Design Document for Congregation and Venue Management System

Overview

This document outlines the design of a system that manages **Congregations**, **Programs**, and **Venues**. The key classes and their interrelations are described to provide a high-level understanding of how these components interact within the system.

Class Structure and Relationships

CongregationManager

The `CongregationManager` class is responsible for managing multiple `Congregation` objects. It provides the functionality to add, delete, and display congregations, as well as manage their associated programs. It also facilitates the reservation and freeing of venues for specific programs within congregations by interacting with the `VenueManager`.

- **Manages**: `Congregation`
- **Interacts With**: `VenueManager` for venue reservations.

Congregation

`Congregation` is an abstract base class that defines common attributes and behaviors for various types of congregations (e.g., `Conference`, `Games`, `Concert`, and `Convention`). Each `Congregation` contains a list of `Program` objects. Subclasses of `Congregation` define specific types of congregations, such as a conference or concert.

- **Contains**: `Program`
- **Inherits**: This class is inherited by `Conference`, `Games`, `Concert`, and `Convention`.

Program

The `Program` class represents an event or activity associated with a congregation, such as a conference talk or a concert. A `Program` can have one or more `Reservation` objects, representing the venues reserved for that program during a specified time.

- **Contains**: `Reservation`
- **Belongs To**: `Congregation`

Reservation

The `Reservation` class represents a reservation of a `Venue` for a particular `Program`. It includes information about the reserved venue, the program it is reserved for, and the start and end dates of the reservation.

- **Belongs To**: `Program`, `Venue`
- **Contains**: `Date` for reservation start and end times.

VenueManager

The `VenueManager` class is responsible for managing a list of `Venue` objects. It provides the ability to add, delete, and display venues. When a congregation reserves a venue for a program, the `VenueManager` is used to find and manage the venues.

- **Manages**: `Venue`

Venue

The `Venue` class represents a physical venue where programs can take place. It contains information about the venue's location, capacity, and the list of `Reservation` objects for programs using that venue. Different types of venues (e.g., `Hotel`, `ConcertHall`, `ConventionCenter`, `Stadium`) inherit from the `Venue` class.

- **Contains**: `Reservation`
- **Located At**: `Location`
- **Inherits**: This class is inherited by `Hotel`, `ConcertHall`, `ConventionCenter`, and `Stadium`.

Location

The `Location` class represents the geographical details of a venue, including address, city, state, postal code, and country. A `Venue` has one `Location` that describes where it is situated.

- **Belongs To**: `Venue`

Date

The `Date` class represents a date object used for start and end times in both `Program` and `Reservation`. It is used to compare and calculate time durations within the system.

- **Used By**: `Program`, `Reservation`, `Congregation`

Interactions

1. **Congregation and Program**: Each `Congregation` contains a list of `Program` objects. Programs represent the events held within that congregation, and each program can be managed (added, deleted, shown) by the `CongregationManager`.
2. **Program and Reservation**: A `Program` can reserve one or more `Venues` by creating a `Reservation`. This reservation specifies the venue, the program, and the dates during which the program will take place.
3. **VenueManager and Venue**: The `VenueManager` is responsible for managing a list of `Venue` objects. It handles venue additions, deletions, and the search for available venues when programs need to be scheduled.
4. **Venue and Location**: A `Venue` has a `Location` object that stores geographical information about the venue.
5. **Reservation and Venue/Program**: The `Reservation` object serves as a bridge between the `Venue` and the `Program`. It associates a specific venue with a program for a particular date range.