University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 3

Date: Oct. 30, 2023

Group Number: 120

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Huxley Bragg	17768847	q2n8a	hbragg16@gmail.com
Kai Alami	25764333	i3z6r	kaikaialami@gmail.com
Callie Mays	20807426	l5j7s	calliemays@outlook.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Summary

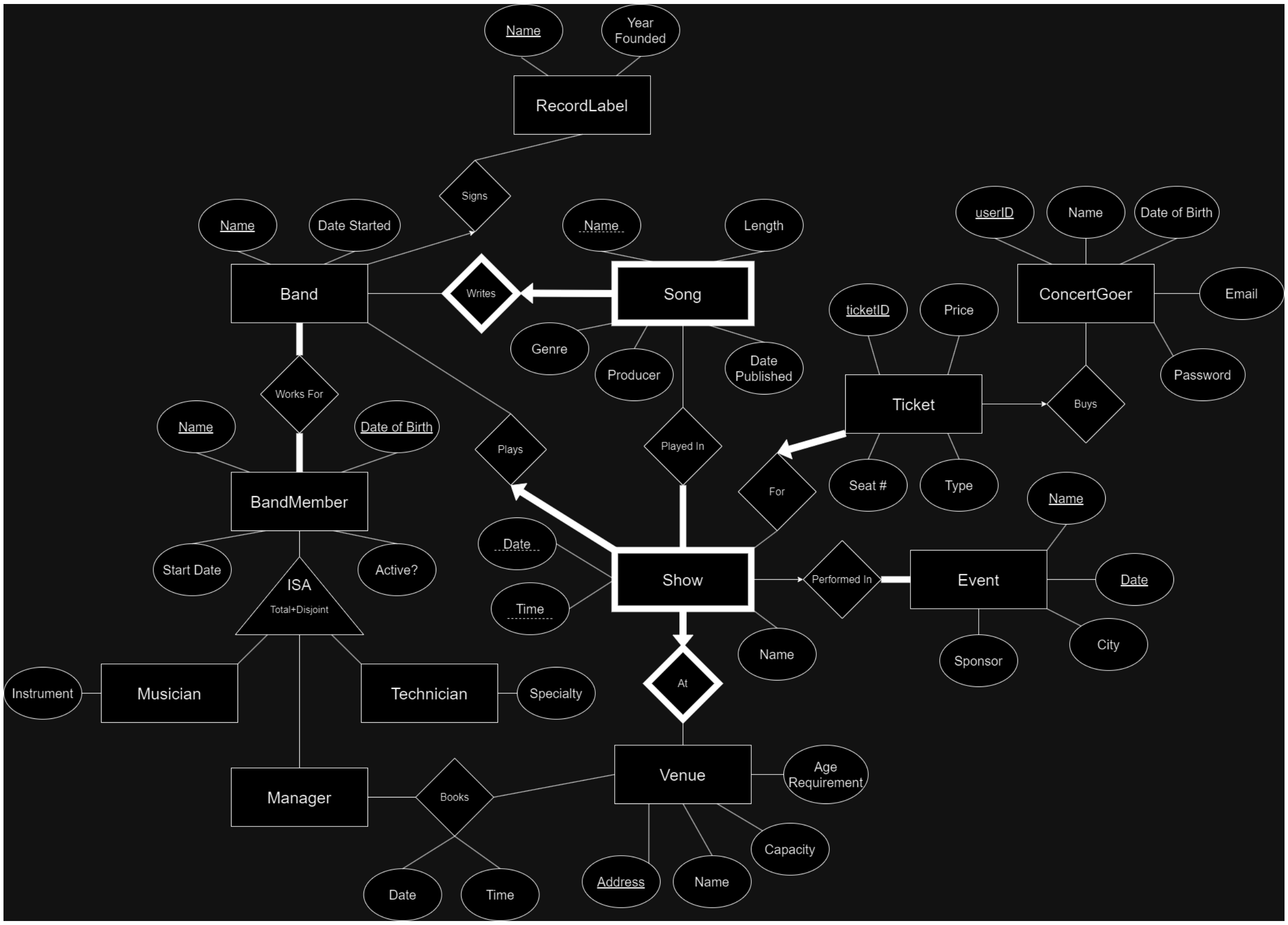
Our project aims to be a way for people interested in seeing bands perform to find performances that they would like to buy tickets for. For example, if someone is a fan of some band, they can use this application to look up upcoming shows by the band, including information about venue, setlist, band members, whether the show is a part of some event, etc. It also allows for potential performers to publish upcoming shows for concert goers to buy tickets for.

Minor Changes

ConcertGoer now has a password field

ConcertGoer userID is a string to represent username

Show no longer has participation constraint on Ticket



Timeline

Nov. 7

- Initial setup of database, populate tables with sample data - all
- Begin frontend and backend implementations, set up / familiarize with SQL / PHP - all

Nov. 13

- Backend implementation started
    - User account setup and login - Callie
    - Query for shows by band/venue/event - Kai
- Frontend implementation started
    - Landing page to navigate to either "mode" - Huxley
        - Logo / graphic design - Huxley
    - Create user account and login interfaces - Callie
    - Search input fields for finding shows - Kai

Nov. 20

- Frontend
    - Page linkage - buttons lead to correct pages - Huxley
    - Band mode GUI
        - Band "log-in"/"create new band" GUI interface - Kai
        - Input fields for adding members to band - Kai
        - Input fields for adding songs to band - Kai
        - Input fields for booking shows - Callie
        - Input fields for creating tickets for shows - Callie
    - Concertgoer mode GUI functional - Huxley
        - Input fields to search for shows by band/venue/event name - Huxley
        - Input field to search for information on a band - Huxley

Nov. 27

- Backend
    - Querying for shows by band, venue or event name - Huxley
    - "Purchasing" a ticket (update tuple in TicketID) - Huxley
    - Querying for available tickets for a show - Callie
    - Band Mode
        - Functionality to create a band and "log in" as a band - Kai
        - Functionality to create a song - Kai

        - Functionality to add band members - Kai
        - Functionality to add songs to setlists via PlayedIn - Callie
        - Functionality to create a show - Callie
        - Functionality to add shows to existing events - Callie
        - Functionality to create a ticket - Callie

Dec. 1

- Debug code - all
- Prepare demo - all

Tasks and Project Outline (things left to do)

Oracle - setup and get used to using it

- https://www.students.cs.ubc.ca/~cs-304/resources.html

Database

- Oracle
- Create tables
- Populate tables with several objects for each

Backend

- PHP
- Provide implementation for
    - Creating a new user account (insert into ConcertGoer)
    - Querying for shows by band, venue or event
    - Querying for available tickets for a show
    - "Purchasing" a ticket (update tuple in TicketID)
    - Creating a new band with one band member, optionally an existing RecordLabel (insert into Band, BandMember, WorksFor, associated ISA table)
    - Adding band members to existing bands (insert into BandMember, WorksFor, associated ISA table)
    - Booking a venue for a show by a band, includes one song by the band and optionally an event (insert into Books, Show, PlayedIn).

    - Create tickets for a show (insert into TicketID, TicketPrice, TicketType)
        - Seat numbers generated as a seat-type-code ('F', 'L', 'U', 'B') suffixed by a 3-digit number in the range ['001', 'n+1'] where n is the number of tickets in that seat-type
        - E.g. 100 balcony seats → seats {'B001', …, 'B101'}
    - Adding songs to the band's discography (insert into Song)
    - Adding songs to a show's setlist (insert into PlayedIn)

Frontend/GUI

- PHP (html and css)
- Two "modes": ConcertGoer (i.e. customer) and Band. User can switch between the two as separate web pages
- ConcertGoer mode
    - Initially has two buttons/collection of input fields:
        - Button to create a new account (insert into ConcertGoer) with the following fields
            - userID (i.e. unique username)
            - Password
            - Name
            - Email (unique)
            - Date of birth (optional) - dropdown menu for year month date
        - Button to "log-in" by entering an existing userID and associated password
    - Once logged in:
        - Search bar to find shows by a specific band (using band name)
        - Search bar to find shows at a specific venue (using venue name)
        - Search bar to find shows at a specific event (using event name)
        - Search bar to see list of band members in a band
- Band mode
    - Input field to specify which band will be using the app by entering the band name
        - If the band exists, move to the main Band mode webpage

        - If it does not exist, asks for the info for one band member and creates the new band with the name and other attributes, then move to the main Band mode webpage
    - Input fields and buttons to add new band member to the band, tick box and input fields for one of each ISA subclass (e.g. musician)
    - Input fields and buttons to add a song to the band's list of songs
    - Search for venues that have an available booking at an input date and time
    - Input fields and buttons for booking a venue for a show using a manager's information. Input includes show and band information, optionally includes an existing event
    - See all the tickets for a show
        - Add tickets of a specific type (e.g. balcony) and a quantity of that type
- Each page will have a logo in the corner that you can click to take you back to the landing page

Potential Challenges

- Unforeseen challenges with entity sets/relations requiring a redesign
    - Complete initial framework early to allow for extra time if required
- Mistakes in implementation resulting from miscommunication.
    - Provide comments and documentation for code
        - Specify plans before implementation