Garcia, Cayden

Cgarcia174@mail.stmarytx.edu

SlackR User Manual

A guide to the exciting new Valorant Statistic tracker

# Overview

This section provides a broad introduction to SlackR, a cutting-edge Valorant stat tracker designed to give players in-depth insights into their gameplay. SlackR combines statistical analysis with a user-friendly interface to present meaningful data about individual and team performance.

SlackR’s unique selling point is its comedic twist, offering not just raw stats but also lighthearted summaries and rank assignments based on playstyle archetypes. Whether you’re a casual gamer or part of a competitive team, SlackR is your ultimate tool for performance evaluation and improvement.

## Background

SlackR is a powerful and fun **Valorant stat tracker** designed for players who want a deeper, more entertaining dive into their gameplay. By leveraging APIs and efficient data processing with **requests** and **pandas**, SlackR gathers valuable performance data, such as **whiff percentage**, **first deaths**, and **rounds clutched against**, giving players a more complete view of their match performance. It goes beyond the traditional K/D ratios, focusing on unique and impactful aspects of gameplay.

Designed for **Valorant players**, **team managers**, and anyone interested in improving their game or having fun with quirky stats, SlackR pulls data directly from the **Valorant API** without the need for logging in. Users simply input their **Riot Name** and **Riot ID** to view their stats.

## System Features

* **Overview Page:** Enter your **Riot Name** and **ID** to get a snapshot of your performance.
* **Games Page:** View a list of your recent games and select one to see detailed stats
* **Breakdown Page:** Detailed match economy analysis to see how your team performed across rounds.
* **Teammate Overview:** Select a teammate to view their performance, as long as their profile is public.

Contents

[Overview 1](#_Toc183505177)

[Background 1](#_Toc183505178)

[System Features 1](#_Toc183505179)

[Requirements 2](#_Toc183505180)

[Hardware Requirements 2](#_Toc183505181)

[Software Requirement 2](#_Toc183505182)

[User Knowledge Assumptions 2](#_Toc183505183)

[Using the System 3](#_Toc183505184)

[Installation Instructions 3](#_Toc183505185)

[1. Download the SlackR Repository 3](#_Toc183505186)

[2. Install Python 3](#_Toc183505187)

[3. Install Required Libraries 3](#_Toc183505188)

[4. Running SlackR 3](#_Toc183505189)

[How to use SlackR 4](#_Toc183505190)

[1. Launch the System 4](#_Toc183505191)

[2. Viewing Your Overview Page 4](#_Toc183505192)

[3. Navigating the Game Pages 4](#_Toc183505193)

[4. Match Stats 4](#_Toc183505194)

[5. Economy Breakdown 5](#_Toc183505195)

[6. Teammate Overview 5](#_Toc183505196)

[Sample Sessions 6](#_Toc183505197)

[Session 1: Overview Page 6](#_Toc183505198)

[Session 2: Games Page 8](#_Toc183505199)

[Session 3: Breakdowns Page 10](#_Toc183505200)

[Session 4: Teammate Overview Page 11](#_Toc183505201)

[Troubleshooting 12](#_Toc183505202)

[References 12](#_Toc183505203)

# Requirements

To ensure the seamless operation of SlackR, this section outlines all the technical and software prerequisites. From the hardware specifications to the software dependencies, you’ll find a detailed list of everything needed to install, run, and use the system effectively.

Additionally, we discuss the knowledge the user must already possess, such as familiarity with Riot’s naming conventions and basic navigation of software interfaces. If you’re new to Riot IDs or data visualization tools, don’t worry—SlackR is intuitive and easy to use for all skill levels.

This section is crucial for understanding what you need to get started and what environments SlackR is optimized for.

## Hardware Requirements

* **Computer** with:
  + At least 4GB RAM (8GB recommended for better performance)
  + 100MB of free disk space for the application
  + A stable internet connection for retrieving data

## Software Requirement

* **Python 3.7 or higher:** This is the version used for developing SlackR
* **Python Libraries:**
  + **PySimpleGUI:** For the GUI
  + **pandas:** For data manipulation and organization
  + **requests:** To fetch data from the Valorant API
  + **os:** For file management (e.g., saving and loading JSON files)
  + **json:** For parsing and handling JSON data
  + **itertools:** For iterating over data efficiently
  + **numpy:** For numerical data processing, especially for handling economy breakdown

You can install all necessary libraries by running:

*pip install pandas PySimpleGUI requests numpy*

## User Knowledge Assumptions

* Basic understanding of **Valorant gameplay** (match statistics, teams, etc.)
* No prior programming knowledge required for the user
* Ability to input Riot Name and ID and navigate through the GUI

# Using the System

This section is your step-by-step guide to installing, launching, and using SlackR. We’ll walk you through the process of entering your Riot Name and Riot ID, navigating the main features, and exploring detailed match and player statistics.

SlackR has been designed with ease of use in mind, and this section ensures that every user, regardless of technical background, can set up and interact with the system effortlessly.

From the initial login screen to advanced features like economy breakdowns and teammate comparisons, this section highlights how you can get the most out of SlackR.

## Installation Instructions

### Download the SlackR Repository

* Download the repository from GitHub or clone it using the following command

*Git clone* [*https://github.com/SlackR*](https://github.com/SlackR)

### Install Python

* Make sure **Python 3.7 or higher** is installed, if not download it from [here](https://www.python.org/downloads/)

### Install Required Libraries

* In your terminal, navigate to the SlackR folder and run:

*pip install -r requirements.txt*

### Running SlackR

* To run the application, open the terminal, navigate to the folder where SlackR is located, and type:

*python slackr.py*

## How to use SlackR

### 1. Launch the System

* After running the program, a window will open where you’ll need to enter your **Riot Name** and **Riot ID**
  + **Riot Name:** The display name you use and Valorant
  + **Riot ID:** Your unique ID associated with your Riot account

### 2. Viewing Your Overview Page

* Once you input your details, SlackR will load your **overview page,** displaying your:
  + **KD**
  + **Win Percentage**
  + **Top Agent**
  + **Headshot Percentage**
  + **Clutches**
  + **First Kills**
  + **First Deaths**
  + **Knife Kills**
  + **149 Damage Done**
  + **Rank**
  + **Archetype**
* This page provides an overview of your stats for the current **ranked** season

### 3. Navigating the Game Pages

* On the **overview page,** there is a tab called ‘Games’.
* This page will show your recent matches. Click on any of your recent games to dive deeper into specific match statistics

### 4. Match Stats

* After selecting a match, you will be able to view detailed stats for you and your teammates. This includes:
  + **Individual kills, deaths, assists, headshot percentage, multi-kills, and agent name**

### 5. Economy Breakdown

* If you want to see a more detailed breakdown of the match economy, you can click the **Breakdowns** button. This page will show you each teams maximum economy, minimum economy, and average economy
* There is also a graph that shows each teams round economy for the length of the game

### 6. Teammate Overview

* You also have the option to select any of your teammates (if their profile is public) and view their individual overview stats
* This gives you insights into their gameplay performance and allows deeper team analysis

# Sample Sessions

To give you a hands-on understanding of SlackR, this section provides real-world examples of how the system is used. Each session illustrates a key workflow, such as viewing your performance overview, analyzing a specific match, or diving into economy breakdowns.

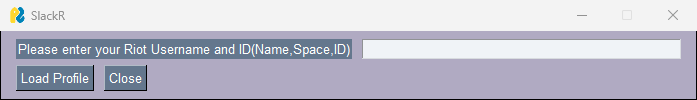
These scenarios simulate common use cases and showcase SlackR’s capabilities, helping you familiarize yourself with its interface and tools. Screenshots are included to make the processes crystal clear.

By the end of this section, you’ll feel confident navigating the system and interpreting the wealth of data SlackR provides.

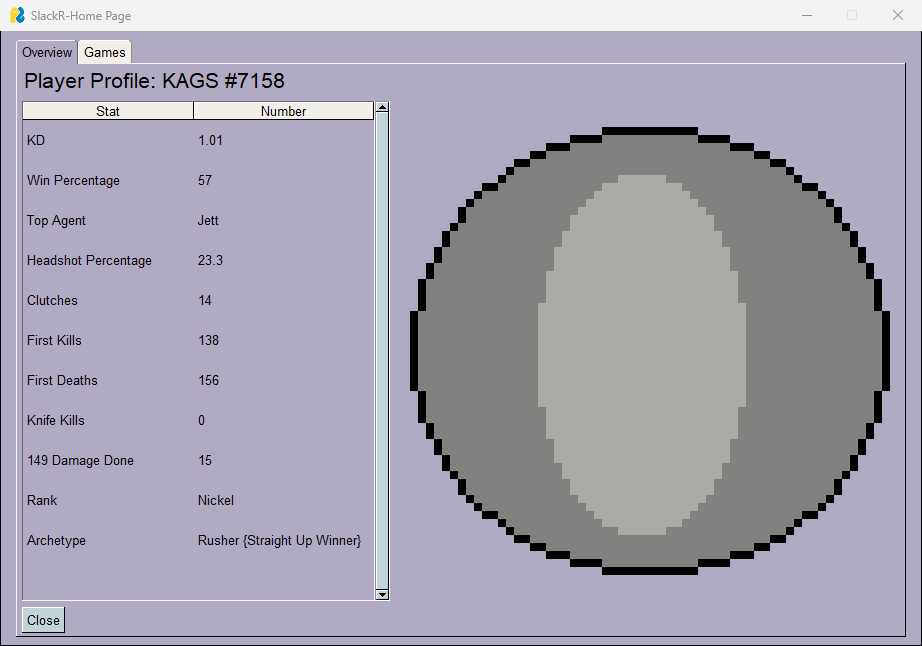
## Session 1: Overview Page

**Scenario:** A user wants to get an overview of their recent performance in Valorant

1. **User action:**
   1. The user launches SlackR
   2. The **Login Page** appears, prompting them to input their **Riot Name** and **Riot ID**
   3. Example:
      1. **Riot Name:** “KAGS”
      2. **Riot ID:** “#7158”

****

1. **System Response:**
   1. After the user enters the necessary details, the system fetches their match history and load the **Overview Page**
   2. The overview page Displays:
      1. **KD**
      2. **Win Percentage**
      3. **Top Agent**
      4. **Headshot Percentage**
      5. **Clutches**
      6. **First Kills**
      7. **First Deaths**
      8. **Knife Kills**
      9. **149 Damage Done**
      10. **Rank**
      11. **Archetype**
2. **Expected Results**
   1. The user sees a clean, readable layout with their stats clearly displayed. The design uses **PySimpleGUI** elements to neatly organize the data and ensure it’s easy to navigate
   2. The system should automatically fetch stats from the **Strats.gg** API and present them in real time without issues, assuming the riot name and ID are valid
3. **User Interaction**
   1. If they want game information they can click on the ‘Games’ tab to view their recent games player

****

## Session 2: Games Page

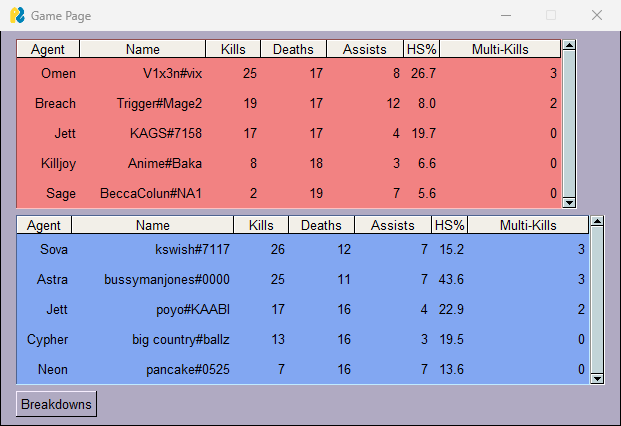
**Scenario:** The user wants to analyze one of their recent games to dive deeper into their performance

1. **User Action:**
   1. The user clicks on ‘Games’ tab from the Overview Page
   2. The system loads the **Games Page**, displaying a list of their last 5 matches with the player’s rank, the Map name, and the agent played
   3. The user selects a match to view the detailed stats

A screenshot of a computer

Description automatically generated

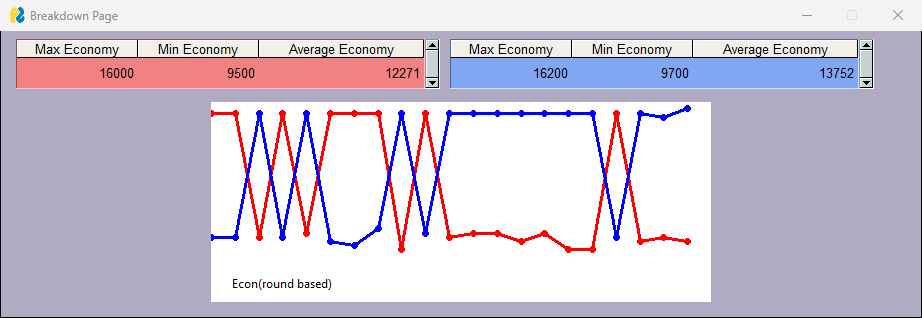
1. **System Response**
   1. After the match is selected, the system displays detailed stats for both the user’s team and the opposing team
   2. **Displayed Stats:**
      1. **Agent**
      2. **Name**
      3. **Kills**
      4. **Deaths**
      5. **Assists**
      6. **HS%**
      7. **Multi-Kills**
2. **Expected Results:**
   1. The user can clearly see how their stats compare to teammates’ stats. The breakdown of performance gives insight into who carried the match, who contributed, and who might need more practice in certain areas
3. **User Interaction:**
   1. The user now has options:
      1. Click on ‘Breakdowns’ to view each teams economy performance for the game
      2. Click on a teammate’s name (e.g. **JohnDoe**) to view that teammate’s individual stats on a separate **Teammate Overview Page**



## Session 3: Breakdowns Page

**Scenario:** The user wants to analyze the match economy to see how well their team performed financially throughout the game

1. **User Action:**
   1. After selecting the game and reviewing basic stats, the user clicks on the‘Breakdowns’ button to view an analysis of the match’s economy
2. **System Response:**
   1. The economy breakdown displays round-by-round data, showing how much money each team had after each round, as well as the overall team economy

****

1. **Expected Results:**
   1. The user can see whether their team had a financial advantage or disadvantage during the game and which rounds were key for economy management
   2. The system also calculates the maximum, minimum, and average amount of money each team had during the game

## Session 4: Teammate Overview Page

**Scenario:** The user wants to analyze a teammate’s stats in depth

1. **User Action:**
   1. On the **Games Page**, the user selects a teammate’s name, i.e. *JohnDoe*, and clicks it to view their detailed stats
   2. If the teammate’s profile is **public**, the **Teammate Overview Page** is displayed
2. **System Response:**
   1. The **Teammate Overview Page** shows detailed stats for *JohnDoe* including:
      1. **Agent**
      2. **Name**
      3. **Kills**
      4. **Deaths**
      5. **Assists**
      6. **HS%**
      7. **Multi-Kills**
3. **Expected Results:**
   1. The user can assess their teammate’s performance during the game and gain insights into areas for improvement. For example, they notice that **JohnDoe** had a higher **First Death** rate and could practice with entering into a site with a team
4. **User Interaction:**
   1. The user can:
      1. Use the stats to give feedback to their teammate
      2. Return to the **Game Overview Page**, to review other teammates stats

A computer screen shot of a black object

Description automatically generated

# Troubleshooting

No software system is perfect, and SlackR is no exception. This section addresses common issues you might encounter while using the system, such as connectivity problems, invalid Riot IDs, or data fetching errors.

|  |  |  |
| --- | --- | --- |
| **Problem** | **Error Message** | **Solution** |
| Incorrect Riot ID or Name | “Not valid username or ID” | Ensure your Riot Name and ID are correct. Double-check for typos. |
| Private Profile | “Private Profile” | N/A |
| Application crashes | “Application Error” | Make sure all dependencies are correctly installed. Reinstall if necessary. |
| API crashing | “JSON file could not be received” | Ensure the Strats.GG website is currently active |
|  |  |  |
|  |  |  |
|  |  |  |

# References

* **Strats.gg Website**  
  Author: Strats.gg Development Team  
  Title: *Valorant Strategies, Stats, and Analysis Tools*  
  Copyright: © Strats.gg, 2024  
  Publisher: Strats.gg  
  Location: <https://strats.gg/valorant>  
  Date Accessed: November 26, 2024
* **PySimpleGUI Documentation**  
  Author: PySimpleGUI Team  
  Title: *PySimpleGUI: A Python GUI Framework*  
  Copyright: © PySimpleGUI, 2024  
  Publisher: PySimpleGUI.org  
  Location: <https://www.pysimplegui.org>  
  Date Accessed: November 26, 2024
* **pandas Documentation**  
  Author: pandas Development Team  
  Title: *pandas: Python Data Analysis Library*  
  Copyright: © pandas Team, 2024  
  Publisher: pandas.pydata.org  
  Location: <https://pandas.pydata.org>  
  Date Accessed: November 26, 2024
* **requests Documentation**  
  Author: Kenneth Reitz and Contributors  
  Title: *Requests: HTTP for Humans*  
  Copyright: © Python Software Foundation, 2024  
  Publisher: requests.readthedocs.io  
  Location: <https://requests.readthedocs.io>  
  Date Accessed: November 26, 2024