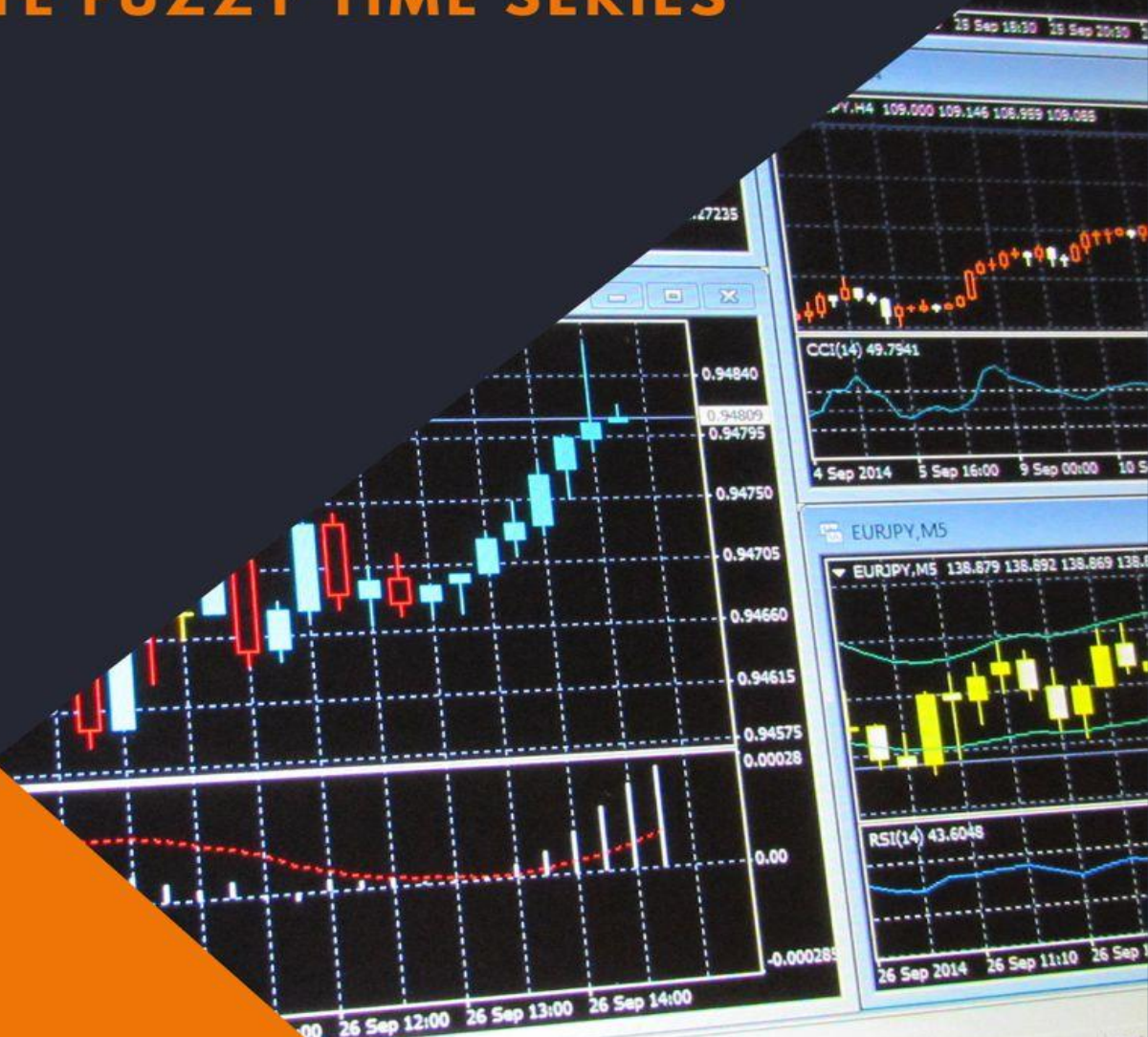


PANDUAN PENGUNAAN

WEBSITE FUZZY TIME SERIES
CHEN

OLEH
IMRAN PARIAMAN TANJUNG
M. YOKA FATHONI, S.KOM., M.KOM
DEDY AGUNG PRABOWO, S.KOM., M.KOM



DAFTAR ISI

DAFTAR ISI.....	2
DAFTAR GAMBAR	4
DAFTAR KODE	5
1. Pendahuluan.....	6
1.1 Tujuan Pembuatan Dokumen	6
2. Deskripsi Website.....	6
3. Petunjuk Penggunaan	6
3.1. Halaman Utama.....	6
3.2. Halaman Data Aktual.....	7
3.3. Halaman Tambah Data Aktual.....	8
3.4. Halaman Kategori <i>Universe of Discourse & Re-Divide Interval</i>	8
3.5. Halaman Kategori <i>Fuzzy Logical Relations</i>	10
3.6. Halaman Kategori <i>Fuzzy Logical Relation Group</i>	11
3.7. Halaman Data Hasil Prediksi	12
3.8. Halaman Grafik Hasil Prediks.....	14
4. Source Code Website Fuzzy Time Series-Chen.....	17
4.1. Dashboard Controller	17
4.2. Data Aktual	17
4.3. Tambah Data Aktual	17
4.4. Edit Data Aktual.....	18
4.5. Hapus Data Aktual	18
4.6. Kategori <i>Universe of Discourse & Re-Divide Interval</i>	19
4.7. Kategori <i>Fuzzy Logical Relations</i>	20
4.8. Kategori <i>Fuzzy Logical Relation Group</i>	20
4.9. Data Hasil Prediksi.....	21
4.10. Grafik Hasil Prediksi	21

DAFTAR GAMBAR

Gambar 1 Halaman Utama.....	7
Gambar 2 Halaman Data Aktual.....	7
Gambar 3 Halaman Tambah Data Aktual.....	8
Gambar 4 Halaman Kategori <i>Universe of Discourse & Re-Divide Interval</i>	9
Gambar 5 Halaman Kategori <i>Fuzzy Logical Relations</i>	9
Gambar 6 Halaman Kategori <i>Fuzzy Logical Relation Group</i>	10
Gambar 7 Halaman Data Hasil Prediksi	11
Gambar 8 Halaman Grafik Hasil Prediksi	12

DAFTAR KODE

Kode 1 Dashboard Controller	17
Kode 2 Data Aktual	17
Kode 3 Tambah Data Aktual	17
Kode 4 Edit Data Aktual.....	18
Kode 5 Hapus Data Aktual	18
Kode 6 Kategori <i>Universe of Discourse & Re-Divide Interval</i>	19
Kode 7 Kategori <i>Fuzzy Logical Relations</i>	20
Kode 8 Kategori <i>Fuzzy Logical Relation Group</i>	20
Kode 9 Data Hasil Prediksi.....	21
Kode 10 Grafik Hasil Prediksi	21

1. Pendahuluan

1.1 Tujuan Pembuatan Dokumen

Dokumen *User Manual Website Fuzzy Time Series-Chen* ini dibuat untuk tujuan sebagai berikut:

- Menggambarkan dan menjelaskan penggunaan *User Manual Website Fuzzy Time Series-Chen* untuk *Admin*.
- Sebagai panduan, konfigurasi dan penggunaan *User Manual Website Fuzzy Time Series-Chen*.

Pihak-pihak yang berkepentingan dan berhak menggunakan dokumen ini adalah *Admin*, merupakan pengguna yang dapat mengakses, menambah, merubah dan menghapus data.

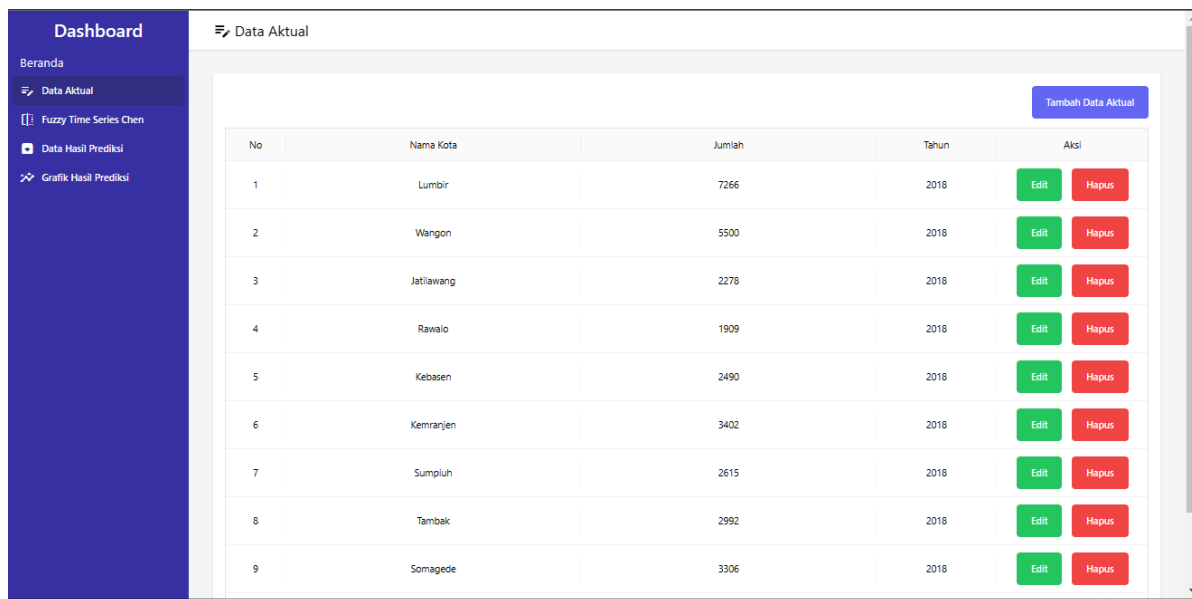
2. Deskripsi Website

User Manual Website Fuzzy Time Series-Chen merupakan sebuah sistem informasi Peramalan Lingkungan. Tujuan sistem informasi ini adalah untuk meramal, mengelola perluasan dan pengecilan ruang terbuka hijau di Kota Banyumas.

3. Petunjuk Penggunaan

3.1. Halaman Utama

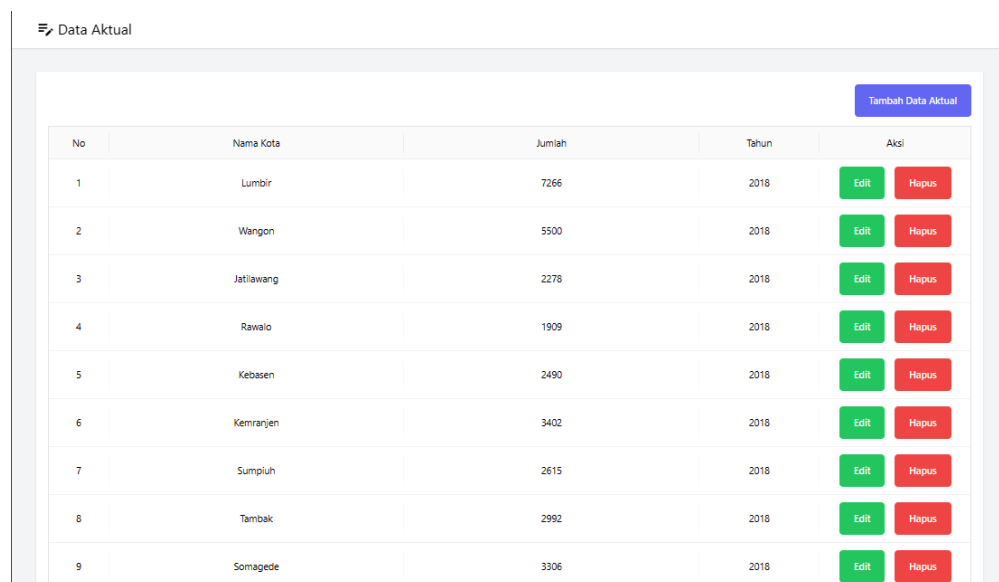
Pada Gambar 1 tampilan halaman utama *User Manual Website Fuzzy Time Series-Chen* dapat diakses melalui url <http://13.212.230.208> pengguna juga dapat mengakses melalui *localhost*.



Gambar 1 Halaman Utama

3.2. Halaman Data Aktual

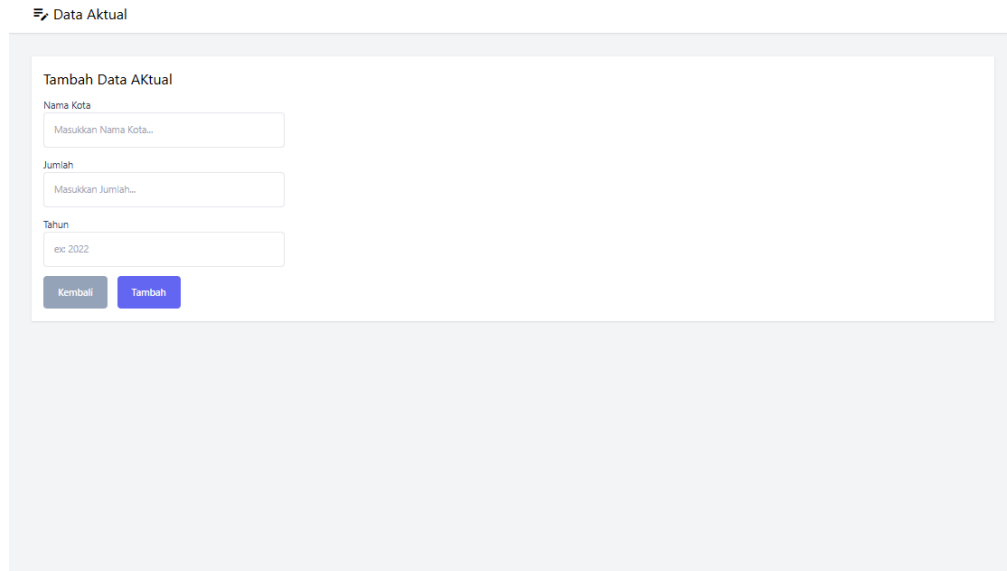
Pada Gambar 2 tampilan halaman Data Aktual yang digunakan oleh admin untuk dapat mengakases data, mengedit dan menambahkan data.



Gambar 2 Halaman Data Aktual

3.3. Halaman Tambah Data Aktual

Pada Gambar 3 tampilan halaman Tambah Data Aktual berisi *form* mengenai nama kota yang akan dimasukkan, jumlah luasan, dan tahun



☰ Data Aktual

Tambah Data Aktual

Nama Kota
Masukkan Nama Kota...

Jumlah
Masukkan Jumlah...

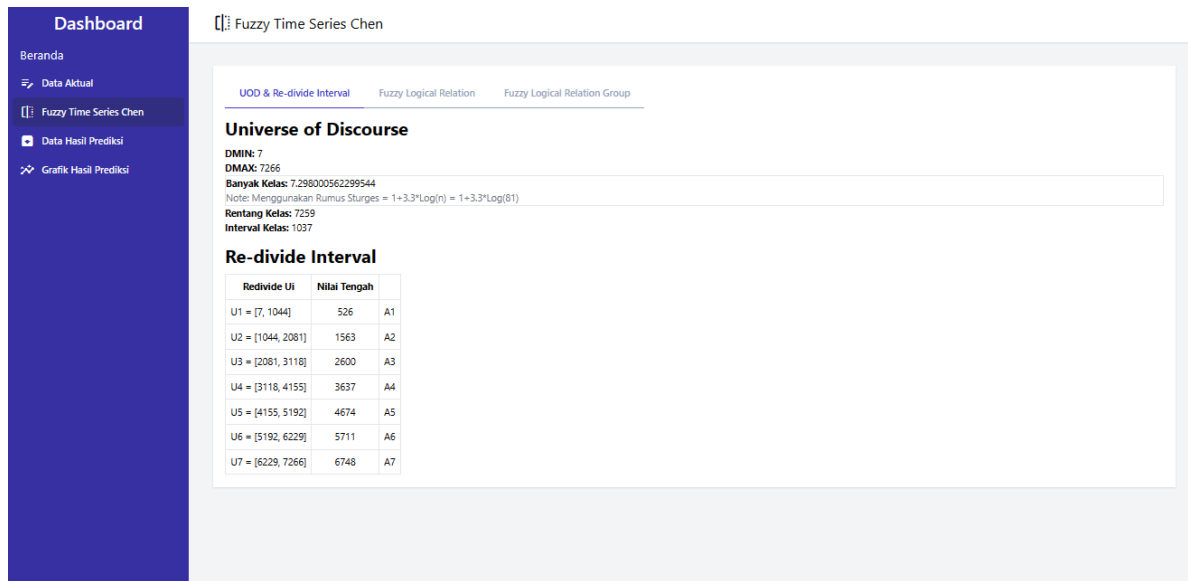
Tahun
ex: 2022

Kembali Tambah

Gambar 3 Halaman Tambah Data Aktual

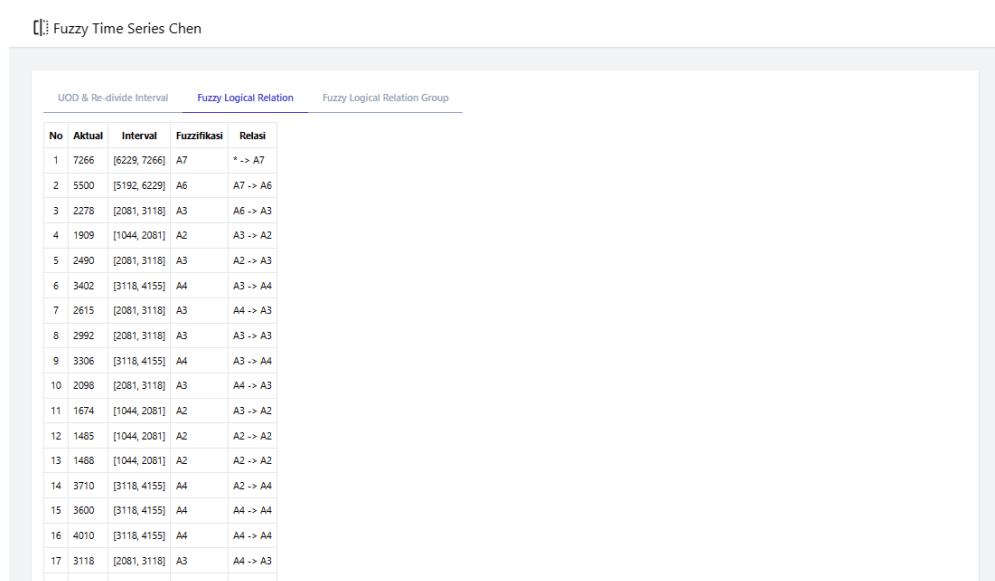
3.4. Halaman *Fuzzy Time Series-Chen*

Pada Gambar 4 tampilan halaman *fuzzy time series-chen* berisi tabel daftar data *universe of discour & re-divide interval*, *fuzzy logical relation*, dan *fuzzy logical relation group* yang dapat ditampilkan bergantian.



Gambar 4 Halaman Kategori *Universe of Discourse & re-divide interval*

Pada Gambar 5 tampilan halaman kategori *fuzzy logical relation* untuk menampilkan hasil perhitungan menggunakan *fuzzy logical relation*.



Gambar 5 Halaman Kategori *Fuzzy Logical Relation*

Pada Gambar 6 tampilan halaman *fuzzy logical relation group*, untuk menampilkan kategori kategori *fuzzy logical relation group* yang diperoleh dari hasil perhitungan FLR

Fuzzy Time Series Chen

UOD & Re-divide Interval Fuzzy Logical Relation **Fuzzy Logical Relation Group**

Group Relasi	Ramalan
G1 -> A4, A1	2082
G2 -> A3, A2, A4, A1	2082
G3 -> A2, A4, A3	2600
G4 -> A3, A4, A6, A5	4156
G5 -> A3, A1	1563
G6 -> A3, A4	3119
G7 -> A6	5711

Gambar 6 Halaman *Fuzzy Logical Relation Group*

3.5. Halaman Data Hasil Prediksi

Pada Gambar 7 tampilan halaman data hasil prediksi berisi hasil peramalan menggunakan *fuzzy time series-chen* yang dapat menampilkan data kota, data aktual, interval antar data, fuzzifikasi, relasi, *fuzzy logical relation group* dan ramalannya.

Dashboard

Beranda

- Data Aktual
- Fuzzy Time Series Chen
- Data Hasil Prediksi**
- Grafik Hasil Prediksi

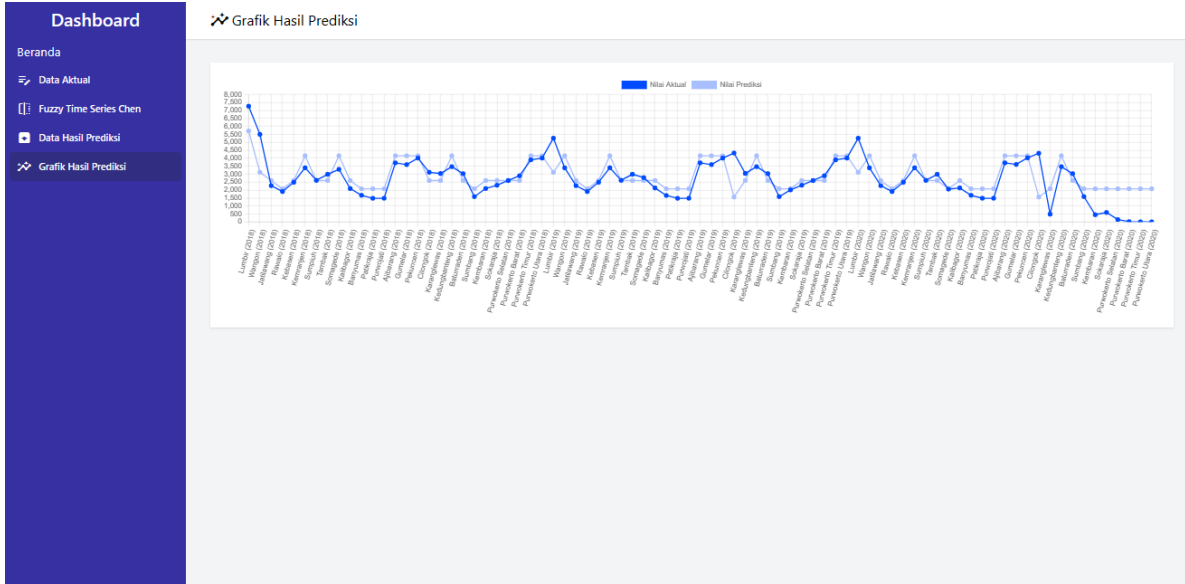
Data Hasil Prediksi

No	Kota	Aktual	Interval	Fuzzifikasi	Relasi	FLRG	Ramalan Fuzzy Chen
1	Lumbir	7266	[6229, 7266]	A7	* -> A7	G7	5711
2	Wangon	5500	[5192, 6229]	A6	A7 -> A6	G6	3119
3	Jatilawang	2278	[2081, 3118]	A3	A6 -> A3	G3	2600
4	Rawalo	1909	[1044, 2081]	A2	A3 -> A2	G2	2082
5	Kebasen	2490	[2081, 3118]	A3	A2 -> A3	G3	2600
6	Kemranjen	3402	[3118, 4155]	A4	A3 -> A4	G4	4156
7	Sumpiuh	2615	[2081, 3118]	A3	A4 -> A3	G3	2600
8	Tambak	2992	[2081, 3118]	A3	A3 -> A3	G3	2600
9	Somagede	3306	[3118, 4155]	A4	A3 -> A4	G4	4156
10	Kalibagor	2098	[2081, 3118]	A3	A4 -> A3	G3	2600
11	Banyumas	1674	[1044, 2081]	A2	A3 -> A2	G2	2082
12	Patikraja	1485	[1044, 2081]	A2	A2 -> A2	G2	2082
13	Purwojati	1488	[1044, 2081]	A2	A2 -> A2	G2	2082
14	Ajibarang	3710	[3118, 4155]	A4	A2 -> A4	G4	4156
15	Gumelar	3600	[3118, 4155]	A4	A4 -> A4	G4	4156
16	Pekuncen	4010	[3118, 4155]	A4	A4 -> A4	G4	4156
17	Cilongok	3118	[2081, 3118]	A3	A4 -> A3	G3	2600
18	Karanglewas	3040	[2081, 3118]	A3	A3 -> A3	G3	2600
19	Kedungbanteng	3471	[3118, 4155]	A4	A3 -> A4	G4	4156

Gambar 7 Halaman Data Hasil Prediksi

3.6. Halaman Grafik Hasil Prediksi

Pada Gambar 8 tampilan halaman grafik hasil prediksi menampilkan daftar data hasil peramalan berbentuk grafik naik turun dan perbandingan dengan data aktualnya.



Gambar 8 Halaman Grafik Hasil Prediks\

4. Source Code Website Fuzzy Time Series-Chen

4.1. Dashboard

```
5  function AppLayout({ children }) {  
6      return (  
7          <div className={'flex h-full'}>  
8              <Sidebar />  
9                
10             <Container>{children}</Container>  
11         </div>  
12     )  
13 }
```

```

7   function Sidebar() {
8     const router = useRouter()
9
10    return (
11      <aside
12        className={
13          'max-w-[64px] md:max-w-[300px] bg-indigo-800 flex-1 h-full transition-[max-width]'
14        }
15      >
16        <div className={'flex flex-1 justify-center items-center h-[64px]}>
17          <span className={'text-[10px] md:text-3xl text-white font-semibold'}>
18            Dashboard
19          </span>
20        </div>
21
22        <ul className={'flex flex-col text-white mx-[8px]}>
23          <li className={'flex px-3 py-2 text-xl'}>
24            <span className={'hidden md:block whitespace-nowrap'}>Beranda</span>
25          </li>
26
27          {_nav.map((navigation) => {
28            const { icon: Icon, label, url } = navigation
29
30            const isActive = getFocusedNavigation(router, navigation)
31
32            const activeClassname = isActive ? 'bg-indigo-900' : ''
33
34            return (
35              <li key={label}>
36                <Link href={url}>
37                  <a
38                    className={'flex items-center gap-x-3 p-3 rounded-md font-medium `${activeClassname}'}
39                  >
40                    <Icon className={'inline-block align-middle'} size={24} />
41                    <span className={'hidden md:block whitespace-nowrap'}>
42                      {label}
43                    </span>
44                  </a>
45                </Link>
46              </li>
47            )
48          )}
49        </ul>
50      </aside>
51    )
52  }

```

Kode 1 *Dashboard*

4.2. Data Aktual

```
10  const View = () => {
11    const { isLoading, data, fetchData, setIsLoading } = useDataAktual()
12
13    return (
14      <AppLayout>
15        <div className={'self-end'}>
16          <Link href={'/data-aktual/tambah'}>
17            <a>
18              <Button className={'bg-indigo-500 text-white whitespace-nowrap'}>
19                Tambah Data Aktual
20              </Button>
21            </a>
22          </Link>
23        </div>
24
25        <ReactTable
26          className={'w-full'}
27          columns={[
28            {Header: 'No'...},
38            {Header: 'Nama Kota'...},
44            {Header: 'Jumlah'...},
50            {Header: 'Tahun'...},
65            {Header: 'Aksi'...}
112        ]}
113        data={data}
114        loading={isLoading}
115        minRows={10}
116        pageSize={10}
117      />
118    </AppLayout>
119  )
120 }
```

```
12    try {
13      const aktuals = await Aktual.find()
14
15      res.status(200).json({ success: true, data: aktuals })
16    } catch (error) {
17      res.status(400).json({ success: false, message: error.message })
18    }
19  }
```

Kode 2 Data Aktual

4.3. Tambah Data Aktual

```
9  const Home = () => {
10     const [namaKota, setNamaKota] = React.useState('')
11     const [tanggal, setTanggal] = React.useState(null)
12     const [jumlah, setJumlah] = React.useState('')
13     const [isSubmitting, setIsSubmitting] = React.useState(false)
14
15     return (
16       <AppLayout>
17         <div>
18           <div className={'text-2xl whitespace-nowrap'}>Tambah Data Aktual</div>
19         </div>
20
21         <form
22           onSubmit={(e : FormEvent<HTMLFormElement> ) => {
23             e.preventDefault()
24             setIsSubmitting(true)
25             fetch('/api/data-aktual', {
26               method: 'POST',
27               headers: {
28                 'Content-Type': 'application/json'
29               },
30               body: JSON.stringify({
31                 namaKota,
32                 tanggal,
33                 jumlah
34               })
35             })
36             .then(async (res : Response ) => {
37               res.jsonData = await res.json()
38               if (!res.ok) {
39                 throw res
40               }
41               setJumlah('')
42               toast.success('Berhasil menambahkan data')
43             })
44           })
45     )
46   }
47 }
```

```
try {
  const aktual = await Aktual.create({
    namaKota: req.body.namaKota,
    tanggal: req.body.tanggal ? dayjs(req.body.tanggal).toDate() : null,
    jumlah: req.body.jumlah
  })

  res.status(201).json({ success: true, data: aktual })
} catch (error) {
  res.status(400).json({ success: false, message: error.message })
}
```

Kode 3 Tambah Data Aktual

4.4. Edit Data Aktual

```
63 <form
64   onSubmit={(e : FormEvent<HTMLFormElement> ) => {
65     e.preventDefault()
66     setIsSubmitting(true)
67     fetch(`/api/data-aktual/${id}`, {
68       method: 'PUT',
69       headers: {
70         'Content-Type': 'application/json'
71       },
72       body: JSON.stringify({
73         namaKota,
74         tanggal,
75         jumlah
76       })
77     })
78     .then(async (res : Response ) => {
79       res.jsonData = await res.json()
80       if (!res.ok) {
81         throw res
82       }
83       toast.success('Berhasil merubah data')
84       router.push('/data-aktual')
85     })
86     .catch((e) => {
87       toast.error(e.jsonData?.message || e?.message)
88     })
89     .finally(() => {
90       setIsSubmitting(false)
91     })
92   }}
93 >
```

```
21  try {
22      const aktual = await Aktual.findByIdAndUpdate(req.query.id, {
23          namaKota: req.body.namaKota,
24          tanggal: req.body.tanggal ? dayjs(req.body.tanggal).toDate() : null,
25          jumlah: req.body.jumlah
26      })
27
28      res.status(200).json({ success: true, data: aktual })
29  } catch (error) {
30      res.status(400).json({ success: false, message: error.message })
31  }
```

Kode 4 *Edit* Data Aktual

4.5. Hapus Data Aktual

```
81 <Button
82   className={'bg-red-500 text-white'}
83   onClick={() => {
84     if (window.confirm('Apakah Anda yakin?')) {
85       setIsLoading(true)
86       fetch(`/api/data-aktual/${_id}`, {
87         method: 'DELETE'
88       })
89       .then(async (res : Response) => {
90         res.jsonData = await res.json()
91         if (!res.ok) {
92           throw res
93         }
94
95         toast.success('Berhasil menghapus data')
96       })
97       .catch((e) => {
98         toast.error(e.jsonData?.message || e.message)
99       })
100      .finally(() => {
101        fetchData()
102      })
103    }
104  }}
105 >
106   Hapus
107 </Button>
```

```
35   try {
36     await Aktual.findByIdAndDelete(req.query.id)
37
38     res.status(201).json({ success: true })
39   } catch (error) {
40     res.status(400).json({ success: false, message: error.message })
41   }
```

Kode 5 Hapus Data Aktual

4.6. Kategori *Universe of Discourse & Re-Divide Interval*

```
74 <div>
75   <p className={'text-3xl font-bold mb-3'}>Universe of Discourse</p>
76   <div>
77     <b>DMIN:</b> {DMin}
78   </div>
79   <div>
80     <b>DMAX:</b> {DMax}
81   </div>
82   <div className={'border'}>
83     <div>
84       <b>Banyak Kelas:</b> {banyakKelas}
85     </div>
86     <div className={'text-gray-500'}>
87       {`Note: Menggunakan Rumus Sturges = 1+3.3*Log(n) = 1+3.3*Log(${totalData})`}
88     </div>
89   </div>
90   <div>
91     <b>Rentang Kelas:</b> {rentangKelas}
92   </div>
93   <div>
94     <b>Interval Kelas:</b> {intervalKelas}
95   </div>
96 </div>
97
98 <div>
99   <p className={'text-3xl font-bold mb-3'}>Re-divide Interval</p>
100  <table className={'border'}>
101    <thead>
102      <tr>
103        <th className={'p-2 border'}>Redivide Ui</th>
104        <th className={'p-2 border'}>Nilai Tengah</th>
105      </tr>
106    </thead>
107    <tbody>
108      {listInterval.map((dataInterval, index) => {
109        const { batasAwal, batasAkhir, nilaiTengah, kelas } = dataInterval
110        return (
111          <tr key={index} className={'p-2'}>
112            <td
113              className={'p-2 border'}
114              >{'U${kelas} = [${batasAwal}, ${batasAkhir}]'}</td>
115            <td className={'p-2 border text-center'}>{nilaiTengah}</td>
116            <td className={'p-2 border text-center'}>A{kelas}</td>
117          </tr>
118        )
119      })}
120    </tbody>
121  </table>
```

```

function calculateReDivideUi(DMin, DMax, intervalKelas, pembulatanBanyakKelas) {
  const listKelas = []

  for (let i = 0; i < pembulatanBanyakKelas; i++) {
    const number = i + 1
    if (i > 0) {
      const previousInterval = listKelas[i - 1]

      const batasAwal = previousInterval.batasAkhir
      const batasAkhir = batasAwal + intervalKelas

      listKelas.push({
        kelas: number,
        batasAwal,
        batasAkhir,
        nilaiTengah: Math.round((batasAwal + batasAkhir) / 2)
      })
    } else {
      const batasAwal = DMin
      const batasAkhir = DMin + intervalKelas
      listKelas.push({
        kelas: number,
        batasAwal,
        batasAkhir,
        nilaiTengah: Math.round((batasAwal + batasAkhir) / 2)
      })
    }
  }

  return listKelas
}

```

```

47     const listJumlah = responseData.map((aktual) => aktual.jumlah)
48     const DMax = Math.max(...listJumlah)
49     const DMin = Math.min(...listJumlah)
50     const totalData = listJumlah.length
51     const banyakKelas = 1 + 3.3 * Math.log10(totalData)
52     const rentangKelas = DMax - DMin
53     const pembulatanBanyakKelas = Math.round(banyakKelas)
54     const intervalKelas = rentangKelas / pembulatanBanyakKelas
55
56     const listInterval = calculateReDivideUi(
57       DMin,
58       DMax,
59       intervalKelas,
60       pembulatanBanyakKelas
61     )
62

```

Kode 6 Kategori *Universe of Discourse & Re-Divide Interval*

4.7. Kategori *Fuzzy Logical Relations*

```
128 function FuzzyLogicalRelation({ result }) {
129   const { fuzzyLogicRelations } = result
130
131   return (
132     <div>
133       <table className={'border'}>
134         <thead>
135           <tr>
136             <th className={'p-2 border'}>No</th>
137             <th className={'p-2 border'}>Aktual</th>
138             <th className={'p-2 border'}>Interval</th>
139             <th className={'p-2 border'}>Fuzzifikasi</th>
140             <th className={'p-2 border'}>Relasi</th>
141           </tr>
142         </thead>
143         <tbody>
144           {fuzzyLogicRelations.map((relation, index) => {
145             const { kelas, prevKelas, jumlah, dataInterval } = relation
146             const { batasAwal, batasAkhir } = dataInterval
147             return (
148               <tr key={index} className={'p-2'}>
149                 <td className={'p-2 border text-center'}>{index + 1}</td>
150                 <td className={'p-2 border'}>{jumlah}</td>
151                 <td className={'p-2 border'}>
152                   [{batasAwal}, {batasAkhir}]
153                 </td>
154                 <td className={'p-2 border'}>A{kelas}</td>
155                 <td className={'p-2 border'}>`${
156                   prevKelas ? `A${[prevKelas]}` : '*'
157                 } -> A${kelas}`</td>
158               </tr>
159             )
160           })}
161         </tbody>
162       </table>
163     </div>
164   )
165 }
```

```
63     const fuzzyLogicRelations = []
64     responseData.forEach((aktual, aktualIndex) => {
65         const { jumlah } = aktual
66         const newAktual = {
67             jumlah,
68             ...aktual
69         }
70
71         for (let i = 0; i < listInterval.length; i++) {
72             const dataInterval = listInterval[i]
73             if (
74                 jumlah >= dataInterval.batasAwal &&
75                 jumlah <= dataInterval.batasAkhir
76             ) {
77                 newAktual.kelas = dataInterval.kelas
78                 newAktual.prevKelas =
79                     fuzzyLogicRelations?.[aktualIndex - 1]?.kelas
80                 newAktual.dataInterval = dataInterval
81                 break
82             }
83         }
84
85         fuzzyLogicRelations.push(newAktual)
86     })
```

Kode 7 Kategori *Fuzzy Logical Relations*

4.8. Kategori *Fuzzy Logical Relation Group*

```
167 function FuzzyLogicalRelationGroup({ result }) {
168   const { fuzzyLogicRelationGroups } = result
169
170   return (
171     <div>
172       <table className='border'>
173         <thead>
174           <tr>
175             <th className='p-2 border' colSpan={2}>
176               Group Relasi
177             </th>
178             <th className='p-2 border'>Ramalan</th>
179           </tr>
180         </thead>
181         <tbody>
182           {Object.entries(fuzzyLogicRelationGroups).map((group : [string, unknown], index : number) => {
183             const [kelas, data] = group
184
185             return (
186               <tr key={index} className='p-2'>
187                 <td className='p-2 border text-center'>
188                   G{kelas} {'-'>'}
189                 </td>
190                 <td className='p-2 border text-center'>
191                   {data.groups.map((group) => `A${group}`).join(', ')}
192                 </td>
193                 <td className='p-2 border text-center'>{data.ramalan}</td>
194               </tr>
195             )
196           )}
197         </tbody>
198       </table>
199     </div>
200   )
201 }
```

```
8      const fuzzyLogicRelationGroups = {}
9      for (let i = 0; i < fuzzyLogicRelations.length; i++) {
10         const relation = fuzzyLogicRelations[i]
11         const { kelas, prevKelas, dataInterval } = relation
12
13         if (!prevKelas) {
14             continue
15         }
16
17         if (!fuzzyLogicRelationGroups[prevKelas]) {
18             fuzzyLogicRelationGroups[prevKelas] = {
19                 groups: [kelas],
20                 ramalan: dataInterval.nilaiTengah,
21                 totalRamalan: dataInterval.nilaiTengah
22             }
23             continue
24         }
25
26         if (!fuzzyLogicRelationGroups[prevKelas].groups.includes(kelas)) {
27             fuzzyLogicRelationGroups[prevKelas].groups.push(kelas)
28             fuzzyLogicRelationGroups[prevKelas].totalRamalan +=
29                 dataInterval.nilaiTengah
30             fuzzyLogicRelationGroups[prevKelas].ramalan = Math.round(
31                 fuzzyLogicRelationGroups[prevKelas].totalRamalan /
32                 fuzzyLogicRelationGroups[prevKelas].groups.length
33             )
34         }
35     }
36 }
```

Kode 8 Kategori *Fuzzy Logical Relation Group*

4.9. Data Hasil Prediksi

```
16     const { dataPrediksi } = result
17
18     return (
19       <AppLayout>
20         <div className={'overflow-auto min-w-0'}>
21           <table className={'border'}>
22             <thead>
23               <tr>
24                 <th className={'p-2 border'}>No</th>
25                 <th className={'p-2 border'}>Kota</th>
26                 <th className={'p-2 border'}>Aktual</th>
27                 <th className={'p-2 border'}>Interval</th>
28                 <th className={'p-2 border'}>Fuzzifikasi</th>
29                 <th className={'p-2 border'}>Relasi</th>
30                 <th className={'p-2 border'}>FLRG</th>
31                 <th className={'p-2 border'}>Ramalan Fuzzy Chen</th>
32               </tr>
33             </thead>
34             <tbody>
35               {dataPrediksi.map((relation, index) => {
36                 const {
37                   kelas,
38                   prevKelas,
39                   jumlah,
40                   dataInterval,
41                   namaKota,
42                   ramalanFuzzyChen
43                 } = relation
44                 const { batasAwal, batasAkhir } = dataInterval
45                 return (
46                   <tr key={index} className={'p-2'}>
47                     <td className={'p-2 border text-center'}>{index + 1}</td>
48                     <td className={'p-2 border'}>{namaKota}</td>
49                     <td className={'p-2 border'}>{jumlah}</td>
50                     <td className={'p-2 border'}>
51                       [{batasAwal}, {batasAkhir}]
52                     </td>
53                     <td className={'p-2 border'}>A{kelas}</td>
54                     <td className={'p-2 border'}>{'${
55                       prevKelas ? `A${[prevKelas]}` : '*'
56                     } -> A${kelas}'</td>
57                     <td className={'p-2 border'}>G{kelas}</td>
58                     <td className={'p-2 border'}>{ramalanFuzzyChen}</td>
59                   </tr>
60                 )
61               })}
62             </tbody>
63           </table>
```



```
const dataPrediksi = fuzzyLogicRelations.map((relation) => {
  const { kelas } = relation

  return {
    ...relation,
    ramalanFuzzyChen: fuzzyLogicRelationGroups[kelas]?.ramalan
  }
})
```

Kode 9 Data Hasil Prediksi

4.10. Grafik Hasil Prediksi

```
14     const { dataPrediksi } = result
15
16     useEffect(() => {
17       if (!dataPrediksi || dataPrediksi.length <= 0) {...}
18
19
20
21       const ctx = document.getElementById('myChart')
22
23       if (refChart.current) {...}
24
25
26
27       const myChart = new Chart(ctx, {
28         type: 'line',
29         data: {
30           labels: dataPrediksi.map(
31             (aktual) =>
32               `${aktual.namaKota} (${dayjs(aktual.tanggal).format('YYYY')})`
33           ),
34           datasets: [
35             {
36               label: 'Nilai Aktual',
37               data: dataPrediksi.map((aktual) => aktual.jumlah),
38               backgroundColor: ['rgb(0,75,255)'],
39               borderColor: ['rgb(0,75,255)'],
40               borderWidth: 2
41             },
42             {
43               label: 'Nilai Prediksi',
44               data: dataPrediksi.map((aktual) => aktual.ramalanFuzzyChen),
45               backgroundColor: ['rgb(167,198,255)'],
46               borderColor: ['rgb(167,198,255)'],
47               borderWidth: 2
48             }
49           ]
50         },
51         options: {
52           scales: {
53             x: {
54               ticks: {
55                 maxRotation: 75,
56                 minRotation: 0
57               }
58             },
59             y: {
60               min: 0,
61               max: 8000,
62               ticks: {
63                 autoSkip: false,
64                 stepSize: 500
65               }
66             }
67           }
68         }
69       }
70     )
71   }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
```

```
const dataPrediksi = fuzzyLogicRelations.map((relation) => {
  const { kelas } = relation

  return {
    ...relation,
    ramalanFuzzyChen: fuzzyLogicRelationGroups[kelas]?.ramalan
  }
})
```

Kode 10 Grafik Hasil Prediksi