1. Record Nr. UNISA996504671903316

Autore	PRYNNE, William <1600-1669.>
Titolo	Concordia discors, or, The dissonant harmony of sacred public oaths, protestations, leagues, covenants, ingagements lately taken by many time-serving saints, officers without scruple of conscience : ... or a discourse tending to prove that the oaths of supremacy and allegiance do in direct words extend not only to the King's person, but his heirs and successors ... / by William Prynne, Esq
Pubbl/distr/stampa	London, : Printed for Edw. Thomas in 1659, and now reprinted by Samuel Lowndes and are to be sold by R. Davis, 1683
Descrizione fisica	Testo elettronico (PDF) (45 p.)
Disciplina	941.5
Soggetti	Gran Bretagna Storia
Lingua di pubblicazione	Inglese
Formato	Risorsa elettronica
Livello bibliografico	Monografia
Note generali	Riproduzione dell'originale nella Huntington Library

2. Record Nr. UNINA9910568257303321

Autore	He M. Holmes
Titolo	Creating apps with React Native : deliver cross-platform 0 crash, 5 star apps / / M. Holmes He
Pubbl/distr/stampa	Berkeley, California : , : Apress, , [2022] ©2022
ISBN	1-4842-8042-3
Descrizione fisica	1 online resource (445 pages) : illustrations
Disciplina	005.35
Soggetti	Mobile apps - Development Cross-platform software development
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di bibliografia	Includes index.
Nota di contenuto	Intro -- Table of Contents -- About the Author -- About the Technical Reviewer -- The Path to a 05 App -- Chapter 1: Start Thinking in React -- 1.1 Component -- 1.1.1 Key Takeaways -- 1.2 The "Hello World" App in Pieces -- 1.2.1 React Native Development Environment -- 1.2.2 JSX -- 1.2.3 props -- 1.2.3.1 Style -- 1.2.3.2 Children -- 1.2.4 JSX Internals -- 1.2.5 States -- 1.2.5.1 State Change on the Current Component -- 1.2.5.2 Cascading State Changes -- 1.2.6 setState() Internals -- 1.2.7 Key Takeaways -- 1.3 Summary -- Chapter 2: Foundations of React -- 2.1 Flexbox, a Practical Guide -- 2.1.1 Component Size -- 2.1.2 Case Study: Feed -- 2.1.3 Key Takeaways -- 2.2 Composition vs. Inheritance, HOC -- 2.2.1 Case Study: Multiple Photo Feeds -- 2.2.2 Key Takeaways -- 2.3 ScrollView and FlatList -- 2.3.1 Case Study: Moment -- 2.3.2 Key Takeaways -- 2.4 Error Handling -- 2.4.1 Case Study: Moment (Reinforced) -- 2.4.2 Key Takeaways -- 2.5 Summary -- Chapter 3: Animation in React Native -- 3.1 Introduction to React Native Animation -- 3.2 Layout Animation -- 3.2.1 Presets -- 3.2.2 LayoutAnimation.create() -- 3.2.3 Raw Animation Config -- 3.2.4 Android -- 3.2.5 Case Study, Read More -- 3.2.6 Key Takeaways -- 3.3 Value Animation -- 3.3.1 Animate the Animation -- 3.3.1.1 Animated.timing() -- 3.3.1.2 Animated. spring() -- 3.3.1.3 Animation Cohort -- 3.3.1.4 setValue() -- 3.3.2 Bind the Animation Value -- 3.3.2.1 The transform props.style --

of a Native Call -- 6.5.4 Execute the Bundle -- 6.5.5 The Two-Way Communication -- 6.5.6 The Native Module Metadata -- 6.5.7 Wrap Up -- 6.6 Animation Inside Out -- 6.6.1 Establish the Animated Node Graph -- 6.6.1.1 JavaScript Pass -- 6.6.1.2 Native Pass -- 6.6.2 Bind the Event Receiver -- 6.6.3 Attach the Event Source -- 6.6.4 Native Event Transmission.
6.6.4.1 Identify Receivers -- 6.6.4.2 Update -- 6.7 Adaptive to All Screens, Layout Design -- 6.8 Time to Say Goodbye -- Index.

Sommario/riassunto

Produce high-quality, cross-platform apps with user experiences almost identical to pure native apps. When evaluating cross-platform frameworks, developers make an assumption that quality will be compromised. But that doesn't have to be true. The principles in this book will show you how to meet quality expectations both from engineering and consumer standpoints. Youll also realize the ideal of a greater front end. That means your whole front-end team, including app side and web side, will be optimized. The shared knowledge base as well as mobilization potential give more flexibility and strength in all front-end facets without the need of increasing team sizes. The market has seen a large amount of high quality React Native apps and successful stories about them. Nevertheless, under optimized apps and unsuccessful stories shadow. The fundamental difference between the two opposing groups is understanding. Discover the critical points in the React and React Native architecture, and develop general best practices that can lead to consistently developing 0 crash, 5 star apps based on an understanding of fundamentals. You will: Measure and define successful app design Create animation based on user need Reduce performance bottleneck throughout your apps.