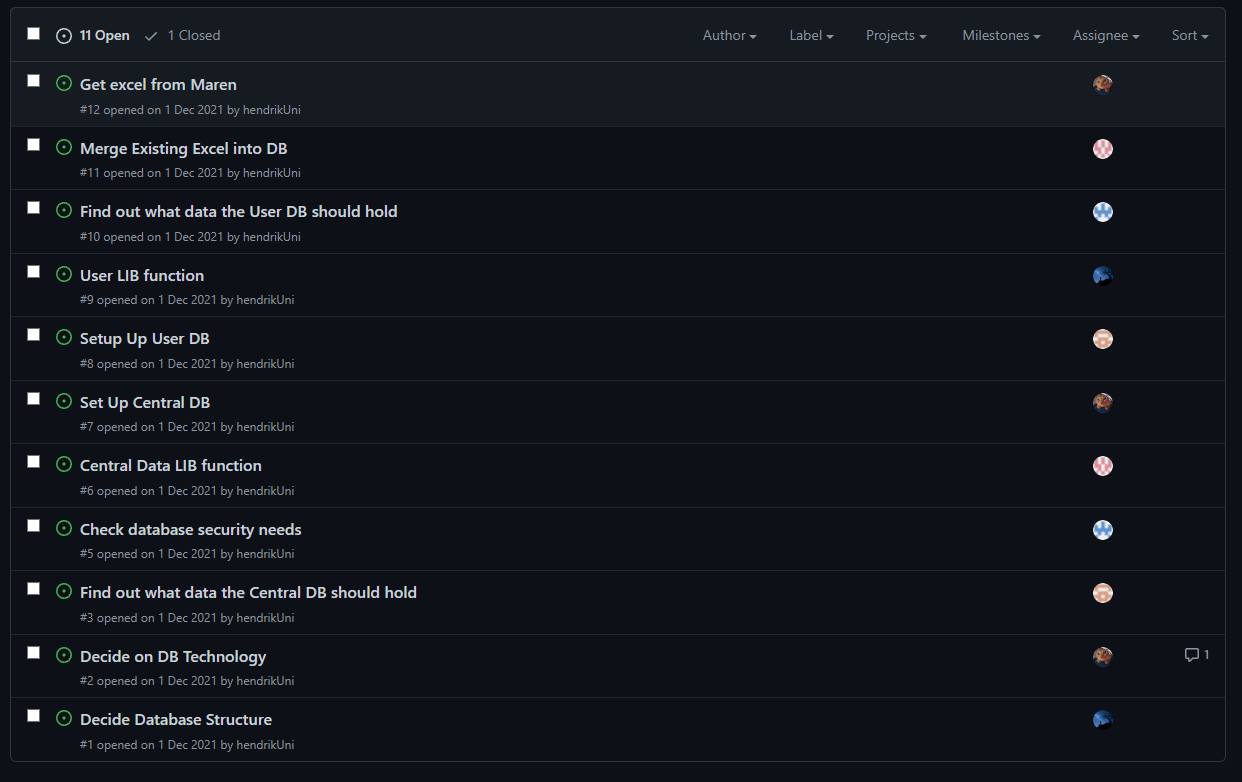
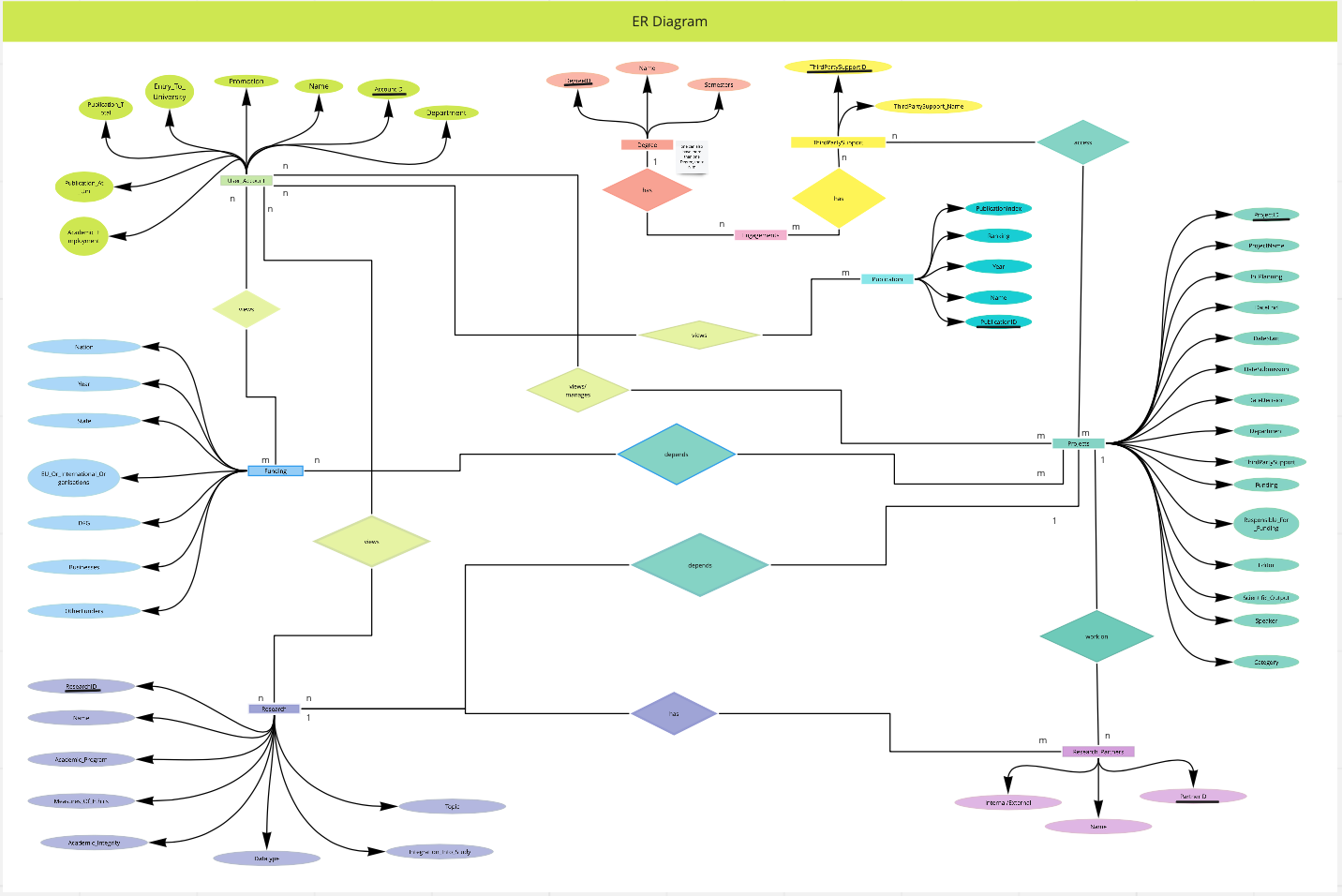
**SE2/2 Team Peacock – Database Design**

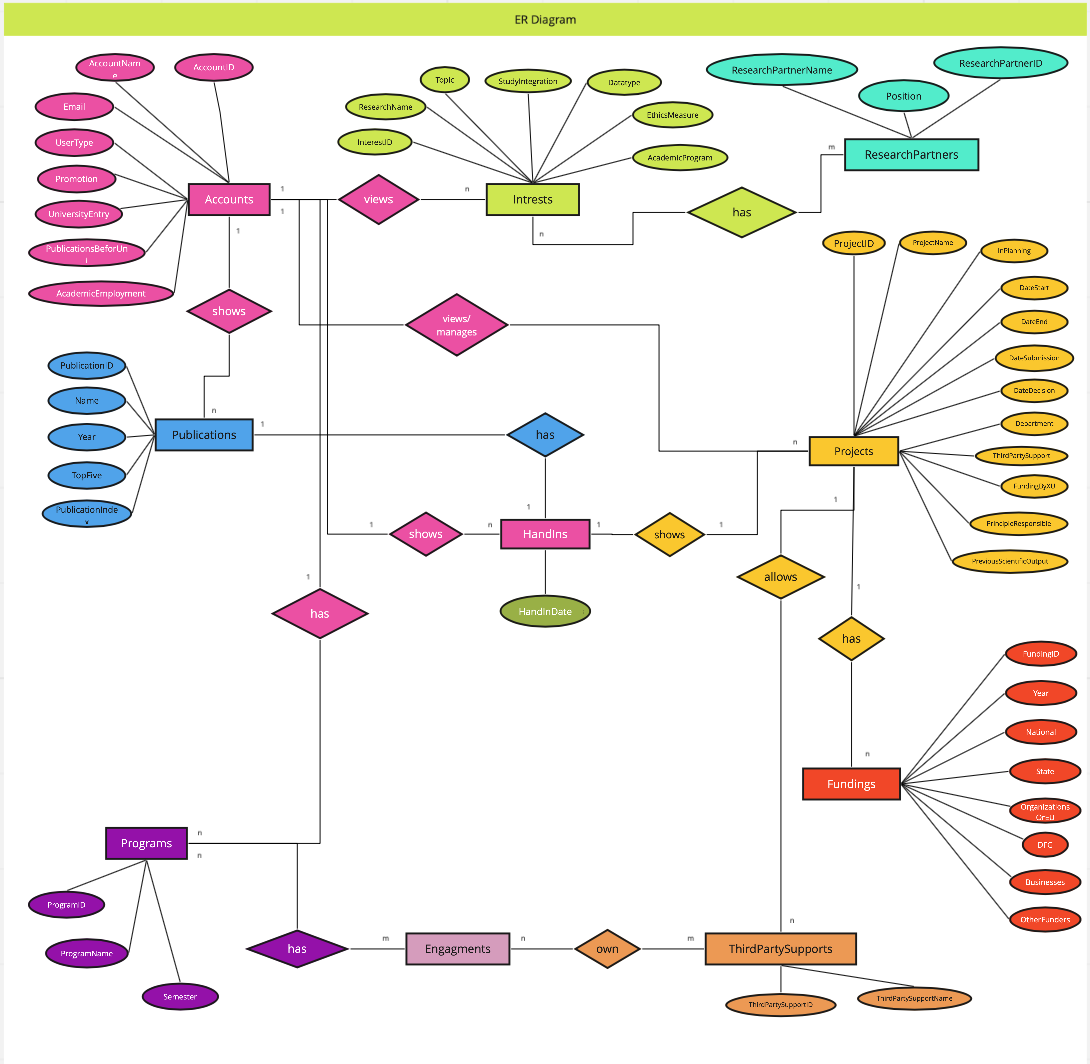
Hendrik Michels, Vaniya Pushparajah, Nick Papenheim, Long Ba Bao Nguyen, Thure Wiegel

Before we started to work, split the project into issues, that we put into the GitHub.  
Every Issue has comments, that explain the task and split them into smaller sub-issues. This makes communicating easier and the task clear for everybody. We created the issues as a whole group together and ordered them according to their importance.

  
We frequently communicated through WhatsApp and Teams. We held weekly Meetings to get updated about the work of the others, to synchronize work with the other Groups and to split the tasks that had to be done. Additionally, Hendrik, Nick and Thure met with Maren in the initial stages of the project to talk about her requirements and make sure we fully understood the data she needed. We made great use of the Teams collaboration features and were able to work on problems simultaneously.

To better visualize the database with all its components and to find issues and conflicts Vaniya created an ERD from a prototype database based on Maren's Excel. We then as a group ironed out potential issues and produced the final tables below.



Updated One:

<https://miro.com/welcomeonboard/am5PdEtyZHNxVEtkY1didzBnaUVSMk14bG1lQW5lS1RocGd2Q01uWGNkV09TRUNBdkR5c2RkOE1EUU55VUNRcXwzMDc0NDU3MzUzNDQ2MTg5MzQy?invite_link_id=600208049794>

|  |  |  |
| --- | --- | --- |
| Names | **Primary/Foreign Key** | **Datatype** |
| **User Account** |  |  |
| ID Account | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
| Department |  | VARCHAR(36) |
| Promotion |  | VARCHAR(100) |
| Entry to university |  | DATE |
| publications total |  | INT(100) |
| publications at uni (?) |  | VARCHAR(200) |
| Academic Employment |  | VARCHAR(100) NOT NULL |
|  |  |  |
| **Publications** |  |  |
|  |  |  |
| ID Account | FK | Not NULL INT |
| ID Publication | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
| Year |  | YEAR / NOW() |
| Ranking |  | ROW\_Number |
| Publication Index |  | INT |
|  |  |  |
| **Projects** |  |  |
|  |  |  |
| ID Project | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
| In Planning |  | BOOL |
| Date start |  | DateTime |
| Date end |  | DATEDIFF |
| Date Submission |  | DATE |
| Date Decision |  | DATE |
| Department |  | VARCHAR(100) |
| ThirdPartySupport |  | VARCHAR(100) |
| Funding |  | INT |
| responsible for funding |  | VARCHAR(100) |
| Editor |  | VARCHAR(100) |
| ID Partner | FK | Not NULL INT |
| Scientific output |  | VARCHAR(100) |
| Speaker |  | VARCHAR(100) |
| Category |  | VARCHAR(100) |
|  |  |  |
| **Degree** |  |  |
|  |  |  |
| ID Degree | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
| Semesters |  | VARCHAR(50) |
|  |  |  |
| **ThirdPartySupport** |  |  |
|  |  |  |
| ID | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
|  |  |  |
| **Engagements** |  |  |
|  |  |  |
| ID Account | FK | Not NULL INT |
| ID Degree | FK | Not NULL INT Autoinc. |
|  |  |  |
| **Research** |  |  |
|  |  |  |
| ID Research | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
| Topic |  | VARCHAR(100) |
| Integration into study |  | BOOL? |
| Datatype (qual, quant, data, mixed) |  | DOUBLE |
| Academic integrity |  | VARCHAR(100) |
| measures of ethics |  | VARCHAR(100) |
| academic program (?) |  | VARCHAR(50) |
|  |  |  |
| **Research partners** |  |  |
|  |  |  |
| ID Partner | PK | AUTO\_INCREMENT NOT NULL INT |
| Name |  | VARCHAR(100) NOT NULL |
| Internal/external |  | VARCHAR(3) |
|  |  |  |
| **Funding** |  |  |
|  |  |  |
| ID Account | FK | INT |
| ID Project | FK | INT |
| Year |  | DATE |
| Nation |  | DOUBLE |
| State |  | DOUBLE |
| EU or International Organizations |  | DOUBLE |
| DFG |  | DOUBLE |
| Businesses |  | DOUBLE |
| Other Funders |  | DOUBLE |

The other groups were given access to these tables and could request changes or even their own custom tables that fit their requirements.

In the end, we as a group turned the tables into a shareable SQL file that can be executed on the hosting server.

Updated SQL – Tables:

DROP SCHEMA CAST\_TEST;

CREATE SCHEMA CAST\_TEST;

USE CAST\_TEST;

#CREATE USER 'cast\_server'@'localhost' IDENTIFIED BY 'cast\_server1234';

#GRANT DELETE, INSERT, UPDATE, SELECT, EXECUTE ON CAST\_TEST . \* TO 'cast\_server'@'localhost';

CREATE TABLE Accounts (

AccountID int auto\_increment Not Null,

Name varchar(100) Not Null,

Email varchar(100),

UserType varchar(100),

Department varchar(36),

Promotion int,

UniversityEntry date,

PublicationsBeforeUni int,

AcademicEmployment varchar(100) Not Null,

PRIMARY KEY (AccountID)

);

CREATE TABLE Intrests (

AccountID int,

InterestID int AUTO\_INCREMENT NOT NULL,

ResearchName VARCHAR(100) NOT NULL,

Topic VARCHAR(100),

StudyIntegration VARCHAR(1000),

Datatype VARCHAR(1000),

EthicsMeasure VARCHAR(1000),

AcademicProgram VARCHAR(100),

PRIMARY KEY (InterestID),

FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID)

);

CREATE TABLE Projects (

AccountID int,

ProjectID int not null auto\_increment,

ProjectName varchar(100) not null,

InPlanning bool,

DateStart datetime,

DateEnd datetime,

DateSubmission date,

DateDecision date,

Department varchar(100),

ThirdPartySupport varchar(100),

FundingByXU int,

PrincipleResponsible varchar(100),

PreviousScientificOutput varchar(500),

PRIMARY KEY (ProjectID),

FOREIGN KEY (AccountID) REFERENCES Accounts(AccountID)

);

CREATE TABLE Fundings (

FundingID int,

ProjectID int,

Year Date,

National DOUBLE,

State DOUBLE,

OrganizationsOrEU DOUBLE,

DFG DOUBLE,

Buisnesses DOUBLE,

OtherFunders DOUBLE,

PRIMARY KEY (FundingID),

FOREIGN KEY (ProjectID) REFERENCES Projects(ProjectID)

);

CREATE TABLE Publications (

AccountID int Not Null,

PublicationID int auto\_increment Not Null,

Name varchar(100) Not Null,

Year int,

TopFive Boolean,

PublicationIndex varchar(128),

PRIMARY KEY (PublicationID),

FOREIGN KEY(AccountID) references Accounts(AccountID)

);

#For Interests

CREATE TABLE ResearchPartners (

ResearchPartnerID int AUTO\_INCREMENT NOT NULL,

ResearchPartnerName VARCHAR(100) NOT NULL,

Position VARCHAR(3), #int for internal, ext for external

PRIMARY KEY (ResearchPartnerID),

Intrest int,

FOREIGN KEY (Intrest) REFERENCES Intrests(InterestID)

);

CREATE TABLE Programs (

ProgramID int AUTO\_INCREMENT NOT NULL,

ProgramName VARCHAR(100) NOT NULL,

Semester VARCHAR(50),

PRIMARY KEY (ProgramID)

);

CREATE TABLE ThirdPartySupports (

ThirdPartySupportID int AUTO\_INCREMENT NOT NULL,

ThirdPartySupportName VARCHAR(200) NOT NULL,

PRIMARY KEY (ThirdPartySupportID)

);

CREATE TABLE Engagements (

ProgramID int,

ThirdPartySupportID int,

FOREIGN KEY (ProgramID) REFERENCES Programs(ProgramID),

FOREIGN KEY (ThirdPartySupportID) REFERENCES ThirdPartySupports(ThirdPartySupportID)

);

CREATE TABLE HandIns (

HandInID int not null AUTO\_INCREMENT,

AccountID int,

Project int,

Publication int,

HandInDate Date,

Primary Key (HandInID),

Foreign Key (AccountID) REFERENCES Accounts(AccountID),

Foreign Key (Project) REFERENCES Projects(ProjectID),

Foreign Key (Publication) REFERENCES Publications(PublicationID)

);