

Konzeption und Implementierung der nutzerzentrierten mobilen Flugnavigationsapplikation „VFR-2-SKY“

Gernot KLINGER¹ und Markus FLATSCHER²

¹Austro Control GmbH, Wien · gernot.klinger@austrocontrol.at

²Universität Salzburg

Dieser Beitrag wurde durch das Programmkomitee als „reviewed paper“ angenommen.

Zusammenfassung

Ziel des Projektes ist die Konzeption einer Flugnavigationsapplikation, die in der Lage sein muss, offizielle Luftfahrtkarten gemäß ICAO Annex 4 darzustellen, sowie die Implementierung als Prototyp für Piloten, die sich unter Sichtflugbedingungen am Boden und in der Luft bewegen. Der Fokus der gesamten Entwicklung liegt dabei stets auf Nutzerzentriertheit. Vor dem Start des Projekts wurden Mindestanforderungen definiert, die in einer späteren Evaluierung durch Privatpiloten nach deren Wichtigkeit gereiht wurden. Zusätzlich diente diese Umfrage dazu, Daten zu gewinnen, die nachfolgend für die Erstellung von Personas und Szenarios verwendet wurden. Diese stellen wiederum einen wesentlichen Bestandteil des Designprozesses dar.

Für die Implementierung eines Prototyps ist zuerst die Aufbereitung der Luftfahrtkarten notwendig, um diese für Smartphones und Tablets „lesbar“ zu machen. Anschließend kann die Applikation mithilfe einer Entwicklungsumgebung (in diesem Fall Eclipse) erstellt werden. In einem zweistufigen Testverfahren, einerseits mit Pkw, andererseits bei einem Testflug unter Realbedingungen, wurde schließlich der Nachweis erbracht, dass der Prototyp seinen zu Beginn definierten Ansprüchen gerecht wird.

1 Einleitung

Ziel des Projektes ist die Entwicklung eines funktionsfähigen Prototyps einer mobilen Flugnavigationsapplikation für das Android-Betriebssystem basierend auf bestehenden Geodaten der Flugsicherung unter intensiver Einbeziehung der Pilotenschaft. Die Entscheidung zugunsten dieses Betriebssystems fiel zum einen aufgrund der Tatsache, dass Google bereits einen weltweiten Marktanteil von über 80 % für sein Android-Betriebssystem beansprucht, womit eine breite potenzielle Kundenschicht erreicht werden kann und ist zum anderen darin begründet, dass beim Programmieren von Android Apps keine Kosten, wie etwa durch anfallende Lizenzgebühren, anfallen (IDC CORPORATE USA 2014). Zum Zeitpunkt des Projektstarts war zudem keine Flugnavigationsapplikation verfügbar, die die offiziellen österreichischen Luftfahrtkarten in ihrem Sortiment inkludiert hat.

Das Projekt erhielt den Namen „VFR-2-SKY“ in Anlehnung an die primäre Kundschaft der Piloten, die unter Sichtflugbedingungen navigieren (im englischen „*Visual Flight Rules*“ – VFR). Im Folgenden wird die Herangehensweise bei der Entwicklung dieser Applikation beschrieben, wobei der Fokus auf die Nutzerzentriertheit gelegt wird, was mithilfe eines Fragebogens, der unter Privatpiloten verteilt wurde, sowie Interviews, Personas und Szenarios gewährleistet wird. Die technische Umsetzung der Applikation steht im vorliegenden Paper nicht im Vordergrund, dennoch werden die wichtigsten Herausforderungen skizziert. Ein abschließendes Kapitel widmet sich dem Testen der Applikation und den Ergebnissen, die aus den Testläufen gewonnen wurden.

2 Anforderungen an die Applikation

Zu Beginn wurden folgende Mindestanforderungen definiert, die in der Applikation „VFR-2-SKY“ umgesetzt werden sollen:

- Offline-Laden von georeferenzierten Flugnavigationskarten in verschiedenen Maßstäben (z. B. Luftfahrtkarte ICAO 1:500.000).
- Implementierung von Zoom- und Pan-Funktion.
- Abrufen und Darstellen der eigenen Position auf der Karte mithilfe des geräteinternen GPS-Empfängers.
- Ableitung und Darstellung von Flughöhe, Geschwindigkeit und Kursrichtung.
- Visualisierung aller österreichischen Flugplätze.
- Dokumentation des Quellcodes.

Basierend auf diesen Mindestanforderungen wurden außerdem, aus Gründen der Performancesteigerung und Usability folgende optionale Anforderungen definiert:

- Kachelung der Karte und Anlegen von Bildpyramiden.
- Implementierung einer Trackingfunktion und Exportmöglichkeit von aufgezeichneten Tracks als .gpx- oder .kml-Datei.
- Umsetzung von aufrufbaren Flugplatzinformationen.
- Automatischer Wechsel zwischen den aeronautischen Basiskarten in Abhängigkeit von Zoomlevel und Flugplatznähe.

Zielgruppe der Applikation sind Privatpiloten, die unter Sichtflugbedingungen fliegen und diese App zur Flugvorbereitung, sowie als zusätzliche Orientierungshilfe auch während des Fluges verwenden. Ein wichtiges Anliegen war es dabei, die Applikation für die Verwendung auf größeren Displays zu optimieren, da davon ausgegangen werden kann, dass die Kartendarstellung sowie die Auffassung des Karteninhalts auf Tablets besser zur Geltung kommt, als auf kleineren Displays von Smartphones. Als „größere Displays“ werden hierbei Bildschirme betrachtet, die eine Diagonale von mindestens 8 Zoll (ca. 20 cm) aufweisen.

3 Herangehensweise

Abbildung 1 zeigt den generellen Entwicklungsprozess einer Software mit Fokus auf Nutzerzentriertheit. Dieser „*user-centered approach*“ findet auch in der Applikation „VFR-2-

SKY“ Anwendung. Die Frage des Funktionsumfangs der App steht an erster Stelle des Entwicklungsprozesses und wurde in diesem Fall bereits durch die Definition der Mindest- und optionalen Anforderungen beantwortet. Um darüber hinaus einen Eindruck davon zu gewinnen, welche Aspekte einer Flugnavigationssapplikation für die Piloten wirklich am wichtigsten sind, wurde ein Fragebogen konzipiert, der auf diese Fragestellung abzielt. Der Fragebogen diente dazu, den Schwerpunkt der Entwicklung auf diejenigen Aspekte zu legen, die auch tatsächlich von den Piloten gewünscht werden. Dabei wurden von den befragten Piloten vor allem die korrekte Darstellung der Luftraumstruktur, die Implementierung von Flugplatzinformationen und die Visualisierung der Flugparameter als besonders wichtig genannt.

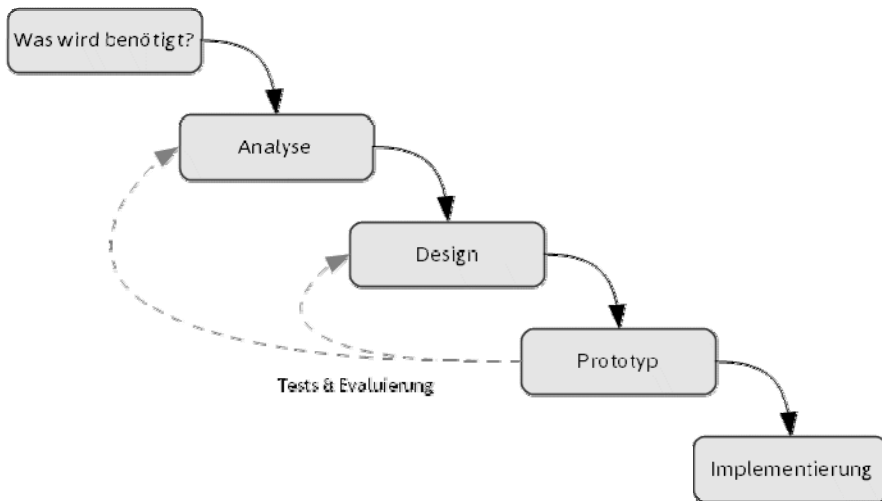


Abb. 1: Design Prozess (eigene Darstellung nach DIX et al. 2004)

Zusätzlich konnten durch den Fragebogen persönliche Daten, wie Alter, Qualifikation oder die Anzahl der jährlich erworbenen Flugstunden erfasst werden. Diese Daten wurden später für die Erstellung von Personas und Szenarios verwendet. Zusätzlich wurden die Piloten auch dazu angehalten, aus mehreren Entwürfen einer Benutzeroberfläche das ihrer Meinung nach Beste auszuwählen.

4 Personas


Personas gelten als aussagekräftiges Werkzeug bei der Entwicklung von Produkten oder Software. Sie repräsentieren fiktive Personen, die in irgendeiner Art und Weise mit dem zu entwickelnden Produkt in Verbindung stehen. Jedoch sind Personas nicht einfach erfundene Persönlichkeiten. Vielmehr werden etwa aus Umfragen und/oder Nachforschung gewonnenen Daten als Grundlage dazu verwendet, Benutzergruppen zu identifizieren, welche anschließend durch Personas repräsentiert werden. COOPER et al. (2007) definieren folgende Charakteristika für Personas:

- sie fußen auf (Nach-)Forschung,
- sie werden als Individuen dargestellt,
- sie repräsentieren User-Gruppen,
- sie haben Motivationen.

Nachfolgende Abbildung zeigt exemplarisch die fiktive Person „Karl“, die beim Design von „VFR-2-SKY“ als zentrales Persona dient.

Karl

technology-affine pilot rookie



Demographics

Age: 26
 Gender: Male
 Occupation: Architectural Draftsman
 Education: College
 Hobbies: Sport, Traveling, Party
 Location: Innsbruck, Austria

Devices & usage

Smartphone: Apple iOS & Android
 Tablet: Apple iOS & Android

Usage:

- always online
- uses many functions of his devices
- uses a lot of apps regularly

Experience

Pilot licenses: PPL
 Since: 2010
 Flight hours p.a.: about 15
 Flying area: Austria

Abb. 2: Beispielhaftes Persona „Karl“ für das Projekt „VFR-2-SKY“

„Karl“ ist ein technikbegeisterter Neuling als Pilot und verwendet stets die neuesten Smartphones und Tablets. Als Pilot hat er noch keine große Erfahrung, da er seine Pilotenlizenz erst 2010 erworben hat und nur etwa 15 Flugstunden pro Jahr sammelt. Deswegen verlässt er sich auf „VFR-2-SKY“ als zusätzliche Orientierungshilfe im dreidimensionalen Luftraum. Die Verwendung einer Flugnavigationsapplikation verleiht ihm ein Gefühl von Sicherheit.

5 Szenarios

Szenarios stellen eine weitere Stufe in der Entwicklung von „VFR-2-SKY“ dar. In Szenarios werden verschiedene Alltagssituationen dargestellt, in denen die vorher erstellten Per-

sonas verschiedenen Herausforderungen gegenüberstehen oder bestimmte Ziele verfolgen (COOPER et al. 2007). Ziel von Szenarios ist es, konkrete Userziele und Anforderungen abzuleiten, bzw. vordefinierte Anforderungen zu konkretisieren. Für das beispielhafte Persona „Karl“ wurde folgendes Szenario entworfen:

Karl ist als Pilot natürlich daran gewöhnt und trainiert, mithilfe von Karten und den Flugzeuginstrumenten zu navigieren. Da er sehr technikbegeistert ist, hat er zusätzlich eine neue Flugnavigationsapplikation auf seinem Tablet installiert.

Vor jedem Flug benutzt er diese Applikation, um eine Flugroute zu planen, diese zu speichern und später auf einer Luftfahrtkarte darzustellen. Nach Aktivierung des geräteinternen GPS-Empfängers, kann das Tablet die eigene Position abrufen und mithilfe eines Flugzeug-Symbols auf dem Display darstellen.

Kurz vor dem Start aktiviert Karl die implementierte Tracking-Funktion der Applikation, um seinen Flug aufzuzeichnen. Diesen aufgezeichneten Flug kann er später in unterschiedliche Dateiformate exportieren und z. B. mit anderen Piloten austauschen.

Beim Anflug auf einen Flugplatz verwendet Karl die integrierten Sichtflugkarten, die einen relativ großen Maßstab aufweisen und nur das Gebiet rund um einen Flugplatz darstellen.

Obwohl er immer analoges Kartenmaterial mit sich führt, fühlt sich Karl sehr wohl bei der Verwendung dieser Flugnavigationsapplikation, die ihm ein Gefühl der Sicherheit verleiht.

Dieses Szenario zeigt beispielhaft, wie dieses Designtool dazu beitragen kann, dass ein Designer oder Programmierer ein besseres Verständnis für das zu entwickelnde Produkt erhält. Wie auch bei den Personas gilt auch für Szenarios der Grundsatz, dass diese nicht frei erfunden werden, sondern auf Daten basieren sollen, die durch Interviews und Nachforschung gewonnen wurden. Dies soll Realitätsnähe gewährleisten und verhindern, dass ein Produkt entwickelt wird, das die Anforderungen des Zielpublikums nicht kennt bzw. berücksichtigt.

6 Datenaufbereitung und Implementierung

Aus den in Fragebögen, persönlichen Interviews, Personas und Szenarios gewonnenen Erkenntnissen, kann anschließend ein Design-Framework erstellt werden, das als Gerüst der Applikationsentwicklung gesehen wird. In einem Design-Framework werden erste Entwürfe eines User-Interfaces erstellt und Funktionen der Applikation konkretisiert sowie Ideen gesammelt, wie diese Funktionen umgesetzt werden sollen und an welcher Stelle Buttons oder Icons platziert werden.

Diese Erkenntnisse dienen nun dazu, eine Applikation entwickeln zu können, die sowohl den Ansprüchen und Wünschen der Piloten, als auch denen der Flugsicherung gerecht wird. Basierend auf der Entscheidung zugunsten des Android-Betriebssystems, wurde als Entwicklungsumgebung die Eclipse IDE gewählt. Das von Google entwickelte Android Software Development Kit (SDK) enthält alle relevanten Java-Bibliotheken, die zur Entwick-

lung von Android Apps benötigt werden. Schließlich kann mit dem Google ADT-Plug-in eine „Brücke“ zwischen Android SDK und Eclipse hergestellt werden, um alle Android-relevanten Funktionen und Bibliotheken direkt in Eclipse verwenden zu können (BLESKE 2012).

Um den Anforderungen der Offline-Darstellung der österreichischen Luftfahrtkarten gerecht zu werden und den Maßstab der Karte auf dem Smartphone oder Tablet verändern zu können, ist es nötig, die im GeoTIFF-Format bereitgestellten Karten weiter zu bearbeiten. Zu diesem Zweck wurde die Software „TileMill“ des Herstellers MapBox verwendet (MAPBOX 2014a). Diese ermöglicht es, Karten in ein für mobile Geräte lesbares .mbtiles-Format umzuwandeln. Dies kann gezielt für bestimmte Zoomlevels bewerkstelligt werden. Allerdings tritt die Einschränkung auf, dass lediglich die Google Mercator Projektion unterstützt wird, was eine Verzerrung der ursprünglichen Karte bedingt, da österreichische Luftfahrtkarten eine international vorgeschriebene konforme Kegelprojektion verwenden (INTERNATIONAL CIVIL AVIATION ORGANIZATION 2009).



Abb. 3: Auswirkungen der Umprojizierung

Anhand der Luftfahrtkarte – ICAO 1:500.000 zeigt Abbildung 3, wie bei der Umprojizierung von einer Kegel- zu einer Zylinderprojektion ein „Ziehharmonikaeffekt“ auftritt. Um die aufbereitete Karte anschließend auf einem Smartphone oder Tablet darstellen zu können, ist die Einbindung einer Mapping Application Programming Interface (API) nötig.

Da die im Android SDK bereits integrierte Google Maps API nicht den erforderlichen Funktionsumfang aufweist, greift „VFR-2-SKY“ auf die Altus Mapping Engine des US-Herstellers BA3 zurück (BA3 LLC 2014). Für das Android-Betriebssystem wurde diese API erst im Juli 2013 veröffentlicht und bietet alle nötigen Funktionen zum Laden von

Basemaps, Karten-Overlays oder Markern. Das ausschlaggebende Argument zugunsten dieser API ist allerdings die Tatsache, dass ein sphärisches Display verwendet wird und sämtliche Overlays auf einem virtuellen Globus dargestellt werden und nicht auf einem planaren Display, wie bei herkömmlichen APIs.

Die in TileMill aufbereiteten Karten können auf diese Weise dargestellt werden, ohne dass durch Projektionen bedingte Verzerrungen auftreten. Insbesondere für die Navigation und die Orientierung der Piloten ist dies essenziell, da weder Winkel, noch Distanzen oder Flächen verzerrt dargestellt werden.

Abbildung 4 zeigt einen Ausschnitt des aktuellen User-Interfaces von „VFR-2-SKY“, das aus den bereits beschriebenen Anforderungen abgeleitet wurde. Insbesondere wurde Wert darauf gelegt, den Piloten in der geräuschintensiven und unruhigen Umgebung eines Flugzeugcockpits eine Designlösung zu bieten, die sowohl alle nötigen Funktionen unterstützt, als auch einfach in der Handhabe ist. So kann z. B. jede Funktion der App mit maximal zwei Fingertipps angesteuert und bedient werden. Außerdem wurde auf die Implementierung von verschachtelten und komplexen Menüs verzichtet.



Abb. 4: Benutzeroberfläche von „VFR-2-SKY“

Die Programmlogik im Hintergrund von „VFR-2-SKY“ zielt darauf ab, dass allgemeine Einstellungen, wie z. B. die Zeitintervalle für Standortupdates, bereits vor dem Flug getroffen werden sollen. Dies reduziert zwar in gewissem Maße die Flexibilität, verringert allerdings im Gegenzug auch die Komplexität und erhöht damit letztendlich die Nutzerfreundlichkeit.

Besonders erwähnenswert ist die Tatsache, dass die Altus Mapping Engine das Verwenden von Basemaps erheblich erleichtert. In diesem Projekt wurde eine Open Street Map (OSM) als Basiskarte verwendet. Alle in TileMill erstellten .mbtiles-Dateien können anschließend als Overlays über die OSM-Karte gelegt werden. Eine .mbtiles-Datei besteht im Wesentlichen aus einer SQLite Datenbank, in der alle erstellten Kartenfliesen (Maptiles) nach Zoom-Levels organisiert und gespeichert werden (MAPBOX 2014B).

Durch die Implementierung zweier zusätzlicher SQLite-Datenbanken konnten außerdem zum einen die Darstellung aller österreichischer Flugplätze und zusätzlich aller österreichischer Navigationshilfen sowie zum anderen das Speichern von aufgezeichneten Tracks umgesetzt werden.

Die erste der beiden Datenbanken wird bereits im Vorhinein mit Daten befüllt und dient der Darstellung von Markern (Flugplätze und Navigationshilfen). Per Fingertipp auf einen Marker können zusätzliche Informationen, wie z. B. die Beschaffenheit einer Landebahn oder die Frequenz eines Drehfunkfeuers eingesehen werden.

Die zweite Datenbank wird erst zur Laufzeit der Applikation mit Daten befüllt, sobald ein User das Kamera-Symbol der Menüleiste drückt und damit das Tracking aktiviert (siehe Abb. 4). Nun werden alle Wegpunkte in die Datenbank geschrieben. Ein erneuter Fingertipp auf das Kamera-Symbol beendet das Tracking schließlich wieder. In einer separaten Kartenansicht können alle gespeicherten Tracks anschließend geladen, eingesehen und als .kml- oder .gpx-Datei exportiert werden. Abbildung 4 zeigt außerdem drei graue Textfelder am unteren Rand des Displays. Diese Textfelder werden dazu verwendet, die aktuellen Flugparameter – Flughöhe, Geschwindigkeit und Kursrichtung – anzuzeigen.

In diesem Entwicklungsstadium konnte schließlich mit dem Testen der Applikation begonnen werden.

7 Testergebnisse

Um sicher zu stellen, dass alle Komponenten der Applikation reibungslos zusammenarbeiten, ist es unumgänglich diverse Tests durchzuführen. Dies wurde zum einen per Pkw durchgeführt, um die Funktionalität von „VFR-2-SKY“ testen zu können und zum anderen bei einem Testflug unter Realbedingungen.

Während des Testlaufs per Pkw erbrachte der Prototyp dabei den Nachweis, dass alle Funktionen und Komponenten der Applikation reibungslos funktionieren. Auch das Aufzeichnen eines Tracks funktionierte dabei einwandfrei, wobei auch Signalunterbrechungen durch Tunnelfahrten simuliert werden konnten. Dies war die Voraussetzung für einen Testflug, der anschließend durchgeführt wurde.

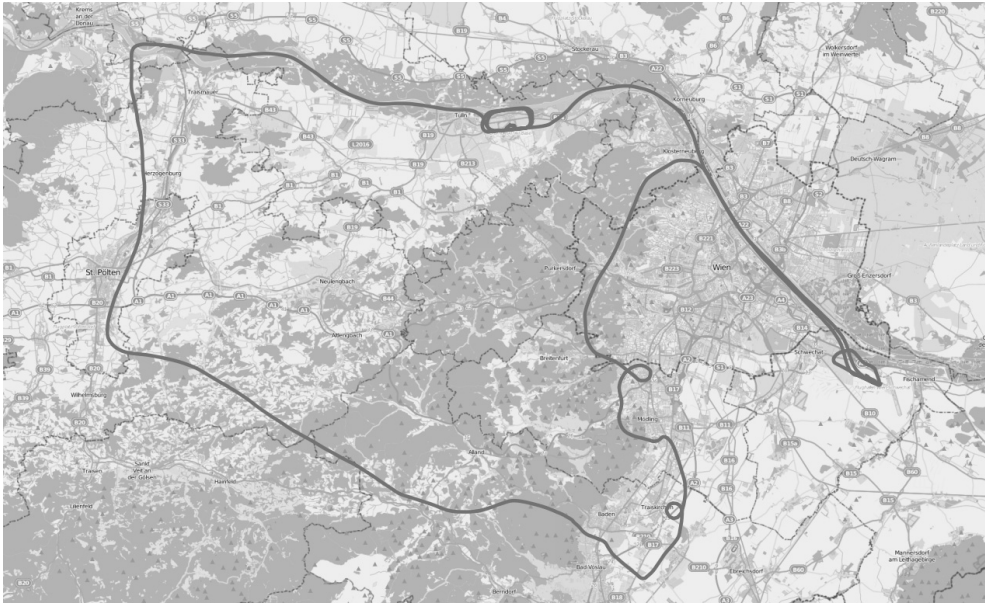


Abb. 5: Testflugstrecke

Ziel dieses Testfluges war es, die Applikation unter Realbedingungen im dreidimensionalen Raum zu testen. Dazu wurden in der Luftfahrt typische Trainingselemente durchgeführt, die das Verhalten und Reaktionsvermögen der Applikation an ihre Grenzen bringen sollten, wie etwa „touch and go“-Anflüge, abrupte Höhenveränderungen oder enge Kurven.

Ein erwähnenswertes Ergebnis ist hierbei, dass es eine Diskrepanz zwischen den vom Smartphone oder Tablet ermittelten GPS-Höhenwerten und den im Flugzeug ermittelten Höhenwerten gibt. Diese betragen beim Testflug bis zu 160 Fuß (ca. 57 m). Die Instrumente in einem Flugzeug sind fein kalibriert und arbeiten höchst präzise, während die GPS-Werte aus der Zeitspanne ermittelt werden, die ein Signal zwischen einem GPS-Satelliten und der Erdoberfläche zurücklegt. Dies führt zu den erwähnten Diskrepanzen. Wie vonseiten der Piloten allerdings kommuniziert wurde, ist dieses Problem nicht gravierend, da eine Flugnavigationsapplikation ohnehin vorwiegend zur lateralen Navigation verwendet wird und bei Bodenannäherung auf Sicht und nicht nach Instrumenten geflogen wird bzw. werden darf.

8 Fazit und Ausblick

Als zentrales Ergebnis der Entwicklung der Applikation „VFR-2-SKY“ kann festgehalten werden, dass es für ein erfolgreiches Design unerlässlich ist, die potenziellen Nutzergruppen von Anfang an in den Designprozess einzubinden. Dies stellt sicher, dass die Applikation den Ansprüchen und Anforderungen der Nutzer gerecht wird, und sichert somit den kommerziellen Erfolg des Produktes.

Die Tests von „VFR-2-SKY“ haben den Beweis erbracht, dass alle geforderten Mindestanforderungen umgesetzt werden konnten und ordnungsgemäß funktionieren. Allerdings gibt es auch Aspekte, die in der aktuellen Version von „VFR-2-SKY“ noch nicht umgesetzt werden konnten, für zukünftige Versionen allerdings nutzerrelevant sind.

Dies betrifft z. B. die Bereitstellung von aktuellen Wetterinformationen und die automatische Darstellung jener Kontrollstellen, in deren Verantwortungsbereich sich das Luftfahrzeug aktuell befindet. Diese Erweiterungen sollen den Piloten zusätzliche Übersicht und Komfort ermöglichen und in zukünftigen Versionen von „VFR-2-SKY“ implementiert werden.

Literatur

- BA3 LLC (2014), Altus Mapping Engine. <http://www.ba3.us/?page=pages/mapping-engine> (18.04.2014).
- BLESKE, C. (2012), Java für Android. Native Android Apps programmieren. Franzis, Haar bei München.
- COOPER, A., REIMANN, R. & CRONIN, D. (2007), About Face 3. The Essentials of Interaction Design. Wiley Publishing Inc.
- DIX, A., FINLAY, J., ABOWD, G. & BEALE, R. (2004), Human-Computer Interaction. Third Edition. Pearson education Ltd.
- IDC CORPORATE USA (2014), Android Pushes Past 80% Market Share While Windows Phone Shipments Leap 156,0% Year Over Year in the Third Quarter, According to IDC. <http://www.idc.com/getdoc.jsp?containerId=prUS24442013> (10.01.2014).
- INTERNATIONAL CIVIL AVIATION ORGANIZATION (2009), Annex 4 to the Convention on International Aviation. Aeronautical Charts. Eleventh Edition. Montreal.
- MAPBOX (2014a), TileMill. Create beautiful interactive maps. <https://www.mapbox.com/tilemill/> (18.04.2014).
- MAPBOX (2014b), TileMill. Storing Tiles. <https://www.mapbox.com/developers/mbtiles/> (18.04.2014).