Dokumentasi Protokol

Nama Aplikasi : Kotak

Dibuat Oleh :

Rezan Achmad (13508104)

Akbar Gumbira (13508106)

Dimas Tri Ciputra (13509602)

# Deskripsi Format Pesan/Protokol

Format pesan antara client dan server yaitu :

## Check

Check merupakan format pesan untuk client melakukan pengecekan apakah struktur di server terdapat perubahan. Struktur yang dimaksud adalah struktur keseluruhan satu working folder untuk tiap user.

Format request dari client yaitu :

1. *check [email] [pass] [client\_revision]*

Parameter email dan pass merupakan email user saat register dan pass merupakan password dari akun yang dimiliki. Parameter client\_revision merupakan parameter revision yang ingin dicek oleh client.

Kemudian, server akan memberikan respon berupa pesan, yaitu :

1. *failed email\_or\_pass\_is\_wrong*

Response ini akan dikirimkan oleh server apabila email atau pass yang dikirim dari client tidak sesuai dengan database yang ada.

1. *success nochange*

Response ini akan dikirimkan oleh server apabila revisi yang dimiliki oleh client sudah merupakan last revisi yang ada di server.

*3. success structure [server\_last\_revision] [structure]*

Response ini akan dikirimkan oleh server apabila terdapat struktur baru di server. Server akan mengembalikan nilai last revision di server serta struktur last revision tersebut.

## GetFile

GetFile merupakan format pesan untuk client mendapatkan suatu file dari server.

Format request dari client yaitu :

1. *getfile [email] [pass] [path] [path\_revision]*

Parameter email dan pass merupakan email user saat register dan pass merupakan password dari akun yang dimiliki. Parameter path merupakan path yang secara logical merupakan path di client yang diminta. Parameter path\_revision merupakan nomor revisi yang diminta untuk path tersebut.

Kemudian, server akan memberikan respon berupa pesan, yaitu :

1. *failed email\_or\_pass\_is\_wrong*

Response ini akan dikirimkan oleh server apabila email atau pass yang dikirim dari client tidak sesuai dengan database yang ada.

1. *failed filecontent\_is\_null*

Response ini akan dikirimkan oleh server apabila filecontent yang diminta oleh client null.

1. *failed no\_such\_requested\_file*

*Response ini akan dikirimkan oleh server apabila path yang diminta oleh client tidak terdapat pada server.*

1. *failed no\_such\_structure*

Response ini akan dikirimkan oleh server apabila struktur yang diminta oleh client dengan revisi yang merupakan parameter masukan path\_revision tidak terdapat pada server.

1. *success filecontent*

Response ini akan dikirimkan oleh server apabila email dan pass sudah sesuai dengan database serta path dan path\_revision cocok dengan database yang ada. Artinya terdapat file dengan pathnya = path serta path\_revision terdapat di field rev\_num pada database.

## AddFile

Addfile merupakan format pesan untuk client melakukan commit pada repository yang dimilikinya. Addfile ini sudah menangani add file (file baru ditambah) atau update file (memperbaharui suatu file).

Format request dari cient yaitu :

1. *addfile [email] [pass] [last\_revision] [path] [content]*

Parameter email dan pass merupakan email user saat register dan pass merupakan password dari akun yang dimiliki. Parameter last revision merupakan parameter revisi terakhir dari repository yang terdapat pada server. Parameter path merupakan path yang secara logical merupakan path di client yang akan ditambahkan. Parameter content byte file yang ditambahkan oleh client.

Kemudian, server akan melakukan respon berupa pesan, yaitu :

1. *failed email\_or\_pass\_is\_wrong*

Response ini akan dikirimkan oleh server apabila email atau pass yang dikirim dari client tidak sesuai dengan database yang ada.

1. *failed add\_failed*

Response ini akan dikirimkan oleh server apabila proses penambahan logically ke database gagal yang diakibatkan oleh gagalnya transaksi insert.

1. *failed revision\_not\_same*

Response ini akan dikirimkan oleh server apabila parameter last\_revision yang dikirimkan dari client tidak sama dengan last revision yang ada pada database.

1. *failed repository\_is\_locked*

Response ini akan dikirimkan oleh server apabila repository sedang dilock. Repository akan dilock apabila terdapat client lain yang melakukan suatu transaksi dengan server (transaksi ini berupa transaksi addfile, serta deletefile).

1. *success [last\_revision+1] date\_modified*

Response ini akan dikirimkan oleh server apabila server sukses menambhkan file baik secara logically pada database maupun physically pada repository. Selain itu server juga akan mengirimkan parameter last\_revision+1 (argar client mengetahui last revision yang ada pada server sekarang) serta data\_modified dari repository tersebut.

## DeleteFile

Delete merupakan format pesan untuk client melakukan deleting suatu file yang dimiliki di storage client. Secara physical pada repository, server sebenarnya tidak melakukan penghapusan file. Hal ini dikarenakan agar client dapat melakukan revert (mengambil kembali file-file sebelumnya yang ada). Server hanya melakukan penghapusan logically pada struktur di database.

Format request dari client yaitu :

1. *delete [email] [pass] [path] [client\_last\_revision]*

Parameter email dan pass merupakan email user saat register dan pass merupakan password dari akun yang dimiliki. Parameter client\_last\_revision merupakan parameter revisi terakhir dari repository yang terdapat pada server. Parameter path merupakan path yang secara logical merupakan path di client yang akan dihapus. Secara physical, file tidak akan dihapus pada repository agar client dapat melakukan revert ke revisi sebelum-sebelumnya.

Kemudian, server akan melakukan respon berupa pesan, yaitu :

1. *failed email\_or\_pass\_is\_wrong*

Response ini akan dikirimkan oleh server apabila email atau pass yang dikirim dari client tidak sesuai dengan database yang ada.

1. *. failed repository\_is\_locked*

Response ini akan dikirimkan oleh server apabila repository sedang dilock. Repository akan dilock apabila terdapat client lain yang melakukan suatu transaksi dengan server (transaksi ini berupa transaksi addfile serta deletefile).

1. *failed revision\_not\_same*

Response ini akan dikirimkan oleh server apabila parameter last\_revision yang dikirimkan dari client tidak sama dengan last revision yang ada pada database.

1. *failed file\_requested\_to\_delete\_not\_exist*

Response ini akan dikirimkan oleh server apabila path yang akan dihapus oleh client tidak terdapat pada server.

1. *failed add\_failed*

Response ini akan dikirimkan oleh server apabila proses penambahan logically ke database gagal yang diakibatkan oleh gagalnya transaksi insert.

1. *success [last\_revision+1] [date\_modified]*

Response ini akan dikirimkan oleh server apabila server sukses menghapus file secara logically pada database. Selain itu server juga akan mengirimkan parameter last\_revision+1 (argar client mengetahui last revision yang ada pada server sekarang) serta data\_modified dari repository tersebut.

# Deskripsi Prosedur / Interaksi (Diagram Alir)

Untuk melihat diagram alir dibawah ini, lakukan zooming pada diagram. Atau kami juga menyertakan diagram tersebut pada file UML dengan nama diagram\_alir\_KOTAK.uml



# Deskripsi Penangan Pesan/Protokol yang salah/tidak valid

Deskripsi penanganan pesan sudah dicantumkan pada bagian pertama (Bagian Deskripsi Format Pesan atau Protokol)