System Documentation

October 2016

Version 1.0

Table of Contents

[**Setting Up MongoDB (Database) and Buy and Sell Project 7**](#_Toc463895060)

[**Install MongoDB Community Edition on Red Hat Enterprise or CentOS Linux 10**](#_Toc463895061)

[**Overview 10**](#_Toc463895062)

[**Packages 10**](#_Toc463895063)

[**Init Scripts 11**](#_Toc463895064)

[**Install MongoDB Community Edition 11**](#_Toc463895065)

[Configure the package management system (yum). 12](#_Toc463895066)

[Install the MongoDB packages and associated tools. 13](#_Toc463895067)

[**Run MongoDB Community Edition 13**](#_Toc463895068)

[Prerequisites 13](#_Toc463895069)

[Procedure 14](#_Toc463895070)

[**Uninstall MongoDB Community Edition 16**](#_Toc463895071)

[Stop MongoDB. 16](#_Toc463895072)

[Remove Packages. 16](#_Toc463895073)

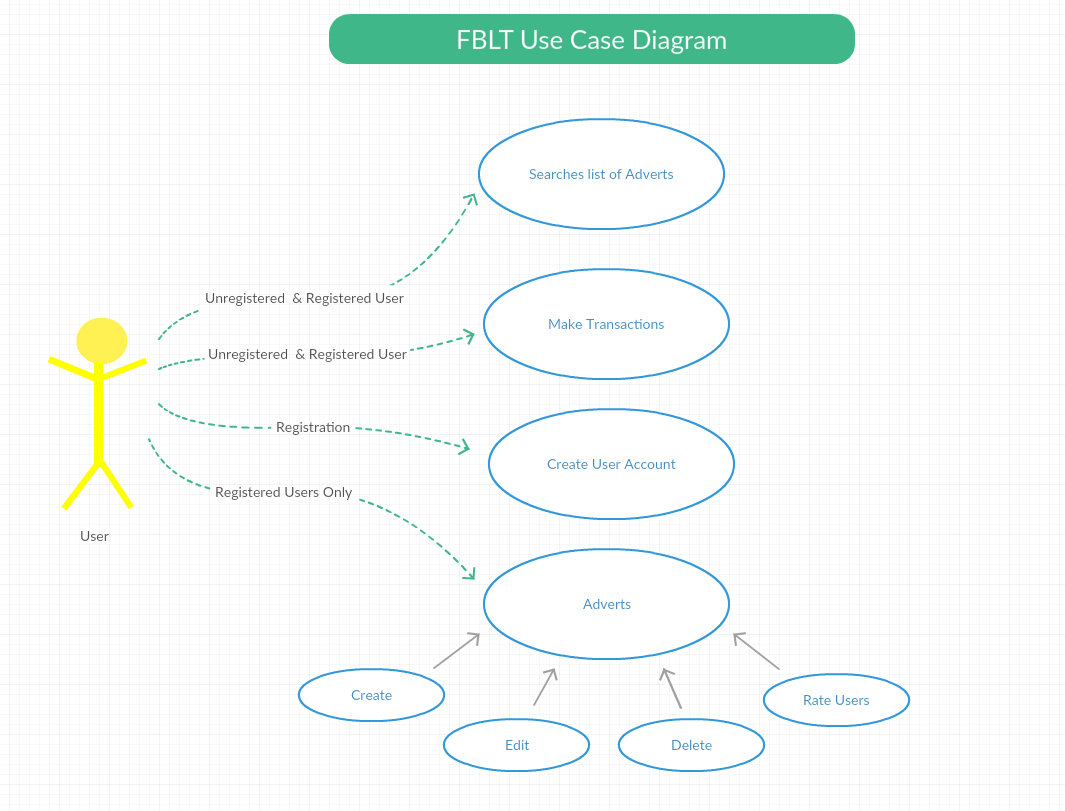
[Remove Data Directories. 16](#_Toc463895074)

1. **Source Code**

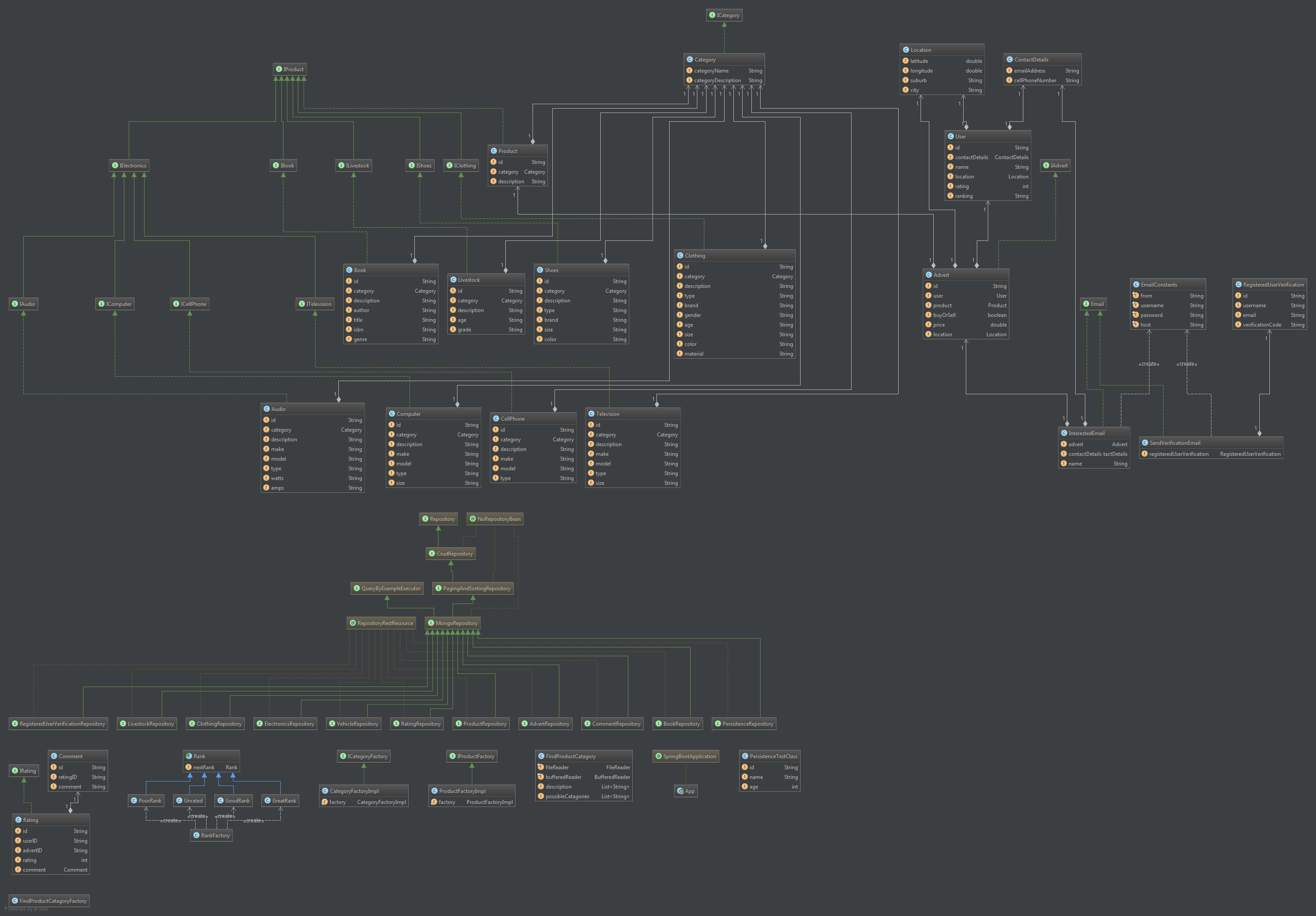
Due to the nature of the Buy and Sell Project being open source the source code can be found on github <https://github.com/FerinT/FBLT> and on our official website which will be created for the purpose of hosting the Buy and Sell Project source code and compiled code (binaries) for download.

1. **Design Documentation**

2..1 Use-case Diagram



2.2 Class Diagram



C:\Users\Ferin\Desktop\Dos 2\Dos\ERD.png **2.2 Entity Relational Diagram**

3. Database schema

Our system uses mongoDB which is based on NoSQL. This database schema differs greatly from your typical SQL based database. mongoDB is based on the concept of dynamic schemas. This mean that unlike with an SQL database the schema does not have to be defined beforehand. This fits perfect into an agile environment where each iteration possibly changes the structure of various features.

MongoDB is built to allow the insertion of data without a predefined schema. Dissimilar data can be stored together.

We are using a Document database which is a form of NoSQL database that does not use the traditional table-and-row model instead all relevant data (Collections) are stored in a single document such as JSON and XML.

4. Testing data

We’ve attached a file which contains test data for the Mongo Database.

To run load the information into the database, you must have Mongo DB installed as a pre-requisite.

Open up a terminal or windows command prompt at the directory you unzipped the FBLT folder to. Then simply run “mongorestore dump” to load test data from the dump folder extracted, int your local database.

5. Explanation of sticky areas

MongoDB Database**:**

One of the tricky parts of putting this system together was the use of MongoDB as the underlying database for this system. It needed some careful planning as to how relationships were to be mapped in the database, as MongoDB is a schema less database. Which means that there are no tables to relate different objects, such as value objects, to one another by means of primary and foreign keys.

Therefore we needed to make a decision on how to model this inside the database and we decided to store the value objects within each document inside the database collections and not separate them into different collections.

This choice was made with database query efficiency in mind. MongoDB does not support joins like other relational databases do so it would take 3 queries to access 3 different collections when using Mongo as opposed to 1 query using a join in a relational table.

**6. Implementation Instructions**

**6.1 Start-up, backup and shutdown procedures**

# Setting Up MongoDB (Database) and Buy and Sell Project

***Note: The below steps assumes that you have successfully compiled the source code and generated the jar file in the target directory or have downloaded the jar file from the official site for the Buy and Sell Project. If not please read the user manual to see how to compile the Buy and Sell Project.***

Setting up The Database

1. **Start-up For Database On Server-Side**

*Note: This step is assumed that you have installed MongoDB on a RedHat Linux System.*

The MongoDB database can be started by running the following command:

**1.1 *sudo service mongod start***

2**. Stopping The Database On Server-Side**

The MongoDB database can be stopped by issuing the following command:

**2.1 *sudo service mongod stop***

3. **Backup MongoDB**

Backing up your database information is crucial due to the fact that the Buy and Sell Project heavily depends on the database for many of its inner workings. Thankfully backing up your database with MongoDB is as easy as ABC.

*Note: Replace <insert your path here> with an actual path where to save the backup file.*

*e.g. /home/peter/Desktop*

**3.1 *mongodump --out <insert your path here>***

4. **Restoring Your Database Backup**

You can restore your saved backup file issung the following:

*Note: <insert directory of backup> refers to the folder directory where the backup file is stored. E.g. /home/peter/Desktop/*

**4.1  *mongorestore <insert directory of backup>***

Setting Up The Buy and Sell Project

Once the database has been successfully installed you can then move on to the starting up of the Buy and Sell Project.

1. **Starting The Buy and Sell Project**

Running the Buy and Sell Project is simple just issue the following command on the location where the jar file is located for the Buy and Sell Project this should be <Buy and Sell Project name>.jar

**1.1 java –jar <project name>.jar**

Once the Buy and Sell Project has completed running, you can then use your browser to view the Buy and Sell Project by typing in localhost:8080 on your local machine or externally <your external IP>:8080.

Note: Default port is 8080.

1. **Stopping The Buy and Sell Project**

The Buy and Sell Project can be stopped in numerous ways such as:

**Rebooting The Server Machine**

**Killing the running service by shutting down the service port default port.**

**Killing the running Buy and Sell Project either by pressing CTRL + C**

1. **Backup and Restoring**

There is no need to backup or restore the Buy and Sell Project as all the information collected by the Buy and Sell Project is stored on the MongoDB database.

**6.2 Database implementation detail**

Install MongoDB Community Edition on Red Hat Enterprise or CentOS Linux

What to know before starting

We use a Red Hat Linux Server to host our back-end for our website. But if you would like to Install MongoDB on other operating systems like Windows or OS X go to: <https://docs.mongodb.com/manual/installation/> for installation details.

## Overview

Use this tutorial to install MongoDB Community Edition on Red Hat Enterprise Linux or CentOS Linux versions 6 and 7 using .rpm packages. While some of these distributions include their own MongoDB packages, the official MongoDB Community Edition packages are generally more up to date.

PLATFORM SUPPORT

This installation guide only supports 64-bit systems. See [Platform Support](https://docs.mongodb.com/manual/release-notes/3.0-compatibility/#compatibility-platform-support) for details.

MongoDB 3.2 deprecates support for Red Hat Enterprise Linux 5.

Packages

MongoDB provides officially supported packages in their own repository. This repository contains the following packages:

|  |  |
| --- | --- |
| mongodb-org | A metapackage that will automatically install the four component packages listed below. |
| mongodb-org-server | Contains the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) daemon and associated configuration and init scripts. |
| mongodb-org-mongos | Contains the [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#bin.mongos) daemon. |
| mongodb-org-shell | Contains the [mongo](https://docs.mongodb.com/manual/reference/program/mongo/#bin.mongo) shell. |
| mongodb-org-tools | Contains the following MongoDB tools: [mongoimport](https://docs.mongodb.com/manual/reference/program/mongoimport/#bin.mongoimport) [bsondump](https://docs.mongodb.com/manual/reference/program/bsondump/#bin.bsondump), [mongodump](https://docs.mongodb.com/manual/reference/program/mongodump/#bin.mongodump),[mongoexport](https://docs.mongodb.com/manual/reference/program/mongoexport/#bin.mongoexport), [mongofiles](https://docs.mongodb.com/manual/reference/program/mongofiles/#bin.mongofiles), [mongooplog](https://docs.mongodb.com/manual/reference/program/mongooplog/#bin.mongooplog), [mongoperf](https://docs.mongodb.com/manual/reference/program/mongoperf/#bin.mongoperf), [mongorestore](https://docs.mongodb.com/manual/reference/program/mongorestore/#bin.mongorestore),[mongostat](https://docs.mongodb.com/manual/reference/program/mongostat/#bin.mongostat), and [mongotop](https://docs.mongodb.com/manual/reference/program/mongotop/" \l "bin.mongotop" \o "mongotop). |

The default /etc/mongod.conf configuration file supplied by the packages have bind\_ip set to127.0.0.1 by default. Modify this setting as needed for your environment before initializing a [replica set](https://docs.mongodb.com/manual/reference/glossary/#term-replica-set).

## Init Scripts

The mongodb-org package includes various [init scripts](https://docs.mongodb.com/manual/reference/glossary/" \l "term-init-script), including the init script/etc/rc.d/init.d/mongod. You can use these scripts to stop, start, and restart daemon processes.

The package configures MongoDB using the /etc/mongod.conf file in conjunction with the init scripts. See the [Configuration File](https://docs.mongodb.com/manual/reference/configuration-options/) reference for documentation of settings available in the configuration file.

As of version 3.2.10, there are no init scripts for [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#bin.mongos). The [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#bin.mongos) process is used only in [sharding](https://docs.mongodb.com/manual/sharding/). You can use the mongod init script to derive your own [mongos](https://docs.mongodb.com/manual/reference/program/mongos/#bin.mongos) init script for use in such environments. See the[mongos](https://docs.mongodb.com/manual/reference/program/mongos/#bin.mongos) reference for configuration details.

The default /etc/mongod.conf configuration file supplied by the packages have bind\_ip set to127.0.0.1 by default. Modify this setting as needed for your environment before initializing a [replica set](https://docs.mongodb.com/manual/reference/glossary/#term-replica-set).

## Install MongoDB Community Edition

This installation guide only supports 64-bit systems.

1

### Configure the package management system (yum).

Create a /etc/yum.repos.d/mongodb-org-3.2.repo file so that you can install MongoDB directly, using yum.

*Changed in version 3.0:*MongoDB Linux packages are in a new repository beginning with 3.0.

#### For the latest stable release of MongoDB

Use the following repository file:

[mongodb-org-3.2]

name=MongoDB Repository

baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/3.2/x86\_64/

gpgcheck=1

enabled=1

gpgkey=https://www.mongodb.org/static/pgp/server-3.2.asc

#### For versions of MongoDB earlier than 3.0

To install the packages from an earlier [release series](https://docs.mongodb.com/manual/release-notes/#release-version-numbers), such as 2.4 or 2.6, you can specify the release series in the repository configuration. For example, to restrict your system to the 2.6 release series, create a /etc/yum.repos.d/mongodb-org-2.6.repo file to hold the following configuration information for the MongoDB 2.6 repository:

[mongodb-org-2.6]

name=MongoDB 2.6 Repository

baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86\_64/

gpgcheck=0

enabled=1

You can find .repo files for each release [in the repository itself](https://repo.mongodb.org/yum/redhat/). Remember that odd-numbered minor release versions (e.g. 2.5) are development versions and are unsuitable for production use.

### 2 Install the MongoDB packages and associated tools.

When you install the packages, you choose whether to install the current release or a previous one. This step provides the commands for both.

To install the latest stable version of MongoDB, issue the following command:

sudo yum install -y mongodb-org

To install a specific release of MongoDB, specify each component package individually and append the version number to the package name, as in the following example:

sudo yum install -y mongodb-org-3.2.10 mongodb-org-server-3.2.10 mongodb-org-shell-3.2.10 mongodb-org-mongos-3.2.10 mongodb-org-tools-3.2.10

You can specify any available version of MongoDB. However yum will upgrade the packages when a newer version becomes available. To prevent unintended upgrades, pin the package. To pin a package, add the following exclude directive to your /etc/yum.conf file:

exclude=mongodb-org,mongodb-org-server,mongodb-org-shell,mongodb-org-mongos,mongodb-org-tools

## Run MongoDB Community Edition

### Prerequisites

#### Configure SELinux

IMPORTANT

If you are using SELinux, you must configure SELinux to allow MongoDB to start on Red Hat Linux-based systems (Red Hat Enterprise Linux or CentOS Linux).

To configure SELinux, administrators have three options:

* If SELinux is in enforcing mode, enable access to the relevant ports that the MongoDB deployment will use (e.g. 27017). See [Default MongoDB Port](https://docs.mongodb.com/manual/reference/default-mongodb-port/) for more information on MongoDB’s default ports. For default settings, this can be accomplished by running
* semanage port -a -t mongod\_port\_t -p tcp 27017
* Disable SELinux by setting the SELINUX setting to disabled in /etc/selinux/config.
* SELINUX=disabled

You must reboot the system for the changes to take effect.

* Set SELinux to permissive mode in /etc/selinux/config by setting the SELINUX setting topermissive.
* SELINUX=permissive

You must reboot the system for the changes to take effect.

You can instead use setenforce to change to permissive mode. setenforce does not require a reboot but is **not** persistent.

Alternatively, you can choose not to install the SELinux packages when you are installing your Linux operating system, or choose to remove the relevant packages. This option is the most invasive and is not recommended.

#### Data Directories and Permissions

WARNING

On RHEL 7.0, if you change the data path, the default SELinux policies will prevent [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) from having write access on the new data path if you do not change the security context.

The MongoDB instance stores its data files in /var/lib/mongo and its log files in /var/log/mongodbby default, and runs using the mongod user account. You can specify alternate log and data file directories in/etc/mongod.conf. See [systemLog.path](https://docs.mongodb.com/manual/reference/configuration-options/" \l "systemLog.path" \o "systemLog.path) and [storage.dbPath](https://docs.mongodb.com/manual/reference/configuration-options/" \l "storage.dbPath" \o "storage.dbPath) for additional information.

If you change the user that runs the MongoDB process, you **must** modify the access control rights to the/var/lib/mongo and /var/log/mongodb directories to give this user access to these directories.

### Procedure

1

#### Start MongoDB.

You can start the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) process by issuing the following command:

sudo service mongod start

2

#### Verify that MongoDB has started successfully

You can verify that the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) process has started successfully by checking the contents of the log file at /var/log/mongodb/mongod.log for a line reading

[initandlisten] waiting for connections on port <port>

where <port> is the port configured in /etc/mongod.conf, 27017 by default.

You can optionally ensure that MongoDB will start following a system reboot by issuing the following command:

sudo chkconfig mongod on

3

#### Stop MongoDB.

As needed, you can stop the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) process by issuing the following command:

sudo service mongod stop

4

#### Restart MongoDB.

You can restart the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) process by issuing the following command:

sudo service mongod restart

You can follow the state of the process for errors or important messages by watching the output in the/var/log/mongodb/mongod.log file.

5

#### Begin using MongoDB.

To help you start using MongoDB, MongoDB provides [Getting Started Guides](https://docs.mongodb.com/manual/#getting-started) in various driver editions. See [Getting Started](https://docs.mongodb.com/manual/#getting-started) for the available editions.

Before deploying MongoDB in a production environment, consider the [Production Notes](https://docs.mongodb.com/manual/administration/production-notes/) document.

Later, to stop MongoDB, press Control+C in the terminal where the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) instance is running.

## Uninstall MongoDB Community Edition

To completely remove MongoDB from a system, you must remove the MongoDB applications themselves, the configuration files, and any directories containing data and logs. The following section guides you through the necessary steps.

WARNING

This process will completely remove MongoDB, its configuration, and all databases. This process is not reversible, so ensure that all of your configuration and data is backed up before proceeding.

1

### Stop MongoDB.

Stop the [mongod](https://docs.mongodb.com/manual/reference/program/mongod/" \l "bin.mongod" \o "mongod) process by issuing the following command:

sudo service mongod stop

2

### Remove Packages.

Remove any MongoDB packages that you had previously installed.

sudo yum erase $(rpm -qa | grep mongodb-org)

3

### Remove Data Directories.

Remove MongoDB databases and log files.

sudo rm -r /var/log/mongodb

sudo rm -r /var/lib/mongo