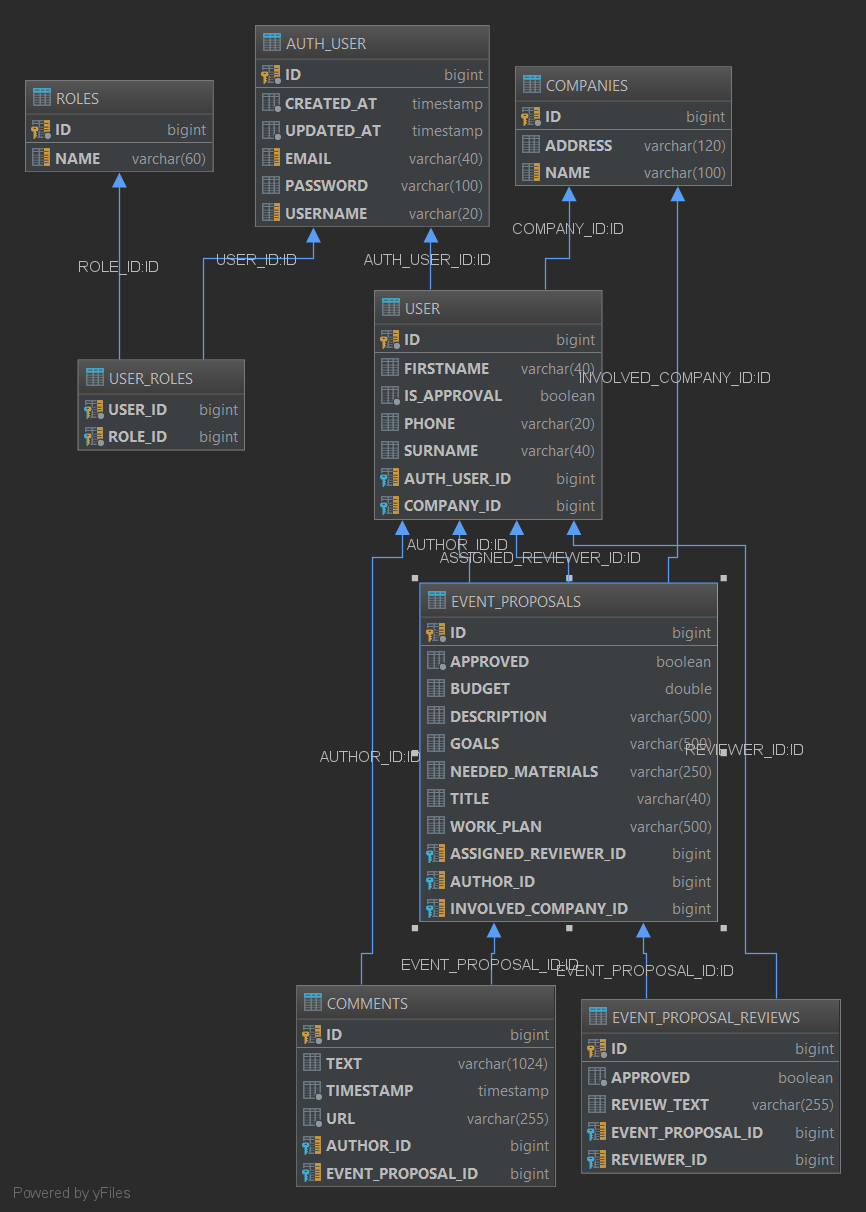
1. Data model



1. Security policies  
   Security is driven mostly by roles. Swagger interface endpoint /swagger-ui.html shows all (un)secured endpoints. If any requires additional authorization/role, it’s written under endpoint *notes* section.
2. Work contribution

* Hubert Olkiewicz  
  Project initialization, security configuration and implementation, swagger configuration (swagger description file).   
  Controllers: authentication, user, company, eventproposal.  
  Tests: two exemplary test of services, one written in Java using *Mockito*, second *Groovy* + *Spock*.
* Ondřej Zemánek:
* Data model
* repositories, controllers and services for:
  + Comment, Company, EventProposal, EventProposalReview
* email notifications
* Duarte Moreira: -

1. Swagger description file

swagger: '2.0'

info:

description: ECMA application

version: 0.1.0

title: ECMA

host: 'localhost:8080'

basePath: /

tags:

- name: authentication-controller

description: Authentication Controller

- name: comment-controller

description: Manages proposal comments

- name: company-controller

description: Manages companies

- name: event-proposal-controller

description: Manages event proposals

- name: event-proposal-review-controller

description: Manages event proposal reviews

- name: user-controller

description: Manages partner users

paths:

/api/auth/signin:

post:

tags:

- authentication-controller

summary: log in

operationId: authenticateUser

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: loginRequest

description: loginRequest

required: true

schema:

$ref: '#/definitions/LoginRequest'

responses:

'200':

description: OK

schema:

type: object

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

deprecated: false

/api/auth/signupPartnerUser:

post:

tags:

- authentication-controller

summary: register partner user

operationId: registerPartnerUser

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: request

description: request

required: true

schema:

$ref: '#/definitions/SignupRequest'

responses:

'200':

description: OK

schema:

type: object

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

deprecated: false

/api/auth/signupUser:

post:

tags:

- authentication-controller

summary: register ECMA user

operationId: registerUser

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: request

description: request

required: true

schema:

$ref: '#/definitions/SignupRequest'

responses:

'200':

description: OK

schema:

type: object

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

deprecated: false

/api/auth/signupUserAdmin:

post:

tags:

- authentication-controller

summary: register ECMA admin user

description: Only ADMIN

operationId: registerUserAdmin

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: request

description: request

required: true

schema:

$ref: '#/definitions/SignupRequest'

responses:

'200':

description: OK

schema:

type: object

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/auth/singupPartnerAdmin:

post:

tags:

- authentication-controller

summary: register partner admin

description: Only ADMIN

operationId: registerPartnerAdmin

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: request

description: request

required: true

schema:

$ref: '#/definitions/SignupRequest'

responses:

'200':

description: OK

schema:

type: object

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/comment/addNewCommentToEventProposal:

put:

tags:

- comment-controller

summary: Adds new comment to proposal

operationId: addNewCommentToEventProposal

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: addCommentRequest

description: addCommentRequest

required: true

schema:

$ref: '#/definitions/AddCommentRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/comment/deleteEventProposalCommentByAdmin:

delete:

tags:

- comment-controller

summary: Deletes proposal comment by admin

operationId: deleteEventProposalCommentByAdmin

produces:

- '\*/\*'

parameters:

- in: body

name: deleteCommentRequest

description: deleteCommentRequest

required: true

schema:

$ref: '#/definitions/DeleteCommentRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'204':

description: No Content

'401':

description: Unauthorized

'403':

description: Forbidden

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/comment/getAllEventProposalComments:

get:

tags:

- comment-controller

summary: Returns all proposal comments

operationId: getAllEventProposalComments

produces:

- '\*/\*'

parameters:

- name: ProposalId

in: query

description: ProposalId

required: true

type: integer

format: int64

responses:

'200':

description: OK

schema:

type: array

items:

$ref: '#/definitions/CommentDTO'

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/company/addNewCompany:

post:

tags:

- company-controller

summary: Adds new company

description: Only ADMIN

operationId: addNewCompany

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: addNewCompanyRequest

description: addNewCompanyRequest

required: true

schema:

$ref: '#/definitions/AddNewCompanyRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/company/deleteCompany:

delete:

tags:

- company-controller

summary: delete company by companyId

description: Only ADMIN

operationId: deleteCompany

produces:

- '\*/\*'

parameters:

- name: companyId

in: query

description: companyId

required: false

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'204':

description: No Content

'401':

description: Unauthorized

'403':

description: Forbidden

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/company/getAllCompanies:

get:

tags:

- company-controller

summary: Get all company

operationId: getAllCompanies

produces:

- '\*/\*'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/company/getCompanyInfo:

get:

tags:

- company-controller

summary: Get company info

description: USER OR PARTNER\_ADMIN

operationId: getCompanyInfo

produces:

- '\*/\*'

parameters:

- name: companyId

in: query

description: companyId

required: false

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/company/manageCompany:

patch:

tags:

- company-controller

summary: manage company

description: Only ADMIN or PARTNER\_ADMIN

operationId: manageCompany

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: manageCompanyRequest

description: manageCompanyRequest

required: true

schema:

$ref: '#/definitions/ManageCompanyRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'204':

description: No Content

'401':

description: Unauthorized

'403':

description: Forbidden

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposal/addEventProposal:

put:

tags:

- event-proposal-controller

summary: Adds new event proposal

operationId: addEventProposal

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: addEventProposalRequest

description: addEventProposalRequest

required: true

schema:

$ref: '#/definitions/AddEventProposalRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposal/approveProposal:

patch:

tags:

- event-proposal-controller

summary: approve proposal

operationId: approveProposal

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- name: proposalId

in: query

description: proposalId

required: true

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'204':

description: No Content

'401':

description: Unauthorized

'403':

description: Forbidden

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposal/deleteEventProposal:

delete:

tags:

- event-proposal-controller

summary: delete eventProposal by eventProposalId

description: Only ADMIN

operationId: deleteEventProposal

produces:

- '\*/\*'

parameters:

- name: eventProposalId

in: query

description: eventProposalId

required: false

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'204':

description: No Content

'401':

description: Unauthorized

'403':

description: Forbidden

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposal/getAllProposals:

get:

tags:

- event-proposal-controller

summary: get all proposals

operationId: getAllProposals

produces:

- '\*/\*'

responses:

'200':

description: OK

schema:

type: array

items:

$ref: '#/definitions/EventProposalDTO'

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposalReview/addEventProposalReview:

put:

tags:

- event-proposal-review-controller

summary: Adds new event proposal review

operationId: addEventProposalReview

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: addEventProposalReviewRequest

description: addEventProposalReviewRequest

required: true

schema:

$ref: '#/definitions/AddEventProposalReviewRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposalReview/assignNewProposalReviewer:

post:

tags:

- event-proposal-review-controller

summary: assign reviewer to proposal

operationId: assignNewProposalReviewer

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- name: proposalId

in: query

description: proposalId

required: true

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposalReview/assignNewProposalReviewerByAdmin:

post:

tags:

- event-proposal-review-controller

summary: assign reviewer to proposal by admin

operationId: getProposalReview

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- name: proposalId

in: query

description: proposalId

required: true

type: integer

format: int64

- name: userId

in: query

description: userId

required: true

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/eventProposalReview/getProposalReview:

get:

tags:

- event-proposal-review-controller

summary: get proposal review

operationId: getProposalReview\_1

produces:

- '\*/\*'

parameters:

- name: proposalId

in: query

description: proposalId

required: true

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/Optional«EventProposalReviewDTO»'

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/partner/deleteUser:

delete:

tags:

- user-controller

summary: delete user by userId

operationId: deleteUser

produces:

- '\*/\*'

parameters:

- name: userId

in: query

description: userId

required: false

type: integer

format: int64

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'204':

description: No Content

'401':

description: Unauthorized

'403':

description: Forbidden

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/partner/getAllCompanyUsers:

get:

tags:

- user-controller

summary: get all company users

description: Only ADMIN or PARTNER\_ADMIN

operationId: getAllCompanyUsers

produces:

- '\*/\*'

parameters:

- name: companyId

in: query

description: companyId

required: false

type: integer

format: int64

responses:

'200':

description: OK

schema:

type: array

items:

$ref: '#/definitions/UserDTO'

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/partner/manageUserAccount:

post:

tags:

- user-controller

summary: Manages partner user account

operationId: getAuthUser

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: manageUserAccountRequest

description: manageUserAccountRequest

required: true

schema:

$ref: '#/definitions/ManageUserAccountRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

/api/partner/manageUserAccountAdmin:

post:

tags:

- user-controller

summary: Manages partner user account

description: Only ADMIN or PARTNER\_ADMIN

operationId: getAuthUser\_1

consumes:

- application/json

produces:

- '\*/\*'

parameters:

- in: body

name: managePartnerUserRequest

description: managePartnerUserRequest

required: true

schema:

$ref: '#/definitions/ManagePartnerUserRequest'

responses:

'200':

description: OK

schema:

$ref: '#/definitions/ResponseEntity'

'201':

description: Created

'401':

description: Unauthorized

'403':

description: Forbidden

'404':

description: Not Found

security:

- Token /auth/signin. Add Bearer before TOKEN:

- global

deprecated: false

securityDefinitions:

Token /auth/signin. Add Bearer before TOKEN:

type: apiKey

name: Authorization

in: header

definitions:

AddCommentRequest:

type: object

properties:

eventProposalId:

type: integer

format: int64

description: event proposal id

text:

type: string

description: comment text

url:

description: comment url

$ref: '#/definitions/URL'

title: AddCommentRequest

description: Adds comment to proposal

AddEventProposalRequest:

type: object

properties:

budget:

type: number

format: double

description: event budget

description:

type: string

description: event description

goals:

type: string

description: event goals

neededMaterials:

type: string

description: event materials

title:

type: string

description: event title

workPlan:

type: string

description: event working plan

title: AddEventProposalRequest

description: Adds event

AddEventProposalReviewRequest:

type: object

properties:

approved:

type: boolean

description: event review approved

eventProposalId:

type: integer

format: int64

description: event review proposalId

reviewText:

type: string

description: event review text

title: AddEventProposalReviewRequest

description: Adds event review

AddNewCompanyRequest:

type: object

required:

- address

- name

properties:

address:

type: string

description: address

companyEmployees:

type: array

description: company employees

items:

type: integer

format: int64

name:

type: string

description: name of company

title: AddNewCompanyRequest

description: add new company

Comment:

type: object

properties:

id:

type: integer

format: int64

text:

type: string

timestamp:

type: string

format: date-time

url:

$ref: '#/definitions/URL'

title: Comment

CommentDTO:

type: object

properties:

author:

$ref: '#/definitions/User'

text:

type: string

title: CommentDTO

Company:

type: object

properties:

address:

type: string

employees:

type: array

items:

$ref: '#/definitions/User'

eventProposals:

type: array

items:

$ref: '#/definitions/EventProposal'

id:

type: integer

format: int64

name:

type: string

title: Company

DeleteCommentRequest:

type: object

properties:

commentId:

type: integer

format: int64

description: comment id

eventProposalId:

type: integer

format: int64

description: event proposal id

title: DeleteCommentRequest

description: Deletes proposal comment

EventProposal:

type: object

properties:

approved:

type: boolean

assignedReviewer:

$ref: '#/definitions/User'

author:

$ref: '#/definitions/User'

budget:

type: number

format: double

comments:

type: array

items:

$ref: '#/definitions/Comment'

description:

type: string

goals:

type: string

id:

type: integer

format: int64

involvedCompany:

$ref: '#/definitions/Company'

neededMaterials:

type: string

review:

$ref: '#/definitions/EventProposalReview'

title:

type: string

workPlan:

type: string

title: EventProposal

EventProposalDTO:

type: object

properties:

budget:

type: number

format: double

description:

type: string

goals:

type: string

id:

type: integer

format: int64

neededMaterials:

type: string

title:

type: string

workPlan:

type: string

title: EventProposalDTO

EventProposalReview:

type: object

properties:

approved:

type: boolean

id:

type: integer

format: int64

reviewText:

type: string

reviewer:

$ref: '#/definitions/User'

title: EventProposalReview

EventProposalReviewDTO:

type: object

properties:

empty:

type: boolean

present:

type: boolean

title: EventProposalReviewDTO

LoginRequest:

type: object

properties:

password:

type: string

usernameOrEmail:

type: string

title: LoginRequest

ManageCompanyRequest:

type: object

required:

- name

properties:

companyEmployeesToAdd:

type: array

description: new company employees

items:

type: integer

format: int64

companyEmployeesToRemove:

type: array

description: company empolyees to remove

items:

type: integer

format: int64

name:

type: string

description: name of company

newAddress:

type: string

description: new address of company

newName:

type: string

description: new name of company

title: ManageCompanyRequest

description: manage existing company

ManagePartnerUserRequest:

type: object

required:

- username

properties:

email:

type: string

description: email

firstname:

type: string

description: new firstname

isApproval:

type: boolean

description: new firstname

newCompany:

type: string

description: Company where user should be assigned

newPassword:

type: string

description: new password

newUsername:

type: string

description: new username

phoneNumber:

type: string

description: phone number

surname:

type: string

description: surname

userCompanyName:

type: string

description: name of user company

username:

type: string

description: username

title: ManagePartnerUserRequest

description: Manage user request

ManageUserAccountRequest:

type: object

properties:

email:

type: string

description: email

firstname:

type: string

description: new firstname

password:

type: string

description: new password

phoneNumber:

type: string

description: phone number

surname:

type: string

description: surname

title: ManageUserAccountRequest

description: Manages user account settings

ResponseEntity:

type: object

properties:

body:

type: object

statusCode:

type: string

enum:

- 100 CONTINUE

- 101 SWITCHING\_PROTOCOLS

- 102 PROCESSING

- 103 CHECKPOINT

- 200 OK

- 201 CREATED

- 202 ACCEPTED

- 203 NON\_AUTHORITATIVE\_INFORMATION

- 204 NO\_CONTENT

- 205 RESET\_CONTENT

- 206 PARTIAL\_CONTENT

- 207 MULTI\_STATUS

- 208 ALREADY\_REPORTED

- 226 IM\_USED

- 300 MULTIPLE\_CHOICES

- 301 MOVED\_PERMANENTLY

- 302 FOUND

- 302 MOVED\_TEMPORARILY

- 303 SEE\_OTHER

- 304 NOT\_MODIFIED

- 305 USE\_PROXY

- 307 TEMPORARY\_REDIRECT

- 308 PERMANENT\_REDIRECT

- 400 BAD\_REQUEST

- 401 UNAUTHORIZED

- 402 PAYMENT\_REQUIRED

- 403 FORBIDDEN

- 404 NOT\_FOUND

- 405 METHOD\_NOT\_ALLOWED

- 406 NOT\_ACCEPTABLE

- 407 PROXY\_AUTHENTICATION\_REQUIRED

- 408 REQUEST\_TIMEOUT

- 409 CONFLICT

- 410 GONE

- 411 LENGTH\_REQUIRED

- 412 PRECONDITION\_FAILED

- 413 PAYLOAD\_TOO\_LARGE

- 413 REQUEST\_ENTITY\_TOO\_LARGE

- 414 URI\_TOO\_LONG

- 414 REQUEST\_URI\_TOO\_LONG

- 415 UNSUPPORTED\_MEDIA\_TYPE

- 416 REQUESTED\_RANGE\_NOT\_SATISFIABLE

- 417 EXPECTATION\_FAILED

- 418 I\_AM\_A\_TEAPOT

- 419 INSUFFICIENT\_SPACE\_ON\_RESOURCE

- 420 METHOD\_FAILURE

- 421 DESTINATION\_LOCKED

- 422 UNPROCESSABLE\_ENTITY

- 423 LOCKED

- 424 FAILED\_DEPENDENCY

- 426 UPGRADE\_REQUIRED

- 428 PRECONDITION\_REQUIRED

- 429 TOO\_MANY\_REQUESTS

- 431 REQUEST\_HEADER\_FIELDS\_TOO\_LARGE

- 451 UNAVAILABLE\_FOR\_LEGAL\_REASONS

- 500 INTERNAL\_SERVER\_ERROR

- 501 NOT\_IMPLEMENTED

- 502 BAD\_GATEWAY

- 503 SERVICE\_UNAVAILABLE

- 504 GATEWAY\_TIMEOUT

- 505 HTTP\_VERSION\_NOT\_SUPPORTED

- 506 VARIANT\_ALSO\_NEGOTIATES

- 507 INSUFFICIENT\_STORAGE

- 508 LOOP\_DETECTED

- 509 BANDWIDTH\_LIMIT\_EXCEEDED

- 510 NOT\_EXTENDED

- 511 NETWORK\_AUTHENTICATION\_REQUIRED

statusCodeValue:

type: integer

format: int32

title: ResponseEntity

SignupRequest:

type: object

properties:

companyName:

type: string

email:

type: string

name:

type: string

password:

type: string

phoneNumber:

type: string

surname:

type: string

username:

type: string

title: SignupRequest

URL:

type: object

properties:

authority:

type: string

content:

type: object

defaultPort:

type: integer

format: int32

file:

type: string

host:

type: string

path:

type: string

port:

type: integer

format: int32

protocol:

type: string

query:

type: string

ref:

type: string

userInfo:

type: string

title: URL

URLStreamHandler:

type: object

title: URLStreamHandler

User:

type: object

properties:

firstname:

type: string

id:

type: integer

format: int64

isApproval:

type: boolean

phone:

type: string

surname:

type: string

title: User

UserDTO:

type: object

properties:

company:

$ref: '#/definitions/Company'

eventProposalReviewAssigments:

type: array

items:

$ref: '#/definitions/EventProposal'

eventProposals:

type: array

items:

$ref: '#/definitions/EventProposal'

firstname:

type: string

id:

type: integer

format: int64

isApproval:

type: boolean

phone:

type: string

reviews:

type: array

items:

$ref: '#/definitions/EventProposalReview'

surname:

type: string

title: UserDTO