

# JavaScript Intelligent Tutor

VIP – ITS Spring 2021

## Team Members

Vikas Barevadia vbarevadia3@gatech.edu	Michael Keohane mkeohane@gatech.edu	Jinwoo Park jpark955@gatech.edu
Colin Cassell col.cassell@gatech.edu	Chinmayi Kompella ckompella3@gatech.edu	Krishan Patel kpatel648@gatech.edu
Roshni Dhanasekar rdhanasekar3@gatech.edu	Khushi Magiawala kmagiawala3@gatech.edu	Lisa Walkosz-Migliacio evilisa@gatech.edu

## Main Problem to Solve

Programing complex functions in Javascript. Not only will the student learn how to write Javascript, but how the DLTI system mathematically works.

## Main Idea

UI that allows students to code the DLTI systems in Javascript resulting in a graphical visualization of the code. Students are given feedback based on the syntax of their code and functionality is validated.

## Details

Create a user interface that has questions how to program mathematical algorithms in a code editor as input for students.

The code editor may give good feedback on syntax errors in their code in a natural language message as opposed to a computer language.

Output the code into a visualization such as the [DLTI graphs](#).

Create test cases that validate how well the students did on the code they provided. These test cases will check not only functionality but completeness.

Create an API with Database that will serve the questions we will ask of the students. It will also hold their scores of how well they did on the questions.

Create a view that shows how well the student did on the questions and where they could return to practice again.

## Interesting Ideas to be Researched

How students best learn a coding language when much of the debugging is in exception errors.

How students best learn a coding language when given multiple test cases for validation and practicing those failures again.

How students best learn a mathematical formula from code and graphical visualization.

## **Technology**

### **UI**

[Javascript Node, React](#)

[CodeMirror](#) for code editor

Output of [DLTI](#) sinusoidal signal graphs

Natural language of Javascript [error messages](#)

[Material UI](#)

### **API**

[Node Javascript](#)

[Cypress](#) / [Jest](#) Test Cases to validate code snippets

### **DB**

[MongoDB](#) (Tier M0 Atlas Cluster)

## **Timeline Breakdown**

***Milestone 1*** *Week 5 (2/8) - Week 9 (3/8) General planning and initializing*

Week 5

- Initialize node app and connect to DB
- Research natural language programming, TDD, other web-based code editors

Week 6

- Connect to DB
- Create Schemas
- Research DSP equations, more code editors, UI Framework

Week 7

- Stub in API calls for each needed Schema GET/POST

Week 8

- Wireframes for views

Week 9

- Stub out routes for views

***Milestone 2*** Week 10 (3/15) to Week 16 (4/26) Finalizing design and presentation prep

Week 10

- Import Codemirror UI make component (maybe we use [Ace](#) or [monaco](#) editor instead)

Week 11

- Implement wireframes

Week 12

- Build out test cases and their values

Week 13

- Build out error handling messages

Week 14

- Import the DLTI Javascript functions

Week 15-16

- Presentation preparation
- Finalize tests and possible user beta testing

## **Who's Responsible for What**

### **Backend Team**

Colin Cassell

- Relevant Skills: Java, Python, JavaScript(minimal), some experience with both mySQL and MongoDB
- Responsibilities: Would prefer to work on API / DB

Mike Keohane

- Relevant Skills:
  - o Frontend: JavaScript, HTML / CSS, ReactJS
  - o Backend: Python, SQL/SQLite
- Responsibilities: Would prefer to work on backend / API (can learn MongoDB), but this can be discussed more in our meeting.

### Chinmayi Kompella

- Relevant Skills: Java, JavaScript(beginner), Python(beginner)
- Relevant coursework: CS 1331, CS 1332, CS 2340
- Responsibilities: I am interested in working on the backend with the API and mySQL or MDB to pick up experience in this area.

### Krishan Patel

- Relevant Skills: HTML / CSS, Java, JavaScript (minimal)
- Relevant Courses: CS 1331, CS 1332, CS 2340
- Preferred Responsibilities: I believe I would be able to provide more input towards frontend/UI things, but I am interested in working on backend processes just so I can get some experience/a feel of how those things work

### Jin Park

- Relevant Skills: Java, C, HTML/CSS(beginner), MySQL(minimal)
- Relevant Coursework: CS 1331, CS 1332 CS 2340, CS 4400
- Preferred Responsibilities: Would like to learn Backend and database/API interaction with Node.js

## **UX Team**

### Vikas Barevadia

- Relevant Skills: some frontend knowledge in HTML, CSS, vanilla JS, node.js
- Responsibilities: Would prefer to work on frontend to learn react framework.

### Roshni Dhanasekar

- Relevant Skills: Have some experience with HTML, CSS, Java, Python, JavaScript (basic)
- Relevant Coursework: CS 1331, 1332, 2340
- Responsibilities: I would prefer to work on the frontend team and learn React/Node.js

### Khushi Magiawala

- Relevant Skills: Java, some HTML and Python
- Relevant Coursework: 1331, 1332, 2340
- Area of interest: React, Node.js (want to learn front-end)

### Lisa Walkosz-Migliacio

- Relevant Skills: Professional Website Experience (all tech in the stack)
- Responsibilities: Maker of the Issues, Keeper of the Pull Requests, Teacher of the Github, Anything UI

## Assignments By Week (to be updated week by week)

### Week 5

#### Backend Work

- Everyone - Get started with Github for those unfamiliar, clone repo, learn pull request process.
- Khushi - Init React UI put in repo
- Mike - Init node app for API put in repo
- Colin & chinmayi - Create DB put in repo (mongodb)

#### Research

- Krishan - Research how programming languages can be easier to understand using natural language.
- Jin - Research how reviewing failed test cases can improve understanding.
- Passed to future week - Research how coding with visual feedback can help in understanding.

#### UX brainstorm

*Let's do some HCI, videos from the [Georgia Tech HCI Masters class](#) are online, but a little odd to find. Find the [3.3 Other Data Gathering Methods](#) and look at Existing UI evaluation. [Codepen](#), [Khan Academy](#), [Github Classroom](#), [Glitch](#), [Codesandbox](#), [Hackerrank](#), ... so many more*

- Roshni – codepen & misc other
  - o We want to do a [3.6 cognitive walkthrough](#) for our existing interfaces listed above (or any other we know of).
  - o If you need some help on [2.2 gulfs of execution and evaluation](#).
  - o After we think about how a user would use other online coding editors, and how we can solve the task of learning to code the functions in Javascript, write down your ideas of how the current interfaces would work well, and how they might need help. Take screenshots to explain each of those important views. We will then use these to [3.5 wireframe prototype](#).
  - o Test cases results interface (how to display failed tests as a learning opportunity)
- Vikas – junit, other online editors that run test cases
  - o Look at existing UI evaluation. Honestly, these interfaces are pretty bad to understand what happened.
  - o Cypress, Unit Tests Jest, junit... so many more

## Week 6

### General

- Researcher extraordinaire - ie, take our research and write up so we can present our ideas more effectively, will do UX survey write up and justify our project in the end
- Keeper of meetings
- Keeper of the git/issues/project

### API/DB

- Chinmayi & colin Db schema
- Mike - connect to mongodb and read/write
- Jin - The math functions we want to teach, how to break them up for our users to program them (need to get them from Instructor Greg, we may need more research around this or some trial and error)
- Mike - <https://github.gatech.edu/VIP-ITS/DLTI-db> & <https://github.gatech.edu/VIP-ITS/DisConvDemo>
- Jin - The exceptions that could happen in the math functions or by programming - eventually end up writing more natural language for them

### UX/UI

- Kushi & roshni - code edits online needfinding leetcode, hackerrank
  - o Code Editor Wireframe/Implementation view(s) orchestrator - one of the main features we want to present as intelligently tutoring
- Vikas - github pull request code review
  - o Test case results - Wireframe/Implementation view(s) orchestrator - one of the main features we want to present as intelligently tutoring
  - o Wireframe view(s) of Overall flow - this will probably come after we have our two main improvements done, this is to make the entire experience work in the end, but not part of the main function