

1	4	7	2	5	8	3	6	9
2	5	8	3	6	9	4	7	1
3	6	9	4	7	1	5	8	2
4	S	U	D	O	K	U	9	3
5	8	2	6	9	3	7	1	4
6	9	3	7	A	P	P	2	5
7	1	4	8	2	5	9	3	6
8	2	5	9	3	6	1	4	7
9	3	6	1	4	7	2	5	8

Robert Spindler & Tobias Zinke

# Gliederung

- Idee
- Funktionsweise
- Arbeitsaufteilung
- Herausforderungen
- Praktische Vorführung
- Future Features
- Fazit

# Idee (Aufgabenstellung)

Erstellen einer funktionstüchtigen Sudoku-App:

1. Generation eines neuen Sudokus in der App
  2. Anzeige von falsch platzierten Eingaben
  3. Benachrichtigung beim Gewinnen
  4. Steigerung der Schwierigkeit
- Option, ein neues Sudoku zu generieren

# Wie man ein Sudoku generiert

1. Grid mit Zahlen füllen
2. Grid shuffeln
3. zufällige Zahlen löschen (je schwieriger desto mehr)

1	4	7	2	5	8	3	6	9
2	5	8	3	6	9	4	7	1
3	6	9	4	7	1	5	8	2
4	7	1	5	8	2	6	9	3
5	8	2	6	9	3	7	1	4
6	9	3	7	1	4	8	2	5
7	1	4	8	2	5	9	3	6
8	2	5	9	3	6	1	4	7
9	3	6	1	4	7	2	5	8

1	4	7	2	5	8	3	6	9
2	5	8	3	6	9	4	7	1
3	6	9	4	7	1	5	8	2
4	7	1	5	8	2	6	9	3
5	8	2	6	9	3	7	1	4
6	9	3	7	1	4	8	2	5
7	1	4	8	2	5	9	3	6
8	2	5	9	3	6	1	4	7
9	3	6	1	4	7	2	5	8

1	4	7	2	5	8	3	6	9
2	5	8	3	6	9	4	7	1
3	6	9	4	7	1	5	8	2
4	7	1	5	8	2	6	9	3
5	8	2	6	9	3	7	1	4
6	9	3	7	1	4	8	2	5
7	1	4	8	2	5	9	3	6
8	2	5	9	3	6	1	4	7
9	3	6	1	4	7	2	5	8

1	4	7	2	5	8	3	6	9
2	5	8	3	6	9	4	7	1
3	6	9	4	7	1	5	8	2
4	7	1	5	8	2	6	9	3
5	8	2	6	9	3	7	1	4
6	9	3	7	1	4	8	2	5
7	1	4	8	2	5	9	3	6
8	2	5	9	3	6	1	4	7
9	3	6	1	4	7	2	5	8

1	4	7	2	5	8	3	6	9
2	5	8	3	6	9	4	7	1
3	6	9	4	7	1	5	8	2
4	7	1	5	8	2	6	9	3
5	8	2	6	9	3	7	1	4
6	9	3	7	1	4	8	2	5
7	1	4	8	2	5	9	3	6
8	2	5	9	3	6	1	4	7
9	3	6	1	4	7	2	5	8

1		7					6	
		8			9	4		1
	6	9				5	8	2
	7				2		9	3
5			6			7		
			7					5
7				2				6
8			9		6			7
		6		4			5	

# Wie man ein Sudoku lösbar macht

1. generiertes Sudoku laden
2. jede Zahl in jeweils 3 Arrays eintragen  
(Zeile, Spalte, 3x3 Block)
3. beim Drücken eines Sudoku-Buttons die  
ausgewählte Zahl eintragen
4. prüfen

		5			6			1
	9							2
4	1					6		3
1	7		8			3		
	5	2	6	9		1		7
		3	7		4	2		
		8					1	
			4	7			2	5
7				8				6

# Wie man ein Sudoku prüft

1. jedes Array (Zeile, Spalte, Block) auf Duplikate prüfen
2. Duplikate hervorheben
3. überprüfen, ob Sudoku fertig gefüllt ist

		5			6			1
	9							2
4	1					6		3
1	7		8		6	3		
	5	2	6	9		1		7
		3	7		4	2		
		8					1	
			4	7			2	5
7				8				6

# Wie man ein Sudoku darstellt

1. Button zum resetten
2. 9x9 Grid aus Buttons
3. RadioGroup aus 9 Buttons zur Zahlenwahl

Sudoku

Round 1 easy

		5			6			1
	9							2
4	1					6		3
1	7		8		6	3		
	5	2	6	9		1		7
		3	7		4	2		
		8					1	
			4	7			2	5
7				8				6

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☒ 6 ☐ 7 ☐ 8 ☐ 9

# Aufgabenverteilung

## Robert

- Verbindung von Code und visuellen Elementen (MainActivity)
- Sudoku Fehlerprüfung

## Gemeinsam

- To-Do Liste
- Optimierung der Kommunikation aller Klassen
- Präsentation

## Tobias

- Basislayout der App (ActivityMain)
- Sudoku Generation



# Herausforderungen (& Schwerpunkte)

- Sudoku-Generation
  - Mischen der Zahlen
  - Beibehalten der Lösbarkeit
- Prüfen auf Fehler im Sudoku
  - Duplikate im Array finden
  - Position im 9x9 Grid anhand der Position im Array ermitteln

		5			6			1
	9							2
4	1					6		3
1	7		8		6	3		
	5	2	6	9		1		7
		3	7		4	2		
		8					1	
			4	7			2	5
7				8				6

# Praktische Vorführung



# Future Features

- Speichern und Laden implementieren um durch Neustart der App keinen Fortschritt mehr zu verlieren
- bessere UI
- Frei wählbarer Schwierigkeitsgrad
- Löschen eines Feldes
- Kandidaten-Notation
- Frei wählbare Grid-Größe

# Fazit (Lessons Learned)

- Früher mehr XML nutzen
- Bessere Unterscheidung der Zahlen (eigene, generierte)
- Zahlen als Button zu realisieren ist zwar einfach, jedoch nicht sehr flexibel

Vielen Dank für Ihre Aufmerksamkeit