**Project 2 Milestone Report 2**

Sawan Chawla

Virginia Tech

Dr. Steven D. Sheetz

ACIS 5504: Information Systems Design and Database Concepts

December 2, 2024

**Table of Contents**

1. **Table of Contents ……………………………………………………………………… 2**
2. **DB Mapping ………..….……………..………………………………………………… 3**
3. **Revisions and Proposals …..……………..…………………………………………… 5**
4. **SQL Create Statements …………………..…………………………………………… 7**
5. **SQL Procedures …………………………..………………………………….……...… 10**
6. **Data Population SQL ….…......……………………………………………..………… 18**

**DB Mapping**





**Revisions and Proposals**

To make sure the data warehouse assists Hokie Resorts Hotels in becoming a market leader, we need to make a few changes to the operational databases. These updates will directly help us tackle management’s three big questions: which locations are most profitable, what unique value we can offer guests, and how to optimize revenue while cutting costs.

First, there are some gaps in the data that make it tough to analyze profitability by location. Corp1 has solid data on geography and revenue, but Corp2 is missing details like sales tax and specific income breakdowns by location. Adding fields like TourismActivity, EventRevenue, or a connection to regional tax rates would give us the insight we need to figure out which locations are thriving and which aren’t. This is huge for deciding where to invest more or where we need to change strategies such as offering more incentives at locations that see more guests and visitors or offering seasonal promotions in specific locations.

Second, when it comes to standing out from competitors, we need better data on customer preferences and behavior. Corp1 tracks some of this, but Corp2 doesn’t go deep enough. We don’t have enough detail to analyze things like repeat visits, feedback, or how guests are using premium services. Adding fields like ServicePreferences or LoyaltyEngagement and standardizing customer data across both systems would help us offer more personalized experiences. That’s how we create value guests can’t get anywhere else and keep them coming back.

On the cost side, optimizing revenue and cutting expenses means we need more visibility into operations. Corp2 doesn’t track things like staff scheduling or maintenance costs well enough. If we added fields like LaborCost, MaintenanceRequests, and ServiceUptime, it would give us the data we need to make more informed and efficient decisions such as adjusting staffing to match demand or spotting high-maintenance areas that could be streamlined.

One additional gap identified in both Corp1 and Corp2 databases is the lack of a dedicated time dimension. This is critical for analyzing trends over time, such as identifying profitability by quarter or month. To address this, the data warehouse will include a TimeDim table that organizes operational date fields into a structured hierarchy (e.g., year, quarter, month, day). This ensures that temporal analyses can be performed efficiently without requiring changes to the existing operational databases.

As for merging the two databases, I don’t think that’s the right move right now. Both systems are deeply tied to their own processes and combining them would require a complete overhaul. That in itself is a process that’s otherwise time consuming and is not worth the disruption it would cause. Instead, we should focus on building a strong ETL process that aligns schemas and handles inconsistencies, like making sure CustomerID and ReservationDates match across both databases. That’ll make reporting smoother without throwing everything into chaos.

By filling in these gaps and focusing on aligning the systems rather than merging them, we can build a data warehouse that answers management’s key questions. This will help us zero in on profitable locations, stand out with unique offerings, and make smarter decisions to cut costs and boost revenue. It’s a practical plan that’ll put Hokie Resorts on the path to becoming a leader in the hospitality market.

*AUTHOR’S NOTE: ALL SQL STATEMENTS WERE MADE WITH THE INTENT OF IMPLEMENTATION OF DATA INTO MICROSOFT ACCESS*

**SQL Create Statements**

DROP TABLE CustomerDim;

DROP TABLE InvoiceFact;

DROP TABLE ReservationFact;

CREATE TABLE CustomerDim (

CustomerID INTEGER PRIMARY KEY,

FirstName TEXT(50),

LastName TEXT(50),

MiddleName TEXT(50),

Address TEXT(100),

PhoneNumber TEXT(15)

);

CREATE TABLE InvoiceFact (

InvoiceID INTEGER PRIMARY KEY,

CustomerID INTEGER,

InvoiceDate DATE,

TotalAmount CURRENCY,

FOREIGN KEY (CustomerID) REFERENCES CustomerDim(CustomerID)

);

CREATE TABLE ReservationFact (

ReservationID INTEGER PRIMARY KEY,

CustomerID INTEGER,

ReservationDate DATE,

TotalReservations INTEGER,

AvgDuration DOUBLE,

FOREIGN KEY (CustomerID) REFERENCES CustomerDim(CustomerID)

);

CREATE TABLE EmployeeDim (

EmployeeID INTEGER PRIMARY KEY,

FirstName TEXT(50),

LastName TEXT(50),

MiddleName TEXT(50),

Address TEXT(100),

PhoneNumber TEXT(15),

Ranking TEXT(20),

Status TEXT(20)

);

**SQL Procedures**

BEGIN TRANSACTION;

INSERT INTO CustomerDim (CustomerID, FirstName, LastName, MiddleName, Address, PhoneNumber)

SELECT

C\_CustomerID\_Cu AS CustomerID,

T\_CustomerFirstName\_Cu AS FirstName,

T\_CustomerLastName\_Cu AS LastName,

T\_CustomerMiddleName\_Cu AS MiddleName,

T\_CustomerAddress\_Cu AS Address,

I\_CustomerPhoneNumber\_Cu AS PhoneNumber

FROM Corp1\_Customer;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp1\_Customer';

EXIT;

END IF;

INSERT INTO CustomerDim (CustomerID, FirstName, LastName, MiddleName, Address, PhoneNumber)

SELECT

C\_CustomerID\_Cu AS CustomerID,

T\_CustomerFirstName\_Cu AS FirstName,

T\_CustomerLastName\_Cu AS LastName,

T\_CustomerMiddleName\_Cu AS MiddleName,

T\_CustomerAddress\_Cu AS Address,

I\_CustomerPhoneNumber\_Cu AS PhoneNumber

FROM Corp2\_Customer;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp2\_Customer';

EXIT;

END IF;

INSERT INTO InvoiceFact (InvoiceID, CustomerID, InvoiceDate, TotalAmount)

SELECT

C\_InvoiceNumber\_In AS InvoiceID,

C\_CustomerID\_In AS CustomerID,

D\_IssuedDate\_In AS InvoiceDate,

N\_InvoiceTotal\_In AS TotalAmount

FROM Corp1\_Invoice;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp1\_Invoice';

EXIT;

END IF;

INSERT INTO InvoiceFact (InvoiceID, CustomerID, InvoiceDate, TotalAmount)

SELECT

C\_InvoiceNumber\_In AS InvoiceID,

C\_CustomerID\_In AS CustomerID,

D\_IssuedDate\_In AS InvoiceDate,

N\_InvoiceTotal\_In AS TotalAmount

FROM Corp2\_Invoice;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp2\_Invoice';

EXIT;

END IF;

INSERT INTO ReservationFact (ReservationID, CustomerID, ReservationDate, TotalReservations, AvgDuration)

SELECT

C\_FacilityReservationNumber\_Fr AS ReservationID,

I\_CustomerID\_Fr AS CustomerID,

D\_RentDateStart\_Fr AS ReservationDate,

COUNT(\*) AS TotalReservations,

AVG(DATEDIFF('d', D\_RentDateStart\_Fr, D\_RentDateEnd\_Fr)) AS AvgDuration

FROM Corp1\_FacilityReservation

GROUP BY I\_CustomerID\_Fr, D\_RentDateStart\_Fr;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp1\_FacilityReservation';

EXIT;

END IF;

INSERT INTO ReservationFact (ReservationID, CustomerID, ReservationDate, TotalReservations, AvgDuration)

SELECT

C\_RoomReservationNumber\_Rr AS ReservationID,

C\_CustomerID\_Rr AS CustomerID,

D\_ArrivalDate\_Rr AS ReservationDate,

COUNT(\*) AS TotalReservations,

AVG(DATEDIFF('d', D\_ArrivalDate\_Rr, D\_DepartureDate\_Rr)) AS AvgDuration

FROM Corp2\_RoomReservation

GROUP BY C\_CustomerID\_Rr, D\_ArrivalDate\_Rr;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp2\_RoomReservation';

EXIT;

END IF;

COMMIT TRANSACTION;

PRINT 'Data successfully loaded into CustomerDim, InvoiceFact, and ReservationFact tables.';

BEGIN TRANSACTION;

INSERT INTO EmployeeDim (EmployeeID, FirstName, LastName, MiddleName, Address, PhoneNumber, Ranking, Status)

SELECT

C\_EmployeeNumber\_Em AS EmployeeID,

T\_EmployeeFirstName\_Em AS FirstName,

T\_EmployeeLastName\_Em AS LastName,

T\_EmployeeMiddleName\_Em AS MiddleName,

T\_EmployeeAddress\_Em AS Address,

I\_EmployeePhoneNumber\_Em AS PhoneNumber,

T\_EmployeeRanking\_Em AS Ranking,

T\_EmployeeStatus\_Em AS Status

FROM Corp1\_Employees;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp1\_Employees';

EXIT;

END IF;

INSERT INTO EmployeeDim (EmployeeID, FirstName, LastName, MiddleName, Address, PhoneNumber, Ranking, Status)

SELECT

C\_EmployeeNumber\_Em AS EmployeeID,

T\_EmployeeFirstName\_Em AS FirstName,

T\_EmployeeLastName\_Em AS LastName,

T\_EmployeeMiddleName\_Em AS MiddleName,

T\_EmployeeAddress\_Em AS Address,

I\_EmployeePhoneNumber\_Em AS PhoneNumber,

T\_EmployeeRanking\_Em AS Ranking,

T\_EmployeeStatus\_Em AS Status

FROM Corp2\_Employees;

IF @@ERROR <> 0 THEN

ROLLBACK TRANSACTION;

PRINT 'Error occurred while inserting from Corp2\_Employees';

EXIT;

END IF;

COMMIT TRANSACTION;

PRINT 'Data successfully loaded into EmployeeDim table.';

**Data Population SQL**

INSERT INTO CustomerDim (CustomerID, FirstName, LastName, MiddleName, Address, PhoneNumber)

VALUES

(1, 'John', 'Doe', 'A', '123 Main St', '555-1234'),

(2, 'Jane', 'Smith', NULL, '456 Elm St', '555-5678'),

(3, 'Alice', 'Johnson', 'M', '789 Oak St', '555-6789'),

(4, 'Bob', 'Williams', 'L', '321 Pine St', '555-4321'),

(5, 'Emily', 'Brown', NULL, '654 Cedar St', '555-8765'),

(6, 'Charlie', 'Davis', 'J', '987 Birch St', '555-3456'),

(7, 'Sophia', 'Wilson', NULL, '111 Maple St', '555-6543'),

(8, 'Liam', 'Taylor', 'K', '222 Cherry St', '555-7654');

INSERT INTO InvoiceFact (InvoiceID, CustomerID, InvoiceDate, TotalAmount)

VALUES

(1001, 1, #2024-01-01#, 150.00),

(1002, 2, #2024-02-01#, 200.00),

(1003, 3, #2024-01-15#, 175.00),

(1004, 4, #2024-02-20#, 300.00),

(1005, 5, #2024-03-10#, 250.00),

(1006, 6, #2024-03-25#, 400.00),

(1007, 7, #2024-04-01#, 350.00),

(1008, 8, #2024-04-15#, 225.00);

INSERT INTO ReservationFact (ReservationID, CustomerID, ReservationDate, TotalReservations, AvgDuration)

VALUES

(5001, 1, #2024-01-10#, 2, 5.0),

(5002, 2, #2024-02-15#, 1, 3.0),

(5003, 3, #2024-03-05#, 4, 6.0),

(5004, 4, #2024-03-20#, 2, 7.0),

(5005, 5, #2024-04-10#, 3, 4.0),

(5006, 6, #2024-04-25#, 2, 5.0),

(5007, 7, #2024-05-05#, 5, 8.0),

(5008, 8, #2024-05-15#, 3, 6.5);

INSERT INTO EmployeeDim (EmployeeID, FirstName, LastName, MiddleName, Address, PhoneNumber, Ranking, Status)

VALUES

(101, 'Alice', 'Smith', 'L', '123 Elm St', '555-1234', 'Manager', 'Active'),

(102, 'Bob', 'Johnson', NULL, '456 Oak St', '555-5678', 'Technician', 'Inactive'),

(103, 'Charlie', 'Brown', 'M', '789 Pine St', '555-6789', 'Supervisor', 'Active'),

(104, 'Diana', 'Davis', 'J', '321 Cedar St', '555-4321', 'Cleaner', 'Active'),

(105, 'Eve', 'White', NULL, '654 Birch St', '555-8765', 'Manager', 'Active'),

(106, 'Frank', 'Black', 'K', '987 Maple St', '555-3456', 'Technician', 'Inactive'),

(107, 'Grace', 'Harris', NULL, '111 Willow St', '555-6543', 'Cleaner', 'Active'),

(108, 'Hank', 'Lee', 'J', '222 Cherry St', '555-7654', 'Supervisor', 'Active');