A Band Meetup web application

Software Requirements Specification Document – (SRS)

BandMeetUp

Rochester Institute of Technology

Golisano College of Computing and Information Sciences

Date: 11/11/2020

Version 1.0: SRS-10/12/2019

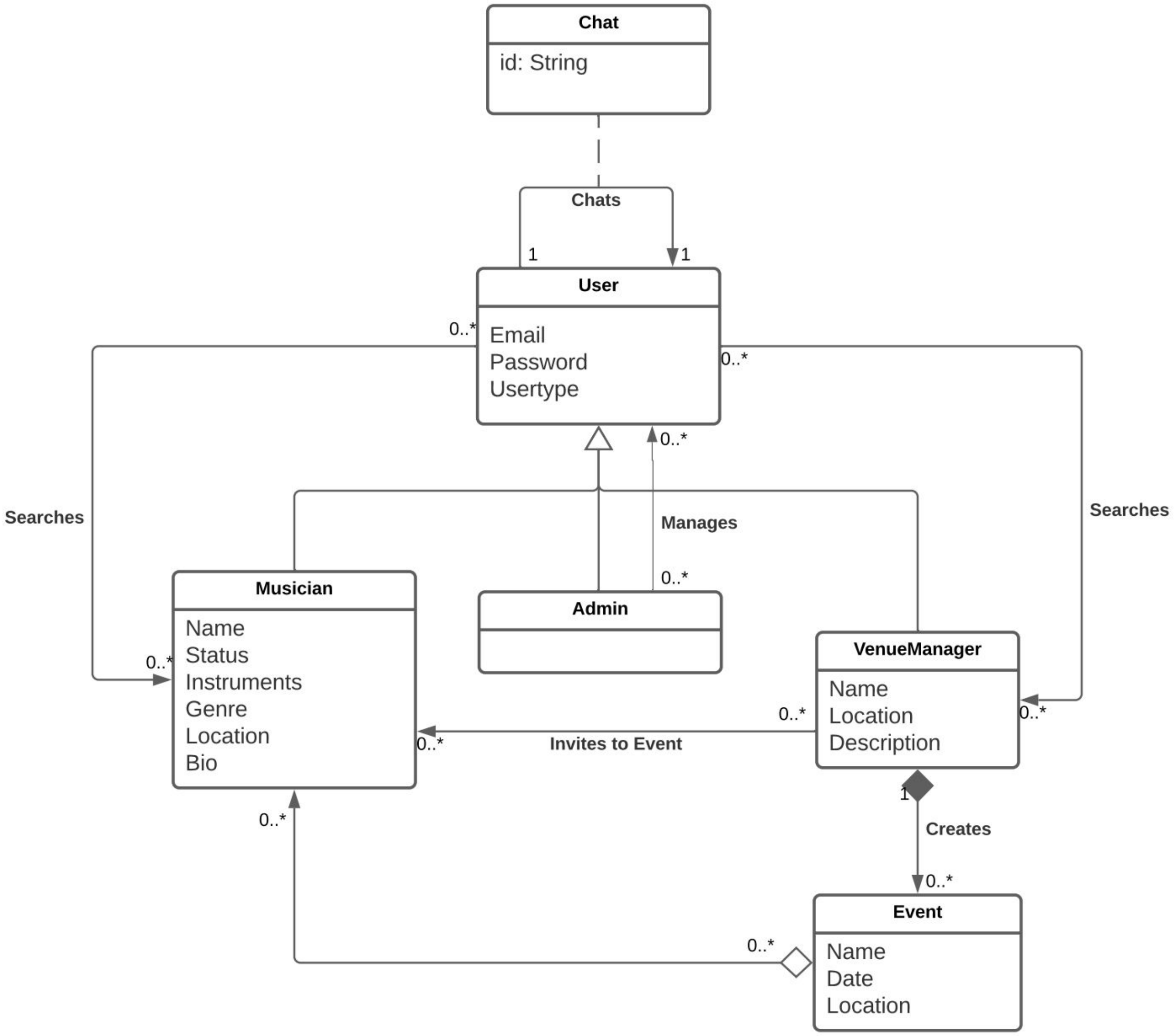
Table of Content

1 Introduction

1.1 System Overview

Band Meetup is a web application that allows musicians to connect to share their common music interest. Musicians can create bands by searching for musicians profiles that can potentially fit their interest. Musicians can also search for bands that are looking for members or searching individual musicians just to jam together. Each musician will have a user profile containing basic information(e.g. name, email, bio, age, etc). There will also be a music section in the user profile containing their music interest and sample clips which are either uploaded or linked to via social media (YouTube, sound cloud, twitter, etc). Venues will be able to post requests for musicians to play at their businesses. When musicians connect there will be an opening of communication via chat room. The invited musicians are not expected to respond to these chat room requests if they are not interested. Musicians and Venues can report profiles and business if they are scam or being racist. The search will be able to filter profiles that fulfill the filter setting(e.g. genre, age, gender, location, etc). The sorting of musicians will be defaulted to closest to farthest.

1.2 Conceptual Model



2 Functional Requirements

2.1 Product Requirements

General Profile Requirements:

1. Users shall be able to register new accounts using an email and password.
2. Users shall be able to choose their account type: Musician or Venue
3. Users shall be able to view and edit their profile information.
4. Users shall be able to set their location.

Musician Profile Requirements:

1. Musicians shall be able to add musical instruments and musical genre to their profile information.
2. Musicians shall be able to add and update a small biography.
3. Musicians shall be able to add and link to external social media profiles.
4. Musicians shall be able to upload audio files to their profile.
5. Musicians shall be able to set profile status.
6. Statuses shall be “Looking to join a band”, “Looking for a band member”, “Looking to start a band”, or “Looking to jam”.
7. Musician Profile displays information: Name, Email, Status, Musical Instrument, Musical Genre, Location and Bio.

Venue Profile Requirements:

1. Venues shall be able to create and link new events
2. Event details shall consist of Date, Location, and Name.
3. Venue Profile displays its information: Name and Location
4. Venue shall be able to invite musicians to join their events.

Profile and Event Searching Requirements:

Event should be displayed in Venue’s profile

1. Users shall be able to search Events.
2. Users shall be able to search Venue.
3. Users shall be able to filter their search of musicians based on four categories:
   - 3.1. Area code
   - 3.2. Genre of the music
   - 3.3. Musical instruments
   - 3.4. Profile Status
4. Users shall be able to view user profiles.

Chat Requirements:

1. Users shall be able to send and receive messages.
2. Users shall be able to invite other users to chat.
3. Chats shall be opened when the invitation is accepted.
4. Users shall be able to view previous chats.

Admin Requirements:

1. Admin shall be able to create new profiles.
2. Admin shall be able to view profiles.
3. Admin shall be able to modify profiles
4. Admin shall be able to delete profiles

2.2 Use Case Flows

Use Case Number:	1
Use Case Name	Login
Overview:	A user can log into their account.
Type:	Primary
Actors:	Musician, Administrator, Venue Manager
Pre-condition	Login page is displayed
Main Flow:	1. System request login information 2. User submits login credentials. 3. System validates and verifies login credentials, If failure go to Alternative Case 3.1 4. System redirects User to the Homepage.
Alternative Flow	3.1 If login credentials are not verified, prompt users to enter credentials again.
Post-Condition	Success: ● User Successfully logged in Failure: ● Invalid Credentials, unable to log in
Cross-reference	

Use Case Number:	2
Use Case Name	Registration
Overview:	A user can create a new account.
Type:	Primary
Actors:	Musician, Venue Manager
Pre-condition	Registration page is displayed
MainFlow:	1. The system requests registration details. 2. The user submits registration information 3. The system verifies (ID) email then store registration information, If failure go to Alternative Case 3.1 4. The system redirects the user to login page
Alternative Case	3.1 If email is used, prompt users to enter another email.
Post-Condition	Success: ● User Successfully registered as an authentic user. Failure: ● Users with invalid email will not be able to register.
Cross-reference	

Use Case Number:	3
Use Case Name	Search and view for user profile
Overview:	Musician or Venue Manager can search for other musician
Type:	Primary
Actors:	Musician, Venue Manager
Pre-condition	User logged in
MainFlow:	1. The system displays home page 2. The user selects search. 3. The user filters attributes based on the selection(e.g. location or type of instruments) 4. The system displays a list of musicians. 5. The user selects a profile to view 6. The system displays the requested profile page
Alternative Flow	

Post-Condition	User has requested profile pages shown
Cross-reference	

Use Case Number:	4
Use Case Name	Editing User Profile
Overview:	Process for user editing their profile
Type:	Primary
Actors:	Musician, Venue Manager, Administrator
Pre-condition	User logged in
MainFlow:	1. User requests edit of profile 2. System displays the profile edit for with preloaded data 3. The user can edit profile information and submit 4. The system saves the profile information
Alternative Flow	
Post-Condition	User is returned to homepage
Cross-reference	

Use Case Number:	5
Use Case Name	Message other user
Overview:	User can message each other
Type:	Primary
Actors:	Musician, Venue Manager, Administrator
Pre-condition	User logged in
MainFlow:	1. The user opens another user’s profile. 2. The system displays user profile 3. The user sends invite to user 4. The system displays a chat 5. The user send a message
Alternative Flow	

Post-Condition	Message page is displayed
Cross-reference	

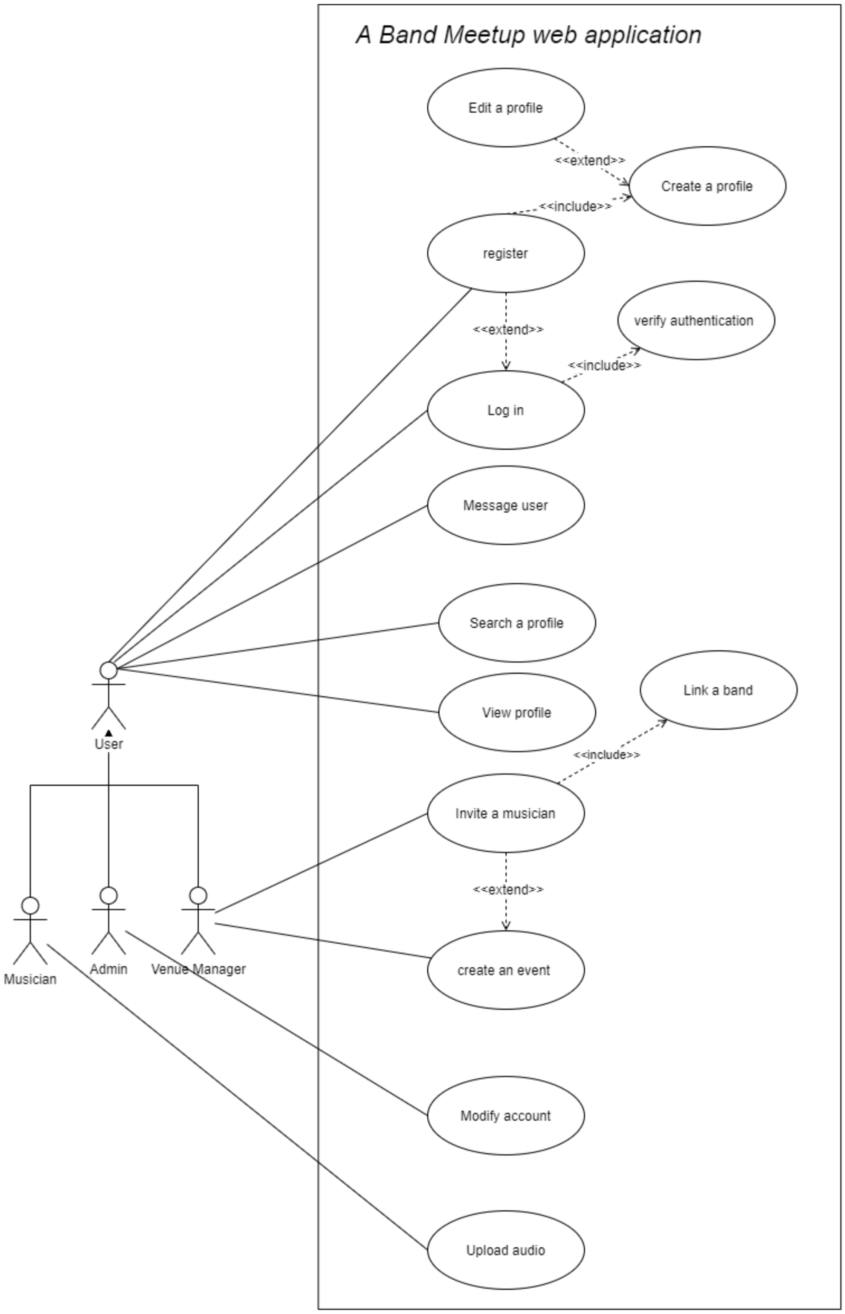
Use Case Number:	6
Use Case Name	Musician uploads their music clips
Overview:	Musician or Venue Manager can search for other musician
Type:	Primary
Actors:	Musician
Pre-condition	User logged in
MainFlow:	1. The system displays the user’s profile. 2. The user selects to upload a new audio clip. 3. The system requests audio clip formatting information. 4. The user submits audio clip formatting information. 5. The system requests the audio clip file for uploading. 6. The user uploads its music clips. In case of failure, Alternative 4.1 7. The system redirects user to user’s profile.
Alternative Flow	4.1 The system indicates failure. 4.2 The system prompts the user to re-upload the music file and redirects to the upload page.
Post-Condition	The system displays the new music clip upload along with the list of other music clips.
Cross-reference	

Use Case Number:	7
Use Case Name	Create event
Overview:	Venue Manager can create event
Type:	Primary
Actors:	Venue Manager
Pre-condition	User logged in
MainFlow:	1. The user navigates user profile

	2. The system displays user profile 3. The user creates an event 4. The system request event information 5. The user enters and submit event information 6. The system creates event and displays event on user profile
Alternative Flow	
Post-Condition	An event is created successfully and displayed in the Venue manager's page.
Cross-reference	

Use Case Number:	8
Use Case Name	Operation on accounts
Overview:	Administrator manages the accounts
Type:	Primary
Actors:	Administrator
Pre-condition	Administrator logged in
MainFlow:	1. Admin selects a user 2. System displays user account 3. Admin updates user account, if admin deletes account go to step 3.1 4. The system requires confirmation from the admin. 5. The admin confirms 6. System updates account
Alternative Flow	3.1 delete account 3.1.1 Admin opts a user to delete. 3.1.2 The system requires confirmation from the admin. 3.1.3 The admin confirms 3.1.4 The system accepts the confirmation
Post-Condition	User account is operated successfully
Cross-reference	

2.3 User Interaction Specifications



2.5 External Interfaces and Database Requirements

- SQLite : database (since the application performs straightforward CRUD operations on account handling)

3 Non-Functional Requirements

Security

1. User’s data shall be confidential.
2. The system shall be able to verify the account’s email address.
3. Each role should have different page
4. Password shall be encrypted

Availability

1. The product shall be available only to registered users.

Usability

1. User should navigate easily in the system
2. System shall support only English language