U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Informática Industrial

Guidelines for Practical Implementation of the Factory Automation System

Mário J. Sousa, Luís Almeida, Pedro M. Santos

2019/2020 – 2º semester

Outline

1. Technology Risk Assessment
2. Development Process
3. Team Management
4. Version Control Systems
5. Object-Oriented Programming
6. Next steps

1. Technology Risk Assessment

- For each technological component in your system:
  1. Identify potential technologies to use;
  2. Select a subset (one or more); build **Minimum Working Example (MWE)** for each
- Rank options regarding:
  - Ease of development
  - Suitability to the requirements of your system architecture
  - Ease of integration with other technological components (i.e., are there libraries to connect to the other components; are these solutions new or well-established/tested)
  - Ease of maintenance when system is in operation (**Product Lifecycle Management**)
  - Other criteria you might find relevant
- Keep this effort constrained in time (apr. 1 week, if that much)

2. Development Process

Hint: Target a **modular** and **bottom-up** development approach

1. Identify the minimum set of implementation to have a working system (the core of your sytem)
    1. Assume one single type of order
    2. Set up one operating cell in the shopfloor
    3. Set up connection MES – PLC (through OPC-UA)
    4. Set up connection to database to obtain shopfloor status
    5. Set up very basic path planning (or even a static path)
2. Then build on top of that & add non-core functionalities
    1. Interpretation of ERP XMLs
    2. Permanent/event-driven update of database with PLC status
    3. Etc…

Takeaway: build a minimally working system, then improve it modularly

3. Team Management

- Task planning & effort allocation: who does what?
  - Divide work in tasks; identify precedences and opportunities for parallel work
  - Assign tasks to developers and assign reasonable timeframe *(suggestion: double your initial estimate)*
- Project Management Tools
  - Tools: MS Project, Redmine,…
  - **Task & Issues definition:** Redmine defines not only “Tasks”, but also “Features” and “Bugs”
  - **Reporting & Accountability:**
    - “Finished this task that was assigned to me”
    - “Found a bug, who’s responsible for it or in charge of addressing it?”
  - **Specification & User Manual:** how is the system actually built, and how to use it?
    - Through Wiki, documents…
- Work Methodology & Time management
  - **Work flow:** Sprints? Weekly deliveries? Ad hoc one-to-one meetings, or weekly meetings?
  - **Leadership/monitoring:** someone in charge of checking everything is going as planned? If indecision, who makes the final call?
  - **Gantt chart:** I suggest you do one to keep track of your evolution

Snapshot from Redmine

https://redmine.fe.up.pt/

Redmine

Visão geral | Download | Actividade | Planificação | **Tarefas** | Notícias | Wiki | Forums | Repositório

Tarefas

Filtros

Estado: aberto — Adicionar filtro

Opções

Aplicar | Limpar

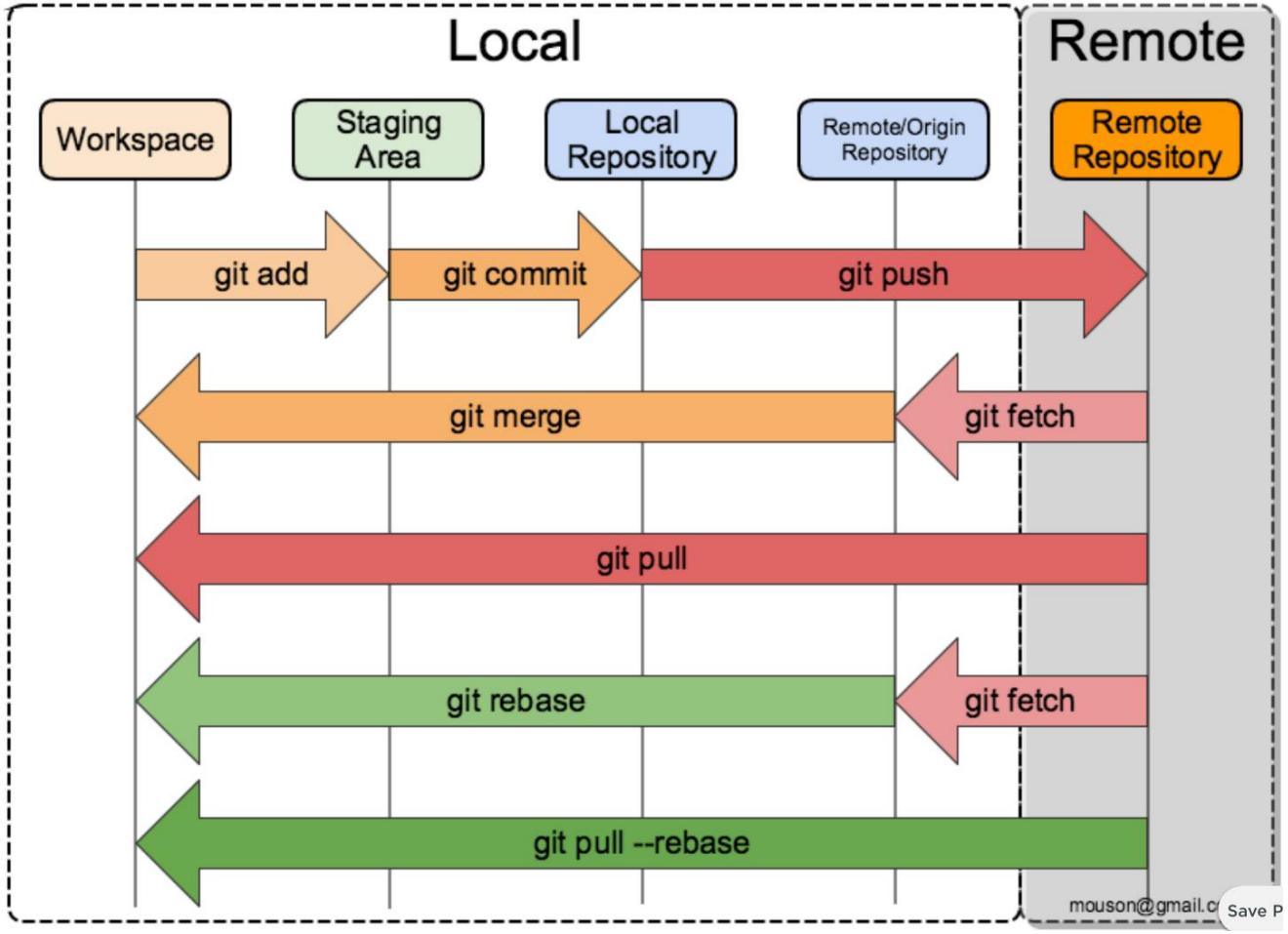
#	Tipo	Estado	Assunto	Alterado	Categoria
33156	Defect	New	allow mobilebrowser zoom	2020-03-16 11:40	UI
33153	Feature	New	UI feature to quickly change issue status	2020-03-16 13:11	UI
33151	Feature	New	Provide status for issue children via REST API	2020-03-15 15:09	REST API
33148	Defect	New	application stuck if query with filter "Issue" and with large amount of IssueIDs	2020-03-13 17:44	Issues filter
33140	Defect	Confirmed	Gantt bar is not displayed if the due date is the first day of the month	2020-03-16 07:07	Gantt
33139	Defect	New	Redmine.pm not working with socks	2020-03-13 04:51	SCM extra
33138	Defect	New	Apache crashes with Redmine.pm and RedmineCacheCredsMax active	2020-03-13 10:32	SCM extra
33129	Feature	New	Restore "Latest Projects" to Home page	2020-03-11 15:20	UI
33127	Defect	New	Assignee icon is misaligned when print stylesheet is applied	2020-03-11 10:43	Issues
33126	Feature	New	Add UserCustomField value to csv of users list	2020-03-15 06:21	Accounts / authentication
33121	Defect	New	IssueQuery not usable from plugins	2020-03-10 15:00	Plugin API
33118	Patch	New	Detect plain diffs in e-mail submitted issues and map to attachments	2020-03-10 11:21	
33117	Patch	New	Redirect to users_path instead of edit_user_path in order to avoid confusion	2020-03-10 22:11	Accounts / authentication

New issue

Tracker * Bug
Subject *
Description
Status * New
Priority * Normal
Assignee
Parent task
Start date 16/03/2020
Due date dd/mm/aaaa
Estimated time Hours
% Done 0 %
Files Escolher Ficheiros Nenhum ficheiro selecionado (Maximum size: 15 MB)
Watchers Ana Aguiar, Luis Miguel Sousa, Pedro Salgueiro Santos
Search for watchers to add

4. Version Control Systems

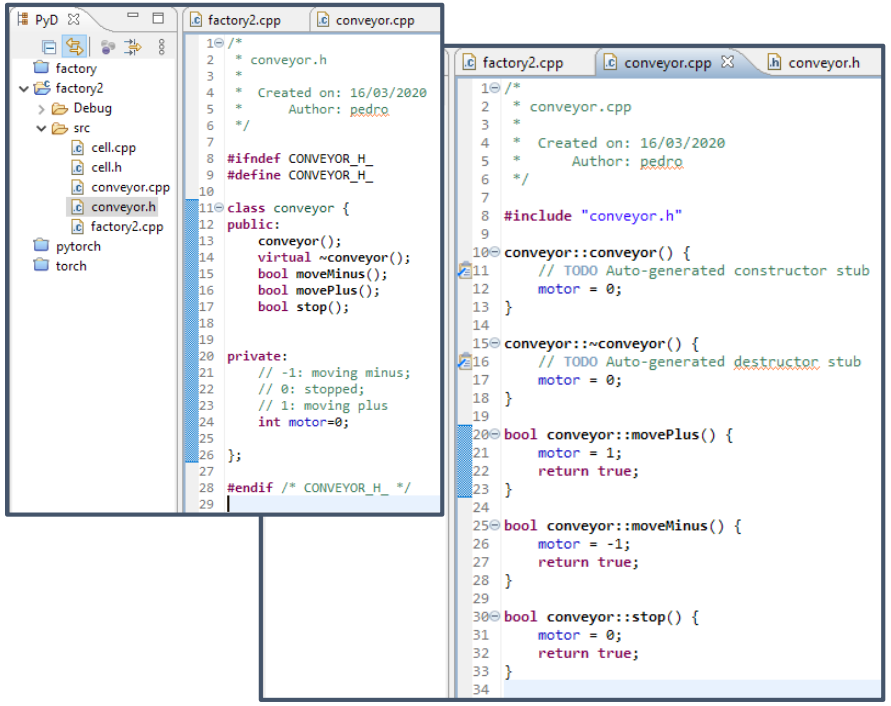
- Git is a widely-used version control system
  - Other well-known system is SVN (centralized as opposed to decentralized, as Git is)
  - Essential to keep track of the evolution of a large software project (no GDrive or Dropbox here ☺)
- Some features
  - History of project evolution is recorded; developers can **ROLL-BACK** to previous version when bug is found
  - Tools for merging disparate versions are available
  - All *Commits* must be commented



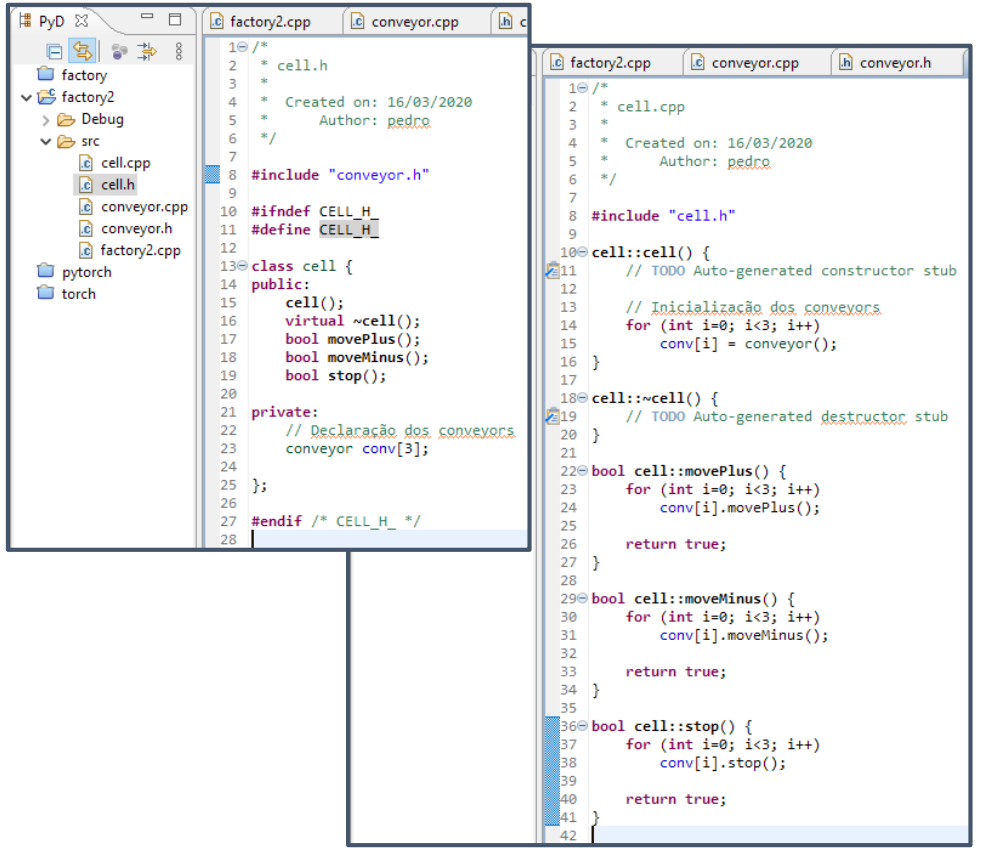
Taken from: Jianping Zeng, https://medium.com/@zjpjack/reverting-modified-in-4-stages-in-git-f3997f526902

5. Object-Oriented Programming

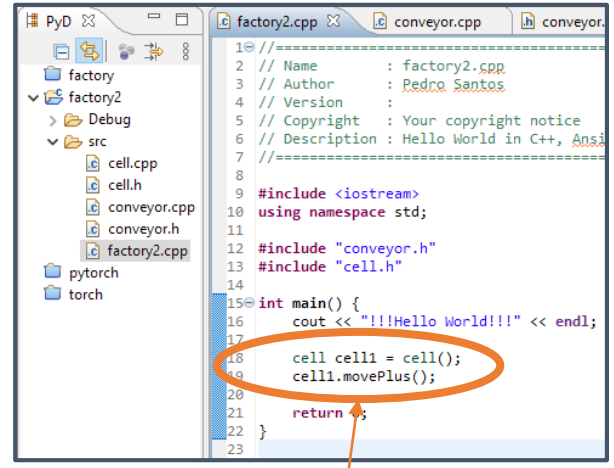
conveyor.h /.cpp



cell.h /.cpp



factory.cpp (main)



Object instantiation

6. Next Steps

- Define tasks
- Assign developers
- Draw timeline

1. Set up project management & VCS tools
2. Create MWE of potential technologies
3. Start development