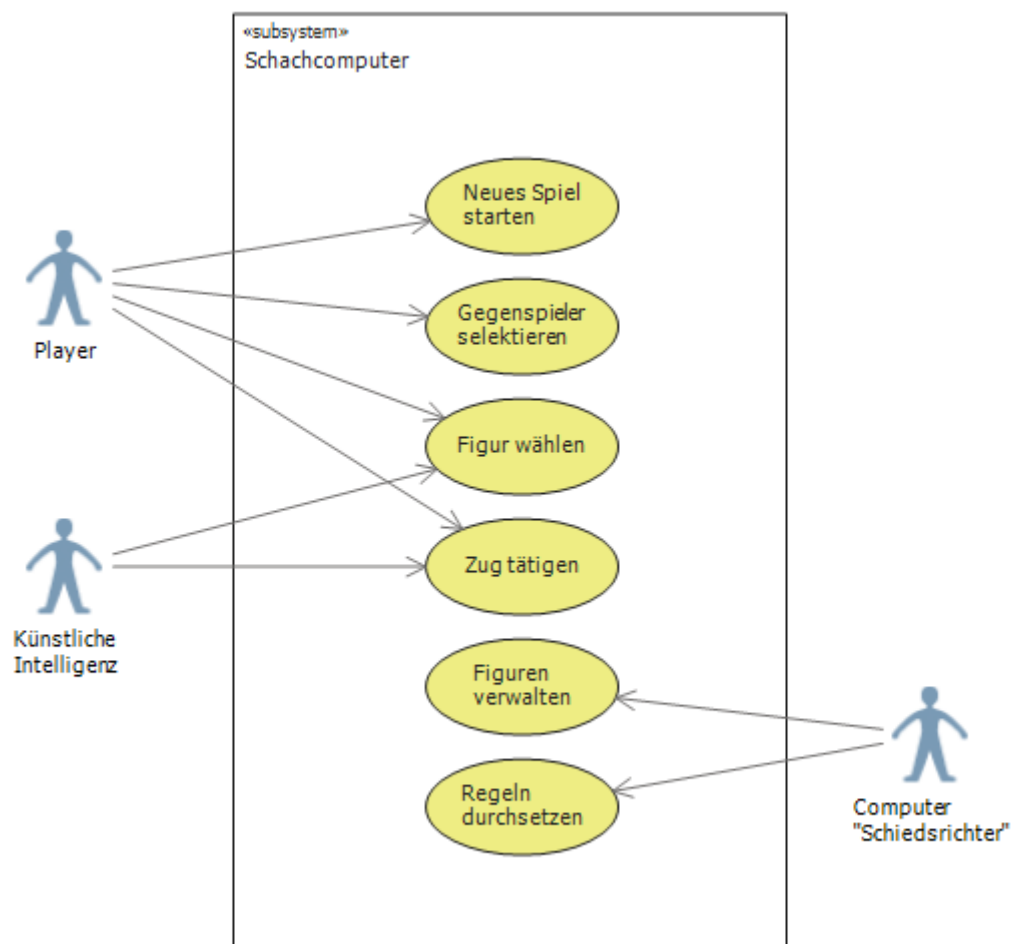


Projektskizze

Idee

Unser Ziel ist ein interessantes Projekt, bei dem jeder Beteiligte dazu lernen kann. Wir entwickeln daher ein Schachspiel. In diesem werden zwei Spielmodi angeboten. Entweder wird gegen einen Menschen gespielt oder gegen den Computer. Dazu kreieren wir eine künstliche Intelligenz, welche die Regeln des Schachspiels kennt und einhält. Der Name des Spiels ist „necaREx“. Es setzt sich zusammen aus „necāre“ (lateinisch für „töten“) und „Rex“ (lateinisch für „König“).

Hauptanwendungsfall



Weitere Anforderungen

- Statistiken über Figurenverlust anzeigen
- Die Masken des Programms werden ansprechend gestaltet. Es soll eine möglichst grosse Usability erreicht werden.
- Die Reaktionszeit des Programms, nicht des Computerspielers, muss unter zwei Sekunden liegen.

Mögliche Erweiterungen / Ideen

Spezielle Schachzüge

- Erreicht ein Bauer das obere Spielfeldende, wählt der Spieler die einzusetzende Figur aus.
- Rochade
- Schlagen „en passant“

Sonstiges

- Verschiedene Computer-Schwierigkeitsstufen
- Spiel zweier Menschen über das Netzwerk

Ressourcen

Für die Realisierung des Projekts wird ein vierköpfiges Team bestehend aus Informatikingenieuren benötigt, von denen einer als Projektleiter zuständig ist. Das Team muss über Fähigkeiten wie Systemarchitektur, Objektorientierte – Programmierung und –Design verfügen.

Da keiner der Teammitglieder grosse Erfahrung mit der Entwicklung von Spielen hat, muss weiteres Wissen für die Realisierung der Software noch angeeignet werden.

Der Gesamtaufwand für das Software Produkt wird auf ca. 250 Personenstunden gerechnet.

Risiken

- Fehlende Erfahrung in der Game-Programmierung
- Auf den Benutzer-Computer könnten die Systemvoraussetzungen nicht gewährleistet sein

Grobplanung

Die Gesamtdauer des Projekts beträgt 14 Wochen, die Planung erfolgt mittels Unified Process mit einer Iterationsdauer von jeweils einer Woche:

Phase	Iteration	Meilenstein	Ziel
Inception	1		Gruppenbildung, Themeneruierung, Wahl des PL
	2	02.10.2012	Projektskizze erstellt, Präsentation erstellt
Elaboration	3		50 % der Anwendungsfälle detailliert ausformuliert, Rest identifiziert und priorisiert
	4		80 % der Anwendungsfälle detailliert ausformuliert, eine erste Architektur wurde erstellt, Zusätzliche Spezifikationen definiert, Anwendungsfalldiagramm erstellt
	5	23.10.2012	System-Sequenzdiagramm und Systemverträge Domänenmodell und Glossar erstellt
Construction	6		20 % der UCs realisiert und getestet, UI Prototyp
	7		40 % der UCs realisiert und getestet
	8	13.11.2012	70 % der UCs realisiert und getestet
	9		85 % der UCs realisiert und getestet
	10		100 % der UCs realisiert und getestet, UI fertig
Transition	11		Systemtest und Dokumentation
	12	11.12.2012	Schlusspräsentation

Kundennutzen

Nachfolgend sind die wichtigsten Kundennutzen aufgeführt:

- In Schachvereinen und Schachkursen kann „necaREx“ als effiziente Lernsoftware eingesetzt werden.
- Durch das schlichte und intuitive Design ist „necaREx“ einfach und schnell zu erlernen.
- Da ein „necaREx“ Lizenzpaket für einen angemessenen Preis erworben werden kann und ein Paket fünf Lizenzen beinhaltet, stellt es eine kostengünstige Investition dar.
- Dank unserer übersichtlichen Benutzeroberfläche kann der Spieler zu jeder Zeit wichtige Informationen dem Spiel entnehmen.
- Durch den modularen Aufbau einfach zu erweitern.

Wirtschaftlichkeit

Wir streben ein Gleichgewicht der Faktoren Qualität, Kosten und Zeit an. Im Fach Software-Entwicklung erhalten wir das notwendige Wissen, um die Qualität garantieren zu können. Der geschätzte Aufwand beträgt insgesamt 250 Personenstunden zu insgesamt 200'000 CHF. Dazu kommen externe Leistungen von 40'000 CHF. Wir rechnen daher mit einem Aufwand von ca. 240'000 CHF. Bei einem Deckungsbeitrag von 20 CHF pro verkauftem Spiel müssen demnach 12'000 Spiele verkauft werden, um die Auslagen der Grundentwicklung zu decken. Nach 20'000 verkauften Spielen hat sich die Investition gelohnt. Dieses Verkaufsvolumen sollte innerhalb von 4 Jahren zu erreichen sein. Am 11.12.2012 wird die finale Version freigegeben.