First Sprint

- **Roles**: Scrum Master (change every week), Quality controller (One for all, all for one), Team member.
- **Project planning** (including table with task dependencies, Gantt Chart for Sprint 1).
- Include **a burndown chart** for the Sprint.
- Perform the following tasks for this Sprint:
  - **Identify 15 features** (5 basic + 7 chosen from existing system + 3 new proposals which should allow communication with other systems' services or devices).
  - **The basic features are**: register account, add contact, post a content, get notifications, send message.
  - **Build the BPMN model** for the System's features.
    - Structure the process specifying a main process.
    - A subprocess for each feature.
  - **Perform Requirements elicitation**:
    - Elicitation (**yes! You have to discover the requirements**) of the system requirements should be performed using SysML Requirements Diagram.
      - Functional requirements (related to 15 features above).
      - Non-functional requirements: identification of contributions and conflicts (4 NF requirements).
  - **Prototype (front-end and back end)**: Implement **the basic features only**, in your preferred programming language / IDE.
  - **Proposal**
    - **For the next Sprint, Besides the 5 basic ones, choose 4 features** from the set of ***7 chosen from existing systems and 1 from the 3 new ones.***
    - Choose **3 Non-Functional** requirements.
    - **These features and NFRs must be negotiated with me.**

Second Sprint

- **Corrections of the 1st Part.**
- **Architectural design.**
  - **Context diagram:** use a Block Definition Diagram (BDD) linking actors to the system as a black box.
  - **Use case view:** A Use Case diagram for all the features (5 basic + 4 chosen + 1 new).
  - **Logical view:** use a Block diagram to show the decomposition of the system for all the features (5 basic + 4 chosen + 1 new), including interface, control and domain objects.
  - **Process View:** Sequence diagrams only for the 4 features chosen + 1 new.
  - **Development view:** Use BDD to organize the systems into subsystems and internal block diagrams (IBD) for each subsystem showing how the subsystem parts are connected.
  - **Deployment view:** Use BDD to show the allocation of subsystems blocks to machines and servers blocks. Use IBD to show how the machine blocks are connected.
  - Justify how the NFRs are satisfied and which architectural patterns are used.
- **Prototype.**
  - Deliver the protype to validate all the system features (5 basic+4 chosen+1 new).
- **Variability modelling.**
  - Think the MyLinkedIn as a Software Product Line (SPL) by building a **feature model** to represent the commonalities and variabilities for different configurations. Have as the basis the 15 initial feature from the 1st Sprint and add new features.
  - Configure 2 possible applications.
- **Gantt chart** updated and **a burndown chart** for the 2nd Sprint.

Due dates for the practical work:

- **1st** Sprint: 26/10
- **2nd** Sprint: 3/12.
  - Discussions: 4-5/12
- Teams of **5**