

---

# **MetaDataManagement Documentation**

*Release 1.0.79*

**Robert Birkelbach, Anne Gärtner, René Reitmann**

**19.02.2019**



---

## Inhaltsverzeichnis:

---

<b>1</b>	<b>User-Rollen</b>	<b>1</b>
1.1	Public User . . . . .	1
1.2	Datengeber (Data Provider) . . . . .	1
1.3	Publisher . . . . .	1
1.4	Admin . . . . .	1
<b>2</b>	<b>Metadatenabgabe (DatengeberInnen)</b>	<b>3</b>
2.1	Allgemeines . . . . .	3
2.2	Vorbereitende Schritte . . . . .	4
2.3	Notwendige Schnittstellen und Dateien . . . . .	7
2.4	Die Abgabe von Metadaten für die einzelnen Ebenen . . . . .	8
2.5	Projekte freigeben . . . . .	23
<b>3</b>	<b>FDZ-MitarbeiterIn (Publisher, Developer)</b>	<b>25</b>
3.1	ID-Vergabe . . . . .	25
3.2	Neues Projekt anlegen . . . . .	26
3.3	Bilderfassung aus PDF Dateien . . . . .	28
3.4	Bilderfassung aus RagTime . . . . .	36
3.5	Excel-Tabellen . . . . .	41
3.6	Templates . . . . .	44
3.7	Excel Makros verwenden . . . . .	45
3.8	Editing Macros Excel . . . . .	45
3.9	Anpassungen LateX Template DSR . . . . .	47
3.10	Datensatzreport (Lektorat) . . . . .	48
3.11	Fragen (questions) . . . . .	51
3.12	Variablen (variables) . . . . .	55
3.13	Erstellung der Variable-JSON Dateien . . . . .	59
3.14	Variables (Zofar) . . . . .	60
3.15	Related Publications . . . . .	60
3.16	Datensatzreport erzeugen . . . . .	61
3.17	Prüfung der Jsons nach Umstellung der Generierungsskripte . . . . .	61
3.18	Skalenniveau prüfen . . . . .	62
3.19	Skalenniveau Diese Informationen dienen dazu die verschiedenen . . . . .	62
3.20	Stata-Skripte . . . . .	63
3.21	Testen von MDM-Issues . . . . .	63
3.22	Datentypen . . . . .	64
3.23	Die korrekte Anordnung der Dateien im Ordner . . . . .	65

3.24	Übersicht über verschiedene R-Helfer-Skripte . . . . .	65
3.25	Verantwortlichkeiten . . . . .	66
3.26	Projektrelease . . . . .	66
3.27	Domänenmodell . . . . .	66
3.28	Javadoc . . . . .	67
<b>4</b>	<b>Indizes und Tabellen</b>	<b>121</b>

Im Metadatenmanagementsystem (MDM) gibt es mehrere User-Rollen, für die jeweils nur bestimmte Teile der Dokumentation von Interesse sind. Im folgenden werden daher die Rollen aufgeführt und kurz erklärt.

### 1.1 Public User

Student, Researcher

### 1.2 Datengeber (Data Provider)

DZHW interne/externe DatengeberIn, siehe *Metadatenabgabe (DatengeberInnen)*.

### 1.3 Publisher

DZHW-FDZ MitarbeiterIn

### 1.4 Admin

Weist User-Rollen zu.



---

## Metadatenabgabe (DatengeberInnen)

---

### 2.1 Allgemeines

#### 2.1.1 Der Datenaufnahmeprozess im FDZ des DZHW

Das FDZ des DZHW stellt Daten quantitativer und qualitativer Erhebungen aus dem Feld der Hochschul- und Wissenschaftsforschung zur Verfügung. Ein fester Bestandteil dieser Arbeit ist zunächst der Prozess der Datenaufnahme, welcher im FDZ des DZHW durch ein selbst entwickeltes System, das Metadatenmanagementsystem (MDM) unterstützt wird. Das Besondere am MDM ist, dass Informationen über die eigentlich erhobenen Forschungsdaten, also Metadaten, erfasst werden. Für die strukturierte Aufnahme der Metadaten sind sieben unterschiedliche Ebenen im MDM vorgesehen: *Studie*, *Erhebungen*, *Erhebungsinstrumente*, *Fragen*, *Datensätze*, *Variablen* und *Publikationen*.

Innerhalb der Aufnahme von Forschungsdaten wird anhand der Metadaten auf diesen Ebenen erfasst, welcher Studie sowie Erhebung die Daten angehören, welche Erhebungsinstrumente genutzt wurden, welche Fragen darin gestellt wurden, welche Datensätze existieren, welche Variablen sich darin befinden und welche Veröffentlichungen bereits mit den Daten realisiert worden sind. Dadurch, dass alle Ebenen miteinander verknüpft sind, wird eine umfassende Durchsuchbarkeit aller verfügbaren Daten, die vom FDZ des DZHW über das MDM verwaltet werden, ermöglicht. Das System ist über die Website <https://metadata.fdz.dzhw.eu> zu erreichen.



Abb. 2.1: Darstellung der verschiedenen Ebenen im MDM, Ebene *Studie* aktiv

#### 2.1.2 Die eigenständige Abgabe von Metadaten

Wenn Sie Ihre Daten im FDZ des DZHW abgeben möchten, erfassen Sie die projektbezogenen Metadaten selbst und können diese teilweise eigenständig in das MDM hochladen.

Die Abgabe der Metadaten ist innerhalb der einzelnen Ebenen unterschiedlich komplex, sodass die Daten für jede der sieben Ebenen separat erfasst werden müssen. Hierfür hat das FDZ des DZHW feste Strukturen entwickelt, die im

weiteren Verlauf dieser Anleitung für jede Ebene detailliert erläutert werden. Diesen Vorgaben ist unbedingt Folge zu leisten, damit eine erfolgreiche Erfassung der einzelnen Metadaten gewährleistet werden kann.

Aktuell können einige Metadaten über eine Eingabemaske im MDM direkt eingetragen werden. Für die übrigen Ebenen müssen Json Dateien hochgeladen werden, bzw eine Exceldatei ans FDZ übergeben werden bzw. hochgeladen werden. In dieser Doku wird lediglich auf den Standardfall eingegangen, bei dem die Metadaten per Eingabemaske eingetragen werden. Zusätzlich ist auf manchen Ebenen das Erstellen von weiteren Anhängen (vgl. *Anhänge*) vorgesehen. Die Excel-Tabellen und eventuelle Anhänge müssen für jede Ebene dann entweder ins MDM hochgeladen werden oder zunächst dem FDZ zur weiteren Bearbeitung geschickt werden. Tabelle 1 zeigt eine erste Übersicht über das Vorgehen der Metadateneingabe auf jeder Ebene, detaillierte Erläuterungen werden in den nachstehenden Kapiteln folgen.

Tabelle 1: Vorgehen bei der Metadateneingabe pro Ebene

Ebene	Metadaten eingeben	Zusätzliche Dateien	Metadaten abgeben
Studie	Eingabemaske	Anhänge (PDF)	Eingabemaske ausfüllen
Erhebungen	Eingabemaske	Anhänge (PDF)	Eingabemaske ausfüllen
Erhebungs- instrumente	Eingabemaske	Anhänge (PDF, Excel)	Eingabemaske ausfüllen
Fragen	Excel-Tabelle (oder Zofar)	Fragebilder (.png)	Im Vorlage-Ordner speichern
Datensätze	Eingabemaske	Anhänge (PDF)	Eingabemaske ausfüllen
Variablen	mind. 1 Excel-Tabelle	mind. 1 Datensatz (Stata)	Im Vorlage-Ordner speichern
Publikationen	•	•	Informationen dem FDZ schicken

## 2.2 Vorbereitende Schritte

### 2.2.1 Registrierung

Sie müssen sich zunächst auf <https://metadata.fdz.dzhw.eu> registrieren, um die Berechtigung für das Erfassen von Metadaten zu erhalten. Dies können Sie über die Sidebar links auf der Website erledigen:



Abb. 2.2: Registrierung im MDM

Im Anschluss erhalten Sie eine Bestätigungsmail. In dieser müssen Sie auf den Aktivierungslink klicken, welcher nach drei Tagen automatisch abläuft. Im Anschluss werden Sie von uns dem Projekt in der Rolle Datengeber/in hinzugefügt. Sie erhalten jeweils eine E-Mail wenn Sie Ihrem Datenaufbereitungsprojekt hinzugefügt wurden und wenn das Projekt Ihnen von einer FDZ-Mitarbeiterin zur Bearbeitung freigegeben wurde.

## 2.2.2 Verwaltung des Projektes im Projekt-Cockpit

Das Projekt-Cockpit dient der Zusammenarbeit zwischen FDZ-MitarbeiterInnen und den DatengeberInnen (das sind Sie und Ihre KollegInnen). In der Navigationsleiste links, welche je nach Fenstergröße des Browsers aufgeklappt werden muss, finden Sie den Zugang zum Projekt-Cockpit (vgl. Abb. 2.3). Wenn Sie ins Projekt-Cockpit gehen, sehen Sie unter dem Punkt Einstellungen (vgl. Abb. 2.4), welche Publisher (FDZ-MitarbeiterInnen) und Datengeber dem Projekt zugewiesen sind und welche Metadaten erwartet werden.



Abb. 2.3: Projekt-Cockpit Button.

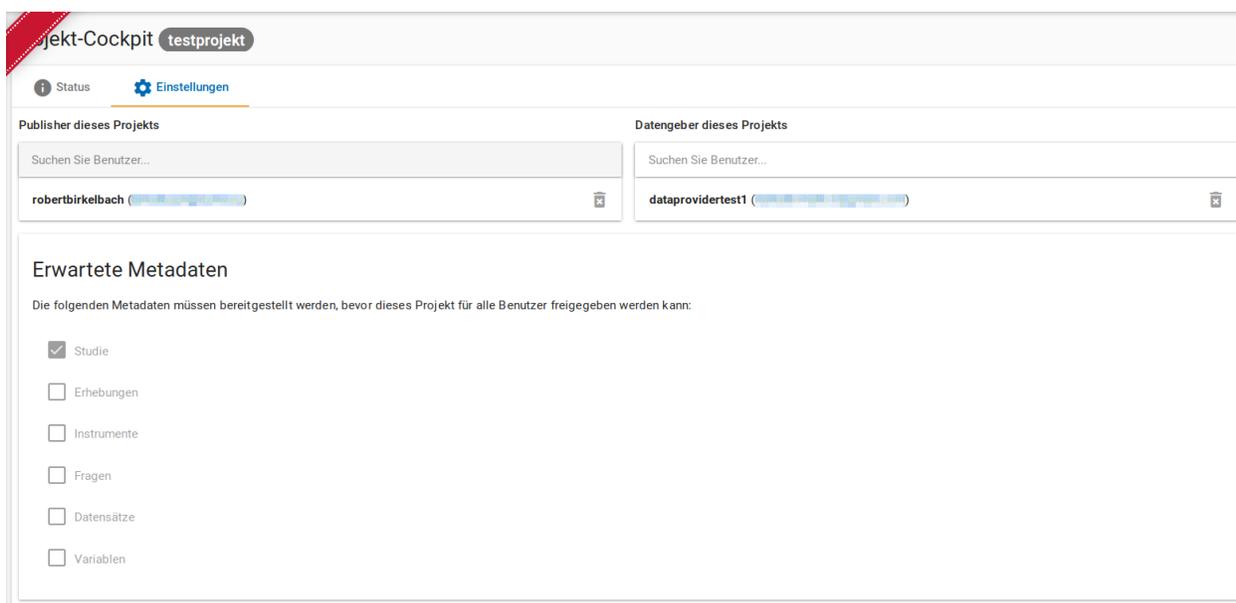


Abb. 2.4: Projekt-Cockpit Einstellungen.

Das Status-Menü (siehe Abb. 2.5) hat einerseits Funktionen zum Projektmanagement und andererseits Funktionen um Metadaten anzulegen:

Es wird angezeigt, ob das Projekt freigegeben ist. Freigegeben bedeutet, dass die eingegebenen Metadaten für alle öffentlichen Nutzer des Systems sichtbar sind. Daneben wird angezeigt, ob das Projekt gerade bei den Publishern liegt, oder bei den DatengeberInnen zur Bearbeitung liegt. Zuerst liegt es bei den Publishern und Sie werden per Mail benachrichtigt, wenn es Ihnen zugewiesen wird. Um das Projekt wieder den Publishern zuzuweisen, klicken Sie den „Papierflieger“-Button (siehe Abb. 2.2.2) über dem „Zugewiesen an Datengeber“ steht.

Die Voraussetzung, dass Sie das Projekt zurückgeben können ist, dass Sie die erwarteten Metadaten eingeben mittels des „Neu“ bzw. „Hochladen“ Buttons und als „fertig“ markiert haben (siehe Abb. 2.2.2). Wenn Sie auf den Neu-Button klicken, gelangen Sie zur Eingabemaske der jeweiligen Ebene und mit Klick auf den Hochladen Button erscheint ein File-Explorer Fenster. Details zur Abgabe der Metadaten bei den einzelnen Ebenen wird im jeweiligen Kapitel erklärt. Wenn Sie fertig mit der Eingabe der Metadaten einer Ebene sind setzen Sie bitte das entsprechende Häkchen und speichern dies ab (Disketten-Button rechts unten).

Sie sehen, dass sich der Statussmiley von traurig hin zu neutral ändert, nachdem Sie „fertig“ angeklickt haben. Nachdem der Papierflieger-Button geklickt wurde, erscheint der „Nachricht an Publisher“ Dialog (siehe Abb. 2.2.2).

Falls die Publisher denken, dass noch irgend etwas vergessen wurde oder anders eingegeben werden sollte, erhalten Sie eine Email und das Projekt wird Ihnen zurückgegeben. Sollte das nicht der Fall sein, markiert der Publisher die Ebene auch als „fertig“, was durch einen glücklichen Smiley signalisiert wird (siehe Abb. 2.2.2). Sind alle Smileys

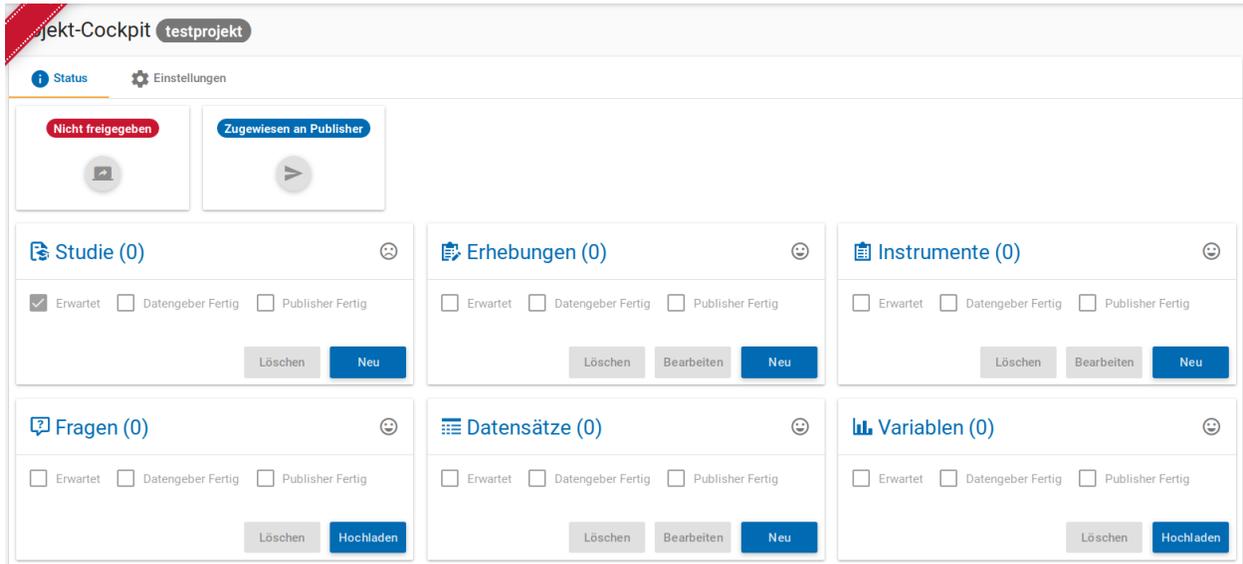
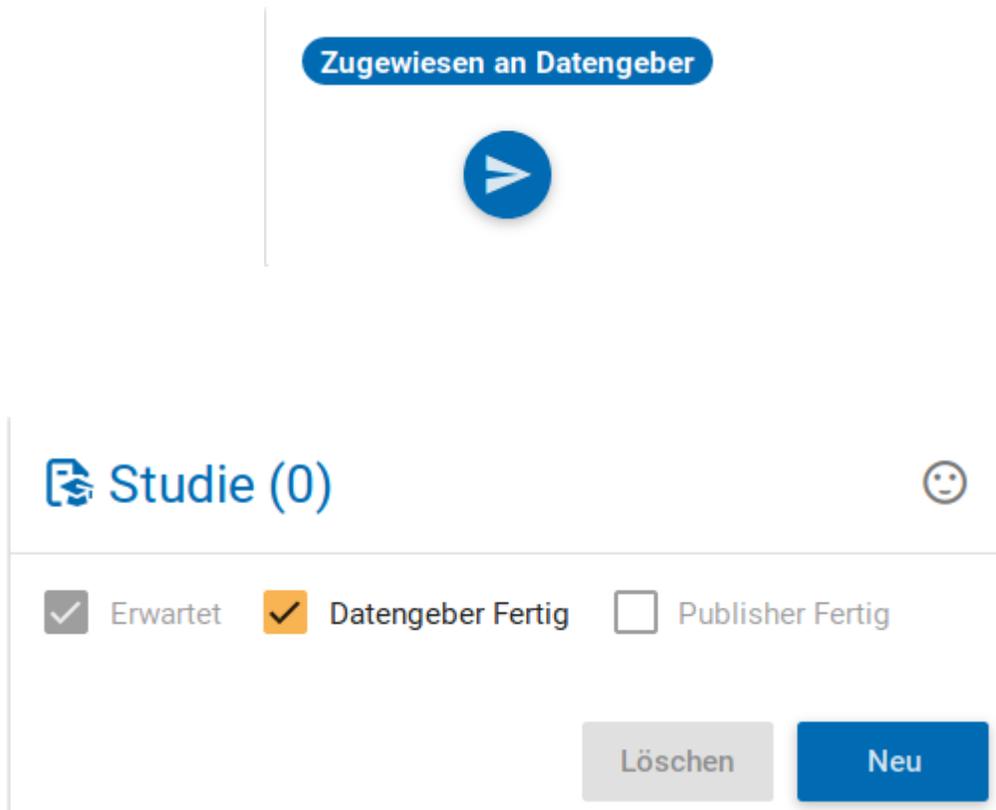


Abb. 2.5: Projekt-Cockpit Status.



Nachricht an Publisher
✕

Geben Sie eine Nachricht ein, die an alle Publisher dieses Projekts per E-Mail verschickt wird.

Nachricht\*  
Hallo, Ihr seid am Zug!

23 / 2048

Abbrechen
Zuweisen

glücklich, können die Publisher das Projekt freigeben, das heißt die Metadaten und damit letztlich die Datenprodukte allen öffentlichen Nutzern des Systems zur Verfügung stellen.

Studie (1)
😊

Erwartet

Datengeber Fertig

Publisher Fertig

Löschen
Bearbeiten

Sie können nun beginnen, Ihr Projekt mit Metadaten zu füllen. Wie genau dies funktioniert, wird im Folgenden zu nächst prinzipiell erläutert, ehe die konkret geforderten Metadaten in den einzelnen Ebenen in *Die Abgabe von Metadaten für die einzelnen Ebenen* detailliert beschrieben werden.

## 2.3 Notwendige Schnittstellen und Dateien

Grundsätzlich gibt es je Ebene verschiedene Möglichkeiten Metadaten zu erfassen: Eingabemasken und der Upload von Dateien welche Metadaten enthalten. Die Eingabemasken ermöglichen eine komfortable Abgabe der Metadaten direkt auf der Website.

### 2.3.1 Eingabemasken

Für die Ebenen Studie, Erhebungen, Datensätze und Instrumente steht die Nutzung von Eingabemasken im MDM zur Verfügung. Jede Maske umfasst verschiedene Felder, welche mit den einzutragenden Informationen beschriftet sind. Einige Felder sind verpflichtend auszufüllen und deshalb mit einem Sternchen versehen. Sie werden beim Speichern der Eingaben automatisch darauf hingewiesen, wenn noch Felder offen sind, die nicht leer bleiben dürfen. Die Bedienung der Eingabemasken ist weitgehend intuitiv und an vielen Stellen selbsterklärend. Im Rahmen der relevanten Ebenen Studie, Erhebungen, Datensätze und Instrumente wird die Handhabung der jeweiligen Eingabemasken dann konkret gezeigt (s. *Erhebungen (surveys) Erhebungsinstrumente (instruments)*).

Für die Ebenen Fragen, Variablen und Publikationen müssen Dateien hochgeladen werden – wenden Sie sich hierfür bitte ans FDZ.

## 2.3.2 Anhänge

Anhänge werden über die Eingabemasken im MDM hochgeladen. Zu den Anhängen zählen z.B. der Daten- und Methodenbericht auf der Studienebene sowie Fragebögen oder Codierlisten auf Instrumentenebene. Diese Dokumente müssen als Dateien im PDF- oder Excel-Format vorliegen (Details dazu finden Sie in den jeweils relevanten Kapiteln zu den einzelnen Ebenen) und zudem nach bestimmten Richtlinien benannt werden. Für die Anhänge im PDF-Format gilt es darüber hinaus zu beachten, dass dokumenteigene Metadaten wie Autor und Titel aus der PDF-Datei gelöscht werden. Dies können Sie im PDF-Dokument über „Datei“ -> „Eigenschaften...“ erledigen.

## 2.4 Die Abgabe von Metadaten für die einzelnen Ebenen

### 2.4.1 Studie (study)

#### Übersicht

Anhand der Informationen, die Sie bzgl. Ihrer Studie im MDM erfassen, wird dort später eine Übersichtsseite erstellt, die im Folgenden am Beispiel des Absolventenpanels 2005 dargestellt wird:

DZHW-Absolventenpanel 2005 stu-gra2005\$

**Details**

**Erhebungsreihe:** DZHW-Absolventenstudien

**Erhebende Institution:** Deutsches Zentrum für Hochschul- und Wissenschaftsforschung (DZHW)

**Projektmitarbeiter(innen):** Minks, Karl-Heinz; Briedis, Kolja; Grotheer, Michael; Isleib, Sören; Netz, Nicolai

**Gefördert von:** Bundesministerium für Bildung und Forschung (BMBF)

**Erhebungsdesign:** Panel

**Verfügbare Wellen:** 2

**Erhebungsdatentyp:** Quantitative Daten

**DOI:** 10.21249/DZHW:gra2005:1.0.0 [🔗](#)

**Version der Datensätze:** 1.0.0

**Verbundene Objekte**

- 📄 Erhebungen: (0)
- 📄 Instrumente: (0)
- 🗨️ Fragen: (146)
- 📄 Datensätze: (4)
- 📊 Variablen: (27)
- 📄 Publikationen: (0)

**Die Querverbindungen zwischen den einzelnen Ebenen werden hier nicht weiter erläutert, da sie automatisch vom System generiert werden**

---

**Studienbeschreibung**

Das DZHW-Absolventenpanel 2005 ist Teil der DZHW-Absolventenstudienreihe, in der anhand von standardisierten Befragungen Informationen zu Studium, Berufseintritt, Berufsverlauf und Weiterqualifizierung von Hochschulabsolvent(innen) erfasst werden. Das erste Absolventenpanel wurde 1989 durchgeführt, seitdem wird jeder vierte Absolvent(innen)jahrgang (Kohorte) untersucht. Für jede Absolvent(innen)kohorte werden mehrere Befragungswellen durchgeführt, wobei jede Welle in unterschiedlichem zeitlichen Abstand zum Studienabschluss stattfindet. Beim Absolventenpanel 2005 handelt es sich um die fünfte Absolvent(innen)kohorte der Studienreihe. Im Unterschied zu vorangegangenen Absolvent(innen)kohorten ist die Studienphase der Kohorte 2005 durch den Hochschulwandel im Rahmen des Bologna-Prozesses geprägt. In diesem Zuge wurden nicht nur Absolvent(innen) traditioneller Studiengänge, sondern erstmalig auch eine nennenswerte Anzahl an Bachelorabsolvent(innen)en befragt, wobei lediglich Absolvent(innen)en aus Fächern berücksichtigt wurden, für die bereits eine größere Absolvent(innen)enzahl vorlag. Zudem ist die berufliche Einstiegs- und Konsolidierungsphase der Kohorte 2005 durch den Beginn der Wirtschafts- und Finanzkrise im Jahr 2008 gekennzeichnet.

---

**Materialien zu der Studie**

Typ	Beschreibung	Titel	Dokumentensprache	Datei
Daten- und Methodenbericht	Daten- und Methodenbericht zu den Erhebungen der Absolvent(innen)kohorte 2005 (1. und 2. Befragungswelle). Version 1.0.0	DZHW-Absolventenpanel 2005	Deutsch	<a href="#">gra2005_MethodReport_de.pdf</a>
Sonstiges	Studienübersicht	Overview of the DZHW Graduate Panel 2005	Englisch	<a href="#">gra2005_Overview_en.pdf</a>

Abb. 2.6: Studienübersicht im MDM am Beispiel des Absolventenpanels 2005

#### Eine neue Studie anlegen

Nachdem ein neues Projekt erstellt wurde, können Sie nun innerhalb des Projektes eine Studie anlegen. Der primäre Weg hierfür ist das Projekt-Cockpit.

### Studie anlegen über das Projekt-Cockpit

Im Projekt-Cockpit sehen Sie im Status-Bereich die Felder zu den einzelnen Metadatenebenen. Wenn Sie unter dem Punkt Studie auf den „Neu“-Button klicken, gelangen Sie zur Eingabemaske. Machen Sie beim übernächsten Punkt *Eingabemaske* weiter.

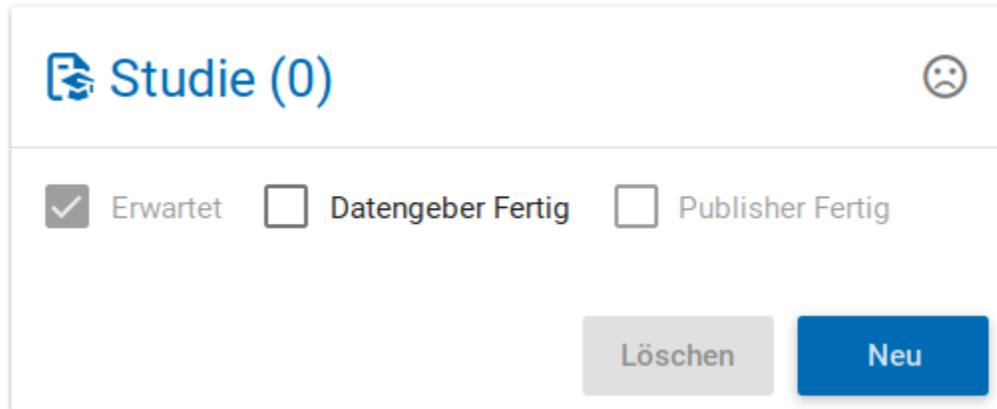


Abb. 2.7: Studie per Cockpit anlegen

### Studie anlegen über die Suche

Dazu finden Sie im Reiter „Studien“ unten rechts auf der Seite einen orangefarbenen Plus-Button (vgl. Abb. 2.8).

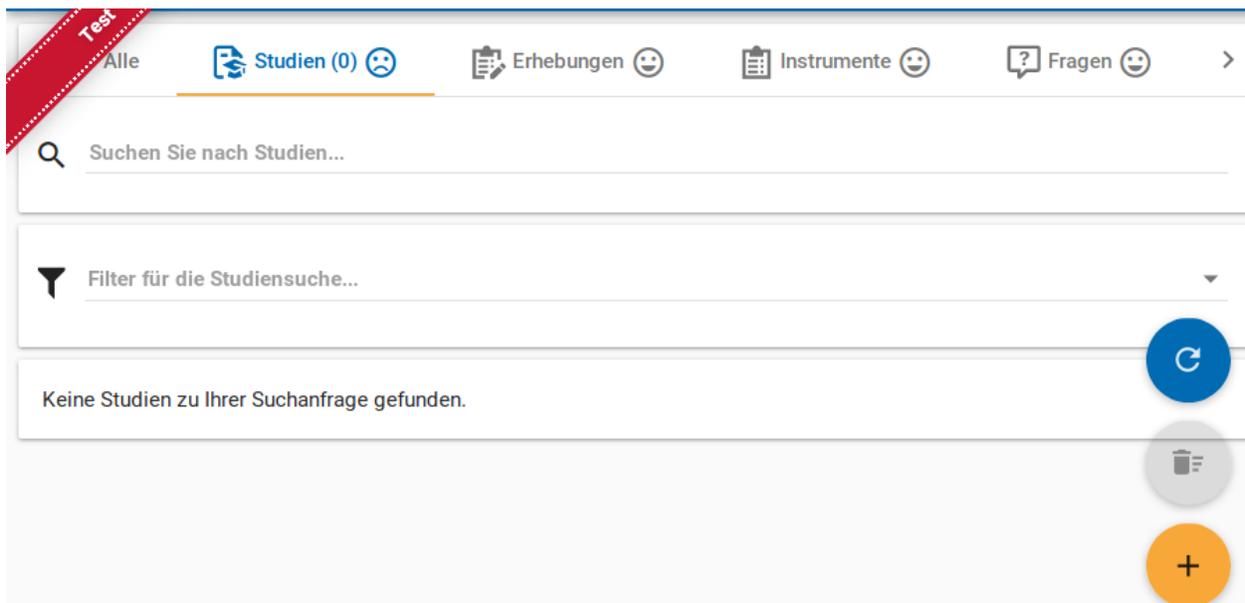


Abb. 2.8: Studie per Suche anlegen

Mit einem Klick auf den Plus-Button öffnet sich die Eingabemaske, in der Sie Ihre Informationen zur Studie ablegen können.

### Eingabemaske

Die Eingabemaske auf Studienebene besteht aus den vier Abschnitten „Details“, „Studienbeschreibung“, „Projektmitarbeiter(innen)“ sowie „Materialien zu der Studie“. Der Abschnitt „Details“ ist der umfangreichste und wird im Folgenden aufgrund der Veranschaulichung mit bereits eingetragenen Informationen dargestellt (hier beispielhaft: 21. Sozialerhebung):

Studie bearbeiten: **stu-ssy21\$** ← automatisch generierte ID

### Details

<b>Titel (auf Deutsch) *</b> 21. Sozialerhebung (2016) Geben Sie den Titel der Studie auf Deutsch ein.	<b>Anzahl der Zeichen pro Feld</b> → 25 / 2048	<b>Titel (auf Englisch) *</b> 21st Social Survey (2016) Geben Sie den Titel der Studie auf Englisch ein.	25 / 2048
<b>Studienreihe (auf Deutsch)</b> Sozialerhebung des Deutschen Studentenwerks (DSW) Geben Sie, falls vorhanden, den Namen der Studienreihe auf Deutsch ein.	×	<b>Studienreihe (auf Englisch)</b> Social Survey of the German Association for Student Services (DSW) Geben Sie, falls vorhanden, den Namen der Studienreihe auf Englisch ein.	×
<b>Erhebende Institution (auf Deutsch) *</b> Deutsches Zentrum für Hochschul- und Wissenschaftsforschung (DZHW) Geben Sie den deutschen Namen der Institution ein, die die Erhebungen durchgeführt hat.	66 / 128	<b>Erhebende Institution (auf Englisch) *</b> German Centre for Higher Education Research and Science Studies (DZHW) Geben Sie den englischen Namen der Institution ein, die die Erhebungen durchgeführt hat.	70 / 128
<b>Gefördert von (auf Deutsch) *</b> Bundesministerium für Bildung und Forschung (BMBF) Geben Sie den deutschen Namen des Geldgebers für diese Studie ein.	50 / 128	<b>Gefördert von (auf Englisch) *</b> Federal Ministry of Education and Research Geben Sie den englischen Namen des Geldgebers für diese Studie ein.	42 / 128
<b>Erhebungsdesign *</b> Querschnitt Wählen Sie das Erhebungsdesign dieser Studie aus.	<b>Drop-Down-Menü</b> →	<input type="text"/>	
<b>Anmerkungen (auf Deutsch)</b> Geben Sie zusätzliche Anmerkungen zur Studie auf Deutsch ein.	0 / 2048	<b>Anmerkungen (auf Englisch)</b> Geben Sie zusätzliche Anmerkungen zur Studie auf Englisch ein.	0 / 2048
<b>Datenverfügbarkeit *</b> In Aufbereitung Wählen Sie den Status aus, der die aktuelle Verfügbarkeit der Daten am Besten beschreibt.			

Abb. 2.9: Eingabemaske auf Studienebene, Abschnitt „Details“ am Beispiel der 21. Sozialerhebung

Nach dem Öffnen der Eingabemaske erscheint ganz oben die aus ihrem Projektnamen automatisch generierte ID für die Studienseite (s. rotes Kästchen in [Abb. 2.9](#)). Einige Felder, die Sie frei ausfüllen können, verfügen über einen Zeichenzähler, der Sie darüber informiert, wie viele Zeichen Sie dort insgesamt eintragen dürfen und wie viele Zeichen Sie bereits eingetragen haben (s. blaues Kästchen in [Abb. 2.9](#)). Außerdem finden Sie teilweise Drop-Down-Menüs vor, in denen Sie aus vorgegebenen Alternativen auswählen können (s. grünes Kästchen in [Abb. 2.9](#)).

Im zweiten Abschnitt der Eingabemaske müssen Sie eine Beschreibung Ihrer Studie sowohl auf Deutsch als auch auf Englisch eingeben. Für ein Beispiel ist im Folgenden die Beschreibung der 21. Sozialerhebung abgebildet:

Studienbeschreibung	
<p>Studienbeschreibung (auf Deutsch) *</p> <p>Die 21. Sozialerhebung ist Teil der Sozialerhebung des Deutschen Studentenwerks (DSW), einer seit 1951 bestehenden Untersuchungsreihe zur wirtschaftlichen und sozialen Situation der Studierenden in Deutschland. Es handelt sich um eine Querschnittserhebung, die meist in dreijährigem Abstand durchgeführt wird. Zum Kernbestand der Sozialerhebung gehören Fragen zum Hochschulzugang, zu Strukturmerkmalen des Studiums und Studienverlaufs, zur sozialen und wirtschaftlichen Lage (Studienfinanzierung, Lebenshaltungskosten, Erwerbstätigkeit, Wohnsituation), zu Themen des Tätigkeitsfeldes der Studentenwerke sowie zu sozio-demographischen Merkmalen.</p> <p>Geben Sie eine Beschreibung der Studie auf Deutsch ein. <span style="float: right;">750 / 2048</span></p>	<p>Studienbeschreibung (auf Englisch) *</p> <p>The 21st Social Survey is part of a survey series regarding the economic and social situation of students conducted since 1951 by the German National Association for Student Affairs (DSW) as part of their social survey. The Social Survey is a cross-sectional study which is usually carried out every three years. Key components of the study include access to higher education, structural aspects of the course and progress of studies, the economic and social situation (financing of studies, living expenses, employment, housing conditions) as well as topics in the field of activity of the German National Association for Student Affairs and socio-demographic characteristics.</p> <p>Geben Sie eine Beschreibung der Studie auf Englisch ein. <span style="float: right;">747 / 2048</span></p>

Abb. 2.10: Eingabemaske auf Studienebene, Abschnitt „Studienbeschreibung“ am Beispiel der 21. Sozialerhebung

Im dritten Abschnitt der Eingabemaske geben Sie die Mitarbeiter(innen) Ihres Projekts ein. Für die Eingabe weiterer Personen klicken Sie einfach auf den blauen Plus-Button (s. [Abb. 2.11](#)). Wenn mindestens zwei Personen eingetragen sind, erscheinen die Pfeil-Buttons als aktiv (Farbwechsel von grau zu blau). Dann können Sie die Reihenfolge der Personen ändern, indem Sie die Namen nach oben oder unten verschieben. Links neben den bereits aufgeführten Personen erscheint in jeder Zeile ein blauer Button mit einem Mülleimer-Symbol, mit dem Sie den jeweiligen Namen wieder löschen können. Mit dem orangefarbenen Save-Button unten rechts können Sie Ihre Eingaben jederzeit abspeichern. Dies müssen Sie spätestens jetzt tun, da Sie ansonsten den letzten Abschnitt der Eingabemaske („Materialien zu der Studie“) nicht bearbeiten können.

**Projektmitarbeiter(innen)**

<p>Vorname *</p> <input type="text" value="Max"/>	<p>Zweiter Vorname</p>	<p>Nachname *</p> <input type="text" value="Mustermann"/>
<p>Vorname *</p> <input type="text" value="Maxi"/>	<p>Zweiter Vorname</p>	<p>Nachname *</p> <input type="text" value="Musterfrau"/>

+
^
v

↶
💾

Abb. 2.11: Eingabemaske auf Studienebene, Abschnitt „Projektmitarbeiter(innen)“

Im vierten und letzten Abschnitt der Eingabemaske können Sie Materialien zur Studie ablegen. Dazu klicken Sie auf den blauen Plus-Button (s. [Abb. 2.12](#)), woraufhin sich ein Dialog öffnet, in dem Sie eine Datei hochladen und diese näher beschreiben können. Die hier relevanten Materialien sind momentan der deutsch- und englischsprachige Daten- und Methodenbericht (DMB) sowie eine englischsprachige *study overview*.<sup>1</sup> Die Sprache der Materialien muss nach ISO 639-1 angegeben werden. Bei den Metadaten der Materialien ist darauf zu achten die Metadaten aus den Dokumenten zu entfernen (Autor und Titel). Die Eingaben müssen Sie anschließend über den orangefarbenen Save-

<sup>1</sup> Bitte beachten Sie, die dokumenteigenen Metadaten der PDF-Dateien vorab zu löschen (vgl. [Anhänge](#)).

Button abspeichern. Mit den Pfeil-Buttons können Sie dann ggf. die Reihenfolge bereits eingegebener Materialien verändern. Wenn Sie eine geänderte Reihenfolge beibehalten möchten, müssen Sie erneut speichern.

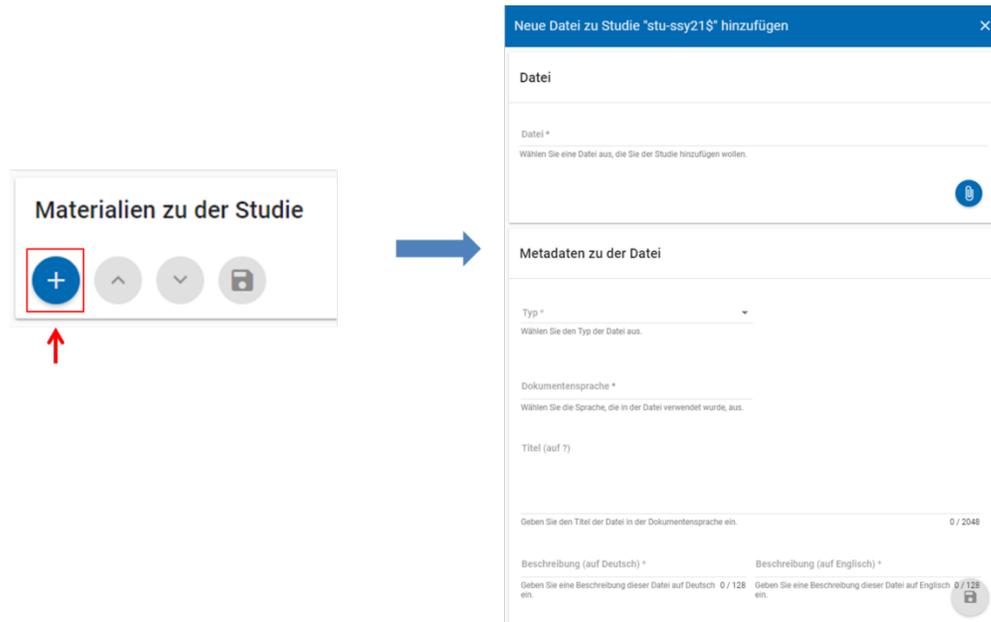


Abb. 2.12: Eingabemasken auf Studienebene, Abschnitt „Materialien zu der Studie“

### Editieren und historisieren

Falls Sie Ihre Informationen auf Studienebene nicht in einem Vorgang eingeben und hochladen können oder möchten, ist es immer möglich, dass Sie Ihre bisherigen Eingaben abspeichern und zu einem späteren Zeitpunkt weiter bearbeiten. Hierfür wird Ihnen im Reiter „Studien“ am rechten Rand neben Ihrer Studie ein Stift-Button angezeigt, über den Sie wieder in die Eingabemaske gelangen (s. Abb. 2.13).

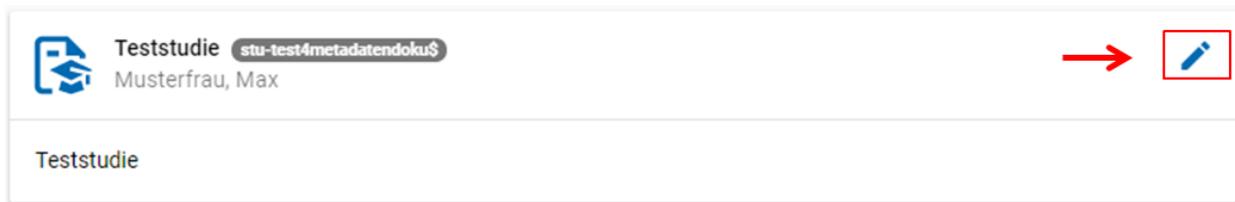


Abb. 2.13: Weitere Bearbeitung einer bereits abgespeicherten Studie

Ebenso können Sie ältere Versionen Ihrer abgespeicherten Eingaben wiederherstellen, indem Sie im Bearbeitungsmodus den Historisierungs-Button (blauer Pfeil-Button über dem Save-Button unten rechts auf der Seite) verwenden (s. Abb. 2.14).

Bei einem Klick auf den Historisierungs-Button öffnet sich ein Dialog, der die verschiedenen Versionen der Studie anzeigt (s. Abb. 2.15). Zudem sind der Name des Nutzers, der die entsprechende Version der Studie gespeichert hat, sowie das Änderungsdatum sichtbar. Durch Klicken auf die Version wird diese wiederhergestellt, aber nicht automatisch als aktuelle Version gespeichert. Dies müsste über einen Klick auf den Save-Button erfolgen. Zu beachten ist, dass Materialien zur Studie nicht historisiert werden.

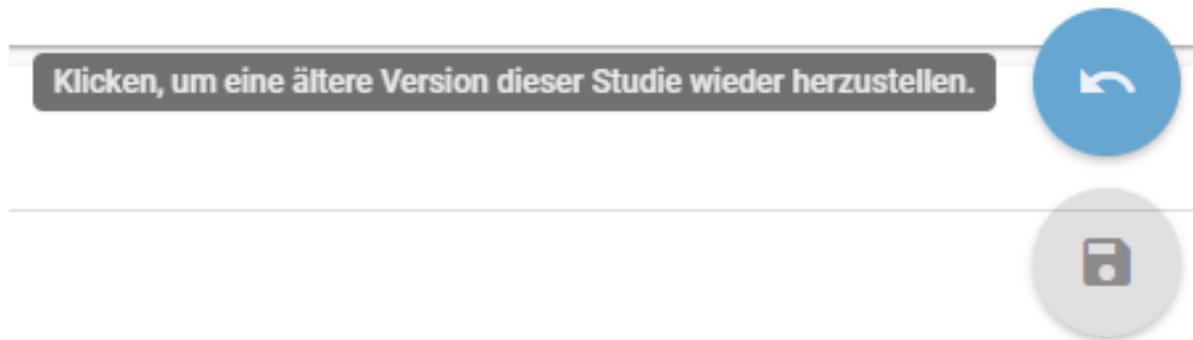


Abb. 2.14: Ältere Versionen einer Studie wiederherstellen

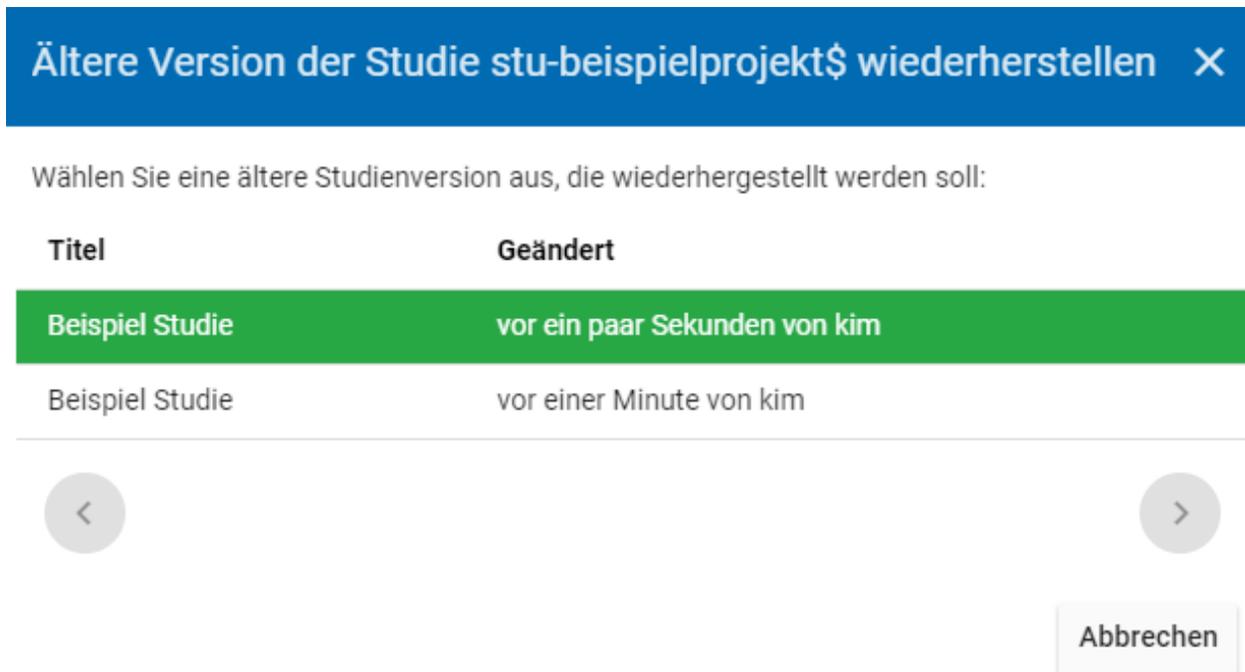


Abb. 2.15: Dialog zur Historisierung innerhalb einer Studie

## 2.4.2 Erhebungen (surveys)

### Übersicht

Mit den Informationen über die Erhebung(en), die Sie innerhalb Ihrer Studie durchgeführt haben, wird im MDM folgende Übersichtsseite erstellt:

DZHW-Absolventenpanel 2005 - erste Welle (Bachelor) sur-gra2005-sy3\$

<p><b>Details</b></p> <p><b>Grundgesamtheit:</b> Hochschulabsolvent(inn)en mit einem Bachelorabschluss, die im Wintersemester 2004/2005 oder im Sommersemester 2005 ihren ersten berufsqualifizierenden Studienabschluss an einer staatlich anerkannten Hochschule in der Bundesrepublik Deutschland erworben haben (mit Ausnahme der Absolvent(inn)en von Bundeswehrhochschulen, Verwaltungsfachhochschulen, Berufsakademien und Fernhochschulen)</p> <p><b>Erhebungsdatentyp:</b> Quantitative Daten</p> <p><b>Stichprobe:</b> Bewusste Auswahl</p> <p><b>Erhebungsmethode:</b> Standardisierte postalische Befragung</p> <p><b>Feldzeit:</b> 01.01.2006 - 18.05.2007</p> <p><b>Nettostichprobe:</b> n = 1.622</p> <p><b>Anmerkungen:</b> Die Bruttostichprobe der ersten Welle umfasst ca. 47.800 Fälle. Es ist nicht rekonstruierbar, wie viele dieser Fälle zur Gruppe der Absolvent(inn)en eines traditionellen Studiengangs und wie viele zur Gruppe der Bachelorabsolvent(inn)en gehörten.</p>	<p><b>Verbundene Objekte</b></p> <ul style="list-style-type: none"> <li>Studie: DZHW-Absolventenpanel 2005</li> <li>Instrument: P...</li> <li>Datensätze: ...</li> <li>Alle Erhebungen: ...</li> <li>Publikationen: ...</li> </ul> <div style="border: 1px dashed gray; padding: 5px; margin-top: 10px;"> <p><b>Die Querverbindungen zwischen den einzelnen Ebenen werden hier nicht weiter erläutert, da sie automatisch vom System generiert werden</b></p> </div>
<p><b>Weitere Informationen zum Rücklauf</b></p>	Empty space for the second column in the lower section

Abb. 2.16: Erhebungsübersicht im MDM am Beispiel der ersten Welle (Bachelor) im Absolventenpanel 2005

### Eine neue Erhebung anlegen

Wenn Sie eine Studie angelegt haben (vgl. Kapitel 4.1), können Sie über den Reiter „Erhebungen“ eine neue Erhebung innerhalb Ihrer Studie erstellen. Hierzu finden Sie unten rechts auf der Seite – ebenso wie bei der Studie – einen orangefarbenen Plus-Button. Wenn Sie mit dem Mauszeiger darüberfahren, erscheinen die beiden weißen Buttons, von denen Sie den Plus-Button anklicken, um die Eingabemaske zu öffnen. Bitte beachten Sie, dass Sie mehrere Erhebungen über die Eingabemaske in der richtigen Reihenfolge eingeben müssen, da die IDs beim Anlegen einer neuen Erhebung automatisch generiert werden und sich später nicht mehr verändern lassen.

### Eingabemaske

Die Eingabemaske auf Erhebungsebene besteht aus den drei Abschnitten „Details“, „Weitere Informationen zum Rücklauf“ sowie „Materialien zu der Erhebung“. Im Folgenden wird der Abschnitt „Details“ – aufgrund der Länge in zwei Teilen – dargestellt:

Beim Anlegen einer Erhebung wird automatisch die ID auf Basis des Projektnamens generiert (s. rotes Kästchen, Abb. 2.17, hier als Beispiel der 21. Sozialerhebung). Neben den bereits aus der Studienebene bekannten Funktionen gibt

Erhebung anlegen: **sur-ssy21-sy1\$** ← automatisch generierte ID

**Details**

Titel (auf Deutsch) \* Titel (auf Englisch) \*

Geben Sie den Titel der Erhebung auf Deutsch ein. 0 / 2048 Geben Sie den Titel der Erhebung auf Englisch ein. 0 / 2048

Welle \*

1

Geben Sie die Nummer der Welle an bzw. lassen Sie die Nummer auf 1 falls nicht zutreffend.

Beginn Feldzeit \* Ende Feldzeit \* ← Kalenderfunktion

Geben Sie den Beginn der Feldzeit ein. Geben Sie das Ende der Feldzeit ein.

Erhebungsmethode (auf Deutsch) \* Erhebungsmethode (auf Englisch) \*

Beschreiben Sie die Erhebungsmethode auf Deutsch. 0 / 128 Beschreiben Sie die Erhebungsmethode auf Englisch. 0 / 128

Erhebungsdatentyp \*

Wählen Sie den Erhebungsdatentyp aus.

Titel der Grundgesamtheit (auf Deutsch) \* Titel der Grundgesamtheit (auf Englisch) \*

Geben Sie einen Titel für die Grundgesamtheit auf Deutsch an. 0 / 128 Geben Sie einen Titel für die Grundgesamtheit auf Englisch an. 0 / 128

Abb. 2.17: Eingabemaske der Erhebungsebene, Abschnitt „Details“ Teil 1

es in dieser Eingabemaske zusätzlich eine Kalenderfunktion (s. blaue Kästchen, Abb. 2.17), welche die Feldzeit des Projekts erfasst und in Abb. 2.18 dargestellt ist:

Im zweiten Teil der Eingabemaske für die Erhebungsebene gibt es die Besonderheit, dass sich die Rücklaufquote automatisch ermitteln lässt (s. Abb. 2.19). Sie können den Rücklauf auch manuell eingeben. Hierbei ist zu jedoch beachten, dass sich bereits eingegebene Zahlen bei Brutto- und Netto-Stichprobe bei nicht automatisch anpassen.

Um den nächsten Abschnitt in der Eingabemaske („Weitere Informationen zum Rücklauf“<sup>2</sup>) bearbeiten zu können, müssen Sie die bisherigen Eingaben abspeichern. Dann können Sie deutschsprachige und/oder englischsprachige Grafiken zum Rücklauf entweder über den blauen Plus-Button oder per Drag & Drop hochladen und dann mit dem Save-Button speichern. Diese Grafiken dürfen im svg-, png- oder auch PDF-Format vorliegen. Über den Button mit dem Mülleimer-Symbol lassen sich hochgeladene Dateien wieder löschen (s. Abb. 2.20).

Im letzten Abschnitt der Eingabemaske können – wie auch bei der Studie – Materialien hinzugefügt werden (s. Abb. 2.21). Die Funktionsweise ist identisch zu der auf Studienebene.<sup>3</sup>

### Editieren und historisieren

Falls Sie Ihre Informationen auf Erhebungsebene nicht in einem Vorgang eingeben und hochladen können oder möchten, ist es immer möglich, dass Sie Ihre bisherigen Eingaben abspeichern und zu einem späteren Zeitpunkt weiter bearbeiten. Hierfür wird Ihnen im Reiter „Erhebungen“ am rechten Rand ein Stift-Button angezeigt, über den Sie wieder in die Eingabemaske gelangen. Außerdem finden Sie dort auch einen Button mit Mülleimer-Symbol, mit dem Sie die Erhebung komplett löschen können (s. Abb. 2.22).

Es ist außerdem möglich, ältere Versionen der bereits gespeicherten Eingaben wiederherzustellen. Im Bearbeitungsmodus gibt es auch auf der Erhebungsebene einen Historisierungs-Button, den Sie rechts unten über dem Save-Button betätigen können (s. Abb. 2.23).

Bei einem Klick auf den Historisierungs-Button öffnet sich ein Dialog, der die verschiedenen Versionen der Erhebung anzeigt (s. Abb. 2.24). Zudem sind der Name des Nutzers, der die entsprechende Version der Studie gespeichert hat, sowie das Änderungsdatum sichtbar. Durch Klicken auf die Version wird diese wiederhergestellt, aber nicht automa-

<sup>2</sup> Rücklaufgrafiken sind nur im Dokumentationsstandard der Stufe 3 gefordert. Die Erläuterungen zu den drei verschiedenen Dokumentationsstandards finden Sie in den Dokumenten „Anforderungen an Daten und Dokumentation im FDZ des DZHW“.

<sup>3</sup> Bitte beachten Sie, die dokumenteigenen Metadaten bei PDF-Dateien vorab zu löschen (vgl. *Anhänge*).

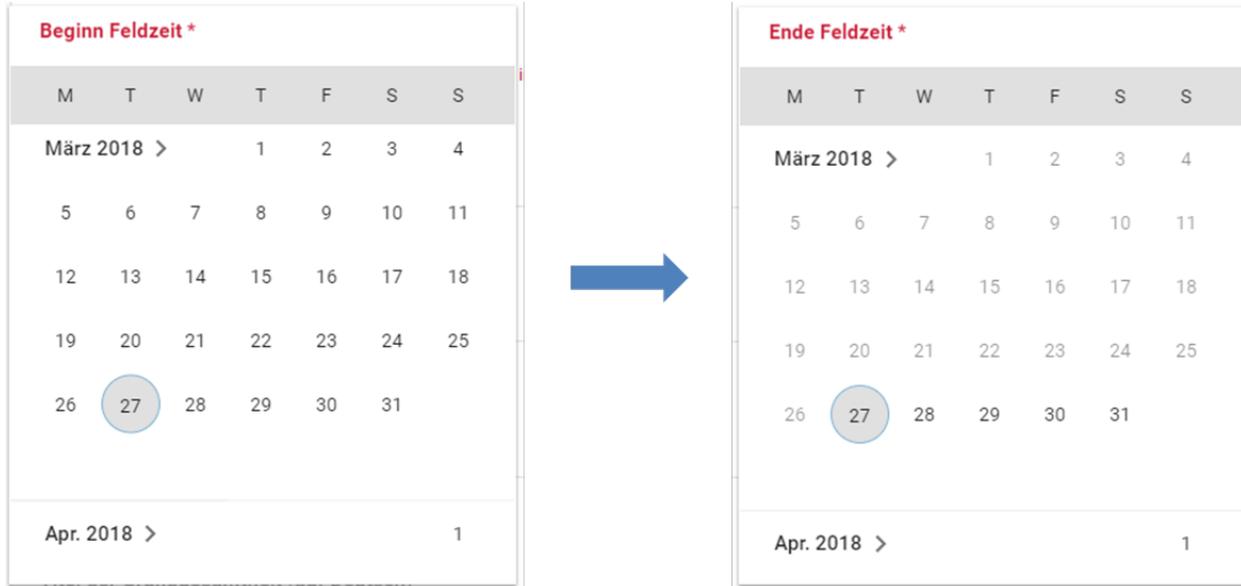


Abb. 2.18: Kalenderfunktion auf der Erhebungsebene

Beschreibung der Grundgesamtheit (auf Deutsch) *		Beschreibung der Grundgesamtheit (auf Englisch) *	
Beschreiben Sie die Grundgesamtheit auf Deutsch. 0 / 2048		Beschreiben Sie die Grundgesamtheit auf Englisch. 0 / 2048	
Stichprobe (auf Deutsch) *		Stichprobe (auf Englisch) *	
Beschreiben Sie die Stichprobe auf Deutsch. 0 / 2048		Beschreiben Sie die Stichprobe auf Englisch. 0 / 2048	
Bruttostichprobe		←	
Geben Sie die Größe Ihrer Stichprobe (brutto) an.		←	
Nettostichprobe *		←	
Geben Sie die tatsächliche Größe Ihrer Stichprobe (netto) an.		←	
Rücklaufquote (in %)	→	wird automatisch berechnet bei Angabe von Brutto- und Netto-SP	
Geben Sie die Rücklaufquote in Prozent an. 0 / 2048			
Anmerkungen (auf Deutsch)		Anmerkungen (auf Englisch)	
Geben Sie zusätzliche Anmerkungen zu der Erhebung hier auf Deutsch an. 0 / 2048		Geben Sie zusätzliche Anmerkungen zu der Erhebung hier auf Englisch an. 0 / 2048	

Abb. 2.19: Eingabemaske der Erhebungsebene, Abschnitt „Details“ Teil 2

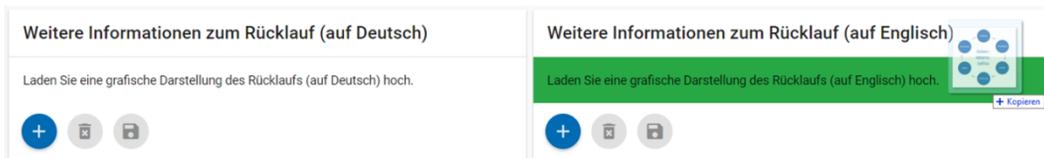


Abb. 2.20: Eingabemaske der Erhebungsebene, Abschnitt „Weitere Informationen zum Rücklauf“

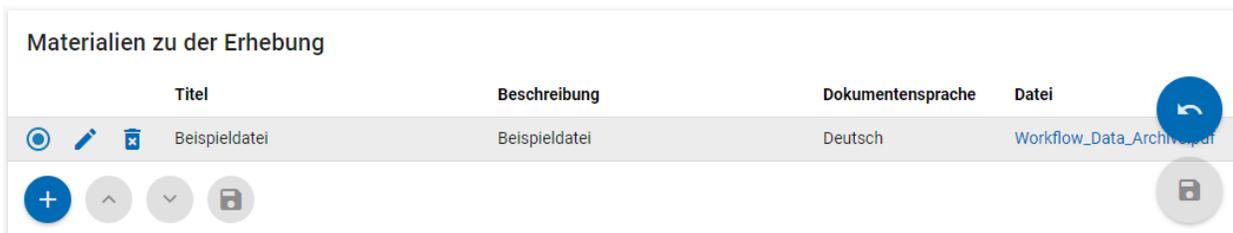


Abb. 2.21: Eingabemaske der Erhebungsebene, Abschnitt „Materialien zu der Erhebung“

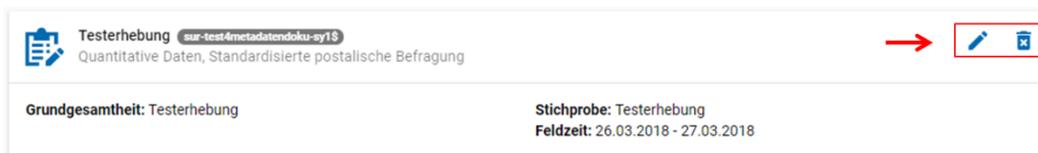


Abb. 2.22: Weitere Bearbeitung einer bereits abgespeicherten Erhebung

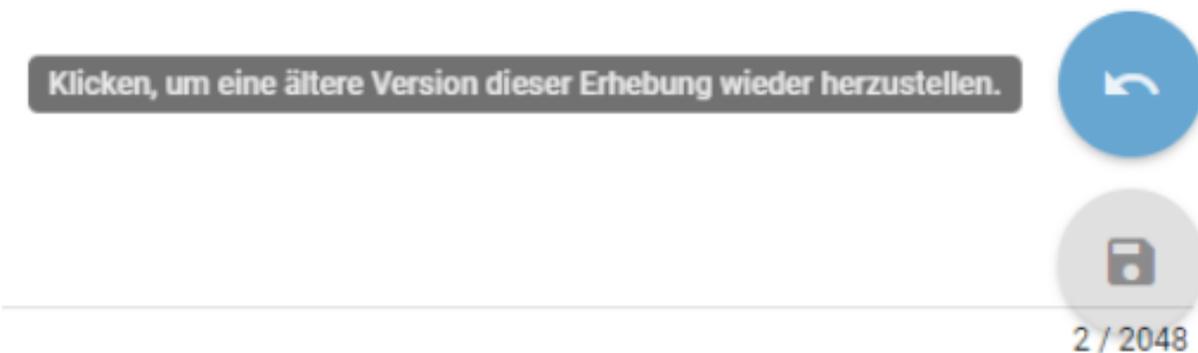


Abb. 2.23: Ältere Versionen einer Erhebung wiederherstellen

tisch als aktuelle Version gespeichert. Dies müsste über einen Klick auf den Save-Button erfolgen. Zu beachten ist, dass Materialien zur Erhebung nicht historisiert werden.

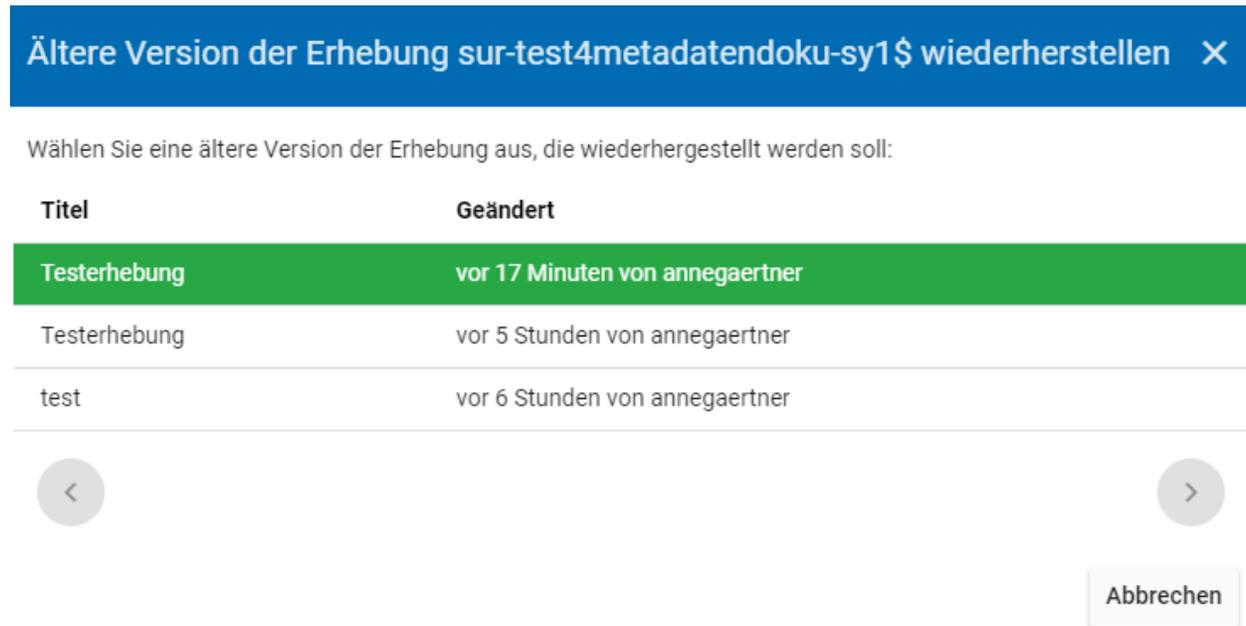


Abb. 2.24: Dialog zur Historisierung innerhalb einer Erhebung

### Prüfschritte

Der Titel der Erhebung wird zukünftig bei dalra vor einige Attribute (z.B. Referenzzeitraum) gehängt. Der Titel der Erhebung muss daher eindeutig sein und im Falle von Panelstudien die Welle enthalten.

## 2.4.3 Erhebungsinstrumente (instruments)

Als Instrument wird das Erhebungsinstrument bezeichnet (z.B. Fragebogen).

### Übersicht

Wenn Sie Informationen über Ihre Erhebungsinstrumente aufnehmen, wird folgende Übersicht im MDM erstellt:

### Eingabemaske

Erhebungsinstrumente lassen sich per Eingabemaske erfassen und editieren. Dafür darf die Studie aktuell nicht released sein. Um ein Erhebungsinstrument mittels Eingabemaske anzulegen, wird im Projektcockpit unter Instrumente auf den Neu-Button oder geht über die Suche in die Instrumentenebene und klickt auf den Plus-Button. Es öffnet sich bei beiden Herangehensweisen die Eingabemaske um ein neues Instrument anzulegen.

Die Eingabemaske besteht aus den Pflichtfeldern Beschreibung, Titel, Typ und Erhebung, sowie den nicht verpflichtenden Feldern Untertitel und Anmerkungen.

Des weiteren können weitere Materialien zum Instrument hochgeladen werden. Um weitere Materialien hochzuladen muss zunächst das Instrument abgespeichert sein. Im Anschluss muss der Plusbutton gedrückt werden, woraufhin sich ein Dialog öffnet (s. Abb. 2.27), in welchem der Anhang hochgeladen werden kann und Metadaten zur Datei eingegeben werden können. Um die Datei hochzuladen wird auf den Büroklammer-Button gedrückt und es öffnet sich ein weiterer Dialog. Alle Felder dieses Dialogs sind verpflichtend. Anschließend lässt sich der Anhang mit dem Speichern-Button (Diskettensymbol unten rechts) speichern.

Fragebogen des DZHW-Absolventenpanels 2005 - erste Welle ins-gra2005-ins1\$

**Details**

**Titel:** Zwischen Hochschule und Arbeitsmarkt  
**Typ:** PAPI

**Verbundene Objekte**

- Studie: DZHW
- Erhebungen: (0)
- Fragen: (74)
- Publikationen

Die Querverbindungen zwischen den einzelnen Ebenen werden hier nicht weiter erläutert, da sie automatisch vom System generiert werden

**Materialien zum Instrument**

Typ	Beschreibung	Dokumentensprache	Datei
Fragebogen	eingesetztes Erhebungsinstrument	Deutsch	<a href="#">gra2005_W1_Questionnaire_de.pdf</a>
Fragebogen	englische Übersetzung des deutschen Originalfragebogens	Englisch	<a href="#">gra2005_W1_Questionnaire_en.pdf</a>
Filterführungsdiagramm	grafische Darstellung des durch Filterfragen gesteuerten Befragungsablaufs	Deutsch	<a href="#">gra2005_W1_QuestionFlow_de.pdf</a>
Variablenfragebogen	Zuordnung zwischen Fragen des Erhebungsinstrumentes und Variablen des Datensatzes	Deutsch	<a href="#">gra2005_W1_VariableQuestionnaire_de.pdf</a>
Variablenfragebogen	englische Übersetzung des deutschen Variablenfragebogens	Englisch	<a href="#">gra2005_W1_VariableQuestionnaire_en.pdf</a>
Sonstige	projekteigene Referenzliste für Länder, aggr. Länder (angelehnt an NEPS, Anpassungen für Europa) und Weltregionen (Ausland)	Deutsch	<a href="#">cl-dzhw-1.xlsx</a>
Sonstige	projekteigene Codierliste für Studienabschlüsse	Deutsch	<a href="#">cl-dzhw-2.xlsx</a>

Abb. 2.25: Instrumentenübersicht im MDM am Beispiel des Fragebogens der ersten Welle im Absolventenpanel 2005

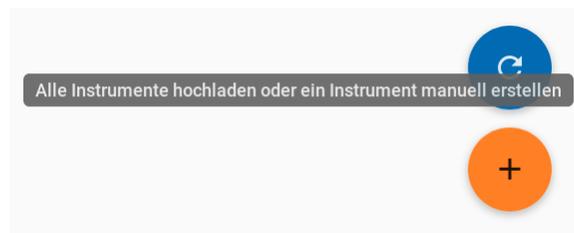


Abb. 2.26: Plusbutton

Abb. 2.27: Instrumente Anhang

Zu den möglichen Anhängen zählen z. B. Fragebögen, Variablenfragebögen sowie Filterführungsdiagramme<sup>4</sup>. Diese müssen als PDF-Dateien vorliegen.<sup>5</sup> Außerdem können an dieser Stelle Codierlisten, welche als Excel-Tabelle vorliegen müssen, erfasst werden.

Sollte es Erhebungsinstrumente geben, welche in einer anderen Sprache als deutsch oder englisch existieren, werden diese nur als Attachment und nicht auf Variablenebene bereitgestellt.

## 2.4.4 Fragen

Sollten Sie mit uns die Eingabe von Frage-Metadaten vereinbart haben, sprechen Sie uns bitte darauf an. Wir erläutern Ihnen dann den Prozess. Metadaten für Fragen müssen als JSON und png Dateien vorliegen. Sollte eine Umfrage mit Zofar durchgeführt worden sein, bitten wir Sie uns die Metadaten zukommen zu lassen. Die Fragen-Metadaten werden von uns hochgeladen.

## 2.4.5 Datensätze (dataSets)

**Übersicht** Für die Dokumentation der Datensätze werden die „Master“(AIP)-Datensätze (siehe Zwiebelmodell) genutzt. Diese Datensätze sind die größte mögliche Vereinheitlichung eines Datensatzes, also keine Teilpopulation oder Teilmenge von Variablen eines Datensatz. Datensätze die sich als Teilmenge eines „Master“-Datensatzes abbilden lassen werden über die Subdatensätzen (SubDataSets) dokumentiert. Mit Subdatensätzen sind solche gemeint, die Sie nach einer Anonymisierung Ihrer Daten erhalten. Sie können mehrere Stufen der Anonymisierung verwenden, wobei jede Stufe einen eigenen Zugangsweg zu den anonymisierten Daten mit sich bringt. Für jeden Zugangsweg wird dann ein eigener Subdatensatz erstellt.

<sup>4</sup> Filterführungsdiagramme sind erst ab der 2. Dokumentationsstufe gefordert. Die Erläuterungen zu den drei verschiedenen Dokumentationsstandards finden Sie in den Dokumenten „Anforderungen an Daten und Dokumentation im FDZ des DZHW“.

<sup>5</sup> Bitte beachten Sie, die dokumenteigenen Metadaten der PDF-Dateien vorab zu löschen (vgl. *Anhänge*).

Mit den Informationen über die Datensätze, welche Sie aus den Daten Ihrer Studie erstellt haben, wird für jeden dieser Datensätze folgende Übersicht im MDM angezeigt:

Personendatensatz DZHW-Absolventenpanel 2005 (Bachelor) dat-gra2005-ds3\$

**Details**

Typ: Personendatensatz  
Format: breit

**Verbundene Objekte**

- Studie: DZHW-Absolventenpanel 2005 (Bachelor)
- Erhebungen: (2)
- Variablen: 135
- Datensätze: 1
- Publikationen: Nicht vorhanden!

Die Querverbindungen zwischen den einzelnen Ebenen werden hier nicht weiter erläutert, da sie automatisch vom System generiert werden

**Verfügbare Subdatensätze**

Zugangsweg	Beschreibung	Name	Analysierbare Variablen	Fälle	Zitation
download-suf	Kann nach Abschluss eines Datennutzungsvertrags heruntergeladen werden	gra2005_ba_p_d_1-0-0	1.215	1.622	””
remote-desktop-suf	Kann über Remote Desktop genutzt werden	gra2005_ba_p_r_1-0-0	1.283	1.622	””
onsite-suf	Ist am Gastwissenschaftlerarbeitsplatz im DZHW in Hannover verfügbar	gra2005_ba_p_o_1-0-0	1.354	1.622	””

Abb. 2.28: Datensatzübersicht im MDM am Beispiel des Personendatensatzes (Bachelor) im Absolventenpanel 2005

### Eingabemaske

Datensätze lassen sich auch per Eingabemaske anlegen und editieren. Hierfür muss man auf den Reiter Datensätze klicken (Abb. 2.1), anschließend auf das Plusymbol (Abb. 2.29) in der unteren rechten Ecke klicken. Anschließend öffnet sich die Eingabemaske (siehe Abb. 2.30).

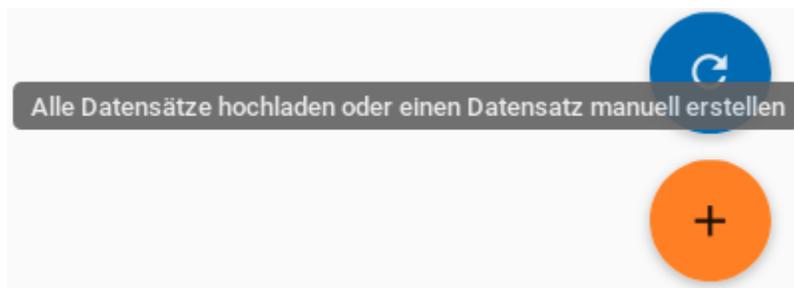


Abb. 2.29: Neuen Datensatz hinzufügen.

Die mit \* markierten Felder sind verpflichtend. Die verknüpften Erhebungen werden nach einem Klick in das Feld „Erhebungen“ automatisch vorgeschlagen und können per Klick ausgewählt werden. Im Anschluss werden die Subdatensätze per Eingabemaske auf der selben Seite eingegeben. Weitere Subdatensätze können per Klick auf das Plusymbol hinzugefügt werden. Nachdem gespeichert wurde, lassen sich weitere Materialien zum Datensatz hinzufügen.

Wenn Sie zusätzliche Materialien (z.B. Variablen-Dokumentation) auf Ebene der Datensätze haben, können Sie diese hinzufügen. Hierfür muss zunächst der Datensatz angelegt sein. Anschließend wird in der unteren linken Ecke auf den blauen Plus-Button geklickt.<sup>6</sup>

<sup>6</sup> Bitte beachten Sie, die dokumenteneigenen Metadaten bei PDF-Dateien vorab zu löschen (vgl. *Anhänge*).

**Datensatz anlegen:** dat-cmp2014-ds3\$

### Details

**Beschreibung (auf Deutsch) \*** 0 / 2048 **Beschreibung (auf Englisch) \*** 0 / 2048  
 Geben Sie eine kurze Beschreibung für den Datensatz auf Deutsch ein. Geben Sie eine kurze Beschreibung für den Datensatz auf Englisch ein.

**Format** ▼  
 Wählen Sie das Datensatzformat aus.

**Typ \*** ▼  
 Geben Sie an ob es sich um einen Personen- oder Episodendatensatz handelt.

**Erhebungen \***  
 Erhebungen suchen...  
 Wählen Sie die Erhebungen aus, aus denen dieser Datensatz resultiert.

**Anmerkungen (auf Deutsch)** 0 / 2048 **Anmerkungen (auf Englisch)** 0 / 2048  
 Geben Sie zusätzliche Anmerkungen zu dem Datensatz hier auf Deutsch an. Geben Sie zusätzliche Anmerkungen zu dem Datensatz hier auf Englisch an.

### Verfügbare Subdatensätze

Name *	Zugangsweg *	Anzahl Fälle/Episoden *
Geben Sie den (Datei-)namen des Subdatensatzes an.	0 / 32 Auf welchem Weg steht der Subdatensatz zur Verfügung?	Wieviele Fälle bzw. Episoden enthält der Subdatensatz?

**Beschreibung (auf Deutsch) \*** 0 / 512 **Beschreibung (auf Englisch) \*** 0 / 512  
 Geben Sie eine kurze Beschreibung für den Subdatensatz auf Deutsch ein. Geben Sie eine kurze Beschreibung für den Subdatensatz auf Englisch ein.

Abb. 2.30: Eingabemaske der Datensebene.

**Neue Datei zu Datensatz "dat-cmp2014-ds2\$" hinzufügen** ×

### Datei

**Datei \***  
 Wählen Sie eine Datei aus, die Sie dem Datensatz hinzufügen wollen.

### Metadaten zu der Datei

**Dokumentensprache \***  
 Wählen Sie die Sprache, die in der Datei verwendet wurde, aus.

**Titel (auf ?) \***  
 Geben Sie den Titel der Datei in der Dokumentensprache ein. 0 / 2048

**Beschreibung (auf Deutsch) \*** 0 / 512 **Beschreibung (auf Englisch) \*** 0 / 512  
 Geben Sie eine Beschreibung dieser Datei auf Deutsch ein. Geben Sie eine Beschreibung dieser Datei auf Englisch ein.

Abb. 2.31: Datensatz Anhänge

Es öffnet sich ein Fenster (siehe Abb. 2.31) in dem Sie eine Datei hochladen können und Metadaten zur Datei angeben müssen.

## 2.4.6 Variablen

Sollten Sie die Bereitstellung von Variablenmetadaten mit uns vereinbart haben, sprechen Sie uns bitte an. Wir erläutern Ihnen dann die notwendigen Schritte.

## 2.4.7 Publikationen (relatedPublications)

### Überblick

Auf der Ebene der Publikationen werden wissenschaftliche Veröffentlichungen, welche auf Grundlage von Daten Ihres Projekts verfasst worden sind, erfasst. Die Informationen, die Sie im Hinblick auf Ihre Publikationen abgeben, werden im MDM für jede Veröffentlichung wie folgt dargestellt:

The screenshot shows a publication entry in the MDM system. The title is "Which mechanisms explain monetary returns to international student mobility?" with a reference code "pub-Kratz.2016\$". The interface is split into two main sections: "Details" on the left and "Verbundene Objekte" (Related Objects) on the right. The "Details" section lists the author(s) as Kratz, Fabian; Netz, Nicolai, the year as 2016, and provides the full reference and DOI (10.1080/03075079.2016.1172307). The "Verbundene Objekte" section lists various data objects: "Studie: DZHW-Ab... (checked)", "Erhebungen: Nicht erf...", "Instrumente: Nicht erf...", "Fragen: Nicht erfasst", "Datensätze: Nicht erf...", and "Variablen: Nicht erfasst". A red dashed box highlights the text: "Die Querverbindungen zwischen den einzelnen Ebenen werden hier nicht weiter erläutert, da sie automatisch vom System generiert werden".

Abb. 2.32: Publikationsübersicht im MDM am Beispiel einer Veröffentlichung, welche im Rahmen des Absolventenpanels 2005 verfasst wurde

Wenn Sie Publikationen zu Ihren Daten abgeben möchten, senden Sie dem FDZ per Mail die PDF-Datei Ihrer Publikation sowie den dazugehörigen Zitationshinweis zu.

## 2.5 Projekte freigeben

Wenn Sie alle Metadaten ausgefüllt bzw. ans FDZ gesendet haben, melden Sie sich beim FDZ mit dem Hinweis, dass Sie Ihre Daten nicht weiter editieren möchten. Das FDZ nimmt ihre Daten dann in die sogenannte Release-Pipeline auf. Die finale Freigabe erfolgt dann über einen dafür benannten Mitarbeiter des FDZ, den Release-Manager. Mit der Freigabe sind Ihre Metadaten und damit auch Ihre Datenprodukte für alle öffentlichen Nutzer des Systems verfügbar.

### 2.5.1 Checkliste für Abgabe der Metadaten

Vor Abgabe bzw. dem Hochladen der Daten sind folgende Punkte zu überprüfen:

Ordnerstruktur und Dateinamen sind unverändert

Falls nötig: Excel-Dateien sind richtig und vollständig ausgefüllt

- verpflichtende Felder sind ausgefüllt
- es sind keine Fehlermeldungen vorhanden

- die Daten wurden auf Richtigkeit überprüft

Metadaten sind aus PDF-Dokumenten entfernt (vgl. *Anhänge*)

Nicht benötigte Zeilen entfernt (falls Fragen oder Variablen abgegeben werden)

- questions.xlsx: löschen bis Zeile 2000
- variables.xlsx: löschen bis Zeile 2000

Dateien sind richtig abgegeben worden

In das Metadatenystem eingegeben

- Studie (study)
- Erhebungen (surveys)

In der Ordnerstruktur abgelegt

- questions.xlsx
- variables.xlsx

---

## FDZ-MitarbeiterIn (Publisher, Developer)

---

### 3.1 ID-Vergabe

Manuell vergebene ids (DAP-ids) müssen in folgender [Tabelle](#) festgehalten werden.

#### 3.1.1 Logik

Metadaten	Id-Generierung
DataAcquisitionProject (DAP-id)	wird manuell vergeben, siehe Tabelle oben Übersicht über alle <a href="#">Projekte</a> des DZ-HW
Study	„stu-“ + DAP-id + „,\$“
Survey	„sur-“ + DAP-id + „-“ + „sy“ + survey.number + „,\$“
DataSet	„dat-“ + DAP-id + „-“ + „ds“ + dataSet.number + „,\$“
Variable	„var-“ + DAP-id + „-“ + „ds“ + variable.dataSetNumber + „-“ + variable name + „,\$“
Instrument	„ins-“ + DAP-id + „-“ + „ins“ + number + „,\$“
Question	„que-“ + DAP-id + „-ins“ + instrumentNumber + „-“ + number + „,\$“
relatedPublication	„pub-“ + citaviId + „,\$“

### 3.1.2 Beispiele am Absolventenpanel 2005

Metadaten	Id
DataAcquisitionProject	gra2005
Study	stu-gra2005\$
Survey	sur-gra2005-sy1\$
DataSet	dat-gra2005-ds1\$
Variable	var-gra2005-ds1-stu01\$
Instrument	ins-gra2005-ins1\$
Question	que-gra2005-ins1-1.1\$
BibliographicalReference	pub-Meier.2010\$

## 3.2 Neues Projekt anlegen

Im Folgenden wird zunächst der Prozess zur Erstellung eines Projektes und dann die Vergabelogik der DataAcquisitionProject-ID erklärt.

### 3.2.1 Prozess

Nach erfolgreicher Registrierung können Sie ihr Projekt im MDM anlegen, indem Sie sich in der Sidebar links anmelden:



Abb. 3.1: Anmeldung im MDM

Nach erfolgreicher Anmeldung erscheint in der Sidebar das folgende Feld:

Über den Plus-Button können *Publisher* ein neues Projekt anlegen (vgl. Abb. 3.3). Als Projektname müssen diese eine bestimmte ID angeben, welche das FDZ zuvor speziell für Ihr Projekt vergeben hat und Ihnen mitteilen muss (z. B. „gra2005“ für das Absolventenpanel 2005).

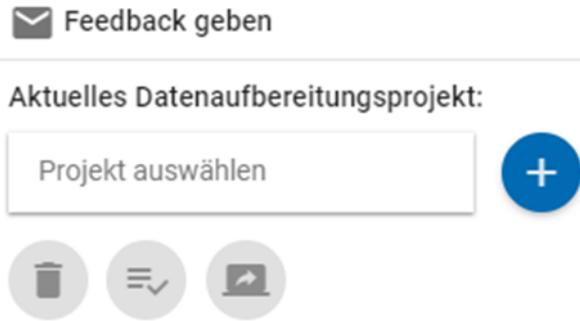


Abb. 3.2: Bereich für Verwaltung von Projekten im MDM

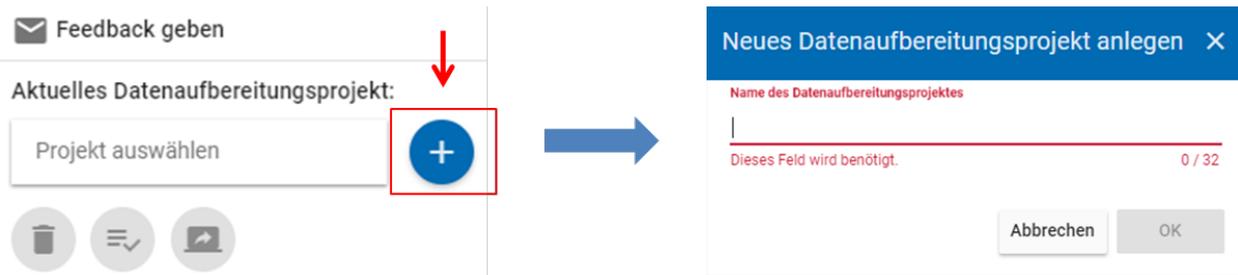


Abb. 3.3: Ein neues Projekt im MDM anlegen

### 3.2.2 DataAcquisitionProject-ID

- **drei Kleinbuchstaben:** Um die internationale Nutzbarkeit der Daten zu erleichtern, ergeben sich die drei Kleinbuchstaben aus der englischsprachigen Abkürzung des Projektes. Es kann Ausnahmen geben, wenn z.B. bestimmte Projekte ein besonders griffiges Kürzel haben wie z.B. bei Libertas 2016 - lib2016.
- **(zwei bis) vier Ziffern:** Die Ziffern sind in der Regel die vier Ziffern des Jahres, das die Zugehörigkeit zur Stichprobe definiert, z. B. der Abschluss des Studiums (z. B. im (Prüfungs-)Jahr 2005 oder der Erwerb der Hochschulzugangsberechtigung im Jahr 2008). Davon kann in begründeten Ausnahmefällen abgewichen werden.
- Beispielsweise ist die Sozialerhebung deutlich stärker unter der jeweiligen Nummer der Studie als unter dem Jahr, das die Zugehörigkeit zur Stichprobe definiert, bekannt, so dass beispielsweise für die 19. Sozialerhebung aus dem Jahr 2009 die Ziffern 19 (und nicht 2009) vergeben werden.
- Bei Befragungen, die sich auf mehrere Jahreszahlen beziehen, kann eine andere eindeutige Jahreszahl verwendet werden. Beispielsweise gehören in der KomPaed-Befragung die zuletzt aktiven Panelteilnehmer mehrerer Kohorten zur Stichprobe. Hier werden die vier Ziffern des Erhebungsjahres (2014) genutzt.

Für jedes weitere Objekt wird später ebenfalls eine ID generiert, die die DAP-id enthält und sich nach der folgenden Logik zusammensetzt:

Metadaten	Id-Generierung
Study	„stu-“ + DAP-id + „,\$“
Survey	„sur-“ + DAP-id + „-“ + „sy“ + survey.number + „,\$“
DataSet	„dat-“ + DAP-id + „-“ + „ds“ + dataSet.number + „,\$“
Variable	„var-“ + DAP-id + „-“ + „ds“ + variable.dataSetNumber + „-“ + variable name + „,\$“
Instrument	„ins-“ + DAP-id + „-“ + „ins“ + number + „,\$“
Question	„que-“ + DAP-id + „-ins“ + instrumentNumber + „-“ + number + „,\$“
relatedPublication	„pub-“ + citaviId + „,\$“

Das Projekt ist dadurch im MDM angelegt, aber noch nicht automatisch freigegeben. Solange das Projekt noch nicht freigegeben wurde, können Nutzer der Gruppe *publisher* es jederzeit über den Mülleimer-Button ganz links (vgl. Abb. 3.3) wieder löschen.

## 3.3 Bilderfassung aus PDF Dateien

### 3.3.1 Voraussetzungen:

Fragebögen, die als eine pdf Datei vorliegen sollen gemäß der nächsten Schritte bearbeitet werden.

Liegt zu dem Fragebogen, den es von Ihnen zu bearbeiten gilt, eine äquivalente Excel Datei vor, so wäre es ratsam diese als Grundlage für die weitere Arbeit zu verwenden. Hierbei ist es notwendig, dass jeder aufgeführten Frage in der Excel Datei eine finale Bilddatei gegenüber steht.

Für die Bilderfassung aus einer pdf Datei kann sowohl der Adobe Acrobat XI Pro als auch der Adobe Acrobat DC genutzt werden. In dieser Anleitung soll es aber im Detail um das Arbeiten mit dem Adobe Acrobat XI Pro gehen, während die Vorgehensweise mit dem Adobe Acrobat DC nur sehr knapp dargestellt wird.

### 3.3.2 Adobe Acrobat DC (knapp):

- Schritt 1: Öffnen der Datei in Adobe Acrobat DC
- Schritt 2: Klicken Sie mittels der linken Maustaste auf „Werkzeuge“. Unter dieser Schlagzeile öffnet sich eine weitere Leiste. Wählen Sie nun „pdf Datei bearbeiten“ und anschließend „Seite beschneiden“ aus. Nun sind Sie in der Lage einzelne Fragen auszuschneiden.
- Schritt 3.1: Nachdem Sie alle Fragen mittels der in Schritt 2 genannten Tools ausgeschnitten haben, können Sie diese nun Extrahieren und speichern. Dies erfolgt mit dem Werkzeug „Seiten verwalten“. Wählen Sie auch dies durch einen Linksklick mittels der Maus aus. Hier sind zwei Wege denkbar um Fortzufahren.
- Schritt 3.2.1 Wählen Sie die jeweilige Seite, die es zu extrahieren gilt aus und ziehen Sie in einen von Ihnen gewählten Ordner (durch stetiges Festhalten der linken Maustaste).

oder

- 3.2.2 Wählen Sie in der rechten Leiste (Werkzeuge) „exportieren“, bestimmen Sie die Seite, die exportiert werden soll und speichern Sie es in einem von Ihnen gewählten Ordner.

Achten Sie darauf, dass der jeweilige Dateiname der Fragennummer entspricht! (zur Not auch nochmal mit Excel Tabelle abgleichen, falls vorhanden)

### 3.3.3 Adobe Acrobat XI Pro

- Schritt 1: Öffnen Sie die Datei mit dem Adobe Acrobat XI Pro.
- Schritt 2.1.1: Es erscheint eine obere Leiste in der rechts der Begriff „Werkzeuge“ zu finden ist. Wählen Sie diesen Befehl durch einen Linksklick der Maus aus (siehe Abb. 1.1).

*Abb. 1.1*

Nun sollte sich ein weiterer Spaltenblock öffnen (siehe Abb. 1.2).

*Abb. 1.2*

An dieser Stelle klicken Sie auf „Seiten“. Auch hier eröffnet sich ein weiterer Spaltenblock mit ergänzenden Werkzeugen. Unter der Rubrik „Seiten manipulieren“ finden Sie die Instrumente „Zuschneiden“ sowie „Extrahieren“. Diese sind für unser weiteres Vorgehen wesentlich (siehe Abb. 1.3)

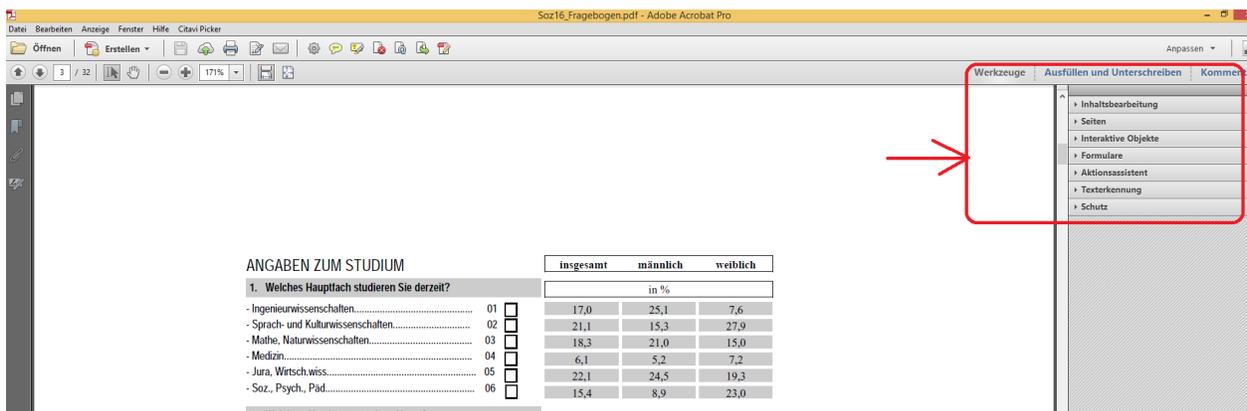
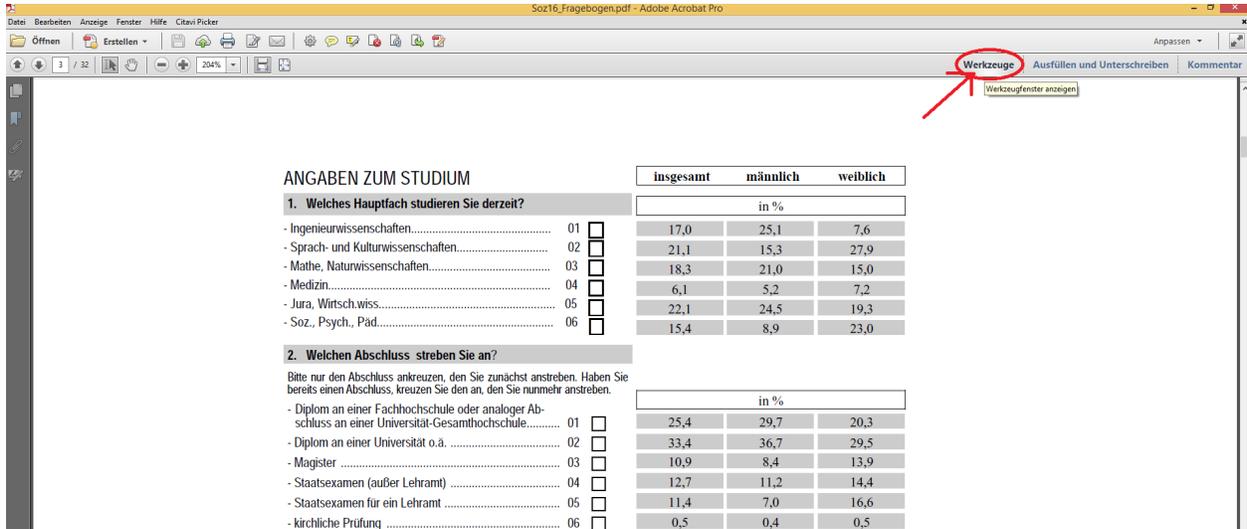
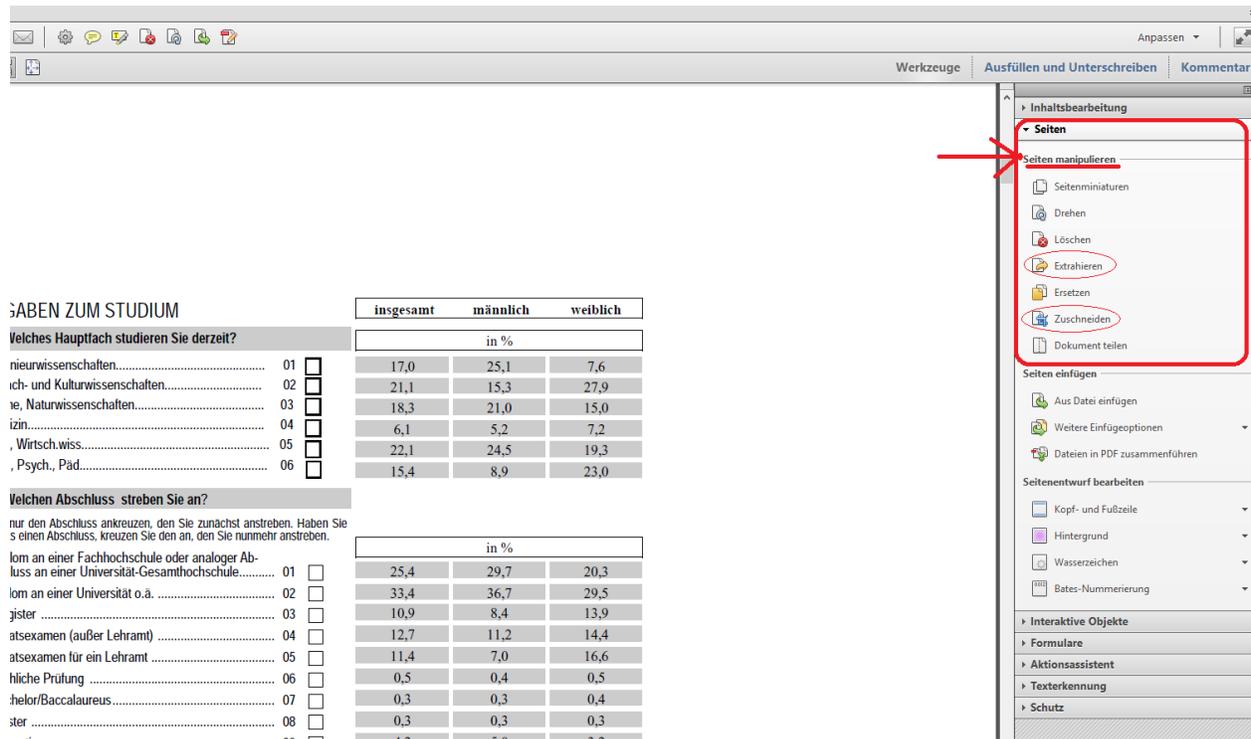
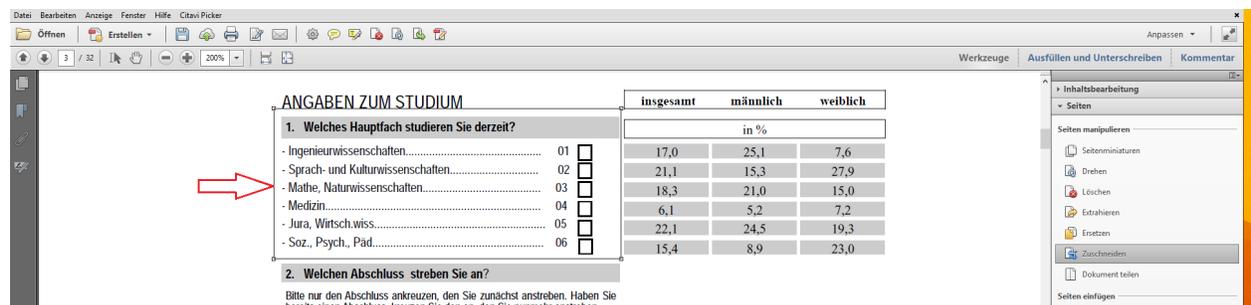


Abb. 1.3



Um nun mit dem Zu- und Ausschneiden der einzelnen Fragen zu beginnen, wählen Sie den Befehl „Zuschneiden“ aus. Klicken sie durch Betätigen der linken Maustaste links oberhalb der Frage, die es auszuschneiden gilt und markieren Sie durch stetiges Festhalten der linken Maustaste alle relevanten Elemente der Frage. Dabei erscheint ein schwarz umrandetes Viereck. In diesem müssen alle Bestandteile der Frage enthalten sein!(siehe Abb. 1.4)

Abb. 1.4



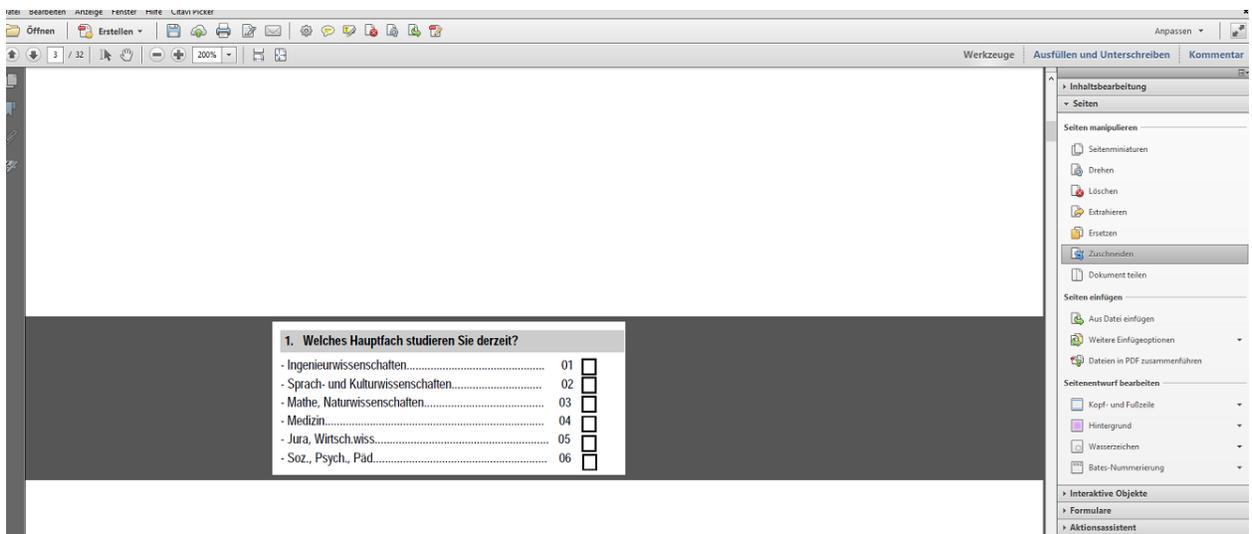
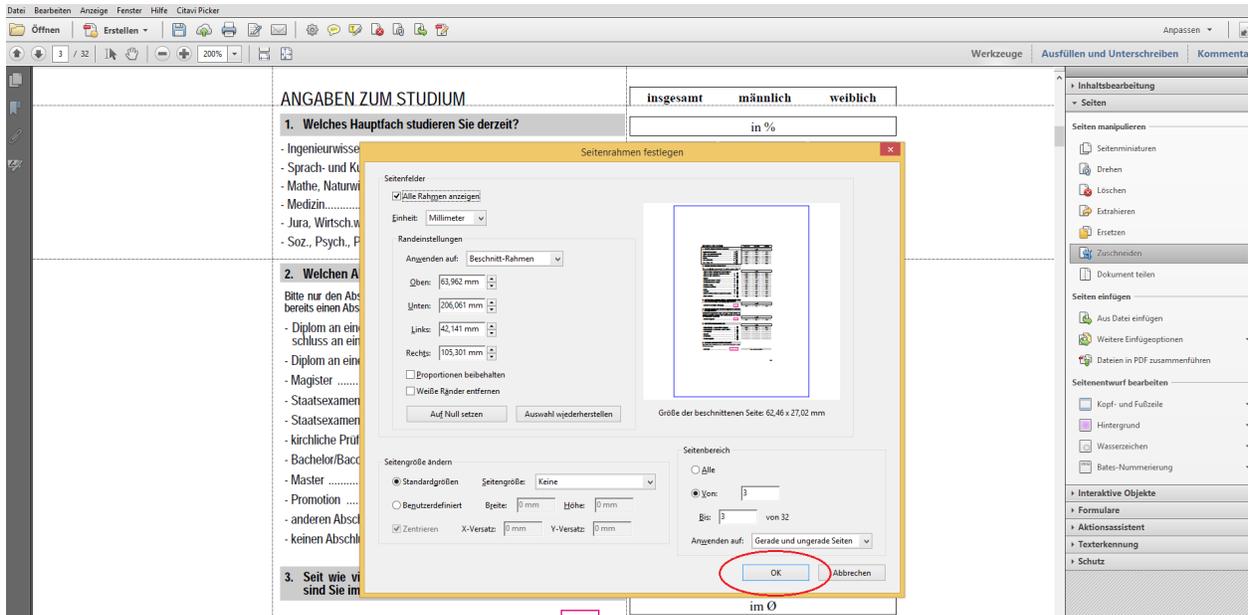
Anschließend betätigen Sie die Enter Taste ihrer Tastatur. Dabei eröffnet sich eine Maske (siehe Abb. 1.5) welche Sie mit „OK“ bestätigen.

Abb. 1.5

Nachdem Sie die Maske durch das Bestätigen des „OK“ Buttons geschlossen haben, sollte auf ihrem Bildschirm nun eine ähnliche Darstellung wie in Abb. 1.6 erscheinen.

Abb. 1.6

**Hinweis:** Dies ist der einfachste Fall der in Schritt 2.1 beschrieben wird. Es kann aber durchaus vorkommen, dass eine Frage sich über zwei Seiten erstreckt, was das Ausschneiden der Frage erschwert. Hierbei ist erst einmal das

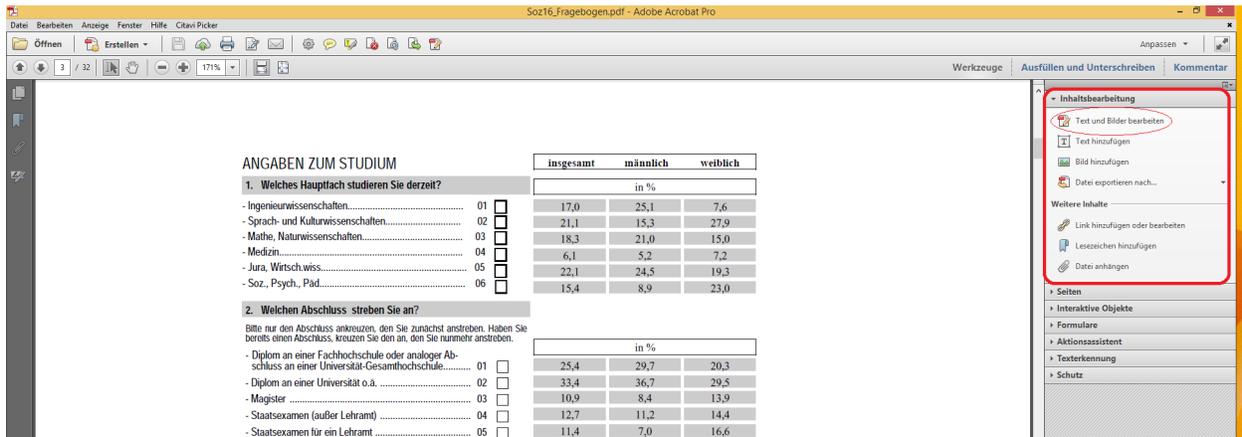


Zusammenführen der Frage auf einer gemeinsamen Seite notwendig, bevor man mit dem Zuschneiden beginnen kann. Wie dies zum Beispiel aussehen kann ist in Abb. 1.7 zu sehen.

Abb. 1.7 .. figure:: ./\_static/pdf\_extraction\_1\_7.png

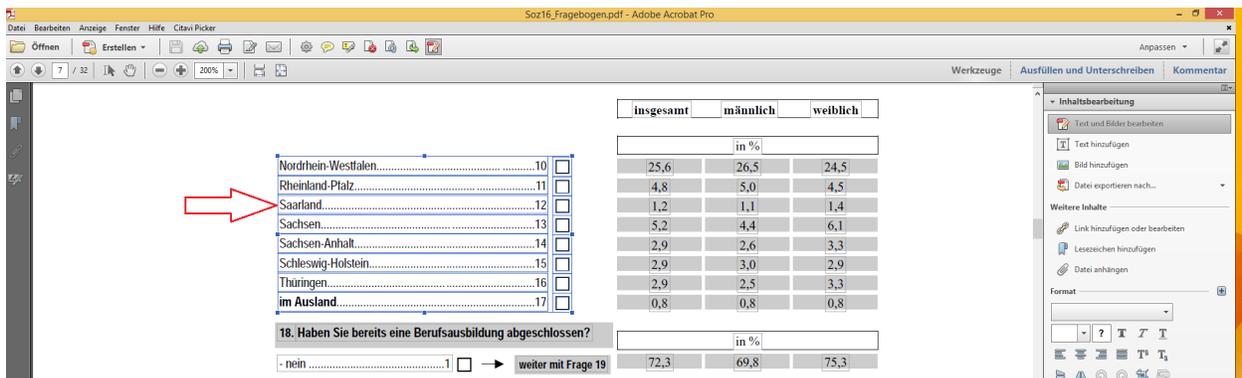
- Schritt 2.1.2 Liegt diese Ausgangssituation vor, so ist es notwendig Werkzeuge unter dem Begriff „Inhaltsbearbeitung“ zu verwenden. Dies ist ebenfalls in der rechten Leiste unter Werkzeuge zu finden (Siehe Abb. 1.8)

Abb. 1.8



In diesem Zusammenhang ist lediglich der Befehl „Text und Bilder bearbeiten“ relevant. Äquivalent zum Vorgang des Zuschneidens, klicken wir auch hier per Linksklick neben die auszuscheidenden Elemente und ziehen durch Halten der linken Maustaste ein Viereck um diese. Alle ausgewählten Elemente müssen nun blau umrandet sein (siehe Abb. 1.9).

Abb. 1.9



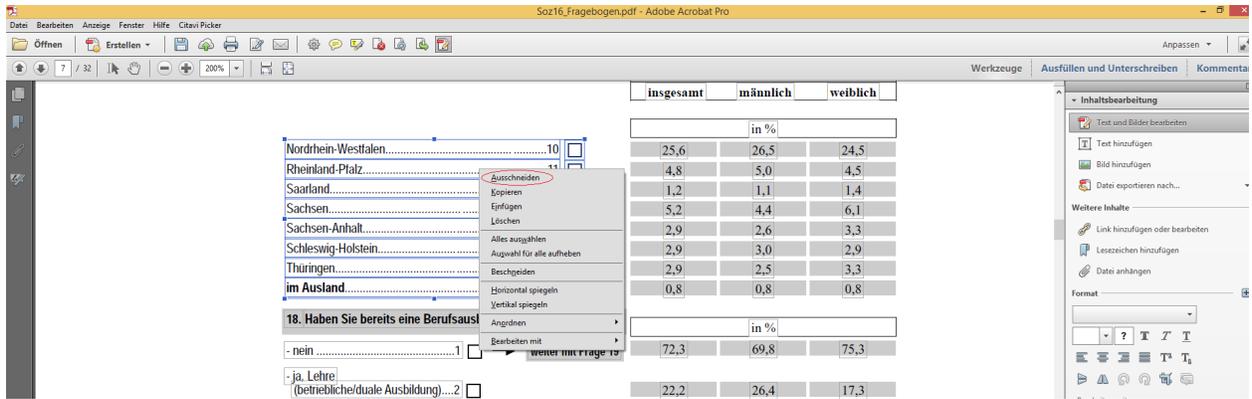
Als nächstes klickt man mittels der rechten Maustaste in das Viereck und wählt schließlich „ausschneiden“ aus (siehe Abb. 1.10).

Abb. 1.10

Um die Frage nun auf einer Seite zusammenzuführen, klicken Sie nun mit der rechten Maustaste auf die Seite, auf der die soeben ausgeschnittenen Elemente hinzugefügt werden sollen. Dabei eröffnet sich ein Spaltenblock in dem Sie bitte „einfügen“ auswählen (siehe Abb. 1.11).

Abb. 1.11

Schließlich können Sie die Frageelemente so verschieben, dass die Frage vollständig auf einer gemeinsamen Seite vorliegt. Verschieben Sie die Frageelemente, indem Sie den Mauszeiger auf den äußersten blauen Rand bewegen, wodurch dann ein Kreuz aufzeigen müsste, welches an jedem Ende mit Pfeilen versehen ist. Klicken sie mittels

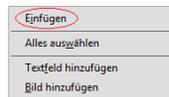


- Sonderprüfung für Begabte .....	7	<input type="checkbox"/>	0,2	0,2	0,2
- als besonders befähigte(r) Berufstätige(r) .....	8	<input type="checkbox"/>	0,3	0,3	0,4
- sonstiger Weg .....	9	<input type="checkbox"/>	1,3	1,3	1,3

**17. Wo haben Sie die Hochschulzugangsberechtigung erworben?**

			in %		
Baden-Württemberg.....01	<input type="checkbox"/>		11,7	12,2	11,2
Bayern.....02	<input type="checkbox"/>		11,4	11,4	11,4
Berlin.....03	<input type="checkbox"/>		4,1	3,8	4,5
Brandenburg.....04	<input type="checkbox"/>		2,6	2,2	3,0
Bremen.....05	<input type="checkbox"/>		1,2	1,4	0,9
Hamburg.....06	<input type="checkbox"/>		2,3	2,4	2,3
Hessen.....07	<input type="checkbox"/>		7,8	8,1	7,6
Mecklenburg-Vorpom.....08	<input type="checkbox"/>		2,1	2,0	2,3
Niedersachsen.....09	<input type="checkbox"/>		10,4	10,6	10,1

452



der linken Maustaste darauf und verschieben Sie die Elemente unter stetigem Halten der linken Maustaste an den gewünschten Ort. Es ist möglich, dass vorher allerdings noch weitere Elemente auf der Seite zu entfernen sind, da es sonst zu Überschneidungen und damit zu Unleserlichkeiten kommen könnte (siehe Abb. 1.12/ 1.13)

Abb. 1.12

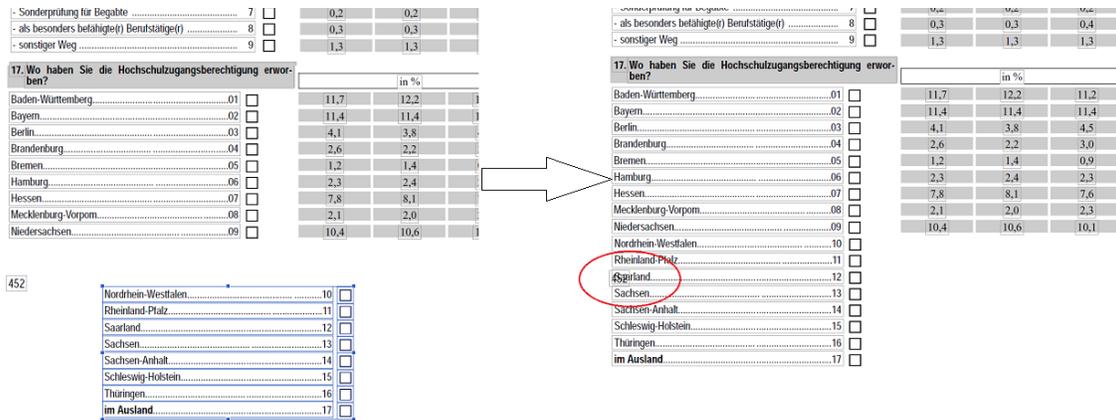
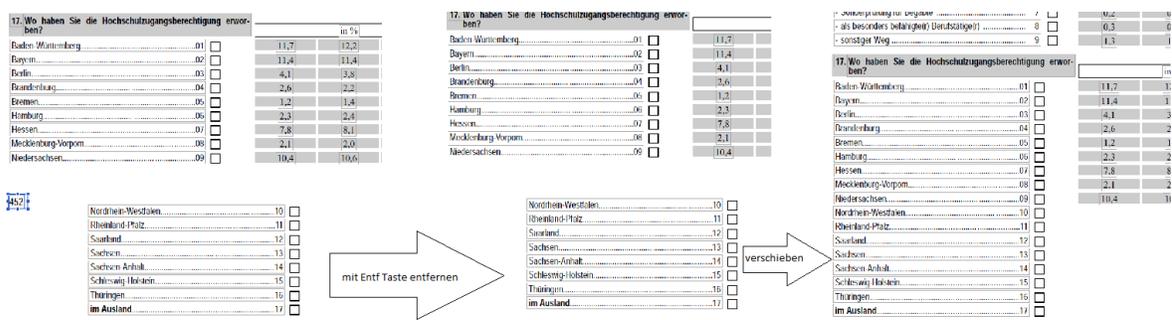


Abb. 1.13



In Abbildung 1.12 und 1.13 sehen Sie eine beispielhafte Vorgehensweise. In Abbildung 1.13 wurde die Zahl 452 entfernt, indem es ebenfalls mittels des Instruments „Texte und Bilder bearbeiten“ markiert wurde und durch anschließendes Drücken der „Entf“ Taste der Tastatur gelöscht worden ist.

Im Anschluss daran können Sie wie in Schritt 2.1.1 dargestellt mit dem Zuschneiden der Frage fortfahren.

- Schritt 2.2: Nun können wir die ausgeschnittene Frage transportieren. Dafür wählen Sie das Instrument „Extrahieren“ aus. Wie bereits in Abb. 1.3 dargestellt, befindet sich dieses Werkzeug ebenfalls unter der Kategorie „Seiten manipulieren“. Dabei öffnet sich erneut eine Maske, in der Sie die zu entnehmende Seite auswählen sollen. Hierbei ist meist, die zuvor zugeschnittene Seite vorausgewählt. Überprüfen Sie es gegebenenfalls noch einmal und Bestätigen Sie dann durch Klicken auf den „OK“ Button. Dies ist in Abb. 1.14 dargestellt.

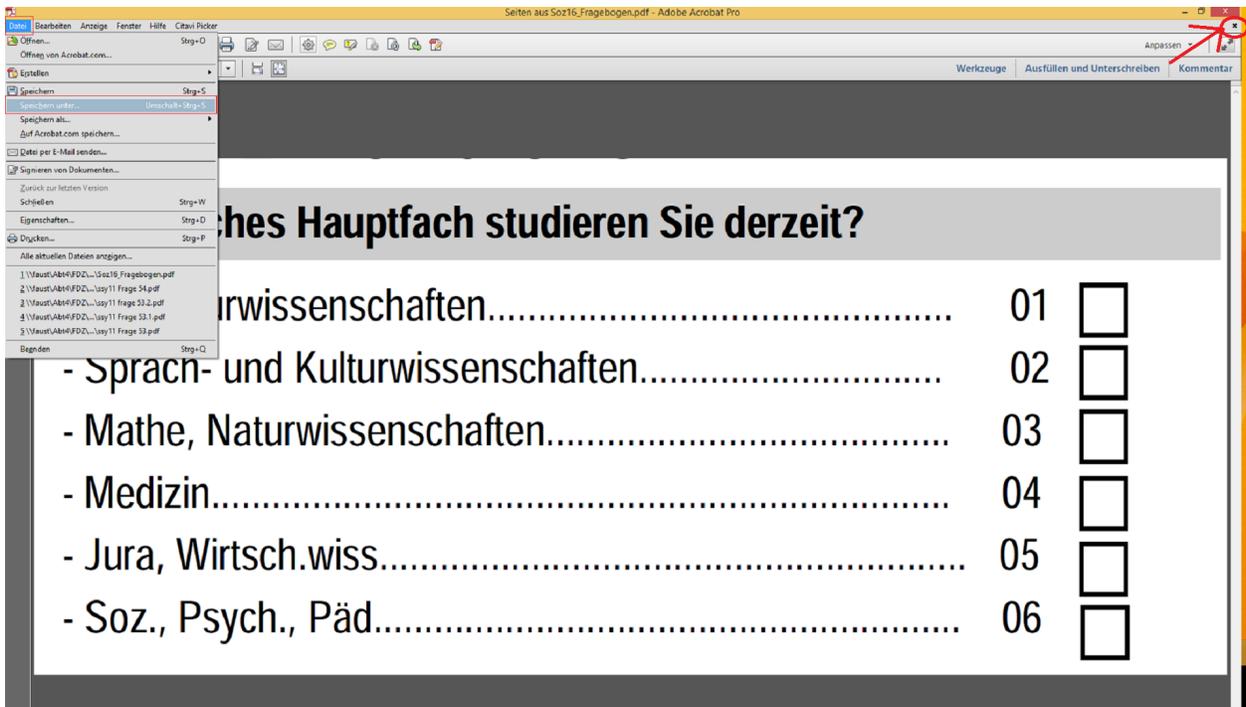
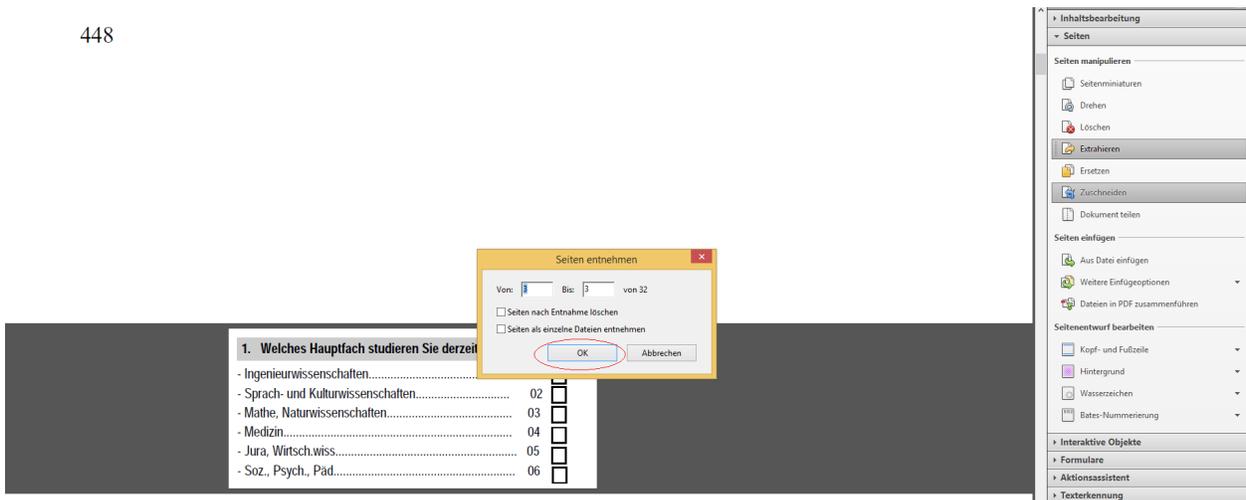
Abb. 1.14

Anschließend sollte auf Ihrem Bildschirm ein ähnliches Bild erscheinen, wie in Abb. 1.15 abgebildet.

Abb. 1.15

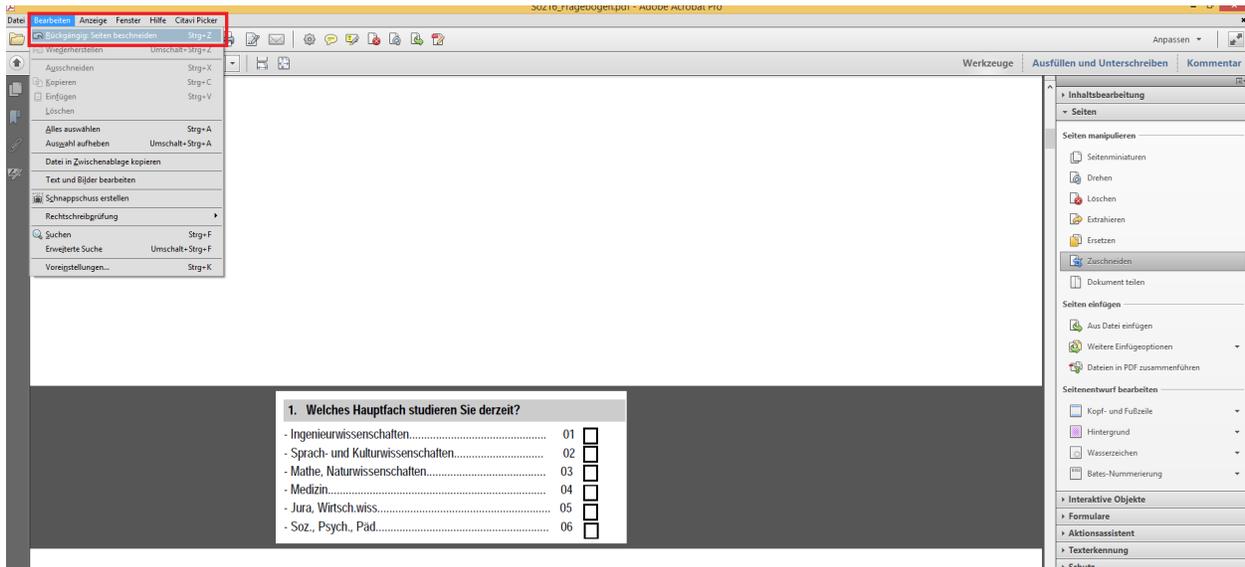
Um das Transportieren nun vollständig abzuschließen klicken Sie oben rechts auf Datei und anschließend auf „Speichern unter“. Wichtig ist hierbei, dass der ausgewählte Dateiname, unter dem die neue Datei gespeichert werden soll der Fragenummer entspricht. (eventuell mit der Excel Tabelle abgleichen)

448



- Schritt 2.3: Um nun fortzufahren und weitere Fragen als pdf Datei zu extrahieren, klicken Sie auf das kleine Kreuz (wie in Abb. 1.15 gekennzeichnet). Dabei wird das Fenster geschlossen. Damit die ursprüngliche Datei wieder als solche vorliegt, wie es vor dem Beschneiden der Seiten der Fall war, wählen Sie nun in der oberen Leiste erst „Bearbeiten“ aus und anschließend „Rückgängig: Seiten beschneiden“ (siehe Abb. 1.16). Dieser Schritt ist vor allem empfehlenswert, wenn sich mehrere Fragen auf einer Seite befinden. Hierdurch wird das stetige Wiederöffnen der Datei vermieden und es wirkt sich zeitsparend aus.

Abb. 1.16



- Schritt 3: Der letzte Schritt beinhaltet das Konvertieren der extrahierten Fragen vom pdf Format ins png Format. Hierfür können Sie den Online Converter „pdf2png“ nutzen. <https://online2pdf.com/de/pdf-zu-png-konvertieren>

**ACHTUNG:** Hier ist nur ein begrenztes Volumen an Konvertierungen pro Tag von 150MB möglich.

## 3.4 Bilderfassung aus RagTime

### 3.4.1 Voraussetzungen

Fragebögen, die als eine Ragtime Datei vorliegen sollen gemäß der nächsten Schritte bearbeitet werden.

Liegt zu dem Fragebogen, den es von Ihnen zu bearbeiten gilt, eine äquivalente Excel Datei vor, so wäre es ratsam diese als Grundlage für die weitere Arbeit zu verwenden. Hierbei ist es notwendig, dass jeder aufgeführten Frage in der Excel Datei eine finale Bilddatei gegenüber steht.

#### Schritt 1:

Zuerst öffnen Sie die zu bearbeitende Datei mit Ragtime.

#### Schritt 2:

#### Schritt 3.0: Markieren von Elementen in Ragtime

Wählen Sie nun die Elemente aus, die es als Bilddatei zu exportieren gilt. Markieren Sie die ausgewählten Inhalte, indem Sie den Mauszeiger vom Beginn des zu markierenden Feldes unter Festhalten der linken Maustaste bis zum Ende des gewünschten Feldes bewegen (ziehen). Während dieses Vorgangs bildet sich ein schwarzes Rechteck, in welchem

1
ANGABEN ZUM STUDIUM

**01.**  **Ist Ihr derzeitiges Studium ein . . .**

Erststudium (auch nach Studiengangwechsel – wichtig: bisher kein Abschluss)

Zweitstudium (nach abgeschlossenem Erststudium)

Weiterbildender Studiengang (Ergänzungs-/Zusatz-/Aufbaustudium oder weiterbildender Masterstudiengang)

Promotionsstudium (nach erstem Abschluss)

**02.**  **Welches Hauptfach bzw. welche Fächer studieren Sie im Sommersemester 2006?**

Bitte tragen Sie hier die Nummer des zutreffenden bzw. weitestgehend zutreffenden Studienbereichs/Studienfachs aus der links abgedruckten Liste der Studienbereiche/Studienfächer ein.

1. Fach

ggf. 2. Fach

ggf. 3. Fach

**03.**  **Welchen Abschluss streben Sie zunächst an?** (Bei mehreren angestrebten Abschlüssen bitte nur den zeitlich nächsten Abschluss in der ersten Spalte ankreuzen)

**Sollten Sie bereits einen Abschluss erworben haben, bitte in der zweiten Spalte ankreuzen.**

	angestrebter Abschluss	erworbener Abschluss
Fachhochschuldiplom	<input type="checkbox"/>	<input type="checkbox"/>
Diplom einer Universität/Kunsthochschule o. ä.	<input type="checkbox"/>	<input type="checkbox"/>
Magister	<input type="checkbox"/>	<input type="checkbox"/>
Bachelor	<input type="checkbox"/>	<input type="checkbox"/>
Master	<input type="checkbox"/>	<input type="checkbox"/>
Staatsexamen	<input type="checkbox"/>	<input type="checkbox"/>
kirchliche Prüfung	<input type="checkbox"/>	<input type="checkbox"/>
Promotion	<input type="checkbox"/>	<input type="checkbox"/>
anderer Abschluss (einschließlich Abschluss im Ausland)	<input type="checkbox"/>	<input type="checkbox"/>
keinen Abschluss	<input type="checkbox"/>	<input type="checkbox"/>

Falls Sie nicht in einem Masterstudiengang eingeschrieben sind:

**Haben Sie die Absicht, nach dem derzeit angestrebten Abschluss einen Masterstudiengang zu absolvieren?**

nein

ja, direkt nach Erwerb des Bachelor bzw. nach meinem ersten Abschluss

ja, aber erst, wenn ich Berufserfahrungen gesammelt habe

weiß ich noch nicht

**04.**  **Streben Sie einen Lehramtsabschluss an?**

nein

ja

**05.**  **Haben Sie seit Ihrer Erstmatrikulation das Hauptstudienfach oder den angestrebten Abschluss gewechselt?** (Als Studiengangwechsel gilt nicht die Aufnahme eines weiteren Studiums nach erfolgreichem Abschluss eines ersten Studiums!)

nein

ja, Fach und Abschluss

ja, nur das Fach

ja, nur den Abschluss

Falls Sie das Hauptstudienfach und/oder den Abschluss gewechselt haben:

**In welchem Fach waren Sie bei der Erstmatrikulation eingeschrieben und welchen Abschluss strebten Sie damals an?**

Hauptstudienfach bei der Erstmatrikulation (Nummer aus der Liste der Studienfächer entnehmen und eintragen)

angestrebter Abschluss bei der Erstmatrikulation (bitte Bezeichnung eintragen):

Wie viele Semester waren Sie in dem ursprünglich gewählten Studiengang eingeschrieben?

Spaltenblock

Textfeldrahmen

alle zu markierenden Elemente vollständig enthalten sein müssen; Elemente außerhalb dieses Vierecks werden nicht markiert.

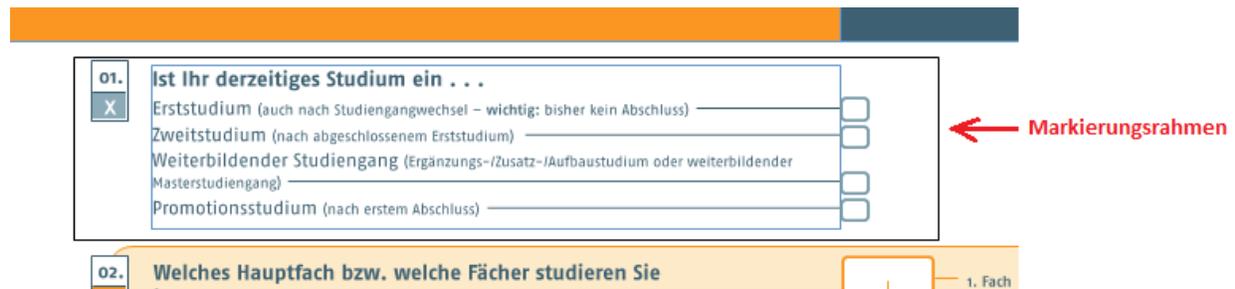


Abb. 3.4: Elemente markieren.

- Falls der Text nicht markiert wurde, da der Textfeldrahmen noch weitere Fragen beinhaltet fahren Sie fort mit Schritt 3.1 ff.
- Ist dies nicht der Fall; weiter mit Schritt 4

### Schritt 3.1: Text in einem Textfeldelement bearbeiten

Um gewünschte Textelemente aus einem zu großen Textfeldrahmen auszuschneiden, muss eine Platzierung des Mauszeigers innerhalb des Textfeldes erfolgen. Durch einen darauffolgenden Rechtsklick öffnet sich ein Menü, in welchem Sie die Auswahlmöglichkeit „Komponente öffnen“ erwählen. Dadurch öffnet sich ein neues Fenster, in dem die Bearbeitung des gesamten Textes des ausgewählten Textfeldrahmens möglich ist. Nun können Sie den Text bearbeiten: Löschen Sie alle irrelevanten Textpassagen und schließen Sie das Fenster.

**ACHTUNG:** Beachten Sie, dass Sie das kleine, graue X drücken, um das Komponentenfenster zu schließen.

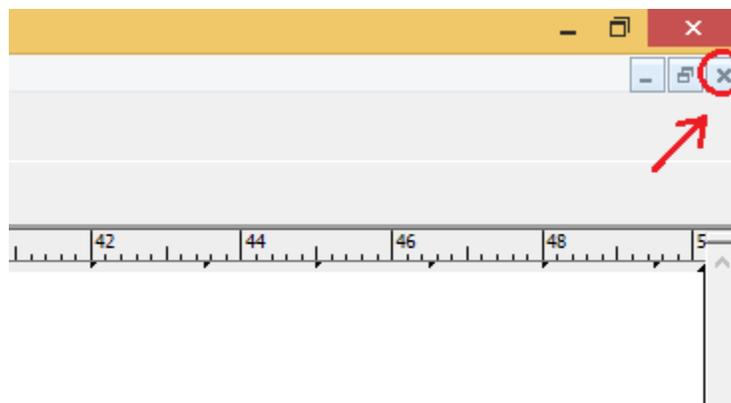


Abb. 3.5: Komponentenfenster schließen.

### Schritt 3.2: Anpassen des Textfeldrahmens

Durch das Löschen irrelevanter Elemente kann sich der Text innerhalb des Textfeldrahmens verschieben. Deswegen muss der Textfeldrahmen in diesem Schritt angepasst werden: Durch Anklicken des Textfeldrahmens ist dies möglich. Bewegen Sie hierfür den Mauszeiger auf den mittleren schwarzen Markierungspunkt des unteren Randes des Textfeldrahmens. Dabei müsste sich der Mauszeiger zu einem Kreuz verändern. Durch Linksklick und anschließendes Festhalten können Sie den Rahmen sowohl nach unten als auch nach oben anpassen. Ziehen Sie den Rahmen bis zum Ende der letzten Textpassage des Textelementes hoch. Dadurch ist gewährleistet, dass der Text ebenfalls bei dem Exportieren markiert werden kann.

### Schritt 3.3: Verschieben der zusätzlichen Elemente

**03. Welchen Abschluss streben Sie zunächst an?** (Bei mehreren angestrebten Abschlüssen bitte nur den zeitlich nächsten Abschluss in der ersten Spalte ankreuzen)  
**Sollten Sie bereits einen Abschluss erworben haben, bitte in der zweiten Spalte ankreuzen.**

	angestrebter Abschluss	erworbener Abschluss
Fachhochschuldiplom	<input type="checkbox"/>	<input type="checkbox"/>
Diplom einer Universität/Kunsthochschule o. ä.	<input type="checkbox"/>	<input type="checkbox"/>
Magister	<input type="checkbox"/>	<input type="checkbox"/>
Bachelor	<input type="checkbox"/>	<input type="checkbox"/>
Master	<input type="checkbox"/>	<input type="checkbox"/>
Staatsexamen	<input type="checkbox"/>	<input type="checkbox"/>
kirchliche Prüfung	<input type="checkbox"/>	<input type="checkbox"/>
Promotion	<input type="checkbox"/>	<input type="checkbox"/>
anderer Abschluss (einschließlich Abschluss im Ausland)	<input type="checkbox"/>	<input type="checkbox"/>
keinen Abschluss	<input type="checkbox"/>	<input type="checkbox"/>

Falls Sie nicht in einem Masterstudiengang eingeschrieben sind:  
**Haben Sie die Absicht, nach dem derzeit angestrebten Abschluss einen Masterstudiengang zu absolvieren?**

nein	<input type="checkbox"/>
ja, direkt nach Erwerb des Bachelor bzw. nach meinem ersten Abschluss	<input type="checkbox"/>
ja, aber erst, wenn ich Berufserfahrungen gesammelt habe	<input type="checkbox"/>
weiß ich noch nicht	<input type="checkbox"/>

**04. Streben Sie einen Lehramtsabschluss an?**

nein	<input type="checkbox"/>
ja	<input type="checkbox"/>

**05. Haben Sie seit Ihrer Erstimmatrikulation das Hauptstudienfach oder den angestrebten Abschluss gewechselt?** (Als Studiengangwechsel gilt nicht die Aufnahme eines weiteren Studiums nach erfolgreichem Abschluss eines ersten Studiums!)

nein	<input type="checkbox"/>
ja, Fach und Abschluss	<input type="checkbox"/>
ja, nur das Fach	<input type="checkbox"/>
ja, nur den Abschluss	<input type="checkbox"/>

Falls Sie das Hauptstudienfach und/oder den Abschluss gewechselt haben:

**>> weiter mit Frage 6**

*hier klicken!*

Abb. 3.6: Textfeldrahmen anpassen

Durch das mögliche Verschieben des Textes, müssen die dazugehörigen Elemente wie z.B. Antwortkästchen einer Frage etc. ebenfalls angepasst werden. Solche Elemente können wie in Schritt 3.0 beschrieben verschoben werden.

### Schritt 3.4: Anpassen des Spaltenblocks

Es kann zudem der Fall eintreten, dass der Spaltenblock angepasst werden muss. Dazu wählt man diesen durch einen Klick aus. Dabei öffnet sich auf der linken Seite des Fensters eine graue, durchnummerierte Leiste (1). In dieser ist es möglich Elemente wie z.B. grau unterlegte Kreuze zu entfernen. Durch das Markieren einer Nummer wird das dazugehörige Kästchen des Spaltenblocks blau unterlegt (2) und man kann es durch Drücken der Entf - Taste löschen (3). Wie bei dem Textfeldrahmen, ist es bei dem Spaltenblock möglich die Ränder durch die schwarzen Punkte zu verschieben.

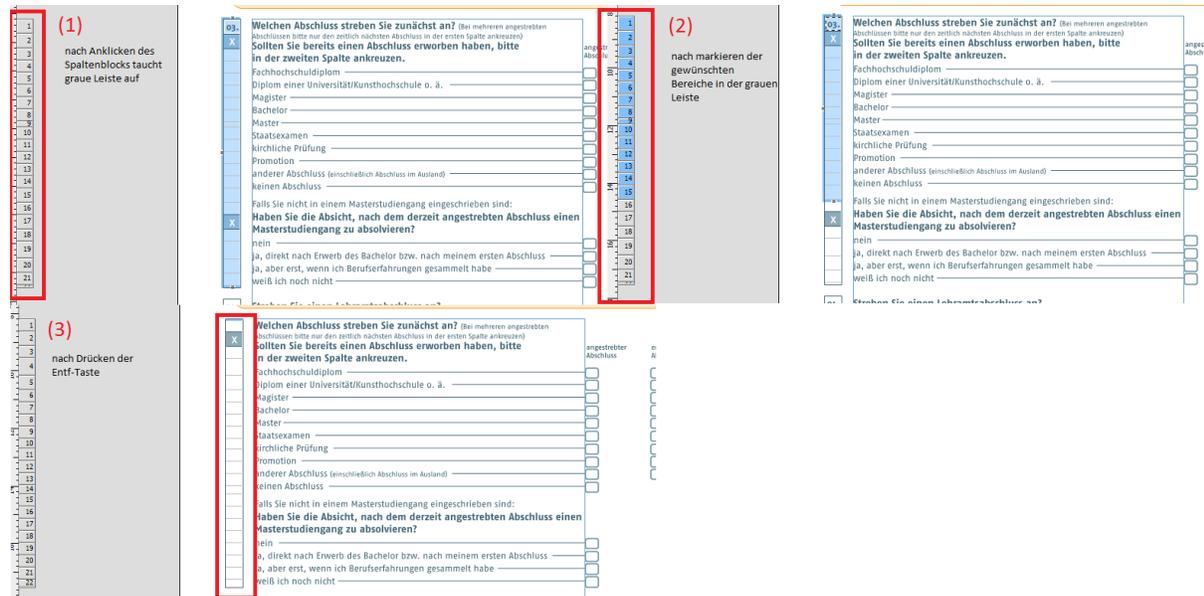


Abb. 3.7: Anpassen des Spaltenblocks

### Schritt 3.5: Fertigstellung

Liegen nun alle gewünschten Elemente in der richtigen Anordnung, werden alle markiert und setzt mit Schritt 4 fort.

### Schritt 4: Exportieren

Sind alle Elemente markiert (WICHTIG: überprüfen Sie, ob wirklich jedes Element von Markierungspunkten umrandet ist, ansonsten werden nicht markierte Elemente nicht exportiert), kann durch Betätigen der rechten Maustaste der markierte Bereich exportiert werden. Hierbei ist es wichtig darauf zu achten, dass der Mauszeiger auf einem der schwarzen Markierungspunkte liegt, da ansonsten die Markierung aufgehoben wird. Im weiteren Verlauf öffnet sich ein Menü, in welchem die Option „Exportieren...“ aufzufinden ist. Wählen Sie diese Option aus um die ausgewählten Elemente gesondert von der ursprünglichen Datei zu speichern.

### Schritt 5: Speichern

Beim Speichern müssen Sie folgende Details beachten:

- Falls eine Excel- Datei vorhanden ist, achten Sie auf übereinstimmenden Dateinamen und der dazugehörigen Fragennummer (die in der Excel-Datei aufgeführt ist).
- Stellen sie sicher, dass der Dateityp „PostScript-Illustration (EPSF) (\*.eps)“ ausgewählt ist.
- Unter dem Punkt „Exportieren“ muss zudem die Option „Auswahl“ gewählt sein.

### Schritt 6: Konvertieren von .eps zu .png

Die abgespeicherten .eps - Dateien müssen in einem weiteren Schritt in .png - Dateien konvertiert werden. Dies erfolgt am Besten mit dem ReaConverter7Pro. (dieser Konverter ist kostenpflichtig, daher handelt es sich hierbei um eine Testversion. Aus diesem Grund können hierbei lediglich fünf Dateien gleichzeitig konvertiert werden.)

- Um die Dateien zu konvertieren, zieht man die eps Dateien in das offene Feld indem man Dateien markiert, die rechte Maustaste festhält und Dateien mittels Bewegens des Mauszeigers schließlich rüberzieht.
- Wenn Sie ab einem bestimmten Punkt alle Objekte markieren wollen, dann klicken Sie mit der linken Maustaste zunächst auf die erste Datei / Ordner und halten Sie die Shift-Taste gedrückt, während Sie auf das letzte Objekt Ihrer Auswahl mit der linken Maustaste klicken.

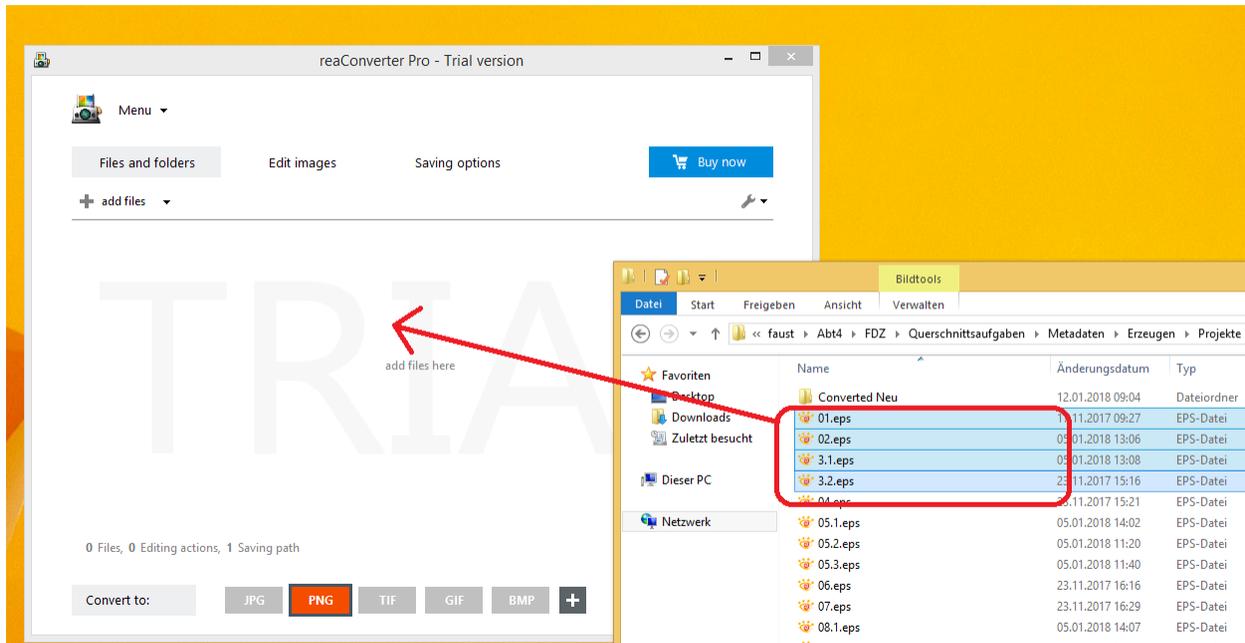


Abb. 3.8: Anpassen des Spaltenblocks

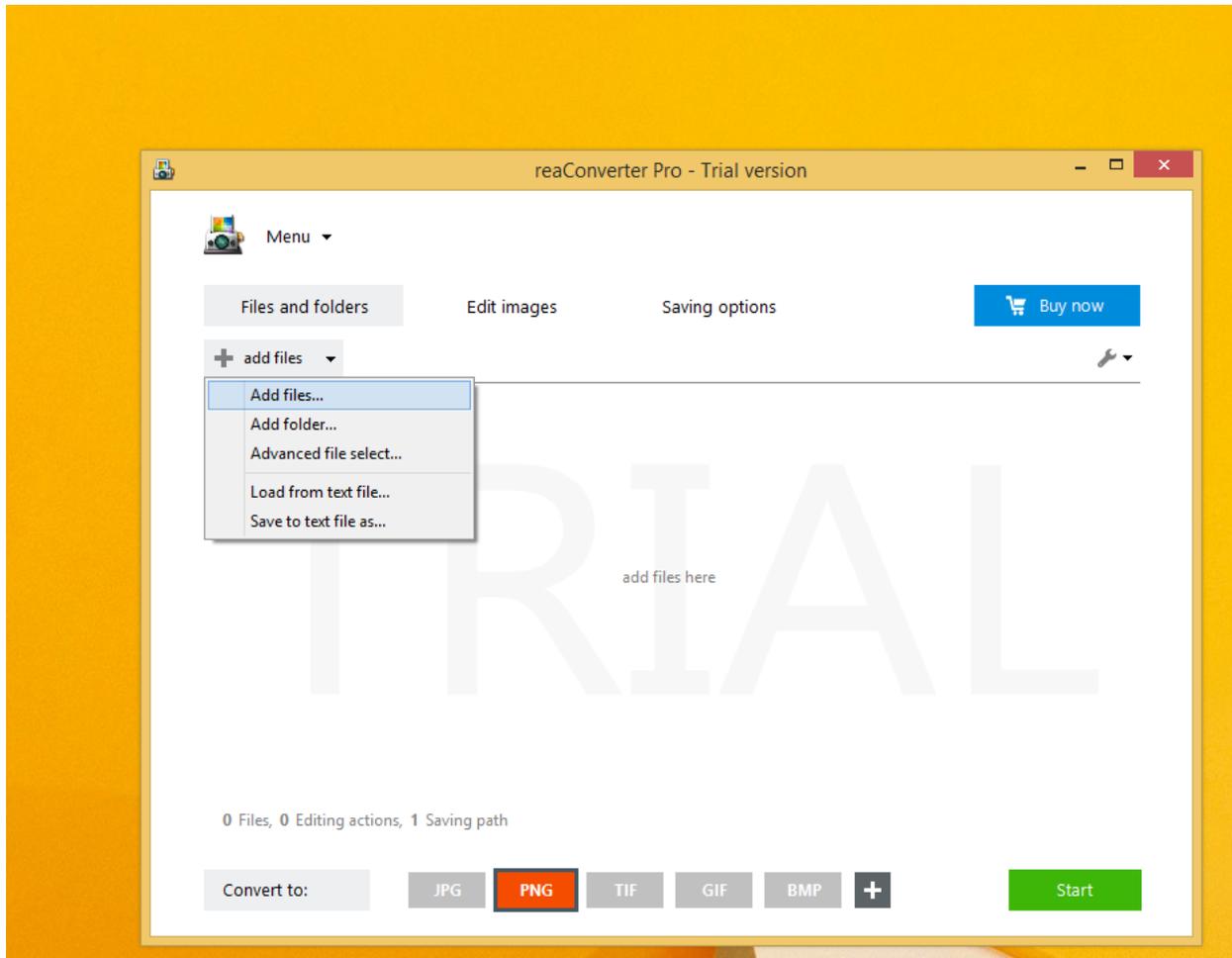
- Oder man klickt auf „add files“. Der Computer kann dann auf Dateien durchsucht werden und ausgewählte Dateien können hinzugefügt werden.
- Als nächstes wählt man unter Convert to: das gewünschte Format „PNG“ aus. Das ausgewählte Format sollte nun orange aufleuchten. Als letztes markiert man die zu konvertierenden Dateien und klickt unten rechts auf den grünen Button „Start“.

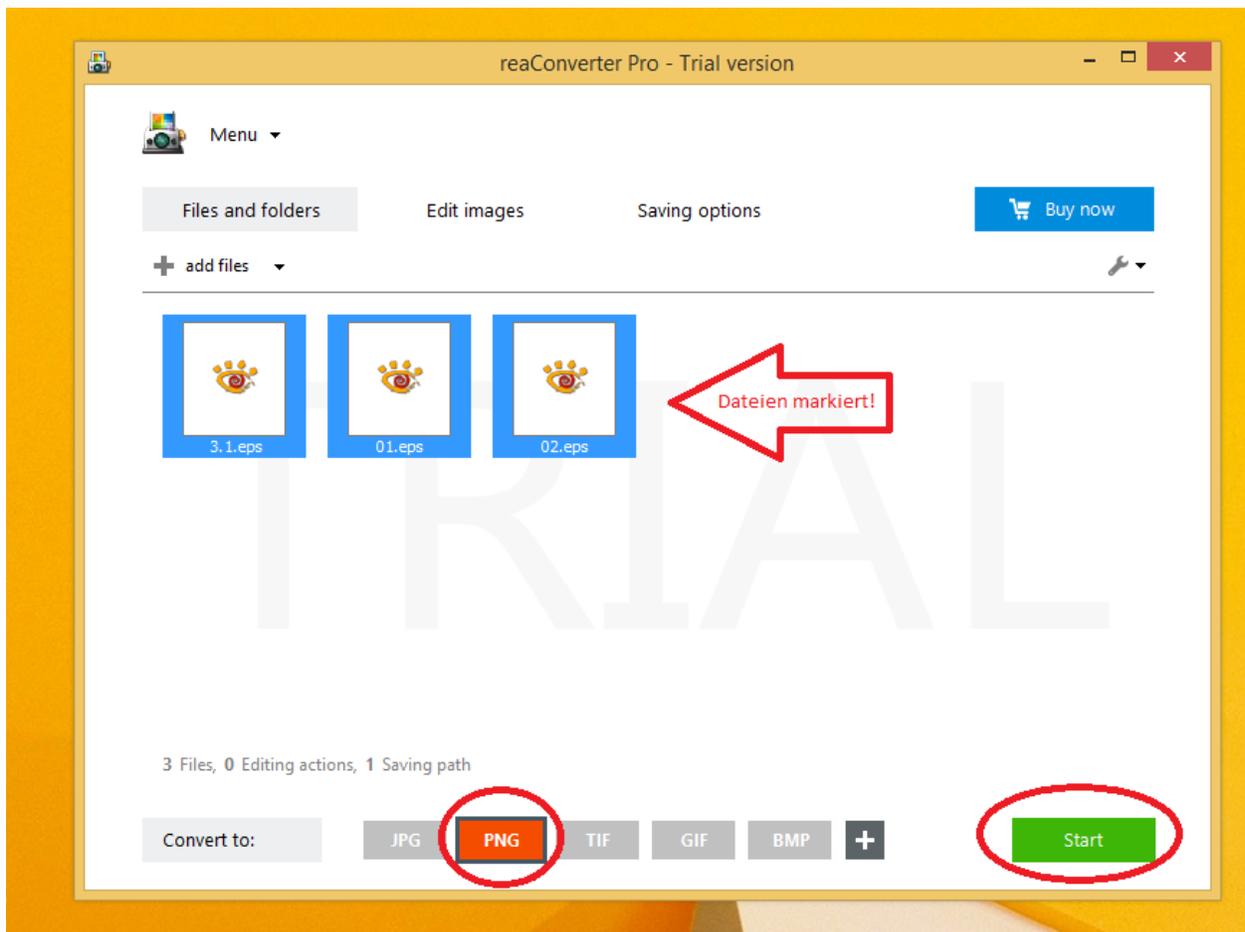
### 3.5 Excel-Tabellen

Für die weiteren Ebenen können Sie bereits vorbereitete Excel-Tabellen verwenden, in denen verschiedene Metadaten spaltenweise erfasst werden. Diese Excel-Tabellen finden Sie im [Vorlage-Ordner](#) der jeweiligen Ebene (vgl. Kapitel 3.4). Welche Metadaten Sie an welcher Stelle in der Excel-Tabelle eintragen müssen, erkennen Sie an den bereits vorgegebenen Spaltenüberschriften in der ersten Zeile der Tabelle. Dort stehen die Bezeichnungen der jeweiligen Metadaten.

Die Suffixe „.de“ und „.en“ sind Teil vieler Spaltenüberschriften und weisen auf die Sprache des einzutragenden Metadatum hin. Ab der zweiten Zeile sind die Tabellenvorlagen leer. Dort können Sie Ihre Inhalte entsprechend der Spaltenüberschriften eintragen. Inhaltliche Hilfen hierfür finden Sie in den Erklärungen zu den einzelnen Ebenen.

Bitte beachten Sie außerdem:





- Die Excel-Tabellen enthalten je nach Ebene unterschiedlich viele Tabellenblätter, die Sie bearbeiten müssen.
- Die Anzahl der Spalten pro Tabellenblatt variiert, sodass ein seitliches Scrollen oftmals notwendig ist.
- Es gibt Metadaten, die Sie ausfüllen *müssen*, und solche, die Sie ausfüllen *können*. Die konkreten Ausfüllanweisungen finden Sie in Kapitel 4.

### Kontrolliertes Vokabular

Für einige Metadaten gibt es ein sogenanntes „kontrolliertes Vokabular“, d.h. dort können nur bestimmte Inhalte in die Spalten eingetragen werden. In diesen Spalten wird Ihnen in jeder Zelle eine Auswahl der möglichen Antworten anhand eines Drop-Down-Menüs angeboten und Sie müssen diese nur auswählen.

Häufig ist es der Fall, dass aus dem kontrollierten Vokabular einer bestimmten Zelle automatisch der Inhalt der nächsten Zelle folgt. Für dieses Szenario sind die [Excel-Vorlagen](#) vorbereitet, d. h. in den betreffenden Spalten sind über mehrere Zeilen bereits Formeln hinterlegt, die die nächste Zelle automatisch füllen und Ihnen viel Tipparbeit ersparen. Die Vorlagen sind für alle Fälle vorbereitet, so dass die Formeln auch in höher nummerierten Zeilen stehen werden, die Sie voraussichtlich nicht mehr benötigen (bis einschließlich Zeile 20 der Excel-Vorlage für die Ebenen Instrumente sowie Datensätze, bis einschließlich Zeile 2000 der Excel-Vorlage für die Fragen- und Variablenebene). Zum leichteren Erkennen sind die betreffenden Zeilen in den Vorlagen bereits grau hinterlegt.

Da Formeln in solchen Zellen, die nicht mit Inhalt befüllt sind, einen fehlerhaften Upload der Excel-Tabelle hervorrufen, müssen die überflüssigen Formeln aus den nicht benötigten Zeilen herausgelöscht werden. Dies können Sie erledigen, indem Sie die nicht benötigten der grau eingefärbten Zeilen bis einschließlich der Zeile 2000 (für Fragen- und Variablenebene) markieren und über das Menü per „Blattzeilen löschen“ komplett entfernen (vgl. dazu analog zur ehemaligen Exceltabelle für die Datensatzebene – mittlerweile gibt es dort nur noch Eingabemasken [Abb. 3.9](#)).

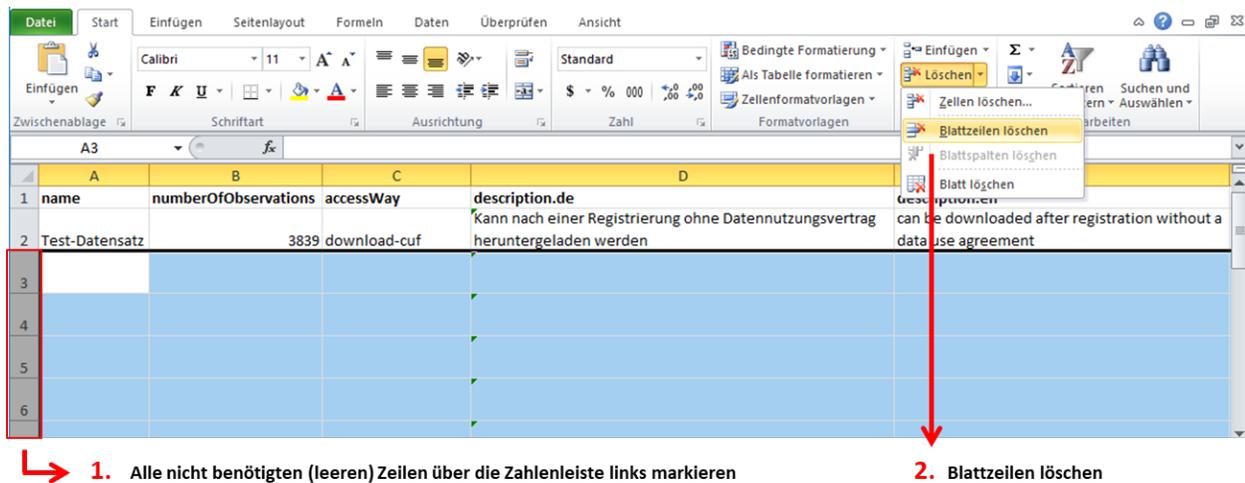


Abb. 3.9: Beispiel für das Löschen nicht benötigter Formeln aus der Excel-Vorlage für die Ebene *Datensätze*

## 3.6 Templates

In dem [Template](#) befinden sich die relevanten Exceldateien für DatengeberInnen. Dies ist nur noch relevant für questions/variables und für die related publications.

Ausfüllhinweis: Die Excel-Interfaces enthalten teilweise Dropdownmenüs und Formeln als Hilfestellung. Alle Zeilen, die grau eingefärbt sind, enthalten diese Hilfestellungen. Der Datengeber kann einfach die Felder ausfüllen. Nach Fertigstellung der Dateien und vor Hochladen der Excel-Interfaces in das MDM müssen alle grauen Zeilen, die nicht genutzt werden gelöscht werden. Fertig! Vom FDZ-Team müssen im Anschluss noch folgende Punkte erledigt werden:

Felder	To Do
Datenaufbereitungsfelder	bei Bedarf für externe Projekte löschen (z.B. varname_alt, Varlabel_alt)
alle Felder	nur einblenden, wenn Datengeber die Informationen liefern (z.B. englische Felder, GenerationDetails, ...)
Zugangswege	<ul style="list-style-type: none"> <li>• Spalte „accessWays“ entfernen, diese wird im Nachhinein vom FDZ auf Basis der Zugangswegspalten ausgefüllt</li> <li>• entspr. Absprache zu Zugangswegen nur relevante Zugangsweg-Spalten drin lassen</li> <li>• wenn nur ein Zugangsweg -&gt; alle Zugangsweg-Spalten rauslassen</li> </ul>
accessWays	raus

### 3.7 Excel Makros verwenden

Um Makros in Excel zu nutzen, wird der VBA Editor benötigt. Dieser wird mit Alt+F11 geöffnet. Über Einfügen > Modul wird ein neues Modul angelegt, in das die fertigen Skripte einfach reinkopiert werden können.

In der Excel Tabelle können die Makros über Ansicht > Makros > Makros anzeigen > Ausführen gestartet werden.

Name	Was kann es?	Wo wird es genutzt?
panel-Identifizier	Generiert den PanelIdentifier durch den Vergleich von Variablennamen. Dabei dürfen Variablennamen verschiedene Versionierungen und/oder Zugangswege haben. Die Eingabe des Projektnamens, der Datensatznummer, der Spalte in die der panelIdentifier eingefügt werden soll und die Auswahl der Variablen erfolgt über eine Inputbox Beispiel: der panelIdentifier zu adem01_g1v1r und bdem01_g1v3r lautet z.B. abs2005-ds1-dem01_g1	panelIdentifier in variables.xlsx
add-Prefix	Fügt einer oder mehreren Variablen in einer Zelle und durch Komma getrennt ein Präfix hinzu (z.B. abs2005). Der Bereich in der die Variablen stehen und das Projektkürzel werden über eine InputBox eingegeben Beispiel: bski01d_v1,bski01a_v1 bski01u_v1 wird zu abs2005-bski01d_v1,abs2005-bski01a_v1,abs2005-bski01u_v1	Panelvariablen in man-Metadaten.xlsx

### 3.8 Editing Macros Excel

Die folgenden Makros können unterstützend nach dem manuellen Ausfüllen der Exceltabelle vimport\_dsNr.xlsx genutzt werden. Vor dem Ausführen der Makros sollte sichergestellt werden, dass die Tabellenblätter der Exceltabelle richtig benannt wurden (variables und relatedQuestions) und die für das jeweilige Makro notwendigen Spalten vorhanden sind.

### 3.8.1 Wie werden Makros ausgeführt?

Um Makros in Excel zu nutzen, wird der VBA Editor benötigt. Dieser wird mit Alt+F11 geöffnet. Über Einfügen > Modul wird ein neues Modul angelegt, in das die fertigen Skripte einfach reinkopiert werden können. In der Excel Tabelle können die Makros über Ansicht > Makros > Makros anzeigen > Ausführen ausgeführt werden.

### 3.8.2 Makro-Übersicht

#### panelIdentifier

Wo wird es genutzt? `panelIdentifier` in `vimport.xlsx`

- generiert den `panelIdentifier` durch den Vergleich vom Variablenstamm (ohne-Präfix)
- dabei dürfen Variablennamen verschiedene Versionierungen und/oder Zugangswege-haben
- das FDZ Variablenschema muss verwendet worden sein (Variablenname z.B.-`astu01_g1v1r` mit den möglichen Zugangswegen `c,d,o,r,a`)
- im Tabellenblatt `variables` müssen die beiden Spaltennamen `name` und `panelIdentifier` vorhanden sein
- der Projektnamen und die Datensatznummer werden über eine Inputbox eingegeben
- Beispiel: der `panelIdentifier` zu `adem01_g1v1r` und `bdem01_g1v3r` lautet z.B.-`abs2005-ds1-dem01_g1`

#### derivedVariablesIdentifier

Wo wird es genutzt? `derivedVariablesIdentifier` in `vimport.xlsx`

- generiert den `derivedVariablesIdentifier` durch den Vergleich vom Variablenstamm (mit Präfix)
- das FDZ Variablenschema muss verwendet worden sein (Variablenname z.B. `astu01_g1v1r` mit den möglichen Zugangswegen `c,d,o,r,a`)
- im Tabellenblatt `variables` müssen die beiden Spaltennamen `name` und `derivedVariablesIdentifier` vorhanden sein
- der Projektname und die Datensatznummer werden über eine Inputbox eingegeben
- Beispiel: der `derivedVariablesIdentifier` zu `adem01` und `adem01_g1r` des Projektes `gra2005-ds1` lautet z.B. `gra2005-ds1-adem01`

#### accessWaysInOneColumn

##### `accessWaysInOneColumn`-Skript

Wo wird es genutzt? `accessWays` in `vimport.xlsx` - wenn die vier Spalten nicht verfügbar im Download-CUF, nicht verfügbar im Download-SUF, nicht verfügbar im Remote-Desktop-SUF und nicht verfügbar im On-Site-SUF ausgefüllt wurden

- wurden in der Excel Tabelle die vier Spalten nicht verfügbar im Download-CUF, nicht verfügbar im Download-SUF, nicht verfügbar im Remote-Desktop-SUF und nicht verfügbar im On-Site-SUF durch ankreuzen mit „x“ ausgefüllt, können daraus die Zugangswege in einer Spalte generiert werden
- dafür müssen die Spaltennamen nicht verfügbar im Download-CUF, nicht verfügbar im Download-SUF, nicht verfügbar im Remote-Desktop-SUF, nicht verfügbar im On-Site-SUF, `accessWays` und `name` im Tabellenblatt `variables` vorhanden sein

- Beispiel: wurden alle vier Spalten einer Variablen mit „x“ ausgefüllt, wird der zugehörige Zugangsweg „not-accessible“ ermittelt

## matchSurveyNumbers

### matchSurveyNumbers

Wo wird es genutzt? surveyNumbers in vimport.xlsx - wenn das Präfix im Variablennamen in Abhängigkeit von der surveyNumber vergeben wurde

- wurden das Präfix des Variablennamen in Abhängigkeit von der surveyNumber vergeben, kann die survey-Number automatisch generiert werden
- es muss ein zusätzlichen Tabellenblatt mit dem Namen optionalEntries eingefügt werden
- dieses Tabellenblatt hat 2 Spalten mit den Namen prefix (hier steht z.B. a, b oder c) und surveyNumber (hier wird die zum Präfix zugehörige Nummer der Erhebung eingetragen)
- das Tabellenblatt variables muss die beiden Spalten name und surveyNumbers enthalten
- Beispiel: alle Variablen mit dem Präfix „a“ (z.B. astu01, adem05) bekommen die surveyNumber 1, alle Variablen mit dem Präfix „b“ (z.B. bstu02, bdem03) bekommen die surveyNumber 2, alle Variablen mit abweichendem (bzw. ohne) Präfix (z.B. pid, wave, wgt-Variablen) bekommen die surveyNumbers 1,2

## addRelatedQuestionsForGeneratedVariables

### addRelatedQuestionsForGeneratedVariables

Wo wird es genutzt? vimport.xlsx > relatedQuestions - das Makro ergänzt im Tabellenblatt relatedQuestions generierte Variablen von Fragebogenvariablen. Das Makro funktioniert nur, wenn die generierten, wenn die generierte Variablen denselben Stamm haben, wie ihre Ausgangsvariablen (z.B. astu01\_g1 von astu01), d.h. wenn das FDZ-Variablennamenschema (mindestens Silber) verwendet wurde!

- Es müssen alle Variablen aus dem Datensatz im Tabellenblatt variables eingetragen sein.
- Es müssen alle Fragebogenvariablen (d. h. „Originalvariablen“) inkl. questionNumber und instrumentNumber im Tabellenblatt relatedQuestions eingetragen sein.
- Das Makro gleicht die Variablenliste vom Tabellenblatt relatedQuestions mit der aus dem Tabellenblatt variables ab und ergänzt die generierten Variablen der eingetragenen Fragebogenvariablen im Tabellenblatt relatedQuestions.
- Die generierten Variablen werden inklusive questionNumber und instrumentNumber am Ende des Tabellenblatts eingefügt
- zur Kontrolle werden die neuen Zellen blau gefüllt
- relatedQuestionString.de/en muss nicht ausgefüllt werden. name im Tabellenblatt variables und name, questionNumber und instrumentNumber im Tabellenblatt relatedQuestions müssen gefüllt sein- name, questionNumber, instrumentNumber im Tabellenblatt relatedQuestions

## 3.9 Anpassungen LateX Template DSR

Diese Seite enthält eine Übersicht über die im LateX-Template zur Erzeugung der Variablendetailseiten des DSR notwendigen Änderungen.

### 3.9.1 nominal skalierte Variablen

- Maßzahlen ergänzen
- Struktur Häufigkeitstabelle anpassen nach Vorlage MDM (Reihenfolge Spalten, Reihenfolge Zeilen)

### 3.9.2 ordinal skalierte Variablen

- Maßzahlen anpassen (wie MDM)
- Boxplot ergänzen
- Struktur Häufigkeitstabelle anpassen nach Vorlage MDM (Reihenfolge Spalten, Reihenfolge Zeilen)

### 3.9.3 intervall skalierte Variablen

- Maßzahlen ergänzen
- Boxplot ergänzen
- Häufigkeitstabelle ergänzen

### 3.9.4 verhältnis skalierte Variablen

- Maßzahlen anpassen (wie MDM)
- Struktur Häufigkeitstabelle ändern (wie MDM)
- sicherstellen, dass alle verhältnis-Variablen ein Histogramm besitzen (Bsp. aocc226h)

### 3.9.5 Skalenniveau-übergreifend

- sicherstellen, dass die Häufigkeitstabelle bei Variablen ohne value-labels wie gewünscht dargestellt wird (Bsp. astu061b)
- sicherstellen, dass Variablen mit String-values wie gewünscht dargestellt werden (Bsp. bfec162h\_g3o)
- sicherstellen, dass Variablen mit mehr als einer zugehörigen Frage wie gewünscht dargestellt werden

## 3.10 Datensatzreport (Lektorat)

Diese Seite ist aktuell nicht relevant, da im Moment der redaktionelle Teil des Datensatzreports nicht verwendet wird. Eventuell wird es in Zukunft wieder relevant und wird daher hier dokumentiert.

Diese Seite dokumentiert das geplante Aussehen des Datensatzreport und die dafür notwendigen Änderungen bei Inhalt, Struktur und Layout sowie den Stand der vorgenommenen Anpassungen.

### 3.10.1 1 geplanter Aufbau des DSR

#### Übersicht

Im folgenden wird der Aufbau des DSR skizziert. Aktuell wird auf den redaktionellen Teil und den Anhang verzichtet.

## Titelseite

```
1.1 Titelblatt
1.2 Zweite Seite
1.3 Inhaltsverzeichnis
1.4 Verzeichnis der Variablenseiten (+ Verlinkung auf entsprechende Seite)
```

## Redaktioneller Teil

```
2.1 Einleitung
2.2 Informationen zum Datensatz / Datensatzstruktur
2.3 Variablenbenennung, Vergabe von Labels
2.4 Codierung fehlender Werte
2.5 Lesehilfe / Legende zu Variablendetailseiten
```

## Variablendetailseiten

## Anhang

```
4.1 Tabelle Übersicht über Panelvariablen
4.2 Tabelle Übersicht über vercodete ursprünglich offen erfragte Variablen
4.3 Tabelle Übersicht über anonymisierte Variablen
4.4 Tabelle Übersicht über generierte Variablen (andere Gründen als Anonymisierung)
```

## bisherige Festlegungen

- Redaktioneller Teil und Anhang werden jeweils eigene LaTeX-Dateien, die in die main.tex eingebunden werden
- Variablenamen im Anhang werden auf entsprechende Variablenseite im DSR verlinkt

## noch offene Aspekte

- Existenz des Anhangs ist abhängig von vorhandenen Personal- und Zeitressourcen; Inhalt des Anhangs muss noch endgültig festgelegt werden
- enthält der Anhang irgendeine Übersicht/Text etc. zu Gewichten?
- soll es ein Verzeichnis der eingesetzten Codierlisten geben (mit oder ohne Links auf die Listen)?
- wird Titelei eigene LaTeX-Datei oder Bestandteil der main.tex

## aktuelle ToDos

- Inhalt Titelblatt und zweite Seite festlegen
- Redaktionellen Teil schreiben (2.1 bis 2.4)
- Inhalt und Darstellung Lesehilfe (2.5) festlegen
- Abbildungs- und Tabellenverzeichnis rausnehmen (klären mit LaTeX-Firma, ob wir grundlegender Typ des Dokumentes ändern, kein *Buch* mit linker/rechter Seite etc.)

### 3.10.2 2 Variablendetailseiten

#### bisherige Festlegungen

Skalenniveau/Block	nominal	ordinal	intervall	verhältnis
Variablendetails	ja	ja	ja	ja
Fragedetails	ja	ja	ja	ja
Maßzahlen	wie MDM	wie MDM	wie MDM	wie MDM
Boxplot	nein	ja	ja	ja
Häufigkeitstabelle	wie MDM	wie MDM	wie MDM	wie MDM
Histogramm	nein	nein	nein	ja

- *Eingangsfiler* und *Generierungsregel* werden nur angezeigt, wenn sie eine noch zu definierende String-Länge nicht überschreiten. Bei längeren Strings in diesen Attributen wird aufs MDM verlinkt (teilweise Darstellung im DSR oder dort nur Standardtext?)

#### noch offene Aspekte

- Block Variablendetails: Sichtbarkeitsbedingung für Attribute *Beschreibung*, *Panelvariablen* und *Eingangsfiler*
- Block Variablendetails: Schreibweise der Skalenniveaus im DSR festlegen (Groß- oder Kleinschreibung von *intervall* und *verhältnis* oder *intervall skaliert*)
- Block Fragedetails: Sichtbarkeitsbedingung für Attribute *Einleitung der Frage* und *Ausfüllanweisung*
- Block Maßzahlen: entscheiden, ob *Devianz* bleibt oder raus soll
- lange *Eingangsfiler* und *Generierungsregel*: festlegen, was im DSR angezeigt wird (s.o.)

#### aktuelle ToDos

- [[Anpassungen LaTeX-Template | Anpassungen LateX-Template DSR]], um Anforderungen (vgl. Tabelle) zu erfüllen
- Sonderfall-Variablen identifizieren, bei denen abweichend von den Vorgaben der Tabelle bestimmte Elemente (z.B. Boxplot bei aocc12) nicht angezeigt werden soll. Sonderfallklassen bilden und im LaTeX-Template behandeln.
- Block Fragedetails: festlegen, welcher Text erscheinen soll, wenn es keine zugehörige Frage gibt
- LaTeX-Template: Fallunterscheidungen für Attribute *Eingangsfiler* und *Generierungsregel* bezüglich String-Länge ergänzen
- LaTeX-Template: Tausendertrennzeichen einbauen
- LaTeX-Template: Darstellung des Histogramms anpassen, so dass die Balken verbunden sind
- Daten: ungewünschte Zeilenumbrüche entfernen

### 3.10.3 3 Layout-Anpassungen

#### aktuelle ToDos

- festlegen, an welchen Stellen das Layout angepasst werden soll und rausfinden, ob das noch im Rahmen des externen LaTeX-Auftrags möglich ist

### 3.10.4 4 Verschiedenes

#### bisherige Festlegungen

- der Datensatzreport wird zunächst nicht übersetzt. Falls die neuen Kolleginnen (Übersetzerinnen) in den nächsten Wochen noch Ressourcen frei haben, wird dieser Aspekt neu diskutiert.

#### noch offene Aspekte

- wie verfahren wir mit Episodendatensätzen? Datensatzreport macht wenig Sinn, evtl. stattdessen nur ein PDF mit einer Beschreibung, was ein Episodendatensatz ist, wie die Struktur des konkreten Datensatzes aussieht (Anzahl Fälle, Anzahl Variablen etc.) und wie man die Episodendaten an den Personendatensatz anspielt, praktisch eine Miniversion des redaktionellen Teils des normalen Datensatzreports

## 3.11 Fragen (questions)<sup>1</sup>

### Übersicht

Zu den einzelnen Fragen eines Instruments (sprich: Fragebogen) können Sie Informationen in das MDM übermitteln, in welchem dann für jede Frage folgende Übersichtsseite erstellt wird:

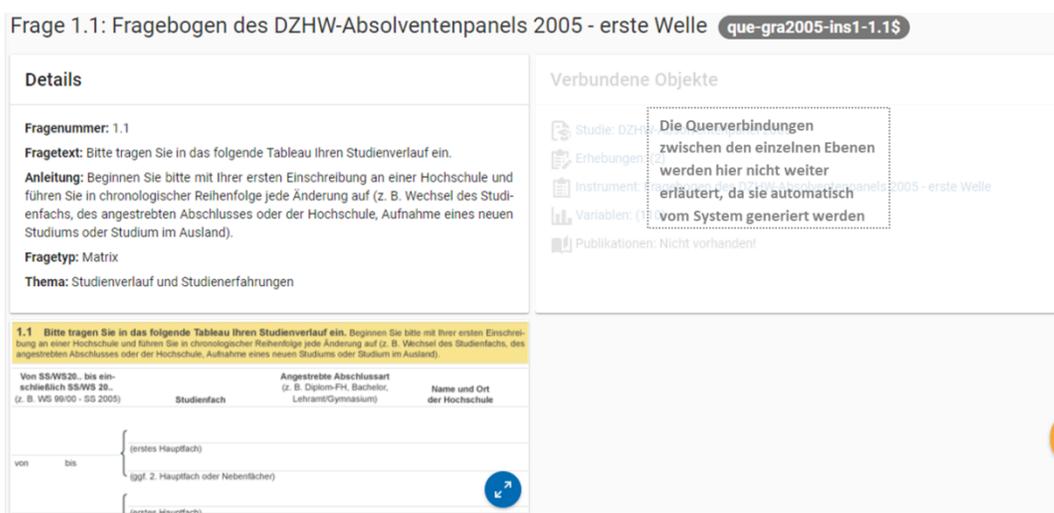


Abb. 3.10: Fragenübersicht im MDM am Beispiel der Frage 1.1 des Fragebogens der ersten Welle im Absolventenpanel 2005

Auf dieser Ebene werden Informationen über alle Fragen für jedes einzelne Erhebungsinstrument einer Studie abgegeben. Der Einspeisungsprozess dieser Informationen hängt vom Typ des Erhebungsinstrumentes ab. Während Daten aus Onlinebefragungen, die mit ZOFAR, dem Datenerhebungssystem den DZHW, durchgeführt wurden, direkt aus dem System heraus extrahiert werden (siehe **Questions (ZOFAR)**), müssen Daten aus allen anderweitig durchgeführten Befragungen – sowohl andere Onlinebefragungen als auch PAPI-Befragungen – manuell erfasst werden (siehe **Questions (manuell)**). Im Folgenden werden beide Vorgehensweisen schrittweise beschrieben.

<sup>1</sup> Metadaten auf Fragenebene sind erst ab der 2. Dokumentationsstufe gefordert. Die Erläuterungen zu den drei verschiedenen Dokumentationsstandards finden Sie in den Dokumenten „Anforderungen an Daten und Dokumentation im FDZ des DZHW“.

### 3.11.1 Fragestruktur

Fragen sind gekennzeichnet durch einen einleitenden/übergreifenden Fragetext, sowie eine „natürliche“ sichtbare Abgrenzung gegenüber anderer Fragen und eine meist „erkennbare“ Nummerierung. Es wird zwischen fünf Fragetypen differenziert:

- **Single Choice:** Auf die Frage kann nur mit einer Antwortmöglichkeit geantwortet werden (z.B. Einfachauswahl aus mehreren Antwortmöglichkeiten oder Angabe eines numerischen Wertes).
- **Mehrfachnennung:** Für die Frage gibt es eine Auswahl an Antwortmöglichkeiten bei denen eine oder mehrere ausgewählt werden können.
- **Itematterie:** Besitzt überleitenden Fragetext, welche jeweils weitere Items mit den gleichen Antwortmöglichkeiten besitzen.
- **Matrix:** Ist ein komplexer Fragetyp in dem viele Unterfragen geschachtelt werden können und die nicht durch die anderen Fragetypen abgedeckt werden (z.B. Tableaufragen des Absolventenpanels ).
- **Undocumented:** Die „Restkategorie“, sollte die Frage nicht mit einem der oben genannten Fragetypen abzubilden sein.

### 3.11.2 Questions (manuell)

Um json Dateien zu erzeugen muss zuerst einmal eine Exceltabelle ausgefüllt werden. Die Exceltabelle hat die beiden Tabellenblätter *questions* und *images*. Spaltennamen und Ausfüllanweisungen sind im nächsten Abschnitt zu finden.

Zusätzlich können zu jeder Frage ein oder mehrere Bilder vorhanden sein (im Gegensatz zu früher nicht mehr verpflichtend). Es ist aber zu beachten, dass im Moment die R-Skripte noch nicht dahingehend aktualisiert wurden. Wie Fragebilder aus Ragtime-Dateien extrahiert werden können, wird erklärt: [Bilderfassung aus RagTime](#) Eine Anleitung zum Ausschneiden von Bildern aus pdf Dateien ist hier zu finden [Bilderfassung aus PDF Dateien](#).

#### Excel-Tabelle

Um Metadaten auf der Fragenebene in manueller Weise zu erfassen, müssen Sie die Excel-Datei *questions.xlsx* ausfüllen, welche die beiden Tabellenblätter *questions* und *images* beinhaltet. Sie können alle Fragen aus allen Erhebungsinstrumenten in einer einzigen Exceltabelle erfassen:

Tabelle 3: Ausfüllanweisungen für die Excel-Tabelle „questions“

<b>Tabellenblatt 1: questions</b>		
Es können mehrere Fragen eingetragen werden (= mehrere Zeilen möglich, eine Frage pro Zeile)		
<b>Spalten- über- schrift</b>	<b>Muss ich das ausfü- llen?</b>	<b>Was muss ich eintragen?</b>
indexIn- Instru- ment	Ja	Nummer der Frage im Fragebogen, nach der die Reihenfolge festgelegt wird (ganzzahlig)
ques- tion- Num- ber	Ja	Fragennummer, idealerweise selbsterklärend aus Instrument (z. B. 1.1). Format: 0-9, a-z, Umlaute, ß, ,, -
instru- ment- Num- ber	Ja	Nummer des Instruments
ques- tions- Text.de/en	Ja	„Übergreifender“ Fragetext, bei Itembatterien oder komplexen Fragen der einleitende Fra- getext. Bei „einfachen“ Fragetypen der komplette Fragetext.
in- structi- on.de/en	Nein	wenn vorhanden, Anweisungstext der Frage
intro- ducti- on.de/en	Nein	wenn vorhanden, Einleitungstext der Frage
ty- pe.de/en	Ja	de: „Einfachnennung“, „Offen“, „Mehrfachnennung“, „Itembatterie“ oder „Matrix“ (ei- ne Anleitung zur Einteilung der verschiedenen Fragetypen kann unter <a href="https://github.com/dzhw/metadatamanagement/files/1421895/Anleitung_Vergabe_Fragetypen.docx">https://github.com/dzhw/metadatamanagement/files/1421895/Anleitung_Vergabe_Fragetypen.docx</a> gefunden werden) en: „Single Choice“, „Open“, „Multiple Choice“, „Item Set“ or „Grid“.
to- pic.de/en	Nein	Themenblock, in dem die Frage im Instrument eingeordnet ist (idealerweise direkt aus In- strument entnehmbar)
succes- sorNum- bers	Nein	Fragennummern der nachfolgenden Frage(n) (Angabe in einer Zeile durch Komma getrennt)
tech- nical- Repre- sentati- on.type	x*	Herkunft des Codeschnipsels (z. B. „ZOFAR-Question Markup Language“)
tech- nical- Repre- sentati- on.language	x*	Technische Sprache des Codeschnipsels (z. B. XML)
tech- nical- Repre- sentati- on.source	x*	Codeschnipsel, um Frage technisch abbilden zu können (z. B. QML-Schnipsel)
addi- tional- Ques- tion- Text.de/en	Nein	Weitere Ausführungen der Frage, die nicht im Fragetext stehen, wie z. B. der Itemtext (bei Itembatterien) oder Antworttext (bei Mehrfachnennungen). Aktuell ist diese Information für den Nutzenden des MDM nicht sichtbar, sondern wird nur bei einer Volltextsuche be- rücksichtigt.
annotati- ons.de/en	Nein	Anmerkungen zur Frage

x\* = nur, wenn technicalRepresentation vorhanden (wird dann automatisch von ZOFAR geliefert)

Tabellenblatt 2 ist nur auszufüllen falls es Bilder zu den Fragen gibt.

Tabellenblatt 2: images		
Es können mehrere Bilder eingetragen werden (= mehrere Zeilen möglich, ein Bild pro Zeile)		
Spaltenüberschrift	Muss ich das ausfüllen?	Was muss ich eintragen?
fileName	Ja	Dateiname des Bildes (z.B. „1.1_1.png“)
questionNumber	Ja	Dem Bild zugeordnete Fragennummer
instrumentNumber	Ja	Nummer des zum Bild gehörenden Instruments
language	Ja	Sprache des Bildes <i>Bitte verwenden Sie eine Abkürzung nach ISO 639-1_: z. B. „de“, „en“</i>
indexInQuestion	Ja	Auf das wievielte Bild der Frage bezieht sich die Zeile? (Liegt pro Frage nur ein Bild vor, steht hier immer 1)

Mit dem zweiten Tabellenblatt *images* erfassen Sie Informationen zu den Fragebildern, welche Sie für jede Frage mit hochladen können. Falls Bilder vorhanden sind, müssen diese im Png Format vorliegen. Die Fragebilder können z.B. mit Ragtime extrahiert werden, sofern der Fragebogen auch mit Ragtime erstellt wurde. Ansonsten lassen sich die Fragebilder auch aus einer PDF-Datei erstellen.<sup>2</sup> Anleitung für beiden Varianten finden Sie unter *Bilderfassung aus RagTime* und *Bilderfassung aus PDF Dateien*.

Die fertig ausgefüllte Excel-Datei sowie ggfs. die Bilder zu den Fragen speichern Sie dann in dem Ordner, den das FDZ für Sie vorbereitet hat. Das FDZ greift daraufhin auf die Dateien zu, verarbeitet sie weiter und lädt die Metadaten für die Frageebene dann selbst ins MDM.

### 3.11.3 Generierung der json Dateien mit R

Doku befindet sich im Aufbau und ist nur für FDZ-MitarbeiterInnen relevant.

Momentan liegen die Question-Exceldateien der Projekte, sowie die Skripte zur Erzeugung der json Dateien im Verzeichnis \\faust\Abt4\FDZ\Querschnittsaufgaben\Metadaten\Erzeugen. Der Aufbau ist wie folgt:

```

|-- Projekte
  |-- projectName
    |-- questions
      |-- out
        |-- projectName.xlsx
|-- Skripte
  |-- question-generation.R
  |-- sort-images.R
  |-- R
    |-- question-generation_main.R
    |-- utils
      |-- question-generation_functions.R

```

Um json Dateien für ein neues Projekt zu erzeugen, muss zuerst ein Projektordner angelegt werden. Außerdem muss die Question-Exceltabelle des Projektes ausgefüllt werden (z.B. projectName.xlsx mit den beiden Tabellenblätter questions und images). Außerdem muss der Ordner out angelegt werden. Danach question-generation.R öffnen und bei project den Projektnamen anpassen, z.B. `project <- "gra2005"`. Das Skript z.B. mit Strg+a -> Strg+Enter ausführen. Im Ordner out sind nun die json Dateien für den Import in der vorgegebenen Ordnerstruktur zu finden.

<sup>2</sup> Bitte beachten Sie, die dokumenteigenen Metadaten der PDF-Dateien vorab zu löschen (vgl. *Anhänge*).

### Einsortierung der Bilder in die Ordnerstruktur

Nun müssen die Bilder noch in die Ordnerstruktur eingepflegt werden. Dafür kann das R-Skript `sort-images.R` verwendet werden. Die pngs zu den Fragen (es können auch mehrere pngs zu einer Frage vorliegen) und das Tabellenblatt `images` der Exceltabelle werden dafür benötigt. Nähere Erklärungen zur Sortierung der Bilder sind im R-Skript selbst zu finden.

Die fertigen jsons und Bilder können nun zu Github ins jeweilige `$projectname-metadata-repository` kopiert werden.

### 3.11.4 Questions (Zofar)

Bei Onlinebefragungen mit Zofar können die Metadaten für Fragen automatisch extrahiert werden (`.jsons` + `.pngs`).

Der Prozess befindet sich gerade im Aufbau. . .

## 3.12 Variablen (variables)<sup>1</sup>

### Übersicht

Anhand der Informationen, die Sie auf Ebene der Variablen abgeben, wird für jede Variable eine Übersichtsseite im MDM erstellt:

Die Erstellung der Variablenebene beinhaltet einerseits recht viel Aufwand, da für jeden Datensatz eine eigene Excel-Tabelle mit Informationen zu allen Variablen geliefert werden muss. Viele Informationen müssen manuell eingetragen werden, einige können – sofern die Befragung über Zofar stattgefunden hat – auch direkt aus Zofar (das Onlinebefragungstool des DZHW) extrahiert werden oder sogar aus der Excel-Tabelle der Frageebene importiert werden.

Die Variablenebene ist andererseits sehr wertvoll im Hinblick auf die Nachnutzbarkeit der Forschungsdaten. Wenn Metadaten auf dieser Ebene vorhanden sind, können die dazugehörigen Daten auch aus inhaltlicher Sicht umfassend durchsucht werden, sodass das Analysepotential auch für sehr spezielle Fragestellungen direkt sichtbar wird.

Für die Darstellung der Metadatenaufnahme auf Variablenebene gilt es noch folgende Dinge zu beachten:

- Wenn Sie mehrere Datensätze liefern: Es darf kein Variablenname doppelt vorkommen.
- Missings müssen global definiert sein, d. h. sie müssen für alle Variablen eines Datensatzes gelten.

### Excel-Tabelle

Ausfüllen müssen Sie je nach Anzahl der Datensätze mindestens eine Excel-Datei mit dem Namen `vimport_ds***Nr*.xlsx`, wobei die „Nr.“ im Dateinamen der Nummer des dazugehörigen Datensatzes entsprechen muss, d. h. die Variablen des Datensatzes mit der Nummer 1 muss `vimport_ds1.xlsx` heißen usw. Die Datei enthält die beiden Tabellenblätter `variables` und `relatedQuestions`.

Tabelle 5: Ausfüllanweisungen für die Excel-Tabelle „`vimport_ds*Nr*`.“

<sup>1</sup> Metadaten auf Variablenebene sind erst ab der 2. Dokumentationsstufe gefordert. Die Erläuterungen zu den drei verschiedenen Dokumentationsstandards finden Sie in den Dokumenten „Anforderungen an Daten und Dokumentation im FDZ des DZHW“.

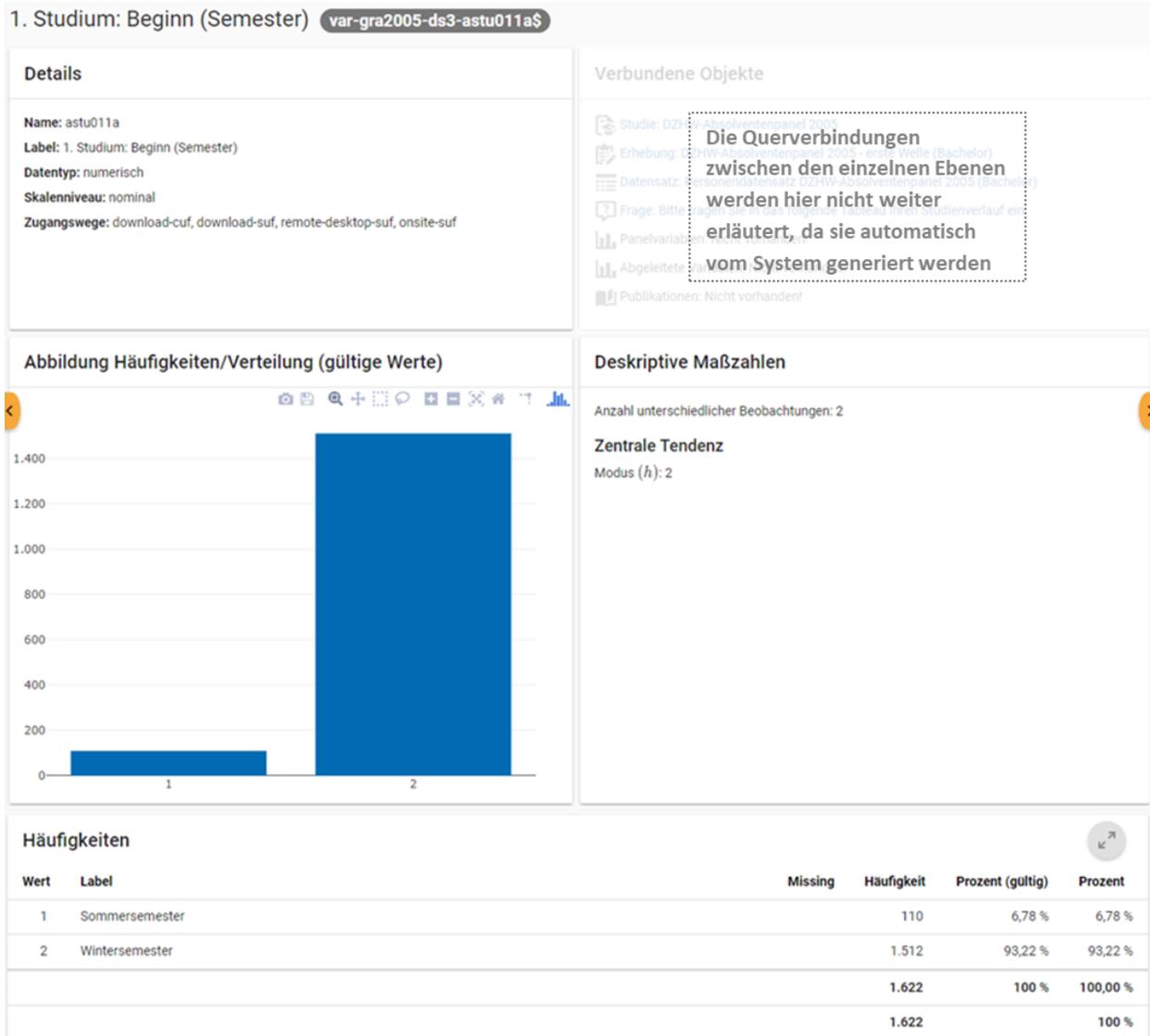


Abb. 3.11: Variablenübersicht im MDM am Beispiel der Variable „1. Studium: Beginn (Semester)“ im Absolventenpanel 2005, erste Welle (BA)

<b>Tabellenblatt 1: variables</b>		
Es können mehrere Variablen eingetragen werden (= mehrere Zeilen möglich, eine Variable pro Zeile)		
<b>Spalten-überschrift</b>	<b>Muss ich das ausfüllen?</b>	<b>Was muss ich eintragen?</b>
name	Ja	Variablenname
survey-Numbers	Ja*	Angabe aller der Variablen zugehörigen Erhebungsnummern (in einer Zelle durch Komma getrennt)
scale-Level.de/en	Ja	de: „nominal“, „ordinal“, „intervall“ oder „verhältnis“ en: „nominal“, „ordinal“, „intervall“ or „ratio“
panel-Identifizier	Nein*	Identifizier zur eindeutigen Zuordnung von Panelvariablen. Präfix muss aus der Projekt-ID + Nummer des Datensatzes bestehen (Beispiel: <i>gra2005-ds1</i> ), der hintere Teil des Identifiziers ist beliebig wählbar, muss aber eindeutig sein. Beispiel: Sind die Variablen <i>astu01a</i> und <i>bstu01a</i> aus dem 1. Datensatz des Projekts <i>gra2005</i> Panelvariablen, so könnte der Identifizier <i>gra2005-ds1-stu01a</i> lauten.
annotations.de/en	Nein	Anmerkungen zur Variablen
access-Ways	Ja*	Mögliche Zugangswege: Download-CUF, Download-SUF, Remote-Desktop-SUF, On-Site-SUF. Bei mehreren Zugangswegen sind den verschiedenen Zugangswegen entsprechend Spalten vorhanden, die mit „nicht verfügbar im ...“ überschrieben sind. Für jede Variable muss dann ein „x“ gesetzt werden, wenn diese über den jeweiligen Zugangsweg nicht vorhanden ist.
filter-Details.description.de/en	Nein	Verbalisierte Beschreibung des Variablenfilters
filter-Details.expression <sup>2</sup>	Ja, wenn Filter vorhanden	Regel, die in der angegebenen „Sprache“ (.expressionLanguage) beschreibt, welche Teilpopulation zu dieser Variable hin gefiltert wurde (auch verschachtelte Filterführung wird beachtet (PAPI))
filter-Details.expressionLanguage <sup>2</sup>	Ja, wenn Filter vorhanden	Sprache des Filterausdrucks: „Stata“
generation-Details.description.de/en	Nein	Beschreibung, wie die Variable erzeugt wurde, wenn sie nicht direkt aus dem Fragebogen abgelesen werden kann (Beispiel, siehe Abschnitt „Generierungsdetails“)
generation-Details.rule.de/en	Ja, wenn Variablen generiert	Regel, die in der angegebenen „Sprache“ (.ruleExpressionLanguage) beschreibt, wie die Variable erzeugt wurde (Beispiel, siehe Abschnitt „Generierungsregel (Stata)“)
<b>3.12. Variablen (variables)<sup>1</sup></b>		
ge-	Ja,	Sprache der Erzeugungsregel: „Stata“ oder „R“

\* Wenn eigene Konventionen verwendet werden, muss das Feld manuell ausgefüllt werden. Bei Verwendung von FDZ-eigenen Schemata kann dieses Feld auch leer gelassen werden.

Tabellenblatt 2: relatedQuestions		
Variablen, die mit mehreren Fragen verbunden sind, können mehrfach aufgeführt werden. Variablen, die keiner Frage (oder keinem Instrument) zugeordnet sind, müssen nicht eingetragen werden.		
Es können mehrere verbundene Fragen eingetragen werden (= mehrere Zeilen, eine verbundene Frage pro Zeile)		
Spaltenüberschrift	Muss ich das ausfüllen?	Was muss ich eintragen?
name	Ja	Variablenname
relatedQuestionStrings.de/en	Nein	Text, der den Frageinhalt der Variable darstellt. Also Fragetext der dazugehörigen Frage plus evtl. weitere Ausführungen wie bspw. der Itemtext (bei Itembatterien) oder der Antworttext (bei Einfach- oder Mehrfachnennungen)
questionNumber	Ja	Nummer der zur Variablen zugehörigen Frage im Fragebogen
instrumentNumber	Ja	Nummer des zur Variablen zugehörigen Fragebogens

Dem Namen entsprechend wird aus den Informationen des zweiten Tabellenblatts die Verknüpfung zwischen einer Variablen und der dazugehörigen Frage aus dem Erhebungsinstrument erstellt. Für eine nachvollziehbare Dokumentation dieser Verbindung ist die Erstellung eines Variablenfragebogens sehr hilfreich. Aus diesem kann die Verknüpfung aus Variable und Frage problemlos abgelesen werden. Abb. 3.12 zeigt beispielhaft, dass den Variablen *astu08a* bis *astu08e* die Frage 1.8 zugeordnet ist.

**1.8 Waren für Sie studienbegleitende Praktika oder Praxissemester vorgeschrieben?**  
(Mehrfachnennung möglich)

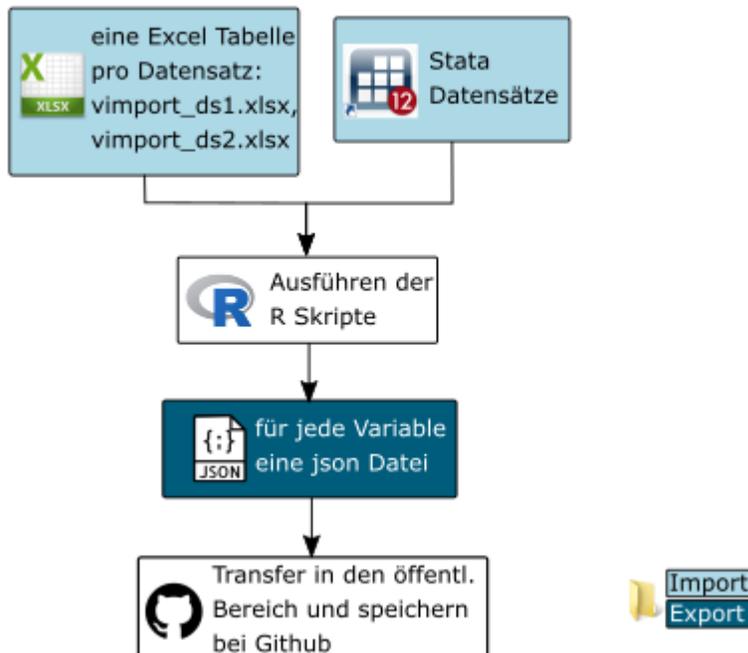
Ja, Praktika an der Hochschule (z. B. Laborpraktikum)	<b>astu08a</b>
Ja, externe Praktika (z. B. Betriebspraktikum)	<b>astu08b</b>
Ja, Praxissemester .....	<b>astu08c</b>
Ein Praktikum war zwar vorgeschrieben, musste von mir aber nicht absolviert werden (z. B. wegen der Anerkennung einer Ausbildung)	<b>astu08d</b>
Nein .....	<b>astu08e</b>

Abb. 3.12: Ausschnitt aus dem Variablenfragebogen des Absolventenpanels 2005, erste Welle, Frage 1.8

<sup>2</sup> Nur in der Dokumentationsstufe 3 gefordert. Die Erläuterungen zu den drei verschiedenen Dokumentationsstandards finden Sie in den Dokumenten „Anforderungen an Daten und Dokumentation im FDZ des DZHW“.

Außer der/den Excel-Tabelle/n müssen Sie für jede Tabelle noch den zugehörigen Stata-Datensatz liefern, aus dem die Variablen stammen. Diese Dateien speichern Sie dann in dem Ordner, den das FDZ für Sie vorbereitet hat. Das FDZ greift daraufhin auf die Dateien zu, verarbeitet sie weiter und lädt die finalisierten Metadaten für die Variablenebene dann selbst ins MDM.

### 3.13 Erstellung der Variable-JSON Dateien



Die Erstellung der Variablen JSONs erfolgt komplett im geschützten Bereich. Benötigt werden pro Datensatz ein zugehöriger Stata-Datensatz und eine Exceltabelle. Die Exceltabelle (vimport\_dsNR.xlsx) enthält die beiden Tabellenblätter variables und relatedQuestions. Pflichtspalten und zugehörige Ausfüllanweisungen werden im folgenden Abschnitt beschrieben.

Es ist erlaubt die Exceltabellen um weitere optionale Spalten zu erweitern, z.B. Varname\_alt, Var\_Erh, Var\_Thema, Var\_Nr, Var\_Indiz, Var\_g, Var\_h, Var\_x, Var\_p, Var\_v, Var\_Zugang, Varlabel\_alt, Varlabel\_neu, On-Site, Remote-Desktop, Download-SUF, Download-CUF, AIP, SIP, delete, ...

Momentan liegen die Import Dateien der Projekte, sowie die Skripte zur Erzeugung der JSONs im geschützten Bereich unter Q:Variablenexport. Der Aufbau der Ordnerstruktur ist wie folgt:

```

|--Variablenexport
  |--Projekte
    |--gra2005
      |--variablesToJsons.bat
      |--output
        |--ds1
        |--ds2
      |--data-raw
        |--stata
          |--ds1.dta
    
```

(Fortsetzung auf der nächsten Seite)

```

|--ds2.dta
|--excel
|--vimport_ds1.xlsx
|--vimport_ds2.xlsx
|--conditions.xlsx
|--variable-generation_productive
|--variablesToJson.bat.tmpl

```

Um json Dateien für ein neues Projekt zu generieren, muss zunächst ein Ordner für das neue Projekt angelegt werden und die oben gezeigt Ordnerstruktur aufgebaut werden. Im Ordner stata befinden sich die jeweiligen Stata Datensätze (ds1, ds2, ds3, ...) und im Ordner excel die zugehörigen Exceltabellen mit den beiden Tabellenblättern variables und relatedQuestions (vimport\_ds1.xlsx, vimport\_ds2.xlsx, vimport\_ds1.xlsx, ...), sowie die Datei mit den missing conditions (conditions.xlsx). Zum Generieren der json Dateien das R-Skript variablesToJson.bat.tmpl in den Projektordner kopieren, das .tmpl entfernen, die Datei anpassen und danach ausführen.

Es ist möglich die Missing Bedingungen für numerische und string Variablen in der datei conditions.xlsx anzupassen. Außerdem können in der batch-Datei Variablennamen angegeben werden, die im MDM keine Verteilung bekommen sollen. Dies sind z.B. id Variablen. Variablen mit accessway not-accessible müssen hier nicht eingetragen werden.

### Missing Conditions

In der Exceltabelle conditions.xlsx können für numerische und string Variablen Missingbedingungen angegeben werden. Die Exceltabelle enthält die beiden Tabellenblättern missingConditionNumeric und missingConditionString. Es ist möglich für numerische und string Variablen jeweils mehrere Bedingungen anzugeben. Die Bedingungen werden mit ODER verknüpft. Das heißt, wenn eine der Bedingungen für einen Wert zutrifft, wird dieser Wert als Missing gewertet. Die verfügbaren Operatoren können in der Exceltabelle über ein Drop-Down Menü ausgewählt werden und sind im Tabellenblatt list of valid operators dokumentiert.

Ein Fehler der auftreten kann ist, dass im Stata-Datensatz nicht die richtige Sprache gewählt wurde. Ist das der Fall können nicht die richtigen Wertelabel zugeordnet werden.

**Transfer in den öffentlichen Bereich** Die Datensatzordner mit den json Dateien müssen noch in den öffentlichen Bereich transferiert werden. Da es nicht möglich ist, Ordner zu transferieren, werden die Ordner gezippt (7-Zip), transferiert und im öffentlichen Bereich wieder entpackt.

Die Variable-JSON Dateien müssen anschließend bei Github in das Repository projectid-metadata in den variables Ordner hochgeladen werden. Siehe z.B. <http://github.com/dzhw/gr2005-metadata/>. Die Ordner werden anschließend auf Variablenebene ins MDM per Drag and Drop oder über den Plusbutton rechts unten hochgeladen.

## 3.14 Variables (Zofar)

Bei Onlinebefragungen mit ZOFAR können fragenbezogene Metadaten auf Variablenebene automatisch extrahiert werden. Eine .csv Tabelle die den Variablennamen, die Instrumentnummer, die Fragennummer und den relatedQuestionString (Fragetext + zugehöriger Variablentext) enthält, wird geliefert.

Der Prozess befindet sich im Aufbau...

## 3.15 Related Publications

### Arbeiten mit der Citavi-Datenbank

Für dieses Objekt wird eine Citavi-Datenbank angelegt. Diese liegt unter: *faust-Abt4FDZQuerschnittsaufgabenMetadatenErzeugenLiteraturexportrelatedPublication*.

Die Citavi-Einträge lassen sich exportieren, indem man einen Eintrag in der Literaturübersicht markiert (linke Seite) und Str+Alt+t drückt. Die Tabellenansicht öffnet sich und durch klicken auf Spalten (oben links) kann ausgewählt werden, welche Spalten exportiert werden sollen. Aus der Tabellenansicht kann die Datei jetzt nach Excel als relatedPublications.xls exportiert werden (Datei > nach Microsoft Excel exportieren). Einige Spaltennamen müssen evtl. später noch manuell umbenannt werden (z.B. BibTeXKey wird zu id). Die Excel-Tabelle wird [hier](https://github.com/dzhw/metadatamanagement-io/tree/master/references/relatedPublications) gepflegt.

Eigenschaft	Ausfüllanweisung	muss ausgefüllt werden?
id	von Citavi erzeugt BibTex-Key	ja
sourceReference	Quellangabe der Publikation (default)	ja
publicationAbstract	Zusammenfassung	nein
doi	doi der Publikation	nein
sourceLink	valide URL	nein
title	Titel	ja
authors	Autoren (Nachname1, Vorname1; Nachname2, Vorname2)	ja
year	Jahr der Veröffentlichung (muss kleiner oder gleich dem aktuellen Jahr sein)	ja
abstractSource.de/en	??	nein
studyIds	Studien-Ids, der zur Publikation gehörenden Studie	Wenn keine studySerieses vorhanden -> ja
dataSetIds		nein
instrumentIds		nein
surveyIds		nein
variableIds		nein
questionIds		nein
studySerieses.de	mindestens eine studyId oder mindestens 1 studySeries	Falls vorhanden ja
language	Sprache der Publikation (2-Buchstaben Code nach ISO 639-1)	ja

### 3.16 Datensatzreport erzeugen

Wenn ein Datensatz und die zugehörigen Variablen im MDM vorliegen, kann mit Hilfe des MDMs ein Datensatzreport erstellt werden. Hierzu wird das [Template](#) auf den Datensatz im MDM gezogen. Nach einiger Zeit (je nach Anzahl an Variablen länger als eine Minute) erfolgt ein Download. Die resultierenden Dateien werden von FDZ-MitarbeiterInnen zu einem PDF kompiliert. Dokumentation zum Umgang mit dem dafür benötigten Docker-Image folgt.

### 3.17 Prüfung der Jsons nach Umstellung der Generierungsskripte

Es ist mögliche, zwei jsons online miteinander zu vergleiche, z.B. [hier](#) . Dazu einfach die Texte aus den zu vergleichenden jsons kopieren und in die jeweiligen Felder einfügen.

#### 3.17.1 Struktur der Jsons

Die vorgegebene Struktur der json Dateien ist hier zu finden:

- [Link to Variable Import File](#)
- [Link to Question Import File](#)

### 3.17.2 Variablenprüfung

Beim Prüfen ist besonders drauf zu achten, Variablen mit:

- verschiedenen Skalenniveaus (besonders **intervall** und **verhältnis**, da viele Maßzahlen)
- verschiedenen Datentypen
- einer oder mehreren zugeordneten Fragen

auszuwählen.

### 3.17.3 Allgemeine Prüfung

- ist ein Attribut leer muss der Wert null sein (alternativ kann das Attribut auch aus dem json herausgenommen werden)
  - richtig: „panelIdentifier“: null
  - falsch: „panelIdentifier“: „“
  - falsch: „validResponses“: []
- der Datentyp muss richtig angegeben werden (wie im Beispiel Import File) -> die häufigsten Datentypen sind:

Datentyp	Beispiel
string	„storageType“: „integer“
Integer	„scaleLevel“: {„en“: „interval“, „de“: „intervall“}
integer	„indexInDataSet“: 642
double	„standardDeviation“: 2.7881
list	„accessWays“: [„download-cuf“, „download-suf“, „remote-desktop-suf“, „onsite-suf“]
boolean	„containsAnnotations“: false

## 3.18 Skalenniveau prüfen

## 3.19 Skalenniveau Diese Informationen dienen dazu die verschiedenen

Skalentypen voneinander unterscheiden zu können und Variablen selbständig einordnen zu können. Das Forschungsdatenzentrum (**FDZ**) hat sich dazu entschieden vier Skalentypen zu unterscheiden `_Nominal-`, `_Ordinal-`, `_Intervall` <> `_` und `_Verhältnisskala` <> `_`. Siehe hierzu [Wikipedia-Eintrag](#)

### 3.19.1 Skalentypen `**_ \ **\ Nominalskala` <> `_` - Nominale Messung besteht

in der Erstellung einer einfachen Klasseneinteilung, die `_` jedes <> `_` Objekt genau `_` einer <> `_` Klasse zuordnet: Weder darf ein bestimmtes Objekt überhaupt nicht zugeordnet werden, noch darf ein Objekt mehreren Klassen zugeordnet werden.

Beispiele (Variable) | :—:—|| (=/) | Geschlecht, Universitätsnamen, Studienfächer

**\*\*\_ \ \*\*\ Ordinalskala <>\_\_** - Sie besitzt eine Rangordnung der Objekte in Bezug auf die interessierende Dimension. Die entsprechend zugeordneten Zahlen müssen diese Rangordnung wiedergeben.

Beispiele (Variable) :— | (= / ; </> ) | Schulnoten, höchster Bildungsabschluss, Zufriedenheitsskalen

**\*\*\_ \ \*\*\ Intervallskala <>\_\_** - ist ein Skalenniveau in der Statistik. Sie zählt zum metrischen Messniveau, da sich die Ausprägungen dieses Skalenniveaus quantitativ mittels Zahlen darstellen lassen. Insbesondere bedeutet das auch, dass Rangunterschiede und Abstand zwischen Werten gemessen werden können; das heißt, quantitative Merkmale gehen in ihren Anforderungen über ordinale oder gar nominale Eigenschaften hinaus.

Beispiele (Variable) | :— :— | (= / ; </> ; + | Temperatur (Celcius, Fahrenheit), Zeitskala (Datum)

**\*\*\_ \ \*\*\ Verhältnisskala <>\_\_** - ist das höchste Skalenniveau in der Statistik. Bei ihr handelt es sich um eine metrische Skala, im Unterschied zur Intervallskala existiert jedoch ein absoluter Nullpunkt

| logische / mathematische Operationen | Beispiele (Variable) | Stata-Beispiel Syntax | :— :— | (= / ; </> ; + / ; ÷ | Alter, Einkommen

## Übersicht

|| Verschiedenartigkeit | natürl. Reihenfolge | Interpretierbarkeit der Verhältnisse der Differenzen | natürl. Nullpunkt | natürl. Einheit |

:— :— :— | ja | nein | nein | nein | nein | ja | ja | nein | nein | nein | ja | ja | ja | nein | nein | ja | ja | ja | ja | ja |

## FAQ (ausfüllen) : - Was ist mit „Sonstige“-Kategorien - Was mit

fehlenden Werten machen? - Prinzipiell immer konservativ (niedriges Niveau) Skalenniveau vergeben

## Typische Vergaben -

[1]: R. Schnell, P. Hill, and E. Esser. Oldenbourg, München u.a., 6., völlig überarb. und erw. Aufl. edition, (1999); S.134ff. [2]: Stata commands sind kursiv. Vor einem kursiven Ausdruck muss ein Zeilenumbruch sein; Bsp.: `_ \ sysuse <>__ auto (Zeilenumbruch) _ \ fre <>__ make`

## 3.20 Stata-Skripte

| Name | Was kann es? | Wo wird es genutzt? | Was muss manuell gemacht werden? | | ————— | ————— | —————  
| [Metadatenexport-master-ProjID.do](#) | definiert die Pfade zu den Import und Export Ordnern

### 3.20.1 Die im Skript Metadatenexport-master-ProjID.do festgelegt

Ordnung der Dateipfade muss eingehalten werden, sonst funktioniert es nicht!

## 3.21 Testen von MDM-Issues

Issues des metadatamanagement-Repos werden im Testsystem getestet: <https://metadatamanagement-test.cfapps.io>  
Für einige Issues ist eine Anmeldung notwendig.

Wie genau beim Testen vorzugehen ist, hängt stark vom jeweiligen Issue ab.

Generell sind die folgenden Punkte zu beachten:

- ist alles so umgesetzt, wie erwartet?
- verschiedene Browser benutzen (Firefox, Chrome, Internet Explorer 11 und - wenn möglich - Edge)
- sowohl deutsche als auch englische Seiten testen
- Issues, die den Upload betreffen, können nur über Chrome und Firefox getestet werden, da der Upload über IE nicht funktioniert
- bei Issues, die den Upload betreffen, auch prüfen, ob das Fehlerprotokoll nachvollziehbar ist
- nicht nur prüfen, ob etwas funktioniert, sondern auch, ob es nicht funktioniert, wenn es nicht funktionieren soll
- um den Upload mit gültigen und nicht-gültigen Daten zu prüfen: Daten im io-Testprojekt manipulieren
- überprüfen, ob Funktionen auch auf verkleinertem Bildschirm funktionieren (responsives Design)
- überprüfen, ob Funktionen auch bei Verwendung eines Handys funktionieren (auch hier versch. Browser testen)
- prüfen, ob die wiki-Dokumentation im metadatamanagement-Repo richtig angelegt wurde

Wenn **keine Fehler gefunden** wurden: Label „status: testing“ entfernen, unassignen und Issue schließen. Wenn **Fehler gefunden** wurden: Fehler im Issue möglichst genau dokumentieren (gerne auch Screenshots), label „status: testing“ entfernen und „status: development“ hinzufügen, sich selbst unassignen und den Developer assignen.

## 3.22 Datentypen

Der jeweilige Datentyp für eine Eigenschaft kann [hier](#) nachgeschaut werden.

Datentyp	Beschreibung	Excel Export	Json Export
string	Text	ein Objekt in einer Zelle	z.B.: "value" : 1
118nString	de: string en: string	2 Spalten, z.B.: annotations.de annotations.en	als Liste, z.B.: „questionText“: { „de“: „Wie alt sind Sie?“, „en“: „How old are you?“ }
Period	start: LocalDate- Time end: LocalDate- Time	2 Spalten, z.B.: fieldPeriod.start fieldPeriod.end	
Local-Date-Time	yyyy-mm-dd, z.B.: 2011-12-23		
Double	Gleitkommazahlen		
Integer	Ganze Zahlen		
List	Eigenschaft mit mehreren Objekten	Objekte werden in einer Zelle durch Komma getrennt, z.B.: download-cuf, download-suf, remote-desktop-suf, onsite-suf	als Array, z.B.: „successorNumbers“: [„1“, „2“]
Image-Type	PNG		

Dabei gibt es für strings und 118nStrings eine Obergrenze für die maximale Länge (Zeichenanzahl):

- SMALL: 32 characters

- MEDIUM: 128 characters
- LARGE: 2048 characters (2KB)
- X-LARGE: 1 MB

### 3.22.1 Vokabular

Es ist zu beachten, dass für einige Eigenschaften nur spezielles Vokabular zugelassen ist! Dieses kann entweder auf der io-wiki Seite der jeweiligen Exportfunktion (Study, Survey, DataSet, Variable, Instrument, Question) oder [hier](#) nachgeschaut werden.

Beispiel:

```
- accesWay: "download-cuf", "download-suf", "remote-desktop-suf", "onsite-suf", "not-
↪accessible"
- scaleLevel.de = "nominal", "ordinal", "intervall" oder "verhältnis"
- scaleLevel.en = "nominal", "ordinal", "interval" oder "ratio"
```

## 3.23 Die korrekte Anordnung der Dateien im Ordner

Für einen erfolgreichen Upload der Metadaten attachments müssen Sie sämtliche Dateien ihrer zugehörigen Ebene entsprechend in den vom FDZ vorbereiteten [Vorlage-Ordner](#) ablegen, welcher nach der jeweiligen Ebene benannt ist. Dieser Ordner sowie auch seine Unterordner sind mit englischen Begriffen betitelt. Der Unterordner, welche alle Anhänge enthält, heißt unabhängig von der Ebene immer „attachments“.

## 3.24 Übersicht über verschiedene R-Helfer-Skripte

Na-me	Was kann es?	Wo wird es ge-nutzt?	Was muss manuell gemacht werden?
csv2js	Erstellt aus der ProjID.csv Tabelle und den FrageNr.pdf für jede Frage eine json Datei die json Dateien werden im Ordner \questions\json\insNr gespeichert	Frage-nexport im öffent-lichen Be-reich	die Tabelle ProjID.csv muss fertig aufbereitet sein und im Ordner \questions liegen zu jeder Frage muss eine FrageNr.pdf Datei im Ordner questions\Bilder\insNr\pdf vorliegen Alternativ kann auch ocr für Bilder genutzt werden, dies muss jedoch im Skript auskommentiert werden Angabe des Projektnames im Skript Ausführen (Strg + r). Bilder sind nicht mehr verpflichtend.
re-spon-se-ra-teSVG	Erstellt für alle csv Tabellen im Ordner \csv jeweils ein deutsches und ein englisches Diagramm die Diagramme werden als responderate_de.svg und responderate_en.svg im Ordner \images gespeichert	Re-spon-se-rate Dia-gramme für den Sur-veyex-port	für eine Survey muss eine csv Tabelle erstellt werden die Angaben einghauf, einghaufcum, datumw müssen in der csv Tabelle stehen Ausführen (Strg + r)

Unter <https://github.com/dzhw/variableMetadataExtractor> findet man die Skripte zur Generierung der Variable-JSONs. Die Struktur der Dateipfade muss immer eingehalten werden (nach dem Beispiel test2017), sonst funktioniert es nicht!

### 3.24.1 Response Rate Diagram

Das Skript [responserateSVG](<http://github.com/dzhw/metadatamanagement-io/blob/master/generation/example/survey/Responserate/responserateSVG.R>) erzeugt Responserate Diagramme als SVG in deutsch und englisch für die verschiedenen Surveys. Benötigt wird dafür je eine csv Tabelle pro Survey, in der folgende Spalten enthalten sind: - einghauf = Rücklauf pro Woche - einghaufcum = Rücklauf kummuliert - datumw = Wochendatum im Format Jahr\_w\_Kalenderwoche (z.B. 2006w52)

## 3.25 Verantwortlichkeiten

Objekt	Teilobjekt	Verantwortlicher
[[Study]]	alles	aufbereitendes Projektteam
[[Survey]]	alles	aufbereitendes Projektteam
[[DataSet]]	alles	aufbereitendes Projektteam
[[Variable]]	vimport.xlsx	aufbereitendes Projektteam
	variable.xlsx	aufbereitendes Projektteam
	variable.json	Metadatenverantwortlic her
[[Instrument]]	alles	aufbereitendes Projektteam
[[Question]]	qimport.xlsx	aufbereitendes Projektteam
	question.json	Metadatenverantwortlic her
	question.bilder	Metadatenverantwortlic her
[[Related Publication]]	alles	Publikationsverantwortlicher (Karsten/Bene)

## 3.26 Projektrelease

User mit der Rolle Publisher können Projekte bei denen alle erwarteten Metadaten als fertig markiert wurden releasen. Dazu muss in der Navbar (Menü links) auf den Release Button geklickt werden. Bei Release wird eine Postvalidierung durchgeführt, näheres dazu findet sich *hier* <<https://github.com/dzhw/metadatamanagement/wiki/Domain-Model#dataacquisitionproject-post-validation>>. Ab Versionsnummer 1.0.0 wird das Projekt an dalra weitergegeben und erhält eine doi. Wenn der Release Button ein weiteres Mal geklickt wird, wird die Freigabe zurückgezogen und Metadaten können weiter editiert werden. Bei erneutem Klick kann das Projekt dann neu released werden. Gegebenfalls wird eine neue Versionsnummer vergeben (Versionierungskonzept folgt).

## 3.27 Domänenmodell

Im *Domänenmodell* werden alle Domänenobjekte, ihre Relationen zueinander und, ob diese verpflichtend auszufüllen sind, dokumentiert. ..\_Domänenmodell: <https://github.com/dzhw/metadatamanagement/wiki/Domain-Model>

## 3.28 Javadoc

### 3.28.1 eu.dzhw.fdz.metadatamanagement.common.domain

Common domain objects which can be used in all other subdomains of this application.

#### AbstractRdcDomainObject

public abstract class **AbstractRdcDomainObject**

Base class for all rdc domain objects. All domain objects inherit the fields from this base class.

#### Fields

##### createdBy

private **String** **createdBy**

The name of the user which has created this object.

##### createdDate

private **LocalDateTime** **createdDate**

The date and time (in UTC) when this domain object was created.

##### lastModifiedBy

private **String** **lastModifiedBy**

The name of the user who last saved this object.

##### lastModifiedDate

private **LocalDateTime** **lastModifiedDate**

The date and time when this object was last saved.

##### version

private **Long** **version**

Number which is incremented on each save of this object.

#### Counter

public class **Counter**

Counter document which can be used to get an incremented sequence number per document id.

## Fields

### id

private `String id`  
The id of the counter, e.g. „orders“.

### seq

private long `seq`  
The current sequence number.

## I18nString

public class `I18nString`  
Strings that can be represented in English and German.

## Fields

### de

private `String de`  
The german version of this string.

### en

private `String en`  
The english version of this string.

## Period

public class `Period`  
Objects representing periods in time. All periods must have a start date and an end date and the start date must be before or equal to the end date.

## Fields

### end

private `LocalDate end`  
The end date of the period. Mandatory and must not be before start date.

### start

private `LocalDate start`  
The start date of the period. Mandatory and must not be after end date.

## Person

public class **Person**  
 A representation of a person.

### Fields

#### firstName

private *String* **firstName**  
 The first name of the person. Must not be empty.

#### lastName

private *String* **lastName**  
 The last name of the person. Must not be empty.

#### middleName

private *String* **middleName**  
 The middle name of the person.

## Resolution

public class **Resolution**  
 Representation of the resolution of images.

### Fields

#### heightY

private *Integer* **heightY**  
 The height in pixel.

#### widthX

private *Integer* **widthX**  
 The width in pixel.

## Task

public class **Task** extends *AbstractRdcDomainObject*  
 Task entity holding the current state of a long running task.

**Author** tgehrke

## Fields

### errorList

private ErrorListDto **errorList**  
The list of errors which occurred during execution of the task.

### id

private String **id**  
The id or task number of the task.

### location

private String **location**  
The location URI of the result of the task.

### state

private *TaskState* **state**  
The current state of the task.

### type

private *TaskType* **type**  
The type of the task.

## Task.TaskState

public enum **TaskState**  
State of tasks.  
**Author** tgehrke

## Enum Constants

### DONE

public static final *Task.TaskState* **DONE**

### FAILURE

public static final *Task.TaskState* **FAILURE**

## RUNNING

public static final *Task.TaskState* **RUNNING**

## Task.TaskType

public enum **TaskType**  
type of tasks.

**Author** tgehrke

## Enum Constants

### DATA\_SET\_REPORT

public static final *Task.TaskType* **DATA\_SET\_REPORT**

## 3.28.2 eu.dzhw.fdz.metadatamanagement.datasetmanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.datasetmanagement.domain.DataSets*.

### DataSet

public class **DataSet** extends *AbstractRdcDomainObject*  
A dataset contains *Variables*. It results from at least one *Survey*.

### Fields

#### annotations

private *II8nString* **annotations**  
Arbitrary additional text for the dataset. Must not contain more than 2048 characters.

#### dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**  
The id of the *DataAcquisitionProject* to which this dataset belongs. The *dataAcquisitionProjectId* must not be empty.

#### description

private *II8nString* **description**  
A short description of the dataset. It must be specified in at least one language and it must not contain more than 2048 characters.

### format

private *String* **format**

The format of the dataset. Must be one of *Format*.

### id

private *String* **id**

The id of the dataset which uniquely identifies the dataset in this application. The id must not be empty and must be of the form `dat-{{dataAcquisitionProjectId}}-ds{{number}}$`. The id must not contain more than 512 characters.

### number

private *Integer* **number**

The number of the dataset. Must not be empty and must be unique within the *DataAcquisitionProject*.

### studyId

private *String* **studyId**

The id of the *OrderedStudy* to which this dataset belongs. Must not be empty.

### subDataSets

private *List<SubDataSet>* **subDataSets**

List of *SubDataSets* (concrete accessible files) within this dataset. Must contain at least one element. There must not be more than one *SubDataSet* per *AccessWays*.

### surveyIds

private *List<String>* **surveyIds**

List of ids of *Surveys* of this *DataAcquisitionProject*. The dataset contains results from these *Surveys*. Must contain at least one element.

### surveyNumbers

private *List<Integer>* **surveyNumbers**

List of numbers of *Surveys* of this *DataAcquisitionProject*. The dataset contains results from these *Surveys*. Must contain at least one element.

### type

private *String* **type**

The type of the dataset. Must be one of *DataSetTypes* and must not be empty.

## DataSetAttachmentMetadata

public class **DataSetAttachmentMetadata** extends *AbstractRdcDomainObject*  
 Metadata which will be stored with each attachment of a *DataSet*.

### Fields

#### dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**  
 The id of the *DataAcquisitionProject* to which the *DataSet* of this attachment belongs. Must not be empty.

#### dataSetId

private *String* **dataSetId**  
 The id of the *DataSet* to which this attachment belongs. Must not be empty.

#### dataSetNumber

private *Integer* **dataSetNumber**  
 The number of the *DataSet* to which this attachment belongs. Must not be empty.

#### description

private *LocalizedString* **description**  
 A description for this attachment. It must be specified in at least one language and it must not contain more than 512 characters.

#### fileName

private *String* **fileName**  
 The filename of the attachment. Must not be empty and must contain only (german) alphanumeric characters and „\_“ and „-“ and „,“.

#### id

private *String* **id**  
 The id of the attachment. Holds the complete path which can be used to download the file.

#### indexInDataSet

private *Integer* **indexInDataSet**  
 The index in the *DataSet* of this attachment. Used for sorting the attachments of this *DataSet*. Must not be empty.

### language

private `String language`

The language of the attachments content. Must not be empty and must be specified as ISO 639 language code.

### title

private `String title`

The title of the attachment in the language of the attachment. Must not be empty and must not contain more than 2048 characters.

### DataSetTypes

public class `DataSetTypes`

All possible types of a `DataSet`.

#### Fields

##### ALL

public static final `Set<I18nString> ALL`

##### EPISODE\_RECORD

public static final `I18nString EPISODE_RECORD`

##### PERSONAL\_RECORD

public static final `I18nString PERSONAL_RECORD`

#### Format

public class `Format`

All possible formats of a `DataSet`.

#### Fields

##### ALL

public static final `Set<I18nString> ALL`

##### LONG

public static final `I18nString LONG`

## WIDE

public static final *String* **WIDE**

## SubDataSet

public class **SubDataSet**

A subdataset is part of a *DataSet* and describes the concrete analyzable file which is accessible by a given access way.

## Fields

### accessWay

private *String* **accessWay**

The access way of this subdataset. Describes how the user will be able to work with the data set. Must not be empty and be one of *AccessWays* but not *AccessWays.NOT\_ACCESSIBLE*.

### citationHint

private *String* **citationHint**

A hint telling how to cite this subdataset in publications. It must be specified in at least one language and it must not contain more than 2048 characters.

### description

private *String* **description**

A description for this subdataset. It must be specified in at least one language and it must not contain more than 512 characters.

### name

private *String* **name**

The filename of the subdataset without extension. Must not be empty and must not contain more than 32 characters.

### numberOfObservations

private *Integer* **numberOfObservations**

The number of rows (observations or episodes) which are present in this subdataset. Must not be empty.

## 3.28.3 eu.dzhw.fdz.metadatamanagement.instrumentmanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.instrumentmanagement.domain.Instruments*.

### Instrument

public class **Instrument** extends *AbstractRdcDomainObject*  
An instrument (e.g. a questionnaire) which was used in at least one *Survey*.

### Fields

#### annotations

private *II8nString* **annotations**  
Arbitrary additional text for this instrument. Must not contain more than 2048 characters.

#### dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**  
The id of the *DataAcquisitionProject* to which this instrument belongs. The `dataAcquisitionProjectId` must not be empty.

#### description

private *II8nString* **description**  
A short description of the instrument. It must be specified in at least one language and it must not contain more than 512 characters.

#### id

private *String* **id**  
The id of the instrument which uniquely identifies the instrument in this application. The id must not be empty and must be of the form `ins-{{dataAcquisitionProjectId}}-ins{{number}}$`. The id must not contain more than 512 characters.

#### number

private *Integer* **number**  
The number of the instrument. Must not be empty and must be unique within the *DataAcquisitionProject*.

#### studyId

private *String* **studyId**  
The id of the *OrderedStudy* to which this instrument belongs. Must not be empty.

#### subtitle

private *II8nString* **subtitle**  
An optional subtitle of the instrument. It must not contain more than 2048 characters.

## surveyIds

private `List<String>` **surveyIds**

List of ids of *Surveys* of this *DataAcquisitionProject*. The instrument has been used in these *Surveys*. Must contain at least one element.

## surveyNumbers

private `List<Integer>` **surveyNumbers**

List of numbers of *Surveys* of this *DataAcquisitionProject*. The instrument has been used in these *Surveys*. Must contain at least one element.

## title

private `II8nString` **title**

The title of the instrument. It must be specified in at least one language and it must not contain more than 2048 characters.

## type

private `String` **type**

The type of this instrument. Must be one of *InstrumentTypes* and must not be empty.

## InstrumentAttachmentMetadata

public class **InstrumentAttachmentMetadata** extends *AbstractRdcDomainObject*

Metadata which will be stored with each attachment of a *Instrument*.

## Fields

### dataAcquisitionProjectId

private `String` **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* to which the *Instrument* of this attachment belongs. Must not be empty.

### description

private `II8nString` **description**

A description for this attachment. It must be specified in at least one language and it must not contain more than 512 characters.

### fileName

private `String` **fileName**

The filename of the attachment. Must not be empty and must contain only (german) alphanumeric characters and „\_“ and „-“ and „,“.

## id

private `String id`

The id of the attachment. Holds the complete path which can be used to download the file.

## indexInInstrument

private `Integer indexInInstrument`

The index in the *Instrument* of this attachment. Used for sorting the attachments of this *Instrument*. Must not be empty.

## instrumentId

private `String instrumentId`

The id of the *Instrument* to which this attachment belongs. Must not be empty.

## instrumentNumber

private `Integer instrumentNumber`

The number of the *Instrument* to which this attachment belongs. Must not be empty.

## language

private `String language`

The language of the attachments content. Must not be empty and must be specified as ISO 639 language code.

## type

private `InstrumentAttachmentTypes type`

The type of this attachment. Must not be empty and must be one of *InstrumentAttachmentTypes*.

## InstrumentAttachmentTypes

public class `InstrumentAttachmentTypes`

All valid types of an instrument attachment.

### Fields

#### ALL

public static final `Set<InstrumentAttachmentTypes> ALL`

#### OTHER

public static final `InstrumentAttachmentTypes OTHER`

## QUESTIONNAIRE

public static final *String* **QUESTIONNAIRE**

## QUESTION\_FLOW

public static final *String* **QUESTION\_FLOW**

## VARIABLE\_QUESTIONNAIRE

public static final *String* **VARIABLE\_QUESTIONNAIRE**

## InstrumentTypes

public class **InstrumentTypes**  
 All valid types of an instrument.

### Fields

#### ALL

public static final *Set*<*String*> **ALL**

#### CAPI

public static final *String* **CAPI**

#### CATI

public static final *String* **CATI**

#### CAWI

public static final *String* **CAWI**

#### PAPI

public static final *String* **PAPI**

### 3.28.4 eu.dzhw.fdz.metadatamanagement.ordermanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.ordermanagement.domain.Orders*.

### Customer

public class **Customer**

Details of a customer who has ordered *Products*.

### Fields

#### email

private *String* **email**

Email address of the customer. Must be a valid email address and must not be empty.

#### name

private *String* **name**

Name of the customer as given in the shopping cart. Must not be empty.

### Order

public class **Order** extends *AbstractRdcDomainObject*

Order (DTO) containing all relevant information of a *Customer* and her *Products*.

### Fields

#### client

private *OrderClient* **client**

The id of the client (one of @link *OrderClient*) who has last modified this order.

#### customer

private *Customer* **customer**

The *Customer* who has placed this order. Must not be null.

#### id

private *String* **id**

The id of an order. It is a number which is generated by a sequence (see *Counter*).

#### languageKey

private *String* **languageKey**

The key of the preferred language (either „de“ or „en“) of the *Customer*. Must not be empty.

## products

private `List<Product>` **products**  
 List of data *Products* the *Customer* want to order.

## state

private `OrderState` **state**  
 The current state of the order. One of *OrderState*.

## OrderAlreadyCompletedException

public class **OrderAlreadyCompletedException** extends `IllegalArgumentException`  
 Orders with *OrderState.ORDERED* must not be updated. This exception should be thrown whenever an update attempt is made on such orders.

## OrderClient

public enum **OrderClient**  
 Enum holding possible clients with write access to the orders.

## Enum Constants

### DLP

public static final `OrderClient` **DLP**

### MDM

public static final `OrderClient` **MDM**

## OrderState

public enum **OrderState**  
 The states an *Order* can have.

**Author** René Reitmann

## Enum Constants

### CREATED

public static final `OrderState` **CREATED**

### NOTIFIED

public static final `OrderState` **NOTIFIED**

## ORDERED

public static final *OrderState* ORDERED

## OrderedStudy

public class **OrderedStudy**

Partial *eu.dzhw.fdz.metadatamanagement.studymanagement.domain.Study* which is part of a *Product*. It is a copy of the *eu.dzhw.fdz.metadatamanagement.studymanagement.domain.Study* attributes which is made when the *Customer* places the orders.

## Fields

### annotations

private *II&String* **annotations**

The annotations of the *eu.dzhw.fdz.metadatamanagement.studymanagement.domain.Study*.

### id

private *String* **id**

The id of the *eu.dzhw.fdz.metadatamanagement.studymanagement.domain.Study*. Must not be empty.

### title

private *II&String* **title**

The title of the *eu.dzhw.fdz.metadatamanagement.studymanagement.domain.Study*. Must not be empty neither in German nor in English.

## Product

public class **Product**

Data Product which can be ordered by a customer.

**Author** René Reitmann

## Fields

### accessWay

private *String* **accessWay**

The access way to the *DataSets* which the *Customer* wants to have.

### dataAcquisitionProjectId

private `String` **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* in which this product was generated. Must not be empty.

### study

private *OrderedStudy* **study**

The (partial) *OrderedStudy* of this product. Must not be empty.

### version

private `String` **version**

The version of the *DataSets* which the *Customer* wants to have.

## 3.28.5 eu.dzhw.fdz.metadatamanagement.projectmanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.projectmanagement.domain.DataAcquisitionProjects*.

### AssigneeGroup

public enum **AssigneeGroup**

Possible assignee groups.

### Enum Constants

#### DATA\_PROVIDER

public static final *AssigneeGroup* **DATA\_PROVIDER**

#### PUBLISHER

public static final *AssigneeGroup* **PUBLISHER**

### Configuration

public class **Configuration**

The project configuration describes which users are publishers or data providers for a project.

### Fields

#### dataProviders

private `List<String>` **dataProviders**

User names having the role of a data provider for a project. Must contain at least one user name.

### **dataSetsState**

private *ProjectState* **dataSetsState**  
The state of data sets.

### **instrumentsState**

private *ProjectState* **instrumentsState**  
The state of instruments.

### **publishers**

private *List<String>* **publishers**  
User names having the role of a publisher for a project. Must contain at least one user name.

### **questionsState**

private *ProjectState* **questionsState**  
The state of questions.

### **requirements**

private *Requirements* **requirements**  
Defines which object types are required before a project can be released.

### **studiesState**

private *ProjectState* **studiesState**  
The state of the study.

### **surveysState**

private *ProjectState* **surveysState**  
The State of surveys.

### **variablesState**

private *ProjectState* **variablesState**  
The state of variables.

### **DaraUpdateQueueItem**

public class **DaraUpdateQueueItem** extends *AbstractRdcDomainObject*  
Publishing metadata to *dalra* will be done asynchronously and repeated as long as the update queue item has not been processed successfully and has therefore been deleted.

## Fields

### id

private `String id`  
 The id of the update queue item. It is generated by the database.

### projectId

private `String projectId`  
 The id of the `DataAcquisitionProject` which needs to be sent to dalra. Must not be empty and there must be at most one update queue item in the database for any project.

### updateStartedAt

private `LocalDateTime updateStartedAt`  
 Timestamp at which the update has been started.

### updateStartedBy

private `String updateStartedBy`  
 Id of the process who started the updated. Stored in order to avoid having multiple concurrent processes sending data to dalra.

## DataAcquisitionProject

public class `DataAcquisitionProject` extends `AbstractRdcDomainObject`  
 The data acquisition project collects the metadata for the data products which are published by our RDC. One project can contain one `Study`, many `Surveys`, many `Instruments` and `Questions`, and many `DataSets` and `Variables`. A project can be currently released (visible to public users) or not. When a publisher releases a project and its version is greater than or equal to 1.0.0 then the metadata is published to dalra.

## Fields

### assigneeGroup

private `AssigneeGroup assigneeGroup`  
 Determines which assignee group is able to edit data on the project.

### configuration

private `Configuration configuration`  
 Contains the project configuration.

### hasBeenReleasedBefore

private **Boolean** **hasBeenReleasedBefore**

Flag indicating whether this project has ever been released in its life. It is used to ensure that project cannot be deleted once they have been released.

### id

private **String** **id**

The id of this project. Must not be empty and must only contain lower cased (english) letters and numbers. Must not contain more than 32 characters.

### lastAssigneeGroupMessage

private **String** **lastAssigneeGroupMessage**

The last message provided by an assignee group user before *DataAcquisitionProject.assigneeGroup* value changed.

### release

private **Release** **release**

A valid *Release* object. Null if the project is currently not released. The version of a *Release* must be a syntactically correct according to semver (major.minor.patch) and must not be decreased.

## ProjectState

public class **ProjectState**

State of a data acquisition project. Used for all metadata

**Author** tgehrke

## Fields

### isDataProviderReady

private boolean **isDataProviderReady**

indicates if the data providers marked it's metadata as ready.

### isPublisherReady

private boolean **isPublisherReady**

indicates if the publisher marked the metadata as ready.

## Release

public class **Release**

The release object contains the version and a timestamp of the current release.

## Fields

### date

private `LocalDateTime` **date**

The timestamp (in UTC) indicates when a publisher has released the *DataAcquisitionProject*. Must not be empty.

### version

private `String` **version**

A valid semver version (major.minor.patch). Must not be empty and must not contain more than 32 characters. A version of a *DataAcquisitionProject* must not be decreased.

## Requirements

public class **Requirements**

This configuration defines which object types have to be delivered before a project can be released.

## Fields

### isDataSetsRequired

private boolean **isDataSetsRequired**

Defines if data set data is required for a release.

### isInstrumentsRequired

private boolean **isInstrumentsRequired**

Defines if instrument data is required for a release.

### isQuestionsRequired

private boolean **isQuestionsRequired**

Defines if question data is required for a release.

### isStudiesRequired

private boolean **isStudiesRequired**

Defines if study data is required for a release (this object type is mandatory and this setting is therefore always `true`).

### isSurveysRequired

private boolean **isSurveysRequired**

Defines if survey data is required for a release.

## isVariablesRequired

private boolean **isVariablesRequired**  
Defines if variable data is required for a release.

## 3.28.6 eu.dzhw.fdz.metadatamanagement.questionmanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.questionmanagement.domain*.  
*Questions*.

## ImageType

public enum **ImageType**  
Enum representing supported types of question images.

## Enum Constants

### PNG

public static final *ImageType* **PNG**

## Question

public class **Question** extends *AbstractRdcDomainObject*  
A question is part of an *Instrument* which has been used in at least one *Surveys*. The responses to a question are stored in *Variables*.

## Fields

### additionalQuestionText

private *II8nString* **additionalQuestionText**  
Arbitrary additional question text which has been presented to the participant. Must not contain more than 1 MB characters.

### annotations

private *II8nString* **annotations**  
Arbitrary annotations to this question. Must not contain more than 2048 characters.

### dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**  
The id of the *DataAcquisitionProject* to which this question belongs. The *dataAcquisitionProjectId* must not be empty.

## id

private `String` **id**

The id of the question which uniquely identifies the question in this application. The id must not be empty and must be of the form `que-{{dataAcquisitionProjectId}}-ins{{instrumentNumber}}-{{number}}$`. The id must not contain more than 512 characters.

## indexInInstrument

private `Integer` **indexInInstrument**

The index of the question in the *Instrument*. Used for sorting the questions.

## instruction

private `String` **instruction**

The instruction for the participant which tells how to give the answers to this question. Must not contain more than 1 MB characters.

## instrumentId

private `String` **instrumentId**

The id of the *Instrument* to which this question belongs. Must not be empty.

## instrumentNumber

private `Integer` **instrumentNumber**

The number of the *Instrument* to which this question belongs. Must not be empty.

## introduction

private `String` **introduction**

The introduction of this question which gives more context to the participant before asking the question. Must not contain more than 2048 characters.

## number

private `String` **number**

The number of the question. Must not be empty and must be unique within the *Instrument*. Must contain only (german) alphanumeric characters and „\_“,“-“ and „,“ and must not contain more than 32 characters.

## questionText

private `String` **questionText**

The question the *Surveys* participant was asked. It must be specified in at least one language and it must not contain more than 2048 characters.

## studyId

private `String` **studyId**

The id of the *OrderedStudy* to which this question belongs. Must not be empty.

## successorNumbers

private `List<String>` **successorNumbers**

List of numbers of the *Questions* which directly follow this question in the *Instrument*.

## successors

private `List<String>` **successors**

List of ids of the *Questions* which directly follow this question in the *Instrument*.

## technicalRepresentation

private *TechnicalRepresentation* **technicalRepresentation**

A *TechnicalRepresentation* of this question. This is optional and can be used to add the source code of the question which was used to generate it.

## topic

private *II8nString* **topic**

The topic or section in the *Instrument* to which this question belongs. It must not contain more than 2048 characters.

## type

private *II8nString* **type**

The type of the question. Must be one of `QuestionTypes` and must not be empty.

## QuestionImageMetadata

public class **QuestionImageMetadata** extends *AbstractRdcDomainObject*

The metadata for one question images. One question image displays the question in one language with one given resolution.

## Fields

### containsAnnotations

private `Boolean` **containsAnnotations**

Flag indicating whether the image contains annotations which highlight parts that were only visible to specific participants. These annotations were not visible to the participants.

### dataAcquisitionProjectId

private `String` **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* of the *Question* to which this image belongs. Must not be empty.

### fileName

private `String` **fileName**

The name of the images file. Must not be empty and must only contain (german) alphanumeric characters and „\_“,“-“ and „.“.

### imageType

private `ImageType` **imageType**

The type of this image. Must be one of *ImageType* and must not be empty.

### indexInQuestion

private `Integer` **indexInQuestion**

The index in the *Question* of this image. Used for sorting the images of this *Question*. Must not be empty.

### language

private `String` **language**

The language of the question text on this image. Must not be empty and must be a valid ISO 639 code.

### questionId

private `String` **questionId**

The id of the *Question* to which this image belongs. Must not be empty.

### resolution

private `Resolution` **resolution**

The resolution of the image.

### QuestionTypes

public class **QuestionTypes**

All valid types of a *Question*.

## Fields

### ALL

public static final Set<I18nString> **ALL**

### GRID

public static final I18nString **GRID**

### ITEM\_SET

public static final I18nString **ITEM\_SET**

### MULTIPLE\_CHOICE

public static final I18nString **MULTIPLE\_CHOICE**

### OPEN

public static final I18nString **OPEN**

### SINGLE\_CHOICE

public static final I18nString **SINGLE\_CHOICE**

### UNDOCUMENTED

public static final I18nString **UNDOCUMENTED**

## TechnicalRepresentation

public class **TechnicalRepresentation**

The technical representation of a *Question* which was used to generate the question for instance in an online *Instrument*.

## Fields

### language

private String **language**

The technical language of the source of this representation. E.g. „qml“. Must not be empty and must not contain more than 32 characters.

### source

private `String` **source**

The source code of the question. Must not be empty and must not contain more than 1 MB characters.

### type

private `String` **type**

The type of the technical representation. E.g. „zofar“. Must not be empty and must not contain more than 32 characters.

## 3.28.7 eu.dzhw.fdz.metadatamanagement.relatedpublicationmanagement.domain

In the domain layer are all domain classes of the related publication.

**author** Daniel Katzberg

### RelatedPublication

public class **RelatedPublication** extends *AbstractRdcDomainObject*  
Domain Object for the Related Publications.

**Author** Daniel Katzberg

### Fields

#### abstractSource

private *118nString* **abstractSource**

#### authors

private `String` **authors**

#### dataSetIds

private `List<String>` **dataSetIds**

#### doi

private `String` **doi**

#### id

private `String` **id**

### **instrumentIds**

private List<String> **instrumentIds**

### **language**

private String **language**

### **publicationAbstract**

private String **publicationAbstract**

### **questionIds**

private List<String> **questionIds**

### **sourceLink**

private String **sourceLink**

### **sourceReference**

private String **sourceReference**

### **studyIds**

private List<String> **studyIds**

### **studySerieses**

private List<I18nString> **studySerieses**

### **surveyIds**

private List<String> **surveyIds**

### **title**

private String **title**

### **variableIds**

private List<String> **variableIds**

**year**

private Integer **year**

### 3.28.8 eu.dzhw.fdz.metadatamanagement.studymanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.studymanagement.domain.Studys*.

#### DataAvailabilities

public class **DataAvailabilities**

The data's availability of a *Study* can be in one of these states.

#### Fields

##### ALL

public static final Set<I18nString> **ALL**

##### AVAILABLE

public static final I18nString **AVAILABLE**

##### IN\_PREPARATION

public static final I18nString **IN\_PREPARATION**

##### NOT\_AVAILABLE

public static final I18nString **NOT\_AVAILABLE**

#### Study

public class **Study** extends *AbstractRdcDomainObject* implements StudySubDocumentProjection

A study contains all metadata of a *DataAcquisitionProject*. It will get a DOI (Digital Object Identifier) when the *DataAcquisitionProject* is released.

#### Fields

##### annotations

private I18nString **annotations**

Arbitrary additional text for this instrument. Must not contain more than 2048 characters.

## authors

private *List*<*Person*> **authors**

List of *Persons* which have performed this study. Must not be empty.

## dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* to which this study belongs. The *dataAcquisitionProjectId* must not be empty.

## dataAvailability

private *IN8nString* **dataAvailability**

The current state of the data's availability. Must be one of *DataAvailabilities* and must not be empty.

## description

private *IN8nString* **description**

A description of the study. It must be specified in at least one language and it must not contain more than 2048 characters.

## id

private *String* **id**

The id of the study which uniquely identifies the study in this application. The id must not be empty and must be of the form `stu-{{dataAcquisitionProjectId}}$`. The id must not contain more than 512 characters.

## institution

private *IN8nString* **institution**

The name of the institution which has performed this study. It must be specified in German and English and it must not contain more than 512 characters.

## sponsor

private *IN8nString* **sponsor**

The name of the sponsor who which has sponsored this study. It must be specified in German and English and it must not contain more than 512 characters.

## studySeries

private *IN8nString* **studySeries**

The name of the series of studies to which this study belongs.. If specified it must be specified in German and English. It must not contain more than 512 characters and must not contain „,“.

## surveyDesign

private *II8nString* **surveyDesign**

The survey design of this *Study*. Must be one of *SurveyDesigns* and must not be empty.

## title

private *II8nString* **title**

The title of the study. It must be specified in German and English and it must not contain more than 2048 characters.

## StudyAttachmentMetadata

public class **StudyAttachmentMetadata** extends *AbstractRdcDomainObject*

Metadata which will be stored with each attachment of a *Study*.

## Fields

### dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* to which the *Study* of this attachment belongs. Must not be empty.

### description

private *II8nString* **description**

A description for this attachment. It must be specified in at least one language and it must not contain more than 512 characters.

### fileName

private *String* **fileName**

The filename of the attachment. Must not be empty and must contain only (german) alphanumeric characters and „\_“ and „-“ and „.“.

### id

private *String* **id**

The id of the attachment. Holds the complete path which can be used to download the file.

### indexInStudy

private *Integer* **indexInStudy**

The index in the *Study* of this attachment. Used for sorting the attachments of this *Study*. Must not be empty.

## language

private *String* **language**

The language of the attachments content. Must not be empty and must be specified as ISO 639 language code.

## studyId

private *String* **studyId**

The id of the *Study* to which this attachment belongs. Must not be empty.

## title

private *String* **title**

An optional title of this attachment in the attachments' language. It must not contain more than 2048 characters.

## type

private *II8nString* **type**

The type of the attachment. Must be one of *StudyAttachmentTypes* and must not be empty.

## StudyAttachmentTypes

public class **StudyAttachmentTypes**

All valid types of a *StudyAttachmentMetadata*.

### Fields

#### ALL

public static final *Set*<*II8nString*> **ALL**

#### METHOD\_REPORT

public static final *II8nString* **METHOD\_REPORT**

#### OTHER

public static final *II8nString* **OTHER**

## SurveyDataTypes

public class **SurveyDataTypes** extends *DataTypes*

List of types of data, which a *Study* can consist of. It will be computed from the *Surveys* of a *Study*.

## Fields

### ALL

public static final Set<I18nString> **ALL**

### MIXED\_METHODS

public static final I18nString **MIXED\_METHODS**

## SurveyDesigns

public class **SurveyDesigns**  
 List of currently supported survey designs.

## Fields

### ALL

public static final Set<I18nString> **ALL**

### CROSS\_SECTION

public static final I18nString **CROSS\_SECTION**

### PANEL

public static final I18nString **PANEL**

## 3.28.9 eu.dzhw.fdz.metadatamanagement.surveymangement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.surveymangement.domain.Surveys*.

## DataTypes

public class **DataTypes**  
 Types of data, which a *Survey* can produce.

## Fields

### ALL

public static final Set<I18nString> **ALL**

## QUALITATIVE\_DATA

public static final *II8nString* **QUALITATIVE\_DATA**

## QUANTITATIVE\_DATA

public static final *II8nString* **QUANTITATIVE\_DATA**

## Population

public class **Population**  
Details of the population of a *Survey*.

### Fields

#### description

private *II8nString* **description**  
A description of the population. It must be specified in at least one language and it must not contain more than 2048 characters.

#### title

private *II8nString* **title**  
A short title for the population. It must be specified in at least one language and it must not contain more than 512 characters.

## Survey

public class **Survey** extends *AbstractRdcDomainObject*  
A survey is conducted to examine a population on the basis of a sample. The resulting *DataSets* can be used to make statements about the population.

### Fields

#### annotations

private *II8nString* **annotations**  
Arbitrary additional text for this survey. Must not contain more than 2048 characters.

#### dataAcquisitionProjectId

private *String* **dataAcquisitionProjectId**  
The id of the *DataAcquisitionProject* to which this survey belongs. The *dataAcquisitionProjectId* must not be empty.

## dataType

private *String* **dataType**

The type of data which the survey produced. Must be one of *DataTypes* and must not be empty.

## fieldPeriod

private *Period* **fieldPeriod**

The period during which the survey has been conducted or is expected to be conducted. Must not be empty.

## grossSampleSize

private *Integer* **grossSampleSize**

The gross sample size represents the number of participants which have been invited to take part in the *Survey*. Must not be negative.

## id

private *String* **id**

The id of the survey which uniquely identifies the survey in this application. The id must not be empty and must be of the form `sur-{{dataAcquisitionProjectId}}-sy{{number}}$`. The id must not contain more than 512 characters.

## number

private *Integer* **number**

The number of the instrument. Must not be empty and must be unique within the *DataAcquisitionProject*.

## population

private *Population* **population**

Details about the *Population*. Must not be empty.

## responseRate

private *Double* **responseRate**

The response rate is the quotient of the gross sample size and the sample size. Must be between 0 and 100.

## sample

private *String* **sample**

The sampling method is the procedure for selecting sample members from a population. It must be specified in at least one language and it must not contain more than 2048 characters.

## sampleSize

private Integer **sampleSize**

The sample size is the number of participant which took part in the survey. Must not be empty and must not be negative.

## studyId

private String **studyId**

The id of the *Study* to which this survey belongs. Must not be empty.

## surveyMethod

private *II8nString* **surveyMethod**

The survey method briefly describes how the data were collected. It must be specified in German and English and it must not contain more than 2048 characters.

## title

private *II8nString* **title**

The title of the instrument. It must be specified in German and English and it must not contain more than 2048 characters.

## wave

private Integer **wave**

Number of the wave which this *Survey* represents. Will be ignored if the *Study* is not organized in waves. Must not be empty and must be greater than or equal to 1.

## SurveyAttachmentMetadata

public class **SurveyAttachmentMetadata** extends *AbstractRdcDomainObject*

Metadata which will be stored with each attachment of a *Survey*.

## Fields

### dataAcquisitionProjectId

private String **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* to which the *Survey* of this attachment belongs. Must not be empty.

### description

private *II8nString* **description**

A description for this attachment. It must be specified in at least one language and it must not contain more than 512 characters.

### fileName

private `String fileName`

The filename of the attachment. Must not be empty and must contain only (german) alphanumeric characters and „\_“ and „-“ and „,“.

### id

private `String id`

The id of the attachment. Holds the complete path which can be used to download the file.

### indexInSurvey

private `Integer indexInSurvey`

The index in the *Survey* of this attachment. Used for sorting the attachments of this *Survey*. Must not be empty.

### language

private `String language`

The language of the attachments content. Must not be empty and must be specified as ISO 639 language code.

### surveyId

private `String surveyId`

The id of the *Survey* to which this attachment belongs. Must not be empty.

### surveyNumber

private `Integer surveyNumber`

The number of the *Survey* to which this attachment belongs. Must not be empty.

### title

private `String title`

A title of this attachment in the attachments' language. It must not contain more than 2048 characters.

### SurveyResponseRateImageMetadata

public class `SurveyResponseRateImageMetadata` extends *AbstractRdcDomainObject*

Metadata which will be stored with each response rate image of a *Survey*.

## Fields

### dataAcquisitionProjectId

private `String` **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* to which the *Survey* of this response rate image belongs. Must not be empty.

### fileName

private `String` **fileName**

The filename of the image. Must not be empty and must contain only (german) alphanumeric characters and „\_“ and „-“ and „.“.

### id

private `String` **id**

The id of the response rate image. Holds the complete path which can be used to download the file.

### language

private `String` **language**

The language used in the response rate image. Must be either „de“ or „en“.

### surveyId

private `String` **surveyId**

The id of the *Survey* to which this response rate image belongs. Must not be empty.

### surveyNumber

private `Integer` **surveyNumber**

The number of the *Survey* to which this response rate image belongs. Must not be empty.

## 3.28.10 eu.dzhw.fdz.metadatamanagement.variablemanagement.domain

Domain objects describing *eu.dzhw.fdz.metadatamanagement.variablemanagement.domain.Variables*.

### AccessWays

public class **AccessWays**

An access way of a *Variable* or a *DataSet* indicates how the data user will be able to work with the data.

## Fields

### ALL

public static final Set<String> **ALL**

### DOWNLOAD\_CUF

public static final String **DOWNLOAD\_CUF**

### DOWNLOAD\_SUF

public static final String **DOWNLOAD\_SUF**

### NOT\_ACCESSIBLE

public static final String **NOT\_ACCESSIBLE**

### ONSITE\_SUF

public static final String **ONSITE\_SUF**

### REMOTE\_DESKTOP

public static final String **REMOTE\_DESKTOP**

## DataTypes

public class **DataTypes**

The technical type which the *ValidResponses* have.

## Fields

### ALL

public static final Set<I18nString> **ALL**

### DATE

public static final I18nString **DATE**

### NUMERIC

public static final I18nString **NUMERIC**

## STRING

public static final *String* **STRING**

## Distribution

public class **Distribution**

A distribution contains the descriptives of a *Variable* meaning its *ValidResponses*, *Missings* and *Statistics*.

## Fields

### maxNumberOfDecimalPlaces

private *Integer* **maxNumberOfDecimalPlaces**

Integer used for rounding the values of this *Variable* when displaying it. It is computed during the import of the *Variable* by finding the maximum number of decimal places in the list of *ValidResponses*.

### missings

private *List*<*Missing*> **missings**

List of *Missings* of this *Variable*. Must not contain more than 7000 entries and the code of the *Missings* must be unique.

### statistics

private *Statistics* **statistics**

Descriptive metrics of this *Variable*.

### totalAbsoluteFrequency

private *Integer* **totalAbsoluteFrequency**

The total absolute number of *ValidResponses* and *Missings*. Must not be empty.

### totalValidAbsoluteFrequency

private *Integer* **totalValidAbsoluteFrequency**

The total absolute number of *ValidResponses*. Must not be empty.

### totalValidRelativeFrequency

private *Double* **totalValidRelativeFrequency**

The quotient from totalValidAbsoluteFrequency and totalAbsoluteFrequency. Must not be empty.

## validResponses

private *List*<*ValidResponse*> **validResponses**

List of *ValidResponses* of this variable. Must not contain more than 7000 entries and the value of the *ValidResponses* must be unique.

## FilterDetails

public class **FilterDetails**

Filter details of a *Variable* describe the condition which must have evaluated to true before a participant was asked a *Question* resulting in this *Variable*. All participants for which the conditions evaluates to false will have a *Missing* in this *Variable*.

## Fields

### description

private *String* **description**

A description of this filter condition. Must not contain more than 2048 characters

### expression

private *String* **expression**

A technical expression describing the condition which must have evaluated to true. The expression is given in the *expressionLanguage*. Must not be empty and must not contain more than 2048 characters.

### expressionLanguage

private *String* **expressionLanguage**

The name of the language in which the expression was given. Must not be empty and must be one of *FilterExpressionLanguages*.

## FilterExpressionLanguages

public class **FilterExpressionLanguages**

All supported expression languages for *FilterDetails*.

## Fields

### ALL

public static final *Set*<*String*> **ALL**

### SPEL

public static final *String* **SPEL**

## STATA

public static final String **STATA**

## GenerationDetails

public class **GenerationDetails**

Generation details describe how a *Variable* was generated from one or more input *Variables*.

## Fields

### description

private *II8nString* **description**

A description of this generation rule. Must not contain more than 2048 characters

### rule

private String **rule**

The computation rule in the ruleExpressionLanguage which was used to generate this *Variable*. Must not contain more than 1 MB characters.

### ruleExpressionLanguage

private String **ruleExpressionLanguage**

The language which was used to describe this rule. Must be one of *RuleExpressionLanguages*.

## Missing

public class **Missing**

A missing or missing value is a value in a *Variable* which represents a reason why no observation (*ValidResponse*) has been stored. It also contains its frequency.

## Fields

### absoluteFrequency

private Integer **absoluteFrequency**

The absolute number of occurrences of this missing. Must not be empty.

### code

private String **code**

A (unique in this *Variable*) code for this missing. Must not be empty.

## label

private *NSString* **label**

A label describing this missing. Must not contain more than 512 characters.

## relativeFrequency

private *Double* **relativeFrequency**

The quotient from absoluteFrequency and *Distribution*.totalAbsoluteFrequency. Must not be empty.

## RelatedQuestion

public class **RelatedQuestion**

A related question is a *Question* which has been asked to generate the values of a *Variable*. It contains the ids of the *Instrument* and the *Question* as well as all Strings of the *Question* which are related to this *Variable*.

## Fields

### instrumentId

private *String* **instrumentId**

The id of the *Instrument* of this *Question*. Must not be empty.

### instrumentNumber

private *String* **instrumentNumber**

The number of the *Instrument* of this *Question*. Must not be empty.

### questionId

private *String* **questionId**

The id of the corresponding *Question*. Must not be empty.

### questionNumber

private *String* **questionNumber**

The number of the corresponding *Question*. Must not be empty.

### relatedQuestionStrings

private *NSString* **relatedQuestionStrings**

All Strings (concatenated) of this *Question* which „belong“ to this *Variable*. These Strings typically overlap with String from other *Variables* of the same *Question*.

## RuleExpressionLanguages

public class **RuleExpressionLanguages**  
All supported expression languages for *GenerationDetails*.

### Fields

#### ALL

public static final Set<String> **ALL**

#### R

public static final String **R**

#### STATA

public static final String **STATA**

## ScaleLevels

public class **ScaleLevels**  
The scale level (or level of measurement) classifies the nature of information within the values assigned to a *Variable* (*ValidResponses*). It determines which mathematical operations can be performed with the values.

### Fields

#### ALL

public static final Set<I18nString> **ALL**

#### INTERVAL

public static final I18nString **INTERVAL**

#### NOMINAL

public static final I18nString **NOMINAL**

#### ORDINAL

public static final I18nString **ORDINAL**

## RATIO

public static final *String* **RATIO**

## Statistics

public class **Statistics**  
 Descriptive metrics of this *Variable*.

## Fields

### deviance

private *Double* **deviance**  
 See *Deviance* (Wikipedia).

### firstQuartile

private *String* **firstQuartile**  
 Splits off the lowest 25% of the values (*ValidResponses*) of this *Variable* from the highest 75%. Must not contain more than 32 characters.

### highWhisker

private *Double* **highWhisker**  
 The highest value still within 1.5 IQR of the third quartile.

### kurtosis

private *Double* **kurtosis**  
 See *Kurtosis* (Wikipedia).

### lowWhisker

private *Double* **lowWhisker**  
 The lowest value still within 1.5 IQR of the first quartile.

### maximum

private *String* **maximum**  
 The maximum of the values (*ValidResponses*) of this *Variable*. Must not contain more than 32 characters.

## meanDeviation

private `Double meanDeviation`  
See Mean Absolute Deviation (Wikipedia).

## meanValue

private `Double meanValue`  
The arithmetic mean of the values (*ValidResponses*) of this *Variable*.

## median

private `String median`  
The median is the value separating the higher half from the lower half of the values (*ValidResponses*) of this *Variable*. Must not contain more than 32 characters.

## minimum

private `String minimum`  
The minimum of the values (*ValidResponses*) of this *Variable*. Must not contain more than 32 characters.

## mode

private `String mode`  
The mode is the value (*ValidResponse*) that appears most often.

## skewness

private `Double skewness`  
See Skewness (Wikipedia).

## standardDeviation

private `Double standardDeviation`  
Measure that is used to quantify the amount of variation of the values (*ValidResponses*) of this *Variable*.

## thirdQuartile

private `String thirdQuartile`  
Splits off the highest 25% of the values (*ValidResponses*) of this *Variable* from the lowest 75%. Must not contain more than 32 characters.

## StorageTypes

public class `StorageTypes`  
All supported storage types of *Variables*.

## Fields

### ALL

public static final Set<String> **ALL**

### ANY

public static final String **ANY**

### BUILTIN

public static final String **BUILTIN**

### BYTECODE

public static final String **BYTECODE**

### CHAR

public static final String **CHAR**

### CHARACTER

public static final String **CHARACTER**

### CLOSURE

public static final String **CLOSURE**

### COMPLEX

public static final String **COMPLEX**

### DOTDOTDOT

public static final String **DOTDOTDOT**

### DOUBLE

public static final String **DOUBLE**

## ENVIRONMENT

public static final String **ENVIRONMENT**

## EXPRESSION

public static final String **EXPRESSION**

## EXTERNALPTR

public static final String **EXTERNALPTR**

## INTEGER

public static final String **INTEGER**

## LANGUAGE

public static final String **LANGUAGE**

## LIST

public static final String **LIST**

## LOGICAL

public static final String **LOGICAL**

## NULL

public static final String **NULL**

## PAIRLIST

public static final String **PAIRLIST**

## PROMISE

public static final String **PROMISE**

## RAW

public static final String **RAW**

## S4

public static final String **S4**

## SPECIAL

public static final String **SPECIAL**

## SYMBOL

public static final String **SYMBOL**

## WEAKREF

public static final String **WEAKREF**

## ValidResponse

public class **ValidResponse**

A valid response represents one observation of a *Variable* and its frequency.

### Fields

#### absoluteFrequency

private Integer **absoluteFrequency**

The absolute number of occurrences of this observation. Must not be empty.

#### label

private *OrNullString* **label**

An optional label for the value of this observation.

#### relativeFrequency

private Double **relativeFrequency**

The quotient from absoluteFrequency and *Distribution*.totalAbsoluteFrequency. Must not be empty.

#### validRelativeFrequency

private Double **validRelativeFrequency**

The quotient from absoluteFrequency and *Distribution*.totalValidAbsoluteFrequency. Must not be empty.

### value

private `String` **value**

The value which has been observed (e.g. was responded by the participant). Must not be empty and must not contain more than 256 characters.

### Variable

public class **Variable** extends *AbstractRdcDomainObject*

A variable contains the results from at least one *Survey*. These results can be the responses from participants of an online survey, hence a variable can result from *RelatedQuestions*. A variable is part of exactly one *DataSet*.

### Fields

#### accessWays

private `List<String>` **accessWays**

The access way of this variable. Depends on the sensitivity of the data and describes how the data user will be able to work with the data. Must not be empty and be one of *AccessWays*.

#### annotations

private *II8nString* **annotations**

Arbitrary additional text for this variable. Must not contain more than 2048 characters.

#### dataAcquisitionProjectId

private `String` **dataAcquisitionProjectId**

The id of the *DataAcquisitionProject* to which this variable belongs. The `dataAcquisitionProjectId` must not be empty.

#### dataSetId

private `String` **dataSetId**

The id of the *DataSet* to which this variable belongs. Must not be empty.

#### dataSetNumber

private `Integer` **dataSetNumber**

The number of the *DataSet* to which this variable belongs. Must not be empty.

#### dataType

private *II8nString* **dataType**

The technical type which the *ValidResponses* have. Must be one of *DataTypes* and must not be empty.

### derivedVariablesIdentifier

private **String** **derivedVariablesIdentifier**

Identifier used to group variables within this *DataSet* which have been derived from each other. For instance one variable might be an aggregated version of the other. Must be of the form `{{dataAcquisitionProjectId}}-ds{{dataSetNumber}}-{{string}}$`. Must not contain more than 512 characters and must contain only (german) alphanumeric characters and „\_“ and „-“.

### distribution

private *Distribution* **distribution**

The *Distribution* contains the descriptives of this variable meaning *ValidResponses*, *Missings* and *Statistics*.

### doNotDisplayThousandsSeparator

private **Boolean** **doNotDisplayThousandsSeparator**

Flag indicating whether the *ValidResponses* should be displayed with a thousands separator or not. For instance years (1970) are numeric but should not be displayed with a thousands separator. Default value is false indicating that the *ValidResponses* are displayed with thousands separator.

### filterDetails

private *FilterDetails* **filterDetails**

*FilterDetails* of a variable describe the condition which must have evaluated to true before a participant was asked a *Question* resulting in this variable.

### generationDetails

private *GenerationDetails* **generationDetails**

*GenerationDetails* describe how this variable was generated from one or more input variables.

### id

private **String** **id**

The id of the variable which uniquely identifies the variable in this application. The id must not be empty and must be of the form `var-{{dataAcquisitionProjectId}}-ds{{dataSetNumber}}-{{name}}$`. The id must not contain more than 512 characters.

### indexInDataSet

private **Integer** **indexInDataSet**

The index in the *DataSet* of this variable. Used for sorting the variables of this *DataSet* and for displaying successors and predecessors of this variable. Must not be empty and the successor of this variable must have *indexInDataSet* incremented by one.

### label

private *II8nString* **label**

The label of the variable should describe its content. It must be specified in at least one language and it must not contain more than 512 characters.

### name

private *String* **name**

The name of the variable as it is used in the *DataSet*. It must not be empty and must be unique in the *DataSet*. It must contain only alphanumeric (english) characters and „\_“. The first character must not be a number. It must not contain more than 32 characters.

### panelIdentifier

private *String* **panelIdentifier**

Identifier used to group variables within this *DataSet* which measure the same across multiple waves. Must be of the form `{{dataAcquisitionProjectId}}-ds{{dataSetNumber}}-{{string}}$`. Must not contain more than 512 characters and must contain only (german) alphanumeric characters and „\_“ and „-“,

### relatedQuestions

private *List<RelatedQuestion>* **relatedQuestions**

List of *RelatedQuestions* which have been asked to generate the values of this variable.

### relatedVariables

private *List<String>* **relatedVariables**

List of ids of variables which are „related“ to this variable. The type of relation is arbitrary.

### scaleLevel

private *II8nString* **scaleLevel**

The scale level (or level of measurement) classifies the nature of information within the values assigned to this variable (*ValidResponses*). It determines which mathematical operations can be performed with the values. It must be one of *ScaleLevels* and must not be empty. If the data type of this variable is *DataTypes.DATE* then the ScaleLevel must be *ScaleLevels.ORDINAL*.

### storageType

private *String* **storageType**

Associated with each data type is a storage type. For instance numerics can be stored as integer or double. Must be one of *StorageTypes* and must not be empty.

### **studyId**

private `String` **studyId**

Id of the *Study* to which this variable belongs.

### **surveyIds**

private `List<String>` **surveyIds**

List of ids of *Surveys* which have been conducted to create this variable. Must not be empty.

### **surveyNumbers**

private `List<Integer>` **surveyNumbers**

List of numbers of *Surveys* which have been conducted to create this variable. Must not be empty.



---

Indizes und Tabellen

---

- genindex



### A

absoluteFrequency (Java field), 108, 115  
AbstractRdcDomainObject (Java class), 67  
abstractSource (Java field), 93  
accessWay (Java field), 75, 82  
AccessWays (Java class), 104  
accessWays (Java field), 116  
additionalQuestionText (Java field), 88  
ALL (Java field), 74, 78, 79, 92, 95, 98, 99, 105, 107, 110, 113  
annotations (Java field), 71, 76, 82, 88, 95, 100, 116  
ANY (Java field), 113  
AssigneeGroup (Java enum), 83  
assigneeGroup (Java field), 85  
authors (Java field), 93, 96  
AVAILABLE (Java field), 95

### B

BUILTIN (Java field), 113  
BYTECODE (Java field), 113

### C

CAPI (Java field), 79  
CATI (Java field), 79  
CAWI (Java field), 79  
CHAR (Java field), 113  
CHARACTER (Java field), 113  
citationHint (Java field), 75  
client (Java field), 80  
CLOSURE (Java field), 113  
code (Java field), 108  
COMPLEX (Java field), 113  
Configuration (Java class), 83  
configuration (Java field), 85  
containsAnnotations (Java field), 90  
Counter (Java class), 67  
CREATED (Java field), 81  
createdBy (Java field), 67  
createdDate (Java field), 67

CROSS\_SECTION (Java field), 99  
Customer (Java class), 80  
customer (Java field), 80

### D

DaraUpdateQueueItem (Java class), 84  
data provider, 3, 26  
DATA\_PROVIDER (Java field), 83  
DATA\_SET\_REPORT (Java field), 71  
DataAcquisitionProject (Java class), 85  
dataAcquisitionProjectId (Java field), 71, 73, 76, 77, 83, 88, 91, 96, 97, 100, 102, 104, 116  
DataAvailabilities (Java class), 95  
dataAvailability (Java field), 96  
dataProviders (Java field), 83  
DataSet (Java class), 71  
DataSetAttachmentMetadata (Java class), 73  
dataSetId (Java field), 73, 116  
dataSetIds (Java field), 93  
dataSetNumber (Java field), 73, 116  
dataSetsState (Java field), 84  
DataSetTypes (Java class), 74  
dataType (Java field), 101, 116  
DataTypes (Java class), 99, 105  
DATE (Java field), 105  
date (Java field), 87  
Datenaufnahme, 3, 26  
de (Java field), 68  
derivedVariablesIdentifier (Java field), 117  
description (Java field), 71, 73, 75–77, 96, 97, 100, 102, 107, 108  
deviance (Java field), 111  
Distribution (Java class), 106  
distribution (Java field), 117  
DLP (Java field), 81  
doi (Java field), 93  
DONE (Java field), 70  
doNotDisplayThousandsSeparator (Java field), 117  
DOTDOTDOT (Java field), 113  
DOUBLE (Java field), 113

DOWNLOAD\_CUF (Java field), 105  
 DOWNLOAD\_SUF (Java field), 105

## E

email (Java field), 80  
 en (Java field), 68  
 end (Java field), 68  
 ENVIRONMENT (Java field), 114  
 EPISODE\_RECORD (Java field), 74  
 errorList (Java field), 70  
 eu.dzhw.fdz.metadatamanagement.common.domain  
 (package), 67  
 eu.dzhw.fdz.metadatamanagement.datasetmanagement.domain  
 (package), 71  
 eu.dzhw.fdz.metadatamanagement.instrumentmanagement.domain  
 (package), 75  
 eu.dzhw.fdz.metadatamanagement.ordermanagement.domain  
 (package), 79  
 eu.dzhw.fdz.metadatamanagement.projectmanagement.domain  
 (package), 83  
 eu.dzhw.fdz.metadatamanagement.questionmanagement.domain  
 (package), 88  
 eu.dzhw.fdz.metadatamanagement.relatedpublicationmanagement.domain  
 (package), 93  
 eu.dzhw.fdz.metadatamanagement.studymanagement.domain  
 (package), 95  
 eu.dzhw.fdz.metadatamanagement.surveymangement.domain  
 (package), 99  
 eu.dzhw.fdz.metadatamanagement.variablemanagement.domain  
 (package), 104

EXPRESSION (Java field), 114  
 expression (Java field), 107  
 expressionLanguage (Java field), 107  
 EXTERNALPTR (Java field), 114

## F

FAILURE (Java field), 70  
 fieldPeriod (Java field), 101  
 fileName (Java field), 73, 77, 91, 97, 103, 104  
 FilterDetails (Java class), 107  
 filterDetails (Java field), 117  
 FilterExpressionLanguages (Java class), 107  
 firstName (Java field), 69  
 firstQuartile (Java field), 111  
 Format (Java class), 74  
 format (Java field), 72

## G

GenerationDetails (Java class), 108  
 generationDetails (Java field), 117  
 GRID (Java field), 92  
 grossSampleSize (Java field), 101

## H

hasBeenReleasedBefore (Java field), 86  
 heightY (Java field), 69  
 highWhisker (Java field), 111

## I

I18nString (Java class), 68  
 id (Java field), 68, 70, 72, 73, 76, 78, 80, 82, 85, 86, 89,  
 93, 96, 97, 101, 103, 104, 117  
 ImageType (Java enum), 88  
 imageType (Java field), 91  
 IN\_PREPARATION (Java field), 95  
 indexInDataSet (Java field), 73, 117  
 indexInInstrument (Java field), 78, 89  
 indexInQuestion (Java field), 91  
 indexInStudy (Java field), 97  
 indexInSurvey (Java field), 103  
 institution (Java field), 96  
 instruction (Java field), 89  
 Instrument (Java class), 76  
 InstrumentAttachmentMetadata (Java class), 77  
 InstrumentAttachmentTypes (Java class), 78  
 instrumentId (Java field), 78, 89, 109  
 instrumentIds (Java field), 94  
 instrumentNumber (Java field), 78, 89, 109  
 instrumentsState (Java field), 84  
 InstrumentTypes (Java class), 79  
 INTEGER (Java field), 114  
 INTERVAL (Java field), 110  
 introduction (Java field), 89  
 isDataProviderReady (Java field), 86  
 isDataSetsRequired (Java field), 87  
 isInstrumentsRequired (Java field), 87  
 isPublisherReady (Java field), 86  
 isQuestionsRequired (Java field), 87  
 isStudiesRequired (Java field), 87  
 isSurveysRequired (Java field), 87  
 isVariablesRequired (Java field), 88  
 ITEM\_SET (Java field), 92

## K

kurtosis (Java field), 111

## L

label (Java field), 109, 115, 118  
 LANGUAGE (Java field), 114  
 language (Java field), 74, 78, 91, 92, 94, 98, 103, 104  
 languageKey (Java field), 80  
 lastAssigneeGroupMessage (Java field), 86  
 lastModifiedBy (Java field), 67  
 lastModifiedDate (Java field), 67  
 lastName (Java field), 69  
 LIST (Java field), 114

location (Java field), 70  
 LOGICAL (Java field), 114  
 LONG (Java field), 74  
 lowWhisker (Java field), 111

## M

maximum (Java field), 111  
 maxNumberOfDecimalPlaces (Java field), 106  
 MDM (Java field), 81  
 meanDeviation (Java field), 112  
 meanValue (Java field), 112  
 median (Java field), 112  
 METHOD\_REPORT (Java field), 98  
 middleName (Java field), 69  
 minimum (Java field), 112  
 Missing (Java class), 108  
 missings (Java field), 106  
 MIXED\_METHODS (Java field), 99  
 mode (Java field), 112  
 MULTIPLE\_CHOICE (Java field), 92

## N

name (Java field), 75, 80, 118  
 NOMINAL (Java field), 110  
 NOT\_ACCESSIBLE (Java field), 105  
 NOT\_AVAILABLE (Java field), 95  
 NOTIFIED (Java field), 81  
 NULL (Java field), 114  
 number (Java field), 72, 76, 89, 101  
 numberOfObservations (Java field), 75  
 NUMERIC (Java field), 105

## O

ONSITE\_SUF (Java field), 105  
 OPEN (Java field), 92  
 Order (Java class), 80  
 OrderAlreadyCompletedException (Java class), 81  
 OrderClient (Java enum), 81  
 ORDERED (Java field), 82  
 OrderedStudy (Java class), 82  
 OrderState (Java enum), 81  
 ORDINAL (Java field), 110  
 OTHER (Java field), 78, 98

## P

PAIRLIST (Java field), 114  
 PANEL (Java field), 99  
 panelIdentifier (Java field), 118  
 PAPI (Java field), 79  
 Period (Java class), 68  
 Person (Java class), 69  
 PERSONAL\_RECORD (Java field), 74  
 PNG (Java field), 88

Population (Java class), 100  
 population (Java field), 101  
 Product (Java class), 82  
 products (Java field), 81  
 projectId (Java field), 85  
 ProjectState (Java class), 86  
 Projekt anlegen, 26  
 PROMISE (Java field), 114  
 publicationAbstract (Java field), 94  
 PUBLISHER (Java field), 83  
 publishers (Java field), 84

## Q

QUALITATIVE\_DATA (Java field), 100  
 QUANTITATIVE\_DATA (Java field), 100  
 Question (Java class), 88  
 QUESTION\_FLOW (Java field), 79  
 questionId (Java field), 91, 109  
 questionIds (Java field), 94  
 QuestionImageMetadata (Java class), 90  
 QUESTIONNAIRE (Java field), 79  
 questionNumber (Java field), 109  
 questionsState (Java field), 84  
 questionText (Java field), 89  
 QuestionTypes (Java class), 91

## R

R (Java field), 110  
 RATIO (Java field), 111  
 RAW (Java field), 114  
 RelatedPublication (Java class), 93  
 RelatedQuestion (Java class), 109  
 relatedQuestions (Java field), 118  
 relatedQuestionStrings (Java field), 109  
 relatedVariables (Java field), 118  
 relativeFrequency (Java field), 109, 115  
 Release (Java class), 86  
 release (Java field), 86  
 REMOTE\_DESKTOP (Java field), 105  
 Requirements (Java class), 87  
 requirements (Java field), 84  
 Resolution (Java class), 69  
 resolution (Java field), 91  
 responseRate (Java field), 101  
 rule (Java field), 108  
 ruleExpressionLanguage (Java field), 108  
 RuleExpressionLanguages (Java class), 110  
 RUNNING (Java field), 71

## S

S4 (Java field), 115  
 sample (Java field), 101  
 sampleSize (Java field), 102  
 scaleLevel (Java field), 118

ScaleLevels (Java class), 110  
seq (Java field), 68  
SINGLE\_CHOICE (Java field), 92  
skewness (Java field), 112  
source (Java field), 93  
sourceLink (Java field), 94  
sourceReference (Java field), 94  
SPECIAL (Java field), 115  
SPEL (Java field), 107  
sponsor (Java field), 96  
standardDeviation (Java field), 112  
start (Java field), 68  
STATA (Java field), 108, 110  
state (Java field), 70, 81  
Statistics (Java class), 111  
statistics (Java field), 106  
storageType (Java field), 118  
StorageTypes (Java class), 112  
STRING (Java field), 106  
studiesState (Java field), 84  
Study (Java class), 95  
study (Java field), 83  
StudyAttachmentMetadata (Java class), 97  
StudyAttachmentTypes (Java class), 98  
studyId (Java field), 72, 76, 90, 98, 102, 119  
studyIds (Java field), 94  
studySeries (Java field), 96  
studySerieses (Java field), 94  
SubDataSet (Java class), 75  
subDataSets (Java field), 72  
subtitle (Java field), 76  
successorNumbers (Java field), 90  
successors (Java field), 90  
Survey (Java class), 100  
SurveyAttachmentMetadata (Java class), 102  
SurveyDataTypes (Java class), 98  
surveyDesign (Java field), 97  
SurveyDesigns (Java class), 99  
surveyId (Java field), 103, 104  
surveyIds (Java field), 72, 77, 94, 119  
surveyMethod (Java field), 102  
surveyNumber (Java field), 103, 104  
surveyNumbers (Java field), 72, 77, 119  
SurveyResponseRateImageMetadata (Java class), 103  
surveysState (Java field), 84  
SYMBOL (Java field), 115

## T

Task (Java class), 69  
TaskState (Java enum), 70  
TaskType (Java enum), 71  
TechnicalRepresentation (Java class), 92  
technicalRepresentation (Java field), 90  
thirdQuartile (Java field), 112

title (Java field), 74, 77, 82, 94, 97, 98, 100, 102, 103  
topic (Java field), 90  
totalAbsoluteFrequency (Java field), 106  
totalValidAbsoluteFrequency (Java field), 106  
totalValidRelativeFrequency (Java field), 106  
type (Java field), 70, 72, 77, 78, 90, 93, 98

## U

UNDOCUMENTED (Java field), 92  
updateStartedAt (Java field), 85  
updateStartedBy (Java field), 85

## V

validRelativeFrequency (Java field), 115  
ValidResponse (Java class), 115  
validResponses (Java field), 107  
value (Java field), 116  
Variable (Java class), 116  
VARIABLE\_QUESTIONNAIRE (Java field), 79  
variableIds (Java field), 94  
variablesState (Java field), 84  
version (Java field), 67, 83, 87

## W

wave (Java field), 102  
WEAKREF (Java field), 115  
WIDE (Java field), 75  
widthX (Java field), 69

## Y

year (Java field), 95