# **GIỚI THIỆU**

Với sự phát triển nhanh chóng của các thiết bị công nghệ nói chung, và các thiết bị di động nói riêng, với những tính năng liên tục được đổi mới, khoa học và công nghệ giúp cho những chiếc điện thoại di động từ những chức năng đơn thuần là gọi điện - nhắn tin, ngày nay chúng ta có thể làm được bất kì công việc gì liên quan đến công nghệ thông qua chiếc điện thoại.

Điện thoại giúp ích cho chúng ta trong rất nhiều hoạt động như liên lạc, xem tin tức, mua sắm, đặt vé, … Ngoài ra, chúng còn giúp ta giải trí như nghe nhạc, xem phim, chat, lướt facebook,…

Trong lĩnh vực giải trí, cụ thể là nghe nhạc trên nền tảng Windows Phone, đã có rất nhiều ứng dụng từ các nhà sản xuất lớn nhỏ trong và ngoài nước đưa ứng dụng của họ lên Store cho người dùng tải về và sử dụng. Những ứng dụng của họ có các tính năng và giao diện khác nhau, tuy nhiên, qua tìm hiểu của nhóm, Store Windows Phone chưa có hoặc còn ít ứng dụng giúp chúng ta điều khiển các chức năng nghe nhạc bằng giọng nói.

Vì vậy, nhóm quyết định viết một ứng dụng trên Windows Phone nhằm giải quyết vấn đề này.

* Đề tài:Ứng dụng phát nhạc sử dụng nhận diện giọng nói.
* Mục tiêu:
* Các chức năng chính:
  + Hiển thị thông tin bài hát (Title, Artist, Album).
  + Hiển thị thời lượng (Progress Bar, thời gian hiện tại, thời lượng của bài hát).
  + Các chức năng nghe nhạc cơ bản (Play/Pause, Stop, Next, Previous).
  + Chế độ phát nhạc: Playback (Order, Random), Repeat (One, All, No).
  + Hiển thị danh sách bài hát theo các chế độ (Title, Artist, Album).
  + Phát nhạc theo Playlist.
  + Chạy nhạc trên nền (khi chuyển sang màn hình danh sách bài hát, khi vào ứng dụng khác (tin nhắn, lịch, thời tiết, ghi chú,…) hoặc khi tắt màn hình).
  + **Sử dụng giọng nói để điều khiển các chức năng nghe nhạc.**

# **PHÂN TÍCH**

//Hoàn thành – Có thể bổ xung thêm chức năng ở nhận diện giọng nói (Mở màn hình list, option<Có thể cài đặt bằng giọng nói, hơi thốn =)) >)

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên yêu cầu** | **Mô tả chi tiết** |
| 1 | Hiển thị thông tin | Cho người dùng nhìn thấy được tên bài hát, ca sĩ, album, thời gian hiện tại, thời lượng tối đa, tiến trình bài hát. |
| 2 | Phát nhạc | Các phím chức năng giúp người dùng thực hiện các thao tác liên quan đến quá trình phát nhạc (Play/Pause, Stop, Next, Previous), phát nhạc khi chọn vào một bài hát, hoặc phát các bài hát trong một playlist khi người dùng bấm vào. |
| 3 | Chạy nền | Ứng dụng phát nhạc ngay cả khi không truy cập ứng dụng, hoặc truy cập ứng dụng khác hay tắt màn hình. |
| 4 | Nhận diện giọng nói | Sử dụng giọng nói để điều khiển các chức năng khi nghe nhạc (Play/Pause, Stop, Next, Previous). |

# **THIẾT KẾ**

* Kiến trúc chương trình, các lớp, đối tượng.

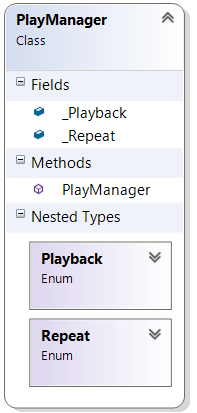
|  |  |  |
| --- | --- | --- |
| **STT** | **Tên lớp (Class)** | **Mô tả** |
| 1 | PlayManager | Quản lý chế độ phát, lặp bài hát. |
| 2 | SettingManager | Quản lý cài đặt chương trình: hình nền, màu chữ và cả PlayManger. |
| 3 | MusicManager | Quản lý các chức năng liên quan đến quá trình phát nhạc (Play/Pause/Stop/Next/Previous), lấy thông tin bài hát (Title, Album, Artist, Duration,…). |
| 4 | AlphaKeyGroup<T> | Dùng để gom nhóm danh sách các bài hát, album, artist theo chữ cái bắt đầu, sắp xếp theo thứ tự sắp xếp quy định trong SortedLocaleGrouping |
| 5 | ArrSong | Chứa thông tin tên bài hát, album, artist, nhằm mục đích tạo List trung gian để hiển thị danh sách bài hát, album, artist. |
| 6 | ArrSongGrouped | Chứa thông tin tên bài hát, album, artist, nhằm mục đích tạo List trung gian để hiển thị danh sách bài hát nhóm theo album, bài hát. |
| 7 | AudioContainer | Là Class của CMU Sphinx để ghi âm và nhận diện giọng nói. |
| 8 | AudioDataEventArgs | Là Class của CMU Sphinx đưa dữ liệu về Byte. |
| 9 | WasapiAudioRecorder | Là Class của CMU Sphinxđể quản lý việc thu âm và đưa vào bộ nhớ đệm cho quá trình xử lý. |
| 10 | ListSongPage (xaml.cs) | Quản lí danh sách bài hát, album, artist |
| 11 | PlayListPage (xaml.cs) | Danh sách bài hát theo album, theo artist |
| 12 | MainPage (xaml.cs) | Màn hình chính giúp người dùng thao tác đến quá trình nghe nhạc (sử dụng các button, hiển thị thông tin cho người dùng,…). |
| 13 | SettingPage (xaml.cs) | Thao tác đến quá trình cài đặt gồm hiển thị thông tin cài đặt, tuỳ chọn cài đặt và lưu cài đặt. |

Class PlayManager:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_Playback | Thuộc tính | enum | public | Chế độ phát bài hát tiếp theo, giá trị nằm trong enum Playback{ORDER, RANDOM} |
| 2 | \_Repeat | enum | public | Chế độ lặp bài hát, giá trị nằm trong enum Repeat {ALL, NO, ONE} |
| 3 | PlayManager() | Phương thức |  | public | Phương thức khởi tạo. |

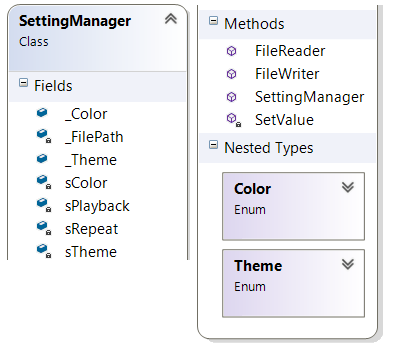
MẪU CHI TIẾT CLASS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
|  |  |  |  |  |  |



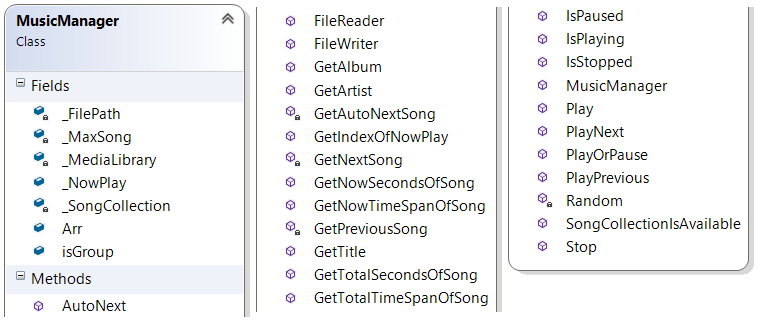
Class SettingManager:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_FilePath | Thuộc tính | string | private | Đường dẫn đến file chứa các thông tin cài đặt. |
| 2 | \_Color | enum | public | Màu chữ hiển thị ở MainPage, giá trị nằm trong enum Color {BLUE, RED, YELLOW} |
| 3 | \_Theme | enum | public | Background hiển thị ở MainPage, giá trị nằm trong enum Theme {WINTER, SPRING} |
| 4 | sColor | string | private | Chứa giá trị màu sắc. |
| 5 | sPlayback | string | private | Chứa giá trị chế độ phát. |
| 6 | sRepeat | string | private | Chứa giá trị chế độ lặp bài hát. |
| 7 | sTheme | string | private | Chứa giá trị nền màn hình chính. |
| 8 | SettingManager() | Phương thức |  | public | Phương thức khởi tạo. |
| 9 | FileReader() | void | public | Đọc từ file lưu giá trị vào các biến sColor, sPlayback, sRepeat, sTheme. |
| 10 | FileWriter() | void | public | Lưu giá trị được chọn ở màn hình SettingPage vào file. |
| 11 | SetValue() | void | private | Thiết lập các giá trị Color, Playback, Repeat, Theme. |



Class MusicManager:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_FilePath | Thuộc tính | string | private | Đường dẫn đến file chứa thông tin bài hát. |
| 2 | \_MaxSong | int | private | Đếm số lượng bài hát của thiết bị. |
| 3 | \_MediaLibrary | MediaLibrary | private | Truy cập thư viện Media của thiết bị. |
| 4 | \_NowPlay | int | public | Chứa chỉ số (theo thứ tự) của bài hát hiện đang phát (hoặc chưa được phát hay đã dừng). |
| 5 | \_SongCollection | SongCollection | private | Truy cập đến thư viện nhạc trong thiết bị. |
| 6 | Arr | int[] | public | Mảng chứa index bài hát gom nhóm theo Album hoặc Artist |
| 7 | isGroup | bool | public | Kiểm tra chế độ phát nhạc danh sách toàn bộ bài hát hay không, dùng để phát danh sách. nhạc theo Album, hoặc Artist |
| 8 | AutoNext() | Phương thức | void | public | Phát bài hát tiếp theo. |
| 9 | FileReader() | void | public | Đọc từ file giá trị của bài hát hiện tại. |
| 10 | FileWriter() | void | public | Lưu vào file giá trị của bài hát hiện tại. |
| 11 | GetAlbum() | string | public | Lấy giá trị tên Album. |
| 12 | GetArtist() | string | public | Lấy giá trị tên Artist. |
| 13 | GetAutoNextSong() | int | private | Lấy chỉ số bài hát tiếp theo khi phát tự động (khi hết bài tự chuyển sang bài khác). |
| 14 | GetIndexOfNowPlay() | int | public | Lấy chỉ số bài hát hiện tại. |
| 15 | GetNextSong() | int | private | Lấy chỉ số bài hát tiếp theo. |
| 16 | GetNowSecondsOfSong() | double | public | Lấy số giây hiện tại của MediaPlayer. |
| 17 | GetNowTimeSpanOfSong() | TimeSpan | public | Lấy TimeSpan hiện tại của MediaPlayer. |
| 18 | GetPreviousSong() | int | private | Lấy chỉ số bài hát trước đó. |
| 19 | GetTitle() | string | public | Lấy tiêu đề bài hát. |
| 20 | GetTotalSecondsOfSong() | double | public | Lấy số giây tổng thời gian bài hát. |
| 21 | GetTotalTimeSpanOfSong() | TimeSpan | public | Lấy TimeSpan tổng thời gian bài hát. |
| 22 | IsPaused() | bool | public | Kiểm tra MediaPlayer có đang tạm dừng không. |
| 23 | IsPlaying() | bool | public | Kiểm tra MediaPlayer có đang phát nhạc không. |
| 24 | IsStopped() | bool | public | Kiểm tra MediaPlayer có đang được dừng không. |
| 25 | MusicManager() |  | public | Phương thức khởi tạo. |
| 26 | Play(int Number) | void | public | Phát bài hát thứ Number trong SongCollection. |
| 27 | PlayNext() | void | public | Phát bài hát tiếp theo. |
| 28 | PlayOrPause() | void | public | Đặt trạng thái phát nhạc tương ứng với mỗi State của MediaState. |
| 29 | PlayPrevious() | void | public | Phát bài hát trước đó. |
| 30 | Random(int Max) | int | private | Trả về giá trị ngẫu nhiên từ 0 đến Max. |
| 31 | SongCollectionIsAvailable() | bool | public | Kiểm tra SongCollection có khả dụng hay không. |
| 32 | Stop() | void | public | Dừng phát nhạc. |



Class MainPage (xaml.cs)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_MusicManager |  |  |  |  |
| 2 | \_SettingManager |  |  |  |  |
| 3 | playTimer |  |  |  |  |
| 4 | ArrAlbum |  | String |  | Dùng để nhận chuỗi chứa index danh sách bài hát phát theo Album từ ListSongPage hoặc PlayListPage khi có sự kiên Navigate. |
| 5 | ArrArtist |  | String |  | Dùng để nhận chuỗi chứa index danh sách bài hát phát theo Artist từ ListSongPage hoặc PlayListPage khi có sự kiên Navigate. |
| 6 | ArrAlbumIndex |  | String[] |  | Chuỗi tạo thành sau khi cắt khoảng trắng từ ArrAlbum |
| 7 | ArrArtistIndex |  | String[] |  | Chuỗi tạo thành sau khi cắt khoảng trắng từ ArrArtist |
| 8 | AlbumIndex |  | int[] |  | Mảng chứa index danh sách bài hát theo Album được chọn để phát sau khi được mã hóa từ ArrAlbumIndex |
| 9 | ArtistIndex |  | int[] |  | Mảng chứa index danh sách bài hát theo Artist được chọn để phát sau khi được mã hóa từ ArrAlbumIndex |
| 10 | indexNavigate |  | String |  | Nhận index bài hát đang được phát sau khi chọn phát theo Album hoặc Artist. |
| 11 | WakeupText |  |  |  | Chuỗi tín hiệu cho phép bắt đầu theo tác bằng giọng nói. |
| 12 | SpeechValues |  |  |  |  |
| 13 | \_mode |  |  |  |  |
| 14 | speechRecognizer |  |  |  |  |
| 15 | audioRecorder |  |  |  |  |

Class SettingPage (xaml.cs)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_ImagePath | Thuộc tính | string | private | Chuỗi chỉ dẫn đến file setup.txt để thực hiện việc lưu và đọc cài đặt. |
| 2 | \_SettingManager | SettingManager | private |  |
| 3 | sColor | string | private | Chuỗi chứa giá trị Color. |
| 4 | sPlayback | string | private | Chuỗi chứa giá trị Playback. |
| 5 | sRepeat | string | private | Chuỗi chứa giá trị Repeat. |
| 6 | sTheme | string | private | Chuỗi chứa giá trị Theme. |
| 7 | SettingPage() | Phương thức |  | public | Phương thức khởi tạo. |
| 8 | LoadImage() | void | private | Đưa hình mẫu của Theme được lưu từ trước đó. |
| 9 | SetDefaultRadioButton() | void | private | Đặt các giá trị mặc định cho các RadioButton đúng với các giá trị đã lưu từ trước. |
| 10 | rbtn\_theme\_winter\_Checked() | void | private | Thay đổi hình mẫu của Theme sang winter theme. |
| 11 | rbtn\_theme\_spring\_Checked() | void | private | Thay đổi hình mẫu của Theme sang spring theme. |
| 12 | btn\_save\_Click() | void | private | Xử lý khi bấm nút lưu. |
| 13 | btn\_cancel\_Click() | void | private | Xử lý khi bấm nút huỷ. |

Class ListSongPage(xaml.cs)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_MediaLibrary | Thuộc tính | MediaLibrary | Private | Truy cập thư viện Media của thiết bị. |
| 2 | \_Song | SongCollection | Private | Truy cập đến thư viện nhạc trong thiết bị. |
| 3 | \_SourceSong | ObservableCollection<ArrSong> | Private | Lưu danh sách Songs của \_Song vào một ObservableCollection<> |
| 4 | \_SourceArtist | ObservableCollection<ArrSong> | Private | Copy từ \_SourceSong để thao tác với danh sách Artist |
| 5 | ArrAlbum | String | Private | Lưu chuỗi index danh sách bài hát phát theo Album |
| 6 | ArrArtist | String | Private | Lưu chuỗi index danh sách bài hát phát theo Artist |
| 7 | ListSongPage() | Phương thức |  | Private | Contructor mặc định |
| 8 | GroupSong() | void | Private | Nhóm danh sách bài hát |
| 9 | GroupAlbum() | void | Private | Nhóm danh sách Album |
| 10 | GroupArtist() | void | Private | Nhóm danh sách Artist |
| 11 | Song\_Tapped | Void | Private | Sự kiện tap TextBlock Song |
| 12 | Artist\_Tapped() | void | Private | Sự kiện tap StackPanel Album |
| 13 | Album\_Tapped() | Void | Private | Sự kiện tap StackPanel Artist |

Class ArrSong

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | Song | Thuộc tính | string | Private | Get, set tên bài hát |
| 2 | Album | string | Private | Get, set tên Album |
| 3 | Artist | string | Private | Get, set tên Artist |
| 4 | ArrSong(string, string, string) | Phương thức |  | Public | Contructor có tham số |

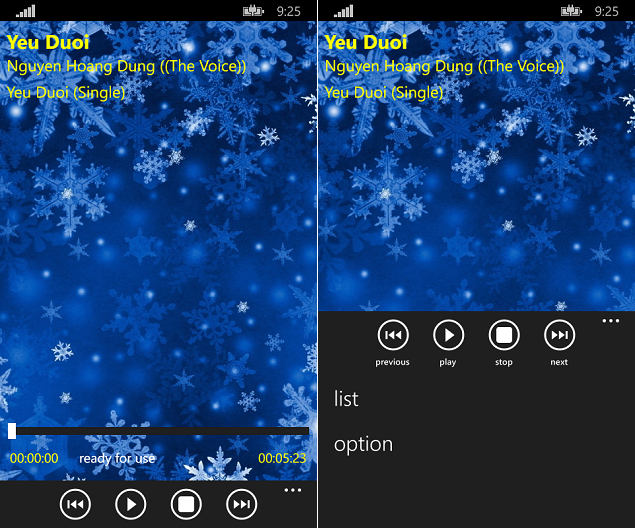
Class ArrSongGrouped

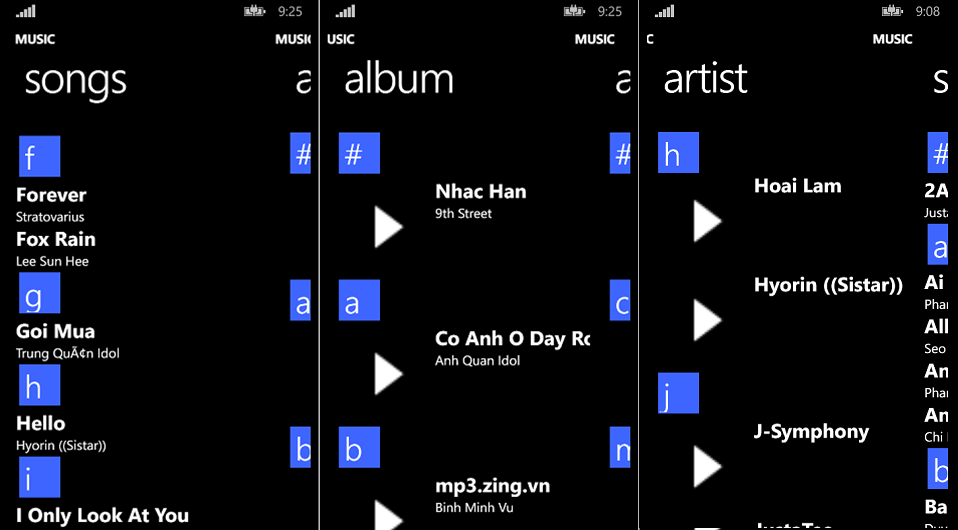
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | Song | Thuộc tính | string | Private | Get, set tên bài hát |
| 2 | Album | string | Private | Get, set tên Album |
| 3 | Artist | string | Private | Get, set tên Artist |
| 4 | ArrSongGrouped(string, string, string) | Phương thức |  | Public | Contructor có tham số |

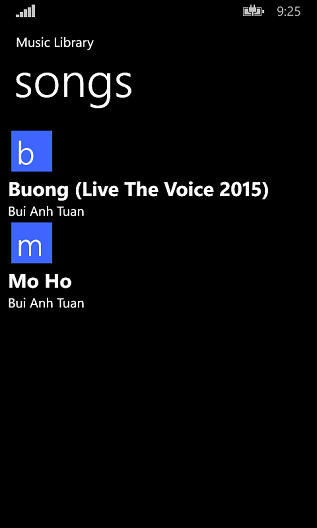
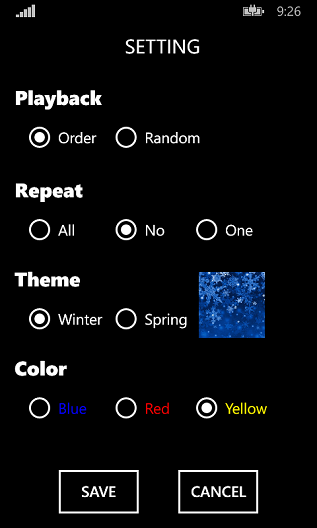
Class PlayListPage (xaml.cs)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính / phương thức | Loại | Kiểu dữ liệu / trả về | Phạm vi truy xuất | Chức năng |
| 1 | \_MediaLibrary | Thuộc tính | MediaLibrary | Private | Truy cập thư viện Media của thiết bị. |
|  | \_Song | SongCollection | Private | Truy cập đến thư viện nhạc trong thiết bị. |
|  | \_SourceSongGrouped | ObservableCollection<ArrSongGrouped> | Private | Copy từ \_Song, thực hiện việc lấy danh sách theo Album hoặc Artist |
| 4 | album | String | Private | Tên Album nhận được khi có sự kiện Navigate từ ListSongPage |
| 5 | artist | String | private | Tên Artist nhận được khi có sự kiện Navigate từ ListSongPage |
| 6 | ArrAlbum | String | Private | Lưu chuỗi index danh sách bài hát phát theo Album |
| 7 | ArrArtist | String | Private | Lưu chuỗi index danh sách bài hát phát theo Artist |
| 8 | kind | Int | Private | Xác định phát nhạc theo Album hay Artist |
| 9 | PlayListPage() | Phương thức |  | Private | Contructor |
| 10 | GroupSong\_Album() | Void | Private | Nhóm danh sách bài hát theo Album |
| 11 | GroupSong\_Artist() | Void | Private | Nhóm danh sách bài hát theo Artist |
| 12 | Tapped\_SongTB() | Void | Private | Sự kiên tap vào bài hát |
| 13 | Song\_Load() | Void | Private | Load PlayList Page |

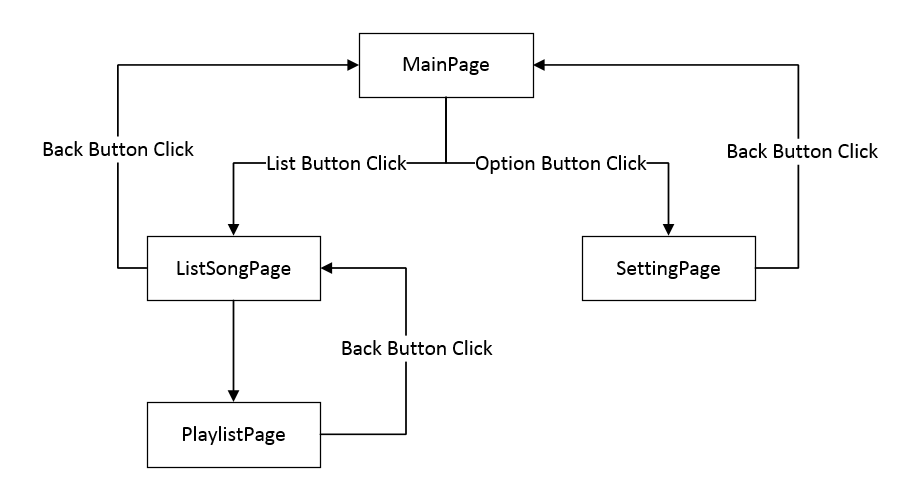
* Giao diện.//Thiết kế giao diện màn hình - PPMHH

Màn hình chính (MainPage).

Màn hình danh sách bài hát (ListSongPage): hiển thị theo tên bài hát (songs), album, ca sĩ (artist).

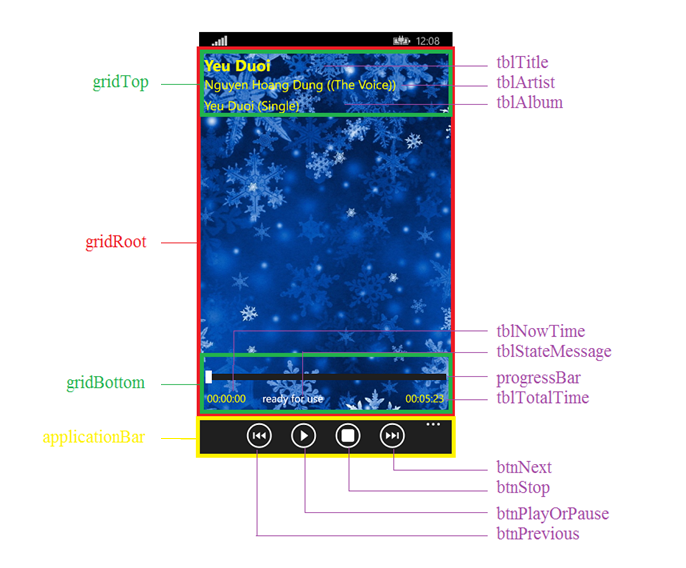
Màn hình bài hát (PlaylistPage). Màn hình cài đặt (SettingPage).

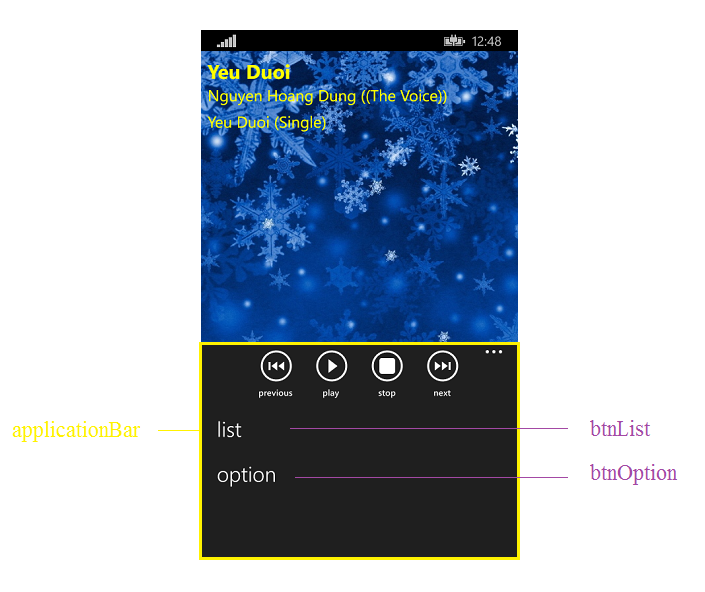
Sơ đồ liên kết giao diện:

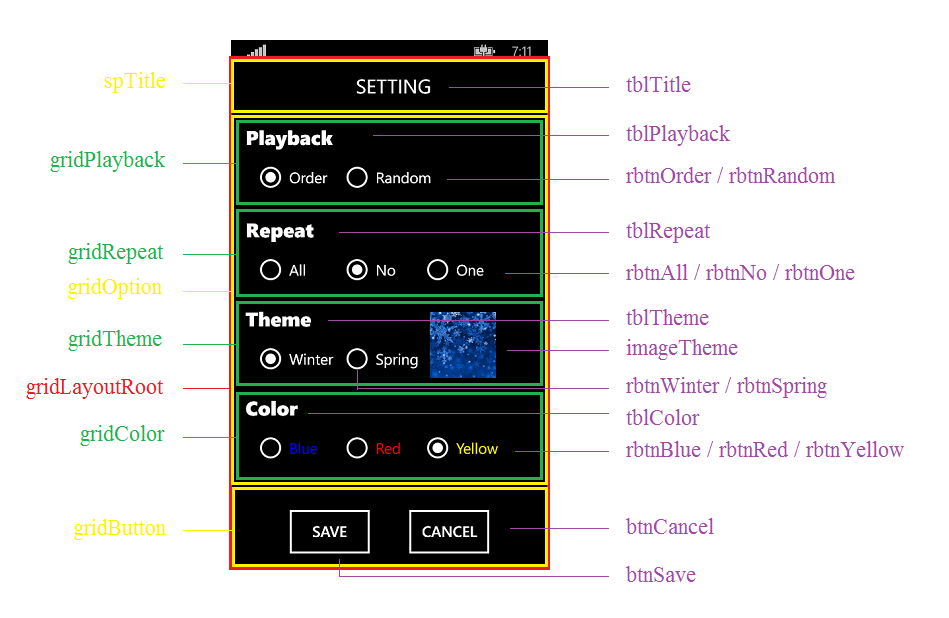


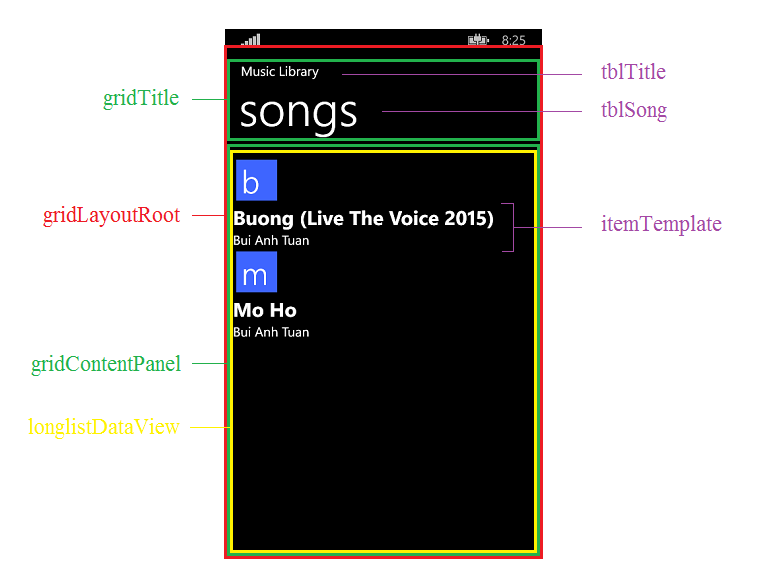
Thiết kế chi tiết:

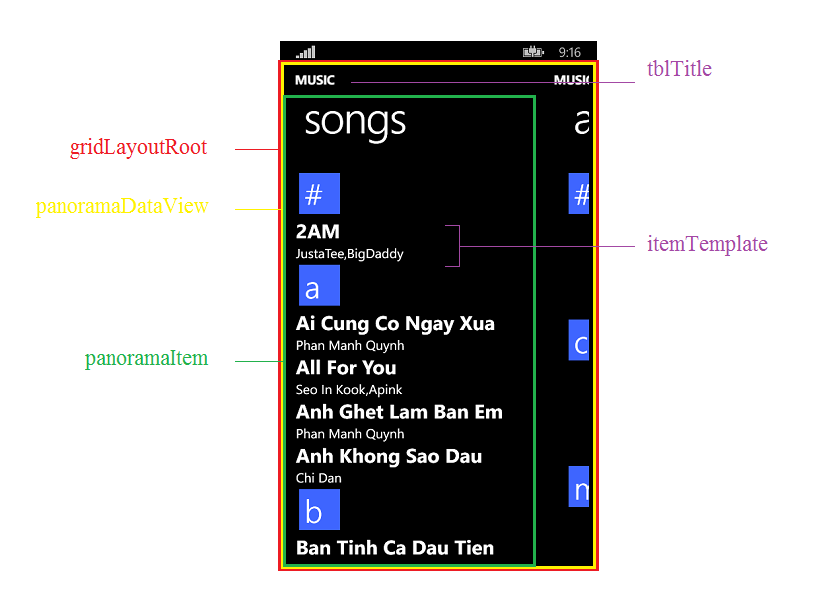
MainPage





SettingPage

ListSongPage

PlaylistPage

# **CÀI ĐẶT VÀ THỬ NGHIỆM**

//Chưa biết ghi thế nào =)) Chém đại

* Môi trường: Visual Studio tích hợp Windows Phone SDK.
* Ngôn ngữ cài đặt: C#
* Thử nghiệm: trên giả lập Windows Phone, trên các thiết bị thật: Lumia 820, Lumia 730, HTC 8X,…
* Đánh giá kết quả: chạy tốt trên các thiết bị đã thử nghiệm.//Chém gió vl =))

# **NHẬN XÉT VÀ KẾT LUẬN**

//Cái này để các bạn ý kiến thêm :v

* Ưu điểm:
  + Là là là ………. //Không biết =))) Chắc k có
  + Thao tác bằng lời nói thay vì thao tác bằng tay, thuận tiện hơn // add by Hang
* Nhược điểm:
  + Xử lý còn chập.
  + Nhận diện giọng nói chưa chuẩn (do tiếng ồn xung quanh) => Đã khắc phục một phần nhờ vào cơ chế Wakeup rồi mới nhận tín hiệu giọng nói tiếp theo để xử lý.
* Kết luận:
  + Còn hơi chuối =))
* Hướng phát triển:
  + Có thể nhận diện giọng nói ở chế độ chạy nền.
  + Không chỉ là phát nhạc, mà còn có thể chạy video, đa dạng hoá các định dạng phát.
  + Thêm chức năng cắt nhạc, ghép nhạc, tạo nhạc chuông.
  + … //Something else