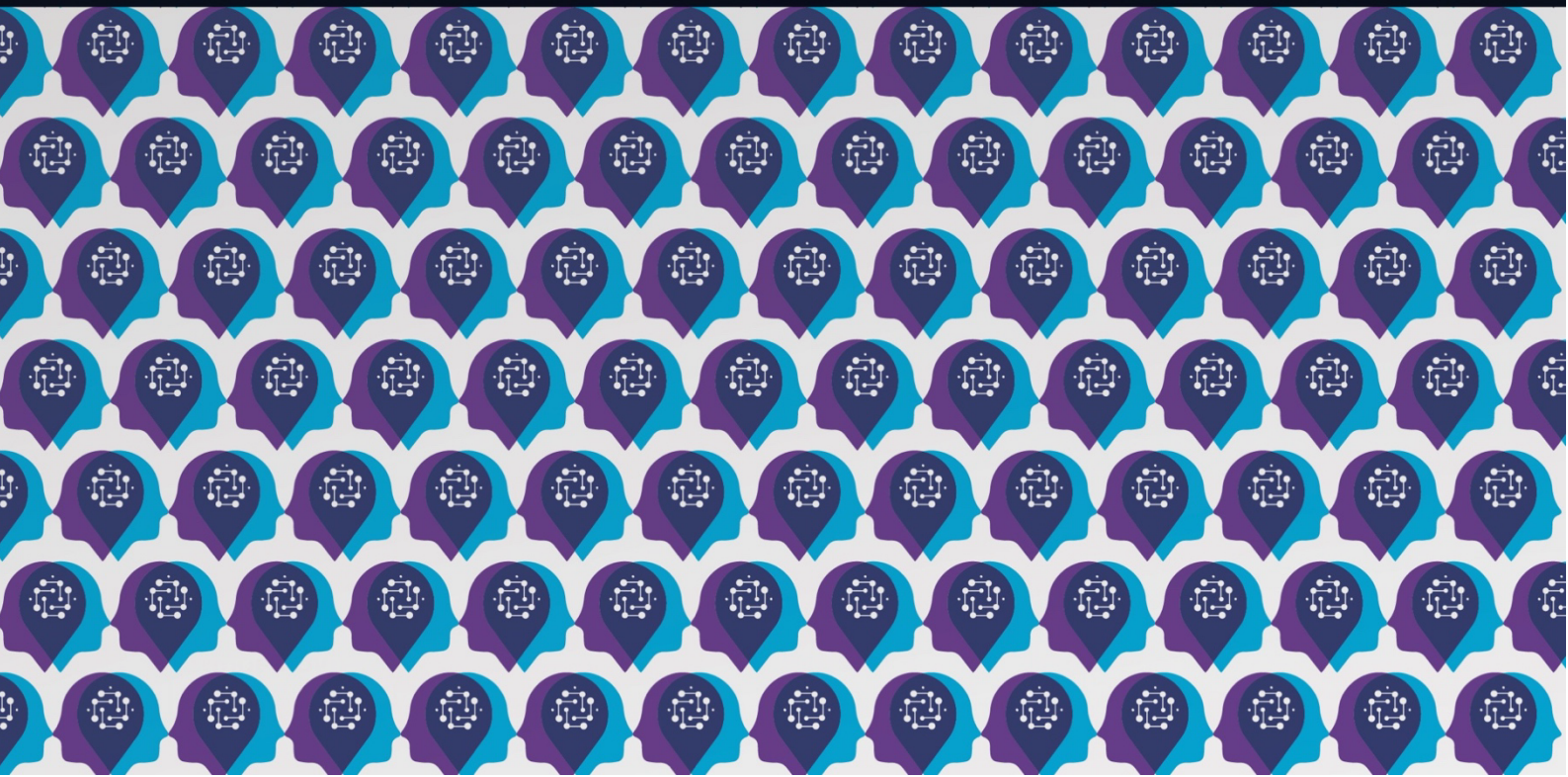
AI4Debunk

D10.2 Report on the definition of the browser extension

February 2026

Funded by
the European Union



Grant Agreement No.: 101135757
Call: HORIZON-CL4-2023-HUMAN-01-CNECT
Topic: HORIZON-CL4-2023-HUMAN-01-05
Type of action: HORIZON Innovation Actions

D10.2 REPORT ON THE DEFINITION OF THE BROWSER EXTENSION

Project Acronym	AI4Debunk
Project Number	101135757
Project Full Title	Participative Assistive AI-powered Tools for Supporting Trustworthy Online Activity of Citizens and Debunking Disinformation
Work package	WP 10
Task	Task 10.2
Due date	28/02/2026
Submission date	23/02/2026
Deliverable lead	Hogeschool Utrecht (HU University of Applied Sciences Utrecht)
Version	1.0
Authors	Chun Fei Lung (HU), Franc van der Bent (HU)
Contributors	Pascaline Gaborit (Pilot4DEV), Marcel Keijzer (IP)
Reviewers	Georgi Gotev (EUalive), Kalina Angelova (EUalive)
Abstract	This report presents the design and development of a browser extension intended to help users debunk online disinformation. It reviews existing research on browser extensions that counter false information. Drawing on social science research, requirements are outlined in the form of user stories, which inform both the user interface design and system architecture. The report also details the technologies used to implement the extension and discusses how it aims to support users in identifying misleading content on the web.
Keywords	disinformation, misinformation, debunking, browser, extension, add-on, plug-in

Funded by
the European Union

DOCUMENT DISSEMINATION LEVEL

Dissemination level	
X	PU - Public
	SEN - Sensitive

DOCUMENT REVISION HISTORY

Version	Date	Description of change	List of contributor(s)
0.1	22/10/2025	First working draft that is made available among all human-centred and technical partners actively working on human-centred user interfaces.	HU
0.2	06/01/2026	Second working draft that is distributed among all partners involved in the development of tools.	HU
0.3	09/02/2026	Final draft for internal review by EUalive.	HU
1.0	22/02/2926	Revised version after internal review by EUalive.	HU

STATEMENT ON MAINSTREAMING GENDER

The AI4Debunk consortium is committed to including gender and intersectionality as a transversal aspect in the project's activities. In line with EU guidelines and objectives, all partners – including the authors of this deliverable – recognise the importance of advancing gender analysis and sex-disaggregated data collection in the development of scientific research. Therefore, we commit to paying particular attention to including, monitoring, and periodically evaluating the participation of different genders in all activities developed within the project, including workshops, webinars and events but also surveys, interviews and research, in general. While applying a non-binary approach to data collection and promoting the participation of all genders in the activities, the partners will periodically reflect and inform about the limitations of their approach. Through an iterative learning process, they commit to plan and implement strategies that maximise the inclusion of more and more intersectional perspectives in their activities.

Funded by
the European Union

DISCLAIMER

The AI4Debunk project has received funding from the European Union's Horizon Europe Programme under the Grant Agreement No. 101135757.

COPYRIGHT NOTICE

How to cite this report: Lung C.F., Van Der Bent, J.F. (2026). AI4Debunk report on the definition of the browser extension. Link from website when deliverable is public.

The AI4Debunk consortium is the following:

Participant number	Participant organisation name	Short name	Country
1	LATVIJAS UNIVERSITATE	UL	LV
2	FREE MEDIA BULGARIA	EUALIVE	BE
3	PILOT4DEV	P4D	BE
4	INTERNEWS UKRAINE	IUA	UA
5	CONSIGLIO NAZIONALE DELLE RICERCHE	CNR-IRPPS	IT
6	UNIVERSITA DEGLI STUDI DI FIRENZE	MICC/UNIFI	IT
6.1	CONSORZIO NAZIONALE INTERUNIVERSITARIO PER LE TELECOMUNICAZIONI	CNIT	IT
7	BARCELONA SUPERCOMPUTING CENTER CENTRO NACIONAL DE SUPERCOMPUTACION	BSC	ES
8	DOTSOFT OLOKLIROMENES EFARMOGES DIADIKTIOY KAI VASEON DEDOMENON AE	DOTSOFT	EL
9	UNIVERSITE DE MONS	UMONS	BE
10	NATIONAL UNIVERSITY OF IRELAND GALWAY	NUIG	IE
11	STICHTING HOGESCHOOL UTRECHT	HU	NL
12	STICHTING INNOVATIVE POWER	IP	NL
13	F6S NETWORK IRELAND LIMITED	F6S	IE

Funded by the European Union

TABLE OF CONTENTS

Funded by the European Union

Funded by
the European Union

LIST OF FIGURES

Funded by
the European Union

LIST OF TABLES

Funded by
the European Union

LIST OF LISTINGS

Funded by
the European Union

ABBREVIATIONS

AI	Artificial Intelligence
API	Application Programming Interface
B2B	Business-to-business
CSS	Cascading Style Sheets
DOM	Document Object Model
EU	European Union
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
I18n	Internationalisation
L10n	Localisation
LLM	Large Language Model
RPC	Remote Procedure Call
SUS	System Usability Scale
TRL	Technology Readiness Level
UI	User Interface
UML	Unified Modeling Language
UX	User Experience
WCAG	Web Content Accessibility Guidelines
WP	Work Package
YAML	YAML Ain’t Markup Language

Funded by
the European Union

EXECUTIVE SUMMARY

AI4Debunk is a European initiative that aims to develop human-centred, multimodal, and collaborative AI tools to combat disinformation. Its human-centred solutions include three AI-powered debunking user interfaces that allow users to check and report content for disinformation – a browser extension, a smartphone app, and a collaborative platform – and an educational virtual reality (VR) game. This report focusses on the design and development of the browser extension.

We first conducted a literature review of existing browser extensions designed to counter online disinformation in order to identify main features of such extensions. Our analysis shows that some extensions allow users to check and collaboratively moderate content on the web, while others take a more active approach by labelling content on visited web pages.

Based on these key elements and previous research within the project – which examined the characteristics of disinformation and the groups most frequently targeted – we collaborated with social scientists in AI4Debunk to compile a set of MoSCoW-prioritised functional and quality requirements expressed as user stories.

These requirements were then translated into a human-centred user interface design, taking into account the extension's intended users, goals, environment, and system context. We describe this user interface using a series of low-fidelity mock-ups that explain the functionality, rationale, and illustrate how the requirements have been incorporated into the design.

Furthermore, we drafted an overall system architecture that shows how the browser extension interacts with the web browser, processes debunking requests using the debunking API, and integrates seamlessly with the accompanying collaborative platform.

Finally, we describe the implementation of the browser extension, using web technologies that facilitate long-term sustainable development in terms of both software maintainability and power efficiency.

Funded by
the European Union

1 INTRODUCTION

AI4Debunk is a European initiative that aims to develop human-centred, multimodal, and collaborative AI tools to combat disinformation and safeguard democratic values. Its human-centred solutions include three debunking user interfaces – a browser extension, a smartphone app, and a collaborative platform – and a virtual reality (VR) serious game.

The three debunking interfaces enable ordinary European citizens to assess the accuracy of online content directly, using a combination of state-of-the-art AI/ML models and a human-in-the-loop mechanism designed to enhance data quality and foster greater trust in the debunking process. Although all interfaces share common functionality, each leverages the unique capabilities of its respective platform.

This report focusses on the AI4Debunk browser extension which will be compatible with all major desktop web browsers. Once installed, it is permanently available to users as a “browsing companion” that assists in debunking disinformation and provides seamless access to the AI4Debunk collaborative platform.

1.1 OVERVIEW

This technical report describes the AI4Debunk browser extension, which is designed to assist users in detecting and verifying potential disinformation encountered while browsing the Web. The remainder of this report is structured as follows: Section 2 reviews prior work on browser extensions that address online disinformation. section 3 outlines the functional and quality requirements that guide the design and development of the browser extension. Section 4 explains how these requirements are translated into a concrete user interface. Section 5 presents an architectural design that supports the envisioned features in a maintainable way. Section 6 details the technologies used to implement the browser extension and the distribution strategy for core target audiences. Finally, Section 7 concludes the report with a discussion of the remaining steps required to finalise the browser extension.

Funded by
the European Union

2 BACKGROUND

Browser extensions – also referred to by their hypernym “browser add-ons” – are lightweight software components designed to extend and enhance the functionality of web browsers (MDN Web Docs, 2025a). They can, for example, dynamically interact with web page content, modify the appearance or behaviour of the current tab, or provide additional tools to the user within the context of the web browser. Extensions are typically developed using standard web technologies like HTML, CSS, and JavaScript, and are supported by all major desktop browsers on Windows, macOS, and Unix-like systems. On mobile platforms such as Android and iOS, support for browser extensions is more limited and varies by browser; many browsers do not support extensions at all.

Historically, browsers relied on plug-ins such as Adobe Flash Player and Java applets to provide features that were not natively supported. However, plug-ins often introduced serious security vulnerabilities and performance issues due to their unrestricted access to system resources. As browsers evolved, plug-ins were gradually phased out in favour of extensions, which may still cause performance issues if their implementation is not sufficiently optimised (Jin et al., 2025) but generally offer a safer and more standardised way to add new functionality through standardised web APIs (MDN Web Docs, 2025b).

Today, browser extensions serve a wide variety of purposes, including productivity enhancement (Zhao, 2023), improving accessibility (Pedemonte et al., 2025), protecting user privacy (Starov & Nikiforakis, 2018), facilitating online shopping, and filtering unwanted content (Hsu et al., 2024). They have also been employed in scientific research, for example, to collect data on algorithmic bias on X (Bartley et al., 2021) and to document cases of femicide across the Americas (D’Ignazio, 2024).

TABLE 1: SUMMARY OF RELATED BROWSER EXTENSIONS

Extension	Debunking approach			Status
	Analysis on demand	Proactive warnings	Miscellaneous	
BRENDA	✓			*Defunct*
ClarifAI			✓	*Defunct*
Décodex (FR)		✓		**Active**
InVID-WeVerify	✓			**Active**
MBFC		✓		**Active**
NewsGuard		✓		**Active**
Provenance	✓	✓		*Defunct*
Reheadline			✓	*Deprecated*
SEMiNExt		✓		*Source only*
SocialTruth	✓			*Defunct*
Stop Clickbait			✓	*Defunct*
Trustnet	✓			*Private only*
The Factual		✓		*Defunct*

Funded by the European Union

2.1 BROWSER EXTENSIONS AGAINST DISINFORMATION

Over the past decade, numerous initiatives have emerged that use AI to combat disinformation, many of which originate from Europe or the United States (Pilati & Venturini, 2025). For example, the European *Disinformation Against AI* cluster[1], which includes AI4Debunk, comprises several projects such as vera.ai, AI4Trust and TITAN. vera.ai[2] develops tools to assist verification professionals, building on earlier work from the WeVerify[3] and InVID[4] projects. Meanwhile, AI4Trust[5] is developing a hybrid system based on human-machine collaboration to support media professionals and policymakers. Finally, TITAN[6] adopts an educational approach, helping citizens become better prepared to recognise and resist disinformation.

Within the AI4Debunk project, Gaborit and Martinsen (2025) have previously outlined the advantages and disadvantages of various existing tools designed to counter disinformation. In this report, we focus primarily on the functionality, user interfaces, and other human-centred aspects of browser extensions that address disinformation, as described in recent scientific literature.

Our analysis of pre-existing browser extensions aimed at countering disinformation identified three major categories: extensions that assist users in debunking specific content, extensions that actively warn users about disinformation encountered online, and extensions that address issues closely related to, but not strictly constituting, disinformation. Table 1 provides an overview of the analysed extensions. Furthermore, we observe that some projects deliberately adopt a human-in-the-loop approach to ensure greater reliability, while others favour a technology-driven approach intended to scale the debunking process.

2.1.1 DEBUNKING DISINFORMATION

Extensions designed to help users debunk online disinformation typically provide a user interface for users to submit content for analysis. These tools check content for signs of disinformation and present their findings to the user in a clear and actionable format.

BRENDA (Botnevik et al., 2020) is a proof-of-concept extension that automates the credibility assessment of online claims. It allows users to submit their current page to an API, which uses a deep neural network architecture to first identify check-worthy claims, then classify claims as true or false, providing supporting evidence found using Google's search API.

Provenance (Yousuf et al., 2021), a Horizon 2020 project, also features a browser extension that eliminates the human in the loop, aiming to scale up the debunking process. Unlike BRENDA, it automatically analyses content on news pages and in users' news feeds, displaying results directly within the context of the page.

[1] https://www.disinfo.eu/ai-against-disinformation/
[2] https://www.veraai.eu/
[3] https://weverify.eu/
[4] https://www.invid-project.eu/
[5] https://ai4trust.eu/
[6] https://www.titanthinking.eu/

Funded by the European Union

Instead of a binary classification, Provenance scores content on seven criteria: recency of publication, geographical proximity of the publisher to the story's location, source reliability, corroboration by other outlets, article tone, writing quality, and whether images or videos have been manipulated. Unlike most other examples, which only support English, Provenance supports three different languages.

SocialTruth (Kozik et al., 2024), another Horizon 2020 project, addresses disinformation from a different angle. Whereas Provenance relies on a centralised server, SocialTruth argues that content verification should not be entrusted to a single authority, proposing a decentralised solution based on blockchain. The project's deliverables include a Digital Companion browser extension for Firefox, although its user base appears to be very limited. The project's focus is primarily technical, which is also reflected in the extension's user interface.

More recently, Jahanbakhsh and Karger (2024) introduced a browser extension for the Trustnet platform. They share the concern that content moderation is often centralised but approach it from a social perspective. The Trustnet platform and its browser extension enable decentralised moderation without requiring cooperation from the platforms where content is consumed. Within Trustnet, users assess the accuracy of online content and can choose whose moderation they trust, based on the idea that people are more receptive to corrections from friends than from strangers.

Finally, the InVID-WeVerify plugin, last maintained by Europe Horizon project vera.ai, is one of the few extensions related to disinformation with more than 100,000 users (Bontcheva et al., 2024). Unlike most other extensions aimed at general internet users, InVID-WeVerify is primarily intended for professional users, such as journalists and fact-checkers. As a result, the extension is designed as a toolbox offering a comprehensive set of functionalities to assist in determining whether content is disinformation. It does not provide easily understandable verdicts for laypeople.

2.1.2 PROACTIVE CONTENT WARNINGS

Many extensions, particularly those developed outside academia, focus on proactively warning users about disinformation they encounter on the web.

Several popular examples provide content warnings based on the trustworthiness of the content publisher, typically using a manually assigned reputation score derived from the publisher's historical record (Yousuf et al., 2021).

NewsGuard shows detailed transparency and credibility scores next to links on web pages, presenting "nutrition labels" based on nine journalistic criteria. Décodex, developed by French newspaper Le Monde, offers a simple traffic light-style indicator for suspicious content, but does not provide further details beyond a basic classification. Media Bias Fact Check (MBFC) is an extensive media bias resource curated by journalists and lay researchers. The extension shows a bias scale in users' feeds on Facebook and X (formerly Twitter), with additional information available on the MBFC website. It also enables users to investigate topics further using a Factual Search feature.

Funded by
the European Union

A potential drawback of these approaches is that they are labour-intensive and time-consuming, which means not all publications are covered. Furthermore, the warnings do not necessarily relate to the specific content currently being viewed by the user. The Provenance browser extension addresses this limitation by analysing the content itself, although it does not appear to consider the publisher's reputation (Yousuf et al., 2021).

The SEMiNExt browser extension, developed during the COVID-19 pandemic, adopts a different strategy. It monitors users' searches on major search engines and, when keywords potentially related to COVID-19 misinformation (such as "covid" or "hydroxychloroquine") are detected, it predicts the likely accuracy of the query – for example, "will consuming alcohol kill coronavirus?" – and provides links to information from reliable sources (Shams et al., 2021).

2.1.3 OTHER TYPES OF FALSE OR MISLEADING INFORMATION

Technically, disinformation refers specifically to false information that is deliberately created or disseminated with the intention of causing harm or gaining profit (Hameleers, 2025). In contrast, misinformation refers to unintentionally false information, while malinformation involves the deliberate sharing of factually correct information in a misleading context. Despite these differences, all three share similar characteristics and have comparable effects on public discourse. This section discusses several extensions that address these types of information.

"Stop Clickbait" (Chakraborty et al., 2016) is an experimental browser extension designed to detect clickbait headlines on web pages. It warns users about identified clickbait and can automatically block such content. Although clickbait headlines are not necessarily false, they can be misleading and divert users' attention from more informative news stories. The primary focus of the study was to assess the accuracy of the clickbait classification rather than the extension itself.

Similarly, Reheadline (Jahanbakhsh et al., 2022) targets misleading headlines, based on the observation that most people only remember headlines. The extension is collaborative in nature, allowing users to edit and share news headlines with their followers. The authors note that a single headline may appear across different publications, sometimes with slight variations in wording.

ClarifAI (Zavolokina et al., 2024) focusses on detecting propaganda in news articles and provides context-rich explanations that encourage users to think critically. It achieves this by directly using OpenAI's GPT-4 model through a few-shot, prompt-based learning approach, thus removing the need for a self-hosted server. An evaluation involving several prototypes suggests that detailed explanations are more effective at promoting critical thinking than simple indicators. The extension prototypes do not appear to have been published publicly.

2.1.4 BROWSER EXTENSION LONGEVITY

Similarly to Yousuf et al. (2021), who developed Provenance, we observe that many projects and services are no longer active, have been taken down entirely, or are otherwise unavailable. For example, some

initiatives have transitioned into commercial B2B services and are therefore no longer accessible to regular users.

Ironically, Provenance itself seems to have suffered a similar fate, with its tools now unavailable. “Stop Clickbait” and The Factual – previously analysed by Gaborit and Martinsen (2025) – have both disappeared from the internet, along with their browser extensions. BRENDA, once distributed as a manually installable ZIP archive, is no longer obtainable because the download link referenced in the original paper is inactive. SEMiNExt’s source code is still available via GitHub but requires users to build the extension themselves – a step few are likely to take. Reheadline remains available for now, although a warning on its Chrome Web Store listing suggests that it may soon become unavailable. SocialTruth also does not appear to be active anymore either, despite having been set up as a distributed system.

We suspect that many projects become abandoned once their initial funding runs out, primarily due to their reliance on APIs and remote servers that handle computationally intensive processing tasks. Maintaining such infrastructure can be costly, especially when factoring in ongoing operational expenses and security updates. Until devices hosting browser extensions become powerful enough to run the required AI models locally, or until these models are sufficiently optimised to run on conventional consumer devices without a remote API, any extension that relies on such infrastructure will remain at continuous risk of discontinuation.

Funded by
the European Union

3 REQUIREMENTS

The definition of the browser extension builds upon work completed earlier in the project. Previously, the third AI4Debunk working paper by Ozoliņa et al. (2025a) identified the groups most targeted by disinformation within Europe, highlighting those who would be likely candidate users of the extension. The fourth AI4Debunk working paper (Ozoliņa et al., 2025b) examined disinformation narratives and foreign interference throughout Europe, offering insights into the characteristics of disinformation in practice. Additionally, Gaborit and Martinsen (2025) explored how disinformation spreads on social media and outlined the first steps towards the design of an AI-based tool to counter disinformation online.

3.1 USER STORIES

In this section, we describe the requirements using user stories, which follow the template in Table 2. User stories are widely adopted in agile software development to incorporate user and business requirements into the development process (Lucassen et al., 2016). They are considered more effective than UML use cases for creating a shared understanding of a problem domain (Dalpiaz & Sturm, 2020) and are easier to apply than goal-oriented requirements engineering methods that are primarily uses in academic contexts, such as i* and KAOS (Mavin et al., 2017).

User stories ensure that the context of use, as defined in ISO/IEC 25002 (2024) – including users, their goals, the user environment, and the system context – is properly considered. Each user story is assigned a unique identifier for traceability throughout the software development lifecycle, labelled with its provenance – AI4Debunk deliverable 5.3 (Berretti & Caldelli, 2025), a social sciences and humanities (SSH) recommendation (appendix A), or our literature review – and tagged as either a functional or quality requirement. Functional requirements describe the capabilities needed by an actor with a particular role to solve a problem or achieve an objective (Van Vliet, 2008, p. 202). Quality requirements are based on the ISO/IEC 25010 (2023) standard, which defines product quality in terms of nine characteristics: functional suitability, performance efficiency, compatibility, interaction capability, reliability, security, maintainability, flexibility, and safety.

We prioritise user stories using the first three MoSCoW priority levels, which Van Vliet (2008, p. 59) defines as follows: **“must haves”** are required for a working system, **“should haves”** are important but not absolutely needed for a usable system, and **“could haves”** are only implemented if time permits.

TABLE 2: COMPONENTS OF A USER STORY TEMPLATE

Purpose	Template	Example
Role	As a(n) …	**As an** Outer Party desk employee at the Ministry of Truth
Goal	I want …	**I want** a memory hole within easy reach of my arm
Benefit	so that …	**so that** I can quickly dispose of truths that have been superseded.

Funded by the European Union

3.1.1 ANALYSIS ON DEMAND (INPUT)

The browser extension enables users to submit and analyse content that they encounter on the web, offline, or through other sources. The user stories in this subsection describe the different mechanisms through which users can provide content for analysis.

MUST FUNC D5.3

R1 As a casual news reader
I want to **check a textual or spoken claim that I encountered offline by providing some text**
so that I can verify its credibility.

MUST FUNC D5.3

R2 As a user
I want to **upload text, audio, image, and video files using a file dialog**
so that I can easily verify the credibility of files that I know are in a specific directory.

SHOULD FUNC D5.3

R3 As a user
I want to **upload text, audio, image and video files via drag and drop**
so that I can easily verify the credibility of files in a directory that I have already opened.

MUST FUNC D5.3

R4 As a casual news reader
I want to **check the current web page for disinformation**
so that I can understand to what extent it is credible without leaving the page.

SHOULD FUNC D5.3

R5 As a casual user of a popular social platform
I want to **check posts for disinformation**
so that I can understand to what extent it is credible without leaving the platform.

COULD FUNC SSH 3.4 D5.3

R6 As a casual user of a niche social platform
I want to **check posts for disinformation**
so that I can understand to what extent it is credible without leaving the platform.

MUST FUNC REVIEW

R7 As a power user
I want **a rich set of debunking tools (e.g. deepfake detection) at my disposal**
so that I can check content for specific signs of disinformation.

Funded by
the European Union

MUST FUNC SSH 4.4

R8 As a casual news reader
I want to **report suspicious content to AI4Debunk**
so that experts can validate it and potentially warn others.

3.1.2 ANALYSIS ON DEMAND (OUTPUT)

Users submit content to the browser extension with the intent to receive an analysis of the submitted content in return. This subsection outlines the user stories that define how the extension must or should present analysed information.

MUST FUNC SSH 1.2 D5.3

R9 As a casual news reader
I want **a disinformation assessment expressed in a single score**
so that I can quickly understand the credibility of content.

MUST FUNC SSH 1.2 REVIEW

R10 As a power user
I want **a disinformation assessment at a more granular level than a single score**
so that I can better understand the credibility of content.

MUST FUNC SSH 5.3

R11 As a casual news reader
I want **an explanation of why content is marked as disinformation**
so that I can trust the analysis and learn from it.

R12 SHOULD FUNC SSH 5.3 REVIEW
As a casual news reader
I want **links to verification resources (fact checks, reliable sources)**
so that I can easily verify flagged material myself.

SHOULD FUNC D5.3

R13 As a casual news reader
I want to **share the results of analyses with friends and family**
so that they can be more informed about a topic.

3.1.3 PROACTIVE WARNINGS

To improve users' online safety, the extension should automatically assess the trustworthiness of the content they view and notify them accordingly. This subsection describes user stories that specify when and how the extension should provide warnings or credibility indicators.

Funded by the European Union

SHOULD FUNC REVIEW

R14 As a casual news reader
I want to **see the credibility of a website based on historical analysis at a glance**
so that I can quickly judge trustworthiness before engaging with its content.

SHOULD FUNC D5.3 REVIEW

R15 As a casual news reader
I want to **see warnings embedded within the context of the page**
so that I can understand credibility without leaving the page.

SHOULD FUNC REVIEW

R16 As a casual news reader
I want **the disinformation analysis to appear automatically if it is already available**
so that I do not have to manually trigger one.

SHOULD FUNC SSH 1.2

R17 As a notification-sensitive user
I want to **choose when I want to receive warnings**
so that I only get relevant notifications.

SHOULD FUNC

R18 As a casual news reader
I want to **disable warnings for a website or page**
so that I can view content without any distractions.

SHOULD FUNC REVIEW

R19 As a casual news reader
I want **assessments for linked web pages**
so that I know their credibility before I click on them.

COULD FUNC REVIEW

R20 As a time-conscious user
I want **to be warned about clickbait headlines**
so that I can avoid wasting attention.

SHOULD FUNC REVIEW

R21 As a casual news reader
I want **to see what other people say about the content or its assessment**
so that I can decide who I want to trust.

SHOULD FUNC SSH 5.3 SSH 8.3

R22 As an informed user or content creator
I want **a dispute workflow accessible from the warning**
so that I can contest flags and provide clarifications.

Funded by
the European Union

3.1.4 USER FRIENDLINESS

If the browser extension is difficult to use, users will be less inclined to rely on it, thereby reducing its overall usefulness and impact. Although implementation of these user stories do not guarantee optimal usability on their own, they increase the likelihood that users will be able to navigate, understand, and benefit from the extension effectively.

MUST QUAL SSH 1.1

R23 As an elderly user
I want **larger, high-contrast UI elements**
so that I can easily notice and read credibility information.

MUST QUAL SSH 1.1

R24 As a power user
I want **a compact and efficient user interface**
so that I can quickly scan the rationale behind a warning.

MUST FUNC SSH 1.1

R25 As a casual news reader
I want **tooltips that provide contextual explanations**
so that I understand how to use the extension and interpret its results.

MUST QUAL SSH 1.2

R26 As a new user
I want **a simple user interface that only shows the most important information**
so that I'm not overwhelmed when I use the extension.

MUST QUAL SSH 1.2

R27 As a power user
I want **an advanced mode that shows all available information**
so that I can deeply understand what potentially makes the content disinformation.

SHOULD QUAL SSH 1.3

R28 As a first-time user
I want **an explanation of the extension's main features**
so that I can quickly get started.

SHOULD QUAL

R29 As a power user
I want **the debunking tools to work across platforms**
so I can use them on all my devices.

Funded by
the European Union

3.1.5 INCLUSIVE INTERFACES

We want the browser extension to be accessible to as many people as possible, regardless of their background, abilities, or preferred ways of interacting with technology. For this to be possible, the extension's design must implement certain features and follow relevant accessibility guidelines.

MUST FUNC SSH 2.4 SSH 6.3

R30 As a multilingual user
I want **to change the extension's user interface language**
so that I can use it in my preferred (European) language.

MUST QUAL SSH 2.4 SSH 2.6 SSH 6.1 SSH 6.3

R31 As a multilingual user
I want **the extension to work well for content in my own language(s)**
so that disinformation analysis is accurate and relevant regardless of language.

MUST QUAL SSH 2.4 SSH 2.5

R32 As a compliance officer
I want **the user interface to meet WCAG 2.1 AA requirements**
so that accessibility is ensured by default.

MUST QUAL SSH 2.1

R33 As a user of any gender
I want **the user interface to use inclusive language and design**
so that I feel respected and represented when using the extension.

MUST QUAL SSH 2.2 SSH 2.5 REVIEW

R34 As a user with an outdated or low-cost computing device
I want **the browser extension to use a minimal amount of system resources**
so that it does not slow down my browsing experience.

MUST QUAL SSH 2.2 SSH 2.5

R35 As a low-bandwidth user
I want **the extension to minimise downloads and uploads**
so that it remains usable on even on slow or metered connections.

3.1.6 SECURITY & PRIVACY

Security and privacy are essential for any software that claims to protect its users' wellbeing. This subsection outlines user stories that ensure that the extension processes data securely, respects users' privacy, and adheres to best practices in responsible data handling.

Funded by the European Union

SHOULD FUNC SSH 5.1

R36 As a privacy-conscious user
I want **a clear consent prompt explaining what data leaves my device**
so that I can make an informed choice before submitting content.

SHOULD QUAL SSH 5.1

R37 As a privacy-conscious user
I want **only the minimum necessary information to leave my device**
so that my privacy is protected during analysis.

MUST QUAL SSH 5.1

R38 As a privacy-conscious user
I want **submitted content to be deleted after usage**
so that the risk of information leaks is kept to a minimum.

MUST QUAL SSH 5.1

R39 As a privacy-conscious user
I want **to keep content I have viewed or submitted private unless I explicitly make it public**
so that my browsing history remains private.

SHOULD QUAL REVIEW

R40 As a user
I want **the extension to not rely on a single centralised server**
so that I can keep using it even if that server goes down.

Funded by
the European Union

4 DESIGN

This section outlines the user interface design of the AI4Debunk browser extension from the user's perspective. It describes the overall design philosophy, the main user interface (UI) components and how users interact with the extension.

The primary objective of the user interface to help users identify and debunk disinformation encountered while browsing the web. In addition, we aim to maintain feature parity and visual consistency with the other two human-centred debunking interfaces to ensure a cohesive experience: the collaborative platform developed by Kragt and Keijzer (2026) and the smartphone app by DOTSOFT (2026b). The design achieves this by following the design language and interaction patterns of other interfaces.

The collaborative platform and browser extension are largely built using the same web technologies, enabling them to reuse technical components with minimal effort. This approach reduces development overhead and allows resources to be allocated to the development of additional functionality.

One notable advantage of the collaborative platform is that it can be used full screen, making it easy to show detailed information to power users. In contrast, the browser extension's user interface is constrained in size as it must exist within the confines of a popup window. To address this limitation, the extension is designed to optionally delegate certain features to the collaborative platform. This effectively makes the browser extension a lightweight platform companion that always remains accessible to users.

4.1 OVERVIEW

The browser extension's user interface consists of a set of distinct views that can be modelled as a state machine. The UML state diagram in Figure 1 summarises the extension's main states and illustrates the possible transitions between them. The extension remembers its current state and automatically restores it when reopened within the same browser session, ensuring that users do not lose progress if they accidentally close the extension window – an issue observed often during early prototype testing.

When the browser extension is opened, it initially displays the main view, which provides quick access to other views for verifying information and adjusting settings. Although not depicted explicitly in Figure 1, users can always return to this main view regardless of the extension's state. A few features that require significant computational and human resources are available only after the user logs in, but most functionality will be immediately accessible to users. When the extension is opened for the first time, it presents an onboarding view to guide the user through the initial setup process.

The following subsections describe the extension's main views. While the core concepts outlined here are expected to remain stable throughout development, certain details – particularly those related to visual design – may evolve as new insights emerge or as changes are introduced to the collaborative platform and smartphone app.

Funded by
the European Union

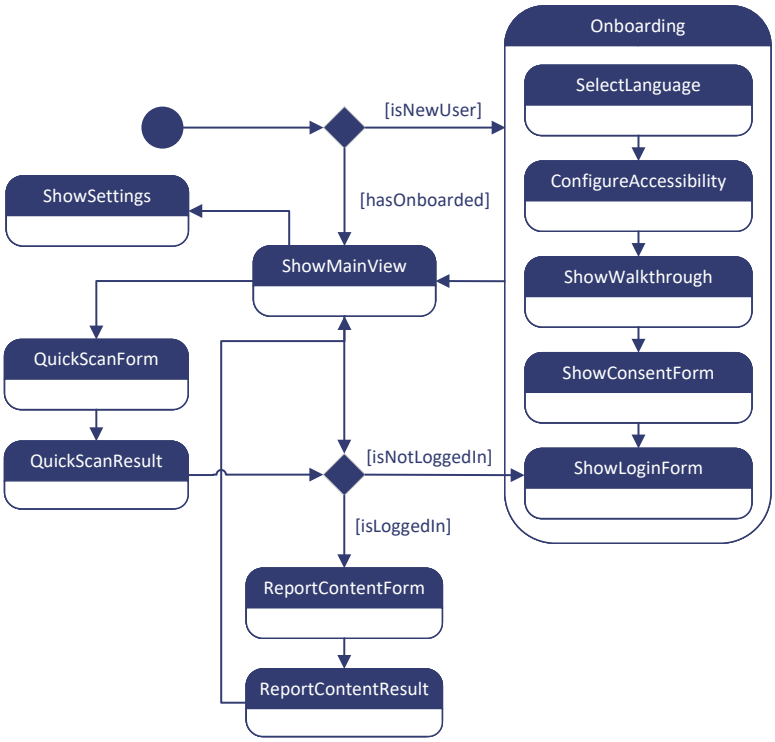


FIGURE 1: UML STATE DIAGRAM SHOWING THE EXTENSION'S MAJOR STATES AND TRANSITIONS

4.2 ONBOARDING

When users open the browser extension for the first time by clicking on its icon, they are guided through an onboarding process, designed to configure the extension and introduce its core features. This process consists of several panes that help users understand how the extension works and encourage continued use.

The onboarding begins with a language selection pane, allowing users to choose from the extension's supported languages (R30). By default, the extension attempts to detect and apply the locale[7] configured in the user's browser (e.g., `nl_BE` for Belgian Dutch). If the exact locale is not supported, the extension will try to match the same language in a different country, e.g. `nl_NL` for the variant of Dutch used in the Netherlands. If no match is found, it defaults to European English that is not specific to any country.

After language selection, users are prompted to configure accessibility settings (see section 4.9.2 for details). This step ensures that users with visual or situational impairments can complete the onboarding process comfortably (R32).

[7] A locale is an identifier that defines the user's preferred language, often for a specific region or script variant (e.g. Cyrillic or Latin).

Funded by the European Union

Next, the extension presents a series of panes that provide a brief overview of key features, including proactive content warnings, quick scans, and content reporting (R28).

Once the extension has explained to the user what it can do for them, it presents a consent form requesting permission for the actions it must perform in order to provide those capabilities, such as temporarily storing and processing submitted data.

Finally, users are shown a login pane, where they can sign in or register for an account via the collaborative platform. The default authentication method is username and password, but users may also log in via popular identity providers such as Google or LinkedIn. User may choose to skip this step. If they do, the extension displays a one-time message indicating that some functionality will remain unavailable until they log in.

4.3 MAIN VIEW

After completing the onboarding process, the browser extension opens to the main view whenever it is launched. Figure 2 shows a low-fidelity mock-up of this view.

The extension's popup interface is designed to fit comfortably on displays of lower-resolution devices (R34) and avoid covering too much of the underlying web page, as excessive intrusion can negatively impact the user experience (R24). This limits the maximum size of the popup window.

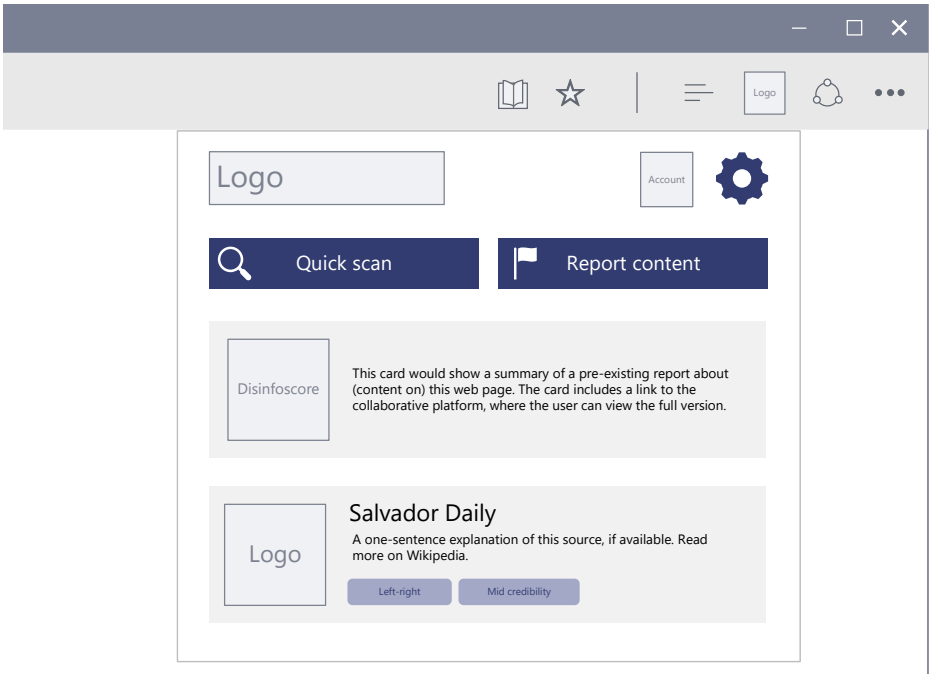


FIGURE 2: LOW-FIDELITY MOCKUP OF THE EXTENSION'S MAIN VIEW

Funded by the European Union

When users click the extension's icon, they likely intend to perform an action. To minimise interaction effort, all primary actions buttons are positioned near the top of the interface, reducing the distance the mouse cursor must travel. These buttons include:

1. **Account icon**: If the user is not logged in yet, this shows an unobtrusive login button. For users that are logged in, their avatar on the collaborative platform is shown here, along with the option to open the profile page in their web browser or log out.
2. **Settings**: Allows users to customise the extension. Section 4.9 provides an overview of available settings.
3. **Quick scan**: Enables users to quickly check content for signs of disinformation using modules developed in WP8–9. This feature is explained in detail in Section 4.4.
4. **Report content**: Allows users to report content to the collaborative platform, where it can be verified and shared with others. If the user is not logged in, clicking this button prompts them to log in. Further details are provided in Section 4.5.

Because screen space is limited, contextual help is provided via tooltips when users hover over a button or focus on it using the keyboard (R25).

Below the interactive section of the main view is an area reserved for contextual information related to the current web page. This section is positioned closer to the content the user is viewing, making it more noticeable. It contains two subsections:

1. **Page information**: Visible only if the current page contained validated reports of disinformation.
2. **Source information**: Displayed if the system already has a pre-calculated reliability score derived from MBFC and similar databases.

If no information is available about the page or source, the extension displays some overall statistics about the amount of content that has been submitted to the AI4Debunk platform, and a call to action encouraging the user to run a quick scan or report content if they are uncertain about its trustworthiness.

4.4 QUICK SCAN

The Quick Scan feature allows users to assess content for potential disinformation and generates a disinfoscore along with some additional information about the submitted content.

The extension provides a minimal interface with two input options: a text field for manual text entry (R1) and an upload area for submitting text, audio, image, or video files via the system file dialog (R2) or using drag-and-drop (R3). Although it is technically possible to record audio and video as in the smartphone app, we choose not to support this functionality because it would require the extension to request additional permissions that users are unlikely to expect, given how infrequently these features are used, and which may therefore cause concern.

TABLE 3: INDICATION OF REQUIRED INPUTS FOR EACH WP8–9 DEBUNKING MODULE

Debunking module	Text	Text file	Audio file	Image file	Video file
Similarity with other news	OR	OR			
Deepfake detection			OR	OR	OR
Cross-modal coherence check	AND			AND	

Table 3 outlines which debunking modules are triggered for which input types (R7), based on the current state of the AI4Debunk system. The similarity check is activated when text and/or a text file is provided. Deepfake detection runs when an audio, image, or video file is uploaded. A cross-modal coherence check is performed when both text and an image file are included in the input. The user interface dynamically indicates which checks will be executed based on information provided by the backend. Users may also explicitly disable checks to speed up the process.

When the "Check content" button is clicked, the extension shows a loading spinner and a textual status update. Processing always happens in the background, allowing users to close the extension window and continue browsing. If processing is expected to take more than a few seconds, an explanatory message is shown. The user is notified once the analysis is complete. When they re-open the extension, it will immediately show the results of the current analysis.

Upon completion, the extension presents the disinfoscore (R9) along with a summary of results for each debunking module (R10, R26). Users can expand detailed information on demand to explore findings further (R11, R27) and optionally report the content to the collaborative platform. The design for presenting results is still under development as it depends on work done in concurrently running work packages and will be refined iteratively via prototypes with user tests.

The low-fidelity mock-up in Figure 3 shows the four major sub-views of the Quick Scan feature. Note that during the initial phases, the extension will only support checks related to climate change and the war in Ukraine. The UI will indicate this in the same way as the collaborative platform and smartphone app.

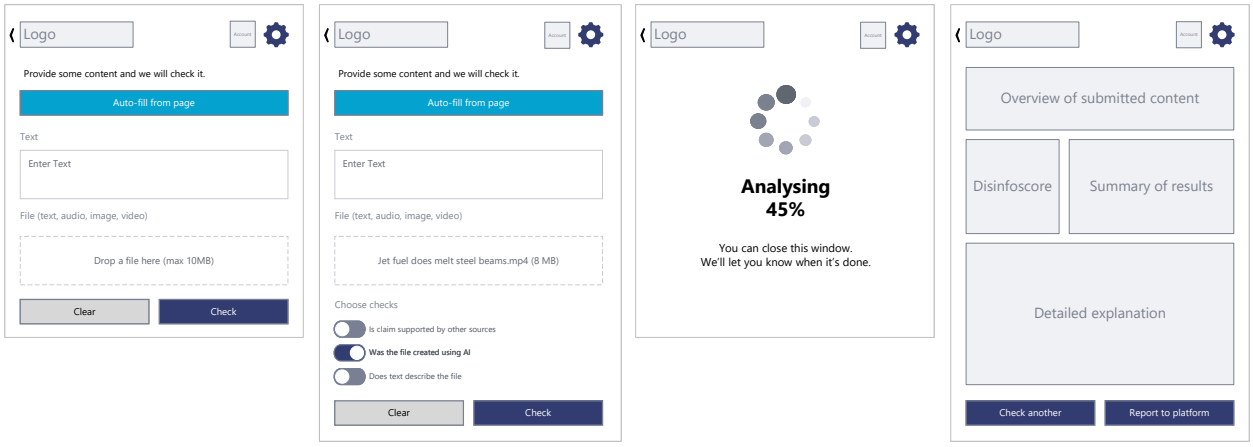


FIGURE 3: LOW-FIDELITY MOCK-UPS OF QUICK SCAN FEATURE

Funded by
the European Union

4.5 REPORTING CONTENT

The Report Content feature allows users to submit reports of disinformation to the AI4Debunk validation system (R8). To create a report, users must provide sufficient information to enable validation of the content through AI, supported by a human-in-the-loop mechanism. Once submitted, these reports may be shared with other users on the platform or made publicly accessible on the internet (R13).

The user interface for this feature closely resembles that of the Quick Scan feature described in Section 4.4. In terms of functionality, it is identical to the corresponding feature in the collaborative platform. For a comprehensive description, we refer to Kragt and Keijzer (2026) rather than reproducing the details here.

4.6 SELECTING CONTENT

All three debunking interfaces allow users to submit content manually via forms. For the browser extension, this means that when users encounter content on the web they wish to verify, they must click the extension icon to open a submission form, temporarily leaving the context of the page.

However, the browser extension offers a key advantage: direct access the user's current web page. This capability enables the browser extension to provide additional methods for completing submission forms without requiring manual input or copy-pasting (R4):

1. **Auto-fill from page**: A button that, when clicked, automatically extracts relevant information from the current tab by analysing its semantic markup and document structure using heuristics. This approach is most suitable when the user is already interacting with the extension, as it avoids the need to return to the original page. The button is shown in Figure 3 and described in more detail in Section 5.2.
2. **Context menu**: Users can select content directly on the web page and submit it to the extension via a dedicated entry in the right-click context menu. This method provides greater control over what content is checked but also introduces challenges. Selection offers virtually unlimited flexibility, whereas the debunking API and its modules only support specific input formats. For example, a user might select an entire article containing multiple images, but the expected behaviour for such input is currently undefined. Large-scale user testing is required to assess the practical viability of this method.

4.7 PROACTIVE CONTENT WARNINGS

The extension will provide warnings when users visit a page containing content that is identical to, or closely resembles, material previously confirmed as disinformation through AI4Debunk's validation process (R16). Because these warnings are shown automatically, without requiring any user interaction, this feature is straightforward to use and encourages users to explore other functionality of the extension.

Funded by
the European Union

Warnings are displayed in two locations: in the browser extension icon and within the page itself. The extension icon shows a simple indicator signalling whether the website or the page contains known disinformation. Clicking the icon opens the main view, which provides details about the website publisher (R14) and the content currently displayed.

The extension may also display warnings adjacent to specific content previously reported to the AI4Debunk platform (R15). Only limited information is shown within the page, but users can hover over the warning to view a summary of the report (R25, R26) or click it to access a comprehensive report on the collaborative platform in a new browser tab (R27).

Initially, warnings will be generated based on a semi-normalised URL[8] and the page content at the time of loading. As a result, the first versions of the extension may not work reliably platforms which use dynamically updating interfaces, such as social media platforms and messaging services. Given that these platforms are significant sources of disinformation, future releases will include explicit support for at least the most popular social platforms (R5) that were previously identified via surveys in earlier work packages.

4.8 INTEGRATION WITH COLLABORATIVE PLATFORM

The browser extension is designed to complement the collaborative platform. Several features discussed in previous sections – such as authentication and access to full content warning reports – already rely on this integration. However, additional possibilities exist to strengthen the connection between the two applications, for example:

- **Seamless authentication**: Instead of requiring users to register and log in within the extension using a username and password or via an identity provider, the browser extension could detect an active session on the collaborative platform and automatically authenticate the user. This would reduce login to a simple one-click action.
- **Profile linking**: The main view of the extension could display the user's avatar that links directly to their profile on the collaborative platform. This would reinforce the relationship between the extension and the platform.
- **Notifications**: The extension could show the number of unread notifications from the collaborative platform, providing additional mechanism to keep users engaged.

At the same time, it is essential that the browser extension remains functional without the collaborative platform and can operate solely with the debunking API. This behaviour can be configured in the settings pane, which is described next.

[8] That is, excluding query parameters, such as "utm_source", and anchors (#).

Funded by the European Union

4.9 SETTINGS

While the browser extension strives to provide an optimal user experience by default, individual needs and preferences vary. To accommodate these differences, the browser extension includes a settings pane that allows users to customise their AI4Debunk experience. This pane is organised into several sections, described below.

4.9.1 LANGUAGE

The language used within the extension is set by the user during onboarding. This setting allows users to change the UI language at any time (R30), which is helpful if additional languages are introduced in future updates or if a wrong language was selected during onboarding.

4.9.2 USER INTERFACE CUSTOMISATION

The browser extension is designed to be accessible out of the box (R32). However, some users may have specific needs go beyond the default configuration. Operating system or browser-level accessibility preferences are generally applied automatically, but not all users wish to configure these globally. Therefore, the extension provides equivalent settings locally, enabling users to override defaults specifically for the browser extension. Accessibility settings can be set during onboarding and modified in the "User interface" section of the settings pane. We use the term "user interface" rather than "accessibility", as these options may benefit all users, not just those with impairments.

Planned customisation options include:

- **Colour scheme**: The default design uses light mode, as this is common and often associated with trustworthiness. However, users may prefer a dark mode, and for some, light text on a dark background improves legibility.
- **Font size**: Users can adjust font size to fit more information on the screen or improve legibility. Larger fonts are particularly helpful for those with visual impairments.
- **Contrast**: A high-contrast mode will be available, using bright and dark colours for maximum visiblity. Research suggests that users with photophobia may benefit from a low-contrast mode (Andrew & Tigwell, 2025). If feasible, the option will also be included.
- **Animations**: Minimal animations are used for visual feedback and guiding attention. However, animations may cause discomfort for users with vestibular motion disorders (MDN Web Docs, 2025c). Animations can therefore be disabled entirely.
- **Colour blindness**: We avoid conveying information solely through colour wherever possible. For cases such as with visualisations, users will be able to customise colours based on their type of colour blindness.

Funded by
the European Union

4.9.3 PRIVACY

Users should remain in control of their privacy at all times. Even after onboarding, they can revisit the settings pane to adjust preferences by enabling or disabling specific features (R36, R37):

- **Proactive warnings**: When enabled, the browser extension checks every visited page for signs of disinformation. This requires sending the page URL (for source trustworthiness) or its content (for content trustworthiness) to the server. Although the server does not store submitted data, privacy-conscious users may choose to disable this feature.
- **Excluded domains**: Users can disable autofill and proactive warnings for specific websites, such as corporate intranets. This allows selective privacy protection without disabling the extension entirely or relying on private browsing mode.
- **Telemetry**: The extension can collect anonymised data about feature usage to improve functionality and support usability studies. Telemetry is disabled by default, as it offers minimal direct benefit to users, but can be manually enabled by the user.

Furthermore, users can request the deactivation of their account from the AI4Debunk system. This will anonymise and disable the account, remove all data collected about the user from AI4Debunk's databases, and delete any submitted data that remains in the system, as well as any reports derived from it.

4.9.4 DEVELOPER OPTIONS

The "Developer options" section is intended for advanced users. Access requires acknowledgement of potential risks and agreement to proceed at their own discretion.

A key feature is the ability to specify an alternative debunking API server (R40). This is primarily for users who wish to run their own server to provide their own debunking modules, authentication, or ensure that all submitted data remains on premises (R39). However, changing this setting is not encouraged, as malicious actors could exploit it by directing users to compromised servers. For this reason, any attempt to modify the server address triggers a warning advising users to ensure they trust the server owner.

4.9.5 ABOUT

This section provides an overview of the browser extension, identifies the organisations responsible for its development and maintenance, and offers background information about AI4Debunk to foster trust. It also includes links to key legal documents, such as the terms of service (ToS), privacy policy and software license. At the time of writing, the terms of service and privacy policy have not yet been drafted. However, they will be developed over the coming months in a coordinated effort among all WP11 technical partners to ensure that all user interfaces present consistent information to users.

Funded by
the European Union

5 ARCHITECTURE

The browser extension is designed according to a client-server architecture in mind. This architecture consists of two primary components that communicate over a computer network: a server that provides services and resources, and a client that requests these services and resources (Van Vliet, 2008, p. 259).

One key advantage of a client-server architecture, compared to a fully client-side setup, is that large language models and other AI models used for debunking are currently too computationally demand to run on most consumer devices, such as laptops and desktop computers. By offloading these intensive tasks to a centralised server, client requirements remain minimal (Geewax, 2021, pp. 4–5), thereby broadening access to the debunking tools for a wider audience (R34).

5.1 CLIENT-SERVER INTERACTIONS

Figure 4 provides a schematic overview of the overall architecture. The browser extension communicates with two complementary server applications: the debunking API and the collaborative platform.

The debunking API, described in D10.1 by Lung and Van Der Bent (2026), serves as the primary server in this architecture. It enables the browser extension to perform remote procedure calls (RPCs) to verify content using the API's debunking models.

The collaborative platform, described in D10.4 by Kragt and Keijzer (2026), acts as a secondary server. As a central user-facing web application within the AI4Debunk system, it manages account creation and authentication, detailed reporting, and push notifications. This design offers several benefits: users can reuse their platform accounts to log in to the browser extension, and the extension can redirect users to the platform for additional functionality and engagement. Furthermore, it allows for technical reuse of functionality that is already present in the platform.

To avoid having to configure the browser extension with two distinct remote servers, the collaborative platform includes an API redirection service. This service transparently exposes endpoints from the

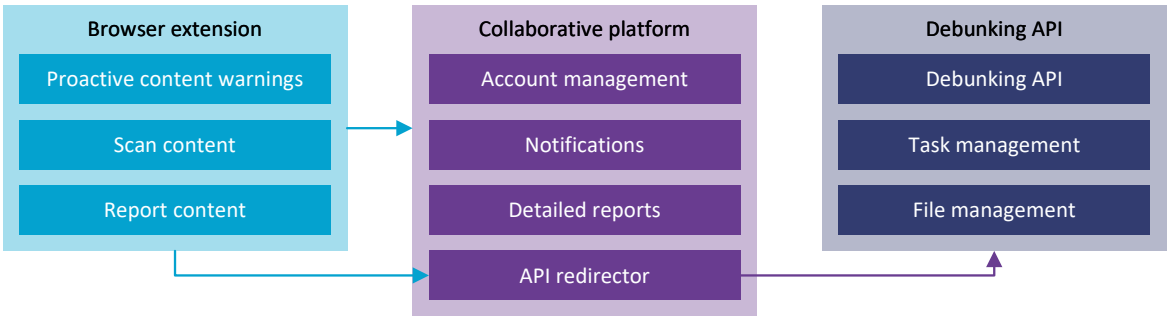


FIGURE 4: INTERACTION BETWEEN THE EXTENSION WITH THE COLLABORATIVE PLATFORM AND DEBUNKING API

Funded by the European Union

debunking API and forwards client requests accordingly. Conceptually, the browser extension only needs to interact with the collaborative platform.

Although the collaborative platform plays a central role in this architecture, it remains optional from a technical perspective. The browser extension will also be capable of communicating directly with the debunking API, as elaborated in Section 6.3.

5.1.1 DYNAMIC USER INTERFACE

Browser extensions are typically distributed via app stores and must undergo a review process before being published to ensure compliance with store guidelines. Only after approval does a new version become available. Depending on the store, this approval process can be time-consuming and resource-intensive, posing a risk to the speed of iteration and improvement of the debunking tools.

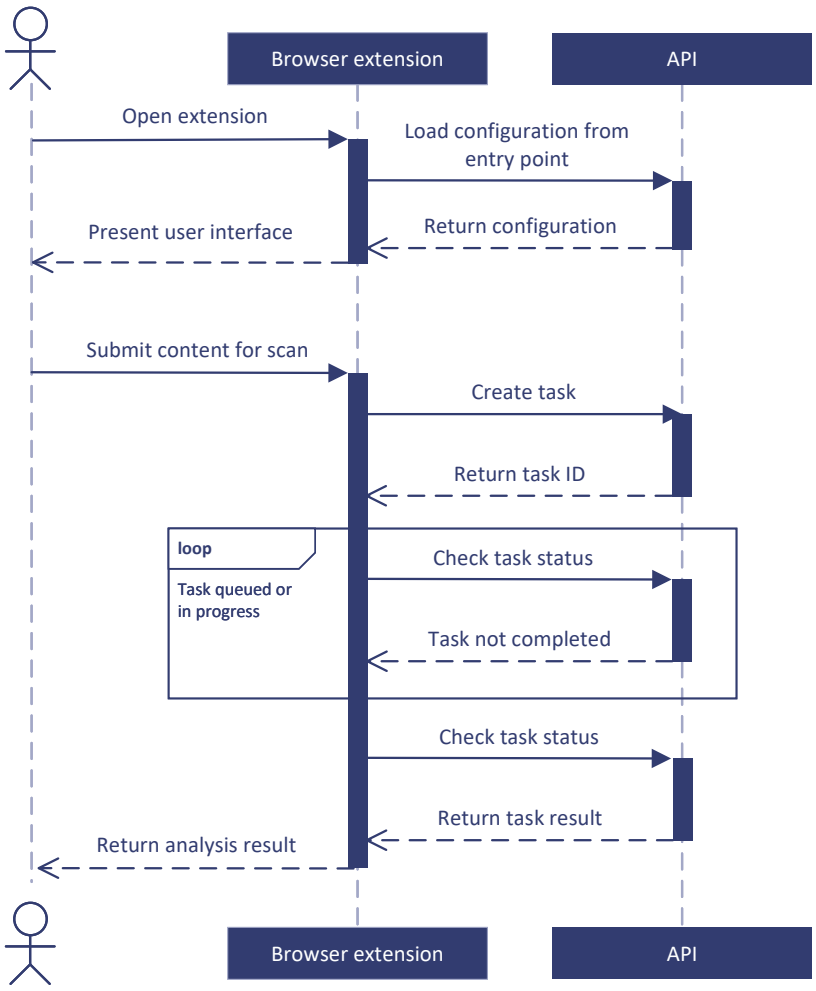


FIGURE 5: SEQUENCE DIAGRAM SHOWING MAIN INTERACTIONS WITH THE DEBUNKING API

To mitigate this risk, the browser extension is designed so that UI components related to debunking are not hardcoded. Instead, the debunking API provides a dedicated entry point for the browser extension that returns hyperlinks to essential legal information (e.g., the server's privacy policy), the available debunking modules, and the required UI layout. This approach enables dynamic updates to the extension's user interface without rebuilding or resubmitting the extension for review. Figure 5 illustrates this process with a UML sequence diagram showing how the UI is constructed, allowing users to verify content via one of the API's debunking modules.

5.2 AUTOFILLING SUBMISSION FORMS

The browser extension enables users to submit text and media files for analysis for signs of disinformation. Our goal is to make this process as seamless as possible. Since the extension has access to the user's current browser tab, it can attempt to automatically populate the submission form with content from the page being viewed via an autofill feature – similar to the AI4Debunk smartphone app, which allows users to check content directly using their camera and augmented reality (Filippidou et al., 2026b).

The autofill feature requires the tool to infer which information the user intends to submit. This can be achieved in several ways, depending on the website. Pages with semantic markup are relatively easy to process, whereas those with less structured markup may require heuristics or site-specific rules to extract relevant information.

We propose a modular extraction system, as illustrated in Figure 6. The system uses an ordered list of content extractor classes, starting with site-specific extractors for pages with dynamic content, a

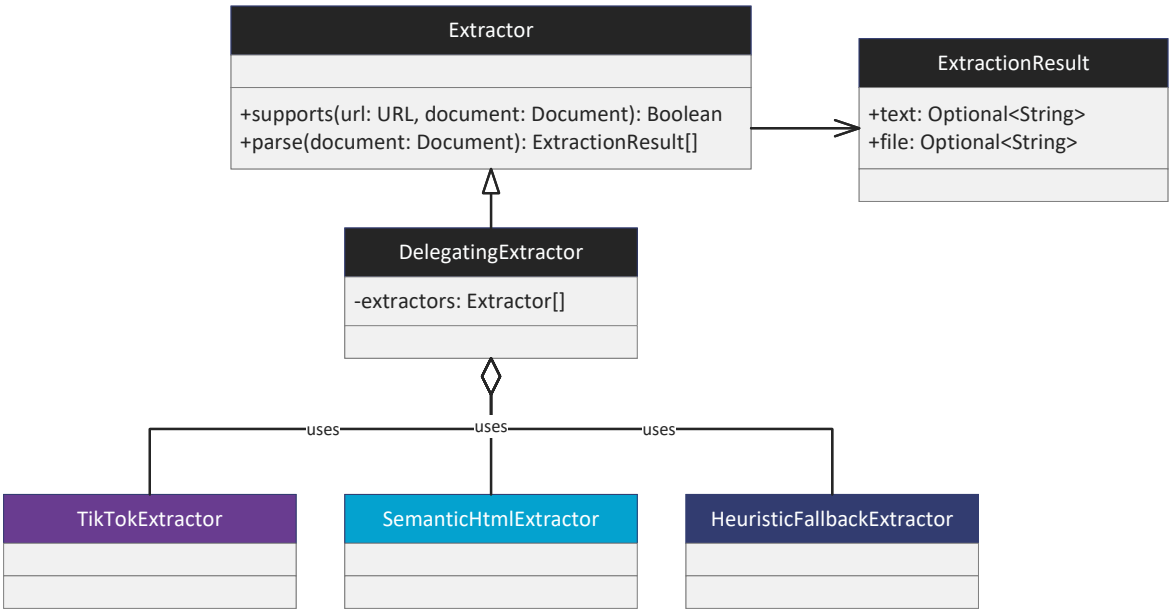


FIGURE 6: CLASS DIAGRAM OF LIGHTWEIGHT CONTENT EXTRACTION SYSTEM

Funded by the European Union

lightweight semantic extractor suitable for most websites, and finally a Readability.js-based[9] heuristic extractor that can produce results even on poorly coded websites. Each extractor is tried in sequence until content is successfully extracted, or all extractors fail – in which case the autofill feature will be disabled for the current page.

A similar system exists in the debunking API, but the version implemented in the browser extension will be much simpler. This design keeps the extractor system fast and lightweight and ensures that the browser extension generally does not need to transmit the user's entire page, which may contain sensitive information. Avoiding such transmission reduces the likelihood that updates to the extension require extensive privacy and security reviews by app stores.

[9] https://github.com/mozilla/readability

Funded by the European Union

6 IMPLEMENTATION

This section outlines the main tools and technologies used to develop the browser extension, including programming languages, frameworks, and libraries. It also explains how we plan to distribute the extension via major app stores and directly from GitHub.

6.1 WEB TECHNOLOGIES

Like regular web applications, browser extensions are developed in JavaScript (MDN Web Docs, 2025a). However, JavaScript as a language has a significant limitation: its lack of static typing makes it more difficult to catch type-related bugs during the development, increases the risk of errors during refactoring, and means that code clarity often depends on comments. Unfortunately, such comments can easily become outdated and misaligned with the code they describe, reducing their reliability over time (Lung, 2021).

TypeScript, introduced in 2012 by Microsoft, is a statically typed superset of JavaScript. Research suggests that TypeScript improves software maintainability by enhancing code quality and understandability (Bogner & Merkel, 2022). Consequently, we use TypeScript to develop our browser extension. However, since TypeScript must be transpiled to JavaScript before it can be executed in a web browser, we refer to the extension as JavaScript-based throughout this section for consistency.

Most modern JavaScript applications are not developed entirely from scratch but make extensive use of libraries and frameworks (Ollila et al., 2022). For our browser extension, we rely on two types of frameworks: one designed to streamline extension development and another focussed on simplifying the creation of dynamic user interfaces.

6.1.1 BROWSER EXTENSION FRAMEWORK

An extensive web search identified three major frameworks for developing browser extensions: Plasmo, WXT, and CRXJS. Among these, Plasmo is the most popular, with almost 13,000 stars on GitHub. It is a comprehensive, batteries-included framework that describes itself as the “Next.js for browser extensions”. WXT, which has close to 9,000 stars, is more focussed on the developer experience. CRXJS, the least popular of the three with 3,900 stars, is easier to set up but offers fewer features out of the box and appears less well-maintained.

After building throwaway prototypes with all three frameworks, we concluded that WXT is the most suitable choice for the AI4Debunk browser extension. Although CRXJS simplifies the initial setup phase, this benefit is short-lived, as setup is only required once. Both Plasmo and WXT come with robust tooling to support ongoing development, but we found WXT to be more effective overall. Its development environment and features are especially helpful for potential future contributors, making it easier for those without prior experience in browser extension development to get started and provide meaningful contributions.

Funded by
the European Union

6.1.2 USER INTERFACE FRAMEWORK

User interface frameworks streamline the development of web applications by providing a declarative paradigm for building user interfaces. This approach is less error-prone than directly manipulating the structure and content of web pages using the browser's imperative DOM API.

React is currently the most popular framework for declarative user interfaces and is used by major news websites such as the BBC and Deutsche Welle. However, a benchmark study by Ollila et al. (2022) compared React with five other popular frameworks and found that React often consumes more resources than competing frameworks. Code generated by the Svelte framework consistently performs best due to its small size and efficient method of updating the user interface.

Because we want our browser extension to be accessible to as many users as possible – including those with older devices and slower internet connections – we consider Svelte a more suitable choice than React. Svelte's efficiency not only improves the user experience (R34), but also reduces the power consumption of the extension, which helps minimise its environmental impact (Rani et al., 2024).

The user interface consists of components such as buttons, input fields, and the AI4Debunk logo. As the browser extension and the collaborative platform share some features and user interface elements, we are developing a custom component library that will be used by both applications (Kragt & Keijzer, 2026). This approach minimises duplication of effort and ensures a consistent user experience across applications.

6.2 INTERNATIONALISATION

To reach a broad audience and maximise impact, the browser extension must be accessible to citizens of all EU countries.

Internationalisation (I18n) refers to designing and developing software so that it can be easily localised for different cultures, regions, and languages. For the browser extension, our primary focus is technical language support. In the initial development phase, the extension will support three languages: English, Dutch, and Klingon. These choices reflect three distinct use cases:

- **English** will serve as the primary language across all user interfaces, as it is widely understood throughout Europe. It functions both as the default interface language and as a reference to assist translators in adapting content to other languages.
- **Dutch** is included as the native language of the development team, providing an early test case to evaluate how the user interface accommodates different linguistic characteristics, such as verbosity and word length. For example, in English and many Romance languages compound words are typically written as sequences of separate words, whereas languages such as Dutch and German often use long compound words that may require hyphenation or UI adjustments to accommodate longer words.

Funded by
the European Union

- **Klingon** is a constructed language which, to the best of our knowledge, is not spoken in any European country. We intend to use Klingon as a placeholder language during design and development. Its obscurity makes it an effective tool for assessing UI clarity, particularly for users with low literacy levels.

Modern web browsers provide a localisation (L10n) API [10] that allows developers to automatically translate browser extensions using developer-supplied translation files. An example of such a translation file is shown in Listing 1. A translation file contains not only the translated static text (the `message`) for a given translation key, but also a description that provides context for where the translation string is used, as well as placeholders that enable customisation of the translated text. This structure makes it easier for translators to understand the intended context and avoid common issues associated with machine translation, such as errors caused by polysemy, where a word is translated with an incorrect meaning.

```
{
  "weHaveAlwaysBeenAtWar": {
    "message": "We have always been at war with $enemy$.",
    "description": "Reminds the user who is responsible for all that is bad.",
    "placeholders": {
      "enemy": {
        "content": "$1",
        "example": "Eurasia"
      }
    }
  }
}
```

LISTING 1: MINIMAL EXAMPLE OF TRANSLATION FILE

6.3 DISTRIBUTION

As of January 2026, the most popular desktop browsers in Europe are Google Chrome, Microsoft Edge, Apple Safari, Mozilla Firefox, and Opera (StatCounter, 2026). Of these five, three share the same browser engine, the component responsible for rendering web pages and executing client-side JavaScript code. Table 4 presents the market share of each web browser alongside its underlying browser engine.

TABLE 4: MOST POPULAR DESKTOP WEB BROWSERS AND BROWSER ENGINES IN EUROPE

Web browser	Browser engine	Market share (%)
Chrome	Chromium	60.07
Edge	Chromium	12.64
Safari	WebKit	7.65
Firefox	Gecko	7.45
Opera	Chromium	4.74

[10] https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions/API/i18n

Our objective is to support all major Chromium-based web browsers. Collectively, these account for close to 80% of all desktop browsers. Firefox's Gecko engine provides the same WebExtensions API[11] as Chromium, and can therefore be supported with minimal additional effort (Hsu et al., 2024). Safari requires more work to support but may be more popular among certain target audiences, making it sensible to include as well. Moreover, by extending support to non-Chromium browsers, we ensure that the extension is accessible to a more diverse range of users.

6.3.1 PUBLICATION IN MAJOR APP STORES

Browser extensions are typically distributed via browser-specific app stores, allowing users to easily discover, search for, and install them with a single click. We intend to release the AI4Debunk browser extension on the app stores of the four largest web browsers: the Chrome Web Store, Microsoft Edge Add-Ons, Add-ons for Firefox, and the Apple App Store. In practice, this approach also enables support for other Chromium-based desktop browsers, such as Brave and Prisma, since all Chromium-based browsers can install extensions from the Chrome Web Store.

6.3.2 DEPLOYMENT IN MANAGED ENVIRONMENTS

Organisations wishing to use the AI4Debunk browser extension may have specific requirements. For instance, an organisation may wish to mandate the use of an internally hosted instance of the AI4Debunk debunking API, prevent users from changing certain settings, or block usage on internal company domains that may contain sensitive information.

The two most popular web browsers, Google Chrome and Microsoft Edge, can be deployed and managed in corporate environments by IT departments[12]. This enables all employees using company-managed devices to have immediate access to the extension, without needing to install or configure it themselves.

6.3.3 DEPLOYMENT OF CUSTOMISED BUILDS

The source code for the browser extension will be published on GitHub[13] under version 1.2 of the European Union Public License (EUPL-1.2). The EUPL is an OSI-approved free and open-source copyleft license designed to be legally robust within the European Union and compatible with other major open-source software licenses such as the GPL (Schmitz, 2013).

An official build will also be made available via GitHub, allowing individuals and organisations to manually install the extension outside of the app store – a practice known as sideloading. This method of installation is primarily intended for users who require a more extensively customised version of the AI4Debunk browser extension than is possible by default. They will be able to fork the repository and use the source code to build and deploy their own tailored version to suit their specific requirements.

[11] https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions
[12] https://support.google.com/chrome/a/answer/9296680
[13] https://github.com/AI4Debunk/extension

Funded by the European Union

7 DISCUSSION

Over the past year, two functional prototypes have been developed alongside working versions of the debunking API. These prototypes served two main purposes: first, to validate architectural design choices within the browser extension and its interaction with the API; and second, to provide demonstratable artefacts that showcase project progress, stimulate discussions about user needs, and gather early feedback on the proposed design, its usability, and overall system performance on real-world data.

Although this initial prototype is operational in a practical sense – it works – it is far from an accurate representation of the final product. For example, the current prototypes do not implement any form of authentication, making them unsuitable for public release. Furthermore, their design and technical implementation do not yet meet the standards expected by end-users and system operators.

In March 2026 (M27), official development of the browser extension will commence as part of WP11. This version will be developed in parallel with the smartphone app, the collaborative platform, and the VR app, each potentially targeting slightly different user groups, as indicated by the preliminary results of a survey conducted among diverse respondents that was presented during the first AI4Debunk multi-stakeholder task force meeting (Gaborit, personal communication, 2 February 2026). The new version will no longer be classified as a prototype but will initially be delivered as a minimum viable product (MVP), which will be iteratively enhanced until it meets the design specifications outlined in this report.

7.1 LIMITATIONS

Although the browser extension design incorporates best practices and findings from social sciences and humanities (SSH), several limitations remain unresolved. These include potential usability issues, technical constraints, and practical barriers that may affect performance in real-world scenarios.

7.1.1 RELIABILITY

For end-users, the reliability of the AI4Debunk browser extension encompasses two key aspects: the quality of its debunking outputs and the availability of the underlying AI4Debunk API.

By the conclusion of the project, the browser extension is expected to achieve Technology Readiness Level (TRL) 7, which requires a prototype to be demonstrated in an operational environment. This means that both the requirements and resulting design must reflect real-world usage scenarios, such as deployment in newsrooms or classrooms, or at home. Importantly, the TRL of the browser extension is partly dependent on readiness level of the debunking API and its subcomponents (Olechowski et al., 2020). This presents a challenge, as several subcomponents within AI4Debunk remain experimental and therefore have lower TRLs. Achieving the target readiness level will thus require ongoing validation and iterative refinement.

Funded by
the European Union

The quality of debunking results is primarily determined by the AI modules developed under WP6–9. The AI models selected for these modules have undergone rigorous benchmarking, and as a result, we expect that these provide state of the art performance. As newer, more advanced modules, are introduced, we expect that the quality of debunking results will gradually improve over time.

To guarantee service availability, the debunking API employs industry-standard tools and robust engineering practices (Lung & Van Der Bent, 2026), similar to those used by major news organisations such as the Nederlandse Omroep Stichting (NOS) in the Netherlands. These practices enable automated scaling of resources to handle unexpected demand and automated recovery mechanisms to mitigate system failures, thereby minimising downtime and ensuring consistent service delivery.

Collectively, these measures aim to provide users with a dependable experience – both in terms of accuracy of debunking results and the continuous availability of the AI4Debunk service.

7.1.2 USER PRIVACY

We implement measures in both the architecture and design of the browser extension to safeguard user privacy. For example, users must provide explicit consent before any submitted information is processed or stored for an extended period, and we avoid collecting information that is not strictly necessary.

However, certain privacy risks remain difficult to fully mitigate. One such risk is browser extension fingerprinting – a technique that enables malicious websites to uniquely identify users and infer sensitive information by detecting browser extensions installed in their browser (Karami et al., 2020). While best practices can reduce the likelihood of detection, it is currently impossible to eliminate this risk entirely.

Another inherent of open-sourcing software is the increased ease with which adversaries can publish malicious copycat versions in app stores, potentially introducing security vulnerabilities or stealing user data (Hsu et al., 2024). Possible mitigations include prohibiting the use of the AI4Debunk name in forks to reduce confusion and prominently linking the official browser extension on all project channels to steer users towards the genuine version.

7.2 FUTURE WORK

While the current design of the AI4Debunk browser extension is tailored to meet the needs of current mainstream European audiences, there remain several opportunities to enhance its functionality and increase its societal impact. Some of these potential directions may be explored further as part of WP 11 or future projects. We briefly outline each of these opportunities below.

7.2.1 LOCALISATION

During early development, the browser extension only supports three languages: English, Dutch, and Klingon. Eventually, the extension is intended to support all major languages spoken within the European

Funded by
the European Union

Union. Initially, we plan to translate the extension using a state-of-the-art large language model, as LLMs – unlike traditional machine translation software – can directly process raw translation files and "understand" the context of each translation string. Nevertheless, such LLM translations may still contain minor errors and unidiomatic phrasing that sound unnatural to native speakers. Including such translations could potentially trigger the horn effect, a cognitive bias that leads users to perceive the overall quality and trustworthiness of the system negatively. Therefore, we aim to collaborate with volunteers across Europe to review and correct translations before publication. The aim is to first support only languages spoken within the AI4Debunk consortium, and then gradually roll out support for other languages spoken in the EU, including dialects such as Catalan and Frisian, as well as languages spoken by foreign diaspora communities, such as Ukrainian and Turkish.

7.2.2 CONVERSATIONAL INTERFACE

Traditionally, information systems have been exposed to users through graphical user interfaces. However, conversational interfaces powered by large language models (LLMs) are now becoming increasingly prevalent. Notable examples include ChatGPT and Microsoft Copilot, which allow users to interact with systems using natural language and can be customised to suit individual preferences and needs.

Recent research indicates that conversations with LLMs can reduce belief in conspiracy theories by 20% on average (Costello et al., 2024). Building on this, we envisage that an LLM directly accessible from the extension – possibly embodied by a mascot such as NUTRU from AI4Debunk's VR app (Filippidou et al., 2026a) – could engage users in discussions about disinformation encountered on the current web page. The LLM would be instructed to draw exclusively on reliable sources, such as the knowledge graph developed as part of WP 6–7.

7.2.3 FACILITATING LONG-TERM MAINTENANCE

Yousuf et al. (2021) observed that many projects and services established in recent years to combat disinformation are no longer active or functioning. This means that the intended societal impact is not realised. Furthermore, even projects that remain operational will inevitably decline in quality and effectiveness unless they are actively maintained, in line with Lehman's seventh law of software evolution (Lehman, 1996).

A similar situation could arise with the browser extension after completion of the AI4Debunk project. To mitigate this risk, we recommend that WP11 proactively explores how agentic coding assistants can facilitate ongoing maintenance and further development of the AI4Debunk browser extension. White et al. (2024) have identified prompt patterns that enable automation of common software engineering tasks, including requirements elicitation, rapid prototyping, code quality assurance, deployment, and testing. Such patterns could be stored in a version-controlled text file alongside documentation detailing the intended use and functionality of the browser extension, as well as any standards it should follow, such as the ACM/IEEE-CS Software Engineering Code of Ethics (Gotterbarn et al., 1997). This approach would guide coding agents in future maintenance activities and support the sustainable development of the extension.

Funded by
the European Union

8 CONCLUSION

As part of the AI4Debunk project, we will develop three human-centred debunking interfaces. This report reviews prior work on browser extensions designed to combat online disinformation and highlights the distinctive aspects of our approach. We also provide an initial overview of the browser extension's user interface design and outline the underlying system architecture required to support it. Finally, we identify the key challenges that must be addressed in WP11 over the coming months to ensure the browser extension fulfils its intended purpose.

Funded by
the European Union

REFERENCES

Andrew, S., & Tigwell, G. W. (2025). Understanding the Experiences of People With and Without Vision Impairments When Using Mobile User Interface Alternative Color Modes. *Proceedings of the ACM on Human-Computer Interaction*, *9*(5), 1–25. https://doi.org/10.1145/3743704

Bartley, N., Abeliuk, A., Ferrara, E., & Lerman, K. (2021). Auditing Algorithmic Bias on Twitter. *13th ACM Web Science Conference 2021*, 65–73. https://doi.org/10.1145/3447535.3462491

Berretti, S., & Caldelli, R. (2025). *Report on requirements* (D5.3; AI4Debunk).

Bogner, J., & Merkel, M. (2022). To Type or Not to Type? A Systematic Comparison of the Software Quality of JavaScript and TypeScript Applications on GitHub. *Proceedings of the 19th International Conference on Mining Software Repositories*, 658–669. https://doi.org/10.1145/3524842.3528454

Bontcheva, K., Papadopoulous, S., Tsalakanidou, F., Gallotti, R., Dutkiewicz, L., Krack, N., Teyssou, D., Severio Nucci, F., Spangenberg, J., Srba, I., Aichroth, P., Cuccovillo, L., & Verdoliva, L. (2024). Generative AI and disinformation: Recent advances, challenges, and opportunities. *European Digital Media Observatory*.

Botnevik, B., Sakariassen, E., & Setty, V. (2020). BRENDA: Browser Extension for Fake News Detection. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2117–2120. https://doi.org/10.1145/3397271.3401396

Chakraborty, A., Paranjape, B., Kakarla, S., & Ganguly, N. (2016). Stop Clickbait: Detecting and preventing clickbaits in online news media. *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 9–16. https://doi.org/10.1109/ASONAM.2016.7752207

Costello, T. H., Pennycook, G., & Rand, D. G. (2024). Durably reducing conspiracy beliefs through dialogues with AI. *Science*, *385*(6714), eadq1814. https://doi.org/10.1126/science.adq1814

Dalpiaz, F., & Sturm, A. (2020). Conceptualizing Requirements Using User Stories and Use Cases: A Controlled Experiment. In N. Madhavji, L. Pasquale, A. Ferrari, & S. Gnesi (Eds), *Requirements Engineering: Foundation for Software Quality* (Vol. 12045, pp. 221–238). Springer International Publishing. https://doi.org/10.1007/978-3-030-44429-7_16

D'Ignazio, C. (2024). *Counting Feminicide: Data Feminism in Action*. MIT Press.

Filippidou, D. E., Karanasios, G., Katsaridis, S., Katsakioris, D., Nikas, M., Simeonidou, A., Nikopoulos, G., & Maragkos, C. (2026a). *Report on the definition of the AR/VR environments applications* (D10.5; AI4Debunk). DOTSOFT.

Filippidou, D. E., Karanasios, G., Katsaridis, S., Katsakioris, D., Nikas, M., Simeonidou, A., Nikopoulos, G., & Maragkos, C. (2026b). *Report on the definition of the smartphone app* (D10.3; AI4Debunk). DOTSOFT.

Gaborit, P., & Martinsen, J. (2025). *Possible impacts of the tool on the perceptions of the citizens and social media users* (D12.1; AI4Debunk).

Geewax, J. J. (2021). *API design patterns*. Simon and Schuster.

Gotterbarn, D., Miller, K., & Rogerson, S. (1997). Software engineering code of ethics. *Communications of the ACM*, *40*(11), 110–118. https://doi.org/10.1145/265684.265699

Hameleers, M. (2025). The Nature of Visual Disinformation Online: A Qualitative Content Analysis of Alternative and Social Media in the Netherlands. *Political Communication*, *42*(1), 108–126. https://doi.org/10.1080/10584609.2024.2354389

Hsu, S., Tran, M., & Fass, A. (2024). What is in the Chrome Web Store? *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, 785–798. https://doi.org/10.1145/3634737.3637636

ISO/IEC. (2023). *ISO/IEC 25010:2023 Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—Product quality model* (ISO/IEC 25010:2023). ISO/IEC.

ISO/IEC. (2024). *ISO/IEC 25002:2024 Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Quality model overview and usage* (ISO/IEC 25002:2024). ISO/IEC.

Jahanbakhsh, F., & Karger, D. R. (2024). A Browser Extension for in-place Signaling and Assessment of Misinformation. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 1–21. https://doi.org/10.1145/3613904.3642473

Jahanbakhsh, F., Zhang, A. X., Karahalios, K., & Karger, D. R. (2022). Our Browser Extension Lets Readers Change the Headlines on News Articles, and You Won't Believe What They Did! *Proceedings of the ACM on Human-Computer Interaction*, *6*(CSCW2), 1–33. https://doi.org/10.1145/3555643

Jin, B., Li, H., & Zou, Y. (2025). Impact of extensions on browser performance: An empirical study on Google Chrome. *Empirical Software Engineering*, *30*(4), 103. https://doi.org/10.1007/s10664-025-10633-1

Karami, S., Ilia, P., Solomos, K., & Polakis, J. (2020). Carnus: Exploring the Privacy Threats of Browser Extension Fingerprinting. *Proceedings 2020 Network and Distributed System Security Symposium*. Network and Distributed System Security Symposium, San Diego, CA. https://doi.org/10.14722/ndss.2020.24383

Kozik, R., Pawlicka, A., Pawlicki, M., & Choraś, M. (2024). From Detection Through Display to Understanding: Bridging AI and UI in Disinformation and Fake News Analysis. In N.-T. Nguyen, B. Franczyk, A. Ludwig, M. Nunez, J. Treur, G. Vossen, & A. Kozierkiewicz (Eds), *Advances in Computational Collective Intelligence* (Vol. 2165, pp. 347–357). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-70248-8_27

Kragt, J., & Keijzer, M. (2026). *Report on the definition of the collaborative platform* (D10.4; AI4Debunk). Innovative Power.

Lehman, M. M. (1996). Laws of software evolution revisited. In C. Montangero (Ed.), *Software Process Technology* (Vol. 1149, pp. 108–124). Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0017737

Lucassen, G., Dalpiaz, F., Werf, J. M. E. M. V. D., & Brinkkemper, S. (2016). The Use and Effectiveness of User Stories in Practice. In M. Daneva & O. Pastor (Eds), *Requirements Engineering: Foundation for Software Quality* (Vol. 9619, pp. 205–222). Springer International Publishing. https://doi.org/10.1007/978-3-319-30282-9_14

Lung, C. F. (2021). *A search for the Ten Commentments: An exploratory study on automated quality assessment of comments in Java source code* [Open University of the Netherlands]. https://research.ou.nl/en/studentTheses/a-search-for-the-ten-commentments-an-exploratory-study-on-automat

Lung, C. F., & Van Der Bent, J. F. (2026). *Report on the definition of the debunking API* (D10.1; AI4Debunk).

Mavin, A., Wilkinson, P., Teufl, S., Femmer, H., Eckhardt, J., & Mund, J. (2017). Does goal-oriented requirements engineering achieve its goal? *2017 IEEE 25th International Requirements Engineering Conference (RE)*, 174–183.

MDN Web Docs. (2025a). *Browser extensions—Mozilla | MDN*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions

MDN Web Docs. (2025b). *Plugin—Glossary | MDN*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Glossary/Plugin

MDN Web Docs. (2025c). *prefers-reduced-motion—CSS | MDN*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/CSS/@media/prefers-reduced-motion

Olechowski, A. L., Eppinger, S. D., Joglekar, N., & Tomaschek, K. (2020). Technology readiness levels: Shortcomings and improvement opportunities. *Systems Engineering*, *23*(4), 395–408. https://doi.org/10.1002/sys.21533

Ollila, R., Mäkitalo, N., & Mikkonen, T. (2022). Modern Web Frameworks: A Comparison of Rendering Performance. *Journal of Web Engineering*. https://doi.org/10.13052/jwe1540-9589.21311

Ozoliņa, Ž., Struberga, S., Šteinbuka, I., Zeibote, Z., Gaborit, P., Martinsen, J., Rao, V., Shcherba, D., Polischuk, K., Hryshko, A., D'Andrea, A., & D'Ulizia, A. (2025a). *Disinformation target groups in the EU member states, sources, and hosts of propaganda* (D5.1; AI4Debunk).

Ozoliņa, Ž., Struberga, S., Šteinbuka, I., Zeibote, Z., Gaborit, P., Martinsen, J., Rao, V., Shcherba, D., Polischuk, K., Hryshko, A., D'Andrea, A., & D'Ulizia, A. (2025b). *Narratives and foreign interference throughout Europe illustrated by case studies* (D5.2; AI4Debunk).

Pedemonte, G., Leotta, M., & Ribaudo, M. (2025). Improving Web Accessibility With an LLM-Based Tool: A Preliminary Evaluation for STEM Images. *IEEE Access*, *13*, 107566–107582. https://doi.org/10.1109/ACCESS.2025.3577519

Pilati, F., & Venturini, T. (2025). The use of artificial intelligence in counter-disinformation: A world wide (web) mapping. *Frontiers in Political Science*, *7*, 1517726. https://doi.org/10.3389/fpos.2025.1517726

Rani, P., Zellweger, J., Kousadianos, V., Cruz, L., Kehrer, T., & Bacchelli, A. (2024). Energy Patterns for Web: An Exploratory Study. *Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Society*, 12–22. https://doi.org/10.1145/3639475.3640110

Schmitz, P.-E. (2013). The European Union Public Licence (EUPL). *International Free and Open Source Software Law Review*, *5*(2), 121–136. https://doi.org/10.5033/ifosslr.v5i2.91

Shams, A. B., Hoque Apu, E., Rahman, A., Sarker Raihan, M. M., Siddika, N., Preo, R. B., Hussein, M. R., Mostari, S., & Kabir, R. (2021). Web Search Engine Misinformation Notifier Extension (SEMiNExt): A Machine Learning Based Approach during COVID-19 Pandemic. *Healthcare*, *9*(2), 156. https://doi.org/10.3390/healthcare9020156

Starov, O., & Nikiforakis, N. (2018). PrivacyMeter: Designing and Developing a Privacy-Preserving Browser Extension. In M. Payer, A. Rashid, & J. M. Such (Eds), *Engineering Secure Software and Systems* (pp. 77–95). Springer International Publishing. https://doi.org/10.1007/978-3-319-94496-8_6

StatCounter. (2026). *Desktop Browser Market Share Europe*. StatCounter Global Stats. https://gs.statcounter.com/browser-market-share/desktop/europe

Funded by the European Union

Van Vliet, H. (2008). *Software engineering: Principles and practice* (Vol. 13). John Wiley & Sons Hoboken, NJ.

White, J., Hays, S., Fu, Q., Spencer-Smith, J., & Schmidt, D. C. (2024). ChatGPT Prompt Patterns for Improving Code Quality, Refactoring, Requirements Elicitation, and Software Design. In A. Nguyen-Duc, P. Abrahamsson, & F. Khomh (Eds), *Generative AI for Effective Software Development* (pp. 71–108). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-55642-5_4

Yousuf, B., Qureshi, M. A., Spillane, B., Munnelly, G., Carroll, O., Runswick, M., Park, K., Culloty, E., Conlan, O., & Suiter, J. (2021). *PROVENANCE: An Intermediary-Free Solution for Digital Content Verification* (arXiv:2111.08791). arXiv. https://doi.org/10.48550/arXiv.2111.08791

Zavolokina, L., Sprenkamp, K., Katashinskaya, Z., Jones, D. G., & Schwabe, G. (2024). Think Fast, Think Slow, Think Critical: Designing an Automated Propaganda Detection Tool. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 1–24. https://doi.org/10.1145/3613904.3642805

Zhao, X. (2023). Leveraging Artificial Intelligence (AI) Technology for English Writing: Introducing Wordtune as a Digital Writing Assistant for EFL Writers. *RELC Journal*, *54*(3), 890–894. https://doi.org/10.1177/00336882221094089

Funded by
the European Union

A KEY RECOMMENDATIONS FOR TOOL DEVELOPERS

The work done in WP10 is preceded by several social sciences and humanities (SSH) work packages, which produced key recommendations for developers of tools to combat disinformation. To ensure that these recommendations are properly integrated into the design and development of human-centred user interfaces, the list below was originally compiled by Pascaline Gaborit (Pilot4DEV) and later organised and numbered by Marcel Keijzer (IP). These recommendations were subsequently discussed at various consortium meetings. This appendix presents all recommendations in their original form.

---

1. User-centric design
   - 1.1. Design intuitive interfaces that make navigation and interpretation of data simple for both technical and non-technical users (including young and elderly people). Include tooltips, visual aids (graphs, videos).
   - 1.2. Enable customization: Users should tailor analysis depth, alerts, and display preferences. Provide “basic” and “advanced” modes to cater to diverse audiences.
   - 1.3. Provide comprehensive user support through integrated FAQs, contextual help, and multimedia learning aids (tutorials, videos, interactive walkthroughs).
2. Gender equality and inclusivity
   - 2.1. Ensure gender-neutral design and eliminate bias in algorithms and datasets. Conduct regular gender audits to test for bias in detection outcomes.
   - 2.2. Also make sure that all vulnerable groups can access the tool and interface (people living in remote areas or in the countryside, elderly people, people from different minorities, e.g. Russian speaking diaspora but not only…)
   - 2.3. Collaborate with gender experts and advocacy groups during design and testing to ensure the tool serves all genders equitably.
   - 2.4. Support inclusivity: Offer multi-language interfaces, culturally neutral symbols, and accessibility features (screen readers, keyboard navigation, voice commands).
   - 2.5. Comply with global accessibility standards and optimize tools for low-tech or low-bandwidth environments.
   - 2.6. Use region-specific data and partner with local experts to make tools sensitive to cultural and linguistic disinformation trends.
3. Integration with social media
   - 3.1. Implement API-based real-time analysis for major platforms (Facebook, X/Twitter, Bluesky, TikTok, YouTube, Instagram).
   - 3.2. Ensure cross-platform tracking to trace how disinformation spreads across ecosystems.
   - 3.3. Access metadata (timestamps, geolocation, engagement) to understand virality and amplification dynamics while maintaining GDPR-compliant privacy protections.
   - 3.4. Monitor emerging and niche platforms (e.g., Telegram, Gab, Parler) and adapt to new platform features like short-lived content or encrypted messaging.
4. Tackling coordinated inauthentic behaviour (CIB)

Funded by the European Union

4.1. Provide real-time alerts and dashboards that visualize spikes in engagement, bot activity, or suspicious hashtag use.
4.2. Educate users through built-in guidance on recognizing coordinated behaviour (e.g., identical posts, synchronized activity).
4.3. Integrate network visualization tools to help users identify clusters and influencers driving disinformation.
4.4. Allow user-driven monitoring and flagging, enabling journalists and analysts to investigate specific narratives and provide corrective feedback.

5. Ethical and transparent AI
   5.1. Prioritize data privacy and user consent. Collect and store only necessary public data, ensuring transparency about use and limitations.
   5.2. Avoid algorithmic bias by maintaining diverse, balanced datasets and documenting model decision-making processes.
   5.3. Provide transparency in flagging: Explain why content is flagged and offer mechanisms for users to dispute or verify flagged material.
   5.4. Establish ethical safeguards to prevent misuse of detection tools for censorship or political manipulation.
6. Multilingual and cultural adaptation
   6.1. Train NLP models per language to recognize regional slang, idioms, and context-sensitive disinformation patterns.
   6.2. Enable cross-language analysis to track narrative migration across languages and regions.
   6.3. Use culturally aware machine translation for languages without native models, ensuring meaning and intent are preserved.
7. Stakeholder and expert engagement
   7.1. Establish regular consultation mechanisms (workshops, focus groups, advisory boards) with journalists, researchers, fact-checkers, and policymakers.
   7.2. Run diverse beta testing programs involving participants from different cultural, linguistic, and professional backgrounds.
   7.3. Continuously integrate feedback from stakeholders into iterative tool updates.
8. Continuous improvement
   8.1. Update algorithms regularly to reflect new disinformation tactics (deepfakes, AI-generated content, bot evolution).
   8.2. Expand and diversify datasets using regional and topical sources, verified fact-checking data, and emerging content types.
   8.3. Maintain feedback loops: allow users to report false positives or missed detections and receive updates on actions taken.
   8.4. Collaborate with experts in AI, disinformation, and digital ethics to ensure tools evolve responsibly and effectively.
9. Overall recommendation
   9.1. Developers should approach disinformation detection as a living process, emphasizing adaptability, transparency, inclusivity, and collaboration. By integrating ethical AI design, user engagement, and continuous learning, tools can remain robust against evolving disinformation landscapes and build user trust globally.

Funded by
the European Union

Review Sheet of Deliverable/ Milestone Report
D10.2 Deliverable Title

Editor(s):	Franc van der Bent (HU) Chun Fei Lung (HU)
Responsible Partner:	Hogeschool Utrecht (HU)
Status-Version:	Draft – v0.3
Date:	09/02/2026
Distribution level (CO, PU):	Public
Reviewer (Name/Organization)	Georgi Gotev, Kalina Angelova (EUalive)
Review date	17/02/2026

Disclaimer: This assessment reflects only the author's views and the European Commission is not responsible for any use that may be made of the information contained therein.

Funded by
the European Union

Mark with X the corresponding column:

Y= yes	N= no	N = not applicable

ELEMENT TO REVIEW	Y	N	NA	COMMENTS
FORMAT: Does the document … ?				
…include editors, deliverable name, version number, dissemination level, date, and status?	y			
…contain a license (in case of public deliverables)?	Y			
…include the names of contributors and reviewers?	Y			
….have a version table consistent with the document's revision?	Y			
… contain an updated table of contents?	Y			Includes working hyperlinks
… contain a list of figures consistent with the document's content?	Y			
… contain a list of tables consistent with the document's content?	Y			
… contain a list of terms and abbreviations?	Y			
… contain an Executive Summary?	Y			
… contain a Conclusions section?	Y			
… contain a List of References (Bibliography) in the adequate format, if relevant?	Y			
… use the fonts and sections defined in the official template?	Y			
… use correct spelling and grammar?	Y			
… conform to length guidelines (50 pages maximum (plus Executive Summary and annexes)	Y			
… conform to guidelines regarding Annexes (inclusion of complementary information)			NA	
… present consistency along the whole document in terms of English quality/style? (to avoid accidental usage of copy-pasted text)	Y			
About the content…				
ELEMENT TO REVIEW	**Y**	**N**	**NA**	**COMMENTS**
Is the overall style of the deliverable correctly organized and presented in a logical order?	Y			
Is the Executive Summary self-contained, following the guidelines and does it include the main conclusions of the document?	Y			

Funded by the European Union

ELEMENT TO REVIEW	Y	N	NA	COMMENTS
Is the body of the deliverable (technique, methodology results, discussion) well enough explained?	Y			
Are the contents of the document treated with the required depth?	Y			
Does the document need additional sections to be considered complete?		N		
Are there any sections in the document that should be removed?		N		
Are all references in the document included in the references list?	Y			
Have you noticed any text in the document not well referenced? (copy and paste of text/picture without including the reference in the reference list)		N		
SOCIAL and TECHNICAL RESEARCH WPs (WP4, 5, 12, 13, 14)				
ELEMENT TO REVIEW	**Y**	**N**	**NA**	**COMMENTS**
Is the deliverable sufficiently innovative?	Y			
Does the document present technical soundness and its methods are correctly explained?	Y			
What do you think is the strongest aspect of the deliverable?				The deliverable presents a strong background and research on browser extensions against disinformation.
What do you think is the weakest aspect of the deliverable?				A structured needs assessment of end users could be beneficial.
Please perform a brief evaluation and/or validation of the results, if applicable.			NA	
AI AND TECNOLOGICAL WPS (WP6 – WP11)				
ELEMENT TO REVIEW	**Y**	**N**	**NA**	**COMMENTS**
Does the document present technical soundness and the methods are correctly explained?	Y			
What do you think is the strongest aspect of the deliverable?				Rigorous benchmarking, learning from other AI tools.
What do you think is the weakest aspect of the deliverable?				Information on training data sources could be added.
Please perform a brief evaluation and/or validation of the results, if applicable.			NA	
DISSEMINATION AND EXPLOITATION WPs (WP15 – WP17)				

Funded by the European Union

ELEMENT TO REVIEW	Y	N	NA	COMMENTS
ELEMENT TO REVIEW	Y	N	NA	COMMENTS
Does the document present a consistent outreach and exploitation strategy?	Y			
Are the methods and means correctly explained?	Y			
What do you think is the strongest aspect of the deliverable?				The browser extension complements the collaborative platform. It also introduces developer options for advanced users and an auto filling form, which is useful for regular users. In more general terms, the alignment with pressing societal challenges.
What do you think is the weakest aspect of the deliverable?				
Please perform a brief evaluation and/or validation of the results, if applicable.			NA	
DISSEMINATION AND EXPLOITATION WPs (WP18)				
ELEMENT TO REVIEW	**Y**	**N**	**NA**	**COMMENTS**
Does the document present the main ethical aspects regarding the use of methods and human involvement?	Y			
What do you think is the strongest aspect of the deliverable?				User friendly interface.
What do you think is the weakest aspect of the deliverable?				
Please perform a brief evaluation and/or validation of the results, if applicable.			NA	

SUGGESTED IMPROVEMENTS

PAGE	SECTION	SUGGESTED IMPROVEMENT
		ADD ROWS AS NECESSARY

CONCLUSION

Mark with X the corresponding line.

X	Document accepted, no changes required.
	Document accepted, changes required.
	Document not accepted, it must be reviewed after changes are implemented.

Please rank this document globally on a scale of 1-5 (1 = poor, 5= excellent) – using a half point scale. Mark with X the corresponding grade.

Funded by the European Union

Document grade	1	1.5	2	2.5	3	3.5	4	4.5	5
									X