Architecture

Downtime Alerter service is a monolith ASP.Net Core web application. All of UI, data access, background services and task queues packaged in a single application.

Software Stack

Backend

- ASP.Net Core 5.0
- ASP.Net Core Identity and UI for authentication and authorization
- Entity Framework Core 5.0
- Serilog for logging
- SendGrid for e-mail notification
- MediatR for mapping entities to models, vice versa

Frontend

- ASP.Net Core Identity UI
- Bootstrap 4
- jQuery
- sweetalert2
- axios for AJAX requests

Components

The essential components of the architecture of the system are:

- Database: `Sqlite`
- Background Service: `DowntimeAlertHostedService`
  - Backlog: `BacklogProcessingService`
- Dispatcher: `MonitorLoop`
- Task Queue: `BackgroundTaskQueue`
- Notification Service: `Mediator`
- Worker: `QueuedHostedService`

Features

- Authentication and Authorization
  - Login
  - Forgot password
  - Role based access
  - Two factor auth

Missing Features

- Handling network, transient error to avoid false positives

- Setting timeout for HTTP requests to preserve unmanaged IO resources (e.g., networks sockets)
- Validation for interval values is not sufficient

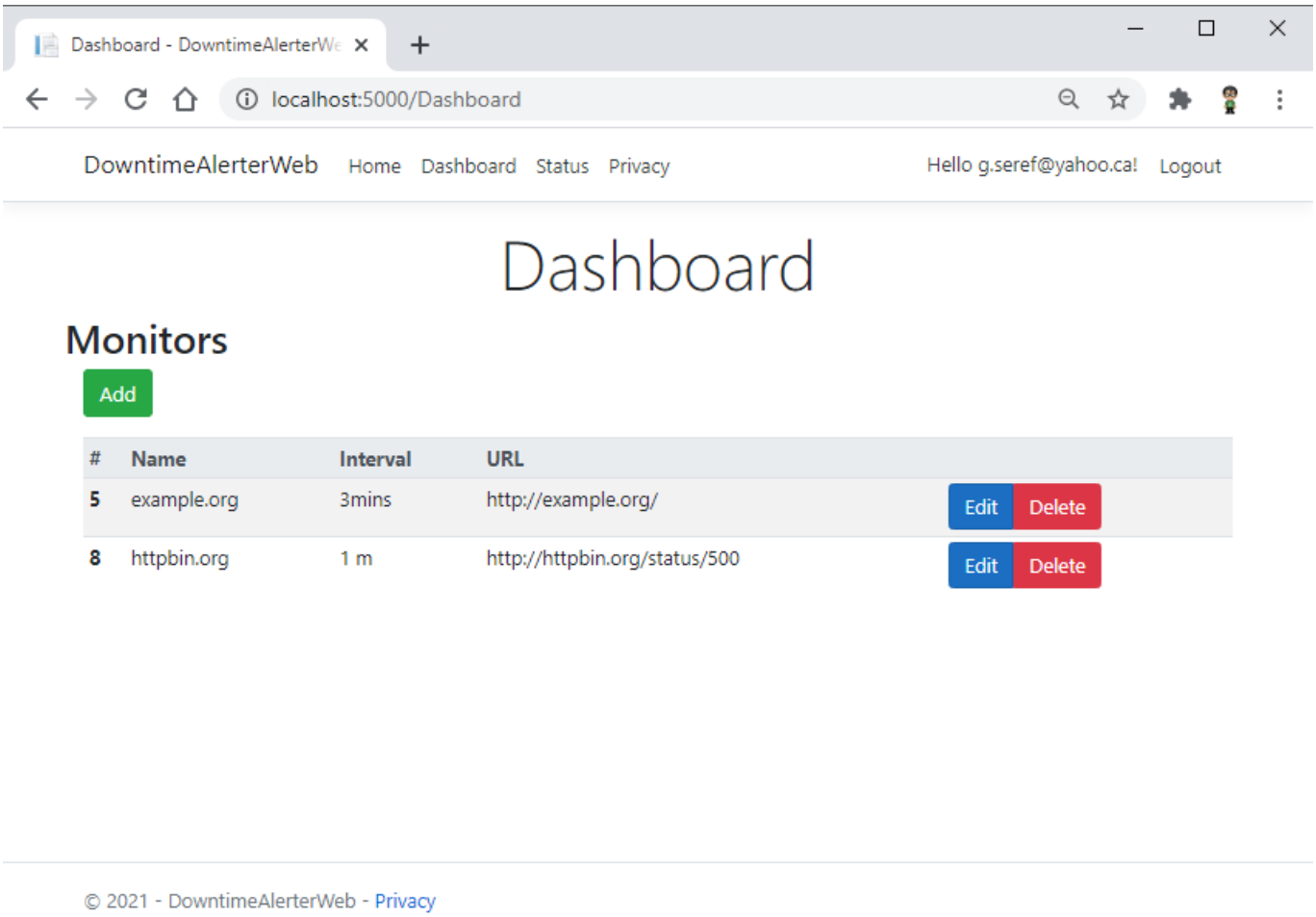
How Does It Work

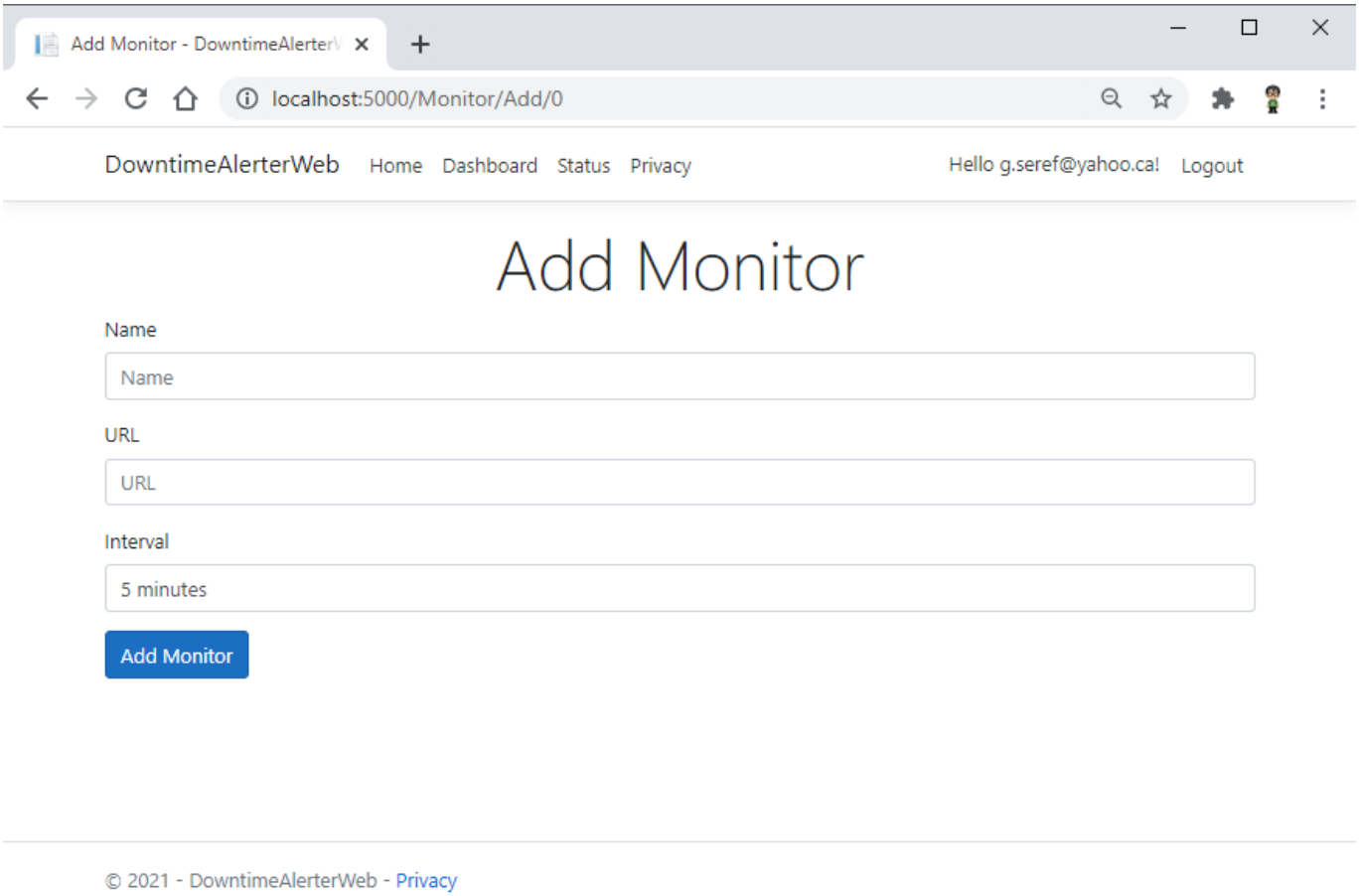
- User enters Name, URL an Inteval.
- Interval value extracted by a simple Regex parser.
- Interval resolution is `1 minute` and can be entered with a permissive syntax. For now only `minute` accepted. Kindly have a look at to the PoC: https://regex101.com/r/9p6frz/1

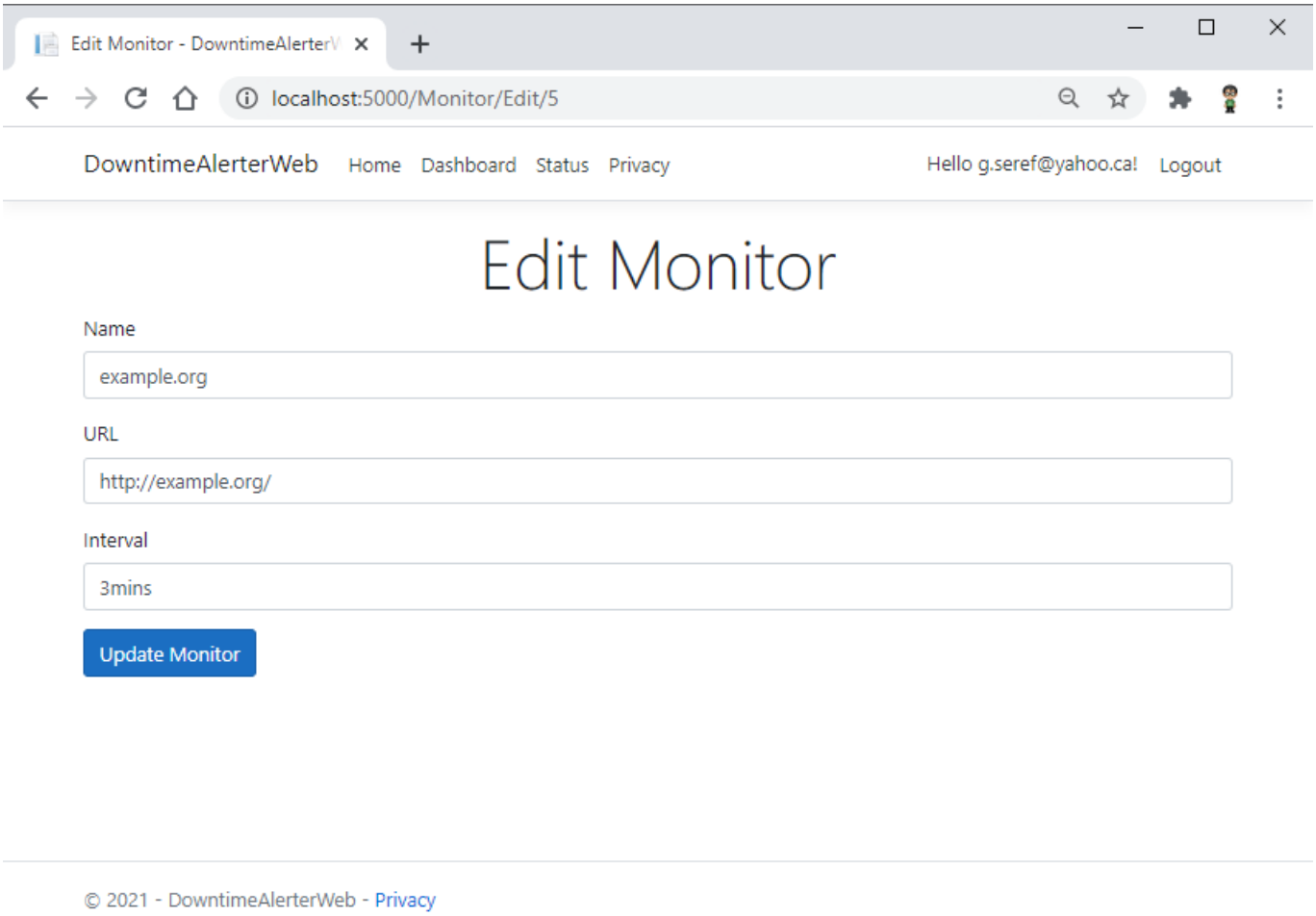
```
1m
1 m
1 min
1 mins
1 minutes
1   minutes
15 m
15 min
15 mins
15   minute
15 minutes
```

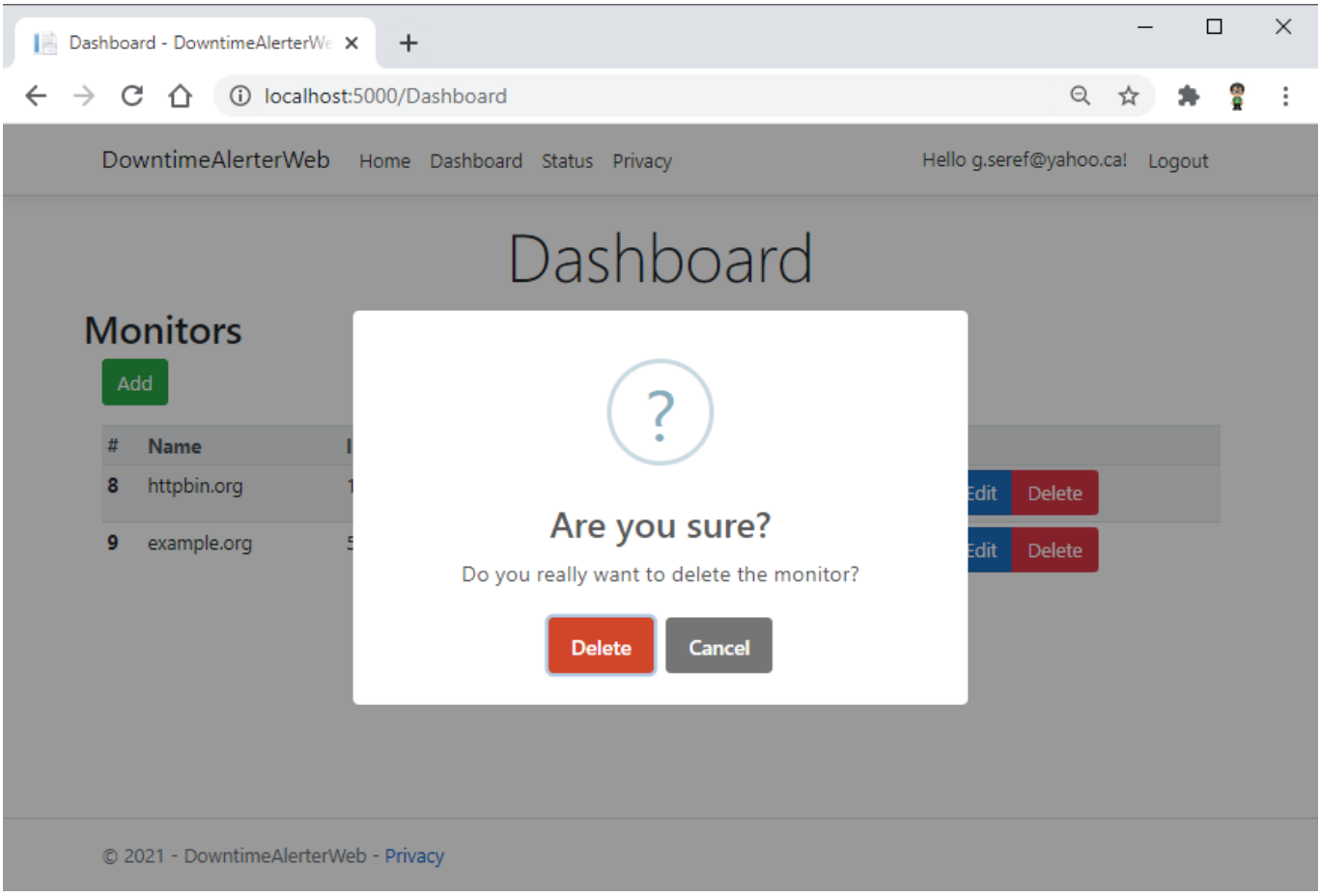
- **Background Service** executes the **Backlog** periodically each minute and fetches the all the monitor configurations,
- The **Backlog**, decides which monitor will be run at current execution by calculating the `MOD(Current Time, Interval)` and sends the task information to the Dispatcher.
- The **Dispatcher**, queues an async Http Request task to the **Task Queue**
- The **Task Queue**, executes the async tasks
- Async task results are broadcasted by `Mediator` design pattern
    - All task results, stored in the database by default **but** any new channel can be easily added the receivers by defining a class that implements: `INotificationHandler<ResponseNotification>`
    - Site down results, broadcasted to Email Notification handler by default, **but** any new channel can be easily added the receivers by defining a class that implements: `INotificationHandler<FailedResponseNotification>`

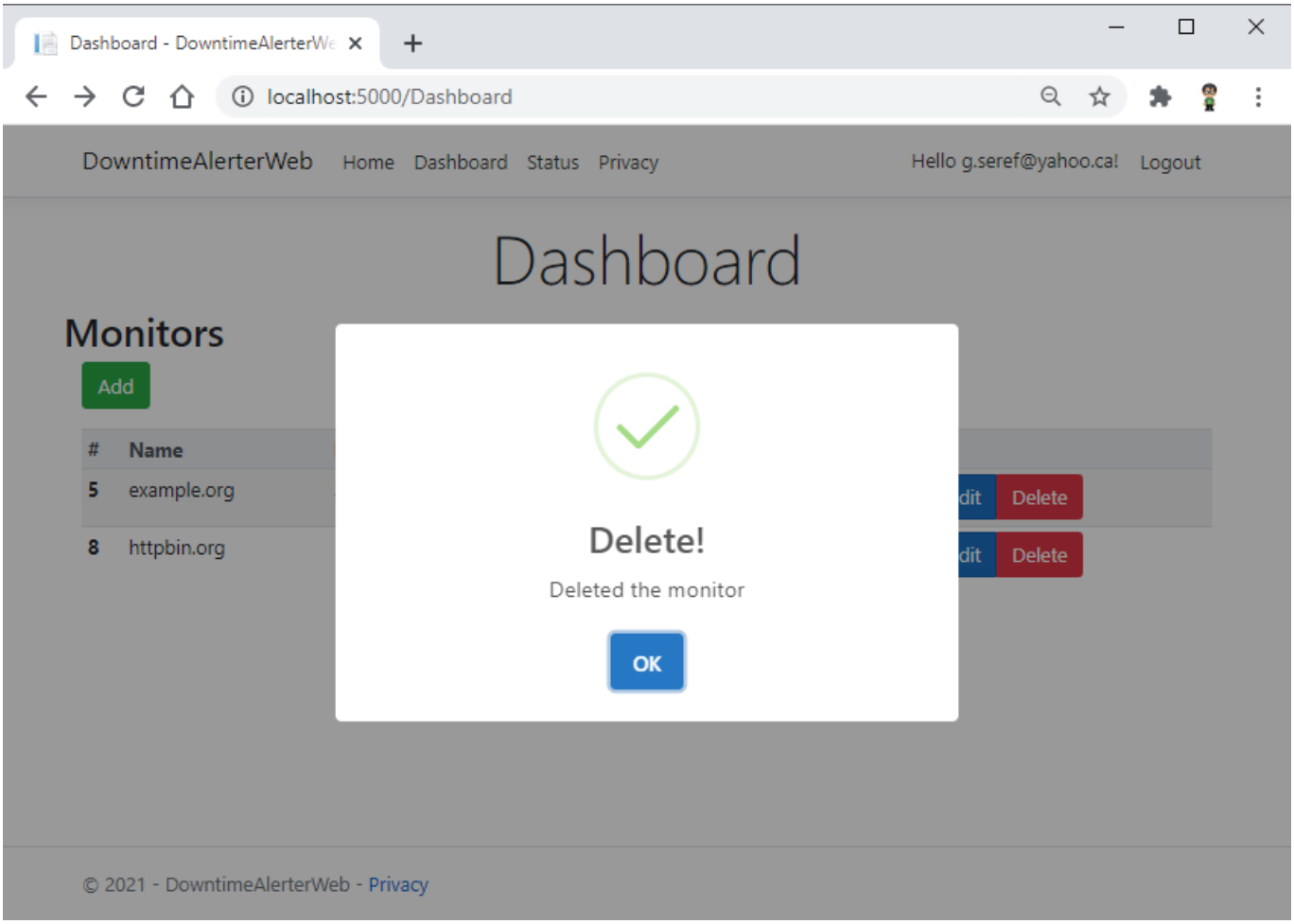
Screenshots

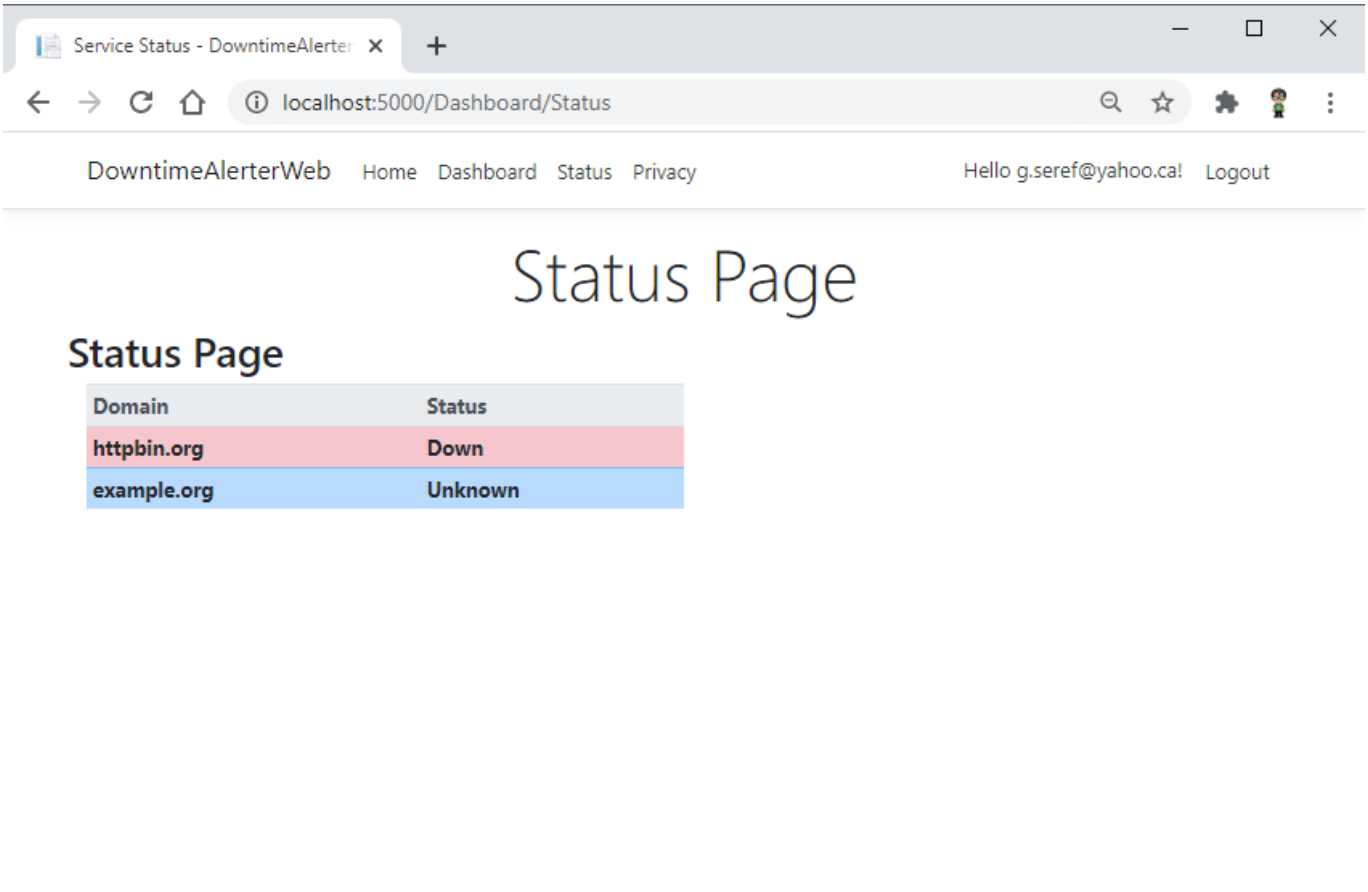

Dashboard - DowntimeAlerterWe
localhost:5000/Dashboard
DowntimeAlerterWeb
Home
Dashboard
Status
Privacy
Hello g.seref@yahoo.ca!
Logout
Dashboard
Monitors
Add
Name Interval URL
5 example.org 3mins http://example.org/ Edit Delete
8 httpbin.org 1 m http://httpbin.org/status/500 Edit Delete
© 2021 - DowntimeAlerterWeb - Privacy

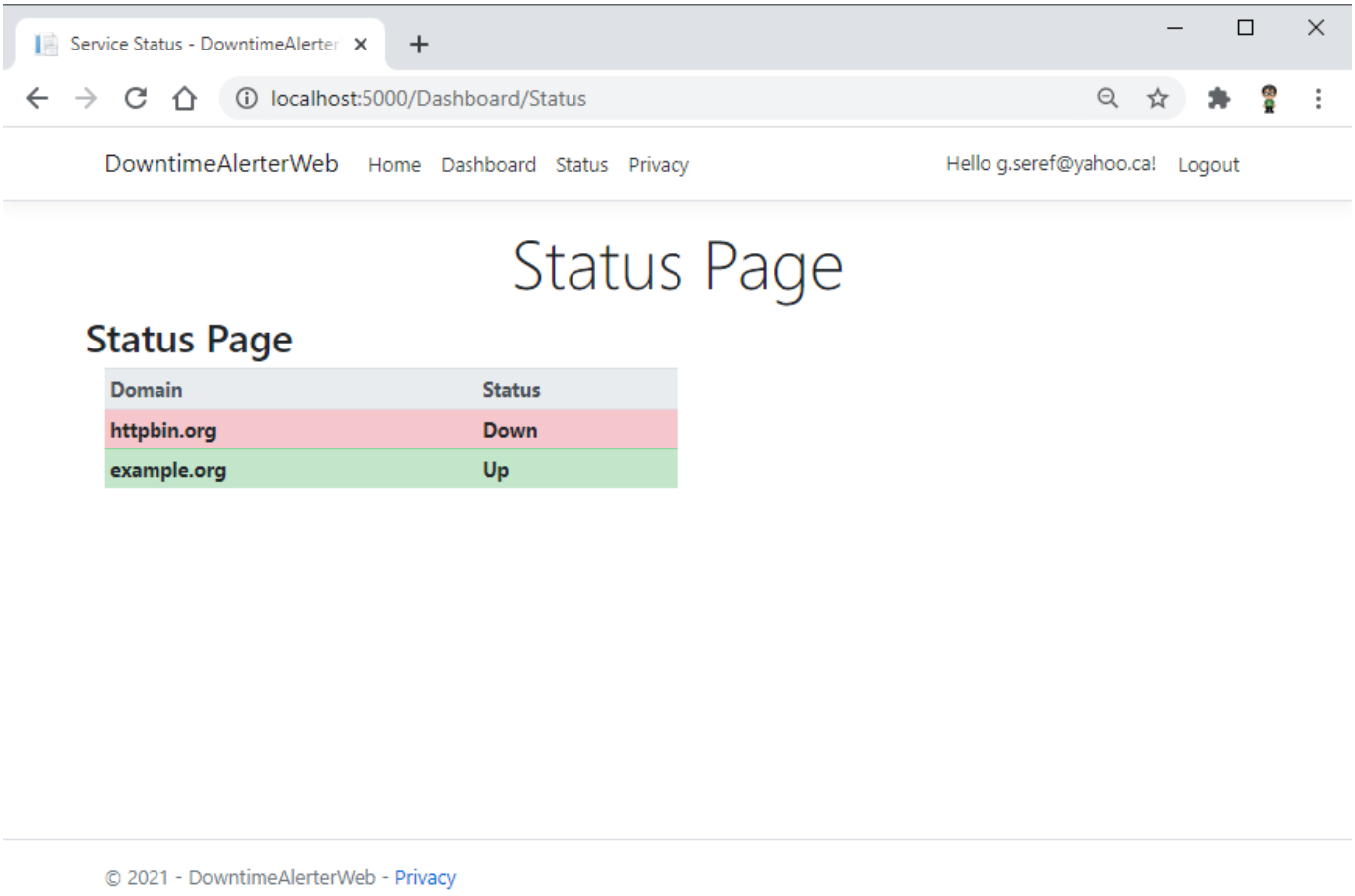

Add Monitor - DowntimeAlerterV
localhost:5000/Monitor/Add/0
DowntimeAlerterWeb
Home
Dashboard
Status
Privacy
Hello g.seref@yahoo.ca!
Logout
Add Monitor
Name
Name
URL
URL
Interval
5 minutes
Add Monitor
© 2021 - DowntimeAlerterWeb - Privacy

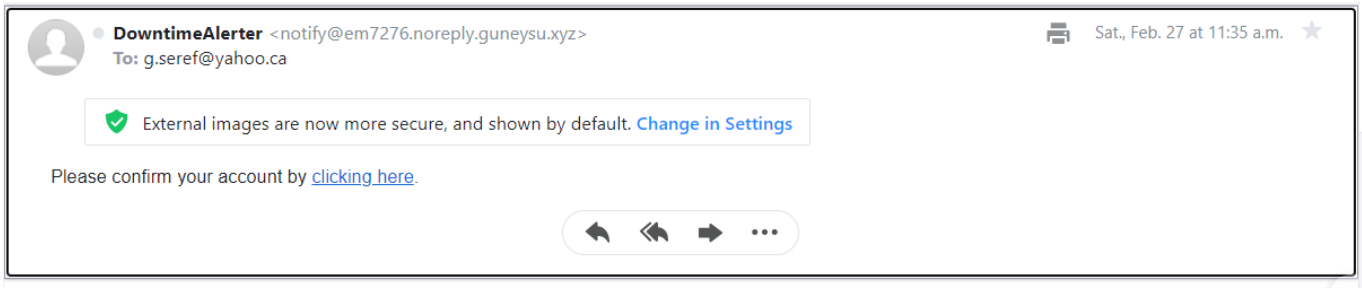

Edit Monitor - DowntimeAlerterW
localhost:5000/Monitor/Edit/5
DowntimeAlerterWeb
Home
Dashboard
Status
Privacy
Hello g.seref@yahoo.ca!
Logout
Edit Monitor
Name
example.org
URL
http://example.org/
Interval
3mins
Update Monitor
© 2021 - DowntimeAlerterWeb - Privacy


Dashboard - DowntimeAlerterWe
localhost:5000/Dashboard
DowntimeAlerterWeb
Home
Dashboard
Status
Privacy
Hello g.seref@yahoo.ca!
Logout
Dashboard
Monitors
Add
Name
8 httpbin.org
9 example.org
Edit
Delete
Edit
Delete
Are you sure?
Do you really want to delete the monitor?
Delete
Cancel
© 2021 - DowntimeAlerterWeb - Privacy


Dashboard - DowntimeAlerterWe
localhost:5000/Dashboard
DowntimeAlerterWeb
Home
Dashboard
Status
Privacy
Hello g.seref@yahoo.ca!
Logout
Dashboard
Monitors
Add
Name
5 example.org
8 httpbin.org
Delete
Delete
Delete!
Deleted the monitor
OK
© 2021 - DowntimeAlerterWeb - Privacy


Service Status - DowntimeAlerter
localhost:5000/Dashboard/Status
DowntimeAlerterWeb
Home
Dashboard
Status
Privacy
Hello g.seref@yahoo.ca!
Logout
Status Page
Status Page
Domain
Status
httpbin.org
Down
example.org
Unknown
© 2021 - DowntimeAlerterWeb - Privacy





References

- Initial layout, created by ASP.Net Core Web and Identity scaffolder tool
- Background services sample codes taken from: https://docs.microsoft.com/en-us/aspnet/core/fundamentals/host/hosted-services