**Comp2120-Lab1 (Three Layered Use Case Realization through Sequence Diagrams)**

**The following is the business description for SMILES dental clinic system along with the Use case diagram and problem domain class model.**

The clinic has three dentists and several dental hygienists and it is in need of an information system to help administer patient records. This system does not keep any medical records. It only processes patient administration.

Each patient has a record with his/her name, date of birth, gender, date of first visit, and date of last visit. Patient records are grouped together under a household. A household has attributes such as name of head of household, address, and telephone number. Each household is also associated with an insurance carrier record. The insurance carrier record contains name of insurance company, address, billing contact person, and telephone number.

In the clinic, each dental staff person also has a record that tracks who works with a patient (dentist, dental hygienist, x-ray technician). Since the system focuses on patient administration records, only minimal information is kept about each dental staff person, such as name, address, and telephone number. Information is maintained about each office visit, such as date, insurance copay amount (amount paid by the patient), paid code, and amount actually paid. Each visit is for a single patient, but, of course, a patient will have many office visits in the system. During each visit, more than one dental staff person may be involved in the patient’s treatment. For example, the x-ray technician, dentist, and dental hygienist may all be involved on a single visit. In fact, it is even possible that more than one dentist may be involved with a patient, since some dentists are specialists in such things as crown work. For each *staff member does procedure in a visit* combination (many-to-many) detailed information is kept about the procedure. This information includes type of procedure, description, tooth involved, the copay amount, the total charge, the amount paid, and the amount the insurance company denied.

Finally, the system also keeps track of invoices. There are two types of invoices: invoices to insurance companies and invoices to heads of household. Both types of invoices are fairly similar, listing each visit, the procedures involved, the patient copay amount, and the total due. Obviously, the totals for the insurance company are different from the patient amounts owed. Even though an invoice is a report (printed out), it also maintains some information such as date sent, total amount, amount already paid, amount due and also the total received, date received, and total denied. (Insurance companies do not always pay the full amount they are billed for.)

The receptionist keeps track of patient and head of household information. He/she will enter information about the patients and head of household. He/she will also keep track of office visits by each patient. Patient information is also entered and maintained by the office business manager. In addition, the business manager maintains the information about the dental staff.

The business manager also prints the invoices. Patient invoices are printed monthly and sent to the head of household. Insurance invoices are printed weekly. When the invoices are printed, the business manager double-checks a few invoices against information in the system to make sure it is being aggregated correctly. The business manager also enters the payment information when it is received.

Each member of the dental staff is responsible for entering information about the dental procedures that he/she performs.

The business manager also prints an overdue invoice report showing heads of household who are behind on their payments. Sometimes dentists like to see a list of the procedures they performed during a week or month, and they can request that report.

**SMILES: Use case diagram**

 **SMILES: Domain class diagram**



1. **Develop the first cut for the design class diagram based on the provided information (simply add attribute visibility and data types)**
2. **Develop the first cut sequence diagram for the use case *“Record Office Visit Information” {add the classes as object lifelines from class diagram}***
3. **Add the view and data access layers to the above diagram to create a three layered use case realization for the above mentioned use case.**
4. **Update the design class by adding the methods identified thru Use case realization (VP does it automatically)**
5. **Develop a package diagram showing classes in the view, domain, and data access layers.**