**MailCrafter Development Environment Setup**

**1. Clone the Code Repositories**

You need to clone code from the following three repositories:

* MailCrafter.MVC
* MailCrafter.Core
* MailCrafter.Worker

Alternatively, you can clone the root module **MailCrafter.PRN**, which contains all submodules mentioned above. If you clone from the root module, note that the three submodules will be in a detached head state. Ensure you pull the latest code from the main branch for each repository.

**2. Configure the Application**

Place the provided configuration file in the following path:

C:\MailCrafter\Development\Core\appsettings.Development.json

Update necessary configurations such as:

* Database connection URI
* RabbitMQ settings
* Encryption key

Ensure these configurations are appropriate for your development environment.

**3. Install and Set Up Docker**

Download and install [Docker Desktop](https://www.docker.com/products/docker-desktop/).

Pull the RabbitMQ image using the following command:

docker pull rabbitmq:3-management

**4. (Optional) Set Up MongoDB with Docker**

You can instantiate a MongoDB instance using Docker as a storage layer for MailCrafter. Use the following command to pull the MongoDB image:

docker pull mongodb/mongodb-community-server:latest

**Important:** Docker containers use ephemeral storage, meaning data will be lost if the container is deleted. To persist data, configure a volume by mounting a persistent storage directory when creating the container.

**5. (Alternative) Use MongoDB Without Docker**

If you prefer not to use Docker for MongoDB, you can:

* Download and set up [Mongoose](https://mongoosejs.com/)
* Use [MongoDB Atlas](https://www.mongodb.com/atlas) (free-tier is sufficient)

With MongoDB Atlas, you do not need to worry about data loss or persistence.

**Running the Development Environment**

**1. Start RabbitMQ and MongoDB Instances**

**Run RabbitMQ**

docker run -d --name rabbitmq -p 5672:5672 -p 15672:15672 rabbitmq:3-management

This starts a RabbitMQ instance accessible on:

* Port 5672 (for application connection)
* Port 15672 (for UI monitoring)

**(Optional) Run MongoDB**

docker run --name mongodb -p 27017:27017 -d mongodb/mongodb-community-server:latest

To persist data, ensure you specify a volume when running MongoDB.

**Note:** You can customize the ports according to your preference. If you change them, make sure to update the configuration file accordingly.

**2. Run the MailCrafter Applications**

Start the MVC and Worker projects. Once running, your development environment is ready for use.