Proof Buddy

Visit our public website below! No VPN required

proofbuddy.cci.drexel.edu

Meet the Team

Member	Role
Nicole Itchon	Team lead, QA lead
Raphael Perez	Full stack developer, UI/UX designer
Viet Pham	Back-end developer
Iftekhar Rahman	Back-end developer
Steve Earth	Stakeholder
Jeremy Johnson	Stakeholder

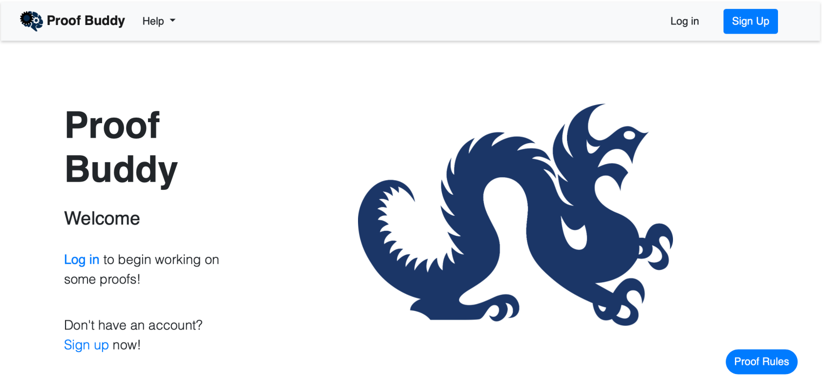
GitHub Repository

- ProofBuddy code:
  - https://github.com/se691-group3/proof-buddy
- ProofBuddy documentation:
  - https://github.com/se691-group3/proof-buddy/tree/main/docs

NICOLE

Description and Justification

- Proof Buddy is an educational tool for teaching computer science students proof techniques and logical reasoning
- Web-based and designed for use by both instructors and students
- Features include:
    - User authentication
    - Course creation and add students to course
    - Assignment and problem creation
    - Solve proofs and obtain feedback as you solve
- Currently does proofs in Boolean Logic and First Order Logic
    - Plans for extension to do Equational Reasoning



Proof Buddy

Welcome

Log in to begin working on some proofs!

Don't have an account? Sign up now!

Currently available at https://proofbuddy.cci.drexel.edu

NICOLE

Changes Added this Term

- Five high-priority features:

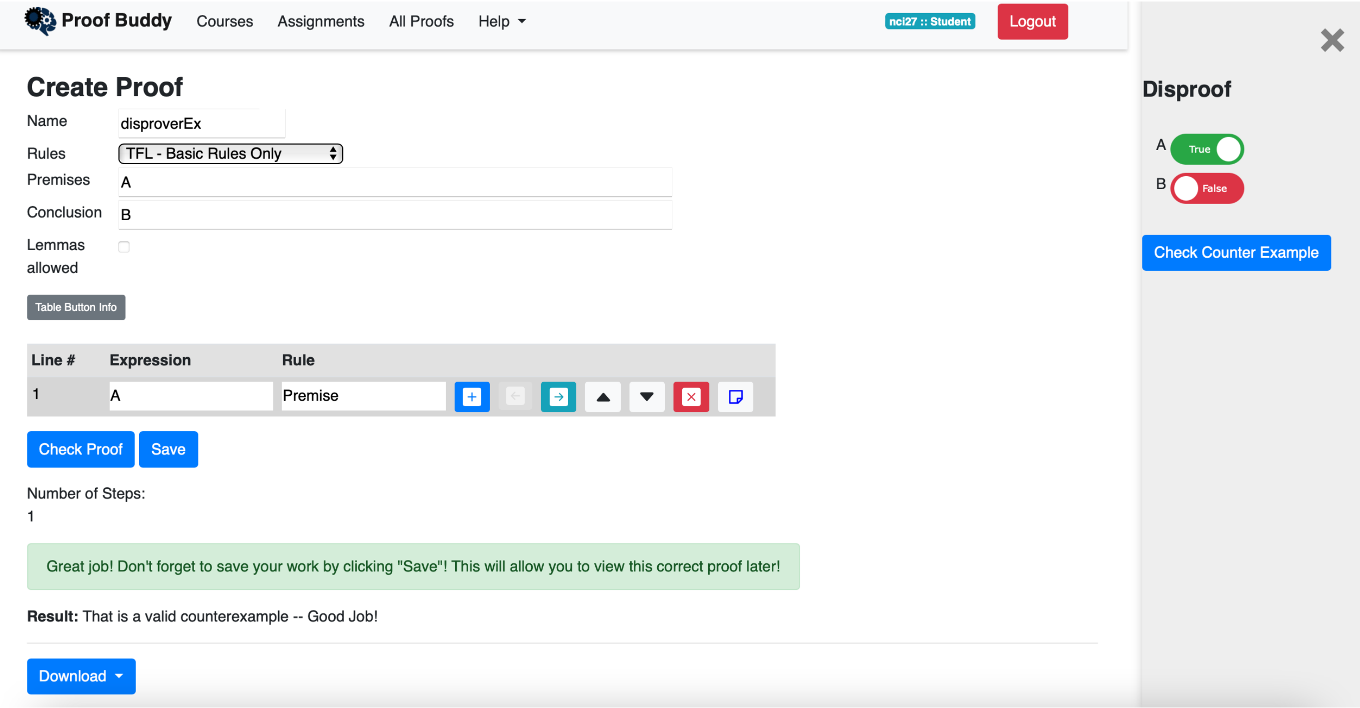
1.	using a saved proof – instructor must say whether allowed, student must have correct proof saved under certain name
2.	“disprove" mode – the user enters in T/F for each variable, and ProofBuddy checks that all the premises evaluate to True, but the conclusion is False
3.	a button which inserts a comment line (i.e. the user can type any text they wish: this line is fully ignored by the parser and does not affect the step count for the proof)
4.	let the instructor import a csv of userids, email_address and then that creates en masse a bunch of student accounts registered for that course
5.	exportation into LaTex

Feature 1: Lemmas

- Proofs which have been correctly proven and saved, can be used as lemmas in different proofs
  - Instructor has option to allow/restrict this on assignments
- Student proofs which are not correct, or valid but incomplete will not usable
  - Appropriate error messages are raised
- Lemmas, which are proved using derived rules, cannot be used in proofs for which derived rules are not permitted

Feature 2: Disprove

- Allows users to practice disproof problems
- Given a premise and conclusion, ProofBuddy checks that all premises evaluate to True, but conclusion is False



Create Proof

Name: disproverEx

Rules: TFL - Basic Rules Only

Premises: A

Conclusion: B

Lemmas allowed ☐

Line #	Expression	Rule
1	A	Premise

Number of Steps:
1

Great job! Don't forget to save your work by clicking "Save"! This will allow you to view this correct proof later!

Result: That is a valid counterexample -- Good Job!

Disproof

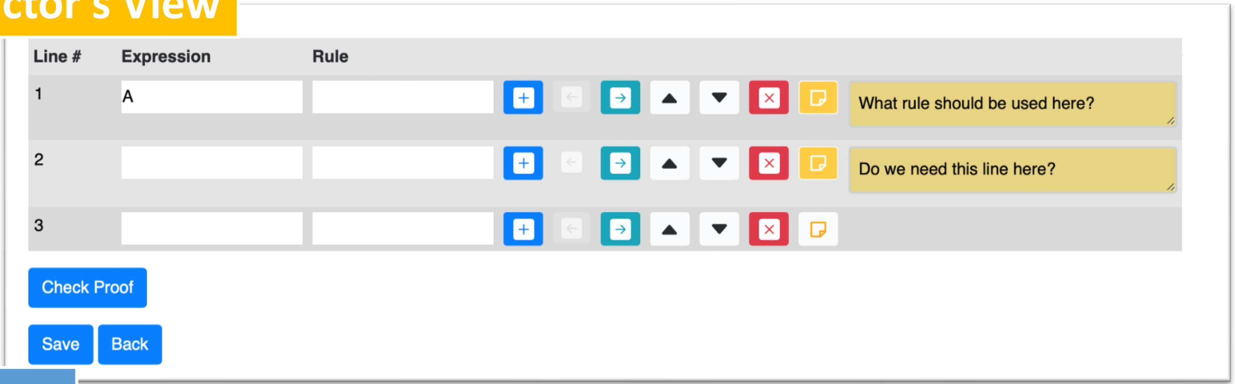
A: True

B: False

NICOLE

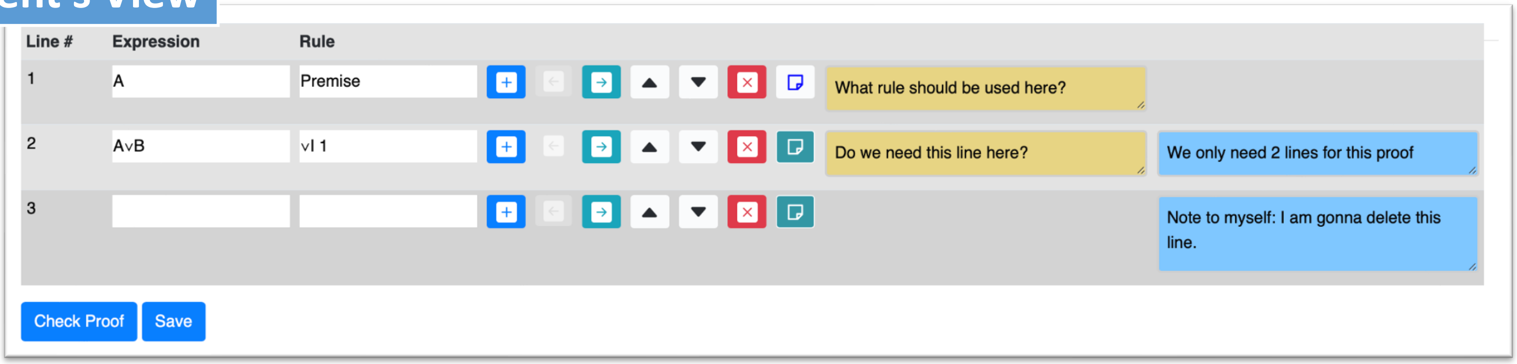
Feature 3: Student-Instructor Communication

Instructor's View



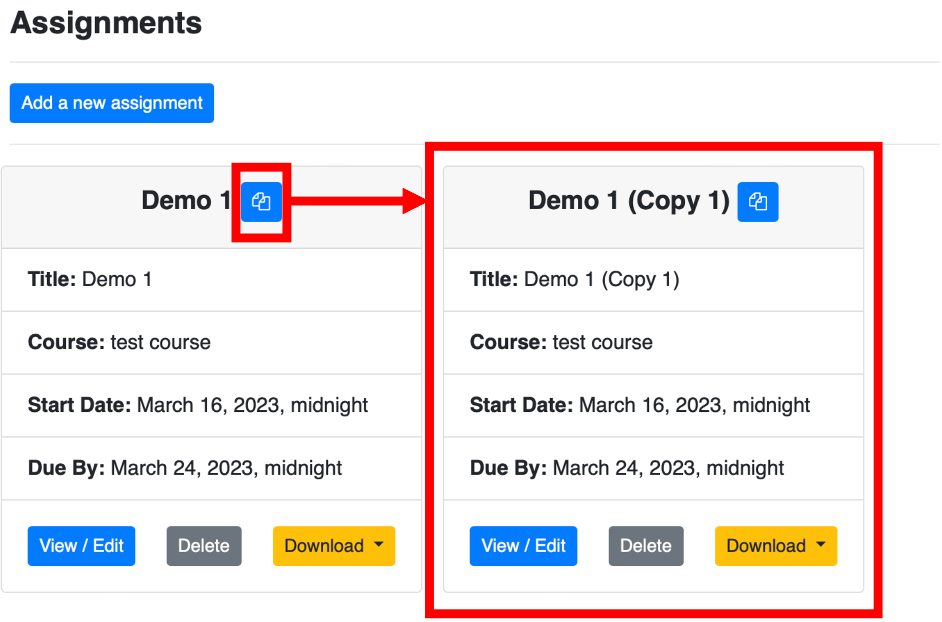
Line #	Expression	Rule	
1	A		What rule should be used here?
2			Do we need this line here?
3			

Student's View



Line #	Expression	Rule		
1	A	Premise	What rule should be used here?	
2	AvB	vI 1	Do we need this line here?	We only need 2 lines for this proof
3				Note to myself: I am gonna delete this line.

Assignment Duplication



Assignments

Demo 1

Title: Demo 1

Course: test course

Start Date: March 16, 2023, midnight

Due By: March 24, 2023, midnight

Demo 1 (Copy 1)

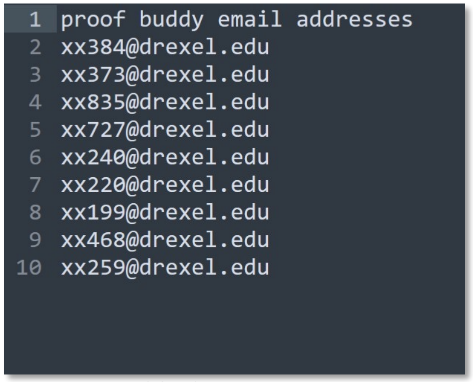
Title: Demo 1 (Copy 1)

Course: test course

Start Date: March 16, 2023, midnight

Due By: March 24, 2023, midnight

Feature 4: CSV Upload Process



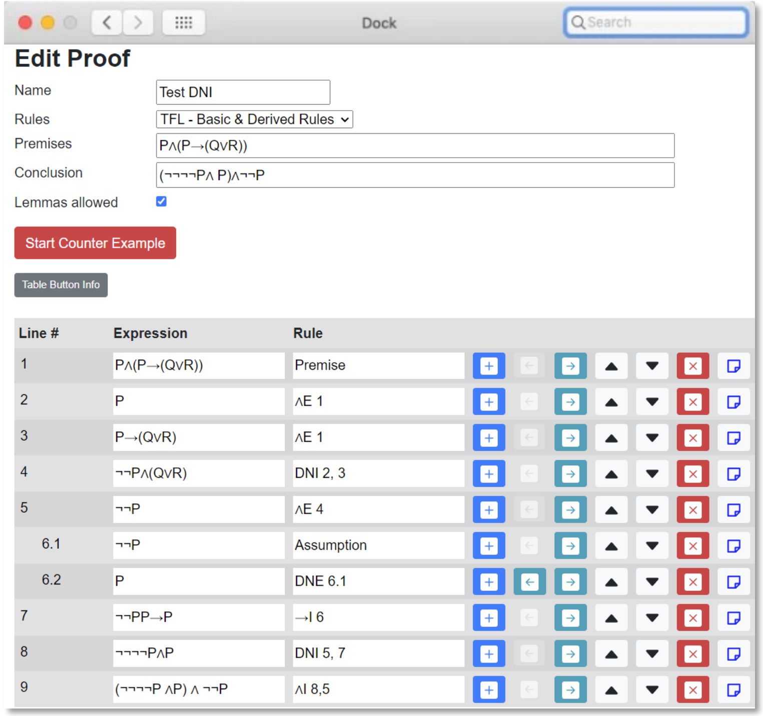
```
proof buddy email addresses
xx384@drexel.edu
xx373@drexel.edu
xx835@drexel.edu
xx727@drexel.edu
xx240@drexel.edu
xx220@drexel.edu
xx199@drexel.edu
xx468@drexel.edu
xx259@drexel.edu
```

An example CSV file with email addresses.

- A demo will be shown showing how this works

RAPH

Feature 5: LaTeX export



Edit Proof

Name: Test DNI

Rules: TFL - Basic & Derived Rules

Premises: P∧(P→(Q∨R))

Conclusion: (¬¬¬¬P∧ P)∧¬¬P

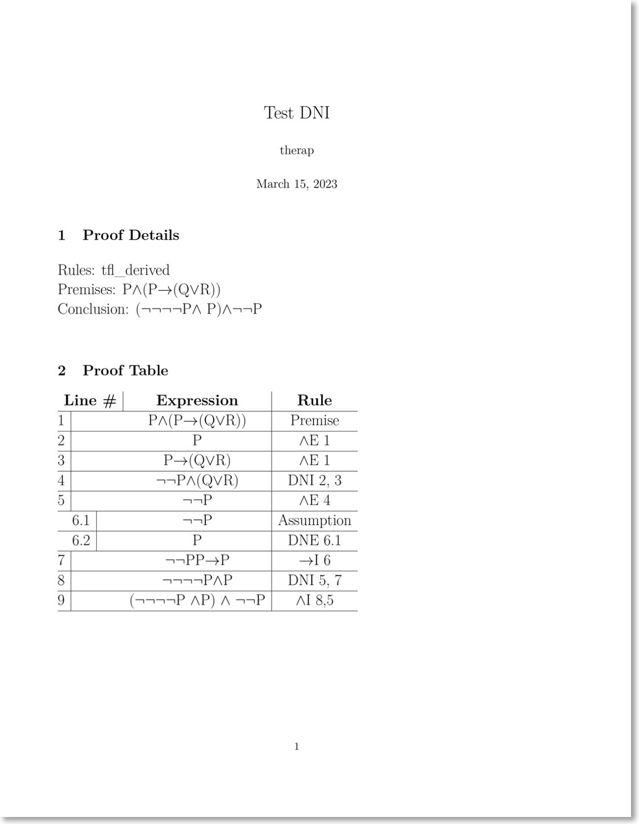
Lemmas allowed ☑

Line #	Expression	Rule
1	P∧(P→(Q∨R))	Premise
2	P	∧E 1
3	P→(Q∨R)	∧E 1
4	¬¬P∧(Q∨R)	DNI 2, 3
5	¬¬P	∧E 4
6.1	¬¬P	Assumption
6.2	P	DNE 6.1
7	¬¬PP→P	→I 6
8	¬¬¬¬P∧P	DNI 5, 7
9	(¬¬¬¬P ∧P) ∧ ¬¬P	∧I 8,5

We added a function to let users export their proofs to a LaTeX .tex file they can view using most standard LaTeX viewers and compilers.

RAPH

Overleaf



Test DNI

therap

March 15, 2023

1 Proof Details

Rules: tfl_derived
Premises: P∧(P→(Q∨R))
Conclusion: (¬¬¬¬P∧ P)∧¬¬P

2 Proof Table

Line #	Expression	Rule
1	P∧(P→(Q∨R))	Premise
2	P	∧E 1
3	P→(Q∨R)	∧E 1
4	¬¬P∧(Q∨R)	DNI 2, 3
5	¬¬P	∧E 4
6.1	¬¬P	Assumption
6.2	P	DNE 6.1
7	¬¬PP→P	→I 6
8	¬¬¬¬P∧P	DNI 5, 7
9	(¬¬¬¬P ∧P) ∧ ¬¬P	∧I 8,5

1

Public Access

Before

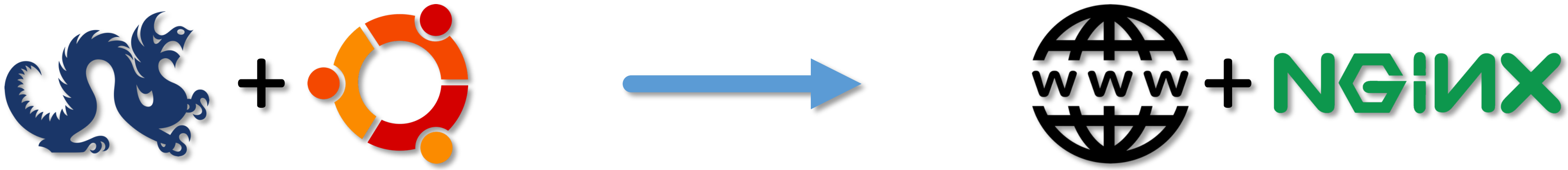
- The site could only be accessed via Drexel VPN using a web address preassigned by a Drexel Admin
- The Drexel VPN could only be access by either Drexel students or faculty.

Addressing the issue

- Created the TLS Certificate
- Configured Nginx to Use SSL
- Adjusted the Firewall to redirect traffic to HTTPS

After

- The website proofbuddy.cci.drexel.edu can be access in the public internet on any device, even a phone
- Other universities can use it in the future



RAPH

System Requirements

Functional requirements

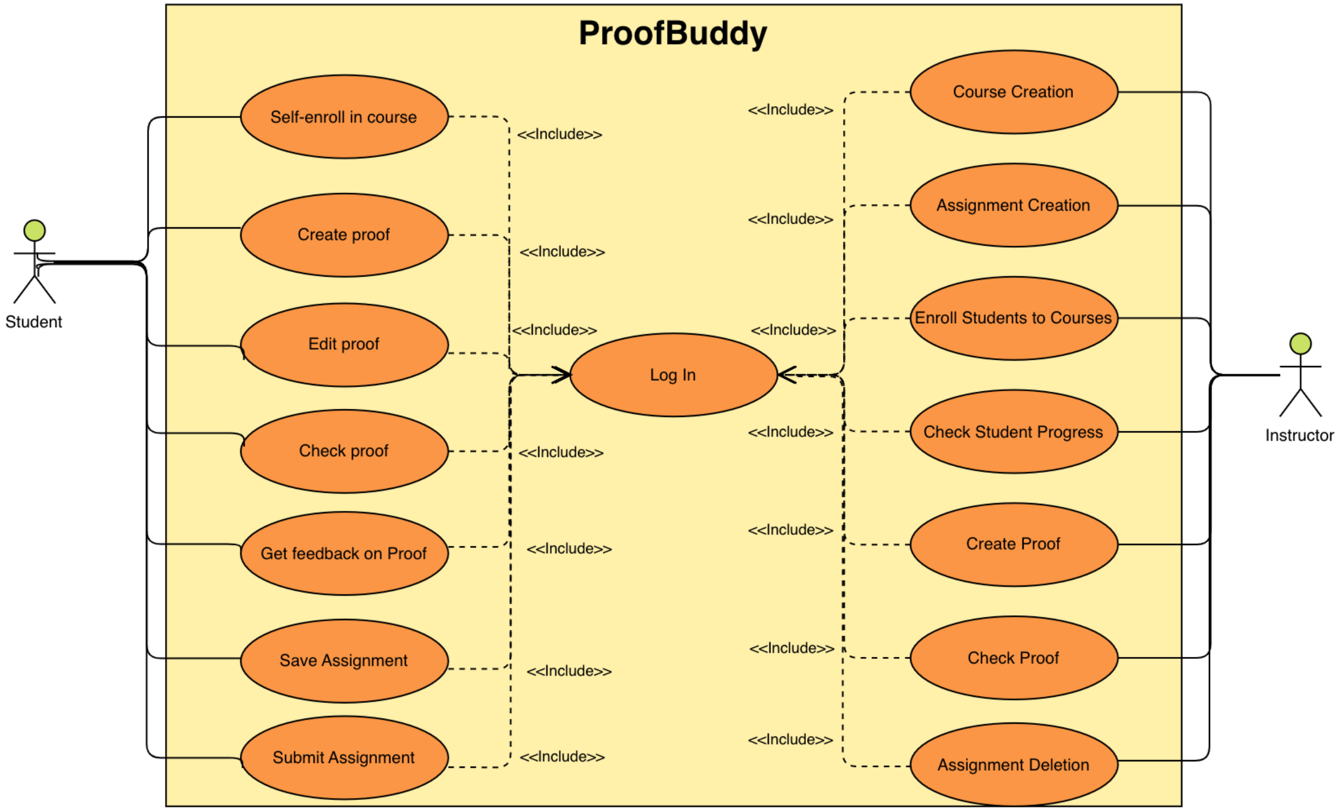
- Allow access to users with verified email address
- Store courses, assignments, and proofs
- Associate students with courses
- Associate problems and courses with assignments
- Verify that proofs are correct
- Provide feedback when proofs contain an error or is incomplete
- Grade assignments

Usability requirements

- Allow instructors to create/edit/delete courses, assignments, and problems
- Courses must have a title, term, section, and students
- Assignments must have a title, course, start date, due date, and problems
- Problems must have a title, points, target steps, lost points, rules, premise, and conclusion
- Allow users to restart, check, or save proofs
- Allow students to submit assignments
- Allow instructors to view student progress on proofs

NICOLE

Use Cases



IFTY

Architecture

4 Main Components (apps)

Accounts

- Account creation
- User Authentication
- Permissions

Courses

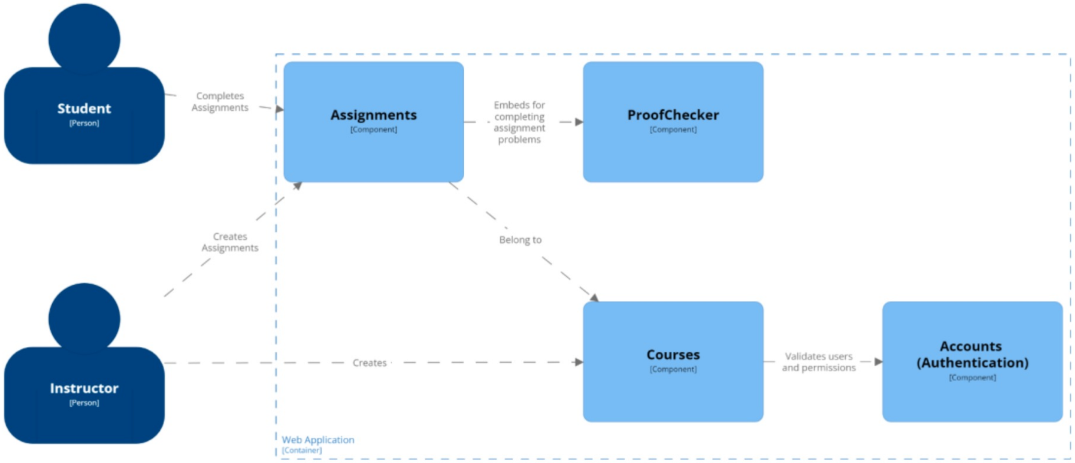
- Instructors create courses and enroll students
- Students self-enroll

Assignments

- Instructors create assignments for courses
- Students complete/submit assignments

ProofChecker

- Users create, edit, delete and save proofs
- Provides feedback and validation of the proofs submitted to the tool

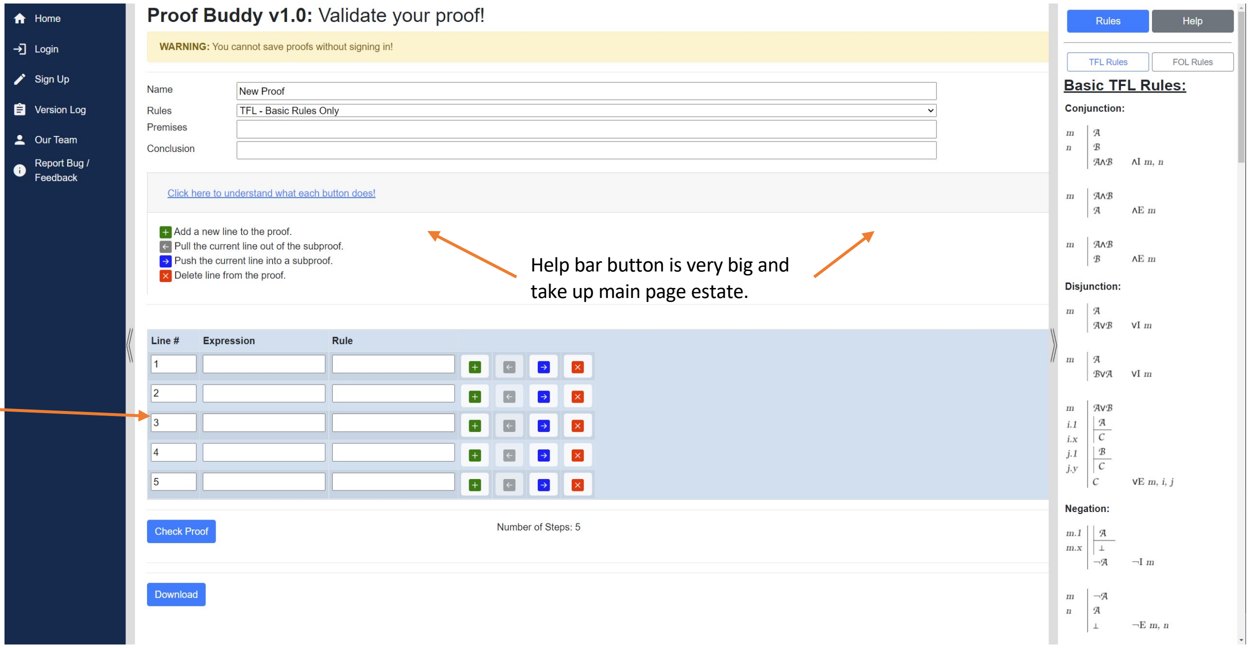


Design and Implementation

The old design as shown was a UI for an app that mainly focused on the functionality of the app and added UI elements as a necessity over form.
The app is fully functional, but some UI elements can be improved.

Left-menu bar that is collapsed by clicking on the thin grey bar and takes up page estate.

Proof buddy engine with some padding issues, the table will either run of page or have an unnecessary scroll bar and the line number are not editable though it appears to be in this version.



Proof Buddy v1.0: Validate your proof!

WARNING: You cannot save proofs without signing in!

Name	New Proof
Rules	TFL - Basic Rules Only
Premises	
Conclusion	

Click here to understand what each button does!

- 🟩 Add a new line to the proof.
- ⬅ Pull the current line out of the subproof.
- ➡ Push the current line into a subproof.
- 🟥 Delete line from the proof.

Line #	Expression	Rule
1		
2		
3		
4		
5		

Number of Steps: 5

Basic TFL Rules:

Conjunction:

m | $\mathcal{A}$
n | $\mathcal{B}$
| $\mathcal{A} \wedge \mathcal{B}$ $\quad \wedge\text{I } m, n$

m | $\mathcal{A} \wedge \mathcal{B}$
| $\mathcal{A}$ $\quad \wedge\text{E } m$

m | $\mathcal{A} \wedge \mathcal{B}$
| $\mathcal{B}$ $\quad \wedge\text{E } m$

Disjunction:

m | $\mathcal{A}$
| $\mathcal{A} \vee \mathcal{B}$ $\quad \vee\text{I } m$

m | $\mathcal{A}$
| $\mathcal{B} \vee \mathcal{A}$ $\quad \vee\text{I } m$

m | $\mathcal{A} \vee \mathcal{B}$
$i.1$ | | $\mathcal{A}$
$i.x$ | | $\mathcal{C}$
$j.1$ | | $\mathcal{B}$
$j.y$ | | $\mathcal{C}$
| $\mathcal{C}$ $\quad \vee\text{E } m, i, j$

Negation:

$m.1$ | | $\mathcal{A}$
$m.x$ | | $\bot$
| $\neg\mathcal{A}$ $\quad \neg\text{I } m$

m | $\neg\mathcal{A}$
n | $\mathcal{A}$
| $\bot$ $\quad \neg\text{E } m, n$

Help bar button is very big and take up main page estate.

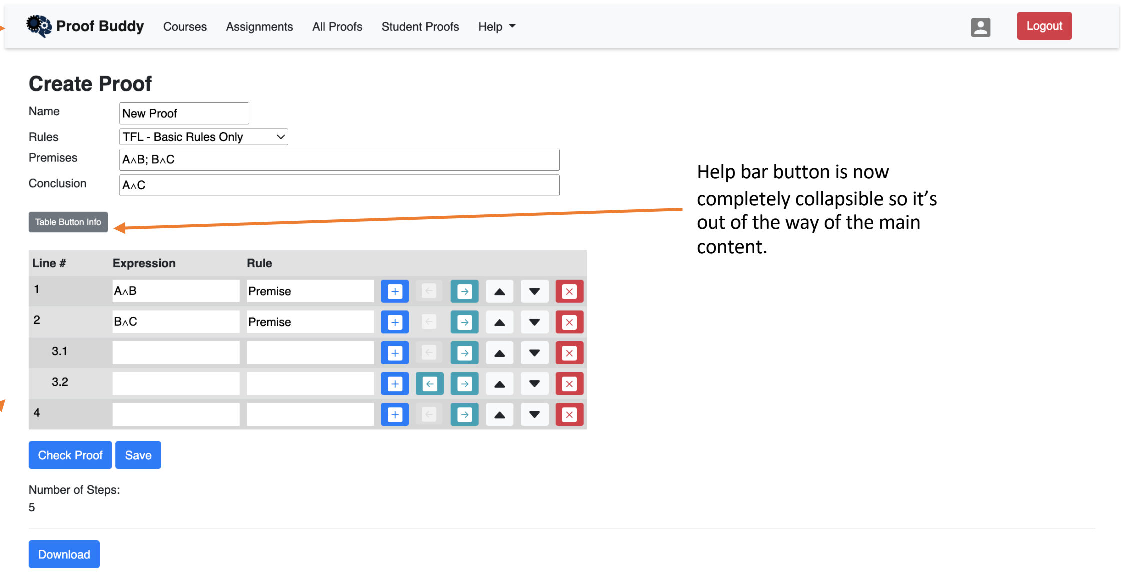
Right-menu bar that is *also* collapsed by clicking on the thin grey bar and takes up page estate. It's ok if this menu takes up space because there are a lot of rules and they're necessary for the assignments.

RAPH

Design and Implementation

The new design is now more focused not only bringing the application to modern design standards, but also allowing the app to make more use of the web page estate to make the user find UI elements more easily.

Left-menu bar has been retooled as a sticky top nav bar that does not get in the way of any tables or text.



Create Proof

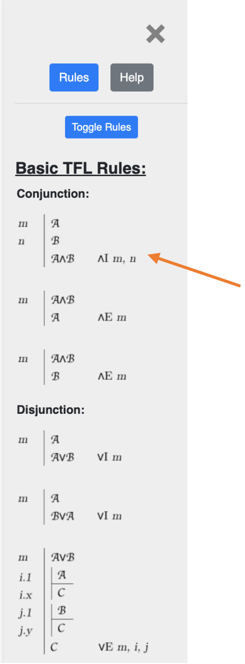
Name	New Proof
Rules	TFL - Basic Rules Only
Premises	A∧B; B∧C
Conclusion	A∧C

Line #	Expression	Rule
1	A∧B	Premise
2	B∧C	Premise
3.1		
3.2		
4		

Number of Steps:
5

Help bar button is now completely collapsible so it's out of the way of the main content.

Proof buddy engine with padding fixes, clearer color definitions for the buttons and its functionality and line numbers do not appear editable for better UI clarity.



Basic TFL Rules:

Conjunction:

$$\begin{array}{l|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{A}\land\mathcal{B} & \land\text{I } m, n \end{array}$$

$$\begin{array}{l|ll} m & \mathcal{A}\land\mathcal{B} & \\ & \mathcal{A} & \land\text{E } m \end{array}$$

$$\begin{array}{l|ll} m & \mathcal{A}\land\mathcal{B} & \\ & \mathcal{B} & \land\text{E } m \end{array}$$

Disjunction:

$$\begin{array}{l|ll} m & \mathcal{A} & \\ & \mathcal{A}\lor\mathcal{B} & \lor\text{I } m \end{array}$$

$$\begin{array}{l|ll} m & \mathcal{A} & \\ & \mathcal{B}\lor\mathcal{A} & \lor\text{I } m \end{array}$$

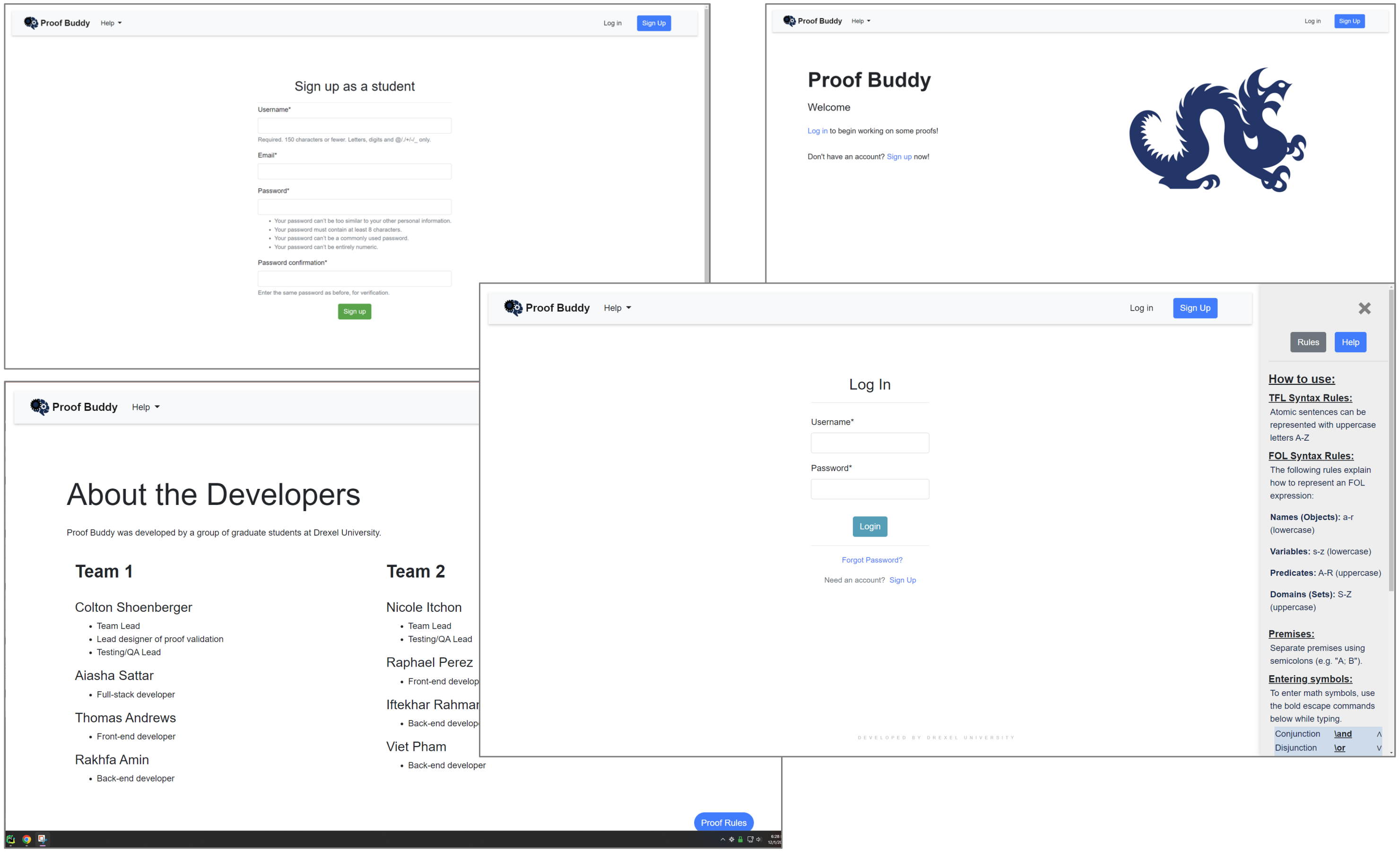
$$\begin{array}{l|ll} m & \mathcal{A}\lor\mathcal{B} & \\ i.1 & \quad\mathcal{A} & \\ i.x & \quad\mathcal{C} & \\ j.1 & \quad\mathcal{B} & \\ j.y & \quad\mathcal{C} & \\ & \mathcal{C} & \lor\text{E } m, i, j \end{array}$$

Right-menu bar now opened and closed with big, easy to find UI elements including the 'X' on top and users can now toggle rules on and off to only display the ones needed.

RAPH

Design and Implementation

The part of the right-menu bar that allows users to hide rules they don't currently need to reduce clutter.



Proof Buddy Help ▾ Log in Sign Up

Sign up as a student

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation*

Enter the same password as before, for verification.

Sign up

Proof Buddy Help ▾ Log in Sign Up

Proof Buddy

Welcome

Log in to begin working on some proofs!

Don't have an account? Sign up now!

Proof Buddy Help ▾

About the Developers

Proof Buddy was developed by a group of graduate students at Drexel University.

Team 1

Colton Shoenberger

- Team Lead
- Lead designer of proof validation
- Testing/QA Lead

Aiasha Sattar

- Full-stack developer

Thomas Andrews

- Front-end developer

Rakhfa Amin

- Back-end developer

Team 2

Nicole Itchon

- Team Lead
- Testing/QA Lead

Raphael Perez

- Front-end develop

Iftekhar Rahman

- Back-end develop

Viet Pham

- Back-end developer

Proof Rules

Proof Buddy Help ▾ Log in Sign Up

Log In

Username*

Password*

Login

Forgot Password?

Need an account? Sign Up

DEVELOPED BY DREXEL UNIVERSITY

Rules Help

How to use:

TFL Syntax Rules:

Atomic sentences can be represented with uppercase letters A-Z

FOL Syntax Rules:

The following rules explain how to represent an FOL expression:

Names (Objects): a-r (lowercase)

Variables: s-z (lowercase)

Predicates: A-R (uppercase)

Domains (Sets): S-Z (uppercase)

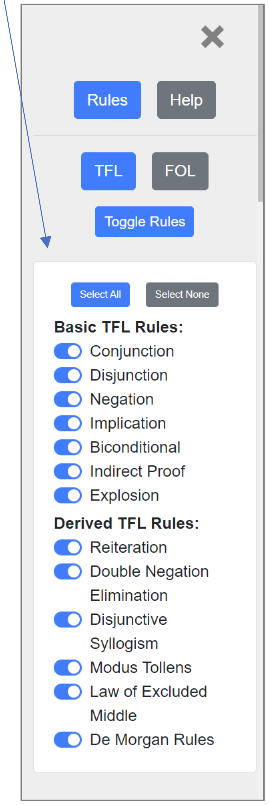
Premises:

Separate premises using semicolons (e.g. "A; B").

Entering symbols:

To enter math symbols, use the bold escape commands below while typing.

Conjunction	**\and**	∧
Disjunction	**\or**	∨



Basic TFL Rules:

- Conjunction
- Disjunction
- Negation
- Implication
- Biconditional
- Indirect Proof
- Explosion

Derived TFL Rules:

- Reiteration
- Double Negation Elimination
- Disjunctive Syllogism
- Modus Tollens
- Law of Excluded Middle
- De Morgan Rules

RAPH

Test Plan

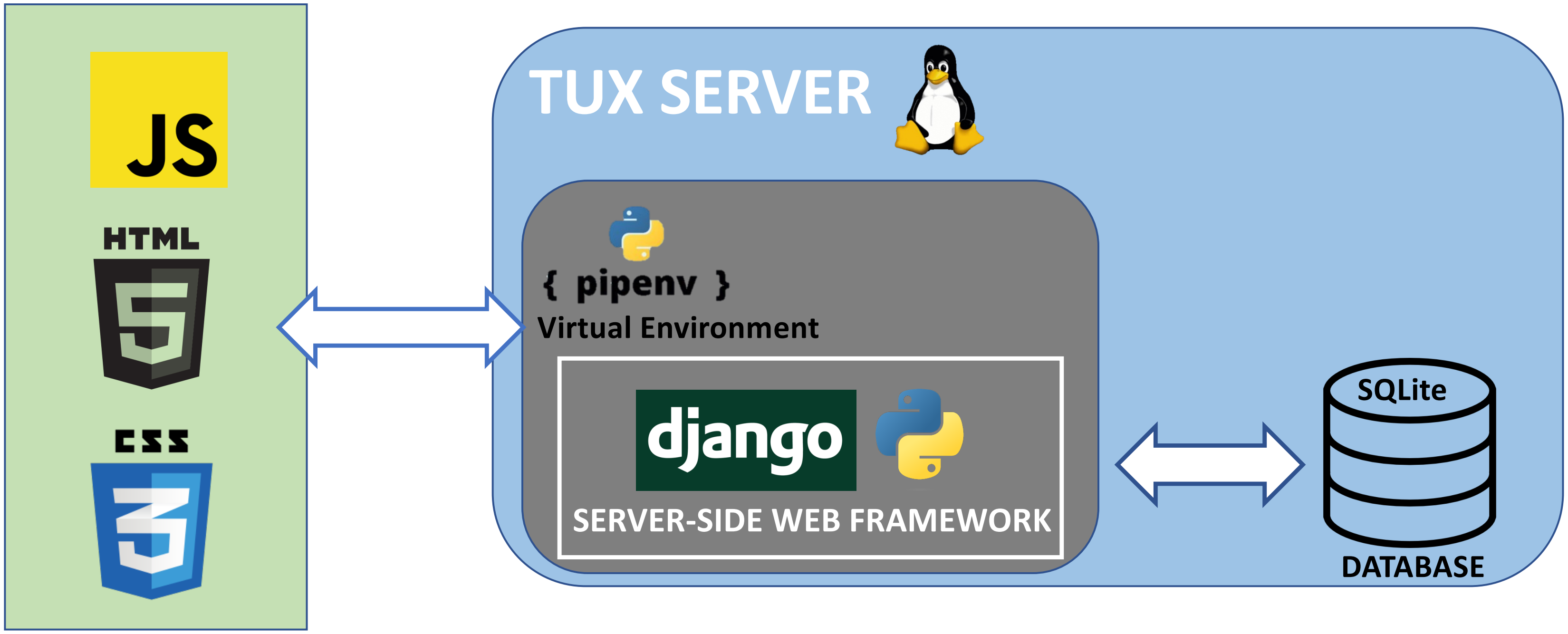
Currently

- Using **`unittest`** module, which is built-in to the Python standard library
- Can run **`python manage.py test`** within project to run existing tests
- Only tests basic functionalities and proofs
- Performed stress testing on public server

Improvements

- Extend testing as new functionalities are added
- Write tests to catch bugs at edge cases
- Implement continuous integration:
  - When developers create pull requests, have an automated build process that verifies code, runs test suite, and runs quality control checks

Supporting Technology - Framework



Accomplishments

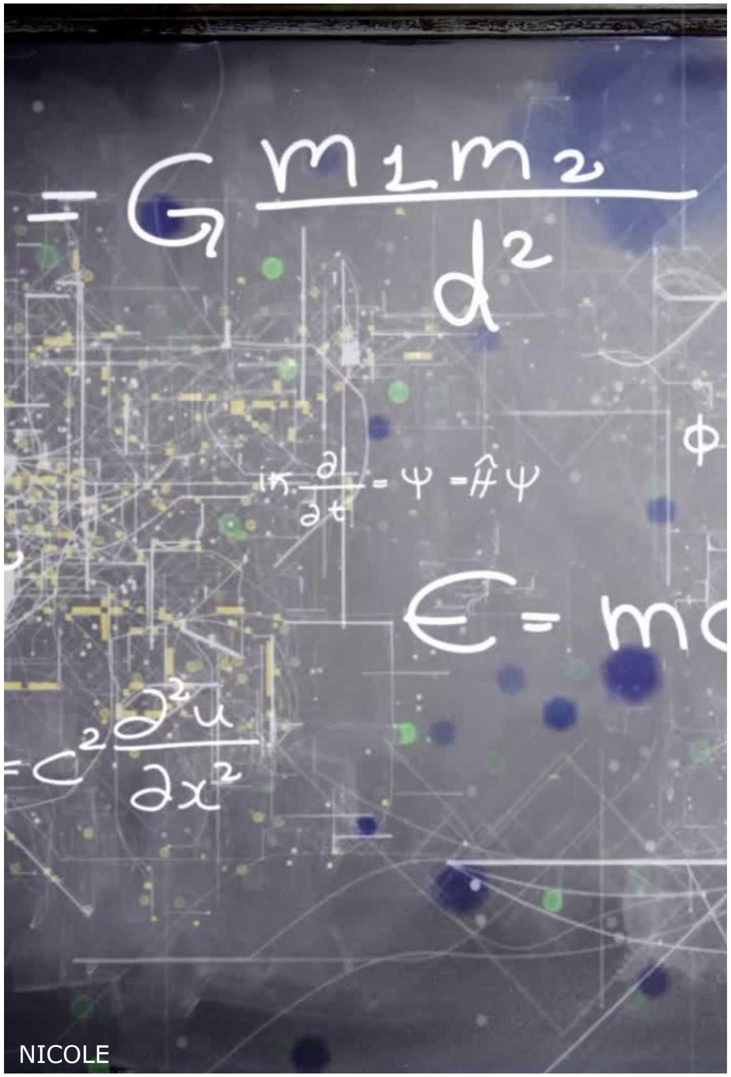
- Conference in Toronto
  - Stakeholder Steve was scheduled to attend conference in Toronto (currently there) and is going to demo ProofBuddy to attendees
- High priority features implemented
  - Steve focused the team on adding 5 high priority features before the end of the term
  - Team was able to complete 5 out of 5 high priority features,
  - Equational Reasoning to be the focus of the next team working on ProofBuddy

NICOLE

Lessons Learned

- Nicole
  - I learned the importance of time management. Since we were aware that our stakeholder was attending a conference, it was important that we implemented the features promised and still make time to manually test and fix bugs if needed
- Viet
  - I learned the benefits and importance of version tracking (e.g. Github) in a project that involved multiple developers.
- Ifty
  - Being able communicate errors and their fixes to support continuity of knowledge as project progresses to different developers.
- Raph
  - I learned it’s important to always clarify what the stakeholder wants because sometimes there can be miscommunications

NICOLE


$-G\frac{m_1m_2}{d^2}$
$i\hbar\frac{\partial}{\partial t}\Psi = \hat{H}\Psi$
ϕ
$E = m$
$= c^2\frac{\partial^2 u}{\partial x^2}$

NICOLE

Future Work

- Framework for Equational Reasoning

ProofBuddy Demo