It seems like you've provided a detailed project description with multiple tasks. Below is an outline of the steps, the number of webpages required, and the functions that you'll need for your Math Game project. I'll also comment on where HTML, JavaScript, and CSS might be used for each task:

Webpages Required:

1. index.html (Registration)
2. play.html (Gameplay)
3. results.html (Results)
4. statistics.html (Statistics display)
5. charts.html (Charts display)

Functions: Global Storage Entity: Use a global JavaScript array called **PlayerRegistrationData** to store player data.

Task 1 (index.html):

- Create an HTML page with a registration form.
- Add fields for First Name, Last Name, Date of Birth, Age (calculated by the system), Gender, and Email.
- Use HTML tag attributes for basic validations.

Task 2 (JavaScript - Register):

- Create a JavaScript function **Register()** to validate and store the registration data in the **PlayerRegistrationData** array.
- Calculate age from the date of birth.
- Ensure the form fields have unique IDs.

Task 3 (index.html):

- Add a "Register" button to the form that calls the **Register()** function when clicked.
- Disable form fields and buttons.

Task 4 (JavaScript - PlayGame):

- Create a function **PlayGame()** to generate random numbers for the math equation.
- Validate user input.

Task 5 (play.html):

- Create a "Start" button that calls the **PlayGame()** function.
- Enable Play area, answer input, Accept button, and Next button.
- Disable math equation input.

Task 6 (JavaScript - CheckAnswer):

- Create a function **CheckAnswer()** to validate and check the player's answer.
- Append the data to **PlayerRegistrationData**.

Task 7 (play.html):

- Add a "CheckAnswer" button that calls the **CheckAnswer()** function.

Task 8 (play.html):

- Add a "Next" button that calls the **PlayGame()** function.

Task 9 (statistics.html):

- Create a statistics display area with the ID 'showpercentage'.
- Display player statistics in this area.

Task 10 (JavaScript - findPercentageScore):

- Create a function **findPercentageScore()** to calculate and display player statistics.
- Clear the 'showpercentage' area before displaying data.
- Call this function when the "End" button is clicked.

Task 11 (index.html):

- Add an "End" button that calls **findPercentageScore()** and performs other end-of-game actions.

Task 12 (index.html):

- Create a new display area with the ID 'showallplayers' to list all entered players.

Task 13 (JavaScript - showAllStats):

- Create a function **showAllStats()** to display all data from **PlayerRegistrationData** in 'showallplayers'.
- Format the data as specified.

Task 14 (JavaScript - CheckAnswer):

- Call **showAllStats()** at the end of the **CheckAnswer()** function.

Task 15 (JavaScript - showCharts):

- Create a function **showCharts()** to display frequency bar charts for gender and percentage scores.
- Calculate percentages and create the charts using HTML tables and stretched images.
- Call this function every 5 seconds after the page loads.

Task 16 (index.html):

- Add a "PercentageScore" button that calls **findPercentageScore()**.

Task 17 (CSS and Design):

- Use Vanilla CSS or Bootstrap for presentation and layout.
- Focus on color, font, readability, and design.
- Ensure the website is responsive to at least two screen sizes.

Task 18 (Documentation):

- Provide an agreed-upon WP Group Project Plan for each group member.
- Include signed Authorization forms from each student.
- Submit a Word document with an XHTML validation report (no errors, less than 6 warnings).
- Include an agreed-upon workplan indicating each group member's responsibilities, function names, and descriptions.

- Share a link to the hosting environment where the project is deployed.

Remember to incorporate HTML, JavaScript, and CSS as appropriate for each task, and ensure that the functions and features are integrated into the webpages as described in the tasks.