SE456 – Software Process Project
Fall - 20081
Updated: 10/19/08

Project A - Room Reservation System

(Odd numbered teams)

In this project your team will develop a general-purpose room reservation system. The application will allow multiple web clients to access a central reservation server. Clients will be able to perform common meeting and conference room reservation services such as booking a room, browsing rooms for open times and canceling or updating a reservation. The user will start with your general solution and be able to configure the product to their specific needs.
The requirements for this system are given in the list below as desired features for room reservation system. The list is neither exhaustive nor definitive. It is up to your team to develop a list of features that best fit scope of your application.

Potential features:

- User (person attempting to reserve a room)
  - Book a room for a specific date, time and duration
  - Cancel a reservation
  - List future reservations
  - Confirm room was used at time requested
  - Identify purpose and participants of meeting
  - Book a room for a repeating time period if authorized
- Administrative (person who controls access to rooms)
  - Assign users a level of access
  - Add, remove rooms from the reservation system
  - Identify characteristics of room – capacity, AV, etc.
  - Block rooms from being reserved
  - Bump reservations
  - Monitor room usage by user
  - Limit reservations in a time period (day, week) per user
  - Produce reports on room usage

Again, these are merely some suggestions. Feel free to add to or modify these requirements based on the design of your system.

Project B - Museum Management System

(Even numbered teams)

This system will manage information typically associated with a museum – members, donors, artifacts, exhibits, collections and so on. The product should have the ability to be applied in a variety of museum settings – art, science, and history and specialty museums. The user will start with the product's general solution and configure the product to meet their specific needs.

The technology to be used for the product is undefined. Ideally, the user interface components should be

web browser based to insure compatibility among different platforms. Additionally the product should have functionality for the user to build and display exhibit information at kiosks around the museum. The same exhibit information could also be used for a "virtual tour" of the museum on the internet.

The information identified below is typical of the museum type of entity being described. Your team may elect to expand on the information types described based on the design of your product. You may also determine that your product needs to model additional entities not identified below.

Membership - Museums maintain information on their members. Common types of data are name, contact information (addresses, phone, email), family members, ages, membership categories (individual, family, senior) and a history of membership fees.

Donors - Donors represent groups or individuals who make donations to the museum in the form of a monetary contribution, the sponsorship of an exhibit or by giving or loaning an actual museum artifact. Common types of data are contact information and data associated with the type of donation being made.

Artifacts - Museums maintain a record of every individual artifact in the museum. Common types of data include a description of the artifact, acquisition information (how, date, cost), an image if available, size and weight information, condition, current location (on display, on loan, in storage, out for repair, etc.).

Exhibits - Museums display artifacts in exhibits. Exhibits may be permanent or temporary. Temporary exhibits are displayed for a fixed period of time and may use artifacts that are displayed in permanent exhibits. Exhibits maintain data associated with the name and type of exhibit, associated artifacts, time duration, location and sponsor if applicable. One or more museum employees or curators are generally assigned responsibility for an exhibit.

Events - Museums hold events on a regular basis. Typical events include exhibit openings, guest speakers, museum classes, movies and social events for members and donors. Data associated with events include location, date and time information, admission costs, responsible museum personnel, external museum support (teachers, speakers) and registration or reservation information if applicable.

Project Deliverables (details will be discussed during class)

.

- Draft Software Development Plan — Fri, October 24 - 12:00 midnight
- Final Software Development Plan — Sun, November 2 -12:00 midnight
- Team Activities — Week 10

Notes (watch this section for updates)

- The goal of the assignment from our course's perspective is to create a ***plan*** for executing the entire project, not a completed implementation.
- Though you will not have time to develop the actual system, it may be good idea to familiarize yourself with the required technology in order to produce a more accurate plan.
- Remember that planning is an iterative process – don't attempt to draft the "final" plan in one swoop.

Key Components of a Software Development Plan:

- A sufficiently complete, ***professional*** document for communicating information to software engineers and management needed to understand what the project entails, how it will be produced and controlled and what the effort and schedule estimates are for the project. The project risks, quality focus, and support needs are specified as well.
- Overview – three to five paragraphs describing product function, platform, customers, schedule and development responsibility.
- State goals and scope (what's in scope, what's out of scope)
- Deliverables – high level releases and content
- High-Level Functionality – one paragraph overview for each major area of product functionality.
- Project organization – management, roles, staffing
- Risk Management – identification of risks, mitigation strategies, how they will be managed.
- Scheduling and estimates – work breakdown structure, overall project schedule, resource allocation, estimation techniques used (justify estimates), how will project be tracked, how will schedule changes be made.
- Measurements – what measurements will be collected, what metrics will be created, how will they be used, why were they chosen. These measurements will form the basis for the project's quality assurance plan and drive process improvement.
- Technical Process – what methodology will be used, what tools and techniques are required, what internal artifacts are required to be maintained. (This section typically references other development documents – requirements, design, test plan, etc. The SDP identifies what artifacts are to be created and how they will be maintained.) ***Note: Select a Plan-Driven methodology when creating your plan.***

Software Project Plan References

- Construx Software templates & examples (McConnell's company) – see "Engineering Management Section". You may use the Project Charter template for the Software Project Outline deliverable. Note that you will need to create a free login for this site. There are many very useful document templates and checklists available here.
- CxOne templates and examples : CxOne Basic Download (zip file)
- Sample Project Plan