

Erstellung von Software-Benutzerdokumentation

Die Benutzerdokumentation, also die Beschreibung der entwickelten Softwareanwendung für den Endanwender, wird bei der Entwicklung oft zum Thema und ebenso oft wieder vom Tisch geschoben. Mitunter bleibt daher die so sorgfältig entwickelte Applikation am Ende ohne Benutzerhandbuch, und so manche nützliche Funktion entgeht dem User. Das ist schade, denn der Anwender würde die neue Software gerne bestmöglich nutzen, um die gewünschten Tätigkeiten besser, schneller oder einfacher durchführen zu können.

Auf diesen Seiten geben wir daher eine Übersicht über den Erstellungsprozess für eine professionelle Benutzerdokumentation.

Wir behandeln dabei folgende Aspekte:

1. Motivation
2. Konzeption & Umsetzung
3. Inhalt
4. Umfang
5. Medien
6. Zeitplan

Die folgenden Abschnitte behandeln alle wesentlichen Punkte, die bei der Erstellung einer Benutzerdokumentation zu berücksichtigen sind.

1. Motivation

Nutzen für den Softwarehersteller

Die Erstellung einer Benutzerdokumentation zum Softwareprodukt bringt für das Software herstellende Unternehmen bzw. die Entwicklungsabteilung konkreten wirtschaftlichen Nutzen:

- *Supportfälle werden minimiert*, da sich Fragen zur Bedienung der Software von selbst erledigen.
- Die Dokumentation *dient intern als Referenz* und als Archiv der Änderungen von Release zu Release.
- *Neue Mitarbeiter* erhalten eine rasche und Ressourcen sparende Möglichkeit, sich anhand des User Manuals mit dem Produkt vertraut zu machen.
- *Wettbewerbsvorteile des Produkts werden besser genutzt*, da man mit guter Dokumentation beim potenziellen Abnehmer punktet.
- Die Software erhält ein *zusätzliches Qualitätsmerkmal*, das man auch vorzeigen kann.

Nutzen für den User

Auf der Seite der Anwender gibt es drei wesentliche Gründe, die für eine gute Benutzerdokumentation sprechen:

- *Referenz*: Auch wenn der User im Rahmen einer Benutzerschulung in den Umgang mit der Software eingewiesen wurde, so stellen sich doch während der täglichen Arbeit immer wieder Fragen zur Anwendung. Man ist sich nicht sicher, wo eine bestimmte Funktion der Software zu finden ist oder welche Eingaben zum gewünschten Ergebnis führen. In diesem Fall dient ein gutes Handbuch als Nachschlagewerk, das eine detaillierte Beschreibung des gewünschten Vorgangs liefert. Fehlt die Benutzerdokumentation, wendet sich der Benutzer im besten Fall an die Hotline des Herstellers, im schlimmsten Fall sucht er nach einem alternativen Produkt.
- *Lehrbuch*: Gerade bei Business Software erhalten viele Anwender nicht die Möglichkeit, Schulungen für die Anwendung zu besuchen. Die Software wird bereits seit längerer Zeit im Unternehmen verwendet oder es ist kein Budget für weitere Schulungen vorhanden. Neue User werden also von Kollegen in die Benutzung der Applikation eingewiesen und verursachen dadurch hohen Zeit- und Kostenaufwand. Eine mitgelieferte Benutzerdokumentation beantwortet viele Anfängerfragen sofort und ermöglicht den produktiven Einsatz neuer Mitarbeiter zu einem früheren Zeitpunkt.
- *Überblick*: Der User erhält eine allgemeine Einführung und eine Beschreibung, was er mit der Software machen kann. Er erfährt, wie sie ihn bei seiner Tätigkeit unterstützt und was er dabei beachten muss. Darüber hinaus bekommt er eine Vorstellung davon, wie die Software Informationen verarbeitet, welche Daten an welcher Stelle relevant sind und welche Auswirkungen seine Eingaben auf den beabsichtigten Arbeitsschritt haben. Der Anwender erfährt, aus welchen Teilen die Software besteht und welche grundsätzlichen Möglichkeiten es gibt, diese Module oder Komponenten kombiniert zu nutzen.

In der Softwareentwicklung sind verschiedene Arten der Dokumentation bekannt und erforderlich (siehe auch die Fachbeiträge "Dokumentation in der Software-Entwicklung" und „Übersicht der UML-Diagramme auf der Infracsoft Website unter www.infracsoft.at/service). Wir möchten hier einerseits eine Definition des Begriffs der Benutzerdokumentation und deren Abgrenzung zur Technischen oder Design Dokumentation liefern. Andererseits werden wir beschreiben, welche Informationsquellen für die Erstellung der Benutzerdokumentation als Grundlage herangezogen werden können.

2. Konzeption & Umsetzung

Wie jedes Entwicklungsprojekt auch, erfordert die Erstellung von Benutzerdokumentation die Phasen Planung, Design und Umsetzung. Jede dieser Stufen ist auch als Teil des Softwareentwicklungsprozesses zu verstehen und läuft am besten parallel zu den korrespondierenden Phasen der Produktentwicklung. So wird die Benutzerdokumentation zu einem integrierten Bestandteil der Applikation und entfaltet ihre nützliche Wirkung für den User. In der Konzeptionsphase definiert man, was beschrieben wird und für wen und in welcher Form die Dokumentation geliefert wird. Spätere Änderungen sind zwar möglich, aber immer mit zusätzlichem Zeit- und Kostenaufwand verbunden.

Planung der Benutzerdokumentation

Die Erfahrung zeigt, dass viele Benutzerdokumentationen bereits in diesem Stadium scheitern, da niemand für die Erstellung des User Manuals zuständig ist. Das ergibt sich aus folgender Problematik: Damit ein Benutzerhandbuch für einen User nützlich und praktisch ist, muss es die Software aus seinem Blickwinkel betrachten. Daher muss sich der Verfasser der Dokumentation - sei er jetzt selbst in der Entwicklung, im Support oder im Produktmanagement des Unternehmens angesiedelt - zum Teil in die Lage eines Users versetzen, der die Software noch nie zuvor gesehen hat. Nur weil man eine Funktion selbst schon tausend Mal getestet oder sie sogar programmiert hat, bedeutet das noch lange nicht, dass ihre Funktionsweise auch einem Benutzer sofort einleuchtend erscheint.

Die Planungsphase legt also als erstes fest, wer die Benutzerdokumentation erstellt und für wen sie erstellt wird. Dem Dokumentationsverantwortlichen kommt eine Art Abteilungsübergreifende Rolle zu. Er muss zu diesem Zweck aus allen verfügbaren Quellen Informationen über die tatsächlichen Endbenutzer der Softwareanwendung zusammentragen. Diese Infos erfragt er beim Produktmanagement, der Entwicklung, dem Support oder der Geschäftsleitung. Bisweilen kommt nützliches Feedback sogar direkt von bestehenden Benutzern im Rahmen von User Clubs oder ähnlichen Informationsveranstaltungen des eigenen Unternehmens.

Ziel der Planungsphase ist es, ein Benutzerprofil zu erstellen, in dem sich die Geschichte eines typischen Users widerspiegelt, und zwar von seinem ersten Kontakt mit der Software bis zur routinemäßigen Verwendung. Dazu werden die Antworten auf folgende Fragen gesucht:

- *Was sind die fachlichen Vorkenntnisse des Users?*
- *Wie tiefgehend sind seine Computerkenntnisse?*
- *Welche Sprache spricht er?*
- *Welche Terminologie ist ihm geläufig, welche nicht?*
- *Welche andere Software verwendet er?*
- *Wie häufig verwendet er die zu dokumentierende Software?*

Aus den Antworten auf diese Fragen ergeben sich Hinweise, wie umfangreich die Dokumentation sein soll und was sie enthalten muss. Hier legt man auch fest, in welchem technischen oder fachlichen Stil die Dokumentation verfasst wird. Es kann sich herausstellen, dass die User aus unterschiedlichen Gruppen mit verschiedenen Anforderungen bestehen. In der Folge kann die Dokumentation in mehrere Teile gegliedert werden. Es entsteht z. B. ein Installationshandbuch, ein Einführungshandbuch und ein Referenzhandbuch zusätzlich zum beschreibenden User Guide.

Design der Benutzerdokumentation

Sobald man weiß, wer die Benutzerdokumentation schreibt und wer eigentlich die User sind, kann man die inhaltlichen Grundlagen für die Benutzerdokumentation schaffen. In der Fachwelt bezeichnet man die Sammlung solcher Regeln und Bestimmungen gerne als Documentation Styleguide. Ein solcher Styleguide ist unerlässlich, wenn mehrere Personen an der Erstellung der Benutzerdokumentation arbeiten.

Der Styleguide ist die Technische Dokumentation und die Design Dokumentation für die Benutzerdokumentation. Er kann Normen enthalten, die von detaillierten Terminologiesammlungen über Checklisten für die standardisierte Beschreibung einzelner Softwareelemente bis zu Designvorgaben für Dokumentvorlagen reichen.

Eine wichtige Aufgabe des Styleguides ist es, sowohl die Erstellung als auch das Aussehen der fertigen Dokumentation im Vorfeld zu definieren und zu vereinheitlichen. Denn eine der Hauptaufgaben von Benutzerdokumentation ist es, dem User übersichtlich und verständlich zu präsentieren, wie die Software aufgebaut ist und wie sie funktioniert. Daher muss auch der Aufbau der Dokumentation selbst klar und eindeutig sein.

Vollständige Styleguides gehen sogar soweit, bestimmte wiederkehrende Formulierungen zu normieren, damit ähnliche Vorgänge in der Software in gleicher Weise beschrieben werden. Dadurch kann sich der Benutzer auf die wesentlichen Anweisungen konzentrieren und wird nicht von ständig wechselnden Formulierungen abgelenkt. Das ist ein Beispiel dafür, dass man sich bei der Erstellung von Benutzerdokumentation immer wieder überlegen sollte, wie man dem Benutzer das Leben möglichst leicht macht. Er ist ohnehin damit gefordert, die Handhabung einer für ihn meist vollkommen neuen Software zu erlernen. Diese Unterstützung für den User ergibt sich nahezu von selbst, wenn man den Aufbau der Dokumentation vom einzelnen Abschnitt bis zum kompletten Handbuch von Beginn an gewissenhaft plant.

Ein wichtiger Teilaspekt der Designplanung von Benutzerdokumentation ist das Layout des Dokuments. Also: Was sieht der Benutzer, wenn er einen raschen Blick in das Handbuch wirft? Marginalien und Icons sind wichtige Stilmittel für gedruckte Dokumentation. Sie erleichtern dem User die Orientierung im Dokument, so dass er die gewünschten Informationen rasch auffindet. Verzeichnisse und Indizes sind unerlässlich bei der Suche nach bestimmten Informationen. Querverweise und Links sind wichtig für Online-Hilfen und PDF-Manuals, ebenso wie übersichtliche Navigationsmöglichkeiten. Die richtige Nomenklatur bei Überschriften und die Verwendung von Nummerierungen und Tabellen bringen die Informationen in eine übersichtliche Form. Der Benutzer erfährt auf einen Blick, was für ihn wichtig ist.

An dieser Stelle im Dokumentationsprozess empfiehlt es sich, auch an eine Analyse der in der Software verwendeten Terminologie zu denken. Dabei sind zwei Aspekte besonders hervorzuheben: Die verwendeten Begriffe müssen eindeutig und einheitlich sein. Für Elemente mit unterschiedlichen Bedeutungen müssen jeweils eindeutige und unverwechselbare Begriffe verwendet werden. Bei gleichbedeutenden Funktionen oder Elementen ist jedoch immer derselbe Begriff zu verwenden. Diese Tätigkeit im Rahmen der Dokumentation kann zur Folge haben, dass Elemente der Benutzeroberfläche abgeändert werden müssen. Unterschiede in der GUI-Nomenklatur treten oft auf, wenn mehrere Entwickler an verschiedenen Modulen oder Subsystemen der Anwendung arbeiten. Bei der Dokumentation wird das Produkt als Ganzes betrachtet. Daher fallen bei der Beschreibung der Funktionalitäten unterschiedliche Begriffe für gleiche Elemente oder Funktionen sofort auf.

Spätestens während der Designphase der Benutzerdokumentation sollte man auch darüber nachdenken, in welcher Form die Dokumentation zur Verfügung gestellt wird. Da hier die Software selbst ein entscheidender Faktor ist, bestimmt auch die Plattform die Möglichkeiten. Hostanwendungen bieten beispielsweise nur sehr eingeschränkte Möglichkeiten einer Online-Hilfe im klassischen Sinn. Bei Web Applications wird man sich vielleicht für eine Embedded Help entscheiden, während bei Windows-basierten Anwendungen oft die F1 Online Help die Hilfe der Wahl ist.

Wenn die Entscheidung über Online-Hilfe nicht oder erst später getroffen werden soll, entscheidet man sich meist für eine Dokumentation in PDF-Format. Grundsätzlich gilt für die Dokumentation: so einfach wie möglich anfangen. Es macht keinen Sinn, sich von Null weg eine hoch komplizierte integrierte Hilfe mit kontextsensitiven Popups zu jedem Bildelement zu wünschen. Man bemerkt dann oft in der Umsetzungsphase, dass dafür das gesamte System neu gestaltet werden muss. Wenn die Anwendung bislang ohne Dokumentation existiert hat, genügt als erster Schritt ein einfaches aber dafür umso besser recherchiertes Handbuch im PDF-Format. Die User können es online verwenden während sie mit der Software arbeiten und bei Bedarf das Manual oder Teile davon ausdrucken. Weitere Medien können im Laufe der Zeit dazu kommen oder das vorhandene Medium ersetzen.

Umsetzung als User Manual

All diese Entscheidungen gilt es jetzt in Form einer Benutzerdokumentation zu Papier zu bringen, oder besser gesagt in ein geeignetes Textverarbeitungs- oder DTP-Programm. Auch hier gilt für die erste Dokumentation: möglichst vorhandene Ressourcen nutzen. Wird im Unternehmen Microsoft Word verwendet, schreibt man den User Guide in Word. Wenn Adobe FrameMaker bereits im Einsatz ist - wunderbar! Er eignet sich hervorragend für die Gestaltung umfangreicher Dokumentationen. Tools zur späteren Generierung der Online-Hilfe gibt es für beide. Wenn keines der beiden Produkte im Einsatz ist, empfiehlt sich ein Help Authoring Tool, das neben einem Texteditor auch mehrere Ausgabeformate, wie z.B. HTML-Help, Compiled Help und PDF unterstützt. So hat man bereits die Basis für spätere Erweiterungen der Benutzerdokumentation.

Wichtig ist jedoch nicht so sehr das Tool, sondern das Tun. Ab nun wird die Software als Eisberg betrachtet, von dem es gilt, die Oberfläche mit allen Einzelheiten zu beschreiben, damit ihn sowohl der unbedarfte Neuling als auch der Produktkenner mit den zur Verfügung stehenden Werkzeugen locker erklimmen kann. Für den klassischen Anwender ist bei einer Software nur interessant, was sich an der Oberfläche abspielt. Die darunter liegenden Funktionen, Datenbankanbindungen und Rechenoperationen mögen als Hintergrundinformationen interessant sein, haben aber für die produktive Handhabung einer Software nur eine untergeordnete Rolle. Daher beschränkt sich die Benutzerdokumentation auf die Erklärung der Abläufe, die der User mit der Software durchführen möchte. Das Interface, mit dem der Benutzer arbeitet, wird in allen Einzelheiten, Abläufen und möglichen Arbeitsschritten beschrieben. Die Beschreibungen sind an Handlungen orientiert und betonen Ergebnisse. Sie geben also Anleitungen, welche Tätigkeitsschritte notwendig sind, um ein gewünschtes Ergebnis zu erreichen.

Für den Aufbau von Benutzerdokumentation haben sich folgende Elemente als besonders benutzerfreundlich herausgestellt:

- *Screenshots*
- *Tabellen*
- *Kapitelgliederung*
- *Querverweise*
- *Schritt-für-Schritt-Anleitungen*
- *Index*

Durch die Verwendung mehrerer Techniken zur Darstellung von Information werden unterschiedliche Lesertypen angesprochen. Jeder User kann sich der von ihm bevorzugten Elemente bedienen, um zu den gewünschten Informationen zu gelangen.

3. Inhalt

Der Inhalt der Dokumentation hängt davon ab, welche Usergruppen während der Designphase identifiziert wurden. Demzufolge kann die Benutzerdokumentation auch aus mehreren Teilen bestehen, die sich jeweils an unterschiedliche Arten von Benutzern wenden. Jedes Handbuch sollte möglichst zu Beginn Hinweise darauf enthalten, wer die intendierte Benutzergruppe ist. Besteht die Dokumentation aus mehreren Teilen für unterschiedliche Anwender, sollten diese auch klar getrennt und als separate Dokumente erkennbar sein. Für Administratoren und Systemverantwortliche ist meist der Anwenderteil nicht relevant. Endbenutzer hingegen sind nicht sehr erfreut, wenn sie sich zuerst durch ein Administrationshandbuch durchblättern müssen, um dann in den Installation Guide zu gelangen, bevor sie endlich zum Teil mit den auf die Anwendung bezogenen Informationen gelangen.

Dennoch zählen alle diese Dokumentationen zur Benutzerdokumentation. Jedes dieser unterschiedlichen Handbücher richtet sich an ganz unterschiedliche „Anwender“ und verfolgt einen eigenen Zweck. Aus diesem Grund muss jedes Handbuch genau auf die definierte Benutzergruppe abgestimmt sein, und zwar in seinem Stil, seiner Gestaltung und seiner Beschreibungstechnik. Administration Guides enthalten technischen Background und sind daher entsprechend diesen Informationen technischer zu formulieren. Installationshandbücher kommen wahrscheinlich mit weniger Screenshots aus als der Anwenderteil. Sie enthalten jedoch möglicherweise mehr Diagramme und Tabellen zur Beschreibung von Abläufen und Zusammenhängen einzelner installierter Komponenten. User Guides für Endanwender enthalten viele Beispiele, um die Vorgangsweise der Software bei einer bestimmten Problemlösung darzustellen.

Aufbereitung der Informationen

Um den Inhalt des Anwenderteils benutzergerecht auszuwählen ist es wichtig, nicht nur die Software sehr gut zu kennen, sondern auch die User und deren Anforderungen an das Produkt. Hierbei hilft Input aus dem Support und dem Produktmanagement, da in diesen Bereichen meist Feedback von Usern zusammenläuft. Eine sinnvolle Strukturierung des Inhalts hängt auch stark von der Software selbst ab. Wenn die Anwendung im Wesentlichen eine Abfolge von Funktionen zur Verfügung stellt, die in einer bestimmten Reihenfolge im GUI realisiert sind und in dieser Reihenfolge auch vom Anwender genutzt werden, empfiehlt es sich, sich bei der Beschreibung an Abläufen zu orientieren. Meist führt in diesem Fall auch ein Anwender diese Tätigkeiten in gleicher oder ähnlicher Reihenfolge durch, wobei nur einige wesentliche Eingabeparameter geändert werden.

In anderen Fällen arbeiten mehrere User mit der gleichen Software, nutzen jedoch nur einzelne Funktionen der Software. Die Nutzung der Funktionen kann auch von Benutzerrechten abhängen, die zuvor vom Administrator vergeben werden. In diesem Fall ist eine am GUI orientierte Beschreibung vorzuziehen, da sich der Anwender dann jeweils den für ihn relevanten Abschnitt des Handbuchs heraussuchen kann.

Das wichtigste am Inhalt ist in jedem Fall dessen Aktualität. Ein Handbuch ist für einen Benutzer nur sinnvoll, wenn es die tatsächlich gelieferte Software beschreibt und nicht den Stand vom vorletzten Jahr. Aus diesem Grund muss die Benutzerdokumentation regelmäßig, also mit jeder neuen Release und/oder Subrelease, aktualisiert werden.

4. Umfang

In erster Linie wird der Umfang der gesamten Dokumentation natürlich durch die Software selbst bestimmt. Ein kleines Tool, mit dem man Screenshots von Bildelementen erstellen kann und dabei zwischen drei verschiedenen Einstellungen wählen kann, wird mit einem Handbuch von etwa 20 bis 30 Seiten auskommen. Dagegen können bei einem komplexen Bankensoftwareprodukt mit über 300 Dialogen schon mehr als tausend Seiten Dokumentation notwendig sein, um die Funktionalität der Software ausreichend zu beschreiben. Als Richtwert gelten in Fachkreisen etwa drei bis fünf Seiten pro Dialog oder vergleichbarer Bildelemente. Bei sehr komplexen Eingabemaschinen kann auch eine ausführlichere Beschreibung erforderlich sein.

Wahl der Informationsmenge

Die Gratwanderung besteht darin, den Umfang der Dokumentation so zu wählen, dass die Software vollständig und ausführlich beschrieben wird, aber dennoch übersichtlich bleibt. Ein Mehr an Informationen liefert zwar unter Umständen nützliches Hintergrundwissen. Man darf jedoch nicht vergessen, dass der Benutzer ein Manual in den meisten Fällen nicht zu seinem Vergnügen liest. Er verfolgt dabei einen bestimmten Zweck: Er möchte Informationen zu einem bestimmten Thema, damit er besser mit seiner Software arbeiten kann. Bei der Suche nach diesen Informationen verliert er sehr leicht die Geduld, wenn er sich zuerst durch zu viele Hintergrundinformationen durchlesen muss. Die Beschreibungen in einem User Manual sollten daher so stark wie möglich am Nutzen orientiert sein. Dabei sollten alle Bildelemente in ihrer Funktion so beschrieben sein, dass der User die Ergebnisse seiner Eingaben und Handlungen genau abschätzen kann.

Auswahl der Informationstiefe

Ein weiterer Aspekt spielt eine einflussreiche Rolle, wenn man sich für einen bestimmten Umfang der Dokumentation entscheiden soll: Es gibt User, die zum ersten Mal mit dieser Software arbeiten und andere, die bereits jahrelange Erfahrung mit der Software haben. Wie deckt man die Bedürfnisse beider Gruppen mit einem User Guide ab? Hier helfen die Informationen, die man in der Designphase als Benutzerprofil erarbeitet hat, ebenso wie Gespräche mit Supportmitarbeitern des eigenen Unternehmens. Grundsätzlich gilt: *Alles beschreiben, was sinnvoll ist, weglassen, was nicht unbedingt nötig ist.*

Viele Anfänger plagt neben ihrer Unkenntnis der Software auch ihr Mangel an fachlichem Hintergrundwissen. Dafür wenden sie sich dann ohnehin an einen erfahrenen Kollegen, der ihnen auch gleichzeitig erklärt, wie man den Arbeitsschritt mit der Software durchführt. *Im Allgemeinen ist es daher nicht erforderlich, den Business Background zu erklären*, also den fachlichen Zweck, bei dem die Software den User unterstützt.

Ausmaß der Hintergrundinformationen

In den meisten Fällen ist es nicht notwendig, den technischen Hintergrund der Softwareentwicklung zu beschreiben. Die Tiefe der programmiertechnischen Informationen, die ein User Guide enthält, hängt von der definierten Benutzergruppe ab. Technische Zusammenhänge, wie die Übergabe von Daten an eine Datenbank, die Umwandlung von Informationen in ein internes Verarbeitungsformat oder genaue Bezeichnungen von Routinen und Programmschritten, sind bis zu einem gewissen Grad für technische Anwender wie Datenbankadministratoren und Systemverantwortliche relevant. Diese Informationen sollten daher in einem separaten Teil des Handbuchs, nämlich in einem Administratorenhandbuch oder einem Installation Guide dokumentiert werden. Für die reine Bedienung einer Software sind tiefer gehende technische Informationen in den meisten Fällen irrelevant. Sie strapazieren nur die Geduld des Users bei der Suche nach den für ihn relevanten Informationen.

Ein vollständiges Handbuch enthält allerdings zumindest eine allgemeine Einführung in die Software und einen Überblick über den grundsätzlichen Zweck der Anwendung. Diese Einführung beschreibt den Nutzen und die Vorteile der Software für den Anwender. Sie kann auch einen kurzen technischen Überblick über den programmiertechnischen Aufbau der Anwendung enthalten. Je nach Benutzerprofil ist dieser Teil zwischen einer halben Seite und einem gesamten Kapitel von mehreren Seiten lang.

5. Medien

Bei der Entscheidung, in welcher Form dem Benutzer die Beschreibung der Software zur Verfügung gestellt werden soll, spielen verschiedene Faktoren eine Rolle. Einerseits hängt es natürlich von der Software selbst und der Entwicklungsplattform ab, ob zum Beispiel eine Online-Hilfe möglich und sinnvoll ist. Auch wenn ein Dokumentationsmedium möglich wäre, bedeutet das nicht unbedingt, dass es auch für die Benutzer sinnvoll und praktisch ist. Online Manuals sind beispielsweise nur sinnvoll, wenn der Benutzer auch Zugang zum Internet bzw. einen Browser installiert hat. Das ist in Unternehmen, die mit sensiblen Kundendaten umgehen, nicht immer der Fall. Hier wirken sich außerdem auch budgetäre Überlegungen aus, da jedes zusätzliche Medium auch unterschiedliche weitere Kosten verursachen kann.

In der Folge geben wir einen kurzen Überblick über mögliche Medien und deren Charakteristika, sowie einige Überlegungen, was bei den einzelnen Dokumentationsformaten jeweils zu bedenken ist.

Online-Dokumentation

Die Online-Hilfe ist meist die erste Antwort, wenn man die Frage nach der Benutzerdokumentation stellt. Sie ist vor allem bei Windows-basierten Produkten bereits als Standard etabliert und man geht davon aus, dass User eine Online Help beim Produkt erwarten. Bei der Online-Hilfe gibt es einerseits verschiedene Formate, andererseits verschiedene Ausprägungen, die jeweils andere Techniken erfordern, um sie zu erstellen.

• *HTML-Help (Compiled)*

Die compilierte HTML-Hilfe liefert als Output .CHM Dateien, die über die Hilfefunktion der Windows-basierten Applikation aufgerufen werden. Die Hilfe kann als Standalone Help (nur F1), als Context Sensitive Dialog Help (F1 + Help Button/F1 in Dialogen) oder als voll kontextsensitive Hilfe (F1 + Help Button/F1 in Dialogen + ?-Button für Feldbezogene Hilfe in Dialogen) konzipiert sein.

Eine compilierte Online-Hilfe erfordert sorgfältige Planung vor der Erstellung. Man benötigt Vorbereitungen in der Software selbst während der Entwicklung für die Anbindung des Hilfesystems und nach der Fertigstellung gründliche Tests vor der Auslieferung der Software. Jeder dieser Schritte erfordert Zeit und personelle Ressourcen für Entwicklung und Dokumentation. Die compilierte HTML-Hilfe steht für Windows-Plattformen zur Verfügung und erfordert für die Lauffähigkeit beim Benutzer die Installation von Browserkomponenten.

Für die Erstellung einer compilierten Hilfe werden Help Compiling Tools benötigt.

Bei Updates, neuen Releases oder Bugfixes ist die Erstellung und Einbindung einer neuen Hilfeversion samt dazugehörigen Funktionstest erforderlich.

- *HTML-Manual (Browser-based)*

Die plattformunabhängigere HTML-Hilfe besteht aus .HTM oder .HTML Dateien, die über die Hilfefunktion der Windows-basierten Anwendung oder unabhängig davon aufgerufen werden. Sie kann als kontextsensitive Dialoghilfe gestaltet werden (abhängig von der Software), steht jedoch meist als Standalone Help zur Verfügung. Sie ist als Onlineversion des Handbuchs zu verstehen und wird daher oft als Ersatz für eine gedruckte Dokumentation mit der Software mitgeliefert.

Für ein HTML-Manual werden nicht unbedingt spezielle Tools benötigt. Die Erstellung kann jedoch in diesem Fall gewisse Nachbearbeitungen erfordern, um ein Inhaltsverzeichnis und einen Suchindex zu generieren.

Bei Updates, neuen Releases oder Bugfixes ist die Erstellung und Einbindung einer neuen Hilfeversion samt dazugehörigen Funktionstest erforderlich.

- *Embedded Help*

Diese Form der Hilfe ist direkt in die Software integriert, beispielsweise in speziell dafür vorgesehenen Bildschirmbereichen. Die Erstellung der Hilfe erfolgt über spezielle Editoren, direkt in den Resource Files der Entwickler oder durch Einlesen der Hilfetexte aus externen Dateien während der Softwaregenerierung. Die Embedded Help wird vor allem bei Web Applications, teilweise auch bei Dialogbasierten Anwendungen verwendet. Manche Host-Anwendungen bieten ebenfalls die Möglichkeit, in eigenen Schirmen Hilfetexte zu integrieren.

Der für die Hilfe zur Verfügung stehende Platz ist jedoch meist stark limitiert. Aus diesem Grund kann eine Ergänzung der Embedded Help durch eine ausführlichere Online-Hilfe oder ein PDF-Manual zum besseren Verständnis der Software notwendig sein.

Eine enge und koordinierte Zusammenarbeit zwischen Entwicklung und Dokumentationsverantwortlichem ist unabdingbar, da die Hilfe einen integrierten Bestandteil des Systems darstellt.

Bei der Embedded Help wird die Anwendung bei jedem Update oder Bugfix mit der jeweils aktuellen Hilfeversion generiert.

Gedruckte Dokumentation

Im Gegensatz zur Online-Dokumentation ist die gedruckte Dokumentation nicht an die Software angebunden. Es besteht meist keine Möglichkeit aus der Anwendung direkt in die Dokumentation zu gelangen. Daraus ergibt sich auch, dass sie meist in der Herstellung einfacher und kostengünstiger ist.

- *PDF-Manual*: Obwohl diese Art der Dokumentation aufgrund ihres Formats als PDF-Datei auch online zur Verfügung steht, zählt sie zu den Printversionen. Der Grund dafür ist, dass für ein PDF-Manual zusätzliche Fertigungsschritte notwendig sind, die sonst nur bei Printversionen anfallen. Dazu zählen Layoutüberlegungen, Seitengestaltung, Seitenumbruch, Screenshotdesign und -anfertigung sowie Covergestaltung. PDFs erfordern auch den Aufbau und die Gestaltung spezieller Navigationsmöglichkeiten.

Das PDF-Manual hat heute das klassische gedruckte Handbuch weitgehend ersetzt. Es wird meist gemeinsam mit der Software auf der Installations-CD ausgeliefert. Ein PDF-Manual steht dem Benutzer während der Arbeit mit der Software am Bildschirm zur Verfügung. Er kann sich jedoch auch, wenn er das vorzieht, Teile daraus oder das gesamte Manual selbst ausdrucken. Für die Verwendung eines PDF-Manuals braucht der Benutzer den Acrobat Reader, den er (bei vorhandenem Internetzugang) selbst downloaden kann oder der ihm auf der Installations-CD mitgeliefert werden kann.

- *Printversion*: Die Dokumentation kann auch als fertig gedrucktes Handbuch mit der Software mitgeliefert werden. Hierbei können zusätzliche externe Kosten für Satz, Druck, Bindung und Umschlaggestaltung anfallen. Darüber hinaus entstehen bei Lieferung höhere Versand- und Lagerkosten. Das gedruckte Handbuch kann jeder Benutzer ohne zusätzliche Tools oder Hilfsmittel sofort verwenden.

Wenn mehrere Medien zur Verfügung gestellt werden sollen, muss der Wartbarkeit der Dokumentation zusätzliche Aufmerksamkeit geschenkt werden. In solchen Fällen ist die Methode des Single Sourcing empfehlenswert. Hierbei wird ein Quelldokument erstellt und daraus mehrere unterschiedliche Outputs generiert. Mögliche Varianten sind:

- Word-Dokument Source -> Online-Hilfe + PDF-Manual
- FrameMaker Source -> HTML-Help + PDF-Manual + Printversion
- Embedded Help -> Embedded Help + PDF-Manual + Printversion

Nahezu beliebige Kombinationen sind vorstellbar und möglich. Was wirklich für den Benutzer und die Software sinnvoll ist, erfordert sorgfältige Überlegung und entsprechende Planung von Zeit und Ressourcen.

6. Zeitplan

Dokumentation von Software braucht Zeit. Daher sollte mit der Konzeption, Planung und Erstellung der Dokumentation so früh wie möglich im Entwicklungsprozess begonnen werden. Nur dadurch kann die Dokumentation gleichzeitig mit der Fertigstellung einer Software Release ausgeliefert werden. Wenn die Arbeit am Handbuch bereits anfängt, während die Software noch in der Entwicklungsphase ist, können die für den Benutzer potenziell unverständlichen Abläufe oder komplizierte Vorgänge rechtzeitig hinterfragt und optimiert werden. In jedem Fall empfiehlt sich eine regelmäßige und enge Zusammenarbeit zwischen Entwicklung, Produktmanagement und Dokumentationsverantwortlichem. Abstimmungen zwischen diesen drei Instanzen erleichtern die Erstellung einer benutzerfreundlichen Beschreibung der Software. Denn nur wenn man den Benutzer und seine Anforderungen und Bedürfnisse kennt, kann man die Software so beschreiben, dass der User sie optimal nutzen kann.

Ergänzend zu diesem Text steht Ihnen auf der InfraSoft-Website ein [Glossar](#) zur Verfügung, in dem Sie die meisten der hier verwendeten Begriffe finden. Über das [aktuelle Angebot an weiteren, kostenlosen Fachbeiträgen zur Softwareentwicklung](#) informieren Sie sich bitte unter www.infrasoft.at/service.

Mag. Karoline Mrazek
Wien, im Februar 2004

© InfraSoft GmbH www.infrasoft.at
© Mrazek Dokuconsult www.dokuconsult.at

Die Autorin dieses Beitrags kooperiert mit InfraSoft, einem Unternehmen, das auf komplexe Softwareentwicklungen spezialisiert ist. Die Experten der InfraSoft haben langjährige Erfahrungen in der Entwicklung und verfügen über fundierte Kenntnisse in Design, Analyse, Realisierung, Test und Projektmanagement.

Für **individuelle Beratungen** zur Entwicklung von Softwarelösungen und die Bereitstellung von **Realisierungsteams** wenden Sie sich bitte an info@infrasoft.at.