



**Daffodil**  
International  
University

**Department of Computing and Information System**

**Final Examination: Spring Semester-2024**

**Program: B.Sc. in CIS**

**Course Code: CIS 323**

**Course Title: Information System ARCHITECTURE AND PLANING**

**Time: 2 Hours**

**Total Marks: 40**

*(Answer all the followings questions)*

### **Summary of the Hospital Admission Process**

This document outlines the process for managing hospital admissions, including interactions between a community doctor and a hospital doctor.

#### **Key Points:**

- **Admission Episode:** Each patient stay in the hospital is considered an "admission episode."
- **Initiation:** A community doctor refers a patient, providing the reason for admission and a planned admission date.
- **Communication:**
  - The patient receives an admission card with a copy sent to the community doctor.
  - Discharge notes are sent to the community doctor upon discharge.
  - The hospital doctor reports any planned treatment and documents it.
  - Discharge advice is provided to the patient.
- **Supervision:** A hospital doctor supervises each admission episode (and can supervise many).
- **Diagnosis & Treatment:**
  - The hospital doctor records diagnoses with codes and dates.
  - Each admission episode may involve treatment (surgery, medication, or both).
    - Surgery: date, code, and description are documented.
    - Medication: drug name, code, dose, start date, frequency, and notes are recorded.
- **Admission States:**
  - **Planned:** Created by the community doctor's referral.
  - **Cancelled Planned:** Cancelled due to changed circumstances (e.g., patient recovery). Cancellation date and reason are recorded.
  - **Delayed:** Planned admission with a delay. Delay length is documented (communicated to community doctor). May be cancelled.
  - **Activated:** Patient arrives at the hospital. Admission date is recorded.

- **Discharged (Treatable):** Admission with a treatable diagnosis. Discharged upon treatment completion.
- **Discharged (Non-Treatable):** No diagnosis or untreatable diagnosis. Discharge marks the episode's end.
- **Doctors:**
  - **Hospital Doctor:** Supervises episodes, discharges patients, has ward/clinic and specialty info.
  - **Community Doctor:** Makes referrals; receives discharge notes and advice copies, has practice location info.
  - **Restriction:** A doctor cannot hold both roles simultaneously.

This system provides a clear structure for managing patient admissions, communication between doctors, and recording relevant medical information.

|    |    |  |            |                      |
|----|----|--|------------|----------------------|
| 1. |    | Provide specific examples related to classes, objects, encapsulation, and any other relevant OOP principles.   | [Marks-5]  | CLO-1<br>Level-1     |
| 2. | a) | Analyze the use of Object-Oriented Analysis and Design (OOAD) techniques over module-based programming in the development of the Hospital Admission Process.                                       | [Marks-5]  | CLO-2,3<br>Level-5,1 |
|    | b) | Create An Use case diagram for the Hospital Admission Process.   | [Marks-5]  |                      |
| 3. |    | Create an advanced Unified Modeling Language (UML) diagram for a portion of the Hospital Admission Process. Illustrate the relevant classes, associations, and behaviors along with class diagram. | [Marks-10] | CLO-3<br>Level-5     |
| 4. |    | Based on the UML diagram describe how you would construct the program (code in java). Include key elements such as class Implementations, methods, and interactions between objects.               | [Marks-10] | CLO-3<br>Level-6     |
| 5. |    | Analyze two well-known design patterns that could be applied in the implementation of the Hospital Admission Process.  | [Marks-5]  | CLO-4<br>Level-3     |

Course: ISAP

code :323

Mark:  $5 \times 3 = 15$

Test 😊 1

1. Explain term software crisis with the respect to stakeholders of software team.
2. Explain human communication with example.
3. Write four pillar of OOP with example0



# Daffodil International University

Faculty of Science & Information Technology

Department of Computing and Information System

Final Examination, Fall-2024

Course Code: CIS-323, Course Title: INFORMATION SYSTEM

ARCHITECTURE AND PLANING

Level: 2 Term:2

Exam Duration: 2 Hours

Marks: 40

## Answer ALL Questions [Optional]

*[The figures in the right margin indicate the full marks and corresponding course outcomes. All portions of each question must be answered sequentially.]*

**Application Story:** A Fitness Companion

**App Name:** Fit Buddy

**Target Audience:** Fitness enthusiasts of all levels, from beginners to seasoned athletes.

**Story:** Imagine a world where your fitness journey is personalized, motivating, and always within reach. Fit Buddy is your ultimate fitness companion, designed to empower you to achieve your health and wellness goals.

**Key Features:**

✓ **Personalized Workout Plans:** Tailored workouts based on your fitness level, goals, and preferences.

**Real-time Workout Tracking:** Monitor your progress with detailed statistics, including calories burned, heart rate, and workout duration.

**Diverse Workout Library:** Access a vast collection of workouts, from yoga and Pilates to high-intensity interval training (HIIT) and strength training.

✓ **Nutrition Tracking:** Log your meals, track macronutrients, and receive personalized nutrition advice.

**Community and Motivation:** Connect with like-minded individuals, share your achievements, and join challenges to stay motivated.

**Expert Guidance:** Access expert tips, articles, and videos to enhance your knowledge and optimize your fitness routine.

**User Journey:**

### 1. Onboarding:

- Create a personalized profile, setting fitness goals and preferences.
- Complete a fitness assessment to determine your starting point.

### ✓ 2. Workout Planning:

- Browse the workout library or let FitBuddy generate a customized workout plan.
- Set reminders for upcoming workouts.

### 3. Workout Execution:

- Start a workout and track your progress in real-time.
- Receive audio cues and visual guidance during workouts.

### 4. Post-Workout Analysis:

- Review detailed workout statistics and insights.
- Share your achievements with the community.

### ✓ 5. Nutrition Tracking:

- Log meals and track macronutrients.
- Receive personalized nutrition advice based on your goals.

**6. Community Engagement:**

- Join fitness challenges and compete with friends.
- Share your progress and motivate others.
- Connect with experts and fitness enthusiasts.

**Brand Voice:**

- **Encouraging:** Empowering users to reach their full potential.
- **Supportive:** Providing constant motivation and guidance.
- **Informative:** Sharing expert knowledge and tips.

By combining cutting-edge technology with a user-centric approach, Fit Buddy aims to revolutionize the way people approach fitness. With Fit Buddy, you'll have a dedicated personal trainer, nutritionist, and community of support, all in the palm of your hand.

|    |    |   |      |      |
|----|----|---|------|------|
| 1. | a) | A real-world project has many people with a stake in the project. Explain term "The Software Crisis" with respect to stakeholder of Fit Buddy.  | [4]  | CO 1 |
|    | b) | What is Analysis, Design? why do we use them?   | [2]  |      |
|    | c) | Differentiate between problem statement, problem domain, project scope.   | [2]  |      |
| 2. | a) | Draw a UML class diagram for Fit Buddy, illustrating the relationships between classes.   | [6]  | CO2  |
|    | b) | Explain NLA is tools define class.  | [2]  |      |
| 3. | a) | Draw a UML Use Case diagram for Fit Buddy. Describe one of Use case using template, UI prototype and collaboration.   | [10] | CO3  |
| 4. | a) | You are tasked with designing a new feature for an existing Java application. Write the tips to effectively apply design patterns in your Java projects .   | [3]  | CO4  |
|    | b) | Refactoring in Java: A Guide to Improving Code Quality, Why?  | [3]  |      |
| 5. | a) | What is the minimal dependency of spring-boot application. write the structure of a spring-boot application. Create following component with example (One of class)<br>1. Property file<br>2. Controller<br>3. Repository<br>4. Service | [8]  | CO4  |

QUESTIONS  
get in mind

Quiz 1

$5 \times 3 = 15$

Time :45

Asap

1. What is Analysis & What is Design? why do we use them?
2. Explain term The Software Crisis with the stakeholders of software team.
3. Write Overview of OO concepts?
4. Why do you use collection framework?
5. Write the different between comparable and comparator.