view_playlist():
    # Duyệt list songs và in ra
    # Ví dụ: 1. Lạc Trôi - Sơn Tùng MTP (240s)
    pass
```

```
def search_by_artist():
    # Nhập tên ca sĩ
    # Duyệt list, so sánh artist (có thể dùng in hoặc ==), in ra bài hát
    pass
```

```
def main():
    while True:
        print("\n--- MUSIC PLAYLIST MANAGER ---")
        print("1. Thêm bài hát")
        print("2. Xem danh sách phát")
        print("3. Tìm bài hát theo ca sĩ")
        print("4. Thoát")
```

```
        choice = input("Chọn chức năng: ")
```

```
        if choice == '1':
            add_song()
        elif choice == '2':
            view_playlist()
        elif choice == '3':
            search_by_artist()
        elif choice == '4':
            print("Kết thúc chương trình.")
            break
        else:
            print("Lựa chọn không hợp lệ.")
```

```
if __name__ == "__main__":
    main()
```