**Clase 1**

Bienvenidos y bienvenidas a la primera clase del curso de testing funcional y accesibilidad web.

**¿Qué veremos en esta clase?**

Esta clase la dedicaremos a conocer o retomar algunos conceptos que nos servirán de apoyo a lo largo del curso.

Si son conceptos que no conoces te recomendamos hacer una primera lectura y luego detenerte en aquellas cuestiones que te parezcan más relevantes para poder marcarlas.

Si algunos conceptos te son familiares podes hacer una única lectura.

El material lo encontrarán en Word (“doc”) para facilitar la navegación como también la edición, subrayado o resaltado. Si lo prefieren podrán descargar la clase también en formato “pdf”.

1. **Software y hardware**

Se conoce como **software​** al conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas dentro de un sistema informático. En contraposición a los componentes físicos que son llamados **hardware**.

*Comencemos con algunas definiciones:* a todo equipo informático lo llamaremos **hardware**, en el caso de una computadora, hablamos de la CPU, la pantalla, el teclado, mouse, cables, chips, memorias, etc. Mientras que llamaremos **software** a los programas que corren en nuestra computadora como así también al sistema operativo.

La interacción entre el software y el hardware hace operativo una computadora, un celular o una tablet. El software envía instrucciones u ordenes que el hardware ejecuta, haciendo posible su funcionamiento.

Todos los días utilizamos diferentes dispositivos más allá una computadora, esos dispositivos o sistemas son conocidos como equipos autónomos o sistemas embebidos o dedicados. Este tipo de equipamiento cuenta con un microprocesador o con un microcontrolador que es el “cerebro” de nuestro sistema, el cual puede poseer o no un sistema operativo.

Al igual que una computadora, posee memoria, almacenamiento, entradas y salidas pero se diferencia porque tiene un objetivo específico, por ejemplo: podemos pensar en los lectores de tarjeta del colectivo, en máquinas de café, bebidas o golosinas. En lugar de contar con un teclado alfabético y numérico, posee un teclado para seleccionar opciones o en lugar de un monitor, cuenta con una pantalla de menor tamaño o un conjunto de visores o display. Este tipo de sistemas suele convivir con otros de su mismo tipo o integrado a un ecosistema de diversos equipos, habitualmente dentro de una red.

Para poder indicar las acciones o pasos que debe realizar cualquier sistema, debemos **programarlo**, es decir, indicar a través de un lenguaje común que es lo que debe hacer.

1. **¿Qué es un programa?**

Un programa es un conjunto de pasos lógicos escritos en un lenguaje de programación que nos permite realizar una tarea específica.

El programa suele contar con una interfaz de usuario, es decir, un medio visual mediante el cual interactuamos con la aplicación. Algunos ejemplos son la calculadora, el navegador de internet, un teclado en pantalla para el celular, etc.

Hoy encontramos programas o aplicaciones que pueden ejecutarse en una computadora, notebooks, tablets y celulares. Estas aplicaciones pueden ser escritas en **diferentes lenguajes de programación**. Como ejemplos encontramos C, Java, PHP, Python, entre otros. Estos programas corren sobre un sistema operativo, por ejemplo, Windows, Linux, Mac OS y Android entre otros.

Los programas para poder correr se deben cargar en la memoria, el responsable de esta tarea es el **sistema operativo**. Un programa puede diseñarse para una computadora o para otro tipo de dispositivos pero su programación suele realizarse en una computadora utilizando un entorno de desarrollo integrado (en ingles IDE).

1. **¿Cómo se clasifican los programas / sofware?**

Son tres los grandes grupos en los que se divide este concepto informático:

1. **Software de sistema:** son elementos que permiten el mantenimiento del sistema en global: sistemas operativos, controladores de dispositivos, servidores, utilidades, herramientas de diagnóstico, de corrección y optimización.
2. **Software de programación:** se trata de diferentes alternativas y lenguajes para desarrollar programas de informática: editores de texto, compiladores, intérpretes, enlazadores, depuradores, entornos de desarrollo integrados (IDE).
3. **Software de aplicación**: le permite a los usuarios llevar a cabo una o varias tareas específicas en cualquier campo de actividad: aplicaciones ofimáticas, para control de sistemas y automatización industrial, software educativo, software empresarial, bases de datos, telecomunicaciones (Internet), videojuegos, software médico, software de diseño asistido (CAD) entre otros.
4. **¿Qué es un sistema operativo?**

Un sistema operativo es un conjunto de programas que permite manejar la memoria, disco, medios de almacenamiento de información y los diferentes periféricos o recursos de nuestra computadora, como son el teclado, el mouse, la impresora, la placa de red, entre otros.

Los periféricos utilizan un driver o controlador y son desarrollados por los fabricantes de cada equipo. Encontramos diferentes sistemas operativos como Windows, Linux, MAS OS, en sus diferentes versiones. También los teléfonos y tablets poseen un sistema operativo.

Dentro de las tareas que realiza el sistema operativo, en particular, se ocupa de gestionar la memoria de nuestro sistema y la carga de los diferentes programas, para ello cada programa tiene una prioridad o jerarquía y en función de la misma contará con los recursos de nuestro sistema por más tiempo que un programa de menor prioridad.

El sistema operativo se ocupa también de correr procesos. Llamamos proceso a la carga en memoria de nuestro programa, si no está cargado en memoria nuestro programa simplemente “no corre”.

Podemos distinguir dos grandes tipos de procesos:

1. **De primer plano:** requieren de la interacción del usuario, es el caso de un navegador web, un editor de texto, un programa de diseño de imágenes.
2. **De segundo plano:** son aquellos programas que no requieren del usuario y habitualmente no poseen una interfaz gráfica o “pantalla”. Un ejemplo de este tipo de procesos es el anti-virus.



1. **Los lenguajes de programación**

Tal como vimos en esta primera clase, el software se escribe en un lenguaje de programación.

Estos lenguajes poseen una sintaxis, que al igual que el castellano, nos permiten expresar una idea, donde esa idea tendrá diferentes componentes o partes.

Un lenguaje de programación es una forma de comunicarnos con una computadora, tablet o celular e indicarle qué queremos hacer.

Existen distintos tipos de lenguaje: principalmente de **bajo nivel** y de **alto nivel**. La diferencia se encuentra en lo cerca o lejos que estemos del hardware de nuestro equipo. Esta cercanía tiene que ver con el control que tengamos sobre el dispositivo, placa o controlador.

Encontramos diferentes lenguajes como C, C++, Java, PHP, Python, C#, ASP, entre otros.

Los programas habitualmente se diseñan o escriben a través de un entorno de desarrollo integrado (IDE por sus siglas en ingles).

Un IDE habitualmente cuenta con un editor de texto (donde escribiremos nuestro código), el compilador, linker / enlazador y el debugger / depurador para el lenguaje que estemos utilizando y la correspondiente plataforma en la que luego correrá.

Existen otras herramientas que **facilitan** nuestro trabajo, por ejemplo para documentar o llevar registro de lo que hacemos (doxygen), para compartir nuestro trabajo y realizarlo en forma colaborativa (SVN / GIT). De esta forma, un equipo de trabajo puede desarrollar diferentes partes de un programa y luego integrarlas en forma más simple.

1. **¿Qué es el software libre?**

El software libre promueve que los usuarios puedan ejecutar, copiar, distribuir, estudiar, modificar y así mejorar un software para beneficio propio y de la comunidad. Un software libre no tiene porqué ser gratuito.

Para ser considerado como libre un software debe cumplir cuatro libertades:

* Poder utilizar el programa con cualquier propósito.
* Contar con la posibilidad de estudiar el programa a través de su código fuente y cambiarlo en función de necesidades o intereses propios.
* Poder distribuir copias del software a otros.
* Poder distribuir o compartir la versión modificada a otras personas.

Obviamente para poder cumplir con estos requisitos, es necesario poder acceder al código fuente.

1. **¿Cómo es el proceso de creación del software?**

El proceso de creación de software puede llegar a ser muy complejo, dependiendo de sus características y criticidad. Por ejemplo, la creación de un sistema operativo es una tarea que requiere proyecto, gestión, numerosos recursos y todo un equipo de trabajo. En el otro extremo, si se trata de un programa simple como una calculadora, este puede ser realizado por un solo programador.

Los procesos de desarrollo de software deben sistematizarse ya que en caso contrario lo más seguro es que el proyecto no logre concluir o termine sin cumplir los objetivos previstos, y con variedad de fallas inaceptables. Cuando un proyecto fracasa, rara vez es debido a fallas técnicas, la principal causa de fallos y fracasos es la falta de aplicación de una buena metodología o proceso de desarrollo.

El proceso de desarrollo puede involucrar numerosas y variadas tareas,​ desde tareas administrativas, pasando por el personal técnico y hasta la gestión y el gerenciamiento.

Podemos organizarlas de la siguiente forma:

* Especificación y análisis de requisitos
* Diseño
* Codificación
* **Pruebas / Testing**
* Instalación y paso a producción
* Mantenimiento

Nuestra tarea en este curso consistirá en conocer las diferentes fases de desarrollo, como se conforman los equipos de trabajo, como se relacionan entre sí y particularmente dedicarlos a las pruebas de funcionamiento. Para ello es necesario entender que es un requisito o requerimiento.

1. **Requisitos**

En la ingeniería de software, un requisito es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Establecen qué debe hacer el sistema, pero no cómo hacerlo.

Entonces, ¿qué es un requisito?

* Condición o capacidad que un usuario necesita para poder resolver un problema o lograr un objetivo
* Condición o capacidad que debe exhibir o poseer un sistema para satisfacer un contrato, estándar, especificación, u otra documentación formalmente impuesta.
* Una condición o capacidad que debe ser conformada por el sistema.
* Algo que el sistema debe hacer o una cualidad que el sistema debe poseer.

1. **Tipos de requisitos**
2. **Requisito funcional**

Un requisito funcional puede ser una descripción de lo que un sistema debe hacer. Este tipo de requisito especifica algo que el sistema entregado debe ser capaz de realizar.

1. **Requisito no funcional**

Un requisito no funcional especifica algo sobre el propio sistema, y cómo debe realizar sus funciones. Algunos ejemplos de aspectos solicitables son la disponibilidad, el testeo, el mantenimiento, la facilidad de uso, etc.

1. **Características de un requisito**

Los requisitos bien formulados deben satisfacer varias características. Si no lo hacen, deben ser reformulados hasta hacerlo.

* **No ambiguo:** el texto debe ser claro, preciso y tener una única interpretación posible.
* **Conciso:** debe redactarse en un lenguaje comprensible por los inversores en lugar de uno de tipo técnico y especializado, aunque aun así debe referenciar los aspectos importantes.
* **Consistente:** ningún requisito debe entrar en conflicto con otro requisito diferente, ni con parte de otro.
* **Completo:** los requisitos deben contener en sí mismos toda la información necesaria, y no remitir a otras fuentes externas que los expliquen con más detalle.
* **Alcanzable:** un requisito debe ser un objetivo realista, posible de ser alcanzado con el dinero, el tiempo y los recursos disponibles.
* **Verificable:** Se debe poder verificar con absoluta certeza, si el requisito fue satisfecho o no.

Estas características suelen ser subjetivas, es decir, no pueden ser calculadas de forma automática por ningún sistema. Es por ello, que se tiende a utilizar métricas o indicadores que sí que pueden ser calculados de forma automática y que, de algún modo, pueden contribuir a ponderar las características mencionadas.

**Felicitaciones! Llegaste al final de la primera clase! Nos vemos en el foro y recorda participar del encuentro sincrónico.**