LAMPIRAN

-Link Github:

-Detail Coding Implementasi SSR pada halaman Auth

```
import axios from 'axios';
import { useCallback, useState } from 'react';
import { NextPageContext } from 'next';
import { getSession, signIn } from 'next-auth/react';
import { useRouter } from 'next/router';
import { FcGoogle } from 'react-icons/fc';
import { FaGithub } from 'react-icons/fa';

import Input from '@/components/Input';

export async function getServerSideProps(context: NextPageContext) {
  const session = await getSession(context);

  if (session) {
    return {
      redirect: {
        destination: '/',
        permanent: false,
      }
    }
  }

  return {
    props: {}
  }
}

const Auth = () => {
  const router = useRouter();

  const [email, setEmail] = useState('');
  const [name, setName] = useState('');
  const [password, setPassword] = useState('');

  const [variant, setVariant] = useState('login');

  const toggleVariant = useCallback(() => {
    setVariant((currentVariant) => currentVariant === 'login' ? 'register' :
'login');
  }, []);

  const login = useCallback(async () => {
    try {
      await signIn('credentials', {
        email,
        password,
        redirect: false,
        callbackUrl: '/'
      });

      router.push('/profiles');
    } catch (error) {
      console.log(error);
    }
  }, [email, password, router]);

  const register = useCallback(async () => {
    try {
      await axios.post('/api/register', {
        email,
        name,
        password
      });

      login();
    } catch (error) {
        console.log(error);
    }
  }, [email, name, password, login]);

  return (
    <div className="relative h-full w-full bg-[url('/images/hero.jpg')] bg-
no-repeat bg-center bg-fixed bg-cover">
      <div className="bg-black w-full h-full lg:bg-opacity-50">
        <nav className="px-12 py-5">
          <img src="/images/logo.png" className="h-12" alt="Logo" />
        </nav>
        <div className="flex justify-center">
          <div className="bg-black bg-opacity-70 px-16 py-16 self-center mt-2
lg:w-2/5 lg:max-w-md rounded-md w-full">
            <h2 className="text-white text-4xl mb-8 font-semibold">
              {variant === 'login' ? 'Sign in' : 'Register'}
            </h2>
            <div className="flex flex-col gap-4">
              {variant === 'register' && (
                <Input
                  id="name"
                  type="text"
                  label="Username"
                  value={name}
                  onChange={(e: any) => setName(e.target.value)}
                />
              )}
              <Input
                id="email"
                type="email"
                label="Email address or phone number"
                value={email}
                onChange={(e: any) => setEmail(e.target.value)}
              />
              <Input
                type="password"
                id="password"
                label="Password"
                value={password}
                onChange={(e: any) => setPassword(e.target.value)}
              />
            </div>
            <button onClick={variant === 'login' ? login : register}
className="bg-red-600 py-3 text-white rounded-md w-full mt-10 hover:bg-red-
700 transition">
              {variant === 'login' ? 'Login' : 'Sign up'}
            </button>
            <div className="flex flex-row items-center gap-4 mt-8 justify-
center">
              <div onClick={() => signIn('google', { callbackUrl: '/profiles'
})} className="w-10 h-10 bg-white rounded-full flex items-center justify-
center cursor-pointer hover:opacity-80 transition">
                <FcGoogle size={32} />
              </div>
              <div onClick={() => signIn('github', { callbackUrl: '/profiles'
})} className="w-10 h-10 bg-white rounded-full flex items-center justify-
center cursor-pointer hover:opacity-80 transition">
                <FaGithub size={32} />
              </div>
            </div>
            <p className="text-neutral-500 mt-12">
              {variant === 'login' ? 'First time using Netflix?' : 'Already
have an account?'}
              <span onClick={toggleVariant} className="text-white ml-1
hover:underline cursor-pointer">
                {variant === 'login' ? 'Create an account' : 'Login'}
              </span>
              .
            </p>
          </div>
        </div>
      </div>
    </div>
  );
}

export default Auth;
```

-Detail Coding Implementasi SSR pada halaman Profil

```
import { NextPageContext } from "next";
import { getSession, useSession } from "next-auth/react";
import { useRouter } from "next/router";
import { useCallback } from "react";

import useCurrentUser from "@/hooks/useCurrentUser";

const images = [
  '/images/default-blue.png',
  '/images/default-red.png',
  '/images/default-slate.png',
  '/images/default-green.png'
]

interface UserCardProps {
  name: string;
}

export async function getServerSideProps(context: NextPageContext) {
  const session = await getSession(context);

  if (!session) {
    return {
      redirect: {
        destination: '/auth',
        permanent: false,
      }
    }
  }

  return {
    props: {}
  }
}

const UserCard: React.FC<UserCardProps> = ({ name }) => {
  const imgSrc = images[Math.floor(Math.random() * 4)];
  return (
    <div className="group flex-row w-44 mx-auto">
        <div className="w-44 h-44 rounded-md flex items-center justify-center
border-2 border-transparent group-hover:cursor-pointer group-hover:border-
white overflow-hidden">
          <img draggable={false} className="w-max h-max object-contain"
src={imgSrc} alt="" />
        </div>
      <div className="mt-4 text-gray-400 text-2xl text-center group-
hover:text-white">{name}</div>
   </div>
  );
}

const App = () => {
  const router = useRouter();
  const { data: currentUser } = useCurrentUser();

  const selectProfile = useCallback(() => {
    router.push('/');
  }, [router]);

  return (
    <div className="flex items-center h-full justify-center">
      <div className="flex flex-col">
        <h1 className="text-3xl md:text-6xl text-white text-center">Who&#39;s
watching?</h1>
        <div className="flex items-center justify-center gap-8 mt-10">
          <div onClick={() => selectProfile()}>
            <UserCard name={currentUser?.name} />
          </div>
        </div>
      </div>
    </div>
  );
}

export default App;
```

-Detail Coding Implementasi SSR pada halaman Utama

```
import React from 'react';
import { NextPageContext } from 'next';
import { getSession } from 'next-auth/react';

import Navbar from '@/components/Navbar';
import Billboard from '@/components/Billboard';
import MovieList from '@/components/MovieList';
import InfoModal from '@/components/InfoModal';
import useMovieList from '@/hooks/useMovieList';
import useFavorites from '@/hooks/useFavorites';
import useInfoModalStore from '@/hooks/useInfoModalStore';

export async function getServerSideProps(context: NextPageContext) {
  const session = await getSession(context);

  if (!session) {
    return {
      redirect: {
        destination: '/auth',
        permanent: false,
      }
    }
  }

  return {
    props: {}
  }
}

const Home = () => {
  const { data: movies = [] } = useMovieList();
  const { data: favorites = [] } = useFavorites();
  const {isOpen, closeModal} = useInfoModalStore();

  return (
    <>
      <InfoModal visible={isOpen} onClose={closeModal} />
      <Navbar />
      <Billboard />
      <div className="pb-40">
        <MovieList title="Trending Now" data={movies} />
        <MovieList title="My List" data={favorites} />
      </div>
    </>
  )
}

export default Home;
```

-Detail Coding Implementasi CSR pada halaman Utama

```
import React, { useEffect, useState } from 'react';
import Navbar from '@/components/Navbar';
import Billboard from '@/components/Billboard';
import MovieList from '@/components/MovieList';
import InfoModal from '@/components/InfoModal';
import useInfoModalStore from '@/hooks/useInfoModalStore';

const Home = () => {
  const [movies, setMovies] = useState([]);
  const [favorites, setFavorites] = useState([]);
  const { isOpen, closeModal } = useInfoModalStore();

  useEffect(() => {
    fetch('/api/movies')
      .then((response) => response.json())
      .then((data) => setMovies(data))
      .catch((error) => console.error('Error fetching movies:', error));

    fetch('/api/favorites')
      .then((response) => response.json())
      .then((data) => setFavorites(data))
      .catch((error) => console.error('Error fetching favorites:', error));
  }, []);

  return (
    <>
      <InfoModal visible={isOpen} onClose={closeModal} />
      <Navbar />
      <Billboard />
      <div className="pb-40">
        <MovieList title="Trending Now" data={movies} />
        <MovieList title="My List" data={favorites} />
      </div>
    </>
  );
};

export default Home;
```

-Detail Coding Implementasi SSG pada halaman Auth

```
import axios from 'axios';
import { useCallback, useState } from 'react';
import { NextPageContext } from 'next';
import { getSession, signIn } from 'next-auth/react';
import { useRouter } from 'next/router';
import { FcGoogle } from 'react-icons/fc';
import { FaGithub } from 'react-icons/fa';

import Input from '@/components/Input';

export async function getStaticProps(context: NextPageContext) {
  const session = await getSession(context);

  if (session) {
    return {
      redirect: {
        destination: '/',
        permanent: false,
      }
    }
  }

  return {
    props: {}
  }
}

const Auth = () => {
  const router = useRouter();

  const [email, setEmail] = useState('');
  const [name, setName] = useState('');
  const [password, setPassword] = useState('');

  const [variant, setVariant] = useState('login');

  const toggleVariant = useCallback(() => {
    setVariant((currentVariant) => currentVariant === 'login' ? 'register' :
'login');
  }, []);

  const login = useCallback(async () => {
    try {
      await signIn('credentials', {
        email,
        password,
        redirect: false,
        callbackUrl: '/'
      });

      router.push('/profiles');
    } catch (error) {
      console.log(error);
    }
  }, [email, password, router]);
  const register = useCallback(async () => {
    try {
      await axios.post('/api/register', {
        email,
        name,
        password
      });

      login();
    } catch (error) {
        console.log(error);
    }
  }, [email, name, password, login]);

  return (
    <div className="relative h-full w-full bg-[url('/images/hero.jpg')] bg-
no-repeat bg-center bg-fixed bg-cover">
      <div className="bg-black w-full h-full lg:bg-opacity-50">
        <nav className="px-12 py-5">
          <img src="/images/logo.png" className="h-12" alt="Logo" />
        </nav>
        <div className="flex justify-center">
          <div className="bg-black bg-opacity-70 px-16 py-16 self-center mt-2
lg:w-2/5 lg:max-w-md rounded-md w-full">
            <h2 className="text-white text-4xl mb-8 font-semibold">
              {variant === 'login' ? 'Sign in' : 'Register'}
            </h2>
            <div className="flex flex-col gap-4">
              {variant === 'register' && (
                <Input
                  id="name"
                  type="text"
                  label="Username"
                  value={name}
                  onChange={(e: any) => setName(e.target.value)}
                />
              )}
              <Input
                id="email"
                type="email"
                label="Email address or phone number"
                value={email}
                onChange={(e: any) => setEmail(e.target.value)}
              />
              <Input
                type="password"
                id="password"
                label="Password"
                value={password}
                onChange={(e: any) => setPassword(e.target.value)}
              />
            </div>
            <button onClick={variant === 'login' ? login : register}
className="bg-red-600 py-3 text-white rounded-md w-full mt-10 hover:bg-red-
700 transition">
              {variant === 'login' ? 'Login' : 'Sign up'}
            </button>
            <div className="flex flex-row items-center gap-4 mt-8 justify-
center">
              <div onClick={() => signIn('google', { callbackUrl: '/profiles'
})} className="w-10 h-10 bg-white rounded-full flex items-center justify-
center cursor-pointer hover:opacity-80 transition">
```

```
                <FcGoogle size={32} />
              </div>
              <div onClick={() => signIn('github', { callbackUrl: '/profiles'
})} className="w-10 h-10 bg-white rounded-full flex items-center justify-
center cursor-pointer hover:opacity-80 transition">
                <FaGithub size={32} />
              </div>
            </div>
            <p className="text-neutral-500 mt-12">
              {variant === 'login' ? 'First time using Netflix?' : 'Already
have an account?'}
              <span onClick={toggleVariant} className="text-white ml-1
hover:underline cursor-pointer">
                {variant === 'login' ? 'Create an account' : 'Login'}
              </span>
              .
            </p>
          </div>
        </div>
      </div>
    </div>
  );
}

export default Auth;
```

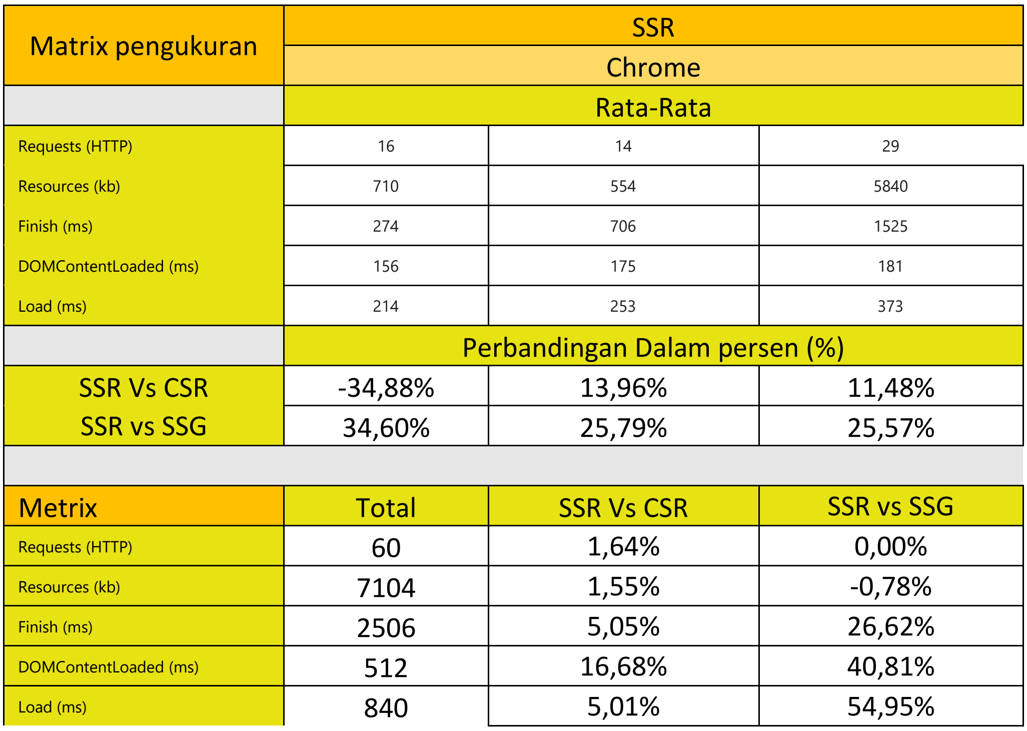
-Detail Coding Implementasi SSG pada halaman Profil

```
import { NextPageContext } from "next";
import { getSession, useSession } from "next-auth/react";
import { useRouter } from "next/router";
import { useCallback } from "react";

import useCurrentUser from "@/hooks/useCurrentUser";

const images = [
  '/images/default-blue.png',
  '/images/default-red.png',
  '/images/default-slate.png',
  '/images/default-green.png'
]

interface UserCardProps {
  name: string;
}

export async function getServerSideProps(context: NextPageContext) {
  const session = await getSession(context);

  if (session) {
    return {
      redirect: {
        destination: '/',
        permanent: false,
      }
    }
  }

  return {
    props: {}
  }
}
```

```
const UserCard: React.FC<UserCardProps> = ({ name }) => {
  const imgSrc = images[Math.floor(Math.random() * 4)];

  return (
    <div className="group flex-row w-44 mx-auto">
        <div className="w-44 h-44 rounded-md flex items-center justify-center
border-2 border-transparent group-hover:cursor-pointer group-hover:border-
white overflow-hidden">
          <img draggable={false} className="w-max h-max object-contain"
src={imgSrc} alt="" />
        </div>
      <div className="mt-4 text-gray-400 text-2xl text-center group-
hover:text-white">{name}</div>
   </div>
  );
}

const App = () => {
  const router = useRouter();
  const { data: currentUser } = useCurrentUser();

  const selectProfile = useCallback(() => {
    router.push('/');
  }, [router]);

  return (
    <div className="flex items-center h-full justify-center">
      <div className="flex flex-col">
        <h1 className="text-3xl md:text-6xl text-white text-center">Who&#39;s
watching?</h1>
        <div className="flex items-center justify-center gap-8 mt-10">
          <div onClick={() => selectProfile()}>
            <UserCard name={currentUser?.name} />
          </div>
        </div>
      </div>
    </div>
  );
}

export default App;
```

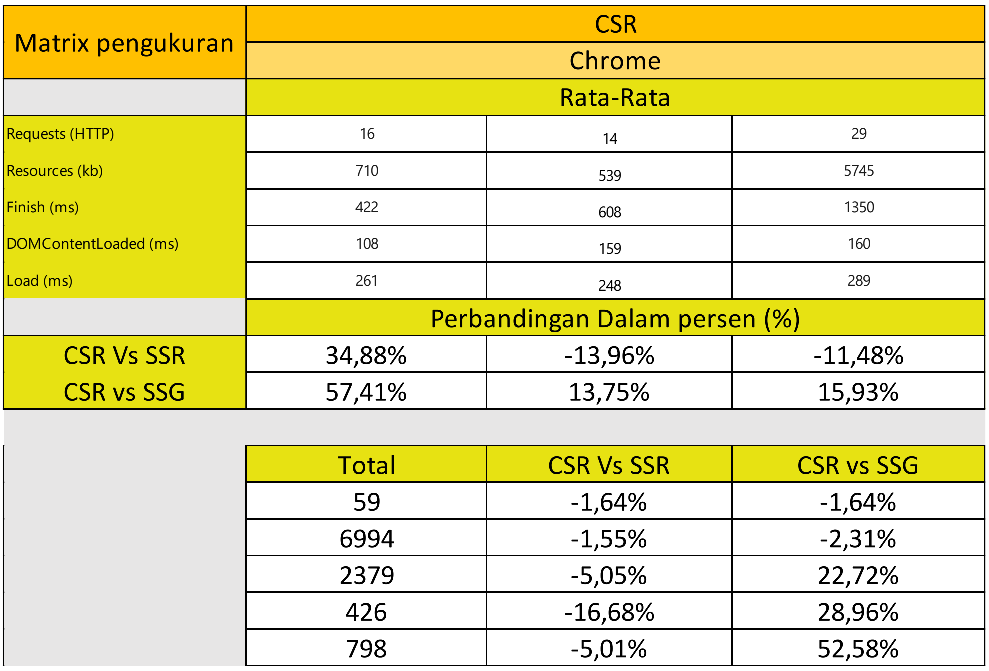
-Detail Coding Implemententasi SSG pada halaman Utama

```
import React from 'react';
import { NextPageContext } from 'next';
import { getSession } from 'next-auth/react';

import Navbar from '@/components/Navbar';
import Billboard from '@/components/Billboard';
import MovieList from '@/components/MovieList';
import InfoModal from '@/components/InfoModal';
import useMovieList from '@/hooks/useMovieList';
import useFavorites from '@/hooks/useFavorites';
import useInfoModalStore from '@/hooks/useInfoModalStore';

export async function getServerSideProps(context: NextPageContext) {
  const session = await getSession(context);

  if (!session) {
    return {
      redirect: {
        destination: '/auth',
        permanent: false,
      }
    }
  }

  return {
    props: {}
  }
}

const Home = () => {
  const { data: movies = [] } = useMovieList();
  const { data: favorites = [] } = useFavorites();
  const {isOpen, closeModal} = useInfoModalStore();

  return (
    <>
      <InfoModal visible={isOpen} onClose={closeModal} />
      <Navbar />
      <Billboard />
      <div className="pb-40">
        <MovieList title="Trending Now" data={movies} />
        <MovieList title="My List" data={favorites} />
      </div>
    </>
  )
}

export default Home;
```



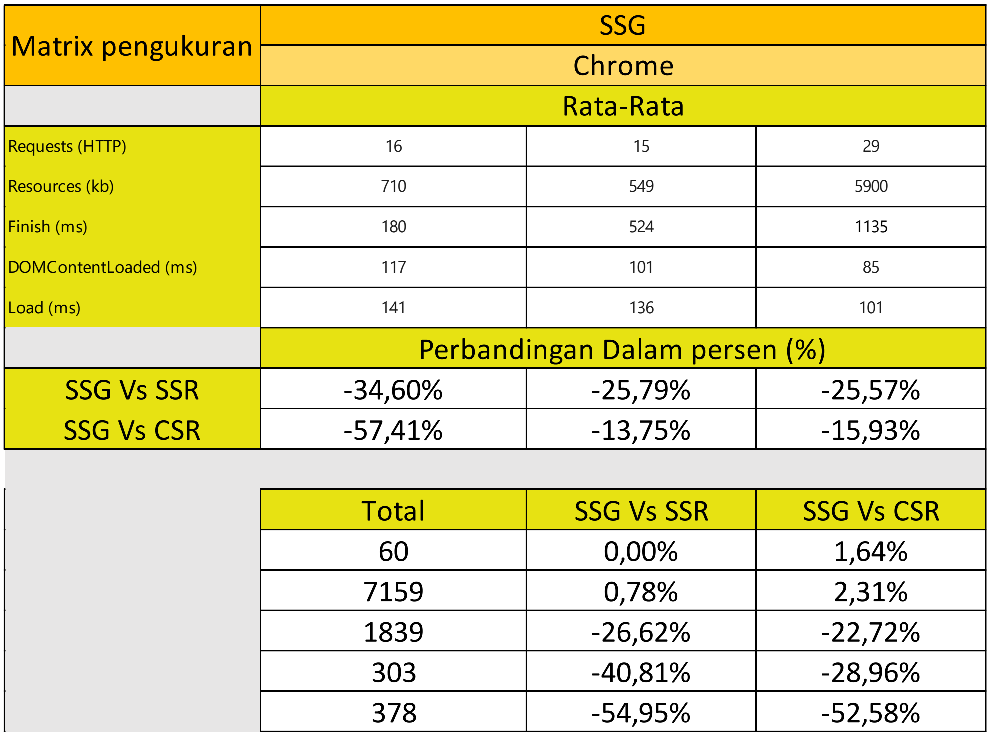
Matrix pengukuran	SSR		
	Chrome		
	Rata-Rata		
Requests (HTTP)	16	14	29
Resources (kb)	710	554	5840
Finish (ms)	274	706	1525
DOMContentLoaded (ms)	156	175	181
Load (ms)	214	253	373
	Perbandingan Dalam persen (%)		
SSR Vs CSR	-34,88%	13,96%	11,48%
SSR vs SSG	34,60%	25,79%	25,57%

Metrix	Total	SSR Vs CSR	SSR vs SSG
Requests (HTTP)	60	1,64%	0,00%
Resources (kb)	7104	1,55%	-0,78%
Finish (ms)	2506	5,05%	26,62%
DOMContentLoaded (ms)	512	16,68%	40,81%
Load (ms)	840	5,01%	54,95%

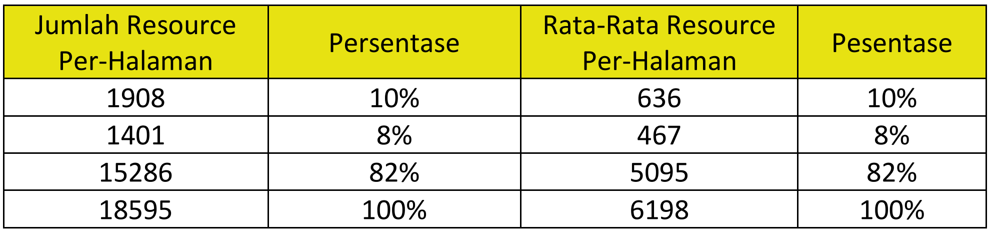


Matrix pengukuran	CSR		
	Chrome		
	Rata-Rata		
Requests (HTTP)	16	14	29
Resources (kb)	710	539	5745
Finish (ms)	422	608	1350
DOMContentLoaded (ms)	108	159	160
Load (ms)	261	248	289
	Perbandingan Dalam persen (%)		
CSR Vs SSR	34,88%	-13,96%	-11,48%
CSR vs SSG	57,41%	13,75%	15,93%

	Total	CSR Vs SSR	CSR vs SSG
	59	-1,64%	-1,64%
	6994	-1,55%	-2,31%
	2379	-5,05%	22,72%
	426	-16,68%	28,96%
	798	-5,01%	52,58%



Matrix pengukuran	SSG		
	Chrome		
	Rata-Rata		
Requests (HTTP)	16	15	29
Resources (kb)	710	549	5900
Finish (ms)	180	524	1135
DOMContentLoaded (ms)	117	101	85
Load (ms)	141	136	101
	Perbandingan Dalam persen (%)		
SSG Vs SSR	-34,60%	-25,79%	-25,57%
SSG Vs CSR	-57,41%	-13,75%	-15,93%
	Total	SSG Vs SSR	SSG Vs CSR
	60	0,00%	1,64%
	7159	0,78%	2,31%
	1839	-26,62%	-22,72%
	303	-40,81%	-28,96%
	378	-54,95%	-52,58%



Jumlah Resource Per-Halaman	Persentase	Rata-Rata Resource Per-Halaman	Pesentase
1908	10%	636	10%
1401	8%	467	8%
15286	82%	5095	82%
18595	100%	6198	100%