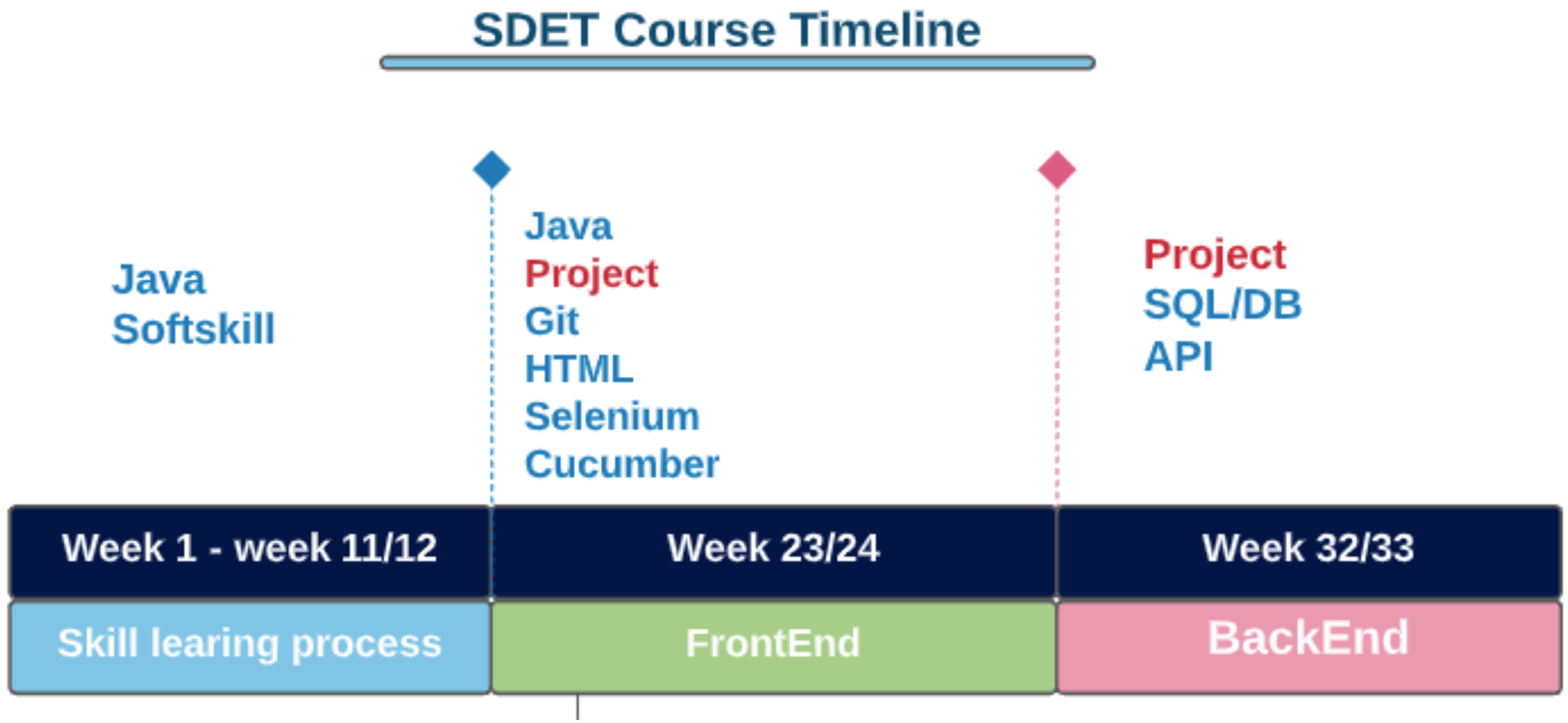
Softskill Introduction



Softskill

Personal skill, personality trait.

Common or core skills which are desirable in IT professions.

Your softskill determinds **how you work** and how you **interact with others**.

- Communication
- Team work
- Leadership
- Time management
- Taking responsibility
- Positive attitude

VS.

Hard/Tech skill

Job-specific skills learned through education. Technical skill is about **how you perform tasks** and how you work **productively** and **efficiently**.

- Coding skill
- Typing skill
- Microsoft office skills
- Management skill
- Operating System
- Technical tools

Softskill class Topics

1. Software Development Life Cycle (SDLC)

Standard steps to build softwares

2. Intro to Software Testing Types

2.1 **Static** testing vs **Dynamic**testing
2.2 **Testing levels** in Dynamic testings
- Level1: **Unit** testing
- Level2: **Integration** testing
- Level3: **System** testing
- Level4: **User Acceptance** Testing (UAT)

2.3 **Functional** testing vs **Non-Functional** testing
2.4 **Smoke** testing & **Regression** testing

3. Software Methodology

3.0 Waterfall **vs Agile**
3.1 Agile - **Scrum**
3.2 Scrum **Role** - PO, SM, the Development team
3.3 Scrum **Artifacts** - Product Backlog, Sprint Backlog, Product I ncrement, Burndown chart
3.4 Scrum **meetings** - Grooming, Sprint Planning, Daily stand up, Demo, Retro

4. Software Testing Life Cycle(STLC)

4.1 Test Plan Creation
4.2 Test Case Creation
4.3 Bug Life Cycle

5. Jira - Project management tool

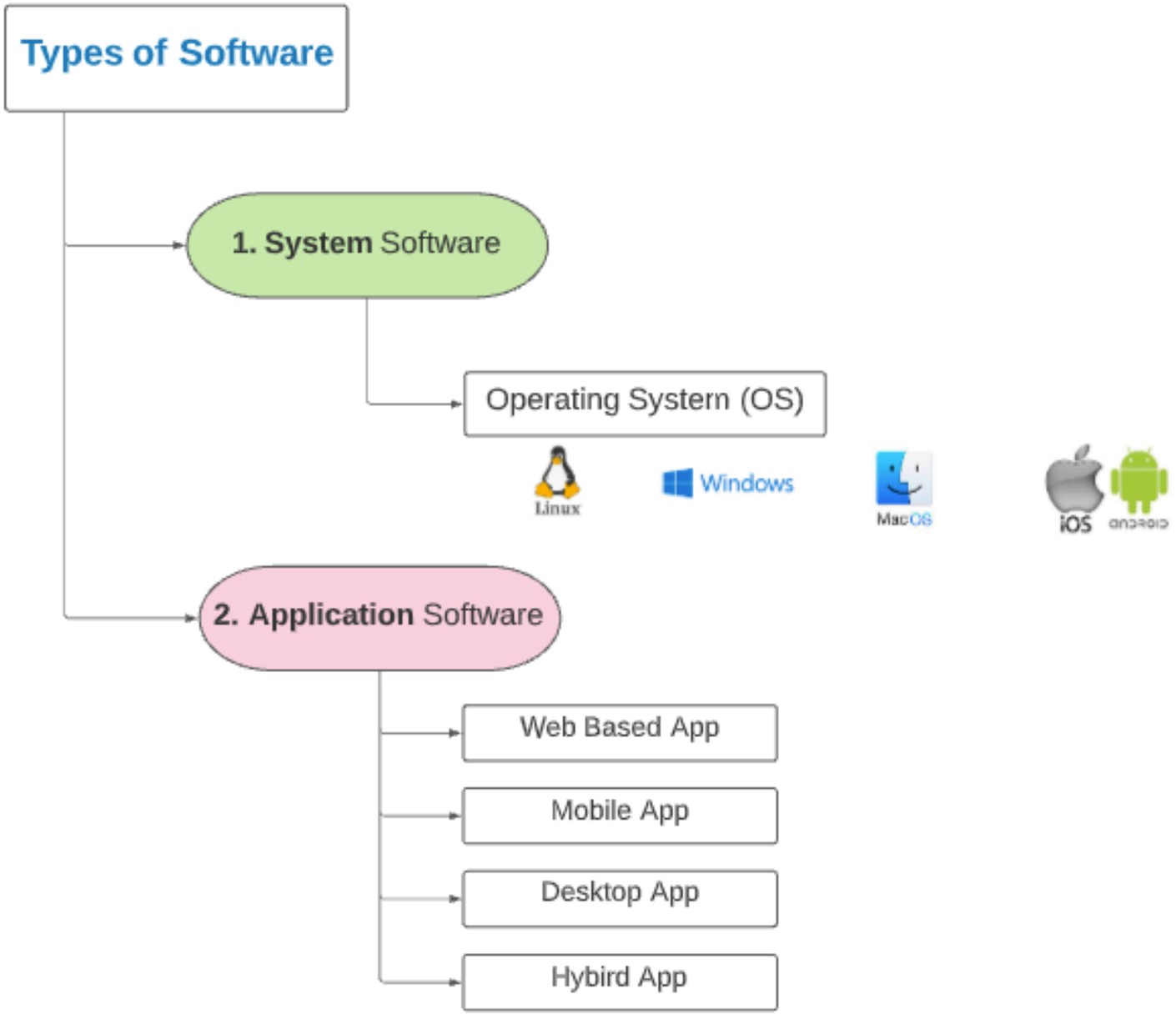
Softskill Keywords

Questions & Answers from this topic:

- What are the application **software types** ?
- What are the **three layers** of software?
- Who works in the **business department**?
- Can you give several examples of the **functionalities** in any software?
- Can you give several examples of the **non-functionalities** in any software?
- What type of tester will you be after the Cydeo-SDET course?

What is Software?

Definition: Set of instructions that perform specific task for users. Softwares are designed to help people to perform activities.


Types of Software
1. System Software
Operating System (OS)
Linux
Windows
MacOS
iOS
android
2. Application Software
Web Based App
Mobile App
Desktop App
Hybird App

1. **WebBased Application:** Web apps are **accessed over network connections** using HTTP/HTTPS with any browser (Safari, Chrome, IE, Firefox, etc.)

2. Mobile Application: Mobile apps are designed to run on **mobile** devices such as phones, tablets, or watches.

3. Desktop Application: Desktop apps that you download and install on your device / on any Operating system (Mac OS, Windows, Linux)

4. Cross-Platform/Hybird Application: Cross-platform software works on multiple Operating Systems(OS) or devices.

Software Layers (3)

FrontEnd, DataBase, API

1.Front-End / User Interface (UI)

Back-End

Functional: Functionality of a software describes **what a software should do,** or should not do.

Example: Users login. logout, uplaod files, edit username, click confirmation button, zoom in/out, etc

Non-Functional: Non-functional specify **how the software should do behave.**

Example: **Respons time**: user should download a file min-5seconds - max 35 secodns.

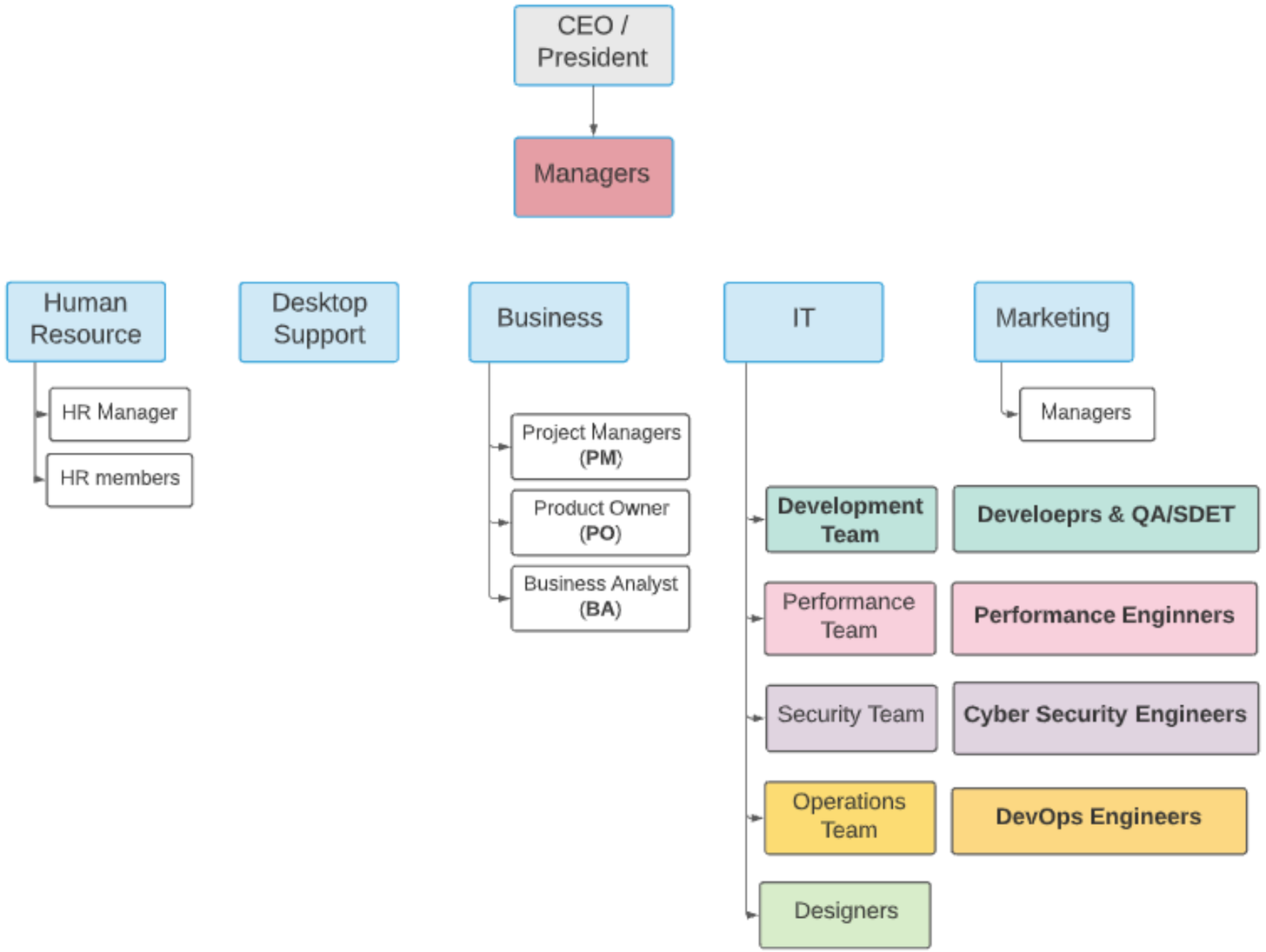
Accessiblity : Web, mobile, ipad, etc

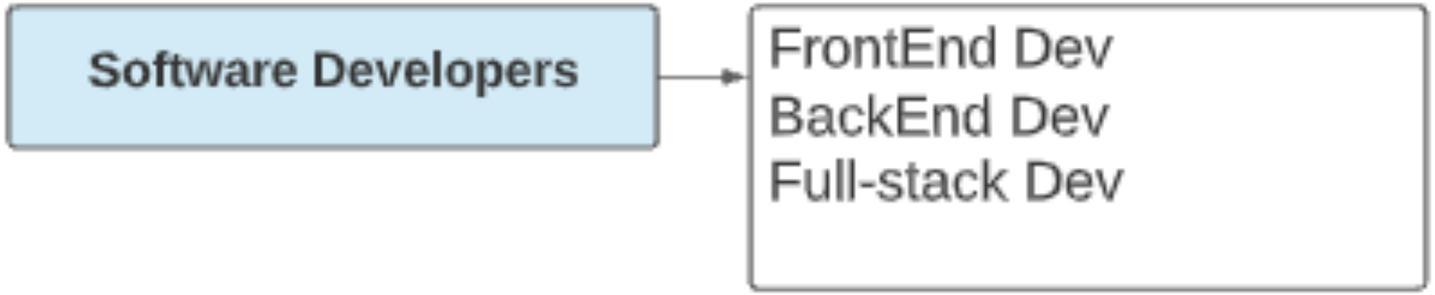
Installation: How to install? avaliable browsers?

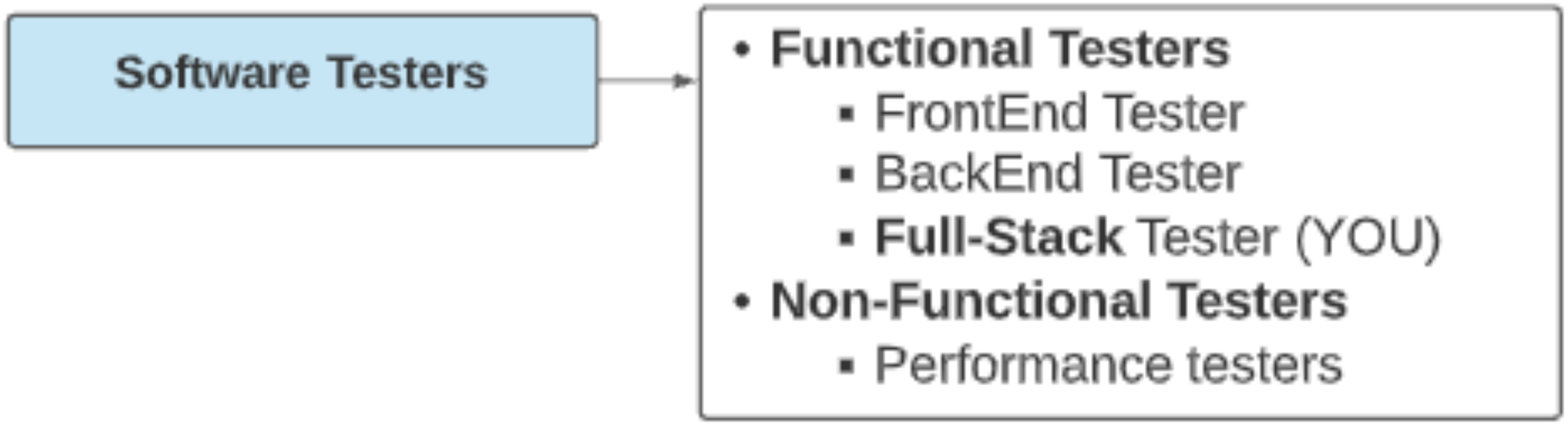
Performance: How fast does it need to operate?

Security: What are the security requirements? a cyber perspective

Departments in IT industry



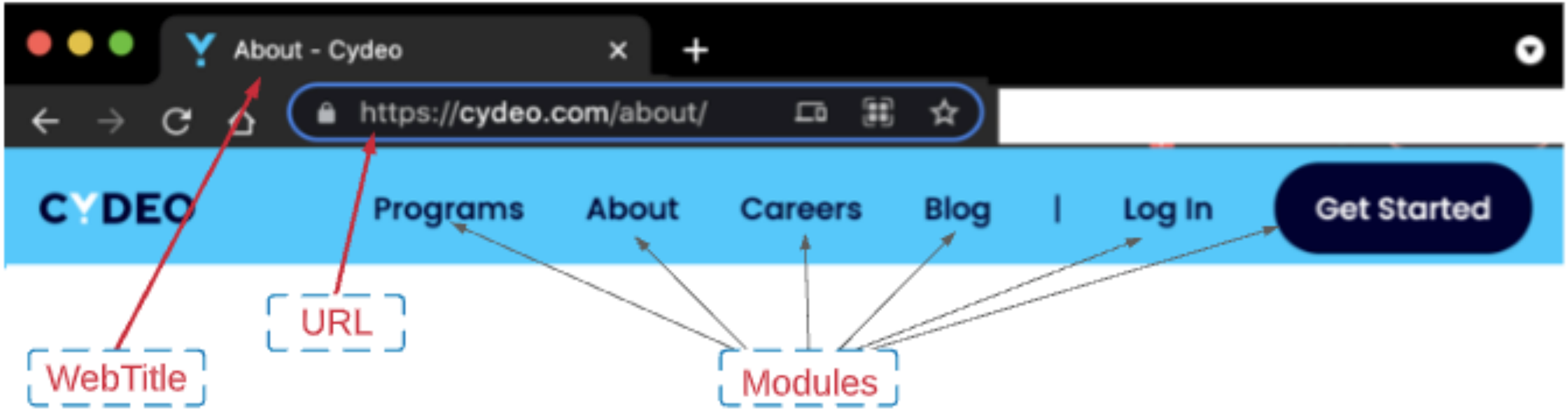

Software Developers
FrontEnd Dev
BackEnd Dev
Full-stack Dev


Software Testers
• Functional Testers
▪ FrontEnd Tester
▪ BackEnd Tester
▪ Full-Stack Tester (YOU)
• Non-Functional Testers
▪ Performance testers

As a Tester, your title at work could be:

1. **SDET** (Software Development Engineer in Test)
2. **QA** (Quality Assurance)
3. Software Tester
4. Automation Tester
5. QA Tester
6. Software Test Engineer
7. QA Automation Engineer
8. QA Engineer
9. Automation Test Engineer
10. Test Automation Engineer
11. Software Engineer in Test

Module, Function, Feature



URL: **URL** stands for Uniform Resource Locator. URL is the address of a given unique resource on the Web.

Web Title: A website title identifies what the web page is about for both web users and search engines.

Module: A unit or component of a software, each module iinclude one or more features of the app.

There are 6 modules on the Cydeo website

Feature: Behavior of a software, what users can do with the app?

Function:how to achieve a feature, what actions users can perform on the app? (Click, zoom in & out, select, scroll page, enter/type a value, upload/download a file, video & voice call, etc)

Example 1 :

Feature: Customers **search** for the different **programs** from the website.

Functions : A user click "Programs" module

User scroll down untill see "our programs"

User view detail info of different programs.

Example 2:

Feature: Customers **apply** to a program/course

Functions : user click "Get Started" module

user fill in the application information form

user submit the form

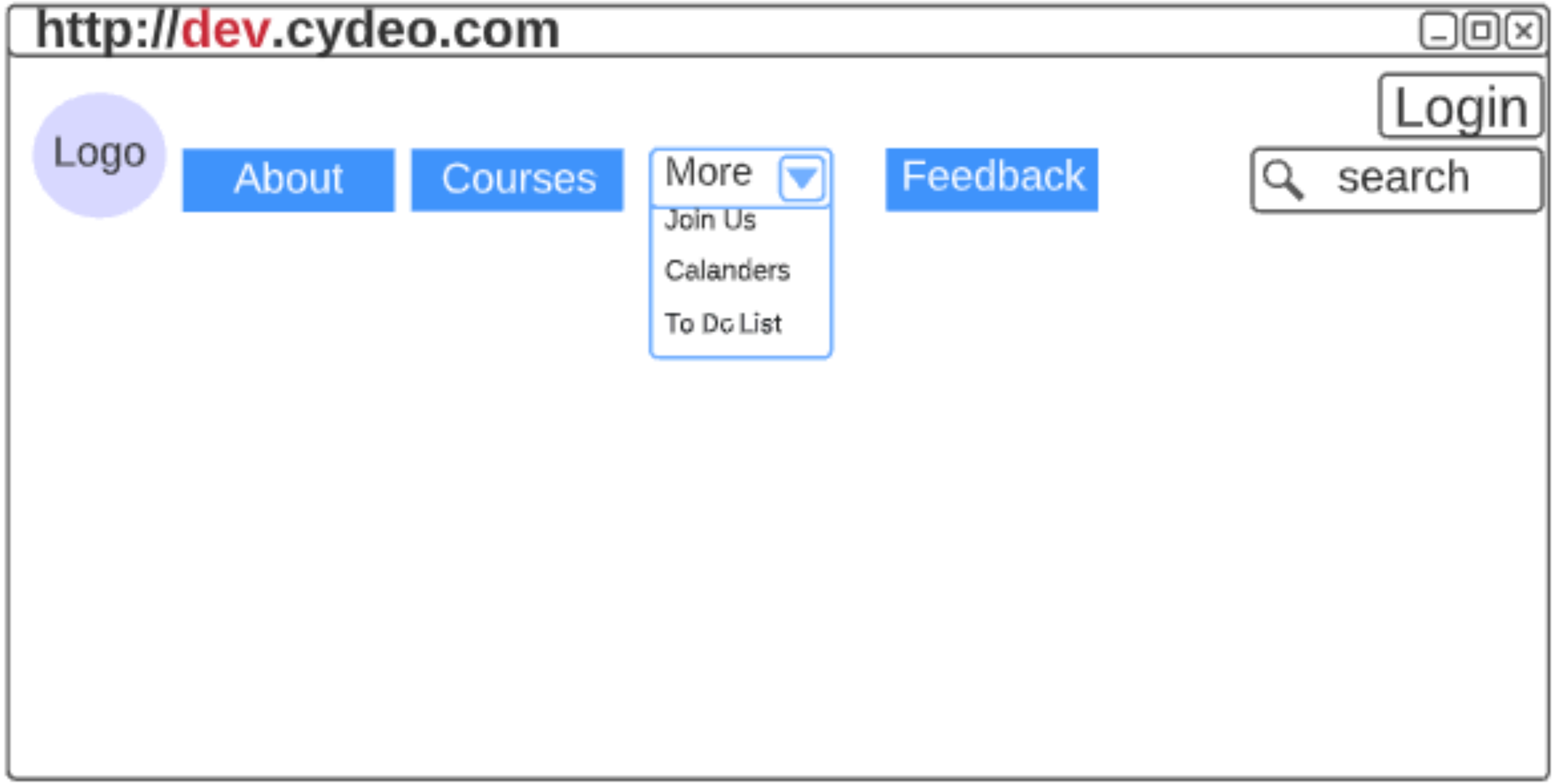
Software Environments

Definition of Environments

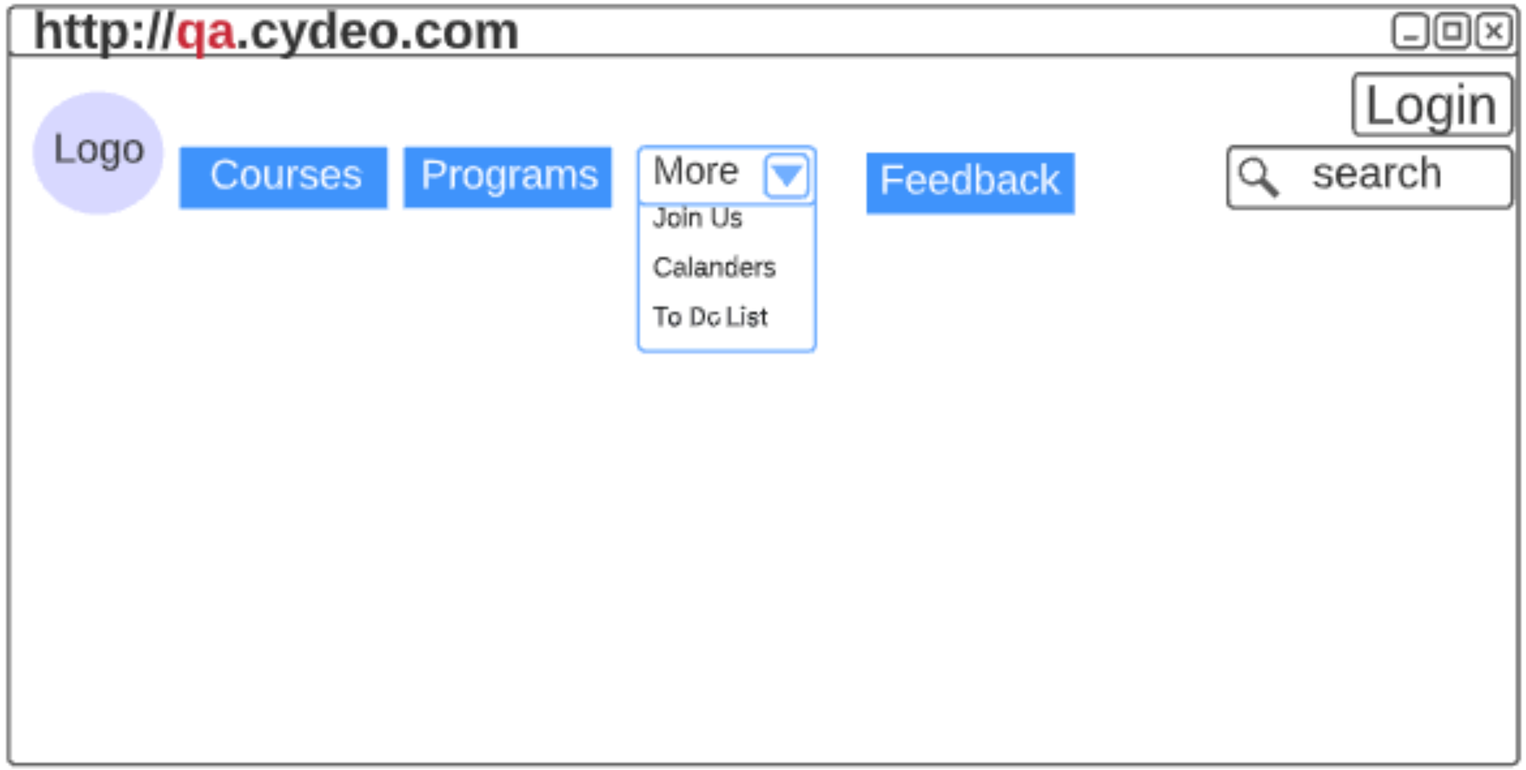
In simple, environments are **the different versions of the software**, and each version has **a different URL** and **purpose of use**. All the Environments are created by the developers.

There are **at least 4 environments for a software**:

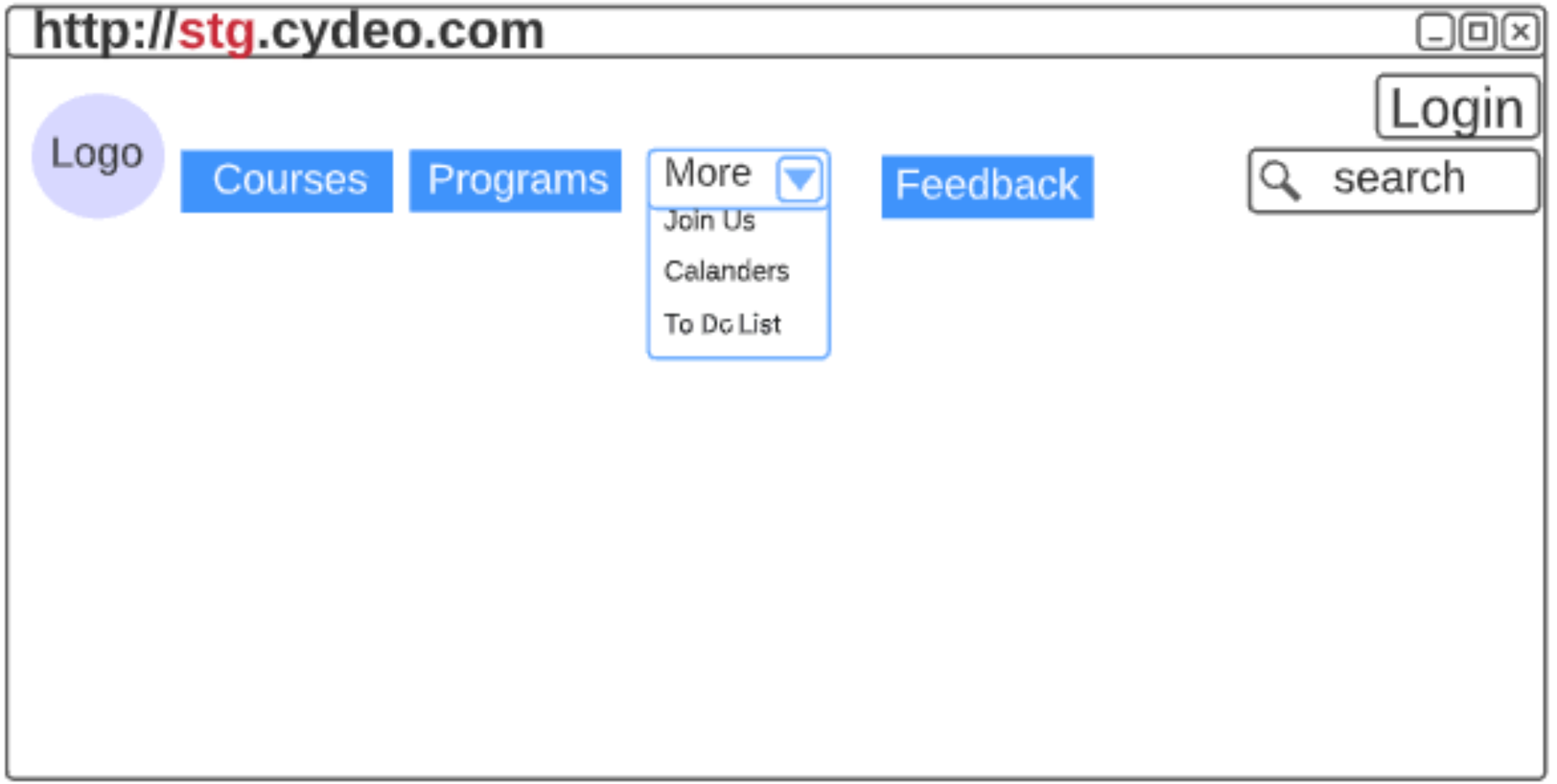
1. **Dev** (development) Environment
2. **QA** / **Test** Environment
3. **Staging** / **pre-profuction** Environment
4. **Production** Environment



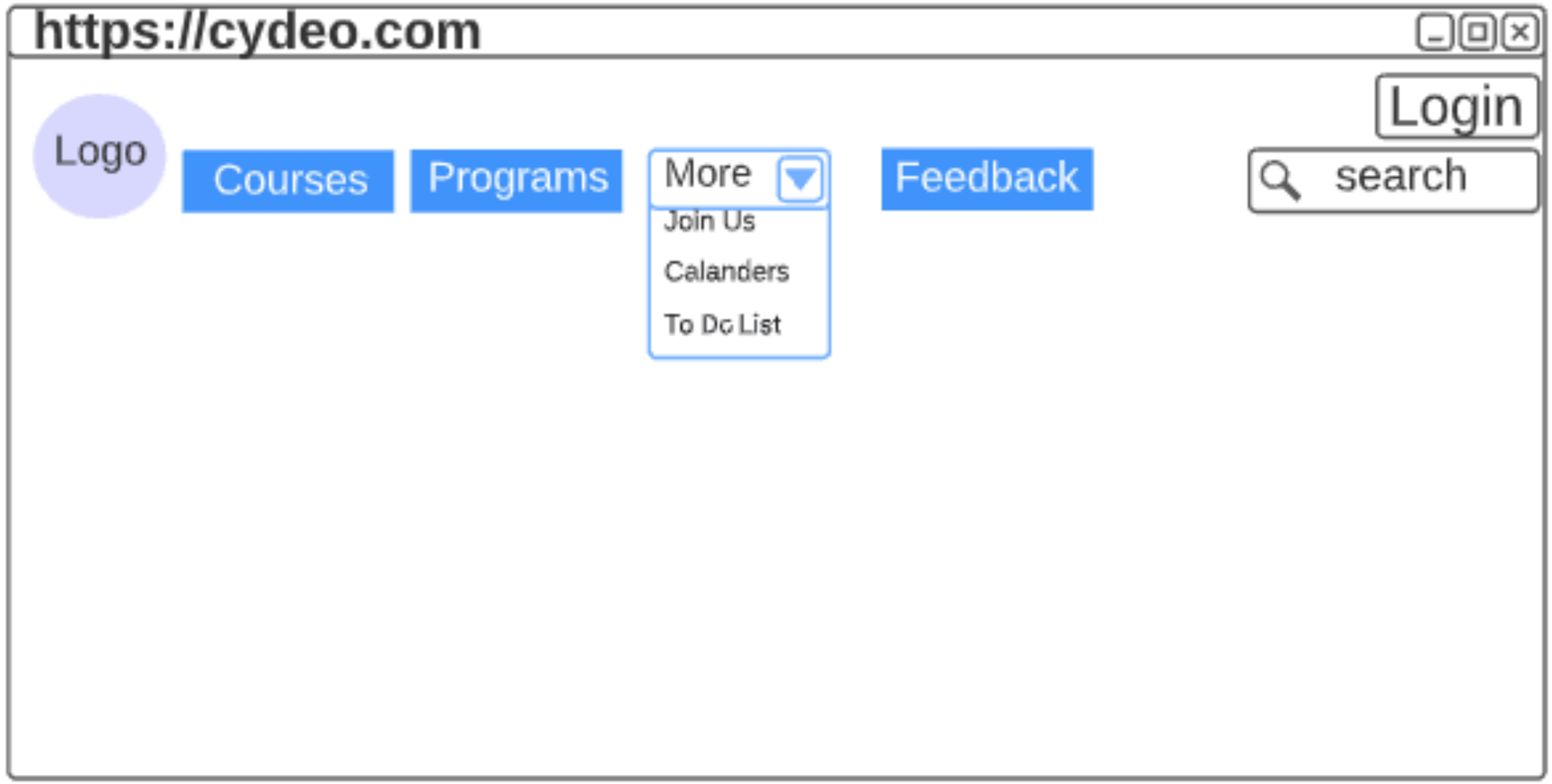
Dev Environment is for the developers to build the software



QA Environment is for the functional and non-functional testers to test the software



Staging Environment is for the client/stakeholders to test the app to eveluate if they accept the final result or not



Production Environment is for the end users for the real life