

**PEMBAGIAN WARISAN MENURUT SYARIAT ISLAM
DENGAN BASIS PENGETAHUAN MENGGUNAKAN
PROLOG**



Disusun Oleh:

N a m a : Raditia Fadillah Akbar

NIM : 17523056

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2020

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**BAGIAN INI ADALAH BAGIAN JUDUL – TULIS JUDUL
DENGAN POLA PIRAMIDA TERBALIK (BARIS ATAS
LEBIH PANJANG DARI BARIS BAWAH)**

TUGAS AKHIR

ISLAM

UNIVERSITAS

INDONESIA

Disusun Oleh:

N a m a : Raditia Fadillah Akbar

NIM : 17523056

الجمعة المباركة الأندونيسية

Yogyakarta, 20 Juli 2021

Pembimbing,



(Taufik Hidayat, S.T., MCS.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**PEMBAGIAN WARISAN MENURUT SYARIAT ISLAM
DENGAN BASIS PENGETAHUAN MENGGUNAKAN
PROLOG**

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 26 November 2021

Tim Penguji

Taufiq Hidayat, S.T., M.CS.

Anggota 1

Zainudin Zukhri, S.T., M.IT.

Anggota 2

Dr. Syarif Hidayat, .S.Kom., M.IT.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Raditia Fadillah Akbar
NIM : 17523056

Tugas akhir dengan judul:

PEMBAGIAN WARISAN MENURUT SYARIAT ISLAM DENGAN BASIS PENGETAHUAN MENGGUNAKAN PROLOG

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, Juli 2021



(Raditia Fadillah Akbar)

HALAMAN PERSEMBAHAN

Puji syukur atas ke-hadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga tugas akhir ini dapat dikerjakan tepat pada waktunya. Sholawat dan salam kepada Rasulllah Muhammad Shallallahu ‘alaihi Wassalam, beserta keluarga, sahabat, dan juga kepada para pengikutnya.

Terima kasih saya ucapkan untuk kedua orang tua saya, yang selalu memberikan doa, semangat, motivasi yang tiada hentinya agar saya dimudahkan dalam urusan duniawi dan akhirat. Ter-khusus untuk mama, terima kasih telah membesarkan saya. Sungguh sangat disayangkan mama pergi sebelum saya dapat melaksanakan siding tugas akhir. Semoga mama diterima di sisi Allah SWT.

Terima kasih untuk Bapak Taufik Hidayat, S.T., MCS. selaku pembimbing, ketua program Studi Informatika serta Para Dosen Informatika yang selalu membimbing dan mengajarkan ilmu perkuliahan hingga pelajaran hidup yang tentunya sangat berharga untuk bekal kehidupan yang akan mendatang.

Semoga Allah SWT membalas kebaikan yang kalian berikan dengan keberkahan dan pahala yang sebanyak-banyaknya. Amin.

HALAMAN MOTO

“Allah tidak membebani seorang hamba melainkan sesuai dengan kemampuannya.”

QS. Al-Baqarah : 286

“Sesungguhnya setelah kesulitan ada kemudahan”

QS. Al-Insyirah : 5 s.d 6

"Remember, You Are a NOOB,

So

Be Best Version Of Yourself

And Beat The World

Caused

The Hero Is You”

Braddit

الجامعة الإسلامية
الاستدراك الاندو

KATA PENGANTAR

Puji dan syukur penulis ucapkan atas ke-hadirat Allah SWT berkat karunia-Nya penulis dapat menyelesaikan skripsi ini sebaik mungkin. Penelitian ini tidak akan pernah berjalan sesuai rencana dan tidak berjalan baik tanpa adanya bimbingan, kritik, saran, doa, dan bantuan dari berbagai pihak. Pada kesempatan ini saya sebagai penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada

1. Bpk. Kustam Amukti sebagai ayah yang tiada hentinya memberikan doa, dukungan, semangat dan motivasi selama ini hingga tugas akhir dan masa perkuliahan dapat diselesaikan dengan baik.
2. Almarhumah Ibu.Nuryati sebagai ibu penulis. Selama masa hidupnya beliau tidak pernah berhenti memberikan *support* untuk penulis yang tidak pernah bisa untuk penulis balas kebaikannya.
3. Bapak Dr. Raden Teduh Dirgahayu, S.T.,M.Sc. selaku Ketua program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
4. Bapak Taufik Hidayat, S.T., MCS. selaku Dosen Pembimbing Skripsi, atas segala bimbingan, waktu, masukan serta arahan dalam proses pengerjaan skripsi ini .
5. Seluruh Dosen Jurusan Informatika Universitas Islam Indonesia atas ilmu yang diberikan selama masa perkuliahan.
6. Kepada teman- teman Informatika Universitas Islam Indonesia seperjuangan yang telah menemani baik di waktu senang ataupun susah.

Penulis menyadari bahwa skripsi ini masih memiliki banyak kelemahan, baik isi maupun tata bahasa yang digunakan, maka dari itu penulis mengharapkan kritik dan saran yang bersifat membantu sehingga nantinya didapat hasil yang lebih baik ke-depannya. Penulis berharap semoga skripsi ini dapat bermanfaat bagi penulis dan semua pihak.

Yogyakarta, 20 Juli 2021



(Raditia Fadillah Akbar)



SARI

Dalam Islam pembagian harta warisan merupakan hal yang sensitif. Hal ini dapat menimbulkan rasa iri dan dengki kepada setiap ahli waris. Di era dewasa ini umat Muslim lebih suka membagikan hartanya dengan menggunakan cara dibagi rata. Hal ini terjadi karena kebanyakan umat muslim tidak mengerti tentang tata cara pembagian waris sesuai hukum Islam. Tujuan diadakannya penelitian ini adalah untuk membangun sebuah *expert system* dengan menggunakan bahasa pemrograman logika Prolog. Pembagian waris dalam Islam yang berbentuk aturan akan lebih mudah dibuat di dalam basis pengetahuan Prolog. *Forward Chaining* digunakan sebagai metode yang akan digunakan di dalam basis pengetahuan. *Forward Chaining* bekerja dengan menelusuri fakta-fakta yang tersedia dalam basis pengetahuan terlebih dahulu, lalu dilanjutkan dengan menelusuri aturan-aturan yang dibuat dalam basis pengetahuan. Hasil dari penelitian ini yaitu bagian-bagian harta yang didapatkan oleh setiap ahli waris yang ada. Dengan adanya sistem pakar ini, diharapkan umat Muslim dapat menjalankan syariat yang sudah ditentukan dan dapat dengan mudah membagikan hartanya.

Kata Kunci : Pembagian waris; sistem pakar; *forward chaining*.

GLOSARIUM

<i>Faraidh</i>	Ilmu <i>faraidh</i> adalah ilmu yang memberikan penjelasan tentang pembagian harta warisan di dalam Islam.
<i>Muwarits</i>	Orang yang meninggal dunia.
Harta Warisan	Segala sesuatu baik berbentuk materi atau benda yang ditinggalkan <i>muwarits</i> .
Ahli Waris	Orang yang menerima warisan.
Bagian Waris	Hasil dari perhitungan pembagian waris
AI	<i>Artificial Intelligence (kecerdasan buatan)</i> simulasi dari kecerdasan manusia yang <i>dimodel</i> -kan dalam mesin dan di program agar bisa <i>berfikir</i> layaknya manusia
<i>Expert System</i>	Salah satu bagian dari kecerdasan buatan. <i>Expert system</i> menggunakan pengetahuan suatu pakar di bidangnya yang di transfer ke dalam komputer sebagai basis pengetahuan. Sistem pakar berisi fakta dan aturan yang dijadikan sebagai basis pengetahuan.
PROLOG	Sebuah bahasa pemrograman logika yang sifatnya non-prosedural. Prolog digunakan untuk melakukan <i>algoritme</i> logika yang biasanya terdapat dalam bidang terapan sistem berbasis pengetahuan atau sistem pakar.
<i>Forward Chaining</i>	Sebuah metode runut maju yang biasa digunakan dalam sistem pakar
<i>White-Box Testing</i>	Sebuah metode pengujian yang berfokus pada internal kode



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERSEMBAHAN.....	v
HALAMAN MOTO.....	vi
KATA PENGANTAR.....	vii
SARI	ix
GLOSARIUM	x
DAFTAR ISI	xi
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Manfaat Penelitian	2
1.5 Batasan Penelitian	2
BAB II TINJAUAN PUSTAKA.....	3
2.1 Ilmu <i>Faraidh</i>	3
2.2 Ayat Waris	3
2.3 Hadist Waris.....	6
2.4 <i>Artificial Inteligence (AI)</i>	8
2.5 Sistem Pakar.....	9
2.5.1 Sistem Pakar Sebagai Bagian dari AI	10
2.5.2 Komponen Sistem Pakar	10
2.6 Ciri dan Karakteristik Sistem Pakar.....	12
2.7 Pendekatan penalaran.....	13
2.8 Data Flow Diagram.....	13
2.8.1 <i>Context Diagram (CD)</i>	14
2.8.2 <i>Data flow diagram (DFD)</i>	14
2.8.3 Fungsi masing-masing simbol pada DFD.....	15
2.9 Prolog.....	16

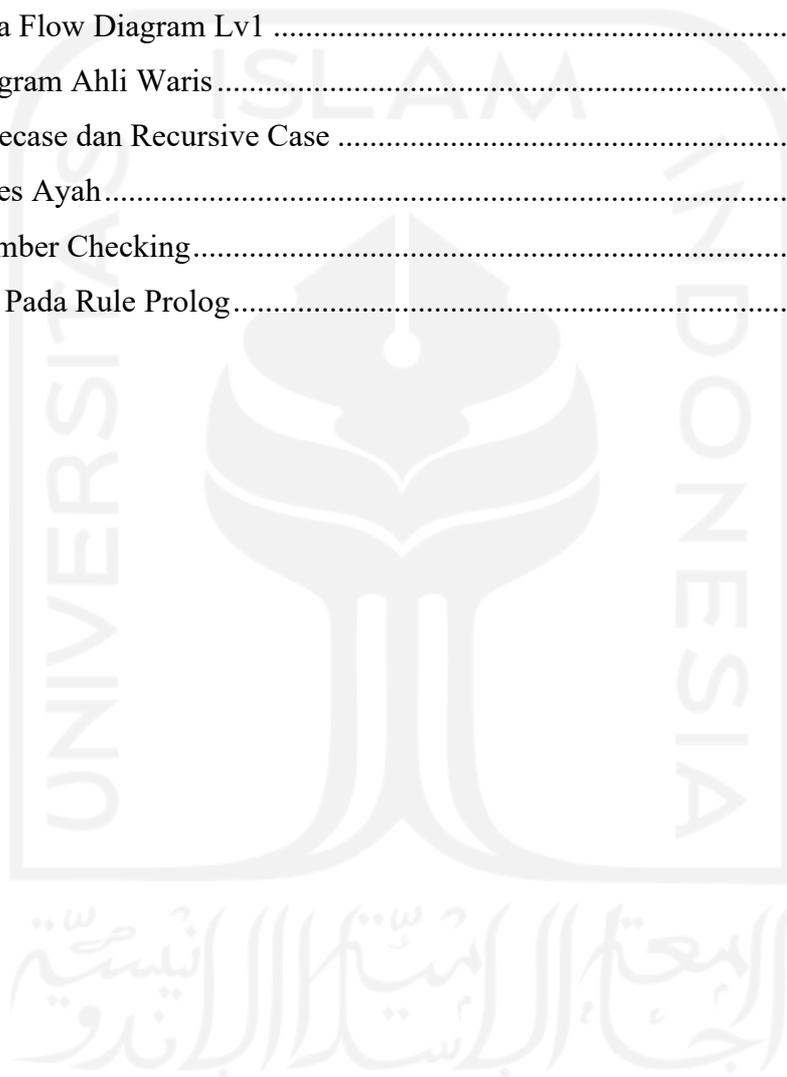
2.10. Forward Chaining	17
2.11. Backward Chaining	18
2.12. Kelebihan dan Kekurangan Sistem Pakar	18
2.13. <i>Black-box Testing</i>	19
2.14. <i>White-box Testing</i>	20
2.15. Rekap Penelitian Terdahulu	21
BAB III METODELOGI PENELITIAN	24
3.1. Metode Penelitian	24
3.2. Identifikasi Masalah	24
3.2.1. Jenis Data	24
3.2.2. Sumber Data	25
3.2.3. Analisis Perangkat Lunak	27
3.2.4. Analisis Perangkat Keras	27
3.3. Perancangan	28
3.3.1. <i>Input dan output</i>	28
3.3.2. Rancangan Sistem	28
3.3.3. <i>User Interface</i>	30
3.3.4. Rancangan <i>Rules</i>	30
3.4. Pengujian	30
BAB IV HASIL DAN PEMBAHASAN	31
4.1. Pembahasan	31
4.2. Implementasi Sistem	37
4.2.1. Fakta dan aturan	37
4.2.2. <i>Base case dan recursive case</i>	37
4.2.3. Implementasi <i>Rules</i>	38
4.2.4. <i>Member Checking</i>	38
4.2.5. Prolog Cut	39
4.3. Pengujian	40
4.4. Validasi	44
BAB V KESIMPULAN DAN SARAN	47
5.1. Kesimpulan	47
5.2. Saran	47
DAFTAR PUSTAKA	49
LAMPIRAN	51

DAFTAR TABEL

Tabel 2.1 Simbol <i>Data Flow Diagram</i>	15
Tabel 2.2 Rekap Penelitian Terdahulu.....	21
Tabel 4.1 Ahli Waris Laki-laki	32
Tabel 4.2 Ahli Waris Perempuan.....	32
Tabel 4.3 <i>Rules</i> Ahli Waris Utama	33
Tabel 4.4 <i>Rules</i> Ahli Waris Kedua	34
Tabel 4.5 <i>Rules</i> Ahli Waris Ketiga	35
Tabel 4.6 Test Case Ahli Waris Pertama	40
Tabel 4.7 <i>Test Case</i> Ahli Waris Kedua.....	42
Tabel 4.8 <i>Test Case</i> Ahli Waris Ketiga	43
Tabel 4.9 Validasi Ahli Waris Pertama	45
Tabel 4.10 Validasi Ahli Waris Kedua.....	45
Tabel 4.11 Validasi Ahli Waris ketiga.....	46

DAFTAR GAMBAR

Gambar 2.1 Domain AI.....	8
Gambar 2.2 Inferensi Forward Chaining	17
Gambar 2.3 Inferensi Backward Chaining.....	18
Gambar 3.1 Alur Penelitian Terapan	24
Gambar 3.2 Data Flow Diagram Lv 0	29
Gambar 3.3 Data Flow Diagram Lv1	29
Gambar 4.1 Diagram Ahli Waris	31
Gambar 4.2 Basecase dan Recursive Case	37
Gambar 4.3 Rules Ayah.....	38
Gambar 4.4 Member Checking.....	39
Gambar 4.5 Cut Pada Rule Prolog.....	40



BAB I PENDAHULUAN

1.1 Latar Belakang

Dalam Islam pembagian warisan dilakukan dengan menggunakan ilmu *Faraidh*. Ilmu *Faraidh* adalah ilmu yang memberikan penjelasan tentang pembagian harta warisan di dalam Islam (Septiana et al., 2015). Ilmu *Faraidh* sendiri merupakan ilmu yang tinggi kedudukannya di mata Allah SWT, sehingga Allah sendiri yang menentukan sendiri takarannya (Minarni et al., 2018). Pembagian warisan dapat menimbulkan rasa dan iri dan dengki kepada setiap ahli waris, baik dalam segi perhitungan ataupun keadilan dalam pembagian waris. Karenanya dibutuhkan ketelitian dalam perhitungan dan keadilan dalam membagikan harta warisan. Allah SWT memerintahkan umat-Nya dalam Q.S *An-Nisa* : 11 s.d 12 dan Q.S *An-Nisa* ayat 176 untuk membagikan hartanya sesuai syariat, karena Allah SWT maha mengetahui yang terbaik bagi umat-Nya. Pembagian waris dalam Islam terdiri dari orang yang meninggal(*muwarits*), orang yang menerima harta warisan(*ahli waris*). Segala benda atau kepemilikan yang ditinggalkan oleh *muwarits* disebut harta warisan (Septiana et al., 2015).

Di era dewasa ini, banyak umat Islam yang enggan membagikan hartanya berdasarkan syariat atau hukum Islam. Hal ini dikarenakan saat ini banyak umat Muslim tidak mengerti bagaimana cara membagikan warisan sesuai syariat Islam. Kebanyakan umat Muslim menganggap pembagian warisan sesuai syariat Islam merupakan sesuatu yang rumit (Septiana et al., 2015). Cukup sulit ditemukannya ahli ilmu *Faraid* untuk dimintai pertolongan dalam pembagian waris. Oleh karena itu umat Islam sekarang cenderung melakukan pembagian waris secara manual dan merata untuk setiap ahli waris (Dahria, 2011). Hal ini dapat menimbulkan rasa tidak adil bagi ahli waris dan menimbulkan sifat iri dan dengki.

Kemajuan teknologi dapat dimanfaatkan untuk membantu dalam menyelesaikan permasalahan ini. Salah satu domain dari AI yaitu, sistem pakar (*expert system*) dapat digunakan untuk mencari bagian-bagian harta untuk setiap ahli waris. Sistem pakar merupakan sistem yang berisikan pengetahuan seorang pakar(*ahli*) agar dapat membantu manusia dalam menyelesaikan masalah layaknya seorang pakar (Dahria, 2011).

Dalam sistem pakar terdapat metode pelacakan ke-depan atau yang biasa disebut *Forward chaining*. Pelacakan akan dimulai dari data yang ada lalu mencari melalui aturan-aturan hingga tercapainya sebuah solusi (Brata et al., 2012). Bahasa pemrograman Prolog digunakan dalam penelitian ini. Penggunaan bahasa pemrograman ini sangat sesuai dengan

hukum pembagian waris yang berbentuk aturan-aturan. Prolog juga mendukung metode pelacakan *Forward chaining*.

Penelitian ini bertujuan untuk mendapatkan bagian-bagian waris untuk setiap ahli waris. Bagian-bagian waris tertera di dalam Al-Quran dan Hadits. Harapannya umat Muslim dapat lebih mudah dalam membagikan harta warisannya tanpa menimbulkan rasa iri dan dengki antar ahli waris.

1.2 Rumusan Masalah

Rumusan masalah dalam tugas akhir ini adalah bagaimana cara pembagian warisan sesuai dengan syariat Islam menggunakan bahasa pemrograman logika Prolog?

1.3 Tujuan Penelitian

Adapun tujuan penelitian ini adalah :

- a. Membuat sistem pakar untuk pembagian warisan menurut syariat Islam menggunakan bahasa pemrograman logika Prolog.
- b. Umat muslim dapat membagikan harta warisannya sesuai syariat.

1.4 Manfaat Penelitian

Manfaat penelitian ini adalah :

- a. Memudahkan pembagian warisan dalam Islam berdasarkan syariat.
- b. Menambah pengetahuan tentang bahasa pemrograman logika dan hukum pembagian waris bagi penulis.

1.5 Batasan Penelitian

Supaya Penelitian ini lebih terarah dan memiliki batasan permasalahan yang ada, maka skripsi ini memiliki batasan penelitian sebagai berikut :

- a. Penelitian ini hanya mengambil pembagian warisan menurut ilmu *Faraidh (ilmu bagi waris)* dari Al-Quran dan Hadits.
- b. Perhitungan warisan hanya dapat dilakukan sampai generasi pertama.
- c. Hasil perhitungan warisan berupa bagian untuk setiap ahli waris.
- d. Metode yang digunakan untuk membuat basis pengetahuan sistem pakar adalah *Forward Chaining*.
- e. Bahasa pemrograman untuk sistem pakar menggunakan bahasa pemrograman logika Prolog.

BAB II TINJAUAN PUSTAKA

2.1 Ilmu *Faraidh*

Ilmu *Faraidh* merupakan ilmu yang akan pertama kali hilang, adapun salah satu penyebab hilangnya yaitu ditinggalkannya ilmu *Faraidh* karena mengartikan bahwa pembagian warisan secara adil itu diukur dengan sama rata, sehingga tidak banyak orang yang ingin mempelajari atau mengimplementasikan ilmu tersebut dan meninggalkan syariat Islam (Faqihuddin et al., 2016). Ilmu *Faraidh* (*pembagian waris*) memberikan penjelasan tentang cara-cara pembagian harta waris seperti siapa saja yang berhak mendapat harta waris dan berapa bagiannya. Karena pembagian harta waris ini merupakan permasalahan yang *sensitif*, maka perlu juga ketelitian saat melakukan perhitungan pembagian harta waris.

Fakta di masyarakat biasanya perhitungan pembagian harta waris masih dilakukan secara manual dan menggunakan alat bantu hitung. Hal ini terkadang menyebabkan tidak akurat dan tidak efisien dalam proses perhitungannya.

Seseorang yang menghitung pembagian harta waris harus sangat teliti agar tidak terjadi kesalahan yang fatal yang bisa mengakibatkan konflik lainnya (Minarni et al., 2018). Dalam agama Islam pembagian harta warisan secara syariat atau hukum Islam adalah wajib hukumnya.

Dalam hukum Islam sebaiknya harta warisan segera dibagikan dan tidak ditunda-tunda. Membagi waris pada dasarnya adalah menentukan berapa bagian setiap ahli waris atas harta warisan, kemudian membagi harta warisan sesuai dengan ketentuan yang sudah ada di dalam hukum Islam yang mana setiap ahli waris mendapatkan bagiannya sesuai dengan porsi atau bagian masing-masing sesuai syariat yang ada (Wahyunadi et al., 2015).

2.2 Ayat Waris

Pembagian waris dalam Islam diatur dalam Q.S An-Nisa ayat 11-12 dan Q.S An-Nisa ayat 176. Adapun pelegkap ayat pembagian waris adalah ada pada Q.S An-Nisa ayat 7.

مَا تَرَكَ وَإِنْ كَانَتْ وَاحِدَةً فَلَهَا النِّصْفُ وَلِأَبَوَيْهِ لِكُلِّ وَاحِدٍ مِّنْهُمَا السُّدُسُ مِمَّا

تَرَكَ إِنْ كَانَ لَهُ وَلَدٌ فَإِنْ لَمْ يَكُنْ لَهُ وَلَدٌ وَوَرِثَهُ أَبُوَاهُ فَلِأُمِّهِ الثُّلُثُ فَإِنْ كَانَ لَهُ

إِخْوَةٌ فَلَأَمِّهِ السُّدُسُ مِنْ بَعْدِ وَصِيَّةٍ يُوصِي بِهَا أَوْ دَيْنٍ آبَاؤُكُمْ وَأَبْنَاؤُكُمْ لَا تَدْرُونَ أَيُّهُمْ أَقْرَبُ لَكُمْ نَفْعًا فَرِيضَةٌ مِنَ اللَّهِ إِنَّ اللَّهَ كَانَ عَلِيمًا حَكِيمًا (١١)

QS An-Nissa Ayat 11

Artinya :

Allah mensyari'atkan kepadamu tentang (pembagian warisan untuk) anak-anakmu, yaitu bagian seorang anak laki-laki sama dengan bagian dua orang anak perempuan. Jika anak itu semuanya perempuan yang jumlahnya lebih dari dua, maka bagian mereka dua pertiga dari harta yang ditinggalkan. Jika anak perempuan itu seorang saja, maka dia memperoleh setengah (harta yang ditinggalkan). Untuk kedua ibu-bapak, bagian masing-masing *seperenam* dari harta yang ditinggalkan, jika yang meninggal itu mempunyai anak. Jika orang yang meninggal tidak mempunyai anak dan dia diwarisi oleh kedua ibu bapaknya (saja), maka ibunya mendapat sepertiga. Jika yang meninggal itu mempunyai beberapa saudara, maka ibunya mendapat *seperenam*. (Pembagian-pembagian tersebut di atas) setelah dipenuhi wasiat yang dibuatnya atau (dan) setelah dibayar hutangnya. (Tentang) orang tuamu dan anak-anakmu, kamu tidak mengetahui siapa di antara mereka yang lebih banyak manfaatnya bagimu. Ini adalah ketetapan Allah. Sungguh, Allah Maha Mengetahui lagi Maha bijaksana.

لَكُمْ نِصْفُ مَا تَرَكَ أَزْوَاجُكُمْ إِنْ لَمْ يَكُنْ لَهُنَّ وَلَدٌ فَإِنْ كَانَ لَهُنَّ وَلَدٌ فَلَكُمْ الرُّبْعُ مِمَّا تَرَكَنَّ مِنْ بَعْدِ وَصِيَّةٍ يُوصِينَ بِهَا أَوْ دَيْنٍ وَلَهُنَّ الرُّبْعُ مِمَّا تَرَكَتُمْ إِنْ لَمْ يَكُنْ لَكُمْ وَلَدٌ فَإِنْ كَانَ لَكُمْ وَلَدٌ فَلَهُنَّ النُّصُبُ مِمَّا تَرَكَتُمْ مِنْ بَعْدِ وَصِيَّةٍ تُوصُونَ بِهَا أَوْ دَيْنٍ وَإِنْ كَانَ رَجُلٌ يُورَثُ كَلَالَةً أَوْ امْرَأَةٌ وَلَهُ أَخٌ أَوْ أُخْتٌ فَلِكُلِّ وَاحِدٍ مِنْهُمَا السُّدُسُ فَإِنْ كَانُوا أَكْثَرَ مِنْ ذَلِكَ فَهُمْ شُرَكَاءُ فِي التُّلْتِ مِنْ بَعْدِ وَصِيَّةٍ يُوصَى بِهَا أَوْ دَيْنٍ غَيْرَ مُضَارٍّ وَصِيَّةً مِنَ اللَّهِ وَاللَّهُ عَلِيمٌ حَلِيمٌ (١٢)

QS An-Nissa Ayat 12

Artinya :

Bagianmu (suami-suami) adalah seperdua dari harta yang ditinggalkan oleh istri-istri *mu*, jika mereka tidak mempunyai anak. Jika mereka (istri-istrimu) itu mempunyai anak, maka kamu mendapat seperempat dari harta yang ditinggalkannya setelah dipenuhi wasiat yang mereka buat atau (dan) setelah dibayar hutangnya. Para istri memperoleh seperempat harta yang kamu tinggalkan jika kamu tidak mempunyai anak. Jika kamu mempunyai anak, maka para istri memperoleh seperdelapan dari harta yang kamu tinggalkan setelah dipenuhi wasiat yang kamu buat atau (dan) setelah dibayar hutang-hutangmu. Jika seseorang meninggal, baik laki-laki maupun perempuan yang tidak meninggalkan ayah dan tidak meninggalkan anak, tetapi mempunyai seorang saudara laki-laki (seibu) atau seorang saudara perempuan (seibu), maka bagi masing-masing dari kedua jenis saudara itu *seperenam* harta. Jika saudara-saudara seibu itu lebih dari seorang, maka mereka bersama-sama dalam bagian yang sepertiga itu, setelah dipenuhi wasiat yang dibuatnya atau (dan) setelah dibayar hutangnya dengan tidak menyusahkan (kepada ahli waris). Demikianlah ketentuan Allah. Allah Maha Mengetahui lagi Maha Penyantun.

يَسْتَفْتُونَكَ قُلِ اللَّهُ يُفْتِيكُمْ فِي الْكَلَالَةِ إِنَّ أَمْرًا هَلَكَ لَيْسَ لَهُ وَلَدٌ وَلَهُ
 أُخْتُ فَلَهَا نِصْفُ مَا تَرَكَ وَهُوَ يَرِثُهَا إِنْ لَمْ يَكُنْ لَهَا وَلَدٌ فَإِنْ كَانَتَا
 اثْنَتَيْنِ فَلَهُمَا الثُّلُثَانِ مِمَّا تَرَكَ وَإِنْ كَانُوا إِخْوَةً رِجَالًا وَنِسَاءً فَلِلذَّكَرِ مِثْلُ
 حَظِّ الْأُنثِيَيْنِ يُبَيِّنُ اللَّهُ لَكُمْ أَنْ تَضِلُّوا وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ (١٧٦)

QS An-Nissa Ayat 176.

Artinya :

Mereka meminta fatwa kepadamu (tentang kalalah). Katakanlah, "Allah memberi fatwa kepadamu tentang kalalah (yaitu), jika seorang meninggal dunia, dan dia tidak mempunyai anak (dan ayah) tetapi mempunyai saudara perempuan, maka bagi saudaranya yang perempuan itu seperdua dari harta yang ditinggalkannya, dan saudaranya yang laki-laki mewarisi (seluruh harta saudara perempuan), jika dia tidak mempunyai anak. Jika saudara perempuan itu dua orang, maka bagi keduanya dua pertiga dari harta yang ditinggalkan. Jika mereka (ahli waris itu terdiri dari) saudara-saudara laki dan perempuan, maka bagian seorang saudara laki-laki sama dengan bagian dua saudara perempuan. Allah menerangkan (hukum ini) kepadamu, agar kamu tidak sesat. Allah Maha mengetahui segala sesuatu.

لِلرِّجَالِ نَصِيبٌ مِّمَّا تَرَكَ الْوَالِدَانِ وَالْأَقْرَبُونَ وَلِلنِّسَاءِ نَصِيبٌ مِّمَّا تَرَكَ الْوَالِدَانِ
وَالْأَقْرَبُونَ مِمَّا قَلَّ مِنْهُ أَوْ كَثُرَ ۗ نَصِيبًا مَّفْرُوضًا

Artinya : Bagi laki-laki ada hak bagian dari harta peninggalan kedua orang tua dan kerabatnya, dan bagi perempuan ada hak bagian (pula) dari harta peninggalan kedua orang tua dan kerabatnya, baik sedikit atau banyak menurut bagian yang telah ditetapkan.

QS An-Nissa Ayat 7.

2.3 Hadist Waris

Selain hadirnya ayat-ayat waris yang diturunkan oleh Allah SWT, ada juga hadits-hadits yang diturunkan kepada sahabat nabi. Fungsi hadits adalah sebagai sumber ilmu kedua setelah Al-Quran.

a. Hadits ke-43 dari Jamiul Ulum wal Hikam Ibnu Rajab

عَنِ ابْنِ عَبَّاسٍ رَضِيَ اللَّهُ عَنْهُمَا قَالَ: قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ
أَلْجَفُوا الْقَرَائِضَ بِأَهْلِهَا، فَمَا أَبَقَتِ الْقَرَائِضُ قَلَاءً لِي رَجُلٍ ذَكَرَ
خَرَجَهُ الْبُخَارِيُّ وَمُسْلِمٌ

Dari Ibnu ‘Abbas *radhiyallahu ‘anhuma*, ia berkata bahwa Rasulullah *shallallahu ‘alaihi wa sallam* bersabda, “Berikan bagian warisan kepada ahli warisnya, selebihnya adalah milik laki-laki yang paling dekat dengan mayit.” (HR. Bukhari, no. 6746 dan Muslim, no. 1615)

b. H.R al-Baihaqi

الخال وارث من لا وارث له

Artinya, “Paman dari jalur ibu mewarisi orang yang tidak memiliki pewaris.” (HR. al-Baihaqi)

c. Hadits sunan Ibnu Majah No. 2728 – kitab waris

حَدَّثَنَا أَبُو بَكْرِ بْنُ أَبِي شَيْبَةَ حَدَّثَنَا شَبَابَةُ ح و حَدَّثَنَا مُحَمَّدُ بْنُ الْوَلِيدِ حَدَّثَنَا مُحَمَّدُ بْنُ جَعْفَرٍ قَالَ حَدَّثَنَا شُعْبَةُ حَدَّثَنِي بُدَيْلُ بْنُ مَيْسَرَةَ الْعُقَيْلِيُّ عَنْ عَلِيِّ بْنِ أَبِي طَلْحَةَ عَنْ رَاشِدِ بْنِ سَعْدٍ عَنْ أَبِي عَامِرٍ الْهُوزَنِيِّ عَنْ الْمُقْدَامِ أَبِي كَرِيمَةَ رَجُلٌ مِنْ أَهْلِ الشَّامِ مِنْ أَصْحَابِ رَسُولِ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ مَنْ تَرَكَ مَالًا فَلِوَرَثَتِهِ وَمَنْ تَرَكَ كَلًّا فَالَيْنَا وَرَبَّمَا قَالَ فَإِلَى اللَّهِ وَإِلَى رَسُولِهِ وَأَنَا وَارِثٌ مَنْ لَا وَارِثَ لَهُ أَعْقَلُ عَنْهُ وَارِثُهُ وَالْخَالُ وَارِثٌ مَنْ لَا وَارِثَ لَهُ يَعْقِلُ عَنْهُ وَيَرِثُهُ

Telah menceritakan kepada kami **Abu Bakar bin Abu Syaibah**; telah menceritakan kepada kami **Syababah**; Demikian juga diriwayatkan dari jalur yang lainnya, telah menceritakan kepada kami **Muhammad bin Al Walid**; telah menceritakan kepada kami **Muhammad bin Ja'far**, keduanya berkata; telah menceritakan kepada kami **Syu'bah**, telah menceritakan kepadaku **Budail bin Maisarah Al 'Uqaili** dari **Ali bin Abu Thalhah** dari **Rasyid bin Sa'ad** dari **Abu Amir Al Hauzani** dari **Al Miqdam Abu Karimah**, ia adalah seorang laki-laki penduduk Syam yang merupakan salah satu sahabat Rasulullah shallallahu 'alaihi wasallam. Ia berkata; Rasulullah shallallahu 'alaihi wasallam bersabda: "Barangsiapa yang meninggalkan harta, maka hal tersebut untuk ahli warisnya. Dan Barangsiapa yang meninggalkan keluarga dan hutang yang memberatkan dirinya, maka diserahkan kepada kami." Dan barang kali Rasulullah shallallahu 'alaihi wasallam bersabda: 'Maka serahkanlah kepada Allah dan Rasul-Nya. Dan aku adalah ahli waris bagi orang yang tidak memiliki ahli waris. Aku adalah orang yang membayarkan diyat dan mewarisinya. Dan paman adalah ahli waris bagi orang yang tidak memiliki ahli waris. Dialah yang membayarkan diyat dan yang mendapatkan harta warisan."

d. Hadits Sunan Abu Dawud No. 2511 – kitab waris

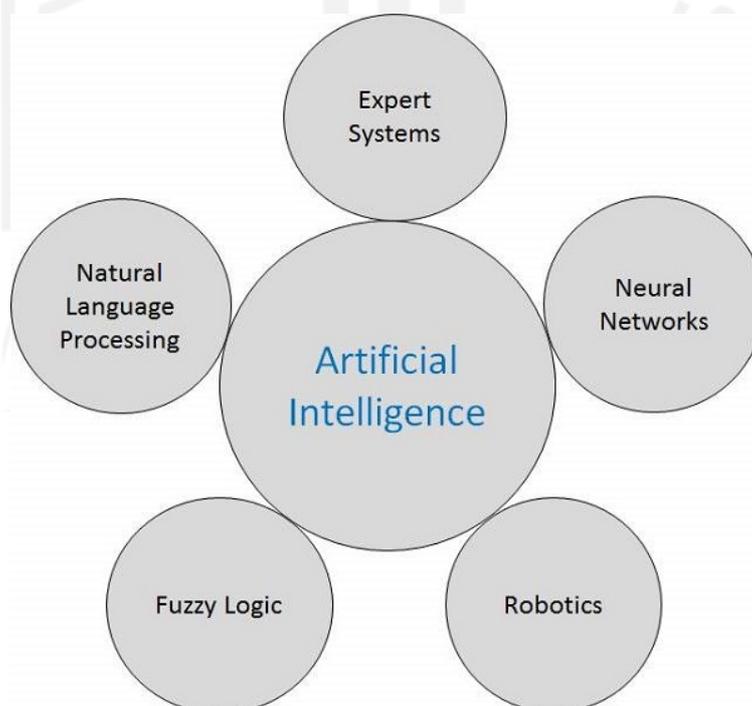
حَدَّثَنَا أَحْمَدُ بْنُ صَالِحٍ وَمَخْلَدُ بْنُ خَالِدٍ وَهَذَا حَدِيثٌ مَخْلَدٍ وَهُوَ الْأَشْبَعُ قَالَ حَدَّثَنَا عَبْدُ الرَّزَّاقِ حَدَّثَنَا مَعْمَرٌ عَنْ ابْنِ طَاوُسٍ عَنْ أَبِيهِ عَنْ ابْنِ عَبَّاسٍ قَالَ

قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ أَفْسِمَ الْمَالَ بَيْنَ أَهْلِ الْفَرَائِضِ عَلَى كِتَابِ
اللَّهِ فَمَا تَرَكَتْ الْفَرَائِضُ فَلِأَوْلَى ذَكَرِ

Telah menceritakan kepada kami **Ahmad bin Shalih**, dan **Makhlad bin Khalid**, dan ini adalah hadits Makhlad dan hadits tersebut lebih bagus (patut diterima). Mereka berdua mengatakan; telah menceritakan kepada kami **Abdurrazzaq**, telah menceritakan kepada kami **Ma'mar** dari **Ibnu Thawus** dari ayahnya dari **Ibnu Abbas**, ia berkata; Rasulullah shallallahu 'alaihi wasallam berkata: "Bagikan harta di-antara para pemilik faraidh (bagian harta waris) berdasarkan Kitab Allah. Maka bagian harta yang tersisa setelah pembagian tersebut, lebih utama diberikan kepada (ahli waris) laki-laki."

2.4 *Artificial Intelligence (AI)*

Artificial Intelligence adalah salah satu cabang ilmu komputer yang memungkinkan sebuah komputer dapat berpikir dan memiliki penalaran layaknya seorang manusia, (Durkin, 1994). Tentu hal ini dilakukan agar komputer dapat lebih dimanfaatkan untuk memudahkan urusan manusia. Saat ini *Artificial Intelligence* dimanfaatkan oleh manusia untuk mencari informasi yang lebih akurat, membuat keputusan, dan membuat kehidupan manusia semakin mudah.



Gambar 2.1 Domain AI

Domain dari setiap *Artificial Intelligence* memiliki peran masing-masing dalam suatu bidang tertentu. Implementasi dari *Artificial Intelligence* di-antaranya :

a. *Fuzzy logic*

Saat ini *fuzzy logic* banyak digunakan untuk keperluan elektronik dan *robotik*. Alat-alat yang dikembangkan menggunakan *fuzzy logic* sudah banyak diimplementasikan ke dalam alat elektronik rumah tangga. Perkembangan *robotik* juga sudah mulai gencar dilakukan pada saat ini. Penggunaan *fuzzy logic* ini memungkinkan sebuah alat dapat berpikir layaknya manusia.

b. *Computer vision*

Bidang *Computer vision* banyak digunakan untuk mengenali citra yang di-masukan ke dalam sistem. Pemanfaatannya banyak digunakan untuk mengenali tulisan, mengenali objek dalam gambar, dan manipulasi citra.

c. *Game*

Saat ini dunia *Game* sudah berkembang sangat pesat. Dengan adanya pemanfaatan dari kecerdasan buatan membuat variasi *game* yang lebih beragam. Kecerdasan buatan dalam dunia *game* banyak digunakan untuk menciptakan *environment* yang lebih realistis. Contohnya pengembangan NPC yang menggunakan AI akan lebih dinamis dengan kondisi *player* saat memainkan sebuah *game*.

d. *Speech recognition*

Penggunaan *speech recognition* memungkinkan sebuah sistem dapat mengenali suara manusia dengan cara di-cocokkan dengan pola yang sudah dibuat sebelumnya. Penggunaan *speech recognition* saat ini dapat kita jumpai dengan mudah pada *google assistant* dan *siri*.

e. *Expert system*

Expert system atau sistem pakar merupakan sebuah cabang dari kecerdasan buatan yang memungkinkan manusia sebagai *user* biasa dapat memiliki keputusan layaknya seorang pakar. Kepakaran seseorang akan dipelajari oleh *engineer* untuk dibuatkan sebuah *knowledge base* sehingga dapat digunakan oleh orang awam.

2.5 Sistem Pakar

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut (Wamiliana, et al. 2016).

Sistem pakar merupakan sistem yang berisikan pengetahuan seorang pakar (*ahli*) agar dapat membantu manusia dalam menyelesaikan masalah layaknya seorang pakar (Dahria et al., 2011). Dengan adanya sistem pakar manusia akan lebih mudah dalam menyelesaikan masalah yang membutuhkan pengetahuan seorang pakar. Oleh karena itu sistem pakar sangat membutuhkan keahlian atau pengalaman yang biasa disebut *knowledge* dari seorang pakar (Listiyono et al., 2008).

Orang yang memiliki suatu keahlian atau *knowledge* dalam suatu *domain* permasalahan disebut pakar. Keahlian atau *knowledge* ini bersifat khusus karena orang lain tidak menguasai atau mengetahui dengan bidang yang ditekuninya (Listiyono et al., 2008).

Seorang pakar akan mencari solusi yang memuaskan manusia atau seseorang yang membutuhkannya (*client*) (Faqihuddin et al., 2016). Oleh karena itu sistem pakar dibuat pada ranah atau domain tertentu yang spesifik dan mendekati kemampuan manusia itu sendiri (*pakar*).

Kajian pokok dalam sistem pakar adalah bagaimana *mentransfer* pengetahuan yang dimiliki oleh seorang pakar ke dalam *komputer*, dan bagaimana membuat keputusan atau mengambil kesimpulan berdasarkan pengetahuan itu (Faqihuddin et al., 2016). Dengan menyimpan *informasi* dan digabungkan dengan himpunan aturan penalaran yang memadai memungkinkan komputer memberikan kesimpulan atau mengambil keputusan seperti seorang pakar.

2.5.1 Sistem Pakar Sebagai Bagian dari AI

Sistem pakar adalah salah satu bagian dari kecerdasan buatan (Brata et al., 2012). Sistem pakar menggunakan pengetahuan suatu pakar di bidangnya yang di *transfer* ke dalam komputer sebagai basis pengetahuan. Sistem pakar berisi fakta dan aturan yang dijadikan sebagai basis pengetahuan. Sistem pakar memungkinkan untuk melakukan penalaran untuk mencari fakta baru atau sebuah kesimpulan atau mengambil keputusan seperti seorang pakar.

2.5.2 Komponen Sistem Pakar

Lingkungan pengembang dan lingkungan konsultasi merupakan bagian utama dari sistem pakar. lingkungan pengembang digunakan untuk memindahkan *knowledge* seorang pakar sedangkan lingkungan *konsultasi* digunakan oleh *user* guna memperoleh pengetahuan seorang pakar (Listiyono, 2008). Komponen-komponen yang terdapat dalam sistem pakar

yaitu antarmuka pengguna, basis pengetahuan (*knowledge-base*), akuisisi pengetahuan, mesin inferensi, *workplace*, fasilitas penjelasan, perbaikan pengetahuan.

a. Antarmuka pengguna (*user interface*)

Antarmuka pengguna merupakan bagian yang digunakan oleh *user* untuk berkomunikasi dengan *sistem*. Sistem pakar akan menampilkan pertanyaan-pertanyaan yang harus dijawab oleh pengguna sesuai dengan masalah yang dihadapi. Proses *input* terjadi dalam antarmuka pengguna dan juga *output* sebagai respons dari *input* yang di-masukan.

b. Basis pengetahuan (*knowledge base*)

Di dalam basis pengetahuan terdapat pengetahuan yang *kodifikasi* dari seorang pakar. Basis pengetahuan erat kaitannya dengan mesin *inferensi* dan tidak bisa dipisahkan. Karena dalam basis pengetahuan terdapat dua elemen dasar, yaitu fakta dan aturan.

c. Akuisisi pengetahuan

Akuisisi pengetahuan merupakan *proses transfer* pengetahuan seorang pakar ke dalam program *komputer*. *Engineer* atau seorang *programmer* berusaha menyerap pengetahuan dari seorang pakar lalu *ditransfer* dan *dikodifikasi* dalam basis pengetahuan.

d. Mesin inferensi

Mesin *inferensi* merupakan program komputer yang memberikan *metodologi* untuk penalaran informasi yang ada di dalam basis pengetahuan (*knowledge base*) dan dalam *workplace* untuk terbentuknya suatu kesimpulan atau solusi (Turban :1995:1). Komponen ini mirip dengan pola pikir dan penalaran yang digunakan oleh seorang pakar dalam menyelesaikan masalah.

e. *Workplace*

Sekumpulan *memori kerja* (*working memory*) terdapat dalam *workplace*. Fungsi *workplace* adalah untuk merekam keputusan. Keputusan yang dapat direkam di-antaranya, rencana, agenda, dan solusi.

f. Fasilitas penjelasan

Fasilitas penjelasan menggambarkan penalaran *sistem* kepada pemakai. Ini merupakan suatu komponen tambahan dalam sistem pakar.

g. Perbaikan pengetahuan

Kemampuan untuk belajar, menganalisis, dan meningkatkan kinerjanya merupakan ciri yang dimiliki oleh seorang pakar. Kemampuan tersebut juga penting dalam pembelajaran

komputerisasi, sehingga program dapat menganalisis penyebab kesuksesan dan kegagalan yang dialaminya.

h. Representasi pengetahuan

Representasi pengetahuan merupakan suatu metode untuk *mengkodekan* pengetahuan ke dalam basis pengetahuan dalam sistem pakar. Representasi pengetahuan bertujuan untuk menangkap sifat-sifat penting suatu permasalahan dan membuat informasi tersebut dapat diakses oleh prosedur pemecahan masalah. Terdiri dari fakta dan formula. Fakta menerangkan objek yang direpresentasikan dan formula menerangkan aturan yang menggambarkan hubungan antar objek.

2.6 Ciri dan Karakteristik Sistem Pakar

Menurut Jogiyanto (2003) terdapat beberapa ciri yang dimiliki oleh sistem pakar. Ciri ini yang membedakan sistem pakar dengan sistem lainnya. Perbedaan ini dapat menjadi pedoman atau petunjuk dalam pengembangan sebuah sistem pakar. Berikut beberapa Ciri dan karakteristik yang dimiliki oleh sistem pakar.

- a. Komputer melakukan pengolahan data secara numerik, sedangkan keahlian yang dimiliki oleh seorang pakar merupakan fakta dan aturan, sehingga pengetahuan dari sistem pakar sendiri merupakan sebuah konsep, dan bukan berbentuk numeris.
- b. Sistem pakar membutuhkan kemampuan untuk dapat belajar sendiri, karena informasi yang ada di dalam sebuah sistem pakar tidak selalu lengkap, subjektif, dan terkadang tidak konsisten. Oleh karena itu, keputusan yang diambil bersifat tidak pasti dan tidak mutlak “ya” atau “tidak” menurut ukuran tertentu.
- c. Solusi yang diberikan dalam suatu permasalahan dapat berjumlah lebih dari satu, sehingga diperlukannya fleksibilitas dari sistem itu sendiri dalam menangani berbagai macam solusi.
- d. Pengetahuan yang ada dalam sistem pakar dapat berubah setiap waktu, sehingga diperlukannya kemudahan dalam memodifikasi kode yang ada.
- e. Setiap pakar dapat memberikan pertimbangan-pertimbangan secara subyektif. Seorang pakar dapat memberikan pandangan yang berbeda dalam suatu domain, oleh karena itu tidak ada solusi sistem pakar yang pasti benar.
- f. Fasilitas penjelasan sistem sangat diperlukan karena sebuah sistem pakar harus memberikan solusi yang akurat kepada pengguna meskipun solusi yang dicari merupakan permasalahan yang rumit.

2.7 Pendekatan penalaran

Menurut (Dahria, n.d.), dalam sistem pakar terdapat dua jenis pendekatan penalaran *dalam knowledge base*, yaitu penalaran berbasis aturan (*rule-based reasoning*) dan penalaran berbasis kasus (*base case reasoning*). Kedua macam pendekatan penalaran ini memiliki peran masing-masing pada setiap kasus atau domain masalah yang dihadapi.

a. Penalaran berbasis aturan (*rule-based reasoning*)

Pengetahuan dari seorang pakar akan di representasikan dalam bentuk aturan *IF-THEN*. Ini baik digunakan jika kita memiliki atau bekerja sama dengan seorang pakar yang sudah memiliki pengetahuan dari suatu domain permasalahan. Penalaran berbasis aturan juga dapat digunakan untuk melihat langkah penalaran dari awal hingga terbentuknya suatu solusi.

b. Penalaran berbasis kasus (*base case reasoning*)

Penalaran berbasis kasus digunakan untuk mencari solusi-solusi baru yang sesuai dengan masalah yang sedang dihadapi (fakta). Basis pengetahuan akan berisikan dengan solusi-solusi yang sudah ada sebelumnya. Solusi yang didapat dari sistem merupakan solusi yang memiliki kemiripan dengan solusi yang sudah ada sebelumnya. Bentuk ini baik digunakan apabila kita sudah memiliki kasus dan situasi yang sudah ada di dalam basis pengetahuan

2.8 Data Flow Diagram

Data flow diagram (DFD) adalah merupakan model proses data yang dirancang ataupun dibuat untuk menggambarkan aliran data, mulai dari mana data tersebut masuk dan *kemana* tujuan dari data itu sendiri. DFD dapat dikatakan juga yaitu, representasi grafis dari aliran data melalui sistem informasi.

DFD merupakan alat bantu dalam menggambarkan atau menjelaskan proses kerja suatu sistem. DFD menurut Mahyuzir, 1991 adalah teknik grafik yang digunakan untuk menjelaskan aliran *transformasi* dan informasi data yang bergerak dari pemasukan data hingga ke keluaran. DFD menggambarkan penyimpanan data dan proses *mentransformasikan* data.

Penggambaran DFD diawali dengan diagram konteks (CD). CD menggunakan tiga simbol, yaitu: entitas eksternal, simbol untuk mewakili aliran data dan simbol untuk mewakili proses.

2.8.1 *Context Diagram (CD)*

CD Menunjukkan seluruh sistem. Semua entitas eksternal harus dideskripsikan sehingga aliran data terlihat pada proses input dan output. CD menggunakan tiga simbol: simbol yang mewakili unit eksternal, simbol yang mewakili aliran data, dan simbol yang mewakili proses. CD dapat dikonfigurasi dengan hanya satu proses, tidak lebih, dan perangkat penyimpanan data tidak ditulis ke CD.

Proses CD biasanya tidak bernomor. Diagram ini adalah diagram tingkat atas dari DFD yang menunjukkan hubungan antara sistem dan lingkungan luarnya, caranya adalah:

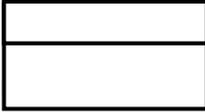
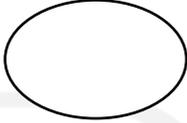
- a. Masukkan nama sistem.
- b. Tentukan batasan sistem.
- c. Menentukan *terminator* dalam sistem.
- d. Menentukan *terminator* mana yang menerima/memberi/memberi dari sistem.
- e. Gambarkan diagram konteks.

2.8.2 *Data flow diagram (DFD)*

Data flow diagram (DFD) juga dikenal sebagai *Data Flow Diagram* (DAD). DFD adalah model atau proses logika data yang dibuat untuk menggambarkan sumber data dan sumber sistem, lokasi data, proses menghasilkan data, dan interaksi antara data yang disimpan dan proses yang dikenakan padanya (Kristanto, 2008). DFD, dalam bahasa Indonesia disebut DAD (*Data Flow Diagram*), menguraikan proses *input/output* sistem/perangkat lunak, aliran objek data yang dihasilkan yang mengalir ke dalam perangkat lunak dan ditransformasikan oleh elemen-elemen pengolah dari sistem / perangkat lunak (S. Pressman, 2012).

DFD menggunakan empat simbol. Artinya, semua simbol yang digunakan pada CD dan satu simbol tambahan yang mewakili memori data. Ada dua teknik dasar untuk menggambar simbol DFD yang banyak digunakan. Salah satunya adalah Gane dan Sarson, dan yang lainnya adalah Yourdon dan DeMarco. Perbedaan mendasar dalam teknologi adalah simbol dari simbol yang digunakan. Perbedaan simbol dapat dilihat dalam tabel 2.1.

Tabel 2.1 Simbol *Data Flow Diagram*

Nama	Gane dan Sarson	Yourdon dan DeMarco
External Entitiy		
Process		
Data Store		
Data Flow		

Gane dan Sarson menggunakan simbol persegi panjang tumpul di bagian atas untuk menggambarkan proses dan simbol persegi terbuka di sebelah kanan untuk menggambarkan penyimpanan data.

Yourdon dan DeMarco menggunakan simbol yem untuk menggambarkan proses dan simbol garis paralel untuk menggambarkan penyimpanan data. Kedua teknik menggunakan simbol yang sama untuk simbol entitas eksternal dan simbol aliran data. Yaitu, persegi yang melambangkan entitas eksternal dan panah yang melambangkan aliran data.

2.8.3 Fungsi masing-masing simbol pada DFD

Menurut (Jogiyanto, 2005) ada empat simbol dalam DFD yang masing-masing digunakan untuk merepresentasikan aliran data.

a. *External Entity* (kesatuan luar)

External Entity (kesatuan luar) atau *boundary* (batas sistem) merupakan pihak di luar sistem, digunakan untuk menyatakan: suatu kantor, departemen atau divisi dalam perusahaan tetapi di luar sistem yang dikembangkan, orang atau sekelompok orang di organisasi tetapi di luar sistem yang sedang dikembangkan.

b. *Process*

Process digunakan untuk merepresentasikan suatu aktivitas atau tugas yang dilakukan oleh seseorang, mesin, atau komputer dari hasil aliran data yang masuk ke proses untuk menghasilkan aliran data yang keluar dari proses atau disebut *input* dan *output*. Proses harus menerima aliran data dan menghasilkan aliran data.

c. Penyimpanan Data (*data storage*)

Penyimpanan Data (*data storage*) Digunakan untuk mewakili penyimpanan data. Penyimpanan data dapat dalam format berikut: Sebuah *file* atau *database* pada sistem komputer. Arsip atau rekaman manual sebuah kotak berisi data di meja seseorang, Tabel referensi manual agenda atau buku. Berikut adalah beberapa hal yang perlu dipertimbangkan ketika menjelaskan penyimpanan data

d. Penyimpanan Data (*data storage*)

Data Flow adalah aliran data yang mengalir antara *terminator* atau *external entity*, proses, dan penyimpanan data atau *data store*. Aliran data diwakili oleh simbol panah dan fungsi utamanya adalah untuk mentransfer informasi dari satu sistem ke sistem lainnya.

2.9 Prolog

Dalam penelitian ini penulis menggunakan bahasa pemrograman logika Prolog dan *software* SWI - Prolog. Colmerauer dan Kowalski 1972, Prolog merupakan sebuah bahasa pemrograman logika yang sifatnya *non-prosedural*. Prolog digunakan untuk melakukan *algoritme* logika yang biasanya terdapat dalam bidang terapan sistem berbasis pengetahuan atau sistem pakar.

Bahasa pemrograman Prolog biasanya digunakan untuk keperluan mengajar dan aplikasi web semantik. SWI Prolog juga memiliki beberapa *fitur* seperti pemrograman kendala logika, *multithreading*, *unit testing*, GUI, *interfacing* ke Java, ODBC, server web, pengembang alat (termasuk sebuah IDE dengan GUI *debugger* dan *profiler*), dan dokumentasi yang ekstensif. SWI-Prolog dapat berjalan pada platform Windows, Linux, dan Macintosh (Ian et al., 2014).

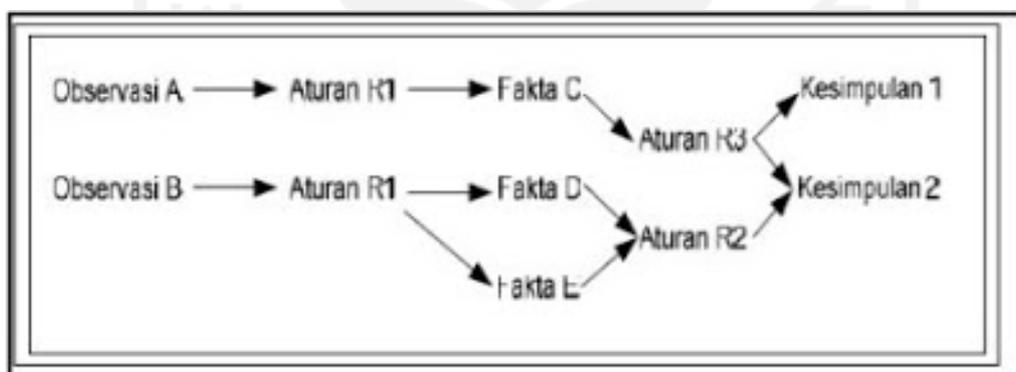
Cara kerja SWI - Prolog yaitu ketika pengguna mengetik suatu pertanyaan, Prolog mencari melalui klausa-klausa kemudian inti dari klausa tersebut disesuaikan dengan pertanyaan yang diajukan. Apabila suatu klausa sesuai dengan pertanyaan yang diajukan, klausa tersebut digunakan untuk mendukung terbentuknya sebuah kesimpulan. Sebaliknya, apabila Prolog gagal untuk menemukan klausa yang sesuai dengan pertanyaan maka Prolog akan kembali kepada kesimpulan (Ian et al., 2014).

Aturan-aturan yang sudah dikumpulkan dan dianalisis *dikodekan* dalam bahasa pemrograman Prolog sehingga menjadi satu basis pengetahuan. Prolog digunakan sebagai bahasa pemrograman logika yang biasanya ada bidang sistem berbasis pengetahuan (*knowledge base*). Prolog bekerja berdasarkan pada fakta dan aturan (*rule*) yang ada, dan ini sesuai dengan pembagian harta warisan sesuai dengan hukum Islam yang juga berbentuk aturan-aturan (*rule*).

Software SWI-Prolog tidak memungkinkan untuk melihat *kode* dan ketika di *run* atau dalam Prolog *consult knowledge base*. Penulis menggunakan *edior online* berbasis web SWISH-Prolog. Dalam SWISH-Prolog *kode* yang dibuat baik berupa *error*, *input*, atau *output* dapat terlihat secara keseluruhan, sehingga memudahkan penulis dalam proses *kodefikasi*.

2.10 Forward Chaining

Forward chaining atau juga biasa disebut dengan pelacakan ke-depan merupakan proses penalaran dengan pendekatan yang dimotori data (*data-driven*). Pendekatan ini merupakan pelacakan ke-depan sesuai dengan aturan IF-THEN dimulai dari data yang ada lalu menelusuri aturan hingga terbentuknya suatu kesimpulan.



Gambar 2.2 Inferensi *Forward Chaining*

Forward chaining merupakan salah satu metode *reasoning* dalam *inference engine*. *Forward chaining* dapat dideskripsikan sebagai *aplikasi* pengulangan dari *modus ponens* (Akil, 2017). *Forward chaining* menggunakan teknik pelacakan ke-depan. Pelacakan akan dimulai dari data yang ada terlebih dahulu lalu menelusuri *rule* hingga tercapai penyelesaian akhir. Jika permasalahan dimulai dengan rekaman informasi di awal, maka *Forward chaining* sangat baik untuk digunakan (Fanny et al., 2017).

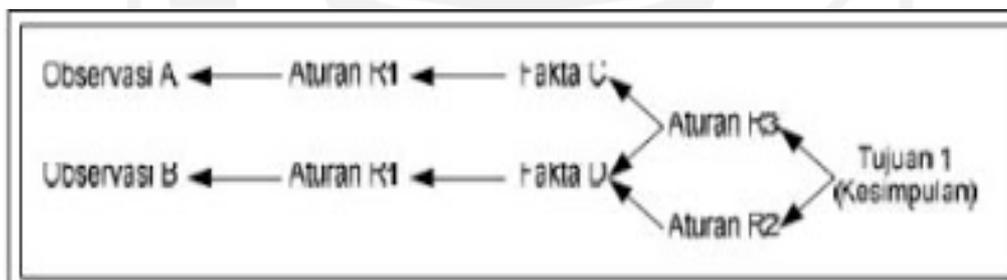
Sebuah *goal* merupakan hasil dari *inferensi Forward chaining*, karena *Forward chaining* melakukan penalaran dari data yang ada atau masalah kepada solusinya (Fanny et al., 2017). Jika berhasil menemukan sebuah solusi maka proses pencarian akan berhenti.

Forward chaining adalah metode pencarian / penalaran kesimpulan yang berdasarkan pada data atau fakta yang ada menuju kesimpulan, penelusuran dimulai dari fakta yang ada lalu bergerak maju melalui premis-premis untuk menuju ke kesimpulan. *Forward chaining* melakukan pencarian dari suatu masalah kepada solusinya (Nursalam et al., 2016).

Forward chaining merupakan metode *inferensi* yang melakukan penalaran dari suatu masalah kepada solusinya. Jika klausa premis sesuai dengan situasi (TRUE.), maka proses akan dinyatakan sebagai *konklusi* (Refni et al., 2019).

2.11 Backward Chaining

Backward chaining atau disebut juga dengan pelacakan ke-belakang merupakan kebalikan dari *Forward chaining*. Pelacakan ini dimotori dengan tujuan (*goal-driven*) (Brata et al., 2012). Tujuan digunakan sebagai titik awal pelacakan, selanjutnya mesin akan mencari aturan yang berhubungan dengan tujuan tersebut untuk mendapatkan kesimpulan.



Gambar 2.3 Inferensi *Backward Chaining*

2.12 Kelebihan dan Kekurangan Sistem Pakar

Setiap sesuatu memiliki kelebihan dan kekurangannya, tidak terkecuali sistem pakar sekalipun. Sistem pakar memiliki kelebihan dan kekurangan sendiri, menurut (Listiyono, 2008) kelebihan dan kekurangan sistem dijelaskan dalam poin berikut :

a. Kelebihan

1. Sistem pakar dapat memberikan jawaban yang sesuai kepada *user* pada saat proses konsultasi. Ini dikarenakan sistem pakar dirancang oleh pengetahuan seorang pakar yang bersifat logis. Sistem pakar juga memberikan jawaban secara konsisten selama bidang domain pengetahuan tidak mengalami perkembangan yang signifikan. Sistem pakar dapat memberikan keputusan secara lebih efisien dan lebih baik.

2. Menggunakan sistem pakar dapat menghemat waktu. Karena sistem pakar yang sudah siap digunakan akan selalu bisa dimanfaatkan untuk mencari solusi. Hal ini menjadi sangat efisien tatkala seorang pakar yang ingin dimintai tolong tidak bisa.
 3. Pengetahuan seorang pakar dapat disimpan dan dapat dimanfaatkan untuk jangka waktu lama. Pengambilan keputusan dalam perusahaan akan tetap bisa dilakukan tatkala seorang yang menguasai domain tersebut sudah tidak bekerja lagi. Karena pengetahuan pakar sudah disimpan di dalam komputer, sehingga perusahaan dapat berjalan normal ketika menghadapi suatu masalah dan membutuhkan pengetahuan dari seorang pakar tersebut.
- b. Kekurangan
1. Sistem pakar terdiri dari fakta dan aturan yang membentuk suatu *knowledge base*. Sistem pakar akan sangat baik ketika pengetahuan dari domain tidak berubah ubah, sistem pakar akan memberikan solusi secara konsisten. Sebaliknya, jika pengetahuan yang dijadikan sistem pakar cepat berubah ubah, maka akan semakin sering fakta dan aturan-aturan yang ada di *knowledge base* juga harus diubah. Hal ini akan menjadi kurang efisien dalam proses pengembangannya
 2. Sistem pakar tidak dapat memberikan solusi secara subjektif. Solusi yang diberikan oleh sistem pakar bersifat logis sesuai dengan fakta dan aturan yang ada.
 3. Tidak seperti *koding* biasanya, sistem pakar di bangun menggunakan aturan-aturan *if – then* untuk menjadi suatu *knowledge base*.

2.13 Black-box Testing

Black-box testing juga disebut sebagai pengujian *fungsiional*, yang artinya pengujian ini dilakukan dengan merancang kasus uji berdasarkan kasus yang ada. *User* juga tidak diperkenankan untuk mengakses kode (*syntax*). Pengujian hanya berfokus pada *output* dari *input* yang di-masukan (Nidhra et al., 2012).

Pengujian Fungsiional dari program perangkat lunak atau sistem yang diuji diamati sebagai "*Black-box*". Pilihan kasus uji untuk pengujian fungsiional didasarkan pada kebutuhan atau *spesifikasi* desain entitas perangkat lunak yang diuji. Contoh hasil yang diharapkan terkadang disebut uji *nubuat*, termasuk persyaratan / spesifikasi *desain*, nilai yang dihitung dengan tangan, dan hasil simulasi. Pengujian *fungsiional* terutama berfokus pada eksternal tingkah laku dari entitas perangkat lunak (Nidhra et al., 2012).

Pengujian *Black-box* adalah pengujian yang *memverifikasi* hasil eksekusi aplikasi berdasarkan masukan yang diberikan (*data uji*) untuk memastikan fungsional dari aplikasi sudah sesuai dengan persyaratan (*requirement*) (Priyaunga et al., 2020).

Black box testing biasanya digunakan untuk mencari kesalahan kesalahan seperti :

- a. *Method* atau fungsi yang salah atau bahkan tidak ada.
- b. *User interface error*.
- c. Adanya eror dalam struktur data dan akses menuju basis data itu sendiri.
- d. Kurang baiknya performa perangkat lunak, ini dikarenakan adanya eror pada performa
- e. Kesalahan dalam *inisialisasi* dan terminasi.

Pengujian *Black-box* terdapat beberapa jenis, berikut jenis-jenis pengujian *black-box* yang paling sering digunakan :

- a. *Equivalence class partitioning*

Equivalence class partitioning merupakan sebuah teknik pengujian pada *black-box testing*. Teknik pengujian ini menggunakan *form* untuk menampung semua data masukan. Teknik ini berfokus untuk membagi data ke dalam domain kelas-kelas untuk diuji valid atau tidak valid. Proses pengujian dimulai dari membuat kasus uji (*test case*) dari perangkat lunak yang akan diuji. Selanjutnya pengujian dilakukan dengan mengeksekusi setiap kasus uji yang sudah dibuat sebelumnya. Pada tahap terakhir adalah proses mendokumentasikan hasil pengujian apakah sudah valid atau belum (Priyaunga et al., 2020).

- b. *Boundary value analisis*

Boundary value analisis merupakan salah satu teknik pengujian dari *black-box testing*. Pengujian ini dilakukan dengan membuat suatu nilai batas bawah dan nilai batas atas pada perangkat lunak yang akan diuji. *Boundary value analisis* digunakan untuk mencari kesalahan dalam masukan. Secara garis besar *Boundary value analisis* digunakan untuk menjaga variabel pada nilai normal atau rata-rata dan *mengijinkan* agar *variable* lain berada pada nilai *ekstrimnya* (Mustaqbal et al., 2015).

2.14 *White-box Testing*

White-box testing merupakan suatu metode pengujian dalam suatu perangkat lunak. *White-box testing* berfokus pada pengecekan *internal kode*. Jika pada saat pengetesan *output*

yang diharapkan tidak sesuai atau salah maka perlu dilakukan pengecekan ulang. Proses *testing* terus dilakukan hingga hasil keluaran sesuai seperti yang diinginkan (Mustaqbal et al., 2015).

White-box testing atau biasa juga disebut pengujian *struktural* merupakan pengujian yang merancang kasus uji berdasarkan informasi yang diperoleh dari kode sumber. Seorang penguji merupakan orang yang mengetahui seperti apa kode itu dan menulis kasus uji dengan menjalankan metode dengan parameter tertentu (Nidhra et al., 2012).

Menurut (Mustaqbal et al., 2015) secara umum pengujian *white-box testing* dilakukan dengan beberapa tahapan. Tahapan - tahapan tersebut dapat dibagi menjadi 3 tahap, yaitu :

- a. Melakukan pengujian secara keseluruhan untuk *syntax* yang berbentuk *logic (if-then)*
- b. Melakukan pengujian secara keseluruhan untuk *syntax* yang berbentuk perulangan (*loop*) yang sudah ditentukan nilai – nilai batasnya.
- c. Melakukan pengujian kepada struktur data yang bersifat internal dan dijamin validitasnya.

White-box testing baik jika digunakan untuk menguji percabangan (*decision*), perulangan (*loop*), dan struktur data yang bersifat internal. Kelebihan dari *White-box testing* adalah untuk mencari kesalahan logika, *ketidaksesuaian* asumsi, dan kesalahan pengetikan (Mustaqbal et al., 2015).

2.15 Rekap Penelitian Terdahulu

Berdasarkan penjabaran dari kelima penelitian terdahulu, bahwa topik penelitian adalah “Membuat Sistem Pakar Untuk Mempermudah Perhitungan Pembagian Waris Sesuai Dengan Syariat Islam”. Di bawah ini adalah rekap penelitian terdahulu pada tabel 2.1 dan rekap penelitian pada tabel 2.2

Tabel 2.2 Rekap Penelitian Terdahulu

No	Peneliti	Tujuan Penelitian
1	Desi Minarni, Indra Griha Tofik Isa, dan Asriyanik(2017)	Untuk membuat sistem pakar pembagian warisan menurut syariat Islam berbasis web
2	Muhammad Faqihuddin(2016)	Untuk membuat sistem pakar pembagian warisan menurut syariat Islam berbasis web

3	Yosep Septiana, Dede Kurniadi, dan Asri Mulyani(2017)	Untuk membuat sistem pendukung keputusan (SPK) dalam pembagian warisan menurut syariat Islam
4	Nadya Andhika Putri, dan Subhan hartanto(2020)	Untuk menganalisis dan membuat sistem pakar untuk pembagian warisan menurut syariat Islam
5	Harmen, Omar Pahlevi, dan Tri Santoso(2019)	Untuk membuat sistem pakar pembagian warisan menurut syariat Islam berbasis web

Dari ke-lima penelitian terdahulu, masing-masing penelitian meneliti pembagian waris sesuai syariat Islam. Metode yang paling sering digunakan adalah metode *Forward chaining*. Metode *Forward chaining* merupakan metode pelacakan solusi ke-depan. Pelacakan solusi akan dimulai dari data yang ada terlebih dahulu lalu mencari melalui aturan-aturan yang ada hingga terbentuknya sebuah solusi. Hal ini serupa dengan kasus pembagian waris dalam Islam yang dimulai dari data yang ada terlebih dahulu (*ahli waris*) lalu melalui aturan-aturan hingga terbentuknya sebuah solusi (*bagian waris*).

Tabel 2.3 Rekap Persamaan dan Perbedaan Penelitian

No	Peneliti	Persamaan	Perbedaan
1	Desi Minarni, Indra Griha Tofik Isa, dan Asriyanik(2017)	- Aplikasi bagi waris Islam - <i>Forward Chaining</i>	- Berbasis Web - <i>PHP & MySQL</i>
2	Muhammad Faqihuddin(2016)	- Aplikasi bagi waris Islam - <i>Forward Chaining</i>	- Berbasis Web - <i>PHP & MySQL</i>

3	Yosep Septiana, Dede Kurniadi, dan Asri Mulyani(2017)	<ul style="list-style-type: none"> - SPK pembagian waris 	<ul style="list-style-type: none"> - <i>evolutionary software process models methode</i> - Berbasis <i>deskop</i>
4	Nadya Andhika Putri, dan Subhan hartanto(2020)	<ul style="list-style-type: none"> - Sistem pakar pembagian waris menurut Islam - <i>Forward Chaining</i> 	<ul style="list-style-type: none"> - Bahasa pemrograman Java - Berbasis <i>deskop</i>
5	Harmen , Omar Pahlevi, dan Tri Santoso(2019)	<ul style="list-style-type: none"> - Aplikasi bagi waris Islam - <i>Forward Chaining</i> 	<ul style="list-style-type: none"> - Berbasis Web - <i>PHP &MySQL</i>

Terdapat beberapa persamaan dan perbedaan penelitian yang dilakukan oleh penulis dan penelitian terdahulu. Dari ke-lima jurnal pada tabel 2.2 persamaan terdapat pada hasil penelitian (Sistem pakar pembagian waris) dan metode yang digunakan (*Forward chaining*). Terdapat pula perbedaan pada penelitian terdahulu, aplikasi yang dibangun berbasis berupa *web* dan *deskop*. Hasil penelitian yang penulis lakukan menghasilkan sebuah *knowledge base* dari pembagian waris, dan ini masih berbentuk *algoritme*.

BAB III METODELOGI PENELITIAN

3.1 Metode Penelitian

Metode yang penulis gunakan adalah metode penelitian terapan atau yang biasa disebut dengan *Applied Research*. Ilmu baru bukanlah tujuan dari dilakukannya penelitian ini tetapi lebih kepada penerapan aplikasi dari ilmu murni. Pemecahan masalah dan pengembangan aplikasi dengan tujuan memudahkan suatu masalah (*praktis*) menjadi fokus utama dalam penelitian terapan (Nugrahani, 2016).

Penelitian terapan merupakan penelitian yang tidak hanya berfokus dalam pemecahan masalah (*problem solving*), namun juga untuk pengembangan aplikasi dari masalah itu sendiri, sehingga nantinya aplikasi yang sudah dibuat dapat digunakan oleh masyarakat dan dapat bermanfaat. Dari pengertian dan tujuan tersebut metode penelitian terapan merupakan penelitian yang sesuai untuk digunakan penulis guna tercapainya tujuan penulis.

Secara garis besar penelitian terapan terbagi ke dalam beberapa langkah, yang meliputi identifikasi masalah (*problem identification*), perancangan (*design*), implementasi (*implementasi*), dan pengujian (*testing*). Keempat langkah dapat dilihat dalam gambar 3.1.



Gambar 3.1 Alur Penelitian Terapan

Dari gambar 3.1 dapat dilihat bahwa pada tahapan pengujian (*testing*) dapat berulang, ini menandakan bahwa jika hasil yang didapat sudah sesuai dengan kebutuhan perangkat lunak dan sudah menjawab atas persoalan yang dihadapi, maka penulis dapat memutuskan penelitian telah berhasil. Jika masih tidak sesuai, itu menandakan bahwa masih dibutuhkan revisi dalam proses implementasi hingga hasil atau keluaran yang didapat sesuai dengan kebutuhan sistem.

3.2 Identifikasi Masalah

3.2.1 Jenis Data

Data yang digunakan dalam penelitian ini merupakan jenis data kualitatif, data kualitatif merupakan data yang tidak berupa angka, namun berupa kata-kata (Hasibuan, 2007). Yang

berarti dalam penelitian ini adalah perkataan atau perintah Allah SWT yang dituangkan ke dalam kitab suci Al-Quran.

Dalam ayat-ayat pembagian waris terdapat aturan-aturan bagian harta untuk setiap ahli waris, data ini merupakan data kualitatif. Untuk menganalisis dibutuhkan proses konversi dari data kualitatif ke dalam data kuantitatif, proses ini dinamakan proses kuantifikasi (Hasibuan, 2007). Perubahan ini biasanya dilakukan dengan memberi skor tertentu, namun dalam kasus pembagian waris sesuai dengan hukum Islam sudah jelas bagian-bagian untuk setiap ahli warisnya, sehingga pemberian skor tidak diperlukan dalam proses kuantifikasi.

3.2.2 Sumber Data

Sumber data yang digunakan oleh penulis adalah data sekunder. Data yang digunakan oleh penulis berasal dari Al-Quran dan Hadits, karena hukum yang tertera di-sana sudah sangat jelas diturunkan oleh Allah SWT dan bersifat mutlak. Penulis juga melengkapi data dengan melakukan *studi literatur* dari jurnal di *internet*. Contoh data yang digunakan adalah aturan-aturan pembagian warisan sesuai hukum Islam.

مَا تَرَكَ وَإِنْ كَانَتْ وَاحِدَةً فَلَهَا النِّصْفُ وَلِأَبَوَيْهِ لِكُلِّ وَاحِدٍ مِّنْهُمَا السُّدُسُ مِمَّا تَرَكَ إِنْ كَانَ لَهُ وَلَدٌ فَإِنْ لَمْ يَكُنْ لَهُ وَلَدٌ وَوَرِثَهُ آبَاؤُهُ فَلِأُمَّهِ الثُّلُثُ فَإِنْ كَانَ لَهُ إِخْوَةٌ فَلِأُمَّهِ السُّدُسُ مِنْ بَعْدِ وَصِيَّةٍ يُوصِي بِهَا أَوْ دَيْنٍ آبَاؤُكُمْ وَأَبْنَاؤُكُمْ لَا تَدْرُونَ أَيُّهُمْ أَقْرَبُ لَكُمْ نَفَعًا فَرِيضَةٌ مِنَ اللَّهِ إِنَّ اللَّهَ كَانَ عَلِيمًا حَكِيمًا (١١)

QS An-Nissa Ayat 11

Artinya :

Allah menyari'atkan kepadamu tentang (pembagian warisan untuk) anak-anakmu, yaitu bagian seorang anak laki-laki sama dengan bagian dua orang anak perempuan. Jika anak itu semuanya perempuan yang jumlahnya lebih dari dua, maka bagian mereka dua pertiga dari harta yang ditinggalkan. Jika anak perempuan itu seorang saja, maka dia memperoleh setengah (harta yang ditinggalkan). Untuk kedua ibu-bapak, bagian masing-masing *seperenam* dari harta yang ditinggalkan, jika yang meninggal itu mempunyai anak. Jika orang yang meninggal tidak mempunyai anak dan dia

diwarisi oleh kedua ibu bapaknya (saja), maka ibunya mendapat sepertiga. Jika yang meninggal itu mempunyai beberapa saudara, maka ibunya mendapat *seperenam*. (Pembagian-pembagian tersebut di atas) setelah dipenuhi wasiat yang dibuatnya atau (dan) setelah dibayar hutangnya. (Tentang) orang tuamu dan anak-anakmu, kamu tidak mengetahui siapa di antara mereka yang lebih banyak manfaatnya bagimu. Ini adalah ketetapan Allah. Sungguh, Allah Maha Mengetahui lagi Maha bijaksana.

لَكُمْ نِصْفُ مَا تَرَكَ أَزْوَاجُكُمْ إِنْ لَمْ يَكُنْ لَهُنَّ وَلَدٌ فَإِنْ كَانَ لَهُنَّ وَلَدٌ فَلَكُمْ الرُّبْعُ
 مِمَّا تَرَكَنَّ مِنْ بَعْدِ وَصِيَّةٍ يُوصِيَنَّ بِهَا أَوْ دَيْنٍ وَلَهُنَّ الرُّبْعُ مِمَّا تَرَكَتُمْ إِنْ لَمْ يَكُنْ
 لَكُمْ وَلَدٌ فَإِنْ كَانَ لَكُمْ وَلَدٌ فَلَهُنَّ النُّصْبُ مِمَّا تَرَكَتُمْ مِنْ بَعْدِ وَصِيَّةٍ تُوصُونَ
 بِهَا أَوْ دَيْنٍ وَإِنْ كَانَ رَجُلٌ يُورَثُ كَلَالَةً أَوْ امْرَأَةٌ وَلَهُ أَخٌ أَوْ أُخْتٌ فَلِكُلِّ وَاحِدٍ
 مِنْهُمَا السُّدُسُ فَإِنْ كَانُوا أَكْثَرَ مِنْ ذَلِكَ فَهُمْ شُرَكَاءُ فِي التُّلْثِ مِنْ بَعْدِ وَصِيَّةٍ
 يُوصَى بِهَا أَوْ دَيْنٍ غَيْرِ مُضَارٍّ وَصِيَّةً مِنَ اللَّهِ وَاللَّهُ عَلِيمٌ حَلِيمٌ (١٢)

QS An-Nissa Ayat 12

Artinya :

Bagianmu (suami-suami) adalah seperdua dari harta yang ditinggalkan oleh istri-istri *mu*, jika mereka tidak mempunyai anak. Jika mereka (istri-istrimu) itu mempunyai anak, maka kamu mendapat seperempat dari harta yang ditinggalkannya setelah dipenuhi wasiat yang mereka buat atau (dan) setelah dibayar hutangnya. Para istri memperoleh seperempat harta yang kamu tinggalkan jika kamu tidak mempunyai anak. Jika kamu mempunyai anak, maka para istri memperoleh seperdelapan dari harta yang kamu tinggalkan setelah dipenuhi wasiat yang kamu buat atau (dan) setelah dibayar hutang-hutangmu. Jika seseorang meninggal, baik laki-laki maupun perempuan yang tidak meninggalkan ayah dan tidak meninggalkan anak, tetapi mempunyai seorang saudara laki-laki (seibu) atau seorang saudara perempuan (seibu), maka bagi masing-masing dari kedua jenis saudara itu *seperenam* harta. Jika saudara-saudara seibu itu lebih dari seorang, maka mereka bersama-sama dalam bagian yang sepertiga itu, setelah dipenuhi wasiat yang dibuatnya atau (dan) setelah dibayar hutangnya dengan tidak menyusahkan (kepada ahli waris). Demikianlah ketentuan Allah. Allah Maha Mengetahui lagi Maha Penyantun.

يَسْتَفْتُونَكَ قُلِ اللَّهُ يُفْتِيكُمْ فِي الْكَلَالَةِ إِنْ امْرُؤٌ هَلَكَ لَيْسَ لَهُ وَلَدٌ وَلَهُ
 أُخْتٌ فَلَهَا نِصْفُ مَا تَرَكَ وَهُوَ يَرِثُهَا إِنْ لَمْ يَكُنْ لَهَا وَلَدٌ فَإِنْ كَانَتَا

اٰثْنَتَيْنِ فَلَهُمَا الثُّلُثَانِ مِمَّا تَرَكَ وَاِنْ كَانُوْا اِخْوَةً رِّجَالًا وَّنِسَاءً فَلِلَّذَكَرِ مِثْلُ
حَظِّ الْاُنثَيَيْنِ يُبَيِّنُ اللهُ لَكُمْ اَنْ تَضِلُّوْا وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيْمٌ (١٧٦)

QS An-Nissa Ayat 176.

Artinya :

Mereka meminta fatwa kepadamu (tentang kalalah). Katakanlah, "Allah memberi fatwa kepadamu tentang kalalah (yaitu), jika seorang meninggal dunia, dan dia tidak mempunyai anak (dan ayah) tetapi mempunyai saudara perempuan, maka bagi saudaranya yang perempuan itu seperdua dari harta yang ditinggalkannya, dan saudaranya yang laki-laki mewarisi (seluruh harta saudara perempuan), jika dia tidak mempunyai anak. Jika saudara perempuan itu dua orang, maka bagi keduanya dua pertiga dari harta yang ditinggalkan. Jika mereka (ahli waris itu terdiri dari) saudara-saudara laki dan perempuan, maka bagian seorang saudara laki-laki sama dengan bagian dua saudara perempuan. Allah menerangkan (hukum ini) kepadamu, agar kamu tidak sesat. Allah Maha mengetahui segala sesuatu.

3.2.3 Analisis Perangkat Lunak

- a. *Sublime text 3*, digunakan sebagai *editor* untuk *source code* dengan *extensi .pl*.
- b. *SWI-Prolog*, digunakan untuk meng-*consult* basis data dan menanyakan *query* kepada *sistem Prolog*.
- c. *Google Chrome*, digunakan sebagai *browser*.
- d. *SWISH-Prolog*, digunakan sebagai *editor online* berbasis *web* untuk membuat *source code*.

3.2.4 Analisis Perangkat Keras

Dalam pembuatan *project* ini dibutuhkan *hardware* yang berguna untuk mempermudah dalam pembuatan *sistem*, perangkat keras yang digunakan di-antaranya adalah :

- a. Laptop
- b. Wifi

3.3 Perancangan

Dalam melakukan perancangan *sistem*, peneliti menggunakan alat bantu atau *tools* agar mempermudah dalam pembuatan perangkat lunak itu sendiri. Alat bantu yang digunakan adalah Diagram Alir (*Data Flow Diagram /DFD*). Selanjutnya adalah dengan perancangan *rule-rule* yang ada dalam ayat Al-Quran ke dalam bentuk tabel.

3.3.1 *Input dan output*

a. *Data Input*

Data yang masukan yang di dalam *project* ini adalah berupa *atom*. *Atom* dalam *project* ini merupakan orang-orang yang berhak menerima harta warisan (*Ahli Waris*). Ahli waris sendiri terdiri dari 15 orang dari pihak laki-laki dan 11 orang dari pihak perempuan. Data akan disajikan dalam bentuk tabel agar lebih memudahkan untuk dibaca.

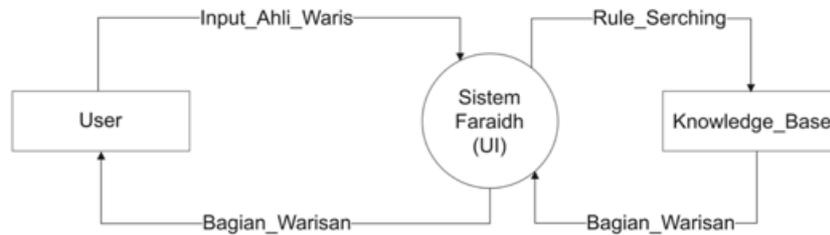
b. *Data Output*

Dalam Prolog *input user* dilakukan di dalam *query*, pada tahap *input user* menanyakan pertanyaan kepada Prolog. Hal ini berhubungan dengan cara kerja Prolog. Prolog akan mencari dari fakta-fakta yang ada terlebih dahulu kemudian mencari melalui aturan-aturan yang tersedia dalam basis pengetahuan. Jika Prolog menemukan solusi yang sesuai dengan *input-an user* maka akan menghasilkan nilai keluaran berupa solusi. Jika Prolog gagal menemukan solusi dari *input-an user* maka Prolog akan menghasilkan nilai *false* sebagai keluaran.

Perancangan aturan dengan menggunakan Prolog akan menghasilkan basis pengetahuan. Dalam basis pengetahuan terdapat fakta dan juga aturan-aturan yang akan menjadi sebuah sistem pakar (Ian et al., 2014). Pada tahap akhir *user* perlu *me*-masukan *query* ke dalam sistem Prolog untuk menanyakan bagian dari setiap ahli waris.

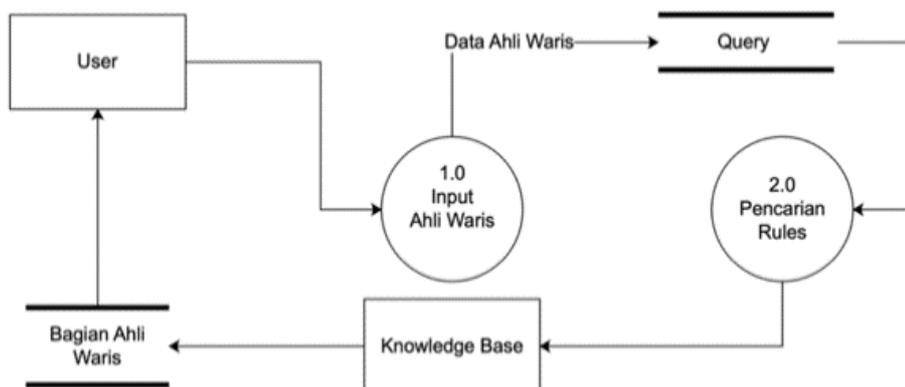
3.3.2 *Rancangan Sistem*

Pembuatan *data flow diagram* bertujuan untuk menggambarkan hubungan antara *entity* luar, masukan (*input*) dan keluaran (*output*). *Data flow diagram* yang digunakan adalah data DFD *Lv 0*. Proses yang terjadi sangat sederhana, sistem digambarkan dengan lingkaran yang berarti adalah sistem Prolog sebagai mesin inferensi. Terdapat dua *external entity* yaitu *user* dan juga *knowledge base*. *Knowledge base* digunakan sebagai pengganti *external entity* yaitu seorang pakar. Gambar dari DFD *Lv 0* dapat dilihat dalam gambar 3.2.



Gambar 3.2 Data Flow Diagram Lv 0

Pembuatan *Data Flow Diagram*(DFD) untuk sistem *Faraidh* tidak terlalu sulit, sehingga cukup menurunkan satu tingkat untuk membuat DFD . Hal ini dikarenakan proses yang terjadi cukup sederhana. *User* akan diminta untuk *meng-input* siapa saja ahli waris yang ada lalu data yang di *input* harus sudah berbentuk query. *Query* yang sudah dibuat akan melakukan pencarian solusi pada *knowledge base* yang sudah dibuat. Hasilnya solusi yang didapat dari *knowledge base* akan menjadi keluaran dan dikirimkan kembali ke *user*. Proses tersebut dapat digambarkan seperti pada gambar 3.3.



Gambar 3.3 Data Flow Diagram Lv1

3.3.3 *User Interface*

Prolog tidak mendukung adanya *kapabisilas* grafis, sehingga tidak memungkinkan dibuatnya *user interface* yang *me-mumpuni*. Prolog mudah untuk dikolaborasikan dengan aplikasi grafis lainnya sebagai alternatif.

Bahasa pemrograman *eksternal* yang populer untuk dikolaborasikan dengan Prolog diantaranya adalah *Java*, *Visual Basic* dan *Delphi*. Prolog akan terhubung dengan sistem lainnya dengan menggunakan *pipes* ataupun mekanisme komunikasi lainnya.

3.3.4 *Rancangan Rules*

Dari ayat Al-Quran surat *An-Nisa* ayat 11 *s.d* 12 dan ayat *An-Nisa* ayat 176, maka aturan yang berbentuk paragraf akan disajikan dalam bentuk tabel. Hal ini dilakukan agar memudahkan dalam proses *pengkodean* nantinya.

Perancangan *rules* akan dibagi menjadi tiga kategori, yaitu ahli waris utama, ahli waris kedua dan ahli waris ketiga. *Rules* ini sudah tercantum dalam sumber data yang digunakan. Pada sumber data terdapat beberapa *rules* untuk satu ahli waris.

3.4 *Pengujian*

Setelah tahap implementasi selesai lalu aturan-aturan sudah dikodekan ke dalam bahasa pemrograman Prolog, lalu akan dilakukan pengujian. Pengujian dilakukan dengan metode pengujian *White-box testing*.

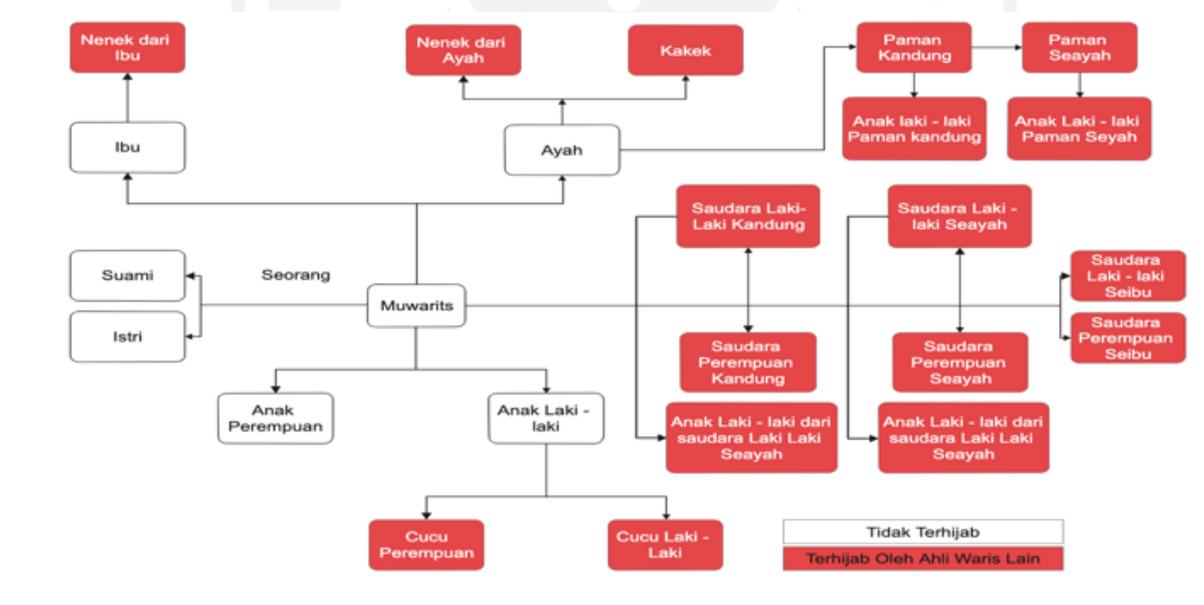
White-box testing digunakan untuk menguji alur program. Karena basis pengetahuan yang telah dibuat berisikan aturan-aturan dan *decision*, maka pengujian menggunakan *White-box testing* sangat efektif untuk dilakukan.

Pengujian akan dilakukan dengan membuat tabel *test case* terlebih dahulu, lalu membuat kasus uji untuk setiap kategori ahli warisnya. Setelah itu kasus uji akan di coba satu per satu untuk dilakukan pengujian. Jika hasil yang dikeluarkan sudah benar maka pengujian akan selesai dan jika hasil pengujian masih mengembalikan nilai *false* maka akan diadakan perbaikan pada *rules*.

BAB IV HASIL DAN PEMBAHASAN

4.1 Pembahasan

Pembagian waris dalam Islam diatur dalam Q.S *An-Nisa* ayat 11 s.d 12 dan *An-Nisa* ayat 176 . Di dalam ayat tersebut terdapat aturan-aturan pembagian waris beserta besarnya bagian yang didapatkan oleh ahli waris. Pembagian waris dalam Islam terdiri dari ahli waris dari keluarga utama yang terdiri dari ayah, ibu, suami atau istri, anak laki-laki dan anak perempuan. Terdapat 15 orang dari pihak laki-laki dan 11 orang dari pihak perempuan. Jumlah tersebut Allah sendiri yang menentukannya di dalam Al-Quran. Ahli waris yang berhak mendapatkan dapat digambarkan di dalam diagram ahli waris dalam gambar 4.1.



Gambar 4.1 Diagram Ahli Waris

Agar lebih memudahkan Ahli Waris disajikan dalam tabel ahli waris. Tabel Ahli waris terdiri dari ahli waris laki-laki dan ahli waris perempuan.

Tabel 4.1 Ahli Waris Laki-laki

No	Ahli Waris Laki-laki	Sinonim
1	Anak laki-laki	Putra, ananda, anak
2	Cucu laki-laki	Anak dari anak, keturunan ketiga
3	Ayah	Bapak, <i>abi</i> , ayahanda
4	Kakek	Aki, datuk, eyang, mbah
5	Suami	Junjungan
6	Saudara kandung laki-laki	Kakak, adik
7	Saudara seayah laki-laki	Saudara tiri
8	saudara seibu laki-laki	Daudara tiri
9	Anak laki-laki dari saudara kandung laki-laki	Saudara sepupu
10	Anak laki-laki dari saudara seayah	Saudara sepupu
11	Paman / Om	Om
12	Paman tiri /Om	Om
13	Anak laki- laki paman kandung	Sepupu
14	Anak laki-laki paman tiri	Sepupu tiri
15	mu,tiq(seorang lelaki yg membebaskan budak)	-

Tabel 4.2 Ahli Waris Perempuan

No	Ahli Waris Perempuan	Sinonim
1	Ibu	Bunda, mama
2	Anak perempuan	Putri, gadis, dara
3	Cucu perempuan	Adan dari anak, keturunan ketiga
4	Nenek dari ibu	Oma, andung, emak
5	Nenek dari ayah	Oma, andung, emak
6	Saudara perempuan sekandung	Kakak, adik

7	Saudara perempuan seapak	Kakak tiri, adik tiri
8	Saudara perempuan seibu	Kakak tiri, adik tiri
9	Istri	Bini
10	buyut	eyang
11	<i>Mu'tiqah</i>	-

Dari diagram ahli waris pada gambar 4.1 ahli waris dapat kita kelompokkan menjadi tiga bagian. Ahli waris utama merupakan ahli waris yang tidak ter-hijab siapa pun. Ahli waris kedua merupakan ahli waris yang dapat ter-hijab karena hubungan orang-tua atau anak. Ahli waris ketiga merupakan Ahli waris yang dapat ter-hijab jika ahli waris pertama dan kedua tidak ada. Berikut disajikan ahli waris pertama dalam bentuk tabel 4.3.

Tabel 4.3 *Rules* Ahli Waris Utama

<i>No</i>	Ahli Waris	<i>Rules</i>
1	Ayah (1)	If seseorang meninggal tidak memiliki anak <i>then</i> 1/2
2	Ayah (2)	If seseorang meninggal meninggalkan anak 1/6
3	Ibu (1)	If seseorang meninggal tidak memiliki anak <i>then</i> 1/3
4	Ibu (2)	If seseorang meninggal meninggalkan anak 1/6
5	Suami (1)	If seseorang meninggal tidak memiliki anak <i>then</i> ½
6	Suami (2)	If seseorang meninggal meninggalkan anak ¼
7	Istri (1)	If seseorang meninggal tidak memiliki anak <i>then</i> ¼
8	Istri (2)	If seseorang meninggal meninggalkan anak <i>then</i> 1/8
9	Anak Laki-laki (1)	If seseorang meninggal tidak mempunyai anak perempuan <i>then</i> 1(sisa)

10	Anak Laki-laki (2)	If seseorang meninggal memiliki anak perempuan <i>then</i> $2/3$
11	Anak Laki-laki >1	If seseorang meninggal memiliki anak perempuan <i>then</i> $2 \times$ anakperempuan
12	Anak Perempuan (1)	If seseorang meninggal tidak mempunyai anak laki-laki <i>then</i> $1/2$
13	Anak Perempuan (2)	If seseorang meninggal mempunyai anak laki laki <i>then</i> $1/3$
14	Anak Perempuan >1	If seseorang meninggal mempunyai anak laki laki <i>then</i> $2/3$

Jika salah satu dari Ahli waris utama kosong, kecuali suami atau istri, maka dapat diwariskan lagi kepada leluhur ataupun generasi di-bawahnya. Dalam tabel 4.4 disajikan aturan Ahli Waris kedua yang masih bisa mendapatkan kesempatan menerima harta warisan, namun akan ter-hijab jika ahli waris utama ada.

Tabel 4.4 *Rules* Ahli Waris Kedua

No	Ahli Waris	Rules
1	Kakek (1)	If seseorang meninggal tidak mempunyai anak dan ayah <i>then</i> $1/3$
2	Kakek (2)	If seseorang meninggal mempunyai anak dan tidak ada ayah <i>then</i> $1/6$
3	Nenek dari (1)	If seseorang meninggal tidak mempunyai anak dan ibu <i>then</i> $1/6$
4	Nenek(2)	If seseorang meninggal mempunyai anak dan tidak ada ibu <i>then</i> $1/6$
5	Cucu Laki-laki (1)	If seseorang meninggal tidak mempunyai cucu perempuan <i>then</i> 1(sisa)

6	Cucu Laki-laki (2)	If seseorang meninggal memiliki cucu perempuan <i>then</i> 2/3
7	Cucu Laki-laki >1	If seseorang meninggal memiliki anak perempuan <i>then</i> 2 x anakperempuan
8	Cucu Perempuan (1)	If seseorang meninggal tidak mempunyai cucu laki-laki <i>then</i> 1/2
9	Cucu Perempuan (2)	If seseorang meninggal mempunyai cucu laki-laki <i>then</i> 1/3
10	Cucu Perempuan > 1	If seseorang meninggal mempunyai cucu laki-laki <i>then</i> 2/3

Umumnya pembagian waris hanya sampai diturunkan sampai cucu saja, tetapi jika cucu tidak ada dan juga tidak ada pengganti ayah(*kakek*), maka akan diturunkan ke bagian samping diagram. Dalam kasus pembagian waris ini jarang terjadi, namun masih memungkinkan terjadi, sehingga dibuatkan aturan untuk Ahli Waris ketiga pada tabel 4.5

Tabel 4.5 *Rules* Ahli Waris Ketiga

<i>No</i>	Ahli Waris	<i>Rules</i>
1	Saudara kandung Laki-laki (1)	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung perempuan <i>then</i> 1(sisa)
2	Saudara kandung Laki-laki (2)	If seseorang meninggal tidak memiliki cucu dan memiliki saudara kandung perempuan <i>then</i> 2/3
3	Saudara kandung Laki-laki >1	If seseorang meninggal tidak memiliki cucu dan memiliki saudara kandung perempuan <i>then</i> 2 x saudara kandung perempuan
4	Saudara kandung Perempuan (1)	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung laki-laki <i>then</i> 1/2
5	Saudara kandung Perempuan (2)	If seseorang meninggal tidak memiliki cucu dan memiliki saudara kandung laki-laki <i>then</i> 1/3
6	Saudara kandung Perempuan > 1	If seseorang meninggal tidak memiliki cucu dan memiliki saudara kandung laki-laki <i>then</i> 2/3
7	Saudara Seayah Laki-laki (1)	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah perempuan <i>then</i> 1(sisa)

8	Saudara seayah Laki-laki (2)	If seseorang meninggal tidak memiliki cucu dan tidak memiliki saudara kandung dan memiliki saudara seayah perempuan <i>then</i> 2/3
9	Saudara seayah Laki-laki >1	If seseorang meninggal tidak memiliki cucu dan tidak memiliki saudara kandung dan memiliki saudara seayah perempuan <i>then</i> 2 x saudara seayah perempuan
10	Saudara seayah Perempuan (1)	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah laki-laki <i>then</i> 1/2
11	Saudara seayah Perempuan (2)	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan memiliki saudara seayah laki-laki <i>then</i> 1/2
12	Saudara seayah Perempuan > 1	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah laki-laki <i>then</i> 2/3
13	Saudara seibu Laki-laki = 1	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah <i>then</i> 1/6
14	Saudara seibu Laki-laki >1	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah <i>then</i> 2/3
15	Saudara seibu Perempuan = 1	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah <i>then</i> 1/6
16	Saudara seibu Perempuan >1	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah <i>then</i> 2/3
17	Paman Kandung (Om) >= 1	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah dan seibu <i>then</i> 1(sisa)
18	Paman (om) Seayah	If seseorang meninggal tidak mempunyai cucu dan tidak memiliki saudara kandung dan tidak memiliki saudara seayah dan seibu dan paman kandung <i>then</i> 1(sisa)

Apabila tidak ada paman kandung maupun paman tiri, maka anak dari paman kandung ataupun anak dari paman tiri mendapatkan kesempatan untuk mendapatkan warisan, berlaku untuk *rule* yang sama. Apabila dari semua ahli waris tidak ada yang bisa menerima warisan(*tidak ada/ sudah meninggal*), maka tidak ada yang menerima warisan.

4.2 Implementasi Sistem

Dalam implementasi sistem, rancangan yang sudah dijabarkan dalam bab 3 akan dibuat pada bab ini. Fakta dan aturan yang dibuat akan *dikodefikasikan* dengan bahasa pemrograman Prolog. Dalam proses *kodefikasi* data dan aturan yang ada akan disajikan dalam bentuk *list* (struktur data). Dibutuhkan *base case* dan *recursive case* dalam proses implementasi kode. Proses *kodefikasi* dilakukan dengan melakukan *coding* pada *software sublime text 3*.

4.2.1 Fakta dan aturan

Dalam pemrograman menggunakan Prolog, fakta dan aturan merupakan bagian dasar dalam pemrograman logika. Dalam Bab III terdapat fakta dan aturan dalam bentuk tabel. Tabel tersebut merupakan hasil pengolahan data dari ayat-ayat maupun Hadist yang ada. Fakta dan aturan tersebut *dikodefikasi* sehingga menjadi sebuah kesatuan yang disebut *knowledge base*.

4.2.2 Base case dan recursive case

Dalam Prolog tidak mendukung metode penyimpanan data dengan *array* dan digantikan dengan *list*. Dalam *list* data yang disimpan dapat berupa berbagai macam tipe, sebaliknya *array* hanya dapat menampung 1 tipe data saja dalam satu *array*.

```
%recursive case
hitungwaris([L], BL) :-
    hitungwaris(L, L, BL).

%basecase(fact)
hitungwaris([], _, []).
```

Gambar 4.2 Basecase dan Recursive Case

Basecase menjadi sebuah klausa awal dan berupa fakta. Predikat *hitungwaris* merupakan sebuah *list* kosong, *anonymous variable*, dan sebuah *list* kosong.

Recursive case merupakan proses pengisian data ke dalam *list*, yang nantinya akan menjadi sebuah bentuk dari pendefinisian aturan-aturan yang ada. *Recursive case* dalam gambar 4.1 dapat dibaca “L adalah *hitungwaris* dari BL jika predikat *hitungwaris* berisi L dan L dan BL”.

4.2.3 Implementasi Rules

Terdapat *rules* yang sudah didefinisikan dalam tabel aturan. Dalam gambar 4.2 bentuk aturan sama seperti *recursive case*. Hanya saja *anonymous variable* diganti menjadi *variable AhliWaris*. *Variable AhliWaris* digunakan untuk menampung semua data yang ada.

```
%AYAH

%rules ayah jika seseorang meninggal namun tidak punya anak
hitungwaris([ayah|L],AhliWaris,[1/2|BL]) :-
    (\+ada_anak_lakilaki(AhliWaris) ;
    \+ada_anak_perempuan(AhliWaris)),!,
    hitungwaris(L,AhliWaris,BL).

%rules ayah jika seseorang meninggal punya anak
hitungwaris([ayah|L],AhliWaris,[1/6|BL]) :-
    (ada_anak_lakilaki(AhliWaris);
    ada_anak_perempuan(AhliWaris)),!,
    hitungwaris(L,AhliWaris,BL).
```

Gambar 4.3 Rules Ayah

List dalam prolog dibagi menjadi dua bagian, yaitu *head* dan *tail* [*Head* | *Tail*]. Gambar 4.3. menunjukkan bahwa ayah merupakan *head* dari sebuah *list* dan L merupakan *Tail* dari *list* tersebut. Data dari predikat ayah akan disimpan di dalam *list AhliWaris* sebagai koleksi data. Lalu nilai ayah yaitu ½ didefinisikan di *list* terakhir dan disimpan sebagai sebuah *list* yang baru.

4.2.4 Member Checking

Terdapat *decision making* dalam kasus pembagian warisan dalam Islam. *Decision maker* yang ada ketika salah satu ahli waris ada ataupun tidak ada maka bagian untuk ahli waris tersebut juga berbeda, sehingga diperlukannya *member checking* untuk mengecek apakah data ahli waris ada atau tiada dalam *list AhliWaris*.

```

%checking member anak

ada_anak_lakilaki(Y) :- member(anaklakilaki,Y).
ada_anak_lakilaki(Y) :- member(anaklakilaki,Y).
ada_anak_perempuan(Y) :- member(anakperempuan,Y).

%checking member cucu
ada_cucu_lakilaki(Y) :- member(cuculakilaki,Y).
ada_cucu_perempuan(Y) :- member(cucuperempuan,Y).

%checking member orangtua
ada_ayah(Y) :- member(ayah,Y).
ada_ibu(Y) :- member(ibu,Y).

%checking member kakek
ada_kakek(Y) :- member(kakek,Y).

%checking member saudara kandung
ada_saudarakandung_lakilaki(Y) :- member(saudarakandunglakilaki,Y).
ada_saudarakandung_perempuan(Y) :- member(saudarakandungperempuan,Y).

%checking member saudara seayah
ada_saudaraseayah_lakilaki(Y) :- member(saudaraseayahlakilaki,Y).
ada_saudaraseayah_perempuan(Y) :- member(saudaraseayahperempuan,Y).

%checking member Paman
ada_paman(Y) :- member(paman,Y).

```

Gambar 4.4 Member Checking

Member checking terdiri dari ada atau tidaknya keberadaan anak, cucu, orang-tua, saudara kandung, saudara seayah dan paman. Prolog sendiri sudah menyediakan fungsi *member checking*, sehingga penulis hanya tinggal menggunakan saja. Penggunaan *member checking* didefinisikan dengan klausa seperti pada gambar 4.3. *Variable Y* digunakan untuk menampung nilai dari predikat yang ingin dicek keberadaannya dalam koleksi data.

4.2.5 Prolog Cut

Prolog dapat menemukan solusi lebih dari satu, sehingga dibutuhkannya perintah berhenti ketika solusi yang didapatkan sudah sesuai. Untuk melakukan *stop searching* digunakan tanda cut “!”. Tanda cut diletakan setelah semua predikat *member checking* didefinisikan.

```

%rules ayah jika seseorang meninggal namun tidak punya anak
hitungwaris([ayah|L],AhliWaris,[1/2|BL]) :-
    (\+ada_anak_lakilaki(AhliWaris),
    \+ada_anak_perempuan(AhliWaris)),!,
    hitungwaris(L,AhliWaris,BL).

%rules ayah jika seseorang meninggal punya anak
hitungwaris([ayah|L],AhliWaris,[1/6|BL]) :-
    (ada_anak_lakilaki(AhliWaris);
    ada_anak_perempuan(AhliWaris)),!,
    hitungwaris(L,AhliWaris,BL).

```

Gambar 4.5 Cut Pada Rule Prolog

4.3 Pengujian

Setelah tahap implementasi selesai lalu aturan-aturan sudah dikodekan ke dalam bahasa pemrograman Prolog akan diuji. Pengujian dilakukan dengan metode *White-box testing*.

White-box testing digunakan untuk menguji alur program. Karena basis pengetahuan yang telah dibuat berisikan aturan-aturan dan *decision*, maka pengujian menggunakan *White-box testing* sangat efektif untuk dilakukan. Pengujian dilakukan dengan membuat 12 *test case*.

Tabel 4.6 Test Case Ahli Waris Pertama

No	Query	Output	Validasi
1	Hitungwaris([ayah,ibu,suami,anak lakilaki,anakperempuan], AhliWaris).	AhliWaris = [1/6, 1/6, 1/4, 0.667, 0.330]	<i>true</i>
2	Hitungwaris([Istri,ibu, anaklakilaki],AhliWaris).	false Gagal menemukan solusi	<i>false</i>
3	hitungwaris([ayah, suami, anakperempuan],AhliWaris).	AhliWaris = [1/6, 1/4, 0.5]	<i>true</i>
4	hitungwaris([suami,ayah,ibu,anak perempuan,cuculakilaki],AhliWari s).	AhliWaris = [1/4, 1/6, 1/6, 0.5, "Terhijab"]	<i>true</i>

Tabel 4.6 menunjukkan hasil pengujian sistem untuk kasus warisan dari Ahli Waris Pertama Terdiri dari 4 pengujian sebagai berikut

- a. Pada *query* 1, sistem diuji dengan kasus warisan dengan ahli waris terdiri dari ayah, ibu, suami, seorang anak laki-laki, dan seorang anak perempuan. Pembagian warisan menurut sistem adalah **AhliWaris** = [1/6, 1/6, 1/4, 0.667, 0.333]

Keluaran sistem ini benar karena

1. Ayah dan Ibu mendapatkan 1/6 bagian karena *muwarits* memiliki Anak
2. Suami mendapatkan 1/4 bagian karena *muwarits* memiliki anak.
3. Anak perempuan mendapatkan 1/3 bagian karena memiliki saudara laki-laki.
4. Anak laki- laki mendapatkan 2 kali anak perempuan bagian karena memiliki saudara perempuan

- b. Pada *query* 2, sistem diuji dengan kasus warisan dengan ahli waris terdiri dari seorang istri, seorang ibu, seorang anak laki-laki. Dari data di atas diketahui bahwa yang meninggal adalah seorang suami. Namun sistem gagal dalam menemukan solusi dan memberikan nilai *false*. Hal ini terjadi karena adanya kesalahan logika pada *rule* ibu. Seharusnya ahli waris mendapatkan hasil sebagai berikut :

1. Istri mendapatkan 1/8 bagian karena *muwarits* mempunyai anak.
2. Ibu mendapatkan 1/6 bagian karena *muwarits* memiliki anak.
3. Anak laki- laki laki-laki mendapatkan 1 sisanya.

- c. Pada *query* 3, sistem diuji dengan kasus warisan dengan ahli waris terdiri dari ayah, suami dan seorang anak perempuan. Pembagian warisan menurut sistem adalah **AhliWaris** = [1/6, 1/4, 0.5]

Keluaran sistem ini benar karena

1. Ayah dan Ibu mendapatkan 1/6 bagian karena *muwarits* memiliki Anak.
2. Suami mendapatkan 1/4 bagian karena *muwarits* memiliki anak.
3. Anak perempuan mendapatkan 1/2 bagian karena memiliki saudara laki-laki.

- d. Pada *query* 4, sistem diuji dengan kasus warisan dengan ahli waris terdiri dari seorang suami, ayah, seorang ibu , dan seorang anak perempuan dan cucu laki-laki. Pembagian warisan menurut sistem adalah **AhliWaris** = [1/4, 1/6, 1/6, 0.5, "Ter-hijab"]

1. Suami mendapatkan 1/4 bagian karena *muwarits* memiliki anak.
2. Ayah dan Ibu mendapatkan 1/6 bagian karena *muwarits* memiliki Anak.
3. Anak perempuan mendapatkan 1/2 bagian sebagai *ashabah*
4. Cucu laki- laki ter-hijab karena keberadaan anak

Tabel 4.7 *Test Case* Ahli Waris Kedua

No	Query	Output	Validasi
1	hitungwaris([kakek, nenek, anaklakilaki],AhliWaris).	AhliWaris = [1/3, 0.167, 1]	<i>true</i>
2	hitungwaris([kakek, nenek, anaklakilaki, cuculaklilaki, cucuperempuan],AhliWaris).	AhliWaris = [1/3, 0.167, 1, "Terhijab", "Terhijab"]	<i>true</i>
3	hitungwaris([kakek, ayah, suami, anakperempuan],AhliWaris).	AhliWaris = ["Terhijab", 1/6, 1/4, 1/2]	<i>true</i>
4	hitungwaris([kakek, nenek],Ahli Waris).	AhliWaris = [1/3, 0.167]	<i>true</i>

Tabel 4.7 menunjukkan hasil pengujian sistem untuk kasus warisan dari Ahli Waris ke dua yang terdiri dari 4 pengujian sebagai berikut

- a. Pada *query* 1, sistem diuji dengan kasus warisan yang terdiri dari seorang kakek, nenek dan anaklakilaki. Dari kasus pada *query* 1 kakek, dan nenek merupakan Ahli Waris kedua. Ke-tidak berada-an ayah dan ibu, membuat mereka bisa mendapat harta warisan. Sedangkan anak tetap dari Ahli Waris Utama. Pembagian warisan menurut sistem adalah **AhliWaris** = [1/3, 0.167, 1]
 1. Kakek mendapatkan 1/3 bagian karena *muwarits* masih memiliki anak
 2. Nenek mendapat 1/6 bagian karena *muwarits* memiliki anak
 3. Anak laki-laki mendapatkan 1 sisanya.
- b. Pada *query* 2, sistem diuji dengan kasus warisan yang terdiri dari seorang kakek, nenek, anak laki-laki, cucu laki-laki dan cucu perempuan. Pembagian warisan menurut sistem adalah **AhliWaris** = [1/3, 0.167, 1, "Terhijab", "Terhijab"]
 1. Kakek mendapatkan 1/3 bagian karena *muwarits* masih memiliki anak
 2. Nenek mendapatkan 1/6 bagian karena *muwarits* memiliki anak
 3. Anak laki-laki mendapatkan 1 sisanya
 4. Cucu laki-laki dan cucu perempuan *terhijab* karena keberadaan anak dari *muwarits*.

- c. Pada *query* 3, sistem diuji dengan kasus warisan yang terdiri dari seorang kakek, ayah, suami, dan anak perempuan. Pembagian warisan menurut sistem adalah **AhliWaris** = ["Terhijab", 1/6, 1/4, 1/2]
1. Kakek *terhijab* oleh keberadaan ayah, sehingga tidak berhak mendapatkan warisan
 2. Ayah mendapatkan 1/6 bagian karena *muwarits* memiliki anak
 3. Suami mendapatkan 1/4 karena *muwarits* memiliki anak perempuan
 4. Anak perempuan mendapatkan 1/2 bagian karena tidak memiliki saudara kandung
- d. Pada *query* 4, sistem diuji dengan kasus warisan yang terdiri dari seorang kakek dan nenek, jika Ahli waris yang lain tidak ada. Pembagian warisan menurut sistem adalah **AhliWaris** = [1/3, 0.167]
1. Kakek mendapatkan 1/3 bagian karena tidak adanya ahli waris anak ataupun cucu
 2. Nenek akan mendapatkan 1/6 bagian.

Tabel 4.8 *Test Case* Ahli Waris Ketiga

	<i>Query</i>	<i>Output</i>	Validasi
1	hitungwaris([saudarakandungperempuan,saudarakandunglakilaki],AhliWaris).	AhliWaris = [0.33, 0.667]	<i>true</i>
2	hitungwaris([anaklakilaki,cuculakilaki,saudarakandungperempuan,saudarakandunglakilaki],AhliWaris).	AhliWaris = [1, "Terhijab", "Terhijab", "Terhijab"]	<i>true</i>
3	hitungwaris([saudaraseibuperempuan,saudarakandunglakilaki],AhliWaris).	AhliWaris = ["Terhijab", 0.667]	<i>true</i>
4	hitungwaris([saudaraseibuperempuan,saudaraseibulakilaki],AhliWaris).	AhliWaris = [0.167, 0.167]	<i>true</i>

Tabel 4.8 menunjukkan hasil pengujian sistem untuk kasus warisan dari Ahli Waris ketiga yang terdiri dari 4 pengujian sebagai berikut

- a. Pada *query* 1, sistem diuji dengan kasus warisan yang terdiri dari seorang saudara kandung perempuan dan saudara kandung laki-laki. Kasus ini dapat terjadi jika tidak ada ahli waris

pertama maupun ahli waris kedua. Pembagian warisan menurut sistem adalah **AhliWaris** = [0.330, 0.667]

1. Bagian dari saudara perempuan adalah 1/3
 2. Bagian saudara kandung Laki-laki adalah 2 x saudara perempuan, yaitu 2/3
- b. Pada *query* 2, sistem diuji dengan kasus warisan yang terdiri dari seorang anak laki-laki, cucu laki-laki, saudara kandung Laki-laki dan saudara kandung perempuan. Pembagian warisan menurut sistem adalah **AhliWaris** = [1, "Terhijab", "Terhijab", "Terhijab"]
1. Anak laki-laki akan mendapatkan 1 sisanya karena seorang *ashabah*
 2. Cucu dan saudara akan terhijab oleh keberadaan saudara kandung laki-laki
- c. Pada *query* 3, sistem diuji dengan kasus warisan yang terdiri dari seorang saudara seibu yang perempuan dan saudara kandung laki-laki. Pembagian warisan menurut sistem adalah **AhliWaris** = ["Terhijab", 0.667]
1. Saudara seibu perempuan akan *terhijab* oleh keberadaan saudara kandung perempuan
 2. Saudara kandung laki-laki akan mendapatkan 2/3 bagian karena tidak adanya saudara kandung perempuan.
- d. Pada *query* 4, sistem diuji dengan kasus warisan yang terdiri dari seorang saudara seibu perempuan dan saudara seibu laki-laki. Jika Ahli waris yang lain tidak ada. Pembagian warisan menurut sistem adalah **AhliWaris** = [0.167, 0.167]
- Saudara seibu perempuan dan laki-laki akan mendapatkan masing-masing 1/6 bagian harta.

4.4 Validasi

Untuk membuat data agar lebih bisa dipercaya, maka dibutuhkan validasi ke seorang pakar. Validasi yang penulis lakukan dilakukan kepada seorang ahli agama. Beliau bernama kiai Asep. Sehari hari beliau berprofesi sebagai penjual bubur dan sebagai *ustad* di desa *Garajati*, kec. *Ciwaru*, kab. *Kuningan* Jawa-Barat.

Validasi ditanyakan kepada setiap kasus pengujian apakah sudah benar atau belum. Dari semua *query* yang ditanyakan ditemukan beberapa kesalahan logika dan kekurangan aturan. Salah satunya bahwa kakek dapat menjadi *ashabah* dan mendapatkan 1(*satu*). Namun kasus kakek menjadi seorang *ashabah* sangat jarang ditemui dalam dunia nyata. Selanjutnya kesalahan-kesalahan yang ada langsung diperbaiki sebagaimana harusnya. Kesalahan yang ada

sudah diperbaiki sebagaimana harusnya dan dapat dilihat *outputnya* pada tabel validasi 4.9, tabel validasi 4.10, dan tabel validasi 4.11

Tabel 4.9 Validasi Ahli Waris Pertama

No	Query	Output	Validasi
1	Hitungwaris([ayah,ibu,suami,anak lakilaki,anakperempuan], AhliWaris).	AhliWaris = [1/6, 1/6, 1/4, 0.667, 0.330]	<i>true</i>
2	Hitungwaris([Istri,ibu, anaklakilaki],AhliWaris).	AhliWaris = [1/8, 1/6, 1]	<i>true</i>
3	hitungwaris([ayah, suami, anakperempuan],AhliWaris).	AhliWaris = [1/6, 1/4, 0.5]	<i>true</i>
4	hitungwaris([suami,ayah,ibu,anak perempuan,cuculakilaki],AhliWaris).	AhliWaris = [1/4, 1/6, 1/6, 0.5, "Terhijab"]	<i>true</i>

Pada tabel validasi 4.6 terdapat Kesalahan logika pada *query* 2 sudah diperbaiki pada *query* tersebut sistem diuji dengan kasus warisan yang terdiri dari seorang istri, seorang ibu, seorang anak laki-laki. Dari data di-atas diketahui bahwa yang meninggal adalah seorang suami. *rule* ibu memiliki sedikit kesalahan logika yang sudah diperbaiki, dengan hasil *output* dari *rule* sebagai berikut.

- a. Istri mendapatkan 1/8 bagian karena *muwarits* mempunyai anak.
- b. Ibu mendapatkan 1/6 bagian karena *muwarits* memiliki anak.
- c. Anak laki-laki mendapatkan 1 sisanya.

Tabel 4.10 Validasi Ahli Waris Kedua

No	Query	Output	Validasi
1	hitungwaris([kakek, nenek, anaklakilaki],AhliWaris).	AhliWaris = [1/6, 0.167, 1]	<i>true</i>
2	hitungwaris([kakek, nenek, anaklakilaki, cuculakilaki, cucuperempuan],AhliWaris).	AhliWaris = [1/6, 0.167, 1, "Terhijab", "Terhijab"]	<i>true</i>
3	hitungwaris([kakek, ayah, suami, anakperempuan],AhliWaris).	AhliWaris = ["Terhijab", 1/6, 1/4, 1/2]	<i>true</i>

4	hitungwaris([kakek,nenek],Ahli Waris).	AhliWaris = [1, 0.167]	<i>true</i>
---	--	-------------------------------	-------------

Pada Tabel 4.7 bagian pada *rule* kakek pada *query* 1 dan 2 terdapat kesalahan bagian warisan. Kakek merupakan pengganti dari ayah. Ketika *muwarits* tidak memiliki anak maka kakek akan mendapat 1/3 dan jika *muwarits* memiliki anak maka kakek akan mendapat 1/6 bagian.

Rule Kakek untuk menjadi *ashabah* juga sudah ditambahkan ke dalam basis pengetahuan, sehingga tabel *test case* 4.7 sudah valid semua.

Tabel 4.11 Validasi Ahli Waris ketiga

	<i>Query</i>	<i>Output</i>	Validasi
1	hitungwaris([saudarakandungperempuan,saudarakandunglakilaki],AhliWaris).	AhliWaris = [0.330, 0.667]	<i>true</i>
2	hitungwaris([anaklakilaki,cuculakilaki,saudarakandungperempuan,saudarakandunglakilaki],AhliWaris).	AhliWaris = [1, "Terhijab", "Terhijab", "Terhijab"]	<i>true</i>
3	hitungwaris([saudaraseibuperempuan,saudarakandunglakilaki],AhliWaris).	AhliWaris = ["Terhijab", 0.667]	<i>true</i>
4	hitungwaris([saudaraseibuperempuan,saudaraseibulakilaki],AhliWaris).	AhliWaris = [0.167, 0.167]	<i>true</i>

Pada *test case* ahli waris ketiga, tidak ditemukannya kesalahan, sehingga hasil *output* sudah sesuai apa yang diharapkan.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil pembagian waris yang didapat dari basis pengetahuan yang dibuat sudah benar sesuai dengan aturan yang ada di dalam Al-Quran. Pembuatan sistem menggunakan bahasa pemrograman Prolog sudah sesuai untuk menjawab permasalahan pembagian waris. Inferensi yang digunakan adalah *Forward chaining*, ini cocok untuk kasus pembagian waris yang melakukan pelacakan ke-depan dengan aturan. Penggunaan *list* dalam Prolog sebagai koleksi data membutuhkan *variable copy* untuk menjaga nilai bagian ahli waris sebelumnya, karena jika tidak di-*copy* maka nilai ahli waris akan menghilang karena proses pemanggilan data secara *rekursif*. Penggunaan metode pengujian *White-box testing* sangat baik digunakan dalam *project* pembagian waris menggunakan Prolog, karena hampir seluruh isi kode terdiri dari percabangan logika. Keluaran dari sistem ini adalah bagian-bagian warisan yang didapat oleh setiap ahli waris.

Dalam bahasa pemrograman Prolog terdapat beberapa keterbatasan untuk melakukan operasi matematika. Adanya keterbatasan untuk membuat UI (*user interface*) yang menarik membutuhkan gabungan dengan bahasa pemrograman lain.

5.2 Saran

Bahasa pemrograman Prolog cukup berbeda dengan bahasa pemrograman lainnya, karena menggunakan paradigma yang berbeda untuk membuat program. Dari mulai data(*atom*), klausa, konjungsi, dan *variable* semua memiliki cara penyajiannya sendiri, sehingga kita sebagai *engineer* harus belajar dari dasar. Oleh karena itu, dibutuhkannya pemahaman yang mendalam tentang bahasa pemrograman itu sendiri. Pengguna bahasa pemrograman Prolog saat ini tidak terlalu banyak, namun masih terdapat banyak sumber ilmu yang berada di internet. Masalah yang ditemukan saat proses *kodefikasi* tidak selalu tersedia di internet, namun kita bisa bertanya dalam *stackoverflow* atau *web* sejenis lainnya.

Karena Prolog tidak mendukung pembuatan *user interface*, maka diharapkan dalam penelitian selanjutnya dilakukan menggabungkan bahasa pemrograman lain untuk membuat *User Interface*(UI) dan Prolog sebagai basis pengetahuan. Dibutuhkannya *user interface* karena saat mengajukan *query* pada prolog *user* harus menulis manual setiap ahli waris dan

kemungkinan kesalahan pengetikan *query* dapat membuat sistem prolog mengembalikan nilai *false*.



DAFTAR PUSTAKA

- Dahria, M. (n.d.). *PENGEMBANGAN SISTEM PAKAR DALAM MEMBANGUN SUATU APLIKASI*.
- Listiyono, H. (2008). Merancang dan Membuat Sistem Pakar. *Jurnal Teknologi Informasi DINAMIK*, XIII(2), 115–124.
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). *PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)*. I(3), 31–36.
- Priyaungga, B. A., Aji, D. B., Syahroni, M., Aji, N. T. S., & Saifudin, A. (2020). Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(3), 150. <https://doi.org/10.32493/jtsi.v3i3.5343>
- KRISTANTO, Andri. (2003). *Perancangan Sistem Informasi dan Aplikasinya* (Cet. 1). YOGYAKARTA: Gava Media.
- S. Pressman, Roger, “Rekayasa Perangkat Lunak Pendekatan Praktisi Buku 1 dan 2”, Andi, Yogyakarta, 2012.
- Jogiyanto, “ Analisis dan Desain Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis”, Andi, Yogyakarta, 2005.
- Akil, I. (2017). Analisa Efektifitas Metode Forward Chaining Dan. *Jurnal Pilar Nusa Mandiri*, 13(1), 35–42. Akil, I. (2017). Analisa Efektifitas Metode Forward Chaining Dan. *Jurnal Pilar Nusa Mandiri*, 13(1), 35–42.
- Brata, D. W. (2012). Perancangan Aplikasi Mobile Al-Faraidh (Penghitungan Hak Waris) Berbasis Sistem Android. *Jurnal Ilmiah Teknologi Informasi Asia*, 6(1), 31–36.
- Fanny, R. R., Hasibuan, N. A., & Buulolo, E. (2017). Perancangan Sistem Pakar Diagnosa Penyakit Asidosis Tubulus Renalis Menggunakan Metode Certainty Factor Dengan Penelusuran Forward Chaining. *Media Informatika Bud Akil*, I. (2017). Analisa Efektifitas Metode Forward Chaining Dan. *Jurnal Pilar Nusa Mandiri*, 13(1), 35–42.
- Fanny, R. R., Hasibuan, N. A., & Buulolo, E. (2017). Perancangan Sistem Pakar Diagnosa Penyakit Asidosis Tubulus Renalis Menggunakan Metode Certainty Factor Dengan Penelusuran Forward Chaining. *Media Informatika Budidarma*, 1(1), 13–16.
- FAQIHUDDIN, M. (2016). Sistem Pakar Penunjang Keputusan Pembagian Harta Waris Menurut Hukum Islam Dengan Metode Forward Chaining.

- Hasibuan, Z. A. (2007). *Metodologi Penelitian Pada Bidang Ilmu Komputer Dan Teknologi Informasi. Konsep, Teknik, Dan Aplikasi*, Universitas Indonesia, 194.
- Ian Gatra Sandika, Adhistya Erna Permanasari, S. S. (2014). Penentuan Karakteristik Pengguna Sebagai Pendukung Keputusan Dalam Memilih Smartphone Menggunakan Forward Chaining. *Prosiding SNATIF Ke-1 Tahun 2014*, 1, 301–308.
- Minarni, D., Isa, I. G. T., & Yanik, A. (2018). Aplikasi Bagi Waris Islam dengan Metode Forward Chaining berbasis Web. *Jurnal Online Informatika*, 2(2), 127.
<https://doi.org/10.15575/join.v2i2.107>
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). *I(3)*, 31–36.
- Nidhra, S., & Dondeti, J. (2012). Black Box and White Box Testing Techniques – A Literature Review. *International Journal of Embedded System and Application (IJESA)*, 2(2), 29–50. <http://airccse.org/journal/ijesa/papers/2212ijesa04.pdf>
- Nugrahani, F. (2014). dalam Penelitian Pendidikan Bahasa. *信阳师范学院*, 1(1), 305.
- Nursalam, 2016, metode penelitian, & Fallis, A. . (2013). Analisa Metode Forward Chaining Untuk Sistem Pakar Pembagian Harta Warisan Sesuai Hukum Islam. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699.
- Priyaungga, B. A., Aji, D. B., Syahroni, M., Aji, N. T. S., & Saifudin, A. (2020). Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(3), 150.
<https://doi.org/10.32493/jtsi.v3i3.5343>
- HARTONO, Jogyanto. (2003). *Pengembangan Sistem Pakar Menggunakan Visual Basic* (Ed. 1). YOGYAKARTA: ANDI.

LAMPIRAN

Link gituhub :

https://github.com/TheDayEater/Tugas-Akhir/blob/main/pembagian_waris_dengan_prolog







