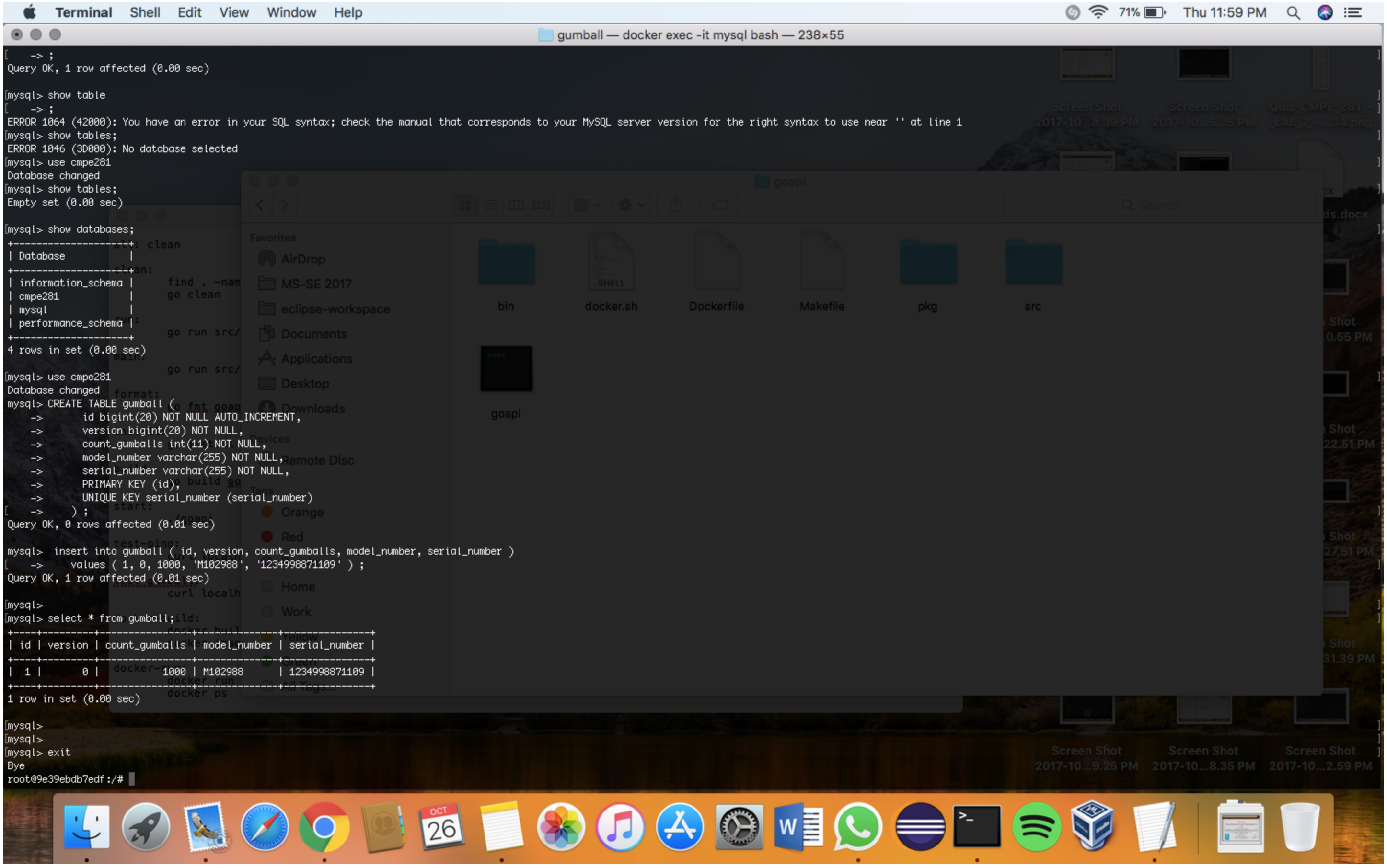
Go Gumball Docker Lab

STEP 1 - INSTALL DOCKER IMAGES & SET UP MYSQL DB

After MySQL Image is install, startup MySQL Docker Image and create the following data in a Schema: (i.e. Connect to MySQL Server inside Docker Container or outside using MySQL Workbench)

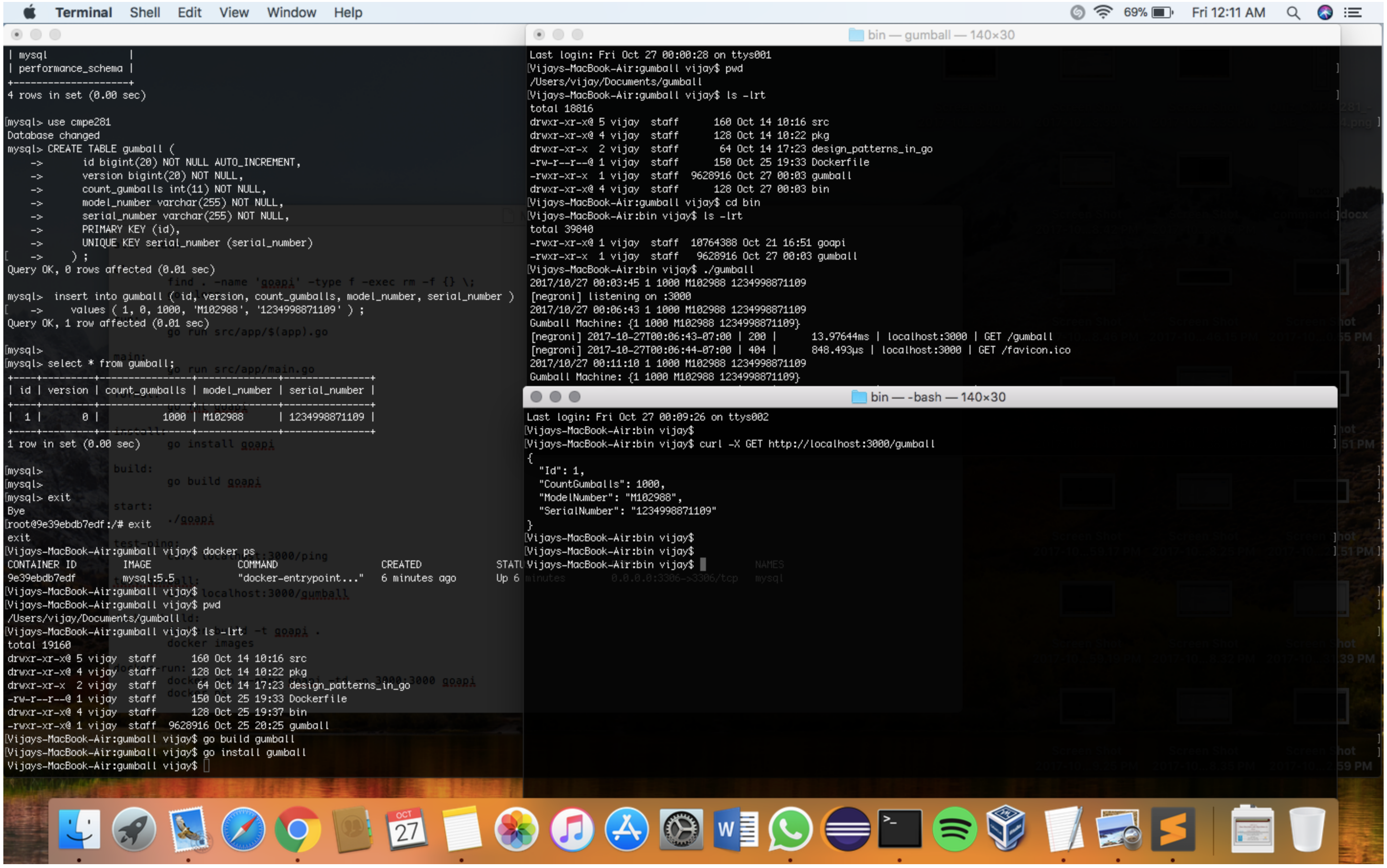
```
CREATE TABLE gumball (
  id bigint(20) NOT NULL AUTO_INCREMENT,
  version bigint(20) NOT NULL,
  count_gumballs int(11) NOT NULL,
  model_number varchar(255) NOT NULL,
  serial_number varchar(255) NOT NULL,
  PRIMARY KEY (id),
  UNIQUE KEY serial_number (serial_number)
) ;

insert into gumball ( id, version, count_gumballs, model_number, serial_number )
values ( 1, 0, 1000, 'M102988', '1234998871109' ) ;
```



```
    -> ;
Query OK, 1 row affected (0.00 sec)

mysql> show table
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1
mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use cmpe281
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| cmpe281            |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.00 sec)

mysql> use cmpe281
Database changed
mysql> CREATE TABLE gumball (
    ->         id bigint(20) NOT NULL AUTO_INCREMENT,
    ->         version bigint(20) NOT NULL,
    ->         count_gumballs int(11) NOT NULL,
    ->         model_number varchar(255) NOT NULL,
    ->         serial_number varchar(255) NOT NULL,
    ->         PRIMARY KEY (id),
    ->         UNIQUE KEY serial_number (serial_number)
    ->       ) ;
Query OK, 0 rows affected (0.01 sec)

mysql>  insert into gumball ( id, version, count_gumballs, model_number, serial_number )
    ->      values ( 1, 0, 1000, 'M102988', '1234998871109' ) ;
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> select * from gumball;
+----+---------+----------------+--------------+---------------+
| id | version | count_gumballs | model_number | serial_number |
+----+---------+----------------+--------------+---------------+
|  1 |       0 |           1000 | M102988      | 1234998871109 |
+----+---------+----------------+--------------+---------------+
1 row in set (0.00 sec)

mysql>
mysql>
mysql> exit
Bye
root@9e39ebdb7edf:/#
```

STEP 2 - Modify and Test Go Gumball API

- Download Source Code: gumball-go.zip
- Make changes to the file: **src/gumball/server.go** to connect to your MySQL DB
- Build and Test your modification
- Run your Go Gumball API locally



```
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.00 sec)

mysql> use cmpe281
Database changed
mysql> CREATE TABLE gumball (
    ->       id bigint(20) NOT NULL AUTO_INCREMENT,
    ->       version bigint(20) NOT NULL,
    ->       count_gumballs int(11) NOT NULL,
    ->       model_number varchar(255) NOT NULL,
    ->       serial_number varchar(255) NOT NULL,
    ->       PRIMARY KEY (id),
    ->       UNIQUE KEY serial_number (serial_number)
    ->     ) ;
Query OK, 0 rows affected (0.01 sec)

mysql>  insert into gumball ( id, version, count_gumballs, model_number, serial_number )
    ->    values ( 1, 0, 1000, 'M102988', '1234998871109' ) ;
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> select * from gumball;
+----+---------+----------------+--------------+---------------+
| id | version | count_gumballs | model_number | serial_number |
+----+---------+----------------+--------------+---------------+
|  1 |       0 |           1000 | M102988      | 1234998871109 |
+----+---------+----------------+--------------+---------------+
1 row in set (0.00 sec)

mysql>
mysql>
mysql> exit
Bye
root@9e39ebdb7edf:/# exit
exit
Vijays-MacBook-Air:gumball vijay$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATU
9e39ebdb7edf        mysql:5.5           "docker-entrypoint..."   6 minutes ago       Up 6
Vijays-MacBook-Air:gumball vijay$
Vijays-MacBook-Air:gumball vijay$ pwd
/Users/vijay/Documents/gumball
Vijays-MacBook-Air:gumball vijay$ ls -lrt
total 19160
drwxr-xr-x@ 5 vijay  staff      160 Oct 14 10:16 src
drwxr-xr-x@ 4 vijay  staff      128 Oct 14 10:22 pkg
drwxr-xr-x  2 vijay  staff       64 Oct 14 17:23 design_patterns_in_go
-rw-r--r--@ 1 vijay  staff      150 Oct 25 19:33 Dockerfile
drwxr-xr-x@ 4 vijay  staff      128 Oct 25 19:37 bin
-rwxr-xr-x@ 1 vijay  staff  9628916 Oct 25 20:25 gumball
Vijays-MacBook-Air:gumball vijay$ go build gumball
Vijays-MacBook-Air:gumball vijay$ go install gumball
Vijays-MacBook-Air:gumball vijay$
```

bin — gumball — 140×30

```
Last login: Fri Oct 27 00:00:28 on ttys001
Vijays-MacBook-Air:gumball vijay$ pwd
/Users/vijay/Documents/gumball
Vijays-MacBook-Air:gumball vijay$ ls -lrt
total 18816
drwxr-xr-x@ 5 vijay  staff      160 Oct 14 10:16 src
drwxr-xr-x@ 4 vijay  staff      128 Oct 14 10:22 pkg
drwxr-xr-x  2 vijay  staff       64 Oct 14 17:23 design_patterns_in_go
-rw-r--r--@ 1 vijay  staff      150 Oct 25 19:33 Dockerfile
-rwxr-xr-x  1 vijay  staff  9628916 Oct 27 00:03 gumball
drwxr-xr-x@ 4 vijay  staff      128 Oct 27 00:03 bin
Vijays-MacBook-Air:gumball vijay$ cd bin
Vijays-MacBook-Air:bin vijay$ ls -lrt
total 39840
-rwxr-xr-x@ 1 vijay  staff  10764388 Oct 21 16:51 goapi
-rwxr-xr-x  1 vijay  staff   9628916 Oct 27 00:03 gumball
Vijays-MacBook-Air:bin vijay$ ./gumball
2017/10/27 00:03:45 1 1000 M102988 1234998871109
[negroni] listening on :3000
2017/10/27 00:06:43 1 1000 M102988 1234998871109
Gumball Machine: {1 1000 M102988 1234998871109}
[negroni] 2017-10-27T00:06:43-07:00 | 200 |      13.97644ms | localhost:3000 | GET /gumball
[negroni] 2017-10-27T00:06:44-07:00 | 404 |      848.493µs | localhost:3000 | GET /favicon.ico
2017/10/27 00:11:10 1 1000 M102988 1234998871109
Gumball Machine: {1 1000 M102988 1234998871109}
```

bin — -bash — 140×30

```
Last login: Fri Oct 27 00:09:26 on ttys002
Vijays-MacBook-Air:bin vijay$
Vijays-MacBook-Air:bin vijay$ curl -X GET http://localhost:3000/gumball
{
  "Id": 1,
  "CountGumballs": 1000,
  "ModelNumber": "M102988",
  "SerialNumber": "1234998871109"
}
Vijays-MacBook-Air:bin vijay$
Vijays-MacBook-Air:bin vijay$
Vijays-MacBook-Air:bin vijay$
```

STEP 3 - Build and Run a Docker Image with your Go Gumball API

- Submit the following:
  1. The contents of your **Dockerfile**
  2. The Docker Commands you used to build and run your Container
- **Submit in the Text Box**

Your Answer:

```
FROM golang:latest
EXPOSE 3000
RUN mkdir /app
ADD . /app/
WORKDIR /app
ENV GOPATH /app
RUN cd /app ; go install gumball
CMD ["/app/bin/gumball"]
```

```
docker build -t gumball .
```

```
docker run -d --name gumball --link mysql:mysql gumball
```

STEP 4 - Deploy and Configure Kong API Gateway.

- **Deploy Kong API Gateway with your Gumball API Stack with the following configuration:**
  1. Gumball API Container connects to MySQL Container
  2. Kong API Container connects to Cassandra and Gumball Containers
  3. Only the Kong API Container exposes ports externally.
  4. Gumball, MySQL and Cassandra Containers should not expose any ports.
- **Configure Kong API Gateway as follows:**
  1. Create an API with request path "/goapi" to route to the Gumball API
  2. Add the "**File Log**" Plugin to your Kong API with file path of "**/tmp/kong.log**"

  -
    ```
    docker run -d --name gumball --link mysql:mysql gumball
    ```
  -

  -
    ```
    docker run -d --name kong-database cassandra:2.2
    ```
  -

  -
    ```
    docker run -d --name kong --link gumball:gumball --link kong-database:kong-database -e "KONG_DATABASE=cassandra" -e "KONG_CASSANDRA_CONTACT_POINTS=kong-database" -e "KONG_PG_HOST=kong-database" -p 8000:8000 -p 8443:8443 -p 8001:8001 -p 7946:7946 -p 7946:7946/udp kong:0.9.9
    ```
  -

  -
    ```
    curl -i -X POST   --url http://192.168.99.100:8001/apis/   -d 'name=goapi'   -d 'request_path=/goapi'   -d 'upstream_url=http://gumball:3000'
    ```
  -

  -
    ```
    curl -X PATCH http://192.168.99.100:8001/apis/goapi --data 'strip_request_path=true'
    ```
  -

  -
    ```
    curl -X POST http://192.168.99.100:8001/apis/goapi/plugins --data "name=file-log" --data "config.path=/tmp/kong.log"
    ```