

Bienvenidxs a la [UNAHUR](#)

[video de presentación](#)

Fue creada a través de la Ley 27.016 en diciembre del año **2014** y comenzó a funcionar en **2016**.

3 funciones: **Docencia, Investigación y Extensión.**

Rector: Lic. Jaime Perczyk. **Vicerrector:** Mg. Walter Wallach

Director del **Instituto de Tecnología e Ingeniería:** Ing. Gustavo Medrano

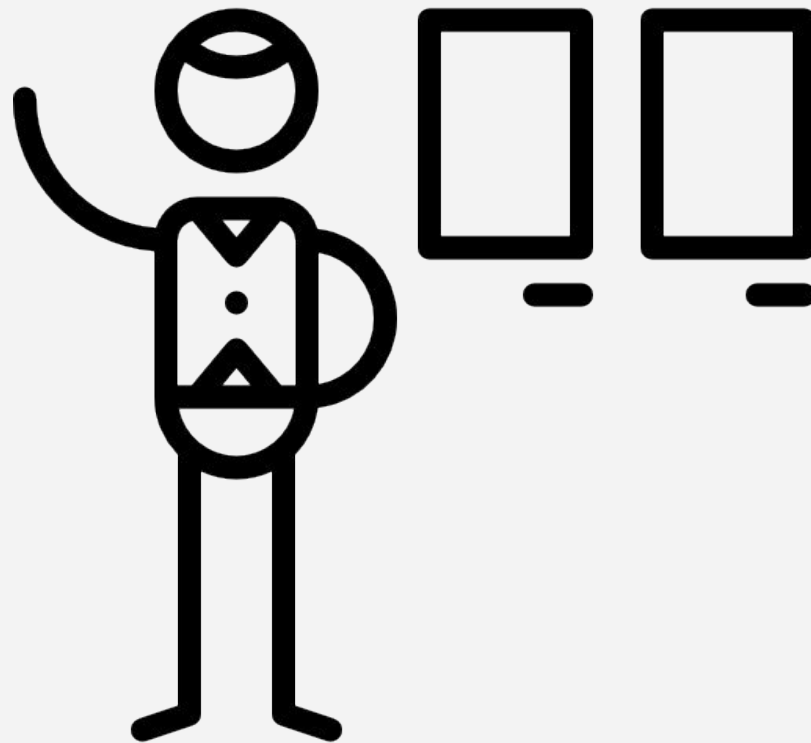
Director de **Tecnicatura Universitaria en Informática:** Ing. Fernando Puricelli

INTRODUCCIÓN AL TESTING

Objetivos

- Conocer el propósito del **software**
- Comprender qué es el **testing** en términos de identificar las diferencias entre una **necesidad** y el **producto** que las resuelve.
- Distinguir en términos generales, distintos **casos de prueba** para una aplicación.

Presentación personal



SOBRE USTEDES

Este taller es **por y para ustedes**. Queremos conocerlos un poco más para ir así adaptando el taller a sus inquietudes, ganas y expectativas.

- **Nombre**
- **Formación**
- **¿Por qué te interesó este curso?**
- **¿Qué esperas aprender?**

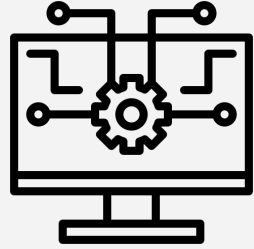


Presentación del Aula en el [Campus UnaHur](#)

- Perfil personal
- ¿Dónde está el curso?
- Cartelera de Avisos.
- Mensajes privados.



SOFTWARE



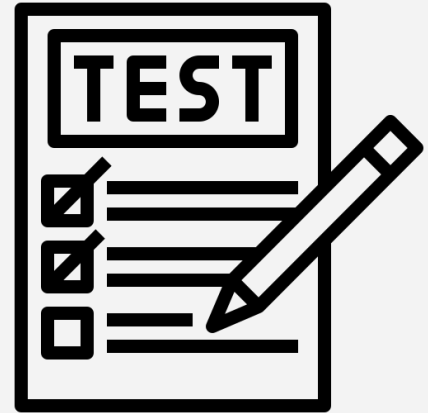
Propósito

- Hay una necesidad de un cliente / usuario y el software aparece como **producto / solución / herramienta** de esa necesidad.
- Todo software tiene un fin. En base a una necesidad surge el software.
- Está en todos lados.
- Es lo que uno **no puede tocar** de una compu. En contraposición, el hardware es todo lo que uno puede tocar.

¿ QUÉ ES EL TESTING? **

Las **pruebas de software** (*testing*) son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar **información** objetiva e independiente sobre la **calidad del producto** a la parte interesada (la que tiene la necesidad). Es una *actividad* más en el proceso de control de calidad.

Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Existen distintos modelos de desarrollo de software, así como modelos de pruebas.



** fuente: Wikipedia

¿Qué es un tester?

Es un **oficio** (como el carpintero, panadero) pero en la industria del software.

- Es quien se encarga de **verificar si una aplicación funciona** como debe. Si cumple con la necesidad **para la que fue creada**.
- Que se cumpla todo lo que está especificado en el documento que explica cómo debería funcionar el software (lo veremos más adelante).

Un tester es un **investigador**. Básicamente se dedica a investigar si un software es de calidad o presenta errores o problemas, siempre pensando en las personas que lo utilizarán.

Para ser un buen tester es necesario desarrollar un conjunto amplio de **habilidades**, como por ejemplo la comunicación, trabajo en equipo, pensamiento crítico, poder de observación, ser meticuloso y detallista, tener conocimiento del negocio (si es una aplicación de ventas, ¡conocer de ventas!), de la aplicación que se prueba y de cómo planificar, diseñar, ejecutar las pruebas.

¿Qué es un tester?

Ejemplos de tareas que hace el tester:

- ¿La aplicación funciona para MP3 y WAV?, ¿la lectora de cd?, ¿qué pasa si funciona con el WAV, qué pasa si no?
- Si la aplicación no cumple con la necesidad pedida, se debe informar al desarrollador. Veremos cuáles son los mecanismos para esto.

¿ Qué es una FUNCIONALIDAD ?

Los **requerimientos funcionales** definen las funciones que el sistema será capaz de realizar – o **funcionalidades** – por ejemplo el manejador-de-compras online tiene entre sus funcionalidades “permitir buscar productos por nombre”, otra de sus funcionalidades es de poder “comprar un producto seleccionado”.

REQUERIMIENTOS

En un proceso de desarrollo de software vamos a encontrar a un cliente y un proveedor:

- **El Cliente** - que es quien tiene una necesidad y es quien solicita el desarrollo del software
- El **equipo de proyecto** (donde estamos nosotros los testers) - que somos quienes vamos a participar en el proceso de desarrollo de ese software para satisfacer dicha necesidad.
 - es muy importante que el cliente y el equipo de proyecto se puedan poner de acuerdo sobre qué va a recibir el cliente cuando el proyecto termine, es decir, como va a funcionar esa aplicación – o cuales son los requerimientos del software.
 - Para ello en el equipo de proyecto vamos a encontrar alguien que cumple la función de “analista de negocios” que es quien se encarga de hablar con el cliente, capturar lo que el cliente necesita y definir los requerimientos.
 - Las formas más comunes en que el Analista define requerimientos son las siguientes:
 - Especificación Funcional
 - Historias de Usuario

Es importante que se logre la mayor claridad posible en cuanto a los requerimientos, debido a que en el desarrollo de software generalmente hay mucha gente involucrada por lo que se vuelve complicada y compleja la comunicación entre todos y asegurarse que frente a un requerimiento dado todos entienden la misma cosa.

Testing de software: ¿Para qué?

El **software** surge por construir algo en base a una necesidad: hay un proceso por el cual la “nada” se transforma en software.

Etapas del proceso de construcción

1. Necesidad
2. Análisis
3. Construcción
4. Aplicación

Testing = comparar 1 con 4

El ***testing*** es una actividad que verifica que una aplicación resuelve la necesidad original que le dio origen y en los casos de no ser cumplida, se identifican las diferencias.

PRÁCTICA

- Pensar en distintos objetos / elementos y elijan uno (ejemplo una pelota, guitarra, etc)
- Identificar necesidad (en base a una necesidad se crea el objeto) / objetivo de ese elemento
- contar como verificarán que se cumpla esa necesidad (si la necesidad es entretener, podemos decir que el objeto pelota cumplió la necesidad para la que fue creada, que es entretener)