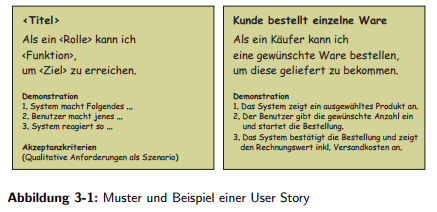
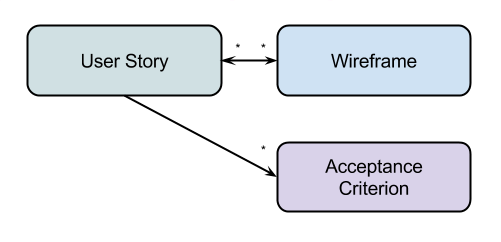
## Beschreibung einer User Story / Conversation

Ziel: Als Veranschaulichung, was die User Story tun soll, werden mögliche Aktivitäten als Demonstration aufgezählt. Diese werden in einem Gespräch (Conversation) mit dem Kunden oder PO herausgefiltert.

Bsp einer User Story mit Aktivitäten aus einer Conversation:



## Wireframe als Unterstützung für die User Story

Ziel: UI Design Wireframes (Scribbles) als Ergänzung zu User Stories.

Oft fehlen in der User Story Details, die Frontend-Entwickler für die Gestaltung der Benutzeroberfläche benötigen. Wireframes (Scribbles) können, ohne unnötig viel Aufwand, adäquate Informationen zu den User Stories bereitzustellen.

Wer, Was, Warum

* … Webshop Kunde möchte ich einen Login, um meine bisherigen Bestellungen zu sehen, zu kontrollieren und Rechnungen auszudrucken.
* … Geschäftsreisender der DB möchte ich meine Reservierungen in der DB App mehrfach täglich ändern, um flexibler umbuchen zu können.
* … AirBnB User, der einen “Business Trip” macht, möchte ich eine Rechnung, die meine Firmendaten enthält und wenigstens in Ansätzen einer formal korrekten Rechnung gleichkommt. Weil er es für seine private E/A Rechunung
* … audible Abonnent möchte ich bestellte Hörbücher einfach zurückgeben können, um mein Guthaben für relevante Titel auszugeben.
* … Atlassian Kunde möchte ich die Subdomain meiner Confluence Instanz ändern, um in der Namenswahl flexibler zu sein.

Userstory: Für den Mitarbeiter sollen alle Kunden aufgelistet werden, um bestimmte Kunden z.B. Für einen Serienbrief auswählen zu können.

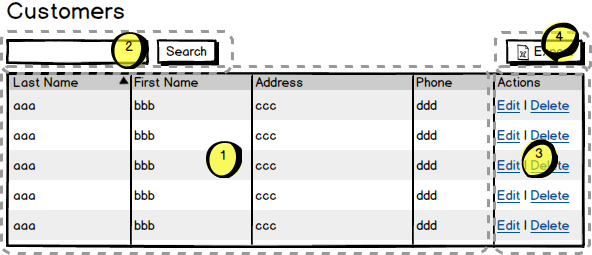
Skizziere wie dein Dialog aussehen kann

## Demo: Weitere Beispiele - User Stories sind Nutzergeschichten

User Stories sind Nutzergeschichten. Der Product Owner beschreibt, wie sich Anwender eine Software wünschen oder Kunden einen Liefergegenstand eines agilen Projekts, um ihre Geschäftsziele zu erreichen. Beispiele:

* Ein neuer Kunde soll sich bei einem E-Learning-Portal registrieren, um sich auf eine Zertifizierung vorzubereiten.
* Ein Kunde möchte Waren in einem Webshop auswählen und dann bestellen.
* Ein IT-Administrator möchte Datenbanken verwalten, indem er Datensätze anlegt, ändert oder löscht.
* Eine Abteilung in einem internationalen Konzern soll in einen neuen Bereich und damit in eine neue Struktur überführt werden.
* Ein E-Learning-Portal für Projektleiter, das Seminare anbietet, könnte für die Registrierung eines neuen Kunden diese User Story texten:
* „Als Neukunde möchte ich mich im E-Learning Portal registrieren, um mich auf die PMI-Zertifizierung vorzubereiten.“

Beispiel für ein Wireframe:



1. The table and its content
2. The search field
3. The edit and delete functionality
4. The ability to export

**Durch den Wireframe können wir auch Akzeptanzkriterien ableiten (Given-When-Then). Hier ein Beispiel für eine Autorensuche in einem Bibliotheksprogramm.**

**Übung: Schreibe für den Dialog Kundensuche alle möglichen Wenn-Dann Kriterien auf und leite mögliche Akzeptanzkriterien für die Abnahme ab.**

**Beispiele Akzeptanzkriterien**

1.User Story: **Als Teilnehmer einer Konferenz möchte ich in der Lage sein, mich online zu registrieren um Zeit und Papier zu sparen.**

**Akzeptanzkriterien**

* Ein Nutzer kann kein Formular absenden, ohne die Pflichtfelder auszufüllen.
* Informationen aus dem Formular werden in der Registrationsdatenbank gespeichert.
* Der Spam-Schutz ist aktiviert.
* Zahlungen können via Kreditkarte durchgeführt werden.
* Eine Bestätigungsmail wird dem Nutzer zugesendet, nachdem das Formular abgeschickt wurde.

2. User Story: **Als Onlinebanking-Kunde möchte ich für meine Alltagskonten einen laufenden Saldo sehen, sodass ich nach jeder Transaktion den aktuellen Kontostand kenne.**

**Akzeptanzkriterien**

* Der laufende Saldo wird dargestellt.
* Der laufende Saldo wird für jede Transaktion ausgerechnet.
* Der Kontostand wird für jede Transaktion während der gesamten Zeit dargestellt, in der Transaktionen verfügbar sind.
* Der Kontostand wird nicht dargestellt, wenn ein Filter angewendet wurde.

**Nimm eine Userstory deines Projektes her und erarbeite mit dem Wissen**

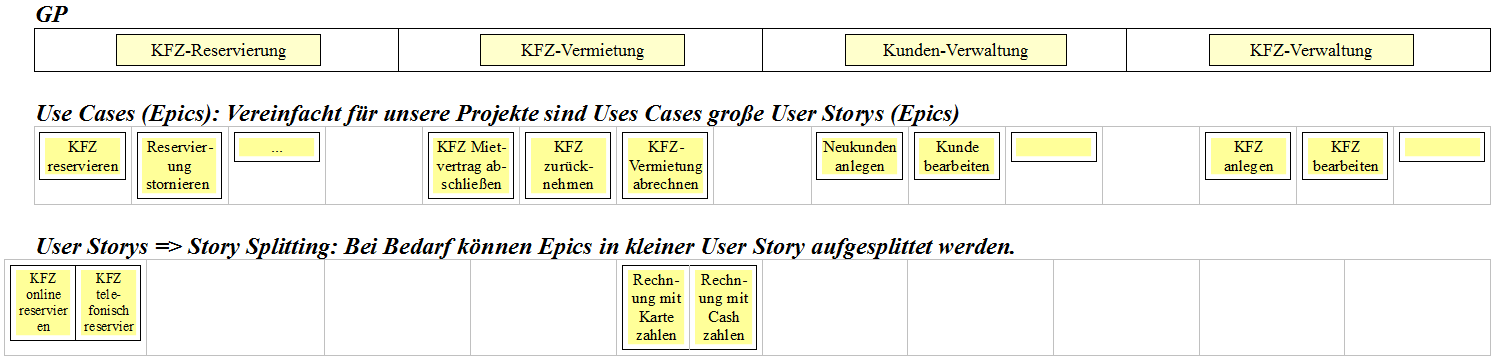
1. **Den Wireframe**
2. **Leite entsprechende Akzeptanzkriterien ab**

**Checklist: "Definition of Done"**

Jede User Story in Trello bekommt eine Checklist "Definition of Done" mit folgenden Elementen:

1. ALL Tasks done
2. ALL Acceptance Criteria done
3. Okay from PO
4. ALL JUnit tests done and green
5. All Java-Doc written
6. Git aktualisiert
7. Trello aktualisiert

Beispiel Fahrzeugreservierung (Teilen von großen Userstories in kleine)



* Als Verkäufer kann ich verschiedene Zahlungsmethoden benutzen, damit ich Bar oder mit Karten das Geld kassieren kann.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Titel: Zahlungsmethoden

Demonstration: Verbal

* Das System hat alles berechnet und es geht nur noch darum das Geld zu kassieren.
  + Eine Zahlungsübersicht wird angezeigt / bzw. kann ausgedruckt werden
  + Der Agent kann wählen zwischen Bar und Kreditkartenzahlung

1. Bei Barzahlung wird das Geld übergeben (Kassa muss vorhanden sein) wenn vom Kreditkartenunternehmen die Freigabe ermöglichen.
2. Bei Kreditkartenzahlung wird die Karte angegeben und dann die notwendigen Daten eingeben
3. Ein Beleg der Transaktion oder ein Mitteilung an den Kunden (mit allen Daten wird übermittelt)

LowLevel Wireframe

Akzeptanzkriterien:

* Es müssen Banktransaktionen, Banküberweisung, Visa, Paypal möglich sein.
* Die Daten werden auf einen Kundenterminal geheim eingegeben werden
* Es darf erst die Freigabe erfolgt sein, wenn vom Kreditkartenunternehmen die Freigabe erfolgt
* Mitteilung muss vorhanden sein mit folgenden Details

Ein Bild, das Entwurf, Text, Zeichnung, Handschrift enthält.

Automatisch generierte Beschreibung

User Story

* Als Kunde möchte ich online das Auto konfigurieren, um bei Abholung nicht zu lange warten zu müssen (bis das Auto mit den Wünschen gefunden wird).

Titel: Auto config

Demonstration: Der Kunde hat sich ein Auto ausgesucht und möchte es nun Konfigurieren

* Es werden ihnen Optionen zur Konfiguration angezeigt (Farbe, etc.)

LowLevel Wireframe  
Ein Bild, das Entwurf, Zeichnung, Text, Lineart enthält.

Automatisch generierte Beschreibung

Akzeptanzkriterien:

* Mehrere Konfigurationsmöglichkeiten sollten angegeben werden
* Am Ende der Konfiguration sollte alle Konfigurationen angegeben werden
* Als Kunde, möchte ich mir im Vorhinein ein 3D-Modell des gewünschten Autos ansehen, um das Fassungsvermögen zu kontrollieren.
* Als Kunde, sich das Auto als AR-Modell in Echtgröße anzusehen, damit der Hund reinpasst.
* Als Kunde, kann ich den Support kontaktieren, bei jeglichen Problemen
* Als Kunde möchte ich alle Autos filtern können um die Mietentscheidung zu erleichtern.
* Als Kunde möchte ich informiert werden, wenn Servicearbeiten (Räderwechsel, Ölwechsel, ...) anstehen, um nicht selbst machen zu müssen
* Als Kunde möchte ich online Termine vereinbaren können, ohne Anruf oder Mail

Aufgabe eine User-Story so machen wie ganz oben

Durchgängiges Beispiel:

<https://agilmanagen.de/user-stories-und-mockups/>

## Aufbau einer User-Story

Eine User-Story besteht immer aus

* einem Titel,
* dem Inhalt und
* den Akzeptanzkriterien.

Der Titel beschreibt kurz und knapp die Funktionalität, um die es in der Story geht – ähnlich dem Betreff einer E‑Mail.

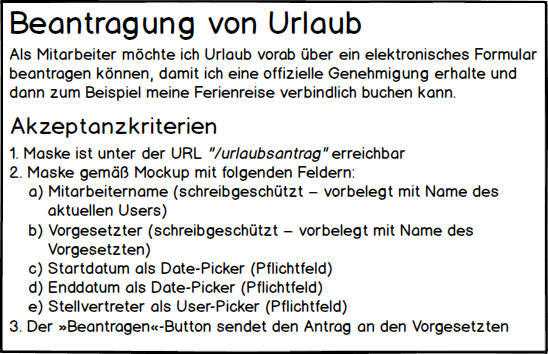
Der Inhalt besteht aus einem Satz und folgt immer dem selben Satzmuster: »Als Rolle möchte ich Wunsch können, um Nutzen.«

Die »Rolle« beschreibt dabei die Zielgruppe des Features. Bevor Du mit der Erstellung der User-Stories beginnst, solltest Du Dir also Gedanken machen, welche Typen von Anwendern für Dein System existieren. Wenn Du zum Beispiel ein System zur Beantragen von Urlauben definierst, wären folgende Rollen denkbar:

* **Mitarbeiter:** beantragt Urlaub und möchte Anträge einsehen
* **Vorgesetzter:** genehmigt Urlaub
* **Administrator:** installiert und administriert das System

Der »Wunsch« in der Satzschablone beschreibt das eigentliche Feature umgangssprachlich aus der Sicht des Anwenders, ohne auf weitere Details einzugehen.

Der »Nutzen« erklärt, warum ich als Anwender in der entsprechenden Rolle dieses Feature benötige.

[](http://agilmanagen.de/wp-content/uploads/2015/09/user-story.png)

User-Story mit Akzeptanzkriterien zur Beschreibung eines Urlaubsantrags

Ein Beispiel für eine vollständige User-Story findest Du im Kasten rechts. Das Schreiben dieser Story war nicht so einfach, wie Du jetzt beim Lesen denken magst. So fiel mir zunächst kein Grund für das Interesse des Mitarbeiters an diesem Vorgang ein.

Du wirst auch feststellen, dass der Nutzen bei einigen Stories relativ offensichtlich ist (Beispiel: »Als Anwender möchte ich eine Datei löschen können, …«. Nichtsdestotrotz habe ich mir angewöhnt ihn aufzuschreiben, damit alle Stories gleichförmig sind.

## Akzeptanzkriterien

Die Story gibt einen Überblick über die gewünschte Funktionalität. Allerdings wäre das zu wenig Information, um zu entscheiden, ob eine Implementierung vollständig ist. Daher nutzen wir Akzeptanzkriterien. Hierbei handelt es sich um eine Liste von Anforderungen, die erfüllt sein müssen, damit die Story als erledigt angesehen werden kann.

Auch diese Liste sollte auf eine Karteikarte – auf die Rückseite der Story – passen. Zu viele Akzeptanzkriterien sind ein Indikator dafür, dass Du die Story aufteilen solltest.

Die Akzeptanzkriterien für die oben genannte Beispiel-Story zum Beantragen von Urlaub findest Du im selben Kasten.

Wie Du siehst, halte ich die Akzeptanzkriterien eher in einem knappen Stil und lege hier keinen Wert auf vollständige Sätze. Bei mir sind die Akzeptanzkriterien stets nummeriert, um die Kommunikation zu vereinfachen.

In den Akzeptanzkriterien soll nur das aufgeführt werden, was Dir für eine Abnahme wirklich wichtig ist. Punkte, bei denen Dir die konkrete Umsetzung egal ist, kannst Du getrost weglassen und auf die Kompetenz Deines Entwicklungsteams vertrauen. Mir war hier zum Beispiel wichtig, dass die Maske unter einer bestimmten URL erreichbar ist und dass für die Datumsfelder Date-Picker anstelle einfacher Eingabefelder angeboten werden.

Die Akzeptanzkriterien sind allerdings verbindlich. Hast Du also an einer bestimmten Stelle etwas erwartet, was nicht explizit in den Akzeptanzkriterien gefordert war und Dein Team liefert das nicht, dann darfst Du die Story nicht deshalb ablehnen. Ein Änderungswunsch müsste hier in Form einer neuen Story erfolgen.

## Mockups

User-Stories sind praktisch. Wenn es aber um die Beschreibung von Oberflächen geht, genügt Text nicht mehr. Natürlich könntest Du das Layout in den Akzeptanzkriterien beschreiben, aber damit würden diese sehr umfangreich werden.

Bei mir gehört zu jeder User-Story, die zu einer Anpassung der Benutzeroberfläche führt ein »Mockup« oder »Wireframe«. Bei diesen grafischen Darstellungen steht das Layout einer Maske im Vordergrund – nicht das Design. Das erreichst Du, indem Du den Grafiken die Anmutung einer Stiftzeichnung gibst. Dadurch vermeidest Du von vornherein Diskussionen wie »das Element würde ich aber grün machen« oder »hier würde ich den Abstand um drei Pixel vergrößern«.

[](http://agilmanagen.de/wp-content/uploads/2015/09/mockup.png)

Ein Mockup beschreibt das Layout einer Maske, nicht das Design.

Im einfachsten Fall erstellst Du die Mockups mit Stift auf Papier. Aber natürlich gibt es dafür auch spezielle Tools. Ich selbst nutze [Balsamiq Mockups](https://balsamiq.com/products/mockups/). Damit kannst Du sehr einfach aussagekräftig Masken zusammenbauen. Diese kann man untereinander verlinken, so dass bei einer Vorführung beim Klick auf einen Button wirklich eine andere Maske geöffnet wird. So kannst Du das Verhalten der Anwendung sehr schön demonstrieren und vorab diskutieren.

Du wirst sehen: Schon beim Entwurf der Mockups musst Du Dir viel mehr über die eigentliche Anforderung nachdenken. Danach stellen sie eine hervorragende Grundlage für eine Diskussion mit den anderen Stakeholdern dar.

## Pseudo-Stories

Immer wieder taucht die Frage auf, wie man mit technischen Anforderungen an das System umgehen soll. Sollen diese auch als User-Stories in der Form »Als Entwickler möchte ich …« erfasst werden?

Am Ende gibt es hier kein richtig oder falsch. Ich habe für mich den Weg gewählt, dass ich ausschließlich Themen, die potentiell für den Kunden interessant sind, als Stories erfasse, nicht aber rein technische Themen. Diese werden bei uns als »Technical Requirements« im JIRA erfasst. Das hat natürlich den Nachteil, dass wir sie bei der Sprint-Planung getrennt berücksichtigen müssen, da sie in der Liste der Stories nicht auftauchen.

Ich hatte gerade kürzlich einen solchen Grenzfall: Es ging darum Sprachpakete aus der Entwicklungsumgebung heraus zu exportieren, die dann von einem externen Übersetzter übersetzt und vom Entwickler wieder in die Entwicklungsumgebung eingespielt werden können. Da diese Operation nicht vom Kunden selbst durchgeführt werden kann, ist es keine klassische User-Story. Die Tatsache, dass wir aber in der Lage sind, Übersetzungen extern durchführen zu lassen stellt durchaus ein Mehrwert für den Kunden dar. Daher hatte ich die User-Story mit aufgenommen: »Als Entwicklungsleiter möchte ich ein Sprachpaket exportieren und wieder importieren können, um so die Texte unserer Software von einem externen Übersetzter in verschiedene Sprachen übersetzen zu lassen.«

## Epics und Story-Decomposition

In Verbindung mit User-Stories begegnet einem häufig der Begriff der »Epics«. Hierbei handelt es sich um – ebenfalls umgangssprachlich formulierte – Platzhalter für mehrere User-Stories auf einem höheren Abstraktionsniveau. Diese werden dann im Laufe des Projekts im Rahmen der »Story-Decomposition« in eine Menge von User-Stories aufgebrochen.

Gerade wenn Du mit der Erfassung von Anforderungen für ein System beginnst, ist es hilfreich, sich von oben nach unten durchzuarbeiten. Du beginnst auf einem sehr abstrakten Niveau (zum Beispiel mit Epics) und verfeinerst dieses dann Stück für Stück.