Mingle Schema

The Events table contains the public events that have been created by other users of the app.
Events (event_id, user_id, location, date, time, event_description, other_information, verified_event)
    Foreign Key user_id references Users
event_id : (int) the unique id of an event
user_id : (int) the unique id of a user
location : (String) the exact place the event will occur
latitude: (float) latitude of the user's current location
longitude: (float) longitude of the user's current location
date_time: (DateTime) the day the event will occur
event_description: (String) other relevant information about the event the planner wishes to add
other_information: (String) other relevant information the planner wishes to add
verified_event: (bool) has this event been verified?

The Friends table contains profile information that can be displayed in the application which is linked to a User account.
Friends (user_id, first_name, age, hometown, state, country, about_me, profile_picture, verified_profile)
    Foreign Key user_id References Users
user_id: (int) unique user ID
first_name: (String) first name of the user
age: (int) the age of the user
hometown: (String) the city/hometown where the person is from
state: (String) the state where the person is from
country: (String) The country where the person is from
about_me: (String) a description of the user
profile_picture (varbinary(MAX)) the profile picture for the user
verified_profile (Bool) Is this user verified or not

The Users table contains basic profile information for each account necessary for login.
Users (user_id, username, password, latitude, longitude, email)
user_id: (int) unique user ID
username: (String) username the user has chosen to be identified by
password: (String) password required for log in.
latitude (float) The latitude of the user's location
longitude (float) The longitude of the user's location
email: (String) user email

The Filters table contains the preferences users are looking for in a friend.
Filters (user_id, distance, lower_age, upper_age)
    Foreign Key user_id References Users,
user_id: (int) unique user ID

distance: (int) represents how far away the user wants to look for other people/events in miles
lower_age: (int) represents a lower bound for the age range the user wants to look for.
upper_age: (int) represents an upper bound for the age range the user wants to look for.

The Matching_Friends table contains the user_id that each user has “matched” with for easy access when a user wishes to start a conversation.
Matching_Friends (user_id, matching_user_id)
Foreign Key user_id References Users,
Foreign Key matching_user_id References Users
user_id: (int) unique user ID
matching_user_id: (int) ID of the matched user

The Matching_Events table contains the event_id that each user has “matched” with. Basically it links the users to the events that they have expressed interest in.
Matching_Events (user_id, event_id)
Foreign Key user_id References Users,
Foreign Key event_id References Events
user_id: (int) unique user ID
event_id: (int) ID of the matched event

**Underlined fields represent primary keys, multiple underlined fields in a schema represent a composite key.*

How the Tables Relate:

Our table's main function is to help separate the information of each user of the Mingle app into sections of security, simplicity, and convenience.

Events: This table contains all of the Events that have been uploaded to the Mingle server for other users to express their interest or disinterest. The primary key, event_id, will be added to a related user’s user_id in the Matching_Events table.

Friends: This table contains all of the People (Possible Friends) that have uploaded their personal profile to the Mingle server for other users to express their interest or disinterest. The primary/foreign key, user_id, will be added to another user’s user_id in the Matching_Friends table as a one to one relationship of user_ids.

Users: This table contains all the user information of all the people who have registered a profile in our Mingle database. Users can create one friend profile (in the friend table) and multiple events (in the event table). The primary key, user_id, will be used to connect to all the other tables within the database.

Filters: This table contains the filters that users will set to the events and friends that a user sees when they're using the swiping algorithm to match with friends and events. The primary/foreign key, user_id, will be added to whatever preferences he or she has within this table as a one to one relationship to the Users table.

Matching_Friends: This table contains the unique user keys of two people who have matched together and now have the option to start talking. The primary key in this table is actually a composite key, consisting of user_id and matching_user_id. Those two keys are also a foreign key to the Users table as a one to one relationship .

Matching_Events: This table contains the unique user and event key to link to the specified event the user has shown interest in and will probably attend. The primary key in this table is actually a composite key, consisting of user_id and event_id. Those two keys are also a foreign key to the Users table and Events table respectively as a one to one relationship .