

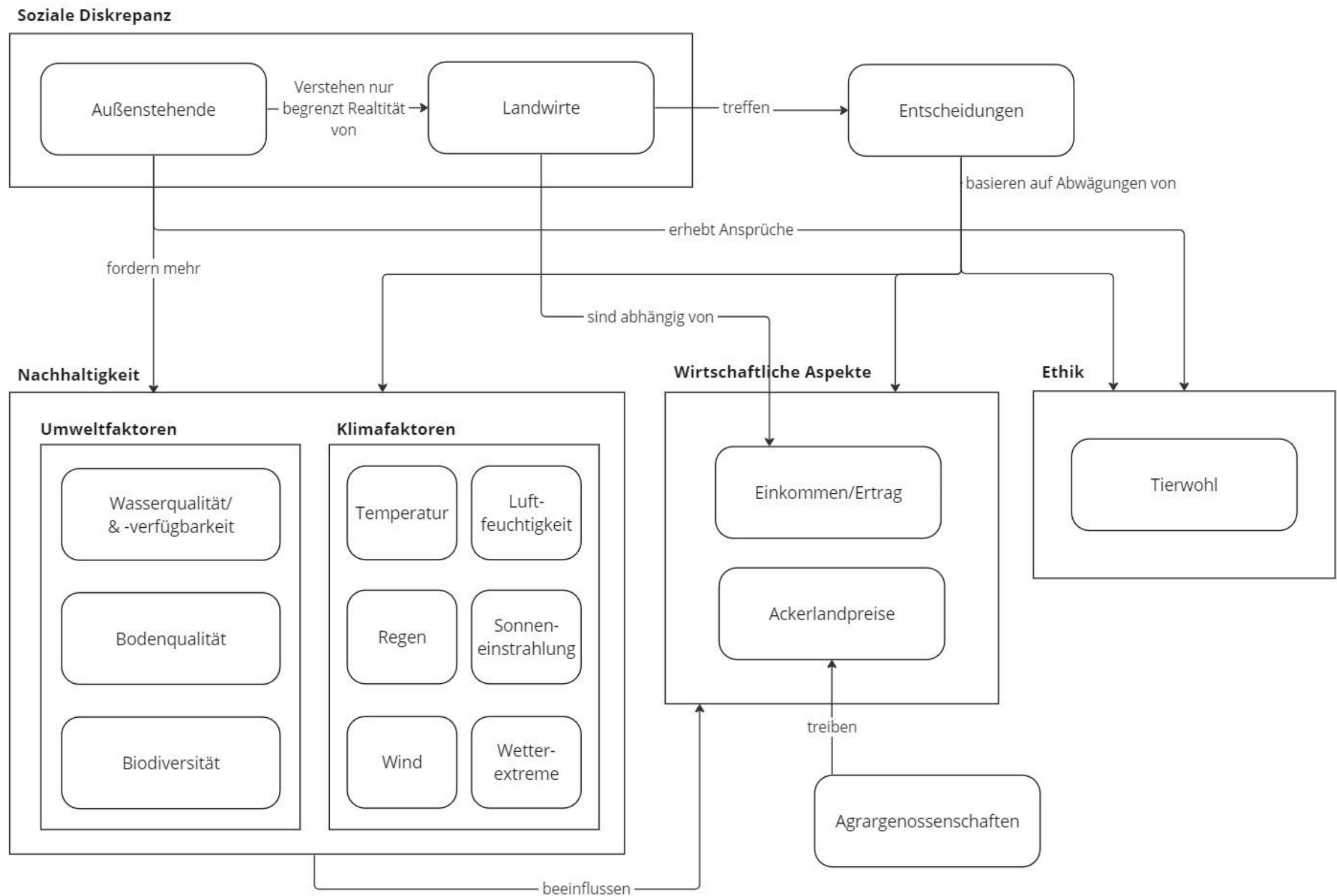
Entwicklungsprojekt: Audit 2

Meike Jungilligens, Mauricio Köppen, Fabian Ngo

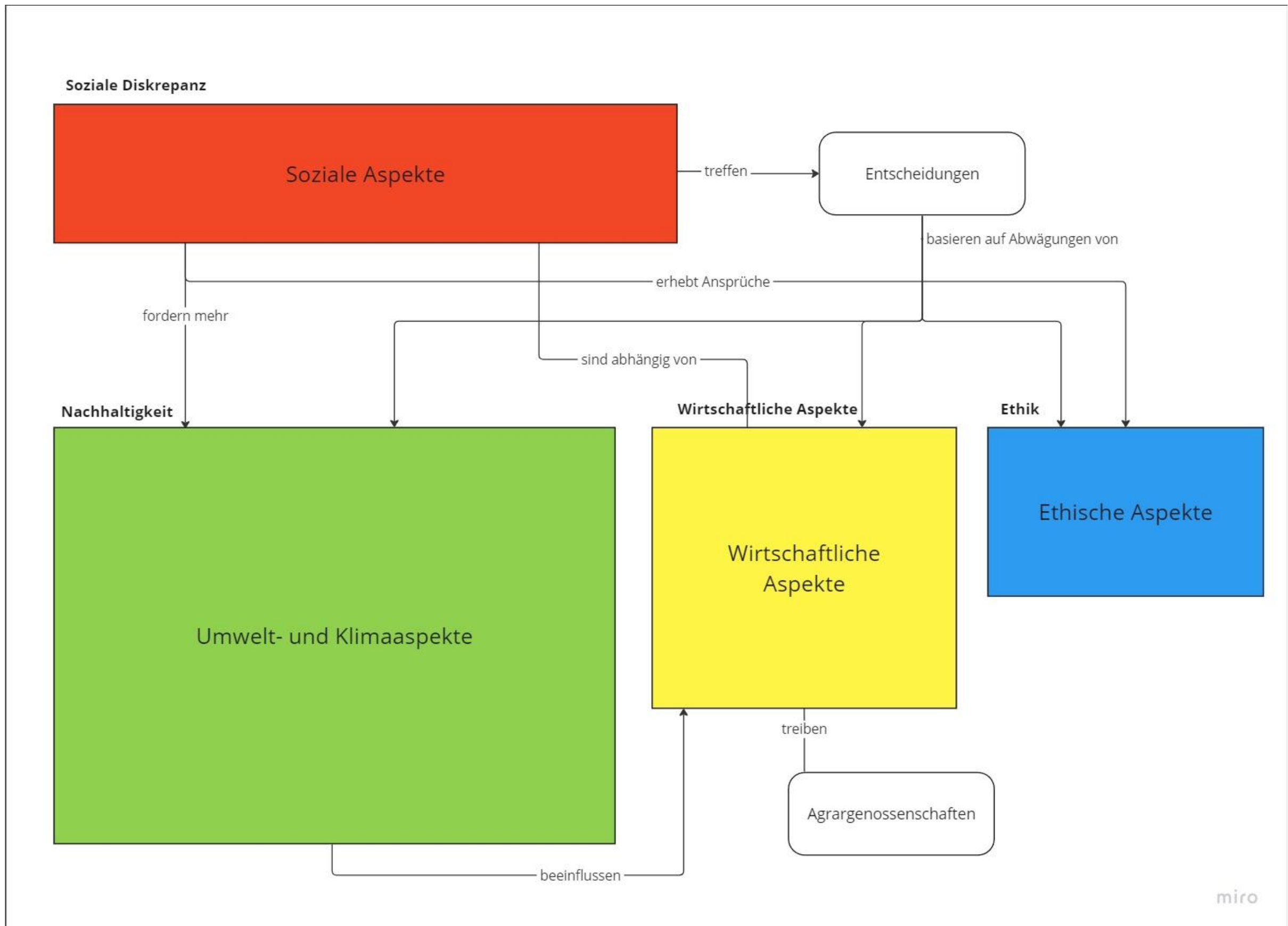
Gliederung

1. **Deskriptives Domänenmodell**
2. **Präskriptives Domänenmodell**
3. **Anforderungen**
4. **Anwendungslogik**
5. **Proof of Concept**
6. **Mock-up**
7. **Festlegen der Artefakte für Audit 3**

Deskriptives Domänenmodell



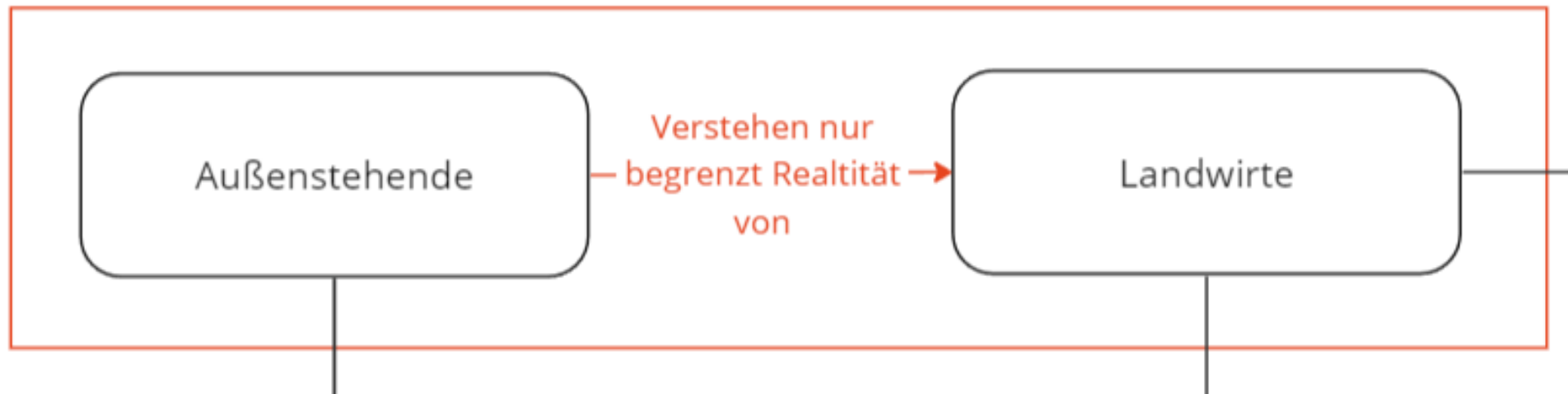
Deskriptives Domänenmodell



Unterschied Deskriptiv / Präskriptives Domänenmodell

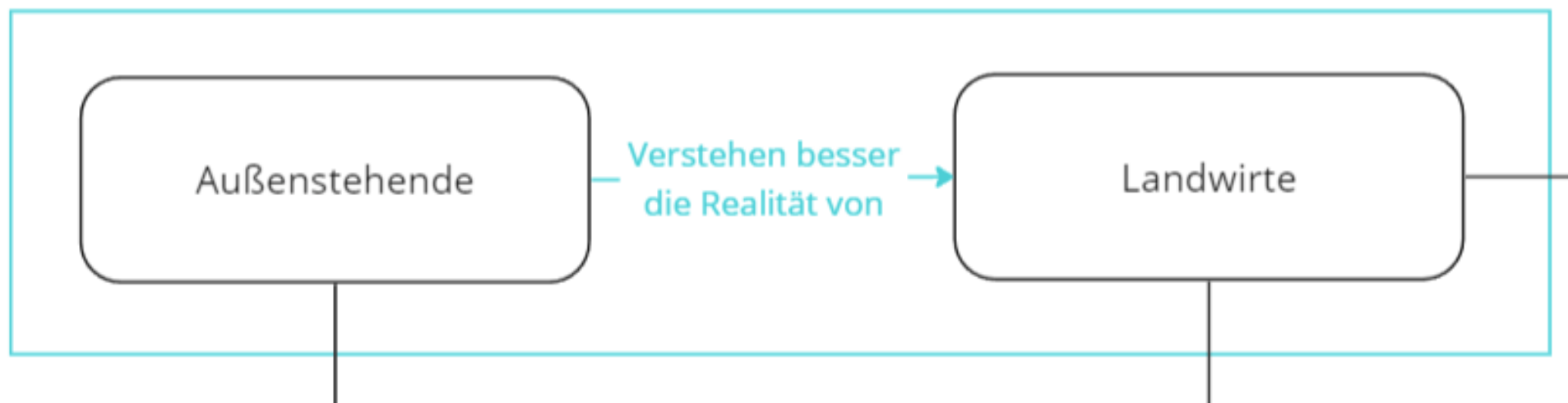
Deskriptiv

Soziale Diskrepanz

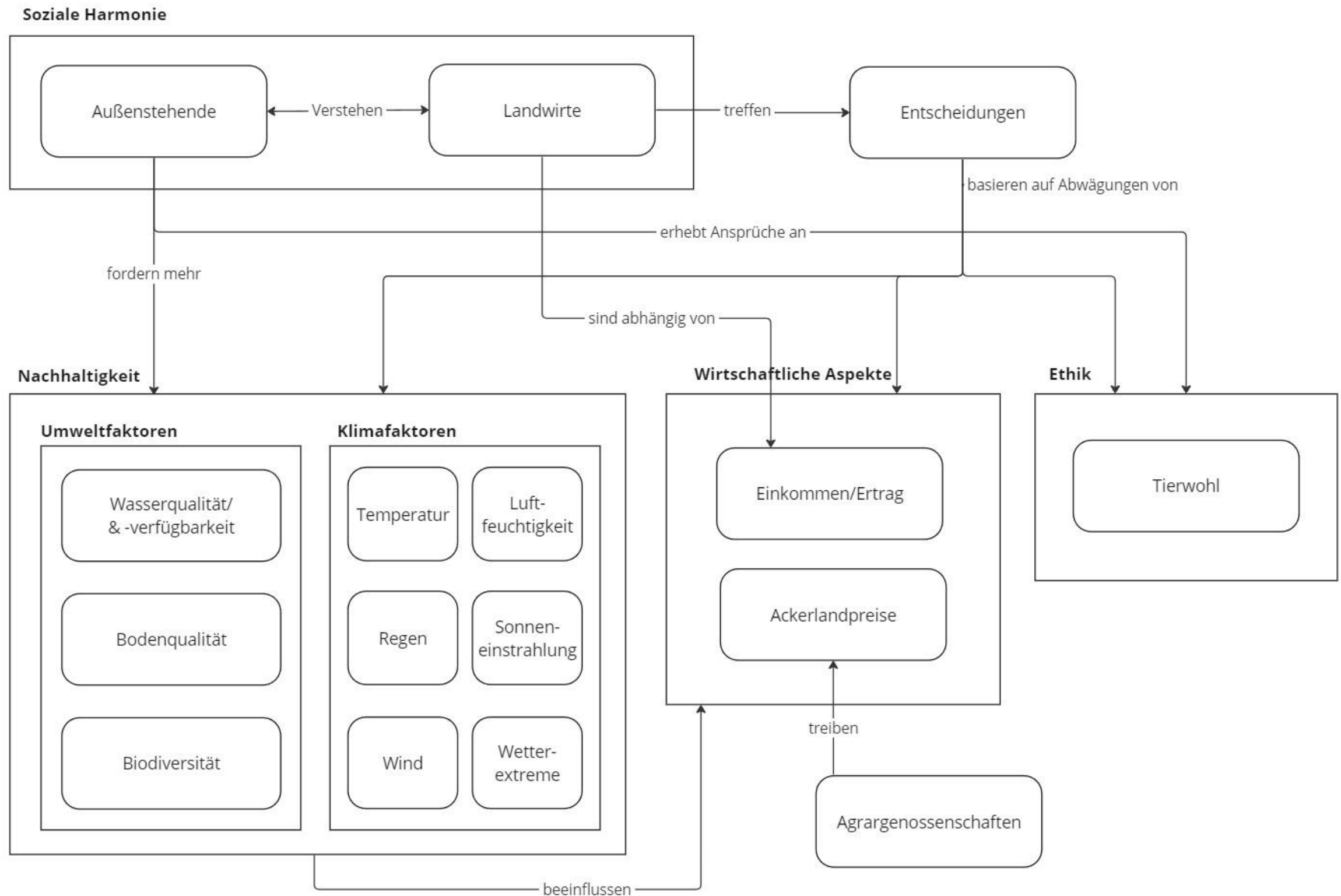


Präskriptiv

Soziale Harmonie



Präskriptives Domänenmodell



Anforderungen

Konzeptionelle Anforderungen

Anforderungen	Muss	Soll	Kann
Entscheidungsstränge mit verschiedenen Ergebnissen	✓		
Konfrontation des Spielers mit abwägungen zwischen Moral und Profit	✓		
Vermittlung der komplexen Herausforderungen der Landwirtschaft aus der Sicht eines Kleinbauers	✓		

Anforderungen an den Code

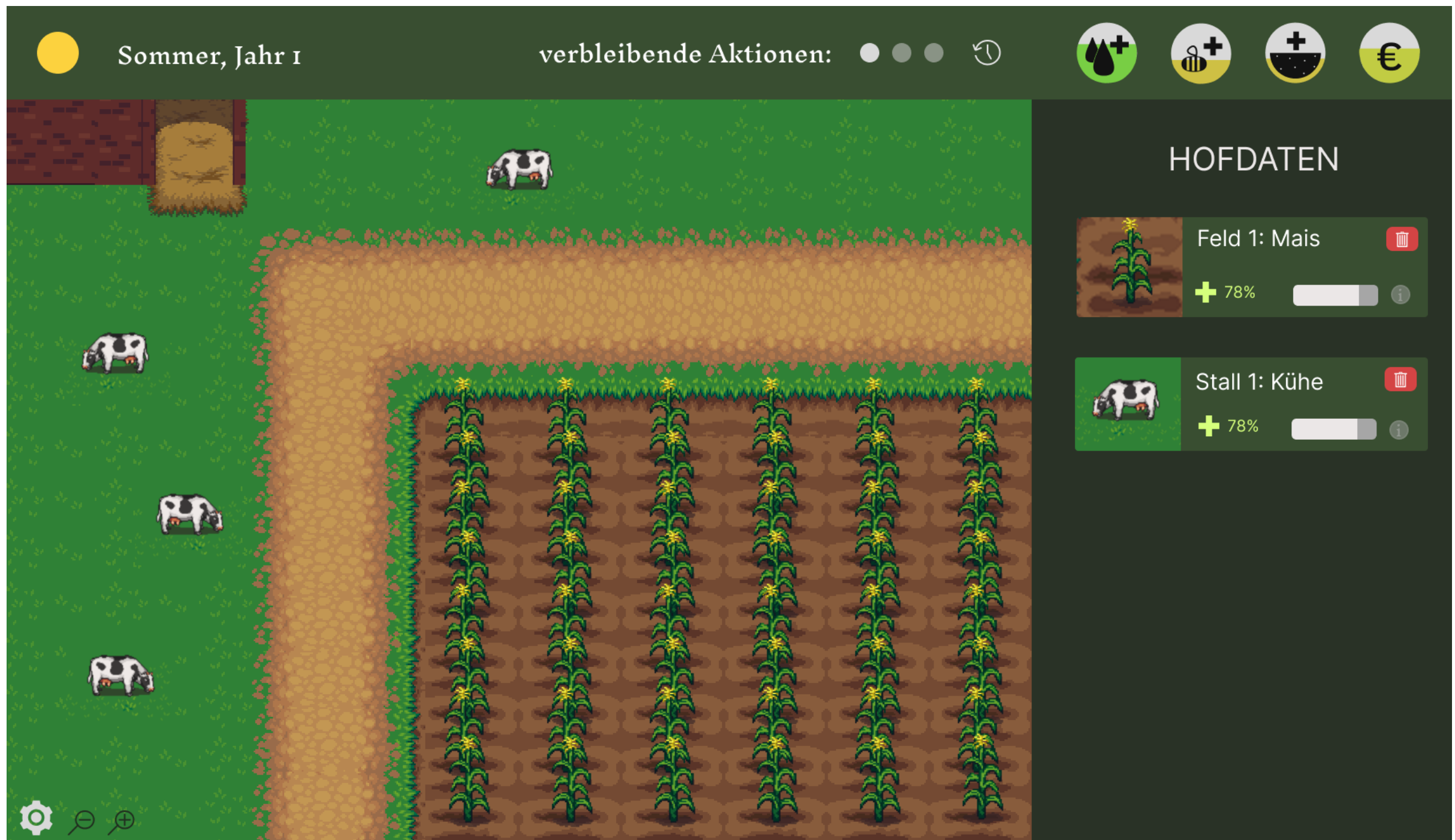
Anforderungen	Muss	Soll	Kann
In Unity Programmieren	✓		
Entscheidungen beeinflussen fortlaufend Zustand der Farm und Parameter	✓		
Zoomen zur Betrachtung der Farm		✓	
Freie Kamerabewegung für den Spieler		✓	
Echtzeit-Anzeige der Entscheidungsfolgen: Echtzeitverbindung Frondend/Backend	✓		
(niedrige/hohe) Parameterwerte können Ereignisse auslösen	✓		

Anforderungen

Anforderungen an das GUI

Anforderungen	Muss	Soll	Kann
4-5 Parameter für wichtige Landwirtschaftsfaktoren	✓		
Sichtbare Veränderungen auf der Farm durch Seasons und Entscheidungen	✓		
Anzeige von Jahr, Season und Anzahl der Aktionen pro Season	✓		
Übersicht über Gesundheit von Tieren und Pflanzen			✓
Jahreszeiten und ihre Auswirkungen auf der Farm sichtbar machen	✓		
Farbliche Darstellung von Parameterveränderungen			✓
Wechsel zwischen verschiedenen Tabs		✓	

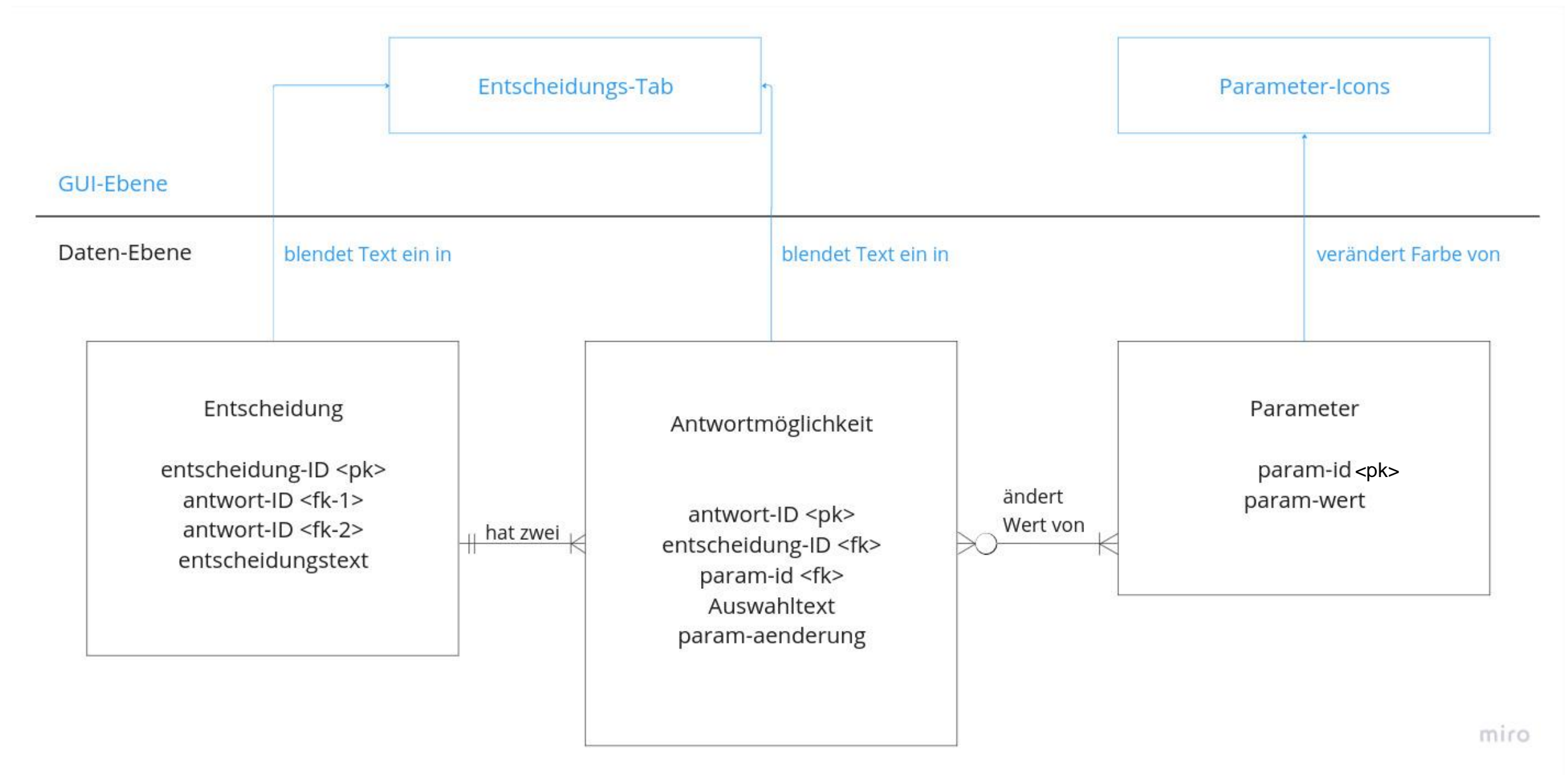
Mock-up



Mock-up



Anwendungslogik - Entscheidungen



Spezifikation von Proof-Of-Concepts

PoC: Entscheidungsmechanik mit visuell erkennbarem Einfluss auf festgelegte Parameter

Implementierung einer interaktiven Entscheidungs-Mechanik, die den Spielern erlaubt, verschiedene Entscheidungsoptionen zu erkunden und die Auswirkungen ihrer Entscheidungen auf den Bauernhof zu verstehen. Jede Entscheidung führt zu einer Steigerung oder Minderung eines oder mehrerer Parameter (Kapital & Umweltfaktoren (Grundwassersauberkeit, Bodenvitalität und Biodiversität)).

EXIT-Kriterium

Dem Spieler wird eine Entscheidung mit zwei Antwortmöglichkeiten im Interface mit einer textuellen Beschreibung angezeigt. Eine der Antwortmöglichkeiten kann ausgewählt werden und senkt oder steigert bestimmte Parameterwerte. Die Änderung ist visuell in Form der Parameter-Icon-Farbe (grün, gelb, rot) und -Füllmenge erkenntlich.

FAIL-Kriterium

Die Entscheidungen beeinflussen die Parameter nicht oder fehlerhaft. Beispiel: Der Verkauf einer Maschine führt zu Geldverlust.

FALLBACK

Überprüfung der programmierten Funktionen für die Berechnung der Parameterwerte (& -änderungen) sowie Sicherstellung einer korrekten Kopplung in der Game Engine zwischen Parameterwert und UI-Anzeige (Parameter-Icon-Farbe, Assets)

Spezifikation von Proof-Of-Concepts

PoC: Persistente Speicherung der Parameterwerte (& evtl. Farm-States)

Dem Spieler ist es möglich, mehrere Entscheidungen hintereinander durchzuführen. Die in vorherigen Entscheidungen geänderten Parameterwerte werden fortlaufend weiter geändert und persistent gespeichert. Dafür wird eine lokale Datenbank (SQLite) genutzt.

EXIT-Kriterium

Die Speicherung nach der erfolgreichen Berechnung (PoC 2) erfolgt über eine korrekte Aktualisierung der Werte in der Datenbank. Auch das Unterbrechen des Spiels/Programms setzt die Parameter nicht auf Default-Werte zurück.

FAIL-Kriterium

Die Parameteränderungen werden gar nicht oder nur flüchtig gespeichert und sind nach Neustart des Programms wieder auf ihre Anfangswerte zurückgesetzt.

FALLBACK

Untersuchung der Verknüpfung zwischen Datenbank und Programmcode. Überprüfung der Richtigkeit der SQL-Befehle. Eventuell Einbinden einer alternativen Datenbank.

Festlegung der Artefakte für Audit 3

- Durchgeführte POC's**
- Prototyp des Spiels**
- Fachspezifische Modellierungsbegründungen**