



Trimester March/April, 2025

CSE6224 SOFTWARE REQUIREMENTS ENGINEERING

Project Part 1

**Topic: Campus Ride-Sharing Platform with
Parking System Integration
Kano Model Documentation**

Name	Student ID	Course
Chee Rui	1211112287	Bachelor of Computer Science
Teh Li Wei	1211109581	Bachelor of Computer Science
Sow Chien Yee	1211210800	Bachelor of Computer Science
Lai Zi Xuan	1211109451	Bachelor of Computer Science

Table of Contents

1 Elicitation Strategy.....	3
1.1 Justification for Using the Kano Model Elicitation Strategy Overview	3
1.2 Classification of Requirements Using Kano Model.....	4
2 Elicitation Execution and Findings.....	5
2.1 Categorized Requirements (Based on Kano)	5
3 Appendices.....	8
3.1 References	8

1 Elicitation Strategy

1.1 Justification for Using the Kano Model Elicitation Strategy Overview

The Kano Analysis model is also known as the “Customer Delight vs. Implementation Investment” approach. It is an analysis method that helps measure customer emotional responses to a product and its features.

Using a Kano model questionnaire, product features are categorized on a two-axis scale: satisfaction and functionality. This allows us to prioritize features of a product based on their potential to satisfy customers versus the effort required to implement them.

The Kano model is especially useful for teams looking to answer key questions such as:

- How can we measure customer satisfaction?
- What features can we create to increase customer satisfaction?
- Do our current features lead to high customer satisfaction?
- How can we enhance our features to reach optimal satisfaction (i.e., “delight” customers)?

By focusing on these questions and aligning product development with customer needs, we can determine what it takes for the product to enter, stay in, and excel within the target market.

In our case, it helps us understand how to keep our parking and carpooling system within scope while meeting the expectations of both students and staff, all while maintaining sufficient control for administrators.

1.2 Classification of Requirements Using Kano Model

Let's break down how the Kano Model works, what each of the categories represents, and how it is classified.

In the Kano Model, all requirements and features are classified into three main categories to help prioritize development:

1. Dissatisfiers (Must-Be Quality)

- These are basic expectations that users assume they will be present.
- Their presence does not increase satisfaction, but their absence causes strong dissatisfaction.

2. Satisfiers (Performance Quality)

- Implementation of these features result in greater user satisfaction, and dissatisfaction when poor or missing.
- Users are consciously aware of these needs.

3. Delighters (Excitement Quality)

- These are unexpected features that greatly increase satisfaction when present, but cause no dissatisfaction when absent.
- Users may not ask for them, but having them grants an advantage over other competitors without them.

How We Sorted Requirements and Features

To classify our system's features using the Kano Model, we applied the following three methods:

- **Kano Model Questionnaire**
The most direct approach. It maps combinations of user responses to functional and dysfunctional questions into one of the Kano categories.
- **Interviews**
By engaging stakeholders and users directly, we gathered detailed opinions and preferences. This helped assess how important each feature is perceived to be, and also uncover new features.
- **Prototype Feedback**
By showing visual mockups of the system, we observed UI expectations, experience-related suggestions, and discovered potential technical challenges that could impact implementation.

Finally, similar requirements and features will be grouped together and categorized as listed below.

2 Elicitation Execution and Findings

2.1 Categorized Requirements (Based on Kano)

2.1.1 Dissatisfiers (Must-be Requirements)

These features are essential and expected by users. Their absence would cause dissatisfaction.

Feature	Justification
Login with Student ID	Users expect secure access.
Drivers can accept/decline ride requests	Considered vital for safety and control, especially by drivers.
Admin login using Admin ID	Basic requirement for admin-level access and management.
Ride info: Time, destination, seat count, price	Required by both drivers and riders during booking.
University verification of users	Key trust and safety measure for all users.
Real-time parking availability and map	Highly valued; considered essential for both safety and convenience.
Admin can assign stable parking spot IDs	Supports consistent UI and map accuracy, especially during resizing.
Show car details directly on the same page(for admin pages)	Admins prefer only important information, extras will affect user experience.

2.1.2 Satisfiers (Performance Requirements)

These features impact satisfaction directly based on performance. Their absence reduces usability.

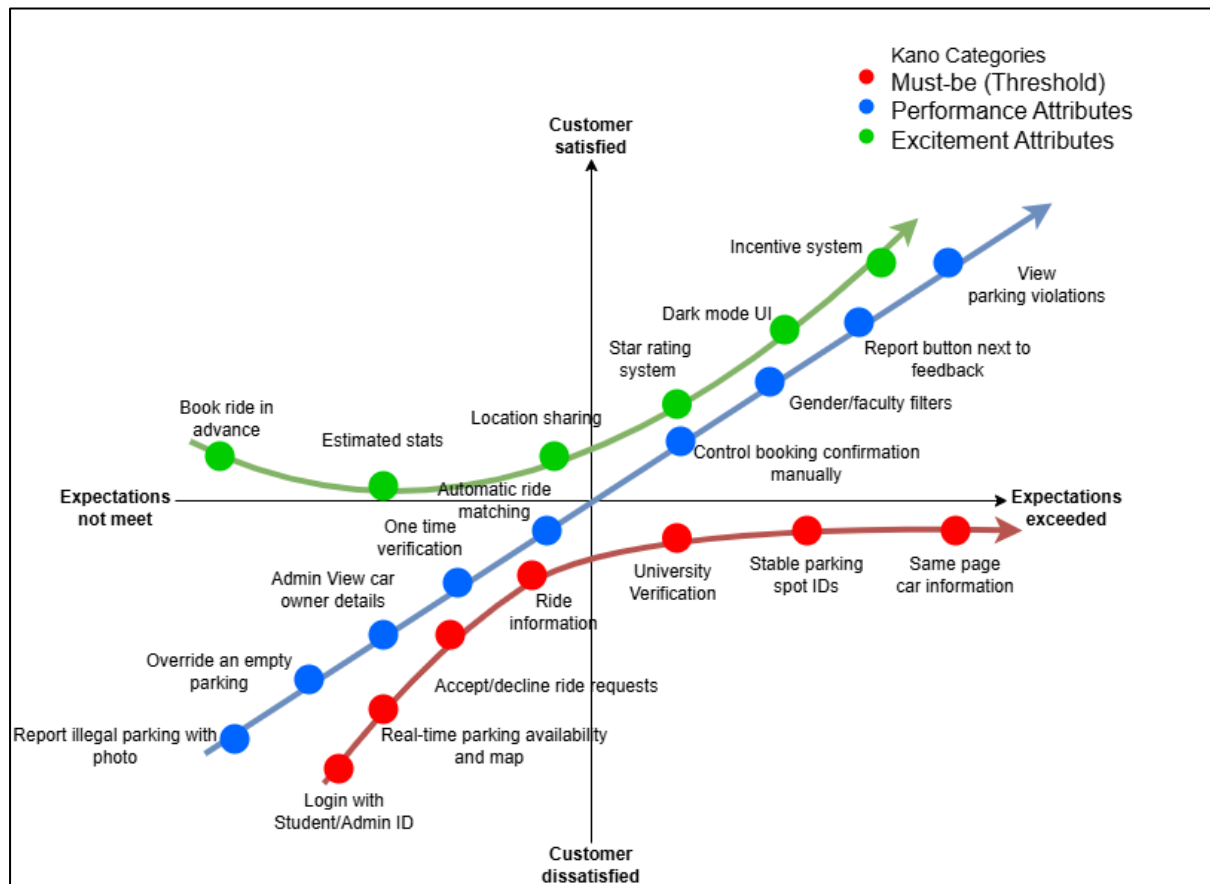
Feature	Justification
One-time verification using ID is sufficient	Users expect seamless access via existing systems. Requiring an extra login would create redundant processes.
Report illegitimate parking (with photo upload)	Useful feature with high appreciation, but users still tolerate its absence.
Admin can view reported parking violations	Improves rule enforcement and confidence in system oversight.
Driver can override reserved parking spot	Mixed views; useful for managing space, but not always expected.
Manual approval of ride requests (by driver)	Preferred by drivers for control, though not critical for function.
Automatic ride matching (by system)	Preferred by riders for ease, though optional in driver view.
Gender/faculty filters when matching riders	Helpful for personal safety and comfort; not required but increases confidence.
Report button place next to feedback button for riders after rides.	This adds useful control and accountability, increasing satisfaction.
Admin can view car owner details	Mixed reactions; privacy concerns exist, but some users find it useful in problem resolution.

2.1.3 Delighters (Excitement Requirements)

These features were unexpected but appreciated. They increase satisfaction when present.

Feature / Requirement	Justification (from elicitation)
Rider can book rides in advance with faculty member	Unexpected but liked by some users; not a basic requirement.
Reward system: fuel compensation or points	Motivates usage but not expected. Users see it as a bonus feature.
Estimated fuel saving / carbon stats	Appeals to eco-conscious users; surprising and appreciated.
Location sharing with friends	Enhances safety and social trust; not expected, but positively received.
Star rating system after rides	Useful for feedback; not critical but appreciated.
Dark mode UI	Purely aesthetic; not expected but liked by night-time drivers.

Kano Model



3 Appendices

3.1 References

1. Qualtrics. "Kano analysis: The kano model explained" Accessed May 20, 2025. <https://www.qualtrics.com/en-au/experience-management/research/kano-analysis/>.
2. American Society for Quality. (n.d.). *What is the Kano Model? Diagram, Analysis & Tutorial*. Retrieved May 22, 2025, from <https://asq.org/quality-resources/kano-model>