**INFO8000 Assignment 3 Section 04 Team 05**

*By signing below, I certify that I have fully complied with the INFO8000 Code of Honor.*

Student 1: Namita Amgain\_\_\_\_\_\_\_\_\_\_\_\_NA\_\_\_\_\_\_\_\_(Done Task 3)

Student 2: Basavraj Jaliminche \_\_\_\_\_\_BJ\_\_\_\_\_\_\_\_\_\_\_\_(Done Task 2 and Task 3)

**Task 1: Domain Classes and Associations**

##  **Answer the Chapter 4 Questions** (based on textbook page 127)

* List the domain classes and their attributes that should be included in the **Social Networking subsystem**. Be creative and add those you think should be included to make the system useful and appealing. *You do not have to submit this list.*
* Based on the domain classes you identified, draw a **Domain Class Diagram (DCD)** showing domain classes with attributes and associations with multiplicity. Start your DCD by adding a **Student** class.

**![Chart, box and whisker chart

Description automatically generated]()**

**Task 2: More Domain Classes and Associations**

 In Assignment #2, you used the event decomposition technique to create an **Event Table** and drew the **Use Case diagram** for the *Student Booking subsystem*. List the domain classes and their attributes that would be required to support the *Student Booking subsystem* as described in the text on **Chapter 3, page 91, question #1**.

 Based on the list of classes and attributes you identified, draw a **Domain Class Diagram (DCD)** showing the domain classes & associations along with their attributes and multiplicity.

 Since there will be some overlap with the classes to support the Social Networking and the Student Booking subsystems, create a **single combined class diagram** for both subsystems.

* Resolve duplicate classes, if any (e.g., Student instead of Traveler)

**Task 3: State Machine Diagram**

The **Vacation Request Approval** lifecycle begins when an employee creates a new vacation request within the **Vacation Tracking System**. It starts with an **Open** state.

1. When in an **Open** state, the employee can make changes to the vacation start date, end date and description. The employee can save the recent changes and it remains in the **Open** state.
2. When the employee submits the vacation request, the system sends an email to the immediate supervisor with the vacation details and a link to launch the **Vacation Tracking System**. Then, the **For Approval** state begins.
3. As soon as the supervisor launches the Vacation Tracking System with the vacation request details displayed on screen, the vacation request moves to the **Under Review** state. The supervisor can make a decision right away or decide later.
4. The supervisor can decide in three ways, each decision comes with an email notification to the employee and supervisor.
5. The request is **Declined** and the lifecycle ends.
6. The request is **Approved** and the employee can now plan a vacation.
7. The supervisor make a *suggested start date and end date* and returns the vacation request back to the employee. The vacation request moves to the **For Revision** state.
8. When the vacation request is either at the **For Approval** or **Under Review** state and the *vacation start date* has reached today’s date, the vacation request is **Expired** (after providing an email notification to the employee and supervisor) and the lifecycle ends.
9. When in the **Approved** state, the supervisor can generate a report of ***all approved vacation requests*** with the following columns: employee name, approved start date and end date. The report will only display approved requests that has a vacation end date that is on or before today’s date. After the report is generated, it remains in the **Approved** state.
10. When the employee receives an email with the *suggested start date and end date* (**For Revision** state), there are three possible outcomes (each with an email notification to the employee and supervisor):
    1. Employee accepts the *suggested dates* and the original vacation start date and end date are updated automatically. The vacation request moves to the **For Approval** state.
    2. Employee cancels the vacation request. It moves to the **Cancelled** state and the lifecycle ends.
    3. When the *suggested start date* has reached today’s date, the vacation request is **Expired** and the lifecycle ends.

**Diagram

Description automatically generated**

|  |  |
| --- | --- |
| **State Name** | **Transition causing Exit** |
| **Employees receive email of vacation request** | **accept** |
| **Employees receive email of vacation request** | **declined** |