

# Project Proposal

## DLTI Expansion

### Group Members

- William Wu
  - o 4th Year CS Major
  - o Java, Python, React, HTML, CSS, JavaScript, SQL
  - o Task : Front End
  
- Will Smith
  - o 3rd Year CS Major
  - o Java, Python, JavaScript, HTML
  - o Task : Database and GUI
  
- Bandhan Patel
  - o 3rd Year CS Major
  - o Java , Python , SQL, MongoDB
  - o Task : Database and Backend

### Project Goals and Description

#### Problems to Solve / Description

The previous year, the DLTI Team was successfully able to create a web GUI corresponding to the MatLab DLTI demo. The user could select from differing DLTI filters and modify parameters to visualize differences between input and output graphs as well as intermediate data. However, we want to be able to save the states of the GUIs to record the

students' answers, opening the door to possible grading automation among other benefits. Our goal is to log the parameters of GUIs into a database so that students and professors can recreate or access the relevant data of these JSX Graphs. The saved states will allow instructors to easily view and sign-off on these completed labs or create an automated grading process. Furthermore, with the saved state information in a database, we can gather data points for the students' answers. We can analyze the data to for possible amendments to the lab in the future. To implement this, we will try to use Node.js with Express to integrate a MySQL database into our project. Eventually, we hope to unite all of this work under a standardized React framework for converting labs from MatLab to web apps. We also hope to be able to finish work implementing Mathjax into our app to better represent mathematical functions, something the team was not able to finish the previous semester.

### Project Timeline

Week	Task
Week 5, 6	New members get ramped up with the previous code. Understand the HTML and JavaScript that is used in the current DLTI Demo.
Phase 1: Resource set-up,	Examine old techniques and explore new options for creating, logging to, and maintaining a database for the project.  <b>Week 6: Have tools selected and begin implementation</b>
Week 7, 8, 9, 10, 11	Begin integrating the database into the project.
Phase 2: Database	<b>Week 8: have some elements of demo able to save/restore from DB.</b>

implementation and state saving	Setup a DB on VIP servers that can be accessed by the web app. <b>Week 10: Operation off non-localhost</b>
Week 12, 13 Phase 3: Front – end design	Begin looking at how to optimize and improve GUI using React, MathJax, and other tools. <b>Week 13: Have finished beta demo</b>
Week 14 Phase 4: Testing	Finalize the project and continue testing that the features work in the correct test cases.
Week 15	Adding any final small changes to the project
Week 16	Final Presentation

### Potential Issues

- Creating the methods to save and restore states
- The database should efficiently store and retrieve data in a timely manner
- Setting up a reliable host for the database and the website
- Learning React and other frameworks
- Learning and using Node.js and Express, since none of our members have used it before

### Implementation Tools and Resources

- GitHub: <https://github.gatech.edu/VIP-ITS>

- W3Schools: <https://www.w3schools.com/>
- Project Documentation Notebook
- JSXGraph Demos: <http://its.vip.gatech.edu/VIP/demos/>
- Flask Documentation: <http://flask.pocoo.org/docs/1.0/>
- Express Documentation: <https://expressjs.com/en/5x/api.html>
- Nodejs Documentation: <https://nodejs.org/en/docs/>
- Mathjax Documentation: <https://www.mathjax.org/#docs>
- MySQL Documentation: <https://dev.mysql.com/doc/>