# CS Task 2: se process

## Plan-driven, agile or combination

To decide which SE Process we’re going to use, we first collect pros and cons:

|  |  |  |
| --- | --- | --- |
| Pros | Cons | |
| **Agile process** | | |
| Flexible | | Customer needs to be available |
| Better scale able | | Customer may has new features every week |
| Permanent releases | | More effort because more meetings, changing requirements |
| Work together with customer, more inputs | |  |
| SW gets tested over and over again by customer | |  |
| Not all requirements are known or/and available | |  |

|  |  |
| --- | --- |
| Pros | Cons |
| **Plan driven** | |
| Requirements are defined | Customer cannot request changes during development |
| Customer only needs to be available during defining processes at project start | Architecture is fixed by requirements |
| Customer has overview over milestones like testing and rollout. | Inflexible for slightly different customers (i.e. different user-interfaces, if not defined) |
|  | No frequently releases during development |

Conclusion

Therefore we don’t know all requirements from the customer, we choose an agile process.

## Decide which SE process

|  |  |
| --- | --- |
| Pros | Cons |
| **XP** | |
| No extra documentation has to be written | Pair programming is a part of XP, and we’re not able to do so. |
|  |  |

|  |  |
| --- | --- |
| Pros | Cons |
| **RUP** | |
|  | RUP is designed for big companies with large projects |
|  |  |

|  |  |
| --- | --- |
| Pros | Cons |
| **Scrum** | |
| Daily meetings (before work) | Overview over whole project is difficult (keeping track of all tasks) |
| Estimate tasks together, assigned to a team member | Scrum-master shouldn’t be a team member (cannot avoid this one in our project) |
| Short sprint periods with releases (agile) |  |
|  |  |
|  |  |

Decision about SE Process: Scrum

## Design

## Process model

Because we have chosen Scrum, requirements have to be defined first.

## Activities

1. Decide who scrum master is.
2. Define requirements
3. Define stories for backlog
4. Define tasks/subtasks for stories
5. Decide which stories will be used in first sprint

## Relation among activities

All the activities have to be done in serial order. After first sprint there is an review and a planning meeting for sprint two.

## Involvement of stakeholders

* Customer:

Intense communications until requirements are defined.

After each sprint cycle, customer is involved in presentation and review.

….

In the end there’s a final release, a project debriefing with an orgy with alcohol and blackjack and hookers.

* Development team:

Scrum meetings (daily scrum, sprint planning meeting, sprint retrospective meeting)

Development itself.