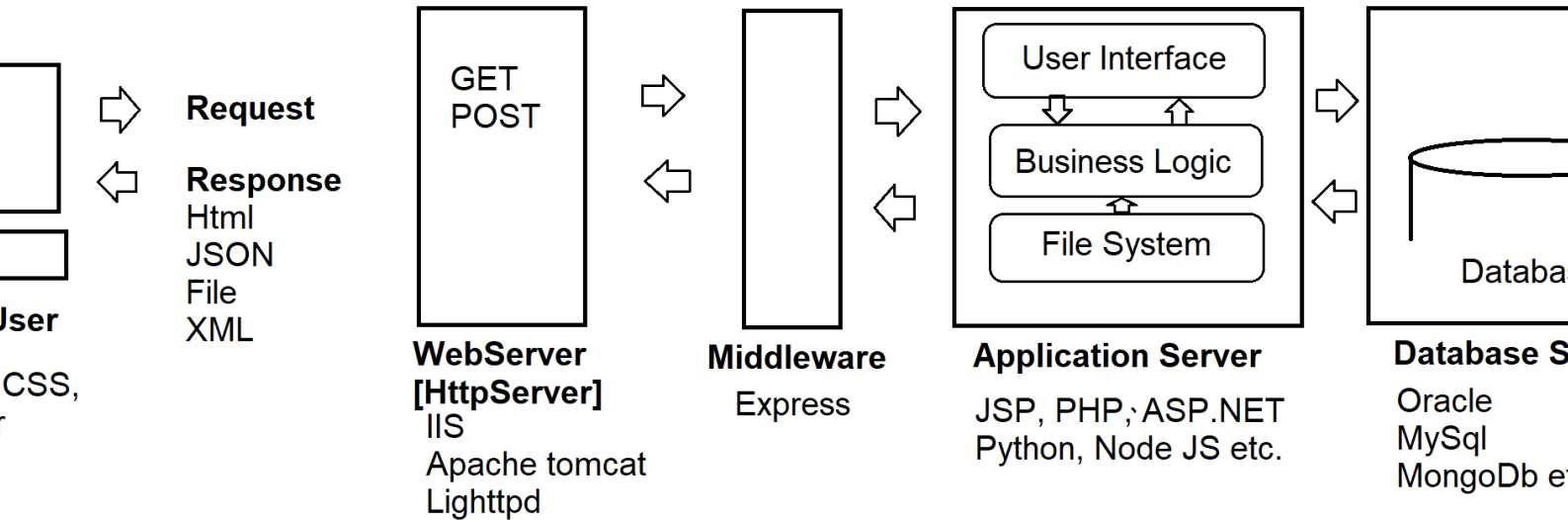


MEAN Stack - for End to End Application

Web Application Architecture



Database:

- MongoDB
 - Document Oriented
 - JavaScript Based
 - Open Source
 - Cross Platform
 - Non-Sql
 - Non-RDBMS

RDBMS	MongoDb
Database	Database
Table	Collection
Row	Document
Column/Field	Field
Join	Embedded Document

- Setup Environment for MongoDB

- Download MongoDB Database Server

<https://www.mongodb.com/try/download/community>

- Start MongoDB Database Server

- Go to “Services.msc” in window run option
- Right Click on “MongoDb” server and Start

- Open MongoDB Client

- Open Command Prompt
- Change to the following and execute the command “mongo.exe”

C:\Program

Files\MongoDB\Server\4.0\bin>mongo.exe

- MongoDB Commands [Case Sensitive]

Command	Description
show dbs	To View the list of all databases.
db	To View the active database.
use	To Switch to existing database or to create a new database. ➤ use new_DatabaseName
db.createCollection(“coll	Create a new Db Table

ectionName")	Ex: db.createCollection("tblcategories") db.createCollection("tblproducts")
show collections	To View the list of Db tables
db.collectionName.insert({ }) db.collectionName.insert([{} , {}])	To insert one or many records into table (collection). Ex: db.tblcategories.insert([{CategoryId:1, CategoryName:'Electronics' }, {CategoryId:2, CategoryName:'Footwear' }]) Ex: db.tblproducts.insert([{ProductId: 1, Name: 'Samsung TV', Price: 45000.55, CategoryId: 1}, {ProductId:2, Name: 'Nike Casuals', Price: 4000.55, CategoryId: 2}, {ProductId: 3, Name: 'JBL Speaker', Price: 5000.55,

	CategoryId: 1}, {ProductId: 4, Name: 'Lee Cooper Boot', Price: 3000.55, CategoryId: 2},])
db.collectionName.find().pretty()	To View the records in table. Ex: db.tblcategories.find().pretty()

Server-Side Node JS

- Node JS is an open source, cross platform JavaScript Based Server-Side Scripting.
- No Buffering
- Asynchronous
- Single Threaded
- Node JS uses “JavaScript Programs” server-side [“.js”]

Node JS requires MongoDB Library to handle communication with database.

- Create a new folder in your projects “ServerSide”
- Install “Mongodb” library for your workspace.
C:\Angular-workspace>npm install mongodb
- Create a new JavaScript file “testconnection.js” in “ServerSide” folder

Testconnection.js

```
//import MongoDB Library
var MongoClient = require("mongodb")
.MongoClient;

//Mongo Connection
var url = "mongodb://127.0.0.1:27017";

mongoClient.connect(url, function(er
r, clientObj){
    if(!err) {
        console.log(`Connected to Mo
ngoDB` );
    } else {
        console.log(err);
    }
})
```

- Compile and Run Node JS Program by using Node Compile

Command Prompt or Terminal:

➤ node testconnection.js

Ex: To Read Records from a Database Table

```
//import MongoDB Library
var MongoClient = require("mongodb").MongoClient;

//Mongo Connection
var url = "mongodb://127.0.0.1:27017";

mongoClient.connect(url, function(err, clientObj){
    if(!err) {
        var database = clientObj.db("apidb");
        database.collection("tblproducts").find().toArray(function(err, documents){
            if(!err){
                console.log(documents);
            } else {
                console.log(err);
            }
        })
    }
})
```

```
    } else {  
        console.log(err);  
    }  
}))
```

Ex: Inserting record into Mogobd Collection

```
//import MongoDB Library  
var MongoClient = require("mongodb").MongoClient;  
  
//Mongo Connection  
var url = "mongodb://127.0.0.1:27017";  
  
mongoClient.connect(url, function(err, clientObj){  
    if(!err) {  
        var database = clientObj.db("apidb");  
        var data = {  
            ProductId: 5,  
            Name: 'EarPods',  
            Price: 5600.66,  
            Category: 1  
        };  
    }  
});
```

```
        database.collection("tblproducts").insert(d
ata, function(err, result){
            if(!err) {
                console.log("Record Inserted");
            } else {
                console.log(err);
            }
        })
    } else {
        console.log(err);
    }
})
```

Middleware

- Install “Express” middleware in your workspace
- Middleware allows communication between client and server-side application
- You can create an API that handle communication.

Ex:

API.JS

```
var express = require("express");
var app = express();
```



```
app.get("/", function(req, res){
    res.send("Welcome to API");
})

app.listen(8080);
console.log("Server Started : http://127.0.0.1:8080");
```

Business Layer - API

[Express – Node JS]

You need Packages

- npm install mongodb : To connect with MongoDB Database
- npm install express : To Create server-side API, which handles GET, POST etc.
- npm install body-parser : Converts the data into JSON

```
// Import Library
```

```
var MongoClient = require("mongodb").MongoClient;
```

```
var express = require("express");
```

```
var bodyParser = require("body-parser");
```

```
// Configure Connection String
var url = "mongodb://127.0.0.1:27017";

// Configure Middleware

var app = express();

// Configure Body Parser
app.use(bodyParser.urlencoded({
  extended: true
}));
app.use(bodyParser.json());

//Configure CORS [Cross Origin Resource Sharing]
app.use(function(req, res, next){
  res.header("Access-Control-Allow-Origin","*");
  res.header("Access-Control-Allow-Headers","Origin, X-
Requested-With, Content-Type, Accept");
  res.header("Access-Control-Allow-
Methods","GET","POST","PUT");
  next();
});
```

```
//Configure API Methods
```

```
app.get("/getproducts", function(req, res){  
    mongoClient.connect(url, function(err, clientObj){  
        if(!err) {  
            var database = clientObj.db("apidb");  
            database.collection("tblproducts").find().toArray(function(err  
            , documents){  
                if(!err){  
                    res.send(documents);  
                } else {  
                    console.log(err);  
                }  
            })  
        } else {  
            console.log(err);  
        }  
    })  
})  
  
app.post("/addProducts", function(req, res){  
    mongoClient.connect(url, function(err, clientObj){  
        if(!err) {
```

```
var database = clientObj.db("apidb");
var data = {
    ProductId: req.body.ProductId,
    Name: req.body.Name,
    Price: req.body.Price,
    CategoryId: req.body.CategoryId
};
database.collection("tblproducts").insert(data,
function(err, result){
    if(!err){
        console.log("Record Inserted");
    } else {
        console.log(err);
    }
})
} else {
    console.log(err);
}
})
})

app.listen(8080);
```

```
console.log("Server Started: http://127.0.0.1:8080");
```