Analysis of Implementing a Smart Contract in Weather Insurance using Chainlink Oracles

Bachelor Thesis

Submitted in partial fulfillment of the requirements for the degree of Bachelor of Science

Author

Aeberhard Elia Andreas

Gmeindhusplatz 4
5223 Riniken

Matriculation Number: **19-925-957**

Email: elia.aeberhard@uzh.ch

Supervisor

Prof. Dr Claudio J. Tessone

Blockchain & Distributed Ledger Technologies

Department of Informatics

University of Zurich

Date of Submission: [Date]

Executive Summary

Write this last. It is an overview of your whole thesis, and is between 200-300words.. . .

TODO: find todos and do them:)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Weather-related disasters are among the most significant challenges of the 21st century. Due to climate change their frequency and severy continue to increase steadily (Van Aalst 2006). These events not only cause destruction but also result in substantial economic losses for individuals, businesses and governments (Monasterolo 2020). In order to mitigate the financial impact of such disasters, insurance has long been the primary tool by providing a financial safety that increases stability. However, traditional insurance face growing challenges in addressing the scale of modern climate risks.

Traditional weather insurance systems struggle with inefficiencies. Manual claim assessments, systemic risk and the lack of transparency and trust higlight the limitations of these systems. To address these shortcomings innovative solutions are needed. Among them, blockchain technology has emerged as a promising alternative to traditional insurance systems. By leveraging key features such as decentralization, transparency and automation a blockchain-based weather insurance system aims to overcome the inefficiencies of traditional models. (Salem et al. 2021) (Omar et al. 2023)

This thesis proposes a blockchain-based weather insurance system as an alternative to traditional insurance models. The study explores how this approach addresses the limitations of traditional models while improving efficiency, transparency and trust. Furthermore, it examines potential real-world applications and evaluates the feasibility and impact of such a system in practical scenarios.

1.1 Background

In 2016, global disasters accounted for USD 175 billion in economic losses. USD 54 billion of these economic losses were insured, resulting in uninsured losses of USD 121 billion (Swiss Re Institute 2017). These losses highlight the importance of wheather insurance in providing financial protection for individuals, business and governments. For comparison, the international humanitarian assistance reached USD 28 billion in 2015, making the unsinsured losses of 2016 over 4 times that amount (Initiatives 2016).

Predictions support a worsening of these losses due to intensifying climate risks. (Cho 2022) states that insurance premiums are projected to increase by more than 5 percent and property losses from natural disasters could increase by up to 60 percent by 2040. These projections align with (Tucker 1997), who states that the increased probability of extreme weather events will result in increased weather insurance premiums.

1.1.1 Existing Weather Insurance

Traditional weather insurance primarily consisted of crop insurance. The policy in this contract was typically conducted bilateral between an individual or business and the insurance company. If a loss incured on the crop of the individual or business due to weather conditions, an assessment had to be done by the insurance company and the insurance payout was determined based on the specific circumstances (Michler et al. 2022).

Crop insurance has some major problems associated with it. One of these problems is the manual process of analyzing and determining the loss in a monetized amount by an insurance company representative. Other problems include systemic risk, where many insurance holders in the same region are in risk of being affected through weather conditions simultaneously, and asymmetric information, where, for example, insurance holders act more riskily than they normally would because they know they are insured (Makki 2002).

A more modern approach to weather insurance compared to the traditional crop insurance is weather-based index insurance. The key difference here is that weather-based index insurance relies on a measurable variable (such as a temperature drop below a certain threshold or a specific amount of rainfall). The underlying weather data is provided by a reference weather station. The goal is for the criteria (e.g. the temperature threshold) to reflect the financial loss experienced by the insurance holder, for example the loss of a corn field due to adverse weather conditions (Kajwang 2022).

1.1.2 Key limitations in existing weather insurance

Administrative costs

In Traditional crop insurance the administrative costs make up 35% to 40% of the insurance outlays while the remaining portion goes towards other costs such as the insurance payout and the reinsurance costs (Glauber 2004). The majority of these administrative costs consist of loss assessment, monitoring, claims, and underwriting expenses.

In index insurance the administrative costs are significantly lower than in crop insurance because the payouts are based on predefined weather indices rather than assessing individual losses. This index-based insurance model also reduces moral hazards since the payouts are triggered by weather events rather than individual actions. (Kusuma et al. 2018) proposes a weather-based index insurance for rice in Indonesia. It is desgined to be cost-effective by basing the insured amount on the cost of inputs (e.g. seed and fertilizer) rather than covering the individual revenue loss.

In section 1.2 the thesis will discuss how smart contracts can be used to lower administrative costs through automated processes even more than weather-based index insurance.

Systemic risk

A key limitation for existing weather insurance is the systemic risk it poses. (Xu et al. 2010) explains how weather risk is systemic in nature, meaning that weather-related events like droughts or floods often affect entire regions rather than isolated areas.In such an event, a large number of insurance holders would file claims simultaneously, making it difficult for the insurance company to payout all these claims at the same time. It further shows that systemic risk is one of the key reasons why existing weather-based insurance markets have struggled and often require government subsidies, especially in the case of crop insurance.

Based on (Salgueiro and Tarrazon-Rodon 2021), which shows how geographic diversification of the insurance solution reduces the systemic risk, this thesis proposes that a solution based on blockchain technology, which allows for global scalability and diversification, could further reduce systemic risk by creating more decentralized risk pools.

Lack of transparency and trust

The relience of the insurance industry on trust is well established (Courbage and Nicolas 2021). (Guiso 2012) finds that in low trust environments the transaction costs and subsequently the insurance premiums are high. For example, (Gennaioli et al. 2022) found that in countries with a low trust index in the financial markets the amount of rejected claims can rise up to 35% while on average hovering around 20% worldwide.

Due to the immutability and the public nature of the blockchain, it offers foundational trustworthy features. (Hawlitschek et al. 2018) notes however that completely trust-free systems may not be possible since they may still rely on trusted intermediaries. In our system we will use intermediary entites as well, for example Chainlink (see section 1.3).

Fraud and manipulation

Closely related to the subject of transprancy and trust is fraud and manipulation. Since blockchain records are stored across a worldwide network of nodes, changes to past transactions are close to impossible without a consensus from the majority. This significantly lowers the risk of fraudulent alterations and manipulation within the blockchain (Eigelshoven et al. 2021).

1.2 Smart contracts in Insurance

In 2017 the insurance company AXA launched fizzy, a flight insurance based on smart contract technology. It was one of the earliest example of the adaption of blockchain technology in the insurance industry. Fizzy was quite simple. A customer entered his flight details and paid a premium for the insurance. Later Fizzy then used an oracle to check whether the flight has been delayed for more than 2 hours and if so would trigger a payment automatically (Hoffmann 2021).

Despite its simplicity, Fizzy highlighted the potential of blockchain in creating transparent, efficient, and tamper-proof insurance solutions. It demonstrated, how oracles are able to bridge the gap between real-world events and blockchain systems. However, the product also faced challenges, such as regulatory issues and limited adoption. AXA discontinued Fizzy in 2019 but its impact as a pioneering experiment that explores the blockchain's capabilities such as automated payouts, fraud prevention and increased trust between insurers and policyholders remains. (Sedkaoui and Chicha 2021)

Another example is Lemonade, a pioneering company in the InsuranceTech industry. It uses a combination of Artificial Intelligence and Smart Contracts to identify, analayze and streamline claims processes automatically. A customer can buy a policy via Lemonade's app. The smart contract then defines and enforces the terms of the coverage. When an insured event, such as a theft or damage, occurs, the customer can file a claim directly through the app. Lemonade's AI then immediatly reviews the claim, validates the circumstances and triggers the smart contract to execute the payout. This process is seamless and transparent and can be completed within minutes in contrast to the lengthy and often cumbersome claim handling of traditional insurance. (La Barbera 2023) (Tardieu et al. 2020)

1.3 Chainlink and Google Cloud Public Datasets

For the prototype this thesis will use Chainlink, a decentralized oracle service, in order for our smart contract to access real-world weather data. The decentralized nature of Chainlink improves the security and reliability of the data-inputs (Beniiche 2020).

Specifically, Global Surface Summary of the Day (GSOD) wlil be used as the weather data source. Over 9000 station provide weather data for the dataset with each station having to report a minimum of 4 observations per day of a set of measured variables including temperature, wind speed, pressure among others. (Environmental Information 2023). The entire dataset is hosted on the Google Cloud Platform (GCP) and is updated daily.

Additionally, the Global Forecast System (GFS) is used which provides weather data prediction globally of up to 16 days. It is a numerical system based on four different models (atmosphere, ocean, land and sea model) and is hosted on the Google Cloud Platform (GCP) alongside Global Surface Summary of the Day (GSOD) (Environmental Information n.d.).

To integrate the data of these datasets onto our Smart Contract running on the Ethereum blockchain, Chainlink acts as middleware. Its decentralized oracle network fetches and verifies the weather data from GCP, ensuring it is tamper-proof. Once validated, the data is then delivered to the smart contract. This integration between off-chain and on-chain data guarantees the accuracy and reliability of the smart contract operations. (Goswami et al. 2022)

1.4 Technical and regulatory challenges of Smart Contracts

(Gatteschi et al. 2018) expresses concerns that the technology is still being explored and not yet ready for its benefits to become more evident. One challenge faced by smart contracts in the insurance industry is technical readiness. Oracles are the external data sources that provide real-world information to smart contracts. The integrity of the smart contract is dependent on the reliability of these oracles and the validity of their data.

Another important point to consider are the different kinds of legal problems associated with smart contracts. For one there is no accepted universal definition of a smart contract which leads to difficulties in defining their legal status across different jurisdictions. There are also concerns about consumer protection laws since an automated transaction initiated by a smart contract may violate local regulations if consumers are unable to legally contest or reverse it (Ferreira 2021).

1.5 Problem Statement

Even though the weather-based index insurance approach poses a significant improvement compared to the traditional weather insurance, there are still a lot of problems associated with it. These problems include high administrative costs, delayed payouts, scalability and lack of trust in the underlying systems and insurance companies. (Skees et al. 2008). These limitations reduce accessability and the range of weather insurance solutions for individuals and business, especially in more developing areas.

To address these problems, this thesis proposes a weather insurance solution based on blockchain technology. Through decentralized, transparent systems and globally available weather data, the solution aims to reduce the administrative costs, enable automatic and instant payouts, improve the scalability and encourage more trust among the insurance holders.

1.6 Objectives

The main objective of this thesis is to analyze possible implementations of a blockchain-based weather insurance solution that addresses the challenges and drawbacks of traditional crop insurance und weather-based index insurance. The specific objectives are as follows:

- Identify and analyze the key challenges of current weather insurance solutions.
- Propose a blockchain-based weather insurance design that utlizes decentralized oracles and globally available weather data.
- Evaluate the proposed solution.
- Compare the blockchain-based solution to traditional crop insurance and weather-based index solutions to analyze the potential and improvements in efficiency, transparency and user trust.

Chapter 2

Methodology

2.1 Research Design

The research design for this thesis is primarily exploratory, focusing on collecting and synthesizing information from a variety of sources in order to develop a comprehensive understanding of smart contracts, blockchain technology, Chainlink and the Google Cloud Platform. This research process involved extensive review and analysis of academic papers, articles, documentations and credible online resources with the goal of identifying technical possibilities and practical implications of integrating real-world data onto blockchain-based systems.

2.1.1 Exploraty research design

The exploratory research design approach was chosen due to the fact that blockchain technology, which is the basis for the proposed system, is still a new topic and not yet established in mainstream technology and society. The flexible and unstructured approach of exploratory research allows the use of various different information sources. This flexibility is crucial to understand the complex connections and interactions between the different technologies and components. Additionally, the rapid development and technological advancement makes it challenging to create structured, enduring documentation. As a result, several components such as GCP Datasets and Chainlink rely on dynamic, short-lived online technical documentation, which is subject to frequent updates and changes by its authors. (Kişi 2022)

2.1.2 Scope

With the aim of developing a blockchain-based insurance system as an alternative to traditional insurance model the research explores the advantages and opportunities of using blockchain technology compared to traditional methods. The scope encompasses the key limitations of traditional insurance methods (section 1.1.2), an analysis of blockchin technology, smart contracts and decentralized oracles. Specifically the intragration of real-world data through Chainlink and Google Cloud Platform datasets is considered a key aspect. In a later step, the exploratory research approach is used to evaluate the feasibility and practical implications of using a blockchain-based weather insurance system. Key areas in this part include policy management, payout mechanisms and the technical challenges of bridging on-chain and off-chain environments. Premium calculation mechanisms are out of scope for this thesis (todo: further out of scopes?).

2.1.3 Research Goals

The primary goal is to support the design of a blockchain-based weather insurance system (chapter 3) with documentation and literature about the relevant components as a technical foundation and gathering the necessary research to analyze the practical limitations and opportunities for such a system. By addressing the objectives in section 1.6, the research aims to provide a comprehensive framework for the development of a prototype as well as its evaluation and comparison with traditional insurance systems (todo: reference to analysis i guess?).

2.2 Data Collection

In a first step, the economic significance of weather insurance is presented by providing financial costs and losses associated with weather-related disasters. This analysis is based on current reports from reputable international organizations, such as Swiss Re, which regularly publishes data on the economic impact of such events. These reports provide the quantitative basis for emphasizing the need for innovative insurance solutions like those explored in this thesis.

Subsequently, qualitative research is conducted to establish the technical and practical foundation for the development of a blockchain-based weather insurance system as an alternative to the traditional model. This research includes an in-depth review of academic literature, industry white papers and technical documentation related to blockchain technology, smart contracr and their application in insurance systems. This qualitative analysis aims to identify potential advantages of using blockchain in weather insurance, such as increased transparency, trust and reduced administrative costs.

By combining qualitative insights with quantitative data, this thesis establishes a coomprehensive foundation for designing, implementing and evalutating a blockchain-based weather insurance system compared to a traditional system.

2.2.1 Source Selection Criteria

Sources were selected based on their relevance to the topic of interest, such as blockchain technology, decentralized oracles, smart contracts and GCP datasets. Given the rapidly evolving nature of blockchain technology and its components recent and regurarly updates sources such as developer documentation and current reports were included in the research to ensure technical relevance (give examples for reports such as swiss RE etc). Financial aspects of blockchain technology are laid out in more scientifically solid research such as (give examples for solid financial source of blockchain technology).

To provide a well-rounded perspective, the research selection included academic literature and scientific papers on top of the above mentioned industry reports and documentation. In the analysis chapter the diverse range of the selected research is synthesized in order to produce a cohesive and meaningful evaluation.

2.2.2 Specific data types

This subsection focuses on the identification and categorization of specific data types essential for the development and evaluation of the blockchain-based weather insurance system. The data tyoes are divided into three primary categories: meteorological data, technical data and miscalleneous. Each category serves a distinct purpose within the system's design and research.

Meteorological data

This data includes real-time and historical weather information, such as temperature, wind speeds and other climate variables (todo: more aspects). This meteorological data forms the foundation for triggering insurance payouts, making its precision and availability critical to the system's success. The meteorological is primarily sourced from governmental meteorological agencies (todo: include sources) and made available to the smart contract via the use of decentralized oracles.

Technical data

Technical data refers to specific information required to design the blockchain-based weather insurance system and organize its components. This includes information about blockchain technology, smart contracts, Chainlink and GCP. This technical data ensures the system is robust, efficient and fulfills the requirements outlined in (include reference). The primary sources for this data are developer documentations and architecturial blueprints about the components used in the blockchain-based system.

Miscalleneous data

This data encompasses diverse information types that support the broader objectives of the thesis, particularly in identifying limitations in traditional weather insurance systems and emphasizing the need for robust alternatives. Specifically, these include economic data about the relevance of weather insurance systems, such as financial losses from weather-related disasters. It also includes regulatory data to assess the compliance challenges faced by traditional and blockchain based systems. Primary sources for this information are academic literature, research papers and current reports.

2.2.3 Challenges in Data collection

The data collection process for analyzing and designing a blockchain-based weather insurance system faces several challenges, originating from the evolving nature of blockchain technology and the inherent complexity of insurance systems. These challenges span across technical and organizational domains, both of which must be addressed to ensure the validity of the proposed system.

One of the primary challenges is the rapid evolvement of blockchain technology. While it does offer innovative aspects like decentralization, transparency and automation, the technology is still in its early stages, with frequent updates to technical documentations and shifts in standards. Ensuring compatibility between blockchain components, decentralized oracles and real-world weather data requires continous monitoring and adaptation, adding complexity to the data collections process.

Additionally, insurance systems are highly complex, involving numerous components that influence policy design, premium calculation, risk assessment and claim management. Collecting comprehensive research to address these variables requires synthesizing diverse sources. However, these data sources often vary significantly in their format and scope, such as structured datasets, unstructured text, historical or current information. The granularity of this data can differ as well, with some sources providing localized information while others offer only high-level insights, making it challenging to align and analyze the data consistently. Moreover, the reliability of these sources can be inconsistent, with some being prone to errors, omissions or outdated information. Consideration of these variations necessitates the use of data transformation and validation techniques, in order to ensure a cohesive and accurate representation of the overall system.

Another challenge lies in the dynamic and regional nature of weather events. Many regions lack comprehensive weather monitoring infrastructure, leading to data scarcity that prevents the use of a blockchain-based system (todo: expand or remove).

2.3 Key technologies and tools

The development of the blockchain-based weather insurance prototype (chapter 3) required deliberate selection of technologies and tools that align with the objective of developing a blockchain-based protoype that utilizes decentralized oracles and globally available weather data (section 1.6).

2.3.1 Etherum Blockchain Technology

Ethereum was chosen as the underlying blockchain due to its support for decentralized applications (dApps) and smart contracts (todo source). It is widely adopted across both academic and industry settings (todo source). Its ecosystem provides flexible and extensive integration tools, libraries and frameworks, making it a suitable choice for an innovative blockchain-based system (todo source). Ethereum's public nature ensures transparency, which is critical for strengthening trust in insurance systems. While alternatives such as Hyperledger Fabric were considered, Ethereum's widespread adoption and compatibility with decentralized oracles made it the preferred option.

2.3.2 Smart Contracts

Smart contracts form the backbone of the system by automating critical functions such as premium calculation, policy management and payout execution. The rationale for their use lies in their ability to eliminate intermediares, reduce administrative costs and ensure rule-based execution. By coding the insurance logic directly onto the blockchain, smart contracts provide a transparent and immutable way for enforcing policy terms. Their native support on the Ethereum blockchain makes this choice ideal for achieving the systems objectives of efficiency, transparency and trust.

2.3.3 Chainlink

A critical functionality needed for a blockchain-based weather insurance system is bridging the gap between off-chain data and on-chain smart contract execution. The use of traditional centralized APIs were rejected since they represent a single point of failure, which could compromise data integrity and reliability of the system. To achieve the transparency objectives of this thesis a decentralized oracle service was chosen. Chainlink's compatibility with Ethereum simplifies the integration process for a functional system and ensures through its decentralized architecture that the weather data can securely be retrieved and used to trigger the logic embedded in the smart contract.

2.3.4 Google Cloud Public Datasets

Google Cloud Public Datasets were chosen as the supplier of historical and current weather information. These datasets are maintained by reputable organizations and provide data on a large scale.

Google Cloud's access to a vast number of diverse datasets and compatibility with modern data processing tools such as oracles provide a practical and efficient solution. (todo write more)

Chapter 3

Development of the Prototype

3.1 Requirements

In order to provide the blockchain-based weather insurance, our system must fulfill several key requirements. In this section we describe each category of requirements in a dedicated subsection.

3.1.1 Functional Requirements

- **User Interaction**
  - The system must provide a user interface for purchasing policies and triggering payout eligibility checks.
- **Weather Data Integration**
  - The system must be able to receive and process global weather data from reliable and trusted external sources such as Global Surface Summary of the Day (GSOD) and Global Forecast System (GFS)
  - Both historical and forecast data must be available
- **Smart Contract**
  - The smart contract must allow end users to purchase and terminate weather-based insurance policies based on parameters provided by the end user and the external sources.
  - The smart contract must store the policy terms.
  - The smart contract must be able to perform eligibility checks on existing policies.
  - The smart contract must be able to payout funds for policies that have passed eligibility checks.

- **Oracle Integration**
  - The system must use Chainlink oracles to retrieve and verify weather data from GCP datasets.

3.1.2 Non-Functional Requirements

1. **Security**
   - All the bilateral interactions between the smart contract, Chainlink oracles, GCP and the end user must be secure.
2. **Scalability**
   - The system must be able to scale to handle large amounts of policies and end users.
3. **Transparency**
   - All transactions and insurance claims must be recorded on the blockchain for transparency purposes.

3.1.3 Technical Requirements

1. **Blockchain Platform**
   - The smart contract must be deployed on the Ethereum blockchain
2. **Data retrieval**
   - The system must be able to retreive weather data from GCP.
3. **Chainlink Oracle**
   - The system must integrate with a Chainlink node to facilitate data retrieval from GCP.

3.2 Architecture

In this section we will propose the architecture for our blockchain-based weather insurance system. First we present a high-level overview of the system and then elaborate on the most important components in a dedicated subsection for each component.

3.2.1 General Overview

In fig. 3.1 we present an overview of the general architecture of our proposed system. The center of the architecture is the smart contract deployed on the Ethereum blockchain. It manages all policies and handles the payout process. The end user interacts directly with the smart contract through the use of a decentralized app (see section 3.2.2), where they can request insurance policies and eligibility checks.

The weather data is provided by Global Surface Summary of the Day (GSOD) and Global Forecast System (GFS) respectively. These datasets can be accessed via GCP BigQuery and GCP Storage through the GCP API interface. To bridge the gap between the off-chain data from GCP and the on-chain smart contract, the system uses a Chainlink Oracle (see section 3.2.3). This Chainlink Oracle retrieves the weather data from GCP through its API literface and passes it on to the smart contract on the Ethereum blockchain.

This architecture effectively addresses the technical requirements defined in section 3.1.3 and ensures compatible and reliable communications between each component.

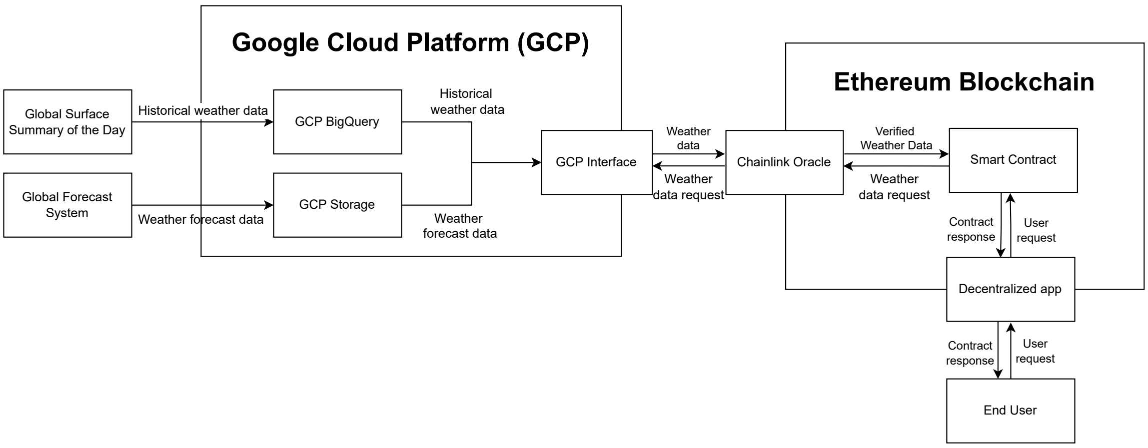

Figure 3.1 – Diagram showing the architecture of the proposed solution. *Source: Author's own representation.*

3.2.2 Decentralized App

The decentralized application (DApp) allows a non-technical end user to directly interact with the blockchain. Through its interface a user can request insurance policies, eligibility checks and receive payout funds. Unlike traditional applications, which interact with a centrally managed backend, decentralized applications interact directly with a blockchain.

This interaction requires a digital wallet (for example MetaMask) which enables the user to initiate and sign a transaction when making a request, such as purchasing an insurance policy. In our architecture, the dApp serves as the primary interface in order for the end-users to interact with the smart contract.

3.2.3 Integration of Chainlink Oracles and the Google Cloud Platform

Since the weather data from GSOD and GFS, which is accessed through the GCP datasets, is not directly available from the on-chain environment of the smart contract we need to leverage oracles, which retreive and verify external data before delivering it to the blockchain.

In our proposed system, Chainlink oracles are used to securely interact with the GCP and pass the data on to the smart contract. The Chainlink oracle network consists of globally distributed nodes. When a request is triggered, multiple nodes independently access GCP and receive the weather data specified in the request. If any nodes return results that differ from the majority, they are flagged as potentially malicious. This decentralized approach ensures that only verified and reliable weather data is passed on to the smart contract.

3.3 Data flow

In this section, we examine the two primary data flow scenarios central to our blockchain-based weather insurance system: the process of purchasing a weather insurance policy (see section 3.3.1) and the process of triggering a policy payout (see section 3.3.2). These scenarios represent the fundamental interactions between the end user, the smart contract and the external data sources. In the subsequent chapters we elaborate on the core functionalities utilized in the two scenarios such as the process of retreiving weather data and calculating the policy conditions.

3.3.1 Purchase Weather Insurance Policy

The sequence diagram in fig. 3.2 outlines the steps involved in purchasing a weather insurance policy in our proposed system. The main components are the end user, the smart contract, Chainlink oracle and the Google Cloud Platform (GCP). Each of these components play a central role in enabling the policy purchase and the associated data retrieval and validation processes. Every request between components includes a set of parameters. These parameters contain the necessary information that each component needs in order to perform its role and functions accurately. For example, in a purchase policy request parameters such as location, coverage duration and type of coverage are sent to the smart contract. Table table 3.1 contains all requests with their respective parameters.

The end user starts the process by requesting a weather insurance policy from the smart contract. This interaction happens through the use of a decentralized app (dApp) section 3.2.2, which is not explicitly mentioned in the diagram but can be thought of as the interface enabling the end user to interact directly with the smart contract. Through the dApp the end user can request a weather insurance policy. In table table 3.1 we find the detailed description of each request. In the policy

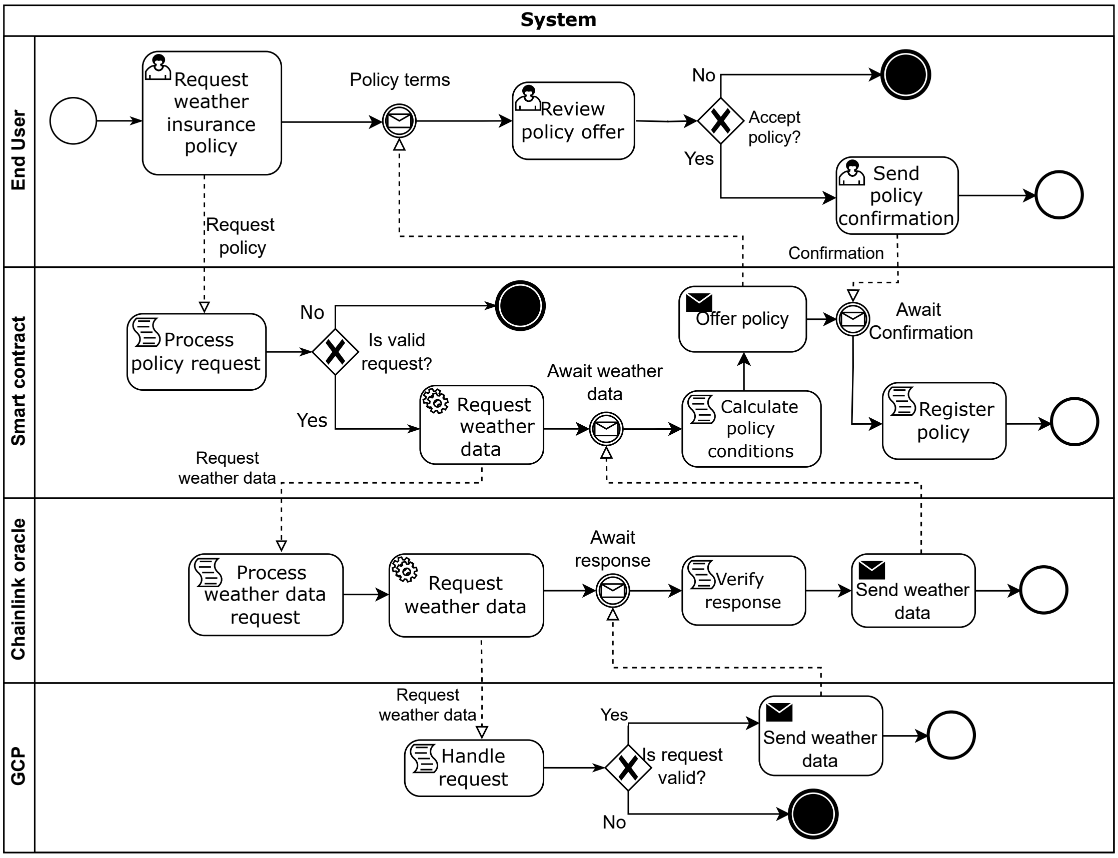

Figure 3.2 – Sequence diagram of purchasing an insurance policy in our proposed system *Source: Author's own representation.*

request the user has to specify the conditions of the policy, these include the location, start- and end-date, the type of coverage (for example drought coverage or storm coverage) as well as the data type (whether the request should return historical or forecast data).

The smart contract then fetches the necessary weather data needed for the calculation of the policy conditions by sending a request to the Chainlink oracle containing the necessary parameters (see "Weather data request" in table table 3.1). The Chainlink oracle propagates this request to the Google Cloud Platform (GCP). Due to the decentralized nature of Chainlink oracles this request is sent multiple times to the GCP from different Chainlink nodes (see section 3.2.3). The responses from each of these requests are then analyzed and verified by the Chainlink network beofre being sent back to the smart contract. The smart contract calculates the policy conditions (such as the premium amount) and sends an offer to the end user. If the end user accepts the terms offered by the smart contract, then the policy is valid and registered on the blockchain.

Note that in the diagram there are two "Request weather data" requests depicted. One is from the smart contract to the Chainlink oracle and one from the Chainlink oracle to the GCP. Even though

Name	Parameters	Description	Response
Request policy	Location: String Coverage start date: Date Coverage end date: Date Type of coverage: Enum	Requests a policy with the given parameters as the underlying conditions	Binding policy offer including financial conditions and a unique policy ID
Request weather data	Location: String Weather start date: Date Weather end date: Date Type of weather data: Enum Frequency: Enum Data Type: Enum	Requests weather data in a specific date range. Examples for weather data types are rainfall, wind speed and temperature. Frequency indicates the data granularity (e.g. hourly, daily). Data Type can be either forecast or historical.	Weather data in JSON format
Confirmation	Policy ID: String Confirmed: Boolean	Sends a confirmation to the smart contract with the specific policy ID	-

Table 3.1 – Requests in fig. 3.2 with their parameters *Source: Author's own representation.*

these requests differ in their technical details (such as type of request and authorization headers) we have combined them in table table 3.1 as one entry. This was chosen specifically for simplicity purposes, since the diagram is an abstraction of a policy purchase process and leaves space for flexibility in technical implementation.

3.3.2 Trigger Policy Payout

The second data flow is shown in fig. 3.3. It presents the process of triggering a policy payout. This payout happens when the specified policy conditions from section 3.3.1 are fulfilled (assuming a policy has been agreed upon between the end user and the smart contract beforehand).

A key decision in the design of the payout process is determining is what triggers an eligibility check and the following payout process. A possible solution would be to use scheduled tasks that periodically trigger an eligibility request on every policy held by the smart contract. Since the Ethereum blockchain does not support scheduled or automated tasks natively, such a solution would require an external solution. For example Chainlink offers an automation service (Chainlink 2024) with which it would be possible to trigger a scheduled eligibility check on the smart contract (for example once a day). This service would however cost gas which has to be paid by the smart contract and result in a lot of unnecessary transactions. In our proposed solution, the end user initiates the eligibility check instead. This decision prioritizes simplicity and cost-efficiency, avoiding unnecessary transactions that would otherwise increase the policy premium for the end user.

After the end user has initiated the eligibility check through the use of a decentralized App (dApp), the smart contract gathers the weather data for the specific location and covered duration of the policy. This process of retrieving the relevant weather data is similar to the one in section 3.3.1, with a key distinciton being that the weather data contains past data and not forecast data. Once the weather data is received, the smart contract evaluates whether the conditions for a payout are met and communicates the outcome to the end user. If the end user is eligible for a payout, they can submit a request to the smart contract, which will then transfer the agreed-upon funds as specified in the policy.

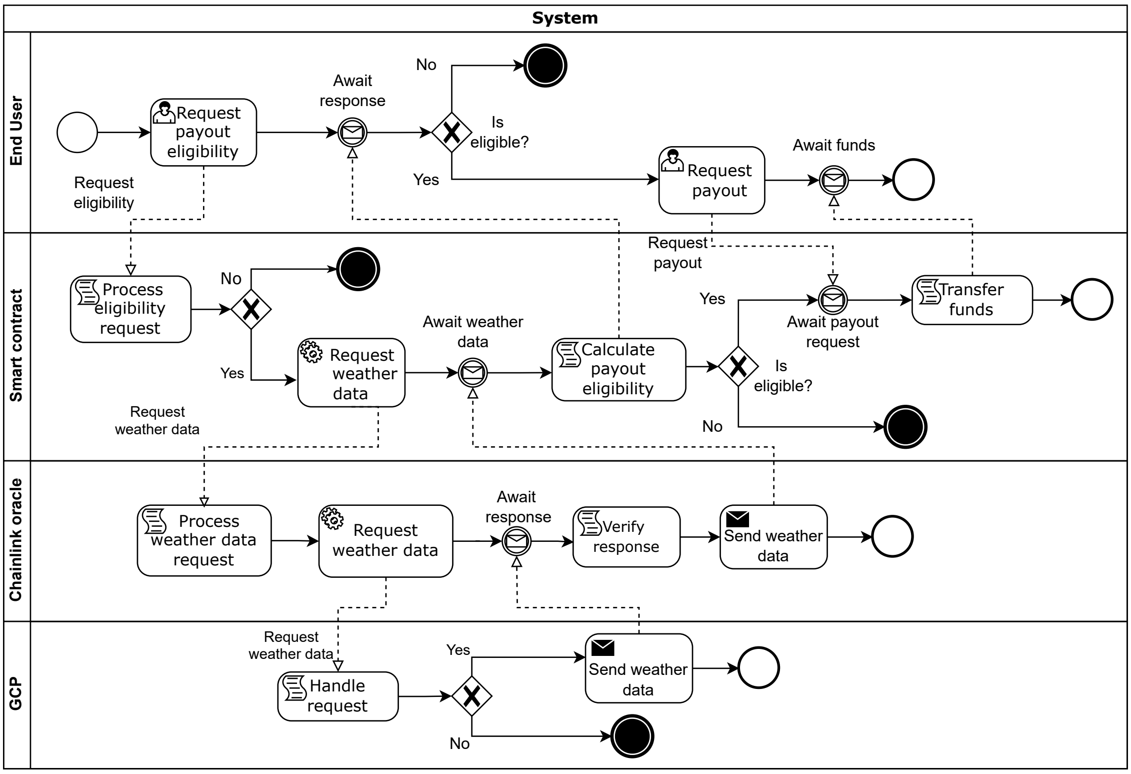

Figure 3.3 – Sequence diagram of triggering a policy payout *Source: Author's own representation.*

3.3.3 Interaction with the Smart Contract

As mentioned in section 3.3.1 and section 3.3.2, the end user communicates with the smart contract through a decentralized app (see section 3.2.2). The dApp interface is indistinguishable from a regular app interface, with the difference being that it interacts directly with the blockchain instead of a traditional backend. Within the dApp the end user can create the requests and enter the relevant parameters. Once the user submits a request, the dApp creates a transaction that is signed using the end user's digital wallet. This transaction is then broadcasted to the blockchain, where it can

Name	Parameters	Description	Response
Request eligibility	Policy ID: String	Requests an eligibility check which is then performed by the Smart Contract	Whether the policy is eligible for a payout or not in JSON format
Request weather data	Location: String Weather start date: Date Weather end date: Date Type of weather data: Enum Frequency: Enum Data Type: Enum	Requests weather data in a specific date range. Examples for weather data types are rainfall, wind speed and temperature. Frequency indicates the data granularity (e.g. hourly, daily). Data Type can be either forecast or historical.	Weather data in JSON format
Request payout	Policy ID: String	Requests the payout of funds	Amount of funds specified in the policy

Table 3.2 – Requests fig. 3.3 with their respective parameters *Source: Author's own representation.*

interact directly with the smart contract.

As the focus of this prototype is primarily on the general architecture of a blockchain-based weather insurance solution and its smart contract functionality, we will not delve further into the specifics of decentralized app (dApp) development, since it is not necessary for the subsequent analysis.

3.3.4 Calculating Policy Conditions

One of the core functions of the smart contract is calculating the policy conditions. In fig. 3.2 it is presented as an internal process of the smart contract. This process involves assessing the parameters provided by the end user in the ”Request policy” request (see table 3.1) as well as the weather data that the smart contract receives from GCP through the ”Request weather data” request (see table 3.1).

In a first validation step, the smart contract ensures that all the parameters are valid, for example that the location is within a supported region and that the coverage is within the allowed limits. Latter is defined through the furtherst available forecast data, which in our solution is 16 days (Environmental Prediction (NCEP) 2016). Next follows the calculation for the policy conditions based on these parameters. These conditions are the following:

- Premium amount (the cost of the policy that is paid by the end user)
- Trigger conditions (specific thresholds or events that must occur for a payout to be issued, for

example rainfall above a certain level)

- Coverage limit (the maximum payout amount available under the policy in the case where the conditions for a payout are met)

Note that the coverage limit is defined by the end used in the "Request policy" request (see table 3.1) but still has to be validated and included in the policy by the smart contract since there may be limits for a maximum coverage amount due to the total funds held by the smart contract.

Calculating the premium amount is the most critical part in the policy conditions. It defines the balance between risk and reward for both the end user and the smart contract. The premium is determined using a formula that considers the following key factors:

- Coverage duration
- Location
- Type of coverage
- Forecast weather data
- The contract margin

This formula represents a fundamental approach for premium calculation. However, it can be further refined by including historical data and actuarial models, allowing for a more dynamic calculation based on real-world risk. Developing a fully-realized mathematical formula suitable for the usage in a production environment is beyond the scope of this thesis. Instead, we focus on identifying and outlining the key factors that are included in the formula. In the analysis chapter (include reference), these factors will be evaluated to compare the premium calculation approach in a blockchain-based insurance system with that of a traditional weather insurance model.

3.3.5 Smart contract fetching weather data

Both in fig. 3.2 and in fig. 3.3 the retrieval of weather data from GCP plays a central role. The security and transparency of our blockchain-based system is only as strong as each component involved in the data retrieval chain. By using Chainlink oracle (include reference), the smart contract can maintain its decentralized nature.

3.4 Real-world application of the prototype

todo: In this section we will ...

3.4.1 Application Scenarios

The following subsections outline three potential real-world applications for our proposed system, presenting how our system's advantages can be effectively leveraged to address specific challenges in each use-case.

Disaster relief efforts

Certain regions and countries are particularly vulnerable to extreme weather events and catastrophes, such as floods, hurricanes or droughts. As descibed in Chapter 1. these disasters can result in significant loss of life and extensive financial damage. Relief efforts for such catastrophes are often concentrated in specific areas and rely heavily on government and aid organization support.

By offering transparent weather insurance to governments and humanitarian organizations, the financial burden of a disaster relief can be better managed and distributed. As these organizations and governments are often funded through taxpayer money, the proposed application ensures that payouts are automated and verifiable to the general public through blockchain technology, enhancing transparency, trust and accountability in the use of such funds.

Tourism industry

The tourism industry can be very sensitive to weather conditions, particularly in regions prone to extreme or unexpected weather events, such as heavy rainfall, storms or prolonged droughts. A traveler visiting a region with a high risk of heavy rainfall might want to complete a short-term insurance to protect their trip costs in case of such a weather event.

In this application, the system's ability to complete quick policy insurance and automated payouts ensures that the customer can obtain a weather insurance on short notice without much bureaucracy involved. This makes the system suitable for spontaneous decisions, which are very common in the tourism sector. Unlike the disaster relief application section 3.4.1, which focuses on large-scale policies for a few big policy holders, this application is tailored to a smaller-scale, high-volume model.

Renewable Energy Sector

Renewable energy projects such as wind farms, solar parks and hydroelectric plants are highly dependent on favorable weather conditions. Bad weather conditions, such as long cloud cover, insufficient wind speeds or droughts can significantly reduce enegery output, which can impact the financial stability and operational efficiency of such projects.

The proposed blockchain-based weather insurance system offers a practical solution to mitigate these risks. Through automated payouts when predefined conditions are met, such as insufficient wind speeds or weak sunlight, the system provides project operators with quick financial compensation.

Similar to the disaster relief efforts application section 3.4.1, big energy projects are often funded by governments. The transparent nature of our blockchain-based system allows the general public to oversee how funds are allocated, used and distributed. This enhances accountability, builds public trust and ensures the funds are managed efficiently and fair.

3.4.2 Scalability and Global Reach

The decentralized architecture and reliance on smart contracts enables the proposed system to scale efficiently and is able to handle high volumes of users and policies without compromising performance or reliability. It reaches global applicability through globally available networks such as Chainlink and widely abailable datasets from GCP. Unlike traditional insurance systems, which may face bottlenecks due to centrally managed infrastructure this system leverages blockchain technology to streamline policy management, claims processing and payouts in a decentralized manner. As a result, a growing number of users and transaction can be handles without significant delays or resource constraints.

Moreover, the use of smart contacts avoids the need for intermediaries while the system grows, remaining cost-efficient in such a scenario. This flexibility allows not only for a wide range geographically, but also a wide range of industries and target groups, from single travelers to entire governments.

3.4.3 Technological and regulatory barriers of the prototype

//optional

Chapter 4

Analysis and Discussion

Idea: Show flexibility of system by marking how different use cases are support in real world application chapter, such as B2B in disaster relief efforts and B2C in tourism industry

4.1 Analysis of the Prototype

4.1.1 Functional Analysis

In this section we evaluate how well the blockchain-based weather insurance prototype fulfills the functional requirenments outline in section 3.1.1. Each core function is evaluated with the specified requirenments.

User Interaction

The protoype successfully provides a decentralized application (dApp) as the user interface, enabling the end user to interact directly with the system. Through the dApp the user can send request for weather insurance policies as well as insurance payouts directly to the smart contract. This interaction is facilitated using a digital wallet (e.g. MetaMask), which allows the user to sign in and broadcast the transaction to the Ethereum blockchain. The user experience is kept to the bare minimum to provide an easy-to-use system, especially for non-technical users.

Weather Data Integration

In sections section 3.3.1 and section 3.3.2 we outline how the smart contract is able to receive the relevant weather data from GCP through Chainlink. Through two seperate endpoints on the side of GCP, the platform which hosts Global Surface Summary of the Day (GSOD) and Global Forecast System (GFS), Chainlink is able to access both historical and forecast weather data, making it available to the smart contact. This integration ensures a decentralized access to GCP, enabling the system to meet the functional requirement of securely retreiving external weather data.

Smart Contract

The function of the smart contract can best seen in fig. 3.3 and fig. 3.2. As the heart of the system, the smart contracts interacts directly and indirectly with the other components, synthesizing the system to be functional. In ... todo: show successful functionality of smart coontract

Oracle Integration

todo: remove this from functional requirements (chapter 3) and merge it with weather data integration

4.1.2 Non-functional Analysis

4.2 Comparative Analysis

4.3 Discussion of Key findings

4.4 Analysis of smart contracts in the insurance industry

//optional

Chapter 5

Summary and Conclusion

5.1 Summary of findings

5.2 Conclusions

5.3 Future work

Appendices

Appendix A

Appendix title 1

Test appendix 1

Bibliography

Beniiche, Abdeljalil (2020). "A study of blockchain oracles". In: *arXiv preprint arXiv:2004.07140*.

Chainlink (2024). *Chainlink Automation*. Accessed: 2024-11-05.

Cho, R (2022). "With Climate Impacts Growing, Insurance Companies Face Big Challenges". In: *State Of The Planet: Columbia Climate School* 3.

Courbage, Christophe and Christina Nicolas (2021). "Trust in insurance: The importance of experiences". In: *Journal of risk and insurance* 88.2, pp. 263–291.

Eigelshoven, Felix, Andre Ullrich, and Douglas Parry (2021). "Cryptocurrency Market Manipulation-A Systematic Literature Review." In: *ICIS*.

Environmental Information, NOAA National Centers for (2023). *Global Surface Summary of the Day - GSOD*. Accessed: 2024-10-21.

— (n.d.). *Global Forecast System (GFS)*. Accessed: 2024-10-21.

Environmental Prediction (NCEP), National Centers for (2016). *Global Forecast System (GFS) - Global Spectral Model (GSM) V13.0.2*. https://vlab.noaa.gov/web/gfs/documentation. Accessed: 2024-11-06.

Ferreira, Agata (2021). "Regulating smart contracts: Legal revolution or simply evolution?" In: *Telecommunications Policy* 45.2, p. 102081.

Gatteschi, Valentina, Fabrizio Lamberti, Claudio Demartini, Chiara Pranteda, and Víctor Santamaría (2018). "Blockchain and smart contracts for insurance: Is the technology mature enough?" In: *Future internet* 10.2, p. 20.

Gennaioli, Nicola, Rafael La Porta, Florencio Lopez-de-Silanes, and Andrei Shleifer (2022). "Trust and insurance contracts". In: *The Review of Financial Studies* 35.12, pp. 5287–5333.

Glauber, Joseph W (2004). "Crop insurance reconsidered". In: *American Journal of Agricultural Economics* 86.5, pp. 1179–1195.

Goswami, Subhasish, Syed Muhammad Danish, and Kaiwen Zhang (2022). "Towards a middleware design for efficient blockchain oracles selection". In: *2022 Fourth international conference on blockchain computing and applications (BCCA)*. IEEE, pp. 55–62.

Guiso, Luigi (2012). "Trust and insurance markets 1". In: *Economic Notes* 41.1-2, pp. 1–26.

Hawlitschek, Florian, Benedikt Notheisen, and Timm Teubner (2018). "The limits of trust-free systems: A literature review on blockchain technology and trust in the sharing economy". In: *Electronic commerce research and applications* 29, pp. 50–63.

Hoffmann, Christian Hugo (2021). “A double design-science perspective of entrepreneurship–the example of smart contracts in the insurance market”. In: *Journal of Work-Applied Management* 13.1, pp. 69–87.

Initiatives, Development (2016). *Global Humanitarian Assistance Report 2016*. Accessed: October 1, 2024. Development Initiatives.

Kajwang, Ben (2022). “Weather based index insurance and its role in agricultural production”. In: *International Journal of Agriculture* 7.1, pp. 13–25.

Kişi, Nermin (2022). “Exploratory research on the use of blockchain technology in recruitment”. In: *Sustainability* 14.16, p. 10098.

Kusuma, Aditya, Bethanna Jackson, and Ilan Noy (2018). “A viable and cost-effective weather index insurance for rice in Indonesia”. In: *The Geneva Risk and Insurance Review* 43, pp. 186–218.

La Barbera, Salvatore (2023). “Insurtech Revolution in the Insurance Sector: A Comprehensive Review of the Transformational Impact and the Lemonade Case Study”. In.

Makki, Shiva (2002). *Crop insurance: inherent problems and innovative solutions*. Ames: Iowa State University Press.

Michler, Jeffrey D, Frederi G Viens, and Gerald E Shively (2022). “Risk, crop yields, and weather index insurance in village India”. In: *Journal of the Agricultural and Applied Economics Association* 1.1, pp. 61–81.

Monasterolo, Irene (2020). “Climate change and the financial system”. In: *Annual Review of Resource Economics* 12.1, pp. 299–320.

Omar, Ilhaam A., Raja Jayaraman, Khaled Salah, Haya R. Hasan, Jiju Antony, and Mohammed A. Omar (2023). “Blockchain-Based Approach for Crop Index Insurance in Agricultural Supply Chain”. In: *IEEE Access* 11, pp. 118660–118675. DOI: 10.1109/ACCESS.2023.3327286.

Salem, Mukuan Junior, Fajar Henri Eranmus Ndolu, Diar Eka Risqi Hidayatullah, and R. F. Sari (2021). “Developing NEO Smart Contract for Weather-Based Insurance”. In: *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, pp. 603–608. DOI: 10.1109/ISRITI54043.2021.9702853.

Salgueiro, Andrea Martinez and Maria-Antonia Tarrazon-Rodon (2021). “Is diversification effective in reducing the systemic risk implied by a market for weather index-based insurance in Spain?” In: *International Journal of Disaster Risk Reduction* 62, p. 102345.

Sedkaoui, Soraya and Nawal Chicha (2021). “Blockchain-based smart contract technology application in the insurance industry: The case of “Fizzy””. In: *MJBS,(2)*.

Skees, Jerry R et al. (2008). “Challenges for use of index-based weather insurance in lower income countries”. In: *Agricultural Finance Review* 68.1, p. 197.

Swiss Re Institute (2017). *Natural catastrophes and man-made disasters in 2016*. Sigma No. 2/2017. Accessed: October 1, 2024. Swiss Re.

Tardieu, Hubert, David Daly, José Esteban-Lauzán, John Hall, George Miller, Hubert Tardieu, David Daly, José Esteban-Lauzán, John Hall, and George Miller (2020). “Case study 4: The digital transformation of insurance”. In: *Deliberately Digital: Rewriting Enterprise DNA for Enduring Success*, pp. 255–264.

Tucker, Michael (1997). “Climate change and the insurance industry: the cost of increased risk and the impetus for action”. In: *Ecological Economics* 22.2, pp. 85–96.

Van Aalst, Maarten K (2006). “The impacts of climate change on the risk of natural disasters”. In: *Disasters* 30.1, pp. 5–18.

Xu, Wei, Guenther Filler, Martin Odening, and Ostap Okhrin (2010). “On the systemic nature of weather risk”. In: *Agricultural Finance Review* 70.2, pp. 267–284.

Eidesstattliche Erklärung

Der/Die Verfasser/in erklärt an Eides statt, dass er/sie die vorliegende Arbeit selbständig, ohne fremde Hilfe und ohne Benutzung anderer als die angegebenen Hilfsmittel angefertigt hat. Die aus fremden Quellen (einschliesslich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind ausnahmslos als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

. . . .
Ort, Datum

. . . .
Unterschrift des/der Verfassers/in