GitHub Username: heymonheymon3000

NBA Schedule

Description

The NBA Schedule App will give users immediate access to the NBA schedule and custom it for their favorite teams.

Intended User

This application is intended to be used by NBA fans who want to know what teams are playing and when.

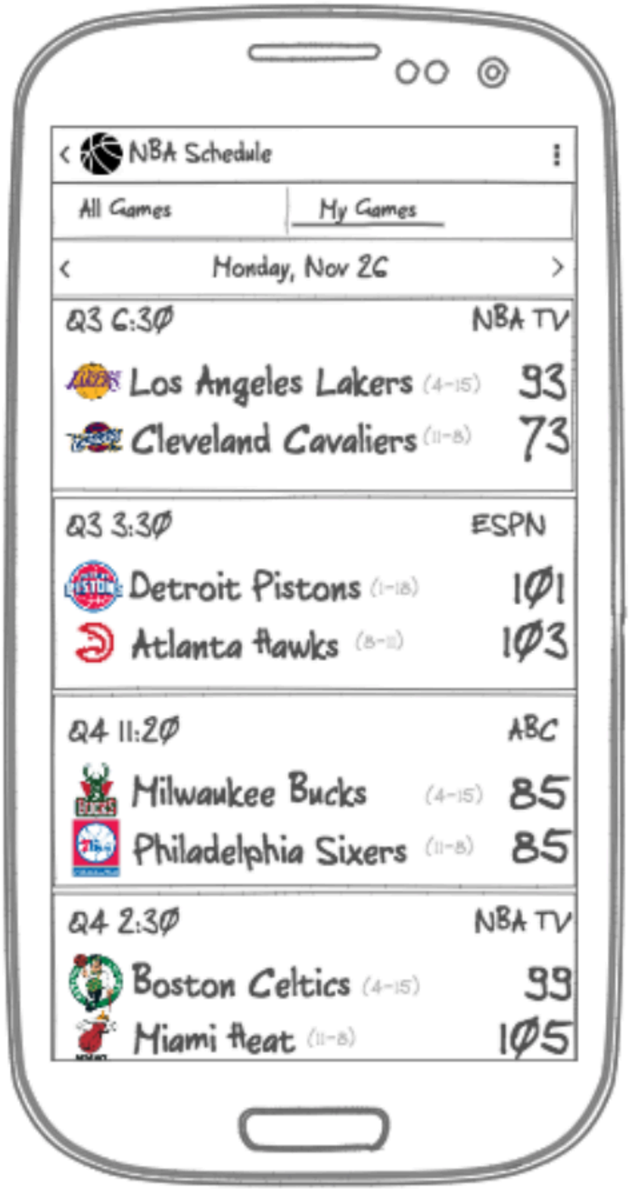
Features

List the main features of your app.

- Displays the NBA schedule based on date along with the score.
- Displays detail box score when you click on the game.
- Allow users to add their favorite team to customize My Games tab

User Interface Mocks

Screen 1



Q3 6:30 NBA TV
Los Angeles Lakers (4-15) 93
Cleveland Cavaliers (11-8) 73

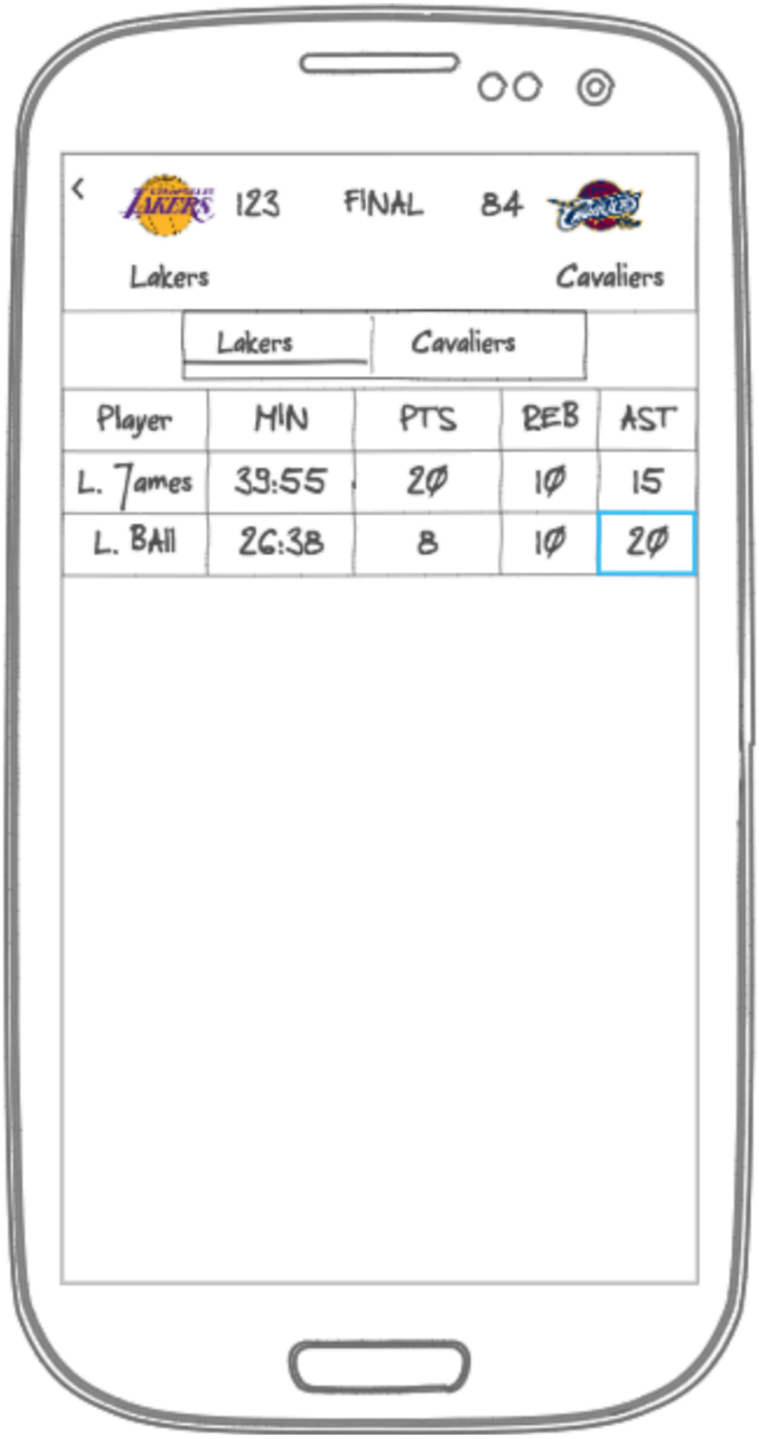
Q3 3:30 ESPN
Detroit Pistons (1-18) 101
Atlanta Hawks (8-11) 103

Q4 11:20 ABC
Milwaukee Bucks (4-15) 85
Philadelphia Sixers (11-8) 85

Q4 2:30 NBA TV
Boston Celtics (4-15) 99
Miami Heat (11-8) 105

At the top of this screen the user can toggle between All Games and My Games. The All Games tab will display all games and the My Games will only display games that the user has chosen as their favorite team. Below the tab is a way to pick the date. When the app first loads, today date will be selected, and the user can click on the left arrow to go back a day or click to the right to go to tomorrow's date. Finally, below the date selector there is a list of games, which displays the teams score, record, time remaining, and channel in which the game will be televised.

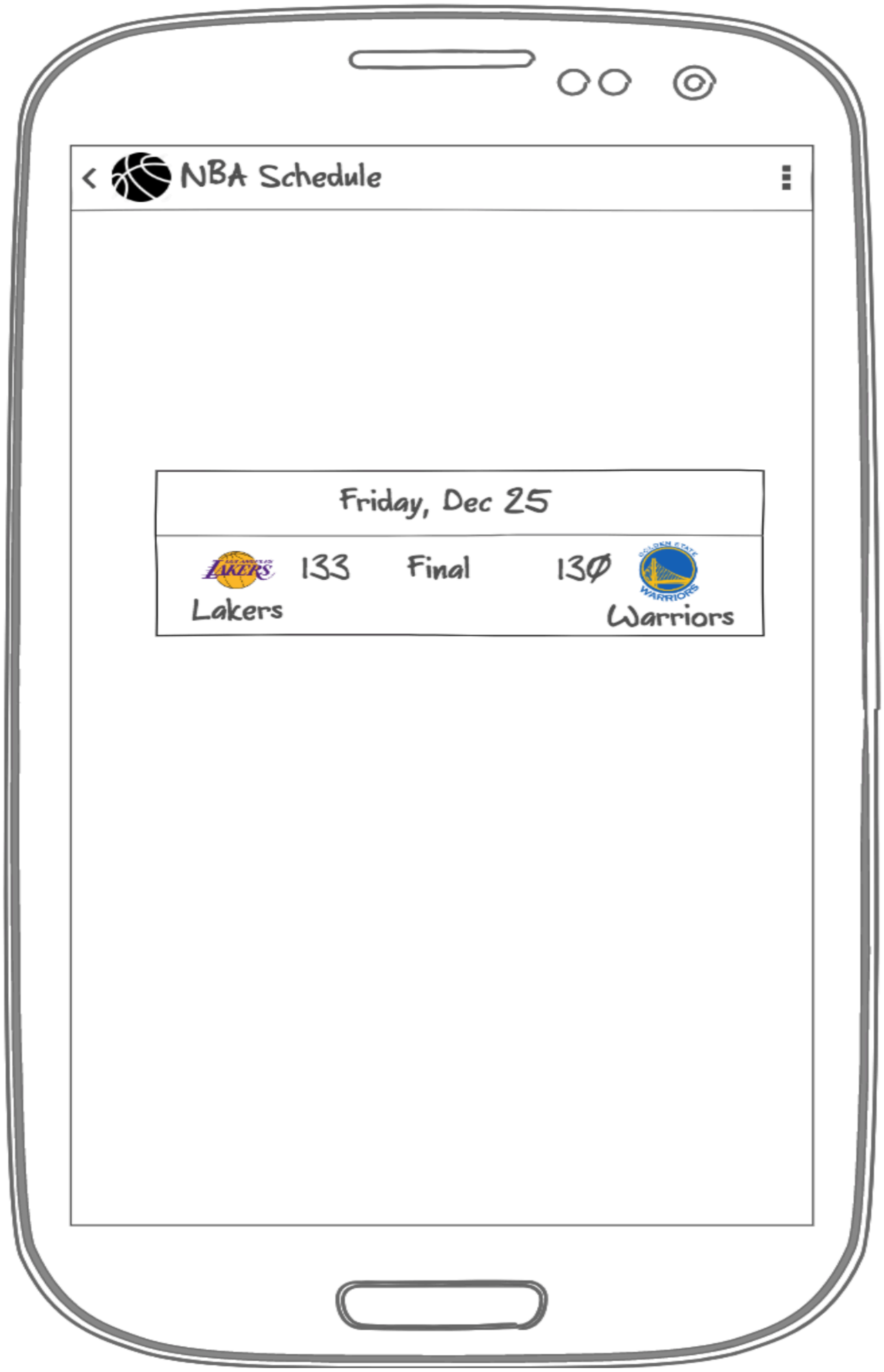
Screen 2



Player	MIN	PTS	REB	AST
L. James	39:55	20	10	15
L. BAll	26:38	8	10	20

At the top is a summary of the teams score and below that is a tab to toggle between each team to get the box score.

Widget



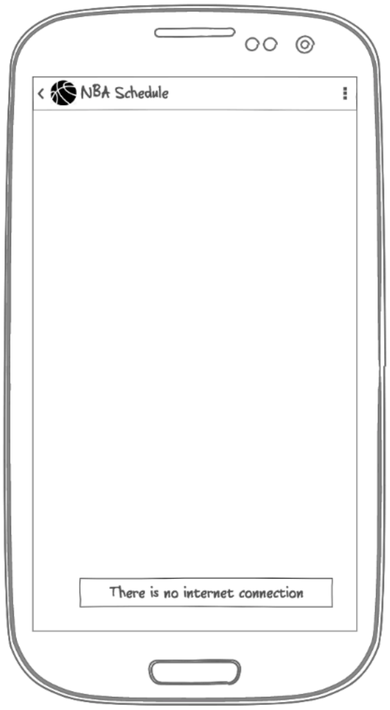
Key Considerations

How will your app handle data persistence?

The schedule will be stored in a native SQLite database and exposed using Room.

Describe any edge or corner cases in the UX.

If there is no internet connection, a toast message will appear at the bottom indicating there is no internet connection.



Describe any libraries you’ll be using and share your reasoning for including them.

Glide to handle the loading and caching of images.

Describe how you will implement Google Play Services or other external services.

Admob – I will use Google-provided test ad units

Analytics – I will use Google Analytics Services

Describe how you will implement pull data from rest api

I will use AsyncTask to fill-full on-demand request.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Working with the https://developer.sportradar.com/ API

- To fetch NBA daily games, make a request to the nba/trial/v5/{language_code}/games/{year}/{month}/{day}/schedule{format} endpoint. This happen when the app first loads up.
- To fetch NBA game box score, make a request to the nba/trial/v5/{language_code}/games/{game_id}/boxscore{format} endpoint. This is called when the user clicks on the game to get more detail.
- Java language will be used for development.
- App keeps all strings in a string.xml file and enables RTL layout switching on all layouts.
- App includes support for accessibility.
- Libraries used:

Library Name	Version
com.github.bumptech.glide:glide	4.7.1
com.android.support:design	27.1.1
com.android.support:support-v4	27.1.1
com.android.support:support-v13	27.1.1
com.android.support:palette-v7	27.1.1
com.android.support:recyclerview-v7	27.1.1
com.android.support:cardview-v7	27.1.1
com.android.support:appcompat-v7	27.1.1
com.squareup.okhttp3:okhttp 3.11.0	3.11.0
com.android.support.constraint:constraint-layout	1.1.3
com.github.bumptech.glide:compiler	4.7.1
junit:junit	4.12
com.android.support.test:runner	1.0.2
com.android.support.test.espresso:espresso-core	3.0.2
Android Studio	3.2.1
Gradle	3.2.1
com.squareup.retrofit2:converter-gson	2.3.0
com.google.code.gson:gson	2.8.1
io.reactivex.rxjava2:rxjava	2.1.0
io.reactivex.rxjava2:rxandroid	2..0.1
io.reactivex.rxjava2:rxjava	2.1.0
io.reactivex.rxjava2:rxandroid	2.0.1
com.squareup.retrofit2:retrofit	2.3.0
android.arch.persistence.room:runtime	1.1.1
android.arch.lifecycle:extensions	1.1.1
android.arch.persistence.room:runtime	1.1.1

android.arch.lifecycle:extensions	1.1.1
android.arch.persistence.room:compiler	1.1.1
android.arch.lifecycle:compiler	1.1.1

Task 2: Implement UI for Each Activity and Fragment

List the subtasks.

- Create layout for MainActivity
- Create fragment to represent the list of games

Task 3: Your Next Task

List the subtasks.

- Create layout for DetailActivity
- Create Fragment for DetailFragment

Task 4: Your Next Task

List the subtasks.

- Create layout for settings for add/remove favorite team

Task 5: Your Next Task

List the subtasks.

- Create layout for widget

Task 6: Your Next Task

List the subtasks.

- Create Content Provider so that the widget can get today’s schedule from the data base.

Task 7: Your Next Task

List the subtasks.

- Add Admob Google Play Services
- Add Analytics Google Play Services