

Programmieraufgabe

1 AUFGABENSTELLUNG

Gegeben sei nachfolgendes Bild eines Sudoku-Spieles, das nachfolgend Schritt für Schritt umgesetzt werden soll. Es gibt für verschiedene Aspekte dieser Problemstellung unterschiedliche Punkte. Orientieren Sie sich an den Punktezahlen.

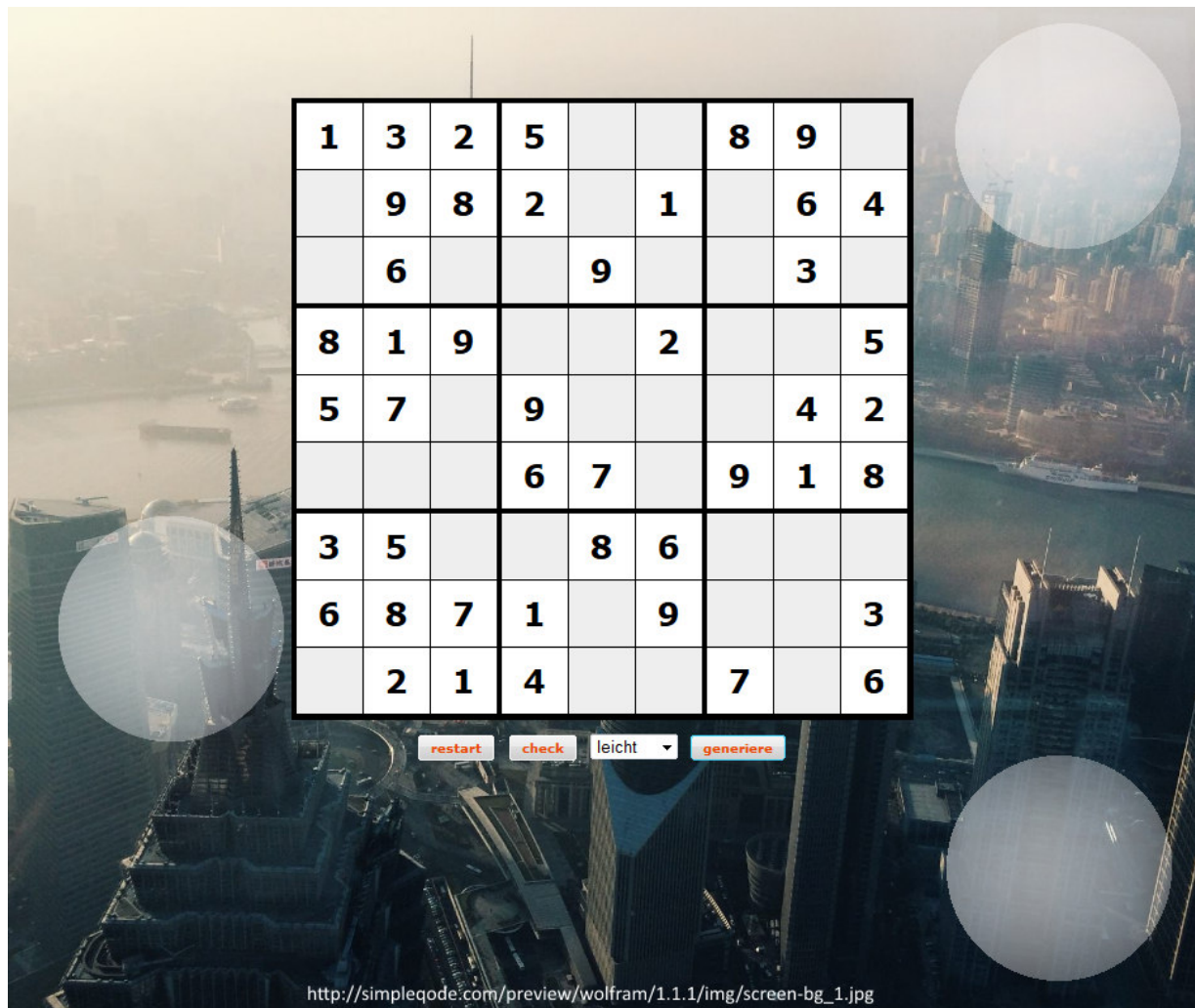


Abbildung 1

1.1 RAHMEN

Das Sudoku-Spiel soll mithilfe von HTML(5), CSS(3) und ASP.Net MVC 4 umgesetzt werden. Die Verwendung jeglicher Javascript-Bibliotheken ist gestattet und sogar erwünscht. Entwickeln Sie einen Rahmen mit einem Hintergrundbild und dem Sudoku-Brett. **(10 Punkte)**

1.2 ANIMATION

Die in Abbildung 1 dargestellten transparenten Kreise sollen initial zufällig auf der Fläche verteilt werden und sich anschließend langsam und „treibend“ im Hintergrund bewegen (lineare Bewegung ist ausreichend). Anschließend sollen sie nach einer bestimmten Zeitspanne ausgeblendet und an einer anderen Stelle wieder eingeblendet werden. **(20 Punkte)**

1.3 SUDOKU-BRETT UND USER INTERFACE

Entwickeln Sie eine Javascript-Klasse, welche in ein bestehendes DIV-Element das in der Abbildung dargestellte Sudoku-Brett „rendert“. Beispiel:

```
<div class="sudoku">  
  
</div>
```

Neben dem Sudoku-Brett sollen ebenfalls auch die in der Abbildung eingezeichneten UI-Elemente (Buttons für „restart“, „check“ und „generiere“ sowie die ComboBox für die Schwierigkeitsstufe) durch die Javascript-Klasse erstellt werden. Sofern der Benutzer auf eines dieser Button drückt, soll eine Operation der Javascript-Klasse (als Callback) aufgerufen werden. Trennen Sie Model- und View-Teile ihrer Implementierung. Überlegen Sie sich eine geeignete Datenstruktur, um das Spielfeld und die in ihr dargestellten Zahlen aufzunehmen. Die View soll das Model darstellen...

Um das Problem zu vereinfachen, können Sie zunächst das Sudoku-Feld wie in der Abbildung dargestellt initialisieren. Klickt der Benutzer auf ein freies Feld, kann er eine Zahl eingeben oder seine bereits erfolgte Eingabe rückgängig machen.

(30 Punkte)

1.4 IMPLEMENTIERUNG DER OPERATIONEN „RESTART“ UND „CHECK“

Das aktuelle Brett soll beim Klick auf „Restart“ zurückgesetzt werden. Alle Eingaben des Benutzers werden dabei gelöscht. Das Spielfeld wird aber nicht neu generiert. Die Operation „Check“ überprüft, ob das Sudoku-Spiel gemäß den Regeln korrekt ausgefüllt wurde. **(10 Punkte)**

1.5 ENTWICKELN SIE EINEN ALGORITHMUS ZUR GENERIERUNG LÖSBARER SUDOKU-SPIELFELDER

Es gibt mehrere Algorithmen zur Generierung neuer Spielfelder. Entwickeln Sie entweder einen eigenen Algorithmus oder implementieren Sie in der Operation „Generiere“ ein bestehendes Verfahren. **(30 Punkte)**