

Fabasoft Cloud Developer Information

Anlage ./1 zur Fabasoft Cloud Developer Vereinbarung

Gültig ab 13. Oktober 2014

Fabasoft[®]

Copyright ©

Fabasoft Cloud GmbH, A-4020 Linz, 2019.

Alle Rechte vorbehalten. Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Marken der jeweiligen Hersteller.

Diese Unterlagen sind streng vertraulich. Durch die Übermittlung und Präsentation dieser Unterlagen alleine werden keine Rechte an unserer Software, an unseren Dienstleistungen und Dienstleistungsergebnissen oder sonstigen geschützten Rechten begründet.

Die Weitergabe, Veröffentlichung oder Vervielfältigung ist nicht gestattet.

Aus Gründen der einfacheren Lesbarkeit wird auf die geschlechtsspezifische Differenzierung, z. B. Benutzer/-innen, verzichtet. Entsprechende Begriffe gelten im Sinne der Gleichbehandlung grundsätzlich für beide Geschlechter.

Inhalt

1 New Cloud Apps auf Basis der Fabasoft Cloud	5
1.1 Fabasoft Cloud Developer-Vereinbarung	5
1.2 Voraussetzungen für die Entwicklung von NCAs	5
1.3 Cloud Apps für die Entwicklung	5
2 Cloud App Development	7
2.1 Entwicklungsprojekt	7
2.1.1 Entwicklungsprojekt erzeugen	7
2.1.2 Kommerzielle Einstellungen festlegen	8
2.1.3 Entwicklungsprojekt initialisieren	8
2.2 Releases verwalten	8
2.3 Update-Zyklus	9
2.4 Use-Cases von Cloud App Development	11
3 Virtual Development Environment	12
3.1 Was ist das Virtual Development Environment (VDE)	12
3.2 Eine Virtuelle Entwicklungsumgebung erzeugen	13
3.3 VDE zurücksetzen	13
3.4 Passwort festlegen	13
3.5 Fabasoft app.telemetry	14
3.6 Use-Cases des Virtual Development Environment	14
4 Entwicklung mit Fabasoft app.ducx	15
4.1 Fabasoft Cloud Subversion Repository	16
4.2 Testen Ihrer Cloud App	16
4.3 Vorlagen erstellen	17
5 Der Release-Prozess	18
5.1 Voraussetzungen für das Übermitteln Ihrer Cloud App	18
5.2 Übermitteln eines Release (Submit)	18
5.2.1 Timeline für das Submit	19
5.3 Manuelles Review durch Fabasoft Reviewer	19
5.4 Hotfixes im Release-Prozess	19
5.5 Use-Cases im Release-Prozess	19
6 Cloud App „Scrum-Projekte“	20
6.1 Der Scrum-Prozess	21
6.2 Scrum-Projekt erzeugen	22
6.3 Stories bzw. Defects erzeugen	22
6.4 Backlog Management	22
6.4.1 Story bzw. Defect in den Backlog aufnehmen	22

6.4.2 Story bzw. Defect verwerfen	22
6.4.3 Story teilen	22
6.4.4 Stories bzw. Defects zusammenführen	22
6.4.5 Story in einen Defect umwandeln	22
6.5 Task erzeugen	23
6.6 Sprint planen	23
6.7 Use-Cases der Cloud App „Scrum-Projekte“	23

1 New Cloud Apps auf Basis der Fabasoft Cloud

Die Fabasoft Cloud bietet Software-Entwicklern eine Plattform für die Entwicklung und den Vertrieb von Applikationssoftware (New Cloud Apps, NCAs) für eine sichere und zuverlässige Zusammenarbeit sowohl unternehmensintern als auch unternehmensübergreifend.

Die „Fabasoft Cloud Developer Information“ beschreibt den Leistungsumfang all jener Komponenten, die Ihnen schnell und effizient die Umsetzung Ihrer eigenen NCAs für die Fabasoft Cloud ermöglichen.

Mit den Komponenten zur Entwicklung von NCAs, den Developer Cloud Apps (DCA),

- verwalten Sie Ihre Entwicklungsprojekte und Ihre Cloud Apps direkt in der Fabasoft Cloud (**Cloud App Development**, CAD),
- stellen wir Ihnen eine eigene virtuelle Entwicklungsumgebung (**Virtual Development Environment**, VDE) zur Verfügung,
- erstellen Sie Ihre Software mit **Fabasoft app.ducx** und speichern den Quellcode in einem zentralen Software-Repository in der Fabasoft Cloud,
- nutzen Sie **Fabasoft app.test primo** zum Erstellen und Ausführen von automatisierten Tests für Ihre Cloud App,
- verwenden Sie **Fabasoft app.telemetry primo** zur Laufzeitanalyse Ihres Source Codes,
- managen Sie Ihr Projekt mit Hilfe der Cloud App „**Scrum-Projekte**“ direkt in der Fabasoft Cloud,
- vermarkten Sie Ihre NCA im Fabasoft **Cloud App Store** (<http://www.cloudappstore.com>).

1.1 Fabasoft Cloud Developer-Vereinbarung

Die Basis dieser Cloud Developer Information bildet die **Fabasoft Cloud Developer-Vereinbarung**, die Sie unter folgendem Link finden: <http://www.fabasoft.com/developeragreement>.


1.2 Voraussetzungen für die Entwicklung von NCAs


Um Cloud Apps zu entwickeln, benötigen Sie insbesondere folgende Voraussetzungen:

- Leidenschaft und Passion zum Programmieren
- eine Idee, die Sie für Ihre Mitmenschen in der Cloud umsetzen möchten
- einen Fabasoft Cloud Account
- die Developer Cloud Apps

1.3 Cloud Apps für die Entwicklung

Bevor Sie mit der Entwicklung Ihrer eigenen Cloud App starten können, müssen Sie zwei Cloud Apps für Entwickler im Fabasoft Cloud App Store aktivieren.

**Cloud App Development**
Entwickeln Sie Ihre eigenen Cloud Apps - individuell und ganz nach Ihren Vorstellungen!



Die Cloud App „Cloud App Development“ erlaubt Ihnen die Verwaltung der Entwicklungsprojekte Ihrer Cloud Apps und stellt auch die Schnittstelle in die Continuous-Integration-Infrastruktur (CI) der Fabasoft Cloud zur Verfügung.

Den Link zu dieser Cloud App im Fabasoft Cloud App Store finden Sie unter:
<http://developer.fabasoft.com/editions/shop>.



Cloud App VDE

Machen Sie sich über Infrastruktur keine Sorgen. Entwickeln Sie Ihre Cloud Apps in einer virtuellen Entwicklungsumgebung, die von Fabasoft betrieben wird.



Die Cloud App „Virtual Development Environment“ stellt Ihnen eine virtuelle Entwicklungsumgebung zur Verfügung, die einer fast vollwertigen Fabasoft Cloud Umgebung entspricht – quasi eine Fabasoft Cloud für Sie persönlich. Sie verwenden die VDE, um Ihre Cloud App zu erstellen und während der Entwicklung unmittelbar auszuprobieren. Auch die Ausführung von automatisierten Tests gegen die VDE ist möglich.

Den Link zu dieser Cloud App im Fabasoft Cloud App Store finden Sie unter:

<http://developer.fabasoft.com/editions/shop>.

Sie programmieren Ihre Cloud App mit Fabasoft app.ducx. Dazu verwenden Sie die weitverbreitete Open-Source-Entwicklungsumgebung (IDE) Eclipse. Fabasoft app.ducx stellt dafür ein entsprechendes Plug-in bereit.

Den Source Code Ihrer Cloud App verwalten Sie im Fabasoft Cloud Subversion (SVN) Repository.

Bei der Initialisierung Ihres Entwicklungsprojekts wird automatisch ein Fabasoft app.ducx- und ein Fabasoft app.test-Projekt für Ihre Cloud App im Subversion Repository angelegt.

2 Cloud App Development

2.1 Entwicklungsprojekt

Das zentrale Objekt der App „Cloud App Development ist das „Entwicklungsprojekt“. Für jede Cloud App ist ein Entwicklungsprojekt zu erstellen. Mit dem Entwicklungsprojekt verwalten Sie die „Releases“ Ihrer Cloud App, die Präsentation Ihrer App im Cloud App Store sowie die kommerziellen Einstellungen. Sie verwenden das Entwicklungsprojekt zum Überprüfen des Freigabeprozesses (z.B.: „Übermittelt zur Prüfung“, „Manuelle Prüfung bestanden“, ...) und finden darin auch die Ergebnisse der automatisierten Tests der zur Freigabe übergebenen Version.


Für jedes Entwicklungsprojekt wird auch ein Verzeichnis im Repository der Fabasoft Cloud erzeugt, in dem Sie den Source Code Ihrer Cloud App verwalten.

2.1.1 Entwicklungsprojekt erzeugen

Erzeugen Sie einen Teamroom, in dem Sie alle relevanten Informationen über Ihre Projekte für das gesamte Entwicklungsteam zur Verfügung stellen und natürlich auch das Team bequem verwalten können.

Sie können in einem Teamroom mehrere Entwicklungsprojekte für unterschiedliche Cloud Apps erstellen.

Mit den Zugriffsrechten des Teamrooms legen Sie auch die Rechte für lesenden und schreibenden Zugriff auf das Source Code Repository fest.

Wichtig: Die Gruppe  [Fabasoft Cloud App Reviewer](#) benötigt „Alle Rechte“ in Ihrem Teamroom, sobald Sie Ihre Cloud App zum Review übergeben. Diese Gruppe wird automatisch beim Initialisieren des Entwicklungsprojektes zum Team Ihres Teamrooms hinzugefügt. Eine Cloud App kann nicht reviewed werden, wenn diese Gruppe nicht berechtigt wurde.

Jede Cloud App benötigt eine eindeutige Referenz. Die Eindeutigkeit der Referenz wird beim „Initialisieren“ des Entwicklungsprojekts geprüft. Die Referenz muss einen Bezug zum Entwickler(team) (Eigentümer-ID) und zur in der Cloud App umgesetzten Lösung (App-Referenz) haben. Eine gültige Referenz muss mit einem Großbuchstaben beginnen und darf nur Großbuchstaben und Ziffern beinhalten. Fabasoft-Apps beginnen mit COO oder FSC, diese Eigentümer-IDs dürfen Sie nicht verwenden.

Driver's Logbook (Entwicklungsprojekt): Lesen		Abbrechen	Bearbeiten	Weiter
Abrechnung	Allgemein	Anmerkungen	Sicherheit	Versionen
Entwicklungsprojekt	Friends	Vorlagen	Veröffentlichung	
Eigentümer-ID, muss mit einem Großbuchstaben beginnen, darf Buchstaben und Ziffern enthalten				
Eigentümer-ID	FSC			
App-Referenz	LOGBOOK			
Entwicklungsdomäne	111.100			
Lokation	https://folio.fabasoft.com/svn/apps/FCB798DAF91D3911AB30860F534FADBA/trunk/FSCLOGBOOK			
Status	Initialisiert			
Repository	Apps Repository			

2.1.2 Kommerzielle Einstellungen festlegen

Auf der Registerkarte „Abrechnung“ legen Sie die für die Abrechnung relevanten Daten fest. Einerseits bestimmen Sie den Preis Ihrer Cloud App bzw. innerhalb Ihrer App bereitgestellten Activity Points, in dem Sie eine vordefinierte Preisstufe zuordnen. Andererseits geben Sie Fabasoft Ihre Bankverbindung und Umsatzsteuer-ID bekannt, damit Fabasoft Ihren Entwickleranteil an den Erlösen aus Ihrer Cloud App auszahlen kann.


Die regelmäßig erstellten Abrechnungen werden Ihnen ebenfalls auf dieser Registerkarte präsentiert.

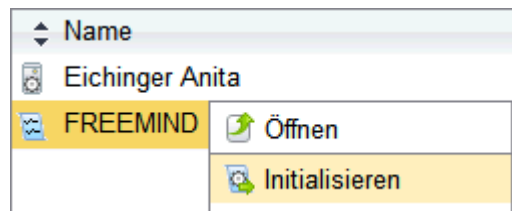
Entwicklungsprojekt		Vorlagen		Abrechnung	
Kontoinhaber		Anita Eichinger			
IBAN					
BIC					
Umsatzsteuerpflichtig *		<input type="checkbox"/>			

Einen Überblick über die zur Verfügung stehenden Preisstufen finden Sie unter <http://www.fabasoft.com/pricetiers>.

2.1.3 Entwicklungsprojekt initialisieren

Mit der Initialisierung Ihres Entwicklungsprojekts starten Sie Ihren Entwicklungsprozess. Dabei wird

- die Verfügbarkeit, Gültigkeit bzw. Eindeutigkeit der Referenz geprüft.
- ein Fabasoft app.ducx und Fabasoft app.test Projekt im Fabasoft Cloud Source Code Repository angelegt.
- der Gruppe  [Fabasoft Cloud App Reviewer](#) Zugriff auf Ihren Teamroom gewährt, in dem Sie zum Team des Teamrooms hinzugefügt wird.



Nur ein Benutzer mit allen Rechten am Teamroom kann das Projekt initialisieren.

Den Repository-Pfad zu Ihrem Projekt finden Sie in den Eigenschaften des Entwicklungsprojektes.

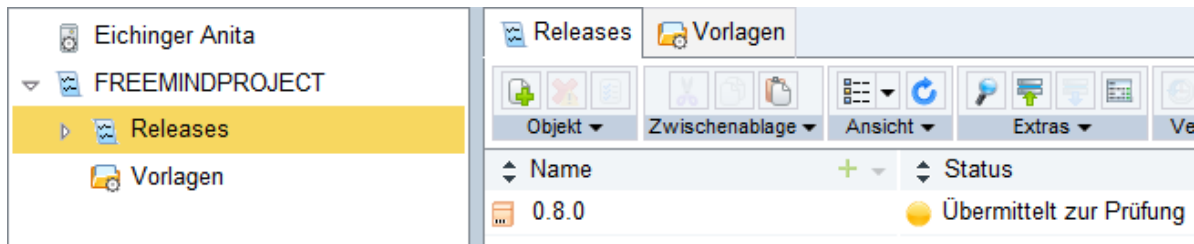
Die Zugriffsrechte auf den Source Code im Fabasoft Cloud Repository bestimmen Sie durch die Berechtigungen des Teamrooms, dem das Entwicklungsprojekt zugeordnet ist.

Des Weiteren wird ein Adressbereich für Ihre Cloud App reserviert, der für die Entwicklung mit Fabasoft app.ducx erforderlich ist.

2.2 Releases verwalten

Wenn Sie Ihre Cloud App fertiggestellt haben und diese in der Fabasoft Cloud installieren möchten, erstellen Sie in Ihrem Entwicklungsprojekt ein Release Ihrer Cloud App.

Dieses Release übermitteln Sie an die  [Fabasoft Cloud App Reviewer](#) zur Prüfung.



Durch die Übermittlung der App zur Prüfung wird für den aktuellen Stand (Stand im „trunk“-Zweig) ein neuer Zweig im Source Code Repository unter „branches“ erstellt. Für die Installation wird nur der Softwarestand für das Release (unter „branches“) herangezogen. Allfällige Änderungen durch den Entwickler im Zuge des Review-Prozesses müssen somit in diesem neuen Zweig vorgenommen werden.

Bevor eine Version zum Review übermittelt wird, müssen Sie sicherstellen, dass Ihre Tests ohne Fehler durchlaufen. Während des Review-Prozesses können Sie keine funktionalen Änderungen an Ihrer Cloud App vornehmen.

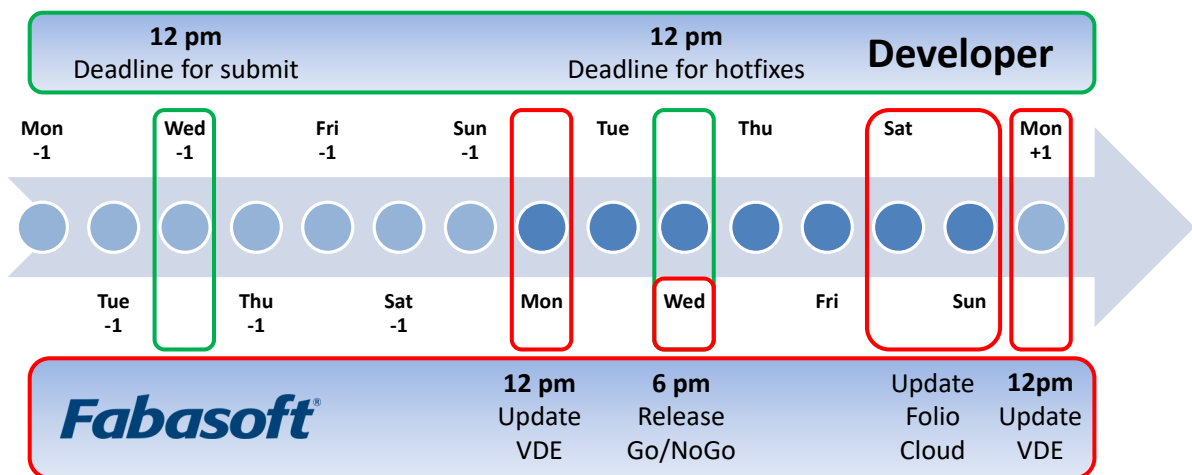
Das zur Prüfung übermittelte Release wird automatisiert im CI mit Hilfe Ihrer Tests (Unit-Test erstellt mit Fabasoft app.ducx und User Acceptance Tests erstellt mit Fabasoft app.test) getestet. Im Zuge dessen wird auch die Coverage Ihrer Tests gemessen. Der Review-Prozess wird nur fortgesetzt, wenn Ihre Tests fehlerfrei durchlaufen und damit auch eine entsprechende Code Coverage erzielt wird (mind. 80 %). Die Testergebnisse und das Ergebnis der Coverage-Messung werden für Sie im Release-Objekt abgelegt.

Fabasoft führt auch eine manuelle Prüfung Ihrer Cloud App durch. Nach der manuellen Prüfung wird Ihre Cloud App auch in einem Integrations-Test im CI-System automatisiert getestet. Dieses Integrations-System entspricht nahezu dem Produktionssystem der Fabasoft Cloud. So sind in diesem System alle bisher freigegebenen Cloud Apps ebenfalls installiert.

Hinweis: Planen Sie Ihre Fabasoft Cloud App Releases gemäß dem Fabasoft Update-Zyklus, um mit jedem neuen Fabasoft Cloud Update auch Ihre eigene Fabasoft Cloud App auf den neuesten Stand zu bringen. Die Übermittlung einer neuen Version zur Prüfung ist nur bis zehn Tage vor dem Update von Fabasoft Cloud möglich.

2.3 Update-Zyklus

Der Fabasoft Update-Zyklus definiert, bis wann ein App-Entwickler neue Features bzw. Hotfixes für ein neues Release einbringen kann und wann Fabasoft Updates der DCAs bereitstellt. Bezugspunkt ist das monatliche Update der Fabasoft Cloud.



All times in CET

Aktion	2 Wochen vor Update							Woche vor Update						
	Mo -1	Di -1	Mi -1	Do -1	Fr -1	Sa -1	So -1	Mo	Di	Mi	Do	Fr	Sa	So
Deadline für funktionale Erweiterungen in der Cloud App (bis 12:00 Uhr MEZ)			X											
Bereitstellung Template-VDE für das manuelle Zurücksetzen von VDEs				X										
Bereitstellung eines Vorabstandes der DCAs (bis 12:00 Uhr MEZ)														
- Fabasoft app.ducx Eclipse-Plugin via update.ducx.com														
- Update Public Subversion								X						
- Automatisches Update der VDEs														
Deadline für Hotfixes (bis 12:00 Uhr MEZ)											X			
Update von Fabasoft Folio Cloud													X	X
Bereitstellung der aktualisierten DCAs (bis 12:00 Uhr MEZ)														
- Fabasoft app.ducx Eclipse-Plugin via update.ducx.com														
- Update Public Subversion														X
- Automatisches Update der VDEs														

Das Update findet in der Regel am ersten Wochenende eines Monats statt. Die genauen Update-Termine werden im Scrum Projekt [Fabasoft Cloud](#) bzw. in Kalender [Fabasoft Cloud](#) publiziert.

Die Deadline für Ihre funktionalen Erweiterungen endet am Mittwoch um 12:00 Uhr Mittag in der zweiten Woche vor dem Update (ca. zehn Tage vor dem Update der Fabasoft Cloud). Nach diesem Termin können Sie keine neue Release zum Review bzw. keine funktionalen Änderungen an einer bereits zum Review übermittelten Release einbringen.

Am Montag in der Woche des Updates werden automatisch alle VDEs auf die vorläufige Version der Fabasoft Cloud aktualisiert, unabhängig davon, ob ein manuelles Rücksetzen erfolgt ist.

Damit bleiben noch zwei Tage Zeit, bis Mittwoch 12:00 Uhr Mittag, um etwaige Hotfixes für Ihre Fabasoft Cloud App bereitzustellen.

Am Montag nach dem Update werden bis 12:00 Uhr Mittag alle VDEs automatisch auf die neue Version der Fabasoft Cloud aktualisiert.

Sollten Sie schon früher Ihre Entwicklung und sämtliche Tests abgeschlossen haben, zögern Sie bitte nicht und übermitteln Sie Ihr Release umgehend zum Review.

Die genannten Zeitangaben beziehen sich auf Mitteleuropäische Zeit (UTC+1) resp. Mitteleuropäische Sommerzeit (UTC+2).

Beim Rücksetzen der VDE werden alle Konfigurationen und Daten in der VDE ebenfalls zurückgesetzt. Es empfiehlt sich daher, solche Konfigurationen und das Erstellen von Testdaten automatisiert mit Fabasoft app.test zu erstellen, so dass die VDE einfach und rasch an die eigenen individuellen Bedürfnisse angepasst wird.

2.4 Use-Cases von Cloud App Development

- Entwicklungsprojekt erzeugen und verwalten
- Kommerzielle Einstellungen festlegen
- Entwicklungsprojekt initialisieren
- Release starten
- Release zum Review übermitteln

3 Virtual Development Environment

3.1 Was ist das Virtual Development Environment (VDE)

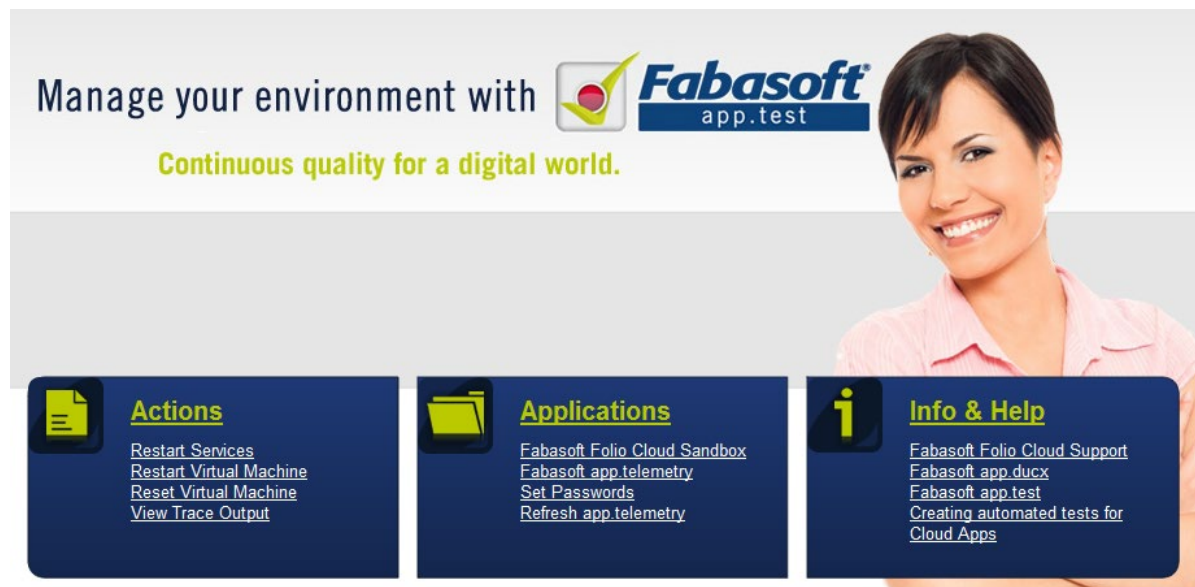
Die Fabasoft Cloud App „Virtual Development Environment“ (kurz VDE) stellt für Sie eine virtuelle Laufzeitumgebung für die Entwicklung Ihrer Cloud App zur Verfügung – im Grunde eine Fabasoft Cloud für Sie und Ihr Entwicklerteam.

Die URL zu Ihrer ganz persönlichen Fabasoft Cloud finden Sie im VDE-Objekt, welches Sie einmalig erzeugen können. In der Regel ist die URL folgendermaßen aufgebaut:

<https://vde.fabasoft.com/dev<ID>/vm<ID2>>

Über diese URL gelangen Sie auch zum Management-Portal der VDE, über das Sie folgende Use-Cases ausführen können:

- Neustarten der Fabasoft Cloud Services
- Neustarten der gesamten VDE
- Anzeigen der Trace-Ausgaben der Fabasoft Cloud Web-Services
- Zurücksetzen der VDE auf die Vorlage
- Starten des Fabasoft Cloud Clients in der Sandbox
- Starten des Fabasoft app.telemetry Clients
- Ändern des Passworts der Testbenutzer
- Anzeigen von Hilfe und zusätzlichen Information der CDAs



Über die URL <https://vde.fabasoft.com/dev<ID>/vm<ID2>/folio> erreichen Sie die Fabasoft Cloud Web-Services der VDE und können so mit unterschiedlichen Testbenutzern Ihre App testen.

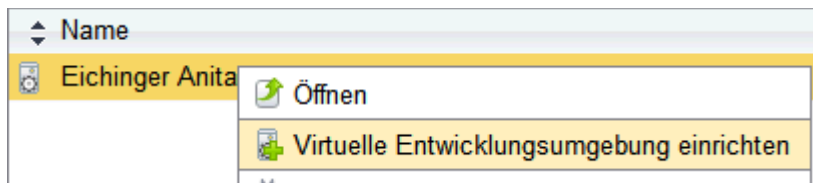
Bitte beachten Sie bei der Verwendung Ihrer VDE folgende Punkte:

- Die VDEs werden nicht gesichert und können jederzeit (auch von Fabasoft) zurückgesetzt werden. Alle temporären Objekte und Einstellungen Ihrer Installation gehen beim Zurücksetzen der VDE verloren.
- Ein Zurücksetzen der VDE auf den Auslieferungszustand Ihrerseits ist jederzeit möglich.

- Für Ihre VDE werden von Fabasoft 15 GB Speichervolumen zur Verfügung gestellt. In diesen 15 GB ist das Betriebssystem, die Datenbank und die Fabasoft Cloud Installation selbst inkludiert. Damit stehen noch ca. 2 GB an Platz für Daten zur Verfügung.
- Für Ihre VDE bestehen seitens Fabasoft keinerlei Service Level Agreements.
- Fabasoft wird die VDE gemäß Update-Zyklus eine Woche vor jeder Aktualisierung von Fabasoft Cloud und unmittelbar nach der Aktualisierung zurücksetzen. Damit können Sie Ihre Apps gegebenenfalls frühzeitig an die neueste Version von Fabasoft Cloud anpassen.
- Sie können durch die Cloud App VDE genau eine VDE erstellen. Sie müssen diese VDE für die Entwicklung all Ihrer Apps verwenden.
- Es stehen Ihnen mehrere Test-Benutzer in bis zu 10 Scopes zur Verfügung.
- Der Benutzername „developer“ berechtigt Sie als System Administrator mit vollem Zugriff. Damit haben Sie alle Berechtigungen in Ihrer Fabasoft Cloud Installation.
- Es steht kein direkter Zugriff auf das Betriebssystem oder der Datenbank der VDE zur Verfügung.

3.2 Eine Virtuelle Entwicklungsumgebung erzeugen

Erzeugen Sie in einem Teamroom eine „Virtuelle Entwicklungsumgebung“ und richten Sie diese ein. Beim Erzeugen wird eine bestehende VDE-Vorlage geklont, an Ihre Umgebung angepasst und für Sie verfügbar gemacht.



3.3 VDE zurücksetzen

Fabasoft stellt bei jedem Update aktualisierte VDEs zur Verfügung. Im Zuge der Aktualisierung von Fabasoft Cloud werden auch alle VDEs auf den neuesten Softwarestand zurückgesetzt.

Sie haben die Möglichkeit, Ihre VDE jederzeit eigenständig auf den aktuell in Fabasoft Cloud verwendeten Softwarestand zurückzusetzen.

Beachten Sie, dass ca. eine Woche vor der Aktualisierung von Fabasoft Cloud die VDEs nur mehr auf den Stand aktualisiert werden können, mit dem Fabasoft Cloud zukünftig betrieben wird. Damit können Sie Ihre Cloud App schon für diesen Softwarestand testen und gegebenenfalls noch Hotfixes bereitstellen.

Die VDE kann entweder über das Kontextmenü des VDE-Objekts oder über das VDE-Management-Portal zurückgesetzt werden.

Bitte beachten Sie, dass nach dem Zurücksetzen Ihrer VDE Ihre Cloud Apps nicht mehr in Ihrer Fabasoft Cloud Domäne enthalten sind. Um diese wieder verfügbar zu machen, müssen Sie Ihre Entwicklungsprojekte erneut laden. Mehr dazu erfahren Sie im Abschnitt „Entwicklung mit Fabasoft app.ducx“.

3.4 Passwort festlegen

Beim Zurücksetzen Ihrer VDE wird das Passwort auf einen zufälligen Wert geändert, damit niemand unberechtigt auf Ihre VDE zugreifen kann. Sie müssen deshalb das Passwort nach jedem Zurücksetzen ändern.

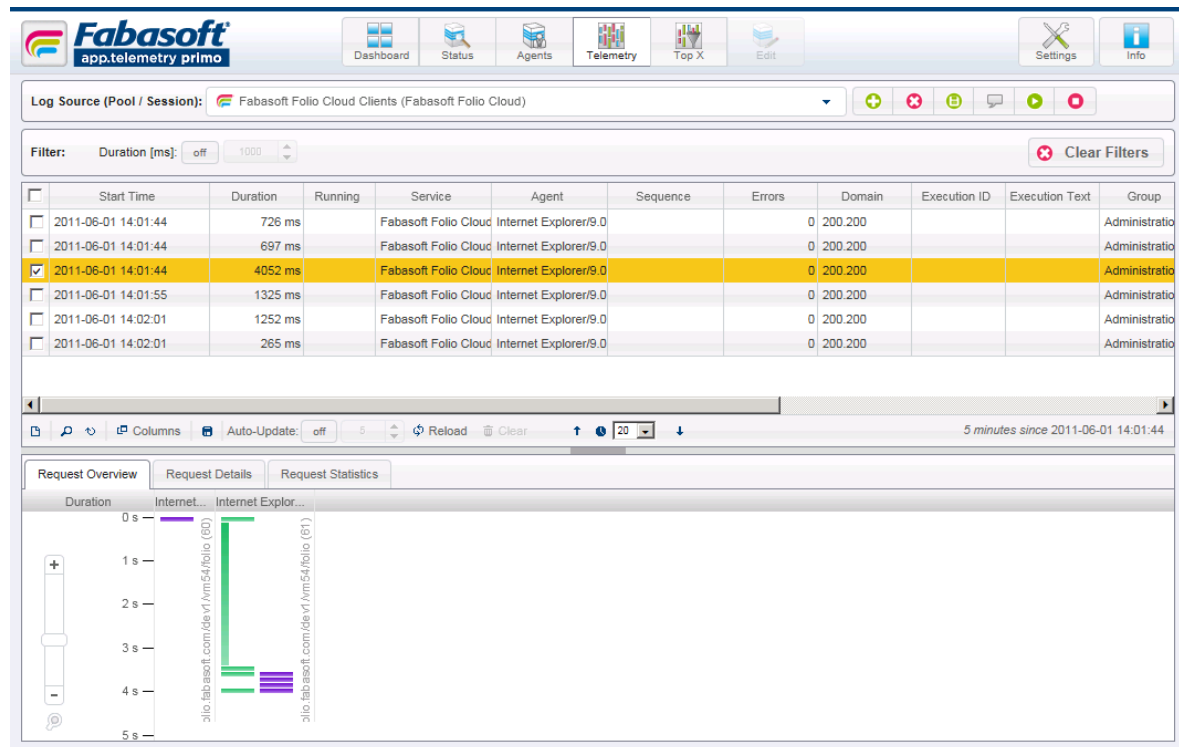
Sie können das Passwort über das Kontextmenü des VDE-Objekts neu festlegen.

3.5 Fabasoft app.telemetry

In der VDE ist auch Fabasoft app.telemetry primo installiert. Mit Hilfe der Software-Telemetrie von Fabasoft app.telemetry primo können Sie Ihre Fabasoft Cloud App gezielt auf Performance hin trimmen.

Beachten Sie jedoch, dass Fabasoft app.telemetry primo nur die Analyse von Requests der letzten 5 Minuten erlaubt.


Mehr Informationen zu app.telemetry finden Sie unter <http://www.apptelemetry.com>.



3.6 Use-Cases des Virtual Development Environment

- Virtuelle Entwicklungsumgebung einrichten
- Virtuelle Entwicklungsumgebung zurücksetzen
- Virtuelle Entwicklungsumgebung zurückgeben
- Passwort der Testbenutzer neu festlegen
- Neustarten der Fabasoft Cloud Services
- Neustarten der gesamten VDE
- Anzeigen der Trace-Ausgaben der Fabasoft Cloud Web-Services
- Starten des Fabasoft Cloud Clients in der Sandbox
- Starten des Fabasoft app.telemetry Clients
- Anzeigen von Hilfe und zusätzlichen Information der CDAs
- Verwenden von app.telemetry Software-Telemetry zur Analyse der Request der letzten 5 Minuten

4 Entwicklung mit Fabasoft app.ducx

Fabasoft app.ducx ist die Entwicklungsplattform für Fabasoft Cloud. Detaillierte Informationen zur Entwicklung von Cloud Apps mit Fabasoft app.ducx finden Sie im Buch  [Developing Fabasoft Cloud Apps](#).

Code-Beispiele können Sie sich auch im Fabasoft Cloud Subversion Repository unter <https://cloud.fabasoft.com/svn/public/Folio%20Cloud%20Apps/samples> ansehen.

Sie verwenden die Entwicklungsumgebung (IDE) Eclipse, um mit Fabasoft app.ducx zu programmieren. Fabasoft app.ducx stellt für Eclipse ein Plug-in zur Verfügung. Die aktuelle, von Fabasoft app.ducx unterstützte, Version von Eclipse inkl. aller erforderlichen Plug-ins finden Sie im Ordner:

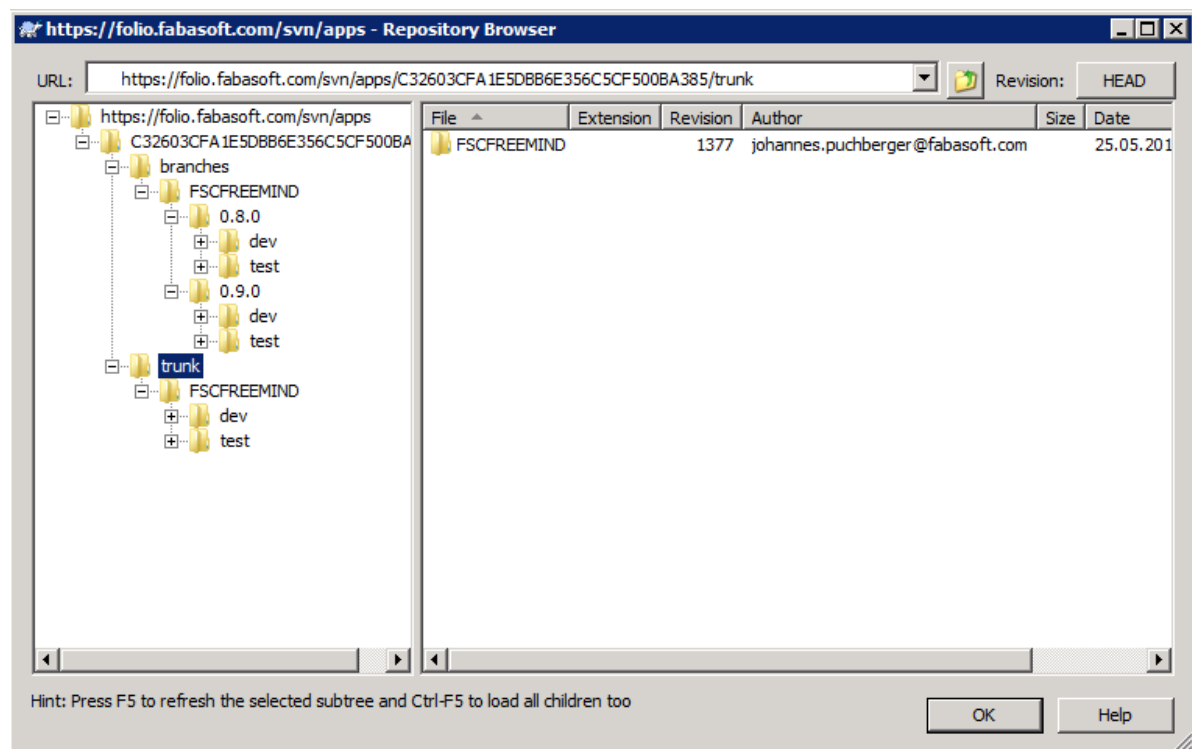
 [Fabasoft app.ducx for Cloud Development](#).

Den Source Code Ihrer Cloud App verwalten Sie im Fabasoft Cloud Subversion (SVN) Repository. Es empfiehlt sich, einen Client wie TortoiseSVN oder das Subversive SVN Eclipse-Plug-in zur Kommunikation mit dem Repository (Checkout und Commit) zu verwenden.

Der Zugriff auf das Fabasoft Cloud Subversion Repository erfolgt mit Ihrem Benutzernamen (E-Mail-Adresse) und Passwort für die Fabasoft Cloud. Für den Zugriff auf das Repository muss der Benutzername (E-Mail-Adresse) in Kleinbuchstaben eingegeben werden.

Bei der Initialisierung Ihres Entwicklungsprojekts wird ein Fabasoft app.ducx und ein Fabasoft app.test Projekt im Subversion Repository angelegt. Den Pfad zu Ihrem Projekt finden Sie in den Eigenschaften des Entwicklungsprojekts.

Der trunk-Zweig wird für die laufende Entwicklung Ihrer Cloud App verwendet. Dieser Zweig wird unter Umständen auch einen instabilen Stand Ihrer Cloud App enthalten. Der Softwarestand im trunk-Zweig wird jedoch im Review-Prozess nicht herangezogen. Möchten Sie eine Version distribuieren, muss der entsprechende Softwarestand im branches-Zweig abgelegt werden. Übermitteln Sie eine Release das erste Mal zur Prüfung, wird automatisch vom trunk-Zweig eine Kopie im branches-Zweig erzeugt.



Während der Entwicklung Ihrer Cloud App mit Fabasoft app.ducx entstehen Objekte, die eine weltweit eindeutige Identifikation benötigen. Im Fabasoft Cloud Umfeld wird dazu die sog. Objektadresse verwendet. Um die Eindeutigkeit zu gewährleisten, wird Ihrem Fabasoft app.ducx Projekt ein Adressbereich zugeordnet.

Bei der Initialisierung des Projekts wird Ihnen ein Adressbereich von 500 Adressen je Entwickler bereitgestellt. Fabasoft app.ducx kann so konfiguriert werden, dass automatisch neue Adressbereiche angefordert werden, sobald diese notwendig sind.

Die Verwaltung dieser Adressbereiche erfolgt ebenfalls in Fabasoft Cloud. Sie müssen deshalb in Eclipse unter *Window > Preferences > Fabasoft app.ducx* Fabasoft Cloud (<https://at.cloud.fabasoft.com/folio>) als *Default range service* konfigurieren. Verwenden Sie dazu wiederum Ihren Benutzernamen und Passwort für die Anmeldeinformationen.

Diese Objektadressen dienen in der Fabasoft Cloud auch als Schlüssel für die Datenhaltung von Eigenschaftswerten. Die Zuordnung einer Objektadresse zu einer Klasse, Eigenschaft oder anderen Elementen darf sich demnach nicht mehr ändern, sobald die Cloud App das erste Mal installiert wurde. Verwenden Sie beim Ändern von Bezeichnungen deshalb unbedingt die Funktion *Refactor > Fabasoft app.ducx > Change Reference ...*

4.1 Fabasoft Cloud Subversion Repository

Der von Ihnen verbrauchte Speicherplatz im Repository wird Ihrem Speicherplatz gemäß Ihrer Fabasoft Cloud Lizenz zugerechnet und ist dadurch limitiert. Es werden mehrmals täglich Backups durchgeführt. Das Repository läuft auf einem Server und ist somit nicht Ausfallsicherheit konfiguriert.

Die Zugriffsrechte auf Ihre Verzeichnisse im Fabasoft Cloud Subversion Repository werden durch den Teamroom bestimmt, dem das Entwicklungsprojekt Ihrer Cloud App zugeordnet ist.

4.2 Testen Ihrer Cloud App

Nur mit qualitativ hochwertiger Software werden Sie Ihre Nutzer zufrieden stellen. Kontinuierliche Qualität lässt sich jedoch nur durch weitgehende Automatisierung von Tests erreichen. Neben der Zeitersparnis durch den Wegfall wiederkehrender manueller Tests kann so auch ein monatlicher Update-Rhythmus eingehalten werden.

Die Fabasoft Cloud bietet deshalb Entwicklern die Möglichkeit, Ihre Cloud App einerseits mit Hilfe von Unit-Tests, die im Rahmen des Fabasoft app.ducx Projekts erstellt werden und andererseits mit User Acceptance Tests auf Basis Fabasoft app.test primo automatisiert zu testen.

Haben Sie Ihre Cloud App zum Review übermittelt, werden Ihre automatischen Tests auch in unterschiedlichen Testsystemen bei Fabasoft ausgeführt. Unter anderem erfolgen die Tests auch in einem Integrations-System, das dem der Fabasoft Cloud nahezu vollständig entspricht. Um die automatisierten Tests in den unterschiedlichen CI-Systemen ausführen zu können, stellt Ihnen Fabasoft eine Bibliothek von Test-Use-Cases für die Erstellung Ihrer User Acceptance Tests mit Fabasoft app.test zur Verfügung. Es ist zwingend erforderlich, dass diese Bibliothek in Ihren Tests eingesetzt wird.

Diese Bibliothek finden Sie unter:

<https://cloud.fabasoft.com/svn/public/Folio%20Cloud%20Apps/apptestlib>.

Eine Beschreibung über die Verwendung der app.test-Bibliothek finden Sie im Dokument



[Creating Automated Tests for Cloud Apps With Fabasoft app.test](#).

Die Code Coverage, die mit diesen automatisierten Tests erreicht wird, muss auf ein Ziel von 100% hinweisen.

4.3 Vorlagen erstellen

Damit Vorlagen erstellt werden können, muss im Entwicklungsprojekt der Typ des Teamrooms angegeben sein.

Um Vorlagen zu erstellen, gehen Sie folgendermaßen vor:

1. Führen Sie „Vorlagen aktivieren“ auf das Entwicklungsprojekt aus.
2. Die „App-Vorlagensammlung“ wird im Teamroom abgelegt.
3. Erstellen Sie in der „App-Vorlagensammlung“ eine Vorlage für einen Teamroom.
4. Wählen Sie aus, wo die Vorlage verfügbar sein soll.
5. Führen Sie „Zur Freigabe weiterleiten“ aus dem Extras-Menü auf die gewünschte Vorlage aus, um sie zur Überprüfung weiterzuleiten.

Wird die Vorlage freigegeben, befindet Sie sich auf der Registerkarte „Freigegebene Vorlagen“ und steht somit allen Benutzern zur Verfügung.

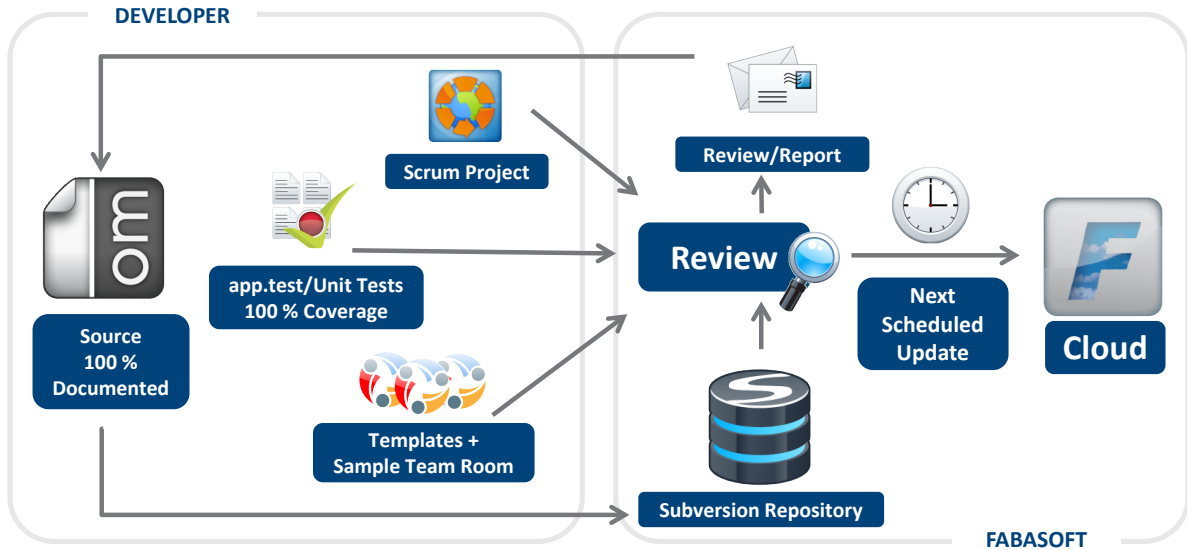
Wird die Vorlage nicht freigegeben, ist sie nur unter der Registerkarte „Vorlagen“ ersichtlich

Wichtig: Die Gruppe  [Fabasoft Cloud App Reviewer](#) muss im Teamroom mit „Alle Rechte“ eingetragen sein, um die Vorlagen freigeben zu können

Innerhalb des Teamrooms, indem Sie Ihr Entwicklungsprojekt erstellt haben, können Sie die Vorlagen testen.

Die Verfügbarkeit der Vorlage gibt an, ob ein Nutzer beim Erzeugen eines Objekts die Vorlage auswählen kann, oder die Vorlage nach dem Aktivieren der App automatisch in „Home“ abgelegt wird.

5 Der Release-Prozess



5.1 Voraussetzungen für das Übermitteln Ihrer Cloud App

Bevor Sie Ihre Cloud App zum Review übermitteln, müssen Sie folgende Punkte überprüfen:

- Ihr Fabasoft app.ducx Projekt kann gebaut und geladen werden.
- Der Source-Code Ihres Projekts ist vollständig und ausführlich dokumentiert.
- Sie haben alle Zeichenketten in allen von Ihrer App unterstützten Sprache bereitgestellt.
- Sie haben durch manuelle Tests die gewünschte Funktionalität überprüft.
- Durch Fabasoft app.tests und/oder Unit-Tests haben Sie automatisiert ausführbare Tests bereitgestellt.
- Die Fabasoft app.tests verwenden die app.test Bibliothek für die Login-Use-Cases.
- Die Überprüfung der Coverage mittels automatisierter Tests weist auf ein Ziel von 100% hin.

5.2 Übermitteln eines Release (Submit)

Nachdem Sie Ihr Release zum Review übermittelt haben, wird Ihr Projekt durch unser CI-System automatisiert ausgecheckt, gebaut, in einem Testsystem eingespielt und die automatisierten Tests ausgeführt.

Wenn alle Punkte erfolgreich ausgeführt wurden, wird das Release zum manuellen Review freigegeben und die [Fabasoft Cloud App Reviewer](#) darüber informiert.


Tritt hingegen ein Fehler auf, so müssen Sie diesen beheben und erneut auf das Ergebnis des automatisierten CI-Systems warten.

Achtung: Durch das erste Übermitteln zum Review eines Release wird ein Branch des aktuellen Trunks erzeugt. Somit müssen Sie beim Ausbessern von Fehlern die entsprechenden Revisions auch in den jeweiligen Branch übernehmen.

5.2.1 Timeline für das Submit

Die genauen Zeiten, bis wann Sie Ihr Release und etwaige Hotfixes spätestens übermitteln müssen, entnehmen Sie dem [Update-Zyklus](#).

5.3 Manuelles Review durch Fabasoft Reviewer

Ein professioneller  [Fabasoft Cloud App Reviewer](#) wird Ihren Source Code auf Herz und Nieren überprüfen und Ihnen entsprechend Feedback zu Verbesserungen und möglichen Schwachstellen liefern, sodass der hohe Qualitätsstandard von Fabasoft Cloud gewahrt werden kann. Das Review ist aber nicht als grundlegende und umfassende Standard-Qualitätssicherung durch Fabasoft zu verstehen, sondern als Kontrolle Ihrer Qualitätssicherungsmaßnahmen anhand von spezifischen Kontrollpunkten.

Sollte der Reviewer Kommentare zu Ihrem Release haben, werden diese in der Eigenschaft „Bisherige Anmerkungen“ des Release-Objekts hinterlegt.

5.4 Hotfixes im Release-Prozess

Sie können innerhalb des Updatezyklus nur zu bereits zum Review übermittelten Releases etwaige Hotfixes bis maximal 4 Tage vor dem Update (Mittwoch) bereitstellen. Wenn Sie diese Frist versäumen, müssen Sie bis zum nächsten Cloud Update warten und erhalten somit die Chance auch gleich weitere Features zu implementieren.

5.5 Use-Cases im Release-Prozess

- Eine Release starten
- Eine Release übermitteln
- Ergebnisse aus den automatisierten Test prüfen

6 Cloud App „Scrum-Projekte“

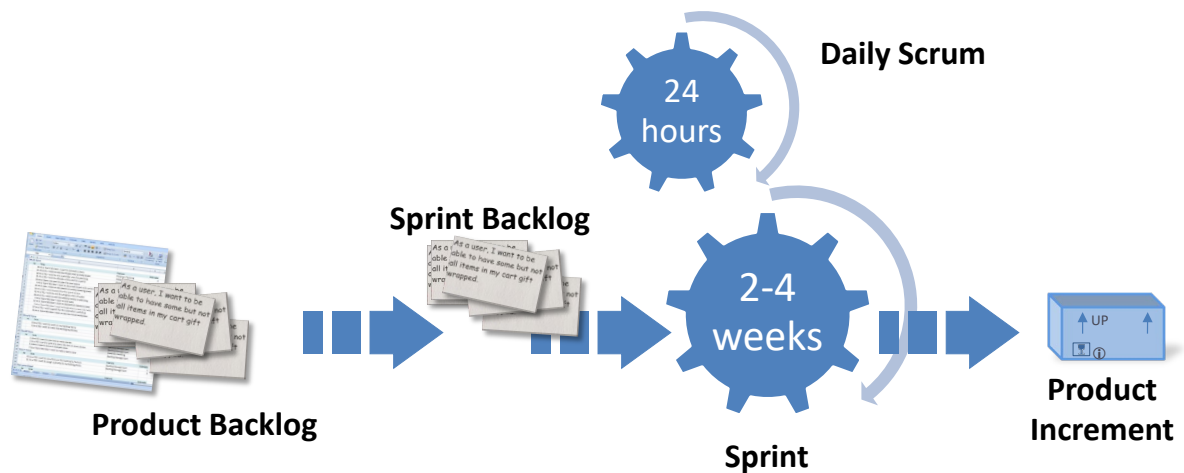
Mit der App für Scrum-Projekte können Teams unterschiedlicher Größe Scrum-Projekte einfach erstellen, bewerten, planen und nachverfolgen - rund um den Globus.

Scrum ist ein iteratives, inkrementelles Framework für das Management von agilen Projekten. Scrum-Teams produzieren in iterativen Sprints verwertbare Ergebnisse (z.B. Produkt-Inkrementes). Nach jedem Sprint werden die Anforderungen (User Stories) im Backlog neu bewertet und priorisiert, um rasch auf Änderungen reagieren zu können. Das Scrum Framework fordert die Scrum-Teams permanent, sich auf Verbesserungen zu konzentrieren und jene Aufgaben zuerst zu erledigen, die zum aktuellen Zeitpunkt (Sprint) den meisten Business Value haben. Diese Eigenschaften haben dazu geführt, dass Scrum eine der beliebtesten Methoden weltweit für das agile Projektmanagement wurde.

Die Cloud App „Scrum-Projekte“ steht jedem Fabasoft Cloud Entwickler zur Verfügung – Sie müssen sie nur aktivieren.

Name	Größe	Priorität	Status
Automatische Nummerierung der SCI	3 Story Points	A - Must have	Planning
Projektphasen	3 Story Points	A - Must have	To Verify
GUI Definition	2 Story Points	A - Must have	In Progress
Datenmodell	3 Story Points	A - Must have	In Progress
Neues Costumer Projekt	5 Story Points	A - Must have	Planning
Elemente Übersicht	2 Story Points	B - Should have	In Progress
XML File erstellen			Planning
Automatische Nummerierung der SCI	3 Story Points	C - Could have	In Progress
Aktivieren/Deaktivieren von Elementen			Planning

6.1 Der Scrum-Prozess



Project Inbox

In der Project Inbox werden User Stories und Defects von Benutzern oder Stakeholdern erstellt, die in weiterer Folge von Product Owner priorisiert und gegebenenfalls in den Backlog aufgenommen oder verworfen werden.

Product Backlog

In einem Product Backlog werden Anforderungen als sogenannte User Stories durch den Product Owner formuliert und priorisiert. Die Stories übernimmt er dazu auch aus der Project Inbox.

Sprints

Ein Sprint ist eine festgelegte Zeitspanne - beispielsweise zwei oder vier Wochen - in denen ein Scrum-Team an einer selbst bestimmten Menge an User Stories aus dem Backlog arbeitet. Das Ergebnis ist ein qualitativ hochwertiges Produktinkrement. Die Abarbeitung der User Stories wird über ein Task Board verfolgt. Die Größe der Stories wird in Story Points vom Team geschätzt.

Task

Für die Umsetzung einer User Story erstellt das Team Tasks, die der Story zugeordnet sind. Jedes Teammitglied übernimmt bestimmte Tasks für die Umsetzung einer User Story. Über den Status des Tasks ist erkennbar, wie weit dessen Erledigung bereits gediehen ist.

Velocity Chart

Das Velocity Chart zeigt in einer grafischen Darstellung, wie gut die Sprints des Projekts über die Zeit abgearbeitet werden. Maßzahl ist die Velocity des Teams, das ist die Summe der pro Sprints umgesetzten Story Points.

Burndown Chart

Das Burndown Chart zeigt in einer grafischen Darstellung die Abarbeitung der User Stories in einer übersichtlichen grafischen Darstellung. Damit ist einfach erkennbar, wie die Abarbeitung des aktuellen Sprints verläuft.

6.2 Scrum-Projekt erzeugen

Scrum Projekte ermöglichen Ihnen die Planung und eine agile Entwicklung Ihrer Fabasoft Cloud App.

6.3 Stories bzw. Defects erzeugen

Definieren Sie die gewünschte Funktionalität und schreiben Sie detaillierte Stories dazu. Mit den Abnahmekriterien können Sie sich bestimmte Ziele stecken, die Sie im Zuge dieser Story erreichen möchten. Sobald Sie die Story ausreichend definiert haben, können Sie diese im nächsten Schritt in den Backlog übernehmen.

Nutzer Ihrer Cloud App können ebenfalls Stories erfassen, wenn sie sich mehr Funktionen wünschen. Fehler in Ihrer Cloud App können von den Nutzern als Defects erfassen in Ihrem Scrum-Projekt erfasst werden.

Damit Nutzer Stories und Defects erstellen können, müssen Sie Ihr Scrumprojekt in einem öffentlichen Teamroom erzeugen.

6.4 Backlog Management

In der Project Inbox eines Scrum-Projekts werden alle Stories und Defects gesammelt, die dieser Project Inbox zugeordnet wurden. Sie können diese Stories bzw. Defects in den Backlog aufnehmen, verwerfen, teilen, zusammenführen oder umwandeln.

6.4.1 Story bzw. Defect in den Backlog aufnehmen

Stories und Defects die Sie umsetzen möchten, werden in den Backlog aufgenommen und gemäß Nutzen und Aufwand priorisiert und entsprechend gereiht. Der Aufwand von Stories wird in Story Points geschätzt. Beachten Sie jedoch, dass Defects stets die höchste Priorität haben sollten.

6.4.2 Story bzw. Defect verwerfen

Sie können Stories bzw. Defects, die noch nicht in den Sprint aufgenommen wurden, verwerfen. Verworfen Stories bzw. Defects werden in die Registerkarte „Backlog Done“ verschoben.

Hinweis: Nur der Product Owner besitzt die nötigen Rechte, um eine Story zu verwerfen. Ein Defect kann von jedem Team-Mitglied verworfen werden.

6.4.3 Story teilen

Wenn Sie während der Planung einer Story merken, dass die Anforderung zu umfangreich ist, um in einem Sprint umgesetzt zu werden, können Sie diese jederzeit teilen und sich somit besser auf die einzelnen Teile konzentrieren.

6.4.4 Stories bzw. Defects zusammenführen

Falls es für eine Anforderung oder ein Problem mehrere Stories bzw. Defects gibt, können Sie diese zusammenführen, solange sie noch nicht in einen Sprint aufgenommen wurden.

6.4.5 Story in einen Defect umwandeln

Wenn es sich bei einer erfassten Story um keine neue Funktionalität handelt, sondern um einen Fehler bzw. Mangel in einer existierenden Funktionalität, können Sie die Story in einen Defect umwandeln. Natürlich können Sie auch einen Defect in eine Story umwandeln, wenn sich

herausstellt, dass es sich bei dem beschriebenen Sachverhalt nicht um einen Fehler sondern ein neues Feature handelt.

Hinweis: Eine Story kann nur in einen Defect umgewandelt werden, wenn sie noch nicht in den Backlog aufgenommen wurde.

6.5 Task erzeugen

„Task hinzufügen“ kann verwendet werden, um eine Story in einzelne Aufgaben zu zergliedern. Teilen Sie die Aufgaben in z.B.:

- Implementierung Objektmodell
- Implementierung Use-Case 1
- app.test
- Unittests
- Dokumentation

6.6 Sprint planen

Planen Sie Ihre Sprints am besten gemäß unserem Update-Zyklus. Damit versäumen Sie es nicht, beim nächsten Cloud Update dabei zu sein.

Schätzen Sie den Aufwand der Stories und nehmen Sie zu Beginn des Sprints so viele Stories auf, wie es Ihre Velocity zulässt.

Am Ende des Sprints können Sie die erledigten Stories abschließen und den gesamten Sprint beenden.

6.7 Use-Cases der Cloud App „Scrum-Projekte“

- Ein Scrum-Projekt erzeugen
- Eine Story erzeugen
- Einen Defect erzeugen
- Story bzw. Defect in den Backlog aufnehmen oder verwerfen
- Story bzw. Defect in den Sprint aufnehmen
- Hotfix anfordern unter Angabe einer Begründung
- Sprint starten
- Sprint durchführen
- Tasks erzeugen
- Story bzw. Defect abschließen
- Hotfix abschließen unter Angabe von Ursache und Maßnahmen
- Sprint beenden