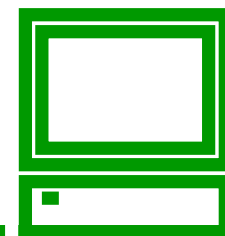
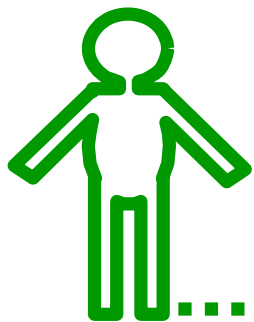


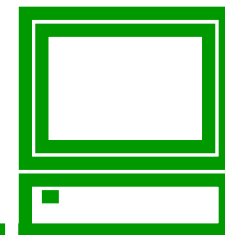
# 第3回 デザインの技法



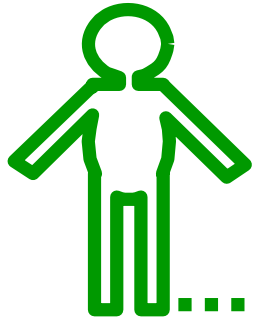
木村 朝子



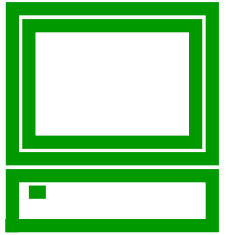
# 第3回の内容

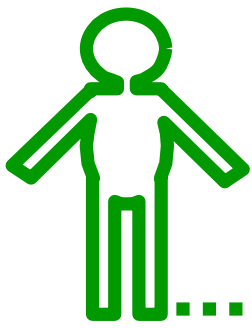


- ◆ デザインプロセスモデル
  - ◇ ウォーターフォールモデルとプロトタイピングモデル
- ◆ デザインの技法
  - ◇ ユーザ中心設計
  - ◇ プロトタイピング
- ◆ ユーザビリティテスト
  - ◇ ユーザビリティテストの目的
  - ◇ ユーザビリティテストの手法
  - ◇ ユーザビリティテストの流れ
  - ◇ テスト環境の整備
- ◆ プロトタイプ作成課題の発表

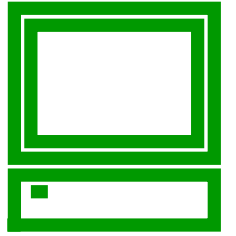


# デザインプロセスモデル





# デザインプロセスモデル



## ◆ プロセスは多様

- ◇ 設計対象の性質
- ◇ 設計目標の設定方法

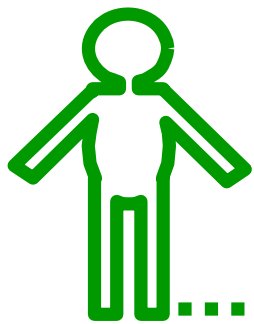
## ◆ 基本パターンがある(設計プロセスモデル)

### ◆ ウォーターフォールモデル

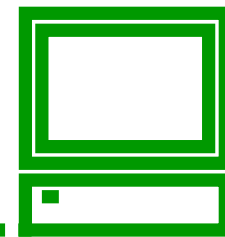
- ◇ 概念設計から始まり, 徐々に詳細を設計していく

### ◆ プロトタイピングモデル

- ◇ イメージに基づいてモデルを作り, 評価を行いながら  
現実に適合したモデルに作り変えていく



# ウォーターフォールモデル

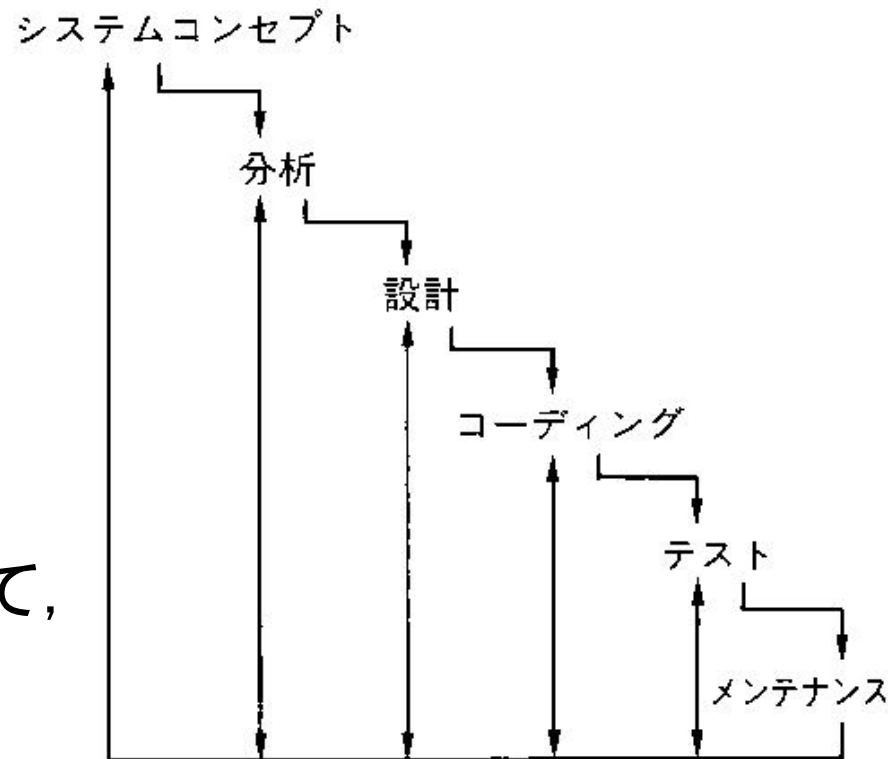


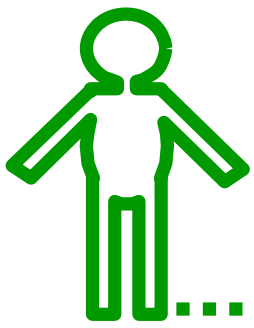
## ◆ 対象

- ◇ 大規模で複雑なシステム
- ◇ 設計者が直感・経験で設計を行うことが難しい

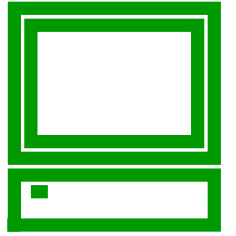
## ◆ 方法

- ◇ 各工程ごとに成果物を検証
- ◇ 検証の結果承認されたものだけが次の工程へ進む
- ◇ 原則この順序を飛び越したり、逆戻りしたりしない
- ◇ 滝の水が流れ落ちる様子に例えて、ウォーターフォール・モデル

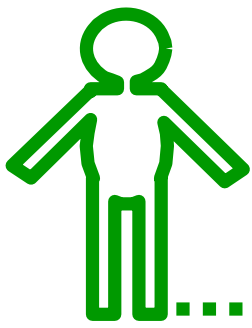




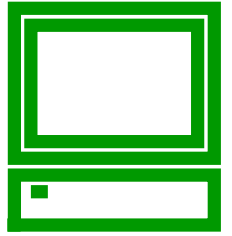
# ウォーターフォールモデル



- ◆ 最も基本的で一般的な開発モデル
- ◆ 利点
  - ◇ 設計を組織的に管理することが容易
- ◆ 欠点
  - ◇ 比較的初期に起こる設計ミスなどの問題を見つけにくい



# プロトタイピングモデル

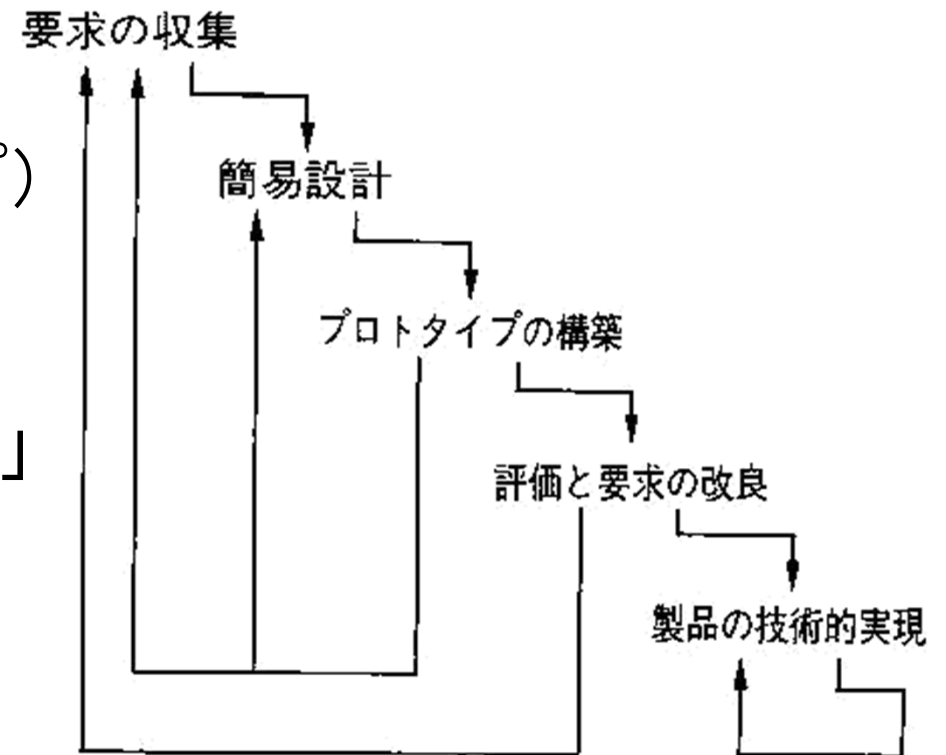


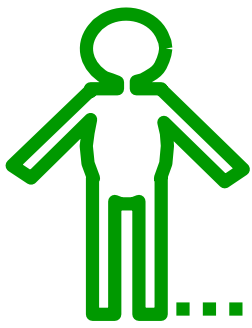
## ◆ 対象

- ◇ 定型的な方法がまだ開発されていない
- ◇ 予測しにくい要素がインタフェースの性能に大きく影響する

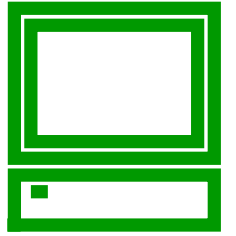
## ◆ 方法

- ◇ 早い段階で、試作品（プロトタイプ）を作成
- ◇ ユーザに試用してもらうことで「ユーザの要求と合っているか？」「使い勝手はどうか？」をチェックし、場合によってはプロトタイプを修正しながら、要求仕様を確定





# プロトタイピングモデル



## ◆ 利点

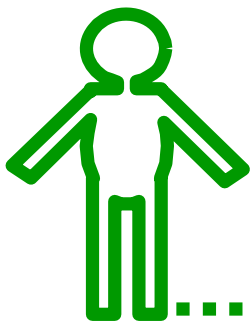
- ◇ 開発側とユーザ側の認識のズレを早期に解消
- ◇ 初期段階での誤りを発見⇒開発期間の短縮・コストの低減
- ◇ ユーザ側にとっては、実際にモノを見ることができるので、認識のズレをなくす非常に有効な方法

## ◆ 欠点

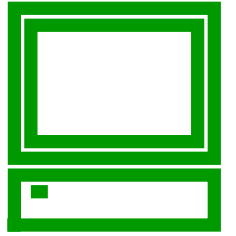
- ◇ 体系的に最適な設計解を求める努力を忘れる可能性

全体的にはウォーターフォールモデル  
局所的にはプロトタイピングモデル



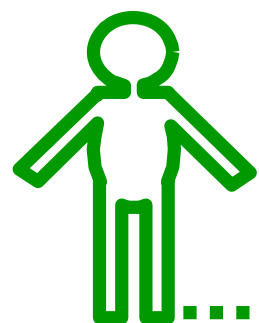


# ユーザ中心設計

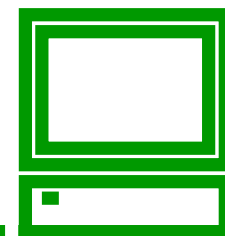


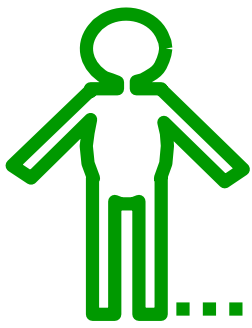
- ◆ ユーザの視点に立って設計する
- ◆ 基本プロセス
  1. ユーザの利用状況を把握する
  2. 利用状況からユーザニーズを探索する
  3. ユーザニーズを満たすような解決策を作る
  4. 解決案を評価する
  5. 評価結果をフィードバックして、解決案を改善する
  6. 評価と改善を繰り返す

プロトタイプ(試用品)の利用が効果的

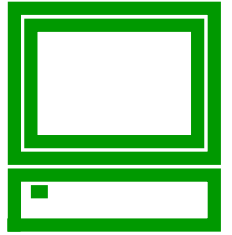


# プロトタイピング





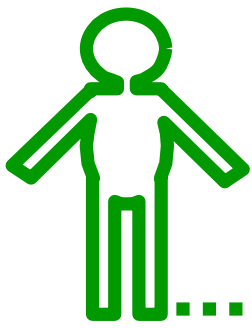
# プロトタイピングとは？



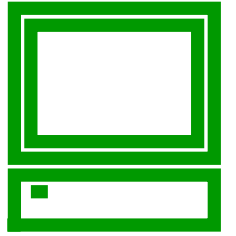
- ◆ 製品のモデルを早期に作成すること
  - ◇ ユーザに試しに使ってもらい、そのデータを元により良い製品を作成

- ◆ 目的
  - ◇ ユーザインタフェースの...
    - ▼ブレインストーミング
    - ▼設計
    - ▼作成
    - ▼テスト
    - ▼情報交換



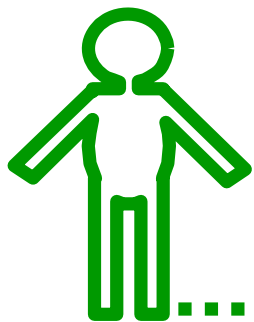


# プロトタイピングの利点

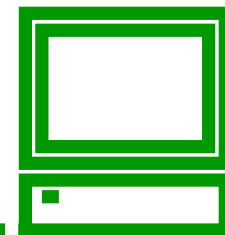


- ◆ 早期に目に見える形で実現することで、最終的にどんなシステムになるのか予想可能
- ◆ ユーザと製作者の間の意見交換
- ◆ ユーザに対してより高度な製品を提供可能
- ◆ 容易に変更可能
- ◆ 開発費用が抑えられる
- ◆ システム開発が加速

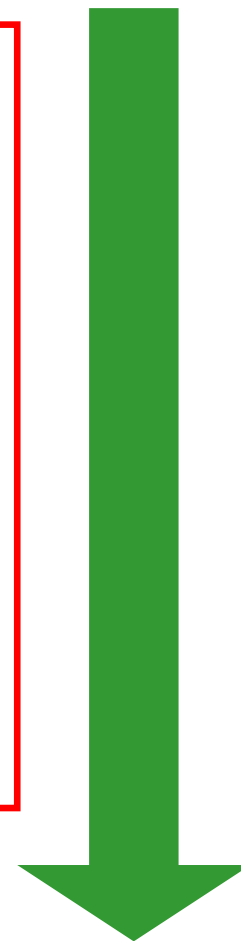


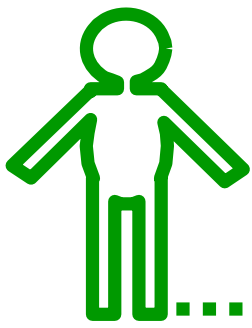


# プロトタイピングの進め方

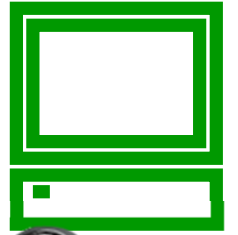


- ◆ 分析段階
- ◆ 設計・試作段階
  - ◇ ラフスケッチ段階
  - ◇ 詳細スケッチ段階
  - ◇ 手順デザイン段階
  - ◇ ハードウェアとソフトウェアの統合段階
- ◆ 評価段階





# 分析段階



## ◆ ユーザの特定

- ◇ 年齢や性別，職業，習熟度などの特性を明確化

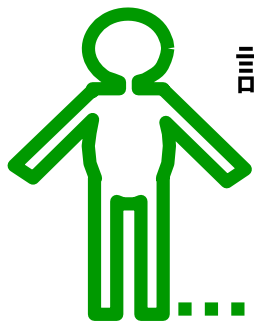


## ◆ 利用目的の特定

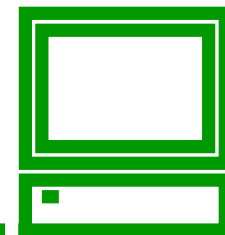
- ◇ その機器がなかった従来の生活場面での行動，そこに機器を導入することによる生活の変化などを検討

## ◆ 使用状況の特定

- ◇ 一種のシナリオを考える
- ◇ 類似機器が存在する場合，その問題点を徹底分析
- ◇ モニタ調査によりユーザ意見を吸い上げる



# ラフスケッチ段階(1)



## ◆ ラフスケッチ

- ◇ 全体形状の特徴的な部分を強調し、細部を省略した図面を作成



- ◇ 機器についてデザイン面から検討

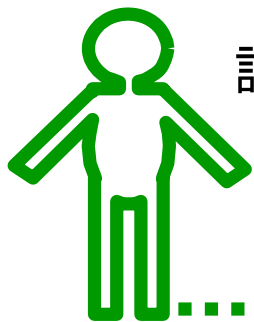
- ▼ 全体をどのようなメタファ(metaphor)で表現するか

- ▽ メタファ: 機能を何か別のものに例えて表現すること

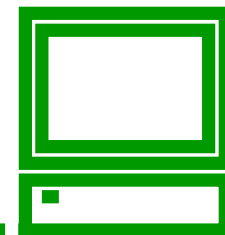
- » 例: PCの「デスクトップ(机上)」「フォルダ」

- ▼ 機能をどのように重点化して外観を表現するか

- ▼ 大きさと形状はどうするか



# ラフスケッチ段階(2)

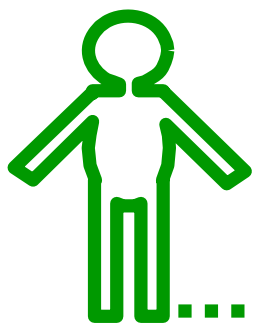


## ◆ ハードウェアの場合

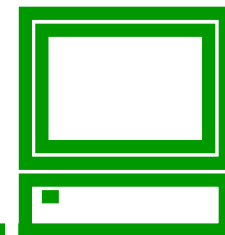
- ◇ ラフモックアップ(=3次元立体モデル)を作成
- ◇ 携帯性や取り扱い性について大まかな検討を行う
  - ▼ スチロール製ボードの組合せ
  - ▼ キーボードの場所にキー配置図,  
画面表示部にサンプルの図面を貼り付ける
  - ▼ 重量感や重心位置は内部に粘土を詰め込んで検討
  - ▼ 小型機器の場合は手の収まり具合も検討







# クレイモデル



日産

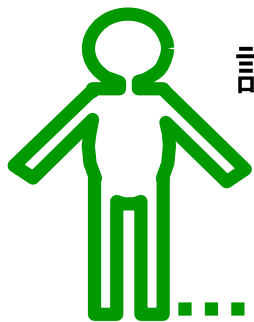


フォルクスワーゲン

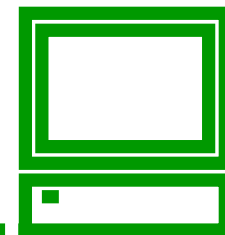


ポルシェ





# ラフスケッチ段階(3)



## ◆ ソフトウェアの場合

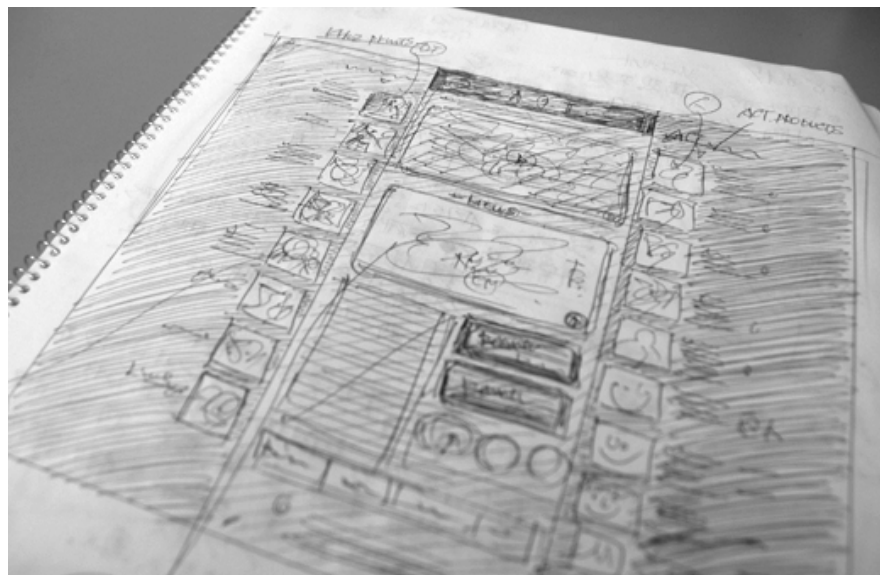
### ◇ 画面のラフスケッチの作成

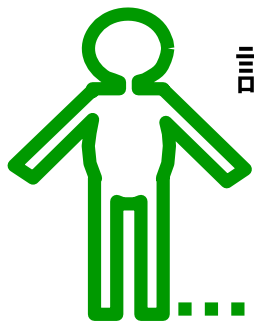
#### ▼ 情報機器の場合

⇒ 起動画面, メインメニュー, 主要作業画面など

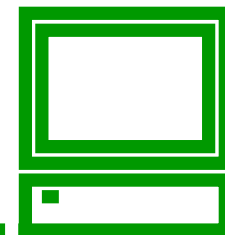
#### ▼ 使用する画面部品 (widget) も検討

▽ メニュー, アイコン, 対話ボックスなど





# 詳細スケッチ段階



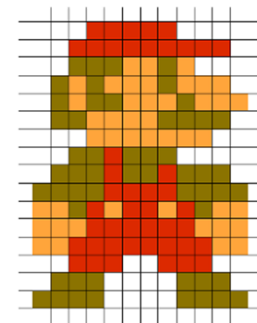
## ◆ ハードウェアの場合

- ◇ 詳細**モックアップ** (detailed mockup) の作成
  - ▼ 本物と同じ素材を使って模型作成
  - ▼ コネクタや個々のスイッチなどの細部も描き込む



## ◆ ソフトウェアの場合

- ◇ 画面デザイン
  - ▼ 画面のピクセルサイズを考慮
  - ▼ 与えられたピクセルの中で、何ドットを使って形状が表現できるのか検討

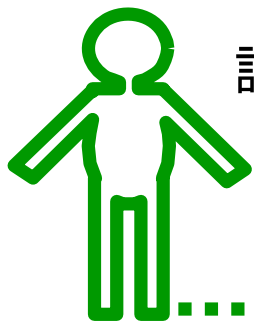


# Virtual Handcrafting

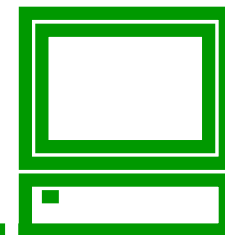


# MAI Painting Expert（実物への描画）



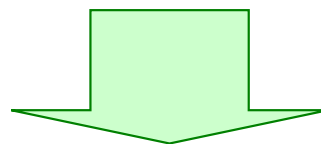


# 手順デザイン段階



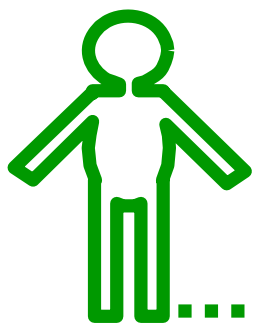
## ◆ ストーリーボードを構成して検討

当該機器の典型的な利用手順に関して、  
全ての画面を作成しておき、  
それを手順通りに配置する

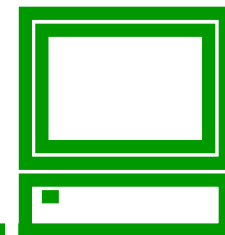


手順の複雑なところがないか、  
自然な手順で操作できるかなどを検討

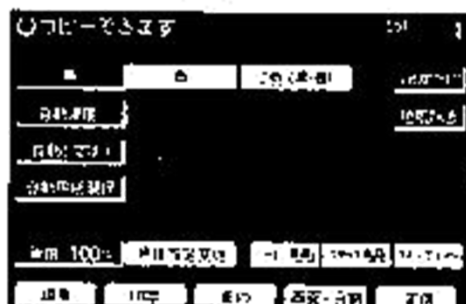




# ストーリーボードの例



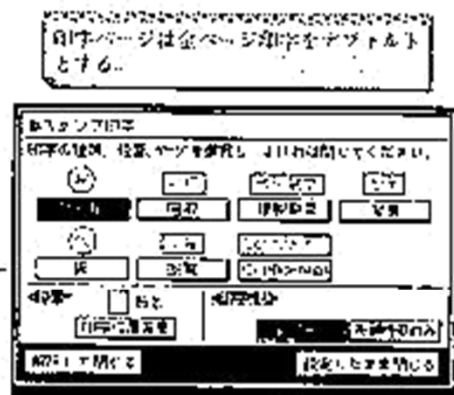
レディ画面



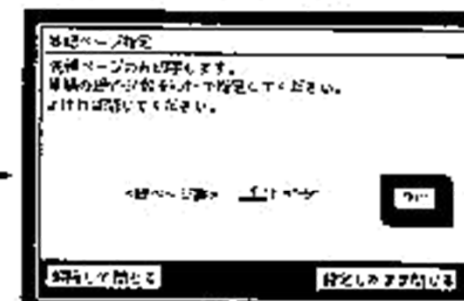
印字を押す



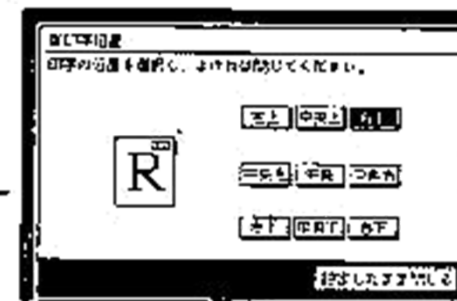
スタンプ印字  
を押す

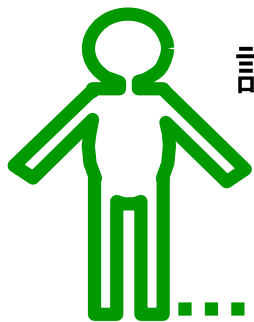


先頭ページのみを押す

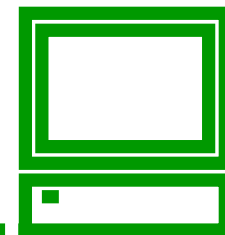


印字位置変更を押す



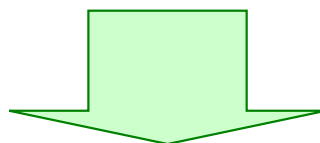


## ハードウェアとソフトウェアの統合段階



### ◆ 機能モックアップ (function mockup) の作成

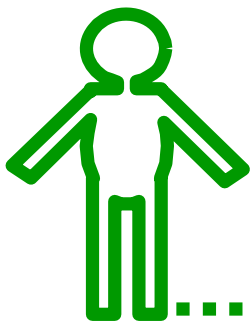
設計がある程度の段階に達してからは、  
ハードウェアとソフトウェアを  
組み合わせた上でインタフェースを検討



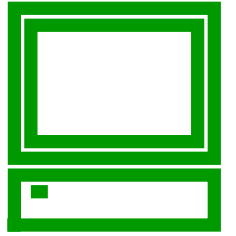
動作するモックアップ ⇒ **機能モックアップ**

入力部: 本物のキースイッチをはめ込む  
表示部: 外部の制御用パソコンにつないだ  
ディスプレイを埋め込んで代用





# プロトタイプ<sup>o</sup>の4つの次元



## ◆「幅」

- ◇ 製品の全機能のうち、どの程度がプロトタイプに組み込まれているか  
⇒ どんなプロトタイプでも課題をカバーするに足る幅が必要

## ◆「深さ」

- ◇ どの程度まで「実際に動作する機能」となっているか

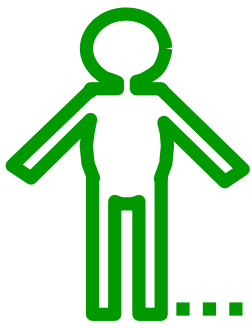
## ◆「見栄え」

- ◇ 目的の外観(フォント, 色, グラフィックなど)を正確に表現しているか

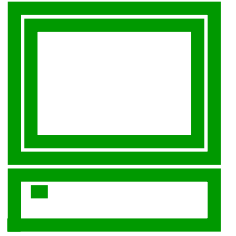
## ◆「インタラクション」

- ◇ 入出力の手法がリアルにシミュレートされているか

プロトタイプによって優先順位が異なる



# オズの魔法使いテスト



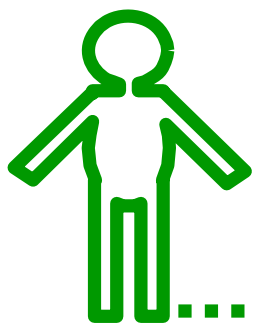
## ◆ オズの魔法使い

- ◇ 恐ろしい姿をした「オズの大王」の正体は、  
実はカーテン裏で老人が操っている機械だった

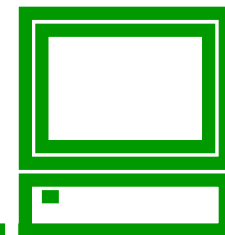
ユーザとマシンの仲立ちとして人間を使う

## ◆ プロトタイピングでは...

- ◇ ユーザがコマンドをコンピュータ役の人間に伝えると、  
コンピュータ役は言われた通り操作
- ◇ プロトタイプではシミュレートできないことを  
ユーザに見せる場合、実験者がコンピュータに入力し、  
得られた出力を示す



# オズの魔法使いテスト



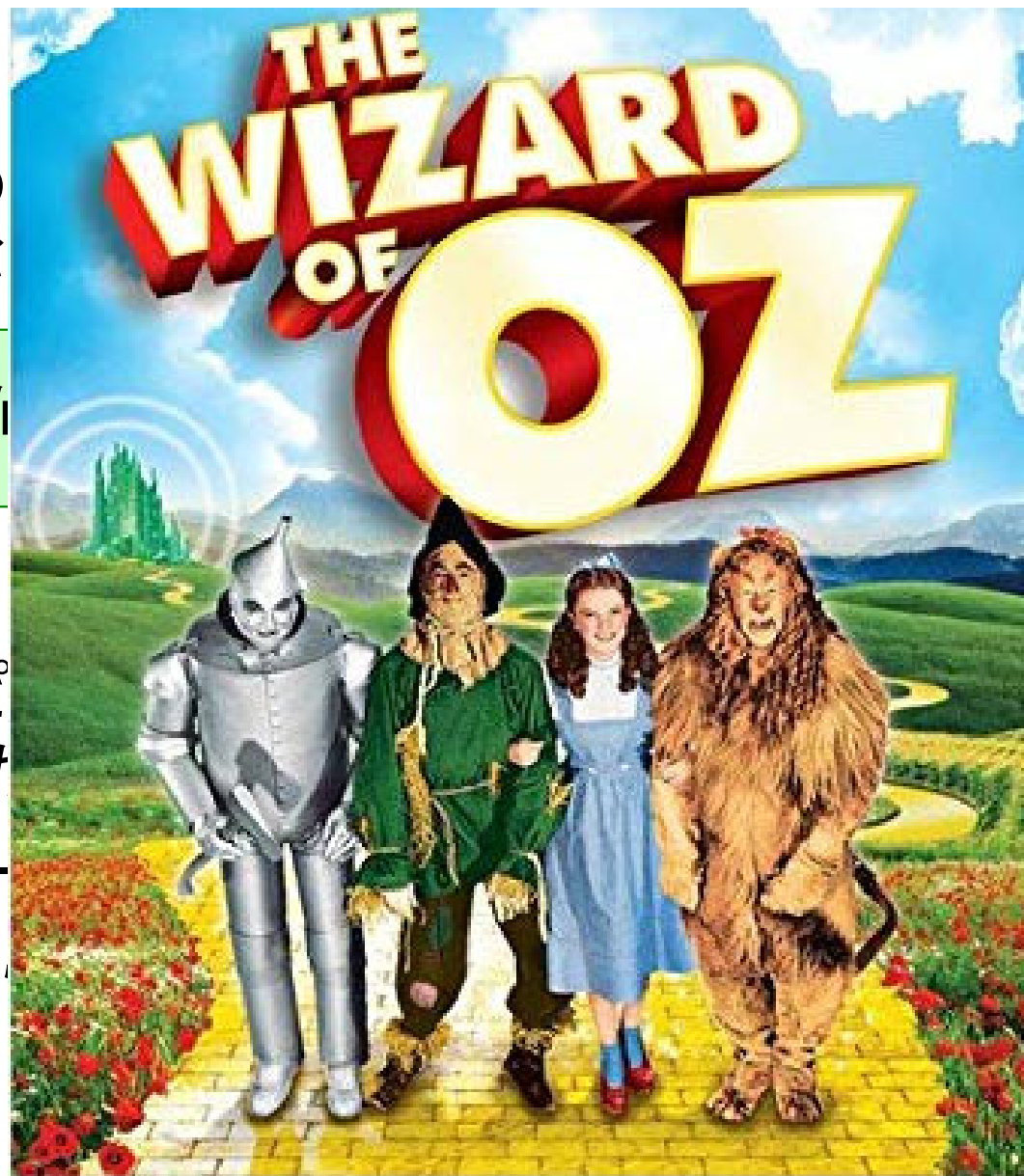
## ◆ オズの魔法使い

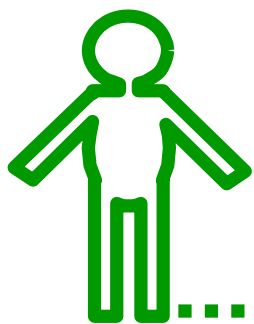
- ◇ 恐ろしい姿をした「オズの実はカーテン裏で老人が

ユーザとマシンの仲

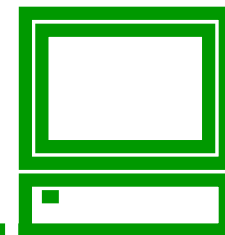
## ◆ プロトタイピングでは...

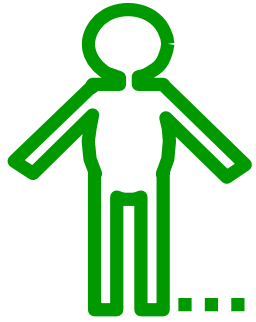
- ◇ ユーザがコマンドをコンピュータ役は言われた
- ◇ プロトタイプではシミュレーションユーザに見せる場合、実際に得られた出力を示す



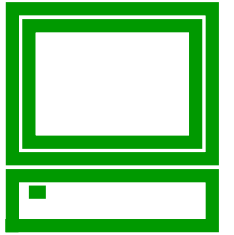


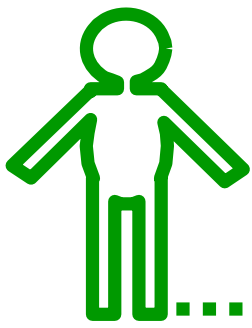
# オズの魔法使い



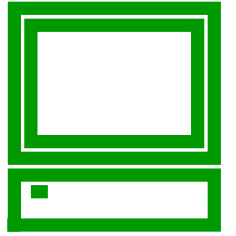


# ユーザビリティテスト



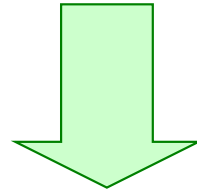


# ユーザビリティテスト



- ◆ プロトタイピングにおいてモデルを作成

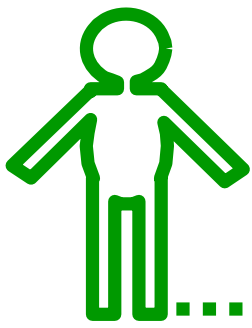
ユーザに使用させ  
ユーザビリティを確認



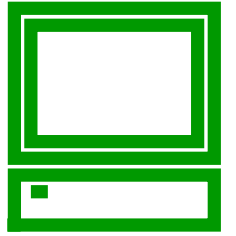
ユーザビリティテスト

- ◆ ユーザが実際に行なった行動や感想から、ユーザビリティの改善を行なう





# ユーザビリティテストの目的



## 1. 問題発見

- ◇ 製品やシステムを利用した際に、ユーザビリティ上の問題がないか確認

## 2. 案の選択, 競合比較

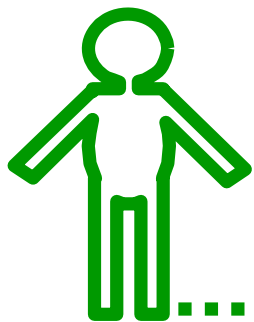
- ◇ 複数のUIデザイン案があった際に最も優れたものを判断
- ◇ 競合他社の製品と比較し, 長所・短所を具体化

## 3. 原因究明

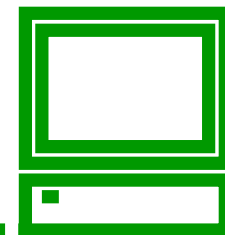
- ◇ 販売後の問い合わせ・クレームに対して, 効果的な改良をするために問題の原因を究明

## 4. 水準測定

- ◇ 製品のユーザビリティ水準の測定



# ユーザビリティテストの手法



## ◆ パフォーマンス(効率)評価

◇ 「使いやすさ」(作業速度, エラー率)を評価

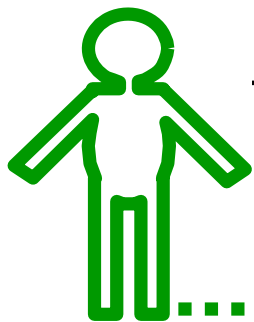
## ◆ 主観評価

◇ 「印象」(安心して, 気分良く, 好感を持っていたか)を評価

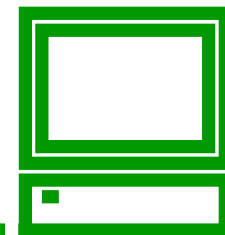
## ◆ インタラクション評価

◇ 「分かりやすさ」(つまずきが少なく, スムーズにできるかなど)を評価



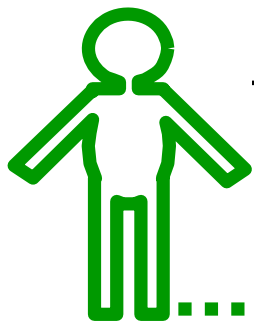


# パフォーマンス評価

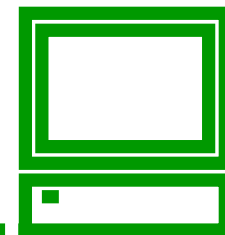


- ◆ 時間計測, エラー数などの事象を計測
- ◆ 複数の評価対象において順位データを取得可能
  
- ◆ 利点
  - ◇ 実施が容易
  - ◇ 定量的な分析が可能
- ◆ 欠点
  - ◇ 結果の適応範囲が小さい
  - ◇ 問題点の指摘が困難
  - ◇ 比較対象が必要



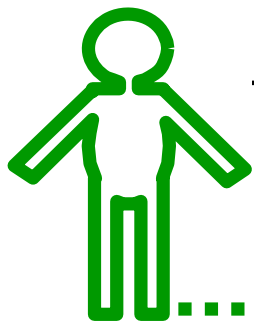


# 主観評価

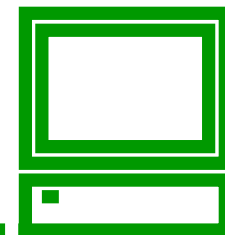


- ◆ 主観的な印象, 感じなどの主観データを質問紙などで採取
- ◆ 利点
  - ◇ 幅広い評価が可能
  - ◇ 多数のユーザの傾向を把握するのに適切
- ◆ 欠点
  - ◇ ユーザの記憶に依存
  - ◇ 問題点の指摘が困難





# インタラクション評価



- ◆ タスクを実行している被験者の行動を観測
- ◆ 被験者は4～6人以上用意

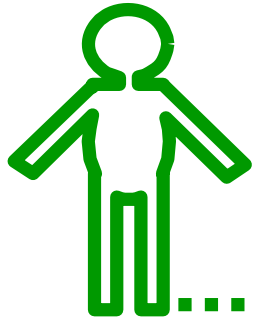
## ◆ 利点

- ◇ 幅広い評価が可能
- ◇ 問題点の指摘が容易
- ◇ 結果が記憶に依存しない

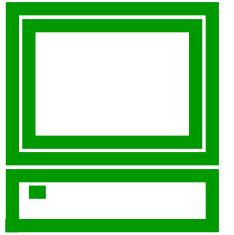
## ◆ 欠点

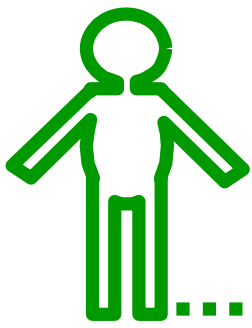
- ◇ ユーザ解析の工数が多い
- ◇ 評価者のスキルに依存



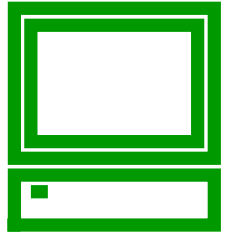


# ユーザビリティテストの流れ

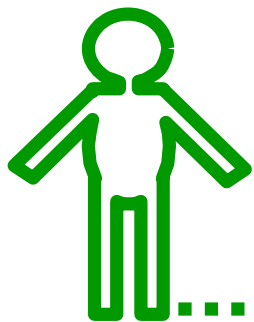




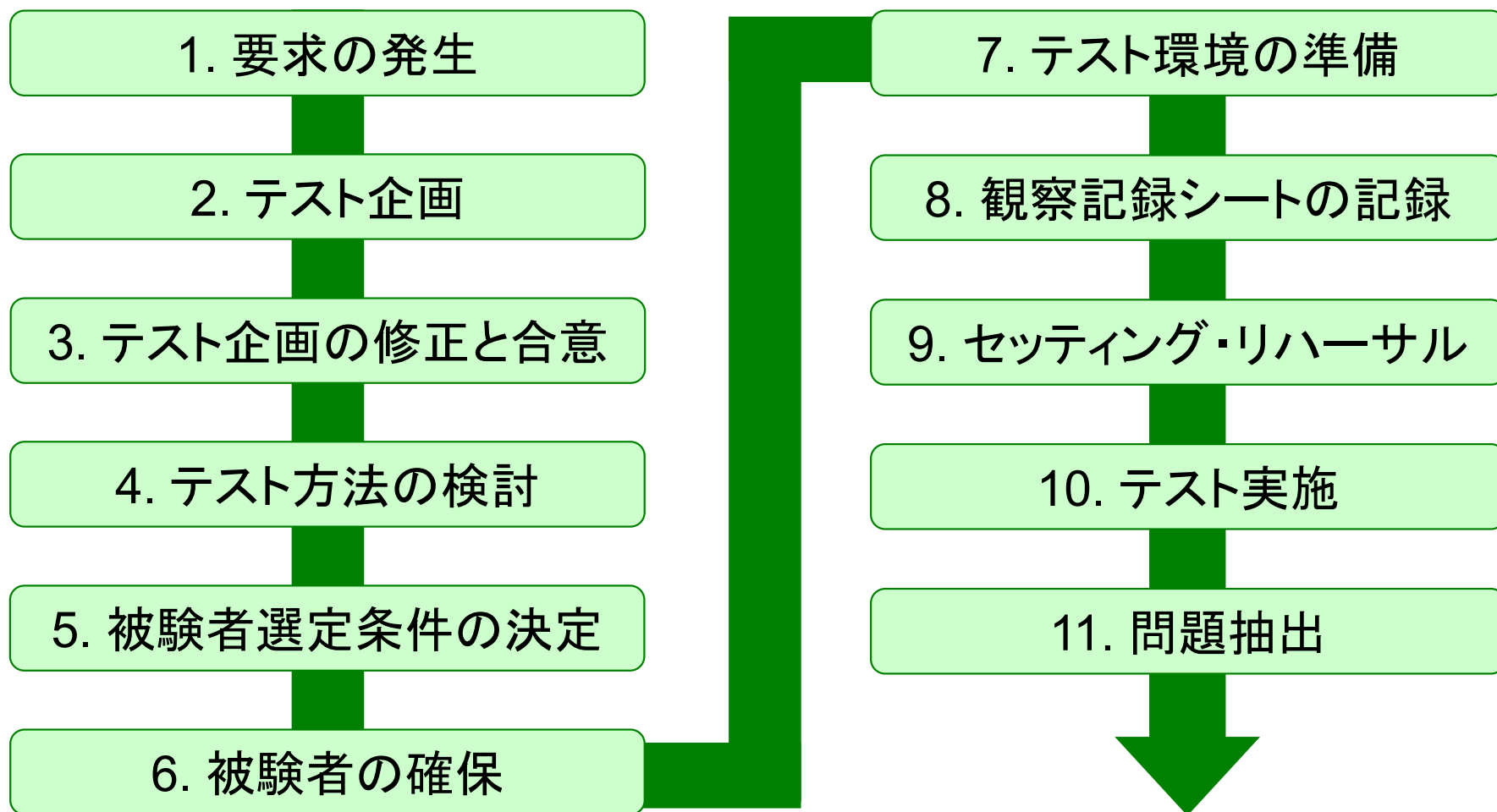
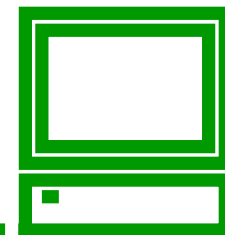
# テストに関わる人

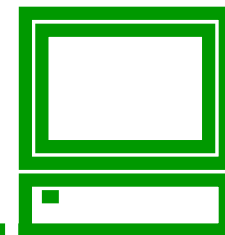
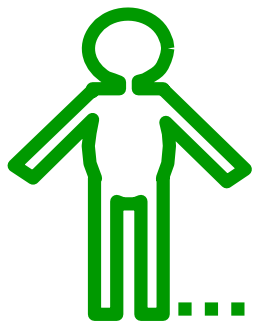


- ◆ 評価者
  - ◇ テストの内容を具体的に考える
- ◆ 進行者
  - ◇ テスト中に、被験者に指示を行うなどテストを進行させる
- ◆ 観察者
  - ◇ テスト中に、第3者としてテスト外から観察を行う
- ◆ 被験者
  - ◇ テスト中に、プロトタイプの実操作を行う



# テスト実施の流れ

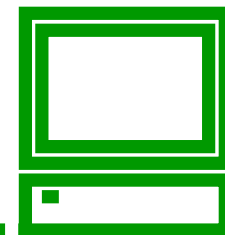
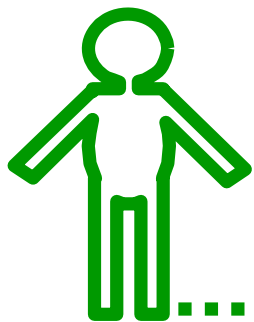




# 1. 要求の発生

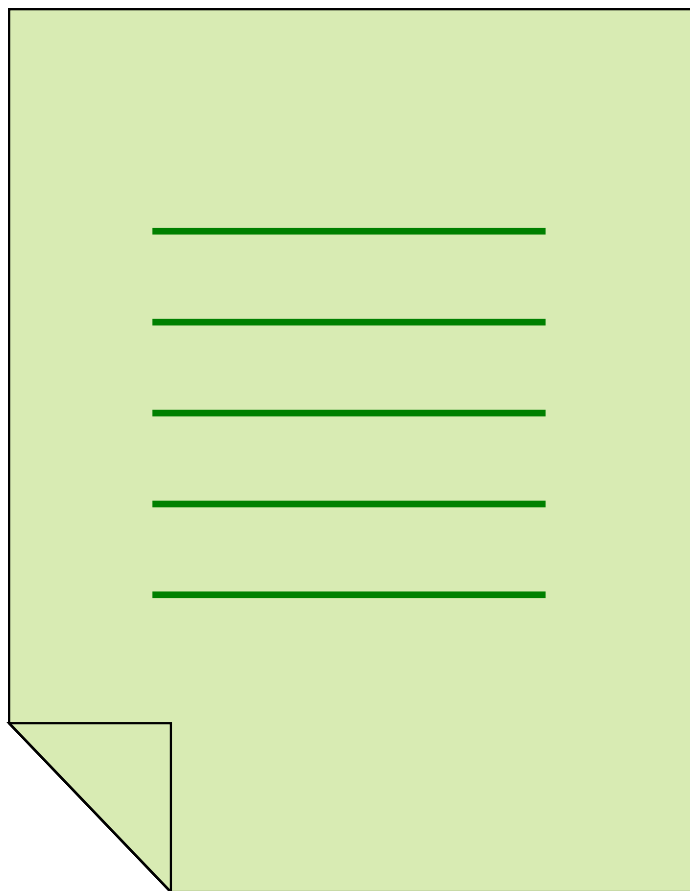
- ◆ 問題発見の要求
  - ◇ 開発段階で最大限に商品をよくしたい
- ◆ ユーザインタフェース案選択の要求
  - ◇ どの案が一番良いか、部分的に他の案のどこを取り入れるべきか検証したい
- ◆ 問題の原因究明の要求
  - ◇ 何故問い合わせが生まれるのかなど、問題の原因を正しくつかみ、的確な改良に結びつけたい

ユーザビリティテストの要求が生まれる



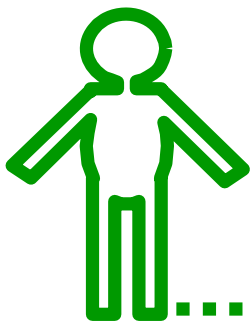
## 2. テスト企画

- ◆ テスト要求に基づき、基本的なユーザビリティ評価の枠組みを決め、企画書に

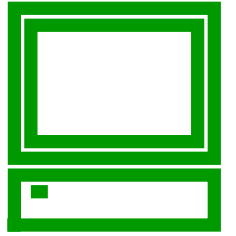


- ◆ 目的(要求)
- ◆ 評価対象
- ◆ 目標(テストで知りたいこと)
- ◆ テスト環境
- ◆ 評価方法
- ◆ タスク
- ◆ 被験者
- ◆ スケジュール(被験者1人分)
- ◆ スケジュール(全体)

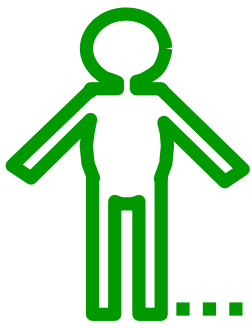




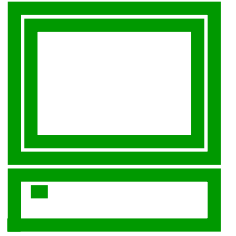
## 4. テスト方法の検討(1)



1. タスクの詳細決定
  - ◇ テスト目標を達成するタスクの詳細を決定
  - ◇ タスクの開始状態とゴールを具体的に設定
2. タスク順序の決定
3. タスク指示書の作成
  - ◇ 被験者によらない, 公平なテスト結果を得るためにタスクの指示内容を文書化



## 4. テスト方法の検討(2)



### 4. タスク時間配分の確認

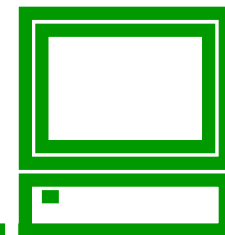
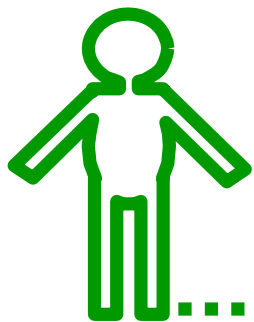
- ◇ タスクが一定の時間内に収まるよう時間を配分

### 5. 教示内容の決定

- ◇ タスク実施前に最小限の必要な知識を被験者に教示
  - ▼ タスク関連情報: 製品の機能, 製品を使うことになった経緯, 環境条件
  - ▼ テストの前提条件: 協力してもらうための目的

### 6. 助言計画

- ◇ つまずきそうな箇所でどのように助言するか



## 5. 被験者選定条件の決定

- ◆ テスト目的に合わせて被験者の選定条件を決める



評価者

- ◆ 対象製品類の仕様経験

- ◆ コンピュータのスキル

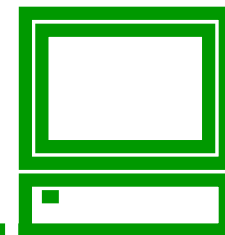
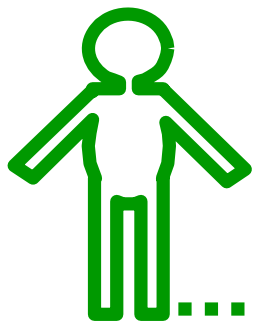
- ◆ 年齢

- ◆ 性別



被験者

MUST条件とWANT条件を決める

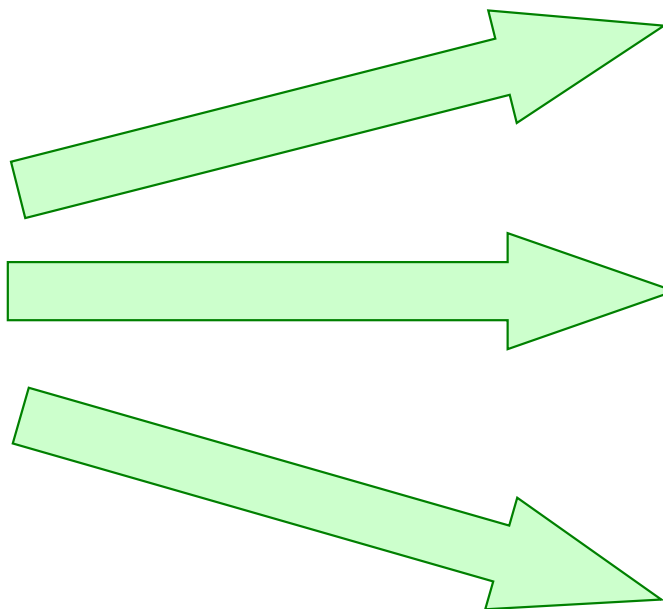


## 6. 被験者確保

「何日のどの時間帯に協力できるか」

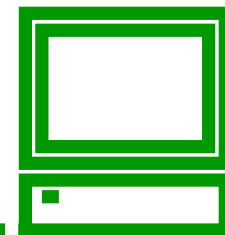
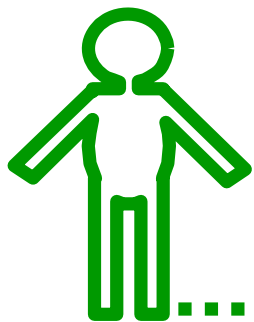


評価者



被験者

複数の応募者からテスト目的に合わせて、  
最適の組み合わせになるよう選定



## 7. テスト環境の準備

- ◆ 被験者が実際に使用する評価対象物の諸条件を決めて調達, 準備

- ◆ 使用するコンピュータ

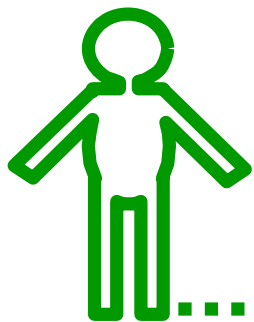
- ◇ CPUの性能
- ◇ OS
- ◇ メモリ容量
- ◇ ノート型／卓上型
- ◇ モニタの表示能力
- ◇ 使用キーボード

- ◆ 通信の条件

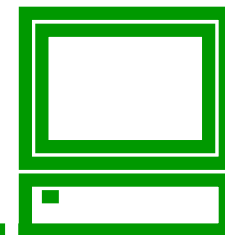
- ◇ 通信回線の種類
- ◇ スタンドアローン／LAN接続

- ◆ テスト室の条件

- ◇ 照明
- ◇ BGM
- ◇ 使用するテーブル, 椅子



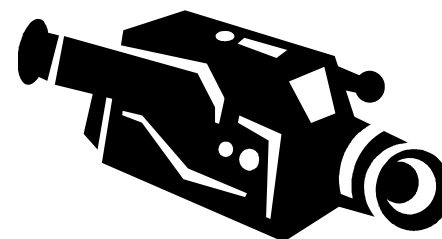
## 8. 観察記録シートの準備



### ◆ 実験中の被験者の観察記録

#### ◇ 基本はビデオ撮影

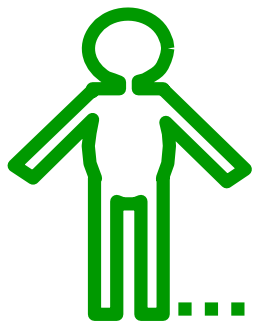
ビデオだけではテスト後の  
分析作業が大変



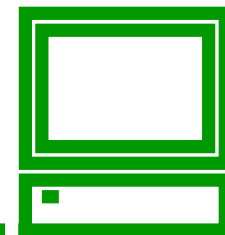
ビデオとは別に被験者の行動を記録

リアルタイムで記録しやすいように  
観察記録シートを用意する



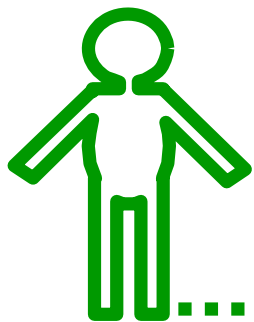


## 9. セッティング・リハーサル

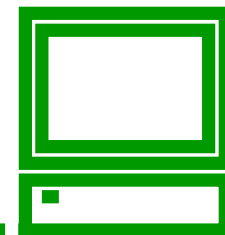


- ◆ テスト室と観察室の条件をテスト当日とできる限り同条件にし, リハーサルを行う

必要があれば  
テストの実施方法の一部を修正



## 10. テスト実施 – テストの準備 –



### ◆ 実験設備の最終確認

- ◇ 撮影機材, マイク音声, ビデオテープなど確認

### ◆ 被験者への事前説明

- ◇ テストの趣旨

- ▼ 何のためのテストか

- ▼ どのくらい時間がかかるか

- ▼ おおよそどのようなことをしてもらうのか

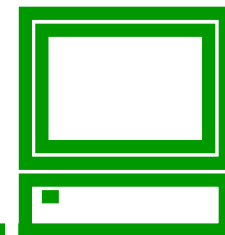
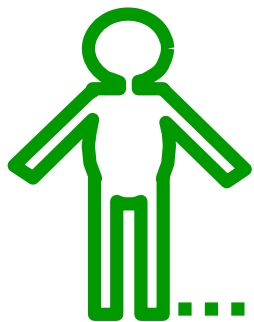
- ◇ 個人データ保護

- ▼ 実名, ビデオなど, 無断で外部に出さないことを確認

- ◇ 開発中の製品を評価する場合

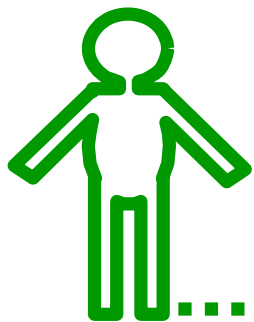
- ▼ テスト終了後, 製品について口外しないよう伝える





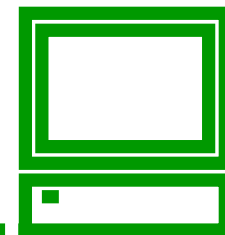
## 10. テスト実施 – テスト –

- ◆ タスクを与え, テストを実施
  - ◇ タスクは, 予め用意した指示シートを提示
    - ▼ 説明のばらつき, 誤解, 評価への影響を防ぐ
  - ◇ タスク終了の判断は被験者に
    - ▼ 操作の途中である可能性もある
  - ◇ 進行中は必要に応じて助言, 質問



- テスト -

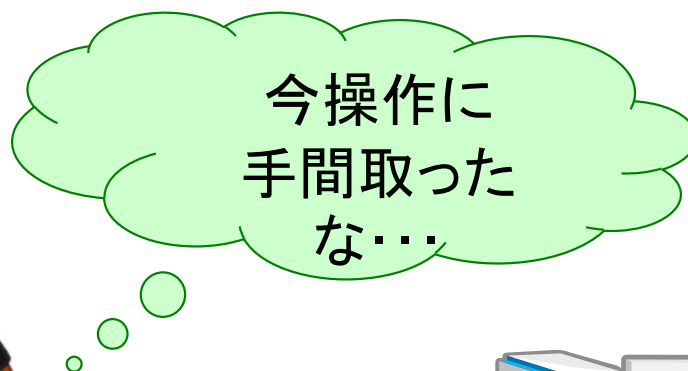
# 観察メモへの記入



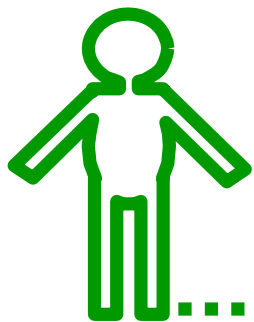
- ◆ タスクの開始時間，終了時間，各手順で起きたこと，被験者の発言など
  - ◇ 観察メモだけから分析できるように



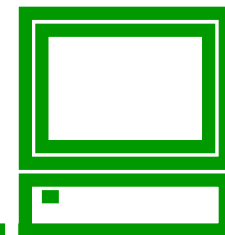
進行者



被験者



## 10. テスト実施 – テスト実施後 –



- ◆ 事後インタビュー
  - ◇ 全体の感想などを尋ねる

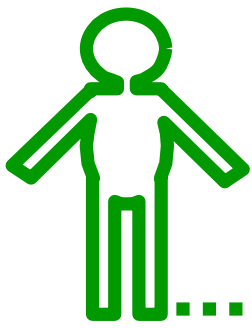


進行者

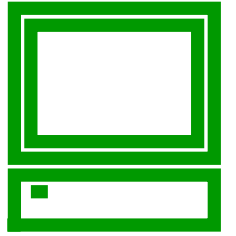
この商品は使ってみて  
どうでしたか



被験者



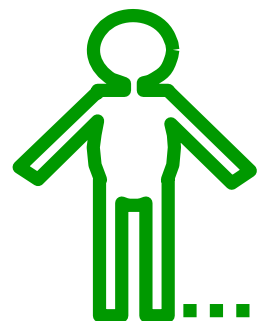
# 11. 問題抽出



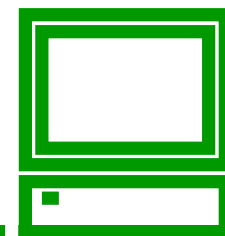
- ◆ 記録から問題と思われる点を抽出
  - ◇ どのタスクで発生したか
  - ◇ どの工程で発生したか
  - ◇ どの被験者で発生したか
  - ◇ 問題の重要度

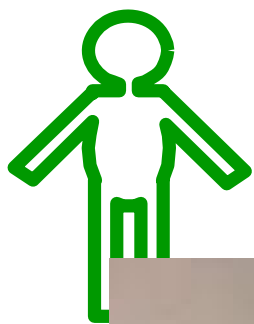
抽出後

- ◆ 問題の整理統合, 重要度の決定



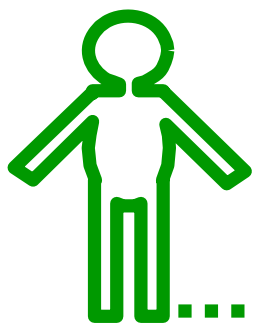
# テスト環境の整備





# ユーザビリティラボの設備





# テストイン

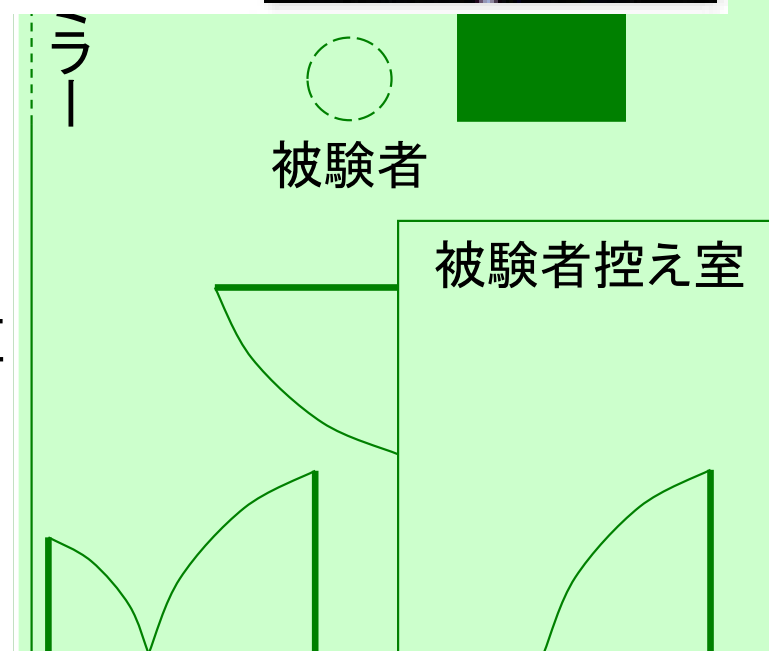
## ◆ 被験者

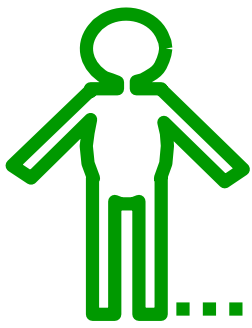
◇ 被験者の操作, 表



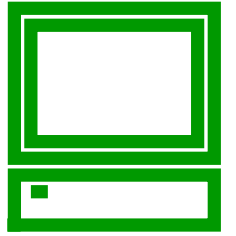
## ◆ 進行者

- ◇ 被験者の視界に入らない
- ◇ 被験者の様子が見え, つぶやきが聞こえる
- ◇ 被験者に自然に助言を行える位置





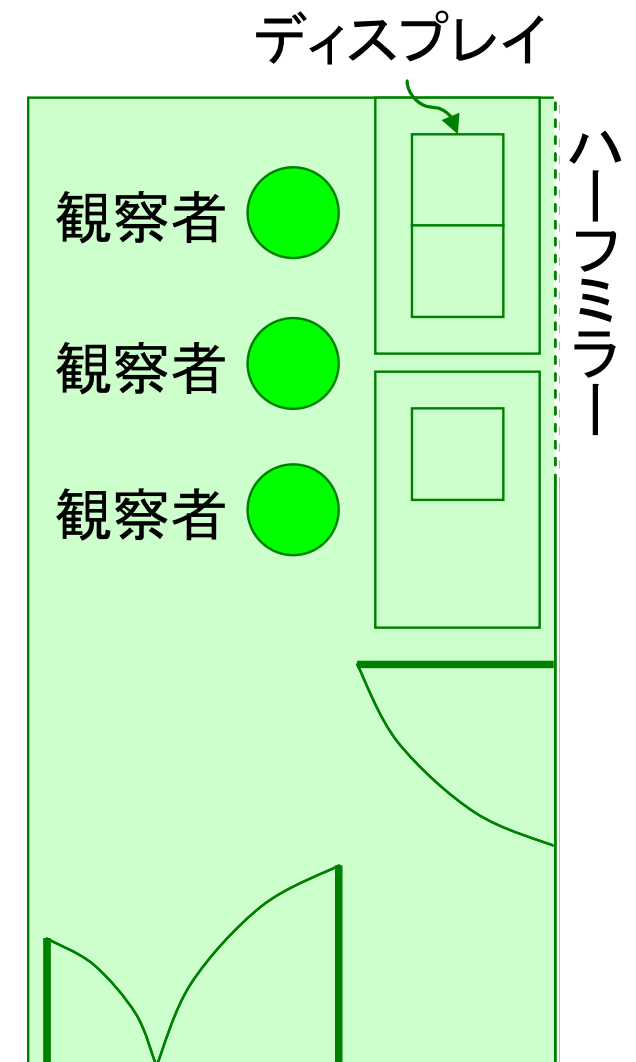
# モニタールームの設備



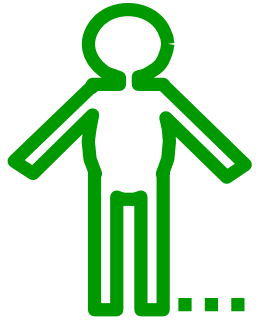
- ◆ ハーフミラーの前にテーブルを配置
  - ◇ 観察中にメモをとる

- ◆ 被験者の映像をディスプレイで確認
- ◆ 防音

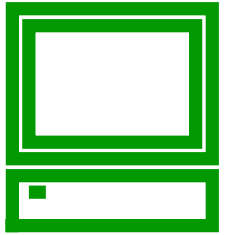
被験者に観察者が分からないように

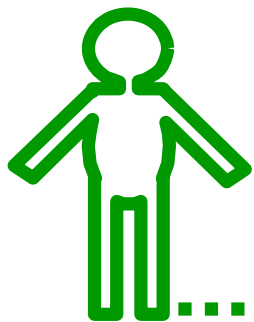




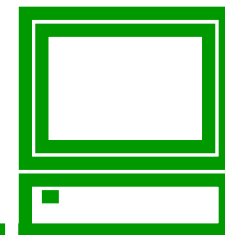


事例





# 事例1 クリーナ



## ◆ 評価の目的

- ◇ ユーザが身体的に楽に掃除作業できるようなユーザビリティが求められる

- ▼ 設計段階での仕様決定のために評価を行う

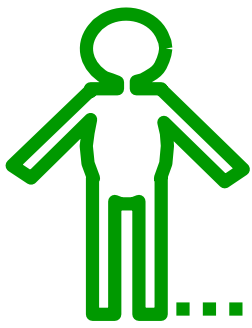
## ◆ 評価方法

- ◇ 数種類の仕様が異なるプロトタイプ(試作品)に対し、身体的な使いやすさを評価

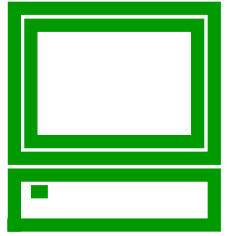
- ▼ 延長管とホースの太さ, ホースの長さ, ヘッド重量, グリップ形状

- ◇ 筋電図, 主観評価, パフォーマンスを計測





# 事例1 クリーナ



## ◆ タスク: 4種類の作業を設定

### ◇ a. 床面ブラシ往復タスク

▼ 通常の床面掃除を想定し、各仕様の相違が及ぼす影響を評価

### ◇ b. ヘッド持ち上げタスク1

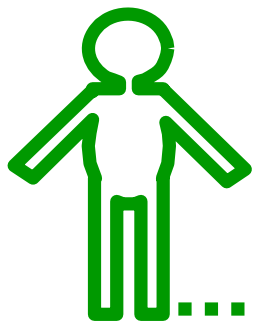
▼ 階段の掃除を想定し、ヘッドの重さ、ホースの長さ、グリップ形状が操作性に及ぼす影響を評価

### ◇ c. ヘッド持ち上げタスク2

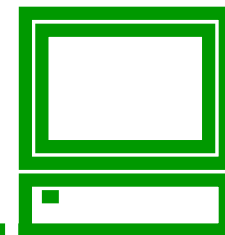
▼ 天井に近い場所の掃除を想定し、ヘッドの重さ、ホースの長さ、グリップ形状が操作性に及ぼす影響を評価

### ◇ d. 一定面積の掃除タスク

▼ ホースの長さが掃除作業に及ぼす影響を評価



# 事例1 クリーナ



## ◆ タスク: 4種類の作業を設定

### ◇ a. 床面ブラシ往復タスク

▼ 通常の床面掃除を想定し、各仕様の相違が及ぼす影響を評価

### ◇ b. ヘッド持ち上げタスク

◆ タスク (a)～(c) は筋電図計測及び主観評価

◆ タスク (d) はタスク達成悲観やストローク数を  
計測するパフォーマンス評価

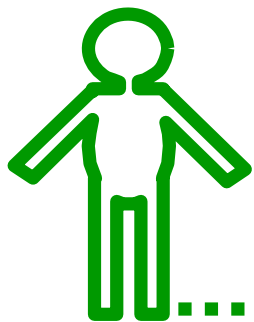
### ◇ c.

◆ ビデオ撮影を行い動作観察

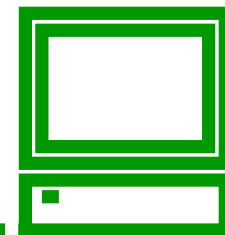
長さ、  
グリップ形状が操作性に及ぼす影響を評価

### ◇ d. 一定面積の掃除タスク

▼ ホースの長さが掃除作業に及ぼす影響を評価



# 事例1 クリーナ



## ◆ 評価結果

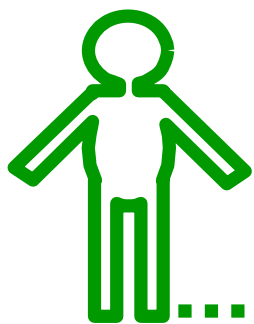
### ◇ タスク (d) の結果

▼ 平均10cm程度のストローク長の違いがホースの長さの違いによって現れている

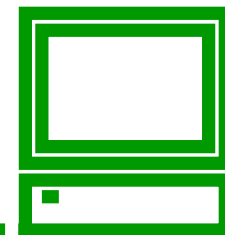
		ストローク長 (cm)	
		900mm	1200mm
被験者	A	65	80
	B	80	100
	C	65	75
	D	55	55
	E	80	90
	F	55	65

### ◇ タスク (a)～(c) の筋電図と主観評価の結果

▼ ヘッド重量, 延長管の太さがユーザビリティに影響を与えている



## 事例2 ATM



### ◆ 目的

- ◇ 高齢者対策は身体的特性に限られがちだが，認知的特性についても検討する必要がある

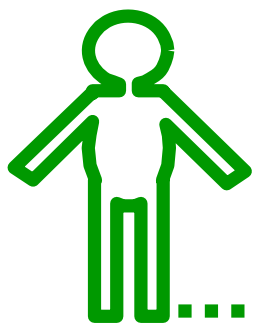
### ◆ 被験者

- ◇ 60歳以上の高齢者
- ◇ 大学生
- ◇ インスタント高齢者用装置を装着した大学生

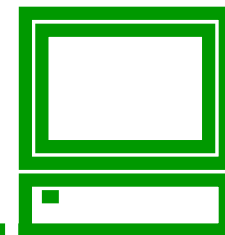
### ◆ 評価方法

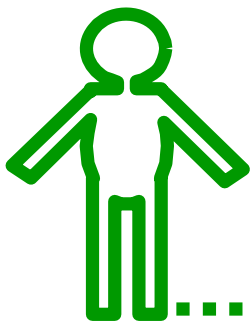
- ◇ ユーザビリティテストの目的説明後，課題実施
  - ▼ 現金引き出し，残高照会，振込みなど
- ◇ 終了後テストに関する事後質問調査を行い，補足的にインタビューを実施



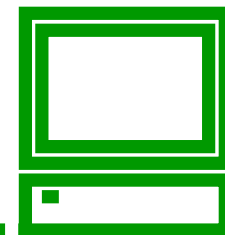


# マタニティスーツ





## 事例2 ATM



### ◆ 装置

- ◇ 対象システム：一般的な銀行向けATM
- ◇ 記録用機器，ビデオカメラ，タイピン型マイク，録音装置など

### ◆ 結果

- ◇ テスト時間
  - ▼ 大学生：約30分
  - ▼ インスタント高齢者用装置を装着した大学生：約1時間
  - ▼ 高齢者：約1時間半
- ◇ 高齢者は，悪いデザインの影響を受けやすい
- ◇ 一度エラーに陥ると抜け出しにくく，同じエラーを繰り返す

認知的高齢化の側面を検討していくことの重要性が示された