

# PROJECT

G. Molines  
2018-2019



# GOALS & EXPECTATIONS

# Goals

- Study and define all aspects of the software architecture
- Experiment with the architect's approach
- Be able to explain the technical choices
- Code the MOST RELEVANT parts

# Expected deliverables

- POC complete by Nov 2<sup>nd</sup>
- Architecture is **documented**
- Weekly status email
  - What was done
  - What is planned
  - Why the difference from last week
  - Issues
  - Risk
  - RYG flag
  - Title: [SI5-AL][TeamX][ProjectStatus]



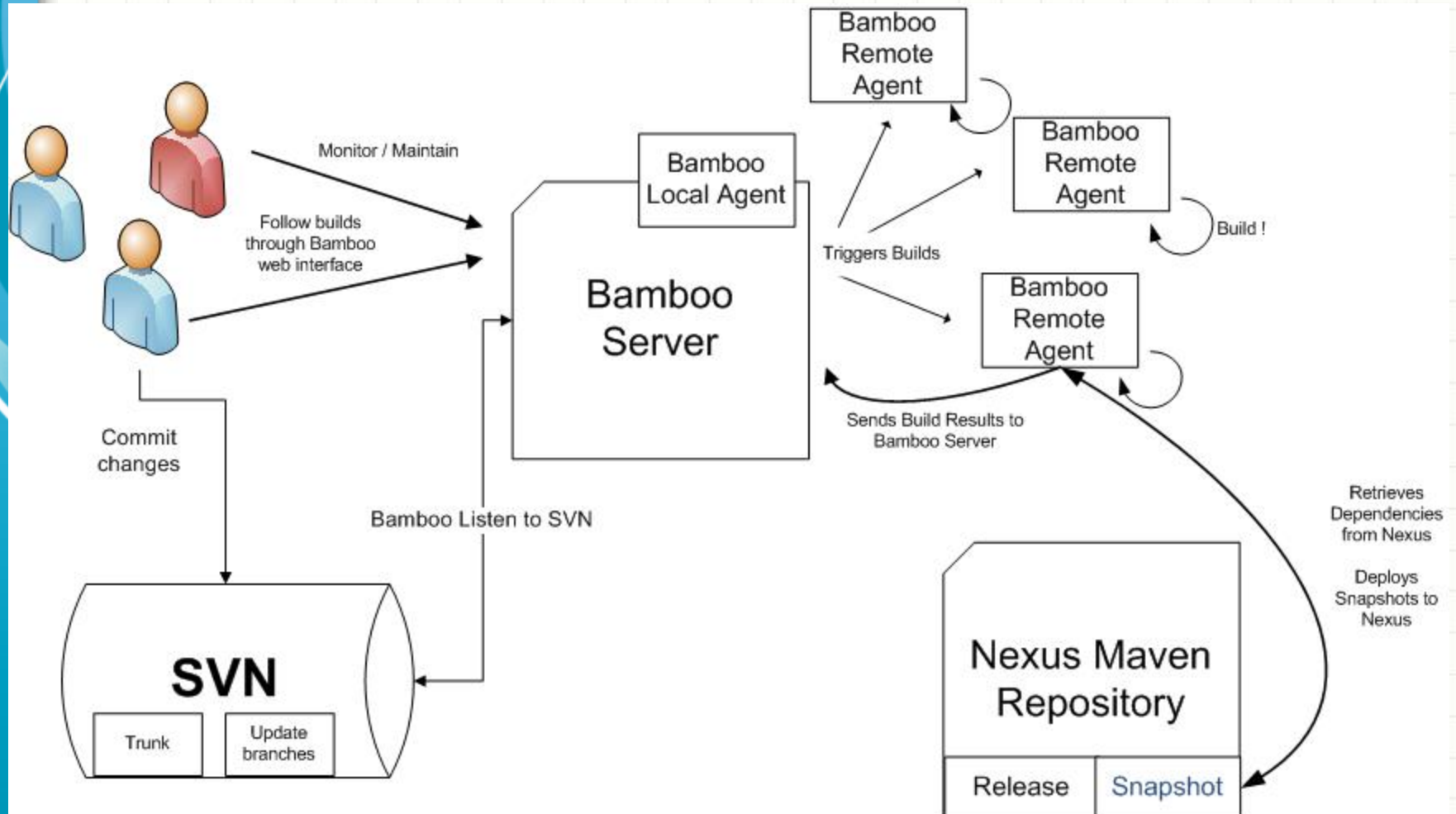
# Expectations

- Read your mail
- Work seriously
  - and regularly
- No copy-paste
- Commits
- Unit-tests
- Continuous Integration
- Register defects





# Continuous integration



# Organization

- Team work (teams of 4)
- Continuous, iterative approach mandated
- Produce architecture deliverables
- Be creative
- This is not a programming class
  - Good programming practices are still a good idea...

# Roles within team

- Agile principle: self-organizing team
- But
  - How are you organized?
  - Role != person
  - All do something



# Timeline

Arch overview

Risk identified

Scope defined

POC demo



Roadmap

Tech. choice

Final demo



Exam

# Timeline

- 38: choose a topic, build the teams
  - 39: define scope, users. Identify risk areas, investigations. Define use cases
  - 40: general architecture, technology choice, external interfaces
  - 41: component details, internal interfaces. Ext. systems mocked.
  - 42: Continuous Integration Env. built. Walking skeleton
  - 43: Main risk mitigated
  - 44: coding enough of the rest for the POC
  - 45: POC complete
- 
- Testing occurs all along
  - This timeline assumes iteratively reviewing past achievements

# TOPIC & VARIANTS



Main topic

***BlablaMove***

# Description

- BlablaMove helps students move for cheap by using space onboard other people cars to move their goods and furniture as a swarm as those other people drive around.
- Participants trade points for car space. Points can be use to move their own goods later on.
- Short trips are combined together to form a longer delivery service

# Variants

- Each team needs to define the architecture of the WHOLE system
- Each team will focus on ONE variant in the list below (or suggest your own)
- The POC implementation will
  - Implement the focus area
  - Mock up the other components



# Variant list

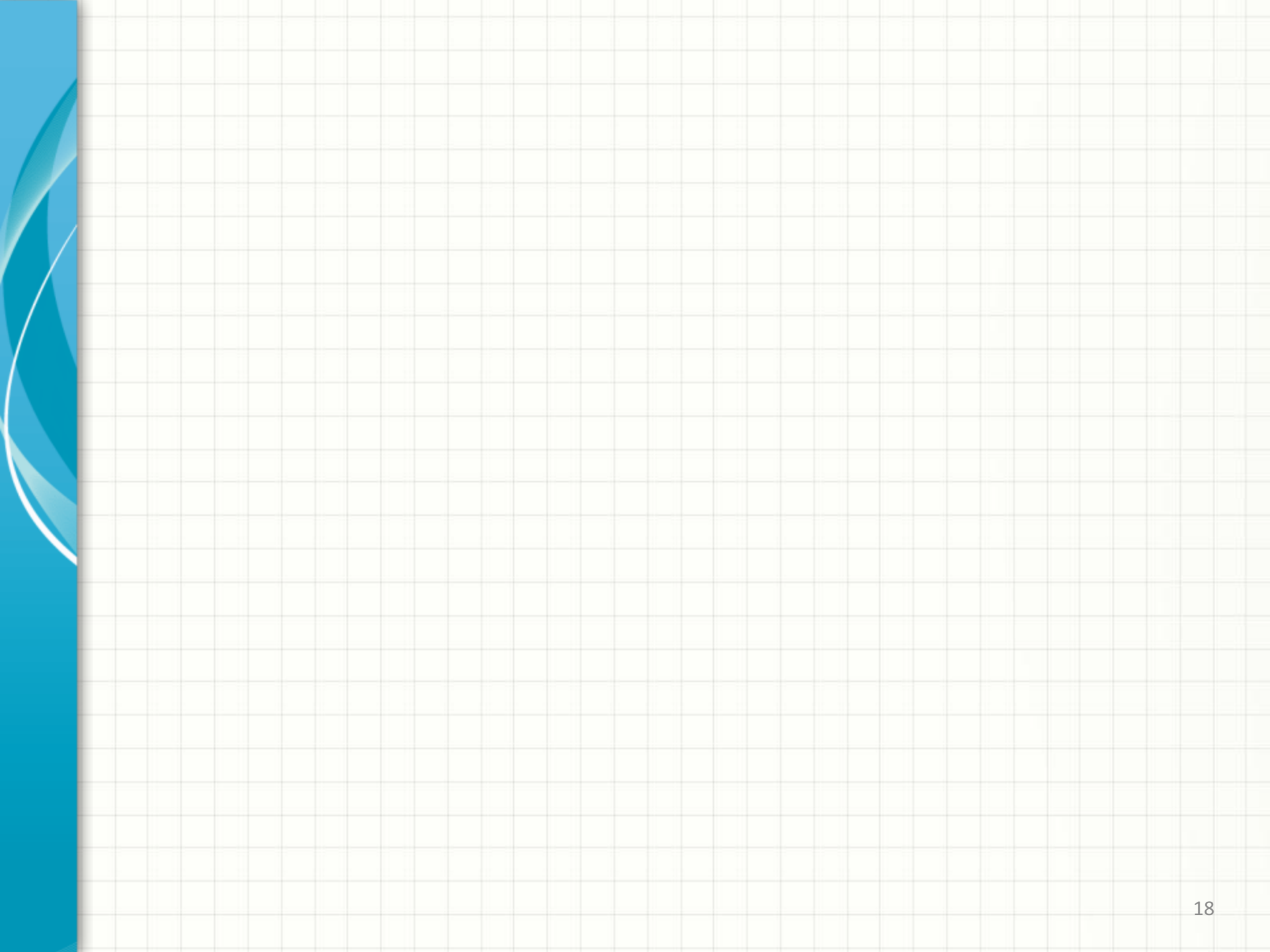
- V1: Routing composition
  - Compose small routes to form the final delivery
- V2: Billing
  - Distribute points according to service
- V3: Volume assessment
  - Snap a picture of your apartment, get an estimate of the volume to be moved
- V4: Mobile tracking app
  - Follow your goods along the way

# Variant list

- V5: change management
  - A car breaks down / changes direction, compute backup plan
- V6: insurance
  - Insurance breakdown for the goods moved, externalized
- V7: dispute resolution
  - Goods not delivered? Broken? Track responsibility
- V8: system monitoring
  - Track events, monitoring dashboard (system + business)
- V9: Mobile app for transporters
  - Offer service, maximize occupancy, commitment



**QUESTIONS?**





# APPENDIX