IFT159
Analyse et programmation
Présentation du cours

Gabriel Girard

Département d'informatique
UNIVERSITÉ DE SHERBROOKE

20 août 2015

Présentation du cours

Présentation du cours

Objectifs

- Apprendre à analyser un problème à partir d'une liste de besoins ;

Objectifs

- Apprendre à analyser un problème à partir d'une liste de besoins ;
- Apprendre à concevoir une solution à partir d'un document d'analyse ;

Objectifs

- Apprendre à analyser un problème à partir d'une liste de besoins ;
- Apprendre à concevoir une solution à partir d'un document d'analyse ;
- Apprendre à implémenter une solution à partir d'un document de conception ;

Objectifs

- Apprendre à analyser un problème à partir d'une liste de besoins ;
- Apprendre à concevoir une solution à partir d'un document d'analyse ;
- Apprendre à implémenter une solution à partir d'un document de conception ;
- Fournir une solution bien codée ;

Objectifs

- Apprendre à analyser un problème à partir d'une liste de besoins ;
- Apprendre à concevoir une solution à partir d'un document d'analyse ;
- Apprendre à implémenter une solution à partir d'un document de conception ;
- Fournir une solution bien codée ;
- Fournir une solution bien documentée ;

Objectifs

- Apprendre à analyser un problème à partir d'une liste de besoins ;
- Apprendre à concevoir une solution à partir d'un document d'analyse ;
- Apprendre à implémenter une solution à partir d'un document de conception ;
- Fournir une solution bien codée ;
- Fournir une solution bien documentée ;
- Apprendre à respecter des contraintes.

Pour la semaine prochaine ! !

Que fait le programme suivant ?

```
#include <iostream>
using namespace std;
int main(){const int T=10;
for (int i=1;i<T;i++){int j;
for(j=0;j<i;j++)cout<<i;
for(j=1;j<=3;j++)cout<<' ';
for(j=T;j>i;j--)cout<<T-i;cout << endl;}}
```

À remettre au plus tard le mardi 1er septembre 2015.

Pour la semaine prochaine!!

Que fait le programme suivant ?

```
#include <iostream>
using namespace std;
int main(){const int T=10;
for (int i=1;i<T;i++){int j;
for(j=0;j<i;j++)cout<<i;
for(j=1;j<=3;j++)cout<<' ';
for(j=T;j>i;j--)cout<<T-i;cout << endl;}}
```

Importance du langage de programmation

- C'est un outil primordial

Importance du langage de programmation

- C'est un outil primordial
- On doit maîtriser sa syntaxe et sa sémantique

Importance du langage de programmation

- C'est un outil primordial
- On doit maîtriser sa syntaxe et sa sémantique
- *« When you are programming, you are teaching possibly to stupidiest thing in the universe, a computer, how to do something »*
  Gabe Logan Newell
  Valve Software
  (Half-life, Portal, ...)

Importance du langage de programmation

- C'est un outil primordial
- On doit maîtriser sa syntaxe et sa sémantique
- *« When you are programming, you are teaching possibly to stupidiest thing in the universe, a computer, how to do something »*
  Gabe Logan Newell
  Valve Software
  (Half-life, Portal, ...)
- Le cours ne se résume pas au langage....

Votre tâche

Vous devez, à partir d'une connaissance des besoins, produire une recette (algorithme) qui permettra ensuite à l'ordinateur de résoudre le problème.

Concepts utiles

- Division du problème en sous-problèmes
- Abstraction

Votre tâche

Si tu vois une personne qui a faim, donne-lui un poisson : tu la nourriras pour un jour.
Mais apprends-lui à pêcher et elle se nourrira toute sa vie.

Lao-Tseu

Présentation du cours

Exemple

Produire les instructions pour monter un meuble en Kit...

Exemple

Créer une recette de petits gâteaux à partir de la description suivante fournie par le client :

Présentation du cours

Les phases

1. *Ingénierie des exigences* ;

Les phases

1. *Ingénierie des exigences* ;
2. Analyse ;

Les phases

1. *Ingénierie des exigences* ;
2. Analyse ;
3. Conception ;

Les phases

1. *Ingénierie des exigences* ;
2. Analyse ;
3. Conception ;
4. Implantation (langage de programmation C++) ;

Les phases

1. *Ingénierie des exigences* ;
2. Analyse ;
3. Conception ;
4. Implantation (langage de programmation C++) ;

5. *Mise au point ;*

Les phases

1. *Ingénierie des exigences* ;
2. Analyse ;
3. Conception ;
4. Implantation (langage de programmation C++) ;

5. *Mise au point ;*
6. *Maintenance.*

Contraintes à respecter

- Efficacité ;

Contraintes à respecter

- Efficacité ;
- Lisibilité et documentation ;

Contraintes à respecter

- Efficacité ;
- Lisibilité et documentation ;
- Fonctionnement sans erreur ;

Contraintes à respecter

- Efficacité ;
- Lisibilité et documentation ;
- Fonctionnement sans erreur ;
- Respect des spécifications ;

Contraintes à respecter

- Efficacité ;
- Lisibilité et documentation ;
- Fonctionnement sans erreur ;
- Respect des spécifications ;
- Fiabilité.

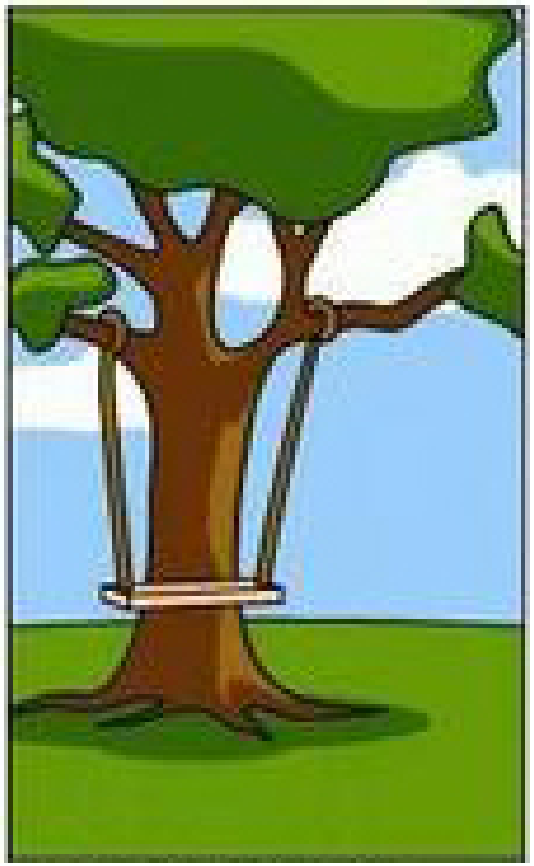
Choix du langage de programmation

- impératif/procédural/objet vs fonctionnel vs logique
- C++, Java, C#, Pascal, VB, ...

Problèmes de développement

Comment le client a exprimé son besoin

Problèmes de développement

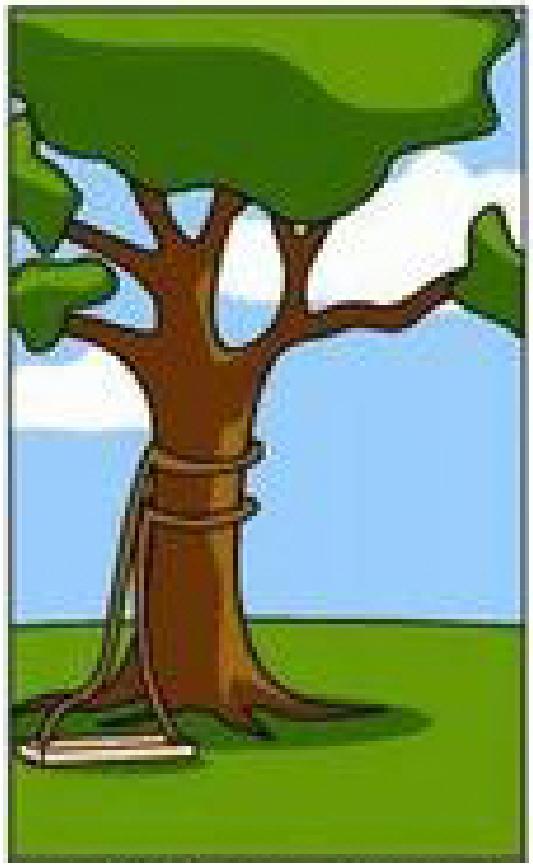


Comment le chef de projet l'a compris

Problèmes de développement

Comment l'analyste l'a conçu

Problèmes de développement

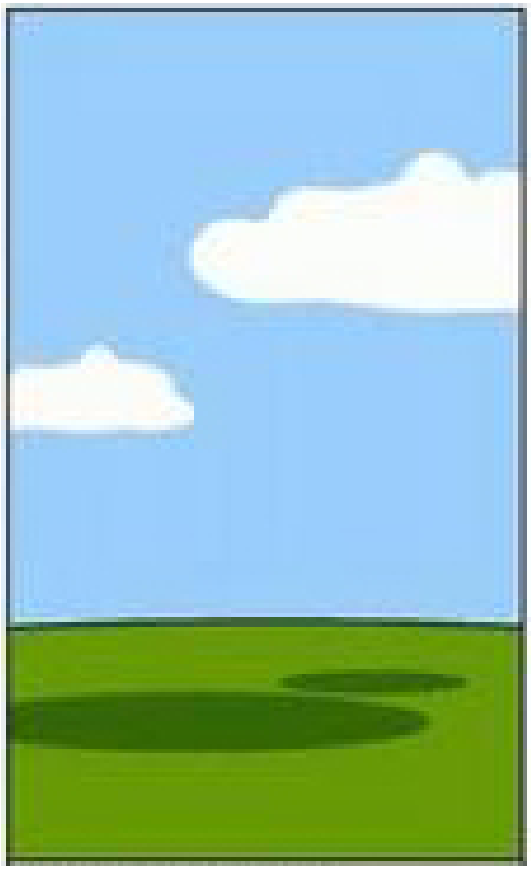


Comment le programmeur l'a écrit

Problèmes de développement

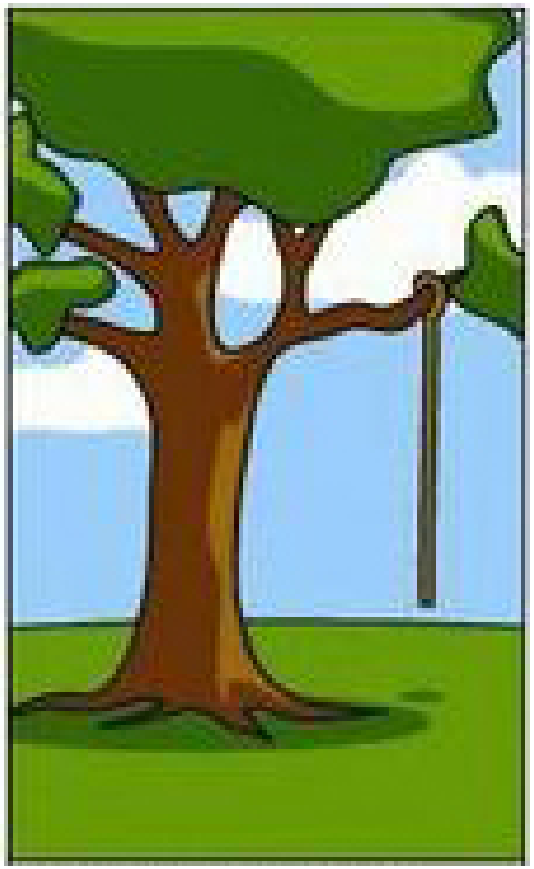
Comment le responsable des ventes l'a décrit

Problèmes de développement



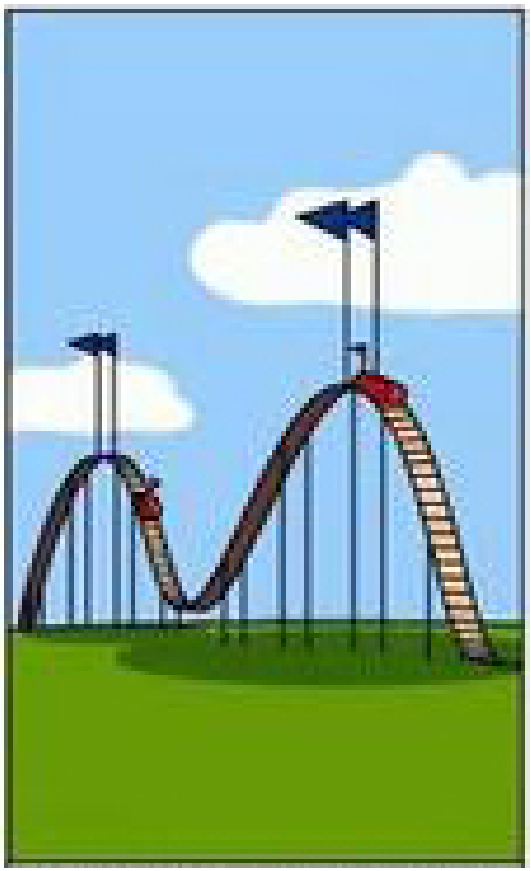
Comment le projet a été documenté

Problèmes de développement



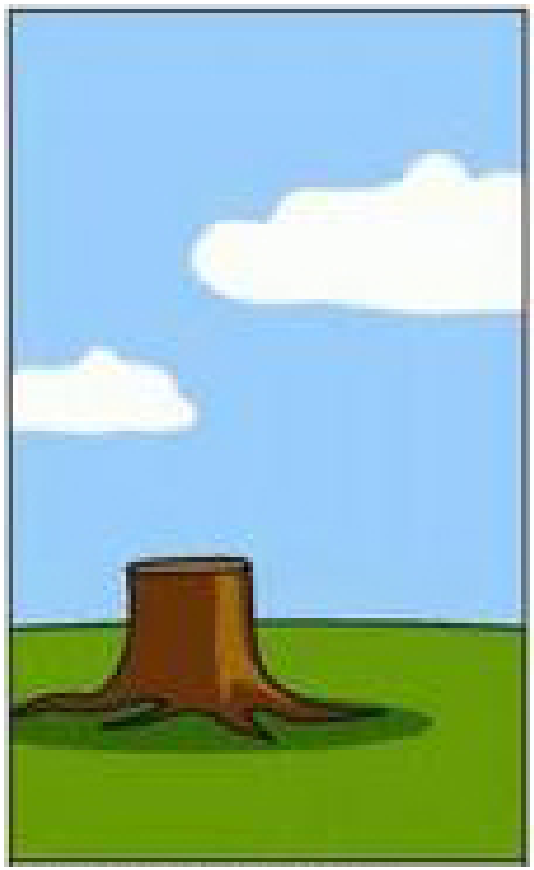
Ce qui a finalement été installé

Problèmes de développement



Comment le client a été facturé

Problèmes de développement

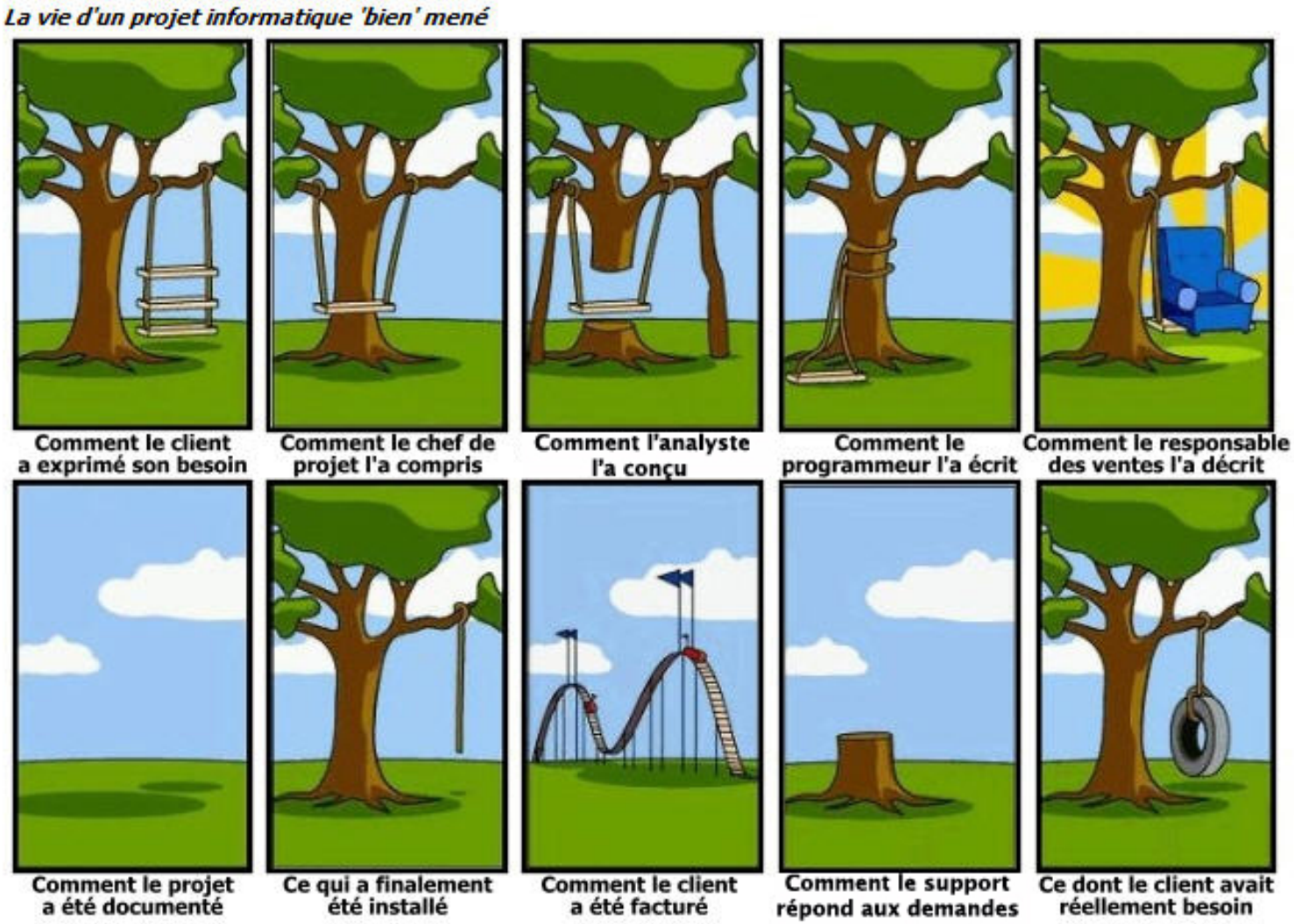


Comment le support répond aux demandes

Problèmes de développement

Ce dont le client avait réellement besoin

Problèmes de développement



La vie d'un projet informatique *bien* mené

Exemples d'erreurs réelles et célèbres

1. F16 (mauvaise analyse des besoins)

Exemples d'erreurs réelles et célèbres

1. F16 (mauvaise analyse des besoins)
2. Cour de Paris (1989)
   Erreur dans la transmission des offenses

Exemples d'erreurs réelles et célèbres

1. F16 (mauvaise analyse des besoins)
2. Cour de Paris (1989)
   Erreur dans la transmission des offenses
3. Bug de l'an 2000

Exemples d'erreurs réelles et célèbres

1. F16 (mauvaise analyse des besoins)
2. Cour de Paris (1989)
   Erreur dans la transmission des offenses
3. Bug de l'an 2000
4. USS Sheffield (Guerre des Falkland)

Exemples d'erreurs réelles et célèbres

1. F16 (mauvaise analyse des besoins)
2. Cour de Paris (1989)
   Erreur dans la transmission des offenses
3. Bug de l'an 2000
4. USS Sheffield (Guerre des Falkland)
5. Boeing 767 (logiciel trop efficace)

Plan de cours

Loi de Murphy

En fait c'est une série de lois et de lois connexes ou corollaires...

Loi de Murphy

En fait c'est une série de lois et de lois connexes ou corollaires...

1. Rien n'est aussi simple qu'il n'y paraît..

Loi de Murphy

En fait c'est une série de lois et de lois connexes ou corollaires...

1. Rien n'est aussi simple qu'il n'y paraît..
2. Tout prend plus de temps que vous ne le pensez.

Loi de Murphy

En fait c'est une série de lois et de lois connexes ou corollaires...

1. Rien n'est aussi simple qu'il n'y paraît..
2. Tout prend plus de temps que vous ne le pensez.
3. Tout ce qui peut aller mal ira mal.

Loi de Murphy

En fait c'est une série de lois et de lois connexes ou corollaires...

1. Rien n'est aussi simple qu'il n'y paraît..
2. Tout prend plus de temps que vous ne le pensez.
3. Tout ce qui peut aller mal ira mal.
4. S'il y a un risque pour que plusieurs choses aillent mal, c'est celle qui causera le plus de dommage qui ira mal.

Loi de Murphy

En fait c'est une série de lois et de lois connexes ou corollaires...

1. Rien n'est aussi simple qu'il n'y paraît..
2. Tout prend plus de temps que vous ne le pensez.
3. Tout ce qui peut aller mal ira mal.
4. S'il y a un risque pour que plusieurs choses aillent mal, c'est celle qui causera le plus de dommage qui ira mal.
5. Corollaire : s'il y a un pire moment pour que quelque chose aille mal, c'est à ce moment-là que cela ira mal.

Conclusion

1. Larry Flon
   "There is no programming language–no matter how structured–that will prevent programmers from making bad programs."
2. Brian Kernigan
   "Controlling complexity is the essence of computer programming."