ASDAN 商业竞赛项目 API 设计

Tomato 小组

2017 年 10 月 18 日

Contents

1. Client API Endpoints: 为特定用户提供相关信息。
2. Admin API Endpoints: 为管理员提供相关信息。
3. Administrative Endpoints: 登录和交易时使用。
4. Utility Endpoints: 询问系统相关信息。

1 Client API Endpoints

为特定用户提供相关信息。

1.1 Get Property

输入 ID，获得与这一 ID 相关的用户的全部信息。用户有默认头像。如果比赛还没有开始，则 rank 为 0。

TO-DO：用户自己设置头像。TO-DO：三个用户同时登录。

Request

```
GET /api/client/property/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 200 OK

{
   "name": "team1",
   "image": "/9j/4QAYRXhpZg ...."
   "members": ["member11", "member12"],
   "wealth": 100,
   "rank": 1
}
```

Error

```
HTTP 404 NOT FOUND
```

1.2 Get Information

输入 ID，获得与这一 ID 相关的用户的信息。包括机器的使用情况和材料的价格。

Request

```
GET /api/client/info/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 200 OK

{
    "name": "team1",
    "wealth": 100,
    "machine":
    [
        {
            "id": "machine1_id",
            "type": "type1",
            "left": "3"
        },
        {
            "id": "machine2_id",
            "type": "type2",
            "left": "2"
        }
    ],
    "material":
    [
        {
            "price": "10",
            "number": "20",
        },
        {
            "price": "20",
            "number": "0",
        },
        {
            "price": "80",
            "number": "150",
        }
    ]
}
```

Error

```
HTTP 404 NOT FOUND
```

1.3 Update Information

向服务器发送更改过的信息。

Request

```
PUT /api/client/info/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json

{
    "name": "team1",
    "wealth": 100,
    "machine":
    [
        {
            "id": "machine1_id",
            "left": "3"
        },
        {
            "id": "machine2_id",
            "left": "2"
        }
    ],
    "material":
    [
        {
            "price": "10",
            "number": "20",
        },
        {
            "price": "20",
            "number": "0",
        },
        {
            "price": "80",
            "number": "150",
        }
    ]
}
```

Returns

```
HTTP 200 OK

{
    "name": "team1",
    "wealth": 100,
    "machine": ["machine1_id", "machine2_id", "machine3_id"],
    "material": ["10", "20", "0"}
}
```

Error

```
HTTP 404 NOT FOUND
{
    "error": "Unable to update. User with id xxx not found."
}
```

1.4 Get History

获取与某一用户相关的历史信息。在发订单的时候客户端手动更新 History。

Request

```
GET /api/client/history/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 200 OK

[
    {
        "time": "hh:MM:ss",
        "target": "team1",
        "action": "sell",
        "content": {"material1": 1},
        "price": "10",
        "status": "success",
    },
    {
        "time": "hh:MM:ss",
        "target": "team1",
        "action": "buy",
        "content": {"material1": 1},
        "price": "10",
        "status": "fail",
    }
]
```

Error

```
HTTP 404 NOT FOUND
```

2 Admin API Endpoints

2.1 Get All Competitions

列出全部比赛。

Request

```
GET /api/admin/competition/getall

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 200 OK

[
   {
      "id": "competiton1_id",
      "name": "competition1",
      "status": "processing"
   },
    {
        "id": "competiton2_id",
        "name": "competition2",
        "status": "end"
    }
]
```

Error

```
HTTP 204 NO CONTENT
```

2.2 Create Competition

新建一场比赛。注意，底层也要生成机器的 id。

Request

```
POST /api/admin/competition/new

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json

{
    "name": "competition_name",
    "round": "2",
    "start_wealth": "1000",
    "round_parameter":
    [
        {
            "machine_start_price": ["300", "350", "400"],
            "material_produce_cost": ["10", "20", "30"],
            "time": "900",
        },
        {
            "machine_start_price": ["300", "350", "400"],
            "material_produce_cost": ["10", "20", "30"],
            "time": "900",
        }
    ]
}
```

Returns

```
HTTP 201 CREATED
{
   "id": "competition_id",
   "name": "competition_name",
   "status": "not started",
   "round": "2",
   "start_wealth": "1000",
   "round_parameter":
   [
      {
         "machine_start_price": ["300", "350", "400"],
         "material_produce_cost": ["10", "20", "30"],
         "time": "900",
      },
      {
         "machine_start_price": ["300", "350", "400"],
         "material_produce_cost": ["10", "20", "30"],
         "time": "900",
      }
   ]
}
```

Error

```
HTTP 404 NOT FOUND
{
   "error":"Unable to delete. Competition with id xxx not found."
}
```

2.3 Delete Competition By ID

通过 ID 删除比赛。

Request

```
DELETE /api/admin/competiton/id={competition_id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 204 NO CONTENT
```

Error

```
{
   "error":"Unable to delete. Competition with id xxx not found."
}
```

2.4 Update Competition Status

更新比赛状态。

Request

```
POST /api/admin/competition/status/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json

{
   "status": "processing"
}
```

Returns

```
HTTP 200 OK
```

Error

```
HTTP 404 NOT FOUND
{
   "error":"Unable to update. Competition with id xxx not found"
}
```

2.5 Generate Accounts

要求系统生成比赛账户和密码。

Request

```
POST /api/admin/competition/accounts/id={id}?number={number}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 201 CREATED
[
   {
      "client_id": "id1",
      "username": "username1",
      "password": "password1",
   },
   {
      "client_id": "id2",
      "username": "username2",
      "password": "password2",
   },
]
```

Error

```
HTTP 404 NOT FOUND
{
   "error":"Competition with id xxxx not found"
}
```

```
HTTP 409 CONFLICT
{
   "error":"Cannot generate accounts for competition with id xxxx"
}
```

2.6 Get Competition Property

从服务器按 id 获取某一比赛的各种属性。如果该比赛的属性尚未被设置，则该项为空。属性包括名称、比赛轮数（如果比赛已开始，则不能删除已开始或结束的轮）、比赛各项参数（不能修改已开始或结束的轮的参数）、机器的 id 等等。

Request

```
GET /api/admin/competition/property/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 200 OK
{
   "id": "competition_id",
   "name": "competition_name",
   "status": "not started",
   "round": "2",
   "start_wealth": "1000",
   "round_parameter":
   [
      {
         "machine_start_price": ["300", "350", "400"],
         "machine_id": ["id1", "id2", "id3"],
         "material_produce_cost": ["10", "20", "30"],
         "time": "900",
      },
      {
         "machine_start_price": ["300", "350", "400"],
         "machine_id": ["id1", "id2", "id3"],
         "material_produce_cost": ["10", "20", "30"],
         "time": "900",
      }
   ]
}
```

Error

```
HTTP 404 NOT FOUND
{
   "error": "Competition with id xxx not found."
}
```

2.7 Update Competition Property

更新比赛的各种属性。属性包括名称、比赛轮数（如果比赛已开始，则不能删除已开始或结束的轮）、比赛各项参数（不能修改已开始或结束的轮的参数）。

Request

```
PUT /api/admin/competition/property/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

```
{
    "name": "competition_name",
    "round": "2",
    "start_wealth": "1000",
    "round_parameter":
    [
        {
            "machine_start_price": ["300", "350", "400"],
            "material_produce_cost": ["10", "20", "30"],
            "time": "900",
        },
        {
            "machine_start_price": ["300", "350", "400"],
            "material_produce_cost": ["10", "20", "30"],
            "time": "900",
        }
    ]
}
```

Returns

```
HTTP 201 CREATED

{
    "id": "competition_id",
    "name": "competition_name",
    "round": "2",
    "start_wealth": "1000",
    "round_parameter":
    [
        {
            "machine_start_price": ["300", "350", "400"],
            "material_produce_cost": ["10", "20", "30"],
            "machine_id": ["id1", "id2", "id3"],
            "time": "900",
        },
        {
            "machine_start_price": ["300", "350", "400"],
            "material_produce_cost": ["10", "20", "30"],
            "machine_id": ["id1", "id2", "id3"],
            "time": "900",
        }
    ]
}
```

Error

```
HTTP 404 NOT FOUND
{
    "error": "Competition with id xxx not found."
}
```

```
HTTP 400 INVALID REQUEST
{
    "error": "Cannot update competition id xxx with given changes."
}
```

2.8 Get Competition Information

获取当前比赛信息，包括队伍的数量、资产、交易记录、机器的使用情况等。

Request

```
GET /api/admin/competition/info/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json
```

Returns

```
HTTP 200 OK
{
    "id": "competition_id",
    "name": "competition_name",
    "status": "not started",
    "round": "2",
    "present_round": "0",
    "team_info":
    [
        {
            "id": "id1",
            "wealth": "100",
            "material": ["30", "40", "50"],
            "machine":
            [
                {
                    "id": "machine1_id",
                    "type": "type1",
                    "left": "3"
                },
                {
                    "id": "machine2_id",
                    "type": "type2",
                    "left": "2"
                }
            ]
        },
        {
            "id": "id2",
            "wealth": "100",
            "material": ["30", "40", "50"],
            "machine":
            [
                {
                    "id": "machine1_id",
                    "type": "type1",
                    "left": "3"
                },
                {
                    "id": "machine2_id",
                    "type": "type2",
                    "left": "2"
                }
            ]
        }
    ],
    "trade_history":
    [
        {
            "time": "hh:MM:ss",
            "sell": "team_id1",
            "buy": "team_id2",
            "content": {"material1": 1},
            "price": "10"
        },
         {
            "time": "hh:MM:ss",
            "sell": "team_id1",
            "buy": "team_id2",
            "content": {"material2": 1},
            "price": "20"
        }
    ]
```

```
}
```

Error

```
HTTP 404 NOT FOUND
{
    "error": "Competition with id 1 not found."
}
```

2.9 Update Competition Information

向服务器发送对比赛的更新信息。增加机器、分配财产之类的。

Request

```
PUT /api/admin/competition/info/id={id}

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json

{
    "round": "2",
    "present_round": "0",
    "team_info":
    [
        {
            "id": "id1",
            "wealth": "100",
            "machine":
            [
                {
                    "id": "machine1_id",
                    "left": "3"
                },
                {
                    "id": "machine2_id",
                    "left": "2"
                }
            ]
        },
        {
            "id": "id2",
            "wealth": "100",
          "machine":
          [
              {
                 "id": "machine1_id",
                 "left": "3"
              },
              {
                 "id": "machine2_id",
                 "left": "2"
              }
          ]
       }
   ]
}
```

Returns

```
HTTP 200 OK
{
   "id": "competition_id",
   "name": "competition_name",
   "status": "not started",
   "round": "2",
   "present_round": "0",
   "team_info":
   [
       {
          "id": "id1",
          "wealth": "100",
          "material": ["30", "40", "50"],
          "machine":
          [
              {
                 "id": "machine1_id",
                 "type": "type1",
                 "left": "3"
              },
              {
                 "id": "machine2_id",
                 "type": "type2",
                 "left": "2"
              }
          ]
       },
       {
          "id": "id2",
          "wealth": "100",
          "material": ["30", "40", "50"],
          "machine":
          [
              {
                "id": "machine1_id",
                "type": "type1",
                "left": "3"
              },
              {
                "id": "machine2_id",
                "type": "type2",
                "left": "2"
              }
          ]
       }
   ],
   "trade_history":
   [
       {
          "time": "hh:MM:ss",
          "sell": "team_id1",
          "buy": "team_id2",
          "content": {"material1": 1},
          "price": "10"
       },
        {
          "time": "hh:MM:ss",
          "sell": "team_id1",
          "buy": "team_id2",
          "content": {"material2": 1},
          "price": "20"
       }
   ]
}
```

Error

```
HTTP 404 NOT FOUND
{
   "error": "Competition with id xxx not found."
}
```

```
HTTP 400 INVALID REQUEST
```

```
{
   "error": "Cannot update competition id xxx with given information."
}
```

3 Administrative Endpoints

登录和交易时使用。

3.1 Login Admin

管理员登录。

Request

```
POST /api/admin/login

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json

{
   "username": "admin",
   "password": "admin",
}
```

Returns

```
HTTP 200 OK
{
   "username": "admin",
   "token": "1283091828021803120",
}
```

Error

```
HTTP 401 NOTAUTHORIZED
{
   "error": "Admin with username admin doesn\'t exist or password is wrong."
}
```

3.2 Login Client

管理员登录。

Request

```
POST /api/client/login

Host: localhost:8080
Auth:
Content-type: application/json
Accept: application/json

{
   "username": "client",
   "password": "client",
}
```

Returns

```
HTTP 200 OK
{
   "username": "client",
   "token": "1283091828021803120",
}
```

Error

```
HTTP 401 NOTAUTHORIZED
{
   "error": "Client with username admin doesn\'t exist or password is wrong."
}
```

4 Utility Endpoints

询问系统相关信息。