

Story Name	Sprint L4 – Webseite	Dauer	6 Wochen
Informelle Beschreibung	Als Mitarbeiter von Plants in Focus S.A.		
	... möchte ich eine Webseite für das Administrieren des Systems entwickeln		
	... damit meine Kunden das Plantimeter-System konfigurieren können.		
Anforderungen laut Arbeitsauftrag	<ul style="list-style-type: none"> <li>• Must <ul style="list-style-type: none"> <li>○ Die Webseite ist grafisch ansprechend</li> <li>○ Die Webseite ist mit für ein 1080p Display ausgelegt.</li> <li>○ Die Design-Richtlinien für Farbpalette, Logo und Schriftart sind respektiert.</li> <li>○ Die Webseite ist nur für eingeloggte Benutzer erreichbar.</li> <li>○ Der Admin-Benutzer ist den Anforderungen nach erstellt.</li> <li>○ Der Admin- Benutzer kann: <ul style="list-style-type: none"> <li>▪ Das eigene Passwort ändern</li> <li>▪ Benutzer hinzufügen und löschen und Passwörter neu setzen</li> <li>▪ Nodes an Benutzer vergeben</li> <li>▪ Nodes löschen</li> <li>▪ Pflanzenarten löschen</li> </ul> </li> <li>○ Benutzer können: <ul style="list-style-type: none"> <li>▪ Das eigene Passwort ändern</li> <li>▪ Pflanzenarten hinzufügen und Schwellen &amp; Richtwerte editieren</li> <li>▪ Ihren Nodes jeweils eine Pflanzenart zuweisen. Das Datum wird korrekt gesetzt</li> <li>▪ Den Status &amp; die Messwerte seiner Nodes über einen bestimmten Zeitraum als Diagramm anzeigen</li> <li>▪ Aufgaben zum Bewässern oder Beleuchten einer Node zu einem bestimmten Zeitpunkt registrieren</li> </ul> </li> <li>○ Das Log-Backend Skript empfängt und schreibt die Logs wie definiert.</li> <li>○ Das Automatisation Skript ist nach Angabe erstellt und kann manuell ausgeführt werden.</li> </ul> </li> <li>• Should <ul style="list-style-type: none"> <li>○ Benutzer können die Logs ihrer Nodes einsehen.</li> <li>○ Webseitenteile welche Messwerte anzeigen aktualisieren sich automatisch in konstanten Intervallen.</li> </ul> </li> <li>• Could</li> </ul>		
Aufgaben laut Arbeitsauftrag			
Erstelle ein einfaches Mockup für deine Webseite.			

Setze eine Entwicklungsumgebung auf zB. Xampp, Uniserver, Raspberry Pi auf.
Programmiere die Login-Seite
Erstelle ein php-Skript, welches nicht authentifizierte Benutzer zur Login-Seite weiterleitet.
Füge den „admin“ Benutzer manuell hinzu und Teste, ob er sich korrekt einloggen kann.
Erstelle eine Template-Webseite mit CSS, auf welcher du die weiteren Webseiten basieren wirst. Stelle sicher, dass du die Design-Anforderungen respektierst.
Erstelle die Webseite, auf welcher ein Benutzer sein Passwort ändern kann.
Erstelle die Webseite, auf welcher der Admin andere Benutzer verwalten kann.
Erstelle die Webseite, auf welcher der Admin Nodes verwalten kann.
Erstelle die Webseite, auf welcher ein Benutzer eine Pflanzenart hinzufügen und ihre Schwellen und Richt-Werte angeben setzen kann.
Erstelle die Webseite, auf welcher der Admin Pflanzenarten löschen kann.
Erstelle die Webseite, auf welcher ein Benutzer seine Nodes einsehen/zuweisen kann.
Erstelle die Webseite, auf welcher ein Benutzer Status und Messwerte einer Node in Form eines Diagramms einsehen kann.
Erweitere diese Webseite mit dem Formular, um Aufgaben zu registrieren und programmiere deren Backend.
Erstelle ein Diagramm pro Messwertreihe einer Node mit Chartjs.
Erstelle das Skript „automate.php“ welches die Aufgaben für Pflanzen basierend auf den aktuellen Messwerten erstellt.
Erstelle das Skript, „writeLog.php“ welches das Schreiben der Logs handhabt.
Teste das Skript log.php anhand der mitgelieferten Datei.