Project Part #3: Design

Due: Friday November 15 2024, at 11:59 pm

Weight: 13% of course grade

0. Cover Page & 1. Table of Contents

Cover Page - should contain the following items:

- Department, university
- Project title
- Team number [also, optional, Team name]
- Team members
- Instructors
- External advisor(s)
- Date

Table of contents

2. Abstract & 3. Introduction

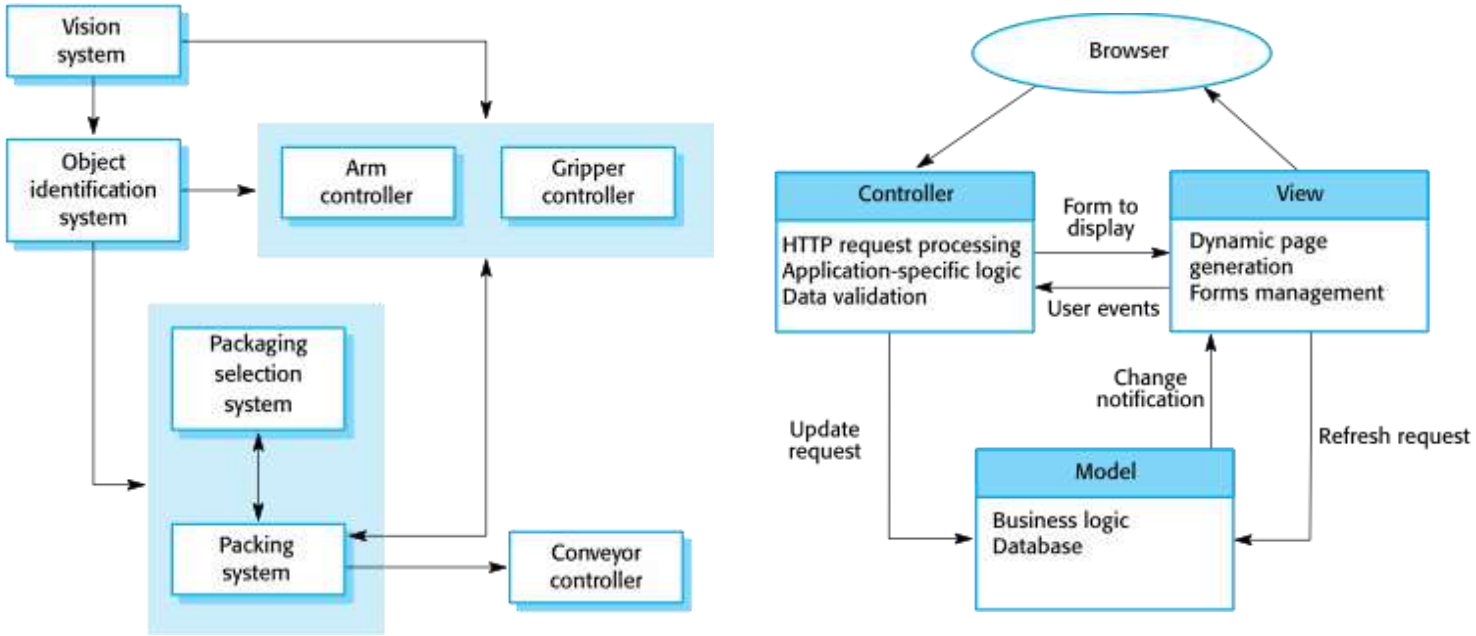
- **Abstract:** A revised version of your project's abstract (100 to 125 words).
- **Introduction**: A concise description (between 250 to 500 words) that re-states the goals of your project and gives a brief account of progress made since the previous report (specification). Also, indicate and briefly explain significant changes and updates made to your project's requirements.
  - You should probably write this last after you complete the design document

4. High-Level and Medium-level design

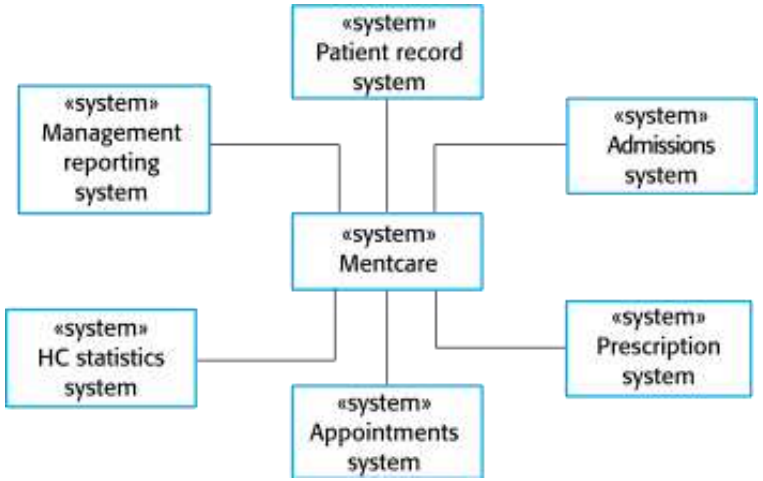
- **High-level and medium-level design**: present the project in terms of *high-level architecture, subsystems, and program units* (modules). Given the diversity of projects, there is significant flexibility here. In any case, you should include, with accompanying textual descriptions, the following:

4. High-Level and Medium-level design Cont.

- At least one *system-level diagram*, e.g., your system's context model (see Chapter 5 of the CS 425 textbook) or your system's block diagram or architectural pattern/style (see Chapter 6).



Chapter 6 Architectural Design



Chapter 5 Context Diagram

4. High-Level and Medium-level design [cont.]

- The structuring of your software in program units.
  - In the case of object-oriented solutions, the classes are examples of such program units, hence *a design class diagram* with details of attributes, operations, relationships, and multiplicity constraints should be provided (at least **<9/12>** classes are expected). Briefly describe the role of *each class* and indicate its main *attributes* and *methods* (in total, at least **<18/24>** methods should be described).

4. High-Level and Medium-level design [cont.]

- In non object-oriented solutions, *program units* can be modules, functions, procedures, subroutines, etc. Show the organization (hierarchical or not) of these units (at least **<15/20>** units are expected) and provide for each of them: name, description, the higher-level unit (e.g., subsystem) to which the program unit belongs, its input, its output, program units called by this unit, its exceptions or interrupts (if any), and any additional comments that could enhance the description of the unit.

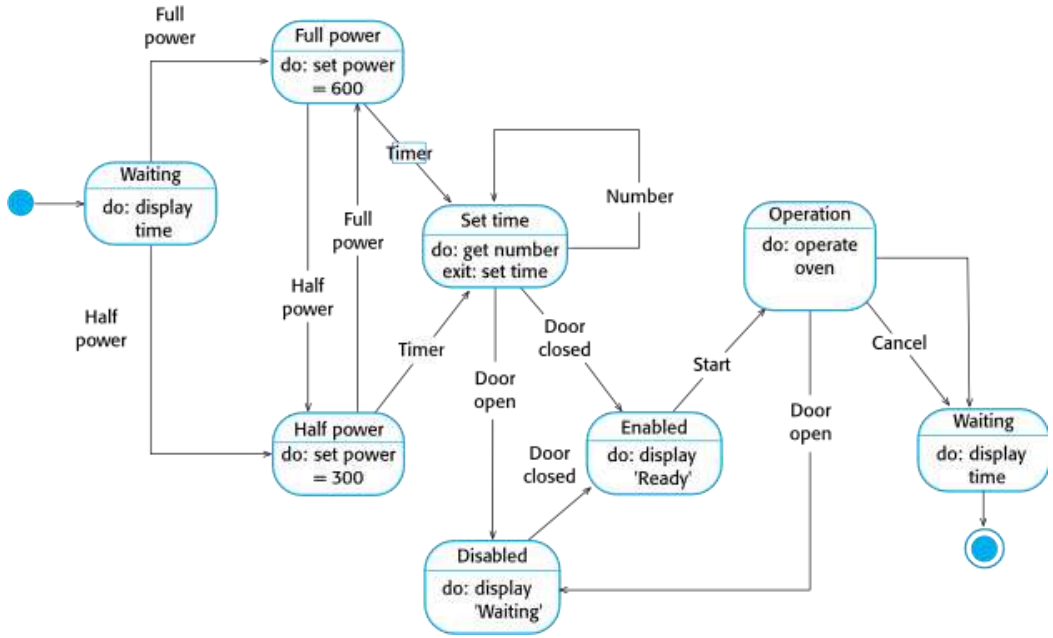
4. High-Level and Medium-level design Cont.

- Describe the main *data structures* that will be used in the project. If *database tables* are used, for each table indicate its fields (columns) and its primary key(s). For instance, a table containing information on employees may look like the following one (note that the primary key, shown in bold, is ID# - see below)

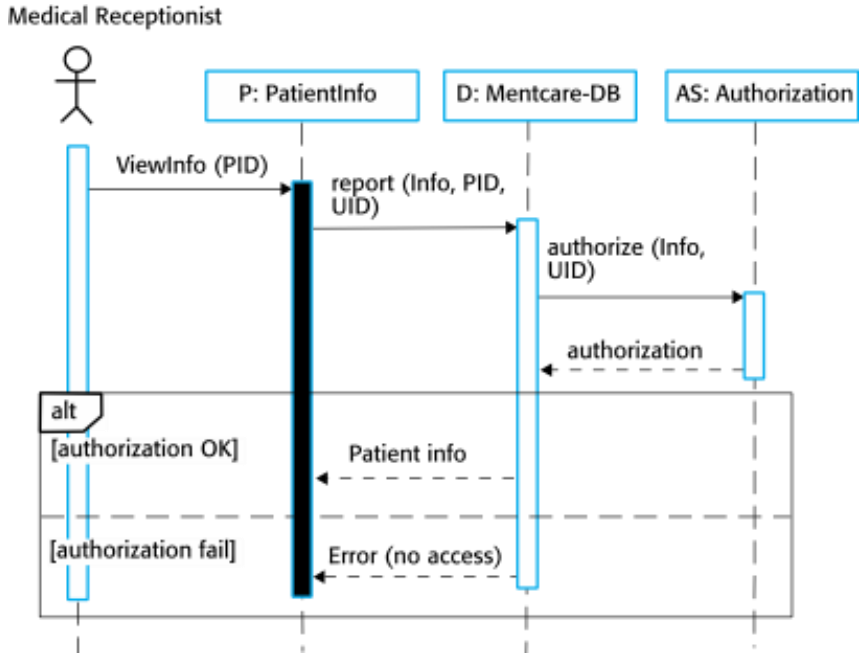
ID#	Last Name	First Name	Position	Department	Office	Telephone	Email

5. Detailed Design

- At least <**3/4**> *state charts, activity diagrams,* and/or *sequence diagrams* describing in details non-trivial components of your system's behavior (operations/functions). Use at least 2 types of detailed design notations (of the above three: state charts, activity diagrams, sequence diagrams).
  - All charts should be drawn electronically and not freehand



State Diagram (State Chart)



Sequence Diagram

6. Initial Hardware Design – only if applicable

- A *high-level diagram* showing the organization of the hardware components of your system (that is, main components and their connections).
- A list of potential *components* with brief descriptions of their roles. Include several snapshots (photos, figures, or diagrams) of components likely to be used. Indicate the sources used for snapshots/figures.

7. User interface design

- Provide at least **<8/10>** *snapshots of the user interface*, with accompanying descriptions. In these snapshots, the main user interface components (e.g., panels, toolbars, menus, menu items, buttons, data entry boxes, etc.) should be presented with details, and the format used in outputting results, messages, reports, and/or statistics should be shown.

8. Version control and software management system

- For your project, create a public repository in GitHub, and include a link to it
- Please note that, if necessary, the teaching team will look at the activity in the repository to decide on certain aspects of the grading

9. Contributions of team members

- Provide an estimate on *how much time* each team member worked on this project part (P-3) and indicate *on what specific activities*.
- For estimated time use multiples of half hours.

Grading

- Both the technical content and the presentation style (including quality of writing and document formatting) of your design document will be graded.

P3 (Design) grading parts

Criteria	Ratings	Pts
Overall presentation, Sections 0, 1, 2, 3	This area will be used by the assessor to leave comments related to this criterion.	15 pts
Section 4	This area will be used by the assessor to leave comments related to this criterion.	30 pts
Section 5	This area will be used by the assessor to leave comments related to this criterion.	20 pts
Sections 6 and 7	This area will be used by the assessor to leave comments related to this criterion.	25 pts
Sections 8 and 9	This area will be used by the assessor to leave comments related to this criterion.	10 pts
		Total Points: 100