LAMPIRAN

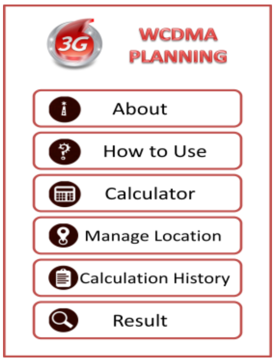
LAMPIRAN A

PERANCANGAN

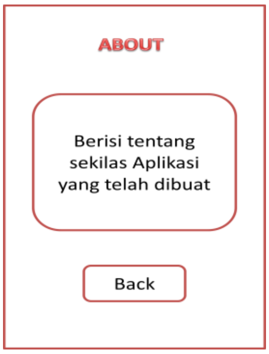
ANTARMUKA APLIKASI

Perancangan antarmuka aplikasi yang dibangun terdiri dari:

Rancangan antarmuka *home* terdiri atas 6 tombol, yaitu tombol “About” mengarahkan ke halaman informasi mengenai aplikasi, “How to Use” masuk ke halaman petunjuk penggunaan aplikasi. Tombol “Calculator” mengarahkan ketahapan perencanaan seluler. Tombol “Manage Location” digunakan melihat hasil perencanaan. Tombol “Calculation History” masuk ke dalam data yang sudah dihitung dan ditampilkan pada layar ini dan tombol “Result”.

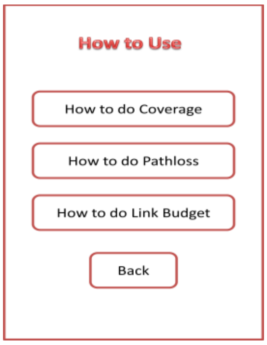


Rancangan antarmuka halaman “About” berisi mengenai informasi mengenai aplikasi yang dibuat.



Berisi tentang sekilas Aplikasi yang telah dibuat

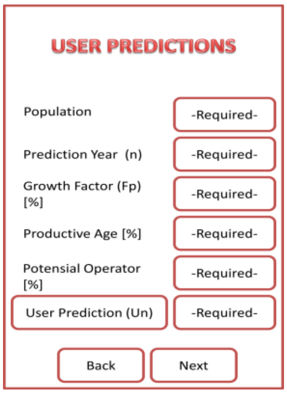
Rancangan antarmuka petunjuk penggunaan aplikasi dengan header “How to Use?” berisi daftar petunjuk penggunaan aplikasi.



Untuk antarmuka *Calculator*, terdiri beberapa antarmuka untuk menghitung tahapan perencanaan seluler. Antarmuka tersebut antara lain:

a. Antarmuka *User Prediction*

Antarmuka perhitungan prediksi jumlah user di tahun mendatang.

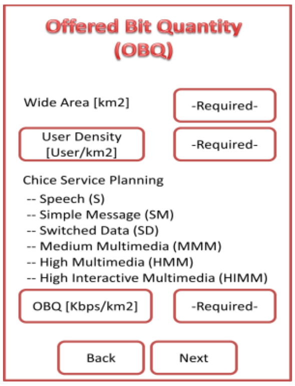


USER PREDICTIONS

Population	-Required-
Prediction Year (n)	-Required-
Growth Factor (Fp) [%]	-Required-
Productive Age [%]	-Required-
Potensial Operator [%]	-Required-
User Prediction (Un)	-Required-

b. Antarmuka *OBQ*

Antarmuka perhitungan trafik berdasarkan layanan perencanaan.



Offered Bit Quantity (OBQ)

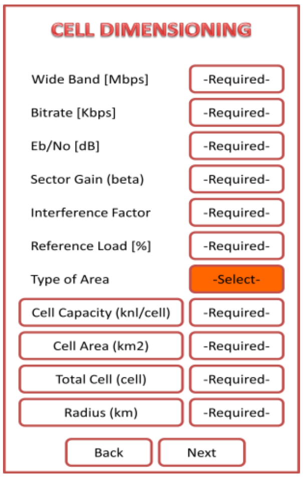
Wide Area [km2]	-Required-
User Density [User/km2]	-Required-

Chice Service Planning
-- Speech (S)
-- Simple Message (SM)
-- Switched Data (SD)
-- Medium Multimedia (MMM)
-- High Multimedia (HMM)
-- High Interactive Multimedia (HIMM)

OBQ [Kbps/km2]	-Required-

c. Antarmuka *Sell Dimensioning*

Antarmuka perhitungan kapasitas sel sehingga diperoleh luas sel, jumlah sel dan jari-jari sel.

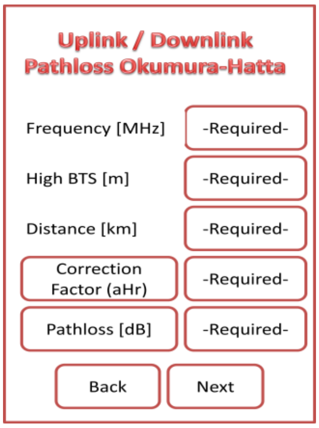


CELL DIMENSIONING

Wide Band [Mbps]	-Required-
Bitrate [Kbps]	-Required-
Eb/No [dB]	-Required-
Sector Gain (beta)	-Required-
Interference Factor	-Required-
Reference Load [%]	-Required-
Type of Area	-Select-
Cell Capacity (knl/cell)	-Required-
Cell Area (km2)	-Required-
Total Cell (cell)	-Required-
Radius (km)	-Required-

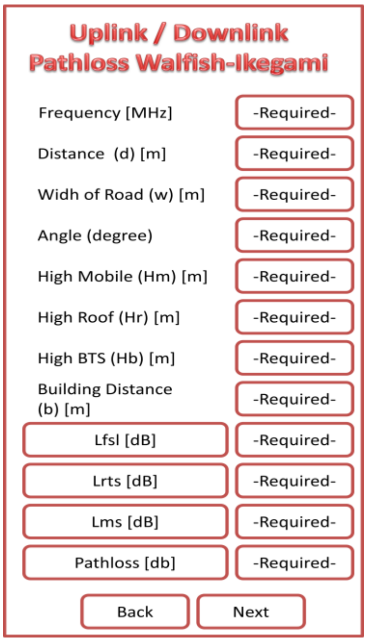
d. Antarmuka *Pathloss Okumura-Hatta Uplink & Downlink*

Antarmuka tersebut untuk menghitung *loss propagasi* pada daerah urban, suburban atau rural.



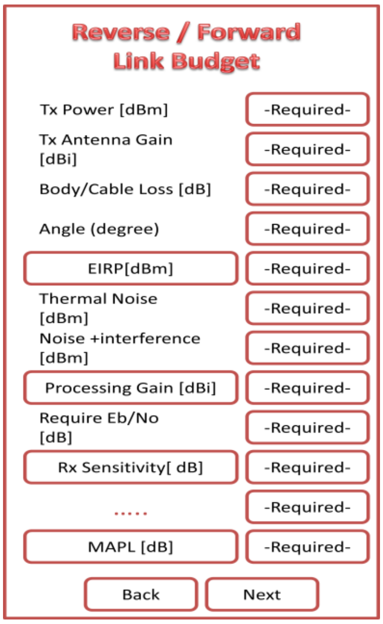
e. Antarmuka *Pathloss Walfish-Ikegami Uplink & Downlink*

Antarmuka tersebut untuk menghitung *loss propagasi* pada daerah dense urban.

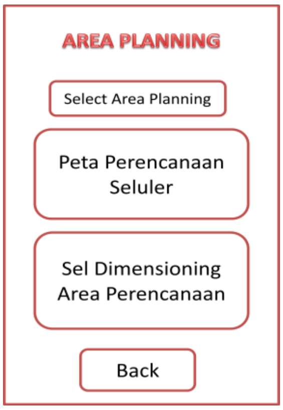


f. Antarmuka *Reverse Link Budget & Forward Link Budget*

Antarmuka *Reverse Link Budget* berisi tentang perhitungan link budget dari arah *uplink* dan *Forward Link Budget* tentang perhitungan link budget dari arah *downlink*.



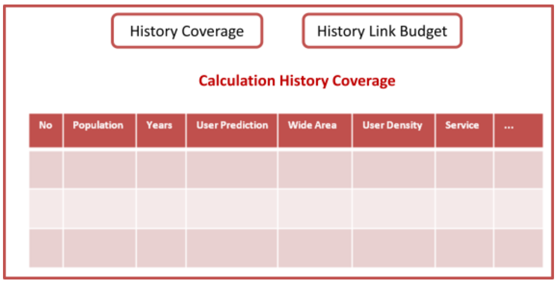
Untuk antarmuka *Manage Location*, melihat hasil perencanaan dalam bentuk peta perencanaan dan gambar sel :



Untuk antarmuka *Calculation History*, terdiri dari dua tab :

a. *History Coverage*

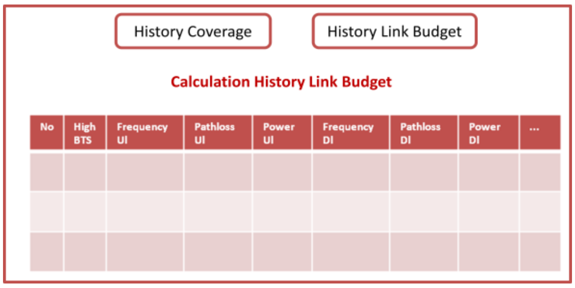
Antarmuka *History Coverage* berisi tentang data perhitungan dari halaman *user prediction* sampai halaman *sel dimensioning* yaitu data hasil perhitungan yang didapat sebagai acuan dalam menggambarkan coverage perencanaan sel.



No	Population	Years	User Prediction	Wide Area	User Density	Service	...

b. *History Link Budget*

Antarmuka *History Link Budget* berisi tentang data perhitungan *Link Budget* yang sudah dihitung.



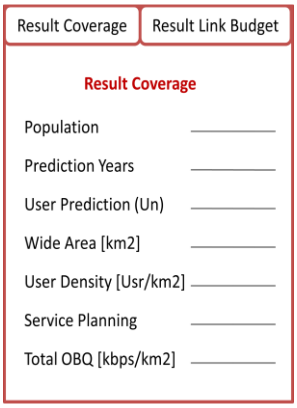
Calculation History Link Budget

No	High BTS	Frequency Ul	Pathloss Ul	Power Ul	Frequency Dl	Pathloss Dl	Power Dl	...

Untuk antarmuka *Result*, terdiri dari dua tab:

a. *Result Coverage*

Antarmuka *Result Coverage* berisi tentang hasil keseluruhan perhitungan *coverage.*



Result Coverage

Population ________

Prediction Years ________

User Prediction (Un) ________

Wide Area [km2] ________

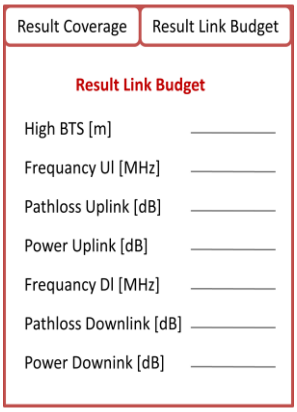
User Density [Usr/km2] ________

Service Planning ________

Total OBQ [kbps/km2] ________

b. *Result Link Budget*

Antarmuka *Result Link Budget* berisi tentang hasil keseluruhan perhitungan *link budget.*



Result Link Budget

High BTS [m] ________

Frequancy Ul [MHz] ________

Pathloss Uplink [dB] ________

Power Uplink [dB] ________

Frequancy Dl [MHz] ________

Pathloss Downlink [dB] ________

Power Downink [dB] ________

LAMPIRAN B

SPREADSHEET CALCULATIONS FOR TERRESTRIAL COMPONENT OF IMT-2000 FOR REGION 3

Spesifikasi Parameter pada Aplikasi

(Bitrate, Penetration Rate, BHCA dan Call Duration)

BITRATE (Kbps)		
Service	Uplink	Downlink
S	16	16
SM	14	14
SD	64	64
MMM	64	384
HMM	128	2000
HIMM	128	128

PENETRATION RATE			
Service	Building	Pedestian	Vehicular
S	0,73	0,73	0,73
SM	0,4	0,4	0,4
SD	0,13	0,13	0,13
MMM	0,15	0,15	0,15
HMM	0,15	0,15	0,15
HIMM	0,25	0,25	0,25

BHCA			
Service	Building	Pedestian	Vehicular
S	3	0,8	0,4
SM	0,6	0,3	0,2
SD	0,2	0,2	0,02
MMM	0,5	0,4	0,008
HMM	0,15	0,06	0,008
HIMM	0,14	0,07	0,011

CALL DURATION (Second)			
Service	Building	Pedestian	Vehicular
S	180	120	120
SM	3	3	3
SD	156	156	156
MMM	300	300	300
HMM	300	300	300
HIMM	120	120	120

LAMPIRAN C

SKENARIO PERHITUNGAN MANUAL

Perhitungan Manual Kecamatan Medan Labuhan:

a. Perhitungan prediksi jumlah pelanggan

$Un = Uo(1 + Fp)^n$

= 114202 (1 + 0,9%)^5

= 119434 jiwa

Jumlah pelanggan = 75% x 22% x 119434

= 19706,641 user

b. Perhitungan Kepadatan Pelanggan

Kepadatan Pelanggan $= \frac{Jumlah\ pelanggan}{Luas\ area}$

$= \frac{19706{,}641\ user}{40{,}68\ km2}$

= 484,431 user/km^2

Distribusi pelanggan : 50% building, 30% pedestrian dan 20% vehicular,

c. Perhitungan *Offered Bit Quantity* (OBQ)

$OBQ = \ \sigma\ x\ p\ x\ d\ x\ BHCA\ xBW$

OBQ Layanan	Type	σ	P	d	BHCA	BW	OBQ
Speech (S)	building	242,215	0,73	180	3	16	1527700,645
	pedestrian	1145,329	0,73	120	0,8	16	162954,735
	vehicular	96,886	0,73	120	0,4	16	54318,245
SM	building	242,215	0,4	3	0,6	14	2441,531
	pedestrian	1145,329	0,4	3	0,3	14	732,459
	vehicular	96,886	0,4	3	0,2	14	325,537
SD	building	242,215	0,13	156	0,2	64	62875,229
	pedestrian	1145,329	0,13	156	0,2	64	37725,137
	vehicular	96,886	0,13	156	0,02	64	2515,009
Total OBQ [Kb/hour/km2]							1851588,527
Total OBQ [Kbps/km2]							514,330

d. Perhitungan Kapasitas Sel

Kapasitas Sel $= \frac{W/R}{Eb/No}\ x\ \frac{\beta}{(1+f)}$

$= \frac{3{,}84\ Mbps/64Kbps}{2\ dB}\ x\ \frac{2{,}5}{(1+0{,}5)}$

= 63,096 → 63,096 x 64 x beban acuan 55%

= 2220,970 kanal/ sel

e. Perhitungan Luas Sel

$$Lcell = \frac{Kapasitas\ Informasi\ Tiap\ Sel}{Offered\ Bit\ Quantity\ (OBQ)}$$

$= \frac{2220{,}970}{514{,}330}$ = 4,318 km^2

f. Perhitungan Jumlah Sel

Jumlah sel $= \frac{Luas\ Area\ Pelayanan}{Luas\ Cakupan\ Sel}$

$= \frac{40{,}68}{4{,}318}$ = 9,421 ≈ 10 sel

g. Perhitungan Radius Sel

L_{cell} = 1,95 x r^2

$r = \sqrt{4{,}318/1{,}95}$ = 1,488 km^2

h. Perhitungan Pathloss Uplink

$a(H_m)$ = $[1{,}1 \log_{10} (f_c) – 0{,}7\ ]\ H_m – [1{,}56 \log_{10} (f_c) – 0{,}8\ ]$

$= [1{,}1 \log_{10} 1920 – 0{,}7\ ]\ 2 – [1{,}56 \log_{10} 1920 – 0{,}8\ ]$

= 1,501

$Lp = 46{,}3 + 33{,}9 \log_{10}(f_c) – 13{,}82 \log_{10}(H_b) – a(H_m) + [44{,}9 – 6{,}55 \log_{10}(H_b)]\ \log_{10}(d) + L_{other}$

$= 46{,}3 + 33{,}9 \log_{10}(1920) – 13{,}82 \log_{10}(40) – 1{,}501 + [44{,}9 – 6{,}55 \log_{10}(40)]\ \log_{10}(1{,}488) + 0$

= 139,901 dB

i. Perhitungan *Link Budget (Reverse dan Forward)*

Untuk perhitungan link budget dapat di lihat pada tabel 2.4 dan 2.5.

j. Perhitungan Pathloss Downlink

$a(H_m) = [1{,}1 \log_{10} (f_c) - 0{,}7\ ]\ H_m - [1{,}56 \log_{10} (f_c) - 0{,}8\ ]$

$= [1{,}1 \log_{10} 2110 - 0{,}7\ ]\ 2 - [1{,}56 \log_{10} 1920 - 0{,}8\ ]$

$= 1{,}528$

$Lp = 46{,}3 + 33{,}9 \log_{10}(f_c) - 13{,}82 \log_{10}(H_b) - a(H_m) + [44{,}9 - 6{,}55 \log_{10}(H_b)]\ \log_{10}(d) + L_{other}$

$= 46{,}3 + 33{,}9 \log_{10}(2110) - 13{,}82 \log_{10}(40) - 1{,}528 + [44{,}9 - 6{,}55 \log_{10}(40)]\ \log_{10}(1{,}488) + 0$

$= 141{,}628$ dB

Perhitungan Manual Kecamatan Medan Petisah

a. Perhitungan prediksi jumlah pelanggan

$Un = Uo(1 + Fp)^n$

$= 72040\ (1 + 0{,}9\%)^5$

$= 75341$ jiwa

Jumlah pelanggan = 75% x 22% x 75341

= 12431,205 user

b. Perhitungan Kepadatan Pelanggan

Kepadatan Pelanggan $= \frac{Jumlah\ pelanggan}{Luas\ area}$

$= \frac{12431{,}205\ user}{13{,}16\ km2}$

$= 944{,}620$ user/km^2

Distribusi pelanggan : 50% building, 30% pedestrian dan 20% vehicular,

c. Perhitungan *Offered Bit Quantity* (OBQ)

$OBQ = \sigma\ x\ p\ x\ d\ x\ BHCA\ xBW$

OBQ Layanan	**Type**	***σ***	**P**	**d**	**BHCA**	**BW**	**OBQ**
Speech (S)	Building	472,310	0,73	180	3	16	2978955,001
	Pedestrian	283,386	0,73	120	0,8	16	317755,200
	Vehicular	188,924	0,73	120	0,4	16	105918,400
SM	Building	472,310	0,4	3	0,6	14	4760,887
	Pedestrian	283,386	0,4	3	0,3	14	1428,266
	Vehicular	188,924	0,4	3	0,2	14	634,785
SD	Building	472,310	0,13	156	0,2	64	122604,175
	Pedestrian	283,386	0,13	156	0,2	64	73562,505
	Vehicular	188,924	0,13	156	0,02	64	4904,167
MMM	Building	472,310	0,15	300	0,5	64	680126,713
	Pedestrian	283,386	0,15	300	0,4	64	326460,822
	Vehicular	188,924	0,15	300	0,008	64	4352,811
Total OBQ [Kb/hour/km2]							4621463,732
Total OBQ [Kbps/km2]							1283,74

d. Perhitungan Kapasitas Sel

Kapasitas Sel $= \frac{W/R}{Eb/No}\ x\ \frac{\beta}{(1+f)}$

$= \frac{3{,}84\ Mbps/64Kbps}{2\ dB}\ x\ \frac{2{,}5}{(1+0{,}5)}$

= 63,096 → 10,516 x 64 x beban acuan 55%

= 2220,970 kanal/ sel

e. Perhitungan Luas Sel

$$Lcell = \frac{Kapasitas\ Informasi\ Tiap\ Sel}{Offered\ Bit\ Quantity\ (OBQ)}$$

$= \frac{2220{,}970}{1283{,}74}$ = 1,731 km^2

f. Perhitungan Jumlah Sel

$$\text{Jumlah sel } = \frac{Luas\ Area\ Pelayanan}{Luas\ Cakupan\ Sel}$$

$$= \frac{13{,}16}{1{,}730} = 7{,}607 \approx 8 \text{ sel}$$

g. Perhitungan Radius Sel

$L_{cell} = 1{,}95 \text{ x } 2{,}6 \text{ x } r^2$

$$r = \sqrt{1{,}731/(1{,}95)} = 0{,}942 \text{ km}^2$$

h. Perhitungan Pathloss Uplink

$a(H_m) = 3{,}2\ [\log_{10}(11{,}75\ H_m)]^2 - 4{,}97$

$= 3{,}2\ [\log_{10}(11{,}75 \text{ x } 2)]^2 - 4{,}97$

$= 1{,}045$

$Lp = 46{,}3 + 33{,}9 \log_{10}(f_c) - 13{,}82 \log_{10}(H_b) - a(H_m) + [44{,}9 - 6{,}55 \log_{10}(H_b)]\ \log_{10}(d) + L_{other}$

$= 46{,}3 + 33{,}9 \log_{10}(1920) - 13{,}82 \log_{10}(40) - 1{,}045 + [44{,}9 - 6{,}55 \log_{10}(40)]\ \log_{10}(0{,}942) + 3$

$= 136{,}525$ dB

i. Perhitungan *Link Budget (Reverse dan Forward)*

Untuk perhitungan link budget dapat di lihat pada tabel 2,4 dan 2,5,

j. Perhitungan Pathloss Downlink

$a(H_m) = 3{,}2\ [\log_{10}(11{,}75\ H_m)]^2 - 4{,}97$

$= 3{,}2\ [\log_{10}(11{,}75 \text{ x } 2)]^2 - 4{,}97$

$= 1{,}045$

$Lp = 46{,}3 + 33{,}9 \log_{10}(f_c) - 13{,}82 \log_{10}(H_b) - a(H_m) + [44{,}9 - 6{,}55 \log_{10}(H_b)]\ \log_{10}(d) + L_{other}$

$= 46{,}3 + 33{,}9 \log_{10}(2110) - 13{,}82 \log_{10}(40) - 1{,}045 + [44{,}9 - 6{,}55 \log_{10}(40)]\ \log_{10}(0{,}942) + 3$

$= 137{,}914$ dB

LAMPIRAN D

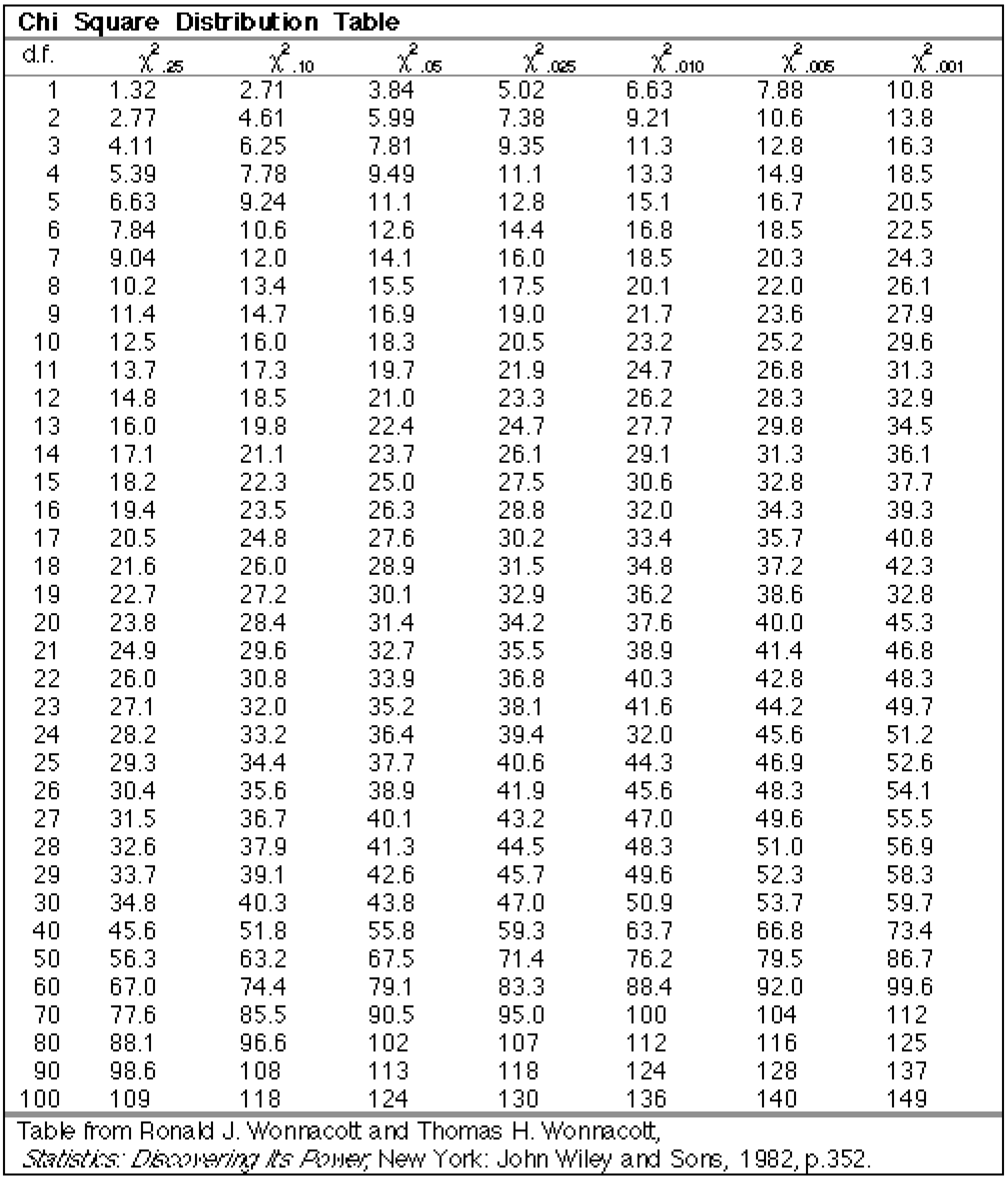
LEMBAR KUISIONER &

TABEL *CHI-SQUARE*

KUISIONER APLIKASI ANDROID "*3G-WCDMA Planning*"

1. Bagaimanakah desain tampilan antar muka (*user interface*) untuk aplikasi *3G-WCDMA Planning* ini ?
   A. Sangat bagus
   B. Bagus
   C. Cukup
   D. Buruk
   E. Sangat buruk
2. Bagaimanakah kelengkapan fitur layanan yang terdapat pada aplikasi ini ?
   A. Sangat lengkap
   B. Lengkap
   C. Cukup
   D. Tidak lengkap
   E. Sangat tidak lengkap
3. Bagaimanakah kemudahan penggunaan fitur-fitur di aplikasi ?
   A. Sangat mudah
   B. Mudah
   C. Cukup
   D. Sulit
   E. Sangat sulit
4. Seberapa jelas fitur "How to Use" menggambarkan cara penggunaan aplikasi ?
   A. Sangat jelas
   B. Jelas
   C. Cukup
   D. Kurang jelas
   E. Tidak jelas
5. Apakah ukuran font dan tombol sudah sesuai pada aplikasi ini ?
   A. Sangat sesuai
   B. Sesuai
   C. Cukup
   D. Kurang sesuai
   E. Tidak sesuai
6. Apakah aplikasi ini bermanfaat dalam perhitungan perencanaan seluler 3G-WCDMA?
   A. Sangat bermanfaat
   B. Bermanfaat
   C. Cukup
   D. Tidak bermanfaat
   E. Sangat tidak bermanfaat
7. Secara keseluruhan, baik dari segi fitur dan user interface, bagaimanakah penilaian anda terhadap aplikasi ini?
   A. Sangat bagus
   B. Bagus
   C. Cukup
   D. Buruk
   E. Sangat buruk

TABEL *CHI-SQUARE*



Chi Square Distribution Table

d.f.	$\chi^2_{.25}$	$\chi^2_{.10}$	$\chi^2_{.05}$	$\chi^2_{.025}$	$\chi^2_{.010}$	$\chi^2_{.005}$	$\chi^2_{.001}$
1	1.32	2.71	3.84	5.02	6.63	7.88	10.8
2	2.77	4.61	5.99	7.38	9.21	10.6	13.8
3	4.11	6.25	7.81	9.35	11.3	12.8	16.3
4	5.39	7.78	9.49	11.1	13.3	14.9	18.5
5	6.63	9.24	11.1	12.8	15.1	16.7	20.5
6	7.84	10.6	12.6	14.4	16.8	18.5	22.5
7	9.04	12.0	14.1	16.0	18.5	20.3	24.3
8	10.2	13.4	15.5	17.5	20.1	22.0	26.1
9	11.4	14.7	16.9	19.0	21.7	23.6	27.9
10	12.5	16.0	18.3	20.5	23.2	25.2	29.6
11	13.7	17.3	19.7	21.9	24.7	26.8	31.3
12	14.8	18.5	21.0	23.3	26.2	28.3	32.9
13	16.0	19.8	22.4	24.7	27.7	29.8	34.5
14	17.1	21.1	23.7	26.1	29.1	31.3	36.1
15	18.2	22.3	25.0	27.5	30.6	32.8	37.7
16	19.4	23.5	26.3	28.8	32.0	34.3	39.3
17	20.5	24.8	27.6	30.2	33.4	35.7	40.8
18	21.6	26.0	28.9	31.5	34.8	37.2	42.3
19	22.7	27.2	30.1	32.9	36.2	38.6	32.8
20	23.8	28.4	31.4	34.2	37.6	40.0	45.3
21	24.9	29.6	32.7	35.5	38.9	41.4	46.8
22	26.0	30.8	33.9	36.8	40.3	42.8	48.3
23	27.1	32.0	35.2	38.1	41.6	44.2	49.7
24	28.2	33.2	36.4	39.4	32.0	45.6	51.2
25	29.3	34.4	37.7	40.6	44.3	46.9	52.6
26	30.4	35.6	38.9	41.9	45.6	48.3	54.1
27	31.5	36.7	40.1	43.2	47.0	49.6	55.5
28	32.6	37.9	41.3	44.5	48.3	51.0	56.9
29	33.7	39.1	42.6	45.7	49.6	52.3	58.3
30	34.8	40.3	43.8	47.0	50.9	53.7	59.7
40	45.6	51.8	55.8	59.3	63.7	66.8	73.4
50	56.3	63.2	67.5	71.4	76.2	79.5	86.7
60	67.0	74.4	79.1	83.3	88.4	92.0	99.6
70	77.6	85.5	90.5	95.0	100	104	112
80	88.1	96.6	102	107	112	116	125
90	98.6	108	113	118	124	128	137
100	109	118	124	130	136	140	149

Table from Ronald J. Wonnacott and Thomas H. Wonnacott,
Statistics: Discovering Its Power, New York: John Wiley and Sons, 1982, p.352.

LAMPIRAN E

HASIL PENGUJIAN PERHITUNGAN

Tingkat kesalahan perhitungan perencanaan daerah Medan Labuhan:

No	Pengujian	Hasil Perhitungan Manual		Hasil Perhitungan Aplikasi		Persentasi Kesalahan	
1	Prediksi Jumlah User	19706,641		19706,681		-0,000202977	
2	Kepadatan User [user/km2]	484,431		484,415		0,003302953	
3	OBQ [kbps/km2]	514,33		514,313		0,00330538	
4	Kapasitas Sel [kanal/sel]	2220,97		2220,97		0	
5	Luas Sel [km2]	4,318		4,321		-0,069428373	
6	Total Sel [sel]	9,421		9,257		1,771632278	
7	Jari-jari Sel [km2]	1,488		1,489		-0,067159167	
Pathloss Okumura Hatta		Uplink	Downlink	Uplink	Downlink		
8	Faktor Koreksi (aHr)	1,501	1,528	1,501	1,528	0	0
9	Pathloss [dB]	139,901	141,264	139,911	141,273	-0,007147401	-0,006370644
10	EIRP [dBm]	21	45	21	45	0	0
11	Rx Noise Power [dBm]	-105,157	-100,157	-105,157	-100,157	0	0
12	Interference Margin [dB]	3,01	3,01	3,01	3,01	0	0
13	Processing Gain [dB]	17,782	17,782	17,782	17,782	0	0
14	Rx Sensitivity [dBm]	-117,928	-109,928	-117,929	-109,929	-0,000847968	-0,000909678
15	Pathloss [dB]	152,128	154,928	152,129	154,929	-0,000657337	-0,000645457
16	MAPL [dB]	138,828	141,628	138,829	141,629	-0,000720311	-0,00070607
rata-rata persentasi salah						**0,064937809**	

Tingkat kesalahan perhitungan perencanaan daerah Medan Perjuangan:

No	Pengujian	Hasil Perhitungan Manual		Hasil Perhitungan Aplikasi		Persentasi Kesalahan	
1	Prediksi Jumlah User	16543,418		16543,486		-0,000411038	
2	Kepadatan User [user/km2]	3794,362		3794,266		0,002530134	
3	OBQ [kbps/km2]	4980,078		4979,952		0,002530145	
4	Kapasitas Sel [kanal/sel]	2550,015		2550,015		0	
5	Luas Sel [km2]	0,512		0,512		0	
6	Total Sel [sel]	8,515		7,812		8,998975934	
7	Jari-jari Sel [km2]	0,512		0,512		0	
Pathloss Walfish-Ikegami		Uplink	Downlink	Uplink	Downlink		
8	Free Space Loss (Lfs) [dB]	98,272	99,092	98,272	99,092	0	0
9	Rooftop to Street (Lrts) [dB]	39,934	40,343	39,934	40,343	0	0
10	Multiscreen Loss (Lms) [dB]	-76,069	-75,142	-76,069	-75,142	0	0
11	Pathloss [dB]	98,272	99,092	98,272	99,092	0	0
12	EIRP [dBm]	24	55	24	55	0	0

13	Rx Noise Power [dBm]	-105,157	-100,157	-105,157	-100,157	0	0
14	Interference Margin [dB]	3,01	3,01	3,01	3,01	0	0
15	Processing Gain [dB]	14,771	2,833	14,771	2,833	0	0
16	Rx Sensitivity [dBm]	-115,518	-96,979	-115,518	-96,98	0	-0,00103114
17	Pathloss [dB]	152,718	151,979	152,718	151,98	0	-0,000657981
18	MAPL [dB]	139,418	138,679	139,418	138,68	0	-0,000721085
rata-rata persentasi salah						**0,310386723**	

Tingkat kesalahan perhitungan perencanaan daerah Medan Petisah:

No	Pengujian	Hasil Perhitungan Manual		Hasil Perhitungan Aplikasi		Persentasi Kesalahan	
1	Prediksi Jumlah User	12431,205		12431,212		-5,63099E-05	
2	Kepadatan User [user/km2]	944,62		944,605		0,001587965	
3	OBQ [kbps/km2]	1283,74		1283,719		0,001635872	
4	Kapasitas Sel [kanal/sel]	2220,97		2220,97		0	
5	Luas Sel [km2]	1,731		1,731		0	
6	Total Sel [sel]	7,607		7,51		1,291611185	
7	Jari-jari Sel [km2]	0,942		0,942		0	
Pathloss Okumura Hatta		Uplink	Downlink	Uplink	Downlink		
8	Faktor Koreksi (aHr)	1,045	1,045	1,045	1,045	0	0
9	Pathloss [dB]	136,525	137,914	136,526	137,915	-0,000732461	-0,000725084
10	EIRP [dBm]	21	50	21	50	0	0
11	Rx Noise Power [dBm]	-105,157	-100,157	-105,157	-100,157	0	0
12	Interference Margin [dB]	3,01	3,01	3,01	3,01	0	0
13	Processing Gain [dB]	17,782	10	17,782	10	0	0
14	Rx Sensitivity [dBm]	-117,928	-102,346	-117,929	-102,347	-0,000847968	-0,000977068
15	Pathloss [dB]	152,128	152,346	152,129	152,347	-0,000657337	-0,000656396
16	MAPL [dB]	138,828	139,046	138,829	139,047	-0,000720311	-0,000719181
rata-rata persentasi salah						**0,051549716**	

Tingkat penyimpangan & akurasi perhitungan perencanaan daerah Medan Labuhan:

No	Manual (ei)	Aplikasi (oi)	(oi-ei)^2
1	19706,641	19706,681	8,11909E-08
2	484,431	484,415	5,28455E-07
3	514,33	514,313	5,61896E-07
4	2220,97	2220,97	0
5	4,318	4,321	2,0843E-06
6	9,421	9,257	0,002854899
7	1,488	1,489	6,72043E-07
8	39,353	39,353	0
9	124,392	124,392	0
10	21	21	0
11	-105,157	-105,157	0
12	3,01	3,01	0
13	17,782	17,782	0
14	-117,928	-117,929	-8,47975E-09
15	152,128	152,129	6,57341E-09
16	138,828	138,829	7,20316E-09
17	39,965	39,965	0
18	125,169	125,169	0
19	43	43	0
20	-100,157	-100,157	0
21	3,01	3,01	0
22	14,771	14,771	0
23	-106,918	-106,918	0
24	149,918	149,918	0
25	136,618	136,618	0
Tingkat Penyimpangan			**0,002858832**
AKURASI [%]			**99,71411682**

Tingkat penyimpangan & akurasi perhitungan perencanaan daerah Medan Perjuangan:

No	Manual (ei)	Aplikasi (oi)	(oi-ei)^2
1	16543,418	16543,486	2,79507E-07
2	3794,362	3794,266	2,42887E-06
3	4980,078	4979,952	3,1879E-06
4	2550,015	2550,015	0
5	0,512	0,512	0
6	8,515	7,812	0,058039812
7	0,512	0,512	0
8	98,272	98,272	0
9	39,934	39,934	0
10	-76,069	-76,069	0
11	98,272	98,272	0
12	24	24	0
13	-105,157	-105,157	0
14	3,01	3,01	0
15	14,771	14,771	0
16	-115,518	-115,518	0
17	152,718	152,718	0
18	139,418	139,418	0
19	99,092	99,092	0
20	40,343	40,343	0
21	-75,142	-75,142	0
22	99,092	99,092	0
23	55	55	0
24	-100,157	-100,157	0
25	3,01	3,01	0
26	2,833	2,833	0
27	-96,979	-96,98	-1,03115E-08
28	151,979	151,98	6,57986E-09
29	138,679	138,68	7,2109E-09
Tingkat Penyimpangan			**0,058045712**
AKURASI [%]			**94,19542881**

Tingkat penyimpangan & akurasi perhitungan perencanaan daerah Medan Perjuangan:

No	Manual (ei)	Aplikasi (oi)	(oi-ei)^2
1	12431,205	12431,212	3,94169E-09
2	944,62	944,605	2,38191E-07
3	1283,74	1283,719	3,43528E-07
4	2220,97	2220,97	0
5	1,731	1,731	0
6	7,607	7,51	0,001236887
7	0,942	0,942	0
8	1,045	1,045	0
9	136,525	136,526	7,32467E-09
10	21	21	0
11	-105,157	-105,157	0
12	3,01	3,01	0
13	17,782	17,782	0
14	-117,928	-117,929	-8,47975E-09
15	152,128	152,129	6,57341E-09
16	138,828	138,829	7,20316E-09
17	1,045	1,045	0
18	137,914	137,915	7,2509E-09
19	50	50	0
20	-100,157	-100,157	0
21	3,01	3,01	0
22	10	10	0
23	-102,346	-102,347	-9,77078E-09
24	152,346	152,347	6,56401E-09
25	139,046	139,047	7,19186E-09
Tingkat Penyimpangan			**0,001237497**
AKURASI [%]			**99,87625034**

Skenario tambahan perhitungan Lrts & Lms pada daerah dense urban dengan model propagasi Walfish-Ikegami:

UPLINK						
No	Lrts	Manual	Aplikasi	Lms	Manual	Aplikasi
1	0-35 (30)	39,934	39,934	hb>hr (hb=40,hr=35)	-76,069	-76,069
2	35-55 (40)	42,189	42,189	hb<hr & d>0,5 (hb=40,hr=45,d>0,512)	35,424	35,424
3	55-90 (60)	42,744	42,744	hb<hr & d<0,5 (hb=40,hr=45,d>0,3)	29,241	29,241
DOWNLINK						
No	Lrts	Manual	Aplikasi	Lms	Manual	Aplikasi
1	0-35 (30)	40,343	40,343	hb>hr (hb=40,hr=35)	-75,142	-75,142
2	35-55 (40)	42,598	42,598	hb<hr & d>0,5 (hb=40,hr=45,d>0,512)	36,351	36,351
3	55-90 (60)	43,153	43,153	hb<hr & d<0,5 (hb=40,hr=45,d>0,3)	30,169	30,169

LAMPIRAN F

SOURCE CODE

Source code MainActivity

```
package com.example.gwcdma;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.Window;
import android.widget.ImageButton;
import android.widget.LinearLayout;

public class MainActivity extends Activity {
  ImageButton about,how,peta,calculate,report,calculation_history;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
  requestWindowFeature(Window.FEATURE_NO_TITLE);
              super.onCreate(savedInstanceState);
              setContentView(R.layout.activity_main);
              LinearLayout ll =
(LinearLayout)findViewById(R.id.hilang);
              ll.setVisibility(View.GONE);

              String destPath = "/data/data/" + getPackageName() +
"/databases";
              File f = new File(destPath);
              try {

  CopyDB(getBaseContext().getAssets().open("db_adapter"),
                                          new
FileOutputStream(destPath + "/db_adapter"));
              } catch (FileNotFoundException e) {
                          // TODO Auto-generated catch block
                          e.printStackTrace();
              } catch (IOException e) {
                          // TODO Auto-generated catch block
                          e.printStackTrace();
              }

              about =(ImageButton)findViewById(R.id.about);
              how =(ImageButton)findViewById(R.id.how);
              calculate =(ImageButton)findViewById(R.id.calculate);
              peta =(ImageButton)findViewById(R.id.location);
              report =(ImageButton)findViewById(R.id.report);

  calculation_history=(ImageButton)findViewById(R.id.calculation_h
istory);

              about.setOnClickListener(new View.OnClickListener() {
                          @Override
                          public void onClick(View v) {
                                      // TODO Auto-generated method
stub
                                      Intent intent1=new
Intent(v.getContext(), about.class);
                                      startActivityForResult(intent1, 0);
                                      }
      });

              how.setOnClickListener(new View.OnClickListener() {
                          @Override
                          public void onClick(View v) {
                                      // TODO Auto-generated method
stub
                                      Intent intent1=new
Intent(v.getContext(), how.class);
                                      startActivityForResult(intent1, 0);
                                      }
      });

  peta.setOnClickListener(new View.OnClickListener() {
              @Override
              public void onClick(View v) {

  Intent intent1=new Intent(v.getContext(), location.class);
                                      startActivity(intent1);
                                      }
      });

              calculate.setOnClickListener(new
View.OnClickListener() {
                          @Override
                          public void onClick(View v) {
                                      // TODO Auto-generated method
stub
                                      Intent intent1=new
Intent(v.getContext(), user_prediction.class);
                                      startActivityForResult(intent1, 0);
                                      }
      });

              report.setOnClickListener(new View.OnClickListener() {
                          @Override
                          public void onClick(View v) {
                                      // TODO Auto-generated method
stub
                                      Intent intent1=new
Intent(v.getContext(), result.class);
                                      startActivityForResult(intent1, 0);
                                      }
      });

              calculation_history.setOnClickListener(new
View.OnClickListener() {
                          @Override
                          public void onClick(View v) {
                                      // TODO Auto-generated method
stub
                                      Intent intent1=new
Intent(v.getContext(), calchistory.class);
                                      startActivityForResult(intent1, 0);
                                      }
      });
  }

  public void CopyDB(InputStream inputStream, OutputStream
outputStream)
                          throws IOException {
              // --copy 1K bytes at a time
              byte[] buffer = new byte[1024];
              int length;
              while ((length = inputStream.read(buffer)) > 0) {
                          outputStream.write(buffer, 0, length);
                          Log.d("copy db", buffer.toString());
              }
              inputStream.close();
              outputStream.close();
  }

  @Override
  public boolean onCreateOptionsMenu(Menu menu) {
              // Inflate the menu; this adds items to the action bar if it is
present.
              getMenuInflater().inflate(R.menu.main, menu);
              return true;
  }
}
```

Source code User_Prediction

```
package com.example.gwcdma;

import java.text.DecimalFormat;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.Toast;

public class user_prediction extends Activity  {
   private EditText user,tahun,fp,usia,jlhuser,operator;
   DecimalFormat df;
   Button hitung,next,back;
   String kata1,wide_area,type_of_area;
   database2 db;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
   requestWindowFeature(Window.FEATURE_NO_TITLE);
   super.onCreate(savedInstanceState);
   setContentView(R.layout.user_prediction);
   LinearLayout ll = (LinearLayout)findViewById(R.id.hilang);
   ll.setVisibility(View.GONE);

   next    = (Button) findViewById(R.id.next);
   back    = (Button) findViewById(R.id.back);
   user    = (EditText) findViewById(R.id.user);
   tahun   = (EditText) findViewById(R.id.tahun);
   fp      = (EditText) findViewById(R.id.fp);
   usia    = (EditText) findViewById(R.id.usia_productive);
   operator = (EditText) findViewById(R.id.operator);
   jlhuser = (EditText) findViewById(R.id.jlhuser);
   hitung = (Button) findViewById(R.id.hitung);

   user.setText(getIntent().getStringExtra("population"));
   df = new DecimalFormat(".###");
   db = new database2(this);
   db.open();

   hitung.setOnClickListener(new OnClickListener() {
   public void onClick(View v) {
   if (user.getText().toString().equals("") ||
tahun.getText().toString().equals("")|| fp.getText().toString().equals("")
|| usia.getText().toString().equals("")||
operator.getText().toString().equals("")){

AlertDialog alert = new
AlertDialog.Builder(user_prediction.this).create();
alert.setTitle("Warning");
alert.setMessage("Please Fill All Form");
alert.setButton("Close", new DialogInterface.OnClickListener() {
@Override
   public void onClick(DialogInterface arg0, int arg1) {
} });
   alert.show();
                                  }
   else {
   String a = usia.getText().toString();
   String b = operator.getText().toString();
   int aa=Integer.parseInt(a);
   int bb=Integer.parseInt(b);
   if ((aa<=100 && aa>=0) && (bb<=100 && bb>=0)){

   jlhuser.setText(String.valueOf (df.format
   (( (Math.pow ( (1+(0.01 *
Double.parseDouble(fp.getText().toString()))),
(Double.parseDouble(tahun.getText().toString())) ) ) *
(Double.parseDouble(user.getText().toString())) )
* (Double.parseDouble(usia.getText().toString())/100) *
(Double.parseDouble(operator.getText().toString())/100))
));
   }else{

   if ((aa>100 && bb<=100) ){
   usia.setError("Error");
   jlhuser.setText("");
    Toast.makeText(user_prediction.this, "Check value of Productive
Age (value=0-100 %)", Toast.LENGTH_LONG).show();
               usia.requestFocus();}
   if ((bb>100 && aa<=100)){
   operator.setError("Error");
   jlhuser.setText("");
   Toast.makeText(user_prediction.this, "Check value of Potential
Operator(value=0-100 %)", Toast.LENGTH_LONG).show();
                           operator.requestFocus();}
                            if((aa>100 && bb>100)){
                                          usia.setError("Error");
                                          operator.setError("Error");
                                                jlhuser.setText("");
Toast.makeText(user_prediction.this, "Check value of Productive Age
& Potential Operator(value=0-100 %)",
Toast.LENGTH_LONG).show();
usia.requestFocus();
operator.requestFocus();}
                            }}      }});

   next.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View v) {

if (jlhuser.getText().toString().equals("") ){
AlertDialog alert = new
AlertDialog.Builder(user_prediction.this).create();
alert.setTitle("Warning");
alert.setMessage("Please Calculate User Prediction (Un)");
alert.setButton("Close", new DialogInterface.OnClickListener() {

@Override
   public void onClick(DialogInterface arg0, int arg1) {

   }
                                                              });
   alert.show();
                            }
                            else{
   String kata1 = user.getText().toString();
   String kata2 = tahun.getText().toString();
   String kata3 = jlhuser.getText().toString();
   double parameter[] =
{Double.parseDouble(kata1),Double.parseDouble(kata2),Double.parse
Double(kata3),0,0,0,0,0,0,0};
String param2[] = {"", ""};
long id = db.insertCalculation(parameter, param2);
Toast.makeText(getBaseContext(), String.valueOf(id),
Toast.LENGTH_SHORT).show();

Intent myintent = new Intent(user_prediction.this, obq.class);
myintent.putExtra("id", String.valueOf(id));
startActivity(myintent);
}
   }});
   back.setOnClickListener(new OnClickListener() {
   @Override
   public void onClick(View v) {
   finish();
}});}          }
```

Source code OBQ

```
package com.example.gwcdma;

import java.text.DecimalFormat;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.Window;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.TextView;

public class obq extends Activity  {
   database2 db;
   Button next,back,userkm2,OBQ1,OBQ2;
   EditText luas,user,totuser,OBQtot1,OBQtot2;
   DecimalFormat df;
   TextView a,b,c,d,e,f;
   CheckBox
checboxS,checboxSM,checboxSD,checboxMMM,checboxHMM,checboxHIMM;
   String service ="" ;
   int bitrate = 0;
   double bitrate1= 0;
   double EbNo = 0;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
   requestWindowFeature(Window.FEATURE_NO_TITLE);
   super.onCreate(savedInstanceState);
   setContentView(R.layout.obq);
   LinearLayout ll = (LinearLayout)findViewById(R.id.hilang);
   ll.setVisibility(View.GONE);
db= new database2(this);
db.open();
luas   = (EditText)findViewById(R.id.luas);
OBQtot1= (EditText)findViewById(R.id.OBQtot1);
OBQtot2= (EditText)findViewById(R.id.OBQtot2);
totuser= (EditText)findViewById(R.id.totuserkm2);
next   = (Button) findViewById(R.id.next);
back   = (Button) findViewById(R.id.back);
userkm2= (Button)findViewById(R.id.userkm2);
OBQ1   = (Button)findViewById(R.id.OBQ1);
OBQ2   = (Button)findViewById(R.id.OBQ2);
a = (TextView) findViewById(R.id.a);
b = (TextView) findViewById(R.id.b);
c = (TextView) findViewById(R.id.c);
d = (TextView) findViewById(R.id.d);
e = (TextView) findViewById(R.id.e);
f = (TextView) findViewById(R.id.f);
checboxS = (CheckBox) findViewById(R.id.checkBox1);
checboxSM = (CheckBox) findViewById(R.id.checkBox2);
checboxSD = (CheckBox) findViewById(R.id.checkBox3);
checboxMMM = (CheckBox) findViewById(R.id.checkBox4);
checboxHMM = (CheckBox) findViewById(R.id.checkBox5);
checboxHIMM = (CheckBox) findViewById(R.id.checkBox6);

luas.setText(getIntent().getStringExtra("luas_area"));
df = new DecimalFormat(".###");
db = new database2(this);
db.open();
userkm2.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
  if (luas.getText().toString().equals("")){
AlertDialog alert = new AlertDialog.Builder(obq.this).create();

  alert.setTitle("Warning");
  alert.setMessage("Please Fill Form Wide");
.setButton("Close", new DialogInterface.OnClickListener() {
  @Override
  public void onClick(DialogInterface arg0, int arg1) {
  }});
alert.show();
totuser.setText("");
                                          }
else{
int id = Integer.parseInt(getIntent().getStringExtra("id"));
Cursor c = db.getdata(id);
c.moveToNext();
double user_prediction =
c.getInt(c.getColumnIndex("user_prediction"));
  totuser.setText(String.valueOf (df.format (user_prediction /
  Double.parseDouble(luas.getText().toString()) )));
  }
  a.setText("0");
  b.setText("0");
  c.setText("0");
  d.setText("0");
  e.setText("0");
  f.setText("0");
        }});

  checboxS.setOnClickListener(new OnClickListener() {
  public void onClick(View v) {
  if ( totuser.getText().toString().equals("") ){
  AlertDialog alert = new AlertDialog.Builder(obq.this).create();
  alert.setTitle("Warning");
  alert.setMessage("Please Calculate User Density");
  alert.setButton("Close", new DialogInterface.OnClickListener() {

  @Override
  public void onClick(DialogInterface arg0, int arg1) {
  }});
  alert.show();
  }
  else{
  if (((CheckBox) v).isChecked()){
a.setText(String.valueOf(df.format( ((0.5*Double.parseDouble(totuser.
getText().toString()))*16 * 0.73 * 3 * 180 ) +
((0.3*Double.parseDouble(totuser.getText().toString()))*16 * 0.73 *
0.8 * 120 ) +
((0.2*Double.parseDouble(totuser.getText().toString()))*16 * 0.73 *
0.4 * 120 ) )));
service+="S,";
if (bitrate < 16){
bitrate= 16; bitrate1= 16;
EbNo= 2.5; }}
else{
a.setText("0");
  }} }});
  checboxSM.setOnClickListener(new OnClickListener() {
  public void onClick(View v) {
  if ( totuser.getText().toString().equals("") ){

AlertDialog alert = new AlertDialog.Builder(obq.this).create();
alert.setTitle("Warning");
alert.setMessage("Please Calculate User Density");
alert.setButton("Close", new DialogInterface.OnClickListener() {
  @Override
  public void onClick(DialogInterface arg0, int arg1) {
  }});
alert.show();
}
  else{
if (((CheckBox) v).isChecked()){
b.setText(String.valueOf(df.format( ((0.5*Double.parseDouble(totuser
.getText().toString()))*14 * 0.4 * 0.6 * 3 ) +
((0.3*Double.parseDouble(totuser.getText().toString()))*14 * 0.4 * 0.3
* 3 ) +((0.2*Double.parseDouble(totuser.getText().toString()))*14 *
0.4 * 0.2 * 3 ) )));
service+="SM,";
if (bitrate < 14)
bitrate= 14; bitrate1= 14; EbNo= 3; }
}
else{
b.setText("0");
} }
   }});

checboxSD.setOnClickListener(new OnClickListener() {
 public void onClick(View v) {
if ( totuser.getText().toString().equals("") ){
AlertDialog alert = new AlertDialog.Builder(obq.this).create();
alert.setTitle("Warning");
alert.setMessage("Please Calculate User Density");
alert.setButton("Close", new DialogInterface.OnClickListener() {

   @Override
   public void onClick(DialogInterface arg0, int arg1) {
   }});
alert.show();}
else{
   if (((CheckBox) v).isChecked()){
   c.setText(String.valueOf(df.format( ((0.5*Double.parseDouble(totus
er.getText().toString()))*64 * 0.13 * 0.2 * 156 ) +
((0.3*Double.parseDouble(totuser.getText().toString()))*64 * 0.13 *
0.2 * 156 ) +
((0.2*Double.parseDouble(totuser.getText().toString()))*64 * 0.13 *
0.02 * 156 ) )));
service+="SD,";

if (bitrate < 64)
{bitrate= 64; bitrate1= 64;
EbNo= 2; }}
else{
c.setText("0");
                    }
   }}});

checboxMMM.setOnClickListener(new OnClickListener() {
   public void onClick(View v) {
   if ( totuser.getText().toString().equals("") ){
   AlertDialog alert = new AlertDialog.Builder(obq.this).create();
   alert.setTitle("Warning");
   alert.setMessage("Please Calculate User Density");
   alert.setButton("Close", new DialogInterface.OnClickListener() {
   @Override
   public void onClick(DialogInterface arg0, int arg1) {
   }});
   alert.show();
}
   else{
if (((CheckBox) v).isChecked()){
   d.setText(String.valueOf(df.format( ((0.5*Double.parseDouble(totus
er.getText().toString()))*64 * 0.15 * 0.5 * 300 ) +
   ((0.3*Double.parseDouble(totuser.getText().toString()))*64 * 0.15 *
0.4 * 300 ) +
((0.2*Double.parseDouble(totuser.getText().toString()))*64 * 0.15 *
0.008 * 300 ) )));
   service+="MMM,";
if (bitrate <= 64)
   {bitrate= 64; bitrate1= 384;
EbNo= 2;  }
                                                                        }
   else{
   d.setText("0");
   }}}});

checboxHMM.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
```

```
if ( totuser.getText().toString().equals("") ){
   AlertDialog alert = new AlertDialog.Builder(obq.this).create();
   alert.setTitle("Warning");
   alert.setMessage("Please Calculate User Density");
   alert.setButton("Close", new DialogInterface.OnClickListener() {
   @Override
   public void onClick(DialogInterface arg0, int arg1) {
   }});
   alert.show();
   }else{
   if (((CheckBox) v).isChecked()){
   e.setText(String.valueOf(df.format( ((0.5*Double.parseDouble(totus
er.getText().toString()))*128 * 0.15 * 0.15 * 300 ) +
   ((0.3*Double.parseDouble(totuser.getText().toString()))*128 * 0.15
* 0.06 * 300 ) +
   ((0.2*Double.parseDouble(totuser.getText().toString()))*128 * 0.15
* 0.008 * 300 ) )));
 service+="HMM,";

if (bitrate <= 128)
{bitrate= 128; bitrate1= 2000;
 EbNo= 1.4;}}
else{
e.setText("0");
   }}}});

   checboxHIMM.setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
   if ( totuser.getText().toString().equals("") ){
   AlertDialog alert = new AlertDialog.Builder(obq.this).create();
   alert.setTitle("Warning");
   alert.setMessage("Please Calculate User Density");
   alert.setButton("Close", new DialogInterface.OnClickListener() {
   @Override
   public void onClick(DialogInterface arg0, int arg1) {
   }});
   alert.show();
                                   }
                                   else{
   if (((CheckBox) v).isChecked()){
   f.setText(String.valueOf(df.format( ((0.5*Double.parseDouble(totus
er.getText().toString()))*128 * 0.25 * 0.14 * 120 ) +
   ((0.3*Double.parseDouble(totuser.getText().toString()))*128 * 0.25
* 0.07 * 120 ) +
   ((0.2*Double.parseDouble(totuser.getText().toString()))*128 * 0.25
* 0.011 * 120 ) )));

   service+="HIMM";
if (bitrate < 128)
{bitrate= 128; bitrate1= 128;
   EbNo= 1.4;} }
                 else{
   f.setText("0");
   }}}});

OBQ1.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
if ( totuser.getText().toString().equals("") ||
a.getText().toString().equals("") ||
 (a.getText().toString().equals("0")&&
b.getText().toString().equals("0") &&
c.getText().toString().equals("0") &&
d.getText().toString().equals("0") &&
e.getText().toString().equals("0") &&
f.getText().toString().equals("0") )){

AlertDialog alert = new AlertDialog.Builder(obq.this).create();
alert.setTitle("Warning");
alert.setMessage("Please Calculate User Density and Check Service
Planning");alert.setButton("Close", new
DialogInterface.OnClickListener() {
   @Override
   public void onClick(DialogInterface arg0, int arg1) {
  }});
alert.show();}
else{          OBQtot1.setText(String.valueOf(df.format( Double.parse
Double(a.getText().toString())+Double.parseDouble(b.getText().toStri
ng())
+Double.parseDouble(c.getText().toString())+Double.parseDouble(d.g
etText().toString()) +Double.parseDouble(e.getText().toString())+
  Double.parseDouble(f.getText().toString()) )));
                    }}});


OBQ2.setOnClickListener(new OnClickListener() {
public void onClick(View v) { if
(OBQtot1.getText().toString().equals("") ||
totuser.getText().toString().equals("")){
AlertDialog alert = new AlertDialog.Builder(obq.this).create();
  alert.setTitle("Warning");
  alert.setMessage("Please Calculate OBQ [Kb/hour/km2]");
                                        alert.setButton("Close", new
DialogInterface.OnClickListener() {
@Override
  public void onClick(DialogInterface arg0, int arg1) {
}});
  alert.show();             }
  else{
OBQtot2.setText(String.valueOf(df.format
(Double.parseDouble(OBQtot1.getText().toString()) / 3600 )));
}}});

next.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
if (totuser.getText().toString().equals("") ||
OBQtot1.getText().toString().equals("")
||OBQtot2.getText().toString().equals("")){
AlertDialog alert = new AlertDialog.Builder(obq.this).create();
  alert.setTitle("Warning");
  alert.setMessage("Please Fill All Form");
  alert.setButton("Close", new DialogInterface.OnClickListener() {
@Override
  public void onClick(DialogInterface arg0, int arg1) {
}});
  alert.show();
  }
else{
 String kata1 = luas.getText().toString();
  String kata2 = totuser.getText().toString();
  String kata3 = OBQtot2.getText().toString();
  String kata4 = service;
Intent intent =getIntent();
int id = Integer.parseInt(intent.getStringExtra("id"));
int affectedrow = db.updatecalculation("wide_area", kata1, id);
int affected    = db.updatecalculation("user_density", kata2, id);
int affecte    = db.updatecalculation("obq", kata3,
id);db.updatecalculation("service", kata4, id);

Intent intent1=new Intent(v.getContext(), cell_dimensioning.class);
intent1.putExtra("id", String.valueOf(id));
  intent1.putExtra("bitrate", String.valueOf(bitrate));
  intent1.putExtra("bitrate1", String.valueOf(bitrate1));
  intent1.putExtra("EbNo", String.valueOf(EbNo));
  startActivity(intent1);
  }}});
back.setOnClickListener(new OnClickListener() {
  @Override
  public void onClick(View v) {
  finish();}});}
@Override
public void onResume() {
                              super.onResume();
                              bitrate = 0;
                              EbNo = 0;
                 }}
```

Source code database2

```
package com.example.gwcdma;

import org.apache.http.impl.entity.StrictContentLengthStrategy;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

public class database2 {
  static final String DATABASE_NAME = "db_adapter";
  private String DATABASE_TABLE;
  static final int DATABASE_VERSION = 1;
  final Context context;
  DatabaseHelper DBHelper;
  SQLiteDatabase db;
  public database2(Context ctx) {
  this.context = ctx;
  DBHelper = new DatabaseHelper(context);}
  private static class DatabaseHelper extends SQLiteOpenHelper {
  DatabaseHelper(Context context) {
  super(context, DATABASE_NAME,
null,DATABASE_VERSION);}
public void onCreate(SQLiteDatabase db) {
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int
newVersion) {                        }}
  public database2 open() throws SQLException {
  db = DBHelper.getWritableDatabase();
  return this;
  }
  public void close() {
  DBHelper.close();
  }
  public boolean checkDataBase() {SQLiteDatabase checkDB = null;
  try {
checkDB =
SQLiteDatabase.openDatabase("/data/data/com.example.gwcdma/data
bases", null, SQLiteDatabase.OPEN_READONLY);
} catch (SQLiteException e) {
  Log.d("check db", "Database doesnt exist yet");
                   }
  if (checkDB != null) {Log.d("check db", "Database  exist ");}
  return checkDB != null ? true : false;
  }

  public Cursor getAllpenetrasi(){
  Cursor cursor = db.rawQuery("SELECT * FROM penetrasi_rate
WHERE _id=1",null);return cursor;
}

public Cursor getAllcoba(){Cursor cursor = db.rawQuery("SELECT
CBD FROM penetrasi_rate ",null); return cursor;}
  public Cursor getS_B(){
Cursor a = db.rawQuery("SELECT building FROM call_duration
WHERE service=S",null);return a;}
  public long insertCalculation (double[] param2, String[] param){
Cursor cursor =db.rawQuery("SELECT MAX(`_id`)as `_id`,
`population` FROM calculation_coverage", new String[] {});
cursor.moveToNext();
long index = cursor.getLong(cursor.getColumnIndex("_id"));
long user = cursor.getLong(cursor.getColumnIndex("population"));
int idx;
if (user == 0 && index == 1){
idx =1 ;
db.delete("calculation_coverage", "_id = ?",new String[]
{Long.toString(idx)});}else{
idx =(int) index +1;
                }
ContentValues initValues = new ContentValues();
initValues.put("_id", idx);
initValues.put("population", param2[0]);
initValues.put("prediction_years", param2[1]);
initValues.put("user_prediction", param2[2]);
initValues.put("wide_area", param2[3]);
initValues.put("user_density", param2[4]);
initValues.put("obq", param2[5]);
initValues.put("cell_capacity", param2[6]);
initValues.put("cell_area", param2[7]);
initValues.put("total_cell", param2[8]);
initValues.put("radius_cell", param2[9]);
initValues.put("service", param[0]);
initValues.put("type_area", param[1]);
long id= db.insert("calculation_coverage", null, initValues);
return id;
  }
  public Cursor getAllresult (){
  Cursor cursor =db.rawQuery("SELECT * FROM
calculation_coverage", new String[] {} );
return cursor;
  }
public int updatecalculation (String col,String val, int id){
ContentValues init = new ContentValues();
init.put(col, val);
int affected = db.update("calculation_coverage", init, "_id = ?", new
String[] {String.valueOf(id)});
return affected;
  }
  public Cursor getdata (int id){
  Cursor c = db.rawQuery("SELECT * FROM calculation_coverage
WHERE _id = ?", new String[] {String.valueOf(id)});
                return c;
  }

public long insertCalculation1 (double[] param2, String[] param){
Cursor cursor =db.rawQuery("SELECT MAX(`_id`)as `_id`,
`high_bts` FROM calculation_link_budget", new String[] {});
cursor.moveToNext();
long index = cursor.getLong(cursor.getColumnIndex("_id"));
long user = cursor.getLong(cursor.getColumnIndex("high_bts"));
int idx;
 if (user == 0 && index == 1){
idx =1 ;
db.delete("calculation_link_budget", "_id = ?",new String[]
{Long.toString(idx)}); }else{
idx =(int) index +1;}
                ContentValues initValues = new ContentValues();
                initValues.put("_id", idx);
                initValues.put("high_bts", param2[0]);
                initValues.put("frequency_ul", param2[1]);
                initValues.put("pathloss_ul", param2[2]);
                initValues.put("power_ul", param2[3]);
                initValues.put("maxPL_ul", param2[4]);
                initValues.put("frequency_dl", param2[5]);
                initValues.put("pathloss_dl", param2[6]);
                initValues.put("power_dl", param2[7]);
                initValues.put("maxPL_dl", param2[8]);
                initValues.put("eksekusi1", param[0]);
                initValues.put("eksekusi2", param[1]);
  long id= db.insert("calculation_link_budget", null, initValues);
  return id;
  }

  public Cursor getAllresult1 (){
Cursor cursor =db.rawQuery("SELECT * FROM
calculation_link_budget", new String[] {} ); return cursor;}
public int updatecalculation1 (String col,String val, int id){
  ContentValues init = new ContentValues();
  init.put(col, val);
  int affected = db.update("calculation_link_budget", init, "_id = ?",
new String[] {String.valueOf(id)});
  return affected; }

  public Cursor getdata1 (int id){
  Cursor c = db.rawQuery("SELECT * FROM
calculation_link_budget WHERE _id = ?", new String[]
{String.valueOf(id)});
                return c;
  }

  public void delete1 (int id){db.delete("calculation_coverage","_id =
" +id, null); }
  public void delete2 (int id){
  db.delete("calculation_link_budget","_id = " +id, null);
  }
}
```

Source code AndroidManifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.gwcdma"
   android:versionCode="1"
   android:versionName="1.0" >

   <uses-sdk
      android:minSdkVersion="12"
      android:targetSdkVersion="17" />

   <uses-permission
android:name="com.example.wcdma3g_new.googlemapsv2.permis
sion.MAPS_RECEIVE" />
   <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"
/>
   <uses-permission
android:name="android.permission.INTERNET" />
   <uses-permission
android:name="com.google.android.providers.gsf.permission.REA
D_GSERVICES" />
   <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAG
E" />

   <!-- Required to show current location -->
   <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATIO
N" />
   <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"
/>

   <!-- Required OpenGL ES 2.0. for Maps V2 -->
   <uses-feature
      android:glEsVersion="0x00020000"
      android:required="true" />

   <application
      android:allowBackup="true"
      android:icon="@drawable/ic_launcher"
      android:label="@string/app_name"
      android:theme="@style/AppTheme" >
      <activity
         android:name="com.example.gwcdma.MainActivity"
         android:label="@string/app_name" >
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category
android:name="android.intent.category.LAUNCHER" />
         </intent-filter>
      </activity>
      <activity
         android:name="com.example.gwcdma.about"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.how"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.how_coverage"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.how_pathloss"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.how_link_budget"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.location"
         android:label="@string/app_name"
android:screenOrientation="portrait">
      </activity>
      <activity
         android:name="com.example.gwcdma.user_prediction"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.obq"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.cell_dimensioning"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.pathloss1_uplink"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.pathloss2_uplink"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.pathloss1_downlink"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.pathloss2_downlink"
         android:label="@string/app_name" >
      </activity>
      <activity

android:name="com.example.gwcdma.reverse_link_budget"
         android:label="@string/app_name" >
      </activity>
      <activity

android:name="com.example.gwcdma.forward_link_budget"
         android:label="@string/app_name" >
      </activity>
       <activity
         android:name="com.example.gwcdma.calchistory"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.history_coverage"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.history_link_budget"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.result"
         android:label="@string/app_name" >
      </activity>
      <activity
         android:name="com.example.gwcdma.result_coverage"
         android:label="@string/app_name" >
      </activity>
       <activity
         android:name="com.example.gwcdma.result_link_budget"
         android:label="@string/app_name" >
      </activity>



      <meta-data
         android:name="com.google.android.maps.v2.API_KEY"

android:value="AIzaSyAw_JjLwGzlbwI6Y4LrFm2jw1J0qGSxCiE"
/>
      <meta-data
         android:name="com.google.android.gms.version"
         android:value="@integer/google_play_services_version"
/>

      <activity
         android:name="com.example.gwcdma.CobaActivity"
         android:label="@string/title_activity_coba" >
      </activity>
   </application>

</manifest>
```