



# 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.

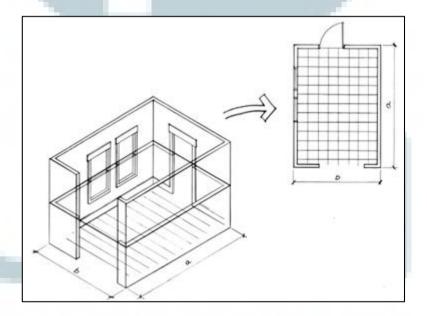
### **BAB II**

### LANDASAN TEORI

# 2.1 Layout Rumah

Dalam merancang tata ruang (*layout*) rumah yang baik diperlukan pedoman dan aturan rancangan, sehingga dapat memenuhi ukuran kualitas tertentu dan sesuai dengan kepribadian penghuni rumah (Bao dkk., 2013). Sebuah rumah setidaknya memiliki satu kamar, satu dapur, dan satu kamar mandi. Agar lebih mudah dilihat dan dimengerti, rancangan *layout* rumah dibuat dalam bentuk *floor plan*.

Floor plan merupakan tampilan ruangan atau bangunan yang strukturnya dipotong secara horizontal, setengah dari bagian atas bangunan dihilangkan, dan dilihat dari atas, sehingga *layout* bangunan dan informasi penting seperti lokasi dinding, pintu, dan jendela terlihat dalam skala tertentu (Mitton, 2004). Gambar 2.1 merupakan contoh *floor plan* dari sebuah ruangan.



Gambar 2.1 *Floor plan* dari gambar ruangan yang dipotong secara horizontal (Mitton, 2004)

### 2.2 Perencanaan Tata Letak (Layout) Ruang Dalam Rumah

Untuk memperoleh desain rumah dan tatanan ruang yang baik dan sesuai keinginan, maka diperlukan perencanaan yang matang. Perencanaan yang dilakukan yaitu bagaimana cara menata rumah dengan baik agar setiap ruang dapat berfungsi secara optimal. Menurut Sastra (2012), selain menggunakan jasa arsitek, perencanaan dalam menata ruangan juga bisa dilakukan sendiri dengan cara mengenali seberapa banyak kebutuhan ruang untuk keluarga dan memahami langkah-langkah perencanaan yang diperlukan sebelum membangun rumah. Langkah-langkah perencanaan tersebut menurut Sastra (2012) yaitu sebagai berikut.

### 1. Menentukan kebutuhan ruang

Untuk memperoleh desain rumah tinggal yang mempunyai jumlah serta jenis ruang yang sesuai dengan keinginan dan kebutuhan, sebelum membangun rumah sebaiknya ditentukan terlebih dahulu kebutuhan-kebutuhan ruang yang akan dimasukkan dalam rumah tinggal tersebut. Hal ini dilakukan karena pada dasarnya, perencanaan yang baik memegang peranan penting dalam sebuah proses untuk mendesain rumah. Beberapa tahapan yang dapat dilakukan untuk menentukan kebutuhan ruang yaitu sebagai berikut.

- a. Buat daftar kebutuhan ruang dengan memperhitungkan semua kegiatan yang dilakukan penghuni rumah.
- b. Buat perencanaan konsep ruang yang akan diterapkan, misalnya menentukan luasan masing-masing ruang, bentuk dan orientasi ruang, dan pencahayaan.
- c. Menentukan jumlah kamar tidur dan jenis ruang lainnya, seperti ruang tamu, teras, ruang keluarga, dan ruang makan.

- d. Menentukan jenis ruang apa yang diutamakan atau yang harus ada berdasarkan skala prioritas.
- e. Jika luas lahan atau luas tanah terbatas, ruang dengan fungsi yang hampir sama dapat digabungkan, atau jika ruang jarang digunakan sebaiknya dihilangkan saja. Sebagai contoh, ruang keluarga dan ruang makan digabung menjadi satu sehingga dapat melakukan efisiensi lahan.
- f. Buat pengelompokan atau zonasi ruang berdasarkan kebutuhan, kegiatan, dan aktivitas penghuni.

### 2. Membuat susunan (*layout*) denah

Untuk menciptakan suasana tinggal yang tenang dan nyaman dalam suatu rumah tinggal, upaya penciptaan sebuah tata letak atau *layout* perlu diperhatikan. Menurut Sastra (2012), beberapa hal yang perlu diperhatikan dalam merancang tata ruang atau *layout* rumah agar efisien dan nyaman adalah sebagai berikut.

- a. Letakkan ruang-ruang dengan fungsi dan sifat yang mirip dalam satu zonasi.

  Teras, ruang tamu, dan garasi diletakkan pada daerah publik (daerah yang mudah dijangkau). Ruang makan, dapur, dan gudang diletakkan pada daerah semi publik. Kamar, kamar mandi, dan ruang lainnya yang membutuhkan ketenangan diletakkan pada daerah privat (daerah tenang).
- b. Prioritaskan ruangan yang membutuhkan cahaya matahari, sirkulasi udara, dan akses keluar-masuk agar diletakkan di depan, belakang, atau samping rumah. Contohnya, garasi diletakkan langsung di bagian depan agar mudah diakses dari jalan, ruang tamu diletakkan di dekat garasi agar mudah diakses, kamar mandi dan gudang yang membutuhkan sirkulasi udara tidak harus di bagian depan.

- c. Tempatkan ruang atau *space* pengikat di antara ruangan utama, misalnya letak ruang makan di antara kamar.
- d. Ciptakan kesan ruang saling terbuka antara ruang makan dan ruang tamu dengan tidak membuat sekat permanen diantaranya, misalnya dibatasi dengan furnitur atau meja.
- e. Batasi jumlah pintu pada ruang agar tidak mengganggu sirkulasi antar ruangan, dan pintu jangan diletakkan menyilang.
- f. Untuk menciptakan suasana makan dan istirahat yang nyaman, tempatkan teras di bagian samping atau belakang rumah tepat di depan ruang makan dan ruang tamu.
- g. Ruangan pelayanan seperti dapur dan ruang *laundry* sebaiknya diletakkan pada daerah yang tidak berhubungan langsung dengan ruang tamu agar tidak menimbulkan gangguan.

Dalam penelitian ini, kebutuhan ruang dan syarat peletakan ruang berdasarkan teori Sastra (2012) mengenai tata letak ruang dan beberapa syarat (*constraint*) tata letak ruang yang digunakan Liliana (2006), adalah sebagai berikut.

#### 1. Kamar

- a. Jenis ruangan ini bersifat wajib atau harus ada dalam sebuah rumah.
- b. Jumlah kamar dalam sebuah rumah adalah satu sampai lima ruangan.
- c. Kamar utama sebaiknya diletakkan di dekat ruang tamu, untuk memudahkan akses keluar masuk penghuni utama.
- d. Kamar lainnya dapat diletakkan di sisi rumah atau di dekat kamar utama, namun sebaiknya tidak di dekat dapur atau ruang *laundry*, karena dapat mengganggu aktivitas istirahat penghuni.

### 2. Ruang tamu

- a. Jenis ruangan ini bersifat wajib atau harus ada dalam sebuah rumah sebanyak satu ruangan.
- b. Ruang tamu diletakkan di tengah rumah atau di depan rumah, sehingga dapat menghubungkan tiap ruangan yang ada pada rumah.
- c. Ruang tamu diletakkan di dekat teras depan, halaman depan, atau garasi untuk memudahkan akses keluar masuk penghuni rumah. Selain itu, ruang tamu dapat diletakkan di dekat ruang makan, karena ruang makan dan ruang tamu merupakan ruangan yang bersifat publik.
- d. Ruang tamu juga dapat diletakkan di dekat kamar utama untuk memudahkan akses ke kamar penghuni utama.

### 3. Kamar mandi

- a. Jenis ruangan ini bersifat wajib atau harus ada dalam sebuah rumah.
- b. Jumlah kamar mandi dalam suatu rumah adalah satu sampai dua ruangan.
- c. Kamar mandi sebaiknya diletakkan di bagian belakang rumah atau tengah rumah, namun di sebelah sisi rumah. Hal ini bertujuan agar kamar mandi mudah diakses oleh penghuni rumah dan tidak mengganggu tata letak ruang lain.
- d. Sebaiknya tidak diletakkan berseberangan atau di dekat dapur atau ruang makan, karena diasumsikan dapat menimbulkan masalah kehigienisan.

# 4. Ruang makan dan dapur

 a. Jenis ruangan ini bersifat wajib atau harus ada dalam sebuah rumah, masingmasing sebanyak satu ruangan.

- b. Dapur dan ruang makan dapat diletakkan bersebelahan untuk mempermudah aktivitas makan penghuni rumah.
- c. Sebaiknya dapur tidak diletakkan di dekat kamar, karena dapat mengganggu aktivitas istirahat penghuni, terutama jika sirkulasi udara dalam rumah buruk.

### 5. Halaman dan teras

- a. Jenis ruangan ini bersifat opsional, karena tidak semua rumah membutuhkan halaman atau teras, terutama untuk ukuran lahan yang kecil. Halaman dan teras dalam sebuah rumah berjumlah satu sampai dua, yaitu halaman atau teras depan, dan halaman atau teras belakang.
- b. Letak halaman atau teras selalu di paling depan atau paling belakang lahan yang tersedia.
- c. Letak vertikal dari halaman depan harus lebih kecil atau sama dengan letak vertikal teras depan, dan letak vertikal halaman belakang harus lebih besar atau sama dengan letak vertikal teras belakang.
- d. Ruangan yang dapat sejajar atau bersebelahan dengan halaman yaitu teras dan garasi. Hal ini bertujuan agar posisi garasi memudahkan kendaraan untuk keluar masuk rumah dan tidak ada ruangan lain yang menghalangi keluar masuknya kendaraan maupun penghuni rumah.

### 6. Ruang *laundry* dan gudang

a. Jenis ruangan ini bersifat opsional, karena ruang *laundry* dan gudang merupakan ruangan untuk kebutuhan khusus, sehingga tidak harus ada dalam sebuah rumah. Jumlah masing-masing ruangan ini adalah satu ruangan dalam sebuah rumah.

- b. Ruang *laundry* dan gudang tidak disarankan berada di tengah rumah atau dekat ruang tamu, sehingga dirancang agar diletakkan di pojok.
- c. Ruang *laundry* sebaiknya diletakkan di dekat kamar mandi, karena kedua ruangan tersebut memiliki sifat dan fungsi yang hampir serupa.

### 7. Garasi

- a. Jenis ruangan ini bersifat opsional, karena tidak semua rumah membutuhkan garasi untuk meletakkan kendaraan. Jumlah garasi dalam sebuah rumah adalah satu ruangan.
- b. Garasi harus diletakkan sejajar atau di luar halaman atau teras (jika ada) sehingga memudahkan keluar masuknya kendaraan.
- c. Disarankan garasi diletakkan dekat dengan ruang tamu, sehingga akses masuk ke dalam rumah lebih mudah.

# 2.3 Algoritma Genetika

Menurut Berlianty dkk. (2010), ide awal algoritma genetika berasal dari teori Charles Darwin tentang evolusi yang berbasis pada konsep *survival of the fittest* yang menyatakan bahwa evolusi jenis-jenis spesies makhluk hidup dan ekosistemnya terjadi karena seleksi alam. Semakin tinggi kemampuan individu untuk beradaptasi, maka semakin tinggi kemungkinan individu tersebut dapat bertahan dan memiliki keturunan. Keturunan dari individu-individu tersebut akan mewarisi sifat-sifat induknya, dimana sifat-sifat tersebut dapat mengalami perubahan yang disebabkan oleh pencampuran sifat kedua induk maupun proses mutasi.

Algoritma Genetika ditemukan pertama kali pada tahun 1960. Algoritma Genetika merupakan salah satu algoritma pemodelan evolusi (evolutionary modelling) yang dikembangkan oleh John Holland pada dekade 1960 dan 1970-an dengan tujuan memodelkan perkembangan kemampuan adaptasi sebuah sistem. Algoritma genetika diimplementasikan sebagai simulasi yang berawal dari sebuah populasi yang dihasilkan secara random dan terdiri dari kromosom-kromosom, seperti halnya anggota tubuh makhluk hidup dan merepresentasikan solusi dari masalah. Populasi tersebut akan menghasilkan keturunan populasi yang baru dan diharapkan lebih baik dari populasi sebelumnya. Semakin baik kondisi suatu populasi, semakin besar kemungkinan populasi itu untuk dikembangkan menjadi populasi selanjutnya. Kondisi ini diulangi sampai mendapatkan kondisi yang diharapkan, dengan kata lain solusi terbaik sudah diperoleh.

### 2.3.1 Kelebihan Algoritma Genetika

Menurut Yanda (2014), ada beberapa kelebihan dalam mengaplikasikan algoritma genetika, yaitu sebagai berikut.

- Algoritma genetika tidak memerlukan kebutuhan matematis banyak mengenai masalah optimasi.
- 2. Kemudahan dan kenyamanan pada operator-operator evolusi membuat algoritma genetika sangat efektif dalam melakukan pencarian global.
- 3. Dapat digunakan untuk penyelesaian masalah optimasi, pemrograman otomatis, *machine learning*, model ekonomi, model sistem imunisasi, model ekologis, interaksi evolusi dan belajar.

Kelebihan lain dalam menggunakan algoritma genetika yaitu hasil yang didapat optimal dengan kesalahan yang kecil, serta dapat dijalankan dengan urutan dan efisiensi operasi yang telah ditentukan (Lestari dkk., 2014).

# 2.3.2 Istilah dalam Algoritma Genetika

Menurut Yanda (2014), istilah-istilah penting yang merupakan definisi untuk membangun penyelesaian masalah dalam algoritma genetika adalah sebagai berikut.

- 1. Gen, yaitu sebuah nilai yang menyatakan satuan dasar yang membentuk arti tertentu, dapat direpresentasikan dalam bentuk nilai biner, float, integer, karakter, daftar aturan, elemen permutasi, elemen program, maupun representasi lain yang dapat diimplementasikan untuk operator genetika.
- 2. Allele, merupakan nilai dari gen.
- 3. Kromosom, yaitu gabungan gen-gen yang membentuk nilai tertentu.
- Individu, yaitu nilai atau keadaan yang menyatakan salah satu solusi yang mungkin dari permasalahan yang diangkat.
- 5. Populasi, merupakan sekumpulan individu yang diproses bersama dalam satu siklus proses evolusi.
- Generasi, menyatakan satu siklus proses evolusi atau satu iterasi di dalam algoritma genetika.

Dalam penelitian ini, gen direpresentasikan dalam bentuk daftar aturan tiap ruangan yang ada. Daftar aturan yang dimaksud yaitu jenis, panjang, lebar, posisi horizontal, dan posisi vertikal ruangan. Oleh karena itu, kromosom

direpresentasikan sebagai ruangan yang dapat menjadi solusi dan jumlah individu dalam satu populasi sama dengan jumlah ruangan yang dimasukkan oleh pengguna.

# 2.3.3 Tahapan Algoritma Genetika

Tahapan algoritma genetika menurut Liliana (2006) antara lain sebagai berikut.

# 1. Inisialisasi populasi

Tujuan dari tahap inisialisasi adalah untuk menghasilkan suatu populasi awal yang terbentuk dari angka acak.

#### 2. Evaluasi Fitness

Tahap evaluasi *fitness* bertujuan untuk menentukan kelayakan tiap kromosom dalam populasi, sesuai atau tidaknya dengan keadaan yang diharapkan. Contoh *fitness*, untuk alasan sanitasi, maka dapur tidak boleh bersebelahan dengan kamar mandi, garasi selalu berada di sebelah depan atau belakang rumah, dan sebagainya.

### 3. Populasi baru

Proses membentuk populasi baru yang terbentuk dari populasi sebelumnya, disertai dengan proses-proses genetik sampai mendapatkan solusi yang terbaik. Setelah proses pembentukan populasi baru dapat terjadi *selection*, *crossover*, *mutation*, dan *accepting*.

Selection yaitu memilih populasi induk yang terbaik untuk dikembangkan menjadi populasi yang baru. Semakin baik kualitas populasi, maka semakin besar kemungkinan populasi ini terpilih sebagai induk. Kualitas suatu populasi

ditentukan dengan sejauh mana populasi tersebut memenuhi kriteria *fitness* yang telah ditentukan.

Crossover yaitu melakukan persilangan antar induk untuk mendapatkan keturunan yang baru. Jika tidak ada persilangan induk, maka keturunan yang dihasilkan akan memiliki sifat yang sama persis dengan induknya.

Mutation merupakan kemungkinan perubahan sifat di luar sifat induk pada keturunan yang dihasilkan. Mutation dapat berakibat perubahan menjadi sifat yang lebih baik dari induknya dan sebaliknya, bisa berubah menjadi lebih jelek dari induknya, tergantung dari fitness yang ada.

Accepting yaitu menggantikan populasi induk dengan populasi keturunan baru yang dihasilkan.

#### 4. Kondisi selesai

Jika kondisi terbaik sudah didapatkan, atau dengan kata lain sudah ada populasi yang sesuai dengan kebutuhan, maka proses dihentikan dan solusi terbaik sudah didapatkan. Jika solusi terbaik belum didapatkan, maka proses diulangi tahap menentukan *fitness*.

Menurut Khoiron dkk. (2013), rangka dari algoritma genetika secara umum adalah seperti yang ditunjukkan pada gambar 2.2.

### Mulai

Inisialisasi populasi baru dengan solusi acak Evaluasi tiap solusi dengan *fitness* 

#### **Iterasi**

Seleksi induk

Crossover sepasang induk

Mutasi keturunan

Evaluasi solusi baru

Seleksi untuk generasi selanjutnya

#### Berhenti

Kondisi kriteria berhenti telah terpenuhi

Gambar 2.2 Rangka Algoritma Genetika (Khoiron dkk., 2013)

Dalam penelitian ini, algoritma genetika yang digunakan adalah sebagai berikut.

# 1. Pembentukan populasi baru

Populasi baru berisi angka acak dan bertujuan untuk mencari populasi baru yang akan dibentuk dari populasi di generasi sebelumnya dan digunakan untuk menentukan posisi ruangan sesuai dengan permintaan pengguna. Berdasarkan penelitian yang dilakukan oleh Liliana (2006), populasi baru ini didapat dengan algoritma yang ditunjukkan pada gambar 2.3.

Diulang sebanyak panjang ruangan yang di-generate (i)

Diulang sebayak lebar ruangan yang di-generate (j)

Unsur (i,j) pada populasi k = crossover dari populasi ke x unsur (a,b)

dan ke y untur (c,d) dari populasi sebelumnya

Dimana:

a = Acak angka dari satu sampai panjang ruang

b = Acak angka dari satu sampai lebar ruang

c = Acak angka dari satu sampai panjang ruang

d = Acak angka dari satu sampai lebar ruang

x =Acak angka dari satu sampai empat (untuk mengoptimalkan solusi)

y = Acak angka dari satu sampai empat (untuk mengoptimalkan solusi)

k = Jumlah populasi

Gambar 2.3 Algoritma Penentuan Populasi Baru (Liliana, 2006)

#### 2. Evaluasi Fitness

Populasi individu baru yang didapat akan melewati proses pengecekan berdasarkan fungsi *fitness*. Nilai *fitness* tidak memerlukan suatu nilai absolut, tetapi yang digunakan adalah nilai relatif terhadap populasi tertentu. Berdasarkan penelitian yang dilakukan Lim dkk. (2004), fungsi *fitness* untuk menentukan apakah ruangan dapat diletakkan pada lahan yaitu sebagai berikut.

$$fitness\ ruangan = \frac{luas\ ruangan}{luas\ lahan\ tersisa} * 100\%.....Rumus\ 2.1$$

Dalam setiap generasi akan muncul kecenderungan konvergen yang diakibatkan terpilihnya kromosom dengan *fitness* tertinggi secara terus menerus,

selama *fitness* tidak melebihi 100%. Hal ini menjadi salah satu faktor berhentinya proses genetika, yang menunjukkan bahwa populasi terbaik telah didapatkan.

Beberapa pengecekan *fitness* yang dilakukan yaitu pengecekan letak (*x* dan *y*) kamar, ruang tamu, ruang makan, kamar mandi, dapur, garasi, gudang, ruang *laundry*, teras, dan halaman. Pengecekan berdasarkan perencanaan tata letak ruang dalam rumah seperti yang dijelaskan pada subbab 2.2. Jika letak dari ruangan sesuai dengan *constraint* yang ada, parameter *fit* jenis ruangan pada populasi yang sedang dievaluasi bertambah, sehingga kemungkinan terpilihnya ruangan tersebut sebagai solusi semakin besar.

Setelah pengecekan *constraint*, selanjutnya diperiksa apakah letak ruangan tersebut melebihi luas tanah yang diminta oleh pengguna tersedia atau tidak, melalui proses seleksi dari fungsi *fitness* tertinggi (rumus 2.1). Jika jenis ruangan pada proses iterasi sudah tersedia, pemeriksaan dilanjutkan ke ruangan selanjutnya. Menurut Liliana (2006), proses ini diulang selama solusi yang diminta belum dipenuhi dan ukuran *fitness*, ukuran panjang lantai, serta ukuran lebar lantai dari tiap ruangan dianggap belum mencukupi. Selama proses menentukan letak tiap ruang, dilakukan pencarian solusi yang terbaik untuk generasi tersebut, yaitu solusi yang memiliki nilai *fitness* terbesar.

### 3. Seleksi Induk

Dalam penelitian Putra dkk. (2012), proses seleksi yang dilakukan menggunakan metode *roulette wheel selection*. Metode yang diajukan John Holland ini pada dasarnya untuk menentukan proporsi probabilitas *survival* pada tiap kromosom sesuai dengan nilai *fitness*-nya. Individu dipetakan dalam suatu segmen garis secara berurutan sedemikian hingga tiap segmen individu memiliki ukuran

yang sama dengan ukuran *fitness*-nya. Sebuah bilangan *random* dibangkitkan dan

individu yang memiliki segmen dalam kawasan bilangan random tersebut akan

terseleksi. Proses ini diulang hingga diperoleh sejumlah individu yang diharapkan,

yaitu sebanyak 30 kali (Liliana, 2006). Pada metode ini induk dipilih berdasarkan

nilai fitness-nya, semakin besar nilai fitness maka akan semakin besar

kemungkinannya untuk terpilih menjadi induk.

4. Crossover

Crossover yang dipergunakan dalam penelitian Putra dkk. (2012) yaitu

position-based crossover, dimana kemungkinan terjadi crossover adalah 0,5.

Setelah didapat parent, langkah-langkah yang dilakukan saat melakukan crossover

yaitu sebagai berikut.

a. Memilih posisi pada *parent* pertama. Misalnya:

Parent 1: ABCDEFG

Parent 2: FGABCDE

b. Menyalin gen yang sudah terpilih sebagai proto-child.

Proto-child: A-C-E-G

c. Menghapus gen-gen pada parent kedua yang sama dengan proto-child.

*Parent* 2 : F - - B − D −

d. Menempatkan gen parent kedua ke proto-child, sehingga didapatkan

offspring.

Offspring: AFCBEDG

5. Mutasi

Proses mutasi menggunakan reciprocal exchange mutation (mutasi dengan

pertukaran timbal balik). Mutasi yang dilakukan bertujuan untuk mendapatkan

variasi individu dalam satu populasi. Kromosom yang mengalami mutasi dipilih secara acak posisi yang akan ditukar berdasarkan probabilitas mutasi, yaitu 0,1. Yanda (2014) menyebutkan bahwa jika semua kondisi telah terpenuhi, maka algoritma genetika akan menghentikan proses pencariannya, tetapi jika belum terpenuhi maka algoritma genetika akan kembali ke evaluasi *fitness*.

#### 6. Kondisi selesai

Proses genetika selesai jika mendapatkan individu terbaik dari evaluasi fitness, atau pada setiap generasi terjadi kecenderungan konvergen (nilai fitness yang sama atau saling mendekati dalam beberapa generasi). Hal ini berarti proses genetika berhasil mendapatkan hasil yang sesuai dengan masukan dari pengguna. Selanjutnya proses akan menggambarkan *layout* rumah berdasarkan nilai-nilai dalam populasi.

Selain itu, ada kemungkinan bahwa proses genetika tidak mendapatkan hasil yang sesuai dengan masukan dari pengguna, sehingga terjadi pengulangan proses secara terus-menerus. Untuk mencegah proses pengulangan tersebut, maka jumlah generasi maksimal yang dapat dibentuk adalah sebanyak 30. Jika proses sudah melebihi 30 generasi dan belum mendapatkan *fitness* yang sesuai, aplikasi akan memberikan pesan bahwa proses gagal.

### 2.4 Pengujian

Dalam suatu penelitian dibutuhkan pengujian terhadap populasi atau sampel penelitian untuk membuktikan penelitian yang dilakukan. Menurut Sugiyono (2010), populasi merupakan wilayah generalisasi yang terdiri atas objek atau subjek yang mempunyai kualitas dan karakteristik tertentu yang ditetapkan oleh peneliti untuk dipelajari dan kemudian ditarik kesimpulannya. Dari pengertian tersebut,

semua subjek berdasarkan karakteristik atau kondisinya dapat menjadi populasi pengujian. Populasi pengujian dalam penelitian menentukan rancangan tata letak ruang, yaitu semua *input* yang mungkin dimasukkan oleh pengguna.

Metode ujicoba aplikasi yang digunakan yaitu *blackbox testing*. Menurut Ayuliana (2009), metode ujicoba *blackbox* memfokuskan pada keperluan fungsional dari aplikasi, sehingga memungkinkan pengembang aplikasi membuat himpunan kondisi *input* yang akan melatih seluruh syarat-syarat fungsional suatu aplikasi. Ujicoba *blackbox* berusaha untuk menemukan kesalahan dalam beberapa kategori, yaitu fungsi yang salah atau hilang, kesalahan antarmuka, kesalahan struktur data, kesalahan performa, dan kesalahan inisialisasi atau terminasi. Pengujian dilakukan dengan melakukan validitas fungsionalitas dan membuat daftar *input* yang akan menghasilkan kasus uji yang baik.

Selain pengujian aplikasi, perlu dilakukan kuesioner untuk mengetahui tingkat kepuasan pengguna terhadap aplikasi (Sugiyono, 2010). Penentuan ukuran sampel untuk pengujian menurut Sugiyono (2010), yaitu sebagai berikut.

- Ukuran sampel yang layak dalam penelitian adalah antara 30 sampai dengan 500.
- Bila sampel dibagi dalam kategori, maka jumlah anggota sampel setiap kategori minimal 30.
- 3. Untuk penelitian eksperimen yang sederhana, menggunakan kelompok pengujian dan kelompok kontrol, maka jumlah anggota sampel masing-masing antara 10 sampai 20.

Berdasarkan penentuan ukuran sampel pada poin 1 dan poin 2 menurut Sugiyono, jumlah minimal pengguna yang dibutuhkan untuk mengisi kuesioner adalah 30, dan berdasarkan poin 3, ukuran sampel untuk kelompok pengujian kondisi minimal adalah sebanyak 10 sampel.

Menurut Sugiyono (2010), pertanyaan dalam kuesioner perlu diberikan bobot untuk mengetahui tingkat kepuasan pengguna, dimana pertanyaan tersebut memiliki 5 jenis jawaban yang memiliki bobot yang berbeda, yaitu sangat baik (bobot 5), baik (bobot 4), cukup (bobot 3), kurang (bobot 2), dan sangat kurang (bobot 1). Rata-rata dari jumlah bobot tiap pertanyaan tersebut dapat digunakan sebagai tolak ukur tingkat kepuasan pengguna.

