



Usability in VR

Virtuelle Räume gestalten

Robert Schmitz, Barbara Streppel

In einer gelungenen VR-Anwendung fühlt sich der Benutzer als Teil der virtuellen Welt, die er kaum oder gar nicht als Illusion wahrnimmt. Das vollständige Eintauchen in eine virtuelle Umgebung bezeichnet man als Immersion. Sie hängt stark von der Bedienbarkeit der VR-Anwendung ab: Wenn die Bedienung nicht intuitiv oder gar unlogisch ist, geht die Immersion verloren; ist sie umständlich oder körperlich anstrengend, ermüdet sie den Anwender; im schlimmsten Fall stolpert er zum Beispiel über das Kabel der Brille und verletzt sich. Um dem vorzubeugen, brauchen gerade VR-Anwendungen eine hohe Usability.

Usability in VR (virtueller Realität) folgt aus unserer Sicht den gleichen Kriterien wie für Web- und Desktopapplikationen auch. Usability bedeutet, eine Anwendung gebrauchstauglich zu gestalten. Teil 11 der ISO 9241 nennt drei Leitkriterien für Usability, zu Deutsch Gebrauchstauglichkeit, einer Software [DIN]:

- Effektivität zur Lösung einer Aufgabe,
- Effizienz der Handhabung des Systems,
- Zufriedenheit der Nutzer einer Software.

Wesentlich für die effektive, effiziente und zufriedenstellende Nutzung einer Software sind die Dialoge. Sie bilden die Brücke zwischen Nutzer und Programm. In einer Idealwelt – oder sogar in gut aufgestellten Teams in der realen Welt – kümmern sich eigens ausgebildete Teammitglieder um die Usability. In der Praxis ist diese Rolle jedoch oft nicht besetzt. Die Gestaltung der Benutzerschnittstelle bleibt beim Entwickler.

Bei Desktopanwendungen und Apps gibt es viele etablierte Patterns; in VR sind sie gerade im Entstehen. Entwickler geraten leicht in einen Zielkonflikt: Schnell zu einer Lösung zu kommen, liegt im Entwicklergen. Die schnellste Lösung ist aber selten diejenige mit der höchsten Usability. Sich das bewusst zu machen, ist ein guter erster Schritt. Wissen aufzubauen, wie eine Anwendung besser benutzbar wird, ein wichtiger zweiter Schritt.



Robert Schmitz ist IT-Architekt für Augmented und Virtual Reality (AR/VR) und Usability-Experte. Er verantwortet das Thema AR/VR bei MaibornWolff.
E-Mail: robert.schmitz@maibornwolff.de



Barbara Streppel ist AR/VR-Entwicklerin. In ihrer vom VDI ausgezeichneten Masterarbeit beschäftigte sie sich mit „Interaktionsmöglichkeiten in der VR“. E-Mail: barbara.streppel@maibornwolff.de

Sieben Grundlagen guter Dialoggestaltung

Der Leitfaden für Usability ist durch die ISO 9241 beschrieben. Sie definiert sieben Grundlagen guter Dialoggestaltung:

- Aufgabenangemessenheit (suitability for the task),
- Lernförderlichkeit (suitability for learning),
- Selbstbeschreibungsfähigkeit (self-descriptiveness),
- Erwartungskonformität (conformity with user expectations),
- Individualisierbarkeit (suitability for individualization),
- Steuerbarkeit (controllability),
- Fehlertoleranz (error tolerance).

Sie haben für VR ebenso Geltung wie für Desktop-, Web- und Mobile-Anwendungen. Ein VR-Developer sollte sie besonders beherrschen. VR kommt dem Nutzer näher als jede andere Anwendung: Nutzer können durch die Anwendung stärker irritiert werden als beispielsweise in einer App. Und: Patterns entstehen gerade erst. Welche guten Beispiele sich durchsetzen werden, ist noch nicht abzusehen. Umso wichtiger ist das eigene Grundverständnis, was ein gutes Bedienkonzept ausmacht.

Wir zeigen in diesem Artikel, welche Bedeutung jede der Grundlagen im VR-Kontext hat.

Aufgabenangemessenheit

Die Aufgabenangemessenheit (englisch: suitability for the task) ist nach ISO 9241 für ein Dialogelement dann erfüllt, wenn „ein Dialog den Benutzer unterstützt, seine Arbeitsaufgabe effektiv und effizient zu erledigen“ [Wiki-a]. Die Forderung nach Effektivität bezieht sich auf das Lösen der gestellten Aufgabe, Effizienz entsteht durch die Handhabung des Systems mit möglichst wenig Aufwand. Kurz gesagt: Genau wie bei Screen-Anwendungen sollen Aufgaben auch in der virtuellen Realität mit möglichst wenigen Interaktionen zu erledigen sein. So kann man die Anwendung effizient benutzen (s. Abb. 1).



Abb. 1: Mit dieser Gießkanne kann man Blumen nicht effektiv und effizient gießen, sie ist für die Aufgabe nicht angemessen

Effektivität kann beispielsweise Übersichtlichkeit bedeuten. Sie reduziert die kognitive Last. In einer Anwendung sollten zum Beispiel nur die Objekte benutzbar sein, die man zum Erledigen

der Aufgabe in der virtuellen Arbeitsumgebung benötigt. Baut man ein Spielzeugauto an einem virtuellen Basteltisch zusammen, liegen auf diesem Tisch im besten Fall nur die Bauteile für das Spielzeugauto.

Falls aus gestalterischen Gründen auch dekorative Objekte gezeigt werden, sollten diese Elemente für den Nutzer als nicht-relevant erkennbar sein. Hier bieten sich gestalterische Tricks an: Zum Beispiel können nicht-relevante Objekte abstrakt statt realistisch modelliert werden. So erfasst sie der User intuitiv als andersartig. Grundsätzlich sollten wir jedoch Dekorationen vermeiden. VR-Anwendungen müssen keine Computerspiele sein. Sie fokussieren auf das Erledigen von Aufgaben.

Der Aspekt Aufgabenangemessenheit ist in einer VR-Welt besonders wichtig, da die Nutzer erst einmal noch nicht wissen, was man darin überhaupt tun kann. Dann hilft es, wenn das Bedienelement für den nächsten Schritt optisch hervorgehoben wird – etwa durch ein Aufleuchten, einen Farbwechsel oder Ähnliches.

Selbstbeschreibungsfähigkeit
Selbstbeschreibungsfähigkeit (englisch: self-descriptiveness, [Wiki-f]) führt dazu, dass ein Nutzer erkennt

- welche Aktionen ausführbar sind,
- wo er sich in der Anwendung befindet,
- ob beziehungsweise wohin er sich bewegen kann und
- welche Konsequenzen die eigene Handlung hat.

Dieses Wissen ist entweder unmittelbar nachvollziehbar oder jederzeit leicht abrufbar. Informationen dazu können zum Beispiel über eine Kontexthilfe (s. Abb. 2) eingeblendet werden.

VR erlaubt einige originäre Bedienkonzepte, die uns in 2D, etwa Apps oder Screen-Anwendungen, nicht zur Verfügung stehen: Entwickler können beispielsweise Button-Belegungen für Controller mit Piktogrammen versehen, die mögliche Aktionen anzeigen. Je nach Kontext lassen sich Buttons unterschiedlich belegen. Auch das sogenannte „Gaze“ ist der VR-Welt eigen: Wenn der Nutzer ein Objekt lange betrachtet – daher „Gaze“, wie der Blick – können Hilfe-Fenster oder Informationen über mögliche Aktionen eingeblendet werden. Zusätzlich bieten viele VR-Anwendungen bereits kleine „Tooltips“ an den Knöpfen des Controllers.

Wenn Selbstbeschreibungsfähigkeit in einer VR-Anwendung gegeben ist, lernt der Nutzer schnell. Hilfsmittel wie Texteinblendungen oder Piktogramme werden aus unserer Sicht in VR die Regel bleiben. Es gibt für ihre Bedienung schlicht zu wenige etab-



Abb. 2: Eine Kontexthilfe macht Informationen unmittelbar nachvollziehbar

lierte Patterns oder Standards, wie wir sie bei Smart Devices mit Wischen und Zoomen kennen. Bisher hat jeder Anbieter ein eigenes Bedienkonzept für seine Controller, und die Knopfbelegungen können je nach Anwendung noch einmal variieren. Deshalb muss jede Anwendung dem Nutzer beibringen, welche Optionen er hat. Hier sind wir beim nächsten Gestaltungsprinzip: der Lernförderlichkeit.

Lernförderlichkeit: den Nutzer ausbilden

Wie schnell ein Anwender lernt, eine VR-Welt zu bedienen, wird mit Lernförderlichkeit (englisch: suitability for learning, [Wiki-d]) beschrieben. Dafür gibt es bereits einige erprobte Ansätze: Der Nutzer lernt zum Beispiel eine VR-Anwendung bei einem Rundgang mit einem Avatar kennen. Das sind virtuelle Personen oder Charaktere, die vom Anwender innerhalb der Anwendung als Begleiter wahrgenommen werden. Sie erklären beispielsweise die Funktion der Controller oder weisen auf wichtige interaktive Elemente der virtuellen Umgebung hin. Während die oben genannten Piktogramme sich vor allem für direkte Kontextinformationen eignen, können Avatare komplexere Steuerungsabläufe erklären.

Eine andere Option sind Tutorien zu Beginn der Anwendung, in denen der Nutzer die Anwendung Schritt für Schritt kennenlernt. Gelerntes lässt sich verstärken, wenn Aktionen wie „Objekt auswählen“ oder „Stift in die Hand nehmen“ von einem optischen, akustischen oder haptischen Feedback begleitet werden. So lernt der Anwender, dass die Aktion geglückt ist. Optimalerweise ist das Feedback durch die VR-Anwendung hindurch konsistent.

Lernförderlichkeit beschreibt die Anforderung, dass ein Nutzer die Bedienung eines Systems schnell erlernen oder sich erschließen kann. Damit das Gelernte in der weiteren Erfahrung mit der Anwendung nutzt, muss sich die Anwendung verhalten, wie erwartet (s. Abb. 3). Damit bekommt Erwartungskonformität in VR einen hohen Wert – paradoxerweise, denn es gibt noch wenig User-Erwartungen.

Erwartungskonformität

Erwartungskonformität (englisch: conformity with user expectations, [Wiki-b]) ist gegeben, wenn die Gestaltung der Anwendung zu Vorwissen, Erfahrung und Fähigkeiten des Anwenders passt. Erwartungen entstehen aus verschiedenen Quellen, zum Beispiel aus Erfahrungen in der realen Welt, aus kultureller Prägung oder als Erwartung an Konsistenz innerhalb einer Anwendung.

Menschen rechnen damit, dass Objekte sich so verhalten, wie sie das aus der realen Welt kennen: Ein geworfener Stift fällt zu Bo-

den, Knöpfe sind drückbar, Türschlüssel öffnen Türen und Schraubenschlüssel passen auf einen Schraubkopf. Wenn diese Elemente sich erwartungskonform verhalten, sind sie übrigens auch lernförderlich. Das Mapping von Aktion auf ausgelösten Effekt ist damit eng verwandt: Wenn wir an unserem virtuellen Basteltisch eine Schraube anziehen, möchten wir wie in der realen Welt einen Schraubendreher nehmen, ihn ansetzen und im Uhrzeigersinn drehen; statt auf einen Stellvertreterknopf zu drücken, der die Schraube wie von Zauberhand auf „fest“ stellt.

Zusätzlich werden Erwartungen vom kulturellen Kontext geprägt, an die sich die Anwendung anpasst: Ein Maßband sollte für Benutzer in den USA in Zoll verfügbar sein und an europäischen Standorten im metrischen System. Das Wording sollte dem Wissen des Nutzers entsprechen. Virtuelle Tools in einer Trainings-Anwendung für Produktionsarbeiter sollten aussehen wie später am realen Band. Und wenn es schon bekannte Bildwelten gibt, sollten sie in der Anwendung aufgegriffen werden.

Außerdem spielt der Grad an visueller Abstraktion in die Erwartungskonformität hinein. Hier geht es um Konsistenz: Wenn eine VR-Anwendung aus zwei Räumen besteht, sollten sie etwa gleich abstrakt oder realistisch gestaltet sein. Eine Comic-Welt im ersten Raum, gefolgt von einer fotorealistischen Darstellung im zweiten Raum irritiert den Nutzer und macht die Orientierung unnötig schwierig – bis hin zur Frage, ob man noch in der gleichen Anwendung ist. Konsistenz vermittelt dem Nutzer über Zeit Orientierung in und Vertrautheit mit der Anwendung.

Individualisierbarkeit

Individualisierbarkeit (englisch: suitability for individualization, [Wiki-e]) gibt dem Nutzer die Möglichkeit, Bedienkonzepte, Tastenbelegungen oder Tastaturbefehle selbst festzulegen. Gamer kennen von ihren Spielkonsolen alternative Controller-Schemata; PC-Anwender wählen zwischen Scroll-Richtungen für Maus oder Touchpads. Mit den Auswahlmöglichkeiten holt man den Nutzer beim Gelernten ab. Doch was bringt das im wenig standardisierten Umfeld von VR-Anwendungen? Geht es doch darum, den Nutzer erst einmal an Patterns zu gewöhnen und Standards lernen zu lassen – oder?

Tatsächlich ist Individualisierbarkeit in VR wesentlich. Denn in diese Kategorie fallen Anpassungen, die VR überhaupt erst benutzbar machen: Linkshänder werden nicht plötzlich zu Rechtshändern, weil sie eine Schraube via Controller an einem nur virtuellen Regal anziehen. Das Spielzeugauto bauen wir leichter an einem virtuellen Werkzeuggestisch zusammen, wenn wir die Beleuchtung auf dem

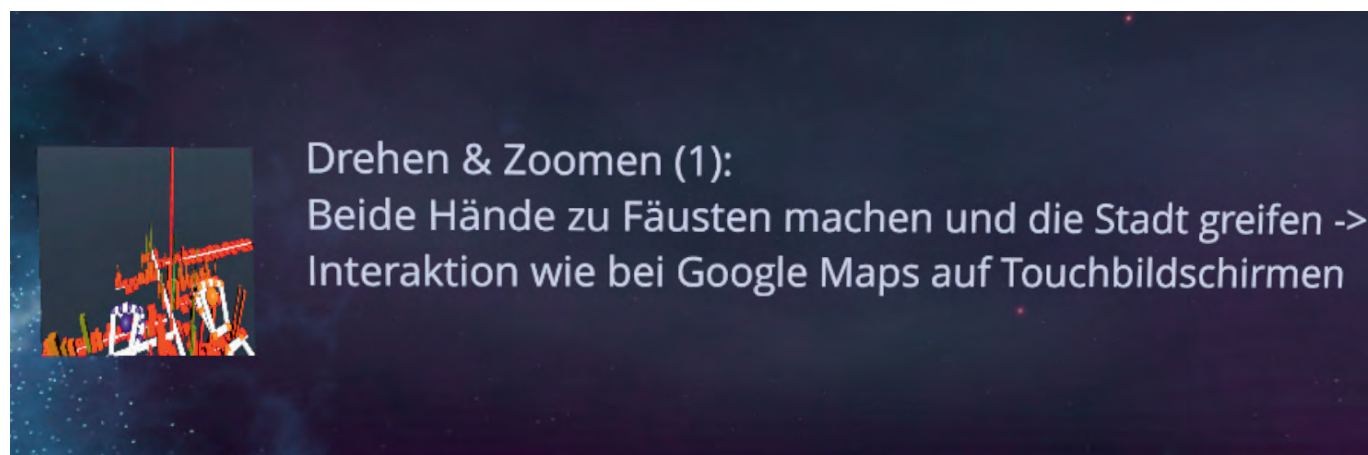


Abb. 3: Da es nur wenige Bedienmuster für VR gibt, lernen Nutzer die Bedienung eines Systems während der Anwendung

Tisch an unseren Lichtbedarf oder die Tischhöhe an die Körpergröße anpassen können.

Ebenso bilden wir verschiedene Fertigniveaus mit Anfänger- und Experten-Modi ab – auch das ist eine Interpretation von Individualisierbarkeit. Neulinge nutzen Bedienhinweise oder die Einführung durch einen Avatar. Geübte Nutzer schalten Hilfeblendungen oder Tutorien ab. Beide Nutzergruppen profitieren von der Auswahlmöglichkeit. Ein anderes Beispiel für Individualisierbarkeit ist der „Comfort-Mode“, etwa bei Google Earth VR. Dieser optionale Betriebsmodus der Anwendung schwächt die Auswirkung von starken Bewegungen durch optische Tricks ab. So wird verhindert, dass Nutzern mit weniger robusten Mägen in der VR-Welt schlecht wird.

Über diese Anpassungen hinaus wird Individualisierbarkeit für VR wichtiger werden, sobald sich Bedienkonzepte etabliert haben: Ein Nutzer fühlt sich wie in der realen Welt, wenn er oder sie zum Beispiel persönliche Fotos an virtuelle Wände hängen kann oder den Lieblingssessel importiert. Je länger der Anwender in der virtuellen Umgebung verbringt, desto relevanter können „Wohlfühlfaktoren“ werden.

Steuerbarkeit

Steuerbarkeit (englisch: controllability, [Wiki-g]) beschreibt den Grundsatz, dem Nutzer jederzeit möglichst viel Kontrolle zu geben. Eine Anwendung ist steuerbar, wenn der Nutzer einen Prozess in VR abbrechen, neu starten oder umkehren kann. Dazu gehört beispielsweise, dass der Nutzer selbst bestimmt, ob er Tutorien ansieht. Ein anderes Beispiel ist die Möglichkeit, Menüs im Blickfeld zu verschieben.

Fehlertoleranz

Fehlertoleranz (englisch: error tolerance, [Wiki-c]) lenkt den Blick darauf, wie eine Anwendung auf Fehler reagiert. Kann ein Benutzer sich beispielsweise in eine Situation manövrieren, aus der er oder sie nicht mehr oder nur mit großem Aufwand herauskommt? Diese Dimension gehört zu den Kriterien guter Dialoggestaltung, weil sie Nicht-Bedienbarkeit vorbeugt und Schaden verhindert.

Für VR-Anwendungen lässt sich das an einem einfachen Beispiel zeigen: Viele Benutzer werfen Stifte, Schraubenzieher und andere Objekte durch die virtuelle Umgebung – einfach, weil es Spaß macht. Dabei kann es vorkommen, dass die Objekte außerhalb der Reichweite des Anwenders landen. Wie bekommt er dann den Schraubenzieher zurück, den er fürs Bauen des Spielzeugautos braucht? Denkbar wäre, dass der Nutzer den Schraubenzieher holen muss. Hier stößt VR buchstäblich, sprich räumlich an Grenzen. Deswegen ist es in diesem Beispiel sinnvoller, den Schraubenzieher über ein Menü neu erzeugen zu lassen – oder wenn er einfach neben dem Nutzer wieder auftaucht.

Auch die Navigationsmechanik muss für VR abgesichert werden. Ein Benutzer kann sich innerhalb eines vom Programmierer definierten Bereiches der Umgebung bewegen. Geht er weiter, steht er plötzlich „in einer Wand“ und kommt nicht mehr zurück; oder er fällt über den Rand der endlichen virtuellen Welt. Das passiert öfter, als man denkt. Eine Lösung wäre das automatische Zurückteleportieren an einen sicheren Ort.

Ebenso wichtig ist es, Fehlerquellen zu vermeiden. Aus dem Web kennen wir zum Beispiel ausgegraute Buttons; sie werden erst anklickbar, wenn beispielsweise der Nutzer die Datenschutzbestimmungen per Haken akzeptiert. Diese Logik der Dialoggestaltung gibt es auch in einer VR-Welt. Beim Werfen oder Teleportieren

zeigen etwa Zielmarker, welche Bereiche als Landezone zulässig sind und welche nicht.

Usability-Grundlagen für Entwickler

Jeder Entwickler kann die sieben Grundlagen guter Dialoggestaltung einsetzen, um seine Anwendung auf Usability zu überprüfen. Das hat nichts mit Schönheit oder Geschmack zu tun. Ziel ist die Gebrauchstauglichkeit einer Anwendung. Menschen sollen sie möglichst effizient und effektiv benutzen können.

Damit gehören die sieben Grundlagen guter Dialoggestaltung aus unserer Sicht in den Werkzeugkasten jedes Anwendungsentwicklers – gleich für welche Plattform. Am besten verankert man sie in den Iterationen des Entwicklungsprozesses, etwa bei Design und Test der Benutzerschnittstelle beziehungsweise der virtuellen Umgebung.

Viele unserer Entwickler-Kolleginnen und -Kollegen haben die Grundlagen im Kartenformat auf dem Schreibtisch. Wichtige Dialogelemente, VR-Welten, Seiten oder App-Screens prüfen wir Schritt für Schritt für jede Kategorie.

Intuitiv geglückt? Das entscheidet der Benutzer

Eine gute VR-Anwendung gibt dem Anwender das Gefühl, Teil einer Situation zu sein, die es in Wirklichkeit gar nicht gibt. Das ist die eingangs erwähnte Immersion. Sie tritt wahrscheinlicher ein, wenn sich Elemente oder Aktionen einer Anwendung erwartungskonform verhalten und wenn sie lernförderlich, selbsterklärend und fehlertolerant gestaltet sind. Denn dann kann der Nutzer eine Anwendung bedienen, ohne darüber nachzudenken – es funktioniert intuitiv. Als Entwickler oder Designer können wir Lösungen auf den Prüfstand der oben erläuterten Grundsätze stellen.

Ein guter Anfang ist es, die eigenen Kreationen an festen Punkten im Entwicklungsprozess kritisch anhand von Usability-Leitlinien zu überprüfen. Das letzte Wort hat jedoch der Nutzer. Der beste Expertentest wird vom Feedback der Nutzer überstimmt. Wenn sie ein Bedienkonzept nicht annehmen, ist es Zeit für eine Iteration!

Dabei besteht in der VR-Entwicklung durchaus die Gefahr, sich in schöne Designs und Bedienkonzepte zu verlieben. Die noch wenig standardisierte VR-Welt lädt dazu ein, eigene Standards und Lösungen zu entwickeln. Natürlich können sich Standards und Patterns nur entwickeln, wenn wir Entwickler sie vorschlagen. Und dann das Feedback der Nutzer ernst nehmen, ob sie unsere Vorschläge verstehen und annehmen. Dabei hilft es, wenn wir rechts und links schauen, welche Lösung für ein Gestaltungsproblem andere finden. Jede gefundene Umsetzung sollten wir als Zwischenlösung betrachten – bis wir eine bessere erfinden.

Links

- [DIN] Deutsches Institut für Normung e. V., <https://www.din.de/de/meta/suche/62730!search?query=9241-110>
- [Wiki-a] <https://de.wikipedia.org/wiki/Aufgabenangemessenheit>
- [Wiki-b] <https://de.wikipedia.org/wiki/Erwartungskonformität>
- [Wiki-c] <https://de.wikipedia.org/wiki/Fehlertoleranz>
- [Wiki-d] <https://de.wikipedia.org/wiki/Lernförderlichkeit>
- [Wiki-e] <https://de.wikipedia.org/wiki/Individualisierbarkeit>
- [Wiki-f] <https://de.wikipedia.org/wiki/Selbstbeschreibungsfähigkeit>
- [Wiki-g] [https://de.wikipedia.org/wiki/Steuerbarkeit_\(Dialog\)](https://de.wikipedia.org/wiki/Steuerbarkeit_(Dialog))