

Laporan Tugas Akhir Normality Lab

Kelompok 6: Wahdana Djunaedi, Hafizh Zaki Rais, Muhammad Daffa Alimuddin Pratama
Dosen Pengampu: Christiana Anggraeni Putri, SST., M.Si.

2025-12-17

Bab 1. Pendahuluan

1.1 Latar Belakang

Uji kenormalan merupakan salah satu tahapan penting dalam analisis statistik inferensia, khususnya pada metode statistik parametrik. Kesalahan dalam menentukan apakah data berdistribusi normal atau tidak dapat menyebabkan pemilihan metode analisis yang tidak tepat dan berdampak pada kesimpulan penelitian.

Dalam praktik analisis data, pengujian normalitas biasanya hanya dilakukan menggunakan satu uji statistik formal tanpa mempertimbangkan karakteristik data secara menyeluruh, seperti ukuran sampel dan bentuk distribusi. Pendekatan tersebut dapat menghasilkan interpretasi yang kurang tepat, terutama ketika hasil uji statistik dipengaruhi oleh sensitivitas terhadap ukuran sampel.

Berdasarkan kondisi tersebut, dikembangkan aplikasi *Normality Lab* berbasis *R Shiny* yang mengintegrasikan statistik deskriptif, visualisasi distribusi, dan uji kenormalan formal dalam satu lingkungan analisis yang interaktif. Aplikasi ini ditujukan bagi pengguna yang telah memiliki pemahaman dasar statistik sehingga dapat digunakan secara langsung tanpa penyediaan panduan penggunaan khusus. Dengan demikian, aplikasi *Normality Lab* diharapkan mampu membantu pengguna dalam mengevaluasi kenormalan data secara cepat, sistematis, dan menyeluruh.

1.2 Tujuan

Tujuan pembangunan aplikasi *Normality Lab* adalah sebagai berikut:

1. Menyediakan aplikasi berbasis web untuk pengujian normalitas data numerik.
2. Menggabungkan statistik deskriptif, visualisasi, dan uji normalitas formal dalam satu aplikasi.
3. Menyesuaikan pemilihan uji normalitas berdasarkan ukuran sampel data.
4. Memberikan interpretasi hasil uji normalitas yang mudah dipahami pengguna.

Bab 2. Pembahasan

2.1 Tahapan Pembangunan Aplikasi

1. Tahap Perencanaan

Kelompok menentukan tujuan pembuatan aplikasi, dari situ kelompok berdiskusi terkait aplikasi seperti apa yang ingin dibuat, fitur-fitur utama yang akan dibuat, serta pembagian tugas anggota kelompok

2. Tahap Perancangan Aplikasi

Aplikasi terdiri dari 6 tab yaitu : 1. Home 2. Unggah dan deskripsi data 3. Visualisasi distribusi 4. Uji Formal Normalitas 5. Analisis Grouping 6. Kesimpulan Final Tab diisi Oleh fungsi fungsi yang sesuai dengan kebutuhan yang ada di server. berikut library yang dipakai, dan helper function yang akan digunakan di server LIBRARY & HELPER FUNCTION

```
library(shiny)          # Framework utama untuk membuat aplikasi web interaktif
library(shinydashboard) # Layout dashboard (sidebar, header, box) ala admin panel
library(plotly)         # Grafik interaktif (zoom, hover, tooltip)
library(bslib)           # Kustomisasi tema Bootstrap (warna, font, mode light/dark)
library(readxl)          # Membaca file Excel (.xls, .xlsx)
library(moments)         # Statistik momen: skewness dan kurtosis
library(tseries)         # Uji statistik time series (Jarque-Bera test, dll)
library(tidyverse)        # Kumpulan paket data science (dplyr, tidyr, readr, ggplot2, dsb)
library(fontawesome)     # Ikon Font Awesome untuk UI Shiny
library(nortest)          # Uji normalitas (Lilliefors)

PRIMARY_COLOR <- "#00A388"    # Warna utama tema
SECONDARY_COLOR <- "#2F4858"   # Warna sekunder tema
ACCENT_COLOR     <- "#FFB600"   # Warna aksen/highlight

# Tema UI Shiny
light_theme <- bs_theme(
  version = 5,
  bootswatch = "flatly",
  primary = PRIMARY_COLOR,
  secondary = SECONDARY_COLOR,
  success = "#2ECC71",
  info = "#3498DB",
  base_font = font_google("Inter"),
  heading_font = font_google("Playfair Display", wght = "700")
)
# HELPER FUNCTION DI SERVER
# Fungsi uji normalitas untuk Report/Laporan dalam bentuk HTML
evaluate_normality <- function(x, alpha = 0.05) {

  x <- x[!is.na(x)]
  n <- length(x)

  if (n < 3) {
    return(list(
      n = n,
      test_name = NA,
      p_value = NA,
      decision = "DATA_TIDAK_CUKUP"
    ))
  }

  if (n >= 3 && n < 30) {
    test_name <- "Shapiro-Wilk"
    p_value <- tryCatch(
      shapiro.test(x)$p.value,
      error = function(e) NA
    )
  }
}
```

```

        )
    } else {
jb <- tryCatch(
  jarque_bera_manual(x),
  error = function(e) NULL
)

test_name <- "Jarque-Bera (Manual)"
p_value <- if (!is.null(jb)) jb$p.value else NA
}

if (is.na(p_value)) {
  decision <- "UJI_GAGAL"
} else if (p_value > alpha) {
  decision <- "GAGAL_TOLAK_H0"
} else {
  decision <- "TOLAK_H0"
}

list(
  n = n,
  test_name = test_name,
  p_value = p_value,
  decision = decision
)
}
#Fungsi chi square test(GoF)
chi_sq_test <- function(x) {
  # Pembagian data ke dalam 5-10 bins (sesuai aturan  $N/k \geq 5$ )
  N <- length(x)
  k <- max(5, floor(N / 50)) # Aturan sederhana untuk jumlah bins
  k <- min(k, 15) # Batas maksimal bins

  # Menghitung frekuensi observasi dan frekuensi harapan
  hist_info <- hist(x, breaks = k, plot = FALSE)
  observed_freq <- hist_info$counts

  # Menghitung probabilitas ( $P(a < X < b)$ ) untuk setiap bin di bawah Normal
  probabilities <- diff(pnorm(hist_info$breaks, mean(x), sd(x)))
  expected_freq <- probabilities * N

  # Menggabungkan bins jika frekuensi harapan terlalu kecil (< 5)
  while (any(expected_freq < 5) && length(expected_freq) > 2) {
    idx_small <- which.min(expected_freq)
    if (idx_small == length(expected_freq)) {
      # Gabung bin terakhir dengan bin sebelumnya
      expected_freq[idx_small - 1] <- expected_freq[idx_small - 1] + expected_freq[idx_small]
      observed_freq[idx_small - 1] <- observed_freq[idx_small - 1] + observed_freq[idx_small]
      expected_freq <- expected_freq[-idx_small]
      observed_freq <- observed_freq[-idx_small]
    } else {
      # Gabung bin kecil dengan bin setelahnya
      expected_freq[idx_small + 1] <- expected_freq[idx_small + 1] + expected_freq[idx_small]
    }
  }
}

```

```

        observed_freq[idx_small + 1] <- observed_freq[idx_small + 1] + observed_freq[idx_small]
        expected_freq <- expected_freq[-idx_small]
        observed_freq <- observed_freq[-idx_small]
    }
}

# Jika masih ada bin < 5 setelah penggabungan, uji tidak dapat diandalkan
if (any(expected_freq < 5)) {
    return(NULL)
}

# Derajat kebebasan = jumlah bins - 1 (estimasi mean) - 1 (estimasi sd) - 1
# df = length(expected_freq) - 3 (jika mean & sd diestimasi)
df_chi <- length(expected_freq) - 3
if (df_chi < 1) return(NULL) # Pastikan df > 0
    # Hitung statistik Chi-Square
chi_sq_stat <- sum((observed_freq - expected_freq)^2 / expected_freq)
p_val <- pchisq(chi_sq_stat, df = df_chi, lower.tail = FALSE)

list(
    statistic = c(`X-squared` = chi_sq_stat),
    p.value = p_val,
    method = "Chi-squared Goodness-of-Fit Test",
    parameter = c(df = df_chi, `N bins` = length(expected_freq))
)
}

#Hitung Statistik Jarque Bera
jarque_bera_manual <- function(x) {
    x <- as.numeric(x)
    x <- x[!is.na(x)]

    n <- length(x)
    if (n < 20) return(NULL)
    if (sd(x) == 0) return(NULL)

    m <- mean(x)
    s <- sd(x)

    skewness <- mean((x - m)^3) / s^3
    kurtosis <- mean((x - m)^4) / s^4

    JB <- n / 6 * (skewness^2 + (kurtosis - 3)^2 / 4)
    p_value <- 1 - pchisq(JB, df = 2)

    structure(
        list(
            statistic = JB,
            p.value = p_value,
            method = "Jarque-Bera Test (Manual)",
            parameter = 2
        ),
        class = "htest"
    )
}

```

```
    )  
}
```

3. Tahap Penerapan UI

1. Header dan Sidebar Header merupakan judul di atas aplikasi, sementara Sidebar adalah menu navigasi yang terletak di samping kiri.

```
ui <- dashboardPage(  
  
  # HEADER  
  dashboardHeader(  
    title = tags$span(" Normality Lab", class = "title-font-header"),  
    tags$li(class = "dropdown", style = "padding: 8px; color: #444;",  
    )  
  ),  
  
  # SIDEBAR  
  dashboardSidebar(  
    tags$head(  
      tags$style(HTML(paste0("/* Warna Sidebar Baru */  
      .main-sidebar { background-color: ", SECONDARY_COLOR, " !important; }  
      .sidebar-menu li.active a { border-left: 5px solid ", PRIMARY_COLOR, " !important; }  
      .logo { background-color: ", SECONDARY_COLOR, " !important; }  
  
      /* Estetika Font dan Box */  
      .content-wrapper, .right-side { background: #F0FDF5 !important; }  
      .box { border-radius: 10px; box-shadow: 0 5px 15px rgba(0,0,0,0.08); }  
      .box-header h3.box-title { color: ", PRIMARY_COLOR, "; font-weight: 600; }  
  
      .tab-title-accent { color: ", SECONDARY_COLOR, "; font-family: 'Playfair Display'; font-  
  
      /* Style untuk Box 'Tentang Aplikasi' */  
      #about-box { min-height: 250px; padding: 25px; position: relative; overflow: hidden; }  
      #about-box .box-header { margin-bottom: 15px; }  
  
      /* Teks lebih besar di About Box */  
      .about-text-large { font-size: 1.15em; line-height: 1.6; }  
  
      /* Hiasan Latar Belakang Home Lebih Ramai */  
      .home-icon-bg { font-size: 150px; color: ", PRIMARY_COLOR, "; opacity: 0.05; position: absolute; left: 0; top: 0; }  
      .home-icon-bg-2 { font-size: 80px; color: ", ACCENT_COLOR, "; opacity: 0.08; position: absolute; left: 0; top: 0; }  
      .home-icon-bg-3 { font-size: 90px; color: ", SECONDARY_COLOR, "; opacity: 0.05; position: absolute; left: 0; top: 0; }  
    ")))  
  ),  
  sidebarMenu(id = "tabs",  
    menuItem("Home", tabName = "home"),  
    menuItem("Unggah & Deskripsi Data", tabName = "desc"),  
    menuItem("Visualisasi Distribusi", tabName = "visual"),  
    menuItem("Uji Formal Normalitas", tabName = "formal"),  
    menuItem("Skewness & Kurtosis", tabName = "skk"),  
  )
```

```

        menuItem("Analisis Grouping", tabName = "groups"),
        menuItem("Kesimpulan Final", tabName = "final")
    )
),

#2. DashboardBody Dashboard Body merupakan isi dari setiap tab yang ada di dalam sidebar. Baris kode ini
dashboardBody(
    theme = light_theme,
    tabItems(
        # Beranda: pengantar aplikasi & tombol mulai analisis
        tabItem("home",
            tags$h2(" Normality Lab", class="tab-title-accent"),
            box(
                id = "about-box",
                width = 12,
                title = "Tentang Aplikasi",
                tags$i(class = "home-icon-bg fas fa-chart-bar"),
                tags$i(class = "home-icon-bg-2 fas fa-flask"),
                tags$i(class = "home-icon-bg-3 fas fa-balance-scale"),
                tags$div(class = "about-text-large",
                    tags$p(
                        "Normality Lab adalah aplikasi Shiny untuk mengevaluasi asumsi ",
                        strong("distribusi normal"),
                        " menggunakan pendekatan visual, statistik deskriptif, dan uji formal. Tujuan "
                    ),
                    tags$ul(
                        tags$li("Menyesuaikan uji dengan ukuran sampel (n < 30: Shapiro-Wilk; 30 < n < 500: Kolmogorov-Smirnov; n > 500: Anderson-Darling)"),
                        tags$li("Menampilkan interpretasi statistik & visual secara komprehensif."),
                        tags$li("Dirancang untuk analisis akademik & praktis.")
                    )
                ),
                tags$div(style = "height: 40px;")
            ),
            fluidRow(
                box(width = 12, actionButton("go_desc", " Mulai Analisis: Unggah & Deskripsi Data", class="button-primary"))
            )
        ),
        # Unggah data, pilih variabel, dan statistik deskriptif awal
        tabItem("desc",
            tags$h2("Unggah & Deskripsi Data", class="tab-title-accent"),
            fluidRow(
                box(
                    title = tags$span(icon("upload"), " Unggah dan Pilih Variabel"),
                    width = 6,
                    fileInput("file_data", "Pilih file CSV / Excel", accept = c(".csv", ".xlsx", ".xls")),
                    uiOutput("select_variable"),

```

```

        uiOutput("select_group")
    ),

    box(
        title = tags$span(icon("sliders-h"), " Parameter Analisis"),
        width = 6,
        sliderInput("alpha", "Tingkat Signifikansi (Alpha)", 0.01, 0.1, 0.05, step = 0.005),
        tags$p("Alpha digunakan sebagai batas keputusan pada uji formal normalitas.")
    )
),

hr(),

#Pintasan
fluidRow(
    box(width = 6,
        actionButton("go_visual", " Visualisasi Distribusi",
                     class = "btn-success", width = "100%")),
    box(width = 6,
        actionButton("go_formal", "Uji Formal Normalitas",
                     class = "btn-success", width = "100%"))
),

hr(),

fluidRow(
    box(
        title=tags$span(icon("table"), " Preview Data (10 Baris Pertama)"),
        width = 8,
        # Fitur scroll horizontal
        div(style = "overflow-x: auto;",
            tableOutput("data_preview"))
    )
),
    box(
        title=tags$span(icon("calculator"), " Statistik Deskriptif"),
        width = 4,
        tableOutput("summary_stats"),
        uiOutput("centrality_note")
    )
)
),

# Visualisasi distribusi: histogram, density, QQ, ECDF

tabItem("visual",
    tags$h2("Visualisasi Distribusi", class="tab-title-accent"),
    fluidRow(
        box(title=tags$span(icon("sliders-h"), " Opsi Plot"),
            width = 12,
            sliderInput("bins","Jumlah Bins Histogram",10,100,30),
            checkboxInput("show_density","Tampilkan Density Plot di Histogram",FALSE),
            checkboxInput("overlay_normal","Overlay Kurva Normal",FALSE)
    )
)
)
)
```

```

        )
),
fluidRow(
  box(title=tags$span(icon("chart-bar"), " Histogram"),
       width = 6, plotlyOutput("hist_plot", height = "350px")),
  box(title=tags$span(icon("chart-line"), " Density Plot"),
       width = 6, plotlyOutput("density_plot", height = "350px"))
),
fluidRow(
  box(title=tags$span(icon("grip-lines"), " Q-Q Plot"),
       width = 6, plotlyOutput("qq_plot", height = "350px")),
  box(title=tags$span(icon("chart-area"), " ECDF vs Normal CDF"),
       width = 6, plotlyOutput("ecdf_plot", height = "350px"))
),
fluidRow(
  box(width = 12, tags$p(class="text-muted", "Next With Sidebar"))
)
),

# Uji formal normalitas & interpretasi statistik

tabItem("formal",
  tags$h2("Uji Formal Normalitas", class="tab-title-accent"),
  uiOutput("sample_info"),
  hr(),
  fluidRow(
    box(title = tags$span(icon("vial"), " Pengaturan Uji"),
         width = 4, tags$i(class="icon-decor fas fa-flask d1"),
         uiOutput("test_selector"),
         tags$p("H0: Data Populasi mengikuti distribusi normal."),
         tags$p("H1: Data Populasi tidak mengikuti distribusi normal."),
         uiOutput("alpha_display")),
    box(title = tags$span(icon("info-circle"), " Ringkasan Data & Rekomendasi"),
         width = 8, tags$i(class="icon-decor fas fa-lightbulb d2"), uiOutput("test_data_summary")),
  ),
  fluidRow(
    box(title = tags$span(icon("table"), " Hasil Uji Statistik"),
         width = 12,
         div(style="overflow-x: auto;", tableOutput("test_results_table")),
         uiOutput("test_interpretation"))
  )
),

# Analisis skewness-kurtosis dan jarak dari normal
tabItem("skk",
  tags$h2("Analisis Skewness dan Kurtosis", class="tab-title-accent"),
  fluidRow(
    box(
      title = tags$span(icon("chart-scatter"), " Plot Skewness vs Kurtosis"),
      width = 8, tags$i(class="icon-decor fas fa-dot-circle d1"),
      tags$p("Normalitas didekati saat Skewness mendekati 0 dan Kurtosis mendekati 3 (Kurtosis"),
      plotlyOutput("sk_kurt_plot", height = "400px")

```

```

),
box(
  title = tags$span(icon("tag"), " Nilai Metrik"),
  width = 4, tags$i(class="icon-decor fas fa-calculator d2"),
  uiOutput("skk_notes"),
  hr(),
  tags$b("Jarak dari Normal (0, 0):"),
  textOutput("sk_kurt_distance")
)
)
),

# Perbandingan normalitas berdasarkan variabel grouping
tabItem("groups",
  tags$h2("Analisis Normalitas Berdasarkan Grouping", class="tab-title-accent"),
  tags$p("Analisis ini membandingkan distribusi variabel yang dipilih di antara kategori-kategori"),
  fluidRow(
    box(title=tags$span(icon("chart-bar"), " Histogram per Group"),
        width = 6, plotlyOutput("hist_groups")),
    box(title=tags$span(icon("grip-lines"), " Q-Q Plot per Group"),
        width = 6, plotlyOutput("qq_groups"))
  )
),
),

# Kesimpulan akhir dan opsi unduh laporan/data

tabItem("final",
  tags$h2("Kesimpulan Akhir Normalitas", class="tab-title-accent"),
  fluidRow(
    box(title=tags$span(icon("comments"), " Kesimpulan"),
        width = 10,
        uiOutput("final_conclusion"))
  ),
  fluidRow(
    box(title=tags$span(icon("download"), " Unduh"),
        width = 6,
        downloadButton("download_report", "Unduh Laporan HTML Sederhana"),
        downloadButton("download_data", "Unduh Data Mentah (.csv)"))
  )
)
)
))

```

4. Tahap Penerapan Pengolahan Data

Tahap pengolahan data bertujuan untuk memastikan data yang diunggah bisa dianalisis dalam kondisi yang layak secara statistik (data harus numerik). Data yang diunggah pengguna terlebih dahulu diproses dengan tahapan berikut:

```

server <- function(input, output, session) {
  #Navigasi Tombol
  observeEvent(input$go_desc, {

```

```

        updateTabItems(session, "tabs", "desc")
    })

observeEvent(input$go_visual, {
    updateTabItems(session, "tabs", "visual")
})

observeEvent(input$go_formal, {
    updateTabItems(session, "tabs", "formal")
})
#Fungsi Load Data
raw_data <- reactive({
    req(input$file_data)

    ext <- tools::file_ext(input$file_data$name)
    df <- tryCatch({
        if (tolower(ext) == "csv") {
            read.csv(input$file_data$datapath, stringsAsFactors = FALSE)
        } else if (tolower(ext) %in% c("xlsx", "xls")) {
            readxl::read_excel(input$file_data$datapath)
        } else {
            NULL
        }
    }, error = function(e) {
        showNotification(paste("Error membaca file:", e$message), type = "error")
        NULL
    })
}

req(!is.null(df))
return(as.data.frame(df))
})

#Memilih Variabel yang ada di dokumen
output$select_variable <- renderUI({
    df <- raw_data()
    nums <- names(df)[sapply(df, is.numeric)]
    if (length(nums) == 0) {
        return(tags$em("Tidak ada variabel numerik di dataset."))
    }
    selectInput("var", "Pilih variabel numerik", nums)
})

#Memilih Variabel Grouping Kualitatif untuk analisis distribusi per kelompok
output$select_group <- renderUI({
    df <- raw_data()
    if (is.null(df)) return(NULL)
    cats <- names(df)[sapply(df, function(x) is.factor(x) || is.character(x))]
    if (length(cats) == 0) return(NULL)
    selectInput("group_var", "Pilih variabel grouping (opsional)", choices = c("None", cats))
})
#Memilih alpha
output$alpha_display <- renderUI({
    req(input$alpha)
})

```

```

tags$em(paste0("Batas penolakan (Alpha): ", input$alpha))
})

#Input data yang ada di dokumen
selected_data <- reactive({
  req(input$var)
  df <- raw_data()
  req(!is.null(df))

  x <- df[[input$var]]
  x <- as.numeric(x)
  x <- x[!is.na(x)]

  req(length(x) >= 3)
  x
})

#Jumlah Sampel
sample_size <- reactive({
  length(selected_data())
})

#Mengambil Preview data di df sebanyak 10 baris
output$data_preview <- renderTable({
  df <- raw_data(); req(df)
  head(df, 10)
})

#Rekomendasi Uji di Tab Formal Sesuai dengan sampel kecil, sampel besar, Sampel sangat besar
test_recommendations <- reactive({
  x <- selected_data(); req(x)
  N <- length(x)

  if (N < 3) {
    return(list(
      status = "Data Terlalu Sedikit",
      recommendation = "Normalitas tidak dapat diuji secara statistik. Min. n=3 untuk Shapiro-Wilk.",
      icon = "fas fa-exclamation-triangle"
    ))
  } else if (N < 30) {
    return(list(
      status = "Sampel Kecil (n < 30)",
      recommendation = "Uji Paling Kuat: Shapiro-Wilk dan Lilliefors. Visualisasikan dengan Q-Q Plot.",
      icon = "fas fa-leaf"
    ))
  } else if (N >= 30 && N <= 100) {
    return(list(
      status = "Sampel Besar (30 < n <= 100)",
      recommendation = "Uji yang Direkomendasikan: Shapiro-Wilk (masih baik), Lilliefors/Kolmogorov-Smirnov",
      icon = "fas fa-balance-scale"
    ))
  } else if (N > 100) {
    return(list(
      status = "Sampel sangat besar (n > 100)",
      recommendation = "Uji yang Direkomendasikan: Jarque-Bera atau Chi-square Goodness-of-Fit. Waspada Outlier!",
      icon = "fas fa-city"
    ))
}
})

```

```

        }
    })
#Uji yang terfilter antara sampel besar (n>30) dan sampel kecil (n<= 30)
output$test_selector <- renderUI({
  n <- sample_size()
  req(n)

  if (n < 30) {
    choices <- c(
      "Shapiro-Wilk" = "shapiro",
      "Lilliefors" = "lillie"
    )
  } else {
    choices <- c(
      "Jarque-Bera" = "jb",
      "Kolmogorov-Smirnov" = "ks",
      "Chi-square GOF" = "chi"
    )
  }
}

selectInput(
  "selected_test",
  "Pilih Uji Normalitas",
  choices = choices
)
})
})
}

```

5. Tahap Penerapan Analisis Statistik

Analisis statistik dimulai dengan perhitungan statistik deskriptif yang mencakup nilai mean, median, standar deviasi, skewness, dan kurtosis. Statistik ini digunakan sebagai indikasi awal bentuk distribusi data.

Selanjutnya, aplikasi menerapkan uji normalitas formal yang dipilih secara adaptif berdasarkan ukuran sampel, antara lain:

- Shapiro-Wilk Test untuk sampel kecil
- Lilliefors Test sebagai pelengkap
- Jarque-Bera Test untuk sampel menengah hingga besar
- Kolmogorov-Smirnov Test sebagai pembanding

Pendekatan ini bertujuan untuk menghindari penggunaan uji yang tidak sesuai dengan karakteristik data.

```

#Ringkasan Statistik Mean, Median, SD, CV, Skewness, Kurtosis, IQR, Min, Max
output$summary_stats <- renderTable({
  x <- selected_data(); req(x)
  data.frame(
    Statistik = c("n (Jumlah Data)", "Mean", "Median", "SD", "CV", "Skewness", "Kurtosis (Excess)", "IQR", "Min", "Max"),
    Nilai = c(
      length(x), # Menambahkan N
      round(mean(x), 6),
      round(sd(x), 6),
      round(skew(x), 6),
      round(kurt(x), 6),
      round(quantile(x), 6),
      min(x),
      max(x)
    )
  )
})

```

```

        round(median(x), 6),
        round(sd(x), 6),
        round(ifelse(mean(x) != 0, sd(x)/mean(x), NA), 6),
        round(skewness(x), 6),
        round(kurtosis(x) - 3, 6), # Menggunakan Kurtosis Excess (Fisher)
        round(min(x), 6),
        round(max(x), 6),
        round(IQR(x), 6)
    )
)
),
),
rownames = FALSE)

#Blok Kode Untuk Memilih Uji Formal
run_tests <- reactive({
  x <- selected_data()
  req(x, input$selected_test)

  test <- input$selected_test

  if (test == "shapiro") {
    return(shapiro.test(x))
  }

  if (test == "lillie") {
    return(lillie.test(x))
  }

  if (test == "jb") {
    res <- jarque_bera_manual(x)
    req(res)
    return(res)
  }

  if (test == "ks") {
    return(ks.test(x, "pnorm", mean(x), sd(x)))
  }

  if (test == "chi") {
    return(chi_sq_test(x))
  }

  NULL
})

#Ringkasan Data dan Rekomendasi Uji pada uji formal agar uji yang dipakai sesuai dengan karakteristik
output$test_data_summary <- renderUI({
  x <- selected_data(); req(x)
  N <- length(x)
  rec <- test_recommendations()

  HTML(paste0(
    "<div style='background-color: #ECF0F1; padding: 15px; border-radius: 8px; border-left: 5px solid"
  ))
})

```

```

    "<h4><i class=''", rec$icon, "'></i> Status Data & Rekomendasi</h4>",
    "<p style='margin-bottom: 5px;'>Jumlah Data (n): <span style='font-size:1.2em; font-weight:700;'>" ,
    "<p style='margin-bottom: 10px;'>Kelompok Sampel: <span style='font-weight:600;'>", rec$status, "</span>",
    "Rekomendasi Uji: ", rec$recommendation,
    "</div>"
  ))
})

#Hasil Dari Uji Formal yang dipilih menggunakan p-value
output$test_results_table <- renderTable({
  res <- run_tests()
  req(res)

  if (!is.null(res$error)) {
    return(data.frame(
      Uji = res$method,
      Statistik = NA,
      P_Value = NA,
      Keputusan = res$error
    ))
  }

  keputusan <- ifelse(
    res$p.value > input$alpha,
    "NORMAL (Gagal Tolak H0)",
    "TIDAK NORMAL (Tolak H0)"
  )

  data.frame(
    Uji = res$method,
    Statistik = round(as.numeric(res$statistic)[1], 6),
    P_Value = formatC(res$p.value, format = "f", digits = 5),
    Keputusan = ifelse(
      res$p.value > input$alpha,
      "NORMAL (Gagal Tolak H0)",
      "TIDAK NORMAL (Tolak H0)"
    )
  )
}, rownames = FALSE)

#Menampilkan catatan kurtosis dan catatan skewness
output$skk_notes <- renderUI({
  x <- selected_data(); req(x)
  sk <- round(skewness(x), 4)
  kt_fish <- round(kurtosis(x) - 3, 4)
  HTML(paste0("Skewness: <b>", sk, "</b><br>Kurtosis Fisher: <b>", kt_fish, "</b>"))
})

#Menghitung Jarak antara Skewness dan Kurtosis Data terhadap Normal
output$sk_kurt_distance <- renderText({
  x <- selected_data(); req(x)
  round(sqrt(skewness(x)^2 + (kurtosis(x)-3)^2), 6)
})

```

```
})
```

6. Tahap Visualisasi

Visualisasi digunakan sebagai alat eksploratif untuk mendukung hasil yang diperoleh dari uji formal. Aplikasi menyediakan beberapa bentuk visualisasi, yaitu:

- Histogram dan density plot untuk melihat bentuk distribusi data
- Q–Q Plot untuk membandingkan distribusi data dengan distribusi normal teoretis
- Plot skewness dan kurtosis untuk mengevaluasi kedekatan data terhadap distribusi normal
- ECDF vs CDF Normal untuk mengevaluasi distribusi data kumulatif dengan distribusi normal kumulatif

```
#Membuat Histogram
output$hist_plot <- renderPlotly({
  x <- selected_data(); req(x, input$bins)

  p <- ggplot(data.frame(x = x), aes(x = x)) +
    geom_histogram(aes(y = after_stat(density)), bins = input$bins,
                   fill = "#00A388", color = "#2F4858", alpha = .7) +
    theme_minimal() +
    labs(title = paste("Histogram & Density dari", input$var), x = input$var, y = "Density")

  if (isTRUE(input$show_density)) {
    p <- p + geom_density(alpha = .3, color="#FFB600", linewidth=1.2)
  }
  if (isTRUE(input$overlay_normal)) {
    p <- p + stat_function(fun = dnorm, args = list(mean(x), sd(x)),
                           color = "#E74C3C", linewidth = 1, linetype="dashed")
  }
  ggplotly(p) %>% config(displayModeBar = FALSE)
})

#Membuat Grafik Fungsi pdf
output$density_plot <- renderPlotly({
  x <- selected_data(); req(x)

  p <- ggplot(data.frame(x = x), aes(x = x)) +
    geom_density(fill = "#00A388", alpha = .35, color="#00A388") +
    theme_minimal() +
    labs(title = paste("Density Plot -", input$var), x = input$var, y = "Density")
  if (isTRUE(input$overlay_normal)) {
    p <- p + stat_function(fun = dnorm, args = list(mean(x), sd(x)), color = "#E74C3C", linewidth=1,
                           linetype="dashed")
  }
  ggplotly(p) %>% config(displayModeBar = FALSE)
})

#Membuat Grafik qq plot
output$qq_plot <- renderPlotly({
  x <- selected_data();
  req(x)

  df_plot <- data.frame(Sample = x)
```

```

p <- ggplot(df_plot, aes(sample = Sample)) +
  stat_qq(color = "#2F4858", size=2) +
  stat_qq_line(color = "#E74C3C", linewidth=1) +
  theme_minimal() +
  labs(title = paste("Q-Q Plot -", input$var), x = "Theoretical Quantiles", y = "Sample Quantiles")

ggplotly(p) %>% config(displayModeBar = FALSE)
})

#Membuat Grafik kumulatif empiris dan Normal
output$ecdf_plot <- renderPlotly({
  x <- selected_data(); req(x)
  ec <- ecdf(x)
  xs <- seq(min(x), max(x), length.out = 200)
  df_plot <- data.frame(x = xs, ECDF = ec(xs), Normal = pnorm(xs, mean(x), sd(x)))
  p <- ggplot(df_plot) +
    geom_line(aes(x, ECDF, text=paste("ECDF Data:", round(ECDF,4))), size = 1, color="#00A388") +
    geom_line(aes(x, Normal, text=paste("Normal CDF:", round(Normal,4))), size = 1, linetype = "dashed",
    theme_minimal() +
    labs(title = paste("ECDF vs Normal CDF -", input$var), x = input$var, y = "Probability")
  ggplotly(p, tooltip = c("x", "ECDF", "Normal")) %>% config(displayModeBar = FALSE)
})

#Membuat Grafik Kurtosis dan Skewness di Koordinat Kartesius
output$sk_kurt_plot <- renderPlotly({
  x <- selected_data(); req(x)
  sk <- skewness(x); kt_fish <- kurtosis(x) - 3
  df_skk <- data.frame(sk = sk, kt = kt_fish)

  p <- ggplot(df_skk, aes(x=sk, y=kt)) +
    geom_point(aes(text = paste("Skew:", round(sk, 3), "<br>Kurtosis:", round(kt, 3))),
    size = 6, color = "#E74C3C") +
    geom_point(x = 0, y = 0, color = "#00A388", size = 8, shape = 4, stroke = 2) +
    annotate("text", x = 0, y = 0.5, label = "Normal", color = "#00A388", size = 4, fontface = "bold") +
    geom_hline(yintercept = 0, linetype = "dashed", color = "#AAB7B8") +
    geom_vline(xintercept = 0, linetype = "dashed", color = "#AAB7B8") +
    labs(title = "Plot Skewness vs Kurtosis (Kurtosis Fisher)", x = "Skewness", y = "Kurtosis Fisher")
  theme_minimal()

  ggplotly(p, tooltip = "text") %>% config(displayModeBar = FALSE)
})

#=====ANALISIS Grouping=====

#Membuat Bar Histogram Per Grup
output$hist_groups <- renderPlotly({
  req(input$group_var, input$group_var != "None")
  df <- raw_data(); req(df)

  p <- ggplot(df, aes_string(x = input$var, fill = input$group_var)) +
    geom_histogram(position = "identity", alpha = .6, bins = input$bins) +
    labs(title=paste("Histogram Grouped by", input$group_var)) +
    theme_minimal()
})

```

```

ggplotly(p) %>% config(displayModeBar = FALSE)
})

#Membuat qq plot per group
output$qq_groups <- renderPlotly({
  req(input$group_var, input$group_var != "None")
  df <- raw_data(); req(df)

  p <- ggplot(df, aes(sample = .data[[input$var]], color = .data[[input$group_var]])) +
    stat_qq() + stat_qq_line() +
    facet_wrap(as.formula(paste("~", input$group_var))) +
    labs(title= paste("Q-Q Plot Grouped by", input$group_var)) +
    theme_minimal()

  ggplotly(p) %>% config(displayModeBar = FALSE)
})

```

7. Tahap Interpretasi

Pada tahap interpretasi, hasil visualisasi dan uji formal dianalisis secara bersama-sama. Aplikasi tidak langsung menyimpulkan normalitas hanya berdasarkan satu uji statistik, melainkan mempertimbangkan keseluruhan informasi yang tersedia.

Interpretasi hasil disajikan dalam bentuk poin-poin sederhana yang menjelaskan apakah data dapat dianggap berdistribusi normal atau tidak, serta faktor-faktor pendukung keputusan tersebut.

```

#Interpretasi statistik deskriptif skewness, kurtosis, dan distribusi
output$centrality_note <- renderUI({
  x <- selected_data(); req(x)
  # Ambang batas 10% dari SD sebagai indikator signifikan perbedaan Mean-Median
  threshold <- 0.1 * sd(x)
  sk <- skewness(x)
  kt_ex <- kurtosis(x) - 3

  # Interpretasi Simetri (Mean vs Median)
  if (abs(mean(x) - median(x)) < threshold) {
    simetri_msg <- "<span style='color:#00A388;'>Mean mendekati Median → Distribusi Simetris</span>"
  } else if (mean(x) > median(x)) {
    simetri_msg <- "<span style='color:#E74C3C;'>Mean > Median → Indikasi Skewness Positif</span>"
  } else {
    simetri_msg <- "<span style='color:#E74C3C;'>Mean < Median → Indikasi Skewness Negatif</span>"
  }

  # Interpretasi Skewness dan Kurtosis (Rules of Thumb)
  sk_msg <- if(abs(sk)==0){
    "Simetri"
  }
  else if (abs(sk) < 0.5) {
    "rendah (mendekati simetri)"
  } else if (abs(sk) < 1) {
    "sedang"
  } else {
    "Kuat"
  }
})

```

```

    }
  kt_msg <- if (abs(kt_ex) < 0.5) {
    "Kurtosis mendekati normal (mesokurtik)."
  } else if (kt_ex >= 0.5) {
    "Distribusi leptokurtik (puncak runcing, ekor berat)."
  } else {
    "Distribusi platykurtik (puncak datar, ekor ringan)."
  }
# Gabungan
HTML(paste0(
  "<div style='background-color: #F7F7F7; padding: 10px; border-radius: 5px; border-left: 3px solid "
  "<h4>Kesimpulan Deskriptif</h4>",
  "<ul>",
  "<li>", simetri_msg, "</li>",
  "<li>", sk_msg, "</li>",
  "<li>", kt_msg, "</li>",
  "</ul>",
  "</div>"
))
})
#Interpretasi Hasil Uji Formal yang dipilih
output$test_interpretation <- renderUI({
  res <- run_tests()
  req(res)

  alpha <- input$alpha
  normal <- res$p.value > alpha
  color <- ifelse(normal, "#00A388", "#E74C3C")

  HTML(paste0(
    "<div style='border-left:5px solid ", color, "; padding:12px;'>",
    "<h4>Kesimpulan Uji</h4>",
    "<p><b>", res$method, "</b></p>",
    "<p>Keputusan: <b style='color:", color, ";'>",
    ifelse(normal, "NORMAL", "TIDAK NORMAL"),
    "</b></p>",
    "</div>"
  ))
})

#Interpretasi Visual untuk kesimpulan final
interpret_visual_summary <- function(x) {

  if (length(x) < 3 || all(is.na(x))) {
    return(list(
      sk_msg = "Data belum cukup untuk interpretasi bentuk distribusi.",
      kt_msg = ""
    ))
  }

  sk <- moments::skewness(x, na.rm = TRUE)
  kt <- moments::kurtosis(x, na.rm = TRUE) - 3
}

```

```

sk_msg <- if (!is.na(sk) && abs(sk) <= 0.5) {
  "Distribusi relatif simetris."
} else if (!is.na(sk) && sk > 0.5) {
  "Distribusi condong ke kanan (positive skew)."
} else {
  "Distribusi condong ke kiri (negative skew)."
}

kt_msg <- if (!is.na(kt) && abs(kt) <= 0.5) {
  "Keruncingan mendekati distribusi normal."
} else if (!is.na(kt) && kt > 0.5) {
  "Distribusi lebih runcing dari normal (leptokurtic)."
} else {
  "Distribusi lebih datar dari normal (platykurtic)."
}

list(
  sk = sk,
  kt = kt,
  sk_msg = sk_msg,
  kt_msg = kt_msg
)
}

#Hasil Final Menggunakan Uji Shapiro Wilks ketika sampel kecil dan Uji JB ketika Sampel besar
final_test_result <- reactive({
  x <- selected_data()
  req(x, input$selected_test)

  test_res <- run_tests()
  req(test_res)

  n <- length(x)
  test_name <- test_res$method
  pval <- test_res$p.value

  decision <- ifelse(
    pval > input$alpha,
    "GAGAL_TOLAK_H0",
    "TOLAK_H0"
  )

  # ===== VALIDITAS UJI =====
  valid <- TRUE
  warning <- NULL

  if (grepl("Jarque", test_name) && n > 200) {
    warning <- "Uji Jarque-Bera sangat sensitif pada ukuran sampel besar. Penyimpangan kecil dari norma"
  }

  if (grepl("Kolmogorov", test_name)) {
    warning <- "Uji Kolmogorov-Smirnov kurang ideal jika parameter distribusi (mean & SD) diestimasi dengan metode yang tidak benar"
  }
})

```

```

list(
  n = n,
  test_name = test_name,
  p_value = pval,
  decision = decision,
  valid = is.null(warning),
  warning = warning
)
})

#Keputusan Final Menggunakan Sapiro Wilks ketika sampel kecil dan Jarque bera ketika sampel besar
output$final_conclusion <- renderUI({
  x <- selected_data(); req(x)
  res <- final_test_result(); req(res)
  vis <- interpret_visual_summary(x)

  normal <- res$decision == "GAGAL_TOLAK_H0"
  color <- ifelse(normal, "#00A388", "#E74C3C")

  HTML(paste0(
    "<div style='padding:15px; background:#F9FBFC; border-radius:8px;
      border-left:5px solid ", color, ";'>",

    "<h4> Kesimpulan Statistik</h4>",
    "<p><b>Uji yang digunakan:</b> ", res$test_name, "</p>",
    "<p><b>P-value:</b> ", round(res$p_value, 5), "</p>",
    "<p><b>Keputusan:</b> ",
    ifelse(normal,
      "<b style='color:#00A388;'>NORMAL (Gagal Tolak H0)</b>",
      "<b style='color:#E74C3C;'>TIDAK NORMAL (Tolak H0)</b>"
    ),
    "</p>",

    if (!is.null(res$warning)) {
      paste0(
        "<div style='margin-top:10px; color:#856404;
          background:#fff3cd; padding:10px;
          border-radius:5px;'>",
        "! ", res$warning,
        "</div>"
      )
    } else "",

    "<h4 style='margin-top:15px;'> Interpretasi Visual</h4>",
    "<ul>",
    "<li>", vis$sk_msg, "</li>",
    "<li>", vis$kt_msg, "</li>",
    "</ul>",

    "<p style='font-size:12px; color:#6c757d;'>
    Catatan: Uji formal sangat sensitif pada ukuran sampel besar.
    Visualisasi digunakan sebagai pendukung interpretasi.
    </p>",
  ))
})

```

```
    "</div>"  
))  
})
```

9. Tahap Pembangunan Output Offline(HTML dan CSV)

Fungsi di sini berguna untuk memberikan laporan test secara offline dalam format HTML dengan fungsi di bawah ini pengguna dapat mengunduh file yang berisi uji normalitas dan statistik deskriptifnya. Apabila pengguna kehilangan file awal, pengguna dapat mengunduh lagi file yang masih tersimpan di aplikasi

```
#Fungsi untuk mengunduh Laporan Uji formal dengan JB yang digunakan apabila sampel >30 dan Shapiro wilk  
output$download_report <- downloadHandler(  
  filename = function() {  
    paste0("normality_report_", Sys.Date(), ".html")  
  },  
  content = function(file) {  
  
    x <- selected_data()  
    if(length(x) < 3) {  
      showNotification("Data tidak cukup untuk analisis normalitas.", type = "error")  
      return(NULL)  
    }  
    n <- length(x)  
    alpha <- input$alpha  
  
    stats <- data.frame(  
      Mean = mean(x),  
      Median = median(x),  
      SD = sd(x),  
      Min = min(x),  
      Max = max(x)  
    )  
  
    sh <- li <- jb <- ks <- chisq <- NA  
  
    # Small sample (n <= 30)  
    if (n <= 30) {  
      sh <- shapiro.test(x)$p.value  
      li <- if (n >= 6) lillie.test(x)$p.value else NA  
    }  
  
    # Medium sample (30 < n <= 100)  
    if (n > 30 && n <= 100) {  
      jb <- tryCatch(  
        jarque_bera_manual(x)$p.value,  
        error = function(e) NA  
      )  
  
      ks <- tryCatch(  
        ks.test(x, "pnorm", mean(x), sd(x))$p.value,  
        error = function(e) NA  
      )  
    }  
  })
```

```

chisq <- tryCatch({
  bins <- floor(sqrt(n))
  observed <- table(cut(x, bins))
  expected <- rep(length(x) / length(observed), length(observed))
  chisq.test(observed, p = expected / sum(expected))$p.value
}, error = function(e) NA
}

# Large sample (n > 100)
if (n > 100) {
  jb <- tryCatch(
    jarque_bera_manual(x)$p.value,
    error = function(e) NA
  )

  bins <- floor(sqrt(n))
  chisq <- tryCatch({
    observed <- table(cut(x, bins))
    expected <- rep(length(x) / length(observed), length(observed))
    chisq.test(observed, p = expected / sum(expected))$p.value
  }, error = function(e) NA
}
}

fmt_p <- function(p) {
  if (is.na(p)) "NA"
  else round(p, 5)
}

keputusan <- function(p) {
  if (is.na(p)) "N/A"
  else if (p > alpha) "NORMAL (Gagal Tolak H0)"
  else "TIDAK NORMAL (Tolak H0)"
}

tests_html <- ""

# n <= 30
if (n <= 30) {
  tests_html <- paste0(
    "<ul>",
    "<li>Shapiro-Wilk p-value: ", fmt_p(sh), " → <b>", keputusan(sh), "</b></li>",
    "<li>Lilliefors p-value: ", fmt_p(li), " → <b>", keputusan(li), "</b></li>",
    "</ul>"
  )
}

# 30 < n <= 100
if (n > 30 && n <= 100) {
  tests_html <- paste0(
    "<ul>",
    "<li>Jarque-Bera p-value: ", fmt_p(jb), " → <b>", keputusan(jb), "</b></li>",
    "<li>Kolmogorov-Smirnov p-value: ", fmt_p(ks), " → <b>", keputusan(ks), "</b></li>",
    "<li>Chi-square GoF p-value: ", fmt_p(chisq), " → <b>", keputusan(chisq), "</b></li>",
    "</ul>"
  )
}

```

```

        )
    }

# n > 100
if (n > 100) {
  tests_html <- paste0(
    "<ul>",
    "<li>Jarque-Bera p-value: ", fmt_p(jb), " → <b>", keputusan(jb), "</b></li>",
    "<li>Chi-square GoF p-value: ", fmt_p(chisq), " → <b>", keputusan(chisq), "</b></li>",
    "</ul>"
  )
}

res <- evaluate_normality(x, alpha)

final_text <- if (res$decision == "GAGAL_TOLAK_H0")
  "Distribusi data mendekati normal."
else if (res$decision == "TOLAK_H0")
  "Distribusi data menunjukkan penyimpangan dari normalitas."
else
  "Data tidak cukup untuk dianalisis."
final_normal <- if(!is.null(res$decision) && res$decision == "GAGAL_TOLAK_H0") TRUE else FALSE

html <- paste0(
  "<!DOCTYPE html>
<html>
<head>
  <meta charset='UTF-8'>
  <title>Normality Analysis Report</title>
  <style>
    body { font-family: Arial; margin: 40px; }
    h2 { color: #00A388; }
    table { border-collapse: collapse; width: 60%; }
    th, td { border: 1px solid #ccc; padding: 8px; text-align:center; }
    th { background-color: #f2f2f2; }
  </style>
</head>
<body>

<h2>Normality Analysis Report</h2>

<p><b>Sample Size (n):</b> ", n, "</p>
<p><b>Alpha:</b> ", alpha, "</p>

<h3>Descriptive Statistics</h3>
<table>
  <tr><th>Mean</th><th>Median</th><th>SD</th><th>Min</th><th>Max</th></tr>
  <tr>
    <td>", round(stats$Mean,4), "</td>
    <td>", round(stats$Median,4), "</td>
    <td>", round(stats$SD,4), "</td>
    <td>", round(stats$Min,4), "</td>
    <td>", round(stats$Max,4), "</td>
  </tr>
</table>

```

```

        </tr>
    </table>

<h3>Normality Tests</h3>
", tests_html, "

<h3>Final Conclusion</h3>
<p>",
    if (final_normal)
        "<b style='color:green;'>Data secara umum BERDISTRIBUSI NORMAL.</b>"
    else
        "<b style='color:red;'>Data menunjukkan indikasi TIDAK NORMAL.</b>",
    "</p>

<hr>
<p><i>Generated by Normality Lab Shiny App</i></p>

</body>
</html>" )
)

writeLines(html, file)
}
)
#Fungsi untuk mengunduh data yang ada
output$download_data <- downloadHandler(
    filename = function() paste0("data_normality_lab_", Sys.Date(), ".csv"),
    content = function(file) {
        df <- raw_data();
        write.csv(df, file, row.names = FALSE)
    }
)

#Penggerak UI dan Server
shinyApp(ui, server)

```

9. Tahap Pengujian dan Perbaikan

Tahap pengujian dilakukan untuk memastikan aplikasi berjalan dengan baik pada berbagai kondisi data, seperti:

- Ukuran sampel kecil dan besar
- Data dengan distribusi normal dan tidak normal
- Data dengan nilai ekstrem atau outlier

Berdasarkan hasil pengujian, dilakukan perbaikan pada validasi input, stabilitas reaktif, serta kejelasan interpretasi hasil agar aplikasi lebih andal dan minim kesalahan. Pada tahap ini kelompok kami memperbaiki logika-logika yang kurang sesuai dan berpotensi menimbulkan error. Untuk tahap yang perlu diperbaiki/sangat sering error adalah : 1. Download report pada tahap ini penulisan logika ditulis ulang untuk menyertakan uji yang ada dengan menggunakan fungsi evaluate_normality untuk mendapat keputusan final 2. Kesimpulan Final memperbaiki kode yang error karena setiap uji memiliki kondisi

sampel masing-masing dan kelemahan masing masing sehingga kami menyertakan *Peringatan* agar bisa dipahami kekuatan dan kelemahan uji formal yang digunakan 3. Perbaikan UI dan tata letak Penambahan seperti pintasan di bagian UI yang sebelumnya tidak ada juga merupakan upgrade dari tahap Pengujian dan Perbaikan

10. Tahap Deployment

Tahap akhir adalah deployment aplikasi sehingga dapat diakses oleh pengguna. Menggunakan shinyapps.io aplikasi mendapatkan alamat web tersendiri