# Laboratorio 2

Esteban Aguilera; Miguel Motta

# Parte 1: en Astah

# Parte 2: en Astah

# Parte 3: en Astah y documento

1. Modelo conceptual. Conceptos. (¿qué conoce?) (ASTAH)

Realicen el diagrama de conceptos4 extendido.

Únicamente extiendan los conceptos del ciclo.

(Consulten la especificación de los tipos de las columnas con el comando correspondiente al motor5).

No olviden indicar para cada uno de los atributos tipo y modificador, cardinalidad y requisito de unicidad, cuando sea necesario. Usen Integer, Real, Boolean, String y DateTime como tipos del modelo conceptual.

Recuerden los atributos a los que decidieron permitir que fueran desconocidos.

Diagrama de conceptos extendido: (ASTAH)

Diagrama

Descripción generada automáticamente

1. Modelo conceptual. Funciones. (¿qué hace?) (ASTAH)
   1. Realicen el diagrama de funciones.

¿Cuáles casos de uso son necesarios para almacenar la información del ciclo?

Usen el estándar Mantener <Objeto>, Mantener <Rol> y Registrar <Evento>

Usen para los casos de uso los colores de los CRUD.

Diagrama

Descripción generada automáticamente

En la primera operación, se establece que el gerente tiene la capacidad de acceder a la información de las habitaciones según su categoría, con el propósito de analizar qué categorías de habitaciones experimentan una mayor demanda. Este análisis permite optimizar la gestión de ingresos y aumentar la afluencia de clientes.

La operación de mantener tipos de habitación implica la capacidad de registrar, eliminar o consultar información relacionada con las diferentes categorías de habitaciones existentes. Para llevar a cabo estas acciones, se requieren permisos equivalentes a los otorgados al gerente, ya que la modificación de las categorías de habitaciones solo puede ser realizada por un miembro de alto rango dentro del equipo administrativo.

Por otro lado, la tarea asignada al recepcionista se centra en mantener las tarifas de habitación. Dado que el recepcionista es responsable de asignar habitaciones, es necesario que registre, elimine (en caso de cancelación) y consulte las tarifas correspondientes según sea necesario.

La operación de consultar habitación se refiere a la capacidad del recepcionista para proporcionar información específica sobre una habitación sin realizar cambios en su estado, dado que las características de cada habitación ya están establecidas.

1. Modelo conceptual. Consultas Operativas. (¿qué ofrece?)
   1. Implemente las siguientes nuevas consultas:
      1. El número y porcentaje de habitaciones de cada tipo de habitación

[id, descripción, número de habitaciones, porcentaje de habitaciones] (Orden de mayor a menor por número de habitaciones)

* + - 1. SELECT a.id, a.description, COUNT(b.id) as number, (COUNT(b.id)/(SELECT COUNT(id) FROM room))\*100 as percentaje FROM room\_type a

JOIN room b ON a.id=b.room\_type

GROUP BY a.id

ORDER BY number DESC

Tabla

Descripción generada automáticamente

* + 1. El precio de cada habitación considerando su máxima ocupación [id, descripcion del tipo de habitación, máxima ocupación, precio] (Ordenado de mayor a menor por precio)
       1. SELECT a.id, b.description, a.max\_occupancy, c.amount FROM room a

JOIN room\_type b ON b.id=a.room\_type

JOIN rate c ON b.id=c.room\_type

ORDER BY c.amount ASC Tabla

Descripción generada automáticamente

* + 1. Propongan una nueva consulta para esta área = El precio total de todas las habitaciones por tipo, sin importar los ocupantes (Mostrar tipo, descripción y precio)
       1. SELECT a.id, a.description, SUM(b.amount) FROM room\_type a

JOIN rate b ON a.id=b.room\_type

GROUP BY a.id

Tabla

Descripción generada automáticamente

* 1. De las consultas easy o medium propuestas en SQLZOO, seleccionen y generalicen la

que consideren más relevante para este ciclo de desarrollo. Implémentenla, si no lo han

hecho.



* + 1. SELECT (rate.amount + SUM(extra.amount)) AS amount

FROM booking JOIN rate ON booking.room\_type\_requested = rate.room\_type AND booking.occupants = rate.occupancy

JOIN extra ON booking.booking\_id = extra.booking\_id

WHERE booking.booking\_id = 5346;

Diagrama

Descripción generada automáticamente con confianza baja

* 1. De las consultas anteriores, diseñen las dos que consideren son las mejores consultas.

Preséntenlas en el diagrama de casos de uso. (ASTAH)

* Manager
  + Consultar número de habitaciones
  + Revisar precio de habitaciones

1. Modelo lógico. (¿cómo se almacena?) (ASTAH)
   1. Editen el modelo lógico general para que en este sólo queden las tablas necesarias para el ciclo: las propias y las de referencia.

# Parte Cuatro. Ciclo dos: en Astah y en documento

Diseñe e implemente este ciclo considerando todos los pasos propuestos para el ciclo

anterior A, B, C, D.

1. Modelo conceptual. Conceptos. (¿qué conoce?) (ASTAH)
   1. Realicen el diagrama de conceptos4 extendido.

Únicamente extiendan los conceptos del ciclo.

(Consulten la especificación de los tipos de las columnas con el comando correspondiente al motor5).

No olviden indicar para cada uno de los atributos tipo y modificador, cardinalidad y requisito de unicidad, cuando sea necesario. Usen Integer, Real, Boolean, String y DateTime como tipos del modelo conceptual.

Recuerden los atributos a los que decidieron permitir que fueran desconocidos.

* 1. Diagrama de conceptos extendido: (ASTAH)

Diagrama

Descripción generada automáticamente

1. Modelo conceptual. Funciones. (¿qué hace?) (ASTAH)
   1. Realicen el diagrama de funciones.

¿Cuáles casos de uso son necesarios para almacenar la información del ciclo?

Usen el estándar Mantener <Objeto>, Mantener <Rol> y Registrar <Evento>

Usen para los casos de uso los colores de los CRUD.

Diagrama

Descripción generada automáticamente

En este caso fue necesario hacer tres implementaciones de Mantener, primero para asignar, eliminar o consultar los extras generados para un cliente, pues es posible que desee añadir un cargo extra, o que se arrepienta y lo cancele, o que el recepcionista necesite conocer el total de los extras de un cliente.

El recepcionista también debe mantener la reservación, en caso de tener que registrarla para un cliente, que eliminarla en caso de que sea cancelado, o consultarla en caso de requerir la información (pues como recepcionista debe tener acceso a esta información).

El cliente debe poder mantener su reserva, puesto que debe poder registrarla, para guardar su cupo en el hotel y programar su estadía en la base de datos, poder eliminarla en el caso que se presente una cancelación, además de consultar la información de su reserva (fecha, habitación, costo).

El recepcionista debe poder registrar la llegada y salida de los clientes, no debe eliminarla, sino registrarla y consultarla de ser necesario.

El cliente debe poder registrar su encuesta y consultar la retroalimentación que dio, pero no debe eliminarla.

El cliente puede consultar su cuenta a pagar, pero no puede modificarla ni registrarla, pues se irá actualizando con los pagos extras y el costo de la reservación, pero no debe afectar esta información, para cambiar esta información se deben tener los permisos requeridos.

1. Modelo conceptual. Consultas Operativas. (¿qué ofrece?)
   1. Implemente las siguientes nuevas consultas:
      1. El número de reservas por dia de la semana [Dia de la semana, número de reservas] (Ordenado por número de reservas)
         1. SELECT DAYOFWEEK(booking\_date) AS dias, COUNT(\*) AS reservas

FROM booking

GROUP BY DAYOFWEEK(booking\_date)

ORDER BY días

Tabla

Descripción generada automáticamente

* + 1. Los mejores clientes del hotel (considerando el valor total de sus estadías (incluyendo extras). Los mejores clientes son aquellos superiores al promedio. [Identificación, nombres del cliente, valor habitación, valores extras, valor total] (Ordenado por valor total) Propongan una nueva consulta para esta área
       1. SELECT a.booking\_id,c.first\_name AS name, b.amount AS amount\_t, SUM(d.amount) AS extra\_t, b.amount + SUM(d.amount) AS total

FROM booking a

JOIN rate b ON a.room\_type\_requested = b.room\_type

JOIN guest c ON a.guest\_id = c.id

JOIN extra d ON a.booking\_id = d.booking\_id

GROUP BY a.booking\_id

HAVING b.amount + SUM(d.amount) > (

SELECT AVG(total\_c)

FROM (

SELECT SUM(a.amount) + SUM(c.amount) AS total\_c

FROM extra a

JOIN booking b ON b.booking\_id = a.booking\_id

JOIN rate c ON b.room\_type\_requested = c.room\_type

GROUP BY a.booking\_id) AS total\_aux)

ORDER BY total DESC

Tabla

Descripción generada automáticamente

* + 1. Propongan una nueva consulta para esta área = La suma total de los extras por habitaciones. Ordenar por habitaciones (Mostrar habitación,nombre, apellido, total extra)
       1. SELECT a.booking\_id, c.first\_name, c.last\_name, SUM(a.amount) AS total\_extra FROM extra a

JOIN booking b ON a.booking\_id=b.booking\_id

JOIN guest c ON b.guest\_id=c.id

GROUP BY a.booking\_id

ORDER BY a.booking\_id

Tabla

Descripción generada automáticamente

* 1. De las consultas easy o medium propuestas en SQLZOO, seleccionen y generalicen la

que consideren más relevante para este ciclo de desarrollo. Implémentenla, si no lo han

hecho.

* + 1. Indique la hora de llegada y los nombres y apellidos de todos los huéspedes que llegarán el 05/11/2016, ordene la salida por hora de llegada.

SELECT arrival\_time, first\_name, last\_name

FROM booking

INNER JOIN guest ON booking.guest\_id = guest.id

WHERE booking\_date = '2016-11-05'

ORDER BY booking.arrival\_time

Tabla

Descripción generada automáticamente

* 1. De las consultas anteriores, diseñen las dos que consideren son las mejores consultas.

Preséntenlas en el diagrama de casos de uso. (ASTAH)

* Supervisor:
  + Revisar mejores clientes
  + Consultar número de reservas

1. Modelo lógico. (¿cómo se almacena?)
   1. Editen el modelo lógico general para que en este sólo queden las tablas necesarias para el ciclo: las propias y las de referencia. (ASTAH)

# PARTE CINCO. Extendiendo un ciclo.

1. Implementen las consultas Hard questions en SQL [Escriban las consultas en los lenguajes pedidos y prueben la consulta SQL en sqlzoo. Si no lograron escribir alguna, Indiquen el punto de problema]
   * SELECT DISTINCT a.last\_name,a.first\_name,b.first\_name

FROM(

SELECT\*

FROM booking

JOIN guest ON (booking.guest\_id = guest.id))

AS a JOIN(

SELECT\*

FROM booking

JOIN guest ON (booking.guest\_id = guest.id)) AS b

ON (a.last\_name = b.last\_name)

AND a.booking\_date <= b.booking\_date

AND DATE\_ADD(a.booking\_date, INTERVAL (a.nights - 1) DAY) >= b.booking\_date

AND a.first\_name > b.first\_name

ORDER BY a.last\_name

Tabla

Descripción generada automáticamente

* + SELECT booking\_date,

SUM(CASE WHEN room\_no LIKE '1%' THEN 1 ELSE 0 END) AS 1st,

SUM(CASE WHEN room\_no LIKE '2%' THEN 1 ELSE 0 END) AS 2nd,

SUM(CASE WHEN room\_no LIKE '3%' THEN 1 ELSE 0 END) AS 3rd

FROM booking

WHERE booking\_Date BETWEEN '2016-11-14' AND '2016-11-20'

GROUP BY booking\_date;

Tabla

Descripción generada automáticamente

* + SELECT DISTINCT(room\_no)

FROM booking

WHERE room\_no NOT IN

(SELECT room\_no

FROM booking

WHERE '2016-11-25' BETWEEN booking\_date AND DATE\_ADD(booking\_date, INTERVAL nights-1 DAY))

Tabla

Descripción generada automáticamente

* + SELECT e.room\_no, e.earliest\_opening

FROM(

SELECT d.room\_no, MAX(DATE\_ADD(d.next\_booking, INTERVAL d.nights DAY)) AS earliest\_opening

FROM(

SELECT c.\*,

TIMESTAMPDIFF(DAY, c.booking\_date, c.next\_booking) - c.nights

FROM

(SELECT a.room\_no, a.nights, a.booking\_date, MIN(b.booking\_date) AS next\_booking

FROM booking AS a JOIN booking AS b ON a.booking\_date < b.booking\_date AND a.room\_no = b.room\_no

WHERE a.room\_no IN (101, 201, 301) AND a.booking\_date LIKE '2016-12%' AND b.booking\_date LIKE '2016-12%'

GROUP BY a.room\_no, a.nights, a.booking\_date

ORDER BY a.room\_no, a.booking\_date) AS c) AS d

GROUP BY d.room\_no) AS e

ORDER BY e.earliest\_opening

LIMIT 1

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

1. De las consultas hard seleccione la que considere más importante para adicionarla a uno de los cíclo de desarrollo. Diséñenla (no olvide generalizarlas). Preséntenla en un diagrama de casos de uso. (ASTAH)
   * Manager
     1. Verificar habitaciones libres

# RETROSPECTIVA

1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno? (Horas/Hombre)
   1. 8/ Esteban Aguilera
   2. /Miguel Motta
2. ¿Cuál es el estado actual del laboratorio? ¿Por qué?
   1. Gracias a nuestro trabajo en equipo, hemos logrado resolver todo el laboratorio de manera completa y satisfactoria.
3. ¿Cuál consideran fue el mayor logro? ¿Por qué?
   1. Abordar la sección de los modelos conceptuales ya que todavía no estamos acostumbrados y familiarizados con ello.
4. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?
   1. Pegar los códigos de SQL y las capturas de pantalla en el archivo docs, ya que a veces el texto se alinea el texto erróneamente y es algo tedioso acomodarlo.
5. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?
   1. Compartir conocimientos y discutir nos permite encontrar soluciones más rápidamente. Al colaborar y aprovechar la diversidad de perspectivas, podemos llegar a respuestas eficientes en poco tiempo
   2. Nos comprometemos a seguir trabajando en equipo como lo estamos haciendo actualmente, abordando los problemas de manera satisfactoria y resolviéndolos de forma efectiva