



# CERIA CHECK IN & PAYMENT SYSTEM

Campus Event Check-In System

Company Name: TT1L\_G5

NAME	STUDENT ID
MUHAMMAD AQIL BIN RAHMAT	1211107976
AHMAD FAIZ BIN ANUAR	1211109154
WAN MUHAMMAD ILHAN BIN WAN ZIL AZHAR	1211102908
MUHAMAD AIMAN BIN MOHD EHWAL	1211103064

## Table of Content

<b>1.0 Purpose of Requirement Elicitation</b>	<b>3</b>
<b>2.0 Elicitation Goals &amp; Objectives</b>	<b>3</b>
Elicitation Goals	3
Elicitation Objectives	3
<b>3.0 Requirement Sources</b>	<b>4</b>
<b>4.0 Elicitation Strategy and Techniques</b>	<b>5</b>
4.1 Kano Model	5
4.2 Kano Model Application	5
4.2.1 Identifying Potential Features	5
4.2.2 Creating the Kano Survey	5
4.2.3 Survey Questions	5
4.3 Survey Result & Analysis	6
Feature 1: The Use Of Student ID For Event Check-Ins.	7
Feature 2: Using QR Code To Scan For Event Check-Ins.	8
Feature 3: On-The-Spot Payment System.	9
Feature 4: Real-Time Notifications For Updates.	10
Feature 5: Storing Information In Database For Future Use.	11
Feature 6: Confirmation Email After Check-Ins.	12
Feature 7: Buying Event Tickets & Checking In Simultaneously.	13
Feature 8: Emergency Contact Button.	14
Feature 9: Rating System.	15
Feature 10: Mobile App For The System.	16
Conclusion	17
<b>6.0 How This Will Be Used In Our Software</b>	<b>18</b>

## 1.0 Purpose of Requirement Elicitation

The purpose of requirement elicitation is to gather detailed insights and understand what the stakeholders need from the CERIA Check-In & Payment System. This is to ensure that the system can solve the actual problems experienced by users. Proper elicitation ensures the system is aligned with the expectations of end-users and can be developed to meet both current and future needs of the university's event operations. Other than that, it makes sure that the system aligns with the university's operational processes and can be implemented with available technology and infrastructure. This ensures that the system meets real-world expectations and integrates effectively into campus operations.

## 2.0 Elicitation Goals & Objectives

### Elicitation Goals

- To identify all the functional and non-functional requirements of the system.
- To ensure that the system meets the needs of the stakeholders
- To collect technical and operational constraints that may impact design.
- To ensure feasibility and scalability of the system for different sizes of campus events.

### Elicitation Objectives

- Gather integration requirements for databases and payment gateways.
- Determine expectations related to user experience, data management and security concerns by understanding stakeholder expectations and needs for both functional and non-functional needs.
- Enable stakeholder collaboration to refine and validate proposed features before development begins.
- Document clear and actionable requirements that guide the system design and development.

### 3.0 Requirement Sources

A requirement source is the reference from which a particular software requirement is derived. It helps trace each requirement back to a specific stakeholder or document, which is crucial for validation and change management.

Requirement	Source Type	Source Name	Description
Qr code check-in for students	Stakeholder	Students	Needed for ease of entry at events
Vendor sales tracking system.	Stakeholder	Vendors	Needed real-time system that allows sales transactions to be processed instantly as they happen, Point of Sale (POS) functionality.
Integration with student ID system	Existing System	CLiC Database	Used for student verification
Payment processing	System	Payment Gateway API	Required for secure in-app payments
Attendance monitoring and reporting	Stakeholder	Admins	Needed for event validation and university reporting

## 4.0 Elicitation Strategy and Techniques

### 4.1 Kano Model

Kano model is a questionnaire-based elicitation technique that categorizes user's answers into three categories. Based on the answers given by the users, we will be able to ascertain whether the functionality is a must-have for the system or not. This way, we will be able to efficiently split our work force in order to minimize development time.

### 4.2 Kano Model Application

#### 4.2.1 Identifying Potential Features

We identified potential features that can be implemented inside the primarily by brainstorming multiple potential ideas that our app can use. After shortlisting a few ideas that we believe should be implemented inside the system. We design the questionnaire for each of the potential features based on the Kano Model.

#### 4.2.2 Creating the Kano Survey

Kano survey works by presenting the features to the potential customers (students) in the form of a tier list. Each feature will have 2 questions each, one will be for if the feature is implemented inside the system and the other is for if the feature is not implemented. The survey will be distributed online using Google Forms and the responses users gave will be converted into a Kano Model Graph.

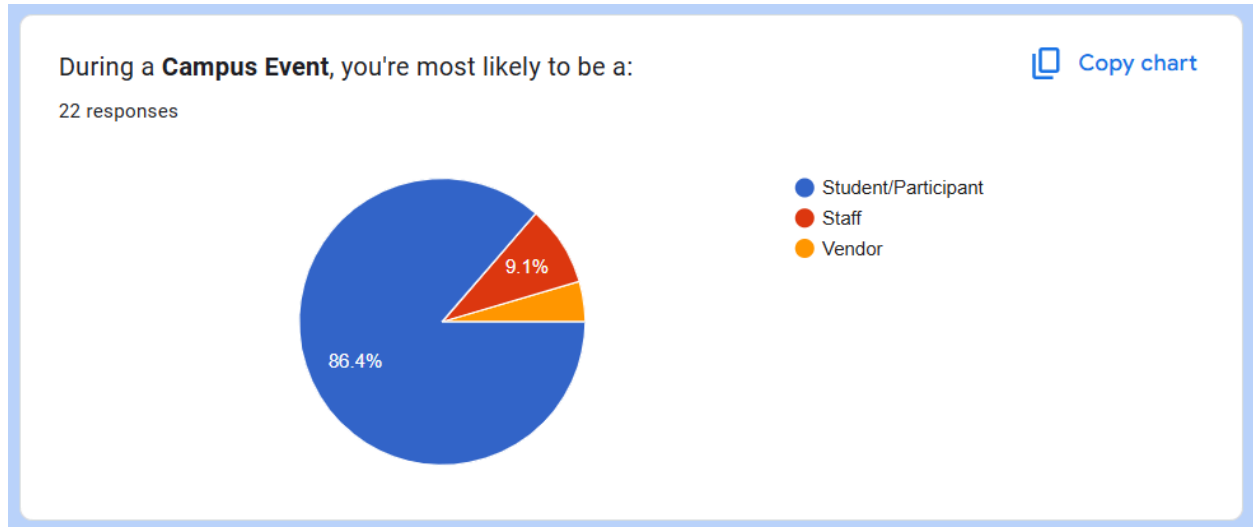
#### 4.2.3 Survey Questions

Based on the features, we will give the customers options between **Very Uncomfortable** until **Very Comfortable**, this will help us judge how the user will respond to the planned feature for the app. The questions inside the survey are as follows.

- 1) Using student id for event check-ins
- 2) Using QR Code for event check-ins
- 3) System allows payment using the app itself
- 4) System sending real-time notifications
- 5) System storing user information for future use
- 6) System sends confirmation emails after check-ins
- 7) User being able to buy event tickets from within the app
- 8) System having an emergency contact button during events
- 9) System letting users rate event after attending
- 10) System having its own mobile app

### 4.3 Survey Result & Analysis

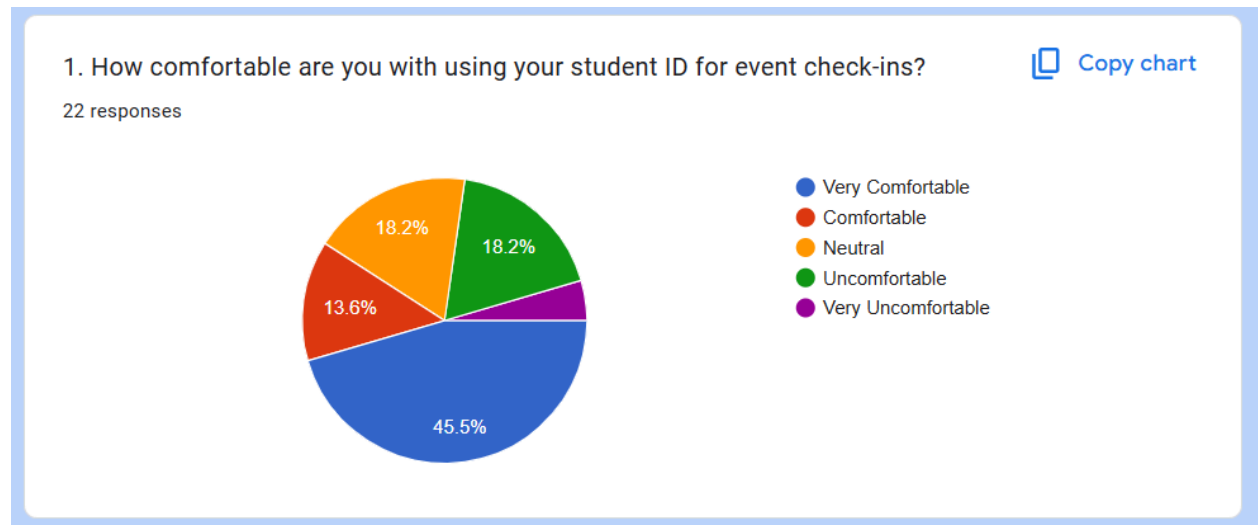
After sharing the survey for approximately a week, we've gathered a total of 22 responses from the students of MMU on our Google Forms.



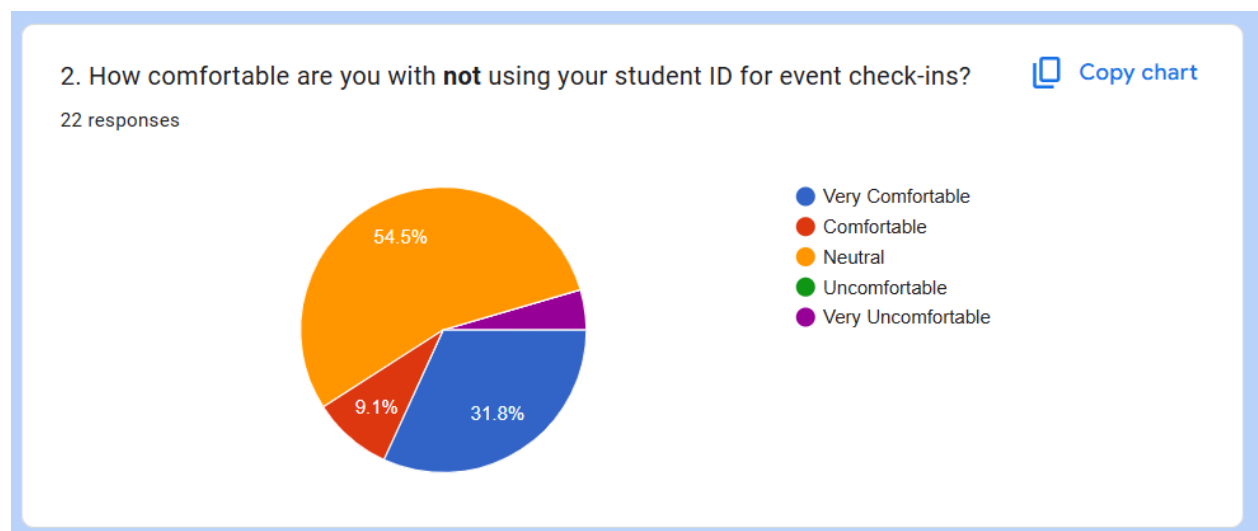
The data above tells us that 86.4% of users are more likely to be a student/participant. With 9.1% of them likely to be a staff of the event and 4.5% are likely to be a vendor.

With this information in mind, we can focus on developing the Ceria Check In & Payment System to suit students for their convenience and usability.

## Feature 1: The Use Of Student ID For Event Check-Ins.



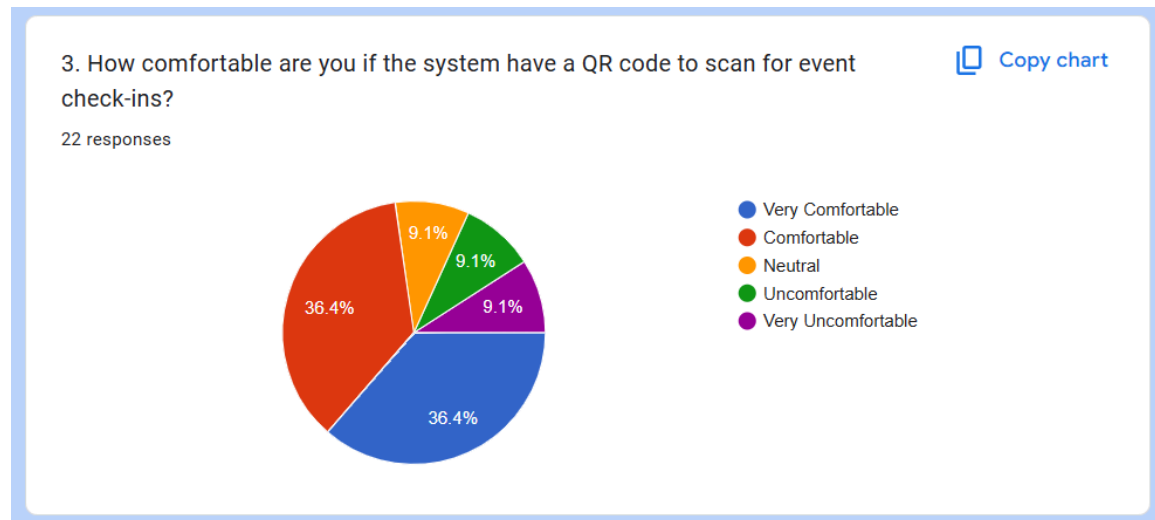
Based on the responses, over 59.1% of users are comfortable with using their student ID for event check-ins. With 18.2% neutral responses, this indicates that the feature is worth developing for the users.



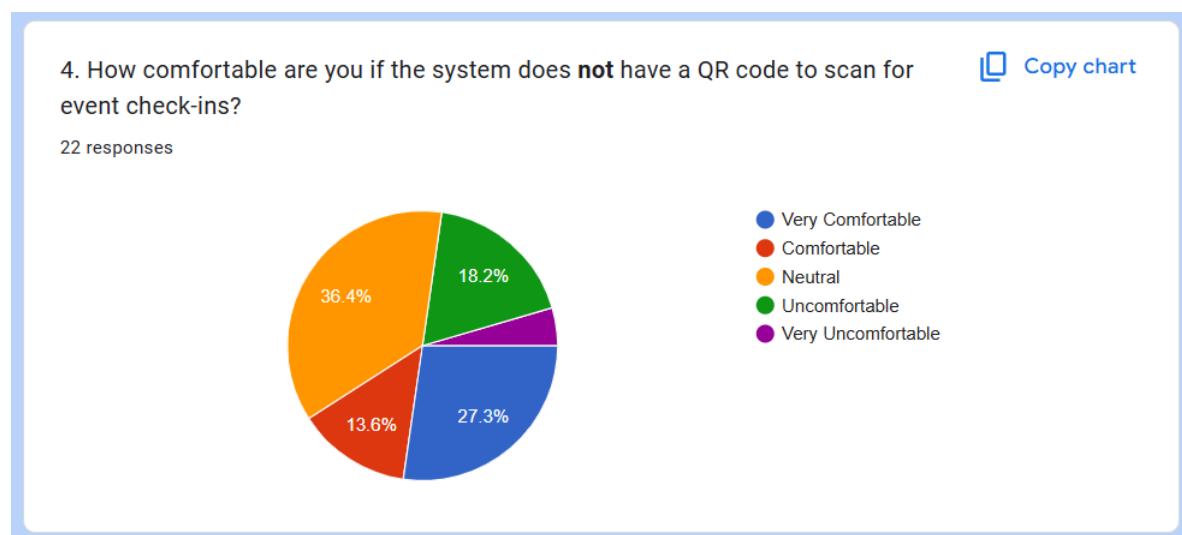
Over half of the respondents are neutral with the feature not being available. 40.9% are comfortable with **not** having the feature.

Based on the responses, with 59.1% users being comfortable with having the feature while more than half are neutral, we can deduct that the feature is leaning more towards a **delighter** classification.

## Feature 2: Using QR Code To Scan For Event Check-Ins.



Based on the responses, 72.8% of users are very comfortable/comfortable to have a QR code to scan for event check-ins. One-third of the other respondents are neutral, uncomfortable and very uncomfortable. Based on the majority, this feature is highly likely to be developed.

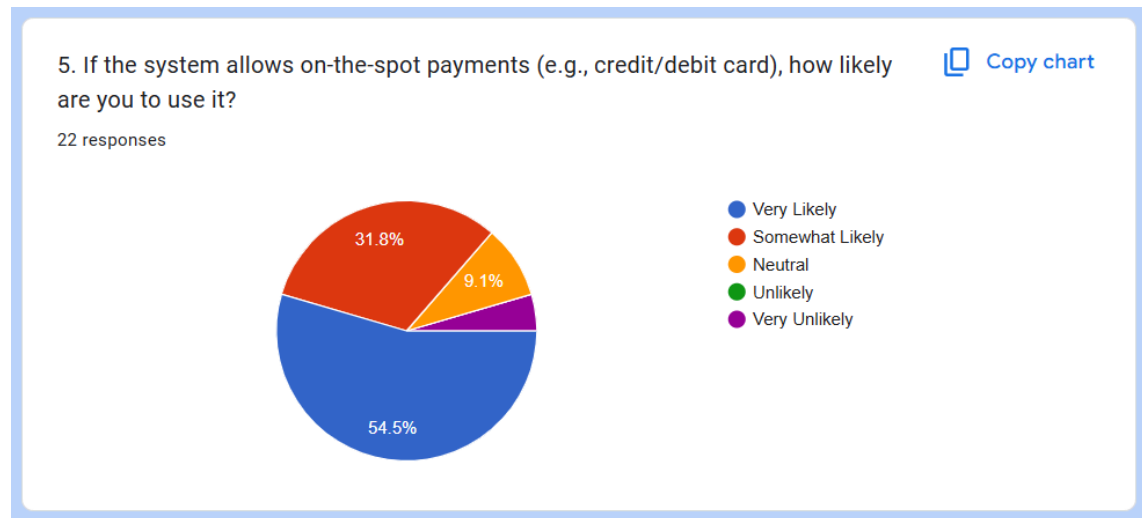


One-third of the respondents are neutral with not having a QR code to scan for event check-ins. While 40.9% of users are comfortable with not having a QR code, 22.7% are not comfortable with not having a QR code.

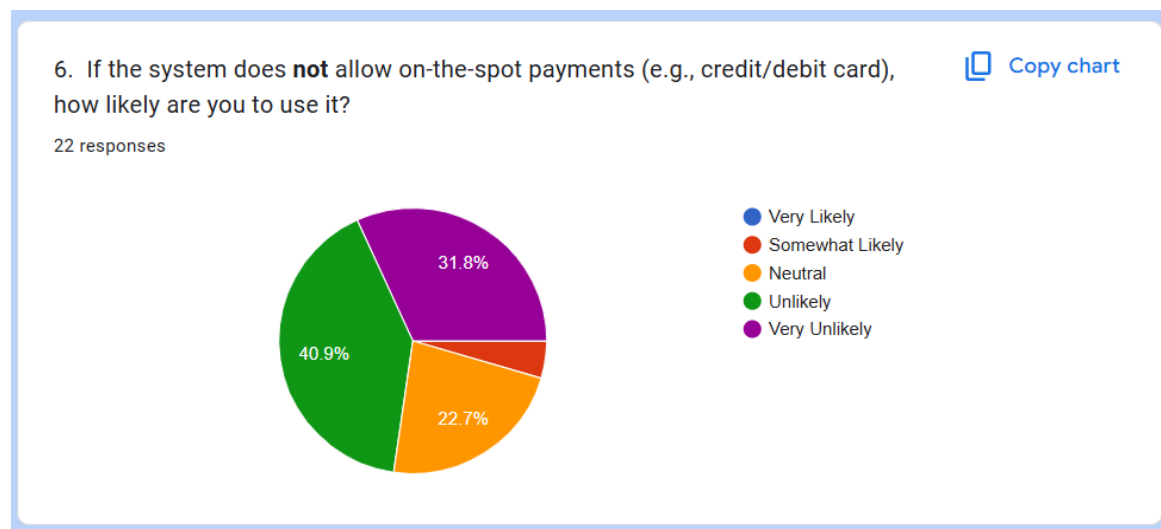
Since over 35% are comfortable with having this feature, and over 15% are uncomfortable with not having the feature, we can deduce that this feature is a **satisfier** classification.



### Feature 3: On-The-Spot Payment System.



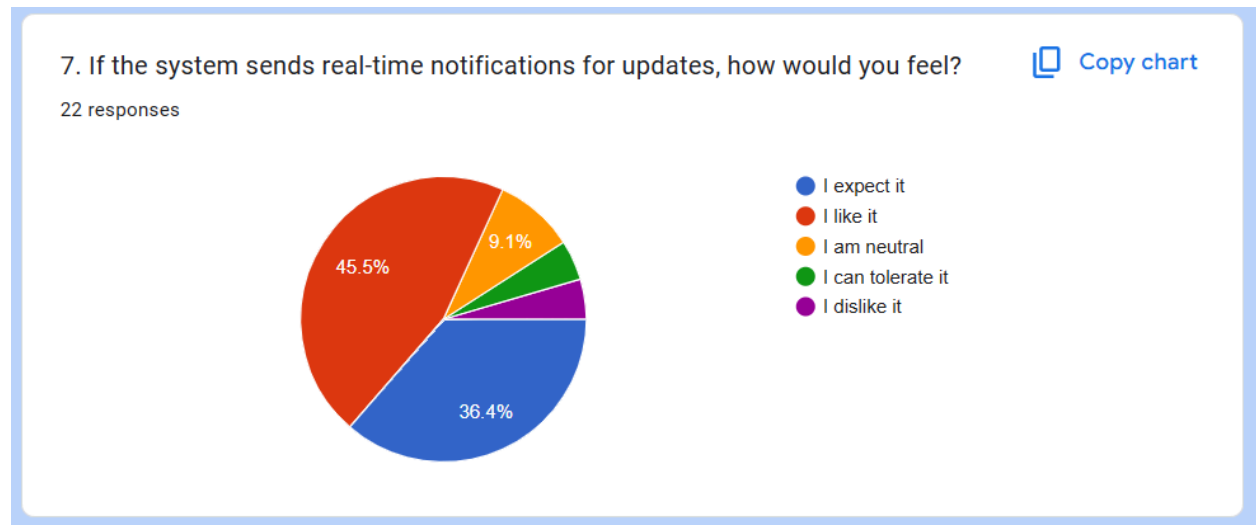
The responses indicate that users would be very likely to use the feature, with 86.3% are very likely and somewhat likely to use it. While 9.1% of respondents are neutral with the feature and 4.5% are very unlikely to use it. The responses suggest that the feature is definitely worth developing for the system.



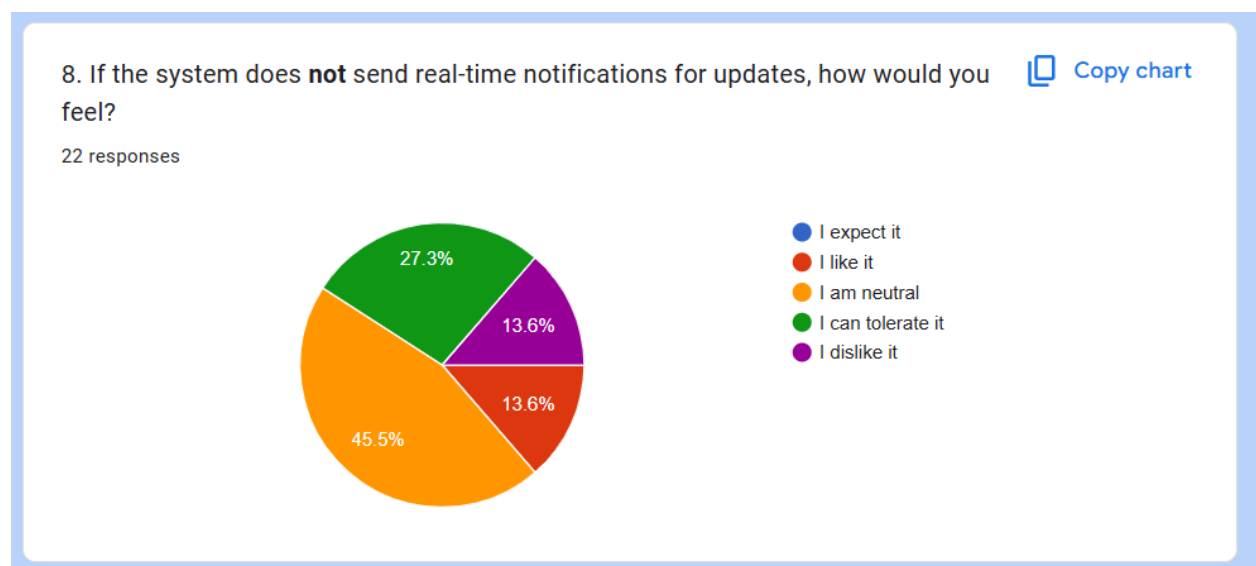
Based on the responses, 72.7% of the respondents are unlikely to use the system if on-the-spot payments are not integrated. With 22.7% being neutral on whether it is added or not.

Since 86.3% are very likely to use it and 72.7% are very unlikely to use the system at all if it's not integrated, this concludes that the feature is a **dissatisfier** classification.

#### Feature 4: Real-Time Notifications For Updates.



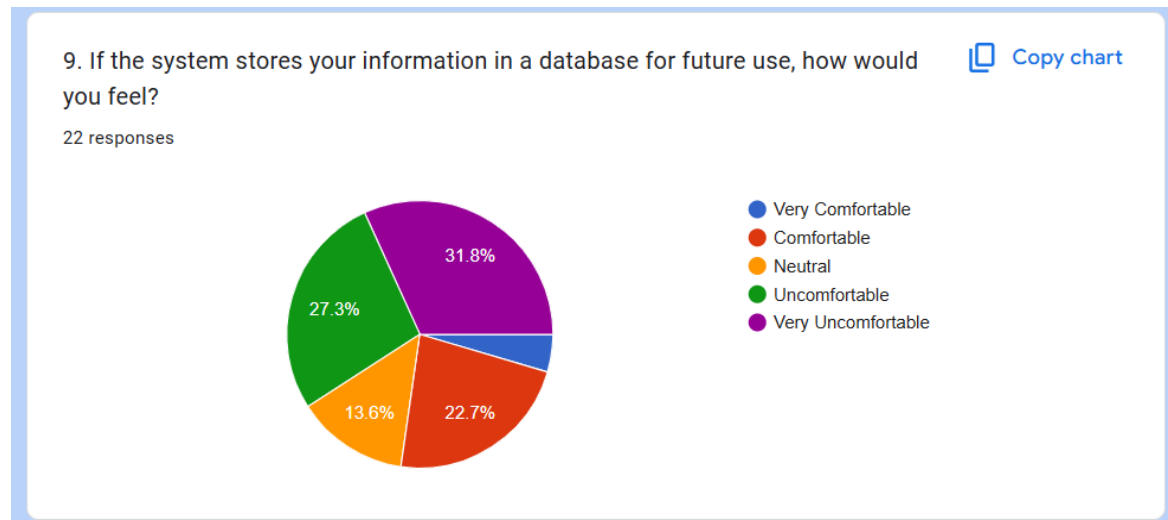
The data collected from the responses shows that 81.9% of the respondents would like/expect the feature, while 9.1% are neutral and 9% prefer not having it. This shows that the feature would be worth developing.



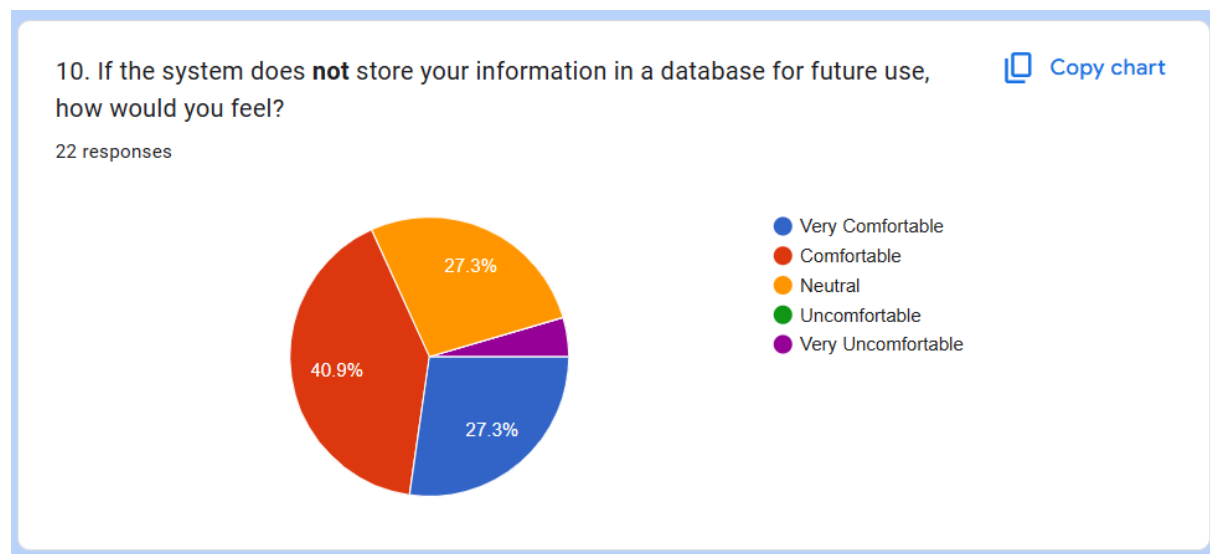
Based on the responses of the question, 45.5% of respondents are neutral with not having real-time notifications and 40.9% dislike not having the feature.

Since 81.9% of potential users would like to have the feature, and 40.9% dislike not having it with a lot of neutral responses, we can conclude that this feature is considered a **dissatisfier** classification.

## Feature 5: Storing Information In Database For Future Use.



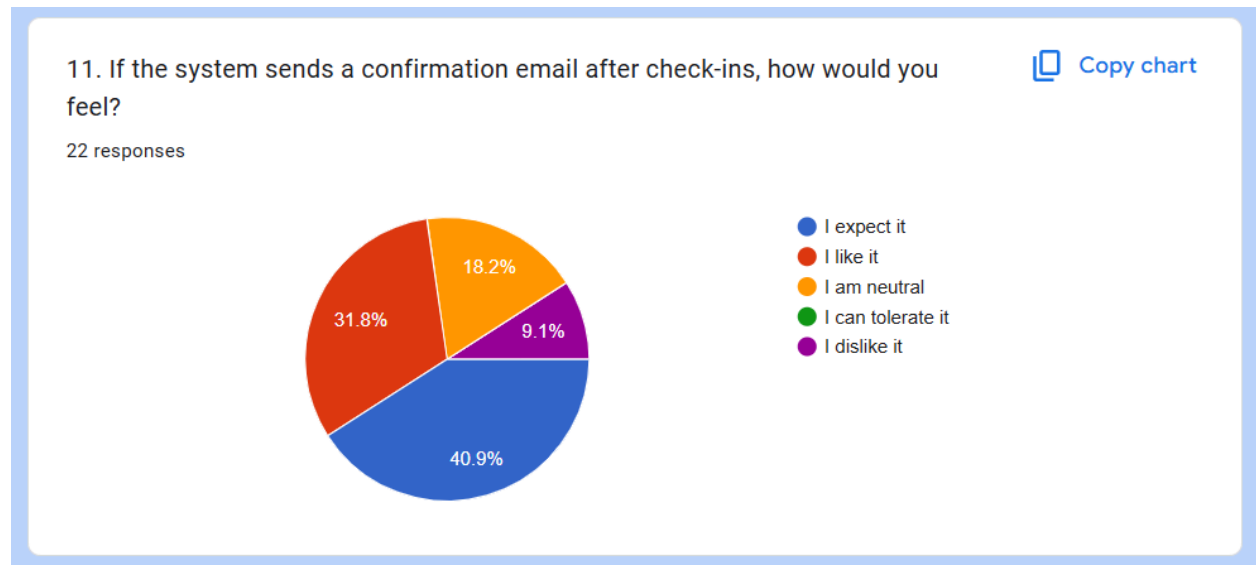
The collected data shows that over half (59.1%) of potential users are uncomfortable having their information stored in a database for future use. While 13.6% are neutral and only 27.3% are comfortable with it.



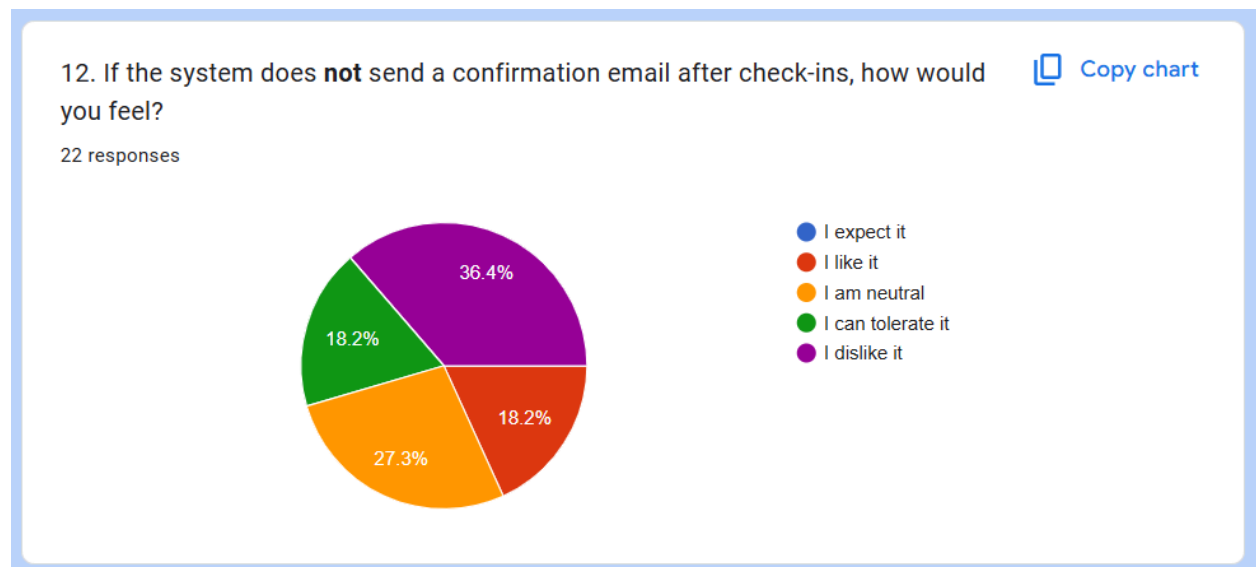
Based on the responses, 68.2% of respondents are comfortable with not having their information stored in a database. Although 27.3% are neutral, this information is enough to tell that the feature would not be worth developing for the system.

Since most of the potential users do not want this feature to be implemented, it is unclassified and won't be developed for the system.

## Feature 6: Confirmation Email After Check-Ins.



Based on the responses, 72.7% from the total chart expect/like the feature of sending confirmation emails. While 18.2% are neutral and 9.1% do not like the feature being implemented. From the majority of the answers, this feature is worth developing for the system.



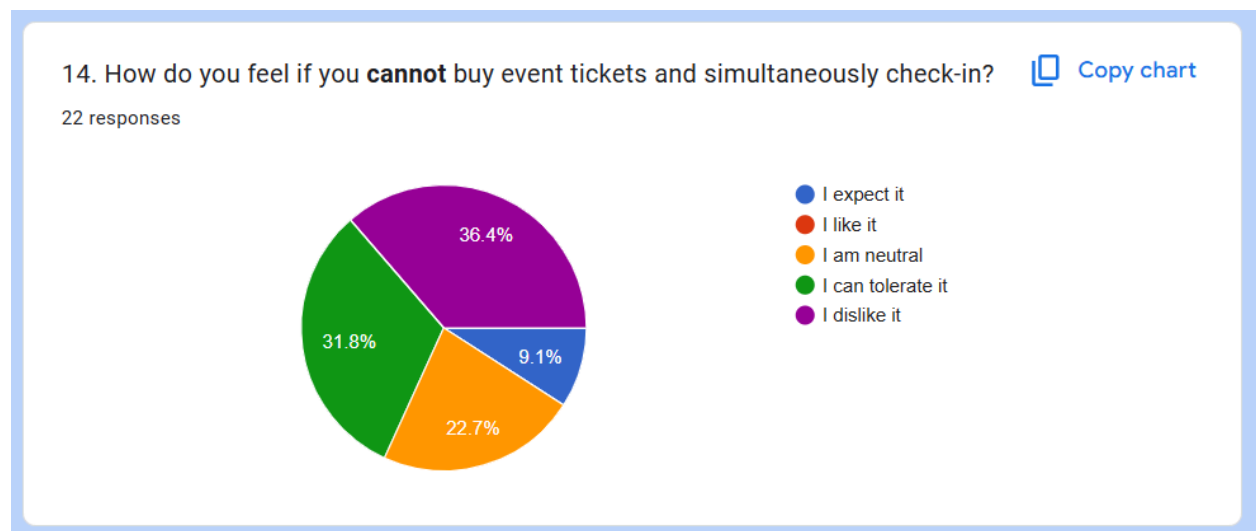
The collected data shows that 54.6% dislike not having a confirmation email after check-ins, while 27.3% are neutral and 18.2% like not having a confirmation email.

Since more than 70% of users expect the system to have this feature and 54.6% dislike not having the feature. That makes it a **dissatisfier** classification feature.

## Feature 7: Buying Event Tickets & Checking In Simultaneously.



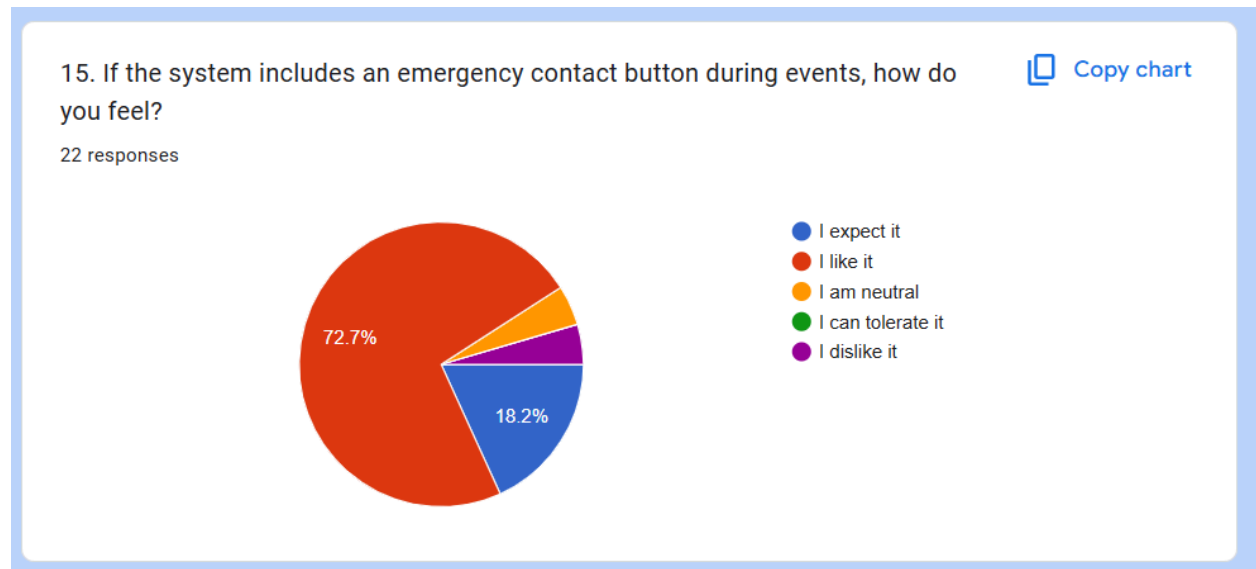
Over 90.9% of respondents like the feature of buying event tickets and being able to simultaneously checking in, which proves that this feature is definitely worth developing.



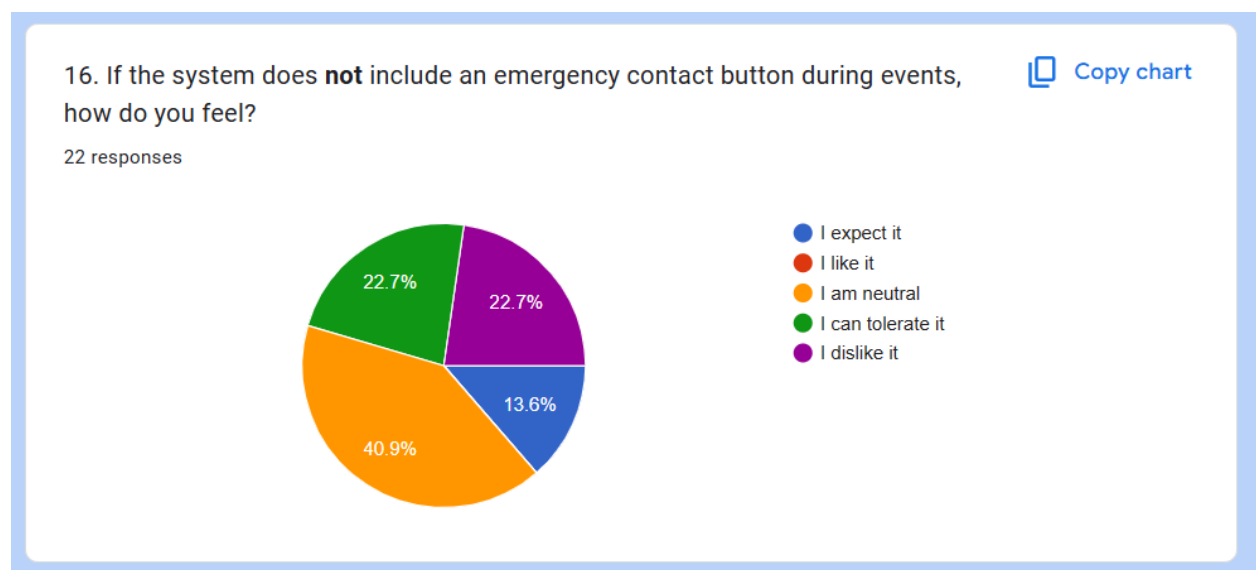
Based on the answers, 36.4% dislike not having the feature while 31.8% can tolerate it, with 22.7% being neutral and 9.1% expect not having the feature.

Since over 90% of users like the feature and over 60% of users dislike not having the feature, it signifies that it is definitely a **dissatisfier** classification.

## Feature 8: Emergency Contact Button.



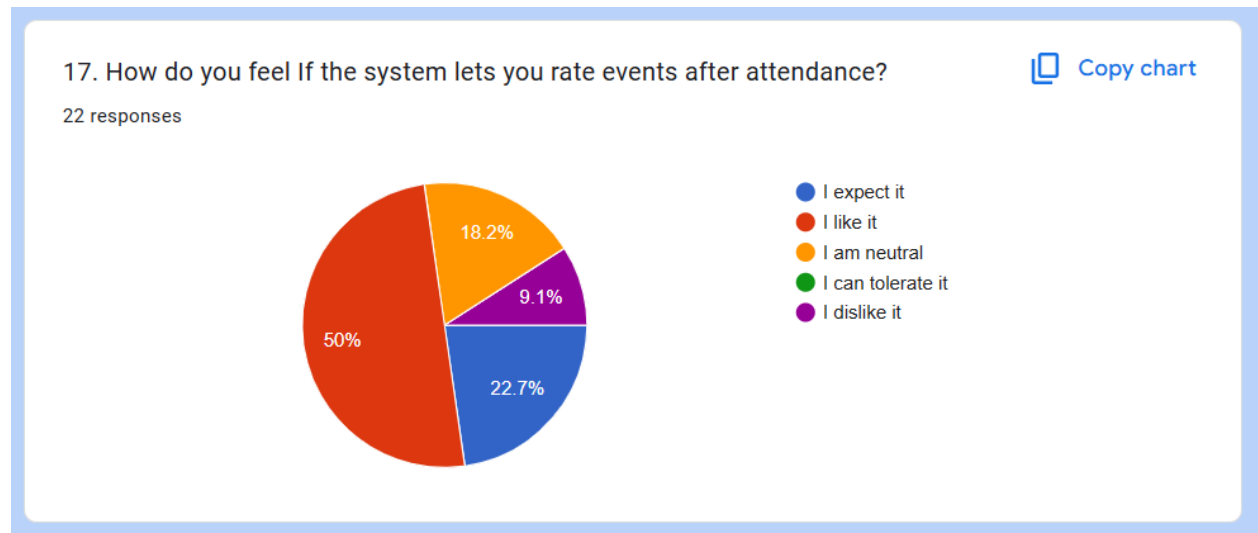
Based on the respondents, over 90.9% of potential users like the idea of having an emergency contact button during events. This data shows that the feature is worth developing.



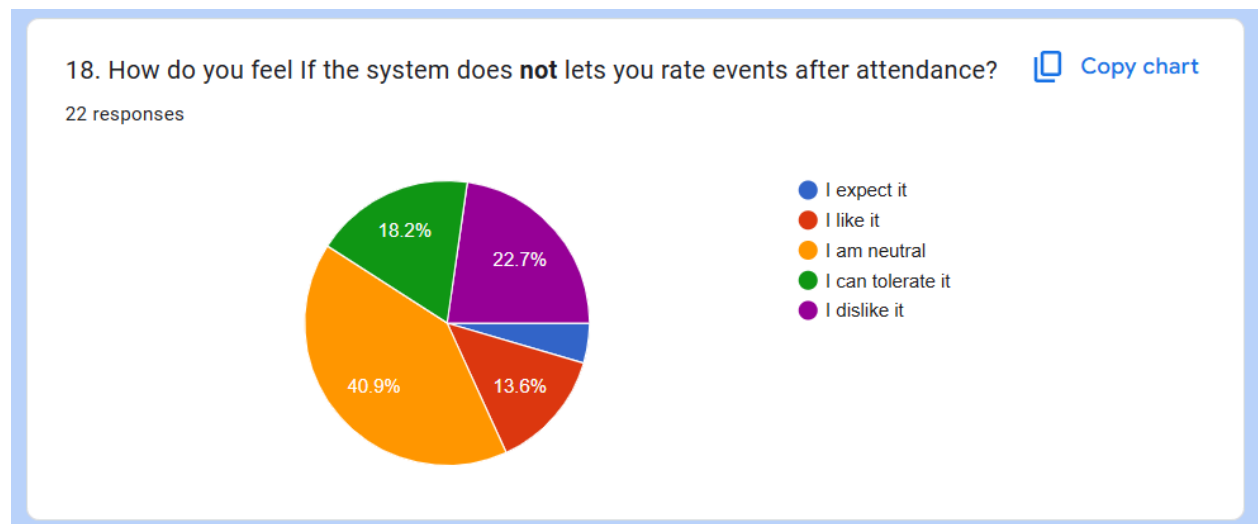
From the data we collected, 40.9% are neutral about not having an emergency contact button, while 45.4% dislike not having said feature, and only 13.6% expect the feature to not be included.

All things considered, with 90% potential users approving of the feature and over 40% of them don't like that it's not implemented. It's safe to assume that this feature is a **dissatisfier**.

## Feature 9: Rating System.



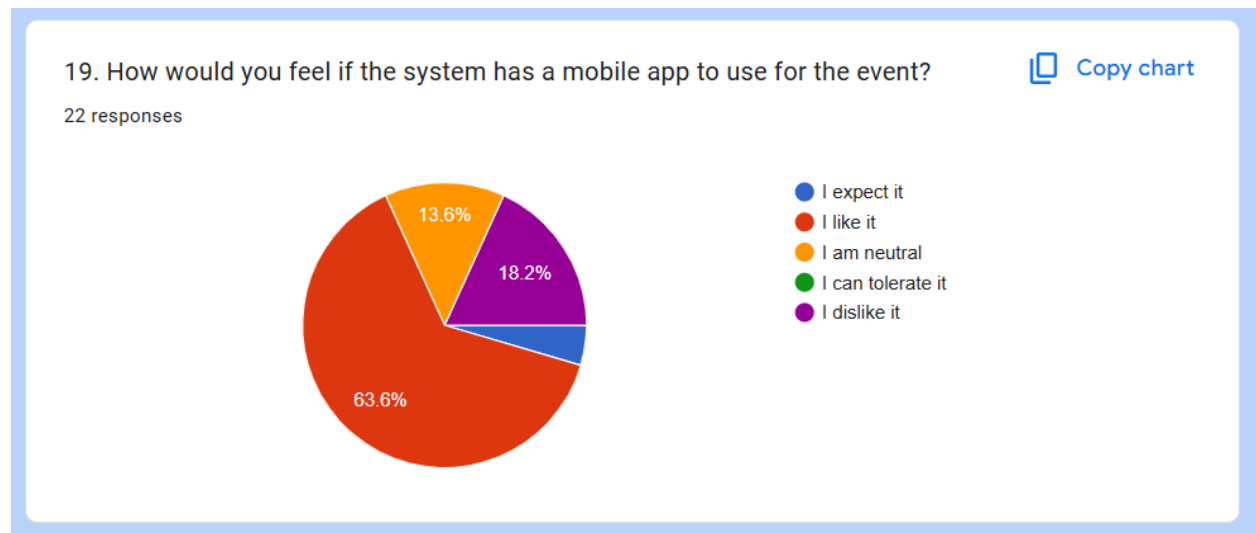
The data collected shows that 72.7% of potential users like/expect the feature, with 18.2% being neutral and 9.1% dislikes it.



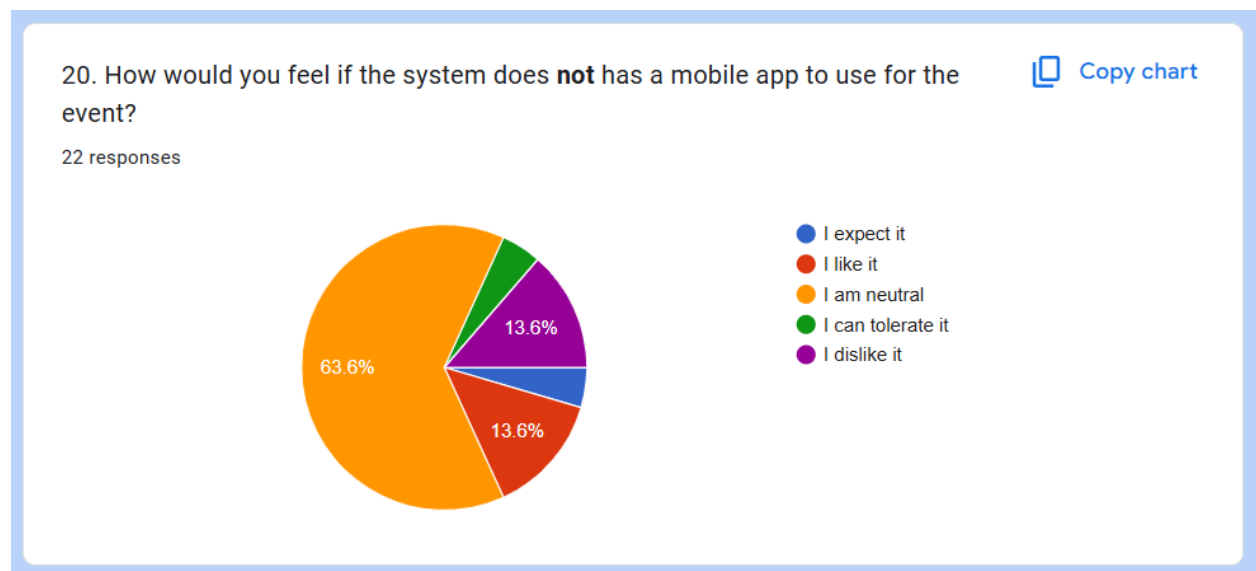
The responses are heavily in the neutral, as 40.9% are neutral with the feature not being available. While 40.9% of users don't like that it's not available, and 18.1% like that the feature is not implemented.

Considering the threshold with 72.7% of users would like to have the feature and 40.9% dislike not having it, it is considered a **satisfier** classification.

## Feature 10: Mobile App For The System.



Based on the responses that we got, 68.1% of potential users would like to have a mobile app to use during the event. With 13.6% neutral responses and 18.2% dislike having it.



Data collected shows that the feature would be heavily neutral (63.6%), with 18.1% of people dislike not having the feature and 18.1% of people like that the feature is not implemented.

Due to the heavy neutral responses, we can conclude that this feature would most likely be a **delighter** classification.



## Conclusion

From the survey that we conducted, we were able to get a lot of useful data and information leading to the development of the Ceria Check-In & Payment System.

The table below shows the feature and classification we deduced based on the responses of the survey using the Kano Model.

Feature	Classification
The Use Of Student ID For Event Check-Ins	Delighter
Using QR Code To Scan For Event Check-Ins	Satisfier
On-The-Spot Payment System	Dissatisfier
Real-Time Notifications For Updates	Dissatisfier
Storing Information In Database For Future Use	Unclassified (won't be developed)
Confirmation Email After Check-Ins	Dissatisfier
Buying Event Tickets & Checking In Simultaneously.	Dissatisfier
Emergency Contact Button	Dissatisfier
Rating System	Satisfier
Mobile App For The System	Delighter

With the **exception** of feature 5 which is not wanted by the respondents, we were able to collect **5 dissatisfiers, 2 satisfiers, and 2 delighters** which will be developed for the system.

## 6.0 How This Will Be Used In Our Software

The elicited requirements guide the software development in focusing on modular integration of student authentication, payment tracking, and event management. A modular system architecture will enable seamless integration of essential components such as student authentication, payment tracking, event creation, QR check-ins, and sales tracking, ensuring the software is scalable, secure, and capable of real-time operations. User interfaces will be intuitively designed and tailored to the specific needs of students, vendors, and administrators, enhancing overall usability and accessibility. Feature prioritization will be driven by Kano analysis, helping the team identify which functionalities are critical for early implementation, such as core event management and student ID database integration, and which features can be added in subsequent phases to increase user satisfaction. The development roadmap will be informed by clear, user-centric objectives, ensuring efficient planning and execution. Additionally, the requirements will serve as a benchmark for validation and testing, supporting user acceptance testing and feedback loops that refine the product and maintain alignment with stakeholder expectations throughout the development lifecycle.