MyMusicList UML Diagram - Ada's Firewall

Views

- ForgotPassword
- LoginPage
- LoginPageError
- Profile
- Result(Album)
- Result(Artist)
- Result(Track)
- ResultConfirmation(Album)
- ResultConfirmation(Artist)
- ResultConfirmation(Track)
- Search
- SearchNoContentSelected
- SearchNoTypeSelected
- Settings
- SettingsEmailChanged
- SettingsEmailsDontMatch
- SettingsImproperEmail
- SettingsPasswordChanged
- SettingsPasswordsDontMatch
- SignUpEmailError
- SignUpPage
- SignUpPasswordError
- SignUpSuccessful
- SignUpUsernameError

Application

Launch
- -ADDRESS: String
- -RESIZE: boolean
- -FIRSTVIEW: String

- +start_stage: Stage): void
- +main(_args:String[]): void

Controllers

LoginController
- -finalchecker: CheckUserInfo
- -dbAdapter: DBInfoRequest
- -NO_ERRORS: int
- -USERNAME_ERROR: int
- -PASSWORD_ERROR: int
- -INVALID_EMAIL_ERROR: int
- -EMAIL_TAKEN_ERROR: int
- -password: TextField
- -username: TextField
- -confirmPassword: TextField
- -email: TextField
- -confirmEmail: TextField
- -ADDRESS: String
- -RESIZE: Boolean
- -user: UserObject
- -PROFILEXML: String
- -LOGINPAGEERROR: String
- -SIGNUPPAGE: String
- -FORGOTPASSWORD: String
- -SIGNUPUSERNAMEERROR: String
- -SIGNUPPASSWORDERROR: String
- -SIGNUPSUCCESSFUL: String
- -SIGNUPIMPROPEREMAILERROR: String
- -SIGNUPEMAILERROR: String
- -LOGINPAGE: String

- #loginButtonClicked(_event: ActionEvent): void
- # signUpButtonClicked(_event: ActionEvent): void
- # forgotPasswordButtonClicked(_event: ActionEvent): void
- #registerButtonClicked(_event: ActionEvent): void
- # returnToLoginPageClicked(_event: ActionEvent): void
- # displayPage(_event: ActionEvent, _fxmlFile: String): void
- +initialize(_url: URL, _rb: ResourceBundle): void

ProfileController
- -tableView: TableView<RatingObject>
- -nameColumn: TableColumn<RatingObject, String>
- -typeColumn: TableColumn<RatingObject, String>
- -ratingColumn: TableColumn<RatingObject, String>
- -rateScore: ChoiceBox
- -SCORES: ObservableList<String>
- -TABLELIST: ObservableList<RatingObject>
- -user: UserObject
- -DBADAPTER: DBInfoRequest
- -RESIZE: Boolean
- -ADDRESS: String
- -LOGINFXML: String
- -RATINGFXML: String
- -SETTINGSFXML: String
- -SEARCHFXML: String
- -PROFILE: String

- #logOutButtonClicked(_event: ActionEvent): void
- # settingsButtonClicked(_event: ActionEvent): void
- #searchButtonClicked(_event: ActionEvent): void
- # deleteItemButtonClicked(_event: ActionEvent): void
- # changeRatingButtonClicked(_event: ActionEvent): void
- - displayPage(_event: ActionEvent, _fxmlFile: String): void
- + initializeUser(_user: UserObject): void
- + initialize(_url: URL, _rb: ResourceBundle): void

ResultController
- -tableView: TableView<MusicObject>
- -trackColumn: TableColumn<MusicObject, String>
- -artistColumn: TableColumn<MusicObject, String>
- -albumColumn: TableColumn<MusicObject, String>
- -releasedColumn: TableColumn<MusicObject, String>
- -rateScore: ChoiceBox
- - SCORES: ObservableList<String>
- - searchContent: String
- - searchType: String
- - item: MusicObject
- - TABLELIST: ObservableList<MusicObject>
- - user: UserObject
- - DBADAPTER: DBInfoRequest
- - ADDRESS: String
- - RESIZE: Boolean
- - ALBUMRESULT: String
- - ARTISTRESULT: String
- - TRACKRESULT: String
- - ALBUMCONFIRMATON: String
- - ARTISTCONFIRMATON: String
- - TRACKCONFIRMATION: String
- - SEARCH: String

- #addToMusicListButtonClicked(_event: ActionEvent): void
- # returnToSearchButtonClicked(_event: ActionEvent): void
- - displayPage(_event: ActionEvent, _fxmlFile: String): void
- + initializeData(_searchContent: String, _searchType: String, _user: UserObject): void
- + initialize(_url: URL, _rb: ResourceBundle): void

SettingsController
- - password: TextField
- - confirmPassword: TextField
- - email: TextField
- - confirmEmail: TextField
- - user: UserObject
- - DBADAPTER: DBInfoRequest
- - USERINFOCHECKER: CheckUserInfo
- -RESIZE: Boolean
- - ADDRESS: String
- - LOGIN: String
- - PROFILE: String
- - PASSWORDCHANGED: String
- - EMAILCHANGED: String
- - SETTINGSPASSWORDSDONTMATCH: String
- - SETTINGSEMAILSDONTMATCH: String
- - SETTINGSIMPROPEREMAIL: String
- - SETTINGS: String

- # changeEmailButtonClicked(_event: ActionEvent): void
- # changePasswordButtonClicked(_event: ActionEvent): void
- # backButtonClicked(_event: ActionEvent): void
- # deleteAccountButtonClicked(_event: ActionEvent): void
- - displayPage(_event: ActionEvent, _fxmlFile: String): void
- + initializeUser(_user: UserObject): void

SearchController
- - searchContent: TextField
- - searchType: ChoiceBox
- -typeOptions: ObservableList<String>
- - user: UserObject
- - RESIZE: Boolean
- - ADDRESS: String
- - NOCONTENTSELECTED: String
- - NOTYPESELECTED: String
- - ALBUMRESULT: String
- - ARTISTRESULT: String
- - TRACKRESULT: String
- - PROFILE: String

- # searchButtonClicked(_event: ActionEvent): void
- # returnToProfileButtonClicked(_event: ActionEvent): void
- - displayPage(_event: ActionEvent): void
- + initialize(_url: URL, _rb: ResourceBundle): void

Data

Database

DatabaseCreate
- +createRecord(_file: File, _record: ArrayList<String>): void
- +createUserRecord(_user: UserObject): void
- +createRating(_rating: RatingObject): void
- +createUserPlanToListen(_planToListen: PlanToListenObject): void

DatabaseDelete
- -active: String
- -inactive: String
- +deleteUserRecord(_user: UserObject): void
- +deleteUserRecord(_file: File, _user: UserObject): void
- +deleteUserRecord(_username: String): void
- +deleteRating(_rating: RatingObject): void
- +deletePlanToListen(_planToListen: PlanToListenObject): void
- -deleteRecord(_file: File, _record: String): void
- -deleteUserLogin(_user: UserObject): void

DatabaseRead
- -indexUUID: int
- -indexUsername: int
- -indexPassword: int
- -indexRating: int
- -indexEmail: int
- -indexSpotifyID: int
- -indexMusicObjectType: int
- -indexFirstName: int
- -indexLastName: int
- -noSuchRecord: int
- -active: String
- -inactive: String
- +readRecord(_file: File, _string: String): ArrayList<String>
- +readRecord(_file: File, _string: String): ArrayList<ArrayList<String>
- +readUserRecord(_username: String): UserObject
- +readUsersRating(_username: String): RecordObjectList
- +readUsersPlanToListen(_username: String): RecordObjectList

DatabaseUpdate
- +updateUserPassword(_user: UserObject, _newPassword: String): void
- +updateUsername(_user: UserObject, _newUsername: String): void
- +updateUserEmail(_user: UserObject, _newEmail: String): void
- +updateUserRating(_user: UserObject, _newMusicList, _newUsersPlanToListen: ArrayList<String>): void
- +updateUsersPlanToListenList(_user: UserObject, _newUsersPlanToListen: ArrayList<String>): void
- -updateUserPassword(_file: File, _user: UserObject, _newUsername: String): void
- -updateUsername(_file: File, _user: UserObject, _newUserName: String): void
- -updateUserEmail(_file: File, _user: UserObject, _newEmail: String): void
- -updateUsersPlanToListenList(_file: File, _user:MusicList, _newUsersPlanToListen: ArrayList<String>): void
- -updateUsersRating(_file: File, _user: RatingObject, _newUsersRating: double): void
- -updateFile(_file: File, _oldRecord: String, _newRecord: String): void

Datastore
- UserInfo.txt
- UserLogin.txt
- UserRating.txt
- UserPlanToListen.txt

DatabaseInterface
- +userInfoFile: File
- +userLoginFile: File
- +userRatingFile: File
- +userPlanToListen: File
- +temporaryFile: File
- +active: boolean
- +inactive: boolean
- +createRecord(_file: File, _record: ArrayList<String>): void
- +createUserRecord(_user: UserObject): void
- +createRating(_rating: RatingObject): void
- +createUserPlanToListen(_planToListen: PlanToListenObject): void
- +deleteUserRecord(_user: UserObject): void
- +deleteUserRecord(_username: String): void
- +deleteRating(_rating: RatingObject): void
- +deletePlanToListen(_planToListen: PlanToListenObject): void
- -deleteRecord(_file: File, _record: String): void
- -deleteUserLogin(_user: UserObject): void
- +updateUserPassword(_user: UserObject, _newPassword: String): void
- +updateUsername(_user: UserObject, _newUsername: String): void
- +updateUserEmail(_user: UserObject, _newEmail: String): void
- +updateUsersPlanToListenList(_user: UserMusicList, _newUsersPlanToListen: ArrayList<String>): void
- +updateUsersRating(_user: RatingObject, _newUsersRating: double): void
- +readRecord(_file: File, _string: String): ArrayList<String>
- +readUserRecord(_username: String): UserObject
- +readUsersRating(_username: String): RecordObjectList
- +readUsersPlanToListen(_username: String): RecordObjectList

API

MusicAPIInterface
- +ARTIST: String
- +ALBUM: String
- +TRACK: String
- +ALL_MUSIC_OBJECT_TYPES: String
- +NAME: String
- +ID: String
- +TYPE: String
- +GENRE: String

- +search(_keyword: String, _type: String, _limit: int): MusicObjectList
- + loadArtistById(_id: String): ArtistObject
- + loadAlbumById(_id: String): AlbumObject
- + loadTrackById(_id: String): TrackObject
- + loadAlbumWithTracks(_album: AlbumObject): AlbumObject
- + loadAlbumsOfArtist(_artist: ArtistObject, _limit: int): MusicObjectList
- + loadAlbumsOfArtist(_artistID: String _limit: int): MusicObjectList

MusicAPIAdapter
- +apiTranslator: MusicAPIInterface
- + search(_keyword: String, _type: String, _limit: int): MusicObjectList
- + loadArtistById(_id: String): ArtistObject
- + loadAlbumById(_id: String): AlbumObject
- + loadTrackById(_id: String): TrackObject
- + loadAlbumWithTracks(_album: AlbumObject): AlbumObject
- + loadAlbumsOfArtist(_artist: ArtistObject, _limit: int): MusicObjectList
- + loadAlbumsOfArtist(_artistID: String _limit: int): MusicObjectList

SpotifyAPITranslator
- -developerCredentials: String
- -accessToken: String
- -apiUrlString: String
- +SpotifyAPITranslator(_encodedDeveloperCredentials:String):
- +search(_keyword: String, _type: String, _limit: int): MusicObjectList
- + loadArtistById(_id: String):ArtistObject
- + loadAlbumById(_id: String): AlbumObject
- + loadTrackById(_id: String): TrackObject
- + loadAlbumWithTracks(_album: AlbumObject): AlbumObject
- + loadAlbumsOfArtist(_artist: ArtistObject, _limit: int): MusicObjectList
- + loadAlbumsOfArtist(_artistID: String _limit: int): MusicObjectList
- -requestAuthorization():String
- -performProcess(_curlCommand: String): JSONObject
- -loadMusicObjectById(_id: String, _type: String): MusicObject
- -convertSearchResults(_searchResults: JSONObject, _type: String): ArrayList<MusicObject>
- -convertJSONArrayOfSongs(_jsonArray: JSONArray): ArrayList<String>
- -convertJSONArrayArtists(_jsonArray: JSONArray): ArrayList<String>
- -convertJSONArtist(_jsonArtist: JSONObject): ArtistObject
- -convertJSONTrackObject(_jsonTrack: JSONObject): TrackObject
- -convertJSONAlbumObject(_jsonAlbum: JSONObject): AlbumObject

Models

CheckUserInfo
- -adapter: DBInfoRequest
- -NO_ERRORS: int
- -USERNAME_ERROR: int
- -PASSWORD_ERROR: int
- -INVALID_EMAIL_ERROR: int
- -EMAIL_TAKEN_ERROR: int
- -INVALID_LOGIN_int

- + CheckUserInfo(): void
- +isNewUserValid(_newUser: NewUserObject): int
- +isUserCredentialsValid(_theUsername: String, _thePassword: String): int
- +isEmailValid(_email: String): boolean

DBInfoRequest
- +DBInfoRequest(): void
- +createRecord(_file: File, _record: ArrayList<String>): void
- +createUserRecord(_user: UserObject): void
- +createUserPlanToListen(_planToListen: PlanToListen): void
- +deleteUserRecord(_user: UserObject): void
- +deleteUserRecord(_username: String): void
- +readRecord(_file: File, _string: String): ArrayList<String>
- +readUserRecord(_username: String): UserObject
- +readUsersRating(_username: String): RecordObjectList
- +readUsersPlanToListen(_username: String): RecordObjectList
- +updateUserPassword(_user: UserObject, _newPassword: String): void
- +updateUsername(_user: UserObject, _newUsername: String): void
- +updateUserEmail(_user: UserObject, _newEmail: String): void

MusicRequest
- #adapter: MusicAPIAdapter
- -DEFAULT_SEARCH_TYPE: String
- -DEFAULT_QUERY_LIMIT: int
- -MIN_QUERY_LIMIT: int
- -MAX_QUERY_LIMIT: int
- +search (_keyword: String): MusicObjectList
- +search(_keyword: String, _type: String, _limit: int): MusicObjectList
- +loadArtistById(_id: String): ArtistObject
- +loadAlbumById(_id: String): AlbumObject
- +loadTrackById(_id: String): TrackObject
- +loadAlbumsOfArtist(_artist: AlbumObject): AlbumObject
- +loadAlbumsOfArtist(_artist: ArtistObject, _limit: int): MusicObjectList
- +loadAlbumsOfArtist(_artistID: String, _limit: int): MusicObjectList
- +help(): void
- -searchHelper(_keyword: String, _type: String, _limit: int): MusicObjectList

Objects

MusicObject
- #name: String
- #id: String
- #type: String
- #genre: String
- +MusicObject(_name: String, _id: String, _type: String, _genres: ArrayList<String>): void
- +MusicObject(_name: String, _id: String, _type: String, _genres: String): void
- +MusicObject(_name: String, _id: String, _type: String, _genres: String []): void

TrackObject
- #artist: String
- #album: String
- #year: String
- +TrackObject(_name: String, _id: String, _type: String, _genres: ArrayList<String> , _artist: String, _album: String, _year: String): void
- +TrackObject(_name: String, _id: String, _type: String, _genres: String, _artist: String, _album: String, _year: String):void
- +TrackObject(_name: String, _id: String, _type: String, _genres: String[], _artist: String, _album: String, _year: String):void
- +toString(): String
- +getArtist(): String
- +getAlbum(): String
- +getYear(): String

ArtistObject
- #url: String
- +ArtistObject(_name: String, _id: String, _type: String, _genres: ArrayList<String> , _url: String): void
- +ArtistObject(_name: String, _id: String, _type: String, _genres: String): void
- +ArtistObject(_name: String, _id: String, _type: String, _genres: String [] , _url: String): void
- +toString(): String
- getURL(): String

PlanToListenObject
- +PlanToListenObject(_username: String, _spotifyID: String, _musicObjectType: String): void
- +PlanToListenObject(_uuid: String, _username: String, _spotifyID: String, _musicObjectType: String): void
- +PlanToListenObject(_nullValue: String): void
- +toArrayList(): ArrayList<String>

AlbumObject
- #artist: String
- #year: String
- #tracks: MusicObjectList
- +AlbumObject(_name: String, _id: String, _type: String, _genres: ArrayList<String> , _artist: String, _year: String, _tracks: MusicObjectList): void
- +AlbumObject(_name: String, _id: String, _type: String, _genres: String, _artist: String, _year: String, _tracks: MusicObjectList): void
- +AlbumObject(_album: AlbumObject, _tracks: MusicObjectList): void
- -tracksLoaded(): boolean
- +getArtist(): String
- +getYear(): String
- +getTracks(): MusicObjectList
- +getTracksInArrayList(): ArrayList<MusicObject>

RecordObject
- #uuid: String
- # name: String
- # id: int
- #active: boolean
- +RecordObject(_name: String, _type: String, _id: int): void
- +RecordObject(_uuid: String , _name: String, _id: int): void
- +RecordObject(): void
- +RecordObject(_uuid: String): void
- +makeActive(): void
- +makeInactive(): void
- +toArrayList(): ArrayList<RecordObject>
- +generateUuid(): String
- +isActive(): boolean
- +toString(): String
- +getName(): String
- +getID(): int
- + setUuid(_uuid: String): void
- +setName(_name: String): void
- +setId(_id: int): void

UserObject
- - userName: String
- - userPassword: String
- - userEmail: String
- - userFirstName: String
- - userLastName: String
- +UserObject(_name: String, _id: int , _userName: String, _userPassword: String, _userEmail: String, _userFirstName: String, _userLastName: String): void
- +UserObject(_uuid: String, _name: String, _id: int , _userName: String, _userPassword: String, _userEmail: String, _userFirstName: String, _userLastName: String): void
- +UserObject(_uuid: String, _name: String, _userName: String, _userPassword: String, _userEmail: String, _userFirstName: String, _userLastName: String): void
- +UserObject(_nullValue: String): void
- +toArrayList():ArrayList<String>
- +toString(): String
- +getUserName(): String
- +getUserPassword(): String
- +getUserEmail(): String
- +getUserFirstName(): String
- +getUserLastName(): String
- + setUserName(_userName: String): void
- + setUserPassword(_userPassword: String): void
- + setUserEmail(_userEmail: String): void
- + setUserFirstName(_userFirstName: String): void
- + setUserLastName(_userLastName: String): void

RatingObject
- - username: String
- - usersRating: double
- - spotifyID: String
- - musicObjectType: String
- +RatingObject(_name: String, _id: int , _usersRating: double, _spotifyID: String, _musicObjectType: String): void
- +RatingObject(_uuid: String, _username: String, _usersRating: double, _spotifyID: String, _musicObjectType: String): void
- + RatingObject(_username: String, _usersRating: double, _spotifyID: String, _musicObjectType: String): void
- +RatingObject(_uuid: String, _username: String, _spotifyID: String, _musicObjectType: String): void
- +RatingObject(_nullValue: String): void
- +RatingObject(_null: String): void
- +toArrayList(): ArrayList<String>
- +toString(): String
- + getUsername(): String
- +getUsersRating(): double
- + getSpotifyID: String
- +getMusicObjectType(): String
- + setUsername(_username: String): void
- + setUsersRating(_usersRating: double): void
- + setSpotifyID(_spotifyID: String): void
- + setUUID(_uuid: String): void
- + setMusicObjectType(_musicObjectType: String): void

NewUserObject
- -confirmPassword: String
- +NewUserObject(_userName: String, _password: String, _confirmPassword: String, _email: String, _firstName: String, _lastName: String): void
- + getConfirmPassword(): String
- + setConfirmPassword(_confirmPassword: String): void

RecordObjectList
- #list: ArrayList<String>
- #name: String
- #type: String
- #length: int
- +RecordObjectList(_name: String, _type: String): void
- +RecordObjectList(_name: String, _type: String, _list: ArrayList<RecordObject>): void
- +add(_recordObject: RecordObject): void
- +remove(_index: int): void
- +get(_index: int): RecordObject
- +toString(): String
- +getName(): String
- +getType(): String
- + getLength(): int
- +getList(): ArrayList<RecordObject>
- +setName(_name: String): void
- +setType(_type: String): void
- +setList(_list: ArrayList<RecordObject>): void

MusicObjectList
- #list: ArrayList<String>
- #name: String
- #type: String
- #length: int
- +MusicObjectList(_name: String , _type: String): void
- +MusicObjectList(_name: String, _type: String, _arrayList:ArrayList<MusicObject>): void
- +add(_item: MusicObject): void
- +remove(item_index: int): void
- + get(_index: int): MusicObject
- +getList(): ArrayList<MusicObject>
- +getAllAlbumObjects(): MusicObjectList
- +getAllArtistObjects(): MusicObjectList
- +getAllTrackObjects(): MusicObjectList
- +clear(): void
- +toArrayList(): ArrayList<MusicObject>
- +toString(): String
- +getName(): String
- +getType(): String
- +getLength(): int
- +setName(_name: String): void
- +setType(_type: String): void
- +setList(_newList: ArrayList<MusicObject>): void

UserMusicList
- - userName: String
- #usersAlbumsCompleted: ArrayList<String>
- #usersPlanToListen: ArrayList<String>
- #usersFavoriteAlbums: ArrayList<String>
- #usersFavoriteArtists: ArrayList<String>
- #usersFavoriteTracks : ArrayList<String>
- +UserMusicList(_userName: String, _usersAlbumsCompleted: ArrayList<String>, _usersPlanToListen: ArrayList<String>, _usersFavoriteAlbums: ArrayList<String>, _usersFavoriteArtists: ArrayList<String>, _usersFavoriteTracks: ArrayList<String>): void
- +getUserName(): String
- +getUsersAlbumsCompleted(): ArrayList<String>
- +getUsersPlanToListen(): ArrayList<String>
- +getUsersFavoriteAlbums(): ArrayList<String>
- +getUsersFavoriteArtists(): ArrayList<String>
- +getUsersFavoriteTracks(): ArrayList<String>
- +setUsersAlbumsCompleted(_usersAlbumsCompleted: ArrayList<String>): void
- +setUsersPlanToListen(_usersPlanToListen: ArrayList<String>): void
- +setUsersFavoriteAlbums(_usersFavoriteAlbums: ArrayList<String>): void
- +setUsersFavoriteArtists(_usersFavoriteArtists: ArrayList<String>): void
- +setUsersFavoriteTracks(_usersFavoriteTracks: ArrayList<String>): void