

Point 1

When enabled (aka the fields filled) produce a RAR file from local/network folder.

Point 2

Backups the original JSON files (database/rules) from [firebase](#) instance

Point 3

User has to upload a custom PHP script (appears when 'script template' clicked), name it as he likes... Then has to enter the URL of this PHP file to textbox! Pick two random passwords (should filled also on PHP file), in big textbox has to write the folder name(s) are near this PHP. Example, I store the file there :

`\public_html\test\testSITE\LILI\test.php`

This folder has sub-folders

`\public_html\test\testSITE\LILI\test2`

`\public_html\test\testSITE\LILI\test3`

`\public_html\test\testSITE\LILI\test2\css`

On desktop program, valid folder names, are :

test2

test3

test2/css (here backups only **css** folder)

Point 4

User has to upload a custom PHP script (appears when 'script template' clicked), name it as he likes... Then has to enter the URL of this PHP file to textbox! This one backups the TABLES schema and PROCEDURES not the rows....

Tip

User can automate the backup process when passing the jobSET filename via command line, example :

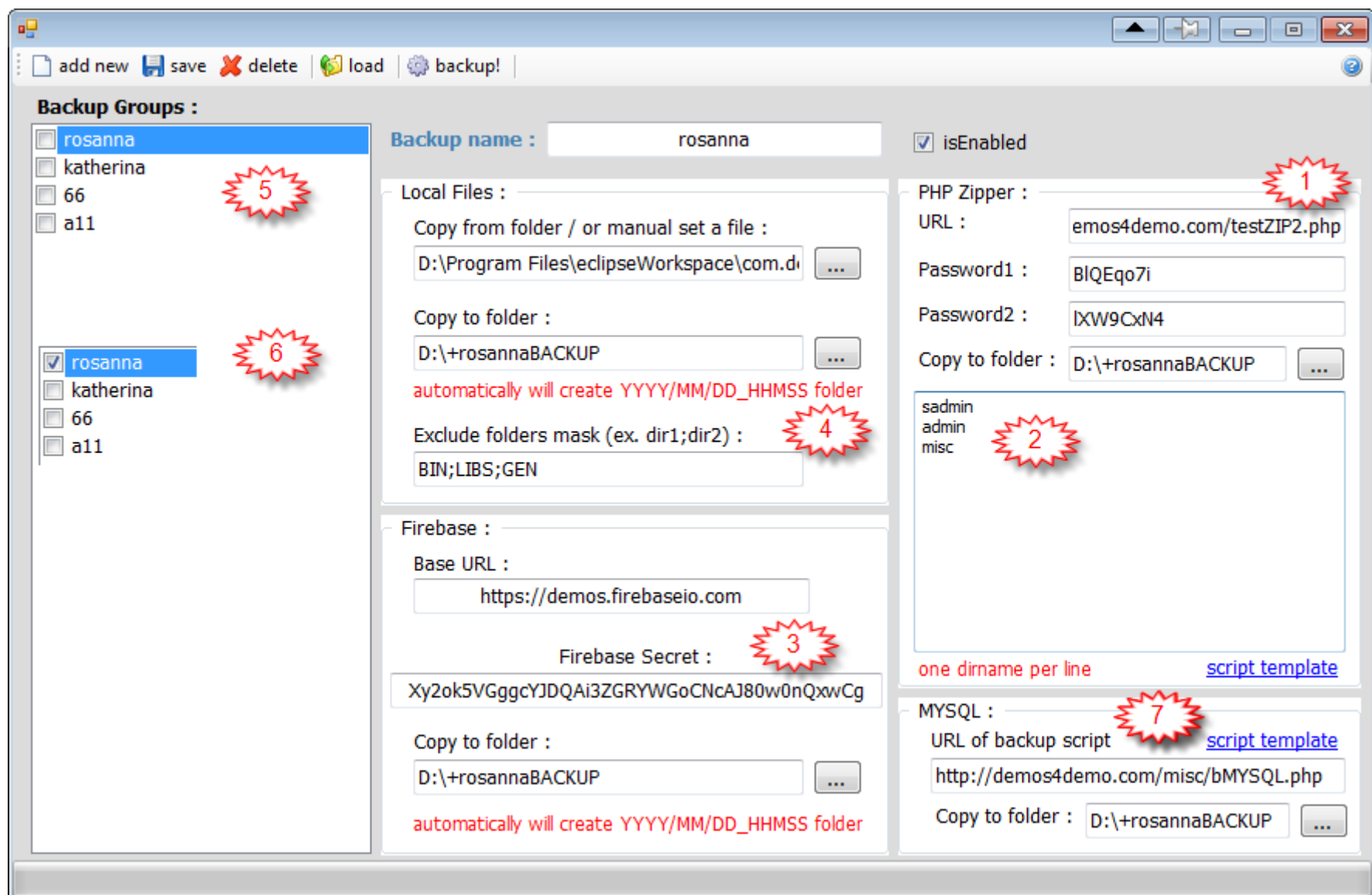


DeveloperBackup.exe

Target type: Application

Target location: DeveloperBackup

Target: `veloperBackup\DeveloperBackup.exe" SAP.json`



Point 1

The complete URL to the script template explained before, the two password texts at PHP file and to desktop program must be equal otherwise an error will occurred.

```
if($_GET["pass_one"] != "BlQEo7i" || $_GET["pass_two"] != "lXW9CxN4"){
    header($_SERVER['SERVER_PROTOCOL'] . ' 500 Internal Server Error', true, 500);
    exit;
}
```

Point 2

The folder names are near **testZIP2.php**

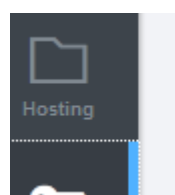
Index of /public_html

Name

- [parent directory]
- admin/
- misc/
- sadmin/

your PHP file here (aka testZIP2.php)

Point 3



You can generate Firebase authentication tokens on your own server.



Point 4

Source & destination locations and folder name exclusions

Point 5

When toolstrip 'backup!' item clicked, by default when nothing is checked from backupjoblist, will backup all jobs where isEnabled=true. On the other way (Point 6) when one or more items are checked and 'backup!' item clicked will proceed only with checked jobs!

Point 7

The complete URL to the script template explained before, there is no passwords, because download only the DB schema...