Arena

An Online Multiplayer Battle Game

Operational Concepts (What is it?)

- Game played over the internet
- Turn-based RPG-style combat
- Players form teams and compete against other teams
- User accounts and profiles saved for continual use and character development

Project components

- Client GUI – Arpi
- Chat Engine, Music – Chris
- Character Design – Steve
- Battle Engine – Jessica
- Database - Zhen
- Web Services – Ron

GUI Specification

- Design
  - Advice from game players in group
  - Looking at other successful RPG GUI's
- Main GUI Components
  - Login
  - Create Account
  - Create Character
  - Ready Room
  - Battle Screen

Login

- Simple Screen
- Logging in connects to game server
- Create Account

Create Account

- Process Funnel
- Quick

Create Character Screen

Ready Room

Battle Screen
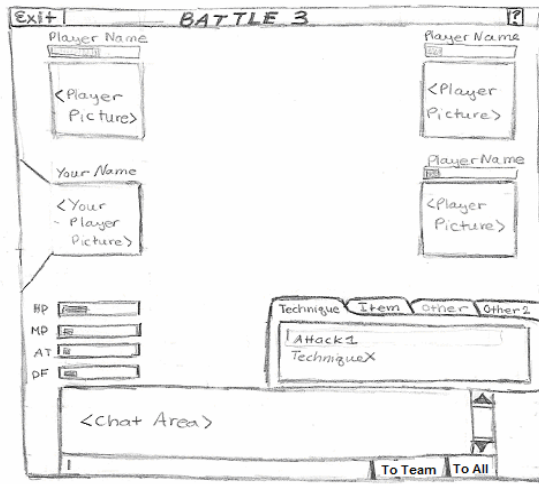
GUI Architecture

- Implement GUI using C# Visual Studio .NET.
- This environment provides useful and generic widgets.
  - Radio buttons, selectors, scrolling bars, buttons
- Other Graphics
  - will be created by our group
  - found through Internet game creation communities.

Chat Engine

- Why player chat is important
- System features
  - Team chat, private chat
- Implementation details
  - Locations, sub-locations

Music

- Plays .mp3 files in the background
  - Don't worry: it's legal
- Classic RPG style
- Music changes with screens
  - Login music different than Battle music
  - Victory music for winners, etc.

Character Design

- Specification
  - Classes, Attributes, Skills, Items
  - Character progression
  - Changing classes
- Architecture
  - Modular design of extensible classes
    - Abstract superclasses, skill reuse

Battle Engine

- Round progression
  - Blind initiative
  - Selecting action
  - Calculating outcome, reporting
- Status effects
  - Beneficial or harmful
  - Do not persist between games
- Experience rewards
  - Only given to winning team
  - Tougher battles worth more

Database

- Persistent storage
  - Character data
    - UserID
    - Password
    - Stats (items, skills, etc.)
- Temporary storage
  - User location
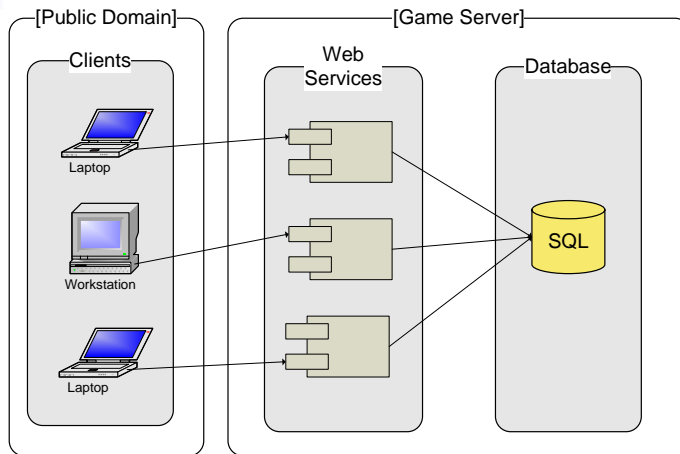   - For chatting/targeting purposes

Web Service Interface

- User login, character storage
- Character creation and progression
- Interactions during battle
- Chat interface
  - Communication Queue (CQ)
- Primarily verification
  - Especially of clients
  - Secures character information

System and Software Architecture

Release Milestones

- Zero-feature (Feb 4)
- Alpha (Feb 11)
- Beta (Feb 18)
- Feature-complete (Feb 25)
- Final (Mar 4)

Risk Assessment

- Inexperienced with development tools
  - Starting to get acquainted early
- Feature creep
  - Frozen at feature-complete release
- Component integration
  - Ensuring compatibility during development
  - Using automated regression tests
- Poor team communication
  - Weekly status reports
  - Active forum participation
- Reliance on external collaboration tools for storage
  - Regular backups of bug database