Stream publishing from Android application in background mode

Overview

To prevent Android application unloading from device memory and stream publishing stopping when application goes to background, it is necessary to establish connection to WCS server and to publish a strem from service that should be launched from applications Activity. To prevent service unloading in its turn, a notification must be created to be in notification bar while service is working. Let's explore example of modification of Android Two WayStreaming application source code.

Application source code modification example

1. Service creation on `Publish` button pressing when camera and microphone permissions are granted

```
@Override
public void onRequestPermissionsResult(int requestCode,
                                       @NonNull String permissions[],
@NonNull int[] grantResults) {
  switch (requestCode) {
      case PUBLISH_REQUEST_CODE: {
          if (grantResults.length == 0 ||
                  grantResults[0] != PackageManager.PERMISSION_GRANTED ||
                  grantResults[1] != PackageManager.PERMISSION_GRANTED) {
              Log.i(TAG, "Permission has been denied by user");
          } else {
              mPublishButton.setEnabled(false);
              ...
              Intent intent = new Intent(StreamingMinActivity.this,
TestService.class);
              intent.putExtra("url", mWcsUrlView.getText().toString());
              intent.putExtra("streamName",
mPublishStreamView.getText().toString());
              startService(intent);

              Log.i(TAG, "Permission has been granted by user");
          }
      }
  }
}
```

2. Session creation and stream publishing when service is started

```
@Override
public int onStartCommand(Intent intent, int flags, int startId) {
  SessionOptions sessionOptions = new
SessionOptions(intent.getStringExtra("url"));
  Session session = Flashphoner.createSession(sessionOptions);
  session.connect(new Connection());
  StreamOptions streamOptions = new
StreamOptions(intent.getStringExtra("streamName"));
  Stream publishStream = session.createStream(streamOptions);
  publishStream.publish();
  Toast.makeText(this, "Start service", Toast.LENGTH_SHORT).show();
  return START_STICKY;
}
```

3. Notification creation

```
private void showNotification() {
  Intent notificationIntent = new Intent(this,
StreamingMinActivity.class);
  notificationIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
  PendingIntent pendingIntent = PendingIntent.getActivity(this, 0,
notificationIntent, 0);
  int iconId = R.mipmap.ic_launcher;
  int uniqueCode = new Random().nextInt(Integer.MAX_VALUE);
  Notification notification = new NotificationCompat.Builder(this)
          .setSmallIcon(iconId)
          .setContentText("Started stream")
          .setContentIntent(pendingIntent).build();
  startForeground(uniqueCode, notification);
}
```

4. Service stopping on ëUnpublishë button pressing

```
mPublishButton.setOnClickListener(new OnClickListener() {
   @Override
   public void onClick(View view) {
       if (mPublishButton.getTag() == null ||
Integer.valueOf(R.string.action_publish).equals(mPublishButton.getTag()))
{
           ...
       } else {
           mPublishButton.setEnabled(false);
           ...
           stopService(new Intent(StreamingMinActivity.this,
TestService.class));
           publishStream = null;
       }
       ...
   }
});
```

5. Stream publishing stopping while service stops

```java
@Override
public void onDestroy() {
  super.onDestroy();
  publishStream.stop();
  Toast.makeText(this, "Stop service",
          Toast.LENGTH_SHORT).show();
  stopForeground(true);
}
```

Full source code of modified file `StreamingMinActivity.java`

StreamingMinActivity.java

Full source code of service implementation file `TestService.java`

TestService.java

Full source code of modified application manifest

AndroidManifest.xml

Known issues

1. It is not possible to display local video while stream is published from service.
2. In this implemetation, if the application is closed from running applications list, service will also be stopped.