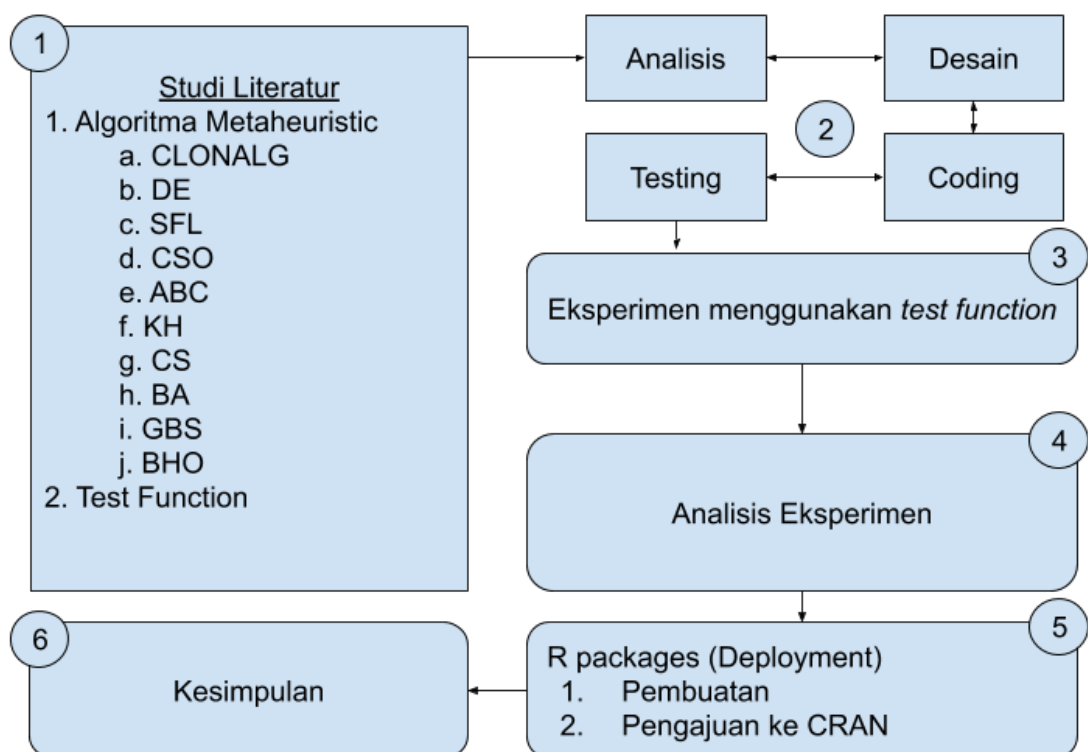


BAB 3 METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan mengenai metodologi penelitian yang dilakukan. Pertama akan dijelaskan desain penelitian. Selanjutnya, akan menjelaskan model pengembangan perangkat lunak yang digunakan dan pada bagian berikutnya akan menjelaskan alat dan bahan penelitian.

3.1 Desain Penelitian

Model Penelitian yang diterapkan bisa dilihat pada gambar 3.1:



Gambar 3.1 Model penelitian

Penjelasan model penelitian:

1. Studi Literatur

Pada tahap ini dilakukan, penulis melakukan pemahaman konsep algoritma metaheuristic terutama mempelajari *pseudocode* setiap algoritma. Algoritma

yang dipelajari antara lain *clonal selection algorithm*, *differential evolution*, *shuffled frog leaping*, *cat swarm optimization*, *artificial bee colony algorithm*, *krill-herd algorithm*, *cuckoo search*, *bat algorithm*, *gravitational based search algorithm* dan *black hole based optimization* (terdapat pada bab2 sub bab 2.3). Selain itu juga akan dipelajari beberapa test function (terdapat pada bab2 sub bab 2.2.4).

2. Implementasi (*Waterfall*)

a. Analisis

Menganalisa setiap kebutuhan baik dari sisi *software library* (*package*) maupun pengguna.

b. Desain

Menjabarkan hasil analisis agar bisa diimplementasikan ke tahap coding.

c. Coding

Melakukan penerapan hasil desain kedalam bahasa pemrograman R dan dibentuk kedalam *package*.

d. Testing

Pengujian fungsi agar terbebas dari *error*, *bug* dan sesuai dengan algoritma yang bersangkutan. Testing dilakukan dengan singkat menggunakan *sphere function* tujuannya agar fungsi bisa dengan mudah *didebug*. Selain itu dilakukan pengecekan kembali apakah kebutuhan *package* sudah terpenuhi.

Detail dari model *waterfall* yang digunakan dijelaskan pada bab 3.2.2 mengenai metode pengembangan perangkat lunak.

3. Eksperimen

Eksperimen dilakukan dengan mencoba beberapa *input* parameter fungsi Seperti fungsi objektif, banyak variabel, banyak populasi dan banyak iterasi. Fungsi objektif dicoba dengan menggunakan 13 *test function* yang dijelaskan pada bab 2.2.4 mengenai *test function* tabel 2.3 dan 2.4. Untuk pengaturan parameter fungsi dijelaskan pada bab 4 mengenai desain penelitian. Selain itu eksperimen juga berfungsi sebagai test kedua.

4. Analisis

Melakukan analisis dari data hasil eksperimen. Tujuan dilakukan analisis ini memastikan kembali setiap fungsi berjalan dengan baik tanpa *error*, *warning* dan sesuai dengan literatur. Juga melihat performa algoritma pada permasalahan yang bermacam-macam. Dua hal yang diperhatikan ketika melakukan analisis yaitu *output* terbaik dan waktu eksekusi. *Output* menentukan apakah fungsi tersebut berjalan dengan baik atau tidak. Suatu fungsi semakin baik jika nilai *output* mendekati global optima. Waktu eksekusi menentukan waktu yang dibutuhkan fungsi dalam melakukan optimasi.

5. R Packages

Setelah setiap fungsi berjalan dengan baik berdasarkan analisis eksperimen, fungsi tersebut akan ditambahkan kedalam R *packages* “*metaheuristicOpt*”. Berikutnya akan diupload ke CRAN. Tahap ini bisa kembali ke tahap sebelumnya jika CRAN mengeluarkan report.

6. Kesimpulan

Dari hasil analisis performa akan dilihat kelebihan dan kekurangan setiap fungsi.

3.2 Metode Pengembangan Perangkat Lunak

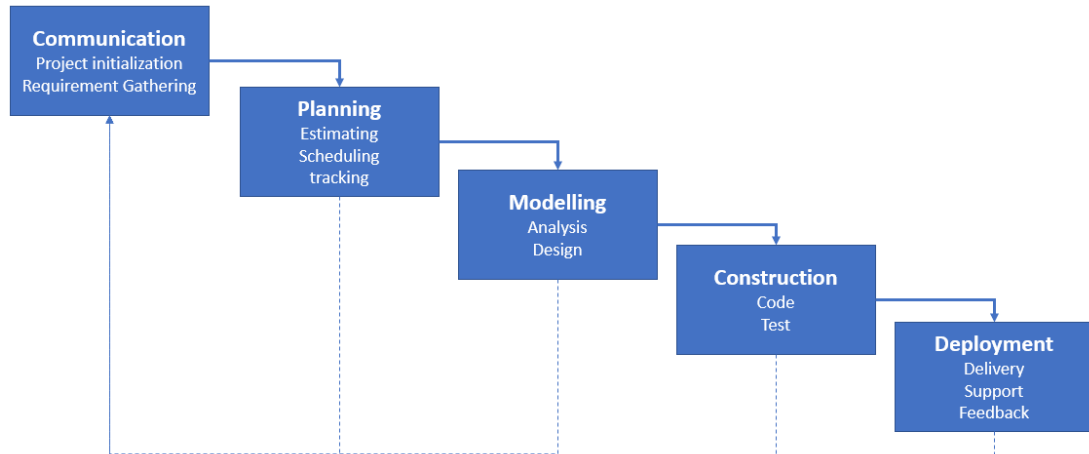
3.2.1 Pengumpulan Data

Data yang dikumpulkan berupa *test function* yang didapat dari studi literatur yang terdiri dari : *sphere model* (F_1), *schwefel's problem 2.22* (F_2), *schwefel's problem 1.2* (F_3), *schwefel's problem 2.21* (F_4), *generalized Rosenbrock's* (F_5), *step function* (F_6), *and quartic function with noise* (F_7), *generalized schwefel's problem 2.26* (F_8), *generalized Rastrigin's function* (F_9), *ackley's function* (F_{10}), *generalized griewank function* (F_{11}), *generalized penalized function 1* (F_{12}) dan *generalized penalized function 2* (F_{13}) akan digunakan untuk test pada tahap implementasi, Semua *test function* akan digunakan pada tahap analisis performa. *Test function* bisa dilihat pada bab 2 sub bab 2.2.4.

3.2.2 Metode Pengembangan Perangkat Lunak

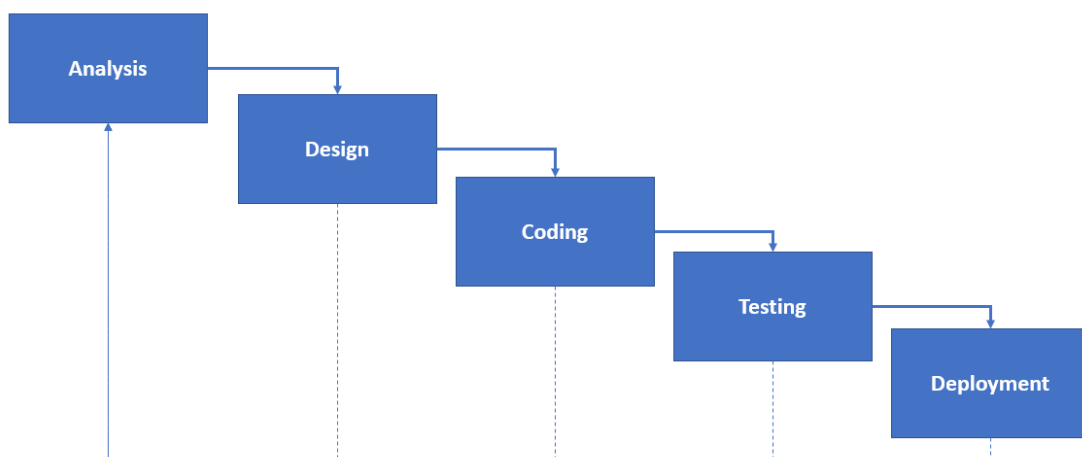
Metode pengembangan yang digunakan adalah *waterfall model*. *Waterfall model* adalah metode pengembangan perangkat lunak yang sangat sequential terdiri

dari *communication*, *planning*, *modelling*, *construction* dan *deployment* (Gambar 3.2) Metode ini sangat cocok untuk pembuatan perangkat lunak yang memiliki *requirement* yang jelas atau untuk mengembangkan perangkat lunak yang sudah ada (Pressman, 2005).



Gambar 3.2 Waterfall model

Terdapat 2 versi *waterfall* model menggunakan *feedback loop* dan linear. *waterfall* model menggunakan *feedback loop* dimana setiap tahapan bisa kembali ketahapan sebelumnya. Tipe *waterfall* ini merupakan *waterfall* model yang pertama kali diperkenalkan (Royce, 1970). Waterfall model linear dimana setiap tahapan tidak bisa kembali ketahapan sebelumnya. Kebanyakan organisasi menganggap *waterfall* hanya memiliki tipe linear. Tipe *waterfall* yang digunakan penulis adalah *waterfall* dengan *feedback loop* seperti pada gambar 3.3.



Gambar 3. 3 Waterfall model yang digunakan

Pada gambar 3.3 bisa dilihat ada beberapa komponen pada *waterfall* yang tidak digguakan seperti *communication* dan *planning*. Alasan penulis tidak menggunakan komponen *planning* karena penulis melakukan *coding* secara individu sehingga tidak diperlukan *scheduling* dan *tracking*. Berikutnya pembuatan *software library* tidak membutuhkan biaya tidak diperlukan *estimasting*. Pembuatan *software library* juga tidak memiliki pengguna yang spesifik sehingga tidak dibutuhkan *communication*. *Requirement* dari *software library* ditentukan oleh potensi pengguna dan karena mengembangkan *software library* yang sudah ada berarti juga ditentukan oleh *software library* itu sendiri. Bagian *requirement* termasuk dalam komponen analisis pada gambar 3.3. Detail dari *requirement software library* akan dijelaskan pada bab 4 sub bab 4.2.1 mengenai analisis. Tahap deployment dilakukan ketika pengajuan *package* ke CRAN.

3.3 Alat dan Bahan Penelitian

3.3.1 Alat Penelitian

Perangkat keras:

- Processor intel CORE i7
- Ram 8 GB, Ram 30 GB
- Monitor
- Keyboard
- Hardisk 1 TB

Perangkat lunak:

- Operating System Windows 10 64 bit
- Operating System Linux Ubuntu 18.04 (bionic beaver) 64 bit
- Visual Studio Code text editor
- R interpreter 64 bit versi 3.5.2 (Eggshell Igloo)
- R interpreter 64 bit versi 3.4.3 (Kite-Eating Tree)
- R interpreter 64 bit versi 3.5.3 (Great Truth)
- R Studio versi 1.1.463
- R Studio Server versi 1.1.463

3.3.2 Bahan Penelitian

Bahan penelitian yang digunakan adalah berkas seperti *papers*, *journal*, *textbook*, *article*, artikel *website*, *thesis* dan dokumentasi lainnya yang bisa didapatkan melalui observasi di perpustakaan, internet, dan sumber lainnya yang digunakan dalam algoritma metaheuristic dan fungsi optimisasi. Untuk pembuatan program R didukung dengan R documentation.