Nanosoft

Requirements – Report Scheduler

Definition & Requirements Documentation

Date: 6 February 2017

Document Number: nano\_int\_req\_01

|  |  |
| --- | --- |
| PREPARED FOR: | PREPARED BY: |
| Graydon / Ryan / Arjun | David Gardner |
| Nanosoft | nanosoft.com.au |

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Changer | Date | Description |
| 1.0 | DG | 6/2/2017 | Initial Draft |

# Summary

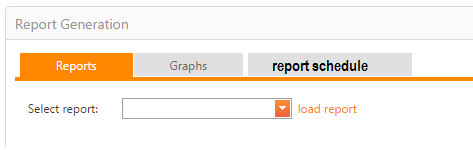
In the GPS application, we can currently create reports in the reporting section. However, we cannot schedule reports. We require a report scheduling feature which enables this which. It needs to do the following:

* Closely resemble the “windows task scheduler” which comes with windows 7+.
* Be able to dynamically allow the filling in of the reports parameters.

# REquirements

## REporting Schedule Tab

The report schedule tab must be placed as a tab in the reporting page similar to below:



### The tab must be between “graphs” and “report schedule”

## content of tab

Below is what the gridview in the tag should look like.



Columns and descriptions:

### New

* When clicking the “new” button, an inline edit box should be shown which allows the adding of new schedule.
* When clicking delete, an ‘are you sure’ dialogue box must be shown to check if the user actually wants to delete the item.
* The edit hyperlink will allow the user to edit the line item using an inline form.

*Please note that the edit and add forms content are covered in different sections throughout this document,**we will likely need to roll our own edit form and or wizard for this to work well.*

### Report

The report column must allow the selecting of a report. This must be a dynamic list of reports which are available to that user.

### Requestor

This Column **should not be visible in the edit and add forms** and is automatically filled in as formatted text of the user who added the scheduled report. It should be a free text field with the username written like “surname,forname”.

### Schedule

The schedule column needs to represent the frequency that the report will be generated. The following types need to be catered for:

* A one-off report (a date in the future, execute report once and that’s it)
* Daily report
* Weekly report
* Monthly report

#### Display

The schedule column should show the text representing the selected schedule (i.e.: “Weekly; At 3:30 PM every Sunday, Monday, Saturday of every week, starting 3/02/2021”)

#### Add / Edit Form

The below scree mock-ups are how we would like the report schedule creation to work.

**One off:**

*Use simple day / time selector.*

**Daily:**



**Weekly:**



**Monthly:**



### Parameters

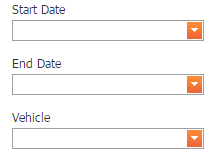
#### Display

The display of the parameters in the “normal view” of the form should be text based and show as something like below:

Start Date: beginning of week  
End Date: current date  
Vehicle: Vehicle A

#### Add / Edit form

This is tricky, ideally we would want the same as what is in the report generator, like below:



We would need to save the selection as something like XML or what-ever is most appropriate.

##### Dealing with dates:

The one caveat is to do with dates, each data must be able to have the following options:

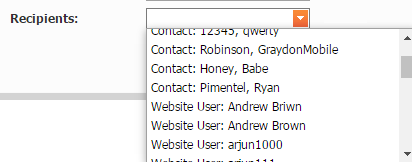
* Select a date / time
* Beginning of week
* Date of report
* Beginning of year
* **Graydon can you think of anything else here**

### Recipients

The recipient’s column must work the same as the following control:

Geofence & alerts > Geofence Alert configuration > New > Message Destination

An example is below:

:

Please note that the recipients list should be a combination of contacts, groups, website users etc etc. The business logic can be reverse engineered from the above mentioned control.

We must also have the ability to add a new contact within the recipient’s selection. This would work like we wanted the booking page to work.

## BAck end Logic

There will need to be a completely new reporting service created for this. The reasoning for this is that in experience, reporting services can take up a lot of resources as product usage matures. If it becomes resource intensive we will want the ability to move the service to a different box. We can also sell that we have a “reporting services server”.

The new reporting services service will have to adhere to the following requirements:

* Use the same “business logic” as the web form (can be a copy of the DLL).
* Be another project in the FMS solution.
* WILL use the nugget package “topshelf”. There is a good resource here:  
  <https://topshelf.readthedocs.io/en/latest/configuration/quickstart.html>
* There will have to be logging applied using NLog, good resource can be found here:  
  <http://nlog-project.org/>
* It is OK to write this in C#, in fact this is preferable and something we should have done from the start (my bad).

## Email

### Content

The email which is sent will be in the following format:

From: [no\_reply@nanosoft.com.au](mailto:no_reply@nanosoft.com.au)

To: <<list of recipients>>

Dear <<forname>> <<surname>>,

Please find attached the <<report name>> report generated on <<date in format dd/MMM/yyyy HH:mm:ss>>.

This was generated by the Nanosoft GPS report generator, if you would like to view more then please click here: <<link to their specific maps page i.e. cannon.nnaosoft.com.au/fleetmap.aspx>>

Thank you,



### Attachment

The email attachment should be a PDF and the name of the PDF should be in the below format:

<<report\_name>>\_<<date in format YYYYMMdd>>.pdf

# Development Rollout

This is only suggested but I think something like the following would make sense: