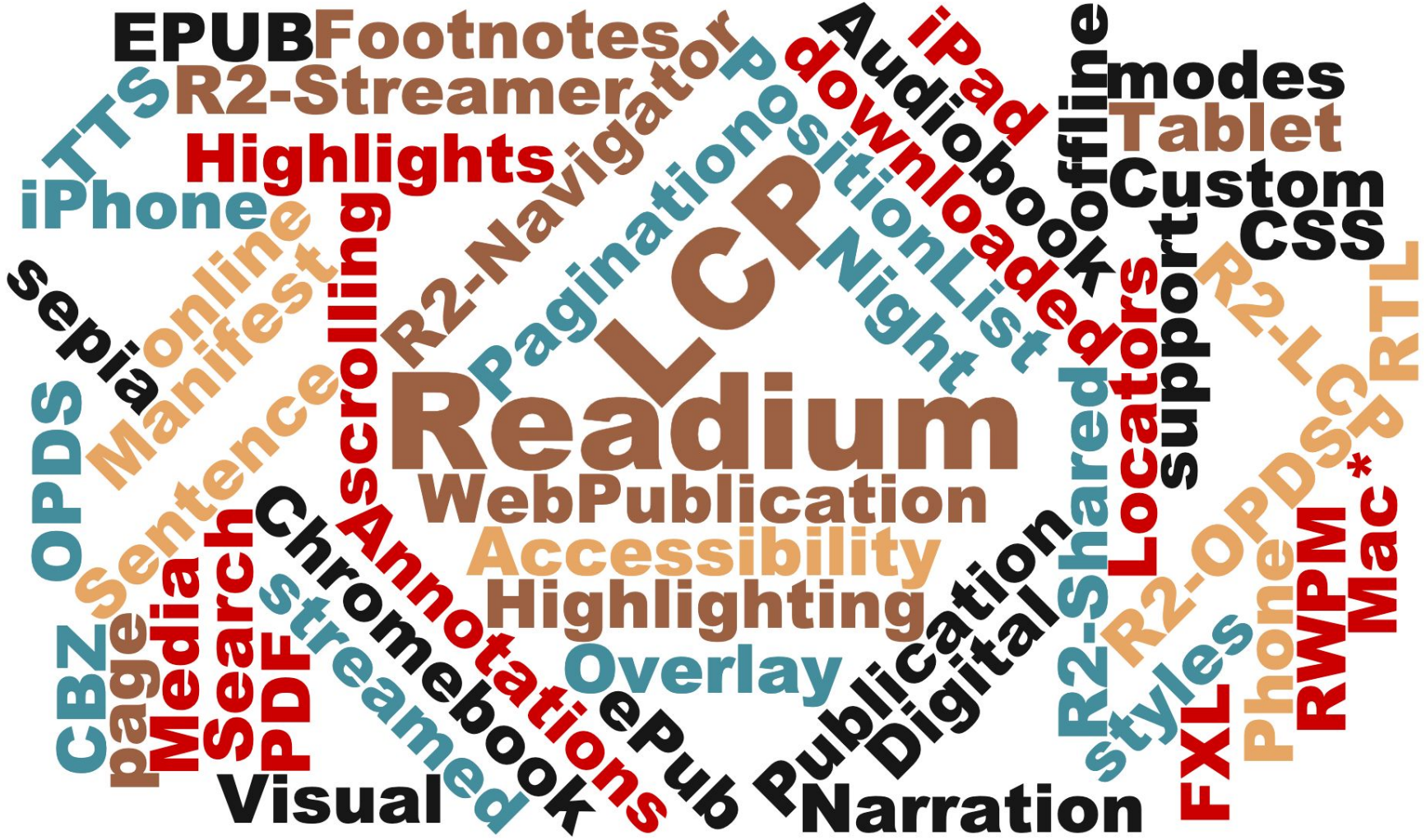
Digital Publishing Summit

2019
June 25-26

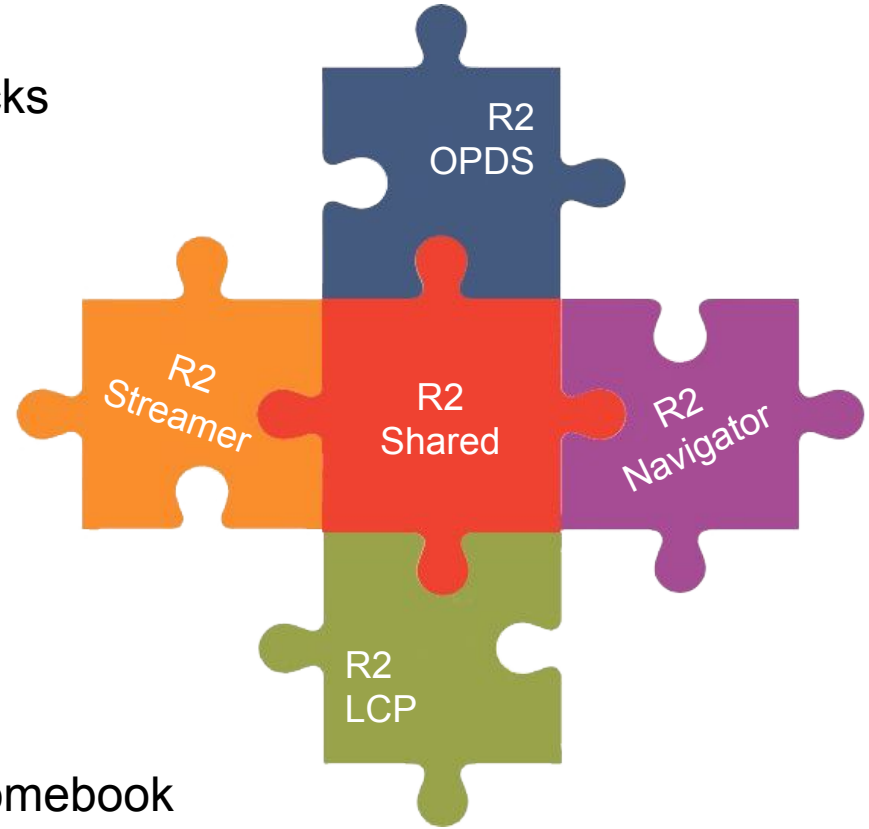
Readium Mobile: Deployment

- Aferdita Muriqi


EPUB
Footnotes
R2-Streamer
TTS
Highlights
iPhone
R2-Navigator
Pagination
Position
LCP
Audiobook
iPad
downloaded
offline
modes
Tablet
Custom
CSS
Night
List
Locators
support
R2-LCP
RTL
sepia
online
Manifest
Scrolling
Sentence
OPDS
Readium
WebPublication
Accessibility
Highlighting
Overlay
Annotations
Chromebook
streamed
ePub
Publication
Digital
R2-Shared
R2-OPDS
Mac*
RWPM
Phone
styles
FXL
Narration
Visual
CBZ
page
Media
Search
PDF

Building Blocks

- Toolkit of 5 core modules / building blocks
- R2 Testapp (sample and test purpose)
- R2 Workspace (getting started quickly)
- Native Environment
  - Xcode 10.1 moving to 10.2
  - \Swift 4.2 moving to 5
  - iOS (min target 9.0)
  - Android Studio 3.4.1
  - Kotlin 1.3.31
  - Android (min sdk 21)



iPhone & iPad,
Android Phone, Tablet & Chromebook

Readium CSS

- Essential building block of R2 iOS, Android and desktop apps (can also be used in any reading system!)
- Some key features:
  - Pagination: Right-To-Left (e.g. Arabic), Vertical (e.g. Japanese)
  - Custom fonts (e.g. dyslexia)
  - Color themes (e.g. dark/night, sepia)
  - Text alignment, line height, word spacing, etc.
- https://github.com/readium/readium-css

http://bit.ly/readium-css-2018

Readium
“Web Publication Manifest”

- Optimized interface between:
  - “Server / backend” (produces JSON)
  - “Client / frontend” (consumes JSON)
- “Stream” publications in smaller chunks (e.g. individual chapters)
- Web features: HTTP caching, prefetching, integrity checking, etc.

http://bit.ly/readium-web-pub-manifest-2018

Where are we?

Mobile - Where are we?

	R2 Swift	R2 Kotlin
ePub 2 & 3 support	✔	✔
Custom Styling	✔	✔
CBZ support	✔	✔
PDF support	✔	
Readium Web Publication Manifest	✔	✔
RWPM EPUB	✔	✔
RWPM Audiobook		✔
OPDS 1 & 2 support	✔	✔

✔ - Done ✔ - In Progress

Mobile - Where are we? (cont.)

	R2 Swift	R2 Kotlin
Readium LCP ePub	✔	✔
FXL support (1 page)	✔	✔
FXL support (2 page)		✔
RTL support	✔	✔
Locators	✔	✔
Accessibility	✔	✔
Popup Footnotes		✔
TTS with Overlay		✔

✔ - Done ✔ - In Progress

So, is it ready yet?

Mobile - Stable in Production (github.com/readium)

	R2 Swift	R2 Kotlin
R2-Shared	1.2.13	1.0.10
R2-Streamer	1.0.14	1.0.10
R2-Navigator	1.0.13	1.0.10
R2-OPDS	1.0.10	1.0.9
R2-LCP	1.0.10	1.0.10
R2 Test App	V2.0.0-beta.7	V2.0.0-beta.6
Public Beta	http://bit.ly/r2-ios-testing	http://bit.ly/r2-android-testing

Where are we going next?

Mobile - Where are we going?

	R2 Swift	R2 Kotlin
RWPM Audiobook	✔	✔
FXL support (2 page)	✔	✔
Popup Footnotes	✔	✔
PDF support	✔	✔
Readium LCP PDF	✔	✔
PositionList	✔	✔
Annotations (Highlights)	✔	✔
RWPM Digital Visual Narratives	✔	✔

✔ - Done ✔ - In Progress ✔ - Planned

Mobile - Where are we going? (cont.)

	R2 Swift	R2 Kotlin
Search	✔️	✔️
TTS with Sentence Highlighting	✔️	✔️
Media Overlay	✔️	✔️
More Accessibility	✔️	✔️

✔️ - In Progress ✔️ - Planned

Mobile - Use Cases (github.com/readium)

	R2-Shared	R2-Streamer	R2-Navigator	R2-OPDS	R2-LCP
I want to read books located on my device offline	✔	✔	✔		
I want to read ePubs that are LCP protected	✔	✔	✔		✔
I want to read books that are not on my device but streamed from a remote server	✔		✔		
I want to download books from an OPDS feed and read them offline	✔	✔	✔	✔	
I want to listen to an audiobook that is provided through an OPDS feed and streamed from a remote server	✔		✔	✔	
I want to read a pdf that is LCP protected	✔	✔	✔		✔
I want to just browse an OPDS catalog	✔			✔	

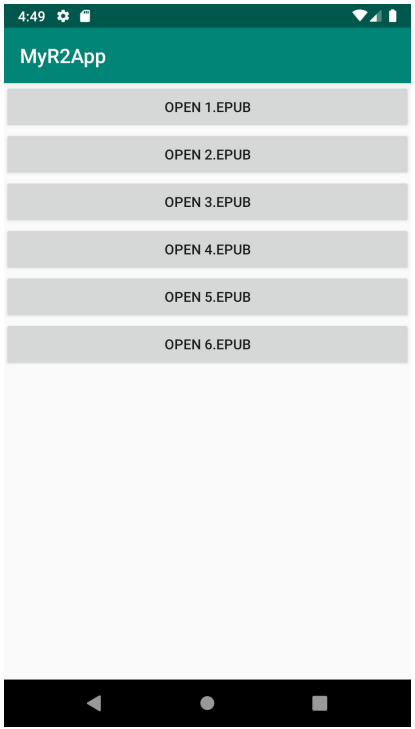
✔ - modules you need ✔ - modules that are included in other modules

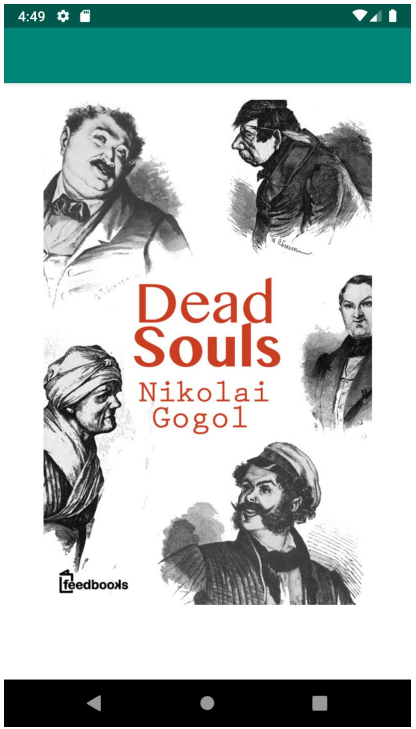
Mobile - Native Platforms

R2 Swift -> iOS, iPadOS?, MacOS?			
	iPhone	✔	
	iPad	✔	
	Mac ?	✔	
R2 Kotlin -> Android			
	Phone	✔	
	Tablet	✔	
	Chromebook	✔	

✔ - fully supported ✔ - needs better keyboard support ✔ - TBD

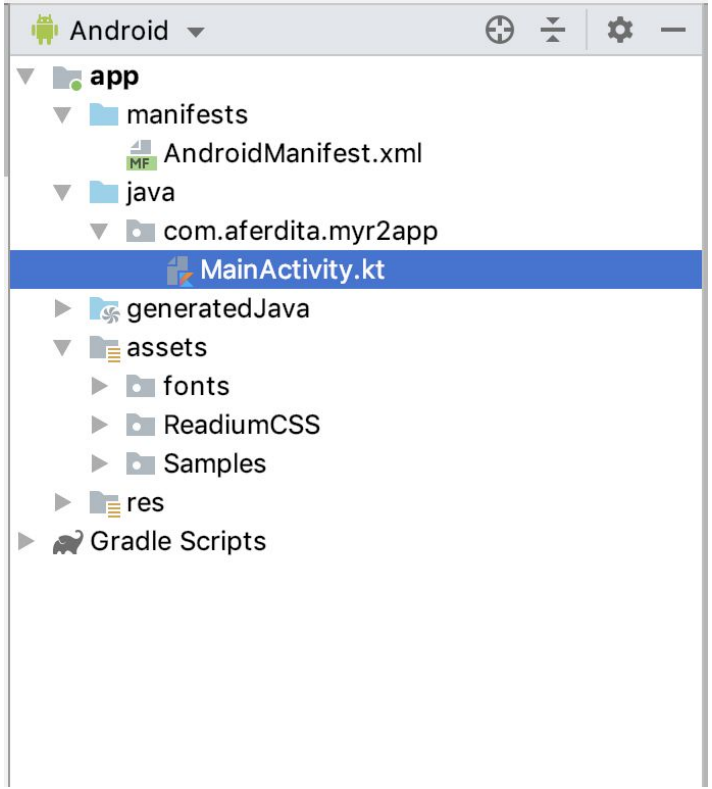
Mobile - How to? (Kotlin example)





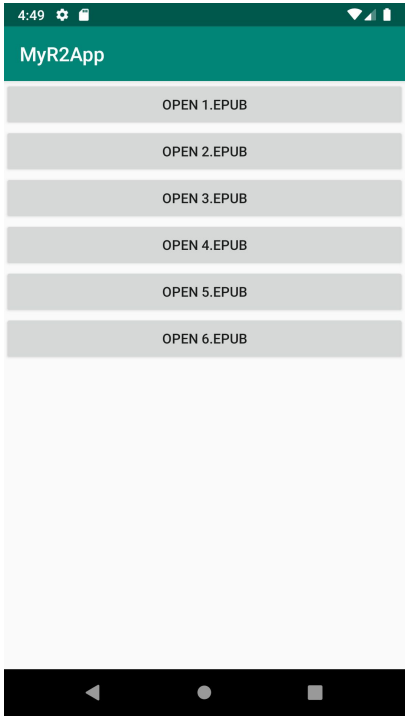
?

Mobile - How to? (Kotlin example)



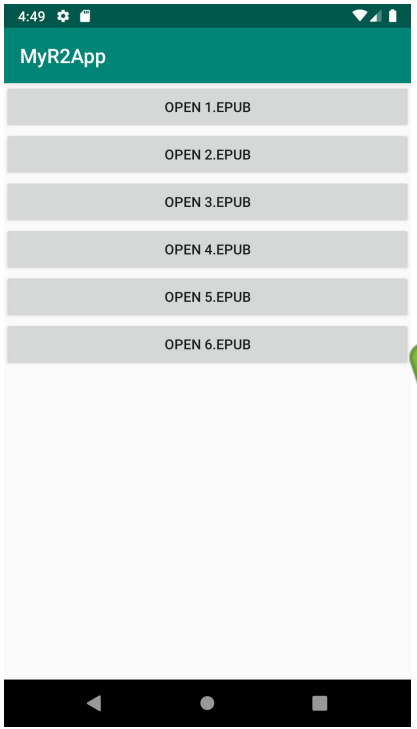

```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk7:$kotlin_version"
    implementation 'androidx.appcompat:appcompat:1.0.2'
    implementation 'androidx.core:core-ktx:1.0.2'
    implementation 'com.google.android.material:material:1.0.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'


    // Readium R2 Modules
    implementation "com.github.readium:r2-shared-kotlin:1.0.10"
    implementation "com.github.readium:r2-navigator-kotlin:1.0.10"
    implementation "com.github.readium:r2-streamer-kotlin:1.0.10"


    implementation 'org.nanohttpd:nanohttpd:2.3.2-SNAPSHOT'
    implementation 'org.nanohttpd:nanohttpd-nanolets:2.3.2-SNAPSHOT'


    // Anko
    implementation "org.jetbrains.anko:anko-commons:0.10.8"
    implementation "org.jetbrains.anko:anko-sdk25:0.10.8"
    implementation "org.jetbrains.anko:anko-appcompat-v7:0.10.8"
    implementation 'org.jetbrains.anko:anko-design:0.10.8'
    implementation 'org.jetbrains.anko:anko-sdk25-coroutines:0.10.8'
    implementation 'org.jetbrains.anko:anko-appcompat-v7-coroutines:0.10.8'

}
```

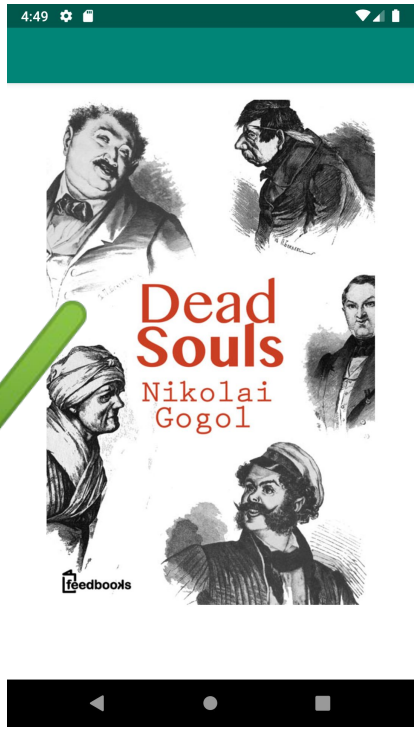
Mobile - How to? (Kotlin example)

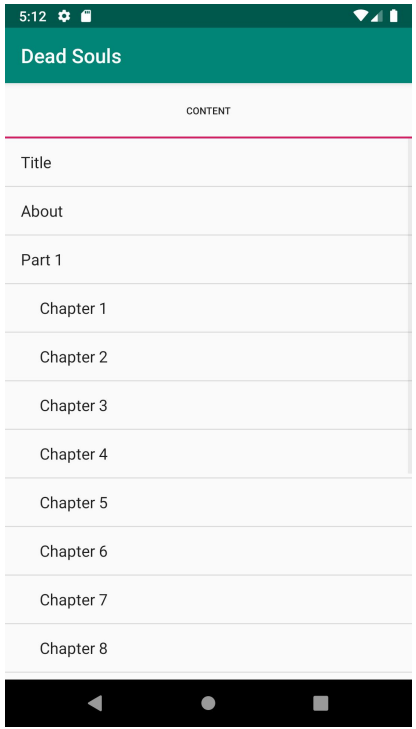


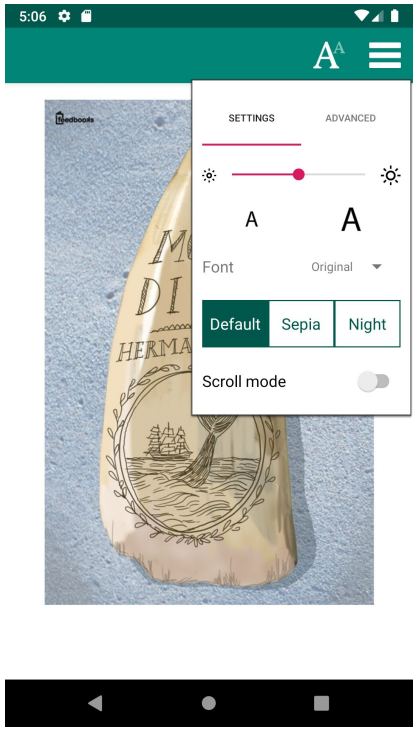

```
class MainActivity : AppCompatActivity() {

    private lateinit var server: Server
    private var localPort: Int = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        val s = ServerSocket( port: 0)
        s.localPort
        s.close()

        localPort = s.localPort
        server = Server(localPort)

        linearLayout { this: _LinearLayout
            orientation = LinearLayout.VERTICAL
            assets.list( path: "Samples")?.filter { it.endsWith( suffix: ".epub") }?.let { list : List<String!> ->
                for (element : String! in list) {
                    val input : InputStream = assets.open( fileName: "Samples/$element")
                    val fileName : String = UUID.randomUUID().toString()
                    val publicationPath : String = appPath + "/" + fileName
                    input.toFile(publicationPath)
                    button { this: Button
                        text = "open $element"
                        setOnClickListener { it: View!
                            val parser = EpubParser()
                            val pub : PubBox? = parser.parse(publicationPath)
                            pub?.let { it: PubBox
                                server.addEpub( pub.publication, pub.container, fileName: "/$fileName", userPropertiesPath: appPath + "/styles/UserProperties.json"
                                )
                                startActivity(
                                    intentFor<EpubActivity>( ...params: "publicationPath" to publicationPath, "epubName" to fileName, "publication" to pub.publication )
                                )
                            }
                        }
                    }
                }
            }
        }
    }
}
```

Mobile - How to? (Kotlin example)







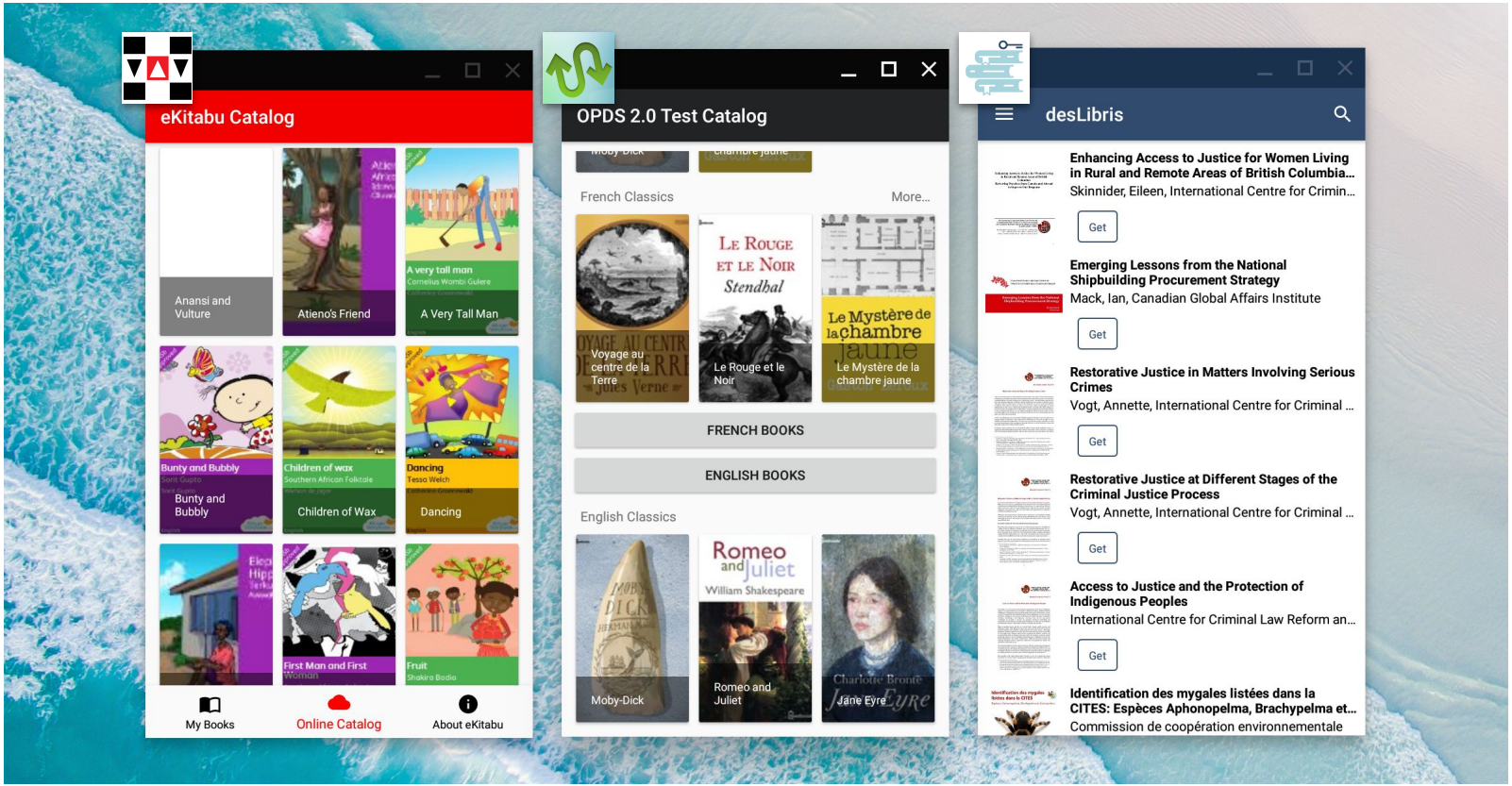


R2 Reader “test apps”

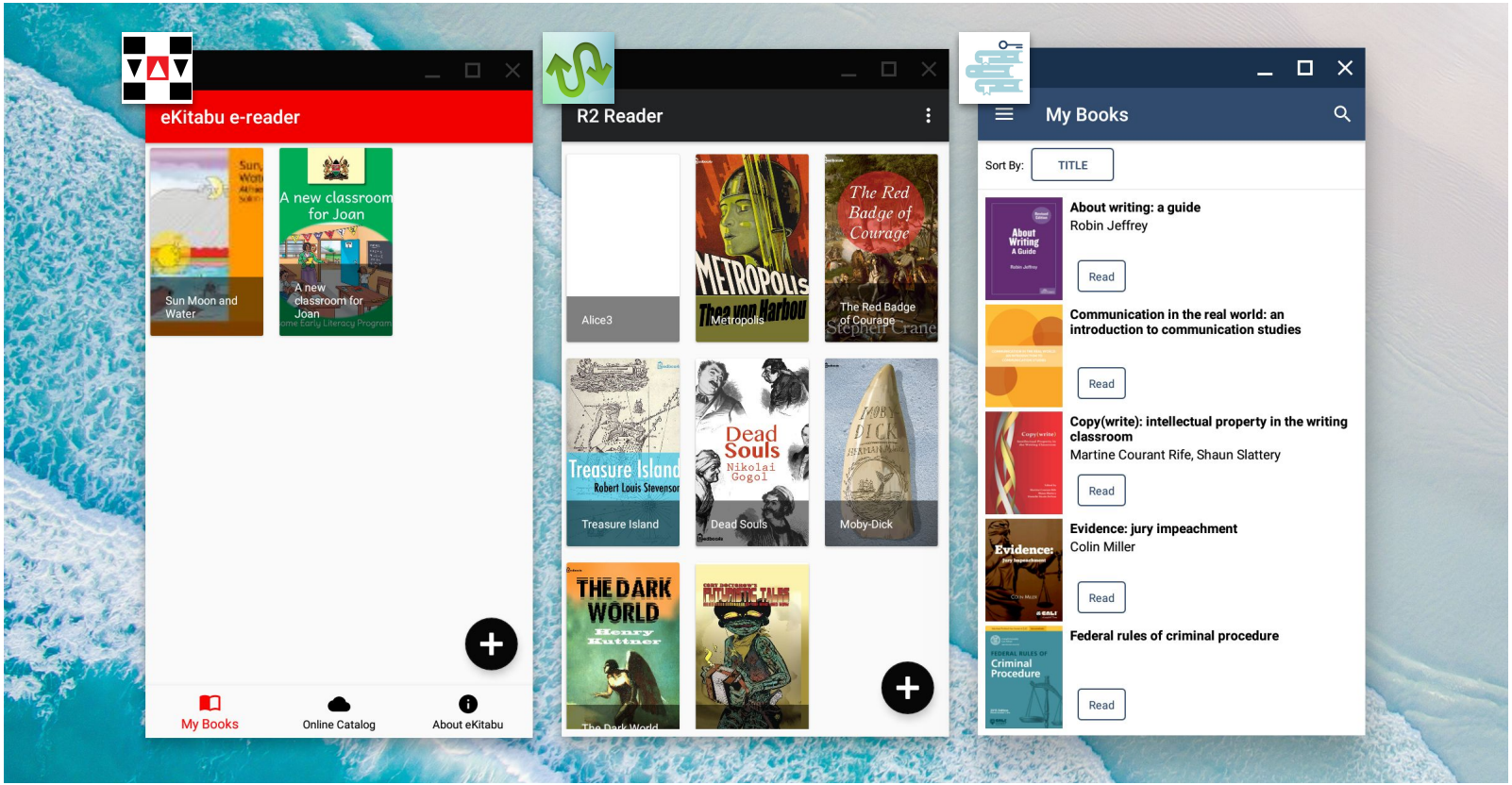
- Distributed via official stores, but only for tests / demos!
- Fluid user experience (instant rendering, fast page turns)
- Accessible user interface
- Minimal bookshelf functionality (basic OPDS browser)
- Minimal reader features (e.g. no shared annotations, dictionary, etc.)

Goal: facilitate the integration of R2 building blocks into publisher apps, vendor ecosystem, etc.

Mobile - Implementations (OPDS)

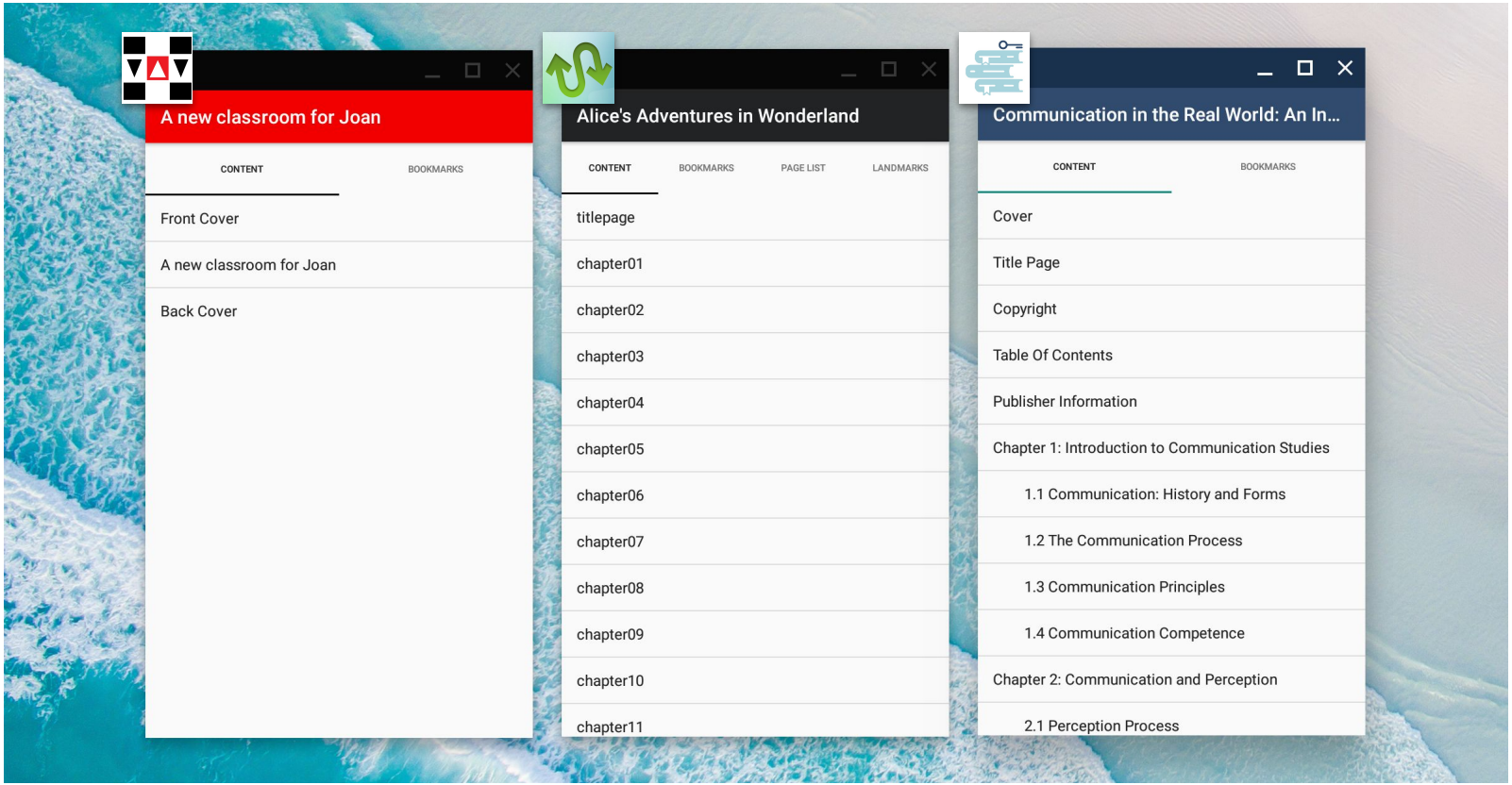


Mobile - Implementations (bookshelf)

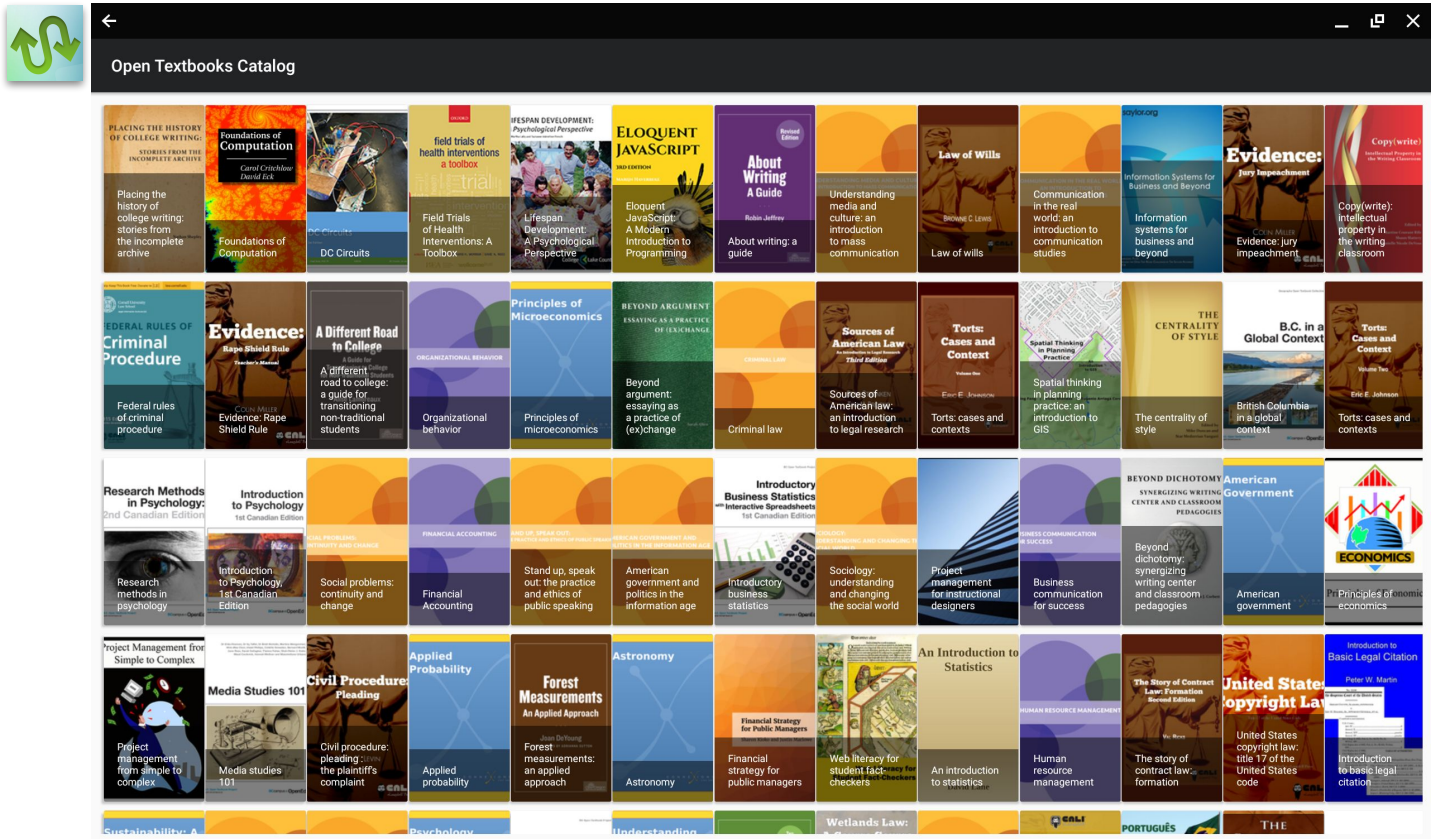


Mobile - Implementations (Navigator)

Mobile - Implementations (Navigator)



Mobile - Implementations (OPDS)



Mobile - Implementations (ePub reflowable)

Chapter 2

The next morning the youth discovered that his tall comrade had been the fast-flying messenger of a mistake. There was much scoffing at the latter by those who had yesterday been firm adherents of his views, and there was even a little sneering by men who had never believed the rumor. The tall one fought with a man from Chatfield Corners and beat him severely.

The youth felt, however, that his problem was in no wise lifted from him. There was, on the contrary, an irritating prolongation. The tale had created in him a great concern for himself. Now, with the newborn question in his mind, he was compelled to sink back into his old place as part of a blue demonstration.

For days he made ceaseless calculations, but they were all wondrously unsatisfactory. He found that he could establish nothing. He finally concluded that the only way to prove himself was to go into the blaze, and then figuratively to watch his legs to discover their merits and faults. He reluctantly admitted that he could not sit still and with a mental slate and pencil derive an answer. To gain it, he must have blaze, blood, and danger, even as a chemist requires this, that, and the other. So he fretted for an opportunity.

Meanwhile he continually tried to measure himself by his comrades. The tall soldier, for one, gave him some assurance. This man's serene unconcern dealt him a measure of confidence, for he had known him since childhood, and from his intimate knowledge he did not see how he could be capable of anything that was beyond him, the youth. Still, he thought that his comrade might

obscurity, but, in reality, made to shine in war.

The youth would have liked to have discovered another who suspected himself. A sympathetic comparison of mental notes would have been a joy to him.

He occasionally tried to fathom a comrade with seductive sentences. He looked about to find men in the proper mood. All attempts failed to bring forth any statement which looked in any way like a confession to those doubts which he privately acknowledged in himself. He was afraid to make an open declaration of his concern, because he dreaded to place some unscrupulous confidant upon the high plane of the unconfessed from which elevation he could be derided.

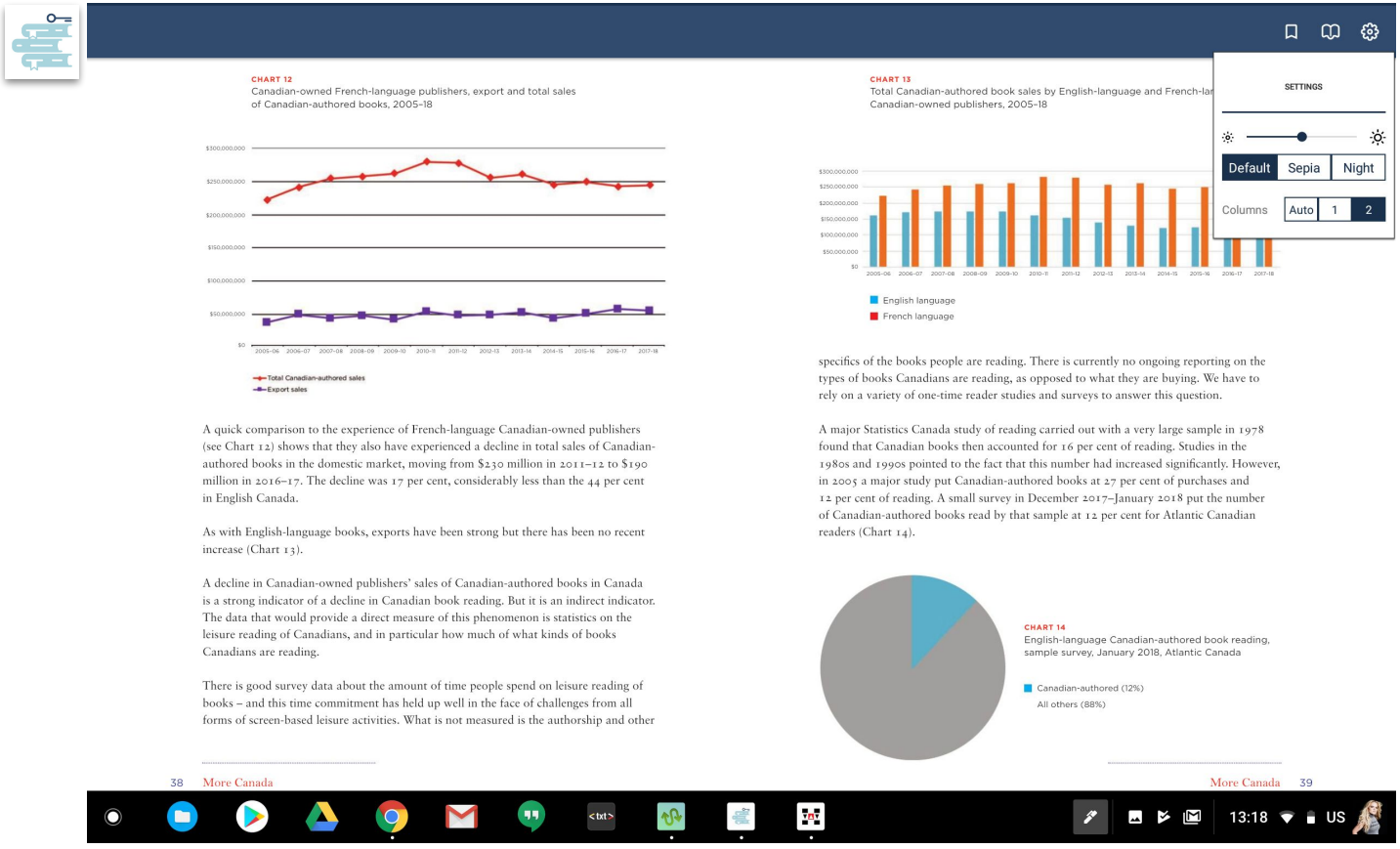
In regard to his companions his mind wavered between two opinions, according to his mood. Sometimes he inclined to believing them all heroes. In fact, he usually admitted in secret the superior development of the higher qualities in others. He could conceive of men going very insignificantly about the world bearing a load of courage unseen, and although he had known many of his comrades through boyhood, he began to fear that his judgment of them had been blind. Then, in other moments, he flouted these theories, and assured himself that his fellows were all privately wondering and quaking.

His emotions made him feel strange in the presence of men who talked excitedly of a prospective battle as of a drama they were about to witness, with nothing but eagerness and curiosity apparent in their faces. It was often that he suspected them to be liars.

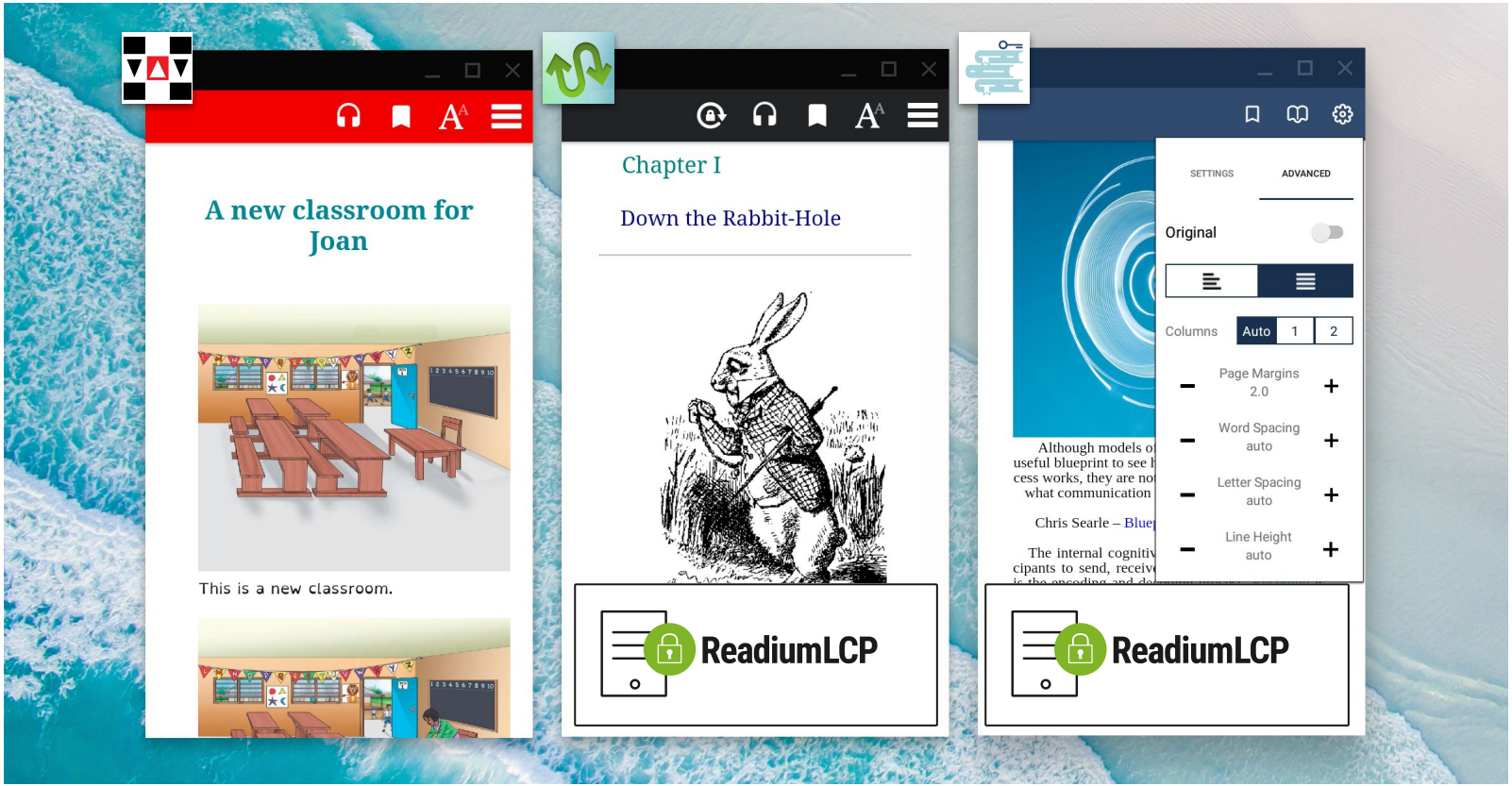
He did not pass such thoughts without severe condemnation of himself. He dinned reproaches at times. He was convicted by himself of many shameful crimes against the gods of traditions.

In his great anxiety his heart was continually clamoring at what he considered the intolerable slowness of the generals. They seemed content to perch tranquilly

Mobile - Implementations (ePub fixed)



Mobile - Implementations (LCP)



Thank you!

- GitHub.com/readium
  - r2-<module>-swift, r2-<module>-kotlin
  - r2-testapp-swift, r2-testapp-kotlin
- … get involved! :)
- https://www.edrlab.org

Questions?

DEMO:

iOS: https://github.com/readium/r2-testapp-swift
Android: https://github.com/readium/r2-testapp-kotlin