Assignment 2: Software Implementation

Alia Lootah

College of Interdisciplinary Studies, Zayed University

ICS220 - 22873 Programming Fundamentals

Professor Kuhail

April 15, 2023

**UML Class Diagram and Description:**

**Classes and Attributes:**

1. DentalBranch
   * address (private)
   * phoneNumber (private)
   * manager (private)
2. Staff
   * name (private)
   * id (private)
   * title (private)
3. Manager (inherits from Staff)
4. Receptionist (inherits from Staff)
5. Hygienist (inherits from Staff)
6. Dentist (inherits from Staff)
7. DentalService
   * serviceName (private)
   * serviceCost (private)
8. Patient
   * name (private)
   * id (private)
   * phoneNumber (private)
9. Appointment
   * appointmentID (private)
   * date (private)
   * time (private)
10. Bill
    * billID (private)
    * totalCost (private)
    * VAT (private)

**UML Class Diagram:**

|  |
| --- |
| DentalService |
| -serviceName: str  \*  -serviceCost: float |

|  |
| --- |
| DentalBranch |
| -address: str  -phoneNumber: str  -manager: Manager |
| +add\_staff(staff: Staff)  +add\_patient(patient: Patient)  +add\_service(service: DentalService) |

\*

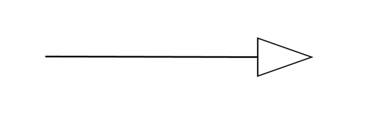
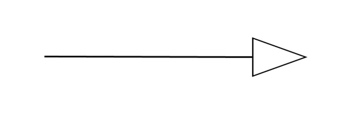
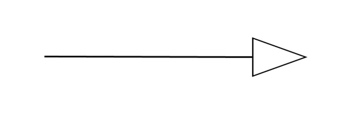
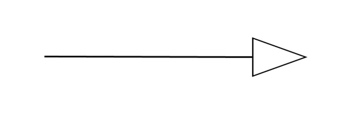
|  |
| --- |
| Patient |
| -name: str  -id: str  -phoneNumber:str |
| +book\_appointment(appointment: Appointment) |

\*



|  |
| --- |
| Manager |

|  |
| --- |
| Staff |
| -name: str  -id: str  -title: str |



\*

|  |
| --- |
| Hygienist |

|  |
| --- |
| Dentist |

|  |
| --- |
| Receptionist |

1

|  |
| --- |
| Patient |
| -name: str  -id: str  -phoneNumber:str |
| +book\_appointment(appointment: Appointment) |

\*

|  |
| --- |
| Appointment |
| -appointmentID:str  -date: Date  -time: Time  -dentist: Dentist  -services: List[DentalService] |
| +generate\_bill(): Bill |

|  |
| --- |
| Bill |
| -billID: str  1  -totalCost: float  -VAT: float |
| +display\_receipt() |

**Relationships:**

1. DentalBranch has a 1-to-many relationship with Staff, as one branch can have multiple staff members.
2. DentalBranch has a 1-to-many relationship with DentalService, as one branch can offer multiple services.
3. DentalBranch has a 1-to-many relationship with Patient, as one branch can have multiple patients.
4. Staff has a generalization relationship with Manager, Receptionist, Hygienist, and Dentist. These classes inherit from the Staff class.
5. Patient has a 1-to-many relationship with Appointment, as one patient can have multiple appointments.
6. Appointment has a 1-to-1 relationship with Dentist, as one appointment is linked to a single dentist.
7. Appointment has a 1-to-many relationship with DentalService, as multiple services can be provided during one appointment.
8. Bill has a 1-to-1 relationship with Appointment, as one bill is linked to one appointment.

The diagram shows the relationships between these classes, including inheritance, composition, and association relationship

1. **Inheritance**: Manager, Receptionist, Hygienist, and Dentist classes inherit from the Staff class. This relationship is shown by the arrows pointing from the derived classes to the base class.

* The Manager "is-a" Staff and has all the properties and methods of the Staff class.
* The Receptionist "is-a" Staff and has all the properties and methods of the Staff class.
* The Hygienist "is-a" Staff and has all the properties and methods of the Staff class.
* The Dentist "is-a" Staff and has all the properties and methods of the Staff class.

1. **Composition**: DentalBranch class has a composition relationship with the Staff class, as it has an attribute of type Manager, which is a subclass of Staff. This is represented by the filled diamond symbol next to the DentalBranch class and the line connecting it to the Staff class. This relationship indicates that the Manager is an integral part of the DentalBranch, and if the DentalBranch ceases to exist, so does its Manager.
2. **Association**:

* DentalBranch - Staff: A DentalBranch can have multiple Staff members. This association is represented by a line connecting the DentalBranch class to the Staff class, with an asterisk (\*) on the Staff side to denote the "many" part of the relationship.
* DentalBranch - Patient: A DentalBranch can have multiple Patients. This association is represented by a line connecting the DentalBranch class to the Patient class, with an asterisk (\*) on the Patient side to denote the "many" part of the relationship.
* DentalBranch - DentalService: A DentalBranch can offer multiple DentalServices. This association is represented by a line connecting the DentalBranch class to the DentalService class, with an asterisk (\*) on the DentalService side to denote the "many" part of the relationship.
* Patient - Appointment: A Patient can have multiple Appointments. This association is represented by a line connecting the Patient class to the Appointment class, with an asterisk (\*) on the Appointment side to denote the "many" part of the relationship.
* Appointment - Dentist: An Appointment has one Dentist. This association is represented by a line connecting the Appointment class to the Dentist class, with a "1" on the Dentist side to denote the "one" part of the relationship.
* Appointment - DentalService: An Appointment can have multiple DentalServices. This association is represented by a line connecting the Appointment class to the DentalService class, with an asterisk (\*) on the DentalService side to denote the "many" part of the relationship.
* Appointment - Bill: An Appointment has one Bill. This association is represented by a line connecting the Appointment class to the Bill class, with a "1" on the Bill side to denote the "one" part of the relationship.

**Assumptions:**

1. Staff can be uniquely identified by their 'id' attribute.
2. Patients can be uniquely identified by their 'id' attribute.
3. Each dental service has a unique 'serviceName' and a corresponding 'serviceCost'.
4. Appointments can be uniquely identified by the 'appointmentID' attribute.
5. Bills can be uniquely identified by the 'billID' attribute.
6. The 5% VAT is a constant value in the Bill class.