

A REPORT OF ONE MONTH TRAINING
at
Science & Technology Entrepreneurs' Park (STEP-GNDEC), Ludhiana

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE
AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY
(Computer Science and Engineering)



JUNE-JULY ,2014

SUBMITTED BY:
NAME: PAWNI BANSAL
UNIVERSITY ROLL NO. : 2302635

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GURU NANAK DEV ENGINEERING COLLEGE LUDHIANA
(An Autonomous College Under UGC ACT)



SCIENCE & TECHNOLOGY ENTREPRENEURS' PARK
Approved TBI under MSMED
(Promoted by DST, Govt. of India, PSCS & T, Govt. of Punjab & NSET)
GURU NANAK DEV ENGINEERING COLLEGE
AN AUTONOMOUS COLLEGE UNDER UGC ACT - 1956
website : www.stepgndec.com



This is to certify that Mr./Ms.

Rawni Bansal

S/o, D/o Mr. Rohit Bansal of Guru Nanak

Dev Engineering College, Ludhiana

has attended a training / programme / workshop of 04 days/weeks/months in

Web Development conducted by STEP -

from 23/06/2025 to 23/07/2025 at STEP -



Date 11/07/2025 Sr. No. 16246

[Signature]
Rajinder Singh
Adm. Officer Executive Director

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

CANDIDATE'S DECLARATION

I PAWNI BANSAL hereby declare that I have undertaken one month training at STEP Guru Nanak Dev Engineering College during the period from 23 June 2025 to 22 July 2025 in partial fulfillment of requirements for the award of degree of B.Tech. (Computer Science and Engineering) at Guru Nanak Dev Engineering College, Ludhiana. The work which is being presented in the training report submitted to the Department of Computer Science and Engineering at GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA is an authentic record of training work.

PAWNI BANSAL

Signature of the Student

The one month industrial training Viva–Voce Examination of _____ has been held on _____ and accepted.

Signature of Internal Examiner

Signature of External Examiner

ABSTRACT

This report presents the work carried out during a one-month industrial training program at the Science & Technology Entrepreneurs' Park (STEP-GNDEC), Ludhiana, undertaken as a mandatory component of the academic curriculum. The primary objective of the training was to bridge the gap between theoretical concepts learned in the classroom and their practical implementation in real-world scenarios. The program was designed to provide in-depth hands-on experience in modern web development technologies, enabling the development of technical competence and professional skills.

The project aims to provide users with an interactive, intuitive, and user-friendly tool for generating customized background gradients that can be efficiently utilized in websites, web applications, and graphical user interfaces. The application is designed to assist designers, developers, and beginners by simplifying the process of creating visually attractive background designs without requiring advanced graphic design skills or extensive coding knowledge. The primary objective of the application is to streamline the background design process while offering real-time visual feedback. Users can dynamically adjust color inputs and immediately observe changes in the gradient output, allowing them to evaluate and refine their designs instantly. This real-time interaction not only improves usability but also enhances the user experience by making the design process faster, more efficient, and engaging.

The Background Generator allows users to choose two different colors through intuitive and user-friendly input controls, enabling them to instantly preview the resulting gradient effect on the screen without any delay or page refresh. As users modify the selected colors, the gradient background updates dynamically in real time, providing immediate visual feedback and enhancing the overall interactive experience. In addition to displaying the visual output, the application automatically generates and presents the corresponding CSS gradient code in a readable and structured format. This code can be easily copied and directly integrated into personal, academic, or professional web projects, eliminating the need for manual coding of complex gradient styles. By combining real-time visualization with automatic code generation, the tool effectively bridges the gap between design creativity and code reusability, making it especially useful for beginners, web developers, and designers who seek both aesthetic flexibility and practical implementation efficiency.

Overall, the project significantly contributed to strengthening foundational web development skills by providing practical exposure to the complete process of designing and implementing a functional web application. The implementation of real-time features enhanced proficiency in DOM manipulation, event handling, and dynamic styling, while the focus on responsive design improved awareness of creating adaptable user interfaces. In addition to technical skill development, the project resulted in a visually appealing and user-centric tool that addresses real-world design requirements, making it both practically useful and academically relevant.

ACKNOWLEDGMENT

I would like to express my sincere gratitude to Science & Technology Entrepreneurs' Park (STEP-GNDEC), Ludhiana, for giving me the opportunity to undergo one month of industrial training and gain practical exposure in the field of web development. This training was an enriching experience that helped bridge the gap between academic knowledge and industry practices.

I am especially thankful to my mentors and instructors at Science & Technology Entrepreneurs' Park for their constant guidance, support, and encouragement throughout the training period. Their deep expertise, valuable feedback, and professional insights played a key role in helping me understand the core concepts of web development and successfully complete my project titled "Background Generator"

I would also like to convey my heartfelt thanks to the Department of Computer Science and Engineering, Guru Nanak Dev Engineering College, Ludhiana, for incorporating industrial training into the academic curriculum and providing continuous academic support. I am grateful to my training coordinator and faculty advisors for their guidance and encouragement throughout the training and report preparation.

A special thanks to my peers and friends who collaborated and shared knowledge during the training phase. Their presence made the learning environment more enjoyable and effective.

Last but not the least, I am deeply thankful to my parents and family members for their unwavering support, patience, and motivation during the training period and throughout my academic journey.

ABOUT THE INSTITUTE

Science & Technology Entrepreneurs' Park (STEP-GNDEC)

Gill Road, Gill Park

Ludhiana-141006

Contact No: **78371-00954, 97817-46007, 78371-00924**

Science & Technology Entrepreneurs' Park (STEP-GNDEC), Ludhiana is a unique and dynamic institution established to promote innovation, entrepreneurship, and technological development among students and professionals in the fields of Science and Technology. It is jointly promoted by Guru Nanak Dev Engineering College (GNDEC), Ludhiana, and the Department of Science & Technology (DST), Government of India, reflecting a strong partnership between academia and national policy initiatives aimed at fostering entrepreneurial growth.

STEP-GNDEC provides a supportive ecosystem that encourages creative thinking, innovation, and research-driven development. The park offers modern infrastructure, incubation facilities, mentoring support, and access to technical resources, enabling aspiring entrepreneurs to convert innovative ideas into market-ready products and sustainable business ventures. Through structured training programs, workshops, and industry-oriented projects, STEP-GNDEC bridges the gap between theoretical learning and practical implementation.

Vision

To become a leading centre of excellence in entrepreneurship and innovation, nurturing technically skilled and self-reliant individuals who contribute to economic growth through technology-driven startups and enterprises.

Mission

- To foster an entrepreneurial mindset among Science and Technology students.
- To provide a conducive environment for innovation, research, and product development.

Overall, STEP-GNDEC plays a pivotal role in developing entrepreneurial talent by creating an ecosystem that supports innovation, skill development, and enterprise creation, thereby contributing to national development and technological progress.

LIST OF FIGURES

FIGURE CAPTION	Page No.
Figure 3.3.1. Home Page.....	24
Figure 3.3.2. Select Direction of the gradient.....	25
Figure 3.3.3. Select Color.....	26
Figure 3.3.4. Refresh color and copy code.....	27

LIST OF TABLES

Figure 3.2 Challenges and Resolutions.....	28
--	----

CONTENTS

Certificate by Company/Industry/Institute	i
Candidate's Declaration	ii
Abstract	iii
Acknowledgement	iv
About the Company/Industry/Institute	v
List of Figures	vi
List of Tables	vii
CHAPTER 1: INTRODUCTION	10
1.1 Overview of the project	10
1.2 Purpose	11
1.3 Technologies used	11
1.4 Working of the project	11
1.4.1 Step by step working of the application	12
1.4.2 Algorithm	13
1.4.3 Pseudocode	13
1.5 Scope of the project	14
1.6 Advantages and applications	15
1.7 Limitations	16
CHAPTER 2: TRAINING WORK UNDERTAKEN	17
2.1 Introduction	17
2.2 Week 1 – Introduction to Web Development and HTML	17-18
2.3 Week 2 – Styling with CSS	18-19
2.4 Week 3 – Introduction to Javascript and Interactivity.....	20-21

2.5 Week 4 – Project Development and Deployment.....	21-22
CHAPTER 3: RESULTS AND DISCUSSION 23	
3.1 Overview of the Project Output	23
3.2 Key Functionalities of Background Generator.....	23-24
3.3 Screenshots and explanations	24-27
3.4 Performance and usability discussion	27-28
3.5 Challenges faced and Resolution	28
CHAPTER 4: CONCLUSION AND FUTURE SCOPE 29	
4.1 Conclusion	29
4.2 Scope and future enhancements.....	29-30
REFERENCES	31
APPENDIX	32-39

CHAPTER 1

INTRODUCTION

1.1 Overview of the Project

The Background Generator is an interactive, web-based application developed to generate dynamic and visually appealing background gradients using core front-end web technologies such as HTML, CSS, and JavaScript. The project was undertaken as a part of the industrial training program with the primary objective of transforming theoretical concepts into practical implementation in a real-world web development environment. It reflects a hands-on approach to learning modern front-end development practices while addressing a commonly encountered design requirement in web applications.

The application is designed to serve a dual purpose. Firstly, it functions as a practical utility tool for web developers and designers who require quick and efficient background designs without spending time manually writing or testing CSS gradient code. Secondly, it acts as an educational platform for beginners, allowing them to gain a deeper understanding of front-end development concepts through interactive usage and real-time feedback. By visualizing the immediate impact of code-driven styling changes, learners can better grasp the relationship between design elements and programming logic.

1.2 Purpose

In modern web development, visual design and user interface aesthetics play a crucial role in enhancing user engagement and overall user experience. Background gradients are extensively used in websites and web applications to create visually attractive layouts, convey brand identity, and improve content readability. However, manually designing CSS gradients often involves dealing with complex syntax, color selection, and trial-and-error adjustments, which can be time-consuming and challenging, especially for beginners and novice developers.

The primary motivation behind developing the Background Generator is to simplify the gradient creation process by providing an automated and interactive solution that eliminates the need for manual coding of complex CSS gradient properties. The tool allows users to generate visually appealing gradients instantly, thereby saving development time and reducing effort. This automation encourages experimentation with various color combinations, enabling users to explore creative design possibilities in an efficient and intuitive manner. Additionally, the project is motivated by the need to provide a practical and hands-on learning experience for aspiring front-end developers. The application demonstrates how HTML structure, CSS styling, and JavaScript logic work together in a dynamic and interactive environment. By observing real-time updates triggered by user actions, learners gain a clear understanding of how UI components respond to events and how design changes are applied programmatically. Overall, the project aims to combine functionality with learning, promoting creativity, improving technical skills, and helping beginners build confidence in developing interactive web applications.

1.3 Technologies Used

The Background Generator project has been developed using a combination of core front-end web technologies and modern deployment tools to ensure optimal performance, responsiveness, and ease of accessibility. Each technology plays a specific and important role in the overall development of the application.

HTML (HyperText Markup Language) is used to define the basic structure of the web application. It provides the framework for organizing content and creating user interface elements such as buttons, color input fields, and display sections for generated CSS code. By using semantic HTML elements, the application maintains a clear and well-structured layout, which also contributes to better accessibility and maintainability.

CSS (Cascading Style Sheets) is responsible for styling the application and enhancing its visual appearance. It is used to design the layout, define typography, and implement dynamic background gradient effects. CSS ensures that the user interface remains clean, visually appealing, and responsive across different screen sizes. The use of modern CSS properties enables smooth visual transitions and consistent styling throughout the application.

JavaScript serves as the core programming language that enables interactivity and dynamic behavior within the application. It is used to handle user events such as button clicks and color selection, manipulate the Document Object Model (DOM), and update the background gradients in real time. JavaScript also generates the corresponding CSS gradient code dynamically, which is displayed to the user for easy reuse. This real-time interaction highlights the use of event-driven programming and enhances the user experience.

For deployment, the application is hosted on the Vercel Hosting Platform, which provides a reliable and scalable environment for modern web applications. Vercel enables seamless deployment and ensures that the project is accessible from any device with an internet connection and a web browser. Hosting the project online also allows users to interact with the application remotely, making it more practical and widely usable.

Together, these technologies ensure that the application is responsive, cross-browser compatible, and aligned with modern web development standards. The effective integration of HTML, CSS, JavaScript, and Vercel demonstrates the practical application of front-end development skills gained during the industrial training program.

1.4 Working of the Project

The working of the Background Generator application is primarily based on user interaction and real-time processing using JavaScript. The application responds to user actions such as clicking buttons or selecting color inputs through JavaScript event listeners. These event listeners continuously monitor user input and trigger specific functions whenever an interaction occurs. Upon activation, the JavaScript logic processes the input values and dynamically generates color combinations used to form gradient backgrounds.

Once the color values are generated or selected, JavaScript applies these values to the background of the web page by updating the corresponding CSS gradient properties. This

dynamic styling ensures that the background changes instantly, providing a smooth and interactive experience. The application does not require page refresh or reload, which enhances performance and usability while maintaining uninterrupted user interaction.

The application makes extensive use of Document Object Model (DOM) manipulation techniques to modify both the visual appearance and the displayed content in real time. Along with updating the background gradient, the application also dynamically generates the CSS gradient code based on the selected colors. This code is displayed on the user interface in a structured and readable format, allowing users to clearly understand the syntax and logic behind CSS gradients.

The real-time preview functionality enables users to immediately visualize the impact of their color choices, helping them experiment efficiently and refine their designs. This feature also serves an educational purpose by allowing learners to observe how design-related CSS properties are updated programmatically. The seamless coordination between HTML elements for structure, CSS for styling, and JavaScript for logic and interactivity effectively demonstrates the principle of event-driven programming.

The real-time preview enables users to instantly visualize the effect of their selected colors, encouraging experimentation and creativity. The interaction between HTML for structure, CSS for styling, and JavaScript for logic and interactivity clearly demonstrates the concept of event-driven programming and highlights how modern front-end web applications function.

Overall, the working of the Background Generator highlights how front-end technologies collaborate to create interactive, responsive, and user-friendly web applications, showcasing practical implementation of fundamental web development concepts learned during the training period.

1.4.1 Step-by-Step Working of the Application

The working of the Background Generator can be explained step by step as follows:

1. The user opens the Background Generator web application in a browser.
2. The application loads the HTML structure and applies default styling using CSS.
3. The user clicks the “Generate Background” button or selects color inputs.
4. JavaScript event listeners detect the user action.
5. JavaScript generates or retrieves color values.
6. A CSS gradient string is dynamically created using the selected colors.
7. The generated gradient is applied to the background of the webpage.
8. Simultaneously, the corresponding CSS gradient code is displayed on the screen.

9. The user can copy the code and reuse it in other projects.
10. The process repeats every time the user interacts with the interface.

1.4.2 Algorithm: Background Generator

1. Start
2. Load HTML elements and CSS styles
3. Attach event listener to button or color input
4. When event occurs:
 - o Generate or fetch color values
 - o Create CSS gradient property
 - o Apply gradient to page background
 - o Display CSS code on screen
5. End

1.4.3 Pseudocode

Start

Select button and input elements

Add event listener to button

On button click:

 Generate two colors

 Create CSS gradient string

 Apply gradient to background

 Display CSS code

End

1.5 Scope of the Project

The scope of the Background Generator project focuses on delivering a simple, intuitive, and efficient solution for generating dynamic background gradients for web-based applications, websites, and user interface components. The primary objective of the tool is to enable users to create visually appealing and modern background effects with minimal effort, eliminating the need for advanced graphic design software or complex coding techniques. By providing an interactive and responsive interface, the application allows users to experiment with different color combinations and instantly preview the results, thereby enhancing productivity and creativity in web design.

The project is designed to cater to a broad spectrum of users, including students, beginners in front-end development, professional web designers, and software developers. For beginners and students, the application serves as an accessible entry point into front-end development, helping them understand how design elements interact with underlying code. For professionals, it acts as a quick and efficient utility for generating gradient backgrounds that can be directly integrated into real-world projects, saving time during the design and development process.

From an educational perspective, the project offers significant scope for learning and applying fundamental front-end development concepts. It provides practical exposure to HTML for structure, CSS for styling and gradient generation, and JavaScript for interactivity and dynamic behavior. Key concepts such as Document Object Model (DOM) manipulation, event handling, real-time UI updates, and responsive design are reinforced through hands-on usage. This experiential learning approach helps bridge the gap between theoretical knowledge and real-world implementation, making the project especially valuable in academic environments.

Additionally, the project has strong potential for scalability and future enhancement. Advanced features such as gradient presets, angle and direction controls, color history tracking, export options for CSS code, theme saving, and integration with design frameworks can be incorporated to expand its functionality. With such enhancements, the application can evolve into a comprehensive web design utility suitable for professional and large-scale projects.

The deployment of the application on modern cloud platforms such as Vercel further extends its scope by ensuring high availability, cross-device accessibility, and ease of maintenance. Online deployment enables users to access the tool from anywhere without installation, encouraging remote usage, collaboration, and real-time design experimentation.

Overall, the Background Generator project serves a dual purpose. It functions as a practical tool for creating aesthetically pleasing web designs while simultaneously acting as an educational resource for learning modern front-end development practices. Its simplicity, flexibility, and potential for expansion make it a valuable addition to the skillset of aspiring developers and designers, while also laying the foundation for future development into a more advanced and professional-grade application.

1.6 Advantages and Applications

The Background Generator project offers several advantages that make it a valuable tool for both learning and practical web development:

Advantages:

1. Time Efficiency: The application eliminates the need for manually writing complex CSS gradient code, allowing developers to create backgrounds quickly and efficiently.
2. Ease of Use: The interface is intuitive and beginner-friendly, enabling users with minimal coding experience to generate gradients effortlessly.
3. Real-Time Preview: Users can instantly visualize the effect of their selected colors, which enhances experimentation and helps refine design choices.
4. Educational Value: The tool provides a hands-on learning experience, helping beginners understand CSS gradients, JavaScript DOM manipulation, and event-driven programming.
5. Cross-Platform Accessibility: Hosted online using platforms like Vercel, the application can be accessed from any device with an internet connection and a web browser.
6. Reusable Output: The generated CSS code can be copied and reused in personal, academic, or professional web projects, bridging the gap between design and implementation.

Applications:

1. Web Development: Designers and developers can quickly generate visually appealing gradient backgrounds for websites, applications, or UI components.
2. Educational Tool: It serves as a practical learning platform for students and beginners to understand front-end concepts, coding logic, and real-time updates.
3. Design Experimentation: Users can experiment with various color combinations and gradients, which aids in creative decision-making for web design projects.
4. Prototyping: The tool can be used to rapidly prototype user interface backgrounds in web applications without requiring advanced design software.

5. Integration with Projects: Generated CSS code can be directly integrated into existing or new projects, reducing development effort and promoting productivity.

1.7 Limitations

Despite its usefulness, the Background Generator project has certain limitations that can be addressed in future versions:

1. Limited Gradient Options: The current version supports basic two-color linear gradients only. Advanced options like multi-color gradients, radial gradients, or animated gradients are not included.
2. Customization Constraints: Features such as angle adjustment, gradient presets, or color history are not available in the current implementation.
3. Dependent on Browser: The tool requires a modern web browser to function correctly; older browsers may not support some CSS or JavaScript features.
4. Internet Requirement (for hosted version): While the application can be run locally, the online hosted version depends on a stable internet connection.
5. Basic User Interface: The design is simple and minimalist, which may limit its appeal for professional designers seeking more advanced styling options.
6. No Offline Export: The tool currently does not provide direct options to download the gradient as an image file; only CSS code is available for reuse.

These limitations, however, provide a clear path for future enhancements. Adding advanced features, more gradient options, export capabilities, and improved UI design can make the tool even more powerful and versatile.

CHAPTER 2

TRAINING WORK UNDERTAKEN

2.1 Introduction

The one-month industrial training at Science & Technology Entrepreneurs' Park (STEP-GNDEC), Ludhiana was aimed at providing practical exposure to front-end web development. The training program was structured to bridge the gap between academic knowledge and real-world applications by allowing students to work on a hands-on project, in this case, the Background Generator web application.

The primary focus of the training was to enhance understanding of HTML, CSS, and JavaScript, including responsive design principles, event-driven programming, and DOM manipulation. The training was designed in a weekly progression, starting with basic concepts and gradually moving towards project implementation and deployment. This structured approach ensured a clear learning path while enabling participants to apply their skills in practical scenarios.

2.2 Week 1: Introduction to Web Development and HTML

Objectives:

- To gain a clear understanding of the basics of web development and the role of different web technologies in building interactive websites.
- To learn the fundamentals of HTML, including its syntax, structure, and semantic elements, which are essential for creating well-organized web pages.
- To create simple web pages that incorporate headings, paragraphs, lists, hyperlinks, images, multimedia, and forms, forming the foundation for more advanced web development.

Activities Undertaken:

- Attended interactive sessions that introduced the fundamental concepts of web development, including the client-server architecture, HTTP requests and responses, and the role of web browsers in rendering web pages.
- Studied HTML5 tags and their proper usage for structuring web content, including semantic elements such as `<header>`, `<footer>`, `<section>`, `<article>`, and `<nav>` for better accessibility and SEO optimization.

- Practiced creating basic web pages from scratch, learning to organize content using headings (`<h1>`-`<h6>`), paragraphs (`<p>`), ordered and unordered lists (`` / ``), hyperlinks (`<a>`), and multimedia elements like images (``), audio (`<audio>`), and video (`<video>`).
- Explored HTML forms and input elements such as text fields, radio buttons, checkboxes, dropdowns, and buttons. Learned form validation techniques using built-in HTML attributes to ensure correct and meaningful user input.
- Completed hands-on exercises to reinforce learning, including small mini-projects like a personal profile page, contact form, and a simple static webpage.

Outcome:

- Developed a strong foundation in HTML, understanding both the structural and semantic aspects of web page creation.
- Gained practical skills in organizing content, embedding multimedia, and designing basic interactive forms.
- Prepared the groundwork for further learning, particularly in CSS styling for visual enhancement and JavaScript for adding interactivity.
- Acquired confidence in using HTML as a fundamental building block for real-world web development projects, including the Background Generator project undertaken in the subsequent weeks.

2.3 Week 2: Styling with CSS

Objectives:

- To understand the importance of CSS (Cascading Style Sheets) in enhancing the visual appeal and usability of web pages.
- To learn various styling techniques, layouts, and design principles, enabling the creation of attractive and organized web interfaces.
- To apply colors, fonts, backgrounds, borders, spacing, and CSS gradients effectively, ensuring consistent styling and improved user experience.
- To develop skills in responsive design, making web pages adaptable to different screen sizes and devices.

Activities Undertaken:

- Studied CSS syntax, selectors, and properties, including class, ID, pseudo-class, and pseudo-element selectors, along with specificity rules for conflict resolution.
- Applied text formatting techniques, including font styles, sizes, colors, and text alignment, to make content more readable and visually appealing.
- Learned to style elements using borders, padding, margins, and box model concepts, ensuring proper spacing and layout.
- Explored background properties, including solid colors, images, and CSS gradients, which later became a key component in the Background Generator project.
- Practiced responsive web design using media queries, flexible grids, and fluid layouts to ensure compatibility across desktops, tablets, and mobile devices.
- Implemented interactive features such as hover effects, transitions, and animations to enhance user engagement and provide smooth visual feedback.
- Created mini-projects, such as styled profile pages and interactive forms, integrating all CSS concepts learned in the week.

Outcome:

- Developed the ability to create visually appealing and well-structured web pages that adhere to modern web design standards.
- Gained practical experience in CSS gradients, a foundational skill directly applied in the development of the Background Generator project.
- Acquired knowledge of responsive design, ensuring that web pages can adapt seamlessly to different devices and screen resolutions.
- Strengthened understanding of styling principles and best practices, which is essential for professional front-end development.
- Prepared to integrate interactivity with JavaScript, combining visual design with dynamic functionality in real-world projects.

2.4 Week 3: Introduction to JavaScript and Interactivity

Objectives:

- To understand the basics of JavaScript and its pivotal role in adding interactivity to web pages.
- To learn about variables, data types, operators, loops, functions, and conditional statements, forming the foundation of programming logic in web development.
- To explore DOM (Document Object Model) manipulation for dynamically updating HTML content and CSS styles.
- To implement interactive web elements, allowing web pages to respond to user actions and provide dynamic content updates.

Activities Undertaken:

- Studied JavaScript fundamentals, including variables, constants, arrays, objects, operators, loops, and conditional statements, to understand programming logic.
- Practiced creating functions and reusable code blocks to perform specific tasks efficiently within web pages.
- Learned to manipulate the DOM, modifying HTML elements, adding or removing nodes, and dynamically updating CSS properties in response to user input.
- Developed interactive mini-projects such as changing text content, updating colors, modifying styles, and toggling visibility based on user actions.
- Applied JavaScript specifically to dynamically update background gradients and generate corresponding CSS code, reinforcing the concepts used in the Background Generator project.
- Debugged and tested scripts to ensure smooth functionality, error handling, and consistent performance across different web browsers.

Outcome:

- Gained hands-on experience in adding interactivity to static web pages, making them dynamic and user-friendly.
- Learned to integrate HTML, CSS, and JavaScript, understanding how these technologies work together to create fully functional web applications.
- Developed problem-solving skills by writing JavaScript functions and handling events effectively.
- Built confidence in using JavaScript to manipulate the DOM and update web content in real time, preparing for larger projects like the Background Generator.
- Strengthened understanding of event-driven programming, a critical concept in modern front-end development.

2.5 Week 4: Project Development and Deployment

Objectives:

- To apply the knowledge and skills acquired in HTML, CSS, and JavaScript to develop a real-world web application.
- To create a fully functional Background Generator application that allows users to generate dynamic CSS gradients with ease.
- To learn the deployment process, hosting the application online to make it accessible from any device with an internet connection.
- To gain practical experience in project planning, development, testing, and deployment, simulating a professional web development workflow.

Activities Undertaken:

- User Interface Design: Developed the interface using HTML and CSS, focusing on simplicity, user-friendliness, and visual appeal. Ensured that the layout was responsive and compatible across different devices and screen sizes.
- Core Functionality Implementation: Used JavaScript to implement the main features of the Background Generator. This included color selection, gradient generation, dynamic application of CSS gradients to the background, and real-time generation of

corresponding CSS code.

- Testing and Debugging: Performed extensive testing for cross-browser compatibility and responsiveness on desktops, tablets, and mobile devices. Debugged JavaScript errors and optimized CSS and HTML structure to ensure smooth functionality.
- Deployment on Vercel: Learned how to host the project on the Vercel platform, making it accessible online. Understood deployment workflows, URL management, and how to update projects after deployment.
- Final Touches: Added enhancements such as real-time preview, dynamic display of generated CSS code, and a minimalistic, intuitive user interface. Focused on improving user experience by ensuring smooth interactions, clean design, and proper alignment of elements.
- Documentation: Documented project features, working, and deployment process for future reference and for inclusion in the training report.

Outcome:

- Successfully developed and deployed the Background Generator application, which is fully functional and accessible online.
- Applied theoretical knowledge in a practical scenario, enhancing technical, problem-solving, and coding skills.
- Gained hands-on experience in end-to-end project development, including planning, design, implementation, testing, debugging, and deployment.
- Learned best practices for responsive design, cross-browser compatibility, and dynamic interactivity, which are critical skills for professional front-end development.
- Strengthened confidence in building real-world web applications and deploying them for public accessibility, bridging the gap between academic learning and practical web development.

CHAPTER 3

RESULTS AND DISCUSSIONS

3.1 Overview of the Project Output

The final output of the project is a fully functional web application that allows users to:

- Select or generate two colors to create a gradient background.
- Preview the gradient in real-time on the application interface.
- View and copy the corresponding CSS gradient code for direct integration into web projects.
- Interact with a minimalistic and intuitive user interface, providing a seamless user experience.

The application has been designed to be responsive, ensuring that it works efficiently across different devices and screen sizes. It demonstrates a practical application of front-end web development concepts, combining structure (HTML), styling (CSS), and interactivity (JavaScript) to achieve a functional and visually appealing tool.

3.2 Key Functionalities of the Background Generator

The main functionalities of the project include:

1. Dynamic Gradient Generation:
 - Users can select two colors manually or generate random colors.
 - The application immediately applies the chosen colors as a CSS linear gradient to the background.
2. Real-Time Preview:
 - Changes in colors are reflected instantly, allowing users to see the effect without reloading the page.
3. CSS Code Display and Copy Feature:
 - The corresponding CSS code for the generated gradient is dynamically displayed.
 - Users can copy the code for direct use in personal or professional projects.

4. Responsive User Interface:

- The interface is clean, minimalist, and intuitive, ensuring ease of use for beginners and professionals alike.
- Works seamlessly on desktops, tablets, and mobile devices.

5. Event-Driven Functionality:

- Uses JavaScript event listeners to detect user actions such as button clicks or color selection.
- Dynamically updates the DOM and applies styles without page reload.

3.3 Screenshots and Explanation

Screenshot 1: Home Page Interface

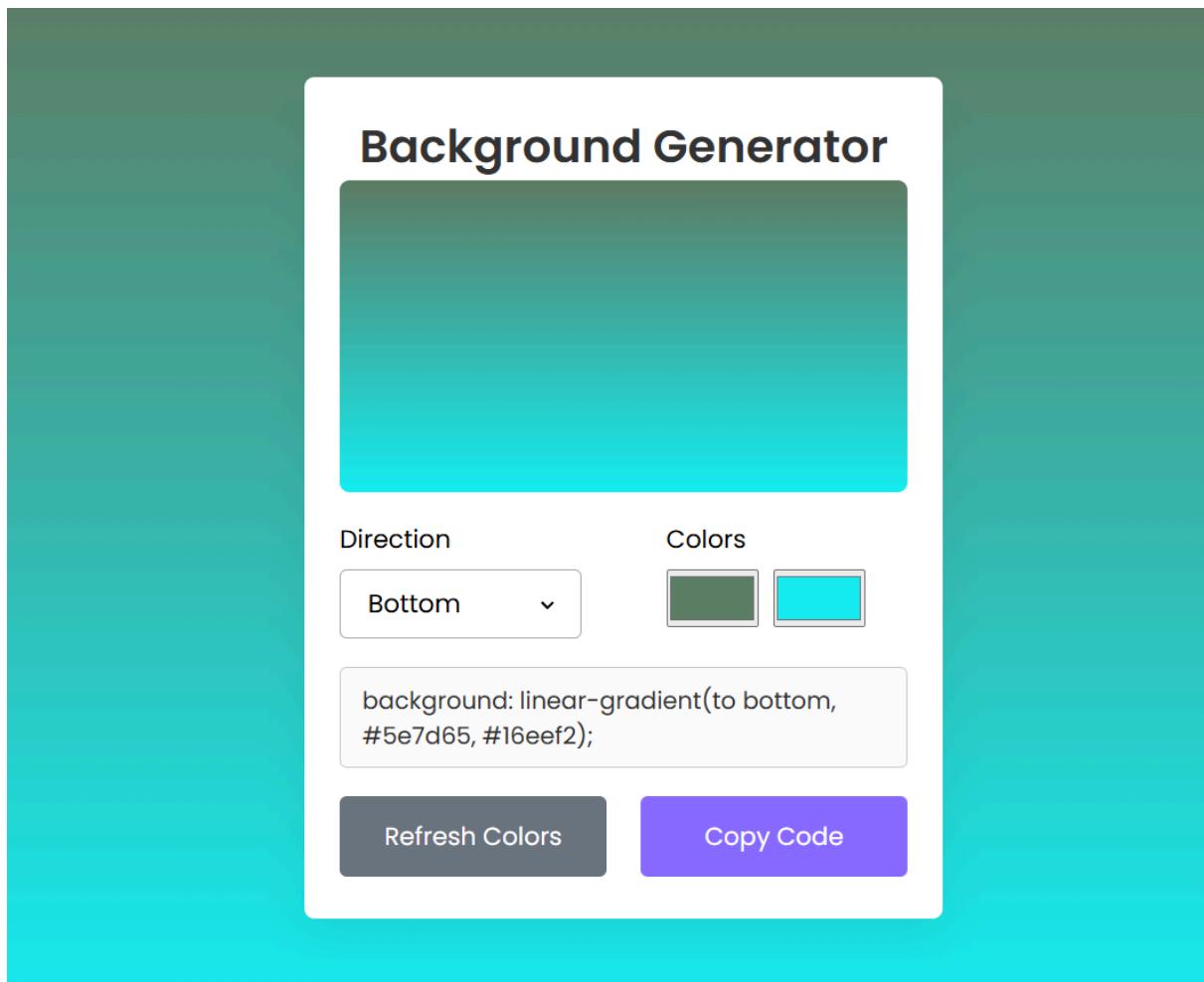


Figure 3.3.1

- Displays the main layout, including color selection inputs, “Generate Background” button, and the gradient preview area.
- Shows a minimalist design to enhance user focus on gradient creation.

Screenshot 2: Select Direction of the gradient

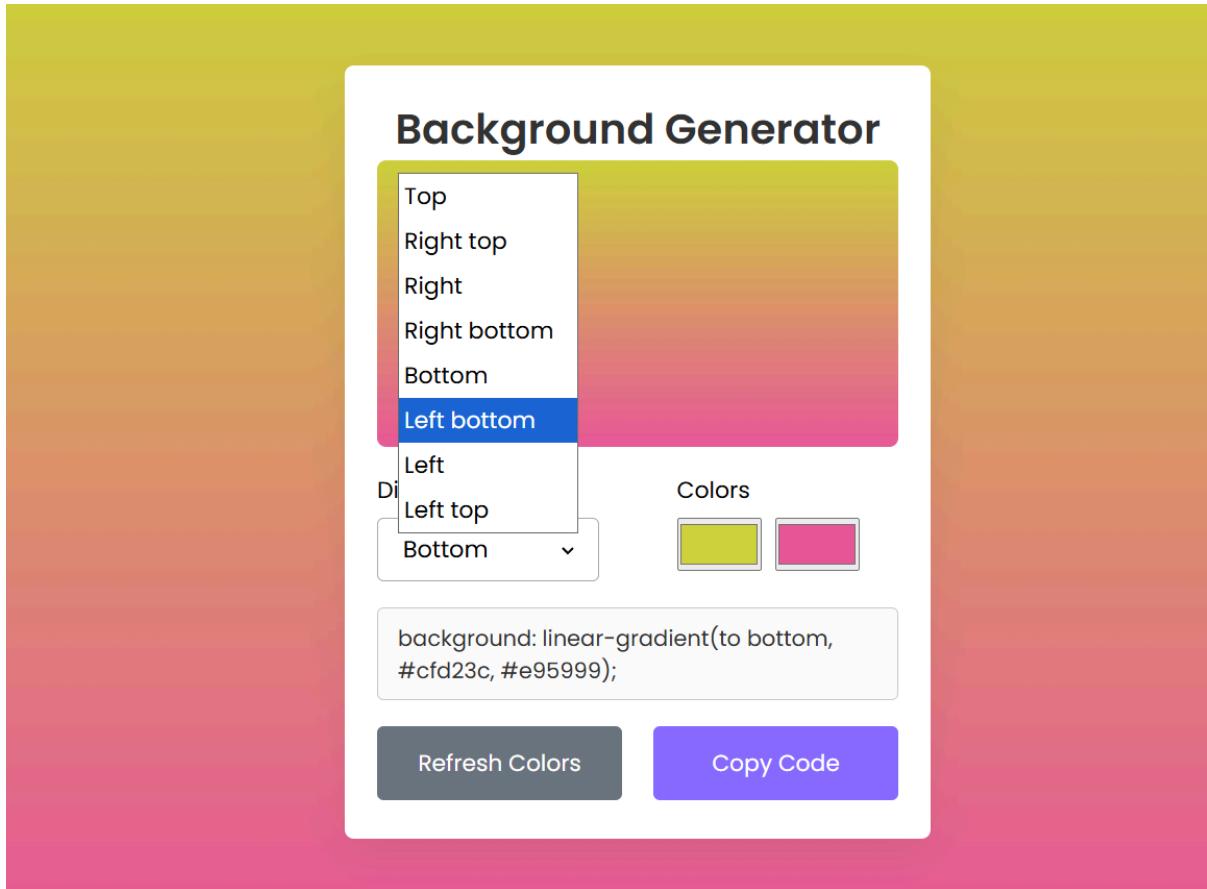


Figure 3.3.2

- Users can choose from predefined gradient directions, such as top-to-bottom, bottom-to-top, left-to-right, right-to-left, diagonal.
- A custom angle input allows users to specify any degree of rotation for the gradient, providing complete control over its visual orientation.
- Once the direction is selected, the gradient background **updates immediately** without reloading the page.
- The **CSS linear-gradient property** is dynamically updated to include the selected direction along with the chosen colors.
- The DOM is updated dynamically to apply the new gradient direction, showcasing the use of **event-driven programming** and **real-time DOM manipulation**.
- By allowing direction selection, the feature encourages users to **experiment with different gradient angles**, producing varied visual effects.

Screenshot 3: Select Color

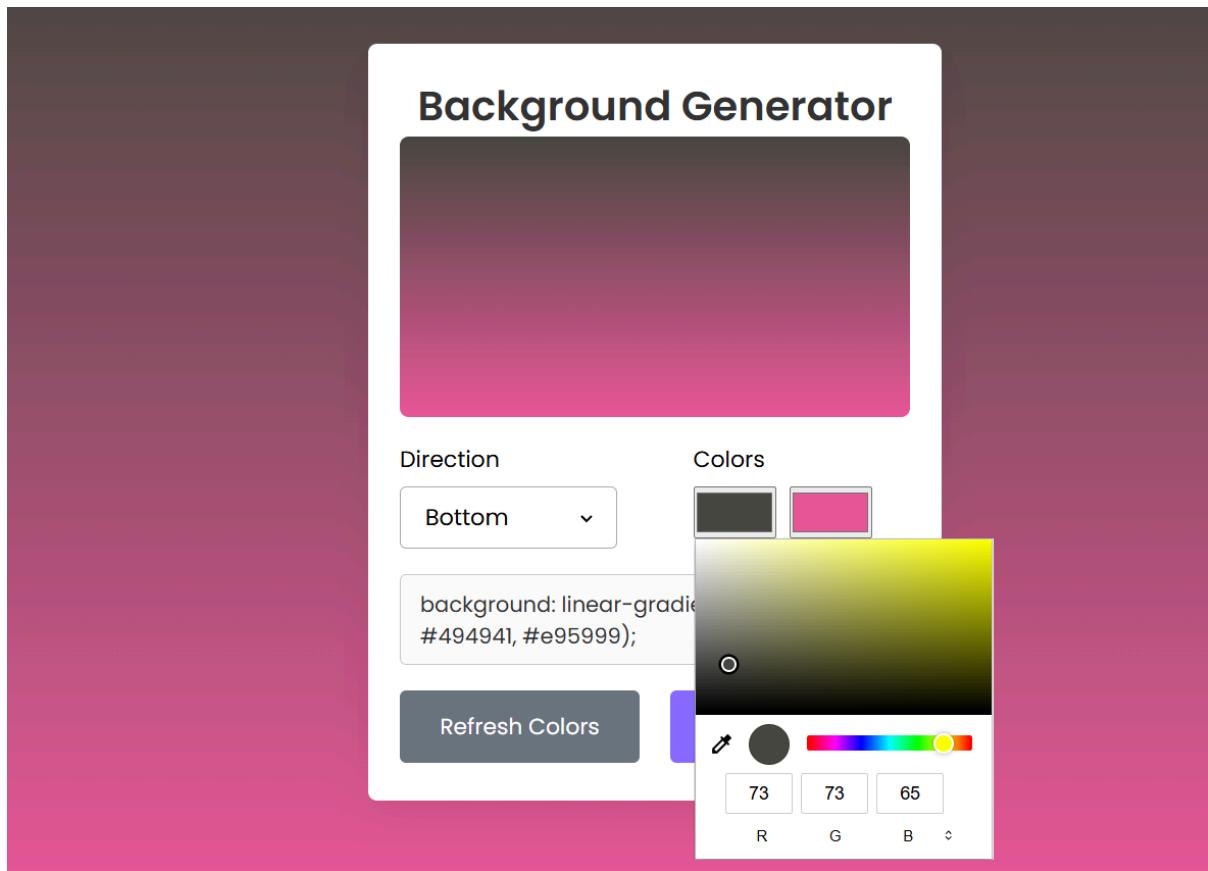


Figure 3.3.3

- Users can select two colors using color picker inputs.
- The interface displays a visual representation of the selected colors, making it easy to understand the color choices and their effect on the gradient.
- Once the colors are selected, the background gradient updates instantly to reflect the chosen combination.
- Some implementations include a “Random Color” button, which automatically generates two random colors.
- Users can copy this code for use in other web projects, ensuring the gradient displayed in the preview matches exactly with the applied CSS.
- When a color is selected or changed, the DOM is dynamically updated to apply the new gradient, demonstrating event-driven programming and real-time DOM manipulation.
- This feature teaches users how HTML color input elements, CSS gradient properties, and JavaScript DOM manipulation work together to create interactive web components.
- The color selection feature allows users to experiment with endless color combinations, enhancing creativity.

Screenshot 4: Refresh color and Copy Code

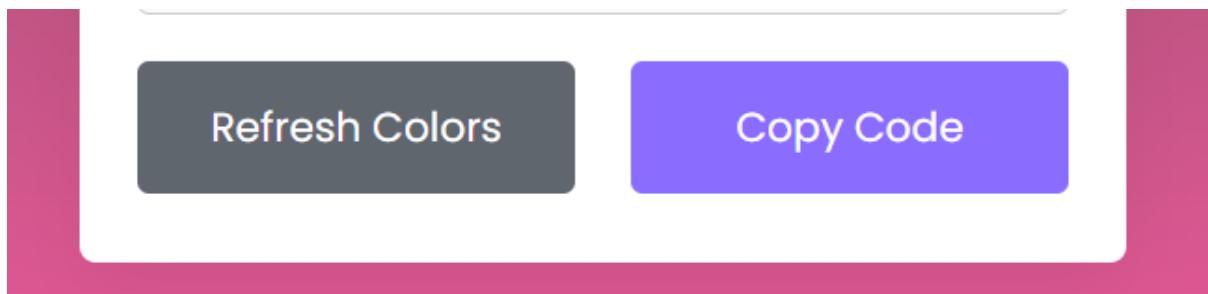


Figure 3.3.4

- The “Refresh Color” button allows users to automatically generate new random color combinations for the gradient.
- When the refresh button is clicked, the background gradient updates instantly to reflect the new colors.
- The “Copy Code” button allows users to copy the generated CSS code to the clipboard with a single click.
- The DOM is dynamically updated to apply the new gradient and update the CSS code display, demonstrating event-driven programming and interactive design.
- These features improve user productivity by allowing quick experimentation with multiple color combinations.
- Users learn how JavaScript can automate UI updates, dynamically generate CSS, and handle clipboard operations.

3.4 Performance and Usability Discussion

The Background Generator application demonstrates efficient performance and high usability due to its lightweight architecture and optimized use of front-end web technologies. Since the project is developed using HTML, CSS, and JavaScript, all operations are executed on the client side, eliminating the need for server-side processing or database interaction. This results in fast load times, minimal resource consumption, and smooth user interaction.

From a performance perspective, the application responds instantly to user actions such as color selection, direction changes, and refresh operations. The use of JavaScript event listeners and DOM manipulation ensures that updates occur in real time without reloading the page. This real-time behavior enhances responsiveness and creates a seamless user experience. Additionally, the absence of heavy libraries or frameworks keeps the application lightweight and improves compatibility across different web browsers.

The application has been tested on multiple devices and browsers, including desktops, tablets, and smartphones. The implementation of responsive design techniques ensures that the layout adjusts smoothly to different screen sizes, maintaining functionality and visual consistency. The deployment on the Vercel hosting platform further contributes to performance reliability by providing stable hosting and fast content delivery.

From a usability standpoint, the application is designed with a clean, minimalist, and intuitive interface. The placement of controls such as color pickers, buttons, and code display areas is logical and easy to understand, reducing the learning curve for first-time users. Clear visual feedback, such as instant gradient updates and CSS code display, helps users quickly understand the effect of their actions.

The copy code feature significantly enhances usability by allowing users to directly reuse the generated CSS gradient code without manual effort. Similarly, the refresh color feature encourages creative experimentation by generating new combinations instantly. These features improve productivity and user satisfaction.

Overall, the Background Generator achieves a balance between high performance and user-friendly design. Its fast response time, intuitive controls, and real-time feedback make it suitable for both beginners learning front-end development and professionals seeking a quick design utility. The application effectively demonstrates how well-structured front-end code can deliver efficient, interactive, and accessible web applications.

3.5 Challenges Faced and Resolutions

S.No	Challenge Faced	Description of the Challenge	Resolution / Solution Implemented
1.	Understanding CSS Gradients	Initially, understanding the syntax and working of CSS linear gradients, including direction and color combinations, was challenging.	Studied CSS gradient documentation, practiced multiple examples, and implemented small test cases to gain clarity.
2.	Real-Time Background Update	Applying gradient changes dynamically without page reload required proper DOM manipulation techniques.	Used JavaScript event listeners and style property updates to apply changes instantly.
3.	JavaScript DOM Manipulation	Manipulating HTML elements dynamically using JavaScript was complex during initial implementation.	Practiced DOM selection methods and event handling through small exercises and guided examples.
4.	Synchronizing UI and CSS Code	Ensuring that the displayed CSS code exactly matched the generated gradient preview was difficult.	Implemented a centralized function to generate both the background style and CSS code simultaneously.
5.	Handling User Interaction	Managing multiple user actions such as color selection, refresh, and copy code required careful logic handling.	Used structured JavaScript functions and separated logic for each user action to improve readability and control.

CHAPTER 4

CONCLUSION AND FUTURE SCOPE

4.1 Conclusion

The one-month industrial training undertaken at Science & Technology Entrepreneurs' Park (STEP-GNDEC), Ludhiana proved to be a valuable learning experience, offering a strong platform to bridge the gap between theoretical concepts and practical implementation. Throughout the training period, emphasis was placed on acquiring hands-on experience in front-end web development, enabling the application of classroom knowledge to real-world scenarios.

The successful development of the Background Generator web application demonstrates the effective integration of HTML, CSS, and JavaScript to create a responsive, interactive, and user-friendly tool. The project fulfilled its intended objectives by allowing users to generate visually appealing background gradients in real time, select colors and directions, preview outputs instantly, and copy reusable CSS code effortlessly.

The project enhanced the understanding of fundamental web development concepts such as DOM manipulation, event-driven programming, responsive design, and dynamic styling. Deploying the application on the Vercel platform further provided practical exposure to modern deployment techniques and real-time accessibility.

Overall, the training strengthened technical skills, improved problem-solving abilities, and developed confidence in building complete web-based applications. The experience gained during this training will be instrumental in future academic projects and professional pursuits in the field of web development and software engineering.

4.2 Scope and Future Enhancements

The Background Generator project has significant scope for future improvement and expansion. While the current version meets its core objectives, several advanced features can be added to enhance functionality and usability.

From a functional perspective, the project can be extended to include:

- Multiple gradient types, such as radial, conic, and multi-color gradients.
- Angle and animation controls, allowing users to create animated or rotating gradients.
- Gradient presets and color history, enabling users to save and reuse their favorite designs.

- Export options, such as downloading gradients as images or exporting them in different formats.
- User-defined themes and dark/light mode support for enhanced accessibility.

From a technical viewpoint, the project can be upgraded by:

- Integrating modern JavaScript frameworks like React or Vue for improved scalability.
- Adding backend support for user accounts, saved designs, and cloud storage.
- Optimizing performance using advanced CSS techniques and code minification.

The project also has strong educational and industrial relevance. It can be integrated into web design tools, UI frameworks, or learning platforms to assist students and designers in understanding gradient creation and front-end design principles.

In conclusion, the Background Generator serves not only as a functional web utility but also as a solid foundation for future enhancements. Its scalability, simplicity, and practical relevance make it a promising project with the potential to evolve into a comprehensive web-design support tool.

REFERENCES

- [1] W3Schools: W3Schools Online Web Tutorials. Web Development Tutorials.
Available at: <https://www.w3schools.com>
- [2] CSS-Tricks: Coyier, C. A Complete Guide to CSS Gradients.
Available at: <https://css-tricks.com>
- [3] JavaScript.info: The Modern JavaScript Tutorial.
Available at: <https://javascript.info>
- [4] Vercel Documentation: Vercel Inc. Deployment and Hosting Guide.
Available at: <https://vercel.com/docs>
- [5] Flanagan, D. : *JavaScript: The Definitive Guide*. O'Reilly Media.
- [6] Guru Nanak Dev Engineering College (GNDEC):
Science & Technology Entrepreneurs' Park (STEP-GNDEC).
Institutional material and training resources provided during industrial training.
- [7] Google Developers: Web Fundamentals and Front-End Best Practices.
Available at: <https://developers.google.com/web>
- [8] GitHub Documentation: Version Control and Project Management.
Available at: <https://docs.github.com>

APPENDIX

A.1 HTML CODE

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
<head>
<meta charset="utf-8" />
<title>Background Generator</title>
<link rel="stylesheet" href="style.css" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<script src="script.js" defer></script>
</head>
<body>

<div class="wrapper">
    <h1>Background Generator</h1>
    <div class="gradient-box"></div>
    <div class="row options">

        <!-- Direction -->
        <div class="column direction">
            <p>Direction</p>

            <div class="select-box">
                <select>
                    <option value="to top">Top</option>
                    <option value="to right top">Right top</option>
                    <option value="to right">Right</option>
                    <option value="to right bottom">Right bottom</option>
                    <option value="to bottom">Bottom</option>
                </select>
            </div>
        </div>
    </div>
</div>
```

```

<option value="to left bottom">Left bottom</option>
<option value="to left">Left</option>
<option value="to left top" selected>Left top</option>
</select>

</div>
</div>

<!-- Color Palette -->
<div class="column palette">
<p>Colors</p>
<div class="colors">
<input type="color" value="#bb47d8" />
<input type="color" value="#e0ae24" />
</div>
</div>
</div>

<!-- Text Area -->
<textarea class="row" disabled>background: linear-gradient(to left top, #bb47d8, #e0ae24);</textarea>

<!-- Buttons -->
<div class="row buttons">
<button class="refresh">Refresh Colors</button>
<button class="copy">Copy Code</button>
</div>
</div>
</body>
</html>

```

A.2 CSS CODE

```
@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600&display=swap');

* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: 'Poppins', sans-serif;
}

body {
    padding: 0 10px;
    display: flex;
    align-items: center;
    justify-content: center;
    min-height: 100vh;
    background: linear-gradient(to left top, #bb47d8, #e0ae24);
}

h1 {
    text-align: center;
    font-weight: bold;
    color: #333;
}

.wrapper {
    width: 450px;
    padding: 25px;
    background: #fff;
    border-radius: 7px;
    box-shadow: 0 15px 30px rgba(0,0,0,0.06);
}
```

```
.wrapper .gradient-box {  
    height: 220px;  
    width: 100%;  
    border-radius: 7px;  
    background: linear-gradient(to left top, #bb47d8, #e0ae24);  
}  
  
.wrapper .row {  
    display: flex;  
    margin: 20px 0;  
    justify-content: space-between;  
}  
  
.options p {  
    font-size: 1.1rem;  
    margin-bottom: 8px;  
}  
  
.row :where(.column, button) {  
    width: calc(100% / 2 - 12px);  
}  
  
.options .select-box {  
    border-radius: 5px;  
    padding: 10px 15px;  
    border: 1px solid #aaa;  
}  
  
.select-box select {  
    width: 100%;  
    border: none;  
    outline: none;  
    font-size: 1.12rem;  
    background: none;  
}  
  
.options .palette {  
    margin-left: 60px;
```

```
}

.palette input {
    height: 41px;
    width: calc(100% / 2 - 20px);
}

.palette input:last-child {
    margin-left: 6px;
}

.wrapper textarea {
    width: 100%;
    color: #333;
    font-size: 1.05rem;
    resize: none;
    padding: 10px 15px;
    border-radius: 5px;
    border: 1px solid #ccc;
}

.buttons button {
    padding: 15px 0;
    border: none;
    outline: none;
    color: #fff;
    margin: 0 0 -15px;
    font-size: 1.09rem;
    border-radius: 5px;
    cursor: pointer;
    transition: 0.3s ease;
}

.buttons .refresh {
    background: #6C757D;
}

.buttons .refresh:hover {
```

```
background: #5f666d;
}

.buttons .copy {
background: #8A6CFF;
}

.buttons .copy:hover {
background: #704dff;
}

@media screen and (max-width: 432px) {

.wrapper {
padding: 25px 20px;
}

.row :where(.column, button) {
width: calc(100% / 2 - 8px);
}

.options .select-box {
padding: 8px 15px;
}

.options .palette {
margin-left: 40px;
}

.options .colors {
display: flex;
justify-content: space-between;
}

.palette input {
width: calc(100% / 2 - 5px);
}

.palette input:last-child {
margin-left: 0;
}
```

```
}
```

A.3 JAVA SCRIPT CODE

```
const gradientBox = document.querySelector(".gradient-box");
const body = document.querySelector("body");
const selectMenu = document.querySelector(".select-box select");
const colorInputs = document.querySelectorAll(".colors input");
const textarea = document.querySelector("textarea");
const refreshBtn = document.querySelector(".refresh");
const copyBtn = document.querySelector(".copy");

const getRandomColor = () => {
    // Generating a random color in hexadecimal format. Example: #5665E9
    const randomHex = Math.floor(Math.random() * 0xffffff).toString(16);
    return `#${randomHex}`;
}

const generateGradient = (isRandom) => {
    if(isRandom) {
        // If isRandom is true, update the colors inputs value with random color
        colorInputs[0].value = getRandomColor();
        colorInputs[1].value = getRandomColor();
    }
    // Creating a gradient string using the select menu value with color input values
    const gradient = `linear-gradient(${selectMenu.value}, ${colorInputs[0].value},
${colorInputs[1].value})`;
    gradientBox.style.background = gradient;
    body.style.background = gradient;
    textarea.value = `background: ${gradient};`;
}
```

```
const copyCode = () => {
    // Copying textarea value and updating the copy button text
    navigator.clipboard.writeText(textarea.value);
    copyBtn.innerText = "Code Copied";
    setTimeout(() => copyBtn.innerText = "Copy Code", 1600);
}

colorInputs.forEach(input => {
    // Calling generateGradient function on each color input clicks
    input.addEventListener("input", () => generateGradient(false));
});

selectMenu.addEventListener("change", () => generateGradient(false));
refreshBtn.addEventListener("click", () => generateGradient(true));
copyBtn.addEventListener("click", copyCode);
```