

BAB III. METODOLOGI PENGEMBANGAN

Pada bab ini terdiri dari langkah-langkah yang akan membimbing penulis memilih metode, teknik, prosedur apa yang tepat, dan tools apa yang akan digunakan sehingga setiap tahapan dapat dilakukan dengan tepat, termasuk desain dan perancangan sistem yang akan dibuat.

3.1 Analisis Kebutuhan Mitra

Analisis mengenai kebutuhan mitra dilakukan dengan wawancara. Hasil dari wawancara tersebut disimpulkan menjadi beberapa poin penting antara lain:

- Sistem yang memiliki kemampuan untuk menyimpan data admin dan data kasus positif COVID-19
- Data admin yang disimpan antara lain: 1. Nama lengkap, 2. Username, 3. Pengelola bagian, 4. Password
- Sistem harus bisa menampilkan data admin yang telah disimpan tanpa harus menunjukkan bagian *password*
- Selain menyimpan, sistem harus bisa melakukan operasi *update* dan *delete* data admin
- Sistem memiliki kemampuan untuk melakukan prediksi jumlah kasus positif COVID-19
- Pengguna sistem harus terlebih dahulu *login* untuk dapat menggunakan aplikasi

3.2 Deskripsi Sistem

Sistem Informasi Prediksi Jumlah Kasus Positif COVID-19 di Kota Probolinggo adalah sistem informasi yang dikembangkan untuk melakukan prediksi terhadap kasus positif COVID-19 di Kota Probolinggo dalam waktu satu hari mendatang. Sistem ini selain melakukan prediksi juga dapat menyimpan data admin atau pengguna yang akan menggunakan aplikasi ini. Data untuk melakukan peramalan merupakan data yang berbentuk *time series* yang telah diperoleh dari Dinas Kesehatan Kota Probolinggo. Sistem informasi ini menggunakan bahasa Python sebagai pengembangan dan MySQL sebagai basis data. Algoritma peramalan dalam sistem informasi ini adalah *Exponential Smoothing*.

3.3 Pengumpulan Data

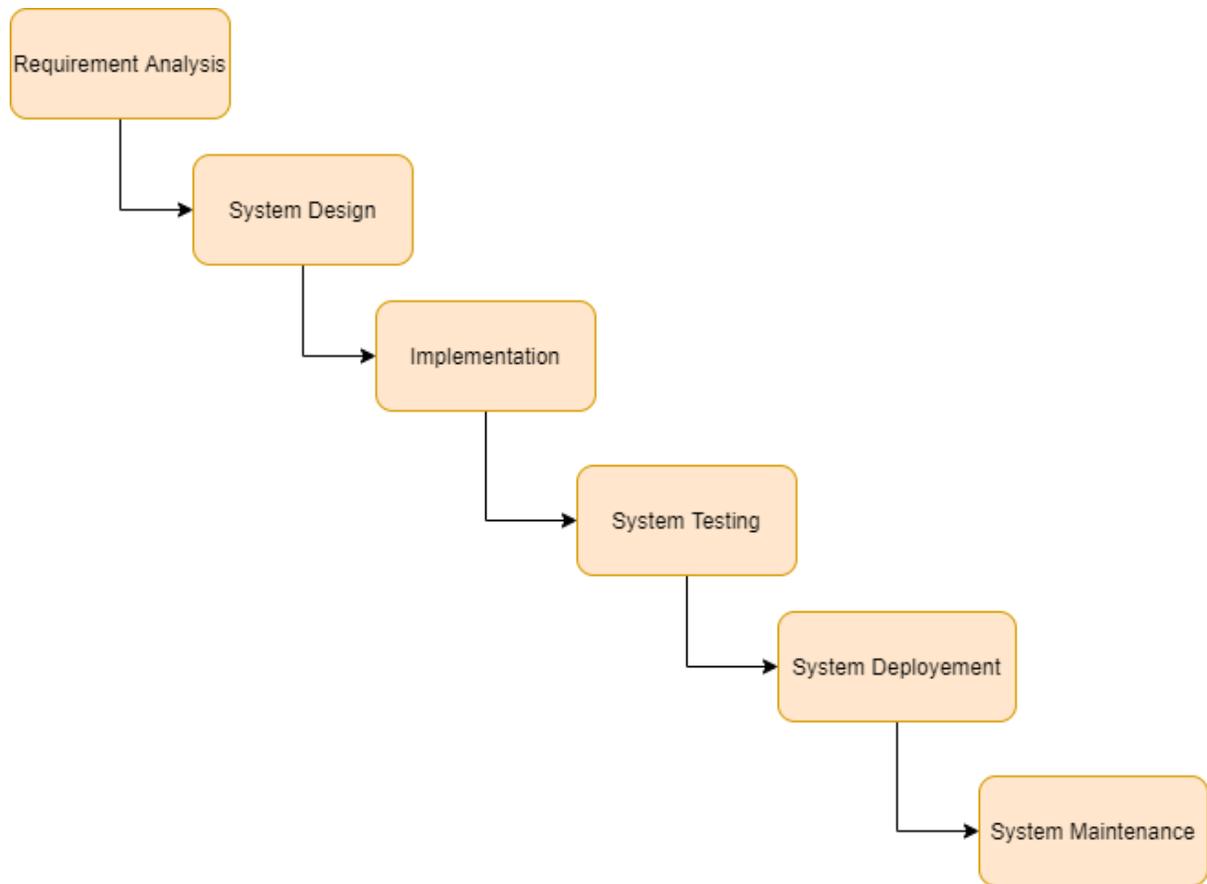
Pengumpulan data yang dilakukan dalam pengembangan ini adalah dengan memanfaatkan data kasus positif COVID-19 yang telah disediakan oleh Dinas Kesehatan Kota Probolinggo. Data yang diambil adalah data dari periode pertama kali COVID-19 masuk ke kota Probolinggo yaitu 4 April 2021 hingga tanggal 8 Mei 2021. Data yang didapatkan adalah data berupa deret waktu dengan jumlah total 370 entri.

3.4 Pengolahan Data

Pada pengembangan ini, akan dibuat aplikasi yang memiliki kemampuan untuk melakukan prediksi mengenai kasus positif COVID-19 di Kota Probolinggo. Data yang telah didapatkan, tidak sepenuhnya data yang utuh namun masih memiliki kekurangan yaitu kekosongan entri pada hari-hari tertentu. Data yang kosong tersebut diolah dengan metode naïve yaitu dengan beranggapan bahwa nilai suatu data yang hilang sama dengan nilai data yang sebelumnya.

3.5 Metode Pengembangan

Berdasarkan analisis kebutuhan mitra maka dapat disimpulkan bahwa sistem informasi ini dikembangkan dengan memanfaatkan model pengembangan *waterfall*. Alasan mengapa metode pengembangan ini digunakan adalah karena metode *waterfall* memiliki beberapa keunggulan dalam mengembangkan aplikasi yang kebutuhannya tidak sering berubah.



Gambar 3.1 SDLC Waterfall

3.5.1 Requirement Analysis

Requirement analysis adalah proses untuk mendefinisikan mengenai apa saja fungsionalitas sistem informasi. Proses ini dilakukan berdasarkan analisis kebutuhan mitra. Hasil dari tahap ini adalah menghasilkan kebutuhan fungsional dan non-fungsional yaitu sebagai berikut:

a. Kebutuhan Fungsional

- Sistem memiliki kemampuan untuk menyimpan data user
- Sistem memiliki kemampuan untuk melakukan prediksi kasus COVID-19 satu hari mendatang
- Sistem mampu melakukan operasi penyuntingan data user
- Sistem mampu melakukan operasi penghapusan data user
- Sistem mampu melakukan penyimpanan data terhadap kasus COVID-19
- Sistem memiliki proses login

b. Kebutuhan Non-fungsional

- Sistem memiliki proses eksekusi program kurang dari satu menit
- Aplikasi memiliki tampilan yang menarik
- Sistem dapat diakses kapan saja dengan kondisi *database online*
- Aplikasi menggunakan ruang penyimpanan kurang dari 20 MB

3.5.2 System Design

Pada tahap *system design* hasil kebutuhan dari *requirement analysis* dibentuk menjadi gambaran desain sistem yang akan dibuat. Desain yang dihasilkan dari tahap ini antara lain *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, dan *mockup*. Hal ini ditujukan agar pada tahap implementasi, seluruh fitur yang dikerjakan sesuai dengan yang direncanakan.

3.5.3 Implementation

Tahap implementasi adalah tahap dimana hasil dari perancangan desain diubah ke bentuk yang mampu dipahami oleh mesin. Implementasi ini dilakukan dengan menggunakan bahasa Python. Hasil dari tahap implementasi adalah berupa aplikasi yang akan siap dilakukan tahap uji coba.

3.5.4 System Testing

Hasil aplikasi dari tahap implementasi akan dilakukan proses uji coba. Uji coba yang dilakukan adalah dengan menggunakan metode *black box*. Pengujian ini dilakukan untuk mencari tahu apakah setiap fungsi yang dirancang telah berjalan sesuai dengan rancangan.

3.5.5 System Deployment

Aplikasi yang sudah melalui tahap pengujian dan seluruh fungsinya telah berjalan seperti yang telah dirancang, maka aplikasi telah siap untuk diterjunkan ke pengguna. Pada tahap *system deployment* aplikasi sudah bisa diterapkan di perangkat yang telah memenuhi kebutuhan minimal perangkat keras aplikasi.

3.5.6 *System Maintenance*

Aplikasi yang telah diterjunkan ke pengguna akan dilakukan perawatan dan evaluasi agar aplikasi bisa terus digunakan. Perawatan dan evaluasi yang akan dilakukan antara lain adalah dengan mencari tahu apakah terdapat *bug* terhadap aplikasi. Hal ini dilakukan agar aplikasi bisa tetap terus berjalan dan berkembang.