# Database Migration

## Migrations

- **Users** Table

  ```
  public function up()
     {
         Schema::create('users', function (Blueprint $table) {
             $table->bigIncrements('user_id'); //Change this to our ID
             $table->string('first_name');
            $table->string('last_name');
            $table->string('middle_name');
             $table->string('email')->unique();
             $table->timestamp('email_verified_at')->nullable();
             $table->string('password');
            $table->enum('status', ['healthy', 'sick', 'quarantine']);
            $table->enum('vaccinated', ['yes', 'no', 'NA']);
            $table->timestamp('last_login'); //Also check time zone
             $table->rememberToken();
         });
     }
  ```

  - **Constraints**
    - Emails must be unique so only one can be associated per account
    - The email verification timestamp is nullable as there will be nothing to record upon creation of the user
  - **Testing inputs with the fillable property**

    ```
    App\Models\User::create(['first_name'=>"Test", 'last_name'=>"one",
    'middle_name'=>"user",'email'=>"User1@gmail.com",
    'password'=>"123",'status'=>"healthy",'vaccinated'=>"yes",'last_login'=> new
    DateTime()]);

    App\Models\User::create(['first_name'=>"Test", 'last_name'=>"Two",
    'middle_name'=>"User",'email'=>"User2@gmail.com",
    'password'=>"123",'status'=>"healthy",'vaccinated'=>"yes",'last_login'=> new
    DateTime()]);
    ```

- **Contacts** table

```
public function up()
   {
       Schema::create('contacts', function (Blueprint $table) {
           $table->unsignedBigInteger->('requester_id');
          $table->unsignedBigInteger->('addressee_id');
          $table->enum('status', ['pending', 'accepted']);
          $table->primary(['requester_id', 'addresse_id']);
          $table->foreign('requester_id')->references('user_id')->on('users');
          $table->foreign('addressee_id')->references('user_id')->on('users');
       });
   }
```

- **Constraints**
  - Since both users ID's are foreign they must be existing users in the **Users** table
- **Testing inputs with the fillable property**

```
App\Models\Contact::create(['requester_id'=>1, 'addressee_id'=>2, 'status' => "pending"]);

App\Models\Contact::create(['requester_id'=>2, 'addressee_id'=>1, 'status' => "pending"]);
```

- **Note**
  - The primary key is made up of both requester and addresse id

- **Reports** table

```
public function up()
   {
       Schema::create('reports', function (Blueprint $table) {
          $table->bigIncrements->('report_id');
          $table->unsignedBigInteger('user_id');
          $table->enum('type',['sick, exposed']);
          $table->timestamp('reported');
          $table->longText('notes');
          $table->boolean('has_attachment');
          $table->boolean('is_anonymous');
          $table->foreign('user_id')->references('user_id')->on('users');
       });
   }
```

- **Testing inputs with the fillable property**

```
App\Models\Report::create(['user_id' => 1, 'type'=>"sick", 'reported'=>new
DateTime(), 'notes'=>"I am very very sick", 'has_attachment'=>false,
'is_anonymous' => true]);

App\Models\Report::create(['user_id' => 2, 'type'=>"sick", 'reported'=>new
DateTime(), 'notes'=>"I am very very sick", 'has_attachment'=>false,
'is_anonymous' => true]);
```

- **Attachment** table

```
public function up()
   {
       Schema::create('attachment', function (Blueprint $table) {
          $table->bigIncrements->('file_id');
          $table->unsignedBigInteger('report_id');
          $table->string('file_location');
          $table->foreign('report_id')->references('report_id')->on('reports');
       });
   }
```

  - **Testing inputs with the fillable property**

```
App\Models\Attachment::create(['report_id'=>2,
'file_location'=>"\data\image2333"]);
```

- **Organizations** table

```
public function up()
   {
       Schema::create('organizations', function (Blueprint $table) {
          $table->bigIncrements->('organization_id');
          $table->unsignedBigInteger('creator_id');
          $table->string('organization_name');
          $table->timestamp('date_created');
          $table->foreign('creator_id')->references('user_id')->on('users');
       });
   }
```

  - **Testing inputs with the fillable property**

```
App\Models\Organization::create(['leader_id'=>2, 'organization_name' => "The
Org"]);
```

- **Organization_Members** table

  ```
  public function up()
     {
         Schema::create('organization_members', function (Blueprint $table) {
            $table->unsignedBigInteger->('organization_id');
            $table->unsignedBigInteger('member_id');
            $table->enum('role',['admin, member', '?']);
            $table->enum('status',['?', '?', '?']);
            $table->primary(['organization_id', 'member_id']);
            $table->foreign('organization_id')->references('organization_id')-
  >on('organizations');
            $table->foreign('member_id')->references('user_id')->on('users');
         });
     }
  ```

  - **Testing inputs with the fillable property**

    ```
    App\Models\Organization_Member::create(['organization_id'=>1, 'member_id'=>1,
    'role'=>"member", 'status'=>"pending"]);
    ```

- **Organization_Networks** table

  ```
  public function up()
     {
         Schema::create('organization_networks', function (Blueprint $table) {
            $table->bigIncrements->('network_id');
            $table->unsignedBigInteger('organization_id');
            $table->unsignedBigInteger('creator_id');
            $table->string('network_name');
            $table->timestamp('date_created');
           //Created by
            $table->foreign('organization_id')->references('organization_id')-
  >on('organizations');
           $table->foreign('creator_id')->references('user_id')->on('users');
         });
     }
  ```

  - **Testing inputs with the fillable property**

    ```
    App\Models\Organization_Network::create(['organization_id'=>1,
    'creator_id'=>1, 'network_name'=>"My sub network"]);
    ```

- **Network_Members** table

  ```
  public function up()
     {
         Schema::create('organization_networks', function (Blueprint $table) {
            $table->unsignedBigInteger->('network_id');
            $table->unsignedBigInteger('member_id');
            $table->foreign('network_id')->references('network_id')-
  >on('organization_networks');
           $table->foreign('member_id')->references('user_id')->on('users');
         });
     }
  ```

  - **Testing inputs with the fillable property**

    ```
    App\Models\Network_Member::create(['network_id'=>1, 'member_id'=>2]);
    ```