

#WWDC18

Introducing ClassKit

Session 215

Pavel Dudrenov, ClassKit team

Marin Eubanks, ClassKit team

General Overview

Framework Classes

Adopting ClassKit

General Overview

Framework Classes

Adopting ClassKit

General Overview

Framework Classes

Adopting ClassKit

General Overview

ClassKit Framework

Features

ClassKit Framework

Features

Focused on education

ClassKit Framework

Features

Focused on education

Define assignable content in your app

ClassKit Framework

Features

Focused on education

Define assignable content in your app

Record and report student progress

ClassKit Framework

Features

Focused on education

Define assignable content in your app

Record and report student progress

Respects student privacy

ClassKit Framework

Reasons to adopt

ClassKit Framework

Reasons to adopt

Provide better teacher workflow

ClassKit Framework

Reasons to adopt

Provide better teacher workflow

Provide insight for teachers

ClassKit Framework

Reasons to adopt

Provide better teacher workflow

Provide insight for teachers

Enables personalized instruction

ClassKit Framework

Reasons to adopt

Provide better teacher workflow

Provide insight for teachers

Enables personalized instruction

Competitive advantage

Apple School Manager

Admin portal

The screenshot displays the Apple School Manager Admin Portal interface. The browser address bar shows `school.apple.com`. The user is logged in as Grant Phelps. The left sidebar contains navigation options: Institution, Activity (with a notification badge), Locations, People, Accounts, Classes, Roles, Devices, MDM Servers, Device Assignments, Assignment History, Content, Apps and Books, and iTunes U. The main content area is divided into Personal Settings and Institution Settings. The 'Managed Apple IDs' option is selected and highlighted in blue, showing 23,223 Managed Apple IDs. Other settings include My Profile (Grant Phelps), Apps and Books, Enrollment Information (Township Schools), Data Source (PowerSchool), and Devices Purchases (2,225 Devices). The right-hand panel is titled 'Managed Apple IDs' and shows 23,223 Managed Apple IDs. It includes fields for Domain (townshipschools.k12.edu), Student Format (S ID Number(full) @townshipschools.k12.edu), and Instructor and Staff Format (First Name . Last Name @townshipschools.k12.edu). There are 'Enable' buttons for 'Student Progress' (currently Disabled) and 'FaceTime & Messages' (currently Disabled). A tooltip for 'Student Progress' explains that it allows instructors to view student progress on activities they assign in Schoolwork and provides a 'Learn More' link.

Apple School Manager

Admin portal

Create managed Apple IDs

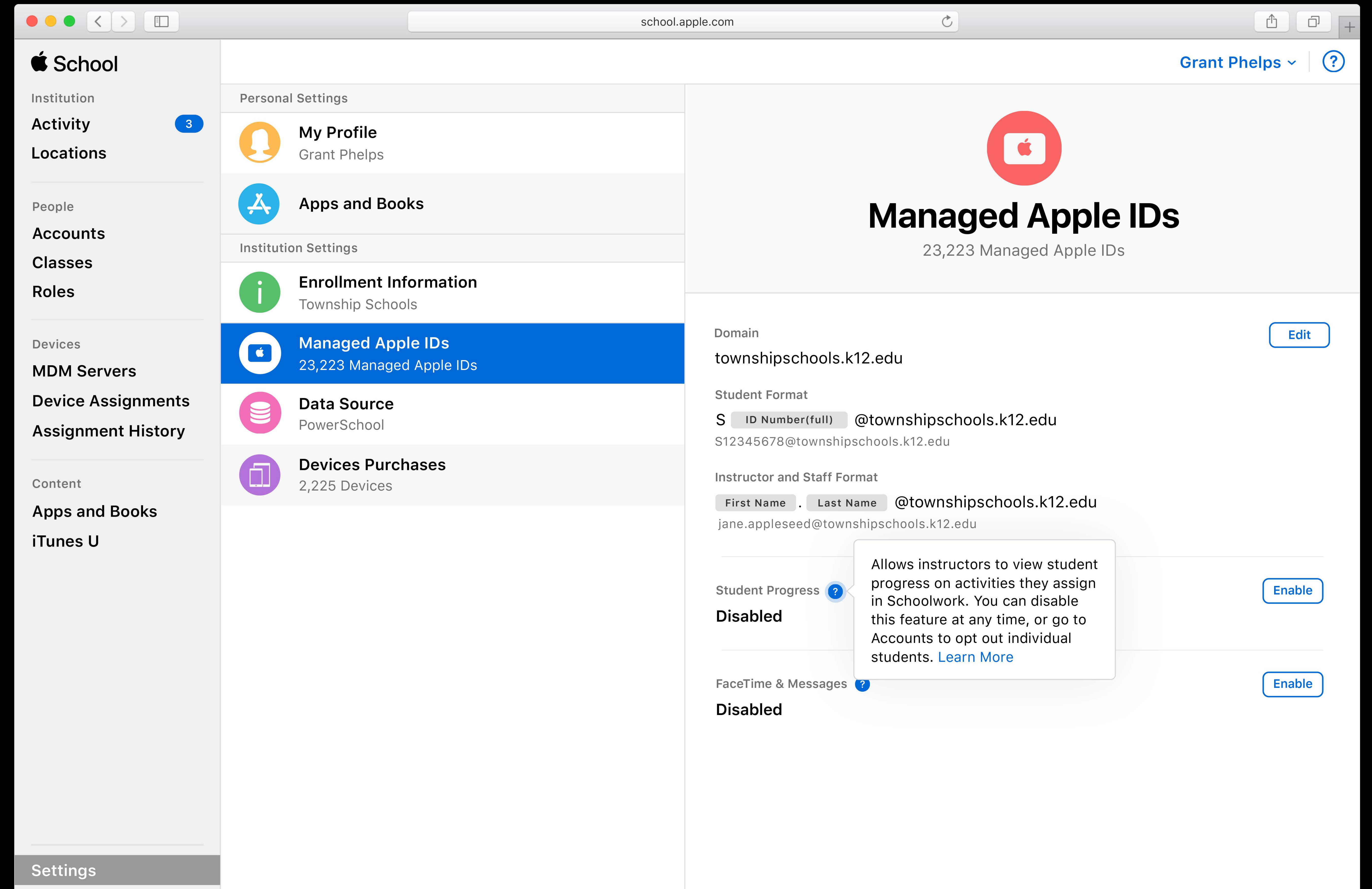
The screenshot displays the Apple School Manager Admin Portal interface. The browser address bar shows 'school.apple.com'. The user is logged in as 'Grant Phelps'. The left sidebar contains navigation options: Institution, Activity (with a notification badge '3'), Locations, People, Accounts, Classes, Roles, Devices, MDM Servers, Device Assignments, Assignment History, Content, Apps and Books, and iTunes U. The main content area is divided into 'Personal Settings' (My Profile, Apps and Books) and 'Institution Settings' (Enrollment Information, Managed Apple IDs, Data Source, Devices Purchases). The 'Managed Apple IDs' section is selected, showing 23,223 Managed Apple IDs. The right-hand configuration panel includes: Domain (townshipschools.k12.edu), Student Format (S ID Number(full) @townshipschools.k12.edu), Instructor and Staff Format (First Name . Last Name @townshipschools.k12.edu), Student Progress (Disabled), and FaceTime & Messages (Disabled). A tooltip for 'Student Progress' explains that it allows instructors to view student progress on activities assigned in Schoolwork and can be disabled at any time.

Apple School Manager

Admin portal

Create managed Apple IDs

Create rosters



Apple School Manager

Admin portal

Create managed Apple IDs

Create rosters

Manage content and devices

The screenshot displays the Apple School Manager Admin Portal interface. The browser address bar shows 'school.apple.com'. The user is logged in as 'Grant Phelps'. The left sidebar contains a navigation menu with categories: Institution (Activity, Locations), People (Accounts, Classes, Roles), Devices (MDM Servers, Device Assignments, Assignment History), and Content (Apps and Books, iTunes U). The 'Managed Apple IDs' option is selected in the 'Institution Settings' section, showing '23,223 Managed Apple IDs'. The main content area is titled 'Managed Apple IDs' with a sub-header '23,223 Managed Apple IDs'. Below this, there are settings for: Domain (townshipschools.k12.edu), Student Format (S ID Number(full) @townshipschools.k12.edu), and Instructor and Staff Format (First Name . Last Name @townshipschools.k12.edu). There are 'Enable' buttons for 'Student Progress' and 'FaceTime & Messages', both currently set to 'Disabled'. A tooltip for 'Student Progress' explains that it allows instructors to view student progress on activities assigned in Schoolwork.

Apple School Manager

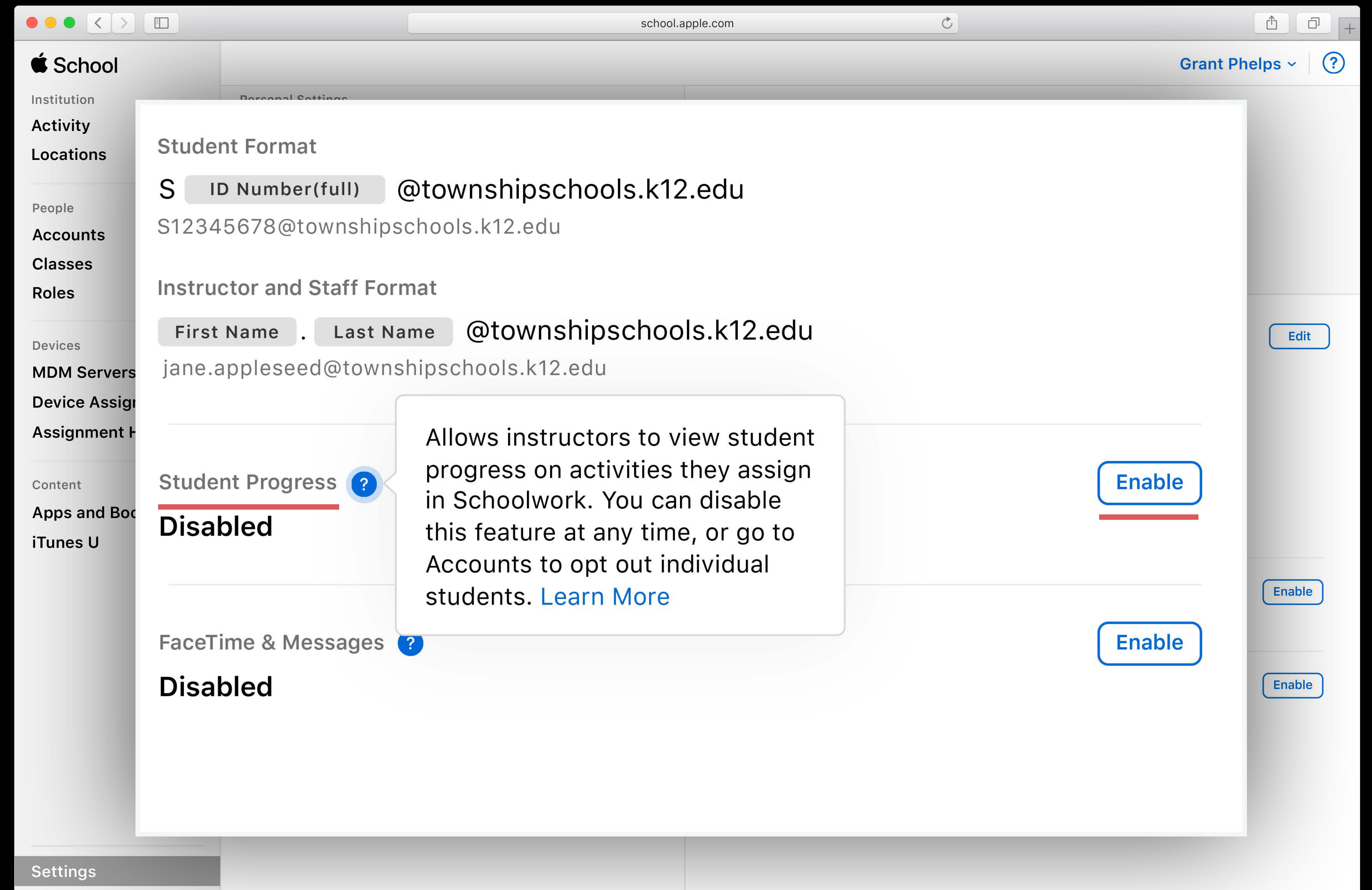
Admin portal

Create managed Apple IDs

Create rosters

Manage content and devices

Opt in for progress reporting



Apple School Manager

Admin portal

Create managed Apple IDs

Create rosters

Manage content and devices

Opt in for progress reporting

Purchase apps in bulk

The screenshot displays the Apple School Manager Admin Portal interface. On the left is a navigation sidebar with categories: Institution, Activity, Locations, People, Accounts, Classes, Roles, Devices, MDM Servers, Device Assignments, Assignment History, Content, Apps and Books, iTunes U, and Settings. The main content area shows a list of apps with columns for app name, developer, price, and usage statistics. The 'Swift Playground' app is highlighted in blue. To the right, the details for 'Swift Playgrounds' are shown, including its icon, developer (Apple Inc.), and a 'Buy Licenses' section with a form for selecting license type, quantity, and price.

App Name	Developer	Price	In-Use	Available
Classroom	Apple, Inc · iOS	FREE	65	15
Clips	Apple, Inc · iOS	FREE	20	50
Keynote	Apple, Inc · iOS	FREE	150	50
Pages	Apple, Inc · iOS	\$4.99	6	3
Swift Playground	Apple, Inc · iPad	FREE	185	21
Swift Playgrounds: Learn...	Apple, Inc · iBook	FREE	6	10
Explain Everything™	Explain Everything sp. z o.o. · iPad	FREE	68	0
Kahoot!	Kahoot! AS · iOS	FREE	126	33
Lexia Reading Core5	Lexia Learning · iPad	FREE	9	6
Molecules	Sunset Lake Software · iOS	FREE	14	22
Incredible Numbers			82	

Swift Playgrounds
Apple Inc · iPad
✓ Device-Assignable ✓ Works with Schoolwork
✓ Optimized for Shared iPad
[View Details](#)

Buy Licenses

License Type: Search Locations:

Price	Quantity	Available Store Credit
FREE	<input type="text" value="0"/>	\$1000.00 Redeem Store Credit

Total Cost: **\$0** [Buy](#)

Apple School Manager

Admin portal

Create managed Apple IDs

Create rosters

Manage content and devices

Opt in for progress reporting

Purchase apps in bulk

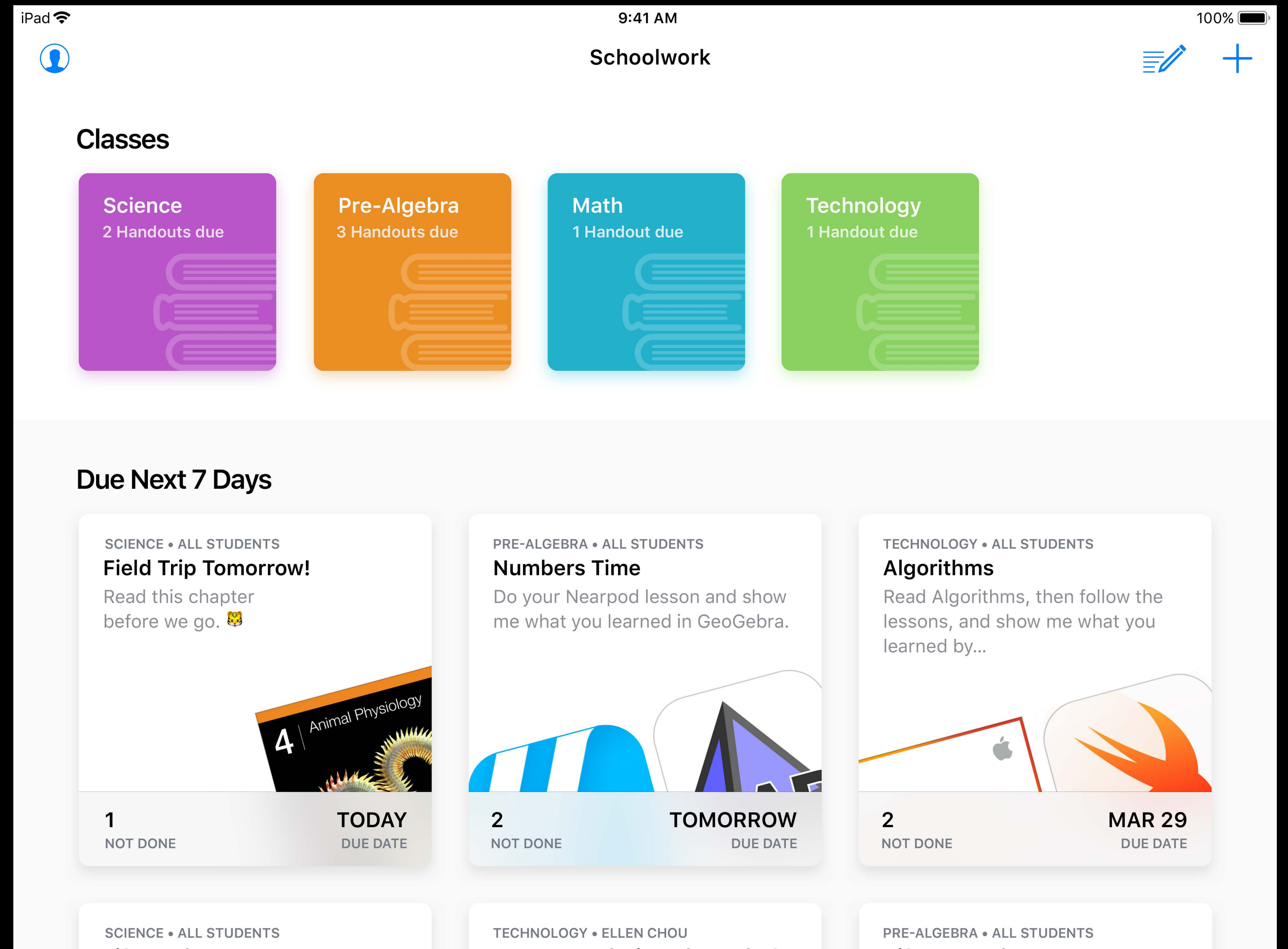
The screenshot shows the Apple School Manager admin portal interface. At the top, there is a search bar and a user profile for Grant Phelps. The main content area displays the details for the 'Swift Playgrounds' app by Apple Inc. for iPad. The app is marked as 'Device-Assignable' and 'Optimized for Shared iPad'. A red checkmark indicates it 'Works with Schoolwork'. A 'View Details' button is visible. Below this, a table lists other apps and their usage statistics:

App Name	Developer	In-Use	Available	Price
Swift Playgrounds: Learn...	Apple, Inc · iBook	6	10	FREE
Explain Everything™	Explain Everything sp. z o.o. · iPad	68	0	FREE
Kahoot!	Kahoot! AS · iOS	126	33	FREE
Lexia Reading Core5	Lexia Learning · iPad	9	6	FREE
Molecules	Sunset Lake Software · iOS	14	22	FREE
Incredible Numbers		82		

At the bottom right, there is a 'Total Cost \$0' and a 'Buy' button.

Schoolwork

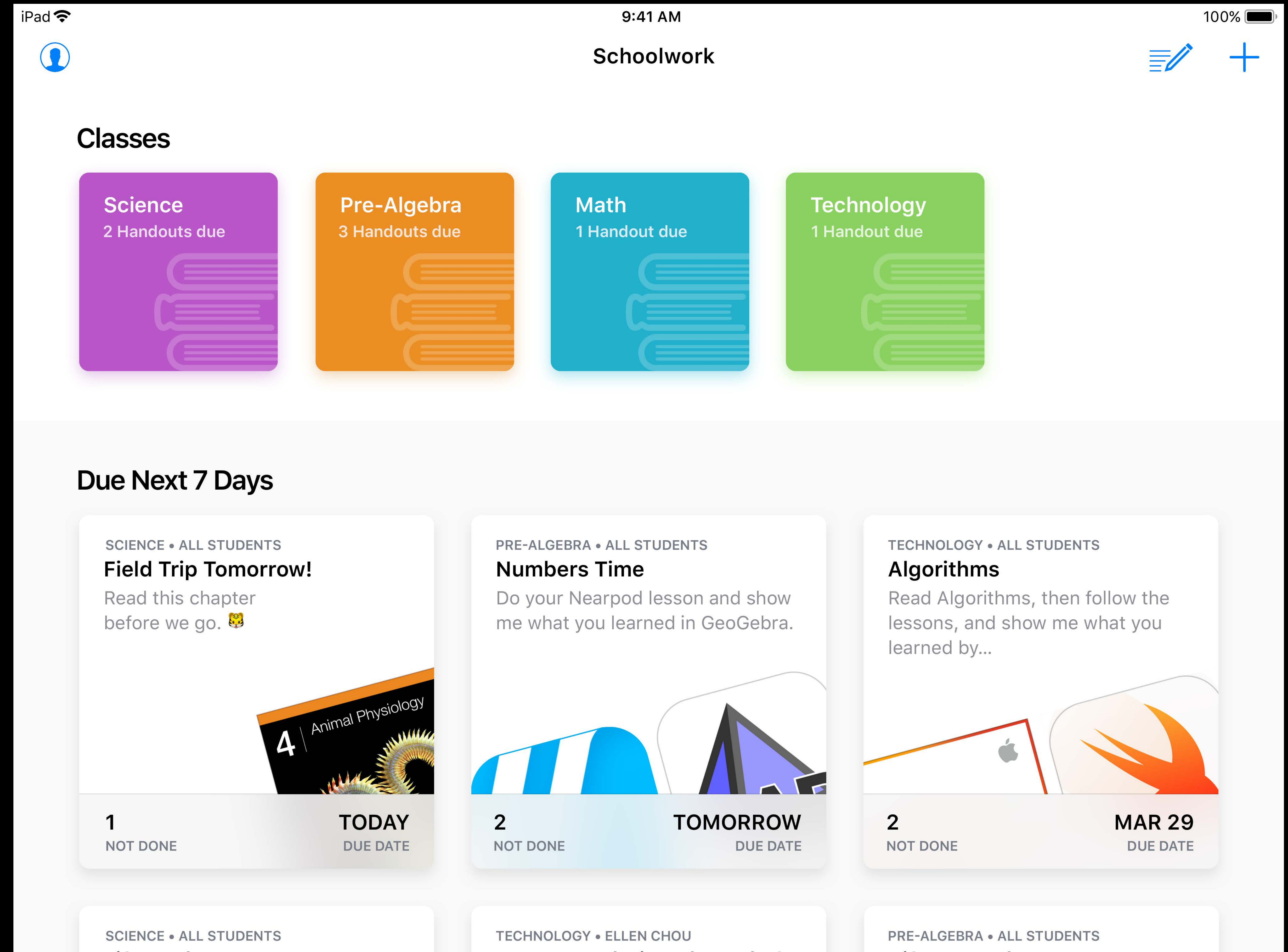
A new education app for students and teachers



Schoolwork

A new education app for students and teachers

View assigned handouts



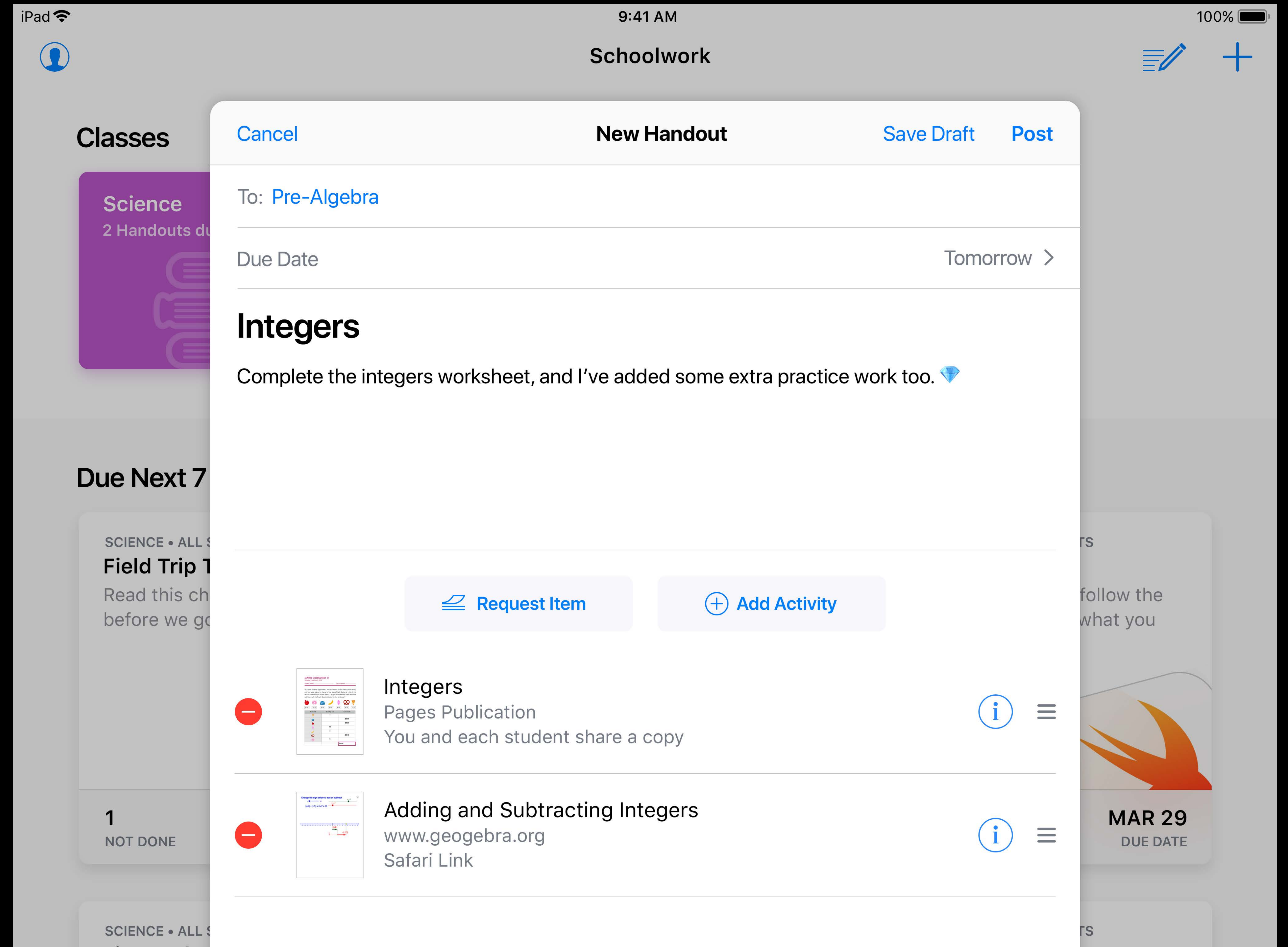
Schoolwork

A new education app

View assigned handouts

Create new handouts

- Handout is a collection of tasks



Schoolwork

A new education app

View assigned handouts

Create new handouts

- Handout is a collection of tasks

View progress reports

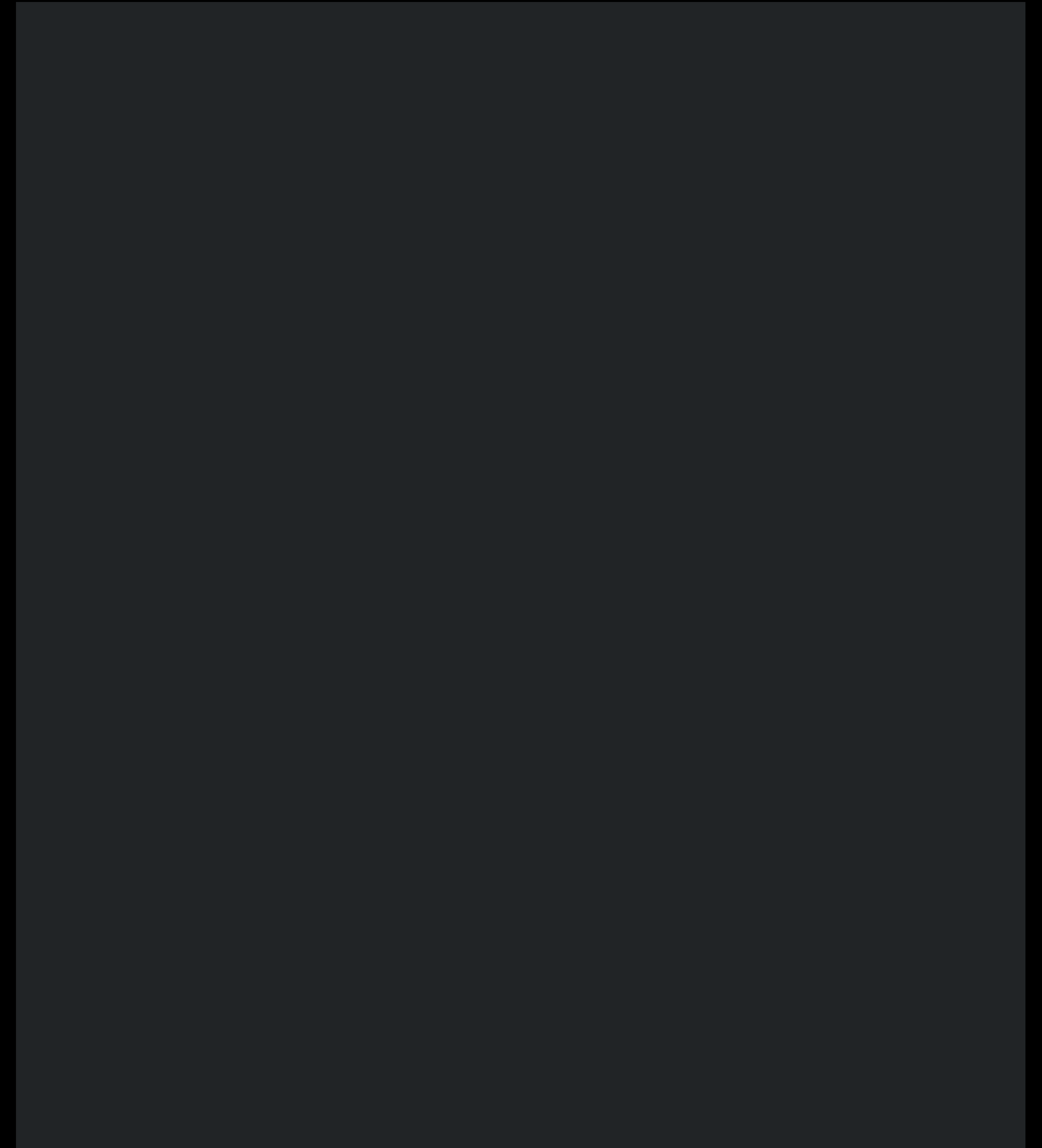
The screenshot displays the 'Integers' handout page in the Schoolwork app. At the top, it shows the subject 'Pre-Algebra' and the handout title 'Integers' posted on Mar 27. The due date is 'Tomorrow'. Below the title, there is a 'CLOSE HANDOUT' button and a notification icon. A summary shows 3 items 'NOT DONE' and 27 items 'COMPLETED'. A Kahoot! challenge titled 'Dividing Integers' is featured, with an average score of 87%, an average time of 22 minutes, and 24 out of 27 students completed it. Below this is a table of student progress:

STUDENTS	SCORE	TOTAL TIME	^ DONE
SC Sarah Castelblanco		41 min	
LS Lee Solon	83 %	20 min	
CB Chella Boehm	100 %	25 min	
EC Ellen Chou	95 %	15 min	✓
JF Jeanne Fox	87 %	20 min	✓
JB Jeremy Butcher	90 %	15 min	✓
GS Greg Scallon	98 %	30 min	✓
DD Darla Davidson	87 %	15 min	✓
AP Andrew Penick	83 %	17 min	✓
RJ Raffi Jilizian	97 %	22 min	✓

Lifecycle of a Handout

Lifecycle of a Handout

Define assignable content



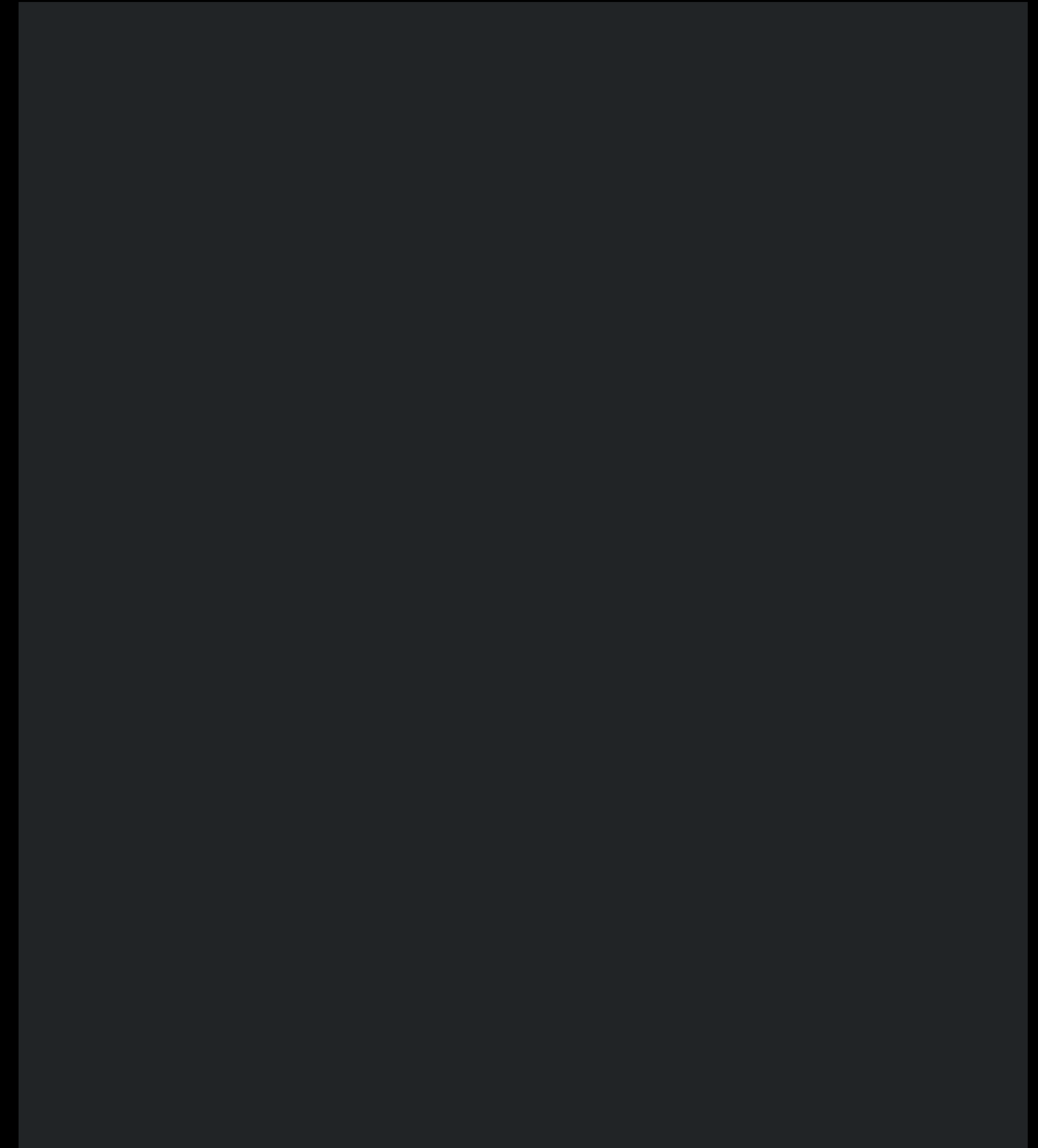
myReaderApp

Lifecycle of a Handout

Define assignable content

Content is represented via `CLSContext`

- Forms a tree structure



Lifecycle of a Handout

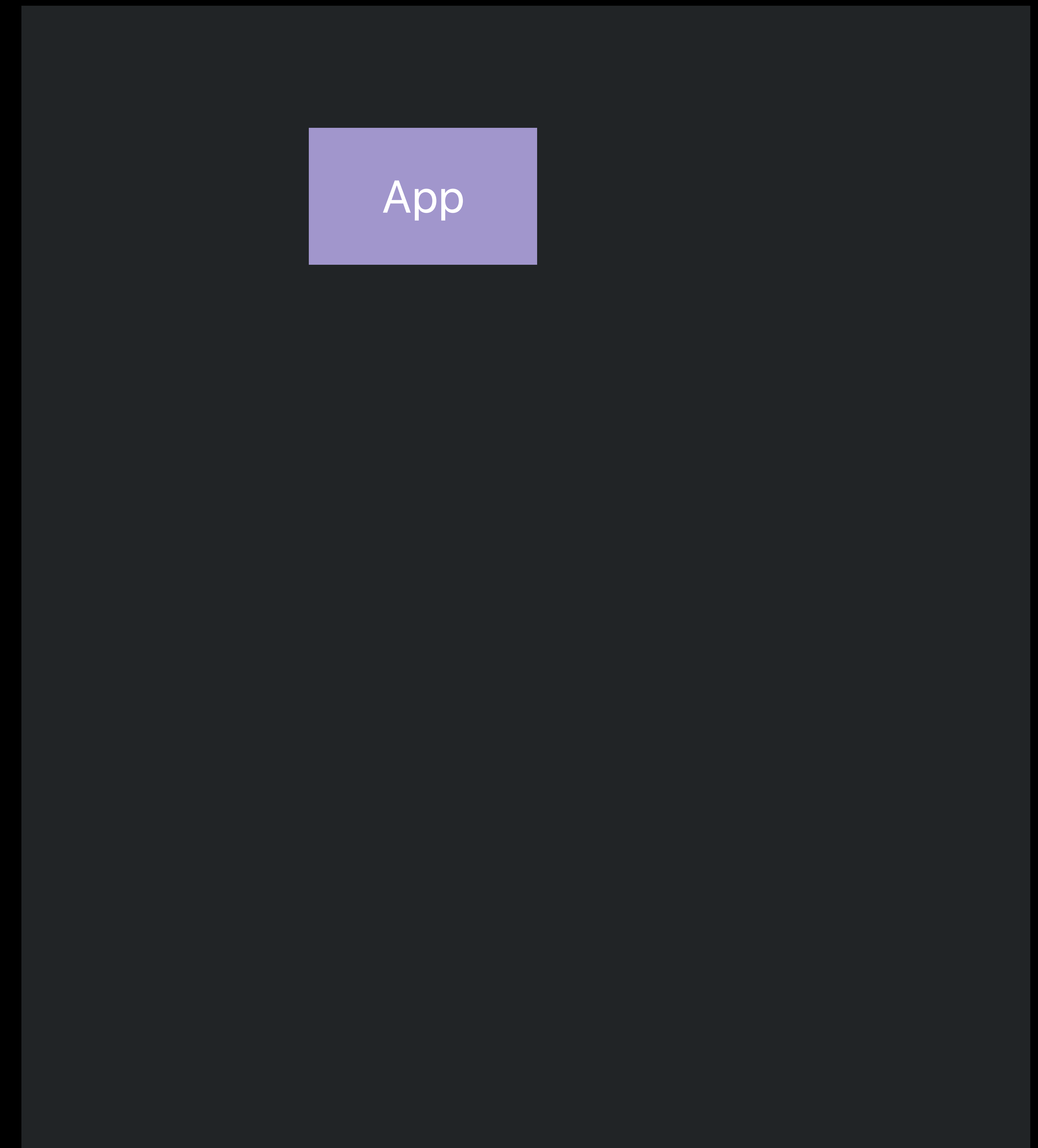
Define assignable content

Content is represented via `CLSContext`

- Forms a tree structure

One main `app context` (root context)

- Created by default



myReaderApp

Lifecycle of a Handout

Define assignable content

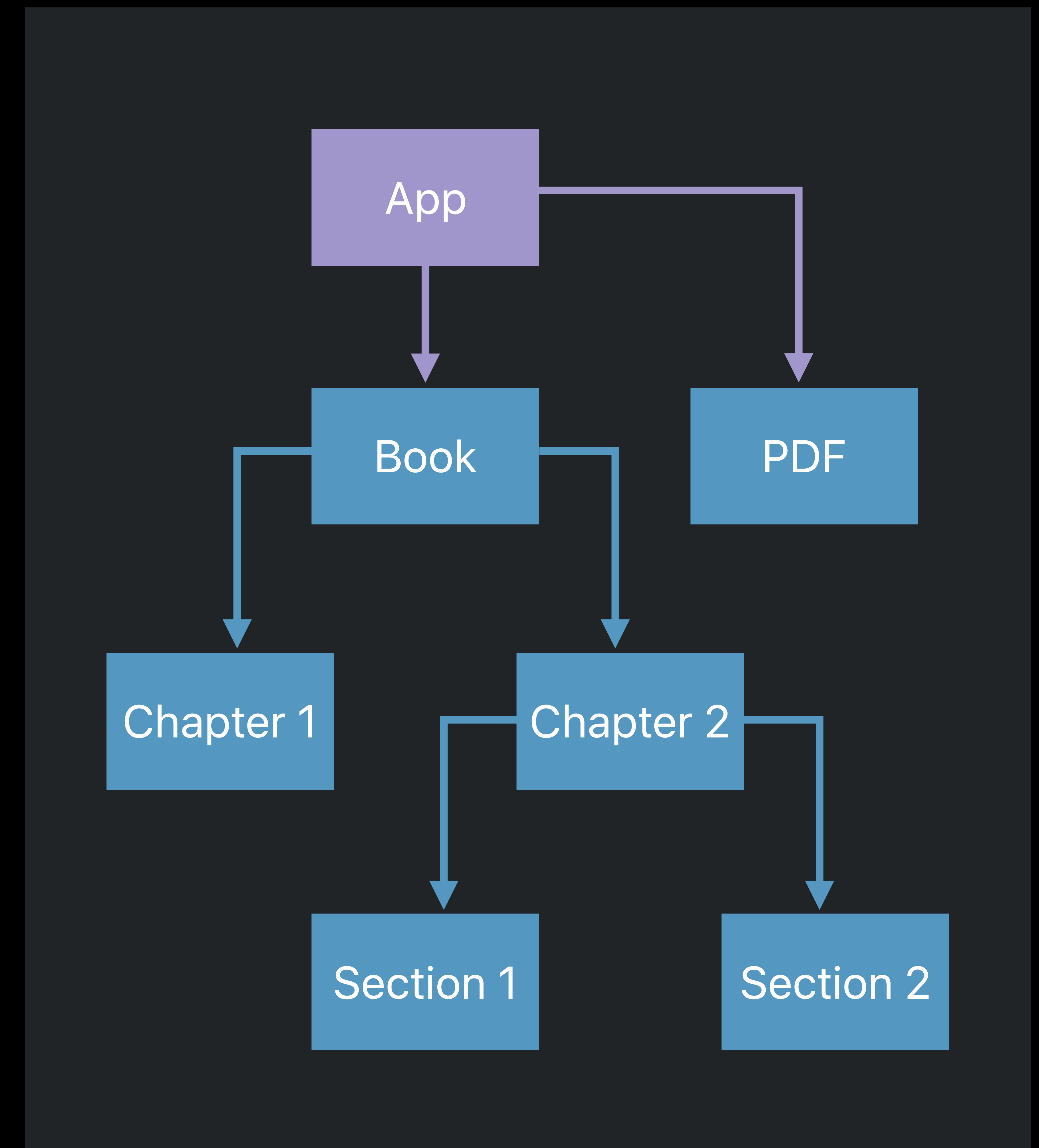
Content is represented via `CLSContext`

- Forms a tree structure

One main app context (root context)

- Created by default

App content is added to the app context



myReaderApp

Lifecycle of a Handout

Define assignable content

Content is represented via `CLSContext`

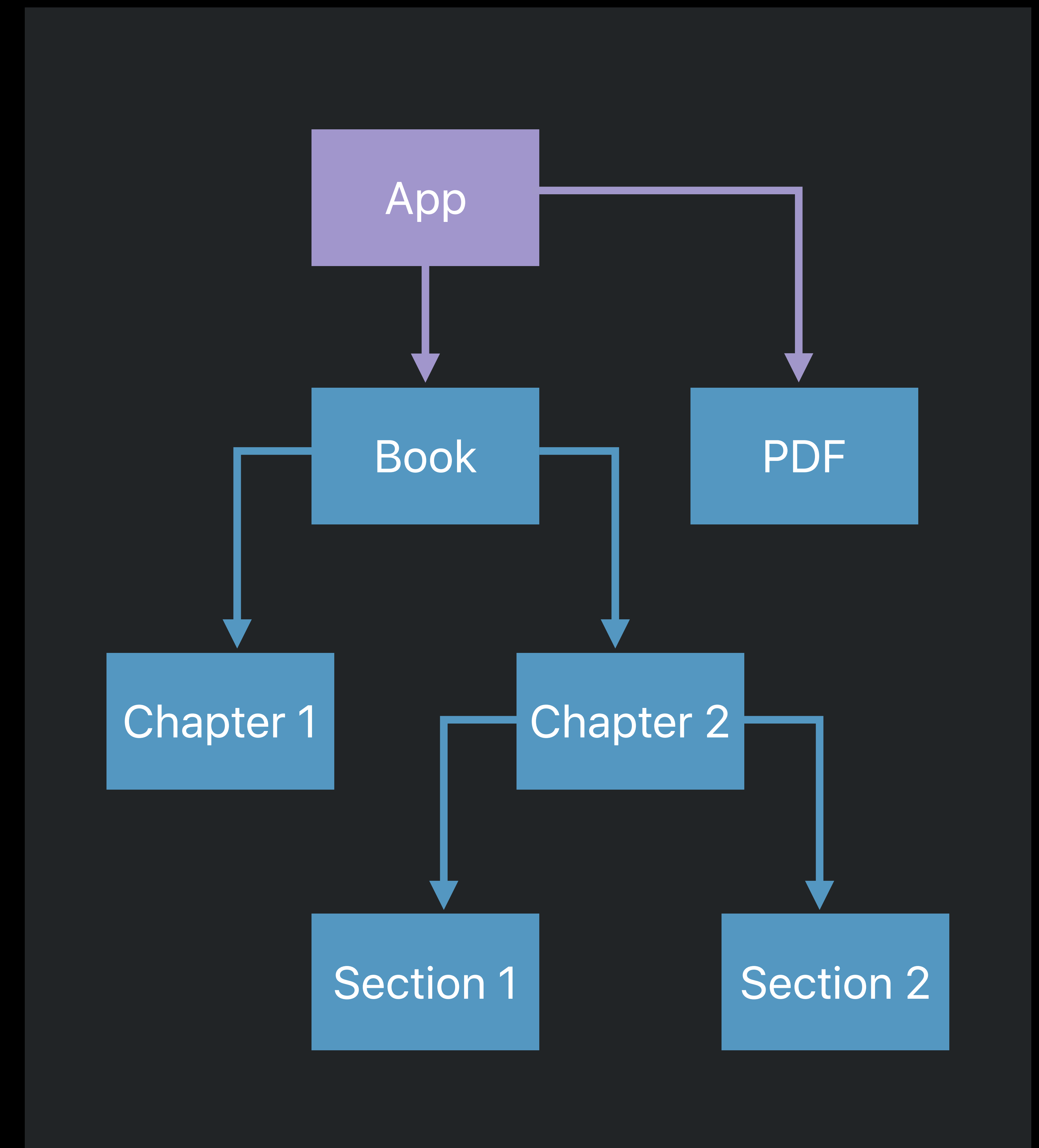
- Forms a tree structure

One main app context (root context)

- Created by default

App content is added to the app context

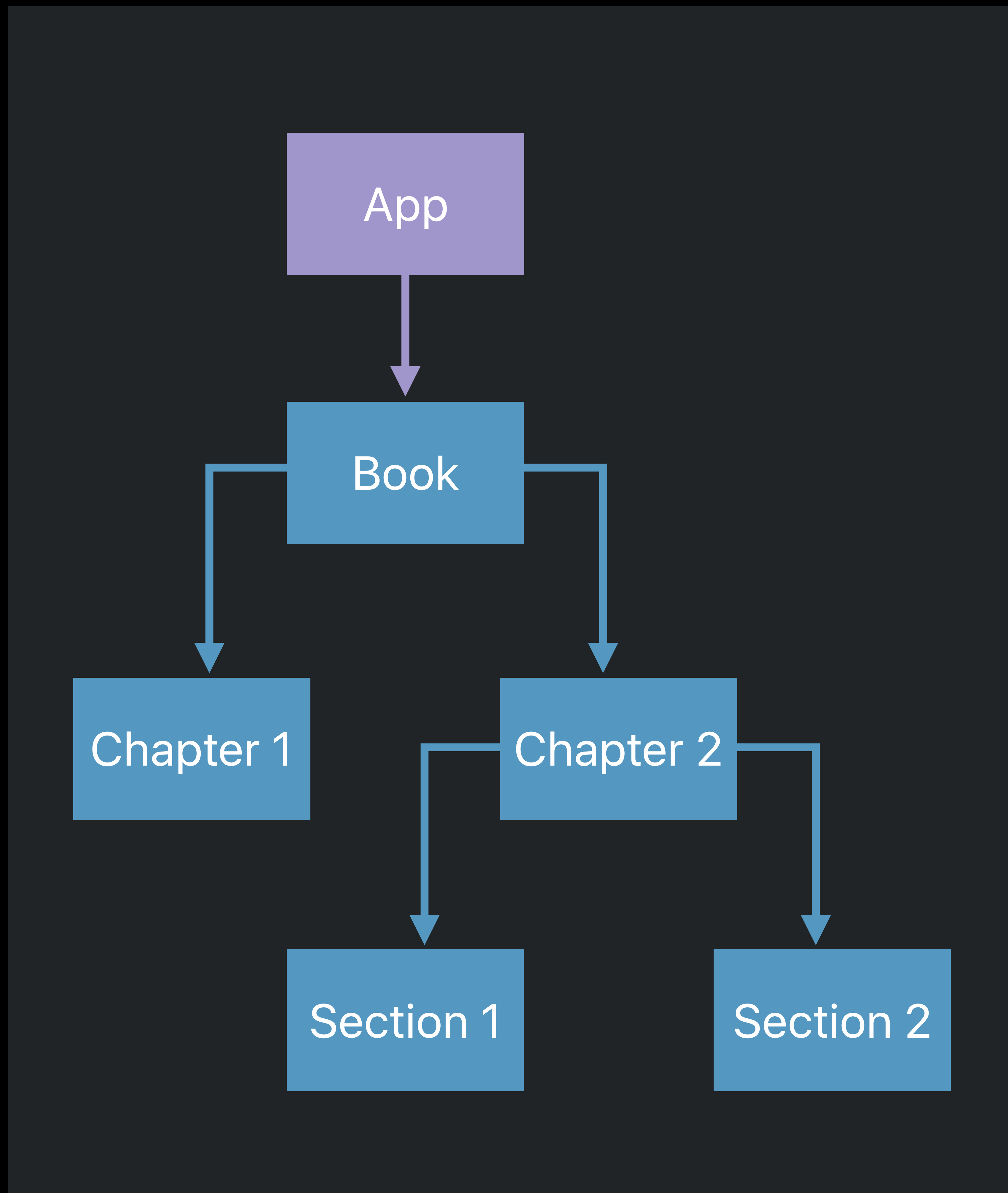
Define content as early as possible



myReaderApp

Lifecycle of a Handout

Assignment



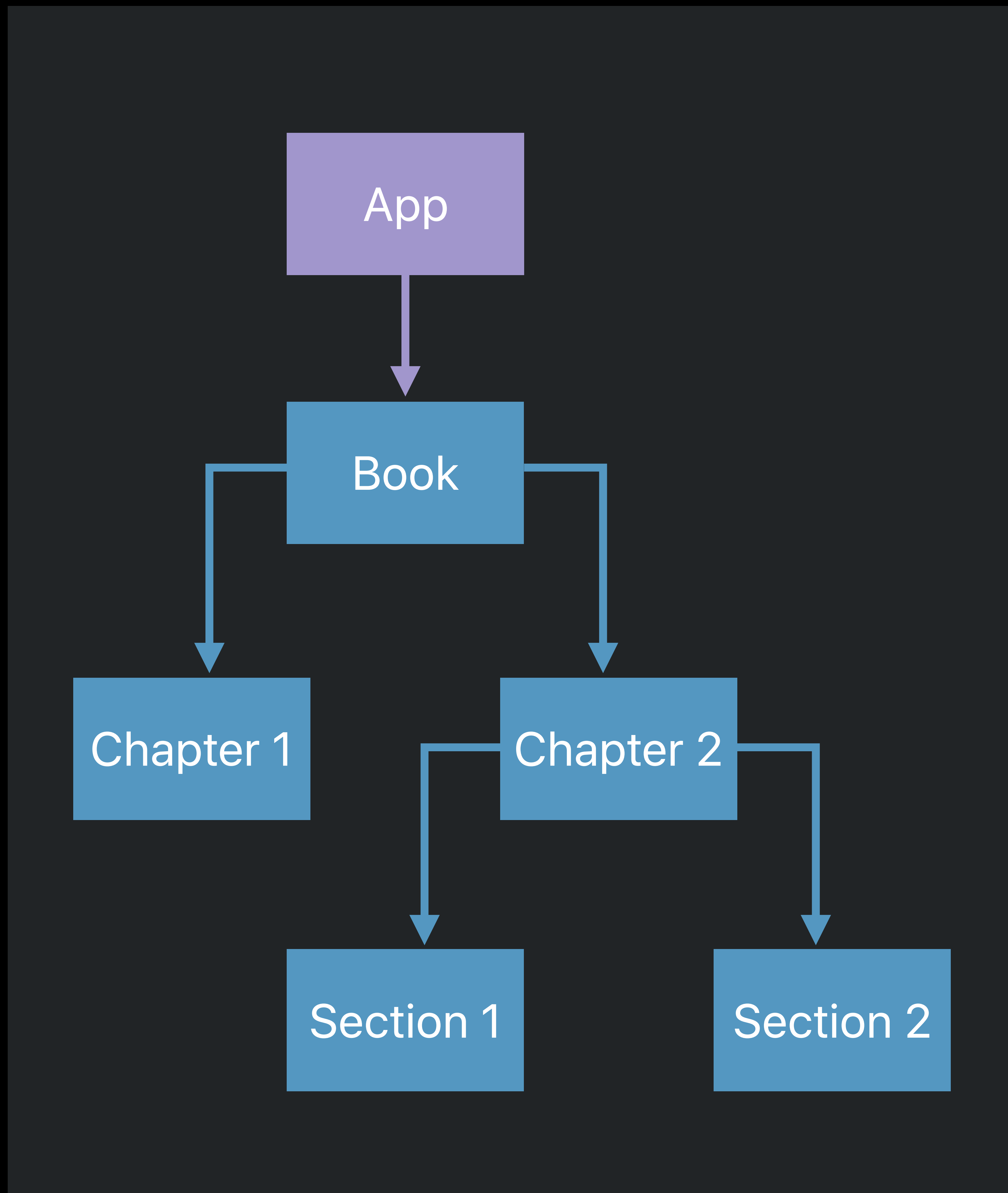
Schoolwork

myReaderApp - Teacher iPad

myReaderApp - Student iPad

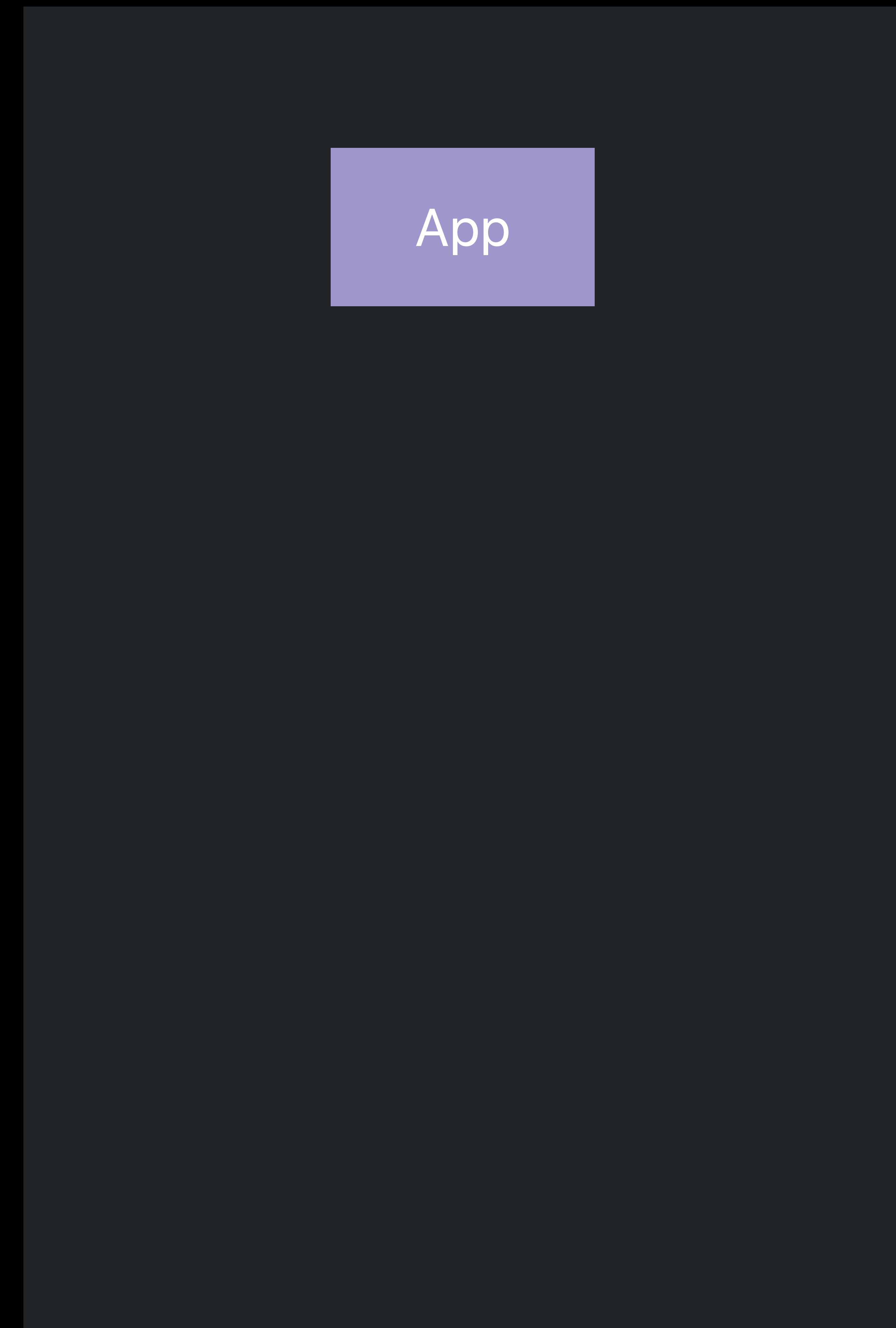
Lifecycle of a Handout

Assignment



myReaderApp - Teacher iPad

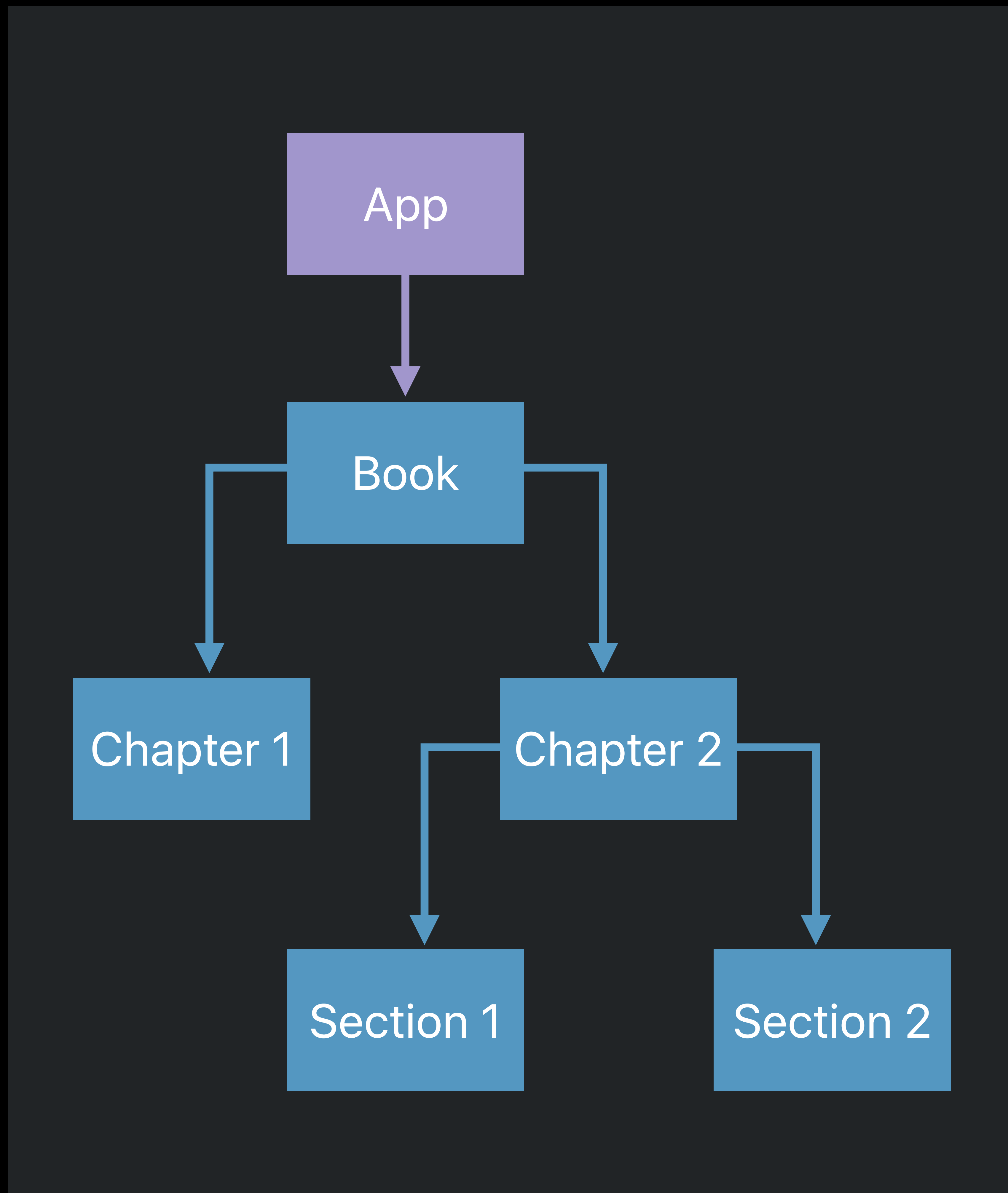
Schoolwork



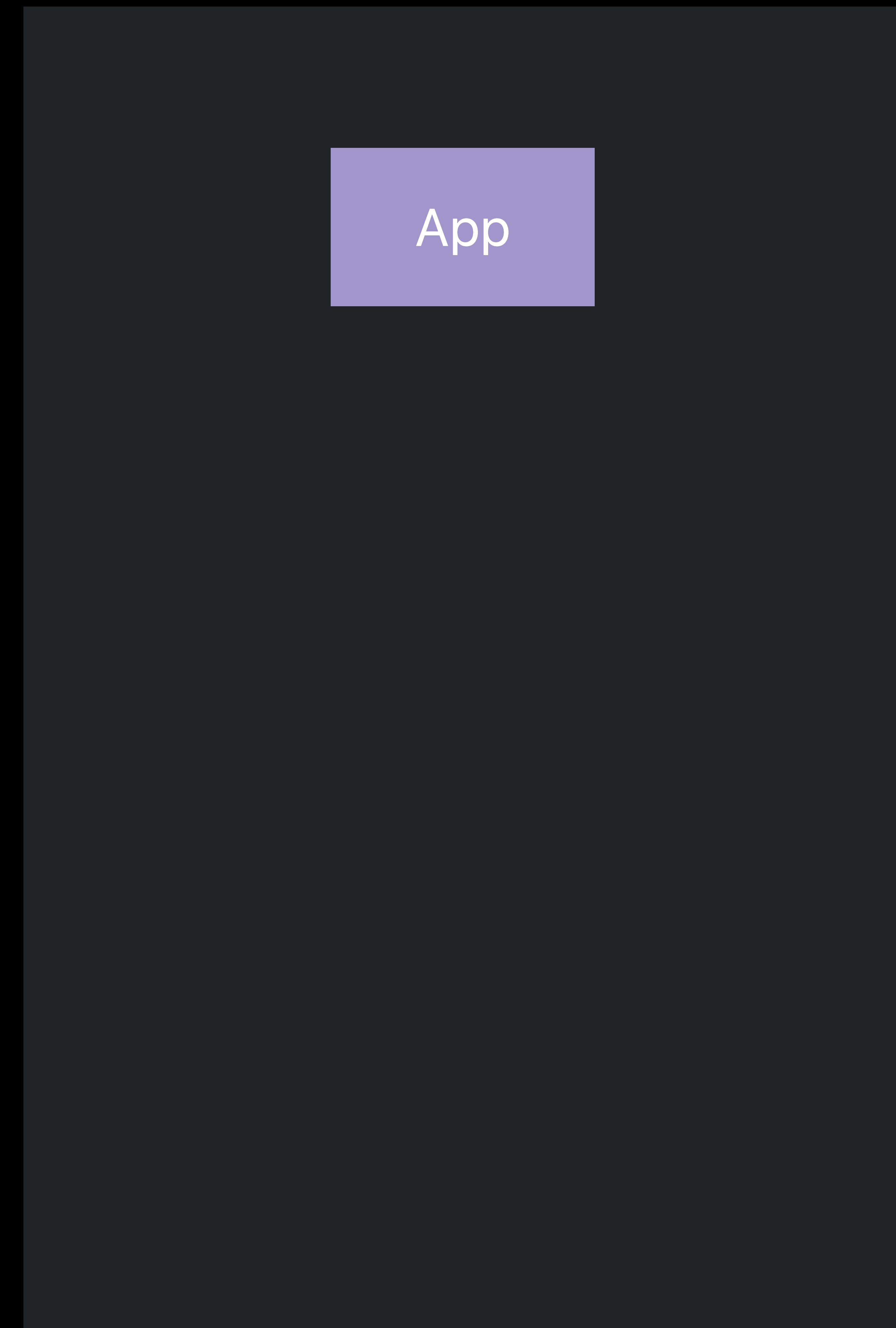
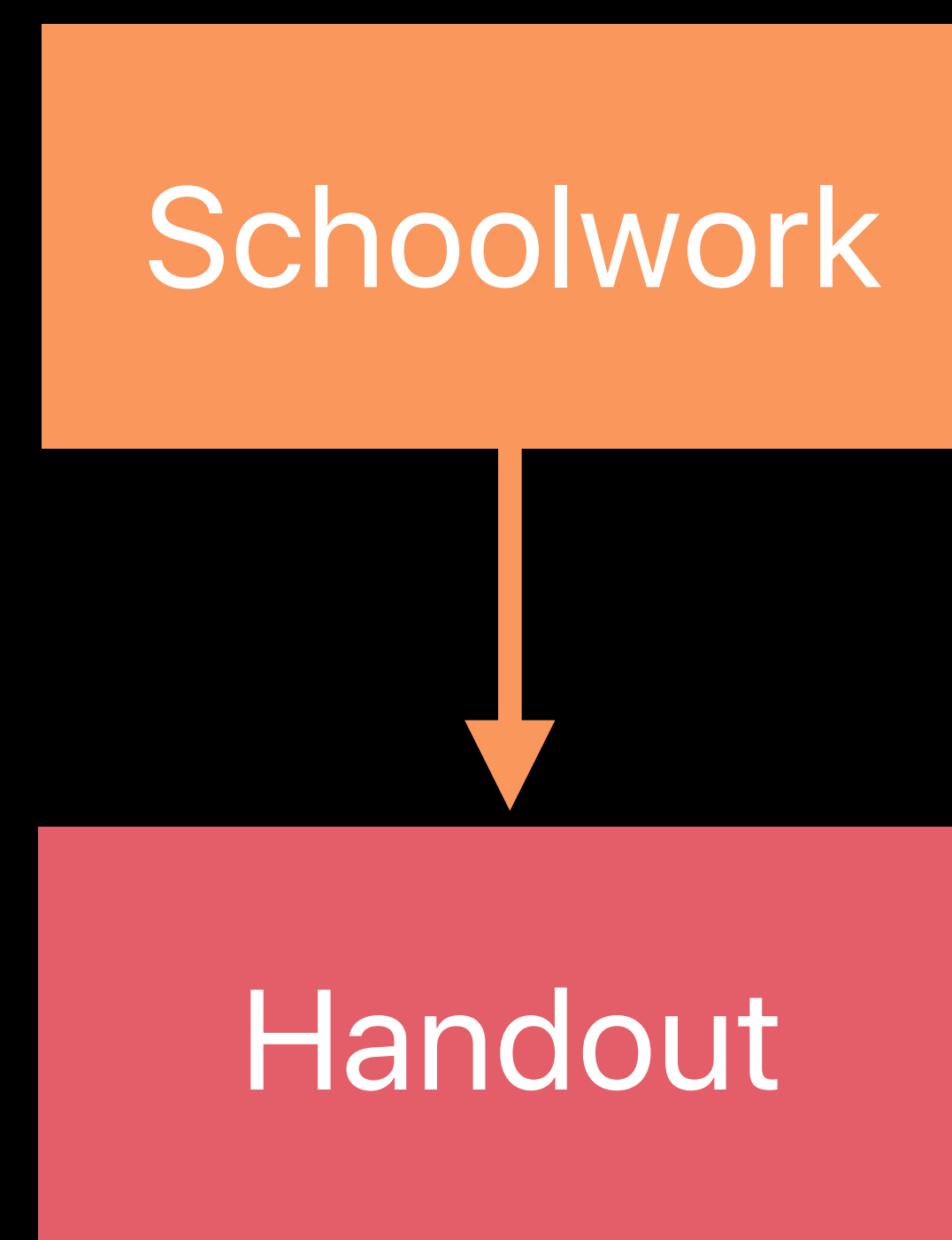
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



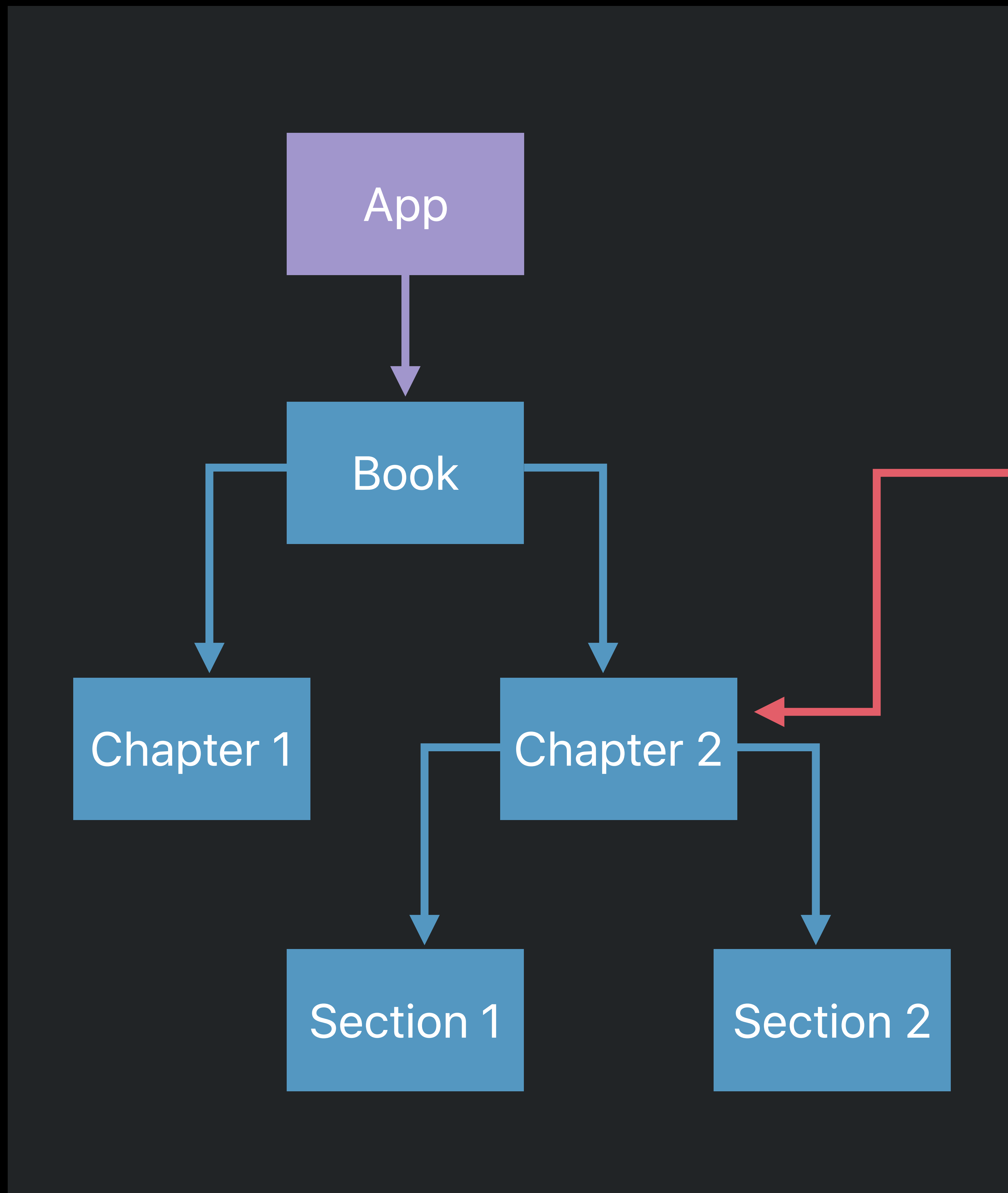
myReaderApp - Teacher iPad



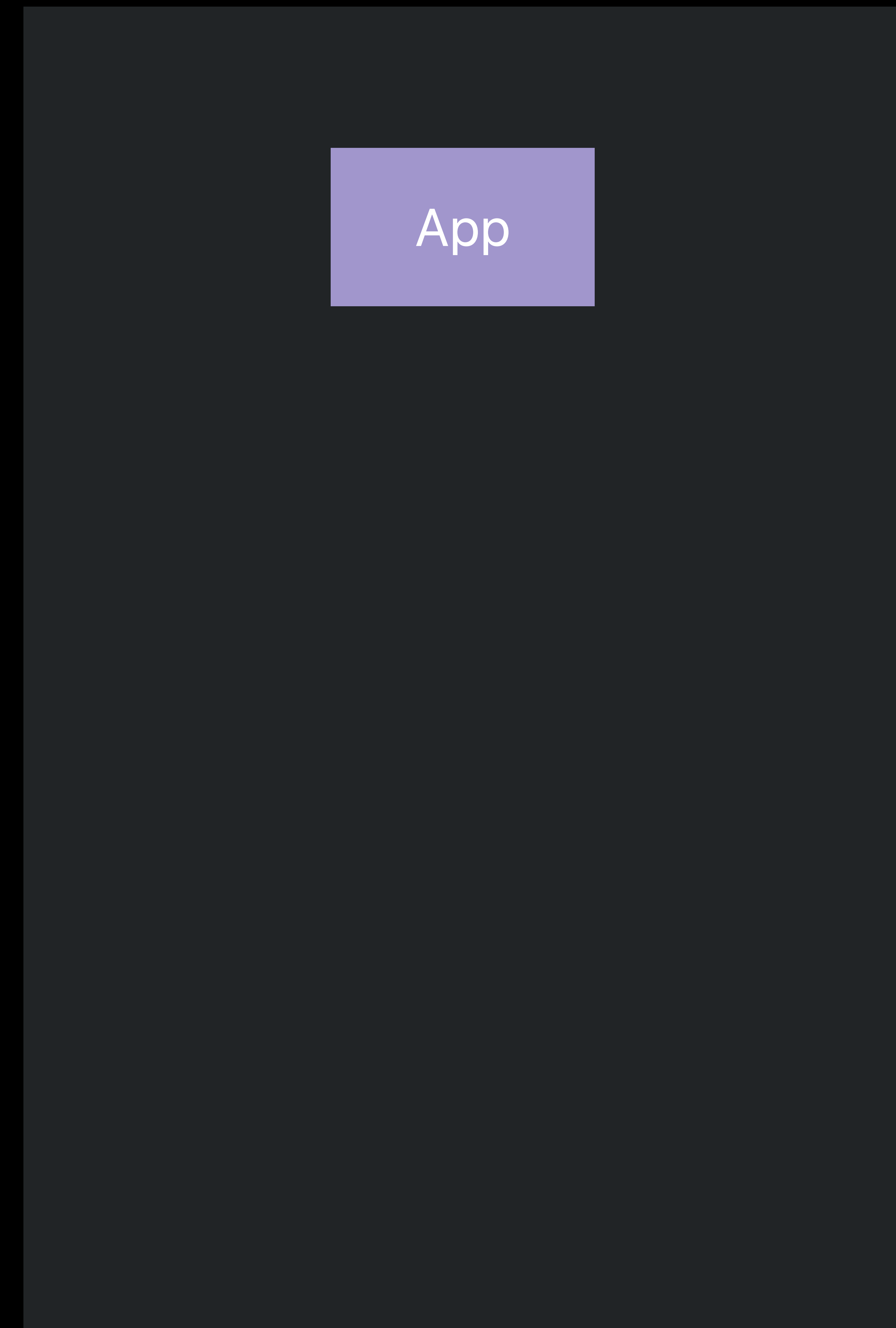
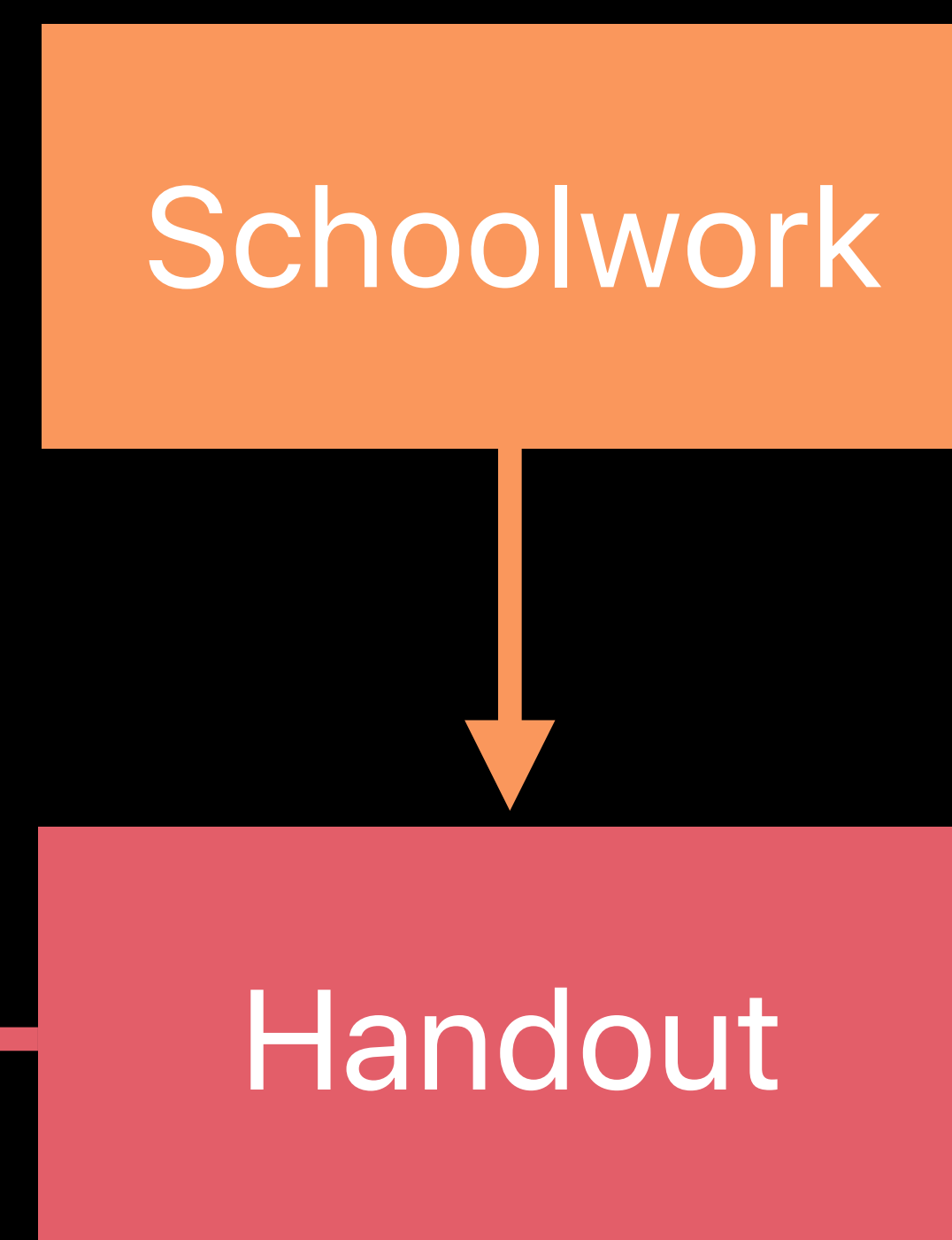
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



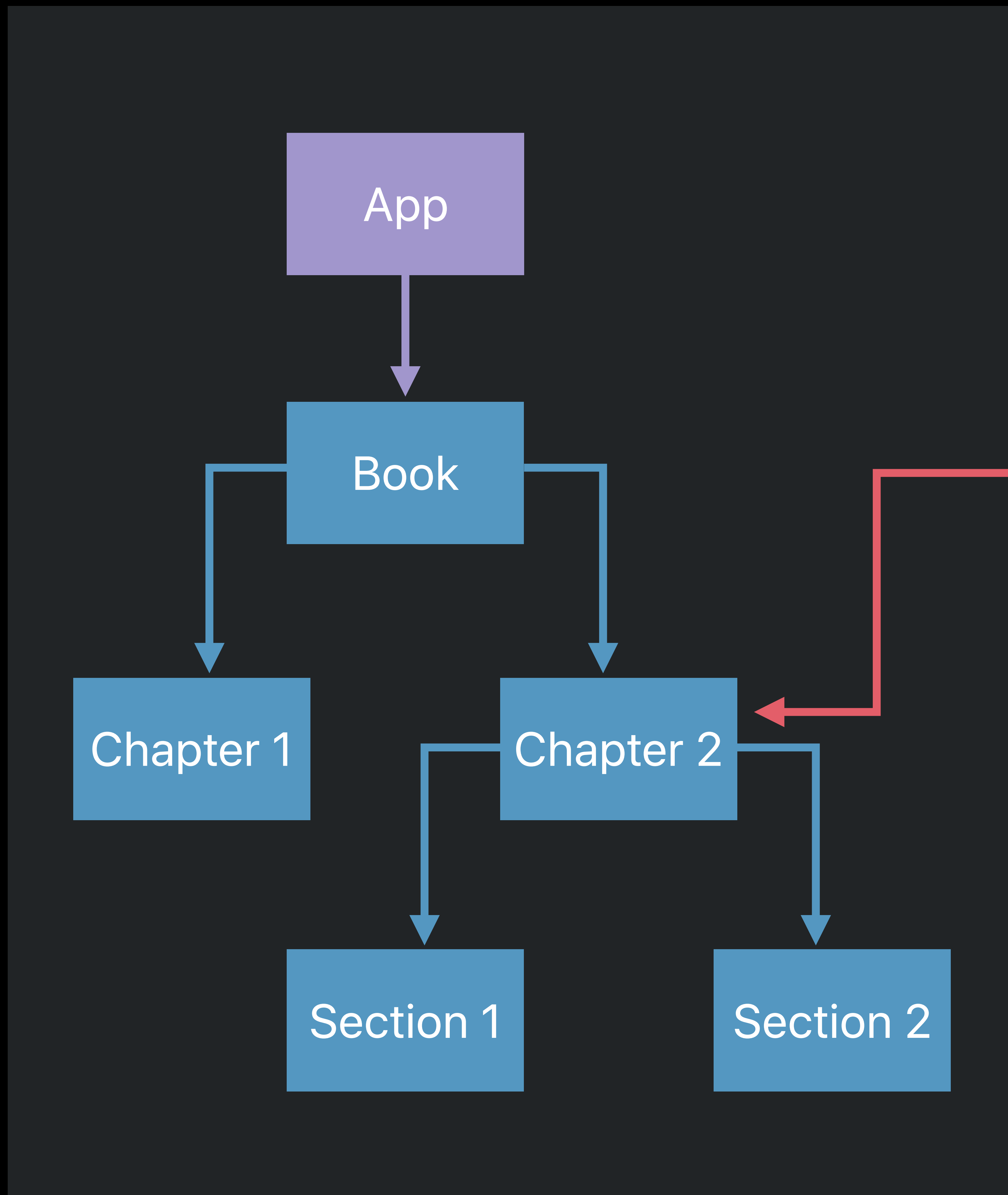
myReaderApp - Teacher iPad



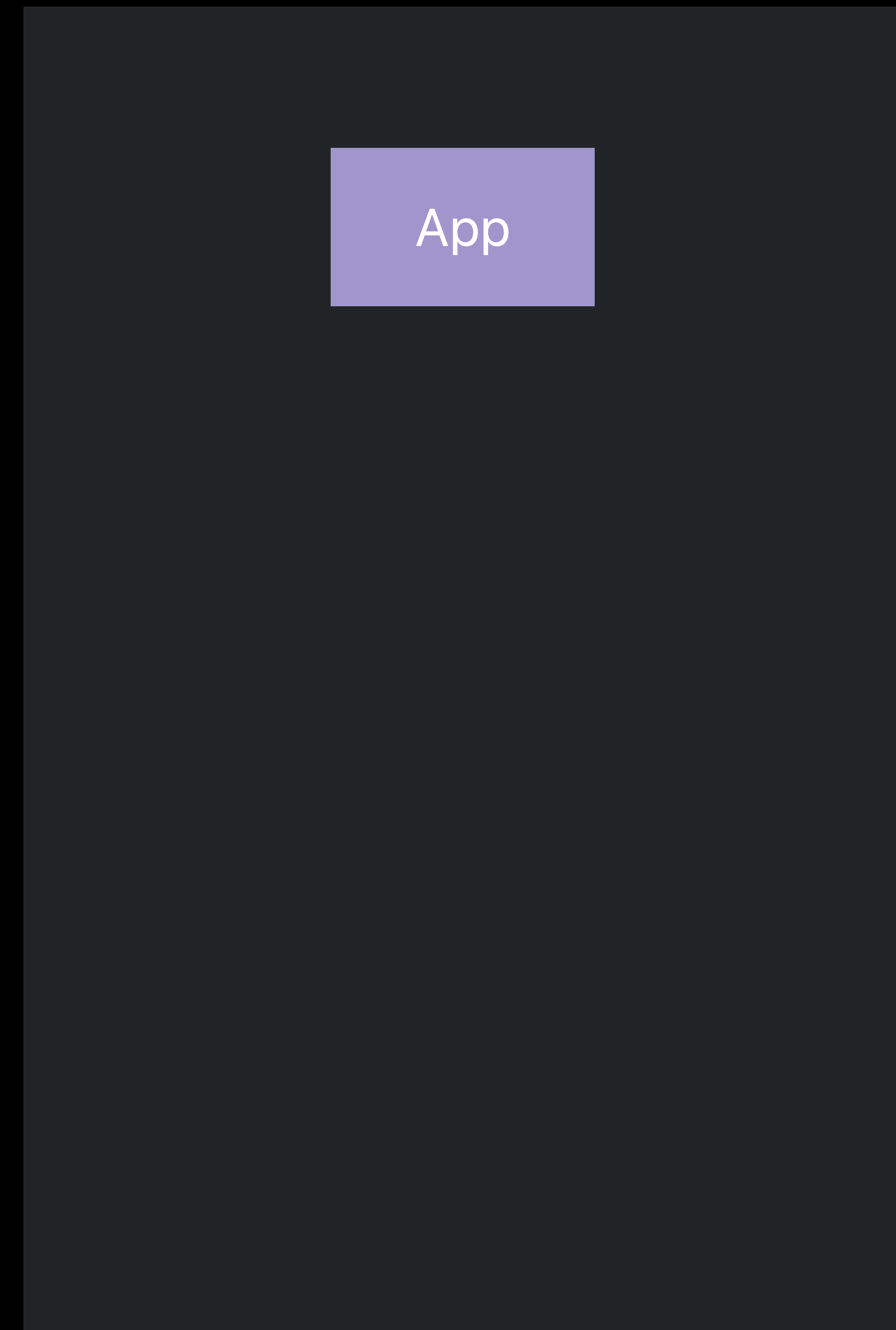
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



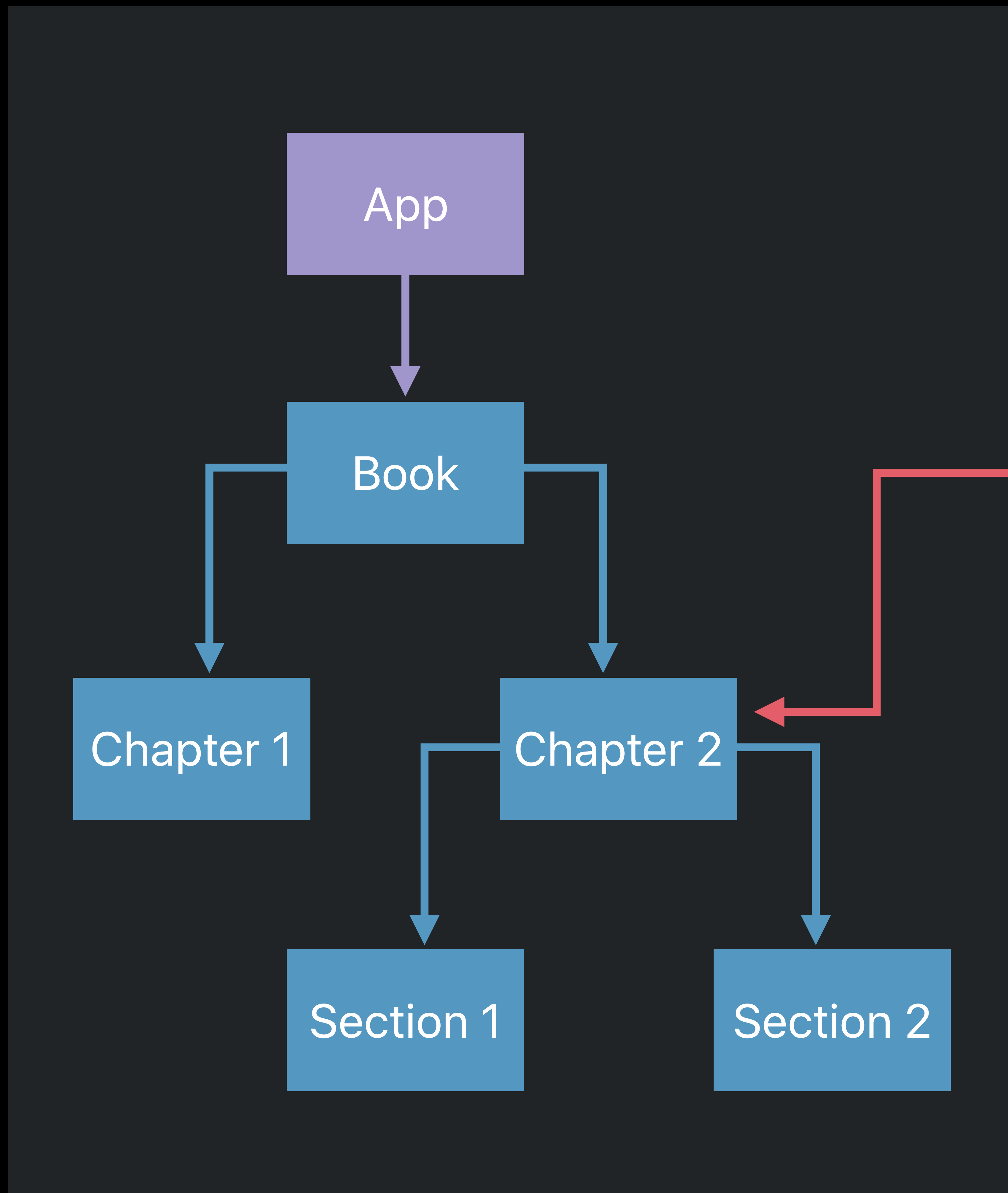
myReaderApp - Teacher iPad



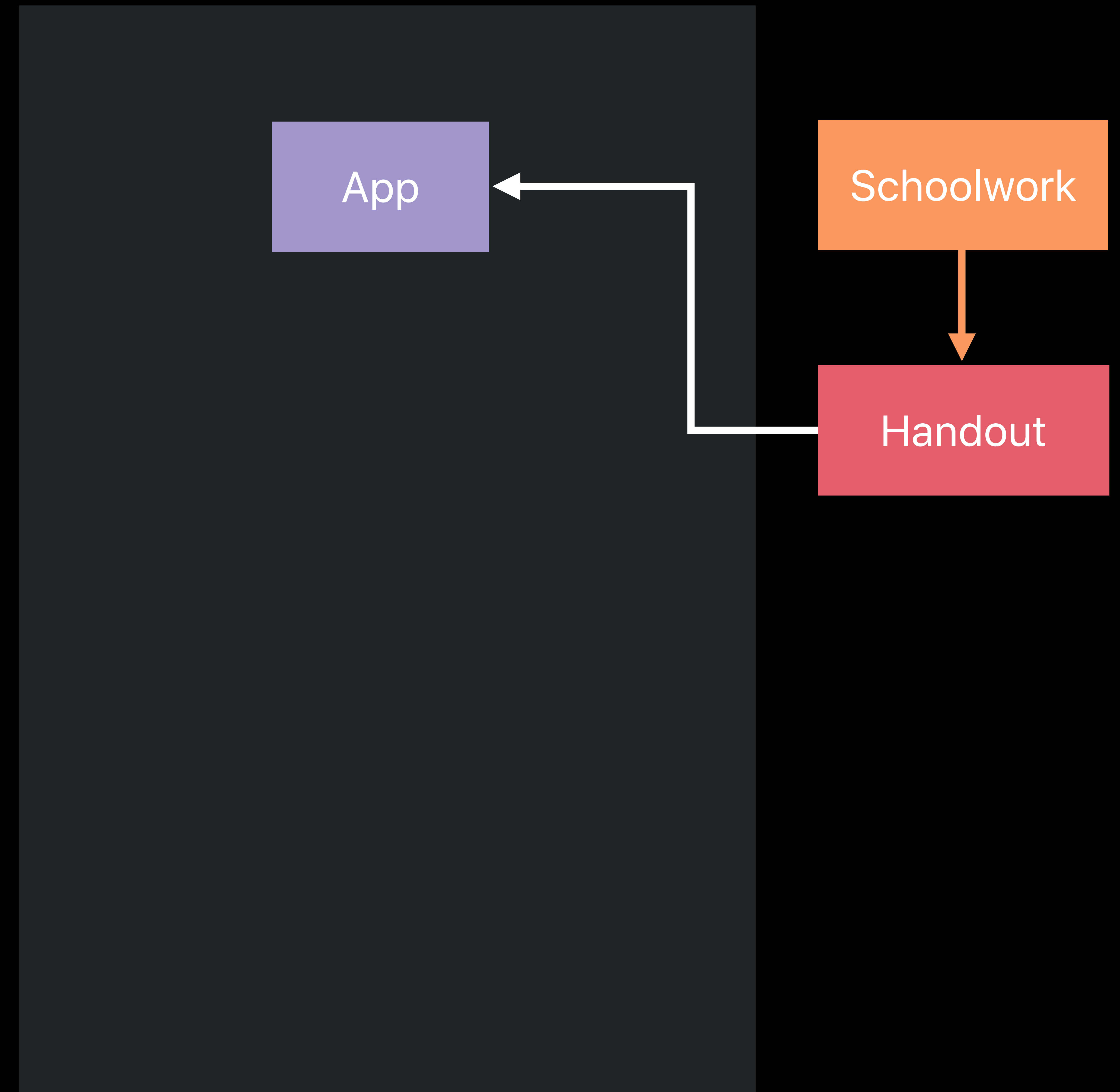
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



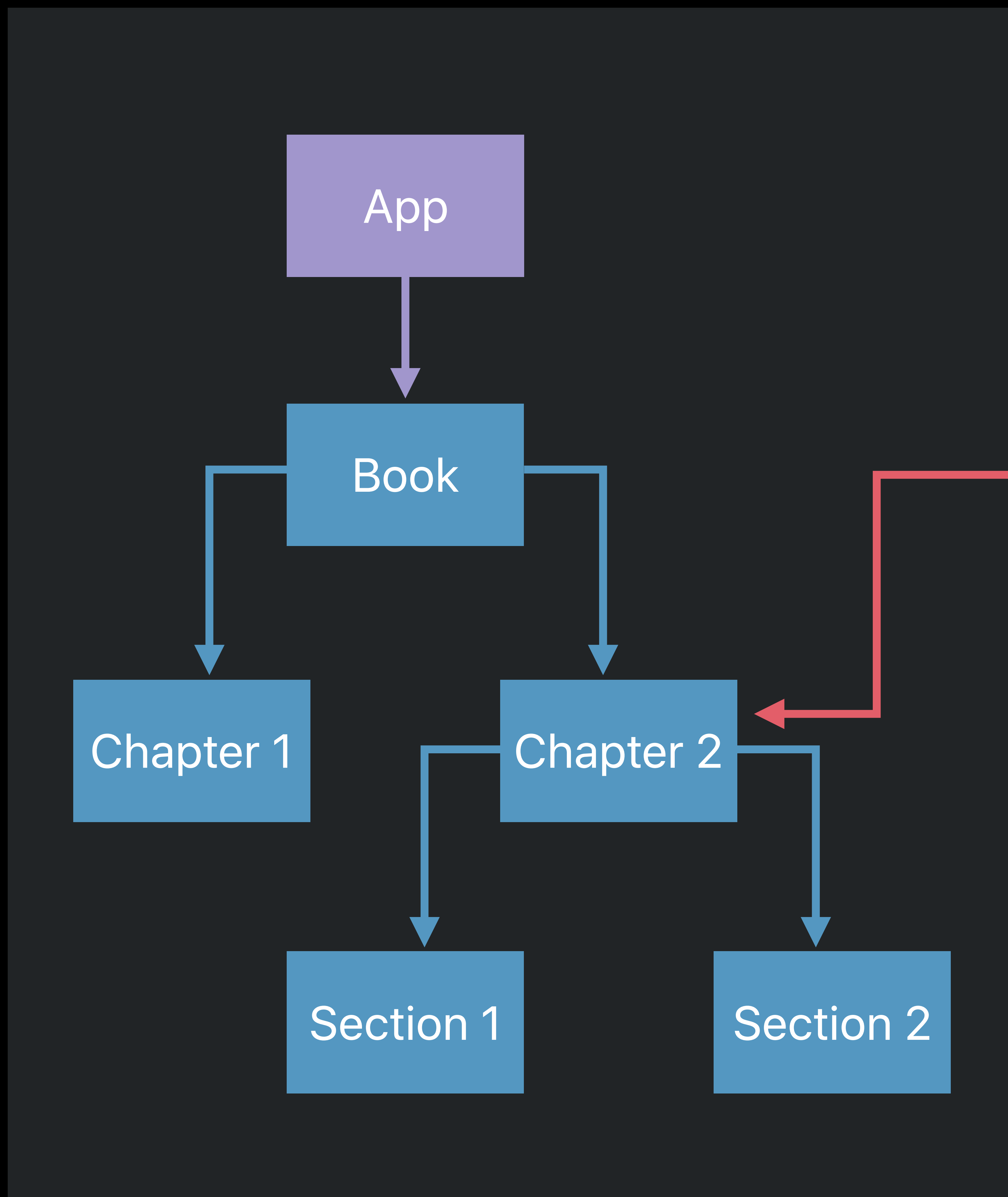
myReaderApp - Teacher iPad



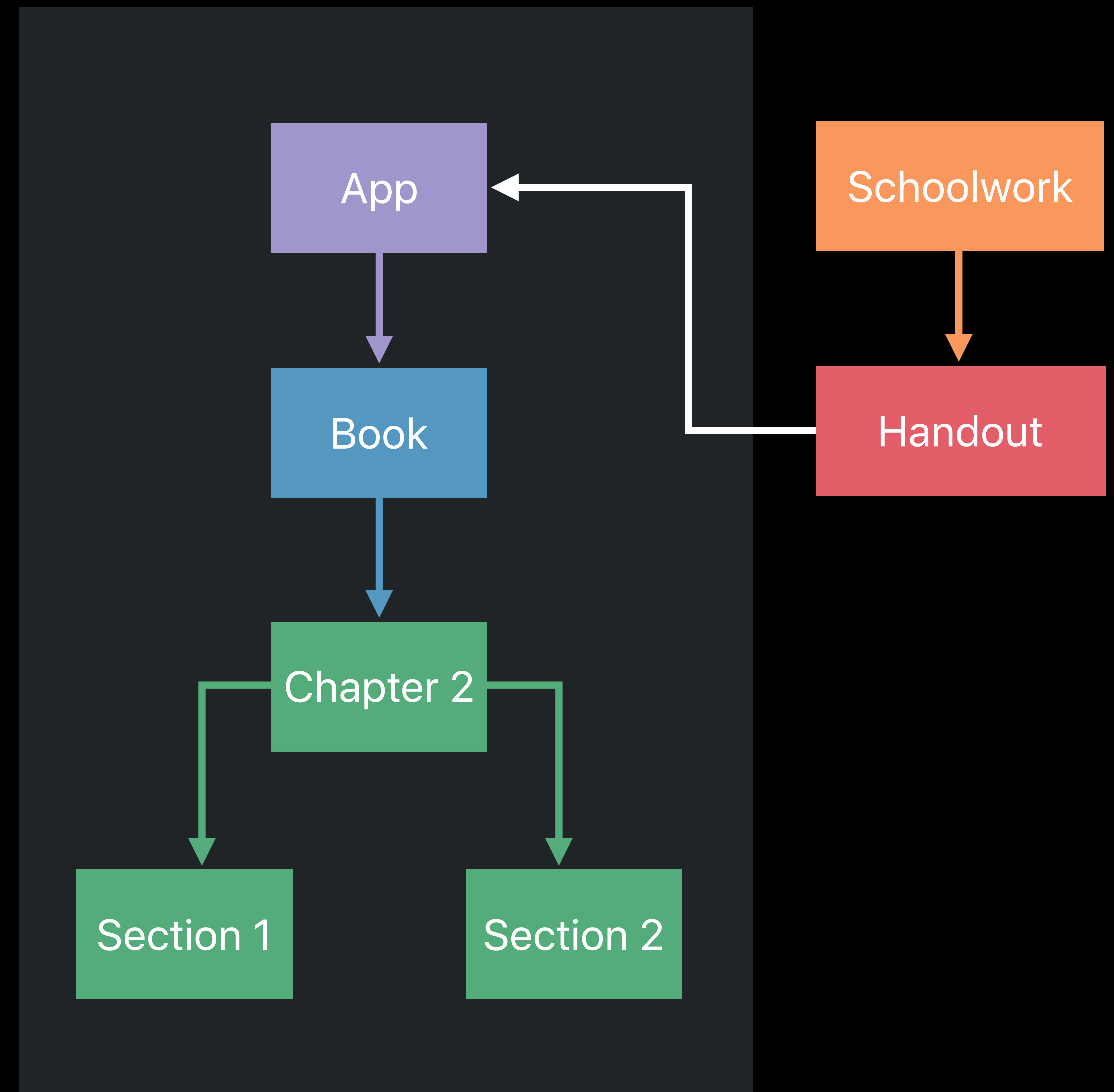
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



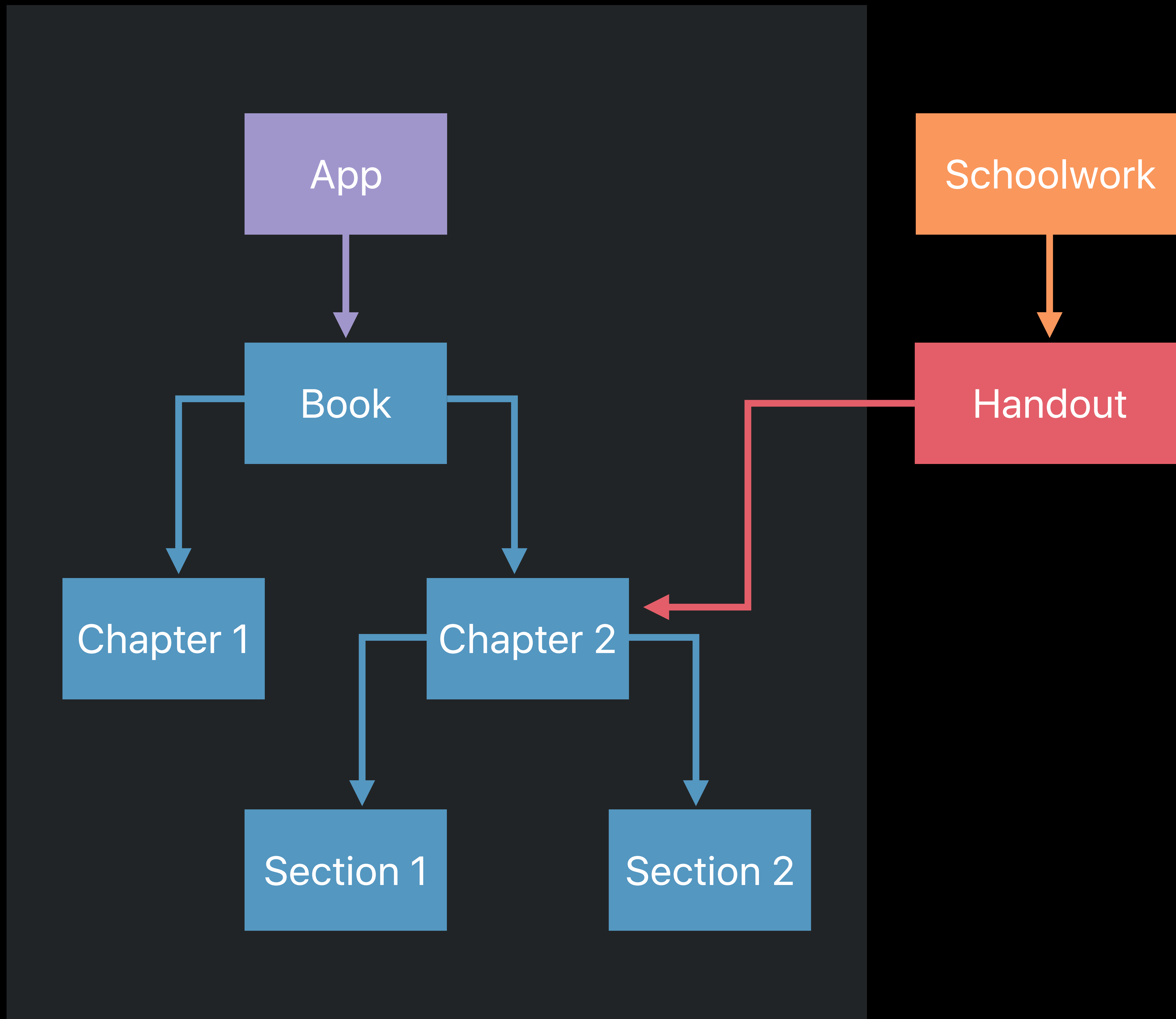
myReaderApp - Teacher iPad



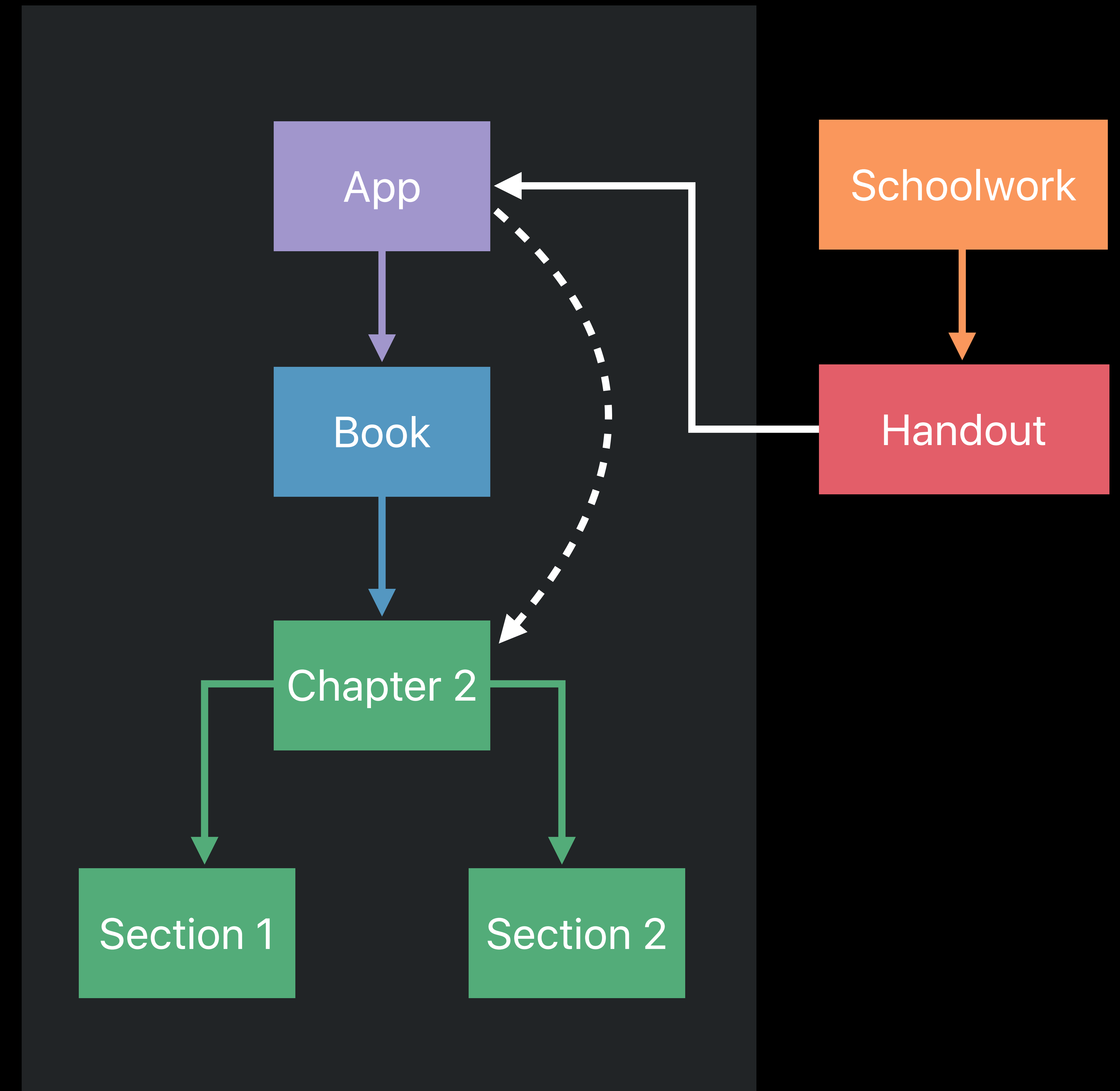
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



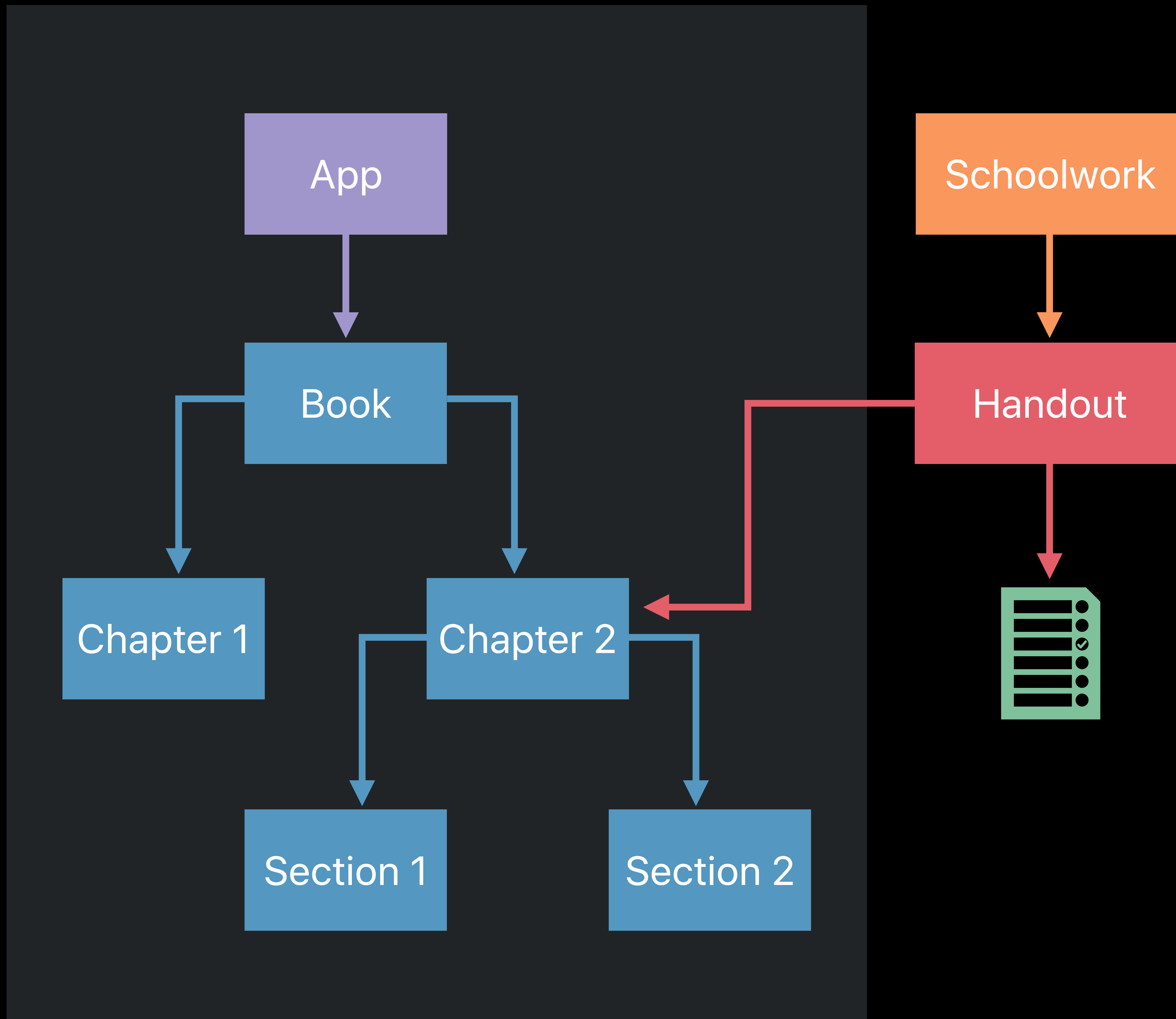
myReaderApp - Teacher iPad



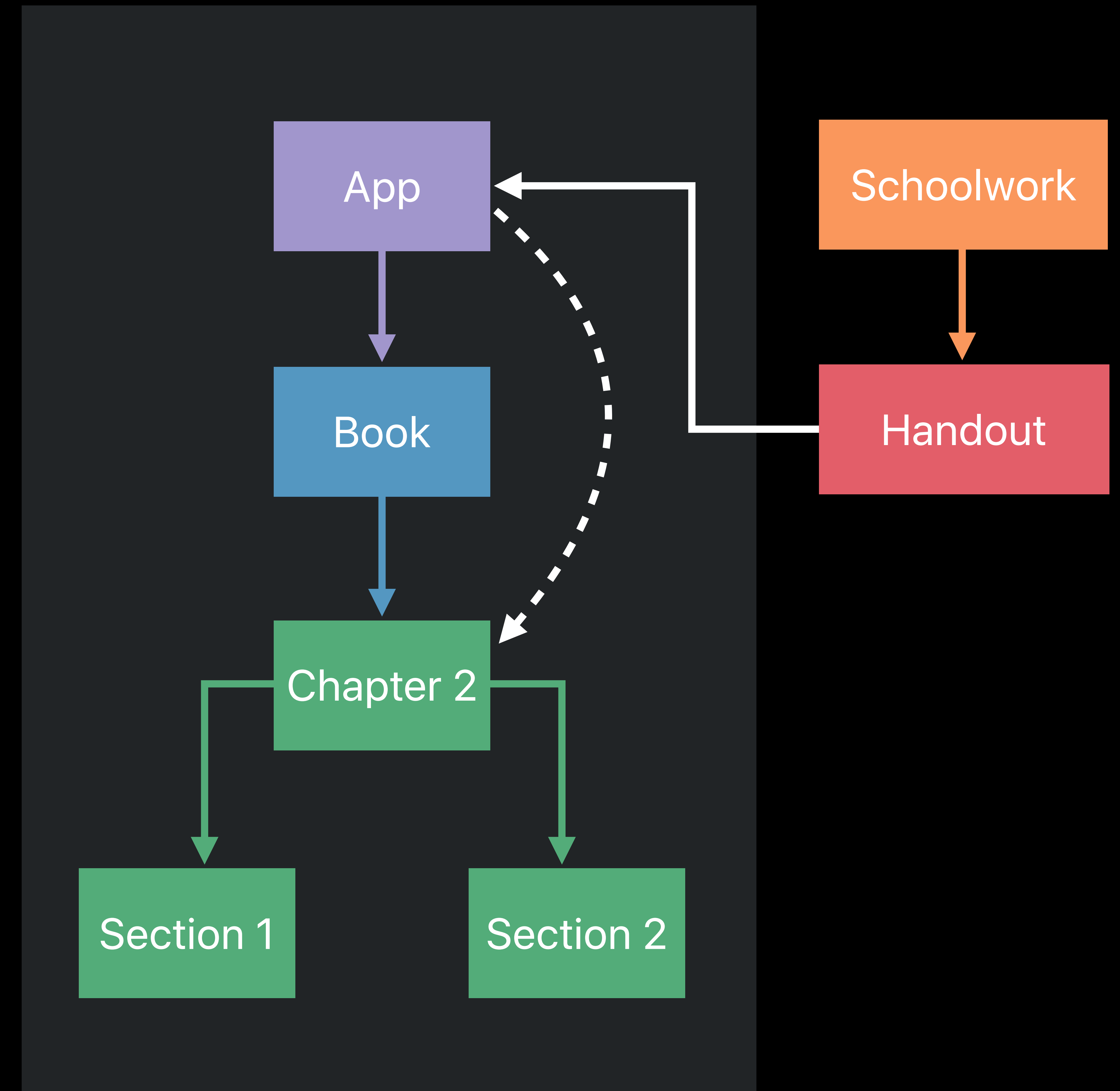
myReaderApp - Student iPad

Lifecycle of a Handout

Assignment



myReaderApp - Teacher iPad



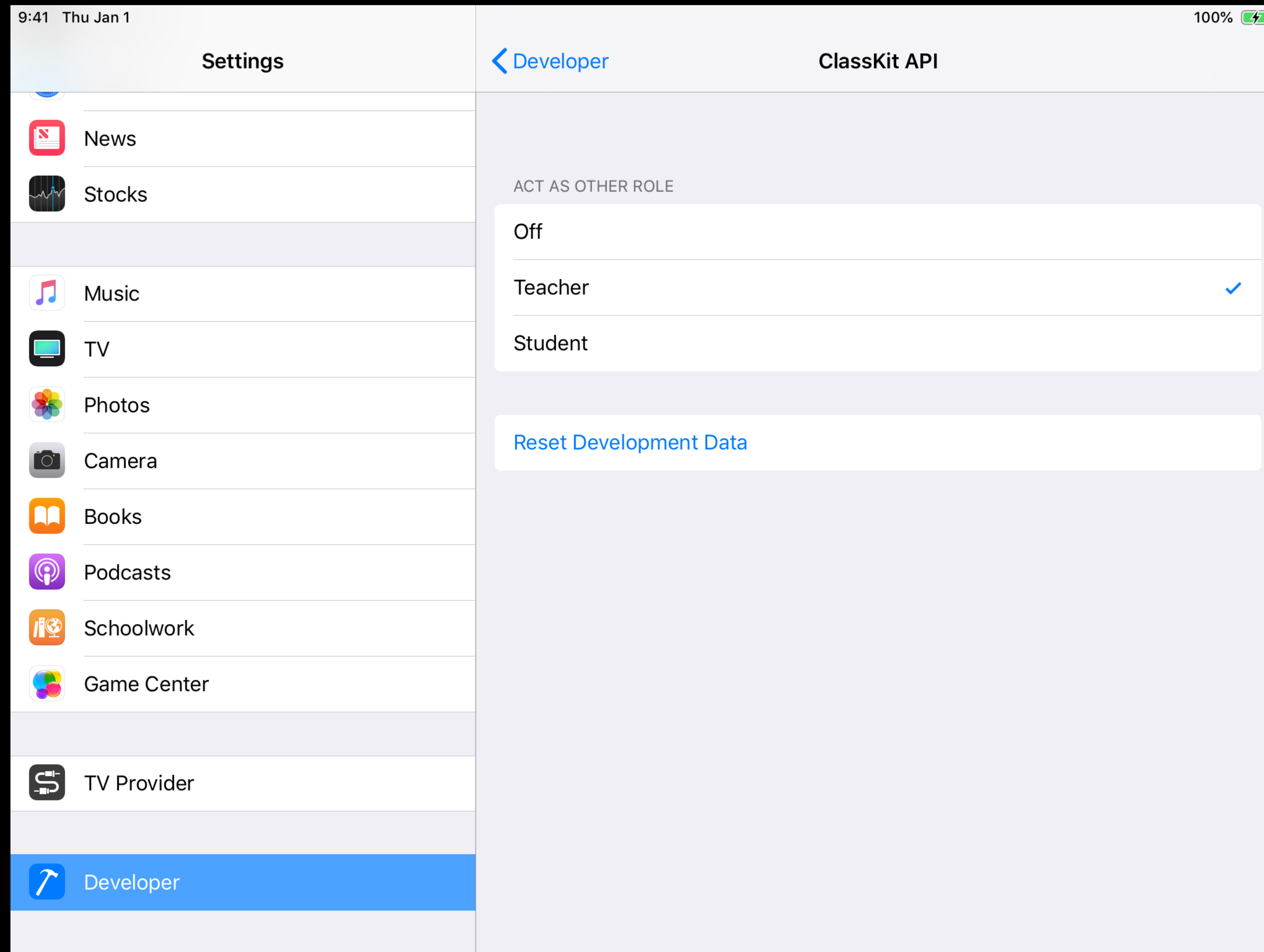
myReaderApp - Student iPad

Developer Mode

Settings > Developer > ClassKit API

Developer Mode

Settings > Developer > ClassKit API



Framework Classes

Framework Classes

CLSDataStore

Data store



Data Store

CLSDataStore

CLSDataStore

Maintains the context tree

CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

Keeps track of modified objects

CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

Keeps track of modified objects

- Objects saved by calling `CLSDataStore.save(completion:)`

CLSDataStore

Maintains the context tree

Exposes the `mainAppContext` as a property

Keeps track of modified objects

- Objects saved by calling `CLSDataStore.save(completion:)`

One shared instance

Framework Classes

CLSDataStore

Data store



Data Store

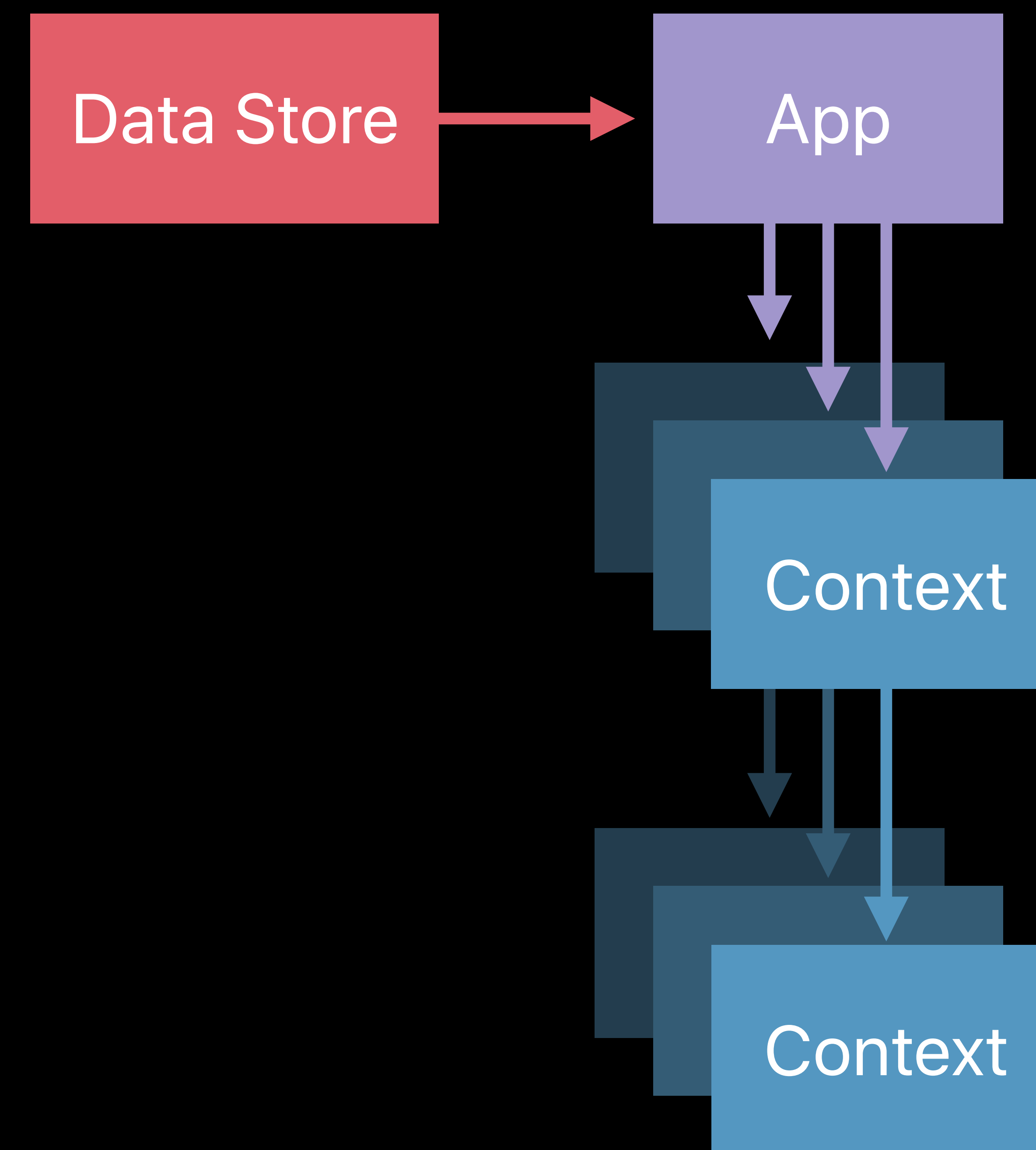
Framework Classes

CLLocation

Data store

Main app context

App defined contexts



Context Identity

Context Identity

identifier property

Context Identity

`identifier` property

- Unique across siblings

Context Identity

`identifier` property

- Unique across siblings

Context Identifier Path

Context Identity

`identifier` property

- Unique across siblings

Context Identifier Path

- Identifies a context within a context tree

Context Identity

`identifier` property

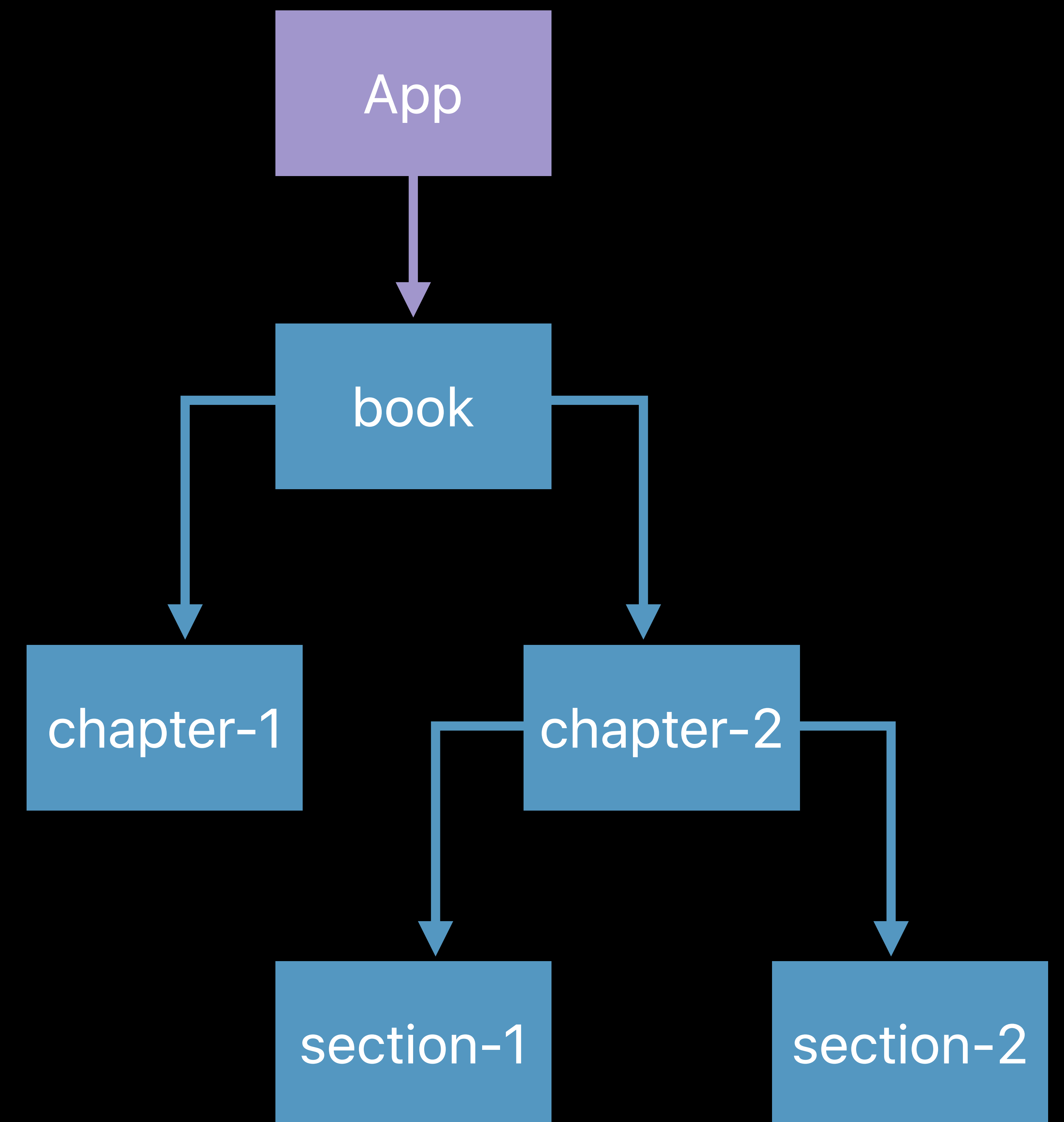
- Unique across siblings

Context Identifier Path

- Identifies a context within a context tree
- Array of context identifiers

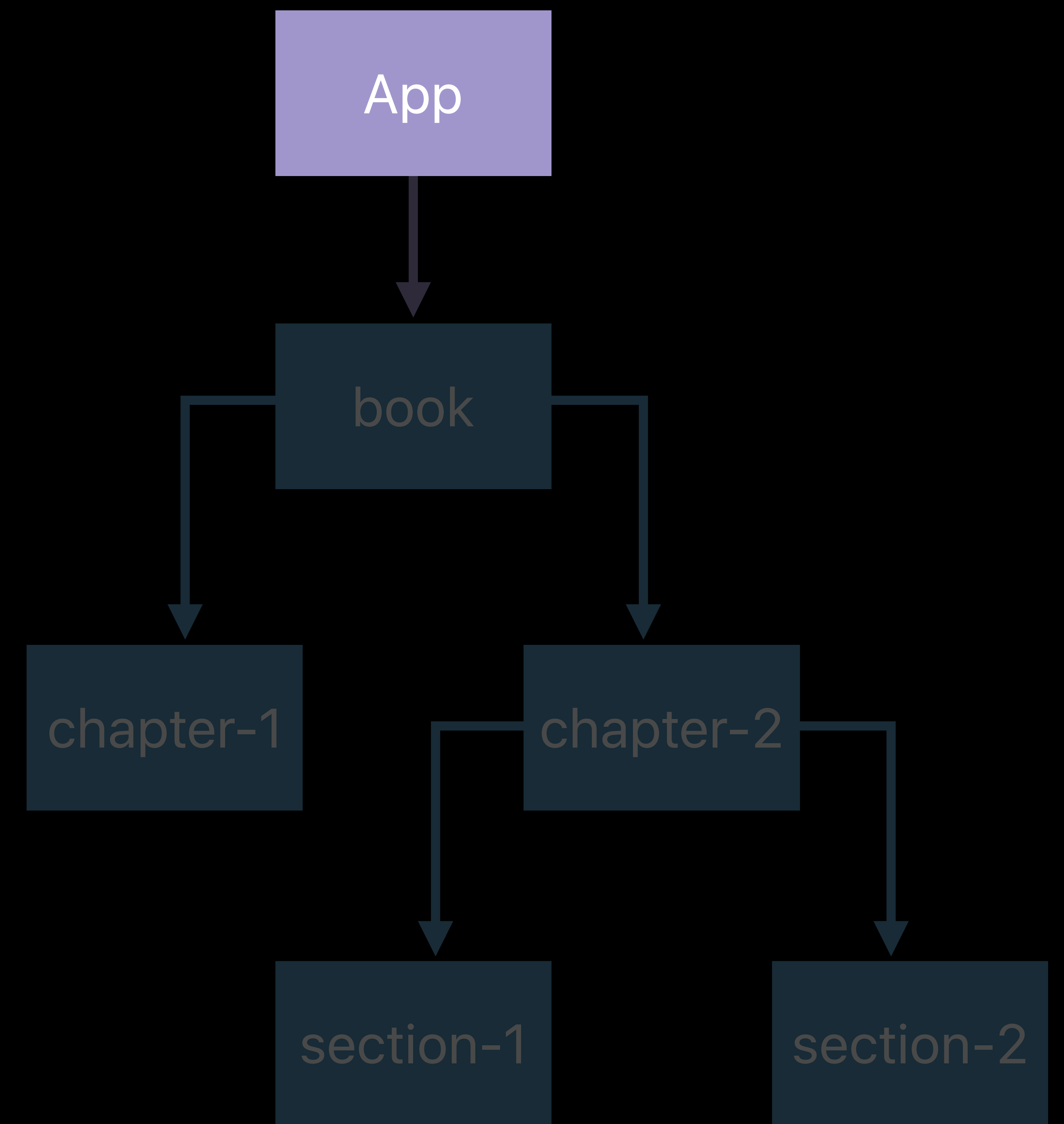
Context Identifier Path

```
["com.company.app", "book", "chapter-2", "section-1"]
```



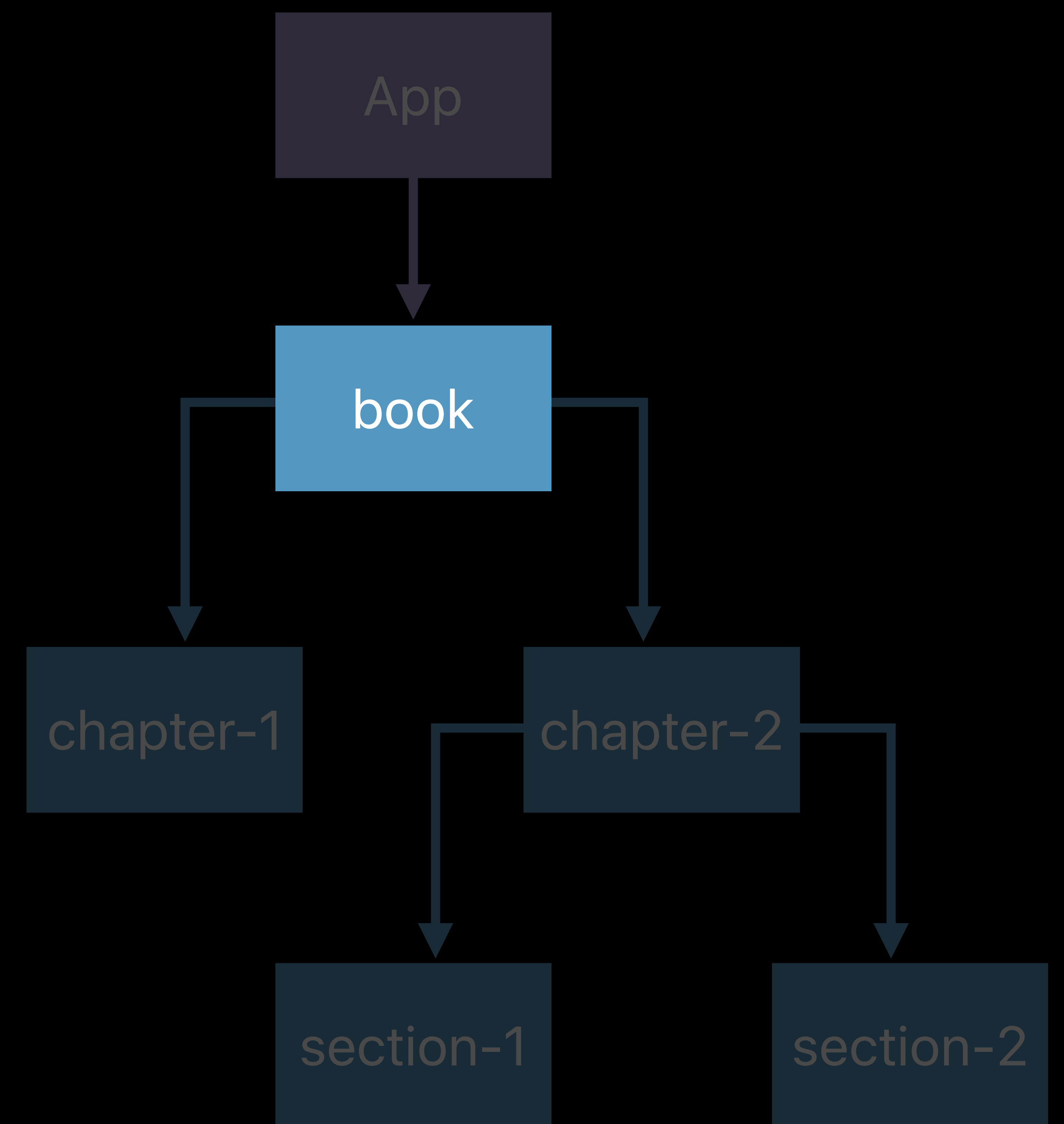
Context Identifier Path

```
["com.company.app", "book", "chapter-2", "section-1"]
```



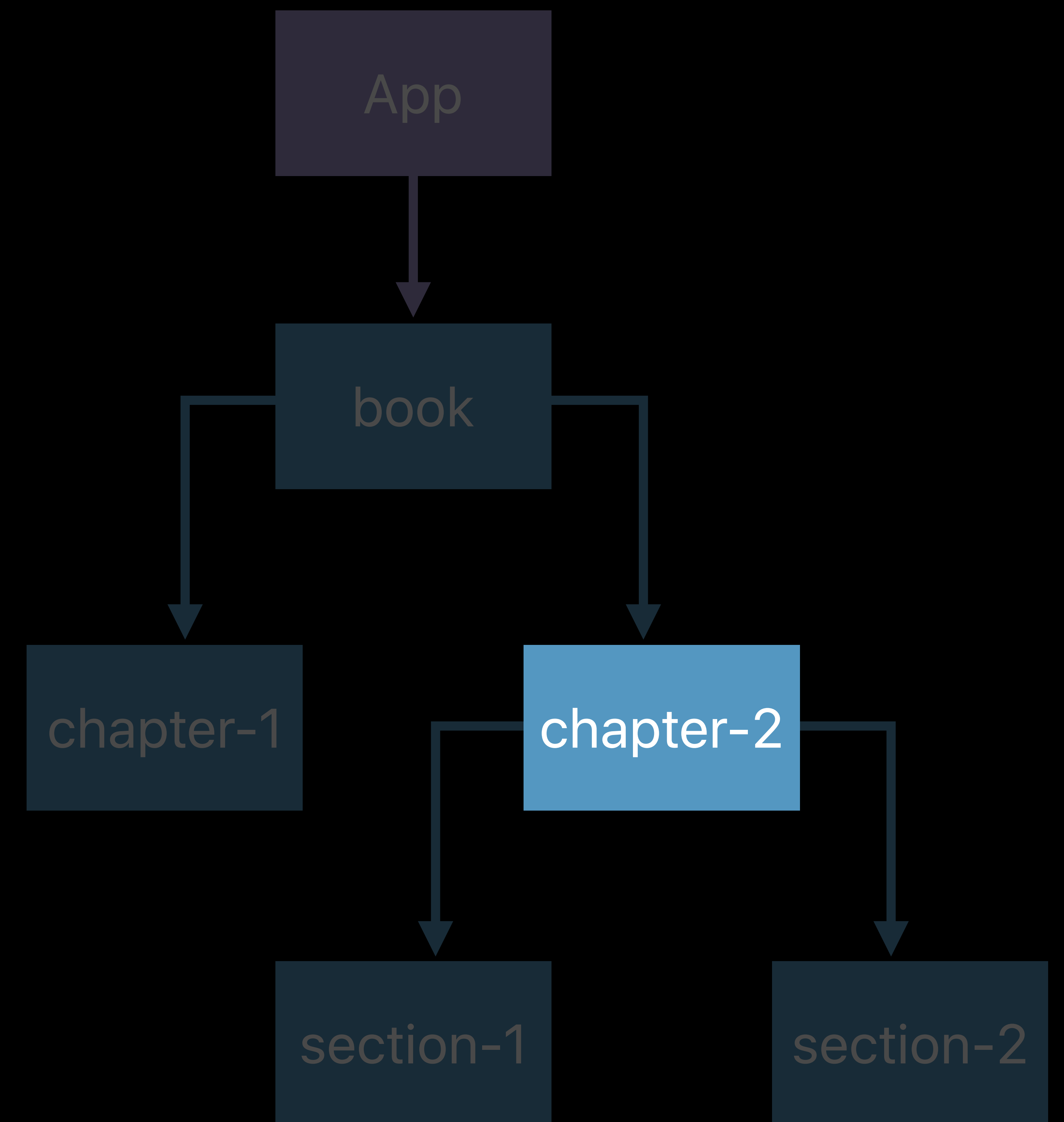
Context Identifier Path

```
["com.company.app", "book", "chapter-2", "section-1"]
```



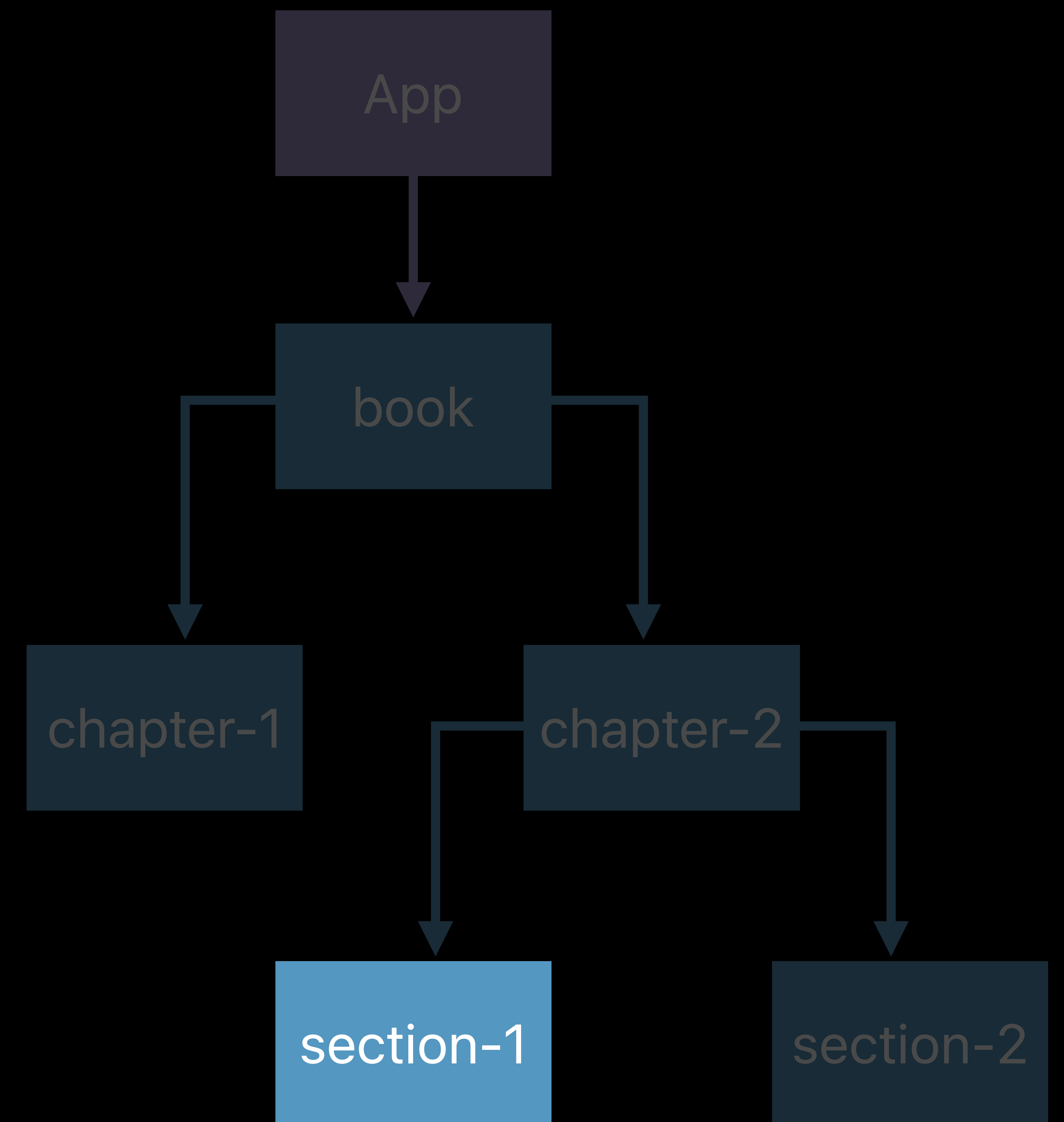
Context Identifier Path

```
["com.company.app", "book", "chapter-2", "section-1"]
```



Context Identifier Path

```
["com.company.app", "book", "chapter-2", "section-1"]
```



Context Lookup

Using absolute context path

```
let path = ["com.company.app", "book-1", "chapter-2", "section-1"]
CLSDataStore.shared.contexts(matchingIdentifierPath: path) { contextsAmongPath, error in
    // process contexts among path
}
```

Context Lookup

Using relative context path

```
let path = ["chapter-2", "section-1"]
bookContext.descendant(matchingIdentifierPath: path) { context, error in
    // use section1 context here
}
```

Context Lookup

Using predicates

```
let predicate = NSPredicate(format: "parent = %@", bookContext)
CLSDataStore.shared.contexts(matching: predicate) { contexts, error in
    // process all chapter contexts
}
```

CLSDataStoreDelegate

```
protocol CLSDataStoreDelegate {  
    func createContext(forIdentifier identifier: String,  
                       parentContext: CLSContext,  
                       parentIdentifierPath: [String]) -> CLSContext?  
}
```

CLSDataStoreDelegate

Works in conjunction with path query methods

```
protocol CLSDataStoreDelegate {  
    func createContext(forIdentifier identifier: String,  
                       parentContext: CLSContext,  
                       parentIdentifierPath: [String]) -> CLSContext?  
}
```

CLSDataStoreDelegate

Works in conjunction with path query methods

Used to create missing contexts on demand

```
protocol CLSDataStoreDelegate {  
    func createContext(forIdentifier identifier: String,  
                       parentContext: CLSContext,  
                       parentIdentifierPath: [String]) -> CLSContext?  
}
```

CLSDataStoreDelegate

Works in conjunction with path query methods

Used to create missing contexts on demand

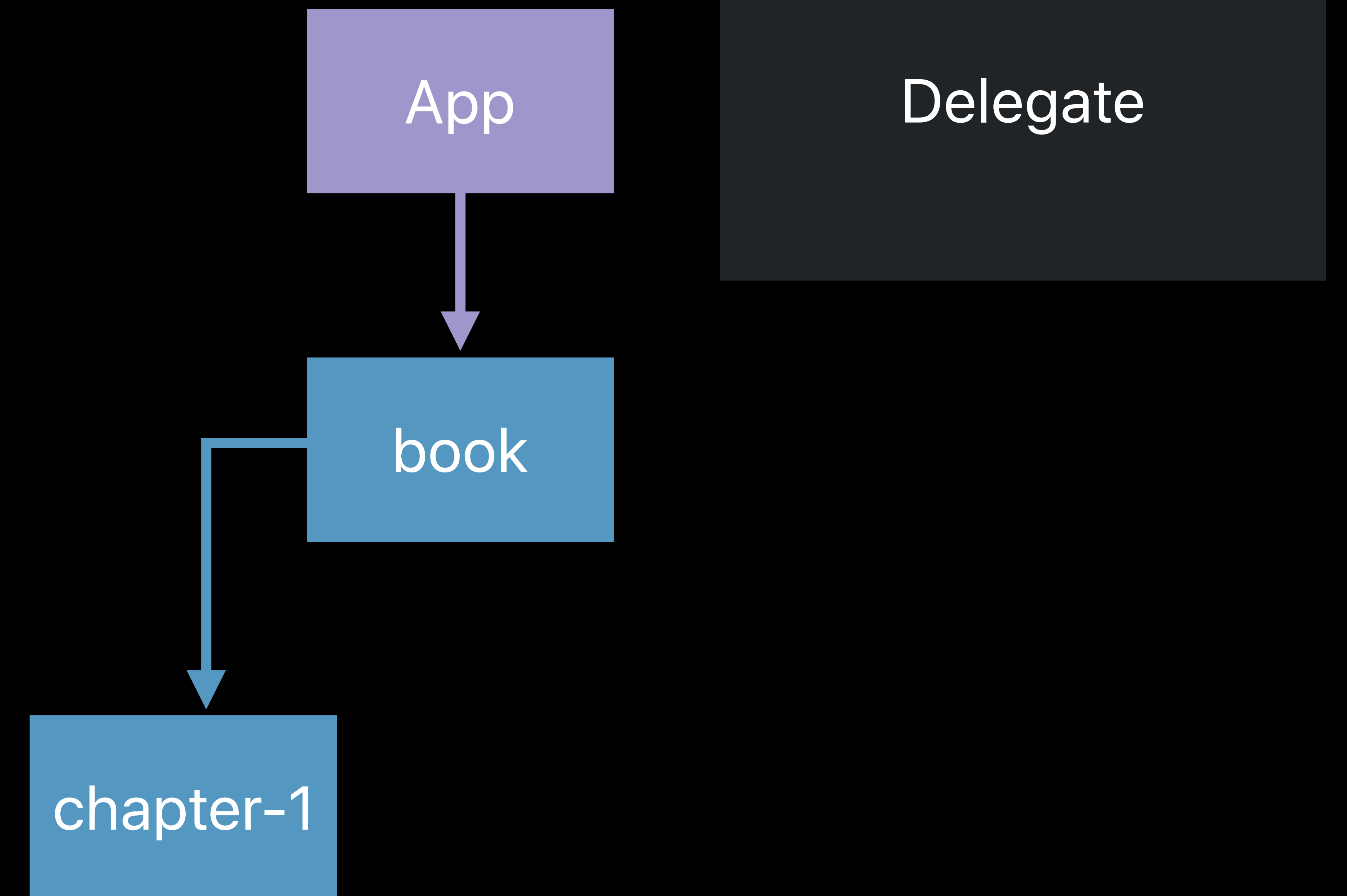
Useful for apps with dynamic content

```
protocol CLSDataStoreDelegate {  
    func createContext(forIdentifier identifier: String,  
                       parentContext: CLSContext,  
                       parentIdentifierPath: [String]) -> CLSContext?  
}
```

Delegate in Action

Defining section 1

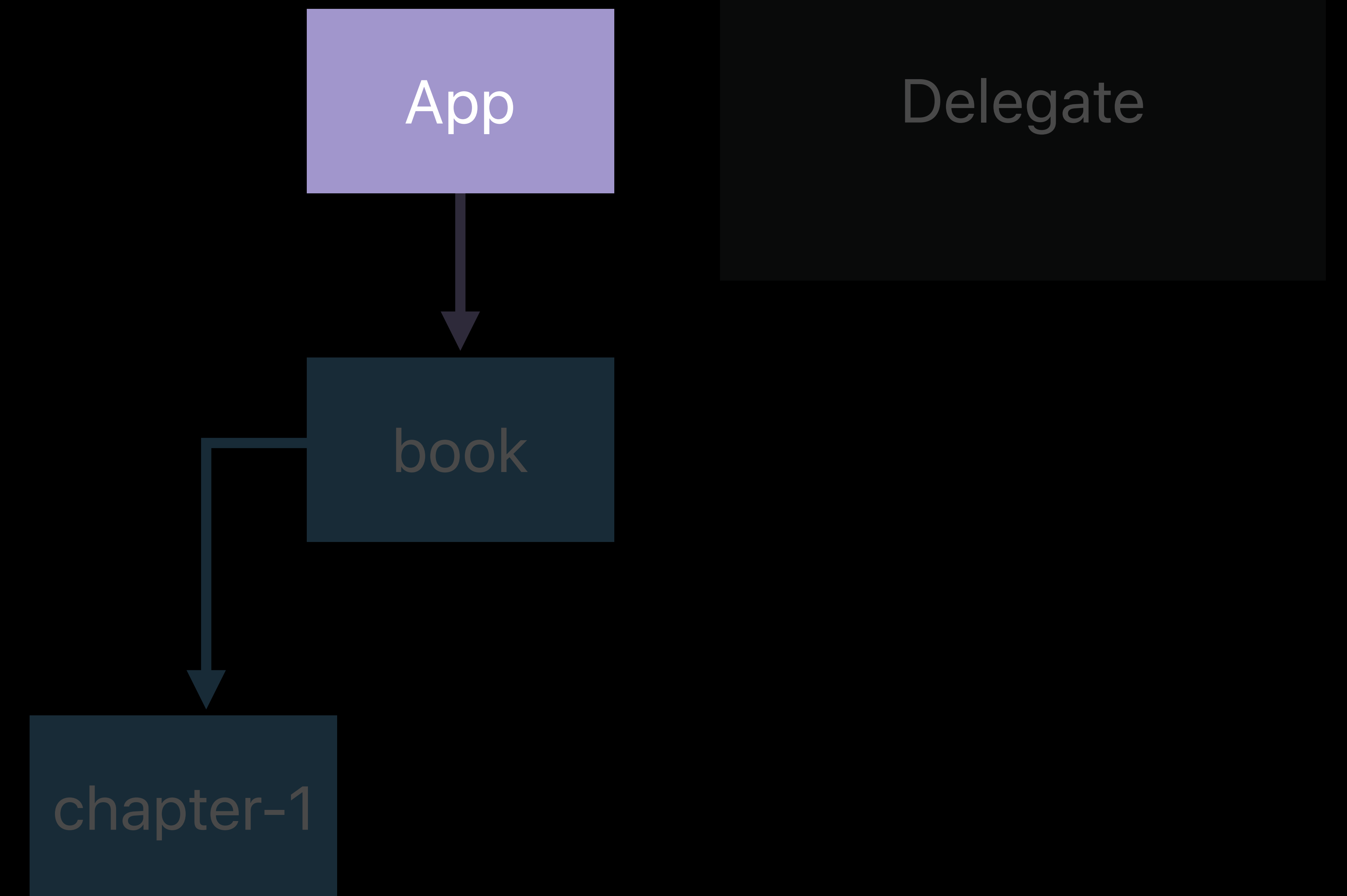
```
["com.company.app", "book", "chapter-2", "section-1"]
```



Delegate in Action

Defining section 1

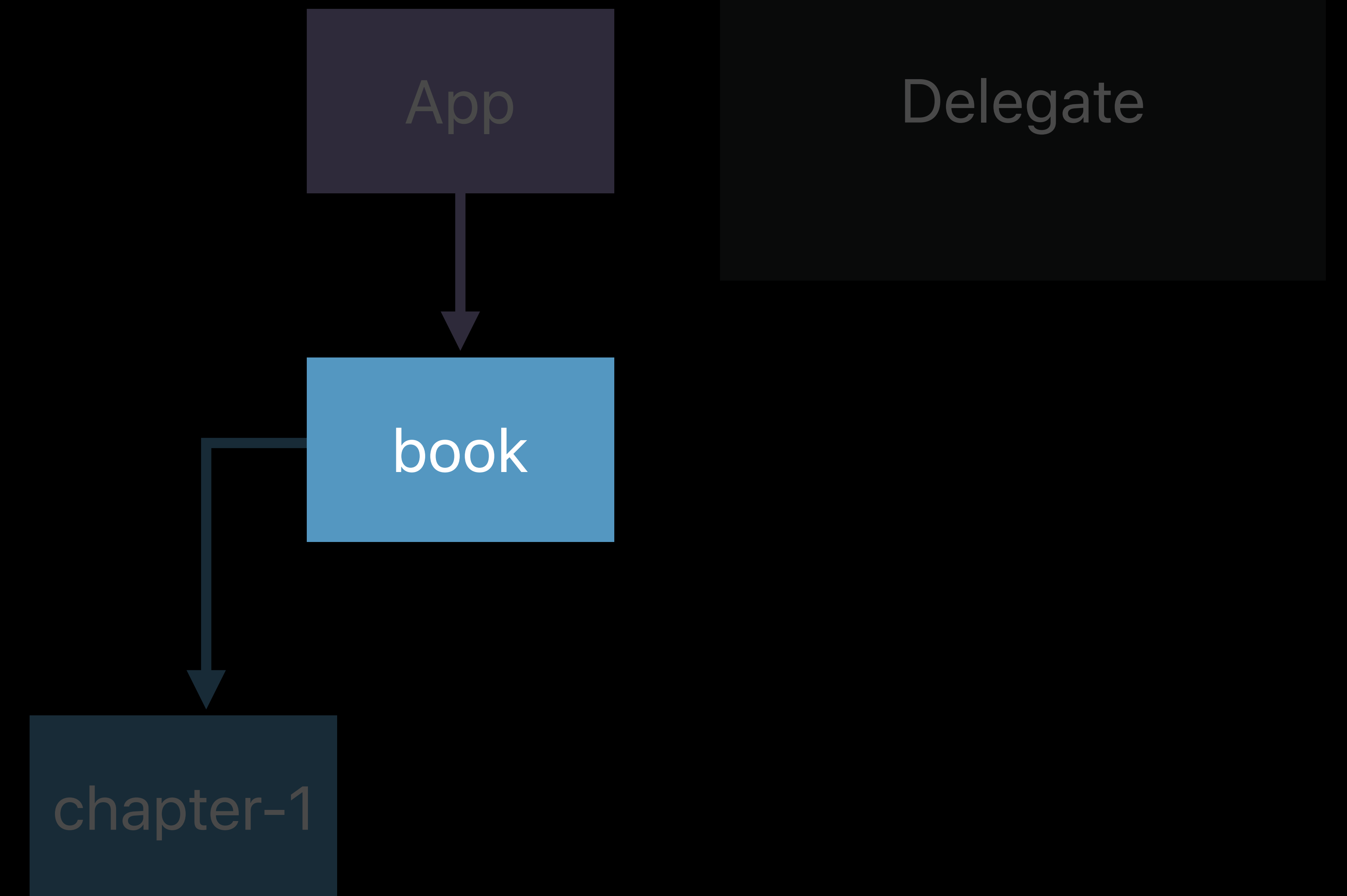
```
["com.company.app", "book", "chapter-2", "section-1"]
```



Delegate in Action

Defining section 1

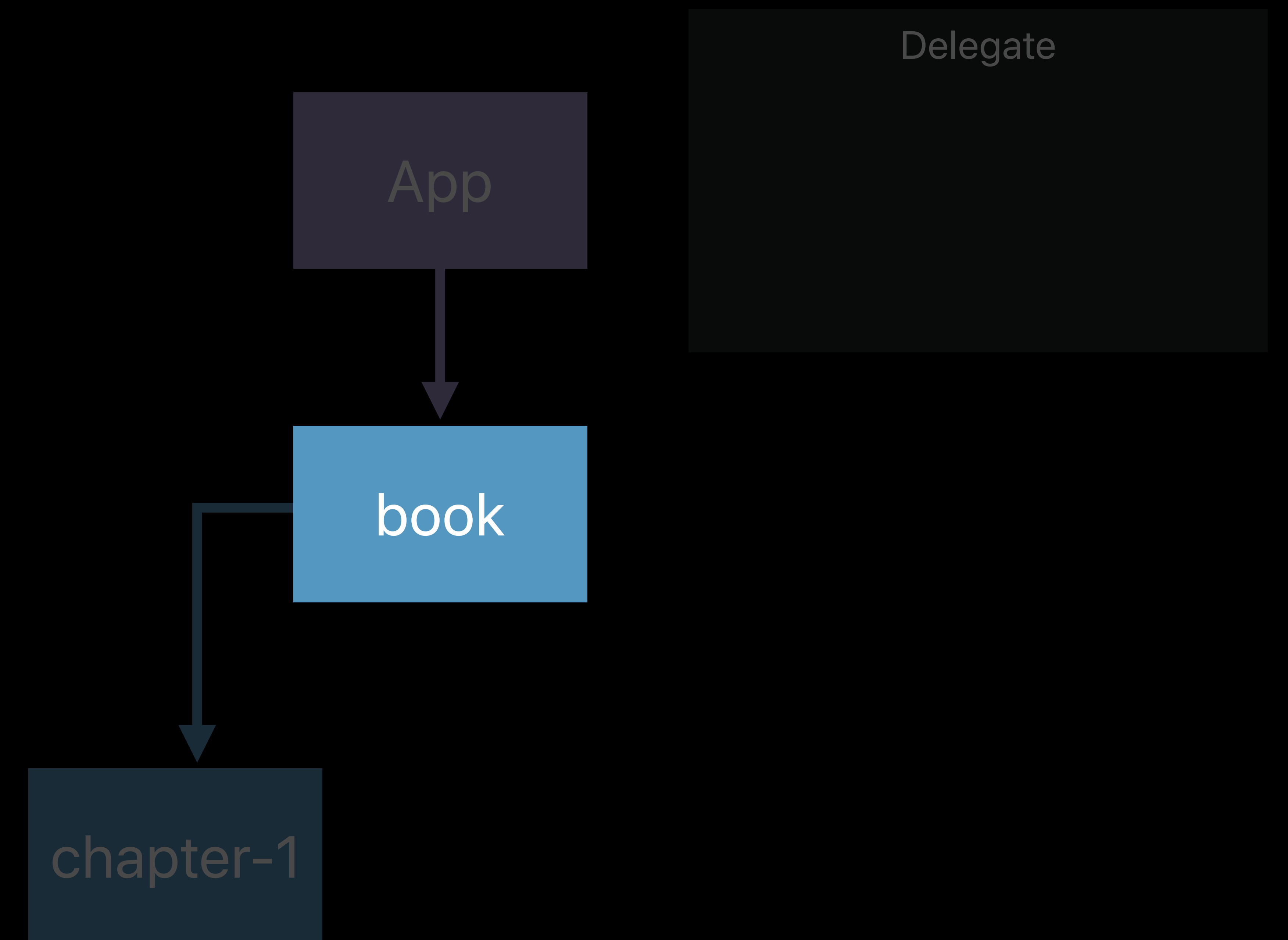
```
[ "com.company.app", "book", "chapter-2", "section-1" ]
```



Delegate in Action

Defining section 1

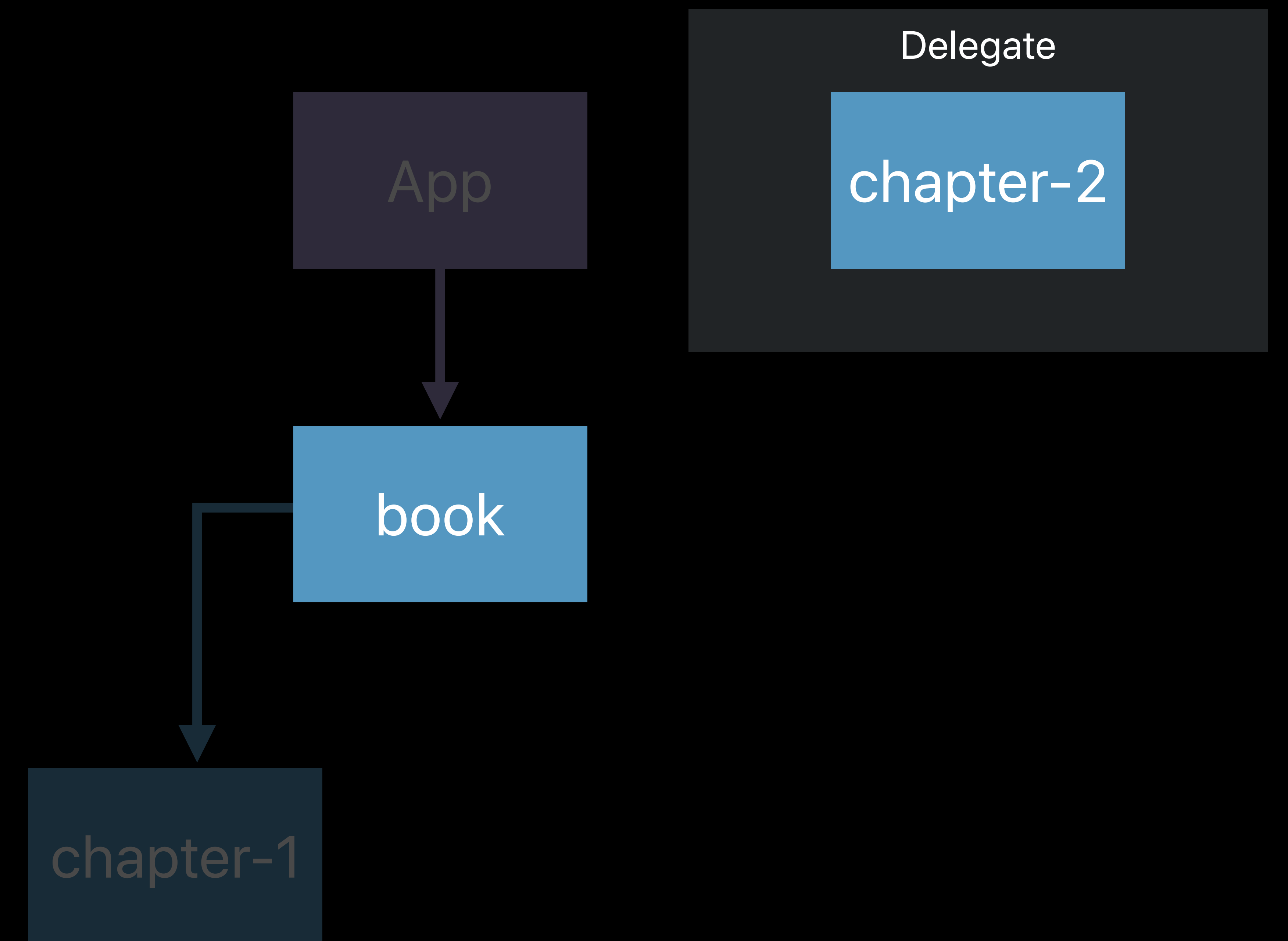
```
["com.company.app", "book", "chapter-2", "section-1"]
```



Delegate in Action

Defining section 1

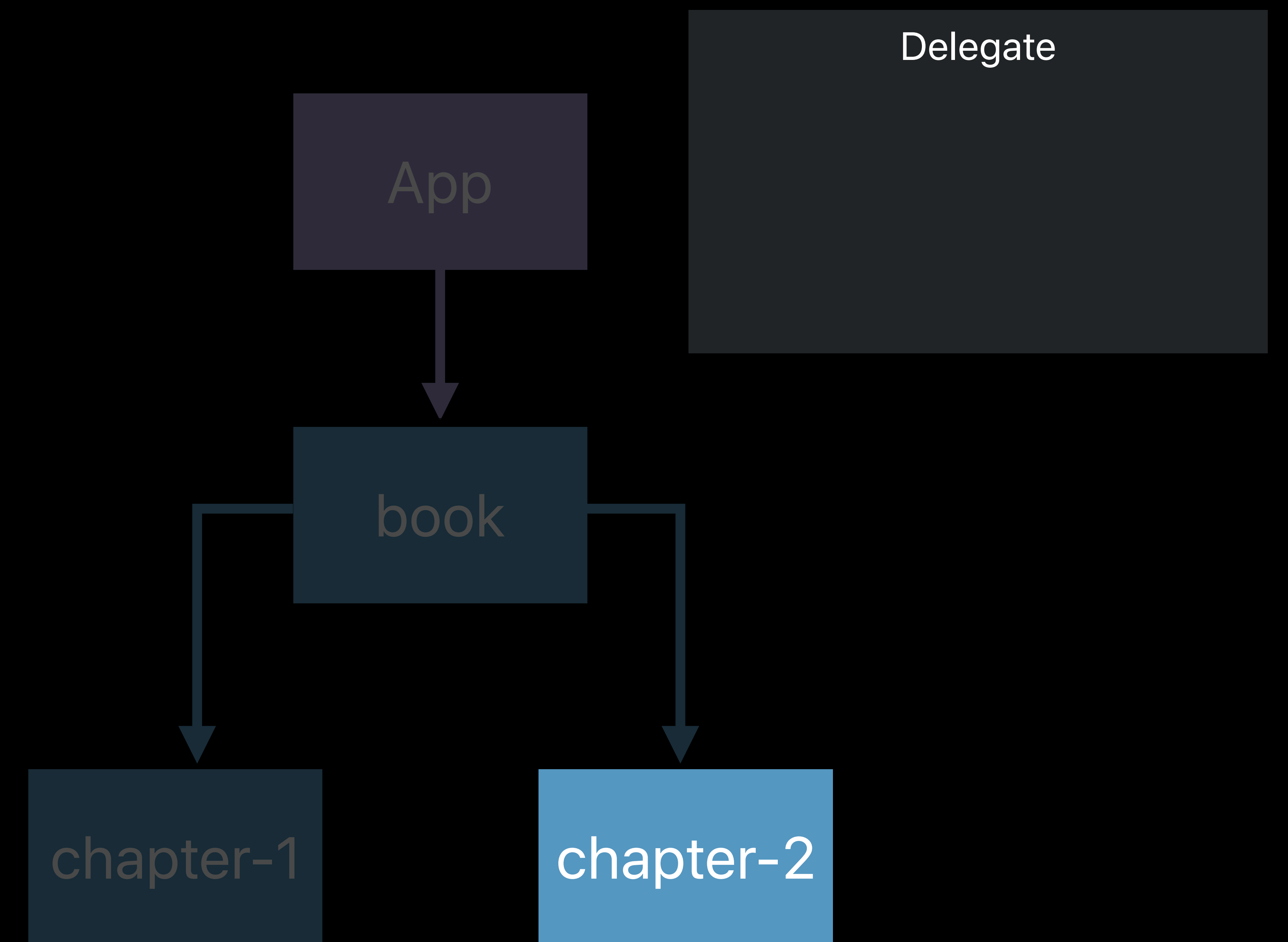
```
["com.company.app", "book", "chapter-2", "section-1"]
```



Delegate in Action

Defining section 1

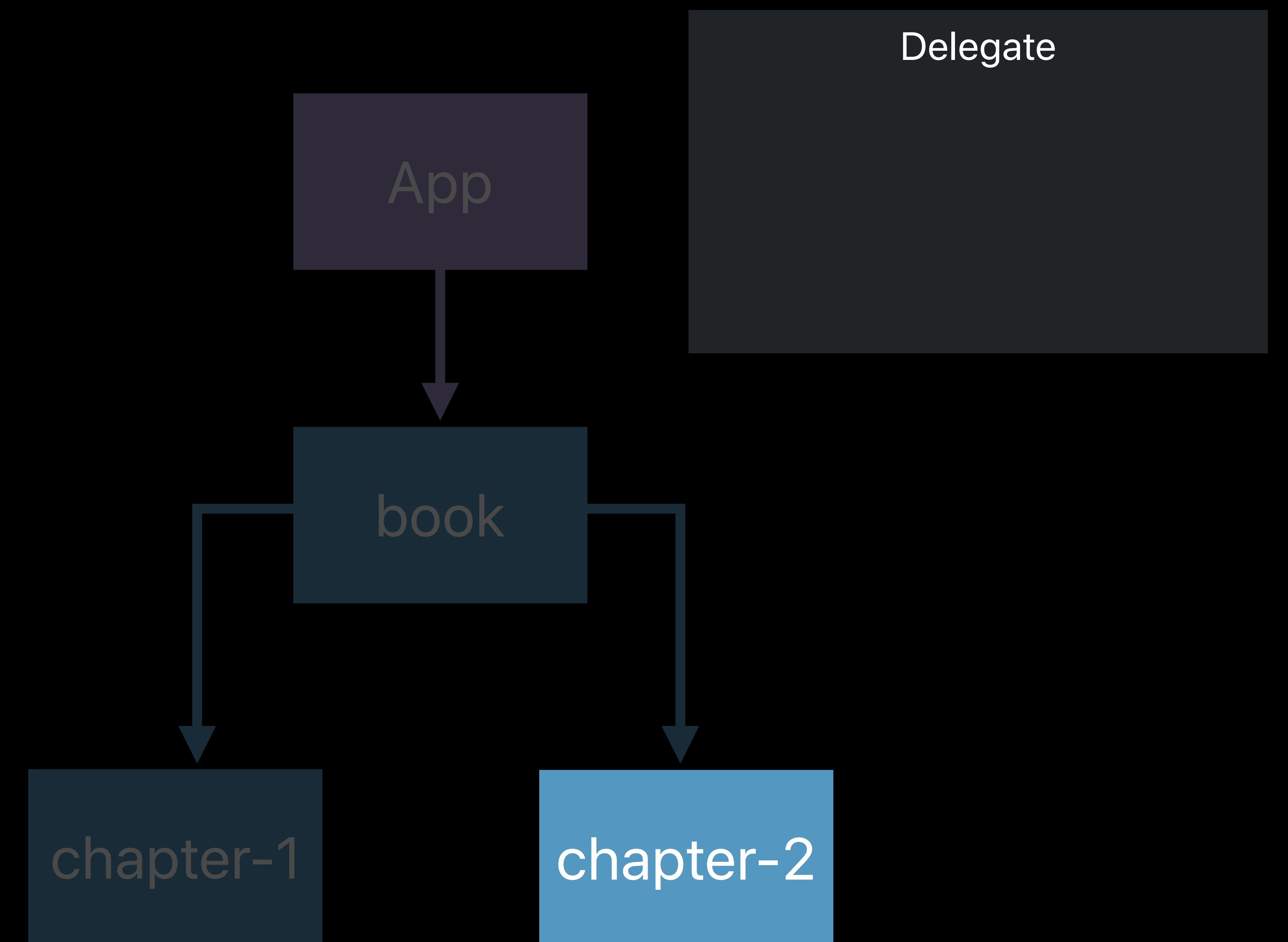
```
["com.company.app", "book", "chapter-2", "section-1"]
```



Delegate in Action

Defining section 1

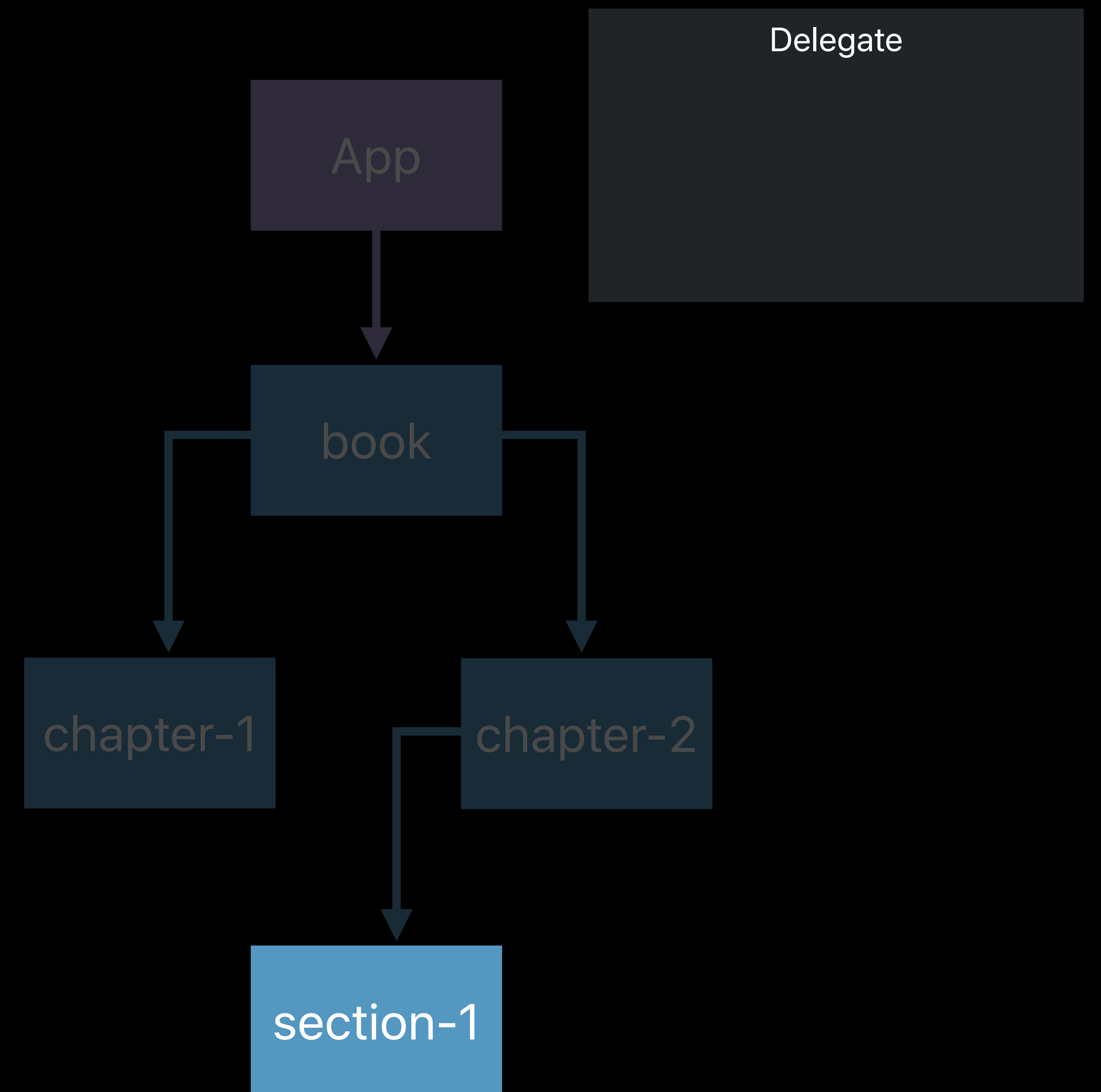
```
["com.company.app", "book", "chapter-2", "section-1"]
```



Delegate in Action

Defining section 1

```
["com.company.app", "book", "chapter-2", "section-1"]
```



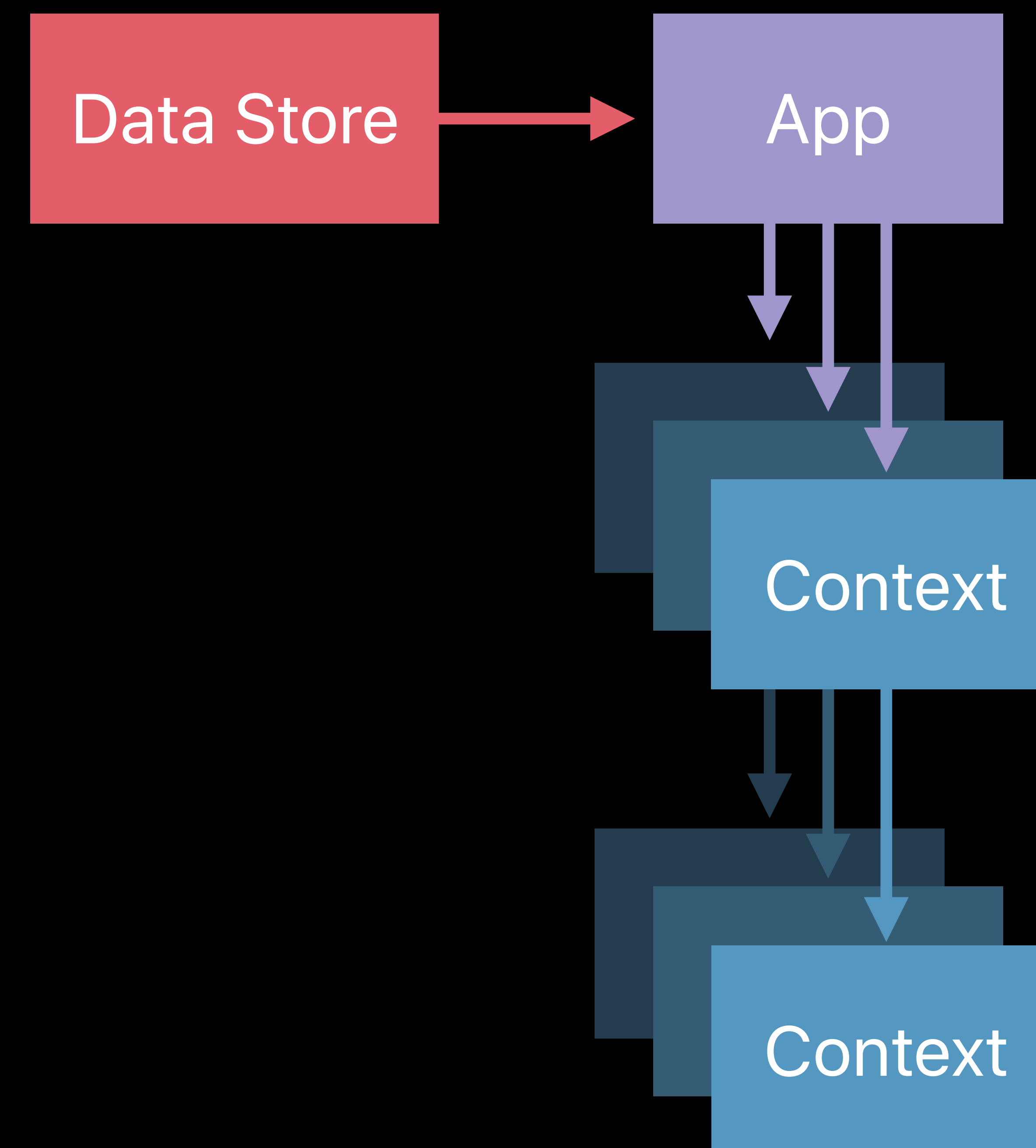
Object Graph

CLSContext

Data store

Main app context

App defined contexts



Object Graph

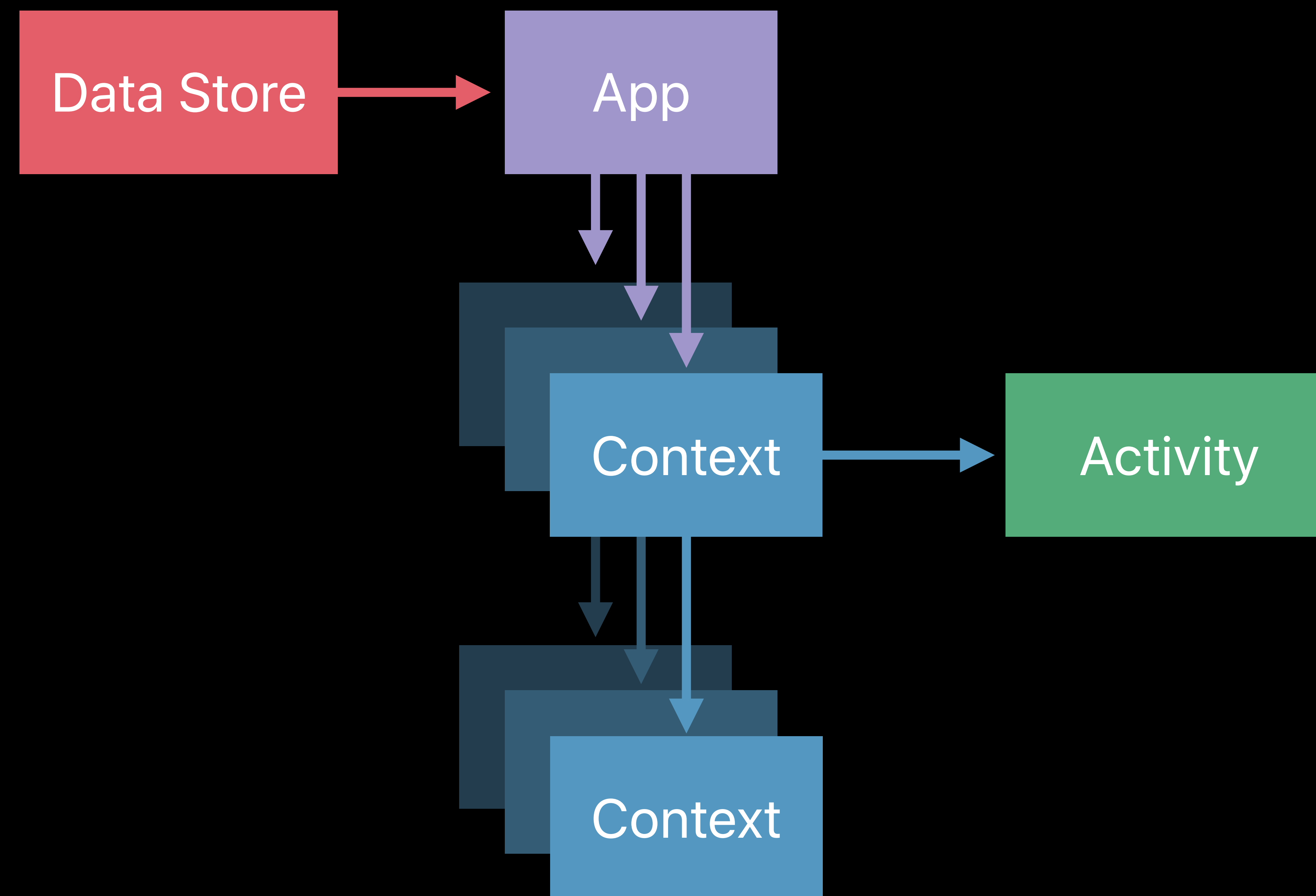
CLSActivity

Data store

Main app context

App defined contexts

Activity



CLSActivity

CLSActivity

Always associated with contexts

CLSActivity

Always associated with contexts

Created by calling `CLSContext.createNewActivity() -> CLSActivity`

CLSActivity

Always associated with contexts

Created by calling `CLSContext.createNewActivity() -> CLSActivity`

Access current using `CLSContext.currentActivity: CLSActivity?`

CLSActivity

Always associated with contexts

Created by calling `CLSContext.createNewActivity() -> CLSActivity`

Access current using `CLSContext.currentActivity: CLSActivity?`

Creating a new activity is akin to a new attempt

CLSActivity

Adding progress

```
// adding progress  
activity.progress = 0.5 // 50%  
  
// or  
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
```

CLSActivity

Adding progress

```
// adding progress
```

```
activity.progress = 0.5 // 50%
```

```
// or
```

```
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
```


CLSActivity

Adding progress

```
// adding progress  
activity.progress = 0.5 // 50%  
  
// or  
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
```

CLSActivity

Adding progress

```
// adding progress  
activity.progress = 0.5 // 50%  
  
// or  
activity.addProgressRange(fromStart: 0.2, toEnd: 0.7)
```

Object Graph

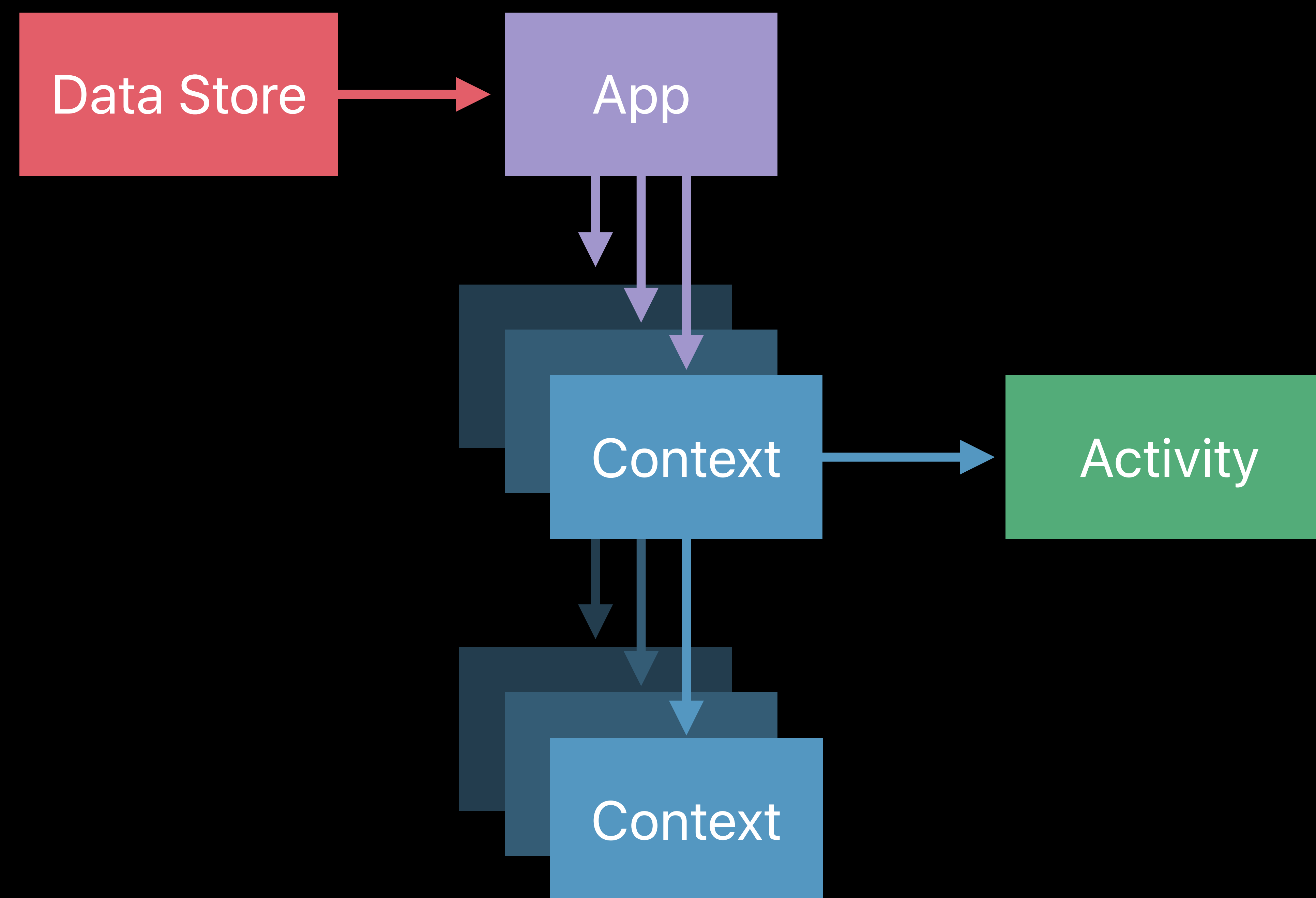
CLSActivity

Data store

Main app context

App defined contexts

Activity



Object Graph

CLSActivityItem

Data store

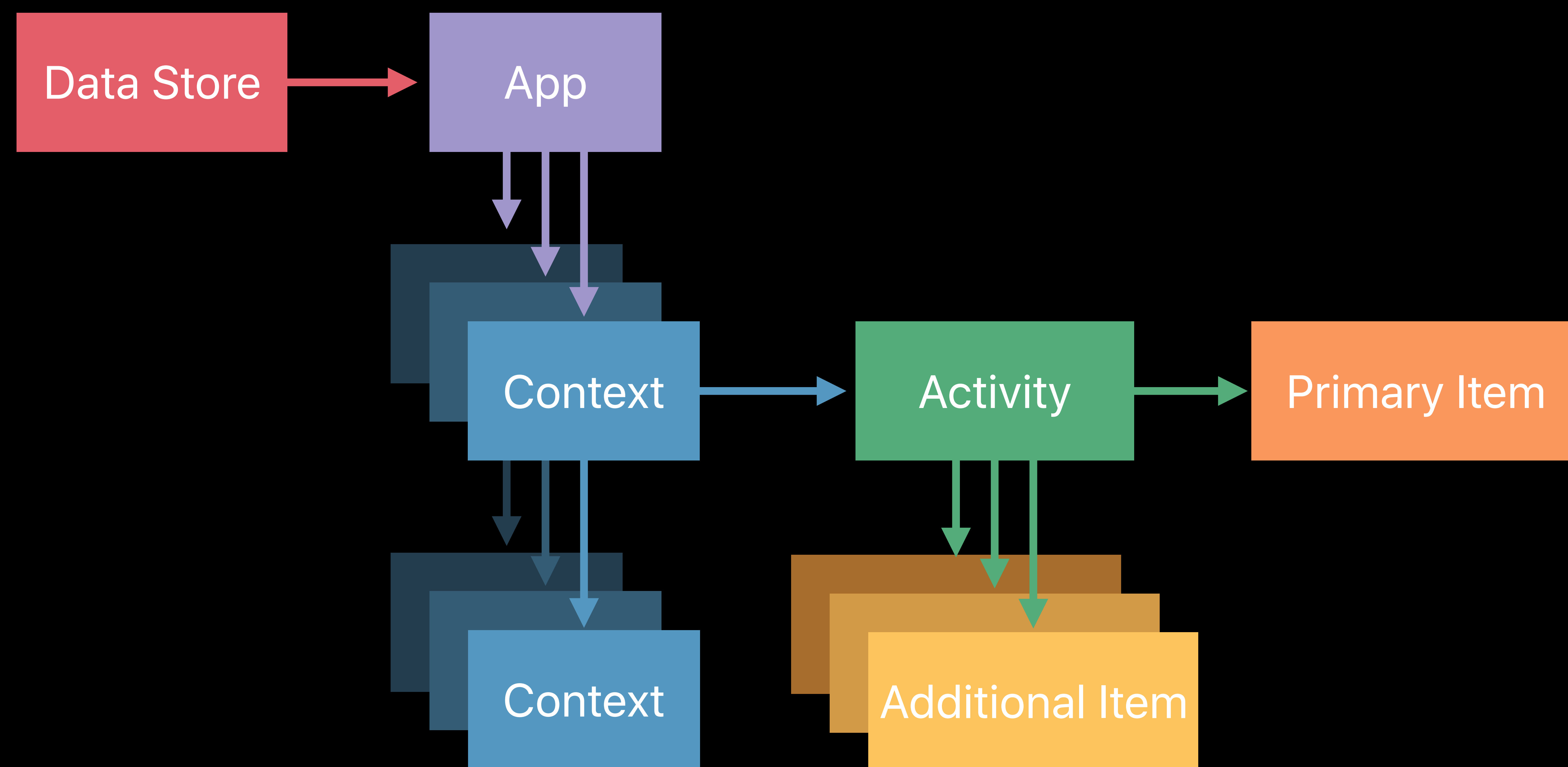
Main app context

App defined contexts

Activity

Primary activity item

Additional activity items



CLSActivityItem

Abstract class with three subclasses

CLSActivityItem

Abstract class with three subclasses

CLSQuantityItem

- Used for scalar values, (e.g. 5 Hints)

CLSActivityItem

Abstract class with three subclasses

CLSQuantityItem

- Used for scalar values, (e.g. 5 Hints)

CLSScoreItem

- Used for X out of Y values (e.g. 5 out 7 questions answered)

CLSActivityItem

Abstract class with three subclasses

CLSQuantityItem

- Used for scalar values, (e.g. 5 Hints)

CLSScoreItem

- Used for X out of Y values (e.g. 5 out 7 questions answered)

CLSBinaryItem

- Used for binary values

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score
```

```
// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```


CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Adding activity items

```
// adding primary item
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score

// adding additional item
let hints = CLSQuantityItem(identifier: "hints", title: "Hints")
hints.quantity = 3

activity.addAdditionalActivityItem(hints)
```

CLSActivityItem

Best practices

CLSActivityItem

Best practices

Always set the same subclass for primary activity items

CLSActivityItem

Best practices

Always set the same subclass for primary activity items

Provide clear and concise titles

CLSActivityItem

Best practices

Always set the same subclass for primary activity items

Provide clear and concise titles

Make use of additional activity items

Adopting ClassKit

Marin Eubanks, ClassKit team

Quizzler

Select which difficulty level you would like to play.

Addition Quiz

Multiplication Quiz

View score board or start quiz.

View Score Board

Start Quiz

$$x + 6 = 10$$

0

5

8

4

9:41 Mon Jun 4

100%

[← Addition Math Quiz](#)

Score

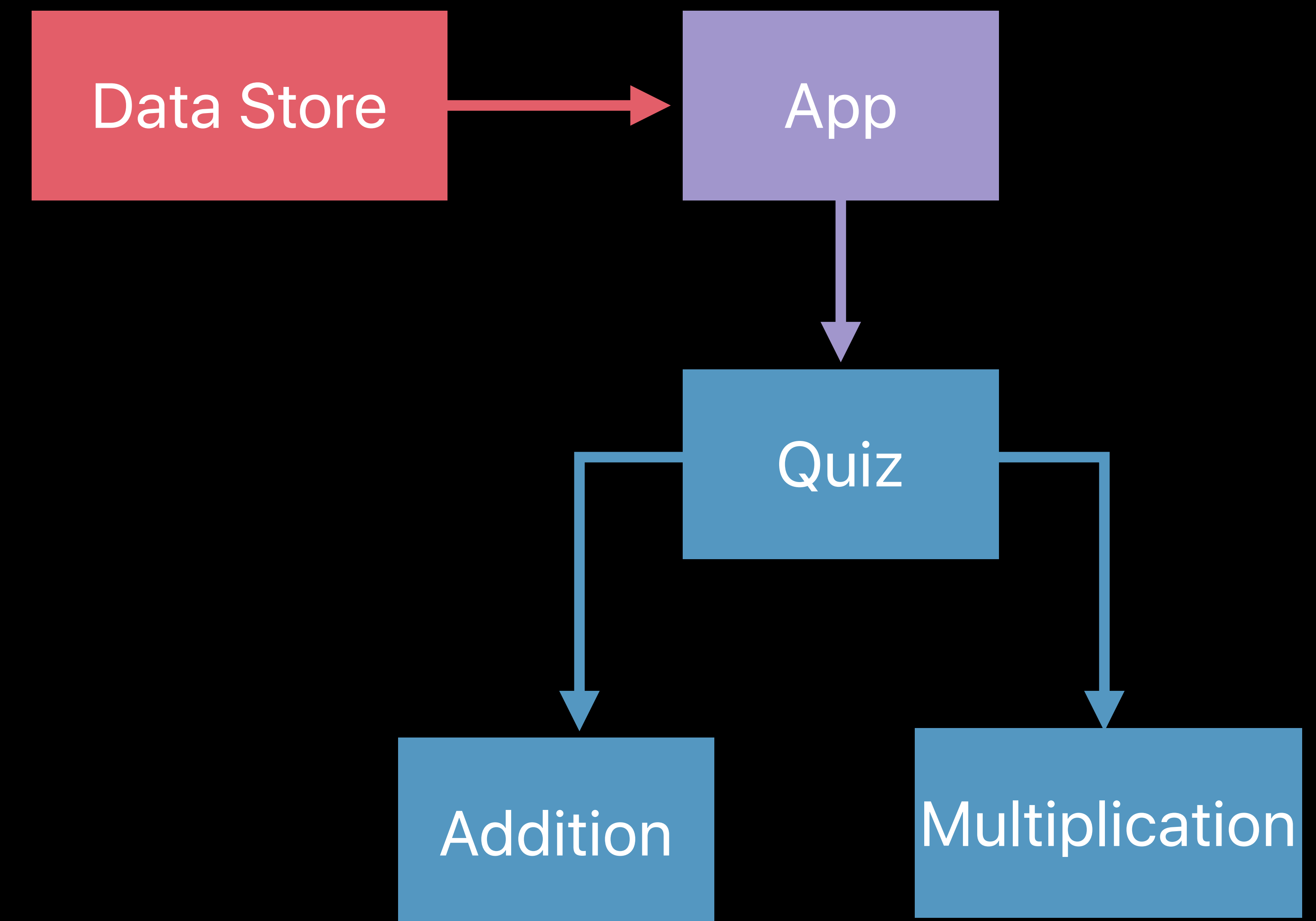
[Done](#)

100%

Steps for Adopting ClassKit

Steps for Adopting ClassKit

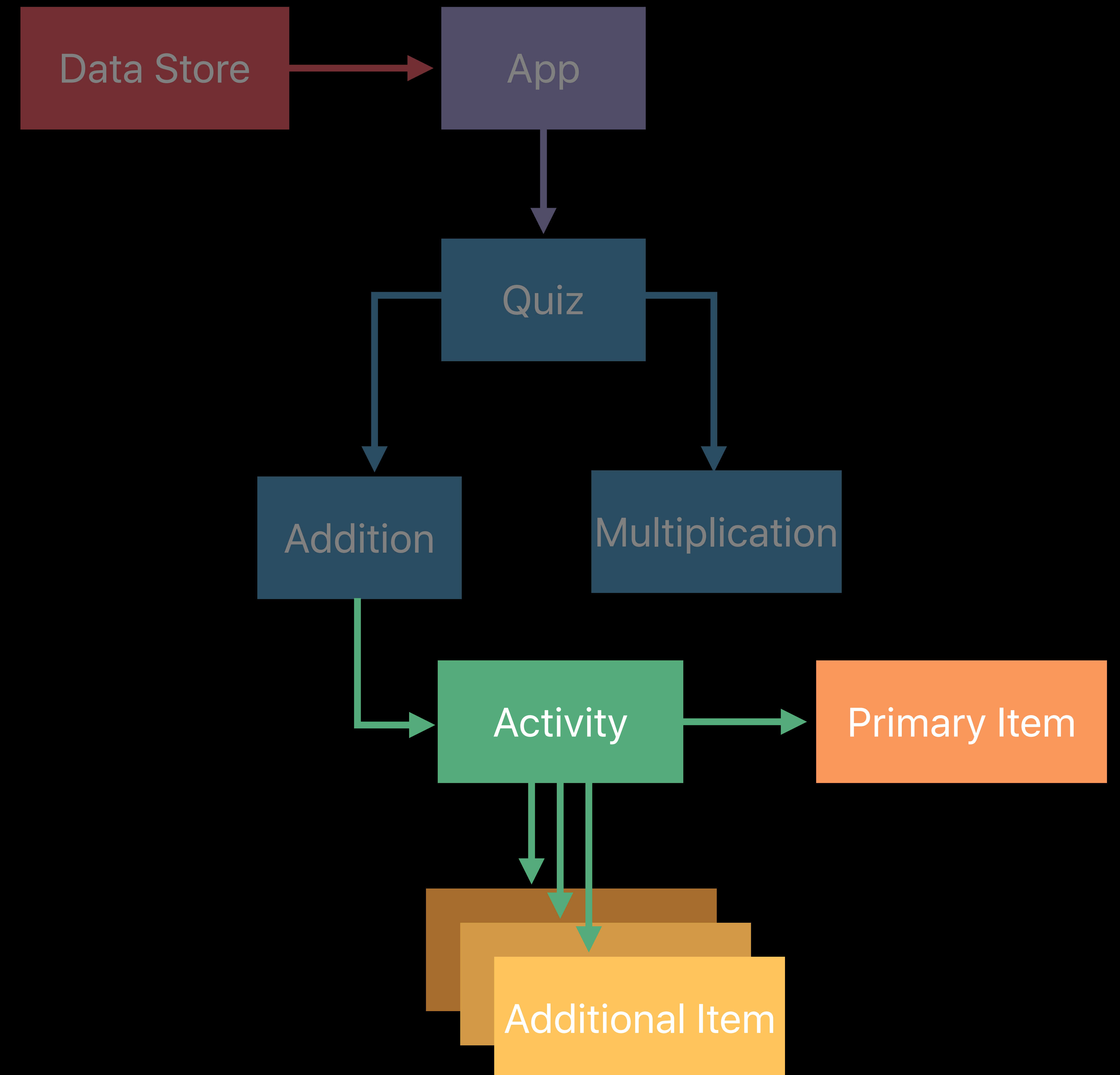
Define `CLKContext` structure



Steps for Adopting ClassKit

Define `CLSContext` structure

Add `CLSActivity` and `CLSActivityItems`



Steps for Adopting ClassKit

Define `CLContext` structure

Add `CLSActivity` and `CLSActivityItems`

Support deep linking

```
func application(_ application: UIApplication,  
                continue userActivity: NSUserActivity,  
                restorationHandler: @escaping ([UIUserActivityRestoring]?) -> Void) -> Bool
```

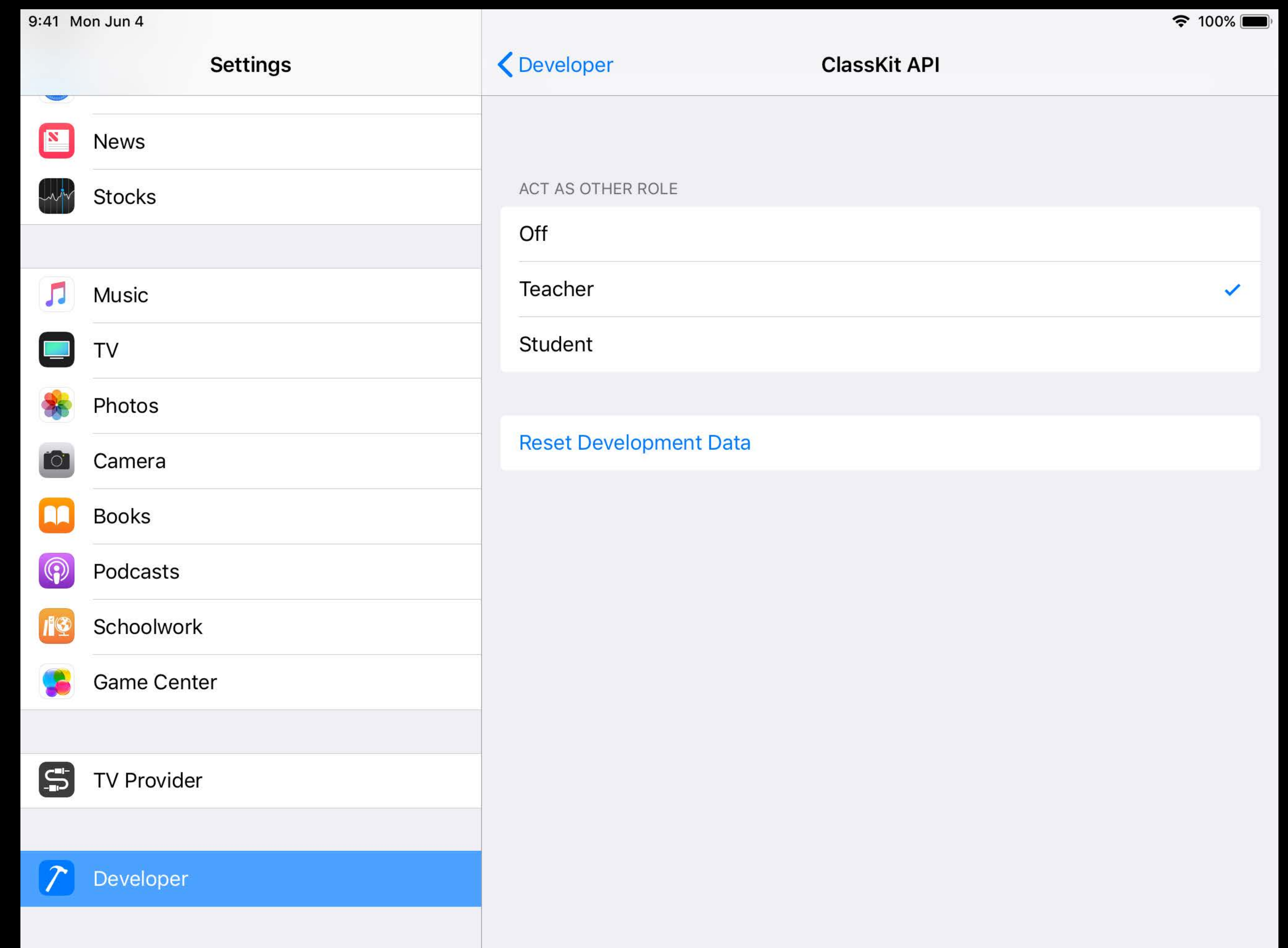
Steps for Adopting ClassKit

Define `CLContext` structure

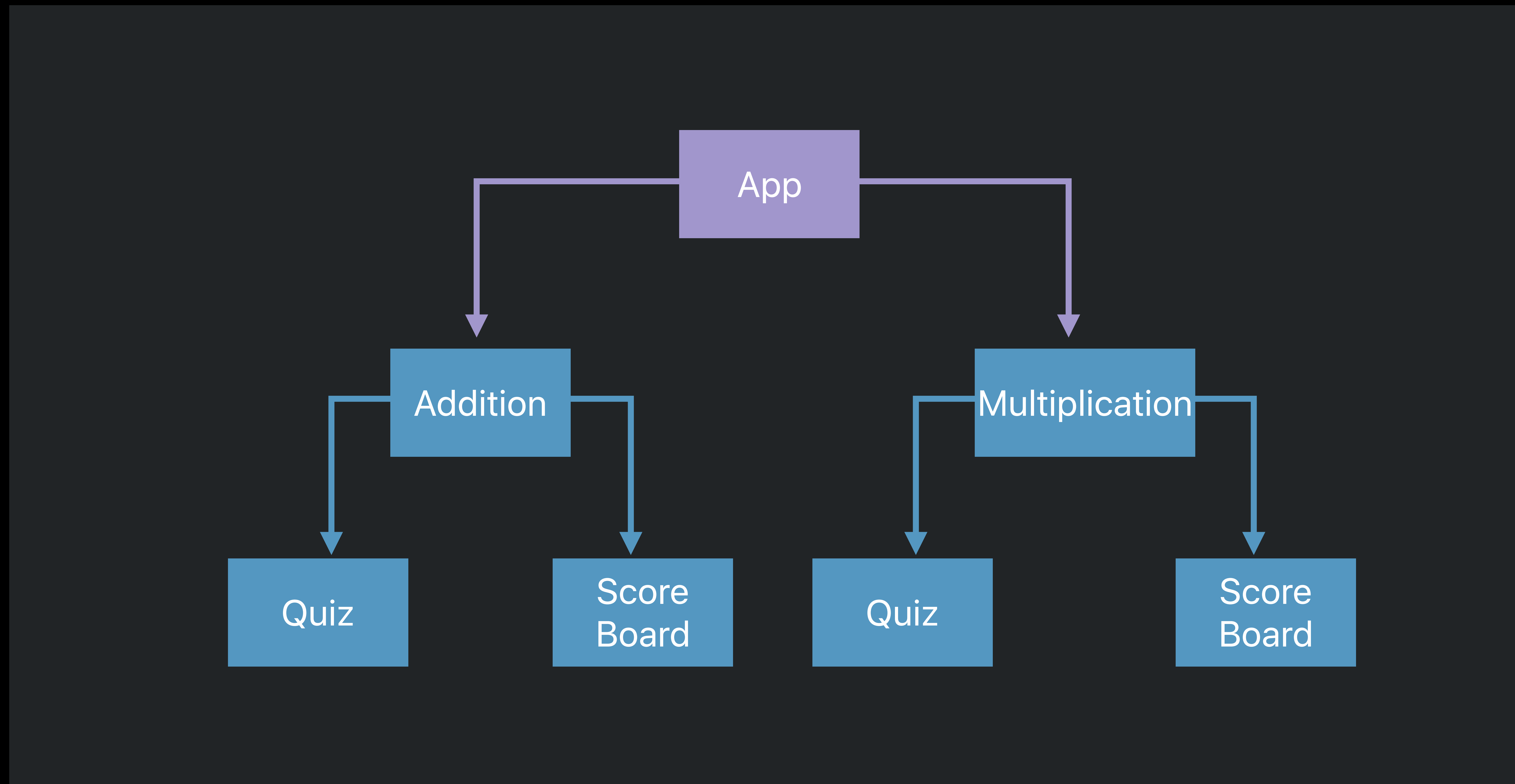
Add `CLSAActivity` and `CLSAActivityItems`

Support deep linking

Test with Developer Mode and Schoolwork

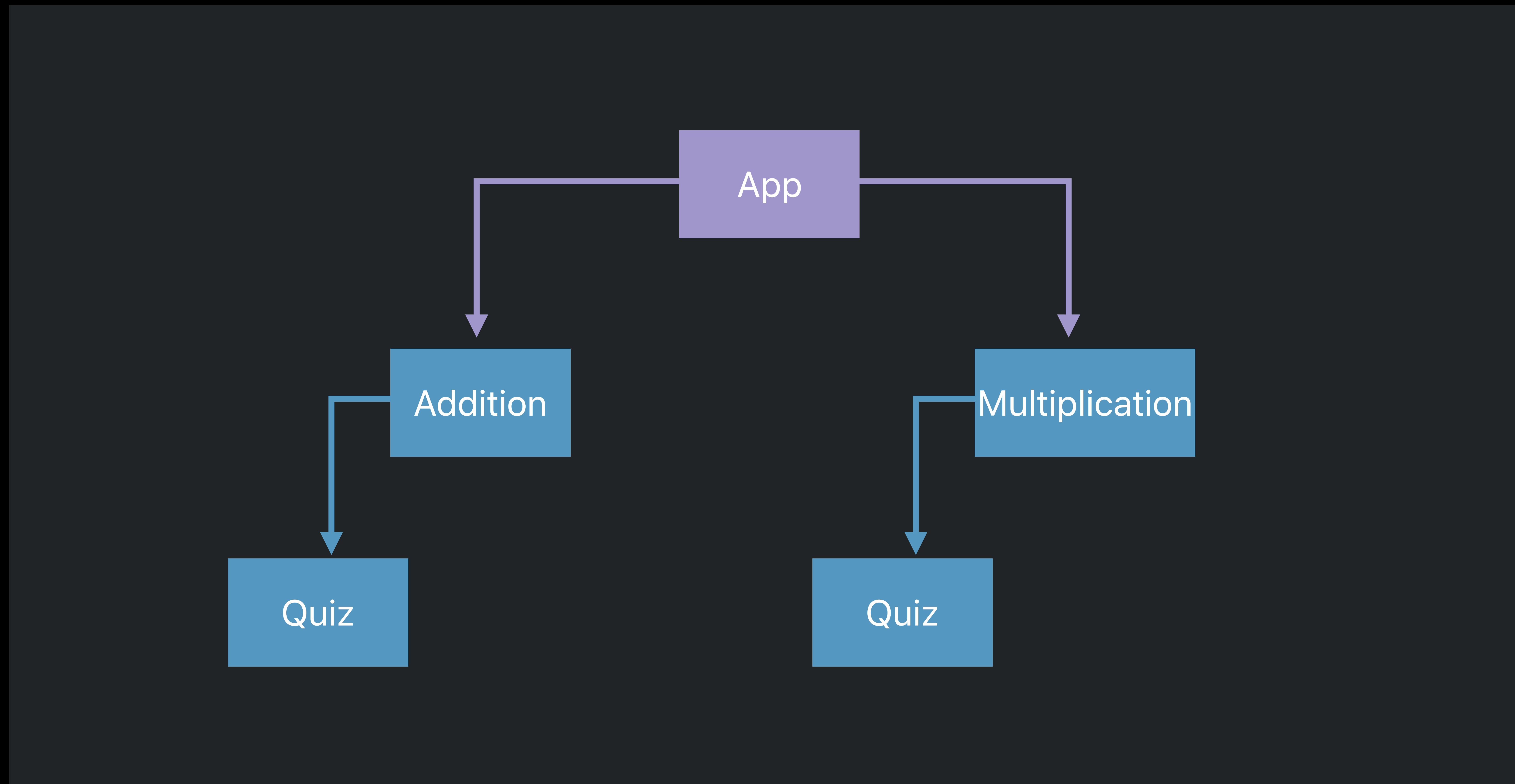


Defining CLSContext Tree



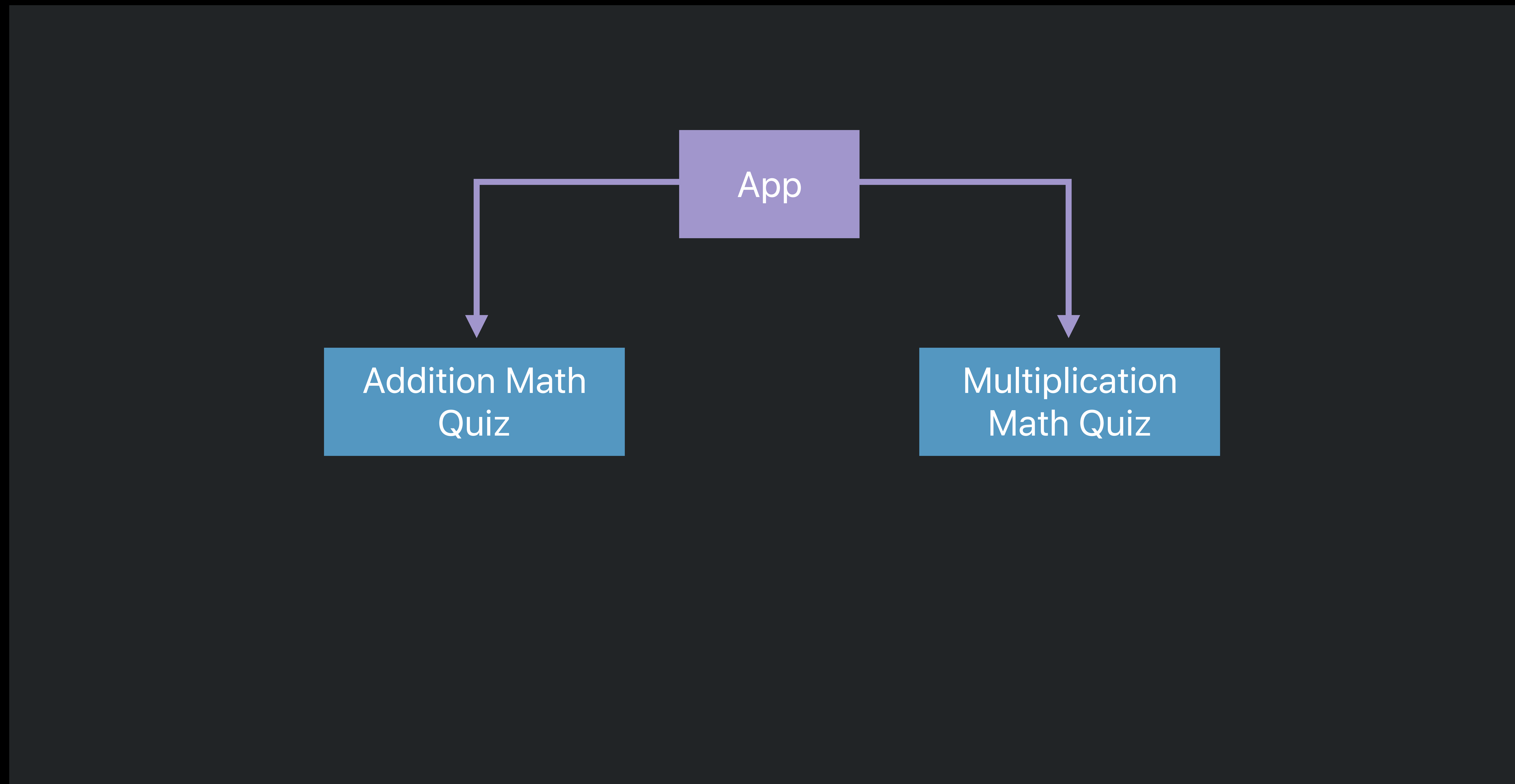
Quizzler.app

Defining CLSContext Tree



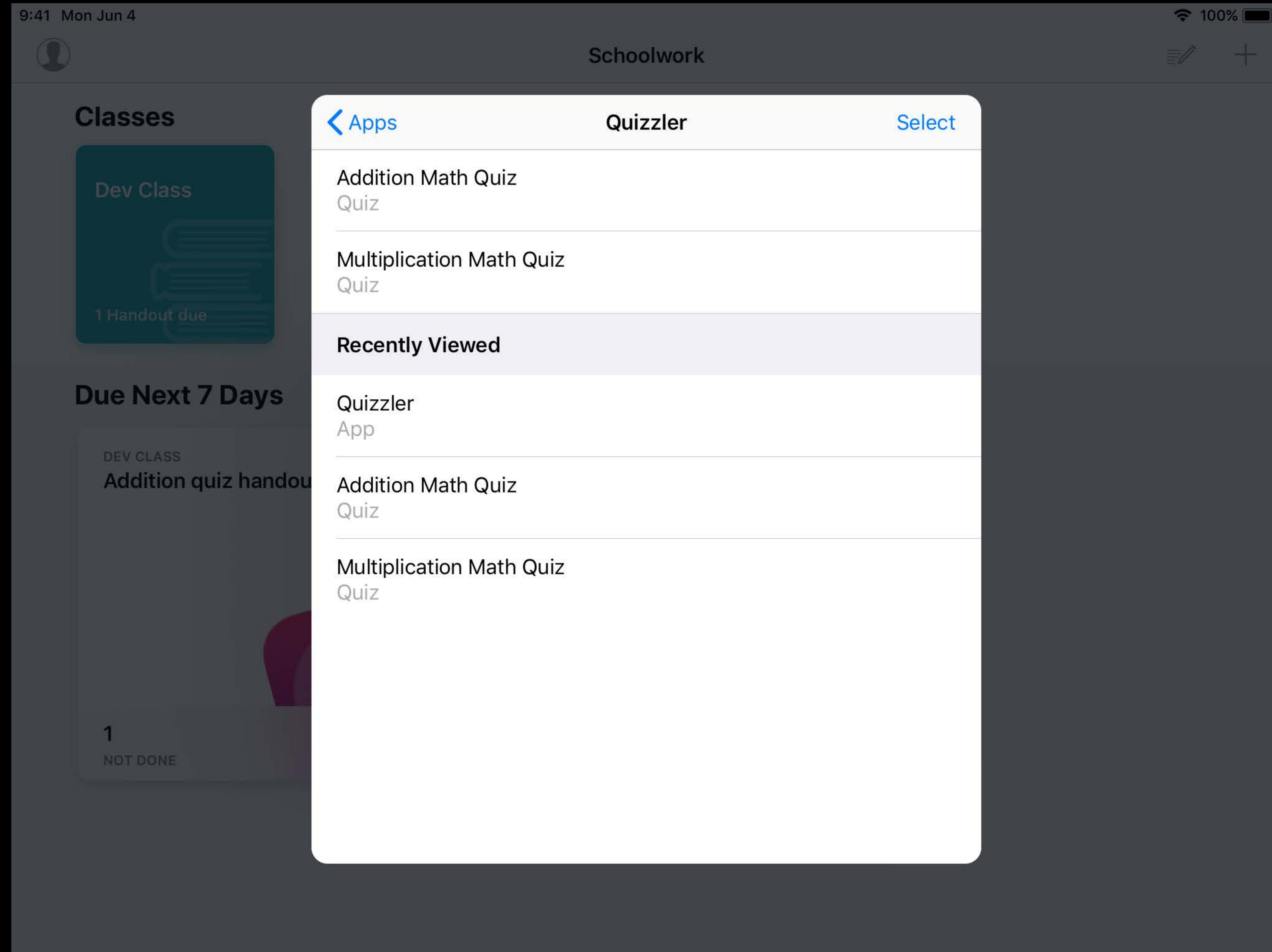
Quizzler.app

Defining CLSContext Tree



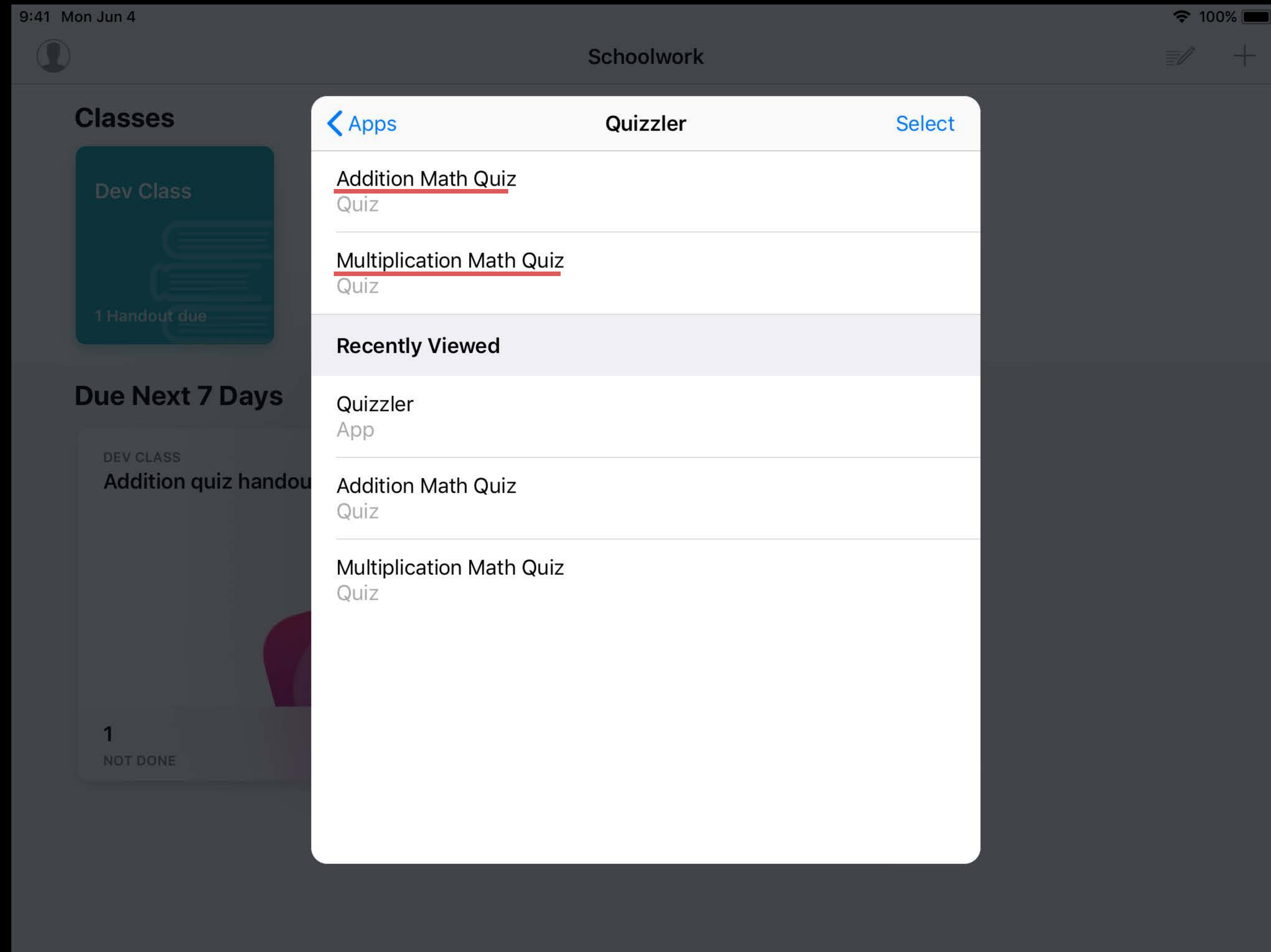
Quizzler.app

CLSContext



CLSContext

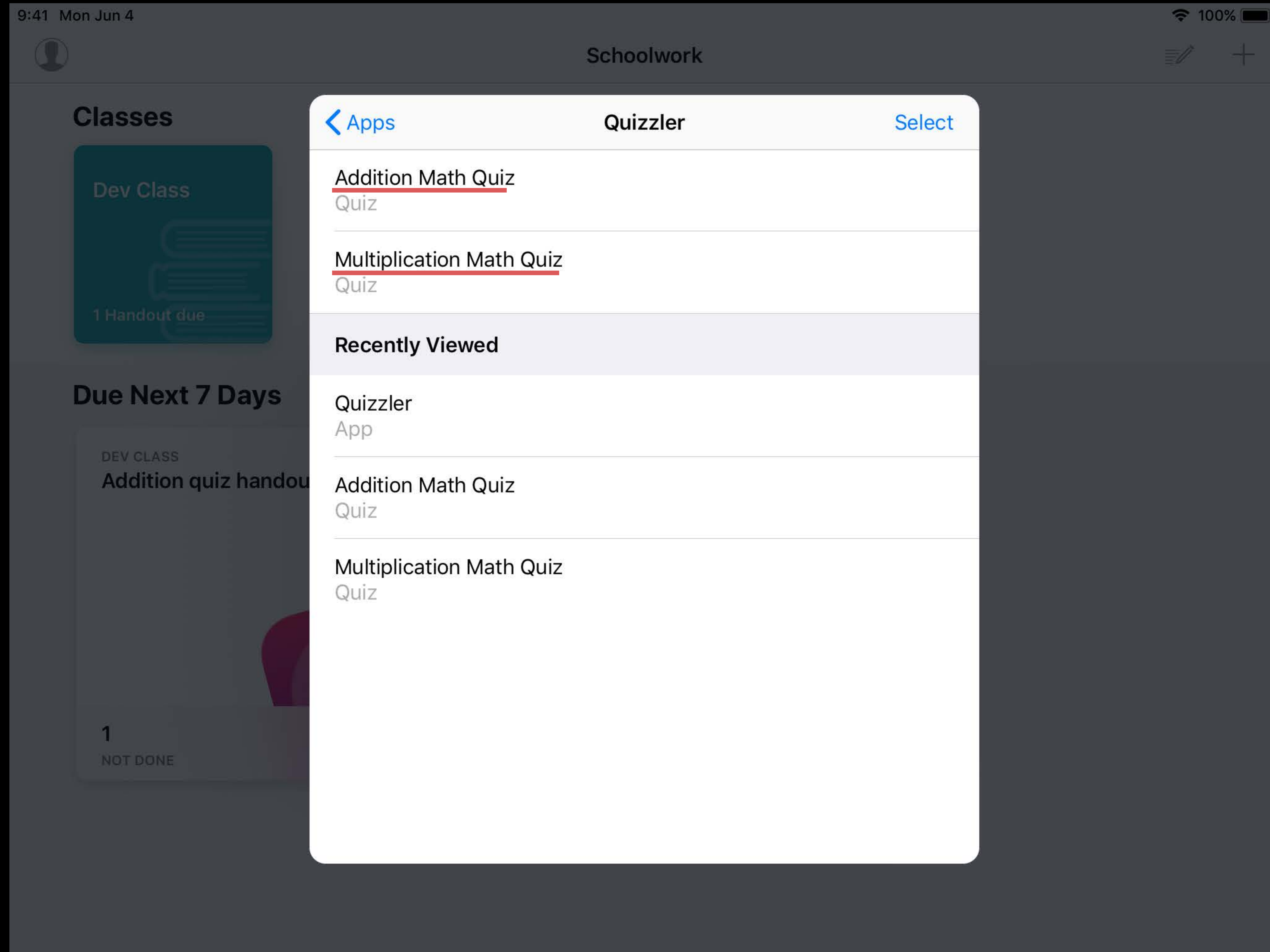
Use clear titles



CLSContext

Use clear titles

Define contexts early

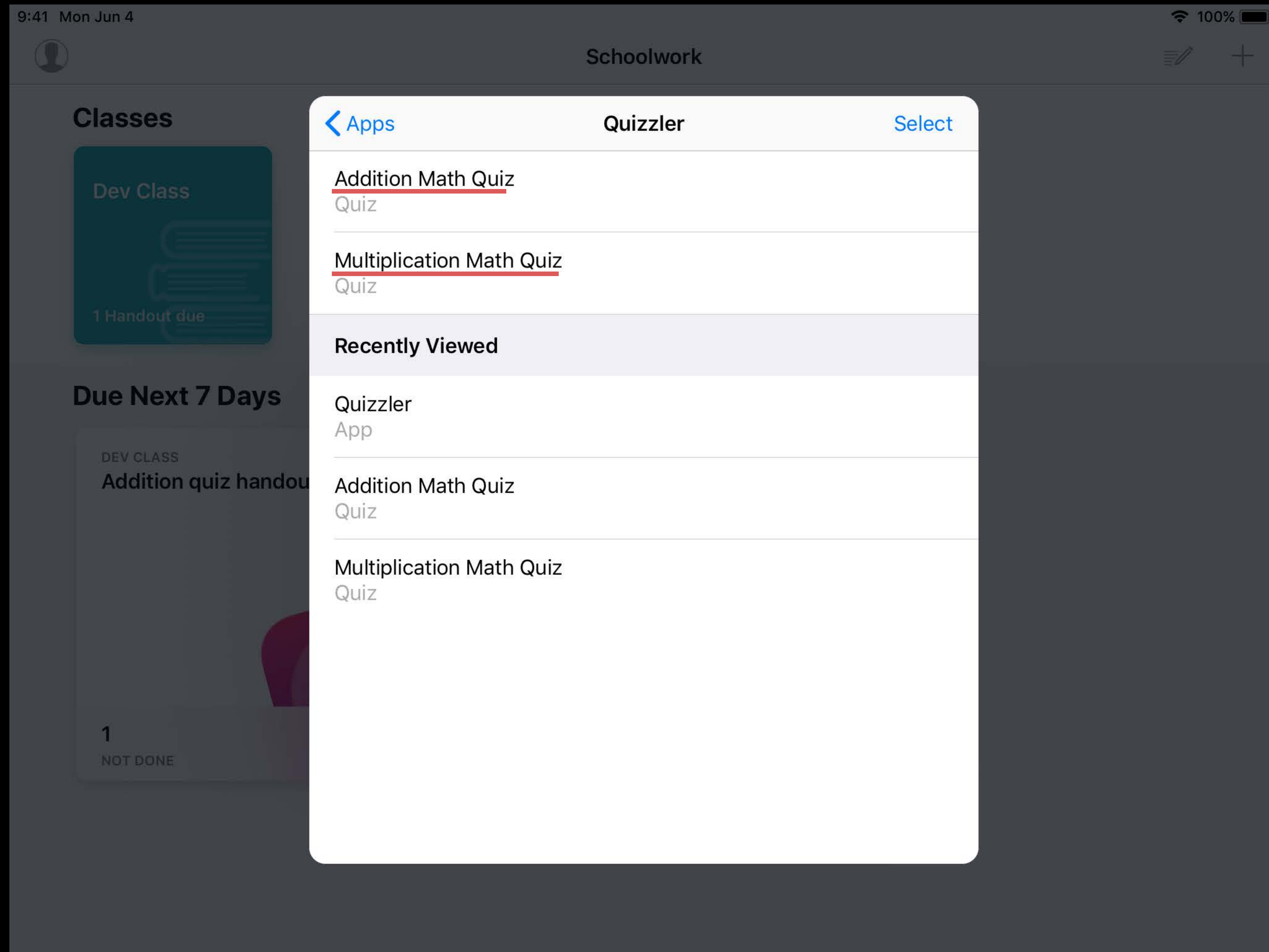


CLSContext

Use clear titles

Define contexts early

Take advantage of `displayOrder`



CLSActivity

9:41 Mon Jun 4 100%

[Schoolwork](#) Handout
Posted Today


DEV CLASS Due
TOMORROW

Addition Quiz Handout


Complete the math quiz on addition.

[CLOSE HANDOUT](#) ⋮

0 NOT DONE **1** DONE

 Addition Math Quiz
Quiz
Quizzler ^

AVG. SCORE	AVG. TOTAL TIME	DONE
100%	2 min	1 1

STUDENTS	SCORE	TOTAL TIME	^ DONE
 Johnny Appleseed	100%	2 min	✓

CLSActivity

9:41 Mon Jun 4 100%

[Schoolwork](#) Handout
Posted Today


DEV CLASS Due
TOMORROW

Addition Quiz Handout


Complete the math quiz on addition.

[CLOSE HANDOUT](#) ⋮

0 NOT DONE **1** DONE

 Addition Math Quiz
Quiz
Quizzler ^

AVG. SCORE	AVG. TOTAL TIME	DONE
100%	2 min	1 1

STUDENTS	SCORE	TOTAL TIME	^ DONE
 Johnny Appleseed	100%	<u>2 min</u>	✓

CLSActivity

```
// create activity  
let activity = quizContext.createNewActivity()  
activity.start()  
//...  
activity.stop()
```

CLSActivity

```
// create activity  
let activity = quizContext.createNewActivity()  
activity.start()  
//...  
activity.stop()
```

CLSActivity

9:41 Mon Jun 4 100%

[Schoolwork](#) Handout
Posted Today


DEV CLASS Due
TOMORROW

Addition Quiz Handout


Complete the math quiz on addition.

[CLOSE HANDOUT](#) ⋮

0 NOT DONE **1** DONE

 Addition Math Quiz
Quiz
Quizzler ^

	AVG. SCORE	AVG. TOTAL TIME	DONE
	100%	2 min	1 1

STUDENTS	SCORE	TOTAL TIME	^ DONE
 Johnny Appleseed	100%	2 min	✓

CLSActivity

9:41 Mon Jun 4 100%

[Schoolwork](#) Handout
Posted Today


DEV CLASS Due
TOMORROW

Addition Quiz Handout


Complete the math quiz on addition.

[CLOSE HANDOUT](#) ⋮

0 NOT DONE **1** DONE

 Addition Math Quiz
Quiz
Quizzler ^

AVG. SCORE	AVG. TOTAL TIME	DONE
100%	2 min	1 1

STUDENTS	SCORE	TOTAL TIME	^ DONE
 Johnny Appleseed	<u>100%</u>	2 min	✓

CLSActivity

```
// create activity
let activity = quizContext.createNewActivity()
// adding primary activity items
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score
```

CLSActivity

```
// create activity
let activity = quizContext.createNewActivity()
// adding primary activity items
let score = CLSScoreItem(identifier: "total", title: "Total Score", score: 5, maxScore: 7)
activity.primaryActivityItem = score
```

CLSActivity

9:41 Mon Jun 4 100%

[Schoolwork](#) Handout
Posted Today


DEV CLASS

Addition Quiz Handout


Complete the math quiz on addition.


[CLOSE HANDOUT](#)


0 NOT DONE **1** DONE

 Addition Math Quiz
Quiz
Quizzler

STUDENTS

 Johnny Appleseed

 Johnny Appleseed

 Addition Math Quiz
Quiz
Quizzler

Score **100%**
TOTAL SCORE Class Avg.: 100%

Total Time **2 min**
Class Avg.: 2 min

Done **Yes**

Passed | Not Passed **Passed**
QUESTION: $X + 2 = 3$

Passed | Not Passed **Passed**
QUESTION: $X + 1 = 3$

Passed | Not Passed **Passed**
QUESTION: $X + 6 = 12$

CLSActivity

9:41 Mon Jun 4 100%

[Schoolwork](#) Handout
Posted Today


DEV CLASS

Addition Quiz Handout


Complete the math quiz on addition.


[CLOSE HANDOUT](#)


0 NOT DONE **1** DONE

 Addition Math Quiz
Quiz
Quizzler

STUDENTS

 Johnny Appleseed

 Johnny Appleseed

 Addition Math Quiz
Quiz
Quizzler

Score **100%**
TOTAL SCORE Class Avg.: 100%

Total Time **2 min**
Class Avg.: 2 min

Done **Yes**

Passed Not Passed QUESTION: $X + 2 = 3$	Passed
Passed Not Passed QUESTION: $X + 1 = 3$	Passed
Passed Not Passed QUESTION: $X + 6 = 12$	Passed

CLSActivity

```
// create activity
let activity = quizContext.createNewActivity()

let binaryItem = CLSBinaryItem(identifier: "question_1", title: "Question 1", type:.passFail)
binaryItem.value = true

activity.addAdditionalActivityItem(binaryItem)
```

CLSActivity

```
// create activity
let activity = quizContext.createNewActivity()

let binaryItem = CLSBinaryItem(identifier: "question_1", title: "Question 1", type:.passFail)
binaryItem.value = true

activity.addAdditionalActivityItem(binaryItem)
```

Demo

Summary

General Overview

Framework Classes

Adopting ClassKit

Testing via Developer Mode and Schoolwork

Summary

ClassKit best practices

Declare contexts early

Not everything needs to be a context

Take advantage of `CLSDataStoreDelegate`

Provide extra information via activity items

Summary

General best practices for education apps

Remove StoreKit dependence

Support purgeable storage

Implement settings access via Managed App Config

More information at: <https://developer.apple.com/education/>

More Information

<https://developer.apple.com/wwdc18/215>

ClassKit and Education Technologies Lab

Technology Lab 4

Wednesday 10:45AM

ClassKit and Education Technologies Lab

Technology Lab 2

Thursday 09:00AM

 **WWDC18**