

Programación Web 3

UNLaM - Tecnicatura en Desarrollo Web

2020 - 1C

Trabajo Práctico
“Ayudando en la pandemia”

1. Objetivo	3
2. Equipo	3
3. Requisitos Técnicos	3
3.1 Proyecto .NET	3
3.2 Base de Datos	3
3.3 Estilos	3
3.4 JavaScript	4
3.5 HTML	4
3.6 Validación	4
3.7 Arquitectura y Consideraciones de Desarrollo	5
4. Objetivo del Proyecto	6
5. Especificación Funcional	6
5.1 Pantalla Principal (/ o /home/inicio)	6
5.1.a Menú, Partial Views y Layouts	7
5.2 Ingresar/Salir	7
5.3 Registro de Usuario	8
5.4 Home	9
5.4.1 Búsqueda de necesidades	9
5.4.2 Mis Necesidades	9
5.5 Crear una necesidad	10
5.6 Detalle de necesidad	11
5.6.1 Valoración de una necesidad	11
5.6.2 Denunciar una necesidad	11
5.6.3 Realizar una donación	12
5.7 Modificar una necesidad	12
5.8 Mi perfil	13
5.8.1 Historial de Necesidades	13
5.9 Usuario Administrador	14
5.9.1 Gestión de denuncias	14
5.10 Error - /home/error	14
5.14 Diagrama de Tablas	15
6. Forma de Entrega	16
7. Condiciones de Aprobación	16

1. Objetivo

Este documento describe el alcance funcional y los requisitos técnicos del trabajo práctico que los alumnos de la materia Programación Web III deberán aprobar a fin de estar en condición de aprobar/promocionar la materia.

2. Equipo

El equipo para realizar el trabajo práctico deberá ser de **4 alumnos**

3. Requisitos Técnicos

3.1 Proyecto .NET

1. El trabajo práctico deberá ser realizado utilizando ASP.NET MVC y Entity Framework. El tipo de proyecto a utilizar es una aplicación web MVC. En caso de incluirse web service se deberá utilizar ASP.NET MVC Web Api.

3.2 Base de Datos

1. Se deberá utilizar Sql Server (como mínimo 2008 Express).
2. La cátedra proveerá el script de Base de Datos con las tablas y se debe usar para el TP. En caso de que consideren que falta algo o que no cumple con alguna funcionalidad requerida, consultar a los profesores.
3. Se adjunta diagrama (ver [5.14 Diagrama de Tablas](#))

3.3 Estilos

1. No se permitirán que se utilicen los estilos ya provistos por Microsoft en la aplicación de ejemplo que provee Visual Studio.
2. Todos los archivos .css deberán estar dentro de una carpeta.
3. No utilizar estilos inline (atributo style="") ni definir estilos dentro de una pagina (tags <style>).
4. Debe de utilizarse algún framework/biblioteca de hojas de estilo. Algunos ejemplos:
 - a. Twitter Bootstrap (<http://getbootstrap.com/> , temas <http://bootswatch.com/>)

- b. Foundation (<http://foundation.zurb.com/docs/>)
- c. KickStart (<http://www.99lime.com/elements/>)
- d. Bulma (<http://bulma.io/>)
- e. Otro definido por los alumnos y validado con el cuerpo docente.

3.4 JavaScript

1. No utilizar JavaScript inline dentro de una página, se deberá referenciar a archivos js.
2. Todos los archivos .js deberán estar dentro de una carpeta.
 - a. Si se decide utilizar algún js que no es propio, el mismo deberá estar dentro de una subcarpeta.
3. Se deberá utilizar Bundle and minification
(<https://geeks.ms/etomas/2012/07/30/bundles-en-asp-net-mvc4/>)

Recomendaciones

Las funciones específicas de una página, deberían estar en un archivo .js con el mismo nombre de la página

Aquellas funciones utilizadas en más de una página, deberían de estar dentro de otro archivo .js de uso común.

3.5 HTML

1. No utilizar tags table para organizar el contenido de una página en columnas, los tags table solo están permitidos para representar una grilla/listado de información.
2. Se requiere el uso de Layouts para estructura los formularios web de la aplicación. Dentro del BaseLayout deberán referenciarse las hojas de estilo y archivos de javascript de uso común por toda la aplicación.
3. Debe utilizarse HTML5.

3.6 Validación

1. Utilizar validaciones tanto del lado del cliente (JavaScript) como del lado del servidor utilizando DataAnnotations. Comentario: No confundir con tener que desarrollar javascript nativo o utilizar frameworks de javascript (por ejemplo JQuery) para realizar validaciones del lado cliente. Los DataAnnotations ya proveen una solución para este punto.
2. Se puede utilizar una lista que detalle todos los campos que no cumplieron con las validaciones.
3. Para todos los campos validar que la cantidad de caracteres ingresados no exceda el

límite impuesto en la base de datos.

4. Para todas las páginas validar que no pueda acceder a páginas con datos privados de otro usuario o acceder a una página que solo se puede acceder estando logueado.

3.7 Arquitectura y Consideraciones de Desarrollo

1. La capa de acceso a datos deberá ser realizada con Entity Framework. Este componente de .NET será explicado en clases a fin de que los alumnos comprendan cómo utilizarlo.
2. La capa Web deberá ser realizada utilizando MVC.
3. En caso de que se requieran web services, deberán utilizarse MVC Web Api.
4. Deberán crear un repositorio privado en **Github** que deberá ser compartido con los profesores (usuarios **pablokuko**, **matipazw**, **juizmariano**) y donde irán subiendo los avances.
5. Utilizar la menor cantidad posible de código en los /Controllers/[Entidad]Controller, e intentar que en los mismos haya llamadas a métodos dentro de otro proyecto que contenga las reglas de negocio.
6. Compatibilidad con exploradores.
 - a. Google Chrome (la última versión para Windows).

4. Objetivo del Proyecto

Desarrollar una aplicación web para las necesidades sociales generadas por la pandemia del COVID-19. Personas registradas en la aplicación web podrán publicar sus necesidades solicitando ayuda o donaciones. Las donaciones podrán ser monetarias o de bienes para ayudar a la contraparte. Las necesidades deberán tener información multimedia así como referencias de terceros para darle credibilidad evitando proyecto fake.

Los usuarios podrán votar el proyecto y agregar comentarios sobre el mismo, después de que se realizó la ayuda para darle una valoración al usuario que lo posteo generando reputación.

5. Especificación Funcional

Páginas/Funcionalidad

- 1. Página Principal (/ o /inicio)**
 - a. Menú, Partial Views y Layouts**
 - b. Ingresar (/ingresar)**
- 2. Home (/home)**
- 3. Mis Necesidades**
- 4. Crear una necesidad**
- 5. Mi Perfil**
- 6. Error (/home/error)**

5.1 Pantalla Principal (/ o /home/inicio)

En la página principal se divide en 3 secciones:

- Nombre de la aplicación web.
- Un link a la pantalla de login de usuarios. (ver 5.2)
- Un link a la pantalla de registro de usuario (Ver 5.3)
- Las 5 principales necesidades activas más valoradas por los usuarios del sitio.

Por cada una de las necesidades se deberá visualizar la siguiente información:

- nombre
- foto principal
- usuario creador.
- valoración actual de los usuarios.
- fecha de creación.
- un link para ir al detalle de la necesidad.

Se deberá validar si el usuario está logueado ya que para visualizar el detalle de la necesidad se requerirá. En caso que no lo esté, se deberá redireccionar al login de usuarios. Una vez logueado el usuario, será redirigido, de forma directa, al detalle de la necesidad.

5.1.a Menú, Partial Views y Layouts

Menú (solo logueado)

- Home (/necesidades/home)
- Crear Necesidad (/necesidades/crear)
- Mi Perfil (/perfil)
- Denuncias (/denuncias) Sólo el usuario administrador
- Acerca de (/acerca-de)
 - **tip:** usar atributo **ActionName** para permitir esta url.
- Salir (visible en caso de estar logueado, /necesidades/salir).

Partial View Integrantes

Crear una Partial View **pv_Footer.cshtml** donde estará el footer con la información:
Programación Web 3 20201C - Grupo {nro grupo} - {integrantes}

Crear los Layouts BaseLayout, AnonimoLayout, NecesidadesLayout

En **BaseLayout.cshtml** se debe poner todo el html compartido (links a css, js, etc, usando [Bundle](#)).

AnonimoLayout.cshtml debe implementar BaseLayout.cshtml y será implementada por la página de Ingresar y Error.

NecesidadesLayout.cshtml debe implementar BaseLayout.cshtml, la misma tiene el menú visible y la partial view **pv_Footer.cshtml**.

5.2 Ingresar/Salir

La pantalla contendrá un campo email y contraseña. Al ingresar en el menú del sitio estará visible un link de logout que al clickearlo borrará los datos de sesión y redirigirá a la página de Inicio.

- **Email** - Requerido y Formato de email.
- **Contraseña** - Requerido.
- **Botón “Ingresar”**. En caso de que se haya accedido a la página de ingresar a través de otra página automáticamente, luego de hacer click en “Ingresar”, se deberá redirigir a la página donde se deseaba ingresar, caso contrario se redirigirá a la página de inicio.

Ejemplo: si un usuario deseaba ingresar a la página de Necesidades, el sitio lo redirige a la página de Ingresar, luego de ingresar se deberá redirigir a la página de Preguntas automáticamente.

- Mostrar “Email y/o Contraseña inválidos” en caso de que no se encuentre la combinación de usuario y contraseña ingresada.

Al momento de loguearse, se deberá validar que el usuario esté activo. En caso de que no, el usuario no podrá ingresar mostrándole un mensaje indicando “su usuario está inactivo. Actívalo desde el email recibido”.

5.3 Registro de Usuario

Un usuario se podrá registrar en la plataforma. Para ello se requerirán los siguientes datos:

- email (**html id y name=Email**)
- password (**html id y name=Password**)
- confirmación de password (**html id y name=RePassword**)
- fecha de nacimiento (**html id y name=FechaNacimiento**)

Todos los datos son obligatorios.

Se debe validar los siguientes puntos:

- el email debe cumplir el formato adecuado.
- el password debe contener al menos una letra mayúscula, un número y no debe ser menor a 8 caracteres.
- el password y la confirmación de password deben coincidir al momento de la registración.
- el usuario debe ser mayor de 18 años para poder utilizar la plataforma.

Una vez registrado se deberá enviar un email de confirmación para que el usuario confirme que el email indicado, le pertenece. Adicionalmente, el usuario será creado en estado inactivo.

Dentro del email existirá un link al sitio web de la plataforma que modificará el estado del usuario a activo. un ejemplo de URL es <http://misitio.com/usuarios/activar?token=324234242>, donde el parámetro token es un código almacenado dentro del usuario que permitirá identificarlo y activar la cuenta.

El token deberá ser autogenerado al momento del registro.

Existirán 2 tipos de usuarios: administrador y normal. Todos los usuarios que se registran serán usuarios normales. El usuario administrador será único y estará cargado a partir de la base de datos para simplificar la gestión de usuarios.

5.4 Home

Al ingresar, luego del login, el usuario será redirigido a una home donde podrá visualizar sus necesidades activas y ver las necesidades del resto de los usuarios de la plataforma.

Cada ítem contendrá la siguiente información de las necesidades:

- nombre
- usuario
- foto principal
- fecha de fin
- porcentaje de aceptación (cant me gusta / total *100. Si 8 usuarios pusieron me gusta, y 2 no me gusta, tendrá 80% de aceptación)
- un link para ir al detalle de la necesidad

5.4.1 Búsqueda de Necesidades

Existirá un buscador de necesidades donde el usuario podrá buscar por un texto libre. Este texto libre se buscará en relación al nombre de las necesidades existentes o bien según el nombre del usuario creador de la necesidad.

El listado de necesidad vendrá ordenado por fecha más cercana de cierre de necesidad y, luego, por mayor valoración de la necesidad. El resultado de la búsqueda no deberá incluir sus propias necesidades ya que las mismas siempre se visualizarán dentro de la home en una sección especial (ver 5.4.2 Mis Necesidades).

Cada necesidad se visualizará en un recuadro donde se verá el contenido mencionado en el punto 5.4. Home.

5.4.2 Mis Necesidades

Dentro de la Home, debajo del buscador, el usuario verá una sección con sus necesidades activas con un checkbox tildado de “solo visualizar activas”. O sea, aquellas que fueron creadas por él y aún no están finalizadas pudiendo recibir donaciones. En caso de que se destilde, se visualizarán todas las del usuario sin importar si están o no finalizadas.

Adicionalmente, si hay necesidades, podrá ser colapsada para visualizar de mejor forma el resultado de la búsqueda que aparece inmediatamente debajo de esta sección.

Cada necesidad tendrá un link/botón para ir a detalle de necesidad (donde están las donaciones) y además otro link para editar la necesidad.

5.5 Crear una necesidad

Desde la home, el usuario tendrá la opción de Crear una Necesidad. Será redirigido a una nueva pantalla donde podrá crear una necesidad completando la siguiente información:

- Nombre de la necesidad (obligatorio) (**html id y name=Nombre**)
- Descripción de la necesidad (obligatorio) (**html id y name=Descripcion**)
- Fecha de Fin (obligatorio) (**html id y name=FechaFin**)
- Teléfono de Contacto (obligatorio) (**html id y name=TelefonoContacto**)
- Tipo de Donación (monetaria, insumos) (obligatorio) (**html id y name=TipoDonacion**)

En caso de que haya elegido monetaria, el usuario ingresará:

- la cantidad de dinero que necesita (obligatorio). (**html id y name=CantDinero**)
- el CBU/Alias de la cuenta donde será recibido el dinero (obligatorio). (**html id y name=CBUAlias**)

En caso de que haya elegido insumos, el usuario ingresará:

- los ítems que necesita. Por cada uno, detalla cantidad y descripción del ítem. La cantidad de ítems es ilimitada y se deberá cargar al menos 1 elemento.

- Foto (obligatorio) (**html id y name=Foto**)
- Referencias (obligatorio) Se deberán ingresar 2 referencias de contacto que den veracidad a la necesidad.

Por cada referencia se deberá ingresar el nombre y teléfono de contacto.

Las necesidades sólo podrán contener un tipo de donación.

Cada necesidad deberá tener la información del tipo de donación solicitada.

Al momento de crear la necesidad se deberá almacenar el usuario creador y la fecha de creación.

Un usuario podrá tener a lo sumo 3 necesidades activas. Caso que desee generar una cuarta necesidad, no deberá permitirse.

Si el usuario aún no completó los datos obligatorios de Mi perfil (ver 5.8 Mi Perfil), no podrá crear una necesidad y deberá visualizarse un mensaje, en la parte superior, con un link a Mi Perfil para que pueda completarlo.

5.6 Detalle de necesidad

El detalle de la necesidad permitirá visualizar la información completa de una necesidad pudiendo conocer más información de la solicitud. Toda la información se visualizará en modo solo lectura.

Para el caso del tipo de donación, se deberá visualizar el detalle de lo que el usuario ha cargado según el tipo de donación seleccionado. La forma de presentación será necesidad por cada equipo y conversada con los profesores. Importante, cada tipo de donación requiere una forma de visualización diferente a fin de brindar una experiencia de usuario adecuada.

Debajo de la donación que compone la necesidad, se deberá visualizar los totales de las donaciones ya realizadas por los usuarios. La información se deberá visualizar de forma totalizada sin mostrar el detalle de cada donación puntual.

Adicionalmente, se deberá visualizar la valoración actual de la necesidad, nombre de usuario y fecha de creación.

Desde esta pantalla los usuarios podrán realizar 3 acciones. Estas acciones sólo podrán ser realizadas por los usuarios que no han creado la necesidad.

5.6.1 Valoración de una necesidad

Los usuarios podrán valorar la marcando con Me Gusta o No Me Gusta. Esta valoración será registrada y permitirá calcular el porcentaje de aceptación de cada necesidad.

Al momento de registrar la valoración se deberá calcular el porcentaje de aceptación guardándolo como dato dentro de la necesidad.

5.6.2 Denunciar una necesidad

Los usuarios podrán denunciar una necesidad. Al presionar Denunciar, serán redirigidos a una nueva pantalla donde deberán ingresar la siguiente información:

- motivo de denuncia. Los posibles motivos son : fraude, indebida, violación de derechos, contiene información política.
- comentarios.

Ambos datos son obligatorios y serán registrados junto al usuario y fecha y hora de la denuncia. Si la necesidad alcanza 5 denuncias diferentes quedará en un estado de revisión por parte del administrador y no se visualizará dentro de las búsquedas o home del sitio.

Las denuncias generadas serán evaluadas por el perfil administrador, quién podrá bloquear la necesidad, luego de una validación manual. O bien, desestimar las denuncias registradas.

5.6.3 Realizar una donación

Como usuario es posible realizar una donación en una necesidad abierta. Para ello existirá un link que redireccionará al usuario a una pantalla donde podrá registrar la donación. En esta pantalla se presentarán, según el tipo de donación, los campos para que el usuario registre la donación deseada.

Todos los campos del tipo de la donación son obligatorios y deberán ser registrados junto con la fecha de creación y el usuario.

En caso de que la donación sea del tipo monetaria:

- se deberá visualizar el total requerido y el total restante
- el usuario ingresará la cantidad de dinero que donará (obligatorio)
- el usuario adjuntará comprobante de transferencia o depósito (obligatorio)

En caso de que sea de insumos:

- se deberá visualizar los ítems requerido y cantidad restante (aquellos que ya se completaron, deberán estar en otro color o tachados o deshabilitados (a elección de los alumnos)
- el usuario deberá ingresar los ítems que donará y por cada uno, detalla cantidad. Se deberá cargar al menos 1 elemento.

5.7 Modificar una necesidad

Los usuarios que creen necesidades podrán modificar su contenido mientras que la necesidad no haya finalizado (fecha de fin). En caso de que estuviera finalizada, la necesidad es de sólo lectura.

Se podrán modificar todos los datos de una necesidad exceptuando el tipo de donación. El tipo de donación deberá ser mantenido permitiéndose modificar los ítems de la donación (ver 5.4 Crear una necesidad).

5.8 Mi perfil

El usuario logueado podrá acceder a su perfil donde deberá cargar/modificar la siguiente información respecto a su identidad:

- nombre (obligatorio)
- apellido (obligatorio)
- fecha de nacimiento (obligatorio)
- foto de perfil (obligatorio).

El nombre de usuario será definido a partir de la primer carga de los datos del perfil. Se deberá concatenar el nombre + el apellido como nombre de usuario separados por un punto. En caso de que el nombre de usuario ya exista (por ejemplo: marcos.perez) se le deberá sumar un valor numérico que surgirá del último nombre igual más 1. Ejemplo: si el nombre de usuario marcos.perez1 ya existe el nombre del nuevo usuario deberá ser marcos.perez2.

Una vez definido el nombre del usuario, no se deberá volver a definir quedando guardado en el registro del usuario en la base de datos.

El email y nombre de usuario se visualizarán, al principio de la pantalla, pero no podrán ser modificados por el usuario.

Se deberá validar los siguientes puntos:

- el usuario deberá ser mayor de 18 años.

5.8.1 Historial de Donaciones

Dentro de Mi Perfil, en la parte superior derecha, habrá un link donde el usuario será redireccionado a la pantalla del Historial de sus Donaciones.

Al ingresar a esta opción, el usuario verá el listado de sus necesidades ordenadas de la más reciente a la más antigua. Los datos a visualizar son:

- fecha de donación.
- nombre.
- tipo de donación.
- estado (abierto, cerrado)
- total recaudado.
- mi donación
- link al detalle (ver 5.6 Detalle Necesidad).

Esta información deberá ser accedida a través de un web service REST y visualizada según el párrafo anterior. La información podrá ser visualizada usando javascript ajax consumiendo el web service o se consume el web service desde el controller (como si fuese una api externa) y se renderiza directamente en la vista.

5.9 Usuario Administrador

El usuario administrador tendrá las mismas opciones que los usuarios normales sumándose la gestión de denuncias.

5.9.1 Gestión de denuncias

Cuando una denuncia es generada el administrador podrá revisarlas para aprobar o desestimar. En el extremo derecho aparecerá un link de Gestión de Denuncias y, al presionarlo, el administrador será redireccionado hacia una nueva pantalla. Esta pantalla tendrá el listado de las denuncias pendientes de revisión ordenadas de forma descendente desde la más antigua a la más nueva.

El listado contendrá los siguientes campos:

- fecha de creación.
- motivo
- link hacia el detalle de la necesidad (ver 5.6 Detalle de Necesidad).
- comentarios
- Botón Desestimar
- Botón Aceptar

El botón desestimar cambiará el estado de la denuncia a desestimada sin afectar a la necesidad vinculada.

El botón Aceptar cambiará el estado de la denuncia a aceptada generando un cambio de estado en la necesidad a denunciada. Al estar denunciada, la necesidad no será visible en la home del sitio ni en los resultados de búsqueda de la plataforma.

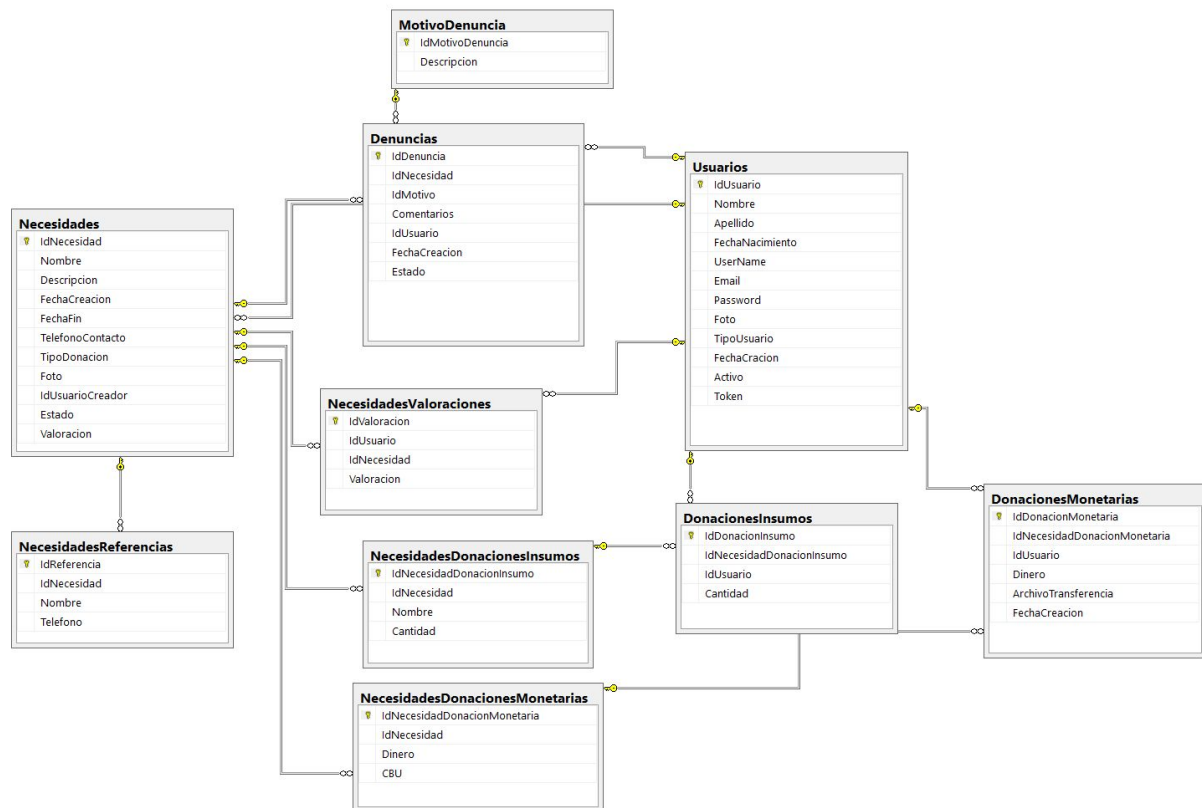
5.10 Error - /home/error

Todos los errores no capturados deberán redirigir a esta pantalla con el fin de no mostrarle al usuario detalles de código.

(ref: <http://anexsoft.com/p/99/implementando-un-custom-error-page-con-asp-net-mvc>)

5.11 Diagrama de Tablas

Deberán utilizar la base de datos y tablas entregadas por la cátedra (el script se subirá próximamente)



6. Forma de Entrega

El TP deberá ser enviado a los siguientes destinatarios, copiando a todos los integrantes del equipo en el email enviado.

- a) mpazwasiuchnik@unlam.edu.ar
- b) pnsanchez@unlam.edu.ar
- c) el resto de los integrantes que participaron del tp (como CC)

El email deberá contener los siguientes puntos:

- a. El asunto del email será GrupoX-(Apellido1_Apellido2_Apellido3_Apellido4)
- b. Respuestas a las siguientes preguntas:
 - i. ¿Qué nota creen que deberían sacar en el tp? (1-10, donde para 7 debe estar toda la funcionalidad pedida) y ¿por qué?
 - ii. ¿Qué cosas creen que podrían mejorarse?
 - iii. ¿Qué les resultó más complicado?
- c. Dar acceso al repositorio a los usuarios de Github de los profesores (**pablokuko**, **matipazw**, **juizmariano**). El permiso deberá permitir clonar el repositorio localmente a fin de revisar el código.

7. Condiciones de Aprobación

- 1. El TP deberá cumplir con todos los requisitos técnicos y funcionales definidos. No se aceptará que alguna funcionalidad o requisito técnicos no se encuentren plasmados en el trabajo práctico.
- 2. Al momento de la entrega del trabajo práctico, todos los alumnos de cada equipo deberán estar presentes para la defensa del trabajo práctico.
- 3. En la defensa del trabajo, se evaluará el grado de conocimiento y participación en el desarrollo del trabajo práctico de cada alumno. Si algún alumno no participó activamente del trabajo práctico, no aprobará el mismo.