Volume 5, Issue 6

Malaysia Journal of Invention and Innovation

https://journal.academicapress.org/aps/index.php/mjii

Research Article

SmartBot Design for Centralised Telegram Community Moderation

Muhammad Hafiq Erfan Abdullah[1], Muhammad Afif Aiman Samizee[1], Hifzhan Irfan Zaihan[1] and Mohamad Rahimi Mohamad Rosman[1,*]

1 Faculty of Information Science, Universiti Teknologi MARA, Malaysia
* Correspondence: rahimimr@uitm.edu.my

Keywords:
Telegram bots
Content moderation
Software as a service
Systems analysis and design
Community management

Abstract:
Manual moderation of Telegram groups requires administrators to monitor messages and respond to violations across multiple communities. This study presents the design of SmartBot, a centralised web platform for Telegram bot management and moderation. Requirements were derived from administrative experience, interviews with group owners and document analysis. The design combines a web dashboard, a Telegram-side moderation engine and persistent records of administrative and moderation activity. Process models, an entity relationship diagram and interface prototypes describe the proposed workflows. The design establishes a basis for implementation and pilot evaluation of moderation accuracy, responsiveness and administrative workload.

Copyright: © 2026 by the authors. Submitted for open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. INTRODUCTION

Telegram group administrators must identify spam, scams and prohibited content while maintaining normal discussion. The paper illustrates the development and workflow documented for DIGIT360 Sdn. Bhd, one of the early university startup in UiTM. The application relies on administrators to notice a message, assess it against group rules and decide whether to delete it or act against the sender. Moderation coverage therefore depends on administrator availability, especially when responsibility extends across several groups.

SmartBot was designed as a web-based software-as-a-service platform for connecting Telegram bots, configuring group-specific moderation rules and reviewing moderation activity. The objective was to specify the requirements, data flows, data model and interfaces needed to support this workflow.

2. LITERATURE REVIEW

Telegram's Bot API provides the integration surface for receiving updates and performing moderation actions, subject to the bot's permissions in each group (Rahman et al., 2026; Telegram, n.d.).

This separates two design concerns. The bot handles message events and applies permitted actions within Telegram, while a web application can provide a common place to configure policies and review their effects across groups.

In the project analysis, rules for known keywords and URLs were paired with planned AI-assisted screening for suspicious messages (Saka, 2025; Bossi et al., 2026). Explicit rules give administrators direct control over recurring violations (Roy et al., 2025). Screening can flag messages outside those predefined patterns, while moderation records support review when a decision affects legitimate discussion (Singhal et al., 2023; Zhong et al., 2024).

3. METHODOLOGY

The team used an iterative systems analysis and design approach (Larman & Basili, 2003; Yusoff et al., 2026; Rosman et al., 2019; Rosman et al., 2021). Requirements were informed by the team's experience administering Telegram groups, interviews with group owners and document analysis of existing moderation bots and reported scam patterns in Malaysia. The team compared the existing manual workflow with the proposed automated workflow and consolidated its observations into functional and non-functional requirements.

The requirements were translated into a context diagram, a data flow diagram, a three-layer architecture, a hierarchical input–process–output representation, an entity relationship diagram and a data dictionary. Interface prototypes were prepared to show how administrators would configure bots and groups and review system activity.

4. FINDINGS

4.1 Requirements and Data Flows

The design specifies account management, bot registration, group configuration, automated moderation, activity records, analytics, notifications and subscription controls. It also identifies performance, usability, reliability, security and privacy requirements, including bot-token encryption, row-level access controls and redaction of secrets from logs.

Figure 1 defines the system boundary and exchanges with users and group administrators. Figure 2 expands the proposed data flows and shows how administrative inputs and moderation activity move through the system.

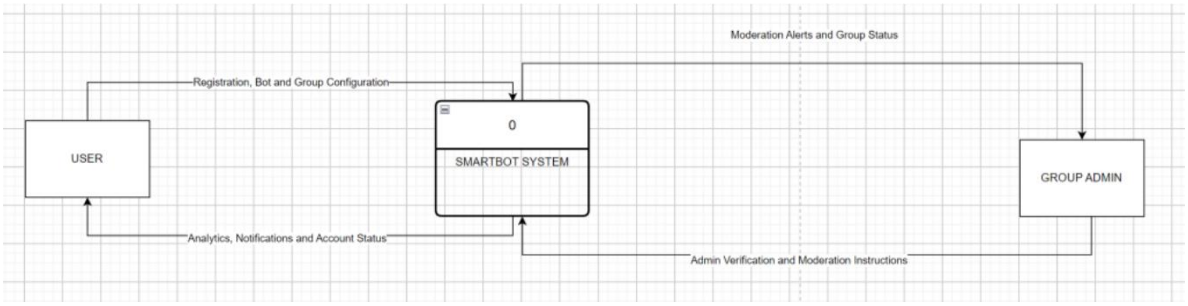


Figure 1. Context diagram for SmartBot

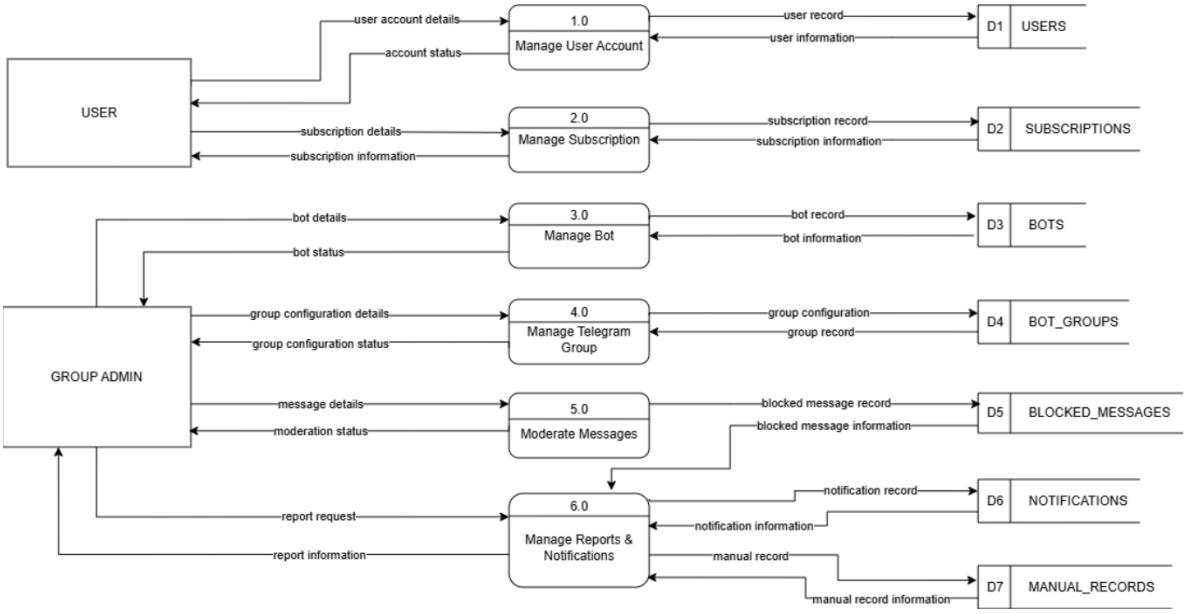


Figure 2. Data flow diagram for SmartBot

4.2 System and Data Architecture

The presentation layer provides the web dashboard. The application layer handles administrative workflows and interaction with the Telegram moderation bot. The data layer retains the records required for configuration, oversight and reporting. Figure 3 presents the entity relationship diagram, which links the information required to manage users, bots, groups and moderation activity.

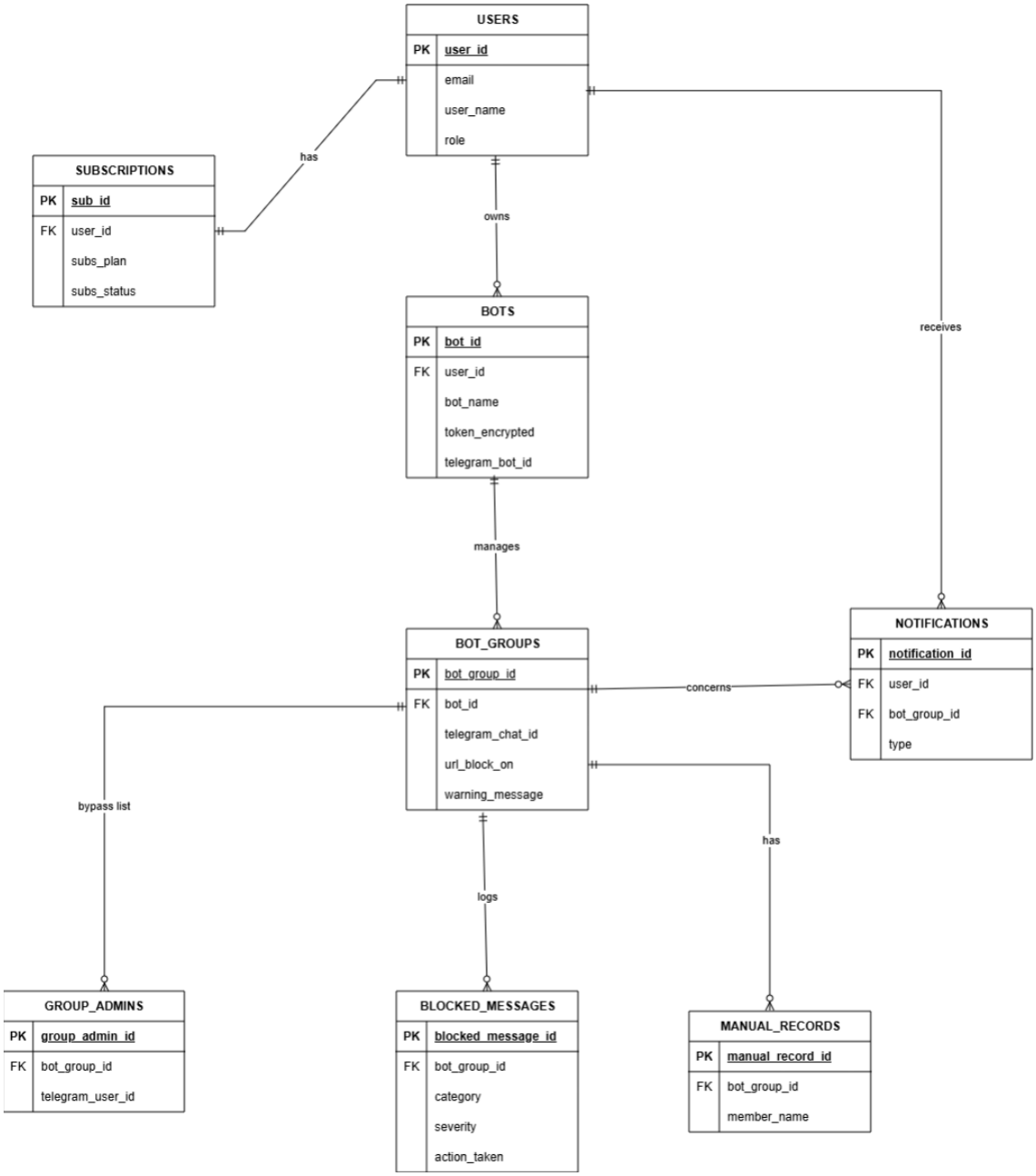


Figure 3. Entity relationship diagram for SmartBot

4.3 User Interface Prototypes

Figures 4–6 show the public landing page, authentication and the administrative overview. Figures 7–8 show how administrators manage bots and groups. Figures 9–11 cover incident records, activity analytics and notifications. Figures 12–13 show the billing and settings interfaces.

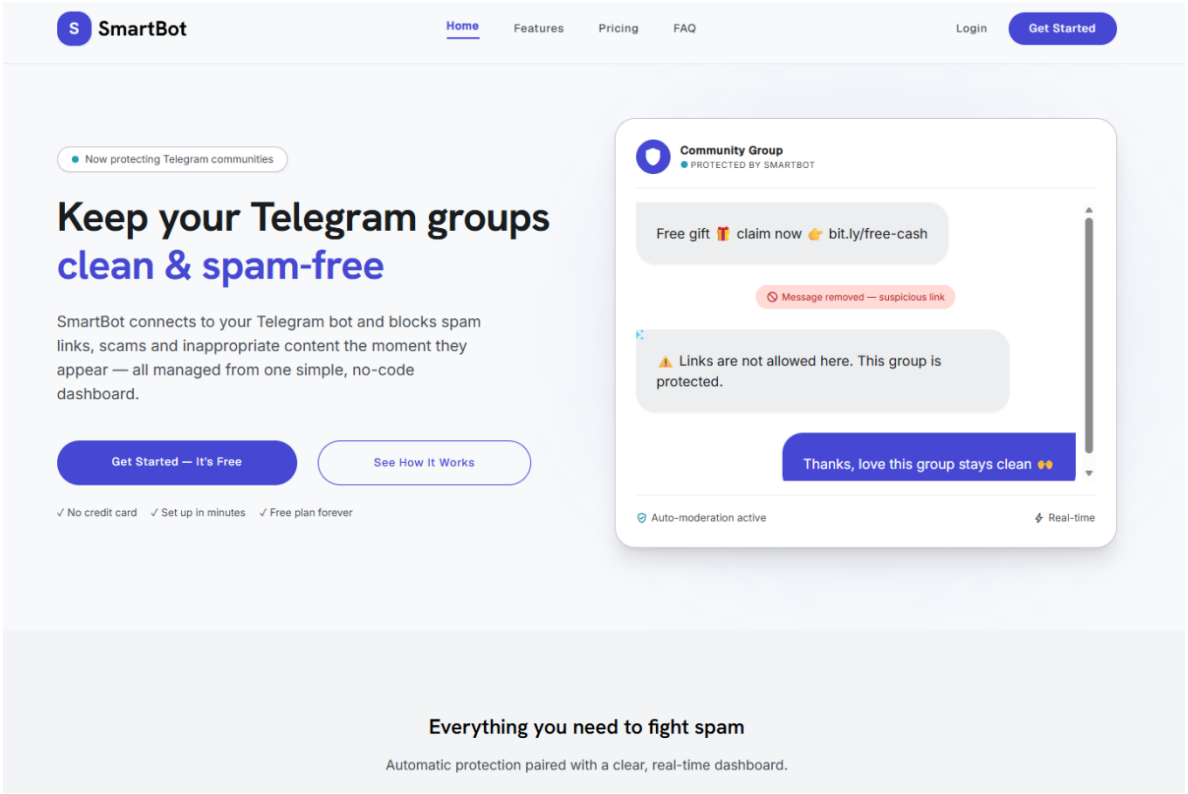


Figure 4. Public landing page



Figure 5. Login page

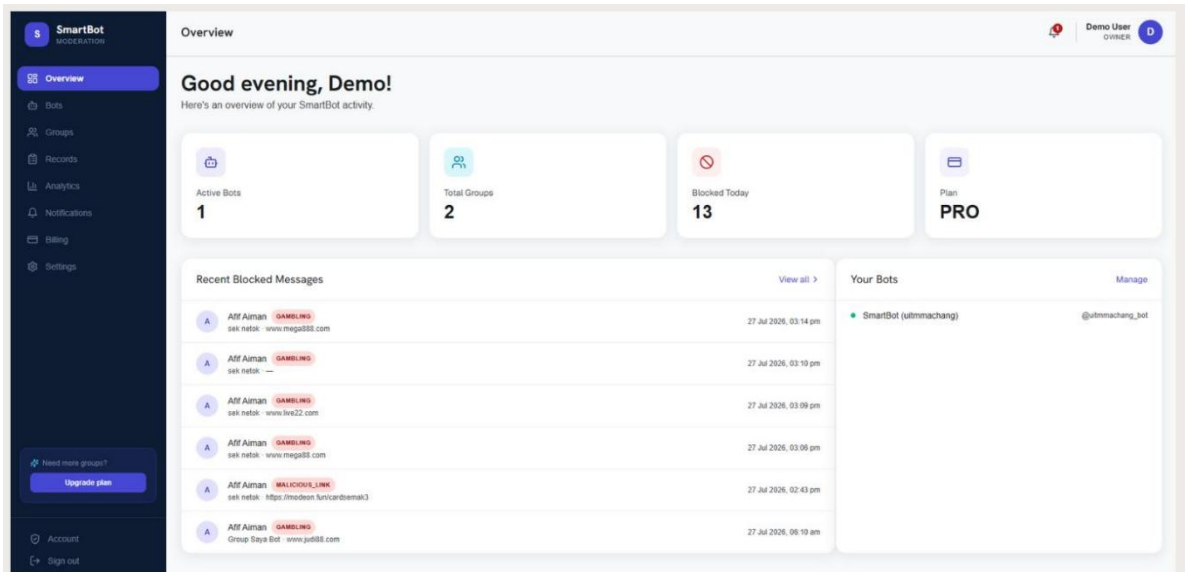


Figure 6. Administrative overview dashboard

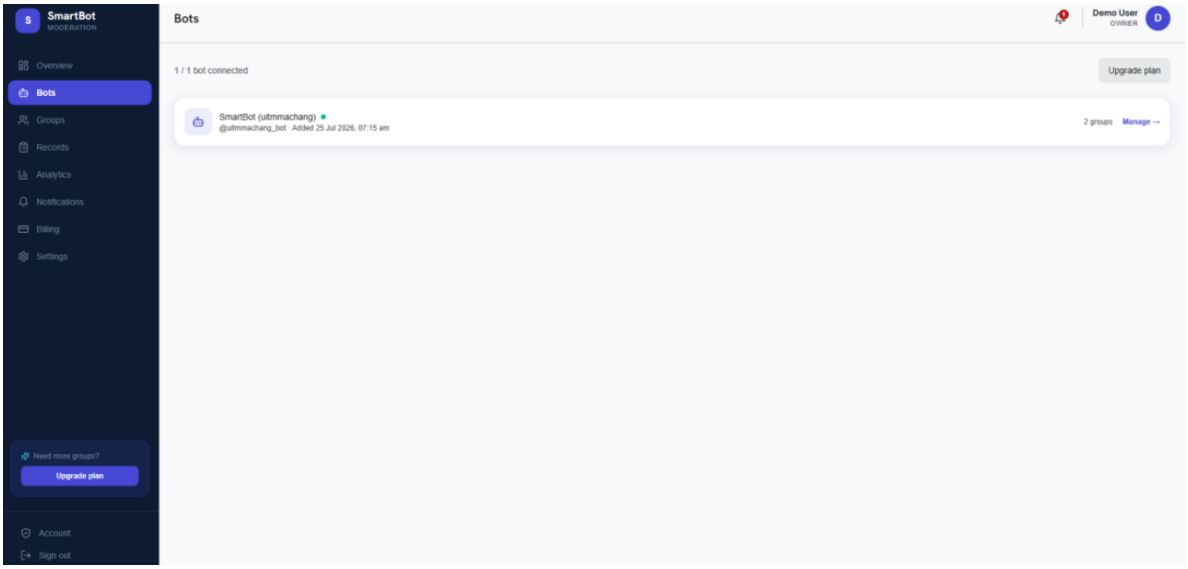


Figure 7. Bot management interface

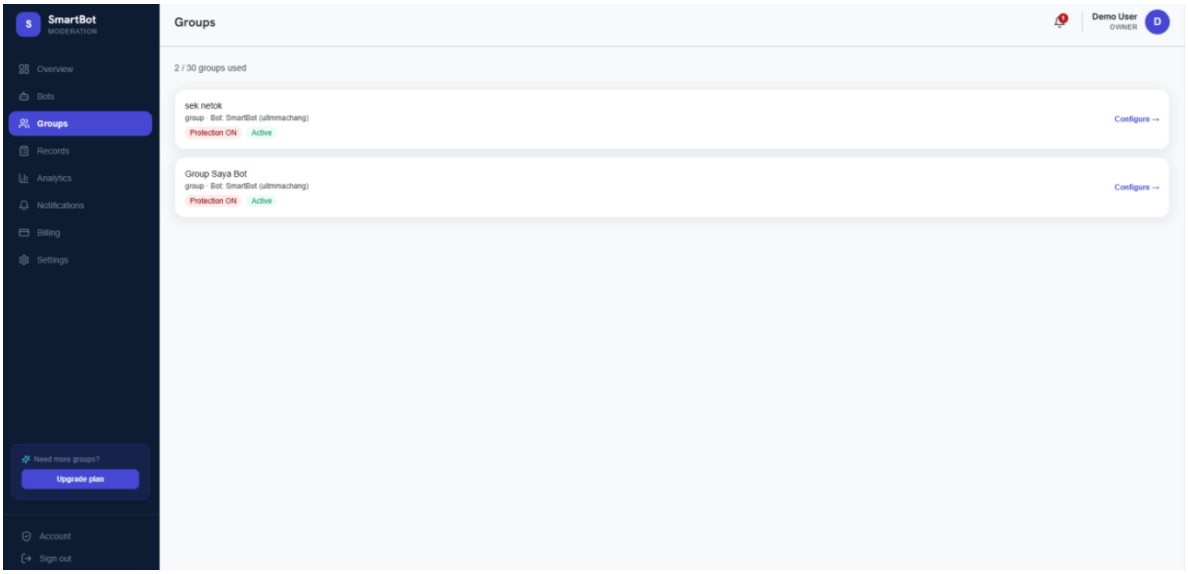


Figure 8. Group management interface

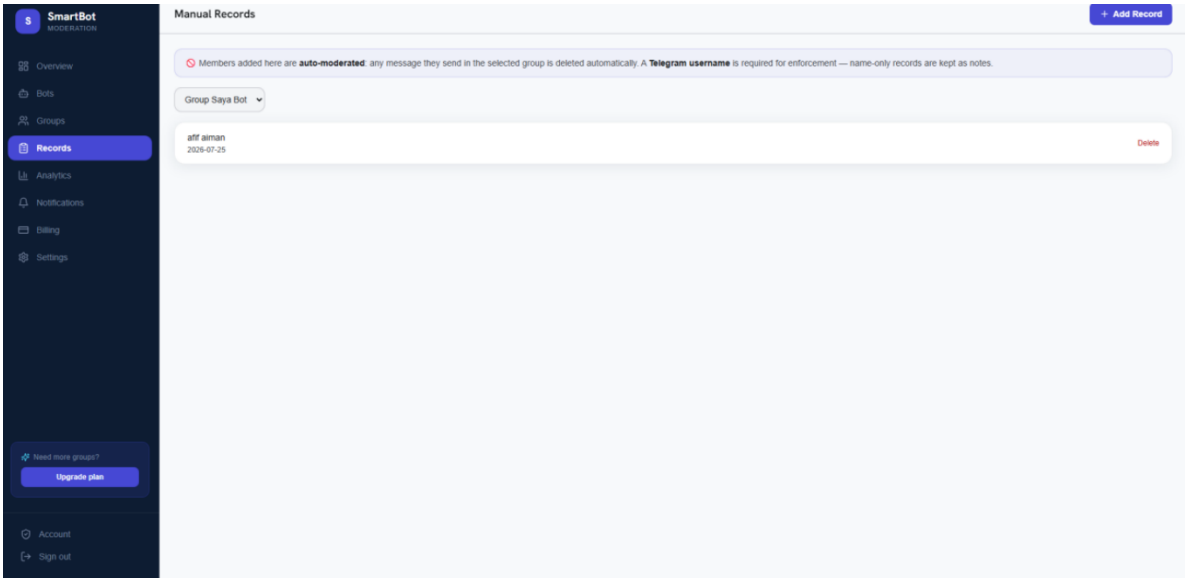


Figure 9. Moderation records interface

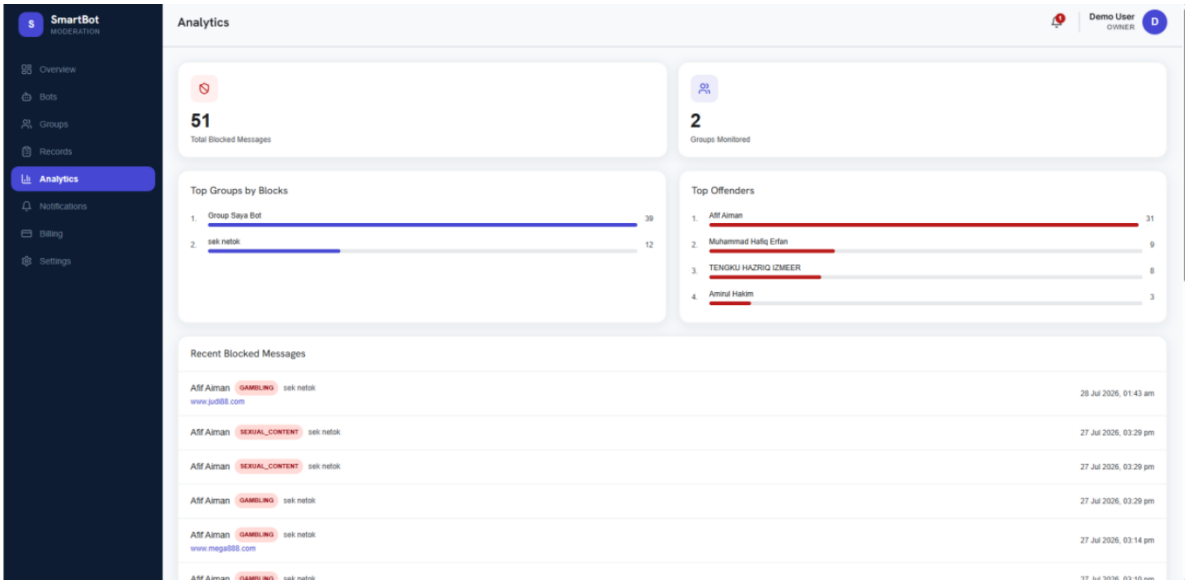


Figure 10. Moderation analytics interface

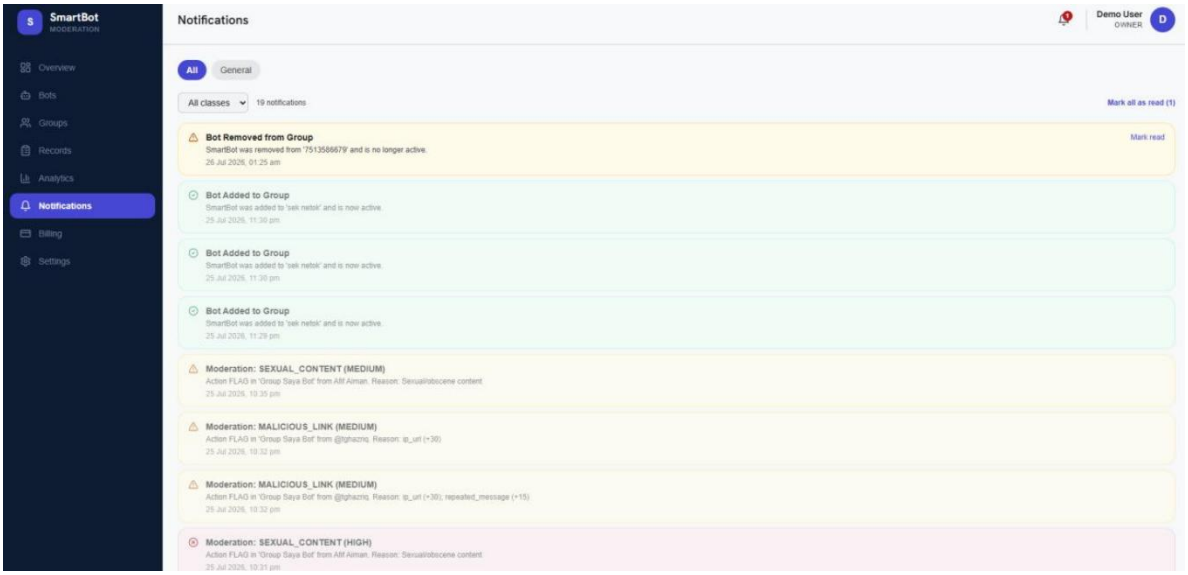


Figure 11. Notification interface

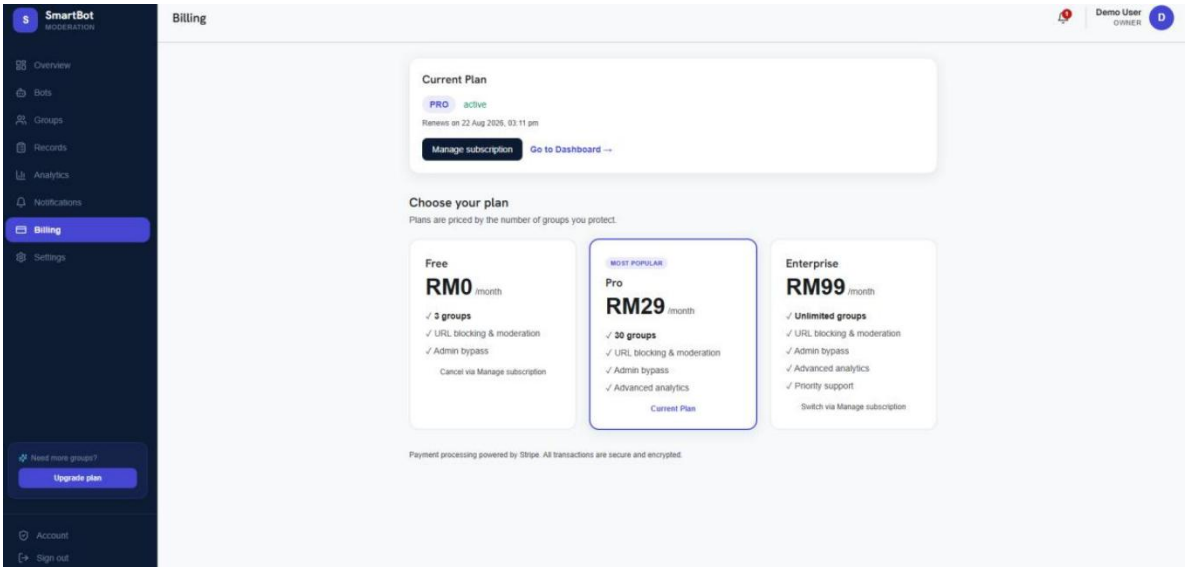


Figure 12. Subscription and billing interface

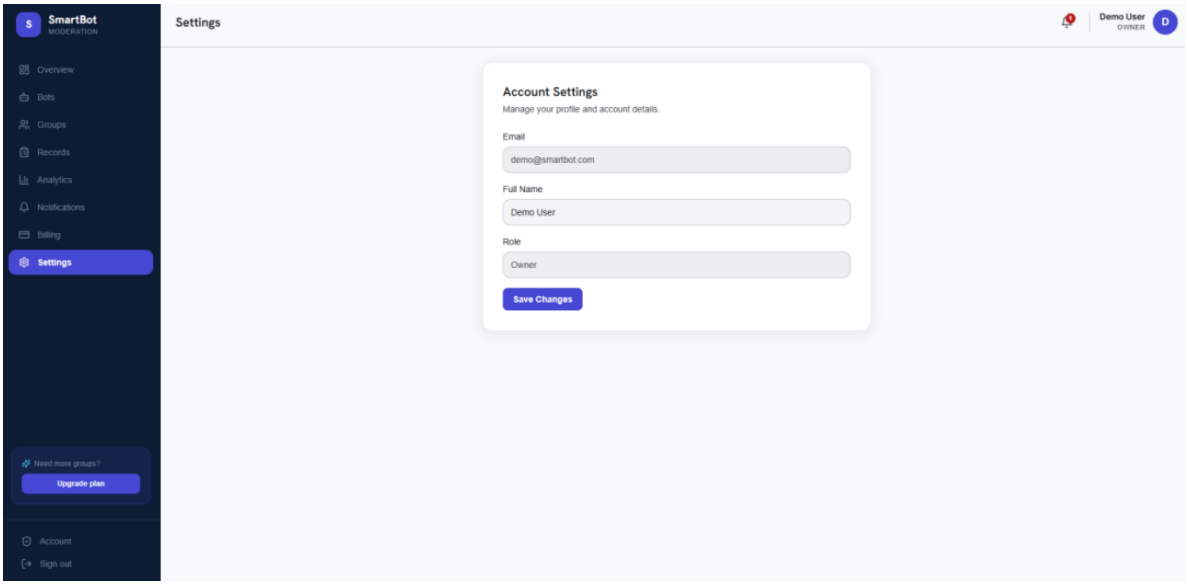


Figure 13. Settings interface

5. DISCUSSION

Centralising bot and group configuration gives administrators a consistent way to review policies and moderation records across groups. The process models identify where data enter the system and where administrative decisions are recorded. The interface prototypes make these interactions concrete and provide a basis for reviewing the design with group administrators.

Incorrect classifications can allow harmful messages through or remove legitimate contributions. The design also depends on appropriate Telegram permissions, secure token handling and reliable webhook delivery. A pilot deployment could measure the proportion of valid and erroneous moderation actions, time from message receipt to action, and administrator workload before and after adoption. These outcomes were outside the scope of the design study.

6. CONCLUSION

SmartBot specifies a centralised architecture for managing Telegram bots and group moderation. The requirements, process diagrams, data model and interface prototypes describe how administrators would configure policies, monitor activity and review moderation decisions. Together, these artefacts provide an implementation basis for evaluating the platform in a controlled pilot.

References

Bossi, L., Saccani, F., Panebianco, F., Maci, A., Zanero, S., Longari, S., & Carminati, M. (2026). Towards Automated Cyber Threat Intelligence Elicitation in Underground Forums. *arXiv preprint arXiv:2608.23185*.

Larman, C., & Basili, V. R. (2003). Iterative and incremental development: A brief history. Computer, 36(6), 47–56. https://doi.org/10.1109/MC.2003.1204375

Rahman, M., Afrika, L., Netri, Z. H., Dewi, R. A., Febriadi, R., & Triana, N. (2026). Comparative Analysis Of Whatsapp And Telegram As Digital Communication Platforms Based On Their Usage And Functions In The Digital Era. *Jurnal Warta Dharmawangsa*, *20*(1), 364-370.

Rosman, M. R. M., Abdullah, N., Arshad, I. H., Saleh, M. S. M., Zawawi, M. Z. M., & Fadzil, F. H. (2021). The Design and Development of the i-Jury Management System. *Advances in Business Research International Journal*, *7*(2), 111-122.

Rosman, M. R. M., Ismail, M. I. U., Dzarawi, M. H. A., & Azman, M. A. Z. M. (2019). Conceptualizing an Attendance Monitoring System for Malaysian Educational Institutions. *Journal of Advanced Research in Computing and Applications*, *16*(1), 10-23.

Roy, S. S., Vafa, E. P., Khanmohamaddi, K., & Nilizadeh, S. (2025). {DarkGram}: A {Large-Scale} analysis of cybercriminal activity channels on telegram. In *34th USENIX Security Symposium (USENIX Security 25)* (pp. 4839-4858).

Saka, T. (2025). *Collaborative human-AI approach to mitigate phishing attacks* (Doctoral dissertation, Phd thesis, The University of Edinburgh, Edinburgh, UK).

Singhal, M., Ling, C., Paudel, P., Thota, P., Kumarswamy, N., Stringhini, G., & Nilizadeh, S. (2023, July). SoK: Content moderation in social media, from guidelines to enforcement, and research to practice. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)* (pp. 868-895). IEEE.

Telegram. (n.d.). *Telegram Bot API.* https://core.telegram.org/bots/api

Yusoff, N. H., Rosman, M. R. M., Zain, M. S. M., & Rosli, N. N. I. N. (2026). Exploring The Design & Development of The Training Evaluation System. *International Journal of Accounting*, *11*(64), 198-206.

Zhong, W., Bailard, C., Broniatowski, D., & Tromble, R. (2024). Proud boys on telegram. *Journal of Quantitative Description: Digital Media*, *4*.