BIRZEIT UNIVERSITY

COMP2331

User-Centered Development (UCD)

By: Mamoun Nawahdah (Ph.D.)
2015/2016

HCI Design and Evaluation Techniques

- Introduce **U**ser **C**entered **D**esign (UCD)
  - Process
  - Methods
- Integrate UCD in software development lifecycle.
- Focus on the end user from the beginning.

HCI Introduced

Definition:

"*Human Computer Interaction is a discipline concerned with the **design**, **evaluation**, and **implementation** of **interactive** computing systems for human use and with the study of major phenomena surrounding them*"

(ACM SIGCHI, 1992)

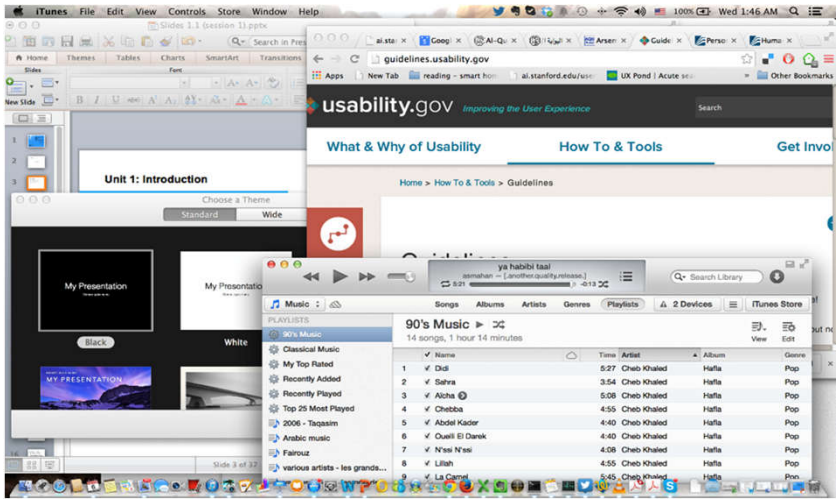
3

HCI Introduced

- HCI Science combines knowledge, techniques and methods from domains like:
  - Psychologists (علم نفس)
    - Experimental, Educational, Social and Industrial Psychology.
  - Computer Science
  - Instructional and Graphic Design
  - Technical Writing
  - Human Factors and Ergonomics (بيئة العمل)
  - Anthropology (علم الانسان) and Sociology (علم الاجتماع)

Look and Feel

- New “look and feel” of computers (Mac)



Look and Feel

- New “look and feel” of computers (Windows)

Devices

- Moving to small devices

Usability

Most accepted definition for usability is:

"The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use"

(ISO 9241-11)

Usability cont.

- Other definitions of usability:

  "***Usability is concerned that humans who use the product can do so quickly and easily to accomplish their own tasks***"

  Focusing on users, use products to be productive.

  (Redish & Dumas, A Practical Guide to Usability Testing, 1999)

Usability Requirements

- Every designer wants to build a **high quality** user interface.
- Careful planning and process is needed.
- System goal is always: to be easy to use, accessible, comprehensible, intelligible, idiot proof, available and ready.
- This requires a systematic process.
- **Goal**: develop usable systems, taking into account the specific users in their specific context.

Goals for Requirements Analysis

- Ways taken to assure the **user needs**:
  - Determine the tasks and subtasks that must be carried out.
  - Cover tasks that are only performed occasionally
    - Common tasks are easy to identify.
  - Functionality must match the user need or else users will reject or underutilize the product.

Goals for Requirements Analysis

- Deliver with **reliability**:
  - Actions must function as specified in needs / requirements.
  - Data displayed on screen must match the actual database records.
  - Ease the user sense of distrust.
  - The system should be available as often as possible.
  - The system must not introduce errors.
  - User privacy and data security should be assured, by protecting against unwarranted access, destruction of data, and bad usage.

Usability Measures

- **ISO 9241** definition of usability
  - focuses on the goals of **effectiveness**, **efficiency** and **satisfaction**.
- Target user community and classification of associated tasks related to the interface need to be identified.

Usability Measures Cont.

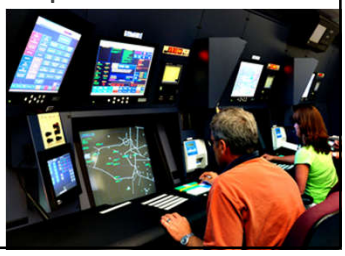
- Communities evolve overtime.
- Needs and models of use change.
- Shneiderman and Nielsen added the **usefulness factors**:
  - **Time to learn:** How long does it take for typical members of the community to learn relevant task?
  - **Speed of performance:** How long does it take to perform relevant benchmarks?
  - **Rate of errors by users:** How many and what kinds of errors are made during benchmark tasks?
  - **Retention over time:** Frequency of use and ease of learning help make for better user retention.
  - **Subjective satisfaction:** Allow for user feedback via interviews, free-form comments and satisfaction scales.

Usability Measures Cont.

- Design options have always **trade-off**s:
  - Changes to the interface in a new version may create **consistency** problems with the previous version.
  - However, the new changes may **improve the interface** in other ways or introduce **new needed functionality**.
- Design **alternatives** can be evaluated in several ways like by designers and users via **mockup**s or **high-fidelity** prototypes.
- Getting feedback early enough is:
  - The basic tradeoff.
  - Perhaps less expensively in the development process versus having a more authentic evaluated interface.

Usability Motivations

- Lots of interfaces are poorly designed; this is true across domains.
- Example 1: **Life-critical systems**: In an air traffic control, nuclear reactors, power utilities, police & fire dispatch systems, medical equipment.
  - High costs, reliability and effectiveness are expected.
  - Length training periods are acceptable despite the financial cost to provide error-free performance and avoid the low frequency but high cost errors.
  - User satisfaction is less an issue due to well motivated users.

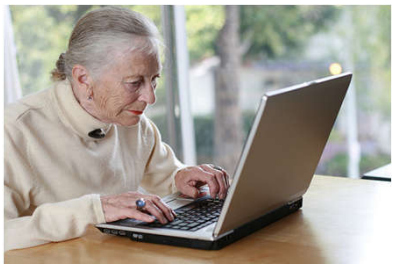


Usability Motivations cont.

- Example 2: **Office, home, and entertainment applications**: Word processing, electronic mail, computer conferencing, and video game systems, educational packages, search engines, mobile device, etc.
  - Ease of learning, low error rates, and subjective satisfaction are very important.
  - Infrequent use of some applications means interfaces must be intuitive and easy to use.
  - Choosing functionality is difficult, the population has a wide range of both novice and expert users.
  - Competition cause the need for low cost.

Universal Usability

- Elderly users
- Younger users



Usability Testing

- Usability testing may put the user in a separate controlled environments and asking them to perform tasks on their own without any offered help and measure their **quantitative performance**.
- The goal of usability testing:
  - Find errors in a particular interface(s), with a small or large number of user in a structured or unstructured style.
  - Not to generalize the results but improve interface of an application.
  - Not stating a statistical significance.

Usability Testing

- Laboratory or non laboratory research methods are available.
- Methods include:
  - **Observations**
  - **Field studies**
  - **Focus groups**
  - **Expert reviews**
  - **Surveys**
  - **Interviews**
  - **Controlled experiments**

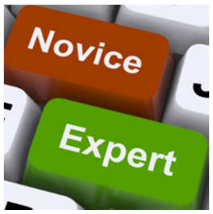
Usability and Games

- Playing games that are frustrating because of usability problems or interface faults is not acceptable.
- Usability should be taken into account in all phases of game development.
- User experience (UX) **MUST** be high in serious games.

21

Know the User (Player)

- Age, gender, physical and cognitive abilities, education, cultural or ethnic background, training, motivation, goals and personality.
- Design goals based on skill level:
  - Novice or first-time players.
  - Knowledgeable intermediate players.
  - Expert players.



"Okay your father managed to get a mouse. Now how do we use it?"

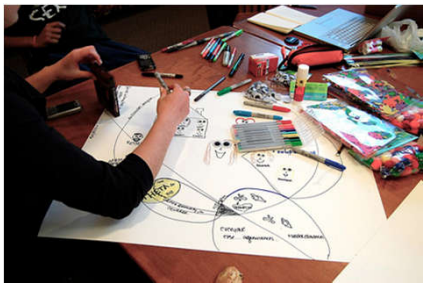
22

Collecting Information about the User!

- Collect data from target players of games, only the player knows how it should behave and the enjoyable interaction.
- NEVER from developers but from players themselves in their contexts.
- Observation.
- Metaphors.

23

Participatory Design



Participatory design where users visualize their game designs in groups.

24

Participatory Design

- Definition: "A design approach used in user centered development, where representatives from **stakeholder**s are actively involved in the processes and procedures of the design."
- In this approach the users/players and the wider public are also recognized as stakeholders and are involved into the process as well.

25

Participatory Design cont.

- Controversial (debatable)
- In one side, more user involvement brings:
  - More accurate information about tasks.
  - More opportunity for players to influence design decisions.
  - A sense of participation that builds players' ego investment in successful implementation.
  - Potential for increased player acceptance of final game.

26

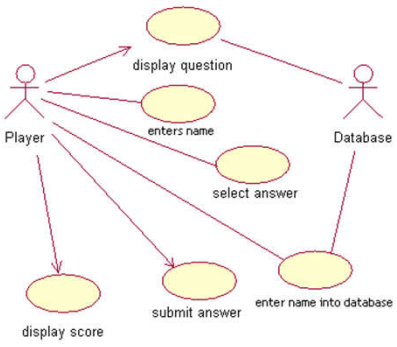
Participatory Design cont.

- On the other side, extensive player involvement may:
  - Be more costly.
  - Lengthen the implementation period.
  - Force designers to compromise their design to satisfy incompetent (useless) participants.
  - Build opposition to some game concepts.
  - Players have no expertise in the game systems nor the serious topic; subject of the game?

27

Scenario of Use (Use Case)

- Scenarios about the daily life activities/actions.
- Describe how play in typical games.
- Provide examples of play as input to design.
- Best to be done in a walkthrough (task based) approach.
- Tools of good help:
  - Table classifying player types and their ways of interaction.
  - Table of gameplay sequences.



28

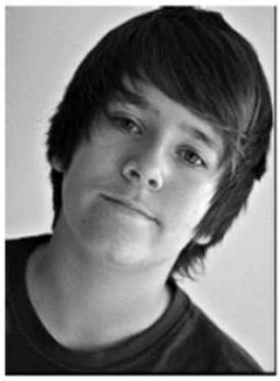
Scenario cont.

- May make use of **personas** (prototypical players)
- Based on hypothetical player.
- Ask questions: “how would Abass react if. . . ?”
- Non-essential details help things appear real.
- Avoid designers.
- Don’t make it for yourself but the player.

29

Persona example

Persona



Name:

Karel

Age:

10

Story:

Karel is a 10-year old boy addicted to videogames. Everyday right after school, he goes home and fires up his PlayStation and plays non-stop. His parents already tried several times to convince him to play outdoors with some friends. He thinks it is boring. During a medical check-up, doctors noticed Karel gained weight significantly, and his physical condition is very poor. They advised Karel’s parents to make him do physical activity at least once a day to improve his overall health.

30

Prototypes

- Paper
- Wireframes and Script (PowerPoint, flash, Lucid Charts, etc.)
- Animated Prototypes, Visual Programming
- Functional Prototypes

31

BIRZEIT UNIVERSITY

COMP2331

Serious Game (SG) Design

By: Mamoun Nawahdah (Ph.D.)
2015/2016

What is Game Design?

Game design is the process of designing the **content** and **rules** of a game in the **pre-production stage** and designing the **game play**, **environment**, **storyline**, and **characters** in the **production stage**.

SG Design

How is this applied in SGs?

In SGs, game design can be applied to create a game to facilitate interaction between players for **playful**, **healthful**, **educational**, or **simulation purposes**, and especially in the virtual games.

Defining the Goal of the SG

- The main goal when designing a SG should be to create an **exceptionally**, **memorable** and **applicable experience**.
- That helps players remembering their experiences, which means that the learning outcomes will naturally be attached to that experiences.
- Designing a great and memorable game experience leads to create a solid foundation to build simulations and serious games.

Defining the Goal of the SG

- Designing and creating applicable and memorable experience is very challenging, because the experience will vary from one player to another, and this is called ***experience variance***.
- To address experience variance problem, the game design must be **iterative**, which means there is no constant version of the game, the game must be redesigned in new versions based on the players feedbacks.

Primary Components of a Game

There are three main components of any game:

1. **The Story:** Story is when a SG scenario is created. Story gives the SG its context and relativity.
2. **The Rules:** Rules defined the rules of the SG. Without the rules the game is unmanageable, with too many rules, the game becomes boring. **Rules must be balanced.**
3. **The Design:** The design is how the game looks. The design is very important, because if it's looked confusing, the players will be confused. The design must be **attractive** and **captivating**.

Primary Components of a Game

- Balancing the three components: The three components are equally in importance, which means none of these components is more important.

SGs should have a great story, a great set of rules and looks great.

Primary Components of a Game

- If you have a very strong game-play but your design is terrible, players will not able to play the game.
- In opposition, if you have well beautifully game design but with poor rules, players will discredit the game.
- To ensure that the components are used in harmony and in balanced, designers have to test the game.
- What meant by "**testing**" is to let players play the game, then to look for certain reactions from them.

Players Feedback

The following set of questions are good to ask to the players:

- ***Do you understand what are you supposed to accomplish?***
  - This question tests and measures the story and the set of rules.
- ***Do you understand how to play?***
  - This question tests and measures both the design and the set of rules.
- ***Was it easy to learn how to play?***
  - This also test and measures the design and the set of rules.

Players Feedback

- ***Do you understand why you're playing this game?***
  - This test and measures the story part.
- ***What would you change?***
  - This question test them all (3 components), and test the game experience in general.
- Good game design (very good balancing of the three components) doesn't necessary mean **100%** acceptance from all the players, because there is no such a game.

Behavioral Game Design

- Computer games are mainly designed around the same element which is the ***player***, since the software and hardware for games change frequently, the **psychology** of how players interact to the game is a constant.
- This topic will discuss the behavioral psychology that looks for general rules for learning of how players minds respond to the game environment.

Behavioral Psychology Rules

Contingencies and Schedules:

- **Contingencies** (احتمال) is a rule or a set of rules controlling when rewards are given to the players.
- It's also referred as a **schedule of rewards**, for example when a player gain 1000 points his level increased by one, or a bonus level that is only available when he kill a certain opponent.

Behavioral Psychology Rules

Ratios and Intervals:

- Ratios provide rewards after specific amount of actions have been completed, and this is called ***fixed ratio***, because the same amount of actions is needed every time to get new reward.
- **Fixed ratio** gives the players a very distinct pattern, ranging between the long pause and then a steady burst of activity. Because players know that the first action will never bring them the reward, so they act as fast as they can to bring the reward quickly.

Behavioral Psychology Rules

Ratios and Intervals:

- There is also a ***variable ratio,*** variable ratio also requires specific amount of actions to get the reward, but this amount changes every time.
- Variable ratio makes the players respond with steady flow of activity, because they never know how many actions required for the next reward.

Behavioral Psychology Rules

Ratios and Intervals:

- ***Fixed Intervals*** is where reinforcement for a behavior is provided after a fixed amount of time has elapsed.
- Players usually respond to a fixed interval schedule by pausing for a while after a reward and then gradually responding faster and faster until another reward is given.

Behavioral Psychology Rules

Ratios and Intervals:

- ***Variable Intervals*** .is where the reward appears only after a certain length of time, as in the fixed interval, but the time is not fixed.
- The activity of players is lower than in the variable ratio, because the appearance of rewards doesn't depend on player's activity, and their motivation spread over the playing time.

How to make players maintain a high, consistent rate of activity?

- The answer is to use **variable ratio** schedule, where every player activity has the chance to produce a reward of the player.
- The activity level is how players expect a reward to occur.
- The more certain they know an interesting or something good will happen soon, the more they play harder.

How to make player play forever?

- This means, how to make players play the game, even if there is no immediate reward.
- Variable schedule is good because it produces constant probability for reward appearance, so, the players always has a reason to do the next thing.
- **Behavioral momentum (زخم)**, game designers needs a lot of players behavioral momentum, which keeps them doing what they're doing even if there is no immediate reward.
- **Avoidance schedule** produce a lot of behavioral momentum, where players play to prevent bad things from happening.

What situations lead players to quit playing?

- **Pausing**, where players motivation to do the next thing is low.
- To avoid pausing, players must have multiple activities possible at any given time, if doing something doesn't give them specific reward, then there is another possible action that gives them that reward.
- Dropping in the reward rate, if the new reward is 10 times the previous one, this will have a big impact on the players, on other hand, if the new reward is weaker than the previous one, this leads players to the state of anger and frustration.

BIRZEIT UNIVERSITY

COMP2331

Story Telling and Storyboards

By: Mamoun Nawahdah (Ph.D.)
2015/2016

Storyboards

- A storyboard is a sketch of how to organize a story and a list of its contents.
- Used to:
  - Define the parameters of a story within available resources and time.
  - Organize and focus a story.
  - Figure out what medium to use for each part of the story.

Storyboards

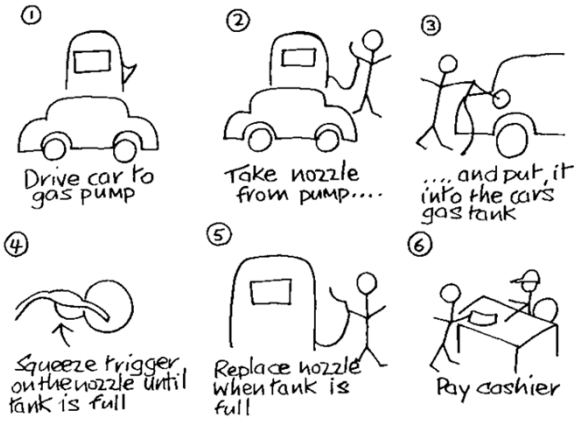
- Often used with scenarios, bringing more detail, and a chance to role play.
- It is a series of sketches showing how a user might progress through a task using the device/software/game.
- Used early in design.

3

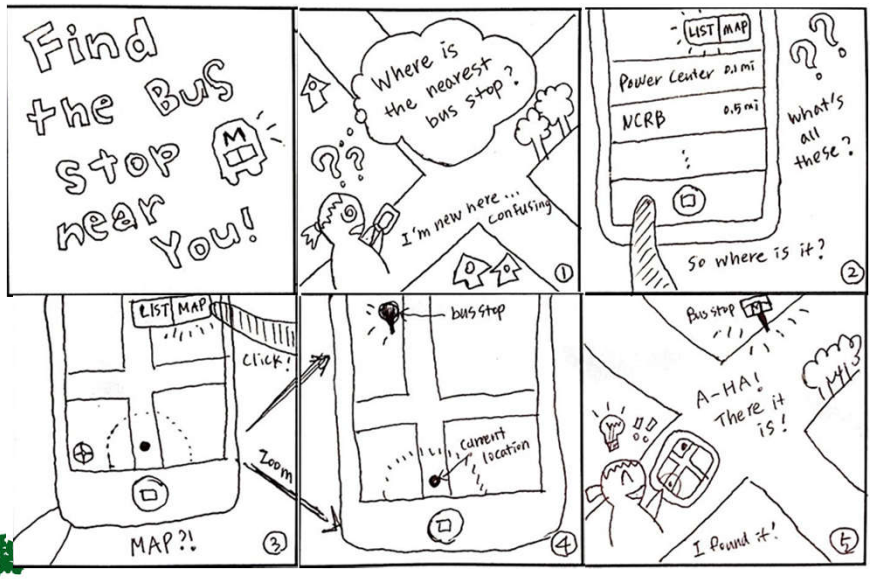
Sketching

- Sketching is important to low-fidelity prototyping.
- Don't be inhibited about drawing ability. Practice simple symbols.
- Example 1:

Gas station activity



Example 2: Find Nearest Bus Stop



Activity: Create a Storyboard

- Consider the following scenario:

  "Izaldeen went and got lunch at a local restaurant "إنسى غرامك راح". This was during world cup madness, so he was watching some football (soccer). He sat at the restaurant and got chicken wings. He watched the game and ate. He read the newspaper a bit too. The food at "إنسى غرامك راح" is always good. The team he picked won so that was also good."

- Turn this scenario into a Storyboard.

6

Storyboards

- Used to represent systems .
- Show the various interfaces (screens).
- Present in a system as well as the links between components.
- Can be used in a variety of contexts.
- Can show:
  - Relationship between data entry screens in a database
  - Scenes in a movie
  - Slides in a presentation
  - Pages in a website

7

Storyboards

- Representation of each interface should be detailed enough for the reader to identify the **purpose**, **contents** and **design elements**.
- Areas used for **input**, **output** and **navigation** should be clearly identified and labeled.
- Any links shown between interfaces should originate from the navigational element that triggers the link.

8

Create Storyboard

- Combination of video, text, still photos, audio, graphics and interactivity presented in a nonlinear format.
- Information in each medium should be complementary, not redundant.
- Storyboard should be put together with all those elements in mind.

9

Your Project Idea

- Squeeze your idea
  - What is it about?
  - Purpose of the game?
  - What are the awards?
  - Messages to pass?
  - etc.

10

BIRZEIT UNIVERSITY

COMP2331

ADDIE vs. Scrum

By: Mamoun Nawahdah (Ph.D.)
2015/2016

How Angry Birds Built a Global Brand?

- The success of Angry Birds was not an accident.
- The Angry Birds development project was a purposeful project created by an experienced game development company.
- Rovio had previously published 52 other games and created 16 original games.
- The 12-person team spent 8 months carefully studying the iPhone applications and developing the game.
- They refined the game many times before it was released.
- It took a great deal of research, engineering, and prototyping to make it success.

ADDIE Process Model

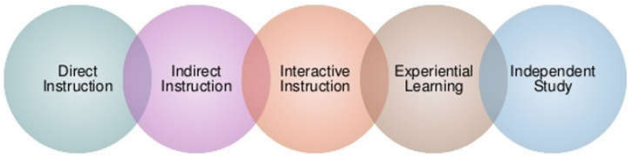
- Used for creating instruction based on five steps:
  - **A**nalysis
  - **D**esign
  - **D**evelopment
  - **I**mplementation
  - **E**valuation
  - **M**anagement

ADDIE: Analysis

- In this phase:
  - The type of problem to be solved by the instruction is analyzed.
  - Takes into account the type of content to be learned:
    - Declarative knowledge
    - Problem-solving knowledge
  - Looks at the learners to see their prerequisite skills.
  - Looks at the technology available for the delivery of the learning solution.

ADDIE: Design

- In this phase a robust design document is produced:
  - The instructional objectives are written. e.g.:
    - "The learner will be to identify... "
  - The appropriate **instructional strategies** are selected.
  - Assessment items are created to match objectives.
  - Storyboards.



- Usually, a client signs off on the design document.

ADDIE: Development

- In this phase:
  - The programming and creation of the instruction occurs.
  - Creating interface items for the instruction.
  - Creating and loading images and sound files.
- At points throughout this phase, a **formative evaluation** process takes place by learners and stakeholders.
- If changes need to be made, the process might go back to the design phase.

Formative Assessment

- The goal of formative assessment is to gather feedback that can be used by the instructor and the students to guide improvements in the ongoing teaching and learning context. These are low stakes assessments for students and instructors. Examples:
  - Asking students to submit one or two sentences identifying the main point of a lecture.
  - Have students submit an outline for a paper.
  - Early course evaluations.

ADDIE: Implementation

- In this phase the actually rollout of the instruction to the learners happen.
- It involves making sure the instruction can run on any required computer systems or the instructors using the materials understand how they are to be utilized.

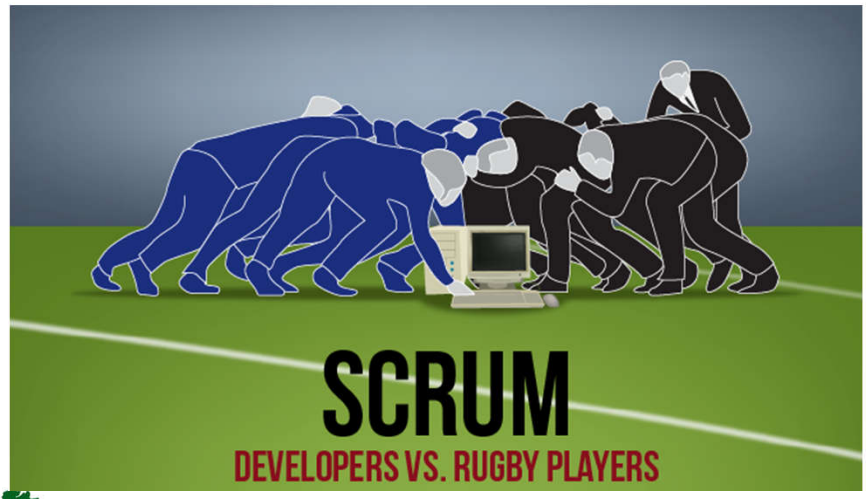
ADDIE: Evaluation

- Divide into two types:
  - **Formative evaluation**: occurs during the design and development.
    - Materials are reviewed, feedback is presented to the team, and changes are made as needed.
  - **Summative evaluation**: occurs at the end when the assessment is made of the utility of the instruction.

Summative Assessment

- The goal of summative assessment is to measure the level of success or proficiency that has been obtained at the end of an instructional unit, by comparing it against some standard or benchmark. Examples:
  - Assigning a grade to a final exam
  - Critique of a Senior presentation
  - University Faculty Course Evaluations

SCRUM (Agile)



SCRUM: Product backlog

- The scrum process starts when a **stakeholder / owner** is handed a **vision** or idea for a new product or update.
- From that vision/idea, a list of requirements is created. ➔ **Product Backlog**
- Product backlog items is **incomplete** as emergent requirements appears.
- With **ADDIE** we define all possible requirements in a design document up-front.

SCRUM: Sprints (عدو سريع)

- The project goes in one of the many iterative cycles called **sprint**s.
- The product owner is responsible for prioritization of the product backlog items.
- Sprint can range from 2 to 5 weeks starts with one day for planning and ends with one day for revision.
- Goal of each sprint is to develop finished functionality that can be reviewed by the produced owner.
- While in sprint, no changes/modification can be made.

SCRUM: Allows for changes

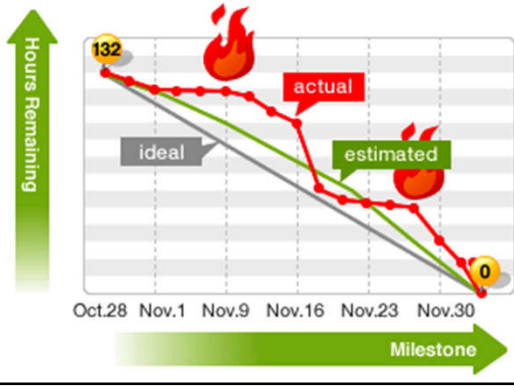
- Upon reviewing the results of the sprint, typically additional emergent requirements are added to the product backlog and reprioritized.
- In the next sprint, the modifications can be made **if** they are of a high enough priority.
- This allows for software to be developed quickly and changes to be easily incorporated into the process.
  - Improves the quality.
  - Reducing the risk.

SCRUM: Daily Meetings

- A daily meetings called **Scrums** are planned.
- These meetings bring the entire team together and address three questions from each member:
  1. What have I done since the last meeting?
  2. What do I still need to do?
  3. What obstacles have I encountered?
- **Scrum master**: to facilitate daily meetings.

SCRUM: Tracking the Progress

- Progress of the entire project is tracked with a tool called **burn down chart**.
- Burn down chart visually shows how much work is left to do versus how much time remains.



SCRUM: an Overview