

Effectiveness of Database with Automation apps using Appsheet for Controlling ATeC (Automotive Technology Centre) Data

Habibah Binti Remeli^{1*}, Nor 'Amirah Binti Talib², Sully Faizura Binti Mohd Rawi²

¹ Automotive Technology Centre, Politeknik Sultan Mizan Zainal Abidin, 23000 Dungun, Terengganu

² Jabatan Kejuruteraan Mekanikal, Politeknik Sultan Mizan Zainal Abidin, 23000 Dungun, Terengganu

*Corresponding author E-mail: habibahremeli@gmail.com

Abstract

The Automotive Technology Centre (ATeC) at Polytechnic Sultan Mizan Zainal Abidin struggles with disorganized data management due to the lack of a centralized digital system, causing inefficiencies across the units. The primary objectives of this study are to centralize all lecturer-related data into a single digital database and to develop a zero-cost mobile application platform for managing and automating data processes. The development process was carried out in three key phases to ensure a structured and efficient implementation. The first phase, data preparation, involved cleaning and transforming raw data into a consistent and usable format. This included tasks such as removing duplicate entries, standardizing formats, and organizing data fields to align with the overall database structure, ensuring accuracy and reliability for further use. The second phase, application design (UI/UX development), focuses on designing a user-friendly interface and intuitive user experience for the mobile application. Elements such as forms, buttons, and navigation menus were created to facilitate seamless data entry and retrieval, making the app accessible and efficient for users on mobile devices. The final phase, report generation, involved developing an automated system to create structured reports based on the collected data. These reports are generated in PDF format and automatically stored in Google Drive, allowing for easy access, sharing, and documentation. The analysis shows that the core performance of AppSheet received an average score of 4.13, indicating a high interpretation. Similarly, the data control capabilities of the platform achieved an average score of 4.00, also reflecting high interpretation. A comparison between Google AppSheet and traditional software development highlights the key reasons for selecting AppSheet as the preferred platform for this project. In terms of cost, AppSheet is either free or low-cost when used with Google Workspace, whereas traditional development often incurs high expenses due to developer fees, hosting services, and software licensing. Regarding development time, AppSheet offers a fast, no-code environment that allows for rapid application creation, while traditional methods can take weeks or even months to complete due to custom coding and testing. Technical skill requirements are minimal with AppSheet, making it accessible to non-programmers. In conclusion, this mobile solution provides a streamlined, cost-effective method for managing academic data, enhancing operational efficiency and decision-making within the ATeC environment.

Keywords: Appsheet, Googlesheet, database, nonprogrammer

1.0 Introduction

Smartphones are now used for more than just making and receiving calls and messages. They perform a variety of functions to support human needs, including small computer systems, digital tools, audio recording devices, and photography equipment. In the modern era, people carry their portable electronics with them everywhere they go and incorporate them into many aspects of their daily activities. Global occupational frameworks are significantly impacted by this phenomenon, which has been shown to increase operational efficiency and enable quick, accurate results. Smartphones are used to observe tasks, giving users the ability to obtain immediate feedback that can be used as actionable input. By integrating portable devices, appropriate mobile software can be developed to meet user requirements and improve procedural workflows. People in the modern era carry their mobile devices with them everywhere they go and incorporate them heavily into their daily routines. This has a significant impact on global work systems, as it has been shown to improve operational efficiency and facilitate quick, accurate results. Users can get real-time results as actionable inputs by using smartphones to track tasks. Developers can use portable devices to optimize processes and create mobile applications that meet user requirements. Regardless of a team's physical location, streamlining workflows and encouraging greater collaboration are two benefits of optimizing processes with portable devices. Organizations can therefore use these developments to increase overall productivity and flexibility in a constantly changing digital environment.

1.1 AppSheet Data Management

AppSheet revolutionizes data management by enabling users to build custom, no-code applications that seamlessly integrate with diverse data sources—from spreadsheets (Google Sheets, Excel) to cloud databases (SQL, Salesforce) and APIs. It turns raw data into dynamic apps that can be accessed online or on mobile devices for CRM, inventory tracking, field data collection, and other uses. Among the essential data management features are:

- i. **Centralized Control:** Ensure real-time accuracy by removing silos and syncing live data across sources.
- ii. **Automated Workflows:** Simplify operations by triggering actions (such as notifications and approvals) in response to data changes.
- iii. **Strong Security:** Establish user roles with audit logs, compliance tools, and granular permissions (view/edit access).
- iv. **Data Integrity:** To keep datasets clean, enforce validation rules (such as required fields and dropdown menus) and error alerts.
- v. **Offline Capability:** Gather and update data without internet access; automatically synchronizes upon reconnecting. This feature increases productivity and flexibility by guaranteeing that users can operate effectively in a variety of settings. Organizations can enhance their decision-making procedures and preserve operational continuity by facilitating smooth updates and data management.

AppSheet democratizes app development, enabling teams to effectively manage data, minimize manual errors, and transform insights into actionable workflows without the need for coding knowledge. In addition to streamlining procedures, this strategy encourages departmental cooperation, which helps organizations quickly adjust. In the end, AppSheet offers a flexible platform that revolutionizes the way teams work with data and create solutions specific to their own problems.

1.2 Issue and Challenge

ATeC's governance has five (5) principal leaders and seventeen (17) subordinate units, each overseen by unit heads, with each unit including three to four personnel from four primary departments. The substantial volume of appointments and 17 minor components complicates the collection of physical data (primary files) and digital data (electronic files). Data will generally be consolidated on a single Uniform Resource Locator (URL), specifically Google Drive, accessible to all individuals. Nonetheless, the Google Drive link is challenging to access during crucial periods, such as the auditing process. The absence of internet and inadequate connectivity further impede access to the link. Frequently, by disseminate various data, including images and videos of activities, committee and student certificates, memos, and correspondence within the WhatsApp Group. An increase in shared data complicates access for certain members, resulting in the potential oversight of information. Consequently, a centralized data gathering center must be established to aggregate data that can be accessed and shared by the organization.

2.0 Literature Review

Low-code/no-code (LCNC) platforms have rapidly emerged as transformative tools in the realm of application development, allowing non-technical users (often referred to as “citizen developers”) to create fully functional apps through visual interfaces without writing traditional code. AppSheet, which was acquired by Google in 2020, is a prime example of this category. It instantly interfaces with Google Sheets, Excel, and SQL databases, allowing businesses to digitize workflows with little technical effort. Liu et al. (2024) found that LCNC platforms use

abstracted logic layers and established templates to reduce coding errors and speed up software delivery by 3-10 times. According to Forrester. (2023), firms saw over 90% reduction in manual entry mistakes and considerable improvement in process speed, especially in inventory and data monitoring. Core strength of AppSheet :

- i. Fast Development and Deployment : AppSheet turns Google Sheets and Excel files into mobile apps in hours, circumventing development periods (AppSheet, 2022)(Forrester,2023).
- ii. Empowering Non-Technical Staff : These tools enable departmental users to customize solutions, boosting operational agility and decentralizing digital innovation (Kirchhof et al., 2023).
- iii. Live Synchronization and Automation AppSheet automates report production, delivers email and WhatsApp notifications, synchronizes data across devices, and automates workflow with event-based triggers (Forrester, 2023).
- iv. Offline security and access Offline data collection, synchronization upon reconnecting, fine-grained user access roles (e.g., view-only, edit), field-level data masking, and IP-based access controls are crucial in settings with unstable internet (AppSheet, 2022).
- v. Cost efficiency and free licence

2.1 Comparative Analysis

Table 1: Comparison between AppSheet and Traditional Software Development

Feature	AppSheet (No-Code Platform)	Traditional Software Development
Setup Time	Hours to a few days	Several weeks to months (depending on project scope)
Technical Expertise	Minimal (no coding required)	High (programming knowledge needed: Java, Python, etc.)
Customization Flexibility	Limited (within AppSheet's UI/UX constraints)	Highly customizable (full control over front/back-end)
Real-Time Sync	Native (built-in Google Sheets/Cloud integration)	Requires additional configuration (API, backend logic)
Analytics and Reporting	Basic (via AppSheet's built-in features, Google Data Studio)	Advanced (custom analytics engine, database queries)
Offline Access	Built-in offline mode for mobile data entry	Requires explicit implementation and caching strategies
Security Features	Role-based access, data masking, IP restriction	Fully customizable encryption, access policies, auditing
Cost and Licensing	Free for small use cases; pay-per-user model	Higher cost: developer salary, hosting, licensing tools

Based on Table 1.0, AppSheet, a no-code platform, differs from traditional software development in speed, skill level, customization, and technical performance. The fast setup process of AppSheet lets customers construct simple mobile apps in hours or days. This contrasts with traditional software development, which might take weeks or months for complicated systems, including requirements analysis, design, coding, testing, and deployment (Liu et al., 2024).

AppSheet's low technical barrier is a major benefit. Non-technical users or “citizen developers” including educators, administrators, and field personnel can use it without programming experience (Kirchhof et al., 2023). Instead, traditional development requires trained programmers who know coding languages, software architecture, and version control.

Accessibility has drawbacks. In its visual interface, AppSheet offers only basic customization. It lacks versatility for complex UI/UX designs and highly integrated functionality and enables limited logic manipulation. In contrast, traditional development provides full customization. Advanced data processing systems, rich user interfaces, and precise API and database integration are possible for developers (Ruscio et al., 2021). Native interaction with Google Sheets makes AppSheet great at real-time data synchronization. Built-in automation tools send messages and generate reports upon data entry. Traditional systems can mimic these characteristics, but they require additional tools or bespoke code, increasing development time and complexity (Forrester, 2023). Offline capabilities matter too. AppSheet collects data offline and synchronizes when online. Development overhead increases when custom-built programs must manually integrate these functionalities, generally through caching methods and local store management (Database Trends, 2024). Finally, cost and upkeep vary greatly. AppSheet's freemium or per-user subscription makes it affordable for small businesses and departments. It minimizes IT staffing needs. Developer salary, server infrastructure, software licenses, and long-term maintenance are more expensive in traditional development (AppSheet, 2022).

In conclusion, AppSheet is ideal for companies seeking fast, economical, and easy-to-use software solutions without technical experience. In educational or administrative settings, it excels at data collection, form submission, and workflow automation. Traditional software development is more powerful and adaptable for sophisticated logic, high-performance processing, and deep system integration.

2.2 AppSheet as a Database Management Solution

AppSheet can be used to design apps and manage databases for small to medium-sized businesses and departments who want to digitize and organize their data without complex infrastructure. AppSheet integrates seamlessly with spreadsheet platforms like Google Sheets and Microsoft Excel, simplifying data administration without SQL, database servers, or trained developers. These spreadsheets are the backend database, while AppSheet offers the interface and logic for data access, modification, and presentation.

AppSheet excels in real-time data synchronization. A mobile or web interface change immediately updates the data source. This maintains uniformity across all users and devices, which is useful in dynamic situations like academic or technical departments where several people submit or review data (Forrester, 2023). AppSheet may also be used offline to collect data in the field and sync it automatically when reconnected, which is useful for fieldwork or remote education institutions with intermittent connectivity.

Data validation, user access control, and process automation are essential to database management, and AppSheet enables all. Mandatory fields, selectable options, and error notifications assure data accuracy and consistency. Administrators can restrict data viewing, editing, and deletion with role-based permissions (Kirchhof et al., 2023). AppSheet may also automate email notifications, PDF reports, and record updates under certain conditions. These features decrease manual interventions, avoid errors, and boost operational efficiency.

AppSheet solves most small-scale database needs without the full querying capabilities or sophisticated relational structures of MySQL or PostgreSQL. It lets users develop moderately relational data models without SQL code by filtering, sorting, grouping, and linking tables (called “refs”). An educational institution can utilize AppSheet to centralize and search student records, attendance logs, certificate archives, and project submissions without constructing a proprietary backend. However, there are constraints. Ruscio et al.

(2021) noted that AppSheet may struggle with huge datasets, high-frequency transactional activities, and sophisticated multi-table joins. Integration with Google Data Studio or Looker Studio is common for advanced analytics and visualizations. Developers with coding experience may find AppSheet's abstraction layer restrictive when trying to apply custom logic beyond formulae and expressions.

In conclusion, AppSheet is a practical and accessible database management solution for moderate data needs, low technical staff, and speedy deployment. Its integration ease, offline capabilities, built-in security, and automation features make it perfect for managing operational data in education, technical training, and SME environments. AppSheet can also be used as a front-end interface for lightweight data interaction in advanced systems.

3.0 Methodology

This article provides a step-by-step guide that describes how to develop a mobile application using AppSheet for the purpose of data processing. This document serves as a reference for users developing applications by leveraging the core functionalities of AppSheet. By utilising Google AppSheet and Google Sheets to automate processes in a systematic manner, this article offers a comprehensive overview of designing a system optimised for mobile devices. Figure 1 provides a description of the logical progression of the database control procedure

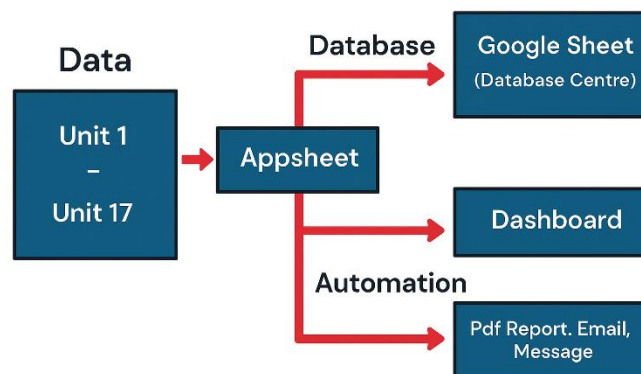


Figure 1: The flow of database using Appsheet

The order processing system's development was segmented into three phases: data preparation, mobile application development, and report customisation, as seen in Figure 2.

Phase 1 (Data Preparation): As a starting point, users can create a form using Google Forms (Figure 3) in the usual way. The form responses should be directed to a Google Sheet, which acts as the database centre. Once this is set up, the user can navigate to the "Add-ons" section and search for "AppSheet" to begin linking the form to a mobile application interface.

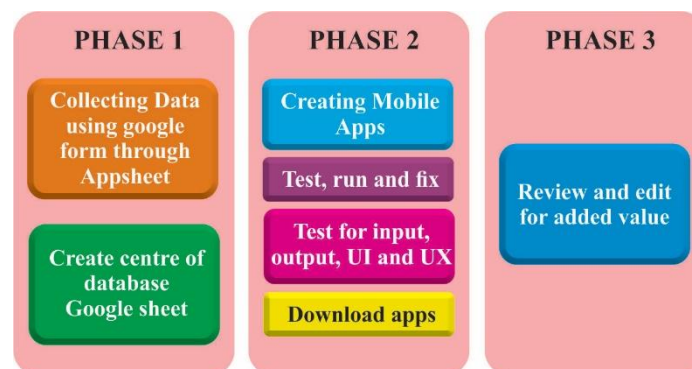


Figure 2: The flow of developing apps.

A	B
1 MENU	ICON
2 DAFTAR PENGGUNA	https://lh3.googleusercontent.com/1a0pGp_FSCuUP3hcaRcbKQg29CAGQJ5
3 PENGURUSAN DOKUMEN	https://lh3.googleusercontent.com/1HYCubQIRAGH0b0Eneef7N4K2z9Ss4S8
4 PENGURUSAN AUDIT	https://lh3.googleusercontent.com/1BEMCg0sBucG0099PzP-C0saSu1Y490
5 PENGURUSAN KEWANGAN	https://lh3.googleusercontent.com/1r0X0D04810F0gY7m547a15H4Pm2H2W9_5A
6 EXPERT SERVICE TEAM	https://lh3.googleusercontent.com/1z1M0Vg0QWQWn5Q0xvVcD0HwEsf8seeH
7 RACING TEAM	MENU Images/RACING TEAM ICON 110428.jpg
8 SPONSOR TEAM	https://lh3.googleusercontent.com/1Cb0yB4t2_ia_4Zt5G0hm8TWX-r8Bw
9 JUALAN TEAM	https://lh3.googleusercontent.com/1ZDMXMa0VUa1TM8qN77H07WTW48H1Nf
10 INCUBATOR AUTOMATIVE TEAM	https://lh3.googleusercontent.com/1NT0brnK2Q4C0Z60U1ZcGBApKEN_FKvU
11 PENYELIDIKAN TEAM	https://lh3.googleusercontent.com/1p0kY5CuNPQWn59g0M0a10qV09pG0g
12 PELANGGAN	https://lh3.googleusercontent.com/1r0X0D04810F0gY7m547a15H4Pm2H2W9_5A
13 ADUAN KEROSAKAN	https://lh3.googleusercontent.com/1wAbe5MzwF98deu6Cwomh3F0m0MQ
14 BAYARAN	https://lh3.googleusercontent.com/1z1M0Vg0QWQWn5Q0xvVcD0HwEsf8seeH
15 SEMAK REWARD	https://lh3.googleusercontent.com/1Z603Qo_K90Kf3na1yZ0YK0d-n5DXVc
16 REWARD TOP	https://lh3.googleusercontent.com/1Q5QLdUaIPxCOYFNN_P0hQhF2TS_8cf
17 HEBAHAN MEDIA/EBOOK	https://lh3.googleusercontent.com/1UuDAH0ARyFRX1H0p07m0K0ffwGyn

Figure 3: Input using Google Form

Phase 2 (Mobile App Development): A new AppSheet project is created by connecting it to the prepared Google Sheet. On the right-hand side of the AppSheet interface, the mobile app preview is displayed, and users can begin customising the layout. This includes renaming the app, selecting data views, adding logos, and choosing between display formats such as deck, table, dashboard, or card. Users can organise how data is grouped, sorted, and displayed through the left-side menu. Once the UI/UX configuration is complete, the app can be tested, and any errors highlighted by the platform can be resolved. AppSheet's no-code approach makes it accessible to developers with no programming experience. Upon finalisation, the application is ready to be downloaded and installed for use.

Importantly, one example of an output that can be generated through AppSheet is a financial report as in Figure 4. Once basic data is entered into the system, the generation of financial reports can be performed quickly and accurately. These reports can be easily accessed in real-time, offering significant advantages in terms of efficiency and decision-making. This feature adds practical value to AppSheet, especially for users seeking streamlined, on-demand data reporting.

Phase 3 (Report Customisation): The developer can further enhance the application to ensure it remains user-friendly and adaptable to ongoing needs, including advanced features like dynamic reporting, data summaries, and user access control.



Laporan Kewangan Program ATeC & COT

Nama Program :kawad air	
Tarikh	5/7/2025
Lokasi	padang bola
Jumlah Pendapatan Diperoleh	rm 7000
Bayaran Atas Talian	rm1000
Sumber Pendapatan	os
Sumber Pendapatan	os
Cadangan Penambahbaikan	Adalah dengan ini disahkan bahawa saya menerima dan bersetuju untuk menyumbangkan kepakaran dalam pelaksanaan P&P merangkumi perancangan, penyediaan dan penilaian P&P seperti buliran yang telah ditetapkan oleh pihak tuan. Nama aktiviti : PENYARAH PELAWAT INDUSTRI (PPI) : PROGRAM STUDENT'S PROJECT COMPETITION INNOVATION AND INVENTION DESIGN (SPEED) SESI 2: 2023/2024 Tarikh : 13 MEI 2024 Tempat : DEWAN DAGANG, POLITEKNIK SULTAN MIZAN ZAINAL ABDIN Masa : 08.00 PAGI HINGGA 5.00 PETANG

Disedikan Oleh,	Disemak Oleh,	Disahkan Oleh,
Bendahari Automotive Atec Centre, Center of Technology Politeknik Sultan Mizan Zainal Abidin	Seliausaha Automotive Atec Centre, Center of Technology Politeknik Sultan Mizan Zainal Abidin	Pengurus Automotive Atec Centre, Center of Technology Politeknik Sultan Mizan Zainal Abidin
Tarikh:	Tarikh:	Tarikh:

Figure 4: Financial Report generated

4.0 Result and Analysis

AppSheet drastically transforms database administration by enabling app creation and real-time data control, resulting in measurable operational and financial benefits. Organisations generally get 80% faster data entry using mobile-friendly forms with automatic validation, decreasing errors by more than 90% and cutting report creation time from hours to minutes. Table 2 represents a mean score interpretation scale, commonly used in surveys, research, and data analysis to interpret the average score (mean) of responses on a Likert scale, usually from 1 to 5.

Table 2: Mean score interpretation scale

Mean Score Range	Interpretation
1.00 – 2.33	Low
2.34 – 3.66	Medium
3.67 – 5.00	High

A pilot study with 50 participants produced a Cronbach's alpha value of 0.88, indicating that the instrument utilised had strong internal consistency and reliability. The results of the core performance assessment sheet in Table 3 show that the mean scores for all five tested items ranged from 3.67 to 5.00, indicating a "High" level of interpretation. Item 5, "Data syncs reliably between devices in real time," had the highest score, with a mean of 4.88, indicating that real-time synchronisation was the most appreciated feature. Almost all respondents agreed that integrating a database with automation tools like AppSheet to manage data at the Automotive Technology Centre (ATec) is both efficient and user-friendly. AppSheet's synchronisation technique is intended to assure data consistency between devices, with the goal of providing near-real-time updates. However, while real-time syncing is the aim, data volume, network circumstances, and application complexity can all have an impact on the actual speed of data updates. Overall, the system is well acclaimed for its speed, reliability, and capacity to manage complicated activities, as seen by the overall mean score of 4.13, with particularly outstanding performance in workflow management, offline capability, and big transaction volumes.

Table 3: Core performance AppSheet

No	Item	Mean	Interpretation
1	Security controls meet out compliance requirements	3.70	High
2	Offline mode works seamlessly without data loss	3.88	High
3	Complex workflows execute without error	4.17	High
4	Handles our peak transaction volume without slowdown	4.00	High
5	Data syncs reliably between devices in real time	4.88	High
Average Min Score		4.13	

Table 4 presents the results of user feedback on the system's data control capabilities. All seven items received mean scores between 3.65 and 4.76, which fall within the "High" interpretation range, showing that users are generally satisfied with the system's data control features. The average mean score is 4.09, indicating overall strong agreement from users. The item "*Real-time data synchronization across devices*" received a score of 3.76, suggesting that users agree the system performs well in keeping data updated across devices. "*Device approval management*" scored 3.87, showing users trust the system's ability to control which devices can access the data. Similarly, "*Control data loss during updates*" received 3.98, indicating users believe the system is reliable during updates. The item "*User role-based permission (edit/view/delete)*" scored 4.29, reflecting users' confidence in how the system controls access based on user roles. "*Field-level data masking*" had a score of 3.65, which is the lowest among the items, suggesting this feature may need improvement to better protect sensitive data. On the other hand, "*Automated data/report generation*" received the highest score of 4.76, showing users find this function very useful and efficient. Lastly, "*IP address access restriction*" scored 4.32, indicating strong user support for this security measure. In conclusion, the findings show that users view the system as effective in managing and securing data, particularly in automation, access control, and permission settings.

Table 4: Data Control Capabilities

No	Item	Mean	Interpretation
1	Real time data synchronization across devices	3.76	High
2	Device approval management	3.87	High
3	Control data loss during updates	3.98	High
4	User role based permission (edit/view/delete)	4.29	High
5	Field level data masking (for sensitive data)	3.65	High
6	Automated data / report generate	4.76	High
7	IP address access restriction	4.32	High
AVERAGE MIN SCORE		4.09	

AppSheet provides a robust and easily accessible platform that does not require any coding to transform spreadsheets and simple databases into functioning data management applications. This drastically reduces the barrier to entry for non-developers and enables rapid deployment of functional apps. Its core strengths lie in its mobile-first design, robust offline capabilities, and seamless integration with Google Workspace—particularly Google Sheets. These features make AppSheet highly suitable for small to medium businesses or departmental solutions that involve basic automation, field data collection, inventory tracking, or simple workflow management. The platform's cost-effectiveness and ease of use distinguish it from other similar no-code or low-code platforms.

However, AppSheet also has certain limitations, especially when dealing with highly complex business logic, sophisticated relational data structures, or large-scale enterprise workflows. While it offers some level of customization and interface control, these are generally limited compared to fully coded solutions. Additionally, advanced reporting features

often require integration with external tools, and performance may degrade when managing very large datasets.

Despite these limitations, AppSheet has proven to be an effective tool for data management at the Automotive Technology Centre (ATec). Its implementation has significantly improved the efficiency and accuracy of data handling, enabling smoother, more organised, and paperless data management processes. This not only simplifies data access and retrieval but also contributes to environmental sustainability by reducing the reliance on physical documentation.

A practical example of the system developed for ATec is the user-friendly mobile interface as in Figure 4, which allows users to easily input, access, and monitor relevant data. While the interface is intuitive and convenient to use, it is still in its initial trial phase and offers room for improvement, particularly in terms of design refinement and the addition of advanced features.

Furthermore, this AppSheet-based application functions as a centralised data system, which significantly improves the delivery and sharing of information across ATec departments. The centralised nature of the system ensures that all data is stored consistently and accessed uniformly, supporting more informed decision-making and streamlined communication.

In essence, AppSheet provides a practical and impactful solution for managing operational data in settings like ATec, where the focus is on functionality, accessibility, and eco-friendly practices. Ultimately, the suitability of AppSheet depends on the complexity and volume of the data management requirements. While it may not meet the demands of complex enterprise-wide systems, it remains a highly effective choice for straightforward data-centric applications, especially within the Google ecosystem.

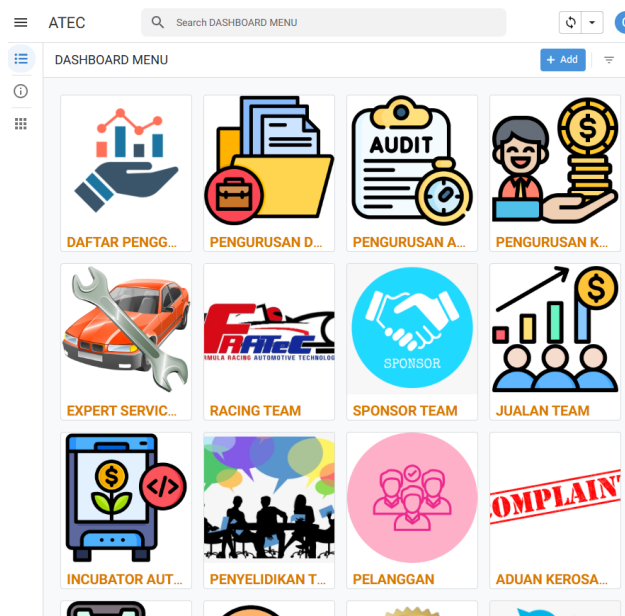


Figure 4: AteC Dashboard Menu interface by AppSheet

References

- AppSheet.(2022). Case study: Unilever supply chain digitization.
<https://solutions.appsheet.com>
- Database Trends. (2024). Scalability limits of no-code data hubs. Database Trends & Applications. Retrieved from <https://www.dbta.com>
- Forrester. (2023). The total economic impact™ of AppSheet.
<https://www.appsheet.com/whitepapers/tei>
- Kirchhof, M., Denner, M. S., & Uebernickel, F. (2023). Governance in citizen development: A framework for responsible low-code deployment. Proceedings of the IEEE International Conference on Software Engineering.
- Liu, G., Sharma, S. K., & Malik, A. (2024). Trends and challenges in low-code development: A systematic literature review. IEEE Access, 12, 33814–33833.
<https://doi.org/10.1109/ACCESS.2024.3381433>
- Ruscio, F. D., Di Penta, M., & Falessi, D. (2021). Low-code and no-code development: Risks, opportunities, and research needs. IEEE Software, 38(6), 80–87.
<https://doi.org/10.1109/MS.2021.3099495>