

# JANUARY

1 Rajab 1446 H

01

## Wednesday

8:00 16/7/25

2025

## LTS Step-In

# Day - 1

Certificare

Agile for beginners - go to link, sign in  
a mail:

10.00 g mail

**\*Enrol for free\***

1100

No need of certificate  $\rightarrow$  send screenshot

12:00 module : [send quiz score]

13.00

Why do we need software process?

14.00

## Evolving role of software:-

1

Major problem of software:- Can't create bug free environment / Software.

Software crisis - we need to overcome

Rocket  
64 bit . failure during conversion of 16 bit  
[Overflow happened.]

19

Software Engineering helps with reducing software crisis.



JANUARY

2 Rajab 1446 H

Thursday

02

2025

كانون الثاني / يناير

٢٠٢٤٦ هـ

الخميس

Software means embedded, hardware or run code in cloud. Create many test cases.

\* Software → Programs + Operating procedures + Documentation.

Maintain source code, obj. code

Documentation manuals can be of 4 types:-

- ① Analysis / Specifcatn based doc. → ~~formal~~
- ② Design Doc.
- ③ Implementation doc.
- ④ testing Doc → IMP.

① → Formal  
→ context diagrams  
→ Data flow Dia.

④ → Test data  
→ Test Results

② → flowcharts  
→ ER Diagrams

③ → Source Code  
Cross Reference

Week 1

February	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	F	S	S	M					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28

JANUARY

3 Rajab 1446 H

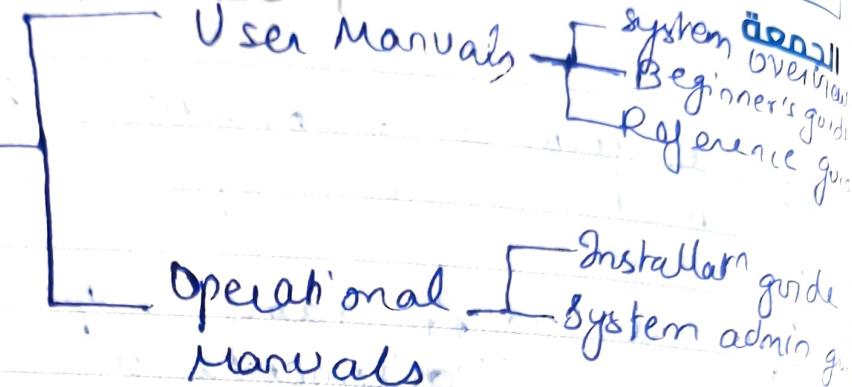
03

كانون الثاني / يناير  
١٤٤٦ - ٢٠٢٥  
ج ٣

Friday

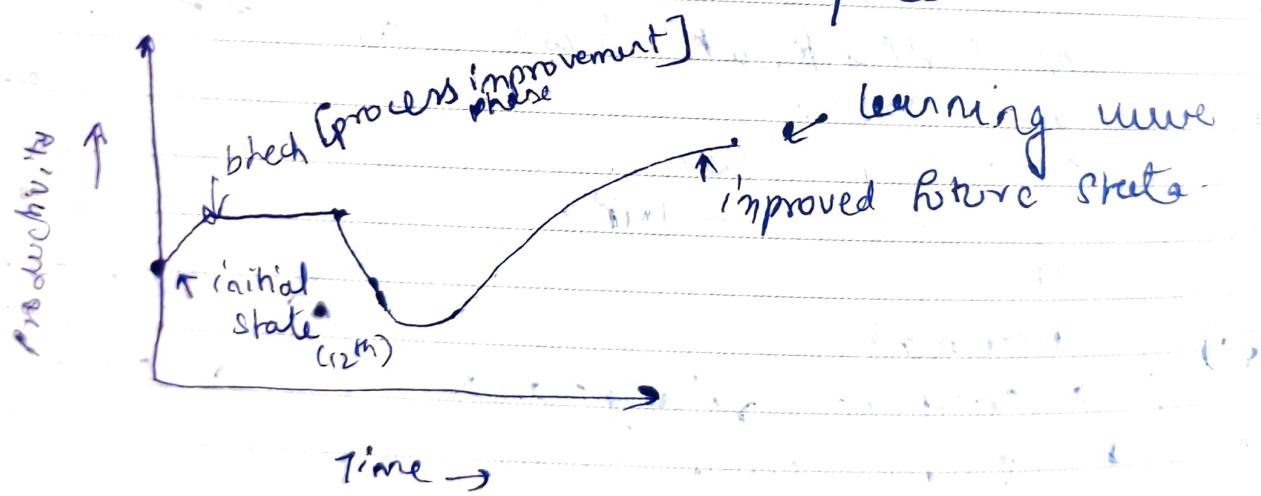
2025

Operating  
procedures



Software Process: Way in which we produce software varies from organization to organization.

We need to use best techniques -



Must make code more sustainable  
Understand problem first!

Many systems now, but y new one needed!  
Coz we need to produce a result which  
can be solved better with better features

W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	W	T	F											
١	٢	٣	٤	٥	٦	٧	٨	٩	١٠	١١	١٢	١٣	١٤	١٥	١٦	١٧	١٨	١٩	٢٠	٢١	٢٢	٢٣	٢٤	٢٥	٢٦	٢٧	٢٨	٢٩	٣٠	٣١

JANUARY  
4 Rajab 1446 H

04

كانون الثاني/يناير

٤ رجب - ١٤٤٦ هـ

Saturday

2025

السبت

8:00 and productivity.

9:00 There's costly & cheap phones. There's bigger market for cheaper phones. So if market size big; we can make same profit as expensive phones [has low/small market].

10:00 Have to know for what client we're making software for.

11:00 Deliverables / Milestones: → What is to be delivered to a client within a certain time-

12:00 Process:

13:00 Product: Understand objectives & scope

Customer:

14:00 Imp ppl while designing Software:-

- 15:00 → Good Manager → give good leadership
- 16:00 → Good works culture
- 17:00 → Put forward thoughts without any holding back.

Week 1



JANUARY

5 Rajab 1446 H

05

2025

كانون الثاني / يناير

١٤٤٦ - ٥

Sunday

اللحد

8:00

Different types of SDLC :-

9:00

Follows certain procedures to deliver project.

10:00

Starts when proj conceived & ends when it's delivered to client.

11:00

conceive  
~~create~~  
proj

→ Design → Implementation → Test phase.

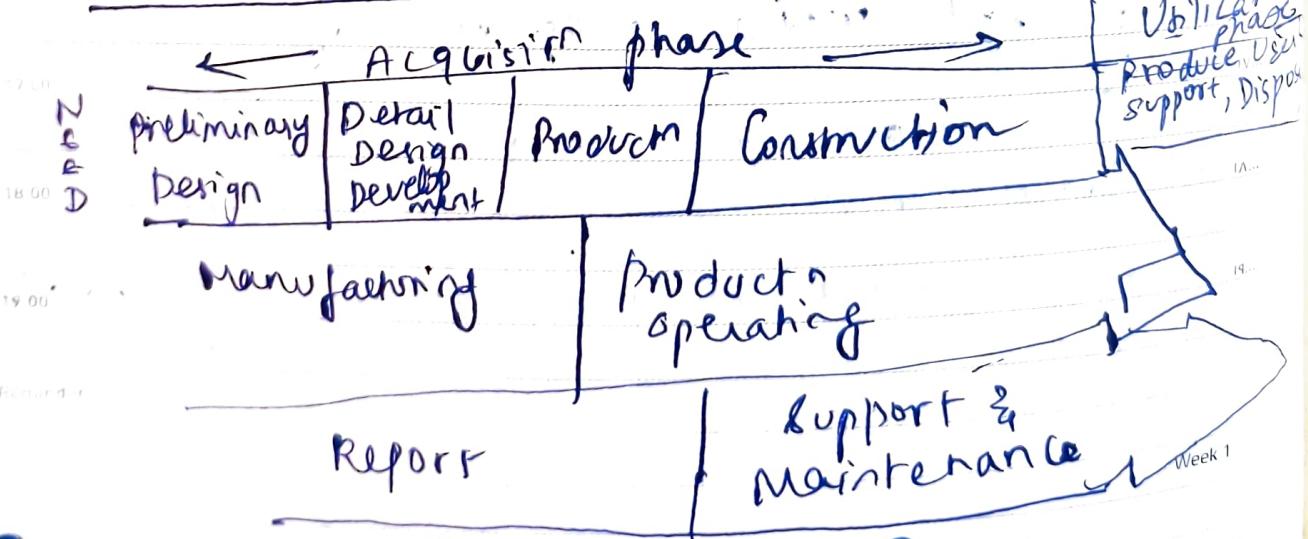
13:00

Maintainance ← Operational ← Install software  
phase in customer end

14:00

SDLC :-

General Model : Acquisition & Utilization phase



	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Wintd Mkt page

JANUARY  
6 Rajab 1446 H

## Monday

06  
2025

كانون الثاني/يناير  
١٤٤٦ هـ - رجب

العنوان

## System Retirement / Disposal

## Micellar phase

## Acquisition phase

Concept

## → Sys. Design

Design & Development  
Implementation

## ① System Complexity

Driver →

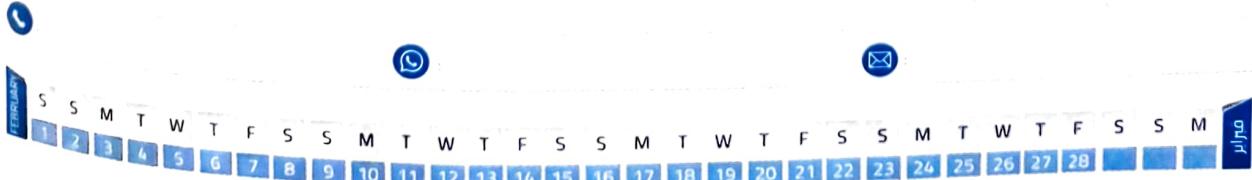
Choices → V-model / spiral model

## ② Risks & Uncertainties

Driver →

choices → Agile

Week 2



JANUARY

- Raah 1446 H

07

2025

الثاني / يناير  
١٤٤٦

Tuesday

## Waterfall Model

[ requirements ]

Stakeholders needs sys req. are gathered  
goal of this phase to understand what sys  
must do.

→ [ Design ] flow of sys.

→ [ Execution / implementation ] Implementation / coding

→ [ Verification ] Testing of component  
take place.

→ [ Deployment ] Installation

→ Sequential or linear approach

→ Document Driven

→ Rigid & Predictable -

→ limited feedback loops -

W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

JANUARY  
8 Rajab 1446 H

08

كانون الثاني/يناير

٨-١٤٤٦

Wednesday

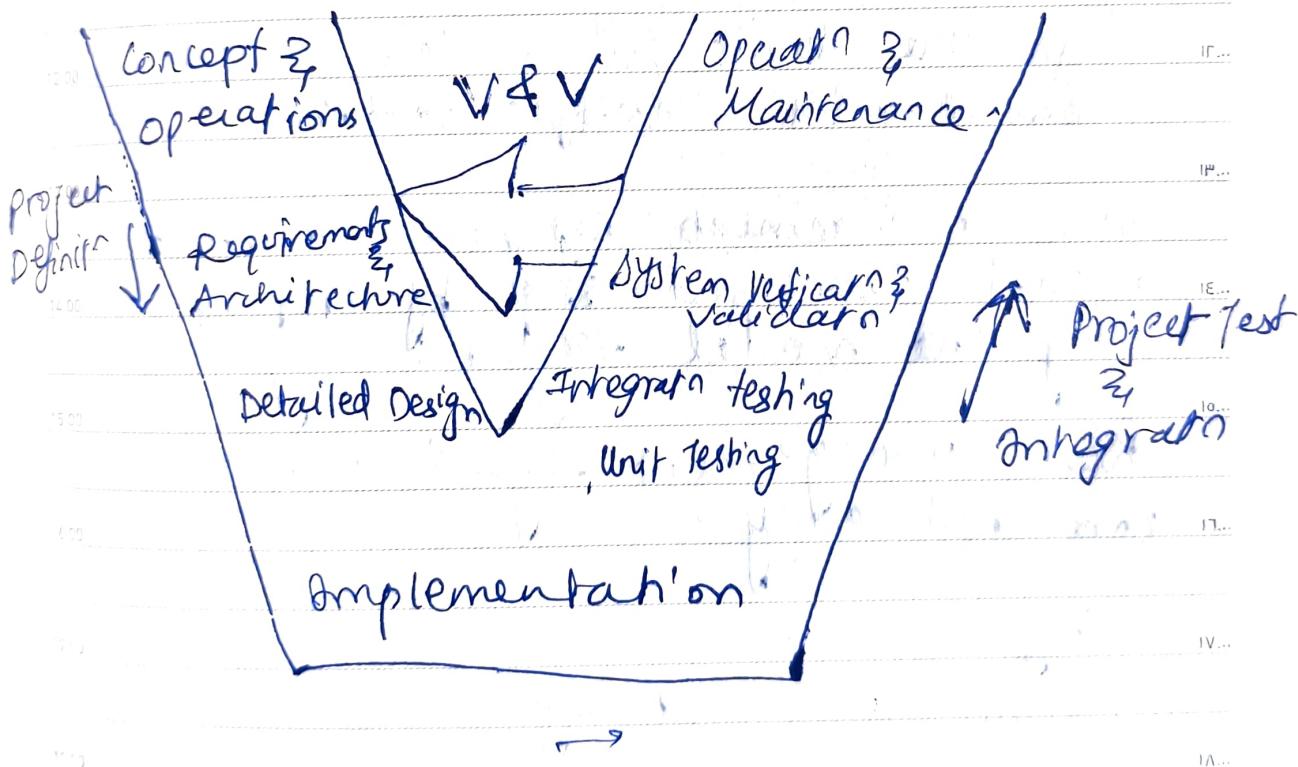
2025

clayhill

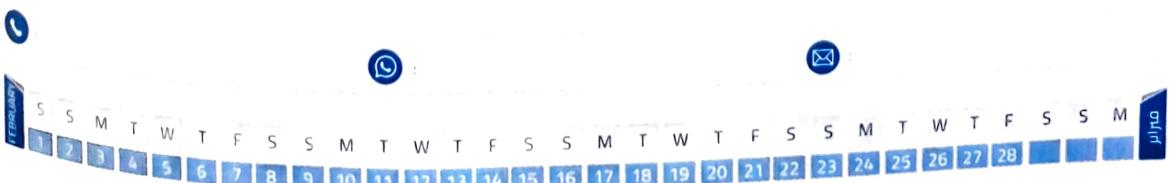
**Wednesday** 2020  
Maintenance not part of waterfall model,  
but it's provided in the end.

Fee Model :-

named after V shaped flowchart



Week 2



# JANUARY

9 Rajab 1446 H

09

2025

## Day 2 - LTTS

SDLC Models → chk ~~geeksforgeeks~~<sup>✓ better</sup>  
+ tutorialspoint

## Iterative Model

Software built incrementally through repeated cycles - multiple iterations.

Entire requirements not taken at start,  
model developed step by step & im-  
ping the model along the way.

Risk management - good. Any mistakes corrected early.

## Stages

- ① requirement gathering
  - ② Design
  - ③ implementation
  - ④ Testing
  - ⑤ Review & feedback

JANUARY  
10 Rajab 1446 H

10

كانون الثاني/يناير

١٤٤٦-رجب

Friday

2025

الجمعة

any feedback; we go back to <sup>(2)</sup> Design and correct it. After all iterations:-

## ⑥ Deployment

Adv → Early detection of issues -

Limitations → Increased cost due to repeated changing of segments -

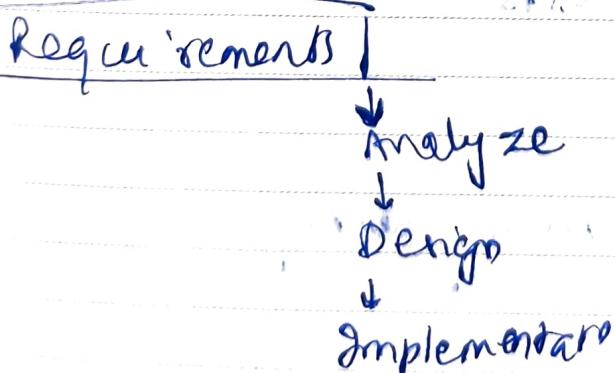
~~Only for large scale enterprise projects~~

Not ideal for smaller projects

## 14.00 Incremental Model

Persuasive approach. Each iteration

16.00 New models keeps on being released.



Release 1

## Deployment & Maintenance



JANUARY

11 Rajab 1446 H

11

2025

يناير الثاني / جانفي  
١٤٤٦ هـ

Saturday

After maintenance its Release again all steps repeated 1, then

A  
L  
D  
↓  
↓

↓  
deploy

Release 2 . . . . . Release N

Any no. of releases possible.

### ★ Agile Model

Uses Iterative & incremental approach

→ Sprints → lasts 1 week to 4 weeks

Each sprint

Needs continuous planning

Focuses on early delivery after each sprint

Bugs fixed

W	T	S	M	T	W	F	S	M	T	W	F	S	M	T	W	F
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3
4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6

JANUARY  
12 Rajab 1446 H

12

2025

## Sunday

## Sunday \* Cross functional team

മനസ്സിലെ വിജയം നേരിട്ട്

كانون الثاني/يناير

١٢ - رجب - ١٤٤٦

卷之三

Sunday \* 2023 Aug functional team ,

④ Spiral Model: High uncertainty [3pics]

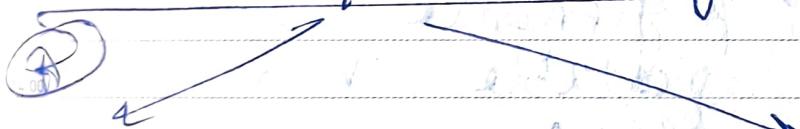
- ↳ Planning
  - ↳ Risk Analysis
  - ↳ Development
  - ↳ Assessment

chile picks  
in what's left

11:37 AM  
17/7/25

SDZ c recap  
no del :

## Needs & Requirement Analysis



## Functional Req.

## Non-Functional Req

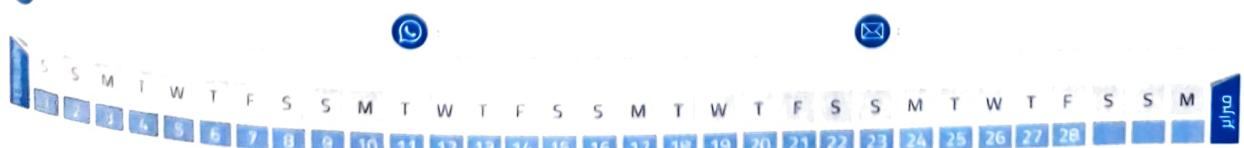
## Stakeholder requirements/needs

↓  
Goals & Objective (defined by stakeholders)

"Expected" outcome (not exact outcome but

Somewhat  
similar)

Week 2



JANUARY

13 Rajab 1446 H

13

Monday

2025

الثاني / يناير  
١٤٤٦ - ٢٠٢٥  
Week 3

8:00

(A) Functional Req → specifies capabilities of system such that it satisfies stakeholder's needs.  
Slow execn, opern, interaction of sys: takes place.

9:00

Functional → system to do → UML Diagram  
Req

10:00

Non-funct → performance is  
Req scalable, it is  
because adaptable

11:00

available  
Usability

12:00

Maintainance

13:00

Portable

14:00

Meets compliance issues

15:00

Free time

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

30:00

31:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

24:00

25:00

26:00

27:00

28:00

29:00

JANUARY  
14 Rajab 1446 H

14

2025

Tuesday

كانون الثاني/يناير

١٤٤٦ - حب رحیم

57

To design robot for surgery needs intricate planning, precision, safety over-

b XYZ med company creates robots

Who are stakeholders? → Surgeons [for debt, info, in etc]

Surgeons will review robot performance  
surgeon controls the robot. They must  
know med as well as tech. expert  
Cross functional team needed.

Nxt Stakeholder → Patient [since he is being operated on]

Hospital Admin is next stakeholders & make sure no electricity outage, make sure tech experts present

Nxt state.  $\rightarrow$  Regulatory authority [no-misuse  
of tech]

All above Stake held.

## 1 Identify Needs

What or functional in non-func. req?



JANUARY

15 Rajab 1446 H

## Wednesday

15

2025

الثاني/ينابر

b 13

11

func. Req → Robot Control Sys. interface  
~~feed~~ Haptic feedback from surgeon, proper usage of surgical instruments, imaging & visualization, safety, communication b/w robot & surgeon / tech experts

Non-FDA Req → Usability, FDA approval for all med. devices, CE Marking, ISO cert. for car to maintain quality,

## Sensing, Robot architecture, data learning

Anothee 8:

## Digital Cockpit

Stakeholders → Manufacturers, Drivers, Fleet

Manager, Regulatory Authority, Technology  
suppliers.



A horizontal digital calendar strip for the month of August, spanning from Monday, August 1st to Sunday, August 31st. Each day is represented by a blue square containing a white number. Above the calendar, there are three small circular icons: a blue phone icon on the far left, a blue sun icon in the middle, and a blue envelope icon on the far right. The days of the week are labeled as follows: W, T, F, S, S, M, T, W, T, F.

JANUARY  
16 Rajab 1446 H

16

2025

كانون الثاني/يناير

١٧- ربـ ١٤٤٦

Thursday

الخامس

Functional Requirement →

- ① Driver Display System
  - ↳ RTD → fuel, spd, engine diagnosis, navigation system
  - ↳ Customizable layout system
  - ↳ ADAS → Advanced driver assistance sys.

② UI 乞 interface?

Touchscreen easy, voice commands - hand free operation

## ③ Safety features

④ ~~Supply~~ ~~Warning~~ SOS signal any danger, driver falls asleep  
connection

## No n - Func. Reg

## ① Reliability

## ② Availability

## ② Performance

Week 3



JANUARY

17 Rajab 1446 H

Friday

17

2025

الثاني / يناير  
١٤٤٦ هـ

- ④ Usability
  - ⑤ Security
  - ⑥ Regulatory Compliance

UML → Unified Modeling Language

# Graphical modelling platform

14.00 Visualize high level pgms.

VML

1

Know requirements → Design

Building blocks are called "things".

I. Things can be structured, behavioral.

## II. Relationships b/w things

### III. Diagrams created based on the relationships

JANUARY  
18 Rajab 1446 H

18

كانون الثاني/يناير

١٨ - حب - ١٤٤٦

Saturday  
Shenaral

2025

السبت

→ class, interface, use cases,  
Components, node.

<sup>CP</sup> Behavioural  $\rightarrow$  Interact<sup>n</sup>, State Machines

Group → packaging

Annotating → Documentation, Notes

II Relationship → Dependency, Association

## II Class Diagrams

State

03

Use base

## Collaboration

*in* *n* *s*

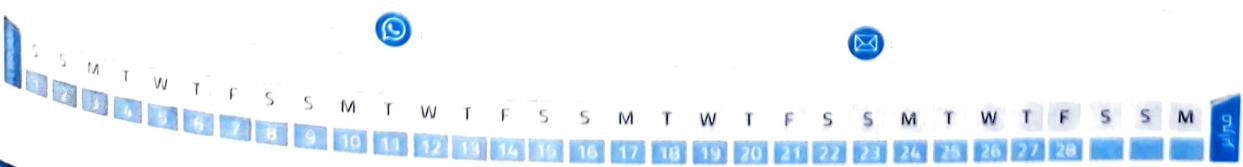
in

IV

1A...

19...

Week 3



# JANUARY

19 Rajab 1446 H

19

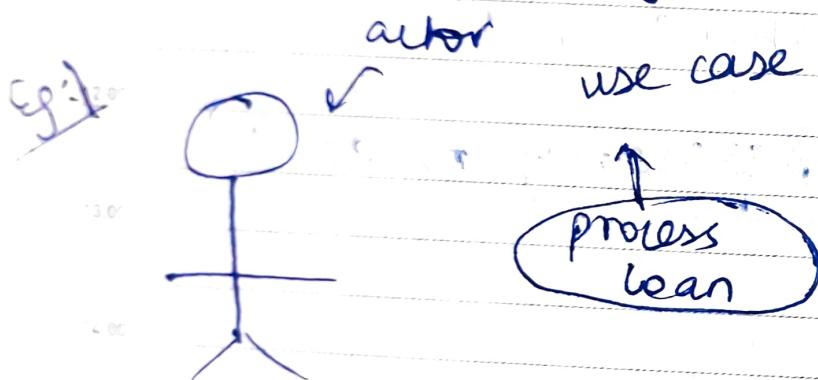
2025

الثاني /يناير

## Sunday

IMP Use Case

- Define set of sequ.
  - Actor → participant in a use case  
(anything interacting with system)



to an office  
CLO

multiple local officer messes; so access given to the LO with particular name.

EZ2 for University Registrar system  
we're developing application

A horizontal blue calendar strip showing the days of the week from Sunday to Saturday. Each day is represented by a blue square containing the day name in white. Above the calendar, there are small circular icons: a magnifying glass over Sunday, a mail icon over Monday, a person icon over Tuesday, a gear icon over Wednesday, a lightbulb icon over Thursday, a checkmark icon over Friday, and a trash bin icon over Saturday.

JANUARY  
20 Rajab 1446 H

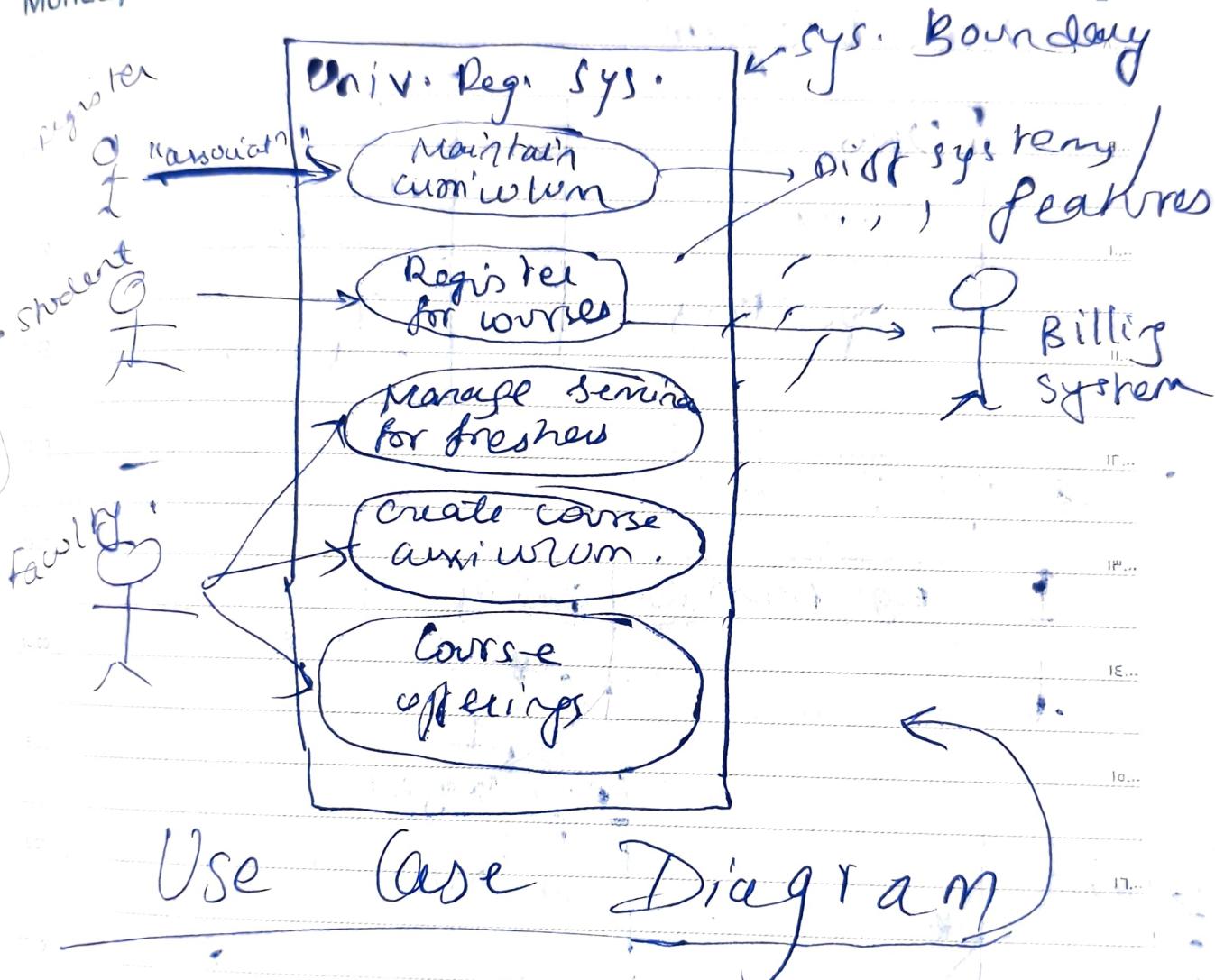
20

2025

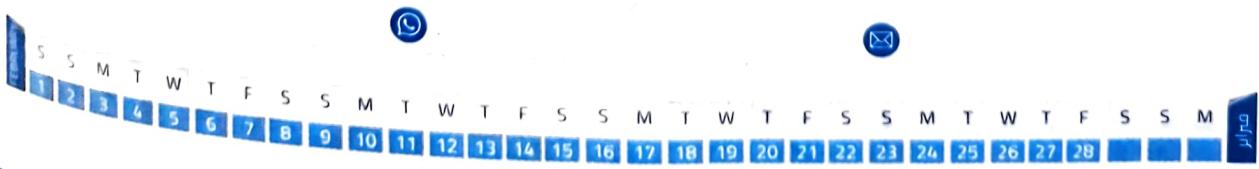
كانون الثاني/يناير

٢- حب- ١٤٤٦

## Monday



Week 4

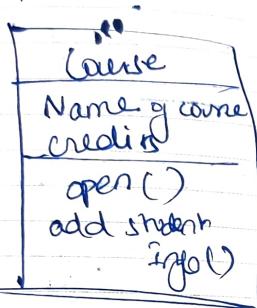
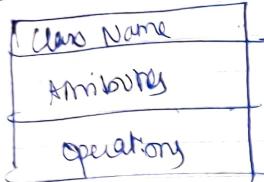


JANUARY  
21 Rajab 1446 H

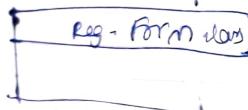
21

Tuesday

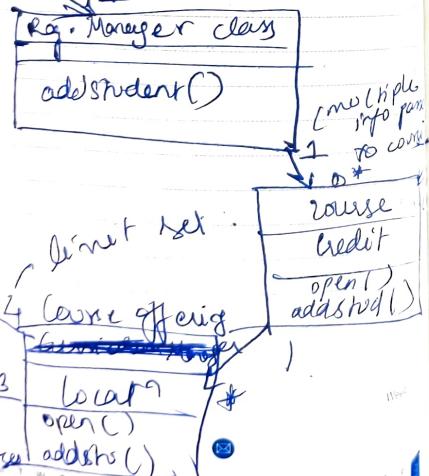
## Class Diagram



Class Dia:



(many registrars passed  
to one manager)



JANUARY

22 Rajab 1446 H

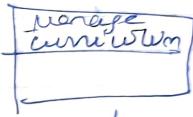
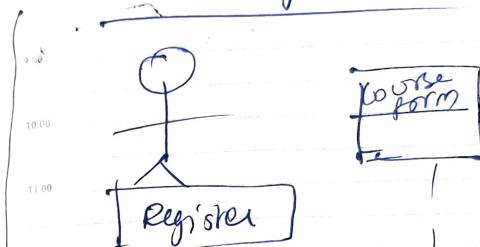
22

2025

كانون الثاني/يناير ٢٢-١٤٤٦ هـ

٢٢ - جب - ١٤٤٦ هـ

الأربعاء



- ```

graph LR
    A[1. Get course info] --> B[2. Start process]
    B --> C[3. Add course to curriculum]

```



Collaborate Dra draws after  
all diagrams



JANUARY

25 Rajab 1446 H

Saturday

25

2025

كانون الثاني / يناير  
١٤٤٦

$$\text{Testing} = V \neq V$$

Test Case → Describes i/p description & expected o/p description

and Test Strategy (TS) :

Major steps to follow for software testing

- TS
- ↓
- determine objective & scope
- ↓
- Type of test
- ↓
- Alt. strategy
- ↓
- Plan for environment where testing is done
- ↓
- Process

| W  | T  | F  | S  | S  | M  | T  | W  | T  | F  | S  | S  | M  | T  | W  | F  | S  | S  | M  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |    |    |    |    |    |    |    |
|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

JANUARY

26 Rajab 1446 H

Sunday

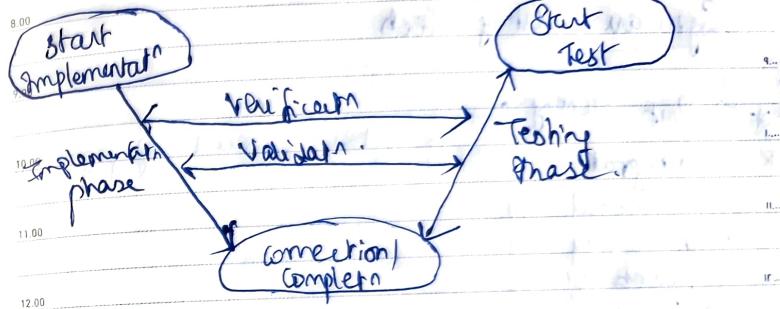
26

2025

كانون الثاني / يناير  
١٤٤٦

٢٧ ربـ ١٤٤٦

الأحد



### Testing Techniques

#### Structural Tech.

- Stress Testing
- Excuse
- Recovery
- Operatn
- Compliance
- security

#### Functional Tech

- Requirement Testing
- Regression
- Error handling
- Manual support
- Inter-system
- control
- Parallel

Week 4

JANUARY

27 Rajab 1446 H

Monday

27

2025

الموافق الثاني /يناير  
١٤٤٦ هـ

## Sofware Testing Tech

- ✓ → Unit testing (UT) gives execution of pgm
- \* → Integration " "
- \* → Functional " "
- \* → Usability " "
- \* → Reliability " "
- \* → Performance " "

### Unit Testing (UT)

Large pgm divided to small modules  
Each module tested separately.

Boundary conditions are tested in unit testing to see if modules operate properly at the boundaries.

Test cases checked in UT.

| WEEK | MON | TUE | WED | THU | FRI | SAT | SUN |
|------|-----|-----|-----|-----|-----|-----|-----|
| 1    | 1   | 2   | 3   | 4   | 5   | 6   | 7   |
| 2    | 8   | 9   | 10  | 11  | 12  | 13  | 14  |
| 3    | 15  | 16  | 17  | 18  | 19  | 20  | 21  |
| 4    | 22  | 23  | 24  | 25  | 26  | 27  | 28  |
| 5    | 29  | 30  | 31  | 1   | 2   | 3   | 4   |
| 6    | 5   | 6   | 7   | 8   | 9   | 10  | 11  |
| 7    | 12  | 13  | 14  | 15  | 16  | 17  | 18  |
| 8    | 19  | 20  | 21  | 22  | 23  | 24  | 25  |
| 9    | 26  | 27  | 28  | 29  | 30  | 31  | 1   |
| 10   | 3   | 4   | 5   | 6   | 7   | 8   | 9   |
| 11   | 10  | 11  | 12  | 13  | 14  | 15  | 16  |
| 12   | 17  | 18  | 19  | 20  | 21  | 22  | 23  |
| 13   | 24  | 25  | 26  | 27  | 28  | 29  | 30  |
| 14   | 31  | 1   | 2   | 3   | 4   | 5   | 6   |
| 15   | 8   | 9   | 10  | 11  | 12  | 13  | 14  |
| 16   | 15  | 16  | 17  | 18  | 19  | 20  | 21  |
| 17   | 22  | 23  | 24  | 25  | 26  | 27  | 28  |
| 18   | 29  | 30  | 31  | 1   | 2   | 3   | 4   |
| 19   | 6   | 7   | 8   | 9   | 10  | 11  | 12  |
| 20   | 13  | 14  | 15  | 16  | 17  | 18  | 19  |
| 21   | 20  | 21  | 22  | 23  | 24  | 25  | 26  |
| 22   | 27  | 28  | 29  | 30  | 31  | 1   | 2   |
| 23   | 4   | 5   | 6   | 7   | 8   | 9   | 10  |
| 24   | 11  | 12  | 13  | 14  | 15  | 16  | 17  |
| 25   | 18  | 19  | 20  | 21  | 22  | 23  | 24  |
| 26   | 25  | 26  | 27  | 28  | 29  | 30  | 1   |
| 27   | 1   | 2   | 3   | 4   | 5   | 6   | 7   |
| 28   | 8   | 9   | 10  | 11  | 12  | 13  | 14  |
| 29   | 15  | 16  | 17  | 18  | 19  | 20  | 21  |
| 30   | 22  | 23  | 24  | 25  | 26  | 27  | 28  |
| 31   | 29  | 30  | 31  | 1   | 2   | 3   | 4   |

JANUARY

28 Rajab 1446 H

كانون الثاني /يناير

٢٨ رجب - ١٤٤٦ هـ

الثلاثاء

28

2025

Tuesday

٩:٠٠

٣٧

Systematic technique for reshaping.  
Take UT components & build a proper pgm structure.

We take results of UT of all individual modules & combined them to ensure pgm tested altogether.

Run as a whole pgm → may cause an error.

Bugs difficult to solve here.

### Types of Integration

#### ① Top Down Integration (TDI)

Incremental approach. Starts from main control module. DFS method will be used for search.

TDI takes 3 approaches:-

- ① Close control correspondence over
- ② Causes overhead
- ③ Simplest method.

Week 5

JANUARY  
29 Rajab 1446 H  
Wednesday

29

2025

الثاني / يناير  
١٤٤٦ هـ

الجمعة

## ② Bottom up Integrations [BUI]

works in diff steps.

### \* System testing -

Not to find fault, but to see how they're working / if it's giving proper performance or not.

Software is incorporated

Software setup made to run on diff hardware so that also must be tested

Under operating conditions, how system performs.

If for eg: if 1000 students join AVMS, it shouldn't crash.

Compatibility should be there, Dependability

JANUARY  
30 Rajab 1446 H

30

2025

كانون الثاني / يناير  
١٤٤٦ هـ

الخميس

Thursday

8:00 Validation → after full pgm developed, working validation from customer POV.

9:00 Validate all functionalities of sys & must match with customer requirements.

10:00 for eg: if ip is for nos., but customer uses smg, then it is not favorable. So sometimes its customer fault. Improvement of Customer for validation is imp for delivery.

11:00 ∵ α, β, acceptance testing is done under validation testing -

### Validation Testing

16:00 → α & β for anonymous customers.

17:00 → for eg: Test done on product

18:00 → β e.g.: customer send invited ppl to sleep on bed for 8 hrs (very late)

19:00 → 11/18 company invite potential customers to test the software.

20:00 → acceptance

→ under supervision of developer

→ " real life " conditions - very common

