UTM

UNIVERSITI TEKNOLOGI MALAYSIA

Sem.1 2022/2023

SECD 2523 Database

Section 09

PHASE 4:

Database Logical Design & SQL

< Pesta Tanglung UTM Management System>

<Team SKTT1>

Team Members:

1. YEO CHUN TECK A21EC0148

2. LEW CHIN HONG A21EC0044

3. NG KAI ZHENG A21EC0101

4. GAN HENG LAI A21EC0176

1.0 Introduction

This project discusses the development of a new data management system for Pesta Tanglung UTM which is a university student club that frequently holds the events, especially Mooncake Festival and Cultural Night at the outdoor venue. The purpose of this project is to assist the Pesta Tanglung Club in developing a new data management system that could function more efficiently than the existing approach used by the club committees. There are various flaws in the PTUTM Club's present process approach that become problematic when the volume of data entering increases year after year. These issues can result in an unproductive process and needless time consumption which would subsequently affect the data that was correctly saved.

Consequently, it is necessary for the new system to take the place of the existing process technique. This will allow the committees to more easily handle things like automatically updating or changing student information and avoiding data duplication in the club database. The information synchronization will let the committee access the information and activity planning of the members.

Phase 4 of the project will involve transforming the conceptual Entity Relationship Diagram (ERD) developed in Phase 3 into a logical ERD by removing non-relational features such as many-to-many relationships and complex relationships. From this logical ERD, the schema of the relation will be derived and the relations will be normalized up to Boyce-Codd Normal Form (BCNF). The final logical ERD will be drawn to represent the BCNF relations, and the data dictionary will be updated based on these normalized relations. The logical ERD will also be validated against the system's transaction requirements.

As a deliverable, a report will be produced containing the logical ERD (global data model), the relational database schemas after normalization, and the updated data dictionary for the normalized logical design. The report will also include SQL statements for Data Definition Language (DDL) and Data Manipulation Language (DML).

2.0 Overview of Project

The project is developed to design a database for the Pesta Tanglung Club in Universiti Teknologi Malaysia (UTM) to improve their student management system. The current system used is a manual and file-based system which could cause a lot of human error and low efficiency. To improve the current system, we proposed a new database which will store all the information related to student management.

Based on the requirement and information gathered from the interview section with the secretary of the club, we have known more about the user requirements, their system weakness and needs. Therefore, we designed the new proposed database system based on the user requirements. The proposed database allows the user to establish different categories of relations from the raw data according to the requirements in a short instant, unification of data and the most essential part in the new system is the maintenance of consistency and accuracy of data which could eliminate the human type in error.

3.0 Database conceptual design

3.1 Update business rule

Pesta Tanglung has one president, one vice president, one general secretary, two deputy secretaries, one general treasurer, and two deputy treasurers and a total of 10 teams exist in the club. These teams are the general affair team, design team, publicity team, sales team, venue team, event organizing team, fundraising team, cultural information team, entertainment team and multimedia team. Each team must have only one team leader. Each team must have only one team leader. Each team must have at least one member. Every member of the club must be assigned to only one team. Every member is managed by their team leader.

Pesta Tanglung Club is in charge of one lecturer in UTM. A UTM lecturer is only in charge of a club. The users of the proposed database could be members of the club and lecturer, The authority to update and control the database was given to the president, the vice president, the general secretary and every team leader. Other members could only view the data from the database. The proposals could be checked by one lecturer.

The users who wish to access the database have to sign up hence log in to the database. All of the members of the club are assigned to 1 of those 10 teams. Each team have a team leader. The responsibility of the team leader is to manage the team member.

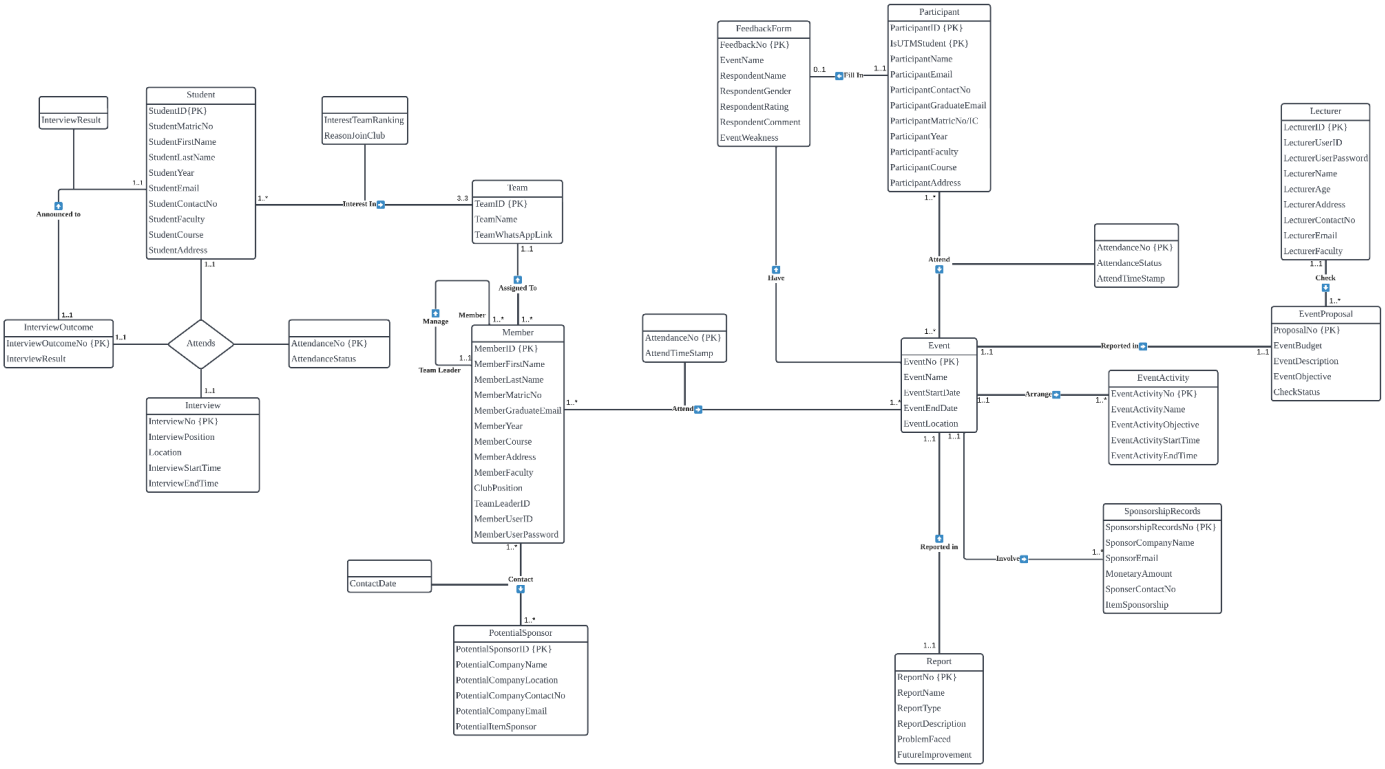
Every year, the club have an interview for a recruitment session. Each of the interviewees could choose the top 3 interest team he/she want to join. However, not all of the interviewees would pass the interview and join the most interesting team. The interviewees' attendance will be recorded. The interviewees who passed the interview will be assigned to a team suitable for their capability according to the decision of the 10 team leaders. The interview outcome will be announced to all of the interviewees no matter he/she passed or failed.

When an event such as Mooncake Festival and Cultural Night event is held, participant attendance will be recorded for analysis use and UTM merit purpose (for those who are UTM students), as well as a feedback form, will be distributed to gather the comment from the

participants so could overcome the event weakness. An event starts with an event proposal that has to submit to the UTM ACAD and wait for the approval. Many activities would be arranged in an event. All of the fundraisers in the club will start to find and contact the potential sponsor, and all of the fundraisers have the responsibility to follow up on the latest circumstances and the contact date will be recorded. Furthermore, for every event organized, each sponsorship will be recorded to know what the item/monetary amount the sponsorship sponsor is.

A report has to be generated after conducting every program for analysis purposes and submitted to HEP. Several analyses will be done in a report, such as the quantity of UTM students and the quantity of Non-UTM students. Lastly, from the feedback form gathered, the member is able to realize the problem faced to perform a future improvement.

3.2 Conceptual ERD

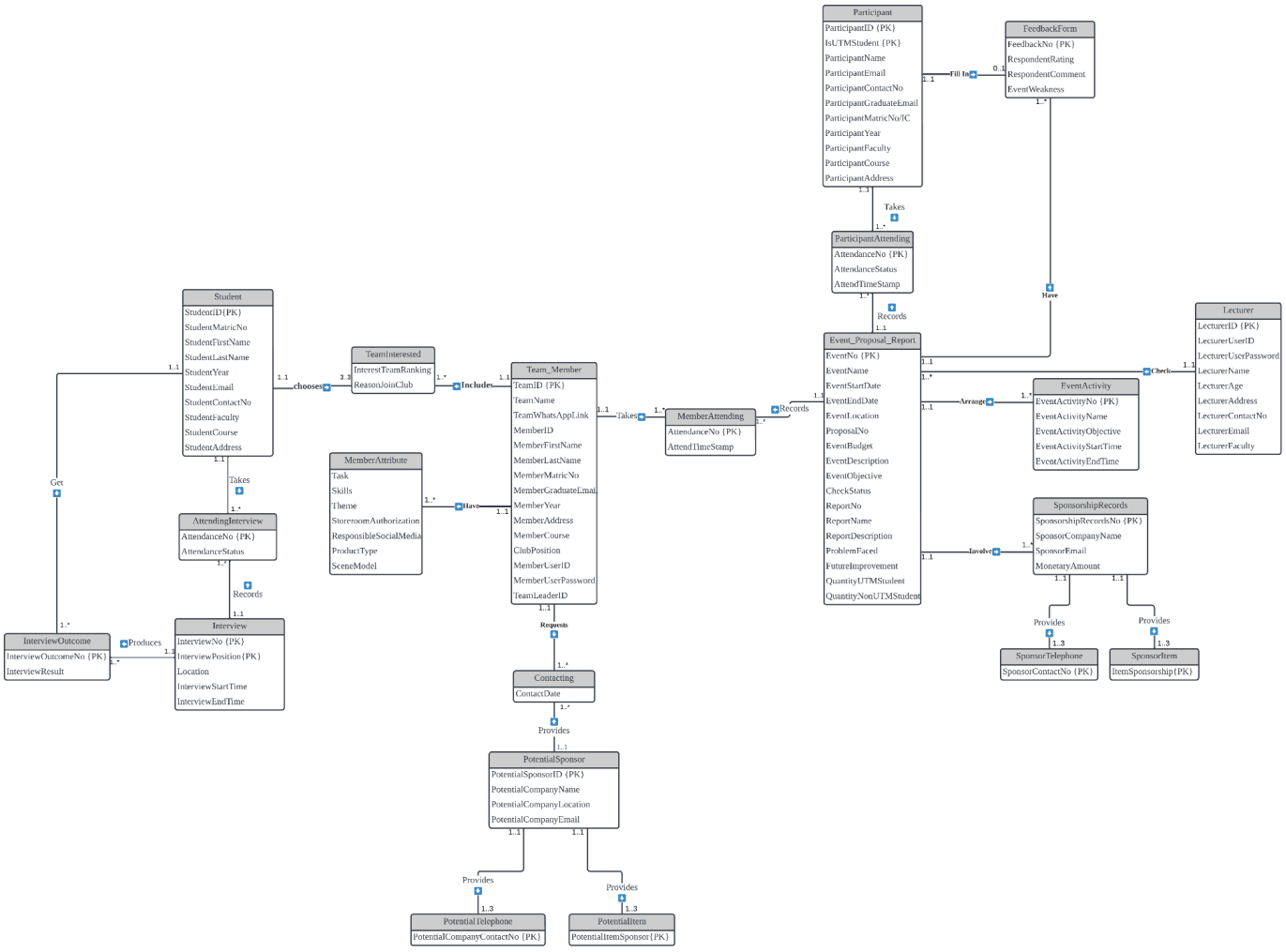


For a better view, please visit:

https://lucid.app/documents/view/61432891-b766-4c7a-a12e-3219ca42ec61

4.0 DB logical design

4.1 Logical ERD



For a better view, please visit:

https://lucid.app/documents/view/e5166a9d-4277-47a8-a43b-582323511a4c

4.2 Updated Data dictionary

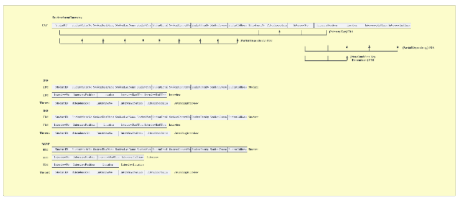
Entity	Data to be stored	Description	Data Length & Type	Null	Multi-value
Team	• TeamID	Unique ID	10 variable characters	NO	NO
	• TeamName	Team's name	20 variable characters	NO	NO
	• TeamWhatsAppLink	Whatsapp application link	50 variable characters	NO	NO
	• MemberID	Unique ID	10 variable characters	NO	NO
	• MemberFirstName	First name of member	20 variable characters	NO	NO
	• MemberLastName	Last name of member	20 variable characters	NO	NO
	• MemberMatricNo	Unique matric no	10 variable characters	NO	NO
	• MemberGraduateEmail	Unique UTM email of member	30 variable characters	NO	NO
	• MemberYear	Degree year of member	Number	NO	NO
	• MemberCourse	Course of member	30 variable characters	NO	NO
	• MemberAddress	Address of member	80 variable characters	NO	NO
	• MemberFaculty	Faculty of member	50 variable characters	NO	NO
	• ClubPosition	Position of member in club	20 variable characters	NO	NO
	• MemberUserID	Unique ID	10 variable characters	NO	NO
	• MemberUserPassword	User password for member to access the system	15 variable characters	NO	NO
	• TeamLeaderID	UniqueID	10 variable characters	NO	NO
MemberAttribute	• Task	Task of each member	100 variable characters	YES	NO
	• Skills	Skills of each member	50 variable characters	YES	NO
	• Theme	Theme of entertainment	50 variable characters	YES	NO
	• StoreroomAuthorization	Member authority status of storeroom	1 variable characters	YES	NO
	• ResponsibleSocialMedia	Handle's social media	50 variable characters	YES	NO
	• ProductType	Type of product	50 variable characters	YES	NO
	• SceneModel	Production of scene props	50 variable characters	YES	NO
EventProposal	• ProposalNo	Unique NO	10 variable characters	NO	NO
	• EventBudget	Budget of event	Number	NO	NO
	• EventDescription	Description of event	200 variable characters	NO	NO
	• EventObjective	Objective of event	200 variable characters	NO	NO
	• CheckStatus	Status of checking	1 variable character	NO	NO
Student	• StudentID	Unique ID	10 variable characters	NO	NO
	• StudentMatricNo	Unique matric no of Student	10 variable characters	NO	NO
	• StudentFirstName	First name of Student	20 variable characters	NO	NO
	• StudentLastName	Last name of Student	20 variable characters	NO	NO
	• StudentYear	Degree year of Student	Number	NO	NO
	• StudentEmail	Email of Student	30 variable characters	NO	NO
	• StudentContactNo	Contact number of Student	Number	NO	NO
	• StudentFaculty	Faculty of Student	50 variable characters	NO	NO
	• StudentCourse	Course of Student	30 variable characters	NO	NO

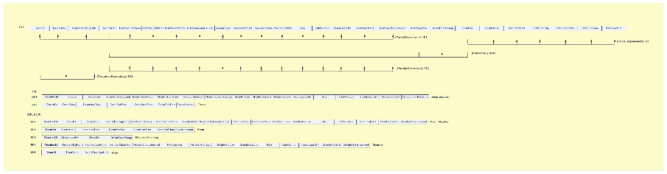
	• StudentAddress	Address of Student	80 variable characters	NO	NO
Interview	• InterviewNo • InterviewPosition • InterviewDate • StartTime • EndTime • Location	Unique ID Interview's club position Date of interview having Starting time for interview Ending time for interview Location of interview	10 variable characters 50 variable characters Date Timestamp Timestamp 80 variable characters	NO NO NO NO NO NO	NO NO NO NO NO NO
InterviewOutcome	• InterviewOutcomeNo • InterviewComment • InterviewStrength • InterviewResult	Unique ID Comment of interviewee Strength of interviewee Result of interview	10 variable characters 100 variable characters 100 variable characters 1 variable characters	NO NO NO NO	NO NO NO NO
TeamInterested	• InterestTeamRanking • ReasonJoinClub	Ranking of team interested Reason of joining club	Number 50 variable characters	NO NO	NO NO
AttendingInterview	• AttendanceNo • AttendanceStatus	Unique NO Status of attendance	10 variable characters 1 variable characters	NO NO	NO NO
Lecturer	• LecturerID • LecturerUserID • LecturerUserPassword • LecturerName • LecturerAge • LecturerAddress • LecturerContactNo • LecturerEmail • LecturerFaculty	Unique ID Unique ID User password for lecturer to access the system Name of lecturer Age of lecturer Address of lecturer Contact number of lecturer Email of lecturer Faculty of lecturer	10 variable characters 10 variable characters 15 variable characters 40 variable characters Number 80 variable characters Number 30 variable characters 50 variable characters	NO NO NO NO NO NO NO NO NO	NO NO NO NO NO NO NO NO NO
Participant	• ParticipantID • IsUTMStudent • ParticipantName • ParticipantEmail • ParticipantContactNo • ParticipantGraduateEmail • ParticipantMatricNo/IC • ParticipantYear • ParticipantFaculty • ParticipantCourse • ParticipantAddress	Unique ID Whether the Participant is UTM Student Participant's name Participant's email address Participant's contact number Participant's UTM email address Participant's MatricNo/IC Participant's degree year Participant's faculty Participant's course Participant's address	10 variable characters 1 variable characters 40 variable characters 30 variable characters Number 30 variable characters 15 variable characters Number 50 variable characters 30 variable characters 80 variable characters	NO NO NO NO NO YES YES YES YES YES YES	NO NO NO NO NO NO NO NO NO NO NO
FeedbackForm	• FeedbackNo • RespondentRating • RespondentComment • EventWeakness	Unique NO Respondent's rating Respondent's comment Event's weakness	10 variable characters Number 200 variable characters 200 variable characters	NO NO YES YES	NO NO NO NO

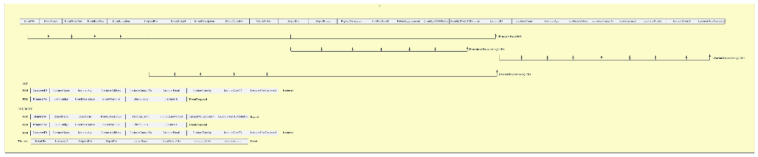
Event	• EventNo • EventName • EventStartDate • EventEndDate • EventLocation	Unique NO Event's name Event's start date and time Event's end date and time Event held location	10 variable characters 20 variable characters Timestamp Timestamp 100 variable characters	NO NO NO NO NO	NO NO NO NO NO
MemberAttending	• AttendanceNo • AttendTimeStamp	Unique NO Time of the member attend	10 variable characters Timestamp	NO NO	NO NO
ParticipantAttending	• AttendanceNo • AttendTimeStamp • AttendanceStatus	Unique NO Time of the participant attend Status of attendance	10 variable characters Timestamp 1 variable characters	NO NO NO	NO NO NO
EventActivity	• EventActivityNo • EventActivityName • EventActivityObjective • EventActivityDescription • EventActivityStartTime • EventActivityEndTime	Unique NO Event activity name Object of event activity Event activity's description Event activity's start time Event activity's end time	10 variable characters 50 variable characters 100 variable characters 100 variable characters Timestamp Timestamp	NO NO NO NO NO NO	NO NO NO NO NO NO
Report	• ReportNo • ReportName • ReportType • ReportDescription • ProblemFaced • FutureImprovement • QuantityUTMStudent • QuantityNonUTMStudent	Unique NO Report's name Report's type Report's description Problem faced during event Future Improvement of event Amount of UTM student Amount of non-UTM student	10 variable characters 50 variable characters 30 variable characters 200 variable characters 200 variable characters 200 variable characters Number Number	NO NO NO NO NO NO NO NO	NO NO NO NO NO NO NO NO
PotentialSponsor	• PotentialSponsorID • PotentialCompanyName • PotentialCompanyLocation • PotentialCompanyEmail	Unique ID Names of potential sponsoring company Location of potential sponsoring company Email address of potential sponsoring company	10 variable characters 50 variable characters 80 variable characters 30 variable characters	NO NO NO NO	NO NO NO NO
PotentialTelephone	• PotentialCompanyContactNo	Contact number of potential sponsoring company	Number	NO	YES
PotentialItem	• PotentialItemSponsor	Potential item sponsored by company	80 variable characters	NO	YES
Reading	• ReadDate	Reading proposal's date	Date	NO	NO
SponsorshipRecords	• SponsorshipRecordsNo • SponsorCompanyName • SponsorEmail • MonetaryAmount	Unique NO The name of the sponsoring company The email address of the sponsor Amount of sponsorship	10 variable characters 50 variable characters 50 variable characters Number	NO NO NO YES	NO NO NO NO

SponsorTelephone	• SponsorContactNo	Contact number of the sponsor	Number	NO	YES
SponsorItem	• ItemSponsorship	Item of sponsor	80 variable characters	NO	YES
Contacting	• ContactDate	Date of contacting	Date	NO	NO

4.3 Normalization







For better view, please visit:

https://lucid.app/documents/view/0c6e6b1a-1f50-4a59-a335-bf9c7614b9fd

5.0 Relational DB Schemas (after normalization)

Team(**TeamID**, TeamName, TeamWhatsAppLink)

Member(**MemberID**, TeamID, MemberFirstName, MemberLastName, MemberMatricNo, MemberGraduateEmail, MemberYear, MemberAddress, MemberCourse, ClubPosition, MemberUserID, MemberUserPassword, TeamLeaderID)

MemberAttribute(Task, Skills, Theme, StoreroomAuthorization, ResponsibleSocialMedia, ProductType, SceneModel, MemberID)

Student(**StudentID,** StudentMatricNo, StudentFirstName, StudentLastName, StudentYear, StudentEmail, StudentContactNo, StudentFaculty, StudentCourse, StudentAddress

TeamInterested(**StudentID, TeamID,** IntereseTeamRanking, ReasonJoinClub)

Interview(**InterviewNo, InterviewPosition**,InterviewStartTime, InterviewEndTime)

InterviewLocation(**InterviewNo, InterviewPosition,** Location)

AttendingInterview(**StudentID, InterviewNo, AttendanceNo, InterviewPosition** AttendanceStatus)

InterviewOutcome(**StudentID, InterviewNo, InterviewOutcomeNo,** InterviewResult)

PotentialSponsor(**PotentialSponsorID,** PotentialCompanyName, PotentialCompanyLocation, PotentialCompanyEmail)

PotentialTelephone(**PotentialCompanyContactNo,** PotentialSponsorID)

PotentialItem(**PotentialCompanySponsor,** PotentialSponsorID)

Contacting(**MemberID, PotentialSponsorID,** ContactDate)

EventProposal(**ProposalNo**, EventBudget, EventDescription, EventObjective, CheckStatus, LecturerID)

ProgramReport(**ReportNo,** ReportName, ReportDescription, ProblemFaced, FutureImprovement, QuantityUTMStudent, QuantityNonUTMStudent)

Lecturer(**LecturerID,** LecturerUserID, LecturerUserPassword, LecturerName, LecturerAge, LecturerAddress, LecturerContactNo, LecturerEmail, LecturerFacutly)

Event(**EventNo,** EventName, EventStartDate, EventEndDate, EventLocation, ProposalNo, ReportNo)

MemberAttending(**MemberID, EventNo, AttendanceNo,** AttendTimeStamp)

Participant(**ParticipantID, IsUTMStudent,** ParticipantName, ParticipantEmail, ParticipantContactNo, ParticipantGraduateEmail, ParticipantMatricNo/IC, ParticipantYear, ParticipantFaculty, ParticipantCourse, ParticipantAddress)

ParticipnatAttending(**EventNo, ParticipantID, IsUTMStudent, AttendanceNo,** AttendanceStatus, AttendanceTimeStamp)

FeedbackForm(**FeedbackNo**, RespondentRating, RespondentComment, EventWeakness, ParticipantID, IsUTMStudent, EventNo)

EventActivity(**EventActivityNo,** EventActivityName, EventActivityObjective, EventActivityStartTime, EventActivityEndTime, EventNo)

SponsorshipRecords(**SponsorshipRecordsNo**, SpnsorCompanyNmae, SponsorEmail, MonetaryAmount, EventNo)

SponsorTelephone(**SponsorContactNo,** SponsorshipRecordsNo)

SponsorItem(**ItemSponsorship,** SponsorshipRecordsNo)

6.0 SQL Statements (DDL & DML)

```
CREATE TABLE Team(
   TeamID VARCHAR2(10) NOT NULL,
   TeamName VARCHAR2(20),
   TeamWhatsAppLink VARCHAR2(50),
   CONSTRAINT PK_Team PRIMARY KEY (TeamID)
);


CREATE TABLE Member(
   MemberID VARCHAR2(10) NOT NULL,
   MemberFirstName VARCHAR2(20),
   MemberLastName VARCHAR2(20),
   MemberMatricNo VARCHAR2(10),
   MemberGraduateEmail VARCHAR2(30),
   MemberYear NUMBER,
   MemberAddress VARCHAR2(80),
   MemberCourse VARCHAR2(30),
   ClubPosition VARCHAR2(20),
   MemberUserID VARCHAR2(10),
   MemberUserPassword VARCHAR2(15),
   TeamLeaderID VARCHAR2(10),
   TeamID VARCHAR2(10),
   CONSTRAINT PK_Member PRIMARY KEY (MemberID),
```

```
        CONSTRAINT fk_MemberTeam FOREIGN KEY (TeamID) REFERENCES TEAM(TeamID)

);


CREATE TABLE MemberAttribute(

   Task VARCHAR2(100),

   Skills VARCHAR2(50),

   Theme VARCHAR2(50),

   StoreroomAuthorization VARCHAR2(1),

   ResponsibleSocialMedia VARCHAR2(50),

   ProductType VARCHAR2(50),

   SceneModel VARCHAR2(50),

   MemberID VARCHAR2(10),

        CONSTRAINT FK_MemberID FOREIGN KEY (MemberID) REFERENCES Member(MemberID)

);


CREATE TABLE Student(

   StudentID VARCHAR2(10) NOT NULL,

   StudentMatricNo VARCHAR2(10),

   StudentFirstName VARCHAR2(20),

   StudentLastName VARCHAR2(20),

   StudentYear NUMBER,

   StudentEmail VARCHAR2(50),
```

```
    StudentContactNo NUMBER,
    StudentFaculty VARCHAR2(50),
    StudentCourse VARCHAR2(30),
    StudentAddress VARCHAR2(80),
    CONSTRAINT PK_Student PRIMARY KEY(StudentID)
);

CREATE TABLE TeamInterested(
    StudentID VARCHAR2(10) NOT NULL,
    TeamID VARCHAR2(10) NOT NULL,
    InteresetTeamRanking NUMBER,
    ReasonJoinClub VARCHAR2(50),
        CONSTRAINT FK_StudentID FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
    CONSTRAINT FK_TeamID FOREIGN KEY (TeamID) REFERENCES Team(TeamID),
    CONSTRAINT PK_TeamInterested PRIMARY KEY(StudentID, TeamID)
);

CREATE TABLE Interview(
    InterviewNo VARCHAR2(10) NOT NULL,
    InterviewPosition VARCHAR2(50) NOT NULL,
    Location VARCHAR2(80),
    InterviewStartTime TIMESTAMP,
    interviewEndTime TIMESTAMP,
```

```
    CONSTRAINT PK_Interview PRIMARY KEY(InterviewNo, InterviewPosition)
);

CREATE TABLE INTERVIEWLOCATION
(
    InterviewNo VARCHAR2(10) NOT NULL,
    InterviewPosition VARCHAR2(50) NOT NULL,
    Location VARCHAR2(80),
        CONSTRAINT PK_InterviewLocation PRIMARY KEY (InterviewNo, InterviewPosition),
        CONSTRAINT FK_InterviewLocation FOREIGN KEY (InterviewNo, InterviewPosition) REFERENCES INTERVIEW(InterviewNo, InterviewPosition)
);

CREATE TABLE AttendingInterview
(
    StudentID VARCHAR2(10) NOT NULL,
    InterviewNo VARCHAR2(10) NOT NULL,
    InterviewPosition VARCHAR2(50) NOT NULL,
    AttendanceNo VARCHAR2(10) NOT NULL,
    AttendanceStatus VARCHAR2(1),
     CONSTRAINT FK_InterviewStudentID FOREIGN KEY(StudentID) REFERENCES Student(StudentID),
      CONSTRAINT FK_InterviewNo FOREIGN KEY(InterviewNo, InterviewPosition) REFERENCES Interview(InterviewNo, InterviewPosition),
```

CONSTRAINT PK_AInterview PRIMARY KEY(StudentID, InterviewNo, InterviewPosition, AttendanceNo)

);

CREATE TABLE InterviewOutcome(

StudentID VARCHAR2(10) NOT NULL,

InterviewNo VARCHAR2(10) NOT NULL,

InterviewPosition VARCHAR2(50) NOT NULL,

InterviewOutcomeNo VARCHAR2(10) NOT NULL,

InterviewResult VARCHAR2(1),

CONSTRAINT PK_InterviewOutcome PRIMARY KEY(StudentID, InterviewNo, InterviewPosition, InterviewOutcomeNo),

CONSTRAINT FK_OutcomeStudentID FOREIGN KEY(StudentID) REFERENCES Student(StudentID),

CONSTRAINT FK_OutcomeInterviewNo FOREIGN KEY(InterviewNo, InterviewPosition) REFERENCES Interview(InterviewNo, InterviewPosition)

);

CREATE TABLE PotentialSponsor(

PotentialSponsorID VARCHAR2(10) NOT NULL,

PotentialCompanyName VARCHAR2(50),

PotentialCompanyLocation VARCHAR2(80),

PotentialCompanyEmail VARCHAR2(50),

CONSTRAINT PK_PotentialSponsor PRIMARY KEY(PotentialSponsorID)

);

```
CREATE TABLE PotentialTelephone(
   PotentialCompanyContactNo NUMBER NOT NULL,
   PotentialSponsorID VARCHAR2(10) NOT NULL,
   CONSTRAINT PK_PTelephone PRIMARY KEY(PotentialCompanyContactNo),
    CONSTRAINT FK_PSponsorID FOREIGN KEY(PotentialSponsorID) REFERENCES PotentialSponsor(PotentialSponsorID)
);

CREATE TABLE PotentialItem(
   PotentialCompanySponsor VARCHAR2(80) NOT NULL,
   PotentialSponsorID VARCHAR2(10) NOT NULL,
   CONSTRAINT PK_PItem PRIMARY KEY(PotentialCompanySponsor),
        CONSTRAINT FK_PItemSponsorID FOREIGN KEY(PotentialSponsorID) REFERENCES PotentialSponsor(PotentialSponsorID)
);

CREATE TABLE Contacting(
   MemberID VARCHAR2(10) NOT NULL,
   PotentialSponsorID VARCHAR2(10) NOT NULL,
   ContactDate DATE,
    CONSTRAINT FK_ContactMemberID FOREIGN KEY(MemberID) REFERENCES Member(MemberID),
        CONSTRAINT FK_ContactSponsorID FOREIGN KEY(PotentialSponsorID) REFERENCES PotentialSponsor(PotentialSponsorID),
```

```
    CONSTRAINT PK_Contacting PRIMARY KEY(MemberID, PotentialSponsorID)
);

CREATE TABLE Lecturer(
        LecturerID VARCHAR2(10) NOT NULL,
        LecturerUserID VARCHAR2(10),
        LecturerUserPassword VARCHAR2(15),
        LecturerName VARCHAR2(40),
        LecturerAge NUMBER,
        LecturerAddress VARCHAR2(80),
        LecturerContactNo NUMBER,
        LecturerEmail VARCHAR2(30),
        LecturerFaculty VARCHAR2(50),
        CONSTRAINT PK_LecturerID PRIMARY KEY (LecturerID)
);

CREATE TABLE EventProposal(
        ProposalNo VARCHAR(10) NOT NULL,
        EventBudget NUMBER,
        EventDescription VARCHAR2(200),
        EventObjective VARCHAR2(200),
        CheckStatus VARCHAR2(1),
        LecturerID VARCHAR2(10),
```

```
    CONSTRAINT PK_EventProposal PRIMARY KEY (ProposalNo),

    CONSTRAINT FK_LecturerID FOREIGN KEY (LecturerID) REFERENCES Lecturer(LecturerID)

);


CREATE TABLE PROGRAMREPORT

(

   ReportNo VARCHAR2(10) NOT NULL,

   ReportName VARCHAR2(50),

   ReportDescription VARCHAR2(200),

   ProblemFaced VARCHAR2(200),

   FutureImprovement VARCHAR2(200),

   QuantityUTMStudent NUMBER,

   QuantityNonUTMStudent NUMBER,

   CONSTRAINT pk_ProgramReport PRIMARY KEY (ReportNo)

);


CREATE TABLE EVENT

(

   EventNo VARCHAR2(10) NOT NULL,

   EventName VARCHAR(20),

   EventStartDate TIMESTAMP,

   EventEndDate TIMESTAMP,

   EventLocation VARCHAR2(100),
```

```
   ProposalNo VARCHAR2(10),

   ReportNo VARCHAR2(10),

   CONSTRAINT pk_Event PRIMARY KEY (EventNo),

      CONSTRAINT fk_EventProposal FOREIGN KEY (ProposalNo) REFERENCES EVENTPROPOSAL(ProposalNo),

       CONSTRAINT fk_EventReport FOREIGN KEY (ReportNo) REFERENCES PROGRAMREPORT(ReportNo)

);


CREATE TABLE PARTICIPANT

(

   ParticipantID VARCHAR2(10) NOT NULL,

   IsUTMStudent VARCHAR2(1) NOT NULL,

   ParticipantName VARCHAR2(40),

   ParticipantEmail VARCHAR2(50),

   ParticipantContactNo NUMBER,

   PariticipantGraduateEmail VARCHAR2(50),

   ParticipantMatricNo_IC VARCHAR2(15),

   ParticipantYear NUMBER,

   ParticipantFaculty VARCHAR2(50),

   ParticipantCourse VARCHAR2(30),

   ParticipantAddress VARCHAR2(80),

   CONSTRAINT pk_Participant PRIMARY KEY (ParticipantID,IsUTMStudent)

);
```

```
CREATE TABLE PARTICIPANTATTENDING

(

   AttendanceNo VARCHAR2(10) NOT NULL,

   AttendTimeStamp TIMESTAMP,

   AttendanceStatus VARCHAR2(1),

   ParticipantID VARCHAR2(10),

   IsUTMStudent VARCHAR2(1),

   EventNo VARCHAR2(10),

         CONSTRAINT pk_ParticipantAttending PRIMARY KEY (AttendanceNo, ParticipantID, IsUTMStudent, EventNo),

         CONSTRAINT fk_ParticipnatAttending FOREIGN KEY (ParticipantID, IsUTMStudent) REFERENCES PARTICIPANT(ParticipantID, IsUTMStudent),

         CONSTRAINT fk_ParticipnatAttendingEvent FOREIGN KEY (EventNo) REFERENCES EVENT(EventNo)

);


CREATE TABLE FEEDBACKFORM

(

   FeedbackNo VARCHAR2(10) NOT NULL,

   RespondentRating NUMBER,

   RespondentComment VARCHAR2(200),

   EventWeakness VARCHAR2(200),

   EventNo VARCHAR2(10),

   ParticipantID VARCHAR2(10),
```

IsUTMStudent VARCHAr2(1),

CONSTRAINT pk_FeedbackForm PRIMARY KEY (FeedbackNo),

CONSTRAINT fk_FeedbackFormEvent FOREIGN KEY (EventNo) REFERENCES EVENT(EVentNo),

CONSTRAINT fk_FeedbackParticipant FOREIGN KEY (ParticipantID, IsUTMStudent) REFERENCES PARTICIPANT(ParticipantID, IsUTMStudent)

);

CREATE TABLE MEMBERATTENDING

(

AttendanceNo VARCHAR2(10) NOT NULL,

AttendTimeStamp TIMESTAMP,

MemberID VARCHAR2(10),

EventNo VARCHAR2(10),

CONSTRAINT pk_MemberAttending PRIMARY KEY (AttendanceNo, MemberID, EventNo),

CONSTRAINT fk_MemberAttendingTeam FOREIGN KEY (MemberID) REFERENCES MEMBER(MemberID),

CONSTRAINT fk_MemberAttendingEvent FOREIGN KEY (EventNo) REFERENCES EVENT(EventNo)

);

CREATE TABLE EVENTACTIVITY

(

EventActivityNo VARCHAR2(10),

```
    EventActivityName VARCHAR2(50),

    EventActivityObjective VARCHAR2(100),

    EventActivityStartTime TIMESTAMP,

    EventAvtivityEndTime TIMESTAMP,

    EventNo VARCHAR(10),

    CONSTRAINT pk_EventActivity PRIMARY KEY (EventActivityNo),

    CONSTRAINT fk_EventActivity FOREIGN KEY (EventNo) REFERENCES EVENT(EventNo)

);


CREATE TABLE SPONSORSHIPRECORDS

(

    SponsorshipRecordsNo VARCHAR2(10) NOT NULL,

    SponsorsCompanyName VARCHAR2(50),

    SponsorEmail VARCHAR2(50),

    MonetaryAmount NUMBER,

    EventNo VARCHAR(10),

    CONSTRAINT pk_SponsorshipRecords PRIMARY KEY (SponsorshipRecordsNo),

    CONSTRAINT fk_SponsorshipRecords FOREIGN KEY (EventNo) REFERENCES EVENT(EventNo)

);


CREATE TABLE SPONSORTELEPHONE

(
```

```
    SponsorContactNo NUMBER NOT NULL,

    SponsorshipRecordsNo VARCHAR2(10),

    CONSTRAINT pk_SponsorTelephone PRIMARY KEY (SponsorContactNo),

    CONSTRAINT fk_SponsorTelephone FOREIGN KEY (SponsorshipRecordsNo) REFERENCES SPONSORSHIPRECORDS(SponsorshipRecordsNo)

);


CREATE TABLE SPONSORITEM

(

    ItemSponsorship VARCHAR2(80) NOT NULL,

    SponsorshipRecordsNo VARCHAR2(10),

    CONSTRAINT  pk_SponsorItem PRIMARY KEY (ItemSponsorship),

      CONSTRAINT fk_SponsorItem FOREIGN KEY (SponsorshipRecordsNo) REFERENCES SPONSORSHIPRECORDS(SponsorshipRecordsNo)

);
```

7.0 Summary

The main focus of the phase 4 project was to design the logical structure of the database for the Pesta Tanglung UTM Club. This included transforming the conceptual ERD from phase 3 into a logical ERD by removing non-relational features such as many-to-many relationships and complex relationships. The relations schema was then derived from the logical ERD and the relations were normalized up to BCNF. The final logical ERD was drawn to represent the BCNF relations, and the data dictionary was updated based on the normalized relations. The logical ERD was also validated against the system's transaction requirements.

The deliverables for this phase included a report containing the following:

- Logical ERD (global data model)
- Relational database schemas (normalized up to BCNF)
- Data dictionary for the normalized logical design
- SQL statements for data definition and manipulation

The report was used to assess the logical design of the database and ensure that it was able to meet the needs of the Pesta Tanglung UTM Club. The report was also used to document the design process and provide information on the normalized relations and data dictionary for future reference.