Crash Tracker  
  
Description:  
Crash Tracker allows users to view detailed records of race crashes, including the crash type, involved racers, race details, and any additional information. The site offers a user-friendly interface for viewing and managing crash data, helping users analyze crash trends or keep records up to date.

# User Stories:

## Adding a crash record:

As a user, I’m able to submit a request to add a crash record to the database by providing details such as crash type, racer, and race, so that I can contribute to expanding the crash records.

### Acceptance criteria:

User can submit a form with all required fields (crash type, racer, race details).

An admin receives the request for the submission form.

## Viewing crash data in different formats:

As a user, I’m able to view crash information in either graph form (e.g., bar chart of crash types by race) or as direct values, so that I can analyze and understand the type of crash I'm looking for.

### Acceptance criteria:

User can switch between table view and graphical view (e.g., pie chart, bar chart).

The crash data displayed is accurate and updates dynamically as new records are added.

## Requesting record removal or edits:

As a user, I’m able to submit a request to remove or edit an existing crash record by explaining the issue, so that the database remains accurate and authentic.

### Acceptance criteria:

User can fill out a form detailing the changes or removal request.

The request is reviewed by an admin before approval.

## Admin management of crash data:

As an admin, I’m able to add, remove, edit, and view crash records using a user-friendly dashboard so that I can efficiently manage the website and maintain data accuracy.

### Acceptance criteria:

Admin dashboard includes options to view, edit, approve, or reject user-submitted crash records.

Admin can directly add new records or modify existing ones with minimal effort.

## Admin user overview page:

## As an admin, I’m able to view all users on a user overview page, so that I can easily manage user accounts.

### Acceptance criteria:

### Admin dashboard includes a user overview page displaying a list of all registered users.

### Admin can view user details such as full name, email, and their role.

## Admin submission overview page:

As an admin, I’m able to view all user submissions in one place, so that I can manage pending requests more effectively.

### Acceptance criteria:

Admin dashboard includes a submission overview page listing all submissions with their type and date.

## User submission history:

As a user, I’m able to view all my submitted forms, so that I can track the status of my contributions.

### Acceptance criteria:

Users have access to a personal submissions page showing all their submitted forms.

# Wireframe stories:

# Domain model:

# Conceptual model:

# Logical model:

