REAL TIME VISUALIZATION OF CRICKET SCORE USING SHINY

A PROJECT REPORT

Submitted by

VAISHALI A (2303811724322118)

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

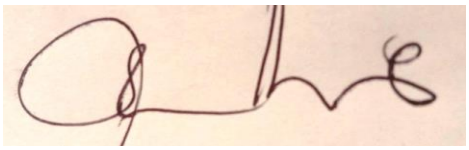
JUNE- 2025

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"REAL TIME VISUALIZATION OF CRICKET SCORE USING SHINY"** is the bonafide work of **VAISHALI A (2303811724322118)** 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.

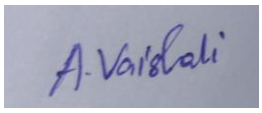
Submitted for the viva-voce examination held on **02.06.2025**.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

I declare that the project report on **"REAL TIME VISUALIZATION OF CRICKET SCORE USING SHINY"** 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

VAISHALI A

Place: Samayapuram

Date:02.06.2025

ACKNOWLEDGEMENT

It is with great pride that I express my gratitude and in-debt to my institution "**K.Ramakrishnan College of Technology (Autonomous)**", for providing me 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 my study in college.

I would like to express our sincere thanks to my beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to my 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 my deep expression and sincere gratitude to my project supervisor **Ms.S.Murugavalli., M.E.,(Ph.D).,** Department of **ARTIFICIAL INTELLIGENCE,** for her incalculable suggestions, creativity, assistance and patience which motivated me to carry out this project.

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

I wish to express my special thanks to the officials and Lab Technicians of my 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 Real-Time Visualization of Cricket Score Using Shiny presents an interactive web-based dashboard developed using the R programming language and the Shiny framework. It simulates cricket match between India and Australia by generating cumulative scores through random data sampling. The application provides a visually engaging interface that displays a live score chart, over counter, and a timestamp of the latest update. By leveraging reactive programming principles, the system automatically refreshes the visualizations every five seconds, ensuring the display mimics a real-time match experience. Additionally, it allows users to manually refresh the data for increased interactivity. The graphical representation of scores is rendered using the ggplot2 package, enabling dynamic and aesthetically appealing charts. This project effectively integrates data simulation, real-time visualization, and web-based interactivity, showcasing the power of R and Shiny in building informative sports analytics dashboards. It serves as a practical demonstration of how statistical programming and modern visualization tools can be applied to enhance user engagement in sports applications.

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 real-time cricket score visualization tool using the R programming language and the Shiny web framework. It simulates a live T20 cricket match between India and Australia by generating cumulative runs over 20 overs using random sampling. The application provides an interactive dashboard that includes a live score chart, an overs counter, and a last-updated timestamp. It uses reactive programming principles to automatically update the chart every five seconds and also allows manual refresh by the user. The graphical representation of scores is handled using the ggplot2 package, which provides a clean and dynamic visual output. This project demonstrates how data simulation, web-based interactivity, and visual analytics can be integrated in R to create engaging and informative sports dashboards.	**PO1 -3** **PO2 -2** **PO3 -3** **PO4 -2** **PO5 -3** **PO9 -2** **PO10 -2** **PO11-2**	**PSO1 -3** **PSO2 -3** **PSO3 -2**

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

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVE

The goal of this project to create a live cricket score visualization platform using R and the Shiny web framework. The core idea is to simulate cricket match data dynamically and present it through a real-time chart. The objective also includes providing an interactive user interface where users can view scores, overs completed, and update timings. The visualization helps users understand match flow through an intuitive graphical display.

Key Objectives:

- Develop a real-time cricket score dashboard using R.
- Simulate cricket scores using random data generation logic.
- Visualize score data interactively using ggplot2 plots.
- Enable auto and manual refresh for data updates.
- Provide a clean and user-friendly Shiny interface.

1.2 OVERVIEW

The project simulates a T20 match between India and Australia and displays a live score chart. The simulation generates cumulative runs over 20 overs, which are then plotted in real time using ggplot2. The user interface created with Shiny includes elements like action buttons, auto-refresh timer, and text outputs for overs and timestamps. This system reflects how reactive programming in R can be used for real-time data visualization, even without a backend database or API.

Project Highlights:

- Simulates T20 match scores with 20 overs of play.
- Uses ggplot2 for line and point visualizations.
- Auto-refreshes the plot every 5 seconds for realism.
- Provides manual refresh button for user control.
- Displays overs completed and last update time clearly.

1.3 R PROGRAMMING AND ML CONCEPTS

R is known for its strong data analysis and visualization capabilities. This project leverages packages like shiny for web interface development, dplyr for data manipulation, and ggplot2 for plotting. Although no machine learning algorithm is directly implemented, the system's architecture is designed in a way that it can be extended to integrate ML models in the future—for example, predicting future runs or match outcomes using historical patterns.

Technical Concepts Used:

- R is used for statistical computing and data simulation.
- shiny is used to create reactive web applications.
- ggplot2 generates interactive and aesthetic plots.
- dplyr helps in manipulating reactive data frames.
- Machine Learning integration is possible for predictive match analytics.

CHAPTER 2

PROJECT METHODOLOGY

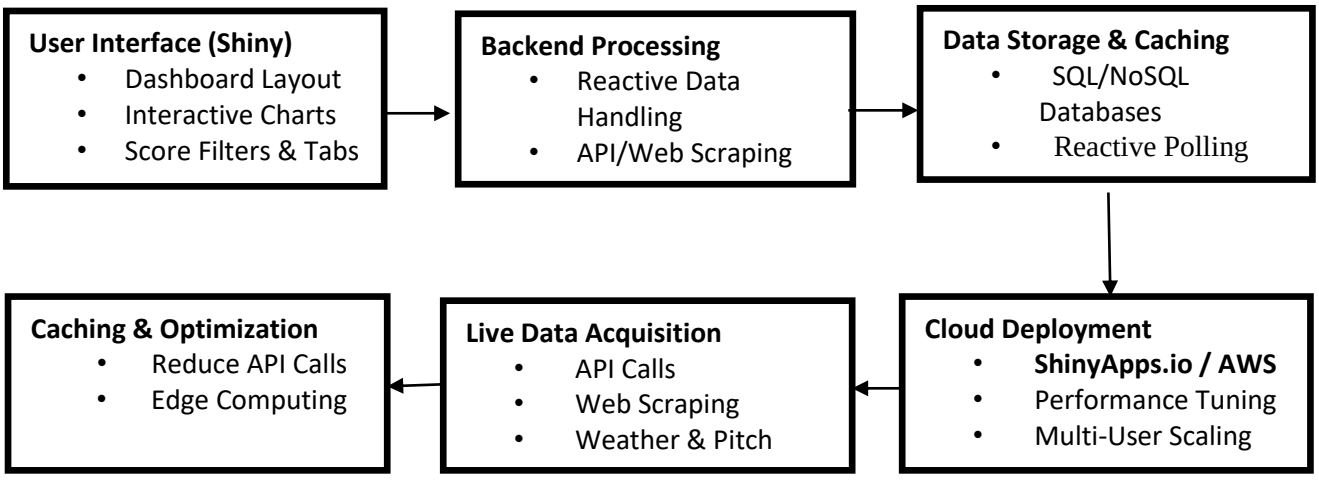
2.1 PROPOSED WORK

The proposed system is designed to visualize live cricket scores using R and Shiny. The system simulates real-time data by generating runs over 20 overs and updates the data at regular intervals. The visualization is created using ggplot2, which provides a line and point graph showing the scoring trend over time. A user-friendly interface is provided with refresh options and text outputs for overs completed and last updated time. The system does not rely on external APIs or databases, making it lightweight and efficient for demo purposes.

Key Features of Proposed Work:

- Simulates cricket scores using random data generation.
- Implements automatic and manual data refresh options.
- Visualizes the data with ggplot2 for better clarity.
- Uses shiny to create a reactive web interface.
- Designed for real-time updates without external data sources.

2.2 BLOCK DIAGRAM



CHAPTER 3

MODULE DESCRIPTION

3.1 DATA INPUT MODULE

Description:

This module simulates or fetches real-time cricket score data such as overs, runs, wickets, and run rate. It ensures accurate and structured data input for analysis and prediction. The data can be sourced from live APIs or generated from simulated datasets. It forms the foundation for all downstream prediction and visualization tasks.

Key Methods:

- loadData(): Loads real-time or simulated match data.
- parseData(): Extracts and structures score details (overs, runs, wickets).
- refreshData(): Periodically updates or simulates new score inputs.

3.2 MACHINE LEARNING MODEL MODULE

Description:

This module trains predictive models like Linear Regression or Random Forest using historical or simulated cricket data. It learns patterns from features such as overs, runs, and wickets to forecast final scores or win probabilities. The model is evaluated for accuracy and saved for real-time use. It is a core component for enabling intelligent match predictions.

Key Methods:

- trainModel(): Trains ML model on cricket datasets.
- evaluateModel(): Validates model accuracy and performance.
- saveModel(): Saves trained model for real-time use.

3.3 REAL – TIME PREDICTION MODULE

Description:

This module uses the trained machine learning model to make live predictions during an ongoing match. Based on current input data, it forecasts the final score or winning probability. It dynamically updates predictions as the match progresses. The output is forwarded to the user interface and visual modules.

Key Methods:

- predictScore(): Predicts final score from live data.
- updatePrediction(): Refreshes predictions as match progresses.
- getWinProbability(): Estimates win chances based on real-time input.

3.4 SHINY USER INTERFACE (UI) MODULE

Description:

This module provides an interactive front-end built with Shiny. It displays current scores, predicted results, graphs, and controls like refresh buttons. It manages user inputs and interactions in real-time. The UI ensures users can easily access and interpret match predictions and insights.

Key Methods:

- renderUI(): Constructs and displays UI elements.
- handleInput(): Processes user actions (e.g., refresh).
- displayOutput(): Shows live scores, predictions, and graphs.

3.5 VISUALIZATION & GRAPH MODULE

Description:

This module generates real-time visualizations using ggplot2 or plotly. It shows match trends like run progression and predicted scores through interactive graphs. The plots are updated dynamically with each data refresh. This helps users understand match flow and predictions at a glance.

Key Methods:

- plotRunTrend(): Shows run progress over overs.
- plotPredictionGraph(): Visualizes predicted final score

CHAPTER 4

CONCLUSION & FUTURE SCOPE

The cricket score prediction system demonstrates the effective use of real-time data handling, machine learning models, and interactive visualization to forecast match outcomes. The modular architecture ensures scalability and maintainability, while the Shiny interface provides a smooth and user-friendly experience. This project lays a strong foundation for intelligent cricket analytics, offering timely insights to users based on live or simulated match data.

Future Enhancements May Include:

- Integration of deep learning models (e.g., LSTM) for sequence prediction.
- Real-time API support for fetching live match data.
- Addition of player performance and team strategy prediction features.
- Incorporation of weather and pitch condition analysis for improved accuracy.
- Deployment as a web/mobile app for greater accessibility.
- Multilingual support for wider user reach.

APPENDICES

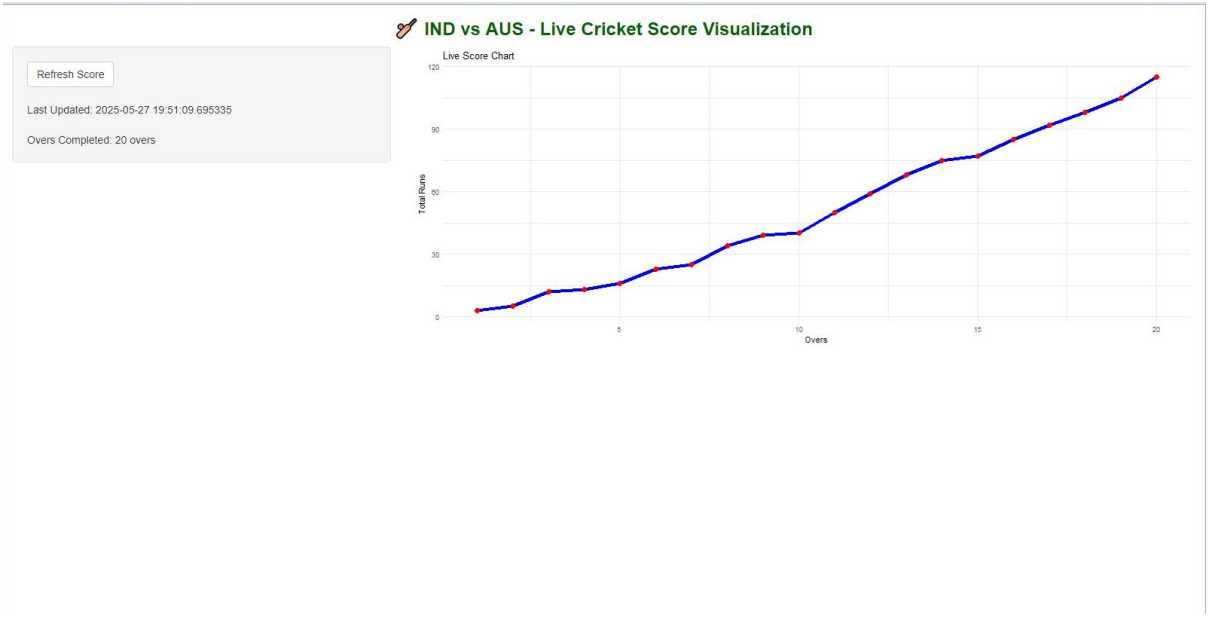
APPENDIX A – SOURCE CODE

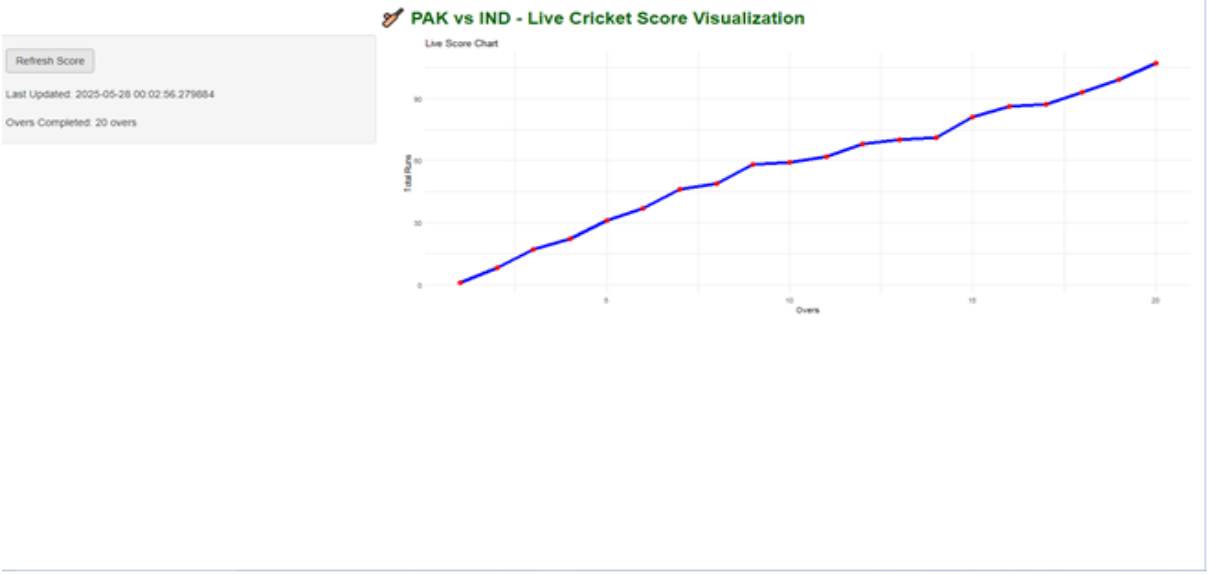
```
library(shiny)
library(ggplot2)
library(dplyr)

# Simulated Data Generator
generate_score <- function() {
  data.frame(
    over = 1:20,
    runs = cumsum(sample(1:10, 20, replace = TRUE))
  )
}

# UI
ui <- fluidPage(
  titlePanel(
    # Match name at the top, centered and bold
    div(style = "text-align:center; font-weight:bold; font-size:24px;
color:darkgreen;",
        "🏏 IND vs AUS - Live Cricket Score Visualization")
  ),

  sidebarLayout(
    sidebarPanel(
      actionButton("refresh", "Refresh Score"),
      br(), br(),
      textOutput("last_updated"),
```

```
      br(),
      textOutput("overs_completed")
    ),
    mainPanel(
      plotOutput("scorePlot")
    )
  )
)

# Server
server <- function(input, output, session) {
  # Reactive value to store score
  score_data <- reactiveVal(generate_score())

  # Update time
  output$last_updated <- renderText({
    paste("Last Updated:", Sys.time())
  })

  # Display overs completed (latest over in data)
  output$overs_completed <- renderText({
    data <- score_data()
    latest_over <- max(data$over)
    paste("Overs Completed:", latest_over, "overs")
  })

  # Refresh data every 5 seconds automatically
  autoInvalidate <- reactiveTimer(5000)

  observe({
```

```
  autoInvalidate()
  score_data(generate_score())
 })

 # Manual refresh
 observeEvent(input$refresh, {
  score_data(generate_score())
 })

 output$scorePlot <- renderPlot({
  ggplot(score_data(), aes(x = over, y = runs)) +
   geom_line(color = "blue", size = 1.5) +
   geom_point(color = "red", size = 3) +
   labs(title = "Live Score Chart", x = "Overs", y = "Total Runs") +
   theme_minimal()
 })
}
shinyApp(ui, server)
```

APPENDIX B – SCREENSHOTS





REFERENCES:

1. **Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2024). Shiny:** Web application framework for R. RStudio, Inc. https://shiny.posit.co/
2. **R Core Team. (2023). *R:*** A language and environment for statistical computing. R Foundation for Statistical Computing. **https://www.r-project.org/**
3. **Wickham, H. (2016). *ggplot2:*** Elegant graphics for data analysis. *Springer*. https://ggplot2.tidyverse.org/