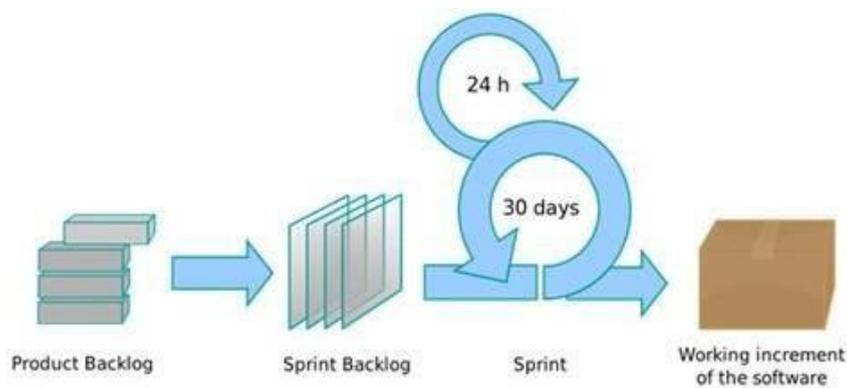


## BAB II TINJAUAN PUSTAKA

### 2.1 Metode Pengembangan *Scrum*

Metode pengembangan yang digunakan dalam pembuatan aplikasi ini yaitu dengan menggunakan metode *scrum*.



**Gambar 2.1 Metode *Scrum***

Model *scrum* merupakan metode pengembangan perangkat lunak secara cepat (*agile*). Model *scrum* memiliki beberapa kelebihan seperti, mampu mentransformasikan proses bisnis yang sulit menjadi mudah dikembangkan, dengan model *scrum* mampu memonitoring dan mengontrol aktivitas pada proses pengembangan sistem (Kuswinanti et al., 2021) Metode pengembangan sistem mengacu pada metode *Scrum* yang merupakan sebuah metode yang mudah dikontrol, fleksibel, memuat strategi pengembangan menyeluruh dimana seluruh tim bekerja sebagai satu unit untuk mencapai goal yang sama (Warkim et al., 2020) Proses pengembangan menggunakan metode *scrum* terdapat empat tahapan pengembangan yaitu : ((1) *product backlog*, (2) *sprint Backlog*, (3) *sprint log*, (4) *Increment*. Adapun kerangka kerja dalam model *scrum* terdiri dari:

*a. Product Backlog*

Tahap *Product Backlog* lebih mengarah kepada pengumpulan kebutuhan, pembaruan, pemeliharaan, dan deskripsi singkat tentang fungsi-fungsi yang diinginkan pada saat aplikasi akan dibangun.

*b. Sprint Backlog*

Tahap *Sprint backlog* dilakukan untuk sebuah proses pemenuhan kebutuhan

sesuai dengan yang diinginkan pada proses *Product Backlog* sebelumnya.

c. *Sprint Log*

Tahap *Sprint Log* merupakan proses dimana paparan aplikasi dalam bentuk sebuah *prototype* dan pemaparan dalam bentuk hal teknis baik berupa *tools* yang dibutuhkan untuk mengembangkan aplikasi.

d. *Increment*

Tahap *Increment* merupakan hasil akhir dari tahap *Product Backlog* yang telah selesai dikembangkan pada saat tahapan *Sprint Log*. Pada tahap ini, diharapkan tahapan *Increment* telah selesai dilakukan sehingga mampu untuk digunakan sesuai dengan yang diinginkan.

## 2.2 Algoritma Genetika

Turban (2005: 882) menyatakan bahwa “algoritma Genetika adalah kumpulan prosedur komputasional yang secara konseptual mengikuti langkah- langkah yang diinspirasi oleh proses evolusi biologis. Solusi yang lebih baik dan lebih baik lagi dikembangkan dari pembangkitan sebelumnya sampai solusi optimal atau hampir optimal didapatkan”.

Algoritma Genetika merupakan metode *heuristic* adaptif yang memiliki dasar pemikiran atau gagasan untuk proses seleksi alam dan genetika berdasarkan penelitian *Charles Darwin*. Dengan kata lain pencarian solusi suatu masalah dengan Algoritma Genetika akan terus berevolusi (Kusumadewi dan Purnomo, 2005). Algoritma ini didasarkan pada proses genetika yang ada dalam makhluk hidup yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti prinsip seleksi alam (siapa yang kuat, dia yang bertahan). Dengan meniru teori evolusi ini, Algoritma Genetika dapat digunakan untuk mencari solusi permasalahan-permasalahan yang ada dalam dunia nyata.

Keuntungan penggunaan Algoritma Genetika sangat jelas terlihat dari kemudahan implementasi dan kemampuannya untuk menentukan solusi seperti:

- a. Ruang masalah sangat besar, kompleks, dan sulit dipahami.
- b. Kurang atau bahkan tidak ada pengetahuan yang menandai untuk mempresentasikan masalah ke dalam ruang pencarian yang lebih sempit.
- c. Tidak tersedianya analisis matematika yang memadai
- d. Ketika metode-metode konvensional sudah tidak mampu menyelesaikan masalah yang dihadapi.

- e. Solusi yang diharapkan tidak harus paling optimal, tetapi cukup bisa diterima.
- f. Terdapat batasan waktu.

Algoritma genetika memiliki istilah-istilah yang digunakan untuk menggambarkan keadaan/permasalahan sebenarnya ke dalam teori Genetika, antara lain :

- a. Gen  
Gen merupakan nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen yang dinamakan kromosom.
- b. Allele  
Allele merupakan nilai dari gen.
- c. Kromosom / Individu  
Kromosom merupakan gabungan dari gen-gen yang membentuk nilai tertentu dan menyatakan solusi yang mungkin dari suatu permasalahan.
- d. Populasi  
Populasi merupakan sekumpulan individu yang akan diproses bersama dalam satu satuan siklus evolusi.
- e. *Fitness*  
*Fitness* menyatakan seberapa baik nilai dari suatu individu yang didapatkan.
- f. Seleksi  
Seleksi merupakan proses untuk mendapatkan calon induk yang baik. *Crossover*  
*Crossover* merupakan proses pertukaran atau kawin silang gen-gen dari dua induk tertentu.
- g. Mutasi  
Mutasi merupakan proses pergantian salah satu gen yang terpilih dengan nilai tertentu.
- h. Generasi  
Generasi merupakan urutan iterasi dimana beberapa kromosom bergabung, yaitu satu siklus proses evolusi atau satu iterasi di dalam algoritma genetika.
- i. *Offspring*  
*Offspring* merupakan kromosom baru yang dihasilkan.

## 2.3 Penerapan Algoritma Genetika

### 2.3.1 Membangun Generasi Awal

Langkah pertama dalam algoritma ini adalah membentuk sejumlah populasi awal yang digunakan untuk mencari penyelesaian optimal. Populasi awal yang dibangun

dalam skripsi ini dengan menggunakan bilangan random (acak) dengan range bilangan yang telah ditentukan (Adriana Fanggida, Fadly Rano Lado, 2015).

### 2.3.2 Fungsi *Fitness*

Fungsi *Fitness* digunakan untuk proses evaluasi kromosom agar memperoleh kromosom yang diinginkan, Fungsi ini membedakan kualitas dari kromosom untuk mengetahui seberapa baik kromosom yang dihasilkan fungsi *Fitness* tersebut sebagai berikut (Susanto Dkk 2018).

$$Fitness = \frac{1}{1 + Penalty}$$

Dari persamaan diatas nilai *Fitness* ditentukan oleh nilai *penalty*. *Penalty* tersebut menunjukkan jumlah pelanggaran kendala pada suatu kromosom semakin kecil nilai *penalty* (jumlah pelanggaran) semakin besar nilai *Fitness* nya. Jadi fungsi *Fitness* :

$$\frac{1}{1 + \sum Bp + \sum Np}$$

Keterangan:

Bp = Bobot pelanggaran

Np = Indikator Pelanggaran

### 2.3.3 Seleksi

Seleksi bertujuan untuk memberikan kesempatan individu dalam reproduksi bagi anggota populasi yang lebih unggul. Langkah pertama yang dilakukan dalam tahap seleksi adalah pencarian nilai *Fitness*, masing –masing individu dalam seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai yang diperoleh dibandingkan dengan anggota individu lainnya. Nilai *Fitness* inilah nantinya akan digunakan dalam tahap seleksi berikutnya (Yusuf Afandi, Wiji setiyaningsih,2019). Kemampuan algoritma genetika dalam menciptakan sebuah kromosom yang lebih unggul tergantung dari penekanan selektif terhadap kromosom dalam populasi penekanan selektif diterapkan dengan dua cara cara pertama adalah menciptakan lebih banyak kromosom anak dalam populasi dan memilih hanya kromosom yang terbaik untuk generasi berikutnya. Metode ini menerapkan seleksi orang tua secara acak namun metode ini akan terus menghasilkan kromosom yang lebih baik.

Cara lain menerapkan penekanan efektif adalah memilih orang tua yang lebih baik untuk proses regenerasi. Dengan metode ini hanya kromosom sebanyak jumlah tertentu dalam satu populasi yang dipelihara untuk generasi berikutnya. Walaupun

penekanan selektif diterapkan kepada orang tua dari calon kromosom terpilih.

#### **2.3.4 Seleksi roulette**

Seleksi Roulette adalah salah satu metode seleksi individu yang tetap melibatkan keanekaragaman populasi sesuai dengan namanya, prinsip kerja dari Roulette Wheel dimana masing-masing individu menempati potongan lingkarannya pada roda Roulette secara proporsional sesuai dengan nilai *Fitness* (Adriana Fanggida, Fadly Rano Lado, 2015).

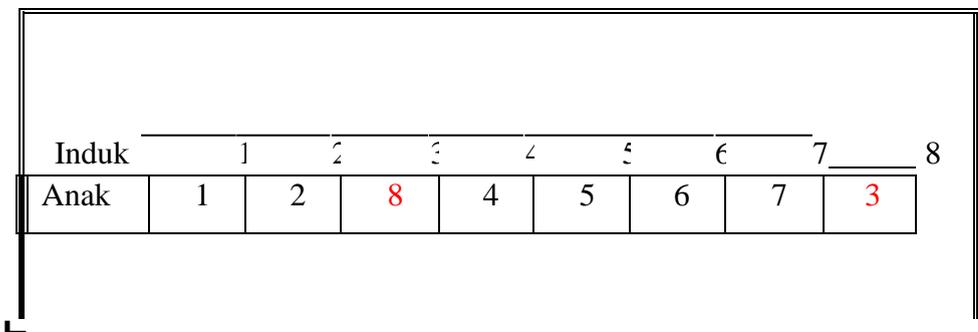
#### **2.3.5 Crossover**

*Crossover* atau kawin silang merupakan metode mengawinkan dua kromosom parent (orang tua). Dengan tujuan untuk mendapatkan anak yang lebih baik. Dalam suatu populasi yang sangat kecil suatu kromosom dengan gen-gen yang kromosom-kromosom lainnya, untuk mengatasi hal ini digunakan suatu aturan bahwa *crossover* hanya melibatkan kromosom-kromosom yang memiliki nilai acak  $[0,1]$  yang nilainya lebih kecil dari probabilitas *crossover* ( $pc$ ) dengan kata lain kromosom yang memiliki nilai acak  $< pc$  adalah menjadikan kromosom-kromosom yang perlu diperbaiki dengan cara menjadikan kromosom-kromosom tersebut sebagai parent dalam *crossover* untuk mendapatkan kromosom-kromosom yang lebih baik (Suryanto, 2007)

#### **2.3.6 Mutation**

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam satu kromosom. Perubahan gen dilakukan dengan berbagai cara : mengubah gen 1 menjadi 0 dan sebaliknya, melakukan pertukaran gen pada suatu posisi dengan gen pada posisi lain, mengubah gen dengan batasan tertentu dan sebagainya bergantung pada representasi individunya. Tujuan dari mutasi adalah mempercepat perbedaan diantara semua kromosom dalam populasi sehingga pencarian dapat dipetakan ke seluruh ruang. Selain itu mutasi bertujuan untuk mengembalikan komponen penting yang hilang ketika proses *crossover* (Suryanto, 2007).

Contoh :



**Gambar 2.2 Contoh Mutasi (Suryanto, 2007)**

Ada beberapa jenis operator :

a. Mutasi Terarah

Mutasi terarah sangat bergantung pada informasi gen. informasi gen berupa nilai pelanggaran gen (bagian dari nilai *Fitness*). Hal ini berarti bahwa adanya kemungkinan untuk gen yang berbeda akan di mutasi. mutasi akan dilakukan jika pelanggaran gen tersebut lebih besar dari gen yang lain dalam satu kromosom.

b. Mutasi biasa

Mutasi biasa tidak memperhitungkan informasi gen. hal ini berarti setiap gen memiliki peluang yang sama untuk terjadi mutasi.

## 2.4 HyperText Preprocessor (PHP)

Menurut Sidik (2017:4), "PHP secara umum dikenal sebagai bahasa pemrograman script-script yang membuat dokumen HTML secara on the fly yang dieksekusi di server web, dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML, dikenal juga sebagai bahasa pemrograman server side." Menurut Harianto,dkk (2019:13), "PHP merupakan software Open-Source yang disebarakan dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya".

## 2.5 MySQL (My Structure Query Language)

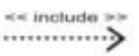
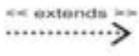
Menurut Harianto, dkk (2019:13), "MySQL adalah salah satu jenis data-base server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang database sebagai sumber dan pengelolaan datanya".

## 2.6 Use Case Diagram

Merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. secara kasar usecase digunakan untuk mengetahui fungsi

apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Simbol dan keterangan use case diagram seperti pada tabel 2.1.

**Tabel 2.1 Simbol dan Keterangan Use case Diagram**

SIMBOL	NAMA	KETERANGAN
	<i>Actor</i>	Seseorang atau apa saja yang berhubungan dengan sistem yang sedang dibangun
	<i>Use case</i>	Menggambarkan bagaimana seseorang menggunakan system
	<i>Relasi asosiasi</i>	Relasi yang dipakai untuk menunjukkan hubungan antara <i>actor</i> dan <i>use case</i>
	<i>Relasi include</i>	Memungkinkan satu use case menggunakan fungsionalitas yang disediakan oleh usecase lainnya
	<i>Relasi extend</i>	Memungkinkan suatu use case secara optional menggunakan fungsionalitas yang disediakan oleh usecase lainnya

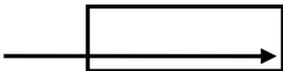
Sumber : (Nugroho, 2010)

## 2.7 Data Flow Diagram (DFD)

DFD merupakan gambaran sistem secara logika yang tidak tergantung pada perangkat keras, lunak, struktur data dan organisasi file. Keuntungan dari DFD adalah untuk memudahkan pemakai yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan atau dikembangkan (Sukrianto & Oktarina, 2019).

Tabel 2.2 berikut ini adalah Simbol-simbol yang digunakan dalam Data flow Diagram :

**Tabel 2.2 Simbol Data Flow Diagram**

SIMBOL	KETERANGAN
<i>External Entity</i> 	Merupakan sumber atau tujuan dari aliran data dari atau ke system
Arus Data ( <i>Data Flow</i> )	Menggambarkan aliran data
Proses	Proses atau fungsi yang mentransformasikan data masukan menjadi keluaran.

Simpanan Data	Komponen yang berfungsi untuk menyimpan data atau file.
---------------	---------------------------------------------------------

## 2.8 Penelitian Sebelumnya

Dalam penyusunan skripsi ini, peneliti terinspirasi dan mereferensi dari penelitian penelitian sebelumnya yang berkaitan dengan skripsi ini. Daftar penelitian terkait ialah sebagai berikut :

**Tabel 2.4 Penelitian Terkait**

No	Judul, Penulis, Tahun	Jumlah dan Atribut	Algoritma/Metode
1.	Perancangan Aplikasi Penjadwalan Mata Kuliah (Studi Kasus : STMIK Provisi Semarang) (Hartadi et al., 2016)	Ada 3 Atribut : <i>Genetic algorithms, Drag and Drop, CourseScheduling</i>	Genetik
2.	Rancang Bangun Aplikasi Penjadwalan Skripsi Dan Skripsi Berbasis Web Menggunakan Framework Laravel (Utama & Nuryana, 2020)	Ada 4 atribut : <i>scheduling, final thesis, web-based, Laravel framework</i>	<i>Waterfall</i>
3	Aplikasi Penjadwalan Ujian Skripsi Dan Kerja Praktek (Wahit Ambo et al., 2021)	Ada 3 atribut : <i>Scheduling Application, Thesis, practical work</i>	<i>prototype</i>

4	Pengembangan Sistem Informasi Penjadwalan Konsultasi Dosen Fakultas Ilmu Komputer Universitas Brawijaya (Mursityo, 2020)	Ada 4Atribut: <i>scheduling,</i> <i>information</i> <i>systemdevelopment,</i> <i>communication,</i> <i>consulting</i>	
5	Prototipe Sistem Pengelolaan Ujian Pendadaran Skripsi Mahasiswa (Rizan et al., 2019)	Ada 5 Atribut : <i>thesis,thesisdefence,</i> <i>object oriented</i> <i>methods, prototype</i> <i>model, web</i>	<i>object oriented</i>
6	Sistem Penjadwalan Skripsi Mahasiswa Program Studi Teknik Informatika Menggunakan Algoritma Genetika Berbasis Website (sutyono., 2022)	Ada 3 atribut : <i>genetic algorithm,</i> <i>scheduling,lecture</i>	<i>Genetika</i>