THE INFORMATION PRESENTER IN THE CONTEXT OF EDUCATION: HTML STUDY NOTES AND AUTOMATION FOR TEACHERS

Miroslav Bonchev Bonchev
"St. Cyril and St. Methodius" University of Veliko Tarnovo, Bulgaria
miroslav.bonchev@proton.me; m.bonchev@ts.uni-vt.bg

Abstract

The Information Presenter is a combined online–desktop software system that delivers structured educational content through a mnemonic matrix supporting learning, retention, gap identification, and conceptual understanding. This paper presents a new feature: the ability to upload and index externally created study notes in styled HTML format. This functionality enables educators to build rich information libraries quickly and efficiently, while students continue to benefit from manual note creation that supports active learning. The flexibility of using external tools, including AI-based assistants, offers clear advantages for teachers, but also raises challenges when applied to student work.

Keywords: *Information Presenter; Educational Software; HTML Study Notes; Automated Content Creation; Active Learning; Content Security.*

INTRODUCTION

The Information Presenter (IP) is a comprehensive educational software system for learning. Schools and universities can use it to support their students in the learning process, self-learning, understanding, and retention of material. The software uses an Integrated Approach, including the following elements:

1. Special presentation modes of operation (IP-1)
2. Multidimensional classified information units (study notes) (IP-2)
3. Comprehensive learning processes integrating IP (IP-3)

The philosophy of the Information Presenter consists of continuous learning through several methods, including:

- Enrichment with new knowledge through passive, semi-active, and active means;
- Maintenance of already acquired knowledge.

The Information Presenter has already been examined in detail regarding its architecture [1] and educational methodology [2]. This article expands upon that work by introducing HTML study notes as a new system feature.

The learning methods are achieved through specialized software, part of IP, designed to provide appropriate presentation modes, creation and use of classified study notes, and comprehensive learning processes. The software can store and provide easy access to all educational material taught to students throughout their years in the educational institution.

Components of the Information Presenter

The Information Presenter consists of an online library (called IP-lib) and a desktop application (called IP-app). IP-app provides information to users which it retrieves from IP-lib. A customized edition is created for each school that wishes to use the software to raise the

knowledge level of its students. The customization encompasses the entire product, so that the identity and specific capabilities of each edition are closely tied to the organization for which it was created. Thus, each edition has its own online information library (IP-lib) and its own desktop application (IP-app) connected to it.

System Architecture and Workflow

The overall functional architecture of the software and its workflows is shown in Figure 1. The system components are depicted with icons, and additional annotations clarify their meaning. The arrows and numbers indicate the flow of information and interactions between them.

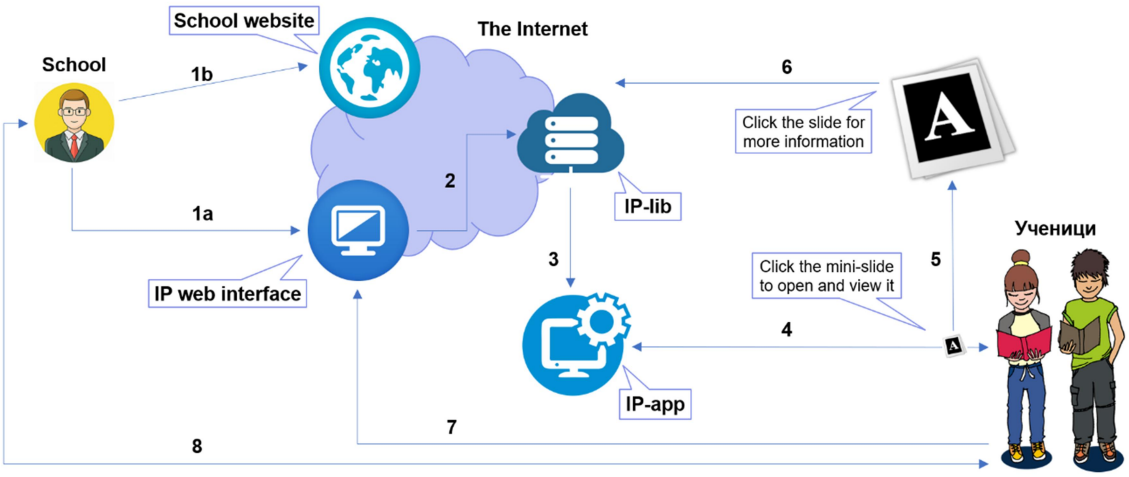

Fig. 1. Architecture and Workflow of the Information Presenter

The main system components are as follows:

- **School**: This is the educational institution that has its own edition of the software.
- **Website**: This is the website of the educational institution when they have one. It is completely independent of the Information Presenter.
- **Students**: These are the students of the educational institution.
- **Library**: This is the IP-lib database that contains the information displayed by the Information Presenter to students.
- **IP web Interface**: This is the interface through which teachers and students manage the information in the IP-lib database, for example adding, editing, deleting, etc.
- **Desktop Application**: This is the IP-app desktop application that students use to draw knowledge from the library.

Special Presentation Modes of Operation and IP-app

IP-app is key to realizing the Information Presenter's philosophy of continuous student learning. It implements the first element of the software's Integrated Approach to learning, IP-1: Special presentation modes of operation.

The presentation modes of operation of IP-app attract students to "check" the loaded information. The ease of viewing and following information engages the students to familiarize themselves with it. However, the quality of the presented information and the comprehensiveness of the teaching processes are of crucial importance as well. They are the ones that determine if the user will use IP, and thus acquire new knowledge through it.

The special presentation modes of operation of IP-app are achieved through a balance of mutually contradictory properties and characteristics:

- **Visibility:**
  - Always visible showing some information from the library, but
  - Does not interfere with the visibility of other applications.

The always-on visibility enables users to see the loaded information with a glance, and thus "glance" the entire library. However, it does not prevent or obstruct users from using their computers for other purposes at the same time without juggling windows. This is an important advantage compared to using search engines, artificial intelligence, websites, PDFs and other electronic media, which require the user to be "reminded" of the information in order to search for it at all.

- **Operability:**
  - Always ready for the next user command, but
  - Does not interfere with the user's work with other applications.

This feature enables the user to instantly control IP and the loaded information. This is an important difference and advantage compared to using paper media, websites, PDFs and other electronic media, which require time and effort to access the information.

- **Attractiveness:**
  - Attractive, but
  - Does not distract the user's attention.

This includes the attractiveness of both IP-app and the information it displays. While IP-app is developed to work beautifully and elegantly, its visible user interface is minimal and the attractiveness of the displayed information is determined by the presented information itself. Since teachers and students create the displayed information, IP becomes also an excellent tool for self-training in this direction.

- **Informativeness:**
  - Display the currently loaded study note, as well as
  - Provides timely system information about errors, but
  - Does not distract the user from their work.

By nature, IP-app is maximally informative, as it is always visible showing the currently loaded information. In case of an error, it does not interrupt the user. Instead, IP-app displays the error when the user opens the currently loaded study note. Thus, IP is itself optimally informative and unobtrusive. The information that IP-app displays comes from a user-selected IP-app gallery. Depending on the type of the information, IP-app uses different, optimized for its nature, display windows to visualize it.

1. Images are displayed in a specialized view for visual information.
2. Video is displayed in a video window.
3. Documents consisting of text, images, video, audio, etc. are displayed in a view capable of showing such information.

- Presence:
  - Always available for quick learning of something, but
  - Never intrusive or annoying to the user.

IP-app is always visible, periodically loading new information from the library. When loading such information, it uses a pleasant smooth transformation from the old to the new information that attracts attention but does not distract it. In this way, the user understands that new information has been loaded, but is not compelled to view it and can ignore it without

paying any attention to it. On the other hand, they can construct an idea of it, and if it is of interest, they can open and examine it in detail with a single mouse or keyboard command.

Multidimensional Classified Units of Information

The second element of the Integrated Approach to learning in the software is IP-2: the use of multidimensional classified units of information—study notes.

Definition: A **study note** is a note that contains one or more educational facts and has characteristics and properties that enable their tracking and understanding.

Teachers and students create study notes in the school's online library via the provided web interface. The software operates with three types of study notes:

1. **Images** – for visual information.
2. **Video** – for video information, recorded lessons, animations, etc.
3. **Documents** – for combined information: text, images, audio, and video.

Study notes have mandatory properties such as a title and description, and optional properties such as links to external resources for additional information. In addition, they may have assigned meta-properties, such as author, composer, material, date of creation, or any other information applicable to the content of the study note. Meta-properties are defined by the creators of the study notes as needed.

Study notes are classified into galleries according to topics and other criteria. School administrators, teachers, and students manage and build the IP-lib through its web interface. There, they have access only to the parts of the library for which they have the appropriate permissions. At the same time, IP-lib allows the creation of viewer groups that are assigned to galleries for visibility by the IP-app users. Thus, on the one hand, users of IP-lib (study note creators) have visibility rights within the web interface, and on the other hand, users of IP-app (students) are organized into groups so that they see only content appropriate for them. For example, a geography teacher may create study notes via the web interface only in their assigned geography galleries, while students from class 7A–Mathematics can view only the content intended for them, but not the content intended for students from class 10E–Biology.

IP-app provides students using computers with the opportunity to view various study notes from their curriculum at any time. When opening the currently loaded study note, the student can easily recognize a gap in specific knowledge and acquire it from the note, or alternatively reinforce existing knowledge. On the other hand, the creation of study notes by students is an even more important method for acquiring knowledge and skills, including the development of active literacy.

Management of IP-lib

The library of study notes of the Information Presenter plays a key role in realizing the philosophy of continuous student learning supported by the Information Presenter. IP-lib is managed by administrators organized hierarchically. This structure allows the creation, editing, and deletion of administrator accounts with different levels of rights and responsibilities.

- **Super-administrators**: They have full control over all aspects of IP-lib. This includes management of content, users, administrators, system settings, and platform security.

- **Standard administrators**: These admins have assigned one or more administrative roles. From a content perspective, the most important roles are the following:
  - **Content Authority (CA) – content administrators**: These administrators have the authority to create and publish study notes in their assigned galleries. The CA role is typically assigned to teachers, allowing them to control their subject area within the library. In addition, CA administrators have the right to create Private Content Only Providers (PCOP). PCOP administrators can create study notes only in hidden galleries belonging to the CA administrator who granted them these rights. PCOP administrators are typically students, which enables them to create study notes as part of the educational process. For example:
    - Teachers grant PCOP rights on hidden galleries to their students.
    - Students create study notes as homework or course assignments.
    - Teachers review the study notes created by students and, if they are correct and well designed, publish them in public galleries.
  - **Viewers management**: These administrators have the rights to create, remove viewer and manage groups and viewers (IP-app users). In schools, this role is usually assigned to staff members who maintain student and class/group records.

Comprehensive Learning Processes Integrating the Information Presenter

The third key element in the realization of the Information Presenter's philosophy of continuous student learning is IP-3: comprehensive learning processes and the integration of IP software into them. In essence, this represents the manner in which the online library IP-lib is built and maintained.

The Information Presenter is delivered to organizations with an empty online library. Each school decides how to organize the structure of its online library, and how to develop it so that it corresponds to the institution's educational system.

An online library whose hierarchy exactly follows the educational structure of the institution is referred to as **momentarily isomorphic**. When the hierarchy of the online library follows the structure of the educational institution for each academic year and for each individual class (group), it is referred to as **fully isomorphic**. An online library whose hierarchy does not follow the educational structure of the institution is referred to as **non-isomorphic**.

Effectiveness of Study Notes:

The effectiveness of study notes can be evaluated through several key factors:

1. **Number of study notes**: A larger number of study notes is a prerequisite for broader and more diverse information and learning resources.
2. **Completeness of content**: A balance between completeness and sufficiency of information within each study note is essential for its usefulness.
3. **Richness of content**: Content-rich study notes employ diverse media formats such as text, images, and video.
4. **Diversity of topics**: Broad coverage of topics and disciplines is important for maintaining student interest.

Learning Processes and the Construction of IP-lib

The correspondence between learning processes and the construction of the online library IP-lib can be examined along two main dimensions.

1. **Source of Study Notes**:
    a. **Principled source**: In this model, study notes are created by current students as part of the learning process. This encourages active participation and engagement, as students themselves build the online library and their knowledge. Teachers participate in the process by guiding it and ensuring the correctness and quality of the study notes.
    b. **Non-principled source**: In this model, study notes are created by someone other than the students of the current academic year—for example, by the teacher or by students from previous years.
    c. **Mixed source**: In this case, study notes are created both by non-principled sources and by current students.

** There are unresolved ethical and other issues related to the use of artificial intelligence in the creation of study notes by students. These issues are beyond the scope of the present paper but will be analyzed in future work.*

2. **Temporal Availability of Study Notes**:
    a. **Permanent availability**: Study notes are always accessible to viewers.
    b. **Synchronized availability**: Study notes are accessible only temporarily.
        i. **Partial availability**: Study notes are accessible to viewers from the time the material is taught until the end of the examination period.
        ii. **Comprehensive availability**: Study notes are accessible to viewers from the time the material is taught until the completion of education or for several years thereafter.

Completeness of IP-lib

Maximum completeness of the Information Presenter for students can be achieved by applying the following approach:

1. **Fully isomorphic structure of IP-lib**: The online library IP-lib strictly follows the educational structure of the school. The study notes for each class (group) remain available in the system until the completion of the respective class.
2. **Synchronized–comprehensive availability**: Study notes are added to IP-lib as the material is being taught. They remain available until students complete their education or for a defined period afterward.
3. **Principled source**: Students create the study notes that they themselves use as part of the learning process. Teachers review and approve them before publication. This ensures quality control while actively engaging students in the creation of learning materials. The process includes one or more of the following practices:
    a. Homework assignments.
    b. Course projects.
    c. Extracurricular activities.

EXPOSITION

The creation of image- and video-based study notes through the IP-lib interface is fast and efficient. The user provides:

1. the title of the study note,
2. a description of the study note,
3. optional links for additional information and meta-properties,
4. the image or video associated with the study note.

The creation of document-based study notes is more labor-intensive, as they must be both informative and visually appealing. Document-based study notes created through the IP-lib web interface represent implicit tables in which each cell must be defined, formatted, and populated with content. While the interface is easy to use, building a well-designed and aesthetically pleasing study note can require substantial effort. Thus, constructing document-based study notes using the built-in document builder may be inefficient for users experienced in creating HTML pages. For this reason, the author added the ability to directly upload pre-built HTML pages into IP-lib, which are then displayed by IP-app as standard document-based study notes.

The Information Presenter was designed with an architecture that allows such extensions. However, a significant number of factors related to system integration and security must be taken into account.

1. User-uploaded HTML study notes must have the same properties and characteristics as document-based study notes created through the IP-lib builder, including:
    a. title,
    b. description,
    c. optional external links,
    d. optional meta-properties,
    e. window settings for display in IP-app,
    f. display ordering integrated with other study notes,
    g. indexing within the system,
    h. maximum allowed sizes for the study note and for its components, etc.
2. User-uploaded HTML study notes must be presented to users and operate in the same manner as study notes created through the IP-lib builder, including:
    a. display in galleries,
    b. editing,
    c. cloning,
    d. copying,
    e. moving,
    f. deletion, and related operations.
3. User-uploaded HTML study notes must be displayed exactly as their creators expect them to appear in a standard web browser.
4. User-uploaded HTML study notes must be protected by IP-lib against unauthorized extraction of information, in the same way as study notes created directly within IP-lib.
5. User-uploaded HTML study notes must not allow malicious content that could cause harm to IP-app users (students).
6. For security and other reasons, the system includes a dedicated permission for creating user-uploaded HTML study notes, so that only administrators explicitly granted this permission can upload such content.

7. The system must also be flexible, allowing easy external management of the types of content permitted to be attached to HTML study notes—for example, configurable limits on file sizes and allowed formats for images, video, audio, and subtitles.

User-uploaded HTML study notes are created in a manner similar to other types of study notes. The user first selects the gallery in which the study note will be created. Next, the user chooses the menu option for creating a new document-based study note via upload. The user then fills in the required information and uploads the necessary files: the HTML file, the CSS file, and the icon displayed by IP-app when the study note is closed, as well as the images, audio, video, and subtitle files referenced by the HTML document. The upload interface for HTML study notes is shown in Figure 2.

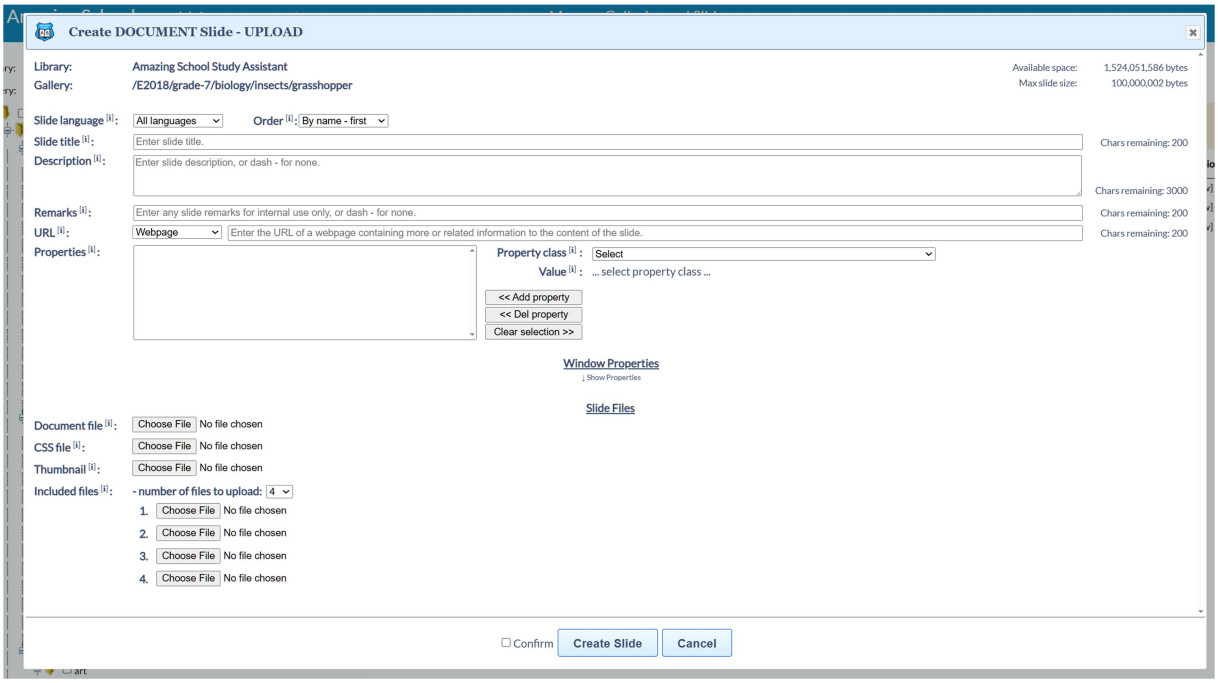

Fig. 2. Information Presenter – HTML study note upload interface.

1. General Rules for Uploaded HTML Study Notes

1.1 Scope and Structure

- **Study note files (slide)**:
  - Exactly one HTML file (e.g., index.html).
  - Exactly one CSS file named styles.css — if the file has a different name, it will be renamed automatically.
  - Optional media files (images, audio, video, subtitles) in permitted formats.

These requirements provide a unified structure for easy indexing and processing within the system, while ensuring reliable content presentation.

- **Folder structure**:
  - All files must be located in a single directory (flat structure).
  - References must use simple relative filenames from the same directory (e.g., src="photo.jpg", not src="./photo.jpg").
  - Subdirectories are not allowed.

This design simplifies processing and eliminates file paths that could otherwise bypass security mechanisms.

- **External resources prohibition**:
  - CDNs, web fonts, and external images or media are not allowed.
  - data: URIs are not permitted.

This ensures that study notes are self-contained and system security is maintained by preventing external dependencies and tracking.

1.2 Scripts and Embedding

- JavaScript is not allowed.
- <script> tags are not permitted.
- Inline event handlers (e.g., onclick, onload) are not permitted.
- <iframe>, <embed>, <object>, and <form> elements are not allowed.

This eliminates potential channels for malicious code execution or data exfiltration.

1.3 Permitted File Formats

- Images: jpg, jpeg, exif, jfif, jpe
- Media: mov, mp4, m4v, mpeg, mpe, mpg, ogv, webm, flv, f4v, mp3, aac, flac, m4a, oga, ogg
- Subtitles: vtt
- SVG is not allowed and must be converted to PNG, JPG, or WebP.

SVG allows script embedding and is therefore unsafe and prohibited.

1.4 Automatic Playback and Media Behavior

- Automatic media playback is disabled.
- Media controls are always enabled.

This avoids unwanted audio or video playback and ensures user control.

2. HTML Rules

2.1 File Structure and References

- Within the <head> element, only the styles.css file may be included.
- Inline CSS using the style attribute and <style> blocks are not permitted.
- All media files must be referenced using simple local filenames.

This prevents the use of embedded styles or external dependencies and ensures consistent processing within the system.

2.2 Media Elements

- **Source**: <video> and <audio> elements must contain one or more valid <source> elements.
- **Subtitles**: Subtitles are optional. When present:
  - the src attribute must reference a local .vtt file;
  - the kind attribute must be one of the permitted values (subtitles, captions, descriptions, chapters, metadata);
  - the srclang attribute must contain a valid language code (e.g., en, fr);
  - the label attribute will be sanitized.
- **Poster (only for video)**: For video elements, the poster attribute is optional but, when used, must reference a locally stored and safe image.
- **Dimensions**: The width and height attributes must be positive integers.
- **ID attributes**: If id attributes are present, they must be valid and unique; otherwise, they are generated automatically by the system.

These requirements ensure correct media playback and safe referencing.

2.3 Sanitization

- Invalid media elements are removed during processing.
- Valid media elements are reconstructed with safe attributes and enabled controls.
- Cache-busting parameters may be added to media source references.
- Unsafe references are removed.

This sanitization process prevents incorrect behavior and protects against injecting malicious content.

3. CSS Rules

3.1 Input Constraints

- The maximum allowed size of the CSS file is 200 KB.
- The maximum number of CSS rules is 5000.

These limits control performance and prevent abuse through excessively large style definitions.

3.2 Types of Rules

- Only standard CSS style rules are permitted.
- All at-rules are removed, including @media, @keyframes, @font-face, and others.

This avoids complex execution scenarios and external dependencies.

3.3 Permitted CSS Properties

- **Layout and box model**: display, position, top/right/bottom/left, margin, padding, border, border-radius, outline, box-sizing, width, height, min/max-*, overflow, float, clear, z-index.
- **Typography and color**: font, font-family, font-size, font-weight, font-style, line-height, text-align, text-decoration, text-transform, letter-spacing, word-break, white-space, color, background-color.
- **Lists and tables**: list-style, list-style-type, list-style-position, caption-side.
- **Flex and grid helpers**: gap, row-gap, column-gap, align-items, justify-content, align-content, flex, flex-grow, flex-shrink, flex-basis, flex-direction, flex-wrap.
- Visibility: visibility, opacity.

3.4 Disallowed CSS Properties

- Explicitly disallowed CSS properties: filter, backdrop-filter, mix-blend-mode, cursor, clip-path, -moz-binding.

These properties may enable unwanted visual effects, excessive resource usage, or security vulnerabilities.

3.5 Value Restrictions

- The use of url(...) or data: values is not permitted in CSS declarations.
- The use of expression(...), behavior(...), and -moz-binding is not permitted.

These mechanisms can embed scripts or unsafe references and thus are prohibited.

3.6 Empty or Invalid Rules

- CSS rules that contain no valid declarations are removed.

This ensures that only meaningful and valid style definitions are processed.

4. Checklist for Authors

- All files are located in the same directory – references use filenames without paths.

- There is exactly one HTML file and one styles.css file.
- There is no JavaScript, no inline CSS, no <style> blocks, and no @import.
- Only permitted media formats are used; SVG is not included.
- Each <video> or <audio> element contains at least one valid <source>.
- There are no external URLs or data: URIs.
- CSS uses only permitted properties; no at-rules are present; size and rule-count limits are respected.
- No assumptions are made about autoplay.

This checklist serves as a final verification step prior to upload to ensure that the study note complies with safety and compatibility standards.

These constraints follow established principles of secure web design. They are intended to prevent script-based attacks and other vulnerabilities that could compromise the safety of students, without reducing the quality of the study notes [3], [4], [5].

The system validates and sanitizes HTML, CSS, and VTT files. HTML and CSS are reconstructed, with the addition of system scripts, hash codes, specialized links, and handlers, in order to ensure uniform operation of uploaded HTML study notes with the rest of the system—for example, the audio volume state, which is stored by IP-app in the Windows Registry and is not accessible to HTML content.

In addition, during the reconstruction of the HTML, lists of referenced files are generated so that appropriate errors are raised in cases where files are uploaded but unused, or required files are missing.

Creation of User-Uploaded HTML Study Notes by Teachers

The ability to upload HTML study notes to IP-lib changes the way teachers create content and the way students use it.

- From the teacher's perspective, HTML study notes provide significant flexibility. Teachers can:
  - upload existing learning materials prepared as web pages;
  - more easily create complex layouts and visually appealing resources;
  - use their experience with web technologies to produce high-quality content;
  - use generative artificial intelligence to create diverse and comprehensive content quickly and efficiently.

In this context, HTML study notes function as an "open interface".

- From the student's perspective, HTML study notes are perceived as completely ordinary study notes within IP-app. Students:
  - interact with a unified interface regardless of whether a document-based study note was created using the builder or uploaded as an HTML file;
  - receive content without the risk of malicious code or external dependencies.

The new functionality for HTML study notes allows teachers to upload a large volume of diverse information, including content generated with the help of modern automation tools and artificial intelligence. As a result, the process of creating learning resources becomes significantly more automated and simplified, enabling the rapid construction of rich educational libraries. Figure 3 shows an uploaded HTML study note created with the help of ChatGPT.

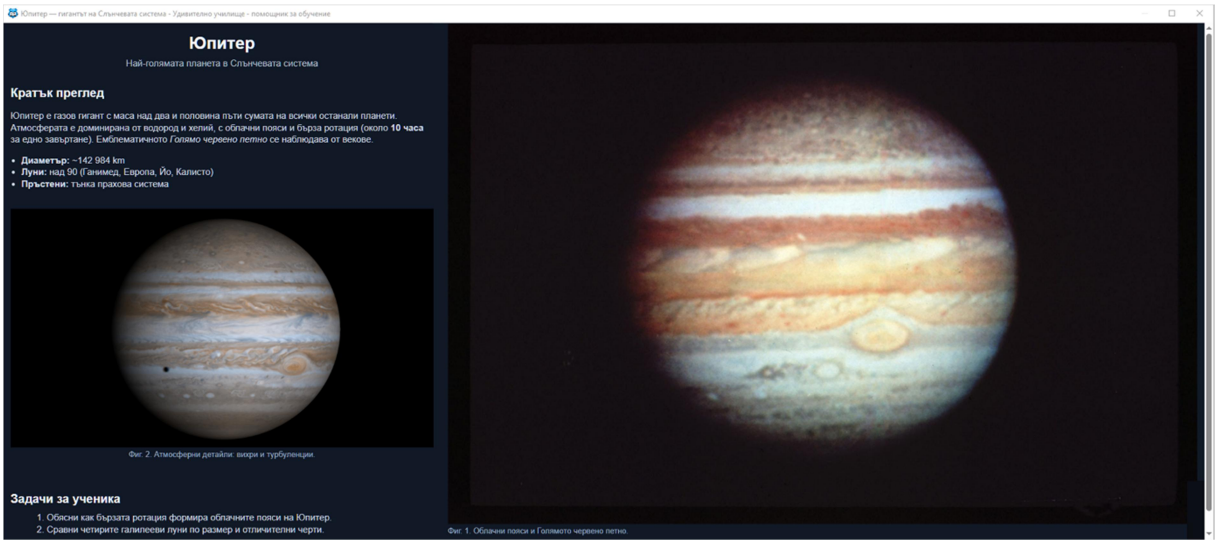

Fig. 3. Information Presenter – uploaded HTML study note (opened).

Creation of User-Uploaded HTML Study Notes by Students

A key aspect of comprehensive learning processes and the integration of IP software into them (IP-3) is that students not only consume information but also create study notes themselves as part of the learning process. In this way, they develop skills related to collecting, understanding, structuring, and presenting knowledge, which are an essential component of active literacy.

The availability of generative tools (including artificial intelligence) for content creation in recent years introduces real challenges in the educational process, as they enable students to present such generated work as their own [6], [7], [8]. This problem is well known but lies outside the scope of the present paper and will be analyzed in future work.

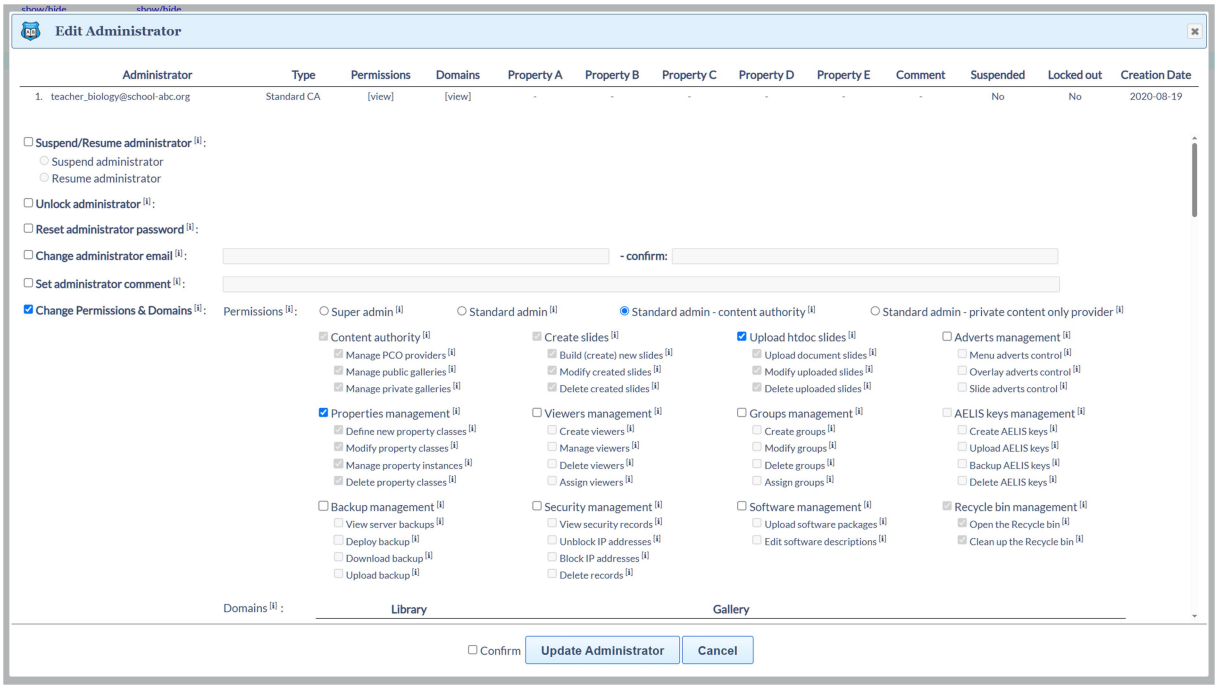

Fig. 4. Information Presenter – permission for uploading HTML study notes.

In the presence of such tools, the ability to upload ready-made HTML study notes creates a risk that some students may use them instead of creating the content themselves.

This would allow students to bypass the educational process. Rather than constructing the content on their own, they could simply upload a machine-generated file. This risk is mitigated, and its consequences are limited, by the system's design, as it includes a dedicated permission for uploading HTML study notes, as shown in Figure 4.

When students are granted the right to create study notes using the IP-lib interface, but not the right to upload HTML study notes, then—even if students copy ready-made text into the builder—the system still requires:

- manual structuring work, such as filling in cells, defining formatting, and arranging elements;
- a minimal degree of personal contribution, even when the content itself is copied.

This "forced effort" in fact has important educational value, even if it is less than when the student creates the text independently, because it provides an opportunity to read and comprehend the material while embedding it into the structure of the study note. In this way, the system provides at least a minimal level of balance:

- Teachers receive the freedom to upload their own HTML materials or ready-made resources.
- Students work with the builder, which ensures that the process of construction remains part of their learning.

It is important to emphasize that the inability of students to create HTML study notes, when they are not permitted to do so, does not change the fact that they can use generative tools in their work, regardless of whether that work is related to the Information Presenter or not. This is an ongoing problem that requires a long-term solution.

Conclusion

The ability to create HTML study notes provides substantial opportunities and advantages for teachers, who can use this functionality to deliver materials and information to their students in a more efficient and effective manner.

The present paper builds upon earlier work on the Information Presenter [1,2] by introducing the new functionality for HTML study notes.

Although this paper presents the HTML study notes feature, the broader pedagogical and ethical questions related to the creation of study notes by students using artificial intelligence are beyond the scope of this work and will be addressed in future research.

Future work will focus on empirical classroom evaluation of the system's impact on learning outcomes and student engagement. Interested teachers or schools are invited to participate in pilot classroom use.

References

1. Bonchev, M. (2024a). Architecture of Educational Information Systems. Part 1: Information Presenter in the Context of Education. In Innovative STEM Education (STEMedu-2024), Science Series, Vol. 6, pp. 257–280. DOI: https://doi.org/10.55630/STEM.2024.0629
2. Bonchev, M. (2024b). Educational Methodology in Software Systems. Part 2: Information Presenter in the Context of Education. In Innovative STEM Education (STEMedu-2024), Science Series, Vol. 6, pp. 281–309. DOI: https://doi.org/10.55630/STEM.2024.0630

3. OWASP Foundation. (2021). OWASP Top Ten: Web Application Security Risks. Available at: https://owasp.org/Top10 (last view: 15-09-2025).
4. Stock, B., Lekies, S., Mueller, T., Spiegel, P., & Johns, M. (2014). Precise client-side protection against DOM-based cross-site scripting. In 23rd USENIX Security Symposium (USENIX Security 14), pp. 655–670.
5. Van der Stock, A., Williams, J., & Wichers, D. (2013). A guide to building secure web applications and web services. OWASP.
6. Floridi, L., & Cowls, J. (2019). A unified framework of five principles for AI in society. Harvard Data Science Review, 1(1).
7. Holmes, W., Bialik, M., & Fadel, C. (2019). Artificial Intelligence in Education: Promises and Implications for Teaching and Learning. Center for Curriculum Redesign.
8. Luckin, R., Holmes, W., Griffiths, M., & Forcier, L. B. (2016). Intelligence unleashed: An argument for AI in education. Pearson.

Received: 19-09-2025 Accepted: 12-12-2025 Published: 29-12-2025

Cite as:
Bonchev, M. (2025). "The Information Presenter in the Context of Education: HTML Study Notes and Automation for Teachers", Science Series "Innovative STEM Education", volume 07, ISSN: 2683-1333, pp. 310-323, 2025. DOI: https://doi.org/10.55630/STEM.2025.0728