

Grammatical Bible Search

Alison B-- ♦ January 2020

Problem

Some important theological debates revolve around the specific grammar of Biblical phrases (like in the Great Commission, shown on the right). Nobody speaks Biblical language natively anymore, and so to find out what a grammatical structure means, we have to find other examples of that structure in the text.

There are some software tools for finding grammatical structures in the Bible, but they're expensive, confusing, and not easily available. That makes it hard to investigate theological issues that involve grammatical questions, something I personally experienced when I took Greek for four semesters.

My goal during J-term was to start building an app to empower anyone who knows a Biblical language to find grammatical structures and study theological problems for themselves, even students just starting to learn a language, pros who aren't comfortable with technology, or pastors who can't afford expensive software.

“going” “make disciples”
πορευθέντες οὖν μαθητεύσατε
participle modifying

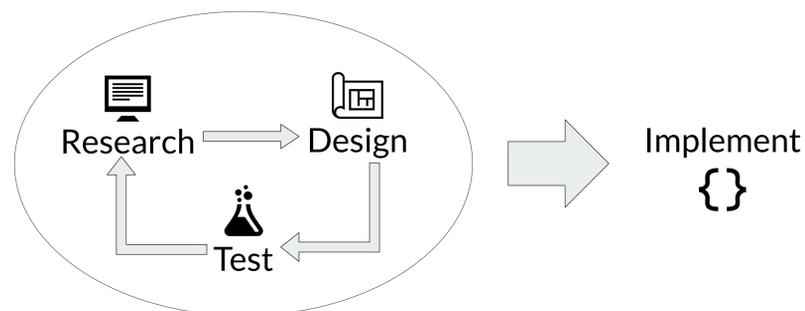
Is a participle modifying a command
also part of the command?

Goal: a ♦ free ♦ easy-to-use ♦ accessible app
for students, professors, and pastors to search the grammar of the Bible

Approach

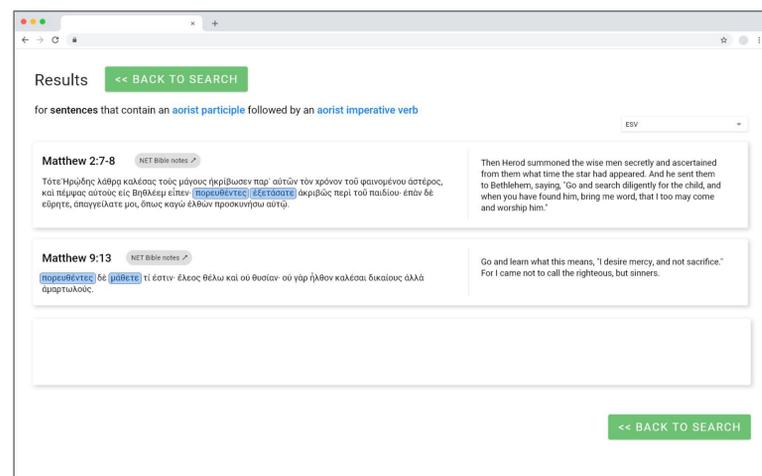
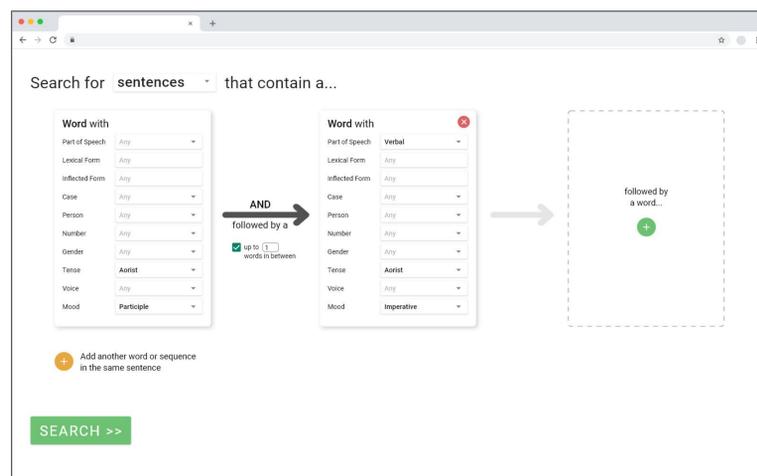
I knew the big challenge wouldn't be the algorithm for grammatical search, since it's been done many times before. Instead, the challenge would be designing an intuitive, friendly interface for grammatical searching - something that hasn't been done before. So I focused on an iterative process of:

- **Research.** I explored existing grammatical search tools and read through Greek books to find examples of searches that would be useful.
- **Design.** I came up with feature concepts and sketched ideas for the design of the app.
- **Testing.** I had people who knew Biblical languages try out my designs and give me feedback.
- Then I started to **implement** the interface and features that came out of the design process.



Design Accomplishments

1. First, I made a big list of feature ideas and sketched a very complicated design.
2. Then I cut it down to the most simple and useful features based on case studies from Greek books and feedback from a Greek professor.
3. Next, I sketched a prototype on paper and had two Bible professors try it out.
4. Based on their feedback, I made a more clean and detailed digital prototype and had three Greek students try it.
5. Then I used their feedback to further improve the design.



The final mockups

Code Accomplishments

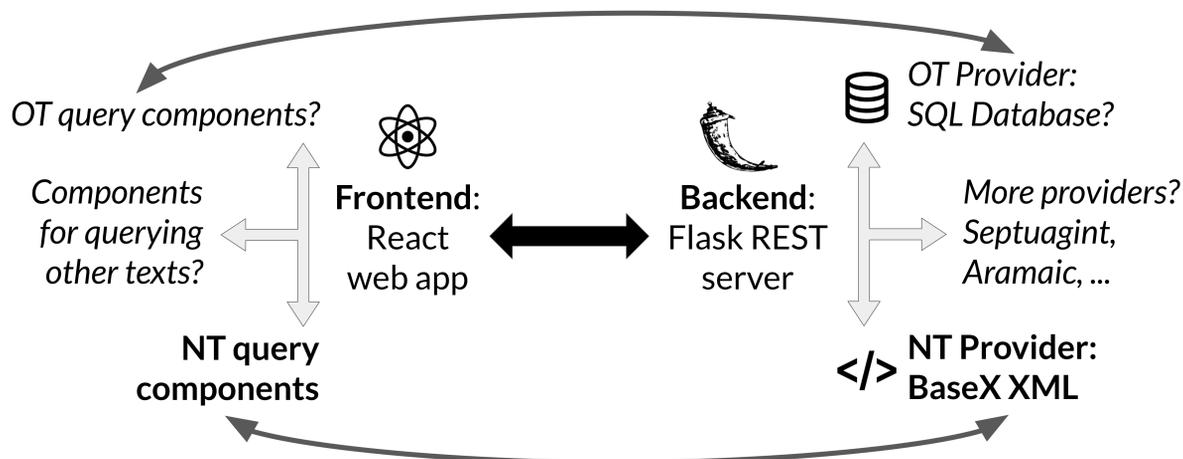
I wanted my implementation to make it quick and easy to:

- Make major changes to the design of the app
- Add new searchable Biblical texts with data stored in different formats
- Have custom UI components to build queries for different Biblical texts

So I built an architecture (see the diagram on the right) with a:

- Decoupled backend (for handling the actual searches) and frontend (for the user to design searches and look at the results)
- Extensible interface-based architecture for the backend to talk to different “text providers”
- Flexible mapping system to define frontend components for each text provider and the kinds of queries it supports

I implemented the backend and most of the frontend for NT Greek searches.



- 📄 2161 lines of code written
- 🔧 5 languages used
- 🗨️ 73 feature requests opened
- ✅ 23 feature requests closed

Conclusion

The project is well on its way with a tested design, a fully working backend, and most of the frontend for NT Greek searches. However, there are still many opportunities for future work filling out the frontend, supporting OT Hebrew searches, and deploying the app on the Internet.

I would like to thank Dr. Nurkalla for making this project possible and Dr. Heth for giving me lots of advice and encouragement. Dr. MaGee, Dr. Young, Dr. Flanders, Drew Shriner, Isaiah Swain, and Jessica Dundas also offered invaluable feedback, and Andrew Blomenberg gave me very useful Python advice.