**Features of system:**

1) The system identifies user (principal or other school persons (Teacher, Student, Coach, etc.)

2) The system supports for the addition of new rooms and their availability to be added to the calendar

3) The system enables the administrator to edit the availability of the room (Time, disable the room all together)

4) The system allows users to view their own calendar featuring their requests and approved bookings

5) The system provides the ability for users to submit booking requests (Room ID, time, date they desire)

6) The system enables users to cancel previous sent requests and cancel approved bookings

7) The system supports the ability for the administrator to view booking requests users have sent

8) The system enables the administrator to approve, deny requests or cancel bookings

9) The system allows users to view his/own calendar showing status of requests and current bookings

10) The system supports the feature of sending notifications to users about the status of their booking requests, such as being received by the system, approved, denied or later canceled.

**Additional use case:**

Principal retroactively denying approved bookings due to higher priority event being requested

1. Principal identifies self

2. Principal accesses system

3. Principal opens calendar of bookings

4. Principal cancels previous booking

5. Principal approves the higher priority event

6. System sends notification to user of canceled booking

7. System sends notification to user of new priority booking

**Traceability Matrix:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 | F10 |
| U1 | X | X | X |  |  |  |  |  |  |  |
| U2 | X |  |  |  | X |  |  |  |  | X |
| U3 | X |  |  | X |  |  |  |  | X |  |
| U4 | X |  |  | X |  | X |  |  |  | X |
| U5 | X |  |  |  |  |  | X | X |  | X |
| U6 | X |  |  |  |  |  | X | X | X | X |

**Possible classes for system and their attributes:**

Room class: This class is used for making new rooms to add to the system

Private string roomID

Private double roomAvailability

Private Boolean isBookable

Public String getRoomID

Public void setRoomID

Public double getAvailiability

Public void setAvailability

Public Boolean getBooking

Public void setBooking

Booking class: This class contains the information of a specific booking

Private User user

Private Room room

Private double timeBooked

Private Date dateBooked

Public User getUser

Public Room getRoom

Public double getDuration

Public Date getDateBooked

Public void setUser

Public void setRoom

Public void setTimeBooked

Public void setDateBooked

User class: This class contains information about a user

Private string name

Private string username

Private string emailAddress

Private string permissions

Private System system

Public void requestBooking

Public void cancelBooking

Public Booking getBooking

Public String getName

Public void setName

Public String getUserName

Public void setUserName

Public String getEmailAddress

Public void setEmailAddress

Public String getPermissions

Public void setPermissions

Public System getSystem

Public void setPermission

Principal class:

Public void addUser

Public void addRoom

Public Booking approveBooking

System class

:

Private ArrayList<Users> Users

Private ArrayList<Room> Rooms

Private ArrayList<Booking> bookings

Public void sendEmail{}

Public User addUser

Public void removeUser

Public Booking addBooking

Public void removeBooking

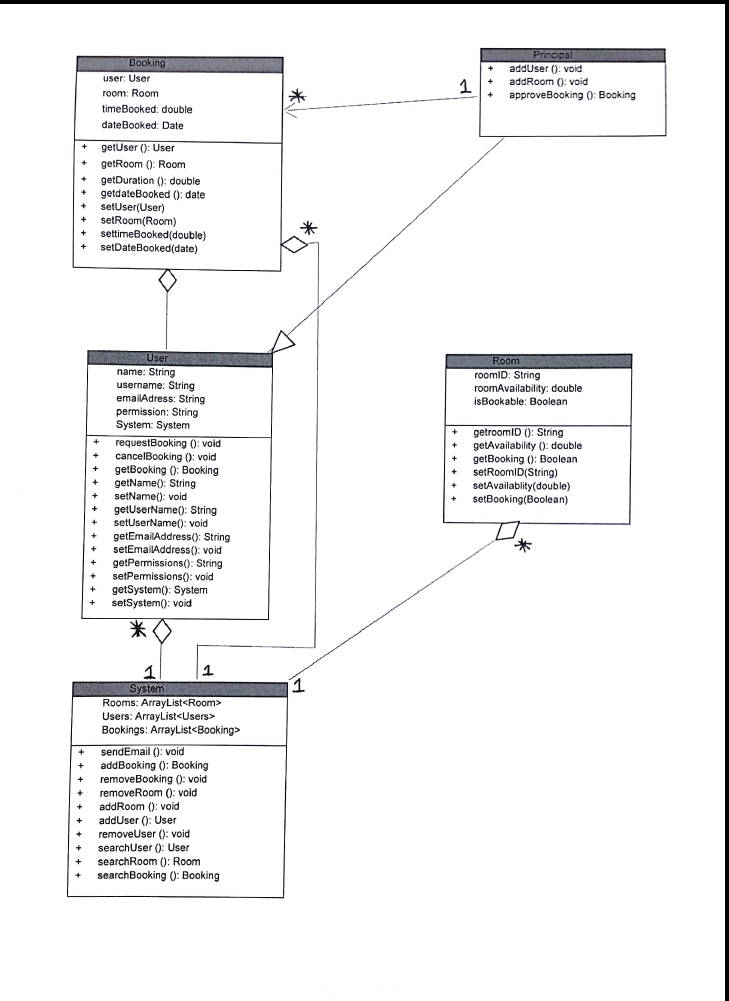
Public Room addRoom

Public void removeRoom

Public User searchUser

Public Room searchRoom

Public Booking searchBooking



**Modules:**

Users – This module will store our User and Principal classes since User and Principal revolve around the same idea of actors controlling the System.

Main – The Main module will be where the System class is stored because the System is the information expert and will be the central entity for the rest of the classes.

Schedule – The schedule module will group the Booking and Room classes. The reason for why is because they are both integral components for requesting space inside a schedule.