Deliverables

Blockchain and Distributed Ledger Technologies / # DE

SAPIENZA
UNIVERSITÀ DI ROMA

Claudio Di Ciccio

claudio.diciccio@uniroma1.it
http://www.di.uniroma1.it/~diciccio

Deliverables

- Presentation (not mandatory) online
  - On an agreed day at the end of the winter semester
  - Timing: 4/5 presentation + 1/5 discussion
  - Slides must be uploaded in the shared folder before the lecture begins
    - Format: PowerPoint or PDF (no time for laptop switch, really)
- Project consignment
  - By 16:59 Rome/CET time two days before the exam
    - Right: before tea time. No deadline extension.
  - Via upload on Google Drive (see the Classroom space)
  - Format: compressed file containing
    - PDF of the report
    - Folder with project files (according to the structure seen in class)
  - Length: Not more than 16 pages
    - EXCLUDING table of contents and other indices, references, front page
    - INCLUDING figures
- Short project demo (mandatory) online (min 5, max 8 minutes)
  - Soon after the written test of the exam, on planned dates
    - No slides needed.
    - Plan and enact scenarios showcasing the main functionalities.

Presentation

- Students *attending the presentation sessions* compose the assembly
- Every student is provided with $\lceil N/2 \rceil$ upvotes, where N is the number of presenting teams
- All the upvotes of an assembly member are discarded if they express
  - more than $\lceil N/2 \rceil$ preferences, or
  - less than $\lceil N/4 \rceil$ preferences.
    - Not voting is an option
- Two voting chambers:
  1. To assess the technical quality
     - Mastery of the topic, appropriateness of the solution, attention to details
  2. To assess the presentation quality
     - Clarity of exposition, neatness of the message, completeness of the story
- Teams receive extra points based on their quartile in the rankings
  - 1st quartile: +1.5 pts.; 2nd quartile: +1 pt.; 3rd quartile: +0.5 pts.
  - One ranking per voting chamber (the maximum bonus is +3 pts.)
- The instructor can add or subtract up to 0.5 extra points
  - On top, and independently of, the assembly bonus

Subject to tuning

Structure of the presentation

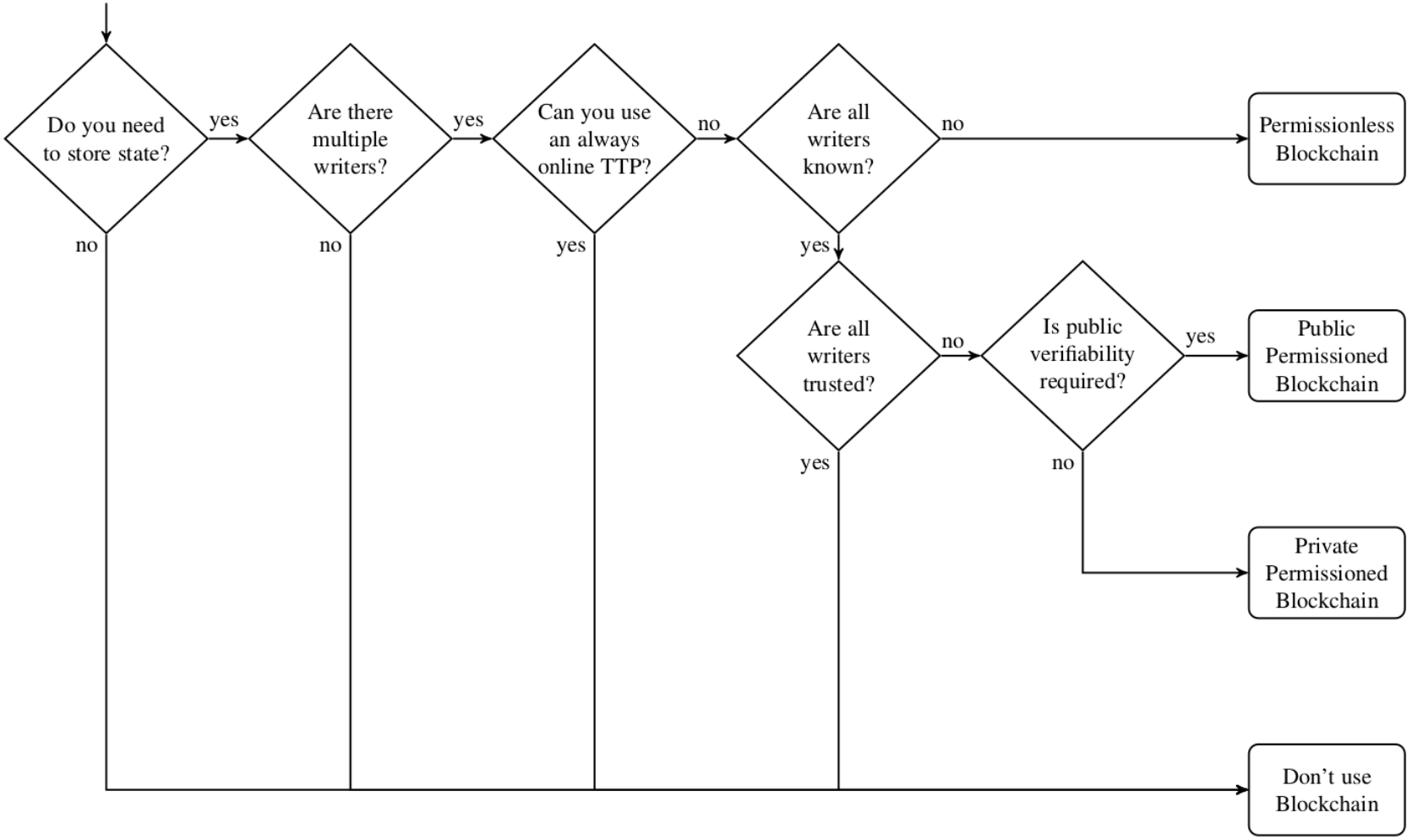
Percentages are demonstrative

Use your ingenuity and unique style

- Introduction (~10%)
  - Title
  - Team members and prospective main responsibilities
    - This does NOT mean every member does one thing – collaboration is paramount
    - Presentation / HTML front-end / Solidity development / Javascript development / Architectural design / Token engineering / …
- Presentation of the context (~20%)
  - Aim of the DApp
  - Why using the blockchain
    - Is it fit for purpose? Why? What are the advantages?
  - What type of blockchain you would use in production and why (private? public? permissioned? permissionless? See following slides for an example)
- Approach (~20%)
  - Expected contracts, tokens, communications in-between, etc.
  - Remarks, e.g., app tokens represent this, ether transactions represent that…
- Early prototype showdown (~20%)
  - Contracts, GUI, logos, etc., or sketchy drawings thereof
  - Software architecture diagrams, or sketchy drawings thereof
- Known issues and limitations (~10%)
- Recap (~10%)
- Slack (~10%)

Looking for inspiration? Watch this brilliant video:
https://youtu.be/Iwpi1Lm6dFo
Also, you may want to check out
my personal guidelines on presentations

Blockchain please?



Diagrams

- A UML **concept** diagram (or an analogous model) to describe the structure and content of the **smart contract**(s)
- A UML **component** model (or an analogous diagram) to indicate the back-end and front-end **modules** and their **dependencies**
- UML **collaboration** diagrams (or analogous ones) to illustrate the interactions among those components
- A UML **use case** diagram to illustrate the main usage scenarios of the DApp or tokens
- UML **activity diagrams** (or analogous ones) to illustrate the lifecycle of the tokens
- Happy diagramming!

Structure of the report (use the Sapienza template)

- Frontpage (~1 page, not counted in the total)
- Preface (~1 page)
  - Title and 3-sentence presentation of the DApp
  - Team members and main responsibilities
  - Outline of the report (abstract)
- Background (~2 pages)
  - Blockchain: history, rationale, concepts…
  - Application domain
- Presentation of the context (~2 pages)
  - Aim of the DApp
  - Why using a blockchain, and what type thereof to use in production
- Software architecture (~5 pages)
  - Diagrams, diagrams, diagrams, and more diagrams
- Implementation (~5 pages)
  - Like the demo, with screenshots and practical details (libraries, tools, …)
- Known issues and limitations (~0.5 pages)
- Conclusions and future remarks (~0.5 pages)
- References (~2 pages, not counted in the total)

Weights in pages are demonstrative

Evaluation (6 points are given based on the written test)

- Report quality criteria (total: 10 points)
  - Preface (0.5 pts.)
  - Background: domain (1 pt.)
  - Background: blockchain basics (1 pts.)
  - Architecture (3 pts.)
  - Implementation (3 pts.)
  - Conclusion (0.5 pts.)
  - Bibliography (1 pt.)
- Project demo criteria (total: 14 points; to be held live and online)
  - Smart contract code quality: 3 pts.
  - DApp UI: 3 pts.
  - Full functionality: 8 pts.
  - No contract code documentation (comments): -1 pt. (it is a malus)
  - No use of events: -1 pt. (it is a malus)
  - Lack of use of Truffle or similar environments: -2 pts. (it is a malus)
  - Missing integration with MetaMask or similar wallet: -2 pts. (it is a malus)