Data Management & Sql

Advertising Management Model pROPOSAL

Natalia Gomez Aycardi, Tommaso Mazzuco

Hult International Business School

**- PART 1 – Executive overview of the system:**

Outsmart labs it's a digital marketing agency that is growing at a fast-paced by acquiring various international clients. They currently manage all the digital marketing initiatives of multiple clients trying to grow their brands' digital presence.

  As many other advertising agencies, Outsmart Labs struggles to centralize all their different campaigns and get unique consumer insights because of the increase of multiple devices and channels. These omnichannel experiences are making for agencies increasingly complex to gain unique insights about the clients leads, their consumer journey, and to understand what campaigns are driving results and which ones are not, and how to optimize accordingly. Therefore, they need an advertising management database capable of providing them a holistic unified view of all the leads and campaign results per account and monitoring their interactions with the campaigns across all channels.

Furthermore, Outsmart Labs' organizational data is highly fragmented at several levels, leading to siloed data and duplicate account records in different databases. This creates a data quality challenge for Outsmart Labs, like storing monthly invoices to their respective account and registering the expenditures per account per month.

Based on the above, the purpose of the following system would be to create an efficient advertising management platform capable of integrating all available data about each account and respective campaign into one view, so Outsmart Labs could deliver a compelling customer experience across all channels. By solving all the data daze and fragmentation, the advertising management model will centralize the processes and technology required to run cross-channel campaigns.

**- PART 2 – FLOW OF THE SYSTEM -**

This advertising management database system will fundamentally focus on campaign data. For each campaign, the idea is to store all related accounts and leads, what product was being advertised, what employees were in charge, the overall performance of the campaign, and all the campaign results store in one platform. We'll also manage invoices between the agency and its clients.

The idea is that the model is capable of storing details related to the services we provide to clients. We can expect that the client will ask for an all-in-one solution for his needs. For example, the client could ask us to run Google Ads and at the same time manage paid social campaigns.

1. **ACCOUNTS**

The first section of the model will revolve around their account management platform. Thus, the flow of the system starts with the accounts table. Each account has a unique identifier, so it is easier to track all the movement across the client's database with that unique ID. This account table is created based on the need of storing your client's data and contact details in one platform.

1. **INVOICES**

The invoices table will allow you to manage each monthly account bill more seamlessly and store it in your system. This table also allows you to send any details about that specifically invoice to the clients and is connected to each client's unique identifier.

1. **PAYMENTS**

The idea behind the payments table is to allow the company to monitor each transaction, by using each invoice Identifier. Invoice ID and Payments ID show as the relation the account table has to these ID's.

1. **AGENCY EMPLOYEES**

The agency has few employees at the moment, but having them in a system is essential, and more so if you will keep growing at the pace you have been in the last few years. It is also important to know which employee was doing what for what campaign and advertising effort. To this category, the first one is Employees, with all their identifying details and their respective Job Title.

1. **PRODUCTS**

The model continues with our Products table. We'll advertise a specific product (or service) or multiple products for each account for each campaign. A link to each product will be provided, in order to store the product's location at the moment it is available. This attribute is specific to the product table and is not present in the campaigns table.

1. **CAMPAIGNS**

The entire system revolves around the agency campaigns. Without these, none of the other tables could exist. For each campaign, the idea is to store all related accounts and leads, what product was being advertised, what employees were in charge, the overall performance of the campaign, and all the campaign results store in one platform. In the campaign table, we'll manage the list of all campaigns our agency has done for a client and its respective channel type. Each campaign is strictly related to only one client and product. But we could have multiple campaigns for the same client at the same time or consecutively. Hence, we must be able to build a solid and concise data infrastructure that can facilitate us to see the results of each present and past campaign.

1. **CAMPIGN\_RESULTS**

We need to get a holistic picture of how each campaign is performing on a monthly and Quartey basis and get a unified view of the results per account. This is going to be very beneficial for the agency when preparing the monthly performance reports for each account. We decided to take four critical conversions metrics that will help us identify if a campaign performs or didn't perform:

- Total\_reach

- Total\_engagement

- Total\_impressions

- Total\_conversions

- ROI

- Clicks

- CTR

1. **LEADS**

We need to capture a bigger scope of data related to each lead per account. The way to promote the clients' products will be directly related to the information you gather from them. We understood this need and based on it and on the logic of the business, so the leads table accompanies campaigns. All following tables are directly or indirectly related to their information. Therefore the amount of data collected from them will be crucial. The relation between this table could potentially allow Outsmart Labs to understand the average age of the customers per account, their gender, when they buy, why, or the frequency of their purchases. We included the channel type and the campaign\_id to actively track which channel is bringing the most customers and potentially strengthen your efforts to that specific channel. With this amount of information, the possibility of focusing our advertising efforts on a client-centric approach is higher (e.g., email marketing).

**- PART 3 – DATABASE STRUCTURE -**

**Account**

**Products**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Account\_ID | INT | NO | PRIMARY | Unique Identifier of the Account |
| Account\_name | VARCHAR (45) | NO |  | Name of the client |
| Account\_address | VARCHAR (45) | NO |  | Address of the client |
| Account email | VARCHAR (45) | NO |  | Email of client |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Product\_ID | INT | NO | PRIMARY | Unique Identifier for products |
| Campaign\_ID | VARCHAR (45) | NO |  | Unique Identifier of campaign |
| Product\_name | VARCHAR (45) | NO |  | Name of product |
| Product\_link | VARCHAR (45) | NO |  | Link that redirects to product |
| Account\_ID | VARCHAR (45) | NO |  | Unique Identifier of the Account |

**Agency Employees**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Employee\_ID | INT | NO | PRIMARY | Unique Identifier Agency Employees |
| Employee\_name | VARCHAR (45) | NO |  | Name of Employee of agency |
| Employee\_last\_name | VARCHAR (45) | NO |  | Last name of Employee |
| JobTitle | VARCHAR (45) | NO |  | Role in the Agency |

**Invoices**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Invoice\_ID | INT | NO | PRIMARY | Unique Identifier for invoices |
| Invoice\_amount | INT | NO |  | Amount $ corresponding to invoice |
| Other\_details | LONGTEXT | YES |  | Details relatively to clients |
| Account\_ID | INT | NO |  | Unique identifier for Accounts |
| Invoice\_issued | DATE |  |  | Date when the invoice is schedule |
| STATUS | VARCHAR (45) |  |  | The status of the invoice to see if it was paid or unpaid |

**Campaigns**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Campaign\_ID | INT | NO | Primary | Unique Identifier Campaign |
| Campaign\_name | VARCHAR (45) | NO |  | Name of Campaign |
| Status | TINYTEXT | NO |  | Status of campaign |
| Start\_date | DATE | NO |  | Start date of Campaign |
| End\_date | DATE | NO |  | End date of campaign |
| Description | LONGTEXT | YES |  | Description of Campaign |
| Channel\_type | VARCHAR (45) | NO |  | Type of channel |
| Budget | INT | NO |  | Amount of Budget |
| Product\_ID | INT | NO |  | Unique Identifier Product |
| Employee\_ID | INT | NO |  | Unique Identifier Employee |
| Objective | VARCHAR (45) | NO |  | The campaign objective |
| Creative\_type | VARCHAR (45) | NO |  | The format of the campaign(e.g.video, display) |

**Payments**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Payment\_ID | INT | NO | PRIMARY | Unique Identifier Campaign |
| Payment\_Amount | INT | NO |  | Amount of Payment |
| Other\_details | INT |  |  | Details of Payments |
| Invoice\_ID | INT | NO |  | Unique Identifier for Invoices |

**Leads**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Leads\_ID | INT | NO | PRIMARY | Unique Identifier Leads |
| User\_name | VARCHAR (45) | NO |  | User name |
| Email | VARCHAR (45) | NO |  | Email of User |
| Age | VARCHAR (45) | NO |  | User’s age |
| Gender | VARCHAR (45) | NO |  | User’s gender |
| Country | VARCHAR (45) | NO |  | Country where user Is from |
| Device | VARCHAR (45) | NO |  | User’s device used |
| Campaign\_ID | INT | NO |  | Unique Identifier of Campaign |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | **DATATYPE** | **NULL** | **KEY** | **DESCRIPTION** |
| Total\_reach | INT | NO |  | # people who have seen campaign |
| Total\_Engagement | INT | NO |  | # reactions and interactions to campaings |
| Purchases | INT | NO |  | # of purchases |
| Campaign\_ID | INT | NO |  | Unique Identifier of Campaign |
| Total\_Impressions | int | no |  | Total # of exposures to the campaign |
| ROI | Decimal (3,2) | NO |  | Return on Investment per each campaign |
| CTR | Decimal(3,2) | NO |  | # of clicks received on the campaigns per # of impressions |
| Clicks | INT | NO |  | # of people who click on the campaign |

**Campaign\_results**

**- PART 4 – ENTITY-RELATIONAL MODEL –**

****

**- PART 5 – SQL QUERIES TO BUILD THE REPORT –**

1. **This example shows how to determine which campaigns are reaching a given demographic. This would be crucial in order to optimize campaigns accordingly to the audiences.**

|  |
| --- |
| SELECT gender AS "User Gender", country AS "User Country", age AS "Customer Age", campaign\_id,  COUNT(user\_id) AS "Number of Customers"  FROM leads  GROUP BY country, gender, Age, campaign\_id  ORDER BY gender, country ASC; |

Table

Description automatically generated

|  |
| --- |
| select a.account\_id, b.invoice\_id, a.account\_name, b.invoice\_amount,b.invoice\_issued, b.STATUS  from accounts a, invoices b  where b.invoice\_issued = (select max(b1.invoice\_issued)  from invoices b1  where b.invoice\_id = b1.invoice\_id)  and a.account\_id = b.Account\_id; |

1. **Show the most recent Invoice per account**

Table

Description automatically generated

1. **Get a holistic unified view of how the most recent campaigns are performing**

|  |
| --- |
| SELECT distinct a.campaign\_id, b.total\_reach, b.total\_engagement, b.total\_impressions, b.purchases, b.CTR ,c.start\_date,c.end\_date  from leads a, Campaign\_results b, campaigns c  where c.end\_date = (select max(c1.end\_date)  from campaigns c1  where c1.campaign\_id = c.campaign\_id)    and a.campaign\_id = b.campaign\_id  and b.campaign\_id = c.campaign\_id  order by b.CTR desc; |

Table

Description automatically generated

1. **Only show the campaigns that are active per account and who is the campaign lead**

|  |
| --- |
| SELECT c.status, c.campaign\_id, a.account\_id, a.account\_name, c.employee\_id, concat\_ws("," ,d.employee\_name, d.employee\_last\_name) AS "Employee Name"  from campaigns c, products p, accounts a, agency\_employees d  where status = "Active"  And c.product\_id = p.product\_id  And p.account\_id = a.account\_id  and c.employee\_id = d.employee\_id; |

Table

Description automatically generated

1. **This example shows how to analyze audiences. Learn which audiences impressions are reaching, and determine if some audiences perform better than others. This knowledge can help balance unique cookie count (putting ads in front of a lot of users) and quality (narrow targeting and viewable impressions), depending on the campaign objectives.**

|  |
| --- |
| SELECT a.campaign\_id, a.device,b.total\_impressions,b.total\_reach,b.clicks,  COUNT(DISTINCT a.user\_id) AS uniques,  ROUND(COUNT(\*) / COUNT(DISTINCT a.user\_id), 1) AS frequency  FROM leads a, campaign\_results b  where a.campaign\_id =b.campaign\_id  GROUP BY a.campaign\_id, b.clicks,b.total\_impressions,b.total\_reach, a.device; |

Table

Description automatically generated

1. **Which products have the most purchases per campaign?**

|  |
| --- |
| SELECT b.product\_id AS "Product ID", b.product\_name AS "Product Name", SUM(a.purchases) AS "Number of Products Sold Per Campign", d.account\_name AS "Account"  FROM campaign\_results AS a , products AS b, campaigns As c, accounts AS d  WHERE a.campaign\_id= c.campaign\_id  AND b.product\_id = c.product\_id  AND b.account\_id = d.account\_id  GROUP BY b.product\_id  ORDER BY SUM(a.purchases) DESC; |

Table

Description automatically generated

1. **In which channels are our campaign performing the best?**

|  |
| --- |
| SELECT ca.channel\_type AS "Channel", SUM(c.purchases) as "Number of Purchases per Channel", c.Total\_impressions, c.CTR, c.Total\_reach  FROM Campaigns AS ca, Campaign\_results AS c  WHERE ca.campaign\_id = c.campaign\_id  GROUP BY ca.channel\_type, c.Total\_impressions, c.CTR, c.Total\_reach  ORDER BY COUNT(c.purchases) DESC; |

Table

Description automatically generated

1. **Does a higher investing in campaigns means a higher ROI?**

|  |
| --- |
| SELECT a.creative\_type, a.budget as 'Campaign Costs', a.campaign\_name, b.total\_impressions, b.ROI, b.Clicks, b.purchases  from campaigns a, campaign\_results b  where a.budget = (select max(a1.budget) -- Don't forget the ()  from campaigns a1  where a.campaign\_id=a1.campaign\_id)  and a.campaign\_id = b.campaign\_id  ORDER BY total\_impressions DESC; |

Table

Description automatically generated

**-**

* **PART 6 – DETAIL SQL PROCEDURES–**

The first built procedure will focus on that automatically updating the status of each campaign when they are due.

This procedure is meant to automatimize the process of storing the information regarding the status of a specific campaign without the help of a manual entry.

For the fact that some campaigns are about or still didn’t start, the procedure is based on current date.

**Input:** end date of campaign as DATE

**Functionality:**

**Step 1:** declare a variable, v\_status and v\_end\_date, to store the following query. Declared respectively as TINYTEXT and DATE

**Step 2:** selecting the previously declared variables and store them as the corresponding variables existing in the dataset

**Step 3:** If statement as:

IF v\_end\_date is equal to the current date then the status of the campaign is UPDATED and SET as SHUTDOWN.

Output: “SHUTDOWN” TINYTEXT

Please find below the code of the procedure we just described

|  |
| --- |
| CREATE DEFINER=`root`@`localhost` PROCEDURE `updatestatus`(IN in\_end\_date DATE ,OUT out\_status\_upd tinytext)  BEGIN  DECLARE v\_status TINYTEXT;  DECLARE v\_end\_date DATE;  select status, end\_date  into v\_status, v\_end\_date  from campaigns  where v\_end\_date = in\_end\_date;  if v\_end\_date = curdate() THEN  UPDATE campaigns  set v\_status = "SHUTDOWN"  where v\_status = ut\_status\_upd;  end if;  END |

As we mentioned in the first section, an effective omnichannel strategy is one of the most important attributes to measure and optimize campaigns. Therefore, there is an opportunity for Outsmart Labs to update the respective budget for a channel type that is proving more results on some campaigns that In other one.

This procedure will Identify all the campaigns id that have that channel type which is given as an input – look cur1 query. It select all the campaigns\_id from campaigns that have the channel\_type that was given as an input. Then the procedure updates the campaign objective to “conversions” for all those ‘Facebook’ campaign types (but it can be change to the channel the company wants to optimize). Lastly, in order to maximize results, the procedures updates the campaign budget by increasing it by 25% for the identified channel types.

**Input**: channel\_type as varchar(45)

**Functionality**

**Step 1** – Declare the variables needed to do the procedure and to store the following query. Declared v\_campaign\_id as varchar(10) and DECLARE C as varchar(10);

**Step 2** – Select campaign\_id of all campaigns of that channel type that was input

**Step 3** – Declare cursor in order to handle the results set

**Step 4** – Declare a NOT FOUND handler to handle the situation when the cursor could not find any row.

**Step 5** – Open the cursor and open the loop

**Step 6** – Inside the loop we used the finished variable

**Step 7** – Begin the loop and fetch the statement with campaign\_id

**Step 8-** Do exception handling. If there are no more records left, then exit the loop

**Step 9 -** Do the queries to update the objectives and the budget for the campaigns that have a channel type of “facebook”. If the channel type is facebook, to update the objective for conversions and increase for 25% the budget.

**Step 10** – End the if statements and close loop

|  |
| --- |
| CREATE DEFINER=`root`@`localhost` PROCEDURE `check\_Budget`(in in\_channel\_type varchar(45))  BEGIN  /\*Declare variables\*/  DECLARE done INT DEFAULT FALSE;  DECLARE v\_campaign\_id varchar(10);  DECLARE C varchar(10);    /\*Select campaign\_id of all campaigns the channel type \*/    DECLARE cur1 CURSOR FOR  select campaign\_id  from campaigns  Where channel\_type = in\_channel\_type;    /\* Exception handling for the cursor\*/    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;  OPEN cur1;    /\*beginning of the loop\*/    read\_loop: LOOP  FETCH cur1 into v\_campaign\_id;    /\*Fetch the next record into variable v\_campaign\_id\*/  /\*exception handling. If there are no more records left, then exit the loop. \*/    IF done THEN  LEAVE read\_loop;  END IF;    /\* Now I can do processing based on the value in v\_campaign\_id.\*/  IF c= 'facebook' then  update campaigns  set objective = "conversions"  where campaign\_id = v\_campaign\_id  and in\_channel\_type=c;  END IF;  IF c= 'facebook' then  update campaigns  set budget= 1.25\*budget  WHERE in\_channel\_type=c  AND campaign\_id = v\_campaign\_id;  end if;  /\*end of the loop\*/  END LOOP;  END |

**Appendix**

1. **ER Model Script**

-- -----------------------------------------------------

-- Schema team\_project

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `team\_project` DEFAULT CHARACTER SET utf8 ;

USE `team\_project` ;

-- -----------------------------------------------------

-- Table `team\_project`.`Campaigns`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Campaigns` (

`campaign\_id` INT NOT NULL,

`campaign\_name` VARCHAR(45) NULL DEFAULT NULL,

`status` TINYTEXT NULL DEFAULT NULL,

`start\_date` DATE NULL DEFAULT NULL,

`end\_date` DATE NULL DEFAULT NULL,

`Description` LONGTEXT NULL DEFAULT NULL,

`channel\_type` VARCHAR(45) NOT NULL,

`budget` INT NULL DEFAULT NULL,

`product\_id` INT NULL DEFAULT NULL,

`employee\_id` INT NOT NULL,

`Objective` VARCHAR(45) NULL DEFAULT NULL,

`creative\_type` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`campaign\_id`, `channel\_type`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Products`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Products` (

`Product\_id` INT NOT NULL,

`campaign\_id` VARCHAR(45) NULL DEFAULT NULL,

`product\_name` VARCHAR(45) NULL DEFAULT NULL,

`product\_link` VARCHAR(45) NULL DEFAULT NULL,

`Account\_id` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`Product\_id`),

CONSTRAINT `fk\_Products\_Campaigns1`

FOREIGN KEY (`Product\_id`)

REFERENCES `team\_project`.`Campaigns` (`product\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Invoices`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Invoices` (

`invoice\_id` INT NOT NULL,

`invoice\_amount` INT NOT NULL,

`other\_details` LONGTEXT NOT NULL,

`account\_id` INT NOT NULL,

`invoice\_issued` DATE NULL DEFAULT NULL,

`STATUS` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`invoice\_id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Accounts`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Accounts` (

`account\_id` INT NOT NULL,

`account\_name` VARCHAR(45) NULL DEFAULT NULL,

`account\_address` VARCHAR(45) NULL DEFAULT NULL,

`account\_email` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`account\_id`),

CONSTRAINT `fk\_Accounts\_Products`

FOREIGN KEY (`account\_id`)

REFERENCES `team\_project`.`Products` (`Account\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Accounts\_Invoices1`

FOREIGN KEY (`account\_id`)

REFERENCES `team\_project`.`Invoices` (`account\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Agency\_Employees`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Agency\_Employees` (

`employee\_id` INT NOT NULL,

`employee\_name` VARCHAR(45) NULL DEFAULT NULL,

`employee\_last\_name` VARCHAR(45) NULL DEFAULT NULL,

`JobTitle` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`employee\_id`),

CONSTRAINT `fk\_Agency\_Employees\_Campaigns1`

FOREIGN KEY (`employee\_id`)

REFERENCES `team\_project`.`Campaigns` (`employee\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Campaign\_results`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Campaign\_results` (

`campaign\_id` INT NOT NULL,

`Total\_reach` INT NOT NULL,

`Total\_engagement` INT NOT NULL,

`Total\_impressions` INT NOT NULL,

`Purchases` INT NOT NULL,

`ROI` DECIMAL(3,2) NOT NULL,

`CTR` DECIMAL(3,2) NULL DEFAULT NULL,

`Clicks` INT NULL DEFAULT NULL,

PRIMARY KEY (`campaign\_id`),

CONSTRAINT `fk\_Campaign\_results\_Campaigns1`

FOREIGN KEY (`campaign\_id`)

REFERENCES `team\_project`.`Campaigns` (`campaign\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Leads`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Leads` (

`user\_id` INT NOT NULL,

`user\_name` VARCHAR(45) NOT NULL,

`email` VARCHAR(45) NOT NULL,

`Age` VARCHAR(45) NOT NULL,

`Gender` VARCHAR(45) NOT NULL,

`Country` VARCHAR(45) NOT NULL,

`device` VARCHAR(45) NOT NULL,

`campaign\_id` INT NOT NULL,

PRIMARY KEY (`user\_id`),

INDEX `fk\_Leads\_Campaigns1\_idx` (`campaign\_id` ASC) VISIBLE,

CONSTRAINT `fk\_Leads\_Campaigns1`

FOREIGN KEY (`campaign\_id`)

REFERENCES `team\_project`.`Campaigns` (`campaign\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `team\_project`.`Payments`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `team\_project`.`Payments` (

`Payment\_id` INT NOT NULL,

`Payment\_Amount` INT NULL DEFAULT NULL,

`other\_details` LONGTEXT NULL DEFAULT NULL,

`invoice\_id` INT NOT NULL,

PRIMARY KEY (`Payment\_id`),

INDEX `fk\_Payments\_Invoices1\_idx` (`invoice\_id` ASC) VISIBLE,

CONSTRAINT `fk\_Payments\_Invoices1`

FOREIGN KEY (`invoice\_id`)

REFERENCES `team\_project`.`Invoices` (`invoice\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

1. Tables Data

Note: The data was imported from an csv file into the SQL Import Wizard

**Campaigns Table**

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (1,'BoingGames','ACTIVE','2021-03-01','2021-04-04','boing\_game','facebook',12208,34,1998,'conversions','video');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (2,'toys\_dolls\_Q1','SHUTDOWN','2021-03-01','2021-04-04','toysmart\_campagin','YouTube',300,20,2535,'traffic','display');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (3,'SVT\_fall\_collection','ACTIVE','2021-01-01','2021-05-08','svt\_campaing','facebook',15260,15,1990,'conversions','display');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (4,'facials\_Q1','ACTIVE','2020-07-09','2021-08-09','facials\_campaign','google',25000,50,6321,'conversions','video');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (5,'MOXY\_rooms','SHUTDOWN','2020-09-10','2021-08-10','moxy\_campaign','google',12500,30,4389,'conversions','native');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (6,'rooms\_ocean\_views','ACTIVE','2021-02-02','2022-09-09','rooms\_campaign','linkedin',20000,71,1111,'conversions','mobile');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (7,'flights\_q1','ACTIVE','2021-01-02','2021-05-04','gflights\_campaign','instagram',6000,66,1693,'engagement','video');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (8,'margaritas\_promotions\_q1','ACTIVE','2021-09-09','2021-06-01','margaritas\_campaign','instagram',500,10,1990,'lead\_generation','display');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (9,'tacos\_promotions\_q1','ACTIVE','2021-09-09','2021-04-12','tacos\_campaign','facebook',48829,33,3476,'conversions','mobile');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (10,'discount\_paintings\_q1','SHUTDOWN','2021-03-01','2021-04-14','paintings\_campaigns','google',1250,27,3476,'conversions','video');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (11,'vacation\_promotion','ACTIVE','2021-03-10','2021-04-10','moxy\_promotions','google',2500,30,1998,'conversions','display');

INSERT INTO `` (`campaign\_id`,`campaign\_name`,`status`,`start\_date`,`end\_date`,`Description`,`channel\_type`,`budget`,`product\_id`,`employee\_id`,`Objective`,`creative\_type`) VALUES (12,'restaurant\_promotion\_Q1','ACTIVE','2021-03-12','2020-04-12','Coyo\_Taco\_digital\_influencer\_campagin','TikTok',8000,33,3476,'store\_visits','display');

**ACCOUNTS TABLE**

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1009,'BoingToys','1st Education Street','BoingToys@gmail.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1220,'ToySmart','4n Canal Park','ToySmart@gmail.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1340,'SilviaTcherassi','1st Street','SVT@hotmail.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1360,'CoyoTaco','111 SW 1st Avenue','Coyotaco@restaurants.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1450,'DesignDistrict','1657 N Boston Avenue','District@designeres.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1580,'MoxyHotel','17315 Collins Ave ','Moxyhotel@gmail.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (1890,'TravelAir','19th Canal Park','Travelair@gmail.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (2020,'HiltlonCaribbean','31st Bonus Street','Hilton@student.hult.edu');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (5060,'NeuroSpa','1st Dalton Street','NeuroSpa@aol.com');

INSERT INTO `` (`account\_id`,`account\_name`,`account\_address`,`account\_email`) VALUES (6040,'Serena','240 Tremont St','Serena@moxyhotels.com');

**Agency Employees**

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (1111,'Marc','Friedman','Director of paid search',1);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (1489,'Gia','Lu','Marketing Coordinator',4);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (1693,'Mariana','Gutierrez','Art Director',3);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (1990,'Mariana','Ballesteros','Social media specialist',6);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (1998,'Tommaso','Mazzucco','CEO',1);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (2535,'Natalia','Gomez','Analyst',3);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (3476,'Alexandra','Rodriguez','Social media manager',9);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (3579,'David','Azar','SEO coordinator',7);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (3924,'Christian','Romero','Campaign Manager',5);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (4389,'Lidia','Calsamiglia','Account Manager',5);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (4664,'Cristina','Angel','Influencer Marketing Manager',2);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (6190,'Rachael','Fredman','Marketing Specialist',2);

INSERT INTO `` (`employee\_id`,`employee\_name`,`employee\_last\_name`,`JobTitle`,`campaign\_id`) VALUES (6321,'martha','Aycardi','Director of paid search',10);

**Invoices**

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (1078,25000,'advertising\_march',1009,'2021-12-01','OVERDUE');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (1368,5000,'advertising\_march',1450,'2021-03-01','UNPAID');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (1645,15000,'advertising\_march',1360,'2021-03-03','PAID');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (2078,60000,'advertising\_march',1580,'2021-03-01','PAID');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (2348,12900,'advertising\_march',6040,'2021-03-01','OVERDUE');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (2575,10000,'advertising\_march',1890,'2021-03-01','PAID');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (2681,8000,'advertising\_march',2020,'2021-03-01','OVERDUE');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (4366,30000,'advertising\_march',5060,'2021-03-01','UNPAID');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (4368,5000,'advertising\_march',1220,'2021-03-01','OVERDUE');

INSERT INTO `` (`invoice\_id`,`invoice\_amount`,`other\_details`,`account\_id`,`invoice\_issued`,`STATUS`) VALUES (9076,1000,'advertising\_march',1340,'2021-03-01','PAID');

**LEADS**

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (1330,'Natalia Hernandez','natalia@gmail.com','31','F','Brazil','mobile',6);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (5672,'Julien Casablancas','julien@music.com','18','M','USA','desktop',4);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (5790,'Carly berry','carly@northeastern.edu','22','F','China','mobile',5);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (6566,'Natalia Castano','casta@hotmail.com','20','F','Peru','mobile',7);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (7658,' Natalia Gomez','natalia@hult.edu','25','F','Colombia','mobile',1);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (8091,'Tommas Mazzucco ','tommy@gmail.com','24','M','Italy','desktop',2);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (8309,'Marc Gomez','marc.132@aol.com','39','M','Germany','desktop',8);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (8589,'Cecicilia Brazil','cecilia@student.hult.edu','17','F','USA','desktop',10);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (9194,'Tuty Rose','tuty@hotmail.com','25','F','Spain','desktop',9);

INSERT INTO `` (`user\_id`,`user\_name`,`email`,`Age`,`Gender`,`Country`,`device`,`campaign\_id`) VALUES (16660,'Ashley chica','ashley@harvard.edu','33','F','USA','mobile',3);

**PAYMENTS**

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (1925,8000,'advertising\_march',1078);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (1987,30000,'advertising\_march',1368);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (2074,15000,'advertising\_march',1645);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (2813,25000,'advertising\_march',2078);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (4305,5000,'advertising\_march',2348);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (4456,5000,'advertising\_march',2575);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (5238,10000,'advertising\_march',2681);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (7642,1000,'advertising\_march',4366);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (7820,12900,'advertising\_march',4368);

INSERT INTO `` (`Payment\_id`,`Payment\_Amount`,`other\_details`,`invoice\_id`) VALUES (8650,50000,'advertising\_march',9076);

**PRODUCTS**

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (10,'8','Margaritas','https.margaritas.com','6040');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (15,'3','Shirt\_fall\_collection','https.shirts-and.coll.com','1340');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (20,'2','Dolls','https.dolls.com','1220');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (27,'10','Paintings','https.paint.com','1450');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (30,'5','Rooms','https.roomscom','1580');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (33,'9','Tacos','https.tacos.amo.','1360');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (34,'1','Games','https.games.com','1009');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (50,'4','Facials','https.facials.com','5060');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (66,'7','Flights','https.flights.com','1890');

INSERT INTO `` (`Product\_id`,`campaign\_id`,`product\_name`,`product\_link`,`Account\_id`) VALUES (71,'6','Rooms\_ocean\_view','https.rooms\_ocean\_view.com','2020');

**CAMPAIGNS RESULTS**

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (1,1390,1000,2000,50,0.01,0.40,800);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (2,8000,2400,10000,185,1.20,0.06,650);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (3,500,160,800,100,6.00,0.12,100);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (4,20000,5000,50000,600,8.15,0.34,17000);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (5,300,130,600,2,3.01,0.08,50);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (6,75000,46000,700,9,5.62,8.57,60000);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (7,1000,67,1500,1,0.05,0.03,37);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (8,164000,145000,180000,1000,9.95,0.83,150000);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (9,1825,1200,1927,7,4.03,0.46,900);

INSERT INTO `` (`campaign\_id`,`Total\_reach`,`Total\_engagement`,`Total\_impressions`,`Purchases`,`ROI`,`CTR`,`Clicks`) VALUES (10,612,311,642,3,1.50,0.66,430);