

<<interface>>
NetworkInterface

setToken(string)
shutdown(): Observable<void>
getQuotaProgress(): Observable<QuotaProgressResponse>
updateMe(UpdateAccountRequest): Observable<AccountResponse>
getMe(): Observable<AccountResponse>
setServerConfig(ServerConfigDto): Observable<AccountResponse>
getServerConfig(): Observable<ServerConfigDto>
updateReport(string, ReportState): Observable<ReportResponse>
createReport(CreateReportRequest, string): Observable<ReportResponse>
getReports(ReportSortParameter, SortDirectionParameter, string, string): Observable<Array<ReportResponse>>
createReplic(CreateReplicRequest): Observable<ReplicResponse>
updateReplicState(string, ReplicState): Observable<Replic>
getReplics(ReplicSortParameter, SortDirectionParameter, string, string, Set<ReplicState>, string): Observable<Array<ReplicResponse>>
getReplicContent(string, string): Observable<string>
requestEmailVerification(boolean): Observable<void>
login(CredentialsReuquest): Observable<AccountWithTokensResponse>
refresh(RefreshRequest): Observable<AccountWithTokensResponse>
signup(CreateAccountRequest, boolean): Observable<AccountWithTokensResponse>
submitEmailVerification(SubmitEmailVerificationRequest): Observable<void>
restart(): Observable<void>
getAccountsFull(AccountSortParameter, SortDirectionParameter, string, Set<AccountState>, string): Observable<Array<PartialAccountResponse>>
getAccountsPartial(AccountSortParameter, SortDirectionParameter, string, string): Observable<Array<PartialAccountResponse>>
createAccount(CreateAccountRequest), boolean, boolean): Observable<AccountResponse>

AngularHttpNetworkInterface

http: HttpClient