**Chaves existentes no arquivo de configuração (config.json)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**server:**

Configurações gerais do servidor

**host**: endereço do servidor

**port**: porta do servidor

**backlog**: quantidade máxima para a fila de conexões pendentes no servidor (default 511)

**maxConnections**: quantidade máxima de conexões simultâneas no servidor

**timeout**: tempo esperado para aguardar uma resposta do cliente (em segundos, default 0 - infinito)

**keepAliveTimeout**: tempo esperado para identificar uma conexão ativa e reutilizável (em segundos, default 5s)

**maxHeadersCount**: quantidade máxima de headers de entrada (default 2000)

**headersTimeout**: tempo esperado para receber o conteúdo completo de um header (em segundos, default 60s)

**eventLoopMonitor**: ativa o monitoramento do loop de eventos do servidor, apenas para fins de análise de performance e testes

**clustered**: ativa servidor clusterizado. Se valor numérico, indica o número de processos executados no cluster (0 / true para todos possíveis)

**saveRouteLogs**: ativa o salvamento dos logs de acessos ao servidor em arquivos no disco

**routePrefix**: informa o prefixo de acesso as rotas (default / - equivale a nenhum prefixo)

\* Existindo, **routePrefix** deve ser anexado no início de **virtualPath** (em **pathVirtualStaticFiles**) e **path** (em **socketIo**)

\* para o último caso como garantia se a lib socket.io utilizar o protocolo pooling ao invés de websockets

**showFrontEndStackTraceErr**: ativa a exibição da pilha de erros no front-end (default falso)

**secure**: ativa o uso do protocolo **https** no servidor e especifica detalhes do certificado como nome e localização (formato PEM - X.509v3 base 64)

**pathVirtualStaticFiles**: array de objetos contendo a definição das pastas de conteúdo estático disponibilizadas pelo servidor

\* **maxAge** em minutos

**pathFavicon**: caminho para o ícone de favorito do servidor

**cors**: configurações de CORS do servidor (compartilhamento de recursos com origens diferentes)

**custom**: configurações para interfaceamento de APIs com serviços externos utilizados pelo servidor (pasta **custom**)

\* só faz sentido se alguma lib custom estiver sendo utilizada e necessitar de configurações globais

**fileUpload**: configurações para a lib de upload como caminho de salvamento e total de extensões permitidas (pasta **helpers**)

**session**: configurações para controle de sessão no servidor (**timeout** em minutos)

**customErrors**: define um ou mais erros personalizados no controller para o cliente. Não modifica a mensagem de erro original

\* formato array de objetos - ex: { **code**: 11000, **message**: 'duplicate key error collection: test.types index', **customMessage**: '' }

\* **code** deve ser exato, **message** pode ser parcial, **customMessage** é a mensagem de erro personalizada

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**auth:**

Configurações gerais para as regras de autenticação do servidor (permissões de acesso)

**sessWrapper**: informa o nome da variável que vai abrigar os dados da sessão no cookie

**authTipo**: define o tipo de autenticação. Se 2 => toda rota livre, exceto por **except** / Default: toda rota protegida, exceto por **except**

**except**: define as rotas de exceção. Aceita rota parcial ou exata; utilizar / para separadores de grupos no caminho ex: /PATH1/PATH2

Chave **path** de **socketIo** (sem **routePrefix**) fica livre em **auth** para o caso da lib socket.io utilizar protocolo pooling ao invés de websockets

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**socketIo:**

Configurações gerais do servidor de socket.io (websockets / pooling) via lib socket-io (pasta server)

**serverHost**: endereço do servidor

**serverPort**: porta do servidor

**path**: rota base que define a conversação bidirecional socket.io – redirecionada via proxy interno do servidor web para o servidor de socket.io

**changeOrigin**: ativa a mudança da origem da url no proxy para o servidor de socket.io

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**crypto:**

Configurações gerais de criptografia via lib crypto-hash (pasta helpers)

**hashAlgorithm**: algoritmo utilizado para a geração do hash

**hashDigestEncoding**: codificação resultante do hash gerado

**encryptInputEncoding**: especificação da codificação de entrada do arquivo a ser cifrado/decifrado

**encryptOutputEncoding**: especificação da codificação de saída do arquivo a ser cifrado/decifrado

**encryptAlgorithm**: algoritmo utilizado para cifragem/decifragem

**encryptKeyLen**: tamanho da chave do algoritmo utilizado para cifragem/decifragem (algo 192: 24 bytes; algo 256: 32 bytes)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**db:**

Configurações gerais de acesso ao banco de dados (MSSQL ou MongoDB)

**msSqlServer**: configurações para acesso ao MS SQL Server via lib db (pasta helpers), utilizando tedious/node-mssql

**connectionType**: define a forma de conexão com o DB. Se 2 => sem pool / Default: com pool

**configDb**: configurações globais de conexão, acesso e utilização. Ver mais detalhes em <https://tediousjs.github.io/node-mssql/>

**mongoose**: configurações para acesso ao MongoDB via lib db (pasta helpers), utilizando mongoose

**connectionString**: string completa de conexão e acesso

**configDb**: configurações globais de utilização do MongoDB, via mongoose

\* **authSource** é opcional

**configSchema**: configurações globais anexadas a geração dos esquemas, via mongoose

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**email:**

Configurações gerais de disparo de e-mails, incluindo queue

**queue**: configurações para o uso da fila de e-mails (queue) no servidor

**on**: ativa o uso da fila de e-mails (queue)

**saveFullLogs**: ativa o salvamento dos logs **tipo** **INFO** em arquivos no disco

**path**: caminho onde os arquivos gerados ficam armazenados, aguardando o disparo (momento do envio)

**fileExtension**": extensão dos arquivos a serem gerados

**limitPerRound**: quantidade máxima de envios por rodada

**timeCheck**: : intervalo entre as rodadas

**timeFirstCheck**: primeira checagem antes de iniciar o loop de rodadas

\* evita que clusters (quando **clustered** ativo) realizem checagens simultâneas na fila, a cada **timeCheck**

**transporter**: configurações para envio de e-mails pelo servidor via lib email (pasta helpers) ou queue (pasta server), utilizando NODEMAILER