Human-Centered User Interface Design for Self Help Group Using a Flutter Application Framework

Abhiram Bulusu, Gudla Dinesh Reddy
Vuppala Naga Sushanth Kumar
Department of Computer Science and Engineering
School of Computing
Amrita Vishwa Vidyapeetham
Amritapuri, India
amenu4aie20103@am.students.amrita.edu,
amenu4aie20130@am.students.amrita.edu,
amenu4aie20171@am.students.amrita.edu

Sruthy Anand, Ramesh Guntha
Center for Wireless Networks Applications (WNA)
Amrita Vishwa Vidyapeetham
Amritapuri, India
sruthyanand@am.amrita.edu,
rameshg@am.amrita.edu

***Abstract*—Designing user interfaces (UI) for marginalized rural women presents unique challenges that require a good understanding of Human-Computer Interaction (HCI) principles to ensure usability. This research paper focuses on developing a contemporary UI for a Self-Help Groups (SHGs) in India, catering to the diverse age groups and educational backgrounds within these communities. The goal is to blend HCI principles into the design to develop an interface that is attractive, user-friendly, and easy to navigate for the users of these communities. This study tries to employ a user-centered design thinking approach to guide the development of the UI prototype. The study details the use of financial module of the application and its implementation. The software utilizes a multi-level client-server architecture using Flutter frameworks for UI design.**

***Index Terms*—Rural Development, Self-Help Groups, HCI, User Interface Design, Mobile Applications**

I. Introduction

Rural development in India involves improving the quality of life, educational access, and healthcare services and increasing the economic well-being of people living in rural areas. Development in rural areas is essential for any nation's comprehensive progress. Empowering women, who comprise a sizable share of the rural workforce, is vital to the community's well-being, social stability, and economic growth [1]. In remote areas, women's development is greatly aided by Self-Help Groups (SHGs) [2]. SHGs enable women to actively engage in income-generating activities and community initiatives by offering a forum for group decision-making, skill development, and financial inclusion. This improves the socioeconomic standing of individual women and has a beneficial knock-on effect on entire rural communities [2]. AmritaSREE, an SHG initiative by the Mata Amritanandamayi Math, is a fine example of supporting women for their empowerment [3], [4]. Researchers have proposed an participatory Internet of Things solution to ease the operations of the SHG [5] with a special focus on enhanced Technology Adoption Model.

However, developing User Interface (UI) design needs to be careful by addressing the needs of varying digital literacy groups. Hence, in this work, we detail how the Human-Computer Interaction (HCI) concepts may play a crucial role in the design and development of the software solution for the SHG. HCI factors are essential for the UI design as they regulate how users interact with technology, ensuring that systems are user-friendly, intuitive, and serve the user's specific needs. When creating applications and user interfaces specifically for rural populations with different digital literacy, HCI principles play an essential part. Because HCI principles guide the design, it focuses on creating user-friendly screens that are relevant to the target users' specific needs and preferences and are user-friendly to ensure usability. A more inclusive and user-centric interface has easy navigation, less cognitive load, and visually acceptable content [6]. Generally, the HCI method emphasizes user-friendliness and simplicity, making the users use the software without much cognitive effort to navigate multiple screens.

Jakob Nielsen's heuristics are a set of general principles used to design user-friendly interfaces [7], [8]. These guidelines are based on real-world research and best practices in the field of HCI. They help designers create interfaces that are easy to use, efficient, and intuitive. By following the principles, designers can improve the usability of their products, leading to better user experiences. These principles aim to ensure users can easily understand the system, recover from mistakes, navigate smoothly, focus on important information, get help when needed, minimize memory strain, and have control over their interactions.

These SHGs transactions such as monthly payments, loan status, etc are tracked in physical ledgers like books and passbooks. It can lead to more administrative overhead and minor mistakes during manual ledger maintenance, which can take time to identify. The software has to keep track of all the transactions happening within and across clusters of SHGs because most of these SHGs work in clusters. This work also details the finance module in the software application, which monitors the member's financial activity and ensures that the finances are accurate. Users can have their own digital

account and transaction details visible in their account, making it easy to monitor their transaction details at any time. These modules are designed by using HCI methods. We use the Flutter framework to develop the software, which is cross-platform and has an appealing developer experience for front-end development because of its capability to craft custom widgets to fulfil specific UI requirements. Flutter's widget framework facilitates the creation of expressive and adaptable user interfaces.

The remainder of this paper is structured as follows. Section II discusses the related works in this research. Section III introduces the software application and its high level functionalities. Section IV discusses the effective human factors for design of the UI. Section V details the design strategy used for the software through different heuristics. Concluding in section VI.

II. Related Works

The research [9] attempts to promote micro-economic activities within the community and make information technology services more accessible to the people. This study aimed to develop a user interface that is both intuitive and culturally sensitive to its users' preferences through qualitative and quantitative data analysis. The creation of mobile applications has grown significantly, and it is now crucial for these apps to attract the attention of their target audience via their User Interface (UI) and User Experience (UX) [10]. This study [11] explores the multifaceted aspects of design, UX, and usability crucial for developing interfaces for rural people especially with less digital literacy. The researchers of the study [12] gathered preferences from a diverse range of concerned individuals. Mobile app developers can use these preferences as a guide, providing valuable insights that will help the mobile app designers and developers to focus on specific UI/UX challenges encountered in such applications.

The authors of this research [13] focused on the emotional aspects of UX design for software applications. Specifically, they embraced the concept of existential feelings, or feelings of being (FoB), within the framework of mobile UX design. They introduced a FoB model for designing mobile applications, drawing from findings of previous studies and literature analysis. The model comprised 13 FoB elements. Empathy in UI design is a fundamental principle that involves understanding users' emotions, goals, and context to design interfaces that resonate personally, ultimately leading to functional UI that deeply connects with users [14]. This study [15] suggests an improved architecture rooted in HCI design principles for smart mHealthcare. Moreover, HCI supports User-Centric Design (UCD) for intelligent mHealthcare models. Furthermore, the 5-layered architecture of the Internet of Things (IoT) combined with HCI holds promise for enhancing User Experience (UX) in mHealthcare design, potentially contributing to saving lives. HCI design enhances user experience and satisfaction by prioritizing ease of use, safety, alerts, and error-resistant end-user displays. This study [16] evaluates the usability of diverse mobile applications across various categories to gain deeper insights into mobile app usability. It accomplishes this by devising a hierarchical approach that depicts the mobile app usability in terms of usability principles and features. In addition to practical implications informing mobile app design, the study uncovers the dependence between usability principles, attributes, and features.

The work [17] explores UI requirements and challenges from various angles, evaluating the UI designs of ICT initiatives aimed at rural developing areas. It compiles and examines three components of UI design recommendations for smaller display devices in rural developing countries: information input, user control, and output presentation. This study focuses solely on the initial phase of the user-centered design approach aimed at creating an interface accessible to illiterate users.

This article aims [18] to enrich user experience by consolidating all features of a travel app—restaurants, lodging options, and must-visit locations—into a single application. The current study focuses on the creation of a comprehensive mobile application for tourism in Timişoara, Romania, employing contemporary technologies such as Firebase and Flutter. To develop a comprehensive stack mobile application for the Google and Apple markets and devices in 2023, this thesis [19] delves into software development from that perspective.

This paper [20] attempts to create a digital museum guide app with a usability focus, employing the Firebase web platform and Flutter framework. Developers often choose cross-platform frameworks for software development due to their convenience, enabling them to write code once and deploy it simultaneously on both iOS and Android platforms. This approach streamlines development efforts by eliminating the need to acquire separate skill sets for each platform [21]. Flutter's widget-based UI framework offers flexibility and adaptability, allowing for the creation of straightforward user interfaces. With a vast library of widgets and a vibrant community, Flutter expedites the development process [22]. Powered by the Dart programming language, Flutter ensures rapid and seamless graphics rendering across diverse platforms [23], [24]. By only updating widgets when changes occur, Flutter's approach to building user interfaces enhances rendering speed [25].

III. Introduction to SHG Application

This section details the client-server technology used for the development, leveraging flutter technology for intuitive design, and the requirement of finance module for the application.

A. Client-Server Architecture

The design of the architecture for the application draws inspiration from the work [26]. It consists of a client-side system incorporating the user interface and application logic as shown in Fig. 1. The client side is developed using Flutter framework and the Server side is developed using Node.js. Within this system, the Screen oversees various Views and Services with Controllers to handle data, while also facilitating API calls to an external Server. On the server-side, a Server

component manages client requests via defined Routes, directing them to specific functions and interfacing with a Database to oversee data management. Helpers offer utility functions, and Models represent the structure of data utilized within the application. The Database serves as the repository for the application's data, responding to queries from the Server. This architectural framework prioritizes scalability, security, maintainability, and streamlined data flow within the software system. This architecture allows developers to to work on distinct aspects of the client-side independently. This approach effectively separates various UI logic while maintaining their interconnections, promoting maintainability and re-usability.

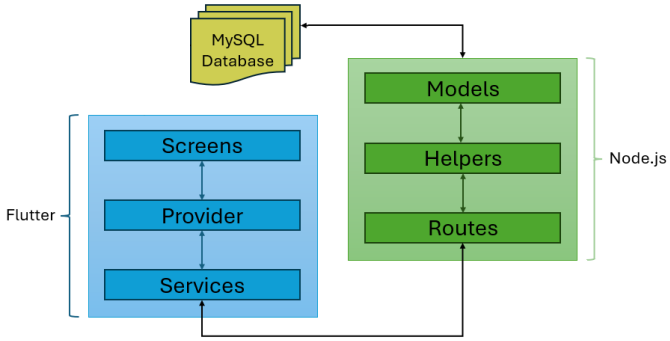


Fig. 1: Client-Server Architecture of the application

B. Flutter SDK Kit

Developers opt for cross-platform application frameworks due to their convenience, with Flutter being a prominent choice. Flutter enables the creation of applications for various platforms, including iOS, Android, web, desktop, and watch, using a unified codebase. This approach significantly reduces development time and effort compared to building separate codebases for each platform. Dart Flutter plays a crucial role in efficiently managing within an application, prioritizing user-centric design and functionality. Moreover, Flutter empowers developers to craft either low-level or high-level custom widgets to fulfill specific UI enhancement requirements. Through its widget framework, Flutter facilitates the creation of expressive and adaptable user interfaces, leveraging its declarative and composable nature. Widgets serve as fundamental building blocks for screens, offering extensive customization options, from basic buttons and text fields to complex layouts and animations. These widgets are organized in a tree structure, defining the visual hierarchy of the application and enabling reusability for the creation of rich, interactive, and responsive user interfaces efficiently. Additionally, Flutter ensures a cohesive color scheme for UI styling to enhance visual appeal and user-friendliness. Its widespread adoption has fostered a vibrant community that provides developers with several resources, plugins, and third-party packages for support and enhancement. The following section detail the implementation of the finance module.

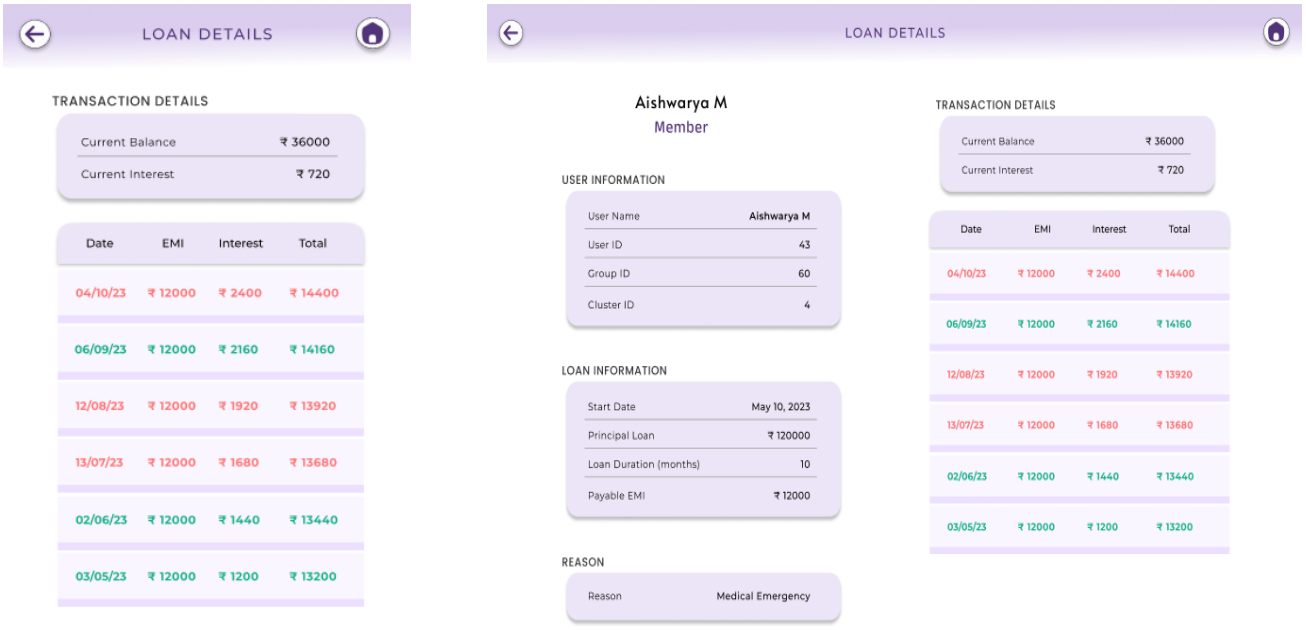
C. Finance Module of the Application

The finance module serves as the key of the application. It represents the core functionality where users are likely to seek the most assistance. It is important that we design this interface adhering to the principles of user-centered design. The objective is to provide an intuitive and user-friendly experience that empowers users to complete financial processes with ease.

1) Users roles and SHG Functioning: The organization follows a hierarchical cluster-based structure for efficient SHG management. 10-20 Members come together to form a SHG. Multiple SHGs collectively form a cluster, with each cluster generally containing 100 to 200 SHGs. Cluster administrators will manage the administrative functions of a cluster. For instance, one of the cluster has four cluster administrators, each responsible for overseeing 50 groups within the cluster. Each SHG has a designated group administrator, treasurer. The group treasurer is responsible for collecting monthly contributions from members and pooling these funds at the cluster level. The accumulated amount from various groups is then utilized to provide loans to those in need within the cluster. The cluster administrators and group admins play a crucial role in monitoring user loan repayments and interest status. The conventional practice of maintaining payment records is done through physical ledgers.

2) Web Interface for Cluster Admins: With more number of groups and users, the volume of data, such as active loans, also grows. Introducing a web interface would significantly ease the workload for cluster administrators. Since the software is developed using Flutter, a cross-platform application framework, extending its capabilities to include a web interface is feasible. Leveraging this web interface would enable cluster admins to efficiently manage groups and monitor loans. The primary advantage of the web interface lies in its ability to display more data. The difference between mobile interface and web interface is shown in Fig. 2.

3) Loan Request Module: Traditionally, when a member requires a loan, they approach the cluster admin along with the group admin during the monthly meeting. At this meeting, the member submits a loan request detailing the principal amount, duration in months, and the reason for the loan. However, this manual process can become hectic for the cluster admin, who has to process all loan requests during each meeting while checking the eligibility criteria manually. With the introduction of this module, the entire process can be streamlined and made more efficient. Now, members simply need to approach their group admin to raise a loan request. The group admin then submits the request through the system. The cluster admin can access all the loan requests for the month through the web interface. The interface provides features such as sorting requests by group, name, and time, allowing the cluster admin to easily manage and review all the requests. Through this module, the cluster admin can quickly assess each loan request, viewing details such as the reason for the loan, and making decisions accordingly.



(a) Mobile Interface

(b) Web Interface

Fig. 2: Interface for Loan Tracking Module

4) Loan Module: Once the loan request is processed by the cluster admin, a new loan will be created in the name of the member and the SHG. The loan module will handle all financial calculations through backend coding to execute loan operations seamlessly. The cluster admin can then track these loans via the web interface, which provides comprehensive details of all loan transactions, including interest and monthly repayment. This streamlined process not only eliminates the need for manual paperwork but also ensures a more efficient loan management system within the organization. The loan module of the system cuts the balance amount to be paid after each monthly repayment. The subsequent loan will be closed once the loan amount and interest have been paid back to the cluster. The UI for this module follows the design principles such as minimalistic and aesthetic design, recognition over recall, consistency and standards to make it simple for the users to understand.

$$L_{\text{remaining}} = L_{\text{remaining}} - L_{\text{repayment}_n} \tag{1}$$

$$R_{\text{month}_n} = L_{\text{repayment}_n} + I(L_{\text{remaining}}) \tag{2}$$

In the equations presented, the symbol $L_{\text{remaining}}$ denotes the remaining loan amount after n months. It represents the balance that remains to be repaid as time progresses. The symbol $L_{\text{repayment}_n}$ signifies the repayment amount due for the nth month. This amount is typically assumed to be a fixed percentage of the remaining loan balance. Furthermore, R_{month_n} denotes the total payment required for the nth month, inclusive of both the repayment amount and the accrued interest. Lastly, the term $I(L_{\text{remaining}})$ denotes the interest calculated which is 2% of $L_{\text{remaining}}$. This interest component contributes to the total payment due in each month, reflecting the cost of borrowing over time.

IV. Human Factors in Designing a User Interface

This section we discuss the effect of human factors on the intuitive UI design of the application. Human factors such as familiarity, efficiency, and error management play a crucial role in designing effective interfaces because human factors refer to how users interact with systems. Understanding the expectations and aspirations of the target users is essential for creating interfaces that meet their current needs and align with their future goals. The design aims to reduce user errors, increase comfort, and improve interactions with technology. Recognizing human abilities such as adaptability, recall, effectiveness, and accessibility are crucial elements for designing applications effectively.

It is important to consider human factors such as memory limitations and error proneness. Human memory has limitations in capacity and retention since most people can only recall seven things at a time [27]. Therefore, letting the user recognize a particular function rather than remember it becomes essential. Also, users have varying levels of technological knowledge. In our context, where we have women who may not be highly experienced with technology, it is especially crucial to consider different preferences for interaction styles and thinking abilities. When designing any aspect of the screen or UI, it is important to remember the specific needs and challenges that women who are less familiar with technology may face. It includes simplifying language, providing clear instructions, offering intuitive navigation, and ensuring a user-

friendly design accommodating varying levels of tech knowledge. By addressing these concerns, we can design an inclusive and user-centered design. The inclusion of this human factor can be addressed by following the ”Recognition of Recall” heuristic which is explained in further sections [28].

Efficiency and effectiveness in design are essential for creating user interfaces that prioritize seamless interactions and user satisfaction. By focusing on making interactions efficient, we, designers, aim to minimize user effort and time required to accomplish tasks. It involves streamlining workflows, simplifying processes, and reducing unnecessary steps to enhance user productivity. Additionally, highlighting effectiveness ensures that users can achieve their goals accurately and successfully within the interface. Efficiency and effectiveness become crucial in SHG digitization because minimizing user effort, minimizing time taken to perform any SHG process and achieving user goals, i.e. proper functioning of SHG is of utmost importance while designing a UI [29]. These principles can be effectively implemented through the ”flexibility and efficiency” heuristic, which will be discussed on in the following section.

Accessibility in design involves considering the needs of users with diverse abilities to ensure that an interface is usable by users with varying capabilities. This aspect focuses on creating inclusive interfaces that everyone can access and utilise, regardless of their physical, cognitive, or sensory abilities. Accessibility in design can be incorporated by various methods based on the target users and requirements. A few include images in place of text, use of icons, colour contrast, readability, etc. By prioritising accessibility, designers aim to provide equal access to information and functionality, promoting a more inclusive and user-friendly experience for all users [30]. Inclusive user interfaces naturally incorporate accessibility when designers prioritize user control and freedom during the UI design process.

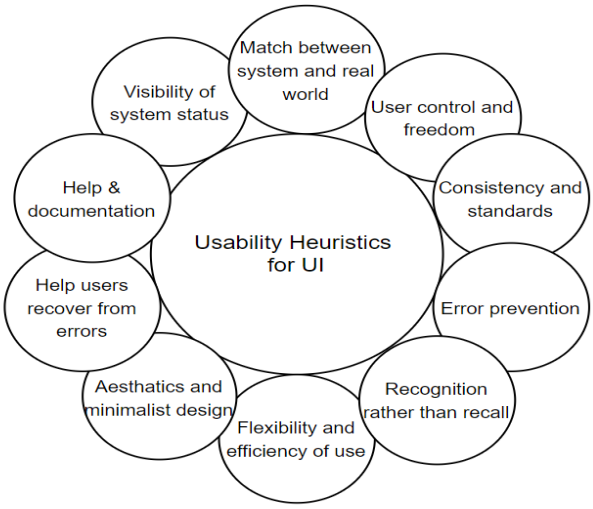


Fig. 3: Nielsen’s Heuristics [7]

V. User Interface Design Principles

Nielsen’s heuristics in the applications are essential for guiding designers in creating interfaces that effectively meet user needs [7], [8]. This sections details the different Nielsen’s heuristics used and corresponding design elements to adhere the principles. Some of these heuristics are utilized to design our UI which is detailed in this section. Some key aspects emphasized by Nielsen’s heuristics include showing visibility of system state, error prevention, flexibility and efficiency of use, aesthetic and minimalist design, help and documentation, recognition over recall, and user control and freedom as shown in Fig. 3.

1) Match between reality and system: In designing the application’s interface, we prioritize aligning the system design which reciprocates the current functioning of the SHG’s. This principle ”match between reality and system” ensures that application is presented in a way that users find familiarity while using.

2) User Control and Freedom: The feature that allows the users set the application’s view according to their own preferences such as theme and font size matches the ”User Control and Freedom” heuristic as shown in Fig. 4.

Fig. 4: Option to switch between light theme and dark theme

3) Consistency and Standards: The application is maintained with a consistent approach of using uniform color palettes, themes, font sizes, and styles across all screens. This adheres to ”Consistency and Standards” heuristic that ensures users have a inclusive visual experience.

4) Error Prevention: The application prioritizes ”error prevention” by implementing effective error messages and eliminating unhanded error conditions. Through the use of dialog boxes, error prompts, and text validations, we guide users to correct any mistakes or incomplete information, ensuring a smooth user experience as shown in Fig. 5. By combining these strategies of eliminating error-prone conditions and

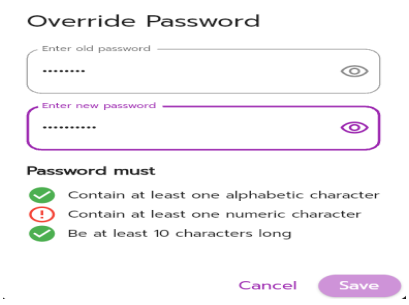
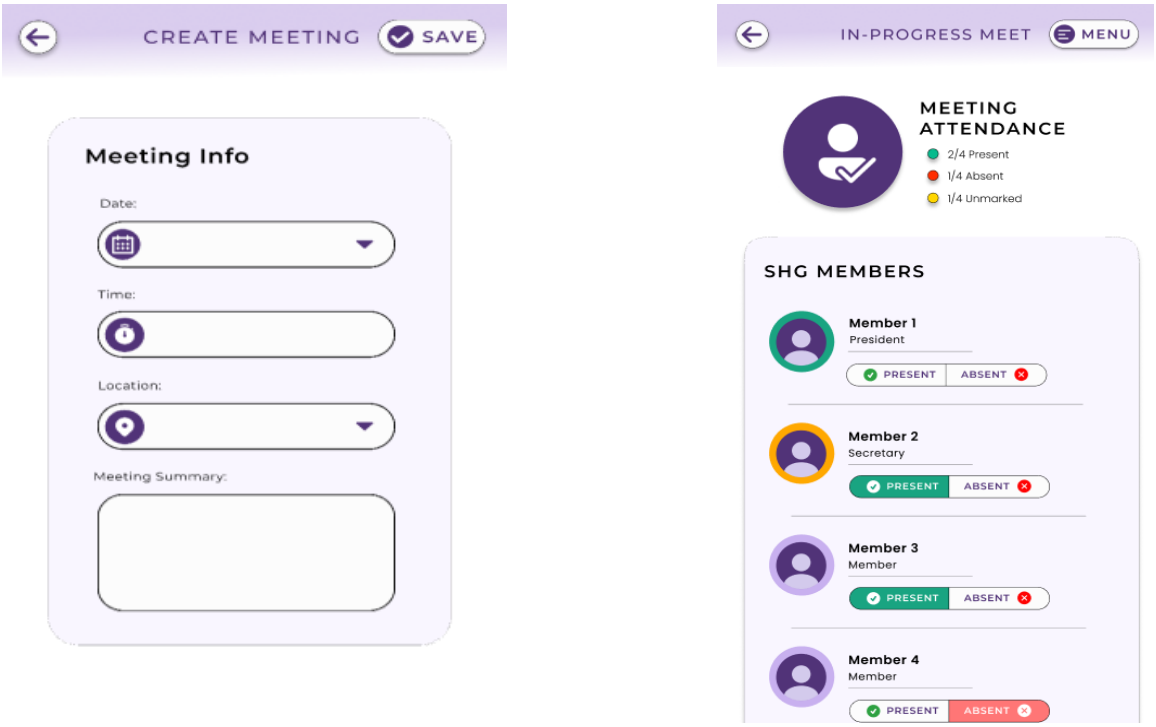


Fig. 5: Text Field Validator to check Passwords



(a) Represents Aesthetics and minimalistic design, consistency and standards, recall over recognition

(b) Represents visibility of system status, Flexibility and efficiency of use,

Fig. 6: Schedule Meeting and Attendance Interface which represents different principles

providing informative error messages, we have successfully integrated the best practice of error prevention into our user interface design. Additionally, our user authentication system enables specific functionalities based on user roles.

5) Recognition over Recall: Incorporating "Recognition over Recall" in the UI design involves prioritizing the visibility of information to aid users in making decisions without relying on memory. This principle emphasizes ensuring that all essential information, functions, and options are visible and easily accessible on the interface as shown in Fig. 6a. To implement this, we focused on using descriptive icons and labels for buttons, as well as a few colour schemes to represent the status of meetings(schedules, completed, ongoing) instead of relying solely on text. Users can quickly recognize functions. This approach enhances user navigation, making the interface more intuitive and efficient.

6) Flexibility and Efficiency: In designing the UI for the mobile application, we integrated the principle of "Flexibility and Efficiency" of use by considering various human factors such as learnability and understanding the educational and technological knowledge of the users. By engaging with individuals working on the ground and creating a persona representing the specific user group, we design to cater to inexperienced and experienced users, enhancing the overall usability of the interface.

7) Aesthetic and Minimalist: In designing the UI for this application, we integrated the principle of "Aesthetic and Minimalist" design by focusing on essential information and visual elements that support the users' primary goals. By following the heuristic that interfaces should not contain irrelevant or rarely needed information, we ensured that every UI component served a purpose and enhanced user experience. To achieve this, we limited the colour palette, with specific colours indicating successful or failed processes, providing visual cues that are meaningful to users as shown Fig. 6. The screen layout was designed uncluttered, avoiding an overload of elements and ensuring that only essential functions were visible at any given time. The application has a minimalistic design that prioritizes functionality without overwhelming the user by a simple and streamlined UI for the SHG processes.

8) Visibility of System Status: Incorporating the "Visibility of System Status" principle in the software interface design involves showing clear messages or prompts at essential activities such as user registration, password changes, and admin role assignments to let users know if these actions were successful. These prompts act as friendly notifications confirming that they have completed a task correctly, helping them feel confident and informed about what's happening in the system. By using these simple prompts, users can easily understand the status of their actions and navigate the interface with ease, making their experience smoother and more user-friendly as shown in Fig. 6b.

VI. Conclusion

Digital Empowerment of SHGs through an user-friendly software application will make a crucial pavement towards ru-

ral development. Previous research indicates that digital tools, with efficient and intuitive user interface (UI) design, enhance usability. In this work, we detailed into various aspects and key principles employed in designing the UI for the software. We designed user interfaces for SHG management after a deep understanding of human factors, and various design principles for a user centered, inclusive and efficient application for the users. We aim to optimize the user experience by implementing intuitive navigation, clear communication of information, and easy interaction to facilitate effective engagement with the software. The necessity of the web interface has been detailed which can be built using flutter's cross-platform capabilities. We explain the features and functionalities of the financial module of the software and its importance, which acts as the core feature. The finance module helps to streamline the process by eliminating the need for manual paperwork and also ensures a more efficient loan management system within the organization. Future study of this study can incorporate various design principles and responsive UI design for the software.

ACKNOWLEDGMENT

We would like to thank our beloved Shri. (Dr) Mata Amritanandamayi Devi, our institution's Chancellor for her support and guidance.

REFERENCES

[1] C. Coley, S. Sheshadri, and B. Rao, "Training india's first female toilet builders: An argument for improving sanitation through women empowerment and social inclusion," 09 2015.

[2] P. Sharma and S. Varma, "Women empowerment through entrepreneurial activities of self help groups," *Indian Research Journal of Extension Education*, vol. 8, pp. 46–51, 01 2008.

[3] S. Anand, M. Karthikeya, A. A. Sai, and O. Balamurali, "Multilingual hybrid chatbot for empowering rural women self-help groups in india," in *2023 International Conference for Advancement in Technology (ICONAT)*. IEEE, 2023, pp. 1–6.

[4] V. Suresh, R. Fishman, J. S. von Lieres, and B. R. Rao, "Impact of the covid-19 lockdown on the economic situation and food security of rural households in india," *Journal of Agribusiness in Developing and Emerging Economies*, vol. 12, no. 3, pp. 491–509, 2022.

[5] S. S. Sreeraj, A. Unnikrishnan, K. Vishnu, N. E. Kennith, S. Anand, and M. V. Ramesh, "Empowerment of women self help groups: Human centered design of a participatory iot solution," in *2020 IEEE Global Humanitarian Technology Conference (GHTC)*, 2020, pp. 1–8.

[6] Y. Liu, "Human-computer interface design based on design psychology," in *2020 International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI)*, 2020, pp. 5–9.

[7] M. Mirkowicz and G. Grodner, "Jakob nielsen's heuristics in selected elements of interface design of selected blogs," *Social Communication*, vol. 4, no. 2, pp. 30–51, 2018.

[8] J. Nielsen, "Enhancing the explanatory power of usability heuristics," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 1994, pp. 152–158.

[9] E. A. Emmanuel and M. H. N., "A user interface for micro-entrepreneurs in a rural community," *THE ELECTRONIC JOURNAL OF INFORMATION SYSTEMS IN DEVELOPING COUNTRIES*, vol. 43, no. 1, pp. 1–19, 2010. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1681-4835.2010.tb00306.x

[10] Z. Huang and Z. Tian, "Analysis and design for mobile applications: A user experience approach," in *Design, User Experience, and Usability: Theory and Practice*, A. Marcus and W. Wang, Eds. Cham: Springer International Publishing, 2018, pp. 91–100.

[11] J. A. Sheikh, H. S. Dar, and F. J. Sheikh, "Usability guidelines for designing knowledge base in rural areas," in *Design, User Experience, and Usability. User Experience Design for Everyday Life Applications and Services*, A. Marcus, Ed. Cham: Springer International Publishing, 2014, pp. 462–469.

[12] M. Sandesara, U. Bodkhe, S. Tanwar, M. D. Alshehri, R. Sharma, B.-C. Neagu, G. Grigoras, and M. S. Raboaca, "Design and experience of mobile applications: A pilot survey," *Mathematics*, vol. 10, no. 14, 2022. [Online]. Available: https://www.mdpi.com/2227-7390/10/14/2380

[13] M. N. Amir Dirin and T. H. Laine, "Feelings of being for mobile user experience design," *International Journal of Human–Computer Interaction*, vol. 39, no. 20, pp. 4059–4079, 2023. [Online]. Available: https://doi.org/10.1080/10447318.2022.2108964

[14] P. Wright and J. McCarthy, "Empathy and experience in hci," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 637–646. [Online]. Available: https://doi.org/10.1145/1357054.1357156

[15] A. A. Mesfer Alrizq, Shauban Ali Solangi, "An architecture supporting intelligent mobile healthcare using human-computer interaction hci principles," *Computer Systems Science and Engineering*, vol. 40, no. 2, pp. 557–569, 2022. [Online]. Available: http://www.techscience.com/csse/v40n2/44476

[16] Z. Huang and M. Benyoucef, "An empirical study of mobile application usability: A unified hierarchical approach," *International Journal of Human–Computer Interaction*, vol. 39, no. 13, pp. 2624–2643, 2023. [Online]. Available: https://doi.org/10.1080/10447318.2022.2082021

[17] C.-L. Chang, "User interface design of small-display devices for the rural developing world," Master's thesis, 2008.

[18] M.-D. Pop and A.-R. Stoia, "Improving the tourists experiences: Application of firebase and flutter technologies in mobile applications development process," in *2021 International Conference Engineering Technologies and Computer Science (EnT)*, 2021, pp. 146–151.

[19] S. Liao-Mäkinen, "Developing a full stack mobile application," 2023.

[20] M. Sävmarker, "How to design a digital guide app in flutter/firebase with a focus on the group experience and text-to-speech," 2023.

[21] "Top 10 Best Cross-Platform App Development Frameworks in 2024 — TechAhead — techaheadcorp.com," https://www.techaheadcorp.com/blog/best-cross-platform-app-development-frameworks/, [Accessed 19-02-2024].

[22] A. Biessek, *Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart 2*. Packt Publishing Ltd, 2019.

[23] A. Praveen, K. Nanda, N. Rajith, N. Giriraj, R. Radhika, N. Mahesh, K. Vishnu, T. Anjali, and S. Sarath, "Conference room booking application using flutter," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, 2020, pp. 0348–0350.

[24] A. B. Nair, G. Surendran, K. P. Prathyun, C. Sourag, V. Sivaprasad, and T. Anjali, "Developing a simple and intuitive smartphone launcher for the elderly using flutter," in *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2022, pp. 450–455.

[25] N. Kuzmin, K. Ignatiev, and D. Grafov, "Experience of developing a mobile application using flutter," in *Information Science and Applications: ICISA 2019*. Springer, 2020, pp. 571–575.

[26] R. Guntha, S. N. Rao, H. Muccini, and M. V. Ramesh, "Rapid yet robust continuous delivery of software for disaster management scenarios," *IEEE Software*, vol. 38, no. 4, pp. 104–113, 2020.

[27] R. C. C. Guntupalli, "User interface design: methods and qualities of a good user interface design," 2008.

[28] S. Z. Ahmed, "A comparison of novices'initial performance, learnability and memorability with a web-based ir interface," *Malaysian Journal of Library and Information Science*, vol. 11, no. 2, pp. 75–85, 2006.

[29] H. Aluri, A. Pal, and P. K. Singh, "Development of a mobile application for the digitalization of rural self-help groups (shgs)," in *2022 IEEE 19th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*. IEEE, 2022, pp. 151–156.

[30] H. Petrie and N. Bevan, "The evaluation of accessibility, usability, and user experience." *The universal access handbook*, vol. 1, pp. 1–16, 2009.