



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB II

### LANDASAN TEORI

#### 2.1. Computer Based Test (CBT)

Keberhasilan siswa dalam memahami materi pelajaran dapat diketahui berdasarkan hasil belajar siswa. Majunya perkembangan teknologi informasi dan komunikasi, penilaian *Computer Based Test (CBT)* mulai diterapkan untuk memberikan penilaian hasil belajar siswa. Hasil dan proses penilaian yang diterapkan pada sistem *CBT* dianggap memiliki beberapa kelebihan dibandingkan dengan sistem penilaian menggunakan *Paper Based Test (PBT)*. Kelebihan yang dimaksud antara lain: lebih efektif dan efisien dalam penggunaan waktu, sumber daya manusia, dan biaya untuk melaksanakan ujian (Simin, 2013).

Pelaksanaan ujian secara *Paper Based Test (PBT)* diperlukan perhatian khusus untuk mengatasi terjadinya tindak kecurangan yang kemungkinan besar akan dilakukan oleh siswa. Salah satu penyebab terjadinya kecurangan dikarenakan adanya kesamaan soal yang dikerjakan oleh masing-masing siswa (McCabe, 1994). Maka dibutuhkan sebuah sistem untuk mengurangi kemungkinan terjadinya kecurangan antar siswa dengan menerapkan sistem pengacakan soal pada aplikasi *CBT* (Hangga, 2016).

#### 2.2. Android

Android adalah aplikasi sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan *platform* terbuka untuk para pengembang dalam menciptakan aplikasi yang digunakan oleh banyak perangkat keras bergerak

(Nazruddin, 2012). Android SDK (*Software Development Kit*) menyediakan *Tools* dan API yang diperlukan untuk pengembangan aplikasi menggunakan *platform* android dengan memanfaatkan bahasa pemrograman Java. Berawal di tahun 2005, Google mulai mengembangkan *platform* android secara intensif hingga pada 12 November 2007, bersama dengan OHA (*Open Handset Alliance*) yaitu konsorsium perangkat *mobile* terbuka, merilis Google Android SDK (Nadif, 2013).

### 2.3. *Pseudo Random Number Generator*

*Pseudo Random Number Generator* adalah algoritma yang menggunakan beberapa jenis model perhitungan matematika atau tabel yang telah dihitung sebelumnya untuk menghasilkan sebuah urutan dari angka yang tampil acak. *Sequence number* yang dihasilkan tidak sepenuhnya acak namun, ditentukan berdasarkan arbitrase *state* awal yang disebut *seed state* (Jagannatam, 2008). Secara umum, *Pseudo Random Number Generator* menggunakan fungsi  $f$  sebagai fungsi pembangkit (L'Ecuyer, 1994),

$$x_i = f(x_{i-1}, x_{i-2}, \dots, x_{i-n}) \quad \dots(2.1)$$

yang setiap nilai baru sebuah  $x_i$  adalah fungsi dari nilai  $n$  sebelumnya (Fog, 2015). Mekanisme *Pseudo Random Number Generator* harus mencapai beberapa ketentuan agar perhitungannya tepat, seperti (keseragaman) *uniformity*, (ketunggalan) *independence*, (dapat diimplementasikan pada jangka panjang) *long period*, (inisialisasi yang tepat) *proper initialization*, (hasil yang tidak dapat diprediksi) *unpredictability*, (efisiensi) *efficiency*, dan (dapat berjalan di berbagai *platform*) *portability* (Jagannatam, 2008).

#### 2.4. *Linear Congruential Generator*

Tidak selalu nilai  $x_{i-1}, x_{i-2}, \dots, x_{i-n}$  dibutuhkan fungsi  $f$  jika memiliki sejumlah jarak  $\varphi$  dan  $f$  bergantung pada  $x_{i-\varphi}$ . Fungsi  $f$  dapat diimplementasikan pada sebuah *vector processor* dengan meregister ukuran dari  $v$ .

$$v \leq w\varphi \quad \dots(2.2)$$

Jika pada rumus 2.2 untuk semua jalur  $\varphi$ , maka  $w$  adalah jumlah dari bit yang dibutuhkan untuk merepresentasikan setiap  $x_i$  yang dihasilkan. Maka didapat hasil  $x_i$  yang berbeda (Fog, 2015):

$$(x_{i+7}, x_{i+6}, \dots, x_i) = f(x_{i-1}, x_{i-2}, \dots, x_{i-n}) \quad \dots(2.3)$$

Algoritma *Linear Congruential Generator* adalah salah satu pembangkit bilangan acak yang sangat terkenal dan sering diimplementasikan untuk membangkitkan bilangan acak di berbagai bahasa pemrograman (Wahyudi, 2015).

$$x_{i+1} = (ax_i + b) \bmod m \quad \dots(2.4)$$

$x_{i+1}$  adalah bilangan acak ke- $i$  dari deretnya,  $a$  adalah variabel pengali,  $x_i$  adalah bilangan acak sebelumnya,  $b$  adalah variabel penambah dan  $m$  adalah variabel modulus (Knuth, 1981). Peraturan konstanta ditetapkan dengan syarat,  $a$  dapat dibagi semua dengan faktor prima dari  $m$ ,  $b$  relatif prima terhadap  $m$ ,  $m >$  maksimum, dan  $a > 0, b > 0$ . *Linear Congruential Generator* tidak dapat digunakan untuk kriptografi namun, tetap berguna pada aplikasi non-kriptografi seperti simulasi, karena mampu memperlihatkan statistik yang baik dan sangat tepat (Wahyudi, 2015).

#### 2.5. **Kuesioner**

Angket atau *questionnaire* adalah daftar pertanyaan yang disebarakan untuk diisi dan dikembalikan kepada peneliti atau dapat juga dijawab langsung oleh

responden dibawah pengawasan peneliti dan dapat bersifat kuesioner tertutup, kuesioner terbuka, dan kombinasi kuesioner tertutup dan kuesioner terbuka berdasarkan sifat jawabannya (Nasution, 2006). Bentuk pengukuran lainnya, penjelasan kualitatif antara hubungan peneliti dengan pengguna harus mencakup sampai sejauh mana peneliti memiliki orientasi pada kepuasan pelanggan. (Sabowo, 2013).

Skala Likert diterapkan salah satu alat psikometrik yang mendasar dan sering digunakan untuk penelitian ilmu sosial dan pendidikan yang secara bersamaan juga sering muncul perdebatan dan kontroversi terkait analisis dan penyertaan poin pada skala (Joshi, 2015). Skala Likert digunakan dalam mengukur sikap atau pendapat seseorang atau sejumlah kelompok terhadap sebuah fenomena sosial yang jawabannya memiliki gradasi dari sangat positif hingga sangat negatif (Ong, 2014). Skala yang digunakan dalam penelitian ini:

1. Kurang Sekali
2. Kurang
3. Baik
4. Sangat Baik

Analisa presentase dan rumus perhitungan nilai untuk setiap pertanyaan pada kuesioner yaitu (Sugiyono, 2012):

$$P = \frac{F}{N} \times 100\% \quad \dots(2.5)$$

Variabel P adalah presentase akhir, F adalah frekuensi nilai yang didapat pada kuesioner yang didapat dari penjumlahan jumlah skor kategori jawaban, dan N adalah jumlah responden atau sampel (Zahra, 2015).