What is SW?

Computer programs & associated documentation.

S/w may be developed for a particular customer or for a general company.

7/8/18

→ SW associated documentations:

↳ User Manual

↳ Installation Guide

↳ Policy

↳ Test sequence → Integration level, Unit level, System level

→ Engineering is activity that uses practices that make it independent of people

→ Quality Attributes expected of s/w

1) Reliability :→ o/p is ✓ for ✓ i/p, works as expected. system not crash for wrong i/p
   Eg. → try-catch block in JAVA
2) Extensibility
3) Performance
4) Usability
5) Security
6) Availability
7) Portability

1) → One measure of Reliability is MTF → Meantime B/w Failures

2) → Info. needs change.

   It is motto of SWE

   ↳ Process of modifying a SW system or component after delivery, to correct faults, enhances functionality

→ MULTICS : One of the ideal O.S (MIT)

↓ opp.

UNIX

PAGE NO.:
DATE: / /

improve performances or other quality attributes, or adapt to a changed environment

3) → avg. turnaround time taken by the s/w for each logically diff. type of service under normal & worst-case scenarios

→ Look ahead computers

Performance on basis of

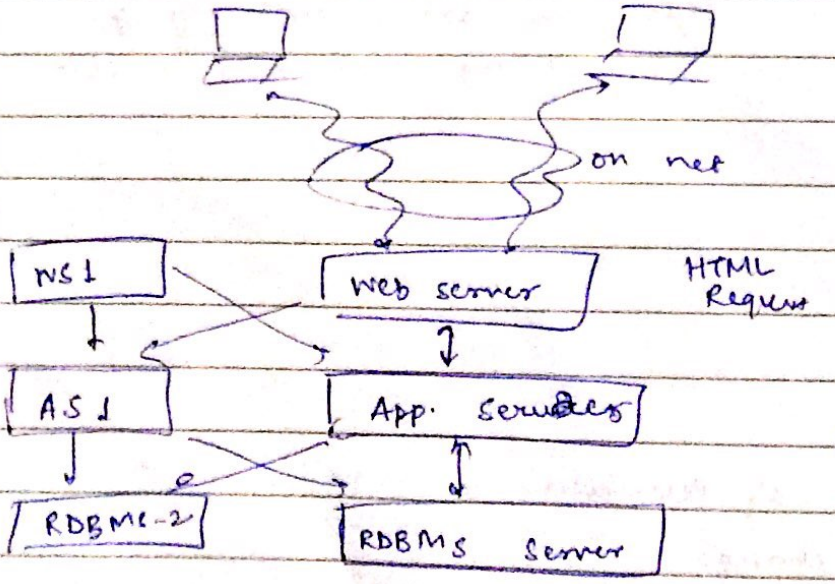
→ Response Time

→ Throughput

4)

5) Unauthorized person shouldn't access my system → Authentication

6) Availability : When transx^n cancels → rollback occurs but the system keep on running



→ Can ↑ throughput

→ can use priority

Teacher's Signature

SAP, Siebel, MS OS, Oracle RDBMS

Pg No: 01
DATE: / /

Availability refers to % time that such as a complex SW package is cont. available measured over a 100 units of time

2) Portability: degree of ease with which I can install & use that SW in different environments

eg: Java based programs run on all OS that support JVM without modifying even 1 line of code.

→ Quality Attributes expected of SWS :

1) Productivity: No. of pieces/op produced per day without defect

9/8/18

S/W Engineering Models

> Target users of SW packages

The way we design SW differs on basis of target user

Business / Scientific / Social Community

> Types of s/w packages

↳ Interactive apps: IRCTC (we have URLs) (All "sign in" apps) Amazon (Multi session)

↳ Stand-alone apps: Scripts

↳ Embedded control systems

> Target market of SW packages

Easy ↳ Single Customer

Tough ↳ Product line: for set of customers

Faculty's Signature .....

In this course

↓

Single customer Interacture Business

PAGE NO.:
DATE: / /

→ 3 main models of SWE → all are SW Development Life Cycle Models

1. Waterfall Development model : will come 1 day, tu tun, everyone ü working
2. Increment : Increment work on daily basis
3. what do we automate in business ? → Business op^n^s

⇒ Phases in the Rational Unified Process

from requirement to delivery to post-delivery

Inception → elaboration

1 → System Analysis & Design

↓

will find the inconsistency in program

2. Incremental : keep on changing the databases & classes.

3. component based development model :→

Based on systematic reuse where systems are integrated from existing components

Teacher's Signature

Automation of SW is diff from automation of calculator.

21/8/12

PAGE NO.:
DATE: / /

Software Appⁿ Architecture
Software Appⁿ Design
Design of Test cases
Requirements Eng & Analysis
Integration Testing
Coding & Unit - Testing
System Testing & Regression Testing

References:

↳ Attach Minutes of Meeting

↳ Brochure

Design constraint:

Why we we coding only 1 part (M/V/C) ?

TO know about the integration of MVC

5.1 → Imp

↳ can have peak hr & non-peak hr.

Any SRS should NOT have

1) Ambiguous sentences

Incompleteness : ↳ SBI gateway facility

↳ If something is provided to 1 see",
it should also be - " others ones

PAGE NO.:
DATE: / /

Inconsistency :

If CPAI < 7.5 then ——

If CPAI ≤ 7.5 : Inconsistency then ——

Verifiable : system should give sufficient throughput

→ Cancellation :- late fee cancellation in case of mrg, hospitalisation, ...

DFD :

↳ flow of data in the system

→ Project - 19 :→ after locked

Gradesheet → triggered by Academic Office

User :-

B Tech MSc M Tech Dual PHD

↓

CSE → 1, 2, 3, 4

↓ select

Union of all courses → Pick 1 course

flag=0 : grades unlocked

AR Academic → send reminder (APS will be given) email 2

Dean ↳ will also see it

can send another reminder

if all locked generate gradesh

Gradesheet : → commit into pdf

col → 2

1 ☑ type

Teacher's Signature

→ Parent sent e-mail :→ authenticated by Dean

Page No.:
DATE: / /

gradesheet generated (any year) branch → send e-mail that gradesheet is generated

→ AR Academic won't Lock

Db → Project ~~DB~~ + API

M — V — C

set of classes which can access db & only one which can access it (where 1 class is created for all other classes)

→ Not allowed to have any public class

We can't call M.
↳ call C (C calls) M.

C can ask M to give data.
M can only return it to C (not V)

1 M → Student Acad Record
1 M → — Medical "
1 M → — Hostel —

→ Have an API where pdf will be generated
↳ should get current sem (but have options of prev. sem also)

↓
Full Transcript
(All sem on 1 sheet)

Teacher's Signature

28/08/18

→ Entity & User tool which interact with my system

Data flow :

→ Name to name the data

29/08/18

Ques → Objects of a Project ?

Ques → Diff. b/w level 0 & level 1 & level 2 DFD ?

Ques →

↳ DFD level 0 should be same as SRS.

→ Physical DFD v/s Logical DFD

↓

used in Management

Q. There're data stores which are + nt in level 0 but - nt in L

A nice- versa. - entities ~ ~ -

Now, I check SRS & it's completely diff. from all levels.

↓

Not acceptable

Teacher's Signature.....

Level 0 DFD

leaflevel process discription

PAGE NO :
DATE : / /

1. When did OOPs come into industry (app) ?

1990's

2. What was before it ?

x ——— x ——— x ——— x ——— x ——— x ——→

→ Coarse v/s Fine Refine

↓ ↓

level 0 DFD (Not much details)

level 1 DFD

→ Tuesday : DFD level 0 & 1

Object Oriented Analysis

→ Change of focus b/w DFD & OOA

↓

~~to~~ tries to find out info. ~~about~~ arised

↓

info about business

Q. → Identify objects (business op^n's are performed on these) in your projects

19 :

Teacher's Signature.....

→ for LNMIIT ~~~~ bus service, what are the Objects ?

1) Bus - Name - Seater ?
2) Schedule (Trips)

Bus module Objects → ? ~~Residents ...~~

↓ Opⁿs ?

who operates on it ?

→ Tuesday :→ DFD, Objects
↓
level 0 → Today evening
level 1 → Sunday

6/9/18

→ Difference b/w a BPM and a Control Flow Diagram and a User-Case Activity Diagram

CFD : for small programs
→ Never write a program before a flow chart.
UCAD : opⁿs on object

→ C/C++, Java, Cobol ⇒ all are Turing Complete

•) → End Term : Case study
↓ Produce DFD - level 0, 1

↔ User Case —modified→ User Case Activity Diagrams

Class diagram

Test Cases

1 extra class on → Test case

PAGE NO:
DATE: / /

class participation (Quiz)	20%	: Best 2 out of 3 (1 Announced + 2 Surprise)
End term	30%.	
Project Report	40%.	
Project Demo	10%.	

Demo : Call individually in office
↓ create video on Lmiit id (Principles of demonstration)

DFD :
DS only → DS arrow : Problem : only writing (not reading)

Level 0 1/2 : 1 jaata with only view right : can have 1-directional flow

↳ don't have entity to directly write on disk, have to go through a s/w

→ API : we don't write, expect it to be provided, we're just using it = show as process

DS : DS1

↳ A user can access a data stor only through process

DS2 → DS1 Can put * L write 1 includes 2
(Because should have abstraction)

PAGE NO.:
DATE: / /

→ Estimating time needed to complete the project.

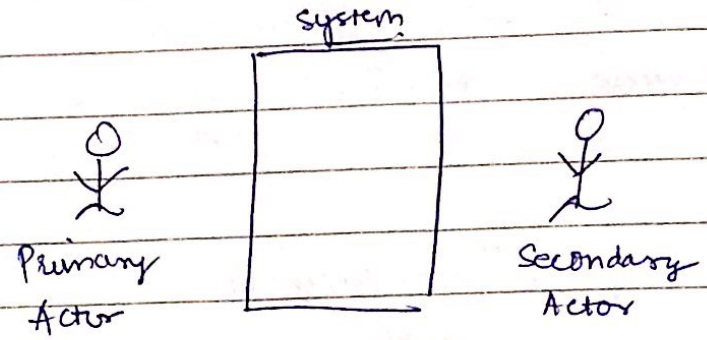
↳ How many DS needs update, how many are for "view only"?

- Entities -

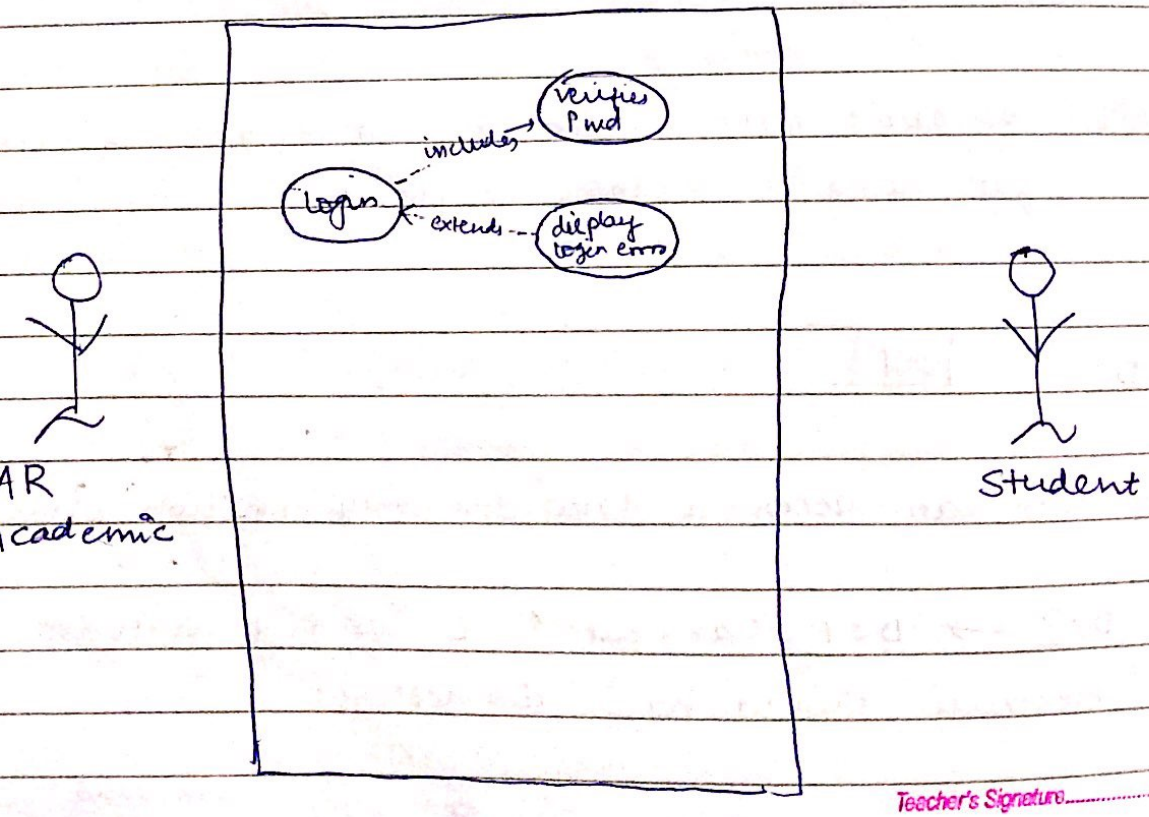
Use Case Diagram

Primary Actors : Initiates the use of the system

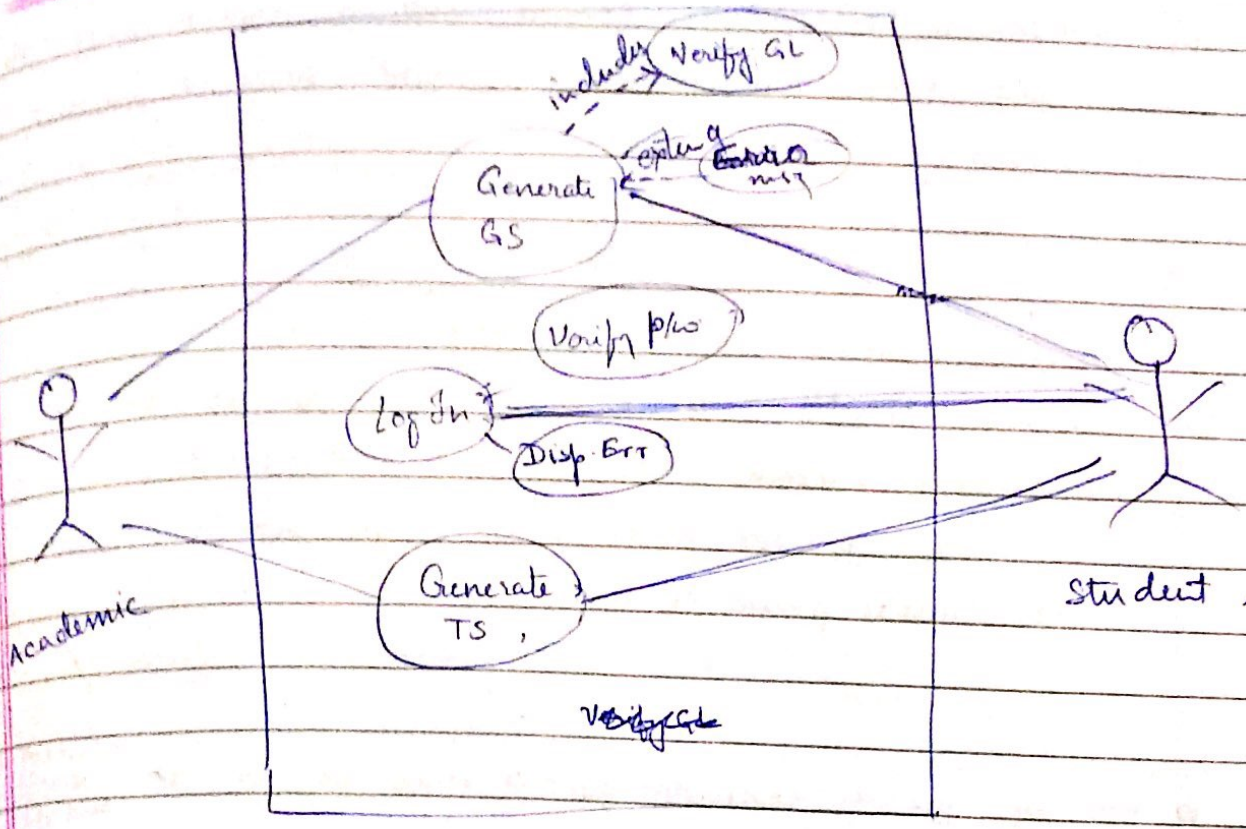
Secondary Actors : Reactionary



→



Teacher's Signature.....



11/9/18

Software Project Management

→ How to Analyse what should be price of this product (including profit) ?

Software Pricing :

Factors affecting :

1) Market Opportunity :

New organisation → quote low price

2) Cost estimate uncertainity : If an organisation is unsure of its cost estimate, it may ↑ its price by a contingency over & above its normal profit.

3) Contractual terms : given source code to developer just to modify it. → Price ↓↓ due to :

1.) No maintainance by developer

2.) Customer also has source code.

4) Requirements volatility : started with x requirements. In b/w add more requirement → Adds price to product

↓

can be done in prototype model / Iterative / Spiral Models (similar to Iterative)

↓

scope of adopting some amount of change

5) Financial Health : If I'm not able to get contract due to any reason (not un-satisfied end-user / ____), I can go for lower price & get bid.

→ Initial phase of start-ups.

- No. of people required for a time period → needed to know for budgeting of a product.

Effort Estimation Methods

Analysis Artifacts : UCDs, UCADs

→ Function-Point / COCOMO are better models. (than no. of pages in SRS)

Func^n - Point Analysis (FPA)

Take I/p, O/p, DS, ____ & categorize ~~or for~~ as #simple / #Avg / #complex → based on Project Manager's experience

	#S	#A	#C
Inputs			
Outputs			
Data Stores			
Processing Updates			
Processing Inquiries			
External Interfaces			

→ search → based on many filters

Teacher's Signature.....

Global Factors → Rank b/w 0 to 5 (any value)

Submission → 1 Word doc

Algo: Count everything & go for Σ opⁿ.

↳ these counts are then weighed / multiplied by their degree of complexity

take entire sum: Unadjusted func° points = 348

↳ adjust

↳ We have 14 global factors, each factor is ranked 0 - 5 based on SRS.

Adjustment Influence (AI): Range → 0 to 70
(14*0) (14*5)

CAF = 0.65 + 0.01 * AI

- 0.65 → AI = 0
- 1.35 → AI = 70

CAF → Complexity Analysis Factor

Adj-FP = Unadj-FP * CAF

Adj-FP → Adjusted; Unadj-FP → Unadjusted

↳ 8 Developer perform at an avg. of 10 func° points per month

420 / 10 = 42 person-month

Salary of each : 2L

Price = 42 x 2 = 84L

Time required to complete product: depends on strength of people working in the org°.

UCD :

PAGE NO.:
DATE: / /

Issues :

- Most of things are done based on past experiences
- Difficult to apply to massively distributed development cenv (offline development / Work from home / Working on-site)
- Diff. companies will calculate fp slightly diff. making comparison blw them questionable.

20/9/18

(1) Use FPA methodology to your Project & ~~collect~~ calculate

→ avg. → 10 fp per month

/10 → 42 person mon

11/10/18

Class Diagram

How to identify a class ?

Principles :

→ High Cohesion : Completely related to one piece of info & a single Responsibility

→ low Coupling

* UCD → is object-centric

starts with 1 Rectangle with name of object

2 types of ~~activities~~ actors :

① Activating actor

② Participating actor

Teacher's Signature.....

Class is responsible for single responsibility.

That's why it is highly cohesive.
all attributes are related to 1 thing

→ low coupling

→ Every NF stores a single responsibility (student details / Book details/ Book borrow details)

Relationships b/w Classes

① superclass - Subclass

Students ⟶ PG | UG

⟶ 1st Y | 2nd Y | 3rd Y | 4th Y

→ multiple Inheritance : Try to avoid
Very difficult to maintain

→ Side : Inheritance is ineffecient.
Func^n call is efficient than writing code in main)
↓
Every time I call func^n, stack is opened → takes some time
So, func^n call is expensive in case of Maintainability
↳ Main's code is more effecient.

② Aggregation Relationships (Is a part of)

1 exists as long as other exist (which 1 is part of)

If not black color (◇) it's 1 to * (1 to Many)

PAGE NO.:
DATE: / /

Eg: Hostel → has Warden, Ass Warden

Warden is an instance by himself (name, ...) ← Person of details

but belongs to BH-2

If BH-2 ✗ ⇒ Warden ✗

Class Diagrams [Submit by Sunday]

→ Attributes are only private

→ Method : Both Public and Private

Component Diagrams

→ All modules are interlinked

→ tell what you'll give others & what you need from others (API's)

Deployment Diagram

→ give clear view what is frontend / back

Tier -1 : web server
Tier -2 : App server
Tier -3 : DBMS server

} have to be in same stack otherwise, components won't work together

~~If they are all for~~

→ Till NOW, we can't call C code from Java code

Teacher's Signature.....

[Nov 8th] (London Ambulance System)
Google Scholar | Therac-25

MOM :- S/W bus
because Java can't call C, it puts call on bus. The bus will reach app^n that is actually responsible & that app will take data from bus

Software Testing

- need to check SRS also because it is the base of S/W.
- One-to-one mapping in black ~~box~~ box.

- Ultimate aim is to uncover the defects which are not in the product. (not just tell a product is faulty or not)
- Can go for testing even after deployment
- Testing tech → from whitebox or blackbox

- sometimes, follow heuristic approaches (based on experience) if don't have enough time for testing.

- Can't afford to have errors in financial trainx^s / rocket launch

- Testing is required when it's
  (1) developed
  (2) modified (If added new module, need to test interconnection b/w modules & also, new product)
  (3) re-engineered

- In testing, we go for verification & validation
  - verification: goes after every step (Are we building the product right?)
  - validation: done at end. (Are we building the right product)

Teacher's Signature .....

DATE: / /

→ Unfortunately, testing can only reveal presence of short coming not their absence

Goals:

→ Identify faults

→ Establish confidence in s/w

Black-box : most of time, on non-functionality

White - " "

→ Programmers → do debugging

→ Usability :- subjective
(Non-functionality)

Risk based Testing

Risk is expected loss attributable to the failures caused by faults remaining in the software.

2 things

① Test data

② Test cases.
+
Coll^n of Test data

Test Suite : Coll^n of Test cases

Test Repository : Coll^n of all test suits for all products

Teacher's Signature.....

What has to be tested. ?

→ All methods of all the classes & their pkgs

→ 100% ~~of~~ ~~testing~~ testing isn't possible

→ Testing should be done throughout the life cycle

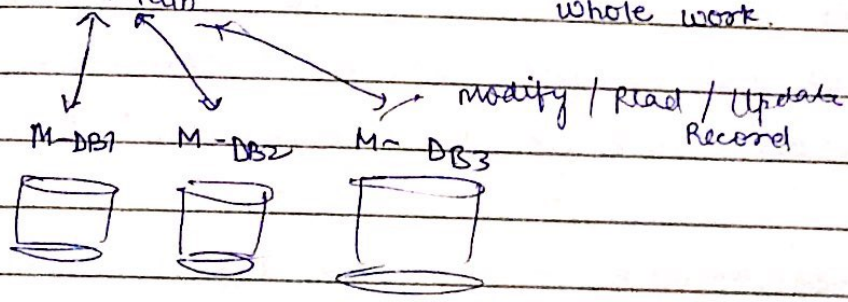
• Halting Problem :
How to guarantee that a s/w is correct

→ maintainence is only thing by which we make money ~~only~~ only after selling a software product.

→ We have :

V (iew) — Laptop Window 10

C (ontrol) — Main → C is the one that does the whole work.



Quizzes : 30% (n-1)

Project : 40% — Team Interactions
10 — 5*2

Report — SRS
— DFD
— UCD | UCAD
— FPA
— Class D + Comp. D
— TestCases + seq D

6 X 5
= 30

End-term : 30%