Final Write Up

List all team members and their roles and contributions to the project.

* Ashley Fram: APIs, Database, and Script
* Matthew Fulwood-Benbow: JavaScript
* Evan Ezeobinwa: HTML/CSS
* Schilosson Chauvet: JavaScript
* Khadijah Shamim: HTML/CSS

Disclosure: We all worked on a bit of everything, though we did have assigned roles (see left), it was easier for us to each do as much as we could within each category to complete the project.

We all did the PowerPoint Presentation and Final Write-Up

Github: [anrfram/CIS](https://github.com/anrfram/CIS)

* + Link to YouTube video

**German Modded Car Marketplace: A Niche Platform for Enthusiasts**

The performance, dependability, and inventiveness of German engineering have long been admired by the car enthusiast community. Audi, Volkswagen, Porsche, Mercedes-Benz, and BMW are just a few of the brands that have come to represent superior craftsmanship and tunability. However, the current internet platform is ill-suited to the unique requirements of this specialized market when it comes to purchasing or selling modified German automobiles. Current platforms, such as AutoTrader or Facebook Marketplace, only offer limited capability, which frequently results in inaccurate listings, inadequate documentation, and a higher risk of scams. Although there are enthusiast forums like Bimmerpost and VWVortex, they are not designed with online purchasing in mind. Inspired by our own interest for the modding scene, we set out to develop a full-stack web application that would provide a clear, well-curated, and easy-to-use platform for German auto enthusiasts to purchase and trade customized cars.

Our interest in this project stemmed not only from our shared appreciation of German automotive culture but also from the opportunity to apply and refine our technical skills in a meaningful way. In addition to pushing ourselves to use cutting-edge web development techniques and tools, we wanted to create something that addressed a genuine community need. The word "modded" describes any car that has been altered from the manufacturer, whether for customization, performance, or appearance. These adjustments include body kits, custom exhausts, forced induction upgrades, ECU tuning, and even minor tweaks known as "OEM+," which increase performance without detracting from the vehicle's stock look.

The history of modding is extensive, particularly in Germany, where businesses such as Alpina, Brabus, and ABT have been modifying BMW, Mercedes, and Audi cars for a long time. Cars like the Mercedes 190E Cosworth and BMW E30 M3 became legendary in the 1980s and 1990s, and they frequently served as the basis for significant performance improvements. With the advent of tuning software, DIY culture, and reasonably priced aftermarket parts, these adjustments became more widely available to the typical enthusiast. Many enthusiasts are still forced to sort through generic listings or unsafe private transactions due to a lack of digital infrastructure facilitating the sale of modified cars, despite this cultural growth. Our goal was to bridge this gap by developing a centralized, enthusiast-driven platform that prioritized transparency, documentation, and community trust.

To achieve this, we established two main goals: The first step is to create a strong, feature-rich marketplace designed especially for German cars that have been upgraded. The second goal is to improve our abilities as full-stack engineers by getting practical experience in creating, implementing, and safeguarding dynamic online applications. Our top priorities were user authentication for safe logins, comprehensive listings for automobiles with modification histories, and robust search/filter capabilities that let users quickly locate cars by make, model, year and price. For added value, we also incorporated external APIs, namely the National Highway Traffic Safety Administration (NHTSA) API, which shows open recall details for every vehicle on the list.

We used a full-stack development approach to the project from a technical standpoint. Using HTML, CSS, and JavaScript, the frontend was constructed with an emphasis on responsive design and user experience. Node.js and Express.js were used in the development of our backend, which managed server-side logic, routing, and API interaction. We switched to SQLite3 for ease of setup and deployment throughout development, even though our original plan was to use PostgreSQL for the relational database. The main table used in the creation of this database included all the makes and models of the German cars we were focused on listing. This relational structure allowed us to query data efficiently and display complex car builds with accompanying images and documentation. For authentication we used Google AuthO which works for temporary use via URIs. After more development, we would implement a domain for this authentication process to ensure it works for all users. For APIs we used both Google AuthO and the National Highway Traffic Safety Administration (NHTSA), which shows recall history on any car the user is interested in.

The project achieved many of its core goals, although not without some setbacks. Our listing system functioned reliably, allowing users to register, log in, and post detailed car listings complete with modification histories and photos. It was simple to browse through listings given the efficient make, model, and price range filtering. An extra degree of transparency was provided by the accurate display of the recall data obtained from the NHTSA API. Our suggested messaging system, which was intended to enable direct communication between buyers and sellers on the platform, could not be fully implemented, though. Similarly, because of time and complexity issues, a planned "Car of the Day" feature that was meant to showcase exceptional cars using dynamic backend logic was abandoned. Our only other major conflict was regarding our Google AuthO API. Though it does work, because of Replit’s lack of domain, for testing purposes we used URIs.

Despite our drawbacks, the project turned out to be a valuable educational opportunity in a number of fields. Technically speaking, we learned a lot about creating relational databases, establishing safe authentication methods, and maintaining backend APIs. Additionally, we learnt how to debug external dependencies, including adjusting to data structures in APIs or resolving problems with picture uploads in different hosting systems. Given that much of the development took place on Replit, which made real-time collaboration challenging, we encountered practical issues with time allocation, scope creep, and team coordination from a project management standpoint. Although GitHub and Git assisted us in managing version control, it took more work than expected to keep branches organized and resolve conflicts.

In our development process, we discovered that the 80/20 rule was quite applicable—roughly 80% of the project's perceived value derived from 20% of its features. The core elements—secure listings, modification documenting, and search filters—were more important than chasing novelty with features like car of the day and automobile comparisons. We were more disciplined as a team in putting these necessities first and putting experimental ideas off for later iterations. Additionally, we gained a greater understanding of modular design, which helped us stay on course by dividing difficult jobs into smaller, more manageable tasks.

To sum up, our German Modded Car Marketplace met most of its goals and established a strong foundation for growth. We expanded our technical resources that extend frontend, backend, and database platforms while developing a useful, specialized platform that fills a genuine market need. The importance of providing a functional product rather than striving for perfection was maybe the most significant lesson we took away. We produced a platform that may expand over time with community feedback and developer iterations by concentrating on usability and essential functionality.

For future development, we intend to switch from SQLite3 to PostgreSQL in order to facilitate growth and enhanced relational integrity. We also want to enable in-platform communication by implementing real-time messaging over Firebase or WebSockets. We intend to add automobile comparison tools, improve the user interface with a mobile-responsive design, and start a community section where users can interact with other modders, create build threads, and exchange reviews.

In the end, our efforts confirmed our belief that innovation is achievable even in highly specialized areas by combining enthusiasm, preparation, and real-world development. This project was our first step in building a contemporary, reliable marketplace that the modding community deserves.