Jabili Gadde, Nicole Kercado, Lakaylah Gage, Anthony Scarpa
CSC 315-01
Stage 4: Design
Link to Project Repository: https://github.com/TCNJ-degoodj/stage-iv-group-2

Step 2:

SPEAKER is in BCNF. The table is in 1NF as each of the values are atomic. Each of the non-prime attributes (Name, Gender, Nationality, Is_jewish) are fully functionally dependent on the primary key (Speaker_number), meaning that the table is in 2NF. In addition, none of the non-prime attributes are transitively dependent on the primary key, so the table is also in 3NF. Lastly, since there are no prime attributes (Speaker_number) that are dependent on a non-prime attribute (the second condition for 3NF regarding prime attributes does not apply), the table is in BCNF.

RECORDING is in BCNF. The table is in 1NF as each of the values are atomic. Each of the non-prime attributes (Interviewer_name, Year_recorded, Publication_date, Number_of_speakers, Family_business, Description) are fully functionally dependent on the primary key ({Recording_title, Audio_file}), meaning that the table is in 2NF. In addition, none of the non-prime attributes are transitively dependent on the primary key, so the table is also in 3NF. Lastly, for each functional dependency in the table (Ex. A->B), A is a superkey. Since there are no prime attributes (Recording_title, Audio_file) that are dependent on a non-prime attribute, the table is in BCNF.

TOPIC_OR_CATEGORY is in BCNF because the relation is a binary relation. The table is in 1NF as each of the values are atomic. The non-prime attribute Topic_or_category_name is fully functionally dependent on the primary key Recording_title, so the table is in 2NF. Since the table only has two attributes, a transitive dependency cannot occur, so the table is also in 3NF. Lastly, for the functional dependency in the table (Recording_title->Topic_or_category_name), Recording_title is a superkey, meaning that the table is also in BCNF.

INTERVIEW_WITH is in BCNF because the relation is a binary relation. The table is in 1NF as each of the values are atomic. Since partial dependency is not possible, the table is in 2NF. In addition, the table is also in 3NF since a transitive dependency cannot occur in a relation with only two attributes. Lastly, the attributes in the table (Recording_title and Speaker_number) can be individual superkeys, or they can be combined as a set to be one superkey, meaning that the table is also in BCNF.

SUPERUSER is in BCNF because the relation is a binary relation. The table is in 1NF as each of the values are atomic. The non-prime attribute Password is fully functionally dependent on the primary key Username, so the table is in 2NF. Since the table only has two attributes, a transitive dependency cannot occur, so the table is also in 3NF. Lastly, for the functional dependency in the table (Username->Password), Username is a superkey, meaning that the table is also in BCNF.

EDITED is in BCNF. The table is in 1NF as each of the values are atomic. Each of the non-prime attributes (Recording_title, Username, Message) are fully functionally dependent on the primary key (Edit_id), meaning that the table is in 2NF. In addition, none of the non-prime attributes are transitively dependent on the primary key, so the table is also in 3NF. Lastly, for each functional dependency in the table (Ex. A->B), A (Edit_id) is a superkey. Since there are no prime attributes (Edit_id) that are dependent on a non-prime attribute (the second condition for 3NF regarding prime attributes does not apply), the table is in BCNF.

TRANSCRIPT is in BCNF because the relation is a binary relation. The table is in 1NF as each of the values are atomic. The attribute Transcript_text is fully functionally dependent on the primary key Transcript_id, so the table is in 2NF. Since the table only has two attributes, a transitive dependency cannot occur, so the table is also in 3NF. Lastly, for the functional dependency in the table (Transcript_id->Transcript_text), Transcript_id is a superkey, meaning that the table is also in BCNF.

HAS_TRANSCRIPT is in BCNF because the relation is a binary relation. The table is in 1NF as each of the values are atomic. Since partial dependency is not possible, the table is in 2NF. In addition, the table is also in 3NF since a transitive dependency cannot occur in a relation with only two attributes. Lastly, the attributes in the table (Transcript_id and Recording_title) can be individual superkeys, or they can be combined as a set to be one superkey, meaning that the table is also in BCNF.

Step 3:

Views:

- **User**
    - A normal user that comes to the website to either listen to recordings, view transcripts, or read the descriptions for recordings.
    - Transaction Requirements:
        - Play a recording
            - Display the recording title and an audio file
        - Display a transcript
            - Display the transcript text as well as the recording title
        - Display description
            - Shows:
                - Description of the recording
                - Publication date
                - Year recorded
        - Filter recordings based on
            - Topics/Categories
            - The year the recording was recorded

            - Whether a Family Business is mentioned in the recording
            - The nationality of the speaker(s)
            - Whether or not the speaker(s) is Jewish
            - Gender of the speaker(s)
            - Number of speakers
        - Search based on recording title
    - Data Requirements:
        - Play a recording
            - Audio_file (attribute of Recording)
                - To play it
            - Recording_title (attribute of Recording)
                - To find the corresponding audio file
        - Display a transcript
            - Transcript_text (attribute of Transcript)
                - To display the text
            - Recording_title (attribute of Recording)
                - To find the corresponding transcript to a recording
                - Will be used to do a join on the Recording and Has_Transcript relations
            - Transcript_ID (attribute of Transcript)
                - Will be used to do a join on the Transcript relation and the already existing join on the Recording and Has_Transcript relations
        - Display description
            - Description (attribute of Recording)
                - To display it
            - Publication_date (attribute of Recording)
                - To display it
            - Year_recorded (attribute of Recording)
                - To display it
            - Recording_title (attribute of Recording)
                - To find the corresponding description, publication date, and year recorded of a recording
        - Filter recordings based on the filters listed above
            - Topic_or_ category_name (multivalued attribute of Recording; attribute of relation Topic_Or_Category)
                - To filter recordings based on user input (which topics and categories the user wants the recordings displayed to be about)
            - Year_recorded (attribute of Recording)
                - To filter recordings based on user input (to display which recordings were recorded in the years that the user selected)
            - Family_business (attribute of Recording)

            - To filter recordings based on user input (to display recordings that mention a family business or not)
        - Number_of_speakers (attribute of Recording)
            - To filter recordings based on user input (to display recordings that have a certain number of speakers)
        - Nationality (attribute of Speaker)
            - To filter recordings based on user input (to display only recordings that have at least one speaker of the nationality selected by the user)
        - Is_Jewish (attribute of Speaker)
            - To filter recordings based on user input (to display only recordings that have at least one speaker that is Jewish)
        - Gender (attribute of Speaker)
            - To filter recordings based on user input (to display only recordings that have at least one speaker that is the gender selected by the user)
        - Recording_title (attribute of Recording)
            - To find the corresponding attributes of Year_recorded, Family_business, Number_of_speakers
            - To display the recording titles after the filter has completed
            - To do a join on the relations Recording and Topic_or_category in order to get all the corresponding topics/categories of a recording so as to filter on it
            - To do a join on the relations Recording and Interview_with so as to later know if any of the speakers are Jewish or are of the selected gender or nationality after doing a join on the resulting join and the relation Speaker
        - Speaker_number (attribute of Speaker)
            - To do a join on the relation Speaker and the resulting relation from the join on Recording and Interview_with
    - Search based on recording title
        - Recording_title
            - Needed to find the corresponding recording
- Example Queries:
    - Filter the recordings to find recordings that were recorded in 1989, have a Jewish speaker, and have exactly 1 speaker.
    - Show the description of the recording “Dr. Paul Loser”.
    - Search for the recording that has the recording title of “Izzy Lynn”.
    - Display a transcript for the recording “Mel Kushner”.

- **Superuser**
    - Can see everything and do everything that a normal user can do, but can also change the contents of the database.
    - Transaction Requirements:
        - Can do all the transactions that a user can do

        - Log in and out of the system
            - The user supplies the username and password
        - Display a recording's attributes and its corresponding speakers and transcript and their attributes as well
            - Upon supplying the title of a recording, the superuser will be able to see this data.
        - Add a recording
            - Add a tuple to the Recording relation which will be able to be accessed by the users after doing so.
        - Add a transcript
            - Add a tuple to the Transcript relation which will be able to be accessed by the users after doing so.
        - Add a speaker
            - Add a tuple to the Speaker relation (the superuser will have to supply the title of the recording)
        - Delete a recording
            - Delete a tuple from the Recording relation which will no longer be able to be accessed by the users after doing so.
        - Delete a transcript
            - Delete a tuple from the Transcript relation which will no longer be able to be accessed by the users after doing so.
        - Delete a speaker
            - Delete a tuple from the Speaker relation (the superuser will have to supply the speaker number)
        - Edit/Update a recording
            - Modify a tuple in the Recording relation which will be updated in the User View after doing so.
        - Edit/Update a transcript
            - Modify a tuple in the Transcript relation which will be updated in the User View after doing so.
        - Edit/Update a speaker
            - Modify a tuple in the Speaker relation (the superuser will have to supply the speaker number)
    - Data Requirements:
        - Log in and out of the system
            - Attributes of Superuser (needed to log into the system as a superuser)
                - Username
                - Password
        - Display a recording's attributes and its corresponding speakers and transcript and their attributes as well
            - Attributes of Recording (needed to display to the superuser):
                - Recording_title

            - Superuser input (needed to find the corresponding recording and its attributes)
            - Also needed to do a join on the relations Recording and Interview_with in order to later find the corresponding speakers
            - Also needed to do a join on the relations Recording and Transcript in order to later find the corresponding transcript
        - Audio_file
        - Year_recorded
        - Interviewer_name
        - Publication_date
        - Number_of_speakers
        - Family_business
        - Description
    - Attributes of Speaker (needed to display to the superuser):
        - Speaker_number
            - Needed to do a join on the relation Speaker and the resulting relation from the join on Recording and Interview_with so as to find the corresponding speakers for a recording
        - Name
        - Gender
        - Nationality
        - Is_Jewish
    - Attributes of Transcript (needed to display to the superuser):
        - Transcript_ID
            - Needed to do a join on the relation Transcript and the resulting relation from the join on Recording and Transcript so as to find the corresponding transcript for a recording
        - Transcript_text
- Add a recording
    - Attributes of Recording (to insert a tuple into the Recording relation):
        - Recording_title
        - Audio_file
        - Year_recorded
        - Interviewer_name
        - Publication_date
        - Number_of_speakers
        - Family_business
        - Description
- Add a transcript

        - Attributes of Transcript (to insert a tuple into the Transcript relation);
            - Transcript_ID
            - Transcript_text
- Add a speaker
    - Attributes of Speaker (to insert a tuple into the Speaker relation):
        - Speaker_number
        - Name
        - Gender
        - Nationality
        - Is_Jewish
- Delete a recording
    - Recording_title
        - Needed to delete a tuple from the Recording relation
- Delete a transcript
    - Transcript_ID
        - Needed to delete a tuple from the Transcript relation
- Delete a speaker
    - Speaker_number
        - Needed to delete a tuple from the Speaker relation
- Edit/Update a recording
    - Attributes of Recording (we don't know which attributes the superuser will edit, so we may potentially need them all):
        - Recording_title
            - Needed to find the recording to be edited
        - Audio_file
        - Year_recorded
        - Interviewer_name
        - Publication_date
        - Number_of_speakers
        - Family_business
        - Description
- Edit/Update a transcript
    - Attributes of Transcript (we don't know which attributes the superuser will edit, so we may potentially need them all);
        - Transcript_ID
            - Needed to find the transcript to be edited
        - Transcript_text
- Edit/Update a speaker
    - Attributes of Speaker (we don't know which attributes the superuser will edit, so we may potentially need them all):
        - Speaker_number
            - Needed to find the recording to be edited
        - Name

        - Gender
        - Nationality
        - Is_Jewish
    - Example Queries:
        - Add a recording called “Johnny Appleseed”.
        - Add a transcript to the recording “Dr. Paul Loser”.
        - Delete a recording called “Mel Kushner”.
        - Edit a recording called “Izzy Lynn”.

Step 4:

List of Transactions and their set of SQL queries:

User:

- Play a recording
    - SELECT Audio_file
      FROM RECORDING
      WHERE Recording_title = title;
    - title is representative of the desired recording
- Display a transcript
    - CREATE VIEW SEL AS
      SELECT *
      FROM RECORDING
      WHERE Recording_title = title;
        - title is representative of the desired recording
    - CREATE VIEW REC_HASTRANS AS
      SELECT *
      FROM SEL
      NATURAL JOIN HAS_TRANSCRIPT;
    - SELECT Transcript_text
      FROM REC_HASTRANS
      NATURAL JOIN TRANSCRIPT;
    - DROP REC_HASTRANS;
    - DROP SEL;
- Display description
    - SELECT Description, Publication_date, Year_recorded
      FROM RECORDING
      WHERE Recording_title = title;
    - title is representative of the desired recording
- Filter recordings based on Topics/Categories
    - CREATE VIEW RECORDING_TOPIC AS
      SELECT *
      FROM RECORDING
      NATURAL JOIN TOPIC_OR_CATEGORY;

    - SELECT Recording_title
      FROM RECORDING_TOPIC
      WHERE Topic_or_category_name = tc1 OR Topic_or_category_name = tc2 OR …;
        - tc1, tc2, … are the topics and categories selected by the user
    - DROP VIEW RECORDING_TOPIC;
- Filter recordings based on the year the recording was recorded
    - SELECT Recording_title
      FROM RECORDING
      WHERE Year_recorded = year1 OR Year_recorded = year2 OR ...;
    - year1, year2, … are the years selected by the user
- Filter recordings based on whether a Family Business is mentioned in the recording
    - SELECT Recording_title
      FROM RECORDING
      WHERE Family_Business = familyBusiness;
    - familyBusiness is representative of whether or not the user wants the recording to mention a family business
- Filter recordings based on the nationality of the speaker(s)
    - CREATE VIEW RECORDING_SPEAKER AS
      SELECT *
      FROM RECORDING
      NATURAL JOIN SPEAKER;
    - SELECT Recording_title
      FROM RECORDING_SPEAKER
      WHERE Nationality = n1 OR Nationality = n2 OR …;
        - n1, n2, … are the nationalities selected by the user
    - DROP VIEW RECORDING_SPEAKER;
- Filter recordings based on whether or not the speaker(s) is Jewish
    - CREATE VIEW RECORDING_SPEAKER AS
      SELECT *
      FROM RECORDING
      NATURAL JOIN SPEAKER;
    - SELECT Recording_title
      FROM RECORDING_SPEAKER
      WHERE Is_jewish = sIsJewish;
        - sIsJewish is representative of whether or not the user wants the recording to have a Jewish speaker
    - DROP VIEW RECORDING_SPEAKER;
- Filter recordings based on gender of the speaker(s)
    - CREATE VIEW RECORDING_GENDER AS
      SELECT *
      FROM RECORDING
      NATURAL JOIN SPEAKER;
    - SELECT Recording_title

FROM RECORDING_GENDER
WHERE Gender = gender1 OR Gender = gender2 OR …;
- gender1, gender2, … are the nationalities selected by the user

- DROP VIEW RECORDING_GENDER;

- Filter recordings based on number of speakers
  - SELECT Recording_title,
    FROM RECORDING
    WHERE Number_of_speakers = numOfSpeakers1 OR Number_of_speakers = numOfSpeakers2 OR ...;
  - numOfSpeakers1, numOfSpeakers2, … are the number of speakers selected by the user
- Search based on recording title
  - CREATE EXTENSION pg_trgm;
  - CREATE INDEX recording_title_trigram ON RECORDING
    USING gist (Recording_title gist_trgm_ops);
  - SELECT Recording_title
    FROM RECORDING
    WHERE Recording_title % 'title';

Superuser:

- Log in and out of the system
  - SELECT Username
    FROM SUPERUSER
    WHERE Username = username AND Password = password;
    - username and password are the username and password that the superuser supplies
    - Return the Username if a matching username and password is found
- Display a recording's attributes and its corresponding speakers and transcript and their attributes as well
  - CREATE VIEW SEL AS
    SELECT *
    FROM RECORDING
    WHERE Recording_title = title;
    - title is user input
  - CREATE VIEW REC_INTERVIEW AS
    SELECT *
    FROM SEL
    NATURAL JOIN INTERVIEW_WITH;
  - SELECT *
    FROM REC_INTERVIEW
    NATURAL JOIN SPEAKER;
    - The recording's attributes and the speaker's attributes are displayed
  - CREATE VIEW REC_HASTRANS AS

SELECT *
FROM SEL
NATURAL JOIN HAS_TRANSCRIPT;
    - SELECT Transcript_ID, Transcript_text
FROM REC_HASTRANS
NATURAL JOIN TRANSCRIPT;
    - DROP REC_HASTRANS;
    - DROP REC_INTERVIEW;
    - DROP SEL;
- Add a recording
    - INSERT INTO RECORDING(Recording_title, Audio_file, Year_recorded, Interviewer_name, Publication_date, Number_of_speakers, Family_business, Description)
VALUES (title, audioFile, year, interviewer, datePublished, numOfSpeakers, familyBusiness, descript);
    - The above values are representative of user input.
- Add a transcript
    - INSERT INTO TRANSCRIPT(Transcript_ID, Transcript_text)
VALUES (transcriptID, textCont);
    - The above values are representative of user input.
- Add a speaker
    - INSERT INTO SPEAKER(Speaker_number, Name, Gender, Nationality, Is_jewish)
VALUES (sNum, sName, sGender, sNationality, sIsJewish);
    - The above values are representative of user input.
- Delete a recording
    - DELETE FROM RECORDING
WHERE Recording_title = title;
    - Title is representative of user input.
- Delete a transcript
    - DELETE FROM TRANSCRIPT
WHERE Transcript_ID = trancriptID;
    - trancriptID is representative of user input.
- Delete a speaker
    - DELETE FROM SPEAKER
WHERE Speaker_number = sNum;
    - sNum is representative of user input.
- Edit/Update a recording
    - UPDATE RECORDING
SET Recording_title = title, Audio_file = audioFile, Year_recorded = year , Interviewer_name = interviewer, Publication_date = datePublished, Number_of_speakers = numOfSpeakers, Family_business = familyBusiness, Description = descript;

    - Depending on the user input, only some of these updates would take place. It depends on what the user wants to change in the tuple.
- Edit/Update a transcript
    - UPDATE TRANSCRIPT
      SET Transcript_ID = transcriptID, Transcript_text = textCont;
    - Depending on the user input, only some of these updates would take place. It depends on what the user wants to change in the tuple.
- Edit/Update a speaker
    - UPDATE SPEAKER
      SET Speaker_number = sNum, Name = sName, Gender = sGender , Nationality = sNationality, Is_jewish = sIsJewish;
    - Depending on the user input, only some of these updates would take place. It depends on what the user wants to change in the tuple.