

Version: V0.1

Wetterarchiv

Softwareentwicklungsprojekt – Abgaben



Hochschule: DHBW-VS

Modul: Projekt / Anwendungsentwicklung

Team: Finja Sterner, Lisa Fritzsche, Jan Goliasch,
Menko Hornstein

Dokumentensteuerung

Dieses Dokument bündelt alle relevanten Abgaben, Meilensteine und Nachweise (z. B. Demos, Tests, Protokolle) für das Projekt „Wetterarchiv“.

Versionshistorie

Version	Datum	Autor(en)	Änderung
v0.1	28.01.2026	Menko Hornstein, Jan Goliasch, Lisa Fritsche, Finja Sterner	Initiale Vorlage erstellt, Protokolle ergänzt, MS1 Ergänzt
V0.1.1	30.01.2026	Finja Sterner, Menko Hornstein	Korrektur von D4 auf Seite 17

Inhaltsverzeichnis

Dokumentensteuerung	2
Versionshistorie	2
Inhaltsverzeichnis	3
Qualitätsanforderungen	4
MS1: Abgabe Use-Case(s) und GUI-Entwurf	6
<i>Use Cases</i>	6
Use-Case <1>: Wetterstation finden und festlegen	6
Use-Case <2>: Wetterdaten Analysieren	9
<i>GUI-Entwurf</i>	12
Ansicht 1 – Stationssuche	12
Ansicht 2 – Chart-Ansicht	13
Ansicht 3 – Table-Ansicht	14
Meeting-Protokolle	15
1. Meeting	15
2. Meeting	17

Qualitätsanforderungen

Architecture Communication Canvas:

1. Vollständigkeit des Canvas
 - Alle Inhalte vorhanden
 - Alle Inhalte verständlich
2. Darstellung des Canvas
 - Format des Canvas angemessen
 - Alle Inhalte sichtbar
3. Abgrenzung des Wertever sprechens und der Kernfunktionalität
 - Wertever sprechen angemessen
 - Alle Kernfunktionalitäten vorhanden
4. Auflistung der Komponenten und Technologien
 - Separierung in Komponenten sinnvoll
 - Beschreibung der Komponenten angemessen
 - Einsatz von Technologien für die jeweiligen Komponenten sinnvoll
5. Auflistung von Stakeholdern und Abgrenzung des Businesskontexts
 - Analyse von Stakeholdern sinnvoll
 - Zuordnung von Rollen zu Stakeholdern sinnvoll
 - Beschreibung des Businesskontexts angemessen
6. Erstellung eines Risikomanagements und Ableitung von Entscheidungen
 - Analyse von Risiken inkl. Eintrittswahrscheinlichkeit und Schadenspotential sinnvoll
 - Auswahl von Entscheidungen/Maßnahmen angemessen

Produkt:

- Funktionale Eignung - Angemessenheit der Datenhaltung
 - Datenstrukturen/-schemata sinnvoll
 - Datenbanktechnologie sinnvoll
 - Einsatz von Cachingmechanismen angemessen
- Effizienz - Laufzeitverhalten
 - Lighthouse-Performancescan
 - Alle Kernfunktionalitäten bei Verfügbarkeit der abhängigen Systeme unter 3 Sekunde ausführbar
 - Oberfläche reagiert bei jeder Interaktion innerhalb von 0,5 Sekunden (z. B. mit Ladeanimation)

- Interaktionskapazitäten - Erlernbarkeit
 - Nutzbarkeit ohne Anleitung
- Interaktionskapazitäten - Inklusivität
 - Lighthouse-Barrierefreiheitsscan
- Interaktionskapazitäten - Selbstbeschreibung
 - Ästhetik
 - Kurze Klickpfade
 - Nutzbarkeit ohne Anleitung
- Wartbarkeit - Modularität
 - Sinnvolle Codestruktur
 - Verständlichkeit des Codes
 - Sinnvoller Einsatz von Bibliotheken
- Wartbarkeit - Testbarkeit
 - Angemessene Teststrategie
 - Testabdeckung
 - Angemessener Einsatz von Mocking/Stubbing
 - Verständlichkeit der Tests
- Flexibilität - Installierbarkeit
 - CI/CD-Pipeline mithilfe von GitHub Actions
 - Bauen von Containerimages und Ablegen in der GitHub Container Registry
 - Bedarfsgerechte Verwendung von OpenSource-Containerimages
 - Erstellung einer Installationskonfiguration

MS1: Abgabe Use-Case(s) und GUI-Entwurf

Use Cases

Use-Case <1>: Wetterstation finden und festlegen

Überblick:

Der Nutzer möchte eine geeignete Wetterstation in der Nähe eines geografischen Standorts finden, um eine Grundlage für eine spätere Wetterdatenanalyse zu erhalten.

Geltungsbereich:

Webanwendung zur Analyse historischer Temperaturdaten auf Basis des GHCN-Datensatzes.

Anwendungsschicht:

- Benutzer

Primärer Akteur:

- Benutzer

Weitere Akteure:

- API (Stationsdaten)

Stakeholder und ihre Belange:

- Nutzer: Möchte geeignete Wetterstationen in räumlicher Nähe finden, um eine fundierte Auswahl für eine spätere Datenanalyse treffen zu können. Erwartet eine übersichtliche und nachvollziehbare Ergebnisliste.

Vorbedingungen:

- Die Wetterstationsdaten sind vorhanden und im System (über die API) verfügbar.

Nachbedingungen:

- Eine Wetterstation ist als Analysegrundlage festgelegt

Hauptzweig:

1. Der Nutzer legt einen geografischen Standort fest.
2. Der Nutzer definiert die Suchkriterien zur Einschränkung der Ergebnismenge (Suchradius, Maximale Anzahl der Wetterstationen (Top N, aufsteigend nach Distanz geordnet)).
3. Nutzer löst die Suche aus
4. Das System ermittelt Wetterstationen, die den definierten Kriterien entsprechen.
5. Das System stellt die gefundenen Wetterstationen als Ergebnis der Suche bereit.
6. Der Nutzer wählt eine Wetterstation aus den Suchergebnissen aus.
7. Das System legt die ausgewählte Wetterstation als Analysegrundlage fest.

Erweiterungen:

3a. Es werden keine Passenden Wetterstationen gefunden.

1. Das System informiert den Nutzer darüber und ermöglicht eine Anpassung der Suchkriterien.
2. Der Anwendungsfall endet ohne Festlegung einer Wetterstation

3b. Die Stationssuche ist nicht durchführbar, weil die API nicht verfügbar ist.

1. Das System informiert den Nutzer über die Nichtverfügbarkeit (z. B. Serverfehler/Timeout).
2. Das System bietet „Erneut versuchen“ an und ermöglicht ggf. die Anpassung der Suchkriterien später.
3. Der Anwendungsfall endet ohne Festlegung einer Wetterstation.

5a. Der Nutzer bricht die Auswahl ab.

1. Der Anwendungsfall endet ohne Festlegung einer Wetterstation

Besondere Anforderungen:

- Die Suche soll innerhalb der Zeitlichen grenzen erfolgen (Maximal 3 Sekunden).
- Die Ergebnisliste soll übersichtlich und nachvollziehbar dargestellt werden.

Variationen in Bezug auf Technologie und Daten:

- Der Startpunkt der Suche kann auf unterschiedliche Weise bestimmt werden, z.B. durch direkte Koordinateneingabe oder durch eine positionsbasierte Auswahl.

Häufigkeit des Auftretens:

- Häufig

Sonstiges:

Use-Case <2>: Wetterdaten Analysieren

Überblick:

Der Nutzer möchte die Temperaturdaten einer festgelegten Wetterstation für definierte Zeiträume und meteorologische Jahreszeiten analysieren, um zeitliche Entwicklungen nachvollziehen zu können.

Geltungsbereich:

Webanwendung zur Analyse historischer Temperaturdaten auf Basis des GHCN-Datensatzes

Anwendungsschicht:

- Benutzer

Primärer Akteur:

- Benutzer

Weitere Akteure:

- API (Stationsdaten)
- Stationsdatenhaltung (GHCN-Datenhaltung)

Stakeholder und ihre Belange:

- Nutzer: Möchte Temperaturentwicklungen über definierte Zeiträume systematisch analysieren, um Trends, saisonale Muster und zeitliche Veränderungen erkennen und interpretieren zu können. Erwartet eine konsistente, nachvollziehbare und vergleichbare Analysegrundlage.

Vorbedingungen:

- Der Anwendungsfall UC1: Wetterstation finden und festlegen wurde erfolgreich durchgeführt.
- Für die festgelegte Wetterstation liegen Temperaturdaten im System vor.

Nachbedingungen:

Die Temperaturdaten der festgelegten Wetterstation sind entsprechend den gewählten Analyseparametern ermittelt und als Analyseergebnis im System verfügbar.

Hauptzweig:

1. Der Nutzer startet die Analyse der Temperaturdaten der festgelegten Wetterstation.
2. Der Nutzer legt fest, welche Temperaturwerte betrachtet werden sollen (z. B. Minimal- und/oder Maximaltemperatur).
3. Der Nutzer definiert die zeitliche Struktur der Analyse (z. B. Jahreswerte oder meteorologische Jahreszeiten).
4. Das System ermittelt die relevanten Temperaturdaten entsprechend den gewählten Analyseparametern.
5. Das System bereitet die ermittelten Temperaturdaten als Analyseergebnis auf.
6. Das System stellt das Analyseergebnis zur Interpretation bereit. (Grafisch und tabellarisch)

Erweiterungen:

4a. Für die gewählte Analysekonfiguration liegen keine Daten vor.

1. Das System informiert den Nutzer darüber.
2. Der Nutzer kann die Analyseparameter anpassen.
3. Der Anwendungsfall endet ohne Analyseergebnis.

4b. Die Analyse ist nicht durchführbar, weil die API nicht verfügbar ist.

1. Das System informiert den Nutzer über die Nichtverfügbarkeit (z. B. Serverfehler/Timeout).
2. Der Nutzer kann „Erneut versuchen“ oder Analyseparameter später anpassen.
3. Der Anwendungsfall endet ohne Analyseergebnis.

4c. Die Analyse ist nicht durchführbar, weil die Original-Datenquelle nicht erreichbar ist.

1. Das System informiert den Nutzer, dass die Datenquelle/Datenhaltung nicht verfügbar ist.
2. Der Anwendungsfall endet ohne (aktuelles) Analyseergebnis.

4d. Die Anfrage dauert zu lange (Zeitüberschreitung).

1. Das System bricht die Anfrage nach einem definierten Timeout ab und informiert den Nutzer.
2. Das System empfiehlt Parameter zu reduzieren (z. B. kürzerer Zeitraum / weniger Aggregationen) und bietet „Erneut versuchen“ an.
3. Der Anwendungsfall endet ohne Analyseergebnis.

Besondere Anforderungen:

- Die Darstellung der Temperaturdaten soll übersichtlich und gut interpretierbar sein. (Ladezeit maximal 3 Sekunden)
- Die numerischen Werte müssen vollständig und exakt einsehbar sein.

Variationen in Bezug auf Technologie und Daten:

- Niederschlagsmenge
- Keine Daten

Häufigkeit des Auftretens:

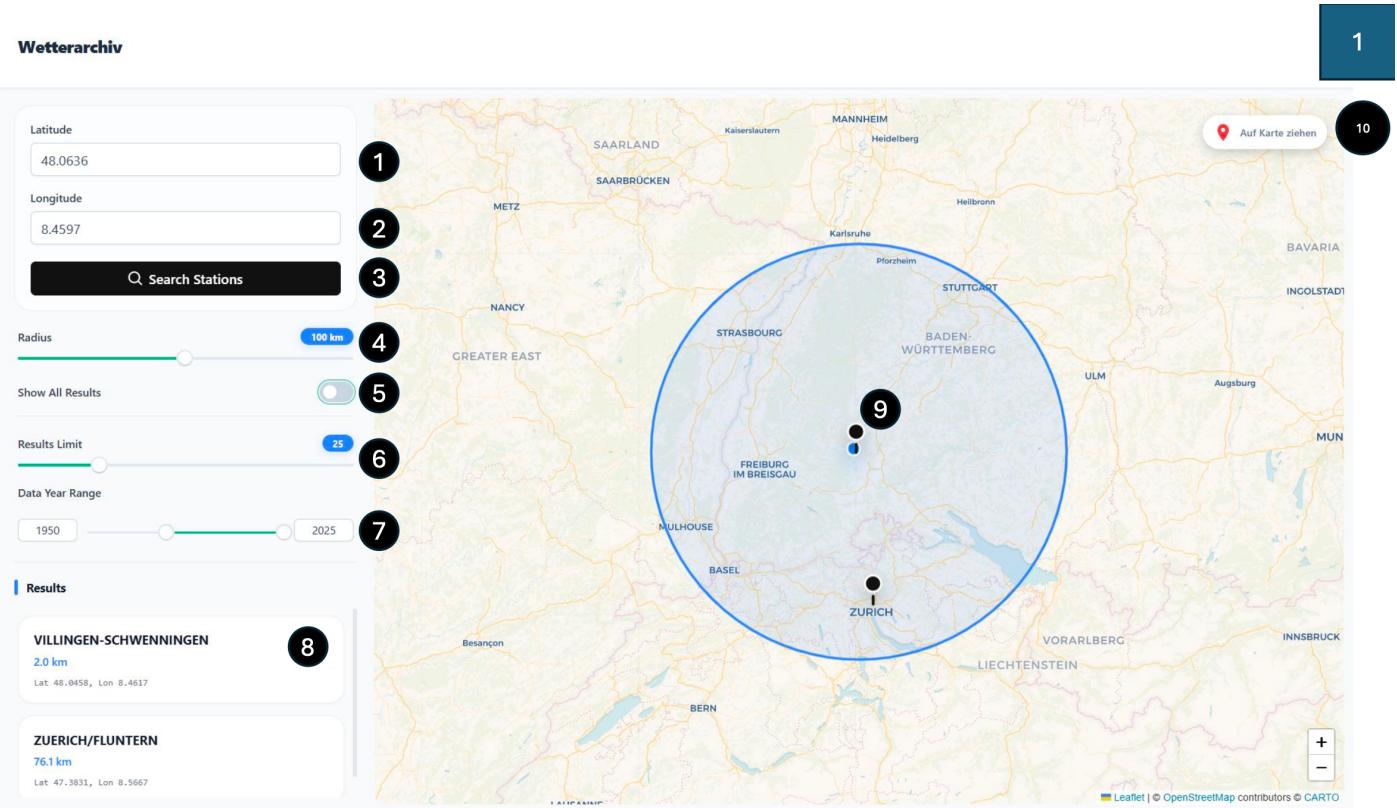
- Häufig

Sonstiges:

-

GUI-Entwurf

Ansicht 1 – Stationssuche



1. Eingabefeld „Latitude“:

Eingabefeld zur Angabe der geografischen Breite des Suchstandorts.

2. Eingabefeld „Longitude“:

Eingabefeld zur Angabe der geografischen Länge des Suchstandorts.

3. Button „Search Stations“:

Startet die Suche nach Wetterstationen basierend auf den eingegebenen Standort- und Suchparametern.

4. Slider „Radius“:

Ermöglicht die Festlegung des Suchradius in Kilometern, innerhalb dessen Wetterstationen berücksichtigt werden.

5. Button „Show All Results“:

Hebt die Begrenzung der Ergebnisanzahl sowie die Einschränkung auf das Vorhandensein der spezifizierten Jahre auf und wählt alle Wetterstationen, die dem Suchstandort und Radius entsprechen.

6. Slider „Results Limit“:

Begrenzt die Anzahl der angezeigten Wetterstationen auf eine definierte Maximalanzahl, sortiert nach der Entfernung zum Suchstandort.

7. Slider „Date Year Range“:

Legt den Zeitraum fest, für den Wetterstationen über verfügbare Messdaten verfügen müssen.

8. Ergebnisfeld „Results“:

Zeigt gefundene Wetterstation(en) an. Durch Auswahl einer Station kann zur Analyse der zugehörigen Wetterdaten gewechselt werden.

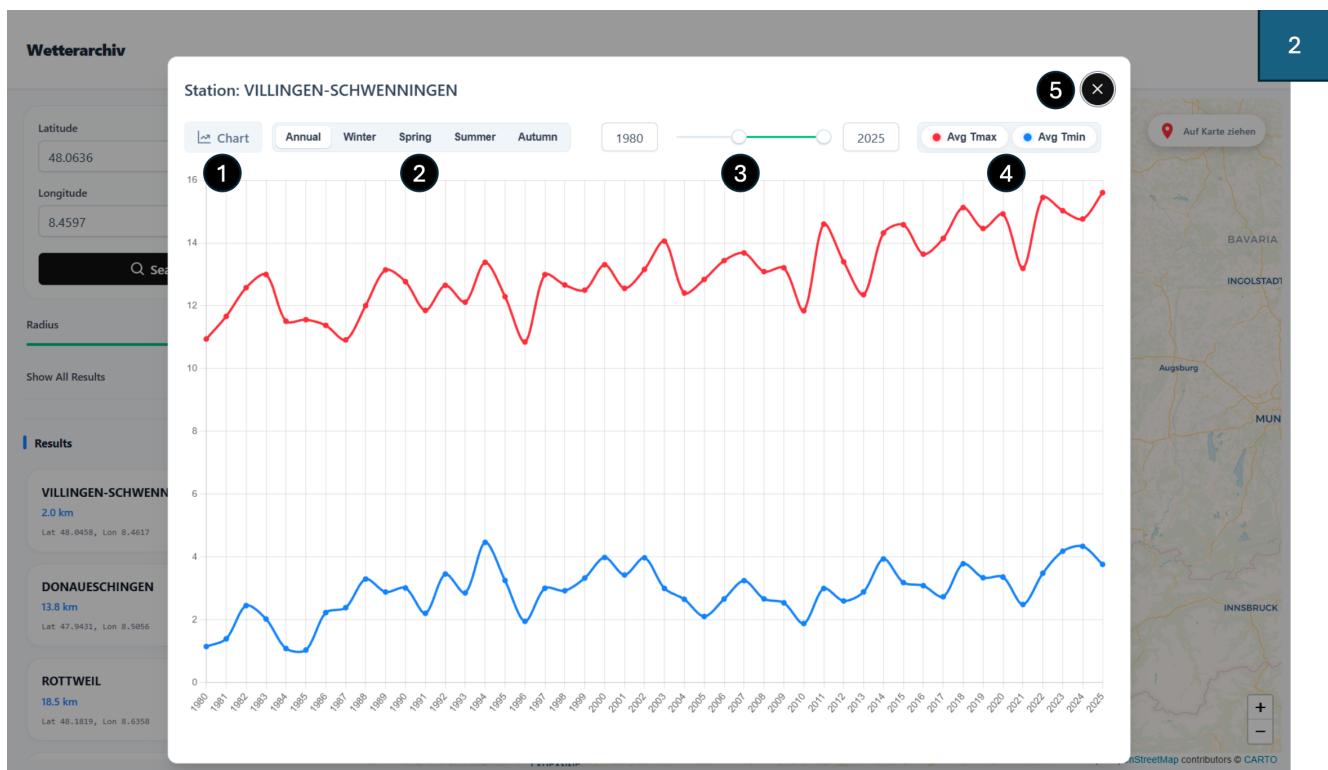
9. Markierung der Wetterstation auf der Karte:

Visualisiert die Position gefundener Wetterstation auf der Karte und ermöglicht alternativ die Auswahl einer Station für die weitere Analyse.

10. Drag-and-Drop-Modul „Auf Karte ziehen“:

Ermöglicht die Festlegung des Suchstandorts durch Platzierung auf der Karte mittels Drag-and-Drop.

Ansicht 2 – Chart-Ansicht



1. Button „Chart“:

Umschaltfunktion zwischen grafischer Darstellung (Chart) und tabellarischer Darstellung (Table) der Temperaturdaten.

2. Auswahlfeld „Annual, Winter, Spring, Summer, Autumn“:

Ermöglicht die Auswahl des zeitlichen Aggregationszeitraums für die Analyse der Temperaturdaten.

3. Slider „Date Year Range“:

Begrenzt den betrachteten Zeitraum der Temperaturdaten innerhalb der grafischen Darstellung.

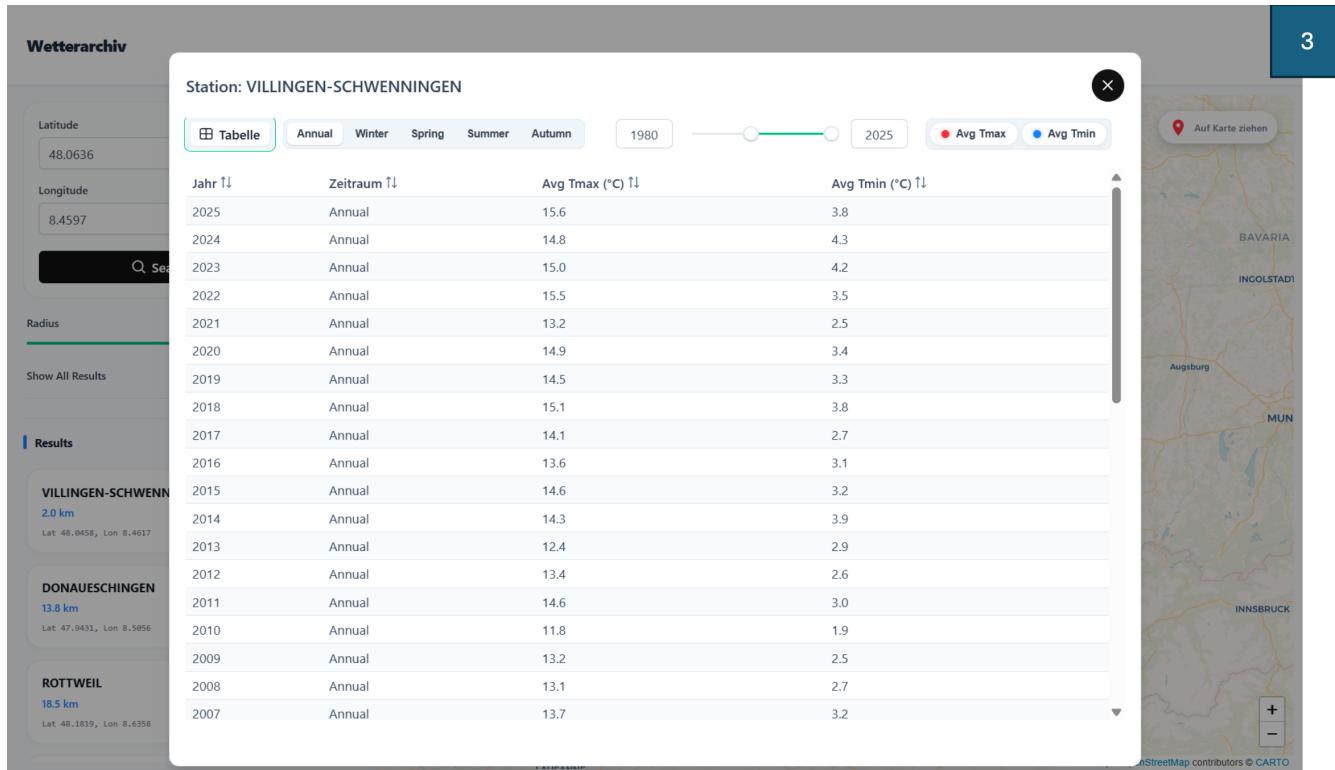
4. Knöpfe „Avg Tmax“, „Avg Tmin“:

Ermöglicht das Ein- und Ausblenden der dargestellten Temperaturreihen für maximale und minimale Durchschnittstemperaturen.

5. Button „Zurück“:

Wechselt zurück zur vorherigen Ansicht zur Auswahl oder Suche weiterer Wetterstationen.

Ansicht 3 – Table-Ansicht



= Stellt die Wetterdaten in tabellarischer Form dar und ermöglicht eine detaillierte Einsicht in die numerischen Werte pro Jahr und Zeitraum.

Meeting-Protokolle

1. Meeting

Datum: 16.01.2026

Uhrzeit: 09:00 – ca. 09:30

Format: Zoom-Meeting

Projektphase: Iteration 1 – Anforderungserfassung (MS1)

Teilnehmer:

- Auftraggeber: Prof. Dr. Wolfgang Funk
- Projektteam:
 - Lisa Fritzsche (Projektleitung)
 - Menko Hornstein
 - Finja Sterner
 - Jan Goliasch

Protokoll: Lisa Fritzsche (Projektleitung)

1) Ziel des Meetings

Abstimmung der funktionalen Anforderungen für die Webanwendung sowie Klärung offener Fragen zur weiteren Vorgehensweise.

2) Zentrale Ergebnisse & Entscheidungen (Decision Log)

Nr.	Entscheidung	Kontext / Begründung	Verantwortlich
D1	Die funktionalen Anforderungen werden primär über Use Cases definiert.	Use Cases dienen als verbindliche Grundlage für GUI-Entwurf, Implementierung und Abnahmetest.	Projektteam
D2	Die Kommunikation mit dem Auftraggeber erfolgt über die DHBW-Mailadresse der Projektleitung.	Sicherstellung einer zentralen, nachvollziehbaren Kommunikation.	Projektleitung
D3	Der bestehende UI-Entwurf wird an die abgestimmten Anforderungen angepasst.	Der aktuelle Entwurf entspricht noch nicht vollständig den besprochenen funktionalen Anforderungen.	Projektteam

3) Offene Punkte / ToDos

Nr.	Aufgabe	Verantwortlich	Deadline	Status
A1	UI-Entwurf an die abgestimmten funktionalen Anforderungen anpassen	Menko Hornstein	offen	open
A2	Klärung des Ablageorts der Datenbankdateien innerhalb der Systemarchitektur mit dem technischen Consultant	Lisa Fritsche	offen	open

4) Relevante fachliche & technische Hinweise

- **Anforderungen / Use Cases:**

Die Use Cases bilden die zentrale Referenz für alle weiteren Projektschritte. Die Use Cases werden mit dem Auftraggeber unter Zusendung abgestimmt

- **Architektur / CI/CD:**

Details zur Datenhaltung und zur Ablage von Datenbankdateien sind noch mit dem technischen Consultant abzustimmen.

- **UI / GUI (konkretisiert):**

- Ein Reset-Button soll den zuletzt gesetzten Zustand der Filter wiederherstellen.
- Die Option „from map“ ist optional
- Die Option „all stations“ ist nicht erforderlich.
- Die Anforderung der maximalen Verzögerung von 3 Sekunden gilt bei max parametern:
 - Suchradius maximal 100 km.
 - maximal die Top 10 nächstgelegenen Stationen

5) Nächster Termin

Ein Folgetermin wird von der Projektleitung mit dem Auftraggeber vereinbart. Ein Folgetermin wird mit dem technischen Consultant vereinbart.

6) Hinweise

Dieses Protokoll gilt als verbindlich, sofern innerhalb von drei Werktagen nach Versand kein Widerspruch erfolgt.

2. Meeting

Datum: 27.01.2026

Uhrzeit: 10:00 – 10:45

Format: Zoom-Meeting

Projektphase: Iteration 1 – Anforderungserfassung (MS1)

Teilnehmer:

- Auftraggeber: Prof. Dr. Wolfgang Funk
- Projektteam:
 - Lisa Fritzsche (Projektleitung)
 - Menko Hornstein
 - Finja Sterner
 - Jan Goliasch

Protokoll: Lisa Fritzsche (Projektleitung)

1) Ziel des Meetings

Konkretisierung der Use Cases, Abstimmung der Systemarchitektur sowie Präzisierung von Anforderungen an Datenhaltung, GUI und Qualitätsanforderungen.

2) Zentrale Ergebnisse & Entscheidungen (Decision Log)

Nr.	Entscheidung	Kontext / Begründung	Verantwortlich
D1	Es werden zwei zentrale Use Cases definiert.	Fokus auf klare fachliche Abgrenzung der Kernfunktionalitäten.	Projektteam
D2	Die Architektur folgt einem Client-Server-Modell mit einer API.	Stations- und Messdaten werden serverseitig über eine API bereitgestellt.	Projektteam
D3	GHCN ist kein Akteur im Use Case.	Die API ist der relevante Akteur; GHCN dient lediglich als Datenquelle.	Projektteam
D4	Die Stationsdaten werden beim Start der API geladen und vollständig in einer Datenbank gespeichert.	Vermeidung von Laufzeitabhängigkeiten externer Systeme.	Projektteam

D5	Die Messdaten werden beim Aufruf einer Station geladen, angezeigt und in der Datenbank gespeichert.	Schnelleres wiederholtes Aufrufen von Stationsdaten.	Projektteam
D6	Die maximale Anzahl der angezeigten Wetterstationen ist begrenzt.	Es werden die Top N Stationen nach Distanz sortiert angezeigt (N = 1–10).	Projektteam
D7	Der Standard-Suchradius beträgt 50 km.	Verbesserung der Usability durch sinnvolle Voreinstellung.	Projektteam
D8	Für das Gesamtjahr werden Minimum- und Maximum-Temperaturen farblich hervorgehoben.	Minimum in Blau, Maximum in Rot zur besseren visuellen Unterscheidung.	Projektteam

3) Offene Punkte / ToDos

Nr.	Aufgabe	Verantwortlich	Deadline	Status
A1	Use Cases gemäß Meeting-Ergebnissen überarbeiten	Projektteam	offen	open
A2	Jahresfilter (Start-/Endjahr) in GUI ergänzen	Projektteam	offen	open
A3	Screenshot des aktuellen GUI-Entwurfs erstellen und ggf. nummerieren	Projektteam	offen	open
A4	Überarbeitete Use Cases und GUI-Screenshot an Auftraggeber senden	Projektleitung	offen	open

4) Relevante fachliche & technische Hinweise

4.1 Use Cases

- Der Nutzer löst aktiv eine Stationssuche aus (fehlte bislang im Use Case).
- Vorbedingung: Daten sind über die API verfügbar.

- Fehlerfall: Ist die API nicht erreichbar, muss eine entsprechende Meldung angezeigt werden.
- Ergänzend mögliche fachliche Erweiterung: Niederschlagsmenge (als Variation, nicht zwingend).

4.2 Architektur & Datenhaltung

- Die API fungiert als Akteur.
- Die GHCN-Server werden nicht direkt zur Laufzeit angesprochen.
- Empfehlung: Nutzung von zwei URLs für die Datenquelle.
- Empfehlung: Einsatz von Amazon S3 Buckets zur Erhöhung der Verfügbarkeit.
- Vorteil: Bei Bedarf kann die Datenquelle durch Austausch der URL gewechselt werden.
- Hintergrund: GHCN-Server sind zeitweise nicht erreichbar.

4.3 Qualitätsanforderungen

- Qualitätsanforderungen werden ausschließlich aus Kundenvorgaben abgeleitet.
- Es sollen keine eigenen Anforderungen ergänzt werden.
- Beispiele:
 - Lauffähig unter Windows 11
 - Unterstützung aktueller Versionen von Firefox und Chrome

4.4 GUI / Visualisierung

- Radius einstellbar von 1 bis 100 km in 1-km-Schritten
- Anzahl der Ergebnisse einstellbar von 1 bis 10
- Einheitliche Sprache im gesamten UI (inkl. Latitude / Longitude)
- Standort-Pin wird rot dargestellt
- Beim Start der Anwendung ist keine Vorauswahl gesetzt
- Jahresfilter (Start- und Endjahr) fehlt aktuell und muss ergänzt werden
- Datenlücken werden nicht durch gestrichelte Linien dargestellt (keine Darstellung)
- Bis zu 10 Kurven gleichzeitig darstellbar
- Visualisierung erfolgt mit Chart.js
- Slider-Komponenten stammen aus einer UI-Bibliothek

5) Nächster Termin

Ein Folgetermin wird bei Bedarf nach Überarbeitung der Use Cases und des GUI-Entwurfs vereinbart.

6) Hinweise

Dieses Protokoll gilt als verbindlich, sofern innerhalb von drei Werktagen nach Versand kein Widerspruch erfolgt.