

Researching Project MongoDB and Mobile App

‘ Week1 Report ’

Lecturer: Mr. HEL Chanthan

Prepared by Mr. VANNAK Sovannroth

2019-2020



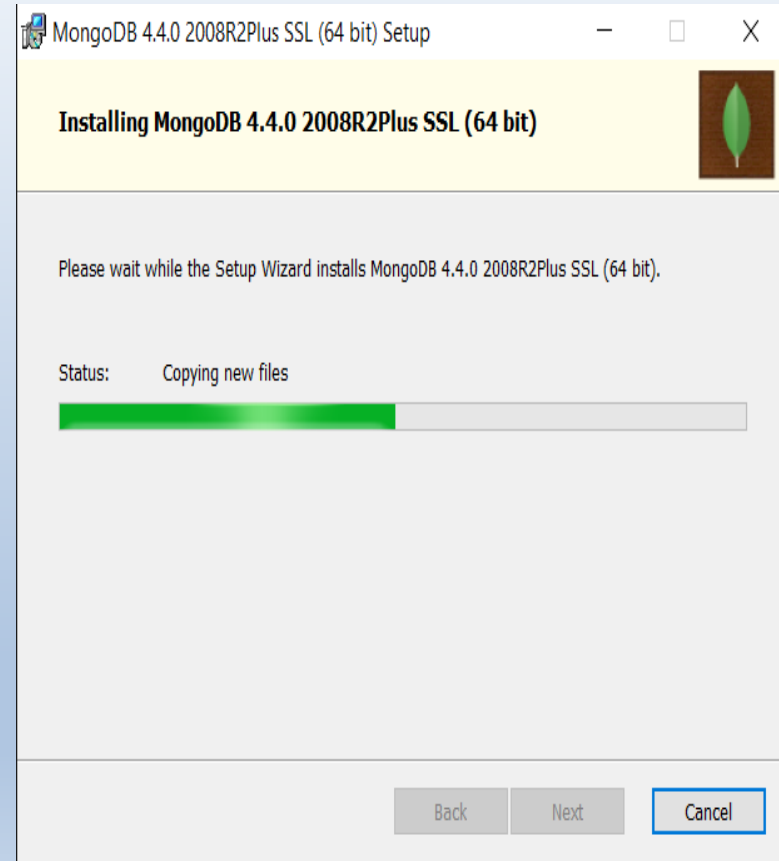
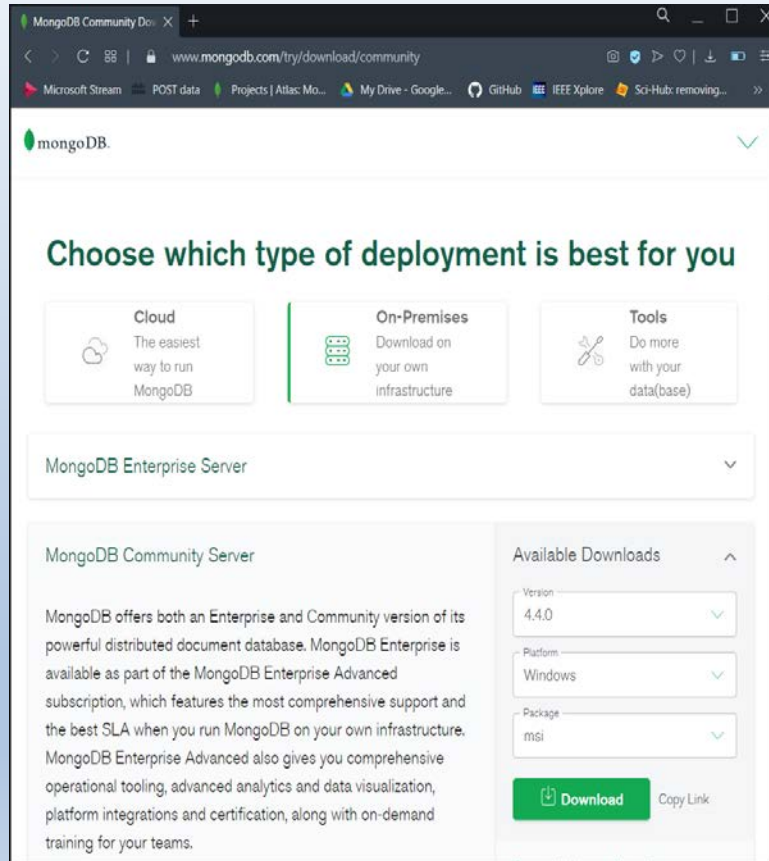
Outline

- ❖ Planning for First Month.
- ❖ Installation MongoDB (Shell and Compass)
- ❖ Connect localhost between Shell and Compass.(Locally)
- ❖ Performing CRUD (Create, Read, Update, Delete) on Mongo Shell and MongoDB Compass.
- ❖ Creating MongoDB Atlas account and Projects.
- ❖ Connect MongoDB Atlas to Mongo Shell or MongoDB Compass, and Driver (Node.js).(Cloud)
- ❖ Installation Visual Studio Code .
- ❖ Connecting Node.js to MongoDB Cluster. (Cloud)
- ❖ Connecting sample Backend to MongoDB Cluster and storing Products in the Database.

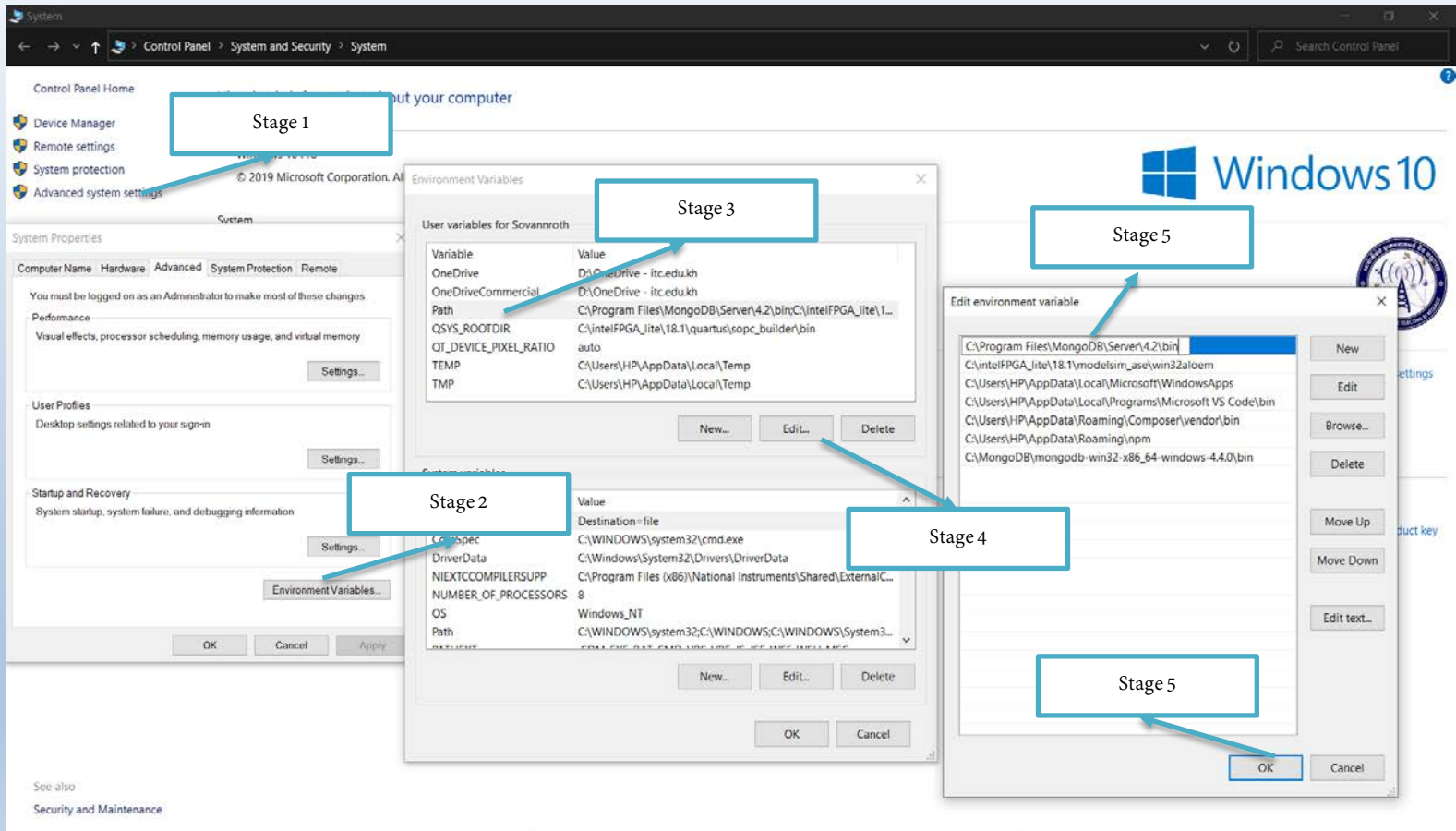
❖ Planning for First Month

Activity (Start on 06.08.2020)	Thursday (Presentation day)	Friday	Monday	Tuesday	Wednesday (Meeting day)
Week 1	Installation MongoDB (Shell and Compass) and connect localhost between Shell and Compass.(Locally)	Performing CRUD (Create, Read, Update, Delete) on Mongo Shell and MongoDB Compass.	Creating MongoDB Atlas account and connect it to Mongo Shell or MongoDB Compass, and Driver (Node.js).(Cloud)	Installation Visual Studio Code and connect driver to MongoDB Atlas.(Cloud)	Connecting sample Backend to MongoDB Cluster and storing Products in the Database.
Week 2	Connecting driver to Mongo Shell.(Locally)	Starting to build sample backend connect with and without MongoDB Atlas.	Build creating, editing and deleting Products on server.	Transmitting and fetching data to/from the Database.	Creating Login and Signup. (Email and password)
Week 3	Adding Users connect to database.	Starting to build backend for project.	Creating Login and Signup.	User management: <ul style="list-style-type: none"> • Create new user • Reset user password • Delete and edit user 	Manage new category: <ul style="list-style-type: none"> • Create new category • Edit and delete exiting categories.
Week 4	Manage news: <ul style="list-style-type: none"> • Post news • Edit news • Delete news 	Manage Info: <ul style="list-style-type: none"> • Logo • Name • Contact info • Map • Social 	Create Dashboard	Manage Calculate: <ul style="list-style-type: none"> • Addition • Subtraction • Multiplication • Division 	Get data from Sensors and Calculate.

❖ Installation MongoDB (Shell and Compass)



❖ Installation MongoDB (Shell and Compass)



❖ Connect localhost between Shell and Compass.(Locally)

➤ Mongo Shell

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.959]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\HP>mongod --port 27018

Select C:\WINDOWS\system32\cmd.exe - mongod --port 27018
2020-08-12T21:27:11.446+0700 I CONTROL [initandlisten] ** addresses it should serve re
sponses from, or with --bind_ip_all to
2020-08-12T21:27:11.447+0700 I CONTROL [initandlisten] ** bind to all interfaces. If t
his behavior is desired, start the
2020-08-12T21:27:11.449+0700 I CONTROL [initandlisten] ** server with --bind_ip 127.0
.0.1 to disable this warning.
2020-08-12T21:27:11.450+0700 I CONTROL [initandlisten]
2020-08-12T21:27:11.456+0700 I SHARDING [initandlisten] Marking collection local.system.replset
as collection version: <unsharded>
2020-08-12T21:27:11.460+0700 I STORAGE [initandlisten] Flow Control is enabled on this deployme
nt.
2020-08-12T21:27:11.461+0700 I SHARDING [initandlisten] Marking collection admin.system.roles as
collection version: <unsharded>
2020-08-12T21:27:11.462+0700 I SHARDING [initandlisten] Marking collection admin.system.version
as collection version: <unsharded>
2020-08-12T21:27:11.464+0700 I SHARDING [initandlisten] Marking collection local.startup_log as
collection version: <unsharded>
2020-08-12T21:27:11.777+0700 I FTDC [initandlisten] Initializing full-time diagnostic data c
apture with directory 'C:/data/db/diagnostic.data'
2020-08-12T21:27:11.782+0700 I SHARDING [LogicalSessionCacheRefresh] Marking collection config.s
ystem.sessions as collection version: <unsharded>
2020-08-12T21:27:11.782+0700 I SHARDING [LogicalSessionCacheReap] Marking collection config.trans
actions as collection version: <unsharded>
2020-08-12T21:27:11.782+0700 I NETWORK [listener] Listening on 127.0.0.1
2020-08-12T21:27:11.783+0700 I NETWORK [listener] waiting for connections on port 27018
2020-08-12T21:27:12.013+0700 I FTDC [ftdc] Unclean full-time diagnostic data capture shutdown
n detected, found interim file, some metrics may have been lost. OK
2020-08-12T21:27:12.108+0700 I SHARDING [ftdc] Marking collection local.oplog.rs as collection v
ersion: <unsharded>
```

```
C:\WINDOWS\system32\cmd.exe - mongo --port 27018
Microsoft Windows [Version 10.0.18363.959]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\HP>mongo --port 27018
MongoDB shell version v4.2.8
connecting to: mongodb://127.0.0.1:27018/?compressors=disabled&gssapiS
Implicit session: session { "id" : UUID("b757c9d5-4334-4b7b-8e06-6d925
MongoDB server version: 4.2.8
Server has startup warnings:
2020-08-12T21:27:11.438+0700 I CONTROL [initandlisten]
2020-08-12T21:27:11.438+0700 I CONTROL [initandlisten] ** WARNING: A
2020-08-12T21:27:11.439+0700 I CONTROL [initandlisten] **
2020-08-12T21:27:11.440+0700 I CONTROL [initandlisten]
2020-08-12T21:27:11.441+0700 I CONTROL [initandlisten] ** WARNING: T
2020-08-12T21:27:11.442+0700 I CONTROL [initandlisten] **
2020-08-12T21:27:11.444+0700 I CONTROL [initandlisten] **
2020-08-12T21:27:11.446+0700 I CONTROL [initandlisten] **
2020-08-12T21:27:11.447+0700 I CONTROL [initandlisten] **
2020-08-12T21:27:11.449+0700 I CONTROL [initandlisten] **
2020-08-12T21:27:11.450+0700 I CONTROL [initandlisten]

---
Enable MongoDB's free cloud-based monitoring service, which will then
metrics about your deployment (disk utilization, CPU, operation statis

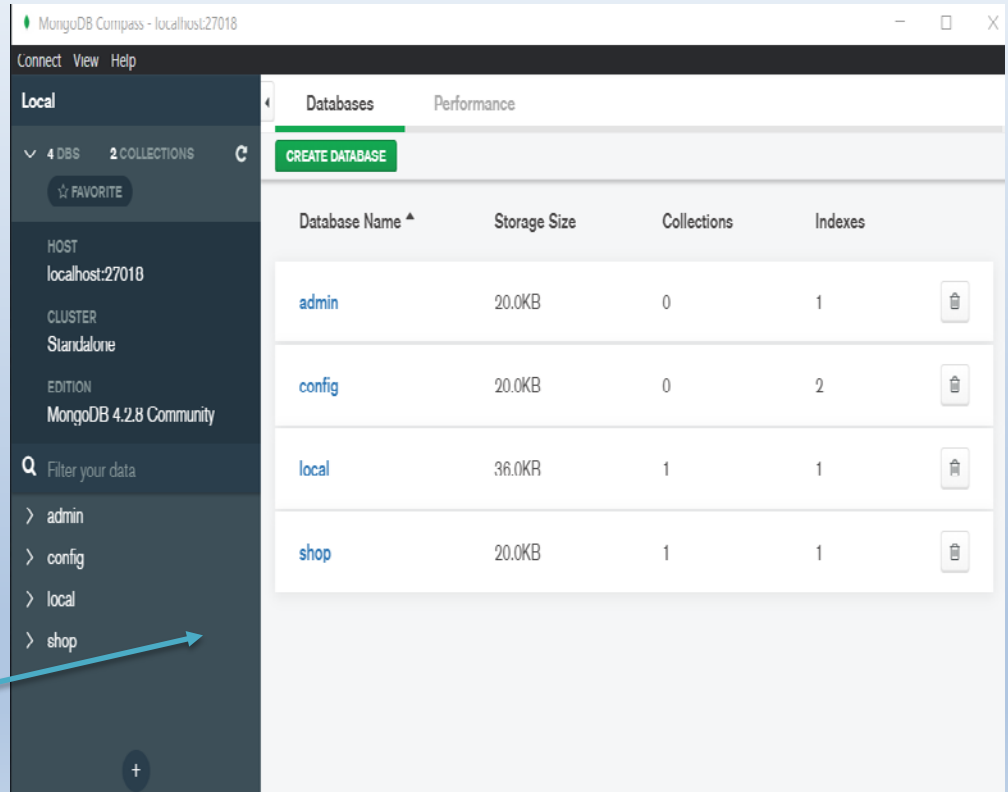
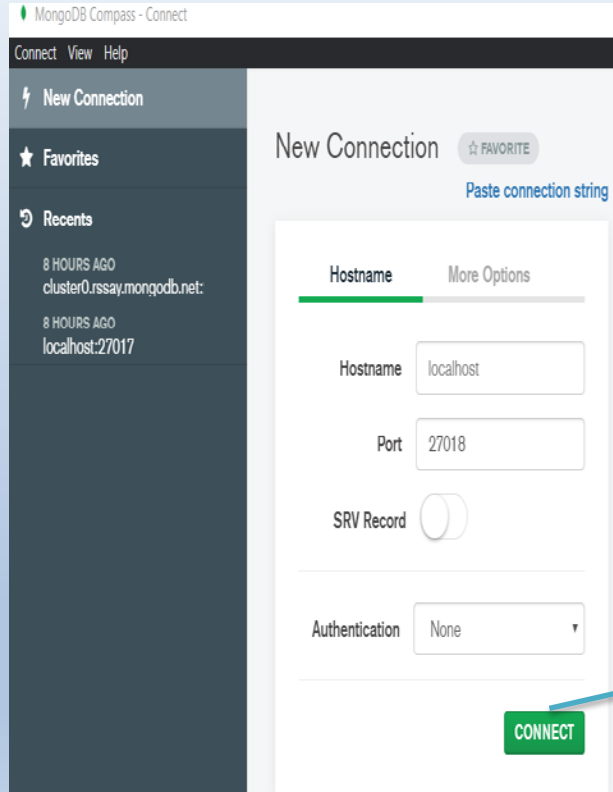
The monitoring data will be available on a MongoDB website with a uniq
and anyone you share the URL with. MongoDB may use this information to
improvements and to suggest MongoDB products and deployment options to

To enable free monitoring, run the following command: db.enableFreeMon
To permanently disable this reminder, run the following command: db.di
---

> show dbs
admin 0.000GB
config 0.000GB
local 0.000GB
> use shop
switched to db shop
```

❖ Connect localhost between Shell and Compass.(Locally)

➤ MongoDB Compass



❖ Performing CRUD (Create, Read, Update, Delete) on Mongo Shell and MongoDB Compass.

➤ Mongo Shell

```
C:\WINDOWS\system32\cmd.exe - mongo --port 27018
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
> use shop
switched to db shop
> show collections
> db.products.insertOne({name: "A Computer", price: 1229.99, description: "A high quality computer.", details: {cpu: "Intel i7 8770", memory: 32}})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("5f340428e24667d16a7c2d13")
}
> db.products.find()
{ "_id" : ObjectId("5f340428e24667d16a7c2d13"), "name" : "A Computer", "price" : 1229.99, "description" : "A high quality computer.", "details" : { "cpu" : "Intel i7 8770", "memory" : 32 } }
> db.products.find().pretty()
{
  "_id" : ObjectId("5f340428e24667d16a7c2d13"),
  "name" : "A Computer",
  "price" : 1229.99,
  "description" : "A high quality computer.",
  "details" : {
    "cpu" : "Intel i7 8770",
    "memory" : 32
  }
}
> db.products.insertMany([ {name: "Max Schwarz", age: 29, address: {street: "Main"} }, {name: "Manuel Lor", age: 30, address: {street: "Tree"}} ])
{
  "acknowledged" : true,
  "insertedIds" : [
```

```
C:\WINDOWS\system32\cmd.exe - mongo --port 27018
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("5f340496e24667d16a7c2d14"),
    ObjectId("5f340496e24667d16a7c2d15")
  ]
}
> db.products.find().pretty()
{
  "_id" : ObjectId("5f340428e24667d16a7c2d13"),
  "name" : "A Computer",
  "price" : 1229.99,
  "description" : "A high quality computer.",
  "details" : {
    "cpu" : "Intel i7 8770",
    "memory" : 32
  }
}
{
  "_id" : ObjectId("5f340496e24667d16a7c2d14"),
  "name" : "Max Schwarz",
  "age" : 29,
  "address" : {
    "street" : "Main"
  }
}
{
  "_id" : ObjectId("5f340496e24667d16a7c2d15"),
  "name" : "Manuel Lor",
  "age" : 30,
  "address" : {
    "street" : "Tree"
  }
}
> show collections
products
> cls
```


❖ Performing CRUD (Create, Read, Update, Delete) on Mongo Shell and MongoDB Compass.

➤ Mongo Shell



```
*MongoDB Note.txt - Notepad
File Edit Format View Help
#Sample command in mongo shell:
> mongo
> show dbs
> use shop
> show collections
> db.products.insertOne({name: "A Computer", price: 1229.99, description: "A high quality computer.", details: {cpu: "Intel i7 8770", memory: 32}})
> db.products.find()
> db.products.find().pretty()
> cls
> mongod --help
> mongod --port 27018
> mongo --port 27018
> db.flightData.deleteOne({departureAirport: "TXL"})
> db.flightData.updateOne({distance: 12000}, {$set: {marker: "delete"}})
> db.flightData.updateMany({}, {$set: {marker: "toDelete"}})
> db.flightData.deleteMany({marker: "toDelete"})
> db.flightData.find({intercontinental: true}).pretty()
> db.flightData.find({distance: {$gt: 10000}})
> db.flightData.updateOne({_id: ObjectId("5b97802ce6d2da95ae6406ab")}, {$set: {delayed: true}})
> db.passengers.find({}, {name: 1}).pretty()
> db.passengers.find({}, {name: 1, _id: 0}).pretty()
> db.flightData.updateMany({}, {$set: {status: {description: "on-time", lastUpdated: "1 hour ago", details: {responsible: "Max Schwarzmuller"}}}})
> db.passengers.updateOne({name: "Albert Twostone"}, {$set: {hobbies: ["sports", "cooking"]}})
> db.passengers.findOne({name: "Albert Twostone"}).hobbies
> db.passengers.findOne(hobbies: "sports").pretty()
> db.flightData.findOne({"status.description": "on-time").pretty()
> db.flightData.findOne({"status.details.responsible": "Max Schwarzmuller").pretty()
> db.companies.insertOne({name: "Fresh Apple Inc", isStartup: true, employees: 33, funding: 12345678901234567890, details: {ceo: "Mark Super"}, tags: [{title: "super"}, {title: "perfect"}]},
> db.stats()
> db.numbers.drop()
> db.patients.findOne().diseasesummary
```

❖ Performing CRUD (Create, Read, Update, Delete) on Mongo Shell and MongoDB Compass.

➤ MongoDB Compass

The screenshot displays the MongoDB Compass interface for the 'shop.products' collection. A modal dialog titled 'Insert to Collection shop.products' is open, showing a list of fields to be inserted into a document. The fields are:

- 1. `_id`: ObjectId("5f348886b04707e45830033f")
- 2. `name`: "A Laptop"
- 3. `price`: "1599.9"
- 4. `description`: "new products"
- 5. `detail`: Object
- 6. `model`: "ASUS"
- 7. `cpu`: "intel core i9"
- 8. `hard disk`: "1T"

The dialog has 'CANCEL' and 'INSERT' buttons. Below the dialog, the 'shop.products' collection documents are visible. The first document is:

```
{
  "_id": "ObjectId('5f348496e24667d16a7c2d14')",
  "name": "Max Schwarz",
  "age": 29,
  "address": {}
}
```

The second document is:

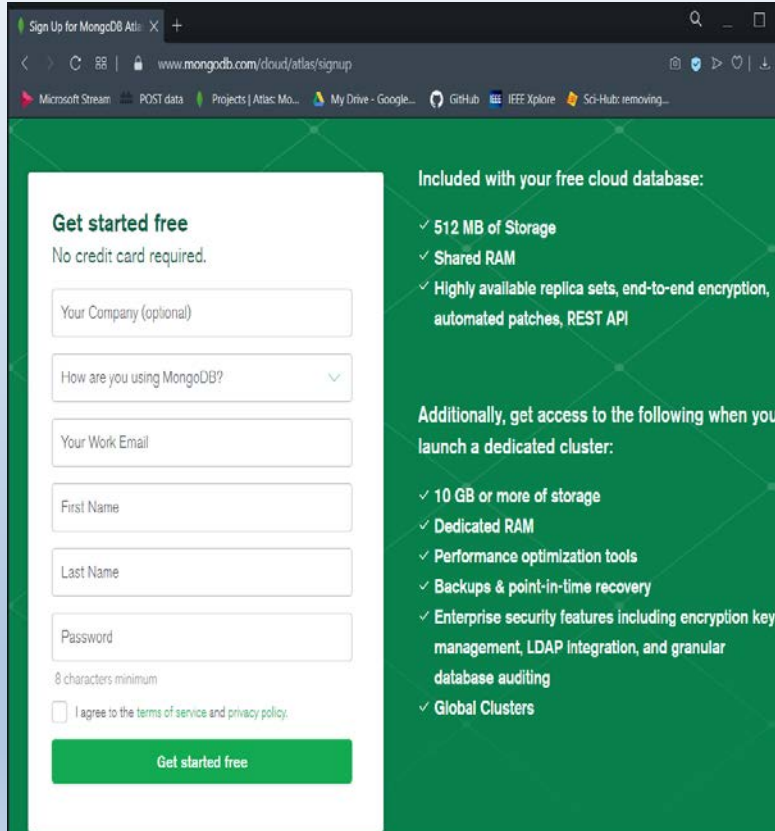
```
{
  "_id": "ObjectId('5f348496e24667d16a7c2d15')",
  "name": "Manuel Lor",
  "age": 30,
  "address": {}
}
```

The third document is:

```
{
  "_id": "ObjectId('5f348886b04707e45830033f')",
  "name": "A Laptop",
  "price": "1599.9",
  "description": "new products",
  "detail": {
    "model": "ASUS",
    "cpu": "intel core i9",
    "hard disk": "1T"
  }
}
```

Arrows indicate the flow of the process: from the 'Insert Document' button in the left sidebar to the 'INSERT' button in the dialog, and from the 'INSERT' button to the 'Edit' and 'Delete' buttons in the document list.

❖ Creating MongoDB Atlas account and Projects.



Sign Up for MongoDB Atlas

www.mongodb.com/cloud/atlas/signup

Get started free
No credit card required.

Your Company (optional)

How are you using MongoDB?

Your Work Email

First Name

Last Name

Password
8 characters minimum

☐ I agree to the terms of service and privacy policy.

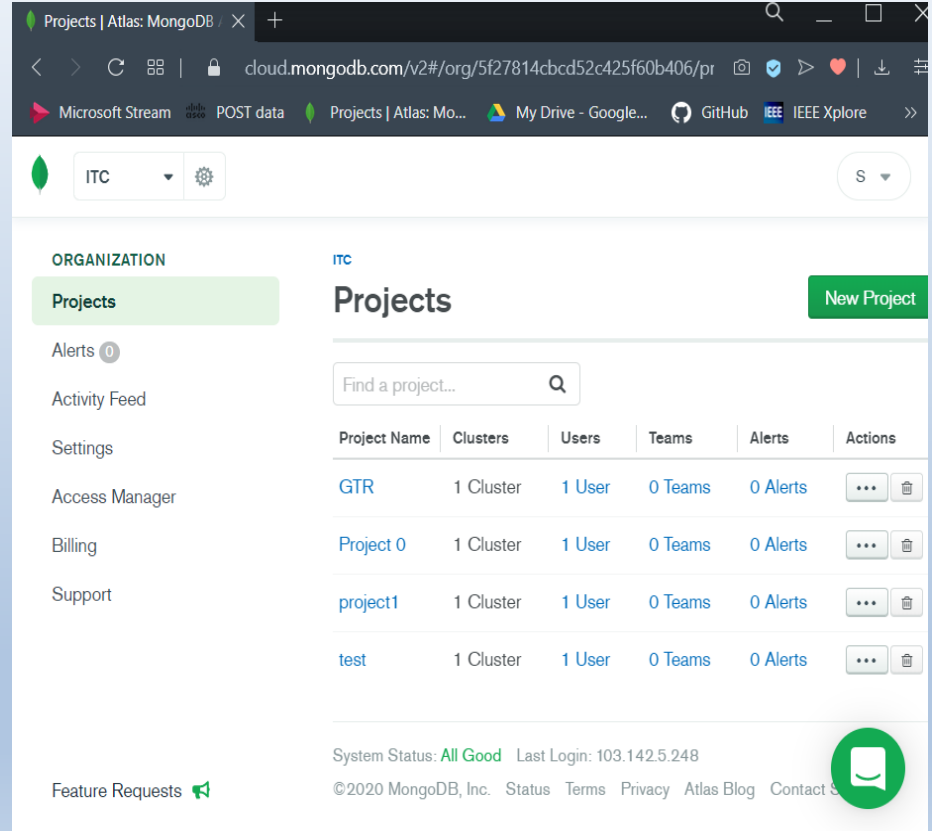
Get started free

Included with your free cloud database:

- ✓ 512 MB of Storage
- ✓ Shared RAM
- ✓ Highly available replica sets, end-to-end encryption, automated patches, REST API

Additionally, get access to the following when you launch a dedicated cluster:

- ✓ 10 GB or more of storage
- ✓ Dedicated RAM
- ✓ Performance optimization tools
- ✓ Backups & point-in-time recovery
- ✓ Enterprise security features including encryption key management, LDAP integration, and granular database auditing
- ✓ Global Clusters



Projects | Atlas: MongoDB

cloud.mongodb.com/v2#/org/5f27814cbcd52c425f60b406/pr

ITC

Projects

New Project

Alerts 0

Activity Feed

Settings

Access Manager

Billing

Support

Find a project...

Project Name	Clusters	Users	Teams	Alerts	Actions
GTR	1 Cluster	1 User	0 Teams	0 Alerts	...
Project 0	1 Cluster	1 User	0 Teams	0 Alerts	...
project1	1 Cluster	1 User	0 Teams	0 Alerts	...
test	1 Cluster	1 User	0 Teams	0 Alerts	...

System Status: All Good Last Login: 103.142.5.248

©2020 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact

❖ Creating MongoDB Atlas account and Projects.

➤ Build Clusters

ITC > NEW PROJECT

DATA STORAGE

Clusters

Triggers

Data Lake

SECURITY

Database Access

Network Access

Advanced

Create a cluster

Choose your cloud provider, region, and

Build a Cluster

Create a cluster

Starting at FREE

Create a cluster

Starting at \$0.08/hr*

*estimated cost \$58.94/month

Create a cluster

Starting at \$0.13/hr*

*estimated cost \$58.94/month

Back

Advanced Configuration Options

Create a Starter Cluster

Welcome to MongoDB Atlas! We've recommended some of our most popular options for your needs. For more information, check our [documentation](#).

Cloud Provider & Region



★ Recommended region ⓘ

NORTH AMERICA

★ N. Virginia (us-east-1) ★

★ Oregon (us-west-2) ★

EUROPE

★ Ireland (eu-west-1) ★

★ Frankfurt (eu-central-1) ★

AUSTRALIA

★ Sydney (ap-southeast-2) ★

Cluster Tier

M0 Sandbox (Shared RAM, 512 MB Storage)

Base hourly rate is for a MongoDB replica set with 3 data bearing servers.

Shared Clusters for development environments and low-traffic applications

Tier	RAM	Storage	vCPU	Base Price
M0 Sandbox	Shared	512 MB	Shared	Free forever
M0 clusters are best for getting started, and are not suitable for production environments.				
500 max connections Low network performance 100 max databases 500 max collections				
M2	Shared	2 GB	Shared	\$9 / MONTH
M5	Shared	5 GB	Shared	\$23 / MONTH

Additional Settings

Select a Version

All clusters launch with the WiredTiger™ storage engine.



FREE

Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

MongoDB 4.2, No Backup

MongoDB 4.2

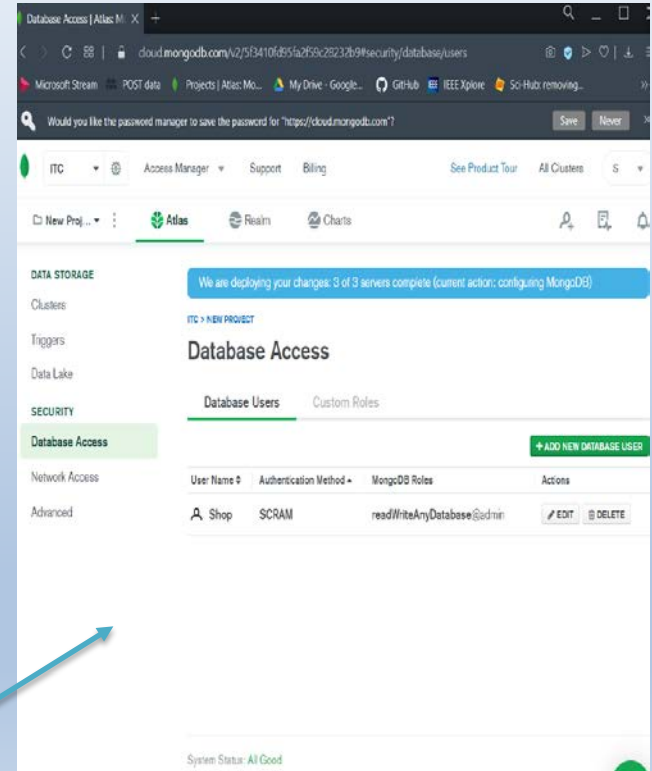
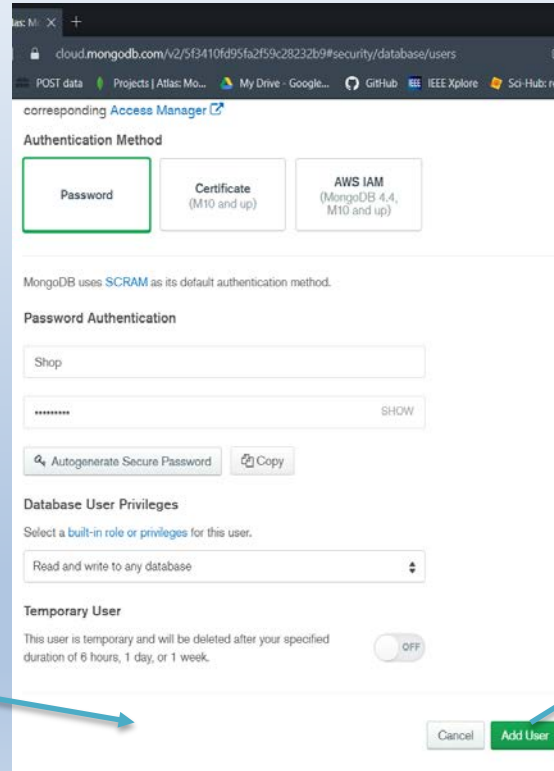
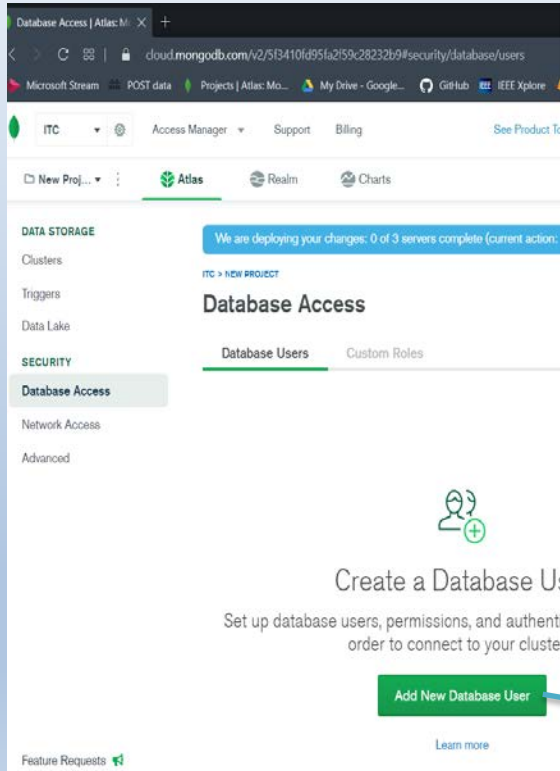
MongoDB 4.2

MongoDB 4.4

Create Cluster

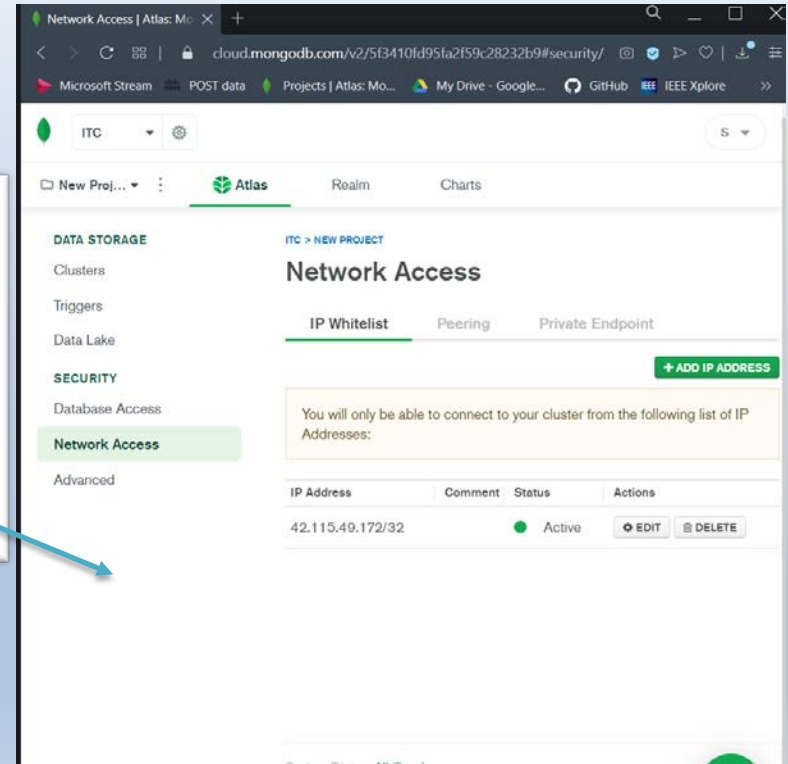
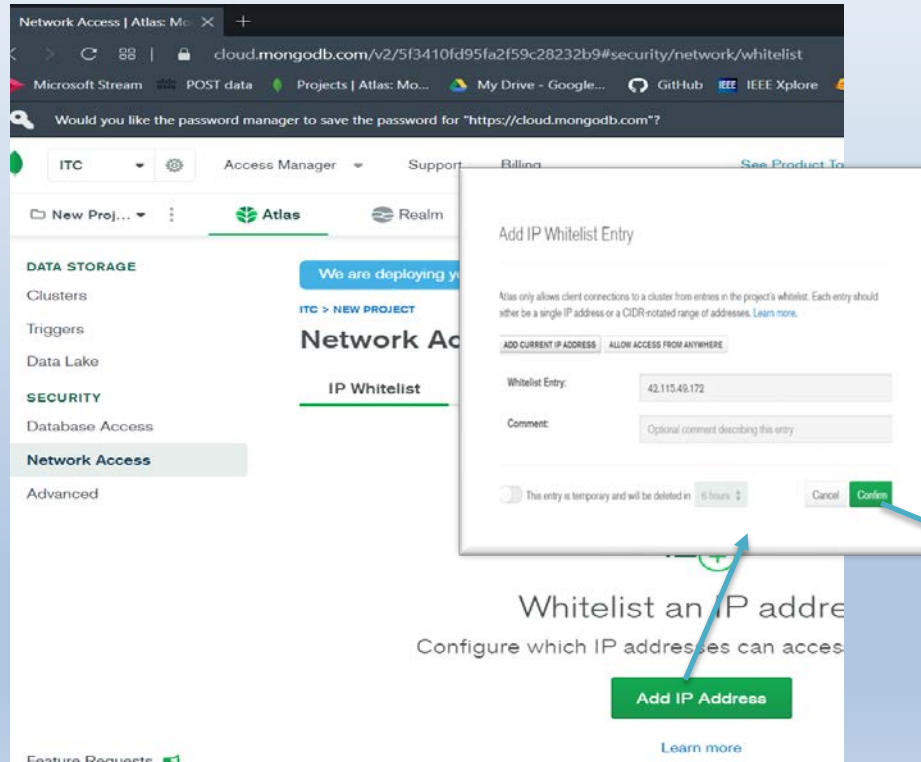
❖ Creating MongoDB Atlas account and Projects.

➤ Create Database User



❖ Creating MongoDB Atlas account and Projects.

➤ Create IP address



❖ Connect MongoDB Atlas to Mongo Shell or MongoDB Compass, and Driver (Node.js).(Cloud)

ITC Access Manager Support Billing See Product Tour All Clusters Sovannroth

New Project Atlas Realm Charts

DATA STORAGE

Clusters

Triggers

Data Lake

SECURITY

Database Access

Network Access

Advanced

ITC > NEW PROJECT

Clusters

Create a New Cluster

Find a cluster...

SANDBOX

Cluster0

Version 4.2.8

CONNECT METRICS COLLECTIONS ...

CLUSTER TIER

M0 Sandbox (General)

REGION

AWS / N. Virginia (us-east-1)

TYPE

Replica Set - 3 nodes

LINKED REALM APP

None linked

Operations R: 0 W: 0 100.0%

Logical Size 0.0 B 512.0 MB max

Connections 0 600 max

Enhance Your Experience

For dedicated throughput, richer metrics and enterprise security options, upgrade your cluster now!

Upgrade

Click here

Connect to Cluster0

✓ Setup connection security Choose a connection method Connect

Choose a connection method [View documentation](#)

Get your pre-formatted connection string by selecting your tool below.

Connect with the mongo shell

Interact with your cluster using MongoDB's interactive Javascript interface

Connect your application

Connect your application to your cluster using MongoDB's native drivers

Connect using MongoDB Compass

Explore, modify, and visualize your data with MongoDB's GUI

Go Back Close

❖ Connect MongoDB Atlas to Mongo Shell or MongoDB Compass, and Driver (Node.js).(Cloud)

Connect to Cluster0

✓ Setup connection security ✓ Choose a connection method Connect

I do not have the mongo shell installed

I have the mongo shell installed

1 Select your mongo shell version

4.2

(To check your shell version, run `mongo --version`)

2 Run your connection string in your command line

Use this connection string in your application:

`mongo "mongodb+srv://cluster0.v97vw.mongodb.net/<dbname>" --username :<username>`

Copy

Replace `<dbname>` with the name of the database that connections will use by default. You will be prompted for the password for the MongoDB user, `Shop`. When entering your password, make sure all special characters are [URL encoded](#).

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

Connect to Cluster0

✓ Setup connection security ✓ Choose a connection method Connect

I do not have MongoDB Compass

I have MongoDB Compass

1 Choose your version of Compass:

1.12 or later

See your Compass version in "About Compass"

2 Copy the connection string, then open MongoDB Compass.

`mongodb+srv://Shop:<password>@cluster0.v97vw.mongodb.net/test`

Copy

MongoDB Compass will auto-detect the connection string you copied. To connect, enter your database username and password into the corresponding fields when prompted. When entering your password, make sure that any special characters are [URL encoded](#).

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

Connect to Cluster0

✓ Setup connection security ✓ Choose a connection method Connect

1 Select your driver and version

DRIVER

Node.js

VERSION

3.6 or later

2 Add your connection string into your application code

☐ Include full driver code example

`mongodb+srv://Shop:<password>@cluster0.v97vw.mongodb.net/<dbname>?ret`

Copy

Replace `<password>` with the password for the `Shop` user. Replace `<dbname>` with the name of the database that connections will use by default. Ensure any option params are [URL encoded](#).

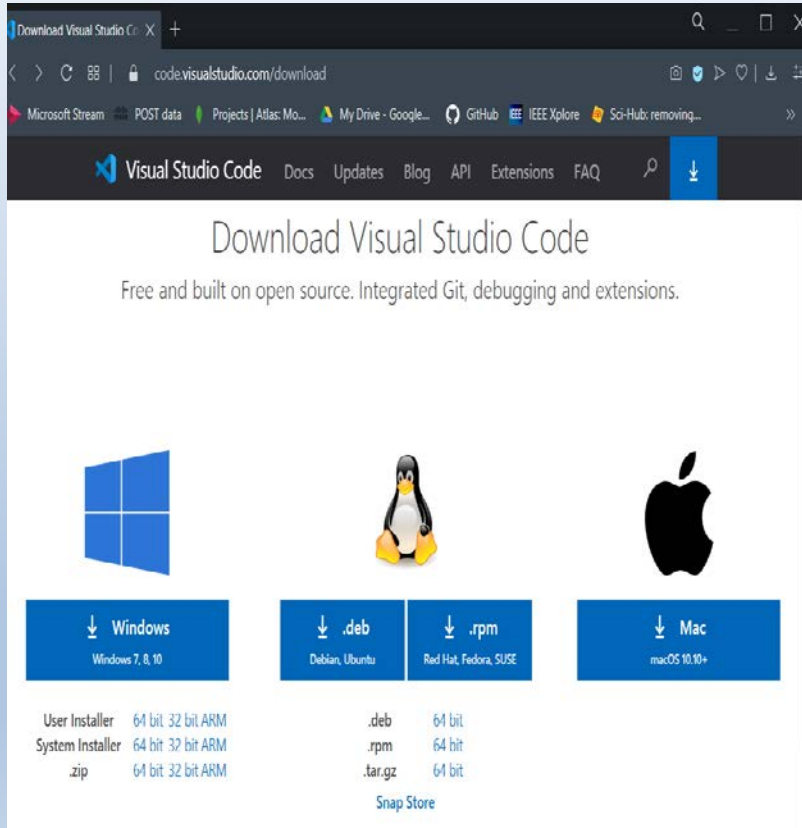
Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

PAGE 15

❖ Installation Visual Studio Code



❖ Connecting Node.js to MongoDB Cluster. (Cloud)

```
backend > JS db.js > mongoDbUrl
1  const mongodb = require('mongodb');
2
3  const MongoClient = mongodb.MongoClient;
4  const mongoDbUrl =
5    'mongodb+srv://Sovannroth:Roth12345@cluster0.rssay.mongodb.net/test?retryWrites=true&w=majority';
6
7  let _db;
8
9  const initDb = callback => {
10    if (_db) {
11      console.log('Database is already initialized!');
12      return callback(null, _db);
13    }
14    MongoClient.connect(mongoDbUrl)
15      .then(client => {
16        _db = client;
17        callback(null, _db);
18      })
19  }
```

Code copy from MongoDB Cluster for application.

```
Sovannroth@DESKTOP-062D89C MINGW64 /d/Programming/MongoDB/shell-to-driver-03-insert-dat
a
$ npm install
npm WARN ajv-keywords@3.2.0 requires a peer of ajv@^6.0.0 but none is installed. You m
st install peer dependencies yourself.
Compiled successfully!

You can now view mongodb-demo in the browser.

Local:    http://localhost:3000/
On Your Network:  http://192.168.139.1:3000/

Note that the development build is not optimized.
To create a production build, use yarn build.

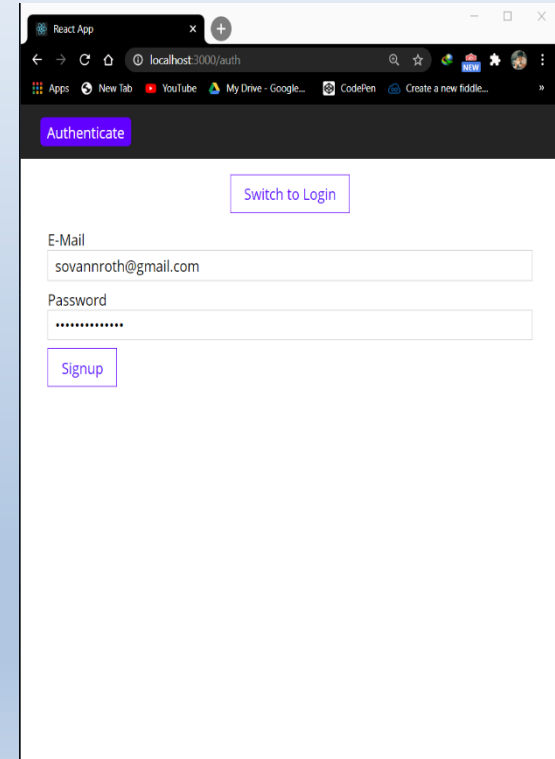
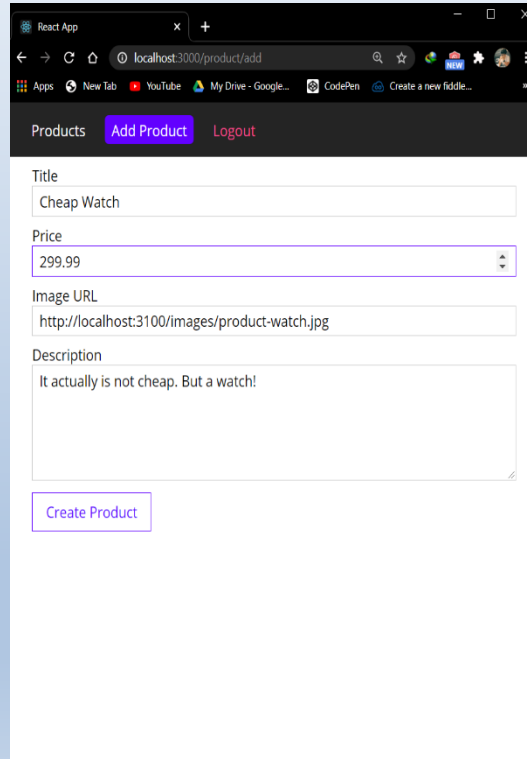
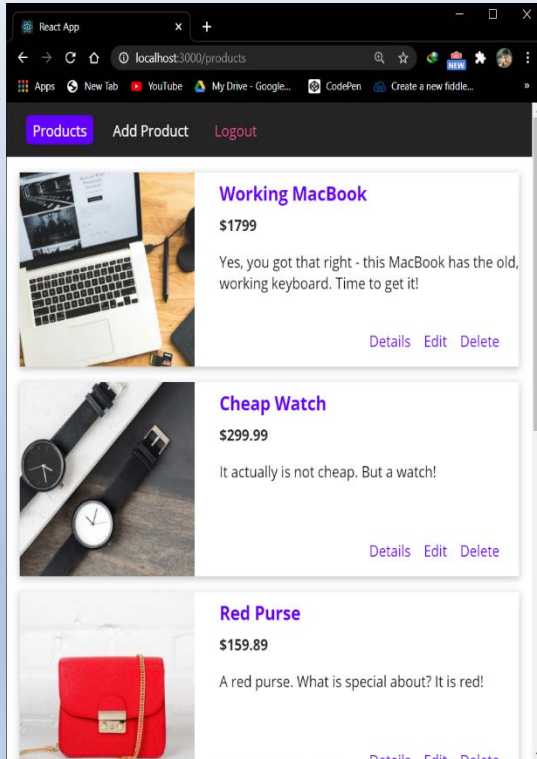
Sovannroth@DESKTOP-062D89C MINGW64 /d/Programming/MongoDB/shell-to-driver-03-insert-dat
a
$ npm run start:server

> mongodb-demo@0.1.0 start:server D:\Programming\MongoDB\shell-to-driver-03-insert-dat
a
> node ./backend/app.js

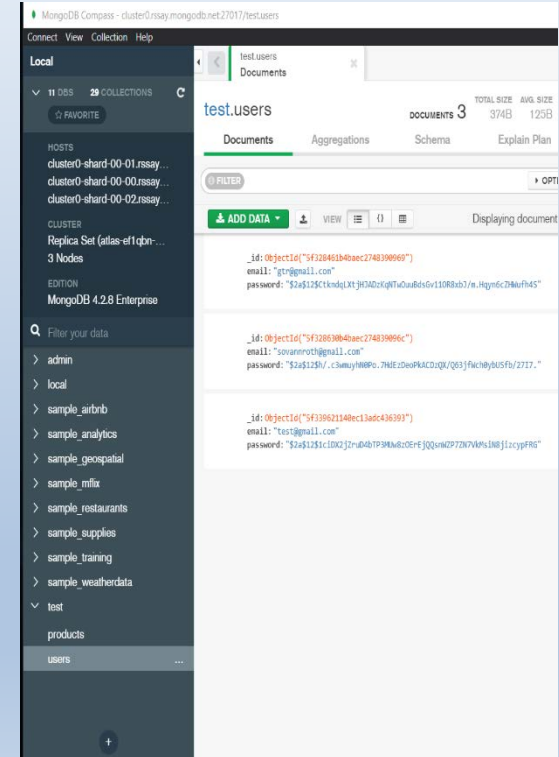
(node:5652) Warning: Accessing non-existent property 'count' of module exports inside c
ircular dependency
(node:5652) Warning: Accessing non-existent property 'findOne' of module exports inside
circular dependency
(node:5652) Warning: Accessing non-existent property 'remove' of module exports inside
circular dependency
(node:5652) Warning: Accessing non-existent property 'updateOne' of module exports insi
de circular dependency
(node:5652) DeprecationWarning: current URL string parser is deprecated, and will be re
moved in a future version. To use the new parser, pass option { useNewUrlParser: true }
to MongoClient.connect.
```

❖ Connecting sample Backend to MongoDB Cluster and storing Products in the Database.

➤ Sample Backend



- Storing products and users signup in the Database.



**Thank
You!**