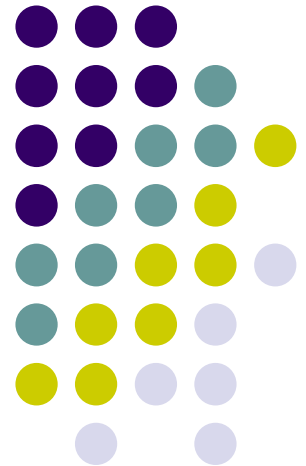
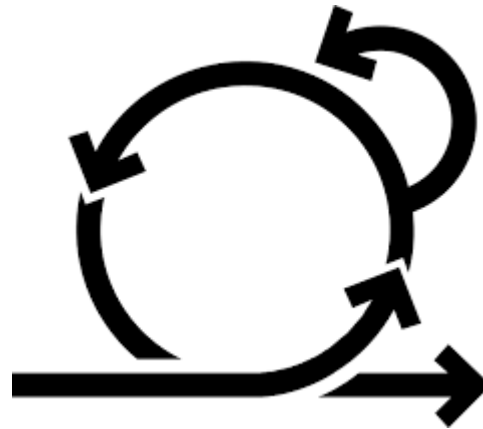
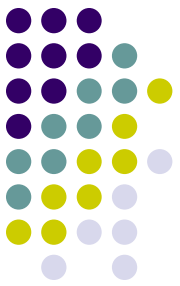




PHP

User Stories



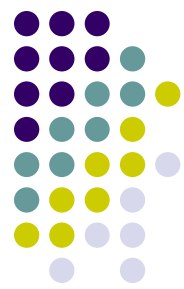


Introduire la notion de User Story



Mettre en œuvre des User Stories en PHP





Présentation



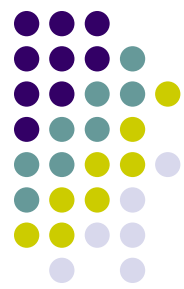
Concept clé dans Scrum



Elément du Backlog



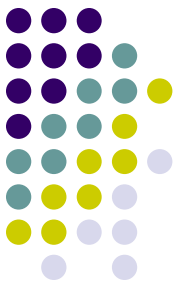
**Description d'une fonctionnalité
souhaitée du point de vue de
l'utilisateur final**



Formalisme

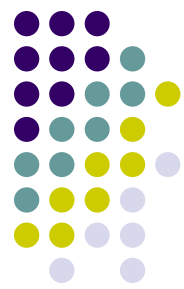


**En tant que [type d'utilisateur],
je veux [une action],
afin de [bénéfice/raison].**



Implémentation

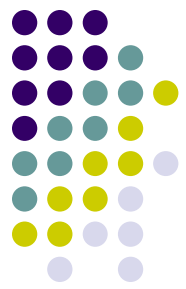
```
class UserStory {  
    public function execute(PARAMETRES) : RETOUR {  
        $validateur = new Valideur();  
    }  
}
```



Exemple



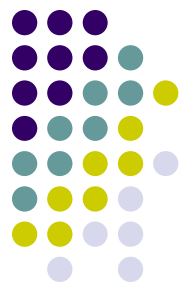
En tant que utilisateur,
je veux créer un compte
afin de pouvoir accéder à mon
espace personnel



Exemple

```
class CreerUtilisateurUS {  
  
    public function execute(string $login, string $password) : bool{  
        $validateur = new Valideur();  
        $validateur->verifierMotPasse($password);  
        // Création de l'utilisateur dans la base de données  
        return true;  
    }  
}
```

Lance une exception



Exemple

```
class CreerUtilisateurUS {  
  
    public function execute(string $login, string $password) : bool{  
        $validateur = new Valideur();  
        $validateur->verifierMotPasse($password);  
        // Création de l'utilisateur dans la base de données  
        return true;  
    }  
}
```

Lance une exception

Pas de try...catch dans le code appelant



Exemple

Propage l'exception

```
class CreerUtilisateurUS {  
  
    public function execute(string $login, string $password) : bool{  
        $validateur = new Valideur();  
        $validateur->verifierMotPasse($password);  
        // Création de l'utilisateur dans la base de données  
        return true;  
    }  
}
```

Lance une exception

Pas de **try...catch** dans le code appelant



Propagation exception

➔ **L'exception n'est pas interceptée dans le code appelant**

➔ **L'exception est donc propagée dans le code appelant du code appelant**





Propagation exception

```
function functionA() {  
    functionB();  
}  
function functionB() {  
    functionC();  
}  
function functionC() {  
    throw new Exception("Une exception est lancée dans functionC");  
}  
  
try {  
    functionA();  
} catch (Exception $e) {  
    echo "Exception capturée : " . $e->getMessage();  
}
```





Injection de dépendance

```
class CreerUtilisateurUS {  
  
    public function execute(string $login, string $password) : bool{  
        $validateur = new Valideur();  
        $validateur->verifierMotPasse($password);  
        // Création de l'utilisateur dans la base de données  
        return true;  
    }  
}
```

dépendance



La dépendance ne doit pas être créée directement dans le code



Injection de dépendance

```
class CreerUtilisateurUS
```

```
{
```

```
    private Valideur $valideur;
```

```
    public function __construct(Valideur $valideur)
```

```
    {
```

```
        $this->valideur = $valideur;
```

```
    }
```

```
    public function execute(string $login, string $password) : bool
```

```
    {
```

```
        $this->valideur->verifierMotPasse($password);
```

```
        // Création de l'utilisateur dans la base de données
```

```
        return true;
```

```
    }
```

```
}
```

**Injection de
dépendance**





Injection de dépendance



Les dépendances sont créées par le code appelant et passées en arguments du constructeur



Injection de dépendance

Création de la dépendance

```
$creerUtilisateurUS = new CreerUtilisateurUS( new Valideur() );  
try {  
    $creerUtilisateurUS->execute("login", "PassePasse12");  
} catch (PasswordException $e) {  
    echo $e->getMessage();  
}
```

Code appelant



Tests

