

# Perancangan Aplikasi Penyusun Jadwal Berbasis Web Menggunakan Algoritma Semut Pada SMA Khalifa IMS Tangerang Selatan

Rahmat Yudha<sup>1\*</sup>, Ari Mulyoto<sup>1</sup>

<sup>1</sup>Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: <sup>1\*</sup>[rahmatyudha0998@gmail.com](mailto:rahmatyudha0998@gmail.com), <sup>2</sup>[dosen00236@unpam.ac.id](mailto:dosen00236@unpam.ac.id)

(\* : coresponding author)

**Abstrak**—Perancangan aplikasi penyusunan jadwal berbasis Web dengan menggunakan algoritma semut mampu memberikan solusi alternatif pada penyusunan jadwal yang menginginkan waktu tertentu secara optimal. Pada penelitian ini metode yang digunakan adalah metode sistem *development life cycle (paradigma waterfall)*. Metode ini terdiri dari berbagai tahap yaitu Analisis kebutuhan, Perancangan system, dan Penulisan program. Dari hasil analisis didapatkan: Penerapan aplikasi penyusunan jadwal berbasis web menggunakan algoritma semut, dapat membantu penyusunan jadwal secara cepat dan mudah sekaligus menghindari bentrok dalam penyusunan jadwal. Disisi lain proses penyusunan jadwal dengan menggunakan metode algoritma semut ini dapat memepersingkat waktu pengerjaannya.

**Kata Kunci:** *Aplikasi Penyusunan Jadwal Berbasis Web, Algoritma Semut*

**Abstract**— *The design of a Web-based scheduling application using the ant algorithm is able to provide an alternative solution for preparing schedules that want a certain time optimally. In this research, the method used is the system development life cycle method (the waterfall paradigm). This method consists of various stages, namely needs analysis, system design, and program writing. From the results of the analysis, it was found: The application of a web-based scheduling application using the ant algorithm, can help prepare schedules quickly and easily while avoiding clashes in scheduling. On the other hand, the process of preparing a schedule using the ant algorithm method can shorten the processing time.*

**Keywords:** *Web-Based Scheduling Application, Ant Algorithm*

## 1. PENDAHULUAN

Pada jaman yang serba cepat ini. banyak hal atau pekerjaan yang harus dilakukan dengan cepat dan tertata. Setiap orang atau instansi memerlukan penjadwalan dalam melakukan kegiatan. Arti dari jadwal adalah pembagian waktu berdasarkan pengaturan urutan kerja, daftar atau tabel kegiatan dan rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Penjadwalan mempunyai arti proses, cara perbuatan menjadwalkan atau memasukkan di jadwal (Setiawan. 2012). Kegiatan atau pekerjaan yang terjadwal dapat mempermudah seseorang atau institusi untuk melakukan kegiatan yang sudah tersusun secara teratur. Banyak sekali contoh dari penjadwalan seperti penjadwalan kereta api, penjadwalan mesin produksi, penjadwalan mata kuliah dan penjadwalan mata pelajaran. Salah satu contoh penjadwalan adalah penjadwalan mata pelajaran pada SMA KHALIFA IMS di Tangerang Selatan.

Bagi SMA KHALIFA IMS di Tangerang Selatan penjadwalan mata pelajaran merupakan hal yang sangat penting, terlebih karena penjadwalan mata pelajaran berfungsi untuk mengatur kegiatan belajar-mengajar. Dalam membuat suatu penjadwalan, banyak faktor yang perlu diperhatikan, diantaranya guru, ruang, waktu belajar, jumlah siswa dan banyak mata pelajaran. Hal ini juga terjadi pada SMA KHALIFA IMS di Tangerang Selatan dalam pembuatan jadwal, seperti sering kali terjadi dimana jadwal pelajaran mengalami bentrok baik dari segi guru, kelas maupun waktu pembelajaran dan faktor lain yang menjadi kesulitan adalah dibutuhkan waktu yang lama untuk membuat satu jadwal dalam satu semester, karena banyaknya data guru, mata pelajaran dan ruang yang mana diperlukan ketelitian karena masih dibuat secara manual.

Terdapat cara pembuatan jadwal secara komputerisasi menggunakan suatu algoritma. Algoritma semut merupakan salah satu algoritma yang dapat digunakan untuk membantu pembuatan jadwal pembelajaran secara komputerisasi. Algoritma semut merupakan suatu algoritma yang dibuat dengan mengadopsi cara semut menemukan jarak terpendek menuju suatu makanan (Dorigo, 1996).

Algoritma ini bekerja seperti semut yang mencari makanan dari sarang menuju sumber makanan. Setiap perjalanannya, semut akan bebas menuju suatu titik yang akan dilalui menuju sumber makanan. Setiap perjalanannya, semut akan meninggalkan jejak berupa feromon, yang nantinya akan berguna ketika dalam perjalanan selanjutnya. Jika suatu titik dilalui oleh banyak semut, maka semut akan berjalan ke titik tersebut dan tidak melalui titik lain yang tidak banyak dilalui oleh semut, sehingga mendapatkan jarak terpendek dalam pencarian makanan. Contoh pemakaian algoritma Semut diantaranya adalah untuk Perancangan Sistem Optimasi Rute Distribusi Pengangkutan Sampah di Surabaya. Tidak berhenti pada penggunaan untuk mencari jarak terpendek, algoritma semut juga dapat digunakan untuk membuat penjadwalan mata pelajaran.

## 2. METODOLOGI PENELITIAN

Metode yang digunakan dalam menyusun jadwal mata pelajaran di SMA KHALIFA IMS adalah metode sistem *development life cycle (paradigma waterfall)*. Metode ini terdiri dari berbagai tahap:

### 2.1 Analisis Kebutuhan

Pada tahap ini, akan dilakukan identifikasi tentang kebutuhan system serta pengumpulan data-data yang diperlukan dalam proses penjadwalan.

### 2.2 Perancangan Kebutuhan

Pada tahap ini, akan dilakukan penerjemahan rancangan system kedalam perangkat lunak. Proses perancangan meliputi perancangan system design basis data, user interpace dan rancangan implementasi program.

### 2.3 Implementasi dan Pengujian

Pada tahap ini, akan dilakukan pengujian program. Pengujian bertujuan untuk menemukan kesalahan-kesalahan hingga program dapat sesuai dengan yang diharapkan.

## 3. ANALISA DAN PEMBAHASAN

Analisa merupakan suatu aktifitas menguraikan, membedakan atau memilih suatu sistem untuk diklasifikasi atau dikelompokkan kembali menurut kriteria tertentu yang kemudian di cari kaitannya dan ditafsirkan maknanya. Analisa juga bisa dimaknai sebagai suatu sikap atau perhatian terhadap suatu sistem sampai mampu untuk menguraikan menjadi bagian - bagian lalu mengenali kaitan antara bagian dalam keseluruhan. (Yusuf, Rohman 2019).

### 3.1 Analisis Kebutuhan Perangkat Lunak

#### 3.1.1 Metode Analisis

Metode penyusunan jadwal yang digunakan oleh SMA Khalifa IMS Tangerang Selatan masih secara manual yaitu dengan membuat tabel di Microsoft Excel, oleh karena itu penyusunan jadwal menjadi tidak efektif, sehingga dibutuhkannya aplikasi untuk membuat penyusunan jadwal agar lebih efektif dan optimal. Aplikasi untuk menentukan jalur terpendek ini dirancang dengan menggunakan algoritma semut. Untuk melihat proses aplikasi yang mencakup proses input dan proses output dinyatakan dengan diagram sistem yang diperjelas dengan diagram alir (*flow chart*). Pada tahap ini digunakan notasi-notasi untuk menggambarkan arus data dimana akan sangat membantu dalam proses komunikasi dengan pemakai.

#### 3.1.2 Hasil Analisis

Input dari aplikasi penentuan jalur terpendek ini, berupa parameter-parameter yang diperlukan dalam penyusunan jadwal dan algoritma semut yaitu:

- a. Jurusan, yang meliputi:
  1. Nama Jurusan

- b. Data Guru, yang meliputi:
  1. Nama Guru
- c. Data Mata Pelajaran, yang meliputi:
  1. Koode mata pelajaran
  2. Nama Mata Pelajaran
  3. Semester
- d. Data Ruang, yang meliputi:
  1. Kode Ruang
  2. Nama Ruang

### 3.1.3 Analisis Kebutuhan Proses

Untuk mendapatkan hasil penjadwalan yang optimal dengan menggunakan algoritma semut dibutuhkan parameter-parameter antara lain:

- a. Intensitas jejak semut antar titik ( $\tau_{ij}$ ) dan perubahannya ( $\Delta\tau_{ij}$ )  $\tau_{ij}$  harus diinisialisasi sebelum siklus.  $\tau_{ij}$  digunakan dalam persamaan probabilitas titik yang akan dikunjungi.  $\Delta\tau_{ij}$  diinisialisasi setelah selesai satu siklus.  $\Delta\tau_{ij}$  digunakan untuk menentukan  $\tau_{ij}$  untuk siklus selanjutnya.
- b. Tetapan siklus semut ( $Q$ ),  $Q$  merupakan kostanta yang digunakan dalam persamaan untuk menentukan  $\Delta\tau_{ij}$ . Nilai  $Q$  ditentukan oleh pengguna.
- c. Tetapan pengendali intensitas jejak semut ( $\alpha$ ),  $\alpha$  digunakan dalam persamaan probabilitas titik yang akan dikunjungi yang berfungsi sebagai penegedali intensitas jejak semut. Nilai  $\alpha$  ditentukan oleh pengguna.
- d. Tetapan pengendali visibilitas ( $\beta$ ),  $\beta$  digunakan dalam persamaan probabilitas titik yang akan dikunjungi dan berfungsi sebagai pengendali visibilitas. Nilai  $\beta$  ditentukan oleh pengguna.
- e. Visibilitas antar titik ( $\eta_{ij}$ ), ( $\eta_{ij}$ ) digunakan dalam persamaan probabilitas titik yang akan dikunjungi. Nilai  $\eta_{ij}$  merupakan hasil dari  $1/d_{ij}$  (jarak antar titik).
- f. Banyak semut ( $m$ ),  $m$  merupakan banyak semut yang akan mealukan siklus dalam algoritma semut. Nilai  $m$  ditentukan oleh pengguna.
- g. Tetapan penguapan jejak semut ( $\rho$ ),  $N_{Cmax}$  adalah jumlah maksimum siklus yang berlangsung. Siklus akan berhenti sesuai dengan  $N_{Cmax}$  yang telah ditentukan atau telah konvergen.  $N_{Cmax}$  ditentukan oleh pengguna.

### 3.1.4 Analisis Kebutuhan Output

Data output yang diperoleh dari proses aplikasi penentuan jalur terpendek ini adalah kombinasi antara kode mata pelajaran, nama mata pelajaran, nama guru, hari, waktu (jam pelajaran dilaksanakan), ruang yang digunakan dan kelas sehingga dihasilkan jadwal yang optimum.

## 3.2 Perancangan Perangkat Lunak

### 3.2.1 Metode Perancangan

Metode perancangan yang dikembangkan untuk membangun pencarian jalur terpendek dengan menggunakan *context diagram* dan *data flow diagram* (DED) yang kemudian diperjelas dengan *flow chart* atau perancangan terstruktur (*structure design method*). *Flow chart* pada dasarnya merupakan konsep perancangan yang mudah dengan penekanan pada sistem modular (*Top Down Design*) dan pemograman terstruktur (*structure programming*).

### 3.2.2 Perancangan

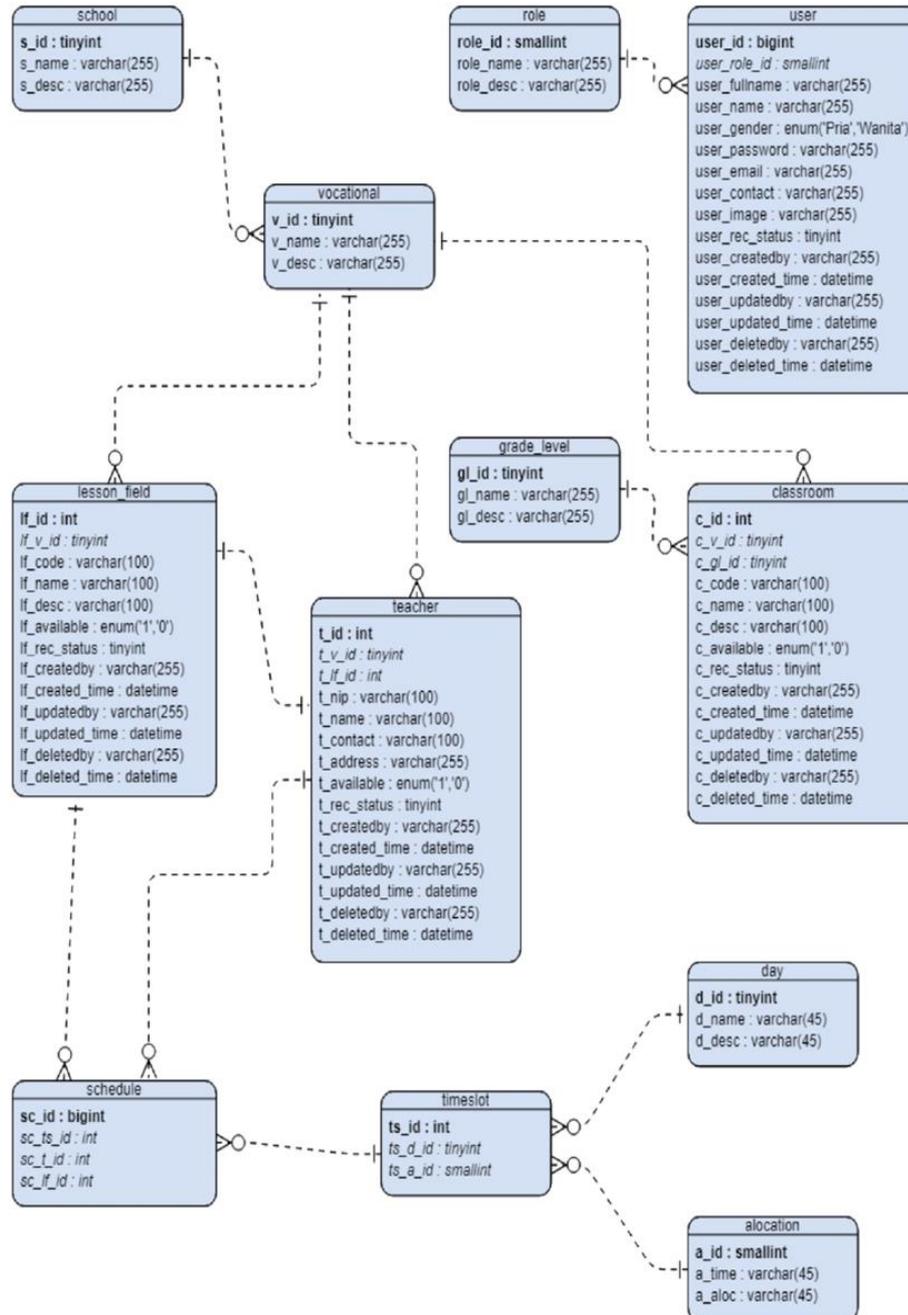
Berdasarkan analisis yang telah dilakukan maka dapat diketahui apa saja yang menjadi masukan sistem, kelauran sistem, metode yang digunakan sistem, serta sistemantar muka (*interface*) yang dibuat, sehingga sistem yang dibuat nantinya sesuai dengan yang diharapkan.

#### a. Perancangan Desain Sistem

Untuk lebih menjelaskan perancangan sistem yang dibangun maka dipergunakan entyti relationship dan *flow chart*.

### 1. Entity Relationship Diagram (ERD)

Pembuatan ERD dimaksudkan agar dapat memodelkan suatu database yang didalamnya terdapat relasi yang dihubungkan oleh *field* kunci.

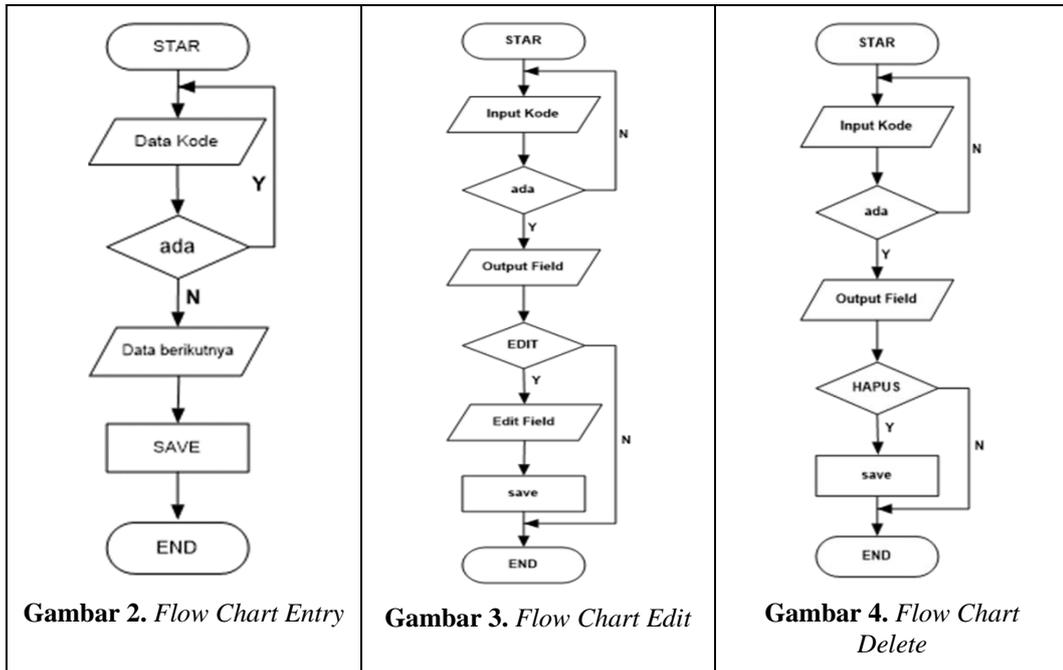


**Gambar 1.** Entity Relationship Diagram (ERD)

### 2. Flow Chart

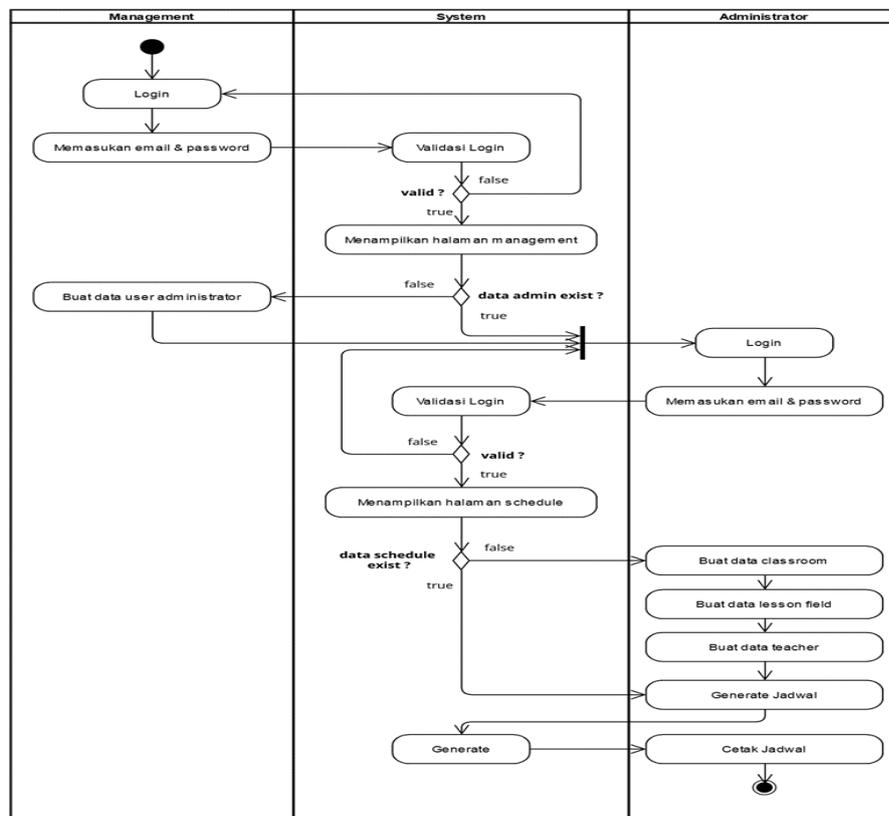
Pada halaman admin terdapat beberapa form untuk memasukan (*entry*) data yang dibutuhkan, meng*update* data yang sudah ada pada database dan menghapus data yang tidak dibutuhkan lagi dalam proses penjadwalan.

Pembuatan *Flow Chart* ditujukan untuk memperjelas urutan sistem program secara umum, untuk lebih jelasnya dapat dilihat pada gambar berikut:



### 3. Activity Diagram

Bentuk visual dari alur kerja yang berisi aktivitas dan tindakan, yang juga dapat berisi pilihan, atau pengulangan. Dalam *Unified Modeling Language (UML)*, diagram aktivitas dibuat untuk menjelaskan aktivitas computer maupun alur aktivitas dalam organisasi.



**Gambar 5. Activity Diagram**

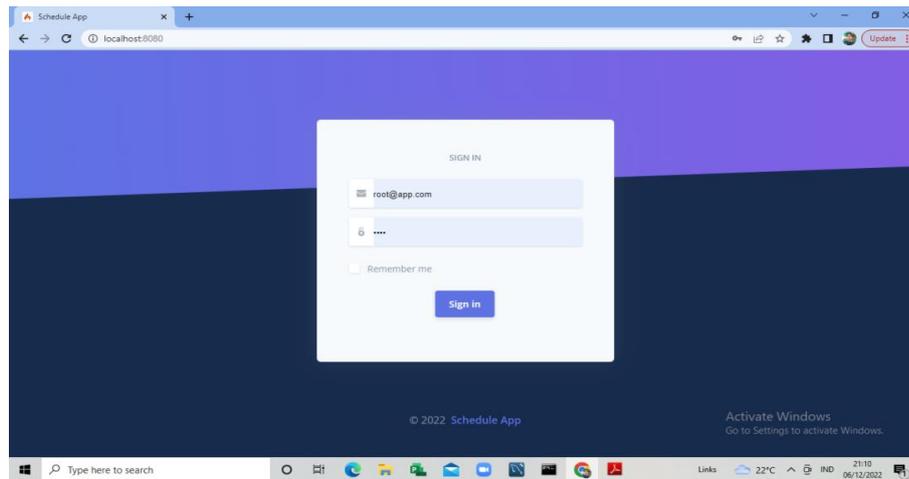
## 4. IMPLEMENTASI

### 4.1 Implementasi Aplikasi

Tahap implementasi system merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan system. Tahap implementasi merupakan menerjemahkan perancangan berdasarkan hasil analisis dalam Bahasa yang dapat dimengerti oleh mesin serta penerapan.

#### 4.1.1 Halaman *Login*

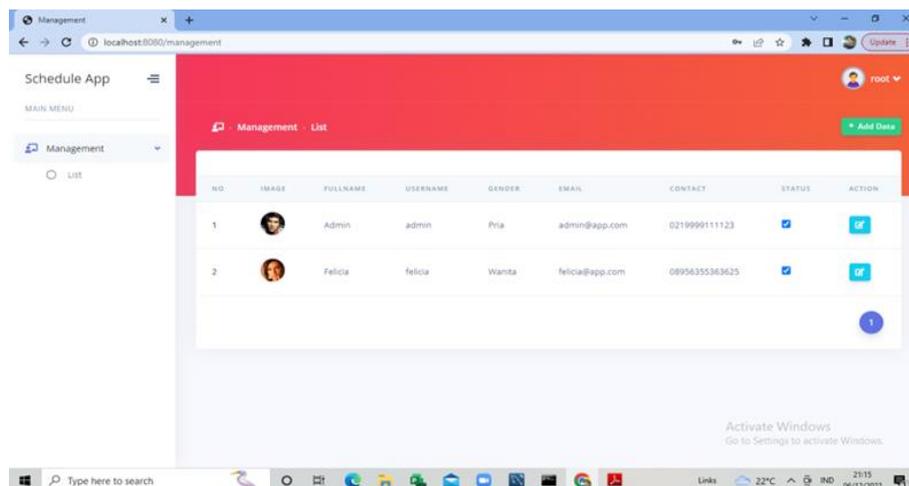
Pada halaman ini terdapat dua pilihan akun, yaitu akun root dan akun admin. Dimana kedua akun tersebut memiliki fungsi yang berbeda saat melakukan akses kedalam sistem.



**Gambar 6.** Halaman *Login*

#### 4.1.2 Halaman *Root*

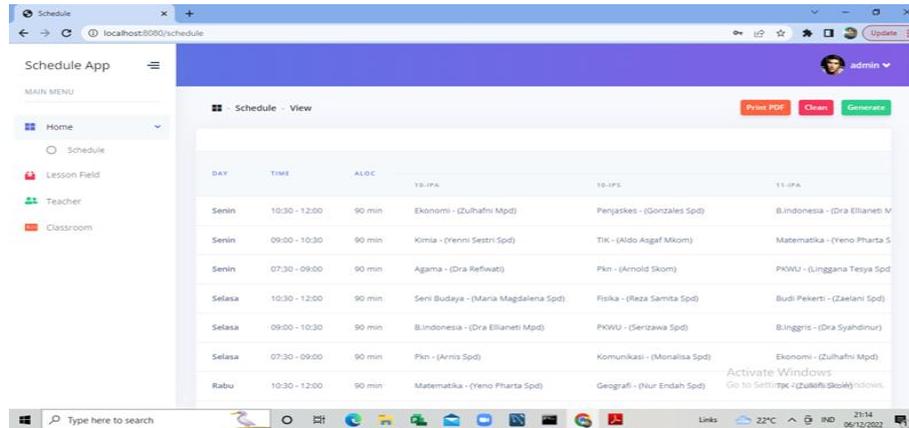
Halaman ini adalah tampilan setelah kita melakukan login menggunakan akun root. Pada akun root ini user dapat menambahkan data admin dan juga dapat membekukan akun admin tersebut dengan cara menghilangkan centang pada salah satu akun yang terdaftar.



**Gambar 7.** Halaman *Root*

#### 4.1.3 Halaman *Admin*

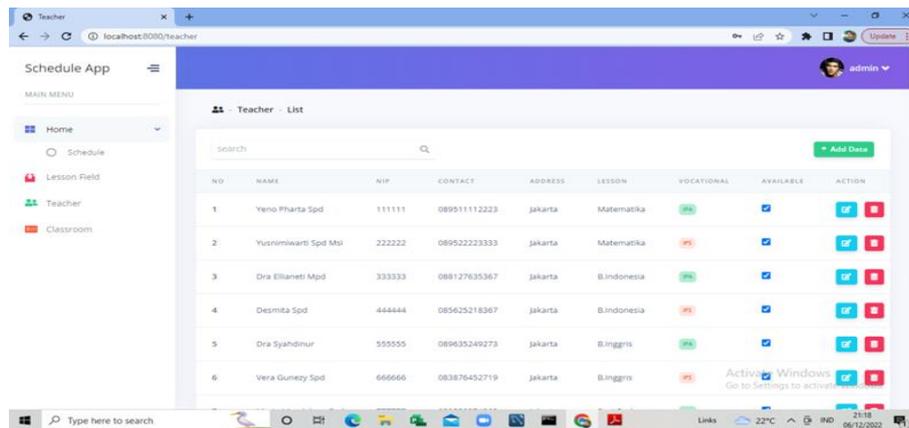
Halaman ini adalah tampilan setelah kita melakukan login menggunakan akun admin. Pada akun admin ini user dapat menambahkan dan menghapus data seperti guru, mata pelajaran dan kelas hingga sampai dengan hasil jadwal pelajaran yang sudah berbentuk pdf.



**Gambar 8.** Halaman Admin

#### 4.1.4 Halaman Guru

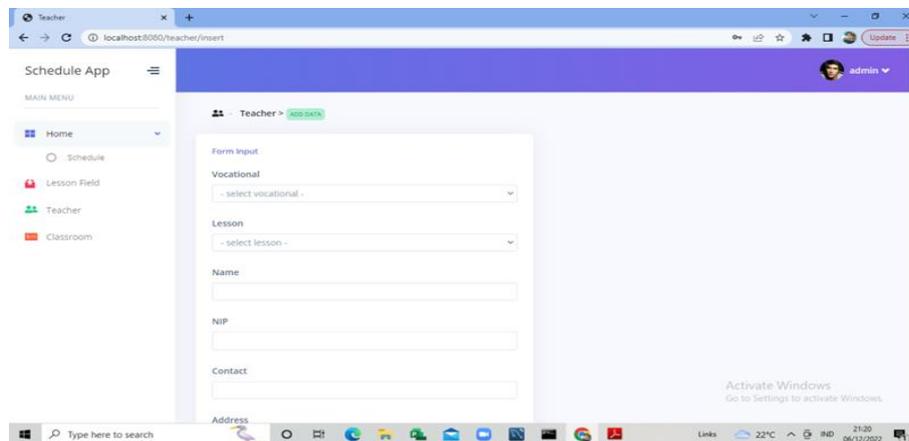
Halaman ini adalah halaman setelah kita pasuk pada pilihan *teacher* . Pada halaman ini terdapat data berupa nama, NIP, No Hp, alamat, bidang pelajaran dan jurusan.



**Gambar 9.** Halaman Guru

#### 4.1.5 Halaman Penambahan Guru

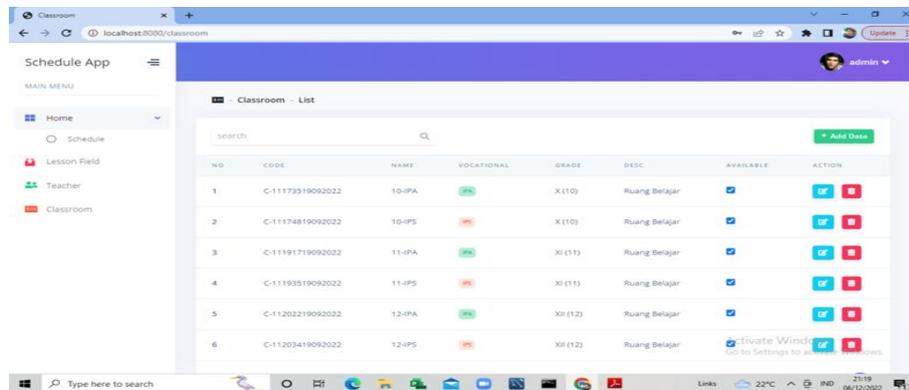
Pada halaman ini admin dapat mengisi pada setiap form yang kosong sesuai data yang tersedia.



**Gambar 10.** Halaman Penmbahan Guru

#### 4.1.6 Halaman Kelas

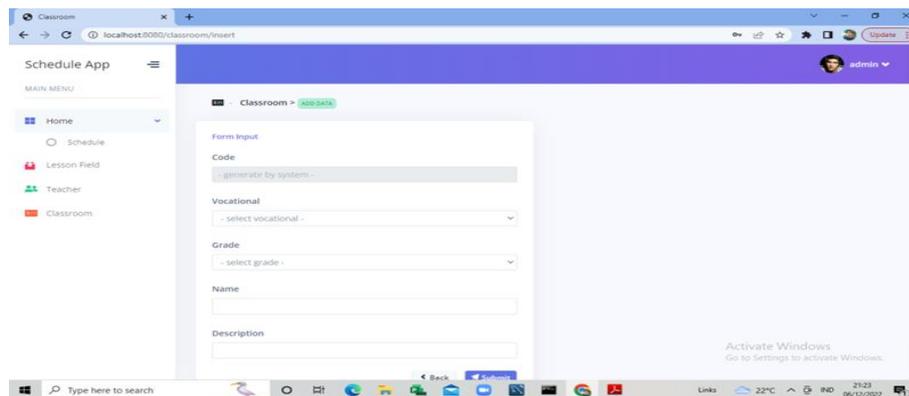
Halaman ini adalah halaman setelah kita masuk pada pilihan *classroom* . Pada halaman ini terdapat data berupa kode kelas dan tingkatan kelas.



Gambar 11. Halaman Kelas

#### 4.1.7 Halaman Penambahan Kelas

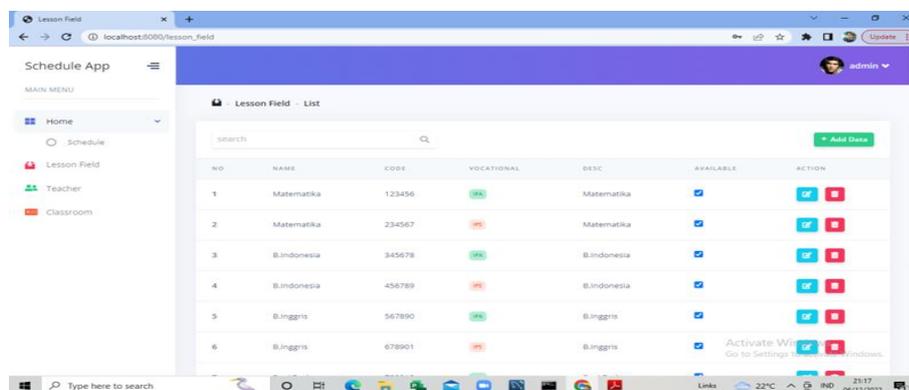
Pada halaman ini admin dapat mengisi pada setiap form yang kosong sesuai data yang tersedia.



Gambar 12. Halaman Penambahan Kelas

#### 4.1.8 Halaman Mata Pelajaran

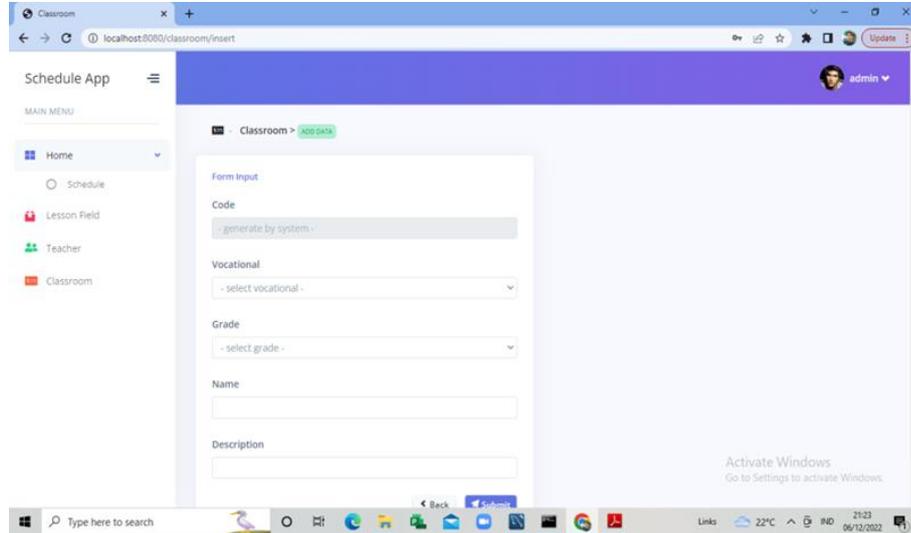
Halaman ini adalah halaman setelah kita masuk pada pilihan *lesson field* . Pada halaman ini terdapat data berupa mata pelajaran, kode mata pelajaran dan jurusan.



Gambar 13. Halaman Mata Pelajaran

#### 4.1.9 Halaman Penambahan Mata Pelajaran

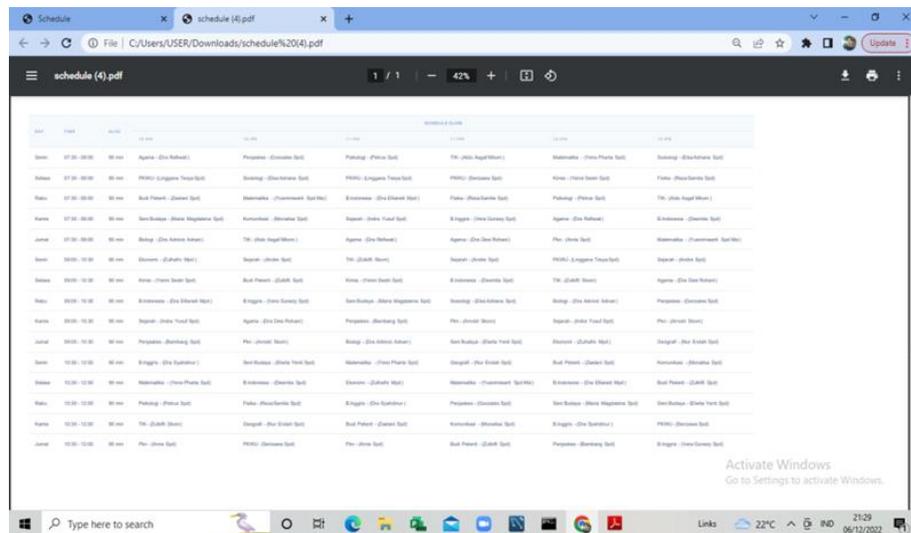
Halaman ini berisi form yang diperlukan pada saat memasukan data mata pelajaran.



**Gambar 14.** Halaman Penambahan Mata Pelajaran

#### 4.1.10 Halaman Jadwal Pelajaran

Halaman ini merupakan hasil dari proses yang telah dilakukan pada proses – proses diatas.



**Gambar 15.** Halaman Jadwal Pelajaran

## 5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan peneliti serta uraian tersebut dikemukakan pada bab-bab sebelumnya mengenai perancangan aplikasi penyusunan jadwal berbasis web menggunakan algoritma semut pada SMA Khalifa IMS Tangerang Selatan. Dengan adanya aplikasi ini dapat mempermudah untuk penyusunan jadwal yang lebih optimal.

Penerapan aplikasi penyusunan jadwal berbasis web menggunakan algoritma semut, akan dapat membantu penyusunan jadwal secara cepat dan mudah sekaligus menghindari bentrok dalam penyusunan jadwal. Disisi lain proses penyusunan jadwal dengan menggunakan metode algoritma semut ini dapat memepersingkat waktu pengerjaannya.



## REFERENCES

- Andi Sunyoto, 2007. *Membangun Web Dengan Teknologi Asynchronous Javascript dan HTML*, Penerbit ANDI, Yogyakarta.
- Hariyadi. M.Amin.MT, 2007, *Al-Quran dan Semut: Inspirasi Al-Quran dalam Membangun Algoritma Ant*, Malang: UIN-Malang Press.
- M. Dorigo dan L. M. Gambardella, *Ant Colonies for the Traveling Salesman Problem*, 1997.
- M. Dorigo, V. Maniezzo, dan A. Colomi, *The Ant System: Optimization By A Colony of Cooperating Agents*, IEEE Transactions on Systems, 1996.
- M. Dorigo, V. Maniezzo, dan A.Colorni, 1996, Positive Feedback As A Search Startegy, (Teach. Rep. 91-016), Milan, Italy: Politecno di Milano, Dipartimento di Elettronica.
- Morton, Thomas., Pentico, David W. (1993). *Heuristic Scheduling Systems: With Applications to Production Systems and Project Management*. Wiley-Interscience, First Edition.
- R. Beckers, J.L Deneubourg, dan S.Goss, 1992, Trail and U-turns in the Selection of the shortest Path by the Ant *Lasius Niger*, *Journal of theoretical biologi*.
- Setiawan, Ebta. 2012. Kamus Besar Bahasa Indonesia (KBBI) "Kamus versi online daring (dalam jaringan)". [online]. (<http://kbbi.web.id/sinonim>, diakses tanggal 22 September 2013).
- Yu, J. (2006). Scheduling of an assembly line with a multi-objective genetic algorithm. *International Journal of Advanced Manufacturing Technology*, 551-555.
- Zukhri, Zainudin, 'Analisis Algoritma Semut Untuk Pemecahan Masalah Penugasan', *Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNASTI 2005)*, 2005.Snati (2005), 47-51