

The Field Linguists' App

1 Project Abstract

The FieldLinguists' App is an OpenSource database that allows language researchers to securely enter, store, organize, annotate, and share linguistic data. The application is accessible everywhere; it runs on three different systems (Mac, Linux, and Windows) and is suitable for both online and offline use. Furthermore, the application will be created with collaborative goals in mind; data will be syncable and sharable with other researchers. Researchers can form teams that contribute to a single corpus, where team members can modify and discuss the data from within the application. The system will also have a simple and friendly user interface, allowing participants to drag and drop files (audio, video, text), or record it directly into the database when using the Android app. In addition, the application will have import and export capabilities for multiple file types. Most importantly, the application is designed intuitively and theory free, so it is not necessary to be a field linguist or programmer to figure out how it works.

2 Statement of Need

The statement of need should describe the problem that the project will attempt to address. Also, describe the population that will be served.

3 App Description

The FieldLinguists' App will enable those interested in language research, preservation, and documentation to securely enter, store, organize, annotate, and share linguistic data. Moreover, the application will be easily customizable to fit specific needs. To accomplish these tasks, the database will be equipped with a variety of features. The system will support:

- data entry akin to other fieldlinguistics databases
- the ability to drag and drop audio to accompany each data field

- insertion of IPA and special symbols
- Hotkeys and keyboard short-cuts for common tasks
- import .csv files
- import ELAN files

These functional requirements are based on a few important considerations where most existing field linguistics/corpus linguistics databases applications fall short. The FieldLinguists' App will:

- be user friendly.
- be cute so that the users feel like entering data, instead of procrastinating.
- be designed for search as this is one of the most fundamental tasks a language researcher must be able to do. The search will go far beyond traditional string matches and database indexes; it will be able to display data in context.
- be OpenData enabled. Corpora often contain sensitive information, informant stories and other information which must be kept confidential. Having confidential data in plain text in a corpus forces the entire corpus to be kept confidential. Instead, the system will encrypt confidential data and store the data in the corpus encrypted. To access the plain text the user will have to log in and use a password to decrypt the data. This design has important ramifications for exporting data, and for editing the data outside the application.
- work offline. Running a webapp offline will have considerable consequences for how data is stored, how data is retrieved, and how much data can be used while offline. Most browsers have limits on the amount of data a webapp can store offline. By delivering a version of the app in a Chrome Extension, which has permission to have unlimited storage, researchers will be able to have a significant portion of their data at their fingertips, regardless of the location.
- run on Mac, Linux, and Windows computers. By providing the application as an installable Chrome Extension, it will run on all platforms as well as Chrome Books.
- be OpenSource. Being OpenSource allows departments to install and customize the database application to tailor specific needs without worry that the company behind the software will disappear or stop maintaining the software. In addition, OpenSourcing the software on GitHub will allow linguists with scripting or programming experience to contribute back to the software to make it more customized to their needs, language typologies, or linguistics research areas.

- store data in Unicode. Encoding problems and losing data should be behind us in the days of unicode. However, many existing fieldlinguistics databases were built in programming languages that did not support unicode, so the unicode support is dangerously fragile.
- be simple. The system will not be designed to replace advanced field linguistics databases, or corpus linguistics databases. It will be designed to replace Word Documents or LaTeX documents which is a very common way fieldlinguists store data because it requires no training, doesn't require a complicated set-up for data categories, and takes no time to add new categories.
- be theory free. The application will not include categories or linguistic frameworks or theoretical constructs that must be tied to the data. However, the system will make it possible for teams to set their own categories and thus optionally tie specific theoretical constructs to data. This is a different approach to many corpus databases; most databases are motivated by typologies and language counting that place a large data-entry burden on the user. The FieldLinguists' App will let the researcher gradually constrain their data entry conventions over time.
- be collaborative The system will have users and teams, and permissions for corpora. Permissions will ensure that data can be safely shared and edited by multiple users. Moreover, the corpus will be versioned so that users can track changes and revert mistakes.
- run on touch devices. Touch tablets are one of the easiest tools to carry and use in field; they have a long battery life; they can play videos or show images for the consultant to elicit complicated contexts; and they permit recording audio and video and direct publishing to YouTube and/or other services. Furthermore, Android tablets are particularly easy to program and integrate the microphone directly into the database.
- be smart. The application will attempt to guess what users do most often, and automate the process for them. Most importantly, the system will have automated predictable glossing information .

4 Goals & Objectives

Describe the project objectives in measurable terms that address the academic and technological needs of language teachers, linguists, etc. (it's for linguists but other people can benefit from it as well i.e. help create dictionaries for endangered languages, which will benefit the communities of these languages; L2 acq. teachers).

5 Staff & Organizational Information

iLanguage Lab is a Montreal based company that develops tools in the form of experimentation and data collection apps for Android and Chrome in collaboration with researchers at UdeM, UQAM, McGill and Concordia. iLanguage Lab has a background in assisting researchers obtain results by creating applications that suit the researcher's needs. Previous research applications includes the Bilingual Aphasia Test, AuBlog, and OPrime. The Bilingual Aphasia Test led to a presentation at the Academy of Aphasia 49th Annual Meeting on Aphasia Assessment on Android: recording voice, eye-gaze and touch for the BAT and a publication in the Academy of Aphasia. The AuBlog application was employed to investigate evolving information structure and audienceless vs. audience oriented prosodies and culminated in a poster presented at Experimental and Theoretical Advances in Prosody Conference.

5.1 Gina Cook M.A.

Gina Cook received her Masters in Field Linguistics & DESS in Computer Science.

6 Budget

Include in the budget all expenses for the app, including necessary training costs. Mention any co-funding that you are using from other sources. You may want to include a brief narrative of expenses along with a table of individual cost components.

7 Timeline

How long does it take us to make it? Project progress over time. What we've done and what we'll do.

8 Evaluation

Provide information on the metrics that will be used to determine the effectiveness of the project. It has to meet certain data management criteria – any scientist can discover and use the data over time; it will benefit the scientific community. We are abiding by the best practice standards set by DataOne.

Example format:

- (1) n-tuop'ti-m
1-window-POSS.SG

'My window'