

CSC 308 Lecture Notes Week 3
Details of the
Requirements Analysis Process

I. Week 3 material:

I. Week 3 material:

A. Milestone 2 writeup and example.

I. Week 3 material:

- A.** Milestone 2 writeup and example.
- B.** Requirements document HTML standards.

I. Week 3 material:

- A. Milestone 2 writeup and example.**
- B. Requirements document HTML standards.**
- C. Conventions for standardized GUIs.**

I. Week 3 material:

- A.** Milestone 2 writeup and example.
- B.** Requirements document HTML standards.
- C.** Conventions for standardized GUIs.
- D.** These lecture notes.

I. Week 3 material:

- A.** Milestone 2 writeup and example.
- B.** Requirements document HTML standards.
- C.** Conventions for standardized GUIs.
- D.** These lecture notes.
- E.** Week 3 lab notes, with more on SVN.

Milestone 2 Writeup

Milestone 2 Writeup

- Due Fri third week

Milestone 2 Writeup

- **Due Fri third week**
- **Deliverables:**

Milestone 2 Writeup

- Due Fri third week
- Deliverables:
 - Initial rough draft of Section 2.

Milestone 2 Writeup

- Due Fri third week
- Deliverables:
 - Initial rough draft of Section 2.
 - Top-Level UI(s).

Milestone 2 Writeup

- Due Fri third week
- Deliverables:
 - Initial rough draft of Section 2.
 - Top-Level UI(s).
 - Draft table of contents.

Milestone 2 Writeup

- Due Fri third week
- Deliverables:
 - Initial rough draft of Section 2.
 - Top-Level UI(s).
 - Draft table of contents.
 - At least one scenario per team member,
minimum three distinct screens per member.

Milestone 2 Writeup

- Due Fri third week
- Deliverables:
 - Initial rough draft of Section 2.
 - Top-Level UI(s).
 - Draft table of contents.
 - At least one scenario per team member,
minimum three distinct screens per member.
 - Updated admin/work-breakdown.html

Milestone 2 Example

Milestone 2 Example

- **Very rough draft of requirements.**

Milestone 2 Example

- Very rough draft of requirements.
- Section 2 of requirements doc.

Milestone 2 Example

- Very rough draft of requirements.
- Section 2 of requirements doc.
- Calendar project is similar to yours.

Milestone 2 Example

- Very rough draft of requirements.
- Section 2 of requirements doc.
- Calendar project is similar to yours.
- Editorial notes provide explanation.

Milestone 2 Example

- Very rough draft of requirements.
- Section 2 of requirements doc.
- Calendar project is similar to yours.
- Editorial notes provide explanation.
- For M2, focus on content primarily.

Section 2: Functional Requirements

Section 2: Functional Requirements

- Definition of all functions and data.

Section 2: Functional Requirements

- Definition of all functions and data.
- In scenarios depicting end-user interactions.

Section 2: Functional Requirements

- Definition of all functions and data.
- In scenarios depicting end-user interactions.
- Scenarios are in tutorial style.

Section 2: Functional Requirements

- Definition of all functions and data.
- In scenarios depicting end-user interactions.
- Scenarios are in tutorial style.
 - Tell interesting and engaging story.

Section 2: Functional Requirements

- Definition of all functions and data.
- In scenarios depicting end-user interactions.
- Scenarios are in tutorial style.
 - Tell interesting and engaging story.
 - Give step-by-step presentation.

Section 2: Functional Requirements

- Definition of all functions and data.
- In scenarios depicting end-user interactions.
- Scenarios are in tutorial style.
 - Tell interesting and engaging story.
 - Give step-by-step presentation.
 - Eventually cover all functionality.

Section 2.1: User-Interface Overview

Section 2.1: User-Interface Overview

- Standard section for all projects.

Section 2.1: User-Interface Overview

- Standard section for all projects.
- Present functional hierarchy of tool operations.

Section 2.1: User-Interface Overview

- Standard section for all projects.
- Present functional hierarchy of tool operations.
- Example uses menubar as concrete representation; *you need not*, but must have equivalent.

UI Overview, cont'd

- Note use of *very simple* GUI.

UI Overview, cont'd

- Note use of *very simple* GUI.
- More on GUI conventions in next Friday lab.

UI Overview, cont'd

- Note use of *very simple* GUI.
- More on GUI conventions in next Friday lab.
- ***IMPORTANT:*** Do not get bogged down in low-level GUI details in early stages of requirements.

UI Overview, cont'd

- Start with "When the user initially invokes ..."

UI Overview, cont'd

- Start with "When the user initially invokes ..."
- Figure 1 shows initial default screen.

UI Overview, cont'd

- Start with "When the user initially invokes ..."
- Figure 1 shows initial default screen.
- E.g., here's Figure 1 for Calendar example:

Calendar Tool						<input type="checkbox"/>	<input type="checkbox"/>
File	Edit	Schedule	View	Admin	Options	Help	

September 2015							<input type="checkbox"/>	<input type="checkbox"/>
Sun	Mon	Tue	Wed	Thu	Fri	Sat		
		1	2	3	4	5		
6	7	8	9	10	11	12		
13	14	15	16	17	18	19		
20	21	22	23	24	25	26		
28	27	29	30					

UI Overview, cont'd

- How system starts "out of the box" for typical user.

UI Overview, cont'd

- How system starts "out of the box" for typical user.
- Prose narrative following screen explains content.

UI Overview, cont'd

- Figure 2 shows expansion of command menus.

UI Overview, cont'd

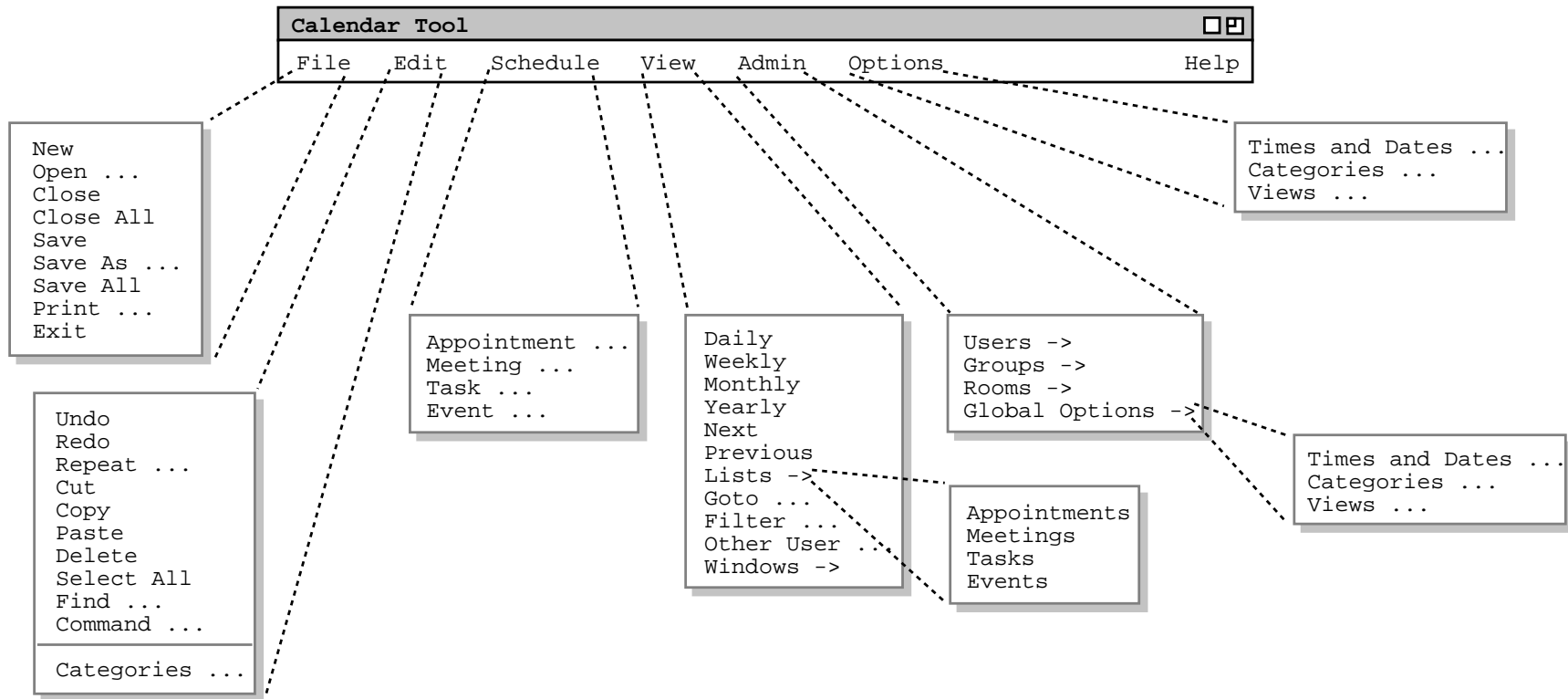
- Figure 2 shows expansion of command menus.
- Concrete representation of pulldown menu is convenient standard format.

UI Overview, cont'd

- Figure 2 shows expansion of command menus.
- Concrete representation of pulldown menu is convenient standard format.
- Conceptually, we are presenting a *functional command hierarchy*.

UI Overview, cont'd

- Figure 2 shows expansion of command menus.
- Concrete representation of pulldown menu is convenient standard format.
- Conceptually, we are presenting a *functional command hierarchy*.
- E.g., here's Figure 2 for Calendar example:



UI Overview, cont'd

- A pulldown menu is not the only way to represent a functional command hierarchy.

UI Overview, cont'd

- A pulldown menu is not the only way to represent a functional command hierarchy.
- It's a widely-recognized UI standard, at present.

UI Overview, cont'd

- A pulldown menu is not the only way to represent a functional command hierarchy.
- It's a widely-recognized UI standard, at present.
- Next slide shows equivalent functional hierarchy in plain text form.

UI Overview, cont'd

- A pulldown menu is not the only way to represent a functional command hierarchy.
- It's a widely-recognized UI standard, at present.
- Next slide shows equivalent functional hierarchy in plain text form.
- Plain text form is acceptable for Milestone 2.

File:

- New
- Open
- Close
- Close All
- Save
- Save As
- Save All
- Print
- Exit

Edit:

- Undo
- Redo
- Repeat
- Cut
- Copy
- Paste
- Delete
- Select All
- Find
- Command
- Categories

Schedule:

- Appointment
- Meeting
- Task
- Event

View:

- Daily
- Weekly
- Monthly
- Yearly
- Next
- Previous
- Lists:
 - Appointments
 - Meetings
 - Tasks
 - Events
- Goto
- Filter
- Other User
- Windows

Admin

- Users
- Groups
- Rooms
- Global Options:
 - Times & Dates
 - Categories
 - Views

Options:

- Times & Dates
- Categories
- Views

Sections 2.2 and Beyond

Sections 2.2 and Beyond

- **These sections differ for each project.**

Sections 2.2 and Beyond

- These sections differ for each project.
- For Milestone 2 they're rough and preliminary.

Sections 2.2 and Beyond

- These sections differ for each project.
- For Milestone 2 they're rough and preliminary.
 - Calendar example is top-down in style.

Sections 2.2 and Beyond

- These sections differ for each project.
- For Milestone 2 they're rough and preliminary.
 - Calendar example is top-down in style.
 - I.e., a detailed outline has been completed.

2.2 and Beyond, cont'd

- **Organizational guidelines:**

2.2 and Beyond, cont'd

- Organizational guidelines:
 - Generally, organize per functional hierarchy.

2.2 and Beyond, cont'd

- Organizational guidelines:
 - Generally, organize per functional hierarchy.
 - Refine organization with stylistic guidelines, to make document more readable.

2.2 and Beyond, cont'd

- Stylistic guidelines include:

2.2 and Beyond, cont'd

- Stylistic guidelines include:
 - Start with common activity for "reader warm up".

2.2 and Beyond, cont'd

- Stylistic guidelines include:
 - Start with common activity for "reader warm up".
 - Simple scenarios first, details later.

2.2 and Beyond, cont'd

- Stylistic guidelines include:
 - Start with common activity for "reader warm up".
 - Simple scenarios first, details later.
 - Separate scenarios for different user groups.

2.2 and Beyond, cont'd

- Stylistic guidelines include:
 - Start with common activity for "reader warm up".
 - Simple scenarios first, details later.
 - Separate scenarios for different user groups.
 - Leave mundane details until later, e.g., File, Edit.

2.2 and Beyond, cont'd

- Stylistic guidelines include:
 - Start with common activity for "reader warm up".
 - Simple scenarios first, details later.
 - Separate scenarios for different user groups.
 - Leave mundane details until later, e.g., File, Edit.
 - Leave details of error handling until later.

2.2 and Beyond, cont'd

- Scenario details:

2.2 and Beyond, cont'd

- Scenario details:
 - Typical scenario shows user selecting an operation.

2.2 and Beyond, cont'd

- Scenario details:
 - Typical scenario shows user selecting an operation.
 - Start with "... the user selects ...".

2.2 and Beyond, cont'd

- Scenario details:
 - Typical scenario shows user selecting an operation.
 - Start with "... the user selects ...".
 - Show resulting screen shot.

2.2 and Beyond, cont'd

- Scenario details:
 - Typical scenario shows user selecting an operation.
 - Start with "... the user selects ...".
 - Show resulting screen shot.
 - Explain screen contents in follow-on narrative.

2.2 and Beyond, cont'd

- Scenario details:
 - Typical scenario shows user selecting an operation.
 - Start with "... the user selects ...".
 - Show resulting screen shot.
 - Explain screen contents in follow-on narrative.
 - Continue in this style, showing user action and results, with generous explanatory narrative.

Section 2.2: Scheduling Appointment

Section 2.2: Scheduling Appointment

- This Calendar example is a typical rough draft.

Section 2.2: Scheduling Appointment

- This Calendar example is a typical rough draft.
- Figure 3 shows result of selecting 'Schedule->Appointment'.

Section 2.2: Scheduling Appointment

- This Calendar example is a typical rough draft.
- Figure 3 shows result of selecting
'Schedule->Appointment'.
- Explanatory narrative follows.

Schedule an Appointment

Title:

Start Date: Start Time:

End Date: Duration:

Recurring? Interval: S M T W Th F S

Type: Security:

Location: Priority:

Remind? When: How:

Details:

Figure 3: Appointment Scheduling Dialog

Scheduling Appointment, cont'd

Typical explanatory narrative following screen:

The title field is a one-line string that describes the appointment briefly. The date is the date on which the appointment is to occur. ...

Scheduling Appointment, cont'd

- **Figures 4-7 show results of additional user actions.**

Scheduling Appointment, cont'd

- Figures 4-7 show results of additional user actions.
- Explanatory narrative interspersed between each screen shot.

Scheduling Appointment, cont'd

... user selects Type : drop-down ...

Scheduling Appointment, cont'd

... user selects Type : drop-down ...

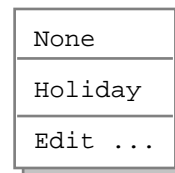


Figure 4: Initial categories menu.

Scheduling Appointment, cont'd

... user selects Type : drop-down ...

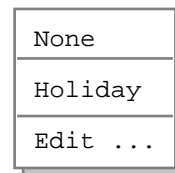


Figure 4: Initial categories menu.

Explanatory narrative ...

Scheduling Appointment, cont'd

... user selects 'Edit ...'

Scheduling Appointment, cont'd

... user selects 'Edit ...'

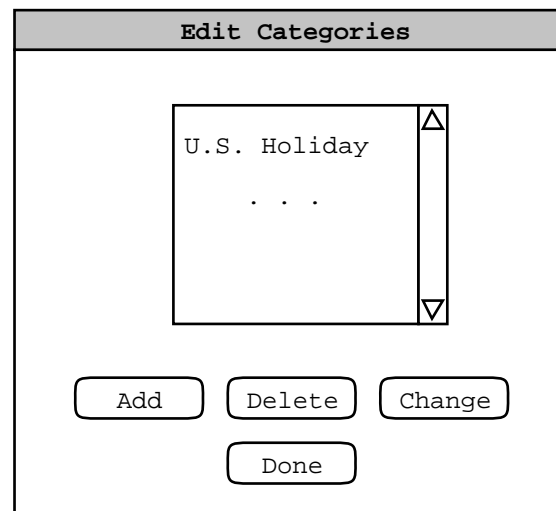


Figure 5: Edit categories dialog.

Scheduling Appointment, cont'd

... user selects 'Edit ...'

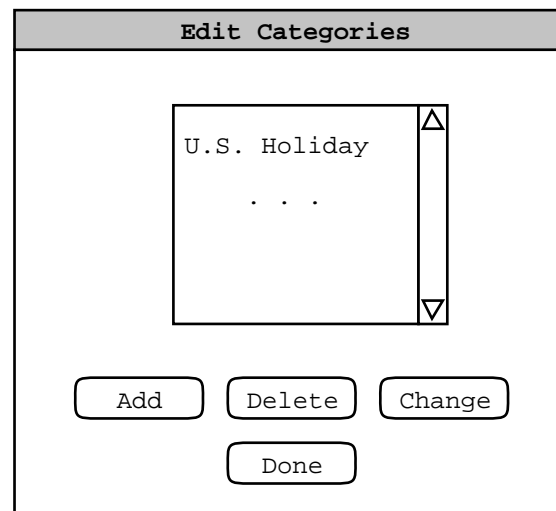


Figure 5: Edit categories dialog.

Explanatory narrative ...

Scheduling Appointment, cont'd

- *Explanatory narrative* will become more refined.

Scheduling Appointment, cont'd

- *Explanatory narrative* will become more refined.
- Eventually, all commands and data formats are covered at least once.

Scheduling Appointment, cont'd

- *Explanatory narrative* will become more refined.
- Eventually, all commands and data formats are covered at least once.
- We'll discuss further in upcoming lectures.

Section 2.3. Browsing

Section 2.3. Browsing

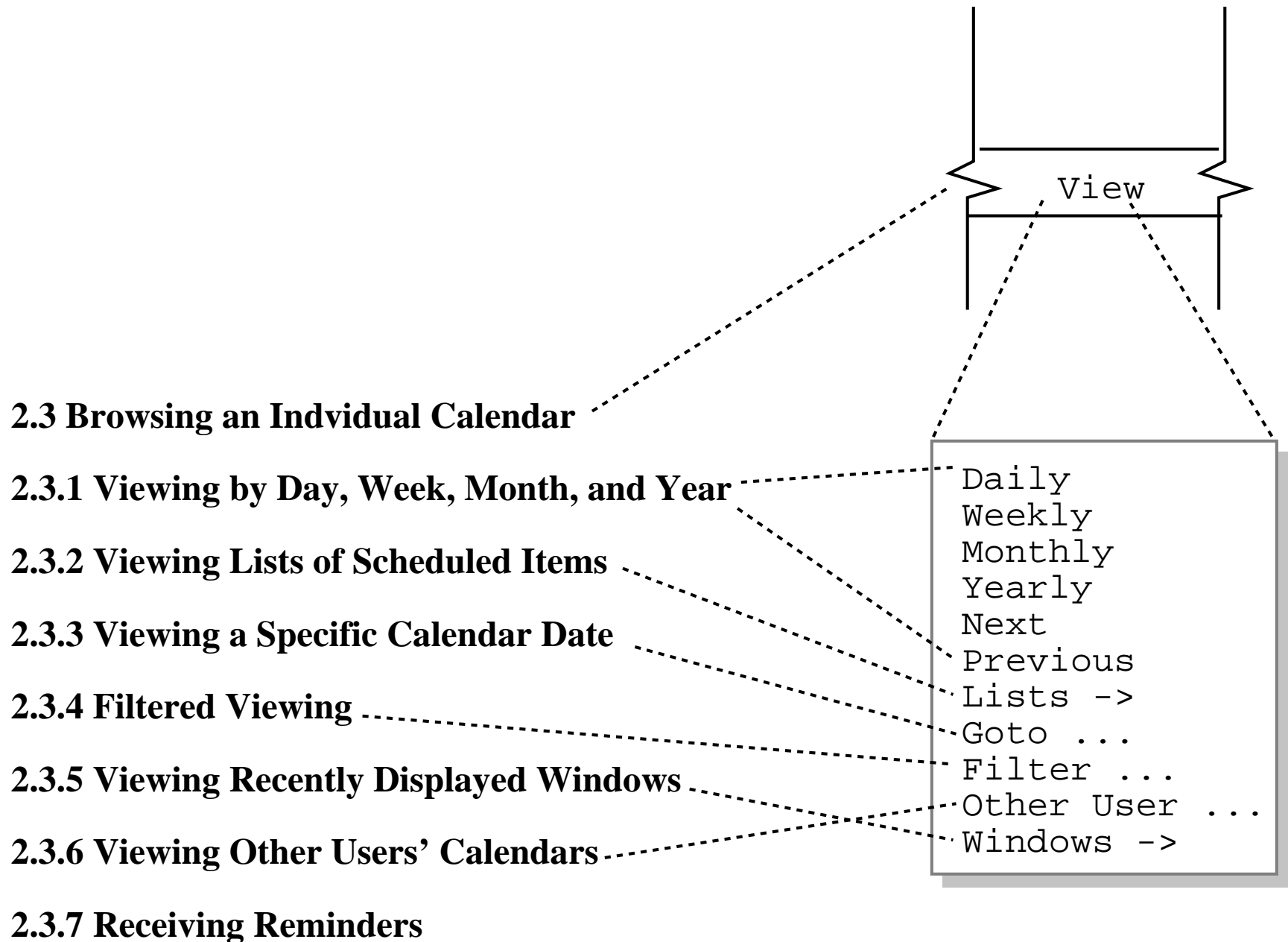
- Editorial remark explains that this and remaining sections are skeletons.

Section 2.3. Browsing

- Editorial remark explains that this and remaining sections are skeletons.
- A number of browsing scenarios are planned.

Section 2.3. Browsing

- Editorial remark explains that this and remaining sections are skeletons.
- A number of browsing scenarios are planned.
- Scenario order generally follows layout of commands in 'View' menu.



Critique of Section 2.3

Rough Draft Organization

Critique of Section 2.3 Rough Draft Organization

- For consistency, use term "Viewing" instead of "Browsing".

Critique of Section 2.3

Rough Draft Organization

- For consistency, use term "Viewing" instead of "Browsing".
- Section 2.3.1 may get too big.

Critique of Section 2.3

Rough Draft Organization

- For consistency, use term "Viewing" instead of "Browsing".
- Section 2.3.1 may get too big.
- Flip order of 2.3.5 and 2.3.6 to be consistent with functional hierarchy.

Critique of Section 2.3

Rough Draft Organization

- For consistency, use term "Viewing" instead of "Browsing".
- Section 2.3.1 may get too big.
- Flip order of 2.3.5 and 2.3.6 to be consistent with functional hierarchy.
- Minor details at this point, but worth noting.

Section 2.4. More Scheduling

Section 2.4. More Scheduling

- These scenarios cover remaining commands in 'Schedule' menu.

Section 2.4. More Scheduling

- These scenarios cover remaining commands in 'Schedule' menu.
- Stylistically, the "simple-to-more-detailed" guideline is being used here.

Section 2.4. More Scheduling

- These scenarios cover remaining commands in 'Schedule' menu.
- Stylistically, the "simple-to-more-detailed" guideline is being used here.
 - I.e., start with simple scenario on basic scheduling (Section 2.2).

Section 2.4. More Scheduling

- These scenarios cover remaining commands in 'Schedule' menu.
- Stylistically, the "simple-to-more-detailed" guideline is being used here.
 - I.e., start with simple scenario on basic scheduling (Section 2.2).
 - Cover remaining details subsequently.

Section 2.5. Scheduling Group Meetings

Section 2.5. Scheduling Group Meetings

- This scenario covers scheduling from a group leader's perspective.

Section 2.5. Scheduling Group Meetings

- This scenario covers scheduling from a group leader's perspective.
- Stylistically, the "user-category" guideline is being used here.

Section 2.5. Scheduling Group Meetings

- This scenario covers scheduling from a group leader's perspective.
- Stylistically, the "user-category" guideline is being used here.
 - I.e., start with scheduling scenario for most common user category (registered user).

Section 2.5. Scheduling Group Meetings

- This scenario covers scheduling from a group leader's perspective.
- Stylistically, the "user-category" guideline is being used here.
 - I.e., start with scheduling scenario for most common user category (registered user).
 - Present subsequent advanced scenarios.

Section 2.6. Admin Functions

Section 2.6. Admin Functions

- **Scenarios for 'Admin' menu commands.**

Section 2.6. Admin Functions

- Scenarios for 'Admin' menu commands.
- Stylistically, things come together naturally here.

Section 2.6. Admin Functions

- Scenarios for 'Admin' menu commands.
- Stylistically, things come together naturally here.
 - Follow the functional command hierarchy.

Section 2.6. Admin Functions

- Scenarios for 'Admin' menu commands.
- Stylistically, things come together naturally here.
 - Follow the functional command hierarchy.
 - Commands for different user category (admin).

Section 2.6. Admin Functions

- Scenarios for 'Admin' menu commands.
- Stylistically, things come together naturally here.
 - Follow the functional command hierarchy.
 - Commands for different user category (admin).
 - Somewhat mundane operations towards end.

Sections 2.7 and 2.8. Options, File, Edit

Sections 2.7 and 2.8. Options, File, Edit

- Again, we're following the "mundane details towards end" guideline.

Sections 2.7 and 2.8. Options, File, Edit

- Again, we're following the "mundane details towards end" guideline.
- These details are important, but not what the Calendar Tool is mainly about.

Sections 2.7 and 2.8. Options, File, Edit

- Again, we're following the "mundane details towards end" guideline.
- These details are important, but not what the Calendar Tool is mainly about.
- The point is, we try to keep the reader engaged without compromising overall organization.

Sections 2.7 and 2.8. Options, File, Edit

- Again, we're following the "mundane details towards end" guideline.
- These details are important, but not what the Calendar Tool is mainly about.
- The point is, we try to keep the reader engaged without compromising overall organization.
- Use your own good judgment for your projects.

Where Things Stand with Milestone 2

Where Things Stand with Milestone 2

- A very rough draft.

Where Things Stand with Milestone 2

- A very rough draft.
- Focus on fundamental functionality.

Where Things Stand with Milestone 2

- A very rough draft.
- Focus on fundamental functionality.
- Error conditions not yet considered.

Where Things Stand with Milestone 2

- A very rough draft.
- Focus on fundamental functionality.
- Error conditions not yet considered.
- Much work yet to do.

Handout on Spec Doc Structure

Handout on Spec Doc Structure

- `index.html` contains linked contents

Handout on Spec Doc Structure

- `index.html` contains linked contents
- Sections 1 through 6 are in the files:

Handout on Spec Doc Structure

- `index.html` contains linked contents
- Sections 1 through 6 are in the files:
 - `intro.html`

Handout on Spec Doc Structure

- `index.html` contains linked contents
- Sections 1 through 6 are in the files:
 - `intro.html`
 - `functional.html`

Handout on Spec Doc Structure

- `index.html` contains linked contents
- Sections 1 through 6 are in the files:
 - `intro.html`
 - `functional.html`
 - `non-functional.html`

Handout on Spec Doc Structure

- `index.html` contains linked contents
- Sections 1 through 6 are in the files:
 - `intro.html`
 - `functional.html`
 - `non-functional.html`
 - `developer-overview.html`

Handout on Spec Doc Structure

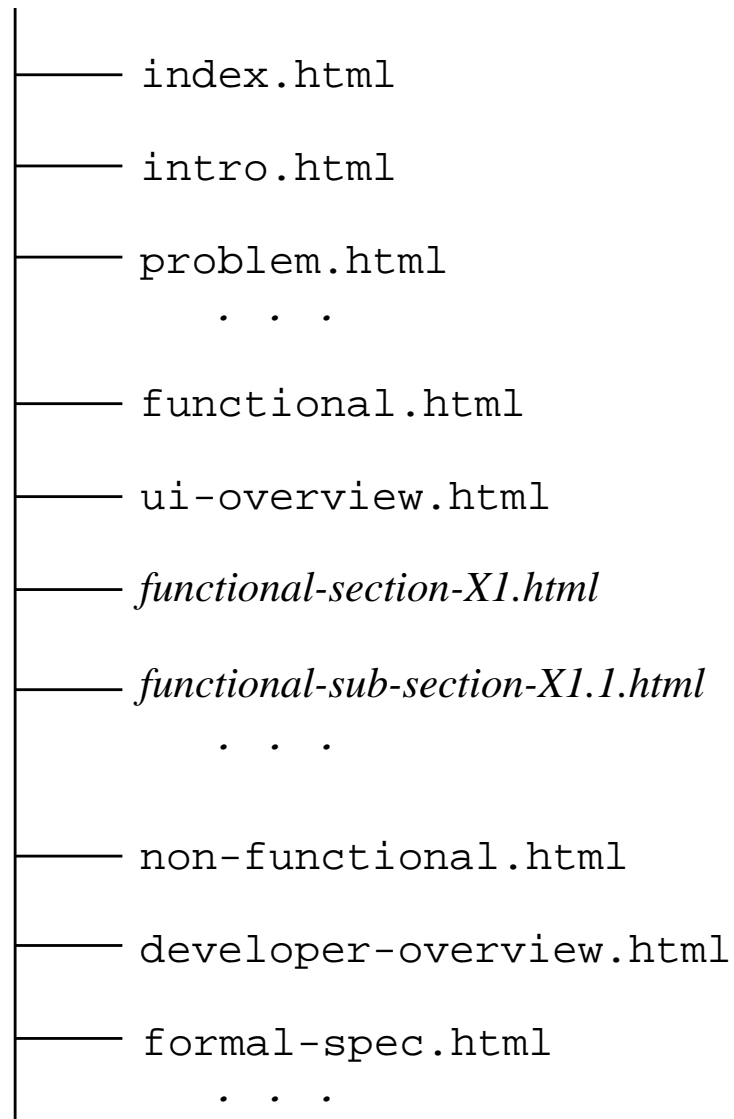
- `index.html` contains linked contents
- Sections 1 through 6 are in the files:
 - `intro.html`
 - `functional.html`
 - `non-functional.html`
 - `developer-overview.html`
 - `formal-spec.html`

Handout on Spec Doc Structure

- `index.html` contains linked contents
- Sections 1 through 6 are in the files:
 - `intro.html`
 - `functional.html`
 - `non-functional.html`
 - `developer-overview.html`
 - `formal-spec.html`
 - `rationale.html`

Doc Structure, cont'd

requirements



Doc Structure, cont'd

- `ui-overview.html` has section 2.1.

Doc Structure, cont'd

- `ui-overview.html` has section 2.1.
- Italic names, with prefix "*functional...*", contain subsections 2.2 through 2.*n*

Doc Structure, cont'd

- `ui-overview.html` has section 2.1.
- Italic names, with prefix "*functional...*", contain subsections 2.2 through 2.*n*
- Italic names stand for an appropriate mnemonic name, e.g., `appt-scheduling` for Sec 2.2.

Doc Structure, cont'd

- Use additional files as appropriate for subsections.

Doc Structure, cont'd

- Use additional files as appropriate for subsections.
- Rule of thumb for separate file is 7 ± 2 screens.

Doc Structure, cont'd

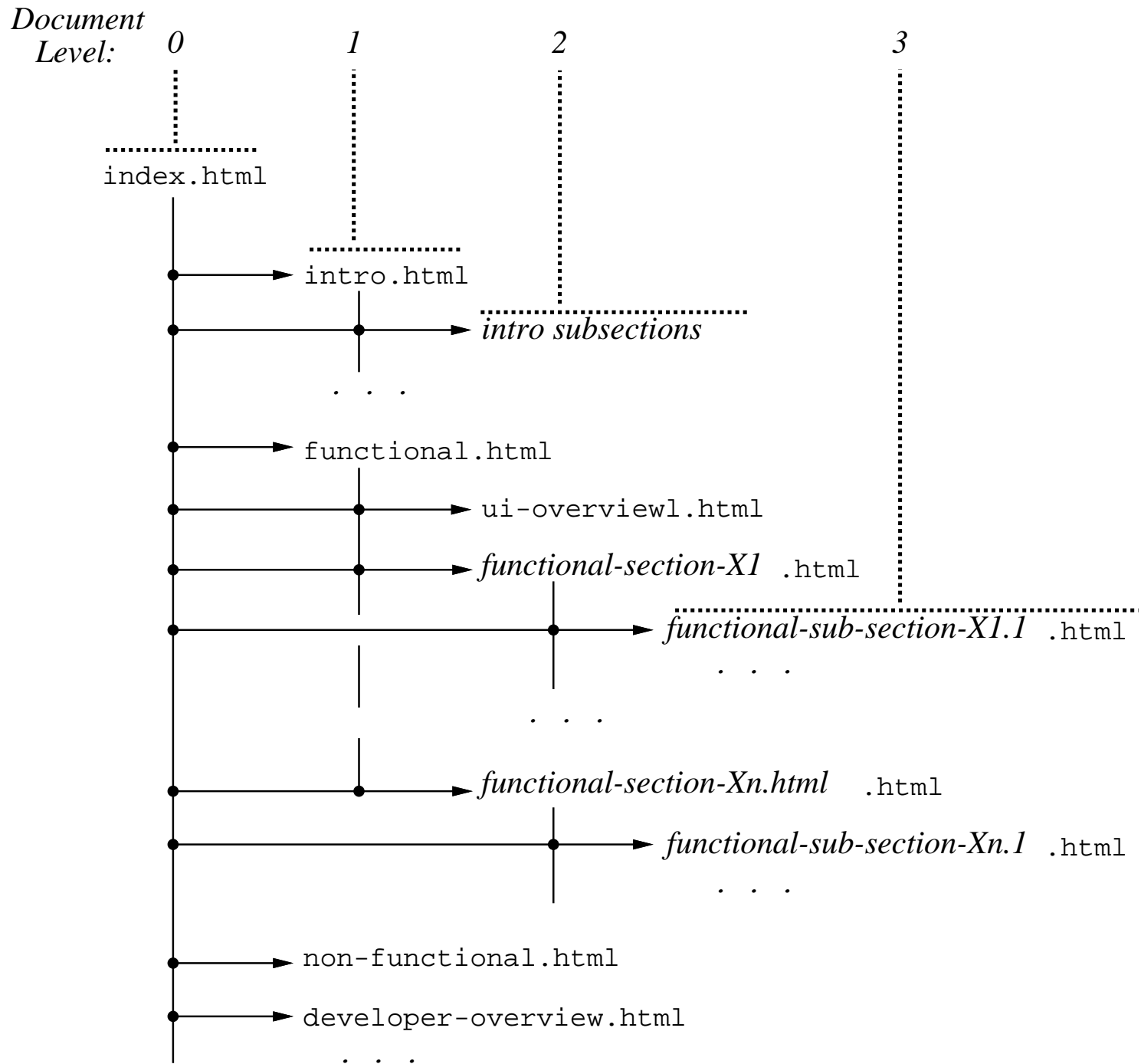
- Use additional files as appropriate for subsections.
- Rule of thumb for separate file is 7 ± 2 screens.
- Structure defined by HTML href links.

Doc Structure, cont'd

- Use additional files as appropriate for subsections.
- Rule of thumb for separate file is $7+ / 2$ screens.
- Structure defined by HTML href links.
- Top-level index has links to all (sub...)sections.

Doc Structure, cont'd

- Use additional files as appropriate for subsections.
- Rule of thumb for separate file is $7+ / 2$ screens.
- Structure defined by HTML `href` links.
- Top-level index has links to all (sub...)sections.
- Section index has links to its subsections only.



Doc Structure, cont'd

- **Hyperlinking based on subsection hierarchy.**

Doc Structure, cont'd

- Hyperlinking based on subsection hierarchy.
- Files at same level have "next", "previous" links.

Doc Structure, cont'd

- Hyperlinking based on subsection hierarchy.
- Files at same level have "next", "previous" links.
- Each file also has "up" link.

Doc Structure, cont'd

- Hyperlinking based on subsection hierarchy.
- Files at same level have "next", "previous" links.
- Each file also has "up" link.
- Each file has "top" link to the index.

Doc Structure, cont'd

- See the online Milestone 2 example in
[http://users.csc.calpoly.edu/~gfisher/
classes/308/examples/milestone2](http://users.csc.calpoly.edu/~gfisher/classes/308/examples/milestone2).

Standard GUI Conventions

Standard GUI Conventions

- Follow these guidelines, or document your own.

Standard GUI Conventions

- Follow these guidelines, or document your own.
- Style is "simple charm", not flash.

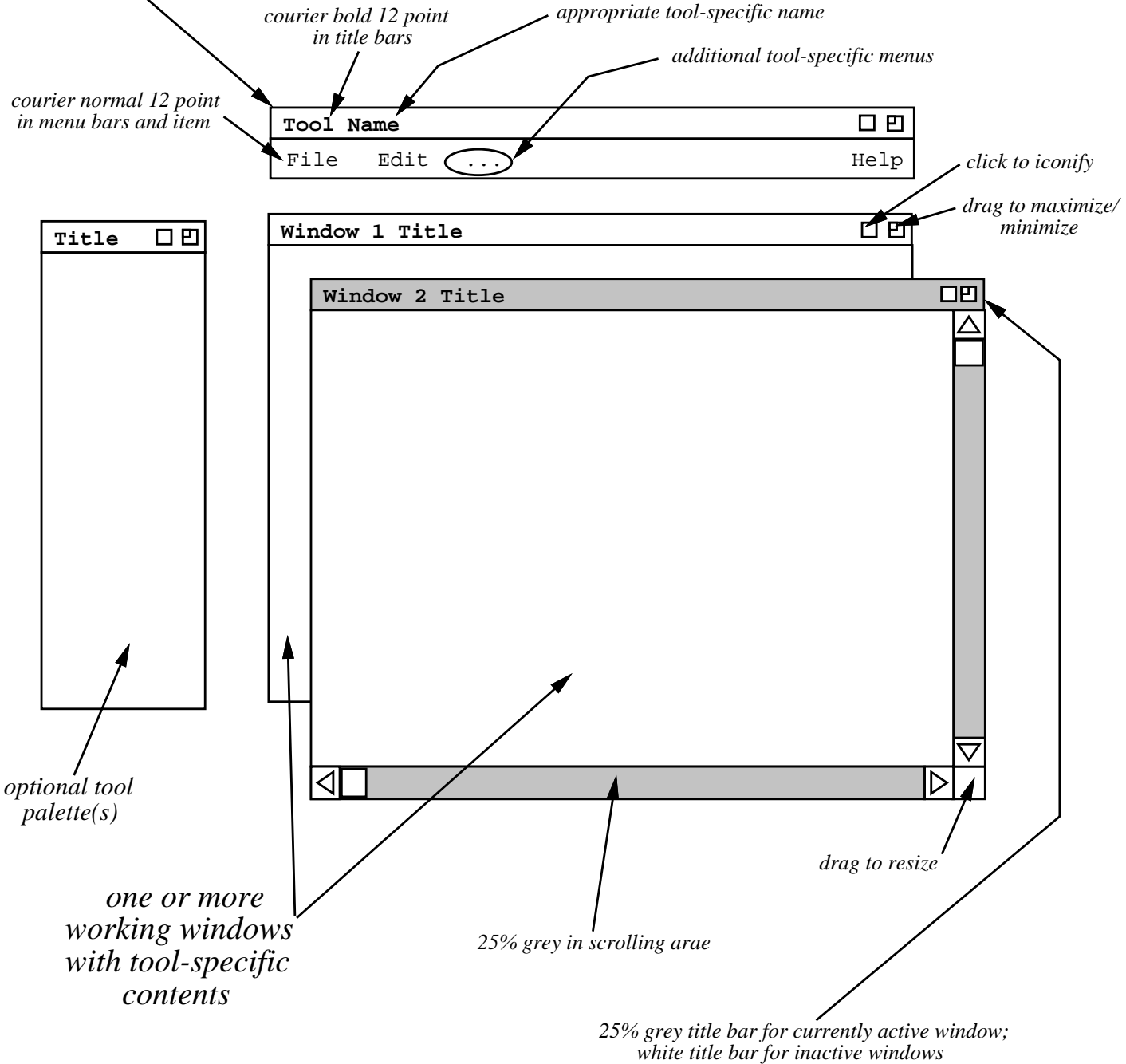
Standard GUI Conventions

- Follow these guidelines, or document your own.
- Style is "simple charm", not flash.
- Emphasizes platform independence.

Standard GUI Conventions

- Follow these guidelines, or document your own.
- Style is "simple charm", not flash.
- Emphasizes platform independence.
- Remember,
don't bog down in minor UI details early on.

Main command menu



GUI Conventions, cont'd

Drawing Editors

GUI Conventions, cont'd

Drawing Editors

- General-purpose drawing tools work fine.

GUI Conventions, cont'd

Drawing Editors

- General-purpose drawing tools work fine.
- Dia is decent for Linux.

GUI Conventions, cont'd

Drawing Editors

- General-purpose drawing tools work fine.
- Dia is decent for Linux.
- Visio is available for Windows, via MSDNAA.

GUI Conventions, cont'd

Drawing Editors

- General-purpose drawing tools work fine.
- Dia is decent for Linux.
- Visio is available for Windows, via MSDNAA.
- Balsamiq may be available free, pending approval.

GUI Conventions, cont'd

Drawing Editors

- General-purpose drawing tools work fine.
- Dia is decent for Linux.
- Visio is available for Windows, via MSDNAA.
- Balsamiq may be available free, pending approval.
- I recommend Pencil over Balsamic or Moqups.

II. Some paper OK for Milestone 2.

II. Some paper OK for Milestone 2.

A. Clearly label paper materials

II. Some paper OK for Milestone 2.

A. Clearly label paper materials

- 1. Figure number and caption.**

II. Some paper OK for Milestone 2.

A. Clearly label paper materials

- 1. Figure number and caption.**
- 2. Title for other notes.**

II. Some paper OK for Milestone 2.

A. Clearly label paper materials

- 1. Figure number and caption.**
- 2. Title for other notes.**

B. Make copies if you want originals.

II. Some paper OK for Milestone 2.

A. Clearly label paper materials

- 1. Figure number and caption.**
- 2. Title for other notes.**

B. Make copies if you want originals.

C. To Fisher's office by 5 PM Friday.

III. General scenario guidelines

III. General scenario guidelines

A. Present scenarios in *tutorial* style.

III. General scenario guidelines

A. Present scenarios in *tutorial* style.

1. Interesting, engaging, ultimately complete story.

III. General scenario guidelines

- A. Present scenarios in *tutorial* style.
 1. Interesting, engaging, ultimately complete story.
 2. A step-by-step presentation.

III. General scenario guidelines

- A. Present scenarios in *tutorial* style.
 1. Interesting, engaging, ultimately complete story.
 2. A step-by-step presentation.

- B. Start with common activities.

General scenario guidelines, cont'd

C. Separate based on user categories.

General scenario guidelines, cont'd

- C. Separate based on user categories.
- D. Stylistic recommendations:

General scenario guidelines, cont'd

- C. Separate based on user categories.
- D. Stylistic recommendations:
 1. Leave mundane details until later.

General scenario guidelines, cont'd

- C. Separate based on user categories.
- D. Stylistic recommendations:
 1. Leave mundane details until later.
 2. Leave error details until later.

IV. Core steps of the scenario process.

IV. Core steps of the scenario process.

A. Describe a user action.

IV. Core steps of the scenario process.

A. Describe a user action.

- 1. In GUIs, it's performed by some gesture.**

IV. Core steps of the scenario process.

A. Describe a user action.

- 1. In GUIs, it's performed by some gesture.**
- 2. Combo of mouse and/or keyboard.**

IV. Core steps of the scenario process.

A. Describe a user action.

- 1. In GUIs, it's performed by some gesture.**
- 2. Combo of mouse and/or keyboard.**
- 3. Most typically, menu item, command button, keystrokes.**

Core steps, cont'd

B. Show system response.

Core steps, cont'd

B. Show system response.

1. Typically appears on screen.

Core steps, cont'd

B. Show system response.

1. Typically appears on screen.
2. Can also be other output medium.

Core steps, cont'd

- B.** Show system response.
 - 1. Typically appears on screen.
 - 2. Can also be other output medium.
 - 3. In some cases, not displayed directly, e.g., saved data store.

Core steps, cont'd

C. Fully describe details of response.

Core steps, cont'd

- C. Fully describe details of response.
 - 1. A prose narrative.

Core steps, cont'd

- C. Fully describe details of response.
 - 1. A prose narrative.
 - 2. All screen components described.

Core steps, cont'd

- C. Fully describe details of response.
 1. A prose narrative.
 2. All screen components described.
 3. All output effects described.

Core steps, cont'd

D. If response is input dialog:

Core steps, cont'd

- D.** If response is input dialog:
 - 1.** Show another picture, filled in.

Core steps, cont'd

- D.** If response is input dialog:
 - 1.** Show another picture, filled in.
 - 2.** Fully describe entered values.

Core steps, cont'd

- D.** If response is input dialog:
 1. Show another picture, filled in.
 2. Fully describe entered values.
 3. For non-atomic interactions:

Core steps, cont'd

- D.** If response is input dialog:
 - 1.** Show another picture, filled in.
 - 2.** Fully describe entered values.
 - 3.** For non-atomic interactions:
 - a.** Simple cases in narrative.

Core steps, cont'd

- D.** If response is input dialog:
 - 1.** Show another picture, filled in.
 - 2.** Fully describe entered values.
 - 3.** For non-atomic interactions.
 - a.** Simple cases in narrative.
 - b.** E.g., toggles or short lists.

Core steps, cont'd

- D.** If response is input dialog:
 - 1.** Show another picture, filled in.
 - 2.** Fully describe entered values.
 - 3.** For non-atomic interactions.
 - a.** Simple cases in narrative.
 - b.** E.g., toggles or short lists.
 - 4.** If input alternatives, cover all cases.

Core steps, cont'd

E. If response is output:

Core steps, cont'd

E. If response is output:

1. One example sufficient if *fully representative*.

Core steps, cont'd

E. If response is output:

1. One example sufficient if *fully representative*.
2. If alternatives, show additional examples and narrative.

V. Milestone 6 example excerpt

V. Milestone 6 example excerpt

A. Illustrates completed scenarios, circa Milestone 6.

V. Milestone 6 example excerpt

- A.** Illustrates completed scenarios, circa Milestone 6.
- B.** This detail not expected for earlier Milestones.

V. Milestone 6 example excerpt

- A.** Illustrates completed scenarios, circa Milestone 6.
- B.** This detail not expected for earlier Milestones.
- C.** It's what you are working towards.

VI. Core steps illustrated

VI. Core steps illustrated

A. Describe a user action:

Section 2.2, paragraph 2

VI. Core steps illustrated

A. Describe a user action:

Section 2.2, paragraph 2

B. Show the resulting screen:

Figure 6

VI. Core steps illustrated

A. Describe a user action:

Section 2.2, paragraph 2

B. Show the resulting screen:

Figure 6

C. Describe screen contents fully:

starting paragraph 2

Schedule an Appointment □ □

Title:

Date: Start Time:

End Date: Duration: hr min

Recurring? Interval:

Category: Security:

Location: Priority:

Remind?

Details:

△
□
▬
▽

OK Clear Cancel

Core steps illustrated, cont'd

D. If screen is input dialog:

1. Show another filled-in screen.

Figure 7

2. Fully describe entered values.

paragraph 5

Schedule an Appointment

Title:

Date: Start Time:

End Date: Duration: ^{hr} ^{min}

Recurring? Interval:

Category: Security:

Location: Priority:

Remind?

Details:

Core steps illustrated, cont'd

3. Cover all non-atomic interactions

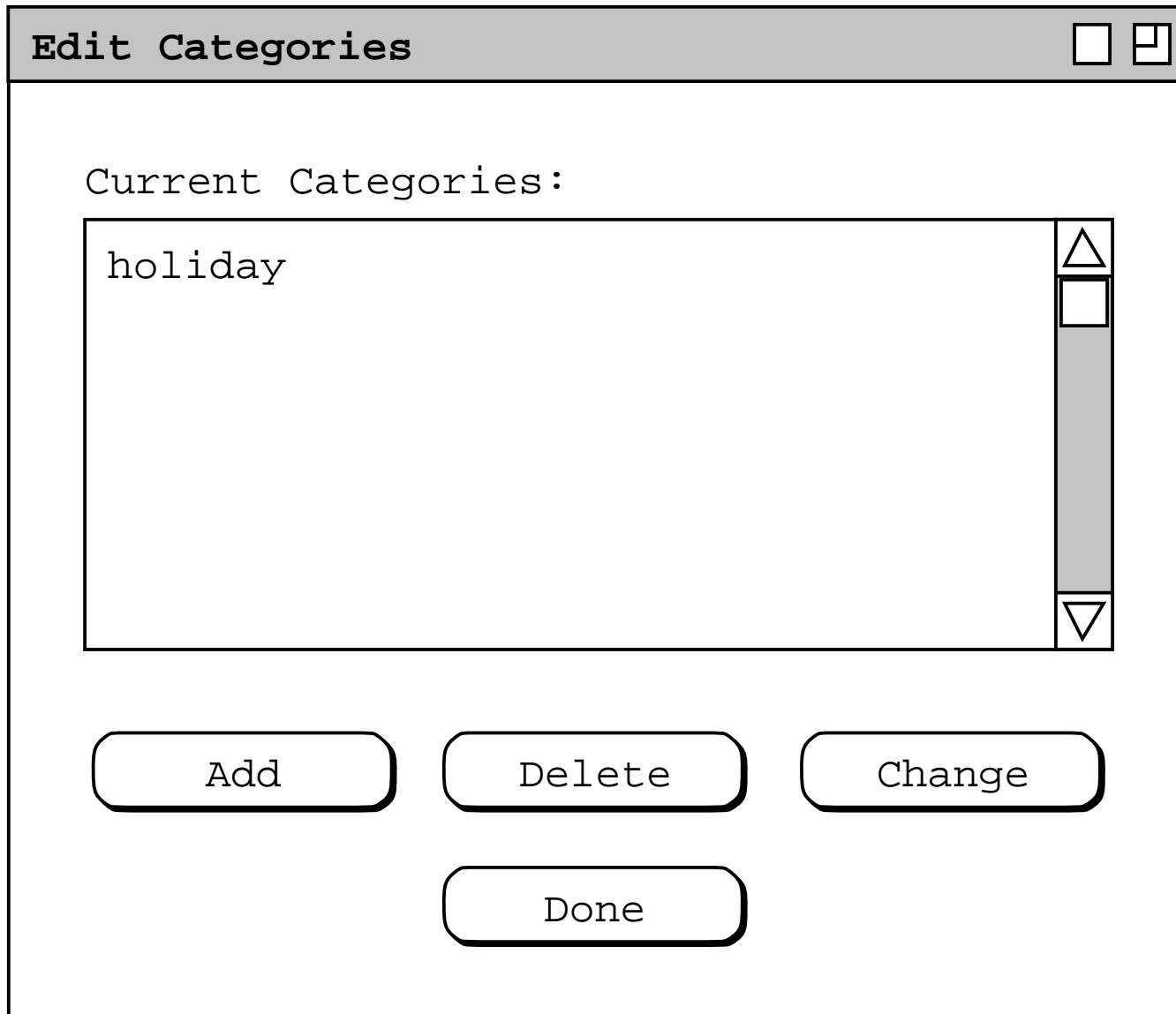
Figures 8 -11


narrative

paragraph 6 page 10.

Category:

<input checked="" type="checkbox"/> none
<input type="checkbox"/> holiday
<input type="checkbox"/> Edit ...



Add Category 

Category Name:

Color: ▼

The image shows a dialog box titled "Add Category". It contains a text input field for "Category Name" with the text "outside appt". Below it is a "Color:" label followed by a dropdown menu. The dropdown menu is open, showing a list of color options: Black, Brown, Red, Orange, Yellow, Green, Blue, and Purple. The "Brown" option is currently selected and highlighted in a dark red background. At the bottom of the dialog box, there are two buttons: "OK" on the left and "Cancel" on the right.

Add Category

Category Name: outside appt

Color: Black
Brown
Red
Orange
Yellow
Green
Blue
Purple

OK Cancel

Core steps illustrated, cont'd

a. Simple cases, narrative only.

b. E.g., toggles, short lists.

paragraph 1-3

Core steps illustrated, cont'd

4. For input alternatives, show additional fill-in's

a. *Screen: Figure 12*

5. *Narrative: paragraphs 8 thru 10*

Schedule an Appointment

Title:

Start Date: Start Time:

End Date: Duration: hr min

Recurring? Interval: S M T W Th F S

Category: Security:

Location: Priority:

Remind?

Details:

Core steps illustrated, cont'd

E. For output screens:

1. One case, if adequately representative.

Figures 20 and 23,

Weekly Agenda □ □						
◀ Week of September 6 - 12, 2015 ▶						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
6	7	8	9	10	11	12
	9-10 AM O 10-11 AM 11 AM-12 3-5 PM So	8-9 AM Ra 9 AM-5 PM	8-9 AM St 8-9:30 AM 9-10 AM O 10-11 AM 11 AM-12 12-1:30 P 2-3 PM Co 2-3:30 PM 2:30-4:30	8-9 AM Ra 9-10:30 A 10:30-12 1:15-2 PM 3-5 PM So	8-9:30 AM 9-10 AM O 10-11 AM 11 AM-12	

Yearly Calendar □ □																				
◀ 2015 ▶																				
Jan							Feb							Mar						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
				1	2	3	1	2	3	4	5	6	7	1	2	3	4	5	6	7
4	5	6	7	8	9	10	8	9	10	11	12	13	14	8	9	10	11	12	13	14
11	12	13	14	15	16	17	15	16	17	18	19	20	21	15	16	17	18	19	20	21
18	19	20	21	22	23	24	22	23	24	25	26	27	28	22	23	24	25	26	27	28
25	26	27	28	29	30	31								29	30	31				
Apr							May							Jun						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
			1	2	3	4						1	2		1	2	3	4	5	6
5	6	7	8	9	10	11	3	4	5	6	7	8	9	7	8	9	10	11	12	13
12	13	14	15	16	17	18	10	11	12	13	14	15	16	14	15	16	17	18	19	20
19	20	21	22	23	24	25	17	18	19	20	21	22	23	21	22	23	24	25	26	27
26	27	28	29	30			24	25	26	27	28	29	30	28	29	30				
							31													
Jul							Aug							Sep						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
			1	2	3	4							1			1	2	3	4	5
5	6	7	8	9	10	11	2	3	4	5	6	7	8	6	7	8	9	10	11	12
12	13	14	15	16	17	18	9	10	11	12	13	14	15	13	14	15	16	17	18	19
19	20	21	22	23	24	25	16	17	18	19	20	21	22	20	21	22	23	24	25	26
26	27	28	29	30	31		23	24	25	26	27	28	29	27	28	29	30			
							30	31												
Oct							Nov							Dec						
S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S
				1	2	3	1	2	3	4	5	6	7			1	2	3	4	5
4	5	6	7	8	9	10	8	9	10	11	12	13	14	6	7	8	9	10	11	12
11	12	13	14	15	16	17	15	16	17	18	19	20	21	13	14	15	16	17	18	19
18	19	20	21	22	23	24	22	23	24	25	26	27	28	20	21	22	23	24	25	26
25	26	27	28	29	30	31	29	30						27	28	29	30	31		

Core steps illustrated, cont'd

2. For alternatives, show more screens + narrative.

a. *Figures 13 -17*

b. *Figures 18 and 19*

c. *Figures 21 and 22.*

Daily Agenda □ □	
◀ Wednesday, September 9, 2015 ▶	
8 AM	Staff Meeting
9 AM	Office Hours
10 AM	Data Structures Lecture
11 AM	Data Structures Lab
12 PM	<u>Lunch with Microsoft</u>
1 PM
2 PM <u>CAD Research Project Meeting</u>
3 PM
4 PM
5 PM	

Daily Agenda □ ▢	
◀ Thursday, September 10, 2015 ▶	
8AM	<u>Racket Ball</u>
9AM	<u>Picnic Day Committee</u>
10AM	----- <u>Software Engineering Colloquium</u>
11AM	
12PM	
1PM	----- <u>Office Hours</u>
2PM	
3PM	<u>Software Engineering Graduate Seminar</u>
4PM	
5PM	

Daily Agenda □ □	
◀ Wednesday, September 9, 2015 ▶	
12:00 AM	
12:30 AM	
1:00 AM	
1:30 AM	
2:00 AM	
2:30 AM	
3:00 AM	
3:30 AM	
4:00 AM	
4:30 AM	
5:00 AM	
5:30 AM	
6:00 AM	
6:30 AM	
7:00 AM	
7:30 AM	
8:00 AM	8-9 AM Staff Meeting
8:30 AM	
9:00 AM	9-10 Office Hours
9:30 AM	
10:00 AM	10-11 AM Data Structures Lecture
10:30 AM	
11:00 AM	11 AM-12 PM Data Structures Lab
11:30 AM	
12:00 PM	12-1:30 PM Lunch with Microsoft
12:30 PM	
1:00 PM	
1:30 PM	
2:00 PM	
2:30 PM	2:30-4:30 PM CAD Research Project Meeting
3:00 PM	
3:30 PM	
4:00 PM	
4:30 PM	
5:00 PM	
5:30 PM	
6:00 PM	
6:30 PM	
7:00 PM	
7:30 PM	
8:00 PM	
8:30 PM	
9:00 PM	
9:30 PM	
10:00 PM	
10:30 PM	
11:00 PM	
11:30 PM	

Daily Agenda □ □			
◀ Wednesday, September 9, 2015 ▶			
8 AM	Staff Meeting	TA Meeting	
9 AM	Office Hours	↓	
10 AM	Data Structures Lectur		
11 AM	Data Structures Lab		
12 PM	Lunch with Microsoft		
1 PM	↓		
2 PM	College Meeting	Special Colloquium	
3 PM		↓	↓
4 PM			↓
5 PM			

Daily Agenda □ □	
◀ Wednesday, September 9, 2015 ▶	
8 AM	8-9 AM Staff Meeting 8-9:30 AM TA Meeting
9 AM	9-10 AM Office Hours
10 AM	10-11 AM Data Structures Lecture
11 AM	11 AM-12 PM Data Structures Lab
12 PM	12-1:30 PM Lunch with Microsoft
1 PM	
2 PM	2-3 PM College Meeting 2-3:30 PM Special Colloquium 2:30-4:30 PM CAD Research Project Meeting
3 PM	
4 PM	
5 PM	

VII. Other scenario presentation issues

A. Ensure complete coverage.

1. Cover all interactions at least once.
2. Provide at least two examples of all input dialogs, *initial and filled in*.
3. Provide at least one example of all outputs.

Other issues, cont'd

B. Interface layout details.

1. Not purely look and feel.

2. E.g., 2.3.1.1, *paragraph 3*

Other issues, cont'd

C. Avoid unnecessarily repetition.

1. Refer to pictures or narrative of common functionality.
2. E.g., description of weekly view options, 2.3.1.2 *paragraph 10*

Other issues, cont'd

- D. Scenarios flow by building on information presented earlier.
 1. Refer to preceding sections.
 2. State assumed user actions.

Other issues, cont'd

3. E.g.,

Sec 2.3.1.1 paragraph 2

and

all the figures in Section 2.3.

Other issues, cont'd

E. Where necessary, gritty details.

1. When functionality is complicated or non-obvious.

2. E.g., overlapping items

Figures 16-19,

Other issues, cont'd

3. Use good judgment, 7+/-2 rule to defer details.
 - a. E.g., 2.3.1.1 *paragraph 8* .
 - b. 2.3.1.5 *Last paragraph*
 - c. In Section 2.2, details of recurring items deferred to Section 2.4.

VIII. Interface style issues.

- A.** Be simple and consistent.
- B.** Use interface forms that end users can easily understand.
- C.** Provide interface options to allow user to select among alternate forms.

IX. More on “Interesting & Engaging Stories”

A. Purpose is two-fold:

1. maintain reader’s interest
2. provide overall context and continuity

B. Point is not to entertain like a novel.

1. Humor should be used sparingly, *if at all*.
2. “*Story*” should stick to the facts.

“Interesting and Engaging Stories”, cont’d

C. Story line sketch for Calendar Tool scenarios:

“Interesting and Engaging Stories”, cont’d

- C.** Story line sketch for Calendar Tool scenarios:
 - 1.** User schedules a couple appointments.

“Interesting and Engaging Stories”, cont’d

- C.** Story line sketch for Calendar Tool scenarios:
 - 1.** User schedules a couple appointments.
 - 2.** User views calendar in various ways.

“Interesting and Engaging Stories”, cont’d

C. Story line sketch for Calendar Tool scenarios:

- 1.** User schedules a couple appointments.
- 2.** User views calendar in various ways.
- 3.** User schedules some other kinds of items.

“Interesting and Engaging Stories”, cont’d

C. Story line sketch for Calendar Tool scenarios:

- 1. User schedules a couple appointments.**
- 2. User views calendar in various ways.**
- 3. User schedules some other kinds of items.**
- 4. User deals with finer points of scheduling.**

“Interesting and Engaging Stories”, cont’d

C. Story line sketch for Calendar Tool scenarios:

1. User schedules a couple appointments.
2. User views calendar in various ways.
3. User schedules some other kinds of items.
4. User deals with finer points of scheduling.
5. Admin user performs specialized functions.

“Interesting and Engaging Stories”, cont’d

C. Story line sketch for Calendar Tool scenarios:

1. User schedules a couple appointments.
2. User views calendar in various ways.
3. User schedules some other kinds of items.
4. User deals with finer points of scheduling.
5. Admin user performs specialized functions.
6. User sets calendar options.

“Interesting and Engaging Stories”, cont’d

D. Story line sketch for CSTutor

“Interesting and Engaging Stories”, cont’d

- D.** Story line sketch for CSTutor
 - 1.** Instructor Builds Simple Lesson

“Interesting and Engaging Stories”, cont’d

- D. Story line sketch for CSTutor**
 - 1. Instructor Builds Simple Lesson**
 - 2. Student Views Simple Lesson**

“Interesting and Engaging Stories”, cont’d

D. Story line sketch for CSTutor

- 1. Instructor Builds Simple Lesson**
- 2. Student Views Simple Lesson**
- 3. Instructor Builds Advanced Lesson**

“Interesting and Engaging Stories”, cont’d

D. Story line sketch for CSTutor

- 1. Instructor Builds Simple Lesson**
- 2. Student Views Simple Lesson**
- 3. Instructor Builds Advanced Lesson**
- 4. Student Views Advanced Lesson**

“Interesting and Engaging Stories”, cont’d

D. Story line sketch for CSTutor

- 1. Instructor Builds Simple Lesson**
- 2. Student Views Simple Lesson**
- 3. Instructor Builds Advanced Lesson**
- 4. Student Views Advanced Lesson**
- 5. Instructors and Students Chat and Interact**

“Interesting and Engaging Stories”, cont’d

D. Story line sketch for CSTutor

- 1. Instructor Builds Simple Lesson**
- 2. Student Views Simple Lesson**
- 3. Instructor Builds Advanced Lesson**
- 4. Student Views Advanced Lesson**
- 5. Instructors and Students Chat and Interact**
- 6. Students View their Stats**

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

- 1.** Instructor downloads and sets up roster.

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

- 1. Instructor downloads and sets up roster.**
- 2. Instructor adds items and students.**

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

1. Instructor downloads and sets up roster.
2. Instructor adds items and students.
3. Instructor views charts and graphs.

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

1. Instructor downloads and sets up roster.
2. Instructor adds items and students.
3. Instructor views charts and graphs.
4. Instructor adds gradesheet details.

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

1. Instructor downloads and sets up roster.
2. Instructor adds items and students.
3. Instructor views charts and graphs.
4. Instructor adds gradesheet details.
5. Student Views and Predicts Grades.

“Interesting and Engaging Stories”, cont’d

E. Story line sketch for Grader:

1. Instructor downloads and sets up roster.
2. Instructor adds items and students.
3. Instructor views charts and graphs.
4. Instructor adds gradesheet details.
5. Student Views and Predicts Grades.
6. Student views historical information.

“Interesting and Engaging Stories”, cont’d

F. Story line sketch for Scheduler

“Interesting and Engaging Stories”, cont’d

- F.** Story line sketch for Scheduler
 - 1.** Instructor Sets Up Preferences

“Interesting and Engaging Stories”, cont’d

F. Story line sketch for Scheduler

- 1. Instructor Sets Up Preferences**
- 2. Admin Creates Simple Schedule**

“Interesting and Engaging Stories”, cont’d

F. Story line sketch for Scheduler

- 1. Instructor Sets Up Preferences**
- 2. Admin Creates Simple Schedule**
- 3. Admin Edits Data Databases**

“Interesting and Engaging Stories”, cont’d

F. Story line sketch for Scheduler

1. Instructor Sets Up Preferences
2. Admin Creates Simple Schedule
3. Admin Edits Data Databases
4. Admin Creates More Complicated Schedule

“Interesting and Engaging Stories”, cont’d

F. Story line sketch for Scheduler

- 1. Instructor Sets Up Preferences**
- 2. Admin Creates Simple Schedule**
- 3. Admin Edits Data Databases**
- 4. Admin Creates More Complicated Schedule**
- 5. Admin Deals with Scheduling Constraints**

“Interesting and Engaging Stories”, cont’d

F. Story line sketch for Scheduler

- 1. Instructor Sets Up Preferences**
- 2. Admin Creates Simple Schedule**
- 3. Admin Edits Data Databases**
- 4. Admin Creates More Complicated Schedule**
- 5. Admin Deals with Scheduling Constraints**
- 6. Student Views and Comments on Schedules**

“Interesting and Engaging Stories”, cont’d

G. Story line sketch for TestTool:

“Interesting and Engaging Stories”, cont’d

- G.** Story line sketch for TestTool:
 - 1.** Instructor creates a simple test.

“Interesting and Engaging Stories”, cont’d

G. Story line sketch for TestTool:

1. Instructor creates a simple test.
2. Instructor edits question database.

“Interesting and Engaging Stories”, cont’d

G. Story line sketch for TestTool:

1. Instructor creates a simple test.
2. Instructor edits question database.
3. Instructor creates more complicated test.

“Interesting and Engaging Stories”, cont’d

G. Story line sketch for TestTool:

1. Instructor creates a simple test.
2. Instructor edits question database.
3. Instructor creates more complicated test.
4. Student takes test.

“Interesting and Engaging Stories”, cont’d

G. Story line sketch for TestTool:

1. Instructor creates a simple test.
2. Instructor edits question database.
3. Instructor creates more complicated test.
4. Student takes test.
5. Instructor grades test.

“Interesting and Engaging Stories”, cont’d

G. Story line sketch for TestTool:

1. Instructor creates a simple test.
2. Instructor edits question database.
3. Instructor creates more complicated test.
4. Student takes test.
5. Instructor grades test.
6. Instructor manages tests and question DB.

IX. Concrete data underlying scenarios.

IX. Concrete data underlying scenarios.

A. Consistent example data.

IX. Concrete data underlying scenarios.

A. Consistent example data.

- 1. Extensive enough to support all scenarios.**

IX. Concrete data underlying scenarios.

A. Consistent example data.

- 1. Extensive enough to support all scenarios.**
- 2. But, no more expansive than necessary.**

IX. Concrete data underlying scenarios.

A. Consistent example data.

- 1. Extensive enough to support all scenarios.**
- 2. But, no more expansive than necessary.**
- 3. Exemplify variety of realistic examples.**

IX. Concrete data underlying scenarios.

A. Consistent example data.

- 1. Extensive enough to support all scenarios.**
- 2. But, no more expansive than necessary.**
- 3. Exemplify variety of realistic examples.**
- 4. Typically, no single scenario shows all data.**

IX. Concrete data underlying scenarios.

A. Consistent example data.

- 1. Extensive enough to support all scenarios.**
- 2. But, no more expansive than necessary.**
- 3. Exemplify variety of realistic examples.**
- 4. Typically, no single scenario shows all data.**
- 5. Appendix can show complete content.**

Concrete underlying data, cont'd

- B.** For data collections, scenarios organized into *data editing* and *data viewing*.

Concrete underlying data, cont'd

- B. For data collections, scenarios organized into *data editing* and *data viewing*.
 1. Data-editing covers add, modify, delete.

Concrete underlying data, cont'd

- B.** For data collections, scenarios organized into *data editing* and *data viewing*.
 - 1.** Data-editing covers add, modify, delete.
 - 2.** Data-viewing scenarios cover search, display.

Concrete underlying data, cont'd

- B. For data collections, scenarios organized into *data editing* and *data viewing*.
 1. Data-editing covers add, modify, delete.
 2. Data-viewing scenarios cover search, display.
 3. Need sufficient representative examples.

Concrete underlying data, cont'd

4. Show representative data being added,

Concrete underlying data, cont'd

4. Show representative data being added,
5. Then say
"The user now proceeds to add more"

Concrete underlying data, cont'd

4. Show representative data being added
5. Then say
"The user now proceeds to add more"
6. Subsequent scenarios show the *same data*.

Concrete underlying data, cont'd

C. The point is *continuity* through story line.

Concrete underlying data, cont'd

- C. The point is *continuity* through story line.
 1. Early scenarios show data being created.

Concrete underlying data, cont'd

- C. The point is *continuity* through story line.
 1. Early scenarios show data being created.
 2. Then scenarios show same data modified.

Concrete underlying data, cont'd

- C. The point is *continuity* through story line.
 1. Early scenarios show data being created.
 2. Then scenarios show same data modified.
 3. After that, scenarios present viewing.

Concrete underlying data, cont'd

- C. The point is *continuity* through story line.
 1. Early scenarios show data being created.
 2. Then scenarios show same data modified.
 3. After that, scenarios present viewing.
 4. In some cases, viewing scenarios may come first, before editing details.

Concrete underlying data, cont'd

a. Narrative says something like

"The following scenarios assume"

Concrete underlying data, cont'd

a. Narrative says something like

"The following scenarios assume"

b. Continuity maintained by having subsequent editing scenarios use *same data* that appeared earlier.

X. Data examples for the 308 projects.

X. Data examples for the 308 projects.

A. For this year's 308 projects.

X. Data examples for the 308 projects.

A. For this year's 308 projects.

- 1. To help with continuity among scenarios.**

X. Data examples for the 308 projects.

A. For this year's 308 projects.

- 1.** To help with continuity among scenarios.
- 2.** Covers major stuff, but not all details.

Project data examples, cont'd

- B.** For Calendar Tool, calendar examples for a number of users, and for each of the databases.

Project data examples, cont'd

- B.** For Calendar Tool, calendar examples for a number of users, and for each of the databases.
 - 1.** Main example is work calendar for one user.

Project data examples, cont'd

- B.** For Calendar Tool, calendar examples for a number of users, and for each of the databases.
 - 1.** Main example is work calendar for one user.
 - 2.** Also smaller examples for other calendars.

Project data examples, cont'd

- B.** For Calendar Tool, calendar examples for a number of users, and for each of the databases.
 1. Main example is work calendar for one user.
 2. Also smaller examples for other calendars.
 3. Also example calendars for other users.

Project data examples, cont'd

- B.** For Calendar Tool, calendar examples for a number of users, and for each of the databases.
 1. Main example is work calendar for one user.
 2. Also smaller examples for other calendars.
 3. Also example calendars for other users.
 4. One full example for each database.

Project data examples, cont'd

- B.** For Calendar Tool, calendar examples for a number of users, and for each of the databases.
 1. Main example is work calendar for one user.
 2. Also smaller examples for other calendars.
 3. Also example calendars for other users.
 4. One full example for each database.
 5. Appendix with complete example content.

Project data examples, cont'd

C. Underlying data for the CSTutor:

Project data examples, cont'd

- C. Underlying data for the CSTutor:
 1. Example lessons and student database.

Project data examples, cont'd

- C. Underlying data for the CSTutor:
 1. Example lessons and student database.
 2. A single main lesson used in most scenarios.

Project data examples, cont'd

- C. Underlying data for the CSTutor:
 1. Example lessons and student database.
 2. A single main lesson used in most scenarios.
 3. Shells for a number of additional lessons.

Project data examples, cont'd

- C. Underlying data for the CSTutor:
 1. Example lessons and student database.
 2. A single main lesson used in most scenarios.
 3. Shells for a number of additional lessons.
 4. Example student user database, for scenarios on student/instructor interaction and student stats.

Project data examples, cont'd

D. Underlying data for the Grader:

Project data examples, cont'd

D. Underlying data for the Grader:

1. Example student roster with 20+ students.

Project data examples, cont'd

D. Underlying data for the Grader:

1. Example student roster with 20+ students.
2. A single main gradesheet used in most scenarios.

Project data examples, cont'd

D. Underlying data for the Grader:

1. Example student roster with 20+ students.
2. A single main gradesheet used in most scenarios.
3. Main gradesheet ample to illustrate prediction and graphics.

Project data examples, cont'd

D. Underlying data for the Grader:

1. Example student roster with 20+ students.
2. A single main gradesheet used in most scenarios.
3. Main gradesheet ample to illustrate prediction and graphics.
4. Additional smaller gradesheet examples for scenarios on grade trends and gradebook archiving.

Project data examples, cont'd

E. Underlying data for the Scheduler:

Project data examples, cont'd

- E.** Underlying data for the Scheduler:
 - 1.** Main example schedule and resources from recent CS department term.

Project data examples, cont'd

- E.** Underlying data for the Scheduler:
 - 1.** Main example schedule and resources from recent CS department term.
 - 2.** Main schedule example to illustrate generating and editing a good schedule.

Project data examples, cont'd

- E.** Underlying data for the Scheduler:
1. Main example schedule and resources from recent CS department term.
 2. Main schedule example to illustrate generating and editing a good schedule.
 3. Smaller example schedules for specialized scheduling constraints and preferences.

Project data examples, cont'd

E. Underlying data for the Scheduler:

1. Main example schedule and resources from recent CS department term.
2. Main schedule example to illustrate generating and editing a good schedule.
3. Smaller example schedules for specialized scheduling constraints and preferences.
4. Ample smaller examples to illustrate scheduling alternatives.

Project data examples, cont'd

F. Underlying data for the TestTool:

Project data examples, cont'd

F. Underlying data for the TestTool:

1. Test questions from recent classes, with two or three examples of each question type.

Project data examples, cont'd

F. Underlying data for the TestTool:

1. Test questions from recent classes, with two or three examples of each question type.
2. Main test example used in most scenarios.

Project data examples, cont'd

F. Underlying data for the TestTool:

1. Test questions from recent classes, with two or three examples of each question type.
2. Main test example used in most scenarios.
3. Smaller test examples for details of test gen.

Project data examples, cont'd

F. Underlying data for the TestTool:

1. Test questions from recent classes, with two or three examples of each question type.
2. Main test example used in most scenarios.
3. Smaller test examples for details of test gen.
4. Ample number of questions to illustrate test gen alternatives.

XI. Screen maps.

XI. Screen maps.

A. Potentially helpful high-level view of GUI.

XI. Screen maps.

- A.** Potentially helpful high-level view of GUI.
- B.** Consists of thumbnails in cascading tree of command selection.

Screen maps, cont'd

C. Calendar Tool samples in notes.

Screen maps, cont'd

C. Calendar Tool samples in notes.

1. Online, each thumbnail can be a link into requirements.

Screen maps, cont'd

C. Calendar Tool samples in notes.

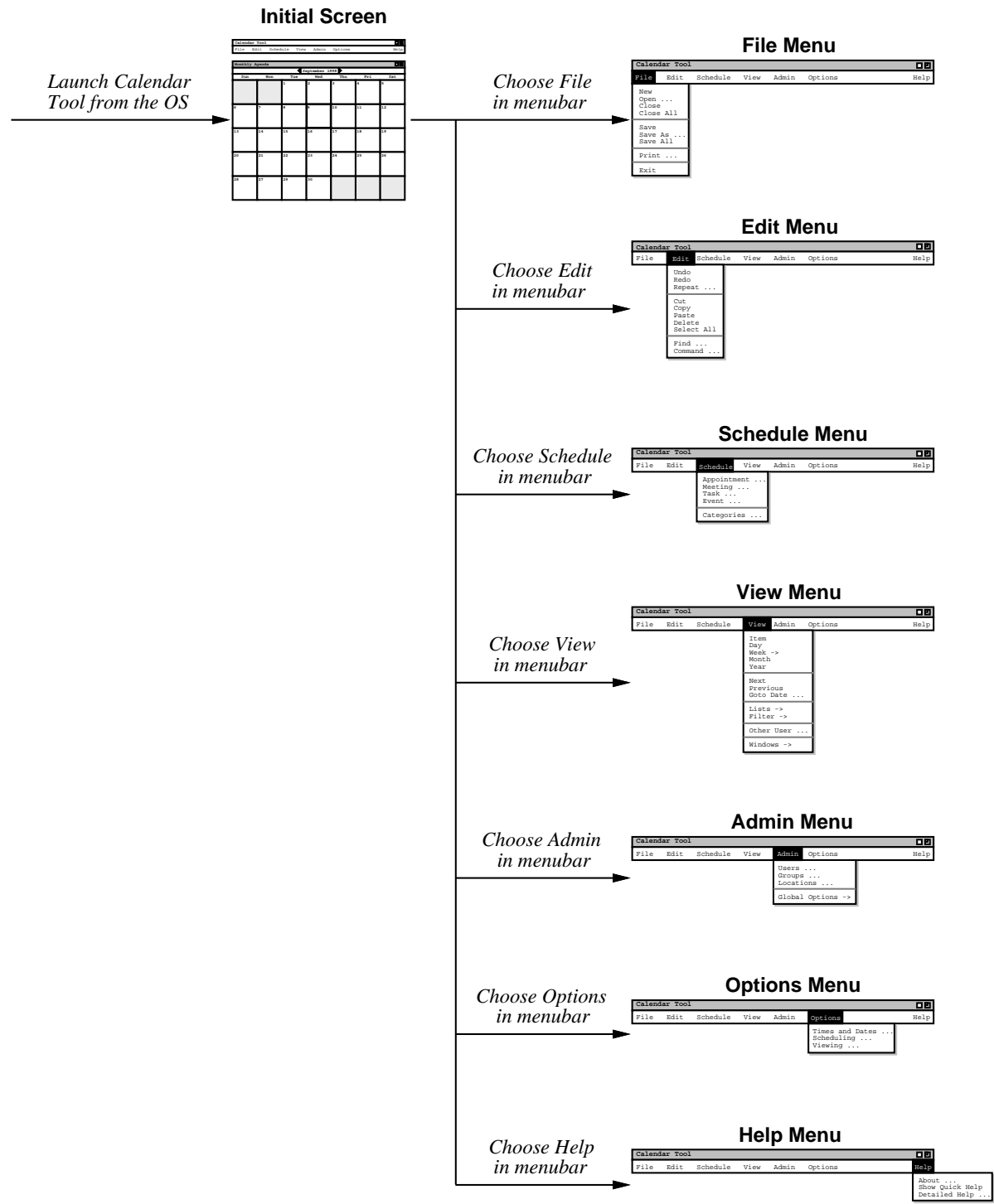
1. Online, each thumbnail can be a link into requirements.
2. Physically, I've seen them spread across the walls of multiple rooms or hallways.

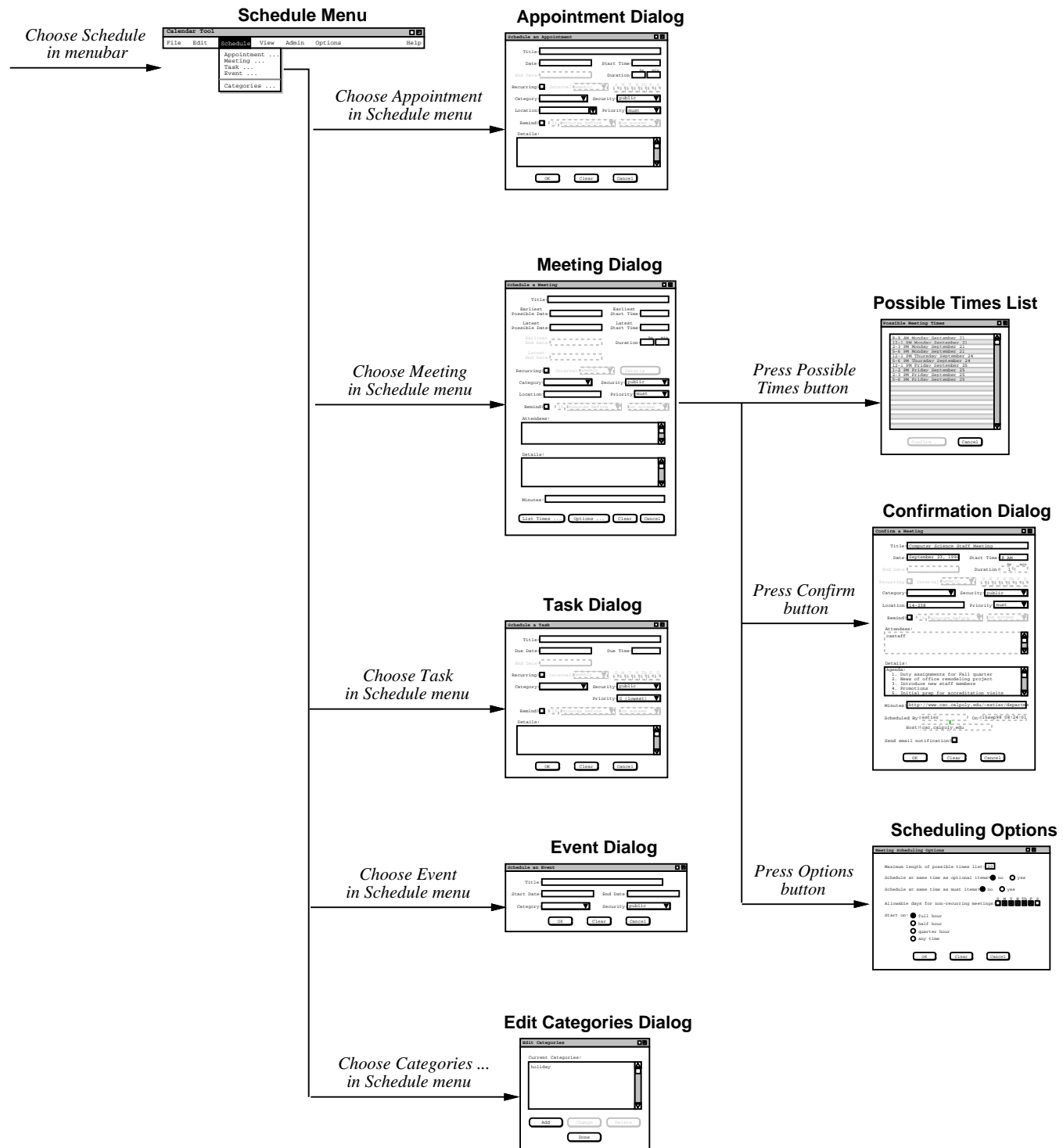
Screen maps, cont'd

C. Calendar Tool samples in notes.

1. Online, each thumbnail can be a link into requirements.
2. Physically, I've seen them spread across the walls of multiple rooms or hallways.

D. Screen maps *not required* for CSC 308.





XII. A view of requirements evolution

XII. A view of requirements evolution

A. SVN log report and snapshots.

XII. A view of requirements evolution

A. SVN log report and snapshots.

B. Reported by 'svn log'.

- 1.** Bookkeeping at top.

- 2.** Versions r1 through r8.

Requirements evolution, cont'd

3. Log messages from '-m' argument to 'svn commit'.

Requirements evolution, cont'd

3. Log messages from '-m' argument to 'svn commit'.
4. History for "menus.ai".

Requirements evolution, cont'd

3. Log messages from '-m' argument to 'svn commit'.
4. History for "menus.ai".
5. For images, checked in source file and JPEG.

Requirements evolution, cont'd

C. Excerpts from SVN log report:

```
-----  
r8 | gfisher | 2014-11-26 15:01:04
```

```
Replaced 'Admin Global Options' with  
'Central Host', 'List Admins', and  
'Login'. Also added 'View Today', 'View  
Windows Close', and 'File Save Config'.
```

```
-----  
...  
-----
```

```
r1 | gfisher | 2014-10-08 14:51:35  
Initial checkin.
```

D. Here are a couple screen shots:

Version 1.1:

