

## Condiciones de corrección y aprobación

1. Cumplir con las reglas de estilo.
2. Cumplir con lo visto en clase.
3. Se deberán manejar posibles Excepciones. Si el programa tiene excepciones no controladas, se cortará la corrección en ese momento, siendo el resultado la desaprobación.
4. Deberá tener al menos un formulario.
5. No deberán estar distintas partes del proyecto contenidas en un único formulario (por ejemplo, la carga de datos e impresión de información de los registros deberían ser 2 formularios separados).
6. Deberá utilizar todos los temas vistos entre la clase 10 y 20 (se tomará como referencia el Classroom, no importa cómo se vieron en el aula).
7. Desarrollar funcionalidades para un sistema de atención de público:
  - a. Deberá proveerse un contexto y tipo de empresa, así como una dinámica realista.
  - b. Se deberá poder cargar personas/animales (al menos 2 diferentes) desde distintas fuentes (archivos, base de datos, a mano) y registrar datos de su atención.
  - c. Se deberá poder generar la atención de las entidades.
  - d. Se deberá poder importar y exportar información, o consultar alguna fuente, o dar datos de entrada, o similar.
  - e. Darle el enfoque que crean correcto, teniendo en cuenta que el sistema tiene que permitir el ingreso y atención del público.
8. Debe estar bien documentado y acompañado por un documento PDF que explique brevemente la funcionalidad pretendida.
9. En dicho PDF deberá indicarse por cada tema utilizado del Punto 6 de este documento (dependiendo la instancia en la que se encuentren), y marcar donde se encuentra su implementación, a fines de facilitar la corrección. Realizar el mismo comentario en el Summary del código entregado.
10. Para la entrega definitiva, deberá tener otro proyecto de Test Unitarios probando al menos 2 funcionalidades creadas.
11. Si DOS o más proyectos se parecen en lo más mínimo, serán desaprobados.
12. Quienes no cumplan **TODAS** estas condiciones, no serán evaluados, siendo el resultado la desaprobación del trabajo.

Generar una Solución nombrada como: *Apellido.Nombre.Division.TPFinal*

Publicar dentro de GIT en la carpeta correspondiente para cada entrega.