

# Full-Stack Developer Technical Assessment

---

## Assessment Overview

**Tasks: 2 main tasks**

**Technologies: Flask, Next.js, TypeScript, React**

---

## Getting Started

### Step 1: Get the Code

Clone the assessment repository:

```
git clone https://github.com/GhiProject/New-Student-Test.git
cd New-Student-Test/flask-app
```

### Step 2: Set Up the Flask App

# Install dependencies

```
pip install -r requirements.txt
```

# Run the application

```
python app.py
```

Visit <http://localhost:5000> in your browser.

Login Credentials: - Email: test@example.com - Password: password123

---

## Your Tasks

### Task 1: Find and Fix 3 Bugs (30 minutes)

The Flask application has 3 intentional bugs that need to be fixed.

What to do: 1. Explore the application and find all 3 bugs

2. Fix each bug in the code

3. Document your findings

Create a file: BUG\_REPORT.md

For each bug, document:

## Bug #1: [Brief Title]

**\*\*What's broken:\*\*** [Describe the issue]

**\*\*Where:\*\*** [File name and line number]

**\*\*Why it matters:\*\*** [Security/Performance/Functionality impact]

**\*\*How I fixed it:\*\*** [Explain your solution]

**\*\*How to test:\*\*** [Steps to verify the fix]

Hints about the bugs: - One is related to authentication/security - One is related to performance/database queries - One is related to file validation

---

## Task 2: Migrate Dashboard to Next.js (25 minutes)

Convert the Flask dashboard page to a modern Next.js application with TypeScript.

Requirements: - Create a Next.js dashboard page that displays files - Use TypeScript for proper type safety - Create an API endpoint to fetch files - Show a loading state while fetching data - Basic styling (TailwindCSS or plain CSS)

What you need to create:

```
next-app/
├── app/
│   ├── dashboard/
│   │   └── page.tsx    # Dashboard page component
│   └── api/
│       └── files/
│           └── route.ts # API endpoint
├── lib/
│   └── types.ts        # TypeScript interfaces
└── README.md          # Setup instructions
```

Don't worry about: - Authentication implementation - Real database connection (mock data is fine) - Perfect design - Full CRUD operations

---

## Submission Instructions

### Step 1: Create Your GitHub Repository

1. Create a new public repository on your GitHub account
2. Name it something like: [YourName]-flask-assessment
3. Initialize it with a README

### Step 2: Organize Your Code

Your repository should have this structure:

```
your-repo/
├── README.md          # Setup instructions
├── BUG_REPORT.md      # Your bug documentation
├── flask-app/         # Fixed Flask application
│   ├── app.py         # Mark your changes with comments
│   └── ...
├── next-app/          # Your Next.js application
│   ├── app/
│   ├── lib/
│   └── README.md
```

### Step 3: Push Your Code

# In your working directory

```
git init
```

```
git add .
```

```
git commit -m "Complete technical assessment"
```

```
git remote add origin https://github.com/YOUR_USERNAME/YOUR_REPO_NAME.git
```

```
git branch -M main
```

```
git push -u origin main
```

### Step 4: Submit

[CS Team Submission – Fill out form](#)

Important: Make sure your repository is PUBLIC so we can review it!

---

### Evaluation Criteria

We'll evaluate your submission based on:

#### Bug Finding & Fixing (60 points)

- Did you find all 3 bugs?
- Do your fixes actually work?
- Are your explanations clear and detailed?

#### Next.js Migration (40 points)

- Does the dashboard page render correctly?
- Are TypeScript types properly used?
- Is the code clean and well-organized?

#### Code Quality (Bonus)

- Clean, readable code
  - Good commit messages
  - Clear documentation
- 

### Tips for Success

1. Time Management: Spend ~30 min on bugs, ~25 min on migration
  2. Commit Often: Make small commits as you progress
  3. Document Well: Clear explanations are valued
  4. Test Everything: Make sure your code runs before submitting
  5. Ask if Stuck: If you're blocked on setup for >30 minutes, email us
- 

### Troubleshooting

Port 5000 already in use?

# Mac/Linux

```
lsof -ti:5000 | xargs kill -9
```

```
# Windows
netstat -ano | findstr :5000
taskkill /PID <PID> /F
Flask not installed?
pip install Flask
# or
pip3 install Flask
Python not found? Make sure Python 3.9+ is installed:
https://www.python.org/downloads/
```

---

### **Questions?**

If you encounter any blocking issues: - Email: [nsavale@iu.edu](mailto:nsavale@iu.edu)  
We want to test your coding skills, not your debugging of environment issues. Don't hesitate to reach out if you're stuck on setup!

---

### **Deadline**

Please complete and submit within 7 days from receiving this email. (02/23/2026, 6:00 pm est)

---

Good luck! We're excited to see your work.

Regards,  
CS Team  
Global-Health-Impact