Collaborative Design Session

A.K.A. a JAD (Joint Application Development) session
Purpose of a CDS

• To work with applications users interactively
• To rapidly define requirements
• To create a “strawman” (a paper mockup) of a possible UI for an application.
• To use to help the design an development team to understand the users needs and wants.
• To help create a UI that will be acceptable to the Users
Risks

- **Risk:** What users think they want and what they will really accept are different
  **Mitigation:** Usability testing

- **R:** Users will think the application is farther along in development than it really is.
  **M:** Use paper prototyping instead of creating software based prototypes

- **R:** Using the time to gather information and/or do initial analysis instead of design.
  **M:** Do initial observations and analysis before the session.
Process

- Well defined roles of participants
- Interactive
- Iterative
- Facilitated brainstorming
- Clearly Defined Goals
- Documented Artifacts (drawings, documents, screens designs, etc.)
Process – Roles

- **Facilitator** (1) – guides and controls the process. Manages the time. Timeboxes discussion (keeps process focused). Does NOT do the design or contribute information to the design. Asks questions but does NOT answer the questions. Like the official in a football or soccer game.

- **Scribe** (1) – documents what significant information is developed during the session. Records the drawings, and other design that is done on the whiteboards/chalkboards/whitepaper, etc. Does NOT ask or answer the questions involved in the design. Good handwriting and drawing skills. Keeps good, concise notes.
Process – Roles (2)

• **Design Team (1 or more people)**—asks questions of the users. Defines and refines the requirements derived from the information gathered previous to the session and during the session. Draws suggested designs with the input of the users. Suggests the layout, controls, placement, flow, information architecture, etc.

• **User Team (1 or more people)** – answers questions about their business processes and business problems. Tells the designers what their needs and wants are. Gives feedback and suggestions about the designs created by a design team.
Process - Interactive

• People ask questions, give answers, suggests designs, get feedback, ask questions, give answers, etc.

• To manage this chaos is the facilitators role.
Process - Facilitated Brainstorming

- There are no stupid questions or stupid answers.
- Many seemingly “stupid” questions often lead to innovative design approaches. Brainstorm helps to discover these ideas in a manageable way.
- The facilitation is to prevent the discussion from getting off track. The facilitator may:
  - take discussions “off line” (if they are important but will take too much time in the current session)
  - declare a “flag on the play” (if someone is out of line, they get a warning, like a “yellow card” in soccer).
  - Timebox a discussion – give it a certain prescribe amount of time and then declare it is done. Often an interval like 5 minutes, 15 minutes, etc.
Process – Goals -> Artifacts

- Understand the users’ wants and needs through interaction and feedback.
- Develop software quicker that matches the user’s needs more.
- Define and refine:
  - Users
  - Actors
  - Actors Goals
  - Workflows
  - Users’ Requirements for the Application
  - Possible UI layouts, flows and functionality
Process - Artifacts

- Document:
  - Business Entities (objects/concepts)
  - Users
  - Actors
  - Actors Goals
  - Workflows – tasks and goals for a specific task
  - Users’ Requirements for the Application
  - Possible UI layouts, flows and functionality
Tools

- Whiteboard / Chalkboard / White Paper
- Markers / Chalk / Markers
- Information
- Observations
- Interviews
- Imagination
- Critical Thinking
Artifacts

Things that are produced in the software development process; info, analysis docs, requirements, design docs, prototypes, code, data, user documentation, etc.
Business Domain

Business Name:
Business Domain (General):
Business Domain (Specific):
Description:
Consumers:
Providers:
Suppliers:
Competitive Businesses:
Related Domains:
User

- real persons / system that interact with the business.
- They utilize the services or consume the products of the business.
- Ex: Jane, Jin, Zhou are users of NYU.
- A few students in each group should act as users.
- Assume the **persona** of a user
  - tell what they want the UI to do and why.
  - tell their goals, tasks, workflows and concerns to the designers.
  - **should not design the UI.**
  - Try to think like a real users NOT like a UI designer.
- They will try to design but you should **gently** encourage them to tell you what the problem is NOT what the solution should be (that is design.
- They can be given a solution and tell the designers why they think it will/won’t work for their problems.
User

Name:
Title(s):
Age:
Education:
Training:
Skills:
Limitations:
Other:
Actor

• Actors are abstractions of users.
• A user may be represented by one or more actor types. Ex: Jane, Jin, Zhou are users of NYU.
• Ex Actor titles:
  – grad student
  – java developer
  – coffee drinker
  – department admin
  – student club member
• Actors play one or more roles.
• Roles:
  – CS Undergrad, CSMasters Candidate, PhD CS Candidate
  – Beginning Java Dev, expert Java Dev, struts Developer, DB Developer
  – President Of Web Dev Club, secretary Of Web Dev Club, treasurer …
• Sometimes the actors name and the role are the same. You don’t always need to go to this level of detail.
Actor

Title:
Role(s):
Users in this group:
Age Range:
Education Range:
Training Range:
Skills Range:
Other:
Goals

• Actors have goals.
• Why they would use a given service or product.
• hope to achieve by using a given product.
• A goal accomplishes something.
• It usually results in a demonstrable change of state for a given actor.
• Ex: Grad Students attend NYU to accomplish the following goals:
  – get a degree, increase their subject knowledge, get a job, get a better job, meet girls/guys, 
    do something with their time, get away from their parents.
• Workflows are done to achieve a goal
• A workflow is a series of tasks/activities that are done in a sequence with decision made in between activities to determine what the next activities is.
• The degree program tracks are examples of a workflow – you take a certain sequence of courses in a certain order and if you get good enough grades in them, pass the core exams required and write a thesis, you will receive a degree.
• Workflows often take an extended amount of time to complete i.e., minutes, hours, days, or years.
• They are usually verb phrases. Example workflows are: FollowDegreeProgram, PurchaseHome, CleanUpApartment, DriveToSanFrancisco, FlyToTheMoonAndReturn, ResearchAndBuyMP3Player, LearnToProgramJava, LearnToDesignUIs, SetupHomeComputerNetwork, etc.
Workflows

- Workflows are done to achieve a goal
- A workflow is a series of tasks/activities that are done in a sequence with decision made in between activities to determine what the next activities is.
- The CS degree program tracks are examples of workflows – you take a certain sequence of courses in a certain order and if you get good enough grades in them, pass the core exams required and write a thesis, you will receive a degree.
- Workflows often take an extended amount of time to complete i.e., minutes, hours, days, or years.
- They are usually verb phrases.
- Example workflows are:
  - FollowDegreeProgram
  - PurchaseHome
  - CleanUpApartment
  - DriveToSanFrancisco
  - FlyToTheMoonAndReturn
  - ResearchAndBuyMP3Player
  - LearnToProgramJava, LearnToDesignUIs, SetupHomeComputerNetwork, etc.
Goal

Name:
Description:
Actors that share it:
Workflows that achieve it:
Other:
Business Entity

- A list of nouns that are relevant to the domain (the business your team is in)
- A person, place, thing, concepts, etc.
- Usually the things that would need to be modeled in a database for an application but could be accomplished without having software at all – remember Universities existed before computers existed
- Examples for NYU:
  - College
  - Students
  - CourseOffering
  - CourseInstance
  - Classroom
  - ClassroomTimeSlot
  - Professor
  - CourseCatalog
  - SemesterCourseOffering
Business Entities

Name:
Brief Description:
Physical or conceptual?
Persistent or transitory?
Attributes:
Owner:
Composition - “Has A”:
Inheritance - “Is A”:
Instances:
Constraints:
Used by what actors:
Workflow (Tasks)

Name:
Summary Description:
Goal(s) it achieves:
Steps and decisions:
  1.
  2.
  3.
  4.
  5.
  6.
  7.
  8.
Business Activities (Actions)

- Compile a list of verbs (activities/actions) relevant to the domain. (Current, past or future tense are OK.)
- An activity is a step (task) of one or more workflows in the domain.
- Ex: For NYU Grad School:
  - registering, graduating, attendingClass, withdrawFromSchool, failingACourse, passingACourse, passedACourse, etc.
- For a voting system:
  - registeringVoter, voterIsVoting, voting, tallyVotes, publishResults, contestTally, recountVotes.
Business States
(attributes of the entities)

- These are often adjectives.
- They modify/describe the state of a business entity.
- Example for NYU:
  - registeredStudent, unregisteredStudent
  - studentOnProbation
  - PhDCandidateThatIsAllButDissertated
  - courseTaughtThisSemester, courseCanceledThisSemester
  - courseWithFullEnrolment, courseWithOpenSlots
- Voter system examples:
  - unregisteredVoter, registeredVoter
  - countedVote, uncountedVote, rejectedVote
  - certifiedElection, uncertifiedElection..
Example Artifacts
Business Domain

- Business Name: **NYU CIMS CS Grad School – UI Class**
- Business Domain (General): **Collegiate Education**
- Business Domain (Specific): **Graduate CS Education**
- Description: **NYU CS department provides classroom instruction at the graduate level. It provides a very high level of service and exacting standards. It grants BS, MS and PhD in CS.**
- Consumers: **People that desire a CS degree or courses in CS (Students)**
- Providers: **Professors and Faculty**
- Suppliers: **The facilities and facilities people. Writers of text books. Software and Hardware manufactures.**
- Competitive Businesses: **Harvard, MIT, Yale.**
- Related Domains: **MIS, EE, AI**
User

Name: Sam
Titles: Student, Employee
Age: 20 something
Education: Closing in on MS in CS
Training: Years of school and an undergraduate degree. On the job experience.
Skills: Great college student skills. Excellent computer skills.
Limitations: none
Other: MS Windows machine
User

Name: Sheng
Titles: Student
Age: 20 something
Education: Undergraduate degree in CS
Training: Years of school and an undergraduate degree. On the job experience.
Skills: Great college student skills. Excellent computer skills.
Limitations:
Other: Linux Machine
User

Name: Ido
Titles: Student, Software Business Owner
Age: 20 something
Education: Undergraduate degree in CS
Training: Years of school and an undergraduate degree. On the job experience.
Skills: Great college student skills. Excellent computer skills.
Limitations: 
Other: Linux Machine
User

Name: Logan
Titles: Professor
Age: 30 something
Education: CS and Psych degrees
Skills: Good teaching skills
Limitations: 1 hand, questionable sense of humor
Other: Windows and Linux Machines
Actor

Title: NYU CS Grad Student
Role(s): Student
Users in this group: Sam, Sheng, Ido
Age Range: 20-30’s
Education Range: At least undergrad in CS.
   Excellent computer skills
Training Range: None for application.
Skills Range: Advanced computer user.
Other:
Title: **Professor**
Role: **Administrator, Educator**
Users in this group: **Logan, Dennis Shasha, ...**
Age Range: **20-90’s**
Education Range: **Advanced degree**
Training Range: **None for application.**
Skills Range: **Advanced computer user.**
Other:
Goal

Name: **InsureRegistration**

Description: **Check to insure all the students in the lecture hall are registered.**

Actors that share it: **Educator**

Workflows that achieve it: **ClassRegistrationProcess**

Other:
Goal

Name: GetIntoClass
Description: Do what is required to get into the course.
Actors that share it: Student
Workflows that achieve it: ClassRegistrationProcess
Other:
Business Entities

- **Name:** Course
- **Brief Description:** A class that has a professor, curriculum, web page, timeslot, lecture hall, student roster and one or more students.
- **Physical or conceptual?** Mixed
- **Persistent or transitory?** Persistent until completion time.
- **Attributes:** Name, ID Number, Course Description, Credit Hours
- **Owner:** NYU
- **Composition - “Has A”:** professor, curriculum, web page, timeslot, lecture hall, student roster, student(s)
- **Inheritance - “Is A”:**
- **Instances:** Only one per semester of this type are allowed
- **Constraints:** Limited number of students allowed. Requires a professor to teach it.
- **Used by what actors:** Educator, Administrator, Student
Workflow (Tasks)

Name: ClassRegistrationProcess
Summary Description: The actor must provide information required to register for the class and there must be space in the class
Goal(s) it achieves: GetIntoClass, InsureRegistration
Steps and decisions:
1. Get Student’s Info
2. Check if student is eligible to register. If no go to REJECT
3. Check if space in course. If no go to REJECT.
4. Add to course roster.
5. If requested, send confirmation email to student
6. Add to student account entry for cost of course.
7. Add to student transcript registration for this course
8. Done
9. REJECT: send notification to actor for reason for rejection.
10. Send information to administrator about REJECTION
Business Entities (Objects)

- Student
- Educator
- Course
- Section
- Roster
- Timeslot
- Lecture Hall Slot
- Credit Hour
- Homework
- LectureSlides
- CourseMaterials
- University
- LectureHall
- CourseWebsite
- Address
- Phone
- EmailAddress
- PersonName
- CourseGrade
- CourseSlot
- StudentCourseGrade
- StudentGPA
- CourseSlot
- StudentCourseRegistration
- StudentSemesterSchedule
- StudentCourseGrade
- TuitionBill
Business Activities (Actions)

- RegisterForCourse
- DropACourse
- AttendLecture
- DoHomework
- TakeTest
- PayTuition
Business States (attributes of the entities)

- Student – Enrolled, Graduated
- CourseSlot – Open, Filled
- StudentCourseRegistration - Registered, Started, Completed
- StudentSemesterSchedule – Empty, PartiallyFull, Full
- TuitionForSemester – Paid, Unpaid, InArrears
CDS 1

• Gather into your groups
• I will assign you to either a chalk board or “post it notes”
• Follow the worksheets
• I will help each group
• Possibly break into small groups to do user interviews in parallel
• Do the UI designing together
CDS 2

• Select a facilitator
  – Facilitator should be a person that speaks loudly (and understandably). Can manage the session.

• Select a scribe
  – Good handwriting and note taker. Willing to type up the session notes and drawings as an RTF doc and send them to the team.
CDS 3

- Spilt the rest of the group into Designers and Users – 50-50 split.

- Users
  - Some of you will act in the role of users. You will use your experience from observations & interviews you did.

- Select a lead designer
  - Person that will initially ask questions about the users wants and needs. Does the drawing on the board, initially. (You should rotate who draws.)
CDS 4

• Gather together and attempt to define the artifacts:
  – Business Entities
  – Users
  – Actors
  – Goals
  – Workflows
  – Design Sketches – pick a goal and sketch out a design for it.
CDS 5

• Design Constraints. Use only:
  – Swing based target
  – Application types and widgets we have discussed so far
  – Sovereign, Parasitic, Daemonic, …
  – Menus
  – Labels
  – Buttons
  – Check and Radio buttons
  – Combo and List Boxes
  – Text Boxes
  – Scroll Bars
  – Spin Control
How good do your drawings need to be?
UI Worksheet

• Title:
• Goals Achieved:
• Workflow it supports:
• Actors used by:
• For each control (other than text labels):
  – Name:
  – Purpose:
  – Action:
  – Enabled / Disabled by:
  – List Contents:
  – Constraints:
Sample Dialog: Register For Class

Register For Class

Name: 

Address: 

Phone: 

Email: 

Notify me by email of class assignments

Undergraduate Degree
Masters
PhD

Submit  Cancel
Dialog Worksheet

- Title: Register For Class
- Goals Achieved: InsureRegistration, GetIntoClass
- Workflow it supports: ClassRegistrationProcess
- Actors used by: Student
• Name: **NameField**   ControlType: **TextBox**
  • Purpose: **All user to enter email via the keyboard**
  • Action: **none**
  • Enabled / Disabled by: **NA**
  • Constraints: **Limited to 64 characters, required filled**

• Name: **AddressField**   ControlType: **TextBox**
  • Purpose: **Allow user to enter address via the keyboard**
  • Action: **none**
  • Enabled / Disabled by: **NA**
  • Constraints: **Limited to 64 characters**

• Name: **PhoneField**   ControlType: **TextBox**
  • Purpose: **All user to enter phone via the keyboard**
  • Action: **none**
  • Enabled / Disabled by: **NA**
  • Constraints: **Limited to 2-3-3-4 numbers. Must be a legal phone number (no alphas) characters. Optional field**
• Name: EmailField  
  ControlType: TextBox  
  Purpose: All user to enter email via the keyboard  
  Action: none  
  Enabled / Disabled by: NA  
  Constraints: Limited to 64 characters, Optional

• Name: SendEmail  
  ControlType: CheckBox  
  Purpose: If checked will send an email for each assignment, else not sent.  
  Action: none  
  Enabled / Disabled by: if no email address entered will be grayed out.  
  Constraints:

• Name: DegreeRadioButtons  
  ControlType: RadioButtons  
  Purpose: Select type of degree being pursued  
  Enabled / Disabled by: NA  
  Constraints: Only one is allowed to be selected. Default to undergrad.
• Name: SubmitButton  ControlType: CommandButton
• Purpose: Allows user to initiate registration
• Action: Saves user information and closes the dialog
• Enabled / Disabled by: disabled if user does not enter data into required fields.
• Constraints:

• Name: CancelButton  ControlType: CommandButton
• Purpose: Allows user to initiate registration
• Action: Saves user information and closes the dialog
• Enabled / Disabled by: disabled if user does not enter data into required fields.
• Constraints:
What is missing from the design?
CDS criteria

• Use the CDS worksheets provided.
• Create each of the text artifacts i.e. NOT the dialog drawings (have the scribe record them).
• Document at least 3 workflows.
• Create at least 3 application UI screens/windows based on those workflows. 1 per workflow at least.
• The facilitator must - have the scribe, the facilitator (and others) type them up and send them to the YOUR GROUP, YOUR TA AND TO ME by Friday, March 14th by 5PM. (no grade for this but a required part of the course). PLEASE ZIP ALL THE DOCS UP INTO A SINGLE ATTACHED FILE!
• The original paper documents must ALSO be submitted to the TA mail box by then.
• The blank documents are available on the class website as RTF format.
• The UI Screens (Wireframes) drawings can be:
  – scanned in and attached as JPG
  – or you can manually create the drawings as GIF/PNG/JPGs and attach them (using a paint program) (redrawing the paper versions)
Time To CDS!