

<<interface>> AccountService
getAccounts(Comparator<Account>, UUID, Set<AccountState>, String): List<Account>

<<interface>> AuthenticationService
authenticateWithCredentials(String, String, String): Optional<Account> authenticateWithJwt(String): Optional<Account> authenticateWithRefreshToken(UUID): Optional<Account> createAccessToken(UUID): String throws NotFoundException createRefreshToken(UUID): UUID throws NotFoundException createAccount(String, String, String, int, boolean, boolean, boolean): Account throws NotUniqueException, OperationDisabledException requestEmailVerification(UUID) throws NotFoundException validateEmail(UUID): Account throws InvalidTokenException

<<interface>> ReplicService
createReplic(URL, MediaMode, String, Instant, String, Account, FileWriterCallback): Replic throws ReplicQuotaMetException, ReplicContentWriteException, InvalidExpirationException getReplics(Comparator<Replic>, UUID, UUID, Set<ReplicState>, String): List<Replic> setReplicState(UUID, ReplicState): Replic throws NotFoundException visitReplic(UUID, UUID): ReplicAccess throws NotFoundException receiveContent(UUID, String): InputStream throws NotFoundException, InvalidPasswordException

<<interface>> ReportService
getReports(Comparator<Report>, String): List<Report> reportReplic(UUID, UUID, String): Report throws NotFoundException, OperationDisabledException

<<interface>> ServerConfigService
get(): ServerConfig save(ServerConfig)