

Torcea Octavian  
324 CA

## Proiect AD

### Generator de numere aleatorii folosind accelerometrul unui telefon mobil

Facultatea de Automatica si Calculatoare  
Universitatea Politehnica Bucuresti

## 1 Descrierea aplicatiei

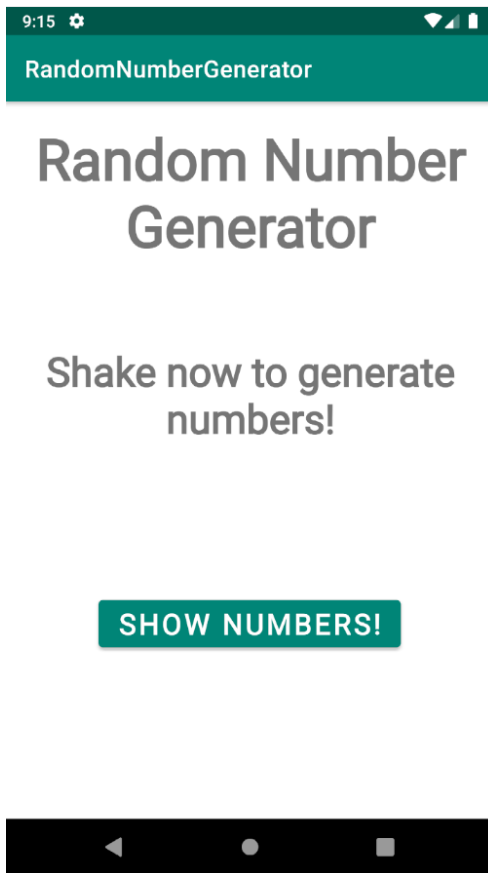
Aplicatie Android ce preia informatii de pe cele 3 axe ale accelerometrului, le prelucreaza folosind un algoritm (descrie mai jos) si le afiseaza utilizatorului. De asemenea, utilizatorul are posibilitatea de a preciza cate numere doreste sa fie afisate (maxim 10), si intervalul in care acele numere vor fi generate (minim 0, maxim 100).

### Ecranul principal

The screenshot shows the main screen of an Android application titled "RandomNumberGenerator". The interface includes three input fields for user configuration and a large green button to generate numbers. Annotations with arrows point to each of these elements, explaining their function.

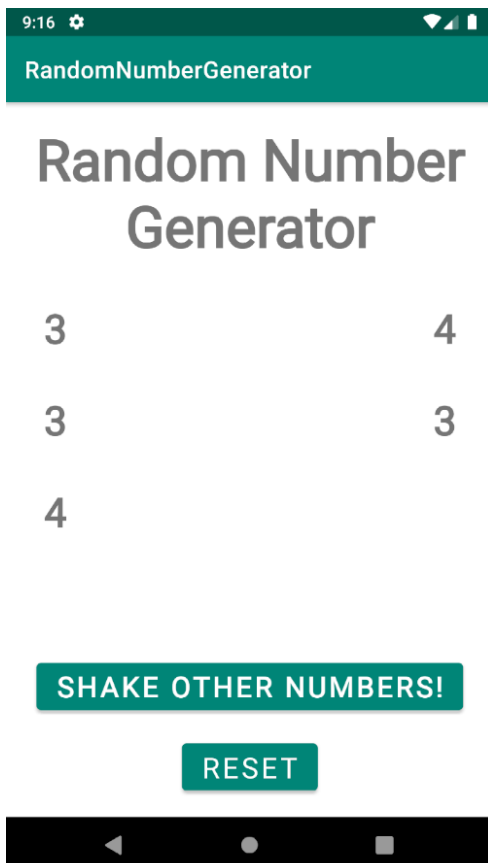
UI Element	Description
how many numbers? (max. 10)	Selectare numar de numere generate
min value (min. 0)	Valoarea minima
max value (max. 100)	Valoarea maxima
GENERATE NUMBERS	Apasand acest buton, utilizatorul confirma datele inserate si aplicatia va avansa la urmatorul "ecran", incepand sa inregistreze datele primite de la accelerometru.

## Ecranul de generare



- Utilizatorul trebuie sa “agite” telefonul pentru a genera numere.
- Cand acesta considera ca a agitat destul telefonul, trebuie sa apese pe butonul “SHOW NUMBERS!” pentru ca aplicatia sa afiseze numerele.
- In momentul in care butonul este apasat, aplicatia va inceta din a primi informatii de la accelerometru.

## Afisarea numerelor



- Numerele generate vor fi afisate.
- Daca se doreste generarea altor numere cu aceleasi constrangeri specificate in ecranul principal (numarul de numere, val. minima, maxima), utilizatorul poate apasa pe butonul “SHAKE OTHER NUMBERS!”.
- Pentru a reveni la ecranul principal, utilizatorul trebuie sa apese pe butonul “RESET”.

OBS. : Aceasta imagine este o captura de ecran de pe un emulator de Android, iar numerele generate nu sunt relevante, deoarece nu exista posibilitatea de a testa functiile accelerometrului folosind un emulator. Videoclipul din arhiva trimisa exemplifica mai bine functia de generare a numerelor aleatoare.

## 2 Implementarea

- Aplicatia este realizata in limbajul Kotlin, folosind Android Studio.
- Minimum SDK: API 19: Android 4.4 (KitKat)
- Testata pe:
  - HTC OnePlus 3T A3003, Versiune Android: 9
  - Huawei P30 Pro, Versiune Android: 10

Pentru a putea prelua datele de la accelerometru, am instantiat un `SensorManager`. Acesta este de tipul „`TYPE_ACCELEROMETER`” si preia datele la un interval de 20 ms („`SENSOR_DELAY_GAME`”).

```
151     private lateinit var sensorManager: SensorManager
152
153     private fun setUpSensorStuff() {
154         sensorManager = getSystemService(SENSOR_SERVICE) as SensorManager
155
156         sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)?.also { it: Sensor
157             sensorManager.registerListener(
158                 listener: this,
159                 it,
160                 SensorManager.SENSOR_DELAY_GAME)
161         }
162     }
```

Cand senzorul detecteaza o schimbare (practic la fiecare 20 ms), datele preluate de pe cele 3 axe sunt adaugate intr-un `ArrayList` de `Floaturi` pentru a fi stocate. Pentru a nu folosi memorie inutil, am decis sa patrez doar cele mai noi 100 de intrari, considerand ca sunt suficiente date pentru a genera toate numerele aleatorii.

```
195     override fun onSensorChanged(event: SensorEvent?) {
196         if (event?.sensor?.type == Sensor.TYPE_ACCELEROMETER) {
197             if (acqData.size > 100) {
198                 acqData.removeAt(index: 0)
199                 acqData.removeAt(index: 0)
200                 acqData.removeAt(index: 0)
201             }
202
203             acqData.add(abs(event.values[0]))
204             acqData.add(abs(event.values[1]))
205             acqData.add(abs(event.values[2]))
206         }
207     }
```

### Algoritmul de prelucrare si generare a numerelor aleatorii:

- 1) Pentru fiecare numar  $i$  de la 0 la 9 de aduna datele receptionate de la pozitiile  $10 * j + i$  (ex.: pt. nr. 0 se folosesc datele de pe pozitiile 0, 10, 20, ..., 90; pt. nr. 1 se folosesc datele de pe pozitiile 1, 11, 21, ..., 91, etc.);
- 2) Fiecare numar astfel obtinut se inmulteste cu 31;
- 3) Se obtine modulul fiecarui numar la impartirea cu diferenta dintre valoarea maxima si valoarea minima la care se adauga 1;
- 4) Se adauga la numarul obtinut valoarea minima specificata de utilizator.

Pasii 3) si 4) se realizeaza pentru a putea satisface constrangerile specificate de utilizator.

```
170 private fun parseNumbers() {
171     while (acqData.size < 100) {
172         acqData.add(0f)
173     }
174
175     for (i in 0..9) {
176         for (j in 0..9) {
177             generatedNr[j] += acqData[10 * i + j]
178         }
179     }
180
181     for (i in 0..9) {
182         generatedNr[i] = generatedNr[i] * 31
183     }
184
185     for (i in 0..9) {
186         generatedNr[i] = generatedNr[i].toInt().toFloat() % (maxVal - minVal + 1)
187         generatedNr[i] = generatedNr[i] + minVal
188     }
189
190     for (i in 0..9) {
191         nrStrings[i].text = generatedNr[i].toInt().toString()
192     }
193 }
```

### Afisarea numerelor:

```
99 private fun showNumbers() {
100     parseNumbers()
101     unregisterSensor()
102
103     binding.apply { this: ActivityMainBinding
104         shakeNowText.visibility = View.GONE
105         showNumbersButton.visibility = View.GONE
106         resetButton.visibility = View.VISIBLE
107         shakeAgainButton.visibility = View.VISIBLE
108     }
109
110     for (i in 0 until nrOfNumbers) {
111         nrStrings[i].visibility = View.VISIBLE
112     }
113 }
```

### 3 Continutul arhivei

- Arhiva cu codul sursa
- Aplicatia compilata (.apk)
- Videoclipul cu functionarea aplicatiei in real-time
- Acest fisier PDF

### 4 Referinte

- [https://www.youtube.com/watch?v=xcsuDDQHrLo&ab\\_channel=CodePalace](https://www.youtube.com/watch?v=xcsuDDQHrLo&ab_channel=CodePalace)
- [https://developer.android.com/guide/topics/sensors/sensors\\_overview.html](https://developer.android.com/guide/topics/sensors/sensors_overview.html)