



# Trabajo Práctico Anual - Diseño de Sistemas

[Objetivos Generales:](#)

[Contexto](#)

[Dominio de alto nivel](#)

[Requerimientos de alto nivel](#)

[Entregas](#)

[Entrega 0A - Análisis del Dominio \[Opcional\]](#)

[Entrega 0B - Prueba de Concepto - Tecnología \[Recomendada\]](#)

[Entrega 1 \[Obligatoria\]](#)

[Cálculo de cercanía](#)

[Cálculo de disponibilidad](#)

[Búsqueda de puntos](#)

[Requerimientos detallados](#)

[Entrega 2 \[Obligatoria\]](#)

[Caso 1: Búsqueda de puntos de interés de nuestro sistema](#)

[Caso 2: CGPs](#)

[Caso 3: Bancos](#)

[Requerimientos detallados](#)



## Objetivos Generales:

- Tomar decisiones de Diseño de una Solución Compleja identificando Ventajas y Desventajas.
- Comunicar el Diseño utilizando distintas herramientas (UML - código/pseudocódigo - prosa - DER).
- Identificar Requerimientos No Funcionales (Atributos de Calidad) y su impacto en la Arquitectura.
- Resolver Situaciones Problemáticas del Diseño utilizando Patrones de Diseño.
- Diseñar Artefactos de Prueba que permitan Validar los Requerimientos definidos.
- Construir un Modelo de Datos persistente acorde a la Implementación.
- Diseñar y Construir Interfaces de Usuario en Distintas Tecnologías.
- Diseñar la Comunicación con una Interfaz Externa al Sistema.
- Validar el Diseño propuesto utilizando herramientas de Testing.
- Comprender y Especificar la Arquitectura del Producto.
- Implementar la Solución en una Tecnología concreta para Validar el Diseño Propuesto.
- Comprender que la tecnología condiciona al diseño.
- Utilizar Herramientas de Soporte para el Diseño y Construcción de Software:
  - o control de versiones
  - o entorno de desarrollo integrado
  - o herramientas de manejo de dependencias
  - o herramientas de construcción
  - o herramientas de concepción, y comunicación de diseño
- La Comunicación y el Trabajo en Equipo como elementos primordiales en el Proceso.



## Evaluación y Condiciones de Aprobación

- El Equipo: 4 a 6 Integrantes.
- Se evaluará en cada Entrega al Equipo y a cada Integrante.
- La aprobación será por Entrega Grupal y por Integrante.
- Se le informará al Equipo y a cada Integrante la Calificación Obtenida.
- La Calificación puede ser:
  - Insuficiente
  - Suficiente
  - Muy Bueno
  - Excelente
- El Alumno/Equipo podrá tener solamente 2(dos) Entregas con calificación Insuficiente que podrá ser recuperada en 2(dos) Instancias. En caso de obtener una Tercer Entrega con calificación Insuficiente el Alumno/Equipo no aprobará el Trabajo Práctico condición necesaria para la aprobación de la Asignatura.



## Contexto

El gobierno de la ciudad, con el objetivo descentralizar el acceso a la información y disminuir la congestión en centros de Gestión y Participación (CGPs), quiere llevar a cabo un proyecto ambicioso: contar con dispositivo táctiles en múltiples esquinas de la ciudad, conectados a internet, que permitan realizar trámites y obtener información.

Tras contactarse con la Regional Buenos Aires, de la Universidad Tecnológica Nacional., y recibir la propuesta en la Cátedra de Diseño de Sistemas, luego de varias reuniones y discusiones sobre cómo abordar el proyecto, todas las partes están de acuerdo en encarar una primera etapa en la que estos dispositivos sólo visualizarán puntos de interés<sup>1</sup>, en algunas locaciones particulares. Si el proyecto tiene éxito<sup>2</sup>, en etapas posteriores se incluirán más dispositivos y más funcionalidades.

## Dominio de alto nivel

Un punto de interés (*POI*, por sus siglas en inglés) es básicamente una posición en un mapa, que suele representarse mediante un ícono.

De este conocemos su dirección, calle principal, entre qué calles se encuentra, número, y datos más específicos como el piso, departamento, unidad, el código postal, localidad, barrio, provincia, país, además de las coordenadas geográficas resultado del proceso de geocodificación: latitud y longitud.

Algunos puntos de interés posibles son:

- Paradas de colectivos
- Centros de Gestión y Participación (CGP)
- Sucursales de Bancos
- Locales comerciales



## Requerimientos de alto nivel

Cada uno de estos dispositivos, más allá de sus particularidades físicas, es una computadora conectada a Internet. Los mismos serán provistos por otra empresa, "Touch.me".

---

<sup>1</sup> [https://es.wikipedia.org/wiki/Punto\\_de\\_inter%C3%A9s](https://es.wikipedia.org/wiki/Punto_de_inter%C3%A9s)

<sup>2</sup> Un proyecto de estas características podría fallar por diversos motivos humanos y técnicos, no relacionados al software en sí: roturas, robos, falta de conexión, problemas de energía, etc. Sin embargo, a fines del TP no serán considerados..



Nuestra responsabilidad será, entonces, diseñar, construir, probar y desplegar el sistema software que se ejecutará en estas máquinas.

El mismo cubrirá, al menos, las siguientes funcionalidades:

- Buscar puntos de interés, según criterios de búsqueda establecidos por el usuario
- Obtener información de un cierto punto
- Visualizar el resultado de las consultas
- Almacenar información de la consulta, con fines estadísticos.

Este sistema no sólo será utilizado por transeúntes en la vía pública, sino también por administradores del sistema, del Gobierno de la ciudad, que cargarán puntos desde distintos orígenes, y obtendrán estadísticas sobre las consultas.



# Entregas

## Entrega 0A - Análisis del Dominio [Opcional]

Para empezar, vamos a hacer un análisis preliminar del sistema: queremos identificar casos de uso y usuarios involucrados. Para ello, se solicita:

1. Modelar los Casos de Uso(CU) del dominio.
2. Modelar Diagramas de Secuencia y de Clases en UML para comunicar el Diseño de los CU.
3. Definir Atributos de calidad y Requerimientos No Funcionales en base al Contexto del Sistema.

## Entrega 0B - Prueba de Concepto - Tecnología [Recomendada]

Para empezar a empaparse en la tecnología, vamos a realizar una pequeña prueba de concepto de la misma: modelar los POIs. De los mismos nos interesá:

- Su coordenada: latitud y longitud
- Un nombre, dirección (calle y altura)
- Si es una parada de colectivos, un CGP, etc

Además, a un POI queremos conocer:

- Si se encuentra a menos de X metros de otro POI<sup>3</sup>
- Si es válido: no puede haber POIs sin geolocalizar ni a los que les falte nombre.

Se solicita:

1. Crear un proyecto en la tecnología orientada a objetos seleccionada
2. Utilizando un entorno de desarrollo integrado (IDE)
3. Mantener los cambios en un sistema de control de versiones

---

<sup>3</sup> Para este punto se puede realizar el cálculo a mano (<http://www.movable-type.co.uk/scripts/latlong.html>) o mejor, buscar y utilizar alguna biblioteca que lo resuelva.



## Entrega 1 [Obligatoria]

Queremos que las personas sean capaces de saber si pueden ir desde nos encontramos hacia otro lugar. Para ello vamos a ofrecerles tres servicios: consulta de cercanía, consulta de disponibilidad (horaria) y búsqueda de puntos.

### Cálculo de cercanía

En general, decimos que estamos cerca de un punto de interés si se encuentra a una distancia menor a 5 cuadras.

Sin embargo, hay excepciones:

1. Un parada de colectivo se considera cercana si estamos a menos de una cuadra.
2. Los CGP cumplen la condición de cercanía, si su coordenada está dentro de la zona delimitada por la comuna.
3. Para los locales comerciales, cada rubro define el radio de cercanía. Ejemplo: para las librerías escolares se considera cerca si está en un radio de 5 cuadras, para un kiosco de diarios en cambio son 2 cuadras, y así sucesivamente.

### Cálculo de disponibilidad

El cálculo de disponibilidad trata de saber si un POI está disponible en un cierto momento. Esto depende de cada tipo de POI:

1. El servicio de transporte de Colectivos está disponible a toda hora
2. Para los CGP:
  - a. si se ingresa un valor x (el nombre de un servicio), la fecha debe estar en el rango de atención. Ej: si es sábado no se atiende Rentas, no está disponible. Si es lunes a las 20hs tampoco se encuentra disponible este servicio. En cambio si es lunes a las 10:30hs sí se encuentra disponible.
  - b. si no se ingresa un valor x, considerar que exista al menos un servicio en el CGP que esté atendiendo en ese momento.
3. Para los Bancos, el mismo comportamiento que para los CGP pero considerando el horario de atención bancario (Lunes a Viernes de 10hs a 15hs)
4. Para los locales comerciales, considerar el horario de atención del local. Ejemplo: para Carrousel el horario de atención es de Lunes a Sábado de 10hs a 13hs y de 17hs a 20:30hs. Si son las 15 horas de un martes no debe estar disponible.



## Búsqueda de puntos

Finalmente, necesitamos buscar puntos en el mapa a partir de un texto libre, por ejemplo:

- Si se ingresa el texto "114", se debería encontrar las paradas del colectivo 114.
- Si se ingresa una palabra que corresponde a un rubro (por ejemplo "mueblería"), se deberían encontrar todos los cales de ese rubro.
- Si se ingresa una palabra clave, se debería encontrar todos los puntos que estén etiquetados mediante esa palabra clave.
- Si se ingresa parte de un servicio (por ejemplo "asesoramiento"), se deben encontrar todos CGPs que provean algún servicio parecido<sup>4</sup> (por ejemplo "asesoramiento legal")

## Requerimientos detallados

1. Determinar cuándo un punto de interés está cerca de una coordenada geográfica.
2. Dado un momento (fecha:hora:minutos:segundos) y un valor x, determinar si un negocio está disponible.
3. Buscar un punto de interés en base a un texto libre<sup>5 6</sup>
4. Diseñar e implementar los casos de prueba para cada uno de los 3 requerimientos anteriores.

---

<sup>4</sup> Como mínimo, si el texto de búsqueda está incluido en el nombre del servicio, debe encontrarlo. Si se desea realizar una búsqueda más sofisticada, se recomienda utilizar bibliotecas especiales que provean algoritmos como por ejemplo, [la distancia de Levenshtein](#)

<sup>5</sup> Si bien este requerimiento se puede resolver mediante un motor de búsqueda de texto libre (búsqueda full text search, como Lucene/ElasticSearch o Postgre Full Text Search), se espera que se resuelva este requerimiento empleando exclusivamente el paradigma de objetos.

<sup>6</sup> NO se espera que los datos sean persistentes en esta entrega.





## Entrega 2 [Obligatoria]

En esta iteración, la búsqueda por puntos de interés se vuelve más compleja:

- Por un lado, necesitamos agregar lo necesario al modelo de dominio para que un administrador pueda agregar, modificar y quitar puntos del mapa.
- Por otro lado, se incorporan nuevos orígenes de datos de puntos de interés, que se detallan a continuación:

### Caso 1: Búsqueda de puntos de interés de nuestro sistema

Muchos de los puntos de interés están almacenados en nuestro sistema. Esto no debe cambiar y seguir funcionando tal como hasta ahora.

### Caso 2: CGPs

Para consultar los CGP se dispone de un componente desarrollado en forma externa<sup>7</sup>. El servicio acepta como parámetro un string que puede corresponder a la calle o zona, y devuelve la lista de CGPs disponibles según el siguiente formato:

Lista de CGPs, un objeto de la clase CentroDTO

- int: número de la comuna (ej: 3)
- string: zonas que incluye (ej. "Balvanera, San Cristóbal" para la comuna 3)
- string: nombre del director
- string: domicilio completo del CGP (ej: Junín 521)
- string: teléfono del CGP (4375-0644/45)
- lista de "serviciosDTO": array de servicios que contiene
  - string nombre del servicio (ej: Atención ciudadana)
  - lista de "rangos servicio DTO": Array con días de servicio que contiene
    - int: número de día de la semana (ej: 1 = Lunes, 2 = Martes, etc.)
    - int: horario desde (9)
    - int: minutos desde (0)
    - int: horario hasta (18)
    - int: minutos hasta (0)

### Caso 3: Bancos

Para consultar las entidades bancarias se cuenta con un servicio desarrollado en forma externa<sup>8</sup>. El servicio acepta como parámetro un string que representa el nombre del banco y otro con un nombre de servicio, y devuelve los bancos que cumplen ambos criterios (AND). La respuesta es en el siguiente formato JSON:

---

<sup>7</sup> Consultá a tu docente para más detalles sobre cómo acceder a este componente externo.

<sup>8</sup> Consultá a tu docente para más detalles sobre cómo acceder a este componente externo



```
[
  { "banco": "Banco de la Plaza",
    "x": -35.9338322,
    "y": 72.348353,
    "sucursal": "Avellaneda",
    "gerente": "Javier Loeschbor",
    "servicios": [ "cobro cheques", "depósitos", "extracciones", "transferencias",
"créditos", "", "", "" ]
  },
  { "banco": "Banco de la Plaza",
    "x": -35.9345681,
    "y": 72.344546,
    "sucursal": "Caballito",
    "gerente": "Fabián Fantaguzzi",
    "servicios": [ "depósitos", "extracciones", "transferencias", "seguros", "", "", ""
  ]
}, ...
]
```

Se debe adaptar el resultado de la búsqueda creando o actualizando la lista de puntos de interés en el repositorio de la aplicación a diseñar (el caso 1).

## Requerimientos detallados

1. ABMC (Alta, Baja, Modificación, Consulta) de puntos del sistema
2. Consulta (Búsqueda) de puntos extendida: se debe lograr que cuando se realice una consulta, se busquen puntos en todos los orígenes de datos. Se debe diseñar teniendo en cuenta que nuevos orígenes de datos pueden incorporarse en el futuro.
3. Desarrollar casos de prueba para los puntos anteriores. Considerar:
  - a. El componente externo del caso 2 es *lento* (ejecutarlo es al menos 10 veces más lento que realizar la consulta contra nuestros datos)
  - b. El servicio externo del caso 3 es rápido, pero tiene un costo de servicio, que se imputa por cada pedido realizado al mismo.