

Методологии за разработване на софтуер

Software Development Models

- Линейни модели:
 - Waterfall model.
 - V-model.
- Циклични модели:
 - Interactive and Incremental model:
 - Agile:
 - Scrum.

Какво е Development model?

- Development model (methodology) – модел/метод за поризводство – набор от практики и процедури за организиране на производствения процес.
 - Това е систематичен инженерен подход, за организиране и управление на софтуерни проекти.



Етапи в софтуерно производство

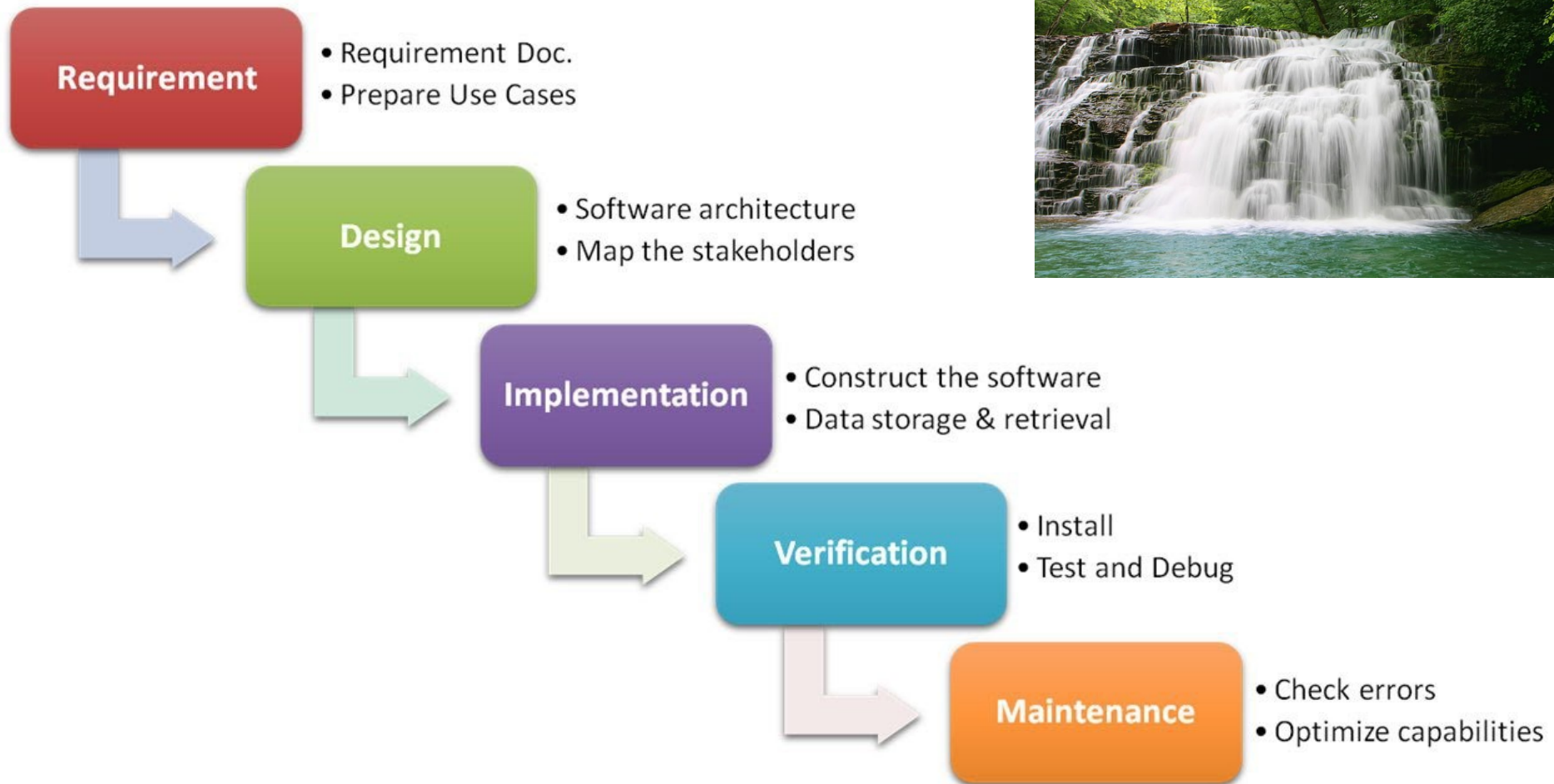
- ▮ Документиране на изискванията.
- ▮ Дизайн на софтуерния продукт.
- ▮ Имплементиране на софтуерния продукт.
- ▮ Тестване на софтуерния продукт.
- ▮ Деплойване на софтуерния продукт.
- ▮ Поддръжка на софтуера.

The Waterfall Model



- ▮ Първия фундаментален модел за софтуерно производство.
- ▮ Прост и добре познат модел.
- ▮ Към следваща стъпка от модела се преминава само след изпълване на предишните.
- ▮ Тестването се извършва накрая на производството.

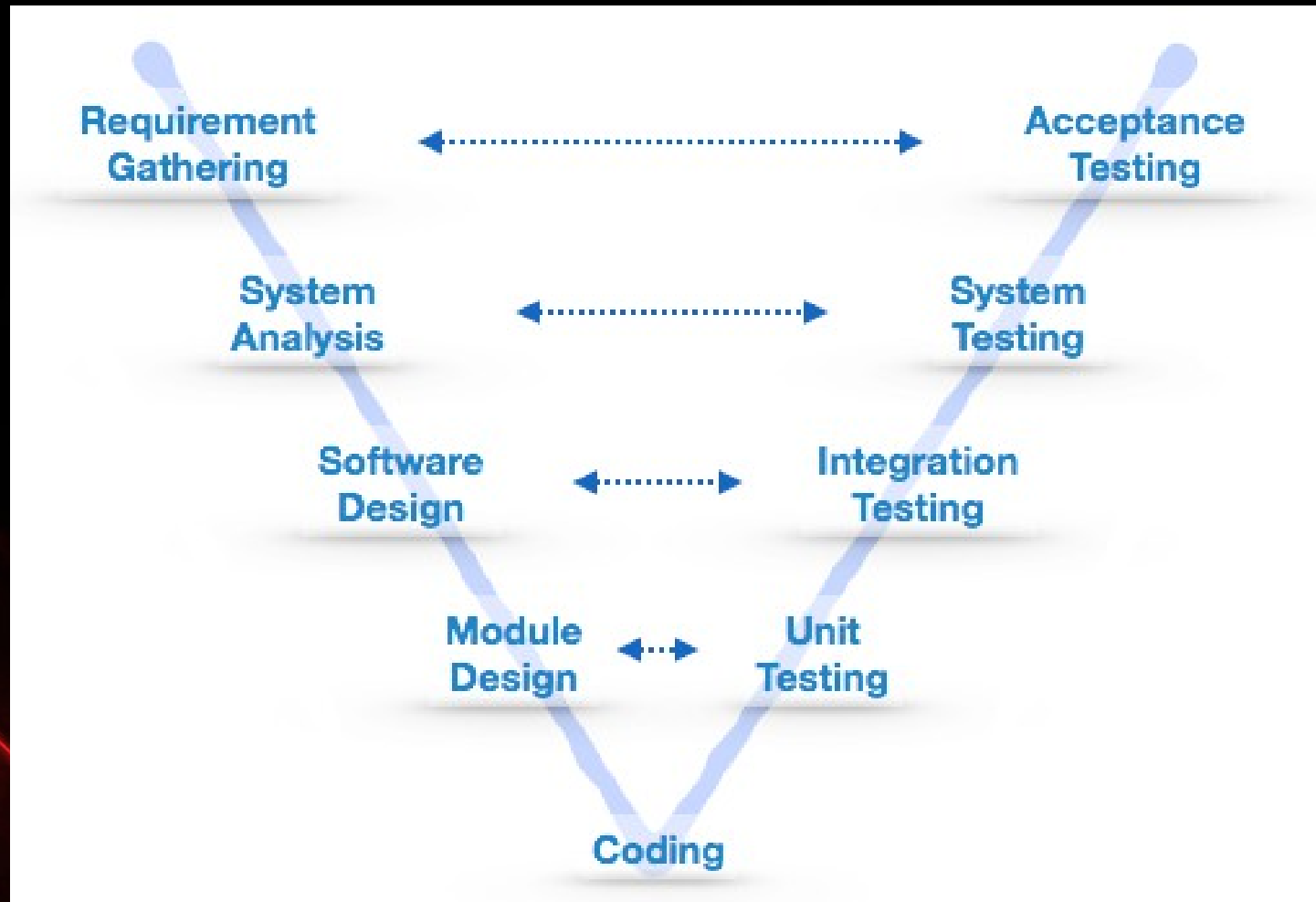
The waterfall process



Sequential Development model

- ▮ Sequential Development Model – известен като V – model.
- ▮ Модела се състои от две разклонения:
 - ▮ Development tasks – процеса на дизайн и имплементиране на кода.
 - ▮ Testing tasks – Верифициране и интегриране на системата.

Схема на V-model



Development Tasks

1. Определяне на изискванията:
 - определяне на клиентските изисквания.
 - определяне целта на системата, характеристики, компоненти и др.
2. Функционален дизайн:
 - Свързване на функционалностите и диалозите на новата система.

Development Tasks

3. Технически системен дизайн:

- Дизайн на имплементацията на системата.
- Определяне на интерфейсите.
- Декомпозиране на системата в малки, лесно разбираеми подсистеми (системна архитектура).



Development Tasks

4. Определяне на компонентите:

- Дефиниране на всеки един от подсистемите.

5. Програмиране:

- Имплементиране на всеки отделен компонент използвайки програмни езици.

Testing Tasks

1. Тестване на компоненти (unit testing):

- Верифициране на всеки компонент поотделно.

2. Тестване на интеграцията:

- Верифицира се взаимната работа на отделните компоненти. Проверява се дали коректно, работят заедно.



Testing Tasks

3. Тестване на системата:

- Верифициране на системата като цяло.
- Проверява се дали покрива системните

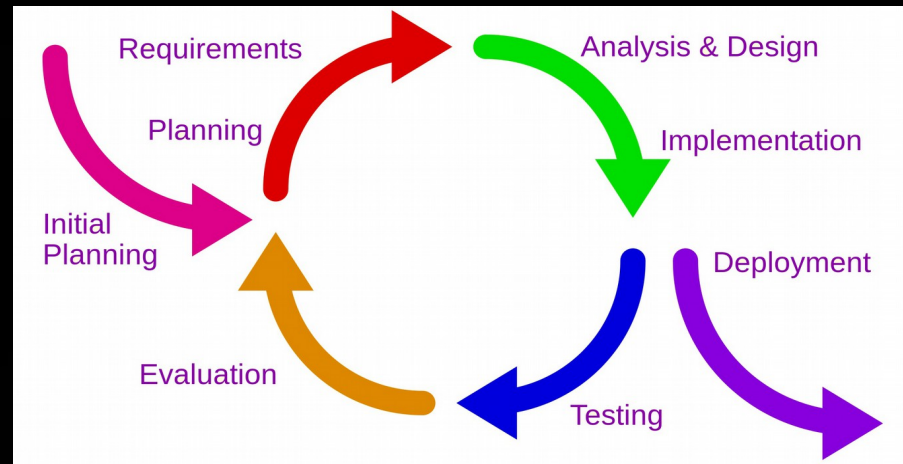
изисквания.

4. Acceptance test:

- Проверява се дали системата покрива клиентските изисквания.

Iterative and Incremental model

- ▮ Iterative and Incremental Software Development model:
 - ▮ Използват се повтарящи се цикли (iterative) за производство на софтуер
 - ▮ Някаква част от софтуера (incremental) се произвежда при всяка итерация.



Iterative and Incremental model

- ▮ Предимства:
 - ▮ Може да се прави Regression testing.
 - ▮ Верифициране и валидиране на всяко парче от софтуера.
- ▮ Недостатъци:
 - ▮ Изисква по-голямо участие на клиента.
 - ▮ Разделянето на функциите и фийчърите, може да е проблемно.

Agile Development

- ▮ Базиран е на Interactive and incremental модела.
- ▮ Изискванията се изграждат чрез колаборация между само организиращи се, мултифункционални отбори (team).



The Agile Manifesto

“Our highest priority is to satisfy the customer through early and continuous delivery of valuable software”

-Manifesto for Agile

Agile характеристики

- ▮ Задачите се разбиват на малки инкременти с минимално планиране.
- ▮ Всяка итерация, целият тим преминава през пълния процес за производство на софтуер (изисквания, дизайн, програмиране, тест и др.)
- ▮ Възможност за релийзване на продукта, на края на всяка итерация.

Agile характеристики

- ▮ Изискванията се коментират лице-в-лице с всеки участник в тийма.
- ▮ Отбора (team) се състои от 5 до 9 човека (Dev, QA, PM, SysAdmin, Support и др.)
- ▮ Рутинни всекидневни срещи лице-в-лице, с участието на всички членове на екипа.

Scrum

- ▮ Scrum - Базиран е на Interactive and incremental модела.
- ▮ Scrum – това е Agile процес, който ни позволява да се фокусираме над бързото и качествено производство на софтуер.



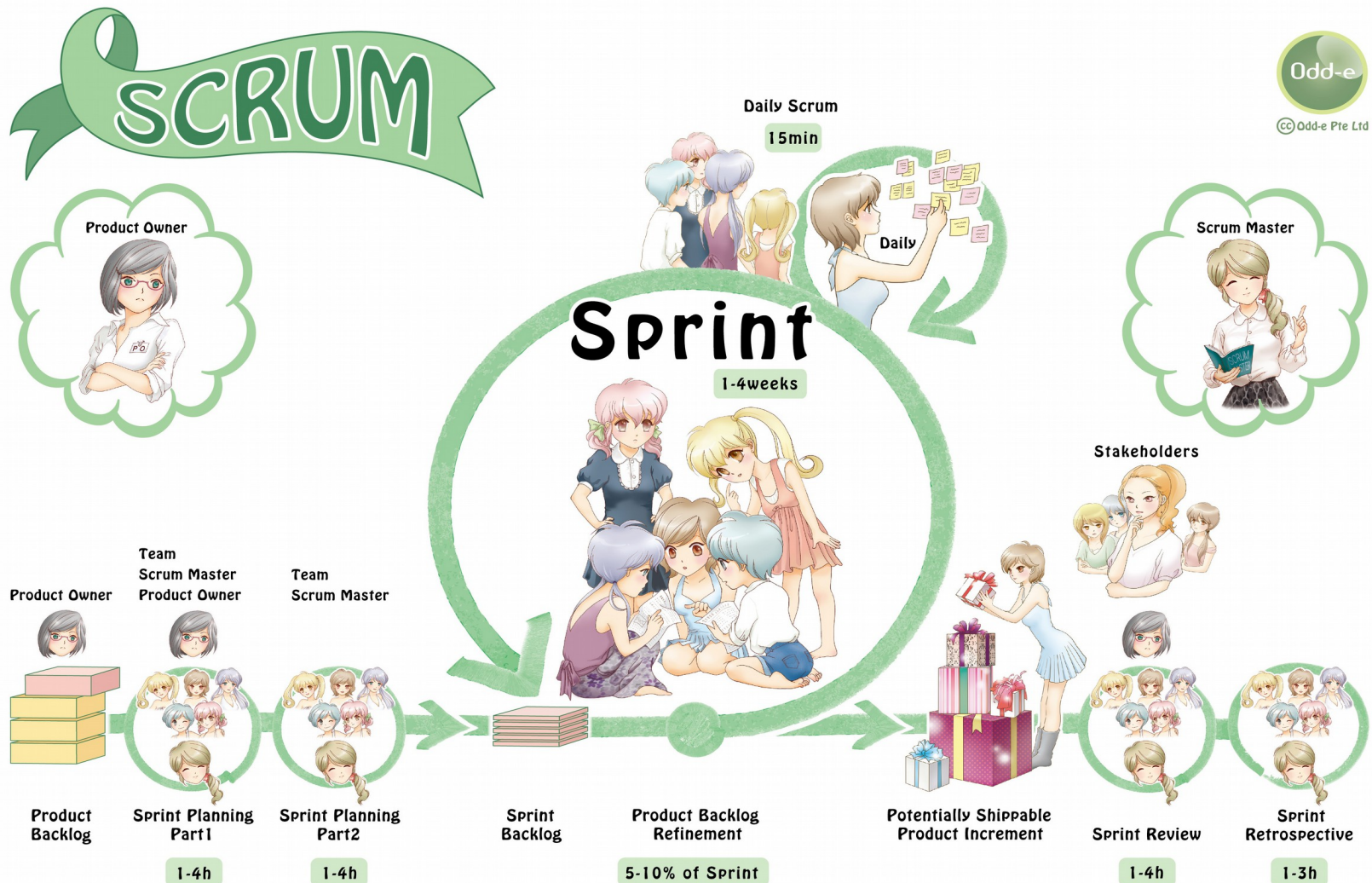
Scrum

- ▮ Отборите се само-организируют и избират най-добрия начин за създаването на високоприоритетните фийчъри.
- ▮ На края на всяка итерация, има стабилна версия.

Основни роли в scrum

- ▮ Scrum master – отговорник за спазването на скръм процесите.
- ▮ Product Owner – представител на клиента.
- ▮ Team – отговорни са за производството на софтуера.
- ▮ Manager – помага за правилното разбиране на задачите и проекта като цяло.

The scrum process



Scrum терминология

- Sprint/Iteration – времеви интервал за един цикъл на скръм методологията (между 1 и 5 седмици).
- Backlog – всички фийчъри, които трябва да се създадат.
- Stand-up среща – всекидневна среща на отбора (около 15 минути).

Въпроси



Задачи за домашна работа

1. Потърсете в Google “The 12 Agile Principles” и ги прочетете.
2. Направете Regression test на мобилната версия на <http://qa.soft-intellect.com/test>
Няма функционални промени, само външния вид на сайта (при мобилни устройства). Желателно е да се тества на Android, iOS и WindowsPhone.