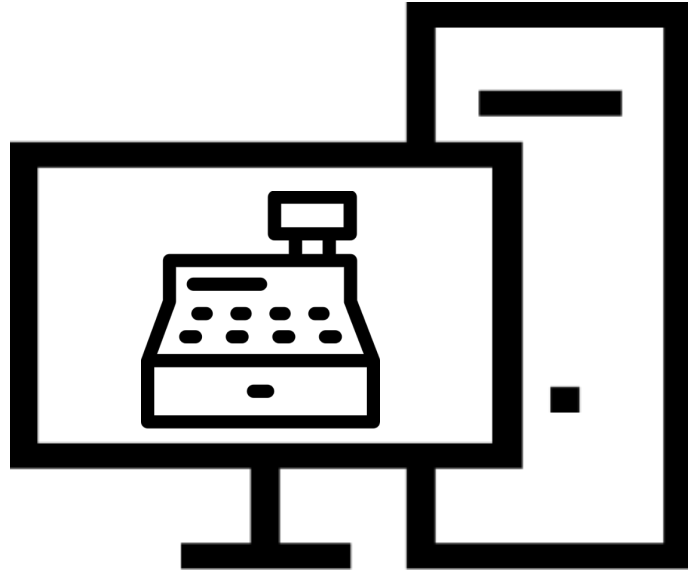




MotoShop

Fábio Alves - 84794
Marcelo Fraga - 84839
Bases de Dados - 2019

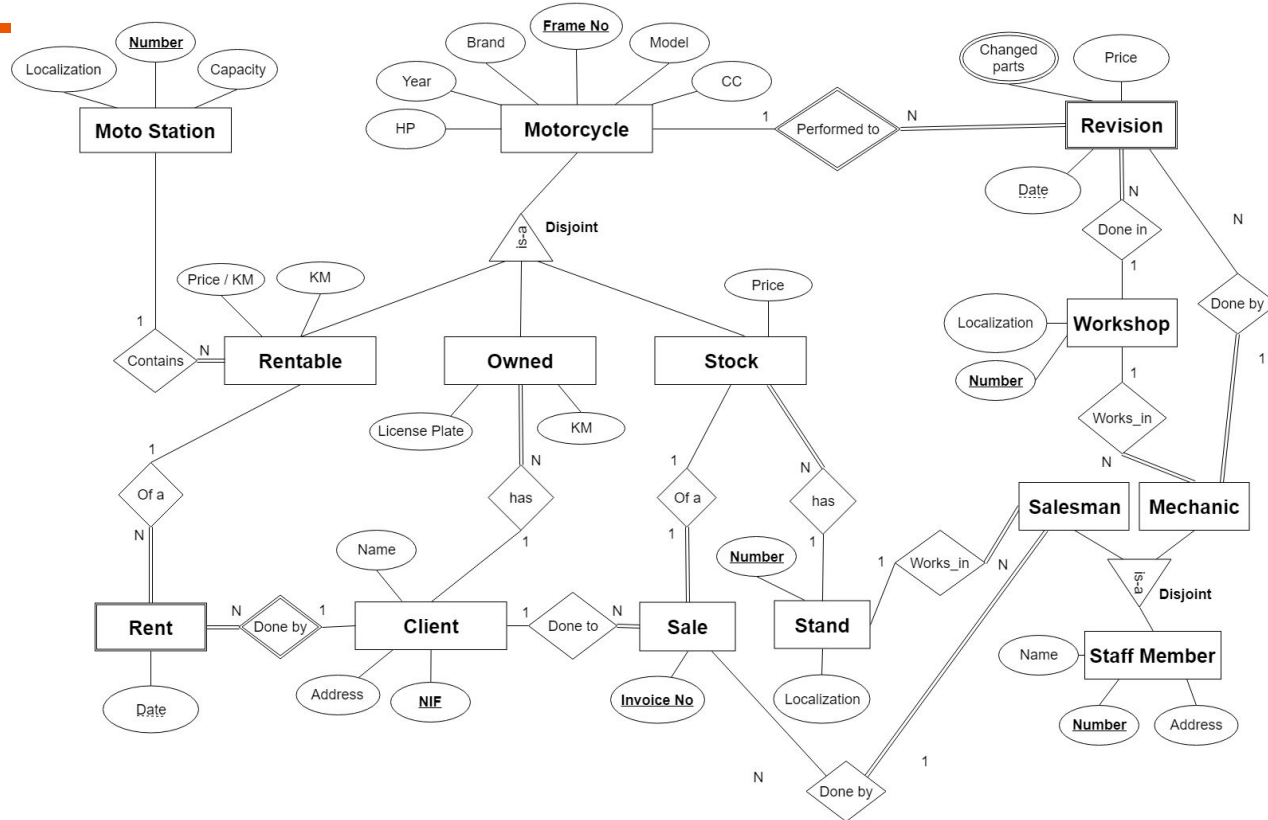
Context



Sistema POS

Gestão de recursos de uma loja de
motas

Entity Relationship Diagram



Views

```
GO
CREATE VIEW vw_REVISIONS AS
    SELECT REVISION.REVISION_NO AS Number, REVISION.R_DATE as Date, BRAND + ' ' + MODEL AS Motorcycle
    FROM (((REVISION JOIN MECHANIC ON REVISION.MECHANIC=MECHANIC.NUMBER)
    JOIN STAFF_MEMBER ON MECHANIC.NUMBER = STAFF_MEMBER.NUMBER)
    JOIN MOTORCYCLE ON MOTORCYCLE.FRAME_NO=REVISION.FRAME_NO);
GO
-----

GO
CREATE VIEW vw_RENTS AS
    SELECT R_DATE AS Date, BRAND + ' ' + MODEL AS Motorcycle
    FROM (((RENT JOIN MOTORCYCLE ON RENT.FRAME_NO=MOTORCYCLE.FRAME_NO)
    JOIN RENTABLE_BIKE ON RENT.FRAME_NO=RENTABLE_BIKE.FRAME_NO)
    JOIN MOTO_STATION ON RENTABLE_BIKE.MOTO_STATION=MOTO_STATION.NUMBER);
GO
-----

GO
CREATE VIEW STAFF_SALESMEN AS
    SELECT Member.NUMBER, Member.S_NAME, LOCALIZATION
    FROM ((SALESMAN JOIN STAFF_MEMBER AS Member ON SALESMAN.NUMBER = Member.NUMBER)
    JOIN STAND ON SALESMAN.WORK_STAND=STAND.NUMBER);
GO
-----
```

User Defined Functions

```
GO
CREATE FUNCTION BIKES_FROM_STAND(@stand INT) returns @Bike_table Table
    (Frame_no INT, Localization varchar(30), Motorcycle varchar(50), Price INT)
AS
BEGIN
    IF(@stand = -1)
    BEGIN
        INSERT @Bike_table SELECT STOCK_BIKE.FRAME_NO, LOCALIZATION As Stand, BRAND + ' ' + MODEL + ' ' + CONVERT(varchar, CC) AS Motorcycle, PRICE As Price
        FROM ((MOTORCYCLE JOIN STOCK_BIKE ON MOTORCYCLE.FRAME_NO=STOCK_BIKE.FRAME_NO)
        JOIN STAND ON STOCK_BIKE.STAND=STAND.NUMBER)

    END
    ELSE
    BEGIN
        INSERT @Bike_table SELECT STOCK_BIKE.FRAME_NO, LOCALIZATION As Stand, BRAND + ' ' + MODEL + ' ' + CONVERT(varchar, CC) AS Motorcycle, PRICE As Price
        FROM ((MOTORCYCLE JOIN STOCK_BIKE ON MOTORCYCLE.FRAME_NO=STOCK_BIKE.FRAME_NO)
        JOIN STAND ON STOCK_BIKE.STAND=STAND.NUMBER)
        WHERE STAND.NUMBER = @stand;
    END

    RETURN;
END
GO
-- Test function
-- select * FROM BIKES_FROM_STAND(-1);
-----
```

Stored Procedures

```
GO
CREATE PROCEDURE ms_INSERT_OWNED_MOTORCYCLE(@FRAME_NO INT, @MODEL VARCHAR(20), @BRAND VARCHAR(20), @YEAR INT, @CC INT, @HP INT,
@LICENSE_PLATE INT, @TOTAL_KM INT, @OWNER INT)
AS
    -- Insert in the parent table
    INSERT MOTORCYCLE (FRAME_NO, MODEL, BRAND, YEAR, CC, HP) VALUES (@FRAME_NO, @MODEL, @BRAND, @YEAR, @CC, @HP);
    -- Insert in the child table
    INSERT OWNED_BIKE(FRAME_NO, LICENSE_PLATE, TOTAL_KM, B_OWNER) VALUES (@FRAME_NO, @LICENSE_PLATE, @TOTAL_KM, @OWNER);
GO
-----

GO
CREATE PROCEDURE ms_ADD_STAND (@Localization varchar(150))
AS
    IF EXISTS (SELECT 1 FROM STAND WHERE LOCALIZATION=@Localization)
        RAISERROR ('Já existe um Stand em %s', 16, 1, @Localization);
    ELSE
        INSERT STAND(LOCALIZATION) VALUES (@Localization)
GO
-- Test Procedure
-- EXEC ms_ADD_STAND 'Lisboa'
-----
```

Triggers

```
GO
CREATE TRIGGER COMPLETE_SALE ON SALE
AFTER INSERT
AS
BEGIN
    DECLARE @sold_bike AS INT;
    DECLARE @sold_to AS INT;
    DECLARE @sold_bike_license AS INT = RAND()*(999999-100000)+100000;
    SELECT @sold_bike = MOTORCYCLE FROM inserted;
    SELECT @sold_to = CLIENT FROM inserted;

    DELETE FROM STOCK_BIKE WHERE FRAME_NO=@sold_bike
    INSERT INTO OWNED_BIKE (FRAME_NO, LICENSE_PLATE, TOTAL_KM, B_OWNER)
    VALUES(@sold_bike, @sold_bike_license, 0, @sold_to);
END
GO
-- Test trigger
-- INSERT INTO SALE (INVOICE_NO, CLIENT, MOTORCYCLE, SELLER) VALUES (4033, 306854700, 875506209, 696);
-----
```



DEMO