International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

A REAL-TIME PARENT-CHILD CONSENT PLATFORM FOR SECURE APPLICATION ACCESS USING MQTT AND JWT.

Nagma Khan[1], Shakila Siddavatam[2]

[1] Master's Student, Department of Computer Science, Abeda Inamdar Senior College, India

[2] Head of Department, Department of Computer Science, Abeda Inamdar Senior College, India

Abstract:

Children are using apps more for learning, playing games, and chatting. This can create problems. They may spend too much time on screens. Their privacy may be at risk. They might see content that is not suitable. Most parental control tools use fixed rules.Some tools block apps completely, while others rely on strict time limits. These controls are simple but not flexible and do not always meet the needs of children or parents. These tools are not flexible and do not let parents respond in real time. They also do not involve children in making decisions.

This paper introduces a real-time Parent–Child Consent Platform. Children can ask for permission before using apps. Parents can approve or deny the requests right away. The system uses MQTT to send messages fast between the child and parent screens. JWT is used for login. Only users who are allowed can log in. For the screen the user sees, we use React and Next.js. The server side works with Node.js and Express. MongoDB keeps the data.

Tests show the system works well. Requests are handled in real time. Access is kept secure. Parents can see what their children are doing more clearly. By combining fast interaction with secure login, this platform provides a practical and easy-to-use solution for managing children's app access.

Keywords:

Parental Control, Consent Management, Child Privacy, MQTT, JWT, Real-Time Communication.

1. INTRODUCTION:

Children now use digital devices almost daily. Mobile phones, tablets, smart TVs, and online platforms have become an integral part of children's lives and are commonly used for schoolwork, video streaming, gaming, and communication. While these technologies provide educational and entertainment benefits, their increasing use has also raised several concerns related to children's safety and well-being [3].

Many children spend excessive time on screens, which can negatively affect their physical health, mental development, and social behavior. In addition, children may be exposed to age-inappropriate content, and their personal data may be collected without sufficient awareness or consent [3]. As a result, privacy and online safety have become major concerns for parents. Parents often feel uncertain about how digital applications are being used, particularly when application use is required unexpectedly or outside predefined schedules.

To address these issues, parental control tools are widely used to block applications, restrict content, or limit screen time. However, most existing systems rely on static rules such as permanent blocking or fixed time limits, which do not support temporary access or real-time decision-making [3]. Furthermore, these systems generally exclude children from the permission process, which can lead to frustration and attempts to bypass imposed restrictions.

In response to these limitations, this research introduces a real-time Parent–Child Consent Management System. The proposed system allows children to request access to applications when needed, while parents can approve or deny such requests instantly. Real-time communication is supported using the MQTT protocol, and secure access is ensured through JWT-based authentication. This approach provides a flexible, transparent, and secure solution for managing children's digital access while maintaining a balance between parental supervision and child independence.

2. Problem Definition:

Most existing parental control systems use fixed rules and cannot handle decisions in real time. Parents and children cannot communicate directly through these systems, and temporary access is often not possible. Many systems do not have up-to-date security. This can let someone access data without permission. We need a platform that is simple. Parents and children can talk to each other in real time. Permissions can be given when needed. Data should stay protected.

2.1 Limitations of Existing Systems

Current parental control systems largely depend on static policies that are not adaptable to real-time requirements. They emphasize monitoring over communication and often lack flexible, time-bound access mechanisms. In addition, proper attention is not given to secure authentication and real-time interaction. Due to this, the system shows poor reliability and slow response time. It becomes difficult to use it properly in dynamic situations.

3. Literature Review:

Previous studies on parental control, consent handling, child data privacy, real-time communication, and secure authentication show that current methods have both strengths and shortcomings. The review also points out several gaps, which the proposed Consent-as-a-Service system attempts to overcome.

3.1 Parental Control Systems

Early parental control tools relied on content filtering, application blocking, and screen-time limits [3]. Such static approaches often fail to adapt to dynamic scenarios, like temporary educational use [3], and raise privacy concerns by collecting excessive data while excluding children from decisions [3].

3.2 Consent Management

Consent frameworks improve transparency and control in digital systems. Policy-based models define access rights but lack real-time adaptability and require technical expertise, limiting their usability for parents [3].

3.3 Blockchain-Based Consent

Blockchain ensures immutable consent records and enhances accountability [4], but introduces high computational overhead and latency, reducing practicality for real-time parental control applications [4].

3.4 Privacy, Real-Time Communication, and Authentication

Many parental control apps compromise child privacy [3]. Lightweight protocols like MQTT support low-latency consent messaging [1], and JWT provides secure, scalable authentication [2], though integration with real-time consent remains limited.

3.5 Research Gap

People have looked at parental control and child digital safety in different ways, but there are still problems. Most systems today follow fixed rules, like blocking apps or setting screen time limits. They cannot give temporary access or change the rules for different situations. These systems also do not let parents interact in real time. This means parents cannot respond immediately when children need access. Some systems let people give consent. Most of the time, they only control data. They can also be hard to use every day. Blockchain can make things safer, but it is slow and requires significant computing power. Therefore, it is not suitable for real-time tasks. Many systems watch children a lot. This can cause privacy problems. It can also make parents and children trust the system less. One big gap is that there is no single system that combines real-time communication with secure authentication methods, like MQTT and JWT, to handle parental consent in a flexible way.

4. Methodology:

The proposed Parent–Child Consent Platform was developed by first identifying practical requirements from real-life usage scenarios. Based on these needs, a suitable system structure was planned, followed by the implementation of a solution that supports secure access and real-time communication, as suggested in existing consent-based and secure system studies [3].

4.1 Information Gathering and Requirement Analysis

In this phase, existing parental control systems and consent-based access models were studied to understand their working and limitations. Most existing systems mainly use fixed controls such as permanent application blocking or preset usage limits and provide very limited real-time interaction [3]. From this analysis, key system requirements were identified, including secure authentication, real-time permission handling, role-based access for parents and children, and reliable storage of consent decisions. Defining these requirements early helped ensure that the system design was based on real-world usage needs rather than assumptions, as recommended in prior privacy and access control research [2].

4.2 System Design Approach

Based on the identified requirements, a client–server design approach was adopted to support modularity and scalability. Separate interfaces were designed for parents and children to simplify interaction and improve usability. The backend of the system manages authentication, permission requests, and data storage. JSON Web Tokens (JWT) were used to enable secure and stateless user authentication, a widely accepted approach in modern web applications [2]. The MQTT protocol was selected for communication because it supports lightweight and low-latency message exchange, making it suitable for real-time systems [1]. This design approach enables smooth communication between parents and children while maintaining security.

4.3 System Implementation and Validation

The frontend of the system was made with React and Next.js. This lets the screens adjust to different devices. The backend part of the system uses Node.js and Express to handle server tasks. It handles the system logic and controls who can access what. MQTT was integrated to enable instant transmission of consent requests and responses between parent and child interfaces, consistent with real-time communication practices discussed in existing studies [1]. MongoDB was used for securely storing user data and consent logs. Functional testing was carried out to validate authentication, real-time communication, and correct handling of approval and denial decisions. The results confirmed that the system operates reliably with minimal delay and secure data handling, similar to observations reported in secure distributed system implementations [5].

5. System Architecture:

The platform is designed using a client–server setup and is divided into five main parts.

Child Interface

Parent Dashboard

Backend Server

MQTT Broker

Database

Child Interface: The Child Interface is a web-based frontend through which the child interacts with the system. It allows the child to log in securely and request permission for accessing restricted applications or actions. The interface does not grant direct access but acts as a request generator. When a request is submitted, it goes to the backend and is sent to the MQTT broker.

What it does:

Secure login using JWT

Application/action selection

Consent request submission

Receiving parent's response

Parent Dashboard: The parent dashboard lets parents see and control consent requests easily. It sends a notification as soon as a child makes a request. Parents can check the details and decide right away whether to allow or deny access.

What it does:

Secure login and authorization

Real-time request reception

Approve/Deny decision handling

Viewing request history and status

Backend Server: The backend server does most of the work. It handles requests and database tasks and connects to the MQTT broker. Only verified users can use the system.

What it does:

JWT token generation and validation

User role management (Parent / Child)

Request processing logic

Database interaction

MQTT publish/subscribe handling

MQTT Broker: The MQTT broker helps the child and parent interfaces talk to each other in real time. It uses a publish–subscribe model, which allows messages to be sent quickly with very little delay. The broker ensures that consent requests and responses are delivered instantly without continuous polling.

What it does:

Message routing

Topic-based communication

Real-time data delivery

Database: The database stores user credentials, consent requests, and decision logs. Only essential information is stored to maintain privacy. The stored data allows parents to review past decisions and system behavior.

What it does:

User data storage

Consent request logs

Decision status tracking

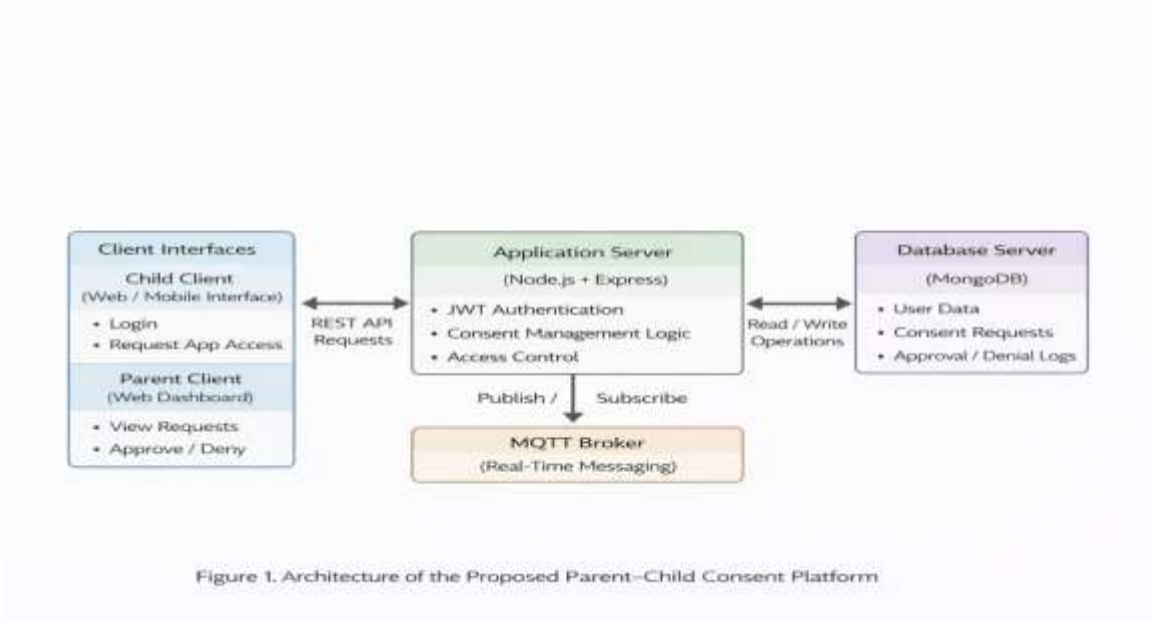
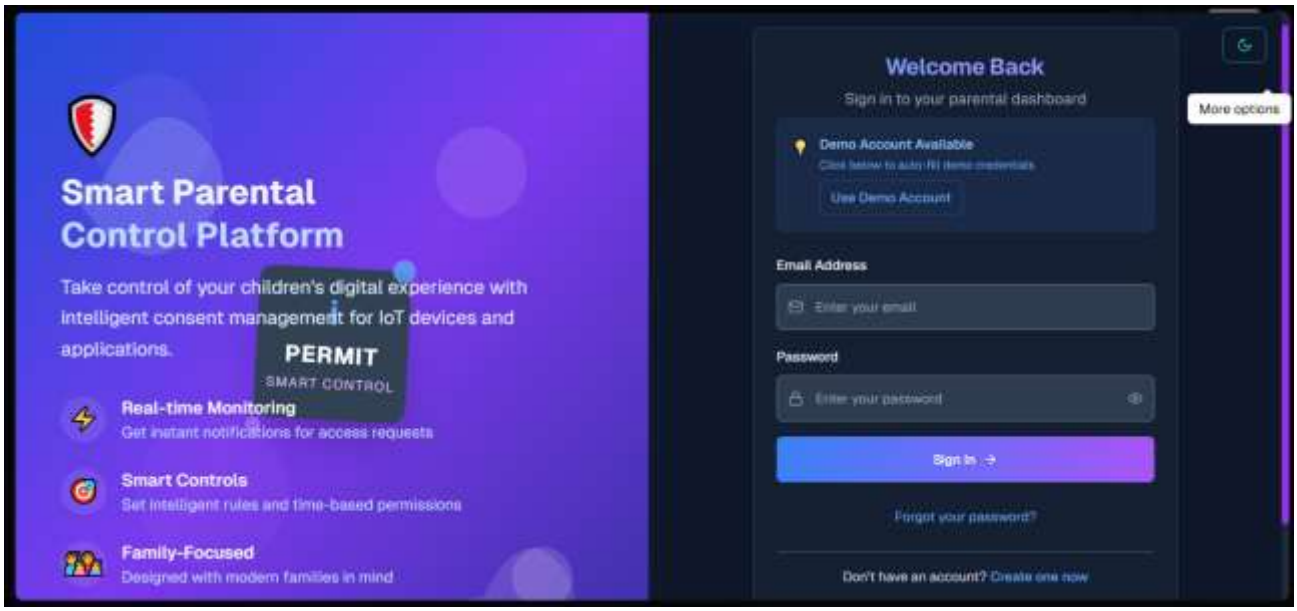


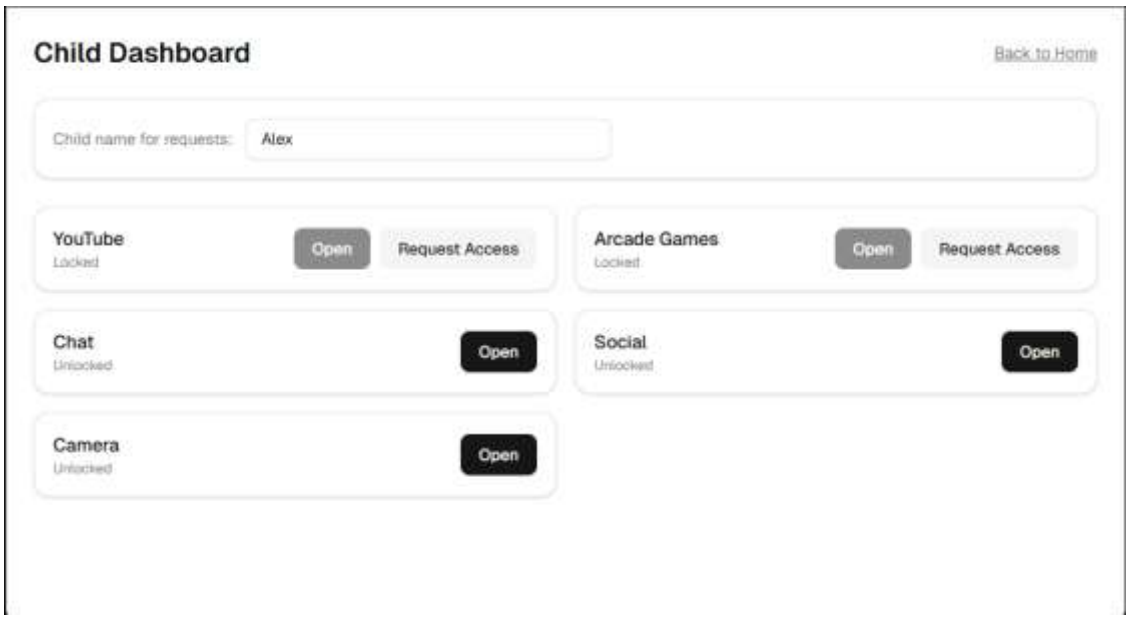
Figure 1. Architecture of the Proposed Parent–Child Consent Platform

5.1 User Interface Implementation

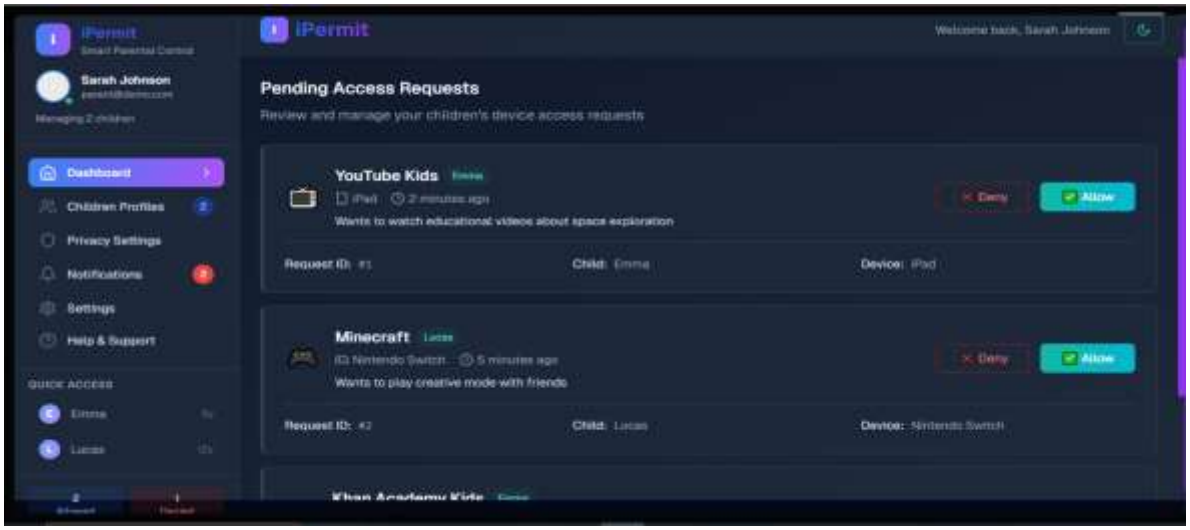
The system is developed using simple web technologies. React and Next.js are used to build the frontend screens. The backend handles all requests and checks user access using JWT. MQTT is used so messages are shared quickly between parent and child without waiting. The main focus during implementation was security and making the system easy to run and manage.

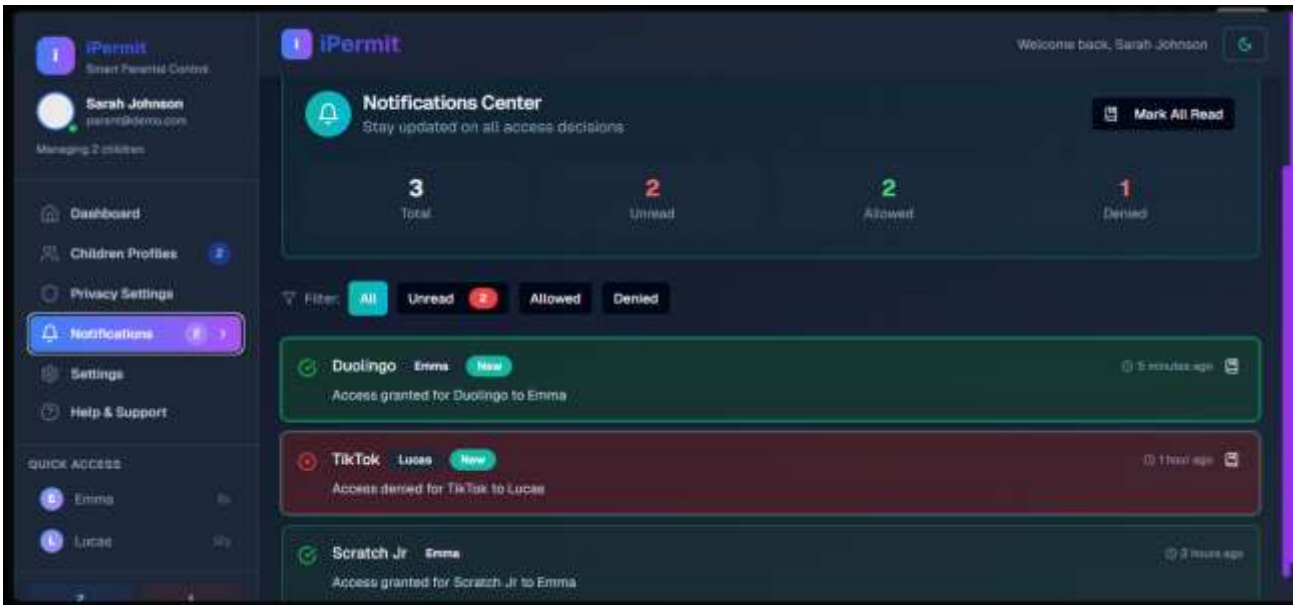
5.2 UI Screenshots





6.





Comparison with Existing Systems:

6.1 Hardware Requirements:

❖ Component	❖ Minimum Requirement
Processor	Intel Core i3 or higher
RAM	4 GB or higher
Hard Disk	250 GB minimum
Display	1366 × 768 resolution or above
Network	Stable internet connection for MQTT communication

6.2 Software Requirements:

❖ Component	❖ Description
❖ Operating System	Windows
❖ Frontend Framework	Next.js (Parent) and React.js (Child)
❖ Backend	Node.js and Express.js
❖ Database	MongoDB for storing user credentials, child profiles, and access requests
❖ Communication Protocol	MQTT (Message Queuing Telemetry Transport) for real-time communication
❖ Authentication	JWT (JSON Web Token) for secure login and user verification
❖ Development Tools	Visual Studio Code, Postman, MQTT Broker (Mosquitto / HiveMQ)
❖ Browser	Google Chrome or any Chromium-based browser

7. Test Cases and Validation

Test Case 1: User Login

I logged in using a valid account. The dashboard came up and worked okay. Loading took a little longer at the beginning, but it wasn't a problem.

Test Case 2: Sending a Consent Request

A child selected an app and sent a request. It showed up on the parent's dashboard very quickly and was saved in the database without problems.

Test Case 3: Real-Time Notifications

The parent dashboard got the requests almost immediately.

It got a bit slow sometimes, but it still worked okay.

Test Case 4: Handling Decisions

When the parent approved or denied a request, the child saw the result right away. The system also saved a record of it.

Overall, the testing confirmed that login was secure, messages moved quickly, and the system reliably stored all data. A few minor delays were noticed, but nothing significant affected usability.

8. Results and Discussion

The developed system was tested to evaluate how effectively it handles real-time communication between the child interface and the parent dashboard. During the testing phase, permission requests generated by the child interface were transmitted to the parent dashboard very quickly. This quick transmission allowed parents to review the requests and either approve or deny them without any visible delay. The authentication mechanism

functioned properly during testing and confirmed that access to the platform was restricted to valid and authorized users only.

In contrast to many traditional parental control systems that depend on fixed rules such as permanent blocking or predefined screen-time limits, the proposed system provides a more flexible approach for managing access. Instead of depending entirely on static restrictions, parents are able to make decisions according to the current situation. This approach improves the practicality of the system and also makes the decision process clearer for both parents and children.

The evaluation results show that combining real-time communication with a secure authentication mechanism provides an effective way to control children's access to applications or device features. During testing, the platform maintained stable communication between components, supported secure login, and successfully recorded the consent decisions made by parents. These results suggest that the system can operate reliably in real-world parental control scenarios.

9. Conclusion

This paper introduced a platform that helps parents and children manage permissions in real time. Instead of set rules, parents can decide on requests as they come, making it more flexible. The system uses MQTT so messages reach instantly and JWT to keep logins secure. It is simple to use and helps parents protect their children online. Experimental evaluation confirmed that the system effectively supports real-time request–response interaction, secure authentication, and reliable data handling. The modular architecture and privacy-aware design further enhance system usability and scalability. Overall, the proposed solution offers a practical, lightweight, and secure alternative to traditional parental control systems and contributes to improved digital safety for children.

10. Future Scope

The proposed system supports real-time, consent-based parental control, but there is still scope for improvement in the future. The system can be extended to work on mobile platforms and native apps so that it is easier to access. Machine learning can also be added to study usage patterns and help parents with useful suggestions.

Extra features like time-limited permissions, location-based consent, and support for multiple children can make the system more flexible. The system can also be connected with smart home and IoT devices to manage consent for more than just application access.

11. References

1. A. Banks and R. Gupta, "MQTT Version 3.1.1," OASIS Standard, 2014.

2. M. Jones, J. Bradley, and N. Sakimura, "JSON Web Token (JWT)," IETF RFC 7519, 2015.

3. L. Cranor, "Necessary but not sufficient: Standardized mechanisms for privacy notice and choice," Journal on Telecommunications and High Technology Law, 2012.

4. S. Sicari et al., "Security, privacy and trust in Internet of Things," Computer Networks, 2015.

5. N. Naik, "Choice of effective messaging protocols for IoT systems," IEEE International Systems Engineering Symposium, pp. 1–6, 2017.