

NR	BEZEICHNUNG	INHALT	UMSETZUNG
	Titel	ChessGame	
	Name	Nick Philipp Häcker	
	Matrikelnummer	262144	
1	Nutzerinteraktion	Der Nutzer kann mit der Applikation interagieren. Mit welchen Mitteln und welchen Aktionen werden welche Reaktionen ausgelöst?	<p>Spieler 1 (schwarze Figuren): Mit A und D wird die entsprechende Schachfigur ausgewählt, mit W und S werden die Movement Positionen ausgewählt, hat man einen Bauer ausgewählt und steht direkt diagonal zum Gegner so kann man mit Q oder E auf den Angriffsmodus umschalten, bestätigt wird das Movement mit der Enter Taste.</p> <p>Spieler 2 (weiße Figuren): Mit < und > wird die entsprechende Schachfigur ausgewählt, mit ^ und v werden die Movement Positionen ausgewählt. Hat man einen Bauer ausgewählt und steht direkt diagonal zum Gegner so kann man mit Q oder E auf Angriffsmodus umschalten, bestätigt wird das Movement mit der Enter Taste.</p>
2	Objektinteraktion	Mit Hilfe von Kollisionsprüfung interagieren Objekte miteinander. Wann passiert dabei was?	Selektiert man eine Zielposition, so wird an die ausgewählte Figur ein MovementController angehängt, welcher überprüft, ob an der Zielstelle eine feindliche Spielfigur steht. Ist das so, so wird diese Spielfigur gelöscht und der Name der Spielfigur wird auf die Grabliste des Besitzers gesetzt.
3	Objektanzahl variabel	Eine variable Anzahl von Objekten wird zur Laufzeit generiert. Welche sind dies und wie geschieht die Erzeugung?	Zur Auswahl der Zielposition einer jeden Schachfigur wird ein gelber Punkt auf den Platz gepawnt, dieser ist ein Child des Places. Zudem reduziert sich die Zahl der Schachfiguren durch Auslöschung. Außerdem wird durch das new Statement und der Auswahlbestätigung einer Schachfigur jeweils ein neuer MovementController erstellt. Selbiges gilt auch für die SoundController, die bei Kollision, Movement oder Auswahl mit dem new Statement neu erstellt werden.
4	Szenenhierarchie	Die Szenenhierarchie ist sinnvoll aufgebaut. Wer ist wessen Parent, wie sind Elemente in anderen gruppiert und warum?	ChessGame ❖ Surface ❖ Figures <ul style="list-style-type: none"> ➤ Player <ul style="list-style-type: none"> ▪ Chessfigures ▪ ... ➤ Enemy <ul style="list-style-type: none"> ▪ Chessfigures

			<ul style="list-style-type: none"> ■ ... ❖ Places <ul style="list-style-type: none"> ➤ Place ➤ Place ➤ ... ❖ Player ❖ Selection <p>In den Sohnknoten vom Knoten Figures befinden sich als Kinder der Knoten Player und Enemy die jeweiligen Schachfiguren wieder. Im Knoten Places sind alle Places des Brettes als Sohnknoten eingereiht. Der Knoten Player bezieht sich auf die Kamera des Spiels, von welcher aus alles gesichtet werden kann. Zudem erhält der Auswahlstein, der über den Schachfiguren schwebt, einen eigenen Knoten. Die Zielpositionen der Auswahlkreise werden als Kindknoten der Knoten Place angehängt.</p>
5	Sound	Sounds sind eingebunden und unterstützen oder ermöglichen die Wahrnehmung der Aktionen. Welche Ereignisse werden durch Geräusche akustisch unterstützt, und durch welche Geräuschkulisse oder Musik die Atmosphäre?	Der Sound AtmoRiver untermauert die Atmosphäre des Spiels. Der Sound SelectChess ertönt, sobald man eine Schachfigur ausgewählt hat. Der Sound ChessMove-v3 ertönt, sobald man die Zielposition ausgewählt hat. Der Sound ChessHit-v1 ertönt, der MovementController eine Collision berechnet hat. Der Sound SelectField ertönt, sobald der Spieler ein mögliches Zielfeld der Schachfigur ausgewählt hat. Der Sound ClockTick ertönt, sobald ein Spieler am Zug ist.
6	GUI	Ein grafisches Interface gibt dem Nutzer die Möglichkeit, Einstellungen beim Programmstart oder während des Programmlaufs vorzunehmen. Was kann er dort tun?	Ein Schriftzug im Viewport zeigt den Namen des Spielers an, der am Zug ist. Beim Start des Spiels müssen beide Spieler ihre Namen in die entsprechenden Felder eingeben und bei Betätigen des Start Buttons beginnt das Spiel. Zudem befindet sich rechts oben ein Soundbutton, bei dem der Sound an und ausgestellt werden kann. Darunter befindet sich eine Übersicht des Friedhofs mit bereits getöteten Spielfiguren des Nutzers. Bei Abschließen des Spiels kann das Spiel durch den Restart Button erneut gestartet werden. Außerdem erscheint, sobald ein Spieler das andere Schach stellt, das Wort Schach am oberen Rand des Bildschirms.

7	Externe Daten	<p>Spielparameter sind extern in einer Datei veränderbar, so dass das Spiel nur neu gestartet, aber nicht neu kompiliert werden muss.</p> <p>Welche Parameter sind dies und was sind die Auswirkungen?</p>	<p>Es kann das Movement der einzelnen Schachfiguren in der ChessFigures.json verändert werden. Zudem kann in der GameSetting.json der Sound verändert oder aktiviert/deaktiviert werden. Man kann auch einstellen, ob die Maus im Spiel gelockt werden soll.</p>
8	Verhaltensklassen	<p>Das Verhalten von Objekten ist in den Methoden von Klassen definiert, die in externen Dateien abgelegt sind.</p> <p>Welche Klassen sind dies und welches Verhalten wird dort beschrieben?</p>	<p>Schachfiguren werden in der Klasse ChessFigure.ts definiert. Über diese Klasse werden MovementController an die Node angehängt, sowie die Eigenschaften der einzelnen Schachfiguren, wie das entsprechende Movement.</p> <p>Die Kamera wird über die CameraController.ts gesteuert. Der einzelne Spielertyp des Spiels wird über ChessPlayer.ts beschrieben, in ihr befindet sich auch die Liste mit bereits getöteten Schachfiguren.</p> <p>Externe Daten werden über den Singleton DataController.ts erreicht. Über enum.ts, helper.ts und interface.ts werden Helfer Funktionen, eigene Datentypen und eigene Enumerations bestimmt.</p> <p>GameController.ts ist die Zentrale des Spiels, in ihr wird die Welt zusammengesteckt und in der Klasse GameController werden Wechsel der Nutzer und das Beobachten eines Möglichen Schachmatts überwacht.</p> <p>GameObject.ts ist die Parent Klasse für ähnliche Sohnklassen wie die Schachfiguren, und der Auswahlstein für die Schachfiguren sowie der Zielpositionen Punkt.</p> <p>InputController.ts bestimmt das Spielerhandling und mit ihr werden auch die einzelnen Zielpositionen der einzelnen Schachfiguren bestimmt.</p> <p>MovementController.ts ist für das Movement der Schachfiguren zuständig und überprüft ob einzelne Zielpositionen eine Kollision erzeugen. MovementSelection.ts erzeugt das Zielposition Auswahl Objekt als Kind eines jeden Feldes.</p> <p>PlaceController.ts verwaltet jeweils ein Feld auf dem Spielfeld, es weiß welche Schachfigur auf ihr steht. Genauso weiß jede Schachfigur auf</p>

			<p>welchem Feld sie aktuell steht. SelectioControl.ts bestimmt den Auswahlstein über den einzelnen Schachfiguren und ist auch eine Sohnklasse von GameObject.ts. SoundController.ts ist für das Abspielen und Erstellen von Sounds da, es wird an den jeweiligen Knoten angehängt. UIController.ts bestimmt das UI, wer gerade am Zug ist. Das weitere UI wird nicht über die UIController.ts Klasse verwaltet.</p>
9	Subklassen	<p>Es existiert eine Klassenhierarchie, einige Objekte sind Instanzen von einer oder mehreren abgeleiteten Subklassen mit gegenüber den anderen Objekten speziellem Verhalten und besonderen Eigenschaften. Welche Klassen sind dies und welches Verhalten wird dort beschrieben?</p>	<p>ChessFigure.ts, MovementSelection.ts und SelectionController.ts sind Sohnklassen von GameObject.ts, sowie GameObject.ts eine Sohnklasse von f.Node ist. Die Scripte die angehängt werden, also CameraController.ts, MovementController.ts, PlaceController.ts und SoundController.ts sind jeweils alle Sohnklassen von f.ComponentScript.</p>
10	Maße und Positionen	<p>Maße, Skala und Positionen sind gut durchdacht. Wie groß sind Spielfiguren, wie ist die Welt angeordnet bezogen auf den Ursprung, wie sind Spielelemente bezogen auf ihre lokalen Koordinatensysteme definiert?</p>	<p>Jedes Feld auf der Schachbrettplatte ist 1m * 0.2m * 1m groß. Der Untergrund unter den einzelnen Flächen ist 16m * 0.1m * 16m groß. Die Standard- Schachfiguren sind 0.8m * 2m * 0.8m Meter groß und haben das Gewicht 1. Die Bauernfiguren sind allerdings nur 1.5m hoch und haben ebenfalls das Gewicht 1. Der Schachfiguren-Selections-Stein ist 0.7m * 0.5m * 0.7m groß und hat das Gewicht 1. Der Gelbe Punkt ist 0.3m * 0.3m * 0.3m groß und hat kein Gewicht.</p>
11	Event-System	<p>Das Event-System wird verwendet, Wer sendet wem Informationen oder Methodenaufrufe und wofür?</p>	<p>Wählt man eine Zielposition einer Schachfigur aus, so wird ein gelber Punkt auf das entsprechende Feld gepawnt, wechsel ich die Position wird der vorangegangene Punkt über das Event CHILD_APPEND gelöscht. Jedes Script, das an Knoten angehängt wird, wird erst aktiviert, sobald es an den Knoten gehängt wird und dies Geschieht durch das Event: COMPONENT_ADD. Zudem löse ich den Eventlistener auf die Fudge Loop bei Auslösen eines Schachmatts.</p>

