

# # Code Critiquer in C

## # Client

Iowa State University  
and Michigan Tech University

## # Advisor

Dr. Diane T. Rover

## # TEAM 34

Nicholas Carber : Regex Support

Conner Cook : AST Support

Brandon Ford : Database Administrator

Emily Huisinga : Frontend

Sage Matt : Frontend

Cade Robison : Test Suite Support

# # Project Vision

- We serve to help novice C programmers learn C by creating an application for students to receive feedback on their code
- The feedback will consist of easy-to-understand and relevant error messages
- Teachers will be able to configure it for their own course



# # Conceptual/Visual Sketch

```
user@host:/tmp/segfault$ ./segfault  
Segmentation fault
```

Critiquer Feedback: On line 34,  
you are accessing an  
out-of-bounds index in array **arr**

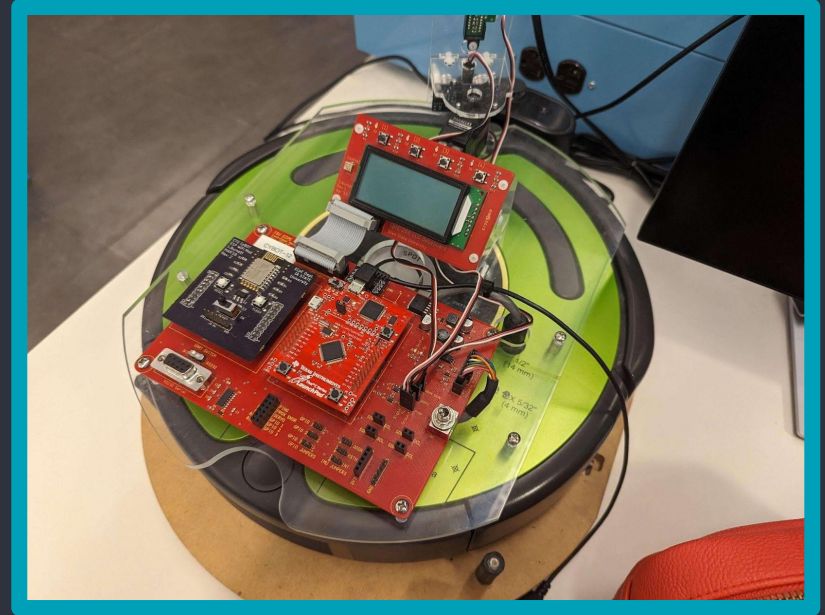


C Critiquer



# # Who will this affect?

- Students & novice programmers in C
  - Specifically CPR E 288 students
- Professors, teaching assistants, tutors



# # Important Terms

- **Antipatterns:** Poor solutions to common programming problems
- **Regexes:** A sequence of characters used to find patterns in text
- **Abstract Syntax Tree:** Tree representation of the syntactic structure of the code



# # Prototype Implementations

- Michigan Tech
  - Work in progress critiquers in other languages
    - Existing code bases reference
    - Existing regular expressions
    - Common antipatterns



# # MTU

# # MATLAB

# # Code

# # Critiquer

## WebTA Code Critique

**Program:** S11 #1 - Vector Creation and Modification  
**Name:** Leo Ureel  
**Created:** 2023-10-03 16:36:12.0  
**Critiqued Files:** S10.m

[Download source files from this critique.](#)

### Summary:



### EngTA found issues with your code. (See below.)

- There are 1 critical issues in your code.  
These issues must be fixed before your code will work as intended.
- There are 16 non-critical concerns about your code.  
These issues should be addressed to make sure your code is robust and maintainable.

## Critiques

#	Code	Critique
000	<code>% S11 #1 - Vector Creation and Modification</code>	 Make sure to have a header comment at the top of every MATLAB source file. <pre><i>% Program Name: myprogram.m % Program Description: % What my program does and how and why. % Name: J. Doe (JDoe@mtu.edu) % Section: L00 % Team: 03</i></pre>
003	<code>% Create a row vector a1 consisting of the numbers in the ordered set { 1, 5, 7, 22, 12} using the square bracket operator.</code>	 For readability and printability, lines of code should be less than 80 characters. This line was over 80 characters, please break it up into two or more lines. For information about how to split up a line of code, see

# # Functional Requirements

- Upload C files successfully
- Program compiles uploaded code
- Provide proper feedback for at least 5 compile errors
- Provide proper feedback for at least 5 style errors
- Upload test suites to run code against
- Program communicates with a database that stores the C antipatterns
- Program runs through the command line and a GUI



# # Non-Functional Requirements

## UI Requirements

- GUI is simplistic and easy to use
- Feedback understandable for novice programmers

## Performance Requirements

- Code analysis process should take no more than a few seconds
- File upload should take no more than a few seconds and have no loss of data

# # Non-Functional Requirements (cont)

## Testing Requirements

- Tested with code from both novice and advanced programmers

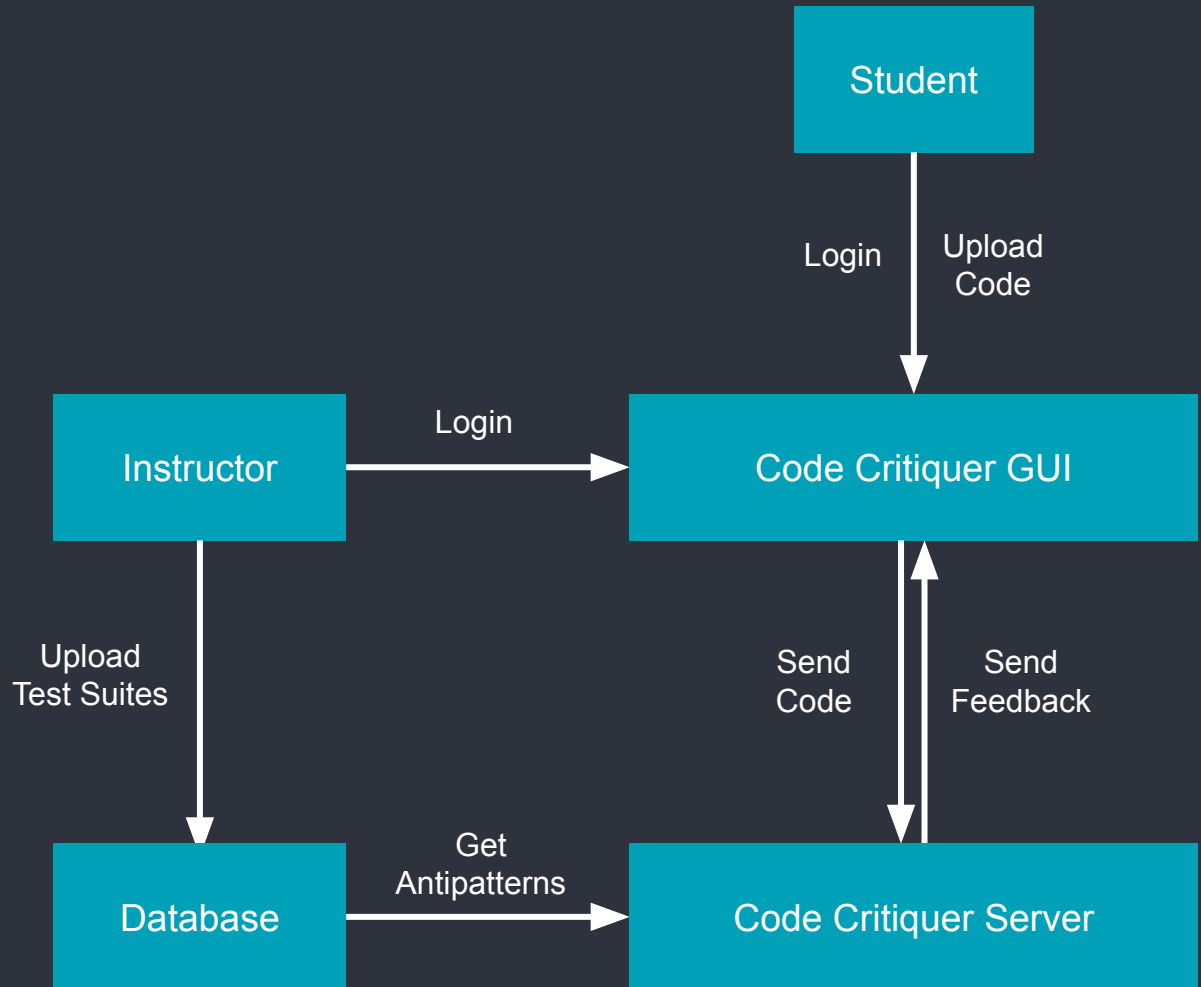
## Maintainability Requirements

- Database easy to update with new antipatterns
- Follow design and standard from MTU
- Well-documented and easy for another team to pick up

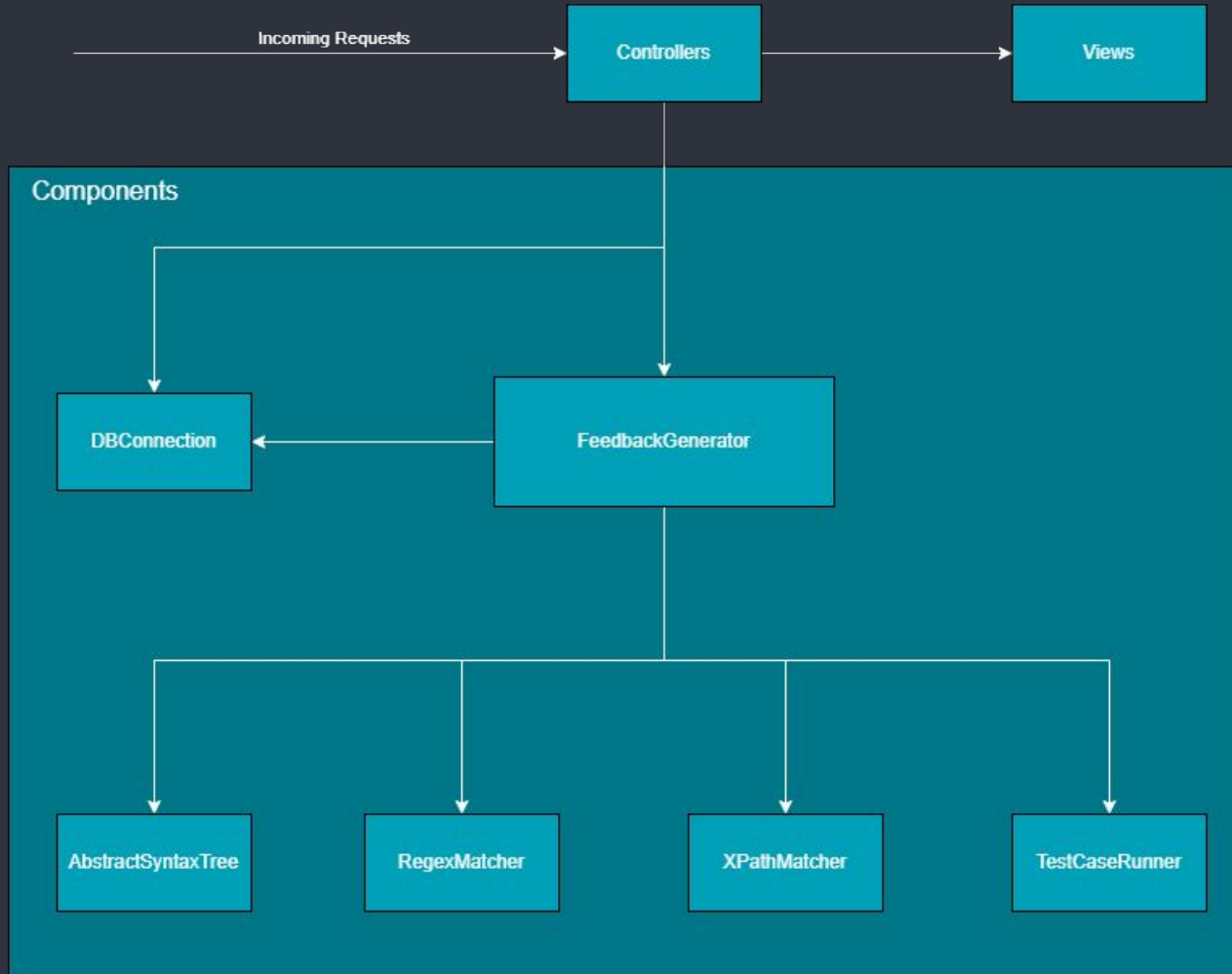
## Legal Requirements

- Complete CITI training
- Get exempt status from IRB to use students' code

# Top  
# Level  
# System  
# Sketch



# System  
# Design -  
# Component  
# Diagram



# # System Design (cont)

- **Hardware** - server to host application on
- **Frameworks** - Flask
- **Standards:**
  - IETF RFC 9110 - HTTP semantics
  - IEEE 12207-1996: Software Lifecycle Processes
  - IEEE 2675-2021: Standard for Devops



**IEEE**

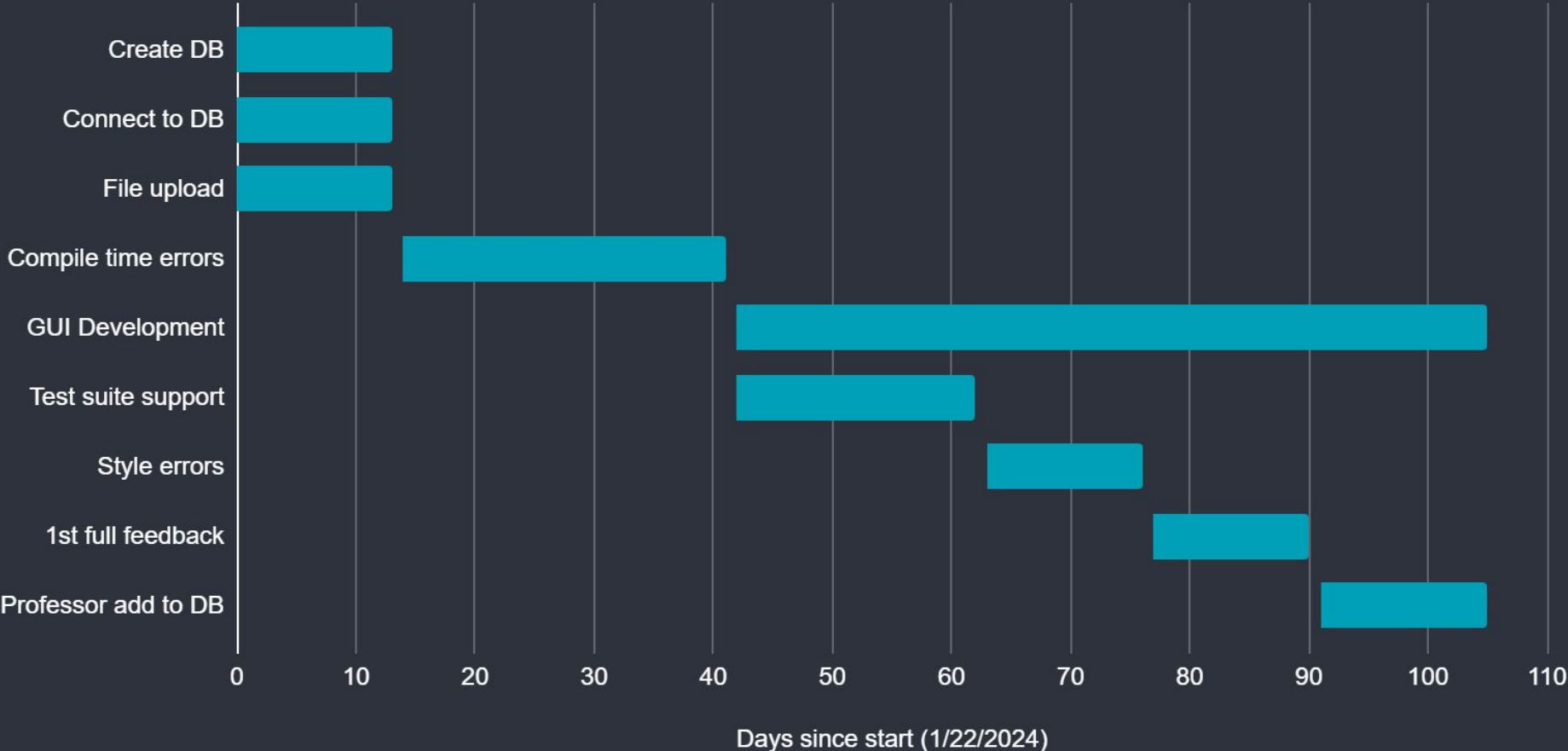
*Advancing Technology  
for Humanity*

# # Design Complexity

- **Code analysis tools**
  - Abstract Syntax Tree generation
  - XPath generation
- **Frontend Development**
  - No longer integrating with Canvas
  - React (Javascript) → Flask (Python)



# Task Timeline



# # Project Plan - Risks & Mitigation

- **File upload**
  - **Concern:** Users could potentially try to upload malicious files.
  - **Solution:** Run uploaded code in its own environment.
- **Error identification**
  - **Concern:** An error we are supposed to detect is not identified
  - **Solution:** Write thorough unit tests to mitigate possible oversights
- **Users adding custom anti-patterns**
  - **Concern:** Users could add application-breaking antipatterns
  - **Solution:** Each user has local storage for new antipatterns



# # Testing

## Unit testing

- Each function in each component should have at least one associated test
- Frameworks for unit testing:
  - Python: unittest
  - Javascript: Jest



# # Testing (cont)

## Integration testing

- Test connection between application and database
  - Check for any errors
  - Test uploading and changing data
- Test connection between UI and application
  - Check data transfer
  - Check for errors and error handling



# # Testing (cont)

## System Testing

- Use predefined code examples with known antipatterns
- Upload code through UI
- Verify critiquer matches expected results

## Acceptance Testing

- Tests should prove
  - Function Requirements
  - Non-Functional Requirements



# # Conclusion

## Progress

- Finished planning of the application design
- Developed descriptions for antipatterns

## Upcoming Plans

- Create a working console application that critiques C code
- Create and connect a database to the console application
- Create a UI for the website

# Instructions for use

If you have a free account, in order to use this template, you must credit [Slidesgo](#) in your final presentation. Please refer to the next slide to read the instructions for premium users.

## **As a Free user, you are allowed to:**

- Modify this template.
- Use it for both personal and commercial projects.

## **You are not allowed to:**

- Sublicense, sell or rent any of Slidesgo Content (or a modified version of Slidesgo Content).
- Distribute Slidesgo Content unless it has been expressly authorized by Slidesgo.
- Include Slidesgo Content in an online or offline database or file.
- Offer Slidesgo templates (or modified versions of Slidesgo templates) for download.
- Acquire the copyright of Slidesgo Content.

For more information about editing slides, please read our FAQs or visit our blog:

<https://slidesgo.com/faqs> and <https://slidesgo.com/slidesgo-school>