ANALYSING AND PLOTTING EXAM RESULTS FOR COLLEGE CLASS

A PROJECT REPORT

Submitted by

MUKILAN K (2303811724321072)

in partial fulfillment of requirements for the award of the course

AGI1252 - FUNDAMENTALS OF DATA SCIENCE USING R

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

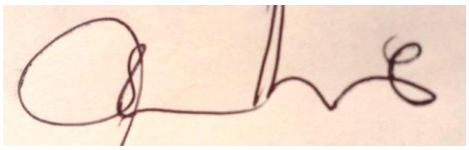
JUNE- 2025

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"ANALYSING AND PLOTTING EXAM RESULTS FOR COLLEGE CLASS"** is the bonafide work of **MUKILAN K (2303811724321072)** who carried out the project work during the academic year 2024 - 2025 under my supervision.



SIGNATURE

Dr.T. AVUDAIAPPAN, M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

PROFESSOR

Department of Artificial Intelligence

K.Ramakrishnan College of Technology (Autonomous)

Samayapuram–621112.

SIGNATURE

Ms.S.Murugavalli., M.E.,(Ph.D).,

SUPERVISOR

ASSISTANT PROFESSOR

Department of Artificial Intelligence

K.Ramakrishnan College of Technology (Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on ….**02.06.2025**……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

I declare that the project report on **"ANALYSING AND PLOTTING EXAM RESULTS FOR COLLEGE CLASS"** is the result of original work done by me and best of my knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF TECHNOLOGY**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **AGI1252 - FUNDAMENTALS OF DATA SCIENCE USING R**

.

Signature

MUKILAN K

Place: Samayapuram

Date:02.06.2025

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

I would like to thank **Dr. N. VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. T. AVUDAIAPPAN, M.E.,Ph.D.,** Head of the department, **ARTIFICIAL INTELLIGENCE** for providing his encourage pursuing this project.

I express our deep expression and sincere gratitude to our project supervisor **Ms.S.Murugavalli., M.E.,(Ph.D).,** Department of **ARTIFICIAL INTELLIGENCE,** for her incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

INSTITUTE

Vision:

- To serve the society by offering top-notch technical education on par with global standards.

Mission:

- Be a center of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all – round personalities respecting moral and ethical values.

DEPARTMENT

Vision:

- To excel in education, innovation, and research in Artificial Intelligence and Data Science to fulfil industrial demands and societal expectations.

Mission

- To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.
- To collaborate with industry and offer top-notch facilities in a conducive learning environment.
- To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.
- To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

PROGRAM EDUCATIONAL OBJECTIVES (PEO)

- **PEO1:** Compete on a global scale for a professional career in Artificial Intelligence and Data Science.
- **PEO2:** Provide industry-specific solutions for the society with effective communication and ethics.
- **PEO3** Enhance their professional skills through research and lifelong learning initiatives.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Capable of finding the important factors in large datasets, simplify the data, and improve predictive model accuracy.
- **PSO2:** Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals, and an engineering specialization to develop solutions to complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.
3. **Design/development of solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required.
4. **Conduct investigations of complex problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions.
5. **Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems.
6. **The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment.

7. **Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws.

8. **Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

9. **Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.

10. **Project management and finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

11. **Life-long learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.

ABSTRACT

The project titled Analysing and Plotting Exam Results for College Class is a web-based application developed using R and the Shiny framework to streamline the evaluation of student academic performance. It allows faculty to upload exam results in CSV format and instantly visualize performance metrics through an interactive dashboard. The system features secure login, a user-friendly UI designed with shinydashboard and bslib, and dynamic plots generated using the plotly package. Core functionalities include calculating pass/fail percentages, subject-wise averages, and grade distributions. Additionally, a prediction module identifies subjects where students need improvement or show strong performance. The application leverages dplyr for data manipulation and reactive programming for real-time updates, making it a practical R-based solution that transforms raw exam data into meaningful educational insights.

ABSTRACT WITH POs AND PSOs MAPPING

CO 5 : BUILD DATA SCIENCE USING R PROGRAMMING FOR SOLVING REAL-TIME PROBLEMS.

ABSTRACT	POs MAPPED	PSOs MAPPED
The project titled Analysing and Plotting Exam Results for College Class is a web-based application developed using R and the Shiny framework to streamline the evaluation of student academic performance. It allows faculty to upload exam results in CSV format and instantly visualize performance metrics through an interactive dashboard. The system features secure login, a user-friendly UI designed with shinydashboard and bslib, and dynamic plots generated using the plotly package. Core functionalities include calculating pass/fail percentages, subject-wise averages, and grade distributions. Additionally, a prediction module identifies subjects where students need improvement or show strong performance. The application leverages dplyr for data manipulation and reactive programming for real-time updates, making it a practical R-based solution that transforms raw exam data into meaningful educational insights.	**PO1 -3** **PO2 -3** **PO4 -2** **PO5 -3** **PO9 -2** **PO10 -3**	**PSO1 -3** **PSO2 -3**

Note: 1- Low, 2-Medium, 3- High

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 Objective

The main objective of the project titled Analysing and Plotting Exam Results for College Class is to develop an intuitive and intelligent performance monitoring system for academic institutions. The system should enable faculty members and administrators to analyze exam results efficiently and make data-driven decisions. By transforming traditional manual result processing into a dynamic, visual, and interactive experience, the platform aims to save time, reduce human error, and enhance the accuracy of student performance assessments. Furthermore, the system is designed to support subject-wise insights and automated recommendations, allowing academic stakeholders to identify weak and strong areas among students, thereby enabling targeted intervention and academic improvement.

1.2 Overview

In many colleges and educational settings, student exam results are often analyzed manually or using static spreadsheets. This process can be slow, labor-intensive, and prone to human error. With increasing student strength and subject diversity, there is a growing need for automated tools that can simplify this analysis and present information in an understandable and actionable form. This project addresses that gap by utilizing R programming and the Shiny framework to develop a web-based dashboard system.

The platform allows users to log in securely, upload student exam data in .csv format, and instantly receive various analytical visualizations including pass/fail percentages, average scores, and grade distributions. It also includes a predictive module to highlight underperforming subjects and areas of academic strength. The use of shinydashboard ensures a professional, responsive layout, while plotly enables interactive plotting.

1.3 Data Science related concepts

This project applies essential data science concepts to evaluate student performance effectively using R programming. It begins with data collection and preprocessing, where exam results are uploaded in CSV format and cleaned using the dplyr package. Irrelevant columns are removed, and subject-specific marks are isolated for analysis.

Using exploratory data analysis (EDA), the system dynamically calculates and updates performance metrics through reactive programming in Shiny. This provides an interactive and real-time environment for users to explore trends and patterns.

The visualization layer is built using the plotly package, allowing the creation of interactive bar and pie charts. These visual tools clearly depict pass/fail rates, grade distributions, and subject-wise averages.

A basic prediction logic is also integrated, identifying strong and weak subjects based on average scores. While simple, this component introduces foundational ideas from predictive analytics, showing the potential of R in educational data science applications.

CHAPTER 2

PROJECT METHODOLOGY

2.1 Proposed Work

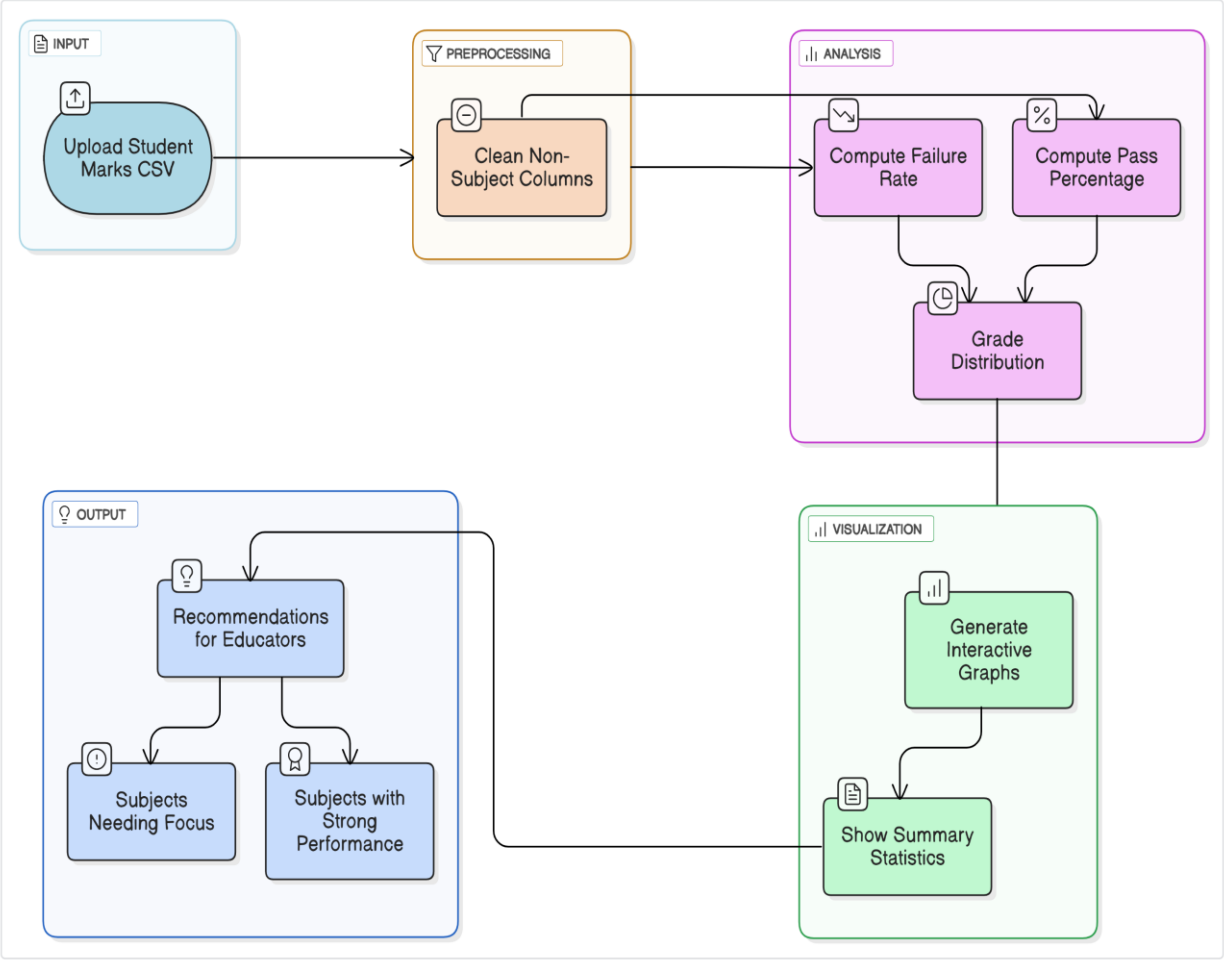
The project titled “Analyzing and Plotting Exam Results for College Class” is implemented using the R programming language and the Shiny web application framework. The main objective is to create an interactive portal that enables institutions to upload, process, and analyze student exam data in real time. The methodology begins with a secure login system that ensures only authorized users can access the portal’s features. Once authenticated, users can upload a CSV file containing student exam marks. The application reads this file, removes any irrelevant columns such as student IDs and total fields, and stores the cleaned data for further analysis.

After data import, the system processes the information to calculate total marks, average scores, and pass or fail status for each student and subject. It computes performance metrics such as subject-wise pass percentages and failure rates. This processed data is then visualized using dynamic graphs including bar plots, pie charts, and interactive dashboards. These visual representations provide insights into individual subject performance, grade distribution across students, and overall class averages.

One of the key features of the application is its ability to perform performance analysis. It identifies subjects where the average marks fall below acceptable levels and highlights them as weak areas needing improvement. Similarly, it detects strong subjects based on higher average scores, thus helping educators focus on strengths and weaknesses. The entire user interface is built using Shiny’s reactive programming capabilities, ensuring that each user action leads to immediate and automatic updates in the visual output. The structured navigation within the dashboard, including tabs for Home, Login, Dashboard, Upload, Prediction, and Logout, enhances the user experience.

Overall, this project demonstrates how data-driven insights can be generated through R programming in a user-friendly web environment, making it easier for educational institutions to monitor and improve student performance effectively.

2.2 Block Diagram



CHAPTER 3

MODULE DESCRIPTION

3.1 Data Import Module

Purpose:

The Data Import Module is responsible for uploading and reading the student exam results data from a CSV file. It ensures that raw exam data is correctly loaded and ready for further processing and analysis.

Description:

This module handles the import of exam result data, allowing users to upload CSV files containing students' marks across various subjects. Using R's read.csv() function, the data is imported and then cleaned with the help of the dplyr package. Non-subject columns such as student identifiers and totals are removed to focus solely on subject-wise performance data. The cleaned data is stored reactively for seamless use by other modules in the application. The module also performs basic validation checks to confirm the data structure before processing. Additionally, this module ensures that data inconsistencies, such as missing values or formatting errors, are minimized to improve downstream analysis quality. This foundational step ensures that all subsequent operations work on reliable and structured data.

Key Functions:

1. Upload CSV File: Allows users to upload student exam results in CSV format.
2. Data Cleaning: Removes unnecessary columns and retains subject-wise marks.
3. Data Validation: Checks file format and structure for consistency.

3.2 Data Processing Module

Purpose:

The Data Processing Module calculates key academic metrics such as total marks, percentages, pass rates, and averages to quantify student performance.

Description:

After data import, this module performs computations on the dataset to derive meaningful performance indicators. Total marks and percentage scores are calculated for each student using row-wise summation. Subject-wise averages are computed using colMeans() to provide a snapshot of overall class performance. The module also determines pass and failure rates based on predefined thresholds (e.g., 40 marks). These calculations prepare the dataset for visualization and grading, enabling educators to quickly interpret the academic standing of their students. Furthermore, the module formats the processed data to ensure compatibility with visualization tools and supports real-time updates to accommodate changes in the uploaded data.

Key Functions:

1. Calculate Total and Percentage: Computes total marks and percentages for all students.
2. Compute Subject Averages: Calculates average marks per subject.
3. Determine Pass/Fail Status: Identifies students passing or failing each subject.
4. Prepare Data for Visualization: Structures data for plotting and analysis.

3.3 Graphical Representation Module

Purpose:

This module generates interactive visual plots that summarize student performance, aiding easy interpretation of academic results.

Description:

The Graphical Representation Module transforms numerical data into visual charts using the plotly package. It creates interactive bar plots to show pass percentages and failure rates per subject, pie charts for grade distribution, and average score bar charts for all subjects. These plots offer dynamic features such as tooltips and zooming, allowing users to explore the data intuitively. The visualizations enable quick identification of trends and outliers, helping faculty understand the overall performance landscape in a visually compelling manner. This module also supports dynamic updates so that any new or modified data is immediately reflected in the graphs, making the analysis always current and relevant.

Key Functions:

1. Create Pass Percentage Bar Plot: Displays pass rates per subject.
2. Generate Failure Rate Bar Chart: Highlights subjects with high failure rates.
3. Display Grade Distribution Pie Chart: Visualizes grades awarded across the class.
4. Show Average Scores: Bar charts presenting average marks per subject.

3.4 Performance Analysis Module

Purpose:

The Performance Analysis Module classifies subjects and students based on performance metrics to identify strengths and weaknesses.

Description:

This module analyzes processed data to categorize subjects as weak or strong based on average marks, with thresholds set at below 40 and above 70 respectively. It also assigns grades to students using their average scores, classified into A+, A, B, C, D, and F grades by applying functions like rowMeans() and cut(). These classifications help educators target academic interventions and recognize top-performing students. Additionally, this module provides summary statistics and generates recommendations for curriculum improvements or remedial actions. It thus acts as a decision support tool that guides educators in enhancing student outcomes based on data-driven insights.

Key Functions:

1. Identify Weak and Strong Subjects: Classifies subjects by average marks.
2. Grade Students: Assigns letter grades based on overall performance.
3. Highlight Improvement Areas: Identifies subjects requiring additional focus.
4. Summarize Class Performance: Provides an overall performance snapshot.

3.5 Login and Navigation Module

Purpose:

The Login and Navigation Module manages secure access to the application and facilitates smooth user navigation.

Description:

This module implements user authentication, restricting access to authorized personnel through a secure login interface. Built using shinydashboard and reactive programming, it controls access to key sections such as Dashboard, Upload, Analysis, and Prediction. The navigation system is designed with sidebar menus and conditional panels, providing an intuitive and user-friendly interface. It enhances user experience by ensuring users only see relevant options based on their login status, preventing unauthorized access. This module is critical for maintaining data confidentiality, especially when handling sensitive academic information.

Key Functions:

1. User Authentication: Validates credentials for secure login.
2. Access Control: Limits features to authenticated users only.
3. Sidebar Navigation: Provides easy access to different sections.
4. Conditional Panels: Dynamically displays UI elements based on user actions.

CHAPTER 4

CONCLUSION & FUTURE SCOPE

Conclusion

This project successfully developed an interactive R Shiny web application for analyzing and visualizing college-level student exam results. Leveraging R programming concepts such as reactive expressions, data manipulation with dplyr, and dynamic plotting with plotly, the portal offers educators a user-friendly platform to upload exam data, monitor subject-wise performance, and identify areas needing improvement. The integration of secure login, responsive UI via shinydashboard and bslib, and modular design has resulted in a scalable, maintainable solution. The automated generation of pass/fail statistics, grade distributions, and subject recommendations empowers academic stakeholders to make data-driven decisions efficiently.

Future Scope

While the current implementation provides core functionalities for student performance analysis, several enhancements can be incorporated in the future to increase its utility and impact:

- Integration with Real-Time Databases: Linking the application to institutional databases for automatic result fetching will streamline the workflow and reduce manual uploads.
- Advanced Predictive Analytics: Incorporating machine learning algorithms to predict student outcomes or dropout risks can provide proactive academic support.
- User Role Management: Adding roles for administrators, teachers, and students with different permissions can make the portal more versatile.
- Expanded Data Inputs: Supporting additional data types such as attendance, assignments, and extracurricular activities could enrich the analysis.

CHAPTER 5

APPENDIX A – SOURCE CODE

```
library(shiny)
library(shinydashboard)
library(plotly)
library(DT)
library(dplyr)
library(shinyjs)
library(bslib)

# Define enhanced UI theme
custom_theme <- bs_theme(
  bootswatch = "minty",
  base_font = font_google("Poppins"),
  version = 5,
  bg = "#f8f9fa",
  fg = "#343a40"
)

ui <- dashboardPage(
  title = "LAN - Student Performance Portal",
  header = dashboardHeader(
    title = span("Student Performance Portal", style = "font-family: 'Poppins',
sans-serif; font-weight: bold;"),
    tags$li(class = "dropdown",
            actionButton("logout_btn", "Logout", class = "btn btn-danger", style =
"margin-top: 10px;")
    )
```

```
  ),
  sidebar = dashboardSidebar(
    width = 250,
    sidebarMenu(
      menuItem("Home", tabName = "home", icon = icon("home")),
      menuItem("Login", tabName = "login", icon = icon("sign-in-alt")),
      menuItem("Dashboard", tabName = "dashboard", icon = icon("chart-bar")),
      menuItem("Upload Results", tabName = "upload", icon = icon("upload")),
      menuItem("Prediction", tabName = "prediction", icon = icon("magic")),
      menuItem("Logout", tabName = "logout", icon = icon("sign-out-alt"))
    )
  ),
  body = dashboardBody(
    useShinyjs(),
    tags$head(
      tags$style(HTML("
        .main-header { background-color: #28a745 !important; }
        .sidebar-menu > li > a { font-size: 14px; font-family: 'Poppins', sans-serif;
}
        .content-wrapper { background-color: #f8f9fa; padding: 20px; }
        .well { background-color: #ffffff; border-radius: 10px; box-shadow: 0 4px
6px rgba(0,0,0,0.1); }
        .btn-primary { background-color: #28a745; border-color: #28a745; }
        .btn-primary:hover { background-color: #218838; border-color: #1e7e34; }
        .btn-danger { background-color: #dc3545; border-color: #dc3545; }
        .btn-danger:hover { background-color: #c82333; border-color: #bd2130; }
        h3 { font-family: 'Poppins', sans-serif; color: #343a40; }
        p { font-family: 'Poppins', sans-serif; color: #6c757d; }
        .shiny-plotly { border-radius: 10px; box-shadow: 0 4px 6px
rgba(0,0,0,0.1); }
```

```
    "))
  ),
  tabItems(
    tabItem(tabName = "home",
            fluidRow(
              column(12,
                     br(),
                     h2("Welcome to the Student Exam Dashboard", align = "center"),
                     p("Analyze, visualize, and get recommendations on student performance in just a few clicks!",
                       align = "center", style = "font-size: 16px;"),
                     br(),
                     img(
                       src = "https://res.cloudinary.com/dpdz5uwvf/image/upload/v1745511056/ac2cb122-7c0a-49f7-8465-fccb71377fc0_wsbcxb.png",
                       height = "300px",
                       style = "display:block; margin:auto;"
                     )
              )
            )
    ),
    tabItem(tabName = "login",
            fluidRow(
              column(4, offset = 4,
                     br(),
                     wellPanel(
                       h3("🔒 Login"),
                       textInput("username", "Username"),
                       passwordInput("password", "Password"),
```

```
                actionButton("login_btn", "Login", class = "btn btn-primary w-
100"),
                div(style = "color:red;", textOutput("login_status"))
              )
            )
          )
        ),
        tabItem(tabName = "dashboard",
                conditionalPanel(
                  condition = "output.loggedIn == true",
                  fluidRow(column(12, h3(" 📊 Dashboard Overview", align =
"center"))),
                  fluidRow(
                    column(6, plotlyOutput("pass_plot")),
                    column(6, plotlyOutput("fail_plot"))
                  ),
                  fluidRow(
                    column(6, plotlyOutput("grade_plot")),
                    column(6, plotlyOutput("average_plot"))
                  )
                )
        ),
        tabItem(tabName = "upload",
                conditionalPanel(
                  condition = "output.loggedIn == true",
                  fluidRow(
                    column(6, fileInput("file", "Upload CSV File", accept = ".csv"))
                  )
                )
        ),
```

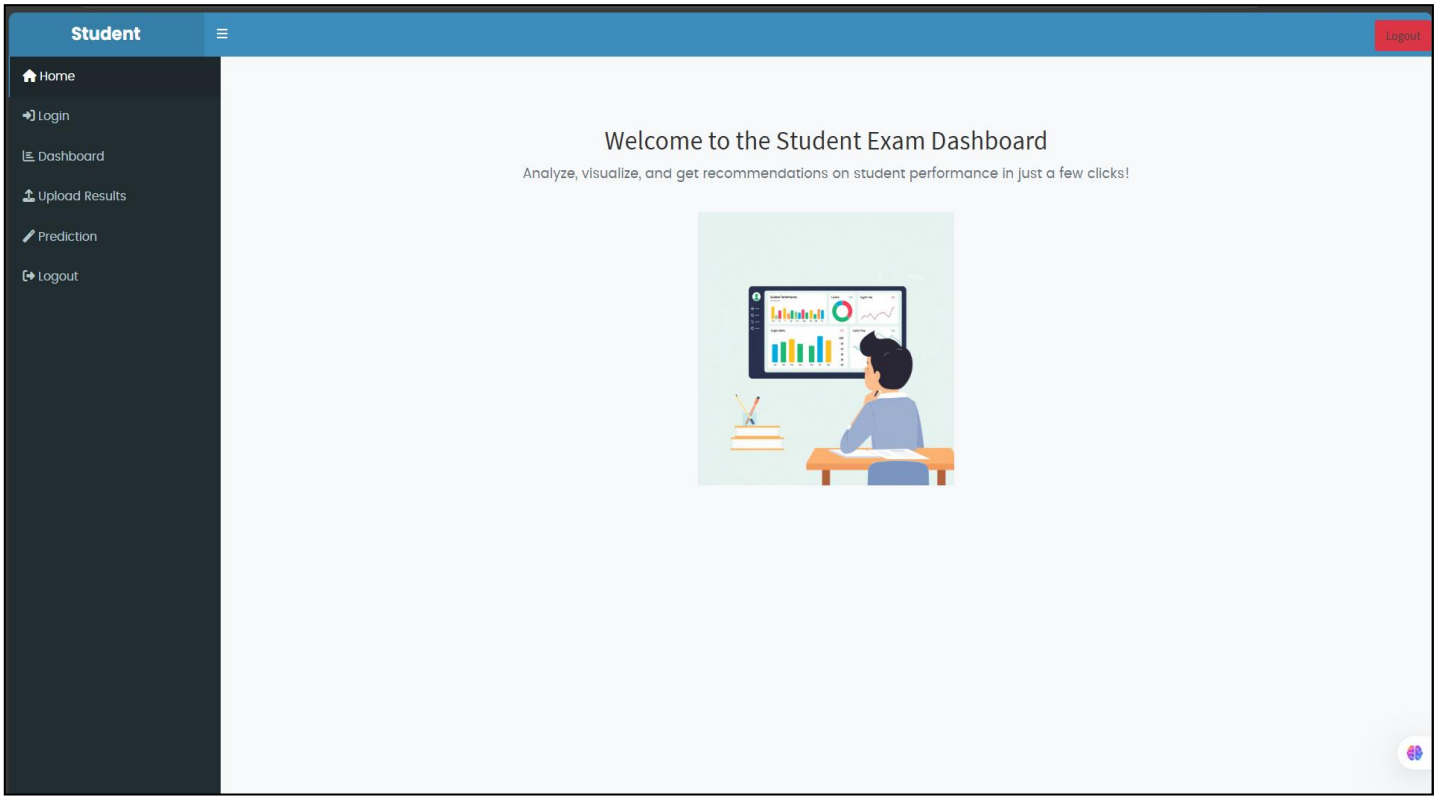
```
    tabItem(tabName = "prediction",
            conditionalPanel(
              condition = "output.loggedIn == true",
              fluidRow(
                column(12,
                       h4(" 🔮 Subject Recommendations"),
                       verbatimTextOutput("prediction"),
                       plotlyOutput("focus_plot")
                )
              )
            )
    ),
    tabItem(tabName = "logout",
            fluidRow(
              column(12,
                     h4("You have been logged out."),
                     actionLink("go_home", "Return to Home", class = "btn btn-
secondary")
              )
            )
    )
  )
 )
)

server <- function(input, output, session) {
  credentials <- list(username = "student", password = "123")
  user_logged_in <- reactiveVal(FALSE)

  output$loggedIn <- reactive({ user_logged_in() })
```

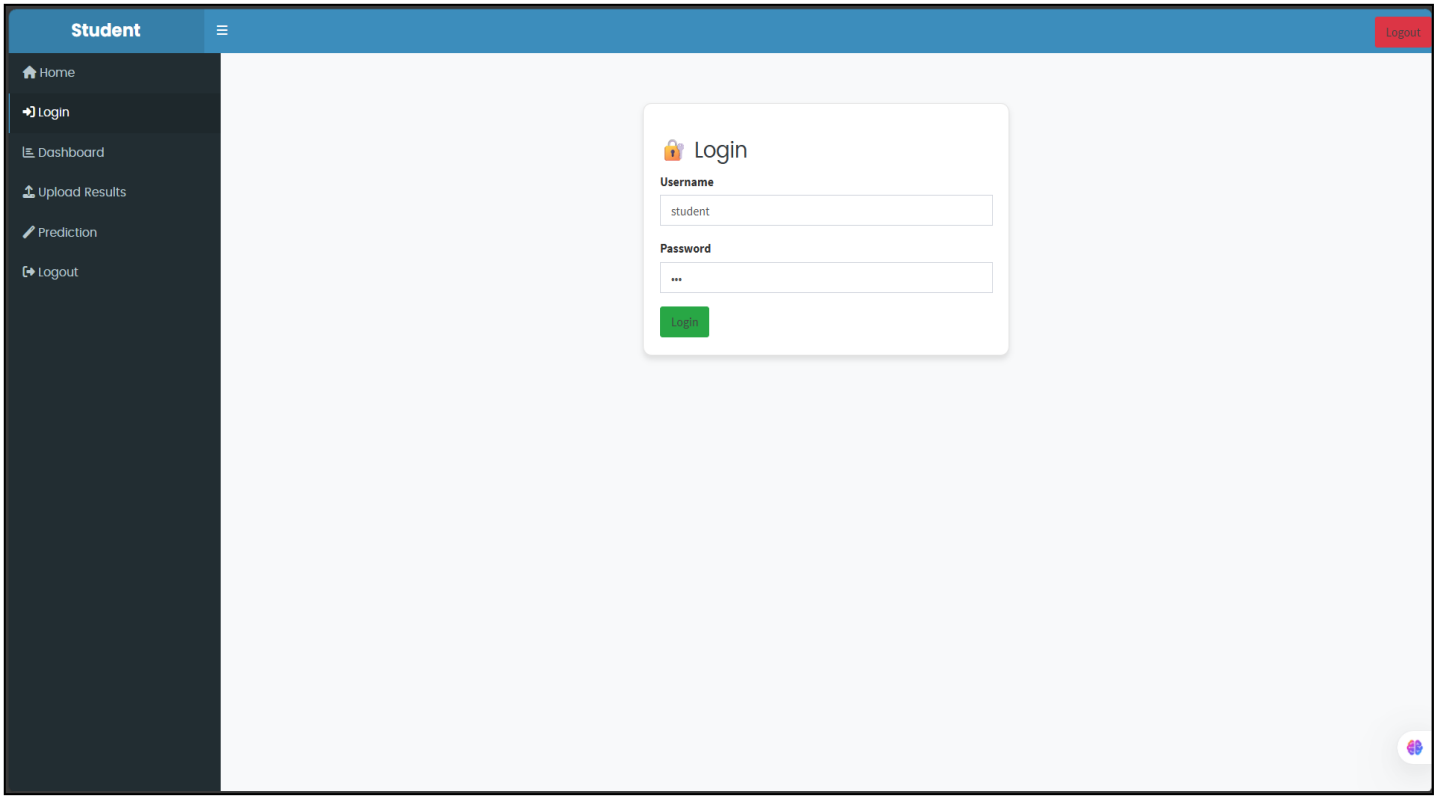
```
  outputOptions(output, "loggedIn", suspendWhenHidden = FALSE)

  observeEvent(input$login_btn, {
    if (input$username == credentials$username && input$password ==
credentials$password) {
      user_logged_in(TRUE)
      updateTabItems(session, "tabs", selected = "dashboard")
    } else {
      output$login_status <- renderText("❌ Invalid username or password.")
    }
  })

  observeEvent(input$logout_btn, {
    user_logged_in(FALSE)
    updateTabItems(session, "tabs", selected = "home")
  })

  dataset <- reactive({
    req(input$file)
    df <- read.csv(input$file$datapath)
    df <- df %>% select(-c(Student_ID, Total))
    return(df)
  })

  output$pass_plot <- renderPlotly({
    df <- dataset()
    pass_rate <- colMeans(df >= 40, na.rm = TRUE) * 100
    plot_ly(x = names(pass_rate), y = pass_rate, type = 'bar', name = 'Pass %')
%>%
      layout(title = "Pass Percentage per Subject")
```

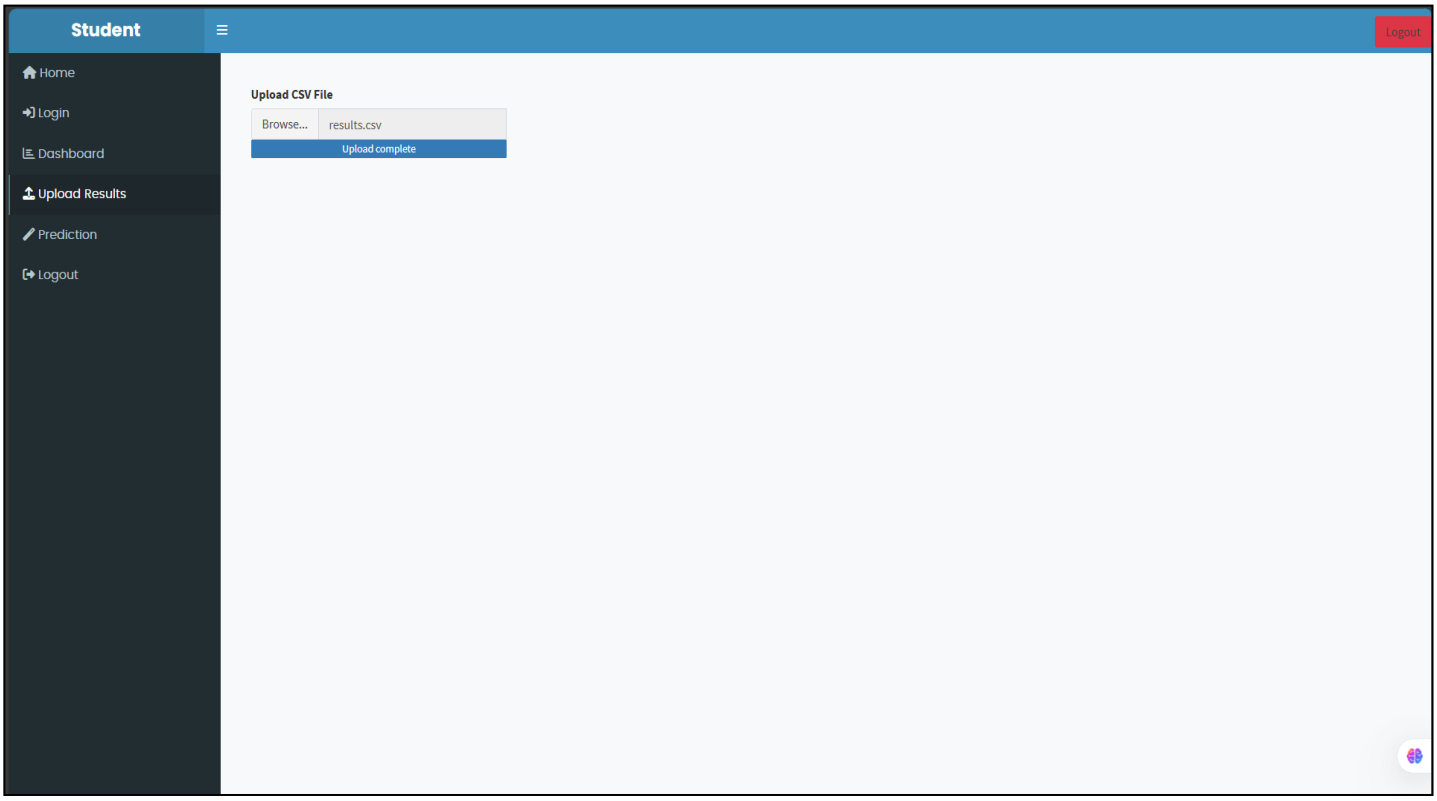
```
  })

  output$fail_plot <- renderPlotly({
    df <- dataset()
    fail_rate <- colMeans(df < 40, na.rm = TRUE) * 100
    plot_ly(x = names(fail_rate), y = fail_rate, type = 'bar', name = 'Fail %') %>%
      layout(title = "Failure Rate per Subject")
  })

  output$grade_plot <- renderPlotly({
    df <- dataset()
    grades <- cut(rowMeans(df), breaks = c(0, 40, 50, 60, 70, 80, 100), labels =
c("F", "D", "C", "B", "A", "A+"))
    grade_data <- table(grades)
    plot_ly(labels = names(grade_data), values = as.numeric(grade_data), type =
'pie') %>%
      layout(title = "Grade Distribution")
  })

  output$average_plot <- renderPlotly({
    df <- dataset()
    avg_scores <- colMeans(df, na.rm = TRUE)
    plot_ly(x = names(avg_scores), y = avg_scores, type = 'bar') %>%
      layout(title = "Average Score per Subject")
  })

  output$prediction <- renderPrint({
    df <- dataset()
    avg <- colMeans(df, na.rm = TRUE)
    cat("📌 Subjects that need focus (avg < 40):\n")
```

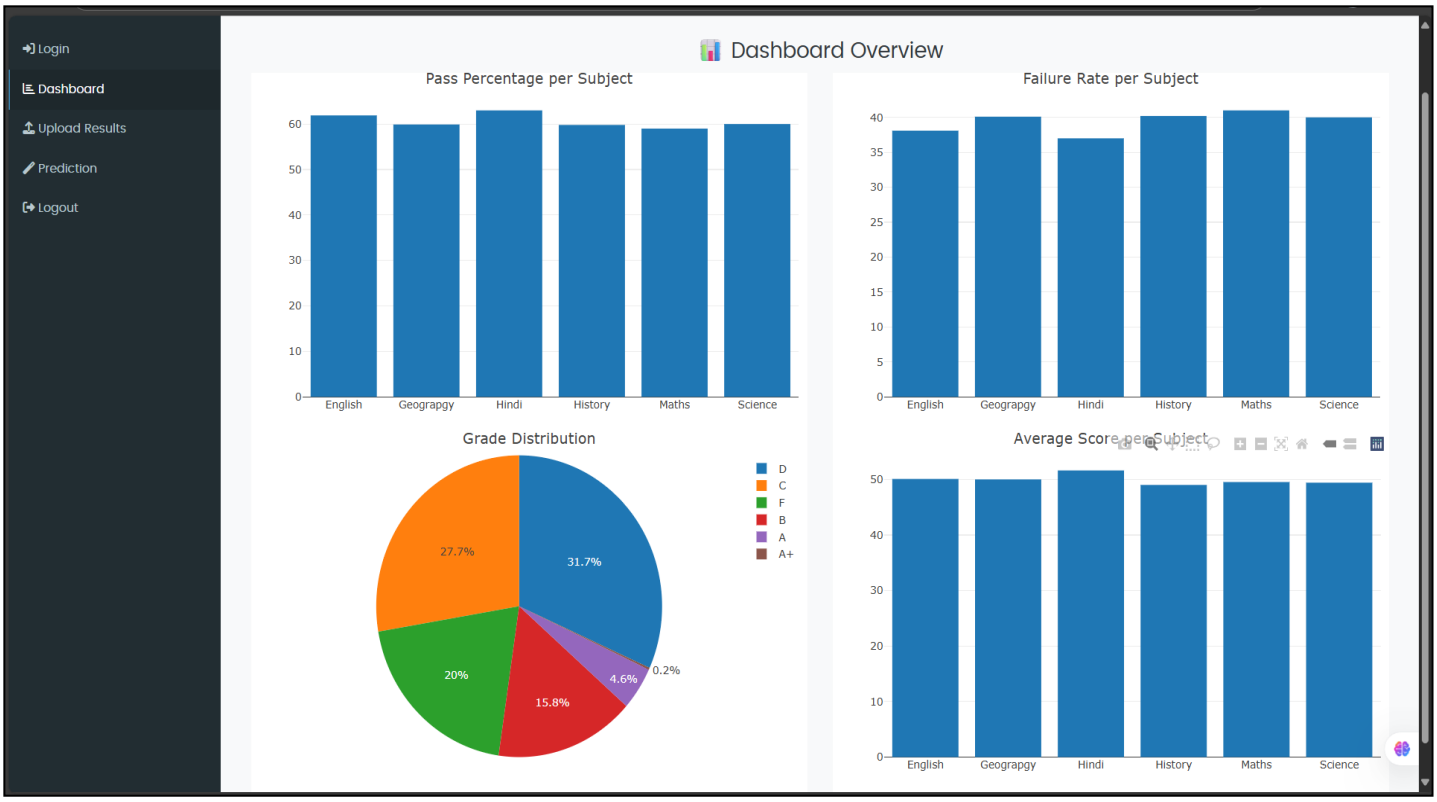
```
    print(names(avg[avg < 40]))
    cat("\n🏆 Strong Subjects (avg > 70):\n")
    print(names(avg[avg > 70]))
  })

  output$focus_plot <- renderPlotly({
    df <- dataset()
    avg_scores <- colMeans(df, na.rm = TRUE)
    plot_ly(x = names(avg_scores), y = avg_scores, type = 'bar', marker =
list(color = avg_scores)) %>%
      layout(title = "📊 Subject Performance Overview")
  })
}

shinyApp(ui = ui, server = server)
```

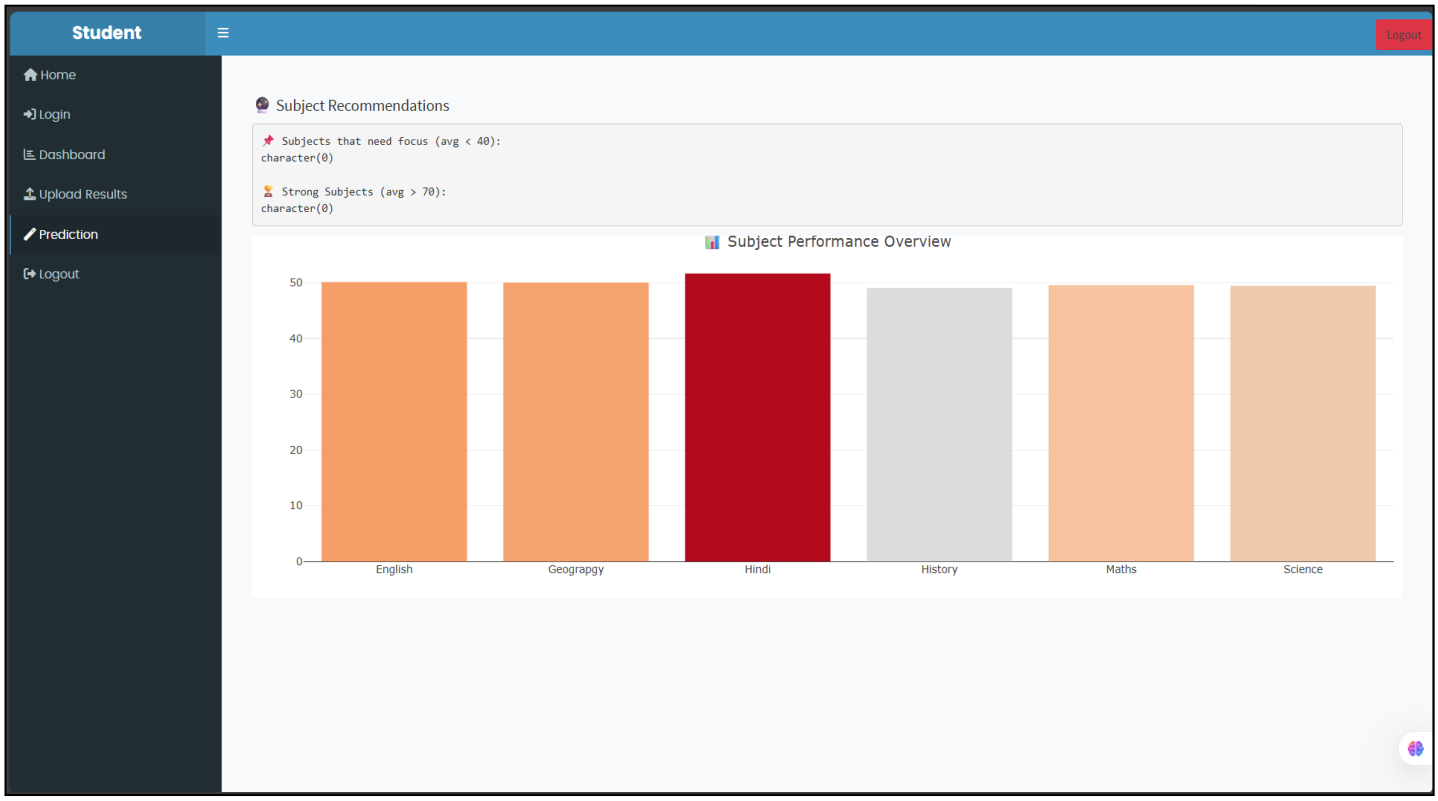
APPENDIX B – SCREENSHOTS






Student
Logout
Home
Login
Dashboard
Upload Results
Prediction
Logout
Upload CSV File
Browse...
results.csv
Upload complete


Login
Dashboard
Upload Results
Prediction
Logout
Dashboard Overview
Pass Percentage per Subject
60
50
40
30
20
10
0
English
Geograpgy
Hindi
History
Maths
Science
Failure Rate per Subject
40
35
30
25
20
15
10
5
0
English
Geograpgy
Hindi
History
Maths
Science
Grade Distribution
27.7%
31.7%
20%
15.8%
4.6%
0.2%
D
C
F
B
A
A+
Average Score per Subject
50
40
30
20
10
0
English
Geograpgy
Hindi
History
Maths
Science


Student
Logout
Home
Login
Dashboard
Upload Results
Prediction
Logout
Subject Recommendations
Subjects that need focus (avg < 40):
character(0)
Strong Subjects (avg > 70):
character(0)
Subject Performance Overview
50
40
30
20
10
0
English
Geograpgy
Hindi
History
Maths
Science

REFERENCES

1. Baumer, B. S., Kaplan, D. T., & Horton, N. J. (2017). *Modern Data Science with R*. CRC Press.
2. Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2024). *Shiny: Web Application Framework for R*. R package version X.X.X. https://CRAN.R-project.org/package=shiny
3. Official R Project Website
4. RStudio. *Shiny Tutorials by RStudio*.
5. Sievert, C. (2023). *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC.
6. Wickham, H., François, R., Henry, L., & Müller, K. (2024). *dplyr: A Grammar of Data Manipulation*. R package version X.X.X.
7. Winston Chang et al. (2024). *shinydashboard: Create Dashboards with 'Shiny'*.