

Integration

- Testing individual modules
- Then integrate all modules and perform a system testing

Development - 100\$

New System — 40\$ ← cost effective

Existing System - 60\$
with updates

Q When an existing system is deprecated and a new system is developed? and why?

- This is migration

<from PDF>

Types of CASE Tools

Computer

Aided

Software

Engineering

C
A
S
E

Upper CASE → 1st 3 STEPS
of SDLC

Lower CASE → last 4 STEPS

Integrated CASE

নথে ভিত্তি এবং

SDLC → full software delivery

→ heavy documentation

→ budget fix, time: usually long

#The Agile Approach

Max industry level → Agile
Software Development

Team Members
5-8
avg. 7

* incremental delivery → called "sprint"

* collaboration

* continual learning

learning the

users' need as

we go

implement

new

technologies

on the fly

"Easy to
Understand"

HARD to
Master"

Based on:

↳ Values (beliefs)

↳ Principles (how they act and think)

↳ Core practices (what they do daily)

flexible application

fast delivery

proper transparent
communication

not unnecessary

documents

users are

directly involved

(user-centric feedback
in every step/phase
of development)

short time/
flexible timing

always
delivers the
best

quality

software

802

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1020

1021

1022

1023

1024

1025

1026

1027

1028

1029

1030

1031

1032

1033

1034

1035

1036

1037

1038

1039

1040

1041

1042

1043

1044

1045

1046

1047

1048

1049

1050

1051

105

Values of Agile

- ① Communication
- ② Simplicity
- ③ Riskier features are implemented later (Courage)

④ Feedback

Max sprint duration

↳ 1/2 weeks (max 4 weeks)

Principles of Agile

- ① Deliver software frequently
- ② Embrace changes
- ③ Work with customers regularly
- ④ Keep a sustainable pace
- ⑤ Improve continuously

Core Practices

- ① work in short cycles
- ② Daily meetings
- ③ Test frequently
- ④ collaborate as a team

Agile Team

"The show must go on"

Every developer can do

Team Transparency

meets

collaboration

SCRUM

Adjustable Resources

- ① Time
- ② Cost
- ③ Quality (Ensuring best quality)
- ④ Scope

5 Stages of Agile Development

- ① Exploration
- ② Planning
- ③ Iterations of first release
- ④ Productionizing
- ⑤ Maintenance

"SCRUM"

has official guide
easy to understand

will be elaborated in next class

When to use

SDLC, Agile, UML?

from PDF

90% Agile Development follows SCRUM

"agile development demands skillful developers"

"agile development is user-centric"

Where to use agile?

- ↳ user-centric development (E-commerce)
- ↳ time and budget is flexible
- ↳ fast software delivery.

Planning
of SDLC

Agile Project Development Process

① Exploration

② Planning

③

Iterations to
the first
release

④ Productionizing

⑤

Maintenance

Phases elaboration

↳ going to
from PDF

Planning

↳ Customer decides what
the development should
tackle first.

↳ riskier customer
suggestions may be
considered

↳ team members
may be rotated on
or off the team

Iterations

↳ continuous
feedback from the customer
and daily meetings

SCRUM

- ① Product Backlog (all requirements gathered)
- ② Sprint planning
(Sprint backlog)
- ③ SCRUM Team

{ new requirements can be added or removed

(cross functional developers)

Po → Product Owner (leading daily SCRUM)

SM →

hosts stand-up meeting
max 15 minutes

- ④ Increment
- ⑤ Sprint review
- ⑥ Sprint retrospective

Team Leader
of projects

Management
Communication
Daily Meeting

reviewing
the previous
sprint