



# DOKUMENTATION ZUM SIMULATIONSPROJEKT

Waldbrand-Simulation

Abgabe: 06.02.2026

Dominik Hinze

FIN 31

# Inhaltsverzeichnis

1.	Einleitung .....	1
1.1.	Projektumfeld .....	1
1.2.	Projektziel .....	1
1.3.	Projektbegründung .....	1
1.4.	Projektschnittstellen .....	1
2.	Projektplanung .....	2
2.1.	Projektphasen .....	2
2.2.	Ressourcenplanung .....	2
3.	Analysephase .....	3
3.1.	Ist-Analyse .....	3
3.2.	Projektkosten .....	3
3.3.	Anwendungsfälle .....	3
3.4.	Risikoanalyse .....	3
3.5.	Lastenheft .....	4
4.	Entwurfsphase .....	4
4.1.	Technologieauswahl .....	4
4.2.	Architekturdesign .....	4
4.3.	Entwurf der Benutzeroberfläche .....	5
4.4.	Pflichtenheft .....	5
5.	Implementierungsphase .....	6
5.1.	Entwicklung der Benutzeroberfläche .....	6
5.2.	Entwicklung der Geschäftslogik .....	7
6.	Projektabschluss .....	8
6.1.	Testen der Anwendung .....	8
6.2.	Dokumentation .....	8
6.3.	Soll-/Ist-Vergleich .....	8
6.4.	Ausblick .....	8
7.	Anhang .....	9
A.1	Zeitplanung .....	9
A.2	Verwendete Ressourcen .....	9
A.3	Use-Case-Diagramm .....	10
A.4	Klassendiagramm (Auszug) .....	10
A.5	Lastenheft (Auszug) .....	11
A.6	Mockups .....	1
A.7	Pflichtenheft (Auszug) .....	2
A.8	Bilder der Anwendung .....	4
A.9	Testprotokoll .....	6
A.10	Benutzerhandbuch .....	9

# 1. Einleitung

## 1.1. Projektumfeld

Der Auftrag für dieses Projekt wurde im Rahmen des Unterrichts für die Ausbildung Fachinformatiker für Anwendungsentwicklung an der Berufsschule „Berufliche Schule der Hanse- und Universitätsstadt Rostock -Technik-“ erteilt. Die Umsetzung erfolgte begleitend zu den Lernfelder 10, 11 und 12, welche im dritten Lehrjahr unterrichtet werden.

Die Umsetzung des Projektes erfolgte im Homeoffice und den Räumlichkeiten der Berufsschule und des Ausbildungsbetriebes Gecko mbH.

## 1.2. Projektziel

Das Ziel dieses Projektes ist die Entwicklung einer Simulation. Eine Simulation zeichnet sich durch ein unvorhersehbares Ergebnis aus. In diesem Projekt wird die Simulation durch Eingabeparameter und Zufallsfaktoren beeinflusst werden.

Das Thema der Simulation ist freigestellt. Der Fokus liegt auf der Untersuchung, wie sich unterschiedliche Umwelt- und Wetterbedingungen auf das Brandverhalten auswirken. Parameter wie Windrichtung, Windstärke, Brandwahrscheinlichkeit und Walddichte können vom Benutzer angepasst werden und beeinflussen den Verlauf der Simulation. Die Simulation wird als Windows- Anwendung mit WPF umgesetzt.

## 1.3. Projektbegründung

Dieses Projekt dient dem Ziel der Vorbereitung für das IHK-Abschlussprojekt. Die Auszubildenden haben mit diesem Projekt die Möglichkeit die eigenständige Durchführung eines Projektes zu proben. Aspekte wie Zeitmanagement und selbstständiges Arbeitsaufteilung können so trainiert werden.

Die Simulation eines Waldbrandes kann mehreren Zwecken dienen. Sie kann als interaktives Lern- und Demonstrationsmedium dienen, um das Interesse an natürlichen Prozessen zu fördern. Darüber hinaus kann eine vereinfachte Waldbrandsimulation in der Forstwirtschaft sowie in der Umweltplanung eingesetzt werden, um theoretisch zu untersuchen, wie sich unterschiedliche Faktoren auf die Ausbreitung eines Brandes auswirken können.

## 1.4. Projektschnittstellen

Die Umsetzung des Projektes erfolgt an mehreren Standorten, aus diesem Grund wurde sich dafür entschieden den Quellcode mit dem Verwaltungstool git auf der Plattform GitHub zu speichern. Daraus ergibt sich die Möglichkeit von verschiedenen Geräten aus an dem Projekt zu arbeiten, außerdem ist der Quellcode so vor Verlust geschützt. Die Plattform wurde darüber hinaus für die Erstellung von Tickets genutzt für einen Überblick über erledigte und offene Aufgaben. Die fertige Simulation wird eine alleinstehende Windows-Anwendung. Sie wird keine externen Programme zur Ausführung benötigen.

## 2. Projektplanung

### 2.1. Projektphasen

Das Projekt wurde vom 04.09.2023 bis zum 06.02.2026 durchgeführt. Der Ablauf orientierte sich am Wasserfallmodell und lässt sich in sechs Phasen unterteilen: Planungsphase, Analysephase, Entwurfsphase, Implementierungsphase, Dokumentation und Übergabe des Projektes. Jede Phase bestand aus mehreren Teilaufgaben. Ein detaillierter Überblick über die Zeitplanung befindet sich im Anhang [A.1 Zeitplanung](#).

### 2.2. Ressourcenplanung

Die Umsetzung eines Projektes erfordert in der Regel erhebliche Ressourcen. Eine vollständige Liste der benötigten Ressourcen ist in im Anhang [A.2 Verwendete Ressourcen](#) zu finden. Diese beinhaltet sowohl die verwendete Hard- und Software als auch das beteiligte Personal. Für die Entwicklung der Anwendung wurden ein Firmenlaptop und ein ausgestatteter Büroarbeitsplatz vom Ausbildungsbetrieb bereitgestellt. In der Berufsschule wurde darüber hinaus ein bereitgestellter Schul-PC genutzt. Wurde im Homeoffice an der Simulation gearbeitet, wurde der ebenfalls der Firmenlaptop genutzt. Bei der genutzten Software wurde darauf geachtet freie Software zu verwenden, um zusätzlichen Kosten zu vermeiden.

### 3. Analysephase

#### 3.1. Ist-Analyse

Waldbrandsimulationen stellen eine weit verbreitete Methode dar, um komplexe Prozesse vereinfacht und nachvollziehbar darzustellen. Diese Simulationen können grundsätzlich in zwei Kategorien unterschieden werden, und zwar Simulationen mit einem hohen wissenschaftlichen Anspruch sowie vereinfachte, visuell orientierte Anwendungen für Lern- oder Demonstrationszwecke. Diese Simulation fällt eher in die zweite Kategorie, da die Ausbreitung des Feuers auch autonom erfolgen kann und der Benutzer lediglich durch das Verändern von Parametern wie Wetterbedingungen oder Waldstruktur Einfluss auf den Verlauf nimmt, aber auch den Wald während der Simulation live anpassen kann. Bekannte Anwendungen in diesem Bereich sind häufig Teil von wissenschaftlichen Modellen oder Lernprogrammen, die sich über Jahre hinweg weiterentwickelt haben und bereits etabliert sind. Waldbrandsimulationen zeigen eine fortlaufende Weiterentwicklung in Bezug auf Realismus und Visualisierung. Zwischen bestehenden Lösungen besteht eine gewisse Konkurrenz, wobei neue Simulationen versuchen müssen, sich durch Benutzerfreundlichkeit und Verständlichkeit von bestehenden Anwendungen abzuheben.

#### 3.2. Projektkosten

Die angefallenen Kosten, die während der Durchführung des Projektes entstanden sind, sollen im Folgenden kalkuliert werden. Die Kosten setzen sich aus den Personal- und Ressourcenkosten zusammen, die für die Realisierung des Projektes notwendig waren. Für Auszubildende wird hier ein Stundensatz von 8.50€ berechnet. Für die Nutzung der Ressourcen, dazu gehören Büroarbeitsplatz, Hard- und Software, wird eine Pauschale von 15€ berechnet.

Vorgang	Mitarbeiter	Zeit	Personal	Ressourcen	Gesamt
Entwicklungskosten	1x Auszubildender	83h	705,50 €	1.245 €	1.950,50 €
<b>Gesamt</b>					<b>1.950,50 €</b>

#### 3.3. Anwendungsfälle

Die Simulation soll verschiedene Anwendungsfälle abdecken, diese sind im Anhang [A.3 Use-Case-Diagramm](#) als Use-Case-Diagramm dargestellt.

#### 3.4 Risikoanalyse

Um den Erfolg des Projektes innerhalb des vorgegebenen Zeitrahmens sicherzustellen, wurden potenzielle Risiken identifiziert und bewertet. In folgender Tabelle stellt **E** für die Eintrittswahrscheinlichkeit, **A** für die Auswirkung und **R** für das Gesamtrisiko. Auf einer Skala 1 (gering) bis 5 (hoch). Das Gesamtrisiko ergibt sich aus dem Produkt beider Werte: (**E** x **A**).

Risiko	E	A	R	Gegenmaßnahmen (Mitigation)
<b>Performance-Probleme</b> bei hoher Baumdichte oder komplexen Effekten	4	3	12	Implementierung eines Grafikeinstellungsfensters zur Deaktivierung rechenintensiver Effekte.
<b>Zeitverzug</b> durch zu hohe Komplexität der Visualisierung	3	4	12	Priorisierung der Kernfunktionen und Nutzung von einer Checkliste zur Fortschrittskontrolle.
<b>Fehlerhafte Simulationslogik</b> (unrealistisches Brandverhalten)	2	4	8	Durchführung von Black-Box-Tests und Definition valider Wertebereiche für Slider.
<b>Datenverlust</b> des Quellcodes während der Entwicklung	1	5	5	Redundante Speicherung und Versionierung mittels Git und GitHub.

### 3.5 Lastenheft

Aus den Ergebnissen der Analysephase wurde das Lastenheft erstellt, dieses umfasst alle Anforderungen, an die zu entwickelnde Simulation. Die Anforderungen wurden im Rahmen des Berufsschulunterrichts definiert. Ein Auszug des Lastenheftes befindet sich im Anhang [A.5 Lastenheft](#).

## 4. Entwurfsphase

### 4.1. Technologieauswahl

Für die Entwicklung der Simulation wurde sich für eine Windows-Desktop-Anwendung auf Basis von WPF (Windows Presentation Foundation) entschieden. Die Vorteile von WPF liegen unter anderem in der klaren Trennung von Benutzeroberfläche und Logik sowie in den umfangreichen Möglichkeiten zur Gestaltung grafischer Benutzeroberflächen. Dadurch eignet sich WPF besonders für die Umsetzung von Diagrammen und Windows nativen Oberflächen. Zudem gibt es eine große und aktive Community, die ihr Wissen teilt. Es gibt viele Foren und Tutorials, um Fähigkeiten zu verbessern und Probleme zu lösen. Das Code-Behind der Seiten, wird in der Programmiersprache C# geschrieben. Dies hat den Vorteil, dass der Autor bereits während der praktischen Ausbildung überwiegend mit dieser Programmiersprache arbeitet und somit schon Erfahrungen mit C# sammeln konnte.

### 4.2. Architekturdesign

Das Architekturdesign in Projekten spielt eine entscheidende Rolle für die Entwicklung von effizienten und wartbaren Anwendungen. In diesem Projekt wurde ein für WPF-Anwendungen gängiges Architekturmuster angewendet, das MVVM-Muster. Dabei wird die Anwendung in drei Hauptkomponenten geteilt: das Modell, die View und das ViewModel. Das Modell repräsentiert die Datenstrukturen sowie die Simulationslogik der Anwendung. Die View ist für die Darstellung der grafischen Benutzeroberfläche zuständig und wird mithilfe von XAML umgesetzt. Das ViewModel fungiert als Bindeglied zwischen Modell und View, verarbeitet Benutzereingaben und stellt Daten über Data Binding für die View bereit. Durch diese Struktur wird eine lose Kopplung der einzelnen Komponenten erreicht.

### 4.3. Entwurf der Benutzeroberfläche

Die Benutzeroberfläche soll möglichst benutzerfreundlich und intuitiv bedienbar sein. Dazu wird die Oberfläche auf die wesentlichen Funktionen reduziert und das User-Interface klar strukturiert aufgebaut. Die Simulation wird zweidimensional dargestellt, um eine übersichtliche und leicht verständliche Vor-konfiguration der Waldbrandsimulation zu ermöglichen. Aus den Anforderungen ergab sich die Unterteilung der Anwendung in drei Oberflächen-Teile. Die erste Oberfläche ermöglicht es dem Benutzer, die Eingabeparameter der Simulation festzulegen. Hier können unter anderem Einstellungen zur Terrain- und Waldgenerierung, zum Feuerverhalten sowie zu Umwelt- und Wetterbedingungen vorgenommen werden. Die Parameter werden übersichtlich dargestellt und können vor dem Start der Simulation angepasst werden.

Der zweite Oberflächen-Teil stellt die eigentliche Simulation dar. Der Wald wird als rasterbasierte 2D-Darstellung aus der Vogelperspektive visualisiert. Einzelne Bäume werden dabei vereinfacht als farbige Zellen dargestellt, wobei die Farben den jeweiligen Zustand (gesund, brennend, verbrannt) repräsentieren und die Helligkeit die Topologische Höhe (Wie zum Beispiel auf einem Berg oder im Tal). Während der Laufzeit werden zusätzlich relevante Informationen wie Simulationsdauer, Walddichte sowie die Anzahl gewachsener und verbrannter Bäume angezeigt.

Der dritte Oberflächen-Teil dient der Auswertung der Simulationsergebnisse. In diesem Auswertungsfenster werden statistische Daten nach Abschluss der Simulation grafisch dargestellt. Dazu gehören unter anderem Diagramme zur Anzahl gewachsener und verbrannter Bäume sowie zur Entwicklung aktiver Bäume über die Zeit. Diese Darstellung ermöglicht es dem Benutzer, den Verlauf der Simulation nachträglich zu analysieren und unterschiedliche Simulationen miteinander zu vergleichen.

Zur Visualisierung wurden die initialen Entwürfe mit ChatGPT erstellt, diese befinden sich im Anhang [A.6 Mockups](#).

### 4.4. Pflichtenheft

Aus den Anforderungen, an die zu entwickelnde Simulation, aus dem Lastenheftes wurde das Pflichtenheft erstellt. Darin wurde beschrieben, wie die Anforderungen in der Simulation umgesetzt werden. Ein Auszug des Pflichtenheftes befindet sich in Anhang [A.7 Pflichtenheft](#).

## 5. Implementierungsphase

In der Implementierungsphase wurden die Resultate der Entwurfsphase umgesetzt. Zur Einleitung dieser Phase wurde ein neues Repository auf der Plattform GitHub sowie ein neues WPF-Projekt in Visual Studio angelegt.

### 5.1. Entwicklung der Benutzeroberfläche

Die Benutzeroberfläche der Simulation wurde mit dem WPF-Editor umgesetzt. Zur Vorlage dienten die Mockups aus Anhang [A.6 Mockups](#). Wie in Abschnitt [4.3 Entwurf der Benutzeroberfläche](#) beschrieben, wird die Simulation aus mehreren Szenen bestehen.

Zuerst wurde die Konfigurations-Oberfläche erstellt. Dieses Fenster besteht aus mehreren klar strukturierten GroupBoxen, die die Eingabeparameter der Simulation enthalten. Die einzelnen Elemente basieren auf WPF-Steuerelementen wie CheckBoxes, Slidern, ComboBoxen, TextBlocks und Buttons. Slider werden beispielsweise zusammen mit TextBlocks verwendet, sodass Werteänderungen sofort sichtbar sind. Buttons lösen Klick-Ereignisse aus, um Aktionen wie das Starten der Simulation zu ermöglichen. Insgesamt deckt die Konfigurations-Oberfläche alle relevanten Einstellungen ab, die die Benutzer beeinflussen können, darunter Landschaft, Wald, Feuerverhalten und Wetterbedingungen. Ein zentrales Steuerelement ist der „Graphics settings“-Button oben links, über das das Grafikeinstellungsfenster geöffnet wird. Dort können grafische Effekte wie Blitze, Funken, Rauch oder verbrannte Bäume ein- und ausgeschaltet werden. Die Gruppierung der Steuerelemente innerhalb der GroupBoxen sorgt für eine übersichtliche und intuitive Bedienung.

Die Simulations-Oberfläche stellt den Wald auf einem Canvas dar, auf dem die Bäume als dynamische Ellipse-Elemente (oder ggf. Rectangles) erzeugt und verwaltet werden. Statusinformationen wie die aktuelle Laufzeit, die insgesamt gewachsenen und verbrannten Bäume sowie die Baumdichte werden über TextBlocks angezeigt. Über die Geschwindigkeitsknöpfe kann die Wiedergabegeschwindigkeit der Simulation angepasst werden, und über Mausklicks auf dem Canvas kann der Benutzer Bäume beeinflussen. Alle Baumobjekte und Effekte werden dynamisch über Canvas-Elemente verwaltet. Die Szene speichert Wachstum und Brandereignisse, die anschließend in einer Auswertung angezeigt werden.

Die Auswertungs-Oberfläche wurde ebenfalls als eigenständiges WPF-Fenster umgesetzt. Sie zeigt nach Abschluss einer Simulation die wichtigsten Kennzahlen in übersichtlicher Form über TextBlocks, darunter Luftfeuchtigkeit, Temperatur und Laufzeit der Simulation. Für die grafische Darstellung der Ergebnisse wurden Diagramme mit OxyPlot integriert. Zwei Hauptdiagramme visualisieren den zeitlichen Verlauf der Bäume: die Gesamtzahl der gewachsenen und verbrannten Bäume sowie die Anzahl der aktuell aktiven Bäume. Die Diagramme wurden gezielt in die Benutzeroberfläche eingebunden, um die Auswertung



übersichtlich und interaktiv zu gestalten.

Im Anhang [A.8 Bilder der Anwendung](#) befinden sich Eindrücke der Szenen nach Fertigstellung.

## 5.2. Entwicklung der Geschäftslogik

Die Geschäftslogik der Waldbrandsimulation basiert auf der zentralen Konfiguration `SimulationConfig`, die `TreeConfig`, `FireConfig`, `EnvironmentConfig`, `PrefillConfig`, `TerrainConfig` und `VisualEffectsConfig` bündelt.

Die Simulation wird über das `SimulationWindow` gesteuert, worin die Klasse `ForestFireSimulation` mit Hilfe von `SimulationClock` die Simulationszeit und zeitlich basierten Events auslöst und `ForestGrid` über die Zustände der Bäume Bescheid weiß. Zunächst wird das Wald-Raster erstellt und je nach Prefill-Option teilweise mit Bäumen befüllt. Aktivierte Terrain-Generierung beeinflusst Baumwachstum und Brandausbreitung über Geländehöhen.

Bäume können wachsen, Feuer fangen und brennen, wobei Umgebungsfaktoren wie Wind, Temperatur und Luftfeuchtigkeit die Wahrscheinlichkeit für Brandausbreitung bestimmen. Zustände werden im `ForestGrid` mit Hilfe von `ForestCellState` verwaltet, und aktive Baumzellen werden auf dem Canvas angezeigt. Visuelle Effekte wie Feuer, Rauch, Blitze oder verbrannte Bäume werden mit `SimulationRenderer` über `ParticleGenerator` und die Konfiguration umgesetzt.

Alle Ereignisse und Statistiken der Simulation werden protokolliert, um eine anschließende Auswertung der Simulation zu ermöglichen. Die Logik ist modular aufgebaut und erlaubt eine flexible Anpassung von Wachstum, Feuerverhalten, Wind und Gelände.

Im Anhang [A.4 Klassendiagramm](#) befindet sich ein Auszug des Klassendiagramms, um die Beziehungen zwischen den Klassen aufzuzeigen.

## 6. Projektabschluss

### 6.1. Testen der Anwendung

Um sicherzustellen, dass sämtliche Funktionen der Anwendung korrekt und wie vorgesehen funktionieren, wurde ein Black-Box-Test durchgeführt. Das dazugehörige Testprotokoll ist im Anhang [A.9 Testprotokoll](#) finden. Dieses bietet einen Einblick in die Testdurchführung sowie die beobachteten Ergebnisse.

### 6.2. Dokumentation

Zur erstellten Software gehörten die Projektdokumentation und Benutzerdokumentation. In der Projektdokumentation werden alle Phasen des Projektes festgehalten, von den Anforderungen, Entwürfen bis zur Implementierung und dem Projektabschluss. Dadurch können getroffene Entscheidungen nachvollzogen werden und sie kann als Kommunikationsmittel für die beteiligten Personen dienen.

Die Benutzerdokumentation umfasst das Benutzerhandbuch, dieses befindet sich im Anhang [A.10 Benutzerhandbuch](#). Das Benutzerhandbuch erläutert die Funktionen und die korrekte Nutzung der Anwendung, so soll die Benutzerfreundlichkeit erhöht werden.

### 6.3. Soll-/Ist-Vergleich

Abschließend ist festzustellen, dass alle Anforderungen gemäß dem Pflichtenheft erfüllt wurden. Die anfangs erstellte Zeitplanung aus Anhang [A.1 Zeitplanung](#) konnte so eingehalten werden.

### 6.4. Ausblick

Mit der Abgabe der Simulation ist dieses Projekt beendet. Eine Fortsetzung der Arbeit an diesem Projekt ist vorerst nicht geplant. Trotzdem gibt es viele Möglichkeiten zur Weiterentwicklung dieser Simulation, die wahlweise umgesetzt werden könnten.

Einführung von Gezeiten, verschiedenen Biomen oder Wetterereignissen, die das Wachstum der Bäume und die Brandausbreitung beeinflussen.

Eine Möglichkeit wäre Nutzung von Bildern oder Animationen für Bäume, Feuer und Wettereffekte, um die visuelle Darstellung der Simulation weiter zu verbessern.

Erweiterung um einen Map-Editor, der es ermöglicht, individuelle Waldlandschaften zu gestalten, Startbedingungen anzupassen oder Brände gezielt zu setzen.

Lebewesen: Integration von Tieren oder anderen Objekten, die auf die Umgebung reagieren und zusätzliches Verhalten ins System bringen.

Analyse: Erweiterung der Auswertungsoberfläche, um Simulationsergebnisse zu visualisieren, einzusehen oder detailliertere Statistiken bereitzustellen.

## 7. Anhang

### A.1 Zeitplanung

<b>Analyse</b>	<b>2 h</b>
Ist-Analyse	1 h
Lastenheft erstellen	1 h
<b>Entwurfsphase</b>	<b>5 h</b>
Entwerfen der Mockups	2 h
Technologieauswahl	1 h
Pflichtenheft erstellen	2 h
<b>Implementierungs- und Testphase</b>	<b>68 h</b>
Entwicklung der Benutzeroberfläche	10 h
Entwicklung der Geschäftslogik	55 h
Durchführung Black-Box-Test	3 h
<b>Dokumentationsphase</b>	<b>8 h</b>
Erstellung der Dokumentation	8 h
<b>Gesamt</b>	<b>83 h</b>

### A.2 Verwendete Ressourcen

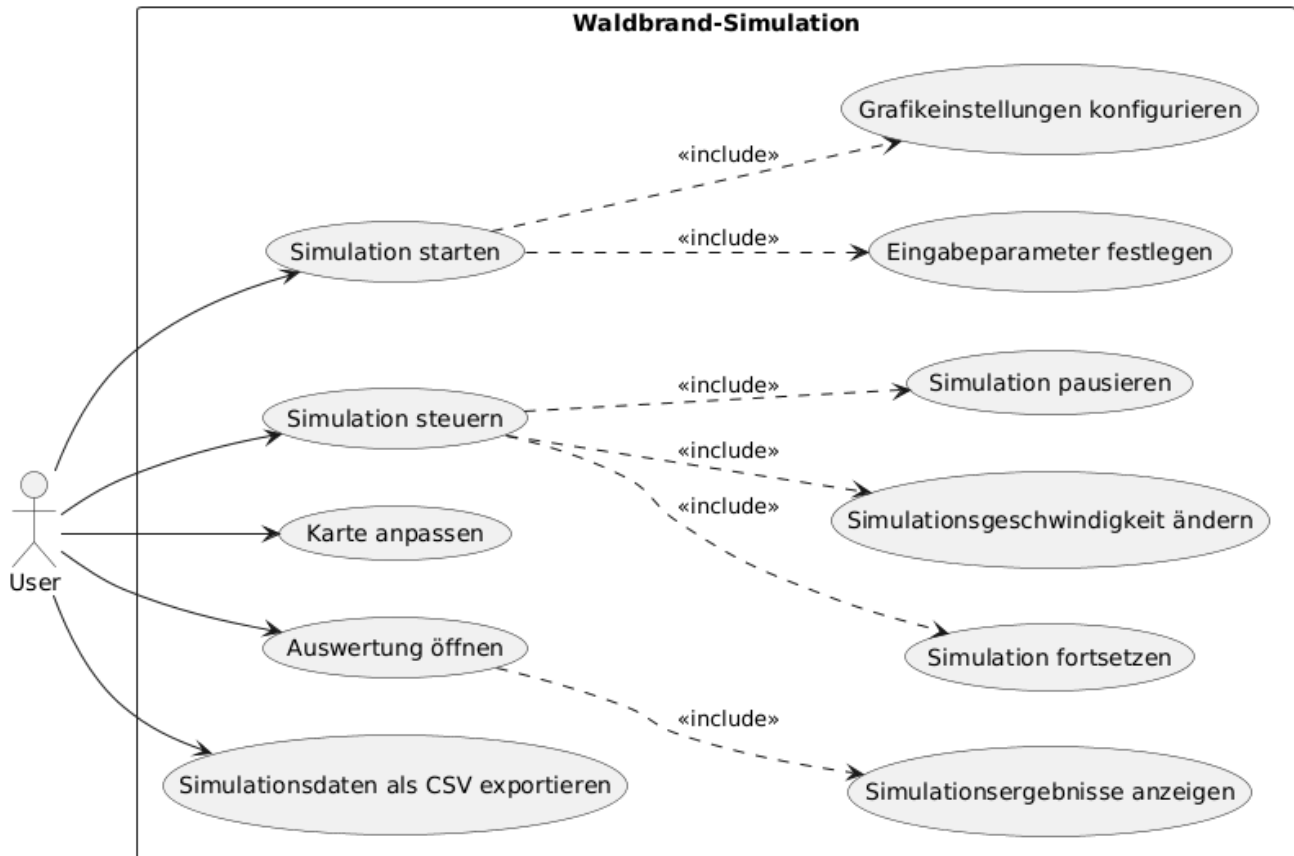
#### Hardware

- Firmenlaptop
- Schul-PC

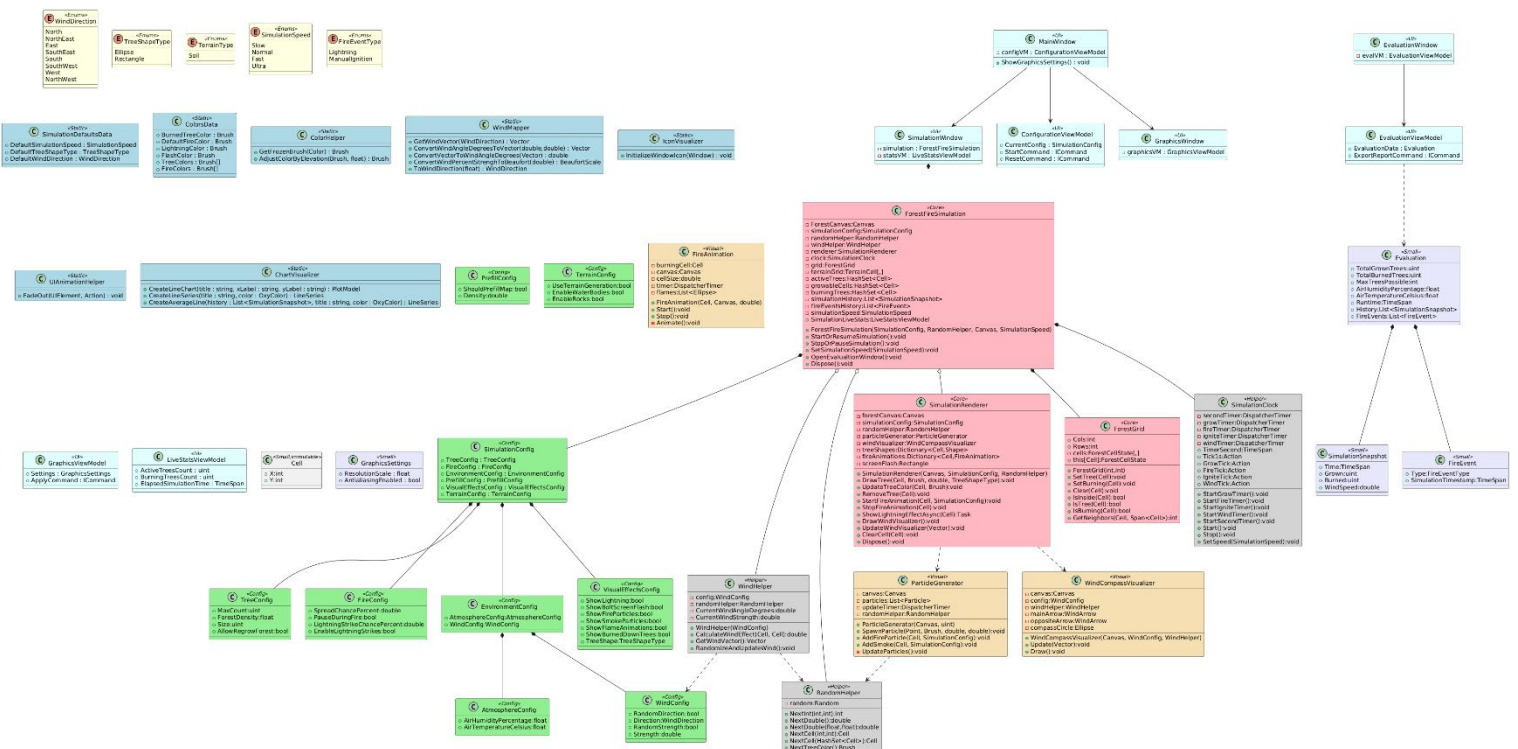
#### Software

- Windows 11
- GitHub Desktop
- Git
- Visual Studio Code
- Visual Studio 2026
- PlantUML
- ChatGPT (GPT-4.1).
- Auszubildende

## A.3 Use-Case-Diagramm



## A.4 Klassendiagramm (Auszug)



## A.5 Lastenheft (Auszug)

Im folgenden Auszug des Lastenheftes werden die Anforderungen, die die zu entwickelnde Simulation erfüllen muss, aufgeführt.

### Zielsetzung

Das Ziel dieses Projektes ist die Entwicklung einer Simulation. Das Ergebnis der Simulation soll beeinflusst werden durch Eingabeparameter und Zufallsfaktoren, sodass das Ergebnis bei jedem Durchlauf unterschiedlich ist.

### Anforderungen

1. Die Oberfläche der Simulation soll eine Animation enthalten.
2. Die Anwendung soll über eine grafische Oberfläche verfügen.
3. Der Benutzer soll die Möglichkeit haben, insgesamt sieben Eingabeparameter verändern zu können. Diese Parameter sollen den Verlauf und dadurch den Ausgang der Simulation maßgeblich beeinflussen.
4. Sinnvolle Ausgangswerte für die Eingabeparameter sollen beim Start der Anwendung bereits eingestellt sein, sodass der Benutzer bei Bedarf die Simulation sofort starten kann.
5. Es sollen drei Zufallsfaktoren eingeführt werden. Ein Faktor soll nach dem Prinzip der Gleichverteilung auftreten. Bei der Gleichverteilung haben alle möglichen Ergebnisse dieselbe Wahrscheinlichkeit einzutreten. Der zweite Faktor soll mit der Wahrscheinlichkeit der Normalverteilung, auch Gauß-Verteilung, vorkommen. Bei dieser Verteilung treten die meisten Werte um einen Mittelwert auf, während extreme Werte seltener auftreten.
6. Die Simulation soll zeitabhängig bzw. endlich sein.
7. Mindestens drei verschiedene Geschwindigkeiten sollen vom Benutzer eingestellt werden können.
8. Nach Ende der Simulation soll dem Benutzer eine Auswertung dargestellt werden.
9. Die Simulation soll als Windowsanwendung verfügbar sein.
10. Es sollen die Clean-Code-Prinzipien eingehalten werden, dadurch soll die Lesbarkeit und Verständlichkeit des Quellcodes gewährleistet werden. Dafür sollte beispielsweise darauf geachtet werden, dass Methoden nur eine Aufgabe erfüllen und Bezeichner für Klassen, Methoden und Variablen sollten sprechend sein, sie sollten nicht zu kurz oder lang sein, außerdem sollte auf die Verwendung von unbekannten Abkürzungen verzichtet werden.

## A.6 Mockups

### Mockups der Haupt-Konfigurations-Seite

```
+-----+
| Waldbrand-Simulation |
| [Simulate →] [Reset Config] |
| Baumdichte: [====|----] Windstärke: [==|-----] |
| Windrichtung: [N/NE/E...] Luftfeuchtigkeit: [==|===] |
| Temperatur: [25°C] Feuer-Startpunkt: [Random/MapClick] |
| Feuer-Ausbreitungschance: [====|----] Blitzhäufigkeit: [==|=====] |
| Terrain-Optionen: [Flat/Hills/Random] |
+-----+
| [Mini-Forest Preview Grid] |
| (Nur Darstellung, darunter nichts weiter) |
+-----+
```

### Mockups der Grafik-Seite

```
+-----+
| [Apply] [Reset] |
| Weather: [ ] Lightning [ ] Bolt-Flashes |
| Fire: [ ] Flame-Animations |
| Particles: [ ] Sparks [ ] Smoke |
| Trees: [ ] Burned Trees [ ] Tree Shapes |
+-----+
```

### Mockups der Simulation-Seite

```
+-----+
| [Pause/Resume] [Auswertung →] [Reset Config] |
| Baumstatus: Intakt: 80 | Brennend: 5 | Verbrannt: 15 |
| Wind: 5 km/h (Bft 2) ↑ | Temp: 25°C | Feuchtigkeit: 30% |
| Geschwindigkeit: [0.5x] [1x] [2x] [40x] |
+-----+
| 2D Forest Grid |
| (Grün=gesund, Rot=brennend, Grau=verbrannt, Terrain Shading) |
| Lightning/Bolt Flashes & Smoke optional |
+-----+
```

### Mockups der Auswertung-Seite

```
+-----+
| [Zurück] [Export CSV] |
| Humidity: 30% | Temperature: 25°C | Avg Wind: 5 km/h |
| Runtime: 00:10 |
+-----+
| Diagramme: |
| 1) Total Burned Trees over Time |
| 2) Grown Trees over Time (Live Tree Count) |
+-----+
```

## A.7 Pflichtenheft (Auszug)

Im folgenden Auszug des Pflichtenheftes wird die Umsetzung der Anforderungen, die im Lastenheft gestellt wurden, beschrieben.

### Zielsetzungen

Das Ziel dieses Projektes ist die Entwicklung einer Simulation. Für die Entwicklung wird das Framework WPF verwendet und der Quellcode in C# geschrieben.

### Anforderungen

1. Die Blitze und Brände sind animiert und Partikel bewegen sich auf der Oberfläche.
2. Die Oberfläche der Simulation wird mit WPF erstellt und besteht aus drei Szenen. Die "Konfigurations-Oberfläche" stellt den Ausgangszustand des Waldes dar, außerdem können Informationen zu den einzelnen Umweltfaktoren gesetzt werden.
3. Es ist dem Benutzer möglich, eine Vielzahl von Umwelt- und Simulationsparametern individuell zu konfigurieren. Dazu gehören unter anderem die Baumdichte, die Windgeschwindigkeit (skaliert nach der Beaufort-Skala), die Windrichtung, die Luftfeuchtigkeit, die Umgebungstemperatur sowie die Wahrscheinlichkeit für Blitzeinschläge. Die Eingabe erfolgt über Slider, wobei für jeden Wert ein definierter Minimal- und Maximalbereich festgelegt ist (z. B. Temperatur von -10 °C bis +50 °C), um eine wissenschaftlich plausible Simulation zu gewährleisten.
4. Es ist dem Benutzer möglich, eine Vielzahl von Umwelt- und Simulationsparametern individuell zu konfigurieren. Dazu gehören unter anderem die Baumdichte, die Windgeschwindigkeit (skaliert nach der Beaufort-Skala), die Windrichtung, die Luftfeuchtigkeit, die Umgebungstemperatur sowie die Wahrscheinlichkeit für Blitzeinschläge. Die Eingabe erfolgt über Slider, wobei für jeden Wert ein definierter Minimal- und Maximalbereich festgelegt ist (z. B. Temperatur von -10 °C bis +50 °C), um eine wissenschaftlich plausible Simulation zu gewährleisten.
5. Das Alter und die Tagesform sind die zwei Zufallsfaktoren in der Simulation. Beide Parameter wirken sich auf die Leistung der Spieler während der Simulation aus. Die Tagesform eines Spielers kann gut, neutral oder schlecht sein. Hier wurde die Gleichverteilung umgesetzt. Jede Tagesform hat die gleiche Wahrscheinlichkeit einzutreten. Die Tagesform eines Spielers kann dessen Schnelligkeit und Zweikampfstärke positiv oder negativ beeinflussen, ausgenommen der neutralen Tagesform, die darauf keinen Einfluss ausübt.
6. Die Simulation nutzt drei verschiedene Zufallsfaktoren, um die Unvorhersehbarkeit natürlicher Prozesse abzubilden:

**Gleichverteilung:** Diese wird für die räumliche Positionierung der Bäume während

der Waldgenerierung sowie für die Bestimmung der Koordinaten von Blitzeinschlägen genutzt. Jede Zelle innerhalb des Simulationsgitters hat dabei die exakt gleiche Wahrscheinlichkeit, ausgewählt zu werden.

**Normalverteilung (Gauß-Verteilung):** Die Wahrscheinlichkeit der Brandübertragung auf benachbarte Zellen folgt einer Normalverteilung.

Während Faktoren wie Wind und Trockenheit die Wahrscheinlichkeit beeinflussen, treten die meisten Ausbreitungsereignisse um einen berechneten Mittelwert auf.

Extreme Abweichungen – wie ein untypisch schnelles Überspringen oder ein plötzliches Erlöschen trotz günstiger Bedingungen – sind statistisch seltener.

**Exponentialverteilung:** Dieser Faktor steuert die zeitlichen Intervalle zwischen zufälligen Ereignissen, wie beispielsweise dem Auftreten von Sekundärbränden durch Funkenflug (Spot Fires). Hierbei sind kürzere Zeitabstände zwischen den Ereignissen wahrscheinlicher, während sehr lange Pausen seltener vorkommen, was die Intensitätsphasen eines realen Waldbrandes widerspiegelt.

7. Eine feste Zeitbegrenzung wie bei einem Sportereignis gibt es nicht; die Simulation ist dynamisch. Sie endet entweder automatisch, wenn alle Brandherde erloschen sind oder der gesamte Waldbestand vernichtet wurde, oder sie wird vom Benutzer manuell gestoppt. Bei Bedarf kann die Simulation jederzeit zurückgesetzt und mit neuen Parametern gestartet werden.
8. Zur Laufzeit der Simulation kann der Benutzer über einen Slider in der Benutzeroberfläche die Geschwindigkeit stufenlos regulieren. Dabei wird ein Bereich von der halben Geschwindigkeit (0.5x) bis hin zu einer vierzigfachen Beschleunigung (40x) abgedeckt. Ein Wechsel zwischen den Geschwindigkeitsstufen ist jederzeit ohne Unterbrechung der Berechnung möglich.
9. Sobald die Simulation gestoppt wird oder endet, wird dem Benutzer ein detailliertes Evaluations-Interface angezeigt. Statt einfacher Spielereignisse werden hier statistische Auswertungen in Form von Liniendiagrammen präsentiert. Diese visualisieren den zeitlichen Verlauf der gesund gewachsenen Bäume im Vergleich zu den verbrannten Bäumen. Zudem werden Endstatistiken zur Laufzeit und den durchschnittlichen Umweltbedingungen eingeblendet. Über eine Export-Funktion können diese Daten als CSV-Datei für externe Analysen gesichert werden.
10. Die Anwendung ist als native Windows-Anwendung ausführbar. Sie wurde für die Nutzung unter Windows 10/11 optimiert und setzt das .NET 9 SDK (Version 9.0.200) voraus.
11. Während der Programmierung wurde konsequent auf die Einhaltung von Clean-Code-Prinzipien geachtet. Dies umfasst die Kapselung der Simulationslogik in eigenen Klassen (Trennung von UI und Logik), die Verwendung sprechender Bezeichner sowie die Einhaltung des Single-Responsibility-Prinzips, um eine hohe Wartbarkeit und Lesbarkeit des Quellcodes zu garantieren.



## A.8 Bilder der Anwendung

Forest fire simulation configuration

**Graphic config** **Reset config** **Simulate**

**Terrain**

☒ Enable terrain generation ⓘ

**Trees**

☒ Grow forest ⓘ

Prefill density: 80% ⓘ

**Fire**

☒ Pause growing during fire ⓘ

Additional fire spread chance: 40% ⓘ

Lightning strike chance: 15% ⓘ

**Environment**

Air humidity: 50% ⓘ

Air temperature: 30 °C ⓘ

☐ Random wind direction ⓘ

Wind direction (from) ⓘ

North

☐ Random wind strength ⓘ

Wind Strength: 25% (3 Bft) ⓘ

Forest fire graphics settings

**Reset config** **✓ Apply**

**Weather Effects**

☒ Show lightning

☐ Show bolt flashes

**Fire**

☒ Show flame animations

**Particle Effects**

☒ Show sparks

☒ Show smoke

**Tree Effects**

☐ Show burned trees

Ellipse

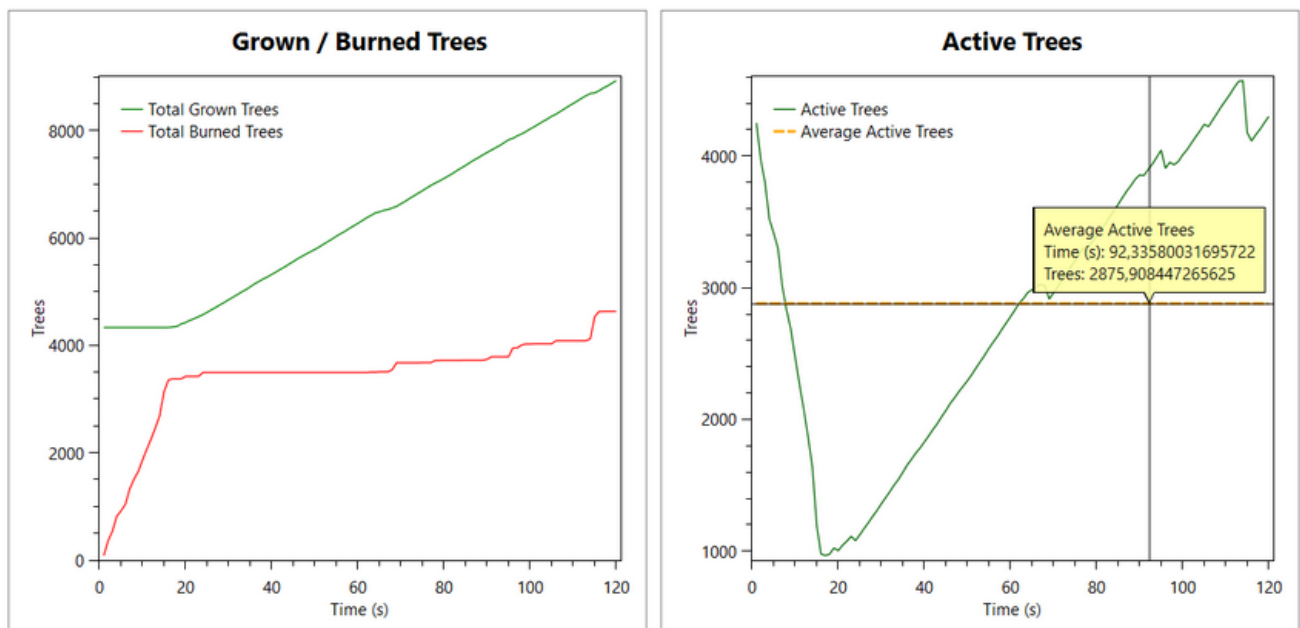


Forest fire simulation evaluation

## Simulation Evaluation

Export CSV

Air Humidity: 50 %  
Air Temperature: 30°C  
Average Wind Speed: 25% (3 Bft)  
Runtime: 00:02:00



## A.9 Testprotokoll

### Projektinformationen

- Projektname: Waldbrand Simulation
- Version: 1.0
- Datum des Tests: 31.01.2026
- Tester: Dominik Hinze
- Ziel des Tests: Mit diesem Test sollen die Funktionalitäten und der richtige Ablauf der Simulation getestet werden.

### Testumgebung

- Betriebssystem: Windows 11
- SDK-Version: 10.0.102

### Getestete Funktionen

1. „Grow-Forest“-Checkbox
2. „Random-Wind-Direction“
3. Blitzeinschlag-Abstände
4. Tastatur-Reset (R)
5. Korrekte Info-Zeichen Tooltips
6. Wiedergabegeschwindigkeit testen
7. Die Auswirkungen der Grafiksettings auf die Brand-Darstellung
8. Die automatische Führung zur Auswertung im nicht wachsenden Wald-Modus
9. Wind-Kompass muss sich mit Random-Wind dynamisch aktualisieren
10. Overlay verschwindet, wenn auf dem Fortsetzen/Start -Knopf drückt
11. Start/Pause mit Leertaste ausführbar
12. Maustasten-Funktionen für Wald-Bearbeitung
13. Höchste Wiedergabegeschwindigkeit beeinflusst Effektdarstellung
14. Windeffekt-Berechnung mit wenig und viel Wind-Stärke
15. Luftfeuchtigkeit beeinflusst Ausbreitungschance
16. Lufttemperatur wirkt sich auf Waldbrand aus
17. Wald-Dichte beeinflusst Chance auf Blitzeinschlag bei Baum-Zelle
18. Exportierung der Auswertungsdaten per Knopfdruck als CSV-Datei
19. Nachpflanzung der Bäume über 100% Dichte möglich
20. Generierung des Terrains

### Testdurchführung

ID	Testfall / Beschreibung	Eingabe / Aktion	Erwartetes Ergebnis	Status
TF 01	Grow-Forest-Checkbox	Checkbox aktivieren / deaktivieren.	Das Nachwachsen der Bäume startet bzw. stoppt sofort. Zudem wird Pause-Fire korrekt deaktiviert.	OK
TF 02	Random-Wind-Direction	Option "Zufällige Windrichtung" aktivieren.	Die Windrichtung ändert sich in Intervallen (15sek Normalspeed) automatisch.	OK
TF 03	Blitzeinschlag-Abstände	Slider für Blitzfrequenz variieren.	Die zeitlichen Abstände zwischen Blitzen entsprechen der Einstellung.	OK
TF 04	Tastatur-Reset (R)	Taste "R" während der Simulation drücken.	Simulation wird zurückgesetzt und auf Startwerte gesetzt.	OK
TF 05	Info-Tooltips	Maus über Info-Icons (i) bewegen.	Korrekte Hilfetexte werden verzögerungsfrei angezeigt.	OK
TF 06	Wiedergabegeschwindigkeit	Wechsel zwischen 0.5x, 1x, 2x und 40x.	Simulationsschritte beschleunigen/verlangsamen sich korrekt.	OK
TF 07	Grafiksettings-Einfluss	Partikeleffekte in Settings umschalten.	Visuelle Brand-Darstellung ändert sich (Rauch/Flammen an/aus).	OK
TF 08	Auto-Auswertung	Waldbrand im "Nicht-Wachsend"-Modus löschen lassen.	Nach Brandende wird automatisch das Auswertungs-UI eingeblendet.	OK
TF 09	Dynamischer Kompass	Random-Wind aktivieren und UI beobachten.	Der Windpfeil im Kompass aktualisiert sich live zur Windänderung.	OK
TF 10	Overlay-Verhalten	Auf "Fortsetzen" oder "Start" klicken.	Das Start/Pause-Menü verschwindet und gibt die Sicht auf das Grid frei.	OK
TF 11	Leertasten-Steuerung	Leertaste während der Simulation drücken.	Simulation wechselt zwischen Pause und Start (Toggle-Funktion).	OK
TF 12	Maus-Interaktion (Wald)	Linksklick/Rechtsklick auf Grid-Zellen.	Manuelles Setzen/Löschen von Bäumen im laufenden Betrieb möglich.	OK
TF 13	Effekt-Performance (40x)	Geschwindigkeit auf 40x stellen.	System bleibt stabil; Partikeleffekte werden ggf. reduziert dargestellt.	OK

ID	Testfall / Beschreibung	Eingabe / Aktion	Erwartetes Ergebnis	Status
TF 14	Windstärke-Logik	Simulation mit 10% vs. 90% Windstärke starten.	Deutlicher Unterschied in der Ausbreitungsgeschwindigkeit erkennbar.	OK
TF 15	Luftfeuchtigkeit	Luftfeuchtigkeit auf Maximum (100%) stellen.	Brand breitet sich kaum noch aus; Feuer erlischt schneller.	OK
TF 16	Lufttemperatur	Temperatur auf Maximum stellen.	Brand-Ausbreitungsrate erhöht sich spürbar (Normalverteilung).	OK
TF 17	Blitz-Dichte-Logik	Vergleich Blitzrate bei 10% vs. 90% Baumdichte.	Blitze führen bei hoher Dichte häufiger zu erfolgreichen Zündungen, kontrolliert nach der „Density“.	OK
TF 18	CSV-Export	Button "Export as CSV" klicken.	Valide Datei wird erzeugt, die alle Simulationsparameter enthält: TimeSeconds, TotalGrown, TotalBurned, ActiveTrees, WindSpeed, WindBft	OK
TF 19	Überfüllung (Nachpflanzen)	Versuchen, Bäume über 100% Dichte zu setzen.	System verhindert Fehler; Dichte wird korrekt gedeckelt.	OK
TF 20	Terrain-Generierung	Map-Generation mit Terrain-Profil starten.	Grid zeigt Höhenunterschiede (Helligkeitsstufen) korrekt an.	OK

#### Zusammenfassung

Abschließend sind die Tests erfolgreich abgeschlossen und entsprechen dem erwarteten Ergebnis.

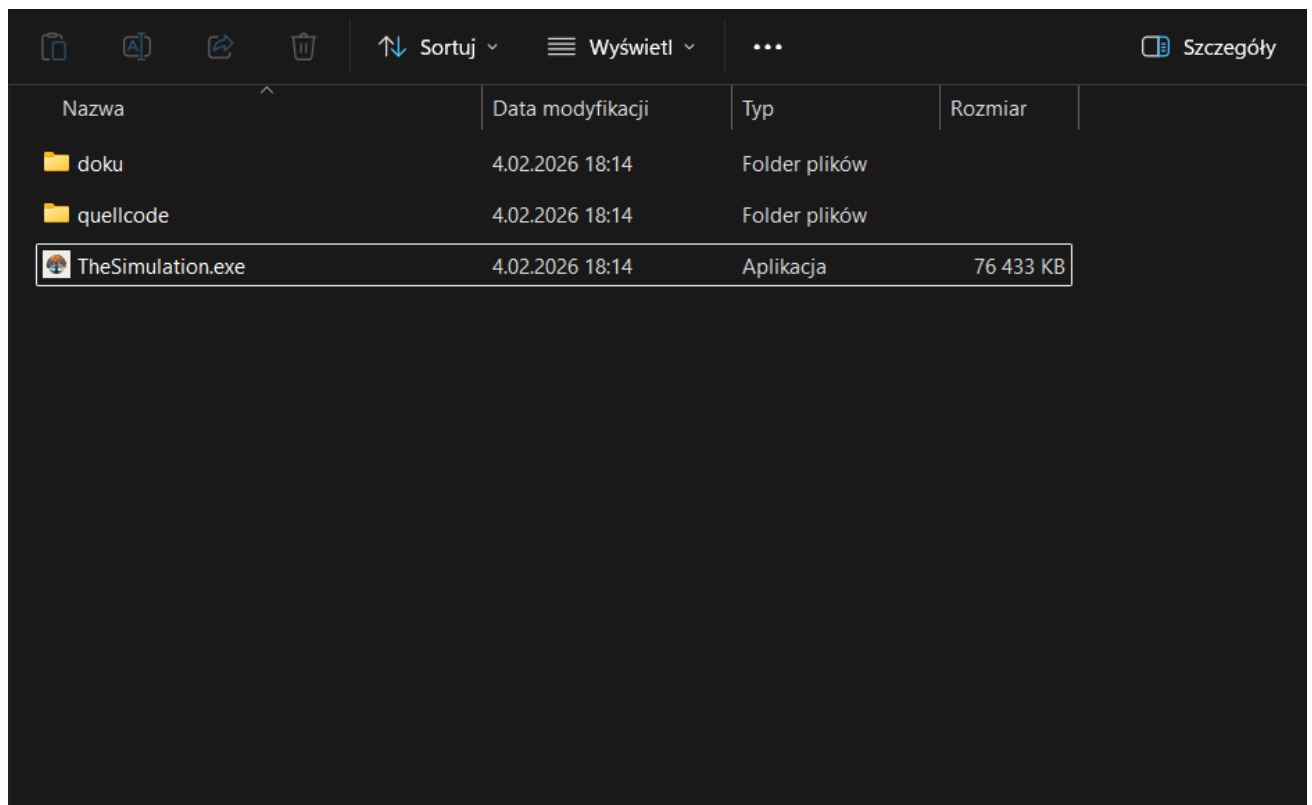
## A.10 Benutzerhandbuch

### Produktbeschreibung

Bei dieser Anwendung handelt es sich um eine 2D-Waldbrandsimulation. In der Simulation wird die Ausbreitung von Bränden über einen Wald. Es können verschiedene Einstellungen verändert werden, die das Verhalten beeinflussen.

### Inbetriebnahme

Um die Simulation starten zu können, ist keine Installation notwendig. Es reicht im Projektordner die „ForestFireSimulaiton.exe“ auszuführen, dadurch startet die Anwendung.



Alternativ kann im gleichen Ordner in den `quellcode`-Ordner navigiert werden und darin die „setup.exe“ ausgeführt werden. Somit kann das Programm auch auf dem PC installiert werden.

### Anleitung: Simulation konfigurieren

Um Simulationswerte zu verändern kann auf der ersten Seite ein Parameter ausgewählt werden, indem auf den entsprechenden Slider oder Checkbox mit der Maus gedrückt wird und ggf. verschoben wird. Dadurch hat der Nutzer die Möglichkeit einzelne Dinge über das Simulationsgeschehen zu beeinflussen. Um zu verstehen was genau geschieht, gibt es Tooltips, die über das `i` neben dem Element zu erreichen sind. Wenn man den Mauszeiger eine gewisse Zeit darüber hält, kann man erfahren, wie genau diese Funktion das Verhalten der Anwendung ändert.

Forest fire simulation configuration

Graphic config
Reset config
Simulate

Terrain
☒ Enable terrain generation ⓘ

Trees
☒ Grow forest ⓘ  
Prefill density: 80% ⓘ

Fire
☒ Pause growing during fire ⓘ  
Additional fire spread chance: 40% ⓘ  
Lightning strike chance: 15% ⓘ

Environment
Air humidity: 50% ⓘ  
Air temperature: 30 ⓘ  
☐ Random wind direction ⓘ  
Wind direction (from) ⓘ  
North  
☐ Random wind strength ⓘ  
Wind Strength: 75% (9 Bft) ⓘ

**Air Humidity:** Ambient moisture level in % (0–100%).  
**Effect on Fire:** - High humidity → easier fire spread -  
Low humidity → easier fire spread  
**Distribution:** Instant effect applied uniformly across the  
simulation grid.

## Anleitung: Grafik-Effekte verändern

Um grafische Anpassungen machen zu können, kann zu den Einstellungen über den „Graphic config“ Knopf gelangen.

Forest fire simulation configuration

Graphic config
Reset config
Simulate

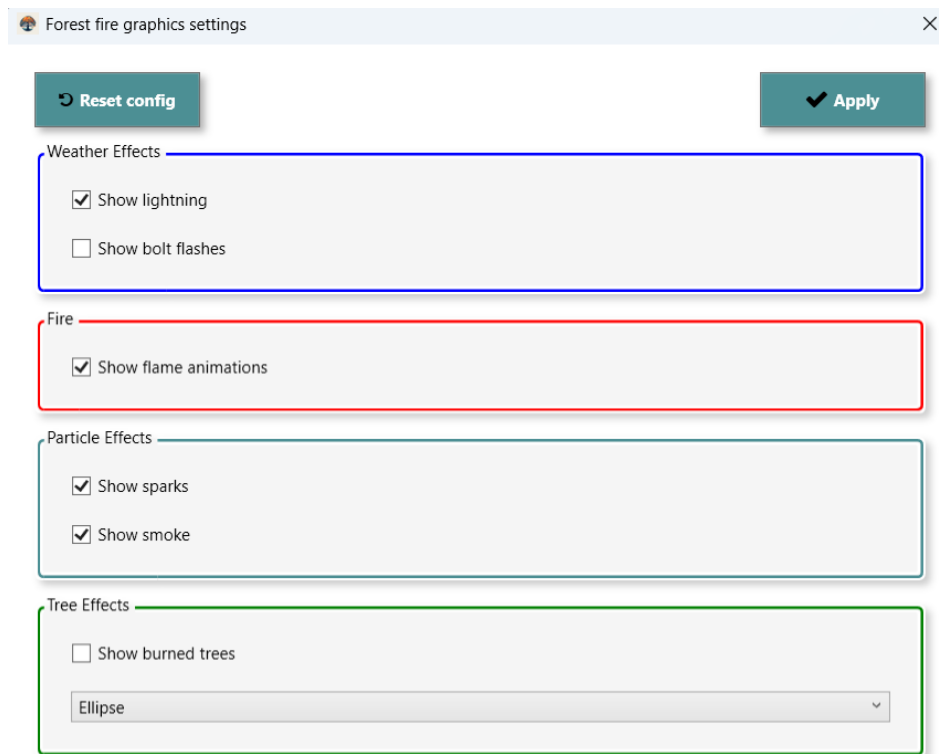
Terrain
☒ Enable terrain generation ⓘ

Trees
☒ Grow forest ⓘ  
Prefill density: 80% ⓘ

Fire
☒ Pause growing during fire ⓘ  
Additional fire spread chance: 40% ⓘ  
Lightning strike chance: 15% ⓘ

Environment
Air humidity: 50% ⓘ  
Air temperature: 30 °C ⓘ  
☐ Random wind direction ⓘ  
Wind direction (from) ⓘ  
North  
☐ Random wind strength ⓘ  
Wind Strength: 25% (3 Bft) ⓘ

Im Grafik-Dialog können Effekte wie Blitze, Feuer-Effekte und auch die Form der Bäume nach Belieben geändert werden.



The image shows a dialog box titled "Forest fire graphics settings" with a close button (X) in the top right corner. At the top left is a "Reset config" button with a circular arrow icon, and at the top right is an "Apply" button with a checkmark icon. The dialog is divided into four sections, each with a colored border and a title bar:

- Weather Effects** (blue border): Contains two checkboxes: "Show lightning" (checked) and "Show bolt flashes" (unchecked).
- Fire** (red border): Contains one checkbox: "Show flame animations" (checked).
- Particle Effects** (teal border): Contains two checkboxes: "Show sparks" (checked) and "Show smoke" (checked).
- Tree Effects** (green border): Contains one checkbox: "Show burned trees" (unchecked), and a dropdown menu below it with "Ellipse" selected.

In diesem Dialog kann man dann die getätigten Einstellungen bestätigen über den Klick auf den „Apply“ Knopf. Falls man wieder zum Ursprungszustand wieder zurück möchte, ist es möglich hier mit dem „Reset config“ Knopf, die Konfiguration und somit alle Änderungen wieder rückgängig zu machen. Dasselbe kann auch immer mit der Taste R erreicht werden.



### Anleitung: Simulation starten

Um das Geschehen zu starten, muss lediglich der Knopf mit der Aufschrift "Simulate" auf der ersten Seite gedrückt werden und ein weiteres Fenster öffnet sich, wo die eigentliche Simulation drin stattfinden wird.

Forest fire simulation configuration

Graphic config Reset config Simulate

Terrain

- ☒ Enable terrain generation ⓘ

Trees

- ☒ Grow forest ⓘ
- Prefill density: 80% ⓘ

Fire

- ☒ Pause growing during fire ⓘ
- Additional fire spread chance: 40% ⓘ
- Lightning strike chance: 15% ⓘ

Environment

- Air humidity: 50% ⓘ
- Air temperature: 30 °C ⓘ
- ☐ Random wind direction ⓘ
- Wind direction (from) ⓘ
- North
- ☐ Random wind strength ⓘ
- Wind Strength: 25% (3 Bft) ⓘ

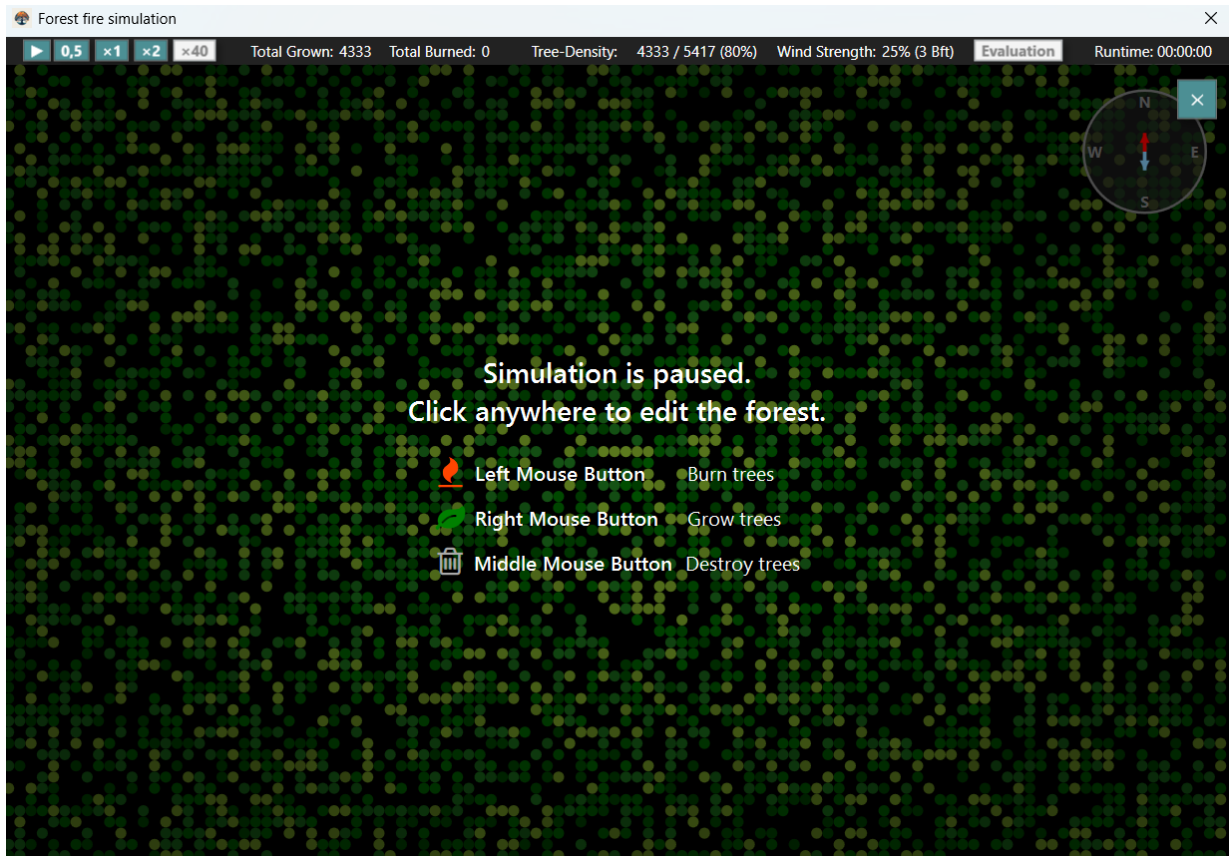
### Anleitung: Brand pausieren und Anwendung verlassen

Um den Brand auf der Simulationsseite zu beenden, kann der Schließen Knopf gedrückt werden.

Auf der gleichen Art kann das Geschehen auf der Simulationsseite unterbrochen werden. So lässt sich die Anwendung zu beenden.

## Erklärung der Funktionstasten des Waldes

Auf der Simulationsseite hat der Benutzer die Möglichkeit, wie in dem Overlay beschrieben, die den Wald anzupassen (zu Beginn und auch während der Laufzeit). Es gibt Icons und auch Text für die einzelnen Mautasten die genutzt werden können. Sobald etwas gedrückt wird, oder die Simulation gestartet wird, schließt sich das Overlay sofort, indem es langsam verschwindet. Overlay schließen ist auch möglich durch den blauen X-Knopf oben rechts.



## Anleitung: Benutzung der Buttons

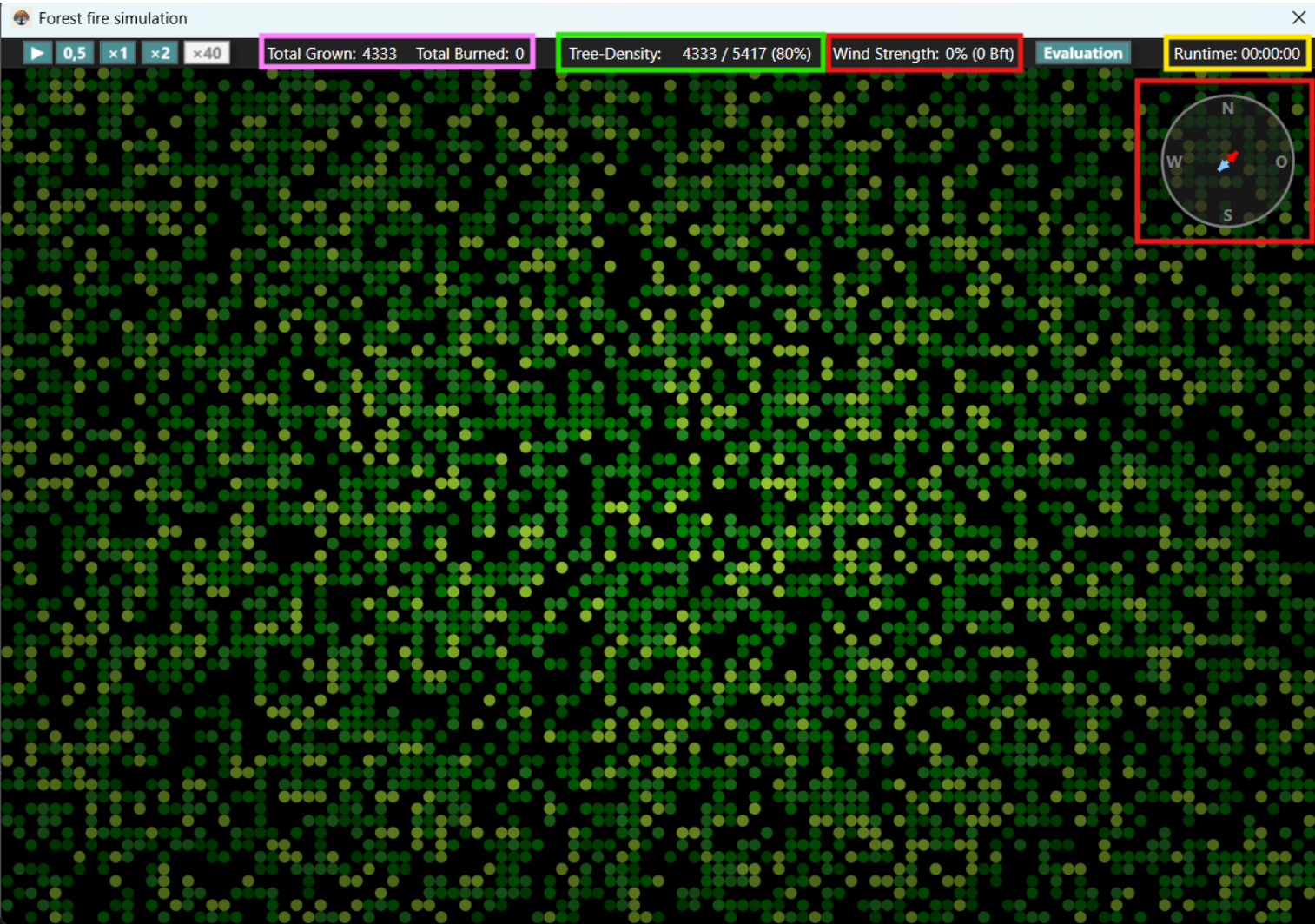
Als erstes in der oberen Leiste finden wir den Pause Knopf. Damit kann man anfangs die Simulation starten. Das Ganze geht auch durch die initiale Betätigung von Leertaste. Danach folgen die Geschwindigkeits-Knöpfe (0,5x, 1x, 2x, und 40x). Standardmäßig ist die Simulation auf 40x gestellt, um ein erlebnisreiches Geschehen zu ermöglichen, in sehr kurzer Zeit. Falls Brände und Editieren der Karte schwerfallen sollte mit der hohen Geschwindigkeit, kann man durchs klicken der anderen Geschwindigkeits-Knöpfe, die Geschwindigkeit ändern, auch währenddessen die Simulation pausiert ist. Durchs Drücken des „Evaluation“-Knopfes, ist es sichergestellt, dass man sich die Laufzeit-Auswertung und Informationen immer beliebig in der Zwischenzeit ansehen und auswerten kann. Man dafür auch beliebig viele Auswertungsfenster öffnen und damit auch Vergleichbarkeit mit mehr Zeitpunkten haben.

## Anleitung: Zusätzliche Info-Panels und Statistiken im Simulatonsfenster

Darüber hinaus ermöglichen die Textblöcke eine ausgiebige Einsicht in Details des Brand-Geschehens. Als allererstes die Gesamtanzahl der insgesamt gewachsenen Bäume und daneben die Gesamtanzahl der bisher verbrannten Bäume. Als nächstes eine Kombination aus jetzt grade stehenden Bäumen im Verhältnis zu wie viele es insgesamt in Wald sein könnten maximal einstellbar, wenn man die Simulation öffnet. Einmal normale Zahlen getrennt mit einem Schrägstrich und dahinter in Prozent. Die Windstärke darf auch nicht fehlen. Diese wird als Prozentzahl angemerkt und aber auch in der Beaufort Skala. Das hat den Hintergrund die Vergleichbarkeit mit der Realität herstellen zu wollen und damit auch die km/h einschätzen zu können. Zuletzt hat man immer einen Live-Überblick über die Gesamtlaufzeit (Runtime).

#### Erklärung: Wind-Kompass

Oben rechts in der Zeichenfläche des Waldes, gibt es einen Kompass der einen immer die Windrichtung anzeigt anhand der Himmelsrichtungen, die es dort zusätzlich darauf abgebildet gibt. Die rote Nadel zeigt immer auf die Richtung wo der Wind momentan herkommt und die blaue Nadel zeigt immer in die Richtung, wo der Wind sich dementsprechend der Wind hinbewegt. Die Länge der Pfeile kann variieren, als ergänzender visueller Indikator wie stark der Wind grade bläst, über des gesamten Waldes, um die Übersichtlichkeit und Verständlichkeit zu erhöhen, sowohl als auch die Zeit, die es braucht, um die Windstärke abzulesen, zu minimieren.



Legende:

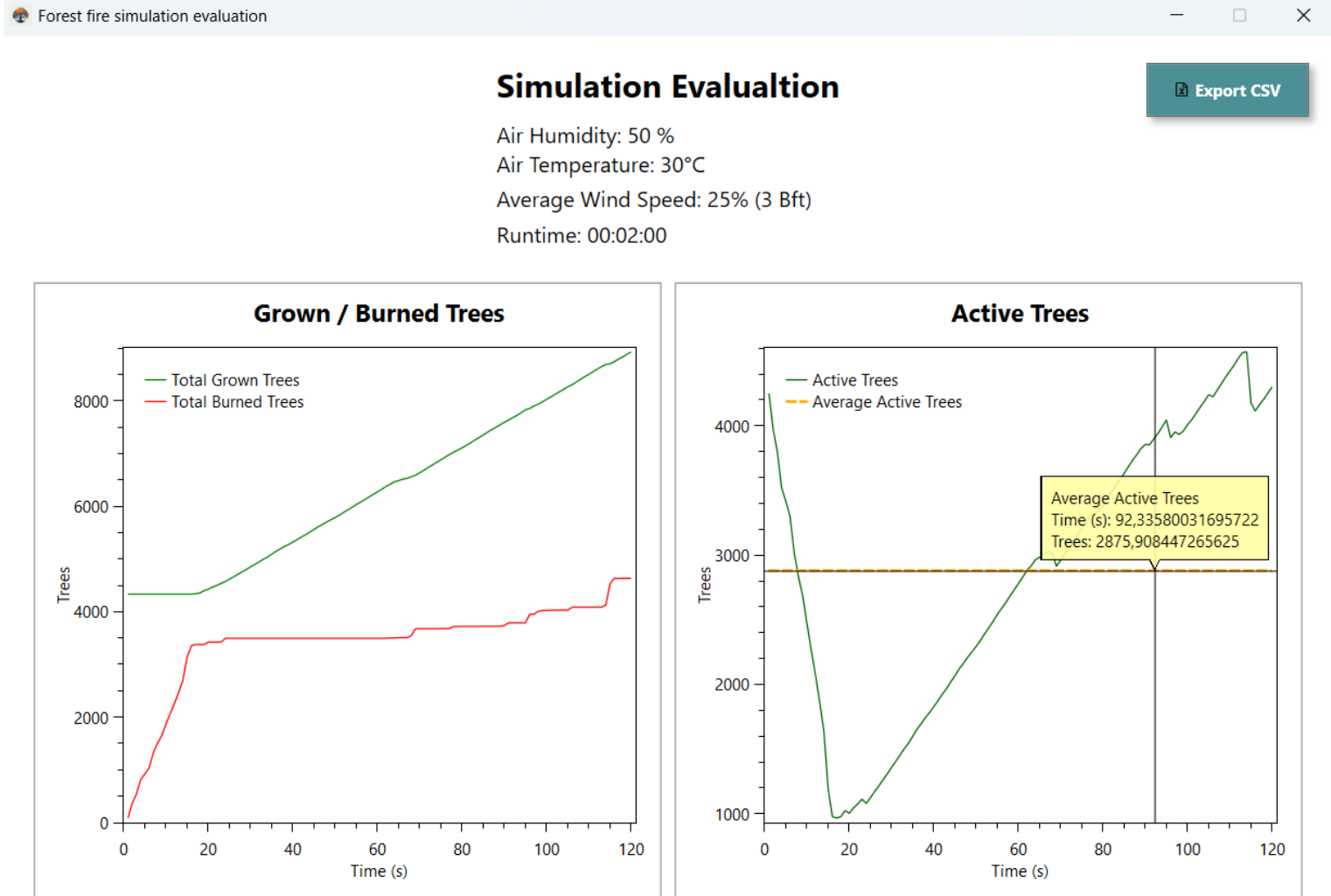
	Gesamtzahl: Bäume
	Live-Ergebnisse
	Wind (Kompass)
	Simulationszeit

## Erklärung: Auswertung & Export

Wenn der Export Knopf gedrückt wird, öffnet sich der Windows-File-Dialog.

Dort kann die Datei beliebig benannt und an den gewünschten Ort abgelegt werden.

CSV ist super die für Excel oder auch für das permanente Vergleichen von alten Simulationen.



## Systemanforderungen

- Betriebssystem: mindestens Windows 10
- Arbeitsspeicher: mindestens 4 GB Ram

## Support-Kontakt

Unseren Support erreichen Sie unter folgender E-Mail-Adresse: [dhi@gecko.de](mailto:dhi@gecko.de)