blacklist_ip

+ user_ip: String (unique index)

+ reason: String

blacklist_user_name

+ user_name: String (unique index)

+ reason: String

+ remaining_time: Number (in seconds)

users

+ _id: default primary object key

+ user_name: String (unique index, length > 3)

+ password: String (Hashed)

+ access_type: Boolean (0 = temporary user, 1 = regular user)

+ role: Boolean (0 = developer, 1 for client)

+ warning_counter: Number

+ creation_time: Number (unix time)

submitted_transactions

+ from_user: ObjID (reference 'users._id

+ to_user: ObjID (reference 'users._id')

+ amount: Number

+ time: Number (unix time)

+ verified: Boolean

temporary_user

+ user_id: ObjID (reference 'users._id')

+ deposit_amount: Number (not necessarily verified)

+ status : Number (rejected = 0, accepted = 1, pending: 2)

+ information : String (if pending : any note from super user, if rejected: reason for rejection)

client

+ user_id: ObjID (reference 'users._id')

+ amount_total: Number (amount of verified money in account of this user for use in this system)

+ penalty_amount: Number

+ interest: String (paragraph text)

+ business_credential: String (Description /+ links to business credential)

developer

+ user_id: ObjID (reference 'users._id')

+ amount_total: Number (amount of verified money in account of this user)

+ penalty_amount: Number

+ interest: String (paragraph text)

+ sample_work: String (text description /+ links to previous work)

For uploading .png/.jpeg/.jpg
profile image and .pdf resume
(required)

profile.files

+ user_id: ObjID (reference 'users._id')

+ all other default fields... (filename prefixed with username)

profile.chunks

+ all default fields...

---

c_system_demand

+ _id: default primary object key

+ c_user_id: ObjID (reference 'users._id')

+ creation_time: Number (unix time)

+ system_spec: String (paragraph)

+ timeline: Array (of object)

  {
      deadline: Number (unix time),
      description: String
  }

+ bidders: Array (of object)
    {
        d_user_name: ObjID (reference 'users._id'),
        timeline: Array (of object) { promised_deadline: Number (unix time) }
    }

+ chosen_bidder: { user_id: ObjID (reference 'users._id'), reason: String }

+ delivery_time: Number (unix time)

ratings

+ foreign_id: object key references c_system_demand

+ r_type: Boolean
            (0 = client to developer
             1 = developer to client)

+ to_user: ObjID (reference 'users._id')

+ from_user: ObjID (reference 'users._id')

+ rating: Number (1 to 5)

+ details: String

For uploading .zip or .tag.gz delivery
files (not required)

delivery.files

+ demand_id: Object ID references 'c_system_demand' _id

+ all other default fields... (filename prefixed with username)

delivery.chunks

+ all default fields...

---

warning_protest

+ user_name: ObjID (reference 'users._id')

+ message: String