

Leistungsbeschreibung

„Programmierung einer digitalen Anwendung für die Testaufgabenentwicklung in der Mathematik der Sekundarstufe I“

Stand 01.06.2026

Ab dem Jahr 2030 sollen im Rahmen des nationalen Bildungsmonitorings (IQB-Bildungstrend) technologiebasierte Testaufgaben auf Grundlage der weiterentwickelten Bildungsstandards in der Mathematik der Sekundarstufe I eingesetzt werden. Dafür werden seit September 2025 im Zuge des Vorhabens BiStaTest Mathematik am Institut zur Qualitätsentwicklung im Bildungswesen (IQB) geeignete Testaufgaben entwickelt bzw. der Aufgabenbestand aus den bisherigen Bildungstrendstudien überarbeitet und modernisiert.

Neben klassischen Formaten (MC-Aufgaben, Kurzantworten, Begründungsaufgaben) sollen auch innovative Aufgabenformate entstehen. Für das Fach Mathematik in der Sekundarstufe I ist in diesem Zusammenhang geplant, für die Bearbeitung einiger Aufgaben einen digitalen Taschenrechner zur Verfügung zu stellen.

Die Programmierungen des IQB sollen hierzu um Komponenten erweitert werden, die als modale Dialoge über die existierende Programmierung gelegt werden. Nachfolgend ist die Leistungserbringung beschrieben.

1 Grundlagen für die Erbringung der Leistungen

- Ergebnis der Leistungserbringung ist eine Webanwendung. Diese Webanwendung ist als eine Html-Seite zu übergeben. Sämtliche Ressourcen müssen eingebettet sein. Diese Html-Seite wird nachfolgend „Widget“ genannt.
- Das IQB-Testsystem wird ein Widget jeweils in ein iframe-Element setzen und in einen modalen Dialog einbetten. Die Datenkommunikation zwischen dem Widget und der IQB-Webanwendung ist über eine veröffentlichte Schnittstelle¹ standardisiert.
- Das Widget muss über TypeScript programmiert sein. Als Framework wird Angular bevorzugt.
- Der Quellcode sowie alle Ressourcen müssen als Open Source veröffentlicht werden. Die Nutzungsrechte, die der Auftragnehmer dem Auftraggeber zur Verfügung stellen muss, haben der MIT-Lizenz, ergänzt um eine für die Rechtspraxis in Deutschland angepasste Variante², zu entsprechen.
- Die Darstellung und die Funktionen müssen für die Zielgruppe 8.-10. Klasse optimiert werden.
- Als Endgeräte müssen Desktop-Computer, Laptops und Tablets unterstützt werden.
- Das Widget muss auf den folgenden Browsern laufen: Chrome (aktuelle und vorherige Version), Firefox (aktuelle und letzte ESR-Version) und Safari (aktuelle und letzte Hauptversion). Als Zeitpunkt für die Festlegung „aktuell“ gilt die Leistungsabnahme.

2 Zu erbringende Leistungen

Testaufgaben für das Fach Mathematik sollen zur Bearbeitung einen digitalen Taschenrechner zur Verfügung stellen. Das Widget „Taschenrechner“ muss folgende Funktionen erfüllen:

¹ <https://verona-interfaces.github.io/widget-docs/calc.html>

² <https://iqb-berlin.github.io/license.html>

- Das Widget muss die Eingabe und korrekte Funktionalität der folgenden Funktionen ermöglichen: Plus, Minus, Mal, geteilt, Ziffern (0-9), Dezimalkomma, Rücksetzen, Löschen, Istgleich (=), Istgleichgerundet (\approx), Klammern, Brüche, Quadratwurzel, Quadrat und die Zahl Pi. Das Ergebnis einer Rechnung soll hierbei sowohl ungerundet (z.B. als gekürzten Bruch) als auch gerundet (als Dezimalzahl) dargestellt werden können.
- Das Widget muss eine Historie der letzten 3 Rechnungen anzeigen können.
- Zum Start des Widgets muss eine über die Kommunikationsschnittstelle mit dem IQB-System übergebene Auswahl sofort visuell sichtbar sein. Das Widget muss also den Stand eines früheren Aufrufs wiederherstellen können.

In Abstimmung mit dem Auftraggeber muss der Auftragnehmer die Darstellung und die Interaktion entwerfen (UI/UX) und anschließend implementieren. Nachfolgend einige Punkte zur Orientierung:

- Der Taschenrechner muss einer im modernen digitalen Raum üblichen Darstellung folgen. Beispiele dafür wären Taschenrechner-Apps unter Android oder iOS. Die Anordnung der Elemente muss in Abstimmung mit dem Auftraggeber implementiert werden.

3 Zeitplan und Abnahmen

Das Arbeitspaket muss nach dem folgenden Muster realisiert werden.

3.1 Auftaktworkshop

Der Termin des Online-Workshops wird in den ersten zwei Wochen nach Vergabe des Auftrags stattfinden und ca. 3 Stunden dauern. Der Auftaktworkshop ist durch den Auftragnehmer zu organisieren.

Dieser Workshop hat drei Ziele: Erstens muss der Auftragnehmer ein gutes Verständnis der inhaltlichen Anforderungen des Widgets entwickeln. Zweitens muss der Auftragnehmer den Zeitplan mit dem Auftraggeber abstimmen. Drittens müssen im Workshop gemeinsam mit dem Auftraggeber Rollen, Zuständigkeiten und Kommunikationswege geklärt werden.

3.2 Erarbeitung Konzept

Die schriftliche Aufbereitung der Ergebnisse des Auftaktworkshops hat durch den Auftragnehmer zu erfolgen. Auf Grundlage der Ergebnisse dieses Workshops hat der Auftragnehmer den im Realisierungskonzept beschriebenen Entwurf für das UI/UX-Konzept weiterzuentwickeln und dem Auftraggeber zwei Wochen nach dem Workshop vorzulegen. Der Auftraggeber teilt dem Auftragnehmer innerhalb von 1 Woche Änderungsbedarf schriftlich per E-Mail mit.

3.3 Implementierung/Programmierung

Der Auftragnehmer beginnt spätestens drei Wochen nach Auftragsvergabe mit der Implementierung. Dabei muss der Auftragnehmer die Änderungswünsche des Auftraggebers (siehe unter 3.2) berücksichtigen. Spätestens fünf Wochen nach Auftragsvergabe präsentiert der Auftragnehmer dem Auftraggeber im Rahmen eines zweistündigen Online-Workshops eine funktionsfähige Version des Widgets. Der Bereich UI/UX und die Datenkommunikation des Widgets mit der IQB-Webanwendung müssen vollständig implementiert sowie die Datenkommunikation mithilfe der vom Auftraggeber gestellten Testumgebung getestet worden sein.

Der Auftragnehmer muss die im Workshop beschlossenen Änderungen anschließend umsetzen und spätestens in KW 37 dem Auftraggeber das Widget in der finalen Version übergeben. Der Auftraggeber hat 2 Wochen Zeit, der finalen Version zuzustimmen. Mit Zustimmung des Auftraggebers gilt das

Widget als abgenommen. Meldet sich der Auftraggeber nicht innerhalb von 2 Wochen, gilt das Arbeitsergebnis ebenfalls als abgenommen.

4 Nutzungsrechte, Lizenz und Veröffentlichung

Der Auftragnehmer hat dem Auftraggeber das zeitlich und räumlich unbeschränkte, ausschließliche und übertragbare Recht zu übertragen, die aus der Leistungserbringung resultierenden Informationen und Arbeitsergebnisse auf sämtlichen Nutzungsarten nutzen zu können, insbesondere zu vervielfältigen, zu be- und verarbeiten und zu verbreiten sowie analog und digital zu veröffentlichen. Daher müssen die Arbeitsergebnisse des Auftragnehmers frei von Rechten Dritter sein, die einer vertragsgemäßen Nutzung entgegenstehen. Der Auftragnehmer darf bei der Erstellung der Programmierung und der UI/UX-Elemente nur Open Source Komponenten einsetzen, deren Nutzung beim Auftraggeber nicht zur Notwendigkeit des Abschlusses kostenpflichtiger Lizenzverträge führt. Die Erstellung der Individualsoftware (= Softwareprogramme, Programm-Module, Tools etc., die zur Vertragserfüllung für die Bedürfnisse des Auftraggebers vom Auftragnehmer erstellt wurden einschließlich der zugehörigen Dokumentation) ist mit dem Pauschalpreis gem. Ziffer 8.2 der Bewerbungsbedingungen abgegolten. Der Auftragnehmer darf bei der Erstellung der Individualsoftware nur Open Source Komponenten einsetzen, deren Nutzung beim Auftraggeber nicht zur Notwendigkeit des Abschlusses kostenpflichtiger Lizenzverträge führt. Der Auftragnehmer muss von allen an der Leistung beteiligten Personen die Zustimmung zur Abtretung der Nutzungsrechte an den Auftraggeber schriftlich einholen und die Dokumente dem Auftraggeber zur Verfügung stellen. Gegebenenfalls anfallende Kosten für Nutzungsrechte hat der Auftragnehmer zu tragen.

Der Auftragnehmer hat folgende Bedingungen einzuhalten:

- Alle hier zu erbringenden Leistungen müssen als Open Source veröffentlicht werden.
- Als Ort der Veröffentlichung muss GitHub.com gewählt werden. In Abstimmung mit dem Auftraggeber kann davon abgewichen werden.
- Alle hier zu erbringenden Leistungen müssen unter der für den deutschen Rechtsverkehr angepassten Lizenz MIT veröffentlicht werden³.
- Sämtliche für eine Installation erforderlichen Komponenten müssen öffentlich und kostenlos verfügbar sein.
- Alle Dokumentationen und die damit bereitgestellten Medien müssen unter der Lizenz „Creative Commons Zero“ 1.0⁴ veröffentlicht werden.

Da es sich bei der Individualsoftware um Open Source Software handelt, findet Ziffer 2.1.2.1 EVB-IT Erstellungs-AGB keine Anwendung. Anstelle dessen gelten ausschließlich die Rechte der Open Source Lizenz/en, die für die Nutzung der vom Auftragnehmer verwendeten Open Source Komponente/n einschlägig ist/sind.

5 Mitwirkungsobliegenheiten des Auftraggebers

Der Auftraggeber hat die folgenden Mitwirkungsobliegenheiten:

- Für die Workshops benennt der Auftraggeber die relevanten Teilnehmenden.
- Der Auftraggeber unterstützt das Verständnis über die Kommunikationsschnittstelle beim Auftragnehmer bestmöglich. Hierzu gehört eine öffentliche Spezifikation sowie ausführliche Erläuterungen.

³ Verbindlicher Text siehe hier: <https://iqb-berlin.github.io/license.html>

⁴ s. <https://open-educational-resources.de/was-die-cc0-freigabe-bedeutet-und-welche-moeglichkeiten-sie-eroeffnen/>

- Der Auftraggeber stellt eine Testumgebung bereit, mit der das Widget einschließlich der Kommunikation getestet werden kann⁵.
- Der Auftraggeber bereitet einen Ort für die Veröffentlichung der Programmierungen auf GitHub.com vor.

6 Rangfolge der Vertragsbestandteile

Die nachstehenden Dokumente werden Vertragsbestandteile:

1. Die Leistungsbeschreibung
2. Die sonstigen Vergabeunterlagen
3. Ggf. Bieterfrage im Vergabeverfahren sowie deren Beantwortung durch den Auftraggeber
4. Der gesamte Inhalt des Angebots des Auftragnehmers vom ...
5. Die Ergänzenden Vertragsbedingungen für die Erstellung bzw. Anpassung von Software (EVB-IT Erstellungs-AGB) in der bei Versand der Vergabeunterlagen geltenden Fassung.
6. Teil B der Vergabe- und Vertragsordnung für Leistungen in der Fassung der Bekanntmachung vom 5. August 2003 (BAVz. Nr. 178a, VOL/B)

Im Falle von Widersprüchen gelten die vorstehenden Dokumente in der dort genannten absteigenden Reihenfolge.

⁵ s. <https://github.com/iqb-berlin/verona-player-testbed>