

Block Diagram

Group 2_do_4

Contribution

Josh Harvey: 25%

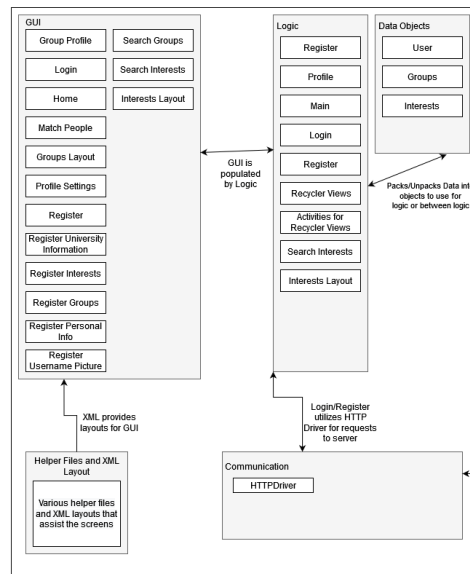
Yadiel Johnson: 25%

Molly Carrick: 25%

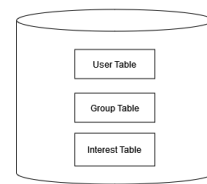
Nick Bergan: 25%

Block Diagram Picture

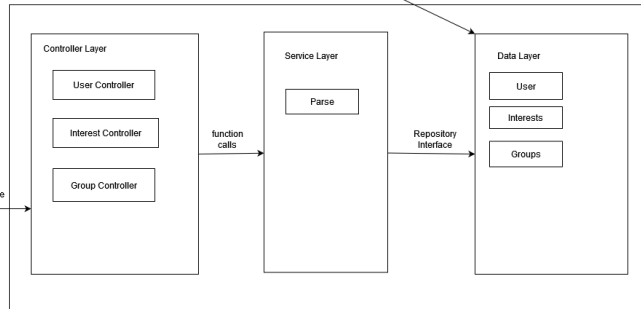
Client



Database



Server



Describe Complex Parts of Design

Client:

Gui: There are many screens that serve as the form of interaction with the application. The screens are assisted by java files and classes that receive and pass entered or queried information as well as populating the pages with data. A recycler view is used when cards of information need to be populated and scroll-able in a grid format.

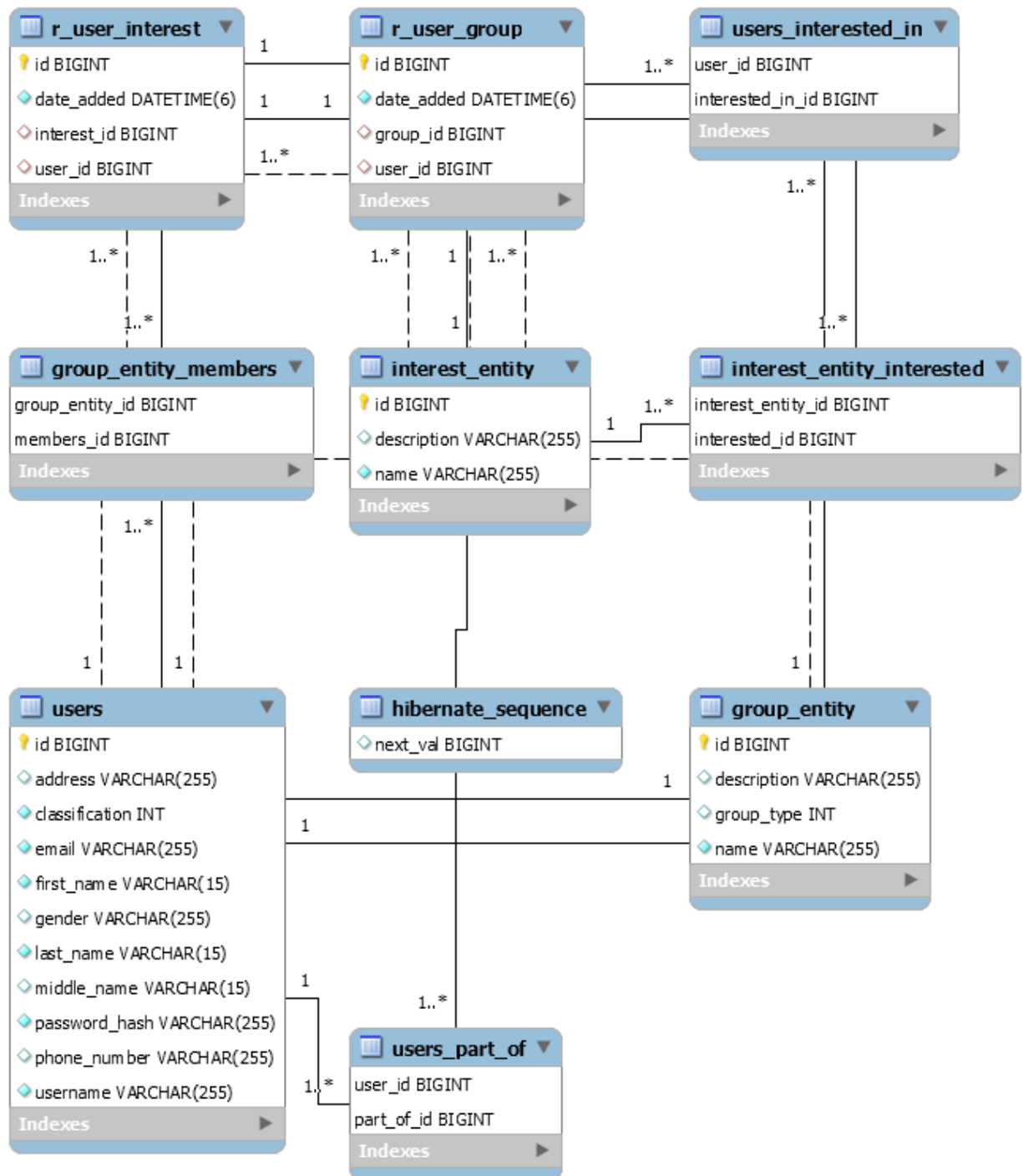
Java: The client communicates via the HTTP Driver, which makes get/post Volley requests. It utilizes raw JSON to pack/unpack structured data. The client can select a trait they want in their “match”, which is sent to the backend. There are many java files that control the xml pages, what is displayed and what is sent to the server. All HTTP requests are event driven and don't retain data on the driver; the HTTP driver consists of a list of independent functions for the different Java activity controllers.

Data Objects: The core features of our application are user profiles, groups, and interests. The user profiles will display user specific information like their groups, interests and friends. Groups allow for users to share a meeting place. Interests make it easier for users to find others with similar interests for matching.

Server & Database:

To implement the relations in the database in an expandable manner, there was a need to evaluate where the application would head as we progressed. There are simpler ways to implement the relations that limit the ability to expand the program's functionality past the bare minimum. The current implementation utilizes multiple tables to establish the relations at the cost of readability and performance.

Table Relationship Diagram



Tables List

- Cybuds: schema
 - **interests: table**
 - Columns
 - description: varchar(250)
 - name: varchar(20) NN
 - id: bigint NN
 - Indices
 - unique (name)
 - Keys
 - PK (id)
 - AK (name)
 - **users: table**
 - Columns
 - id: bigint NN
 - address: varchar(255)
 - email: varchar(255) NN
 - first_name: varchar(15) NN
 - gender: varchar(255)
 - last_name: varchar(15) NN
 - middle_name: varchar(15)
 - password_hash: varchar(255) NN
 - phone_number: varchar(255)
 - username: varchar(255) NN
 - student_class: int
 - classification: int NN
 - Indices
 - unique (email)
 - unique (username)
 - Keys
 - PK (id)
 - AK (email)
 - AK (username)
 - **group_entity: table**
 - Columns
 - id: bigint NN
 - description: varchar(255)
 - group_type: int
 - name: varchar(255) NN
 - Keys
 - PK (id)
 - **group_entity_members: table**
 - Columns
 - group_entity_id: bigint NN
 - members_id: bigint NN
 - Indices
 - unique (members_id)
 - Keys
 - PK (group_entity_id, members_id)
 - AK (members_id)
 - Foreign keys

- foreign key (members_id) -> r_user_group (id)
 - foreign key (group_entity_id) -> group_entity (id)
- **interest_entity: table**
 - Columns
 - id: bigint NN
 - description: varchar(255)
 - name: varchar(255) NN
 - Indices
 - unique (name)
 - Keys
 - PK (id)
 - AK (name)
- **interest_entity_interested: table**
 - Columns
 - interest_entity_id: bigint NN
 - interested_id: bigint NN
 - Indices
 - unique (interested_id)
 - Keys
 - PK (interest_entity_id, interested_id)
 - AK (interested_id)
 - Foreign-keys
 - foreign key (interest_entity_id) -> interest_entity (id)
 - foreign key (interested_id) -> r_user_interest (id)
- **r_user_group: table**
 - Columns
 - id: bigint NN
 - date_added: datetime(6) NN
 - group_id: bigint
 - user_id: bigint
 - Indices
 - index (user_id)
 - index (group_id)
 - Keys
 - PK (id)
 - Foreign-keys
 - foreign key (user_id) -> users (id)
 - foreign key (group_id) -> group_entity (id)
- **r_user_interest: table**
 - Columns
 - id: bigint NN
 - date_added: datetime(6) NN
 - interest_id: bigint
 - user_id: bigint
 - Indices
 - index (user_id)
 - index (interest_id)
 - Keys
 - PK (id)
 - Foreign-keys
 - foreign key (user_id) -> users (id)

- foreign key (interest_id) -> interest_entity (id)
- **relation_user_interest: table**
 - Columns
 - id: bigint NN
 - date_added: datetime(6) NN
 - date_removed: datetime(6)
 - weight: float NN
 - interest_id: bigint
 - user_id: bigint
 - Indices
 - index (user_id)
 - index (interest_id)
 - Keys
 - PK (id)
 - Foreign-keys
 - foreign key (user_id) -> users (id)
 - foreign key (interest_id) -> interest_entity (id)
- **users_interested_in: table**
 - Columns
 - user_id: bigint NN
 - interested_in_id: bigint NN
 - Indices
 - unique (interested_in_id)
 - Keys
 - PK (user_id, interested_in_id)
 - AK (interested_in_id)
 - Foreign-keys
 - foreign key (user_id) -> users (id)
 - foreign key (interested_in_id) -> r_user_interest (id)
- **users_part_of: table**
 - Columns
 - user_id: bigint NN
 - part_of_id: bigint NN
 - Indices
 - unique (part_of_id)
 - Keys
 - PK (user_id, part_of_id)
 - AK (part_of_id)
 - Foreign-keys
 - foreign key (user_id) -> users (id)
 - foreign key (part_of_id) -> r_user_group (id)