User Manual -HEAVY MACHINARY MONITORING SYSTEM

Login Screen

- Enter your **email or username** and **password** in the respective fields.
- Tap the eye icon to show or hide the password while typing.
- Tap the **Login** button to authenticate with the server. A loading spinner shows progress.
- On successful login, the user icon shrinks and moves with animation before proceeding.
- If login fails due to invalid credentials or network issues, a popup alert explains the issue clearly.

Register Screen

- Fill in **First Name, Last Name, Username, Email, Password**, and **Confirm Password**.
- Password and Confirm Password must match or you will see an error.
- Tap the **Register** button to submit your information. A spinner appears while processing.
- A green checkmark icon and success message confirm account creation.
- After successful registration, the app navigates back to the Login screen automatically.

Machine Dashboard Screen

Component:

- Displays a dynamic, animated greeting using the logged-in user's name, stored from `AsyncStorage`.

- A donut chart shows **ON% activity by machine (GFRID)** over the past week. Long press navigates to the Info screen. Can be hidden via close icon.

- Users can type in the GFRID to filter machines live, aiding faster lookup especially in large deployments.
- Each machine card shows:
    - GFRID
    - Current Status
    - Last Seen timestamp

Machine Detail Screen

Component:

- Users can explore **three analysis tabs** – Status, Movement, and Cumulative – for any selected machine (`GFRID`) using a material top tab navigator.
    - Choose `1h` or `1d` for auto-calculated ranges.
    - Select `Custom` to use date-time pickers and manually set a range.
    - Tapping the refresh icon reloads data and updates graphs based on the selected time range.
    - A horizontal scroll view at the bottom allows users to switch between available machines by GFRID, carrying over selected date/time filters.
    - Animations enhance feedback, while page visits and selected date ranges are logged using `logPageVisit()` for usage analytics.

MachineStatusGraph

Component Overview:

`MachineStatusGraph.jsx` visualizes the **ON/OFF status timeline** and **hourly usage** of a selected machine (via `GFRID`) across a chosen date range.

Features:

- Displays total machine **ON time** and **OFF time** using a runtime summary.
- Time format: `HH:MM:SS`.
- Graph toggles between **ON (1)** and **OFF (0)** over time.
- Each data point is timestamped with **time and date**.
- Only 50 points per page for performance and readability.
- Use arrows to scroll through pages.
- Shows usage in **minutes/hour** over the selected time range.
- Useful for spotting machine activity spikes or idle times.
- Both charts scroll horizontally if data exceeds screen width.

MovementAnalysisGraph

Component Overview:

`MovementAnalysisGraph.jsx` shows a breakdown of **movement types** or **alert events** over a specified time window, visualized via pop-up charts.

Features:

- Movement data grouped by type (e.g., movement names or alerts).
- Each movement shows count of events and can be clicked for deep analysis.
- Modal with animated entry (`scale`, `fade`, `slide`) shows the selected group's data.
- **Status**: Timeline graph (start → end) with 1/0 values.
- **Analysis**: Hour-wise distribution of total duration.
- Movement records are paginated with forward/back navigation.

- Only `MAX_DATA_POINTS` (100) are shown at a time to optimize rendering.
- Each movement label gets a unique color to visually distinguish it.
- Easily switch between different movement categories using horizontal scrollable buttons in the modal.
- Friendly image and message shown if no records found.

CumulativeAnalysisGraph

- Displays total **ON time** and **OFF time** in hours with percentages using animated cards.
- Clearly shows how much time each **alert ID** contributes using a colorful pie chart.
- Tapping any alert ID opens a **modal popup** showing percentage, duration, and alert ID details.
- Clean transitions for chart load, modals, and user interactions to enhance UX.
- If no data is found, shows a **friendly message** with a placeholder image instead of blank UI.

Profile Screen

Component:

- Displays the logged-in user's **name, email, and username**. Data is fetched from `AsyncStorage` and animated upon load.
- Drag the screen downward to **refresh and reload user info**.
  - **Download Reports**: Opens the screen to download data.

- **Logs History**: Opens a log of past usage.
- **View History**: Redirects to past machine activity.
- **Logout**: Opens a confirmation modal.
- Smooth fade, slide, and scale animations enhance the greeting experience when user data is shown.
- On confirming logout:
- Sends a request to invalidate session on the server.
- Clears all user data from `AsyncStorage`.
- Returns to login screen via `onLogout()` callback.

History Screen

Component:

- Displays a list of previously downloaded reports, showing the **type**, **from/to dates**, and **downloaded date**.
- Users can **filter history** by selecting a **start and end date**. Filtered results appear instantly.
- Each history entry can be **soft-deleted** with a delete button. You get **7 seconds** to undo before it's permanently removed.
- Animations are used for list entry, deletion (binDrop), and undo bar, offering an engaging UI experience.
- If no history exists or filters yield no results, the screen shows a **friendly message** with an option to clear filters.

Logs Screen

Component:

- Shows login sessions with metadata such as **IP**, **device info**, **visited pages**, and **timestamps** grouped by date.
- Each session displays a **timeline bar** with animated indicator showing **session duration** and **active status**.
- Tapping a log opens a modal with **full session details**, including **page visits** and **filters applied**.

- Users can delete any session log directly from the list, with **confirmation prompts** to prevent accidental deletion.

- Uses **fade**, **zoom**, and **slide** animations to improve usability and make the log tracking more intuitive.

Download Screen

Component: `DownloadScreen.jsx`

- Users can download **PDF reports** either for **all machines** or by entering a **specific GFRID** (machine ID), based on selected date ranges.
- Users must choose both **From** and **To** dates using a calendar picker before downloading. The dates are validated and formatted appropriately.
- Toggle between **All Machines** and **By GFRID** report type. If “By GFRID” is selected, a **GFRID input field** becomes visible and mandatory.

- Upon successful download:

    - The report is shared using device-native sharing.

    - The download activity is **logged** and sent to the backend (`/api/auth/history/record/`) with user and filter metadata.

    - Uses animations for better UX (`fadeIn`, `bounceInDown`, `shake` etc.). Any download errors are caught and displayed, and users are prompted to fix missing fields.

Info Wrapper Screen

Component:

- Displays machine activity trends using donut and line charts.
- Shows status records by GFRID.
- Tracks and paginates data per machine (scrollable graphs).

Offers quick filters:

- 1 Day (1d)
- 1 Week (1w)
- 1 Month (1m)
- **Custom Range** (via date pickers)
- **Donut Chart**: Displays ON% by GFRID.
- **Paginated Line Charts**: Visualizes status changes over time with horizontal scroll and pagination.

- If custom date range exceeds **1 month**, the app suggests downloading a report instead of loading large data on-screen.
- A dedicated **Download** button is available, allowing users to download full reports for long-term analytics.