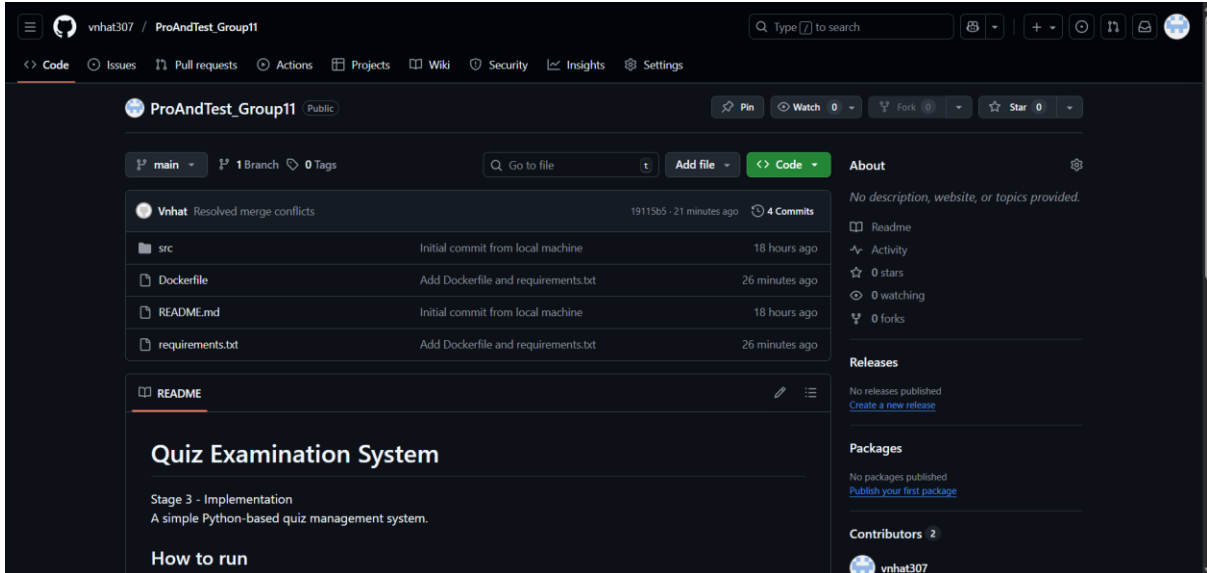


## Part 1: Evidence of tool usage

### 1. Github repository

Link: [https://github.com/vnhat307/ProAndTest\\_Group11.git](https://github.com/vnhat307/ProAndTest_Group11.git)

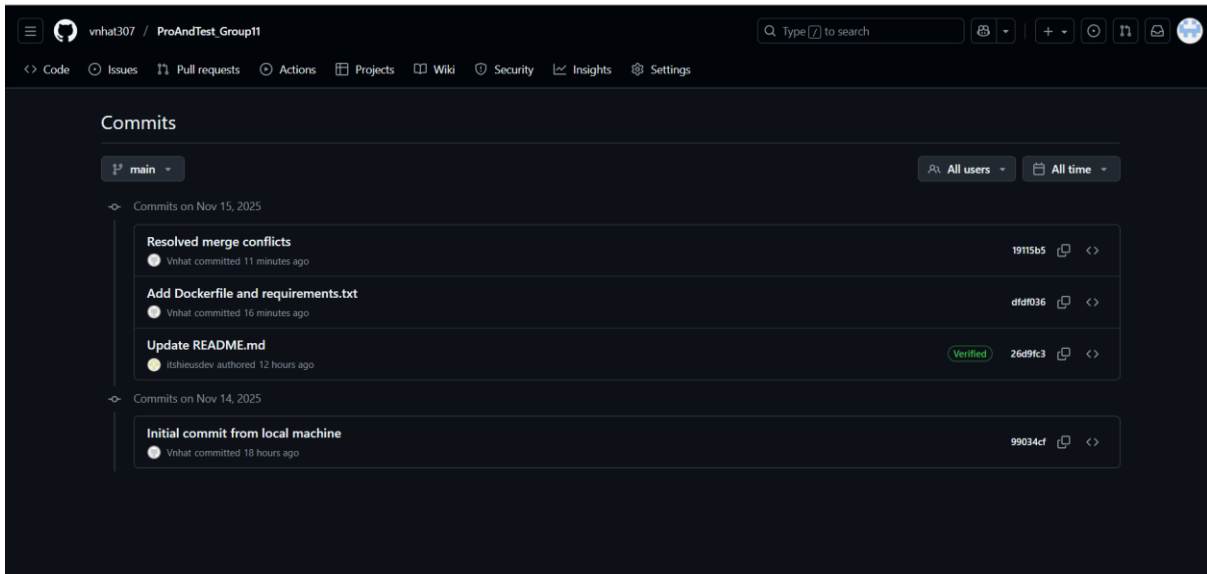


The screenshot shows the GitHub repository page for **ProAndTest\_Group11** by user **vnhat307**. The repository is public and has 0 stars, 0 forks, and 0 watchers. The main branch is **main**. The repository contains 4 commits. The commit history is as follows:

Commit Message	Commit Hash	Time Ago
Resolved merge conflicts	19115b5	21 minutes ago
src		18 hours ago
Dockerfile		26 minutes ago
README.md		18 hours ago
requirements.txt		26 minutes ago

The README file is titled **Quiz Examination System** and describes a simple Python-based quiz management system. It includes a section for **How to run**.

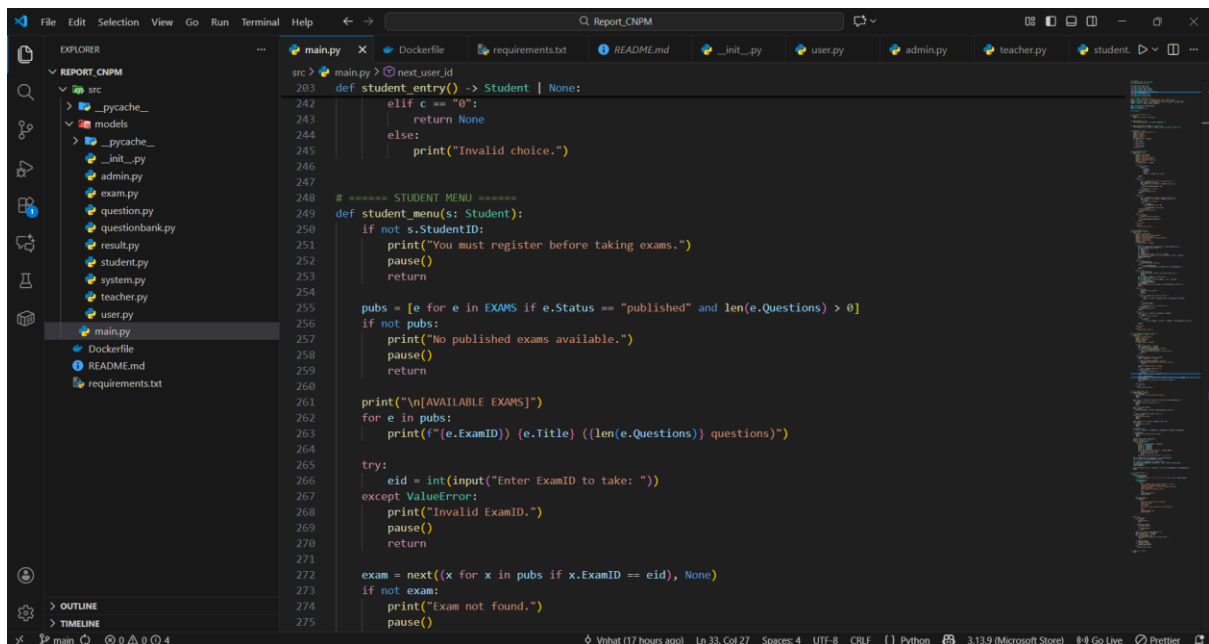
### Commit History



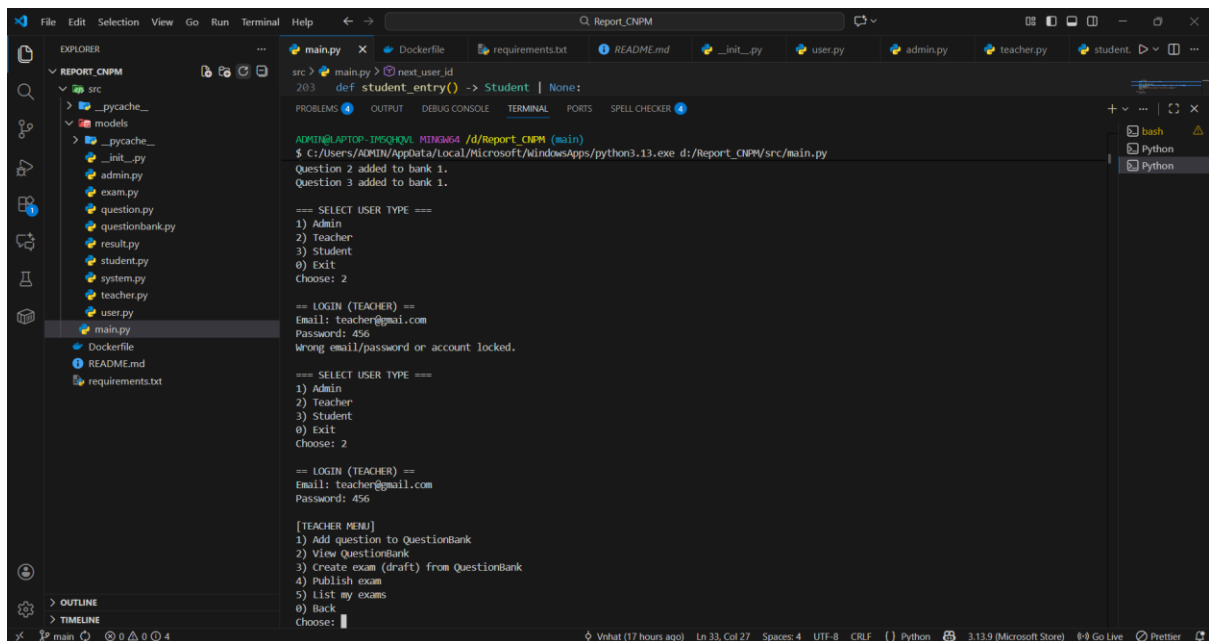
The screenshot shows the **Commits** page for the **ProAndTest\_Group11** repository. The page displays a list of commits, filtered by the **main** branch. The commit history is as follows:

Commit Message	Commit Hash	Time Ago
Resolved merge conflicts	19115b5	11 minutes ago
Add Dockerfile and requirements.txt	d4d036	16 minutes ago
Update README.md	26d9fc3	12 hours ago
Initial commit from local machine	99034cf	18 hours ago

## 2. VsCode



```
src > main.py > next_user_id
203 def student_entry() -> Student | None:
242     elif c == "0":
243         return None
244     else:
245         print("Invalid choice.")
246
247
248 # ===== STUDENT MENU =====
249 def student_menu(s: Student):
250     if not s.StudentID:
251         print("You must register before taking exams.")
252         pause()
253         return
254
255     pubs = [e for e in EXAMS if e.Status == "published" and len(e.Questions) > 0]
256     if not pubs:
257         print("No published exams available.")
258         pause()
259         return
260
261     print("\n[AVAILABLE EXAMS]")
262     for e in pubs:
263         print(f"({e.ExamID}) ({e.Title}) ({len(e.Questions)} questions)")
264
265     try:
266         eid = int(input("Enter ExamID to take: "))
267     except ValueError:
268         print("Invalid ExamID.")
269         pause()
270         return
271
272     exam = next((x for x in pubs if x.ExamID == eid), None)
273     if not exam:
274         print("Exam not found.")
275         pause()
```



```
ADMIN@LAPTOP-INSQHQV MINGW64 /d/Report_CNPW (main)
$ C:/Users/ADMIN/AppData/Local/Microsoft/WindowsApps/python3.13.exe d:/Report_CNPW/src/main.py
Question 2 added to bank 1.
Question 3 added to bank 1.

=== SELECT USER TYPE ===
1) Admin
2) Teacher
3) Student
0) Exit
Choose: 2

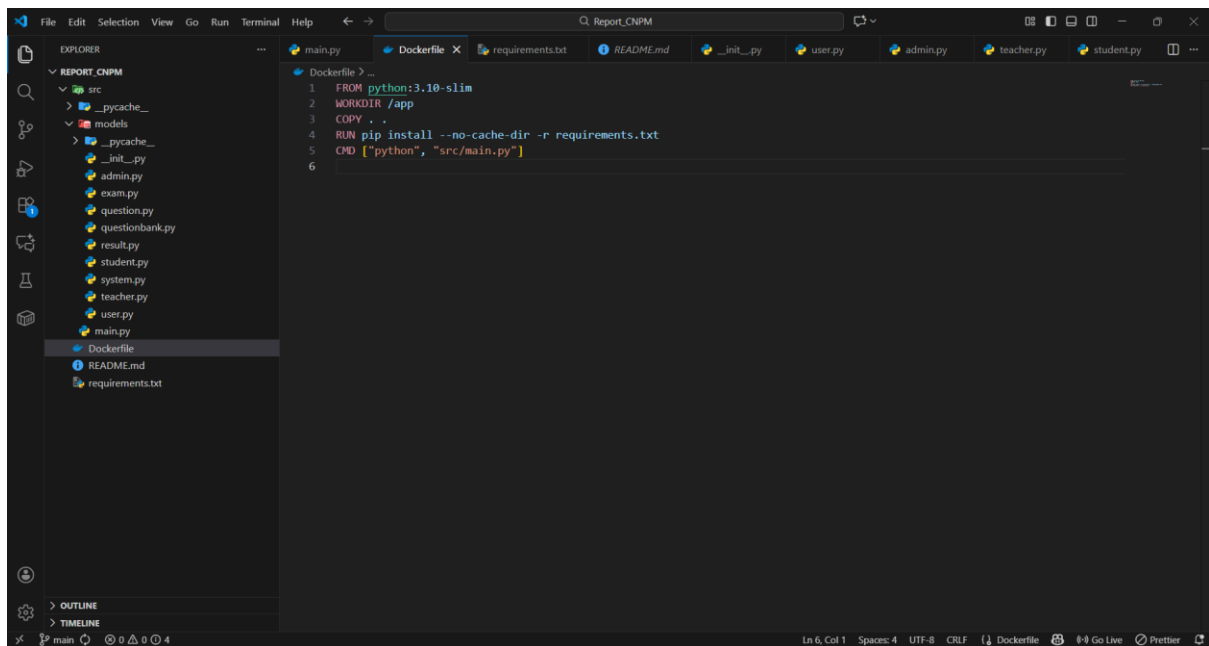
=== LOGIN (TEACHER) ===
Email: teacher@gmail.com
Password: 456
Wrong email/password or account locked.

=== SELECT USER TYPE ===
1) Admin
2) Teacher
3) Student
0) Exit
Choose: 2

=== LOGIN (TEACHER) ===
Email: teacher@gmail.com
Password: 456

[TEACHER MENU]
1) Add question to QuestionBank
2) View QuestionBank
3) Create exam (draft) from QuestionBank
4) Publish exam
5) List my exams
0) Back
Choose: 1
```

### 3. Dockerfile

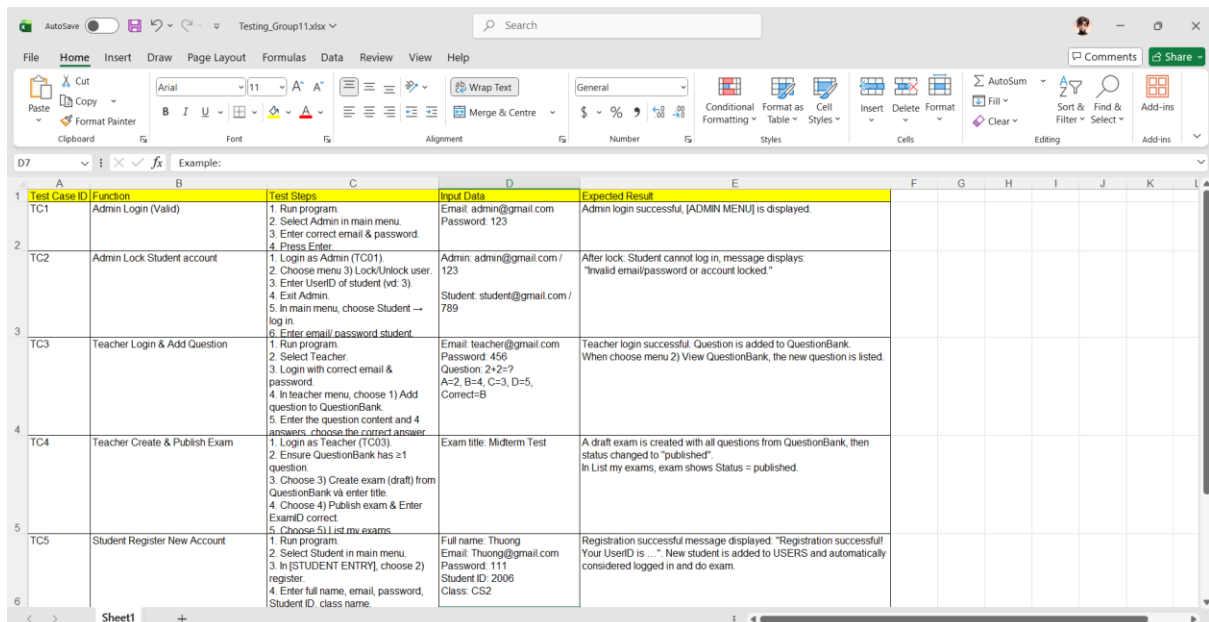


The screenshot shows the Visual Studio Code editor with a Dockerfile open. The Dockerfile contains the following instructions:

```
1 FROM python:3.10-slim
2 WORKDIR /app
3 COPY . .
4 RUN pip install --no-cache-dir -r requirements.txt
5 CMD ["python", "src/main.py"]
```

The Explorer sidebar on the left shows the project structure for 'REPORT\_CNPM', including files like 'src', 'models', 'init.py', 'admin.py', 'exam.py', 'question.py', 'questionbank.py', 'result.py', 'student.py', 'system.py', 'teacher.py', 'user.py', 'main.py', 'Dockerfile', 'README.md', and 'requirements.txt'.

### 4. Testing



Test Case ID	Function	Test Steps	Input Data	Expected Result
TC1	Admin Login (Valid)	1. Run program. 2. Select Admin in main menu. 3. Enter correct email & password. 4. Press Enter.	Email: admin@gmail.com Password: 123	Admin login successful. [ADMIN MENU] is displayed.
TC2	Admin Lock Student account	1. Login as Admin (TC01). 2. Choose menu 3) Lock/Unlock user. 3. Enter UserID of student (vid: 3). 4. Exit Admin. 5. In main menu, choose Student → log in. 6. Enter email: password student.	Admin: admin@gmail.com / 123 Student: student@gmail.com / 789	After lock: Student cannot log in, message displays: "Invalid email/password or account locked."
TC3	Teacher Login & Add Question	1. Run program. 2. Select Teacher. 3. Login with correct email & password. 4. In teacher menu, choose 1) Add question to QuestionBank. 5. Enter the question content and 4 answers, choose the correct answer.	Email: teacher@gmail.com Password: 456 Question: 2+2=? A-2, B-4, C-3, D-5, Correct=B	Teacher login successful. Question is added to QuestionBank. When choose menu 2) View QuestionBank, the new question is listed.
TC4	Teacher Create & Publish Exam	1. Run program. 2. Ensure QuestionBank has ≥1 question. 3. Choose 3) Create exam (draft) from QuestionBank via enter title. 4. Choose 4) Publish exam & Enter ExamID correct. 5. Choose 5) List my exams.	Exam title: Midterm Test	A draft exam is created with all questions from QuestionBank, then status changed to "published". In List my exams, exam shows Status = published.
TC5	Student Register New Account	1. Run program. 2. Select Student in main menu. 3. In [STUDENT ENTRY], choose 2) register. 4. Enter full name, email, password, Student ID, class name.	Full name: Thuong Email: Thuong@gmail.com Password: 111 Student ID: 2006 Class: CS2	Registration successful message displayed: "Registration successful! Your UserID is ...". New student is added to USERS and automatically considered logged in and do exam.

## Part 2: Assignment

Name	ID	Task	Rate
Nguyễn Đình Việt Nhật (Leader)	049206008327	<ul style="list-style-type: none"><li>- Implement main.py file for Stage 3 (Handle user flow for admin, teacher, student)</li><li>- Complete all tasks in Stage 4 (Testing document, Dockerfile)</li></ul>	100%
Lê Thị Thương	045306007998	<ul style="list-style-type: none"><li>- Implement user, teacher, and admin functionalities</li><li>- Create and manage the admin, teacher, and student classes</li></ul>	100%
Võ Thanh Hiếu	051206003008	<ul style="list-style-type: none"><li>- Implement the system functionalities</li><li>- Handle the system logic (auto grading, statistics, etc.)</li></ul>	50%