

**MODUL PRAKTIKUM
JARINGAN KOMPUTER UNTUK MULTIMEDIA**



Oleh :

Mohammad Robihul Mufid, S.ST., M.Tr.Kom.

**PROGRAM STUDI D3 TEKNOLOGI MULTIMEDIA
BROADCASTING PSDKU PENS LAMONGAN
DEPARTEMEN TEKNOLOGI MULTIMEDIA KREATIF
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**

TAHUN 2021

PERTEMUAN 1

Multimedia & Application

A. Sub Capaian Pembelajaran Mata Kuliah :

1. Mahasiswa mampu menyebutkan definisi multimedia,
2. Mahasiswa mampu menjelaskan jenis-jenis multimedia
3. Mahasiswa mampu memahami pengaplikasian dari multimedia

Penilaian Mata Kuliah : UTS (30%), UAS (30%), Tugas (30%), Keaktifan (10%)

B. Dasar Teori

Menurut bahasa, multimedia berasal dari dua kata, yaitu multi dan media. Multi artinya banyak, sedangkan media berarti sesuatu yang digunakan untuk mengekspresikan pesan atau perantara. Jadi, Pengertian dari multimedia adalah perantara pesan yang meliputi lebih dari 1 elemen atau banyak. Definisi lain dari Multimedia diantaranya sebagai sarana komunikasi yang menggabungkan teks, grafik, gambar diam dan bergerak, animasi audio, ataupun media lainnya. Beberapa kombinasi elemen tersebut bisa, disimpan, ditampilkan, dikirim, atau diproses dengan sebuah perangkat tertentu.

Pengertian Jenis Multimedia Menurut Para Ahli :

- Menurut Rosch, tahun 1996: Multimedia merupakan Kombinasi dari komputer dan video.
- Menurut McComick tahun 1996: Multimedia merupakan Kombinasi dari tiga elemen, yakni suara, gambar, dan teks.
- Menurut Turban dan kawan-kawan tahun 2002: Multimedia merupakan suatu Kombinasi dari paling sedikitnya dua media input dan output. Media ini bisa berupa audio (suara, atau musik), animasi, video, teks, grafik serta gambar.
- Menurut Robin dan Linda di tahun 2001: Multimedia merupakan Alat yang dapat menciptakan presentasi secara dinamis dan interaktif yang menggabungkan teks, grafik, animasi, audio serta video.
- Menurut Vaughan tahun 2004: Multimedia merupakan kombinasi dari beberapa teks, gambar, suara, animasi dan video yang dikirim ke anda melalui komputer atau alat elektronik lainnya atau dengan manipulasi digital.

Kesimpulannya, pengertian Multimedia adalah suatu kombinasi yang dibentuk dari gabungan teks, gambar, animasi, audio atau video untuk menghasilkan output tertentu berupa informasi yang menarik atau hal lainnya.

Secara **umum** multimedia dapat dibedakan menjadi dua jenis, yaitu multimedia conten production dan multimedia communication.

1. Produksi Konten Multimedia

Multimedia content production merupakan hasil atau produk dari proses penggabungan beberapa media seperti, teks, audio, animasi, graphics, interactivity, serta video untuk menyampaikan sebuah informasi.

Kemudian hasil dari proses itu disebut produk multimedia (music, video, film, game, entertainment, dll). Untuk multimedia jenis ini, media yang digunakan berupa: media audio, media teks, media animasi, media video, grafik media / gambar, media spesial effect, dan interaktivitas media.

2. Komunikasi Multimedia

Multimedia communication adalah proses pemakaian atau penggunaan media (massa), seperti media cetak, radio, televisi, serta internet. Tujuannya adalah sebagai media untuk menyampaikan atau mempublikasikan sebuah pesan atau informasi, seperti, material advertising (iklan), news (berita), publicity (publikasi), entertainment (hiburan), education (pembelajaran atau pendidikan), tutorial dan sebagainya. Untuk multimedia jenis ini, media yang digunakan berupa: media cetak, radio, tv film, permainan, musik, hiburan, ICT (internet).

Adapun **jenis-jenis multimedia jika ditinjau dari multimedia dalam proses pembelajaran**, bisa dibedakan sebagai berikut :

1. Multimedia Interaktif

Pengguna atau user secara penuh dapat mengontrol seputar apa dan kapan elemen multimedia akan dikirimkan atau di tampilkan. Contoh Game, aplikasi program, CD interaktif, dan virtual reality. Ada beberapa jenis atau model multimedia interaktif yang digunakan untuk tujuan pembelajaran, diantaranya tutorial, drill and practice, simulasi, instructional games, hybrid, socratic, inquiry dan informational.

2. Multimedia Hiperaktif

Jenis Multimedia Hiperaktif mempunyai struktur dengan elemen – elemen terkait yang dapat di arahkan oleh pengguna dengan tautan (link) menuju elemen – elemen multimedia yang ada. Misalnya seperti Word wide web, Website, Mobile banking, Game online dan lain-lain.

3. Multimedia Linear (Berurutan)

Multimedia Linear merupakan jenis multimedia yang berjalan lurus, multimedia seperti ini biasanya banyak kita temukan pada semua jenis film, Tutorial video, dan sebagainya. Sedangkan untuk multimedia interaktif merupakan jenis multimedia interaksi, artinya suatu interaksi antara media dan pengguna media dengan melalui bantuan komputer, mouse, keyboard dan sebagainya. Interaksi multimedia seperti ini berlangsung tanpa kontrol navigasi dari pengguna, Penyajian multimedia liner harus berurutan atau sekuensial dari awal sampai akhir. Contoh dari multimedia linier adalah movie/film, e-book, musik, siaran TV.

4. Multimedia Presentasi Pembelajaran

Multimedia presentasi pembelajaran merupakan alat bantu bagi guru untuk proses pembelajaran di kelas, namun tidak menggantikan peran guru secara keseluruhan dalam mengajar. Contohnya adalah Microsoft Powerpoint.

5. Multimedia Pembelajaran Mandiri

Multimedia pembelajaran mandiri merupakan suatu software yang bisa dimanfaatkan secara mandiri oleh siswa tanpa bantuan guru. Multimedia pembelajaran mandiri juga harus bisa dalam menggabungkan explicit, knowledge, dan tacit knowledge mengandung fitur asemen untuk latihan ujian dan simulasi termasuk juga tahapan dalam pemecahan suatu masalah. Contoh dari multimedia pembelajaran mandiri adalah, Macromedia Authorware atau Adobe Flash.

6. Perangkat Multimedia

Multimedia kits merupakan suatu kumpulan materi atau pembelajaran yang meliputi lebih dari satu jenis media yang diorganisir di sekitar topik utama. Didalamnya juga termasuk CD ROOM, KASET, AUDIO, GAMBAR DIAM, STUDY CETAK, TRANSPARANSI OVERHEAD. Tujuannya adalah untuk proses presentasi di dalam kelas keunggulan: multimedia kits membakitkan karena mereka multi sensorik. Multimedia kits menjadi maksime serta ideal sebagai media untuk merangsang kerja kelompok proyek kecil, kits mempunyai keunggulan tersendiri, yaitu dapat di angkut dan digunakan di luar kelas (logiostik).

7. Multimedia Hypermedia

Ini merupakan perpanjangan dari Hipertex, yaitu dokumen yang terdiri dari teks, audio, informasi visual yang disimpan dalam komputer. Contoh dari hypermedia yaitu ketika dalam suatu pembelajaran menggunakan link pada sebuah web.

8. Multimedia (Virtual Realitas)

Virtual reality atau dikenal juga dengan realitas maya merupakan teknologi untuk pembelajaran yang memungkinkan pengguna untuk dapat berinteraksi dalam suatu lingkungan yang disimulasikan oleh komputer. Entah itu suatu lingkungan yang sebenarnya ditiru atau benar-benar suatu lingkungan yang hanya ada dalam imajinasi.

Multimedia hampir bisa digunakan pada semua bidang kehidupan manusia, apalagi dengan perkembangan teknologi yang sangat pesat sekarang ini memungkinkan multimedia selalu hadir dan menjadi kebutuhan setiap orang. Multimedia sering dijumpai pada kehidupan sehari-hari kita, contoh **pengaplikasian dari multimedia** diantaranya adalah :

- **Bisnis**

Aplikasi multimedia dalam bisnis meliputi presentasi, pengajaran, pemasaran, periklanan, demo produk, database, catalog, instant message dan komunikasi jaringan. Tidak ketinggalan video conference, yang memungkinkan adanya tatap muka tanpa harus berada didalam suatu tempat yang sama. Dengan adanya aplikasi multimedia inilah, perusahaan - perusahaan dapat menjalani bisnisnya lebih lancar.

- **Pendidikan**

Aplikasi multimedia dalam bidang ini mengubah proses belajar mengajar yang konvensional menjadi lebih menarik dan interaktif, sehingga proses belajar – mengajar tidak terlalu monoton seperti selama ini yang dilakukan di sekolah-sekolah pada umumnya. Contohnya seperti aplikasi multimedia untuk meningkatkan kemampuan membaca pada anak – anak. Aplikasi tersebut dapat disisipkan animasi – animasi yang tentunya menarik bagi anak-anak sehingga dapat membantu meningkatkan minat mereka dalam membaca maupun belajar hal lainnya.

- **Rumah**

Aplikasi multimedia dalam rumah sangat beraneka ragam. Contohnya komputer yang menggunakan CD-ROM atau DVD-ROM sebagai alat penyalur multimedia. Terdapat juga mesin-mesin permainan yang menggunakan televisi sebagai penyalur multimedia, seperti : Sega, Nintendo, playstation, x-box, dan sebagainya dimana semua permainannya menggunakan elemen-elemen multimedia.

- **Tempat umum**

Aplikasi multimedia di tempat-tempat umum dapat berupa kios informasi seperti yang terdapat pada hotel-hotel, pusat perbelanjaan, museum, pusat hiburan dan sebagainya. Aplikasi-aplikasi tersebut dapat berguna untuk memberikan informasi-informasi dan bantuan mengenai tempat yang bersangkutan. Misalnya kios informasi pada pusat hiburan yang memberikan letak-letak suatu toko atau sarana hiburan yang terdapat didalam cakupan wilayah kios informasi tersebut.

- **Internet**

Multimedia di internet siaran langsung dari ribuan stasiun radio, melihat animasi bagaimana cara kerja sesuatu dan melihat video.

- **Presentasi**

Multimedia memungkinkan seorang presenter beralih dari overhead projector yang menampilkan gambar dan teks yang kaku kepada gambar bergerak, suara dan animasi untuk menghidupkan presentasi yang dibawakan.

- **Lisensi Dokumen**

Seluruh dokumen di komputer dapat digunakan, dimodifikasi dan disebarakan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu.

- **Kios**

Kios yang interaktif dengan layar sentuh dapat menyediakan berbagai informasi dengan lengkap ditempat-tempat umum, misalnya informasi mengenai produk, informasi yang disajikan jauh lebih menarik bagi pengguna dibandingkan informasi yang tercetak.

- **Tutorial**

Multimedia dengan cepat telah menjadi dasar pelatihan berbasis komputer, sebagai contoh perusahaan menyediakan tutorial yang interaktif bagi karyawan baru untuk mempelajari prosedur-prosedur di perusahaan. Dan lain sebagainya

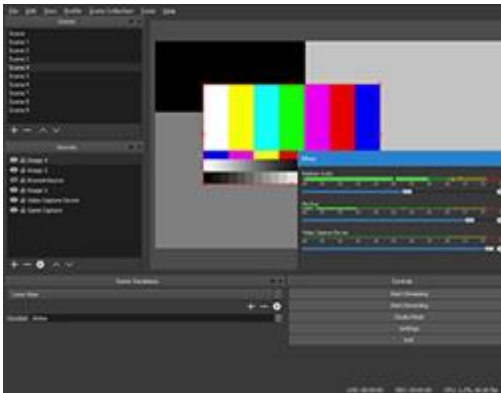
C. Percobaan

Pada percobaan kali ini kita akan menerapkan salah satu jenis multimedia dalam proses pembelajaran yaitu jenis multimedia linear, yaitu dengan cara membuat video tutorial atau video ajar dengan menggunakan microsoft power point dan Open Broadcaster Software (OBS).

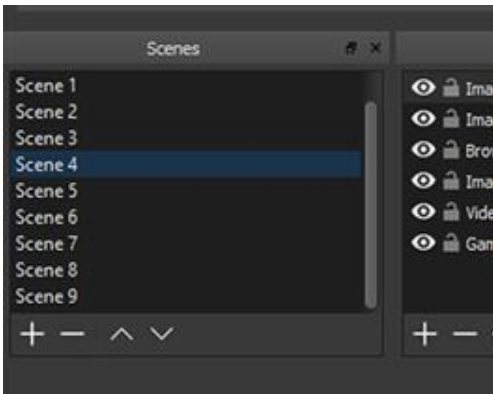
Open Broadcaster Software (OBS) adalah perangkat lunak untuk perekaman (dan streaming) secara lintas-platform. Perangkat lunak bersifat open-source dan dapat digunakan secara gratis. OBS memungkinkan perekaman slide perkuliahan digabung dengan rekaman video, perangkat lunak lain dan lain sebagainya. Hal ini menyebabkan OBS sebagai perangkat lunak yang sangat fleksibel dan ampuh untuk membuat video ajar.

Keunggulan OBS adalah:

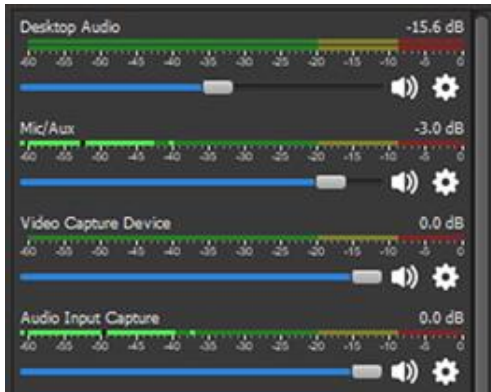
1. Performa yang baik melakukan capture dan mixing video atau audio secara real time. Membuat scenes yang terdiri dari banyak sumber termasuk capture window, images, text, browser windows, webcam, capture card dan lain lain



2. Jumlah scene yang tidak dibatasi dan dapat dialihkan dari satu scene ke scene yang lain dengan mulus menggunakan transisi khusus.



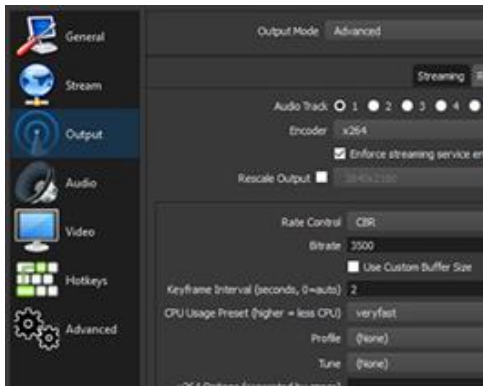
3. Audio mixer yang dapat memfilter suara seperti noise gate, noise suppression, dan gain. Dan dapat mengendalikan secara penuh dengan dukungan plugin VST.



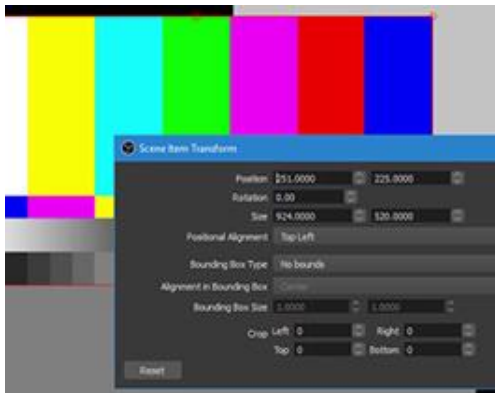
4. Opsi konfigurasi yang powerful dan mudah digunakan. Menambah Sources baru, menduplikasi yang sudah ada, dan mengatur propertiesnya dengan mudah.



5. Panel Pengaturan yang Efisien memberi Anda akses ke beragam opsi konfigurasi untuk mengubah setiap aspek siaran (broadcast) atau rekaman Anda.



6. UI 'Dock' modular memungkinkan Anda mengatur ulang tata letak persis seperti yang Anda inginkan. Anda bahkan dapat mengeluarkan setiap Dock individual ke jendelanya sendiri.

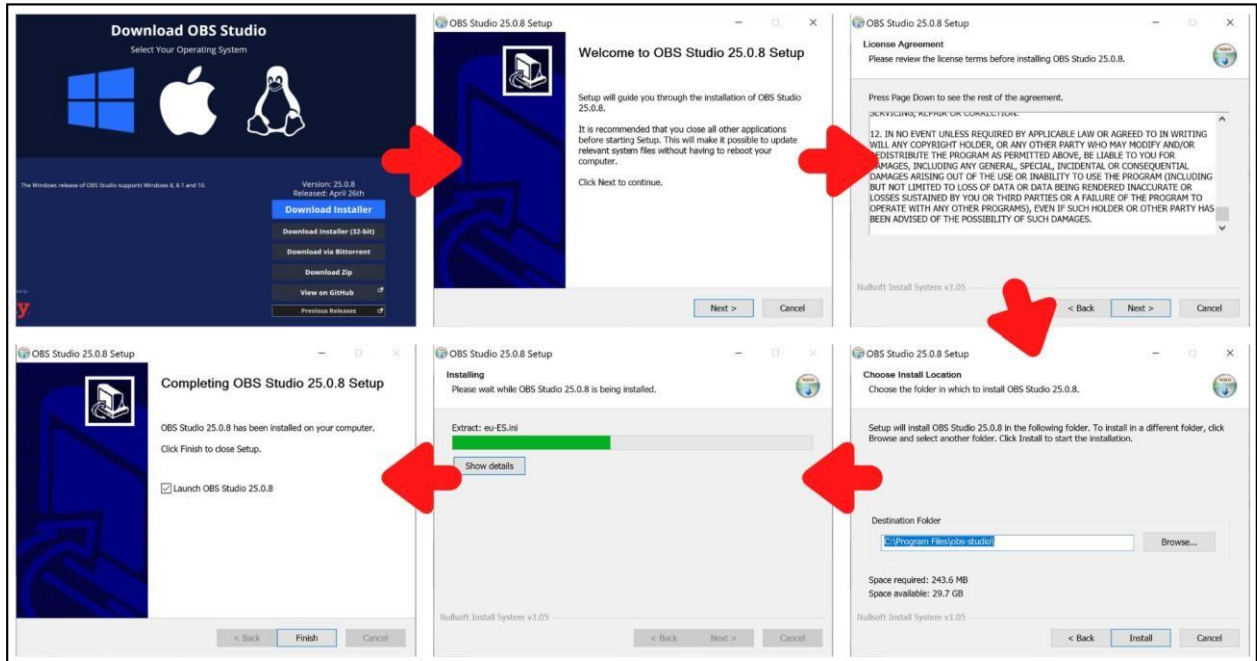


Langkah-Langkah Praktikum :

1. Instalasi OBS

Unduh OBS dari <https://obsproject.com/download> sesuai dengan sistem operasi yang Anda gunakan dan lakukan instalasi.

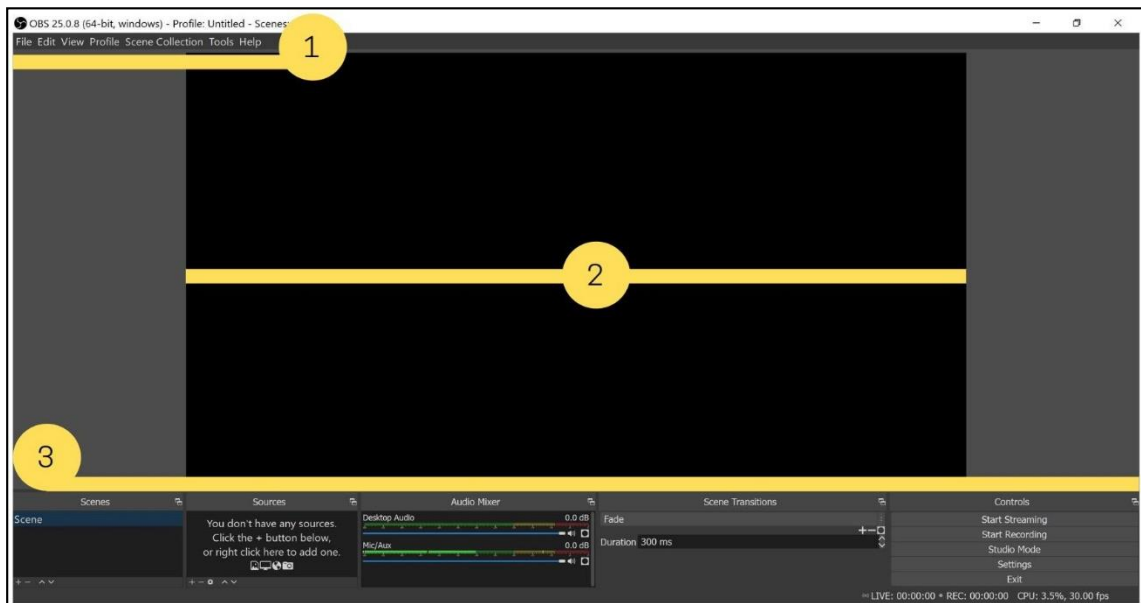
Contoh instalasi untuk sistem operasi Windows ditunjukkan pada Gambar 1.



Gambar 1 Contoh instalasi untuk sistem operasi Windows

Setelah proses instalasi selesai akan muncul tampilan OBS Studio seperti yang ditunjukkan pada Gambar 2 (halaman selanjutnya). OBS Studio terdiri dari beberapa area utama sebagai berikut (Gambar 2):

WORK AREA



Gambar 2 Tampilan OBS

1. Menu Bar

Menu bar berisi menu untuk dengan perintah-perintah yang dikelompokkan berdasarkan topik atau kategori yaitu *File, Edit, Profile, Scene Collection, Tools, Help*.

2. **Canvas Preview Area**

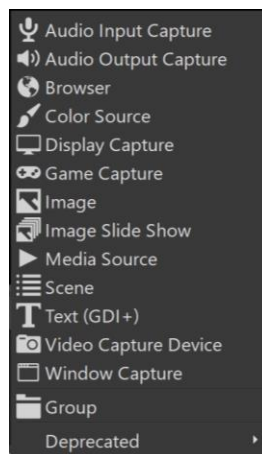
Canvas preview area menyediakan *real-time preview* dari *output*.

3. **Scene Management and Control Options**

Salah satu fitur yang terdapat di OBS adalah sistem *Scene* dan *Sources* yang dapat digunakan untuk mengatur *stream layout*, menambahkan *webcam* dan perangkat atau media lain yang diinginkan dalam *output*.

Klik kanan pada kotak di bawah *Scene* (atau gunakan tanda tambah di bagian bawah) untuk menambahkan *scene*. *Scene* dapat dibuat sebanyak yang Anda inginkan, masing-masing *scene* harus diberi nama yang berbeda agar setiap *scene* dapat dibedakan dengan mudah. Tombol panah pada kolom *Scene* dapat digunakan untuk mengubah urutan *scene*. Sebagai catatan penting, semua *scene* dan *sources* bersifat global di OBS Studio, sehingga *scene* dan *sources* tidak dapat memiliki nama yang sama, misalnya jika Anda memberi nama “*Image*” untuk suatu *sources*, Anda tidak dapat memiliki *scene* dengan nama “*Image*”.

Setelah suatu *scene* dibuat, klik kanan di kotak *Sources* (atau gunakan tanda tambah di bagian bawah) untuk menambahkan hal-hal yang ingin ditampilkan pada *output*, baik itu suatu *window* tertentu, *webcam*, gambar, teks, dan masih banyak lagi. Pada OBS Studio terdapat banyak *sources* berbeda yang dapat Anda pilih, diantaranya yaitu (Gambar 3):



Gambar 3 Pilihan sources pada OBS Studio

- **Audio Input/Output Capture**
Sources ini memungkinkan Anda untuk menambahkan perangkat *input* atau *output* audio (misalnya mikrofon atau headset) pada suatu *scene* tertentu.
- **Browser**
Source ini dipilih jika Anda ingin menampilkan *web browser* pada suatu *scene*.

- **Color Source**
Source ini dapat digunakan untuk menambahkan warna, misalnya warna latar belakang pada suatu *scene*.
- **Display Capture**
Display capture digunakan untuk menampilkan keseluruhan tampilan monitor pada suatu *scene*. Anda hanya dapat menambahkan satu *source display capture* per *display*. Jika Anda membutuhkan *display* yang sama dalam beberapa *scene*, pastikan Anda menggunakan *Add Existing*.
- **Game Capture**
Source ini memungkinkan Anda menampilkan langsung *game* yang sedang Anda mainkan. *Game Capture* adalah cara paling efisien untuk menambahkan game pada OBS.
- **Image**
Source ini memungkinkan Anda untuk menambahkan berbagai jenis gambar pada *scene*. *Image source* mendukung format gambar berikut: .bmp, .tga, .png, .jpeg, .jpg, dan .gif.
- **Image Slide Show**
Source ini memungkinkan Anda untuk menambahkan beberapa gambar yang akan diputar sebagai *slide show*. Untuk menggunakan *source* ini, klik tanda + untuk menambahkan file individual atau direktori yang akan dimuat. Setelah Anda memiliki semua gambar yang Anda inginkan, Anda dapat mengonfigurasi bagaimana gambar-gambar tersebut akan ditampilkan.
- **Media Source**
Media source adalah digunakan untuk menambahkan semua jenis media ke dalam *scene*. Jenis *file* yang didukung diantaranya yaitu:
 - Video: .mp4, .ts, .mov, .flv, .mkv, .avi, .gif, .webm
 - Audio: .mp3, .aac, .ogg, .wav
 Untuk menambahkan *media source*, klik tombol *Browse* untuk memilih *file*, atau hapus centang pada kotak "*Local File*" untuk menambahkan URL (*remote files*). Untuk *remote files*, URL dituliskan dalam kotak *Input* dan umumnya kotak *Format Input* dapat dibiarkan kosong.
- **Scene**
Source ini memungkinkan Anda untuk menambahkan *scene* sebagai *source*.
- **Text (GDI+)**
Text source dapat digunakan untuk menambahkan teks pada *scene*. Teks ini dapat dibaca dari suatu *file* atau diketikkan secara langsung pada

kotak berlabel "Text".

- **Video Capture Device**

Video capture device source memungkinkan Anda untuk menambahkan berbagai perangkat video, termasuk tetapi tidak terbatas pada *webcam*.

- **Window Capture**

Window capture digunakan untuk menampilkan *window* tertentu dan isinya. Keuntungan menggunakan sumber ini daripada *Display Capture* yaitu hanya *window* yang dipilih yang akan ditampilkan, bahkan jika terdapat *window* lain di depannya.

- **Deprecated**

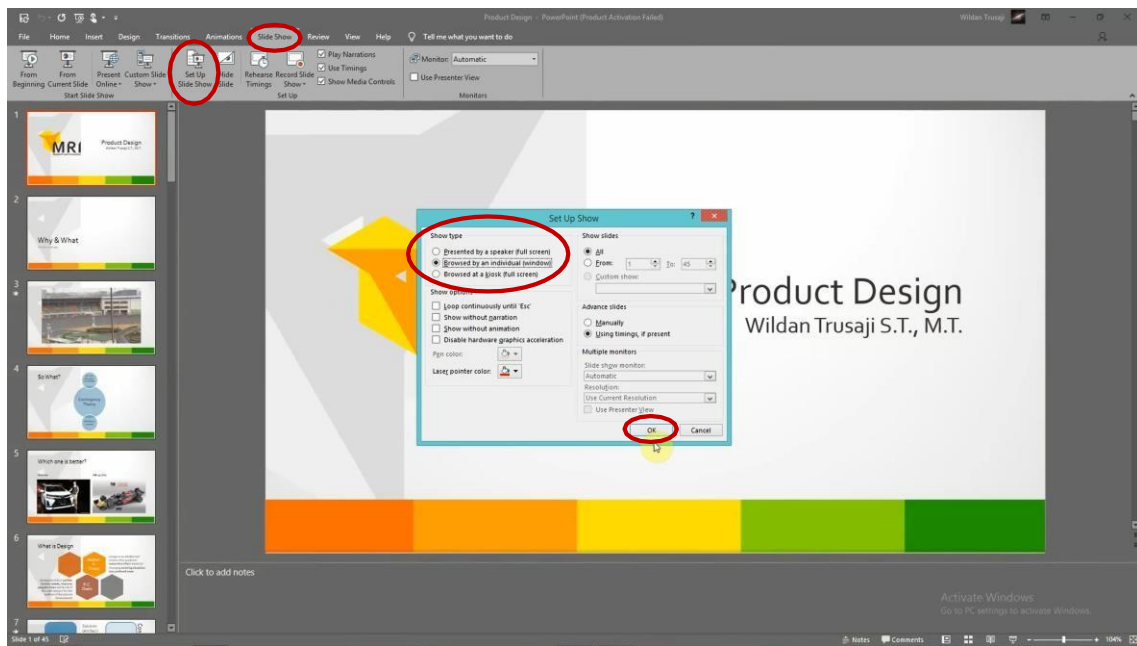
Deprecated merupakan daftar *source* yang saat ini tidak digunakan lagi. *Source* yang tercantum di sini tidak boleh digunakan kecuali jika diperlukan. *Source* ini dibiarkan karena alasan *backwards compatibility*.

2. PANDUAN PEMBUATAN VIDEO AJAR TIPE TEMBAGA/PERUNGGU

Video ajar tipe perunggu berisi slide materi dan narasi tanpa disertai oleh video yang menunjukkan wajah dosen.

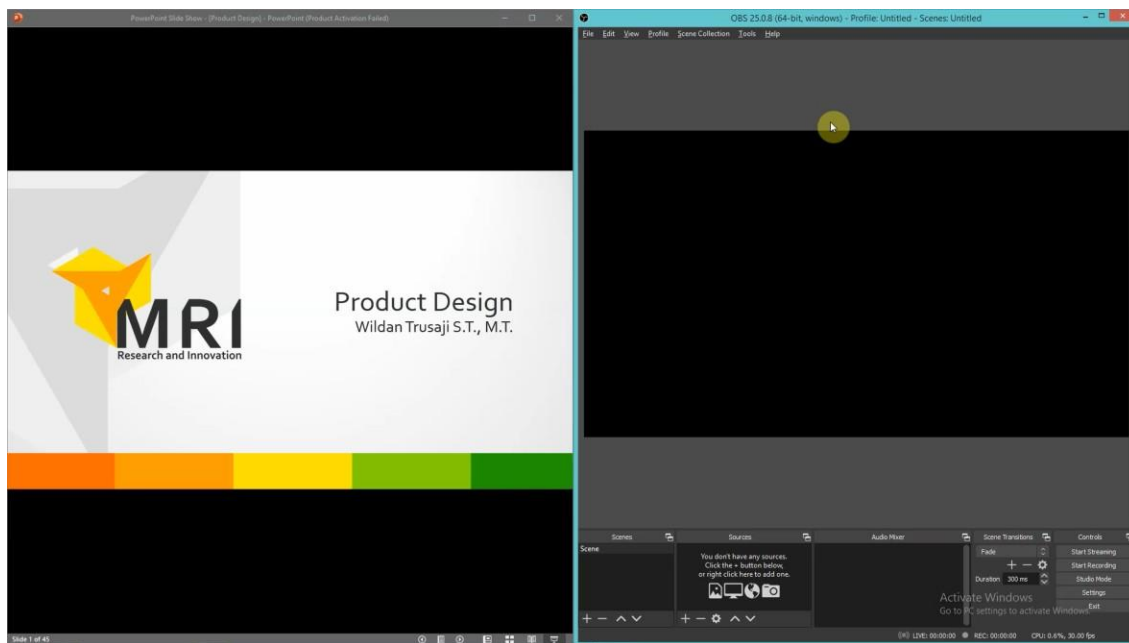
Langkah-langkah pembuatan video ajar dengan ‘Kualitas Perunggu’ adalah sebagai berikut:

1. Sebelum melakukan pengaturan (*setting*) pada OBS, terlebih dahulu buka *file* PPT bahan ajar yang nantinya akan ditampilkan.
2. Lakukan pengaturan *Slide Show* dari PPT yang sudah dibuka dengan memilih *menu bar Slide Show* → *Set Up Slide Show* hingga muncul kotak dialog “*Set Up Show*”. Pada bagian “*Show Type*” pilih “*Browsed by an individual (window)*” seperti yang ditunjukkan pada Gambar 4. Setelah itu, klik OK.



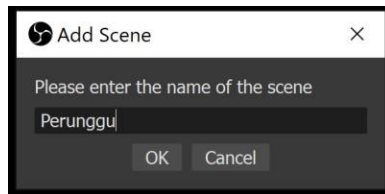
Gambar 4 Kotak dialog Set Uo Show

Pengaturan ini dilakukan agar ketika PPT ditampilkan dalam bentuk *slide show* (klik F5), *slide show* yang muncul tidak dalam bentuk *full screen*. Posisi *window* dari PPT ini kemudian diatur agar memenuhi setengah layar PC sehingga pada PC akan terdapat dua tampilan, tampilan kiri yaitu PPT dan tampilan kanan yaitu OBS Studio seperti yang ditunjukkan pada Gambar 5.

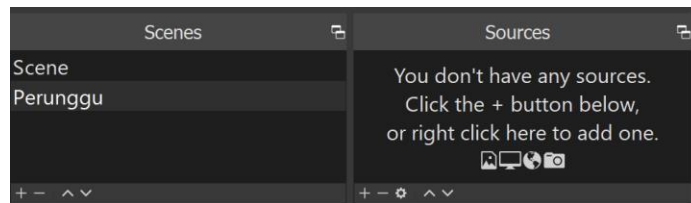


Gambar 5 Tampilan PC

3. Klik tanda “+” pada bagian *Scene* di OBS Studio untuk menambahkan *scene* dan kemudian beri nama *scene* tersebut, misalnya “Perunggu” (Gambar 6). Klik OK. Pada bagian *Scene* nantinya akan muncul *scene* baru dengan nama “Perunggu” (Gambar 7).

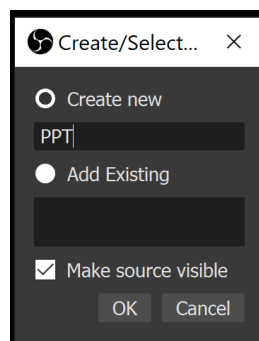


Gambar 6 Kotak dialog Add Scene

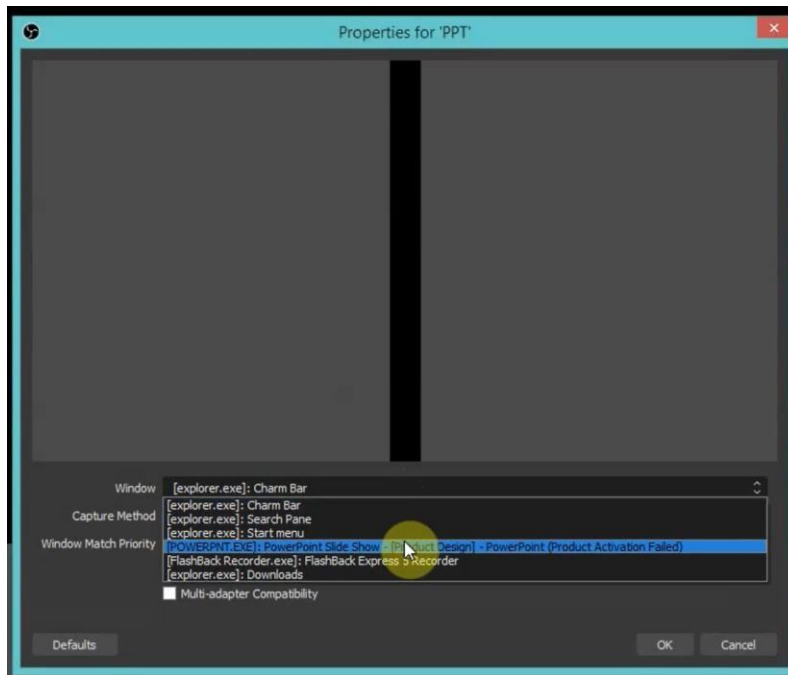


Gambar 7 Daftar scene

4. Pilih *scene* “Perunggu” yang ada pada daftar *scene*, kemudian pada bagian *Sources* klik tanda “+” untuk menambahkan *source* yang akan dimunculkan pada *preview area* di OBS. Karena ingin menampilkan PPT yang sebelumnya sudah dibuka, pilih “*Window Capture*”. Karena *window* PPT ini belum pernah ditambahkan sebelumnya, maka pilih “*Create New*” dan beri nama, misalnya “PPT” (Gambar 8). Klik OK dan kemudian akan muncul kotak dialog “*Properties for PPT*”. Pada bagian “*Window*” pilih *window* yang akan ditampilkan (Gambar 9). Klik OK.



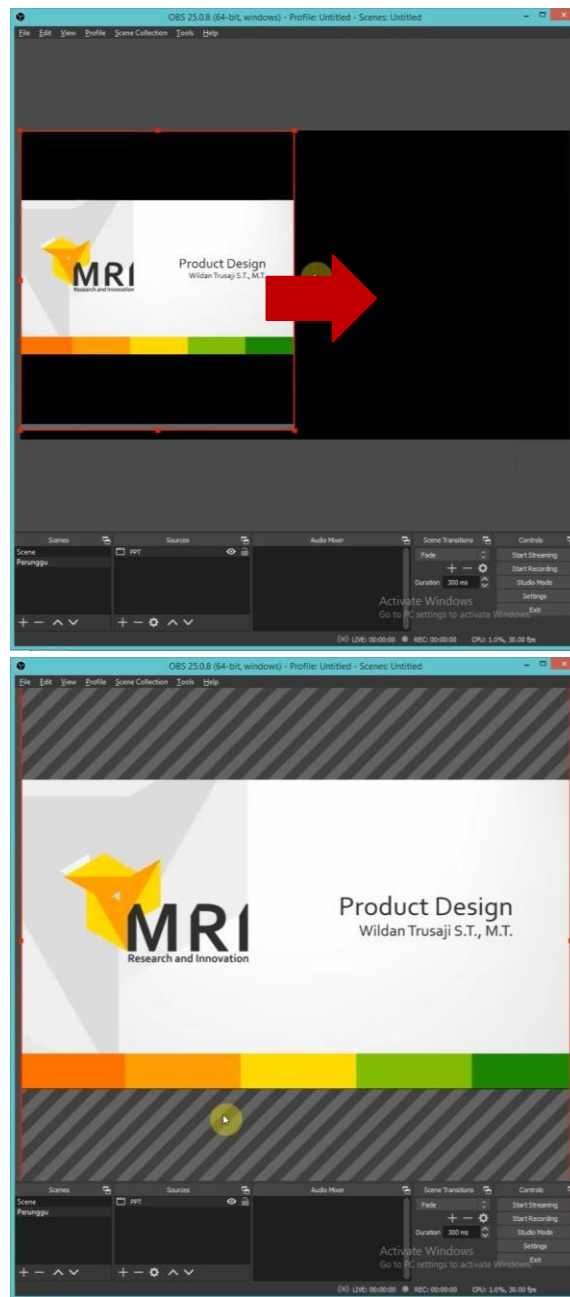
Gambar 8 Menambahkan window capture baru



Gambar 9 Memilih window yang akan ditampilkan

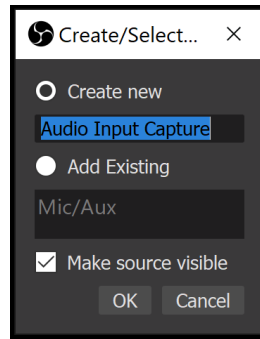
Catatan: *Window capture* ini tidak hanya dapat menampilkan *window* PPT, namun dapat juga menampilkan *window* lain seperti excel, file pdf, dsb.

5. Lakukan pengaturan (*drag*) tampilan PPT agar memenuhi *canvas preview area* pada OBS Studio (Gambar 10).



Gambar 10 Menyesuaikan tampilan PPT

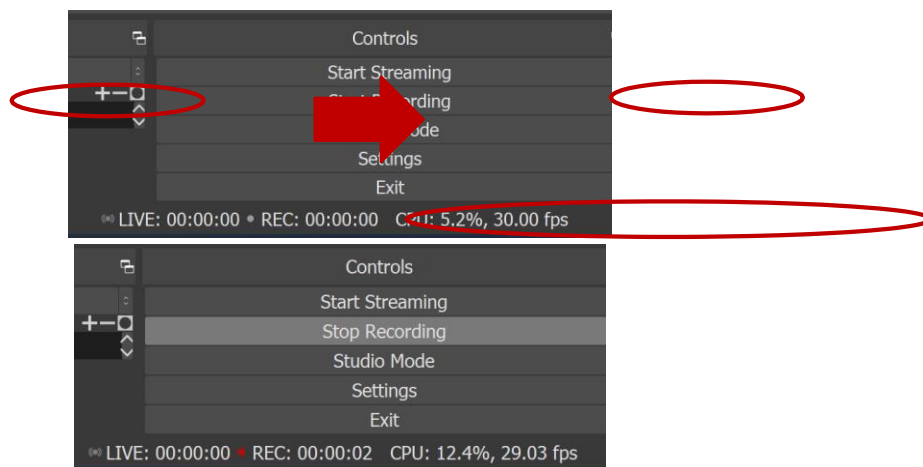
6. Untuk memasukkan suara, klik kembali tanda “+” yang ada pada bagian *Sources* kemudian pilih “*Audio Input Capture*”. Fitur ini digunakan untuk menangkap audio yang masuk ke PC. Beri nama, kemudian klik OK (Gambar 11). Pada kotak dialog “*Properties for Audio Input Capture*” yang muncul, klik OK.



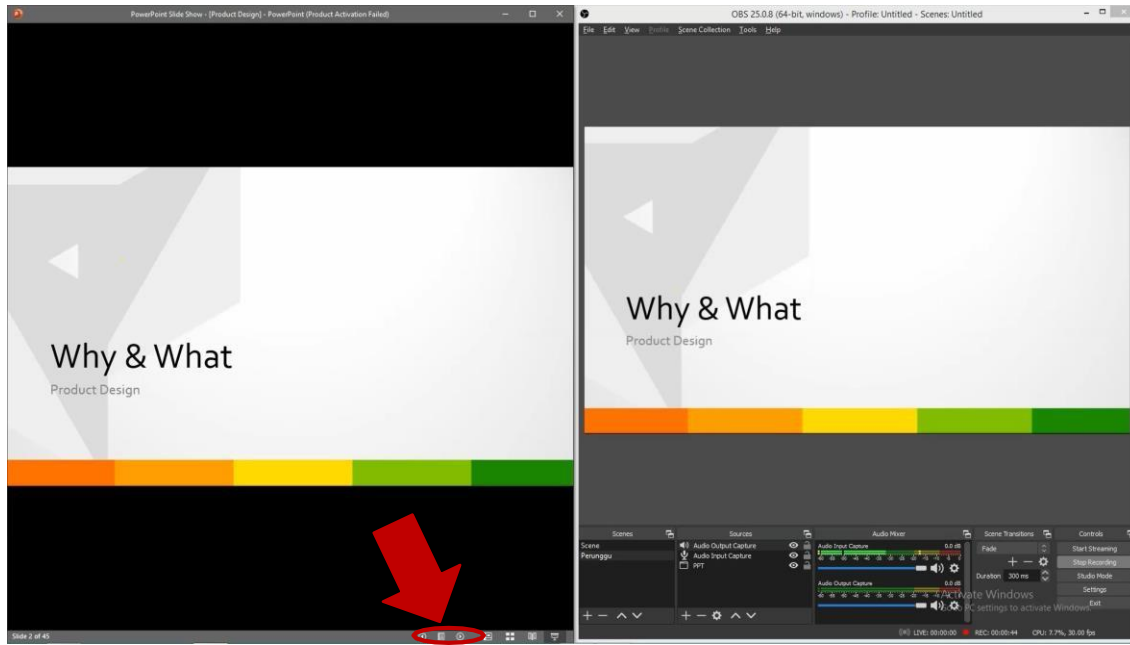
Gambar 11 Menambahkan audio input capture baru

Jika ingin memunculkan dan merekam suara yang ada pada PPT, pada *sources* tambahkan “*Audio Output Capture*”. Fitur ini berfungsi untuk menangkap audio yang keluar dari PC.

7. Pengaturan untuk membuat video ajar dengan “Kualitas Perunggu” selesai. Jika ingin memulai proses rekaman, pada bagian *Controls* klik “*Start Recording*”. Pada bagian kanan bawah akan muncul keterangan *record* (Gambar12). Untuk menavigasi PPT, klik *window* PPT, kemudian lakukan operasi *next* atau *back* seperti menggunakan PPT pada umumnya (Gambar 13). Dengan OBS, audio yang direkam dan video yang ditampilkan dapat langsung disinkronisasi. Ketika *recording* selesai, klik “*Stop Recording*”. Untuk melihat hasil *recording*, klik menu *File* → *Show Recordings*.



Gambar 12 Start recording



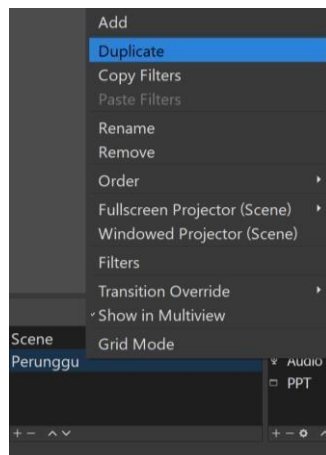
Gambar 13 Navigasi PPT

3. PANDUAN PEMBUATAN VIDEO AJAR KUALITAS PERAK

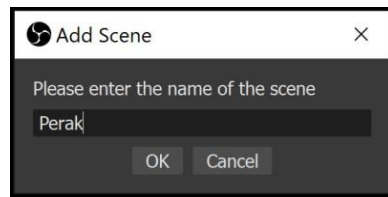
Video ajar tipe perak berisi *slide* materi dan narasi disertai oleh *preview webcam* yang menunjukkan wajah dosen.

Langkah-langkah pembuatan video ajar dengan ‘Kualitas Perak’ adalah sebagai berikut:

1. Jika sebelumnya Anda sudah membuat video ajar dengan ‘Kualitas Perunggu’, pembuatan video ajar dengan ‘Kualitas Perak’ tidak perlu dilakukan dari awal lagi. Klik kanan *scene* ‘Perunggu’ → *Duplicate* (Gambar 14). Beri nama *scene* tersebut, misalnya ‘Perak’ (Gambar 15). Dengan cara ini, ketika *scene* ‘Perak’ dibuat, *scene* tersebut sudah memiliki *source* yang sama dengan *scene* ‘Perunggu’.

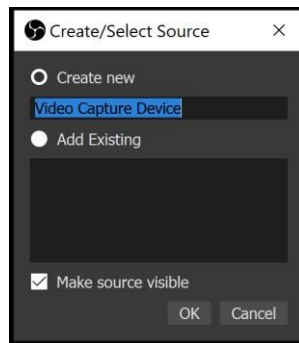


Gambar 14 Menduplikasi scene

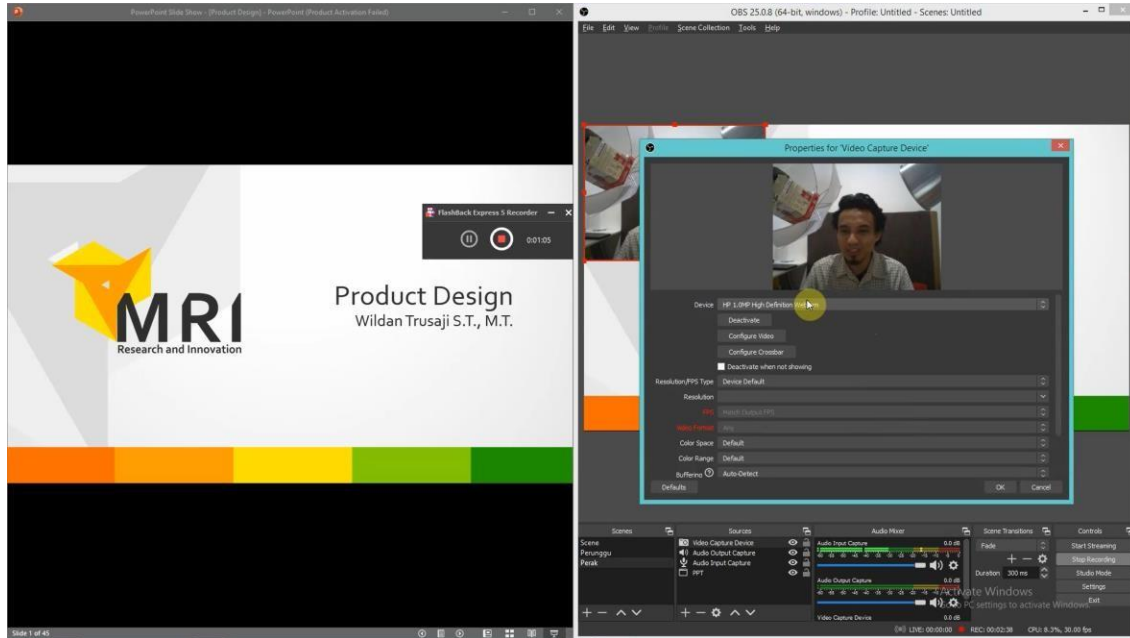


Gambar 15 Memberi nama scene baru

2. Video dengan “Kualitas Perunggu” dapat ditingkatkan menjadi “Kualitan Perak” dengan menambahkan *talking head*. Pilih scene “Perak” yang ada pada daftar *scene*, kemudian pada bagian *Sources* klik tanda “+”. Untuk menambahkan *talking head* pada *scene*, pilih “*Video Capture Device*”. Fitur ini memungkinkan OBS untuk mengambil gambar yang ditangkap oleh *webcam*. Beri nama, kemudian klik OK (Gambar 16). Pada kotak dialog “*Properties for Video Capture Device*” yang muncul, pilih *device* yang akan digunakan (Gambar 17), kemudian klik OK.



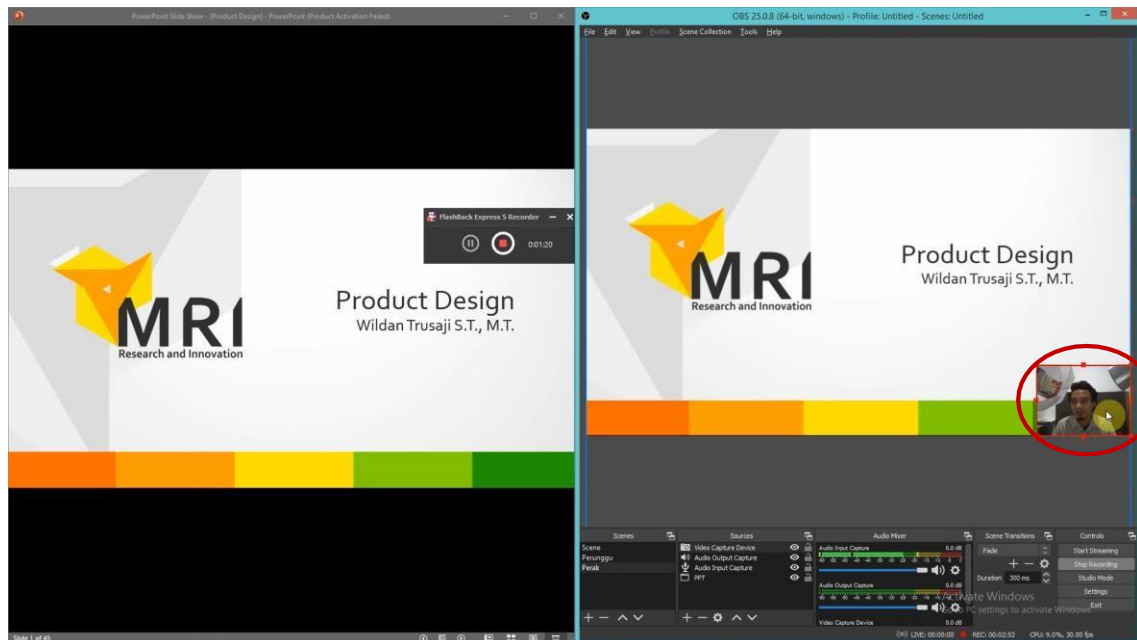
Gambar 16 Menambahkan video capture device baru



Gambar 17 Memilih device yang akan digunakan

3. Atur ukuran dan posisi tampilan webcam (Gambar 18).

Saran: Jika ingin kualitas video *talking head* lebih baik, Anda bisa memanfaatkan lampu meja belajar dan mengarahkan cahaya dari lampu meja belajar tersebut ke wajah Anda (Gambar 19). Hal ini lebih baik daripada tidak menggunakan pencahayaan sama sekali.



Gambar 18 Mengatur ukuran dan posisi tampilan webcam



Gambar 19 Memanfaatkan lampu meja belajar

4. Pengaturan untuk membuat video ajar dengan “Kualitas Perak” selesai. Jika ingin memulai proses rekaman, pada bagian *Controls* klik “*Start Recording*”. Ketika *recording* selesai, klik “*Stop Recording*”. Untuk melihat hasil *recording*, klik menu *File*

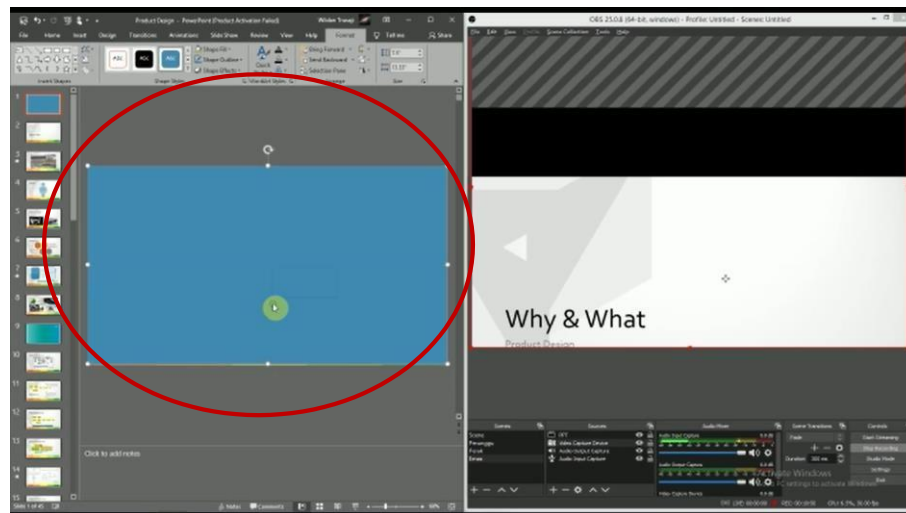
→ *Show Recordings*.

4. PANDUAN PEMBUATAN VIDEO AJAR KUALITAS EMAS

Pada video ajar tipe emas berisi *slide* materi menyatu dengan video yang menunjukkan wajah dosen.

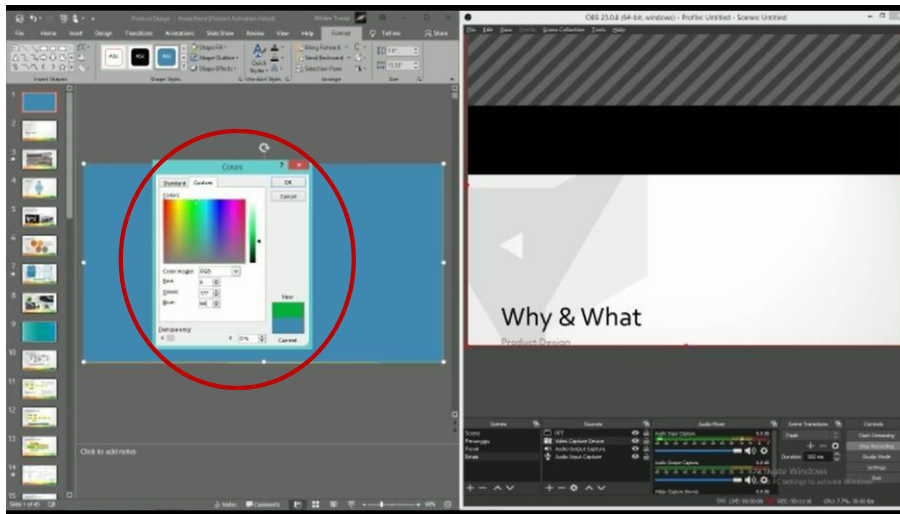
Langkah-langkah pembuatan video ajar dengan “Kualitas Emas” adalah sebagai berikut:

1. Sebelum melakukan pengaturan (*setting*) pada OBS, terlebih dahulu lakukan pengaturan pada *file* PPT bahan ajar yang nantinya akan ditampilkan. Ubah warna *background* PPT yang akan ditampilkan menjadi warna *green chroma key* dengan langkah-langkah sebagai berikut:
 - a) Buat kotak seperti yang ditunjukkan pada Gambar 20.



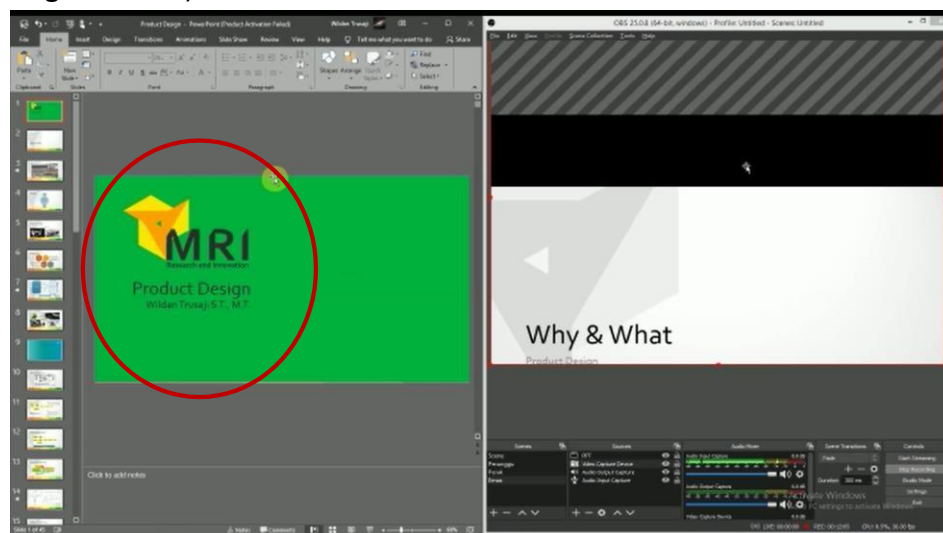
Gambar 20 Membuat kotak pada PPT

- b) Pilih *Fill* → *More Fill Colors* → Atur ukuran RGB (Red=0, Green=177, Blue=64) seperti yang ditunjukkan pada Gambar 21.



Gambar 21 Mengatur ukuran RGB

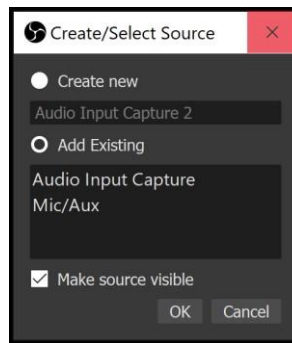
- c) Klik kanan pada kotak yang sudah dibuat, kemudian pilih “*Send to Back*” agar kotak berada di belakang tulisan.
- d) Atur PPT agar konten PPT hanya ada pada setengah bagian *slide* (misal: bagian kiri PPT) seperti yang ditunjukkan pada Gambar 22. Hal ini dilakukan karena wajah presenter akan dimunculkan pada setengah bagian lainnya (misal: bagian kanan).



Gambar 22 Mengatur layout PPT

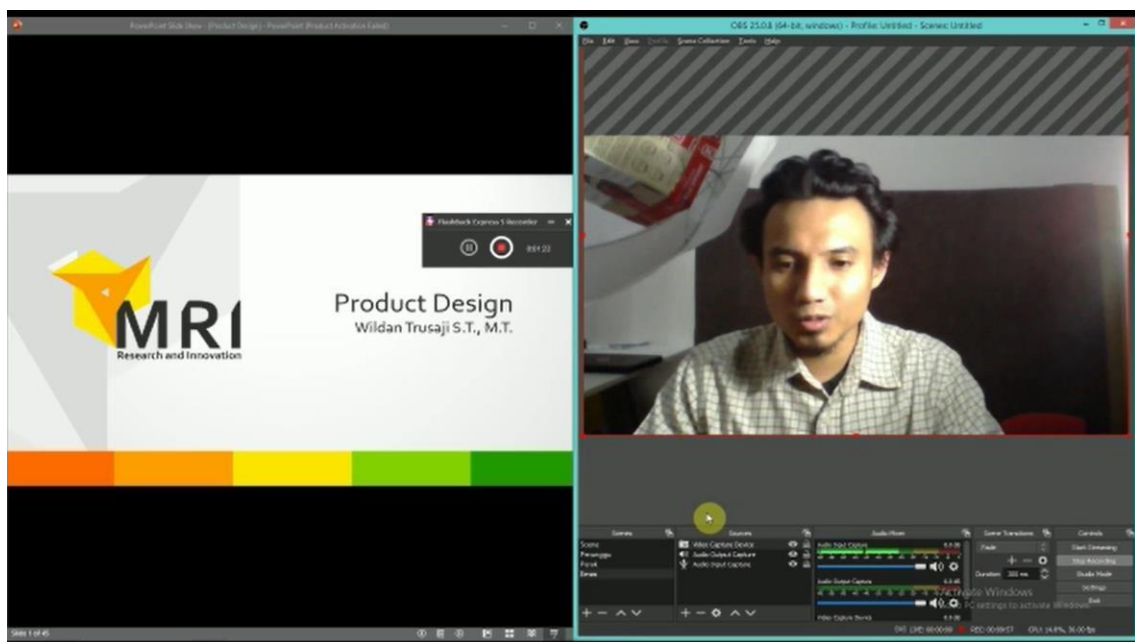
- e) Setelah pengaturan PPT selesai dilakukan, tampilkan kembali PPT dalam bentuk *slide show* (tekan tombol F5).

2. Klik tanda “+” pada bagian *Scene* di OBS Studio untuk menambahkan *scene* dan kemudian beri nama *scene* tersebut, misalnya “Emas”. Klik OK. Pada bagian *Scene* nantinya akan muncul *scene* baru dengan nama “Emas”.
3. Pilih *scene* “Emas” yang ada pada daftar *scene*, kemudian pada bagian *Sources* klik tanda “+”, pilih “*Audio Input Capture*”, dan pilih “*Add Existing*” karena di awal sudah pernah ditambahkan, kemudian klik OK (Gambar 23). Klik kembali tanda “+”, pilih “*Audio Output Capture*”, dan pilih “*Add Existing*” karena di awal juga sudah pernah ditambahkan, kemudian klik OK.



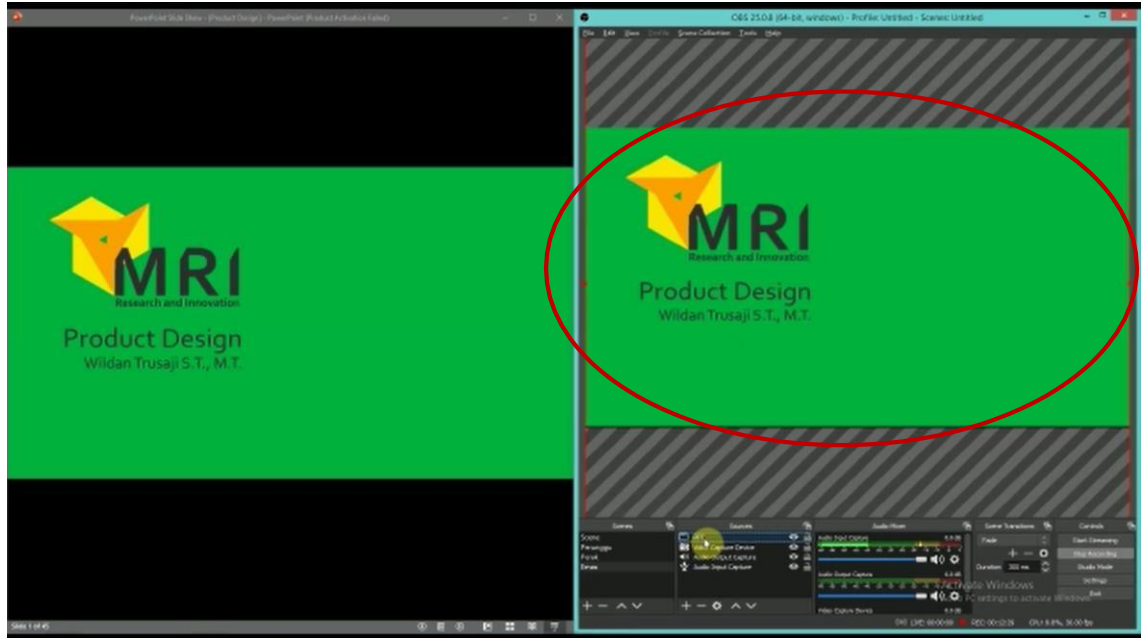
Gambar 23 Add existing audio input capture

4. Setelah itu, tambahkan *webcam*. Untuk menambahkan *webcam*, klik tanda “+”, pilih “*Video Capture Device*”, dan pilih “*Add Existing*” karena sebelumnya sudah pernah ditambahkan, kemudian klik OK. Atur ukuran dan posisi tampilan *webcam* agar memenuhi *canvas preview area* (Gambar 24).



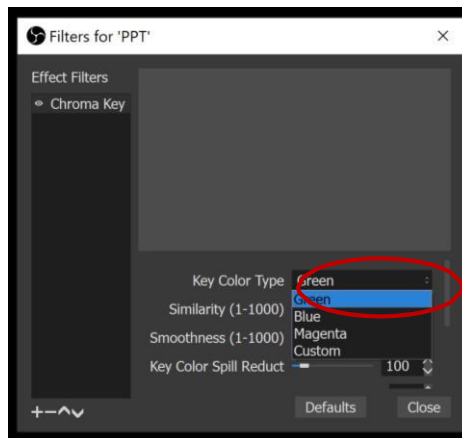
Gambar 24 Mengatur ukuran dan posisi webcam

5. Untuk menampilkan PPT, klik tanda “+” dan pilih “Window Capture”. Jika PPT ini sudah pernah ditambahkan sebelumnya, maka pilih “Add Existing”. Klik OK. Atur ukuran dan posisi tampilan PPT agar memenuhi *canvas preview area* (Gambar 25).

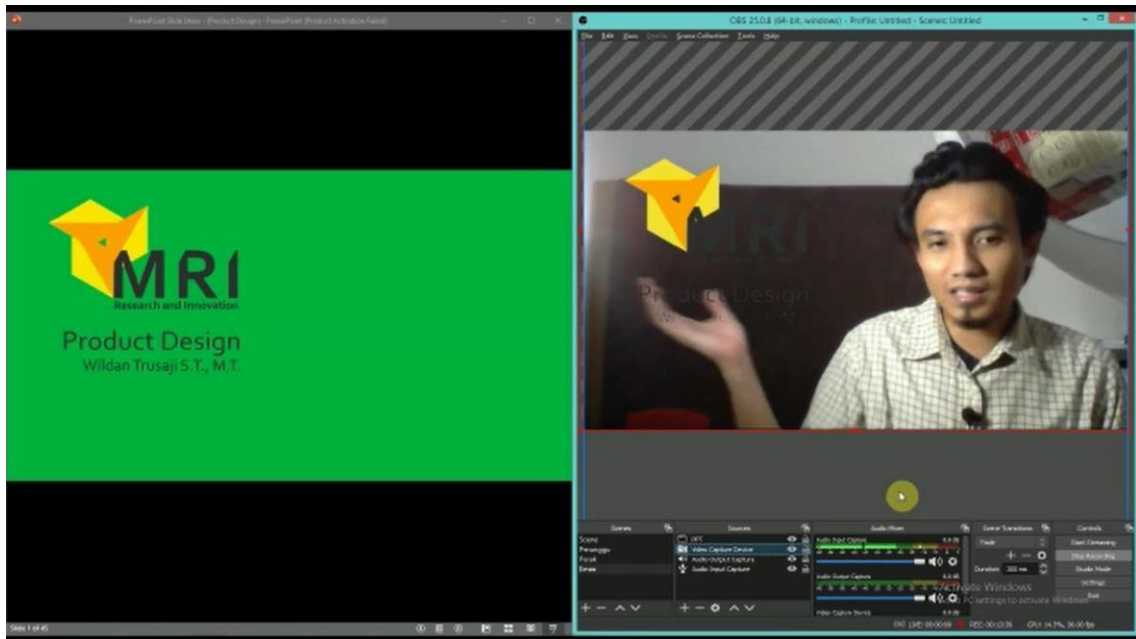


Gambar 25 Mengatur ukuran dan posisi PPT

6. Klik kanan pada *source* “PPT” dan pilih *Filters*. Klik tanda “+” yang ada pada bagian kiri bawah kotak dialog “Filters for PPT” yang muncul. Pilih “Chroma Key”. Klik OK. Pilih *Green* (Gambar 26). Klik OK, *background* slide yang sebelumnya berwarna hijau berubah menjadi tampilan *webcam* (Gambar XX).

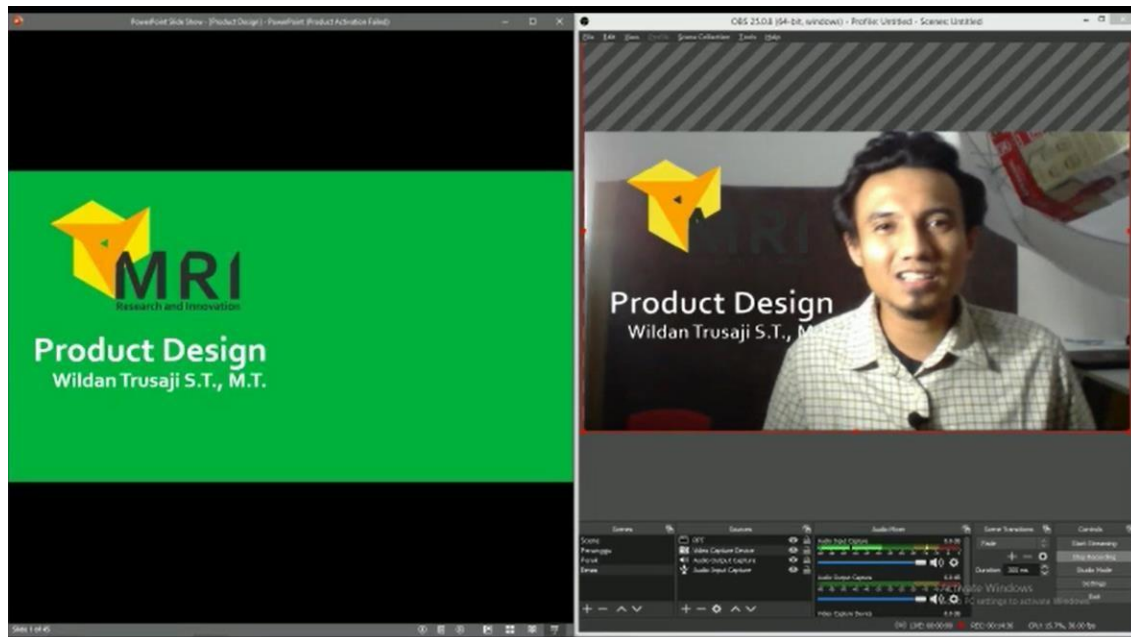


Gambar 26 Menambah filter chroma key pada PPT



Gambar 27 Hasil filter dengan chroma key

7. Untuk melakukan *mirroring* tampilan *webcam*, klik kanan pada source “*Video Capture Device*”, pilih “*Transform*”, dan kemudian pilih “*Transform Horizontal*”.
8. Sesuaikan *font* dan hal-hal lain yang ada di *slide* agar kontras dengan *background* (Gambar 28).



Gambar 28 Menyesuaikan warna font PPT agar kontras dengan background

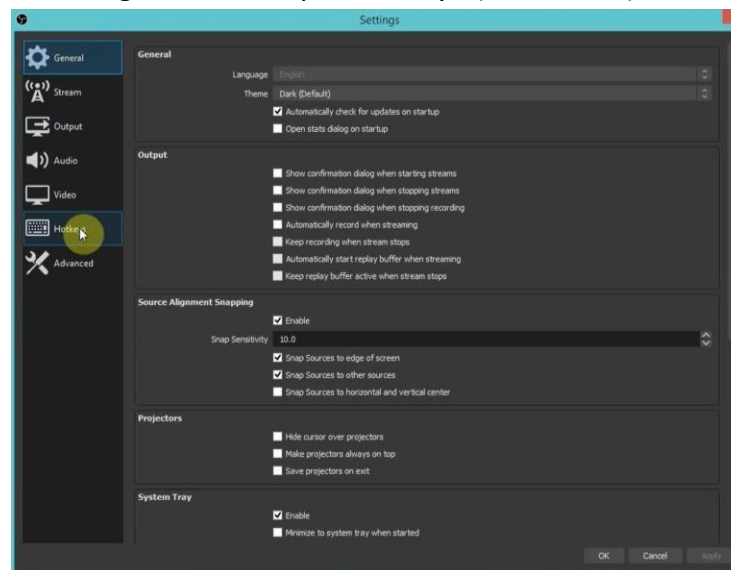
- Pengaturan untuk membuat video ajar dengan “Kualitas Emas” selesai. Jika ingin memulai proses rekaman, pada bagian *Controls* klik “*Start Recording*”. Ketika *recording* selesai, klik “*Stop Recording*”. Untuk melihat hasil *recording*, klik menu *File*

→ *Show Recordings*.

5. HOTKEYS

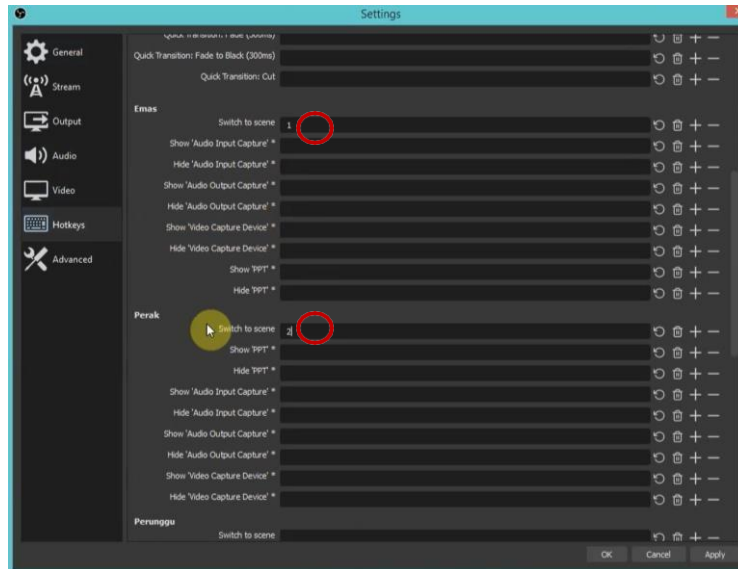
Hotkeys digunakan untuk memudahkan perpindahan *scene* dari *scene* yang satu ke *scene* lainnya. Langkah-langkah untuk mengatur *hotkeys* ini adalah sebagai berikut:

- Klik menu *File*, pilih *Settings*, kemudian pilih *Hotkeys* (Gambar 29).



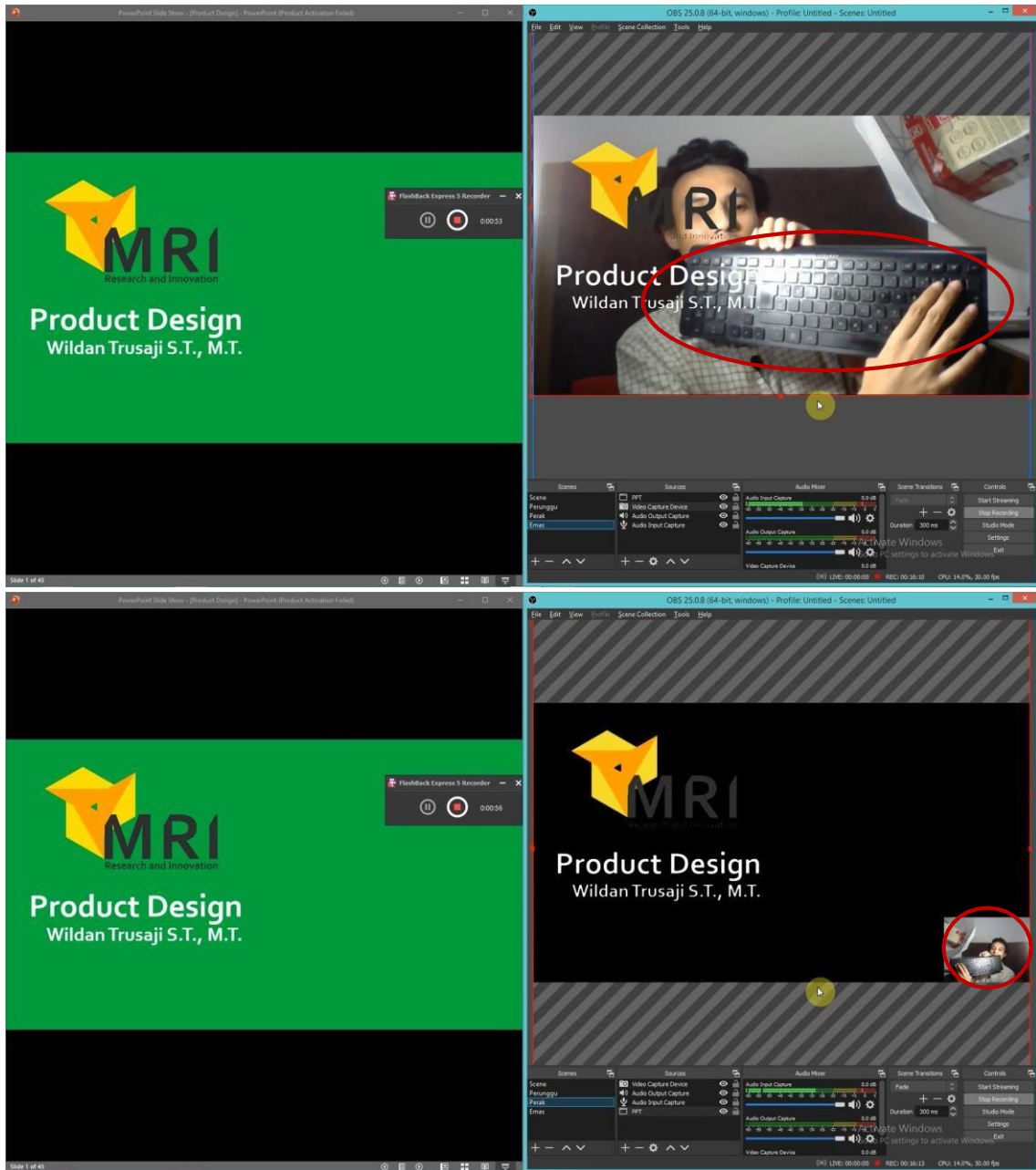
Gambar 29 Hotkeys

- Input short key* yang akan digunakan untuk berpindah ke masing-masing *scene*. Sebagai contoh, *short key* “1” untuk *scene* “Emas” dan *short key* “2” untuk *scene* “Perak” seperti yang ditunjukkan pada Gambar 30. Klik OK.



Gambar 30 Input short key untuk masing-masing scene

3. Dengan menggunakan *hotkeys*, perpindahan dari satu *scene* ke *scene* lainnya dapat dilakukan dengan mudah yaitu dengan menekan *keys* pada *keyboard* sesuai dengan *short key* yang telah ditentukan sebelumnya (Gambar 31). *Hotkeys* ini nantinya dapat memudahkan dosen jika ingin melakukan perpindahan dari PPT ke pdf, excel, atau bahan ajar lainnya. Hal yang perlu dilakukan adalah membuat *scene* yang menampilkan bahan ajar lainnya tersebut.



Gambar 31 Contoh penggunaan hotkeys

D. Tugas Praktikum

1. Buatlah video ajar yang menjelaskan konsep Tugas Akhir kalian, dengan menggunakan OBS dengan video ajar kualitas perunggu, perak, dan emas!
2. Buatlah laporan resmi dari pembuatan video tersebut!

PERTEMUAN 2 & 3

Produksi Konten Multimedia

A. Sub Capaian Pembelajaran Mata Kuliah :

- 1) Mahasiswa dapat menjelaskan tahapan-tahapan dalam menghasilkan konten multimedia

Penilaian Mata Kuliah : UTS (30%), UAS (30%), Tugas (30%), Keaktifan (10%)

B. Dasar Teori

Metodologi yang paling umum dipakai pada proses produksi Multimedia adalah yang biasa disebut dengan alur produksi 3 tahap. Secara umum, proses produksi multimedia dirancang dengan menjalankan 3 tahap sebagai berikut;

1. PRE PRODUCTION

- 1) Concept definition (definisi dan Konsep dari produk Multimedia)
Definisi : iklan Televisi Produk Kuku Bima, iklan cetak, iklan radio, dll
Konsep : *berhubungan untuk menciptakan Daya Tarik (lucu, Menyedihkan, patriotic, sex dll)*
Eg. Figur mBah Marijan dgn Tagline : "Roso-Roso" //
- 2) Design
Perancangan Awal menerapkan Definisi dan konsep Awal biasanya berbentuk "coretan-coretan" dan sketsa
- 3) Production plan (*Perencanaan Produksi*)
Pengembangan dari Design kedalam Naskah, Storyline, Sinopsis, Storyboard, dll
- 4) Documentation (*Pemberkasan*)
Pembuatan Proposal Naskah / pengarsipan (berisi : tujuan pembuatan produk, target pengamat/pasar, Perencanaan Produksi, Property yang dibutuhkan, Penghitungan Anggaran dll)
- 5) Assemble team
Pembentukan Tim Ahli dan Property yang tertulis dalam Documentation
Misal : Cameramen, Fotografer, Animator, Talent, dll)
- 6) Building prototype
Membangun prototipe (*contoh awal produk* sebelum diproduksi)
- 7) Clear right
Penyelesaian dalam hal yang berhubungan dengan per undang-undangan (eg: Ijin Produksi, Pendaftaran Paten/HAKI, dll)
- 8) Client sign-off and funding
Persetujuan dengan Klien/Pemesan beserta Pendanaan.

2. PRODUCTION

- 1) Content creation (*Pembuatan Konten*) : *Pembuatan konten sesuai design*
Eg: take gambar, pembuatan stok shot animasi, dll
- 2) Content processing (*Pemrosesan konten*) : *Pemrosesan konten yg telah diambil secara analog (non computerized)*
Eg : setting tempat, organisasi property/ perlengkapan take gambar
- 3) Software creation (*pembuatan (proses) konten melalui Software(komputer)*)
Eg ; Editing, visual FX
- 4) Integration of content and software (*Penggabungan seluruh konten dengan diproses dalam software finishing*)
Eg : hasil dubbing audio, VFX, ANimasi di render akhir untuk di burning dalam bentuk DVD
- 5) Revise design (*Revisi / Peninjauan ulang berdasar pada Perancangan*)
dari hasil tahap ke 4 di tinjau ulang dan di analisa (belum di buat masal)
- 6) Freeze design (*Penyatuan seluruh project yang telah sesuai dengan design*)
Apabila hasil analisa dan telah direvisi di satukan menjadi sebuah product "jadi" untuk ditunjukkan terhadap klien
- 7) Revise content and software according to final design
(Revisi/Peninjauan Ulang Konten & software di cocokkan dengan Perancangan Akhir (final Design)
- 8) Build Alpha version (*membangun versi Alfa*)
Pembuatan contoh awal produk yang dibuat untuk kalangan tertentu
- 9) First testing (*Uji Coba Produk*)
- 10) Evaluation (*Evaluasi produk*)
- 11) Revise software and content based on evaluation (*Revisi/Peninjauan Ulang Produk berdasar hasil Evaluasi*)
- 12) Build Beta version (*membangun versi Beta*)
Pada tahap ini produk di buat dgn jumlah lebih banyak tetapi didistribusikan untuk kalangan tertentu untuk di coba sebelum di distribusikan secara umum)

3. POST PRODUCTION

- 1) Beta testing (*Uji cob Produk Beta oleh kalangan tertentu*)
- 2) Evaluation (*Evaluasi*)
- 3) Revise software and content (*Peninjauan ulang produk*)
- 4) Release Golden Master (*Peluncuran Produk secara Masal untuk Umum*)
- 5) Archive all production material (*peng Arsip an semua Materi /bahan mentah dari proses Produksi*)

C. Percobaan

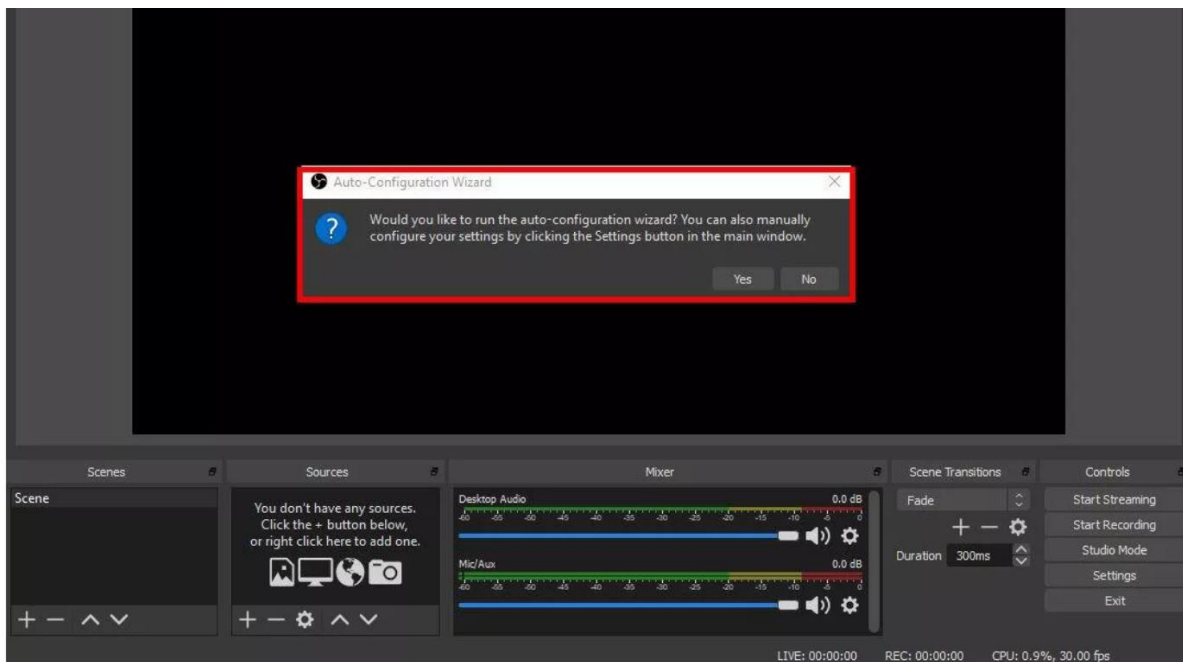
Pada percobaan ini kita akan mencoba membuat sebuah konten multimedia berupa video gaming dan live streaming gaming di Youtube. Dimana proses pembuatan video gaming ini kita lakukan dengan aplikasi OBS. Berikut ini adalah langkah-langkah percobaan yang akan kita lakukan:

1. Membuat Video Gaming dengan aplikasi OBS

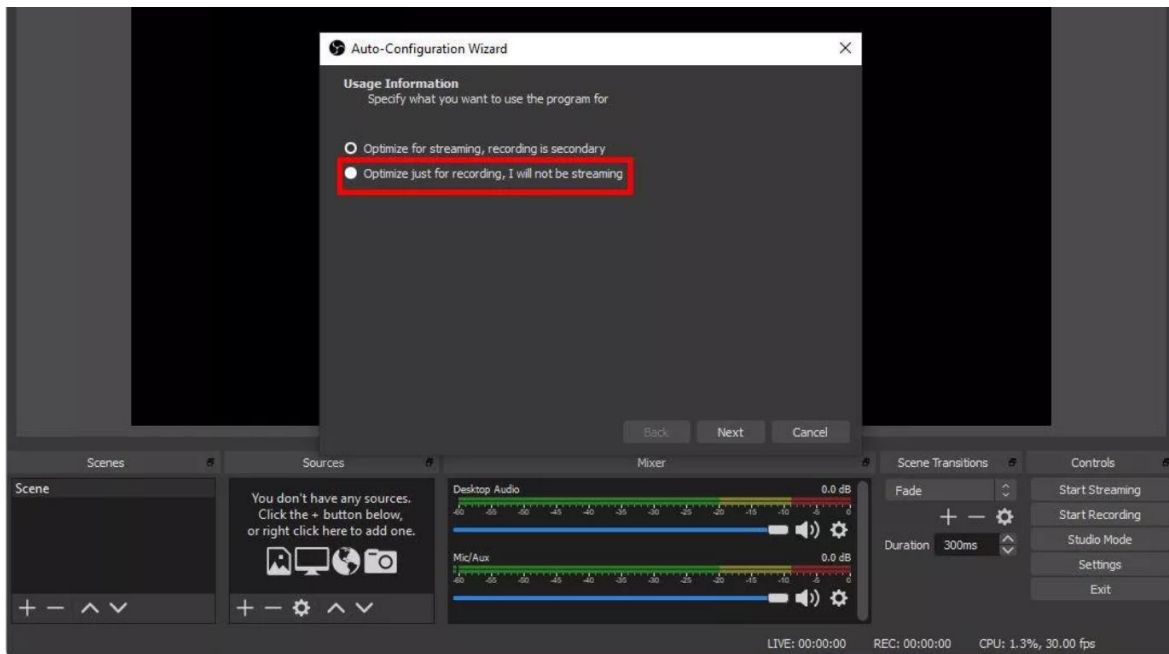
Langkah 1 - Buka Aplikasi OBS Studio

Setelah kamu men-*download* dan *install* OBS Studio di PC atau laptop, kamu bisa membuka aplikasi tersebut.

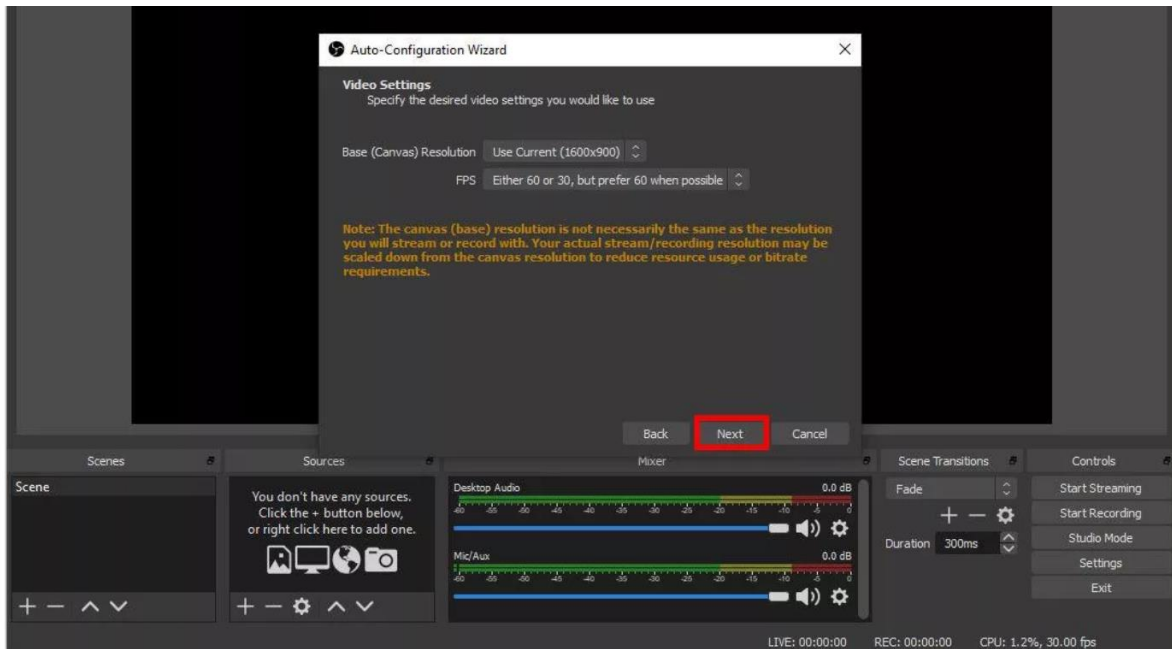
Pada halaman utama, kamu akan diberikan pilihan untuk melakukan **Auto-Configuration**. Kamu bebas mau menjalankan opsi ini atau tidak.



Kalau kamu memilih untuk menjalankan **Auto-Configuration**, kamu bisa memilih akan memprioritaskan OBS Studio untuk *recording* atau *streaming*. Pilih **Optimize just for recording, I will not be streaming** untuk merekam game saja.



Pilih resolusi rekaman dan *frame per second* (FPS) yang kamu inginkan. Kamu bisa memilih untuk merekam sesuai dengan resolusi layar yang kamu lakukan, lalu klik **Next**.

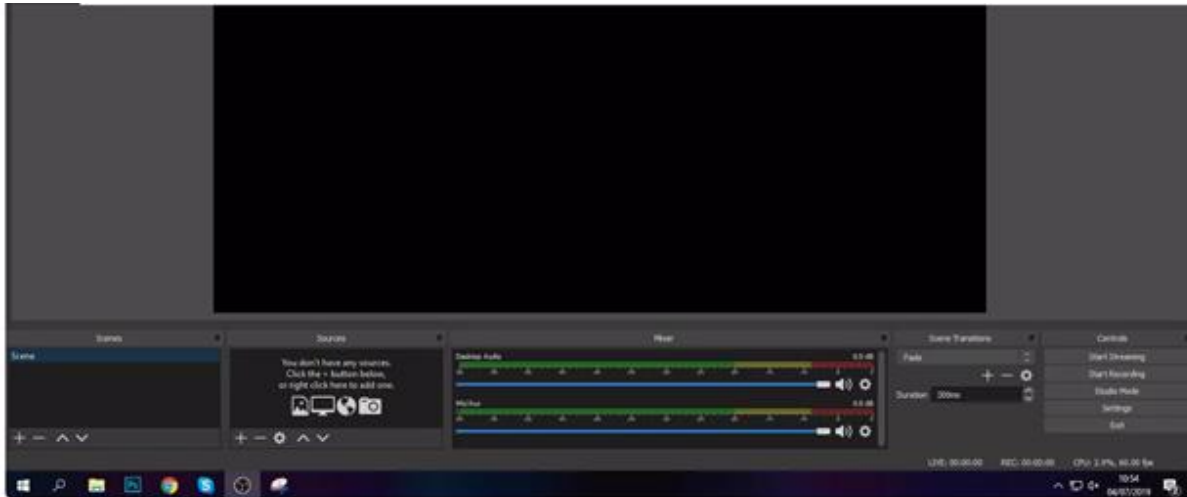


OBS akan mengkonfigurasi rekaman kamu secara otomatis berdasarkan spesifikasi PC yang kamu punya.

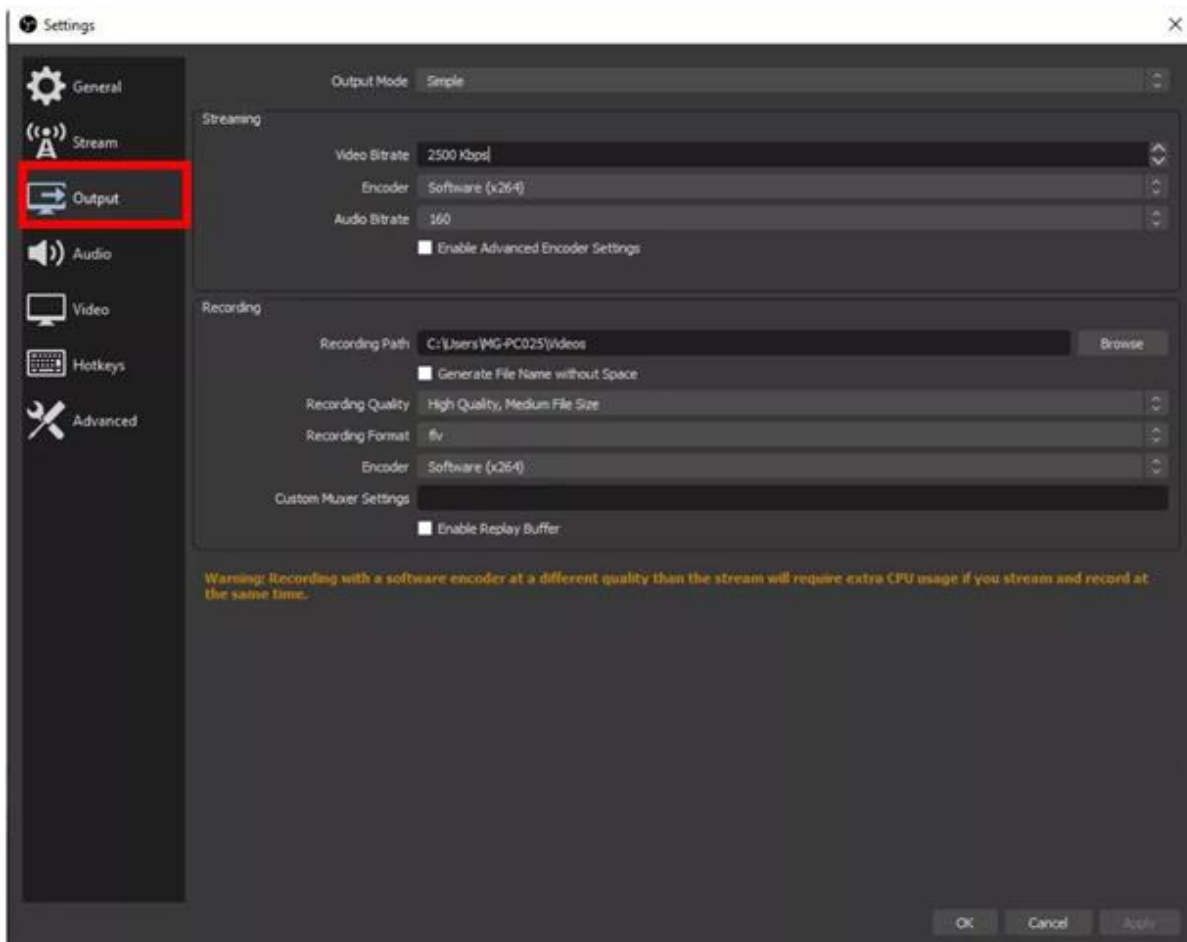
Langkah 2 - Mengatur *Settingan* OBS Studio untuk *Record Game*

Sebelum mulai merekam, kamu bisa mengatur *settingan* OBS Studio sesuai dengan keinginan kamu. Jaka hanya memberi tahu cara simpel merekam video dengan OBS serta rekomendasi setelan OBS.

Pada tampilan utama, klik **File**, lalu pilih opsi **Settings** untuk membuka tampilan setelan.

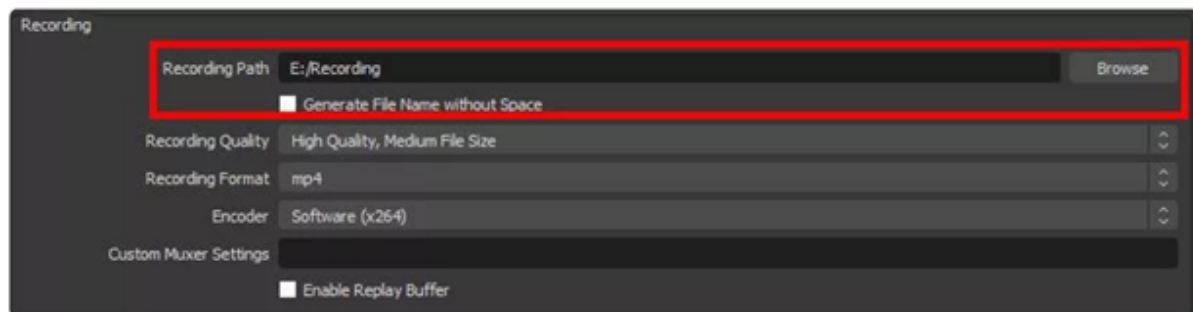


Buka **tab Output** yang ada di sebelah kiri *window*, lalu perhatikan pada segmen **Recording** untuk mengatur setelan *recording* game kamu.

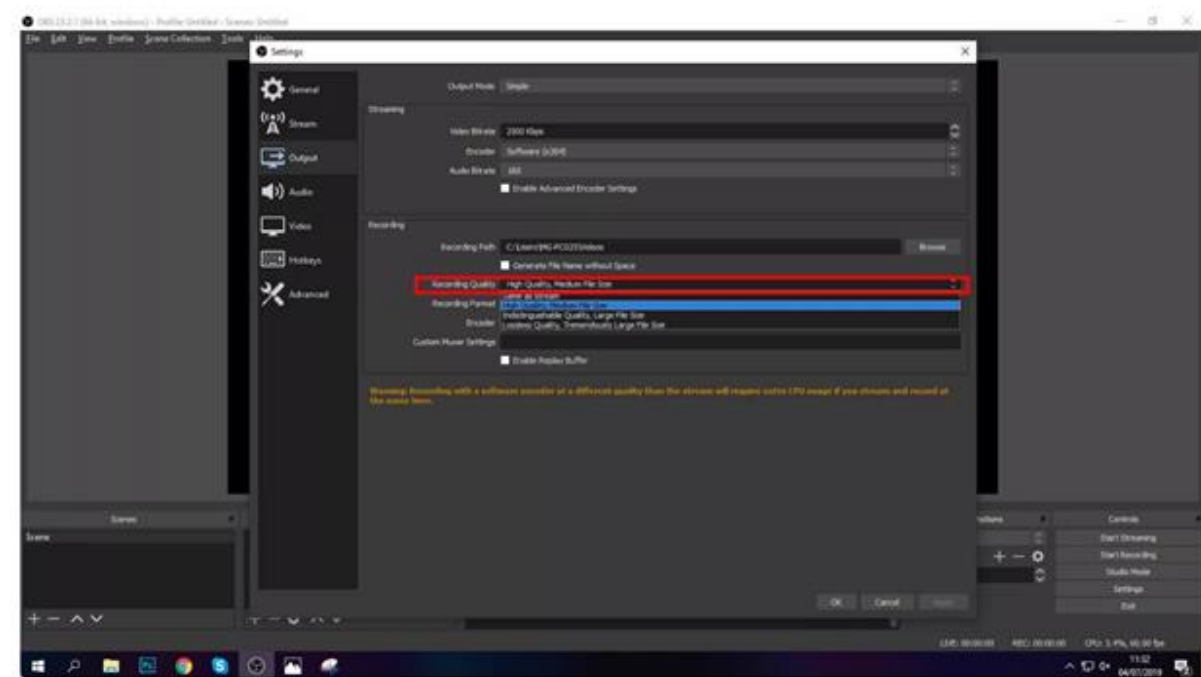


- Pada tahap ini, kamu dapat mengikuti rekomendasi setelan dari modul ini untuk memaksimalkan rekaman game kamu.
- Opsi **Recording Path** berfungsi untuk mengatur pada folder mana kamu akan menyimpan rekaman kamu tersebut.

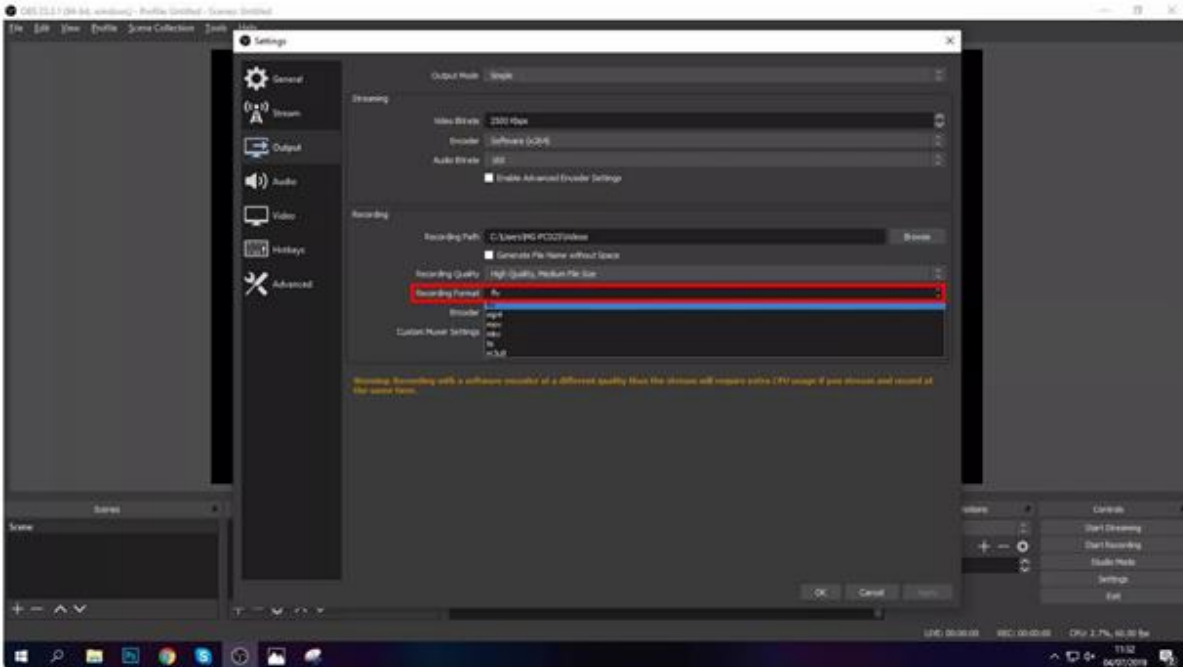
- Perlu dicatat bahwa rekaman akan memiliki ukuran yang besar. Saran saya, simpan rekaman kamu di **harddisk yang memiliki kapasitas besar** supaya nggak cepat penuh.



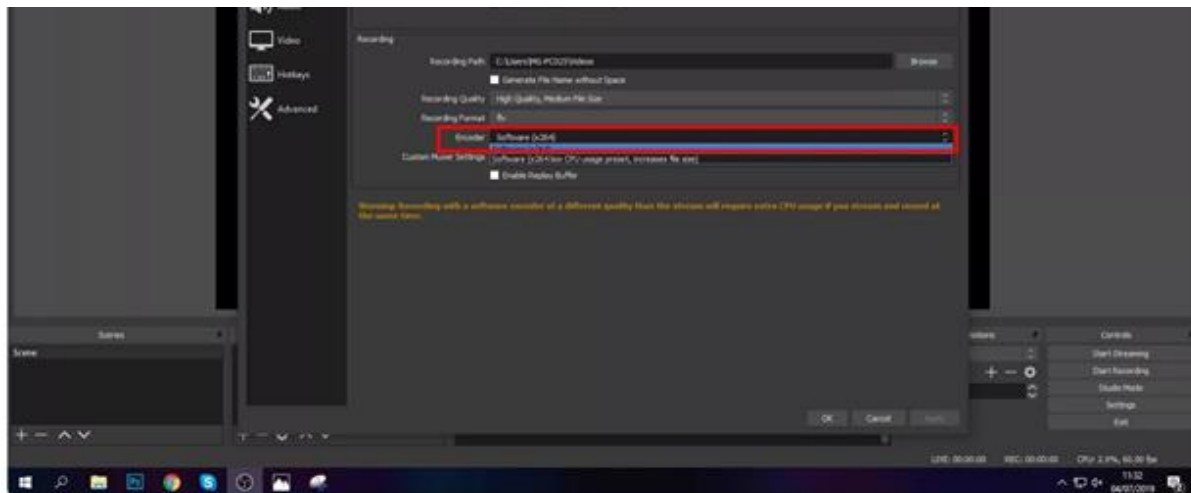
- Atur kualitas rekamanmu di opsi **Recording Quality**. Terdapat empat opsi yang dapat kamu pilih. Semakin tinggi kualitas, ukuran file juga akan semakin besar.
- Jika menyarankan kamu untuk memilih opsi **High quality, medium file size** karena kualitas gambar tetap tinggi meskipun ukuran file tidak terlalu besar.



Pilih format atau ekstensi file rekaman yang kamu inginkan pada opsi **Recording Format**. Di sini kamu bisa menggunakan format MP4 yang punya kualitas baik dan ukuran terbilang kecil.

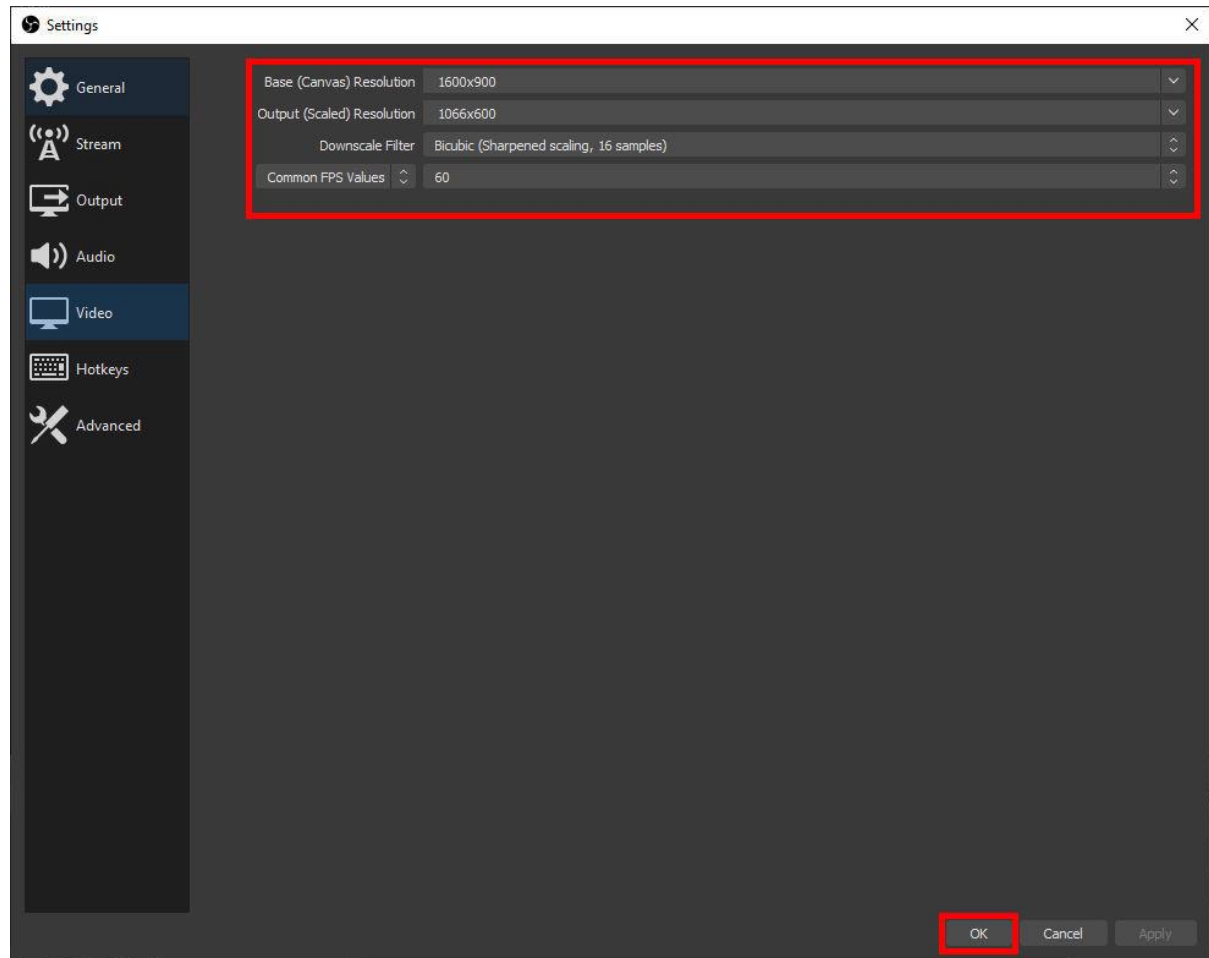


- Terakhir, Pilih opsi **NVENC** (kalau ada) pada kolom **Encoder** untuk memilih *hardware* yang akan kamu fokuskan untuk merekam.
- Di sini, **NVENC** akan memaksimalkan kualitas rekaman serta meringankan beban CPU kamu karena OBS Studio akan membebankan proses *encoding* ke kartu grafis kamu.



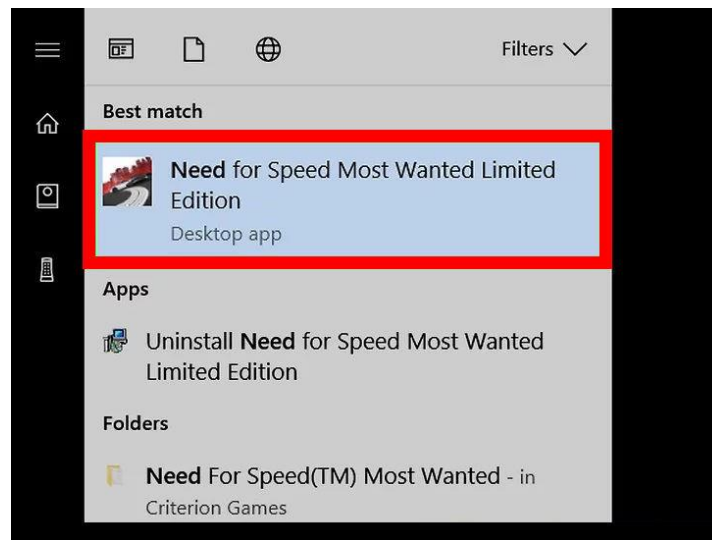
- Setelah selesai mengatur **Output**, pilihlah **tab Video** untuk mengatur resolusi rekaman.
- **Base (Canvas) resolution** adalah resolusi asli monitor kamu. Sedangkan, **Output (Scaled) Resolution** adalah resolusi rekaman kamu nanti.
- Gunakan resolusi Canvas dan Scaled di 1920x1080 untuk mendapatkan **resolusi 1080p** atau 1280x720p untuk **resolusi 720p**. Kamu juga bisa memilih resolusi lain yang kamu inginkan

- Usahakan **Output Resolution** tidak melebihi ukuran **Base Resolution** karena hasil rekaman kamu akan menjadi pecah.
- Kalau sudah selesai mengatur *settingan*, kamu bisa klik **OK** untuk memulai rekaman.

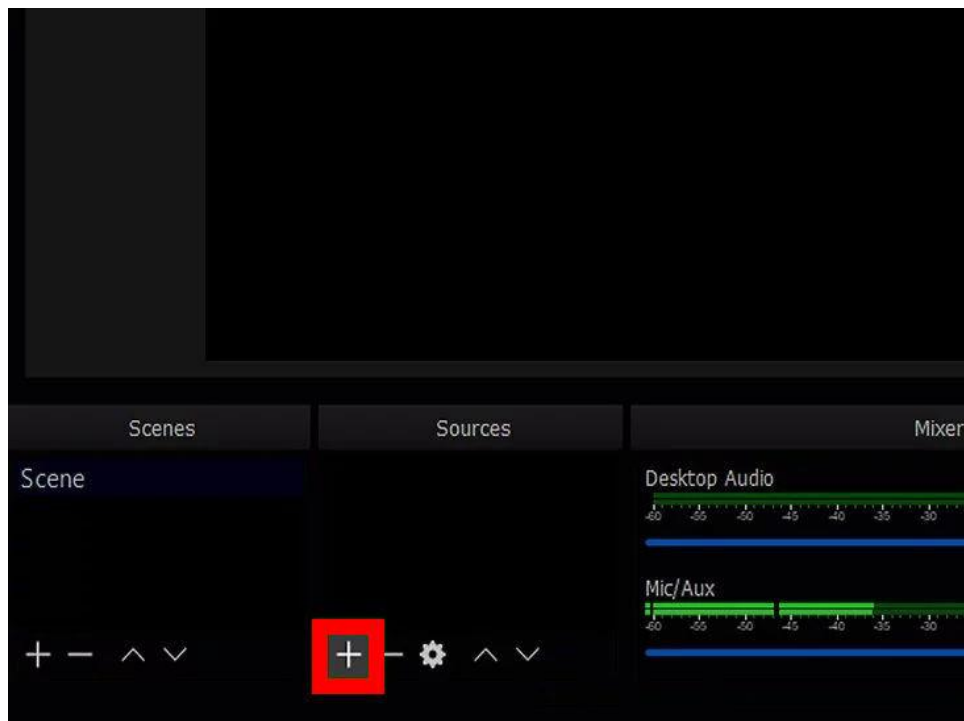


Langkah 3 - Merekam Game dengan OBS Studio

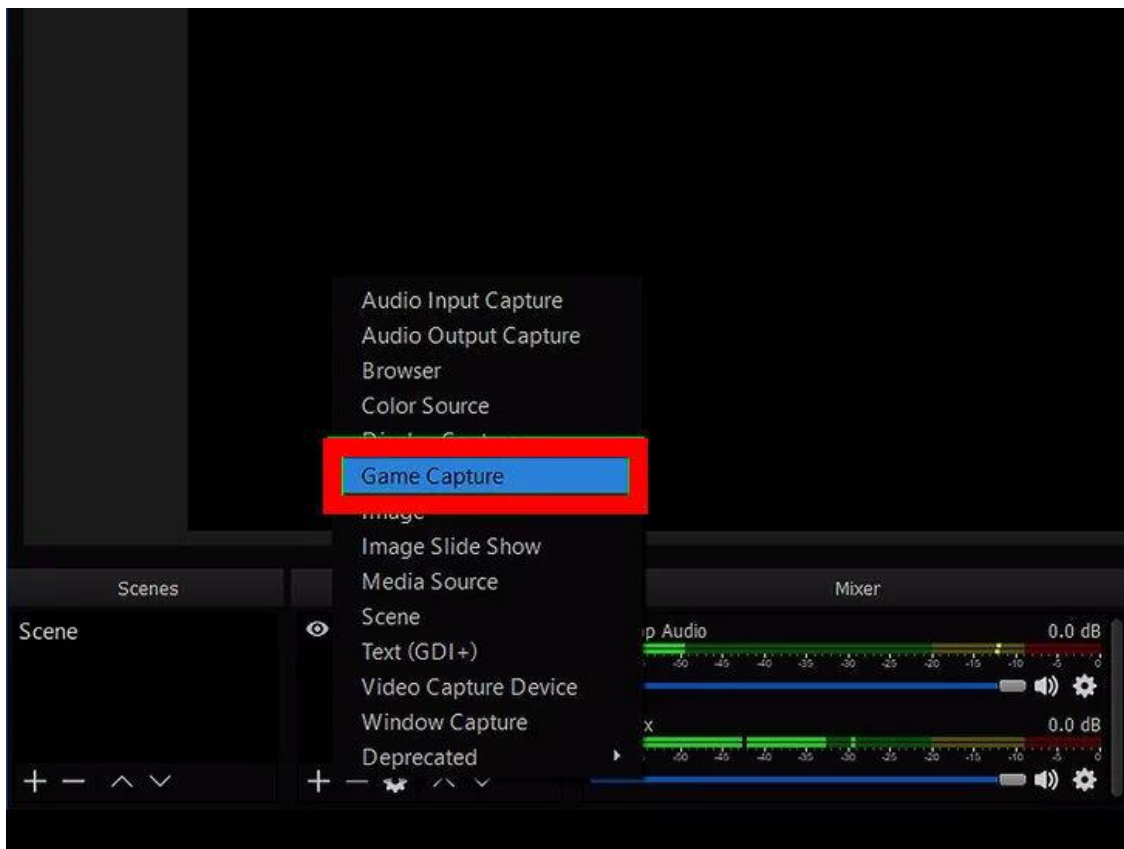
- Buka game yang ingin kamu rekam terlebih dahulu, sebagai contoh Jaka bakal bermain game balapan PC terbaik, **Need for Speed Most Wanted**.



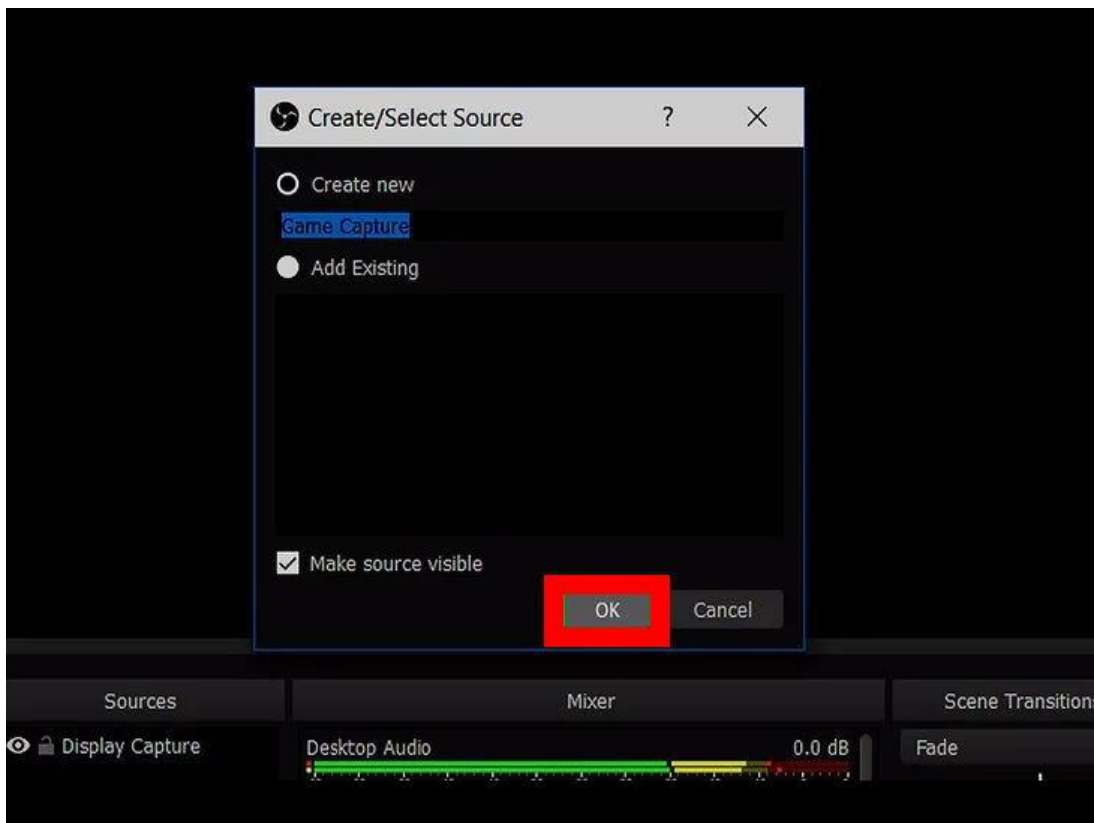
- Buka kembali program OBS. Pada menu **Sources**, lalu klik tombol + untuk membuka opsi *source* yang ada.



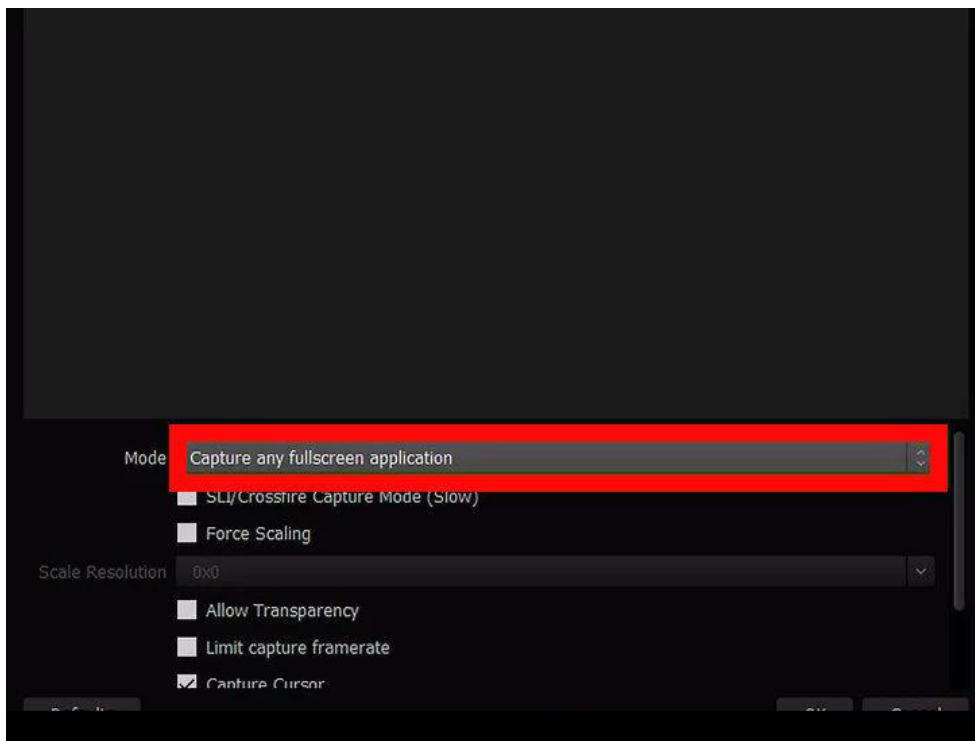
- Pilih opsi **Game Capture** untuk pilihan mudah dalam merekam game yang sebelumnya telah kamu jalankan dan *minimize*.



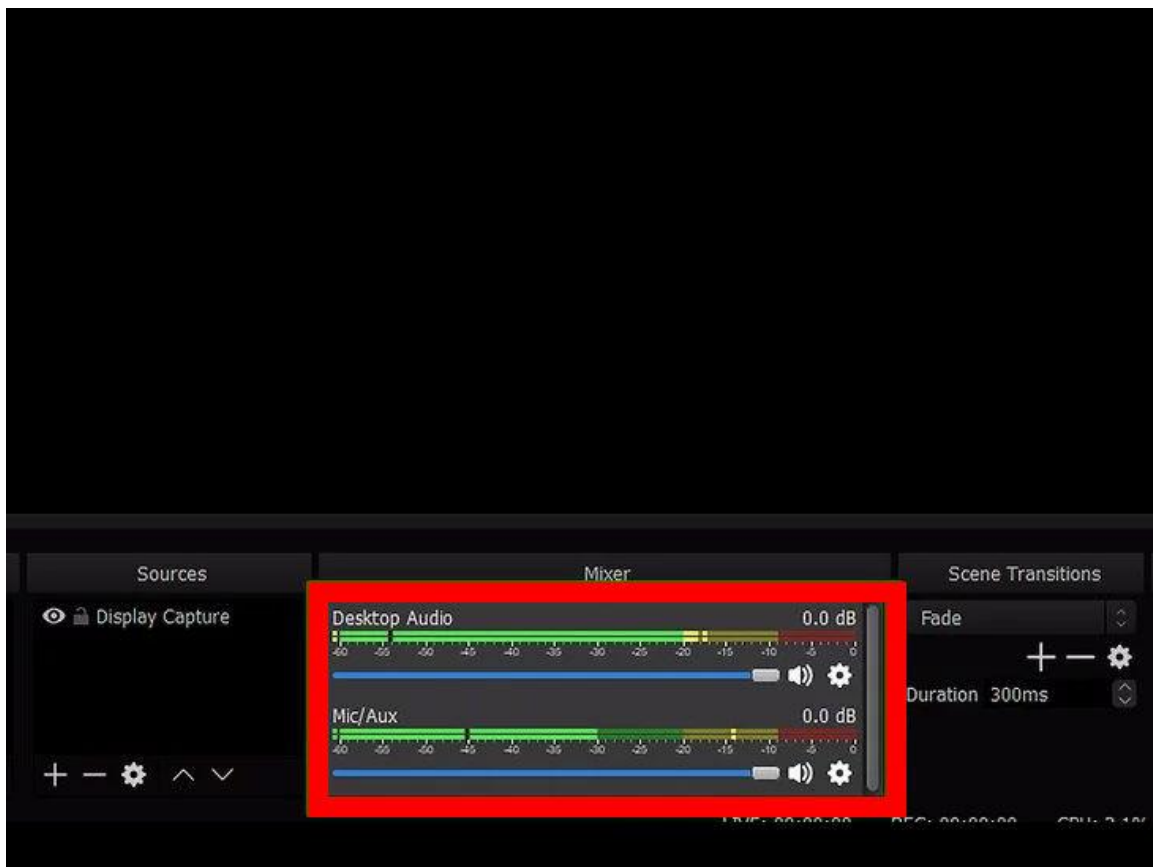
- Kamu bisa memberi nama source tersebut sesuai keinginan kamu. Kalau sudah selesai, klik tombol OK.



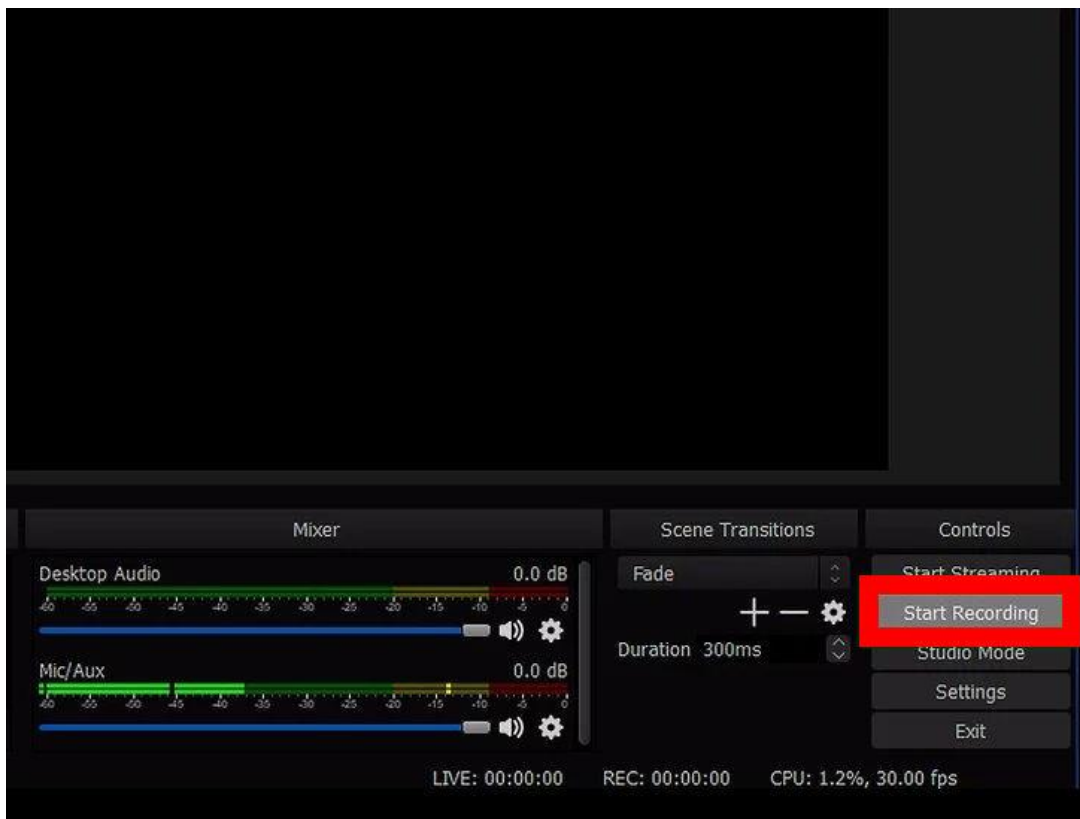
- Pada halaman selanjutnya, kamu bisa memilih mode **Capture any full screen application** untuk otomatis merekam semua aplikasi yang berjalan dengan resolusi **Fullscreen**.
- Kalau kamu hanya ingin merekam game kamu secara spesifik, pilih **Capture Specific Window**, lalu cari judul game yang kamu jalankan.



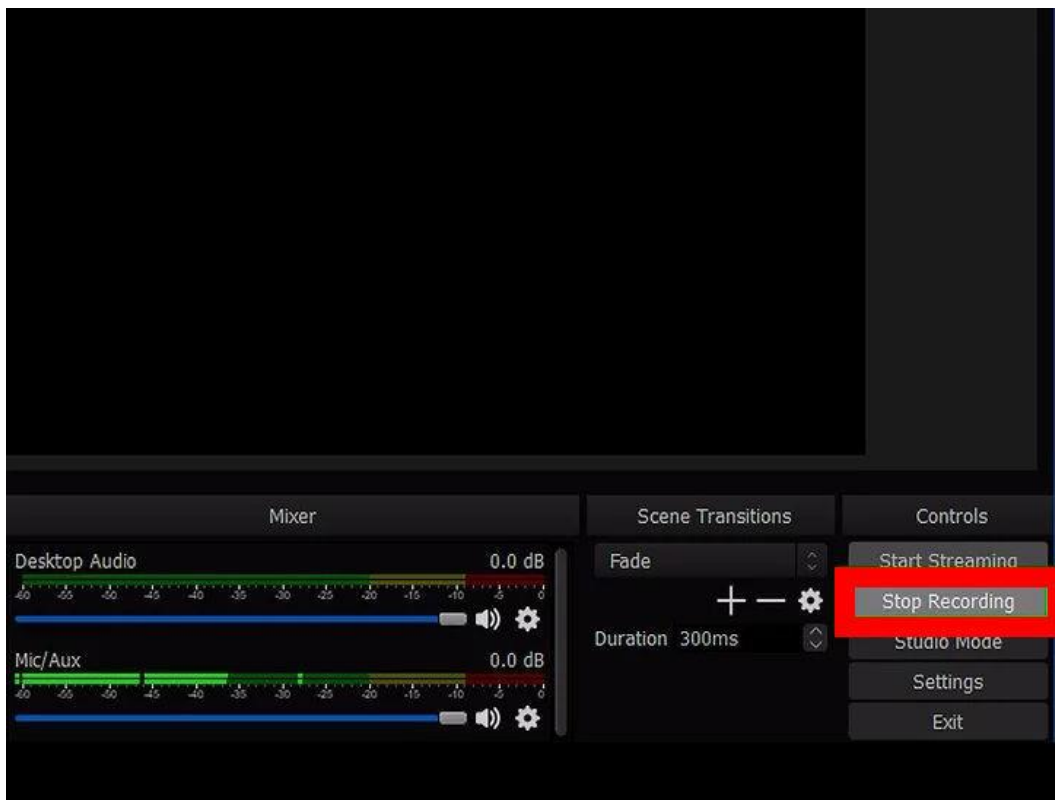
- Setelah selesai mengatur mode *capture*, klik **OK** untuk masuk ke setelan selanjutnya.
- Pilih opsi **source Desktop Audio** untuk menangkap suara yang berasal dari game atau aplikasi yang kamu buka dan opsi **source Mic/Aux** untuk suara yang keluar dari Mic kamu.



- Kalau punya *webcam* untuk merekam *facecam*, kamu bisa memilih opsi **source Video Capture Device** untuk mengaktifkan opsi ini. Kamu bebas *drag* posisi dan ubah ukuran gambar *webcam* yang muncul di OBS.
- Nah, Kamu sudah siap untuk merekam game kamu, geng. Untuk memulai, klik tombol **Start Recording**.



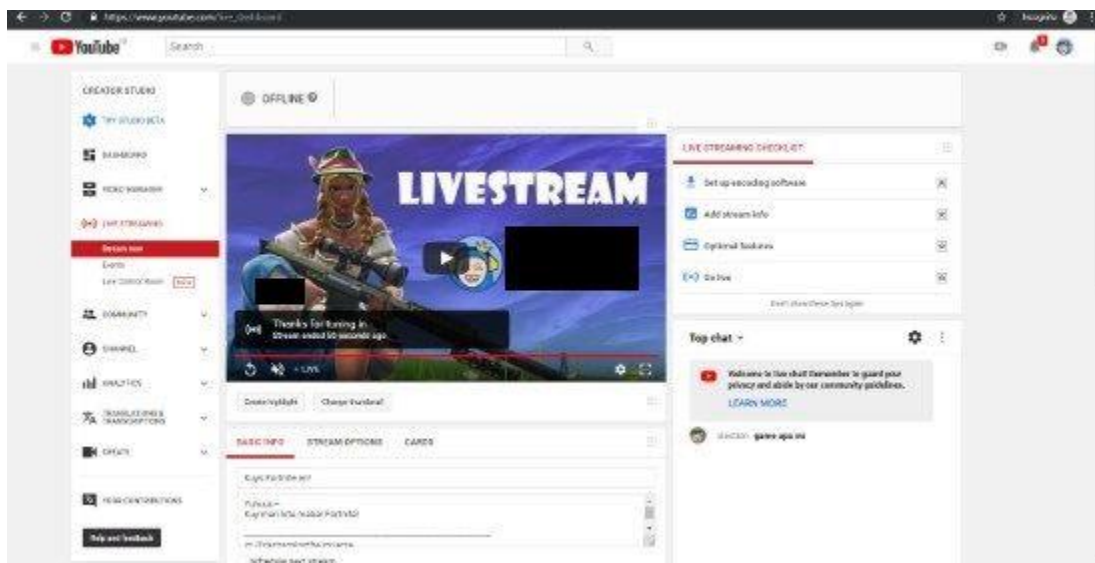
- Kalau sudah selesai merekam, buka kembali OBS kamu, lalu pilih **Stop Recording**.
- Langkah terakhir, kamu bisa menemukan rekaman video kamu pada folder yang telah kamu atur sebelumnya di **Recording Path**



2. Membuat Live Streaming Gaming di Youtube

Langkah 1 - Klik Live Streaming di Channel YouTube Kamu

Kunjungi YouTube di browser PC kamu dan login ke akun YouTube kamu. Lalu masuk ke halaman **Live Dashboard**, seperti gambar di bawah ini.



- **Copy** atau salin **Stream Key**. Untuk menampilkan Stream Key, kamu klik **Reveal**. Lalu blok kode tersebut dan **klik kanan** pada mouse dan **copy**.

ENCODER SETUP

Server URL

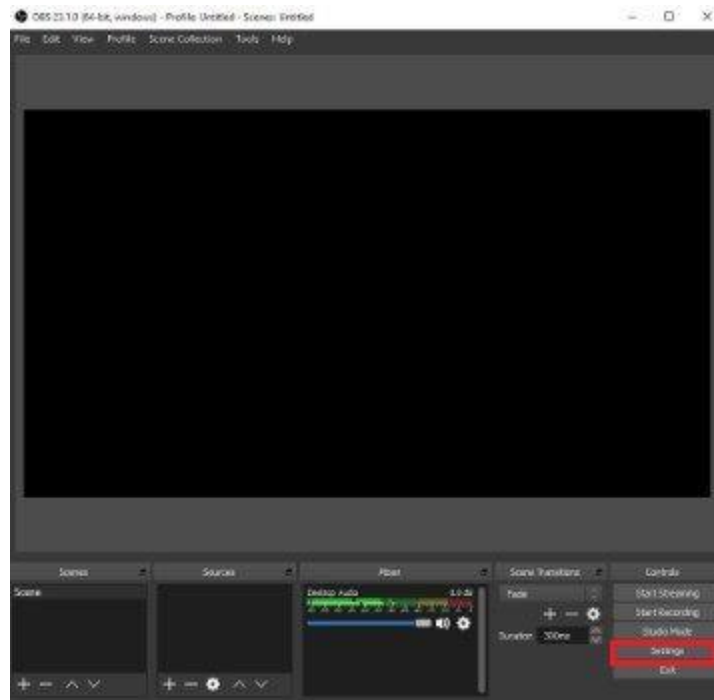
rtmp://a.rtmp.youtube.com/live2

Stream name/key

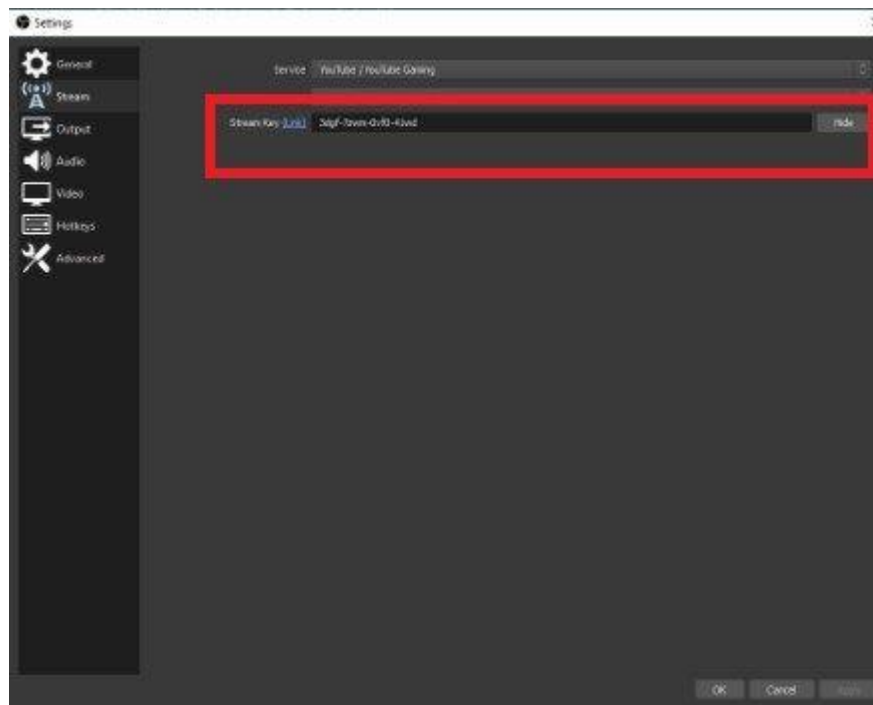
Reveal

Langkah 2 - Masukkan *Stream Key* ke OBS

- Buka OBS dan klik **Settings** yang berada di kanan bawah.

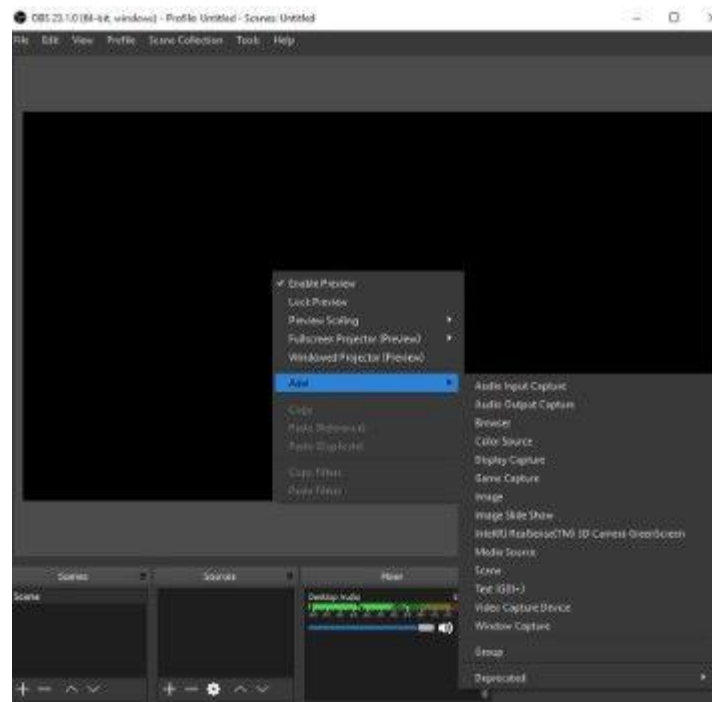


- Klik **Stream** lalu masukan **Stream Key** dan Klik **OK**.
- Jangan lupa atur kolom Service menjadi **Youtube/Youtube Gaming**

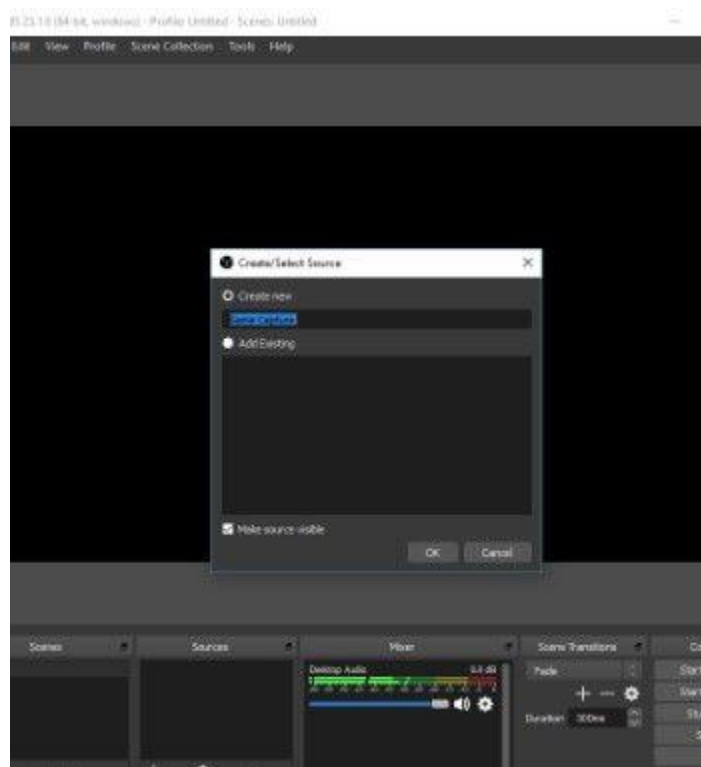


Langkah 3 - Memasukkan Game dan Webcam yang Akan Ditampilkan

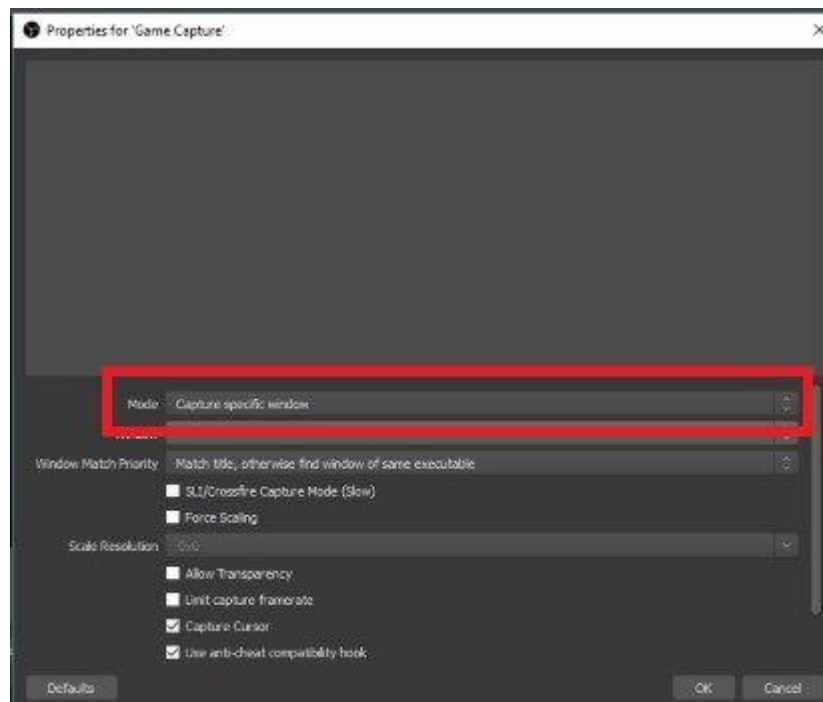
- Kini kamu harus memasukkan game yang akan ditampilkan. **Klik kanan pada mouse** di layar hitam. Klik **Add**, lalu **Game Capture**.
- Kamu juga bisa melakukan step di atas dengan klik simbol **+** pada kolom **Sources** di sebelah bawah OBS



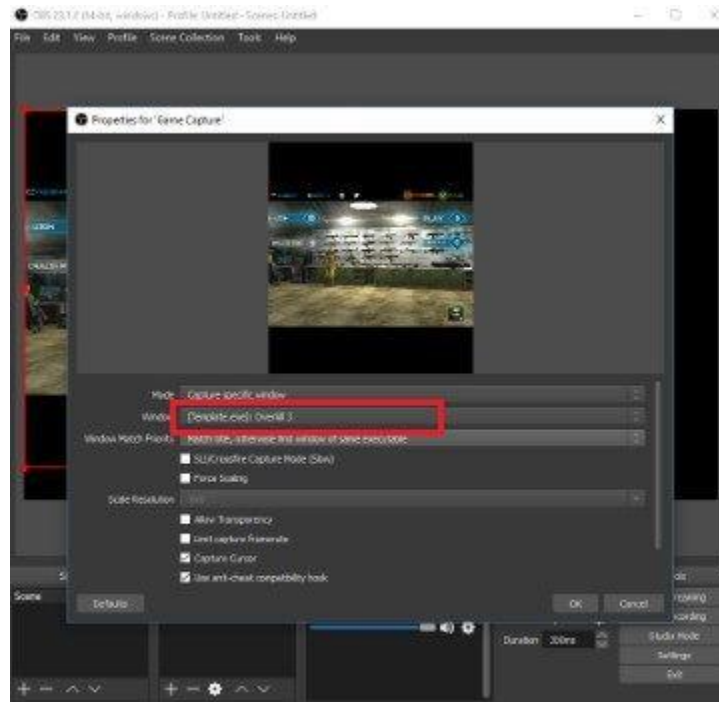
- Kamu klik **Create New** dan klik lagi **OK**.



- Di kolom **Mode**, pilih **Capture Specific Window**.



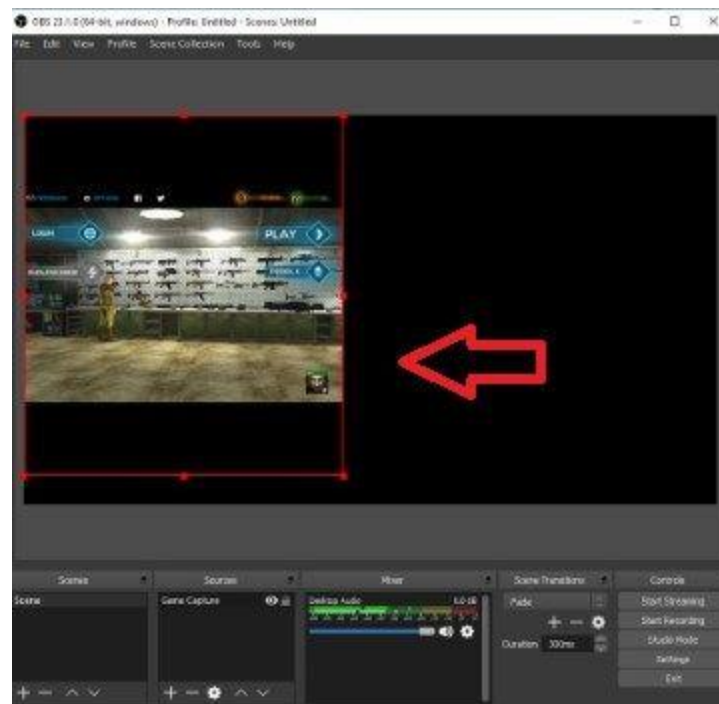
- Pada kolom **Window**, pilih game yang ingin kamu tampilkan di *live streaming*. Jika gamenya sudah muncul seperti di bawah ini, kamu klik **ok**.



- Kalau kamu menggunakan Webcam, klik **Video Capture Device** dan pilih alat Webcam yang kamu gunakan pada kolom **Device**

Langkah 4 - Persiapan Live Streaming

- Kamu bisa mengatur tampilan game dengan cara menarik garis merah, sesuaikan dengan yang kamu inginkan.



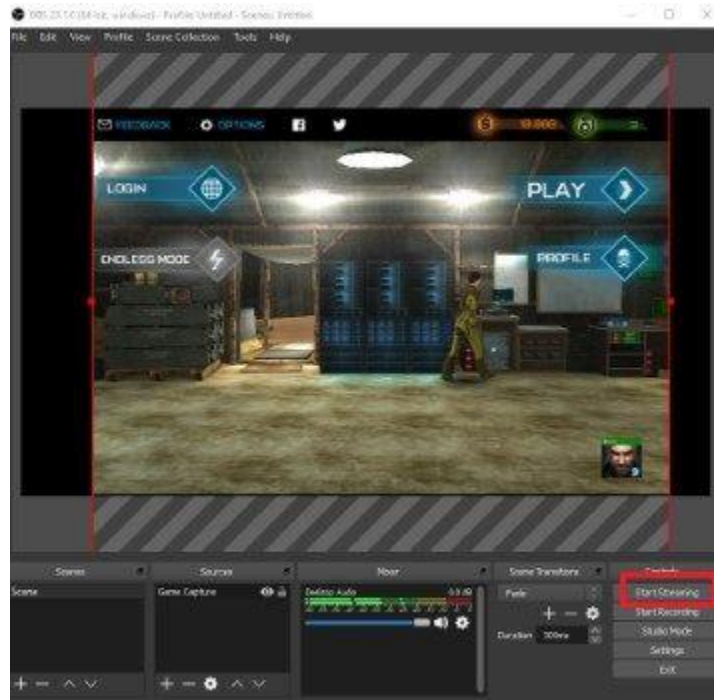
Jangan lupa! Setel atau setting resolusi video stream kamu di menu Video Settings di OBS. Sesuaikan dengan kecepatan internet yang kamu punya. Banyak YouTuber mengatur resolusi pada 1080p

- Masukkan informasi Live Streaming di YouTube pada kolom **Basic Info**. Kamu bisa isi info game apa yang sedang kamu mainkan.

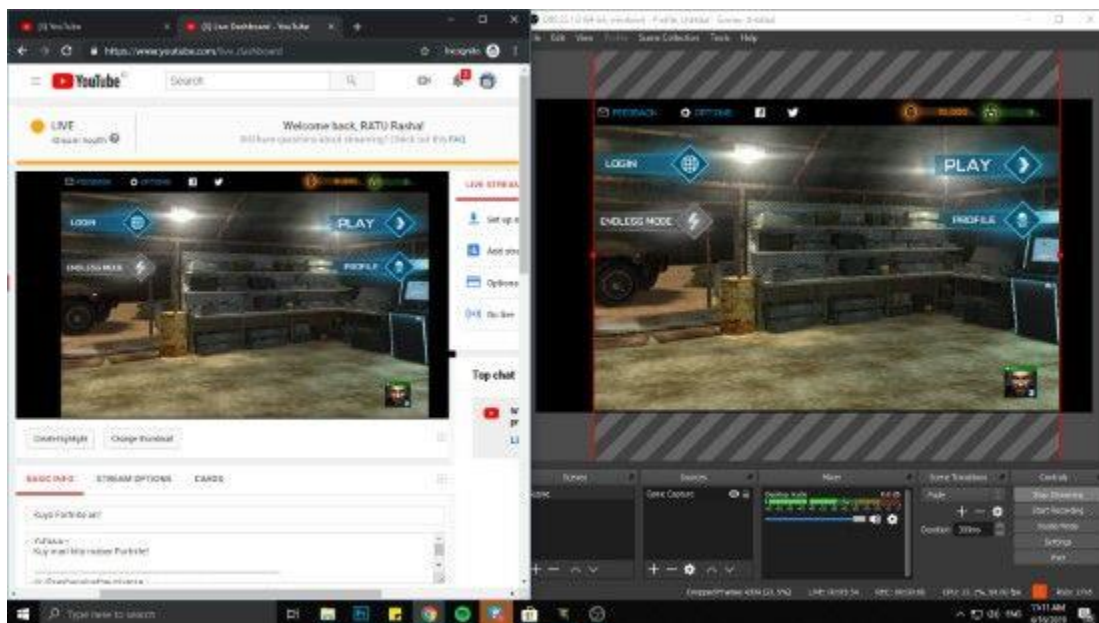
The screenshot displays the YouTube Live Dashboard in an Incognito browser window. At the top, there's a 'LIVE' status and 'Stream health' indicator. A video player shows a game scene with 'ENDLESS MODE' and 'PROFILE' labels. Below the player are 'Create highlight' and 'Change thumbnail' buttons. The 'BASIC INFO' tab is active, showing a title field with 'Kuys Main Overkill 3', a description field with 'Yuhuuu~ Kuy mari kita mabar Overkill 3!', a 'Schedule next stream' checkbox, a 'Category' dropdown set to 'Gaming', a 'Game title (recommended)' field, and a 'Privacy' dropdown set to 'Public'. On the right, there is a 'Top chat' section with several chat messages and a 'Go live' button.

Langkah 5 - Mulai Live Streaming

- Klik **Start Streaming** di OBS untuk memulai live streaming. Selamat streaming, geng!



Perlu diperhatikan, koneksi internet yang stabil dan kencang sangat diperlukan.



D. Tugas

1. Buatlah video gaming dan live streaming gaming di Youtube!
2. Buatlah laporan resmi dari pembuatan video dan live streaming tersebut!

PERTEMUAN 4 & 5

Kompresi Data Multimedia

A. Sub Capaian Pembelajaran Mata Kuliah :

- 1) Mahasiswa mengenal dan mampu melakukan kompresi data multimedia dengan menggunakan berbagai metode

Penilaian Mata Kuliah : UTS (30%), UAS (30%), Tugas (30%), Keaktifan (10%)

B. Dasar Teori

Kebutuhan akan kapasitas penyimpanan yang besar nampaknya makin penting. Kebutuhan ini, disebabkan oleh data yang harus disimpan makin lama semakin bertambah banyak, khususnya bagi dunia usaha dan perbankan. Perusahaan tersebut umumnya sangat membutuhkan kapasitas yang sangat besar, untuk menyimpan semua data dan file-file penting. Penyimpanan tersebut bukan hanya dialokasikan pada satu tempat saja. Tapi mereka juga akan menyimpan data atau file-file tersebut pada tempat yang lain. Meskipun yang disimpan tersebut sama, hal ini berguna untuk backup data. Backup perlu dilakukan karena tidak ada yang menjamin suatu data pada tempat penyimpanan didalam komputer tidak akan mengalami kerusakan. Alangkah banyaknya kapasitas penyimpanan yang harus disediakan untuk menampung semua hal tersebut.

Untunglah data-data tersebut dapat dimampatkan (compress) terlebih dahulu sehingga tempat yang dibutuhkan dimemori semakin sedikit dan waktu yang dibutuhkan untuk berkomunikasi lebih pendek, sehingga kegagalan dalam manipulasi data lebih sedikit. Metoda pemampatan yang digunakan pada saat ini adalah dengan cara memampatkan file-file yang sudah jadi (file asal) lalu dimampatkan dan baru di komunikasikan. Setelah sampai data hasil pemampatan tersebut pada si penerima, lalu dilakukan penirmampatan (de-compress) untuk mengembalikan ke bentuk asal, baru file tersebut dapat digunakan.

A. Pengertian Kompresing

Dalam ilmu komputer dan teori informasi, kompresi data atau sumber pengkodean adalah proses encoding informasi dengan menggunakan lebih sedikit bit (atau unit informasi-bantalan lainnya) dari sebuah unencoded representasi akan menggunakan, melalui penggunaan khusus pengkodean skema.

Dalam komputasi, deduplication data adalah teknik kompresi data khusus untuk menghilangkan data-grained berlebihan kasar, biasanya untuk meningkatkan utilisasi storage.

Seperti komunikasi apapun, dikompresi komunikasi data hanya bekerja jika kedua pengirim dan penerima informasi memahami skema pengkodean. Misalnya, teks ini masuk akal hanya jika penerima mengerti bahwa itu adalah dimaksudkan untuk ditafsirkan sebagai karakter yang mewakili bahasa Inggris. Demikian pula, data terkompresi hanya dapat dipahami jika metode decoding diketahui oleh penerima.

Kompresi berguna karena membantu mengurangi konsumsi sumber daya mahal, seperti hard disk space atau transmisi bandwidth. Pada sisi negatifnya, data dikompresi harus didekompresi untuk digunakan, dan ini pengolahan tambahan mungkin merugikan beberapa aplikasi. Sebagai contoh, skema kompresi untuk video mungkin memerlukan perangkat keras mahal untuk video yang akan didekompresi cukup cepat untuk dilihat karena sedang decompressed (pilihan untuk dekompresi video secara penuh sebelum menonton mungkin nyaman, dan membutuhkan ruang penyimpanan untuk decompressed video). Rancangan skema kompresi data sehingga melibatkan trade-off antara berbagai faktor, termasuk tingkat kompresi, jumlah distorsi memperkenalkan (jika menggunakan skema kompresi lossy), dan sumber daya komputasi yang dibutuhkan untuk kompres dan uncompress data.

Ada 2 kompresi data yaitu :

a. Lossy

Lossy kompresi citra digunakan dalam kamera digital, untuk meningkatkan kapasitas penyimpanan dengan minimal penurunan kualitas gambar. Demikian pula, DVD menggunakan lossy MPEG-2 Video codec untuk kompresi video.

Dalam lossy kompresi audio, metode psychoacoustics digunakan untuk menghapus non-terdengar (atau kurang terdengar) komponen dari sinyal. Kompresi berbicara manusia sering dilakukan dengan teknik khusus bahkan lebih, sehingga "pidato kompresi "atau" suara coding" kadang-kadang dibedakan sebagai suatu disiplin yang terpisah dari "kompresi audio". audio yang berbeda dan kompresi standar pidato terdaftar di bawah codec audio. Suara kompresi akan digunakan dalam telepon Internet misalnya, sementara kompresi audio yang digunakan untuk CD ripping dan diterjemahkan oleh pemain audio.

Berikut ciri-cirinya

- Terdapat informasi yang hilang pada saat sampai pada telinga dan mata manusia.
- Digunakan pada kompresi objek audio, image, video dimana keakuratan data absolut tidak diperlukan.
- Contoh: bila video image dikompres dengan basis frame-by- frame hilangnya data pada satu frame tidak mempengaruhi penglihatan.
- Aplikasi: medical screening systems, video conferencing, dan multimedia messaging systems.
- Metode kompresi yang banyak digunakan adalah standar JPEG.

b. Lossless

Berikut ini cirri-ciri:

- Data tidak berubah atau hilang pada proses kompresi atau dekompresi
- Membuat satu replika dari objek asli

- Menghilangkan perulangan karakter
- Digunakan pada data teks dan image
- Pada saat dilakukan dekompres, perulangan karakter diinstal kembali

Standart compressing lossless yaitu:

1. Packbits encoding (Run-length encoding)

- Kompresi data paling sederhana dan digunakan pada awal penggunaan kompresi.
- Digunakan untuk kompresi image hitam-putih (binary).
- String karakter yang berulang menempati dua byte:
- Byte pertama berisi jumlah dari banyaknya perulangan
- Byte kedua berisi karakter itu sendiri
- Dilakukan pada satu baris (atau scanline), dan tidak digunakan pada baris yang mempunyai jumlah scanline banyak.
- Byte lebih besar dari pada byte image asli. Efek ini disebut reverse compression atau negative compression.

2. CCITT (International Telegraph and Telephone Consultative Committee) Group 3 1D

- Berdasarkan run-length encoding, scanline dilakukan pada pixel dari warna yang sama (hitam atau putih).
- Hanya untuk image hitam-putih, bukan grayscale atau warna.
- Aplikasi utama digunakan pada faksimil dan pada awal document imaging.
- Menggunakan Huffman encoding untuk encoding pixel runlength pada CCIT Group 3 dan Group 4.

Keuntungan:

- Sederhana pada implementasi
- Menjadi standar faksimil dan aplikasi document imaging

Kerugian:

- Satu dimensi dengan code setiap baris atau garis terpisah.
- Tanpa mekanisme untuk melindungi dari kesalahan.

3. CCITT Group 3 2D

4. CCITT Group 4

- Dua dimensi tanpa faktor K, yaitu garis seluruhnya.
- Garis referensi pertama adalah semua garis putih pada image bagian atas.
- Group pertama dari pixel (scanline) dikode yang menganggap garis putih sebagai garis referensi dari garis berikutnya.
- Mendapatkan level kompresi yang tinggi.

5. Lempel-Ziv and Welch algorithm LZW

Setelah proses kompresi, maka data yang dihasilkan akan berukuran 35 byte. Dengan proses kompresi tersebut, kita telah menghemat tempat penyimpanan sebesar 31 byte (47%) !!.

Huffman Coding

Kompresi dengan algoritma Huffman Coding dilakukan dengan cara :

1. Hitung frekuensi kemunculan setiap simbol.
2. Pilih dua buah simbol dengan frekuensi terkecil, lalu gabungkan dalam satu tangkai.
3. Ulangi langkah kedua hingga tidak ada lagi tangkai yang dapat digabungkan.

Misalnya, terdapat sebuah pesan : "ABABAAAADDDCCCFBB". Pesan tersebut berukuran 17 byte (termasuk spasi).

Pertama, kita akan menghitung kemunculan setiap karakter :

Simbol	Kemunculan
A	6
B	4
C	3
D	3
F	1

Pilih dua buah simbol dengan frekuensi terkecil, yaitu simbol F dan D, lalu gabungkan.

Simbol	Kemunculan
A	6
B	4
C	3
(D,F)	4

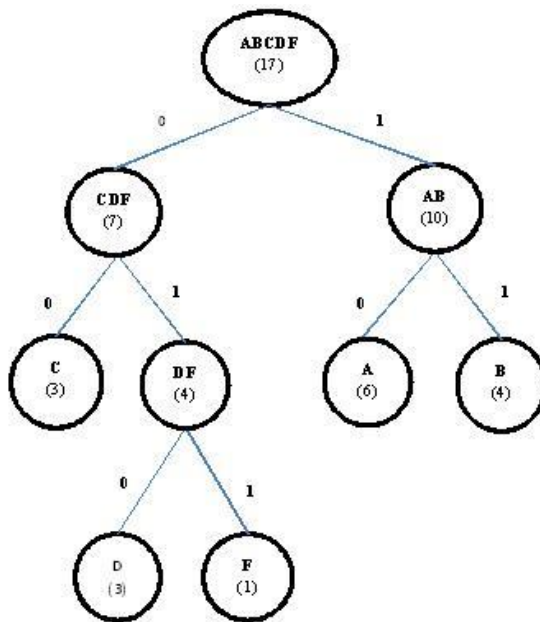
Pilih kembali dua buah simbol dengan frekuensi terkecil, lalu gabungkan. Ulangi hal ini hingga tidak dapat lagi digabungkan.

Simbol	Kemunculan
A	6
B	4
(C,(D,F))	7

Simbol	Kemunculan
(A,B)	10
(C,(D,F))	7

Simbol	Kemunculan
((A,B), (C,(D,F)))	17

Pembentukan pohon Huffman :



Dari pohon diatas, maka huruf 'D' dapat kita kodekan dengan : 000. Berikut ini merupakan tabel lengkap hasil pengkodean seluruh simbol :

Simbol	Kode
A	10
B	11
C	00
D	010
F	011

Berdasarkan tabel diatas, maka "ABABAAAADDDCCCFBB" dapat kita kodekan menjadi seperti berikut : 10111011101010100100100100000001101111. Data hasil kompresi berukuran **29 bit / 4 byte**. Dengan demikian, kita telah menghemat sebanyak **13 byte (76%) !!!**.

Algoritma Lempel Ziv Welch (LZW)

Algoritma Lempel Ziv Welch (LZW) merupakan salah satu algoritma string compression yang diciptakan oleh tiga sekawan bernama Lempel, Ziv dan Welch pada tahun 1977.

LZW termasuk algoritma dengan metode **lossless compression**, artinya mengurangi bit dengan mengidentifikasi dan menghilangkan redundansi statistik. Tidak ada informasi yang hilang dalam kompresi lossless.

Algoritma **LZW** adalah teknik kompresi yang sangat umum. Algoritma ini biasanya digunakan dalam GIF dan opsional dalam PDF dan TIFF

Cara kerjanya: (*geekforgeeks*)

1. Kompresi LZW menggunakan tabel kode, dengan 4096 sebagai pilihan umum untuk jumlah entri tabel. Kode 0-255 dalam tabel kode selalu ditugaskan untuk mewakili byte tunggal dari file input.
2. Ketika encoding dimulai, tabel kode hanya berisi 256 entri pertama, dengan sisa tabel kosong. Kompresi dicapai dengan menggunakan kode 256 hingga 4095 untuk mewakili urutan byte.
3. Ketika Encoding berlanjut, LZW mengidentifikasi urutan berulang dalam data, dan menambahkannya ke tabel kode.

4. Proses Decoding dicapai dengan mengambil setiap kode dari file terkompresi dan menerjemahkannya melalui tabel kode untuk menemukan karakter atau karakter yang diwakilinya.
5. Table ASCII, ini digunakan untuk mengetahui biner code dari string.

ASCII - Binary Character Table

Letter	ASCII Code	Binary	Letter	ASCII Code	Binary
a	097	01100001	A	065	01000001
b	098	01100010	B	066	01000010
c	099	01100011	C	067	01000011
d	100	01100100	D	068	01000100
e	101	01100101	E	069	01000101
f	102	01100110	F	070	01000110
g	103	01100111	G	071	01000111
h	104	01101000	H	072	01001000
i	105	01101001	I	073	01001001
j	106	01101010	J	074	01001010
k	107	01101011	K	075	01001011
l	108	01101100	L	076	01001100
m	109	01101101	M	077	01001101
n	110	01101110	N	078	01001110
o	111	01101111	O	079	01001111
p	112	01110000	P	080	01010000
q	113	01110001	Q	081	01010001
r	114	01110010	R	082	01010010
s	115	01110011	S	083	01010011
t	116	01110100	T	084	01010100
u	117	01110101	U	085	01010101
v	118	01110110	V	086	01010110
w	119	01110111	W	087	01010111
x	120	01111000	X	088	01011000
y	121	01111001	Y	089	01011001
z	122	01111010	Z	090	01011010

ASCII

Contoh proses **encoding** kompresi:

Encoding

Example:

Compress String Text: **thisisthe**

On ASCII:

116 104 105 115 105 115 116 104 101

01110100 01101000 01101001 01110011 01101001 01110011 01110100 01101000 01100101 = 72 bits

Nah ASCII: 72 bits, kita bisa mendapatkan bits yang lebih sedikit (baik) dengan melakukan kompresi

Current	Next	Output	Add to dictionary
t116	h104	t116	th256
h104	i105	h104	hi257
i105	s115	i105	is258
s115	i105	s115	si259
i105	s115	is sudah ada di dictionary	
is258	t116	is258	ist260
t116	h104	is sudah ada di dictionary	
th256	e101	th256	the261
e101	-	e101	-

Kode ASCII. Biasanya, setiap karakter disimpan dengan 8 bit biner, memungkinkan hingga 256 simbol unik untuk data

116 104 105 115 258 256 101

Save: 72 - 63 = 9bits

01110100 01101000 01101001 01110011 10000010 10000000 001100101 = 63 bits

87.5%

hendroprasetyo.com

encode compressi

Contoh proses **decoding** kompresi:

decompress hasil encode?

Decoding

116 104 105 115 258 256 101

Current	Next	Output	Add to dictionary
116	104	116	116 104 (256)
104	105	104	104 105 (257)
105	115	105	105 115 (258)
115	258	115	115 105 115 (259)
258	256	105 115	105 115 116 (260)
256	101	116 104	116 104 101 (261)
101	-	101	-

116 104 105 115 105 115 116 104 101

=

thisisthe

Kenapa kita tidak menulis 258 dan 256 ke dalam dictionary?

Karena kita mendapatkan angka tersebut dari kombinasi (105 115 116 104)

Itu bisa saja kita lakukan jika dalam tracing contoh perhitungan, tapi akan menjadi masalah ketika itu dibuat dalam code program

hendroprasetyo.com

decode

compressi

Lossless versus kompresi lossy

Lossless algoritma kompresi memanfaatkan redundansi biasanya statistik sedemikian rupa untuk mewakili pengirim data lebih singkat tanpa kesalahan. Kompresi Lossless dimungkinkan karena sebagian besar dunia nyata telah redundansi data statistik. Sebagai contoh, dalam teks bahasa Inggris, 'e' huruf jauh lebih umum daripada huruf 'z', dan probabilitas bahwa 'q' huruf akan diikuti oleh huruf 'z' sangat kecil. Kompresi jenis lain, disebut *kompresi lossy data* atau persepsi coding, adalah mungkin jika beberapa kehilangan kesetiaan diterima. Umumnya, sebuah kompresi data lossy akan dipandu oleh penelitian tentang bagaimana orang melihat data tersebut. Sebagai contoh, mata manusia lebih sensitif terhadap variasi halus dalam terang daripada variasi warna. JPEG kompresi gambar yang bekerja di sebagian oleh "pembulatan" beberapa informasi penting ini-kurang. Lossy kompresi data menyediakan cara untuk mendapatkan kesetiaan terbaik untuk jumlah yang diberikan kompresi. Dalam beberapa kasus, *transparan* (unnoticeable) kompresi yang diinginkan, dalam kasus lain, kesetiaan adalah dikorbankan untuk mengurangi jumlah data sebanyak mungkin.

Skema kompresi Lossless adalah reversibel sehingga data asli dapat direkonstruksi, sementara skema lossy menerima beberapa hilangnya data untuk mencapai kompresi yang lebih tinggi. Namun, algoritma kompresi lossless data akan selalu gagal untuk kompres beberapa file, memang, setiap algoritma kompresi tentu akan gagal untuk kompres data tidak berisi pola-pola yang jelas. Upaya untuk kompres data yang telah dikompres biasanya sudah demikian akan menghasilkan sebuah ekspansi, seperti yang akan mencoba untuk menekan semua tapi yang paling sepele dienkripsi data.

C. Percobaan

Pada percobaan ini kita akan mencoba membuat sebuah kode program untuk melakukan kompresi data dengan menggunakan metode RLE dan Huffman Coding.

Source Code RLE dengan menggunakan PHP :

```
<?php
function encode($str)
{
    return preg_replace_callback('/(.)\1*/', function ($match) {
        return strlen($match[0]) . $match[1];
    }, $str);
}

function decode($str)
{
    return preg_replace_callback('/(\d+)(\D)/', function($match) {
        return str_repeat($match[2], $match[1]);
    }, $str);
}

echo encode('WWWWWWWWWWWWBWWWWWWWWWWWWBBBWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWBWWWWWW
WWWWWWWWWW'), PHP_EOL;
echo decode('12W1B12W3B24W1B14W'), PHP_EOL;
?>
```

Output :

```
12W1B12W3B24W1B14W
WWWWWWWWWWWWBWWWWWWWWWWWWBBBWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWBWWWWWWWWWWWWWW
```

Source Code RLE dengan menggunakan Java :

```
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import java.util.*;
// import java.applet.*;
// import java.awt.*;
public class RunLengthEncoding {

    public static String encode(String source) {
        StringBuffer dest = new StringBuffer();
        for (int i = 0; i < source.length(); i++) {
            int runLength = 1;
            while (i+1 < source.length() && source.charAt(i) == source.
charAt(i+1)) {
                runLength++;
                i++;
            }
            dest.append(runLength);
            dest.append(source.charAt(i));
        }
        return dest.toString();
    }

    public static String decode(String source) {
        StringBuffer dest = new StringBuffer();
        Pattern pattern = Pattern.compile("[0-9]+|[a-zA-Z]");
        Matcher matcher = pattern.matcher(source);
        while (matcher.find()) {
            int number = Integer.parseInt(matcher.group());
            matcher.find();
            while (number-- != 0) {
                dest.append(matcher.group());
            }
        }
        return dest.toString();
    }

    public static void main(String[] args) {
        String example = "WWWWWWWWWWWWWWBWWWWWWWWWWWWBBBWWWWWWWWWWWWWWWWWW
WWWWWWWWBWWWWWWWWWWWWWW";
        System.out.println(example);
        System.out.println(encode(example));
        System.out.println(decode("1W1B1W1B1W1B1W1B1W1B1W1B1W1B"));
        // System.out.println("Hello Word");
    }
}
```


Source Code Huffman Coding dengan menggunakan PHP :

```
<?php
function encode($symb2freq) {
    $heap = new SplPriorityQueue;
    $heap->setExtractFlags(SplPriorityQueue::EXTR_BOTH);
    foreach ($symb2freq as $sym => $wt)
        $heap->insert(array($sym => ''), -$wt);

    while ($heap->count() > 1) {
        $lo = $heap->extract();
        $hi = $heap->extract();
        foreach ($lo['data'] as &$x)
            $x = '0'.$x;
        foreach ($hi['data'] as &$x)
            $x = '1'.$x;
        $heap->insert($lo['data'] + $hi['data'],
                    $lo['priority'] + $hi['priority']);
    }
    $result = $heap->extract();
    return $result['data'];
}

$txt = 'this is an example for huffman encoding';
$symb2freq = array_count_values(str_split($txt));
$huff = encode($symb2freq);
echo "Symbol\tWeight\tHuffman Code <br>";
foreach ($huff as $sym => $code)
    echo "$sym&nbsp;&nbsp;&nbsp;$symb2freq[$sym]&nbsp;&nbsp;&nbsp;$code<br>";
?>
```

Output :

Symbol	Weight	Huffman Code
n	4	000
m	2	0010
o	2	0011
t	1	01000
g	1	01001
x	1	01010
u	1	01011
s	2	0110
c	1	01110
d	1	01111
p	1	10000
l	1	10001
a	3	1001
	6	101
f	3	1100
i	3	1101
r	1	11100
h	2	11101
e	3	1111

Source Code Huffman Coding dengan menggunakan Java :

```
<?php
function encode($symb2freq) {
    $heap = new SplPriorityQueue;
    $heap->setExtractFlags(SplPriorityQueue::EXTR_BOTH);
    foreach ($symb2freq as $sym => $wt)
        $heap->insert(array($sym => ''), -$wt);

    while ($heap->count() > 1) {
        $lo = $heap->extract();
        $hi = $heap->extract();
        foreach ($lo['data'] as &$x)
            $x = '0'.$x;
        foreach ($hi['data'] as &$x)
            $x = '1'.$x;
        $heap->insert($lo['data'] + $hi['data'],
                    $lo['priority'] + $hi['priority']);
    }
    $result = $heap->extract();
    return $result['data'];
}

$txt = 'this is an example for huffman encoding';
$symb2freq = array_count_values(str_split($txt));
$huff = encode($symb2freq);
echo "Symbol\tWeight\tHuffman Code <br>";
foreach ($huff as $sym => $code)
    echo "$sym     $symb2freq[$sym]&nbsp;    $code<br>";
?>
```

Output :

SYMBOL	WEIGHT	HUFFMAN CODE
d	1	00000
t	1	00001
h	2	0001
s	2	0010
c	1	00110
x	1	00111
m	2	0100
o	2	0101
n	4	011
u	1	10000
l	1	10001
a	3	1001
r	1	10100
g	1	101010
p	1	101011
e	3	1011
i	3	1100
f	3	1101
	6	111

Source Code LZW dengan menggunakan PHP :

```
<?php
class LZW
{
    function compress($unc) {
        $i;$c;$wc;
        $w = "";
        $dictionary = array();
        $result = array();
        $dictSize = 256;
        for ($i = 0; $i < 256; $i += 1) {
            $dictionary[chr($i)] = $i;
        }
        for ($i = 0; $i < strlen($unc); $i++) {
            $c = $unc[$i];
            $wc = $w.$c;
            if (array_key_exists($w.$c, $dictionary)) {
                $w = $w.$c;
            } else {
                array_push($result,$dictionary[$w]);
                $dictionary[$wc] = $dictSize++;
                $w = (string)$c;}
        }
        if ($w != "") {
            array_push($result,$dictionary[$w]);}
        return implode(",",$result);
    }
    function decompress($com) {
        $com = explode(",",$com);
        $i;$w;$k;$result;
        $dictionary = array();
        $entry = "";
        $dictSize = 256;
        for ($i = 0; $i < 256; $i++) {
            $dictionary[$i] = chr($i);}
        $w = chr($com[0]);
        $result = $w;
        for ($i = 1; $i < count($com);$i++) {
            $k = $com[$i];
            if ($dictionary[$k]) {
                $entry = $dictionary[$k];
            } else {
                if ($k === $dictSize) {
                    $entry = $w.$w[0];
                } else {
                    return null;}
            }
            $result .= $entry;
            $dictionary[$dictSize++] = $w . $entry[0];
            $w = $entry;
        }
        return $result;}
}
//How to use
$str = 'TOBEORNOTTOBEORTOBEORNOT';
$lzw = new LZW();
$com = $lzw->compress($str);
$dec = $lzw->decompress($com);
echo $com . "<br>" . $dec;
?>
```

Output :

84,79,66,69,79,82,78,79,84,256,258,260,265,259,261,263
TOBEORNOTTOBEORTOBEORNOT

Source Code LZW dengan menggunakan Java :

```
import java.util.*;
public class LZW {
    /** Compress a string to a list of output symbols. */
    public static List<Integer> compress(String uncompressed) {
        // Build the dictionary.
        int dictSize = 256;
        Map<String,Integer> dictionary = new HashMap<String,Integer>();
        for (int i = 0; i < 256; i++)
            dictionary.put("" + (char)i, i);
        String w = "";
        List<Integer> result = new ArrayList<Integer>();
        for (char c : uncompressed.toCharArray()) {
            String wc = w + c;
            if (dictionary.containsKey(wc))
                w = wc;
            else {
                result.add(dictionary.get(w));
                // Add wc to the dictionary.
                dictionary.put(wc, dictSize++);
                w = "" + c;
            }
        }
        // Output the code for w.
        if (!w.equals(""))
            result.add(dictionary.get(w));
        return result;
    }
    /** Decompress a list of output ks to a string. */
    public static String decompress(List<Integer> compressed) {
        // Build the dictionary.
        int dictSize = 256;
        Map<Integer,String> dictionary = new HashMap<Integer,String>();
        for (int i = 0; i < 256; i++)
            dictionary.put(i, "" + (char)i);
        String w = "" + (char)(int)compressed.remove(0);
        StringBuffer result = new StringBuffer(w);
        for (int k : compressed) {
            String entry;
            if (dictionary.containsKey(k))
                entry = dictionary.get(k);
```

```
        else if (k == dictSize)
            entry = w + w.charAt(0);
        else
            throw new IllegalArgumentException("Bad compressed k: " + k);

        result.append(entry);

        // Add w+entry[0] to the dictionary.
        dictionary.put(dictSize++, w + entry.charAt(0));

        w = entry;
    }
    return result.toString();
}

public static void main(String[] args) {
    List<Integer> compressed = compress("TOBEORNOTTOBEORTOBEORNOT");
    System.out.println(compressed);
    String decompressed = decompress(compressed);
    System.out.println(decompressed);
}
}
```

Output :

```
PS E:\xampp\htdocs\kompresi_data> e;; cd 'e:\xampp\htdocs\kompresi_data'; & 'c:\Users\User\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\AdoptOpenJDK\jdk-11.0.10-hotspot\bin\java.exe' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\User\AppData\Roaming\Code\User\workspaceStorage\135873d20ded123d796b36d38eaf563\redhat.java\jdt_ws\kompresi_data_1cde0b6\bin' 'LZW'
[84, 79, 66, 69, 79, 82, 78, 79, 84, 256, 258, 260, 265, 259, 261, 263]
TOBEORNOTTOBEORTOBEORNOT
```

D. Tugas

1. Buatlah 4 kelompok yang terdiri dari 4 sampai 5 mahasiswa!
2. Buatlah sebuah paper dari salah satu topik berikut untuk diterapkan pada sebuah studi kasus tertentu (text, audio, gambar, atau video):
 - a. Analisa perbandingan metode kempresi data menggunakan RLE dan Huffman Coding
 - b. Analisa perbandingan metode kempresi data menggunakan RLE dan LZW
 - c. Analisa perbandingan metode kempresi data menggunakan Huffman Coding dan LZW
 - d. Analisa perbandingan metode kempresi data menggunakan RLE, Huffman Coding, dan LZW

PERTEMUAN 6 & 7

Multimedia Networking Requirements

A. Sub Capaian Pembelajaran Mata Kuliah :

- 1) Mahasiswa mampu menyebutkan jenis-jenis peralatan atau media akses jaringan multimedia

Penilaian Mata Kuliah : UTS (30%), UAS (30%), Tugas (30%), Keaktifan (10%)

B. Dasar Teori

FFmpeg adalah perangkat lunak populer untuk memproses, mengonversi, atau memanipulasi file video dan audio. Program ini digunakan untuk hal-hal yang tidak ada habisnya seperti memutar video, menskala video, mengekstraksi informasi tentang video dan banyak lainnya. Ini adalah program open source yang luar biasa untuk penskalaan video, format transcoding, decoding, encoding, demuxing, streaming, filtering, mengonversi file audio, merekam audio / video langsung, menangani video dan file multimedia lainnya. Kerangka kerja multimedia ini dirancang untuk pemrosesan file multimedia berbasis baris perintah, dan Anda dapat mengedit dasar dengan satu perintah baris.

Walaupun program ini mudah digunakan, mengakses FFmpeg untuk menjalankan perintah agak membingungkan. Menginstal dan menggunakan FFmpeg dapat memakan banyak waktu karena Anda perlu membuka folder FFmpeg secara manual dalam command prompt atau terminal Power Shell untuk mengakses file eksekutif FFmpeg untuk menjalankan perintah yang relevan untuk pengeditan dasar. Jika Anda terjebak dengan menginstal dan menggunakan FFmpeg pada mesin Anda, Anda berada di tempat yang tepat.

Untuk mempermudah, yang harus Anda lakukan adalah menginstal FFmpeg dan menambahkan program FFmpeg ke jalur Windows 10 menggunakan variabel Lingkungan. Dengan cara ini Anda dapat langsung mengakses FFmpeg dari Command prompt atau Power Shell di direktori mana pun. Pada artikel ini, kami menjelaskan cara menginstal FFmpeg pada Windows 10 untuk mengakses FFmpeg langsung dari Command Prompt atau PowerShell.

C. Percobaan

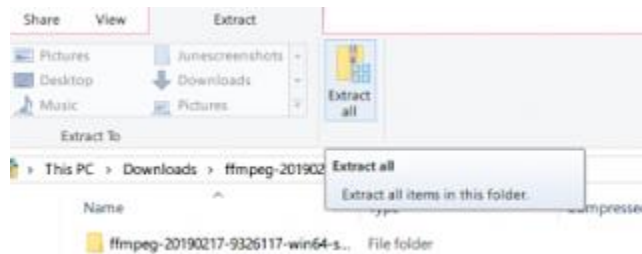
Pada percobaan ini kita akan mencoba membuat sebuah uji coba untuk memanipulasi audio dan multimedia yang lain dengan menggunakan aplikasi FFMPEG.

Instal FFmpeg pada Windows 10 :

Dapatkan build FFmpeg terbaru dari situs web resmi di <https://ffmpeg.org/download.html>. Anda dapat mengunduh versi 32-bit atau 64-bit yang sesuai dengan sistem Anda. Klik tombol Unduh Bangun untuk mulai mengunduh.

Program tidak bekerja secara langsung di Windows 10. Anda perlu menambahkan program ke jalur sistem menggunakan Variabel Lingkungan.

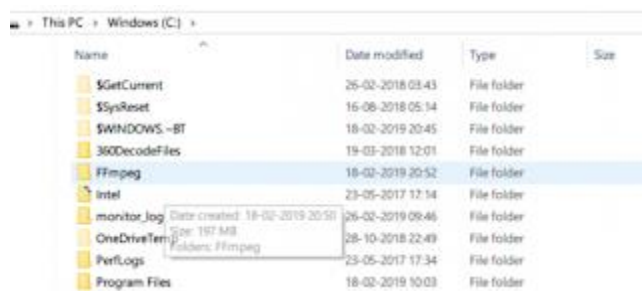
Jadi navigasikan ke folder yang diunduh di mana file zip FFmpeg adalah toko. Klik kanan pada folder Zip dan klik ekstrak dari menu drop-down.



Pilih folder atau drive tempat Anda ingin mengekstrak.

Selanjutnya, ganti nama folder yang diekstraksi. Untuk mengganti nama folder, klik kanan pada folder dan klik Ubah nama dari menu drop-down dengan judul FFmpeg-20180424-d9706f7-win64-static ke FFmpeg.

Salin folder FFmpeg dan navigasikan ke drive C. Tempel folder ke root drive C.



Setelah selesai, langkah selanjutnya adalah mengaktifkan FFmpeg menggunakan Command Prompt.

Tambahkan FFmpeg ke jalur Windows menggunakan variabel Lingkungan

Untuk menggunakan FFmpeg di Command Prompt, Anda harus terlebih dahulu menambahkan folder bin yang berisi file executable FFmpeg ke jalur Windows Anda

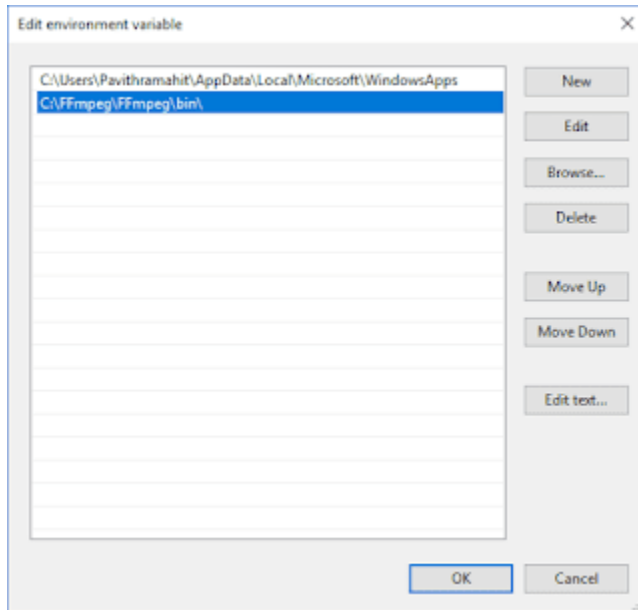
Di menu pencarian Windows, ketik Edit variabel lingkungan sistem dan klik Enter. Ini akan membuka jendela properti sistem.

Arahkan ke tombol Advanced dan klik Variabel Lingkungan di bagian bawah jendela.

Di jendela Environment Variables, Pilih Path variabel dan klik Edit untuk mengubah variabel Path.

Klik Baru dan ketik lintasan folder FFmpeg "C: \ ffmpeg \ bin \" dan klik OK.

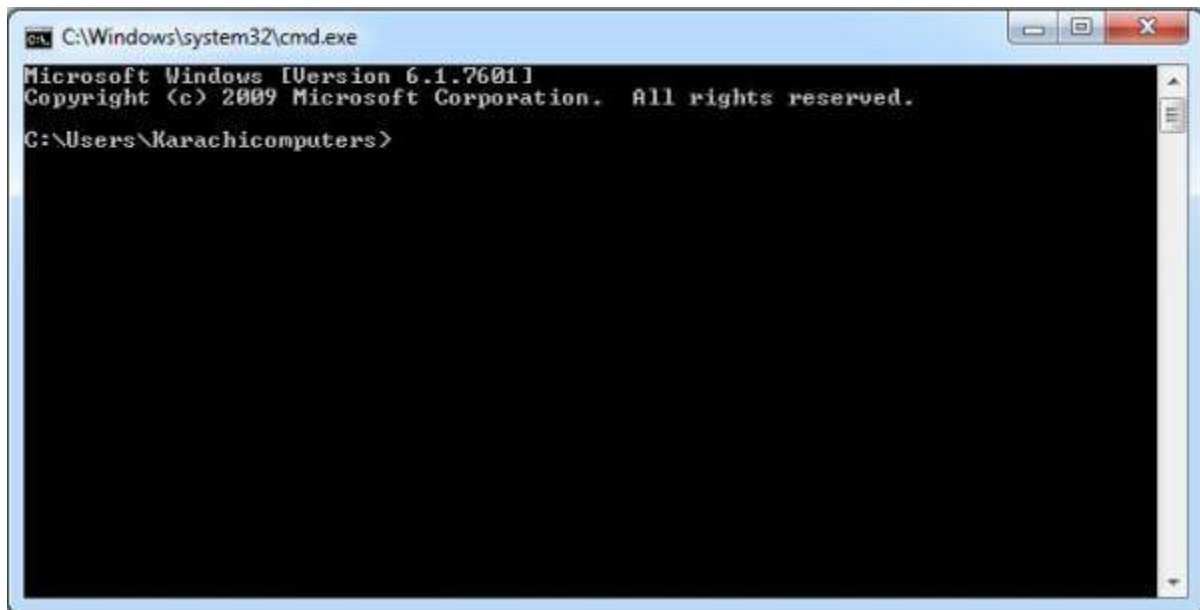
Ketikkan direktori jalur sesuai dengan drive atau folder tempat Anda meletakkan folder FFmpeg.



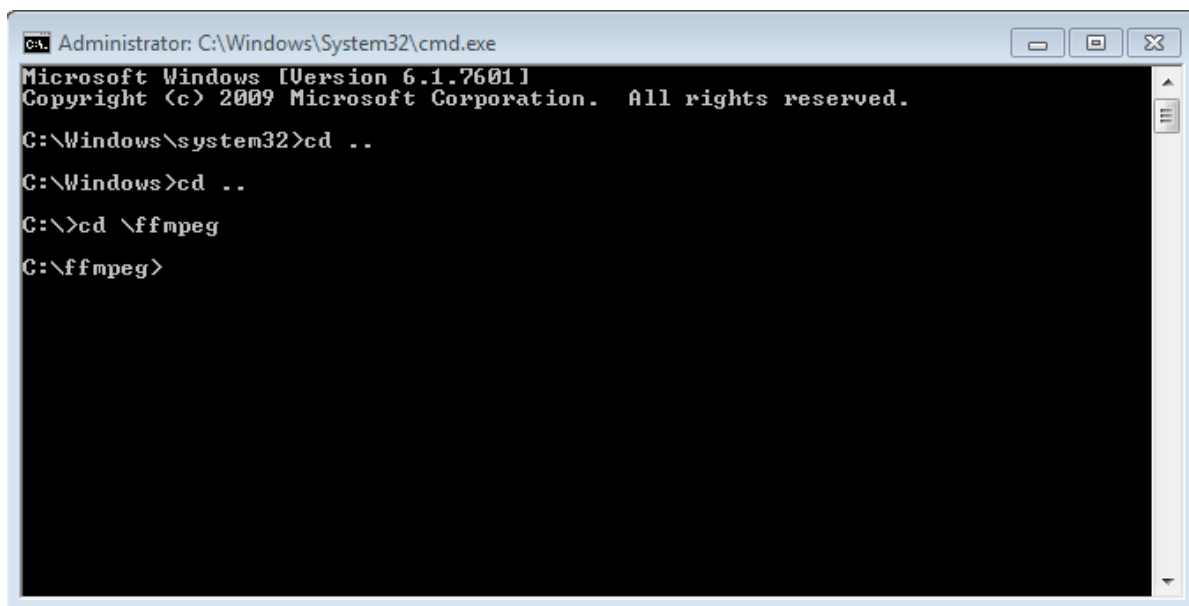
Setelah selesai, jalur akan ditambahkan ke variabel Lingkungan. Klik OK untuk menerapkan perubahan.

Starting FFmpeg

Buka Run, ketik cmd dan tekan tombol Enter. Sebuah jendela baris perintah akan muncul sebagaimana ditunjukkan di sini.



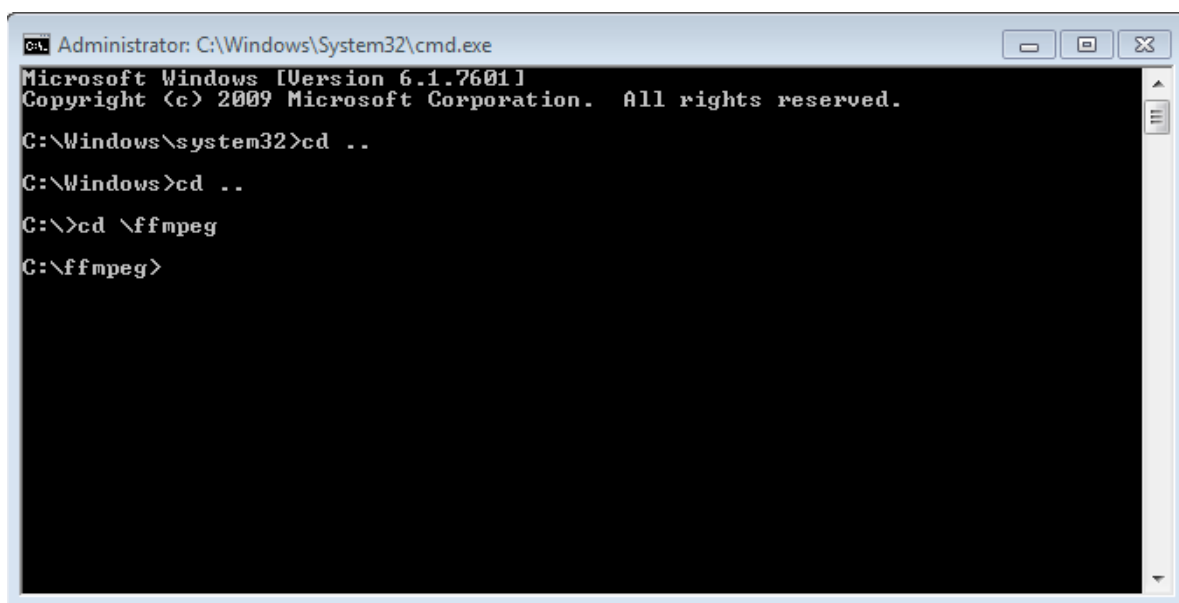
Ketik cd\ dan tekan tombol Enter. Ini akan muncul seperti di bawah ini.



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd ..
C:\Windows>cd ..
C:\>cd \ffmpeg
C:\ffmpeg>
```

Sekarang ketik `cd FFmpegTool`, perintah ini akan memungkinkan Anda untuk masuk ke dalam folder `FFmpegTool` yang telah kita buat sebelumnya (hati-hati dengan ejaan karena setiap kesalahan ejaan dalam perintah akan menyebabkan kesalahan).



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd ..
C:\Windows>cd ..
C:\>cd \ffmpeg
C:\ffmpeg>
```

Ketik `cd bin` dan tekan tombol Enter untuk mengakses konten dalam folder `bin`. Anda harus melihat berikut.

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd ..
C:\Windows>cd..
C:\>cd \ffmpeg
C:\ffmpeg>cd bin
C:\ffmpeg\bin>
```

Akhirnya, jenis ffmpeg.exe dan tekan tombol Enter. Ini akan memakan waktu 1 atau 2 detik untuk mengeksekusi perintah dan Anda harus dapat melihat hasil yang ditampilkan di bawah ini.

```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg
ffmpeg version N-64601-g0089fb7 Copyright (c) 2000-2014 the FFmpeg developers
  built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
  configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
  isynth --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
  le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreetyp
  e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
  libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
  njpeg --enable-libopus --enable-librtmp --enable-libschrödinger --enable-libsox
  r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
  enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
  --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
  libxavs --enable-libxvid --enable-decklink --enable-zlib
  libavutil      52. 92.100 / 52. 92.100
  libavcodec     55. 69.100 / 55. 69.100
  libavformat    55. 45.100 / 55. 45.100
  libavdevice    55. 13.102 / 55. 13.102
  libavfilter     4. 10.100 /  4. 10.100
  libswscale     2.  6.100 /  2.  6.100
  libswresample  0. 19.100 /  0. 19.100
  libpostproc   52.  3.100 / 52.  3.100
Hyper fast Audio and Video encoder
usage: ffmpeg [options] [[infile options] -i infile]... [[outfile options] outfi
le]...

Use -h to get full help or, even better, run 'man ffmpeg'
```

Jika Anda melihat garis-garis teks tertulis pada layar Anda (sesuai gambar di atas) maka itu berarti alat ini bekerja dengan baik dan siap untuk digunakan.

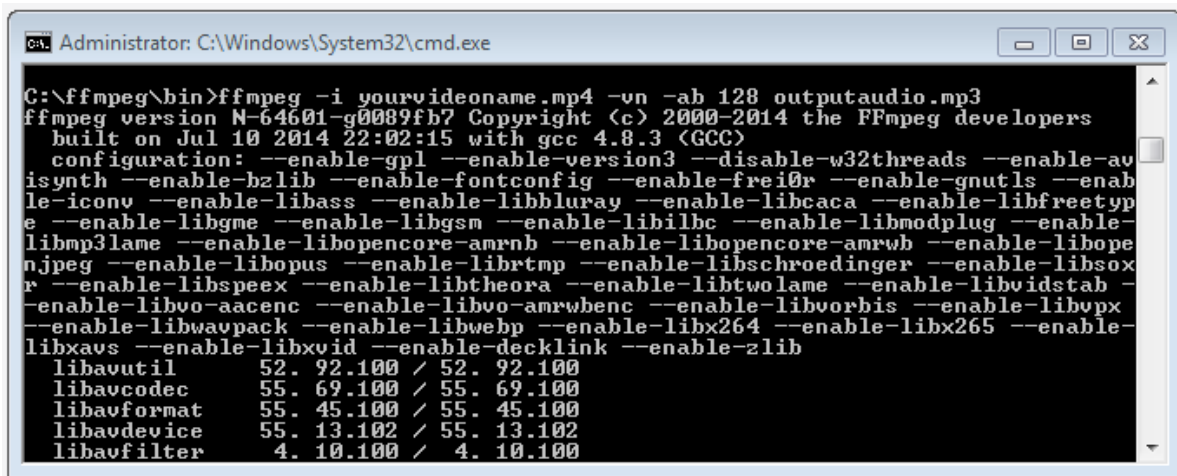
Perintah Perintah Penting

Sekarang, kita akan pergi melalui beberapa perintah penting bagi FFmpeg. Salin video dalam folder bin di mana Anda akan melakukan operasi. Saya telah menyalin video yang disebut samplevideo dan akan menggunakan video ini sepanjang sisa posting ini

1. Ekstrak Audio hanya File Video

Setelah Anda sudah menguji bahwa FFmpeg bekerja dengan baik kemudian ketik perintah berikut:

```
ffmpeg -i yourvideoname.mp4 -vn -ab 128 outputaudio.mp3
```



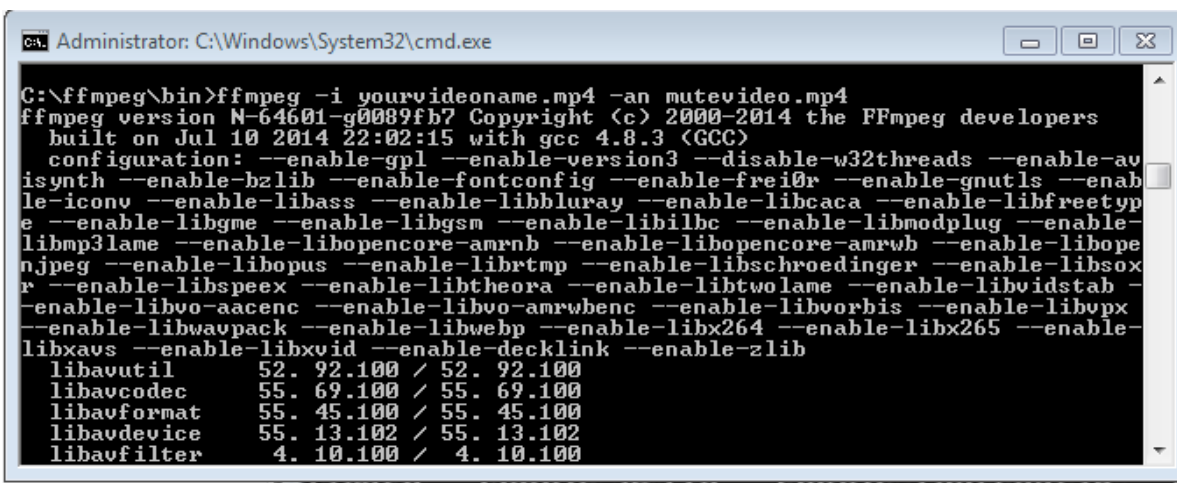
```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -i yourvideoname.mp4 -vn -ab 128 outputaudio.mp3
ffmpeg version N-64601-g0089fb7 Copyright (c) 2000-2014 the FFmpeg developers
  built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
  configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
  isynth --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
  le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreetyp
  e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
  libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
  njpeg --enable-libopus --enable-librtmp --enable-libschrödinger --enable-libsox
  r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
  enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
  --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
  libxavs --enable-libxvid --enable-decklink --enable-zlib
  libavutil      52. 92.100 / 52. 92.100
  libavcodec     55. 69.100 / 55. 69.100
  libavformat    55. 45.100 / 55. 45.100
  libavdevice    55. 13.102 / 55. 13.102
  libavfilter     4. 10.100 / 4. 10.100
```

Here -vn is used to extract audio and -ab digunakan untuk menyimpan audio 128Kbps file MP3. Anda dapat mengubah bit rate ke 256Kbps atau sesuatu yang lain. Hanya mengubah nilai setelah -AB. Semua file output akan disimpan secara otomatis dalam folder bin.

2. Mengekstrak Video Sementara Muting Audio

Dalam perintah tertulis di bawah, -an digunakan untuk menghapus audio dari file video. Perintah akan terlihat seperti:

```
ffmpeg -i yourvideoname.mp4 -an mutevideo.mp4
```

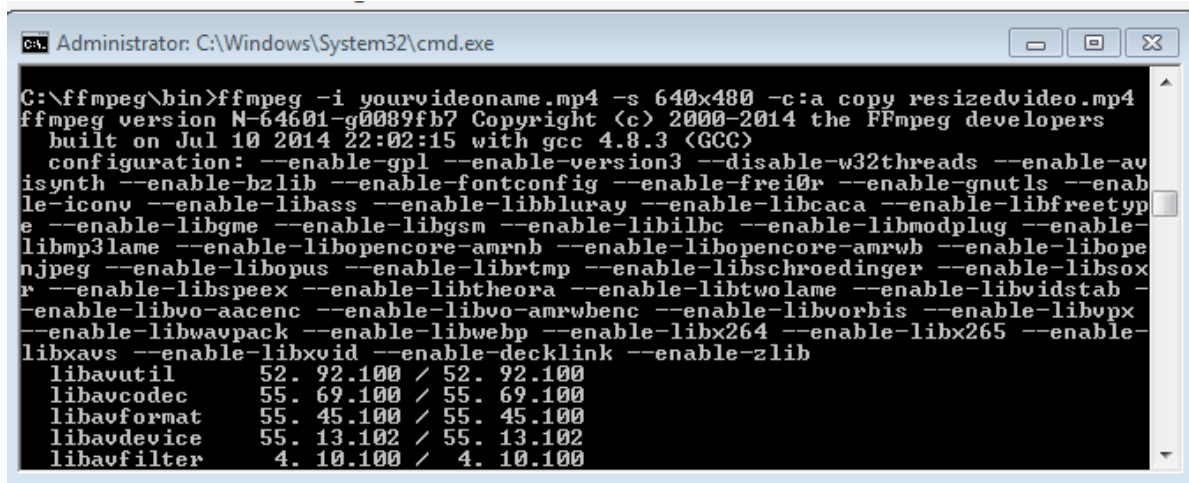


```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -i yourvideoname.mp4 -an mutevideo.mp4
ffmpeg version N-64601-g0089fb7 Copyright (c) 2000-2014 the FFmpeg developers
  built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
  configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
  isynth --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
  le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreetyp
  e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
  libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
  njpeg --enable-libopus --enable-librtmp --enable-libschrödinger --enable-libsox
  r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
  enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
  --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
  libxavs --enable-libxvid --enable-decklink --enable-zlib
  libavutil      52. 92.100 / 52. 92.100
  libavcodec     55. 69.100 / 55. 69.100
  libavformat    55. 45.100 / 55. 45.100
  libavdevice    55. 13.102 / 55. 13.102
  libavfilter     4. 10.100 / 4. 10.100
```

3. Mengubah ukuran File Video

Dalam perintah tertulis di bawah ini, -s digunakan untuk mengubah ukuran video perintah file. perintahnya sbb:

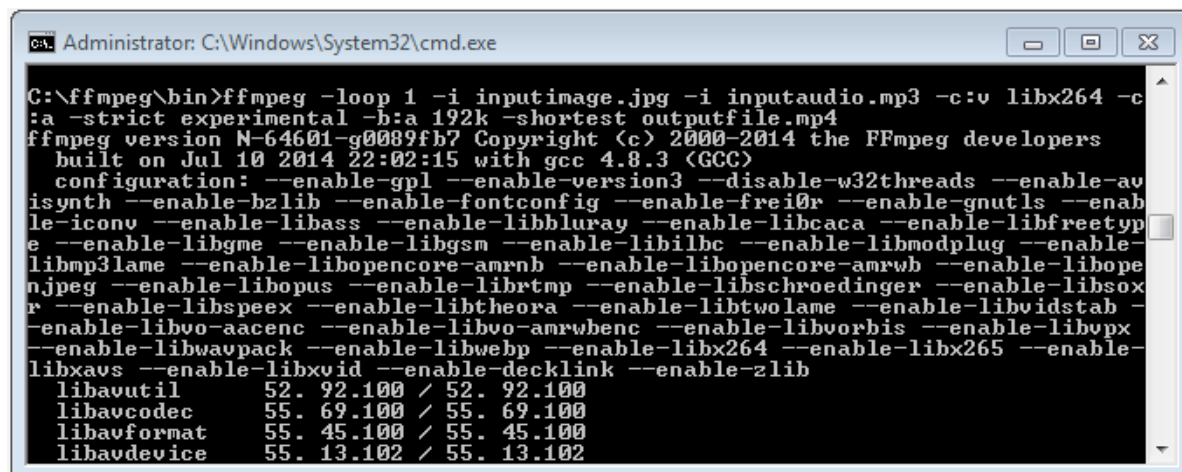
```
ffmpeg -i yourvideoname.mp4 -s 640x480 -c:a copy resizedvideo.mp4
```



```
C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -i yourvideoname.mp4 -s 640x480 -c:a copy resizedvideo.mp4
ffmpeg version N-64601-g0089fb7 Copyright (c) 2000-2014 the FFmpeg developers
  built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
  configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
  isynth --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
  le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreetyp
  e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
  libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
  njpeg --enable-libopus --enable-librtmp --enable-libschrödinger --enable-libsox
  r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
  enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
  --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
  libxavs --enable-libxvid --enable-decklink --enable-zlib
  libavutil      52. 92.100 / 52. 92.100
  libavcodec     55. 69.100 / 55. 69.100
  libavformat    55. 45.100 / 55. 45.100
  libavdevice    55. 13.102 / 55. 13.102
  libavfilter     4. 10.100 / 4. 10.100
```

4. Menambah Poster Gambar ke Audio File

Anda dapat menambahkan gambar poster untuk file audio Anda dengan mudah dan output akan menjadi file video dengan gambar yang ditampilkan di depan dan audio di latar belakang. Ini benar-benar berguna ketika meng-upload file MP3 ke video hosting dan situs berbagi. Anda harus menyalin gambar dalam folder bin. Kemudian jalankan perintah ini: `ffmpeg -loop 1 -i inputimage.jpg -i inputaudio.mp3 -c:v libx264 -c:a -strict experimental -b:a 192k -shortest outputfile.mp4`

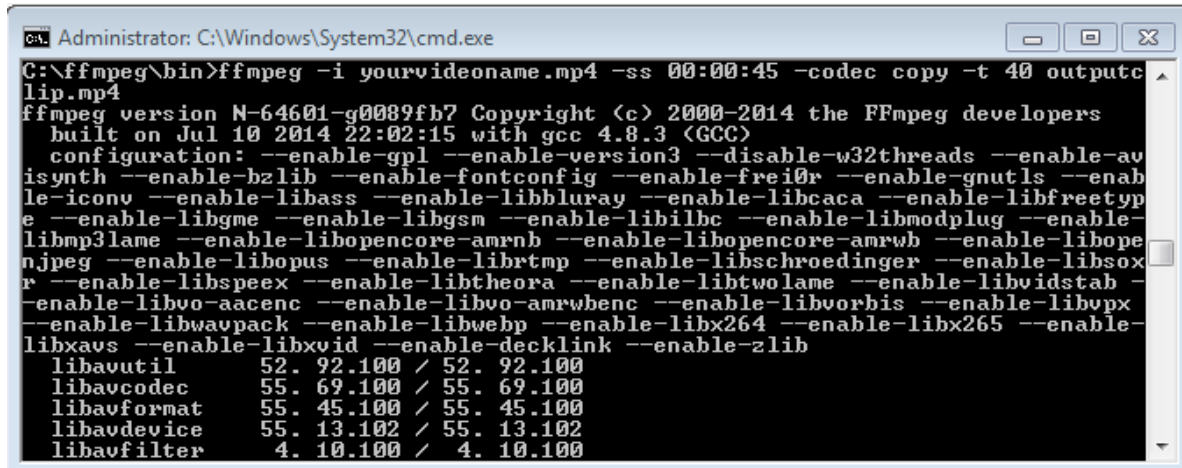


```
C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -loop 1 -i inputimage.jpg -i inputaudio.mp3 -c:v libx264 -c
:a -strict experimental -b:a 192k -shortest outputfile.mp4
ffmpeg version N-64601-g0089fb7 Copyright (c) 2000-2014 the FFmpeg developers
  built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
  configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
  isynth --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
  le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreetyp
  e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
  libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
  njpeg --enable-libopus --enable-librtmp --enable-libschrödinger --enable-libsox
  r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
  enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
  --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
  libxavs --enable-libxvid --enable-decklink --enable-zlib
  libavutil      52. 92.100 / 52. 92.100
  libavcodec     55. 69.100 / 55. 69.100
  libavformat    55. 45.100 / 55. 45.100
  libavdevice    55. 13.102 / 55. 13.102
```

5. Memotong File video ke dalam Clip kecil

The `-ss` Mendefinisikan starting time stamp (waktu di sini mulai adalah 45 detik) dan `t` mengatakan total durasi waktu untuk klip. Jadi, `rt 40` berarti 40 durasi kedua. Perintah harus seperti ini:

```
ffmpeg -i yourvideoname.mp4 -ss 00:00:45 -codec copy -t 40 outputclip.mp4
```



```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -i yourvideoname.mp4 -ss 00:00:45 -codec copy -t 40 outputclip.mp4
ffmpeg version N-64601-g00089fb7 Copyright (c) 2000-2014 the FFmpeg developers
  built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
  configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
  isynth --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
  le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreety
  e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
  libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
  njpeg --enable-libopus --enable-librtmp --enable-libschrödinger --enable-libsox
  r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
  enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
  --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
  libxavs --enable-libxvid --enable-decklink --enable-zlib
  libavutil      52. 92.100 / 52. 92.100
  libavcodec     55. 69.100 / 55. 69.100
  libavformat    55. 45.100 / 55. 45.100
  libavdevice    55. 13.102 / 55. 13.102
  libavfilter     4. 10.100 / 4. 10.100
```

Berikut `-t 00:00:59` merupakan bagian yang dibuat dari awal video untuk kedua 59 video. `-ss 00:00:59` menunjukkan cap waktu mulai untuk video. Ini berarti bahwa bagian-2 akan dimulai dari 59 detik dan akan berlanjut sampai dengan akhir file video asli.

7. Mengkonversi File Video dari Satu Format ke Format lain

Anda dapat melihat semua format yang didukung oleh FFmpeg dengan menggunakan perintah berikut:

`ffmpeg-formats` Untuk mengkonversi file video dari satu format ke format lain, perintah berikut ini digunakan:

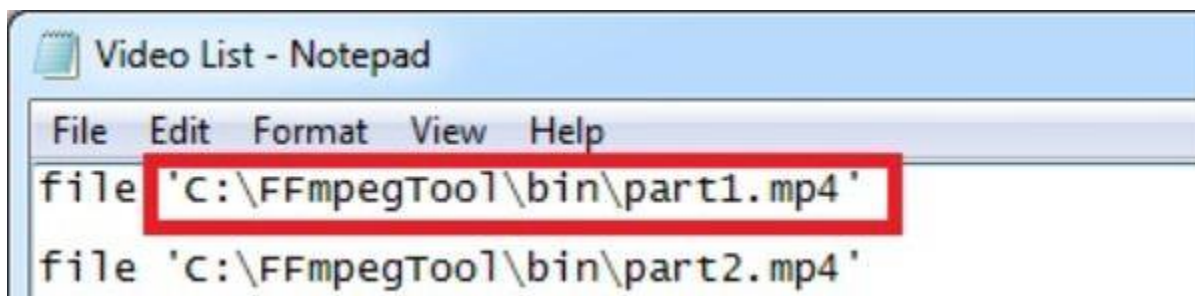
```
ffmpeg -i yourvideoname.mp4 -c:v libx264 outputfilename.wmv
```

Perintah di bawah ini adalah contoh ketika mengkonversi file `.mp4` ke file `.wmv`.

```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -i yourvideoname.mp4 -c:v libx264 outputfilename.wmv
ffmpeg version N-64601-g0089fb7 Copyright (c) 2000-2014 the FFmpeg developers
built on Jul 10 2014 22:02:15 with gcc 4.8.3 (GCC)
configuration: --enable-gpl --enable-version3 --disable-w32threads --enable-av
isyntn --enable-bzlib --enable-fontconfig --enable-frei0r --enable-gnutls --enab
le-iconv --enable-libass --enable-libbluray --enable-libcaca --enable-libfreety
e --enable-libgme --enable-libgsm --enable-libilbc --enable-libmodplug --enable-
libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libope
njpeg --enable-libopus --enable-librtmp --enable-libschoedinger --enable-libsox
r --enable-libspeex --enable-libtheora --enable-libtwolame --enable-libvidstab --
enable-libvo-aacenc --enable-libvo-amrwbenc --enable-libvorbis --enable-libvpx
--enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --enable-
libxavs --enable-libxvid --enable-decklink --enable-zlib
libavutil      52. 92.100 / 52. 92.100
libavcodec     55. 69.100 / 55. 69.100
libavformat    55. 45.100 / 55. 45.100
libavdevice    55. 13.102 / 55. 13.102
libavfilter     4. 10.100 / 4. 10.100
```

8. Menggabungkan File Video

FFmpeg juga dapat bergabung beberapa video atau audio file dengan codec yang sama. Buat file txt termasuk daftar semua file video masukan yang seharusnya digabung. File kunci diikuti dengan nama, path dan format file video. Tambahkan semua file dengan cara yang sama dalam file .txt dan menyimpan file ini Txt dalam folder bin.

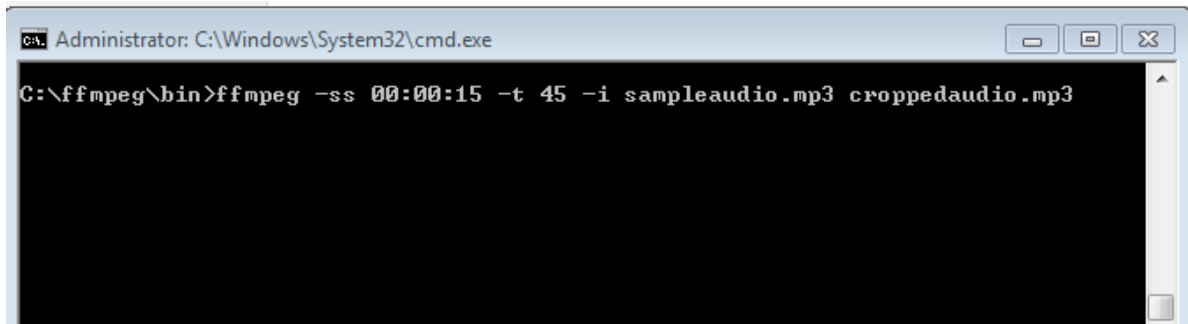


Sekarang ketik perintah berikut untuk bergabung dengan file video:
ffmpeg -f concat -i file-list.txt -c copy outputfile.mp4

```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -f concat -i file-list.txt -c copy outputfile.mp4
```

9. Memotong file Audio

Untuk tanaman bagian dari file audio, perintah berikut dapat digunakan:
ffmpeg -ss 00:00:15 -t 45 -i sampleaudio.mp3 croppedaudio.mp3
Di sini, -ss 00:00:15 adalah waktu menatap dan -t 45 adalah durasi dari file dipotong.



```
Administrator: C:\Windows\System32\cmd.exe
C:\ffmpeg\bin>ffmpeg -ss 00:00:15 -t 45 -i sampleaudio.mp3 croppedaudio.mp3
```

D. Tugas

1. Berikan Kesimpulan Praktikum yang anda lakukan!
2. Kumpulkan Hasil Percobaan diatas, tambahkan dalam laporan resmi anda tugas berikut ini!

PERTEMUAN 9 & 10

Streaming

A. Sub Capaian Pembelajaran Mata Kuliah :

- 1) Mahasiswa mampu membuat aplikasi multimedia streaming, live streaming, real-time interactive dan storage streaming

Penilaian Mata Kuliah : UTS (30%), UAS (30%), Tugas (30%), Keaktifan (10%)

B. Dasar Teori

Dalam dunia multimedia, saat ini kita berada ditahap pemakai jaringan rumah yang mulai bercampur dengan pemakai jaringan elektronik dan jaringan media. Menurut laporan dari In-Stat (www.in-stat.com), bahwa pasar menginginkan jaringan ini bergerak dengan cepat tetapi pada kenyataannya semua itu masih berjalan sangat lambat. Walaupun riset yang dilakukan oleh In-Stat menunjukkan adanya kemajuan, tetapi mayoritas pemakai masih menggunakan jaringan rumah untuk melakukan browsing di internet.

Dalam menghadapi era global dan memasuki pasar bebas di dunia internasional, teknologi komputer memegang peranan yang sangat penting dalam melakukan hubungan komunikasi melalui jaringan internet dan intranet secara efisien dan optimal. Dalam kenyataannya dunia pendidikan sebagai penyaji sumber daya manusia belum banyak yang mengetahui bagaimana sistem teknologi informasi dan komunikasi tersebut secara global dan menyeluruh. Dalam dunia pendidikan, video digunakan secara luas dalam sistem multimedia.

Videoconferencing memungkinkan kita untuk mengadakan konferensi jarak jauh secara visual, sehingga para peserta seolah-olah bertemu langsung dalam suatu konferensi. Video juga merupakan komponen penting dalam distance learning, suatu metode pengajaran yang tidak lagi terikat oleh ruang.

Pengertian video streaming

Streaming adalah sebuah teknologi untuk memanipulasi file video atau audio secara langsung ataupun dengan pre-recorder dari sebuah mesin server (web server). Dengan kata lain, file video ataupun audio yang terletak dalam sebuah server dapat secara langsung dijalankan pada UE sesaat setelah ada permintaan dari user, sehingga proses running aplikasi yang didownload berupa waktu yang lama dapat dihindari tanpa harus melakukan proses penyimpanan terlebih dahulu. Saat file video atau audio di stream, akan berbentuk sebuah buffer di komputer client, dan data video - audio tersebut akan bulai di download ke dalam buffer yang telah terbentuk pada mesin client. Dalam waktu sepersekian detik, buffer telah terisi penuh dan secara otomatis file videoaudio dijalankan oleh sistem. Sistem akan membaca informasi dari buffer dan tetap melakukan proses download file, sehingga proses streaming tetap berlangsung.

Cara simple dan gampang memahaminya, Video adalah sebuah file yang biasanya adalah yang hanya dibuka dan dimainkan oleh aplikasi multimedia yang digunakan seperti Windows Media Player, Winamp, dan lain sebagainya, yang dimaksud dengan video streaming adalah Anda bisa melihat langsung video pada satu halaman web tanpa harus download terlebih dahulu.

Cara kerja video streaming

Pada awalnya, data dari source (bisa berupa audio maupun video) akan di-capture dan disimpan pada sebuah buffer yang berada pada memori komputer (bukan media penyimpanan seperti harddisk) dan kemudian di-encode sesuai dengan format yang diinginkan. Dalam proses encode ini, user dapat mengkompresi data sehingga ukurannya tidak terlalu besar (bersifat optional). Namun pada aplikasi streaming menggunakan jaringan, biasanya data akan dikompresi terlebih dahulu sebelum dilakukan streaming, karena keterbatasan bandwidth jaringan. Setelah di-encode, data akan di-stream ke user yang lain. User akan melakukan decode data dan menampilkan hasilnya ke layar user. Waktu yang dibutuhkan agar sebuah data sampai mulai dari pemancar sampai penerima disebut dengan latency.

Beberapa hal yang berkaitan dengan video streaming.

1. Media server

Media server. digunakan untuk mendistribusikan on-demand atau webcast suatu konten ke klien. Juga bertanggung jawab untuk mencatat semua aktivitas streaming, yang nantinya digunakan untuk billing dan statistik. Player, dibutuhkan untuk

menampilkan atau mempresentasikan konten multimedia (data stream) yang diterima dari media server. File-file khusus yang disebut metafile digunakan untuk mengaktifkan player dari halaman web. Metafile berisi keterangan dari konten multimedia. Browser web menunduh dan meneruskan ke player yang tepat untuk mempresentasikannya. Selain itu, juga berfungsi untuk melakukan dekompresi.

a. Media streaming

Streaming media merupakan suatu sistem yang terdiri dari komponen-komponen yang saling mendukung. Sebuah teknologi yang memungkinkan distribusi data audio, video dan multimedia secara real-time melalui Internet.

Media streaming merupakan pengiriman media digital (berupa video, suara dan data) agar bisa diterima secara terus-menerus (stream). Data tersebut dikirim dari sebuah server aplikasi dan diterima serta ditampilkan secara real-time oleh aplikasi pada komputer klien Streaming suara sering juga disebut sebagai streaming media.

b. Format streaming

Format Real Player (.rm/.ra/.ram) Windows media (.asf/.wmx/.asx) QuickTime (.mov)

c. Komponen dalam streaming media

Media source, yaitu sumber yang akan menampilkan suatu konten presentasi. Media source dapat berupa sumber yang sifatnya live, seperti kamera video ataupun microphone.

2. Multimedia streaming

Streaming multimedia adalah suatu teknologi yang mampu mengirimkan file audio dan video digital secara realtime pada jaringan computer.

Streaming dapat dilakukan pada bandwidth dengan kecepatan rendah Web master tidak perlu risau dengan bandwidth. Web master tidak dibatasi oleh besar file Hanya dapat dilihat pada saat online kualitas gambar jelek.

Kendala-kendala yang dapat terjadi dalam melakukan streaming multimedia

Bandwidth : Bandwidth sangat berpengaruh terhadap kualitas presentasi suatu data stream. Di samping kondisi jaringan juga mempengaruhi bandwidth, hal yang perlu diperhatikan adalah ukuran data stream harus sesuai dengan kapasitas bandwidth jaringan. Untuk mengatasinya digunakan kompresi data dan penggunaan buffer.

Sinkronisasi dan delay : Agar media yang berbeda sampai dan dipresentasikan pada user seperti aslinya, maka media tersebut harus tersinkronisasikan sesuai dengan timeline presentasi tersebut dan delay seminimal mungkin. Adanya kerugian sinkronisasi dan delay dapat disebabkan oleh kondisi jaringan yang buruk, sehingga mengakibatkan timeline presentasi menjadi kacau.

Karakteristik Multimedia Data

- Voluminous : Membutuhkan data rate tinggi dan berukuran besar
- Real-time and Interactive : Membutuhkan low delay, membutuhkan sinkronisasi dan interaktif.

3. Video On Demand

Video on Demand adalah sebuah istilah penyajian video yang bisa diakses secara online melalui jaringan internet atau intranet, dimana pemirsa bisa melihat kapan pun sepuasnya dan berulang ulang tanpa harus terikat waktu dan tempat. Video bisa disajikan langsung secara streaming atau didownload.

Ada beberapa pengetahuan dasar yang perlu diketahui sebelum menjelaskan Video/Modul on Demand antara lain :Local Area Network, Karakteristik video digital, Format File Video, Kompresi Video Digital, Standar Kompresi Video, Streaming, Internet broadcasting, Streaming Interaktif, Format Streaming, Codec, Bandwidth.

4. Internet Protocol Based Television (IPTV) / Digital TV

IPTV merupakan penyediaan layanan streaming TV secara langsung via jaringan IP bandwidth lebar. Layanan ini bersifat multicast, yakni dari satu sumber untuk banyak pengakses secara bersamaan. “Seperti TV biasa, tetapi punya layanan plus, plus, plus. Jadi, IPTV ini akan memanjakan penonton dengan program-program TV interaktif. Ada minimal 4 tipe layanan yang harus didukung oleh IPTV, yaitu Live TV, VoD (Video on Demand), TSTV (time-shifted TV) , PVR (Personal VideoRecording).

5. Video Conferencing

Arti istilah video conferencing dianggap berkaitan erat dengan pengertian berikut termasuk di dalamnya penggunaan audio sistem atau audio conferencing, yaitu penggunaan komputer jaringan yang memungkinkan penggunaanya untuk melakukan interaksi berupa gambar dan suara.

HTML 5

HTML5 merupakan standar berikutnya dari revisi **HTML** versi sebelumnya. seperti generasi sebelumnya, **HTML 4.01** dan **XHTML 1.1**, **HTML 5.0** merupakan standar ntuk struktur dan menampilkan isi di **WWW** (World Wide Web).

Standar baru dari **HTML5** ini juga berhubungan dengan fitur seperti video playback dan drag-and-drop, yang pada versi sebelumnya masih tergantung pada plug-in dari pihak ketiga seperti **Adobe Flash** dan **Microsoft Silverlight**.

Secara umum, format file video yang bisa dikenali antara lain ogg/ogv, mp4, dan WebM. Tiap-tiap format video di atas ditangani oleh video codec yang berbeda-beda, ogg menggunakan video codec bernama Theora, mp4 menggunakan H.264 dan WebM menggunakan VP8. Video codec ini sangat tergantung pada browser, sebagai contoh Internet Explorer 9 baru bisa mengenali video codec H.264, sedangkan Firefox sejak versi 3.5 mampu mengenali file WebM dan ogg sekaligus.

Tabel 1. Format Video dan browser yang support

Browser	MP4	WebM	Ogg
Internet Explorer 9	YES	NO	NO
Firefox 4.0	NO	YES	YES
Google Chrome 6	YES	YES	YES
Apple Safari 5	YES	NO	NO
Opera 10.6	NO	YES	YES

MP4 = MPEG 4 files with H264 video codec and AAC audio codec
 WebM = WebM files with VP8 video codec and Vorbis audio codec
 Ogg = Ogg files with Theora video codec and Vorbis audio codec.

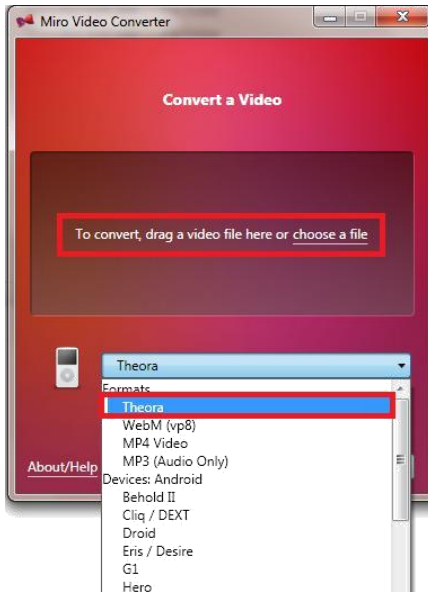
TUGAS PENDAHULUAN

1. Siapkan file video yang berdurasi 2 menit dengan format mp4.

C. Percobaan

A. Convert file video

1. Jalankan program Miro Video Converter untuk melakukan perubahan format video dari mp4 ke ogv.

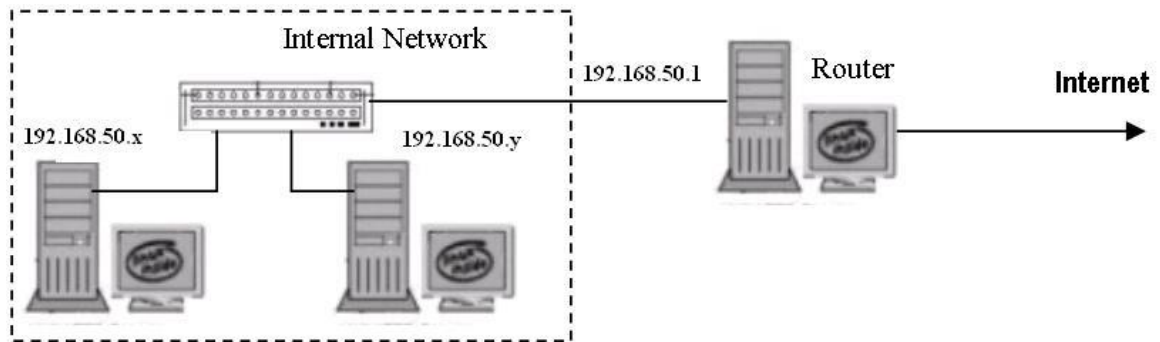


NB : Pilih file mp4 yang akan diubah formatnya. Dan pilih Theora untuk format video tujuan. Klik Convert untuk start proses.

2. Lakukan perubahan ke format WebM juga.

B. Instalasi Server dan HTML5

1. Bangunlah jaringan seperti berikut :



NB:

Setting satu sebagai server dan satunya sebagai client.

2. Lakukan instalasi XAMPP pada sisi server.
3. Buat file HTML 5 untuk menampilkan data video sebagai berikut :

```
<html>
<head>
<title>File Video</title>
</head>
<body>
<p>
<video src="pens.ogv" controls>
Browser tidak support.
</video>
</p>
</body>
</html>
```

NB:

src : file hasil convert di atas

controls : untuk menampilkan tombol play, pause dan volume.

Atribut lainnya adalah

```
<video src="pens.ogv" width="320" height="240" loop controls></video>
```

Width & height : untuk menentukan ukuran video di browser

Loop : agar video jalan terus menerus

4. Lakukan pengaksesan dari client dengan menggunakan google chrome.
http://no_ip_server/html5/coba.html



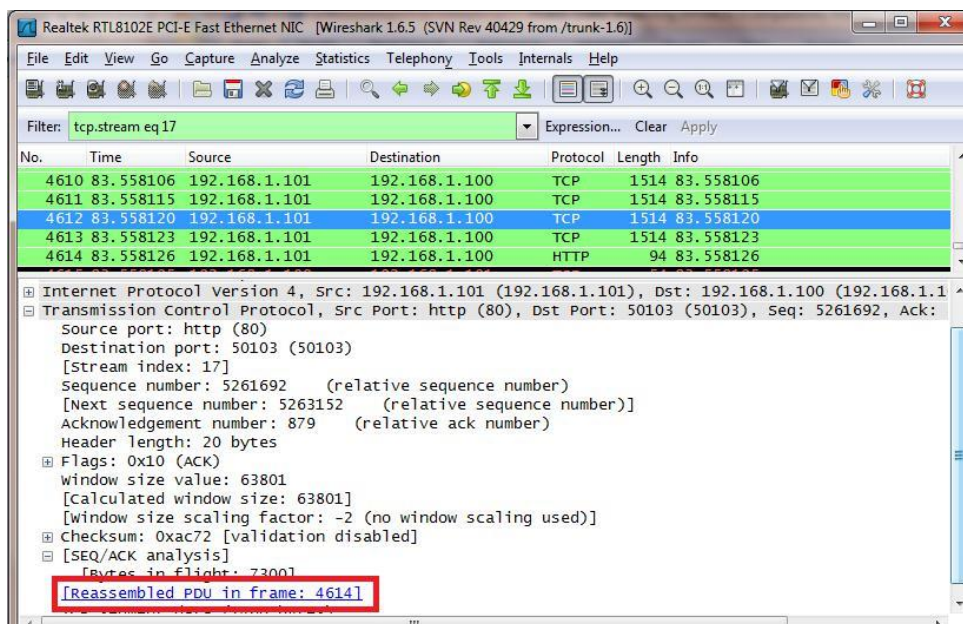
5. Lakukan pengaksesan juga dari browser yang lainnya seperti internet explorer atau firefox. Isi tabel berikut :

Format video/Browser	Google chrome	IE	Firefox
Ogv			
mp4			
WebM			

Centang jika berhasil, dan silang jika gagal.

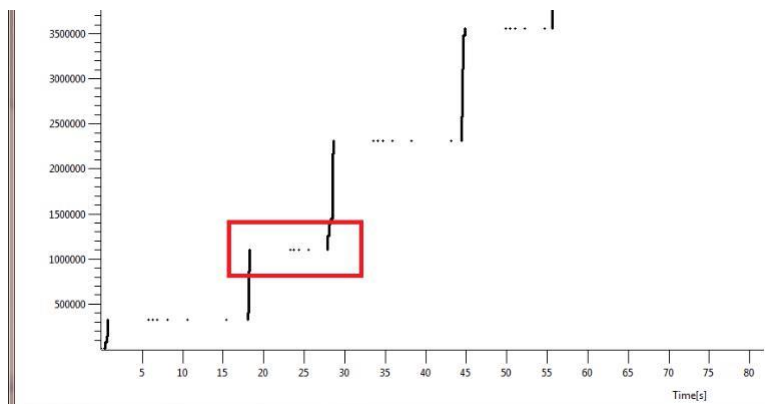
C. Analisa QoS dengan wireshark

1. Tutuplah browser anda yang digunakan untuk akses ke server.
2. Jalankan wireshark.
3. Buka kembali browser yang support html5 untuk koneksi ke server.
4. Setelah selesai, stop wireshark.
5. Analisa paket data pada wireshark :
 - a. Pada port berapa yang digunakan oleh client untuk koneksi ke server ?
 - b. Amati apakah paket-paket data video merupakan hasil fragmentasi atau bukan ?
 - c. Berapa banyak segmen data yang digunakan oleh protokol TCP untuk mengirim data video (bandingkan dengan ukuran file video yang sebenarnya). Cek dari gambar berikut ini :



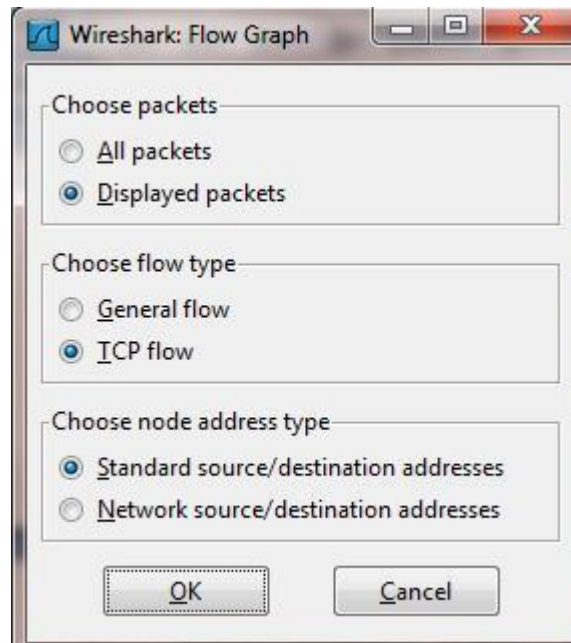
NB: Double click pada tanda di atas.

6. Cek buffering yang dilakukan oleh client ketika menerima data. Dari Statistics | TCP Stream Graphs | Time Sequence Graph (Stevens)

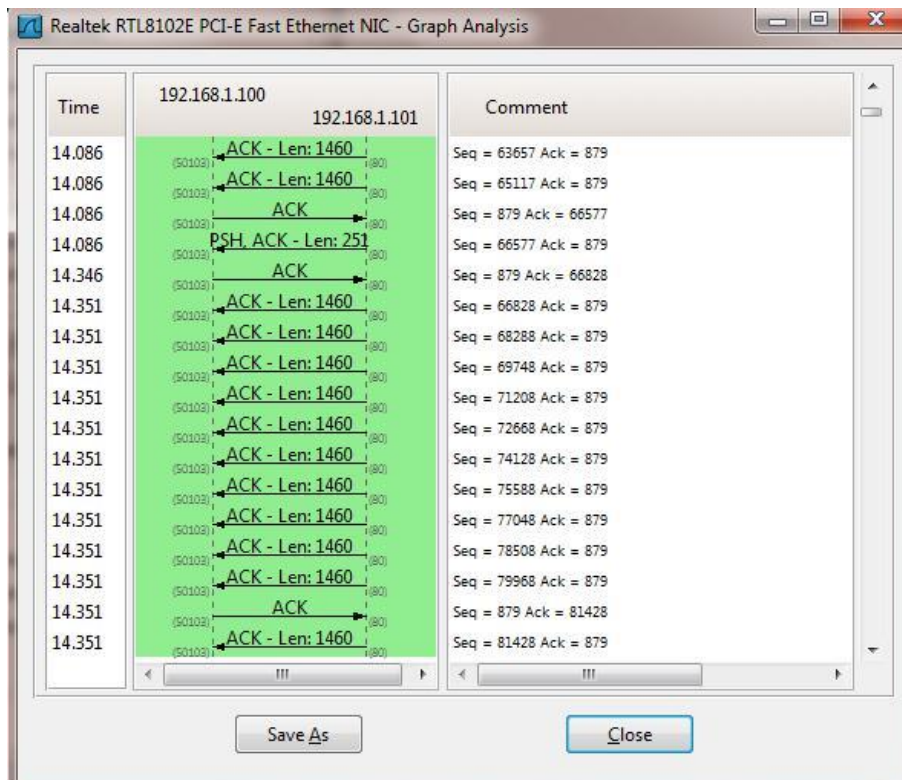


NB:

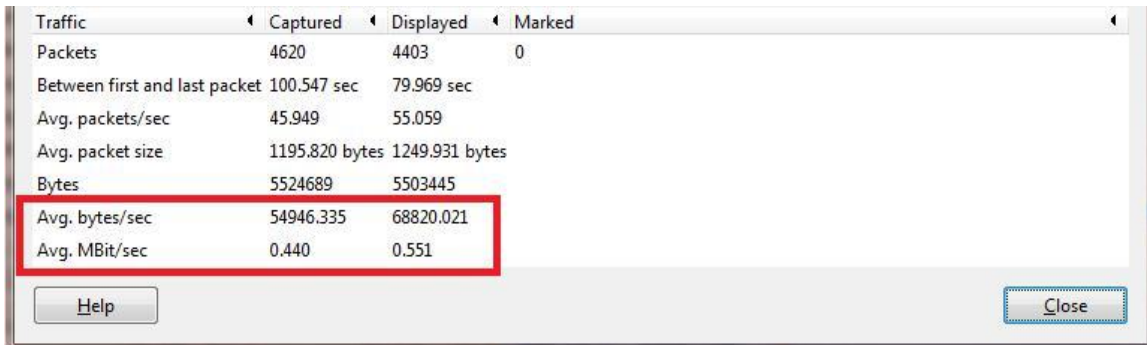
Buffering juga bisa diamati dari Statistics | Flow Graph.



Hasilnya :



7. Catat throughput jaringan yang dihasilkan. Dari Statistics | Summary.



The screenshot shows the 'Statistics Summary' window in Wireshark. It displays a table with columns for 'Captured', 'Displayed', and 'Marked'. The 'Avg. bytes/sec' row is highlighted with a red border. Below the table are 'Help' and 'Close' buttons.

Traffic	Captured	Displayed	Marked
Packets	4620	4403	0
Between first and last packet	100.547 sec	79.969 sec	
Avg. packets/sec	45.949	55.059	
Avg. packet size	1195.820 bytes	1249.931 bytes	
Bytes	5524689	5503445	
Avg. bytes/sec	54946.335	68820.021	
Avg. MBit/sec	0.440	0.551	

8. Catat delay yang dihasilkan. Petunjuk :

Untuk mencari delay

Untuk perhitungan delay setting terlebih dulu wireshark

Pilih tab edit preference (bagian atas-bergambar obeng bersilang) Lalu pada pilihan 'user interface' klik 'columns'

Pada pilihan paling bawah yaitu 'info' ganti dengan 'time (format as specified)' Lalu pilih 'file' kemudian 'export' file lalu save as type csv (*.csv)

Buka microsoft office untuk membuka file yg telah disimpan Jika ada pilihan tentang format file yg berbeda pilih 'yes'

Hitung rata-rata nilai delay pada bagian 'info'

Letakkan cursor dan klik pada bagian bawah nilai -nilai 'info' Lalu pada pilihan 'autosum' pilih 'average'

NB:

Untuk membaca file csv di excel 2007, pilih Data | From Text, pilih file csv di atas sehingga akan muncul berikut :

Text Import Wizard - Step 1 of 3

The Text Wizard has determined that your data is Delimited.
If this is correct, choose Next, or choose the data type that best describes your data.

Original data type

Choose the file type that best describes your data:

Delimited - Characters such as commas or tabs separate each field.

Fixed width - Fields are aligned in columns with spaces between each field.

Start import at row: File origin:

Preview of file C:\Users\zenhadi\Documents\ws3..

1	"No."	"Time"	"Source"	"Destination"	"Protocol"	"Length"	"Info"
2	"1"	"0.000000"	"192.168.1.101"	"192.168.1.255"	"NBNS"	"92"	"0.000000"
3	"2"	"0.749808"	"192.168.1.101"	"192.168.1.255"	"NBNS"	"92"	"0.749808"
4	"3"	"1.499790"	"192.168.1.101"	"192.168.1.255"	"NBNS"	"92"	"1.499790"
5	"4"	"2.252211"	"fe80::ec62:ca9b:804a:8b21"	"ff02::1:3"	"LLMNR"	"86"	"2.252211"

Buttons: Cancel, < Back, Next >, Finish

Text Import Wizard - Step 2 of 3

This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

Delimiters

Tab

Semicolon

Comma

Space

Other:

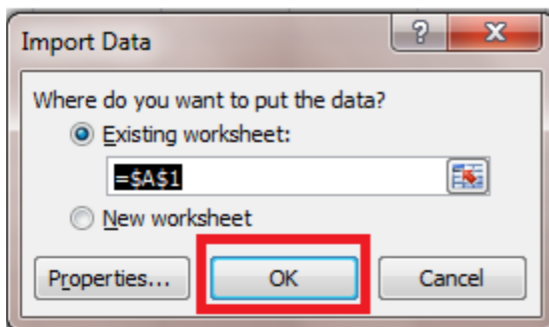
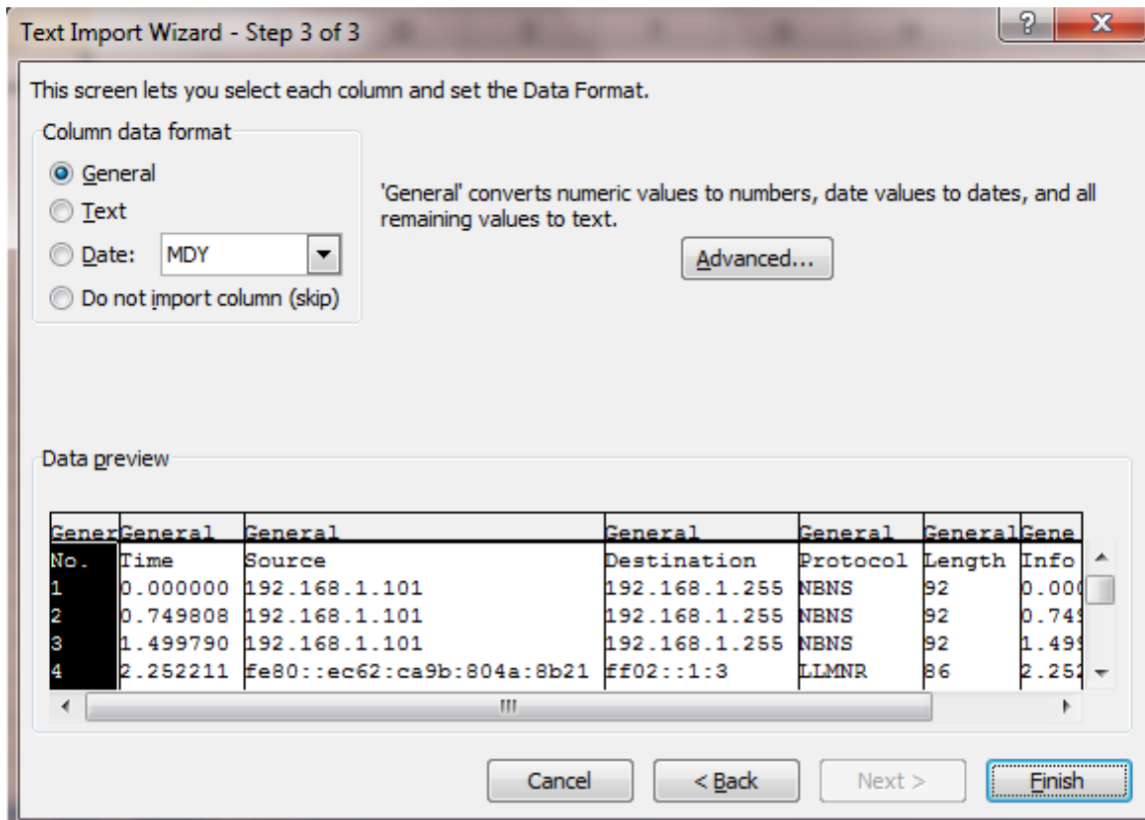
Treat consecutive delimiters as one

Text qualifier:

Data preview

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.101	192.168.1.255	NBNS	92	0.000000
2	0.749808	192.168.1.101	192.168.1.255	NBNS	92	0.749808
3	1.499790	192.168.1.101	192.168.1.255	NBNS	92	1.499790
4	2.252211	fe80::ec62:ca9b:804a:8b21	ff02::1:3	LLMNR	86	2.252211

Buttons: Cancel, < Back, Next >, Finish



	A	B	C	D	E	F	G
1	No.	Time	Source	Destination	Protocol	Length	Info
2	1	0	192.168.1.101	192.168.1.255	NBNS	92	0
3	2	0.749808	192.168.1.101	192.168.1.255	NBNS	92	0.749808
4	3	1.49979	192.168.1.101	192.168.1.255	NBNS	92	1.49979
5	4	2.252211	fe80::ec62:ca9b:804a:8b21	ff02::1:3	LLMNR	86	2.252211
6	5	2.252447	192.168.1.101	224.0.0.252	LLMNR	66	2.252447
7	6	2.351885	fe80::ec62:ca9b:804a:8b21	ff02::1:3	LLMNR	86	2.351885
8	7	2.352018	192.168.1.101	224.0.0.252	LLMNR	66	2.352018
9	8	2.552226	192.168.1.101	192.168.1.255	NBNS	92	2.552226
10	9	3.302033	192.168.1.101	192.168.1.255	NBNS	92	3.302033
11	10	4.051955	192.168.1.101	192.168.1.255	NBNS	92	4.051955
12	11	4.804434	fe80::ec62:ca9b:804a:8b21	ff02::1:3	LLMNR	86	4.804434
13	12	4.804656	192.168.1.101	224.0.0.252	LLMNR	66	4.804656
14	13	4.904019	fe80::ec62:ca9b:804a:8b21	ff02::1:3	LLMNR	86	4.904019
15	14	4.904142	192.168.1.101	224.0.0.252	LLMNR	66	4.904142

Lakukan perhitungan average, dengan memperhatikan paket data pertama kali akses video sampai berakhirnya proses streaming.

D. Tugas

1. Berikan Kesimpulan Praktikum yang anda lakukan!
2. Kumpulkan Hasil Percobaan diatas, tambahkan dalam laporan resmi anda tugas untuk membuat web video streaming dengan tampilan yang lebih bagus!