

# Evaluación Parcial 2

## Los recursos son finitos

Sigla	Nombre Asignatura	Tiempo Asignado	% Ponderación
PGY4121	Programación de Aplicaciones móviles	5 horas	40%

### 01. Agente evaluativo

<input checked="" type="checkbox"/>	Heteroevaluación	<input type="checkbox"/>	Coevaluación	<input type="checkbox"/>	Autoevaluación
-------------------------------------	------------------	--------------------------	--------------	--------------------------	----------------

### 02. Tabla de Especificaciones

Resultado de Aprendizaje	Indicador de Logro (IL)	Ponderación Indicador Logro
RA2	IL2.1 Genera una experiencia de usuario en las aplicaciones, favoreciendo la navegación, usabilidad y las prestaciones del equipo.	25%
	IL2.2 Maneja el estado y los datos de la aplicación utilizando memoria interna Stores y persistencia de datos para dar solución a los requerimientos del cliente.	25%
	IL2.3 Programa la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	25%
	IL2.4 Conecta con API para la programación de consultas síncronas y/o asíncronas con persistencia de datos entregando una mejor solución a los requerimientos del cliente.	25%

### 03. Instrucciones para el/la estudiante

Esta es una evaluación que corresponde a una **entrega de encargo con presentación** y tiene un **40%** de ponderación sobre la nota final de la asignatura.

El tiempo para desarrollar la presentación de esta evaluación es de 15 minutos y se realiza en el **Taller de Computación Móvil**.

#### Instrucciones generales:

- Los equipos de trabajo tendrán todo el semestre para desarrollar el proyecto completo. Cada experiencia de aprendizaje finaliza con 1 evaluación sumativa correspondiente a los contenidos de dicha experiencia.
- Cada evaluación, contará con funcionalidades que se irán incluyendo al proyecto, es decir, el desarrollo será de forma incremental.
- El tiempo para desarrollar la Presentación es de 10 minutos por equipo, más 5 minutos para una posible ronda de preguntas.

#### Entregables

1. Una vez desarrollado la solución se debe comprimir el proyecto excluyendo las carpetas e2e y node\_modules y enviarlo de forma digital al docente.
2. El equipo de trabajo debe presentar la aplicación desarrollada con apoyo de recursos gráficos donde deben presentar el contexto, problema y la solución generada aplicando un lenguaje técnico con el apoyo del siguiente material complementario 1.4.7 MC\_[Nombre de aplicación] Grupo N°.pptx

Con relación a la presentación, ésta tendrá una duración máxima de 15 minutos, dejando 5 minutos para una posible ronda de preguntas que realizará el docente de la asignatura y/o los otros equipos de trabajo.

La Presentación (diapositivas en formato ppt, pptx, odp)

- Contenidos
- Contexto
- Problemática
- Solución propuesta
- Principales características del producto desarrollado
- Presentación del producto desarrollado.
- Reflexión

Los trabajos entregados fuera de plazo obtendrán la calificación mínima.

## Caso

En el contexto actual de la pandemia que a nivel mundial se está viviendo, han surgido una serie de situaciones por las que las instituciones de educación superior han tenido que sobrellevar. Una de ellas es el hecho de tener una continuidad en la educación llevando las clases de la modalidad presencial a la modalidad remota. Sin embargo, con el paso de los meses y con el actuar de distintos actores, a saber, autoridades sanitarias, colaboración con entidades comunales, cuidados y responsabilidad de la ciudadanía, entre otros, ha llevado a volver a cierta presencialidad.

Este retorno a las actividades en forma presencial ha generado una serie de otras dificultades, las cuales han quedado en evidencia con el pasar de los días. Una de estas dificultades es el retorno de los alumnos a sus hogares al término de la jornada vespertina.

Los motivos de estas dificultades son variados, sin embargo, se pueden identificar las más comunes:

- Falta de transporte público en ese horario.
- Alto costo de servicios de transporte particular (Uber, taxi, etc.).
- No contar con movilización propia.
- Falta de transporte facilitado por la institución de educación.



Con el fin de terminar con esta problemática en forma definitiva, se ha contactado a alumnos de Duoc UC para que desarrollen una Aplicación Móvil **TeLlevoAPP** capaz de ser utilizada en plataformas tanto como Android e iOS que permita que los alumnos se organicen a través de la App para que puedan retornar a sus hogares utilizando vehículos de algunos alumnos que si tienen movilización propia.

### La lógica de la solución sería la siguiente:

1. Alumnos que SI cuentan con movilización propia ingresan a la App para programar su viaje de retorno y fijan el costo del viaje por persona.

2. Alumnos que NO cuentan con movilización propia ingresan a la App para verificar si hay vehículos con capacidad para transportarlos.
  - a) Si no hay capacidad de transporte, vuelven al paso 2.
  - b) Si hay capacidad de transporte, solicitan viajar
3. Los alumnos se reúnen para viajar y pagan su tarifa.
4. Comienza el viaje.

Los principales objetivos de la App son:

- Reducir los tiempos destinados a encontrar movilización de retorno a hogares.
- Fomentar el compañerismo entre alumnos.
- Instar a los alumnos a asistir presencialmente a clases.  
Reducir la huella de carbono.

## Evaluación Sumativa 2

### Requisitos específicos para desarrollar en esta entrega:

Para el desarrollo de la entrega solicitada, los equipos de trabajo deben desarrollar los siguientes aspectos:

- Optimizar las páginas de Ingreso de Usuario, Restablecer Contraseña y Página de Inicio con las siguientes características:
- Implementar la navegación optimizada, velando por la seguridad de los Page ante los accesos no autorizados de usuarios. **(Semana 1).**
- Implementar el diseño según sea el caso generando la lógica de los modelos en los .ts del Page o Component. **(Semana 1).**
- Implementar el servicio que les permita realizar las consultas hacia el API Rest junto con las funciones permitidas por el servicio **(Semana 1).**
- Ya teniendo el servicio implementado, se deberá implementar las funciones desarrolladas en los Page y Componente que corresponda tanto para casos de éxito como casos de error **(Semana 2).**

○ Aplicar la correcta Persistencia (**Semana 2**).

- Los direccionamientos de una página a otra deben ser validados correctamente.
- Además, la lógica para verificar roles de la App (dueño del auto, pasajero) no se considera en esta Evaluación.
- La App debe ser capaz de generar el viaje con horario de salida y costo de viaje por persona.
- Con relación a los Casos de Uso, se necesita para esta entrega:
- Ingreso de Usuario
  - Alumno: Ingresar Nombre de Usuario
  - Alumno: Ingresar Contraseña
  - Alumno: Presionar botón “INGRESAR”
  - Sistema: Lo lleva a la Página de Inicio
- Restablecer Contraseña
  - Alumno: Ingresar Nombre de Usuario
  - Alumno: Presionar botón “RECUPERAR”
  - Sistema: Lo lleva a la Página de Ingreso de Usuario
- Página de Inicio
  - Sistema: Saluda al usuario con un label indicando “Bienvenido [nombre\_usuario]”

**Características del producto y del equipo de desarrollo en esta entrega**

- El equipo de trabajo deberá analizar el caso entregado y desarrollar un proyecto Ionic Angular, en el cual deben implementar su interfaz gráfica siguiendo los Mockups considerando las funcionalidades solicitadas e implementando la navegación y animaciones, de acuerdo con los requerimientos del caso.
- El sistema debe ser construido en su totalidad por los integrantes del equipo, intercambiando opiniones y buenas prácticas de programación.

**Restricciones en esta entrega**

- La aplicación debe ser adaptativa a dispositivos móviles con sistema operativo Android e iOS.

# Pauta de Evaluación

## Pauta tipo: Escala de valoración / Rúbrica

Categoría	% logro	Descripción niveles de logro
Muy buen desempeño	100%	Demuestra un desempeño destacado, evidenciando el logro de todos los aspectos evaluados en el indicador.
Desempeño aceptable	60%	Demuestra un desempeño competente, evidenciando el logro de los elementos básicos del indicador, pero con omisiones, dificultades o errores.
Desempeño incipiente	30%	Presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador, por lo que no puede ser considerado competente.
Desempeño no logrado	0%	Presenta ausencia o incorrecto desempeño.

Indicador de Logro	Categorías de Respuesta					Ponderación del Indicador de Evaluación
	Muy buen desempeño 100%	Buen desempeño 80%	Desempeño aceptable 60%	Desempeño incipiente 30%	Desempeño no logrado 0%	
IL2.1 Genera una experiencia de usuario en las aplicaciones, favoreciendo la navegación, usabilidad y las prestaciones del equipo.	Genera una experiencia de usuario en el 100% de la aplicación, favoreciendo la navegación, usabilidad y las prestaciones del dispositivo.	Genera una experiencia de usuario en el 80% de la aplicación, favoreciendo la navegación, usabilidad y las prestaciones del dispositivo.	Genera una experiencia de usuario en el 60% de la aplicación, favoreciendo la navegación, usabilidad y las prestaciones del dispositivo.	Genera una experiencia de usuario en el 30% de la aplicación, favoreciendo la navegación, usabilidad y las prestaciones del dispositivo.	No genera una experiencia de usuario en la aplicación, favoreciendo la navegación, usabilidad y las prestaciones del dispositivo.	25%
IL2.2 Maneja el estado y los datos de la aplicación utilizando memoria interna Stores y persistencia de datos para dar	Maneja en un 100% el estado y los datos de	Maneja en un 80% el estado y los datos de	Maneja en un 60% el estado y los datos de	Maneja en un 30% el estado y los datos de	No maneja el estado y los datos de la	25%

solución a los requerimientos del cliente.	la aplicación utilizando memoria interna Stores y persistencia de datos para dar solución a los requerimientos del cliente.	la aplicación utilizando memoria interna Stores y persistencia de datos para dar solución a los requerimientos del cliente.	la aplicación utilizando memoria interna Stores y persistencia de datos para dar solución a los requerimientos del cliente.	la aplicación utilizando memoria interna Stores y persistencia de datos para dar solución a los requerimientos del cliente.	aplicación utilizando memoria interna Stores y persistencia de datos para dar solución a los requerimientos del cliente.	
IL2.3 Programa la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	Programa en un 100% la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	Programa en un 80% la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	Programa en un 60% la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	Programa en un 30% la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	No programa la navegación de la aplicación considerando la performance en el uso de los componentes del dispositivo para una mejor solución a los requerimientos del cliente.	25%
IL2.4 Conecta con API para la programación de consultas síncronas y/o asíncronas con persistencia de datos entregando una mejor solución a los requerimientos del cliente.	Conecta con APIs para la programación de consultas síncronas y/o asíncronas usando persistencia de datos en el 100% de la aplicación, para una mejor solución de los requerimientos del cliente en la	Conecta con APIs para la programación de consultas síncronas y/o asíncronas usando persistencia de datos en el 80% de la aplicación, para una mejor solución de los requerimientos	Conecta con APIs para la programación de consultas síncronas y/o asíncronas usando persistencia de datos en el 60% de la aplicación, para una mejor solución de los requerimientos del cliente en	Conecta con APIs para la programación de consultas síncronas y/o asíncronas usando persistencia de datos en el 30% de la aplicación, para una mejor solución de los requerimientos	No conecta con APIs para la programación de consultas síncronas y/o asíncronas usando persistencia de datos en la aplicación, para una mejor solución de los requerimientos	25%



	aplicación móvil.	del cliente en la aplicación móvil.	la aplicación móvil.	del cliente en la aplicación móvil.	del cliente en la aplicación móvil.	
	<b>Total</b>					<b>100%</b>