|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TRƯỜNG ĐẠI HỌC CÔNG NGHỆ  KHOA CÔNG NGHỆ THÔNG TIN      BÁO CÁO BỘ CÔNG CỤ HỌC KIỂU CHUYỂN ĐỘNG  CỦA CÁC LOẠI HÌNH NGHỆ THUẬT  Lab: HMI   |  |  |  | | --- | --- | --- | | **Nguyễn Quang Minh** | **15021773** | **Lớp: K60-CA** | | **Nguyễn Anh Quang** | **15021800** | **Lớp: K60-CA** |     Hà Nội, 2018 |

Contents

[PHẦN 1. TỔNG QUAN PHẦN MỀM 3](#_Toc513497663)

[1.1. Giới thiệu bài toán 3](#_Toc513497664)

[1.2 Mô tả phần mềm 3](#_Toc513497665)

[1.3 Tham khảo 3](#_Toc513497666)

[1.4 Các định nghĩa và thuật ngữ 4](#_Toc513497667)

[1.5 Chức năng chính 4](#_Toc513497668)

[1.6 Mô tả cấu trúc tập tin đầu vào 5](#_Toc513497669)

[1.7 Quy trình cơ bản 8](#_Toc513497670)

[1.8 Usecase tổng quan của phần mềm 10](#_Toc513497671)

[1.9 Kịch bản sử dụng phần mềm 13](#_Toc513497672)

[PHẦN 2. KIẾN TRÚC TỔNG THỂ CỦA HỆ THỐNG 14](#_Toc513497673)

[2.1. Sơ đồ kiến trúc tổng thể 14](#_Toc513497674)

[2.2. Mô hình lập trình 14](#_Toc513497675)

[2.3 Công nghệ sử dụng 15](#_Toc513497676)

[PHẦN 3. THIẾT KẾ HỆ THỐNG 16](#_Toc513497677)

[PHẦN 4. DEMO 19](#_Toc513497678)

# PHẦN 1. TỔNG QUAN PHẦN MỀM

## Giới thiệu bài toán

Hiện nay trên đất nước Việt Nam có vô vàn điệu múa truyền thống khác nhau.

Cùng với đó là sự hội nhập liên tục của xã hội quốc tế đã khiến cho giới trẻ ngày nay ngày càng xa vời hơn với những nét đẹp truyền thống văn hóa Việt.

Vì vậy bộ công cụ “học kiểu chuyển động của các loại hình nghệ thuật” được phát triển với mong muốn sẽ phần nào hỗ trợ phát huy và lưu giữ các điệu múa truyền thống Việt Nam.

## 1.2 Mô tả phần mềm

Phần mềm là một công cụ hỗ trợ người dùng thông qua giao diện bắt mắt và tính năng dễ sử dụng, với đầu vào là hai file chứa dữ liệu về chuyển động thông qua các thuật toán sử lý, học chuyển động sẽ cho đầu ra là một file dữ liệu truyển động được học từ hai chuyển động ban đầu.

## 1.3 Tham khảo

Tài liệu hướng dẫn nghiên cứu:

* Bộ bài giảng Môn PTTK Hướng Đối Tượng
* Head First Object Oriented Design and Analysis
* CS0111: Motion Capture File Formats Explained

## 1.4 Các định nghĩa và thuật ngữ

* Ngôn ngữ UML: Thiết lập một phương thức thống nhất để xây dựng và “vẽ” ra các yêu cầu & thiết kế theo hướng đối tượng trong quá trình phân tích và thiết kế phần mềm, có thể được áp dụng để phát triển nhiều Phần mềm khác nhau, và được sử dụng với phần lớn các ngôn ngữ lập trình phổ biến như : Java, .NET,…
  + Hệ thống (System): là một hệ thống thực tế cần phải mô tả hoặc xây dựng.
  + Mô hình (Model): là kết quả của quá trình trừu tượng hóa của hệ thống thực.
  + Khung nhìn (View): phản ánh một khía cạnh nào đó của mô hình. Mỗi một Model có thể có nhiều View khác nhau tùy thuộc vào mục đích của View đó.
  + Ký hiệu (Notation): là một ký hiệu đồ họa hoặc văn bản cho phép mô tả khung nhìn.
* Mô hình UseCase (Use Case Diagram): Mô hình hóa các chức năng mà Hệ thống sẽ thực thi
  + Actor (tác nhân): Thể hiện “một cái gì đó” tương tác với Hệ thống
  + Use Case: Mô tả chức năng mà Hệ thống có
  + Relationship (các quan hệ): Quan hệ giao tiếp, quan hệ bao gồm, quan hệ mở rộng, quan hệ tổng quát hóa
  + Package (tạo các gói): Nhóm các thành phần tác nhân, Use Caset và các quan hệ vào thành một nhóm chung, nhóm này là một quy trình (module) tương đối hoàn thiện

## 1.5 Chức năng chính

+ Chức năng học chuyển động:

* + - Nhận đầu vào là hai file với format .bvh.
    - Đầu ra là một file với format .bvh là kết quả học chuyển động từ hai file đầu vào.

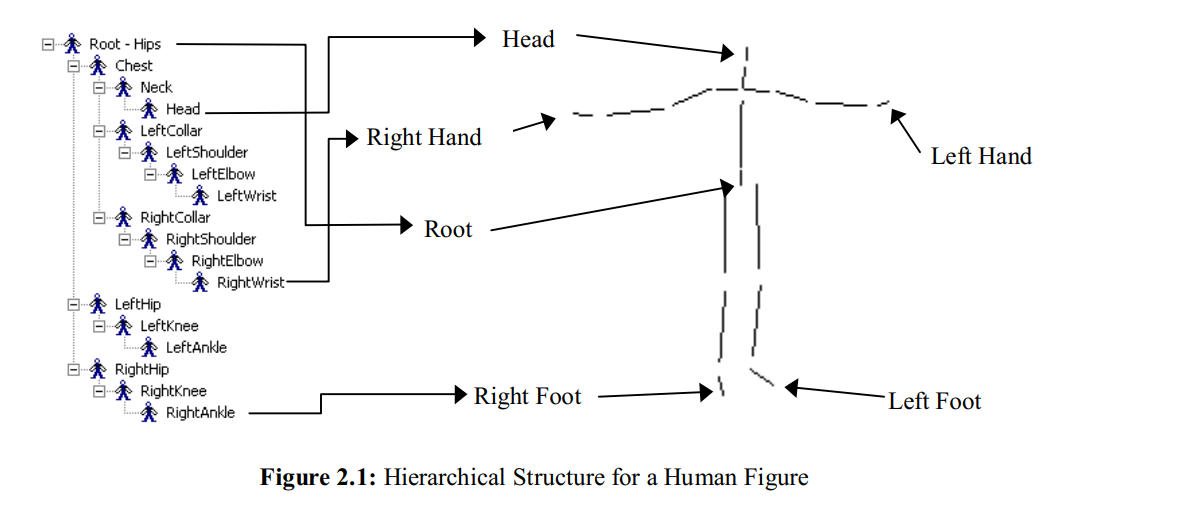
+ Chức năng hiển thị chuyển động:

* + - Nhận đầu vào là một file với format .bvh.
    - Hiện thị ra màn hình video chuyển động được lưu trữ với format .mp4.

## 1.6 Mô tả cấu trúc tập tin đầu vào

**BVH format:**

Cấu trúc phân tầng:



Gồm 2 phần:

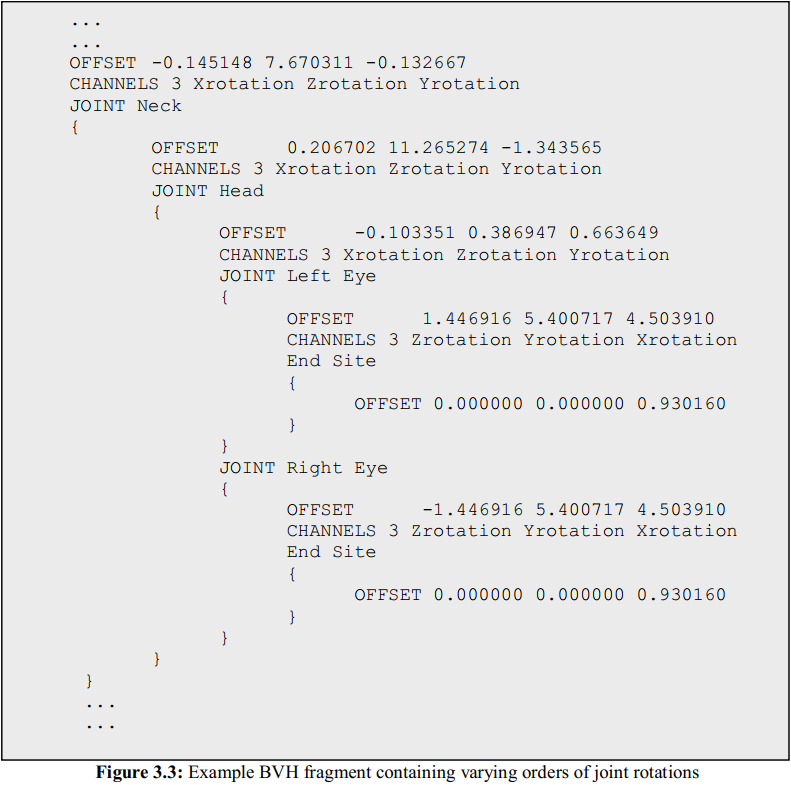
* Phần cấu trúc (hierarchy section): chi tiết về cấu trúc tầng và pose của skeleton
* Phần chuyển động (motion section): chứa dữ liệu chuyển động của skeleton, với mỗi frame là các thông số cho từng channel

*Phần cấu trúc*:

Bắt đầu bằng từ khóa HIERARCHY, sau đó là ROOT đi cùng với tên của bộ phần được chọn làm root của của bộ xương. Sau đó là các bộ phận khác được sắp xếp theo cấu trúc đệ quy, khời đầu bằng từ khóa JOINT đi cùng với tên bộ phận.

Mỗi bộ phận có OFFSET chứa thông tin dịch chuyển tọa độ các trục x y z so với trục tọa độ của lớp cha (trong trường hợp ROOT thì chính là so với gốc tọa độ). Ngoài ra OFFSET còn giúp tính hướng và độ dài của xương lớp cha, dùng offset của con so với offset cha để suy ra độ dài xương.

Tiếp đến CHANNEL là thông tin định nghĩa về cử động của xương. Đầu tiên là số channel tiếp đó lần lượt là tên của các channel theo thứ tự. Chú ý thứ tự này có ý nghĩa trong motion section của file. Cụ thể là: trong motion section sẽ chứa lần lượt thông số các channel của ROOT theo thứ tự được mô tả ở hierarchy section, tiếp đến là thông số các channel của JOIN con, …. Và cứ lần lượt như vậy cho đến hết đệ quy. Một điểm nữa lần lưu ý là thứ tự nối góc Euler khi tạo rotation matrix của xương cần phải làm theo thứ tự của channel, và vì thứ tự Euler này là khác nhau đối với mỗi xương nên thứ tự của channel cũng khác nhau với mỗi xương.



Cuối cùng khớp kết thúc có ký hiệu là End Site, trong đó chứa thông tin OFFSET (để tính độ dài và hướng của xương)

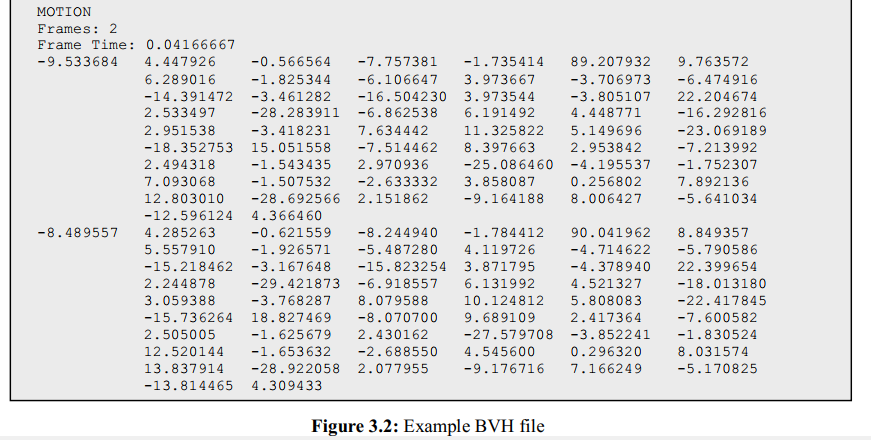
*Phần chuyển động*:

    Bắt đầu bằng từ khóa MOTION, gồm số frames, frames rate và channel data (thông số của channel)

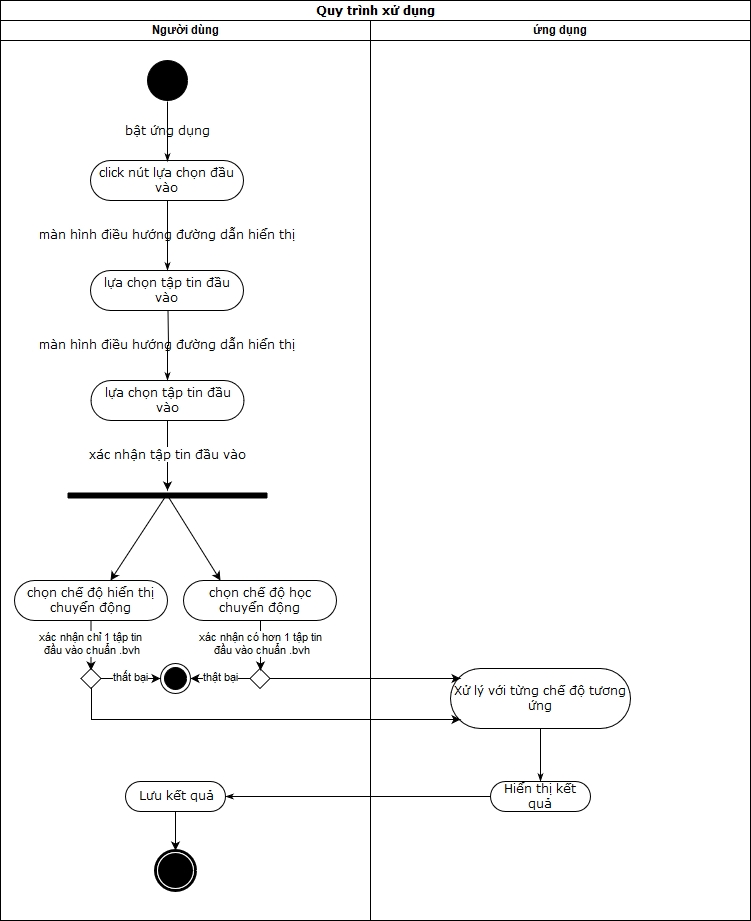
    Từ khóa “Frames: “ tiếp đến là 1  số chỉ số frames

    Từ khóa “Frames Time: “ sẽ cho thời lượng của 1 frames. Để chuyển thành frames per second thì chỉ cần lấy 1 chia cho frames time.

    Phần còn lại của file chứa channel data của từng xương theo thứ tự được định nghĩa ở hierarchy section. Mỗi dòng tương ứng 1 frame.



## 1.7 Quy trình cơ bản



* Các ký hiệu trong sơ đồ

|  |  |
| --- | --- |
| **Ký hiệu** | **Giải thích** |
|  | Nút Start thể hiện điểm bắt đầu quy trình |
|  | Activity mô tả một hoạt động trong hệ thống. Các hoạt động này do các đối tượng thực hiện. |
|  | Branch thể hiện rẽ nhánh trong mệnh đề điều kiện. |
|  | Dòng điều khiển |
|  | End thể hiện điểm kết thúc qui trình. |
|  | Fork thể hiện cho trường hợp thực hiện xong một hoạt động rồi sẽ rẽ nhánh thực hiện nhiều hoạt động tiếp theo |

## 1.8 Usecase tổng quan của phần mềm

*Mô tả các tác nhân trong Usecase:*

+ Tác nhân Người dùng: người sử dụng phần mềm, lựa chọn và đưa vào các file input.

+ Tác nhân Phần mềm học chuyển động: đóng vai trò sau khi tiếp nhận hai tập tin đúng yêu cầu, sẽ xử lý và xuất ra tập tin là mô hình chuyển động sau khi đã học được từ hai tập ban đầu.

+ Tác nhân Phần mềm hiển thị chuyển động: đóng vai trò tiếp nhận vào tập tin chứa thông tin chuyển động hiện thị chúng dưới dạng một video cho người dùng và lưu lại.

*Mô tả các Usecase trong phần mềm:*

UC#01: Lựa chọn dầu vào

Usecase này cho phép người dùng tìm kiếm và lựa chọn các tập tin đầu vào cho phần mềm.

*Luồng sự kiện*:

1. Người dùng ấn vào nút “tìm kiếm”.
2. Người dùng chọn đường dẫn vào thư mục mong muốn.
3. Người dùng chọn các tập tin đầu vào mong muốn.

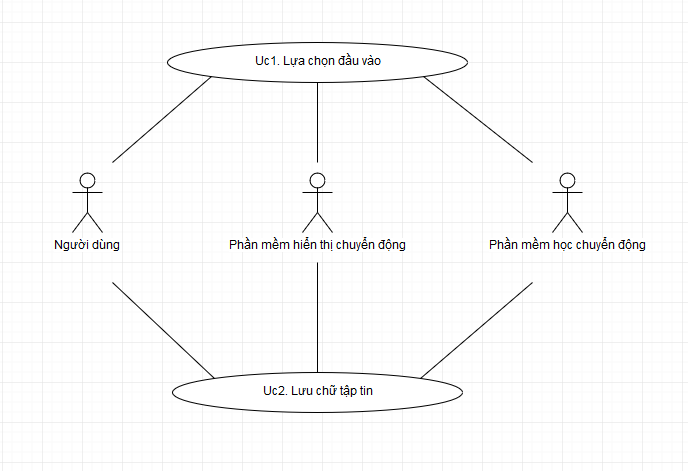
UC#02: Lưu trữ tập tin

Usecase này cho phép người dùng đặt tên và lựa chọn vị trí lưu trữ cho tập tin đầu ra của phần mềm.

*Luồng sự kiện*:

1. Sau khi hệ thống sử lý xong, người dùng ấn vào nút “lưu trữ”.
2. Người dùng chọn đường dẫn lưu trữ và đặt tên cho cho tập tin đầu ra của hệ thống,

**Mô hình Usecase:**



* Các kí hiệu trong sơ đồ

|  |  |
| --- | --- |
| **Ký hiệu** | **Giải thích** |
|  | Actor được dùng để chỉ người sử dụng hoặc một đối tượng nào đó bên ngoài tương tác với hệ thống chúng ta đang xem xét. |
|  | Use Case là chức năng mà các Actor sẽ sử dụng. Nó được ký hiệu như sau |
|  | Association thường được dùng để mô tả mối quan hệ giữa Actor và Use Case và giữa các Use Case với nhau |

## 1.9 Kịch bản sử dụng phần mềm

(quy trình sử dụng tính năng học chuyển động)

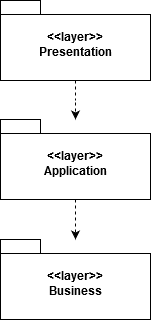
* + Bước 1: Người dùng khời động phần mềm.
  + Bước 2: Người dùng ấn nút và lựa chọn các tệp tin đầu vào.
  + Bước 3: Người dùng chọn chế độ học chuyển động.
  + Bước 4: Học chuyển động thành công, người dùng lựa chọn tên và nơi lưu trữ tập tin chuyển động mới do phần mềm xuất ra.

(quy trình sử dụng tính năng hiện thị chuyển động)

* + Bước 1: Người dùng khời động phần mềm.
  + Bước 2: Người dùng ấn nút và lựa chọn các tệp tin đầu vào.
  + Bước 3: Người dùng chọn chế độ hiển thị chuyển động.
  + Bước 4: Video hiển thị chuyển động xuất hiện trên màn hình, người dùng xem video.
  + Bước 5: Người dùng chọn lưu video, chọn tên và nơi lưu trữ video chuyển động.

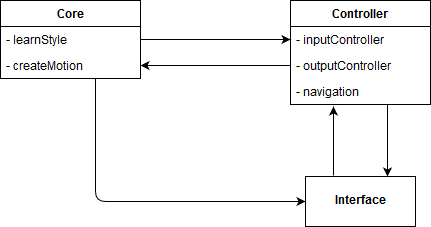
# PHẦN 2. KIẾN TRÚC TỔNG THỂ CỦA HỆ THỐNG

## Sơ đồ kiến trúc tổng thể



Phần mềm được thiết kế theo mô hình layer. Mô hình layer là mô hình phân ra nhiều tầng, mỗi tầng sẽ có nhiệm vụ/ công việc riêng nhưng tương tác chặt chẽ với nhau.

## Mô hình lập trình



**Core:**

Core chứa công nghệ lỗi thực thi xử lý học chuyển động vào tạo chuyển động mới.

**Controller:**

Controller xử lý công việc điều hướng, tiếp nhận input và đưa ra output.

**Interface:**

Interface gồm icon, nút bấm, hiển thị giao diện người dùng.

## 2.3 Công nghệ sử dụng

**Python:**

*Python* là ngôn ngữ lập trình hướng đối tượng, cấp cao, mạnh mẽ, được tạo ra bởi Guido van Rossum. Nó dễ dàng để tìm hiểu và đang nổi lên như một trong những ngôn ngữ lập trình nhập môn tốt nhất cho người lần đầu tiếp xúc với ngôn ngữ lập trình. Python hoàn toàn tạo kiểu động và sử dụng cơ chế cấp phát bộ nhớ tự động. Python có cấu trúc dữ liệu cấp cao mạnh mẽ và cách tiếp cận đơn giản nhưng hiệu quả đối với lập trình hướng đối tượng. Cú pháp lệnh của Python là điểm cộng vô cùng lớn vì sự rõ ràng, dễ hiểu và cách gõ linh động làm cho nó nhanh chóng trở thành một ngôn ngữ lý tưởng để viết script và phát triển ứng dụng trong nhiều lĩnh vực, ở hầu hết các nền tảng.

**TkInter:**

*Tkinter*là một gói trong Python có chứa module *Tk* hỗ trợ cho việc lập trình GUI*. Tk* ban đầu được viết cho ngôn ngữ **Tcl***.* Sau đó Tkinter được viết ra để sử dụng Tk bằng trình thông dịch Tcl trên nền Python. Ngoài Tkinter ra còn có một số công cụ khác giúp tạo một ứng dụng GUI viết bằng Python như wxPython, PyQt, và PyGTK.

**OpenCV:**

*OpenCV* (Open Computer Vision) là một thư viện mã nguồn mở hàng đầu cho thị giác máy tính (Computer Vision), xử lý ảnh (Image Processing) và máy học (Machine Learnning). Nó chứa hàng ngàn thuật toán tối ưu hoá , trong đó cung cấp một bộ công cụ phổ biến cho các ứng dụng về thị giác máy tính. OpenCV đang được sử dụng trong rất nhiều ứng dụng , từ khâu hình ảnh Street View của Google tới việc chạy các chương trình nghệ thuật tương tác , nhận diện khuôn mặt , hay Robot , xe hơi tự lái ..

# PHẦN 3. THIẾT KẾ HỆ THỐNG

*Thiết kế usecase 01*:

|  |  |  |
| --- | --- | --- |
| **UC#01** | | **LỰA CHỌN ĐẦU VÀO** |
| **Mô tả** | | Chức năng này cho phép người dùng lựa chọn đường dẫn, tìm kiếm đầu vào cho bộ công cụ |
| **Tác nhân** | **Chính** | Người dùng |
| **Phụ** | Không |
| **Tiền điều kiện** | | Sau khi người dùng bật ứng dụng |
| **Hậu điều kiện** | **Thành công** | Các tệp đầu vào đã được xác nhận. |
| **Lỗi** | Trạng thái hệ thống sẽ không thay đổi. |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| Chức năng này bắt đầu khi người dùng muộn chọn tệp tin đầu vào để dùng các chức năng khác trong ứng dụng  Sau khi người dùng chọn nút “browse”, hệ thống hiển thị danh sách thư mục và đường dẫn.  Người dùng có thể chọn một hoặc hai tệp tin đầu vào.  Sau khi xác nhận lựa chọn, người dùng sẽ có thể lựa chọn giữa hai chức năng của ứng dụng: “Học chuyển động” và “Hiển thị chuyển động”. | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | |
| **Chưa chọn tệp tin**  Khi người dùng chưa chọn tệp tin cần thiết.  Trạng thái hệ thống sẽ không đổi. | | |
| **THIẾT KẾ UML** | | |
| **Sơ đồ Lớp phân tích (Analysis class diagram)** | | |
| C:\Users\Admin\Downloads\HMI5.png | | |
| **Sơ đồ Trình tự (Sequence diagram)** | | |
| C:\Users\Admin\Downloads\HMI3.png | | |

*Thiết kế usecase 02*:

|  |  |  |
| --- | --- | --- |
| **UC#02** | | **LƯU TRỮ TẬP TIN** |
| **Mô tả** | | Chức năng này cho phép người dùng lưu chữ tập tin kết quả của phần mềm |
| **Tác nhân** | **Chính** | Người dùng |
| **Phụ** | Phần mềm học chuyển động, Phần mềm hiển thị chuyển động |
| **Tiền điều kiện** | | Sau khi người dùng đã xử lý dữ liệu dưới một trong hai chế độ “học chuyển động” và “hiển thị chuyển động” |
| **Hậu điều kiện** | **Thành công** | Tệp tin đã được lưu |
| **Lỗi** | Trạng thái hệ thống sẽ không thay đổi. |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| Chức năng này bắt đầu khi “Core” đã xử lý xong tệp tin theo yêu cầu trước đó của người dùng và giờ người dùng có thể lưu kết quả lại  Sau khi người dùng chọn nút “Save”, hệ thống hiển thị danh sách thư mục và đường dẫn.  Người dùng có thể đặt tên cho thư mục đầu ra.  Sau khi xác nhận lưu chữ, tệp tin sẽ xuất hiện ở thư mục mong muốn. | | |
| **Luồng sự kiện phát sinh/Kịch bản phát sinh** | | |
| **Chưa đặt tên tập tin**  Khi người dùng chưa đặt tên cho tập tin đầu ra.  Trạng thái hệ thống sẽ không đổi. | | |
| **THIẾT KẾ UML** | | |
| **Sơ đồ Lớp phân tích (Analysis class diagram)** | | |
| C:\Users\Admin\Downloads\HMI5(1).png | | |
| **Sơ đồ Trình tự (Sequence diagram)** | | |
| C:\Users\Admin\Downloads\HMI3(1).png | | |

# PHẦN 4. DEMO