A joint
Digital Transformation
and
Software Engineering
PROJECT

INTRODUCTION

EFREI Paris wants to develop a solution that will help improve the follow-up of internships.
It must include the following:
1) An electronic document management system to archive all the reports about internships.
2) An internship follow-up system for internship tutors.

FEATURES OF THE ELECTRONIC DOCUMENT MANAGEMENT SYSTEM

Existing system
The students send their reports to the tutors by e-mail (academic and enterprise). All the students must also submit intermediate documents (see the Excel screenshot on the last page).

New system
1) Only the final report and the "Cahier des charges" will be saved. Please propose solutions to digitalize the other documents.
2) Search engine:
- It must allow an easy search of documents and make available the access to students.
- The search could be by keyword(s).

3) It allows the students to submit their reports before a deadline.
4) The report becomes readable for the students and teachers only after validation of the academic and enterprise tutors.
5) The access must be restricted to three (3) profiles: an admin, a tutor, and a student profile.
Please come up with propositions and ideas about what each of these profiles can or cannot do.
6) All reports will have a level of confidentiality :
Normal level : The document can be downloaded, copied (copy-paste), and printed.
or
Sensitive level: Read-only online but prohibited to print, copy, and download.

FEATURES OF THE FOLLOW-UP SYSTEM

1) Management of user input on the login screen

2) Each tutor only has access to the interns assigned to them.

3) List of all the interns of the current year

4) Add or remove an internship.

5) Possibility to modify fields; which fields will depend on the profile.

Propose a solution to manage as many columns as possible from the Excel model on the next page.

UNLEASH YOUR CREATIVITY!

You may suggest ideas for the layout and design of the pages, but :

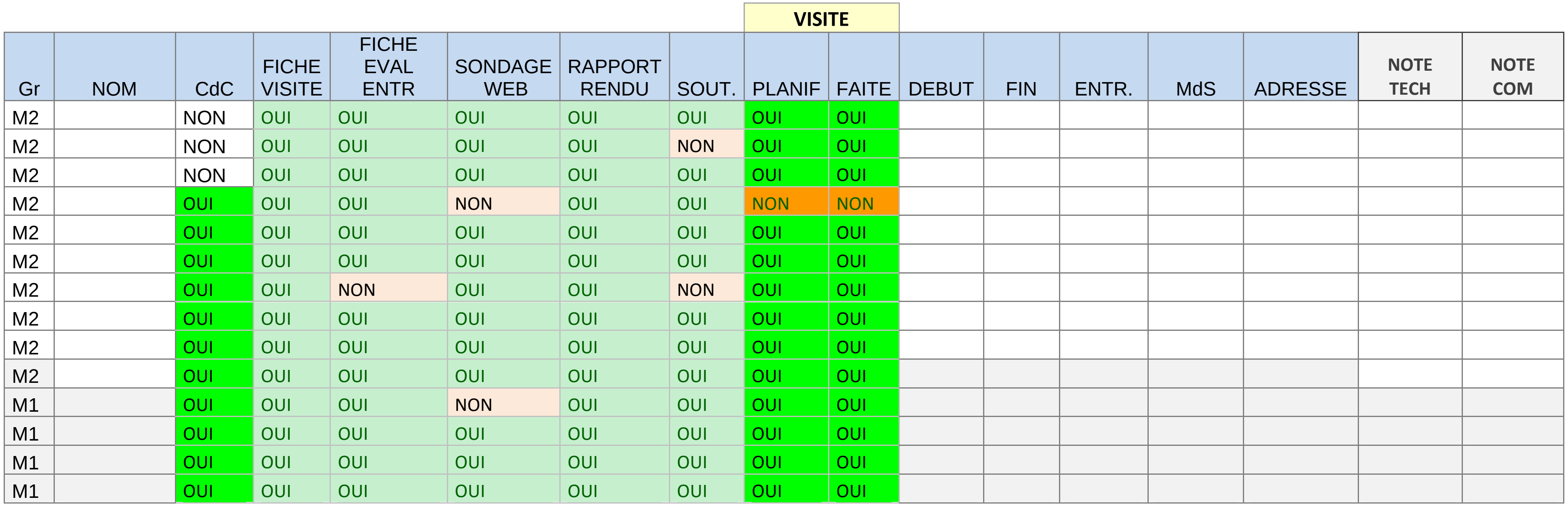
a) Your idea must **comply with the requirements**

b) It must be :
   - **a. Practical**
   - **b. Simple**
   - **c. User-friendly**

c) The navigation must be **intuitive**

MAIN SCREEN

Below is the existing Excel file for doing the same task.
Please design a Web interface that will at least help manage the same level of information.
Keep in mind that more actions can be possible in your solution.



								VISITE								
Gr	NOM	CdC	FICHE VISITE	FICHE EVAL ENTR	SONDAGE WEB	RAPPORT RENDU	SOUT.	PLANIF	FAITE	DEBUT	FIN	ENTR.	MdS	ADRESSE	NOTE TECH	NOTE COM
M2		NON	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M2		NON	OUI	OUI	OUI	OUI	NON	OUI	OUI							
M2		NON	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M2		OUI	OUI	OUI	NON	OUI	OUI	NON	NON							
M2		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M2		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M2		OUI	OUI	NON	OUI	OUI	NON	OUI	OUI							
M2		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M2		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M2		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M1		OUI	OUI	OUI	NON	OUI	OUI	OUI	OUI							
M1		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M1		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							
M1		OUI	OUI	OUI	OUI	OUI	OUI	OUI	OUI							

EXPECTATIONS

1. Clean code compliant:
    - a. S.O.L.I.D. Principles
    - b. Naming
    - c. Error handling
    - d. Object and Data Structures
2. As easily maintainable as possible (good documentation is a part of that)
3. Scalable
4. Secure
5. Acceptable level of performance
6. UX Design
7. Architecture(s)
8. Pattern(s)
9. Paradigm(s) / Language(s)
10. Technologies
11. Frameworks
12. Delivery methodology
13. Easy setup/deployment/configuration
14. **Risky and daring choices (including AI or ML in your solution, implementation of design/architectural patterns, elegant architecture, TDD or BDD, etc.)**

This order is not based on the importance of the above criteria.

DELIVERABLES

1. A **REPORT** containing -at least- the following:
    - a. A link to your remote repositories (GitHub or GitLab or...).
    - b. The URL of your deployed solution (eventually).
    - c. A clear diagram representing the architecture of your solution.
    - d. A dedicated section explaining how you think your team met the expectations (see "Expectations" section above).
    - e. Sets of credentials to test your application.
    - f. A readme section with :
        - i. Detailed instructions on how to setup/initialize/install/launch your application.
        - ii. What features did you not have time for or would like to implement in a potential future release of your solution?
    - g. A link to your video (see below)
2. The SQL script to initialize your database (can be included within your project).
3. A 10 mins max video clip :
    - a. Create this clip using the tool of your choice.
    - b. In it, you will make a convincing presentation of your solution :
    - c. A quick technical presentation: modules, packages, files, overall structure, etc.…
    - d. A tour of the features via a demo
    - e. In the video, you must answer the following questions :
        - i. What is unique, original, and creative about your solution?
        - ii. What part was the most difficult to implement? And why?

iii. What do you think of this type of project that starts with deliverables done in another course by a completely different group of students?
iv. How did you work together, and how did everyone contribute?
v. What are the three most important things you got from this project?

YOUR ZIP FILE MUST BE UPLOADED ON MOODLE.

DEADLINE

05/02/2024 @ 23:59