

SKRIPSI

**SIMULATOR PERTUMBUHAN WIRAUSAHA BERBASIS
CELLULAR AUTOMATA**



Vanessa Sukamto

NPM: 2014730010

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2018**

UNDERGRADUATE THESIS

**CELLULAR AUTOMATA BASED ENTREPRENEURIAL
GROWTH SIMULATOR**



Vanessa Sukamto

NPM: 2014730010

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2018**

LEMBAR PENGESAHAN

SIMULATOR PERTUMBUHAN WIRAUSAHA BERBASIS CELLULAR AUTOMATA

Vanessa Sukamto

NPM:2014730010

Bandung, 6 Desember 2018

Menyetujui,

Pembimbing

Dr.rer.nat. Cecilia Esti Nugraheni

Ketua Tim Penguji

Anggota Tim Penguji

Elisati Hulu, M.T.

Luciana Abednego, M.T.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

SIMULATOR PERTUMBUHAN WIRAUSAHA BERBASIS CELLULAR AUTOMATA

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 6 Desember 2018

Meterai Rp. 6000

Vanessa Sukamto
NPM: 2014730010

ABSTRAK

Kewirausahaan sangat diperlukan guna mendorong pertumbuhan ekonomi suatu negara. Untuk itu pertumbuhan kewirausahaan perlu dipantau secara terus-menerus. Selain dengan pemantauan langsung, pemantauan juga dapat dilakukan secara tidak langsung dengan melakukan simulasi. ECA (*Entrepreneurial Cellular Automata*) merupakan model matematis berdasarkan *cellular automata* yang dapat digunakan untuk memodelkan dan mensimulasikan pertumbuhan wirausaha. ECA menggunakan model wirausaha yang diusulkan oleh Global Entrepreneurship Monitor (GEM). Dari GEM ini diperoleh parameter apa saja yang mempengaruhi pertumbuhan wirausaha.

Skripsi ini bertujuan untuk mengimplementasikan ECA ke dalam sebuah perangkat lunak simulator. Simulator ini diharapkan dapat mempermudah simulasi pertumbuhan wirausaha dengan memungkinkan pengguna mengubah parameter-parameter yang diinginkan.

Simulator yang berhasil dibangun terdiri atas beberapa tampilan yang masing-masing tampilannya akan meminta *user* untuk mengisi bobot atau nilai dari parameter yang digunakan untuk menjalankan simulasi. Pengujian simulator ini dilakukan dengan pengujian fungsional, pengujian pembacaan parameter, pengujian *file* data wirausaha dan pengujian hasil dari simulasi. Hasil dari pengujian tersebut, simulator ini dapat berjalan dengan baik.

Kata-kata kunci: Entrepreneurial Cellular Automata, Cellular Automata, GEM, Pertumbuhan Wirausaha

ABSTRACT

Entrepreneurship is needed to encourage economic growth of a country. Besides to direct monitoring, monitoring can also be done indirectly by doing a simulation.

ECA (Entrepreneurial Cellular Automata) is a mathematical model based on cellular automata that can be used to model and simulate entrepreneurial growth. ECA uses an entrepreneurial model proposed by the Global Entrepreneurship Monitor (GEM). From this GEM, the parameters that influence entrepreneurial growth are obtained.

This thesis aims to implement the ECA into a simulator software. This simulator is expected to facilitate the simulation of entrepreneurial growth by enabling users to change the desired parameters.

The simulator that was successfully built consists of several interfaces that will ask user to fill the weight or value of the parameters to run the simulation. Simulator testing is done with functional testing, parameter reading testing, file input testing and the result of the simulation testing. The result of these tests, the simulator can run well.

Keywords: Entrepreneurial Cellular Automata, Cellular Automata, The Growth of Entrepreneurship, GEM

Tuhan Yesus, Papah, Mamah dan Diri Sendiri

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas berkat yang diberikan kepada penulis sehingga dapat menyelesaikan skripsi dengan judul Simulator Pertumbuhan Wirausaha Berbasis Cellular Automata dengan baik dan tepat waktu. Penulis juga berterima kasih kepada pihak-pihak yang telah membantu dan memberikan dukungan untuk menyelesaikan skripsi ini, yaitu :

1. Orang tua yang senantiasa memberi dukungan dan semangat kepada penulis.
2. Ibu Cecilia Nugraheni sebagai dosen pembimbing yang sudah membimbing penulis hingga dapat menyelesaikan skripsi ini dengan sabar dan memberi semangat juga kepada penulis.
3. Bapak Elisati Hulu dan Ibu Luciana Abednego sebagai dosen penguji yang telah membantu dalam menguji dan memperbaiki skripsi ini.
4. Adriani Sukanto dan Tommy Aditya The, yang memberi dukungan serta bertukar pikiran dalam menyelesaikan skripsi ini.
5. Lydia Febtriani, Stephanie Tania dan Gabriella yang memberi dukungan dan semangat dalam mengerjakan skripsi ini.
6. Desy Yudianto, Lady, Della Kezia dan Steffy Wijayanti yang telah memberi dukungan dan semangat untuk melancarkan sidang.
7. Ferdi Subagia, Samuel Yudhistira dan Hendri Susanto yang memberi semangat untuk melancarkan sidang.
8. Teman-teman Teknik Informatika UNPAR angkatan 2014 yang telah berbagi ilmu kepada penulis.
9. Pihak-pihak lain yang belum disebutkan, yang berperan dalam penyelesaian skripsi ini.

Akhir kata, penulis berharap agar skripsi ini dapat bermanfaat bagi pembaca yang hendak melakukan penelitian dan pengembangan yang terkait dengan skripsi ini.

Bandung, Desember 2018

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	3
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 Arti Kewirausahaan	5
2.2 Cellular Automata	19
2.2.1 Dimensi CA	20
2.2.2 Aplikasi CA	22
2.3 Graf	22
2.3.1 Adjacency List	23
2.3.2 Adjacency Matriks	24
2.4 Entrepreneurial Cellular Automata	25
3 ANALISIS	29
3.1 Analisis Model Pertumbuhan Wirausaha dengan ECA	29
3.2 Deskripsi Perangkat Lunak	36
3.3 Analisis Perangkat Lunak	36
3.3.1 Diagram <i>Use Case</i>	36
3.3.2 Diagram Kelas	39
3.3.3 Kelas EGM	39
3.3.4 Kelas CA	39
3.3.5 Kelas Entrepreneurs	41
3.3.6 Kelas Neighbor	42
3.3.7 Kelas Neighborhood	42
3.3.8 Kelas Public Factor	42
3.3.9 Kelas State	42
4 PERANCANGAN	43
4.1 Diagram Kelas	43
4.1.1 Kelas CA	43

4.1.2	Kelas Tampilan Bobot Ketetangaan	44
4.1.3	Kelas Tampilan Kondisi Ketetangaan	44
4.1.4	Kelas Tampilan Kondisi Eksternal	45
4.1.5	Kelas Data Wirausaha	45
4.1.6	Kelas Tampilan Simulasi	45
4.1.7	Kelas Tampilan Hasil	45
4.1.8	Kelas Input Data Handler	45
4.2	Rancangan Antarmuka	47
4.2.1	Tampilan Bobot Ketetangaan	47
4.2.2	Tampilan Kondisi Ketetangaan	48
4.2.3	Tampilan Kondisi Eksternal	49
4.2.4	Tampilan Data Wirausaha	50
4.2.5	Tampilan Simulasi	51
4.2.6	Tampilan Hasil	51
4.3	Rancangan File Input	52
5	IMPLEMENTASI DAN PENGUJIAN	55
5.1	Implementasi	55
5.1.1	Implementasi	55
5.1.2	Hasil Implementasi	55
5.2	Pengujian	61
5.2.1	Pengujian Fungsional	61
5.2.2	Pengujian Pembacaan Parameter	62
5.2.3	Pengujian Pembacaan File	64
5.2.4	Pengujian Hasil dari Simulasi	65
5.3	Eksperimen dan Analisis Hasil	66
6	KESIMPULAN DAN SARAN	75
6.1	Kesimpulan	75
6.2	Saran	76
	DAFTAR REFERENSI	77
	A KODE PROGRAM	79

DAFTAR GAMBAR

2.1	Fase Wirausaha	6
2.2	Komposisi perceived capabilities untuk selang usia yang berbeda	8
2.3	Komposisi perceived capabilities untuk tingkat pendidikan yang berbeda	9
2.4	Komposisi perceived capabilities untuk tingkat pendapatan	9
2.5	Komposisi perceived capabilities untuk wilayah Indonesia	10
2.6	Komposisi role model untuk umur	10
2.7	Komposisi role model untuk tingkat pendapatan yang berbeda	11
2.8	Komposisi Perceived Opportunities usia wanita dan pria	11
2.9	Komposisi perceived opportunities untuk tingkat pendidikan yang berbeda	12
2.10	Komposisi Perceived Opportunities untuk wilayah Indonesia	12
2.11	Komposisi Perceived Opportunities untuk tingkat pendapatan	13
2.12	Komposisi Fear of Failure untuk usia wanita dan pria	13
2.13	Komposisi Fear of Failure untuk tingkat pendidikan	14
2.14	Komposisi Fear of Failure untuk wilayah Indonesia	15
2.15	Komposisi High Status of Successful untuk kategori usia	16
2.16	Komposisi High Status of Successful berdasarkan kota tinggal	17
2.17	Komposisi High Status of Successful berdasarkan tingkat pendapatan	18
2.18	Komposisi Media Attention berdasarkan kota tinggal	18
2.19	Komposisi Media Attention berdasarkan tingkat pendapatan	19
2.20	CA 1 Dimensi	20
2.21	Aturan 30 dari Wolfram	20
2.22	Ilustrasi penerapan aturan 30 dari Wolfram	20
2.23	CA 2 Dimensi	21
2.24	Aturan Dasar Conway's Game of Life	21
2.25	Ilustrasi Conway's Game of Life	21
2.26	Contoh Graf	23
2.27	<i>Undirected Cyclic Graph</i>	23
2.28	<i>Undirected Cyclic Graph</i>	24
2.29	<i>Directed Cyclic Graph</i>	24
2.30	Diagram Transisi Level Wirausaha	27
3.1	Gambar ketetangaan tiga entrepreneur pada saat awal	30
3.2	Gambar ketetangaan tiga entrepreneur pada saat $t = 0$	32
3.3	Gambar ketetangaan tiga entrepreneur pada saat $t = 1$	33
3.4	Gambar ketetangaan tiga entrepreneur pada saat $t = 2$	34
3.5	Gambar ketetangaan tiga entrepreneur pada saat $t = 3$	35
3.6	Gambar ketetangaan tiga entrepreneur pada saat $t = 4$	36
3.7	Use Case ECA	37
3.8	Diagram Kelas ECA	39
4.1	Diagram Kelas Simulator ECA	44
4.2	Gambar Tampilan Bobot Ketetangaan	47
4.3	Gambar Tampilan Kondisi Ketetangaan	48

4.4	Gambar Tampilan Kondisi Eksternal	49
4.5	Gambar Tampilan Data Wirausaha	50
4.6	Gambar Tampilan Simulasi	51
4.7	Gambar Tampilan Hasil	52
4.8	Contoh Format File Data Wirausaha	54
5.1	Gambar Tampilan Bobot Ketetanggaan	56
5.2	Gambar Tampilan Bobot Ketetanggaan	56
5.3	Gambar Tampilan Kondisi Ketetanggaan	57
5.4	Gambar Tampilan Kondisi Ketetanggaan	57
5.5	Gambar Tampilan Ketetanggaan	58
5.6	Gambar Tampilan Data Wirausaha	59
5.7	Gambar Tampilan Data Wirausaha	59
5.8	Gambar Tampilan Simulasi	60
5.9	Hasil keluaran perubahan individu wirausaha pada <i>file</i> CSV	60
5.10	Hasil keluaran perubahan individu wirausaha pada <i>file</i> CSV	61
5.11	Gambar Tampilan Hasil	61
5.12	Tampilan Pesan Error pada saat <i>text field</i> tidak terisi	62
5.13	Tampilan Pesan Error pada saat isi dari <i>text field</i> tidak berjumlah 100%	63
5.14	Tampilan Pesan Error pada saat <i>radio button</i> tidak terisi	63
5.15	Tampilan Pesan Error pada saat <i>text field</i> tidak terisi seluruhnya	63
5.16	Tampilan Pesan Error pada saat <i>text field</i> tidak terisi	63
5.17	Tampilan Pesan Error pada saat isi dari <i>text field</i> tidak berjumlah 100%	63
5.18	Tampilan Pesan Error pada saat nilai a,b dan c tidak berjumlah 1	64
5.19	Contoh format <i>file</i> data wirausaha	64
5.20	Contoh format <i>file</i> data wirausaha	64
5.21	Tampilan pesan kesalahan apabila <i>file</i> data wirausaha belum dipilih	65
5.22	Hasil Iterasi bulan pertama	65
5.23	Hasil Iterasi bulan kedua	65
5.24	Hasil Iterasi bulan ketiga	65
5.25	Hasil Iterasi bulan keempat	66
5.26	Hasil Iterasi bulan kelima	66
5.27	Hasil dari simulasi 1	67
5.28	Hasil dari simulasi 2	68
5.29	Hasil dari simulasi 3	68
5.30	Hasil dari simulasi 4	68
5.31	Hasil dari simulasi 5	69
5.32	Hasil dari simulasi 6	69
5.33	Hasil dari simulasi 7	69
5.34	Hasil dari simulasi 8	70
5.35	Hasil dari simulasi 9	70
5.36	Hasil dari simulasi 10	70
5.37	Hasil dari simulasi 11	71
5.38	Hasil dari simulasi 12	71
5.39	Hasil dari simulasi 13	71
5.40	Hasil dari simulasi 19	72
5.41	Hasil dari simulasi 20	72
5.42	Hasil dari simulasi 21	72
5.43	Hasil dari simulasi 22	73
5.44	Hasil dari simulasi 23	73
5.45	Hasil dari simulasi 24	73

DAFTAR TABEL

2.1	Indikator Kewirausahaan	7
2.2	Lanjutan Indikator Kewirausahaan	8
2.3	Komposisi High Status of Successful untuk tingkat pendidikan	16
2.4	Faktor Publik	19
2.5	Tabel Representasi Adjacency List	24
2.6	Tabel Representasi Adjacency Matriks	24
2.7	Tabel Representasi Adjacency Matriks	25
2.8	Transisi Level Wirausaha	27
3.1	Data wirausahawan	29
3.2	Data Bobot Atribut	29
3.3	Tabel Skenario Memasukkan Parameter Simulasi	38
3.4	Tabel Skenario Memasukkan <i>file</i> data wirausaha dalam format text	38
3.5	Tabel Skenario Menjalankan Simulasi	38
5.1	Tabel Pengujian Fungsional <i>User</i>	62
5.2	Tabel Parameter Settings	67

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pada saat ini, lapangan kerja pada suatu negara tidak bisa diprediksi, tetapi kenyataannya lapangan kerja dari tahun ke tahun semakin terbatas [1]. Dengan melihat situasi tersebut maka bisa dipastikan tingkat pengangguran di suatu negara akan semakin tinggi. Solusi terbaik untuk mengurangi permasalahan tersebut adalah dengan berwirausaha. Kewirausahaan adalah kemampuan seseorang untuk membuat suatu usaha yang dimulai dari 0 yang dirintis hingga usaha tersebut benar-benar sukses. Tentu saja hal ini memberikan pengaruh positif terhadap pertumbuhan ekonomi suatu negara, karena kewirausahaan juga sekaligus membuka lapangan kerja bagi masyarakat. Jika usaha yang dirintis semakin besar, otomatis perusahaan tersebut akan merekrut tenaga kerja yang semakin banyak lagi.

Pada zaman sekarang, sudah banyak sekali orang yang lebih memilih untuk berwirausaha daripada bekerja di kantor atau di sebuah perusahaan. Alasan mengapa banyak orang lebih memilih berwirausaha pun bervariasi contohnya orang tersebut tidak terlalu menyukai waktu kerjanya diatur oleh orang lain melainkan ia lebih menyukai waktu kerjanya diatur oleh dirinya sendiri. Tidak hanya pada jaman sekarang, dari jaman dahulu juga sudah ada wirausahawan yang namanya tidak asing lagi didengar oleh telinga kita salah satunya yaitu Bob Sadino. Untuk menjadi wirausahawan yang sukses seperti Bob Sadino tidaklah mudah, pasti ada beberapa faktor dari luar maupun dalam yang mempengaruhi keberlangsungan wirausaha. Dalam berwirausaha dibutuhkan usaha yang besar untuk menjadi sukses, usaha tersebut juga harus dijaga kekonsistennannya agar tidak mengalami kebangkrutan.

Kewirausahaan sangat diperlukan guna mendorong perekonomian suatu negara karena dapat mengurangi tingkat pengangguran di Indonesia. Secara ekonomis, kewirausahaan akan membantu meningkatkan pendapatan masyarakat atau meningkatkan kesejahteraan melalui penciptaan produk baru, serta mengurangi kemiskinan. Ideal besarnya populasi wirausaha dalam suatu negara adalah 2% dari total penduduk suatu negara. Saat ini Indonesia baru mencapai 1.5% pengusaha dari total penduduk [2]. Maka dari itu, kondisi wirausaha ini perlu dipantau terus-menerus perkembangannya agar dapat memajukan perekonomian di Indonesia. Pemantauan ini dilakukan oleh pemerintah dan lembaga-lembaga swasta yang berkepentingan. Salah satu lembaga yang memantau kewirausahaan adalah GEM (Global Entrepreneurship Monitor). GEM merupakan konsorsium yang bertujuan untuk mengukur dan memantau kegiatan kewirausahaan.

Selain pemantauan terhadap kondisi riil, salah satu kegiatan yang mendukung pemantauan adalah pengamatan secara tidak langsung. Salah satu pengamatan tidak langsung adalah dengan membuat model matematika dari pertumbuhan wirausaha dan kemudian melakukan simulasi terhadap model tersebut. Salah satu model matematika yang dapat digunakan untuk memodelkan pertumbuhan wirausaha adalah *Entrepreneurial Cellular Automata* (ECA) yang diusulkan oleh Nugraheni dan Natali [2]. ECA adalah pengembangan dari *Cellular Automata* standar dari Ulam dan von Neumann. *Cellular Automata* (CA) sendiri merupakan suatu model matematika yang digunakan untuk memodelkan suatu sistem dinamis. Pada [2] dijelaskan bagaimana struktur dari ECA dan diberikan ilustrasi bagaimana menggunakan ECA untuk memprediksi pertumbuhan

wirausaha berdasarkan parameter wirausaha dari GEM.

Dalam hasil penelitian ECA setiap wirausahawan mempunyai beberapa atribut yang bersifat statis maupun dinamis. Contoh atribut yang bersifat statis yaitu bidang usaha, kategori usaha, lokasi geografis dan jenis kelamin. Sementara contoh untuk atribut dinamis adalah usia, level wirausaha dan usia usaha. Diantara atribut dinamis, level wirausaha menjadi atribut penting karena atribut ini yang akan menjadi acuan untuk menentukan perkembangan dari kewirausahaan. *Continuity Index* digunakan untuk menentukan apakah seorang wirausahawan pada suatu saat tertentu akan meneruskan usahanya pada waktu selanjutnya.

Akan tetapi ECA yang telah dibuat belum bisa menggambarkan kepada *user* awam atau pemantau tentang parameter mana saja yang dibutuhkan untuk dapat melihat pertumbuhan wirausaha dalam waktu tertentu. Selain untuk melihat pertumbuhan wirausaha, dengan adanya simulator pemantau juga dapat mengetahui faktor apa saja yang paling berpengaruh dalam menaikkan pertumbuhan wirausaha dan faktor apa saja yang membuat pertumbuhan wirausaha menurun. Skripsi ini bertujuan untuk membangun sebuah simulator ECA dengan memperhitungkan beberapa parameter yang belum diperhatikan pada penelitian sebelumnya dan menampilkan hasil simulasi dalam bentuk tabel.

1.2 Rumusan Masalah

Berikut adalah susunan permasalahan yang akan dibahas pada penelitian ini:

1. Faktor apa saja yang mempengaruhi keberlangsungan wirausaha?
2. Bagaimana memodelkan pertumbuhan wirausaha dengan *Entrepreneurial Cellular Automata*?
3. Bagaimana membangun simulator keberlangsungan wirausaha dengan *Entrepreneurial Cellular Automata*?

1.3 Tujuan

Berdasarkan rumusan masalah yang telah dibuat, maka tujuan penelitian ini dijelaskan ke dalam poin-poin sebagai berikut :

1. Mempelajari faktor yang berpengaruh pada keberlangsungan wirausaha.
2. Memodelkan pertumbuhan wirausaha dengan *Entrepreneurial Cellular Automata*.
3. Membangun simulator keberlangsungan wirausaha dengan *Entrepreneurial Cellular Automata*.

1.4 Batasan Masalah

1. Tidak bertujuan untuk menguji kualitas atau kebenaran dari ECA, tetapi hanya membangun simulator untuk ECA saja.
2. Perangkat lunak yang dibuat hanya bisa dijalankan pada komputer / *laptop*.
3. Hanya mempelajari perkembangan wirausaha dari GEM.
4. Data bersifat statis artinya hanya menangani wirausaha yang sudah ada, tidak menangani wirausaha baru yang muncul pada saat periode tertentu.
5. Data wirausaha yang diuji bukan data nyata.
6. Nilai-nilai beberapa konstanta yang digunakan pada simulator didasarkan pada data dari GEM.

1.5 Metodologi

Langkah-langkah yang akan dijalani untuk menyelesaikan penelitian ini :

1. Melakukan studi pustaka untuk hal-hal berikut :
 - (a) *Cellular Automata* khususnya ECA
 - (b) Kewirausahaan khususnya GEM
2. Menganalisis masalah kewirausahaan untuk membangun simulator pertumbuhan wirausaha menggunakan *Entrepreneurial Cellular Automata*.
3. Merancang perangkat lunak berdasarkan hasil pemodelan.
4. Mengimplementasikan perangkat lunak sesuai rancangan.
5. Menguji perangkat lunak yang dibuat.
6. Menulis dokumen skripsi.

1.6 Sistematika Pembahasan

Setiap bab dalam penelitian ini memiliki sistematika penulisan yang dijelaskan ke dalam poin-poin sebagai berikut :

1. Bab 1: Pendahuluan berisi latar belakang masalah, rumusan masalah, tujuan, batasan masalah, metodologi dan sistematika pembahasan.
2. Bab 2: Landasan Teori yaitu akan membahas mengenai arti kewirausahaan, penjelasan *cellular automata*, penjelasan *entrepreneurial cellular automata* dan penjelasan tentang graf.
3. Bab 3: Analisis, yaitu berisi analisis pertumbuhan wirausaha, analisis pemodelan *entrepreneurial cellular automata*, analisis model pertumbuhan wirausaha dengan *entrepreneurial cellular automata*, deskripsi perangkat lunak dan analisis perangkat lunak.
4. Bab 4: Perancangan, membahas mengenai diagram kelas, rancangan antarmuka dan rancangan *file* input.
5. Bab 5: Implementasi dan Pengujian, pada bab ini berisi tentang implementasi, hasil implementasi dan contoh simulasi. Dalam pengujian akan dijelaskan pengujian fungsional, pengujian pembacaan parameter, pengujian pembacaan *file* dan pengujian hasil dari simulasi.
6. Bab 6: Kesimpulan dan Saran, yaitu membahas hasil kesimpulan dari keseluruhan penelitian ini dan saran-saran yang dapat diberikan untuk penelitian berikutnya.

BAB 2

LANDASAN TEORI

Pada bab ini akan dibahas mengenai landasan teori yang digunakan pada penyusunan tugas akhir. Pembahasan pertama mencakup hal-hal yang berkaitan dengan pengertian kewirausahaan dari umum sampai khusus yaitu kewirausahaan menurut GEM. Pembahasan kedua yaitu tentang graf. Pembahasan terakhir yaitu tentang teori dan aplikasi dari CA (Cellular Automata) khususnya tentang ECA (Entrepreneurial Cellular Automata).

2.1 Arti Kewirausahaan

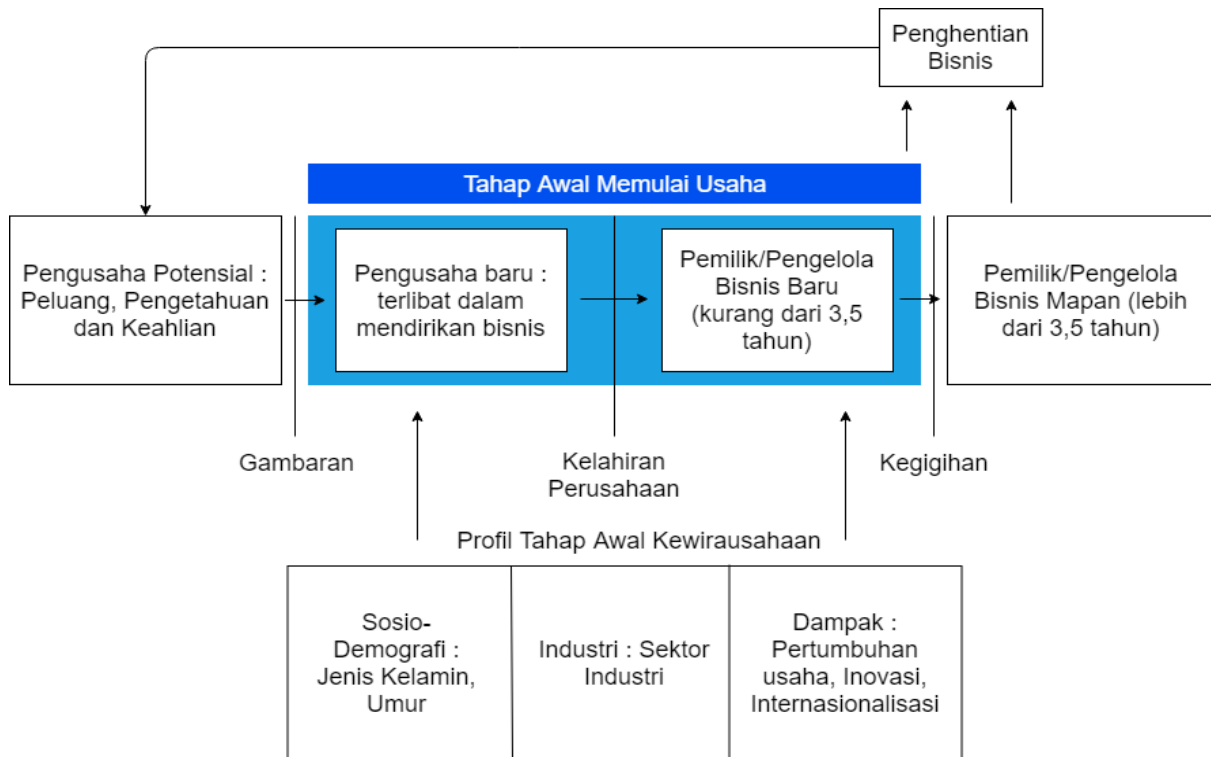
Wirausaha berasal dari kata wira dan usaha. Wira artinya unggul, mulia, luhur sedangkan usaha berarti kemampuan melakukan usaha atas kekuatan diri sendiri. Jadi wirausahawan adalah manusia yang unggul yang memiliki kemampuan membangun usaha sendiri. Kewirausahaan sendiri merupakan kepribadian wirausahawan. Wirausahawan merupakan orang atau manusia yang memperjuangkan kemajuan terutama pada bidang ekonomi demi masyarakat seperti menciptakan lapangan pekerjaan, membantu memenuhi kebutuhan masyarakat yang semakin meningkat dan berusaha mengurangi ketergantungan dari luar negeri. Istilah kewirausahaan pada umumnya merupakan suatu ilmu yang mempelajari tentang kemampuan seseorang dalam menghadapi tantangan hidup untuk memperoleh peluang dan menghadapi segala risiko yang ada dengan mengandalkan kekuatan diri sendiri tanpa bergantung pada orang lain. [3]

GEM (Global Entrepreneurship Monitor) merupakan lembaga yang memantau dan mengukur pertumbuhan wirausaha di berbagai negara yang didirikan pada tahun 1997 oleh Michael Hay dan Bill Bygrave. GEM telah memantau kewirausahaan di 104 ekonomi negara dan telah mendapat pengakuan luas sebagai penelitian kewirausahaan secara longitudinal yang memiliki kewenangan kuat di dunia. Pada tahun 2006, Indonesia sempat bergabung dengan GEM untuk mempelajari kewirausahaan. Setelah absen selama 6 tahun, Indonesia kembali bergabung dengan GEM pada tahun 2013.[2]

GEM melakukan penelitiannya berdasarkan pada beberapa premis. Pertama, keadaan ekonomi suatu negara. Jika keadaan ekonomi suatu negara sedang sulit itu artinya dengan adanya wirausaha dapat membantu memperluas lapangan pekerjaan (memotivasi orang untuk menjadi seorang wirausaha juga lebih meningkat), sedangkan jika keadaan ekonomi suatu negara sudah baik keberadaan wirausaha tidak terlalu dibutuhkan (memotivasi orang untuk menjadi seorang wirausaha sudah kurang menarik). Kedua, kemampuan dan motivasi individu untuk memulai sebuah usaha dan pandangan masyarakat tentang wirausaha. Ketiga, pertumbuhan tinggi kewirausahaan dan persaingan antar negara tentang seberapa inovatif usaha tersebut. [4]

Kewirausahaan menurut GEM merupakan sebuah proses yang memiliki tahapan-tahapan yang berbeda (Gambar 2.1). Tahapan-tahapannya antara lain adalah dimulai dari niat mendirikan usaha, menjalankan usaha dan yang terakhir adalah berhentinya usaha yang dibuat. Tahapan pertama yaitu wirausaha *potential*. Wirausaha *potential* merupakan individu yang berpotensi untuk menjadi wirausaha, mereka percaya bahwa mereka memiliki kemampuan untuk memulai usaha, individu yang melihat kesempatan untuk berwirausaha, dan individu yang tidak takut gagal dalam memulai suatu usaha. Tahapan kedua yaitu wirausaha *nascent*. Wirausaha *nascent* ini merupakan tahapan

dimana seseorang memulai usahanya dalam waktu kurang dari tiga bulan. Tahapan ketiga yaitu wirausaha *new business owner*. Wirausaha *new business owner* merupakan wirausaha *nascent* yang usia bisnisnya sudah lebih dari 3 bulan tetapi kurang dari tiga tahun.



Gambar 2.1: Fase Wirausaha

Wirausaha *nascent* dan wirausaha *new business owner* masuk ke dalam TEA (Total Early-Stage Entrepreneurial Activity). TEA merupakan persentase populasi antara usia 18 sampai 64 tahun yang berada pada tahap memulai usaha maupun pemilik bisnis yang waktunya kurang dari 42 bulan [5]. Tahapan terakhir adalah wirausaha *established* yaitu seseorang yang sudah menjalankan usahanya lebih dari tiga tahun dan tentunya sudah bisa menggaji orang.[4]

Di setiap fase terdapat kemungkinan wirausahawan berhenti menjalankan usahanya. Berbagai faktor dapat mempengaruhi keberlangsungan wirausaha ini, diantaranya yaitu kondisi sosio-demografi, sektor industri, pertumbuhan wirausaha, inovasi,dll. Terdapat dua tipe atribut internal yang dimiliki setiap wirausahawan. Tipe pertama yaitu atribut umum seperti umur, pendapatan, pendidikan, bidang usaha, dll. Tipe kedua yaitu atribut individual dari GEM yang digunakan sebagai indikator kewirausahaan [2]. Penjelasan beberapa indikator akan dijelaskan pada tabel 2.1 dan tabel 2.2

Tabel 2.1: Indikator Kewirausahaan

Indikator	Deskripsi
New Product Early-stage Entrepreneurial (TEA) Activity	persentase dari TEA yang mengindikasikan bahwa produk atau jasa mereka masih baru
Growth Expectation Early-stage Entrepreneurial Activity : Relative Prevalence	persentase dari TEA yang berharap untuk meperkerjakan paling sedikit lima karyawan dalam waktu lima tahun kedepan
Informal Investors Rate	persentase dari populasi berusia 18-64 yang telah menyediakan dana untuk sebuah usaha baru, didirikan oleh orang lain, dalam waktu 3 tahun terakhir.
Total Early-stage Entrepreneurial Activity for Female Working Age Population	persentase dari populasi wanita berusia 18-64 yang antara lain merupakan seorang wirausaha <i>nascent</i> atau pemilik manager dari sebuah usaha baru.
Total Early-stage Entrepreneurial Activity for Male Working Age Population	persentase dari populasi pria berusia 18-64 yang antara lain merupakan seorang wirausaha <i>nascent</i> atau pemilik manager dari sebuah usaha baru.
Improvement-Driven Opportunity Entrepreneurial Activity : Relative Prevalence	persentase orang yang terlibat dalam TEA yang mengklaim bahwa mereka didorong oleh kesempatan, bukan karena kurangnya pilihan pekerjaan.
Necessity-Driven Entrepreneurial Activity : Relative Prevalence	persentase orang yang terlibat dalam TEA yang berwirausaha karena mereka tak punya pilihan pekerjaan lain
Established Business Ownership Rate	Persentase dari populasi berusia 18-64 yang merupakan pemilik manager dari sebuah usaha mapan dan sudah menghasilkan gaji atau untung apapun ke pemiliknya selama lebih dari 42 bulan.
Total Early-stage Entrepreneurial Activity	persentase dari populasi berusia 18-64 yang merupakan wirausaha <i>nascent</i> .
New Business Ownership Rate	Persentase dari populasi 18-64 yang merupakan pemilik manager dari sebuah usaha mapan yang sudah menghasilkan gaji atau untung selama lebih dari 3 bulan tetapi tidak lebih dari 42 bulan.
Nascent Entrepreneurship Rate	Persentase dari populasi 18-64 yang merupakan wirausaha <i>nascent</i> terlibat secara aktif memulai suatu usaha yang mereka miliki sendiri/bersama.
Media Attention for Entrepreneurship	persentase dari populasi berusia 18-64 yang setuju dengan pernyataan bahwa di negara mereka, mereka sering melihat atau mendengar di media tentang usaha baru yang sukses.
High status successful Entrepreneur	persentase dari populasi berusia 18-64 yang setuju dengan pernyataan bahwa di negara mereka, wirausaha yang sukses dihormati dan bercitra tinggi.
Entrepreneurship as Desirable Career	persentase dari populasi berusia 18-64 yang setuju dengan pernyataan bahwa di negara mereka, kebanyakan orang mempertimbangkan untuk memulai usaha baru sebagai karir yang diinginkan.
Know Startup Entrepreneur Rate (Role Model)	persentase dari populasi berusia 18-64 yang kenal seseorang yang mendirikan suatu usaha dalam waktu 2 tahun terakhir secara pribadi.

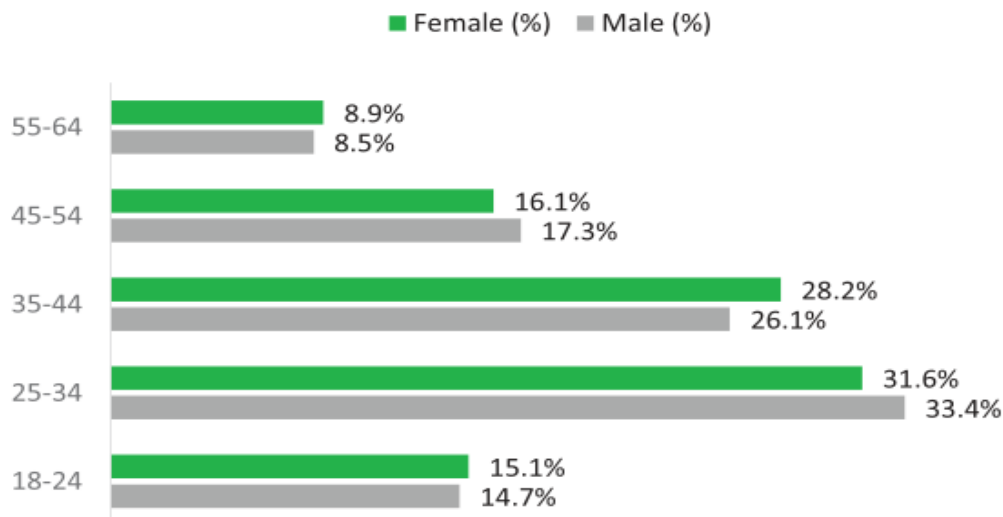
Tabel 2.2: Lanjutan Indikator Kewirausahaan

Entrepreneurial Intention	persentase dari populasi berusia 18-64 (individu yang terlibat dalam kegiatan wirausaha tidak termasuk) yang bertekad untuk mendirikan suatu usaha dalam waktu tiga tahun kedepan
Fear of Failure Rate	persentase dari populasi berusia 18-64 dengan perceived opportunities yang positif mengindikasikan bahwa takut pada kegagalan dapat menghambat mereka dalam mendirikan suatu usaha
Perceived Opportunities	persentase dari populasi berusia 18-64 yang melihat kesempatan bagus untuk memulai suatu usaha di daerah tempat tinggal mereka
Perceived Capabilities	persentase dari populasi berusia 18-64 yang merasa mempunyai kemampuan dan pengetahuan yang cukup untuk mendirikan suatu usaha

Indikator-indikator menurut GEM yang paling berpengaruh dalam perkembangan kewirausahaan di Indonesia yaitu Perceived Capabilities, Role Model, Perceived Opportunities, Entrepreneurial of Intention yang terdiri dari High Status of Successful dan Media Attention, serta indikator terakhir yaitu Fear of Failure.

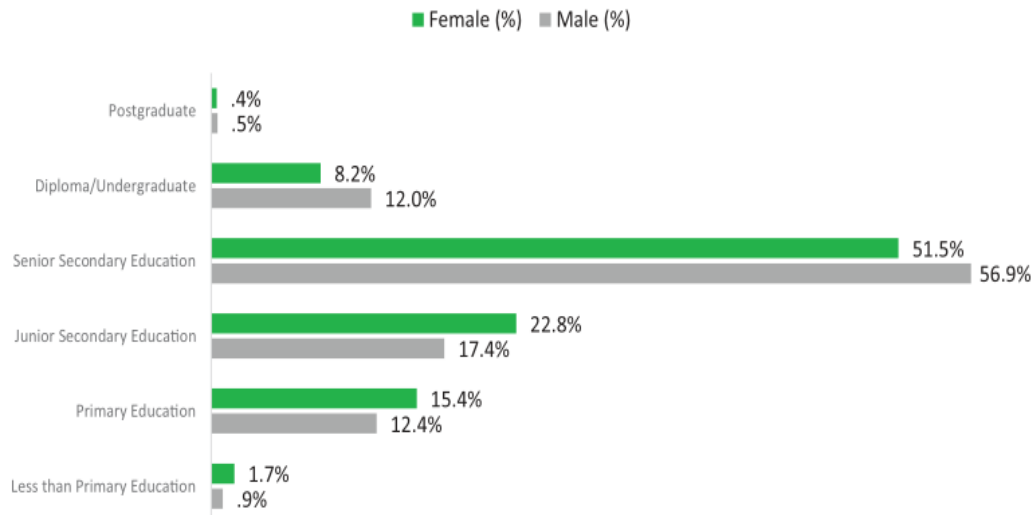
Berikut contoh data usia, pendidikan, pendapatan dan lokasi yang diambil dari GEM tahun 2013 [4].

1. Data Perceived Capabilities



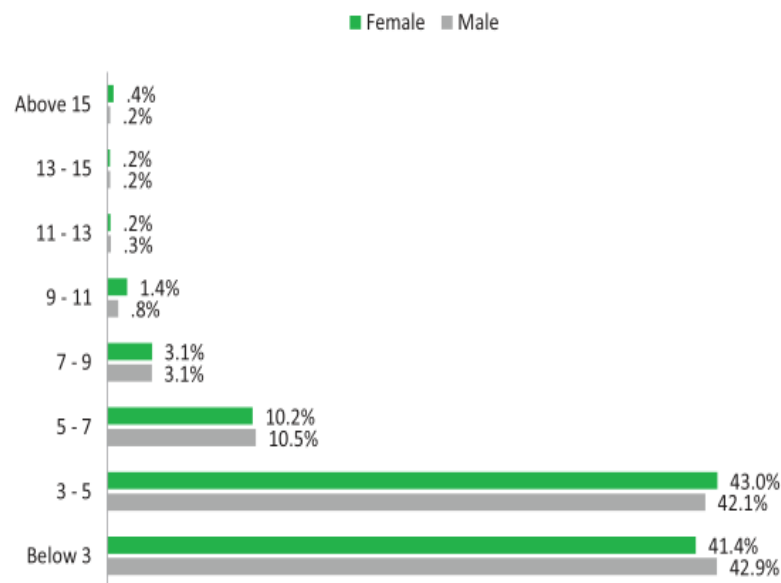
Gambar 2.2: Komposisi perceived capabilities untuk selang usia yang berbeda

Dapat dilihat pada gambar 2.2 bahwa Perceived Capabilities (percaya bahwa mereka memiliki kemampuan dan pengalaman dalam memulai usaha baru) tertinggi terletak pada mereka yang berusia 25 sampai 34 tahun. Perceived Capabilities terendah terletak pada mereka yang berada pada usia 55 sampai 64 tahun.



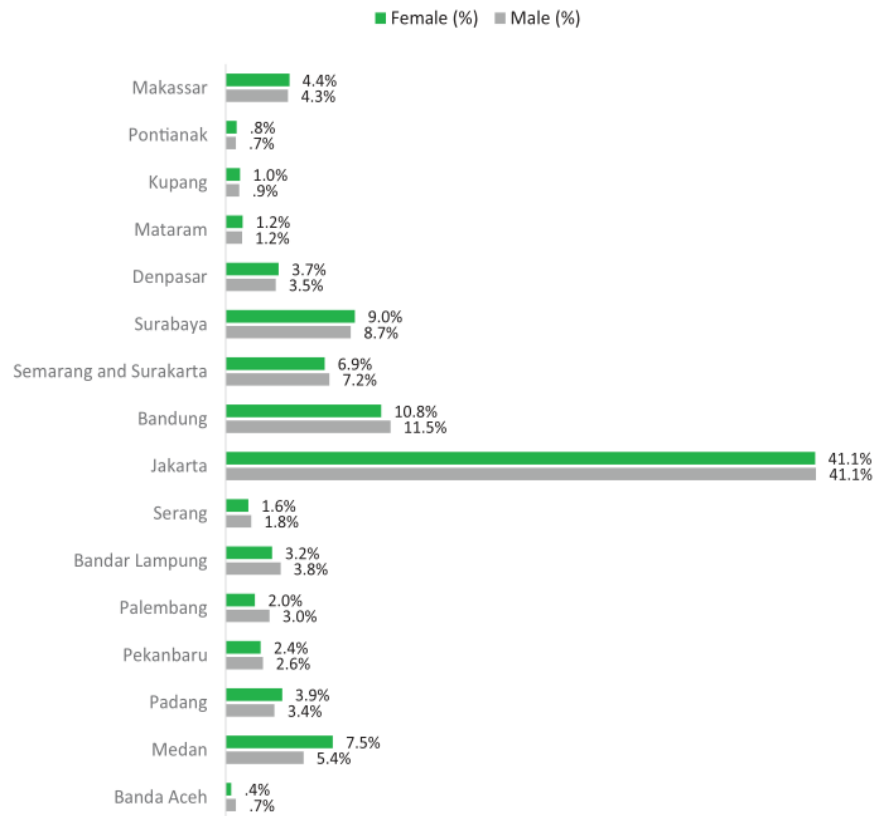
Gambar 2.3: Komposisi perceived capabilities untuk tingkat pendidikan yang berbeda

Dapat dilihat pada gambar 2.3 dijelaskan bahwa individu yang memiliki Perceived Capabilities tertinggi yaitu pada mereka yang telah menyelesaikan Sekolah Menengah Atas. Namun, Perceived Capabilities cenderung rendah bagi mereka yang menyelesaikan pendidikan ditingkat Universitas.



Gambar 2.4: Komposisi perceived capabilities untuk tingkat pendapatan

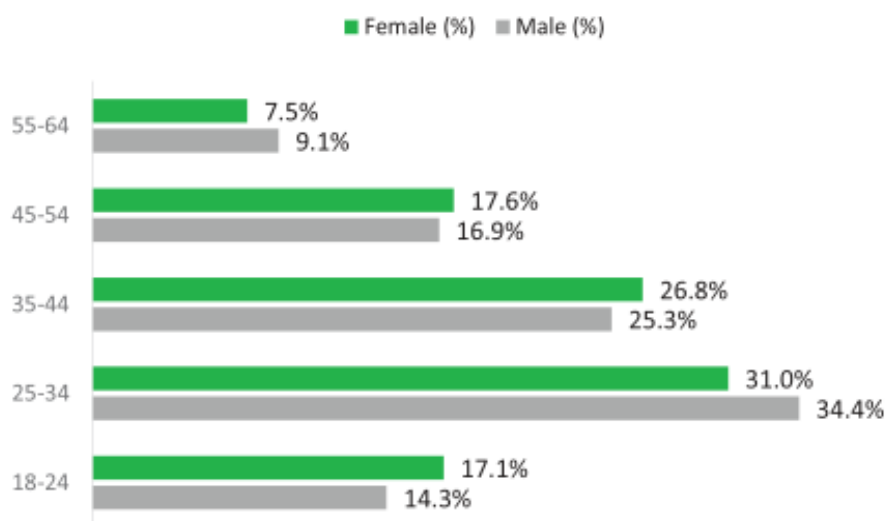
Dapat dilihat pada gambar 2.4 bahwa Perceived Capabilities tertinggi terletak pada mereka yang memiliki pendapatan di bawah 7 juta. Perceived Capabilities terendah terletak pada mereka yang pendapatannya diatas 11 juta.



Gambar 2.5: Komposisi perceived capabilities untuk wilayah Indonesia

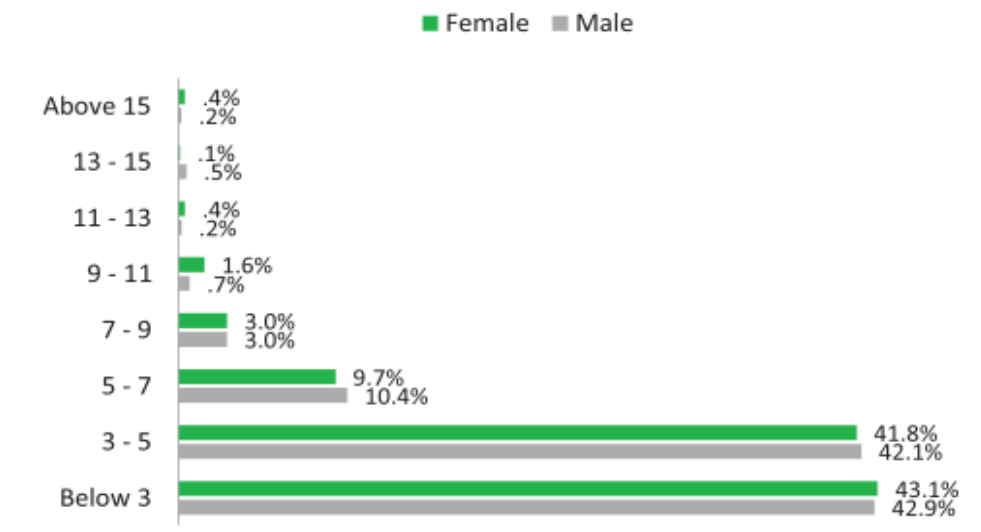
Dapat dilihat pada gambar 2.5 dijelaskan bahwa Jakarta memperoleh Perceived Capabilities tertinggi yang artinya banyak orang di Jakarta yang percaya memiliki kemampuan, pengetahuan dan pengalaman untuk memulai usaha baru. Sedangkan Banda Aceh memperoleh Perceived Capabilities terendah untuk wanita sebesar 0.4% dan untuk pria memiliki dua wilayah yang Perceived Capabilitiesnya rendah yaitu Pontianak dan Banda Aceh masing-masing sebesar 0.7%.

2. Data Role Model



Gambar 2.6: Komposisi role model untuk umur

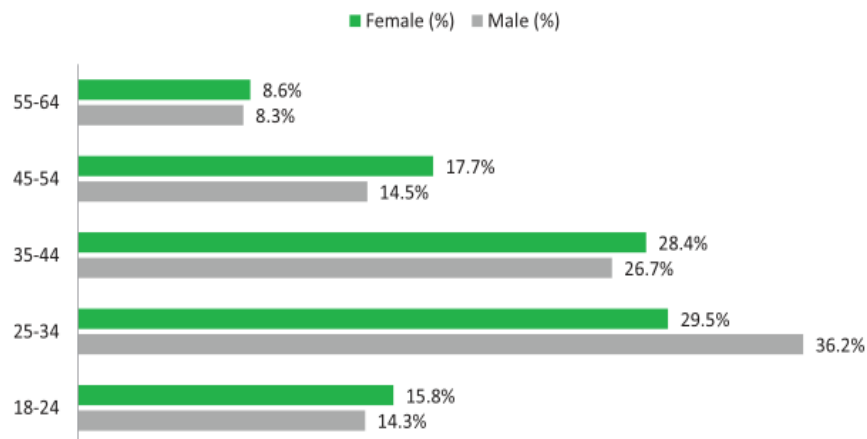
Pada gambar 2.6 dijelaskan individu yang memahami Role Model tertinggi yaitu oleh pria pada selang umur 25 sampai 34 tahun sebesar 34.4% sedangkan untuk wanita sebesar 31.0%. Pemahaman Role Model terendahnya yaitu pada selang waktu 55 sampai 64 tahun yang masing-masing nilainya yaitu pria 9.1% dan wanita 7.5%.



Gambar 2.7: Komposisi role model untuk tingkat pendapatan yang berbeda

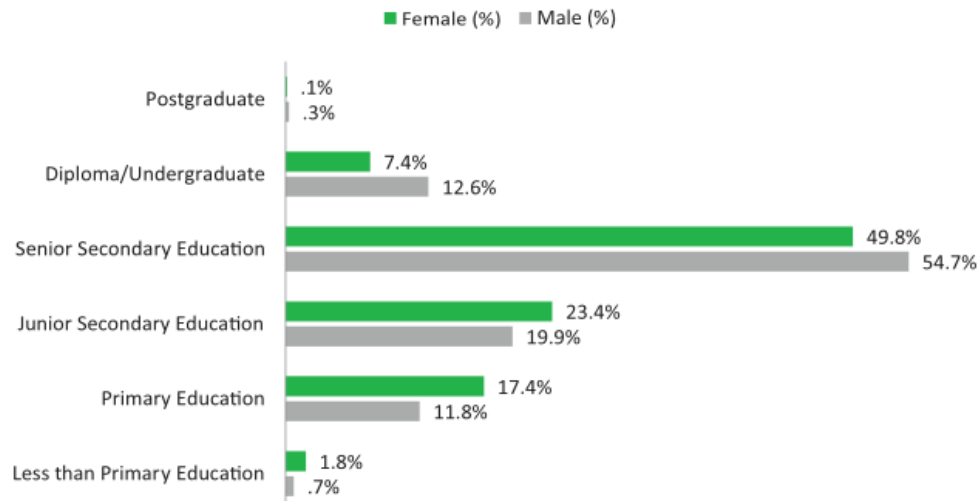
Pada gambar 2.7 dijelaskan Role Model memiliki peran penting terhadap tingkat pendapatan dibawah 7 juta rupiah. Pada tingkat pendapatan di atas 15 juta rupiah, wanita lebih mempertimbangkan Role Model dibandingkan pria.

3. Data Perceived Opportunities



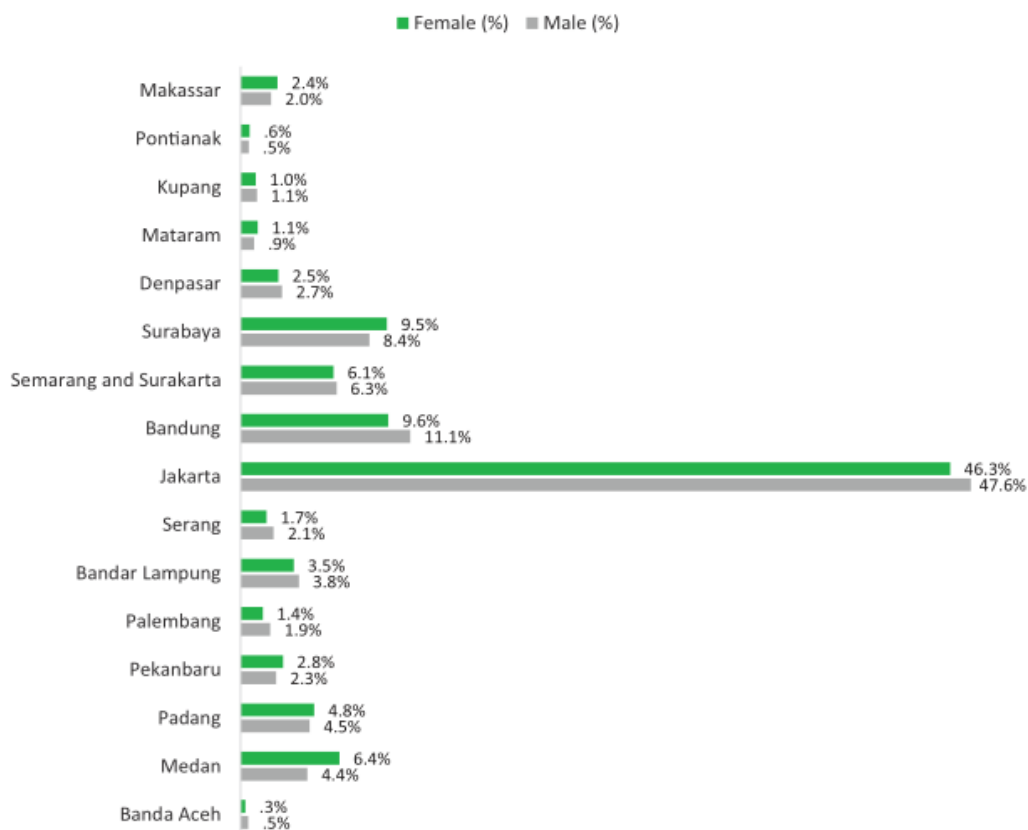
Gambar 2.8: Komposisi Perceived Opportunities usia wanita dan pria

Seperti dapat dilihat pada gambar 2.8, diantara semuanya yang melihat adanya peluang baik untuk memulai usaha baru yaitu pria berusia antara 25 sampai 34 tahun sebesar 36.2%, nilai untuk pria memiliki nilai lebih tinggi dibandingkan wanita. Sedangkan pada umur di atas 34 tahun, wanita memiliki nilai yang lebih tinggi dibandingkan pria.



Gambar 2.9: Komposisi perceived opportunities untuk tingkat pendidikan yang berbeda

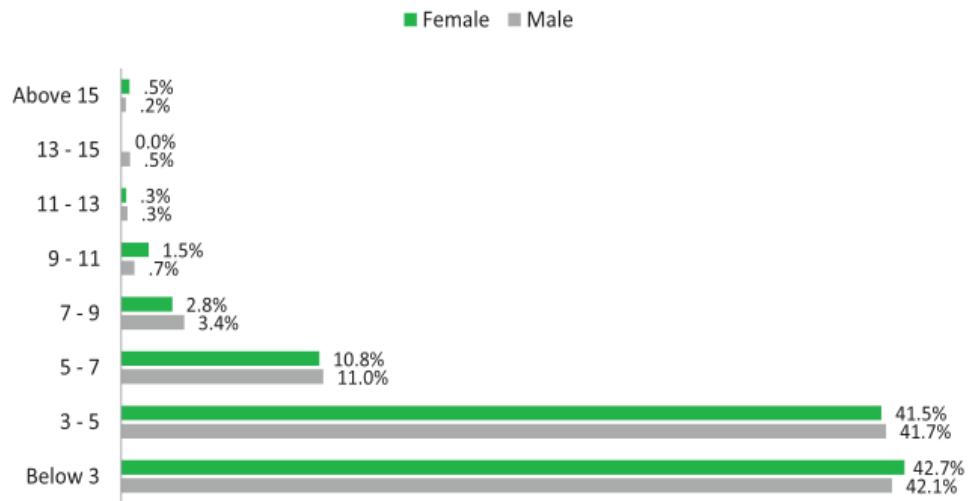
Gambar 2.9 menjelaskan yang memiliki Perceived Opportunities tertinggi yaitu mereka yang menyelesaikan pendidikannya di sekolah menengah atas, komposisi nilai untuk pria lebih tinggi dibandingkan wanita. Perceived Opportunities akan semakin menurun jika tingkat pendidikannya semakin tinggi.



Gambar 2.10: Komposisi Perceived Opportunities untuk wilayah Indonesia

Gambar 2.10 menjelaskan bahwa orang-orang yang tinggal di wilayah Jakarta memiliki Perceived Opportunities tertinggi dibandingkan kota-kota yang lain. Perceived Opportunities

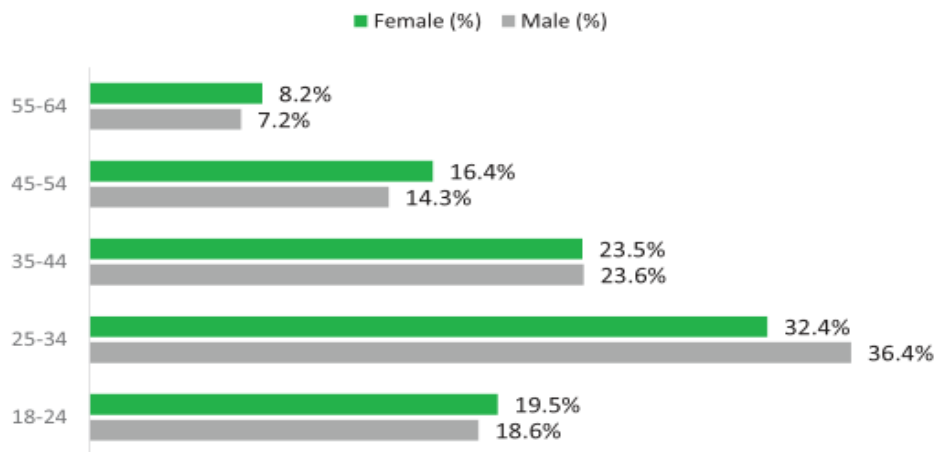
cenderung rendah pada wilayah-wilayah di luar pulau Jawa seperti pada wilayah Banda Aceh dan Pontianak.



Gambar 2.11: Komposisi Perceived Opportunities untuk tingkat pendapatan

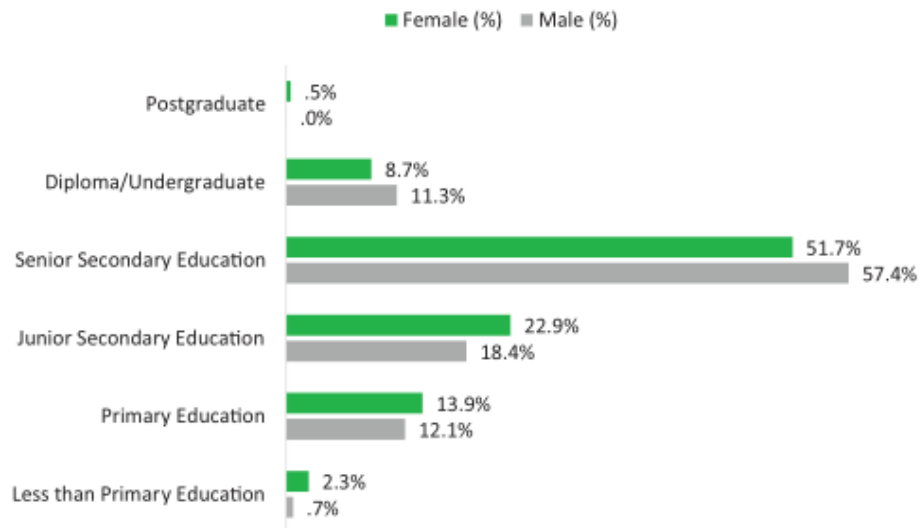
Gambar 2.11 memperlihatkan bahwa mereka yang pendapatannya di bawah 7 juta rupiah memiliki Perceived Opportunities lebih tinggi dibandingkan pendapatan di atas 7 juta rupiah. Rata-rata, wanita dengan pendapatan lebih dari 15 juta rupiah lebih bisa melihat adanya kesempatan memulai usaha baru dibandingkan pria.

4. Data Fear of Failure



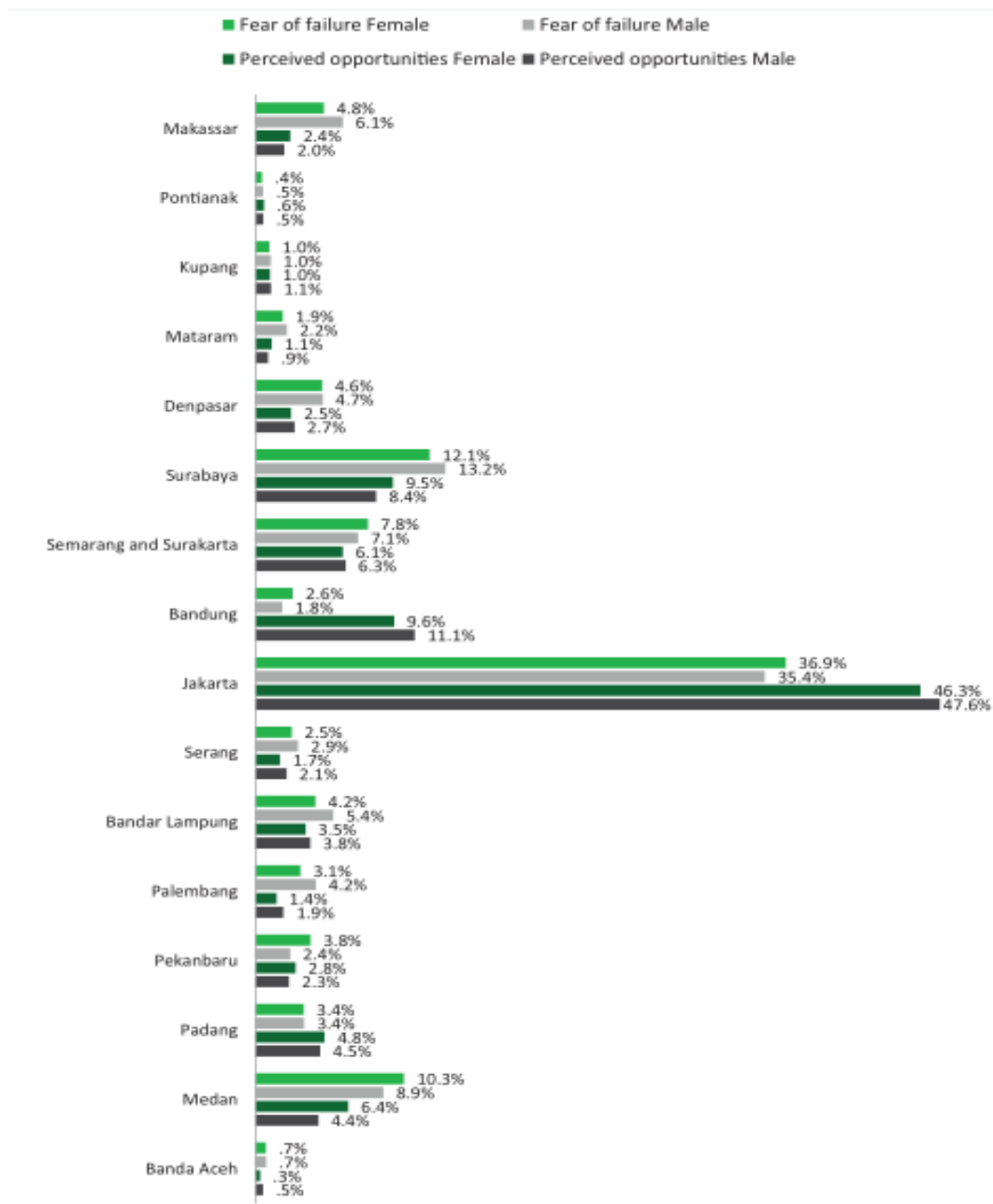
Gambar 2.12: Komposisi Fear of Failure untuk usia wanita dan pria

Dapat dilihat pada gambar 2.12, Fear of Failure tertinggi dimiliki oleh pria berumur antara 25 sampai 34 tahun. Wanita pada usia di atas 44 tahun memiliki Fear of Failure lebih tinggi dibandingkan pria.



Gambar 2.13: Komposisi Fear of Failure untuk tingkat pendidikan

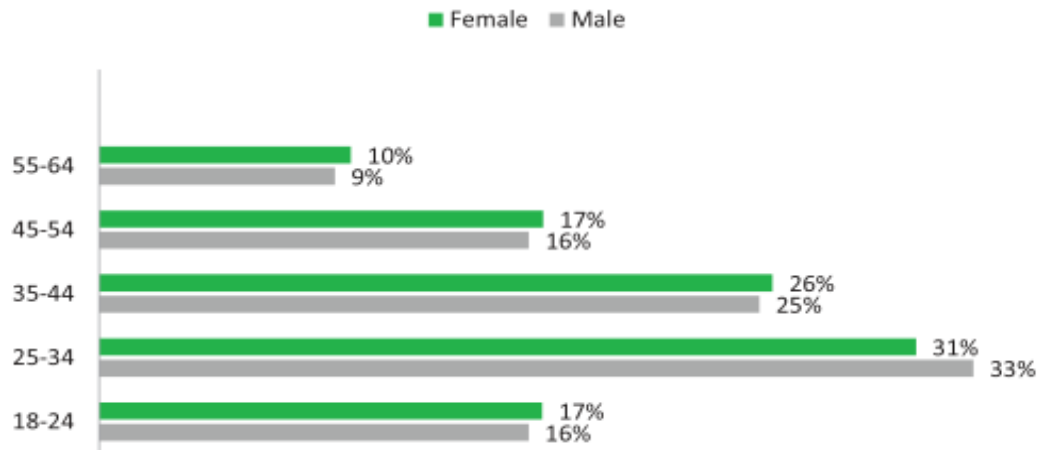
Pada gambar 2.13, Fear of Failure tertinggi dimiliki oleh mereka yang menyelesaikan pendidikannya pada sekolah menengah atas. Semakin tinggi tingkat pendidikan, Fear of Failure menjadi menurun.



Gambar 2.14: Komposisi Fear of Failure untuk wilayah Indonesia

Pada gambar 2.14, sama seperti faktor psikologis lainnya ibukota Indonesia yaitu Jakarta menjadi nilai tertinggi untuk Fear of Failure daripada kota-kota lainnya.

5. Data High Status of Successful



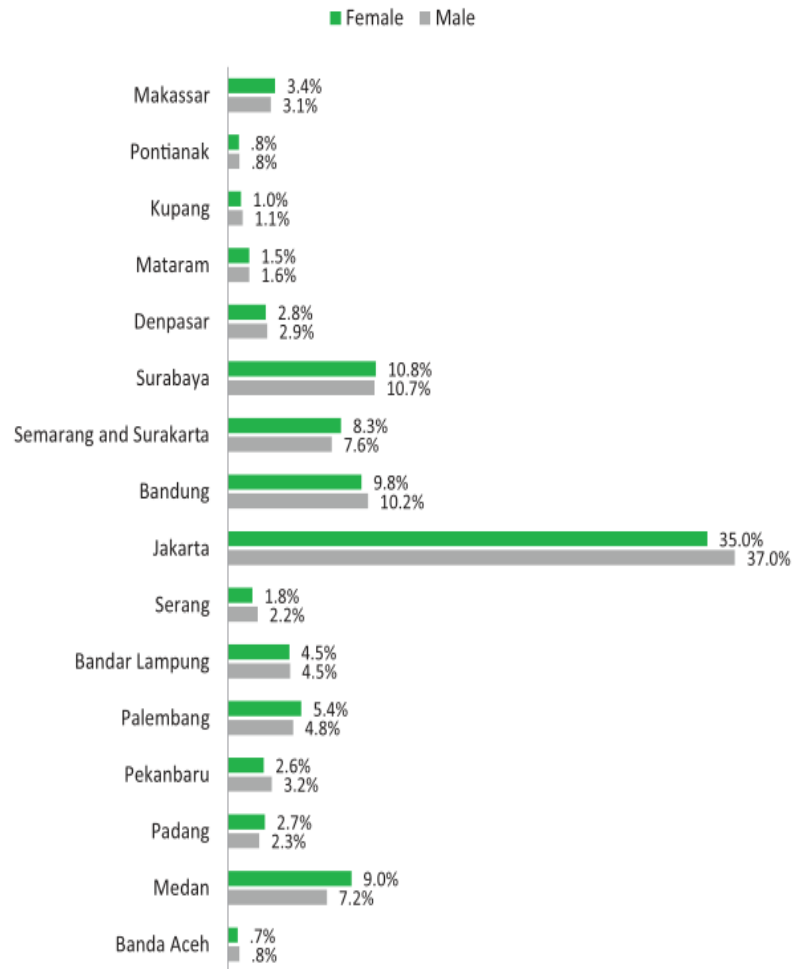
Gambar 2.15: Komposisi High Status of Successful untuk kategori usia

Dilihat pada gambar 2.15, individu pada usia 25 sampai 34 tahun memiliki persepsi positif bahwa pengusaha yang sukses dihormati dan bercitra tinggi.

Tabel 2.3: Komposisi High Status of Successful untuk tingkat pendidikan

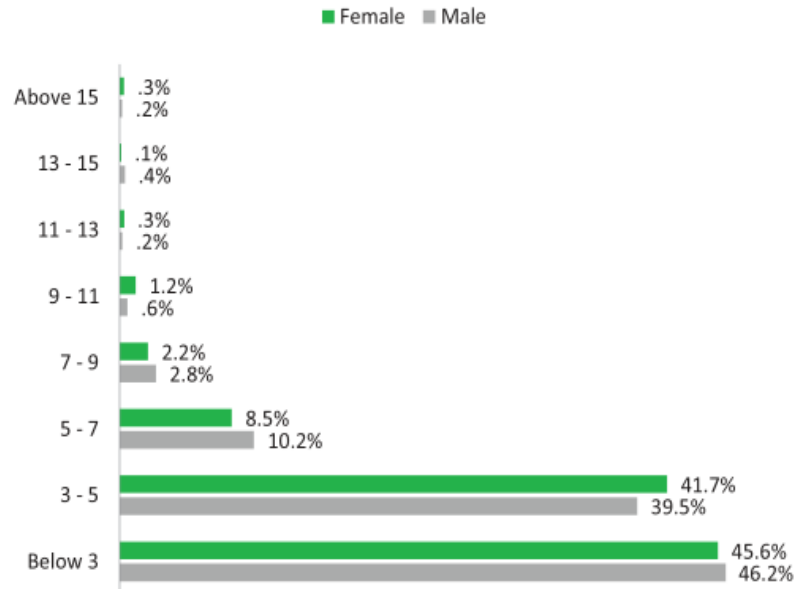
Tingkat Pendidikan	Pria	Wanita
Tidak Tamat Pendidikan Dasar	1%	2%
Pendidikan Dasar	12%	15%
Pendidikan Menengah Awal	19%	23%
Pendidikan Menengah Lanjutan	56%	52%
Diploma	11%	8%
Pascasarjana	0%	0%

Pada tabel 2.3, dapat dievaluasi bahwa wanita dengan tingkat pendidikan rendah memiliki persepsi lebih tinggi bahwa pengusaha yang sukses akan dihormati. Untuk mereka yang berada pada tingkat pendidikan menengah lanjutan, pria memiliki persepsi lebih tinggi mengenai hal tersebut daripada wanita.



Gambar 2.16: Komposisi High Status of Successful berdasarkan kota tinggal

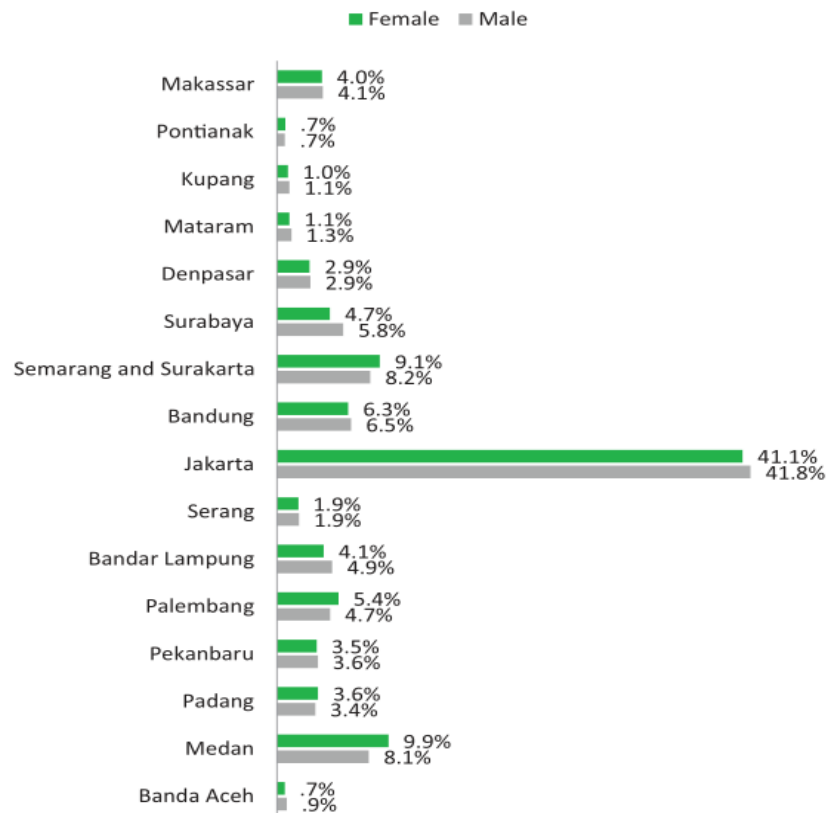
Pada gambar 2.16, orang-orang yang tinggal di kota Jakarta memiliki persepsi lebih tinggi mengenai pengusaha sukses memiliki status tinggi. Selanjutnya akan diteruskan oleh kota Bandung, Surabaya, dsb. Kota yang berada diluar pulau Jawa memiliki persepsi rendah dibandingkan kota-kota yang ada di pulau Jawa.



Gambar 2.17: Komposisi High Status of Successful berdasarkan tingkat pendapatan

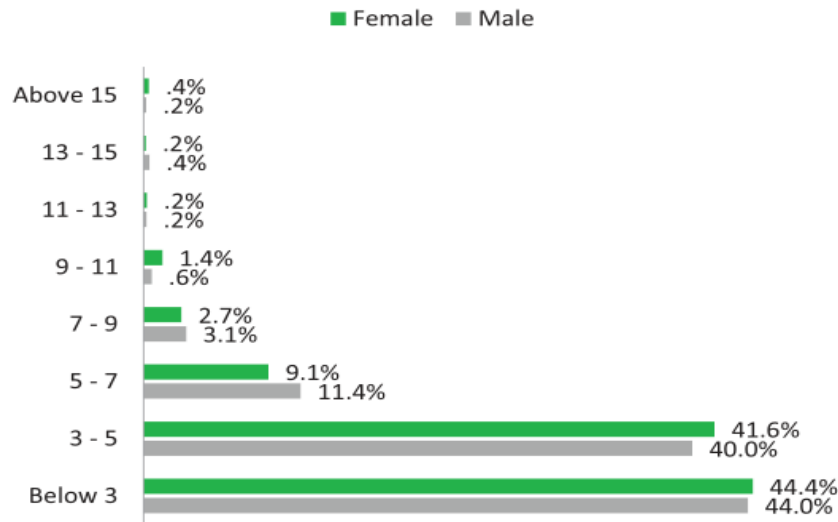
Dapat dilihat pada gambar 2.17, orang-orang dengan pendapatan di bawah 7 juta rupiah memiliki persepsi lebih tinggi mengenai High Status of Successful dibandingkan mereka yang memiliki pendapatan lebih dari 7 juta rupiah. Data selanjutnya yaitu dari Media Attention.

6. Data Media Attention



Gambar 2.18: Komposisi Media Attention berdasarkan kota tinggal

Dilihat pada gambar 2.18, dapat disimpulkan walaupun orang-orang yang berada di Jakarta memiliki persepsi lebih tinggi pada niat media untuk melaporkan cerita usaha yang sukses, persepsi tertinggi kedua justru terletak pada daerah di luar pulau jawa yaitu kota Medan.



Gambar 2.19: Komposisi Media Attention berdasarkan tingkat pendapatan

Pada gambar 2.19, dapat disimpulkan bahwa mereka yang memiliki pendapatan di bawah 7 juta rupiah memiliki persepsi lebih tinggi pada niat media untuk melaporkan cerita usaha yang sukses dibandingkan dengan mereka yang memiliki pendapatan di atas 7 juta rupiah.

Selain indikator wirausahawan, faktor publik juga mempengaruhi pertumbuhan wirausaha di Indonesia. Berikut data dari faktor publik yang diambil dari GEM 2013.

Tabel 2.4: Faktor Publik

Faktor Publik	Bobot
Keuangan terkait dengan kewirausahaan	3.06
Kebijakan pemerintah terkait ekonomi	2.69
Kebijakan pemerintah terkait pajak	2.22
Program Pemerintah	2.53
Pendidikan kewirausahaan pada SD dan SMP	2.54
Pendidikan kewirausahaan pada SMK, professional dan universitas	3.3
Transfer penelitian dan pengembangan	2.31
Infrastruktur komersial dan legal	3.25
Dinamika Pasar	3.92
Keterbukaan Pasar	2.82
Infrastruktur fisik dan akses layanan	3.45
Norma sosial dan budaya	3.29

2.2 Cellular Automata

Pertama-tama kita akan membahas tentang *Cellular Automata* agar dapat memahami dasar dari *Entrepreneurial Cellular Automata*. *Cellular Automata* (CA) diperkenalkan pertama kali oleh Ulam dan von Neumann pada tahun 1940. *Cellular Automata* sendiri merupakan model matematis untuk sistem yang terdapat banyak komponen sederhana bertindak bersama untuk menghasilkan pola perilaku yang rumit [6]. Sebuah CA terdiri atas sekumpulan sel, tersusun dalam larik-larik (*grid*).

Setiap sel mempunyai satu dari sejumlah *state* (kondisi) yang mungkin. *State* dapat berubah sesuai dengan aturan tertentu. Perubahan *state* dari sebuah sel dipengaruhi oleh *state* dari sel-sel di sekitarnya atau disebut dengan sel tetangga.

2.2.1 Dimensi CA

1. CA Satu Dimensi



Gambar 2.20: CA 1 Dimensi

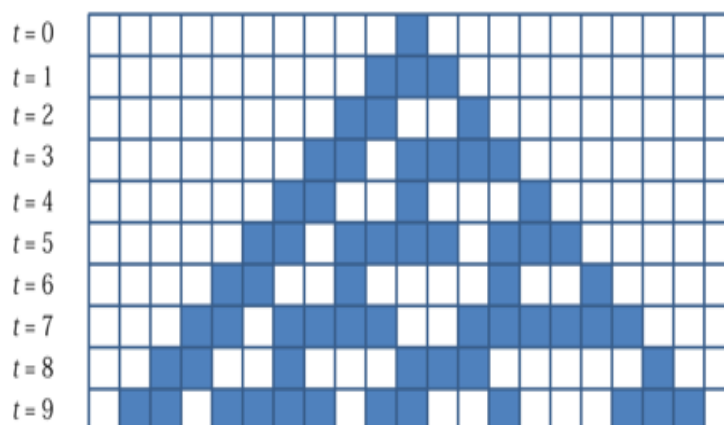
Cellular Automata satu dimensi adalah *cellular automata* yang ruang selnya berupa array satu dimensi, sehingga masing-masing sel hanya memiliki dua tetangga yang tepat bersebelahan, kecuali sel paling pinggir yang hanya mempunyai satu tetangga. CA satu dimensi biasanya memakai aturan yang diusulkan oleh Wolfram. Sebagai contoh berikut aturan no. 30 diberikan pada gambar 2.21



Gambar 2.21: Aturan 30 dari Wolfram

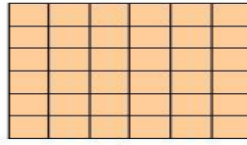
Cara membaca aturan tersebut adalah pada baris pertama terdapat 3 sel pada suatu saat (iterasi) tertentu, sel yang ditinjau adalah sel yang berada di tengah. Tetangga dari sel tersebut yaitu tetangga kiri dan kanan. Baris kedua menunjukkan keadaan sel pada *state* berikutnya. Sebagai contoh pada gambar paling kiri, sel pada bagian tengah (gelap) mempunyai tetangga kiri gelap dan tetangga kanan gelap maka iterasi berikutnya *state* sel tersebut berubah menjadi putih.

Sebagai ilustrasi, pada gambar 2.22 diberikan contoh penerapan aturan 30 dari Wolfram yang dimulai dari kondisi awal ($t=0$) dengan sel gelap yang berada di tengah hingga $t=9$. [2]



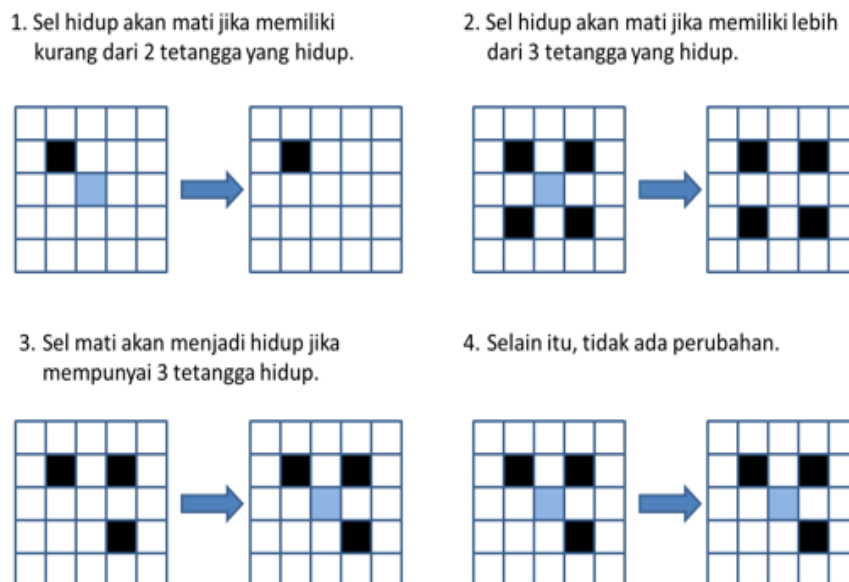
Gambar 2.22: Ilustrasi penerapan aturan 30 dari Wolfram

2. CA Dua Dimensi



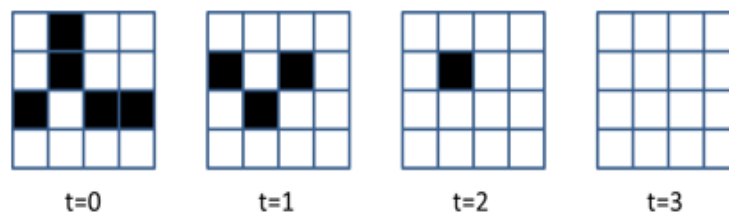
Gambar 2.23: CA 2 Dimensi

Cellular Automata dua dimensi adalah *cellular automata* yang ruang selnya biasanya berupa matriks, sehingga masing-masing sel memiliki lebih dari dua tetangga. CA dua dimensi yang sangat terkenal adalah Conway's *Game of Life*. Setiap sel pada CA menggambarkan suatu individu yang dapat berada pada *state* hidup atau mati. Sel hidup dapat berubah menjadi mati dan sel mati dapat berubah menjadi sel hidup. Aturan dasar Conway's diberikan pada gambar 2.24



Gambar 2.24: Aturan Dasar Conway's Game of Life

Berikut ilustrasi Conway yang menggambarkan perubahan yang terjadi pada sekumpulan sel mulai dari kondisi awal ($t=0$) sampai dengan kondisi akhir ($t=3$) yang dilakukan secara iteratif. Banyaknya sel hidup pada kondisi awal berkurang sedikit demi sedikit sampai pada kondisi akhir tidak ada lagi sel hidup. [2]



Gambar 2.25: Ilustrasi Conway's Game of Life

2.2.2 Aplikasi CA

1. Bidang Transportasi

CA banyak digunakan untuk memodelkan lalu lintas, dengan tujuan utama biasanya adalah untuk mempelajari beban dari jalan-jalan di area tertentu. Contoh aplikasi CA di bidang transportasi ini adalah simulasi pengaturan lampu lalu lintas. Model dalam penelitian ini menggunakan CA 1 dimensi.

2. Bidang Kesehatan

Pada bidang kesehatan, CA juga sering digunakan untuk pemodelan penyebaran penyakit. Biasanya masalah penyebaran penyakit dimodelkan dengan CA dua dimensi dan menggunakan aturan Game of Life dari Conway. Contoh aplikasi yang diterapkan di dunia nyata yaitu simulasi infeksi virus influenza A menggunakan *cellular automaton*. Pada penelitian ini *cellular automata* yang digunakan adalah CA dua dimensi. CA yang dibangun akan memodelkan CA yang memiliki *lattice* berbentuk segienam sebagai penyederhanaan dari bentuk bola ke dalam dua dimensi, hal ini dikarenakan sel tubuh manusia berbentuk seperti bola. Pada penelitian ini digunakan batasan secara *periodic*, dengan asumsi sel yang berseberangan sebenarnya bersebelahan pada kondisi aslinya karena masing-masing virus hanya dapat menginfeksi jaringan tubuh tertentu saja. [7]

3. Bidang Lingkungan / Ekologi

CA juga dapat digunakan untuk pemodelan pada bidang lingkungan. Contoh penerapan *cellular automata* pada bidang lingkungan adalah simulasi dan pemodelan perubahan penggunaan lahan. Penelitian ini menggunakan algoritma DINAMICA, algoritma ini merupakan algoritma *cellular automata* hibrida yang mendukung pemodelan statistik untuk menemukan area yang berpotensi mengalami perubahan berdasarkan faktor pemicu yang telah ditentukan.

4. Bidang Sains

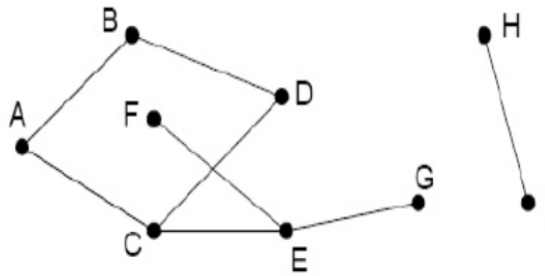
Pada bidang sains, khususnya fisika CA dapat digunakan untuk memodelkan pergerakan partikel dan juga permasalahan lainnya terkait dengan fisika kuantum. Pada bidang biologi, CA digunakan untuk memodelkan sel biologis.

2.3 Graf

Pembahasan ketiga yaitu tentang graf, hal ini dikarenakan struktur yang dipakai pada *Entrepreneurial Cellular Automata* adalah graf. Graf dalam matematika dan ilmu komputer adalah himpunan benda-benda yang disebut simpul (*vertex* atau *node*) yang terhubung oleh sisi (*edge*). Sebuah graf biasanya digambarkan dengan sekumpulan titik-titik yang dihubungkan oleh garis-garis. Suatu sisi dapat menghubungkan suatu simpul dengan simpul yang sama, sisi ini disebut dengan *loop*.

Graf biasanya dinyatakan sebagai $G = \langle V, E \rangle$, dimana V adalah simpul pada graf sedangkan E adalah sisi pada graf. Sebagai contoh definisi dari graf terdapat $V = A, B, C, D, E, F, G, H, I$ dan $E = (A, B), (A, C), (B, D), (C, D), (C, E), (E, F), (E, G), (H, I)$ berikut gambar graf sesuai dengan pernyataan V dan E di atas :

Gambar 2.26: Contoh Graf



Graf memiliki banyak jenis, jenis-jenis graf ini didasarkan pada ada tidaknya *loop* pada suatu graf dan sisi pada graf yang mempunyai orientasi arah. Berdasarkan ada tidaknya *loop* pada suatu graf digolongkan menjadi dua jenis :

1. Graf Sederhana

Graf ini tidak mempunyai sisi *loop*.

2. Graf tak-sederhana

Graf ini mempunyai sisi *loop*.

Berdasarkan orientasi arah pada sisi, secara umum graf dibedakan menjadi 2 jenis :

1. Graf tak-berarah

Graf yang sisinya tidak mempunyai arah. Pada graf ini urutan simpul tidak diperhatikan.

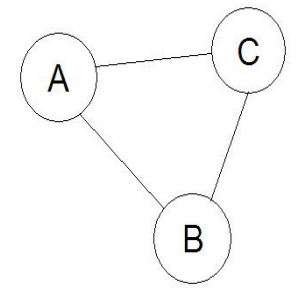
2. Graf berarah

Graf yang sisinya mempunyai arah. Pada graf ini urutan simpul diperhatikan. [8]

Sebuah graf dinyatakan sebagai struktur data yang terdiri dari simpul dan sisi yang membangun hubungan antar simpul. Terdapat dua macam representasi graf yaitu *adjacency list* dan *adjacency matrix*. [9]

2.3.1 Adjacency List

Adjacency list merupakan bentuk representasi dari seluruh sisi dalam sebuah graf sebagai suatu senarai (*linked list*). Simpul-simpul yang dihubungkan merupakan simpul-simpul yang saling terkait. Dalam implementasinya, *adjacency list* menggunakan *hash table* untuk menghubungkan satu simpul dengan simpul lain yang saling terkait. Contoh implementasi *adjacency list* yaitu sebagai berikut :



Gambar 2.27: Undirected Cyclic Graph

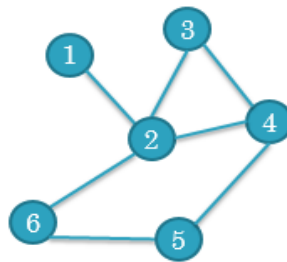
Graf pada gambar 2.27 dapat direpresentasikan melalui tabel 2.5 :

Tabel 2.5: Tabel Representasi Adjacency List

Vertex	Adjacency	Array of Adjacent
a	adjacent to	b,c
b	adjacent to	a,c
c	adjacent to	a,b

2.3.2 Adjacency Matriks

Adjacency Matrix merupakan representasi matriks $N \times N$ yang menyatakan hubungan antar simpul dalam suatu graf. Kolom dan baris menyatakan simpul-simpul, sedangkan nilai entri dari matriks menyatakan hubungan antar simpul. Contoh implementasi *adjacency matrix* pada graf tidak berarah yaitu sebagai berikut :

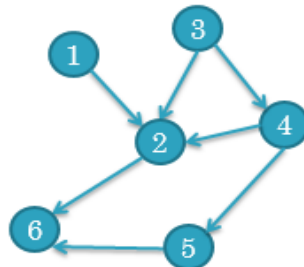
Gambar 2.28: *Undirected Cyclic Graph*

Graf pada gambar 2.28 dapat direpresentasikan melalui tabel 2.6 :

Tabel 2.6: Tabel Representasi Adjacency Matriks

v	1	2	3	4	5	6
1	0	1	0	0	0	0
2	1	0	1	1	0	1
3	0	1	0	1	0	0
4	0	1	1	0	1	0
5	0	0	0	1	0	1
6	0	1	0	0	1	0

Contoh *adjacency matrix* pada graf berarah yaitu sebagai berikut :

Gambar 2.29: *Directed Cyclic Graph*

Graf pada gambar 2.29 dapat direpresentasikan melalui tabel 2.7 :

Tabel 2.7: Tabel Representasi Adjacency Matriks

v	1	2	3	4	5	6
1	0	1	0	0	0	0
2	0	0	0	0	0	1
3	0	1	0	1	0	0
4	0	1	0	0	1	0
5	0	0	0	0	0	1
6	0	0	0	0	0	0

2.4 Entrepreneurial Cellular Automata

Entrepreneurial Cellular Automata merupakan pengembangan model dari *Cellular Automata* yang digunakan untuk mensimulasikan pertumbuhan kewirausahaan di Indonesia. Dalam kasus *Entrepreneurial Cellular Automata* (ECA), sel akan merepresentasikan wirausahawan dan ketetanggaannya akan merepresentasikan hubungan antar wirausahawan. Setiap wirausahawan mempunyai dua sifat atribut yaitu statis (nilainya tidak berubah) dan dinamis (nilainya dapat berubah). Contoh atribut statis adalah bidang usaha, kategori usaha, lokasi geografis dan jenis kelamin. Contoh atribut dinamis adalah usia, level wirausaha dan usia usaha.

Berikut definisi ECA :

Diberikan p himpunan nilai atribut: A_1, \dots, A_p dan sebuah indikator $Pub = p_1, \dots, p_m$, sebuah ECA M adalah sebuah tupel

$$M = (E, \alpha, N, \omega, \rho, \delta, \sigma)$$

dimana :

- $E = e_1, \dots, e_n$ adalah himpunan berhingga wirausahaan,
- $\alpha = \alpha_1, \dots, \alpha_p$ adalah himpunan berhingga atribut dimana setiap α_i didefinisikan sebagai $\alpha_i : E \rightarrow A_i$,
- $N = N_1, \dots, N_k$ adalah himpunan berhingga ketetanggaan dimana setiap N_i didefinisikan sebagai $N_i : E \times E \rightarrow \mathbb{R}$,
- $\omega = \omega_1, \dots, \omega_k$ adalah himpunan fungsi bobot atau nilai ketetanggaan dimana $\omega_i : N_i \rightarrow \mathbb{R}$ memetakan setiap fungsi ketetanggaan ke sebuah bilangan riil,
- $\rho = \rho_1, \dots, \rho_p$ adalah himpunan indikator publik dimana setiap ρ_i didefinisikan sebagai $\rho_i : p_i \rightarrow \mathbb{R}$,
- $\delta : \beta \rightarrow \beta$ adalah fungsi transisi state, dan
- $\sigma : N \rightarrow N$ adalah sebuah fungsi transformasi ketetanggaan. Contohnya, diasumsikan terdapat wirausahawan $e1$ dan $e2$ bertetanggaan pada waktu t , jika $e1$ berubah keadaannya pada $t + 1$ maka $e1$ dan $e2$ tidak lagi bertetanggaan pada saat $t + 1$.

Perubahan atribut dinamis dari waktu ke waktu didefinisikan dengan fungsi transisi atau δ . Fungsi transisi adalah aturan yang mengubah kondisi level wirausaha dari waktu sekarang dan waktu selanjutnya, aturan tersebut dapat dilihat pada tabel 2.8. Dalam fungsi transisi, atribut yang paling penting adalah level wirausaha karena atribut ini digunakan untuk menentukan perkembangan dari kewirausahaan. Selain itu fungsi transisi juga ditentukan oleh sebuah angka yang disebut *Continuity Index* (CIDx). CIDx adalah cara menentukan seorang wirausahawan akan meneruskan usahanya dalam waktu tertentu. CIDx dari seorang wirausahawan tidak hanya dipengaruhi oleh faktor dari dalam tetapi juga dipengaruhi oleh faktor dari luar. Faktor luar dipengaruhi oleh tetangganya dan faktor publik seperti kebijakan pemerintah, kondisi perekonomian dunia, dsb.

Seorang wirausahawan akan meneruskan usahanya jika CIDx-nya memenuhi nilai ambang tertentu. Berikut rumus *continuity index* :

$$CIDx_i(t) = a.Cint_i(t) + b.Cneg_i(t) + c.Cpub(t)$$

dimana a,b,c merupakan bilangan riil sedemikian sehingga $0 \leq a, b, c \leq 1$ dan $a + b + c = 1.0$.

Berikut penjelasan mengenai rumus *Continuity Index* :

1. $CIDx_i(t)$ merupakan perhitungan CIDx untuk individu wirausahawan dalam waktu tertentu (t).
2. $Cint_i(t)$ merupakan perhitungan dari kondisi internal dari individu wirausahawan yang berisi penjumlahan nilai dari atribut individual seperti :
 - Perceived Opportunities
 - Perceived Capabilities
 - Role Model
 - Entrepreneurial of Intention (High Status Successful Entrepreneurship dan Public Media Attention for Entrepreneurship)
 - Fear of Failure

Dari masing-masing atribut individual terdapat data umur, pendapatan, pendidikan, dll. Data tersebut dapat dilihat pada subbab 2.1. Dari data tersebut terdapat persentase untuk wanita dan pria berdasarkan atribut tertentu. Nilai itulah yang akan digunakan untuk menghitung kondisi internal individu wirausahawan.

3. $Cneg_i(t)$ merupakan perhitungan ketetangaan dari setiap individu wirausahawan dengan wirausahawan lainnya bergantung pada relasi ketetangaannya pada saat waktu tertentu. Relasinya yaitu lebih dari sama dengan, kurang dari sama dengan dan sama dengan. Relasi antar wirausaha berdasarkan pada atribut umum dari wirausahawan yaitu umur, level wirausaha, pendapatan dan pendidikan.

Contoh relasi kurang dari sama dengan adalah wirausaha A mempunyai umur 24 tahun, sedangkan wirausaha B mempunyai umur 30 tahun. Artinya yaitu wirausaha A mempunyai relasi umur kurang dari sama dengan wirausaha B, hal ini dikarenakan umur wirausaha A kurang dari umur wirausaha B.

Contoh relasi sama dengan adalah wirausaha A mempunyai level wirausaha *nascent*, sedangkan wirausaha B mempunyai level wirausaha *nascent*. Artinya wirausaha A memiliki relasi level sama dengan wirausaha B, hal ini dikarenakan mereka berada pada kedudukan level wirausaha yang sama.

Contoh relasi lebih dari sama dengan adalah wirausaha A berada pada tingkat pendidikan SMP, sedangkan wirausaha B berada pada tingkat pendidikan SMA. Artinya wirausaha B memiliki relasi pendidikan lebih besar sama dengan wirausaha A, hal ini dikarenakan wirausaha B memiliki tingkat pendidikan lebih besar atau lebih tinggi dibandingkan wirausaha A.

4. $Cpub(t)$ merupakan perhitungan dari kondisi publik seperti :
 - Keuangan terkait dengan kewirausahaan
 - Kebijakan pemerintah terkait ekonomi
 - Kebijakan pemerintah terkait pajak
 - Program Pemerintah
 - Pendidikan kewirausahaan pada SD dan SMP
 - Pendidikan kewirausahaan pada SMK, professional dan universitas

- Transfer penelitian dan pengembangan
- Infrastruktur komersial dan legal
- Keterbukaan Pasar
- Norma, Sosial dan Budaya
- Infrastruktur Fisik dan Akses Layanan
- Dinamika Pasar

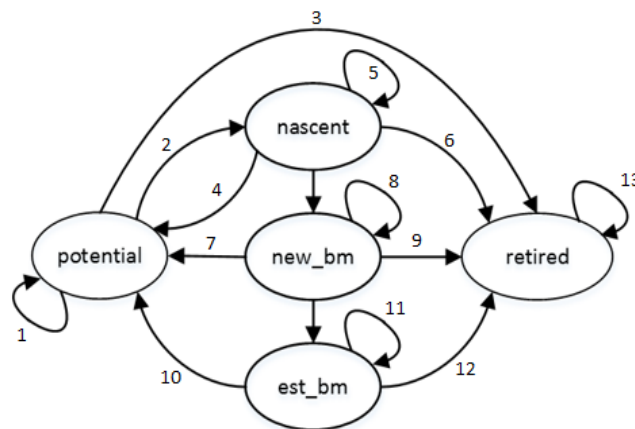
Perhitungannya berisi penjumlahan nilai dari kondisi publik yang masing-masing nilainya dapat dilihat pada data GEM 2013 yang dapat dilihat pada tabel 2.4 pada saat waktu tertentu.

Seperti yang sudah dijelaskan fungsi transisi merupakan sekumpulan aturan yang bertujuan untuk melakukan perubahan level wirausaha pada individu wirausahawan. Pada tabel 2.8 akan dijelaskan mengenai transisi level dengan menggunakan lambang-lambang $CIdx$, bl , a , b dan th untuk menyatakan *Continuity Index*, level, usia individu, usia usaha dan nilai ambang.

Tabel 2.8: Transisi Level Wirausaha

No	Waktu sekarang	Waktu berikutnya
1	$bl = \text{potential}, CIdx < th, a < 64 \times 12$	$bl = \text{potential}$
2	$bl = \text{potential}, CIdx \geq th, a < 64 \times 12$	$bl = \text{nascent}$
3	$bl = \text{potential}, a \geq 64 \times 12$	$bl = \text{retired}$
4	$bl = \text{nascent}, CIdx < th, a < 64 \times 12$	$bl = \text{potential}$
5	$bl = \text{nascent}, CIdx \geq th, b < 3$	$bl = \text{nascent}$
6	$bl = \text{nascent}, a \geq 64 \times 12$	$bl = \text{retired}$
7	$bl = \text{new_bm}, CIdx < th, a < 64 \times 12$	$bl = \text{potential}$
8	$bl = \text{new_bm}, CIdx \geq th, b < 42$	$bl = \text{potential}$
9	$bl = \text{new_bm}, a \geq 64 \times 12$	$bl = \text{retired}$
10	$bl = \text{est_bm}, CIdx < th, a < 64 \times 12$	$bl = \text{potential}$
11	$bl = \text{est_bm}, CIdx \geq th, a < 64 \times 12$	$bl = \text{est_bm}$
12	$bl = \text{est_bm}, a \geq 64 \times 12$	$bl = \text{retired}$
13	$bl = \text{retired}, a \geq 64 \times 12$	$bl = \text{retired}$

Berikut diagram transisi level wirausaha sesuai dengan fungsi transisinya masing-masing :



Gambar 2.30: Diagram Transisi Level Wirausaha

Berdasarkan diagram 2.30 selain keempat model kewirausahaan GEM, ditambahkan pula tingkatan wirausaha yang menyatakan wirausahawan di atas umur 64 tahun yaitu *retired*. *New*

business manager akan dinyatakan sebagai *new_bm* dan *est_bm* dinyatakan sebagai *manager of established business*.

BAB 3

ANALISIS

Pada bab ini akan dilakukan analisis mengenai pembangunan simulator pertumbuhan wirausaha dengan *Entrepreneurial Cellular Automata* (ECA). Pembahasan akan dimulai dari analisa pertumbuhan wirausaha di Indonesia yang menjadi pokok permasalahan. Lalu dari analisis ini akan dilanjutkan dengan analisis kebutuhan perangkat lunak agar mampu memodelkan pertumbuhan wirausaha di Indonesia.

3.1 Analisis Model Pertumbuhan Wirausaha dengan ECA

Analisis model pertumbuhan wirausaha bergantung terhadap nilai *Continuity Index*, nilai ambang (*threshold*), umur (*a*) dan usia bisnis (*bl*). Seperti yang sudah dijelaskan pada bab 2, *Continuity Index* adalah angka yang menentukan seorang wirausaha akan meneruskan usahanya atau tidak. Sedangkan nilai ambang berfungsi untuk acuan (patokan) perubahan wirausaha dari waktu ke waktu. Rumus CIDx ada pada bab 2 subbab 2.4.

Untuk mempermudah pemahaman mengenai *Continuity Index*, akan diberikan contoh simulasi dari data tidak real, yaitu terdapat nilai $a = 0.5$, $b = 0.4$ dan $c = 0.1$, nilai ambangnya 15, serta periodenya dalam waktu 5 bulan. Nilai dari faktor psikologis diasumsikan Perceived Opportunities bernilai 0.2, Perceived Capabilities bernilai 0.25, High Status of Successful bernilai 0.1, Public Media Attention bernilai 0.05, Role Model bernilai 0.3 dan Fear of Failure bernilai 0.1. Diasumsikan terdapat tiga wirausahawan dan berikut data dari masing-masing wirausaha :

Tabel 3.1: Data wirausahawan

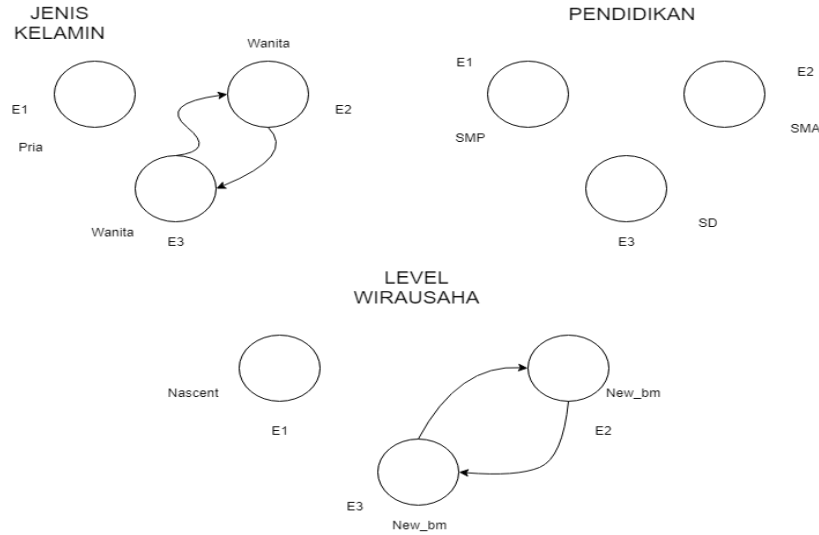
	Jenis Kela- min	Umur	Usia Bis- nis	Kateg- ori	Sub Katego- ri	Educa- tion	Lokasi	Income	Level
E1	P	18th	0	Minum- an	Minuman bersoda	SMP	Medan	5-7jt	Nascent
E2	W	30th	0	Tas	Tas anak-anak	SMA	Pekanbaru	3-5jt	New_bm
E3	W	45th	0	Makan- an	Makanan berat	SD	Palembang	7-9jt	New_bm

Asumsi ketetanggaan antara wirausaha satu dengan wirausaha lainnya hanya 3 atribut yaitu :

Tabel 3.2: Data Bobot Atribut

Atribut	Bobot
Level Wirausaha	30%
Pendidikan	40%
Jenis Kelamin	30%

Masing-masing tetangga relasinya yaitu sama dengan.



Gambar 3.1: Gambar ketetangaan tiga entrepreneur pada saat awal

Dalam simulasi ini terdapat 12 faktor publik yang dapat dilihat pada bab 2 pada tabel 2.4. Berikut rumus CIDx :

$$CIDx_i(t) = a.Cint_i(t) + b.Cneg_i(t) + c.Cpub(t)$$

Untuk lebih jelasnya berikut contoh perhitungan *continuity index* untuk E2 :

1. Perhitungan $Cint_2(0)$ atau bulan ke 1 :

- Mengambil nilai dari data GEM 2013 sesuai dengan data wirausahawan. Contoh untuk data wirausahawan 1 (E1), terdapat data wanita, 30 tahun, pendidikannya SMA, lokasi di Pekanbaru, pendapatan 3-5 juta rupiah dan level wirausahanya *new_bm*. Untuk data atribut individual *Perceived Opportunities* nilai untuk wanita 30 tahun adalah 29.5%, nilai untuk wanita berpendidikan SMA adalah 49.8%, nilai untuk wanita berlokasi di Pekanbaru adalah 2.8% dan wanita berpendapatan 3-5juta adalah 41.5%.
- Setelah mendapatkan nilai-nilai dari GEM 2013, nilai-nilai tersebut dijumlahkan lalu dikalikan dengan bobot masing-masing atribut individual. Dalam contoh langkah sebelumnya akan dikalikan dengan 0.2 (*Perceived Opportunities*), berikut penjumlahannya $((29.5 + 49.8 + 2.8 + 41.5) \times 0.2)$.
- Lakukan langkah pertama dan kedua untuk atribut individual lainnya (*Perceived Capabilities*, *Role Model*, *High Status of Successful*, *Media Attention* dan *Fear of Failure*).
- Setelah itu hasil dari masing-masing atribut individual akan dijumlahkan dan dikalikan dengan nilai a (0.5). Berikut perhitungannya :

$$CIDx_2(t=0) = 0.5 \times (((29.5 + 49.8 + 2.8 + 41.5) \times 0.2) + ((31.6 + 51.5 + 2.4 + 43) \times 0.25) + ((31 + 41.8) \times 0.3) + ((31 + 52 + 2.6 + 41.7) \times 0.1) + ((3.5 + 41.6) \times 0.05) + ((32.4 + 51.7 + 3.8) \times 0.1)) \quad (3.1)$$

- Perhitungan untuk $Cneg_2(0)$, kondisi ketetangaan pada saat awal dapat dilihat pada 3.1. Dapat dilihat bahwa E2 memiliki ketetangaan berdasarkan jenis kelamin dan level wirausaha dengan E3, sedangkan E2 tidak memiliki ketetangaan berdasarkan pendidikan maka akan diberi nilai 0. Untuk nilai ketetangaan berdasarkan jenis kelamin akan diberikan nilai $\frac{1}{2}$, 1 didapat dari jumlah wirausahawan yang bertetangaan dengan E2 yaitu E3, sedangkan 2 didapat dari total wirausahawan dikurangi 1. Untuk nilai ketetangaan berdasarkan level

wirausaha sama dengan nilai ketetangaan berdasarkan jenis kelamin. Setelah nilai tersebut didapatkan, nilai tersebut akan dikalikan dengan bobot masing-masing ketetangaan. Berikut perhitungannya : $(\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)$.

3. Perhitungan pada $C_{pub}(0)$:

- (a) Mengalikan nilai-nilai yang ada di tabel 2.4 dengan bobot masing-masing faktor publik. Bobot didapatkan dari masukan *user*. Contoh nilai dari faktor publik keuangan terkait dengan kewirausahaan adalah 3.06 lalu dikali dengan bobotnya, sebagai contoh bobotnya 10%. Maka perkaliannya adalah 3.06×0.1 .
- (b) Menjumlahkan hasil perkalian dari langkah pertama. Setelah dijumlahkan hasilnya dibagi dengan banyaknya faktor publik (12) lalu hasilnya dikali dengan nilai c (0.1). Berikut perhitungannya :

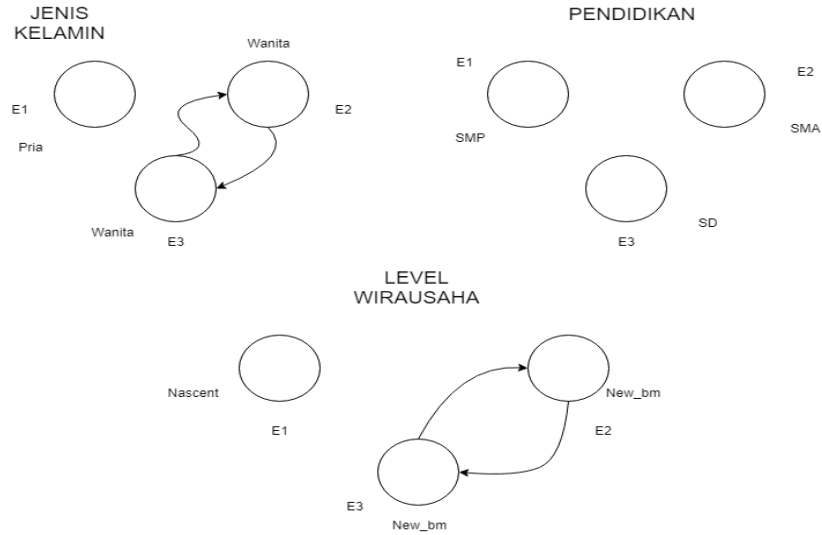
$$\begin{aligned} C_{pub}(0) &= 0.1 \times ((3.06 \times 0.1) + (2.69 \times 0.1) + (2.22 \times 0.1) + (2.53 \times 0.05) + (2.54 \times 0.1) + (3.3 \times 0.1) \\ &+ (2.31 \times 0.05) + (3.25 \times 0.05) + (3.92 \times 0.1) + (2.82 \times 0.05) + (3.45 \times 0.1) + (3.29 \times 0.1)) \\ &= 0.29925/12 = 0.0249375 \quad (3.2) \end{aligned}$$

Berikut perhitungan $CIDx(t=0)$:

$$\begin{aligned} CIDx_1(t=0) &= 0.5 \times (((14.3+4.4+19.9+11) \times 0.2) + ((14.7+17.4+5.4+10.5) \times 0.25) + ((14.3+10.4) \times 0.3) \\ &+ ((16+19+7.2+10.2) \times 0.1) + ((8.1+11.4) \times 0.05) + ((18.6+18.4+8.9) \times 0.1)) + 0.4 \times (0+0+0) \\ &+ 0.0249375 = 20.09243 \quad (3.3) \end{aligned}$$

$$\begin{aligned} CIDx_2(t=0) &= 0.5 \times (((29.5+49.8+2.8+41.5) \times 0.2) + ((31.6+51.5+2.4+43) \times 0.25) + ((31+41.8) \times 0.3) \\ &+ ((31+52+2.6+41.7) \times 0.1) + ((3.5+41.6) \times 0.05) + ((32.4+51.7+3.8) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)) \\ &+ 0.0249375 = 51.3749 \quad (3.4) \end{aligned}$$

$$\begin{aligned} CIDx_3(t=0) &= 0.5 \times (((17.7+17.4+1.4+2.8) \times 0.2) + ((16.1+15.4+3.1+2) \times 0.25) + ((17.6+3) \times 0.3) \\ &+ ((17+15+5.4+2.2) \times 0.1) + ((5.4+2.7) \times 0.05) + ((16.4+13.9) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)) \\ &+ 0.0249375 = 15.4374 \quad (3.5) \end{aligned}$$



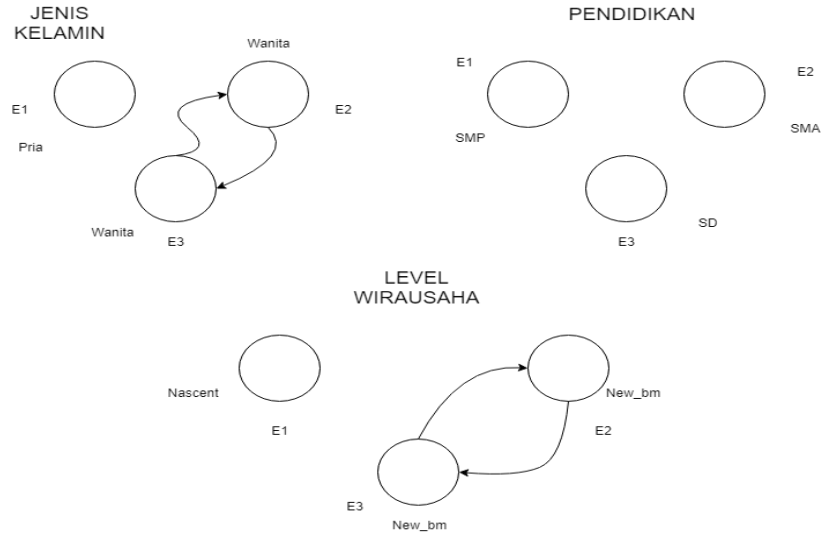
Gambar 3.2: Gambar ketetangaan tiga entrepreneur pada saat $t = 0$

Perhitungan CIDx ($t=1$)

$$\begin{aligned}
 CIDx_1(t=1) &= 0.5 \times (((14.3+4.4+19.9+11) \times 0.2) + ((14.7+17.4+5.4+10.5) \times 0.25) + ((14.3+10.4) \times 0.3) \\
 &+ ((16+19+7.2+10.2) \times 0.1) + ((8.1+11.4) \times 0.05) + ((18.6+18.4+8.9) \times 0.1)) + 0.4 \times (0+0+0) \\
 &+ 0.0249375 = 20.09243 \quad (3.6)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_2(t=1) &= 0.5 \times (((29.5+49.8+2.8+41.5) \times 0.2) + ((31.6+51.5+2.4+43) \times 0.25) + ((31+41.8) \times 0.3) \\
 &+ ((31+52+2.6+41.7) \times 0.1) + ((3.5+41.6) \times 0.05) + ((32.4+51.7+3.8) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)) \\
 &+ 0.0249375 = 51.3749 \quad (3.7)
 \end{aligned}$$

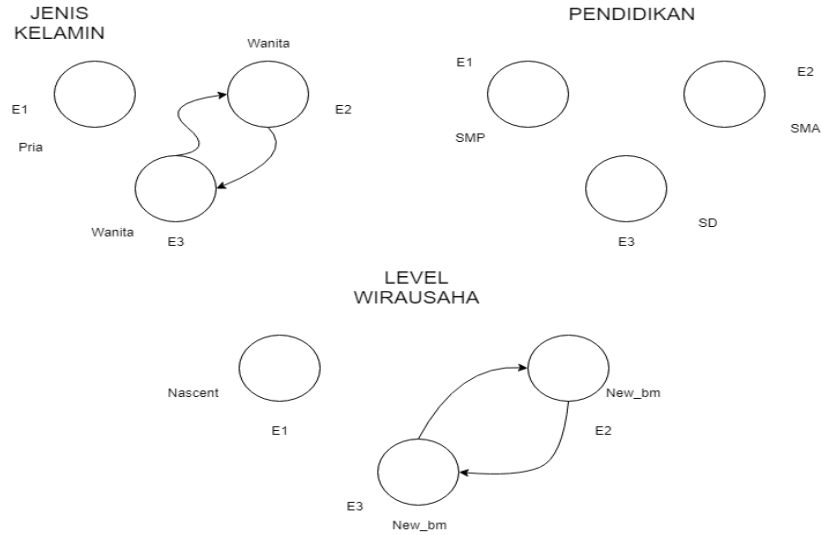
$$\begin{aligned}
 CIDx_3(t=1) &= 0.5 \times (((17.7+17.4+1.4+2.8) \times 0.2) + ((16.1+15.4+3.1+2) \times 0.25) + ((17.6+3) \times 0.3) \\
 &+ ((17+15+5.4+2.2) \times 0.1) + ((5.4+2.7) \times 0.05) + ((16.4+13.9) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)) \\
 &+ 0.0249375 = 15.4374 \quad (3.8)
 \end{aligned}$$

Gambar 3.3: Gambar ketetangaan tiga entrepreneur pada saat $t = 1$ Perhitungan CIDx ($t=2$)

$$\begin{aligned}
 CIDx_1(t=2) &= 0.5 \times (((14.3+4.4+19.9+11) \times 0.2) + ((14.7+17.4+5.4+10.5) \times 0.25) + ((14.3+10.4) \times 0.3) \\
 &+ ((16+19+7.2+10.2) \times 0.1) + ((8.1+11.4) \times 0.05) + ((18.6+18.4+8.9) \times 0.1)) + 0.4 \times (0+0+0) \\
 &+ 0.0249375 = 20.09243 \quad (3.9)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_2(t=2) &= 0.5 \times (((29.5+49.8+2.8+41.5) \times 0.2) + ((31.6+51.5+2.4+43) \times 0.25) + ((31+41.8) \times 0.3) \\
 &+ ((31+52+2.6+41.7) \times 0.1) + ((3.5+41.6) \times 0.05) + ((32.4+51.7+3.8) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)) \\
 &+ 0.0249375 = 51.3749 \quad (3.10)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_3(t=2) &= 0.5 \times (((17.7+17.4+1.4+2.8) \times 0.2) + ((16.1+15.4+3.1+2) \times 0.25) + ((17.6+3) \times 0.3) \\
 &+ ((17+15+5.4+2.2) \times 0.1) + ((5.4+2.7) \times 0.05) + ((16.4+13.9) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{1}{2} \times 0.3)) \\
 &+ 0.0249375 = 15.4374 \quad (3.11)
 \end{aligned}$$



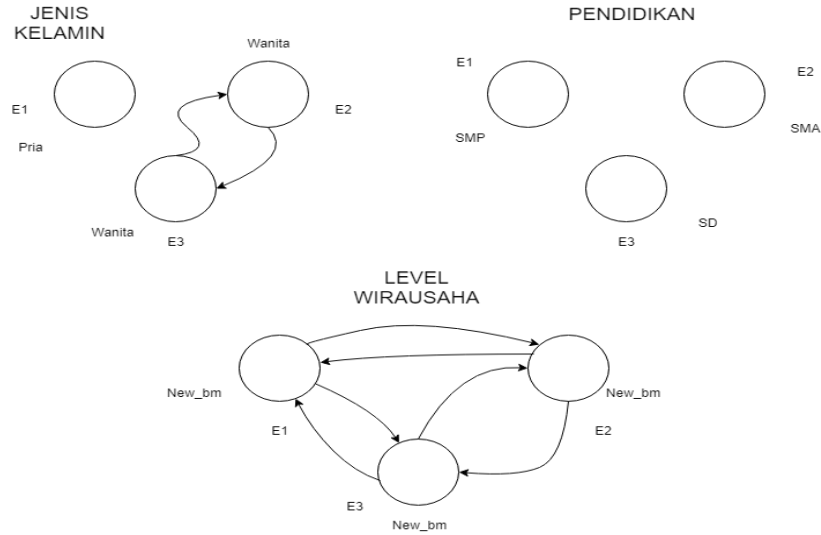
Gambar 3.4: Gambar ketetangaan tiga entrepreneur pada saat $t = 2$

Perhitungan CIDx ($t=3$)

$$\begin{aligned}
 CIDx_1(t=3) &= 0.5 \times (((14.3+4.4+19.9+11) \times 0.2) + ((14.7+17.4+5.4+10.5) \times 0.25) + ((14.3+10.4) \times 0.3) \\
 &+ ((16+19+7.2+10.2) \times 0.1) + ((8.1+11.4) \times 0.05) + ((18.6+18.4+8.9) \times 0.1)) + 0.4 \times (0+0+\frac{2}{2} \times 0.3) \\
 &+ 0.0249375 = 20.2124 \quad (3.12)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_2(t=3) &= 0.5 \times (((29.5+49.8+2.8+41.5) \times 0.2) + ((31.6+51.5+2.4+43) \times 0.25) + ((31+41.8) \times 0.3) \\
 &+ ((31+52+2.6+41.7) \times 0.1) + ((3.5+41.6) \times 0.05) + ((32.4+51.7+3.8) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{2}{2} \times 0.3)) \\
 &+ 0.0249375 = 51.4349 \quad (3.13)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_3(t=3) &= 0.5 \times (((17.7+17.4+1.4+2.8) \times 0.2) + ((16.1+15.4+3.1+2) \times 0.25) + ((17.6+3) \times 0.3) \\
 &+ ((17+15+5.4+2.2) \times 0.1) + ((5.4+2.7) \times 0.05) + ((16.4+13.9) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{2}{2} \times 0.3)) \\
 &+ 0.0249375 = 15.4974 \quad (3.14)
 \end{aligned}$$

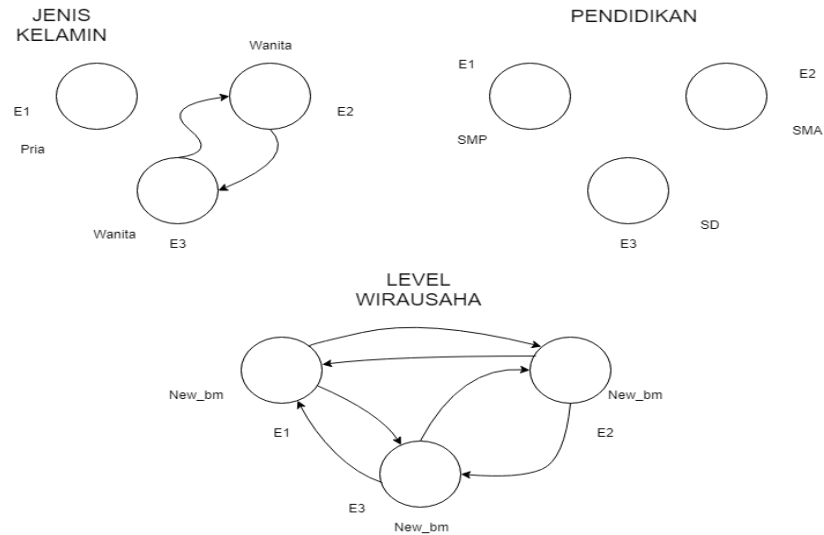
Gambar 3.5: Gambar ketetangaan tiga entrepreneur pada saat $t = 3$

Perhitungan CIDx ($t=4$)

$$\begin{aligned}
 CIDx_1(t = 4) &= 0.5 \times (((14.3 + 4.4 + 19.9 + 11) \times 0.2) + ((14.7 + 17.4 + 5.4 + 10.5) \times 0.25) + ((14.3 + 10.4) \times 0.3) \\
 &+ ((16 + 19 + 7.2 + 10.2) \times 0.1) + ((8.1 + 11.4) \times 0.05) + ((18.6 + 18.4 + 8.9) \times 0.1)) + 0.4 \times (0 + 0 + \frac{2}{2} \times 0.3) \\
 &+ 0.0249375 = 20.48675 \quad (3.15)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_2(t = 4) &= 0.5 \times (((29.5 + 49.8 + 2.8 + 41.5) \times 0.2) + ((31.6 + 51.5 + 2.4 + 43) \times 0.25) + ((31 + 41.8) \times 0.3) \\
 &+ ((31 + 52 + 2.6 + 41.7) \times 0.1) + ((3.5 + 41.6) \times 0.05) + ((32.4 + 51.7 + 3.8) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{2}{2} \times 0.3)) \\
 &+ 0.0249375 = 51.70925 \quad (3.16)
 \end{aligned}$$

$$\begin{aligned}
 CIDx_3(t = 4) &= 0.5 \times (((17.7 + 17.4 + 1.4 + 2.8) \times 0.2) + ((16.1 + 15.4 + 3.1 + 2) \times 0.25) + ((17.6 + 3) \times 0.3) \\
 &+ ((17 + 15 + 5.4 + 2.2) \times 0.1) + ((5.4 + 2.7) \times 0.05) + ((16.4 + 13.9) \times 0.1)) + 0.4 \times ((\frac{1}{2} \times 0.3) + 0 + (\frac{2}{2} \times 0.3)) \\
 &+ 0.0249375 = 15.77175 \quad (3.17)
 \end{aligned}$$



Gambar 3.6: Gambar ketetangaan tiga entrepreneur pada saat $t = 4$

Jadi hasil dari simulasi ini adalah pada bulan pertama wirausaha 1 berada pada level *nascent* dan wirausaha 2 dan 3 berada pada level *new_bm*. Bulan kedua dan ketiga masih sama, bulan keempat mengalami perubahan pada level wirausaha 1 yaitu dari *nascent* berubah menjadi *new_bm* sehingga ketiga wirausaha pada bulan keempat berada pada level wirausaha yang sama, begitu juga pada bulan kelima.

3.2 Deskripsi Perangkat Lunak

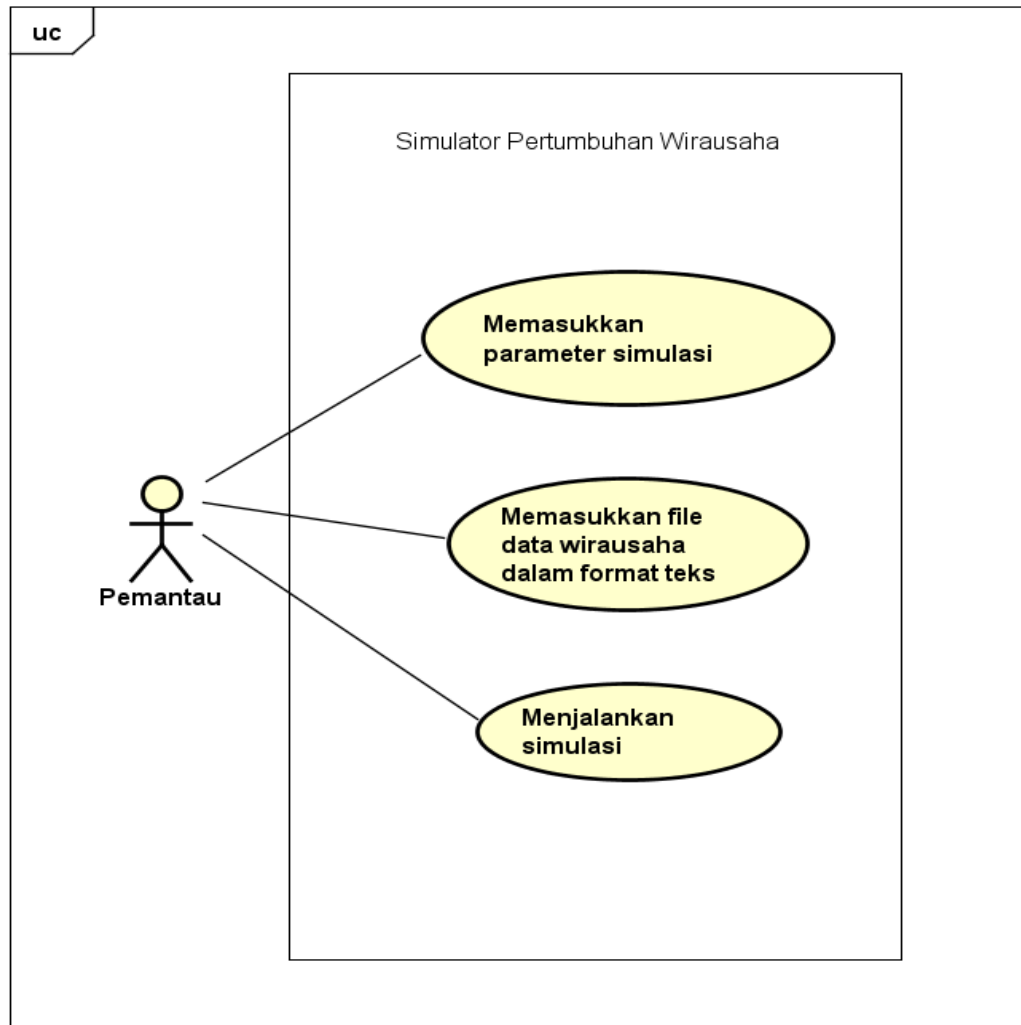
Dalam skripsi ini penulis merancang sebuah simulator dari Entrepreneurial Cellular Automata (ECA) yang sebelumnya telah dikembangkan oleh Cecilia Esti Nugraheni dan Vania Natali [2]. Simulator ini dinamakan Simulator Pertumbuhan Wirausaha Berbasis Cellular Automata.

Perangkat lunak ini dibuat untuk memberi gambaran kepada pemerintah atau lembaga umum mengenai pergerakan wirausaha dalam waktu tertentu. Masukan dari simulator ECA ini yaitu berupa parameter-parameter simulasi yang terdiri dari bobot atribut, relasi antar wirausaha dan nilai a, b, c , *threshold* dan periode. Proses yang dijalankan yaitu pada perhitungan *Continuity Index* yang perhitungannya terbagi menjadi 3 tahap yaitu perhitungan pada faktor internal, perhitungan pada faktor ketetangaan dan perhitungan pada faktor publik. Hasil keluaran dari simulator ini terdiri dari dua keluaran yaitu keluaran yang ditampilkan pada layar yang berupa jumlah wirausaha pada level tertentu yang ditampilkan per bulan, hasil keluaran kedua yaitu berupa perubahan setiap individu wirausaha dalam setiap bulannya pada *file* CSV yang dapat dibuka pada Microsoft Excel.

3.3 Analisis Perangkat Lunak

3.3.1 Diagram Use Case

Pada diagram *use case* hanya terdapat satu aktor yaitu pemerintah sebagai *user*. Diagram *use case* dapat dilihat pada gambar 3.7.



Gambar 3.7: Use Case ECA

Berdasarkan hasil analisis, dibentuk 3 *use case* dengan 1 aktor, yaitu :

1. Memasukkan parameter simulasi

User dapat memasukkan parameter seperti bobot setiap ketetanggaan, relasi ketetanggaan, bobot faktor publik, mengisi nilai *a, b, c* dan *threshold* serta periode.

2. Memasukkan file data wirausaha dalam format text

User dapat memasukkan data wirausaha yang akan disimulasikan berupa *file text*.

3. Menjalankan simulasi

User dapat menjalankan simulasi dan melihat hasil simulasi setiap bulannya.

Skenario Use Case

1. Memasukkan parameter simulasi

- Nama : Memasukkan Parameter Simulasi
- Aktor : *User*
- Deskripsi : Memasukkan bobot untuk setiap atribut dan parameter penting dalam simulasi.

- Kondisi awal : *User* belum mengisi bobot untuk setiap atribut dan parameter dalam simulasi.
- Kondisi akhir : *User* telah mengisi bobot untuk setiap atribut dan parameter dalam simulasi.
- Skenario utama :

Tabel 3.3: Tabel Skenario Memasukkan Parameter Simulasi

No	Aksi	Reaksi Sistem
1	<i>User</i> memasukkan parameter simulasi	Sistem akan menyimpan masukan parameter dari <i>user</i> .

2. Memasukkan *File* Data Wirausaha Dalam Format Text

- Nama : Memasukkan *file* data wirausaha dalam format text.
- Aktor : *User*.
- Deskripsi : Memasukkan *file* data wirausaha yang akan disimulasikan.
- Kondisi awal : *User* memasukkan *file* data wirausaha dalam format text.
- Kondisi akhir : Sistem akan menampilkan isi data pada tabel.
- Skenario utama:

Tabel 3.4: Tabel Skenario Memasukkan *file* data wirausaha dalam format text

No	Aksi	Reaksi Sistem
1	<i>User</i> memilih <i>file</i> dan memasukkan <i>file</i> data wirausaha dalam format text.	Sistem akan menampilkan isi data pada tabel.

3. Menjalankan Simulasi

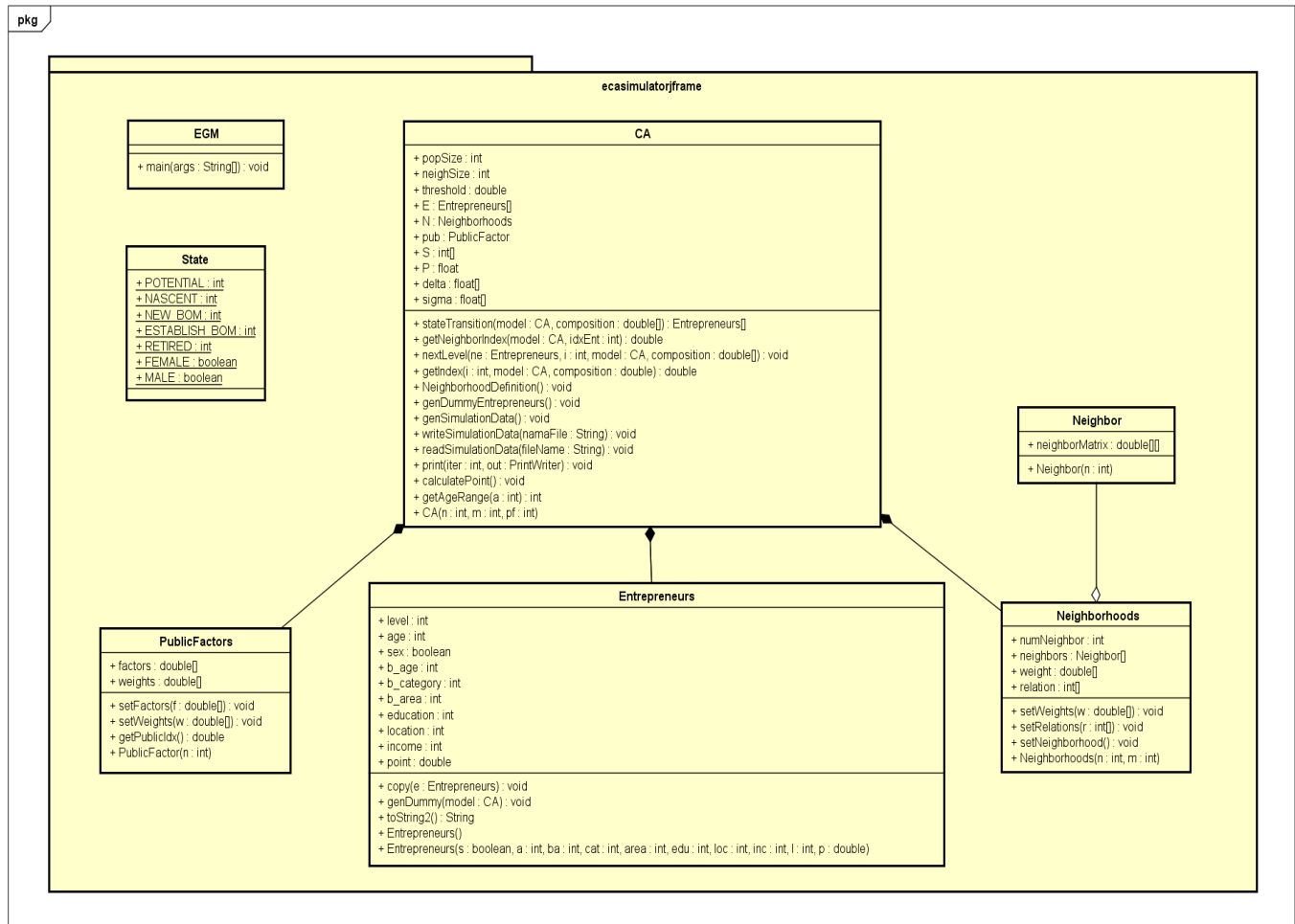
- Nama : Menjalankan Simulasi
- Aktor : *User*
- Deskripsi : Menjalankan simulasi dan melihat hasil simulasi
- Kondisi awal : *User* menjalankan program
- Kondisi akhir : Sistem akan menampilkan hasil di tabel dan sistem juga akan mengeluarkan hasil rincian perubahan individu wirausaha pada *file* CSV.
- Skenario utama:

Tabel 3.5: Tabel Skenario Menjalankan Simulasi

No	Aksi	Reaksi Sistem
1	<i>User</i> menjalankan program	Sistem akan menampilkan hasil di tabel dan sistem juga akan mengeluarkan hasil rincian perubahan individu wirausaha pada <i>file</i> CSV

3.3.2 Diagram Kelas

Pada bagian ini akan diberikan diagram kelas ECA yang merupakan hasil akhir dari penelitian Nugraheni dan Natali.



Gambar 3.8: Diagram Kelas ECA

3.3.3 Kelas EGM

Kelas EGM merupakan kelas untuk menjalankan perhitungan CIDx, CIDx merupakan angka yang mengindikasikan kemungkinan seorang wirausahawan untuk meneruskan usahanya. Perhitungan CIDx ini menggunakan data dari GEM 2013.

3.3.4 Kelas CA

Kelas CA merupakan kelas yang merepresentasikan *cellular automata*. Untuk *method calculatePoint()* pada diagram kelas 3.8 tidak dituliskan parameternya karena parameternya sangat banyak, maka dari itu akan dijelaskan lebih lanjut pada nomor ke 11. Berikut akan dijelaskan beberapa *method* yang ada di kelas CA :

1. `public Entrepreneurs[] stateTransition(CA model, double[] composition)`
Merupakan *method* untuk menentukan perubahan transisi pada seorang wirausaha yang bergantung pada umur dan nilai ambang.
Parameter:
 - `model` merupakan objek dari kelas CA.

- `composition` merupakan nilai `a`, `b` dan `c`.
2. `public double getNeighborIndex(CA model, int idxEnt)`
Merupakan method untuk menghitung nilai dari kondisi ketetangaan setiap wirausaha.
Parameter:
 - `model` merupakan objek dari kelas `CA`.
 - `idxEnt` merupakan indeks dari wirausaha.
 3. `public void nextLevel(Entrepreneurs ne, int i, CA model, double[] composition)`
Merupakan method untuk menentukan perubahan level usaha dari seorang wirausaha.
Parameter:
 - `ne` merupakan objek dari kelas `Entrepreneurs`.
 - `i` merupakan indeks.
 - `model` merupakan objek dari kelas `CA`.
 - `composition` merupakan nilai dari `a`, `b` dan `c`.
 4. `public double getIndex(int i, CA model, double[] composition)`
Merupakan method untuk menghitung `CIDx`.
Parameter:
 - `i` merupakan indeks.
 - `model` merupakan objek dari kelas `CA`.
 - `composition` merupakan nilai dari `a`, `b` dan `c`.
 5. `public void NeighborhoodDefinition()`
Merupakan method untuk mendefinisikan jenis-jenis ketetangaan seperti lebih dari sama dengan, sama dengan dan lebih kecil sama dengan.
 6. `public void genDummyEntrepreneurs()`
Merupakan method untuk membuat data *dummy* wirausaha.
 7. `public void genSimulationData()`
Merupakan method untuk membuat data wirausaha secara *random*.
 8. `public void writeSimulationData(String namaFile)`
Merupakan method untuk menampilkan hasil simulasi ke dalam suatu file.
Parameter:
 - `namaFile` merupakan file tempat hasil simulasi akan ditampilkan.
 9. `public void readSimulationData(String fileName)`
Merupakan method untuk membaca dan memasukkan data file yang akan yang akan disimulasi.
Parameter:
 - `fileName` merupakan file untuk menyimpan hasil simulasi.
 10. `public void print(int iter, PrintWriter out)`
Merupakan method untuk menampilkan jumlah dari masing-masing level wirausaha.
Parameter:
 - `iter` merupakan iterasi per bulan.
 - `out` untuk menge-*print* hasil.

11. `public void calculatePoint(double[] POAm, double[] POAf, double[] POEf, double[] POEm, double[] POLm, double[] POLf, double[] POIm, double[] POIf, double[] PCAf, double[] PCAm, double[] PCEm, double[] PCEf, double[] PCLm, double[] PCLf, double[] PCIm, double[] PCIf, double[] RMAm, double[] RMAf, double[] RMIm, double[] RMIf)`

Merupakan method untuk menghitung kondisi internal dari seorang wirausaha.

Parameter:

- POAm merupakan kumpulan nilai dari Perceived Opportunities berdasarkan umur (pria).
- POAf merupakan kumpulan nilai dari Perceived Opportunities berdasarkan umur (wanita).
- POEm merupakan kumpulan nilai dari Perceived Opportunities berdasarkan pendidikan (pria).
- POEf merupakan kumpulan nilai dari Perceived Opportunities berdasarkan pendidikan (wanita).
- POLm merupakan kumpulan nilai dari Perceived Opportunities berdasarkan lokasi (pria).
- POLf merupakan kumpulan nilai dari Perceived Opportunities berdasarkan lokasi (wanita).
- POIm merupakan kumpulan nilai dari Perceived Opportunities berdasarkan pendapatan (pria).
- POIf merupakan kumpulan nilai dari Perceived Opportunities berdasarkan pendapatan (wanita).
- PCAm merupakan kumpulan nilai dari Perceived Capabilities berdasarkan umur (pria).
- PCAf merupakan kumpulan nilai dari Perceived Capabilities berdasarkan umur (wanita).
- PCEm merupakan kumpulan nilai dari Perceived Capabilities berdasarkan pendidikan (pria).
- PCEf merupakan kumpulan nilai dari Perceived Capabilities berdasarkan pendidikan (wanita).
- PCLm merupakan kumpulan nilai dari Perceived Capabilities berdasarkan lokasi (pria).
- PCLf merupakan kumpulan nilai dari Perceived Capabilities berdasarkan lokasi (wanita).
- PCIm merupakan kumpulan nilai dari Perceived Capabilities berdasarkan pendapatan (pria).
- PCIf merupakan kumpulan nilai dari Perceived Capabilities berdasarkan pendapatan (wanita).
- RMAm merupakan kumpulan nilai dari Role Model berdasarkan umur (pria).
- RMAf merupakan kumpulan nilai dari Role Model berdasarkan umur (wanita).
- RMIm merupakan kumpulan nilai dari Role Model berdasarkan pendapatan (pria).
- RMIf merupakan kumpulan nilai dari Role Model berdasarkan pendapatan (wanita).

12. `public int getAgeRange(int a)`

Merupakan method untuk membedakan rentang usia yang telah ditentukan oleh GEM 2013.[\[4\]](#)

Parameter:

- a merupakan umur wirausaha.

3.3.5 Kelas Entrepreneurs

Kelas Entrepreneur merupakan kelas untuk merepresentasikan individu wirausahawan.

3.3.6 Kelas Neighbor

Kelas Neighbor merupakan kelas untuk merepresentasikan ketetanggaan untuk satu aspek tertentu. Setiap aspeknya didefinisikan sebagai satu neighbor yang berupa adjacency matrix.

3.3.7 Kelas Neighborhood

Kelas Neighborhood merupakan kelas untuk merepresentasikan himpunan ketetanggaan yang tersusun atas sejumlah ketetanggaan.

3.3.8 Kelas Public Factor

Kelas PublicFactor merupakan kelas untuk merepresentasikan faktor publik.

3.3.9 Kelas State

Kelas State merupakan kelas untuk memberi nilai untuk setiap level wirausaha.

BAB 4

PERANCANGAN

Pada bab ini akan dijelaskan perancangan mengenai simulator yang akan dibangun untuk pertumbuhan wirausaha. Perancangan yang dibuat akan meliputi diagram kelas beserta penjelasannya, rancangan antarmuka dari perangkat lunak, serta rancangan *file* input.

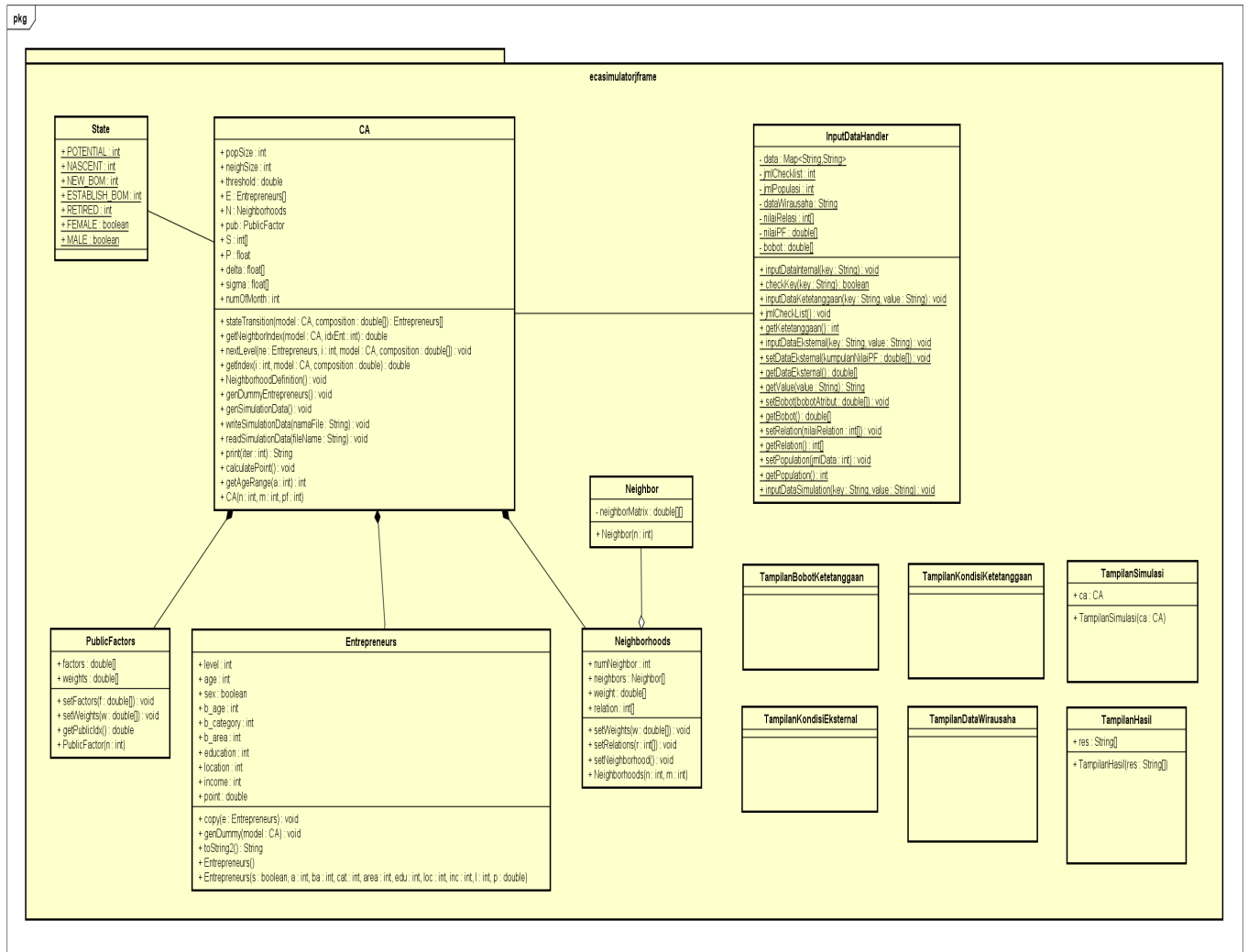
4.1 Diagram Kelas

Dalam membuat simulator diperlukan sebuah GUI atau Interface untuk bisa menggambarkan kinerja suatu sistem. Berdasarkan diagram kelas pada bab analisis 3.8, dibuatlah perubahan diagram kelas rinci yang dibuat oleh penulis untuk memenuhi kebutuhan dalam membangun simulator. Perubahan pada diagram kelas berupa penambahan kelas baru yaitu kelas InputDataHandler dan perubahan pada kelas ECA dan kelas Entrepreneurs. Deskripsi kelas beserta fungsinya akan dijelaskan pada subbab selanjutnya. (Gambar 4.1)

4.1.1 Kelas CA

Pada kelas CA, *method* calculatePoint() di kelas diagram 4.1 tidak dituliskan parameternya karena sangat banyak, maka dari itu akan dijelaskan lebih lanjut pada nomor ke 3. Dilakukan perubahan pada tiga *method* di kelas CA yaitu :

- `public Entrepreneur[] stateTransition(CA model, double[] composition)`
Perubahan yang dilakukan adalah pada saat menambahkan umur wirausaha. Umur wirausaha akan ditambah jika bulannya sudah mencapai 12 bulan atau kelipatan 12 bulan. Dilakukan perubahan agar pada setiap iterasi (bulan), umur wirausaha tidak bertambah secara terus-menerus melainkan ditambah pada saat sudah 1 tahun (12 bulan).
- `public void NeighborhoodDefinition()`
Perubahan yang dilakukan adalah penambahan pada faktor (umur, pendidikan, pendapatan dan jenis kelamin) dan relasi (lebih dari sama dengan).
- `public void calculatePoint(double[] POAm, double[] POAf, double[] POEf, double[] POEm, double[] POLm, double[] POLf, double[] POIm, double[] POIf, double[] PCAf, double[] PCAm, double[] PCEm, double[] PCEf, double[] PCLm, double[] PCLf, double[] PCIm, double[] PCIf, double[] RMAm, double[] RMAf, double[] RMIm, double[] RMIf, double[] FFAf, double[] FFAm, double[] FFEf, double[] FFEIm, double[] FFLf, double[] FFLm, double[] MALf, double[] MALm, double[] MAIf, double[] MAIm, double[] HSSIIf, double[] HSSIIm, double[] HSSLf, double[] HSSLm, double[] HSSAf, double[] HSSAm, double[] HSSEf, double[] HSSEm)`
Perubahan yang dilakukan adalah penambahan pada indikator yang mendukung intensi masyarakat untuk memulai usaha. Indikator-indikator tersebut yaitu Entrepreneurial Intentions (High Status Successful Entrepreneurship, Media Attention) dan Fear of Failure.



Gambar 4.1: Diagram Kelas Simulator ECA

4.1.2 Kelas Tampilan Bobot Ketetangaan

Kelas ini merupakan kelas untuk menampilkan seluruh atribut umum dari seorang wirausaha yang dapat dipilih menggunakan *checkbox*, atribut yang dipilih nantinya akan mempengaruhi ketetangaan antara wirausaha yang satu dengan wirausaha lainnya. Setelah itu, *user* diminta mengisi bobot untuk masing-masing atribut yang sudah dichecklist melalui *textfield*.

4.1.3 Kelas Tampilan Kondisi Ketetangaan

Kelas ini merupakan kelas untuk menampilkan atribut yang sudah dipilih dari kelas Tampilan-KondisiInternal. *User* dapat memilih atribut mana saja yang akan ditetapkan menjadi kondisi ketetangaan untuk satu wirausaha ke wirausaha lainnya. Selain itu, *user* diminta untuk mengisi hubungan ketetangaan khusus untuk 4 atribut yaitu umur, level, pendapatan dan pendidikan jika *user* men-checklist salah satu atau bahkan keempat-empatnya dari atribut tersebut. Untuk atribut jenis kelamin, lokasi usaha dan bidang usaha tidak dapat ditetapkan menjadi 3 jenis karena jenisnya hanya satu yaitu sama dengan. Alasan ketiga atribut tersebut tidak bisa ditetapkan menjadi 3 jenis karena ketiga atribut tersebut tidak bisa diurutkan atau dibandingkan seperti atribut a lebih besar dari atribut b.

4.1.4 Kelas Tampilan Kondisi Eksternal

Kelas ini merupakan kelas untuk menampilkan faktor eksternal yang mempengaruhi pertumbuhan wirausaha. Dalam kasus ini ditetapkan 12 faktor publik yaitu keuangan terkait dengan kewirausahaan, kebijakan pemerintah terkait ekonomi, kebijakan pemerintah terkait pajak, program pemerintah, pendidikan kewirausahaan pada SD dan SMP, pendidikan kewirausahaan pada SMK, professional dan universitas, transfer penelitian dan pengembangan, infrastruktur komersial dan legal, dinamika pasar, keterbukaan pasar, infrastruktur fisik dan akses layanan, serta norma sosial dan budaya. *User* diminta untuk mengisi bobot untuk setiap faktor dan total dari semua bobot harus 100%.

4.1.5 Kelas Data Wirausaha

Kelas ini merupakan kelas untuk membuka *file* data wirausaha yang akan disimulasikan, lalu menampilkannya ke tabel. Isi datanya berupa :

1. Jenis Kelamin
2. Umur
3. Usia Bisnis
4. Kategori Usaha
5. Subkategori
6. Pendidikan
7. Lokasi
8. Pendapatan
9. Level
10. Point

4.1.6 Kelas Tampilan Simulasi

Kelas ini berfungsi untuk mengisi nilai *a*, *b*, *c*, *threshold* dan periode. Nilai *a*, *b*, *c* dan *threshold* bertipe double, sedangkan periode bertipe integer. Periode ini dihitung dalam bulan. Kelas ini juga untuk menghitung *Continuity Index* yang hasil iterasinya akan dikirim ke kelas TampilanHasil dalam bentuk tabel. Selain itu, kelas ini juga akan menampilkan hasil perubahan setiap individu wirausaha dalam setiap bulannya pada *file* CSV.

4.1.7 Kelas Tampilan Hasil

Kelas ini berfungsi untuk menampilkan iterasi (per bulan) banyaknya wirausaha yang berada pada level tertentu dalam bentuk tabel. Untuk hasil keluaran yang dikeluarkan pada *file* CSV dapat dibuka pada Microsoft Excel.

4.1.8 Kelas Input Data Handler

Kelas ini merupakan kelas untuk mengambil dan menyimpan data masukan dari *user* yang nantinya akan dipakai untuk menghitung *Continuity Index*. Berikut penjelasan method-method yang ada di kelas InputDataHandler :

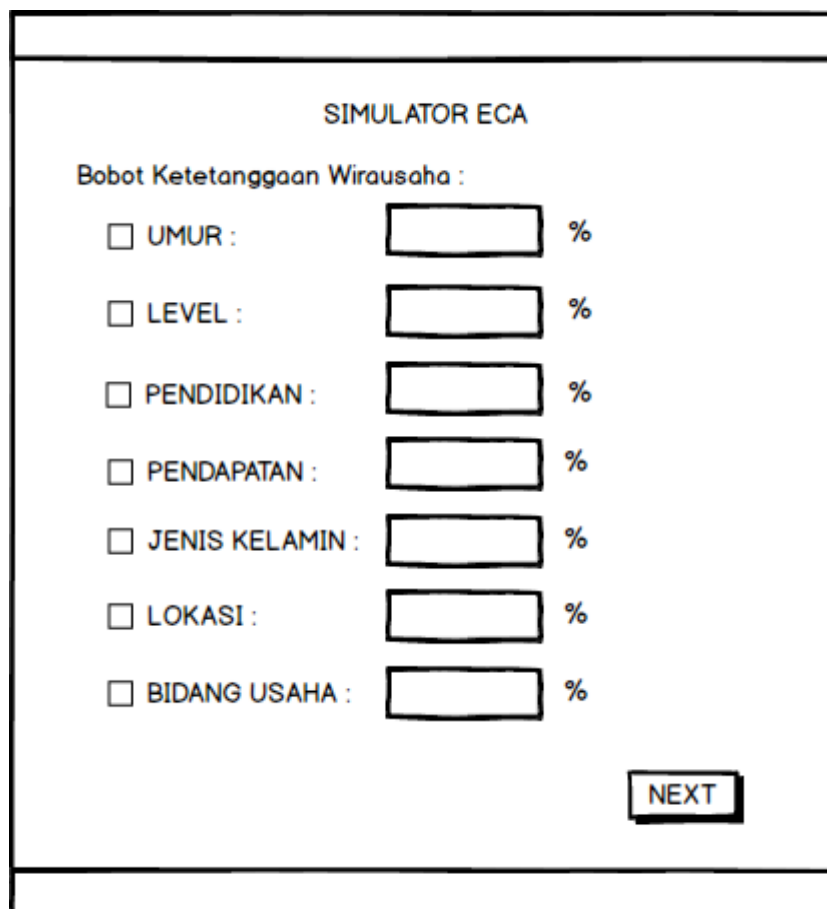
- `public static void inputDataInternal(String key, String value)`
Berfungsi untuk menyimpan masukan pada kelas TampilanBobotKetetapan.
Parameter :

- **key** merupakan kata kunci dari setiap masukan.
- **value** merupakan nilai dari kata kunci.
- **public static boolean checkKey(String key)**
Berfungsi untuk memeriksa isi nilai dari kata kunci. Return *true* jika kata kunci tersebut mempunyai nilai. Return *false* jika kata kunci tersebut tidak mempunyai nilai.
Parameter :
 - **key** merupakan kata kunci dari setiap masukan.
- **public static void inputDataKetetanggaan(String key, String value)**
Berfungsi untuk menyimpan masukan pada kelas TampilanKondisiKetetanggaan.
Parameter :
 - **key** merupakan kata kunci dari setiap masukan.
 - **value** merupakan nilai dari kata kunci.
- **public static void jmlCheckList()**
Berfungsi untuk menambahkan jumlah *checklist* pada kelas TampilanBobotKetetanggaan.
- **public static int getKetetanggaan()**
Berfungsi untuk mengambil nilai ketetanggaan.
- **public static void inputDataEksternal(String key, String value)**
Berfungsi untuk menyimpan masukan dari kelas TampilanKondisiEksternal.
Parameter:
 - **key** merupakan kata kunci dari setiap masukan.
 - **value** merupakan nilai dari kata kunci.
- **public static void setDataEksternal(double[] kumpulanNilaiPF)**
Berfungsi untuk mengubah nilai-nilai dari faktor publik.
Parameter:
 - **kumpulanNilaiPF** merupakan kumpulan nilai faktor publik.
- **public static double[] getDataEksternal()**
Berfungsi untuk mengambil nilai-nilai dari faktor publik.
- **public static String getValue(String key)**
Berfungsi untuk mengambil nilai dari kata kunci.
Parameter:
 - **key** merupakan kata kunci dari setiap masukan.
- **public static void setBobot(double[] bobotAtribut)**
Berfungsi untuk mengubah nilai-nilai bobot dari setiap atribut.
Parameter:
 - **bobotAtribut** merupakan kumpulan bobot dari setiap atribut.
- **public static void getBobot()**
Berfungsi untuk mengambil nilai dari bobot.
- **public static void setRelation(int[] nilaiRelation)**
Berfungsi untuk mengubah nilai-nilai dari setiap relasi.
Parameter:

- nilaiRelation merupakan kumpulan nilai dari setiap relasi.
- `public static int[] getRelation()`
Berfungsi untuk mengambil nilai dari setiap relasi.
- `public static void setPopulation(int jmlData)`
Berfungsi untuk mengubah nilai dari populasi.
Parameter:
 - jmlData merupakan jumlah dari data masukan *user*.
- `public static int getPopulation()`
Berfungsi untuk mengembalikan nilai dari populasi.
- `public static void inputDataSimulasi(String key, String value)`
Berfungsi untuk menyimpan masukan dari kelas TampilanSimulasi.
Parameter :
 - key merupakan kata kunci dari setiap masukan.
 - value merupakan nilai dari kata kunci.

4.2 Rancangan Antarmuka

4.2.1 Tampilan Bobot Ketetangaan



SIMULATOR ECA

Bobot Ketetangaan Wirausaha :

<input type="checkbox"/> UMUR :	<input type="text"/>	%
<input type="checkbox"/> LEVEL :	<input type="text"/>	%
<input type="checkbox"/> PENDIDIKAN :	<input type="text"/>	%
<input type="checkbox"/> PENDAPATAN :	<input type="text"/>	%
<input type="checkbox"/> JENIS KELAMIN :	<input type="text"/>	%
<input type="checkbox"/> LOKASI :	<input type="text"/>	%
<input type="checkbox"/> BIDANG USAHA :	<input type="text"/>	%

NEXT

Gambar 4.2: Gambar Tampilan Bobot Ketetangaan

Dapat dilihat pada gambar 4.2, pada kondisi awal terdapat 7 atribut umum (umur, level wirausaha, pendidikan, pendapatan, jenis kelamin, lokasi dan bidang usaha) dari seorang wirausahawan yang dapat dipilih oleh *user* melalui *checkbox*. *User* dapat memilih lebih dari 1 atribut. Jika *user* tidak mengisi *checkbox* terlebih dahulu, *user* tidak akan bisa mengisi bobot atribut. Atribut yang dipilih melalui *checkbox*, akan menjadi atribut ketetangaan dari wirausahawan satu dengan wirausahawan lainnya. Setelah *user* memilih atribut wirausaha, *user* harus mengisi bobot dari masing-masing atribut melalui *text field*. Total dari bobot atribut yang dipilih jumlahnya harus 100%. Jika *user* tidak mengisi seluruh *checkbox*, *user* tidak akan bisa melanjutkan ke proses selanjutnya. Begitu juga jika *user* tidak mengisi bobot berdasarkan atribut yang sudah dipilih, *user* tidak dapat melanjutkan ke proses selanjutnya.

4.2.2 Tampilan Kondisi Ketetangaan

SIMULATOR ECA

Parameter Setting Kondisi Ketetangaan :
Berdasarkan relasi :

UMUR	<input type="radio"/> <=	<input type="radio"/> =	<input type="radio"/> >=
LEVEL	<input type="radio"/> <=	<input type="radio"/> =	<input type="radio"/> >=
PENDIDIKAN	<input type="radio"/> <=	<input type="radio"/> =	<input type="radio"/> >=
PENDAPATAN	<input type="radio"/> <=	<input type="radio"/> =	<input type="radio"/> >=
JENIS KELAMIN			
LOKASI			
BIDANG USAHA			

Gambar 4.3: Gambar Tampilan Kondisi Ketetangaan

Dapat dilihat pada gambar 4.3, terdapat 7 atribut tetangga yang telah dipilih oleh *user* pada kelas TampilanBobotKetetangaan. Atribut yang ditampilkan pada kondisi ketetangaan bergantung pada pemilihan atribut pada tampilan sebelumnya 4.2. Contohnya jika *user* memilih 3 atribut (level wirausaha, jenis kelamin dan lokasi) pada tampilan sebelumnya, atribut yang akan ditampilkan pada kondisi ketetangaan hanya 3 atribut (level wirausaha, jenis kelamin dan lokasi). Pada tampilan ini *user* diminta untuk mengisi relasi ketetangaan khususnya pada atribut umur, level, pendapatan dan pendidikan. 3 atribut lainnya tidak terdapat relasi ketetangaan, hal ini dikarenakan ketiga atribut tersebut tidak bisa dibanding-bandingkan. Contohnya seperti lokasi, wirausaha A membangun usahanya di kota Jakarta, sedangkan wirausaha B membangun usahanya di kota Bandung. Tentu saja hal ini tidak dapat ditetapkan sebagai kota Jakarta lebih dari kota Bandung atau kota Bandung

kurang dari kota Jakarta.

4.2.3 Tampilan Kondisi Eksternal

The screenshot displays the 'SIMULATOR ECA' interface. The title 'SIMULATOR ECA' is centered at the top. Below it, the section 'Parameter Setting Kondisi Eksternal :' is followed by a list of 12 external factors. Each factor has a corresponding input box for its weight, followed by a '%' symbol. The factors are: Keuangan terkait Kewirausahaan, Kebijakan Pemerintah terkait Ekonomi, Kebijakan Pemerintah terkait Pajak, Program Pemerintah, Pendidikan Kewirausahaan pada SD dan SMP, Pendidikan Kewirausahaan pada SMK, Profesional dan Universitas, Transfer Penelitian dan Pengembangan, Infrastruktur Komersial dan Legal, Dinamika Pasar, Keterbukaan Pasar, Infrastruktur Fisik dan Akses Layanan, and Norma, Sosial dan Budaya. At the bottom of the list, there is a 'Total:' label followed by a larger input box and a '%' symbol. Below the 'Total' field are two buttons: 'BACK' on the left and 'NEXT' on the right.

Parameter Setting Kondisi Eksternal :	Input Box	%
Keuangan terkait Kewirausahaan :	<input type="text"/>	%
Kebijakan Pemerintah terkait Ekonomi :	<input type="text"/>	%
Kebijakan Pemerintah terkait Pajak :	<input type="text"/>	%
Program Pemerintah :	<input type="text"/>	%
Pendidikan Kewirausahaan pada SD dan SMP :	<input type="text"/>	%
Pendidikan Kewirausahaan pada SMK, Profesional dan Universitas :	<input type="text"/>	%
Transfer Penelitian dan Pengembangan :	<input type="text"/>	%
Infrastruktur Komersial dan Legal :	<input type="text"/>	%
Dinamika Pasar :	<input type="text"/>	%
Keterbukaan Pasar :	<input type="text"/>	%
Infrastruktur Fisik dan Akses Layanan :	<input type="text"/>	%
Norma, Sosial dan Budaya :	<input type="text"/>	%
Total :	<input type="text"/>	%

BACK NEXT

Gambar 4.4: Gambar Tampilan Kondisi Eksternal

Pada tampilan kondisi eksternal terdapat 12 faktor publik yang mempengaruhi pertumbuhan wirausaha di Indonesia. Keduabelas faktor publik ini didapatkan dari data GEM 2013. Untuk keduabelas faktor ini, *user* harus mengisi bobot setiap faktor publik yang total bobotnya harus 100%. Untuk memeriksa jumlah masukan bobot, *user* bisa melihatnya di total.

4.2.4 Tampilan Data Wirausaha

SIMULATOR ECA

Data Simulasi : OPEN FILE

Jenis Kelamin	Umur	Usia Bisnis	Kategori	Sub Kategori	Pendidikan	Lokasi	Pendapatan	Level	Point

BACKNEXT

Gambar 4.5: Gambar Tampilan Data Wirausaha

Pada tampilan ini, *user* akan memasukkan *file* masukan data wirausaha dalam format teks, rancangan *file input* akan dibahas pada subbab 4.3. Setelah *file* dipilih, data wirausaha akan ditampilkan pada tabel.

4.2.5 Tampilan Simulasi

The image shows a software interface titled "SIMULATOR ECA". Below the title, there is a label "Simulasi :". Underneath, there are five input fields with labels: "a:", "b:", "c:", "Threshold:", and "Periode:". The "Periode:" label is followed by the word "bulan". Below these input fields is a button labeled "SIMULATE".

Gambar 4.6: Gambar Tampilan Simulasi

Pada tampilan simulasi, *user* diminta untuk mengisi nilai *a*, *b*, *c*, *threshold* dan periode. Nilai-nilai tersebut digunakan untuk menghitung CIDx wirausahawan. Total dari nilai *a*, *b* dan *c* harus 1. Periode merupakan berapa lama iterasi tersebut akan berjalan (dalam bulan). Sedangkan *button* "SIMULATE" berfungsi untuk menjalankan simulasi yang hasilnya akan ditampilkan dalam bentuk tabel.

4.2.6 Tampilan Hasil

Pada tampilan hasil, akan ditampilkan hasil simulasi berupa tabel, yang masing-masing kolomnya berisi iterasi (bulan), jumlah wirausaha pada level *potential*, jumlah wirausaha pada level *nascent*, jumlah wirausaha pada level *new_bm*, jumlah wirausaha pada level *est_bm*, jumlah wirausaha pada level *retired*. Untuk kolom pertama, di setiap barisnya akan berisi lamanya iterasi (dalam bulan). Contoh periode 5 bulan maka setiap barisnya berisi bulan ke-0, bulan ke-1, bulan ke-2 sampai bulan ke-4. Iterasi dimulai dari bulan ke-0 yang artinya bulan ke-1, bulan ke-1 yang artinya bulan ke-2, dst.

- 1 untuk makanan berat
 - 2 untuk makanan cepat saji
 - Kategori minuman :
 - 0 untuk minuman sehat
 - 1 untuk minuman bersoda
 - 2 untuk minuman *sachet*
 - Kategori tas :
 - 0 untuk tas pria
 - 1 untuk tas anak-anak
 - 2 untuk tas wanita
6. Pendidikan bertipe bilangan bulat, masing-masing angka mendeskripsikan tingkat pendidikan yang berbeda, yaitu :
- 0 untuk tingkat pendidikan rendah
 - 1 untuk sekolah dasar
 - 2 untuk sekolah menengah pertama
 - 3 untuk sekolah menengah ke atas
 - 4 untuk sarjana (S1)
 - 5 untuk diploma (S2)
 - 6 untuk profesor (S3)
7. Lokasi, bertipe bilangan bulat yang masing-masing angkanya mendeskripsikan lokasi yang berbeda, yaitu :
- 0 untuk Banda Aceh
 - 1 untuk Medan
 - 2 untuk Padang
 - 3 untuk Pekanbaru
 - 4 untuk Palembang
 - 5 untuk Bandar Lampung
 - 6 untuk Serang
 - 7 untuk Jakarta
 - 8 untuk Bandung
 - 9 untuk Semarang dan Surakarta
 - 10 untuk Surabaya
 - 11 untuk Denpasar
 - 12 untuk Mataram
 - 13 untuk Kupang
 - 14 untuk Pontianak
 - 15 untuk Makassar
8. Pendapatan bertipe bilangan bulat, masing-masing angka mendeskripsikan tingkat pendapatan yang berbeda yaitu :
- 0 untuk pendapatan dibawah 3 juta rupiah

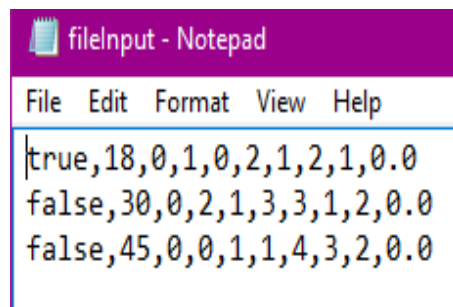
- 1 untuk pendapatan 3 juta rupiah sampai 5 juta rupiah
- 2 untuk pendapatan 5 juta rupiah sampai 7 juta rupiah
- 3 untuk pendapatan 7 juta rupiah sampai 9 juta rupiah
- 4 untuk pendapatan 9 juta rupiah sampai 11 juta rupiah
- 5 untuk pendapatan 11 juta rupiah sampai 13 juta rupiah
- 6 untuk pendapatan 13 juta rupiah sampai 15 juta rupiah
- 7 untuk pendapatan diatas 15 juta rupiah

9. Level, bertipe bilangan bulat, masing-masing angka mendeskripsikan level yang berbeda yaitu :

- 0 untuk level potential
- 1 untuk level nascent
- 2 untuk level new business manager
- 3 untuk level established
- 4 untuk level retired

10. Point merupakan nilai dari kondisi internal individu wirausaha. Point mempunyai tipe data double.

Berikut contoh untuk *file input* :



Gambar 4.8: Contoh Format File Data Wirausaha

Berdasarkan contoh di atas 4.8, terdapat 3 data wirausahawan. Baris pertama berisi pria, berumur 18 tahun, usia bisnisnya 0 bulan, kategori atau bidang usahanya minuman, sub kategori minuman bersoda, pendidikan SMP, usahanya berlokasi di Medan, pendapatan 5 sampai 7 juta rupiah, level usahanya *nascent* dan nilai dari kondisi internalnya masih 0.

BAB 5

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini terdapat tiga bagian, yaitu implementasi perangkat lunak, pengujian perangkat lunak dan contoh simulasi. Bagian implementasi akan menjelaskan tentang lingkungan pengembangan perangkat lunak dan hasil implementasi. Bagian pengujian akan berisi hasil pengujian fungsional terhadap perangkat lunak yang telah dibangun dan bagian terakhir yaitu contoh simulasi.

5.1 Implementasi

5.1.1 Implementasi

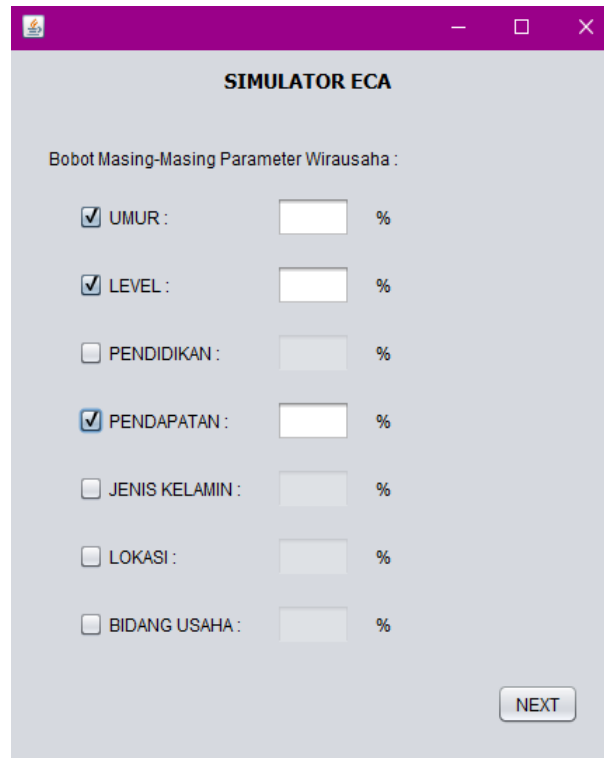
Implementasi dilakukan dengan menggunakan laptop dengan spesifikasi sebagai berikut :

1. *Processor* : Intel(R) Core(TM) i5-4200U CPU @ 1.60GHz 2.30GHz
2. RAM : 4.00 GB
3. Sistem Operasi : Windows 10 Pro 64-bit
4. Versi Netbeans : 8.0.2
5. Microsoft Excel : 2013

5.1.2 Hasil Implementasi

1. Tampilan Bobot Ketetangaan

Seperti yang telah dijelaskan pada bab [4](#), tampilan ini berfungsi untuk mengisi atribut dari masing-masing wirausaha. *User* dapat memilih atribut mana yang akan dijadikan sebagai ketetangaan dari masing-masing wirausaha dengan cara men-*checkbox* atribut yang diinginkan. (Gambar [5.1](#))



SIMULATOR ECA

Bobot Masing-Masing Parameter Wirausaha :

☒ UMUR : %

☒ LEVEL : %

☐ PENDIDIKAN : %

☒ PENDAPATAN : %

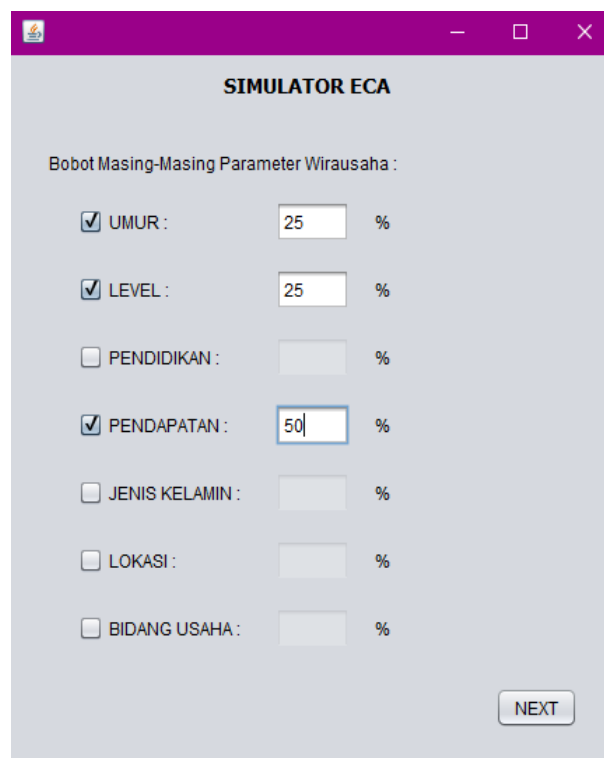
☐ JENIS KELAMIN : %

☐ LOKASI : %

☐ BIDANG USAHA : %

NEXT

Gambar 5.1: Gambar Tampilan Bobot Ketetangaan pada saat men-*checklist checkbox*



SIMULATOR ECA

Bobot Masing-Masing Parameter Wirausaha :

☒ UMUR : %

☒ LEVEL : %

☐ PENDIDIKAN : %

☒ PENDAPATAN : %

☐ JENIS KELAMIN : %

☐ LOKASI : %

☐ BIDANG USAHA : %

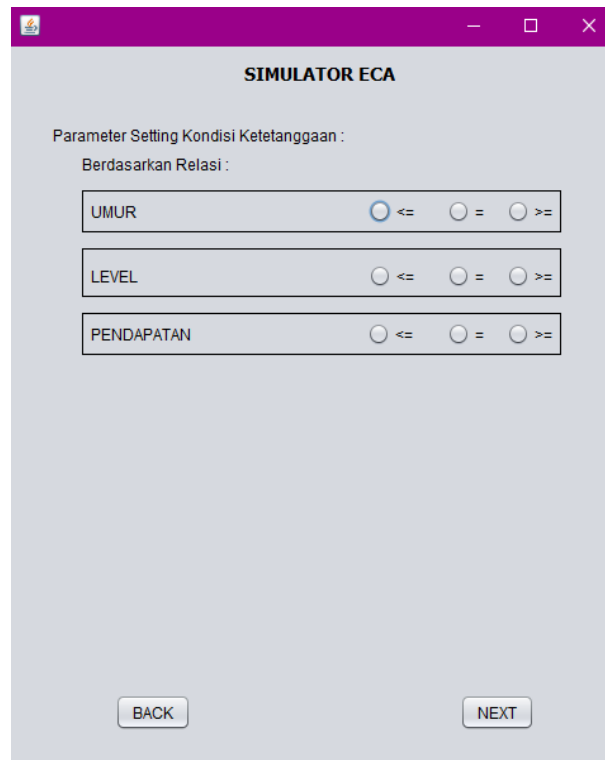
NEXT

Gambar 5.2: Gambar Tampilan Bobot Ketetangaan pada saat mengisi bobot masing-masing atribut

Pada implementasi ini, sebagai contoh *user* memilih 3 atribut yaitu umur, level dan pendapatan. Masing-masing bobotnya yaitu 25%, 25% dan 50%.

2. Tampilan Kondisi Ketetangaan

Pada tampilan ini, *user* diminta untuk mengisi relasi ketetangaan pada atribut yang telah dipilih sebelumnya. (Gambar 5.3)



SIMULATOR ECA

Parameter Setting Kondisi Ketetangaan :

Berdasarkan Relasi :

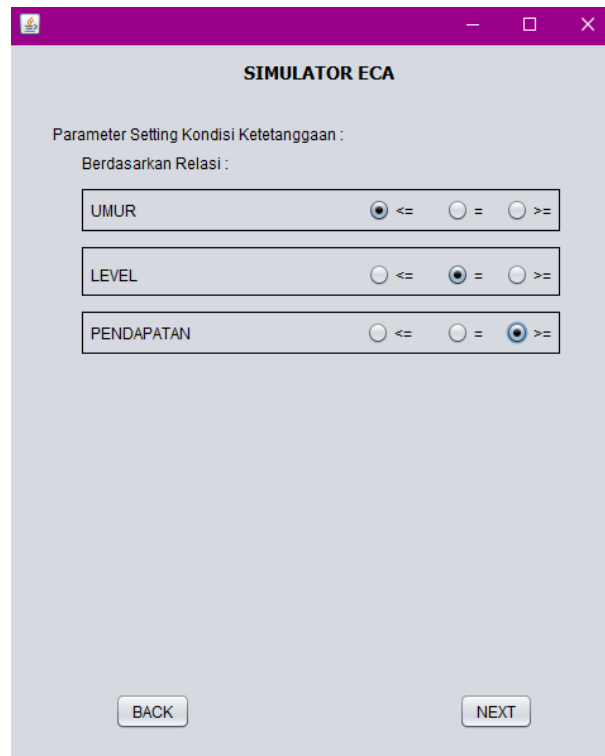
UMUR ☒ <= ☐ = ☐ >=

LEVEL ☐ <= ☐ = ☐ >=

PENDAPATAN ☐ <= ☐ = ☐ >=

BACK NEXT

Gambar 5.3: Gambar Tampilan Kondisi Ketetangaan untuk atribut yang telah dipilih sebelumnya.



SIMULATOR ECA

Parameter Setting Kondisi Ketetangaan :

Berdasarkan Relasi :

UMUR ☒ <= ☐ = ☐ >=

LEVEL ☐ <= ☒ = ☐ >=

PENDAPATAN ☐ <= ☐ = ☒ >=

BACK NEXT

Gambar 5.4: Gambar Tampilan Kondisi Ketetangaan pada saat mengisi relasi ketetangaan

Pada implementasi ini, *user* mengisi masing-masing relasi ketetanggaan untuk umur yaitu kurang dari sama dengan, level wirausaha relasinya sama dengan dan pendapatan relasinya lebih dari sama dengan.

3. Tampilan Kondisi Eksternal

Pada tampilan ini, *user* akan mengisi bobot masing-masing faktor publik. Jumlah dari seluruh bobot harus 100%. *User* mengisi masing-masing bobot dengan 5%, 5%, 10%, 10%, 10%, 5%, 10%, 5%, 10%, 10%, 10% dan 10%. Total dari bobot yang dimasukan *user* sudah 100%, hal ini bisa dilihat di *textfield* total. Bobot-bobot ini nantinya akan dikalikan dengan nilai dari masing-masing faktor publik yang dapat dilihat pada tabel 2.4. (Gambar 5.5)

SIMULATOR ECA

Parameter Setting Kondisi Eksternal :

Faktor Publik :

Keuangan terkait Kewirausahaan :	5	%
Kebijakan Pemerintah terkait Ekonomi :	5	%
Kebijakan Pemerintah terkait Pajak :	10	%
Program Pemerintah :	10	%
Pendidikan Kewirausahaan pada SD dan SMP :	10	%
Pendidikan Kewirausahaan pada SMK, Profesional dan Universitas :	5	%
Transfer Penelitian dan Pengembangan :	10	%
Infrastruktur Komersial dan Legal :	5	%
Dinamika Pasar :	10	%
Keterbukaan Pasar :	10	%
Infrastruktur Fisik dan Akses Layanan :	10	%
Norma, Sosial dan Budaya :	10	%

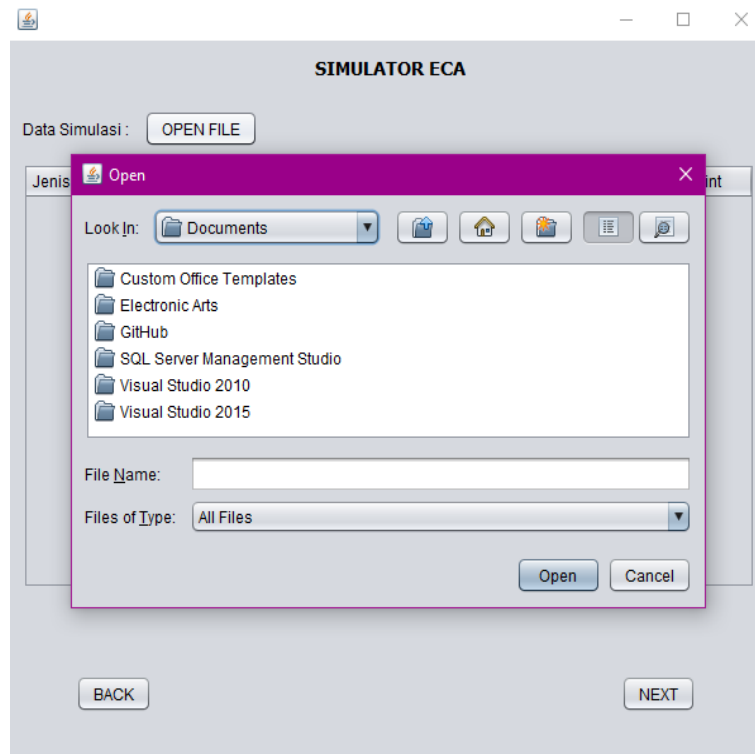
Total : 100.0 %

BACK NEXT

Gambar 5.5: Gambar Tampilan Kondisi Eksternal pada saat mengisi bobot faktor publik

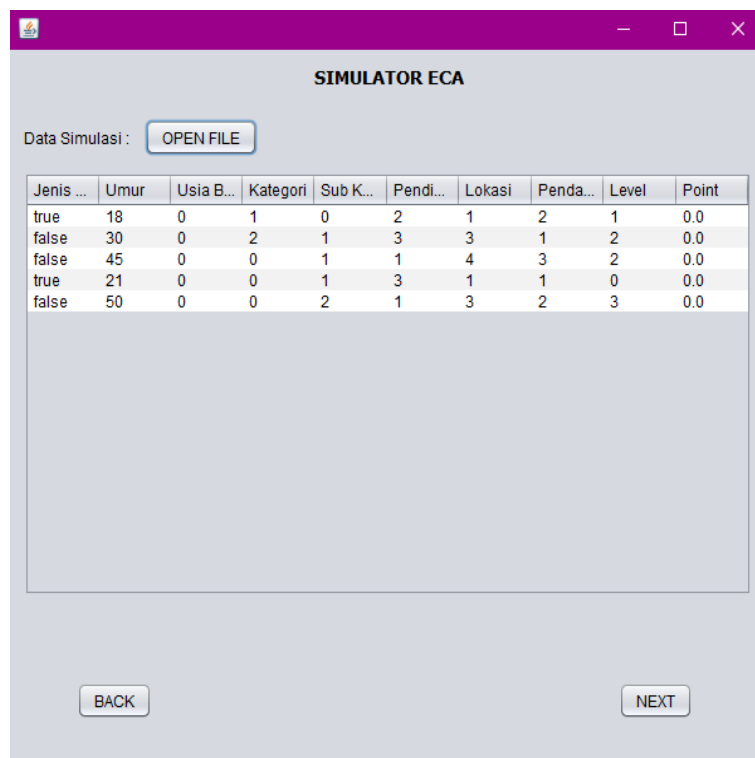
4. Tampilan Data Wirausaha

Pada tampilan data wirausaha *user* dapat meng-klik *button* "OPEN FILE" yang fungsinya untuk membuka *file* data wirausaha yang akan disimulasikan. Data wirausaha berisi jenis kelamin, umur, usia bisnis, kategori usaha, subkategori usaha, pendidikan, lokasi, pendapatan, level dan point 4.8. Point merupakan hasil perhitungan masing-masing wirausaha pada kondisi internal. (Gambar 5.6)



Gambar 5.6: Gambar Tampilan Data Wirausaha pada saat membuka *button* "OPEN FILE"

Berikut merupakan tampilan data wirausaha (data *dummy*) yang telah dipilih oleh *user*. (Gambar 5.7)



Gambar 5.7: Gambar Tampilan Data Wirausaha saat menampilkan isi dari file

5. Tampilan Simulasi

Pada tampilan ini *user* diminta untuk mengisi bobot dari a,b,c,threshold dan periode yang digunakan untuk menjalankan simulasi. Total nilai dari a,b dan c harus 1. Setelah mengisi masing-masing nilai, *user* dapat melakukan simulasi dengan cara meng-klik *button* "SIMULATE". (Gambar 5.8)

Gambar 5.8: Gambar Tampilan Simulasi pada saat mengisi bobot a,b,c,threshold dan periode

User mengisi nilai a,b,c masing-masingnya yaitu 0.3, 0.4 dan 0.3, sedangkan *threshold* atau nilai ambangnya 20 dan simulasi akan dilakukan selama 5 bulan atau sama dengan 5 iterasi. Setelah itu, *user* memilih tombol "SIMULATE" untuk menjalankan simulasi. hasil perubahan setiap individu wirausaha dalam setiap bulannya akan dikeluarkan pada *file* CSV yang dapat dibuka di Microsoft Excel. Berikut hasil keluaran pada *file* CSV. (Gambar 5.9 dan Gambar 5.10)

	A	B	C	D	E	F	G	H	I
1	Bulan ke-0								
2	true	18	0	1	0	2	1	2	0
3	false	30	1	2	1	3	3	1	2
4	false	45	0	0	1	1	4	3	0
5	true	21	1	0	1	3	1	1	1
6	false	50	0	0	2	1	3	2	0
7	Bulan ke-1								
8	true	18	0	1	0	2	1	2	0
9	false	30	2	2	1	3	3	1	2
10	false	45	0	0	1	1	4	3	0
11	true	21	2	0	1	3	1	1	1
12	false	50	0	0	2	1	3	2	0
13	Bulan ke-2								
14	true	18	0	1	0	2	1	2	0
15	false	30	3	2	1	3	3	1	2
16	false	45	0	0	1	1	4	3	0
17	true	21	3	0	1	3	1	1	1
18	false	50	0	0	2	1	3	2	0
19	Bulan ke-3								
20	true	18	0	1	0	2	1	2	0
21	false	30	4	2	1	3	3	1	2
22	false	45	0	0	1	1	4	3	0
23	true	21	4	0	1	3	1	1	2

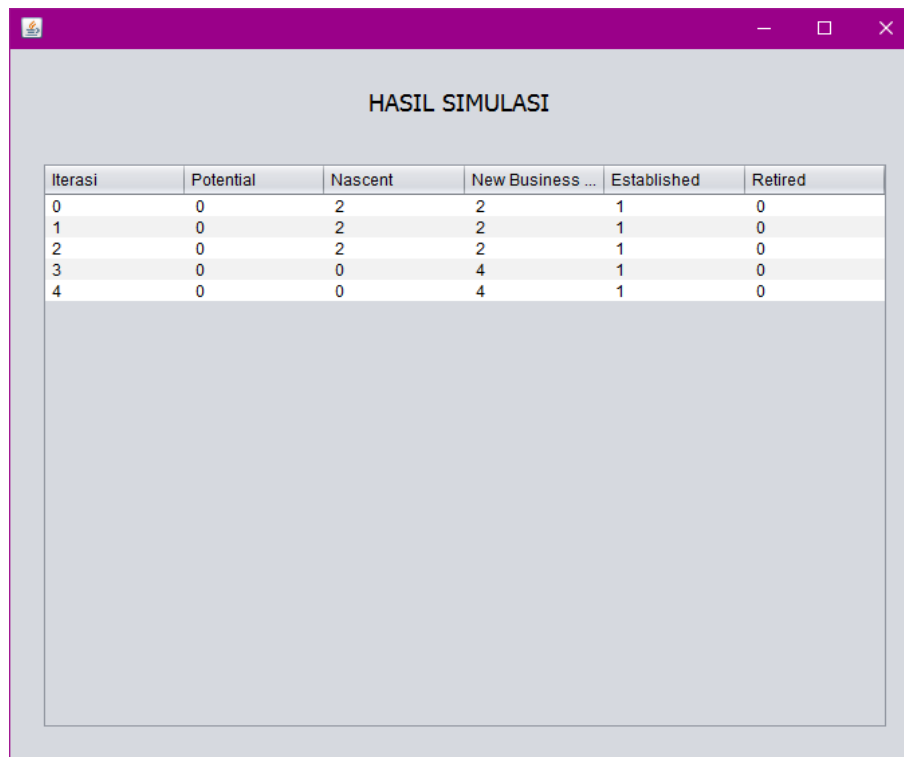
Gambar 5.9: Hasil keluaran perubahan individu wirausaha pada *file* CSV

24	false	50	0	0	2	1	3	2	0
25	Bulan ke-4								
26	true	18	0	1	0	2	1	2	0
27	false	30	5	2	1	3	3	1	2
28	false	45	0	0	1	1	4	3	0
29	true	21	5	0	1	3	1	1	2
30	false	50	0	0	2	1	3	2	0

Gambar 5.10: Lanjutan hasil keluaran perubahan individu wirausaha pada *file CSV*

6. Tampilan Hasil

Pada tampilan ini akan ditampilkan hasil dari simulasi berupa tabel yang isi setiap kolomnya adalah iterasi (bulan), jumlah wirausaha yang berada pada level *potential*, jumlah wirausaha yang berada pada level *nascent*, jumlah wirausaha yang berada pada level *new_bm*, jumlah wirausaha yang berada pada level *est_bm* dan jumlah wirausaha yang berada pada level *retired*. (Gambar 5.11)



Iterasi	Potential	Nascent	New Business ...	Established	Retired
0	0	2	2	1	0
1	0	2	2	1	0
2	0	2	2	1	0
3	0	0	4	1	0
4	0	0	4	1	0

Gambar 5.11: Gambar Tampilan Hasil

5.2 Pengujian

5.2.1 Pengujian Fungsional

Pengujian fungsional dilakukan untuk mengetahui kesesuaian reaksi perangkat lunak dengan reaksi yang diharapkan berdasarkan aksi *user* terhadap perangkat lunak. Pengujian ini ditujukan pada 1 pengguna yaitu *user*.

Terdapat 8 tes kasus yang diujikan. Detail dan hasilnya dapat dilihat pada tabel 5.1

Tabel 5.1: Tabel Pengujian Fungsional *User*

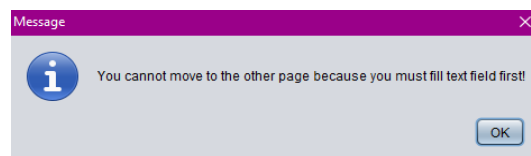
No	Aksi Pengguna	Reaksi yang diharapkan	Reaksi Perangkat Lunak
1	<i>User</i> menjalankan simulator / aplikasi	Tampilan Bobot Ketetanggaan akan ditampilkan	Sesuai
2	<i>User</i> melanjutkan pengisian dengan memilih <i>button</i> "NEXT"	Tampilan Kondisi Ketetanggaan akan ditampilkan	Sesuai
3	<i>User</i> melanjutkan pengisian dengan memilih <i>button</i> "NEXT"	Tampilan Kondisi Eksternal akan ditampilkan	Sesuai
4	<i>User</i> melanjutkan pengisian dengan memilih <i>button</i> "NEXT"	Tampilan Data Wirausaha akan ditampilkan	Sesuai
5	<i>User</i> memasukkan data wirausaha dengan memilih <i>button</i> "OPEN FILE"	Muncul <i>pop up windows</i> yang menyediakan beberapa <i>file</i> , salah satu <i>file</i> akan dipilih oleh <i>user</i>	Sesuai
6	Setelah <i>User</i> memilih <i>file</i> dan memilih <i>button</i> "OPEN"	Data wirausaha akan ditampilkan di tabel	Sesuai
7	<i>User</i> melanjutkan proses simulasi dengan memilih <i>button</i> "NEXT"	Tampilan Simulasi akan ditampilkan	Sesuai
8	<i>User</i> selesai mengisi <i>text field</i> dan memilih <i>button</i> "SIMULATE"	Hasil simulasi akan ditampilkan di tabel dan pada <i>file</i> CSV	Sesuai

5.2.2 Pengujian Pembacaan Parameter

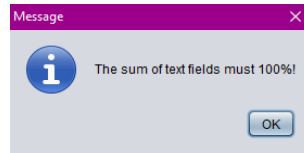
Pengujian ini dilakukan agar tidak terjadi kesalahan *input* dari *user* yang mengakibatkan hasil simulasi tidak sesuai dengan yang diharapkan.

1. Pengisian *Text Field* pada saat mengisi bobot ketetanggaan

- Jika *user* sudah mengisi *check box* tetapi tidak mengisi *text field*, akan terdapat pesan kesalahan "You cannot move to the other page because you must fill text field first!". (Gambar 5.16)

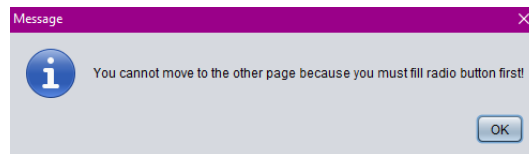
Gambar 5.12: Tampilan Pesan Error pada saat *text field* tidak terisi

- Jika *user* sudah mengisi *text field* tetapi totalnya tidak 100%, akan terdapat pesan kesalahan "The sum of text fields must 100%!". (Gambar 5.17)



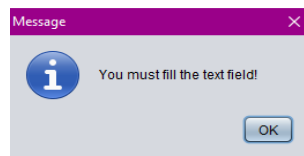
Gambar 5.13: Tampilan Pesan Error pada saat isi dari *text field* tidak berjumlah 100%

2. Pengisian *Radio Button* pada saat mengisi relasi ketetanggaan
Jika *user* tidak mengisi radio button, akan ada pesan kesalahan yaitu "You cannot move to the other page because you must fill radio button first!". (Gambar 5.14)



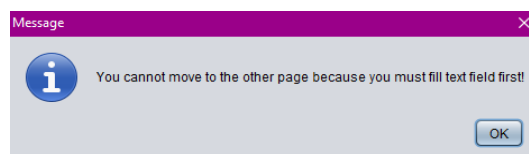
Gambar 5.14: Tampilan Pesan Error pada saat *radio button* tidak terisi

3. Pengisian *Text Field* pada saat mengisi bobot faktor eksternal
 - Jika *user* tidak mengisi seluruh *text field*, akan terdapat pesan kesalahan " You must fill the textfield!". (Gambar 5.15)



Gambar 5.15: Tampilan Pesan Error pada saat *text field* tidak terisi seluruhnya

- Jika *user* tidak mengisi *text field*, akan terdapat pesan kesalahan "You cannot move to the other page because you must fill text field first!". (Gambar 5.16)



Gambar 5.16: Tampilan Pesan Error pada saat *text field* tidak terisi

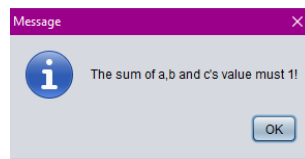
- Jika *user* sudah mengisi *text field* tetapi totalnya tidak 100%, akan terdapat pesan kesalahan "The sum of text fields must 100%!". (Gambar 5.17)



Gambar 5.17: Tampilan Pesan Error pada saat isi dari *text field* tidak berjumlah 100%

4. Pengisian nilai a,b dan c

Jika *user* mengisi nilai a,b dan c jumlahnya tidak 1, akan ada pesan kesalahan yaitu "The sum of a,b and c's value must 1!". (Gambar 5.18)

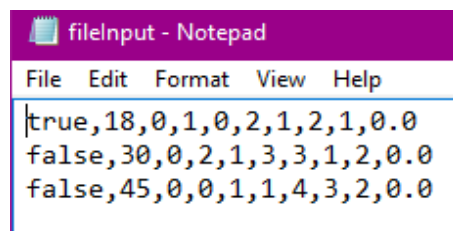


Gambar 5.18: Tampilan Pesan Error pada saat nilai a,b dan c tidak berjumlah 1

5.2.3 Pengujian Pembacaan File

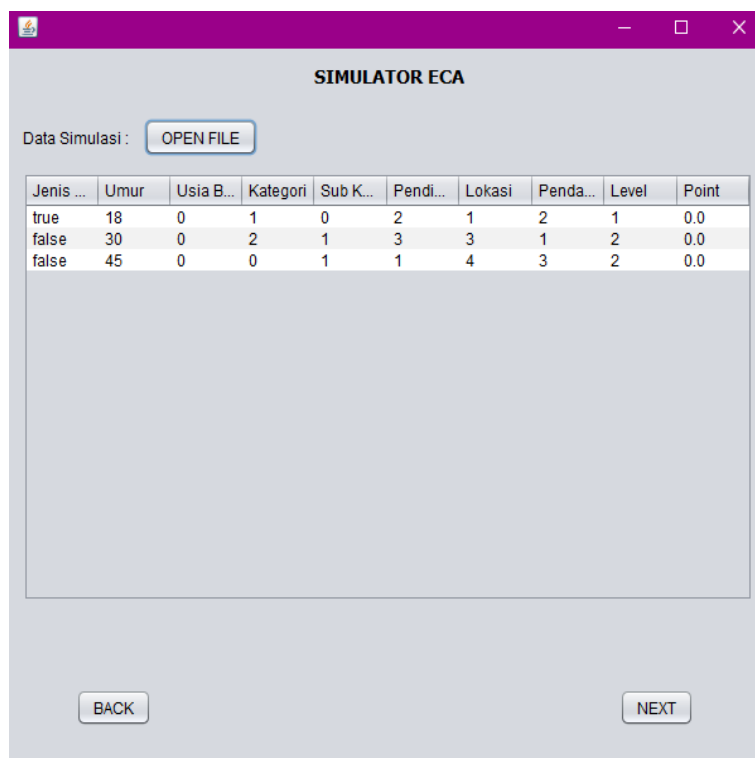
Pengujian ini bertujuan untuk membuktikan kebenaran antara *file* masukan yang *user* berikan dengan akan ditampilkan pada tabel.

Berikut contoh *file* data wirausaha yang diberikan *user* (Gambar 5.19)



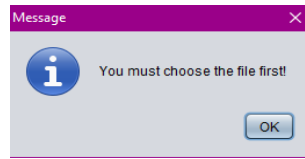
Gambar 5.19: Contoh format *file* data wirausaha

Berikut hasil yang ditampilkan pada tabel : (Gambar 5.20)



Gambar 5.20: Contoh format *file* data wirausaha

Pada pengujian pembacaan *file* data wirausaha jika *user* tidak memasukkan *file* data wirausaha, akan ada pesan kesalahan berupa "You must choose the file first!". (Gambar 5.21)



Gambar 5.21: Tampilan pesan kesalahan apabila *file* data wirausaha belum dipilih

5.2.4 Pengujian Hasil dari Simulasi

Pengujian ini bertujuan untuk menguji hasil perhitungan CIDx dari program sama dengan hasil perhitungan secara manual.

Contoh perhitungan menggunakan hasil perhitungan dari bab 3 pada subbab 3.1.

- Hasil Perhitungan CIDx Program

Berikut hasil perhitungan *Continuity Index* untuk masing-masing wirausahawan:

- Iterasi pada bulan pertama

Dapat dilihat pada gambar 5.22 hasil dari masing-masing wirausahawan mulai dari wirausahawan 1,2 dan 3 pada iterasi pertama.

```
total hasil 1: 20.292437500000002
total hasil 2: 51.5349375
total hasil 3: 15.7524375
```

Gambar 5.22: Hasil iterasi bulan pertama

20.2924

- Iterasi pada bulan kedua

Dapat dilihat pada gambar 5.23 hasil dari masing-masing wirausahawan mulai dari wirausahawan 1,2 dan 3 pada iterasi kedua.

```
total hasil 1: 20.292437500000002
total hasil 2: 51.5349375
total hasil 3: 15.7524375
```

Gambar 5.23: Hasil iterasi bulan kedua

- Iterasi pada bulan ketiga

Dapat dilihat pada gambar 5.24 hasil dari masing-masing wirausahawan mulai dari wirausahawan 1,2 dan 3 pada iterasi ketiga.

```
total hasil 1: 20.292437500000002
total hasil 2: 51.5349375
total hasil 3: 15.7524375
```

Gambar 5.24: Hasil iterasi bulan ketiga

- Iterasi pada bulan keempat

Dapat dilihat pada gambar 5.25 hasil dari masing-masing wirausahawan mulai dari wirausahawan 1,2 dan 3 pada iterasi keempat

```
total hasil 1: 20.292437500000002
total hasil 2: 51.5349375
total hasil 3: 15.7524375
```

Gambar 5.25: Hasil iterasi bulan keempat

- Iterasi pada bulan kelima

Dapat dilihat pada gambar 5.26 hasil dari masing-masing wirausahawan mulai dari wirausahawan 1,2 dan 3 pada iterasi kelima.

```
total hasil 1: 20.452437500000002
total hasil 2: 51.614937499999996
total hasil 3: 15.8324375
```

Gambar 5.26: Hasil iterasi bulan kelima

- Hasil Perhitungan CIDx Manual

Berikut hasil dari perhitungan CIDx manual :

	Entrepreneur 1	Entrepreneur 2	Entrepreneur 3
Bulan pertama	20.09243	51.3749	15.4374
Bulan kedua	20.09243	51.3749	15.4374
Bulan ketiga	20.09243	51.3749	15.4374
Bulan keempat	20.2124	51.4349	15.4974
Bulan kelima	20.2124	51.4349	15.4974

5.3 Eksperimen dan Analisis Hasil

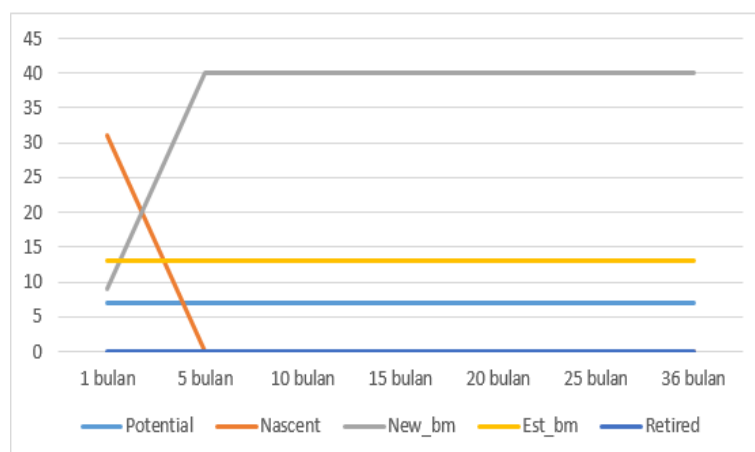
Diberikan eksperimen untuk penelitian ini yaitu sebagai berikut :

Sebuah populasi yang terdiri dari 60 wirausaha. Pada eksperimen ini digunakan tiga buah kriteria ketetangaan, yaitu: level wirausaha, bidang usaha dan lokasi usaha. Masing-masing bobotnya yaitu 30%, 40% dan 30%. Eksperimen ini mengacu pada [4] untuk perhitungan kondisi publik. Tujuan dari eksperimen ini adalah untuk menunjukkan pengaruh dari komposisi a,b,c dan *threshold*. Masing-masing simulasi sebanyak enam kali dengan parameter yang diberikan pada tabel 5.2. Masing-masing simulasi terdiri atas 36 iterasi yang merepresentasikan 36 bulan atau 3 tahun. Pada penelitian ini dimisalkan 1 iterasi adalah 1 bulan.

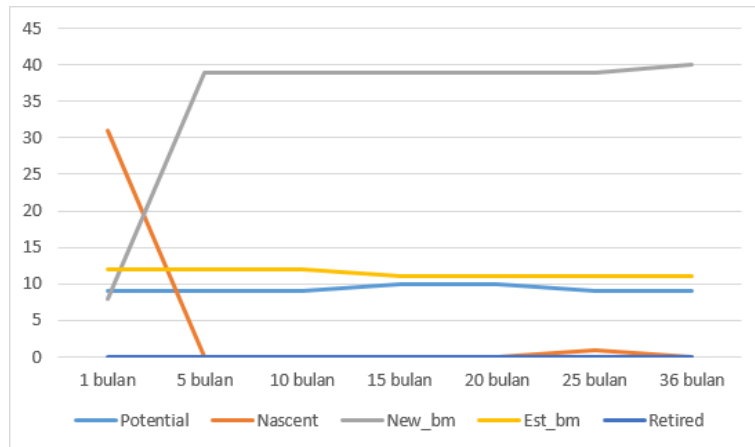
Tabel 5.2: Tabel Parameter Settings

Simulasi	a	b	c	<i>threshold</i>
1	0.7	0.2	0.1	20
2	0.6	0.3	0.1	20
3	0.5	0.4	0.1	20
4	0.5	0.3	0.2	20
5	0.5	0.3	0.2	15
6	0.5	0.3	0.2	10
7	0.6	0.2	0.2	20
8	0.5	0.2	0.3	20
9	0.4	0.2	0.4	20
10	0.3	0.2	0.5	20
11	0.4	0.2	0.4	20
12	0.5	0.2	0.3	20
13	0.3	0.2	0.5	20
14	0.3	0.3	0.4	20
15	0.3	0.4	0.3	20
16	0.3	0.5	0.2	20
17	0.3	0.4	0.3	20
18	0.3	0.3	0.4	20
19	0.6	0.3	0.1	20
20	0.7	0.2	0.1	20
21	0.8	0.1	0.1	20
22	0.5	0.3	0.2	10
23	0.5	0.3	0.2	15
24	0.5	0.3	0.2	20

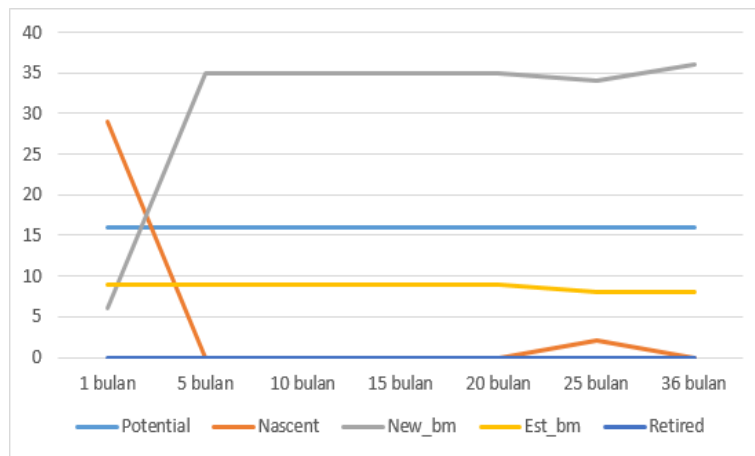
Hasil simulasi akan diberikan pada gambar 5.27, 5.28, 5.29, 5.30, 5.31, 5.32, 5.33, 5.34 dan 5.35, dimana sumbu x menyatakan lamanya iterasi berjalan (bulan) dan sumbu y menyatakan jumlah wirausahawan.



Gambar 5.27: Hasil dari simulasi 1

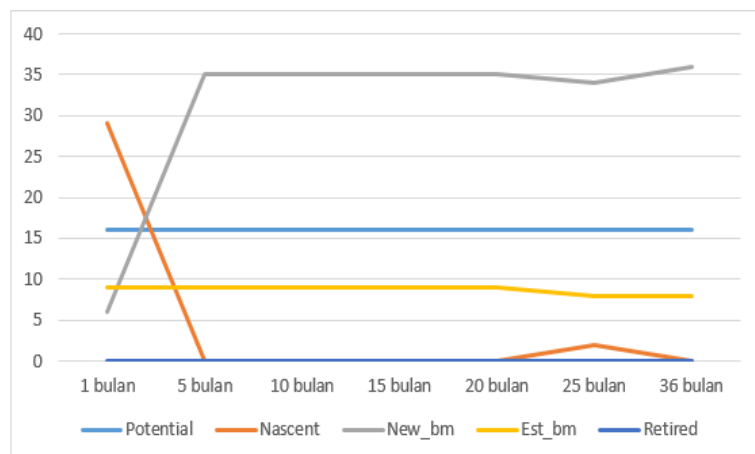


Gambar 5.28: Hasil dari simulasi 2

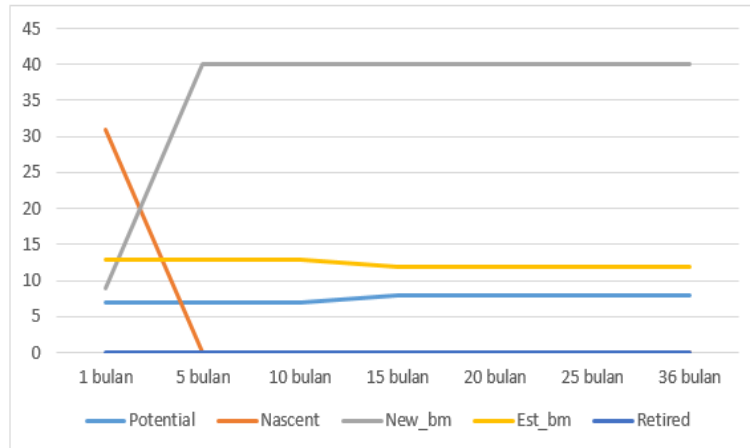


Gambar 5.29: Hasil dari simulasi 3

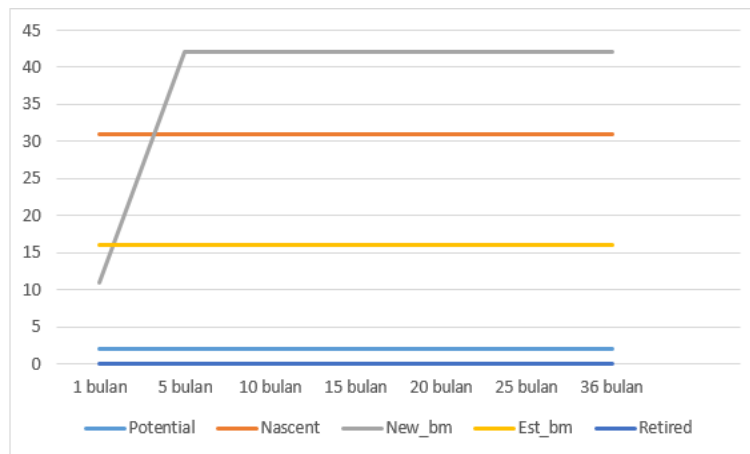
Analisis hasil pada simulasi 1,2 dan 3 yaitu pada gambar 5.27, 5.28 dan 5.29 menunjukkan bahwa semakin besar pengaruh tetangga maka pertumbuhan wirausaha semakin melambat hal ini dapat dibuktikan pada jumlah wirausaha *est_bm* yang menurun dari tahun pertama sampai tahun ketiga.



Gambar 5.30: Hasil dari simulasi 4

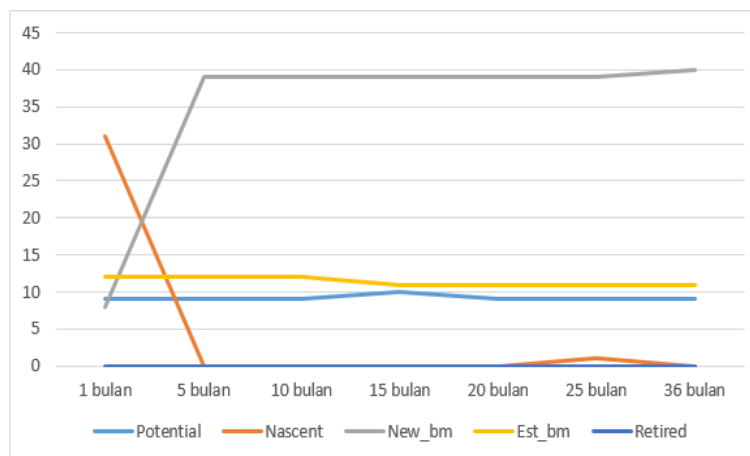


Gambar 5.31: Hasil dari simulasi 5

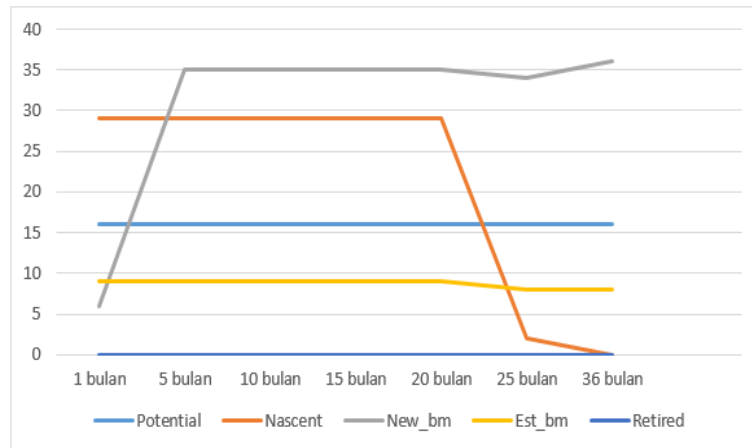


Gambar 5.32: Hasil dari simulasi 6

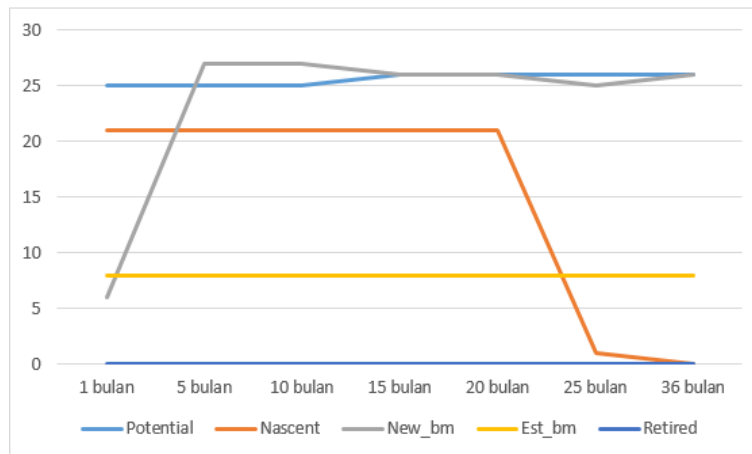
Analisis hasil pada simulasi 4,5 dan 6 yaitu pada gambar 5.30, 5.31 dan 5.32 menunjukkan bahwa wirausaha *new_bm* dan *est_bm* berbanding terbalik dengan nilai *threshold*.



Gambar 5.33: Hasil dari simulasi 7

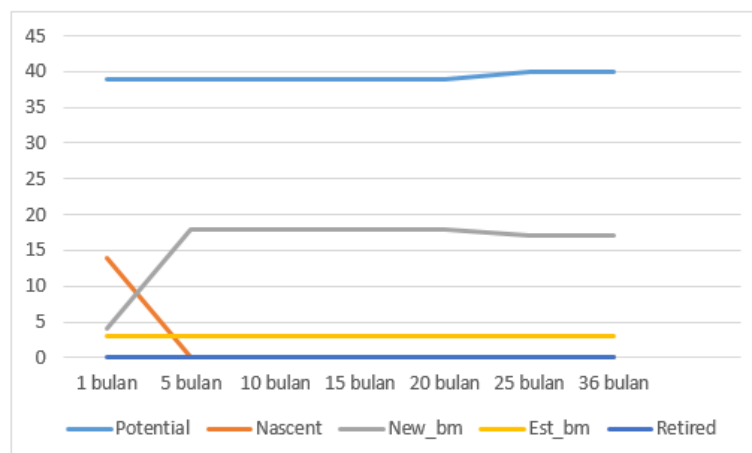


Gambar 5.34: Hasil dari simulasi 8

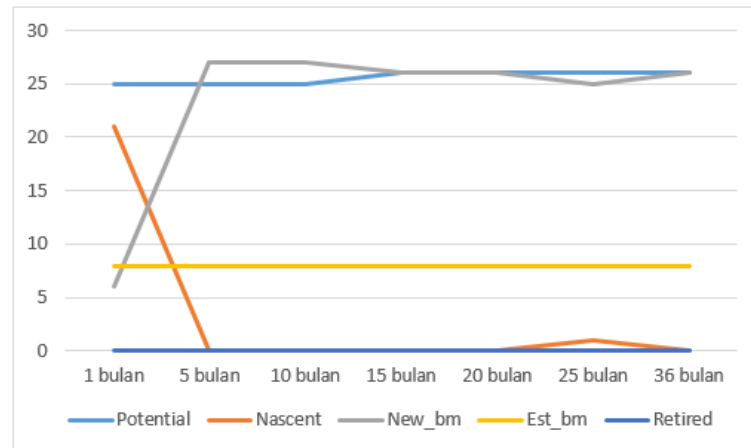


Gambar 5.35: Hasil dari simulasi 9

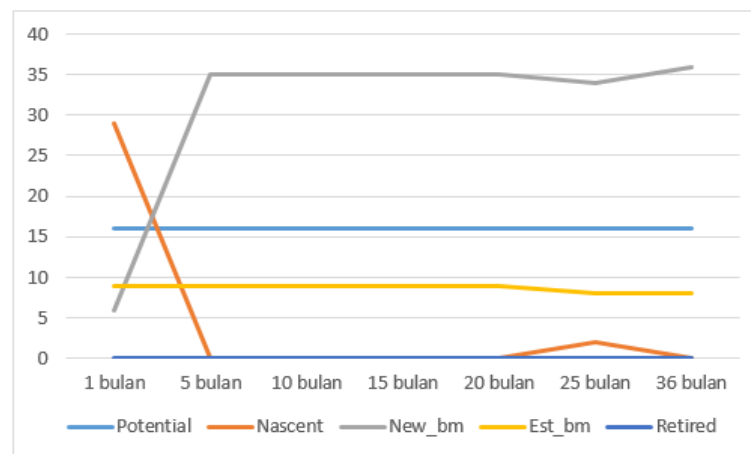
Analisis hasil pada simulasi 7,8 dan 9 yaitu pada gambar 5.33, 5.34 dan 5.35 menunjukkan bahwa semakin besar faktor publik maka pertumbuhan wirausaha semakin melambat hal ini dapat dilihat dari menurunnya jumlah wirausaha *new_bm*, wirausaha *nascent* dan wirausaha *est_bm*.



Gambar 5.36: Hasil dari simulasi 10

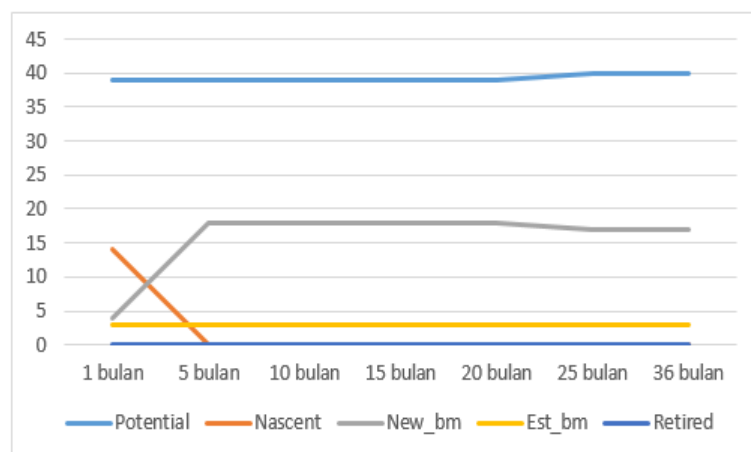


Gambar 5.37: Hasil dari simulasi 11



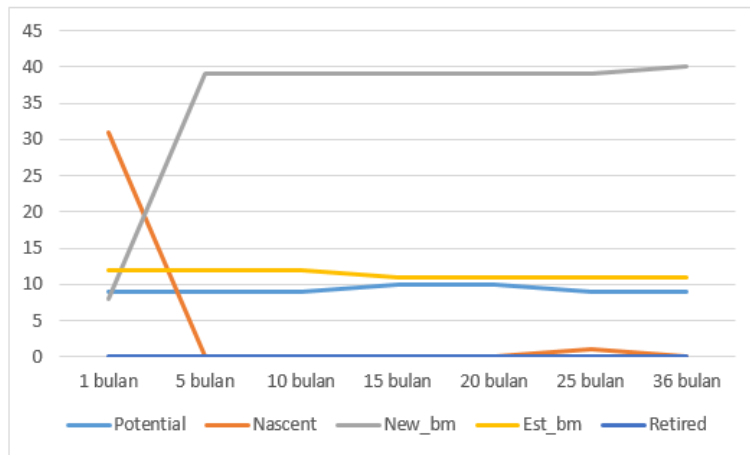
Gambar 5.38: Hasil dari simulasi 12

Analisis hasil pada simulasi 10,11 dan 12 yaitu pada gambar 5.36, 5.37 dan 5.38 menunjukkan bahwa semakin besar pengaruh kondisi internal wirausahawan dan semakin menurunnya pengaruh kondisi publik, pertumbuhan wirausaha akan semakin meningkat. Hal ini dapat dilihat dari simulasi 10,11 dan 12 jumlah wirausahawan pada level new_bm, nascent dan est_bm menaik hal ini berarti banyak wirausahawan yang sudah berhasil dalam menjalankan usahanya.

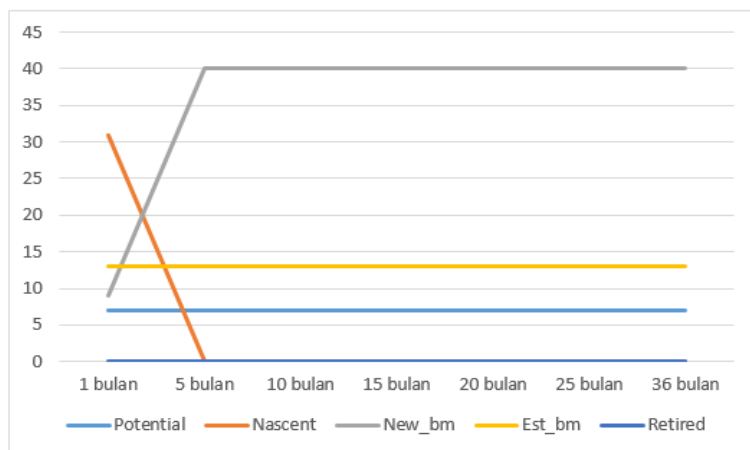


Gambar 5.39: Hasil dari simulasi 13

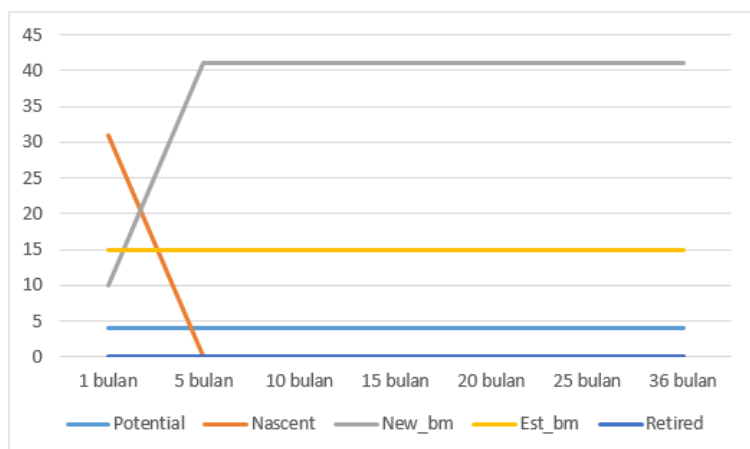
Untuk hasil simulasi 13 sampai 18 grafiknya sama yaitu gambar 5.39. Hal ini menunjukkan bahwa walaupun nilai kondisi tetangga dan kondisi publik diubah, hal ini tidak berpengaruh pada pertumbuhan wirausaha.



Gambar 5.40: Hasil dari simulasi 19



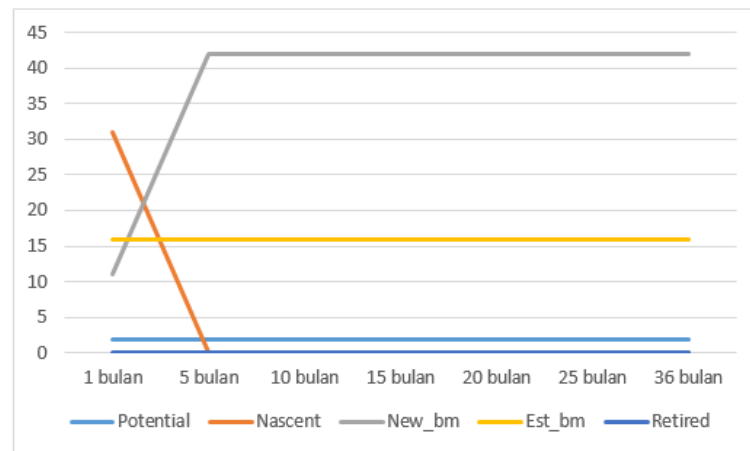
Gambar 5.41: Hasil dari simulasi 20



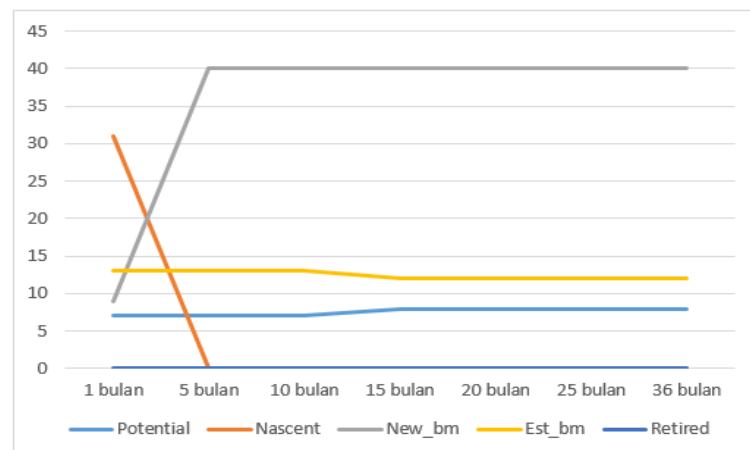
Gambar 5.42: Hasil dari simulasi 21

Analisis hasil pada simulasi 19,20 dan 21 yaitu pada gambar 5.40, 5.41 dan 5.42 menunjukkan

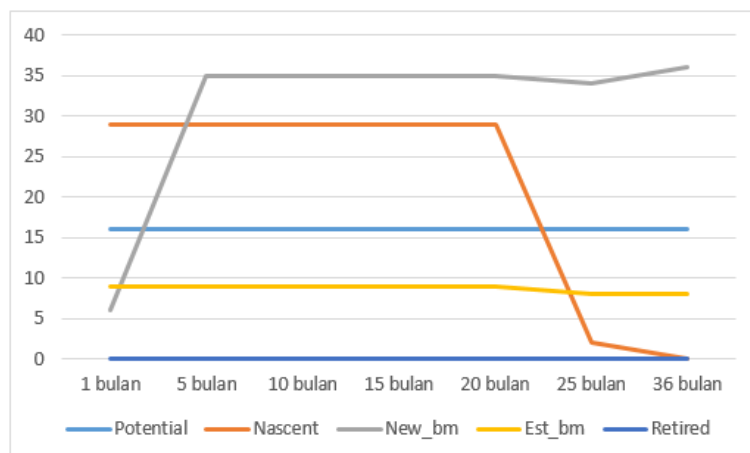
bahwa semakin besar pengaruh kondisi internal wirausahawan dan semakin kecil pengaruh kondisi tetangga, pertumbuhan wirausaha semakin meningkat. Hal ini dapat dilihat bahwa jumlah wirausahawan pada level *new_bm* dan *est_bm* meningkat.



Gambar 5.43: Hasil dari simulasi 22



Gambar 5.44: Hasil dari simulasi 23



Gambar 5.45: Hasil dari simulasi 24

Analisis hasil pada simulasi 22,23 dan 24 yaitu pada gambar 5.43, 5.44 dan 5.45 menunjukkan

bahwa semakin besar pengaruh nilai ambang maka semakin menghambat pertumbuhan wirausaha. Hal ini dapat dilihat bahwa jumlah wirausahawan pada level `new_bm` dan `est_bm` menurun.

BAB 6

KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan terhadap simulator yang telah dibuat, juga saran-saran untuk penelitian ini.

6.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, didapatkan kesimpulan-kesimpulan sebagai berikut :

1. Faktor yang mempengaruhi keberlangsungan wirausaha terdiri dari faktor luar dan faktor internal. Faktor luar dibagi menjadi dua yaitu faktor publik dan faktor tetangga. Berikut akan dijelaskan secara detail :

- Faktor Internal

Faktor yang berasal dari atribut wirausaha itu sendiri, atribut wirausaha dibagi menjadi dua macam yaitu atribut umum (jenis kelamin, umur, level wirausaha, pendapatan, pendidikan, bidang usaha, lokasi) dan atribut psikologis (Perceived Opportunities, Perceived Capabilities, Role Model, Fear of Failure, Entrepreneurial of Intention).

- Faktor Luar

- Faktor publik

Faktor publik terdiri dari:

- (a) Keuangan terkait dengan kewirausahaan
 - (b) Kebijakan pemerintah terkait ekonomi
 - (c) Kebijakan pemerintah terkait pajak
 - (d) Program Pemerintah
 - (e) Pendidikan kewirausahaan pada SD dan SMP
 - (f) Pendidikan kewirausahaan pada SMK, professional dan universitas
 - (g) Transfer penelitian dan pengembangan
 - (h) Infrastruktur komersial dan legal
 - (i) Keterbukaan Pasar
 - (j) Norma, Sosial dan Budaya
 - (k) Infrastruktur Fisik dan Akses Layanan
 - (l) Dinamika Pasar

- Faktor tetangga

Faktor tetangga berasal dari relasi individu wirausaha dengan wirausaha lainnya. Relasi tersebut yaitu lebih dari sama dengan, sama dengan dan kurang dari sama dengan.

2. Dalam memodelkan pertumbuhan wirausaha dengan *Entrepreneurial Cellular Automata* dibutuhkan beberapa proses yaitu :

- (a) Menyesuaikan data wirausaha yang diberikan dengan nilai masing-masing atribut yang ada di GEM 2013.
 - (b) Menghitung nilai *Continuity Index* yang dibagi menjadi 3 bagian yaitu:
 - Menghitung faktor internal dengan cara menjumlahkan nilai atribut pada setiap atribut psikologis lalu dikali dengan bobot atribut psikologis, hasilnya akan dijumlahkan dengan jumlah atribut psikologis lainnya lalu dikali dengan nilai a.
 - Menghitung hubungan ketetanggaan dengan melihat relasi antara wirausaha yang satu dengan wirausaha yang lain. Hasilnya dikalikan dengan nilai b.
 - Menghitung faktor publik dengan cara mengalikan bobot faktor publik (masukan *user*) dengan nilai faktor publik yang ada di GEM 2013. Hasilnya dikalikan dengan nilai c.
 - (c) Mengevaluasi hasil dari perhitungan *Continuity Index* dengan tabel transisi pada subbab 2.8 untuk menentukan wirausaha tersebut mengalami perubahan pada level wirausaha atau tidak. Jika iya, akan terjadi perubahan ketetanggaan pada level wirausaha yang mempengaruhi perhitungan *Continuity Index* selanjutnya.
3. Untuk membangun simulator ECA dibutuhkan pemilihan untuk beberapa atribut (umur, jenis kelamin, level wirausaha, pendapatan, pendidikan, lokasi usaha dan bidang usaha) individu wirausaha dan untuk masing-masing atribut harus disediakan pengisian bobot. Selain atribut yang sudah disebutkan di atas, terdapat 4 atribut yang harus diisi pada relasi ketetanggaan ($\leq, =, \geq$), yaitu umur, pendapatan, level wirausaha dan pendidikan. Pengisian bobot untuk masing-masing faktor publik, pemilihan data wirausaha, dan juga pengisian variabel a,b,c, threshold, dan periode juga dibutuhkan untuk melakukan pengujian simulasi.

6.2 Saran

Berdasarkan hasil penelitian yang dilakukan, berikut adalah beberapa saran untuk mengembangkan perangkat lunak :

1. Memasukkan lebih banyak atribut atau faktor yang mempengaruhi pertumbuhan kewirausahaan, seperti modal, pemasaran, dll.
2. Mengembangkan atau memodifikasi ECA sehingga dapat memperhitungkan pertumbuhan penduduk.
3. Melakukan simulasi dengan data nyata.

DAFTAR REFERENSI

- [1] NPM : 1315351060 (2016) Pengaruh Perkembangan Kewirausahaan Terhadap Tingkat Perekonomian Indonesia. <https://student.unud.ac.id/1315351060/news/13052>. [Online, Diakses 22-Maret-2018].
- [2] Dr.rer.nat. Cecilia Esti Nugraheni, S.T., M.T. dan Vania Natali, S.Kom., M.T. (2017) Pengembangan Model Keberlangsungan Wirausaha dengan Cellular Automata. [Diakses 22-Maret-2018].
- [3] Fery Agus Priana (2012) Pengertian dan definisi wirausaha menurut para ahli. <http://afeyaja.blogspot.co.id/2011/02/pengertian-dan-definisi-wirausaha.html>. [Online; diakses 15-Maret-2018].
- [4] Catharina Badra Nawangpalupi, Gandhi Pawitan, Agus Gunawan, Maria Widyarini, Triyana Iskandarajah (2014) Global Entrepreneurship Monitor 2013 Indonesia Report. [Diakses 15-Maret-2018].
- [5] Global Entrepreneurship Research Association (GERA) (2017) Global Entrepreneurship Monitor Global Report. [Diakses 15-Maret-2018].
- [6] Niloy Ganguly A Survey on Cellular Automata. [Diakses 26-Maret-2018].
- [7] Amanda, Valentina W (2014) Simulasi Infeksi Virus Influenza A Menggunakan Cellular Automaton. [Diakses 26-Maret-2018].
- [8] Dany Satrio Kintoko (2013) Teori Dasar Graf. <http://danysatriokintoko.blogspot.co.id/>. [Online, diakses 28-Maret-2018].
- [9] Wisnu Suhoko (2011) Struktur Data Graf. <https://wisnusuhoko.wordpress.com/2011/01/16/struktur-data-graf/>. [Online, diakses 12-April-2018].

LAMPIRAN A

KODE PROGRAM

Listing A.1: CA.java

```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  import java.util.Random;
9  import java.io.BufferedReader;
10 import java.io.BufferedWriter;
11 import java.io.FileReader;
12 import java.io.FileWriter;
13 import java.io.IOException;
14 import java.io.PrintWriter;
15
16 /**
17 *
18 * @author Vanessa
19 */
20 public class CA {
21
22     int popSize;
23     int neighSize;
24     double threshold; //ambang minimum untuk usaha berlanjut
25     Entrepreneurs[] E;
26     Neighborhoods N;
27     PublicFactor pub;
28     int[] S;
29     float P;
30     float[] delta;
31     float[] sigma;
32     int numOfMonth;
33     CA(int n, int m, int pf) {
34         popSize = n;
35         neighSize = m;
36         E = new Entrepreneurs[n];
37         N = new Neighborhoods(n, m);
38         pub = new PublicFactor(pf);
39         this.numOfMonth = 1; // bulan ke 1
40     }
41     /*
42     * Method untuk menentukan perubahan individu wirausaha
43     */
44     Entrepreneurs[] stateTransition(CA model, double[] composition) {
45         int size = model.popSize;
46         Entrepreneurs[] nextEnt = new Entrepreneurs[size];
47
48         for (int i = 0; i < size; i++) {
49             nextEnt[i] = new Entrepreneurs();
50             model.E[i].copy(nextEnt[i]);
51             if (this.numOfMonth % 12 == 0) {
52                 nextEnt[i].age++; // tiap kelipatan 12 umurnya nambah
53             }
54             nextEnt[i].b_age++;
55             nextLevel(nextEnt[i], i, model, composition);
56         }
57         this.numOfMonth++;
58         return nextEnt;
59     }
60
61     /*
62     * Method untuk menghitung kondisi ketetanggaan
63     */
64     double getNeighborIndex(CA model, int idxEnt) {
65         int size = model.neighSize;
66         double sum = 0.0;
67         for (int i = 0; i < size; i++) {
68             double sum1 = 0.0;
69             for (int j = 0; j < model.popSize; j++) {
70                 sum1 = sum1 + model.N.neighbors[i].neighborMatrix[idxEnt][j];
71             }
72             sum = sum + sum1 / (model.popSize - 1) * model.N.weight[i];
73         }
74         return sum;
75     }
76 }
```

```

76  /*
77  * Method untuk menentukan level wirausaha
78  */
79  void nextLevel(Entrepreneurs ne, int i, CA model, double[] composition) {
80      //kasus umur yang sudah lebih dari 64th
81      if (ne.age > (64)) {
82          ne.level = State.RETIRED;
83          ne.b_age = 0;
84      } else {
85          double idx = getIndex(i, model, composition);
86          threshold = Double.parseDouble(InputDataHandler.getValue("threshold"));
87          if (idx < threshold) {
88              ne.level = State.POTENTIAL;
89              ne.b_age = 0;
90          } else {
91              switch (ne.level) {
92                  case 0: //potential
93                      ne.level = State.NASCENT;
94                      break;
95                  case 1: //nascent
96                      if (ne.b_age > 3) {
97                          ne.level = State.NEW_BOM;
98                          break;
99                      }
100                  case 2: //new_bm
101                      if (ne.b_age > 42) {
102                          ne.level = State.ESTABLISH_BOM;
103                          break;
104                      }
105              }
106          }
107      }
108  }
109  /*
110  * Method untuk menghitung Continuity Index
111  */
112  double getIndex(int i, CA model, double[] composition) {
113      double hasil = composition[0] * model.E[i].point + composition[1] * this.getNeighborIndex(model, i) + composition[2] *
114      this.pub.getPublicIdx();
115      System.out.println("total_hasil: "+hasil);
116      return hasil;
117  }
118  /*
119  * Method untuk mendefinisikan ketetanggaan
120  * 0 jika sama dengan
121  * 1 jika kurang dari sama dengan
122  * 2 jika lebih dari sama dengan
123  */
124  //perubahan -> ditambah casenya
125  void NeighborhoodDefinition() {
126      int n = this.N.numNeighbor;
127      int ng = this.popSize;
128      for (int i = 0; i < n; i++) {
129          for (int j = 0; j < ng; j++) {
130              for (int k = 0; k < ng; k++) {
131                  switch (i) {
132                      case 0: // level
133                          // kalau relasinya sama dengan
134                          if ((this.N.relation[i] == 0) && (this.E[j].level == this.E[k].level)) {
135                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
136                          }
137                          // kalau relasinya kurang dari sama dengan
138                          if ((this.N.relation[i] == 1) && (this.E[j].level <= this.E[k].level)) {
139                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
140                          }
141                          // kalau relasinya lebih dari sama dengan
142                          if ((this.N.relation[i] == 2) && (this.E[j].level >= this.E[k].level)) {
143                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
144                          }
145                          break;
146                      case 1: // b_area
147                          // kalau relasinya sama dengan
148                          if ((this.N.relation[i] == 0) && (this.E[j].b_category == this.E[k].b_category) && (this.E[j].b_area
149                          == this.E[k].b_area)) {
150                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
151                          }
152                          break;
153                      case 2: //location
154                          // kalau relasinya sama dengan
155                          if ((this.N.relation[i] == 0) && (this.E[j].location == this.E[k].location)) {
156                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
157                          }
158                          break;
159                      case 3: // jenis kelamin
160                          if ((this.N.relation[i] == 0) && (this.E[j].sex == this.E[k].sex)) {
161                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
162                          }
163                          break;
164                      case 4: // umur
165                          if ((this.N.relation[i] == 0) && (this.E[j].age == this.E[k].age)) {
166                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
167                          }
168                          // kalau relasinya kurang dari sama dengan
169                          if ((this.N.relation[i] == 1) && (this.E[j].age <= this.E[k].age)) {
170                              this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
171                          }
172                          // kalau relasinya lebih dari sama dengan
173                          if ((this.N.relation[i] == 2) && (this.E[j].age >= this.E[k].age)) {

```

```

173         this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
174     }
175     break;
176
177     case 5: // pendidikan
178         if ((this.N.relation[i] == 0) && (this.E[j].education == this.E[k].education)) {
179             this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
180         }
181         // kalau relasinya kurang dari sama dengan
182         if ((this.N.relation[i] == 1) && (this.E[j].education <= this.E[k].education)) {
183             this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
184         }
185         // kalau relasinya lebih dari sama dengan
186         if ((this.N.relation[i] == 2) && (this.E[j].education >= this.E[k].education)) {
187             this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
188         }
189         break;
190     case 6: // pendapatan
191         if ((this.N.relation[i] == 0) && (this.E[j].income == this.E[k].income)) {
192             this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
193         }
194         // kalau relasinya kurang dari sama dengan
195         if ((this.N.relation[i] == 1) && (this.E[j].income <= this.E[k].income)) {
196             this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
197         }
198         // kalau relasinya lebih dari sama dengan
199         if ((this.N.relation[i] == 2) && (this.E[j].income >= this.E[k].income)) {
200             this.N.neighbors[i].neighborMatrix[j][k] = 1.0;
201         }
202         break;
203     }
204 }
205 }
206 }
207 }
208
209 void genDummyEntrepreneurs() {
210     this.E[0] = new Entrepreneurs(false, 18 * 12, 0, 0, 3, 4, 3, 4, 0, 0.0);
211     this.E[1] = new Entrepreneurs(true, 35 * 12, 0, 0, 2, 4, 0, 4, 1, 0.0);
212     this.E[2] = new Entrepreneurs(false, 55 * 12, 0, 0, 1, 4, 0, 4, 2, 0.0);
213     this.E[3] = new Entrepreneurs(true, 27 * 12, 0, 0, 3, 4, 1, 4, 1, 0.0);
214     this.E[4] = new Entrepreneurs(true, 30 * 12, 0, 0, 1, 4, 1, 4, 0, 0.0);
215     this.E[5] = new Entrepreneurs(false, 45 * 12, 0, 0, 1, 4, 2, 4, 4, 0.0);
216     this.E[6] = new Entrepreneurs(false, 33 * 12, 0, 0, 2, 4, 3, 4, 2, 0.0);
217     this.E[7] = new Entrepreneurs(true, 20 * 12, 0, 0, 3, 4, 2, 4, 0, 0.0);
218     this.E[8] = new Entrepreneurs(false, 38 * 12, 0, 0, 5, 4, 3, 4, 1, 0.0);
219     this.E[9] = new Entrepreneurs(false, 41 * 12, 0, 0, 5, 4, 0, 4, 0, 0.0);
220 }
221
222 void genSimulationData() {
223     int nSim = this.popSize;
224     Random r = new Random();
225     int n;
226     for (int i = 0; i < nSim; i++) {
227         this.E[i] = new Entrepreneurs();
228         n = r.nextInt(nSim);
229         if (n < nSim * 0.6) {
230             this.E[i].sex = State.FEMALE;
231         } else {
232             this.E[i].sex = State.MALE;
233         }
234
235         //location
236         n = r.nextInt(16);
237         this.E[i].location = n;
238         //category business, ada 3 dan area bisnis
239         n = r.nextInt();
240         this.E[i].b_category = n;
241         switch (this.E[i].b_category) {
242             case 0:
243                 this.E[i].b_area = r.nextInt(3);
244                 break;
245             case 1:
246                 this.E[i].b_area = r.nextInt(12);
247                 break;
248             case 2:
249                 this.E[i].b_area = r.nextInt(16);
250                 break;
251         }
252
253         //income
254         n = r.nextInt(6);
255         this.E[i].income = n;
256
257         //education
258         n = r.nextInt(6);
259         this.E[i].education = n;
260
261         int m = r.nextInt(100);
262         if (m > 80) { // 18-24
263             this.E[i].age = (r.nextInt(7) + 18) * 12;
264         } else if (m > 60) { // 25-34
265             this.E[i].age = (r.nextInt(10) + 25) * 12;
266         } else if (m > 40) { // 35-44
267             this.E[i].age = (r.nextInt(10) + 35) * 12;
268         } else if (m > 20) { // 45-54
269             this.E[i].age = (r.nextInt(10) + 45) * 12;
270         } else if (m > 5) { // 55-64
271             this.E[i].age = (r.nextInt(10) + 55) * 12;

```

```

272     } else {
273         this.E[i].age = 65 * 12;
274     }
275
276     if (this.E[i].age > 64 * 12) {
277         this.E[i].level = State.RETIRED;
278     } else {
279         if (this.E[i].age < 25 * 12) {
280             n = r.nextInt(100);
281             if (n < 70) {
282                 this.E[i].level = State.POTENTIAL;
283             } else if (n < 90) {
284                 this.E[i].level = State.NASCENT;
285             } else {
286                 this.E[i].level = State.NEW_BOM;
287             }
288         } else {
289             if (this.E[i].age < 35 * 12) {
290                 n = r.nextInt(100);
291                 if (n < 60) {
292                     this.E[i].level = State.POTENTIAL;
293                 } else if (n < 85) {
294                     this.E[i].level = State.NASCENT;
295                 } else if (n < 95) {
296                     this.E[i].level = State.NEW_BOM;
297                 } else {
298                     this.E[i].level = State.ESTABLISH_BOM;
299                 }
300             } else {
301                 if (this.E[i].age < 45 * 12) {
302                     n = r.nextInt(100);
303                     if (n < 45) {
304                         this.E[i].level = State.POTENTIAL;
305                     } else if (n < 60) {
306                         this.E[i].level = State.NASCENT;
307                     } else if (n < 80) {
308                         this.E[i].level = State.NEW_BOM;
309                     } else {
310                         this.E[i].level = State.ESTABLISH_BOM;
311                     }
312                 } else {
313                     if (this.E[i].age < 55 * 12) {
314                         n = r.nextInt(100);
315                         if (n < 25) {
316                             this.E[i].level = State.POTENTIAL;
317                         } else if (n < 55) {
318                             this.E[i].level = State.NASCENT;
319                         } else if (n < 75) {
320                             this.E[i].level = State.NEW_BOM;
321                         } else {
322                             this.E[i].level = State.ESTABLISH_BOM;
323                         }
324                     } else {
325                         n = r.nextInt(100);
326                         if (n < 10) {
327                             this.E[i].level = State.POTENTIAL;
328                         } else if (n < 20) {
329                             this.E[i].level = State.NASCENT;
330                         } else if (n < 50) {
331                             this.E[i].level = State.NEW_BOM;
332                         } else {
333                             this.E[i].level = State.ESTABLISH_BOM;
334                         }
335                     }
336                     //umur bisnis
337                     switch (this.E[i].level) {
338                         case 1:
339                             this.E[i].b_age = r.nextInt(3) + 1;
340                             break;
341                         case 2:
342                             this.E[i].b_age = r.nextInt(39) + 4;
343                             break;
344                         case 3:
345                             this.E[i].b_age = r.nextInt(this.E[i].age / 18 * 12) + 1;
346                             break;
347                     }
348                 }
349             }
350         }
351     }
352 }
353
354 void writeSimulationData(String namaFile) {
355     String teks = "";
356     try {
357         PrintWriter out = new PrintWriter(new BufferedWriter(new FileWriter(namaFile, true)));
358         System.out.println(this.popSize);
359         for (int i = 0; i < this.popSize; i++) {
360             teks = "";
361             if (this.E[i].sex) {
362                 teks = teks + "1, ";
363             } else {
364                 teks = teks + "0, ";
365             }
366             teks = teks + this.E[i].age + ", ";
367             teks = teks + this.E[i].b_age + ", ";
368             teks = teks + this.E[i].b_category + ", ";
369             teks = teks + this.E[i].b_area + ", ";
370         }
371     }

```



```

371         teks = teks + this.E[i].education + ",";
372         teks = teks + this.E[i].location + ",";
373         teks = teks + this.E[i].income + ",";
374         teks = teks + this.E[i].level;
375         System.out.println(teks);
376     }
377     System.out.close();
378 } catch (IOException e) {
379     System.out.println("Gagal_menulis_ke_file_" + namaFile);
380     e.printStackTrace();
381 }
382 }
383
384 void readSimulationData(String fileName) {
385     String line = "";
386     String separator = ",";
387     BufferedReader br = null;
388     String teks = "";
389     String[] jm;
390
391     try {
392         br = new BufferedReader(new FileReader(fileName));
393         System.out.println("proses_baca_file...");
394         line = br.readLine();
395         line = br.readLine();
396         this.popSize = Integer.parseInt(line.trim());
397         for (int i = 0; i < this.popSize; i++) {
398             line = br.readLine();
399             jm = line.split(separator);
400
401             this.E[i] = new Entrepreneurs();
402             // sex, false = 0, true = 1
403             if (Integer.parseInt(jm[0].trim()) == 0) {
404                 this.E[i].sex = false;
405             } else {
406                 this.E[i].sex = true;
407             }
408             this.E[i].age = Integer.parseInt(jm[1].trim());
409             this.E[i].b_age = Integer.parseInt(jm[2].trim());
410             this.E[i].b_category = Integer.parseInt(jm[3].trim());
411             this.E[i].b_area = Integer.parseInt(jm[4].trim());
412             this.E[i].education = Integer.parseInt(jm[5].trim());
413             this.E[i].location = Integer.parseInt(jm[6].trim());
414             this.E[i].income = Integer.parseInt(jm[7].trim());
415             this.E[i].level = Integer.parseInt(jm[8].trim());
416             this.E[i].point = 0.0;
417         }
418         br.close();
419     } catch (IOException e) {
420         System.out.println("Gagal_membaca_dari_file_" + fileName);
421         e.printStackTrace();
422     }
423 }
424
425 }
426 /*
427  * Method untuk mengeluarkan jumlah wirausaha pada level tertentu
428  */
429 String print(int iter) {
430     int l0 = 0;
431     int l1 = 0;
432     int l2 = 0;
433     int l3 = 0;
434     int l4 = 0;
435     for (int i = 0; i < this.popSize; i++) {
436         switch (this.E[i].level) {
437             case 0:
438                 l0++;
439                 break;
440             case 1:
441                 l1++;
442                 break;
443             case 2:
444                 l2++;
445                 break;
446             case 3:
447                 l3++;
448                 break;
449             case 4:
450                 l4++;
451                 break;
452         }
453     }
454     return (iter + "," + l0 + "," + l1 + "," + l2 + "," + l3 + "," + l4);
455 }
456 /*
457  * Method untuk menghitung kondisi internal wirausaha
458  * parameternya berisi dengan nilai-nilai atribut psikologis dari GEM 2013
459  */
460 // perubahan : ditambahn faktor psikologisnya
461 void calculatePoint(double[] POAm, double[] POAf, double[] POEm, double[] POEf, double[] POLm, double[] POLf, double[] POIm,
    double[] POIf, double[] PCAm, double[] PCAf, double[] PCEm, double[] PCEf, double[] PCLm, double[] PCLf, double[] PCIm,
    double[] PCIf, double[] RMAm, double[] RMAf, double[] RMIm, double[] RMIf, double[] FFAf, double[] FFAm, double[] FFEf,
    double[] FFEIm, double[] FFLf, double[] FFLm, double[] MALf, double[] MALm, double[] MAIf, double[] MAIm, double[] HSSIf,
    double[] HSSIm, double[] HSSLf, double[] HSSLm, double[] HSSAf, double[] HSSAm, double[] HSSef, double[] HSSEm) {
462     for (int i = 0; i < this.popSize; i++) {
463         int a = getAgeRange(E[i].age);
464         if (this.E[i].sex) {

```

```

465         E[i].point = (POAm[a] + POEm[E[i].education] + POLm[E[i].location] + POIm[E[i].income]) * 0.2 + (PCAm[a] + PCEm[E[
            i].education] + PCLm[E[i].location] + PCIm[E[i].income]) * 0.25 + (RMAm[a] + RMIm[E[i].income]) * 0.3 + (
            FFAm[a] + FFEf[E[i].education] + FFLm[E[i].location]) * 0.1 + (MALm[E[i].location] + MAIm[E[i].income]) *
            0.05 + (HSSAm[a] + HSSIm[E[i].income] + HSSLm[E[i].location] + HSSEm[E[i].education]) * 0.1;
466     } else {
467         E[i].point = (POAf[a] + POEf[E[i].education] + POLf[E[i].location] + POIf[E[i].income]) * 0.2 + (PCAf[a] + PCEf[E[
            i].education] + PCLf[E[i].location] + PCIf[E[i].income]) * 0.25 + (RMAf[a] + RMIf[E[i].income]) * 0.3 + (
            FFAf[a] + FFEf[E[i].education] + FFLf[E[i].location]) * 0.1 + (MALf[E[i].location] + MAIf[E[i].income]) *
            0.05 + (HSSAf[a] + HSSIf[E[i].income] + HSSLf[E[i].location] + HSSEf[E[i].education]) * 0.1;
468     }
469 }
470 }
471 /*
472  * Method untuk mengelompokkan rentang umur sesuai dengan GEM 2013
473  * a merupakan umur dari wirausaha
474  */
475 int getAgeRange(int a) {
476     int ageC = -1;
477     if (a >= 55 && a <= 64) {
478         ageC = 0;
479     }
480     if (a >= 45 && a <= 54) {
481         ageC = 1;
482     }
483     if (a >= 35 && a <= 44) {
484         ageC = 2;
485     }
486     if (a >= 25 && a <= 34) {
487         ageC = 3;
488     }
489     if (a >= 18 && a <= 24) {
490         ageC = 4;
491     }
492     return ageC;
493 }
494 }
495 }

```

Listing A.2: Entrepreneurs.java

```

1  /*
2  /*
3  * To change this license header, choose License Headers in Project Properties.
4  * To change this template file, choose Tools | Templates
5  * and open the template in the editor.
6  */
7  package ecasimulatorjframe;
8
9  /**
10   *
11   * @author Vanessa
12   */
13  public class Entrepreneurs {
14      int level;
15      int age;
16      boolean sex;
17      int b_age;
18      int b_category; // bidang usaha, misal makanan
19      int b_area; // makanan ringan, makanan berat
20      int education;
21      int location;
22      int income;
23      double point;
24      //penambahan
25
26      Entrepreneurs(){
27          sex=false;
28          age = 0;
29          b_age = 0;
30          b_category = 0;
31          b_area = 0;
32          education = 0;
33          location = 0;
34          income = 0;
35          level = 0;
36          point = 0.0;
37      }
38
39      Entrepreneurs(boolean s, int a, int ba, int cat,int area, int edu, int loc, int inc, int l, double p){
40          this.sex = s;
41          this.age = a;
42          this.b_age = ba;
43          this.b_category = cat;
44          this.b_area = area;
45          this.education = edu;
46          this.location = loc;
47          this.income = inc; //pendapatan
48          this.level = l;
49          this.point = p;
50      }
51
52      void copy(Entrepreneurs e){
53          e.sex = this.sex;
54          e.age = this.age;
55          e.b_age = this.b_age;
56          e.b_category = this.b_category;
57          e.b_area = this.b_area;
58          e.education = this.education;

```

```

59     e.location = this.location;
60     e.income = this.income;
61     e.level = this.level;
62 }
63 /*
64  * method untuk mengeluarkan hasil perubahan individu wirausahawan
65  */
66 public String toString2(){
67     return sex + "," + age + "," + b_age + "," + b_category + "," + b_area + "," + education + "," + location + "," + income + "," + level;
68 }
69
70 void genDummy(CA model){
71 }
72 }
73 }

```

Listing A.3: Neighbor.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  /**
9   *
10  * @author Vanessa
11  */
12  public class Neighbor {
13      double[][] neighborMatrix;
14      /*
15       * Method untuk membuat matriks berdasarkan atribut tetangga
16       * n merupakan jumlah wirausaha
17       */
18      Neighbor(int n){
19          neighborMatrix = new double[n][n];
20          for (int i = 0; i < n; i++) {
21              for (int j = 0; j < n; j++) {
22                  neighborMatrix[i][j] = 0.0;
23              }
24          }
25      }
26  }

```

Listing A.4: Neighborhoods.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  /**
9   *
10  * @author Vanessa
11  * himpunan ketetanggaan tersusun atas sejumlah ketetanggaan
12  */
13  public class Neighborhoods {
14      int numNeighbor; // banyaknya ketetanggaan
15      Neighbor[] neighbors;
16      double[] weight;
17      int[] relation; // jenis hubungan ketetanggaan, sama dengan, lebih kecil atau yang lain --> perlu didefinisikan
18      /*
19       * konstruktor untuk membuat matriks neighbor berdasarkan banyaknya tetangga
20       * n untuk jumlah wirausaha
21       * m untuk banyaknya tetangga
22       */
23      Neighborhoods(int n, int m){
24          this.numNeighbor = m;
25          neighbors = new Neighbor[m];
26          weight = new double[m];
27          relation = new int[m];
28
29          for (int i = 0; i < m; i++) {
30              this.numNeighbor = m;
31              neighbors[i] = new Neighbor(n);
32              weight[i] = 0.0;
33              relation[i] = 0;
34          }
35      }
36
37      public void setWeight(double[] weight) {
38          this.weight = weight;
39      }
40
41      public void setRelation(int[] relation) {
42          this.relation = relation;
43      }
44
45      public void setNumNeighbor(int numNeighbor) {
46          this.numNeighbor = numNeighbor;
47      }
48
49  }

```

```
50| }
```

Listing A.5: PublicFactor.java

```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  /**
9   *
10  * @author Vanessa
11  */
12  public class PublicFactor {
13
14      double[] factors;
15      double[] weights;
16
17      PublicFactor(int n) {
18          factors = new double[n];
19          weights = new double[n];
20      }
21
22      public void setFactors(double[] f) {
23          this.factors = f;
24      }
25
26      public void setWeights(double[] w) {
27          this.weights = w;
28      }
29      /*
30      * Method untuk menghitung hasil faktor publik
31      */
32      double getPublicIdx() {
33          double idx = 0.0;
34          for (int i = 0; i < factors.length; i++) {
35              idx = idx + factors[i] * weights[i];
36          }
37          return idx / factors.length;
38      }
39  }
40 }
```

Listing A.6: State.java

```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  /**
9   *
10  * @author Vanessa
11  */
12  public class State {
13      public static int POTENTIAL = 0;
14      public static int NASCENT = 1;
15      public static int NEW_BOM = 2;
16      public static int ESTABLISH_BOM = 3;
17      public static int RETIRED = 4;
18      public static boolean FEMALE = false;
19      public static boolean MALE = true;
20  }
```

Listing A.7: TampilanBobotKetetanggaan.java

```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  import javax.swing.JOptionPane;
9
10 /**
11  *
12  * @author Vanessa
13  */
14  public class TampilanBobotKetetanggaan extends javax.swing.JFrame {
15
16      /**
17       * Creates new form ECASimulator
18       */
19      public TampilanBobotKetetanggaan() {
20          initComponents();
21          nilaiUmurInternal.setEnabled(false);
22          nilaiLokasiInternal.setEnabled(false);
23          nilaiPendapatanInternal.setEnabled(false);
24          nilaiPendidikanInternal.setEnabled(false);
25          nilaiUsahaInternal.setEnabled(false);
26      }
27  }
```

```

26 |         nilaiJenisKelaminInternal.setEnabled(false);
27 |         nilaiLevelInternal.setEnabled(false);
28 |     }
29 |
30 |
31 | /**
32 |  * This method is called from within the constructor to initialize the form.
33 |  * WARNING: Do NOT modify this code. The content of this method is always
34 |  * regenerated by the Form Editor.
35 |  */
36 | @SuppressWarnings("unchecked")
37 | // <editor-fold defaultstate="collapsed" desc="Generated Code">
38 | private void initComponents() {
39 |
40 |     jPanel1 = new javax.swing.JPanel();
41 |     jLabel1 = new javax.swing.JLabel();
42 |     jLabel2 = new javax.swing.JLabel();
43 |     umurCBInternal = new javax.swing.JCheckBox();
44 |     levelCBInternal = new javax.swing.JCheckBox();
45 |     pendidikanCBInternal = new javax.swing.JCheckBox();
46 |     pendapatanCBInternal = new javax.swing.JCheckBox();
47 |     jenisKelaminCBInternal = new javax.swing.JCheckBox();
48 |     lokasiCBInternal = new javax.swing.JCheckBox();
49 |     bUsahaCBInternal = new javax.swing.JCheckBox();
50 |     nilaiUmurInternal = new javax.swing.JTextField();
51 |     nilaiLevelInternal = new javax.swing.JTextField();
52 |     nilaiPendidikanInternal = new javax.swing.JTextField();
53 |     nilaiPendapatanInternal = new javax.swing.JTextField();
54 |     nilaiJenisKelaminInternal = new javax.swing.JTextField();
55 |     nilaiLokasiInternal = new javax.swing.JTextField();
56 |     nilaiUsahaInternal = new javax.swing.JTextField();
57 |     jLabel3 = new javax.swing.JLabel();
58 |     jLabel4 = new javax.swing.JLabel();
59 |     jLabel5 = new javax.swing.JLabel();
60 |     jLabel6 = new javax.swing.JLabel();
61 |     jLabel7 = new javax.swing.JLabel();
62 |     jLabel8 = new javax.swing.JLabel();
63 |     jLabel9 = new javax.swing.JLabel();
64 |     nextButton = new javax.swing.JButton();
65 |
66 |     setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
67 |
68 |     jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
69 |     jLabel1.setText("SIMULATOR_ECA");
70 |
71 |     jLabel2.setText("Bobot_Ketetangaan_Wirusaha:");
72 |
73 |     umurCBInternal.setText("UMUR:");
74 |     umurCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
75 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
76 |             umurCBInternalMouseClicked(evt);
77 |         }
78 |     });
79 |     umurCBInternal.addActionListener(new java.awt.event.ActionListener() {
80 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
81 |             umurCBInternalActionPerformed(evt);
82 |         }
83 |     });
84 |
85 |     levelCBInternal.setText("LEVEL:");
86 |     levelCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
87 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
88 |             levelCBInternalMouseClicked(evt);
89 |         }
90 |     });
91 |
92 |     pendidikanCBInternal.setText("PENDIDIKAN:");
93 |     pendidikanCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
94 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
95 |             pendidikanCBInternalMouseClicked(evt);
96 |         }
97 |     });
98 |
99 |     pendapatanCBInternal.setText("PENDAPATAN:");
100 |     pendapatanCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
101 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
102 |             pendapatanCBInternalMouseClicked(evt);
103 |         }
104 |     });
105 |     pendapatanCBInternal.addActionListener(new java.awt.event.ActionListener() {
106 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
107 |             pendapatanCBInternalActionPerformed(evt);
108 |         }
109 |     });
110 |
111 |     jenisKelaminCBInternal.setText("JENIS_KELAMIN:");
112 |     jenisKelaminCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
113 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
114 |             jenisKelaminCBInternalMouseClicked(evt);
115 |         }
116 |     });
117 |
118 |     lokasiCBInternal.setText("LOKASI:");
119 |     lokasiCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
120 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
121 |             lokasiCBInternalMouseClicked(evt);
122 |         }
123 |     });
124 |

```

```

125     bUsahaCBInternal.setText("BIDANG_USAHA_");
126     bUsahaCBInternal.addMouseListener(new java.awt.event.MouseAdapter() {
127         public void mouseClicked(java.awt.event.MouseEvent evt) {
128             bUsahaCBInternalMouseClicked(evt);
129         }
130     });
131
132     nilaiUmurInternal.addContainerListener(new java.awt.event.ContainerAdapter() {
133         public void componentAdded(java.awt.event.ContainerEvent evt) {
134             nilaiUmurInternalComponentAdded(evt);
135         }
136     });
137     nilaiUmurInternal.addMouseListener(new java.awt.event.MouseAdapter() {
138         public void mouseClicked(java.awt.event.MouseEvent evt) {
139             nilaiUmurInternalMouseClicked(evt);
140         }
141     });
142     nilaiUmurInternal.addInputMethodListener(new java.awt.event.InputMethodListener() {
143         public void caretPositionChanged(java.awt.event.InputMethodEvent evt) {
144         }
145         public void inputMethodTextChanged(java.awt.event.InputMethodEvent evt) {
146             nilaiUmurInternalInputMethodTextChanged(evt);
147         }
148     });
149     nilaiUmurInternal.addActionListener(new java.awt.event.ActionListener() {
150         public void actionPerformed(java.awt.event.ActionEvent evt) {
151             nilaiUmurInternalActionPerformed(evt);
152         }
153     });
154
155     nilaiLevelInternal.addActionListener(new java.awt.event.ActionListener() {
156         public void actionPerformed(java.awt.event.ActionEvent evt) {
157             nilaiLevelInternalActionPerformed(evt);
158         }
159     });
160
161     nilaiPendapatanInternal.addActionListener(new java.awt.event.ActionListener() {
162         public void actionPerformed(java.awt.event.ActionEvent evt) {
163             nilaiPendapatanInternalActionPerformed(evt);
164         }
165     });
166
167     nilaiLokasiInternal.addActionListener(new java.awt.event.ActionListener() {
168         public void actionPerformed(java.awt.event.ActionEvent evt) {
169             nilaiLokasiInternalActionPerformed(evt);
170         }
171     });
172
173     jLabel3.setText("");
174     jLabel4.setText("");
175     jLabel5.setText("");
176     jLabel6.setText("");
177     jLabel7.setText("");
178     jLabel8.setText("");
179     jLabel9.setText("");
180
181     nextButton.setText("NEXT");
182     nextButton.addMouseListener(new java.awt.event.MouseAdapter() {
183         public void mouseClicked(java.awt.event.MouseEvent evt) {
184             nextButtonMouseClicked(evt);
185         }
186     });
187     nextButton.addActionListener(new java.awt.event.ActionListener() {
188         public void actionPerformed(java.awt.event.ActionEvent evt) {
189             nextButtonActionPerformed(evt);
190         }
191     });
192
193     javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
194     jPanel1.setLayout(jPanel1Layout);
195     jPanel1Layout.setHorizontalGroup(
196         jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
197             .addGroup(jPanel1Layout.createSequentialGroup()
198                 .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
199                     .addGroup(jPanel1Layout.createSequentialGroup()
200                         .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
201                             .add(jPanel1Layout.createSequentialGroup()
202                                 .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
203                                     .add(jPanel1Layout.createSequentialGroup()
204                                         .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
205                                             .add(jPanel1Layout.createSequentialGroup()
206                                                 .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
207                                                     .add(jPanel1Layout.createSequentialGroup()
208                                                         .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
209                                                             .add(jPanel1Layout.createSequentialGroup()
210                                                                 .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
211                                                                     .add(jPanel1Layout.createSequentialGroup()
212                                                                         .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
213                                                                             .add(jPanel1Layout.createSequentialGroup()
214                                                                                 .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
215                                                                                     .add(jPanel1Layout.createSequentialGroup()
216                                                                                         .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
217                                                                     .add(jPanel1Layout.createSequentialGroup()
218                                                                         .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
219                                                                             .add(jPanel1Layout.createSequentialGroup()
220                                                                                 .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
221                                                                                     .add(jPanel1Layout.createSequentialGroup()
222

```

```

223         .addGap(34, 34, 34)
224         .addComponent(nilaiUmurInternal, javax.swing.GroupLayout.PREFERRED_SIZE, 53,
225             javax.swing.GroupLayout.PREFERRED_SIZE))
226     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.
227         createSequentialGroup())
228     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
229     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
230         LEADING)
231         .addComponent(nilaiPendidikanInternal, javax.swing.GroupLayout.Alignment.
232             TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 53, javax.swing.
233             GroupLayout.PREFERRED_SIZE)
234         .addComponent(nilaiPendapatanInternal, javax.swing.GroupLayout.Alignment.
235             TRAILING, javax.swing.GroupLayout.PREFERRED_SIZE, 53, javax.swing.
236             GroupLayout.PREFERRED_SIZE))))
237     .addComponent(nilaiJenisKelaminInternal, javax.swing.GroupLayout.PREFERRED_SIZE, 53, javax
238         .swing.GroupLayout.PREFERRED_SIZE)
239     .addComponent(nilaiLokasiInternal, javax.swing.GroupLayout.PREFERRED_SIZE, 53, javax.swing
240         .GroupLayout.PREFERRED_SIZE)
241     .addComponent(nilaiUsahaInternal, javax.swing.GroupLayout.PREFERRED_SIZE, 53, javax.swing.
242         GroupLayout.PREFERRED_SIZE))
243     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
244     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup
245         ()
246         .addComponent(jenisKelaminCBInternal)
247         .addGap(68, 68, 68))
248         .addComponent(lokalasiCBInternal)
249         .addComponent(bUsahaCBInternal)))
250     .addGap(18, 18, 18)
251     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
252         .addComponent(jLabel3)
253         .addComponent(jLabel4)
254         .addComponent(jLabel5)
255         .addComponent(jLabel6)
256         .addComponent(jLabel7)
257         .addComponent(jLabel8)
258         .addComponent(jLabel9))
259     .addGroup(jPanel1Layout.createSequentialGroup())
260     .addGap(131, 131, 131)
261     .addComponent(jLabel1))
262     .addGap(0, 129, Short.MAX_VALUE))
263     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup())
264     .addGap(0, 0, Short.MAX_VALUE)
265     .addComponent(nextButton))
266     .addContainerGap()
267 );
268 jPanel1Layout.setVerticalGroup(
269     jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
270     .addGroup(jPanel1Layout.createSequentialGroup())
271     .addContainerGap()
272     .addComponent(jLabel1)
273     .addGap(36, 36, 36)
274     .addComponent(jLabel2)
275     .addGap(18, 18, 18)
276     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
277         .addComponent(umurCBInternal)
278         .addComponent(nilaiUmurInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
279             javax.swing.GroupLayout.PREFERRED_SIZE)
280         .addComponent(jLabel3))
281     .addGap(18, 18, 18)
282     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
283         .addComponent(levelCBInternal)
284         .addComponent(nilaiLevelInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE
285             , javax.swing.GroupLayout.PREFERRED_SIZE)
286         .addComponent(jLabel4))
287     .addGap(18, 18, 18)
288     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
289         .addComponent(pendidikanCBInternal)
290         .addComponent(nilaiPendidikanInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
291             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
292         .addComponent(jLabel5))
293     .addGap(18, 18, 18)
294     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
295         .addComponent(pendapatanCBInternal)
296         .addComponent(nilaiPendapatanInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
297             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
298         .addComponent(jLabel6))
299     .addGap(18, 18, 18)
300     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
301         .addComponent(jenisKelaminCBInternal)
302         .addComponent(nilaiJenisKelaminInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
303             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
304         .addComponent(jLabel7))
305     .addGap(18, 18, 18)
306     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
307         .addComponent(lokalasiCBInternal)
308         .addComponent(nilaiLokasiInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
309             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
310         .addComponent(jLabel8))
311     .addGap(18, 18, 18)
312     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
313         .addComponent(bUsahaCBInternal)
314         .addComponent(nilaiUsahaInternal, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE
315             , javax.swing.GroupLayout.PREFERRED_SIZE)
316         .addComponent(jLabel9))
317     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 26, Short.MAX_VALUE)
318     .addComponent(nextButton)
319     .addGap(21, 21, 21))
320 );

```

```

304     javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
305     getContentPane().setLayout(layout);
306     layout.setHorizontalGroup(
307         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
308         .addGroup(layout.createSequentialGroup()
309             .addGap(21, 21, 21)
310             .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.
311                 GroupLayout.PREFERRED_SIZE)
312             .addContainerGap(22, Short.MAX_VALUE))
313         );
314     layout.setVerticalGroup(
315         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
316         .addGroup(layout.createSequentialGroup()
317             .addContainerGap()
318             .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
319             .addContainerGap())
320         );
321     pack();
322 } // </editor-fold>
323
324 private void nilaiPendapatanInternalActionPerformed(java.awt.event.ActionEvent evt) {
325     // TODO add your handling code here:
326 }
327
328 private void nilaiLevelInternalActionPerformed(java.awt.event.ActionEvent evt) {
329     // TODO add your handling code here:
330 }
331
332 private void pendapatanCBInternalActionPerformed(java.awt.event.ActionEvent evt) {
333     // TODO add your handling code here:
334 }
335
336 private void nilaiLokasiInternalActionPerformed(java.awt.event.ActionEvent evt) {
337     // TODO add your handling code here:
338 }
339
340 private void nextButtonActionPerformed(java.awt.event.ActionEvent evt) {
341
342 }
343
344 private void umurCBInternalActionPerformed(java.awt.event.ActionEvent evt) {
345
346 }
347
348 private void nilaiUmurInternalActionPerformed(java.awt.event.ActionEvent evt) {
349
350 }
351
352 private void nextButtonMouseClicked(java.awt.event.MouseEvent evt) {
353     double nilaiUmur = 0.0;
354     double nilaiLevel = 0.0;
355     double nilaiPendidikan = 0.0;
356     double nilaiPendapatan = 0.0;
357     double nilaiLokasi = 0.0;
358     double nilaiUsaha = 0.0;
359     double nilaiJenisKelamin = 0.0;
360     boolean checker = true;
361     if (umurCBInternal.isSelected()) {
362         InputDataHandler.jmlChecklist();
363         if (nilaiUmurInternal.getText().equals("")) {
364             InputDataHandler.inputDataInternal("umurInternal", null);
365             checker = false; // false karena nilainya null
366         } else {
367             nilaiUmur = Double.parseDouble(nilaiUmurInternal.getText()) / 100.0;
368             String nilaiU = Double.toString(nilaiUmur);
369             System.out.println(nilaiU);
370             InputDataHandler.inputDataInternal("umurInternal", nilaiU);
371         }
372     }
373
374 }
375
376 if (levelCBInternal.isSelected()) {
377     InputDataHandler.jmlChecklist();
378     if (nilaiLevelInternal.getText().equals("")) {
379         InputDataHandler.inputDataInternal("levelInternal", null);
380         checker = false; // false karena nilainya null
381     } else {
382         nilaiLevel = Double.parseDouble(nilaiLevelInternal.getText()) / 100.0;
383         String nilaiL = Double.toString(nilaiLevel);
384         System.out.println(nilaiL);
385         InputDataHandler.inputDataInternal("levelInternal", nilaiL);
386     }
387 }
388
389 if (pendidikanCBInternal.isSelected()) {
390     InputDataHandler.jmlChecklist();
391     if (nilaiPendidikanInternal.getText().equals("")) {
392         InputDataHandler.inputDataInternal("pendidikanInternal", null);
393         checker = false; // false karena nilainya null
394     } else {
395         nilaiPendidikan = Double.parseDouble(nilaiPendidikanInternal.getText()) / 100.0;
396         String nilaiPendi = Double.toString(nilaiPendidikan);
397         System.out.println(nilaiPendi);
398         InputDataHandler.inputDataInternal("pendidikanInternal", nilaiPendi);
399     }
400 }

```



```

401 |
402 |         if (pendapatanCBInternal.isSelected()) {
403 |             InputDataHandler.jmlChecklist();
404 |             if (nilaiPendapatanInternal.getText().equals("")) {
405 |                 InputDataHandler.inputDataInternal("pendapatanInternal", null);
406 |                 checker = false; // false karena nilainya null
407 |             } else {
408 |                 nilaiPendapatan = Double.parseDouble(nilaiPendapatanInternal.getText()) / 100.0;
409 |                 String nilaiPenda= Double.toString(nilaiPendapatan);
410 |                 // System.out.println(nilaiU);
411 |                 InputDataHandler.inputDataInternal("pendapatanInternal", nilaiPenda);
412 |             }
413 |         }
414 |
415 |         if (jenisKelaminCBInternal.isSelected()) {
416 |             InputDataHandler.jmlChecklist();
417 |             if (nilaiJenisKelaminInternal.getText().equals("")) {
418 |                 InputDataHandler.inputDataInternal("jenisKelaminInternal", null);
419 |                 checker = false; // false karena nilainya null
420 |             } else {
421 |                 nilaiJenisKelamin = Double.parseDouble(nilaiJenisKelaminInternal.getText()) / 100.0;
422 |                 String nilaiJK = Double.toString(nilaiJenisKelamin);
423 |                 // System.out.println(nilaiJK);
424 |                 InputDataHandler.inputDataInternal("jenisKelaminInternal", nilaiJK);
425 |             }
426 |         }
427 |
428 |         if (lokasiCBInternal.isSelected()) {
429 |             InputDataHandler.jmlChecklist();
430 |             if (nilaiLokasiInternal.getText().equals("")) {
431 |                 InputDataHandler.inputDataInternal("lokasiInternal", null);
432 |                 checker = false; // false karena nilainya null
433 |             } else {
434 |                 nilaiLokasi = Double.parseDouble(nilaiLokasiInternal.getText()) / 100.0;
435 |                 String nilaiL = Double.toString(nilaiLokasi);
436 |                 // System.out.println(nilaiU);
437 |                 InputDataHandler.inputDataInternal("lokasiInternal", nilaiL);
438 |             }
439 |         }
440 |
441 |         if (bUsahaCBInternal.isSelected()) {
442 |             InputDataHandler.jmlChecklist();
443 |             if (nilaiUsahaInternal.getText().equals("")) {
444 |                 InputDataHandler.inputDataInternal("usahaInternal", null);
445 |                 checker = false; // false karena nilainya null
446 |             } else {
447 |                 nilaiUsaha = Double.parseDouble(nilaiUsahaInternal.getText()) / 100.0;
448 |                 String nilaiUs = Double.toString(nilaiUsaha);
449 |                 // System.out.println(nilaiU);
450 |                 InputDataHandler.inputDataInternal("usahaInternal", nilaiUs);
451 |             }
452 |         }
453 |         if (InputDataHandler.getKetetangaan() == 0) {
454 |             checker = false;
455 |         }
456 |         double umur = 0.0;
457 |         double pendidikan = 0.0;
458 |         double level = 0.0;
459 |         double pendapatan = 0.0;
460 |         double jenisKelamin = 0.0;
461 |         double lokasi = 0.0;
462 |         double usaha = 0.0;
463 |
464 |         double[] kumpulanBobot = new double[InputDataHandler.getKetetangaan()];
465 |         int m = 0;
466 |         // kalau umur dichecklist, dimasukkan ke variabel umur
467 |         if (InputDataHandler.checkKey("umurInternal")) {
468 |             umur = Double.parseDouble(InputDataHandler.getValue("umurInternal"));
469 |             kumpulanBobot[m] = umur;
470 |             m++;
471 |         }
472 |         if (InputDataHandler.checkKey("pendidikanInternal")) {
473 |             pendidikan = Double.parseDouble(InputDataHandler.getValue("pendidikanInternal"));
474 |             kumpulanBobot[m] = pendidikan;
475 |             m++;
476 |         }
477 |         if (InputDataHandler.checkKey("pendapatanInternal")) {
478 |             pendapatan = Double.parseDouble(InputDataHandler.getValue("pendapatanInternal"));
479 |             kumpulanBobot[m] = pendapatan;
480 |             m++;
481 |         }
482 |         if (InputDataHandler.checkKey("levelInternal")) {
483 |             level = Double.parseDouble(InputDataHandler.getValue("levelInternal"));
484 |             kumpulanBobot[m] = level;
485 |             m++;
486 |         }
487 |         if (InputDataHandler.checkKey("jenisKelaminInternal")) {
488 |             jenisKelamin = Double.parseDouble(InputDataHandler.getValue("jenisKelaminInternal"));
489 |             kumpulanBobot[m] = jenisKelamin;
490 |             m++;
491 |         }
492 |         if (InputDataHandler.checkKey("lokasiInternal")) {
493 |             lokasi = Double.parseDouble(InputDataHandler.getValue("lokasiInternal"));
494 |             kumpulanBobot[m] = lokasi;
495 |             m++;
496 |         }
497 |         if (InputDataHandler.checkKey("usahaInternal")) {
498 |             usaha = Double.parseDouble(InputDataHandler.getValue("usahaInternal"));
499 |             kumpulanBobot[m] = usaha;

```

```

500         m++;
501     }
502
503     int totalNilai=0;
504     for (int i = 0; i < kumpulanBobot.length; i++) {
505         totalNilai+=kumpulanBobot[i]*100;
506     }
507
508     if (totalNilai != 100) {
509         JOptionPane.showMessageDialog(null, "The_sum_of_text_fields_must_100!");
510         checker = false;
511     }
512     InputDataHandler.setBobot(kumpulanBobot);
513
514
515
516     if (checker == true) {
517         this.hide();
518         TampilanKondisiKetetangaan kk = new TampilanKondisiKetetangaan();
519         kk.setVisible(true);
520     } else {
521         JOptionPane.showMessageDialog(null, "You_cannot_move_to_the_other_page_because_you_must_fill_text_field_first!");
522     }
523
524 }
525
526 private void umurCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
527     if (umurCBInternal.isSelected()) {
528         nilaiUmurInternal.setEnabled(true);
529     } else {
530         nilaiUmurInternal.setEnabled(false);
531     }
532 }
533
534 private void levelCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
535     if (levelCBInternal.isSelected()) {
536         nilaiLevelInternal.setEnabled(true);
537     } else {
538         nilaiLevelInternal.setEnabled(false);
539     }
540 }
541
542 private void pendidikanCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
543     if (pendidikanCBInternal.isSelected()) {
544         nilaiPendidikanInternal.setEnabled(true);
545     } else {
546         nilaiPendidikanInternal.setEnabled(false);
547     }
548 }
549
550
551 private void pendapatanCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
552     if (pendapatanCBInternal.isSelected()) {
553         nilaiPendapatanInternal.setEnabled(true);
554     } else {
555         nilaiPendapatanInternal.setEnabled(false);
556     }
557 }
558
559
560 private void jenisKelaminCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
561     if (jenisKelaminCBInternal.isSelected()) {
562         nilaiJenisKelaminInternal.setEnabled(true);
563     } else {
564         nilaiJenisKelaminInternal.setEnabled(false);
565     }
566 }
567
568 private void lokasiCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
569     if (lokasiCBInternal.isSelected()) {
570         nilaiLokasiInternal.setEnabled(true);
571     } else {
572         nilaiLokasiInternal.setEnabled(false);
573     }
574 }
575
576
577 private void bUsahaCBInternalMouseClicked(java.awt.event.MouseEvent evt) {
578     if (bUsahaCBInternal.isSelected()) {
579         nilaiUsahaInternal.setEnabled(true);
580     } else {
581         nilaiUsahaInternal.setEnabled(false);
582     }
583 }
584
585 private void nilaiUmurInternalInputMethodTextChanged(java.awt.event.InputMethodEvent evt) {
586 }
587
588
589 private void nilaiUmurInternalMouseClicked(java.awt.event.MouseEvent evt) {
590 }
591
592
593 private void nilaiUmurInternalComponentAdded(java.awt.event.ContainerEvent evt) {
594 }
595
596
597 /**
598  * @param args the command line arguments

```

```

599  */
600  public static void main(String args[]) {
601      /* Set the Nimbus look and feel */
602      //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
603      /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
604       * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
605       */
606      try {
607          for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
608              if ("Nimbus".equals(info.getName())) {
609                  javax.swing.UIManager.setLookAndFeel(info.getClassName());
610                  break;
611              }
612          }
613      } catch (ClassNotFoundException ex) {
614          java.util.logging.Logger.getLogger(TampilanBobotKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null
615              , ex);
616      } catch (InstantiationException ex) {
617          java.util.logging.Logger.getLogger(TampilanBobotKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null
618              , ex);
619      } catch (IllegalAccessException ex) {
620          java.util.logging.Logger.getLogger(TampilanBobotKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null
621              , ex);
622      } catch (javax.swing.UnsupportedLookAndFeelException ex) {
623          java.util.logging.Logger.getLogger(TampilanBobotKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null
624              , ex);
625      }
626      //</editor-fold>
627      //</editor-fold>
628      //</editor-fold>
629      //</editor-fold>
630      /* Create and display the form */
631      java.awt.EventQueue.invokeLater(new Runnable() {
632          public void run() {
633              new TampilanBobotKetetanggaan().setVisible(true);
634          }
635      });
636  }
637
638  // Variables declaration - do not modify
639  private javax.swing.JCheckBox bUsahaCBInternal;
640  private javax.swing.JLabel jLabel1;
641  private javax.swing.JLabel jLabel2;
642  private javax.swing.JLabel jLabel3;
643  private javax.swing.JLabel jLabel4;
644  private javax.swing.JLabel jLabel5;
645  private javax.swing.JLabel jLabel6;
646  private javax.swing.JLabel jLabel7;
647  private javax.swing.JLabel jLabel8;
648  private javax.swing.JLabel jLabel9;
649  private javax.swing.JPanel jPanel1;
650  private javax.swing.JCheckBox jenisKelaminCBInternal;
651  private javax.swing.JCheckBox levelCBInternal;
652  private javax.swing.JCheckBox lokasiCBInternal;
653  public javax.swing.JButton nextButton;
654  private javax.swing.JTextField nilaiJenisKelaminInternal;
655  private javax.swing.JTextField nilaiLevelInternal;
656  private javax.swing.JTextField nilaiLokasiInternal;
657  private javax.swing.JTextField nilaiPendapatanInternal;
658  private javax.swing.JTextField nilaiPendidikanInternal;
659  private javax.swing.JTextField nilaiUmurInternal;
660  private javax.swing.JTextField nilaiUsahaInternal;
661  private javax.swing.JCheckBox pendapatanCBInternal;
662  private javax.swing.JCheckBox pendidikanCBInternal;
663  private javax.swing.JCheckBox umurCBInternal;
664  // End of variables declaration

```

Listing A.8: TampilanKondisiKetetanggaan.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  import javax.swing.ButtonGroup;
9  import javax.swing.JOptionPane;
10 import javax.swing.JPanel;
11
12 /**
13  *
14  * @author Vanessa
15  */
16 public class TampilanKondisiKetetanggaan extends javax.swing.JFrame {
17
18     /**
19      * Creates new form TampilanKondisiKetetanggaan
20      */
21     JPanel[] kumpulanJPanel;
22
23     //int jmlCheckListInternal = 0;
24     //double[] bobot;
25     public TampilanKondisiKetetanggaan() {
26         initComponents();
27     }

```

```

28     kumpulanJPanel = new JPanel[]{jUmur, jLevel, jPendidikan, jPendapatan, jJenisKelamin, jLokasi, jbidangUsaha};
29     for (int i = 0; i < kumpulanJPanel.length; i++) {
30         kumpulanJPanel[i].setVisible(false);
31         kumpulanJPanel[i].setLocation(29, 103); //ditumpuk di jUmur
32     }
33     int i = 0;
34     if (InputDataHandler.checkKey("umurInternal")) {
35         kumpulanJPanel[0].setVisible(true);
36     }
37     if (InputDataHandler.checkKey("levelInternal")) {
38         kumpulanJPanel[1].setVisible(true);
39     }
40     if (InputDataHandler.checkKey("pendidikanInternal")) {
41         kumpulanJPanel[2].setVisible(true);
42     }
43     if (InputDataHandler.checkKey("pendapatanInternal")) {
44         kumpulanJPanel[3].setVisible(true);
45     }
46     if (InputDataHandler.checkKey("jenisKelaminInternal")) {
47         kumpulanJPanel[4].setVisible(true);
48     }
49     if (InputDataHandler.checkKey("lokasiInternal")) {
50         kumpulanJPanel[5].setVisible(true);
51     }
52     if (InputDataHandler.checkKey("usahaInternal")) {
53         kumpulanJPanel[6].setVisible(true);
54     }
55
56     ButtonGroup group1 = new ButtonGroup();
57     group1.add(umurLbhdR);
58     group1.add(umurSmDgn);
59     group1.add(umurKrgDr);
60
61     ButtonGroup group2 = new ButtonGroup();
62     group2.add(levelLbhdR);
63     group2.add(levelSmDgn);
64     group2.add(levelKrgDr);
65
66     ButtonGroup group3 = new ButtonGroup();
67     group3.add(pendapatanLbhdR);
68     group3.add(pendapatanSmDgn);
69     group3.add(pendapatanKrgDr);
70
71     ButtonGroup group4 = new ButtonGroup();
72     group4.add(pendidikanLbhdR);
73     group4.add(pendidikanSmDgn);
74     group4.add(pendidikanKrgDr);
75
76 }
77
78 /**
79  * This method is called from within the constructor to initialize the form.
80  * WARNING: Do NOT modify this code. The content of this method is always
81  * regenerated by the Form Editor.
82  */
83 @SuppressWarnings("unchecked")
84 // <editor-fold defaultstate="collapsed" desc="Generated Code">
85 private void initComponents() {
86
87     jPanel1 = new javax.swing.JPanel();
88     jLabel1 = new javax.swing.JLabel();
89     jLabel2 = new javax.swing.JLabel();
90     jLabel3 = new javax.swing.JLabel();
91     jUmur = new javax.swing.JPanel();
92     umurKrgDr = new javax.swing.JRadioButton();
93     umurSmDgn = new javax.swing.JRadioButton();
94     umurLbhdR = new javax.swing.JRadioButton();
95     jLabel4 = new javax.swing.JLabel();
96     jLevel = new javax.swing.JPanel();
97     levelKrgDr = new javax.swing.JRadioButton();
98     levelSmDgn = new javax.swing.JRadioButton();
99     levelLbhdR = new javax.swing.JRadioButton();
100    jLabel5 = new javax.swing.JLabel();
101    jPendidikan = new javax.swing.JPanel();
102    pendidikanKrgDr = new javax.swing.JRadioButton();
103    pendidikanSmDgn = new javax.swing.JRadioButton();
104    pendidikanLbhdR = new javax.swing.JRadioButton();
105    jLabel6 = new javax.swing.JLabel();
106    jPendapatan = new javax.swing.JPanel();
107    pendapatanKrgDr = new javax.swing.JRadioButton();
108    pendapatanSmDgn = new javax.swing.JRadioButton();
109    pendapatanLbhdR = new javax.swing.JRadioButton();
110    jLabel7 = new javax.swing.JLabel();
111    jPanel2 = new javax.swing.JPanel();
112    jJenisKelamin = new javax.swing.JPanel();
113    jLabel8 = new javax.swing.JLabel();
114    jbidangUsaha = new javax.swing.JPanel();
115    jLabel10 = new javax.swing.JLabel();
116    jLokasi = new javax.swing.JPanel();
117    jLabel9 = new javax.swing.JLabel();
118    jPendapatan2 = new javax.swing.JPanel();
119    pendapatanCBNeg2 = new javax.swing.JCheckBox();
120    pendapatanKrgDr2 = new javax.swing.JRadioButton();
121    pendapatanSmDgn2 = new javax.swing.JRadioButton();
122    pendapatanLbhdR2 = new javax.swing.JRadioButton();
123    jPanel4 = new javax.swing.JPanel();
124    backButton = new javax.swing.JButton();
125    nextButton = new javax.swing.JButton();
126

```

```

127 | setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
128 |
129 | jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
130 | jLabel1.setText("SIMULATOR_ECA");
131 |
132 | jLabel2.setText("Parameter_Setting_Kondisi_Ketetanggaan_");
133 |
134 | jLabel3.setText("Berdasarkan_Relasinya");
135 |
136 | jUmur.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
137 |
138 | umurKrgDr.setText("<=");
139 |
140 | umurSmDgn.setText("=");
141 |
142 | umurLbhDr.setText(">=");
143 |
144 | jLabel4.setText("UMUR");
145 |
146 | javax.swing.GroupLayout jUmurLayout = new javax.swing.GroupLayout(jUmur);
147 | jUmur.setLayout(jUmurLayout);
148 | jUmurLayout.setHorizontalGroup(
149 |     jUmurLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
150 |         .addGroup(jUmurLayout.createSequentialGroup()
151 |             .addContainerGap()
152 |             .addComponent(jLabel4)
153 |             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
154 |             .addComponent(umurKrgDr)
155 |             .addGap(26, 26, 26)
156 |             .addComponent(umurSmDgn)
157 |             .addGap(18, 18, 18)
158 |             .addComponent(umurLbhDr)
159 |             .addContainerGap()
160 |         );
161 | jUmurLayout.setVerticalGroup(
162 |     jUmurLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
163 |         .addGroup(jUmurLayout.createSequentialGroup()
164 |             .addContainerGap()
165 |             .addGroup(jUmurLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
166 |                 .addComponent(umurKrgDr)
167 |                 .addComponent(umurSmDgn)
168 |                 .addComponent(umurLbhDr)
169 |                 .addComponent(jLabel4)
170 |             )
171 |             .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
172 |         );
173 | jLevel.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
174 |
175 | levelKrgDr.setText("<=");
176 |
177 | levelSmDgn.setText("=");
178 | levelSmDgn.addActionListener(new java.awt.event.ActionListener() {
179 |     public void actionPerformed(java.awt.event.ActionEvent evt) {
180 |         levelSmDgnActionPerformed(evt);
181 |     }
182 | });
183 |
184 | levelLbhDr.setText(">=");
185 | levelLbhDr.addActionListener(new java.awt.event.ActionListener() {
186 |     public void actionPerformed(java.awt.event.ActionEvent evt) {
187 |         levelLbhDrActionPerformed(evt);
188 |     }
189 | });
190 |
191 | jLabel5.setText("LEVEL");
192 |
193 | javax.swing.GroupLayout jLevelLayout = new javax.swing.GroupLayout(jLevel);
194 | jLevel.setLayout(jLevelLayout);
195 | jLevelLayout.setHorizontalGroup(
196 |     jLevelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
197 |         .addGroup(jLevelLayout.createSequentialGroup()
198 |             .addContainerGap()
199 |             .addComponent(jLabel5)
200 |             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
201 |             .addComponent(levelKrgDr)
202 |             .addGap(26, 26, 26)
203 |             .addComponent(levelSmDgn)
204 |             .addGap(18, 18, 18)
205 |             .addComponent(levelLbhDr)
206 |             .addContainerGap()
207 |         );
208 | jLevelLayout.setVerticalGroup(
209 |     jLevelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
210 |         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jLevelLayout.createSequentialGroup()
211 |             .addGap(11, 11, 11)
212 |             .addGroup(jLevelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
213 |                 .addComponent(levelKrgDr)
214 |                 .addComponent(levelSmDgn)
215 |                 .addComponent(levelLbhDr, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
216 |                 .addComponent(jLabel5)
217 |             )
218 |             .addContainerGap()
219 |         );
220 | jPendidikan.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
221 |
222 | pendidikanKrgDr.setText("<=");

```

```

223 pendidikanSmDgn.setText("=");
224
225 pendidikanLbhDr.setText(">=");
226
227 jLabel6.setText("PENDIDIKAN");
228
229 javax.swing.GroupLayout jPendidikanLayout = new javax.swing.GroupLayout(jPendidikan);
230 jPendidikan.setLayout(jPendidikanLayout);
231 jPendidikanLayout.setHorizontalGroup(
232     jPendidikanLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
233     .addGroup(jPendidikanLayout.createSequentialGroup()
234         .add(jPendidikanLayout.createParallelGroup(
235             javax.swing.GroupLayout.Alignment.LEADING)
236             .add(jPendidikanLayout.createSequentialGroup()
237                 .addContainerGap()
238                 .addComponent(jLabel6)
239                 .addGap(145, 145, 145)
240                 .addComponent(pendidikanKrgDr)
241                 .addGap(26, 26, 26)
242                 .addComponent(pendidikanSmDgn)
243                 .addGap(18, 18, Short.MAX_VALUE)
244                 .addComponent(pendidikanLbhDr)
245                 .addContainerGap())
246             .add(jPendidikanLayout.createSequentialGroup()
247                 .addContainerGap()
248                 .addGroup(jPendidikanLayout.createParallelGroup(
249                     javax.swing.GroupLayout.Alignment.BASELINE)
250                     .addComponent(pendidikanKrgDr)
251                     .addComponent(pendidikanSmDgn)
252                     .addComponent(pendidikanLbhDr)
253                     .addComponent(jLabel6))
254                     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
255             ))
256     );
257 jPendapatan.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
258
259 pendapatanKrgDr.setText("<=");
260
261 pendapatanSmDgn.setText("=");
262
263 pendapatanLbhDr.setText(">=");
264
265 jLabel7.setText("PENDAPATAN");
266
267 javax.swing.GroupLayout jPendapatanLayout = new javax.swing.GroupLayout(jPendapatan);
268 jPendapatan.setLayout(jPendapatanLayout);
269 jPendapatanLayout.setHorizontalGroup(
270     jPendapatanLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
271     .addGroup(jPendapatanLayout.createSequentialGroup()
272         .add(jPendapatanLayout.createParallelGroup(
273             javax.swing.GroupLayout.Alignment.LEADING)
274             .add(jPendapatanLayout.createSequentialGroup()
275                 .addContainerGap()
276                 .addComponent(jLabel7)
277                 .addGap(139, 139, 139)
278                 .addComponent(pendapatanKrgDr)
279                 .addGap(27, 27, 27)
280                 .addComponent(pendapatanSmDgn)
281                 .addGap(18, 18, 18)
282                 .addComponent(pendapatanLbhDr, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.
283                     MAX_VALUE)
284                 .addContainerGap())
285             .add(jPendapatanLayout.createSequentialGroup()
286                 .addContainerGap()
287                 .addGroup(jPendapatanLayout.createParallelGroup(
288                     javax.swing.GroupLayout.Alignment.TRAILING)
289                     .add(jPendapatanLayout.createSequentialGroup()
290                         .addContainerGap()
291                         .addComponent(pendapatanKrgDr)
292                         .addComponent(pendapatanSmDgn)
293                         .addComponent(pendapatanLbhDr)
294                         .addComponent(jLabel7))
295                     .addContainerGap())
296             ))
297     );
298 jJenisKelamin.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
299
300 jLabel8.setText("JENIS_KELAMIN");
301
302 javax.swing.GroupLayout jJenisKelaminLayout = new javax.swing.GroupLayout(jJenisKelamin);
303 jJenisKelamin.setLayout(jJenisKelaminLayout);
304 jJenisKelaminLayout.setHorizontalGroup(
305     jJenisKelaminLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
306     .addGroup(jJenisKelaminLayout.createSequentialGroup()
307         .add(jJenisKelaminLayout.createParallelGroup(
308             javax.swing.GroupLayout.Alignment.LEADING)
309             .add(jJenisKelaminLayout.createSequentialGroup()
310                 .addContainerGap()
311                 .addComponent(jLabel8)
312                 .addContainerGap(283, Short.MAX_VALUE))
313             .add(jJenisKelaminLayout.createSequentialGroup()
314                 .addContainerGap()
315                 .addGroup(jJenisKelaminLayout.createParallelGroup(
316                     javax.swing.GroupLayout.Alignment.LEADING)
317                     .add(jJenisKelaminLayout.createSequentialGroup()
318                         .addContainerGap()
319                         .addComponent(pendapatanKrgDr)
320                         .addComponent(pendapatanSmDgn)
321                         .addComponent(pendapatanLbhDr)
322                         .addComponent(jLabel7))
323                     .addContainerGap())
324             ))
325     );
326 jBidangUsaha.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
327
328 jLabel10.setText("BIDANG_USAHA");
329
330 javax.swing.GroupLayout jBidangUsahaLayout = new javax.swing.GroupLayout(jBidangUsaha);
331 jBidangUsaha.setLayout(jBidangUsahaLayout);

```

```

321 | jbidangUsahaLayout.setHorizontalGroup(
322 |     jbidangUsahaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
323 |     .addGroup(jbidangUsahaLayout.createSequentialGroup())
324 |     .addContainerGap()
325 |     .addComponent(jLabel10)
326 |     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
327 | );
328 | jbidangUsahaLayout.setVerticalGroup(
329 |     jbidangUsahaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
330 |     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jbidangUsahaLayout.createSequentialGroup())
331 |     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
332 |     .addComponent(jLabel10)
333 |     .addContainerGap())
334 | );
335 |
336 | jLokasi.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
337 |
338 | jLabel9.setText("LOKASI");
339 |
340 | javax.swing.GroupLayout jLokasiLayout = new javax.swing.GroupLayout(jLokasi);
341 | jLokasi.setLayout(jLokasiLayout);
342 | jLokasiLayout.setHorizontalGroup(
343 |     jLokasiLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
344 |     .addGroup(jLokasiLayout.createSequentialGroup())
345 |     .addContainerGap()
346 |     .addComponent(jLabel9)
347 |     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
348 | );
349 | jLokasiLayout.setVerticalGroup(
350 |     jLokasiLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
351 |     .addGroup(jLokasiLayout.createSequentialGroup())
352 |     .addContainerGap()
353 |     .addComponent(jLabel9)
354 |     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
355 | );
356 |
357 | javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
358 | jPanel2.setLayout(jPanel2Layout);
359 | jPanel2Layout.setHorizontalGroup(
360 |     jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
361 |     .addGroup(jPanel2Layout.createSequentialGroup())
362 |     .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
363 |         .addComponent(jLokasi, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.
364 |             MAX_VALUE)
365 |         .addComponent(jbidangUsaha, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.
366 |             MAX_VALUE)
367 |         .addComponent(jJenisKelamin, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
368 |             javax.swing.GroupLayout.PREFERRED_SIZE))
369 |     .addGap(0, 83, Short.MAX_VALUE))
370 | );
371 | jPanel2Layout.setVerticalGroup(
372 |     jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
373 |     .addGroup(jPanel2Layout.createSequentialGroup())
374 |     .addGap(7, 7, 7)
375 |     .addComponent(jJenisKelamin, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
376 |         swing.GroupLayout.PREFERRED_SIZE)
377 |     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
378 |     .addComponent(jLokasi, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.
379 |         GroupLayout.PREFERRED_SIZE)
380 |     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
381 |     .addComponent(jbidangUsaha, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
382 |         swing.GroupLayout.PREFERRED_SIZE)
383 |     .addContainerGap(45, Short.MAX_VALUE))
384 | );
385 |
386 | jPendapatan2.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));
387 |
388 | pendapatanCBNeg2.setText("PENDAPATAN");
389 | pendapatanCBNeg2.addActionListener(new java.awt.event.ActionListener() {
390 |     public void actionPerformed(java.awt.event.ActionEvent evt) {
391 |         pendapatanCBNeg2ActionPerformed(evt);
392 |     }
393 | });
394 |
395 | pendapatanKrgDr2.setText("<=");
396 |
397 | pendapatanSmDgn2.setText("=");
398 |
399 | pendapatanLbhDr2.setText(">=");
400 |
401 | javax.swing.GroupLayout jPendapatan2Layout = new javax.swing.GroupLayout(jPendapatan2);
402 | jPendapatan2.setLayout(jPendapatan2Layout);
403 | jPendapatan2Layout.setHorizontalGroup(
404 |     jPendapatan2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
405 |     .addGroup(jPendapatan2Layout.createSequentialGroup())
406 |     .addContainerGap()
407 |     .addComponent(pendapatanCBNeg2)
408 |     .addGap(92, 92, 92)
409 |     .addComponent(pendapatanKrgDr2)
410 |     .addGap(27, 27, 27)
411 |     .addComponent(pendapatanSmDgn2)
412 |     .addGap(18, 18, 18)
413 |     .addComponent(pendapatanLbhDr2)
414 |     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
415 | );
416 | jPendapatan2Layout.setVerticalGroup(
417 |     jPendapatan2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
418 |     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPendapatan2Layout.createSequentialGroup())
419 |     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

```


[illegible]


```

502         .addGap(20, 20, 20)
503         .addComponent(backButton)
504         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 212, Short.MAX_VALUE)
505         .addComponent(nextButton)
506         .addContainerGap()
507     );
508     jPanel4Layout.setVerticalGroup(
509     jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
510     .addGroup(jPanel4Layout.createSequentialGroup())
511     .addContainerGap()
512     .addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
513     .addComponent(backButton)
514     .addComponent(nextButton))
515     .addContainerGap(20, Short.MAX_VALUE))
516 );
517
518     javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
519     getContentPane().setLayout(layout);
520     layout.setHorizontalGroup(
521     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
522     .addGroup(layout.createSequentialGroup()
523     .addGap(28, 28, 28)
524     .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, 419, javax.swing.GroupLayout.PREFERRED_SIZE)
525     .addContainerGap(33, Short.MAX_VALUE))
526     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
527     .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
528     .addComponent(jPanel4, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.
529     GroupLayout.PREFERRED_SIZE)
530     .addGap(60, 60, 60))
531 );
532     layout.setVerticalGroup(
533     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
534     .addGroup(layout.createSequentialGroup()
535     .addContainerGap()
536     .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, 461, javax.swing.GroupLayout.PREFERRED_SIZE)
537     .addGap(18, 18, 18)
538     .addComponent(jPanel4, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
539     .addContainerGap())
540 );
541     pack();
542 } // </editor-fold>
543
544 private void levelLbhdActionPerformed(java.awt.event.ActionEvent evt) {
545     // TODO add your handling code here:
546 }
547
548 private void backButtonActionPerformed(java.awt.event.ActionEvent evt) {
549     // TODO add your handling code here:
550     this.hide();
551     TampilanBobotKetetanggaan ki = new TampilanBobotKetetanggaan();
552     ki.setVisible(true);
553 }
554
555 private void nextButtonActionPerformed(java.awt.event.ActionEvent evt) {
556     // TODO add your handling code here:
557 }
558
559 private void levelSmDgnActionPerformed(java.awt.event.ActionEvent evt) {
560     // TODO add your handling code here:
561 }
562
563 private void pendapatanCBNeg2ActionPerformed(java.awt.event.ActionEvent evt) {
564     // TODO add your handling code here:
565 }
566
567 private void nextButtonMouseClicked(java.awt.event.MouseEvent evt) {
568     boolean checker = true;
569     String nilaiRB = "";
570
571     double[] bobot = new double[InputDataHandler.getKetetanggaan()];
572
573     // set relasi 0 kalau sama dengan
574     // set relasi 1 kalau kurang dari sama dengan
575     // set relasi 2 kalau lebih dari sama dengan
576     int n = 0;
577     int[] kumpulanNilaiRelasi = new int[InputDataHandler.getKetetanggaan()];
578
579     //umur
580     if (InputDataHandler.checkKey("umurInternal")) {
581         if (umurLbhd.isSelected()) {
582             InputDataHandler.inputDataKetetanggaan("umurLbhd", umurLbhd.getText());
583             nilaiRB = InputDataHandler.getValue("umurLbhd");
584             if (nilaiRB.equals(">=")) {
585                 nilaiRB = "2";
586             }
587             kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
588             n++;
589             checker = true;
590         } else {
591             if (umurSmDgn.isSelected()) {
592                 InputDataHandler.inputDataKetetanggaan("umurSmDgn", umurSmDgn.getText());
593                 nilaiRB = InputDataHandler.getValue("umurSmDgn");
594                 if (nilaiRB.equals("=")) {
595                     nilaiRB = "0";
596                 }
597             }
598             kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);

```

```

599         n++;
600         checker = true;
601     } else {
602         if (umurKrgDr.isSelected()) {
603             InputDataHandler.inputDataKetetanggaan("umurKurangDari", umurKrgDr.getText());
604             nilaiRB = InputDataHandler.getValue("umurKurangDari");
605             if (nilaiRB.equals("<=")) {
606                 nilaiRB = "1";
607             }
608             kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
609             n++;
610             checker = true;
611         }
612     }
613 }
614
615 if (!umurLbhDr.isSelected()) {
616     if (!umurSmDgn.isSelected()) {
617         if (!umurKrgDr.isSelected()) {
618             checker = false;
619         }
620     }
621 }
622 }
623
624 // level
625 if (InputDataHandler.checkKey("levelInternal")) {
626     if (levelLbhDr.isSelected()) {
627         InputDataHandler.inputDataKetetanggaan("levelLbhDr", levelLbhDr.getText());
628         nilaiRB = InputDataHandler.getValue("levelLbhDr");
629         if (nilaiRB.equals(">=")) {
630             nilaiRB = "2";
631         }
632         kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
633         n++;
634         checker = true;
635     } else {
636         if (levelSmDgn.isSelected()) {
637             InputDataHandler.inputDataKetetanggaan("levelSmDgn", levelSmDgn.getText());
638             nilaiRB = InputDataHandler.getValue("levelSmDgn");
639             if (nilaiRB.equals("=")) {
640                 nilaiRB = "0";
641             }
642             kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
643             n++;
644             checker = true;
645         } else {
646             if (levelKrgDr.isSelected()) {
647                 InputDataHandler.inputDataKetetanggaan("levelKurangDari", levelKrgDr.getText());
648                 nilaiRB = InputDataHandler.getValue("levelKurangDari");
649                 if (nilaiRB.equals("<=")) {
650                     nilaiRB = "1";
651                 }
652                 kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
653                 n++;
654                 checker = true;
655             }
656         }
657     }
658
659     if (!levelLbhDr.isSelected()) {
660         if (!levelSmDgn.isSelected()) {
661             if (!levelKrgDr.isSelected()) {
662                 checker = false;
663             }
664         }
665     }
666 }
667
668 //pendidikan
669 if (InputDataHandler.checkKey("pendidikanInternal")) {
670     if (pendidikanLbhDr.isSelected()) {
671         InputDataHandler.inputDataKetetanggaan("pendidikanLbhDr", pendidikanLbhDr.getText());
672         nilaiRB = InputDataHandler.getValue("pendidikanLbhDr");
673         if (nilaiRB.equals(">=")) {
674             nilaiRB = "2";
675         }
676         kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
677         n++;
678         checker = true;
679     } else {
680         if (pendidikanSmDgn.isSelected()) {
681             InputDataHandler.inputDataKetetanggaan("pendidikanSmDgn", pendidikanSmDgn.getText());
682             nilaiRB = InputDataHandler.getValue("pendidikanSmDgn");
683             if (nilaiRB.equals("=")) {
684                 nilaiRB = "0";
685             }
686             kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
687             n++;
688             checker = true;
689         } else {
690             if (pendidikanKrgDr.isSelected()) {
691                 InputDataHandler.inputDataKetetanggaan("pendidikanKurangDari", pendidikanKrgDr.getText());
692                 nilaiRB = InputDataHandler.getValue("pendidikanKurangDari");
693                 if (nilaiRB.equals("<=")) {
694                     nilaiRB = "1";
695                 }
696                 kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
697                 n++;

```

```

698         checker = true;
699     }
700 }
701 }
702 if (!pendidikanLbhDr.isSelected()) {
703     if (!pendidikanSmDgn.isSelected()) {
704         if (!pendidikanKrgDr.isSelected()) {
705             checker = false;
706         }
707     }
708 }
709 }
710
711 // pendapatan
712 if (InputDataHandler.checkKey("pendapatanInternal")) {
713     if (pendapatanLbhDr.isSelected()) {
714         InputDataHandler.inputDataKetetanggaan("pendapatanLbhDr", pendapatanLbhDr.getText());
715         nilaiRB = InputDataHandler.getValue("pendapatanLbhDr");
716         if (nilaiRB.equals(">=")) {
717             nilaiRB = "2";
718         }
719         kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
720         n++;
721         checker = true;
722     } else {
723         if (pendapatanSmDgn.isSelected()) {
724             InputDataHandler.inputDataKetetanggaan("pendapatanSmDgn", pendapatanSmDgn.getText());
725             nilaiRB = InputDataHandler.getValue("pendapatanSmDgn");
726             if (nilaiRB.equals("=")) {
727                 nilaiRB = "0";
728             }
729             kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
730             n++;
731             checker = true;
732         } else {
733             if (pendapatanKrgDr.isSelected()) {
734                 InputDataHandler.inputDataKetetanggaan("pendapatanKurangDari", pendapatanKrgDr.getText());
735                 nilaiRB = InputDataHandler.getValue("pendapatanKurangDari");
736                 if (nilaiRB.equals("<=")) {
737                     nilaiRB = "1";
738                 }
739                 kumpulanNilaiRelasi[n] = Integer.parseInt(nilaiRB);
740                 n++;
741                 checker = true;
742             }
743         }
744     }
745     if (!pendapatanLbhDr.isSelected()) {
746         if (!pendapatanSmDgn.isSelected()) {
747             if (!pendapatanKrgDr.isSelected()) {
748                 checker = false;
749             }
750         }
751     }
752 }
753
754 InputDataHandler.setRelation(kumpulanNilaiRelasi);
755 if (checker == true) {
756     this.hide();
757     TampilanKondisiEksternal ke = new TampilanKondisiEksternal();
758     ke.setVisible(true);
759 } else {
760     JOptionPane.showMessageDialog(null, "You_cannot_move_to_the_other_page_because_you_must_fill_radio_button_first!");
761 }
762 }
763
764 /**
765  * @param args the command line arguments
766  */
767 public static void main(String args[]) {
768     /* Set the Nimbus look and feel */
769     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
770     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
771      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
772      */
773     try {
774         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
775             if ("Nimbus".equals(info.getName())) {
776                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
777                 break;
778             }
779         }
780     } catch (ClassNotFoundException ex) {
781         java.util.logging.Logger.getLogger(TampilanKondisiKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
782     } catch (InstantiationException ex) {
783         java.util.logging.Logger.getLogger(TampilanKondisiKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
784     } catch (IllegalAccessException ex) {
785         java.util.logging.Logger.getLogger(TampilanKondisiKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
786     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
787         java.util.logging.Logger.getLogger(TampilanKondisiKetetanggaan.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
788     }
789     //</editor-fold>
790
791     /* Create and display the form */
792     java.awt.EventQueue.invokeLater(new Runnable() {

```

```

793         public void run() {
794             new TampilanKondisiKetetanggaan().setVisible(true);
795         }
796     });
797 }
798
799 // Variables declaration - do not modify
800 private javax.swing.JButton backButton;
801 private javax.swing.JPanel jJenisKelamin;
802 private javax.swing.JLabel jLabel1;
803 private javax.swing.JLabel jLabel10;
804 private javax.swing.JLabel jLabel2;
805 private javax.swing.JLabel jLabel3;
806 private javax.swing.JLabel jLabel4;
807 private javax.swing.JLabel jLabel5;
808 private javax.swing.JLabel jLabel6;
809 private javax.swing.JLabel jLabel7;
810 private javax.swing.JLabel jLabel8;
811 private javax.swing.JLabel jLabel9;
812 private javax.swing.JPanel jLevel;
813 private javax.swing.JPanel jLokasi;
814 private javax.swing.JPanel jPanel1;
815 private javax.swing.JPanel jPanel2;
816 private javax.swing.JPanel jPanel4;
817 private javax.swing.JPanel jPendapatan;
818 private javax.swing.JPanel jPendapatan2;
819 private javax.swing.JPanel jPendidikan;
820 private javax.swing.JPanel jUmur;
821 private javax.swing.JPanel jbidangUsaha;
822 private javax.swing.JRadioButton levelKrgDr;
823 private javax.swing.JRadioButton levelLbhDr;
824 private javax.swing.JRadioButton levelSmDgn;
825 public javax.swing.JButton nextButton;
826 private javax.swing.JCheckBox pendapatanCBNeg2;
827 private javax.swing.JRadioButton pendapatanKrgDr;
828 private javax.swing.JRadioButton pendapatanKrgDr2;
829 private javax.swing.JRadioButton pendapatanLbhDr;
830 private javax.swing.JRadioButton pendapatanLbhDr2;
831 private javax.swing.JRadioButton pendapatanSmDgn;
832 private javax.swing.JRadioButton pendapatanSmDgn2;
833 private javax.swing.JRadioButton pendidikanKrgDr;
834 private javax.swing.JRadioButton pendidikanLbhDr;
835 private javax.swing.JRadioButton pendidikanSmDgn;
836 private javax.swing.JRadioButton umurKrgDr;
837 private javax.swing.JRadioButton umurLbhDr;
838 private javax.swing.JRadioButton umurSmDgn;
839 // End of variables declaration
840 }

```

Listing A.9: TampilanKondisiEksternal.java

```

1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package ecasimulatorjframe;
7
8  import javax.swing.JOptionPane;
9
10 /**
11  *
12  * @author Vanessa
13  */
14 public class TampilanKondisiEksternal extends javax.swing.JFrame {
15
16     /**
17      * Creates new form TampilanKondisiEksternal
18      */
19     double[] bobotPF;
20     PublicFactor pf;
21
22     public TampilanKondisiEksternal() {
23         initComponents();
24     }
25
26     /**
27      * This method is called from within the constructor to initialize the form.
28      * WARNING: Do NOT modify this code. The content of this method is always
29      * regenerated by the Form Editor.
30      */
31     @SuppressWarnings("unchecked")
32     // <editor-fold defaultstate="collapsed" desc="Generated Code">
33     private void initComponents() {
34
35         jPanel1 = new javax.swing.JPanel();
36         jLabel1 = new javax.swing.JLabel();
37         jLabel2 = new javax.swing.JLabel();
38         jLabel3 = new javax.swing.JLabel();
39         jLabel4 = new javax.swing.JLabel();
40         jLabel5 = new javax.swing.JLabel();
41         jLabel6 = new javax.swing.JLabel();
42         jLabel7 = new javax.swing.JLabel();
43         nilaiPP = new javax.swing.JTextField();
44         nilaiDP = new javax.swing.JTextField();
45         nilaiNSB = new javax.swing.JTextField();
46         nilaiIFA = new javax.swing.JTextField();
47         nextButton = new javax.swing.JButton();

```

```

48 |     jLabel8 = new javax.swing.JLabel();
49 |     jLabel9 = new javax.swing.JLabel();
50 |     jLabel10 = new javax.swing.JLabel();
51 |     jLabel11 = new javax.swing.JLabel();
52 |     backButton = new javax.swing.JButton();
53 |     jLabel12 = new javax.swing.JLabel();
54 |     nilaiKeterbukaanPasar = new javax.swing.JTextField();
55 |     jLabel13 = new javax.swing.JLabel();
56 |     jLabel14 = new javax.swing.JLabel();
57 |     jLabel15 = new javax.swing.JLabel();
58 |     nilaiInfrastrukturKomersial = new javax.swing.JTextField();
59 |     jLabel16 = new javax.swing.JLabel();
60 |     jLabel17 = new javax.swing.JLabel();
61 |     nilaiTransferPenelitian = new javax.swing.JTextField();
62 |     jLabel18 = new javax.swing.JLabel();
63 |     jLabel19 = new javax.swing.JLabel();
64 |     nilaiPendidikanSMK = new javax.swing.JTextField();
65 |     jLabel20 = new javax.swing.JLabel();
66 |     jLabel21 = new javax.swing.JLabel();
67 |     nilaiPendidikanSDSMP = new javax.swing.JTextField();
68 |     jLabel22 = new javax.swing.JLabel();
69 |     jLabel23 = new javax.swing.JLabel();
70 |     nilaiKPPajak = new javax.swing.JTextField();
71 |     jLabel24 = new javax.swing.JLabel();
72 |     jLabel25 = new javax.swing.JLabel();
73 |     nilaiKPEkonomi = new javax.swing.JTextField();
74 |     jLabel26 = new javax.swing.JLabel();
75 |     jLabel27 = new javax.swing.JLabel();
76 |     nilaiKeuanganKewirausahaan = new javax.swing.JTextField();
77 |     jLabel28 = new javax.swing.JLabel();
78 |
79 |     setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
80 |
81 |     jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
82 |     jLabel1.setText("SIMULATOR_ECA");
83 |
84 |     jLabel2.setText("Parameter_Setting_Kondisi_Eksternal_");
85 |
86 |     jLabel3.setText("Faktor_Publik_");
87 |
88 |     jLabel4.setText("Program_Pemerintah_");
89 |
90 |     jLabel5.setText("Dinamika_Pasar_");
91 |
92 |     jLabel6.setText("Norma,_Sosial_dan_Budaya_");
93 |
94 |     jLabel7.setText("Infrastruktur_Fisik_dan_Akses_Layanan_");
95 |
96 |     nilaiPP.addActionListener(new java.awt.event.ActionListener() {
97 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
98 |             nilaiPPActionPerformed(evt);
99 |         }
100 |    });
101 |
102 |     nilaiDP.addActionListener(new java.awt.event.ActionListener() {
103 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
104 |             nilaiDPActionPerformed(evt);
105 |         }
106 |    });
107 |
108 |     nilaiNSB.addActionListener(new java.awt.event.ActionListener() {
109 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
110 |             nilaiNSBActionPerformed(evt);
111 |         }
112 |    });
113 |
114 |     nilaiIFA.addActionListener(new java.awt.event.ActionListener() {
115 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
116 |             nilaiIFAActionPerformed(evt);
117 |         }
118 |    });
119 |
120 |     nextButton.setText("NEXT");
121 |     nextButton.addMouseListener(new java.awt.event.MouseAdapter() {
122 |         public void mouseClicked(java.awt.event.MouseEvent evt) {
123 |             nextButtonMouseClicked(evt);
124 |         }
125 |    });
126 |     nextButton.addActionListener(new java.awt.event.ActionListener() {
127 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
128 |             nextButtonActionPerformed(evt);
129 |         }
130 |    });
131 |
132 |     jLabel8.setText("%");
133 |
134 |     jLabel9.setText("%");
135 |
136 |     jLabel10.setText("%");
137 |
138 |     jLabel11.setText("%");
139 |
140 |     backButton.setText("BACK");
141 |     backButton.addActionListener(new java.awt.event.ActionListener() {
142 |         public void actionPerformed(java.awt.event.ActionEvent evt) {
143 |             backButtonActionPerformed(evt);
144 |         }
145 |    });
146 |

```

```

147     jLabel12.setText("Keterbukaan_Pasar_");
148
149     nilaiKeterbukaanPasar.addActionListener(new java.awt.event.ActionListener() {
150         public void actionPerformed(java.awt.event.ActionEvent evt) {
151             nilaiKeterbukaanPasarActionPerformed(evt);
152         }
153     });
154
155     jLabel13.setText("%");
156
157     jLabel15.setText("Infrastruktur_Komersial_dan_Legal_");
158
159     nilaiInfrastrukturKomersial.addActionListener(new java.awt.event.ActionListener() {
160         public void actionPerformed(java.awt.event.ActionEvent evt) {
161             nilaiInfrastrukturKomersialActionPerformed(evt);
162         }
163     });
164
165     jLabel16.setText("%");
166
167     jLabel17.setText("Transfer_Penelitian_dan_Pengembangan_");
168
169     jLabel18.setText("%");
170
171     jLabel19.setText("Pendidikan_Kewirausahaan_pada_SMK,_Profesional_dan_Universitas_");
172
173     jLabel20.setText("%");
174
175     jLabel21.setText("Pendidikan_Kewirausahaan_pada_SD_dan_SMP_");
176
177     nilaiPendidikanSDSMP.addActionListener(new java.awt.event.ActionListener() {
178         public void actionPerformed(java.awt.event.ActionEvent evt) {
179             nilaiPendidikanSDSMPActionPerformed(evt);
180         }
181     });
182
183     jLabel22.setText("%");
184
185     jLabel23.setText("Kebijakan_Pemerintah_terkait_Pajak_");
186
187     jLabel24.setText("%");
188
189     jLabel25.setText("Kebijakan_Pemerintah_terkait_Ekonomi_");
190
191     jLabel26.setText("%");
192
193     jLabel27.setText("Keuangan_terkait_Kewirausahaan_");
194
195     jLabel28.setText("%");
196
197     javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
198     jPanel1.setLayout(jPanel1Layout);
199     jPanel1Layout.setHorizontalGroup(
200         jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
201             .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()
202                 .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
203                 .addComponent(nextButton)
204                 .addGap(29, 29, 29))
205             .addGroup(jPanel1Layout.createSequentialGroup()
206                 .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
207                     .addGroup(jPanel1Layout.createSequentialGroup()
208                         .addGap(34, 34, 34)
209                         .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
210                             .addComponent(jLabel3)
211                             .addGroup(jPanel1Layout.createSequentialGroup()
212                                 .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
213                                     .addComponent(jLabel7)
214                                     .addComponent(backButton))
215                                 .addGap(18, 18, 18)
216                                 .addComponent(nilaiIFA, javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE)
217                                 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
218                                 .addComponent(jLabel11))
219                             .addGroup(jPanel1Layout.createSequentialGroup()
220                                 .addComponent(jLabel6)
221                                 .addGap(18, 18, 18)
222                                 .addComponent(nilaiNSB, javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE)
223                                 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
224                                 .addComponent(jLabel10))
225                             .addGroup(jPanel1Layout.createSequentialGroup()
226                                 .addComponent(jLabel12)
227                                 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
228                                 .addComponent(nilaiKeterbukaanPasar, javax.swing.GroupLayout.PREFERRED_SIZE, 51, javax.swing.GroupLayout.PREFERRED_SIZE)
229                                 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
230                                 .addComponent(jLabel13))
231                             .addComponent(jLabel14)
232                             .addGroup(jPanel1Layout.createSequentialGroup()
233                                 .addComponent(jLabel5)
234                                 .addGap(18, 18, 18)
235                                 .addComponent(nilaiDP, javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE)
236                                 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
237                                 .addComponent(jLabel9))
238                             .addGroup(jPanel1Layout.createSequentialGroup()
239                                 .addComponent(jLabel15)
240                                 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

```

```

241         .addComponent(nilaiInfrastrukturKomersial, javax.swing.GroupLayout.PREFERRED_SIZE, 44, javax.swing
242             GroupLayout.PREFERRED_SIZE)
243         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
244         .addComponent(jLabel16))
245     .addGroup(jPanel1Layout.createSequentialGroup())
246     .addComponent(jLabel17)
247     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
248     .addComponent(nilaiTransferPenelitian, javax.swing.GroupLayout.PREFERRED_SIZE, 42, javax.swing.
249         GroupLayout.PREFERRED_SIZE)
250     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
251     .addComponent(jLabel18))
252     .addGroup(jPanel1Layout.createSequentialGroup())
253     .addComponent(jLabel14)
254     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
255     .addComponent(nilaiPP, javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.
256         PREFERRED_SIZE)
257     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
258     .addComponent(jLabel8))
259     .addGroup(jPanel1Layout.createSequentialGroup())
260     .addComponent(jLabel21)
261     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
262     .addComponent(nilaiPendidikanSDSMP, javax.swing.GroupLayout.PREFERRED_SIZE, 41, javax.swing.
263         GroupLayout.PREFERRED_SIZE)
264     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
265     .addComponent(jLabel22))
266     .addGroup(jPanel1Layout.createSequentialGroup())
267     .addComponent(jLabel19)
268     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
269     .addComponent(nilaiPendidikanSMK, javax.swing.GroupLayout.PREFERRED_SIZE, 41, javax.swing.
270         GroupLayout.PREFERRED_SIZE)
271     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
272     .addComponent(jLabel20))
273     .addGroup(jPanel1Layout.createSequentialGroup())
274     .addComponent(jLabel23)
275     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
276     .addComponent(nilaiKPPajak, javax.swing.GroupLayout.PREFERRED_SIZE, 46, javax.swing.GroupLayout.
277         PREFERRED_SIZE)
278     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
279     .addComponent(jLabel24, javax.swing.GroupLayout.PREFERRED_SIZE, 17, javax.swing.GroupLayout.
280         PREFERRED_SIZE))
281     .addGroup(jPanel1Layout.createSequentialGroup())
282     .addComponent(jLabel25)
283     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
284     .addComponent(nilaiKEkonomi, javax.swing.GroupLayout.PREFERRED_SIZE, 47, javax.swing.GroupLayout.
285         PREFERRED_SIZE)
286     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
287     .addComponent(jLabel26))
288     .addGroup(jPanel1Layout.createSequentialGroup())
289     .addComponent(jLabel27)
290     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
291     .addComponent(nilaiKeuanganKewirausahaan, javax.swing.GroupLayout.PREFERRED_SIZE, 42, javax.swing.
292         GroupLayout.PREFERRED_SIZE)
293     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
294     .addComponent(jLabel28)))
295     .addGroup(jPanel1Layout.createSequentialGroup())
296     .addContainerGap()
297     .addComponent(jLabel2))
298     .addContainerGap(86, Short.MAX_VALUE)
299 );
300 jPanel1Layout.setVerticalGroup(
301     jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
302     .addGroup(jPanel1Layout.createSequentialGroup())
303     .addContainerGap()
304     .addComponent(jLabel1)
305     .addGap(18, 18, 18)
306     .addComponent(jLabel2)
307     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
308     .addComponent(jLabel3)
309     .addGap(18, 18, 18)
310     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
311     .addComponent(jLabel27)
312     .addComponent(nilaiKeuanganKewirausahaan, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
313         DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
314     .addComponent(jLabel28))
315     .addGap(18, 18, 18)
316     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
317     .addComponent(jLabel25)
318     .addComponent(nilaiKEkonomi, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
319         javax.swing.GroupLayout.PREFERRED_SIZE)
320     .addComponent(jLabel26))
321     .addGap(18, 18, 18)
322     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
323     .addComponent(jLabel23)
324     .addComponent(nilaiKPPajak, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
325         javax.swing.GroupLayout.PREFERRED_SIZE)
326     .addComponent(jLabel24))
327     .addGap(18, 18, Short.MAX_VALUE)
328     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
329     .addComponent(jLabel14)
330     .addComponent(nilaiPP, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
331         swing.GroupLayout.PREFERRED_SIZE)
332     .addComponent(jLabel8))
333     .addGap(18, 18, 18)
334     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
335     .addComponent(jLabel21)

```

```

326         .addComponent(nilaiPendidikanSDSMP, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
327             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
328         .addComponent(jLabel22))
329     .addGap(18, 18, 18)
330     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
331         .addComponent(jLabel19)
332         .addComponent(nilaiPendidikanSMK, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE
333             , javax.swing.GroupLayout.PREFERRED_SIZE)
334         .addComponent(jLabel20))
335     .addGap(18, 18, 18)
336     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
337         .addComponent(jLabel17)
338         .addComponent(nilaiTransferPenelitian, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
339             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
340         .addComponent(jLabel18))
341     .addGap(16, 16, 16)
342     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
343         .addComponent(jLabel15, javax.swing.GroupLayout.Alignment.TRAILING)
344         .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
345             .addComponent(nilaiInfrastrukturKomersial, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
346                 DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
347             .addComponent(jLabel16)))
348     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
349     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
350         .addComponent(jLabel5)
351         .addComponent(nilaiDP, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
352             swing.GroupLayout.PREFERRED_SIZE)
353         .addComponent(jLabel9))
354     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
355     .addComponent(jLabel14)
356     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
357     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
358         .addComponent(jLabel12)
359         .addComponent(nilaiKeterbukaanPasar, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.
360             DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
361         .addComponent(jLabel13))
362     .addGap(18, 18, 18)
363     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
364         .addComponent(jLabel7)
365         .addComponent(nilaiIFA, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
366             swing.GroupLayout.PREFERRED_SIZE)
367         .addComponent(jLabel11))
368     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
369     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
370         .addComponent(jLabel6)
371         .addComponent(nilaiNSB, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
372             swing.GroupLayout.PREFERRED_SIZE)
373         .addComponent(jLabel10))
374     .addGap(35, 35, 35)
375     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
376         .addComponent(nextButton)
377         .addComponent(backButton))
378     .addContainerGap()
379 );
380
381 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
382 getContentPane().setLayout(layout);
383 layout.setHorizontalGroup(
384     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
385     .addGroup(layout.createSequentialGroup()
386         .addContainerGap()
387         .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
388         .addContainerGap())
389 );
390 layout.setVerticalGroup(
391     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
392     .addGroup(layout.createSequentialGroup()
393         .addContainerGap()
394         .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
395         .addContainerGap())
396 );
397 pack();
398 }
399
400 private void backButtonActionPerformed(java.awt.event.ActionEvent evt) {
401     // TODO add your handling code here:
402     this.hide();
403     TampilanKondisiKetetanggaan ki = new TampilanKondisiKetetanggaan();
404     ki.setVisible(true);
405 }
406
407 private void nextButtonActionPerformed(java.awt.event.ActionEvent evt) {
408     // TODO add your handling code here:
409 }
410
411 private void nilaiPPActionPerformed(java.awt.event.ActionEvent evt) {
412 }
413
414 private void nilaiNSBActionPerformed(java.awt.event.ActionEvent evt) {

```



```

415 // TODO add your handling code here:
416
417 }
418
419 private void nilaiIFAActionPerformed(java.awt.event.ActionEvent evt) {
420
421 }
422
423 private void nextButtonMouseClicked(java.awt.event.MouseEvent evt) {
424     boolean checker = true;
425     // String isiNilaiDP;
426     // String isiNilaiIFA;
427     // String isiNilaiNSB;
428     // String isiNilaiPP;
429     double isiNilaiKK = 0.0;
430     double isiNilaiIK = 0.0;
431     double isiNilaiKPE = 0.0;
432     double isiNilaiKPP = 0.0;
433     double isiNilaiKP = 0.0;
434     double isiNilaiPSS = 0.0;
435     double isiNilaiPS = 0.0;
436     double isiNilaiTP = 0.0;
437     double[] kumpulanNilaiPF = new double[12];
438     double isiNilaiDP = 0.0;
439     double isiNilaiIFA = 0.0;
440     double isiNilaiNSB = 0.0;
441     double isiNilaiPP = 0.0;
442     if (nilaiDP.getText().equals("")) {
443         InputDataHandler.inputDataEksternal("dinamikaPasar", null);
444         checker = false;
445     }
446     else if (nilaiIFA.getText().equals("")) {
447         InputDataHandler.inputDataEksternal("InfrastrukturListrik", null);
448         checker = false;
449     }
450     else if (nilaiNSB.getText().equals("")) {
451         InputDataHandler.inputDataEksternal("NormaSosialBudaya", null);
452         checker = false;
453     }
454     else if (nilaiPP.getText().equals("")) {
455         InputDataHandler.inputDataEksternal("ProgramPemerintah", null);
456         checker = false;
457     }
458     else if (nilaiInfrastrukturKomersial.getText().equals("")) {
459         InputDataHandler.inputDataEksternal("InfrastrukturKomersial", null);
460         checker = false;
461     }
462     else if (nilaiKPEkonomi.getText().equals("")) {
463         InputDataHandler.inputDataEksternal("NilaiKPEkonomi", null);
464         checker = false;
465     }
466     else if (nilaiKPPajak.getText().equals("")) {
467         InputDataHandler.inputDataEksternal("nilaiKPPajak", null);
468         checker = false;
469     }
470     else if (nilaiKeterbukaanPasar.getText().equals("")) {
471         InputDataHandler.inputDataEksternal("nilaiKeterbukaanPasar", null);
472         checker = false;
473     }
474     else if (nilaiKeuanganKewirausahaan.getText().equals("")) {
475         InputDataHandler.inputDataEksternal("nilaiKeuanganKewirausahaan", null);
476         checker = false;
477     }
478     else if (nilaiPendidikanSDSMP.getText().equals("")) {
479         InputDataHandler.inputDataEksternal("nilaiPendidikanSDSMP", null);
480         checker = false;
481     }
482     else if (nilaiPendidikanSMK.getText().equals("")) {
483         InputDataHandler.inputDataEksternal("nilaiPendidikanSMK", null);
484         checker = false;
485     }
486     else if (nilaiTransferPenelitian.getText().equals("")) {
487         InputDataHandler.inputDataEksternal("nilaiTransferPenelitian", null);
488         checker = false;
489     }
490     else {
491         if (!nilaiKeuanganKewirausahaan.equals("")) {
492             isiNilaiKK = Double.parseDouble(nilaiKeuanganKewirausahaan.getText()) / 100.0;
493             String nilaiKK = Double.toString(isiNilaiKK);
494             InputDataHandler.inputDataEksternal("nilaiKeuanganKewirausahaan", nilaiKK);
495             kumpulanNilaiPF[0] = Double.parseDouble(InputDataHandler.getValue("nilaiKeuanganKewirausahaan"));
496             if (!nilaiKPEkonomi.getText().equals("")) {
497                 isiNilaiKPE = Double.parseDouble(nilaiKPEkonomi.getText()) / 100.0;
498                 String nilaiKP = Double.toString(isiNilaiKPE);
499                 InputDataHandler.inputDataEksternal("nilaiKPEkonomi", nilaiKP);
500                 kumpulanNilaiPF[1] = Double.parseDouble(InputDataHandler.getValue("nilaiKPEkonomi"));
501                 if (!nilaiKPPajak.getText().equals("")) {
502                     isiNilaiKPP = Double.parseDouble(nilaiKPPajak.getText()) / 100.0;
503                     String nilaiKPP = Double.toString(isiNilaiKPP);
504                     InputDataHandler.inputDataEksternal("nilaiKPPajak", nilaiKPP);
505                     kumpulanNilaiPF[2] = Double.parseDouble(InputDataHandler.getValue("nilaiKPPajak"));
506                     if (!nilaiPP.getText().equals("")) {
507                         isiNilaiPP = Double.parseDouble(nilaiPP.getText()) / 100.0;
508                         String nilaiPP = Double.toString(isiNilaiPP);
509                         InputDataHandler.inputDataEksternal("ProgramPemerintah", nilaiPP);
510                         kumpulanNilaiPF[3] = Double.parseDouble(InputDataHandler.getValue("ProgramPemerintah"));
511                     }
512                     if (!nilaiPendidikanSDSMP.getText().equals("")) {
513                         isiNilaiPSS = Double.parseDouble(nilaiPendidikanSDSMP.getText()) / 100.0;
514                         String nilaiPSS = Double.toString(isiNilaiPSS);
515                         InputDataHandler.inputDataEksternal("nilaiPendidikanSDSMP", nilaiPSS);
516                         kumpulanNilaiPF[4] = Double.parseDouble(InputDataHandler.getValue("nilaiPendidikanSDSMP"));
517                         if (!nilaiPendidikanSMK.getText().equals("")) {
518                             isiNilaiPS = Double.parseDouble(nilaiPendidikanSMK.getText()) / 100.0;
519                             String nilaiPS = Double.toString(isiNilaiPS);
520                             InputDataHandler.inputDataEksternal("nilaiPendidikanSMK", nilaiPS);
521                             kumpulanNilaiPF[5] = Double.parseDouble(InputDataHandler.getValue("nilaiPendidikanSMK"));
522                             if (!nilaiTransferPenelitian.getText().equals("")) {
523                                 isiNilaiTP = Double.parseDouble(nilaiTransferPenelitian.getText()) / 100.0;
524                                 String nilaiTP = Double.toString(isiNilaiTP);

```

```

514 InputDataHandler.inputDataEksternal("nilaiTransferPenelitian", nilaiTP);
515 kumpulanNilaiPF[6] = Double.parseDouble(InputDataHandler.getValue("nilaiTransferPenelitian"));
516 if (!nilaiInfrastrukturKomersial.getText().equals("")) {
517     isiNilaiIK = Double.parseDouble(nilaiInfrastrukturKomersial.getText()) / 100.0;
518     String nilaiIK = Double.toString(isiNilaiIK);
519     InputDataHandler.inputDataEksternal("nilaiInfrastrukturKomersial", nilaiIK);
520     kumpulanNilaiPF[7] = Double.parseDouble(InputDataHandler.getValue("nilaiInfrastrukturKomersial"));
521     if (!nilaiDP.getText().equals("")) {
522         isiNilaiDP = Double.parseDouble(nilaiDP.getText()) / 100.0;
523         String nilaiDP = Double.toString(isiNilaiDP);
524         InputDataHandler.inputDataEksternal("DinamikaPasar", nilaiDP);
525         kumpulanNilaiPF[8] = Double.parseDouble(InputDataHandler.getValue("DinamikaPasar"));
526     }
527     if (!nilaiKeterbukaanPasar.getText().equals("")) {
528         isiNilaiKP = Double.parseDouble(nilaiKeterbukaanPasar.getText()) / 100.0;
529         String nilaiKPas = Double.toString(isiNilaiKP);
530         InputDataHandler.inputDataEksternal("nilaiKeterbukaanPasar", nilaiKPas);
531         kumpulanNilaiPF[9] = Double.parseDouble(InputDataHandler.getValue("nilaiKeterbukaanPasar"));
532     }
533     if (!nilaiIFA.getText().equals("")) {
534         isiNilaiIFA = Double.parseDouble(nilaiIFA.getText()) / 100.0;
535         String nilaiIFA = Double.toString(isiNilaiIFA);
536         InputDataHandler.inputDataEksternal("InfrastrukturListrik", nilaiIFA);
537         kumpulanNilaiPF[10] = Double.parseDouble(InputDataHandler.getValue("InfrastrukturListrik"));
538     }
539     if (!nilaiNSB.getText().equals("")) {
540         isiNilaiNSB = Double.parseDouble(nilaiNSB.getText()) / 100.0;
541         String nilaiNSB = Double.toString(isiNilaiNSB);
542         InputDataHandler.inputDataEksternal("NormaSosialBudaya", nilaiNSB);
543         kumpulanNilaiPF[11] = Double.parseDouble(InputDataHandler.getValue("NormaSosialBudaya"));
544     }
545 }
546 }
547 }
548 }
549 }
550 }
551 }
552 }
553 }
554 }
555 int totalNilai=0;
556 for (int i = 0; i < kumpulanNilaiPF.length; i++) {
557     totalNilai+=kumpulanNilaiPF[i]*100;
558 }
559
560 if (totalNilai != 100) {
561     JOptionPane.showMessageDialog(null, "The_sum_of_text_fields_must_100%!");
562     checker = false;
563 }
564 if (checker == true) {
565     this.hide();
566     TampilanDataWirusaha ks = new TampilanDataWirusaha();
567     ks.setVisible(true);
568 } else {
569     JOptionPane.showMessageDialog(null, "You_must_fill_the_text_field!");
570 }
571 InputDataHandler.setDataEksternal(kumpulanNilaiPF);
572 }
573
574 private void nilaiKeterbukaanPasarActionPerformed(java.awt.event.ActionEvent evt) {
575     // TODO add your handling code here:
576 }
577
578 private void nilaiInfrastrukturKomersialActionPerformed(java.awt.event.ActionEvent evt) {
579     // TODO add your handling code here:
580 }
581
582 private void nilaiPendidikanSDSMPActionPerformed(java.awt.event.ActionEvent evt) {
583     // TODO add your handling code here:
584 }
585
586 /**
587  * @param args the command line arguments
588  */
589 public static void main(String args[]) {
590     /* Set the Nimbus look and feel */
591     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
592     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
593      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
594      */
595     try {
596         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
597             if ("Nimbus".equals(info.getName())) {
598                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
599                 break;
600             }
601         }
602     } catch (ClassNotFoundException ex) {
603         java.util.logging.Logger.getLogger(TampilanKondisiEksternal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
604     } catch (InstantiationException ex) {

```

```

605     java.util.logging.Logger.getLogger(TampilanKondisiEksternal.class.getName()).log(java.util.logging.Level.SEVERE, null,
606         ex);
607     } catch (IllegalAccessException ex) {
608         java.util.logging.Logger.getLogger(TampilanKondisiEksternal.class.getName()).log(java.util.logging.Level.SEVERE, null,
609             ex);
610     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
611         java.util.logging.Logger.getLogger(TampilanKondisiEksternal.class.getName()).log(java.util.logging.Level.SEVERE, null,
612             ex);
613     }
614     //</editor-fold>
615
616     /* Create and display the form */
617     java.awt.EventQueue.invokeLater(new Runnable() {
618         public void run() {
619             new TampilanKondisiEksternal().setVisible(true);
620         }
621     });
622 }
623
624 // Variables declaration - do not modify
625 private javax.swing.JButton backButton;
626 private javax.swing.JLabel jLabel1;
627 private javax.swing.JLabel jLabel10;
628 private javax.swing.JLabel jLabel11;
629 private javax.swing.JLabel jLabel12;
630 private javax.swing.JLabel jLabel13;
631 private javax.swing.JLabel jLabel14;
632 private javax.swing.JLabel jLabel15;
633 private javax.swing.JLabel jLabel16;
634 private javax.swing.JLabel jLabel17;
635 private javax.swing.JLabel jLabel18;
636 private javax.swing.JLabel jLabel19;
637 private javax.swing.JLabel jLabel2;
638 private javax.swing.JLabel jLabel20;
639 private javax.swing.JLabel jLabel21;
640 private javax.swing.JLabel jLabel22;
641 private javax.swing.JLabel jLabel23;
642 private javax.swing.JLabel jLabel24;
643 private javax.swing.JLabel jLabel25;
644 private javax.swing.JLabel jLabel26;
645 private javax.swing.JLabel jLabel27;
646 private javax.swing.JLabel jLabel28;
647 private javax.swing.JLabel jLabel3;
648 private javax.swing.JLabel jLabel4;
649 private javax.swing.JLabel jLabel5;
650 private javax.swing.JLabel jLabel6;
651 private javax.swing.JLabel jLabel7;
652 private javax.swing.JLabel jLabel8;
653 private javax.swing.JLabel jLabel9;
654 private javax.swing.JPanel jPanel1;
655 public javax.swing.JButton nextButton;
656 private javax.swing.JTextField nilaiDP;
657 private javax.swing.JTextField nilaiIFA;
658 private javax.swing.JTextField nilaiInfrastrukturKomersial;
659 private javax.swing.JTextField nilaiKPEkonomi;
660 private javax.swing.JTextField nilaiKPPajak;
661 private javax.swing.JTextField nilaiKeterbukaanPasar;
662 private javax.swing.JTextField nilaiKeuanganKewirausahaan;
663 private javax.swing.JTextField nilaiNSB;
664 private javax.swing.JTextField nilaiPP;
665 private javax.swing.JTextField nilaiPendidikanSDSMP;
666 private javax.swing.JTextField nilaiPendidikanSMK;
667 private javax.swing.JTextField nilaiTransferPenelitian;
668 // End of variables declaration
669 }

```

Listing A.10: TampilanDataWirausaha.java

```

1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package ecasimulatorjframe;
7
8  import java.io.BufferedReader;
9  import java.io.File;
10 import java.io.FileReader;
11 import java.io.IOException;
12 import java.util.logging.Level;
13 import java.util.logging.Logger;
14 import javax.swing.JFileChooser;
15 import javax.swing.JOptionPane;
16 import javax.swing.table.DefaultTableModel;
17
18 /**
19  *
20  * @author Vanessa
21  */
22 public class TampilanDataWirausaha extends javax.swing.JFrame {
23
24     /**
25      * Creates new form TampilanSimulasi
26      */
27     CA ca;
28     public BufferedReader br;
29     private final JFileChooser openFileChooser;
30 }

```

```

31 public TampilanDataWirausaha() {
32     initComponents();
33     openFileChooser = new JFileChooser();
34     openFileChooser.setSelectedFile(new File("D:\\text.txt"));
35 }
36
37 /**
38  * This method is called from within the constructor to initialize the form.
39  * WARNING: Do NOT modify this code. The content of this method is always
40  * regenerated by the Form Editor.
41  */
42 @SuppressWarnings("unchecked")
43 // <editor-fold defaultstate="collapsed" desc="Generated Code">
44 private void initComponents() {
45
46     jPanel1 = new javax.swing.JPanel();
47     jPanel2 = new javax.swing.JPanel();
48     jLabel1 = new javax.swing.JLabel();
49     jLabel2 = new javax.swing.JLabel();
50     openFileButton = new javax.swing.JButton();
51     messageLabel = new javax.swing.JLabel();
52     jScrollPane1 = new javax.swing.JScrollPane();
53     jTable1 = new javax.swing.JTable();
54     nextButton = new javax.swing.JButton();
55     backButton = new javax.swing.JButton();
56
57     javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
58     jPanel1.setLayout(jPanel1Layout);
59     jPanel1Layout.setHorizontalGroup(
60         jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
61             .addGap(0, 100, Short.MAX_VALUE)
62     );
63     jPanel1Layout.setVerticalGroup(
64         jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
65             .addGap(0, 100, Short.MAX_VALUE)
66     );
67
68     setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
69
70     jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
71     jLabel1.setText("SIMULATOR_ECA");
72
73     jLabel2.setText("Data_Simulasi.");
74
75     openFileButton.setText("OPEN_FILE");
76     openFileButton.addMouseListener(new java.awt.event.MouseAdapter() {
77         public void mouseClicked(java.awt.event.MouseEvent evt) {
78             openFileButtonMouseClicked(evt);
79         }
80     });
81     openFileButton.addActionListener(new java.awt.event.ActionListener() {
82         public void actionPerformed(java.awt.event.ActionEvent evt) {
83             openFileButtonActionPerformed(evt);
84         }
85     });
86
87     jTable1.setModel(new javax.swing.table.DefaultTableModel(
88         new Object [][] {
89
90             },
91         new String [] {
92             "Jenis_Kelamin", "Umur", "Usia_Bisnis_(bulan)", "Kategori", "Sub_Kategori", "Pendidikan", "Lokasi", "Pendapatan",
93             "Level", "Point"
94         }
95     ));
96     jTable1.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
97     jScrollPane1.setViewportView(jTable1);
98
99     nextButton.setText("NEXT");
100    nextButton.addMouseListener(new java.awt.event.MouseAdapter() {
101        public void mouseClicked(java.awt.event.MouseEvent evt) {
102            nextButtonMouseClicked(evt);
103        }
104    });
105    nextButton.addActionListener(new java.awt.event.ActionListener() {
106        public void actionPerformed(java.awt.event.ActionEvent evt) {
107            nextButtonActionPerformed(evt);
108        }
109    });
110
111    backButton.setText("BACK");
112    backButton.addActionListener(new java.awt.event.ActionListener() {
113        public void actionPerformed(java.awt.event.ActionEvent evt) {
114            backButtonActionPerformed(evt);
115        }
116    });
117
118    javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
119    jPanel2.setLayout(jPanel2Layout);
120    jPanel2Layout.setHorizontalGroup(
121        jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
122            .addGroup(jPanel2Layout.createSequentialGroup()
123                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
124                    .addContainerGap()
125                    .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
126                        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
127                            .addComponent(jLabel2)
128                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

```

```

129         .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
130             .addComponent(openFileButton)
131             .addGroup(jPanel2Layout.createSequentialGroup()
132                 .addGap(111, 111, 111)
133                 .addComponent(messageLabel, javax.swing.GroupLayout.PREFERRED_SIZE, 122, javax.swing.
134                     GroupLayout.PREFERRED_SIZE)))
135             .addGap(0, 0, Short.MAX_VALUE))
136         .addComponent(jScrollPane1, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.
137             DEFAULT_SIZE, 579, Short.MAX_VALUE)
138         .addContainerGap()
139         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup())
140             .addGap(0, 0, Short.MAX_VALUE)
141             .addComponent(jLabel1)
142             .addGap(235, 235, 235)))
143     .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup())
144         .addGap(48, 48, 48)
145         .addComponent(backButton)
146         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.
147             MAX_VALUE)
148         .addComponent(nextButton)
149         .addGap(53, 53, 53))
150 );
151 jPanel2Layout.setVerticalGroup(
152     jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
153     .addGroup(jPanel2Layout.createSequentialGroup()
154         .addContainerGap()
155         .addComponent(jLabel1)
156         .addGap(23, 23, 23)
157         .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
158             .addComponent(jLabel2)
159             .addComponent(openFileButton)
160             .addComponent(messageLabel))
161         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
162         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 319, javax.swing.GroupLayout.PREFERRED_SIZE)
163         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 65, Short.MAX_VALUE)
164         .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
165             .addComponent(nextButton)
166             .addComponent(backButton))
167         .addGap(31, 31, 31))
168     );
169 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
170 getContentPane().setLayout(layout);
171 layout.setHorizontalGroup(
172     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
173     .addGroup(layout.createSequentialGroup()
174         .addContainerGap()
175         .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
176         .addContainerGap())
177     );
178 layout.setVerticalGroup(
179     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
180     .addGroup(layout.createSequentialGroup()
181         .addContainerGap()
182         .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
183         .addContainerGap())
184     );
185 pack();
186 // </editor-fold>
187 private void openFileButtonActionPerformed(java.awt.event.ActionEvent evt) {
188
189 }
190
191 private void nextButtonActionPerformed(java.awt.event.ActionEvent evt) {
192
193     if (!openFileChooser.getSelectedFile().exists()) {
194         JOptionPane.showMessageDialog(null, "You must choose the file first!");
195         return;
196     }
197     this.hide();
198     TampilanSimulasi ts = new TampilanSimulasi(this.ca);
199     ts.setVisible(true);
200 }
201
202 private void backButtonActionPerformed(java.awt.event.ActionEvent evt) {
203     this.hide();
204     TampilanKondisiEksternal ke = new TampilanKondisiEksternal();
205     ke.setVisible(true);
206 }
207
208 private void openFileButtonMouseClicked(java.awt.event.MouseEvent evt) {
209     double[] kumpulanBobot;
210     int returnValue = openFileChooser.showOpenDialog(this);
211     StringBuilder sb = new StringBuilder();
212     if (returnValue == JFileChooser.APPROVE_OPTION) {
213         try {
214             br = new BufferedReader(new FileReader(openFileChooser.getSelectedFile()));
215             DefaultTableModel model = (DefaultTableModel) jTable1.getModel();
216
217             Object[] tableLines = br.lines().toArray();
218             ca = new CA(tableLines.length, InputDataHandler.getKetetanggaan(), 4);
219             kumpulanBobot = new double[InputDataHandler.getKetetanggaan()];
220             int i;
221             for (i = 0; i < tableLines.length; i++) {
222                 String lines = tableLines[i].toString().trim();

```

```

223         String[] dataRow = lines.split(",");
224         model.addRow(dataRow);
225
226         //memasukkan data dari fileInput ke kelas Entrepreneurs ca
227         ca.E[i] = new Entrepreneurs();
228         if (dataRow[0].equals("false")) {
229             ca.E[i].sex = false; // pria
230         } else {
231             ca.E[i].sex = true; //wanita
232         }
233         ca.E[i].age = Integer.parseInt(dataRow[1]);
234         ca.E[i].b_age = Integer.parseInt(dataRow[2]);
235         ca.E[i].b_category = Integer.parseInt(dataRow[3]);
236         ca.E[i].b_area = Integer.parseInt(dataRow[4]);
237         ca.E[i].education = Integer.parseInt(dataRow[5]);
238         ca.E[i].location = Integer.parseInt(dataRow[6]);
239         ca.E[i].income = Integer.parseInt(dataRow[7]);
240         ca.E[i].level = Integer.parseInt(dataRow[8]);
241         ca.E[i].point = 0.0;
242     }
243     } catch (IOException e) {
244         //messageLabel.setText("failed to load the file!");
245         Logger.getLogger(TampilanDataWirusaha.class.getName()).log(Level.SEVERE, null, e);
246     }
247 }
248
249 private void nextButtonMouseClicked(java.awt.event.MouseEvent evt) {
250
251 }
252
253
254 /**
255  * @param args the command line arguments
256  */
257 public static void main(String args[]) {
258     /* Set the Nimbus look and feel */
259     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
260     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
261      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
262      */
263     try {
264         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
265             if ("Nimbus".equals(info.getName())) {
266                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
267                 break;
268             }
269         }
270     } catch (ClassNotFoundException ex) {
271         java.util.logging.Logger.getLogger(TampilanDataWirusaha.class.getName()).log(java.util.logging.Level.SEVERE, null, ex
272     );
273     } catch (InstantiationException ex) {
274         java.util.logging.Logger.getLogger(TampilanDataWirusaha.class.getName()).log(java.util.logging.Level.SEVERE, null, ex
275     );
276     } catch (IllegalAccessException ex) {
277         java.util.logging.Logger.getLogger(TampilanDataWirusaha.class.getName()).log(java.util.logging.Level.SEVERE, null, ex
278     );
279     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
280     }
281     //</editor-fold>
282     /* Create and display the form */
283     java.awt.EventQueue.invokeLater(new Runnable() {
284         public void run() {
285             new TampilanDataWirusaha().setVisible(true);
286         }
287     });
288 }
289
290 // Variables declaration - do not modify
291 private javax.swing.JButton backButton;
292 private javax.swing.JLabel jLabel1;
293 private javax.swing.JLabel jLabel2;
294 private javax.swing.JPanel jPanel1;
295 private javax.swing.JPanel jPanel2;
296 private javax.swing.JScrollPane jScrollPane1;
297 private javax.swing.JTable jTable1;
298 private javax.swing.JLabel messageLabel;
299 public javax.swing.JButton nextButton;
300 private javax.swing.JButton openFileButton;
301 // End of variables declaration
302 }

```

Listing A.11: Entrepreneurs.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  import java.io.BufferedWriter;
9  import java.io.File;
10 import java.io.FileNotFoundException;
11 import java.io.FileWriter;

```

```

12 import java.io.IOException;
13 import java.io.PrintWriter;
14 import java.util.logging.Level;
15 import java.util.logging.Logger;
16 import javax.swing.JOptionPane;
17
18 /**
19  *
20  * @author Vanessa
21  */
22 public class TampilanSimulasi extends javax.swing.JFrame {
23
24     /**
25      * Creates new form TampilanSimulasi
26      */
27     CA ca;
28
29     public TampilanSimulasi(CA ca) {
30         initComponents();
31         this.ca = ca;
32     }
33
34     /**
35      * This method is called from within the constructor to initialize the form.
36      * WARNING: Do NOT modify this code. The content of this method is always
37      * regenerated by the Form Editor.
38      */
39     @SuppressWarnings("unchecked")
40     // <editor-fold defaultstate="collapsed" desc="Generated Code">
41     private void initComponents() {
42
43         jLabel1 = new javax.swing.JLabel();
44         jLabel2 = new javax.swing.JLabel();
45         jLabel3 = new javax.swing.JLabel();
46         jLabel4 = new javax.swing.JLabel();
47         jLabel5 = new javax.swing.JLabel();
48         jLabel6 = new javax.swing.JLabel();
49         nilaiA = new javax.swing.JTextField();
50         nilaiB = new javax.swing.JTextField();
51         nilaiC = new javax.swing.JTextField();
52         nilaiThreshold = new javax.swing.JTextField();
53         jLabel7 = new javax.swing.JLabel();
54         simulateButton = new javax.swing.JButton();
55         nilaiPeriode = new javax.swing.JTextField();
56         jLabel8 = new javax.swing.JLabel();
57
58         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
59
60         jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
61         jLabel1.setText("SIMULATOR_ECA");
62
63         jLabel2.setText("Simulasi:");
64
65         jLabel3.setText("a:");
66
67         jLabel4.setText("b:");
68
69         jLabel5.setText("c:");
70
71         jLabel6.setText("Threshold:");
72
73         nilaiA.addActionListener(new java.awt.event.ActionListener() {
74             public void actionPerformed(java.awt.event.ActionEvent evt) {
75                 nilaiAActionPerformed(evt);
76             }
77         });
78
79         nilaiB.addActionListener(new java.awt.event.ActionListener() {
80             public void actionPerformed(java.awt.event.ActionEvent evt) {
81                 nilaiBActionPerformed(evt);
82             }
83         });
84
85         nilaiC.addActionListener(new java.awt.event.ActionListener() {
86             public void actionPerformed(java.awt.event.ActionEvent evt) {
87                 nilaiCActionPerformed(evt);
88             }
89         });
90
91         nilaiThreshold.addActionListener(new java.awt.event.ActionListener() {
92             public void actionPerformed(java.awt.event.ActionEvent evt) {
93                 nilaiThresholdActionPerformed(evt);
94             }
95         });
96
97         jLabel7.setText("Periode:");
98
99         simulateButton.setText("SIMULATE");
100        simulateButton.addMouseListener(new java.awt.event.MouseAdapter() {
101            public void mouseClicked(java.awt.event.MouseEvent evt) {
102                simulateButtonMouseClicked(evt);
103            }
104        });
105        simulateButton.addActionListener(new java.awt.event.ActionListener() {
106            public void actionPerformed(java.awt.event.ActionEvent evt) {
107                simulateButtonActionPerformed(evt);
108            }
109        });
110

```



```

111     nilaiPeriode.addActionListener(new java.awt.event.ActionListener() {
112         public void actionPerformed(java.awt.event.ActionEvent evt) {
113             nilaiPeriodeActionPerformed(evt);
114         }
115     });
116
117     jLabel8.setText("_bulan");
118
119     javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
120     getContentPane().setLayout(layout);
121     layout.setHorizontalGroup(
122         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
123         .addGroup(layout.createSequentialGroup()
124             .addGap(33, 33, 33)
125             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
126                 .addComponent(jLabel2)
127                 .addGroup(layout.createSequentialGroup()
128                     .addComponent(jLabel3)
129                     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
130                     .addComponent(nilaiA)))
131             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
132             .addComponent(jLabel4)
133             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
134             .addComponent(nilaiB, javax.swing.GroupLayout.PREFERRED_SIZE, 29, javax.swing.GroupLayout.PREFERRED_SIZE)
135             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
136             .addComponent(jLabel5)
137             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
138             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
139                 .addComponent(jLabel1)
140                 .addGroup(layout.createSequentialGroup()
141                     .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
142                         .addComponent(simulateButton)
143                         .addGroup(layout.createSequentialGroup()
144                             .addComponent(nilaiC, javax.swing.GroupLayout.PREFERRED_SIZE, 31, javax.swing.GroupLayout.PREFERRED_SIZE)
145                             .addGap(18, 18, 18)
146                             .addComponent(jLabel6)))
147                     .addGap(13, 13, 13)
148                     .addComponent(nilaiThreshold, javax.swing.GroupLayout.PREFERRED_SIZE, 34, javax.swing.GroupLayout.PREFERRED_SIZE)
149                     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
150                     .addComponent(jLabel7)))
151             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
152             .addComponent(nilaiPeriode, javax.swing.GroupLayout.PREFERRED_SIZE, 37, javax.swing.GroupLayout.PREFERRED_SIZE)
153             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
154             .addComponent(jLabel8)
155             .addGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
156         );
157     layout.setVerticalGroup(
158         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
159         .addGroup(layout.createSequentialGroup()
160             .addGap(10, 10, 10)
161             .addComponent(jLabel1)
162             .addGap(30, 30, 30)
163             .addComponent(jLabel2)
164             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
165             .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
166                 .addComponent(jLabel3)
167                 .addComponent(jLabel4)
168                 .addComponent(jLabel5)
169                 .addComponent(jLabel6)
170                 .addComponent(nilaiA, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
171                     swing.GroupLayout.PREFERRED_SIZE)
172                 .addComponent(nilaiB, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
173                     swing.GroupLayout.PREFERRED_SIZE)
174                 .addComponent(nilaiC, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.
175                     swing.GroupLayout.PREFERRED_SIZE)
176                 .addComponent(nilaiThreshold, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
177                     javax.swing.GroupLayout.PREFERRED_SIZE)
178                 .addComponent(jLabel7)
179                 .addComponent(nilaiPeriode, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
180                     javax.swing.GroupLayout.PREFERRED_SIZE)
181                 .addComponent(jLabel8))
182             .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 27, Short.MAX_VALUE)
183             .addComponent(simulateButton)
184             .addGap(21, 21, 21))
185         );
186
187     pack();
188 } // </editor-fold>
189
190 private void nilaiAActionPerformed(java.awt.event.ActionEvent evt) {
191 }
192
193 private void simulateButtonActionPerformed(java.awt.event.ActionEvent evt) {
194 }
195
196 private void nilaiBActionPerformed(java.awt.event.ActionEvent evt) {
197 }
198
199 private void nilaiCActionPerformed(java.awt.event.ActionEvent evt) {
200 }
201
202 private void nilaiThresholdActionPerformed(java.awt.event.ActionEvent evt) {

```



```

203     }
204
205     private void nilaiPeriodeActionPerformed(java.awt.event.ActionEvent evt) {
206
207     }
208
209     private void simulateButtonMouseClicked(java.awt.event.MouseEvent evt) {
210         boolean checker = true;
211         double a = 0.0;
212         double b = 0.0;
213         double c = 0.0;
214         double[] kumpulanBobot = new double[3];
215         int m = 0;
216         if (nilaiA.getText().equals("")) {
217             InputDataHandler.inputDataSimulasi("nilaiA", null);
218             checker = false;
219         } else if (nilaiB.getText().equals("")) {
220             InputDataHandler.inputDataSimulasi("nilaiB", null);
221             checker = false;
222         } else if (nilaiC.getText().equals("")) {
223             InputDataHandler.inputDataSimulasi("nilaiC", null);
224             checker = false;
225         } else if (nilaiPeriode.getText().equals("")) {
226             InputDataHandler.inputDataSimulasi("periode", null);
227             checker = false;
228         } else if (nilaiThreshold.getText().equals("")) {
229             InputDataHandler.inputDataSimulasi("threshold", null);
230             checker = false;
231         } else {
232             if (nilaiA.getText().equals("")) {
233                 InputDataHandler.inputDataSimulasi("nilaiA", nilaiA.getText());
234                 a = Double.parseDouble(nilaiA.getText());
235                 kumpulanBobot[m] = a;
236                 m++;
237             } if (nilaiB.getText().equals("")) {
238                 InputDataHandler.inputDataSimulasi("nilaiB", nilaiB.getText());
239                 b = Double.parseDouble(nilaiB.getText());
240                 kumpulanBobot[m] = b;
241                 m++;
242             } if (nilaiC.getText().equals("")) {
243                 InputDataHandler.inputDataSimulasi("nilaiC", nilaiC.getText());
244                 c = Double.parseDouble(nilaiC.getText());
245                 kumpulanBobot[m] = c;
246                 m++;
247             } if (nilaiPeriode.getText().equals("")) {
248                 InputDataHandler.inputDataSimulasi("periode", nilaiPeriode.getText());
249                 if (nilaiThreshold.getText().equals("")) {
250                     InputDataHandler.inputDataSimulasi("threshold", nilaiThreshold.getText());
251                 }
252             }
253         }
254     }
255 }
256
257 int totalNilai = 0;
258 for (int i = 0; i < kumpulanBobot.length; i++) {
259     totalNilai += kumpulanBobot[i]*100;
260 }
261 if (totalNilai!=100) {
262     JOptionPane.showMessageDialog(null, "The_sum_of_a,b_and_c's_value_must_1!");
263 }
264
265 if (checker == false) {
266     JOptionPane.showMessageDialog(null, "You_must_fill_the_text_field_first!");
267 }
268
269 double[] composition = new double[]{a, b, c};
270
271 double[] POAf = new double[]{8.6, 17.7, 28.4, 29.5, 15.8}; // female
272 double[] POAm = new double[]{8.3, 14.5, 26.7, 36.2, 14.3}; // male
273
274 // Perceived Opportunities Education
275 double[] POEf = new double[]{1.8, 17.4, 23.4, 49.8, 7.4, 0.1};
276 double[] POEm = new double[]{0.7, 11.8, 19.9, 54.7, 12.6, 0.3};
277
278 // Perceived Opportunities Location
279 double[] POLf = new double[]{0.3, 6.4, 4.8, 2.8, 1.4, 3.5, 1.7, 46.3, 9.6, 6.1, 9.5, 2.5, 1.1, 1.0, 0.6, 2.4};
280 double[] POLm = new double[]{0.5, 4.4, 4.5, 2.3, 1.9, 3.8, 2.1, 47.6, 11.1, 6.3, 8.4, 2.7, 0.9, 1.1, 0.5, 2.0};
281
282 // Perceived Opportunities Income
283 double[] POIf = new double[]{42.7, 41.5, 10.8, 2.8, 1.5, 0.3, 0, 0.5};
284 double[] POIm = new double[]{42.1, 41.7, 11.0, 3.4, 0.7, 0.3, 0.5, 0.2};
285
286 // Perceived Capabilities Age
287 double[] PCAf = new double[]{8.9, 16.1, 28.2, 31.6, 15.1};
288 double[] PCAm = new double[]{8.5, 17.3, 26.1, 33.4, 14.7};
289
290 // Perceived Capabilities Education
291 double[] PCEf = new double[]{1.7, 15.4, 22.8, 51.5, 8.2, 0.4};
292 double[] PCEm = new double[]{0.9, 12.4, 17.4, 56.9, 12.0, 0.5};
293
294 // Perceived Capabilities Income
295 double[] PCIf = new double[]{41.4, 43.0, 10.2, 3.1, 1.4, 0.2, 0.2, 0.4};
296 double[] PCIm = new double[]{42.9, 42.1, 10.5, 3.1, 0.8, 0.3, 0.2, 0.2};
297
298 // Perceived Capabilities Location
299 double[] PCLf = new double[]{0.4, 7.5, 3.9, 2.4, 2.0, 3.2, 1.6, 41.1, 10.8, 6.9, 9.0, 3.7, 1.2, 1.0, 0.8, 4.4};
300 double[] PCLm = new double[]{0.7, 5.4, 3.4, 2.6, 3.0, 3.8, 1.8, 41.1, 11.5, 7.2, 8.7, 3.5, 1.2, 0.9, 0.7, 4.3};
301 // Role Model Age

```

```

302 double[] RMAf = new double[]{7.5, 17.6, 26.8, 31.0, 17.1};
303 double[] RMAm = new double[]{9.1, 16.9, 25.3, 34.4, 14.3};
304
305 // Role Model Income
306 double[] RMIf = new double[]{43.1, 41.8, 9.7, 3.0, 1.6, 0.4, 0.1, 0.4};
307 double[] RMIm = new double[]{42.9, 42.1, 10.4, 3.0, 0.7, 0.2, 0.5, 0.2};
308
309 // Fear of Failuer Age
310 double[] FFAf = new double[]{8.2, 16.4, 23.5, 32.4, 19.5};
311 double[] FFAm = new double[]{7.2, 14.3, 23.6, 36.4, 18.6};
312
313 // Fear of Failure Education
314 double[] FFEf = new double[]{2.3, 13.9, 22.9, 51.7, 8.7, 0.5};
315 double[] FFEm = new double[]{0.7, 12.1, 18.4, 57.4, 11.3, 0};
316
317 // Fear of Failure Location
318 double[] FFLf = new double[]{0.7, 10.3, 3.4, 3.8, 3.1, 4.2, 2.5, 36.9, 2.6, 7.8, 12.1, 4.6, 1.9, 1.0, 0.4, 4.8};
319 double[] FFLm = new double[]{0.7, 8.9, 3.4, 2.4, 4.2, 5.4, 2.9, 35.4, 1.8, 7.1, 13.2, 4.7, 2.2, 1.0, 0.5, 6.1};
320
321 // Media Attention Location
322 double[] MALf = new double[]{0.7, 9.9, 3.6, 3.5, 5.4, 4.1, 1.9, 41.1, 6.3, 9.1, 4.7, 2.9, 1.1, 1.0, 0.7, 4.0};
323 double[] MALm = new double[]{0.9, 8.1, 3.4, 3.6, 4.7, 4.9, 1.9, 41.8, 6.5, 8.2, 5.8, 2.9, 1.3, 1.1, 0.7, 4.1};
324
325 // Media Attention Income
326 double[] MAIf = new double[]{44.4, 41.6, 9.1, 2.7, 1.4, 0.2, 0.2, 0.4};
327 double[] MAIm = new double[]{44.0, 40.0, 11.4, 3.1, 0.6, 0.2, 0.4, 0.2};
328
329 // High Status Successful Income
330 double[] HSSIf = new double[]{45.6, 41.7, 8.5, 2.2, 1.2, 0.3, 0.1, 0.3};
331 double[] HSSIm = new double[]{46.2, 39.5, 10.2, 2.8, 0.6, 0.2, 0.4, 0.2};
332
333 //High Status Successful Location
334 double[] HSSLf = new double[]{0.7, 9.0, 2.7, 2.6, 5.4, 4.5, 1.8, 35.0, 9.8, 8.3, 10.8, 2.8, 1.5, 1.0, 0.8, 3.4};
335 double[] HSSLm = new double[]{0.8, 7.2, 2.3, 3.2, 4.8, 4.5, 2.2, 37.0, 10.2, 7.6, 10.7, 2.9, 1.6, 1.1, 0.8, 3.1};
336
337 // High Status Successful Age
338 double[] HSSAf = new double[]{10, 17, 26, 31, 17};
339 double[] HSSAm = new double[]{9, 16, 25, 33, 16};
340
341 // High Status Successful Education
342 double[] HSSEf = new double[]{2, 15, 23, 52, 8, 0};
343 double[] HSSEm = new double[]{1, 12, 19, 56, 11, 0};
344
345 // Faktor Publik
346 double[] pfs = new double[]{3.06, 2.69, 2.22, 2.53, 2.54, 3.3, 2.31, 3.25, 3.92, 2.82, 3.45, 3.29};
347 double[] pfw = InputDataHandler.getDataEksternal();
348
349 double[] nw = InputDataHandler.getBobot();
350 int[] nr = InputDataHandler.getRelation();
351
352 ca.pub.setFactors(pfs);
353 ca.pub.setWeights(pfw);
354
355 ca.N.setWeight(nw);
356 ca.N.setRelation(nr);
357
358 int maxIter = Integer.parseInt(InputDataHandler.getValue("periode")); // masukan periode
359
360 Entrepreneurs[][] e = new Entrepreneurs[maxIter][];
361 String[] line = new String[maxIter];
362 try {
363     PrintWriter pw = new PrintWriter(new File("D:\\output.csv"));
364     StringBuilder sb = new StringBuilder();
365     for (int i = 0; i < maxIter; i++) {
366         sb.append("Bulan_ke-" + i);
367         sb.append('\n');
368
369         ca.NeighborhoodDefinition();
370         ca.calculatePoint(POAm, POAf, POEm, POEf, POLm, POLf, POIm, POIf, PCAm, PCAf, PCEm, PCEf, PCLm, PCLf, PCIm, PCIf,
371             RMAm, RMAf, RMIm, RMIf, FFAf, FFAm, FFEf, FFEm, FFLf, FFLm, MALf, MALm, MAIf, MAIm, HSSIf, HSSIm, HSSLf,
372             HSSLm, HSSAf, HSSAm, HSSEf, HSSEm);
373         Entrepreneurs[] nE;
374         e[i] = ca.stateTransition(ca, composition);
375
376         for (int j = 0; j < e[i].length; j++) {
377             sb.append(e[i][j].toString2());
378             sb.append('\n');
379         }
380         // perubahan disimpan dulu
381         ca.E = e[i];
382         // lalu baru diprint
383         line[i] = ca.print(i);
384     }
385     pw.write(sb.toString());
386     pw.close();
387 } catch (FileNotFoundException ex) {
388     Logger.getLogger(TampilanSimulasi.class.getName()).log(Level.SEVERE, null, ex);
389 }
390 this.hide();
391 TampilanHasil th = new TampilanHasil(line);
392
393 th.setVisible(true);
394
395
396 /**
397  * @param args the command line arguments
398  */

```

```

399 public static void main(String args[]) {
400     /* Set the Nimbus look and feel */
401     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
402     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
403      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
404      */
405     try {
406         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
407             if ("Nimbus".equals(info.getName())) {
408                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
409                 break;
410             }
411         }
412     } catch (ClassNotFoundException ex) {
413         java.util.logging.Logger.getLogger(TampilanSimulasi.class
414             .getName()).log(java.util.logging.Level.SEVERE, null, ex);
415     } catch (InstantiationException ex) {
416         java.util.logging.Logger.getLogger(TampilanSimulasi.class
417             .getName()).log(java.util.logging.Level.SEVERE, null, ex);
418     } catch (IllegalAccessException ex) {
419         java.util.logging.Logger.getLogger(TampilanSimulasi.class
420             .getName()).log(java.util.logging.Level.SEVERE, null, ex);
421     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
422         java.util.logging.Logger.getLogger(TampilanSimulasi.class
423             .getName()).log(java.util.logging.Level.SEVERE, null, ex);
424     }
425     //</editor-fold>
426
427     /* Create and display the form */
428     java.awt.EventQueue.invokeLater(new Runnable() {
429         public void run() {
430             //
431             new TampilanSimulasi(this.ca).setVisible(true);
432         }
433     });
434 }
435
436 // Variables declaration - do not modify
437 private javax.swing.JLabel jLabel1;
438 private javax.swing.JLabel jLabel2;
439 private javax.swing.JLabel jLabel3;
440 private javax.swing.JLabel jLabel4;
441 private javax.swing.JLabel jLabel5;
442 private javax.swing.JLabel jLabel6;
443 private javax.swing.JLabel jLabel7;
444 private javax.swing.JLabel jLabel8;
445 private javax.swing.JTextField nilaiA;
446 private javax.swing.JTextField nilaiB;
447 private javax.swing.JTextField nilaiC;
448 private javax.swing.JTextField nilaiPeriode;
449 private javax.swing.JTextField nilaiThreshold;
450 public javax.swing.JButton simulateButton;
451 // End of variables declaration
452 }

```

Listing A.12: TampilanHasil.java

```

1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package ecasimulatorjframe;
7
8  import javax.swing.table.DefaultTableModel;
9
10 /**
11  *
12  * @author Vanessa
13  */
14 public class TampilanHasil extends javax.swing.JFrame {
15
16     /**
17      * Creates new form TampilanHasil
18      */
19     String[] res;
20
21     public TampilanHasil(String[] res) {
22         initComponents();
23         this.res = res;
24     }
25
26     /**
27      * This method is called from within the constructor to initialize the form.
28      * WARNING: Do NOT modify this code. The content of this method is always
29      * regenerated by the Form Editor.
30      */
31     @SuppressWarnings("unchecked")
32     // <editor-fold defaultstate="collapsed" desc="Generated Code">
33     private void initComponents() {
34
35         jScrollPane1 = new javax.swing.JScrollPane();
36         jTable1 = new javax.swing.JTable();
37         jLabel1 = new javax.swing.JLabel();
38
39         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
40         addWindowListener(new java.awt.event.WindowAdapter() {
41             public void windowActivated(java.awt.event.WindowEvent evt) {

```

```

42         formWindowActivated(evt);
43     }
44     public void windowOpened(java.awt.event.WindowEvent evt) {
45         formWindowOpened(evt);
46     }
47 });
48
49 jTable1.setModel(new javax.swing.table.DefaultTableModel(
50     new Object [][] {
51
52     },
53     new String [] {
54         "Iterasi", "Potential", "Nascent", "New_Business_Manager", "Established", "Retired"
55     }
56 ));
57 jScrollPane1.setViewportView(jTable1);
58
59 jLabel1.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
60 jLabel1.setText("HASIL_SIMULASI");
61
62 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
63 getContentPane().setLayout(layout);
64 layout.setHorizontalGroup(
65     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
66     .addGroup(layout.createSequentialGroup()
67         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
68             .addGroup(layout.createSequentialGroup()
69                 .addGap(23, 23, 23)
70                 .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 632, javax.swing.GroupLayout.PREFERRED_SIZE))
71             .addGroup(layout.createSequentialGroup()
72                 .addGap(267, 267, 267)
73                 .addComponent(jLabel1)))
74         .addContainerGap(21, Short.MAX_VALUE))
75 );
76 layout.setVerticalGroup(
77     layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
78     .addGroup(layout.createSequentialGroup()
79         .addGap(29, 29, 29)
80         .addComponent(jLabel1)
81         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 34, Short.MAX_VALUE)
82         .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
83         .addGap(27, 27, 27))
84 );
85
86 pack();
87 } // </editor-fold>
88
89 private void formWindowActivated(java.awt.event.WindowEvent evt) {
90
91 }
92
93 private void formWindowOpened(java.awt.event.WindowEvent evt) {
94     DefaultTableModel model = (DefaultTableModel) jTable1.getModel();
95     for (int i = 0; i < res.length; i++) {
96         String[] dataRow1 = res[i].split(",");
97         model.addRow(dataRow1);
98     }
99 }
100
101 /**
102  * @param args the command line arguments
103  */
104 public static void main(String args[]) {
105     /* Set the Nimbus look and feel */
106     //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
107     /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
108      * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
109      */
110     try {
111         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
112             if ("Nimbus".equals(info.getName())) {
113                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
114                 break;
115             }
116         }
117     } catch (ClassNotFoundException ex) {
118         java.util.logging.Logger.getLogger(TampilanHasil.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
119     } catch (InstantiationException ex) {
120         java.util.logging.Logger.getLogger(TampilanHasil.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
121     } catch (IllegalAccessException ex) {
122         java.util.logging.Logger.getLogger(TampilanHasil.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
123     } catch (javax.swing.UnsupportedLookAndFeelException ex) {
124         java.util.logging.Logger.getLogger(TampilanHasil.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
125     }
126     //</editor-fold>
127
128     /* Create and display the form */
129     java.awt.EventQueue.invokeLater(new Runnable() {
130         public void run() {
131             //
132             new TampilanHasil().setVisible(true);
133         }
134     });
135 }
136
137 // Variables declaration - do not modify
138 private javax.swing.JLabel jLabel1;

```

```
139| private javax.swing.JScrollPane jScrollPane1;  
140| private javax.swing.JTable jTable1;  
141| // End of variables declaration  
142|  
143| }
```