

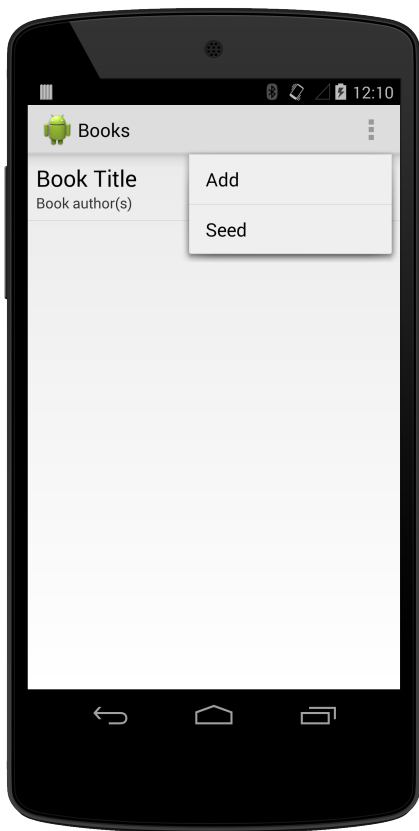
The Prolific Library

The SWAG committee is looking for a way to track who has which book from our library. The goal of this exercise is to create a simple app that connects to a web server and performs a `GET`, a `POST`, and a `PUT`. You may use any 3rd party libraries and tools that you'd like.

Requirements

- You should use git version control
- Use [gradle](#) (we should be able to install your apk from the command line using `./gradlew installDebug`)
- All API requests should be made using [retrofit](#)
- Demonstrate usage of the [Android design principles](#)

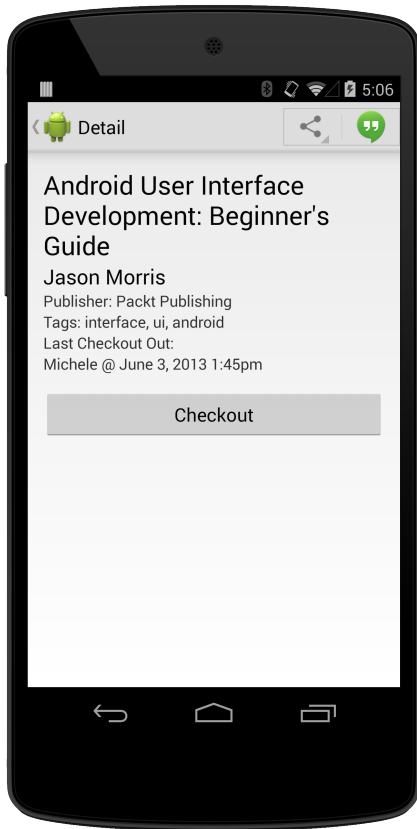
Wireframes



Screen 1:

This screen lists the books in the library. Pressing the `Add` button brings you to the Add Book

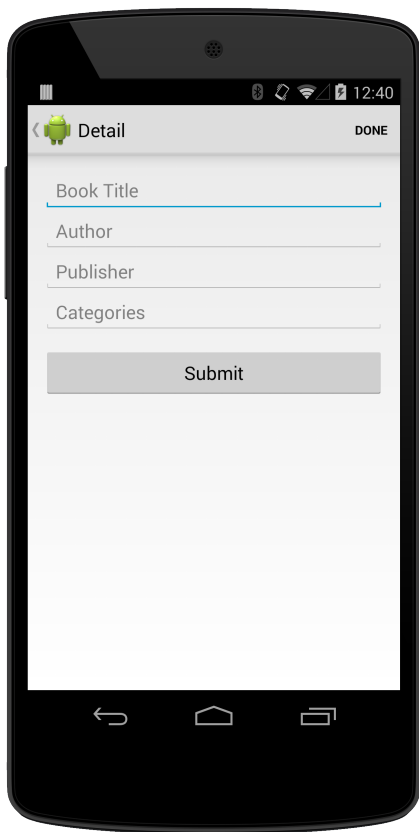
screen in a modal. Pressing an item in the table brings you to the Detail screen. Pressing the **Seed** button should seed the database from the provided **seeds.json**.



Screen 2:

This screen provides detail about a book. Pressing the **Checkout** button asks you for your name, and will update the book entry to list your name and the current date for the **lastCheckedOut** and **lastCheckedOutBy** fields. Pressing the **caret** or back button should back out of the detail screen.

You can also share this book with the share on the top right. This should use the built-in [ShareActionProvider](#).



Screen 3:

This screen is a form to add a new book to the library. These fields are all required. If any are empty and **Submit** is pressed, an alert with an error message is shown. Pressing the **DONE**, **caret**, or **back** button will close this screen. If there is text entered in any of the fields, it will confirm that the user wants to leave the screen with unsaved changes.

Endpoints

Server: **HOST URL WILL BE PROVIDED BY INTERVIEWER**

This server requires dates to be formatted like so: **yyyy-MM-dd HH:mm:ss zzz**

List all Library Books

GET /books

Sample Response:

Status 200 Ok

```
[
  {
    "author": "Jason Morris",
    "categories": "interface, ui, android",
    "lastCheckedOut": null,
    "lastCheckedOutBy": null,
    "publisher": "Packt Publishing",
    "title": "Android User Interface Development: Beginner's Guide",
    "url": "/books/1"
  },
  {
    "author": "Zigurd Mednieks, Laird Dornin, G. Blake Meike, Masumi
Nakamura",
    "categories": "android",
    "lastCheckedOut": "2013-06-06 18:07:00 +0000",
    "lastCheckedOutBy": null,
    "publisher": "O'Reilly Media",
    "title": "Programming Android",
    "url": "/books/2"
  },
  {
    "author": "Reto Meier",
    "categories": "android,professional",
    "lastCheckedOut": null,
    "lastCheckedOutBy": null,
    "publisher": "Wrox",
    "title": "Professional Android 4 Application Development",
    "url": "/books/3"
  }
]
```

Add a Library Book

POST /books

Sample Params:

```
{
  author="Diego Torres Milano"
  categories="android,testing"
  title="Android Application Testing Guide"
  publisher="Packt Publishing"
  lastCheckedOutBy="Joe"
}
```

Sample Response:

```
Status 201 Created
{
  "author": "Diego Torres Milano",
  "categories": "android,testing",
  "lastCheckedOut": null,
  "lastCheckedOutBy": "Joe",
  "publisher": "Packt Publishing",
  "title": "Android Application Testing Guide",
  "url": "/books/5"
}
```

Get a Library Book

GET /books/1

Sample Response:

```
Status 200 Ok
{
  "author": "Jason Morris",
  "categories": "interface, ui, android",
  "lastCheckedOut": null,
  "lastCheckedOutBy": null,
  "publisher": "Packt Publishing",
  "title": "Android User Interface Development: Beginner's Guide",
  "url": "/books/1"
}
```

Update a Library Book

PUT /books/1

Sample Params:

```
{
  lastCheckedOutBy="Michele"
}
```

Sample Response:

```
Status 200 Ok
{
  "author": "Jason Morris",
  "categories": "interface, ui, android",
  "lastCheckedOut": null,
  "lastCheckedOutBy": "Michele",
  "publisher": "Packt Publishing",
  "title": "Android User Interface Development: Beginner's Guide",
  "url": "/books/1"
}
```

Delete a Library Book

DELETE /books/1

Response:

```
Status 204 No Content
```

Clear all Books

DELETE /clean

Response:

```
Status 200 Ok
```