IN104 : Projet informatique

Build a navigation app for your smartphone

Stephen CREFF - stephen.creff@gmail.com

2021

Evaluation (remainder)

1. 25% **Source code**: features, tests, code documentation, etc. Source code must be documented, the report including a link to the repository (GitHub) must be in a .zip file. The repository will contain the sources, as well as a plain text file README.md which indicates the actual operational features and limitations. Python code should compile, the teaching assistants are not supposed to make significant corrections for it to compile: a code with few features, but that compiles and does not crash will be preferred to a more complete code but which is not directly operational!
2. 25% **Defense** (10 minutes presentation, 5 min. questions): The formal quality of the presentation will be an important element. The defense needs to include a demonstration on the basis of the source code, an analysis of the difficulties encountered and implemented solutions. It will not include a presentation of the problem or the method of resolution that the teaching assistant obviously knows already well. The defense is open to everyone (subject to the acceptance of the pair that will present). The chronological order of defense will be given by the list of each group.
3. 20% **Practice Analysis**: You will return a critical (report max. 5 pages) report before the defense in which you analyze and criticize the progress of your project and its success and failure factors. This evaluation component also includes the oral treatment of this question during the defense.
4. 30% **Continuous Progress Evaluation** of the practical work during the practical lab sessions (based on Git commits).

Deliverables and due dates

- Report: 1 single PDF to send by mail, containing link to repository:
  - Quality over Quantity
  - Substance over form
  - [24/05/2021, 20:00]
- Code:
  - On GitHub and by zip
  - [24/05/2021, 20:00]

- Defence day
  - [25/05/2021]

Report’s content, briefly (in French or English)

- Product description (some highlights)
  - Main requirements, services, … (what you choose to implement)
- Technical environment:
  - Software language, libraries, … (how, with what tools you worked)
- Software development plan followed & Project status (how you organized your work)
  - Initial plan: Activities, roles (task allocation in the group),
  - Directions followed,
  - Key results, and issues faced
    - Identified risk, issues, scope changes
- Lessons learned