**What is MongoDB**

* MongoDB stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time
* The document model maps to the objects in your application code, making data easy to work with
* Ad hoc queries, indexing, and real time aggregation provide powerful ways to access and analyse your data
* MongoDB is a distributed database at its core, so high availability, horizontal scaling, and geographic distribution are built in and easy to use

When all of the other components of your networked application are designed to be fast and seamless, your data shouldn’t be the bottleneck—and MongoDB is able to meet new data challenges that are difficult (if not impossible) to accomplish well with relational databases. MongoDB enables you to:

1. Store large volumes of data that often have little to no structure. Relational databases store structured data like a phonebook. But for growing, unstructured data—for example, a customer’s preferences, location, past purchases, and Facebook likes—a NoSQL database sets no limits, and allows you to add different types of data as your needs change. Because MongoDB is flexible and document-based, you can store these JSON-like binary data points (referred to as BSON) in one place without having to define what “types” of data those are in advance.
2. Make the most of cloud computing and storage. Cloud-based storage is an excellent cost-saving solution, but requires data to be easily spread across multiple servers to scale up. MongoDB can load a high volume of data and give you lots of flexibility and availability in a cloud-based environment, with built-in sharing solutions that make it easy to partition and spread out data across multiple servers.
3. Develop and release quickly. If you’re developing within two-week Agile sprints, cranking out quick iterations, or needing to make frequent updates to the data structure without a lot of downtime between versions, modifying a relational database will slow you down. With MongoDB’s dynamic schemas, you can try new things, and fast. Your data doesn’t need to be prepped ahead of time, and your team can incorporate anything new, quickly, and at a lower cost.
4. Scale database architecture efficiently and inexpensively. With MongoDB, it’s easy to spread data out across commodity hardware on-site or in the cloud without needing additional software.

**NoSql**

A NoSQL (originally referring to "non SQL" or "non-relational")database provides a mechanism for storage and retrieval of data that is modelled in means other than the tabular relations used in relational databases.

NoSQL databases have risen to the occasion when it comes to unstructured modern data and the kind of real-time analytics we expect from it. Depending on your own unique business needs, MongoDB might be just the right solution for you. In this article, we’ll cover the basics of NoSQL databases and some of the challenges of modern big data analysis that MongoDB is particularly well-suited to tackle.

**Connecting to Java**

// 1. Connect to MongoDB instance running on localhost

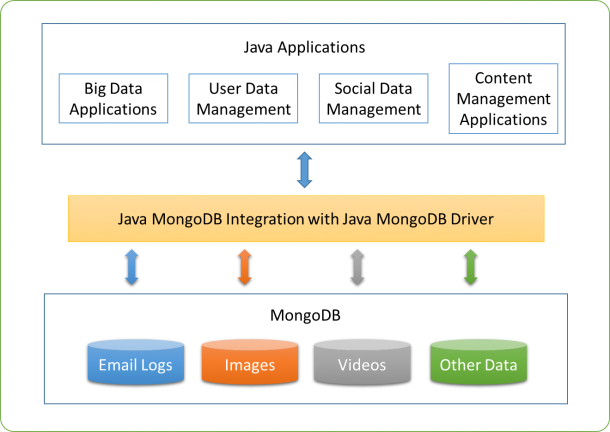
MongoClient mongoClient = new MongoClient();

//2. Access database named 'test'

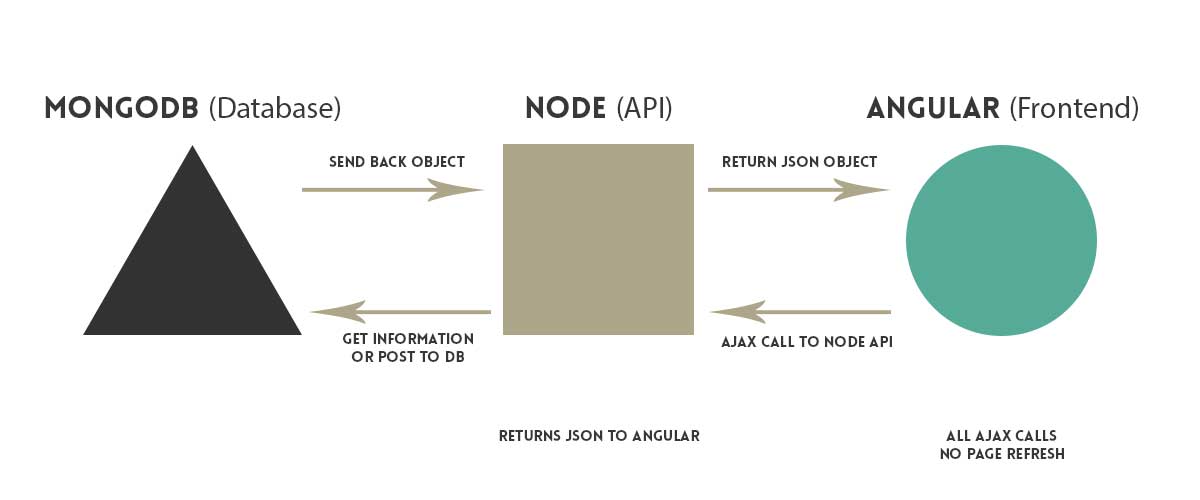
MongoDatabase database = mongoClient.getDatabase("test");

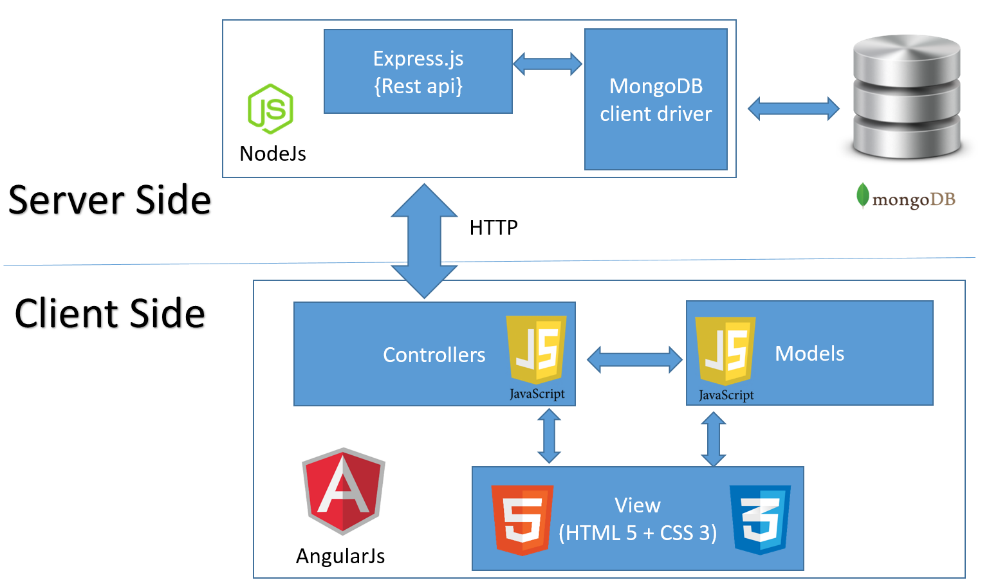
//3. Access collection named ‘insurance’

MongoCollection collection = database.getCollection("insurance");



**MongoDB with Angular**





**Companies which use MongoDB**

