Enhanced User Experience in Digital TV: Recommending Content in Free-To-Air TV Contexts

Sergillam B. Oliveira*†, Tiago Souza*, Petrina Kimura*, Haydê Machado*, Orlewilson Maia*, Eddie B. de L. Filho*†, Edson W. A. do Nascimento‡, Leandro Negreiros§, Rafaell Guimarães Santana§, Lucas C. Cordeiro†¶

*TPV Technology, Manaus, Brazil

†Federal University of Amazonas, Manaus, Brazil

‡Núcleo de PD&I em Tec. da Informação, Com. e Automação - Virtus/UFCG, Campina Grande, Brazil

§FPF Tech, Manaus, Brazil

¶University of Manchester, Manchester, UK

{sergillam.oliveira, tiago.souza, petrina.kimura, hayde.machado, orlewilson.maia, eddie.filho}@tpv-tech.com, edson.nascimento@virtus.ufcg.edu.br, {leandro.negreiros,rafaell.santana}@fpf.br, lucas.cordeiro@manchester.ac.uk

***Abstract*—Digital television (DTV) systems represent a major leap forward from analog ones, offering superior audio and video quality and interactive capabilities. The latter allow viewers to engage with broadcasts in a manner similar to navigating the Internet, opening up possibilities for new services such as recommendation systems and targeted advertising. However, even with such capabilities, free-to-air TV is still mostly offered according to the push paradigm, where the available content is just sent while neither required by users nor based on their preferences. The present proposal addresses this gap and introduces a personalized recommendation methodology for DTV, employing three distinct approaches: active search, background, and on-the-fly recommendation modes, thus leveraging the capabilities of modern DTV receivers, mobile devices, and centralized data repositories. Our system enhances user control over personal data by using a smartphone as the primary interface for user interaction and specialized data processing, thus keeping sensitive information within it. Consequently, this approach supports privacy and reduces the need for centralized processing, thereby lowering costs.**

***Index Terms*—Recommendation System, Interactive Digital TV, Mobile App**

I. Introduction

Digital Television (DTV) systems mark a significant advancement over analog ones, providing high-quality audio and video along with interactivity in free-to-air (FTA) DTV. This allows viewers to actively engage with broadcast content, in a way that is similar to Internet navigation [1]. This transformation paves the way for new interactive services, including recommendation systems, targeted advertising, and educational platforms [2].

Among the new features that are possible with DTV systems, content recommendation emerges as a paradigm shift from the push strategy and a solution for the information overload caused by the vast amount of available programming. Additionally, the integration of new technologies, such as representational state transfer (REST) application programming interfaces (APIs) [3], is becoming a relevant trend to enhance DTV receivers. Through this approach, it is possible to efficiently connect receivers and external services, boosting recommendation capabilities and content personalization. Besides, the capacities of peripheral devices can also be used, which alleviates hardware requirements in DTV receivers and promptly extends their features without hardware or software upgrades.

Our proposal addresses these points, aiming to develop a personalized recommendation methodology for FTA DTV receivers, utilizing three recommendation approaches: active search, background, and on-the-fly modes. The first allows users to actively search for content of interest, using textual queries to find programs or web content that match their queries. The second, in turn, operates passively, continuously analyzing a user's consumption patterns and updating recommendations according to their preferences and viewing history. Finally, the third offers real-time recommendations based on a user's immediate behavior, such as changing channels or selecting a specific program.

Furthermore, it retrieves information from a DTV receiver and a mobile application, as well as from the Internet and a dedicated data server containing movies, series, soap operas, TV shows, and FTA programming. Recommendations are presented on either the same mobile device or DTV receiver, depending on which event triggered them.

In summary, this work creates a methodology that significantly enhances the user experience by offering personalized recommendations that combine personal preferences with television consumption patterns. This innovative approach, supported by modern technologies such as REST APIs and context-aware recommendation systems, can provide a more engaging and satisfying experience while optimizing the utilization of the vast array of content available on DTV networks and environments.

The remainder of this paper is organized as follows. Section II presents related studies, which help define the associated scientific basis. Then, Section III introduces the proposed methodology. After that, Section IV presents an implementation of it, exposes our experiments, and discusses application aspects, with the goal of proposal validation. Finally, the associated conclusions are shown in Section V.

II. Related Studies

Extensive research has been conducted to explore innovative solutions that aim to enhance the user experience within broadcast DTV systems [4]. These investigations address critical aspects such as content personalization, interactivity, and service quality focused on user aspects.

Content personalization, for instance, seeks to tailor television broadcasts to the specific interests of each viewer, employing advanced algorithms that analyze user behavior and recommend programs, movies, or series matching their profile [5]. Furthermore, interactivity has been a significant focus, enabling users to engage directly with content through features such as real-time voting, reviews, or additional information during a given FTA program.

Another key aspect is service quality, where the optimization of available technologies, such as data compression and high-definition transmission, aims to ensure that a viewing experience is smooth and engaging [6]. These studies enhance user engagement with television content and illustrate how new technologies can maximize the use of existing infrastructures, making media consumption more efficient and satisfying.

Costa [7] proposed the development of a REST platform designed to provide comprehensive TV programming data, along with personalized recommendations for users. It is accessible via any device with an internet connection and employs a robust and efficient search engine. Recommendations from it are generated through a hybrid algorithm that considers the similarity between users and attributes of the available programs. To ensure recommendation accuracy, both external and internal data were used, achieving a 15% success rate. This solution stands out in the Portuguese market as unique and promising, with significant business potential. Besides, there are expectations that it will be acquired by a major operator, thereby expanding its user base and features.

Santos Júnior and Ferraz [8] introduced a cloud-based flexible architecture for executing TV program recommendation algorithms, addressing the information overload caused by the large available programming. Their architecture enables the execution of various algorithms decoupled from the hardware of DTV set-top boxes, offering personalized recommendations as a service. The flexibility of this architecture was demonstrated through the implementation of two recommendation algorithms and a client application, highlighting its potential for adaptation and scalability.

Ávila and Zorzo [9] presented an architecture designed to support the development of recommendation systems in interactive DTV environments, aligning with the reference implementation of the middleware Ginga, which is currently known as DTV Play. Developing this architecture required the incorporation of new features into DTV Play that were not included in its original standard. It also reports on the results of experiments with a recommendation system, highlighting the effectiveness of the proposed architecture in enhancing viewer interaction. Indeed, this study is pertinent to our research as it demonstrates a practical application and the effectiveness of integrated solutions for content recommendation in DTV environments.

Lucas [5] introduced a recommendation system designed for multi-user environments in DTV networks, aimed at suggesting content based on viewer behavior patterns. Its architecture is embedded within a Set-top Box and employs implicit information collection, data mining using the Apriori algorithm [10], and information filtering techniques. The respective implementation was carried out using the Java TV API [11], thus adhering to the Brazilian DTV standard. The effectiveness of this system was assessed using data provided by the Brazilian Institute of Public Opinion and Statistics, showing positive results in personalized content recommendations. This study is also pertinent to our research as it offers a practical approach to recommendation systems in digital TV environments, showcasing techniques and technologies that could be integrated into our strategy.

Silva [12] exposed a software infrastructure designed for the development and execution of context-aware recommendation systems in interactive DTV networks. The proposed solution streamlines system design by handling low-level tasks, therefore allowing developers to focus on presentation logic aspects. This system employs flexible user and context modeling based on metadata standards, such as moving picture expert group number 7 [13] and TV-Anytime [14], and incorporates advanced techniques for content recommendation and context processing. Its architecture consists of two subsystems, with context-aware information filtering performed using content-based techniques. To validate its functionalities, a context-aware recommendation system was developed as a case study. This research is pertinent to our study as it provides a detailed and practical approach to implementing recommendation systems in interactive DTV environments, highlighting techniques that could be adapted to enhance our solution.

Despite the advancements achieved by existing approaches, there are still gaps that require more robust and innovative solutions. Firstly, many rely on large volumes of centralized processing, which can lead to scalability issues and increased operational costs. Secondly, while some initiatives leverage user profiles for content recommendation, few explore mobile devices as primary interaction interfaces with recommendation systems. This aspect is especially pertinent within the context of free-to-air TV environments, where there is a wide range of connected devices, a need for greater flexibility in data collection, and low processing requirements. Moreover, previous studies do not sufficiently address user privacy and personal data protection, which are critical issues. Therefore, our work addresses these gaps by proposing the integration of recommendation approaches that utilize mobile devices and distributed systems, thus providing greater flexibility, reducing processing costs, and ensuring user data protection.

III. The Proposed Architeture

Our work aims to develop an architecture capable of providing content recommendations to users of DTV services. It is in charge of suggesting content from DTV programming schedules and streaming services, at least, as well as videos, images, and texts found on the Internet.

The proposed architecture is illustrated in Fig. 1. There are three main elements: a DTV Receiver, a mobile device, and a data service. These elements form the core components of the architecture by distributing the tasks of content collection, processing, and display among digital a TV receiver, a mobile device, and a data service.

Currently, most DTV receivers provide external access via a local network, including wireless fidelity (WiFi) or Ethernet interfaces. In fact, peripherals may be able to connect to DTV receivers to request services, trigger actions, and access information, which is the case with DTV Play in Brazil [1]. Besides, middleware standards already provide enhanced connectivity [15], which includes a REST API for requesting programming and even table information [16]. Such capabilities favor data acquisition for several purposes, including content recommendation.

To interact with a DTV receiver and embed our recommendation methodology, we chose a smartphone due to its pervasive characteristics and hardware increasingly capable of performing advanced operations. An application on a smartphone can connect to a DTV receiver via REST API, using the hypertext transfer protocol, and access information about user behavior within it. In addition, this application can keep personal information in the smartphone's storage, giving a user more privacy and control over his data.

Indeed, to keep user data only within a smartphone, all processing related to it must be implemented on this same device. On the one hand, our recommendation algorithms must be built into a smartphone and operate via an on-device machine. This strategy, in addition to contributing to privacy,

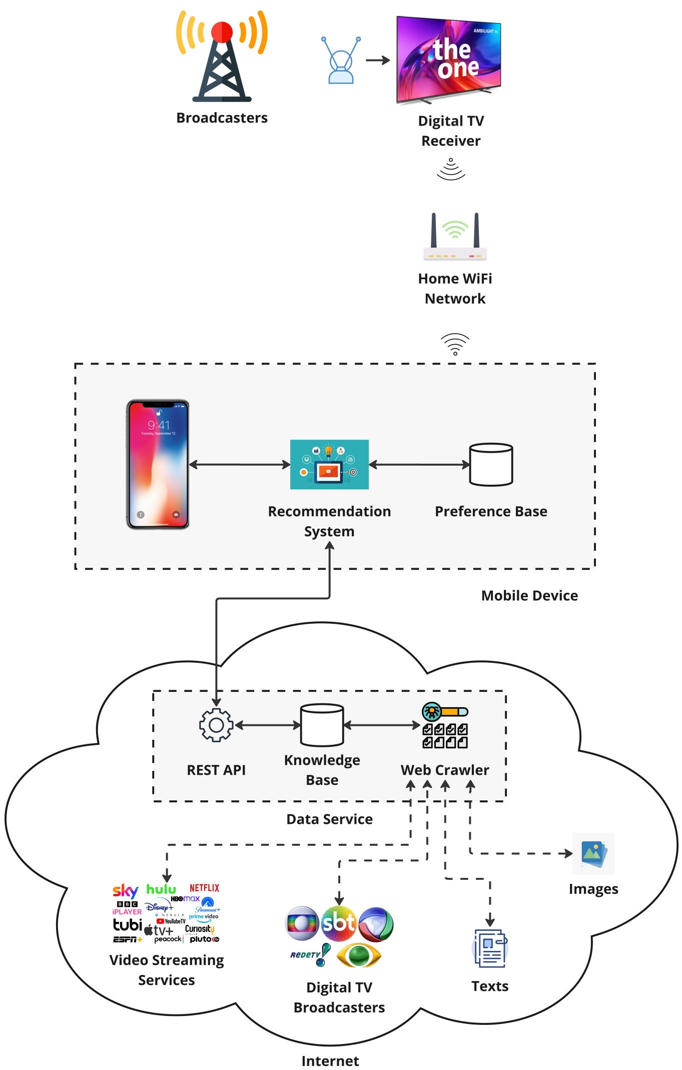

Fig. 1. Architecture of the proposed system.

also eliminates processing costs on a centralized server. On the other hand, a catalog of diverse content is too large to be kept on a smartphone, from which arises the need for a centralized repository to store and provide the catalog of items to the application. However, this general information is used to find content to recommend and has no relation to the personal data stored on a smartphone.

Personal data is stored only on a preference base. Its objective is to keep user behavior, which can be obtained explicitly by asking the user about their preferences or implicitly collecting user interaction, such as viewing history on DTV receiver and smartphone. This data will be used by the underlying recommendation algorithm, incorporated into a mobile application, both to obtain candidate content in a centralized repository and rank them according to a user's preferences.

The data service element is responsible for collecting, transforming, and storing all information that is necessary for recommendation procedures within a mobile application. To keep this data updated, there can be a routine for acquiring raw content data from various public sources on the Internet. Besides, to make the management and retrieval of the data easier, it can be held in a common representation in a relational database.

Indeed, our recommendation system aims to suggest content from TV and streaming services. Consequently, we need to collect items from both sources. A DTV receiver can provide information about the TV catalog through the electronic programming guides (EPGs) sent by broadcasters, but consulting this source faces significant challenges that compromise its effectiveness. For instance, we can mention the absence of this information when its tables are sent empty by broadcasters and generic descriptions where information about programs tends to be superficial, not offering details about specific topics addressed during transmissions [17].

To overcome this challenge, we can take advantage of EPG web portals that centralize this data and make it publicly available, e.g., *mi.tv*. Moreover, streaming content is also publicly available on web portals such as IMDB and Rotten Tomatoes. This variety of sources provides content diversity but also raises the need for a specific data engineering pipeline for each source, i.e., we cannot use the same pipeline of web scraping for both IMDB and Rotten Tomatoes, which must be included into the data Service element. For suggestions of text and images, data Service can behave as a proxy for parsing search engine result pages, thus extracting their content and returning it for the recommender application[1,2,3].

Given its components, we can summarize how the architecture works as follows. A DTV receiver provides a query interface about what the user is watching, in real-time. Then, a smartphone retrieves this data and uses it to build a user profile in conjunction with an explicit collection of information. Next, the same application retrieves candidates from data service and classifies them, using its user preference base, for presentation.

It is worth noticing that by prioritizing user privacy, thus keeping user data and its processing only within a smartphone application, it is difficult to apply a recommendation through collaborative filtering, in the background. Therefore, it is necessary to use a content filtering-based approach [18].

IV. An Implementation of the Proposed Architecture and Some Results

To validate our methodology, we have implemented it using a DTV receiver emulator [19] adapted to the Brazilian digital television system [15], which provides a REST API that we can use to acquire information related to the DTV program currently being broadcast, such as broadcaster name and program description. Indeed, very often, DTV broadcasters do not provide the program description within transport stream tables [20]. However, we can work it around by searching for program data online using a broadcaster's name and current time, which is called EPG online.

We have also extended this emulator to provide API methods for (i) displaying notifications and consulting interactions with these notifications; (ii) displaying complementary content on-screen, such as texts and images combined with the current program, also retrieving a user's interactions with these complementations; (iii) consulting a user's watch history; and (iv) retrieving the current DTV program's transcription The latter appears as an alternative to get around the lack of descriptions about the current program in EPGs sent by the broadcasters, where a DTV receiver provides an endpoint to retrieve the

[1]https://mi.tv
[2]https://www.imdb.com/
[3]https://www.rottentomatoes.com/

current audio stream. From this, we were able to trigger a task using the Vosk API to transform audio into text [4].

The recommendation system processing was implemented on a mobile application developed with Flutter. It provides the three main modes of the proposed recommendation methodology: (i) background recommendation, (ii) on-the-fly complementation, and (iii) active search. The first aims to suggest content aligned with a user's interests. However, a challenge faced by any recommender system regarding this kind of content is the cold start. We addressed this problem by explicitly asking users to choose some options that represent their preferences from an initial survey list, which are taken as the first user profile data. The other modes do not exactly suffer from it [5].

For generating suggestions in background mode, this mobile application retrieves a set of candidates from the data service element, being responsible for providing structured data gathered by web scraping routines that access various sources, e.g., EPG online, streaming catalogs, and public online movie databases such as TMDB. Data service was implemented in a way to provide a REST API for data retrieval along with data collection routines, keeping it in a PostgreSQL database [6,7].

In the recommender application, the candidates retrieved from the data service element are ranked according to a user's profile data. The latter is a set of contents generated by a user according to his interaction with the same application and a DTV receiver, including watch history, ratings, and initial survey choices. Due to privacy issues, all user profile data is maintained only within a mobile device. Besides, suggestions are presented to a user on the application's home screen, and a user can interact with them by viewing details of, watching, or rating them, as shown in Fig. 2.

The active search mode involves utilizing user text inputs as search keys to retrieve data from distinct sources, including the Internet and data service. The search on the web is made by web-scrapping search engines' result pages. In addition, the associated textual query is submitted to the data service element to retrieve matching contents. We implemented a speech-to-text feature to allow the easy use of this feature.

The on-the-fly mode retrieves and uses content transcriptions related to what a user is currently watching on a DTV receiver. To understand the context of the material being consumed, we use Latent Dirichlet Allocation (LDA) and a named entity recognition (NER) algorithm [21], [22]. LDA is employed to obtain the predominant topics from content, while NER extracts possible named entities. The information derived from these techniques is then used as keywords for a search on the web, whose result is presented to a user with notifications on the screen of a DTV receiver and a smartphone. This process occurs repetitively in the background of the recommender mobile application, i.e., it frequently retrieves transcriptions and seeks content aligned with them.

This implementation provides a way to complement what is happening, in real-time, on a DTV receiver while seeking to understand user preferences to recommend personalized content. Fig. 3 presents a view of the implementation design and two use-case scenarios, which were developed to test the integration of our architecture's components: DTV receiver, smartphone, and data Service.

[4]https://github.com/alphacep/vosk-api
[5]https://flutter.dev/
[6]https://www.postgresql.org/
[7]https://www.themoviedb.org/

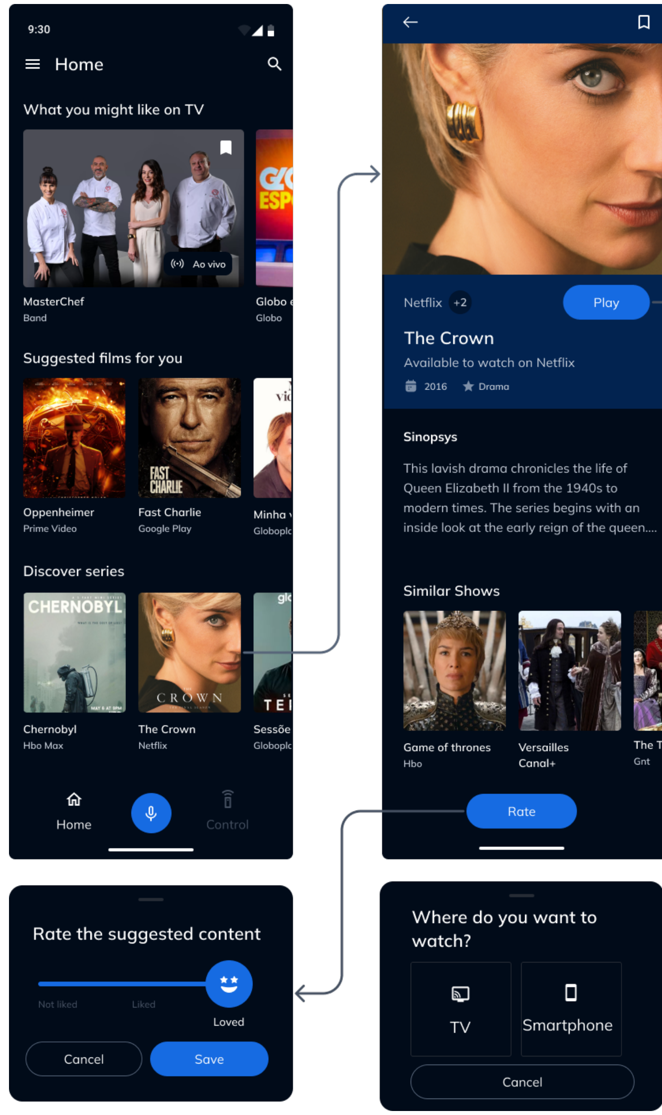

Fig. 2. The recommender application's home and content detail screens, where a user can watch or rate the selected content.

Our experiments, utilizing the developed infrastructure, were conducted using a DTV emulator running on a standard PC equipped with 16 GB of random access memory (RAM) and an 11th Generation Core i7 central processing unit (CPU) with a clock of up to 4.70 GHz, running the operating system Ubuntu. The peripheral device, implemented as a mobile application, was executed on smartphones equipped with 4 GB of RAM and a 2.3 GHz octa-core CPU.

The first scenario regards the suggestion of content similar to the current TV program. The smartphone running our recommendation application checks if the current program is close to finishing and then consults the data service element to retrieve similar content. After that, the recommendation system sends a notification to the target DTV receiver, informing that a user can watch content similar to what is already being presented. This same user can then choose to check these suggestions or dismiss the respective notification. If he chooses the first option, the recommending system will present a screen with suggestions ordered according to their similarity with the current program.

The second scenario refers to the on-the-fly complementation mode. After recovering content on the web related to the transcription of the current program, the recommender application sends a notification to the DTV receiver, allowing a user to read complementary on-screen content along with the current program. If he accepts it, the current program is resized and the found web content is displayed. At this stage, a user can also choose to read it on his smartphone's screen, while the program presented by a DTV receiver returns to full screen.

The developed mobile application presented a final size of 98.5 MB on Android. During application execution, the average RAM consumption was 82 MB when in idle mode, reaching a maximum of 2.7 GB under intensive usage conditions. These results indicate that the memory usage impact on smartphones is within acceptable limits of mid-end smartphones, ensuring stable and efficient performance.

Regarding the module embedded in a DTV receiver, which is used for audio transcription and communication, and also considering its shared libraries, the total RAM footprint obtained stayed around 180 MB. As a reference, low-end commercial DTV receivers, which are massively used in DTV networks, usually possess 1GB of RAM. Consequently, taken into consideration parallel operation with other services, such a value is high, which undoubtedly leads to optimization regarding the receiver's memory consumption.

V. Conclusion

The proposed architecture for content recommendation in DTV networks leverages the strengths of modern DTV receivers, mobile devices, and centralized data repositories. By utilizing a smartphone as the primary interface for user interaction and data processing, it ensures enhanced privacy and avoids new requirements related to DTV receivers. In other words, the integration of recommendation algorithms directly into a smartphone supports user privacy, not exchanging personal data, and reduces the need for centralized processing, thereby cutting costs.

Moreover, our architecture was designed to mitigate processing overhead by distributing recommendation and complementary tasks between a DTV receiver, a mobile device, and a data service element. This way, it significantly reduces computational resource usage on the DTV receiver itself. In addition, its implementation demonstrates the feasibility and effectiveness of the whole content recommendation system for digital TV services.

Regarding our experiments, they showed that by utilizing a DTV receiver emulator with an extended REST API, we were able to simulate real-world scenarios and validate our approach. Besides, the mobile application, developed using the Flutter framework, successfully integrates the three main components of our recommendation methodology: background recommendation, on-the-fly complementation, and active search.

Our quantitative results demonstrate that the mobile application operates within acceptable limits of RAM and storage consumption, regarding mid-end devices, thus ensuring stable and efficient performance. The application's size on Android remained under 100 MB, with an average RAM consumption of 82 MB in idle state, which may increase to 2.7 GB during intensive computation.

However, the embedded module in DTV receivers needs optimization for RAM consumption. Indeed, the current values are limiting, regarding low-end products.

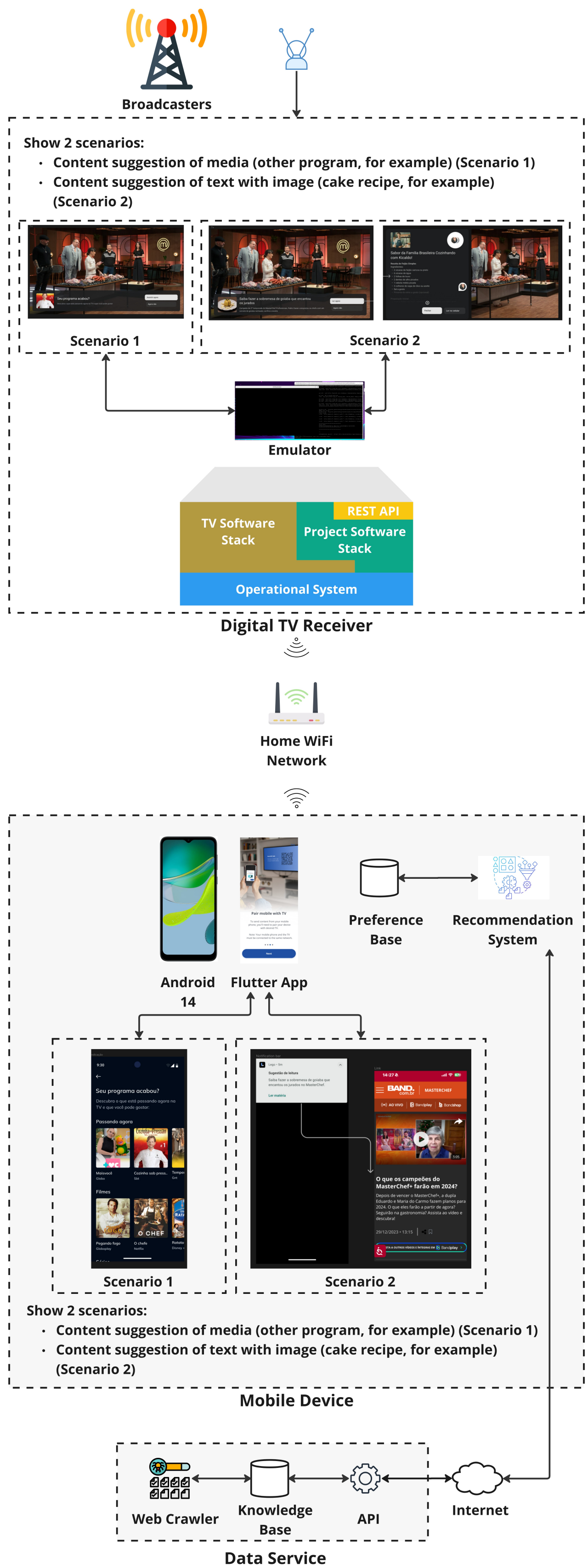

Fig. 3. The implementation design and use case scenarios for our experiments.

Overall, our system provides a robust solution for enhancing user experience in DTV services by offering personalized content recommendations. Future work will focus on refining the recommendation algorithms, expanding the data sources, and improving the user interface to further enhance the system's performance and user satisfaction.

ACKNOWLEDGEMENT

The results presented in this paper were sponsored by ENVISION Indústria de Produtos Eletrônicos LTDA, under the terms of Brazilian federal law No. 8.387/91 (SUFRAMA). This research was conducted by Envision (TPV Group).

REFERENCES

[1] P. Kimura, B. Farias, J. Costa, R. Fabrício, and E. B. de Lima Filho, "A methodology for upgrading legacy middleware ginga implementations to profile ginga-d," in *2020 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–5, IEEE, 2020.

[2] B. de Sousa Monteiro, "Amadeus-tv: Educational portal on digital tv integrated into a learning management system," Master's thesis, Federal University of Pernambuco, 2009. (in Portuguese).

[3] D. Xavier, F. Bandeira, A. Conceição, R. Fabrício, J. Fernandes, E. de Lima Filho, D. Araújo, R. Mello, S. Silva, and T. Lima, "Augmenting digital tv receivers with rest apis," in *2024 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–6, IEEE, 2024.

[4] J. E. Fernandes, A. R. D. S. Conceição, O. B. Maia, E. B. de Lima Filho, L. L. de Paula, and M. V. Said, "Conducting usability testing on consumer electronics: A real experience," in *2024 International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan)*, pp. 629–630, IEEE, 2024.

[5] A. d. S. Lucas, "Personalization for digital television using the recommendation system strategy for multi-user environments," 2009. (in Portuguese).

[6] O. B. Maia, H. C. Yehia, and L. de Errico, "A concise review of the quality of experience assessment for video streaming," *Computer communications*, vol. 57, pp. 1–12, 2015.

[7] T. F. M. da Costa, "Television information system: Research and recommendations," Master's thesis, Polytechnic Institute of Porto (Portugal), 2017. (in Portuguese).

[8] J. R. dos Santos Júnior and C. A. G. Ferraz, "A flexible architecture to support the execution of program recommendation algorithms in connected tv," in *Proceedings of the XVII Brazilian Symposium on Multimedia and the Web*, pp. 97–102, SBC, 2011. (in Portuguese).

[9] P. M. de Ávila and S. D. Zorzo, "Support for the development of recommendation systems in interactive digital television environments," Master's thesis, Federal University of São Carlos, 2010. (in Portuguese).

[10] B. Sinaga, M. Marpaung, I. R. B. Tarigan, and K. Tania, "Implementation of stock goods data mining using the apriori algorithm," *Sinkron: jurnal dan penelitian teknik informatika*, vol. 7, no. 3, pp. 1980–1992, 2023.

[11] M. Vidakovic, T. Maruna, N. Teslic, and V. Mihic, "A java api interface for the integration of dtv services in embedded multimedia devices," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 3, pp. 1063–1069, 2012.

[12] F. S. d. Silva, *PersonalTVware: an Infrastructure to Support Context-Aware Recommendation Systems for Personalized Digital TV*. PhD thesis, University of São Paulo, 2011. (in Portuguese).

[13] ISO – International Organization for Standardizatio, "Iso/iec 15948-1: Mpeg-7 overview," technical specification, ISO/IEC, 2004.

[14] TV-Anytime Forum, "Tv-anytime – part 3: Metadata," Technical Specification TS102 822-3-1 v1.4.1, ETSI, 2007.

[15] Brazilian Association of Technical Standards, *ABNT NBR 15606-11, Digital terrestrial television – Data coding and transmission specification for digital broadcasting – Part 11: Ginga CC WebServices – Ginga Common Core WebServices specification*, 2021.

[16] B. C. de Farias, E. B. de Lima Filho, O. B. Maia, and E. Souto, "Extensions to middleware ginga for integration with iot environments," in *2020 IEEE International conference on consumer electronics (ICCE)*, pp. 1–5, IEEE, 2020.

[17] F. Narducci, C. Musto, M. de Gemmis, P. Lops, and G. Semeraro, "Tv-program retrieval and classification: A comparison of approaches based on machine learning," *Information Systems Frontiers*, vol. 20, pp. 1157 – 1171, 2017.

[18] S. K. Raghuwanshi and R. Pateriya, "Collaborative filtering techniques in recommendation systems," *Data, Engineering and Applications: Volume 1*, pp. 11–21, 2019.

[19] B. Farias, I. Machado, C. Pinheiro, T. Souza, E. de Lima Filho, R. Cal, W. Luiz, M. Oliveira, and M. Said, "An architecture to emulate television receivers," in *2024 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–6, IEEE, 2024.

[20] F. Izumi, E. B. de Lima Filho, L. C. Cordeiro, O. Maia, R. Fabrício, B. Farias, and A. Silva, "A fuzzing-based test-creation approach for evaluating digital tv receivers via transport streams," *Software: Testing, Verification and Reliability*, vol. 33, pp. 1–31, October 2022.

[21] K. W. Church, "A stochastic parts program and noun phrase parser for unrestricted text," in *Proceedings of the Second Conference on Applied Natural Language Processing*, ANLC '88, (USA), p. 136–143, Association for Computational Linguistics, 1988.

[22] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.