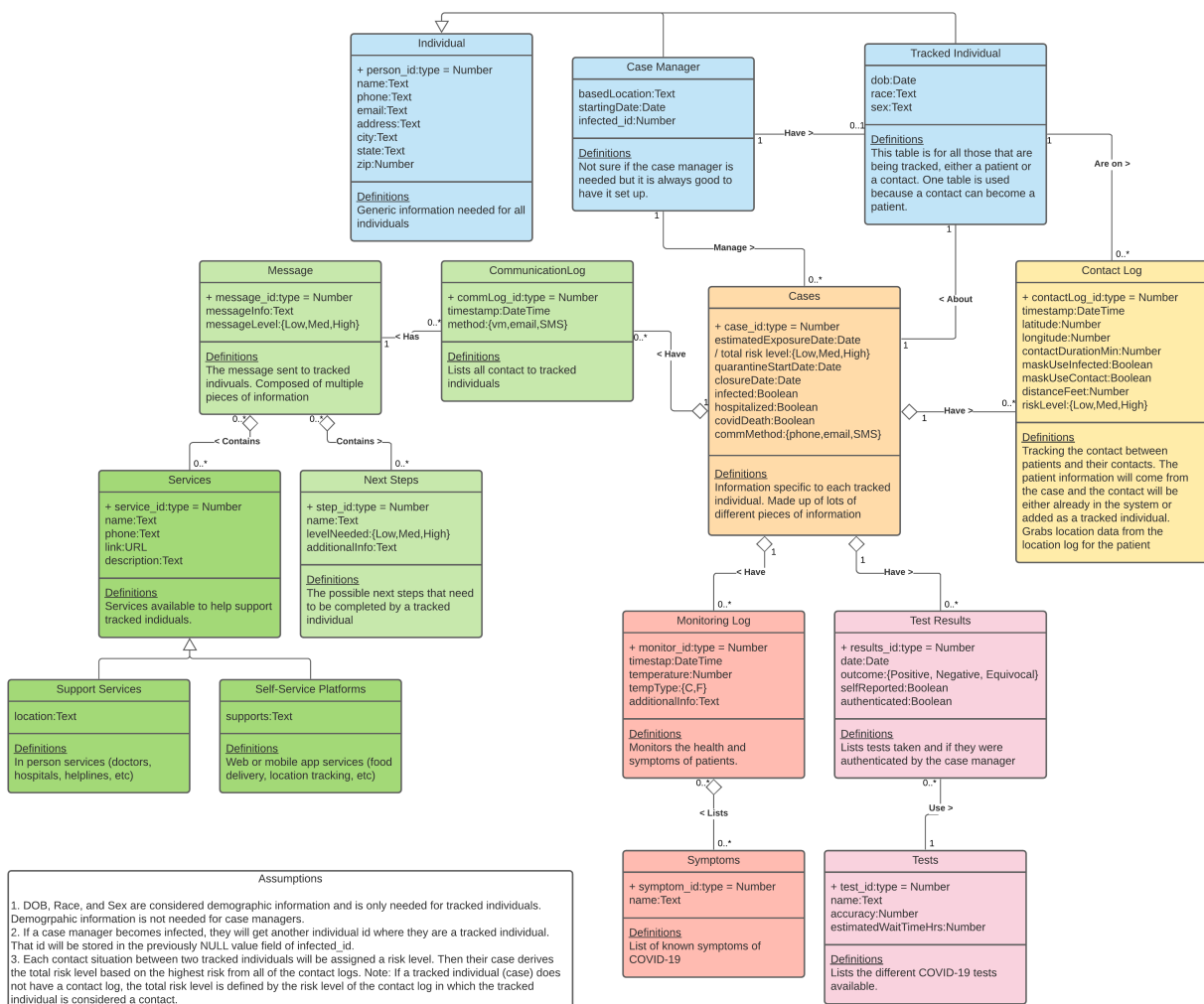# CS5200 Fall 2020: Practicum 1

Chandra Davis, Evan Douglass

Conceptual Model: UML

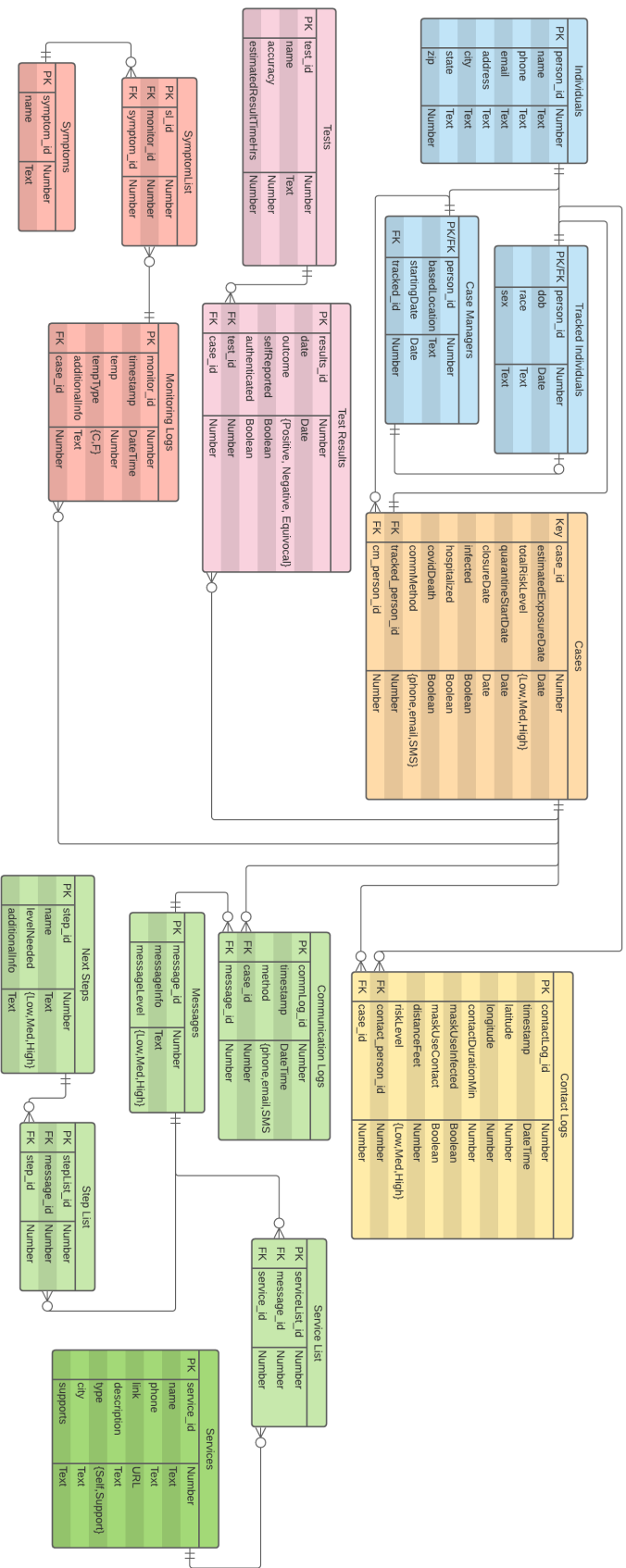View the conceptual model in Lucid Chart here:

https://app.lucidchart.com/invitations/accept/5602312e-dfc3-4423-975c-47190ce6022e

Logical Model: ERD

View the logical model in Lucid Chart here:

https://app.lucidchart.com/invitations/accept/7b497cbf-268d-4a03-b1a7-822b5a844fea

Individuals

PK	person_id	Number
	name	Text
	phone	Text
	email	Text
	address	Text
	city	Text
	state	Text
	zip	Number

Case Managers

PK/FK	person_id	Number
	basedLocation	Text
	startingDate	Date
FK	tracked_id	Number

Tracked Individuals

PK/FK	person_id	Number
	dob	Date
	race	Text
	sex	Text

Tests

PK	test_id	Number
	name	Text
	accuracy	Number
	estimatedResultTimeHrs	Number

SymptomList

PK	sl_id	Number
FK	monitor_id	Number
FK	symptom_id	Number

Symptoms

PK	symptom_id	Number
	name	Text

Test Results

PK	results_id	Number
	date	Date
	outcome	{Positive, Negative, Equivocal}
	selfReported	Boolean
	authenticated	Boolean
FK	test_id	Number
FK	case_id	Number

Monitoring Logs

PK	monitor_id	Number
	timestamp	DateTime
	temp	Number
	tempType	(C,F)
	additionalInfo	Text
FK	case_id	Number

Cases

Key	case_id	Number
	estimatedExposureDate	Date
	totalRiskLevel	{Low,Med,High}
	quarantineStartDate	Date
	closureDate	Date
	infected	Boolean
	hospitalized	Boolean
	covidDeath	Boolean
	commMethod	{phone,email,SMS}
FK	tracked_person_id	Number
FK	cm_person_id	Number

Contact Logs

PK	contactLog_id	Number
	timestamp	DateTime
	latitude	Number
	longitude	Number
	contactDurationMin	Number
	maskUseInfected	Boolean
	maskUseContact	Boolean
	distanceFeet	Number
	riskLevel	{Low,Med,High}
FK	contact_person_id	Number
FK	case_id	Number

Communication Logs

PK	commLog_id	Number
	timestamp	DateTime
	method	{phone,email,SMS}
FK	case_id	Number
FK	message_id	Number

Messages

PK	message_id	Number
	messageInfo	Text
	messageLevel	{Low,Med,High}

Next Steps

PK	step_id	Number
	name	Text
	levelNeeded	{Low,Med,High}
	additionalInfo	Text

Step List

PK	stepList_id	Number
FK	message_id	Number
FK	step_id	Number

Service List

PK	serviceList_id	Number
FK	message_id	Number
FK	service_id	Number

Services

PK	service_id	Number
	name	Text
	phone	Text
	link	URL
	description	Text
	type	{Self,Support}
	city	Text
	supports	Text

Schema

View the schema in Google Docs here:

https://docs.google.com/document/d/1o8pk51aed3BJSaBcwO2EMbT8I3ru_wpGTcTN4W-DbIM/edit?usp=sharing

Individuals(person_id: Number, name: Text, phone: Text, email: Text, address: Text, city: Text, state: Text, zip: Number)

TrackedIndividuals(person_id: Number, dob: Date, race: Text, sex: Text)

CaseManagers(person_id: Number, basedCity: Text, basedState: Text, startingDate: Date, infected_id: Number)

Tests(test_id: Number, name: Text, accuracy: Number, estimatedResultTimeHrs: Number)

TestResults(results_id: Number, date: Date, outcome: {Positive, Negative, Equivocal}, selfReported: Boolean, authenticated: Boolean, test_id: Number, case_id: Number)

MonitoringLogs(monitor_id: Number, timestamp: DateTime, temp: Number, tempType: {C, F}, additionalInfo: Text, case_id: Number)

SymptomList(sl_id: Number, monitor_id: Number, symptom_id: Number)

Symptoms(symptom_id: Number, name: Text)

Cases(case_id: Number, estimatedExposureDate: Date, totalRiskLevel: {Low, Med, High}, quarantineStartDate: Date, closureDate: Date, infected: Boolean, hospitalized: Boolean, covidDeath: Boolean, commMethod: {phone, email, SMS}, cm_person_id: Number, infected_person_id: Number)

ContactLogs(contactLog_id: Number, timestamp: DateTime, latitude: Number, longitude: Number, contactDurationMin: Number, maskUseInfected: Boolean, maskUseContact: Boolean, distanceFeet: Number, riskLevel: {Low, Med, High}, contact_person_id: Number, case_id: Number)

CommunicationLogs(commLog_id: Number, timestamp: DateTime, method: {phone, email, SMS}, case_id: Number, message_id: Number)

Messages(message_id: Number, messageLevel: {Low, Med, High}, messageInfo: Text)

StepList(stepList_id: Number, message_id: Number, step_id: Number)

NextSteps(step_id: Number, name: Text, levelNeeded: {Low, Med, High}, additionalInfo: Text)

ServiceList(serviceList_id: Number, message_id: Number, service_id: Number)

Services(service_id: Number, name: Text, phone: Text, link: Text, description: Text, type: {Self, Support}, city: Text, supports: Text)

Creating Database Tables

Should you wish to inspect the scripts that create the database and populate data, they can be found at:

https://github.com/eldss-classwork/databases-practicum1-scripts

The following images will show a progression from an empty database through table creation in MySQL Workbench.
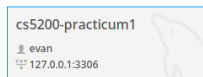
The MySQL Workbench start screen.

The newly created, empty `test` database.
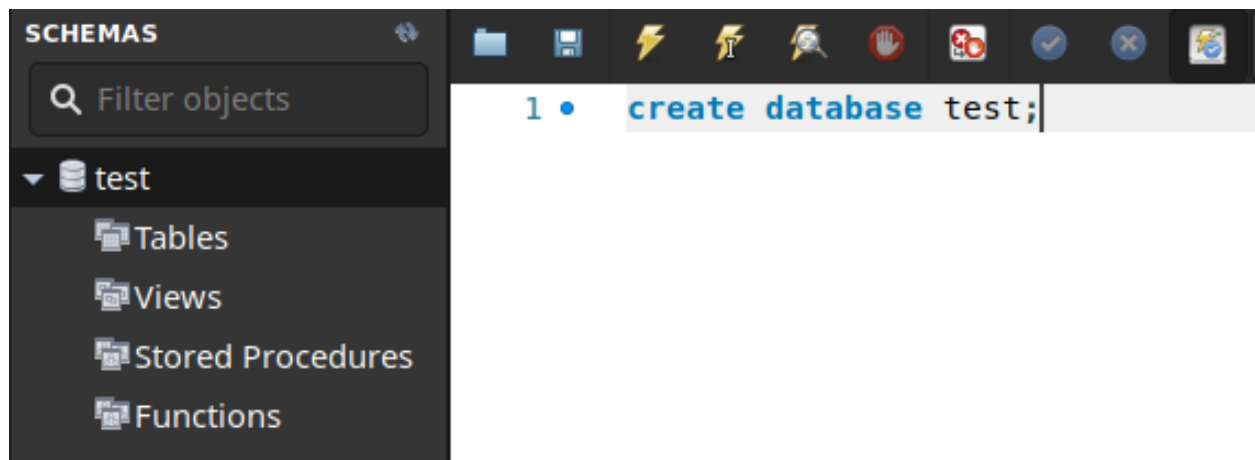
The `test` database after table creation.

```
SCHEMAS
Filter objects
test
  Tables
    CaseManagers
    Cases
    CommunicationLogs
    CommunicationMethods
    ContactLogs
    Individuals
    Messages
    MonitoringLogs
    NextSteps
    PossibleTestOutcomes
    RiskLevels
    ServiceList
    Services
    StepList
    SymptomList
    Symptoms
    TestResults
    Tests
    TrackedIndividuals
  Views
  Stored Procedures
  Functions

Added new script editor
```

```
Limit to 1000 rows
 1    -- This script provides statements to build the tables required
 2    -- of a contact tracing database.
 3
 4    -- Start by dropping tables
 5  • SET foreign_key_checks = 0;
 6  • DROP TABLE IF EXISTS `ContactLogs`;
 7  • DROP TABLE IF EXISTS `CommunicationLogs`;
 8  • DROP TABLE IF EXISTS `ServiceList`;
 9  • DROP TABLE IF EXISTS `Services`;
10  • DROP TABLE IF EXISTS `StepList`;
11  • DROP TABLE IF EXISTS `Individuals`;
12  • DROP TABLE IF EXISTS `NextSteps`;
13  • DROP TABLE IF EXISTS `CaseManagers`;
14  • DROP TABLE IF EXISTS `Cases`;
15  • DROP TABLE IF EXISTS `TrackedIndividuals`;
16  • DROP TABLE IF EXISTS `Messages`;
17  • DROP TABLE IF EXISTS `SymptomList`;
18  • DROP TABLE IF EXISTS `Symptoms`;
19  • DROP TABLE IF EXISTS `MonitoringLogs`;
20  • DROP TABLE IF EXISTS `TestResults`;
21  • DROP TABLE IF EXISTS `PossibleTestOutcomes`;
22  • DROP TABLE IF EXISTS `Tests`;
23  • DROP TABLE IF EXISTS `CommunicationMethods`;
24  • DROP TABLE IF EXISTS `RiskLevels`;
25  • SET foreign_key_checks = 1;
26
27    -- Recreate the tables
28  • CREATE TABLE `Individuals` (
29      `person_id` Int PRIMARY KEY auto_increment,
30      `name` Text NOT NULL,
31      -- Tracked people demographics are voluntary
32      `phone` Text,
33      `email` Text,
34      `address` Text,
35      `city` Text,
36      `state` Text,
```

Populating The Database

Queries