SmartBall Hackathon Example App

Example Android Studio project for the Munich TechFest.

Prequisites

- Android Studio
- Android SDK v24.0.0
- An Android device (at least 4.4) with Bluetooth Low Energy support

Build and run the app

- Import the project into your Android Studio
- Run the app using the "play" or "debug" button
- Make sure the application has the Location permission

Ball Discovery

The ball will advertise over BT LE, specifically advertising the service ***AD04***. See *SelectBallActivity.java* how to scan for balls.

Ball Information

See *BallInfoActivity.java* on how to retrieve additional ball Information such as battery, firmware revision, status, etc ...

Kicking the Ball

To capture kick information the ball needs to be "in a still position" and in the "logging" state. To do so you need to follow steps bellow.

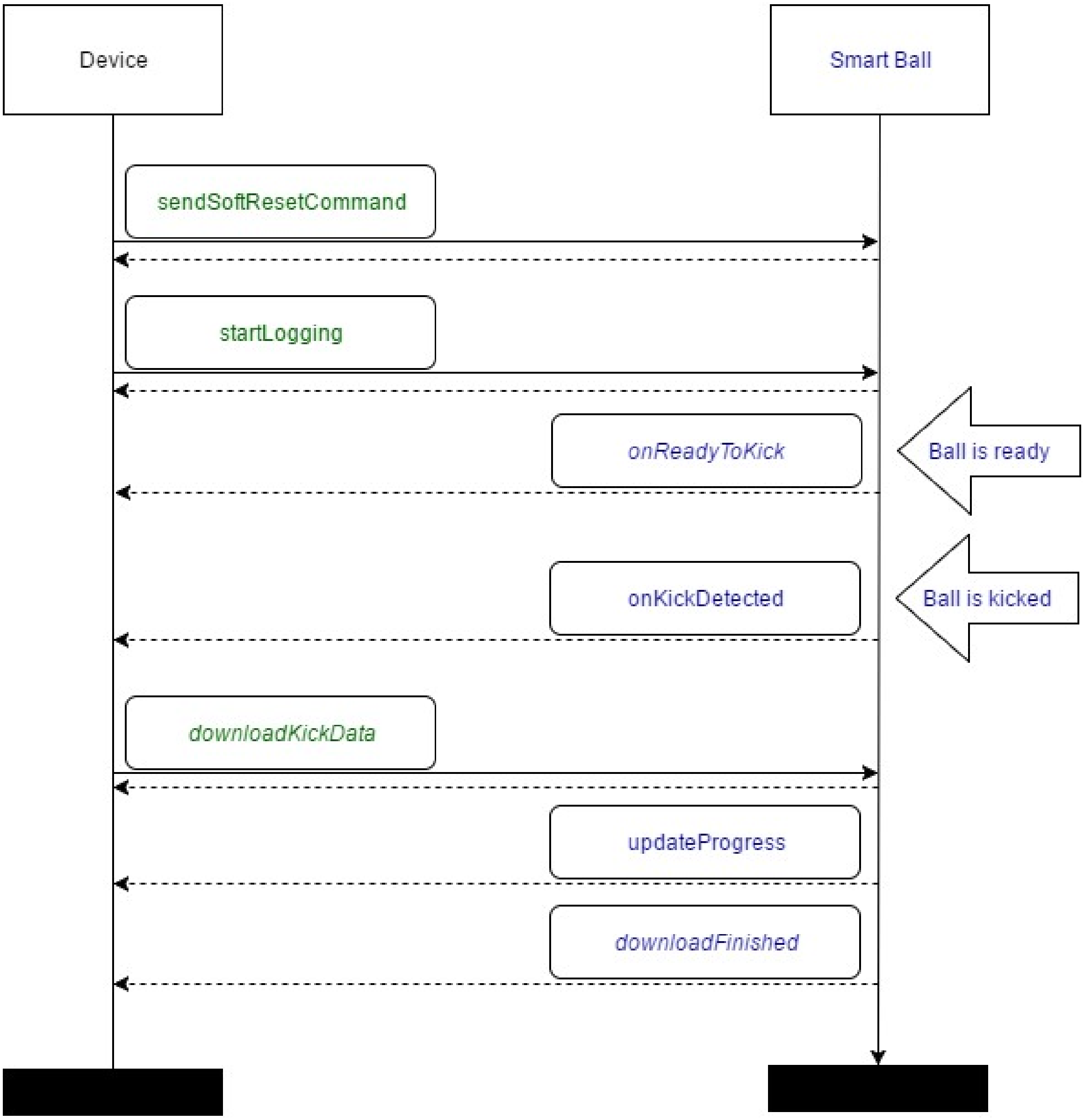
1. Obtain a SmartBallService instance *with Sensor.obtainService(context, SmartBallService.class)*
2. Get the sampling rate; if not what is desired (i.e. 1kHZ) then set the ball sampling rate via *SmartBallService#setSamplingRate*
3. Once the ball is still, call *sendSoftResetCommand* to clear any left-over state (old kick data)
4. Set the KickListener instance with *SmartBallService#setOnKickListener*
5. Put the ball into logging mode with *SmartBallService#startLogging*
6. *KickListener#onReadyToKick* will be invoked when the ball is ready to be kicked
7. ***(kick happens here ...)***
8. *KickListener#onKickDetected* when the kick was detected and logged by the ball
9. Now you ready to download the data.

Downloading the kick data

Once a kick has been detected you can download the data by calling *SmartBallService#downloadKickData* and passing in *DataDownloaderListener*. During the download the listener will be notified of progress via *updateProgress*. *DataDownloaderListener#downloadFinished* once all the data has been downloaded.

To retry a download send a soft reset command and call *downloadKickData* again.

Flow



KickData

The KickData instance will contain all the acceleration data for x,y,z axes. The acceleration is in milli-g, i.e 1000 milli-g is equal to "free-fall". See https://en.wikipedia.org/wiki/G-force#Measuring_gforce_using_an_accelerometer for more information. To get the time between two samples in milliseconds call *KickData#getSamplePeriod*.

The *KickData* class implements *Parcelable* so it's easy to pass to other Activities and also has utility methods to save/load data to a file.