**1. Introduction :**

Software Requirements Specification (SRS) for Java Project: Color Chooser

***1.1 Purpose***

The purpose of the Color Chooser project is to develop a Java application that provides users with a graphical interface to dynamically change the background color of a text box based on RGB values. This project aims to serve as an educational tool for understanding GUI programming in Java, specifically using the Swing framework, and for demonstrating color manipulation concepts.

***1.2 Scope***

The Color Chooser application will include the following key features:

* A graphical user interface (GUI) with a text box where users can input text.
* Controls for adjusting RGB values using sliders or input fields.
* Real-time updating of the background color of the text box based on the selected RGB values.
* Options to customize the text within the text box, including font size, style, and color.
* Export functionality to save and share color configurations for later use.

***1.3 Definitions, Acronyms, and Abbreviations***

* RGB: Red-Green-Blue, a color model used for digital displays.
* GUI: Graphical User Interface.
* API: Application Programming Interface.
* SRS: Software Requirements Specification, a document detailing the requirements for a software system.
* IDE: Integrated Development Environment.

***1.4 References***

No specific external references are required beyond standard Java documentation, Swing API documentation, and Eclipse IDE documentation.

**2. Overall Description :**

***2.1 Product Perspective***

The Color Chooser application will be a standalone desktop software tool developed using Java programming language, leveraging Eclipse IDE for development and Swing framework for building the GUI. It will manage color representation using Java's built-in Color class and provide an interactive environment for users to experiment with color combinations effectively.

***2.2 Product Functions***

The core functionalities of the Color Chooser application include:

* Displaying a GUI with a text box where the background color can be dynamically changed based on RGB values.
* Providing sliders or input fields for users to adjust RGB values.
* Real-time updating of the background color of the text box based on the selected RGB values.
* Allowing users to customize the text within the text box, including font size, style, and color.
* Supporting export functionality to save and share color configurations, including selected RGB values and text settings.

***2.3 User Classes and Characteristics***

The primary user classes for the Color Chooser application include:

* Students: Users learning GUI programming in Java who need a practical tool for hands-on exercises in color manipulation.
* Developers: Individuals interested in exploring color theory and experimenting with graphical user interfaces.
* Design enthusiasts: Users looking to visualize and experiment with different color combinations for creative projects.

***2.4 Operating Environment***  
The Color Chooser application will be designed to operate on Java-supported platforms, including Windows, macOS, and Linux. It will require Java Runtime Environment (JRE) installed on the user's system for execution.

***2.5 Design and Implementation Constraints***

* Development shall be carried out using Eclipse IDE due to its robust features for Java development, including debugging and project management capabilities.
* The GUI shall be implemented using Swing framework, chosen for its simplicity, cross-platform compatibility, and integration with Java applications.
* The application architecture should be modular and well-structured to facilitate future enhancements, such as adding additional features or supporting different text and color manipulation options.
* Compatibility with various screen resolutions and operating system versions shall be considered during GUI design to ensure a consistent user experience across different environments.

***2.6 Assumptions and Dependencies***

* It is assumed that users have basic familiarity with operating GUI applications and interacting with graphical elements such as sliders and text boxes.
* The application may depend on standard Java libraries for color management and GUI components provided by the Swing framework.

**3. Specific Requirements :**

***3.1 Functional Requirements***Functional requirements specify the behavior and capabilities expected from the Color Chooser application.

***3.1.1 GUI Design and Interaction***

* FR-1: The application shall display a graphical user interface (GUI) containing a text box and controls for adjusting RGB values.
* FR-2: Users shall be able to input RGB values manually or use sliders to adjust them dynamically.
* FR-3: The background color of the text box shall update in real-time based on the selected RGB values.
* FR-4: Users shall have options to customize the text within the text box, including font size, style, and color.

***3.1.2 Export and Sharing***

* FR-5: The application shall provide functionality to export color configurations, including selected RGB values and text settings.
* FR-6: Exported configurations shall be saved in a format suitable for future use or integration into other applications.

***3.2 Non-Functional Requirements***  
Non-functional requirements specify criteria that can be used to evaluate the operation and performance of the system.

***3.2.1 Performance***

* NFR-1: The application shall respond to user inputs and update the GUI within 100 milliseconds under normal operating conditions.
* NFR-2: It shall handle a minimum of 1000 concurrent users without significant performance degradation.

***3.2.2 Usability***

* NFR-3: The GUI shall be intuitive and user-friendly, requiring minimal training for new users to understand and operate effectively.
* NFR-4: Accessibility features such as keyboard navigation and screen reader compatibility shall be implemented to ensure usability for all users.

***3.2.3 Reliability***

* NFR-5: The application shall operate reliably without crashes or significant errors during normal usage.
* NFR-6: Error handling mechanisms shall be implemented to manage unexpected scenarios and prevent data loss.

***3.2.4 Security***

* NFR-7: The application shall not collect or transmit any user data without explicit consent.
* NFR-8: Measures shall be taken to secure exported color configurations from unauthorized access or modification.

***3.2.5 Compatibility***

* NFR-9: The application shall be compatible with Java Runtime Environment (JRE) versions 8 and above.
* NFR-10: It shall support major operating systems including Windows, macOS, and popular Linux distributions.

***3.3 Interfaces***  
Interfaces describe interactions between the system and its users, as well as with other systems or components.

***3.3.1 User Interfaces***

* UI-1: Main Application Window: Displays the text box with dynamic background color based on selected RGB values and controls for adjusting RGB sliders.
* UI-2: Settings Dialog: Allows users to customize text settings including font size, style, and color.

***3.3.2 Hardware Interfaces***

* HI-1: The application requires standard input devices such as a keyboard and mouse or touchpad for user interaction.

***3.3.3 Software Interfaces***

* SI-1: The application interfaces with Java's Color class and Swing GUI components for managing color representation and user interface elements.

**4. Other Requirements :**

***4.1 Legal and Regulatory Requirements***

There are no specific legal or regulatory requirements applicable to the Color Chooser application beyond standard software development practices and intellectual property considerations.

***4.2 Documentation Requirements***

* DOC-1: A comprehensive user manual shall be provided with the application, detailing its features, usage instructions, and troubleshooting tips.
* DOC-2: Source code documentation shall adhere to JavaDoc standards, providing clear and concise comments for methods and classes.

***4.3 Maintenance and Support Requirements***

* MAINT-1: The application shall include error reporting and logging mechanisms to facilitate troubleshooting and maintenance.
* MAINT-2: Updates and bug fixes shall be provided periodically to address issues and enhance functionality based on user feedback.

**5. Conclusion :**

This Software Requirements Specification (SRS) for the Color Chooser project provides a detailed framework for developing a Java-based application that allows users to interactively change the background color of a text box based on selected RGB values. By adhering to these specifications, the project aims to deliver a user-friendly tool for color experimentation and visualization, suitable for designers, developers, and students alike.