

System Breach

Projektziel:

Entwicklung eines strategischen Tower-Defense-Spiels, das Spieler gegen digitale Bedrohungen wie Viren, Trojaner und Ransomware kämpfen lässt. Das Spiel soll modular aufgebaut, objektorientiert implementiert und mit einer übersichtlichen Benutzeroberfläche versehen sein.

Kurzbeschreibung:

Der Spieler verteidigt seine digitale Basis vor ankommenden Bedrohungen, indem er verschiedene Tower-Typen strategisch aufstellt. Unterschiedliche Enemy-Typen (Viren, Trojaner, Ransomware) besitzen besondere Eigenschaften und Angriffsmuster. Das Spiel bietet ein Levelsystem mit Wellen von Gegnern, Upgrades für Türme und ein HUD zur Anzeige von Ressourcen und Fortschritt.

Woche	Aktivität	Aufgaben	Verantwortliche	Meilenstein	Ressourcen	Risiken	Zeit
1	Projektinitiierung	<ul style="list-style-type: none">• Kick-off & Zieldefinition• Repository & Versionskontrolle einrichten• Entwicklungsumgebung vorbereiten	Dev	Projektziele festgelegt	GitHub / IDE / UML-Tool	Unklare Zieldefinition	2h + 4h + 4h = 10h
2	Anforderungsanalyse	<ul style="list-style-type: none">• Gameplay & Tower-/Enemy-Arten definieren• Spielregeln & Fortschrittslogik• Erstellung erster Use-Cases	Dev	Anforderungen dokumentiert	Notizen, UML	Unvollständige Anforderungen	4h + 3h + 3h = 10h
3	Anforderungsdokumentation	<ul style="list-style-type: none">• Erstellung Pflichtenheft• Use-Case-Diagramme vorbereiten• Review & Feedback	Dev, Tutor	Anforderungsdokument genehmigt	Notizen	Fehlende Abnahme	5h + 3h + 2h = 10h
4	Klassenmodell fertigstellen	<ul style="list-style-type: none">• Vollständiges Klassendiagramm (Entities, Controller, Manager,	Dev	Klassendiagramm fertig	Astah	Überkomplexes Design	5h + 3h + 2h = 10h

		Multiplikitäten) • Review mit Betreuer • Anpassung auf Basis Feedback					
5	Implementierung Phase 1	• GameLoop & GameWorld implementieren • Enemy-/Tower-Basisklassen • Erste Interaktionen (Spawning, Movement)	Dev	Grundgerüst lauffähig	IDE / JavaFX / Git	Bugs bei GameLoop oder Bewegung	4h + 4h + 2h = 10h
6	Implementierung Phase 2	• MapManager & WaveManager, JSON Loader • HUD Grundimplementierung • Controller ↔ Manager Integration	Dev	Kernspiel mit ersten Funktionen	IDE / JavaFX / JSON	Parsingfehler, Integrationsprobleme	4h + 3h + 3h = 10h
7	Implementierung Phase 3	• Menü, Level-Auswahl, Settings-Screen • Tower Upgrades & Schadenslogik • Projectile Management	Dev	Interaktive Grundfunktionen fertig	IDE / JavaFX / JSON	UI Bugs, Logikfehler	4h + 3h + 3h = 10h
8	Implementierung Phase 4 / Feinschliff	• Ransomware & Spezial-Enemies • Balancing der Wellen • Weitere Playtests	Dev	Spiel vollständig spielbar	IDE / Testumgebung	Bugs in Enemy-Logik, Balancingprobleme	4h + 3h + 3h = 10h
9	Integration & Testen der Anwendungsfälle	• Komponentenintegration • Unit-Tests (Kernlogik) • Erste manuelle Playtests	Dev, Team	Funktionale Tests bestanden	Testframework / IDE	Unentdeckte Integrationsfehler	4h + 3h + 3h = 10h
10	Fehlerbehebung & Optimierung	• Debugging & Performanceanalyse	Dev	System stabil & performant	IDE	Zeitüberschreitung bei	4h + 3h + 3h = 10h

		<ul style="list-style-type: none"> • Code Cleanup & Refactoring • Balancing der Gegnerwellen 				Optimierung	
11	Benutzerakzeptanztest (UAT)	<ul style="list-style-type: none"> • Durchführung von UAT mit Testspielern • Feedback auswerten & einarbeiten 	Dev, Freunde	UAT abgeschlossen, Feedback eingearbeitet	IDE / Feedback	Stakeholder unzufrieden, UX-Probleme	3h + 3h + 4h = 10h
12	Projektabschluss & Dokumentation	<ul style="list-style-type: none"> • Abschlussbericht und technische Doku • UML finalisieren • Präsentation & Übergabe 	Dev, Tutor, Prof	Projekt abgeschlossen	Office / UML / Präsentationstool	Zeitdruck bei Dokumentation	4h + 3h + 3h = 10h