# Teilnehmer/innen des Teams:

|  |  |
| --- | --- |
| Klasse:  AP18a | Team:  Jonuzi- Steffen |

# Anforderungsdefinition (Meilenstein A)

|  |  |
| --- | --- |
| „Spicy-stock“ | |
| **Fachlicher Inhalt:**  (Allgemeine Beschreibung) | Nutzen: Mit dem Programm Spicy-stock kann ein User seine Favoriten Aktien beobachten. Jeder Morgen wird automatisch eine Nachricht per Telegram geschickt die die Werte der Favoriten Aktien zeigt. Man kann auch ein Wert (min. und max.) definieren der, wenn überschneidert wird das User benachrichtigt. **Automation:**  Jede Stunde (Ausser Wochenende) werden Daten von Yahoo-Finance API geholt, diese werden analysiert und falls nötig wird das User automatisch per Telegram benachrichtigt  **Details:**   * Konfiguration:   Die Namen der Aktien die Beobachtet werden mit dem Min. und Max. Wert für eine Benachrichtigung. Uhrzeit der Tägliche Benachrichtigung. Jedes Mal wo PC eingeschaltet wird ein bash-skript lauft der Spicy-stock startet.   * Integration:   Python  Yahoo-finace library für Aktien Data  Telethon library für Telgram message schicken   * Administration:   Log Files werden generiert, falls ein Fehler bei der Automatisierung gibt.   * Sicherheitsaspekte:   Yahoo-finance ist Open-source Library  Telegram Token und api hash und api id von [https://my.telegram.org](https://my.telegram.org/) |
| **MUSS**  **Kriterien:**  (Konkrete Features, die umzusetzen sind) | **Folgende Features sollen implementiert werden, um einen produktiven Ablauf sicherzustellen:**   * Banachritigung jeden Tag mit dem Aktienwert von gewünschte Firmen * Benachritigung bei min. max. Wert erreicht * Jede Stunde daten Holen und analysieren * Benachritigung durch Telegram * Das Programm muss laufen wann das PC auch lauft. |

|  |  |
| --- | --- |
| **KANN**  **Kriterien:**  (Konkrete Features, die optional sind) | **Folgende Features können zusätzlich implementiert werden: (Varianten, Kreativität)**   * Log files werden generiert für administratives zweck * Benachtigung bei Trends (Wenn ein Aktie gut dran ist) |

## Planung Meilensteine (LB1)

|  |  |  |  |
| --- | --- | --- | --- |
| *MS* | *Tätigkeit / Abgabe* | *Soll-Datum* | *Ist-Datum* |
| A | ProjektstartTeam Bildung, Kollaborationsplattform, GitHub Repos \*, LehrerzugangWahl / Ausarbeitung der Anforderungsdefinition Abnahme Anforderungsdefinition durch Lehrperson | 02.02.2021 | 02.02.2021 |
| B | Teamaufgabe 1:Abgabe: Lösungsdesign |  |  |
| C | Teamaufgabe 2:Implementation |  |  |
| D | Teamaufgabe 3:Testen |  |  |
| C | Teamaufgabe 4:Abgabe Programmcode und DokumentationFachgespräch Projektabnahme |  |  |

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

Namenskonvention URL: **M122\_Klasse\_Thema\_Name\_Name**

# Lösungsdesign (Meilenstein B: Teamaufgabe 1)

Anhand der Analyse wurde folgendes Lösungsdesign entworfen:

## Schematische Darstellung der Funktionalität, sog. Funktionsmodell

Im Folgenden ist die erwartete Funktionalität dargestellt und erklärt:

...

(**Funktionsmodell**: Skizze, Bild, Pictogramm, Mindmap, Blockdiagramm, UseCase (API) zur obigen Anforderungsdefinition **mit Legende**)

## Ablauf der Automation

Aus Benutzersicht ist folgender Ablauf des Programms zu erwarten:

...

Aus Administratorsicht ist folgender Ablauf des Programms zu erwarten:

...

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

# Testvorschrift (LB2 Meilenstein B2: Teamaufgabe 2)

Testbeschrieb und vorbereitetes Testprotokoll siehe Dokument   
***M122\_LB2\_Testvorschrift\_MS-B2\_Namen.docx***

# Testprotokoll (LB2 Meilenstein C2: individuelle Aufgabe 4)

Ausgefülltes Testprotokoll siehe Dokument   
***M122\_LB2\_Testvorschrift\_MS-C2\_Namen.docx***

# 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*

## Umfang / Abgrenzung / Änderungen gegenüber Design

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

...

Umstände / Anpassungen / Veränderungen

## Funktionalität der Implementation.

Zusätzlich zu der Inline-Dokumentation sind hier folgende Funktionen / Eventhandler detailliert beschrieben:

...

Ausführliche Beschreibung der internen Funktionen (Eventhandler wie z.B: $btn\_ok\_Click), der Parameter und der Rückgabewerte

**Struktogramm** (BET / SYS) oder **UML Aktivitätsdiagramm** (API)

# Betriebsdokumentation (Meilenstein C: individuelle Aufgabe 3)

Für Administrator und Benutzer wird folgende Anleitung ausgeliefert ...

## Installationsanleitung für Administratoren

Das Programm ist folgendermassen zu installieren und konfiguriert ...

## Bedienungsanleitung für Benutzer

Das Programm ist folgendermassen zu bedienen ...