

## Description

In this group project you will develop a decentralised light-Twitter. No need to sign up or log in, but this is a simple complete CRUD project in several steps:

- Development of a CRUD API in Solidity
- Deployment to Ropsten
- Development of a front-end to interact with your smart contracts

You can have an example of such a project here:

<http://twitterclone.co/>

## Learning Objectives and Outcomes

You will learn how to lead an Ethereum project, end-to-end:

- Solidity development
- Unit tests
- Deployment

## Specs

- Any user should be able to read all the tweets
  - All the tweets should be ordered in a chronological order (either of 2 ways)
- Any user should be able to write tweets
- A user should be able to update a tweet he wrote
- A user should be able to delete a tweet he wrote
  - Deleting a tweet should NOT affect the access to the other tweets
  - Deleting a tweet should NOT affect the order of the other tweets
- Smart contract(s) should be deployed to Ropsten
- Smart contract(s) should be tested with at least 80% line coverage
- Smart contract(s) should have a containerisation using Docker to run, run tests, migrate to Ropsten (with coverage)

## Group project grading

Feature				Total
<b>Smart contracts</b>	Main code	Unit tests		14
Create tweet	2	1,5		3,5
Read tweets	2	1,5		3,5
Update tweet	2	1,5		3,5
Delete tweet	2	1,5		3,5
<b>DevOps</b>				2
Deployment to Ropsten	2			2
<b>Mandatory</b>				4
Documentation	1,5			2
Clean code (indentations, linters)	1,5			2
<b>Total</b>				20
<b>Bonus: front-end</b>				
<b>Interface</b>	Integration	Communicates with blockchain		12
Create tweet	1	2		
Read Tweet	1	2		
Update tweet	1	2		
Delete tweet	1	2		
<b>Functional</b>				8
web3 instance	3			
Button to connect Metamask	2			
Documentation	1,5			
Clean Code (formatters, linters)	1,5			