

Software-Projekt 1

Kick-Off Projekt 3: Siedler



Aufgabe

- Programmieren Sie eine Textversion des Spiels "Siedler von Catan"
- Bemerkungen zu "Siedler von Catan"
 - Regelbuch: https://www.catan.de/sites/default/files/2021-06/CATAN_DasSpiel_Spielregel.pdf
https://www.catan.de/sites/default/files/2021-06/CATAN_DasSpiel_zuguebersicht.pdf
 - Spielphasen
 1. Variabler Spielaufbau (Spielfeld)
 2. Gründungsphase (2 Runden)
 3. Eigentliches Spiel

Spielumfang

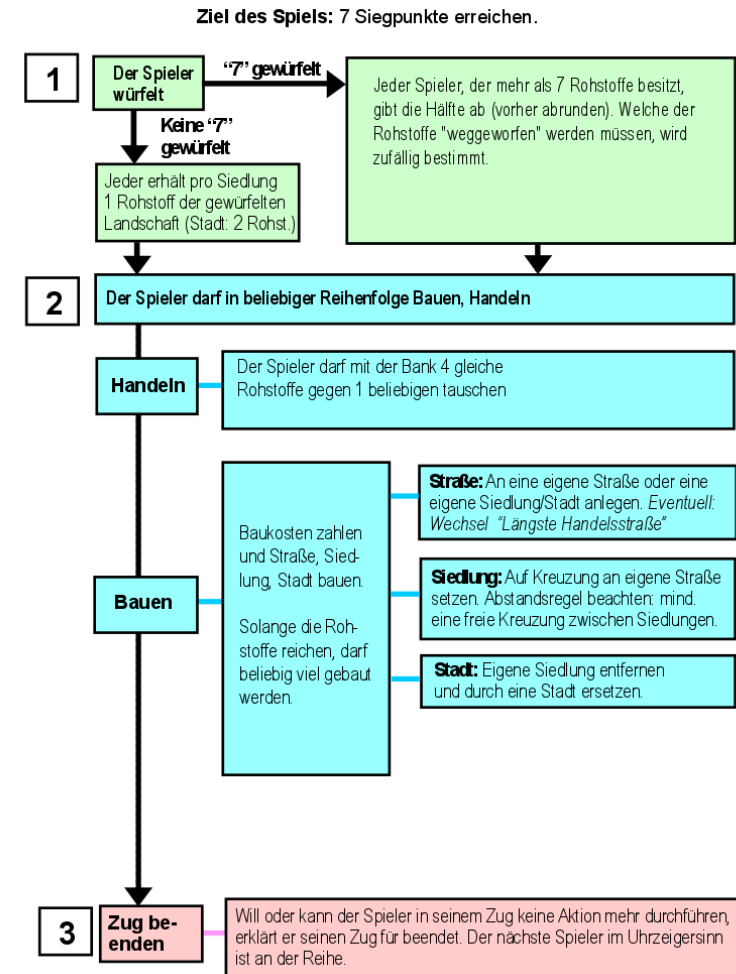
- Grundfunktionalität (18 P.): Gemäss Spielanleitung, ohne:
 - Ereigniskarten
 - Ritter
 - Räuber
 - Handeln mit anderen Spielern
- Zusatzfunktionalität
 - Erweiterung Städte (5 P.)
 - Erweiterungen "Längste Strasse" **oder** "Räuber" (7 P.)

Vereinfachungen Phasen 1 und 2

- Phase 1: Fixer Spielaufbau
- Phase 2
 - Eingabe der Anzahl Spieler [2-4]
 - Jeder Spieler kann der Reihe nach eine seiner Siedlungen platzieren, sowie eine Strasse an die Siedlung anlegen.
 - Nun platzieren nochmals alle Spieler der Reihe nach eine Siedlung plus Strasse, diesmal jedoch in umgekehrter Reihenfolge.
 - Der Spieler, der vorhin zuletzt die Siedlung plus Strasse platziert hat, darf nun zuerst ziehen.
 - Jeder Spieler erhält sofort nach der Gründung seiner zweiten Siedlung seine ersten Rohstoffträge: Für jedes Landfeld, das an diese zweite Siedlung angrenzt, erhält er einen entsprechenden Rohstoff vom Vorrat der Bank

Vereinfachung Phase 3 (Spielphase)

- Der Startspieler (derjenige, der wie oben beschrieben als Letzter seine zweite Siedlung gegründet hat) kommt nun als erster zum Zug.
- Danach folgen die anderen Spieler in fixer Reihenfolge.
- Ein Zug läuft gemäss dem nebenstehend gezeigten Flussdiagramm ab.



- Spielcode
 - Dummy.java
(Demo-Anwendung)
 - Config.java
(Konstanten und Utility-Methoden)
 - SiedlerBoard.java
(erweitert HexBoard)
 - SiedlerBoardTextView.java
(erweitert HexBoardView)
 - SiedlerGame.java
(Grundgerüst Ihres Programms)

- TextIO
- ch.zhaw.hexboard



Umsetzung und Vorgehen

- Erlaubte Konstrukte
 - Alles bis und mit SW12 (d.h. alles bis und mit Kapitel 12 im BlueJ-Buch und den "Exkursen" in der Vorlesung, jedoch keine Strategy Pattern, Stubs, etc. aus Exkursen SW13 und 14)
 - TextIO und java.util(.*)
- Vorgehen
 - Code Download von Moodle
 - Datei ZHAWsiedler-Students.zip
 - Anforderungsanalyse, Entscheidung über Scope
 - Tipp: Optionale Erweiterungen (s.o.) vorsehen, aber zurückstellen
 - Lösungsansatz und Klassenmodell entwickeln, Verantwortlichkeiten verteilen
 - Umsetzung

Bewertung

- All-or-nothing
 - Lauffähiges Programm
 - Alle beteiligt
- Teilaufgabe 1: Grundfunktionalität (18 P.)
 - Funktionalität
 - Clean Code (komplette Sammlung) und erlaubte Konstrukte (s.o.) eingehalten
 - Sinnvolle Aufteilung in Klassen, Dokumentation
 - Test Cases definiert und ausgeführt, Dokumentation
- Teilaufgaben 2/3: Erweiterte Funktionalität (5 + 7 P.)
 - Mittels Vererbung realisierte Funktionalität für Städte (5 P.)
 - Spiellogik für "längste Strasse" **oder** Spiellogik für "Räuber" (7 P.)

The image displays a 22x14 grid with a complex, symmetrical pattern of symbols and text. The grid is divided into four quadrants by a central vertical line at column 7 and a central horizontal line at row 11. The symbols include various characters like '||', '()', '~', and 'WD', along with alphanumeric codes like '06', '03', '05', '10', '09', '11', '12'. The pattern is highly symmetrical and repetitive, suggesting a structured data layout or a specific encoding scheme.