SOEN 390 Logistics

Dr. Nikolaos Tsantalis

Department of Computer Science and Software Engineering

Concordia University

Team formation

- You are responsible for forming your own team

- Rules:
  - Minimum #members: **8**
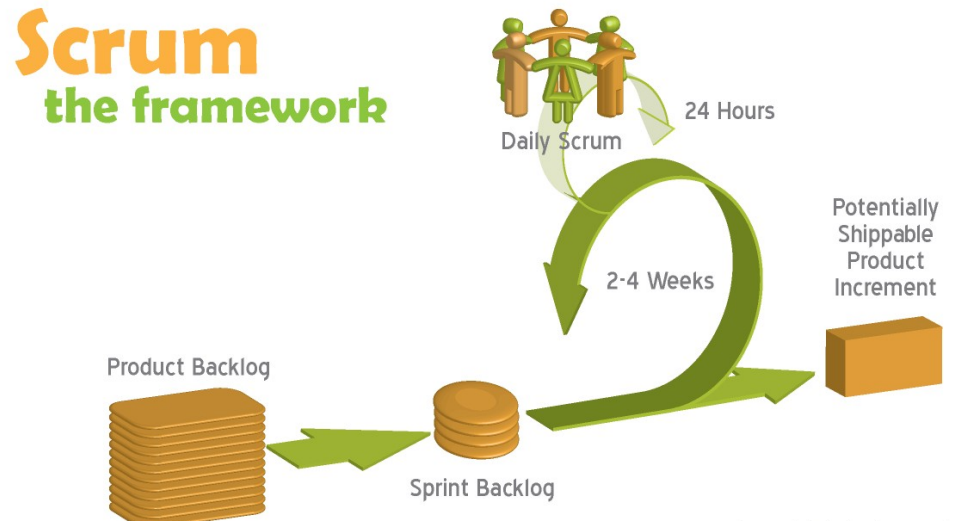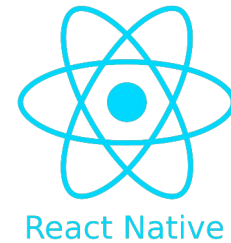 - Maximum #members: **10**

Bidding document

- Project bidding on **Slack**

- **Deadline**: Friday, January 17th

- Write a bidding document explaining why your team is **qualified** for the project/platform
  - Previous experience/knowledge
  - Programming skills in Java (Android), C# (Xamarin), HTML5/JavaScript/CSS (Ionic)
  - Testing frameworks
  - Research on similar mobile apps

Development process

- Fixed iterations (sprints) of **2 weeks**

- **6** sprints in total

- Every 2 sprints, a **release** is made (working prototype)

- **3** releases in total (releases will be marked, 30% each)

- **Lightweight** documentation (incremental updates)

Scrum
the framework

24 Hours

Daily Scrum

Potentially
Shippable
Product
Increment

2-4 Weeks

Product Backlog

Sprint Backlog

Development Platforms

ZenHub

Sprint/Release structure
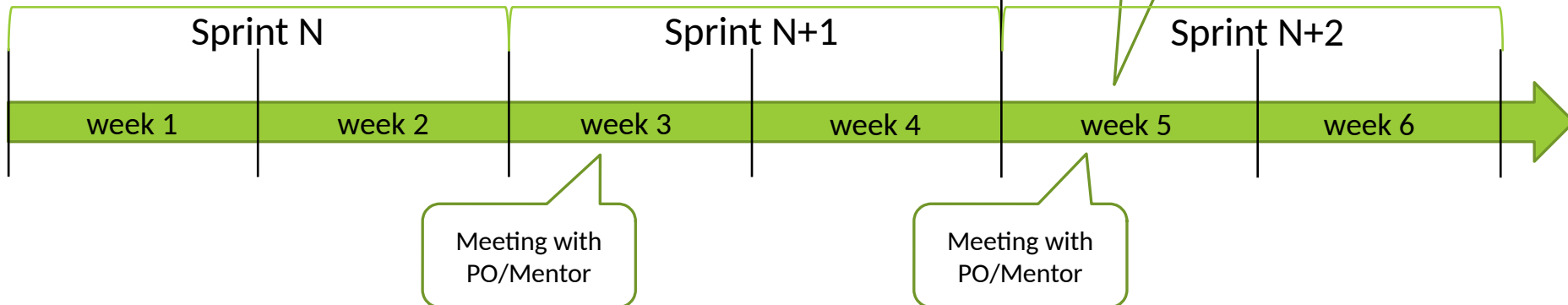
Recommended Plan Sprint #1

1. Setup your GitHub repository and yml scripts for GitHub workflows

2. Create your project Backlog in ZenHub (make a user story for each project requirement)

3. Define the personas representing potential users and describe some of the tasks they will perform on the system. You should consider how to make your software accessible to more people. People with disabilities (color-blind, blind, deaf, people with hand injuries), illiterate people, or marginalized people.

4. Create UI mockups for the user stories you will implement in Sprint 2

5. Think about and document the design of your app (basic domain model, component diagram). Explore external libraries/frameworks that might be useful for your project and discuss their pros and cons.

6. Plan your Sprint 2 in ZenHub by adding some user stories from your backlog (add features than can be completed within 2 weeks)

7. Conduct and document your Sprint 1 Retrospective. Meet with your product owner and get feedback for your mockups and your backlog

Recommended Plan Sprint #2

1. Implement the planned features

2. Write unit tests for your code

3. Submit your features in Pull Requests. Assign team members for code review. There should be active discussions and suggestions for improvements.

4. Create UI mockups for the user stories you will implement in Sprint 3

5. **Investigate** how you can make **automated System/UI tests** for mobile apps. In the meantime, some team members will do manual system testing for the implemented features and report well documented bug reports on GitHub as issues.

6. Install and run SonarQube for your project. Summarize the issues detected by SonarQube and discuss how and when you will address them.

7. Conduct and document your Sprint 2 Retrospective. Meet with your product owner for acceptance testing and getting feedback

8. Prepare your presentation for the Instructor and practice it with all team members at least once.

9. Submit your Release 1 report on Slack

10. Plan your Sprint 3 in ZenHub by adding user stories from your backlog (add features than can be completed within 2 weeks)

Recommended Plan Sprint #3

1. Implement the planned features

2. Write unit tests for your code

3. Submit your features in Pull Requests. Assign team members for code review. There should be active discussions and suggestions for improvements. These suggestions could also include refactorings. Document systematically your refactoring efforts by collecting commits and PRs with specific types of refactorings

4. Create UI mockups for the user stories you will implement in Sprint 4

5. At this point you should have automated System/UI tests for all implemented features so far. You will do manual system testing for the ongoing features and report well documented bug reports on GitHub as issues.

6. **Investigate** potential design patterns that could be useful for your project. Justify their need in your project.

7. **Investigate** how you will conduct usability testing and create the required infrastructure (setup and try usability testing tools).

8. Plan your Sprint 4 in ZenHub by adding some user stories from your backlog (add features than can be completed within 2 weeks. Note that in the next Sprint you will conduct usability testing. Make a realistic plan)

9. Conduct and document your Sprint 3 Retrospective. Meet with your product owner for acceptance testing and getting feedback

Recommended Plan Sprint #4

1. Implement the planned features following the same practices (PRs, code reviews, unit tests)

2. Introduce design patterns in your project. Document commits where your design pattern instances were introduced and evolved (adding new abstract methods, adding new subtypes)

3. Address technical debt and security warnings from SonarQube. Update your SonarQube (code quality) report

4. Create UI mockups for the user stories you will implement in Release 3 (Sprints 5 and 6).

5. Add automated System/UI tests for the newly implemented features.

6. Recruit users and **conduct your first usability testing**. Collect and analyze your usability test results. Make well documented issues for UI improvements and bugs found during testing.

7. Conduct and document your Sprint 4 Retrospective. Meet with your product owner for acceptance testing and getting feedback

8. Prepare your presentation for the Instructor and practice it with all team members at least once.

9. Submit your Release 2 report on Slack

10. Plan your Sprint 5 in ZenHub by adding some user stories from your backlog (add features than can be completed within 2 weeks)

Recommended Plan Sprint #5

1. Address all UI issues discovered in usability testing

2. Implement the remaining planned features

3. Write unit tests for the new features

4. Submit your features in Pull Requests. Assign team members for code review. There should be active discussions and suggestions for improvements. These suggestions could also include refactorings. Document systematically your refactoring efforts by collecting commits and PRs with specific types of refactorings

5. Add automated System/UI tests for the newly implemented features.

6. Address technical debt and security warnings from SonarQube. Update your SonarQube (code quality) report

7. Plan your Sprint 6 in ZenHub by adding any remaining user stories from your backlog.

8. Conduct and document your Sprint 5 Retrospective. Meet with your product owner for acceptance testing and getting feedback

Recommended Plan Sprint #6

1. **Conduct a second round of usability testing**. Compare the current usability test results with the previous ones to show the improvement in user satisfaction or other usability metrics.

2. Add automated System/UI tests for all implemented features.

3. Address all remaining technical debt and security warnings from SonarQube. Update your SonarQube (code quality) report

4. Conduct and document your Sprint 6 Retrospective. Meet with your product owner for any remaining acceptance testing

5. Prepare your presentation for the Instructor and practice it with all team members at least once.

6. Submit your Release 3 report on Slack

How are you evaluated?

- The instructor evaluates your **software engineering practices**
  - Planning
  - Team management
  - Reports (design, testing, quality)
  - Source code reviewing
  - Proper use of engineering tools
  - Research/investigation
  - Communication
  - Professionalism
- The exact **evaluation criteria** can be found at marking.pdf

First meeting with the Product Owner

- Schedule a meeting **time** that works for the team and PO.

- Prepare the backlog with **user stories** before the visit

- Prepare some **UI prototypes** before the visit

- Make sure you leave the meeting with all your questions answered & doubts clarified

- Be **positive** and **enthusiastic**

Tutorial Schedule

- January 20, Zenhub
- January 27, CI pipeline on GitHub
- February 3, Testing Mobile apps
- February 10, SonarQube

Q QA Monday 17:45-18:35

Q QC Monday 19:15-20:05

R SA Monday 17:45-18:35

R SC Monday 19:15-20:05