Use Case Diagram

Admin
- Backup and recovery of monitoring system data
- Monitoring system software update
- Creating and configuring monitoring scripts
- Managing users and their access rights

User
- Taking action to correct faults/anomalies
- Receive notifications about system anomalies
- View reports system
- Viewing the current system status
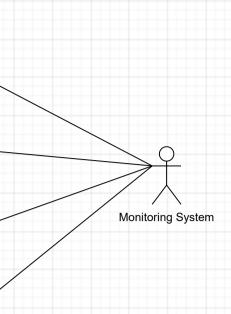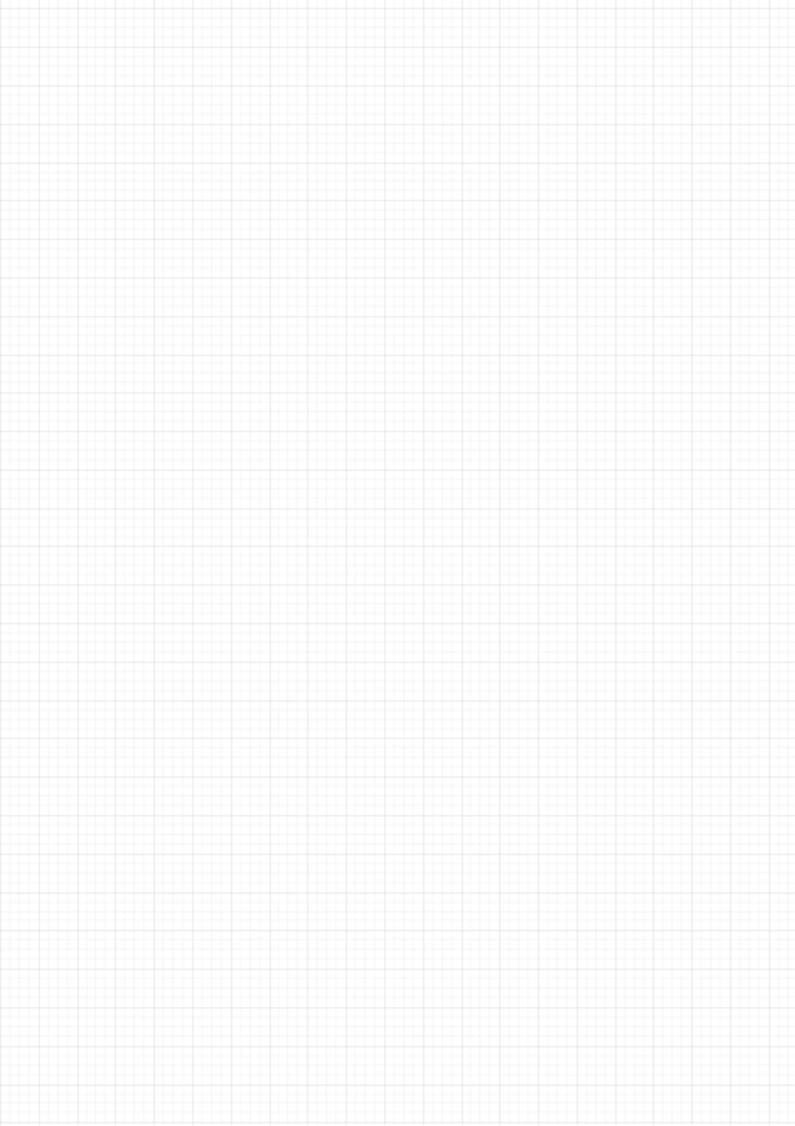- Login

System use cases:
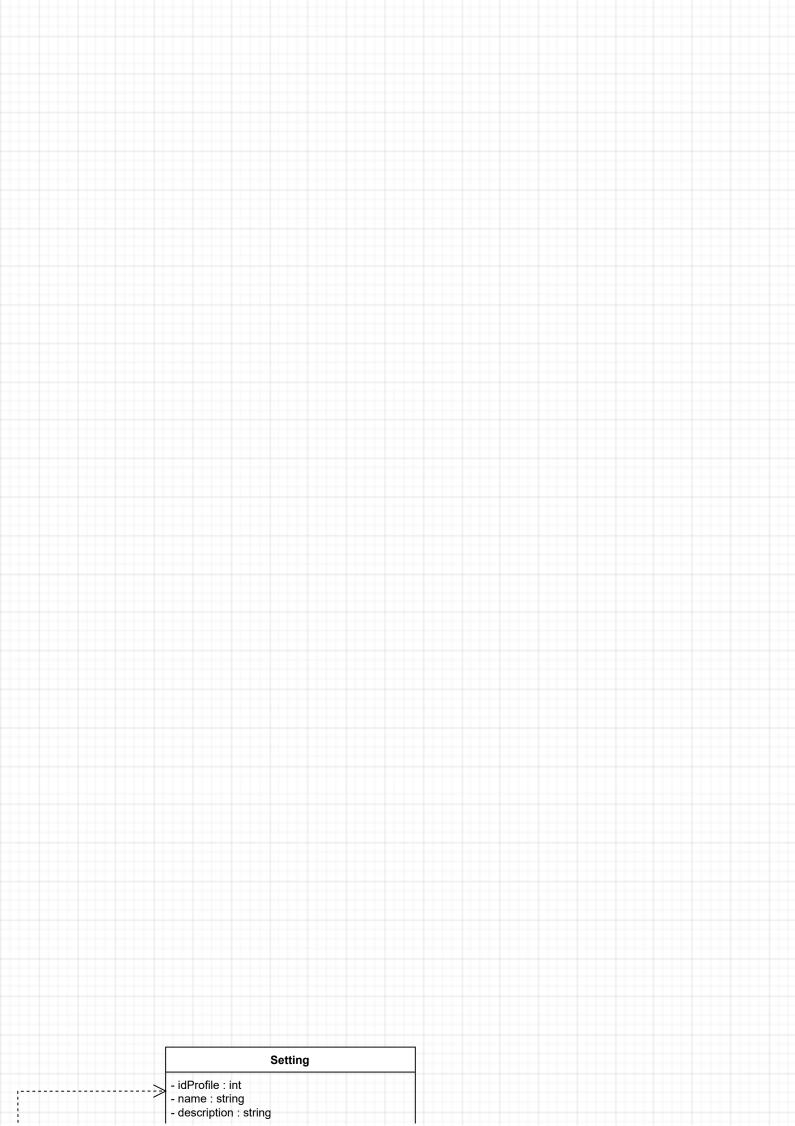- Collect data and generate reports on network health and performance
- Generating and sending notifications and alarms
- Response to anomalies
- Data Analysis and Anomaly Detection

Monitoring System

Setting
- idProfile : int
- name : string
- description : string

User

- id : Int
- name : string
- password : string
- accessRights : string
- dateOfRegistration : date

+ Login()
+ ViewTheSystemStatus()
+ ViewingSystemReports()
+ takeActionOnAnomaly()

IAdmin

- accessLocations : string
- profiles : <container> settings
- users : <container> users

+ StartMonitoringSystem()
+ OffMonitoringSystem()
+ ChangeUserInfo()
+ ConfigureSystemSettings()
+ GenerateReports()

Event

- id : int
- date : date
- description : string
- dangerLevel : int
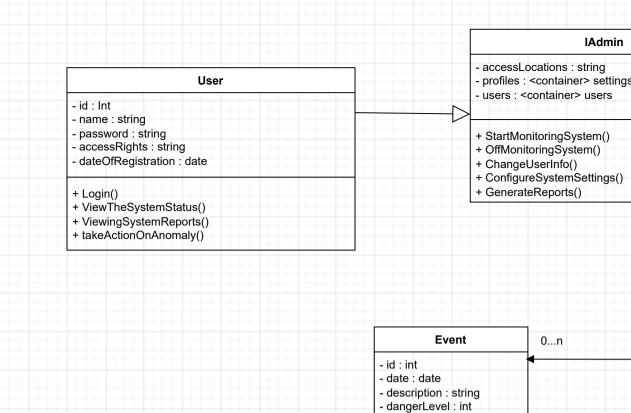
+ GetIdEvent() : int
+ GetDateEvent() : date
+ GetDescription() : string
+ GetDangerLevel() : int

0...n

```
                                        + addProfile()
                                        + ManageUserPermissions(user, permissions)
                                        + createMonitoringScenarios()
                                        + backupData()
```

NetworkDevice

- id : int
- name : string
- ip : string
- location : string
- additionalInfo : string

+ turn on()
+ switch off()
+ changeSelfInfo()

1...n

1...n

0... n

0...n

MonitoringSystem

- reports : type
- configureSystem : type
- networkParameters : type
- NetworkDevice<container> devices

+ collectNetworkData(devices)
+ analyzeNetworkData()
+ generateNotifications() : events
+ addDeviceInSystem()
+ deleteDeviceInSystem(0
+ createPerformanceReport() : type
+ reactToAnomalies()
+ ShutdownSystem()

0... n

Administrator | MonitoringSystem

- (start) ● → **Login**
- **Login** → **collectNetworkData**
- **collectNetworkData** → **AnalyzeData**
- **AnalyzeData** → **IsAnomalys?**
- **IsAnomalys?** — yes → **Generate Notifications**
- **IsAnomalys?** — no → ◉ (end)
- **Generate Notifications** → (fork bar)
- (fork bar) → **Take action on anomaly**
- (fork bar) → **React to anomalies**
- **React to anomalies** → (join bar)
- **Take action on anomaly** → (join bar)
- (join bar) → **collectNetworkData**

<<Component>
«Event»

<<Component>
«Setting»

<<Component>>
«MonitoringSystem»

<<Component>>
«NetworkDevice»

<<Component>
«Admin»

<<Component>
«User»

Monitoring System

Admin

Monitoring System

Admin