

**SISTEM PAKAR MENENTUKAN JIWA  
*ENTREPRENUERSHIP***

**SKRIPSI**



**Oleh:  
Septiadi  
140210020**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK DAN KOMPUTER  
UNIVERSITAS PUTERA BATAM  
TAHUN 2018**

**SISTEM PAKAR MENENTUKAN JIWA  
*ENTREPRENUERSHIP***

**SKRIPSI**  
**Untuk memenuhi salah satu syarat**  
**guna memperoleh gelar Sarjana**



**Oleh:**  
**Septiadi**  
**140210020**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK DAN KOMPUTER**  
**UNIVERSITAS PUTERA BATAM**  
**TAHUN 2018**

## **PERNYATAAN HALAMAN PERNYATAAN**

Dengan ini menyatakan bahwa:

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (sarjana, dan/atau magister), baik di Universitas Putera Batam maupun di perguruan tinggi lain.
2. Skripsi ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing.
3. Dalam skripsi ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh, serta saksi lainnya sesuai dengan norma yang berlaku di perguruan tinggi.

Batam, 28 Juli 2018  
Yang membuat pernyataan,

materai 6.000,-

Septiadi  
NPM: 140210020

**SISTEM PAKAR MENENTUKAN JIWA  
*ENTREPRENUERSHIP***

**HALAMAN PENGESAHAN**

Oleh  
**Septiadi**  
**140210020**

**SKRIPSI**

Untuk memenuhi salah satu syarat  
guna memperoleh gelar Sarjana

Telah disetujui oleh Pembimbing pada tanggal  
seperti tertera dibawah ini

**Batam, 28 Juli 2018**

**Algifanri Maulana, S.SI., M.MSI.**

**Pembimbing**

## ABSTRAK

Penelitian ini dilatar belakangi oleh permasalahan ekonomi yang sedang rendah di kota batam. Hal ini dikarenakan kurangnya pengusaha yang ada dikota batam. Wirausaha adalah seseorang yang berani membuka suatu usaha yang dapat menciptakan lapangan pekerjaan bagi orang lain, dan berjiwa gigih yang berani mengambil segala resiko yang ada untuk membuka usaha dalam berbagai kesempatan. Tetapi banyak individu yang tidak menyadari apakah mereka sudah mempunyai jiwa *entrepreneuership*. Oleh karena itu dibutuhkan untuk membangun suatu aplikasi berbasis sistem pakar yang mempunyai pengetahuan untuk bantu menentukan jiwa *entrepreneuership*. Sistem ini menggunakan metode *forward chaining* yaitu metode interferensi yang menggunakan penalaran yang dimulai dari fakta dahulu untuk menguji kebenaran hipotesis. Sistem ini dibuat berbasis Web yang menggunakan bahasa pemograman yang diterapkan dalam pembangunan aplikasi tersebut adalah bahasa pemograman *PHP* dan basis data *MySQL* sehingga merupakan sistem yang mudah diakses para pengguna baik dikomputer ataupun *smartphone* hasil penelitian menunjukan bahwa aplikasi sistem pakar tersebut dapat membantu para pengguna untuk menentukan jiwa *entrepreneuership*. Kesimpulan yang bisa diambil pada penelitian ini adalah. Aplikasi sistem pakar telah berhasil diterapkan dengan metode *forward Chaining* dan bahasa pemograman php untuk menentukan jiwa *entrepreneuership*. Sehingga bisa Membuat dan memperluas kesempatan kerja jadi dengan adanya ini mengurangi pengangguran yang ada dikota Batam.

**Kata Kunci:** Sistem Pakar, Jiwa *entrepreneur*, *MYSQL*, *PHP*, *Forward Chaining*

## ABSTRACT

*This research is based on the low economic problem in batam city. This is because the lack of entrepreneurs in the city batam. The entrepreneur is someone who dares to open a business that can create jobs for other people, and indomitable spirit who dared to take all the risks that exist to open a business in a variety of occasions. But many individuals are unaware whether they already have an entrepreneurial characteristic. Therefore it is necessary to build an expert system based application that has knowledge to help determine probality succes of entrepreneurship. This system uses forward chaining method that is interference method using reasoning starting from the first fact to test the truth of the hypothesis. This system created a Web-based programming language that is applied in the development of such applications is the programming language PHP and MySQL database so it is a system that is easily accessible to users either's computer or smartphone Research results show that the application of expert systems can help users to determine the soul entrepreneurship. The conclusion drawn in this study is through this application, users can consult a computer or smartphone with appropriate systems in direct consultation with an expert to determine the entrepreneurship soul. So it can Make and expand employment opportunities so with this is expected to reduce unemployment in the city batam.*

**Keyword:** *Expert System, Entrepreneur, MYSQL, PHP, Forward Chaining*

## KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati.

Dengan segala keterbatasan, penulis menyadari pula bahwa skripsi ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Rektor Universitas Putera Batam.
2. Ketua Program Studi Teknik Informatika Universitas Putera Batam.
3. Algifanri Maulana, S.SI., M.MSI. selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam.
4. Dosen dan Staff Universitas Putera Batam.
5. Kepada orang tua penulis, yang terus mendoakan keberhasilan penulis menyelesaikan skripsi ini.
6. Teman-teman seperjuangan yang juga selalu memberikan motivasi baik berupa *sharing* pendapat, motivasi dan hal-hal lainnya dalam rangka pembuatan skripsi ini.

7. Serta semua pihak yang tak dapat penulis sebutkan satu persatu yang telah membantu dalam penyusunan skripsi ini.

Semoga Tuhan Yang Maha Esa membalas kebaikan dan selalu mencurahkan hidayah serta taufik-Nya, Amin.

Batam, 28 Juli 2018

Penulis



## DAFTAR ISI

<b>HALAMAN PERNYATAAN</b> .....	<b>i</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>ii</b>
<b>ABSTRAK</b> .....	<b>iii</b>
<b>ABSTRACT</b> .....	<b>iv</b>
<b>KATA PENGANTAR</b> .....	<b>v</b>
<b>DAFTAR ISI</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>x</b>
<b>DAFTAR GAMBAR</b> .....	<b>xi</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1. 1. Latar Belakang .....	1
1.2. Identifikasi Masalah .....	4
1.3. Batasan Masalah.....	4
1.4. Rumusan masalah.....	5
1.5. Tujuan Penelitian .....	5
1.6. Manfaat Penelitian .....	6
1.6.1. Manfaat Teoritis .....	6
1.6.2. Manfaat Praktis .....	6
<b>BAB II LANDASAN TEORI</b> .....	<b>7</b>
2.1. Teori Dasar.....	7
2.1.1. Kecerdasan Buatan (Artificial Intelligence).....	7
2.1.2. Sistem Pakar (Expert System).....	10
2.1.2.1. Konsep Sistem Pakar.....	11
2.1.2.2. Keuntungan Sistem Pakar .....	12
2.1.2.3. Kelemahan Sistem Pakar.....	13
2.1.2.4. Bentuk Sistem Pakar .....	13
2.1.2.5. Struktur Sistem Pakar.....	14
2.1.2.6. Mengembangkan Sistem pakar .....	17
2.1.2.7. Ciri-Ciri Sistem Pakar .....	19
2.1.2.8. Teknik inferensi foward chaining dan backward chaining .....	19
2.1.2.9. Strategi Penyelesaian Konflik.....	21
2.1.3. Jaringan Syaraf Tiruan .....	21

2.1.4. <i>Fuzzy Logic</i> .....	22
2.2. Sifat .....	24
2.3. Software Pendukung .....	26
2.3.1 UML (Unified Modeling Language).....	26
2.3.1.1. <i>Use Case Diagram</i> .....	27
2.3.1.2. <i>Activity Diagram</i> .....	29
2.3.1.3. <i>Sequence Diagram</i> .....	31
2.3.1.4. <i>Class Diagram</i> .....	32
2.3.2. <i>Xampp</i> .....	35
2.3.2.1. <i>PHP</i> .....	35
2.3.2.2. <i>MYSQL</i> .....	37
2.4. Penelitian Terdahulu .....	39
2.5. Kerangka Pemikiran.....	43
<b>BAB III METODE PENELITIAN .....</b>	<b>44</b>
3.1 Desain Penelitian.....	44
3.2 Pengumpulan data .....	46
3.2.1 Wawancara.....	46
3.2.2 Metode Literatur.....	46
3.3 Operasional Variabel.....	47
3.3.1 <i>Forward Chaining</i> .....	47
3.4 Perancangan Sistem .....	51
3.4.1 Diagram <i>UML</i> .....	51
3.4.2 Desain Basis Data .....	62
3.4.3 Desain Antarmuka.....	63
3.5 Lokasi Dan Jadwal Penelitian .....	68
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>69</b>
4.1 Hasil Penelitian .....	69
4.1.1 Implementasi Antar Muka.....	69
4.2 Pembahasan.....	76
4.2.1 Pengujian Validasi .....	76
<b>BAB V SIMPUL DAN SARAN.....</b>	<b>77</b>
5.1 Simpulan .....	77
5.2 Saran.....	77
<b>DAFTAR PUSTAKA</b>	

**DAFTAR RIWAYAT HIDUP  
SURAT KETERANGAN PENELITIAN  
LAMPIRAN**

## DAFTAR TABEL

<b>Tabel 2.1</b> Sistem Pakar yang terkenal .....	14
<b>Tabel 2.2</b> Simbol-simbol pada <i>Use Case</i> .....	27
<b>Tabel 2.3</b> Simbol-simbol pada <i>activity diagram</i> .....	29
<b>Tabel 2.4</b> Simbol-simbol pada <i>Sequence</i> diagram.....	31
<b>Tabel 2.5</b> Simbol-simbol pada <i>class diagram</i> .....	34
<b>Tabel 3.1</b> Operasional Variabel.....	47
<b>Tabel 3.2</b> Tabel sifat .....	48
<b>Tabel 3.3</b> Tabel hasil .....	49
<b>Tabel 3.4</b> Tabel aturan.....	49
<b>Tabel 3.5</b> Tabel Basis Data user .....	62
<b>Tabel 3.6</b> Tabel Basis Data Pertanyaan.....	62
<b>Tabel 3.7</b> Tabel Basis Data hasil .....	63
<b>Tabel 4.1</b> Hasil Pengujian Validasi .....	76

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Struktur Sistem Pakar .....	15
<b>Gambar 2.2</b> Tahap-Tahap Pengembangan sistem pakar .....	18
<b>Gambar 2.3</b> Kerangka pemikiran.....	43
<b>Gambar 3.1</b> Desain penelitian .....	44
<b>Gambar 3.2</b> Pohon Keputusan .....	50
<b>Gambar 3.3</b> Use Case Diagram .....	51
<b>Gambar 3.4</b> Activity Diagram Admin .....	52
<b>Gambar 3.5</b> Activity Diagram Member.....	53
<b>Gambar 3.6</b> Activity Diagram Non-member.....	54
<b>Gambar 3.7</b> Sequence Diagram Login Admin .....	55
<b>Gambar 3.8</b> Sequence Diagram input data pertanyaan .....	56
<b>Gambar 3.9</b> Sequence Diagram ubah data pertanyaan .....	57
<b>Gambar 3.10</b> Sequence Diagram Input data hasil .....	58
<b>Gambar 3.11</b> Sequence <i>Diagram</i> ubah data hasil.....	59
<b>Gambar 3.12</b> Sequence <i>Diagram</i> Ubah <i>Password</i> .....	60
<b>Gambar 3.13</b> <i>Sequence Diagram</i> <i>User Access</i> .....	61
<b>Gambar 3.14</b> <i>Class Diagram</i> .....	62
<b>Gambar 3.15</b> Menu Form Login.....	63
<b>Gambar 3.16</b> Menu Form Daftar .....	64
<b>Gambar 3.17</b> Menu form kelola user.....	64
<b>Gambar 3.18</b> menu tambah, edit, hapus pertanyaan.....	65
<b>Gambar 3.19</b> Menu tambah, edit, hapus hasil .....	66
<b>Gambar 3.20</b> Menu Menjalani tes .....	66
<b>Gambar 3.21</b> Menu Hasil Tes.....	67
<b>Gambar 3.22</b> Menu Ubah Password.....	67
<b>Gambar 4.1</b> Tampilan utama .....	69
<b>Gambar 4.2</b> Halaman pendaftaran .....	70
<b>Gambar 4.3</b> Halaman Kelola <i>user</i> .....	70
<b>Gambar 4.4</b> Halaman Kelola Pertanyaan .....	71
<b>Gambar 4.5</b> Halaman Kelola Hasil.....	72
<b>Gambar 4.6</b> Halaman menjalani tes.....	73
<b>Gambar 4.7</b> Halaman hasil tes.....	74
<b>Gambar 4.8</b> Halaman ubah password.....	75

# **BAB I**

## **PENDAHULUAN**

### **1. 1. Latar Belakang**

Pada kondisi saat ini perekonomian *global* tidak stabil, efek yang dapat dirasakan kalangan masyarakat adalah pasar yang semakin lesu atau daya jual beli yang semakin berkurang. Sehingga pada saat kondisi seperti ini pemerintah semakin menganjurkan bagi para penerus generasi muda untuk menjadi seorang pengusaha atau menciptakan lapangan pekerjaan agar bisa membantu menekan perekonomian negara yang semakin menurun.

Kota Batam merupakan salah satu kota yang letak sangat strategis. Selain berada di jalur perairan internasional, dan berada dekat/ berada di pebatasan dengan negara Singapura dan Malaysia. Kota Batam merupakan salah satu kota dengan pertumbuhan terpesat di Indonesia. Tetapi dalam 3 tahun terakhir ini, kondisi ekonomi semakin memburuk. Hal ini dikarenakan banyak perusahaan yang tutup pada 3 tahun terakhir ini, sehingga menurunnya lowongan kerja yang tersedia di kota Batam, sehingga membuat pengangguran di kota Batam terus meningkat. Tingkat pengangguran pada 2015-2017 telah mencapai 89.500 orang.

*Entrepreneur* atau pengusaha adalah seseorang yang berani membuka suatu usaha yang dapat menciptakan lapangan pekerjaan bagi orang lain, dan berjiwa gigih yang berani mengambil segala resiko yang ada untuk membuka usaha dalam berbagai kesempatan. Berjiwa gigih berani mengambil resiko yang artinya sanggup

mengatasi atau menyelesaikan suatu masalah yang dihadapinya tanpa diluputi rasa takut atau cemas sekalipun.

Pemikiran dan sikap pengusaha selalu berusaha untuk mencari peluang usaha yang memberikan keuntungan bagi mereka sendiri. pengusaha harus memiliki kemampuan berpikir yang kreatif dan inovatif dalam menemukan atau menciptakan berbagai ide yang bermanfaat bagi usaha tersebut. Selain itu sikap juga menentukan sukses atau tidaknya seorang pengusaha. Sikap-sikap yang akan terlihat pada pengusaha sukses antara lain mempunyai kejujuran dalam bertindak dan bersikap, rajin, tepat waktu, tidak bermalasan, sopan santun, menghormati pendapat orang lain, pandai bergaul, fleksibel, serius dan memiliki rasa tanggung jawab yang besar.

Hampir semua orang ingin menjadi *entrepreneur* yang sukses, tetapi tidak semua orang punya jiwa atau minat untuk menjadi seorang *entrepreneur*. Untuk membangun sebuah usaha sendiri, sebagai seorang *entrepreneur* harus mempunyai jiwa atau minat dalam wirausaha. Bahkan untuk menjadi seorang *entrepreneur* yang sukses, memiliki bakat saja tidak cukup, tetapi juga perlu memiliki niat atau pengetahuan tentang segala sudut pandang usaha yang akan ditekuninya. Tanggung jawab dari seorang wirausaha sangatlah banyak, antara lain mengambil keputusan, kepemimpinan teknis, kepemimpinan organisatoris dan komersial, penyediaan modal dan lain-lain. Dalam hal pendidikan kewirausahaan, Indonesia tertinggal dengan negara lain. Bahkan pendidikan kewirausahaan sudah dilakukan di beberapa negara pada puluhan tahun yang lalu. Dalam hal pendidikan

kewirausahaan sangat penting punya keberanian untuk memulai wirausaha. Biasanya kendala kita untuk memulai.

Wirausaha adalah rasa takut akan rugi terhadap usaha tersebut. Namun sebagian orang sudah mempunyai jiwa atau minat dalam wirausaha, tapi bingung dari mana wirausaha tersebut dimulai.

Kecerdasan Buatan atau *Artificial Intelligence* adalah salah satu bidang yang menerapkan suatu keahlian khusus dari seseorang atau seorang pakar dengan memberikan suatu perintah kedalam Komputer untuk menghasilkan suatu *output* atau *program* yang dianggap cerdas.

Kecerdasan buatan atau *Artificial Intelligence* bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Bidang ini telah berkembang sangat pesat di 20 tahun terakhir seiring dengan pertumbuhan kebutuhan akan perangkat cerdas pada industri dan rumah tangga. Dengan kemajuan teknologi saat ini dapat membantu setiap individu bekerja dengan efisien. Sistem pakar merupakan salah satu bagian dalam kecerdasan buatan yang mengandung berbagai pengetahuan dan pengalaman yang di masukkan oleh berbagai pakar di dalam berbagai bidang ke dalam suatu sistem, sehingga semua individu bisa menggunakannya untuk mencari solusi.

Sebelum penelitian di lakukan, peneliti membaca dan mempelajari beberapa hasil penelitian lain yang berhubungan dengan penelitian ini, sehingga dapat mempermudah proses penyelesaian penelitian ini.



Terkadang seseorang kurang memahami atau kurang mengetahui jiwa atau minatnya di bidang wirausaha. Maka pada penelitian ini, aplikasi sistem pakar ini mampu menentukan jiwa atau minat *entrepreneurship* seseorang. Hal inilah yang mendorong peneliti mengambil judul “**SISTEM PAKAR MENENTUKAN JIWA ENTREPRENEURSHIP**”.

### **1.2. Identifikasi Masalah**

Berdasarkan uraian di atas, maka dapat di identifikasikan permasalahan sebagai berikut.

1. Masih banyak individu belum mengetahui apakah mereka mempunyai jiwa *entrepreneurship*.
2. Kondisi ekonomi yang semakin lesu sehingga butuh inisiatif untuk membantu pemerintah menciptakan/memperbanyak *entrepreneur* dikota batam.
3. Masih banyaknya pengangguran di kota Batam karena ekonomi yang sedang melemah.

### **1.3. Batasan Masalah**

Agar penelitian ini dapat dilakukan secara lebih mendalam, akurat, dan tidak menyimpang dari apa yang telah dirumuskan, maka penelitian yang diangkat perlu dibatasi.

1. Penggunaan metode pada sistem pakar ini adalah metode *forward chaining*.
2. Aplikasi sistem pakar ini di rancang dengan bahasa pemograman PHP dan *database* MySQL berbasis *web*.

3. Data tersebut di ambil dengan melakukan wawancara dengan dokter psikolog di kota batam.
4. Objek penelitian ini akan berfokus pada karyawan di daerah Citra Buana *Centre Park*.

#### **1.4. Rumusan masalah**

Berdasarkan identifikasi masalah di atas, maka dapat di rumuskan permasalahan yang dapat pada penelitian ini, yaitu:

1. Bagaimana merancang dan membuat sistem pakar menentukan jiwa *entrepreneuership* dengan menggunakan metode *forward chaining*?
2. Bagaimana metode *forward chaining* dan bahasa pemograman PHP dapat menentukan jiwa *entrepreneuership*.

#### **1.5. Tujuan Penelitian**

Berdasarkan rumusan masalah diatas, maka pada penelitian ini tujuan yang ingin di capai adalah

1. Untuk merancang sistem pakar dengan menggunakan bahasa pemograman PHP berbasis *web* yang efektif agar penggunan bisa mengetahui jiwa *entrepreneuership* mereka sendiri.
2. Untuk juga membantu meningkatkan jumlah *entrepreneur* yang sudah ada.

## **1.6. Manfaat Penelitian**

Berdasarkan penelitian ini, dapat di harapkan memberikan manfaat bagi peneliti sendiri dan objek yang di teliti.

### **1.6.1. Manfaat Teoritis**

1. Diharapkan bisa memberi masukan dan bahan bacaan bagi kepada mahasiswa yang ingin menyempurnakan penelitian mereka.
2. Untuk mengetahui secara dalam tentang sistem pakar dengan menggunakan metode *forward chaining* berbasis *web* untuk menentukan jiwa *entrepreneurship*.

### **1.6.2. Manfaat Praktis**

1. Hasil penelitian ini di harapkan bisa menjadi referensi untuk penelitian lainnya.
2. Diharapkan hasil penelitian ini bisa di jadikan media informasi untuk masyarakat dan menambah wawasan tentang jiwa *entrepreneurship*.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Teori Dasar**

##### **2.1.1. Kecerdasan Buatan (Artificial Intelligence)**

Menurut (Budiharto & Suhartono, 2014:2) kecerdasan buatan atau *Artificial Intelligence* (AI) merupakan bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Bidang ini telah berkembang sangat pesat di 20 tahun terakhir seiring dengan pertumbuhan kebutuhan akan perangkat cerdas pada industri dan rumah tangga. Buku ini akan memaparkan berbagai pandangan modern dan hasil riset terkini yang perlu dikuasai oleh para akademis, mahasiswa, pelajar, dan praktisi.

Tujuan dari kecerdasan buatan adalah membuat mesin menjadi pintar (tujuan utama), memahami apa itu kecerdasan (tujuan ilmiah) dan membuat mesin lebih bermanfaat (tujuan entrepreneurial) (Sutojo, Mulyanto, & Suhartono, 2011:3). Berdasarkan definisi ini, maka kecerdasan buatan menawarkan media maupun uji teori tentang kecerdasan. Teori-teori ini nantinya dapat dinyatakan dalam bahasa pemrograman dan eksekusinya dapat dibuktikan pada komputer nyata.

Menurut Turban dalam (Sutojo et al., 2011:10.11) jika dibanding dengan kecerdasan buatan memiliki keuntungan komersial, antara lain:

1. Kecerdasan buatan lebih bersifat permanen. Kecerdasan alami akan cepat mengalami perubahan. Kemampuan kecerdasan buatan tidak akan pernah berubah selama programnya tidak diubah oleh *programmer*. Berbeda dengan kecerdasan alami. Karena sifat manusia yang subjektif, pelupa dan makin lama makin tua hingga kemampuan berpikirnya berkurang seiring bertambahnya waktu, kemampuan kecerdasan alami cenderung tidak permanen.
2. Kecerdasan buatan lebih mudah diduplikasi dan disebar. Misalnya saja pemerintah membutuhkan 10.000 orang pakar penyakit jantung untuk ditempatkan di seluruh Indonesia. Bayangkan kalau pemerintah harus menyekolahkan anak bangsa sejumlah 10.000 orang, mulai dari SD sampai lulus sarjana kedokteran spesialis penyakit jantung. Waktu yang dibutuhkan minimal 20 tahun. Jika biaya pendidikan 1 orang Rp100 juta, maka untuk 10.000 orang biaya yang diperlukan adalah Rp1 triliun. Belum lagi kendala jika orangnya meninggal sebelum menjadi sarjana. Kalau sudah jadi sarjana, pemerintah akan kesulitan untuk mendistribusikan mereka ke pelosok-pelosok Indonesia. Hal ini sangat tidak efisien, Sementara itu, untuk kecerdasan buatan, pemerintah cukup membuat 1 sistem pakar penyakit jantung dengan waktu yang relatif lebih cepat dan biaya yang jauh lebih murah. Proses duplikasi dan pendistribusiannya ke seluruh pelosok tanah air pun sangat mudah.
3. Kecerdasan buatan lebih murah dibanding kecerdasan alami. Hal ini tergambar seperti dijelaskan dalam nomor 2.

4. Kecerdasan buatan bersifat konsisten. Misalnya saja telah dibuat sistem pakar hakim pengadilan untuk mengadili kasus-kasus pidana di Indonesia. Untuk kasus yang sama solusi dan keputusan yang dibuat oleh kecerdasan buatan tidak pernah berubah. Berbeda dengan kecerdasan alami. Arti kata hakim bisa berubah menjadi “Hubungi Aku Kalau Ingin Menang”. Untuk kasus yang sama, solusi dan keputusan yang dibuat oleh kecerdasan alami bisa berubah-ubah tergantung orang yang terkena kasus telah menghubungi hakim atau tidak.
5. Kecerdasan buatan dapat didokumentasi. Solusi dan keputusan yang dibuat oleh kecerdasan buatan dapat didokumentasi dengan mudah karena disimpan dalam *hard disk* dan pencarian datanya relatif lebih mudah dilacak. Sedangkan untuk kecerdasan alami, hal ini sangat sulit dilakukan.
6. Kecerdasan buatan dapat mengerjakan pekerjaan lebih cepat dibanding dengan kecerdasan alami. Tentu saja karena kecepatan berpikir dari sebuah prosesor jauh lebih cepat dibanding kecepatan berpikir dari otak manusia.
7. Kecerdasan buatan dapat mengerjakan pekerjaan lebih baik dibanding dengan kecerdasan alami.  
  
Sementara itu, kecerdasan alami memberikan keuntungan sebagai berikut.
1. Kreatif. Pengetahuan seorang manusia selalu bertambah seiring dengan perkembangan waktu. Sifat bosan manusia pun mengakibatkan ia harus berpikir kreatif untuk mencari solusi-solusi terbaru. Berbeda dengan kecerdasan buatan, penambahan pengetahuan harus dilakukan pada sistem yang telah dibangun.

2. Kecerdasan alami memungkinkan orang menggunakan pengalaman secara langsung. Sedangkan pada kecerdasan buatan harus bekerja dengan input-input simbolik.
3. Pemikiran manusia dapat digunakan secara luas, sedangkan kecerdasan buatan sangat terbatas.

### **2.1.2. Sistem Pakar (*Expert System*)**

Menurut (Budiharto & Suhartono, 2014:132.133) sistem pakar adalah program komputer yang menyimulasi penilaian dan perilaku manusia atau organisasi yang memiliki pengetahuan dan pengalaman ahli dalam bidang tertentu. Biasanya, sistem seperti ini berisi basis pengetahuan yang berisi akumulasi pengalaman dan satu set aturan untuk menerapkan pengetahuan dasar untuk setiap situasi tertentu. Sistem pakar yang canggih dapat ditingkatkan dengan penambahan basis pengetahuan atau set aturan. Di antara banyak sistem pakar yang ada, yang terkenal adalah aplikasi bermain catur dan sistem diagnosis medis. Definisi sistem pakar yang paling dikenal adalah:

- a. Menurut Durkin : Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.
- b. Menurut Ignizio : Sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.

- c. Menurut Giarratano dan Riley: Sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar

#### **2.1.2.1. Konsep Sistem Pakar**

Sistem pakar memiliki 6 konsep dasar yaitu (Sutojo, 2011: 163):

1. Kepakaran (*Expertise*)

Kepakaran merupakan suatu pengetahuan yang di peroleh dari pelatihan, membaca, dan pengalaman. Kepakaran inilah yang memungkinkan para ahli dapat mengambil keputusan lebih cepat dari dapat mengambil keputusan lebih cepat daripada seorang yang bukan pakar.

2. Pakar (*Expert*)

Pakar adalah seseorang yang mempunyai pengetahuan, pengalaman, dan metode khusus serta mampu menerapkannya untuk memecahkan masalah atau memberi nasehat.

3. Pemindahan Kepakaran (*Transferring Expertise*)

Tujuan sistem pakar adalah memindahkan kepakaran dari seorang pakar ke dalam komputer, kemudian di transfer kepada orang lain yang bukan pakar.

4. Inferensi (*Inferencing*)

Inferensi adalah sebuah prosedur (*program*) yang mempunyai kemampuan dalam melakukan penalaran. Inferensi di tampilkan pada suatu komponen di sebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah.



#### 5. Aturan-aturan (*Rule*)

Kebanyakan *software* sistem pakar komersial adalah sistem berbasis *rule* (*rule-based system*), yaitu pengetahuan disimpan terutama dalam bentuk *rule*, sebagai prosedur-prosedur pemecahan masalah

#### 6. Kemampuan menjelaskan (*Explanation Capability*)

Fasilitas lain dari sistem pakar adalah kemampuannya untuk menjelaskan saran atau rekomendasi yang berikannya. Penjelasan ini dilakukan dalam subsistem yang disebut subsistem penjelasan

### 2.1.2.2. Keuntungan Sistem Pakar

Secara garis besar, banyak manfaat yang dapat diambil dengan adanya sistem pakar, antara lain (Kusumadewi, 2003:110):

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.
2. Bisa melakukan proses secara berulang secara otomatis.
3. Menyimpan pengetahuan dan keahlian pakar.
4. Meningkatkan output dan produktivitas.
5. Meningkatkan kualitas.
6. Mampu mengambil keputusan dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka).
7. Mampu beroperasi dalam lingkungan yang berbahaya.
8. Memiliki kemampuan untuk mengakses pengetahuan.
9. Memiliki reliabilitas.
10. Meningkatkan kapabilitas sistem komputer.

11. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian
12. Sebagai media pelengkap dalam pelatihan

### **2.1.2.3. Kelemahan Sistem**

Di samping memiliki beberapa keuntungan, sistem pakar juga memiliki beberapa kelemahan, antara lain (Kusumadewi, 2003:111):

1. Biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal
2. Sulit di kembangkan. Hal ini tentu saja erat kaitannya dengan ketersediaan pakar di bidangnya
3. Sistem pakar tidak 100% bernilai benar

### **2.1.2.4. Bentuk Sistem Pakar**

Ada 4 bentuk sistem pakar, yaitu (Kusumadewi, 2003:113);

1. Berdiri sendiri. Sistem pakar jenis ini merupakan *software* yang berdiri-sendiri tidak tergabung dengan *software* yang lainnya. Semua contoh sistem pakar pada Tabel 2.1 merupakan sistem pakar jenis ini.
2. Tergabung. Sistem pakar jenis ini merupakan bagian program yang terkandung di dalam suatu algoritma (konvensional), atau merupakan program dimana di dalamnya memanggil algoritma subrutin lain (konvensional)
3. Menghubungkan ke *software* Lain, bentuk ini biasanya merupakan sistem pakar yang menghubungkan ke suatu paket program tertentu, misalnya dengan DBMS

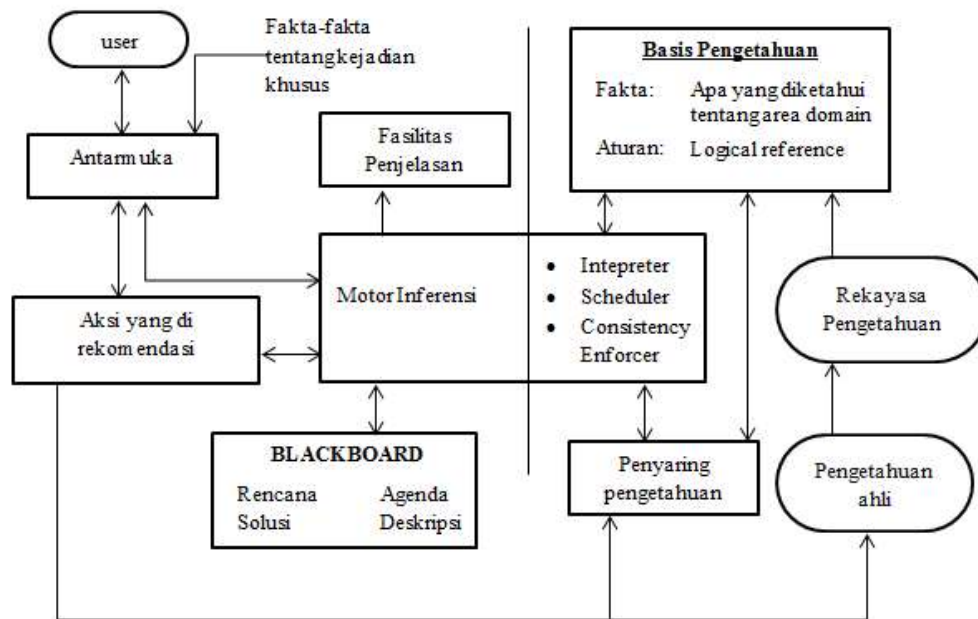
4. Sistem Mengabdi. Sistem pakar merupakan bagian dari komputer khusus yang di hubungkan dengan suatu fungsi tertentu. Misalnya sistem pakar yang di gunakan untuk membantu menganalisis data radar

**Tabel 2.1** Sistem Pakar yang terkenal

Sistem Pakar	Kegunaan
MYCIN	Diagnosa Penyakit
DENDRAL	Mengidentifikasi struktur molekular campuran yang tak di kenal
SOPHIE	Analisis sirkut elektronik
Prospector	Digunakan dalam geologi untuk membantu mencari dan menemukan deposit
XCON & XSEL	Membantu konfigurasi sistem komputer besar
FOLIO	Membantu memberikan kuputusan bagi seorang manajer dalam hal stok broker dan investasi
DELTA	Pemeliharaan lokomotif listrik disel

#### 2.1.2.5. Struktur Sistem Pakar

Struktur sistem pakar terdiri dari 2 bagian pokok, yaitu: lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan di gunaka sebagai pembangun sistem pakar baik dari segi pembangun komponen maupun basis pengetahuan. Lingkungan konsultasi di gunakan oleh seorang yang bukan ahli konsultasi (Kusumadewi, 2003:113).



**Gambar 2.1** Struktur Sistem Pakar

Komponen-komponen yang ada pada sistem pakar adalah gambar 2.1 (Kusumadewi, 2003:113):

1. Subsistem penambahan pengetahuan. Bagian ini di gunakan untuk memasukan pengetahuan, mengkonstruksi atau memperluas pengetahuan dalam basis pengetahuan. Pengetahuan itu berasal dari ahli, buku, basis data, penelitian dan gambar.
2. Basis pengetahuan. Berisi pengetahuan-pengetahuan yang di butuhkan untuk memahami, memformulasikan dan menyelesaikan masalah.
3. Motor inferensi (*inference engine*). Program yang berisi metodologi yang di gunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan dan *blackboard*, serta di gunakan untuk memformulasikan konklusi, ada 3 elemen utama dalam motor inferensi, yaitu:

- *Intepreter*: mengeksekusi item-item agenda yang terpilih dengan menggunakan aturan-aturan dalam basis pengetahuan yang sesuai
  - *Scheduler*: akan mengontrol agenda
  - *Consistency enforcer*: akan berusaha memelihara kekonsistenan dalam mempresentasikan solusi yang bersifat darurat
4. *Blackboard*. Merupakan area dalam memori yang di gunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada 3 tipe keputusan yang dapat di rekam, yaitu
- Rencana: bagaimana menghadapi masalah
  - Agenda: aksi-aksi yang potensial yang sedang menunggu di eksekusi
  - Solusi: calon aksi yang akan di bangkitkan
5. Antarmuka, di gunakan untuk media komunikasi antara user dan program
6. Subsistem penjelasan. Di gunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pertanyaan:
- Mengapa suatu pertanyaan di tanyakan oleh sistem pakar?
  - Bagaimana konklusi di capai?
  - Mengapa ada alternatif yang di batalkan?
  - Rencana apa yang di gunakan untuk mendapatkan solusi?
7. Sistem penyaring pengetahuan. Sistem ini di gunakan untuk mengevaluasi kinerja sistem pakar itu sendiri untuk melihat apakah pengetahuan-pengetahuan yang ada masih cocok untuk di gunakan di masa mendatang.

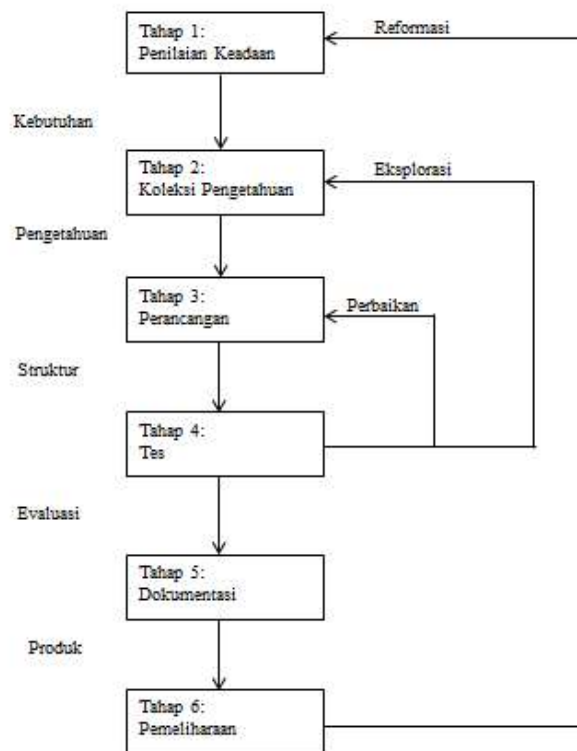
### 2.1.2.6. Mengembangkan Sistem pakar

Secara garis besar pengembangan sistem pakar pada gambar adalah sebagai berikut (Kusumadewi, 2003:122):

1. Mengidentifikasi masalah dan kebutuhan. Mengkaji situasi dan memutuskan dengan kepastian tentang masalah yang akan di komputerisasi dan apakah dengan sistem pakar bisa lebih membantu atau tidak.
2. Menentukan masalah yang cocok, ada beberapa syarat yang harus di penuhi agar sistem pakar dapat bekerja dengan baik.
3. Mempertimbangkan alternatif. Dalam hal ini ada 2 alternatif yaitu menggunakan sistem pakar atau komputer tradisional.
4. Menghitung pengembalian investasi, termasuk diantaranya: biaya pembuatan sistem pakar, dan biaya training.
5. Memilih alat pengembangan, bisa di gunaka *software* pembuat sistem pakar atau dirancang dengan bahasa pemograman sendiri.
6. Rekayasa Pengetahuan. Perlu di lakukan dengan penyempurnaan terhadap aturan-aturan yang sesuai.
7. Merancang sistem. Bagian ini termasuk pembuatan *protoype*, serta menterjemahkan pengetahuan menjadi aturan-aturan.
8. Melengkapi pengembangan, termasuk pengembangan *protoype* apabila sistem yang telah ada sudah sesuai dengan keinginan.
9. Menguji dan mencari kesalahan sistem.
10. Memelihara sistem. Dalam hal ini harus di lakukan: memperbaharui pengetahuan, mengganti pengetahuan, mengganti pengetahuan yang sudah

ketinggalan, dan meluweskan sistem agar bisa lebih baik lagi dalam menyelesaikan masalah.

Seperti dari penjelasan pengembangan perangkat lunak di atas, terdapat gambar tahap-tahap pengembangan sistem pakar seperti gambar 2.2



**Gambar 2.2** Tahap-Tahap Pengembangan sistem pakar

### **2.1.2.7. Ciri-Ciri Sistem Pakar**

Menurut (Sutojo et al., 2011:162) Ciri-ciri sistem pakar adalah sebagai berikut.

1. Terbatas pada dominan keahlian tertentu.
2. ada dominan keahlian tertentu.
3. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau pasti.
4. Dapat menjelaskan alasan – alasan dengan cara yang dapat di pahami.
5. Bekerja berdasarkan kaidah/ rule tertentu.
6. Mudah di modifikasi.
7. Basis pengetahuan dan mekanisme inferensi terpisah
8. Keluarannya bersifat anjuran.
9. Sistem dapat mengaktifkan kaidah
10. secara searah yang sesuai, di tuntun oleh dialog dengan pengguna.

### **2.1.2.8. Teknik inferensi *foward chaining* dan *backward chaining***

Menurut (Sutojo et al., 2011:171) Pada sistem pakar berbasis rule, domain pengetahuan di representasikan dalam sebuah kumpulan rule berbentuk *IF-THEN*, sedangkan data direpresentasikan dalam sebuah kumpulan fakta-fakta tentang kejadian saat ini. Mesin inferensi membandingkan masing-masing rule yang tersimpan dalam basis pengetahuan dengan fakta-fakta yang terdapat dalam database. Jika bagian *IF* (kondisi) dari *rule* cocok dengan fakta, maka rule dieksekusi dan bagian *THEN* (aksi) di letakkan oleh dalam database sebagai fakta baru yang di tambahkan.



### 1. *Forward Chaining*

*Forward chaining* adalah teknik pencarian yang di mulai dengan fakta yang di ketahui. kemudian mencocokkan fakta-fakta tersebut dengan bagian *if* dari *rule IF-THEN*. Bila ada fakta yang cocok dengan bagian *if*, maka *rule* tersebut di eksekusi. Bila sebuah *rule* di eksekusi, maka sebuah fakta baru (bagian *THEN*) di tambahkan ke dalam database. Setiap kali pencocokan, di mulai dari *rule* teratas. Setiap *rule* hanya boleh dieksekusi sekali saja, proses pencocokan berhenti bila tidak ada lagi *rule* yang bisa di eksekusi. Metode pencarian yang di gunakan adalah *Depth-First Search (DFS)*, *Breadth-First Search (BFS)* atau *Best First Search*.

### 2. *Backward Chaining*

*Backward chaining* adalah metode inferensi yang bekerja mundur ke arah kondisi awal. Proses diawali dari Goal (yang berada di bagian *THEN* dari *rule IF-THEN*), kemudian pencarian mulai di jalankan untuk mencocokkan apakah fakta-fakta yang ada cocok dengan premis-premis di bagian *IF*. Jika cocok, *rule* di eksekusi, kemudian di hipotesis di bagian *THEN* di tempatkan di basis data sebagai fakta baru. Jika tidak cocok, simpan premis di bagian *IF* ke dalam stack sebagai subGoal. Proses berakhir jika Goal di temukan atau tidak ada *rule* yang bisa membuktikan kebenaran dari subGoal atau Goal.

### **2.1.2.9. Strategi Penyelesaian Konflik**

Menurut (Sutojo et al., 2011:186) Apabila ada suatu fakta yang dapat memicu lebih dari satu rule, maka ada kemungkinan akan terjadi konflik dalam memori kerja. Artinya rule mana yang harus di pilih oleh sistem pakar. Sistem pakar tidak bisa memilih semua rule sekaligus. Ia harus memutuskan untuk memilih satu rule. Dalam melakukan pemilihan, sistem pakar menggunakan cara-cara untuk memilih rule-rule yang akan di terapkan apabila terdapat lebih dari satu rule yang cocok dengan fakta yang terdapat dalam memori kerja, diantaranya adalah:

1. *No Duplication*. Tidak boleh memicu sebuah rule dua kali menggunakan fakta yang sama agar tidak ada fakta yang di tambahkan kedalam memori kerja lebih dari sekali
2. *Regency*. Fakta yang paling baru dalam memori kerja harus di gunakan dalam memilih *rule*.
3. *Specificity*. Fakta yang lebih spesifik (khusus) harus di gunakan dalam memicu *rule*.
4. *Operation priority*. Pilih aturan dengan prioritas yang lebih tinggi.

### **2.1.3. Jaringan Syaraf Tiruan**

Menurut (Sutojo, 2011:283) Jaringan Syaraf Tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem saraf secara biologis, Seperti Proses Informasi pada otak manusia. Cara kerja sistem saraf tiruan seperti cara kerja manusia yaitu belajar melalui contoh.

Kelebihan-kelebihan yang di berikan JST antara lain:

- a) *Belajar Adaptive*: Kemampuan untuk mempelajari bagaimana melakukan pekerjaan berdasarkan data yang di berikan untuk pelatihan atau pengalaman awal.
- b) *Self-Organisation*: sebuah JST dapat membuat organisasi sendiri atau representasi dari informasi yang di terimanya selama waktu belajar
- c) *Real Time Protection*: Perhitungan JST dapat di lakukan secara paralel sehingga perangkat keras yang di rancang dan di produksi secara khusus dapat mengambil keuntungan dari kemampuan ini.

Selain mempunyai kelebihan-kelebihan tersebut, JST juga mempunyai kelemahan-kelemahan berikut.

- a) Tidak efektif jika digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi.
- b) Tidak efisien jika digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi.
- c) Untuk beroperasi JST butuh pelatihan sehingga bila jumlah datanya besar, waktu yang di gunakan untuk proses pelatihan sangat lama.

#### **2.1.4. Fuzzy Logic**

Menurut (Sutojo et al., 2011:211) logika *fuzzy* adalah metodologi sistem kontrol pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan PC, *multi-channel*, atau *workstation* berbasis akuisisi data, dan sistem kontrol.

Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak, atau kombinasi keduanya. Dalam logika klasik dinyatakan bahwa segala sesuatu bersifat biner, yang artinya adalah hanya mempunyai dua kemungkinan, “Ya atau Tidak”, “Benar atau Salah”, “Baik atau Buruk”, dan lain-lain.

Ada 3 metode dalam logika *fuzzy*, yaitu:

1. Metode Tsukamoto

Menurut (Kusumadewi, 2013) pada Metode Tsukamoto, Setiap Konsukuen pada aturan yang berbentuk *IF-Then* harus di presentasikan dengan suatu himpunan fuzzy dengan fungsi keanggotaan yang menonton Sebagai hasilnya, output hasil referensi dari tiap-tiap aturan yang di berikan secara tegas berdasarkan  $\alpha$ -predikat. Hasil akhirnya di peroleh dengan menggunakan rata-rata terbobot

2. Metode Mamdani

Menurut (Kusumadewi, 2013) Metode Mamdani sering di kenal sebagai *Max-Min*. Metode ini di perkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan output, di perlukan 4 tahapan:

- a) Pembentukan himpunan *fuzzy*

Pada metode mamdani, baik variabel *input* maupun variabel *output* di bagi menjadi satu atau lebih himpunan *fuzzy*.

- b) Aplikasi fungsi implikasi

Pada metode mamdani, fungsi implikasi yang di gunakan adalah Min.

- c) Komposisi aturan

Tidak seperti penalaran menonton, apabila sistem terdiri dari beberapa aturan, maka inferensi di peroleh dari kumpulan dan kolerasi antar aturan.

Ada 3 metode yang di gunakan dalam melakukan inferensi sisem fuzzy, yaitu: *max additive* dan probabilistik OR (probor)

d) Penegasan (*defuzzy*)

*Input* dari proses defuzzifikasi adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga diberikan suatu *fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai crisp tertentu sebagai *output*.

3. Metode Sugeno

Menurut (Kusumadewi, 2013) penalaran metode SUGENO hampir sama dengan penalaran MAMDANI, hanya saja *output* (Konsukuen) sistem tidak dapat berupa himpunan *fuzzy*; melainkan berupa konstanta atau persamaan linear. Metode ini di perkenalkan oleh Takagi-Sugeno Kang pada tahun 1985, sehinggann metode ini sering juga dinamakan metode TSK.

## 2.2. SIFAT

Pada hakikatnya pola pikir seorang entrepreneur hendaknya di mulai dengan berpikir tentang keinginan menjadi pribadi yang sukses. Yakni hakikat entrepreneur adalah orang-orang yang memiliki kemampuan dalam melihat dan menilai kesempatan-kesempatan bisnis, mengumpulkan sumber daya - sumber daya yang di butuhkan untuk mengambil tindakan yang tepat, mengambil keuntungan serta memiliki ciri, sifat, watak dan kemauan untuk mewujudkan gagasan inovatif kedalam dunia nyata secara kreatif dalam rangka sukses meningkatkan pendapatan. (Tontowi, 2016:11)

Berikut adalah sifat-sifat entrepreneur yang di katakan sebagai sukses:

1. Percaya diri

Sifat utama dari percaya diri dimulai dari pribadi yang mantap, tidak mudah terombang-ambing oleh pendapat dan saran orang lain melainkan menggunakan sebagian saran tersebut sebagai masukan.

2. Berorientasi pada tugas dan hasil

Sifat seorang entrepreneur tidak mengutamakan *prestige* dahulu melainkan fokus kepada prestasi yang ingin di capai

3. Pengambilan resiko.

Ciri pengambilan resiko berpengaruh penting dalam dunia wirausaha yang penuh resiko dan tantangan. Hal penting yang harus diperhatikan adalah bahwa bagaimana seorang *entrepreneur* mengambil sebuah risiko penuh pertimbangan

4. kepemimpinan.

Dalam diri seorang *entrepreneur* mutlak memiliki jiwa kepemimpinan. Seorang pemimpin yang baik harus mendengar saran dan kritik dari bawahnya demi kemajuan kinerja perusahaan.

5. Kreativitas

Seorang entrepreneur tidak hanya mengekor pada orang lain, tetapi memiliki pendapat sendiri, ide yang orisinal dan mampu merealisasikan ide tersebut.

6. Berorientasi pada masa depan

Seorang entrepreneur harus perspektif, mempunyai visi ke depan. Sebab sebuah usaha bukan didirikan untuk sementara tetapi untuk selamanya. Untuk

menyiapkan visi yang jauh ke depan, entrepreneur perlu menyusun perencanaan dan strategi yang matang

#### 7. Jujur dan tekun

Kejujuran merupakan landasan moral yang terkadang dilupakan oleh seorang wirausahawan. Kejujuran dalam berperilaku bersifat kompleks

### 2.3. Software Pendukung

#### 2.3.1 UML (*Unified Modeling Language*)

*“The UML defines a diagrammatic notation for describing the artefacts of an OOAD. Through the UML we can visualize, specify, construct and document our software application”*. (Noni Rahmawati; Herry Mulyono, 2016:106-107)

UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membantu analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. Banyak orang telah membuat bahasa pemodelan pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan banyak pihak adalah *Data Flow Diagram* (DFD) untuk memodelkan perangkat lunak yang menggunakan pemrograman procedural atau structural, kemudian juga ada *State Transition Diagram* (STD) yang digunakan untuk memodelkan sistem *real time* (waktu nyata).

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan model

visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.(Rosa A.S; M.Shalahuddin, 2011 :113).

### 2.3.1.1. Use Case Diagram

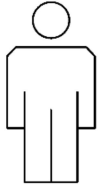



*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Ada duahal utama pada *use case* yaitu pendefinisian apa yang disebut *actor* dan *use case*.(Rosa A.S; M.Shalahuddin, 2011)

1. *Actor* merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari *actor* adalah gambar orang, tapi *actor* belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau *actor*.

**Tabel 2.2** Simbol-simbol pada *Use Case*

Simbol	Deskripsi
Use Case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor,



	biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama <i>use case</i>
<p>Aktor / <i>actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
<p>Asosiasi / <i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
<p>Ekstensi / <i>extend</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.
<p>Generalisasi / <i>generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
<p>Menggunakan / <i>include</i></p>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini

--      -->	untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.
-------------	---

Sumber: (Rosa A.S; M.Shalahuddin, 2011)

### 2.3.1.2. *Activity Diagram*


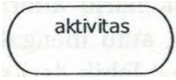
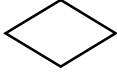


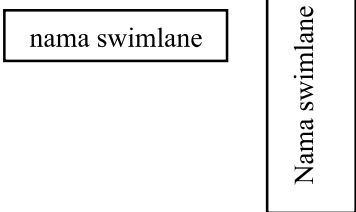
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan *actor*, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut: (Rosa A.S; M.Shalahuddin, 2011)

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*userinterface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

**Tabel 2.3** Simbol-simbol pada *activity diagram*

Simbol	Deskripsi

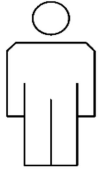



<p>Status awal</p> 	<p>Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal</p>
<p>Aktivitas</p> 	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja</p>
<p>Percabangan / <i>decision</i></p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu</p>
<p>Penggabungan / <i>join</i></p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu</p>
<p>Status akhir</p> 	<p>Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir</p>
<p>Swimlane</p> 	<p>Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p>





Sumber: (Rosa A.S; M.Shalahuddin, 2011)

### 2.3.1.3. Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antara objek. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. (Rosa A.S; M.Shalahuddin, 2011)

**Tabel 2.4** Simbol-simbol pada *Sequence diagram*

Simbol	Deskripsi
Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. Biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
Garis hidup / <i>lifeline</i> 	Menyatakan kehidupan suatu objek.
Objek 	Menyatakan objek yang berinteraksi pesan.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.

Pesan tipe <i>create</i> <<create>> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe <i>call</i> 1 : nama_metode() 	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.
Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarahkan pada objek yang dikirim.
Pesan tipe <i>return</i>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan sesuatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe <i>destroy</i> <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

Sumber: (Rosa A.S; M.Shalahuddin, 2011)

#### 2.3.1.4. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut: (Rosa A.S; M.Shalahuddin, 2011)

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

2. Kelas yang menangani tampilan sistem (*View*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

3. Kelas yang diambil dari pendefinisian *use case* (*controller*)

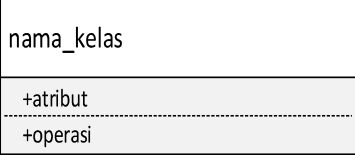


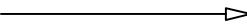
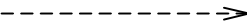
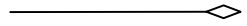
Kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada diagram kelas:

Tabel 2.5 Simbol-simbol pada *class diagram*

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka / <i>interface</i></p> 	Sama dengan konsep interface dalam pemrograman berorientasi objek
<p>Asosiasi berarah / <i>direct association</i></p> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
<p>Kebergantungan / <i>dependency</i></p> 	Relasi antar kelas dengan makna kebergantungan kelas
<p>Agresi / <i>aggregation</i></p> 	Relasi antar kelas dengan makna semua-bagian ( <i>whole-part</i> )

Sumber: (Rosa A.S; M.Shalahuddin, 2011)

### **2.3.2. Xampp**

*Xampp* merupakan *software* bundel yang terdiri dari kumpulan paket program yang berhubungan dengan *database server*, *web server*, dan lain sebagainya. Didalam *xampp* terdapat *Apache* sebagai *server web*, *MySQL* sebagai *server* basis data, *Filezila* sebagai *FTP server*, dan beberapa fitur tambahan seperti *Mercury* dan *Tomcat*. Aplikasi ini dapat di unduh di alamat “*apachefriends.org*”. (Angga Reza Palevi; Krisnawati, 2013:4)

*XAMPP* adalah sebuah *software web server apache* yang didalamnya sudah tersedia *database server MySQL* dan dapat mendukung pemrograman *PHP*. *XAMPP* merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di *Linux* dan *Windows*. Keuntungan lainnya adalah cuma menginstal satu kali sudah tersedia *Apache Web Server*, *MySQL Database Server*, *PHP Support (PHP 4 dan PHP 5)* dan beberapa module lainnya. (Herny Februariyanti; Eri Zuliarso, 2012:129)

#### **2.3.2.1. PHP**

*PHP stands for Hypertext Preprocessor. It is a server-side scripting language that powers some of the most popular websites in the world, including WordPress and Facebook. It is open source, relatively easy to learn, and works perfectly with MySQL, making it a popular choice for web developers. (Er. Saurabh Walia; Er. Satinderjit Kaur Gill, 2014:27)*

*PHP (Hypertext Preprocessor)* adalah bahasa *server-sidescripting* yang menyatu dengan *HTML* untuk membuat halaman web yang dinamis. Karena *PHP* merupakan *server-sidescripting* maka sintaks dan perintah-perintah *PHP* akan



dieksekusi di *server* kemudian hasilnya akan dikirim ke *browser*. (Arip Aryanto; Tri Irianto Tjendrowasono; Berliana Kusuma Riasti, 2012).

*PHP (Hypertext Preprocessor)* merupakan bahasa pemrograman berbasis *web* yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis. *PHP* dapat dikatakan sebagai sebuah *server-side embedded script language*, artinya semua sintaks dan pemerintah program yang anda tulis akan sepenuhnya dijalankan oleh *server*, tetapi dapat disertakan pada halaman HTML biasa. Pada umumnya, semua aplikasi yang dibangun menggunakan *PHP* akan memberikan hasil *web browser*, tetapi prosesnya secara seluruhan dijalankan pada *server*. (Wahana Komputer, 2009 :3)

*PHP* mempunyai beberapa kemampuan yang merupakan salah satu kelebihan *PHP*. Kemampuan tersebut antara lain: (Wahana Komputer, 2009 :10)

1. Cara koneksi dan *query database* yang sederhana.
2. Dapat bekerja pada sistem operasi berbasis *Windows, Linux, Mac OS*, dan kebanyakan varian *UNIX*.
3. Biaya yang dibutuhkan untuk menggunakan *PHP* tidak mahal, atau bahkan gratis.
4. Mudah digunakan karena memiliki fitur dan fungsi khusus untuk membuat *web* dinamis. Bahasa pemrograman *PHP* dirancang untuk dapat dimasukkan dalam *HTML (embedded script)*.
5. *Security* sistem yang cukup tinggi.

6. Waktu eksekusi yang lebih cepat dibandingkan dengan bahasa pemrograman *web* lainnya berorientasi pada *serverside scripting*.
7. Akses ke sistem *database* yang lebih fleksibel dan mudah, seperti *MySQL*.

Selain memiliki berbagai keunggulan, *PHP* juga memiliki beberapa kekurangan. Dari segi bahasa *PHP* bukanlah bahasa yang cocok untuk pengembangan berskala besar. Kekurangan yang utama adalah tidak adanya *namespace*. *Namespace* adalah sebuah cara untuk mengelompokkan fungsi atau nama variable dalam susunan hierarki. (Wahana Komputer, 2009 :12)

#### **2.3.2.2. MySQL**

*MySQL* adalah multi user database yang menggunakan bahasa *SQL* (*Structured Query Language*). *SQL* adalah bahasa standar yang digunakan untuk mengakses *database server*.

Dalam konteks bahasa *SQL*, pada umumnya informasi tersimpan dalam tabel-tabel yang secara logika merupakan struktur dua dimensi yang terdiri atas baris-baris data yang berada dalam satu atau lebih kolom. Baris pada tabel sering disebut sebagai *instance* dari data, sedangkan kolom sering disebut sebagai *attribute* atau *field*. Keseluruhan tabel tersebut dihimpun dalam satu kesatuan yang disebut *database*. (Veronica Lusiana, 2009:28)

*MySQL* (*My Structured Query Language*) atau yang biasa dibaca mai-sekuel adalah sebuah program pembuat dan pengelolah *database* atau yang sering disebut dengan *DBMS* (*Database Management System*), sifat dari *DBMS* ini adalah *Open Source*. *MySQL* sebenarnya produk yang berjalan pada *platform Linux*, dengan

adanya perkembangan dan banyaknya pengguna, serta lisensi dari database ini adalah *Open Source*, maka para pengembang kemudian merilis versi *windows*.

Selain itu *MySQL* juga merupakan program pengakses *database* yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak Pengguna). Kelebihan lain dari *MySQL* adalah menggunakan bahasa *query* (permintaan) standar *SQL* (Structures Query Language). *SQL* adalah suatu bahasa permintaan yang terstruktur, *SQL* telah distandartkan untuk semua program pengakses *database* seperti *Oracle*, *PosgreSQL*, *SQL server* dan lain-lain.

Sebuah program penghasil *database*, *MySQL* tidak mungkin berjalan sendiri tanpa adanya sebuah aplikasi pengguna (*interface*) yang berguna sebagai program aplikasi pengakses *database* yang dihasilkan. *MySQL* dapat didukung oleh hamper semua program aplikasi baik yang *Open Source* seperti *PHP* maupun yang tidak *OpenSource* yang ada pada platform *windows* seperti *visual basic*, *Delphi* dan lainnya. (Bunafit Nugroho, 2008 :91)

Pada umumnya, perintah yang paling sering digunakan dalam *MySQL* adalah *SELECT* (mengambil), *INSERT* (menambah), *UPDATE* (mengubah), dan *DELETE* (menghapus). Selain itu, *SQL* juga menyediakan perintah untuk membuat *database*, *field*, ataupun *index* untuk menambah atau menghapus data.

Ada beberapa alasan yang menjadikan *database MySQL* sangat diminati oleh para programmer, diantaranya:

1. Bersifat *open source*.

2. Menggunakan bahasa *SQL (Structure Query Language)*, yang merupakan standar bahasa dalam pengolahan data.
3. *Performance* dan *reliable*, pemrosesan database-nya sangat cepat dan stabil.
4. Sangat mudah dipelajari (*ease of use*).
5. Memiliki dukungan (*group*) pengguna *MySQL*.
6. Lintas *Platform*, dapat digunakan pada berbagai Sistem Operasi berbeda.
7. *Multiuser*, dimana *MySQL* dapat digunakan oleh banyak user dalam waktu yang bersamaan tanpa mengalami konflik.

#### **2.4. Penelitian Terdahulu**

Sebagai bahan pertimbangan dalam penelitian ini, maka penulis mencantumkan beberapa penelitian yang di ambil beberapa jurnal ilmiah, yaitu:

1. Penelitian yang di lakukan oleh (Iskandar, Sholeh, & Iswahyudi, 2015), **SISTEM PAKAR UNTUK MENENTUKAN KEPRIBADIAN SESEORANG BERDASARKAN TES PERSONALITAS FLORENCE LITTAUER BERBASIS WEB**, dalam penelitian ini dapat di simpulkan bahwa Sistem Pakar tes kepribadian dapat membantu user mendiagnosa tipe kepribadian diri sendiri, memberikan rekomendasi jenis pekerjaan yang sesuai dengan tipe kepribadian dan memberikan pengetahuan tentang kelebihan dan kelemahan dari setiap tipe kepribadian. Sistem ini dibangun untuk menyimpan pengetahuan keahlian seorang pakar psikologi khususnya psikologi kepribadian. Pembangunan sistem dirancang sedemikian rupa

sehingga dapat mengadopsi perkembangan profil kepribadian yang digunakan berbasis aturan tes personalitas *Florence* Littauer dan metode inferensi forward chaining. Implementasi sistem pakar dalam bentuk *web* sangat membantu memberikan kemudahan bagi user dalam mengaksesnya. Sistem ini juga masih sebatas simulasi dan belum dikonsultasikan terhadap ahli psikologi.

2. (Mahmudi & Fatkhudin, 2013), **PENENTUAN BANTUAN MODAL WIRAUUSAHA BARU MENGGUNAKAN METODE FUZZY INFERENSI SYSTEM (FIS) MAMDANI** Berdasarkan kegiatan selama perancangan dan implementasi pada proses pembuatan aplikasi Fuzzy Inference System Untuk Penentuan Bantuan Modal Wirausaha menggunakan Metode Mamdani maka dapat di ambil kesimpulan bahwa Aplikasi Penentuan Bantuan Modal Wirausaha yang dikembangkan dapat membantu para pengambil kebijakan dalam hal ini Dinsosnakertrans dalam memilih para pengusaha mikro yang mengajukan permohonan bantuan dengan kemungkinan hasil yang terbaik karena setiap perhitungan diperoleh dari kriteria – kriteria yang dikehendaki oleh pengambil kebijakan.
3. (Ade Setiadi, 2015), **ANALISA SISTEM PAKAR UNTUK IDENTIFIKASI KEPRIBADIAN SISWA MENGGUNAKAN ALGORITMA FUZZY PADA SISWA SLTA**, Berdasarkan hasil penelitian dan pembahasan yang penulis lakukan mengenai “ Analisa Sistem Pakar Untuk Identifikasi Kepribadian Siswa Menggunakan Algoritma Fuzzy”. Maka penulis pada akhirnya mengambil beberapa

kesimpulan dalam merepresentasikan pengetahuan dengan kaidah produksi pada sistem pakar yang diterapkan untuk mendiagnosa yaitu Dengan adanya sistem pakar ini masalah kekurangan tenaga pakar dapat terselesaikan, dengan sistem pakar ini user dapat berinteraksi dengan sistem seperti halnya berinteraksi dengan pakar.

4. (Andreyana, Piarsa, & Buana, 2015), **SISTEM PAKAR ANALISIS KEPERIBADIAN DIRI DENGAN METODE CERTAINTY FACTOR**, kesimpulan dari penelitian ini adalah Sistem Pakar Analisis Kepribadian Diri berbasis *web* menghasilkan jenis-jenis kepribadian menggunakan teori Big Five Personality. *Metode Certainty Factor* digunakan dalam Sistem Pakar Analisis Kepribadian Diri untuk mendapat nilai persentase kepercayaan pakar pada *output* sistem dengan tingkat akurasi mencapai 74% berdasarkan hasil *survey* yang dilakukan kepada 25 orang koresponden untuk mencoba dan memberi penilaian mengenai ketepatan pertanyaan-pertanyaan dan hasil dari sistem. Sistem Pakar Analisis Kepribadian Diri menghasilkan output jenis-jenis kepribadian yang didapat dari jawaban pengguna dengan memberi pertanyaan-pertanyaan mengenai sifat kepribadian. *Output* tes kepribadian yang dihasilkan sistem sesuai dengan pengetahuan yang dimiliki oleh pakar psikolog.
5. (Abu Haris & Rahman, 2017) **A STUDY ON APPLICATION OF FUZZY METHODS IN ENTREPRENEURSHIP DOMAIN** kesimpulan pada penelitian ini adalah Fuzzy methods can be versatile and flexible tool for data that are complex, vague and imprecise. Fuzzy addresses application

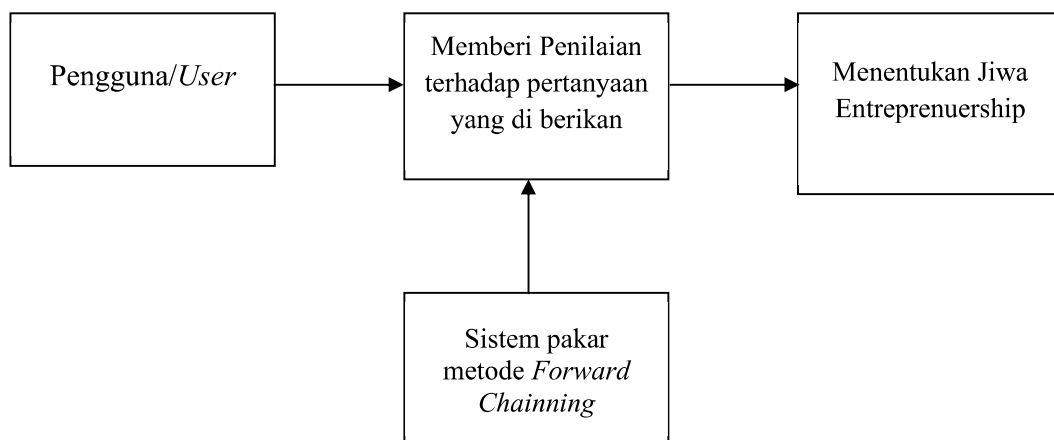
that resembles human in decision making. Fuzzy methods have the ability to generate precise solutions for certain or approximate information and the data generated through fuzzy methods has the advantage of reducing uncertainty.

6. (Al-lamki, Al-sumri, Al-ismaili, & Al-busaidi, 2016) **AN EXPERT SYSTEM TO DETERMINE THE PROBABILITY OF SUCCESS AND READINESS IN ENTREPRENEURSHIP** study presented the development of an expert system to assess the probability of success and readiness in entrepreneurship, based on personal traits. The study also indicated, based on the evaluation of 15 users, that the system adds value and provides good quality in terms of inputs, outputs, and overall system quality.

## 2.5. Kerangka Pemikiran

Kerangka pemikiran adalah merupakan konseptual mengenai bagaimana satu teori berhubungan diantara berbagai faktor yang telah di identifikasikan penting terhadap masalah penelitian. Dalam kerangka pemikiran, peneliti harus menguraikan konsep atau variabel penelitiannya secara lebih terperinci. Tidak hanya mendefinisikan variabel tadi, tetapi juga menjelaskan keterkaitan diantara varibel tadi (Noor, 2011:76).

Penjelasan sementara terhadap suatu gejala yang menjadi objek permasalahan peneliti. Kerangka berpikir ini disusun dengan berdasarkan pada tinjauan pustaka dan hasil penelitian yang relevan atau terkait.



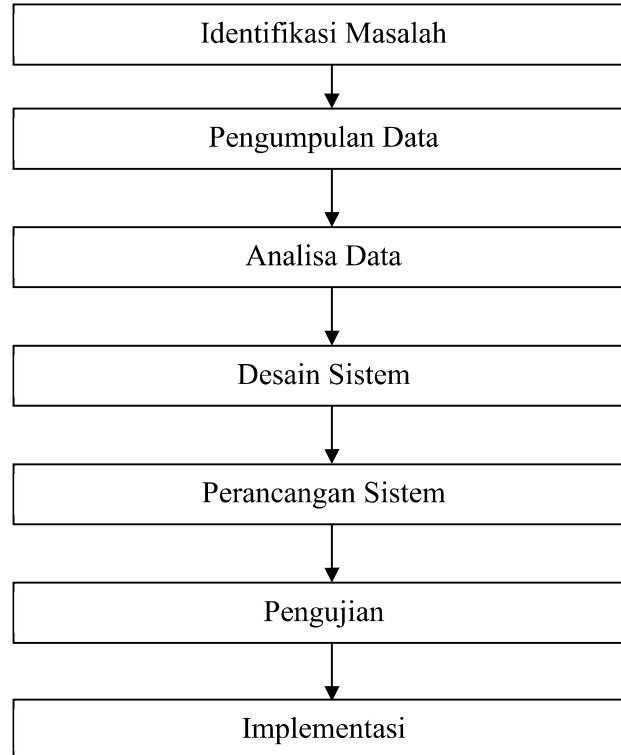
**Gambar 2.3** Kerangka pemikiran  
**Sumber:** Data Penelitian (2018)



## **BAB III METODE PENELITIAN**

### **3.1 Desain Penelitian**

Desain penelitian adalah rencana yang meliputi cara panggilan data dan cara analisi data. Umumnya pada sebuah skripsi, desain penelitian di uraikan pada bab III. Ketika membaca desain penelitian yang di sebutkan oleh peneliti, seorang pembaca akan tahu bagaimana data itu akan di analisis.



**Gambar 3.1** Desain penelitian  
**Sumber:** Data Penelitian (2018)

### Penjelasan Desain Penelitian:

#### A. Identifikasi Masalah

Tahap ini untuk mendapatkan masalah yang terdapat dalam penelitian, yaitu kepribadian manusia dalam menentukan kepribadiannya

#### B. Pengumpulan data

Pengumpulan data di lakukan dengan metode Wawancara dan Literatur. Metode studi Literatur yaitu mengumpulkan data dari buku-buku referensi yang membahas mengenai kepribadian dan sistem pakar, modul-modul yang relevan dengan objek permasalahan. Metode Wawancara yaitu mengumpulkan data dengan bertanya dan konsultasi kepada psikolog.

#### C. Analisis Data

Analisis data adalah upaya atau cara untuk mengolah data menjadi informasi sehingga karakteristik data tersebut bisa dipahami dan bermanfaat untuk solusi permasalahan, terutama masalah yang berkaitan dengan penelitian. Analisis data pada penelitian ini menggunakan metode *forward chaining*

#### D. Desain Sistem

Teknik pemecahan masalah yang saling melengkapi yang merangkai kembali bagian-bagian komponen menjadi sistem yang lengkap. Sistem ini di desain dengan menggunakan Microsoft Visio.

#### E. Perancangan sistem

perancangan sistem mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan

#### F. Pengujian

Melakukan demonstrasi sistem apakah telah memenuhi syarat yang sudah di spesifikasikan

#### G. Implementasi

Implementasi sistem merupakan tahap terakhir dari peneltian yang menjalankan resolusi dari permasalahan yang teridentifikasi dari fase pengujian

### **3.2 Pengumpulan data**

pengumpulan data adalah teknik atau cara yang dilakukan oleh peneliti untuk mengumpulkan data. Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian.

#### **3.2.1 Wawancara**

wawancara atau interview adalah suatu cara pengumpulan data yang digunakan untuk memperoleh informasi langsung dari sumbernya. Wawancara tersebut akan di lakukan dengan Dokter Psikologi.

#### **3.2.2 Metode Literatur**

Metode Literatur adalah metode pengumpulan data dengan cara membaca buku-buku dan situs-situs internet yang mendukung dan menunjang dalam pembuatan Skripsi dan penyusunan laporan, sekaligus dijadikan sebagai landasan dalam penulisan laporan Skripsi.

### 3.3 Operasional Variabel

Menurut (Sudaryono, 2015) variabel adalah segala sesuatu yang memiliki variasi nilai. Contohnya, nilai ujian bervariasi dari 0-100; tingkat motivasi bisa bervariasi dari sangat rendah hingga sangat tinggi; tingkat kepuasan konsumen bervariasi dari sangat rendah hingga sangat tinggi. Menurut (Widoyoko, 2012) dalam (Sudaryono, 2015, p. 18) dalam penelitian, variabel mempunyai tiga ciri, yaitu memiliki variasi nilai, membedakan satu objek dengan objek lain dalam satu populasi, dan dapat diukur. Karena membedakan objek-objek dalam satu populasi, variabel harus mempunyai nilai yang bervariasi. Sebagai contoh, dari populasi penduduk yang mendiami suatu wilayah, jenis pekerjaan atau profesi bukan merupakan variabel apabila seluruh penduduk memiliki pekerjaan atau profesi yang sama.

**Tabel 3.1** Operasional Variabel

Variabel	Indikator
Sifat	Kepercayaan diri
	Berorientasi pada tugas dan hasil
	Pengambilan resiko
	Kepemimpinan
	Kreativitas
	Berorientasi pada hasil
	Jujur dan tekun

**Sumber:** Data Penelitian (2018)

#### 3.3.1 Forward Chaining

*Forward chaining* adalah teknik pencarian yang di mulai dengan fakta yang di ketahui. kemudian mencocokkan fakta-fakta tersebut dengan bagian *if* dari *rule IF-THEN*. Bila ada fakta yang cocok dengan bagian *if*, maka *rule* tersebut di eksekusi. Bila sebuah *rule* di eksekusi, maka sebuah fakta baru (bagian *THEN*) di tambahkan ke dalam database. Setiap kali pencocokan, di mulai dari *rule* teratas. Setiap *rule*

hanya boleh dieksekusi sekali saja, proses pencocokan berhenti bila tidak ada lagi rule yang bisa di eksekusi. Metode pencarian yang di gunakan adalah *Depth-First Search (DFS)*, *Breadth-First Search (BFS)* atau *Best First Search*.

**Tabel 3.2** Tabel sifat

<b>Kode</b>	<b>Sifat</b>	<b>Indikator</b>
S1	Evaluasi diri secara objektif	Kepercayaan Diri
S2	Positive Thinking	
S3	Selalu berfikir negative pada sendiri.	
S4	Kebutuhan akan prestasi	Berorientasi pada tugas dan hasil
S5	Memiliki tekad yang kuat	
S6	Kurang konsisten dalam mengerjakan tugas	
S7	Berani mengambil resiko	Pengambilan resiko
S8	Suka pada tantangan	
S9	Takut menghadapi kegagalan	
S10	Berjiwa disiplin	Kepemimpinan
S11	Dapat bergaul dengan orang lain	
S12	Tidak bertanggung jawab	
S13	Memiliki inovasi	Kreativitas
S14	Kreativitas tinggi	
S15	Tidak memiliki kreavitas yang tinggi	
S16	Persepsi	Berorientasi pada masa depan
S17	Memiliki cara pandang yang berorientasi pada masa depan	
S18	Tidak ada perencanaan dalam masa depan	
S19	Mengutamakan kejujuran dalam bekerja	Jujur dan tekun
S20	Tekun dalam menyelesaikan suatu pekerjaan	
S21	Mudah menyerah jika ada masalah	

**Sumber:** Data Penelitian (2018)

**Tabel 3.3** Tabel hasil

<b>Kode</b>	<b>Hasil</b>
H1	Mempunyai jiwa <i>entrepreneuership</i>
H2	Kurang kepercayaan diri
H3	Kurang berorientasi pada tugas dan hasil
H4	Kurang pengambilan resiko
H5	Kurang kememimpinan
H6	Kurang Kreativitas
H7	Kurang berorientasi pada masa depan
H8	Kurang jujur dan tekun

**Sumber:** Data Penelitian (2018)

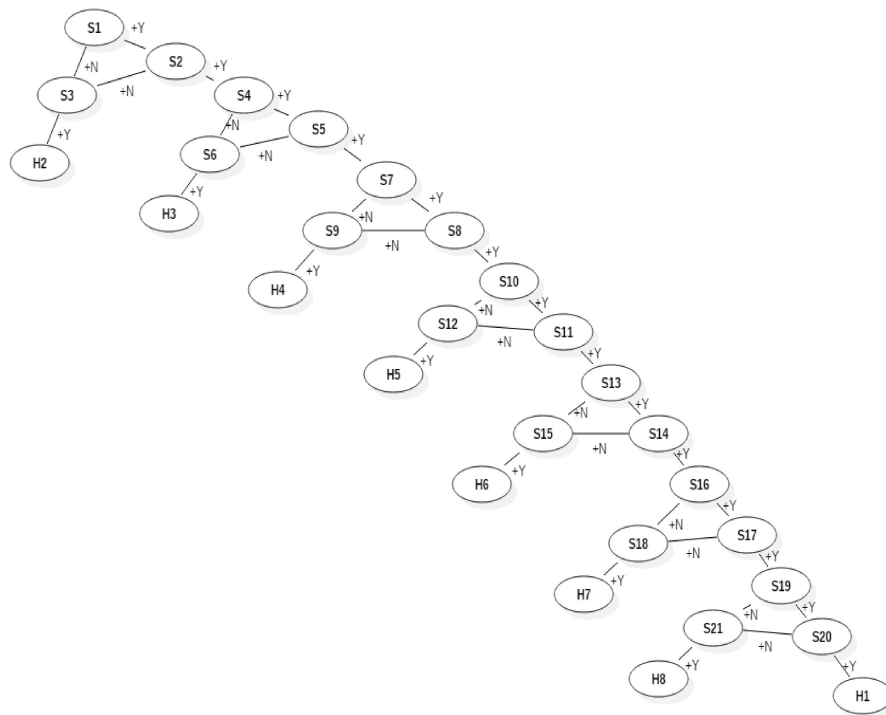
**Tabel 3.4** Tabel aturan

<b>Kode Sifat</b>	<b>Kode Hasil</b>							
	<b>H1</b>	<b>H2</b>	<b>H3</b>	<b>H4</b>	<b>H5</b>	<b>H6</b>	<b>H7</b>	<b>H8</b>
S1	√							
S2	√							
S3		√						
S4	√							
S5	√							
S6			√					
S7	√							
S8	√							
S9				√				
S10	√							
S11	√							
S12					√			
S13	√							
S14	√							
S15						√		
S16	√							
S17	√							
S18							√	
S19	√							
S20	√							
S21								√

**Sumber:** Data Penelitian (2018)

Berdasarkan dari kaidah atas:

1. IF S1 AND S2 AND S4 AND S5 AND S7 AND S8 AND S10 AND S11 AND S13 AND 14 AND S16 AND S17 AND S19 AND S20 THEN H1
2. IF S3 THEN H2
3. IF S6 THEN H3
4. IF S9 THEN H4
5. IF S12 THEN H5
6. IF S15 THEN H6
7. IF S18 THEN H7
8. IF S21 THEN H8



**Gambar 3.2** Pohon Keputusan  
**Sumber:** Data Penelitian

### 3.4 Perancangan Sistem

#### 3.4.1 Diagram UML

Salah satu pemodelan yang saat ini paling banyak digunakan adalah UML. UML (*Unified Modeling Language*) adalah salah standar bahasa yang banyak digunakan di dunia industry untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa & Shalahuddin, 2011).

##### A. Use Case Diagram

Diagram yang menggambarkan aliran sistem yang dapat dilakukan oleh aktor. Aktor disini adalah admin dan pengguna aplikasi ini, di bawah ini merupakan gambar diagram *use case* tersebut.



**Gambar 3.3** Use Case Diagram  
**Sumber:** Data Penelitian (2018)

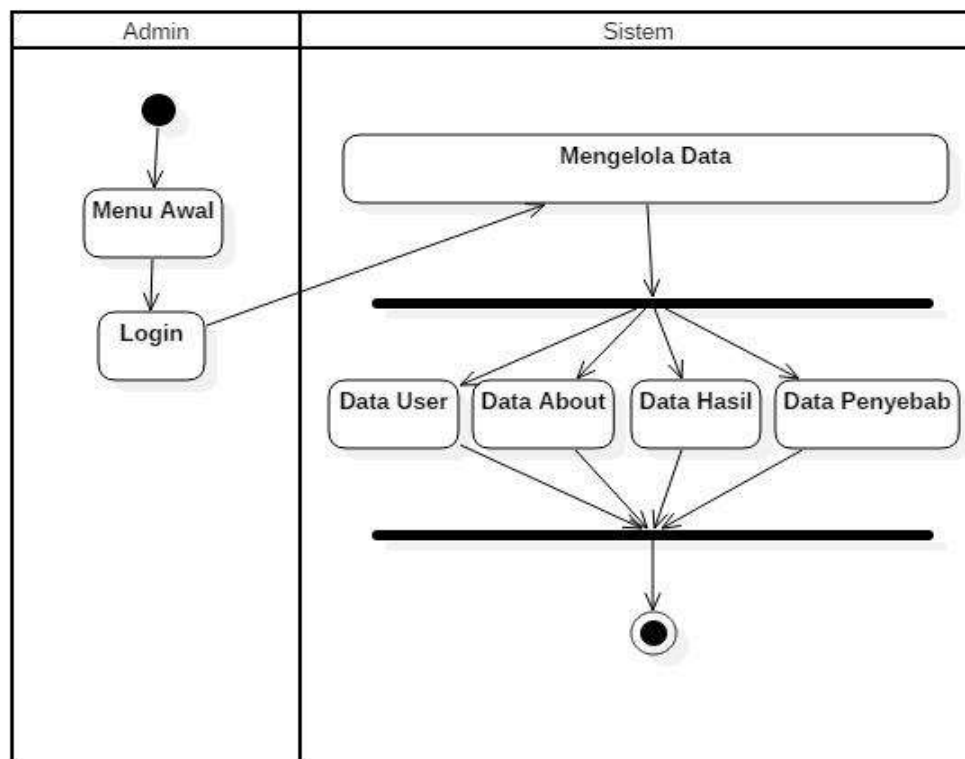


## B. Activity Diagram

*Activity diagram* yaitu aktivitas diagram yang menggambarkan proses-proses yang terjadi pada sistem.

### I. Activity Diagram Admin

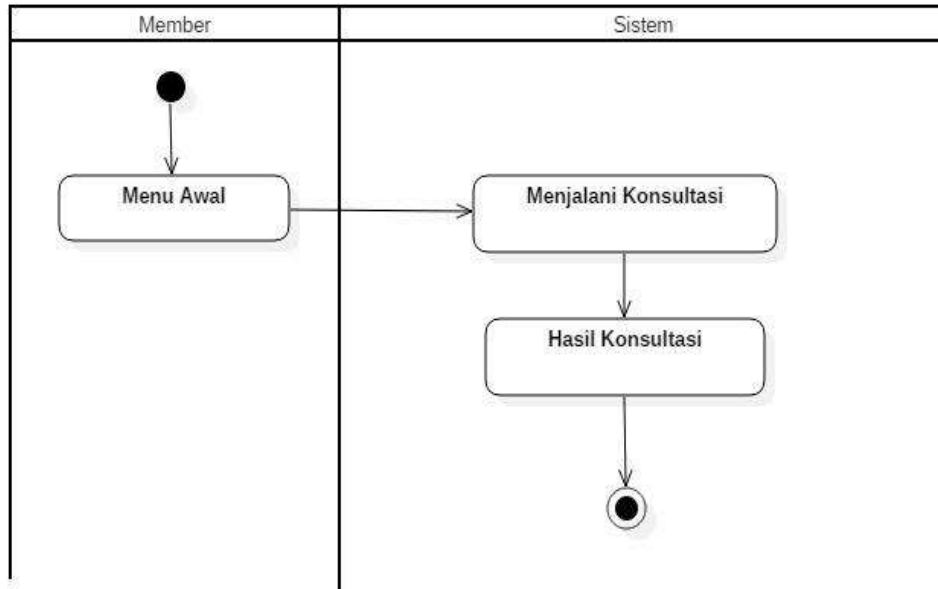
*Activity diagram* ini menjelaskan proses admin, dimana admin melakukan login untuk mengelola data.



**Gambar 3.4** Activity Diagram Admin  
**Sumber:** Data Penelitian (2018)

## II. Activity Diagram Member

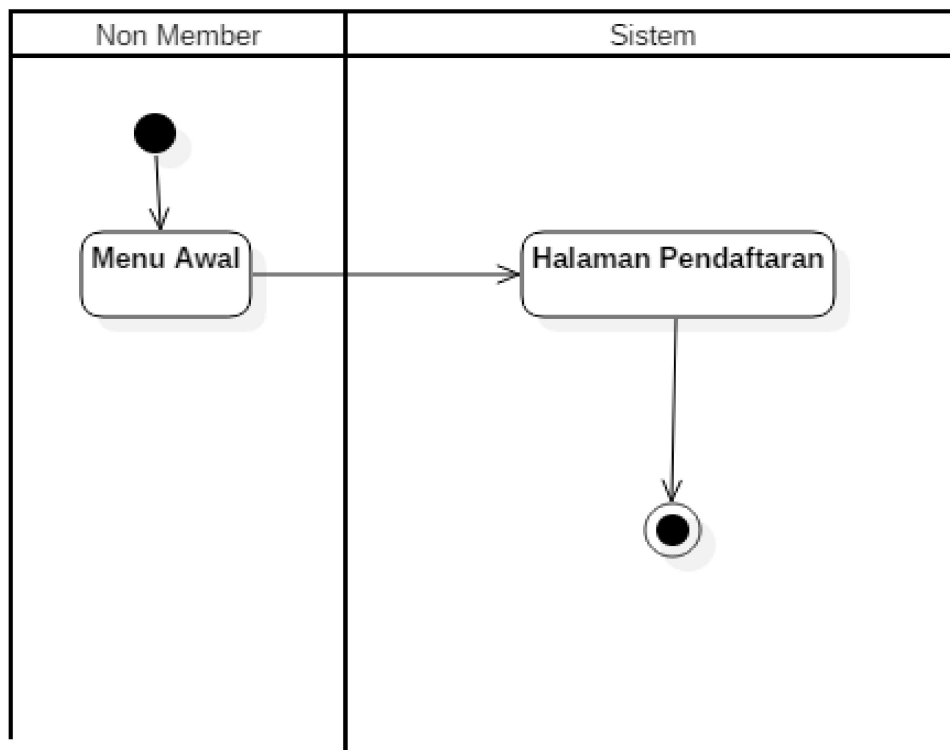
Proses dimana *user* atau pengguna menjalani tes, setelah menjalani tes tersebut *user* akan mengetahui kepribadian yang di miliki.



**Gambar 3.5** Activity Diagram Member  
**Sumber:** Data Penelitian (2018)

### III. Activity Diagram Non-member

Proses ini menjelaskan segala aktivitas yang dilakukan oleh *Non-Member*, sebelum menjalani tes tersebut, akan dipastikan bahwa pengguna tersebut merupakan sudah terdaftar sebagai *Member*, jika belum menjadi *Member* maka akan melakukan pendaftaran.

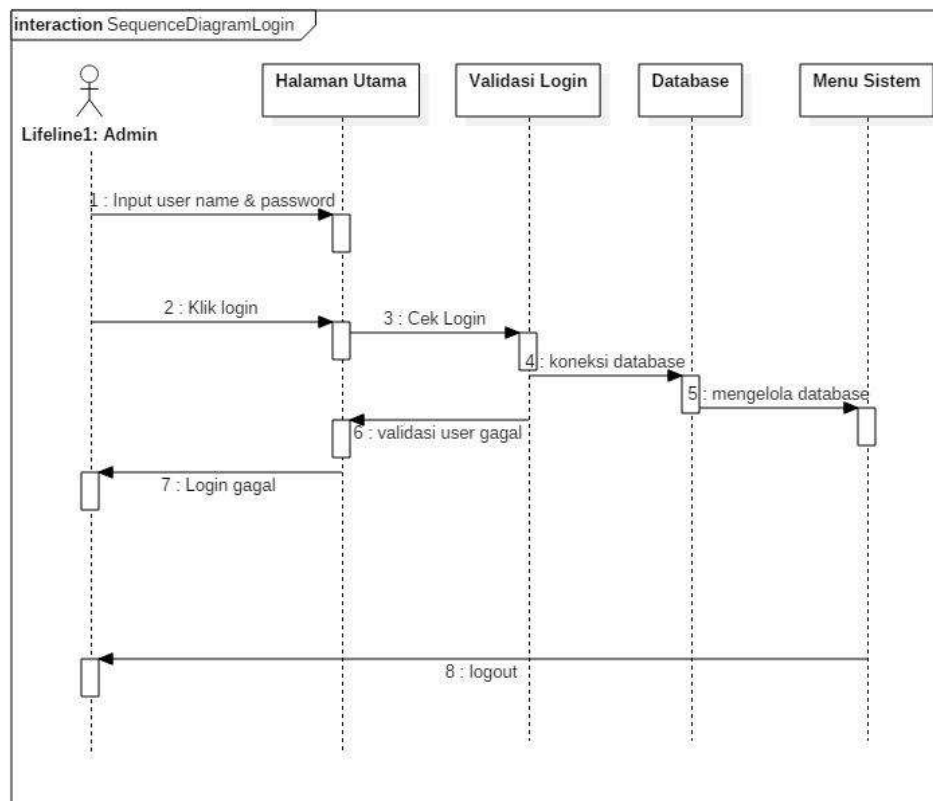


**Gambar 3.6** Activity Diagram Non-member  
Sumber: Data Penelitian (2018)

### C. Sequence Diagram

#### I. Sequence Diagram Login Admin

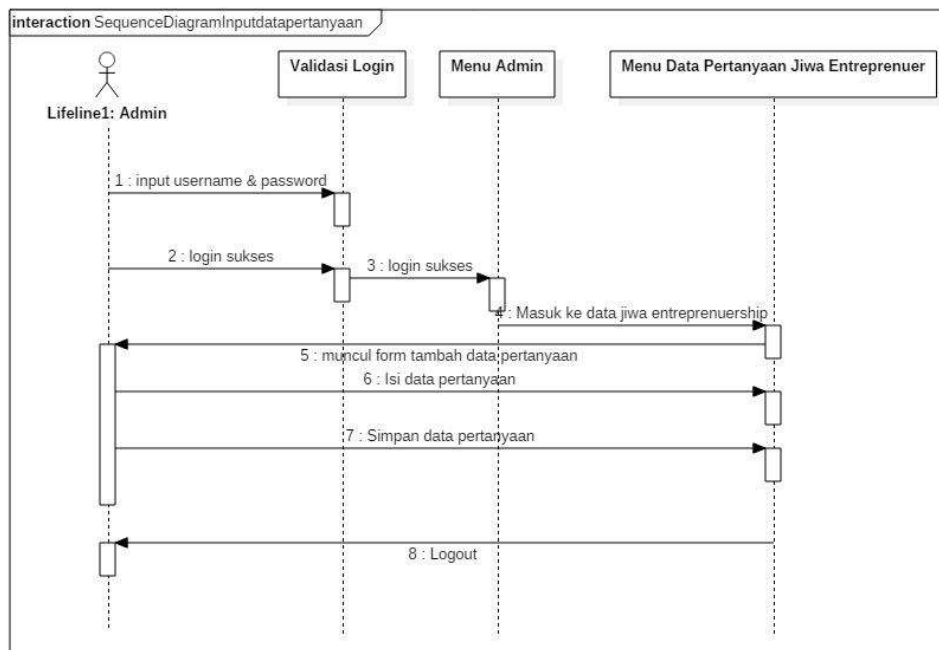
Admin menginput ID dan Password, dilanjutkan dengan klik Login kemudian validasi login, jika valid admin akan masuk ke database dan mengelola data jika invalid atau salah ID/Password maka Login gagal, setelah mengelola data admin logout.



**Gambar 3.7** *Sequence Diagram Login Admin*  
**Sumber:** Data Penelitian (2018)

## II. Sequence Diagram Input Data Pertanyaan

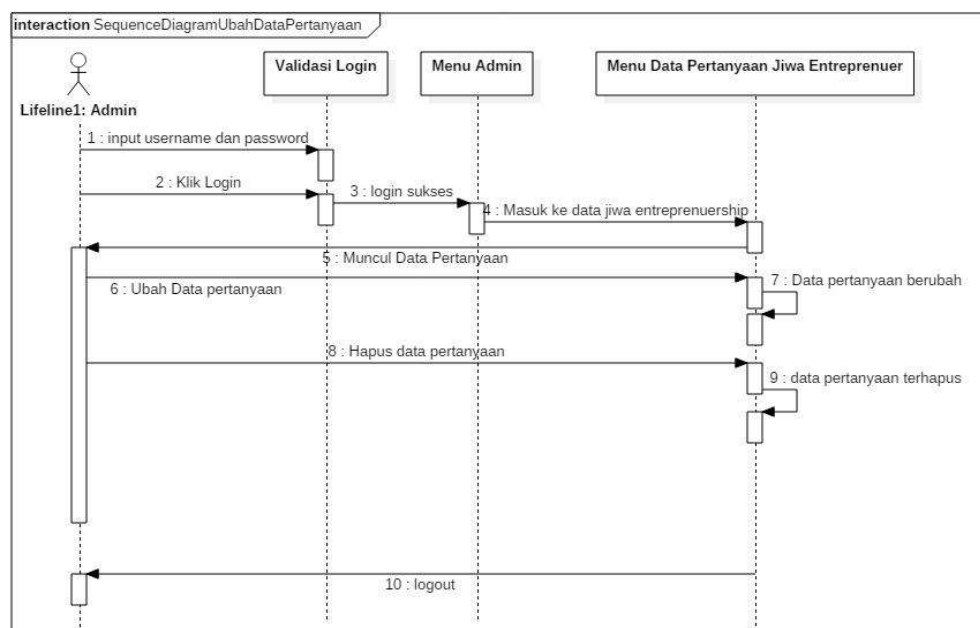
Admin melakukan login, klik login jika login sukses admin akan masuk ke data pertanyaan, disini admin muncul form tambah data pertanyaan, selanjutnya admin akan mengisi form data pertanyaan dan menyimpannya.



**Gambar 3.8** Sequence Diagram input data pertanyaan  
**Sumber:** Data Penelitian (2018)

### III. *Sequence Diagram* Ubah Data Pertanyaan

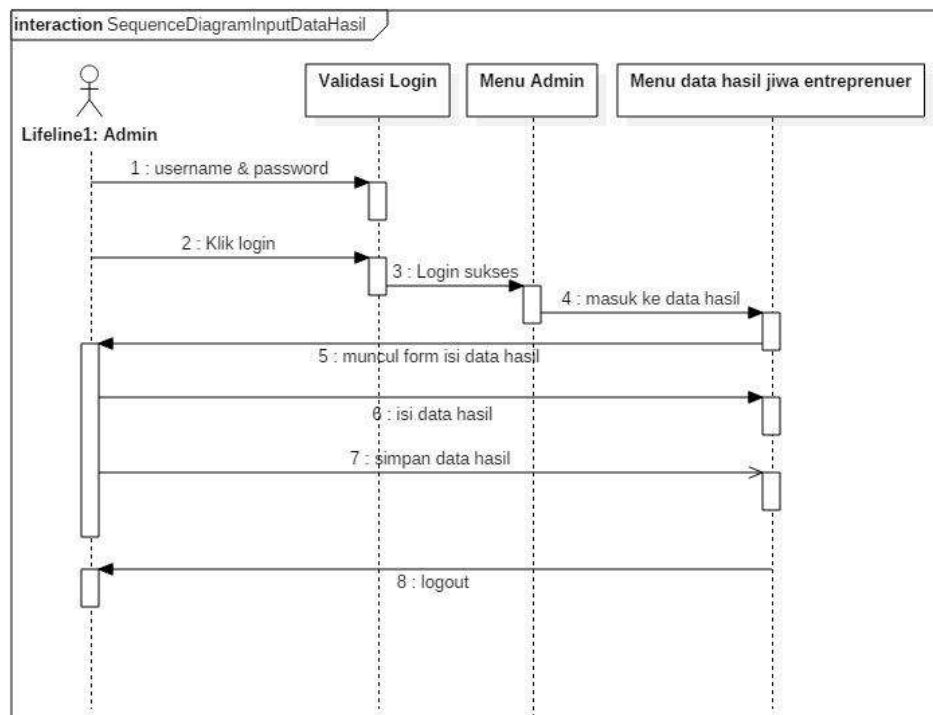
Admin melakukan login, klik login jika login sukses admin akan masuk ke data kerusakan atau penyebab, disini admin muncul form edit dan hapus data kerusakan atau penyebab, jika admin mengedit, data kerusakan atau penyebab akan mengubah, jika admin menghapus, maka data tersebut akan terhapus.



**Gambar 3.9** Sequence Diagram ubah data pertanyaan  
**Sumber:** Data Penelitian (2018)

#### IV. *Sequence Diagram Input data hasil*

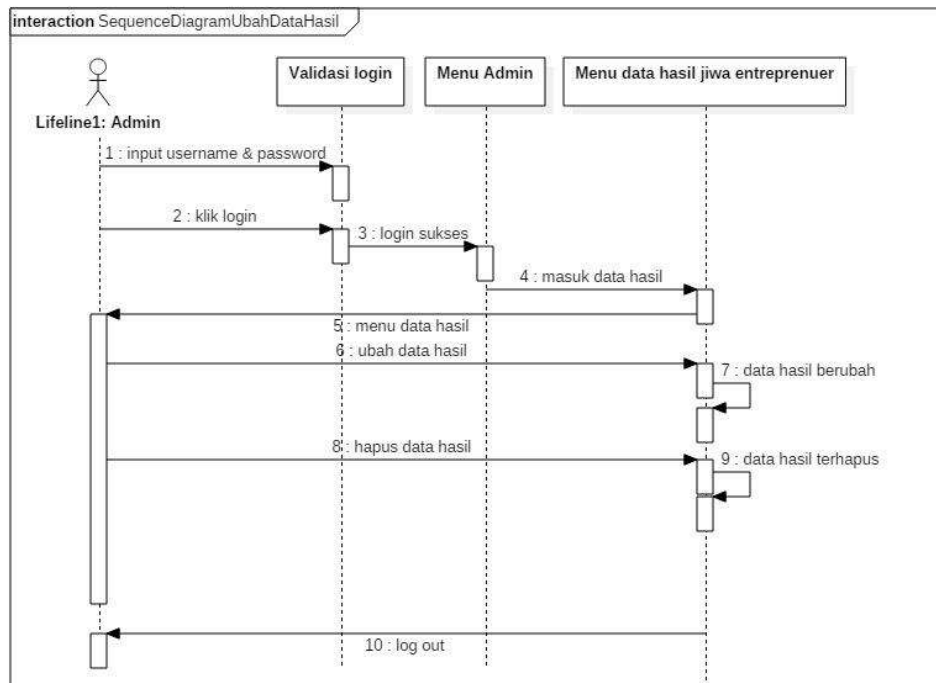
Admin melakukan login, klik login jika login sukses admin akan masuk ke data hasil, disini admin muncul form tambah data hasil, selanjutnya admin akan mengisi form data hasil dan menyimpannya.



**Gambar 3.10** *Sequence Diagram* Input data hasil  
**Sumber:** Data Penelitian (2018)

## V. *Sequence Diagram* ubah data hasil

Admin melakukan login, klik login jika login sukses admin akan masuk ke data hasil, disini admin muncul form edit dan hapus data hasil, jika admin mengedit, data hasil akan mengubah, jika admin menghapus, maka data hasil akan terhapus

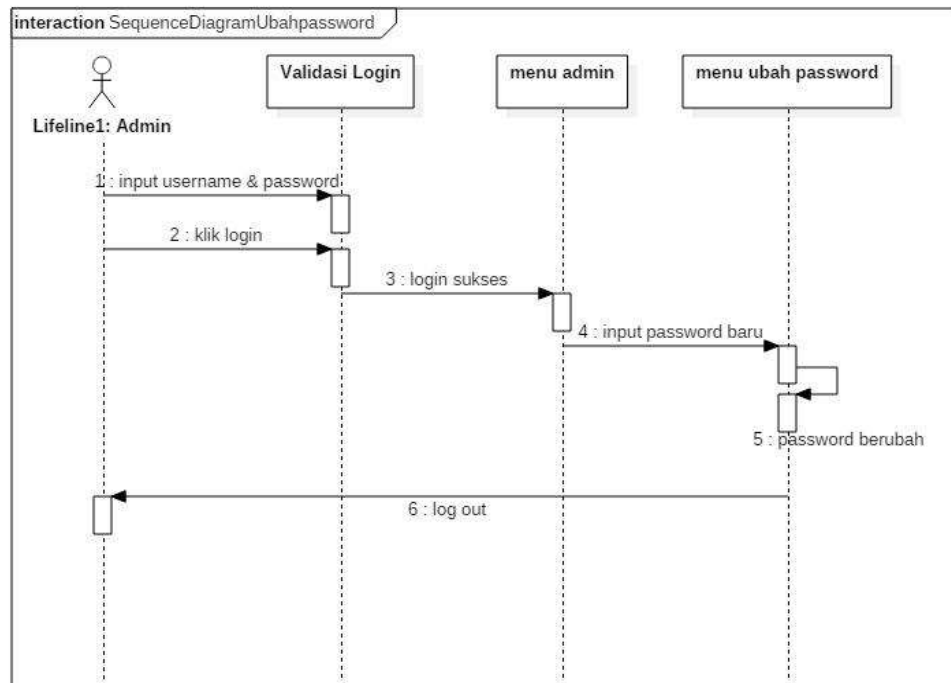


**Gambar 3.11** *Sequence Diagram* ubah data hasil  
**Sumber:** Data Penelitian (2018)



## VI. Sequence Diagram Ubah Password

Admin melakukan login, klik login jika login sukses admin maka akan muncul menu khusus admin dimana terdapat salah satu menu ubah password yang dapat mengubah *password* admin.

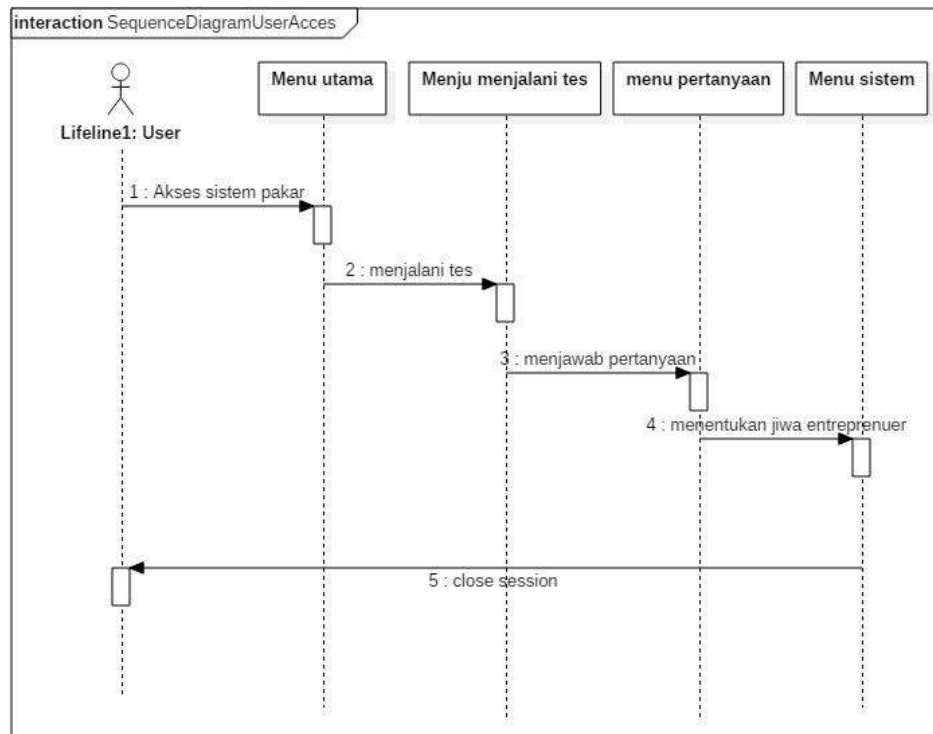


**Gambar 3.12** Sequence Diagram Ubah Password

**Sumber:** Data Penelitian (2018)

## VII. Sequence Diagram User Access

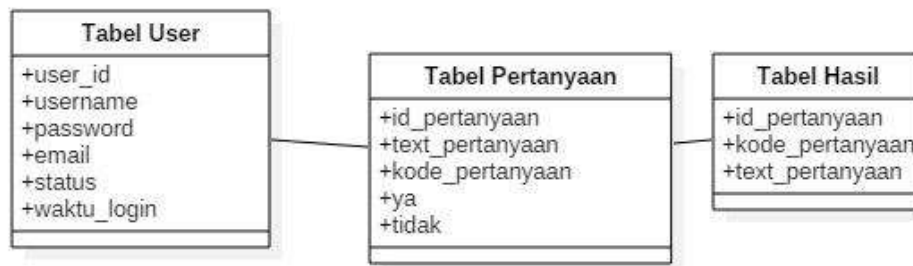
User atau pengguna akan mengakses sistem pakar tersebut dan melakukan tes serta menentukan jiwa entrepreneur sesuai dengan jawaban yang telah dipilih.



**Gambar 3.13** Sequence Diagram User Access  
**Sumber:** Data Penelitian (2018)

#### D. Class diagram

Diagram berikut adalah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut.



**Gambar 3.14** *Class Diagram*  
**Sumber:** Data Penelitian (2018)

#### 3.4.2 Desain Basis Data

##### 1. Tabel basis data user

**Tabel 3.5** Tabel Basis Data user

Field	Tipe	Panjang	Keterangan
User_id	Int	10	Primary Key
Username	Varchar	255	
Password	Varchar	255	
Email	Varchar	255	
Status	Varchar	60	
Waktu_login	Timestamp		

**Sumber:** Data Penelitian (2018)

##### 2. Tabel basis data Pertanyaan

**Tabel 3.6** Tabel Basis Data Pertanyaan

Field	Tipe	Panjang	Keterangan
Id_pertanyaan	Int	11	Primary Key
Kode_pertanyaan	varchar	5	
Text_pertanyaan	Text		
Ya	varchar	5	
Tidak	Varchar	5	

**Sumber:** Data Penelitian (2018)

### 3. Tabel basis data hasil

**Tabel 3.7** Tabel Basis Data hasil

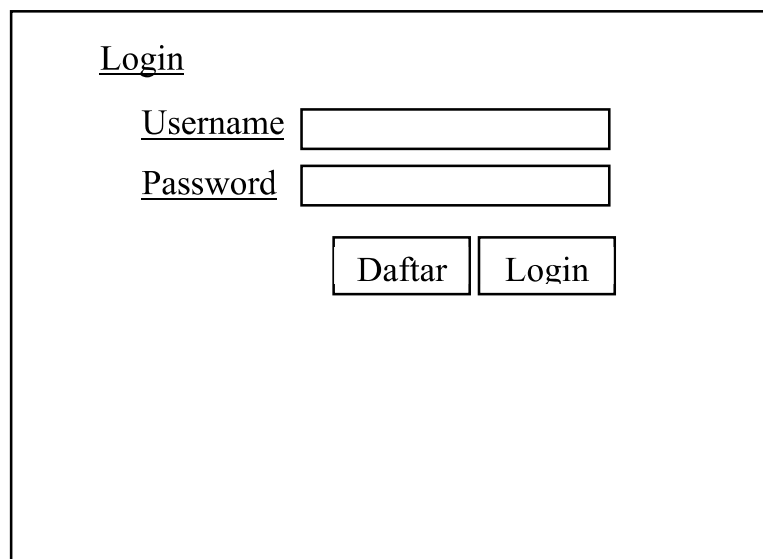
Field	Tipe	Panjang	Keterangan
Id hasil	Int	11	Primary Key
Kode hasil	Varchar	5	
Text hasil	Text		

**Sumber:** Data Penelitian (2018)

### 3.4.3 Desain Antarmuka

#### 1. Menu Form Login

Form ini menampilkan menu login untuk user dan admin.



Login

Username

Password

**Gambar 3.15** Menu Form Login

**Sumber:** Data Penelitian (2018)

## 2. Menu Form Daftar

Pada form ini akan menampilkan bagian pendaftaran member

The image shows a registration form with the following elements:

- Title: Daftar
- Field: Username [input box]
- Field: Email [input box]
- Field: Password [input box]
- Button: **Daftar**

**Gambar 3.16** Menu Form Daftar

**Sumber:** Data Penelitian (2018)

## 3. Menu Form Kelola user

Menu ini akan menampilkan jumlah user yang sudah terdaftar, menu ini hanya digunakan oleh pihak admin dimana untuk menghapus user yang sudah tidak aktif.

The image shows a user management interface with the following elements:

- Navigation buttons: Data Pertanyaan, Data Hasil, User, Logout
- Table of user data:

1. <u>username</u>	<u>Waktu Login</u>	<u>Hapus</u>
2. <u>username</u>	<u>Waktu Login</u>	<u>Hapus</u>
3. <u>username</u>	<u>Waktu Login</u>	<u>Hapus</u>
4. <u>username</u>	<u>Waktu Login</u>	<u>Hapus</u>
5. <u>username</u>	<u>Waktu Login</u>	<u>Hapus</u>

**Gambar 3.17** Menu form kelola user

**Sumber:** Data Penelitian (2018)

## 4. Menu tambah, edit, hapus pertanyaan

Halaman ini merupakan bagian admin untuk menginput, mengubah, dan menghapus data pertanyaan dan sekaligus melakukan *rule*.

<a href="#">Data Pertanyaan</a>	<a href="#">Data Hasil</a>	<a href="#">User</a>	<a href="#">Logout</a>
---------------------------------	----------------------------	----------------------	------------------------

<u>Pertanyaan</u>	<input type="text"/>
<u>Kode</u>	<input type="text"/>
<u>Ya</u>	<input type="text"/>
<u>Tidak</u>	<input type="text"/>

[Simpan](#)

No	kode	Pertanyaan	Ya	Tidak	
1	No	Pertanyaan 1	kode	No	<a href="#">Hapus</a> <a href="#">Edit</a>
2	No	Pertanyaan 2	2	No	<a href="#">Hapus</a> <a href="#">Edit</a>

**Gambar 3.18** menu tambah, edit, hapus pertanyaan

**Sumber:** Data Penelitian (2018)

### 5. Menu tambah, edit, hapus hasil

Halaman ini merupakan bagian admin untuk menginput, mengubah, dan menghapus data hasil.

No	kode	Hasil
1	No	Hasil 1
2	No	Hasil 2

**Gambar 3.19** Menu tambah, edit, hapus hasil

**Sumber:** Data Penelitian (2018)

### 6. Menu menjalani tes

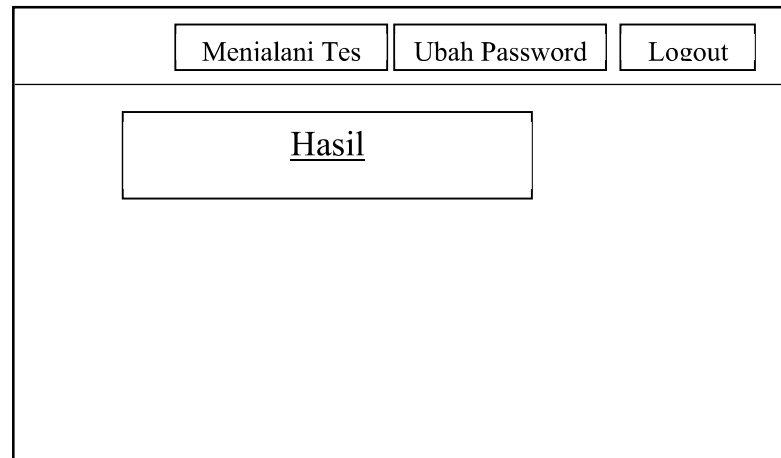
Halaman di sini adalah tampilan halaman untuk menjalani tes untuk menentukan jiwa entrepreneurship bagi yang sudah menjadi member.

**Gambar 3.20** Menu Menjalani tes

**Sumber:** Data Penelitian (2018)

## 7. Menu Hasil Tes

Halaman di sini adalah tampilan halaman setelah menjalani tes jiwa entrepreneurship.

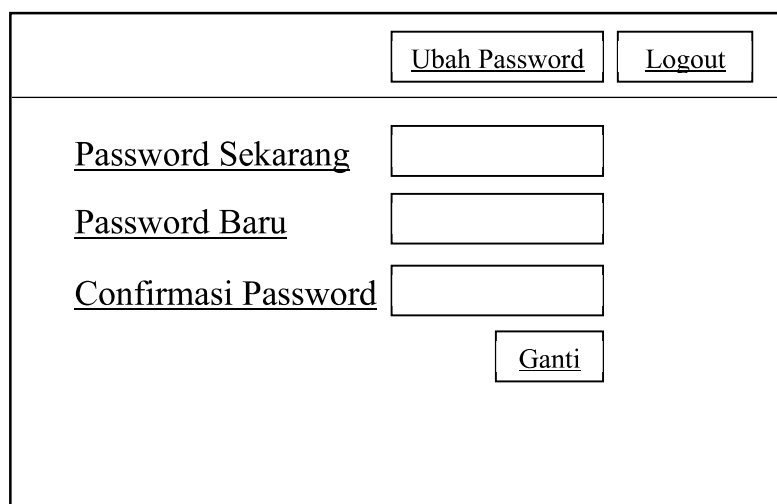


<a href="#">Menjalani Tes</a> <a href="#">Ubah Password</a> <a href="#">Logout</a>
<a href="#">Hasil</a>

**Gambar 3.21** Menu Hasil Tes  
**Sumber:** Data Penelitian (2018)

## 8. Menu Ubah Password

Di sini halaman dimana user mengganti password mereka.



	<a href="#">Ubah Password</a> <a href="#">Logout</a>
<u>Password Sekarang</u>	<input type="text"/>
<u>Password Baru</u>	<input type="text"/>
<u>Confirmasi Password</u>	<input type="text"/>
	<a href="#">Ganti</a>

**Gambar 3.22** Menu Ubah Password  
**Sumber:** Data Penelitian (2018)



