

Inhalt

Dank — V

Vorwort — VII

1 Einleitung — 1

- 1.1 Cross-Plattform Entwicklung — 2
- 1.2 Motivation — 2
- 1.3 Zielgruppe — 3
- 1.4 Aufbau und Ziel des Buches — 3

Teil I: Grundlagen

2 Grundlagen von Dart — 7

- 2.1 Die Sprache Dart — 7
 - 2.1.1 Mobile und Web — 8
 - 2.1.2 Installation — 8
- 2.2 Die Dart Syntax — 9
 - 2.2.1 Datentypen — 9
 - 2.2.2 Kontrollfluss — 15
 - 2.2.3 Ausnahmebehandlung — 18
 - 2.2.4 Objektorientierte Programmierung — 19
 - 2.2.5 Kürzere Schreibweisen — 21
 - 2.2.6 Benannte Konstruktoren, optionale Parameter, Fat-Arrow — 22
 - 2.2.7 Immutable, Annotationen — 25
 - 2.2.8 Pakete und Abhängigkeiten — 26
 - 2.2.9 Factory Konstruktor und Initialisierungsliste — 28
 - 2.2.10 Vererbung, Mixins und Interfaces — 29
 - 2.2.11 Dynamische Erweiterung von Typen — 31
 - 2.2.12 Generische Klassen — 32
 - 2.2.13 Typparameter — 32
 - 2.2.14 Beschränkte parametrische Polymorphie — 33
 - 2.2.15 Asynchrone Operationen — 35
 - 2.2.16 Generatoren: Iteratoren und Streams — 36

3 Werkzeuge — 40

- 3.1 Installation — 40
 - 3.1.1 Zusätzliche Software — 41
 - 3.1.2 Freie Software, Ressourcen und Lizenzen — 41

3.1.3	Installation des Flutter Frameworks —	43
3.1.4	Installation von Android Studio —	45
3.1.5	Installation von Visual Studio Code —	45
3.1.6	Projekte anlegen und starten mit Android Studio —	50
3.1.7	Projekte anlegen und starten mit Visual Studio Code —	52
3.1.8	Projekte anlegen und debuggen über die Kommando-Zeile —	53
3.1.9	Dev-Tools —	53
3.2	Tiled-Editor —	54
3.3	Rive und Nima —	54
4	Grundlagen von Flutter —	59
4.1	Warum Flutter? —	59
4.2	Material Design und mehr —	60
4.3	Flutter Layouts —	62
4.3.1	Alles ist ein Widget —	63
4.3.2	Layout-Beispiel ohne State —	65
4.3.3	Der Navigator —	67
4.3.4	Ein Layout mit State —	72
4.4	Flutter Packages —	78
4.4.1	Die Nutzung einer Map-Erweiterung —	80
4.4.2	Flutter Erweiterungen selbst programmieren —	84
4.4.3	Einfache Animationen —	86
4.4.4	Animationen mit Nima und Rive —	90
4.4.5	Eine animierte Backdrop-Komponente —	94
4.5	Automatisierte Tests —	99

Teil II: Praxis

5	Cloud-Anwendung —	105
5.1	Google Firebase —	105
5.1.1	Einrichtung und Setup —	105
5.1.2	Firestore —	107
5.1.3	Messenger —	109
5.1.4	Google Play Games Services —	112
6	Chicken Maze —	119
6.1	Überblick —	119
6.1.1	Technische Besonderheiten —	119
6.2	Konzeption —	123
6.3	Der Code im Detail —	123
6.3.1	Start und Initialisierungen —	127

6.3.2	Lokalisierung —	130
6.3.3	Die Main-Funktion und Routes —	132
6.4	Die Screens —	134
6.4.1	Die About-Page —	135
6.4.2	Der eigene Look —	136
6.4.3	Das Ausklappenmenü —	138
6.4.4	Die Start-Page —	140
6.4.5	Die Settings-Page —	143
6.4.6	Die Pause-Page —	148
6.4.7	Die Game Over-Page —	150
6.4.8	Die Highscore-Page —	154
6.5	Server-Abfrage —	158
6.5.1	Client-Seite —	158
6.5.2	Server-Seite —	161
6.6	Die Spiele-Logik —	166
6.6.1	Die Spiele-Seite —	166
6.6.2	Das Chicken-Game —	167
6.6.3	Der Inputhandler —	176
6.6.4	Die Spielfiguren —	180
6.6.5	Das Chicken —	182
6.6.6	Die Feinde —	185
6.6.7	Das Labyrinth —	190
6.6.8	Hilfsklassen —	194

7 Deployment — 204

7.1	Releases —	204
7.1.1	Icons erstellen —	204
7.1.2	Signieren bei Android —	205
7.1.3	Build-Prozess für Android —	206
7.2	Google Play Verwaltung —	207
7.2.1	Signieren im Play-Store —	207
7.2.2	Store-Eintrag —	209

8 Zusammenfassung — 212

8.1	Ausblick —	212
-----	------------	------------

Literatur — 215

Abbildungsverzeichnis — 217

Stichwortverzeichnis — 219

