1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 0

Code Critiquer

The Critters
02/06/2024

1 0 1 0 0 1 0 1 0 1 0 0 1 0 1 0

Table of contents

Description of users and environment

1001010100101010

Novice Programmers

Users with less experience are more likely to encounter problems caused by common anti-patterns

Language Learners

Users working in unfamiliar languages or environments are more likely to encounter problems caused by common anti-patterns.

Programming Tutors

Users trying to clean or optimize their code can benefit from resolving common anti-patterns or implementing efficient programming patterns.

1010010010110111

1 0 1 0 1 0 1 1 1 0 0 0 1 1 1 0

1 1 0 0 1 0 1 0 1 0 1 0 1 1 1 0

Environment

The environment that the app would be used is a school environment by students and professors as it will be used to help people improve their coding skills and find anti patterns(errors, bugs, and fails) in their code.

Use scenario description

1010101011100011010

Nominal

A student writes a program in Java for one of their classes. The program encounters runtime errors that the student doesn't know how to resolve. Their IDE does not give them any useful information. Without logging in, the student uploads their .java file to the app and submits it for critique. After a moment, the app loads a page showing the student their code, with blocks of text displayed at the lines where anti-patterns were discovered. The student downloads the critique file containing the information shown in the app. When they close the app, their information is discarded.

Error Use

"In this showcase of past projects, delve into The student uploads their project code in a different language then initially expected by the project's description. The code critiquer errors out and informs the user that this is the wrong expected language, and they must submit in the expected language in order to receive any credit for their program.example that underscores my versatility and commitment to excellence. From concept to completion, witness how I navigated complexities and contributed to the success of this [industry/field] initiative"

1100101010101011010

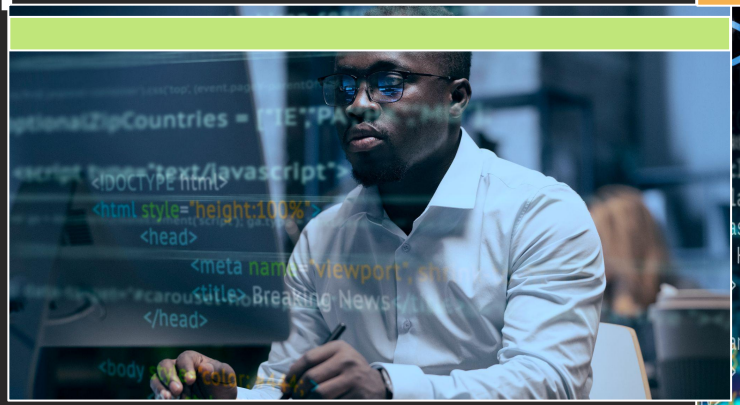
1 0 1 0 1 0 1 1 1 0 0 0 0 1 1 1 0

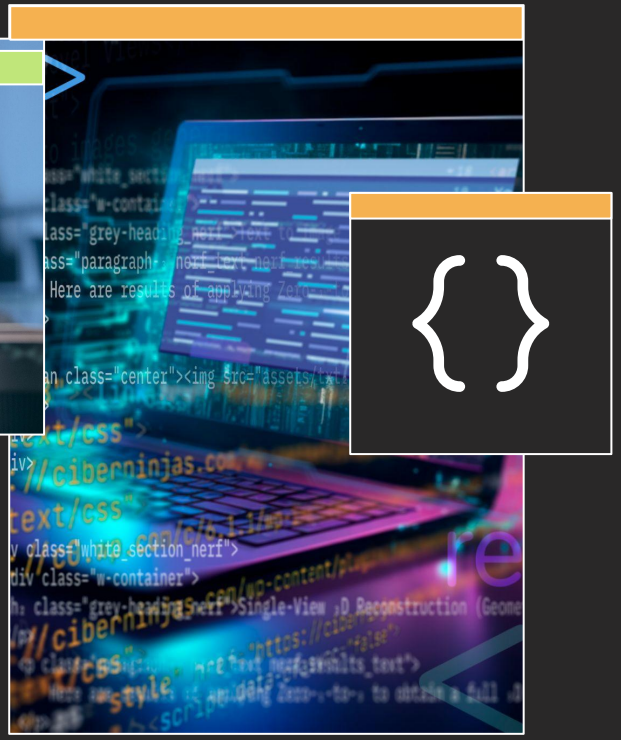
1 1 0 0 1 0 1 0 1 0 1 0 1 0 1 1 1 0

Personas

fictional representations of people that are created from real data about your users

Persona 1





Student is using the app to study
Persona 1: Jane Doe

- Age:18
- 1st year Computer Science Major
- Has not written code before starting degree program
- Link to application was provided by professor
- Uses application frequently throughout the coding process

Persona 2

Independent learner, has a full time job, learning how to code independently as a hobby

Persona 2: Rufus Xavier

- Age:25
- Electrician
- Learning programming independently using online resources
- Writes code specifically to learn programming, attempting to learn through experimentation

Persona 3

Professor is using the code the intended way to help students improve their code

Persona 3: Dr. Steven Hawk

- Age: 66
- Jane Doe's intro to programming professor
- Provided link to application to all students
- Will not be using the application himself

1 0 1 0 1 0 1 1 1 0 0 0 1 1 1 0

Persona 4

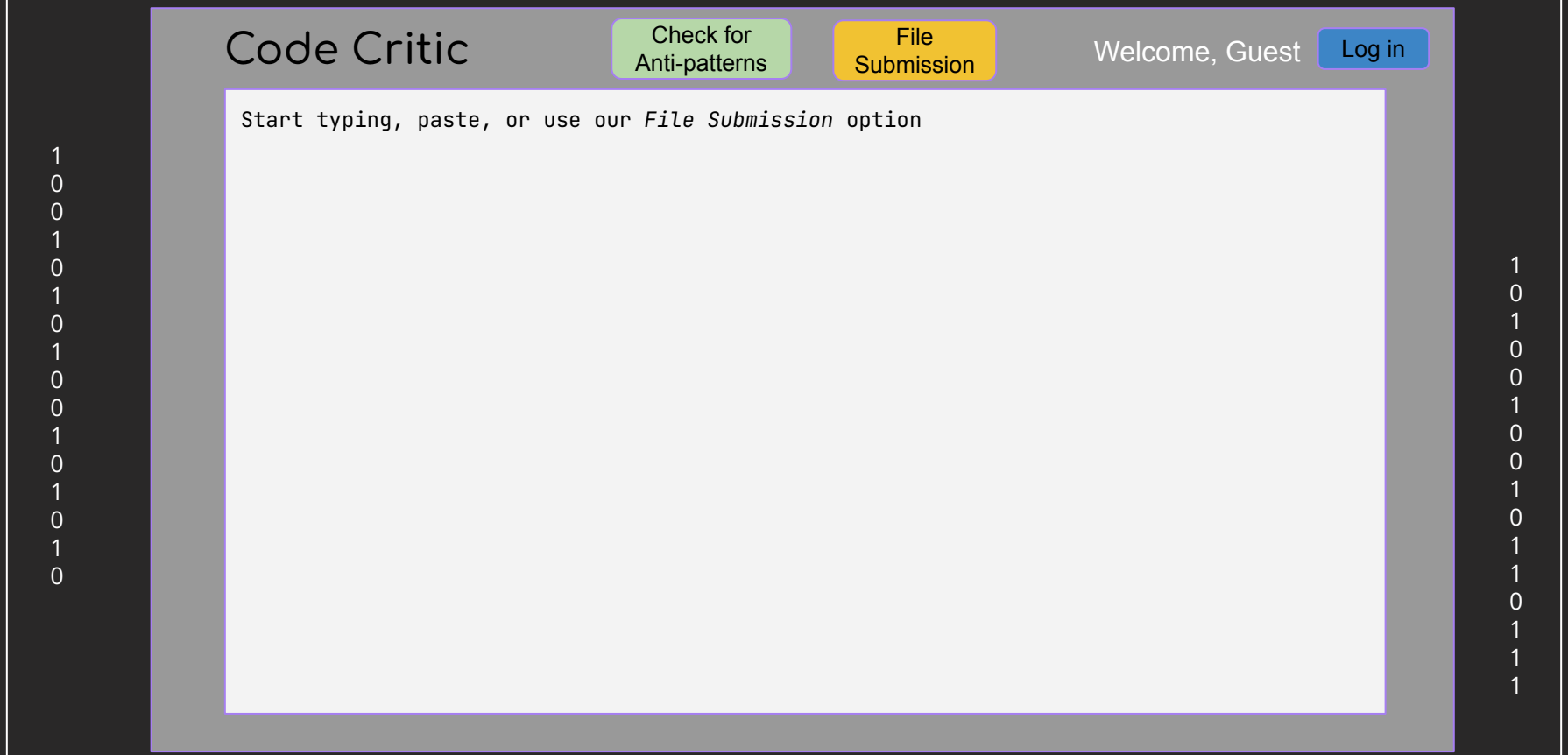
Professor is not using the app the intended way and is using it to

Persona 4: Dr. Thomas Atkins

- Age: 45
- Data structures professor
- No programming students using the program
- Using the application as an automatic grader for assignments

1 1 0 0 1 0 1 0 1 0 1 0 1 1 1 0

UI: Fresh Load Page



UI: Guest Text-Insert Submission

Code Critic

Check for Anti-patterns

File Submission

Welcome, Guest

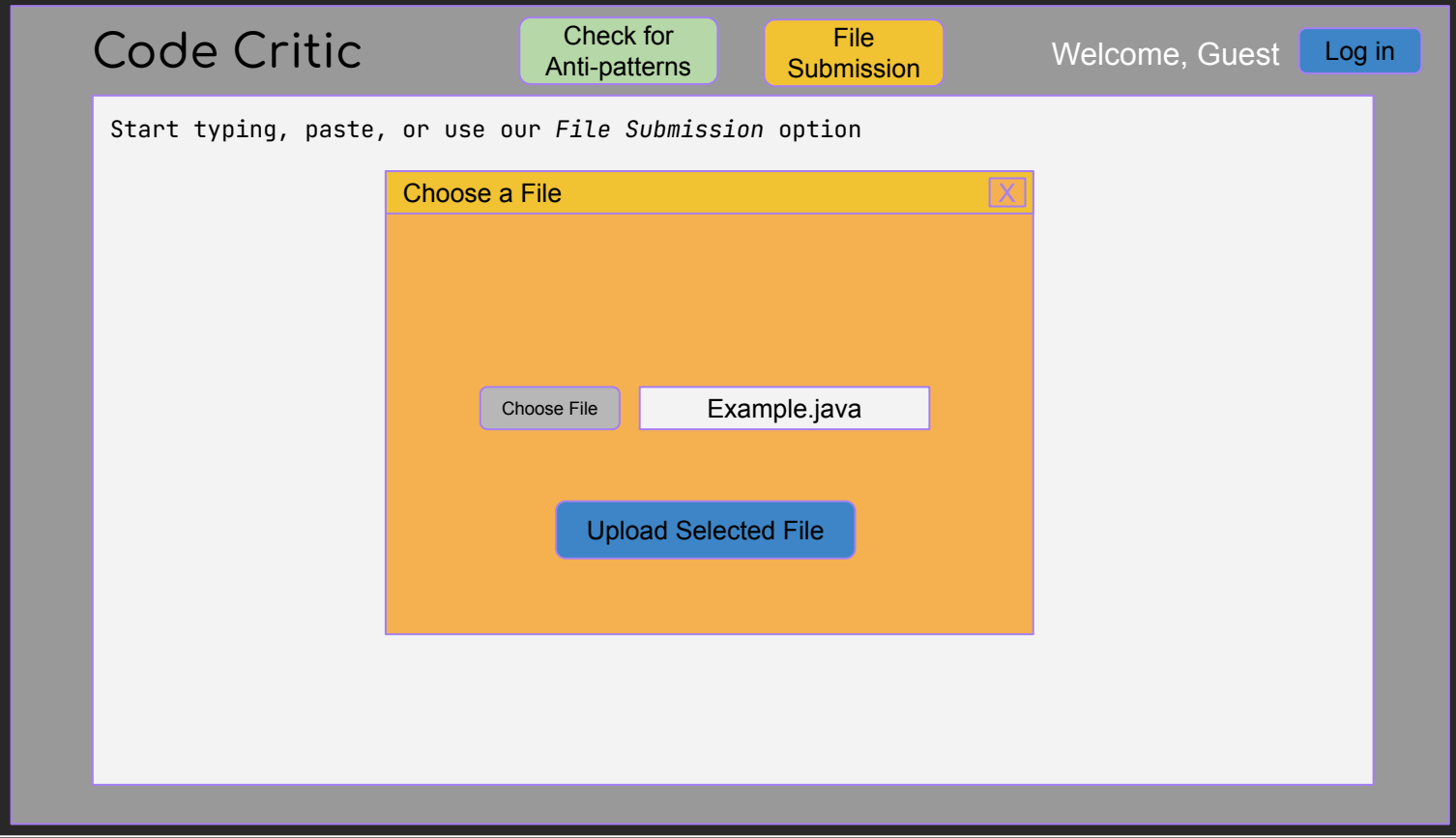
Log in

```
01 public class Example {
02
03   private int x;
04   private boolean isY;
05   private int C = 45;
06
07   public Example() {
08     this.x = 55;
09     this.isY = x < C;
10   }
11 }
```

1 0 0 1 0 1 0 1 0 0 1 0 1 0 1 0

1 0 1 0 0 1 0 0 1 0 1 1 0 1 1 1

UI: Guest File-Insert Submission



1 0 0 1 0 1 0 1 0 0 1 0 1 0 1 0

1 0 1 0 0 1 0 0 1 0 1 1 0 1 1 1

UI : Login View

User Log in

Username:

Password:

Register

Log in

1 0 0 1 0 1 0 1 0 0 1 0 1 0 1 0

1 0 1 0 0 1 0 0 1 0 1 1 0 1 1 1

UI: Guest Submission Result

UI : Side by Side Code Comparison (logged in only)

"Our application should be easy to learn and use, efficiently critique the code provided to it, and the critique views are effectively presented to the user."

Usability Goals

Usability Concerns

Understanding parsing

The premade parser that will be provided to us may have a learning curve for the team

User Progress Tracking

Another challenge is determining how our team will track, store and display progress of the user.

Database Storage

Precisely determining how and where this data is stored will be a challenge

Unique Link Authentication

Avoiding full user authentication should be easier but regardless, unique link authentications come with challenges that need to be considered

Critique Display

Designing UI elements that convey only critical information to the user is crucial to our project's usability

Thanks!

Any Question?

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**