

Team Report 1.3 – Technical Report

DeepFakeChain

Revision: v.1.0

Due date	16/01/2023
Submission date	16/01/2023
Version	1.0
Authors	Nikolaos Giatsoglou (CERTH) Symeon Papadopoulos (CERTH) Dora Kallipolitou (Zelus) Stella Markopoulou (Zelus)



Grant Agreement No.: 957228
Call: H2020-ICT-2018-2020
Topic: ICT-54-2020
Type of action: RIA

Document Revision History

Version	Date	Description of change	List of contributor(s)
v0.1	01/12/2021	ToC version circulated by TruBlo consortium	TruBlo consortium
v0.2	20/12/2022	Created structure and assigned sections	CERTH
v0.3	4/1/2023	Completed core diagrams	CERTH & Zelus
v0.4	9/1/2023	Completed first draft	CERTH & Zelus
v0.5	13/1/2023	Revisions	CERTH & Zelus
v1.0	15/1/2023	Version ready for submission.	CERTH

DISCLAIMER

The information, documentation and figures available in this deliverable are written by the DeepFakeChain team and do not necessarily reflect the views of the TruBlo consortium or of the European Commission. The TruBlo consortium and the European Commission are not liable for any use that may be made of the information contained herein.

Project co-funded by the European Commission in the H2020 Programme	
Nature of the deliverable:	R: Document, report
Dissemination Level:	CO: Confidential to TruBlo project and Commission Services

EXECUTIVE SUMMARY

This document describes the technical implementation of DeepFakeChain, including the architecture of the platform, the data model, the trust & security aspects, and the technical standards that will be applied during development. The document also presents the already implemented components of the platform that form part of DeepFakeChain's initial prototype and the tests that are planned to evaluate it.

TABLE OF CONTENTS

EXECUTIVE SUMMARY.....	3
TABLE OF CONTENTS.....	4
LIST OF FIGURES.....	6
LIST OF TABLES	7
ABBREVIATIONS.....	8
1 PROJECT DESCRIPTION	9
2 TECHNICAL ARCHITECTURE – BEHAVIORAL VIEW.....	11
2.1 CONCEPTS.....	11
2.2 AUTHENTICATION USE CASES	11
2.3 CHANNEL MANAGEMENT USE CASES.....	12
2.4 MEDIA MANAGEMENT USE CASES	13
2.5 MEDIA VERIFICATION USE CASES.....	14
2.6 SEQUENCE DIAGRAMS.....	15
3 TECHNICAL ARCHITECTURE – STRUCTURAL VIEW.....	22
3.1 OVERVIEW	22
3.2 COMPONENT DIAGRAMS	23
4 TRUST & SECURITY FRAMEWORK	25
4.1 TRUST FRAMEWORK.....	25
4.1.1 USER TRUST.....	25
4.1.2 ALGORITHM TRUST.....	26
4.1.3 PLATFORM TRUST	26
4.2 SECURITY FRAMEWORK.....	26
5 TECHNICAL REQUIREMENTS	28
5.1 DEVELOPMENT REQUIREMENTS	28
5.2 DEPLOYMENT REQUIREMENTS	28
5.3 INTERFACE REQUIREMENTS	29
5.4 SECURITY REQUIREMENTS.....	29
5.5 PERFORMANCE REQUIREMENTS.....	30
6 DATA MODEL.....	31
7 DATA PROTECTION & PRIVACY	32
8 APPLICABLE STANDARDS	33
9 INITIAL PROTOTYPE INFORMATION	34
9.1 DEEPFAKECHAIN MOCKUP INTERFACE	34
9.2 REVERSE MEDIA SEARCH	39
10 METRICS & PLANNED TESTS	40
11 TECHNICAL INNOVATION.....	41

12	RISKS AND MITIGATION	41
	REFERENCES.....	42



LIST OF FIGURES

FIGURE 1: THE AUTHENTICATION USE CASES OF DEEPFAKECHAIN.....	12
FIGURE 2: THE CHANNEL MANAGEMENT USE CASES OF DEEPFAKECHAIN.....	13
FIGURE 3: THE MEDIA MANAGEMENT USE CASES OF DEEPFAKECHAIN.....	14
FIGURE 4: THE MEDIA VERIFICATION USE CASES OF DEEPFAKECHAIN.....	15
FIGURE 5: THE BASIC COMPONENTS OF DEEPFAKECHAIN.....	16
FIGURE 6: SEQUENCE DIAGRAM OF A VIEW OBJECT OPERATION	17
FIGURE 7: SEQUENCE DIAGRAM OF AN ADD/EDIT OBJECT OPERATION.....	17
FIGURE 8: SEQUENCE DIAGRAM OF A DELETE OBJECT OPERATION	18
FIGURE 9: SEQUENCE DIAGRAM OF THE MEDIA UPLOAD OPERATION.....	19
FIGURE 10: SEQUENCE DIAGRAM OF THE MACHINE EVALUATION OPERATION	20
FIGURE 11: SEQUENCE DIAGRAM OF THE HUMAN VERIFICATION OPERATION.....	21
FIGURE 12: THE GENERAL STRUCTURE OF DEEPFAKECHAIN.....	22
FIGURE 13: THE COMPONENT DIAGRAM OF DEEPFAKECHAIN	23
FIGURE 14: THE DEPLOYMENT DIAGRAM OF DEEPFAKECHAIN	24
FIGURE 15: THE DATA MODEL OF DEEPFAKECHAIN.....	31
FIGURE 16: THE REGISTRATION PAGE OF THE DEEPFAKECHAIN MOCKUP.....	34
FIGURE 17: THE LOGIN PAGE OF THE DEEPFAKECHAIN MOCKUP.....	35
FIGURE 18: SEARCH MEDIA FUNCTIONALITY IN THE DEEPFAKECHAIN MOCKUP.....	35
FIGURE 19: UPLOAD MEDIA FUNCTIONALITY IN THE DEEPFAKECHAIN MOCKUP.....	36
FIGURE 20: SEARCH CHANNEL FUNCTIONALITY IN THE DEEPFAKECHAIN MOCKUP....	36
FIGURE 21: THE CHANNEL PAGE OF THE DEEPFAKECHAIN MOCKUP	37
FIGURE 22: VIEWING CHANNEL DETAILS IN THE DEEPFAKECHAIN MOCKUP	37
FIGURE 23: REVERSE MEDIA SEARCH IN THE DEEPFAKECHAIN MOCKUP.....	38
FIGURE 24: THE MEDIA PAGE OF THE DEEPFAKECHAIN MOCKUP	38

LIST OF TABLES

TABLE 1: DEVELOPMENT REQUIREMENTS	28
TABLE 2: DEPLOYMENT REQUIREMENTS.....	28
TABLE 3: SECURITY REQUIREMENTS.....	29
TABLE 4: PERFORMANCE REQUIREMENTS.....	30
TABLE 5: STANDARDS FOR THE DEVELOPMENT OF DEEPFAKECHAIN	33
TABLE 6: EVALUATION METRICS FOR DEEPFAKECHAIN	40
TABLE 7: PLANNED TESTS FOR THE EVALUATION OF DEEPFAKECHAIN.....	40
TABLE 8: FEATURE RISK OF DEEPFAKECHAIN	41
TABLE 9: SCALABILITY RISK OF DEEPFAKECHAIN	42

ABBREVIATIONS

Abbreviations	Definitions
AES	Advanced encryption standard
AI	Artificial intelligence
API	Application programming interface
CERTH	Centre of research and technology Hellas
CPU	Computer processing unit
CRUD	Create read update delete
DDOS	Distributed denial-of-service
DFC	DeepFakeChain
DFDC	Deepfake detection challenge
DOS	Denial-of-service
DPO	Data protection officer
GDPR	General data protection regulation
GPU	Graphics processing unit
HTTP	Hypertext transfer protocol
HTTPS	Hypertext transfer protocol secure
ID	Identity
ISO	International organization for standardization
JSON	Javascript object notation
P2P	Peer-to-peer
RAM	Random access memory
REST	Representational state transfer
SHA	Secure hash algorithms
TCP	Transmission control protocol
ToS	Terms of service
UI	User interface
URL	Uniform resource locator
XML	Extensible markup language

1 PROJECT DESCRIPTION

Project name	DeepFakeChain
Link to project on TruBlo website	https://www.trublo.eu/deepfakechain/
Primary contact	Dr Symeon Papadopoulos, papadop@iti.gr
Project members	Mr Nikolaos Giatsoglou, ngiatsog@iti.gr Dr George Kordopatis, georgekordopatis@iti.gr Ms Stella Markopoulou, s.markopoulou@zelus.gr
Organisation(s)	CERTH, Zelus
Organisation's website	https://www.certh.gr , https://www.zelus.gr
Short project summary	
What is the focus of your project?	The development of a scientific testbed to research innovative deepfake detection algorithms and combinations with human judgment. The project targets the media sector and can also be extended to the content moderation use case by extending to the detection of more general harmful content. Blockchain technology will be used to notarise the data that is uploaded on our testbed and enhance the trustworthiness of the media evaluations.
Why is a new/better solution needed?	The generation of deepfake synthetic media that are exceedingly difficult to detect is a worrying trend with potentially devastating impact on society. Currently, no solution exists for perfect automatic detection, nor one is expected in the near future due to the ongoing refinement of generated deepfakes. To address this issue, DeepFakeChain combines algorithmic solutions with human expert opinion, which is better in discerning context and valuable for detection. Our solution could also cover content moderation, which is an emerging market with few established solutions.
How will your solution be better?	By integrating expert human judgement with automatic evaluations, capitalising on the best of both worlds. By offering access and explainability features of cutting-edge AI algorithms to non-technical professionals. By increasing the trustworthiness of our platform's data and decisions through blockchain.
Extra: How does this project contribute to "trustable content on future blockchains"	By storing proofs of authenticity of our platform's data (media, annotations, user profiles, decisions) in a distributed blockchain network, making them available to the platform's end users and third parties for validation and auditing.
Type of project	<input checked="" type="checkbox"/> (X) Scientific/research <input type="checkbox"/> () Commercial, potential startup <input type="checkbox"/> () Open source, non-commercial <input type="checkbox"/> () Other, pls add 1-4 words if selected

Technologies used	AI-based deepfake detection algorithms, ensemble learning, consensus and truth-discovery algorithms, permissioned blockchain network, reverse media search
Use of Alastria resources	Yes

2 TECHNICAL ARCHITECTURE – BEHAVIORAL VIEW

The behavioural view of the DeepFakeChain architecture consists of the use cases that the platform supports, which were first described in the sections 5.2 & 6 of TR1.2. In this deliverable, we have refined the use cases and tailored them to the needs of our prototype that is under development. In more detail, the use cases are split in categories related to i) authentication, ii) channel management, iii) media management, and iv) media verification. These categories are elaborated in the following sections, preceded by a small section that defines the main concepts of DeepFakeChain, repeated from TR1.2 for easier reference.

2.1 CONCEPTS

DeepFakeChain is designed around the following concepts:

- **User profiles**, standard profiles containing personal information such as name, affiliations, and contact information. All users receive a user profile upon registration.
- **Media**, images or videos uploaded from local or online sources, and stored within the platform in case of future take-downs. Evidence of their authenticity is stored at the blockchain network.
- **Channels**, collaborative workspaces where users from different organisations can jointly verify media of a given topic, e.g., the Ukrainian war. Users can join and assign media to them.
- **Annotations**, notes left by the users on the authenticity of a media, possibly pointing to specific spatiotemporal windows inside the media. Only channel members can annotate the media of a channel.
- **Comments**, general notes left by the users on the media's page. All users can leave comments regardless of channel membership. In contrast to annotations, comments form dialogues.

2.2 AUTHENTICATION USE CASES

The authentication use cases of DeepFakeChain are depicted in Figure 1. We see that **registration and approval from the platform's administrators (i.e., the DeepFakeChain team) are required to access DeepFakeChain's services**. In particular, the approval process helps to screen the users, verify their affiliations, and establish a pre-existing level of trust. This is reasonable considering that the platform targets professionals, at least in its initial design, while future designs could incorporate more transparent processes and criteria for approval. If the registration is successful, a new user profile is created, the user is notified via email, and is free to login the platform with the registered credentials.

Excluding the administrators, **the platform foresees a flat user hierarchy**, consistently with TR1.2. We note that in TR1.1, we had described detailed user roles, which were simplified in TR1.2 to focus on the scientific aspects of the platforms in Phase 1. These user roles will be considered in Phase 2 to evolve the platform towards a more complete product. Besides, the current flat hierarchy is attractive considering the platform's professional and collaborative nature. **Some user privileges concerning asset management arise naturally based on asset ownership and channel membership**. They will be elaborated in the use cases of the next sections.

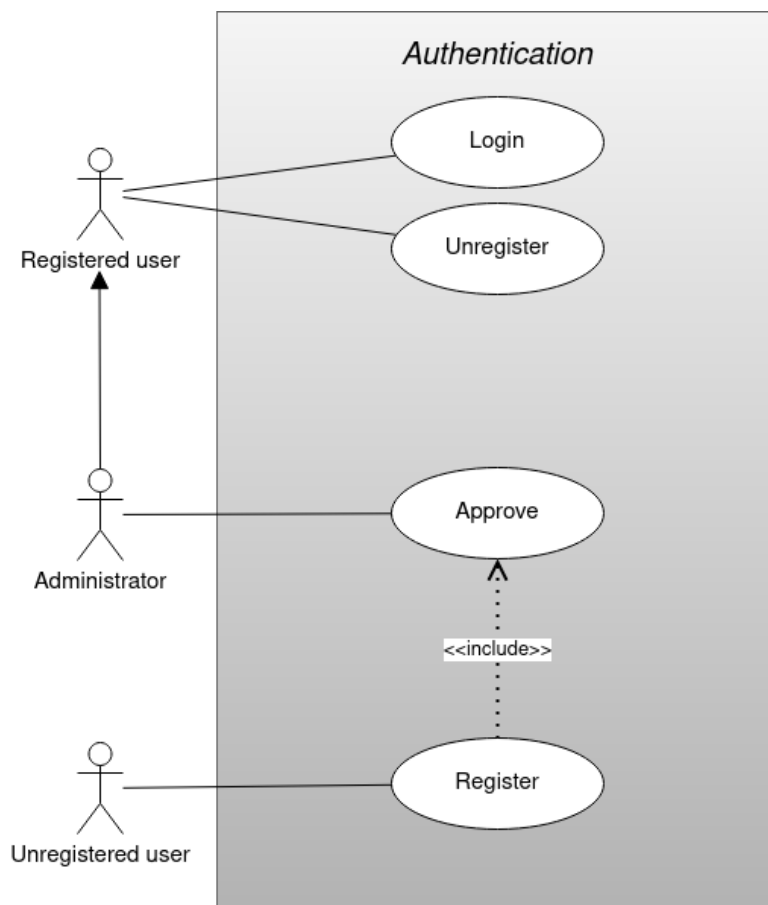


FIGURE 1: THE AUTHENTICATION USE CASES OF DEEPFAKECHAIN

2.3 CHANNEL MANAGEMENT USE CASES

The channel management use cases of DeepFakeChain are depicted in Figure 2. We see that all users are free to search, view, create, and join channels. While channel membership does not appear to add any privileges, it is required for verification-related activities such as annotating and evaluating media, which will be described in section 2.5. In addition, editing and deleting channels are more sensitive operations, hence they are permitted only to the channel creators. Please note that **deleting a channel does not delete its associated media**, which are tied to their uploaders.

In Phase 2, we will consider adding more fine-grained visibility options so that channel creators can decide if their channels should be private or public and if membership is open or through approvals / invitations.

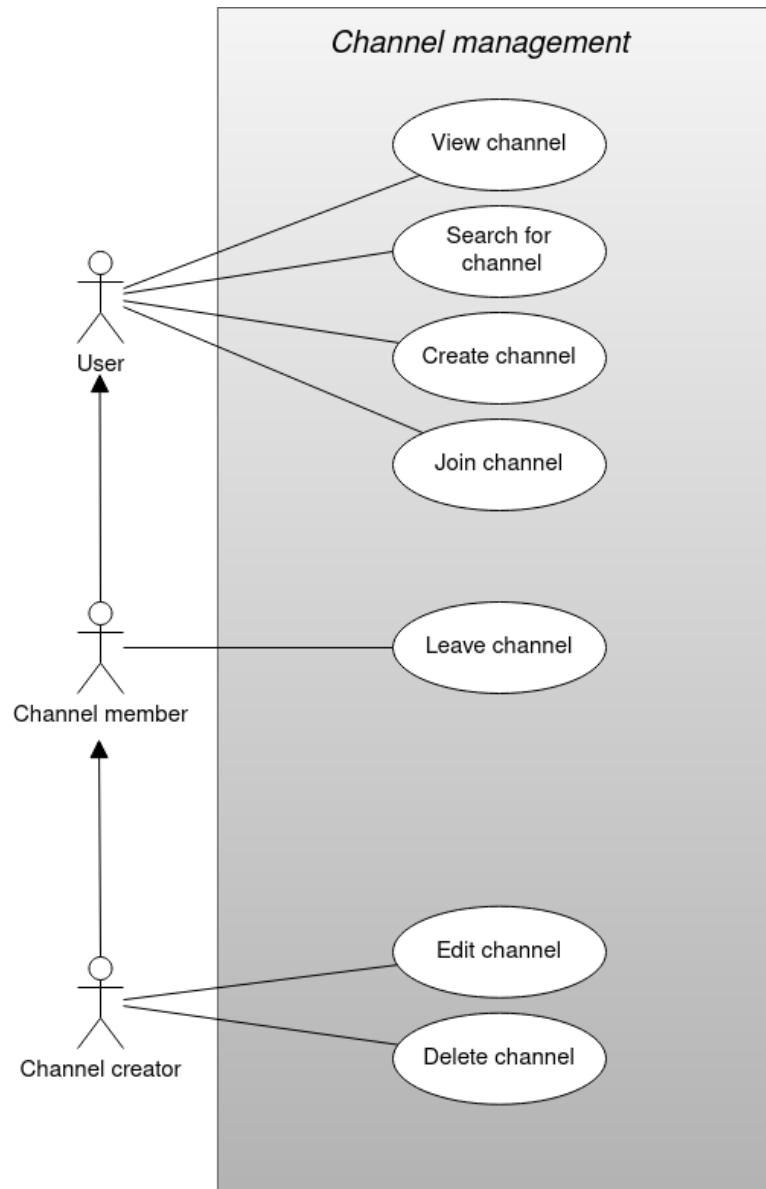


FIGURE 2: THE CHANNEL MANAGEMENT USE CASES OF DEEPFAKECHAIN

2.4 MEDIA MANAGEMENT USE CASES

The media management use cases of DeepFakeChain are depicted in Figure 3. We see that the DeepFakeChain users are free to view, search, and upload media, as well as mark them as favourites for easier reference. We note that DeepFakeChain supports both keyword search and reverse media or near duplicate search. As with channels, sensitive operations such as **edit and delete are permitted only to the original media uploaders**. Delete operations in particular are especially sensitive if the media has already gathered annotations and comments from other users. Due to this reason, if such an operation is requested, the user will be warned and requested to confirm deletion. Furthermore, **only the media uploaders are allowed to associate media with channels**, provided that they are channel members. This implies that other users will not be allowed to associate a foreign media to their channels unilaterally, significantly easing management. On the other hand, media uploaders can associate a media with multiple channels of which they are members.

As with channel management, in Phase 2, we will consider more fine-grained visibility options for media, allowing different levels of access to the platform's media and their metadata, as well as granting rights from the media uploaders to other users.

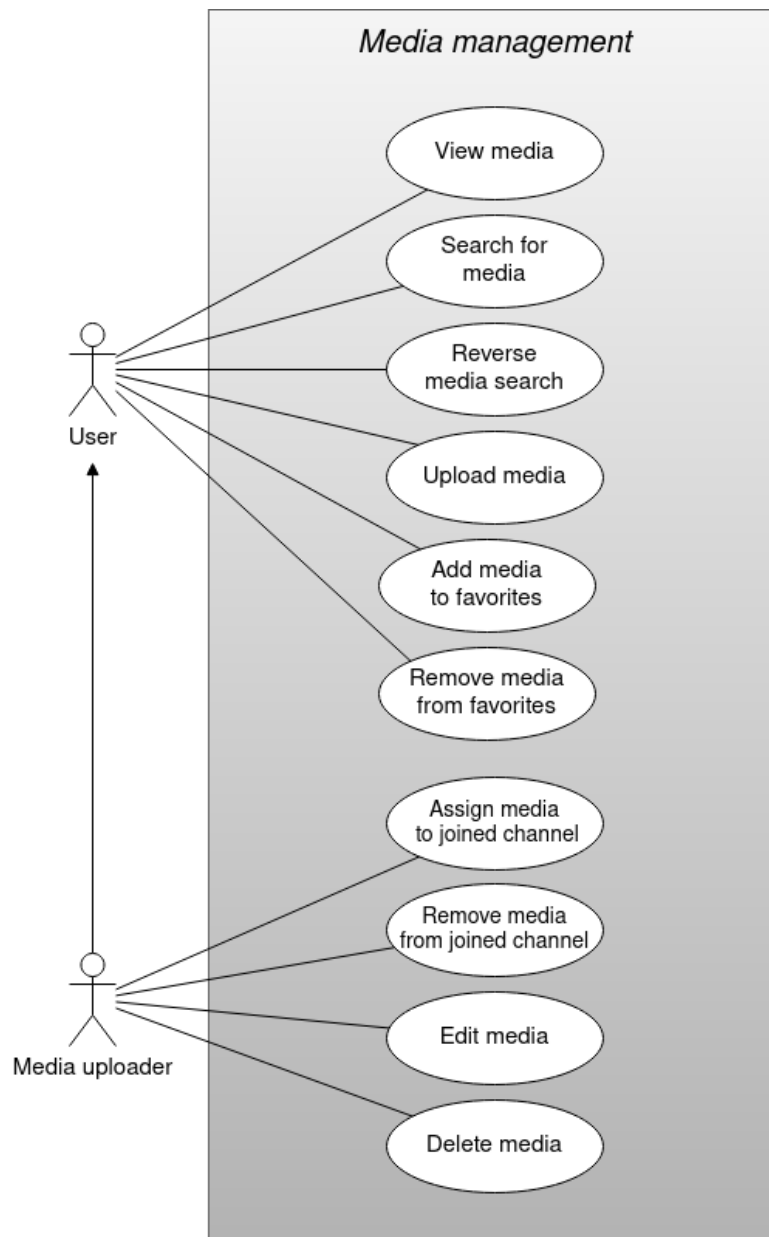


FIGURE 3: THE MEDIA MANAGEMENT USE CASES OF DEEPFAKECHAIN

2.5 MEDIA VERIFICATION USE CASES

The media verification use cases of DeepFakeChain are depicted in Figure 4. We see that all users can playback media and view their metadata, which include existing comments, annotations, and previous evaluations by humans and machines. In addition, comments are open to all users in an effort to aid evaluation. However, **verification in DeepFakeChain is achieved only through the consensus of human users and only channel members can actively participate in it**. Specifically, while the verification process is active, channel members can add annotations and evaluations of genuineness. **The verification process is initiated and finalised by the original media uploader** who has the full control over the uploaded media. Alternative procedures are discussed in section 3.

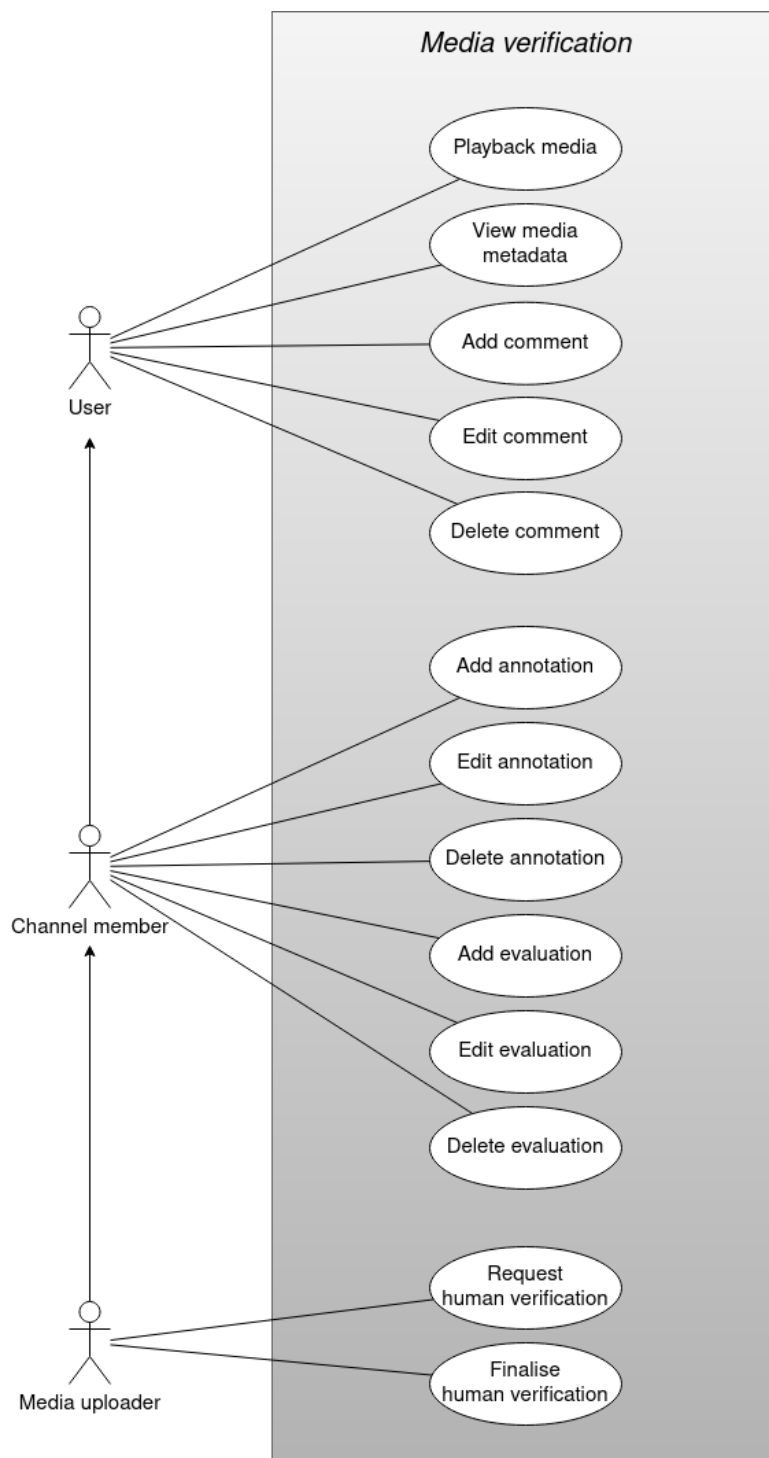


FIGURE 4: THE MEDIA VERIFICATION USE CASES OF DEEPPFAKECHAIN

2.6 SEQUENCE DIAGRAMS

We conclude the presentation of the behavioural view of the DeepFakeChain platform with sequence diagrams of the most important functions. In Figure 5, we summarise in abstract form the basic components of the DeepFakeChain system that appear in these diagrams:

- The **DFC UI** is the browser page, which the user communicates with. It relays actions to the backend server.

- The **DFC Server** is the backend server that implements the control logic of the platform. It mediates between the DFC UI and the DFC Storage.
- The **DFC Storage** is the conventional storage of the platform.
- The **Blockchain Service** is the blockchain storage on which the platform stores evidence of metadata.
- The **AI Service** performs automatic deepfake detection on media.
- The **Reverse Media Search Service** (or Near Duplicate Detection Service) searches for similar or near duplicate media in a collection of media.
- The **Consensus Service** computes the consensus score of human deepfake evaluations.

The above components will be described more concretely in the Structural View of section 3. Here, we mention that the DFC UI, Server, and Storage components form part of the core platform, while the services can potentially wrap around external applications.

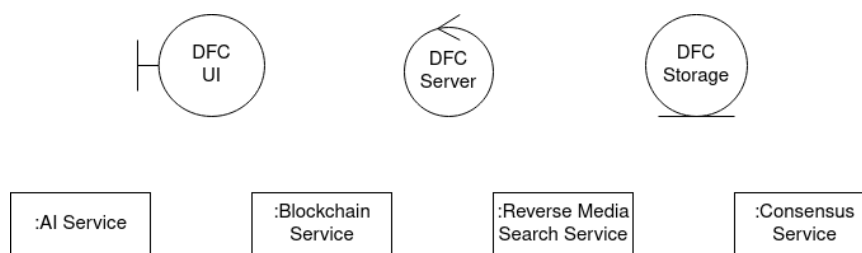


FIGURE 5: THE BASIC COMPONENTS OF DEEPFAKECHAIN

Figures 6-8 depict in compact form the basic CRUD operations on platform objects, which may refer to channels, media, user profiles, comments, annotations, and deepfake evaluations. Cases that deviate from the depicted patterns will be discussed afterwards.

The view operations of Figure 6 represent a basic read-only operation on the platform, which does not require any special privileges from the users, at least in the initial design. In more detail, the end user requests to view an object through the web interface, which is handled by the backend server in communication with the backend storage. In the case of trustworthy objects like media, comments, annotations, and evaluations, the platform can also request the hashes from the blockchain storage and verify them.

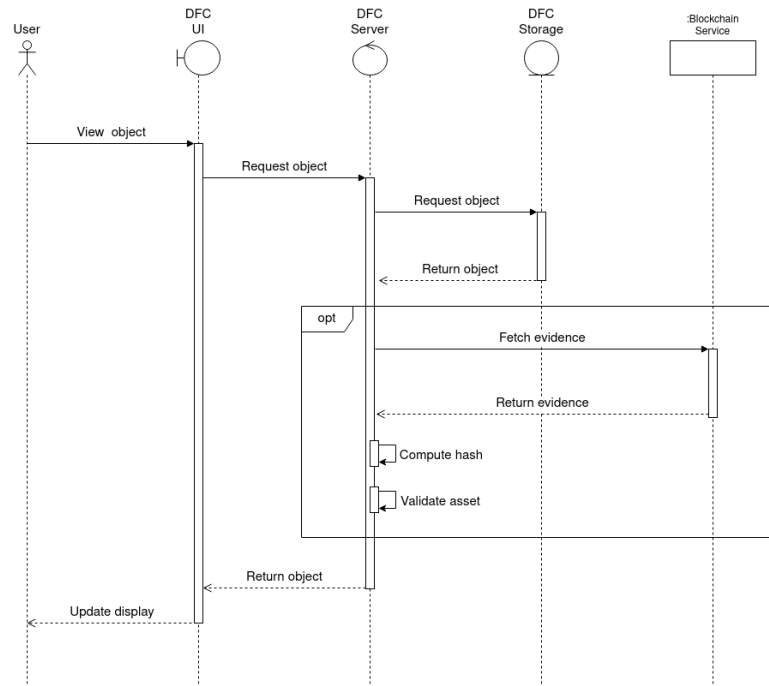


FIGURE 6: SEQUENCE DIAGRAM OF A VIEW OBJECT OPERATION

The add / edit operations of Figure 7 are distinct from the view operations of Figure 6 in that they are not permitted to all users. As described in the use cases, while all users can upload and comment on media, only channel members can add annotations and evaluations, and only channel creators and media uploaders can edit. Furthermore, the creation of a new profile is part of the registration process and the media uploading is a significantly more complex operation that will be described afterwards. With respect to the blockchain storage, notice that the communication is asynchronous as it requires validation by the blockchain's consensus algorithm. We highlight that **the edit operation does not mutate the underlying ledger but simply adds an edit transaction to it**. Edit operations are also tracked for accountability.

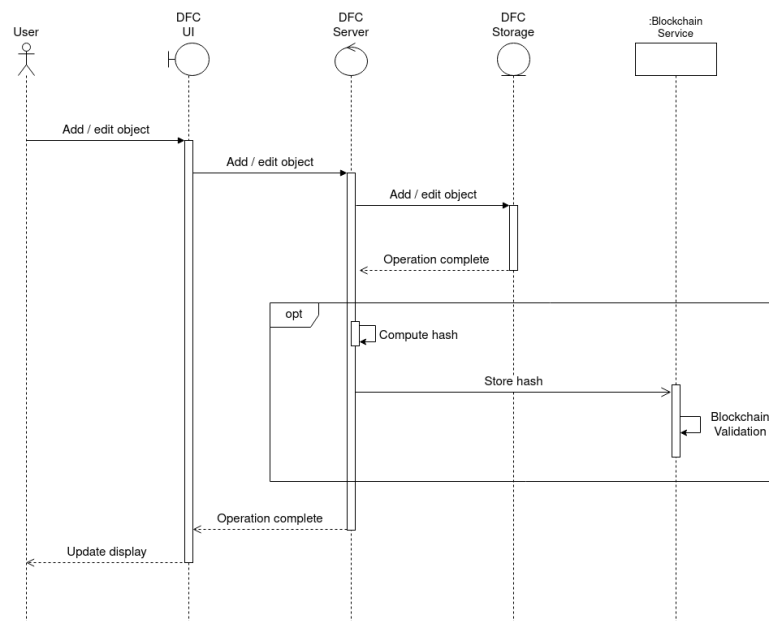


FIGURE 7: SEQUENCE DIAGRAM OF AN ADD/EDIT OBJECT OPERATION

The delete operations of Figure 8 are also write operations like the add/edit operations of Figure 7 but they require stronger privileges due to their sensitive nature. In particular, only channel creators can delete their own channels, and media uploaders their own media. This is why delete operations trigger a warning to the user by the UI, as shown in the beginning of the sequence diagram. It is important to understand that **a delete operation does not always delete the media from the platform immediately, especially in the cases of media, annotations, and comments**. This is to comply with security and law enforcement requirements, e.g. in order to support the investigation of cyberattacks or Internet crime. Of course, during this period the data is hidden from the user interface. Another difference is that the delete operation on the blockchain does not require to compute the hash value, as the stored value will be deleted with the object's key. As with edit operations, **deleting a value on the blockchain does not mutate the ledger but simply adds a delete transaction to it**.

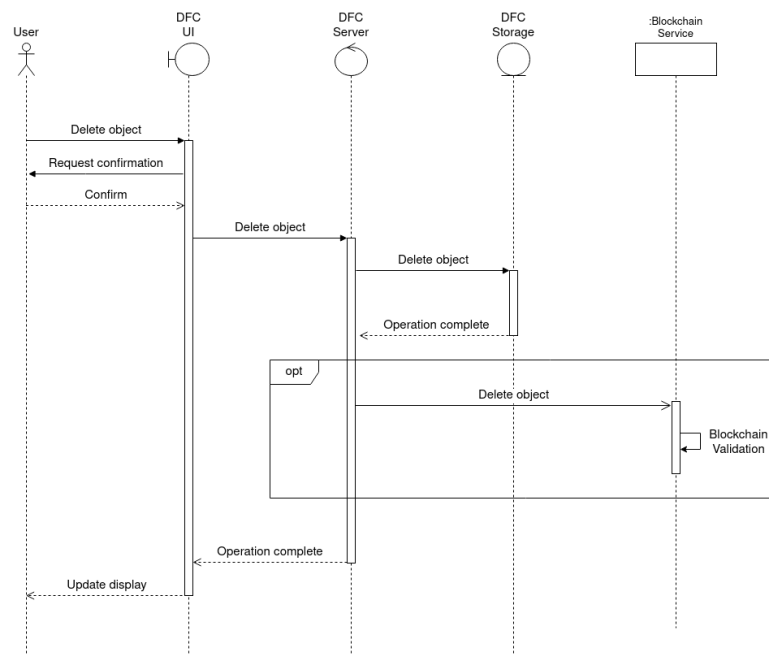


FIGURE 8: SEQUENCE DIAGRAM OF A DELETE OBJECT OPERATION

In the following, we present more complex interactions of the DeepFakeChain platform. Figure 9 depicts the media uploading process, which deviates significantly from the add operations of Figure 7. Initially, we see how users can upload media directly from their local storage or through a URL to an online source. In the latter case, the platform will request available metadata that the online source provides, e.g., YouTube metadata. After the platform receives the media file, it will request further metadata from the users (title, description, tags), and will initiate four store operations: i) store the media to the local file system, ii) index its metadata for keyword search, and iii) send the file to the reverse media search service for indexing, and iv) store the media file's hash value in the blockchain. We note that the reverse media search service maintains its own storage and is synchronised with DeepFakeChain with this process. An interesting aspect of the upload media use case is that the AI results can be optionally cached to the storage so that view requests will not take time. This does not limit the upgrading of the AI service as the system could run the AI service if the output of that particular model is not cached. This is depicted in the sequence diagram of Figure 10.

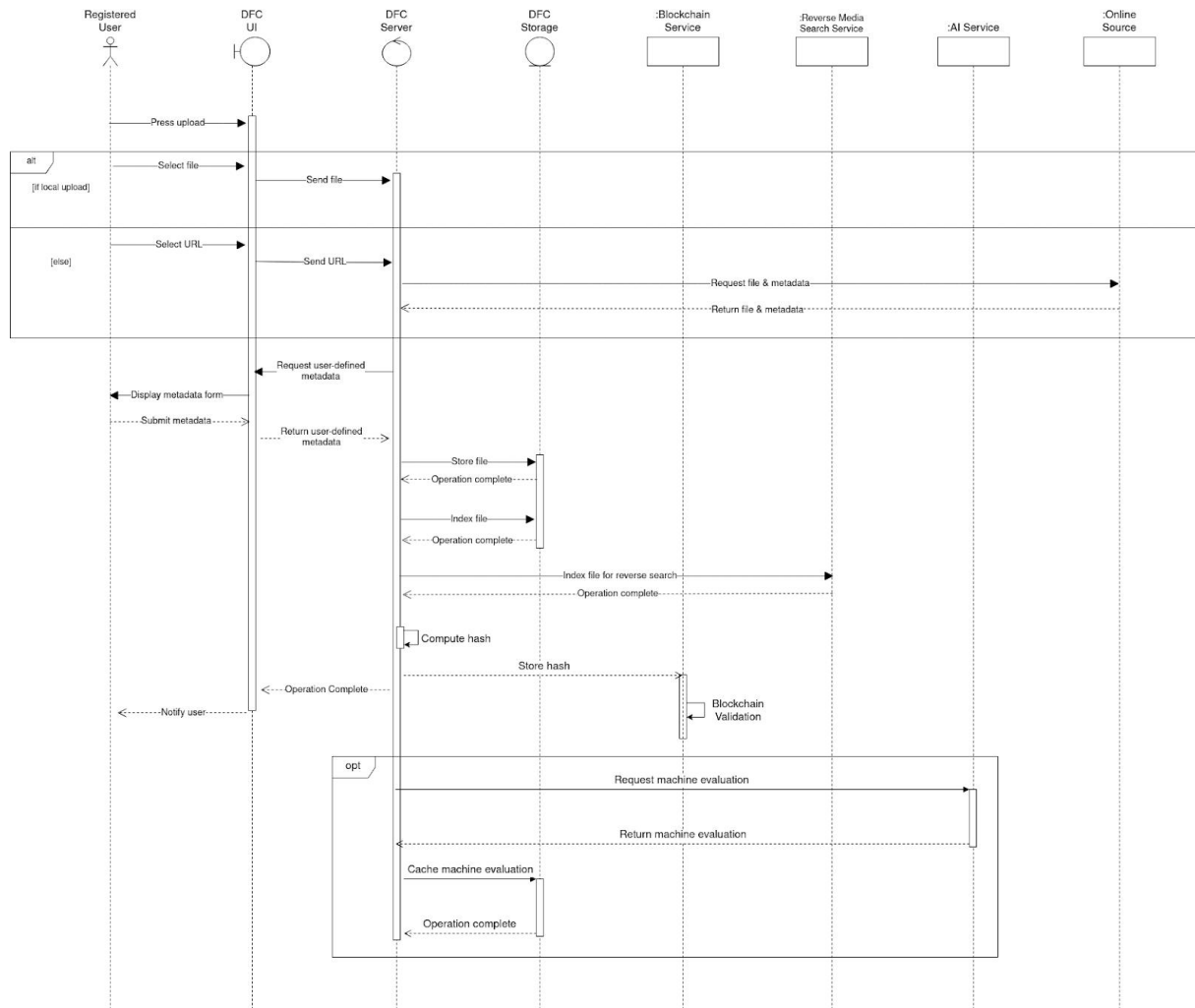


FIGURE 9: SEQUENCE DIAGRAM OF THE MEDIA UPLOAD OPERATION

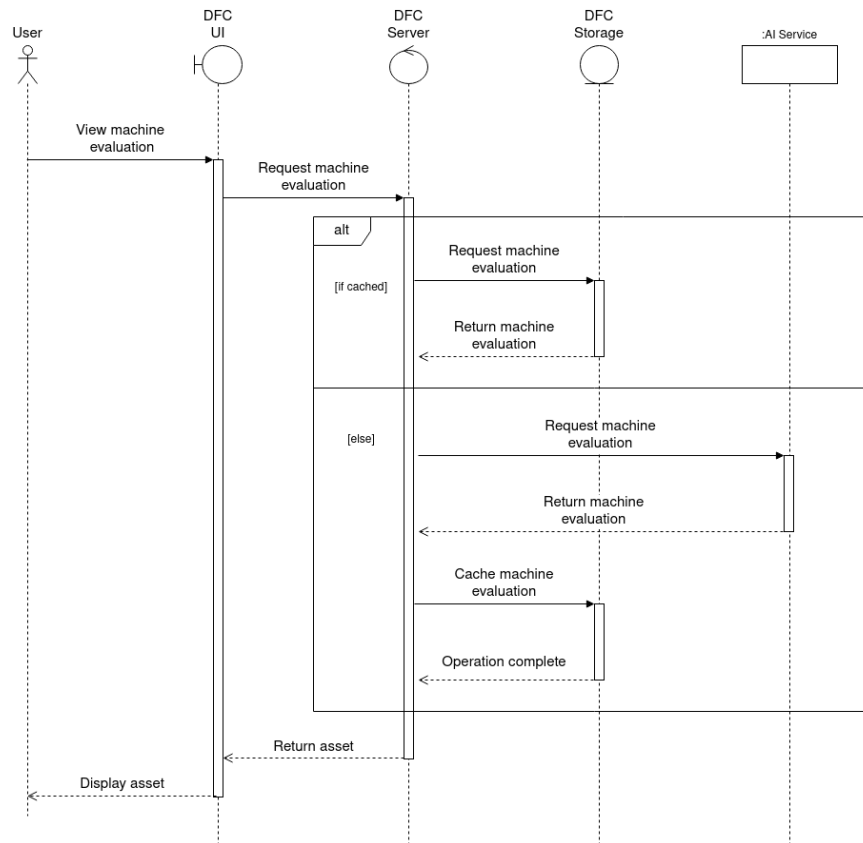


FIGURE 10: SEQUENCE DIAGRAM OF THE MACHINE EVALUATION OPERATION

Finally, the sequence diagram of Figure 11 depicts the human verification process, which is initiated by the media uploader and executed by all channel members. Once the media uploader requests human consensus on a media, the platform changes the media verification status to a pending state. During this time, all channel members are free to evaluate the veracity of the media via a 5-point Likert scale, as well as modify/remove their evaluations. This process ends by the action of the media uploader, which prompts the server to collect all evaluations and send them to the consensus module. After the consensus evaluation is derived, it is stored in both the conventional and blockchain storage.

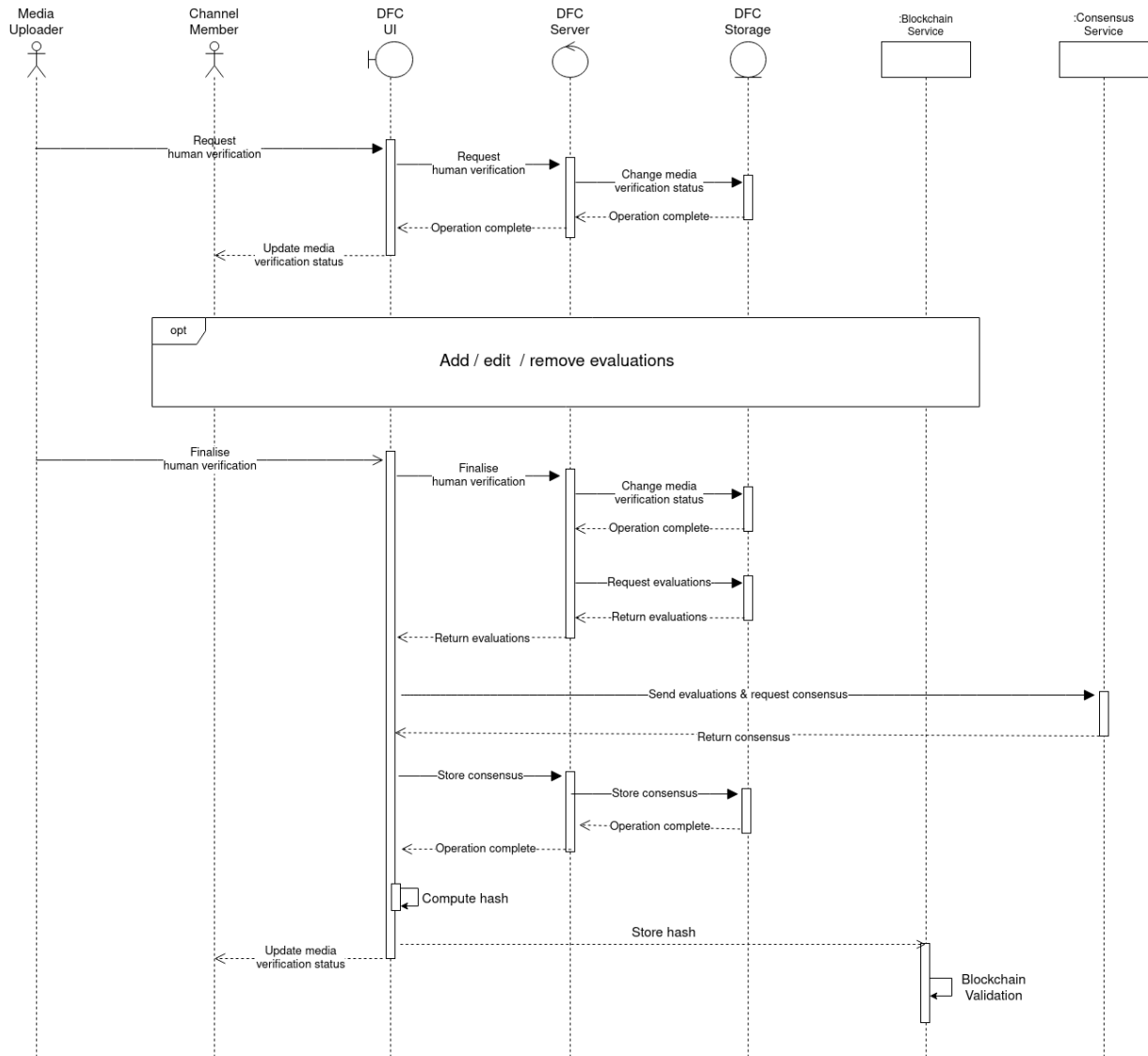


FIGURE 11: SEQUENCE DIAGRAM OF THE HUMAN VERIFICATION OPERATION

3 TECHNICAL ARCHITECTURE – STRUCTURAL VIEW

The structural view of the DeepFakeChain architecture contains the its components and their interrelations. We present an overview of the platform's architecture, followed by the UML component diagrams.

3.1 OVERVIEW

The architecture of DeepFakeChain has no major changes from TR1.1 and is presented in Figure 12. It contains the **User Layer** that communicates with the end users and the backend services, the **AI Layer** that contain the reverse media search service and the deepfake detection AI models, the **Data Layer** that stores and indexes the platform's data, and the **Blockchain Layer** that notarises the platform's metadata.

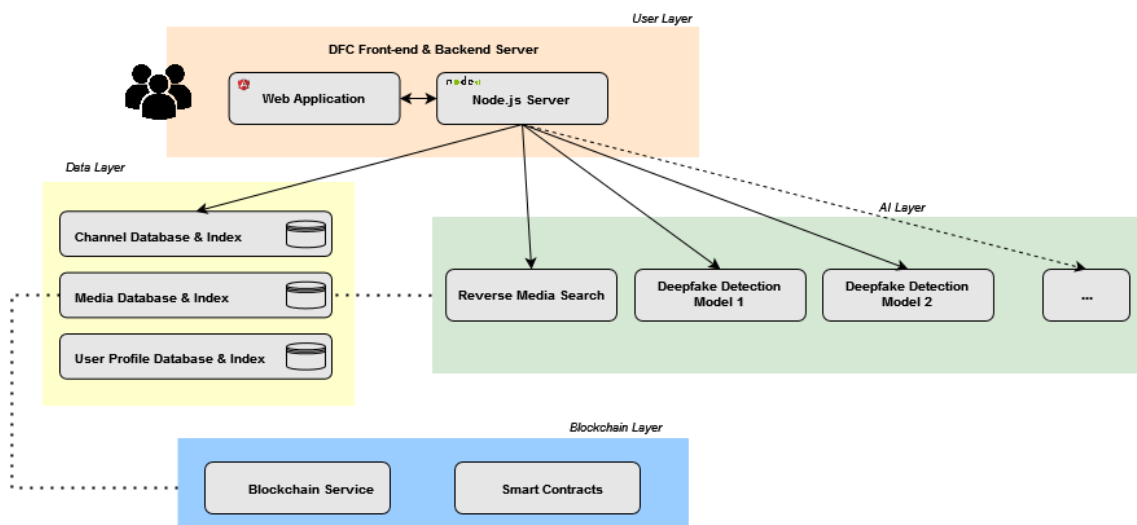


FIGURE 12: THE GENERAL STRUCTURE OF DEEPFAKECHAIN

In more technical detail:

- The **web application** is part of the User Layer and provides secure access to DeepFakeChain's services. The implementation will be based on the SmartViz tool by Zelus, which uses the Angular framework. We also consider using the Keycloak service¹ for user authentication.
- The **Node.js server** is also part of the User Layer and orchestrates all the backend services. It will be written in the *Node.js Express* web development framework².
- The **reverse search engine** is part of the AI Layer and can be customised to support similar or near duplicate media search. It is available by CERTH.

¹ <https://www.keycloak.org/>

² <http://expressjs.com/>

- The **deepfake detection models** are part of the AI Layer and can detect deepfake media. The AI models will be developed with the Python PyTorch framework³.
- The **databases** of the Data Layer represent the off-chain storage of our system that stores the media-related, channel-related, and user profile data. The video files will be stored directly to the file system while the queryable metadata will be stored at separate *Elasticsearch*⁴ indexes. User profile data will also be encrypted for privacy.
- The **blockchain services** belong to the Blockchain Layer and comprise the smart contracts that govern the blockchain storage. We have chosen Alastria for this storage and have access to a pre-setup node on Alastria's *Red-B* network⁵.

3.2 COMPONENT DIAGRAMS

Our technical architecture is further clarified in the UML component diagram of Figure 13 below, which depicts the system’s components and their dependencies. Figure 14 also presents the architecture of DeepFakeChain as deployment artifacts to deployment targets. In particular, the components of DeepFakeChain will be deployed in distinct Docker containers that will communicate through appropriate TCP ports. The main object that will be shared between the containers is the media, which is the core artifact of the deployment. Other artifacts include the annotations, comments, and evaluations generated by the users, which will be shared with the Blockchain services.

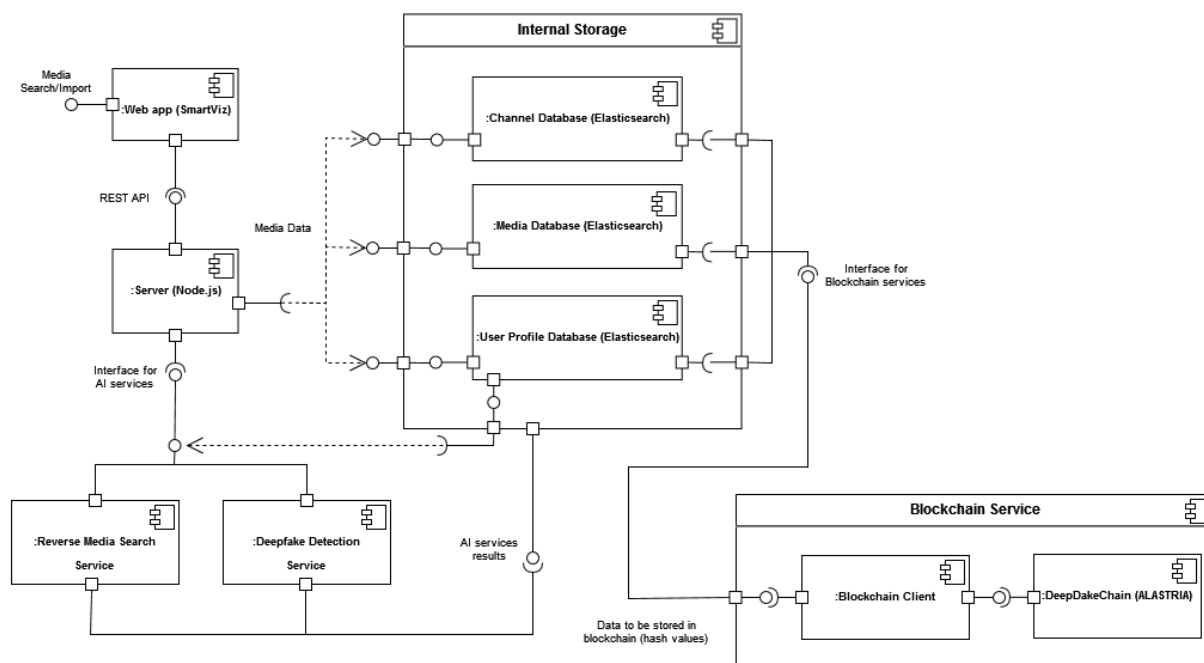


FIGURE 13: THE COMPONENT DIAGRAM OF DEEPEAKECHAIN

³ <https://pytorch.org/>

⁴ <https://www.elastic.co/elasticsearch/>

⁵ <https://alastria.io/en/home/>

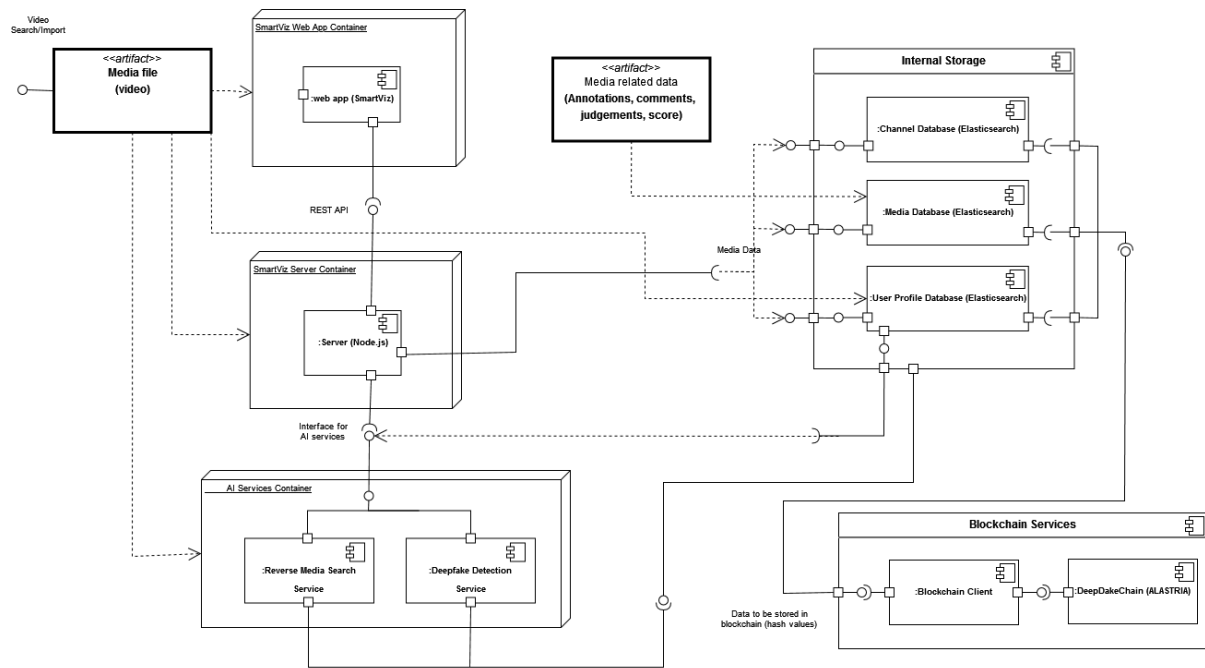


FIGURE 14: THE DEPLOYMENT DIAGRAM OF DEEPFAKECHAIN

4 TRUST & SECURITY FRAMEWORK

In the following subsections, we describe the trust and the overall security frameworks of DeepFakeChain.

4.1 TRUST FRAMEWORK

There are three types of actors in DeepFakeChain, the users, the algorithms, and the platform itself. All of these face distinct challenges with respect to trust: i) the users must be trusted to provide reliable metadata to the platform such as comments, annotations, and evaluations, ii) the algorithms must be trusted to provide reliable evaluations of a media being a deepfake, and iii) the platform must be trusted not to tamper with the uploaded data or interfere in any way with the verification process. In the following, we elaborate on how we address these issues.

4.1.1 USER TRUST

As the platform targets professional journalists, at least in the initial design, we expect a certain level of a priori trust. This will be established by the screening of the users and their affiliations before registration by the platform's administrators, or a more transparent procedure in the future. On the other hand, this measure cannot fully preclude dishonest or incompetent behaviour, especially in the light of future applications for our platform. This is why the platform's decisions will be ratified through consensus.

In conventional fact-checking platforms such as Meedan's Check, the fact checking process for an item is delegated to an evaluating user, typically the item's uploader, who is responsible for orchestrating the verification effort and delivering the final decision. Other users can collaborate but only through providing comments and evidence for the evaluator. This model simplifies control but fails if an evaluator is malicious, potentially colluding with the platform, and delivers a false evaluation for the item. While the metadata leaves a public trail to justify the decision, this may be overlooked by viewers, spreading false information in the process. To address this issue, we propose human consensus for a media to reach a definitive verification status. In more detail, the verification process will begin once the media uploader requests consensus from the channel's members, who will be able to annotate the media and evaluate its veracity on a 5-star Likert scale. The 5-star scale was selected because it represents the "certain fake", "certain genuine" and "uncertain" cases, as well as two intermediate evaluations. Users can freely edit their evaluations while the verification process is active and as more evidence accumulates but a consensus decision is taken from the existing evaluations once the process is finalised. In our current design, the choice for finalising the verification lies on the media uploader, however alternatives include setting a fixed time period for verification and requiring the approval of multiple users to finalise the verification.

While the majority consensus is vulnerable to a 51% attack, especially in channels with few members, we argue that this is not a big issue in a platform with controlled user registration. The same argument applies to colluding attacks of multiple users and Sybil attacks of fake user accounts. In Phase 2, we consider adding the feature of **challenging** the established evaluation and weighing the user's votes with their reputation, which could be calculated by the deviations of the users' decisions from the consensus opinion. In addition, the users' inputs such as comments and annotations could be enhanced with up/down voting and flag options, to identify users that employ obvious manipulation tactics.

4.1.2 ALGORITHM TRUST

We assume that DeepFakeChain's deepfake detection algorithms can be unreliable but not misleading. This means that we expect a high chance of identifying a known generation method while tolerating a lower performance in mistaking fake media generated by unseen methods as genuine (false negative). To ensure that an algorithm is effective in its domain, it should be always evaluated on relevant datasets before integration in the platform. This information could be presented to the platform's users through a model card, as has been done in a previous work by CERTH [1].

In DeepFakeChain, we assume a human-driven collaboration model between AI and humans, in which the AI algorithms are provided as helping tools to the users. In particular, each time a media is uploaded to the platform, it will be analysed by the algorithmic layer and the results will be presented to the users to facilitate their evaluations. We do not permit machine evaluations to directly influence because currently AI algorithms are not mature enough for this approach. Human professionals may also not appreciate a design which reduces their control.

The inclusion of the algorithms' evaluations pose interesting questions related to the interface design of the platform. As algorithmic evaluations cannot be completely reliable at the moment, the interface should not present them with undue confidence. In other words, a given algorithmic evaluation, especially of a media being genuine, should not lull the user into believing it is correct. Towards this goal, instead of presenting the algorithm's confidence as a probability/percentage, we consider the statements "some / no indication of this media being a deepfake were found" along with a panel offering explanations. Taking this approach to the extreme, we could **blind** the algorithmic decisions to the voting users to avoid biasing them and compare the human and machine results after voting. This approach defeats the whole purpose of having an algorithmic tool but it may be appropriate in specific circumstances. Planned interviews with end users will help us decide the usefulness of these features.

4.1.3 PLATFORM TRUST

Although the DeepFakeChain is a centralised platform, we note that it is not in its best interest to interfere with the evaluation process, as this would critically harm its reputation and drive its users away. This applies to overt cases of tampering such as manipulating data related to freshly uploaded media that are processed by users. Nevertheless, the platform could conceivably tamper with past evidence in order to mislead users. To mitigate this risk, the blockchain storage will be used to store proofs of authenticity (hashes) of uploaded information, specifically, the media file itself, the comments, the annotations, and the final evaluation. Yet, formally proving that the presented information is tamperproof is a hard problem since the platform should not be trusted to check the hashes itself. Evaluating the platform through an independent interface is similarly ineffective, as the media's information such as the media's ID and metadata should be downloaded from the platform and the platform could easily store a media with a new ID with the tampered information. We believe that the best way to check the platform is to allow regular audits, which would compare the conventional storage of the platform against the notarised transactions stored in the blockchain.

4.2 SECURITY FRAMEWORK

Multiple factors have been considered for the overall security of the DeepFakeChain platform. One of the most important aspects is the authentication mechanism through the Keycloak system. Keycloak secures our web application using single sign-on with identity and access management. Using Keycloak we can defend our system and our users over the most common types of attacks since the web page will be served under HTTPS and all the communications will be encrypted. This feature protects the users over **man-in-the-middle** attacks and any kind of **eavesdropping**, i.e., real-time interception of private communications. Eavesdropping

attacks in particular are very common and easy to execute by experienced individuals, especially if public networks or networks with low security are used. Keycloak also protects against basic forms of **denial-of-service** (DoS) attacks. This is done by specifying a minimum interval between two requests to Keycloak, which will prevent users/bots from spamming our infrastructure.

DeepFakeChain will further take active measures against **phishing**, which can be one of the most catastrophic and widely used types of attacks. For its mitigation, we will include in our website and continually update easy to read information and instructions on the newest types of phishing attacks, how social engineering attacks work, and simple steps to protect their network activity and personal data. In cases of observed phishing attacks, all DeepFakeChain users will be notified via their registration emails.

Finally, the blockchain layer of DeepFakeChain requires its own layer of security. For our solution, we have chosen Alastria's Red-B, which both suits our requirements and is highly secure, following modern security guidelines. Through our investigation and experimentation with the blockchain technologies, we are further aware and continuously monitor the most critical blockchain attacks. The latter belong broadly to two categories: i) **network attacks** on the Blockchain peer-to-peer (P2P) network, and ii) **platform attacks** on the platforms that support the blockchain, either internally in the entire blockchain-based platform or externally when interfacing with other websites or other platforms.

Regarding the network attacks, our periodically updated catalogue includes:

- Sybil attacks,
- double spending or 51 percent attacks,
- miner ransomware attacks,
- eclipse attacks,
- routing attacks.

Regarding the platform attacks, our periodically updated catalogue includes:

- credential attacks,
- dependency backdoor attacks,
- DoS and distributed DoS (DdoS) attacks,
- faulty code.

5 TECHNICAL REQUIREMENTS

Based on the solution design and the final architecture of the platform, we present our technical requirements in this section.

5.1 DEVELOPMENT REQUIREMENTS

TABLE 1: DEVELOPMENT REQUIREMENTS

ID	Requirement
DEV 1	The system shall be developed in the microservice format. Docker containers shall be used for modularity and extensibility.
DEV 2	The web application shall be dynamic. The Angular and the Node.js Express frameworks shall be considered for the front-end and the backend design respectively for the initial design. Both of them are JavaScript-based for uniformity.
DEV 3	The AI deepfake detection services shall be developed for production-grade deployment. The Torchscript format of Python's PyTorch library shall be considered for the initial design. Other approaches shall be possible due to the dockerisation of the services.
DEV 4	The off-chain metadata shall be indexed in order to be searchable. The Elasticsearch search engine shall be considered for storage and indexing. The media files shall be stored directly to the file system.
DEV 5	The blockchain services shall use Alastria's Red-B network, based on Hyperledger Besu.

5.2 DEPLOYMENT REQUIREMENTS

TABLE 2: DEPLOYMENT REQUIREMENTS

ID	Requirement
DEP 1	The Node.js server shall use at least 16 GB of RAM and 1 core of CPU operation. This is due to Node.js' single-threaded operation.
DEP 2	The off-chain storage shall use at least 1 TB of data.
DEP 3	The AI services shall use at least 8 GB of GPU RAM.
DEP 4	The blockchain services shall use a pre-setup node in Alastria's Red-B. A Linux server instance with 2 cores, 8 GB RAM, and 64 GB storage has been reserved.

5.3 INTERFACE REQUIREMENTS

TABLE 3: INTERFACE REQUIREMENTS

ID	Requirement
INT 1	The platform shall be accessed as a website via browser.
INT 2	The browser compatibility shall be validated for the latest stable versions of the Microsoft Edge, Mozilla Firefox, and Google Chrome browsers.
INT 3	The interface design shall be reactive in order to appear correctly in handheld devices.
INT 4	The reactive appearance shall be validated on Android devices.
INT 5	The interface shall be in English, at least for Phase 1.

5.4 SECURITY REQUIREMENTS

TABLE 3: SECURITY REQUIREMENTS

ID	Requirement
SEC 1	The web application will communicate with the user's browsers via HTTPS.
SEC 2	The DeepFakeChain Docker components will communicate via HTTPS.
SEC 3	The application shall encrypt user profile data before indexing them in Elasticsearch via AES-128 or stronger.
SEC 4	The system's authentication shall use the Keycloak service.

5.5 PERFORMANCE REQUIREMENTS

TABLE 4: PERFORMANCE REQUIREMENTS

ID	Requirement
PER 1	The home page shall be loaded in less than 2 seconds.
PER 2	The media shall be uploaded in less than 3 seconds (for a video of standard length 1 minute or less).
PER 3	The search of media with keyword queries shall be completed within 2 seconds.
PER 4	The reverse search of media shall be completed within 3 seconds (for a video of standard length 1 minute or less).
PER 5	The automatic evaluation of deepfake shall be completed within 10 seconds (for a video of standard length 1 minute or less).
PER 6	The notarisation of DeepFakeChain's metadata in the blockchain storage shall be completed within 1 second.

6 DATA MODEL

Figure 15 depicts the data model of the DeepFakeChain platform. It contains all the main concepts that have been elaborated in the TR1.2 deliverable. In particular, we can see:

- The user-related data, which include the users' real names, affiliations, contact information, personal descriptions, and their roles in the platform.
- The channel-related data, which include descriptions, the uploaded multimedia, and their connection to the users via ownership and membership relations.
- The media-related data, which contain descriptions, tags, comments, annotations, evaluations of genuineness, as well as metadata retrieved by the source platform (e.g., YouTube).
- The evidence of the uploaded content, which are the uploaded multimedia, the comments, the annotations, and the evaluations.
- The AI model-related data, which contain a model type (e.g., ResNet, EfficientNet, Visual Transformer, etc) and a description. The description can be further extended to a full model card.

We note that the annotations differ from comments because they are related to a specific spatiotemporal window and they do not represent a dialogue. In addition, while users can create multiple comments and annotations, they can only offer a single evaluation which can be edited at a later time. The same applies for machine-originated evaluation.

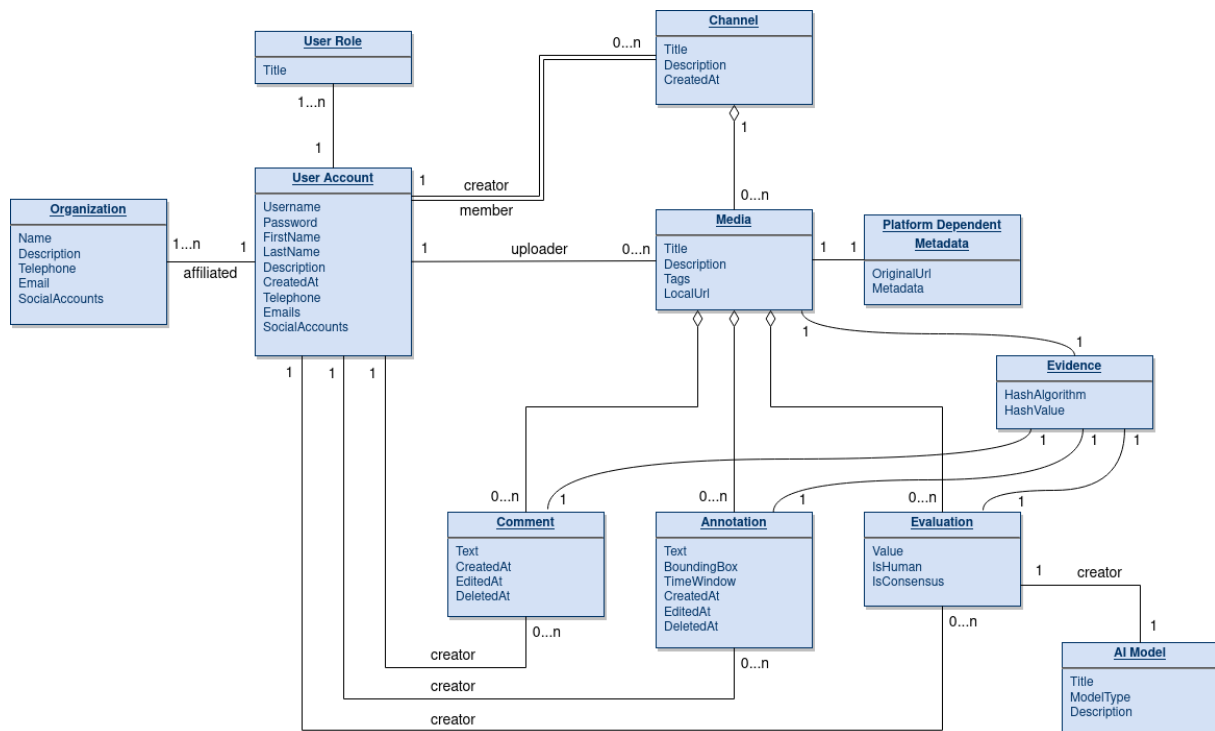


FIGURE 15: THE DATA MODEL OF DEEPFAKECHAIN

7 DATA PROTECTION & PRIVACY

With respect to the Data Model diagram of Section 6, we note that user-, channel-, media-, and AI model-related information will be stored in the conventional storage of the DeepFakeChain platform, while the hash-generated evidence will be stored on-chain. Since the uploaded media are not expected to be original and come from third sources, proper attribution will be enforced. In particular, for media uploaded through URL, the source will be retrieved automatically. For media uploaded from local storage, the platform will require the media uploader to describe the source.

Our platform will also fully comply with GDPR regulation. In particular, all users will be informed on the type of personal data that will be stored in the platform and its intended uses through the terms-of-service (ToS) document. This document will be written in plain language and will be presented to the users upon registration. We stress here that due to the research-oriented focus of DeepFakeChain, at least in this phase, the personal data collected by the platform will be minimal and will only be used for authentication, accountability, and trust-building among the users.

In addition to the above, we will appoint a member of the DeepChainChain project as a data protection officer (DPO) who will be responsible for handling personal data-related issues and requests. In detail, the DPO will be available to inform users about their personal data in addition to the ToS (right to be informed), to return the user data in a format that will facilitate its usage by the users in other contexts (right to access and data portability), to edit / correct data and delete it from the platform (right to rectification and erasure), to restrict its processing in case of objection (right to object), and notify users in case of data breach (right to be informed). All these operations will be executed by the DPO within a reasonable time frame and the limits imposed by the Law.

Finally, regarding privacy, all personal data of user profiles will be encrypted before storing them in the off-chain storage to protect them in case of a data breach. Since DeepFakeChain targets professionals and relies on trust, the users will be required to input their real personal identifying information, which will be publicly visible to all other users. This requirement and its necessity will be explained to the users via the ToS document. Considering future use cases of the platform in less formal contexts, e.g., moderation of social networks, only the usernames of the users could be visible inside the platform.

8 APPLICABLE STANDARDS

The main standards used in the development of the DeepFakeChain platform are shown in the following table.

TABLE 5: STANDARDS FOR THE DEVELOPMENT OF DEEPFAKECHAIN

Category	Standards	Uses
Communication Standards	HTTP/HTTPS	Communication with users and among DeepFakeChain's Docker components.
File Format Standards	JSON, Protocol buffer	Data formats for metadata transfer.
Cryptographic Standards	AES-128, SHA2/3	User profile encryption and cryptographic hashes.
Evaluation Standards	Likert 5 scale, Balanced Accuracy, AUC	Evaluation of deepfake media.
Privacy Standards	GDPR 2016/679	Compliance with European data protection law.

9 INITIAL PROTOTYPE INFORMATION

The following sections describe the components that are available for DeepFakeChain's initial prototype.

9.1 DEEPFAKECHAIN MOCKUP INTERFACE

A mockup for DeepFakeChain's initial web interface is shown in the pictures below. Figures 16 and 17 depict the registration and login screens respectively. According to the, the users are prompted to declare their affiliations, which need to be verified by the DeepFakeChain team to approve registration.

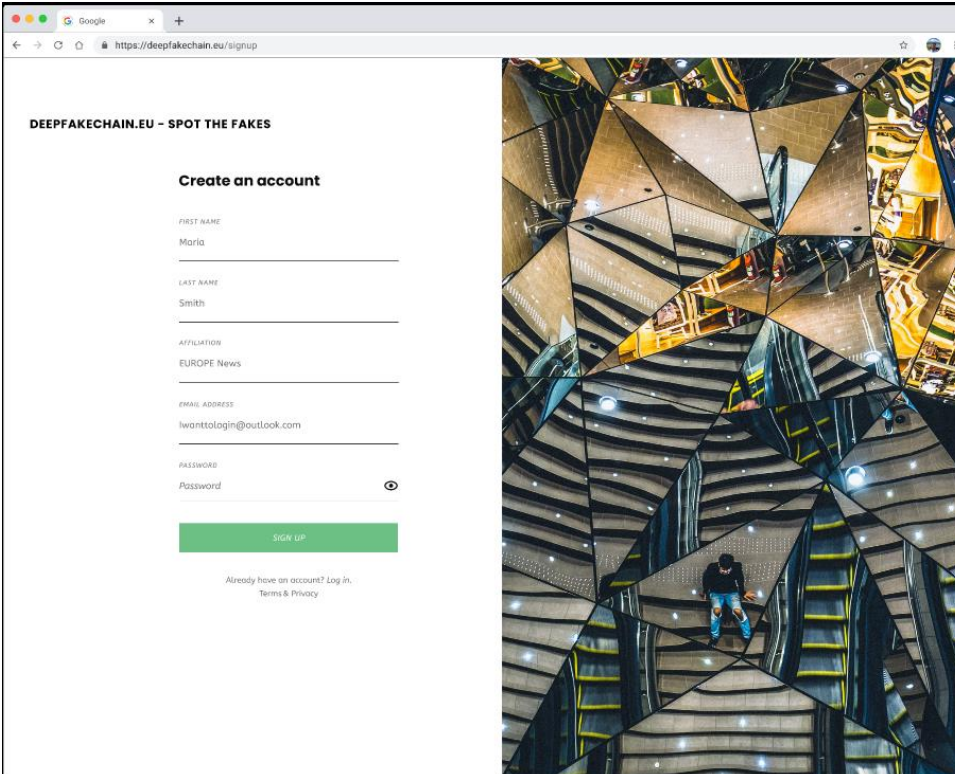


FIGURE 16: THE REGISTRATION PAGE OF THE DEEPFAKECHAIN MOCKUP

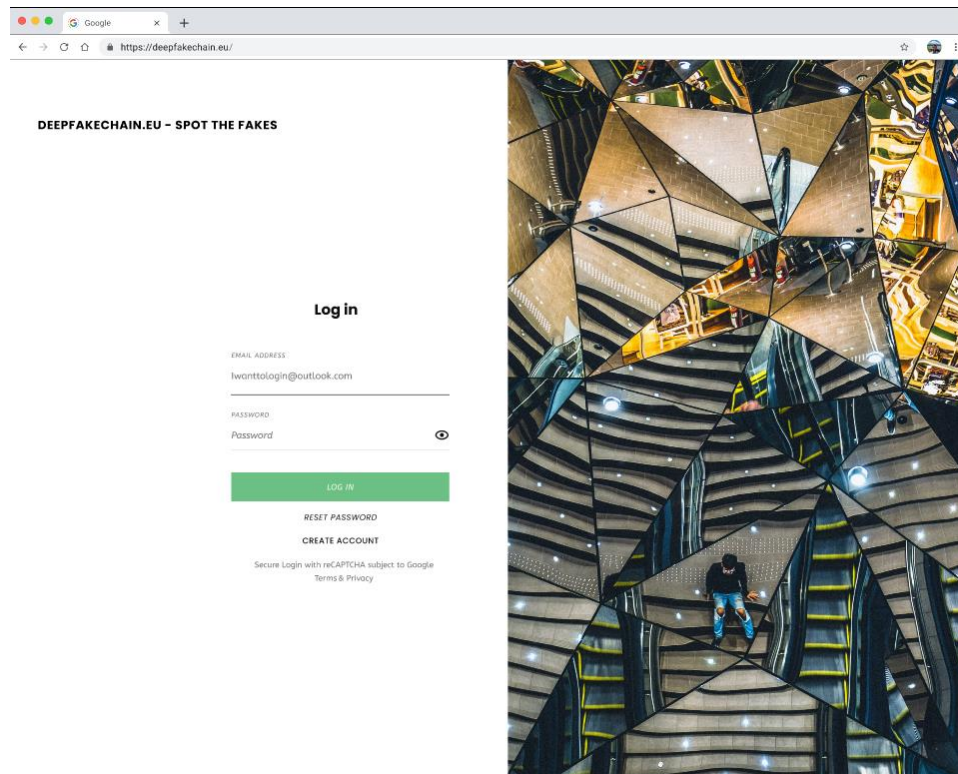


FIGURE 17: THE LOGIN PAGE OF THE DEEPFAKECHAIN MOCKUP

After login, from the landing page, the users can search for existing media based on available filters (Figure 18) or upload new media through local storage or a URL (Figure 19).

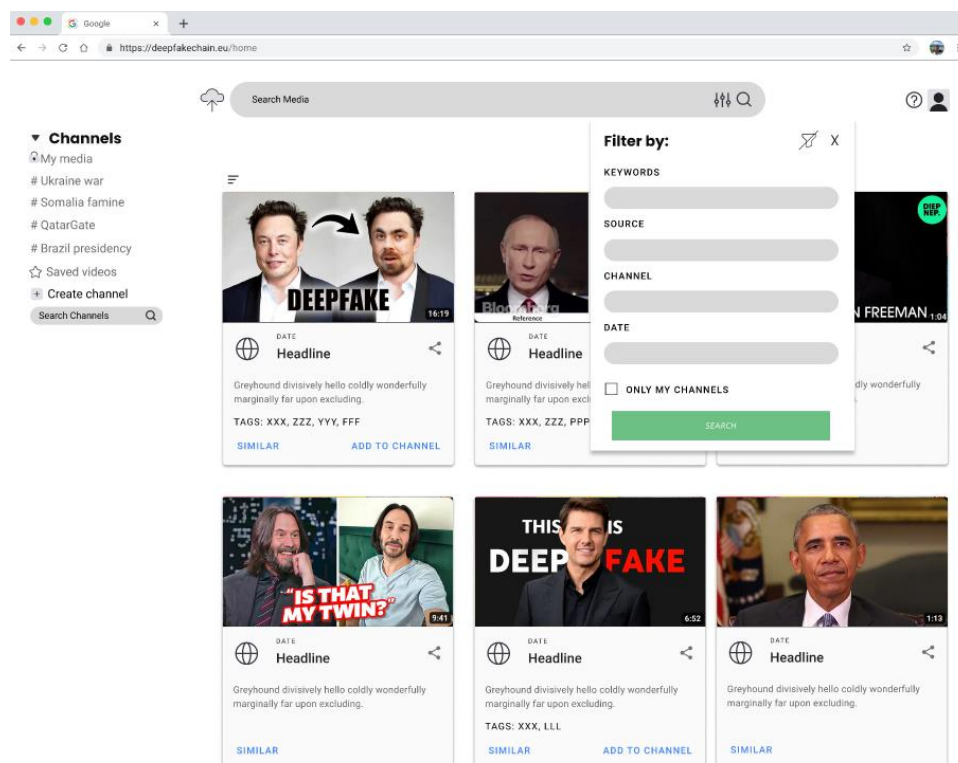


FIGURE 18: SEARCH MEDIA FUNCTIONALITY IN THE DEEPFAKECHAIN MOCKUP

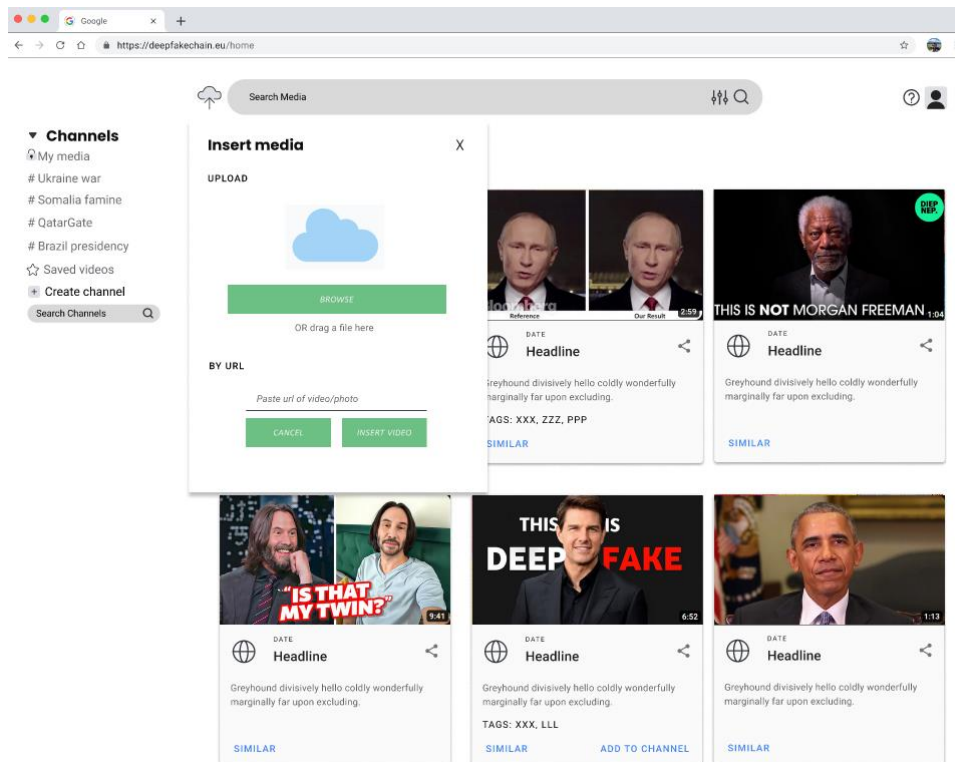


FIGURE 19: UPLOAD MEDIA FUNCTIONALITY IN THE DEEPFAKECHAIN MOCKUP

Additionally, the users can view their collaboration channels and search for new ones from the left vertical menu. After search, a list of relevant channels is returned in the main screen offering a brief description and the option to join them (Figure 20).

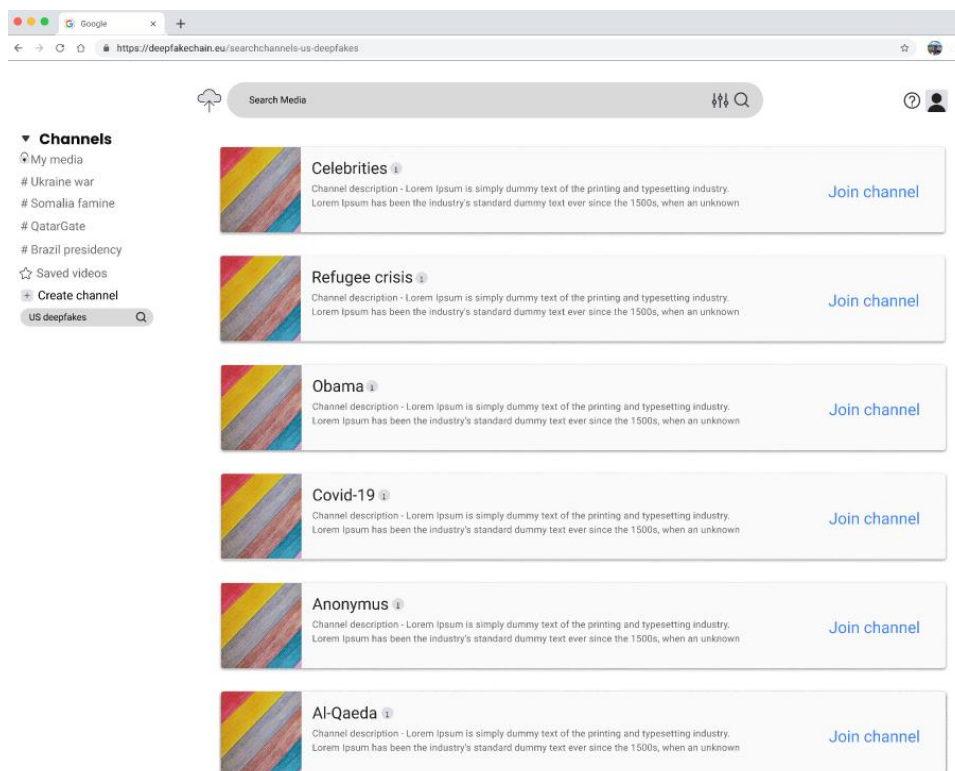


FIGURE 20: SEARCH CHANNEL FUNCTIONALITY IN THE DEEPFAKECHAIN MOCKUP

When the users select one of their channels, they can view its associated media and with further information such as the number of members and a description (Figures 21 and 22)

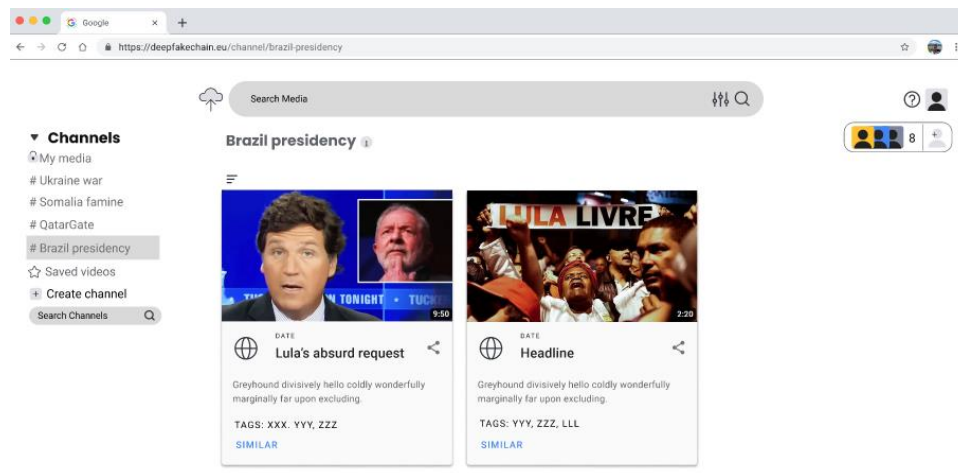


FIGURE 21: THE CHANNEL PAGE OF THE DEEPFAKECHAIN MOCKUP

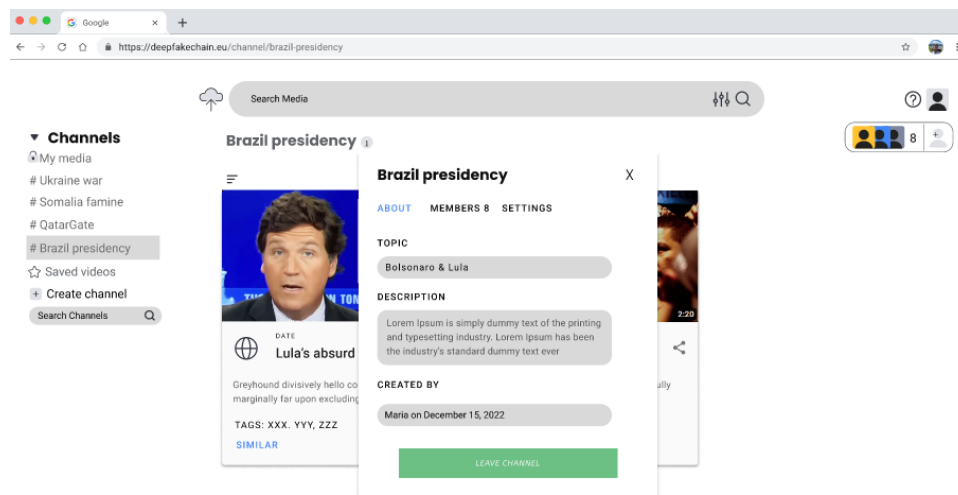


FIGURE 22: VIEWING CHANNEL DETAILS IN THE DEEPFAKECHAIN MOCKUP

From the channel's panel, the users can also search for similar or near duplicate media that are not necessarily associated with this channel (Figure 23).

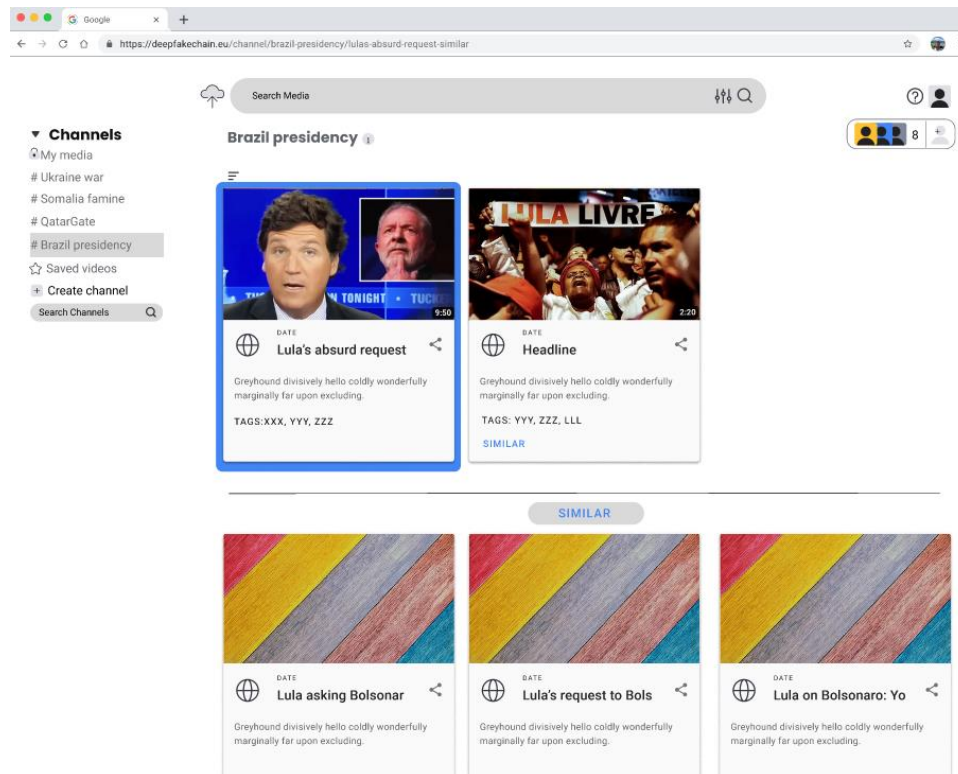


FIGURE 23: REVERSE MEDIA SEARCH IN THE DEEPFAKECHAIN MOCKUP

When the users select a media item, they can view its metadata (Figure 24). This panel will display the results of the AI deepfake detection models along with automatically generated explanations. Media uploaders can also request human evaluation from the channel's members.

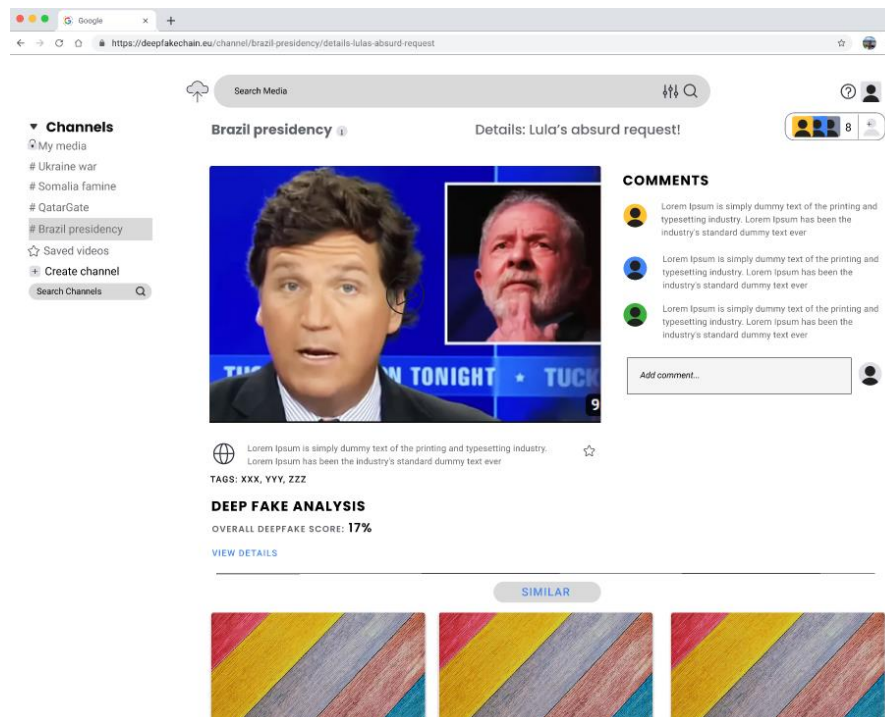


FIGURE 24: THE MEDIA PAGE OF THE DEEPFAKECHAIN MOCKUP

9.2 REVERSE MEDIA SEARCH

The reverse media or near duplicate search service has been developed by CERTH and an instance of it is already deployed at CERTH premises. The service is accessible through a REST API, which is formally specified with the Swagger/OpenAPI specification⁶, and provides indexing and search methods.

Regarding indexing, the service supports the creation of *collections*, i.e., searchable indexes where both images and videos can be uploaded by providing URLs. Internally, the service transforms the multimedia to appropriate numerical (feature) vectors and stores them locally in a format that facilitates search. We highlight here that the storage of the service is separate from DeepFakeChain's conventional and blockchain storage but it is synchronised every time new multimedia are uploaded on the platform.

Regarding search, the service returns indexed items that are similar to given multimedia, which act as *queries*. The queries can be in the form of multimedia URLs, IDs of indexed multimedia, or even raw feature vectors. The latter two options are clearly faster since they avoid downloading and encoding a new media file. Once a query is received, the service returns the indexed item with the nearest vector representation. The level of similarity can be configured by specifying a similarity threshold along with the query, where a higher similarity threshold implies multimedia that are closer to being identical. Besides the standard image-by-image and video-by-video retrieval, the service can retrieve videos given image-based queries, images given video-based queries, videos with nearly duplicate shots, and videos with similar audio

⁶ <https://mever.iti.gr/ndd/docs/v3/swagger/>

10 METRICS & PLANNED TESTS

This section presents the metrics to evaluate DeepFakeChain's performance and the tests that will be planned to validate them. These metrics are consistent with and refine the KPIs of TR1.1. Please note that these metrics and tests concern only the Phase 1 of the project. The metrics and the tests in the following tests correspond one-to-one with each other.

TABLE 6: EVALUATION METRICS FOR DEEPFAKECHAIN

ID	Metric	Target
1	Number of stored media	> 100
2	Loading time for home page	< 2 sec
3	Uploading time for media	< 3 sec for videos < 1 min
4	Response time for keyword queries for media	< 2 sec
5	Response time for reverse queries for media	< 3 sec for videos < 1 min
6	Response time for AI evaluation of media	< 10 sec for videos < 1 min
7	Response time for blockchain storage	< 1 sec
8	Balanced accuracy of AI deepfake evaluation	> 75%

TABLE 7: PLANNED TESTS FOR THE EVALUATION OF DEEPFAKECHAIN

Metric ID	Test
1	Upload 100 videos to the platform during an internal testing session.
2	Compute the loading time of the home page during an internal testing session.
3	Compute the average delay for media uploading during an internal testing session.
4	Compute the average response time for keyword search for media during an internal testing session.
5	Compute the average response time for reverse search for media during an internal testing session.
6	Compute the average response time for the AI evaluation of the ingested videos during an internal testing session.
7	Compute the average response time for the notarisation of DeepFakeChain's metadata during an internal testing session.
8	Compute the balanced accuracy on the test set of the DFDC dataset during an internal testing session.

11 TECHNICAL INNOVATION

The main technical innovations of DeepFakeChain are related to the deepfake evaluation of online multimedia and the usage of blockchain technology to enhance their trustworthiness.

Regarding the deepfake evaluation, the platform will offer access to new algorithmic services that generalise better than the state-of-the-art and explain their limitations to the users. Combined with careful UI design, the explainability features, as the ones described in TR1.2, will greatly enhance the reliability of the AI tools, as they will offer actionable evaluations to the users without biasing them. In addition, the platform will support a human-oriented model of human-machine collaboration where the human users drive the decision making procedure and the AI tools have an advisory role. This is in line with the human-centric nature of NGI and respects the agency of professional human users. It is worth noting that existing fact-checking platforms typically assign the verification task to a single user, limiting collaboration to the input of comments and annotations. Considering the flat user hierarchy of DeepFakeChain and the potential of malpractice by single users, DeepFakeChain introduces a consensus process for the evaluation of deepfakes, further enhancing its trustworthiness.

Regarding the usage of blockchain technology, the platform will notarise all the uploaded content and the evaluation outputs to an external blockchain platform. This will help users trust the platform's processes, as they will be verifiable through the external public blockchain. The proposed technical approach to achieve this verification is through audits.

12 RISKS AND MITIGATION

The following tables complement the risks identified in TR1.2, focusing on the technical implementation of the DeepFakeChain platform.

TABLE 8: FEATURE RISK OF DEEPFAKECHAIN

Risk	Failure to implement all described features and requirements of DeepFakeChain.
Description	The user requirements of DeepFakeChain have been comprehensively described in TR1.1, including detailed user roles, privileges, and reputation models. Further system requirements of lower priority have been described in TR1.2 that are necessary for a fully-fledged platform but not the core features of the platform, e.g., visibility constraints and direct communication among users. Some of these features may not be implemented in DeepFakeChain in Phase 1.
Reasons	During Phase 1, the highest priority is to develop new AI algorithms for generalisable and explainable deepfake detection and establish the blockchain infrastructure, which could impact the development of secondary features.
Mitigation	The DeepFakeChain team will focus on delivering the core features of the platform with the highest priority, specifically, the deepfake detection services and the blockchain storage. This will result in a first functional prototype and scientific innovation, backed by a research paper submission. The team will further work towards implementing the majority of the remaining features. If some features are not implemented during Phase 1, they will be addressed in Phase 2 of the project.

TABLE 9: SCALABILITY RISK OF DEEPFAKECHAIN

Risk	Scalability limitations of DeepFakeChain.
Description	The scalability of the DeepFakeChain platform refers to the number of users and the number of uploaded multimedia, which influences both the conventional and the blockchain storage. The first prototype of DeepFakeChain may not reach the scalability level of a consumer-grade web application.
Reasons	The potential scalability limitations of DeepFakeChain may be due to the scientific focus of the project, the limited resources available for the engineering and deployment of the platform, and/or the limited number of users available to test the platform and identify the bottlenecks.
Mitigation	The DeepFakeChain team will take into account scalability concerns during development and choose appropriate tools and technologies. Any performance bottlenecks that are identified in Phase 1 will be promptly addressed. Phase 2 will offer further opportunities for testing under more realistic conditions, considering its focus to develop a fully-fledged web application.

REFERENCES

- [1] Spiros Baxevanakis, Giorgos Kordopatis-Zilos, Panagiotis Galopoulos, Lazaros Apostolidis, Killian Levacher, Ipek Baris Schlicht, Denis Teyssou, Ioannis Kompatsiaris, and Symeon Papadopoulos. The meyer deepfake detection service: Lessons learnt from developing and deploying in the wild. In *Proceedings of the 1st International Workshop on Multimedia AI against Disinformation*, pages 59–68, 2022.