

Hướng dẫn sử dụng Notification API Service

1. Các chức năng đăng nhập/đăng xuất

1.1. Chức năng đăng nhập

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

```
POST http://localhost:53771/api/login
```

```
{
  "UserName": "TNY",
  "Password": "12345"
}
```

- Server sẽ trả về thông báo OK cùng một tham số SessionGuid
- Tham số này được dùng để giữ liên kết với user vừa đăng nhập

1.2. Chức năng đăng xuất

- Sử dụng URI bên dưới

```
POST http://localhost:53771/api/logout
```

- Thêm tham số SessionGuid được trả về khi đăng nhập vào header của API

2. Các chức năng gửi thông báo tới web app

2.1. Chức năng gửi thông báo ngay hiện tại

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

```
POST http://localhost:53771/api/Notification/SendToList
```

```
{
  "Content": "My notification",
  "AppIDs": ["5f82e7ebe280cc251f9fadd5"],
  "RecipientIDs": [
    "cat", "dog"
  ]
}
```

- Thêm cả tham số SessionGuid được trả về khi đăng nhập vào header của API
- Server sẽ thông báo OK và gửi trả về Id của thông báo vừa gửi

2.2. Chức năng gửi thông báo hẹn giờ

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/Notification/SendToList_Schedule

```
{
  "Content": "My schedule notification",
  "AppIDs": ["5f82e7ebe280cc251f9fadd5"],
  "RecipientIDs": [
    "cat", "dog"
  ],
  "ScheduleTime": "2020-12-03 15:35:00"
}
```

- Thêm cả tham số SessionGuid được trả về khi đăng nhập vào header của API
- Server sẽ thông báo OK và gửi trả về Id của thông báo vừa gửi

2.3. Chức năng huỷ hẹn giờ

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/Notification/CancelScheduledNotif

```
{
  "Id": "5f82e7ebe280cc251f9fadd7"
}
```

- Tham số Id chính là Id của thông báo được server trả về

2.4. Chức năng gửi thông báo theo lịch trình

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/Notification/SendToList_Routine

```
{
  "Content": "My routine notification",
  "AppIDs": ["5f82e7ebe280cc251f9fadd5"],
  "RecipientIDs": [
    "cat", "dog"
  ],
  Routine: { "Type": "daily", "DayOfWeek": 1, "Time": "2021-01-01 21:00" }
}
```

- Thêm cả tham số SessionGuid được trả về khi đăng nhập vào header của API
- Server sẽ thông báo OK và gửi trả về Id của thông báo vừa gửi

3. Chức năng liên quan tới bộ gửi thông báo hẹn giờ

3.1. Chức năng bật bộ gửi thông báo hẹn giờ

- Sử dụng URI bên dưới, API này không có tham số thêm vào

```
POST http://localhost:53771/api/ScheduledNotifSender/Start
```

3.2. Chức năng tắt bộ gửi thông báo hẹn giờ

**Chức năng đang phát triển*

4. Các chức năng quản lý thông báo

4.1. Chức năng xem thông báo hẹn giờ chưa gửi

- Sử dụng URI bên dưới, API này không có tham số thêm vào

```
GET http://localhost:53771/api/Notification/GetAllScheduledNotif
```

4.2. Chức năng tra cứu thông báo theo người gửi

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

```
GET http://localhost:53771/api/Notification/GetNotif_ByUserID
```

```
{
  "UserID": "5f82e7ebe280cc251f9fadd7"
}
```

4.3. Chức năng tra cứu thông báo theo thời gian gửi

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

```
GET http://localhost:53771/api/Notification/GetNotif_ByTime
```

```
{
  "FromTime": "2020-12-01",
  "ToTime": "2020-12-03"
}
```

4.4. Chức năng tra cứu thông báo theo Id

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

```
GET http://localhost:53771/api/Notification/GetNotif_ById
```

```
{
  "Id": "5fc5fb26684f47ef5c3f7939"
}
```

- Tham số Id chính là Id của thông báo được server trả về

5. Các chức năng quản lý User

5.1. Chức năng xem tất cả User

- Sử dụng URI bên dưới, API này không có tham số thêm vào

```
GET http://localhost:53771/api/User/GetAll
```

5.2. Chức năng thêm User

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

```
POST http://localhost:53771/api/User/Add
```

```
{
  "UserName": "TNY",
}
```

```
"DisplayName": "Tran Tuan Khoa",
"Password": "12345"
}
```

- Server sẽ trả về Id của User mới tạo

5.3. Chức năng cập nhật User

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/User/Update

```
{
  "Id": "xxxxxxx",
  "UserName": "TNY",
  "DisplayName": "Tran Tuan Khoa",
  "Password": "12345"
}
```

5.4. Chức năng xoá User

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/User/Remove

```
{
  "Id": "5fc5fb26684f47ef5c3f7939"
}
```

6. Các chức năng quản lý App thông báo trong hệ thống

6.1. Chức năng thêm App

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/App/Add

```
{
  "Name": "TNY",
}
```

```
"Type": "WebApp",
}
```

- Server sẽ trả về Id của App mới tạo

6.2. Chức năng cập nhật App

POST http://localhost:53771/api/App/Update

```
{
  "Id": "5fc5fb26684f47ef5c3f7939"
  "Name": "TNY",
  "Type": "WebApp",
}
```

6.2. Chức năng xoá App

POST http://localhost:53771/api/App/Remove

```
{
  "Id": "5fc5fb26684f47ef5c3f7939"
}
```

6.3. Chức xem tất cả App

- Sử dụng URI bên dưới, API này không có tham số thêm vào

GET http://localhost:53771/api/App/GetAll

6. Các chức năng quản lý Template thông báo trong hệ thống

6.1. Chức năng thêm Template

- Sử dụng URI bên dưới và thêm vào tham số vào body của API

POST http://localhost:53771/api/Templates/Add

```
{
  "Name": "TNY",
  "Type": "Thong bao",
  "Content": "xxxxx"
}
```

- Server sẽ trả về Id của Template mới tạo

6.2. Chức năng cập nhật Template

POST <http://localhost:53771/api/Templates/Update>

```
{
  "Id": "5fc5fb26684f47ef5c3f7939"
  "Name": "Ten template",
  "Type": "Thong bao",
  "Content": "xxxxx"
}
```

6.2. Chức năng xoá Template

POST <http://localhost:53771/api/Templates/Remove>

```
{
  "Id": "5fc5fb26684f47ef5c3f7939"
}
```

6.3. Chức xem tất cả Template

- Sử dụng URI bên dưới, API này không có tham số thêm vào

GET <http://localhost:53771/api/Templates/GetAll>