VoCol Experiments Description

1. System's specifications
    a. Some questions
        i. Does VoCol support other formats than turtle (.ttl) ? yes, but syntax checker is only for ttl format.
        ii. What traces are saved?
        iii. How are users aware of on-going remote file edition? (is the file locked?)
        iv. How can users cooperate? Synchronously, Asynchronously?
        v. Notes about syntax validation report (observed when trying VoCol on a sample ttl file)
            1. Having an error in a turtle file will not prevent from updating it to github. Should this be improved?
            2. The syntax validation report only shows the first error. Could the next errors be shown?
            3. In case of solving the old errors and creating a new one, the report is not updated. Can you confirm? Can it be improved?
            4. The syntax validation report doesn't trace users errors (which user has done the error). Could it be possible to keep track of each action?

    b. Installation and Configuration of VoCol
        i. UJM's responsibilities:
            1. Provide an initial ontology file
        ii. BONN's responsibilities:
            1. Configure VoCol with the initial ontology file, and provide a public access link to use it by the participants.
            2. Provide all the statistical reports to study them by UJM team. (At the end/during the experiments)

2. Experiments Specifications
    a. People Involved: We can start with around 10 people (i.e. 5 are experts at the books domain and the other 5 will be experts in ontological domain).
        i. St Etienne/books domain : Cedric, Antoine Fauchier,
        ii. St Etienne/ontological domain: Omar, Antoine, Pierre-René, Maxime
        iii. Bonn/books domain : Y, Z
        iv. Bonn/ontological domain : Lavdim, Irlan, …?
    b. Duration of the experiments: Initially 1 month, however this period can be extended based on the participants collaboration.
    c. Users (except some books domain experts) receive an URL to connect to the system. The login is the github's.
    d. Users (ontological experts) visualize the ontology (proposed in section 1.b), by checking the turtle format or graphic format, and then they add/modify/delete terms by editing the turtle file.

e. Committing procedure: each commit should be fully commented by the name of the user who did it, date of commit, new things added/deleted/modified, etc. This will help tracking the changes during the development process.
f. Sessions with book experts are organized by an ontological expert in order to get their inputs. These inputs are then entered by the ontological expert in the shared ontology.
g. Actions of users are logged and will be analysed.
h. Users receive a questionnaire to estimate a) usefulness, b) satisfaction and c) Ease of use (when? After some days? At the end?)

3. Experiments Outcomes
    a. Experiments Aspects (Must discuss with BONN)
        i. What to test?
        ii. Bonn's vision at the end of the experiments?
    b. Use of the tool
        i. Cooperation between users:
            1. Save the commits/ errors that are done by each user, and create a log file for them (1 per user or 1 single file for all users), to trace user's actions.
    c. The ontology
        i. Size of the ontology (properties, classes)
        ii. Contribution per user,
        iii. Measurement of the dynamic of the ontology creation

4. Ontology Development
    a. An initial version of the ontology will be uploaded to VoCol, with a proper documentation.
    b. Development process will start by the involved persons, and each one has to take care of the notes discussed on points (2. d, e , f).

5. Validation
    a. We can use the USE questionnaire with some modification, in order to cover all aspects that will be tested on point (3.a).
        i. Questionnaire will tests 3 main issues: a) usefulness, b) satisfaction and c) Ease of use.
        ii. Estimation of the quality of the results we get from the questionnaire.

Ontology Specification

This section presents a description about the ontology that we will develop in order to test VoCol functionality. An initial version of the ontology will be uploaded to VoCol (ttl format), and each of the participants will contribute to the ontology based on his/her understanding after reading this documentation.

1D company, works on a certain architecture that should be followed during the process of developing the ontology. In this document we are listing the requirements of 1D company based on their architecture. All participants are welcome to include their own contribution and comments during the process.

The most important terms that should be included in the ontology.

1. **Project:** will indicates different types of projects that the user will be involved in.
    a. Each project might be a Collection or Handiwork.
    b. Each collection might be a Series.
    c. Each series is a “Handiwork “, which contains different types of projects (books, magazines, comics, etc.)
2. **Person:** will indicates different types of involved persons in certain project (FOAF can be used). Can be one of the following:
    a. Author: to indicate the author of certain project.
    b. Editors: to indicate the editors (publishers) of certain project.
    c. Readers/Users: to indicate the users of certain project. This term will be connected to the “Person” term with a “has account” relationship.
    d. Example:
        i. Having a Handiwork of type book named Harrypotter.
        ii. Harry potter is a project
        iii. Harry potter is of Fantasy literature collection.
        iv. Harry potter is a series Handiwork.
        v. There are 7 books of Harry Potter that are all considered as a handiworks of type books.
        vi. The author of Harrypotter is a person named (J. K. Rowling). JK Rowling has some information, address, affiliation, etc. These could be mapped using foaf.
3. **Capsule**: this term will be connected with (Readers/Users) term in a way that each (reader/user) will create a capsule with “create capsule” relationship. Each capsule will include the following:
    a. Activity: to track activities of the user; it has 3 main possibilities; (a) Reading list, (b) Feedback and (c) Interest (whether like or dislike).
    b. Project Annotation Type: which indicates the type of certain project (e.g. Action, Thriller, Romance, etc.) .
    c. Collection: which contains both of “Book List” and “Quotes List”, that are connected to the projectType.
    d. Example : user (x) creates a capsule, in which he will used to list all his activities, and collections. For instance user X adds Harry potter to his collection.

4. **Tagging:** this term mostly connected with all other terms, and it should follows SKOS concept. An example of this term could be “ User X Tagging Address Y”, so user X will be tagged to address Y using Tagging term.