Enhancing Users Collaboration By Vocal Annotations

Enas Asmar, Saja Salahat, Farah Zubdeh and Amjad Hawash*
Information & Computer Science Department
An-Najah National University
Nablus, Palestine, P.O. Box 7
Email: enasasmar46@gmail.com, saja.tash1999@gmail.com, farahzubdeh@gmail.com, amjad@najah.edu

***Abstract*—Several methods have been established to enhance users' discussions and collaboration over the Internet. Websites are considered good media for that, in which users attach their notes (annotations) to websites components (texts, images, and videos) as a method of conducting online discussions. Annotations have several formats: textual, vocal, drawings, and visual. However, although the textual annotations are very famous, creating vocal annotations (and other types) has become a widespread activity through which users add their own voice notes to the components of HTML web documents as a method of having different discussions and thoughts exchange. The work conducted here is related to implementing a collaboration technique by creating and submitting voice annotations (private, public, and custom) as well as the ability to support annotations with a set of textual tags to be used for annotations searching purposes. The work also contains an experimental test to measure the degree of enhancement of users' collaboration and thought exchange by moving from textual annotations to vocal ones where 20 participants are involved of which 88% are over 18 years old. 123 annotations were submitted of different types with an average of 3 tags per annotation. Promising results were collected in which most of the participants (65%) are with vocal annotations.**

***Index Terms*—Voice Annotation, Google Extension, Users Collaboration.**

I. Introduction

Digital annotation has become an excellent tool to exchange people's ideas globally. With the existence of Internet services and tools and by the enhancement of programming offered by different scripting languages, several types of annotations are developed to enhance people collaboration and discussion by exchanging ideas with respect to some topic appears in several websites [1] [2].

Taking digital notes (annotations) is very important when reading different types of digital materials and this becomes very useful with the abundance of note types. This also becomes more fruitful when these notes are exchanged with others as a kind of collective discussion about some topics [3]. With the tremendous and increasing amount of websites and the variety of topics they present, they become a good media to conduct global collaborative discussions between users[1].

Corresponding Author: *amjad@najah.edu.

[1]From now and go on, the terms **users** and **annotators** will be used exchangeably.

Textual-based annotation is the most famous technique used (and still) to express ideas by annotating digital contents of websites by texts written by annotators as a method of thoughts exchange about some common subjects by attaching these texts to digital components of websites that are related to the subjects being discussed [4]. Although textual annotations introduced a good method of thoughts exchange, they lack expressing emotions that can be embedded within annotations themselves that will be reflected positively on better ideas expressions. Punctuation marks could help in this but still suffer from giving the exact emotion of the annotator. However, recently, creating voice annotations became very famous and preferable by most annotators due to the simplicity of annotations creation and the ability to express emotions more than textual ones. Users became able to create vocal annotations and embed them with vocal tones that express users' emotions like strange, anger, happiness, ...etc [5].

In this work, users can markup texts of websites by vocal annotations created by their own voices and saved in audio files to be heard by other annotators and hence conduct some collaboration between them vocally [6]. These voice annotations are digitized and stored in a specially dedicated database. When previously annotated websites are revisited by users who created vocal annotations in these websites, or other users who use the tool, annotated texts are viewed highlighted to be distinguished from other texts. In order to hear the annotations of any part, annotators can click the highlighted ones, and then a special pop-up window appears containing all recorded annotations (the original ones and all their replies).

Vocal annotations in this work are of three types: **Public**, **Private** and **Custom**. During the creation of annotations, users are able to choose between these types. The public annotations are those that are visible (the highlighted texts) to all users of the tool. The private ones are just visible by their owners (the annotators who created these annotations). Public annotations enhance collaboration since they are viewed by all users of the tool, while private ones enhance privacy and limits collaboration since they can be viewed by their owners. To solve this conflict, annotators can choose the custom option to determine who can view their annotations by selecting them from a drop-down menu contains a list of

users. This is similar to create a private virtual room of related users who can collaborate within a closed environment. By custom annotations, privacy is preserved with the existence of conducting collaborative discussions.

In order to increase the collaboration between users, it is important for them to locate annotations of their interests. To make this feasible, users can add some textual tags (separated by commas) in a text box that appears in the pop-up annotation window and these tags are attached with each vocal annotation and saved in the database as well. Tags are chosen by the user who is creating the vocal annotation and they reflect his/her own intent in creating the annotation. They somehow describe the essence of the annotation itself. By using a simple searching facility, users can search for these tags (from their portals) and a list of public clickable annotations appear in which they can visit the websites where the annotations have been created and participate in the discussions corresponding to the clicked annotations. Of course, a filtering process takes place here:

1) The public annotations are retrieved for all users of the tool despite who created them.
2) The private annotations are retrieved only for their owners.
3) The custom annotations are retrieved for their owners and for the users that are previously added by the owners of these annotations at the time of annotations creation.

Of course, the reply to the three types of annotations is also obeyed this filtering process. For example, if a public annotation has been replied to by a custom one, the users who have been added by the custom reply can retrieve the reply itself.

The rest of this paper is organized as follows: Previous work is proposed in Section 2. Section 3 discusses the system architecture of the tool while Section 4 is related to Annotations Creation and Retrieval of the tool. Section 5 demonstrates the experimental results. Finally, Section 6 concludes this paper.

II. Related Work

The literature has a lot of works related to annotating web content. Many works have been conducted with respect to web annotations of different methods: textual, vocal, and drawing-based.

Textual-based annotation tools are a lot since the this type of annotations are considered the initial efforts paid for digital annotation. Doccano (an open source tool) [7], Brat [8], Prodigy [9], Tagtog [10], and MADCOW [11] are all examples of such tools.

Creating annotations by drawing shapes over the contents of websites is also becomes a famous technique. Using this type of annotations, users are able to draw shapes over some contents of the websites and attach these drawings by textual notes. These drawings are shared between users of such tools to conduct some kind of discussion based on the drawings and their notes. Several tools exist on the web to enhance this kind of annotation. Sketchpad [12], Sketch to Web [13] and MADCOW [14] are all examples of famous tools related to this kind of annotations.

Despite that textual and/or drawing annotations are very useful and introduces an excellent way of collaboration between users, they suffer from expressing the emotions between users that definitely add some extra value for the conducted collaboration. Users are unable to express their exact feeling using textual and/or drawing annotations like what vocal ones can introduce. By vocal annotations, users are able to create fully emoted annotations that exactly express their feelings, as well as their opinion, regards the discussed material in which an extra value is added to the conducted discussion.

With respect to vocal annotations, the works are limited so that very few tools exist on the web. From those, we can mention XODO [15] that is used to create vocal annotations that can be attached in PDF files only of the mobile devices. VISITView [16] that can be used to record voice annotations for teaching purposes only. Finally, Chrome Audio Capture [17] which is a Google Extension that enables users to record their voice notes without relating them to some web contents. This tool is similar to our work in that it is implemented by Google Extension but it is not used to attach vocal notes to websites contents, it is just a recording tool that works under the environment of Google Chrome browser.

Creating visual annotations (by video) is a promising way of conducting collaborative sessions globally. Users here are able to create their own videos and attach them to some web content as a way of web-based collaboration techniques. However, the work still immature enough and lacks several supporting services.

The work conducted in [18] studying representations of practice, in video clips and voice annotations, for professional collaborative learning in distributed online environments. The project ran two main studies with learning technology professionals. The key results emerging from this work are that different kinds of video clips were found to offer support for professional development needs. The subjects agreed that watching themselves in video clips was still a relatively novel experience and that using artifacts relating to the practice helped to 'ground' or anchor a representation in realistic ways for reflective learning. The studies found video clips and voice annotations could be rapidly created. Speed was seen as a significant benefit and regarded as more important than creating a polished product, especially in fluid, evolving areas such as the learning technology field.

In [19], this paper the authors present a novel framework and its prototype tool for indexing and retrieving specific fragments of voice recordings obtained during discussions about physical objects such as text documents, pictures, or 3D models. When a specific part of an object is mentioned, it is tagged with an ink dot that is immediately registered in a database by capturing a microscopic image of the dot. Simultaneously, an index of the recording fragment is created

and linked with the dot. After the recording, a dot can be scanned and identified by matching its microscopic image with the database to retrieve the linked recording fragment for playback. A handy tool was developed to facilitate these operations while the user concentrates on the ongoing discussion. Performance tests of the dot identification have shown genuine matches without error. In demonstrations of a realistic usage scenario, the tool successfully facilitated the creation of indexes with dots during a voice recording and correctly played back all the specific recording fragments linked to the dots.

Despite the mentioned tools and works introduced a good base for vocal digital annotations, they did not pay attention to the type of annotations themselves nor locating the related annotations to users. Here comes our contribution of this work that enables users to create 3 types of annotations: **public**, **private** and **custom** ones. This enables (as we mentioned before) the collaboration between users with the preservation of some privacy. Moreover, the ability of tagging annotations by a set of comma-delimited words or either simple sentences makes it easier for users to search for annotations related to their interests in which the mentioned tools also did not introduce. Added to that, the tool ease of use as emerged by the experimental tests, represented by navigation from annotated website to the vocal annotations created by users and located in their portals and vice versa. The indentation display of annotations based on whether they are original created annotations or replies on them is also supports the ease of use of the tool.

III. System Architecture

The idea of annotating web content is not more than adding a transparent layer above an annotated website. This invisible layer represents the tier where annotations exist. The tool implemented in this work follows the web-based client/server architecture. Users add their own voice annotations to texts on an HTML web page and upon the save action, the necessary data are saved in a dedicated database. The tool and according to this mechanism is composed of a set of layers: **Presentation**, **Processing** and **Database**. The Presentation Layer deals with the interaction between users and the browser in which annotations are created, attached by placeholders to selected texts, and then saved in the dedicated database. The Processing Layer, located between Presentation and Database layers, deals with all processing needed to color the highlighted text, recording the voice annotation, and saving it in an audio file. Finally, the Database Layer is related to the actual saving of all data related to the created annotations represented in the annotated text, the type of the annotation, the URL, the audio file, and the annotator related data. Moreover, the Database layer is responsible for the retrieval process of the annotations for a given user and website. Figure 1 below depicts the structure of the tool where the request/reply protocol is used to save/retrieve annotations respectively.

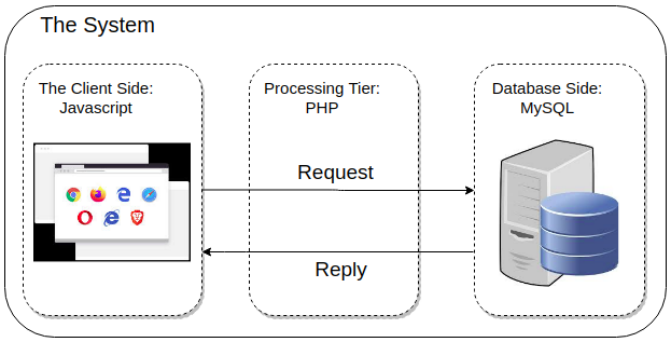

Fig. 1. The architectural design of the tool.

Google extension is used to implement the interaction part between the users and the tool. A special google extension is programmed for this purpose where all annotation functionality is implemented. The extent ion contains a set of JavaScript functions that are injected in each visited website in order to execute the functionality of text selection and vocal annotations recording. User login and the creation of new users' accounts are also implemented in the extension.

Figure 2 depicts the components of the tool where credentials of users are implemented using the sign-up and login components. Whereas the activities related to the annotation process are implemented in the Google extension and database components.

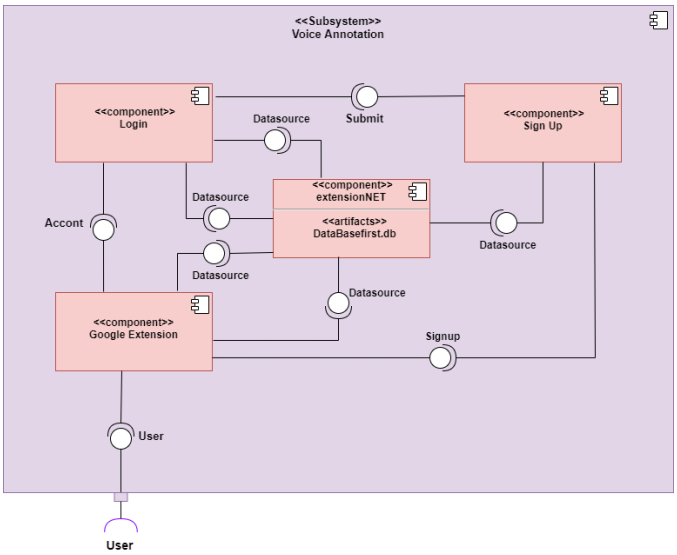

Fig. 2. The Component Diagram of the tool.

Figure 3 below represents the logical schema of the database represented by an Entity-Relationship diagram. The diagram contains 3 different entities: **User**, **Voice** and **Text** alongside the entity relationships between them. The user entity reflects all attributes related to users (Annotators), Text is used to save data about the annotated text and Voice entity is to save data

about the vocal annotation itself.

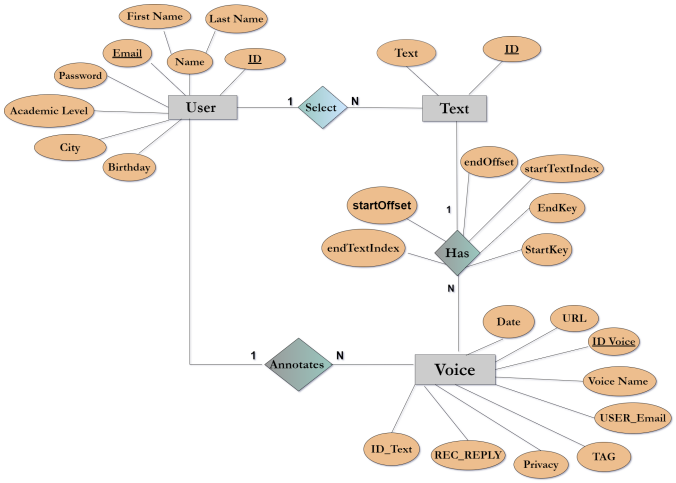

Fig. 3. The Entity Relationship Diagram.

Figure 4 represents the sequence diagram for the annotation creation activity. The activity begins with some user who selects some text on some website. This fires upon the Browser object that executes the function **createAnnotation(selectedText)** identified in JavaScript object. The later executes the defines the problem statement while method **recordVoice()** on the object Voice Recorder that returns the recorded voice saved in **recordedVoice** object and returns it back to the JavaScript object. After that, the Javascript object executes the function **Save(selectedText, recordedVoice, URL)** identified in the Database object that saved all related data in the database of the system. Finally, the user is notified by a **Notify()** function that prompts the user with the successful creation of the vocal annotation.

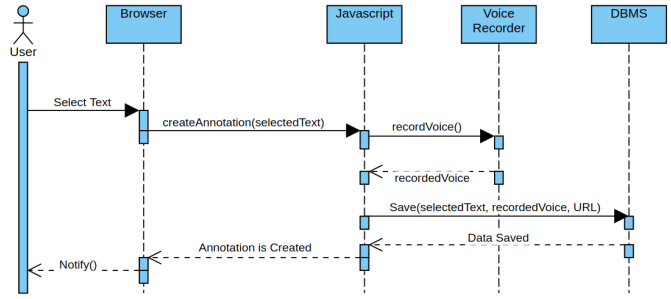

Fig. 4. The Sequence Diagram for the annotation creation activity.

The following pseudocode segment summarizes both the vocal annotation submission and retrieval.

```
while(true){
    // List of Google Extension embedded listner functions
    GoogleExtension.onChangeURL() = injectJavaScript(URL);
    GoogleExtension.onLoadAnnotations() = loadAnnotations(
        URL);
    GoogleExtension.onSelectText() = createAnnotation(URL,
        userID);
    GoogleExtension.onHighlightTextClicked() =
        retriveAnnotations(URL, clickedText);
}

function createAnnotation(URL, userID){
    placeHolder = selectText(selectedText);
    displayPopup(placeHolder, selectedText);
    recordedAudio = recordAudio(recordedFile);
    saveAnnotation(userID, selectedText, recordedAudio,
        placeHolder, URL);
}

function retriveAnnotations(URL, clickedText){
    AudioFile Files[];
    popUpWindow = new popUpWindow();
    Files = getAnnotations(URL, clickedText);
    for(i = 0; i < Files.size(); i++)
        popUpWindow.insert(Files[i]);
}
```

Upon the Google Chrome browser launching, the Google Extension is loaded and the mentioned code is executed. The code keeps revolving in an endless loop by executing the listeners mentioned inside the loop. The listener function *GoogleExtension.onChangeURL()* is executed whenever the logged-in user navigates a new URL that leads to executing the function *injectJavaScript(URL)* in which injects the suitable JavaScript code that will be responsible for vocal annotations handling.

The listener function *GoogleExtension.onLoadAnnotations()* is executed when the user clicks the Google Extension to retrieve all annotations previously submitted to the current URL by invoking the function *loadAnnotations(URL)*. This function also is responsible for re-highlight all text mapped with annotations for the current URL and takes into account the filtering of these annotations according to their types: public, private, and custom.

The listener function *GoogleExtension.onSelectText()* is executed when the user selects some text located in the current URL. Upon the execution of this listener, the function *createAnnotation(URL, userID)* is executed with the *URL* and *userID* as parameters. The last listener function is *GoogleExtension.onHighlightTextClicked()* which executed when the user clicked a highlighted text in the current URL in which the function *retriveAnnotations(URL, clickedText)* function is executed to load all annotations related to the *clickedText*.

The function *createAnnotation(URL, userID)* is responsible to create a new annotation of the current *URL* and for the current logged-in user represented by *userID*. The code inside the function *createAnnotation(URL, userID)* is initiated by executing the function *selectText(selectedText)* that returns the placeholder (the position inofrmation of the selected text) and saves it in the variable *placeHolder*. Upon the text selection, the function *displayPopup(placeHolder, selectedText)* is executed in which the selected text and its place Holder information are inserted in the window itself. This popup window contains all necessary controls to create the vocal annotation and save it inside the database. Figure 5 depicts these contents. After that the function *recordAudio(recordedFile)* is executed to begin recording the voice annotation and upon the completion, the

recorded file is saved in the variable *recordedAudio*. The function *saveAnnotation(userID, selectedText, recordedAudio, placeHolder, URL)* is then executed to save all annotation related data inside the dedicated database.

The retrieval of previously submitted annotations takes place with the invocation of the function *retriveAnnotations(URL, clickedText)* that takes the current *URL* and the *clickedText* as parameters. This function creates an empty array *Files* of type AudioFile to be used to save all retrieved audio files related to the highlighted clicked text. The constructor *popUpWindow()* is then executed to create a new pop-up window that saved in the object *popUpWindow*. The function *getAnnotations(URL, clickedText)* is then executed to retrieve all related audio files to be saved in the array *Files*. Finally, the loop evolves the array of audio files to add necessary HTML and JavaScript codes to append and playback the inserted audio files by executing the member funcotion *popUpWindow.insert(Files[i])*. Of course, the load of the audio files is filtered with the annotation type: public, private or customer.

Fig. 5. The pop-up window to create an annotation.

IV. Annotations Creation and Retrieval

After a user visits some interesting website, s/he can select some text of his/her own interest to adorn it with a vocal annotation. The user then clicks the system extension icon located in the Google extensions area of the Google Chrome browser in which creates a pop-up window containing the selected text as well as a set of controls to create and save an audio file. The user then begins recording his/her own voice annotation after selecting the type of the annotation as well as supplying a set of optional tags that represents the tenor of the annotation. These tags will be used for the annotations searching process. Upon clicking the save button located in the pop-up window, the tool saves the annotation and all its related data in the database of the tool. Figure 5 below depicts the pop-up window used to create the annotation where the yellow highlighted selected text is copied from the original website to the pop-up window. The custom option is selected as well as the list of custom users are determined. The figure also shows the list of tags filled by the annotator.

From Figure 5, when the user clicks *Start* button, the recording process starts in which the user begins recording his/her note that stops when s/he clicks the *Stop* button. Upon clicking the *Save* button, the Google extension JavaScript embedded code contacts the necessary PHP code to save the audio file in a special folder in the server under an automatically generated file name that saved in the database to create a link between the saved annotation and the actual audio file related to it. In this context, the actual audio file is saved in a dedicated server folder, while the name of the file itself is saved in the database. This creates dome mapping between the name of the file and the actual file in the server for file retrieval purposes when the vocal annotation needs to be retrieved later on.

The users are able -after logging into their portals- to list all their own annotations with their filtered replies. They can play the audio files associated with them with the ability to visit the website's associates with these annotations. Figure 6 illustrates the way the annotations are listed in some user's portal. The figure contains the annotated text and the set of all vocal annotations related to it. The white background ones are original annotations, while the blue background ones are replies to the original ones (for example the blue annotations that appear in the figure are replies for the annotation labeled with *mySound55*). Each annotation contains the audio control for the audio file associated with it and contains data related to the owner of the annotation, the date it was published, its type, and the set of tags related to it. The reader may notice the indentation of the replies for the original annotation and may notice also that *mySound304* is another annotation related to the same highlighted text. Of course, these annotations are clickable so that each one is linked to the website where the associated annotation is created. Upon clicking one of them, the tool opens the associated website and uses its URL to retrieve all annotations associated with it taking into account the custom and private filters. All texts that appear on that website will be re-highlighted and will be clickable so that a user can view all filtered annotations related to the clickable texts.

On the other side, if a logged-in user navigates a sample previously annotated website, s/he can ask the google Extension to retrieve all annotations submitted to this website (if there is any). Upon that request, the tool re-highlighted all annotated texts so that the user can click any of them to open a pop-up menu that contains all clicked text related filtered

annotations [2].

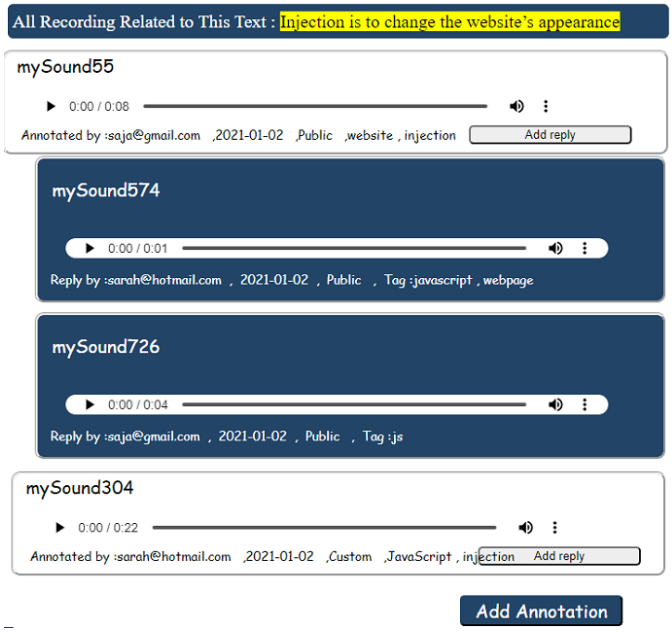

Fig. 6. The list of saved vocal annotations and their replies.

Users of the tool also can use their portals to search for published annotations of their interest. Of course, the searching process takes into account the filtering of permeable annotations. The tags-based search displays all public annotations related to that tag as well as the private ones (if the user doing the search is the same as their owner) and the custom ones if the user is previously added by the owner of the annotation at the time the annotation is created.

V. Experimental Test

In order to compare vocal and textual annotations, we conducted an experimental test where 20 participants are involved in the test. The aim of the test is to differentiate between both methods in terms of collaboration between participants where have been chosen of different ages (88% are over 18 years old) and from different educational backgrounds.

During the supervised test that lasts for 5 days, the participants visited several websites related to the subjects being discussed as appears in Figure 8 and they used the tool in creating different vocal annotations as well as they used other textual based tools (as appears in Figure 9 that reflects the percentages of annotations). In order to better measure the collaboration between the participants in both cases: using vocal and textual annotations, we asked them to unify both the discussed subjects and the annotated websites in both annotating methods to be fair in the differentiation between the

[2]Please notice that Figure 6 serves both situations of annotation retrieval process: either the user asks to list his/her own annotations from the portal or s/he clicks a highlighted text.

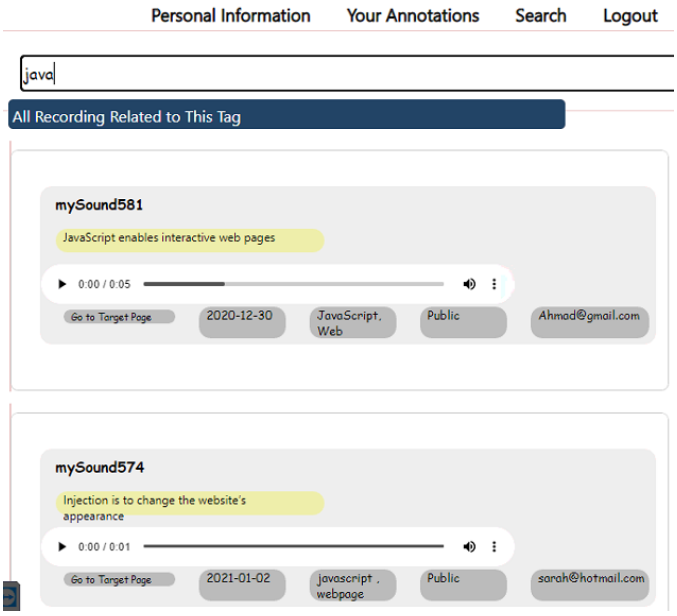

Fig. 7. Searching for some tag and displaying its related annotations.

two annotating methods. The participants were asked to submit different types of annotations: public, private, and custom. Moreover, they were asked to use the tags-based searching facility in order to search for other participants who submitted public and custom annotations.

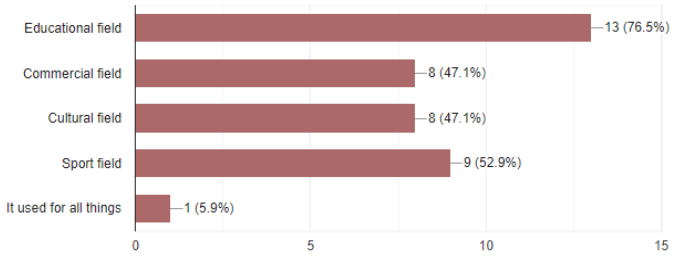

Fig. 8. Discussed subjects by participants.

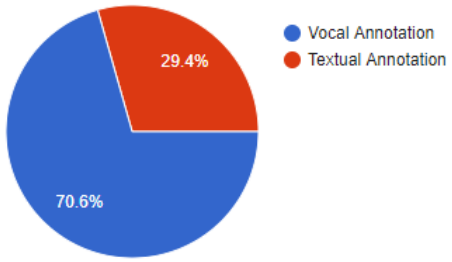

Fig. 9. Participants percentages in terms of kind of annotations used.

At the end of the test, we found that 123 annotations were submitted (67 public, 17 private, and 39 custom), 25 different

websites were annotated and 3 tags as average per annotation. The participants were asked to fill a special questionnaire in order to know their opinion about the experiment and the collaboration gained using the vocal annotations. The questionnaire has a set of questions like:

1) What is the effect of vocal annotations compared with textual ones on the amount of participants comprehension of the subjects being discussed? (Reducing time and effort, Increasing time and effort, No effect) The result of this question is depicted in Figure 10 where around 71% go with the reduction of time and effort.

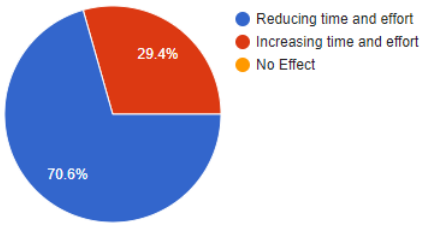

Fig. 10. Effect of vocal annotations on both time and effort.

2) How do you find collaboration enhancement using vocal annotations vs. textual ones? (Excellent, Good, Bad). The result of this question is depicted in Figure 11 where the majority go with excellent.

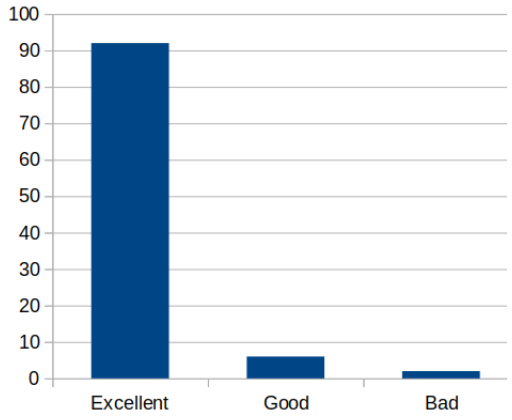

Fig. 11. Participants' collaboration level.

3) What is the level of tool usability? (Excellent, Very Good, Good, Fair). The answer to this question is reflected in Figure 12 where the percentages reflect the ease of use in general.
4) Which is better to express your emotions embedded in your submitted annotations? (Vocal, Textual). We got 65% for Vocal and the rest for Textual.
5) Which takes less time to annotate? (Vocal, Textual). We got 100% for Vocal since speaking is much faster that writing and no need to use hands like in typing.
6) Does the location of the annotator have an effect on choosing Vocal or Textual? %92 of the participants go with the Vocal annotations since several places limits the use of hands for typing, like waiting in a public transportation stop, or standing in the bus or tram, or any other place where talking is much easier than writing.

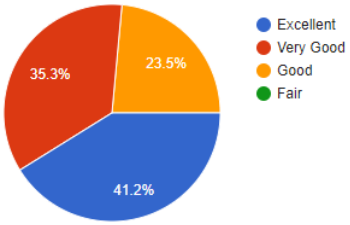

Fig. 12. Ease of use level.

VI. Conclusion and Future Works

A vocal-based annotation is implemented and tested in this work. Users are able to conduct collaborative sessions using voice annotations in order to discuss some shared interests issues over the web. Annotations are of three types: **Public**, **Private** and **Custom** with the ability to adorn them with a set of related textual tags for searching purposes and with the ability to choose users to be able to view and reply to the custom annotations. A supervised experimental test was conducted in which promising results were achieved.

For future works, we have several plans: Enhance the search facility by introducing a voice recognition mechanism to search for related annotations using their accompanying audio files. Selecting more related tags by analyzing the highlighted text and express the most important tags the better describe the text. Also annotating other web content like images and videos is one of our future works. Moreover, implementing visual annotations by recording video instead of recording voice is under our considerations.

References

[1] M. Chen, D. Zhang, and L. Zhou, "Empowering collaborative commerce with web services enabled business process management systems," *Decision Support Systems*, vol. 43, no. 2, pp. 530 – 546, 2007, emerging Issues in Collaborative Commerce. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923605000795

[2] M. Addisu, D. Avola, P. Bianchi, P. Bottoni, S. Levialdi, and E. Panizzi, *Annotating Significant Relations on Multimedia Web Documents*. John Wiley & Sons, Ltd, 2012, ch. 24, pp. 401–417. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118219546.ch24

[3] G. Ben-Yehudah and Y. Eshet-Alkalai, "The influence of text annotation tools on print and digital reading comprehension," 01 2014, pp. 28–35.

[4] I. A. OVSIANNIKOV, M. A. ARBIB, and T. H. MCNEILL, "Annotation technology," *International Journal of Human-Computer Studies*, vol. 50, no. 4, pp. 329 – 362, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1071581999902471

[5] S. Ades, "Voice annotation and editing in a workstation environment," *Proceedings of the AVIOS '86 Voice Input/Ouput Systems Applications Conference, Alexandria, VA*, pp. 13–28, 2010.

[6] J. J. Cadiz, A. Gupta, and J. Grudin, "Using web annotations for asynchronous collaboration around documents," in *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '00. New York, NY, USA: Association for Computing Machinery, 2000, p. 309–318. [Online]. Available: https://doi.org/10.1145/358916.359002

[7] "Doccano: Text annotation for human," https://github.com/doccano/doccano, accessed: 2021-01-30.
[8] "Brat rapid annotation tool," https://brat.nlplab.org, accessed: 2021-01-30.
[9] "Prodigy: An annotation tool powered by active learning," https://prodi.gy, accessed: 2021-01-30.
[10] "Tagtog: The text annotation tool to train ai," https://tagtog.net, accessed: 2021-01-30.
[11] "Madcow: Multimedia annotation of digital content over the web," http://www.web-annotations.com, accessed: 2021-01-30.
[12] "Sketchpad," https://sketch.io/sketchpad, accessed: 2021-01-30.
[13] "Sketch to web," https://sketch-to-web.com, accessed: 2021-01-30.
[14] M. Antico, D. Avola, P. Bottoni, A. Hawash, K. Kanev, and F. P. Presicce, "An interactive tool for sketch-based annotation," *JJAP Conference Proceedings*, vol. 011604, p. 4, 2016. [Online]. Available: https://journals.jsap.jp/jjapproceedings/online/4-011604
[15] "Xodo: Creating sound annotations," http://feedback.xodo.com/knowledgebase/articles/1888654-creating-sound-annotations, accessed: 2021-01-30.
[16] "Recording audio and annotations," https://www.ssec.wisc.edu/visitview/tutorial/img7.html, accessed: 2021-01-30.
[17] "Chrome audio capturer," https://github.com/arblast/Chrome-Audio-Capturer, accessed: 2021-01-30.
[18] C. Steeples, "Voice annotation of multimedia artifacts: reflective learning in distributed professional communities," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, 2002, pp. 10 pp.–.
[19] T. van Bergen, M. Engineering, Y. Kudo, R. Ishiyama, T. Takahashi, K. Makino, and J. Goosen, "Indexing and retrieving voice recordings by instantly tagging mentioned objects with dots," 04 2019.