TRABAJO PRACTICO

1º CUATRIMESTRE

2019

UTN – FACULTAD REGIONAL BUENOS AIRES

* **Título**: Gestión de Datos
* **Año**: 3°
* **Código**: 082030
* **Curso**: K3012

INTEGRANTES LEGAJO

COBOS, Bruno 159676-7

JUGO, German 158917-9

NIKCEVICH, Carlos Alexis 146698-7

VERDILE, Guillermo 158962-3

utn - frba

geStión de datos

/\*

Mail Asunto / NombreArchivo.zip123

TP1C2019 K3012 MACACO\_NOT\_NULL ¿6?

Cuerpo del Mail

Grupo: MACACO\_NOT\_NULL

Curso: K3012

Integrantes: <apellido>, <nombres> - <legajo>

COBOS, Bruno - 1596767

JUGO, German - 1589179

NIKCEVICH, Carlos Alexis - 1466987

VERDILE, Guillermo - 1589623

\*/

# Estrategia.pdf

\*Archivo PDF en donde se deberá explicar en forma detallada y extensa la estrategia utilizada para desarrollar el TP.

\*Debe incluir una descripción de todas las estructuras de datos relevantes utilizadas en el algoritmo, explicando la razón de la elección de dichas estructuras.

\*Cualquier consideración tomada o asumida deberá ser aclarada en este documento.

\*Se debe incluir un DER (legible y entendible) del modelo de datos creado con una explicación detallada de cada entidad, relaciones, claves primarias y foráneas, índices, stored procedures, triggers, vistas, etc, que será estrictamente necesario para la corrección del modelo, de ser posible también entregar el DER en formato de imagen PNG, JPEG, etc.

\*El archivo de estrategia deberá ser en formato PDF obligatoriamente, con carátula e índice. En caso de no cumplir esta condición, el TP será rechazado sin evaluar

Índice

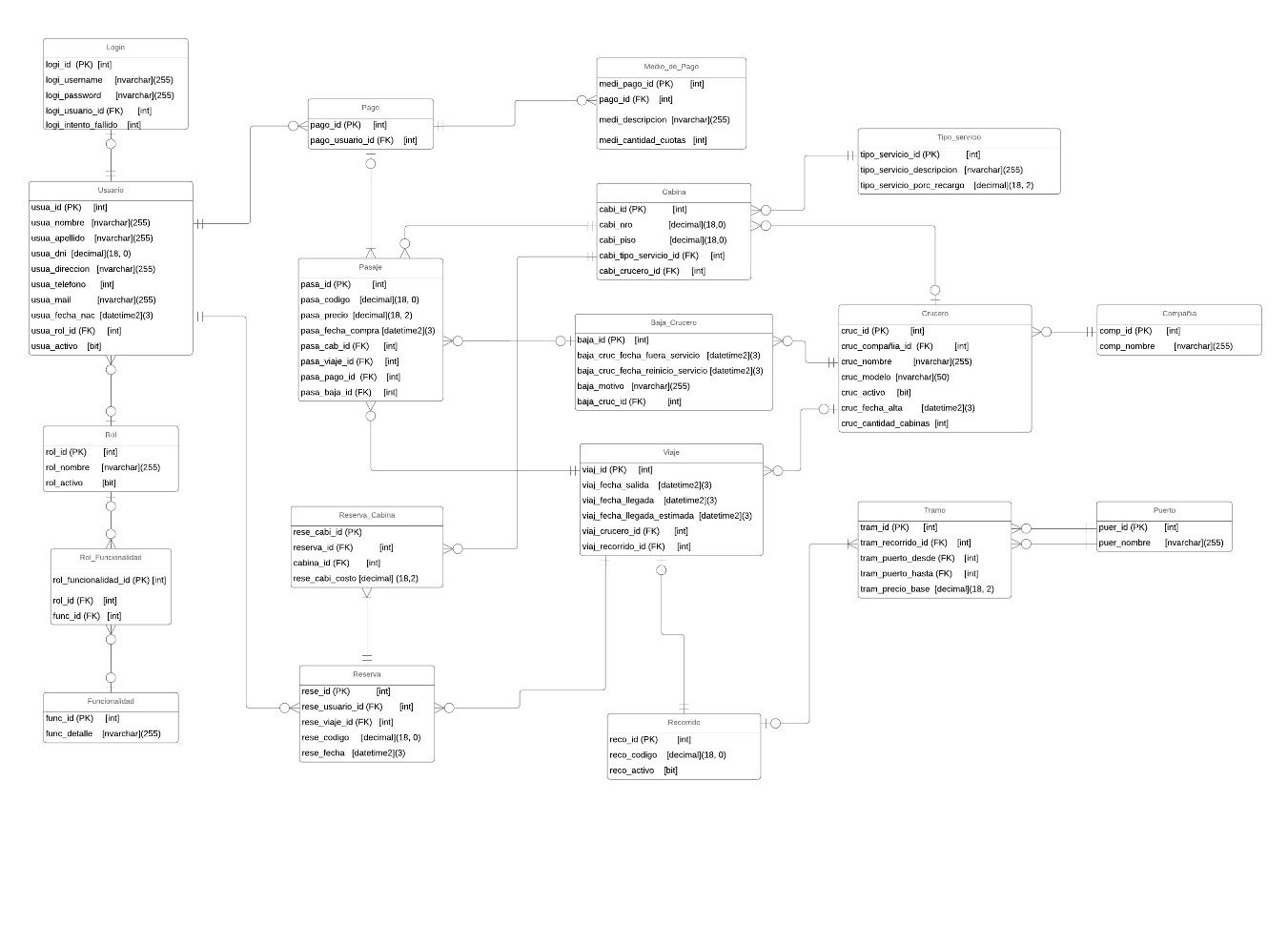
[Estrategia.pdf 1](#_Toc12659086)

[DER 2](#_Toc12659087)

[Desarrollo 2](#_Toc12659088)

[Decisiones de diseño 3](#_Toc12659089)

# DER



# Desarrollo

Al momento de empezar a diseñar el proyecto empezamos a analizar la tabla maestra para poder obtener los requerimientos solicitados de una manera clara y concisa, respetando el modelo de datos y las reglas del negocio existente. Para empezar a documentar esta información utilizamos un diagrama de entidad relación que nos ayudó a poder identificar cada una de las distintas entidades que utilizamos a lo largo del desarrollo del trabajo práctico, como así también la relación que existía entre cada una de ellas, haciendo principal foco en que el objetivo final es desarrollar un programa que permita administrar la venta de pasajes en viajes de crucero. Partiendo de esto, es que decidimos modelar a los usuarios con distintas funcionalidades de acuerdo con el rol que ocupen dentro del sistema, pudiendo ser administradores o clientes de este, con lo cual nos vimos obligados a contemplar la seguridad para los usuarios administradores, haciendo énfasis en evitar guardar contraseñas en texto plano. Aquí se decidió utilizar el mecanismo de tablas de hashing, por sus beneficios sobre encriptación de datos sensibles como lo son las contraseñas de los usuarios administradores. Otro punto que consideramos para reforzar la seguridad es implementar un contador de intentos fallidos que bloquea un usuario cuando éste ingresa 3 veces mal su contraseña, cumpliendo con los requerimientos establecidos en el trabajo práctico.

Posterior a eso, pasamos a identificar las entidades referidas a lo que implica hacer un viaje de crucero, donde las entidades principales que destacamos fueron *Crucero*, *Cabina*, *Viaje*, *Tramo*, *Pasaje* y *Reserva*. En este punto se decidió modelar una entidad aparte para la *Baja de Crucero*. Las entidades restantes se fueron completando en base a la modalidad y cardinalidad qué iban surgiendo en los casos de uso que se mencionaban en el enunciado del trabajo práctico, como fueron los *medios de pago*, los distintos *tipos de servicio* de las cabinas, las *compañías de crucero* y los distintos *puertos de origen y destino* por donde operan los mismos.

Completada la etapa anterior, se confeccionó el script inicial que permitió hacer una migración de los datos existentes manteniendo la integridad con el nuevo sistema. De forma conjunta se desarrollaron los *procedures* para las entidades de *cruceros*, *recorridos*, *roles* y *viajes* en forma modular, para luego unificar todo en un único script a ejecutarse.

Paralelamente se diseñaron las vistas del programa teniendo en cuenta los lineamientos pautados en el trabajo práctico, tanto para el desarrollo de los ABM, como para las clases auxiliares que utilizamos.

# Decisiones de diseño

*En el caso de que un Crucero fuera dado de BAJA ya sea por fuera de servicio o por vida útil, se decidió qué ante esta situación, todos los pasajes que estén involucrados en dicho crucero se cancelen de forma automática.*

*Al momento de dar de ALTA un Crucero, cuándo se eligen la cantidad de cabinas con sus respectivos servicios, sólo se puede agregar un tipo de servicio por piso, con la intención de simplificar la manipulación de los datos y su representación en las vistas.*

*Cuando se desea MODIFICAR un Crucero que ya está creado, primero se lo busca desde el Listado Cruceros. Después se selecciona con la Celda de Modificar, y luego se abre la ventana de Modificación Crucero con los datos actuales del mismo. En este punto no se habilita la opción para Agregar o Eliminar cabinas del crucero que estás modificando, sólo se considera el hecho de poder cambiar el Nombre, Modelo o Compañía del Crucero.*