Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT)
Gandhinagar, Gujarat.

MSc (IT) Semester III Final Examination
IT619: Design of Software Systems
Duration: 3 hours

Date: Nov 28, 2023

Max. Marks: 100

Instructions:

- All questions are compulsory.
- Figures to the right indicate full marks.
- Do not insist about interacting with your course instructor(s) during the examination.
- Right understanding (without any assistance) of questions is part of your solution.

Q.1. Answer the following questions briefly. **(20 x 2) = 40**

a) Define: Temporal cohesion, Procedural cohesion.
b) In the context of Dell case study, what do you understand by rationalization? What is done to achieve it?
c) How could good software design assist in addressing supportability issues?
d) Define application usability. List two key issues related to it.
e) List popular software architecture frameworks. Summarize key guidelines to use them.
f) List key responsibilities of a software architect.
g) What lessons app-tech start-ups should remember from the Netscape-Microsoft tussle?
h) Summarize difficulties in describing architecture significant requirements (ASRs).
i) What do you understand by the term data architecture?
j) List key challenges in software architecture evaluation.
k) Summarize elements of UML state model.
l) Summarize key issues related to systems availability in web apps.
m) What is contained in a timesheet? How is that information useful?
n) Define interoperability giving your understanding of the notion on boundary blurring.
o) Summarize OCL collection types.
p) Visualize ivory tower anti-pattern using pie-chart. How software architects could avoid it?
q) How to make technical debt more visible?
r) Summarize key rules to construct data flow diagrams (DFDs).
s) Define: Product-line architecture, Execution architecture.
t) List key issues and related solutions on maintainability.

Q.2. Answer the following questions in detail. **(5 x 3) = 15**

a) Narrate coupling & its types in context of software design summarizing what each type means.
b) How are enterprise, segment, and solution architectures related to each other?
c) How are Type I and Type II technical debt (TD) different? Give key reasons to incur TD.
d) How could IT companies avoid getting software architects trapped into different anti-patterns?
e) Summarize problems addressed by abstraction-occurrence design pattern using suitable design models.

Q.3. Visualize following concepts using suitable diagram(s) for each of the following. **(3 x 5) = 15**

a) All possible elements of a typical UML sequence diagram with a short comment on each of the elements in suitable UML notes.

b) Typical scenarios targeting quality attributes during software architectural design.

c) Model depicting assets, threats, risks, counter measures, and vulnerabilities while considering system security.

Q.4. Give suitable design models as directed for each of the following. **(5 x 6) = 30**

a) For an online shopping application, design **UML class diagram** narrating typical relationships, attributes, methods, etc. of its **Order System**. Assume that we have classes like ***Order, Order Item, Customer, Order Detail, Payment,*** etc. You may add more of your classes, if needed.

b) Design a higher level **UML system sequence diagram** (SSD) (run-time objects EXCLUDED) for an online shopping application considering functionalities for the following scenarios.

- **Login,**
- **Add Item to Cart,** and
- **Item Order.**

Assume three key objects namely ***Customer, Web Portal*** (for shopping app) and its ***DB Server***.

c) Design a **UML activity model** for an online shopping application considering necessary flow for adding items to the cart till checkout giving the customer a pleasant shopping experience.

d) Design **UML use-case model** for an online shopping application considering the following.

- Top-level use-cases: ***View Item, Make Purchase, Client Register, Check-Out.***
- Actors: ***Customer*** (New, Registered)

Assume that ***View Item*** use-case could have related optional use-cases like ***Search Item, Add to Cart, Add to Wishlist*** and much more. Also, ***Check-Out*** use-case could have key use-cases related to single sign-on (SSO), authentication, credit card/UPI/net-banking payment, etc.

e) Represent General-Hierarchy, Player-Role, and Adapter design patterns using suitable UML diagrams.

Mammals

Reusability

Dynamic property

-x-x-x-