

Dokumentation Projekt LogBonk

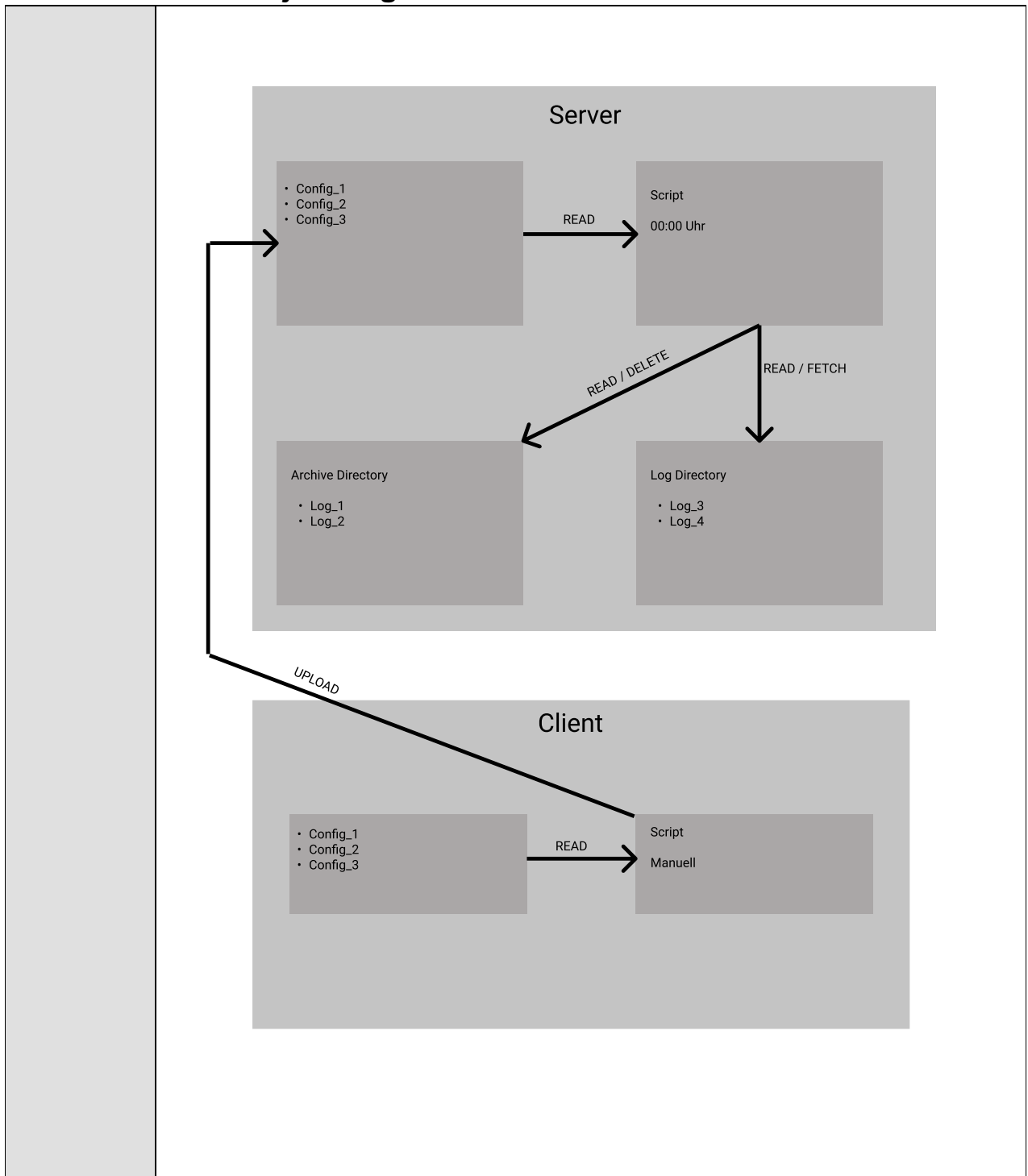
1 Teilnehmer/innen des Teams:

Klasse: AP18c	Team: Danilo Jakob Stefan Ninkovic
------------------	--

2 Anforderungsdefinition (Meilenstein A)

„LogBonk“	
Fachlicher Inhalt: (Allgemeine Beschreibung)	<p>Nutzen: Mit dem Programm sollen Logdateien verarbeitet und verwaltet werden.</p> <p>Automation:</p> <p>Das Programm verwaltet Logdateien anhand von ihren Alter, dass heisst sie werden automatisch archiviert, und nach einen Monat aus dem Archiv gelöscht.</p> <p>Das Programm verarbeitet aktive Logdateien, es schickt ein Mail anhand von Filter welche in einer Konfigurationsdatei definiert worden sind.</p> <p>Ein zweites Script ermöglicht es verschiedenen Benutzern Konfigurationsdateien über FTP zum Server zu schicken.</p> <p>Details:</p> <ul style="list-style-type: none">• Konfiguration: Das Verhalten kann mit einer oder mehreren Konfigurationsdateien verändert werden.• Integration: Um 00:00 wird das Main Script auf dem Server automatisch aufgerufen.• Administration: Ein Benutzer wird auf dem Server erstellt, welcher nur auf die Dateien bzw. Ordner zugreifen kann, welche für die Funktionalität benötigt werden.• Sicherheitsaspekte: Mit dem Benutzer wird sichergestellt, das niemand mit den Zugangsdaten auf andere sensible oder kritische Dateien vom Server zugreifen kann.

Dokumentation Projekt LogBonk



Dokumentation Projekt LogBonk

MUSS Kriterien: (Konkrete Features, die umzusetzen sind)	Folgende Features sollen implementiert werden, um einen produktiven Ablauf sicherzustellen: <ul style="list-style-type: none">• Log archivieren• Mail senden, wenn Suchwörter gefunden werden• Konfigurationsdateien in einem Server-Ordner speichern• Logs zu einer bestimmten Zeit durchsuchen
--	--

Dokumentation Projekt LogBonk

KANN Kriterien: (Konkrete Features, die optional sind)	Folgende Features können zusätzlich implementiert werden: (Varianten, Kreativität) <ul style="list-style-type: none"> • Key in der Konfigurationsdatei, um die Dateien zu verifizieren • Konfigurationsdatei für Main Script
--	---

2.1 Planung Meilensteine (LB1 / LB2)

MS	Tätigkeit / Abgabe	Soll-Datum	Ist-Datum
A	Projektstart <ul style="list-style-type: none"> ➤ Team Bildung, Kollaborationsplattform, GitHub Repos *, Lehrerzugang ➤ Wahl / Ausarbeitung der Anforderungsdefinition Abnahme Anforderungsdefinition durch Lehrperson	21.05.2021	21.05.2021
B	Teamaufgabe 1: <ul style="list-style-type: none"> ➤ Abgabe: Lösungsdesign (Funktionsmodell / GUI / PAP / Storyboard) 	28.05.2021	28.05.2021
B2	Teamaufgabe 2: (Nur LB2) <ul style="list-style-type: none"> ➤ Abgabe: Testvorschrift und Testfälle 	X	X
C	Einzelaufgabe 2 (LB1) / 3 (LB2): <ul style="list-style-type: none"> ➤ Abgabe Programmcode und Dokumentation ➤ Fachgespräch Projektabnahme 		
C2	Einzelaufgabe 4: (Nur LB2) <ul style="list-style-type: none"> ➤ Abgabe: Ausgefüllter Systemtest 		

*) Öffentliche GitHub-URLs im Ablageordner auf dem BSCW ablegen! (pro Team)

Namenskonvention URL: **M122_Klasse_Thema_Name_Name**

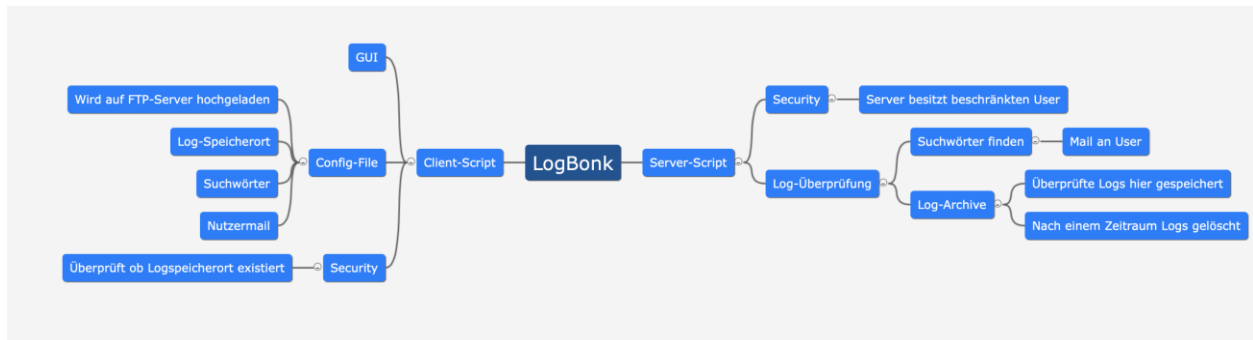
Dokumentation Projekt LogBonk

3 Lösungsdesign (Meilenstein B: Teamaufgabe 1)

Anhand der Analyse wurde folgendes Lösungsdesign entworfen:

3.1 Schematische Darstellung der Funktionalität, sog. Funktionsmodell

Im Folgenden ist die erwartete Funktionalität dargestellt und erklärt:



3.2 Graphische Benutzer Schnittstelle (GUI) zur Konfiguration des Ablaufs

Das zu erwartende GUI ist hier dargestellt:

LogBonk

Config-Name

Search-Words (seperated by ,)

E-Mail

Log-Path

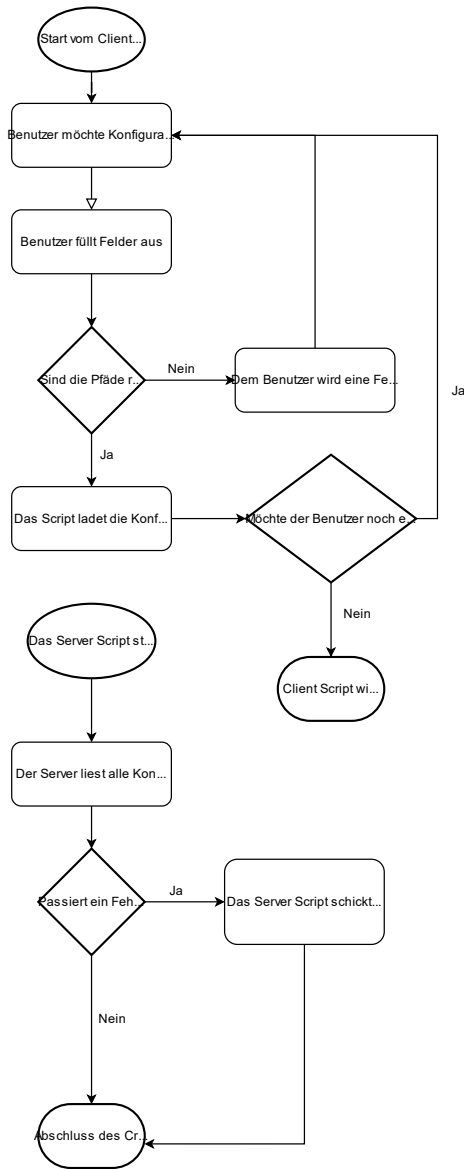
2021, Danilo Jakob & Stefan Ninkovic, M122

UPLOAD

Dokumentation Projekt LogBonk

3.3 Ablauf der Automation

Aus Benutzersicht ist folgender Ablauf des Programms zu erwarten:



Viewer does not support full SVG 1.1

(Flussdiagramm (PAP / APIs: UML Aktivitätsdiagramm) / Storyboard)

Dokumentation Projekt LogBonk

4 Testvorschrift (LB2 Meilenstein B2: Teamaufgabe 2)

Testbeschreibung und vorbereitetes Testprotokoll siehe Dokument
[*M122_LB2_Testvorschrift_MS-B2_Namen.docx*](#)

5 Testprotokoll (LB2 Meilenstein C2: individuelle Aufgabe 4)

Ausgefülltes Testprotokoll siehe Dokument
[*M122_LB2_Testvorschrift_MS-C2_Namen.docx*](#)

Dokumentation Projekt LogBonk

6 Systemdokumentation (Meilenstein C: individuelle Aufgabe 3)

Die erstellten Projekt-WPS-Scripts sind hier abgelegt und für Entwickler dokumentiert:

Öffentliche GitHub-URLs im Ablageordner auf dem BSCW ablegen! (pro Team)

Namenskonvention URL: **M122_Klasse_Thema_Name_Name**

→ Ein **Branch** und separater Doku-Ordner pro Teammitglied erstellen

6.1 Umfang / Abgrenzung / Änderungen gegenüber Design

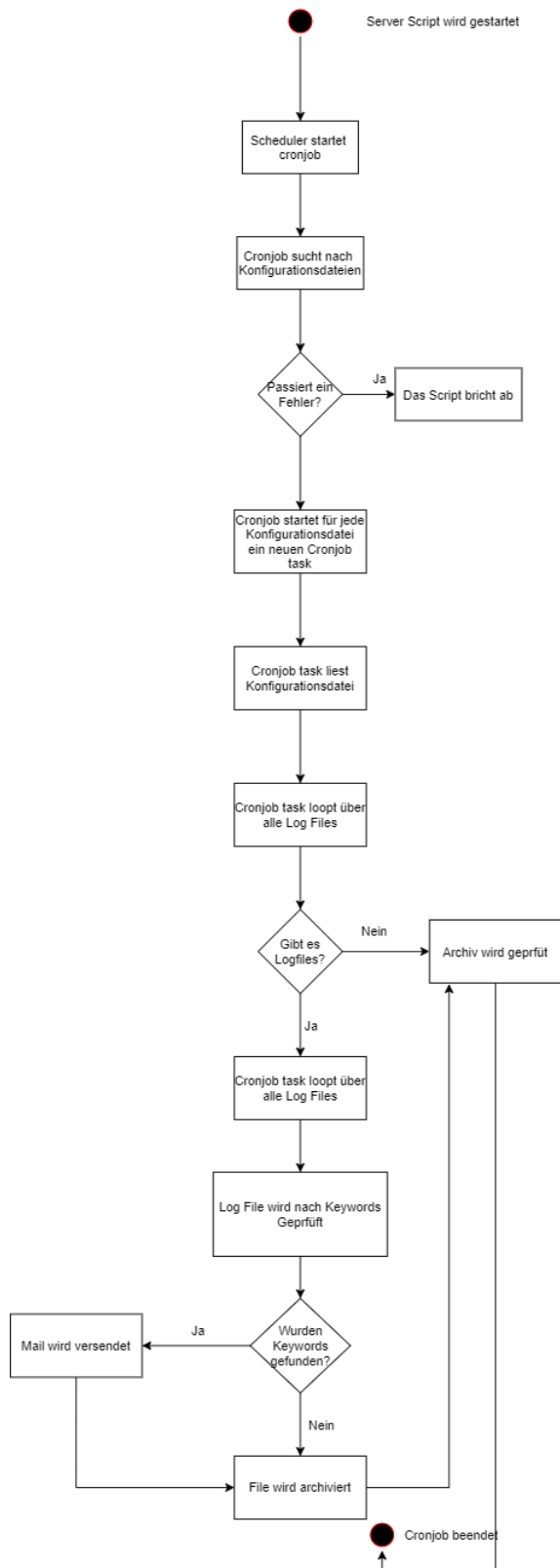
Aufgrund unten beschriebener Umstände sind Anpassungen des ursprünglichen Lösungsdesigns gemacht worden:

...

Umstände / Anpassungen / Veränderungen

Dokumentation Projekt LogBonk

6.2 Funktionalität der Implementation.



Dokumentation Projekt LogBonk

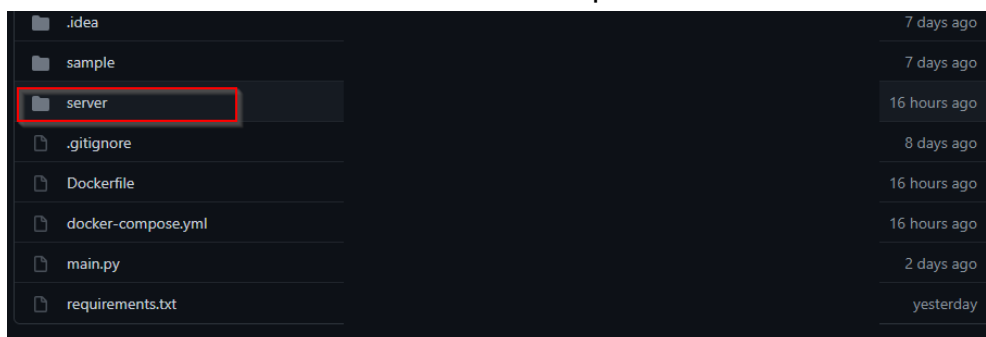
7 Betriebsdokumentation (Meilenstein C: individuelle Aufgabe 3)

Für Administratoren steht eine Anleitung zur Verfügung, in welcher beschrieben wird wie das LogBonk Server Script auf einem Server installiert und konfiguriert werden muss.

Für den Benutzer steht auch eine Anleitung zur Verfügung, hierbei werden nur die Eingabefelder und deren Inhalt erläutert.

7.1 Installationsanleitung für Administratoren

Zum das Server Script für LogBonk zu installieren, reicht es, wenn man das «server.py» File vom Server Ordner auf dem Server speichert:



Es muss sichergestellt werden, dass auf dem Server Python 3.8 installiert ist, somit kann das Script danach auch gestartet werden. Ausserdem muss noch die Schedule Library installiert werden. Falls pip nicht installiert ist, dies kann der Fall bei Linux distros sein, muss noch erst pip mit folgendem Befehl installiert werden: «apt install python3-pip». Danach kann man die Library mit folgendem Befehl installieren: «pip install schedule».

Zum das Script zu konfigurieren, müssen folgende Schritte gemacht werden:

- Der Pfad für die Konfigurationsdateien muss im Code angegeben werden

```
config_files = os.listdir("./config")
for config_file in config_files:
    fqcn = "./config/" + config_file
    cron_job_task(fqcn)
```

- Es muss ein Feil im gleichen Ordner wie das Server Script erstellen, welches „email_config.py“ heisst, dies wird für das Mail versende benötigt und muss wie folgt aussehen:

Dokumentation Projekt LogBonk

```
smtp_server=""  
smtp_username=""  
smtp_password=""  
smtp_port=0
```

- Der Scheduler im Code muss auch noch angepasst werden, damit der Service bei eigener Zeit startet. Dies sieht wie folgt aus:

```
schedule.every(24).hours.do(cron_job)
```

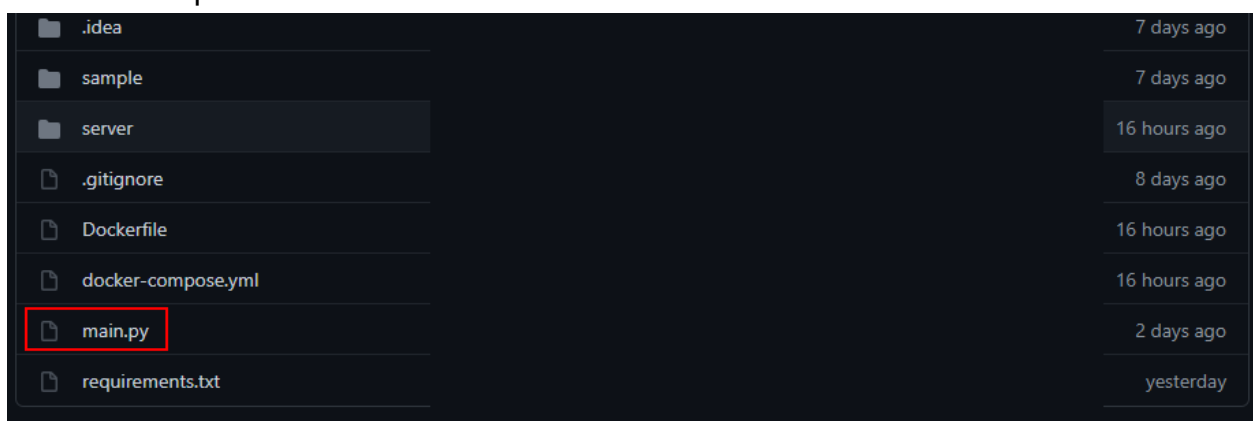
Falls man nicht stündlich das Script ausführen möchte, kann man im code „hours“ zu „minutes“, „seconds“ und „days“ setzen.

Wenn diese Einstellungen gemacht wurden, kann man das Script laufen lassen mit folgendem Befehl: «python3 <PFAD_ZUM_SCRIPT>/server.py».

Wenn alles korrekt konfiguriert wurde, wird der cronjob in den angegebenen Zyklus aufgerufen.

7.2 Bedienungsanleitung für Benutzer

Zum das Client Script von LogBonk zu installieren, muss nur das «main.py» script auf den Client kopiert werden:



Es muss sichergestellt werden, dass auf dem Server Python 3.8 installiert ist, somit kann das Script danach auch gestartet werden.

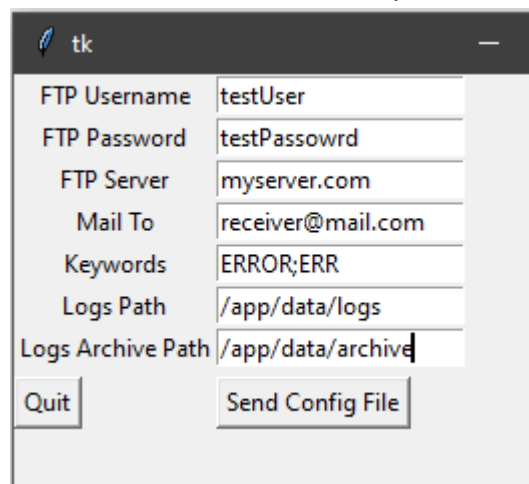
Dokumentation Projekt LogBonk

Das Einzige was konfiguriert werden muss, ist der Pfad, an den die Konfigurationsdateien gespeichert werden. Dies muss an folgender Stelle im Code angepasst werden:

```
5 ftp.cwd("/app/data/config")
```

Danach kann das Script mit folgendem Befehl gestartet werden: «python3 main.py».

Nach dem Starten des Scripts sollte folgendes Fenster erscheinen (mit leeren Feldern):



FTP Username	testUser
FTP Password	testPassowrd
FTP Server	myserver.com
Mail To	receiver@mail.com
Keywords	ERROR;ERR
Logs Path	/app/data/logs
Logs Archive Path	/app/data/archive

Quit Send Config File

Das obige Fenster wurde schon ausgefüllt. Hier müssen nur die FTP Credentials mitgegeben werden, sprich Benutzername, Password und der Hostname vom FTP Server.

Danach muss die E-Mail-Adresse angegeben werden, an welche die Alert Mails geschickt werden.

Die Keywords sind die Filter-Wörter bei welchen LogBonk, falls diese im Log File gefunden werden, ein Mail verschickt wird. Hier kann man mehrere angeben, diese müssen jedoch mit einem Semikolon getrennt werden.

Zum Schluss muss noch angegeben werden, wo sich die Log Files, welche man verarbeiten möchte, befinden und wo die Logs archiviert werden sollen.

Wenn alle Informationen korrekt eingegeben wurden, kann man die Konfigurationsdatei am Server schicken, in dem man auf «Send Config File» drückt.