

# Atlanta Tasks

## Information

These are the tasks that should be completed for the Atlanta milestone (Dec 4th, 2006). Tasks are listed in order of criticality. The developer assigned to each task is listed to the right. If more than one developer is listed it indicates that the additional ones are assist as soon as their critical tasks are complete.

Critical tasks must be completed by the milestone date, and non-critical tasks should be completed by the milestone date if possible. Any tasks remaining after the milestone date must be completed as soon as possible.

Mark tasks as complete using this key:

<b>Key:</b>	Complete= 	Testing/QA= 	In Progress= 
-------------	---	---	--

## Tasks

- Bug fixes from test group [~fengr, Kapil Ahuja, Will Humphries](#)
  - Address all critical errors from the QA team
  - Mark non-critical errors to be fixed after Atlanta
- Summary page [Aaron Zeckoski](#)
  - The summary page must at least list evaluations as indicated in the wireframe and variable links to tools and control panel/admin
  - Ideally it should also hide boxes that are not useful at the time and list the sites correctly
- Take Evaluation functionality [Kapil Ahuja, ~fengr](#)
  - Students must be able to take the evaluations that are assigned and results must be recorded
  - Easy access to the evaluation must be provided via summary page
- Basic reporting [Will Humphries](#)
  - Must at least list all items and the numbers for each answer
  - Should provide a way to view the written responses (probably on a new window)
- Fixes for hibernate persistence [Aaron Zeckoski](#)
  - Add wrapper to allow lazy loading to work
  - Add batch update functionality to DAO
- Control Panel [~fengr](#)
  - Must at least list the available templates and currently running evaluations
  - Should allow the evaluation owner to immediately close a running evaluation
  - Should work as indicated in the requirements and wireframe
- Put groups of persistence operations into batch transactions [~fengr, Kapil Ahuja](#)
  - All operations which must happen together must be put into a batch write operation
    - Examples:
      - Saving template with first item, both saves should be done as a batch
      - Reordering items in the template, saves should all be done as a batch
      - Assigning evaluations, all assignments should be saved as a batch
- Download reports [Will Humphries](#)
  - Must at least provide a CSV download of the results of an evaluation
- Poster and handout [Will Humphries](#)
  - Should list poster items as indicated in the meeting
    - RSF framework
    - Sakai permissions integration (architecture)
    - Large variety of question item rendering options
    - Reusable templates and downloadable reports
  - Base handout on the handout from Vancouver with new screenshots and up to date links and features
- Do not allow users to get into an invalid state [~fengr](#)
  - All pages must not throw errors that user will see in any circumstance
  - All input should be validated so that invalid inputs do not result in errors
  - Ideally submit buttons should not work until required fields are filled in but error messages are ok
- RSF widgets for valid input [~fengr](#)
  - RSF Date widgets for date input
  - RSF WYSIWYG widget for textarea input
- Add summary page link and breadcrumbs to all pages [Will Humphries](#)
- page style [~fengr](#)
  - Include the evaluation css on every template page,
  - check button style on every page
  - add `<div class="portletBody">` for every template

## Non critical tasks

- Add testing for all logic and dao operations [~fengr, Kapil Ahuja, Will Humphries](#)
  - Add testing for all methods in the logic beans
  - Add testing for all methods in the DAO bean
- Code operation corrections [~fengr, Kapil Ahuja](#)
  - There should be no direct setting of session scope bean values
  - Logic beans should be injected into producers as needed instead of passing all values through the backing beans
- Email notifications for students [~fengr](#)
  - Setup emails to go out when evaluation first assigned

- Setup scheduling for reminder checks via quartz and send out reminders
- Email notifications for instructors ~fengr
  - Setup emails to go out when evaluation results viewable
  - Allow instructors to send email reminders to students using interface
- Comment added to all evaluations when admin option checked ?\*\* Should be added to the template but should not be visible or controllable by the owner
- Fix logging problems ~fengr
  - remove all System.out
  - Add appropriate logging as per appdev specs