

Berechnungsmodule für die Grundsteuerreform – Eine Blaupause für die Zusammenarbeit von Steuerfachexperten und Softwareentwicklern

Text — Dr. Rainer Bräutigam, Dr. Grigoriy Volovskiy, Steffen Kirchhoff

1. Einleitung

Rückblickend auf das Jahr 2022 ist festzustellen, dass das Thema Grundsteuerreform sicherlich eine der wesentlichen, wenn nicht die Herausforderung für den Berufsstand der Steuerberater war. Angefangen von der ursprünglich eng bemessenen Frist von nur vier Monaten für die Abgabe von 36 Millionen Erklärungen über spät veröffentlichte Anwendungserlasse bis hin zu technischen Problemen verging kaum ein Monat, in dem die Grundsteuer nicht im Fokus der steuerlichen Öffentlichkeit stand.

In gleicher Weise wie für die Steuerberater selbst boten die Monate Januar bis Oktober 2022 auch für Softwarehersteller wie Taxy.io mit dem Produkt SmartGrundsteuer eine Vielzahl von Tiefschlägen und Erfolgserlebnissen. Eine besondere Herausforderung bestand sicherlich darin, dass aufgrund der unterschiedlichen gesetzlichen Regelungen in einzelnen Bundesländern sechs verschiedene Formularmodelle mit insgesamt über 1.000 Formularfeldern quasi gleichzeitig

implementiert werden mussten. Dazu kam noch die laufende Aktualisierung der erforderlichen Datensätze durch die Finanzverwaltung. Wie sollte es da gelingen, bis zum Frühsommer 2022 die Abgabe der Erklärungen über die ELSTER-Schnittstelle zu ermöglichen.

Bereits bei der benutzerfreundlichen Gestaltung und Aufbereitung der Formulardatensätze ist eine hohe steuerfachliche Kompetenz gefragt.¹ Hier konnte man immerhin auf der von der Finanzverwaltung zur Kommunikation mit der ELSTER-Schnittstelle bereitgestellten ERiC Library aufbauen, die exakt die erforderlichen Formate und notwendigen Felder beschreibt.² Demgegenüber gab es jedoch weitaus weniger Vorgaben bzw. Orientierungshilfen, anhand

¹ Über eine einfache Verschaltung von spezifischen Fragen in Smart-Grundsteuer wurden dem Nutzer nur die jeweils für ihn maßgeblichen Fragen angezeigt, was die Anzahl der Fragen erheblich reduziert und die Verständlichkeit erheblich erhöht hat.

² Vgl. <https://fmos.link/9753> (Abruf: 04.01.2023).

derer die Berechnungen für die Grundsteuer erstellt werden konnten. Dabei sind für den Steuerberater gerade die Berechnungsergebnisse sowohl als Arbeitsergebnis als auch als eine mögliche Grundlage für die Abrechnung essenziell.

Die unterschiedliche Ausgangssituation bei der Erstellung der Formulare (ERIC Library) gegenüber der Erstellung der Berechnungen (keine Grundlagen) führten auch dazu, dass die Zusammenarbeit zwischen Steuerfachexperten und Softwareentwicklern komplett unterschiedlich gestaltet werden musste. Während wesentliche Grundlagen bei der Formulardatenerstellung durch die Schnittstelle vorgegeben waren, musste bei den Berechnungen eine viel intensivere und grundlegendere Zusammenarbeit zwischen den beiden Disziplinen erfolgen. Insbesondere musste eine geeignete gemeinsame Sprache gefunden werden, die als Transfermedium für die Übertragung der in Steuerfachsprache verfassten gesetzlichen Vorgaben in Softwarecode fungieren konnte.

Das Ziel des vorliegenden Beitrags ist es aufzuzeigen, wie eine erfolgreiche interdisziplinäre Zusammenarbeit zwischen Steuerfachexperten und Softwareentwicklern gelingen kann und welche Herausforderungen zu meistern sind. Als Beispiel dient hierbei die Definition, Entwicklung und Implementierung der Berechnungsmodule zu den Vorgaben der Grundsteuerreform.

2. Vorüberlegungen

2.1 Strukturierung und Priorisierung

Wie oben bereits erläutert, war es bereits eine immense Herausforderung, allein die erklärungs-technische Abgabe der Grundsteuer bis zum Frühsommer 2022 zu ermöglichen. Um sich dem Thema der Grundsteuerberechnungen zu nähern, wurde in zwei Stufen vorgegangen: (1) Zunächst wurden die zu erstellenden Berechnungsmodule anhand verschiedener Kriterien unterteilt. (2) Darauf aufbauend wurde eine Priorisierung der zu implementierenden Berechnungsmodule vorgenommen und (3) deren jeweiliger Umfang festgelegt.

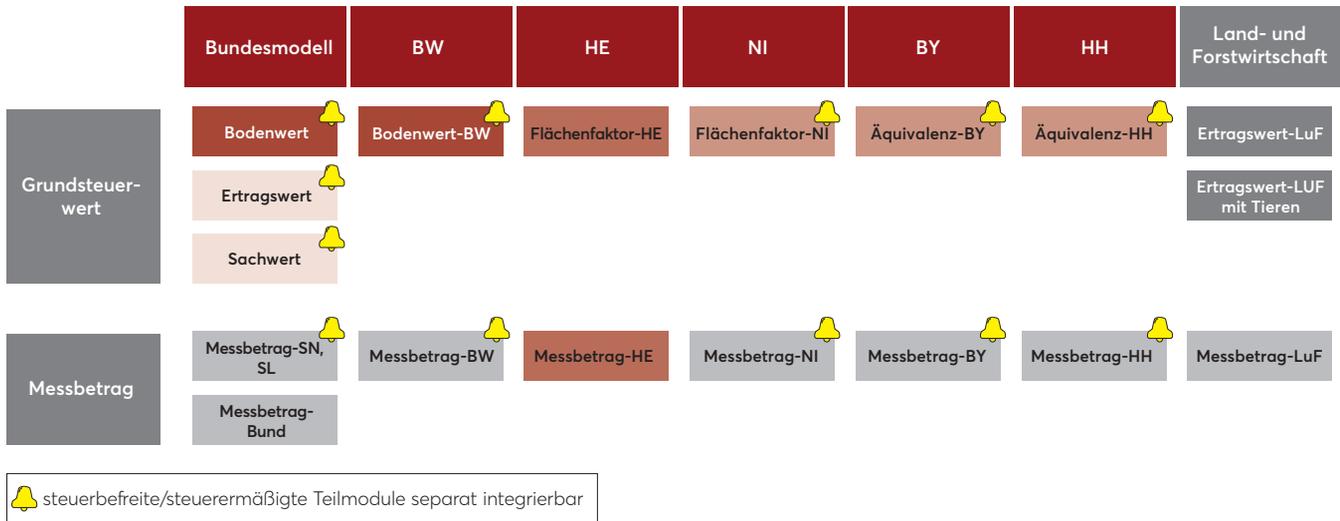
Als Basis für die Priorisierung musste zunächst eine Modellstruktur entwickelt werden. Einen validen Ausgangspunkt hierfür bildet das grundsätzlich dreistufige Berechnungsschema der Grundsteuer, das aus der Feststellung eines

Grundsteuerwerts (z.B. § 220 BewG) sowie der Festsetzung eines Steuermessbetrags (z.B. § 16 GrStG) besteht, auf den dann schlussendlich ein Hebesatz (z.B. § 27 GrStG) angewandt wird. Beim Bundesmodell war darüber hinaus von Bedeutung, dass für die einzelnen Berechnungsverfahren nach § 179, § 182 BewG separate Teilmodule erstellt werden müssen. Nach einer weiteren Durchsicht der Datensätze konnte für die meisten Bundesländer ergänzend eine weitere Abgrenzung genutzt werden, um eine initiale Implementierung zu vereinfachen. So sehen die einzelnen Formular Datensätze viele Felder vor, die Angaben zu Grundsteuerbefreiungen oder Grundsteuerermäßigungen betreffen. Beispiele sind Grundbesitz von kirchlichen Einrichtungen oder denkmalgeschützte Immobilien, die von Befreiungs- oder Ermäßigungstatbeständen profitieren können. Mit Ausnahme von Hessen muss hierzu ein sogenanntes "Globalfeld" im Mantelbogen (offizielle Bezeichnung: "GW-1") gewählt werden, bevor dann innerhalb weiterer Felder die einzelnen Befreiungs- und Ermäßigungstatbestände genauer anzugeben sind. Um die initiale Implementierung zu vereinfachen, wurde daher entschieden, Grundsteuerbefreiungen bzw. Grundsteuerermäßigungen erst später in die Berechnungen zu integrieren. Neben der separaten Bildung einzelner Module für die wirtschaftlichen Einheiten des Grundvermögens wurde schließlich berücksichtigt, dass die Formular Datensätze im Bereich der Land- und Forstwirtschaft (§ 218 Abs. 1 BewG) über alle Bundesländer hinweg nahezu identisch sind und daher nur einmal implementiert werden müssen. Als Ergebnis der vorstehend genannten Überlegungen hat sich die in Abbildung 1 dargestellte Modulstruktur ergeben.

Ausgehend von dieser Strukturierung wurde anschließend die Priorisierung der einzelnen Module festgelegt. In Anbetracht von insgesamt 36 Mio. wirtschaftlichen Einheiten, wovon 31 Mio. dem Grundvermögen zuzurechnen sind, wurde entschieden, den Fokus zunächst auf die Berechnungsmodule für das Grundvermögen zu richten.³ Aufgrund der Ähnlichkeit der Formular Datensätze innerhalb der verschiedenen Modelle ließen sich ergänzend spezifische "Modulgrup-

³ Vgl. zur Anzahl der wirtschaftlichen Einheiten BMF, Monatsbericht Juni 2022, S. 18.

Abbildung 1: Modulstruktur für die Grundsteuerberechnungen



pen” bilden.⁴ Begonnen wurde schließlich mit dem Ertragswertverfahren des Bundesmodells. Der Grund dafür lag in der breiten Anwendbarkeit, wohl wissend, dass es sich dabei um eines der komplexesten Berechnungsverfahren handeln würde.

2.2 Verfügbare Informationen und Herangehensweise

Zur Definition der Berechnungen standen aus steuerfachlicher Sicht zu Beginn des Prozesses lediglich die Gesetze, einige wenige Verwaltungsanweisungen sowie die Formulardatensätze und Ausfüllanleitungen zur Verfügung.⁵ Weitergehende Informationen vonseiten der Finanzverwaltung waren zu dem damaligen Zeitpunkt nicht vorhanden.

In Anbetracht dessen wurde zunächst anhand der Formulardatensätze untersucht, welche Formularfelder für die Berechnungen relevant sind. Anders ausgedrückt musste quasi deduktiv anhand der Formularfelder überlegt werden, zu welcher Gesetzesstelle bzw. Verwaltungsanweisung die entsprechenden Formularfelder passen. Bereits an dieser Stelle waren so manche

Merkwürdigkeiten feststellbar: So sieht das Ertragswertverfahren in bestimmten Fällen die Berücksichtigung einer selbstständig nutzbaren Teilfläche vor (§ 257 Abs. 3 BewG sowie die Übersicht in A 252 AEBewGrStG); in den Formularen sind hierzu jedoch keinerlei entsprechende Formularfelder vorgesehen, sodass dies auch in die entsprechenden Berechnungen nicht mit aufgenommen werden konnte.

Viele Unklarheiten konnten erst mit der Veröffentlichung von Musterbescheiden durch die Finanzverwaltung Anfang Juli 2022 beseitigt werden.⁶ Die Krux dabei: Es wurden lediglich Musterbescheide für das Bundesmodell sowie das bayerische Modell zur Verfügung gestellt. Für die weiteren Modelle bzw. auch die Berechnungen für die land- und forstwirtschaftlichen Einheiten existieren hingegen bis heute keine Musterbescheide von der Finanzverwaltung. In all diesen Fällen wurden die Berechnungen ausschließlich anhand der im vorigen Abschnitt beschriebenen deduktiven Vorgehensweise erstellt.

Zur Verdeutlichung der deduktiven Vorgehensweise ist in Abbildung 2 die Analyse des Datensatzes aus dem Ertragswertverfahren dargestellt. Die Unterteilung der Datenfelder wurde dabei

⁴ Vgl. für eine ausführliche Beschreibung der Datensätze Bräutigam/Küttig, NWB 2022 S. 2221. In Abbildung 1 sind diese Modulgruppen durch eine entsprechende Einfärbung hervorgehoben.

⁵ Bestimmte Verwaltungsanweisungen wurden erst Ende Juli (Baden-Württemberg) bzw. Ende September (Bayern) veröffentlicht.

⁶ Diese wurden im Rahmen der Bereitstellung der finalen Formulardatensätze Anfang Juli 2022 über das ELSTER-Portal bereitgestellt.

Abbildung 2: Strukturierung des Datensatzes im Bereich des Ertragswertverfahrens

Kontext	Name	Beschreibung	Grundsteuerwert	Grundsteuerbefreiungen	Grundsteuerermäßigungen
Ang_Wohn/Ang_Durchschn/Wohn_unter60	E7403131	Anzahl der Wohnungen mit einer Wohnfläche von 60m ²	Grundsteuerwert		
Ang_Wohn/Ang_Durchschn/Wohn_unter60	E7403132	mit einer gesamten Wohnfläche in m ²	Grundsteuerwert		
Ang_Wohn/Ang_Durchschn/Wohn_unter60/Befreiung	E7421101	Lage der Räume/Bezeichnung			
Ang_Wohn/Ang_Durchschn/Wohn_unter60/Befreiung	E7421102	Steuerfreie Fläche in m ²		Grundsteuerbefreiungen	
Ang_Wohn/Ang_Durchschn/Wohn_unter60	E7421103	Nutzungsart:		Grundsteuerbefreiungen	
Ang_Wohn/Ang_Durchschn/Wohn_unter60/Ver	E7421104	Lage der Räume/Bezeichnung			
Ang_Wohn/Ang_Durchschn/Wohn_unter60	E7421105	Vergünstigte Fläche in m ²			Grundsteuerermäßigungen
Ang_Wohn/Ang_Durchschn/Wohn_unter60/Ver	E7421106	Art der Vergünstigung:			Grundsteuerermäßigungen

stets mit parallelem Blick ins Gesetz vorgenommen. So ist die „Anzahl der Wohnungen“ bedeutsam für die Bestimmung des Liegenschaftszinssatzes nach § 256 BewG; die Unterteilung der Formularfelder in verschiedene Größenklassen (z.B. unter 60 qm) wiederum spiegelt die Struktur der Standardmieten der Anlage 39 BewG wider (vgl. Abbildung 2).

3. Finden einer gemeinsamen Sprache am Beispiel des Ertragswertverfahrens

Mit der deduktiven Ableitung der für die Berechnung erforderlichen Felder wurde schließlich eruiert, welche Inputdaten für die Berechnungen notwendig sind. Hier entsteht nun jedoch die eigentliche Herausforderung, nämlich wie genau die Rechenoperationen in die Software implementiert werden sollen.

Die anfängliche Überlegung bestand darin, eine Low-Code-Plattform eines Drittanbieters einzusetzen und den Entwicklungsprozess somit unmittelbar an Steuerfachexperten zu übertragen.

Diese Überlegungen wurden jedoch aufgrund des hohen Zeitdrucks bei gleichzeitiger notwendiger Einarbeitungs- und Schulungszeit für eine neue Softwarelösung wieder verworfen. Auch wurde mit Blick auf das Ertragswertverfahrens deutlich, dass die notwendigen Berechnungsoperationen mitunter sehr komplex sind und ein tiefgreifendes Verständnis der Datenarchitektur verlangen. Im Gegensatz zu im Internet verfügbaren Grundsteuerrechnern⁷ genügt es als Softwarehersteller nicht, nur manche Fallkonstellationen zur Verfügung zu stellen. Ganz im Gegenteil, eine wettbewerbsfähige Softwarelösung muss zwingend alle erdenklich möglichen Konstellationen abbilden und berechnen, die sich anhand der Formulareingaben ergeben können. Ein prägnantes Beispiel stellt etwa ein kernsaniertes Einfamilienhaus auf einem Grundstück mit einer Abbruchverpflich-

⁷ Vgl. z.B. <https://grundsteuer.de/rechner/> oder <https://fmos.link/18740> (jew. Abruf: 04.01.2023).

Abbildung 3: Beispiel für den "Pseudocode" im Bereich des Ertragswertverfahrens

for item with path("GW2/Ang_Wohn/Ang_Durchschn")

	Anz_Wohnungen_X = Anz_Wohnungen_60_X + Anz_Wohnungen_60_100_X + Anz_Wohnungen_100_X
	AnzWohnungen_wEinheit = SUM(Anz_Wohnungen_X)

tung oder die Erfassung mehrerer Gebäude mit teilweisen Steuerbefreiungen bei gleichzeitigem Vorliegen eines Denkmalschutzes dar. Hinzu kam, dass aufgrund der modulbasierten Vorgehensweise eventuell auch Datenmigrationen bei bestehenden Objekten durchzuführen sind. Alles letztlich Gründe, die gegen die eine Low-Code-Lösung sprechen. Warum dies dennoch die Zukunft darstellt, dazu später mehr.

In enger Zusammenarbeit zwischen Steuerfachexperten und Entwicklern wurde schließlich der Entschluss gefasst, eine eigene, passgenaue Lösung zu entwickeln. Doch auch hierfür bedurfte es einer gemeinsamen Sprache. So wurden zunächst gewisse, einfach verständliche Ausdrücke festgelegt („if-Bedingungen“, „for-Schleifen“), die ohne Beachtung einer bestimmten Syntax von den Steuerfachexperten vordefiniert werden konnten („Pseudocode“). Die Verknüpfung dieser Ausdrücke mit den jeweiligen Feldkennungen ermöglichte eine effiziente und schnelle Formulierung von komplexen Rechenoperationen, die dann Grundlage für die Implementierung im Backend waren. Abbildung 3 zeigt beispielhaft die Berechnung der Gesamtzahl der angegebenen Wohnungen für eine wirtschaftliche Einheit im Ertragswertverfahren. Hierzu ist es in einem ersten Schritt erforderlich, die Wohnungsanzahl pro Gebäude („Wohnungen_X“) zu bestimmen, bevor dann die Gesamtzahl der Wohnungen der wirtschaftlichen Einheit ermittelt werden kann („_wEinheit“). Dies ist wiederum Voraussetzung, um (gerade bei Mietwohngrundstücken) den korrekten Liegenschaftszinssatz nach § 256 BewG für die Berechnung heranzuziehen (vgl. Abbildung 3).

4. Umsetzung der steuerfachlichen Anforderungen in der Software

Aufgrund des engen Zeitrahmens, der bei der Entwicklung von SmartGrundsteuer im Allgemeinen und bei der Entwicklung des Berechnungsmoduls im Besonderen einzuhalten war, wurde festgelegt,

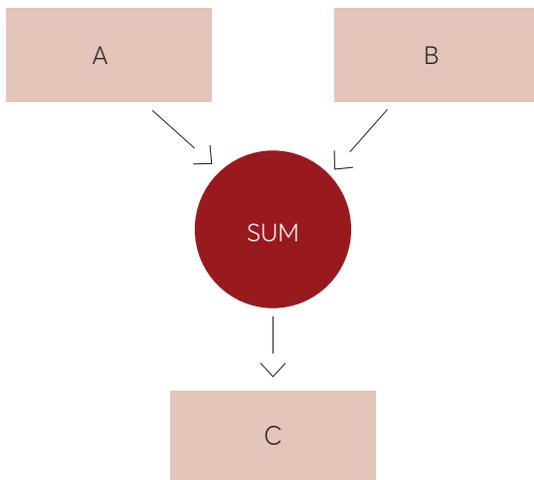
das Berechnungsmodul in Iterationsstufen zu entwickeln. Die Iterationsstufen folgten dabei der in Abbildung 1 dargestellten Modulstruktur. Am Ende einer jeden solchen Iterationsphase sollte das Berechnungsmodul eine Teilmenge der noch fehlenden Berechnungsmodelle unterstützen und/oder bereits umgesetzte Modelle erweitern. Entsprechend wurde das Ertragswertverfahren zunächst für Standardfälle implementiert, bevor in einem späteren Schritt dann eine Erweiterung für steuerbefreite bzw. steuerermäßigte Einheiten im Ertragswertverfahren vorgenommen wurde.

Die iterative Vorgehensweise war dabei nicht nur für die programmtechnische Implementierung der von Steuerfachexperten vorgegebenen Berechnungslogik naheliegend, sondern auch für die zeitintensive Spezifizierung der Berechnungslogik durch Steuerfachexperten selbst. Mit der ersten Iterationsphase, in der lediglich die Spezifikation für das Ertragswertverfahren vorlag, ging die zusätzliche Herausforderung einher, nicht nur die Implementierung des Berechnungsmodells zu vollziehen, sondern vielmehr auch eine nachhaltige Berechnungsarchitektur zu schaffen. Diese Architektur sollte einen flexiblen softwaretechnischen Rahmen („Framework“) darstellen, der ohne großen Aufwand die Integration neuer bzw. die Erweiterung vorhandener Berechnungsmodelle erlauben würde. Eine der Hauptanforderung an das übergreifende Framework zur Entwicklung und Bereitstellung von Grundsteuerberechnungsmodellen innerhalb von SmartGrundsteuer bestand somit in der Schaffung einer geeigneten Abstraktionsschicht, welche die nötigen Elemente bereitstellt, um alle Berechnungsmodelle zu beschreiben und umzusetzen.

Auch wenn anfänglich nicht alle Berechnungsmodelle spezifiziert waren, so wurde anhand einer vorläufigen Analyse offensichtlich, dass jedes Modell als eine lineare Aneinanderreihung von sich aufeinander beziehenden mathemati-

schen Rechenoperationen verstanden werden kann. Diese elementare Einsicht legte es nahe, ein Berechnungsmodell mathematisch als einen Berechnungsgraphen abzubilden und als solchen softwaretechnisch umzusetzen. In der Mathematik wird ein Graph als Ensemble von verbundenen Einheiten verstanden, die ihrerseits als Knoten bezeichnet werden. Diese Knoten sind miteinander durch Kanten verbunden. Haben die Kanten eine Richtung (man spricht von einem gerichteten Graphen), so kann der Graph als Modell für einen Datenfluss fungieren. In einem Berechnungsgraphen repräsentieren die Knoten mathematische Operationen, während Kanten Input- bzw. Output-Daten darstellen, je nachdem ob es sich um eine eingehende oder ausgehende Kante handelt. In Abbildung 4 ist exemplarisch die mathematische Operation der Summation von zwei Zahlen ("A" und "B") als Graph dargestellt. Eine Kante ist jeweils durch einen Pfeil sowie ein Rechteck mit einem Identifikator der durch sie repräsentierten Daten dargestellt.

Abbildung 4: Darstellung der Addition zweier Zahlen in einem Berechnungsgraphen



Kanten, die nicht von Knoten ausgehen, stellen Eintrittspunkte in den Berechnungsgraphen dar und stehen in einer 1:1-Beziehung zu den Input-Daten des Modells. Kanten ohne nachfolgende Knoten sind die Austrittspunkte aus dem Berechnungsgraphen und entsprechen den finalen Berechnungsergebnissen. Zusammen mit den Zwischenergebnissen bilden sie die Output-Daten des Modells, welche für die Darstellung der Berechnung in

einer an einen Steuerbescheid angelehnten Form im Benutzerinterface verwendet werden. Unter Berücksichtigung aller relevanten Formularfelder und ihrer jeweiligen Berechnungsabhängigkeiten können sich hier sehr komplexe Graphen ergeben. Ein zu der in Abbildung 3 dargestellten Berechnungslogik korrespondierender Berechnungsgraph ist in Abbildung 5 graphisch dargestellt.

Das gesamte Ertragswertmodell, dargestellt als Berechnungsgraph, findet sich schließlich in Abbildung 6, die unter dem folgenden QR-Code abrufbar ist. Dem Berechnungsgraphen liegt dabei die Annahme einer wirtschaftlichen Einheit mit zwei Wohngebäuden zugrunde. Auf die für Steuerermäßigungen und Steuerbefreiungen nötige Berechnungslogik wurde aus Gründen der Übersichtlichkeit verzichtet.

<https://fmos.link/18882>



Nachdem die für die Definition und Ausführung eines Berechnungsgraphen notwendige Softwareschicht fertiggestellt war, konnte mit der Implementierung eines Berechnungsmodells begonnen werden. Die Hauptaufgabe bestand dabei darin, den durch Steuerexperten formulierten Pseudocode in einen entsprechenden spezifischen Berechnungsgraphen zu transferieren. Dieser Transferschritt wurde aufgrund der fehlenden zeitlichen Ressourcen manuell umgesetzt, in Zukunft soll der Schritt jedoch automatisiert erfolgen und den Kern einer Low-Code-Lösung für die Implementierung von Berechnungslogiken durch Steuerfachexperten bilden.

Zusammenfassend war eine Vielzahl von technischen Herausforderungen auf dem Weg zu einem einwandfrei funktionierenden Berechnungsmodul in SmartGrundsteuer zu meistern. Da war die Notwendigkeit der Implementierung einer modellabhängigen Logik für die Validierung der vom Nutzer eingegebenen Formulardaten mit der Zielsetzung, Berechnungen bereits durchzuführen, ohne dass jedes Mal zunächst eine (mitunter zeitaufwändige) Validierung des gesamten Datensatzes durch die ELSTER-Schnittstelle

Abbildung 5: Darstellung der Berechnungslogik aus Abbildung 3 in einem Berechnungsgraphen



erforderlich ist. Da waren die zu implementierenden Regeln, welche von Fachexperten als Pseudocode bereitgestellt und anschließend von Entwicklern in die Software integriert werden mussten. Dazu gab es auch vermeintlich triviale Aufgabenstellungen. So etwa die Implementierung korrekter Rundungsregeln oder die Extraktion der zugehörigen Steuerklasse aus den amtlichen Tabellen nach der MietNEinV basierend auf den freitextlichen Benutzereingaben.

Bezüglich der Verwendung des Pseudocodes ist noch anzumerken, dass es vordergründig zwar so erscheinen mag, als wäre es möglich, diesen lediglich als abstrakte Modellrechnung zu betrachten und ohne große intellektuelle Anstrengung in Programmcode zu übertragen. Es hat sich jedoch gezeigt, dass es durchaus von Vorteil ist, wenn die Entwickler bei der Arbeit mit dem Pseudocode diesen, wenn nicht aus steuerfachlicher Perspektive, so doch unter Zuhilfenahme des gesunden Menschenverstandes interpretieren und dabei

einer Überprüfung auf logische Stimmigkeit unterziehen. Auf diese Weise konnte so mancher Fehler bereits im Vorfeld der Implementierung eliminiert werden. Zudem gewannen die Entwickler mit der Zeit auch eine gewisse Sicherheit im Umgang mit der Materie und scheuten sogar nicht davor zurück, den ein oder anderen Abschnitt des Pseudocodes mit dem Gesetzestext abzugleichen.

5. Low-Code-Lösung zur effizienten Zusammenarbeit von Steuerfachexperten und Entwicklern

Am Beispiel der Entwicklung der Berechnungsmodule für die Grundsteuer tritt die Wichtigkeit einer effizienten Zusammenarbeit zwischen Steuerfachexperten und Entwicklern deutlich zu Tage. Der Erfolg dieser interdisziplinären Zusammenarbeit setzt insbesondere voraus, dass eine gemeinsame Sprache gefunden und gesprochen wird. Im vorliegenden Beispiel besteht die gemeinsame Sprache in den für beide Seiten einfach zu verstehenden Berechnungsgraphen,

die sowohl mit dem Pseudocode der Steuerfachexperten wie auch mit dem Programmcode der Entwickler korrespondieren. Bei den Berechnungsmodulen für die Grundsteuer wurde der Pseudocode der Steuerfachexperten noch manuell durch Entwickler interpretiert und über Programmcode in Berechnungsgraphen überführt. Dieser Schritt soll zukünftig über eine Low-Code-Lösung automatisiert werden, mit der Steuerfachexperten ihre steuerliche Logik direkt über Berechnungsgraphen modellieren können. Konkret sollen die Steuerfachexperten ihre Berechnungen – z.B. per Drag-and-Drop von Berechnungsblocken – in einer intuitiven, visuellen Umgebung erstellen. Dabei kann vereinfachend auf eine Vielzahl von Berechnungsfunktionen zurückgegriffen werden, die bereits in das Modul integriert sind. Diese Module sollen dann direkt vom Steuerfachexperten in die Anwendung integriert werden können, ohne dass diese dazu Programmierkenntnisse besitzen müssen.

Die Entwicklung einer solchen Low-Code-Lösung, mit der Steuerfachexperten selbst in die Lage versetzt werden, steuerliche Anwendungen zu implementieren, ist zentraler Bestandteil zukünftiger Anwendungen von Taxy.io. Die Vision besteht darin, eine Art Betriebssystem für Steuerberater und damit die führende B2B-Plattform für eine proaktive und automatisierte Steuerberatung zu etablieren. Für Steuerfachexperten soll Taxy.io das werden, was Personio für HR-Teams ist: eine umfassende Cloud-Plattform, die ihnen mit unterschiedlichen steuerlichen Anwendungen bei all ihren Aufgaben hilft, wie etwa bei Recherche- und Analysetätigkeiten, bei der Beantwortung von fachlichen Fragen, bei der Kommunikation mit Mandanten oder bei der Erstellung von Steuererklärungen, vom automatischen Datenimport über das kollaborative Ausfüllen der Formulare gemeinsam mit dem Mandanten bis hin zur Abgabe der Erklärung über die ELSTER-Schnittstelle und dem Prüfen des Bescheids. Neben der Zeitersparnis soll es darüber hinaus möglich sein, neue Beratungsanlässe zu identifizieren, um proaktive Steuerberatung zu leisten.

6. Zusammenfassung

Die Definition, Entwicklung und Implementierung der Berechnungsmodule zu den Vorgaben der Grundsteuerreform hat gelehrt, dass gerade das Finden einer gemeinsamen Sprache sowie die Beschäftigung mit einer nachhaltigen Berechnungsarchitektur der Schlüssel zum Erfolg

war. Ausgehend davon konnten die weiteren Berechnungsmodule zeitnah und ohne größere Schwierigkeiten jeweils im Wochenrhythmus in die Applikation integriert werden.

Dazu wurde mit dem Aufbau der Berechnungsarchitektur ein wichtiger Grundstein für weitere Applikationen und eine Art künftiges Betriebssystem für Steuerberater gelegt. Die damit einhergehenden Potenziale zur Effizienzsteigerung sind dabei umso wichtiger, da aktuell davon auszugehen ist, dass die Finanzverwaltung ihren eigenen Berechnungskern den Softwareherstellern auch in Zukunft nicht zur Verfügung stellen wird.

Abschließend lässt sich zusammenfassen, dass die Zeit der intensiven Zusammenarbeit zwischen Steuerfachexperten und Entwicklern von beiden Seiten als sehr bereichernd empfunden wurde. Auch wenn die Entwickler am Ende der Zusammenarbeit sicherlich nicht behaupten können, kompetente Ansprechpartner für das Thema Grundsteuer zu sein, so können sie jetzt doch selbstbewusst an einem Gespräch zu diesem Thema teilnehmen. Den Steuerfachexperten wiederum wurde bewusst, dass eine Softwareimplementierung keine Fehler verzeiht und bei der Spezifikation die höchste Sorgfalt anzulegen ist, da ansonsten die spätere Fehlerbehebung umso zeitaufwändiger ist. ■



Dr. Rainer Brütigam
ist bei Taxy.io als Senior Tax Expert/Syndikus-Steuerberater tätig.

Aktuell ist er unter anderem an der steuerfachlichen Konzeption von Softwareanwendungen beteiligt. Zugleich kennt er aus einer mehrjährigen Tätigkeit auch die Beratungspraxis in einer Steuerkanzlei.



Dr. Grigoriy Volovskiy
ist bei Taxy.io als Senior Data Scientist tätig.

Sein Interesse gilt der Deep-Learning-basierten Textverarbeitung.



Steffen Kirchoff
verantwortet als Mitgründer und CTO die Produktentwicklung der Taxy.io GmbH.

Zudem leitet er im Institut für Digitalisierung im Steuerrecht (IDSt) die Arbeitsgruppe zum Einsatz von KI im steuerlichen Kontext. Zuvor hat er an der Harvard-Universität und der RWTH Aachen an Algorithmen und Anwendungen im Bereich Machine Learning geforscht und war als Data Scientist bei einem internationalen Versicherungsunternehmen tätig.