Table of Contents

INTRODUCTION

Scenario

This report is based on the design of an information system for an international humanitarian organization, *Aid for All*, which receives donations from donors globally to alleviate specific humanitarian crisis and to provide relief for victims. Donations are made into bank accounts run by the organization. This information system will focus on how donations towards these humanitarian crises are received and managed.

Specifically, this IS will manage information about donations, donors, crises, and receiving bank accounts. This data will provide meaningful insights into the donations received, biodata of donors, humanitarian crises each donation goes to, staff in charge of coordinating efforts to tackle specific crises, bank accounts receiving donations and the staff in charge of managing accounts etc. This will help the organization to perform financial, administrative and operational analysis.

Business Rules

Specific business rules may be defined as follows:

- Each Donor MUST make one or more donations.
- Each Donor MAY donate towards ANY Crisis.
- Each Crisis may or may NOT receive donations.
- Each Crisis MUST have only one staff assigned as Crisis Coordinator.
- Each Crisis Coordinator MUST be a staff, but not all Staff are Crisis Coordinators.
- Each Crisis Coordinator may coordinate efforts towards tackling one or more Crises.
- Each Donation MUST go into only ONE Account, but it could be ANY of the available Accounts.
- Each Account may or may NOT receive donations.
- Each Account MUST have only one staff assigned as account manager.
- Each account manager MUST be a staff, but not all Staff are account managers.

- Each account manager may manage one or more bank accounts.

Information Needs

Several levels of information are required by internal and external stakeholders in the organization for several purposes.

Operational

- Donors need to know which crises they can donate to.
- Donors need information of the accounts they can donate into.
- The organization needs information on donations to each crisis it seeks to alleviate, in order to plan its activities.
- The organization needs biodata of donors to inform fundraising strategies i.e., to know which demographics to target generally, and/or according to each crisis.

Administrative

- The organization needs information of donors and their donations for reference purposes.
- The organization needs information of accounts and staff in charge of account management.
- The organization needs information of crises and staff in charge of crisis coordination.
- The organization needs information about staff and volunteers.

Accounting & Budgeting

- The organization needs information of funds received in each account for financial analysis and budgeting purposes.
- The organization needs information on disbursement of funds from these accounts to each crisis.

Data Modelling

This information system is designed to accommodate the information needs of all stakeholders of the organization by the provision of relevant data. In order to meet these needs, the relevant entities containing the required data attributes are modelled as:

1. Donation: this entity provides information about each donation the organization receives. Attributes include **donationID** (donation unique identifier/primary key), **donationDate** (date of donation), **donationAmount** (donation amount), **donorID** (donor unique identifier), **crisisID** (crisis unique identifier), **accountID** (account unique identifier).
2. Donor: This provides donor biodata. Attributes include **donorID** (donor unique identifier/primary key), **donorName** (donor's full name), **donorGender** (donor's gender), **donorLocation** (donor's country), **donorJobfield** (donor's jobfield).
3. Crisis: this entity provides information on the several humanitarian crises the organization aims to tackle or alleviate. Attributes include **crisisID** (crisis unique identifier), **crisis** (the specific crisis donations are made towards), **crisisLocation**(location of specific crisis), **crisisCoordinatorID** (unique identifier for staff in charge of managing and coordinating activities against each crisis).
4. Account: This entity provides information about bank accounts which donations can be made into. Attributes include **accountID** (the bank account the donation was sent into), **bankName** (name of the bank where account belongs), **accountName** (name which the account bears), **accountManagerID** (unique identifier for staff in charge of managing the account/account manager.
5. Staff: This entity provides information about members of staff and volunteers. Attributes include **staffID** (staff unique identifier), **staffName** (Name of staff), **staffRole** (the position the staff occupies within the organization), **staffSalary** (salary received by each staff).

ENTITY-RELATIONSHIP (ER) MODEL

An Entity-Relationship model is a diagram which explains the logical structure of a database. The ER model displays the relationship between entities in a database. The main components of an ER model are Entities, Relationships, and Attributes.

Entities & Attributes

Entities of the database have been defined as follows (with primary key attributes underlined):

1. Donation: **(<u>donationID</u>, donationDate, donationAmount, donorID, crisisID, accountID)**
    - *Foreign key(s): **donorID, crisisID, accountID***
2. Donor: (**<u>donorID</u>** , **donorName**, **donorGender**, **donorLocation**, **donorJobfield)**
3. Crisis: (**<u>crisisID</u>**, **crisis**, **crisisLocation**, **crisisCoordinatorID)**
    - *Foreign key(s): **crisisCoordinatorID***
4. Account: (**<u>accountID</u>**, **bankName**, **accountName, accountManagerID)**
    - *Foreign key(s): **accountManagerID***
5. Staff: **<u>staffID</u>**, **staffName**, **staffRole**, **staffSalary**

Relationships

These are associations between two or more entities of a database. The following relationships are defined for this database:

a. Donor and Donation
    - *Relationship*: Donor (1 .. 1) **makes** Donation (1 ..*)

      Each Donor makes one or more Donations. Each Donation comes from one and only Donor.
    - *Cardinality*: One-to-Many (1:N)
    - *Participation*: Source & Target Mandatory

b. Donation and Account
    - *Relationship*: Account (0..*) **is credited with** Donation (1 …1)

      Each Donation is deposited into one and only one Account. Each Account receives zero to many Donations.
    - *Cardinality*: One-to-many (1:N)
    - *Participation*: Source Optional, Target Mandatory

c. Donation and Crisis

- Relationship: Crisis (0..*) **receives** (1…1) Donation (1 .. 1)

  Each Donation is made towards tackling a Crisis. Each Crisis receives zero to many Donations.

- *Cardinality*: One-to-many (1:N)
- *Participation*: Source Optional, Target Mandatory

d. Staff and Account

- Relationship: Staff (1 .. 1) **manages** (0…*) Account

  Each Account is managed by a Staff. Each staff manages zero to many Accounts.

- *Cardinality*: One-to-many (1:N)
- *Participation*: Source Optional, Target Mandatory

e. Staff and Crisis

- Relationship: Staff (1..1) **coordinates** (0…*) Crisis.

  Each Crisis is coordinated by a Staff. Each staff coordinates zero to many Accounts.

- *Cardinality*: One-to-many (1:N)
- *Participation*: Source Optional, Target Mandatory

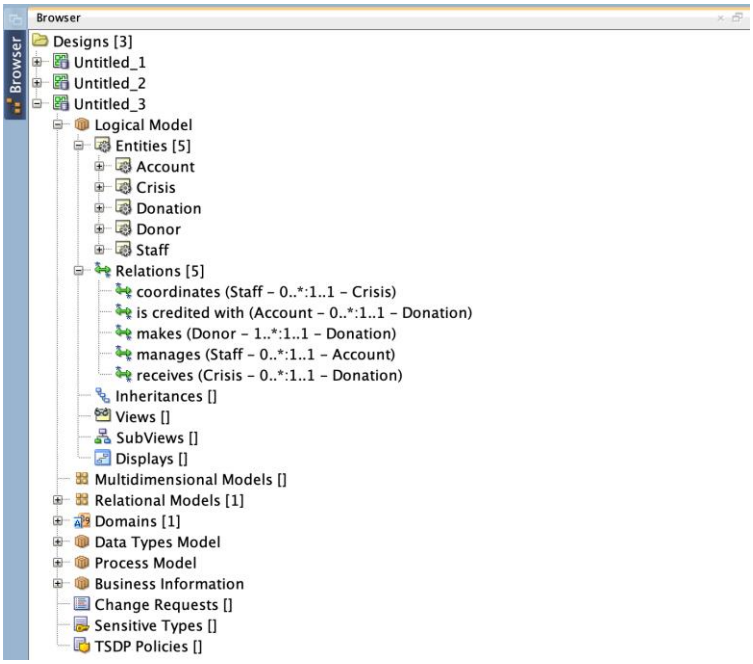

Figure 1: Entity relationship definitions showing multiplicity constraints

Logical Model

A logical model is built using Oracle SQL Data Modeller by defining entities, attributes, and the relationship between attributes. Figure 1 shows this step and Figure 2 shows logical model.

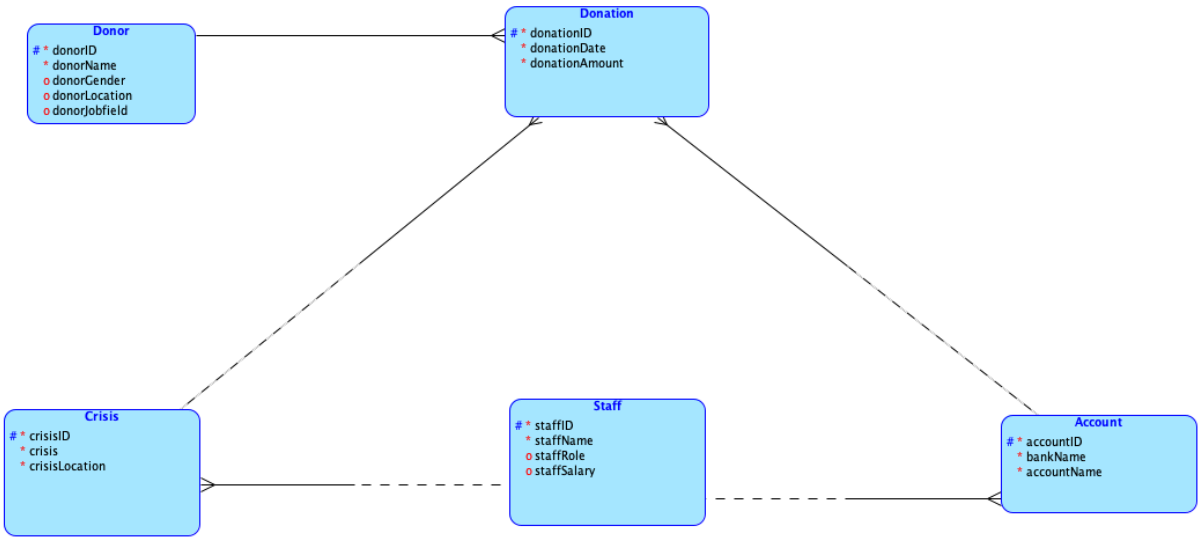

Figure 2: Logical ER Model

Relational Model

Oracle SQL Data Modeller automatically engineers a relational model based on relations defined in logical model. Figure 3 shows relational model.

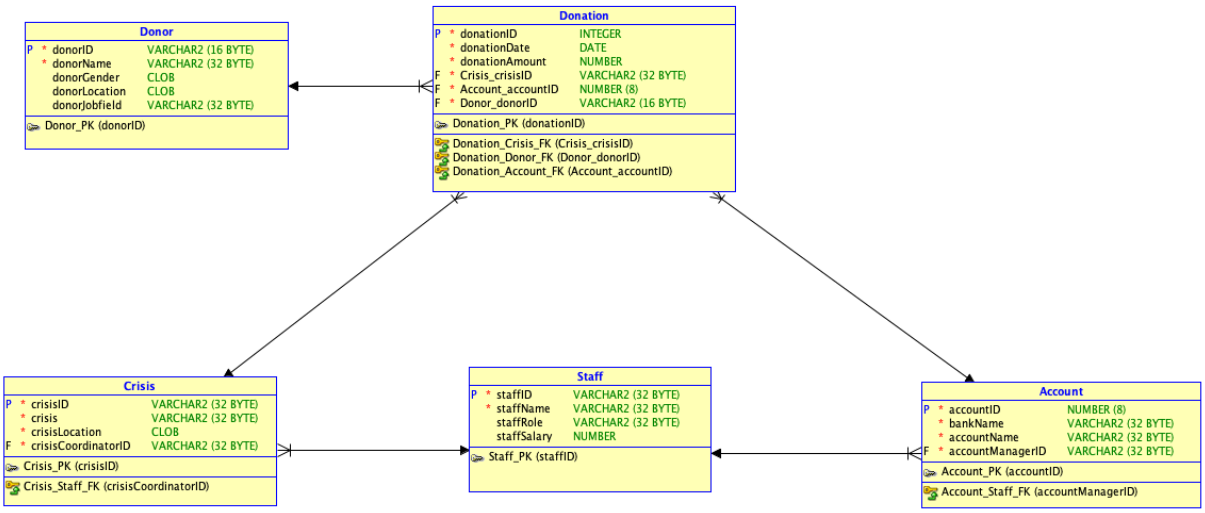

Figure 3: Relational ER Model

NORMALIZATION

Normalization is a technique for producing a set of relations with desirable properties, given the data requirements of an organization. Normalization removes data redundancy, and eliminates Insert, Update, and Deletion Anomalies.

Figure 4 shows a dummy dataset for Donor donations in Unnormalized form (UNF). I would attempt to normalize this relation to 3NF.

Normalization Process

The following describe steps taken to normalize my dataset from UNF to 3NF in relation to *Aid for All*'s information system:

1. Unnormalized Form (UNF): This means that our table contains repeating groups i.e., there is more than one value at the intersection of each row and column. Figure 4 below shows that my dummy data is currently unnormalized because it contains repeating groups for a single occurrence of the nominated key attribute (**donorID**) for that table.To put this in context, donors with more than one donation have

repeating groups for each value of their donation attributes (**donationID, donationDate, donationAmount, crisisID, crisis, crisisLocation, crisisCoordinatorID, crisisCoordinatorName, accountID, bankName, accountName, accountManagerID, accountManagerName)**. I eliminate this by duplicating values for donor information for each repeating group. The ensuing data table is shown in Figure 5. Upon further analysis of the ensuing relation, it would make more logical sense to identify **donationID** as the key attribute because it fully functionally determines every attribute in the relation. I will call this relation (entity) **Donation**.

donorID	donationID	donationDate	donationAmount	donorName	donorGender	donorLocation	donorJobfield	crisisID	crisis	crisisLocation	crisisCoordinatorID	crisisCoordinatorName	accountID	bankName	accountName	accountManagerID	accountManagerName
A	1	2022-01-20	£ 1,000.00	Ingo Fernandez	Male	United Kingdom	Education	C1	Natural Disaster	Turkey	S8	Adamu Garba	81264975	HSBC UK	Aid for All Crisis Response	S1	John Doe
	6	2022-01-25	£ 50,000.00					C15	War	Syria	S10	Peter Joseph	50041005	CIBC	Rapid Response AFA International	S6	Amos Leye
B	2	2022-01-21	£ 2,000.00	Vinita Bach	Female	India	Engineering	C2	Food Insecurity	Congo	S8	Adamu Garba	50041005	CIBC	Rapid Response AFA International	S6	Amos Leye
	7	2022-01-26	£ 300,000.00					C5	HIV	South Africa	S13	Jonathan Vincent	10091010	HSBC UK	AFA International Org.	S7	Helen Adebola
C	3	2022-01-22	£ 2,599.00	Ayo James	Female	Nigeria	IT	C7	Cancer	USA	S13	Jonathan Vincent	20011002	Revolut	AFA Crisis Donations	S5	Jack Reacher
	12	2022-01-31	£ 98,435.99										11101011	Bank of America	Aid for All Organization	S3	Barack Obama
D	4	2022-01-23	£ 360.00	Kunle John	Female	Nigeria	Healthcare	C11	Homelessness	Afghanistan	S16	Peter Obi	30021003	Barclays	AFAResponse	S4	Emily Rodden
E	5	2022-01-24	£ 1,200.00	titi grace	Male	Zimbabwe	IT	C1	Natural Disaster	Turkey	S8	Adamu Garba	40031004	Barclays	Aid for All Crisis Response	S5	Jack Reacher
H	8	2022-01-27	£ 15.00	michael jackson	Male	Poland	Sales	C4	Natural Disaster	Haiti	S9	Paul Blake	70061007	Wells Fargo	AFA International	S2	Mai Atafo
I	9	2022-01-28	£ 600.00	Fela Kuti	Male	Australia	Human Resources	C3	War	Ukraine	S8	Adamu Garba	80071008	First bank	AFA International Donations	S3	Barack Obama
J	10	2022-01-29	£ 364.55	Bob Marley	Female	New Zealand	Business	C99	Food Insecurity	Germany	S15	James St Patrick	90081009	Gtbank	Aid for All humanitarian organization	S1	John Doe
K	11	2022-01-30	£ 1,005.78	josiah Ayo	Female	Canada	Healthcare	C9	Homelessness	United Kingdom	S16	Peter Obi	10091010	HSBC UK	AFA International Org.	S7	Helen Adebola
M	13	2022-02-01	£ 5,231.00	banji samuel	Male	USA	Business	C5	HIV	South Africa	S13	Jonathan Vincent	12111112	Bank of Montreal	AFA donations	S3	Barack Obama
N	14	2022-02-02	£ 90.00	lemy emmanuel	Female	USA	IT	C15	War	Syria	S10	Peter Joseph	50041005	CIBC	Rapid Response AFA International	S6	Amos Leye

Figure 4: UNF Donor donations data table

2. First Normal Form (1NF): This means that the table does not repeating groups. This is demonstrated in Figure 5 below. The relation, **Donation**, contains data describing each **donationID**, the date and amount of donation, name, gender, location and jobfield of the donor, the crisis the donation is meant for, including details of the crisis, location, and crisis coordinator. It also contains data describing the account which the donation was deposited, the name of the bank, the account name, and name and ID of the staff managing the account.

donationID	donationDate	donationAmount	donorID	donorName	donorGender	donorLocation	donorJobfield	crisisID	crisis	crisisLocation	crisisCoordinatorID	risisCoordinatorNam	accountID	bankName	accountName	accountManagerID	accountManagerName
1	2022-01-20	£ 1,000.00	A	Ingo Fernandez	Male	United Kingdom	Education	C1	Natural Disaster	Turkey	S8	Adamu Garba	81264975	HSBC UK	Aid for All Crisis Response	S1	John Doe
2	2022-01-21	£ 2,000.00	B	Vinita Bach	Female	India	Engineering	C2	Food Insecurity	Congo	S8	Adamu Garba	50041005	CIBC	Rapid Response AFA International	S6	Amos Leye
3	2022-01-22	£ 2,599.00	C	Ayo James	Female	Nigeria	IT	C7	Cancer	USA	S13	Jonathan Vincent	20011002	Revolut	AFA Crisis Donations	S5	Jack Reacher
4	2022-01-23	£ 360.00	D	Kunle John	Female	Nigeria	Healthcare	C11	Homelessness	Afghanistan	S16	Peter Obi	30021003	Barclays	AFAResponse	S4	Emily Rodden
5	2022-01-24	£ 1,200.00	E	titi grace	Male	Zimbabwe	IT	C1	Natural Disaster	Turkey	S8	Adamu Garba	40031004	Barclays	Aid for All Crisis Response	S5	Jack Reacher
6	2022-01-25	£ 50,000.00	A	Ingo Fernandez	Male	United Kingdom	Education	C15	War	Syria	S10	Peter Joseph	50041005	CIBC	Rapid Response AFA International	S6	Amos Leye
7	2022-01-26	£ 300,000.00	B	Vinita Bach	Female	India	Engineering	C5	HIV	South Africa	S13	Jonathan Vincent	10091010	HSBC UK	AFA International Org.	S7	Helen Adebola
8	2022-01-27	£ 15.00	H	michael jackson	Male	Poland	Sales	C4	Natural Disaster	Haiti	S9	Paul Blake	70061007	Wells Fargo	AFA International	S2	Mai Atafo
9	2022-01-28	£ 600.00	I	Fela Kuti	Male	Australia	Human Resources	C3	War	Ukraine	S8	Adamu Garba	80071008	First bank	AFA International Donations	S3	Barack Obama
10	2022-01-29	£ 364.55	J	Bob Marley	Female	New Zealand	Business	C99	Food Insecurity	Germany	S15	James St Patrick	90081009	Gtbank	Aid for All humanitarian organization	S1	John Doe
11	2022-01-30	£ 1,005.78	K	josiah Ayo	Female	Canada	Healthcare	C9	Homelessness	United Kingdom	S16	Peter Obi	10091010	HSBC UK	AFA International Org.	S7	Helen Adebola
12	2022-01-31	£ 98,435.99	C	Ayo James	Female	Nigeria	IT	C7	Cancer	USA	S13	Jonathan Vincent	11101011	Bank of America	Aid for All Organization	S3	Barack Obama
13	2022-02-01	£ 5,231.00	M	banji samuel	Male	USA	Business	C5	HIV	South Africa	S13	Jonathan Vincent	12111112	Bank of Montreal	AFA donations	S3	Barack Obama
14	2022-02-02	£ 90.00	N	lemy emmanuel	Female	USA	IT	C15	War	Syria	S10	Peter Joseph	50041005	CIBC	Rapid Response AFA International	S6	Amos Leye

Figure 5: 2NF Donation data table

3. Second Normal Form (2NF): For a table to be in 2NF, the table must be in 1NF and contain no partial dependencies. Partial dependencies can only exist in a relation with a composite primary key i.e., a primary key with a combination of more than one key attribute. This means that any relation having a primary key of a single attribute automatically gets to its 2NF. To identify partial dependencies, I must first define functional dependency for the **Donation** relation**.**

Functional dependency: **donationID** $\Longrightarrow$ **donationDate, donationAmount, donorID, donorGender, donorLocation, donorJobfield, crisisID, crisis, crisisLocation, crisisCoordinatorID, crisisCoordinatorName, accountID, bankName, accountName, accountManagerID, accountManagerName**

The primary key for the table is **donationID**, as it fully functionally determines every other attribute. There is a one-to-one relationship between **donationID**, and attributes on the right-hand side of the functional dependency. This holds for all time and the determinant has the minimal number of attributes necessary to maintain full functional dependency with the attributes on the right side.
Hence, my **Donation** relation is already in 2NF because there are no partial dependencies.

4. Third Normal Form (3NF): A relation is said to be in 3NF when it is in 2NF and no transitive dependency exists for non-prime attributes. Transitive dependency occurs when some non-prime attribute determines some other attribute. Transitive dependencies which exist in the 2NF **Donation** relation are as follows:

TD1: **donorID** ⟹ **donorName, donorGender,donorLocation, donorJobfield**

TD2: **crisisID** ⟹ **crisis, crisisLocation, crisisCoordinatorID, crisisCoordinatorName**

TD3: **accountID** ⟹ **bankName, accountName, accountManagerID, accountManagerName**

To eliminate these transitive dependencies from the **Donation** relation, transitively dependent attributes must be removed from the relation. This is done by creating new relations for these attributes along with the determinant attributes. The determinant attributes then become foreign keys in the **Donation** relation. TD1, TD2 & TD3 would be used to create **Donor**, **Crisis**, and **Account** entities respectively.

It is observed that TD2 and TD3 contain transitive dependencies:

TD4: **crisisCoordinatorID** ⟹ **crisisCoordinatorName**

TD5: **accountManagerID** ⟹ **accountMangerName**

This means that **Crisis** and **Account** relations are in 2NF. To bring to 3NF, I eliminate these transitive dependencies, then create a **Staff** relation since crisis coordinator and account manager are members of staff of the organization. Upon doing this, the database and its individual relations are in 3NF as shown in the figures below.

donationID	donationDate	donationAmount	donorID	crisisID	accountID
1	2022-01-20	£ 1,000.00	A	C1	81264975
2	2022-01-21	£ 2,000.00	B	C2	50041005
3	2022-01-22	£ 2,599.00	C	C7	20011002
4	2022-01-23	£ 360.00	D	C11	30021003
5	2022-01-24	£ 1,200.00	E	C1	40031004
6	2022-01-25	£ 50,000.00	A	C15	50041005
7	2022-01-26	£ 300,000.00	B	C5	10091010
8	2022-01-27	£ 15.00	H	C4	70061007
9	2022-01-28	£ 600.00	I	C3	80071008
10	2022-01-29	£ 364.55	J	C99	90081009
11	2022-01-30	£ 1,005.78	K	C9	10091010
12	2022-01-31	£ 98,435.99	C	C7	11101011
13	2022-02-01	£ 5,231.00	M	C5	12111112
14	2022-02-02	£ 90.00	N	C15	50041005

Figure 6:3NF Donation table

donorID	donorName	donorGender	donorLocation	donorJobfield
A	Ingo Fernandez	Male	United Kingdom	Education
B	Vinita Bach	Female	India	Engineering
C	Ayo James	Female	Nigeria	IT
D	Kunle John	Female	Nigeria	Healthcare
E	titi grace	Male	Zimbabwe	IT
H	michael jackson	Male	Poland	Sales
I	Fela Kuti	Male	Australia	Human Resources
J	Bob Marley	Female	New Zealand	Business
K	josiah Ayo	Female	Canada	Healthcare
M	banji samuel	Male	USA	Business
N	lemy emmanuel	Female	USA	IT

Figure 7: 3NF Donor table

crisisID	crisis	crisisLocation	crisisCoordinatorID
C1	Natural Disaster	Turkey	S8
C2	Food Insecurity	Congo	S8
C7	Cancer	USA	S13
C11	Homelessness	Afghanistan	S16
C15	War	Syria	S10
C5	HIV	South Africa	S13
C4	Natural Disaster	Haiti	S9
C3	War	Ukraine	S8
C99	Food Insecurity	Germany	S15
C9	Homelessness	United Kingdom	S16

Figure 8: 3NF Crisis table

account_ID	bank_name	account_name	accountManagerID
81264975	HSBC UK	Aid for All Crisis Response	S1
10001001	Lloyds	Aid for All International	S4
20011002	Revolut	AFA Crisis Donations	S5
30021003	Barclays	AFAResponse	S4
40031004	Barclays	Aid for All Crisis Response	S5
50041005	CIBC	Rapid Response AFA International	S6
60051006	Wells Fargo	Aid for All	S2
70061007	Wells Fargo	AFA International	S2
80071008	First bank	AFA International Donations	S3
90081009	Gtbank	Aid for All humanitarian organization	S1
10091010	HSBC UK	AFA International Org.	S7
11101011	Bank of America	Aid for All Organization	S3
12111112	Bank of Montreal	AFA donations	S3
13121113	Lloyds	Aid for All crisis Organization	S4

Figure 9: 3NF Account table

staffID	staffName	staffRole	staffSalary
S1	John Doe	Account Manager	5000
S41	Beyonce Knowles	Cleaner	1000
S10	Peter Joseph	Crisis Coordinator	3500
S13	Jonathan Vincent	Crisis Coordinator	2900
S90	Burna Boy	Volunteer	0
S32	Lionel Messi	Volunteer	0
S15	James St Patrick	Crisis Coordinator	6800
S16	Peter Obi	Crisis Coordinator	700
S33	Ginger Baker	Volunteer	0
S2	Mai Atafo	Account Manager	5000
S3	Barack Obama	Account Manager	3400
S4	Emily Rodden	Account Manager	9000
S88	Ayokunle McMath	Analyst	1200
S5	Jack Reacher	Account Manager	4500
S6	Amos Leye	Account Manager	6500
S56	Enoch Adeboye	Analyst	9000
S7	Helen Adebola	Account Manager	8000
S8	Adamu Garba	Crisis Coordinator	750
S9	Paul Blake	Crisis Coordinator	4300

Figure 10: 3NF Staff table

Dummy Data shows the relations at each step of the normalization process (i.e., 3NF, 2NF, 1NF, UNF) for better readability.

Table Creation and Population

The database schema is created and populated using PostgreSQL (PgAdmin4) as shown in Figure 11 below. Five tables were created for Donor, Donation, Crisis, Staff, and Account. The SQL statements used in creating, altering, and populating these tables are shown in the following sections.

Query | Query History

```
1  -- SCHEMA: humanitarianOrg
2
3  -- DROP SCHEMA IF EXISTS "humanitarianOrg" ;
4
5  CREATE SCHEMA IF NOT EXISTS "humanitarianOrg"
6      AUTHORIZATION postgres;
```

Figure 11: CREATE SCHEMA statement.

Creating tables

The account, donor, donation, crisis, and staff tables respectively are created using CREATE TABLE statements, as demonstrated in the figures below.

```
CREATE TABLE IF NOT EXISTS "humanitarianOrg".account (
    accountid        numeric(8) NOT NULL,
    bankname         character varying NOT NULL,
    accountname      character varying NOT NULL,
    accountmanagerid character varying NOT NULL
);
```

Figure 12: CREATE TABLE account statement.

```
CREATE TABLE IF NOT EXISTS "humanitarianOrg".donor (
    donorid       character varying NOT NULL,
    donorname     character varying NOT NULL,
    donorgender   TEXT,
    donorlocation character varying,
    donorjobfield character varying
);
```

Figure 13: CREATE TABLE donor statement.

```
CREATE TABLE IF NOT EXISTS "humanitarianOrg".donation (
    donationid        INTEGER NOT NULL,
    donationdate      DATE NOT NULL,
    donationamount    NUMERIC NOT NULL,
    crisis_crisisid   character varying NOT NULL,
    account_accountid NUMERIC(8) NOT NULL,
    donor_donorid     character varying NOT NULL
);
```

Figure 14: CREATE TABLE donation statement.

```
CREATE TABLE IF NOT EXISTS "humanitarianOrg".crisis (
    crisisid            character varying NOT NULL,
    crisis              character varying NOT NULL,
    crisislocation      character varying NOT NULL,
    crisiscoordinatorid character varying NOT NULL
);
```

Figure 15: CREATE TABLE crisis statement.

```
CREATE TABLE IF NOT EXISTS "humanitarianOrg".staff (
    staffid     character varying NOT NULL,
    staffname   character varying NOT NULL,
    staffrole   character varying,
    staffsalary NUMERIC
);
```

Figure 16: CREATE TABLE staff statement.

Populating tables

Mock values are inserted into the account, donation, donor, crisis, and staff tables respectively using INSERT statements, as demonstrated in the figures below.

Query | Query History

```
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (81264975, 'HSBC UK', 'Aid for All Crisis Response', 'S1');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (10001001, 'Lloyds', 'Aid for All International', 'S4');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (20011002, 'Revolut', 'AFA Crisis Donations', 'S5');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (30021003, 'Barclays', 'AFAResponse', 'S4');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (40031004, 'Barclays', 'Aid for All Crisis Response', 'S5');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (50041005, 'CIBC', 'Rapid Response AFA International', 'S6');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (60051006, 'Wells Fargo', 'Aid for All', 'S2');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (70061007, 'Wells Fargo', 'AFA International ', 'S2');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (80071008, 'First bank', 'AFA International Donations', 'S3');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (90081009, 'Gtbank', 'Aid for All  humanitarian organization', 'S1');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (10091010, 'HSBC UK', 'AFA International Org.', 'S7');
INSERT INTO "humanitarianOrg".account (
    accountid, bankname, accountname, accountmanagerid)
    VALUES (11101011, 'Bank of America', 'Aid for All Organization', 'S3');
INSERT INTO "humanitarianOrg".account (
```

Figure 17: INSERT INTO account.

Query Query History

```
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (1, '2022-01-20', 1000, 'A', 'C1', 81264975);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (2, '2022-01-21', 2000, 'B', 'C2', 50041005);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (3, '2022-01-22', 2599, 'C', 'C7', 20011002);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (4, '2022-01-23', 360, 'D', 'C11', 30021003);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (5, '2022-01-24', 1200, 'E', 'C1', 40031004);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (6, '2022-01-25', 50000, 'A', 'C15', 50041005);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (7, '2022-01-26', 300000, 'B', 'C5', 10091010);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (8, '2022-01-27', 15, 'H', 'C4', 70061007);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (9, '2022-01-28', 600, 'I', 'C3', 80071008);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (10, '2022-01-29', 365, 'J', 'C99', 90081009);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (11, '2022-01-30', 1006, 'K', 'C9', 10091010);
INSERT INTO "humanitarianOrg".donation (
    donationid, donationdate, donationamount, donorid, crisisid, accountid)
    VALUES (12, '2022-01-31', 98436, 'C', 'C7', 11101011);
INSERT INTO "humanitarianOrg".donation (
```

Figure 18: INSERT INTO donation.

Query Query History

```
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('A', 'Ingo Fernandez', 'Male', 'United Kingdom', 'Education');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('B','Vinita Bach','Female','India','Engineering');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('C','Ayo James','Female','Nigeria','IT');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('D','Kunle John','Female','Nigeria','Healthcare');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('E','titi grace','Male','Zimbabwe','IT');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('H','michael jackson','Male','Poland','Sales');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('I','Fela Kuti','Male','Australia','Human Resources');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('J','Bob Marley','Female','New Zealand','Business');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('K','josiah Ayo','Female','Canada','Healthcare');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('M','banji samuel','Male','USA','Business');
INSERT INTO "humanitarianOrg".donor(
    donorid, donorname, donorgender, donorlocation, donorjobfield)
    VALUES ('N','lemy emmanuel','Female','USA','IT');


```

Figure 19: INSERT INTO donor.

Query Query History

```
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C1', 'Natural Disaster', 'Turkey', 'S8');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C2', 'Food Insecurity', 'Congo', 'S8');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C7', 'Cancer', 'USA', 'S13');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C11', 'Homelessness', 'Afghanistan', 'S16');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C15', 'War', 'Syria', 'S10');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C5', 'HIV', 'South Africa', 'S13');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C4', 'Natural Disaster', 'Haiti', 'S9');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C3', 'War', 'Ukraine', 'S8');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C99', 'Food Insecurity', 'Germany', 'S15');
INSERT INTO "humanitarianOrg".crisis (
    crisisid, crisis, crisislocation, crisiscoordinatorid)
    VALUES ('C9', 'Homelessness', 'United Kingdom', 'S16');
```

Figure 20: INSERT INTO crisis.

Query Query History

```
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S1', 'John Doe', 'Account Manager', 5000);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S41', 'Beyonce Knowles', 'Cleaner', 1000);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S10', 'Peter Joseph', 'Crisis Coordinator', 3500);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S13', 'Jonathan Vincent', 'Crisis Coordinator', 2900);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S90', 'Burna Boy', 'Volunteer', 0);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S32', 'Lionel Messi', 'Volunteer', 0);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S15', 'James St Patrick', 'Crisis Coordinator', 6800);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S16', 'Peter Obi', 'Crisis Coordinator', 700);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S33', 'Ginger Baker', 'Volunteer', 0);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S2', 'Mai Atafo', 'Account Manager', 5000);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S3', 'Barack Obama', 'Account Manager', 3400);
INSERT INTO "humanitarianOrg".staff (
    staffid, staffname, staffrole, staffsalary)
    VALUES ('S4', 'Emily Rodden', 'Account Manager', 9000);
INSERT INTO "humanitarianOrg".staff (
```

Figure 21: INSERT INTO staff.

Querying Database

Query 1: The query below selects total donation amount of each donor.

Query | Query History

```
1  -- Query 1: This query selects total donation amount of each donor in descending order.
2  select
3    "humanitarianOrg".donation.donorid,
4    "humanitarianOrg".donor.donorname,
5    SUM("humanitarianOrg".donation.donationamount)
6  from "humanitarianOrg".donation
7  left join "humanitarianOrg".donor
8  on "humanitarianOrg".donor.donorid = "humanitarianOrg".donation.donorid
9  group by 1,2
10 order by 3 desc
11 ;
```

Data Output | Messages | Notifications

	donorid character varying	donorname character varying	sum numeric
1	B	Vinita Bach	302000
2	C	Ayo James	101035
3	A	Ingo Fernandez	51000
4	M	banji samuel	5231
5	E	titi grace	1200
6	K	josiah Ayo	1006
7	I	Fela Kuti	600
8	J	Bob Marley	365
9	D	Kunle John	360
10	N	lemy emmanuel	90
11	H	michael jackson	15

Figure 22: Donation amount by Donor

Query 2: The query below selects total donation amount received by each crisis.

Query | Query History

```
1  --Query 2: This query selects total donation amount received by each crisis.
2  select
3    "humanitarianOrg".crisis.crisisid,
4    "humanitarianOrg".crisis.crisis,
5    SUM("humanitarianOrg".donation.donationamount)
6  from "humanitarianOrg".crisis
7  left join "humanitarianOrg".donation
8  on "humanitarianOrg".crisis.crisisid = "humanitarianOrg".donation.crisisid
9  group by 1,2
10 order by 3 desc
11 ;
```

Data Output | Messages | Notifications

	crisisid [PK] character varying	crisis character varying	sum numeric
1	C5	HIV	305231
2	C7	Cancer	101035
3	C15	War	50090
4	C1	Natural Disaster	2200
5	C2	Food Insecurity	2000
6	C9	Homelessness	1006
7	C3	War	600
8	C99	Food Insecurity	365
9	C11	Homelessness	360
10	C4	Natural Disaster	15

Figure 23: Donation amount by Crisis

Query 3: The query below selects each account and number of donations received.

Query | Query History

```
1  -- Query 3: The query below selects each bank account and number of donations received.
2  select
3    "humanitarianOrg".account.accountid,
4    "humanitarianOrg".account.bankname,
5    COUNT("humanitarianOrg".donation.donationamount)
6  from "humanitarianOrg".account
7  left join "humanitarianOrg".donation
8    on "humanitarianOrg".account.accountid = "humanitarianOrg".donation.accountid
9  group by 1,2
10 order by 3 desc
11 ;
```

Data Output | Messages | Notifications

	accountid [PK] numeric (8)	bankname character varying	count bigint
1	50041005	CIBC	3
2	10091010	HSBC UK	2
3	40031004	Barclays	1
4	90081009	Gtbank	1
5	20011002	Revolut	1
6	12111112	Bank of Montreal	1
7	70061007	Wells Fargo	1
8	81264975	HSBC UK	1
9	30021003	Barclays	1
10	11101011	Bank of America	1
11	80071008	First bank	1
12	10001001	Lloyds	0
13	13121113	Lloyds	0
14	60051006	Wells Fargo	0

Figure 24: Count of donations received by each account.

Query 4: The query below selects total salary grouped by staff role.

Query | Query History

```
1  -- Query 4: The query below selects total salary grouped by staff role.
2  select
3    "humanitarianOrg".staff.staffrole,
4    SUM("humanitarianOrg".staff.staffsalary)
5  from "humanitarianOrg".staff
6  group by 1
7  order by 2 desc
8  ;
```

Data Output | Messages | Notifications

	staffrole character varying	sum numeric
1	Account Manager	41400
2	Crisis Coordinator	18950
3	Analyst	10200
4	Cleaner	1000
5	Volunteer	0

Figure 25: Salary by staff role query.

Query 5: The query below shows total donations received.

Query | Query History

```
1  -- Query 5: The query below shows total donations received.
2  select
3    SUM(donationamount)
4  from "humanitarianOrg".donation
5  ;
6
```

Data Output | Messages | Notifications

	sum numeric
1	462902

Figure 26: Total donation amount received query.

Security, Integrity, and Ethical Aspects of Data Governance

Security

Data security is critical for humanitarian organizations, which may deal with sensitive personal information such as personal identification information, and data related to communities in conflict or disaster situations. The organization should have a clear policy regarding data access and should ensure that staff members are trained in secure data handling practices.

The following are some ways to ensure data security in a humanitarian organization's database:

- **Use strong passwords**: Require strong passwords that include a mix of letters, numbers, and special characters. Implement policies for password expiration and complexity.
- **Access control**: Limit access to sensitive data to authorized personnel only. Implement access control policies to ensure that only those who need to access specific data can do so.
- **Encryption**: Use encryption to protect sensitive data both in transit and at rest. This includes data in databases, backups, and archives.
- **Regular backups**: Regularly back up data to ensure that it can be restored in case of a security breach or system failure.

Here is an example of SQL code that can be used to set up access control policies for the donor table:

```
GRANT SELECT, INSERT, UPDATE, DELETE ON "humanitarianOrg".donor TO privileged_user;
```

Figure 27: GRANT statement for access control.

This code grants privileges to a user called privileged user, allowing them to select, insert, update, and delete data from the donor table.

Integrity

Data in a humanitarian organization's database should be reliable, and there should be no chance of losing it or altering it unintentionally.

Data integrity is made up of the following parts:

- **Entity integrity**

To identify each row in a table, the table must have a primary key. The primary key is a unique value that identifies each row. This requirement is called the entity integrity constraint.

```
ALTER TABLE "humanitarianOrg".account ADD CONSTRAINT account_pk PRIMARY KEY ( accountid );
```

Figure 28: account entity constraint

```
ALTER TABLE "humanitarianOrg".crisis ADD CONSTRAINT crisis_pk PRIMARY KEY ( crisisid );
```

Figure 29: crisis entity constraint

```
ALTER TABLE "humanitarianOrg".donation ADD CONSTRAINT donation_pk PRIMARY KEY ( donationid );
```

Figure 30: donation entity constraint

```
ALTER TABLE "humanitarianOrg".donor ADD CONSTRAINT donor_pk PRIMARY KEY ( donorid );
```

Figure 31:donor entity constraint

```
ALTER TABLE "humanitarianOrg".staff ADD CONSTRAINT staff_pk PRIMARY KEY ( staffid );
```

Figure 32: staff entity constraint

- **Semantic integrity**

Semantic integrity ensures that data entered into a row reflects an allowable value for that row. The value must be within the domain, or allowable set of values, for that column. For example, the ACCOUNTID column of the **account** table permits only 8-digit numbers. If a value outside the domain is entered into a column, the semantic integrity of the data is violated.

Data Output | Messages | Notifications

	table_name name	column_name name	is_nullable character varying (3)	data_type character varying	numeric_precision integer
1	account	accountid	NO	numeric	8
2	account	bankname	NO	character varying	[null]
3	account	accountname	NO	character varying	[null]
4	account	accountmanagerid	NO	character varying	[null]
5	crisis	crisis	NO	character varying	[null]
6	crisis	crisiscoordinatorid	NO	character varying	[null]
7	crisis	crisisid	NO	character varying	[null]
8	crisis	crisislocation	NO	character varying	[null]
9	donation	donationamount	NO	numeric	[null]
10	donation	donationid	NO	integer	32
11	donation	accountid	NO	numeric	8
12	donation	donationdate	NO	date	[null]
13	donation	donorid	NO	character varying	[null]
14	donation	crisisid	NO	character varying	[null]
15	donor	donorjobfield	YES	character varying	[null]
16	donor	donorlocation	YES	character varying	[null]
17	donor	donorid	NO	character varying	[null]
18	donor	donorname	NO	character varying	[null]
19	donor	donorgender	YES	text	[null]
20	staff	staffrole	YES	character varying	[null]
21	staff	staffid	NO	character varying	[null]
22	staff	staffsalary	YES	numeric	[null]
23	staff	staffname	NO	character varying	[null]

Figure 33: Semantic Integrity constraints

- **Referential integrity**

Referential integrity refers to the logical dependency of a foreign key on a primary key. The integrity of a row containing a foreign key depends on the integrity of the row it references i.e., the row containing the matching primary key.

```
ALTER TABLE "humanitarianOrg".account
    ADD CONSTRAINT account_staff_fk FOREIGN KEY ( accountmanagerid )
        REFERENCES "humanitarianOrg".staff ( staffid );

ALTER TABLE "humanitarianOrg".crisis
    ADD CONSTRAINT crisis_staff_fk FOREIGN KEY ( crisiscoordinatorid )
        REFERENCES "humanitarianOrg".staff ( staffid );

ALTER TABLE "humanitarianOrg".donation
    ADD CONSTRAINT donation_account_fk FOREIGN KEY ( account_accountid )
        REFERENCES "humanitarianOrg".account ( accountid );

ALTER TABLE "humanitarianOrg".donation
    ADD CONSTRAINT donation_crisis_fk FOREIGN KEY ( crisis_crisisid )
        REFERENCES "humanitarianOrg".crisis ( crisisid );

ALTER TABLE "humanitarianOrg".donation
    ADD CONSTRAINT donation_donor_fk FOREIGN KEY ( donor_donorid )
        REFERENCES "humanitarianOrg".donor ( donorid );
```

Figure 34: Referential Integrity constraints.

Data Ethics

organizations need to ensure that they are using data in ways that are consistent with their mission and values. Here are some ways to ensure ethical data use:

- **Privacy policies**: Create clear and concise privacy policies that explain how data is collected, stored, and used.
- **Data anonymization**: Remove personally identifiable information from datasets to protect the privacy of individuals.

- **Informed consent**: Obtain informed consent from individuals before collecting their data.
- **Data transparency**: Be transparent about how data is collected, used, and shared.

CONCLUSION

I successfully designed an information system for the humanitarian organization, *Aid for All*. I defined the scenario, defined the business rules, articulated information needs and conceptual model. I created an appropriate Entity-Relationship (ER) model with properly defined entity relationships. I also normalized the database with every entity in Third Normal Form (3NF).
I implemented the information system by creating the required entities, defined integrity constraints for each entity. I also populated each entity with mock values and queried the tables to gather insights.
In conclusion, I discussed data security, ethics, and integrity in the context of my scenario.

REFERENCES

Adrienne Watt (n.d.) Chapter 11 Functional Dependencies. Available at: https://opentextbc.ca/dbdesign01/chapter/chapter-11-functional-dependencies/ [Accessed 16 Feb 2023].

Adrienne Watt (n.d.) Chapter 8 The Entity Relationship Data Model. Available at: https://opentextbc.ca/dbdesign01/chapter/chapter-8-entity-relationship-model/ [Accessed 16 Feb 2023].

Amaresan, S. (14 Jan 2022). The Straightforward Guide to Data Governance (DG). Available at: https://blog.hubspot.com/website/data-governance . [Accessed 7 Mar 2023].

BYJU'S Exam Prep (n.d.). Second Normal Form in DBMS. Available at: https://byjus.com/gate/second-normal-form-in-dbms/ [Accessed 16 Feb 2023].

Connolly, T., & Begg, C. (2014). Database systems: A practical approach to design, implementation, and management (Global Edition). Pearson Education Limited.

Datanamic. (n.d.). Database Normalization. [online] Available at: https://www.datanamic.com/support/database-normalization.html [Accessed 13 Feb 2023].

IBM. (21 Oct 2022). Data Integrity. Available at: https://www.ibm.com/docs/en/informix-servers/14.10?topic=data-integrity . [Accessed 7 Mar 2023].

Michelle A. Poolet. (28 Feb 1999). SQL Design: Why You Need Database Normalization. [online] Available at: https://www.itprotoday.com/sql-server/sql-design-why-you-need-database-normalization [Accessed 13 Feb 2023].