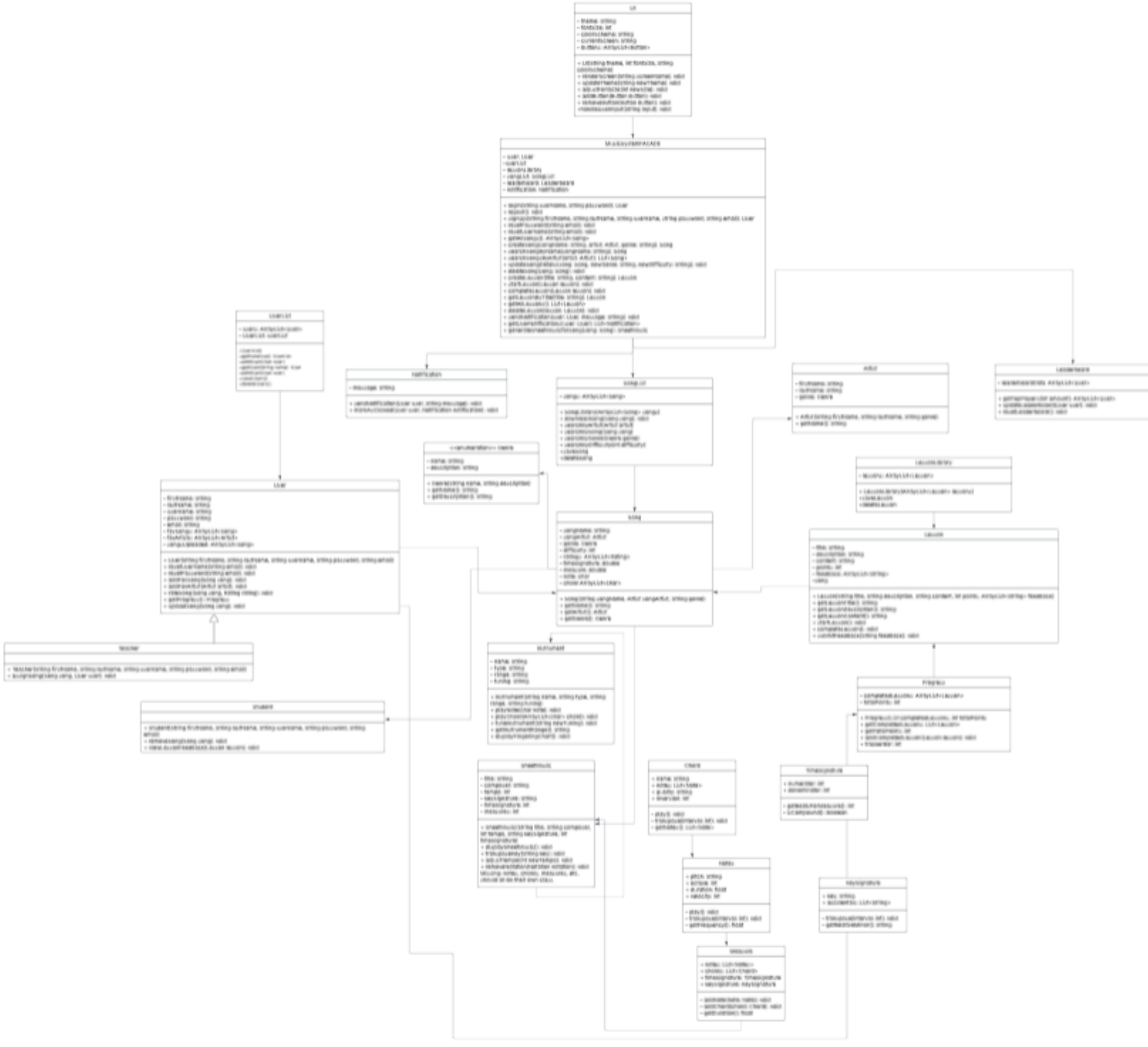
UI
- frame: String
- frameSize: int
- currentName: String
- currentUsers: String
- buttons: ArrayList<Button>
+ UI(String frame, int frameSize, String currentName)
+ initializeUser(String username): void
+ updateFriendsList(String newFriend): void
+ updateFriendsSize(int newSize): void
+ addButton(Button button): void
+ removeButton(Button button): void
+ handleUserInput(String input): void
MusicSyncFACADE
- user: User
- userList
- musicLibrary
- songList: SongList
- leaderboard: LeaderboardS
- notification: Notification
+ login(String username, String password): User
+ logout(): void
+ signup(String firstname, String surname, String username, String password, String email): User
+ modifyUserEmail(String email): void
+ modifyUsername(String email): void
+ getAllSongs(): ArrayList<song>
+ addSong(String songname, String artist, Artist, genre: String): song
+ updateSongname(songname, String): song
+ updateSongLyrics(Artist): List<song>
+ updateSongDifficulty(song: song, newName: String, newDifficulty: String): void
+ deleteSong(song: song): void
+ createLesson(title: String, content: String): Lesson
+ startLesson(lesson lesson): void
+ completeLesson(Lesson lesson): void
+ getLessonByTitle(title: String): Lesson
+ getAllLessons(): List<Lesson>
+ deleteLesson(Lesson): void
+ sendNotification(user: User, message: String): void
+ getUserNotifications(user: User): List<Notification>
+ generateNewNotificationsFromSyncing: song): snapshot
Notification
- message: String
+ sendNotification(user: User, String message): void
+ markNotificationRead(user: User, notification notification): void
SongList
- songs: ArrayList<song>
+ song(ArrayList<song> songs)
+ addSong(song song): void
+ updateSongTitle(Artist artist)
+ updateSongLyrics(song: song)
+ updateSongGenre(genre genre)
+ updateSongDifficulty(int difficulty)
+ toString()
+ hashCode()
Artist
- firstname: String
- surname: String
- genre: Genre
+ Artist(String firstname, String surname, String genre)
+ getName(): String
LeaderboardS
- leaderboardSets: ArrayList<User>
+ getTopPlayers(int amount): ArrayList<User>
+ updateLeaderboard(User user): void
+ displayLeaderboard(): void
<<enumeration>> Genre
- name: String
- description: String
+ Genre(String name, String description)
+ getName(): String
+ getDescription(): String
User
- firstname: String
- surname: String
- username: String
- password: String
- email: String
- friends: ArrayList<song>
- favorites: ArrayList<Artist>
- completedLessons: ArrayList<song>
+ User(String firstname, String surname, String username, String password, String email)
+ modifyUsernameEmail(String email): void
+ modifyUsernameEmail(String email): void
+ addFriend(String song): void
+ addFavoriteArtist(Artist artist): void
+ removeSong(Song song, String rating): void
+ getFriendsList(): Friendlist
+ updateSong(Song song): void
LessonLibrary
- lessons: ArrayList<Lesson>
+ LessonLibrary(ArrayList<Lesson> lessons)
+ getLesson
+ getAllLessons
Lesson
- title: String
- description: String
- content: String
- points: int
- feedback: ArrayList<String>
- song
+ Lesson(String title, String description, String content, int points, ArrayList<String> feedback)
+ getLessonTitle(): String
+ getLessonDescription(): String
+ getLessonContent(): String
+ startLesson(): void
+ completeLesson(): void
+ updateFeedback(String feedback): void
Song
- songname: String
- songartist: Artist
- genre: Genre
- difficulty: int
- rating: ArrayList<Rating>
- timesignature: double
- tempo: double
- key: char
- chords: ArrayList<Chord>
+ song(String songname, Artist songartist, String genre)
+ getName(): String
+ getArtist(): Artist
+ getGenre(): Genre
Teacher
+ Teacher(String firstname, String surname, String username, String password, String email)
+ assignsong(Song song, User user): void
Student
+ Student(String firstname, String surname, String username, String password, String email)
+ addsong(Song song): void
+ removeLessonFromLibrary(Lesson lesson): void
Friendlist
- connectedFriends: ArrayList<User>
- totalFriends: int
+ Friendlist(ArrayList<User> friends, int totalFriends)
+ getConnectedFriends(): List<User>
+ getFriendsCount(): int
+ addConnectedFriend(User user): void
+ removeFriend(int): void
Rhythm
- name: String
- type: String
- tempo: int
- duration: int
- timesignature: int
- bitsPerBeat: int
+ Rhythm(String name, String type, int tempo, int timesignature, int bitsPerBeat)
+ getDurationInMilliseconds(): int
+ toDurationArray(String sec): void
+ setTempo(int newTempo): void
+ setNewNotation(int[] bitsPerBeat): void
Strumming, Arpeggio, Chords, Melodies, etc. (should be in that main class)
Chord
- name: String
- notes: List<Note>
- quality: String
- inversion: int
+ Chord(): void
+ Chord(String name, int): void
+ getNotes(): List<Note>
TimeSignature
+ numerator: int
+ denominator: int
+ getBeatsPerMeasure(): int
+ toCompound(): boolean
Note
- pitch: String
- octave: int
- duration: float
- velocity: int
+ Note(): void
+ Note(pitch String, int): void
+ getFrequency(): float
KeySignature
- key: String
- accidentals: List<String>
+ KeySignature(int): void
+ getKeyNameAndAccidentals(): String
Measure
- notes: List<Note>
- chords: List<Chord>
- timesignature: TimeSignature
- keysignature: KeySignature
+ addNote(Note note): void
+ addChord(Chord chord): void
+ getLength(): float