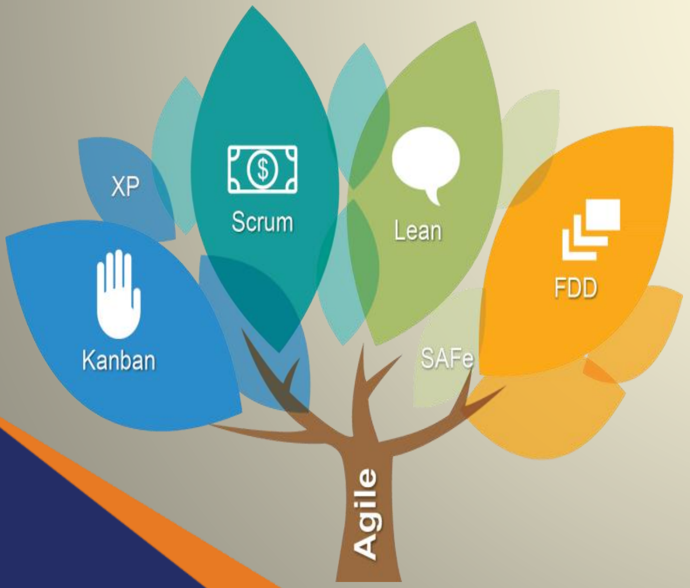
Software Process Model 2020

Introduction



Session outcomes

- Introduction to Module
- SPM- Introduction

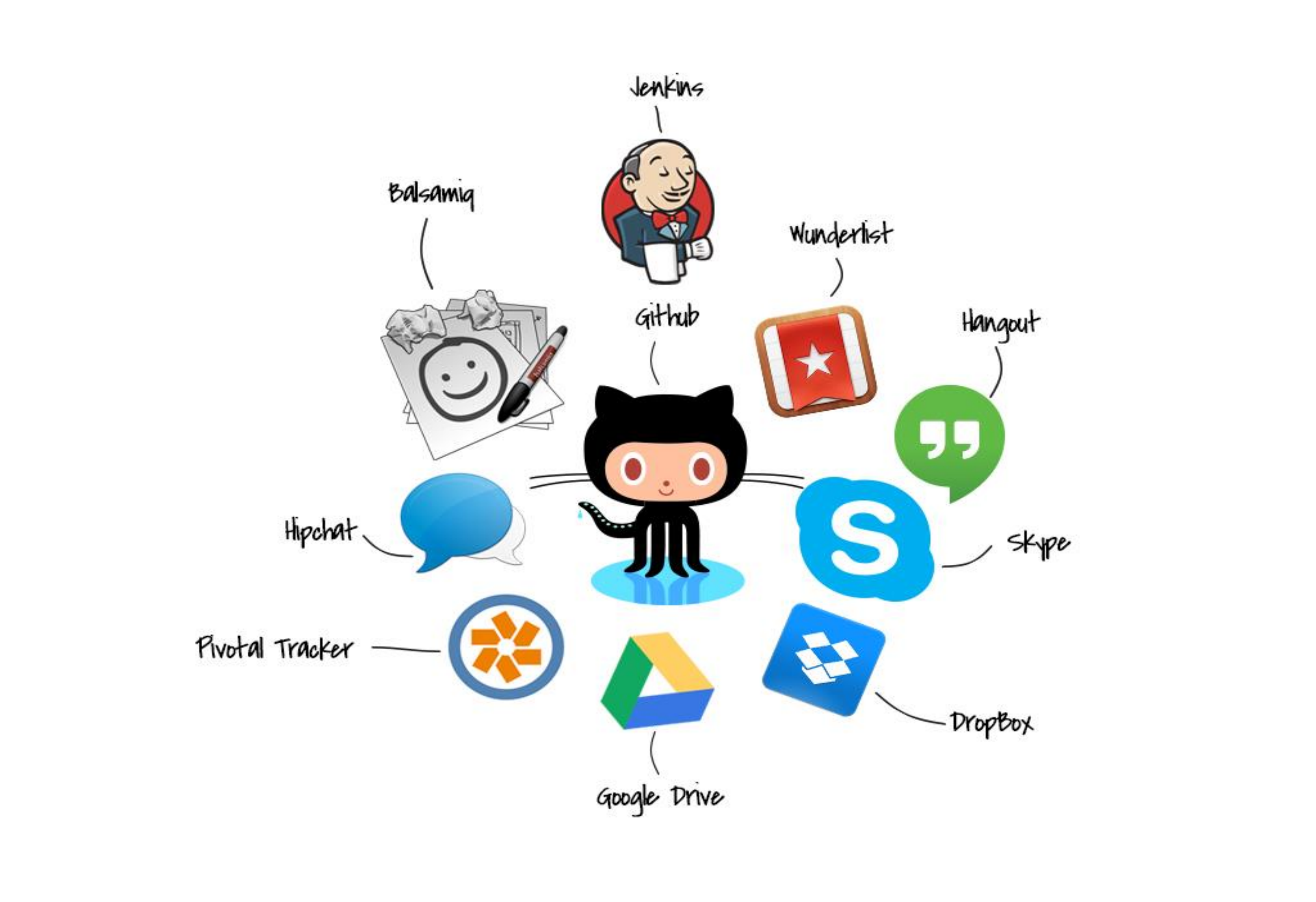
Academic Integrity Policy

- Are you aware that following are not accepted in SLIIT???
  - Plagiarism - using work and ideas of other individuals intentionally or unintentionally
  - Collusion - preparing individual assignments together and submitting similar work for assessment.
  - Cheating - obtaining or giving assistance during the course of an examination or assessment without approval
  - Falsification – providing fabricated information or making use of such materials
- Committing above offenses come with serious consequences !
- See General support section of Courseweb for full information.

Progression Criteria

- In order to progress from one academic year to the next, you must maintain the following minimum academic standard.
  - From year 1 to year 2 – No more than **5 failed or incomplete** modules
  - From year 2 to year 3 – No more than **3 failed or incomplete** modules
  - From year 3 to year 4 – No more than **2 failed or incomplete** modules
- If you do not meet the above criteria, you will not be able to progress to the next year.

MODULE INTRODUCTION



Module contents

- Course web - SPM-IT1060
  - IT1060
- General
  - Module outline
  - Notices
  - Marks
- Weekly updates
  - Lecture
  - Lab
  - Tutorial
  - Additional Reading

Learning outcomes

Differentiate the characteristics and effects of different types of software engineering processes.

Describe the requirement engineering process and components of a formal requirements document for a software project.

Apply the knowledge of UML to model and represent system requirements.

Describe software design strategies and the importance of design models.

Apply the knowledge of software implementation and testing to write test cases.

Apply Agile development methodology.

Assessment Criteria

Mid Term Examination	30%	LO1-LO4
Assignment I	10%	LO3-LO5
Assignment II	10%	LO4-LO5
Final Examination	50%	LO1-LO9

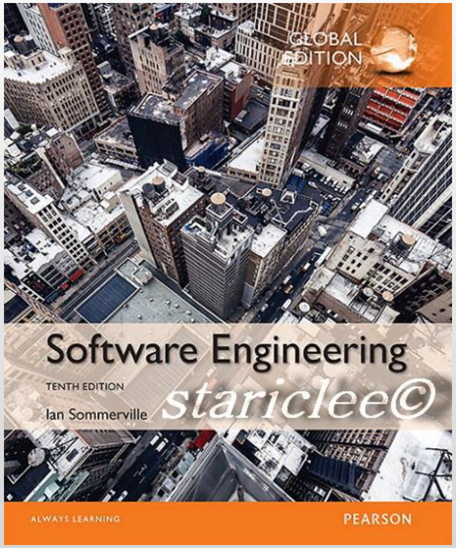
To pass this module, students need to obtain a pass mark in both “Continuous Assessments” and “End of the Semester Examination” components which would result in an overall mark that would qualify for a “C” grade or above.

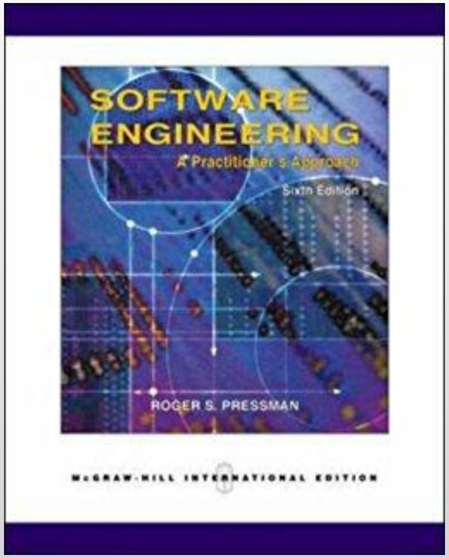
Assignments

- **Five members** in one group
- Randomly chosen case study
- Two submissions
  - Before Mid term – Week 7
    - Based on requirements engineering and use case diagrams
  - After Mid term – Week 12
    - Based on Activity Diagram

Recommend Texts

- ***Ian Sommerville, "Software Engineering", Pearson Education Limited, 10th edition, 2016***
- ***R. Pressman, "Software Engineering: a practitioner's approach", McGraw-Hill Education; 8th edition, 2014***
- ***K.S. Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison-Wesley, 2012***
- *SWEBOK, Guide to the Software Engineering Body of Knowledge, 2014*

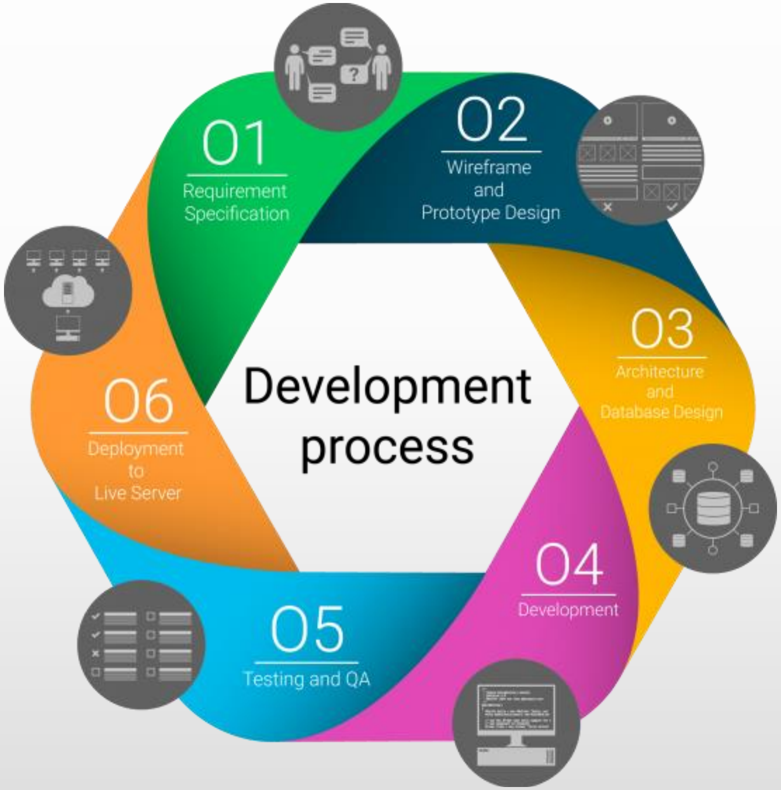




Lab Schedule

Time	Lessons	Labs/Tutorial
Week 1	Introduction to Software Process Modeling	
Week 2	SDLC Models	Lab – Introduction to modeling Tools
Week 3	SDLC Models	
Week 4	Requirements Engineering	Tutorial - SDLC
Week 5	Use Case Diagram	Tutorial – Requirements Eng.
Week 6	Use Case Diagram	Lab - Use Case Diagram
Week 7	Activity Diagram	Lab - Use Case Diagram
Week 8	**Midterm Examination**	
Week 9	Software Design	Lab - Activity Diagram
Week10	Implementation and Testing	Lab - Activity Diagram
Week 11	Implementation and Testing	Tutorial – Software Design, Implementation, testing
Week 12	Modern Software Development Methodologies	Lab - Agile
Week 13	Modern Software Development Methodologies	Lab - Agile
Week14	Revision	

SPM- INTRODUCTION


Development process
01
Requirement Specification
02
Wireframe and Prototype Design
03
Architecture and Database Design
04
Development
05
Testing and QA
06
Deployment to Live Server

1^{st} Year 1^{st} Semester IP

```
/* adding two numbers*/
#include <stdio.h>

int main(void)
{
        int no1, no2;
        int sum;

        no1 = 25;  //  assign value to no1 variable
        no2 = 12;  //  assign value to no2 variable

        sum = no1 + no2;  // add numbers

        printf( " Sum is %d\n",  sum);    // print sun
        return 0;
 } // end of main function
```

```
/* adding two numbers*/
#include <stdio.h>

int main(void)
{
        int no1, no2;
        int sum;

        printf("Enter first number:  ");   /* prompt */
        scanf("%d", &no1);  /* read the value */

        printf("Enter second number:  ");    /* prompt*/
        scanf("%d", &no2);  /* read the value */

        sum =  no1  + no2;    /* assign total to sum */

        printf( " Sum is %d\n",  sum);    /* print sum */

        return 0;
} // end of main function
```

- Are these Software ?
- What are things that you need to do to develop Software?

Session Outcomes

1. What is a Software
2. What is Software Engineering
3. Software Process
4. Software Process Activities
5. Software process model
6. Software Development Life Cycle
7. Software Engineering Ethics

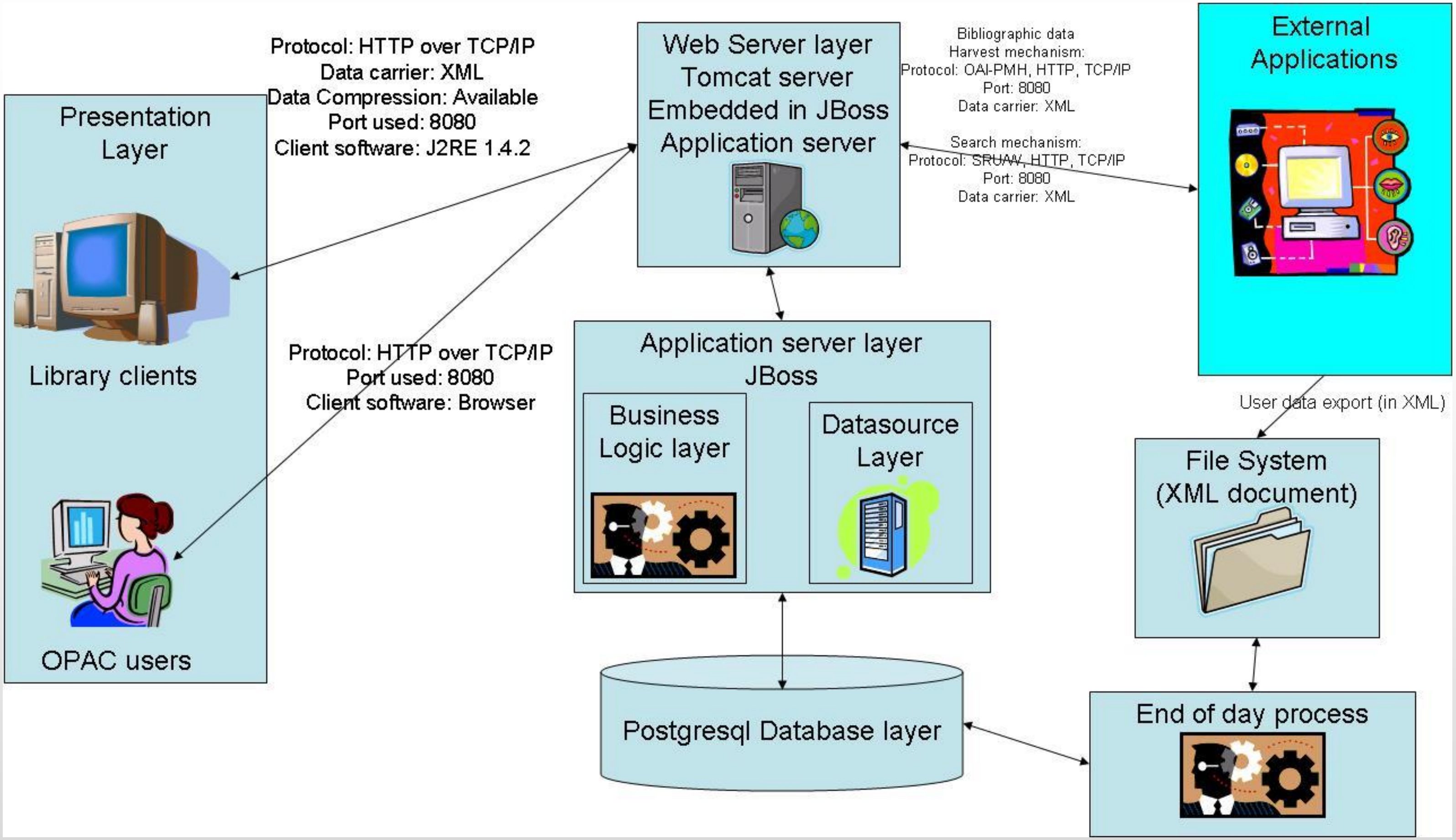
What is Software?

Software is **not only** the computer programs, but also associated documentation and configuration files, needed to make the programs operate correctly.

Popular Software

W
WinRAR®
Linux™
Ps
ANDROID
TM
Microsoft®
.NET
MySQL®
nero
Burning ROM

Library Software



http://www.verussolutions.biz/technology.php

Programs Vs. Software Products

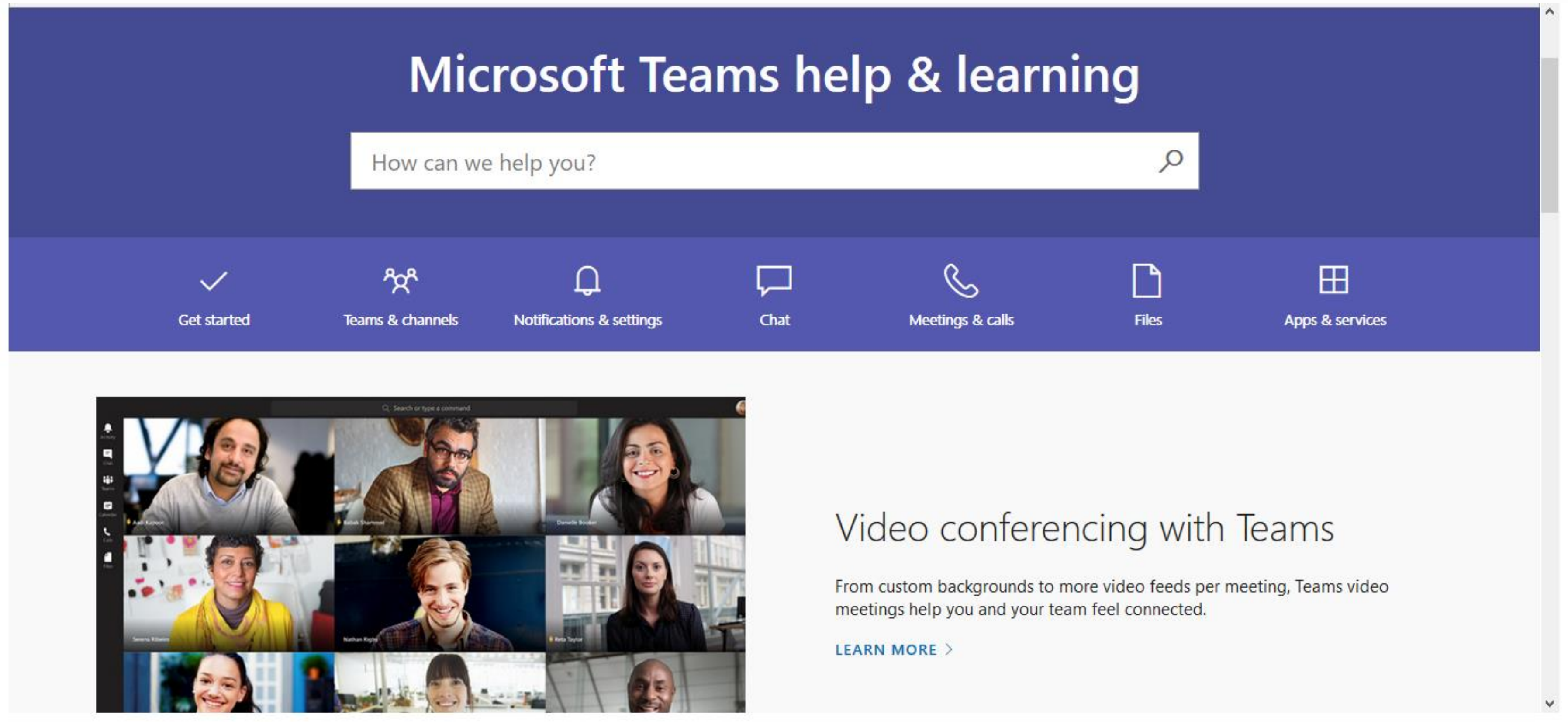
Program

- Small
- Single developer
- Small in size
- Limited Functionality
- Single user (author)
- Simple user interface
- Sparse documentation
- No user manual
- Ad hoc development

Software Product

- Large
- Team of developers
- Multiple users (customer)
- Complex user interfaces
- Detailed documentation
- User manual
- Systematic development

MS Teams



Microsoft Teams help & learning

How can we help you?

Get started

Teams & channels

Notifications & settings

Chat

Meetings & calls

Files

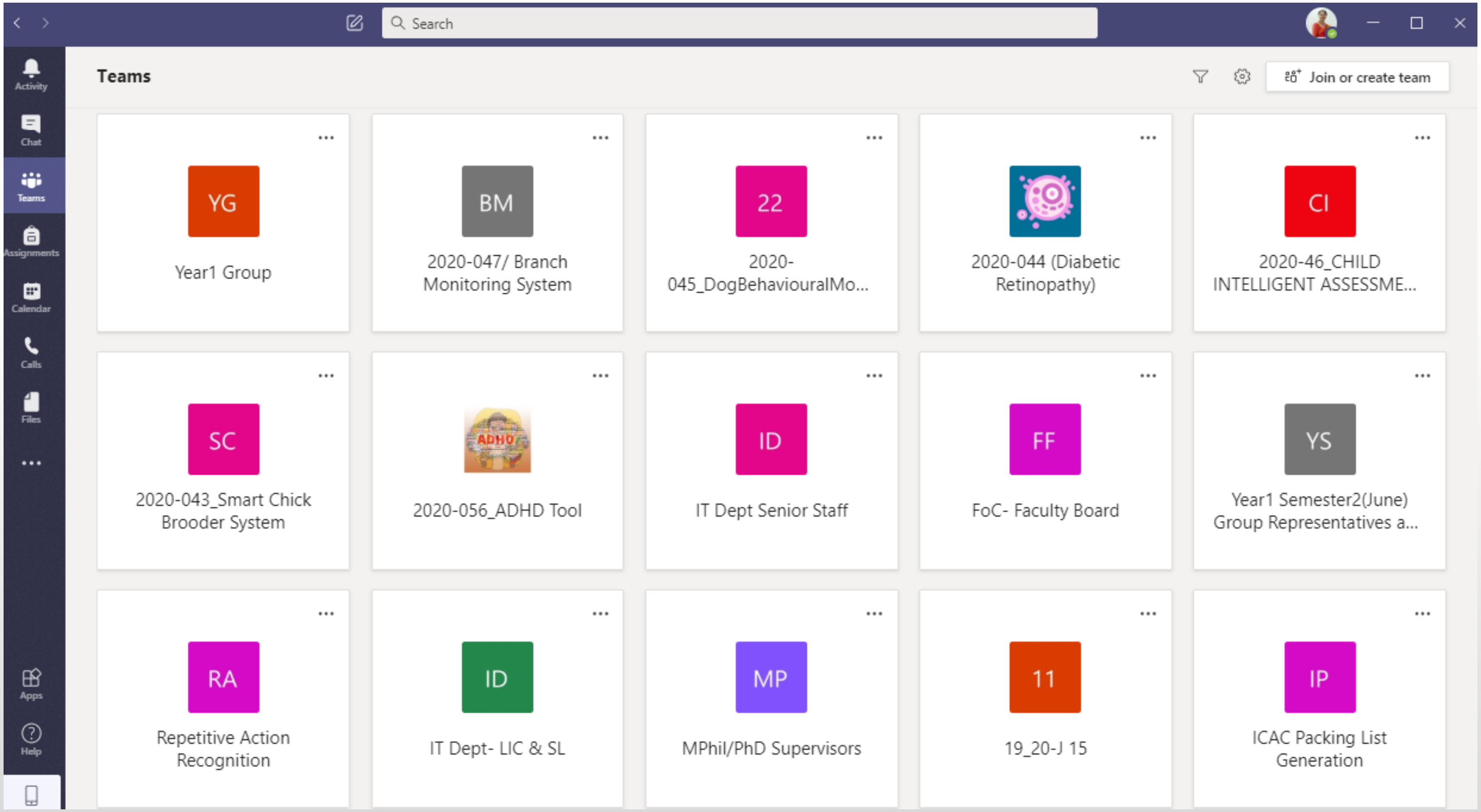
Apps & services

Video conferencing with Teams

From custom backgrounds to more video feeds per meeting, Teams video meetings help you and your team feel connected.

LEARN MORE >

MS Teams



Teams

YG Year1 Group

BM 2020-047/ Branch Monitoring System

22 2020-045_DogBehaviouralMo...

2020-044 (Diabetic Retinopathy)

CI 2020-46_CHILD INTELLIGENT ASSESSME...

SC 2020-043_Smart Chick Brooder System

2020-056_ADHD Tool

ID IT Dept Senior Staff

FF FoC- Faculty Board

YS Year1 Semester2(June) Group Representatives a...

RA Repetitive Action Recognition

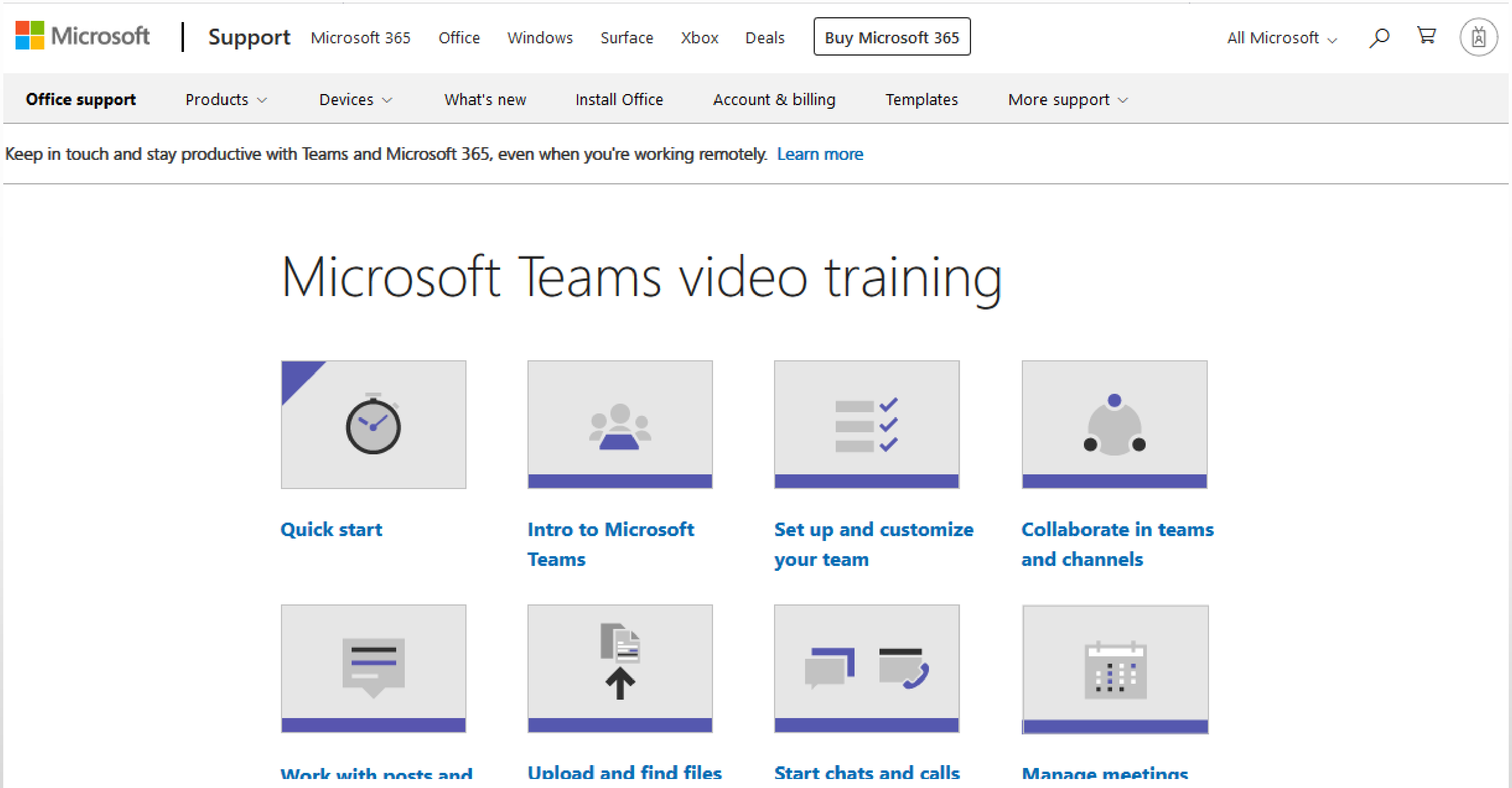
ID IT Dept- LIC & SL

MP MPhil/PhD Supervisors

11 19_20-J 15

IP ICAC Packing List Generation

Teams Documentation



Keep in touch and stay productive with Teams and Microsoft 365, even when you're working remotely. Learn more

Microsoft Teams video training

- Quick start
- Intro to Microsoft Teams
- Set up and customize your team
- Collaborate in teams and channels
- Work with posts and
- Upload and find files
- Start chats and calls
- Manage meetings

Software products can be

- Generic
  - These are stand alone systems that are produced by a development organization and sold on the open market to any customer who is able to buy them.
- Customized
  - These are systems that are developed for a particular customer requirements

How do we develop a real software?

- There will be a real user (Customer) who would need to use the software.
  1. Feasibly study (whether it is technical feasible and financially worthwhile)
  2. You have to find out what the customer wants (Requirements Gathering)
  3. Analyze the problem
  4. Develop a solution (Design)
  5. Code the solution
  6. Test and Debug
  7. Maintenance

Suggest Something Innovative?

- Suggest your dream software
  - Do not think about technical barriers
  - You can think beyond of the reality

“New Ideas will lead you to highest point of the Software Engineering”

Software Engineering

- IEEE Definition of Software Engineering:

The application of a *systematic*, *disciplined*, *quantifiable* approach to the development, operation, and maintenance of software;

that is, the application of engineering to software.

IEEE Standard 610.12-1990, 1993.

Software Engineering Cont.

- Engineering discipline

  make things work by applying theories, methods and tools where these are appropriate and also try to discover solutions to problems even when there’s no proper theories/methods.

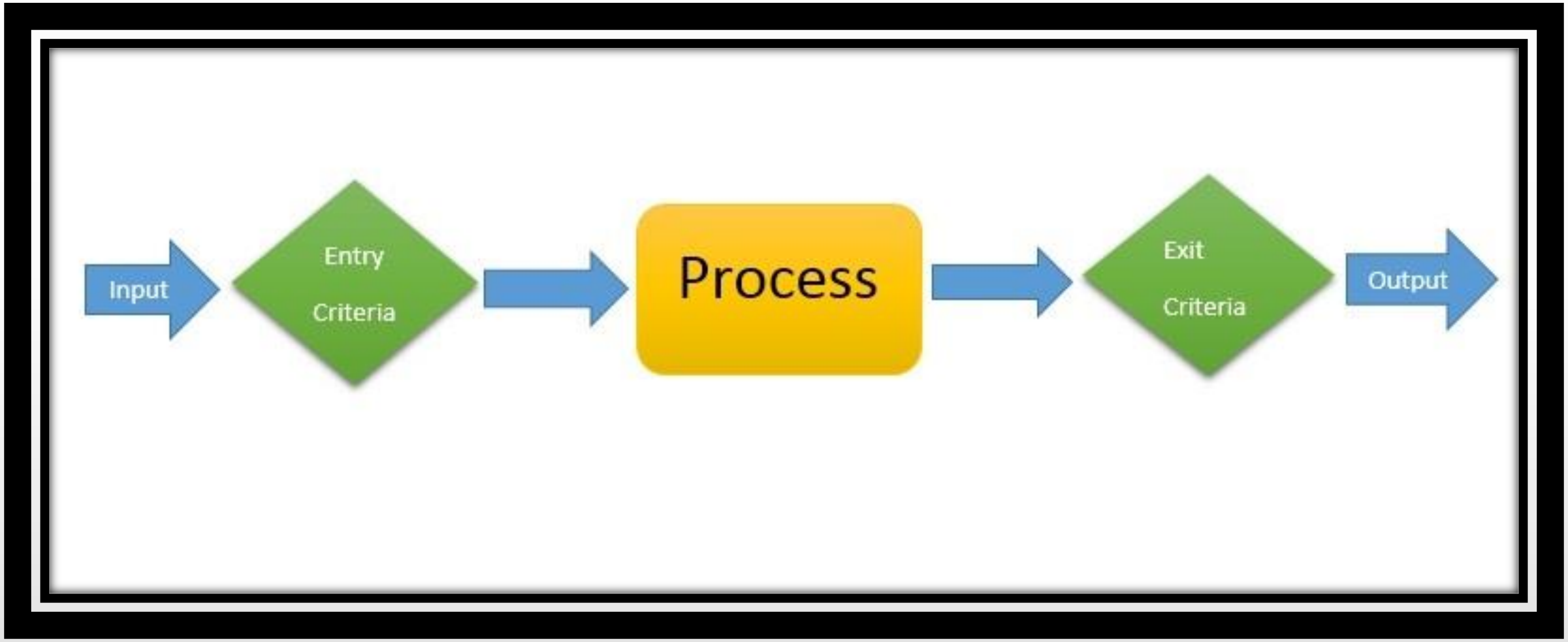
- All aspects of software production

  Not only technical processes of software development, but also project management and development of tools, methods and theories to support S/W production.

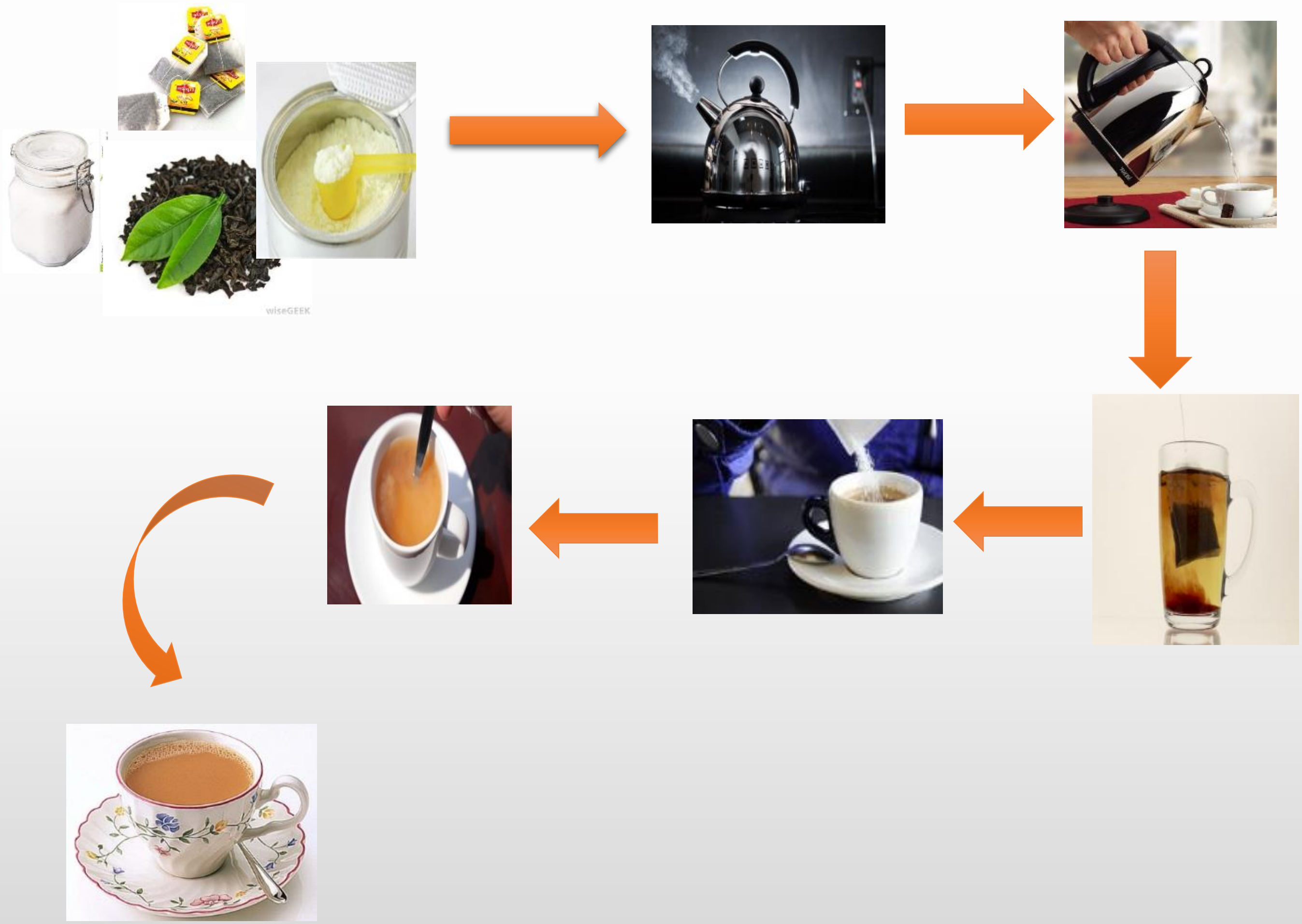
Key Challenges

- The product is intangible
- Deliver Quality software to the customer at the agreed time
- Keep overall costs within budget
- Software processes are available and organization/product specific

Process



Making A Cup of Tea



Making A Cup of Tea

- Ingredients : Tea Leaves, Sugar, Milk Powder, Boiled Water
- Process

  Boil the water

  Pour boiled water into cup

  Put a tea bag inside a cup

  Leave it few minutes

  Put Sugar and Milk (if necessary)

  Stir few seconds

  Arrange it nicely

- Output: Tea

Software Process

- A software process is a set of interrelated activities and tasks that transform input work products into output work products. (SWEBOK V3 – Chapter 8)

Software Process Activities

- Software Specification
- Software Development
- Software Validation
- Software Evolution

Software Process Activities

- Software Specification
  - The functionality of the software and constraints
- Software Development
  - The software is designed and programmed.
- Software Validation
  - The software must be validated
- Software Evolution
  - The software must evolve

Software Processes

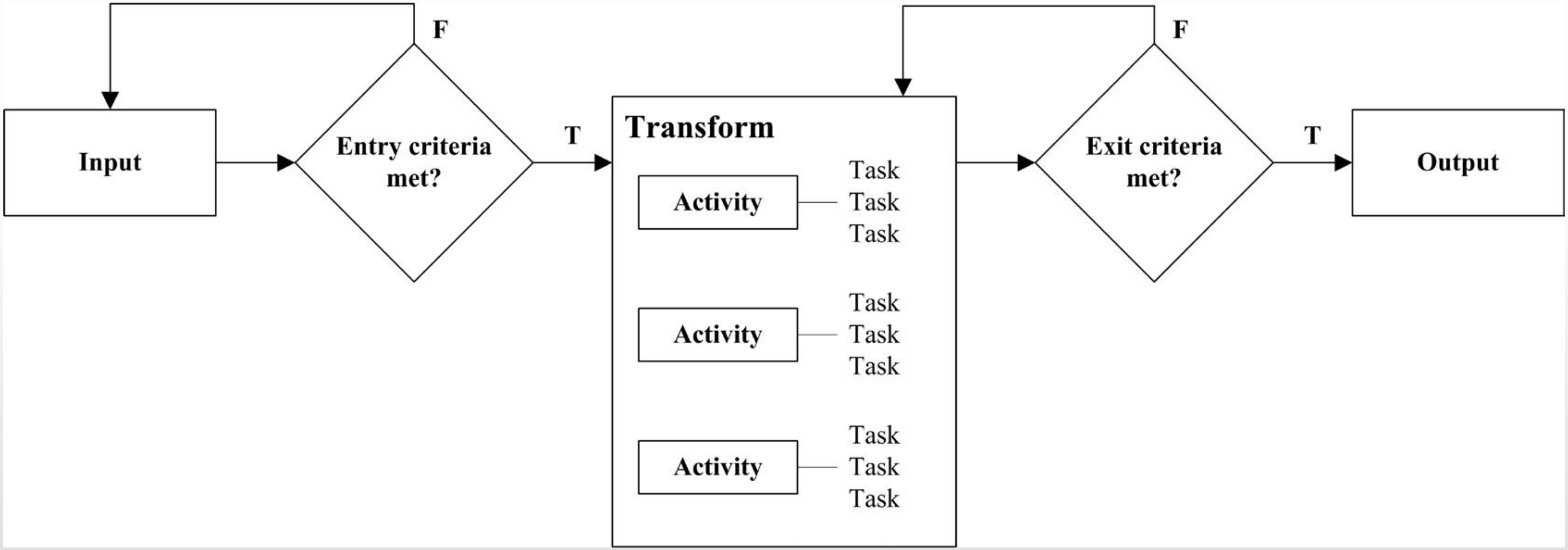
“There is no universal process that is right for all kinds of software”

Ex:

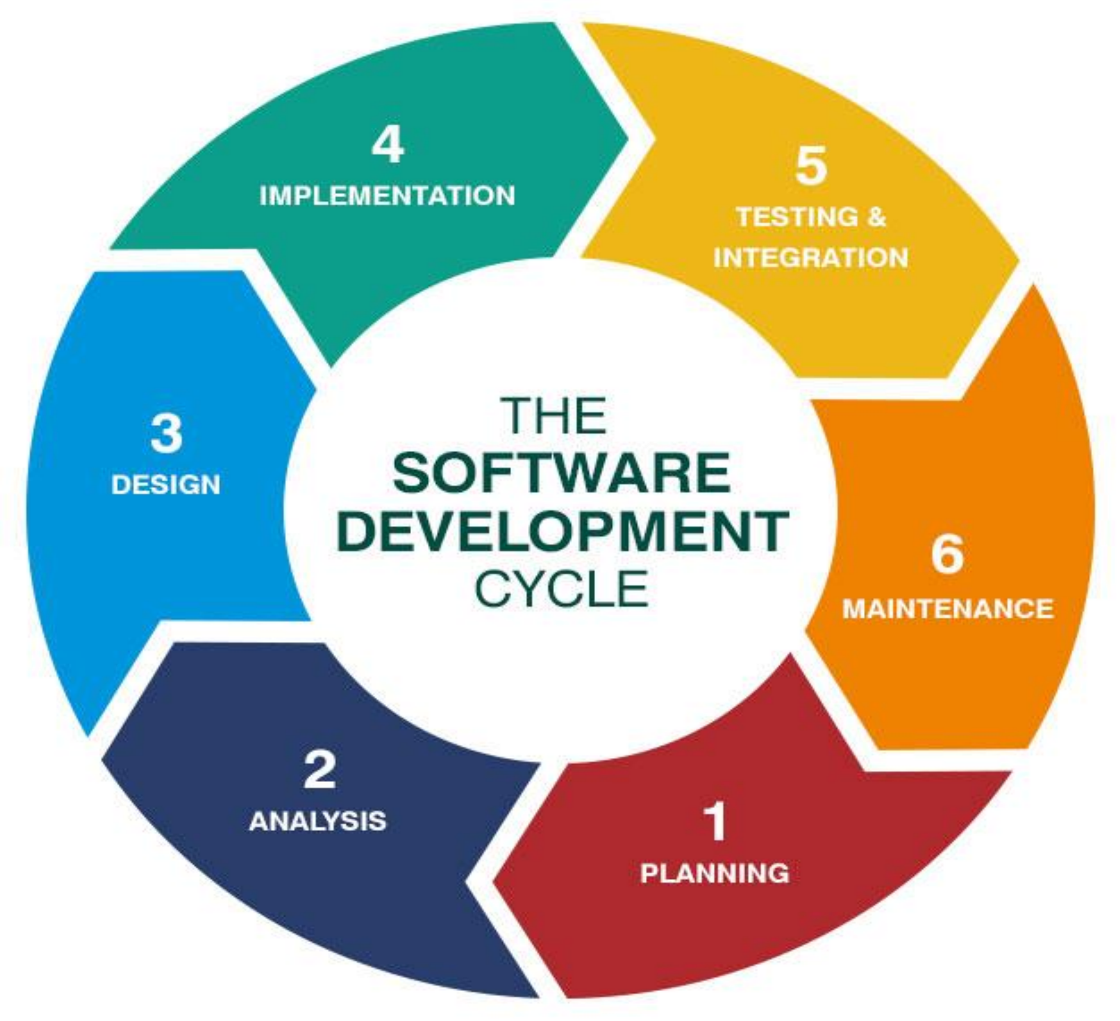
- For safety-critical systems, a very structured development process is required where detailed records are maintained.
- For business systems, with rapidly changing requirements, amore flexible ,agile process is likely to be better

Software process model

- It is a simplified representation of software process.



Software Development Life Cycle



Software Development Life Cycle Models

- A Software Development Life Cycle Model
    - has a series of stages that a software product undergoes during its life time.
    - is a descriptive and diagrammatic representation of the software life cycle.
    - is often referred as software process model.
    - maps the basic development activities to phases in different ways

General Software Process Models

- Waterfall Model
  - Classic
  - Iterative
- Prototyping
- Evolutionary Model
  - Incremental
  - Spiral
- Rapid application development.
- Agile development.

Software Engineering Ethics

As a Professional Software Engineer,

- You should accept that your work involves wider responsibility than simply application of technical skills
- You should behave in an ethical way and morally responsible way
- You should not use your skills and abilities to behave in a dishonest way that will bring disrepute to the software engineering profession

Software Engineering Ethics Con.

Standards

- Confidentiality
- Competence
- Intellectual Property rights
- Computer misuse

Case Studies

- Library Management System

Library Management System

- Sri Lanka Institute of Information Technology (SLIIT) is the largest degree awarding institute in Sri Lanka with degree programs diversified to computing, business and engineering. In order to cater to its growing need of knowledge the institute maintains a Library Information System connecting Malabe, Metropolitan and Matara campuses. Each holds a latest collection of books and periodicals, particularly in the field of Information Technology, business management, engineering, general English, architecture and quantity surveying. The library of the Malabe Campus acts as the main resource center through which all library development activities are coordinated. SLIIT libraries are open to SLIIT students daily including weekends from 7.30 AM to 7.00 PM.

Tasks carried out at the library

- Add library materials
- Manage Library membership
- borrow books
- return books
- Pay fine on overdue materials
- Refund library deposit
- Replace lost library material
- Search library materials
- Generate reports

Next Lecture

Software Development Life Cycle Models