**Product Design**

| **Team** | <Team Account & Team’s Cool Name In **Bold Black**> |
| --- | --- |

* ***Delete or replace everything in blue. At the end, all of the text should be black.***
* ***This is a “living document”, meaning its content and format will grow with the implementation of the project. Use it to capture key project concepts and to document important design decisions.***
* ***You may use any drawing tool for your UML diagrams. ArgoUML or Lucid are two options. Please make the images included in this document legible and include all images within the document.***

# **Architectural Overview**

***This diagram represents the major subsystems of the product. Initially focus on the domain layer and its components before decomposing the user interface component. Note that a common interface allows both the GUI and a Command Line Interface to access the domain model in the same manner without regard to the type of presentation technique.****Components and Functions*

*Identify the components (logical groupings of software modules that provide a related set of services). For each component, specify the information it maintains and the functionality it provides. Provide sufficient detail so that the purpose of each component in the design is clear, and so your instructor can assess each component’s appropriateness for the problem at hand.*

| <Component No. 1> | <Component state   * What information is the component responsible for maintaining? * E.g., a printing subsystem might hold the current status of all the printers it controls as well as the queue of print jobs waiting to be printed.   Component behavior   * What services does the component provide to other components? * E.g., the printing subsystem might support the queuing up of new jobs, estimating the time until a given job completes, or emailing status information at the end of a job. |
| --- | --- |
| <Component No. 2> | <Component state   * What information is the component responsible for maintaining?   Component behavior   * What services does the component provide to other components?> |
| <Component No. 3> add more rows as needed. | <Component state   * Etc.   Component behavior   * Etc.> |

# **Class Diagram(s)**

This is for UML class diagram capturing the relationships between classes. It is only necessary to show methods that are publically accessible by other classes. Only show an instance variable of a class if it is publically accessible

# **Sequence Diagram(s)**

UML sequence diagram(s) for the agreed significant use case(s)*.*

# **Design Rationale**

This is a running list of issues that arise as your design process proceeds. This is an important section of the design document as it captures the **thought process** of the product's designers. It includes why or why not (rejected solutions) a design decision was made and supports future changes to the product. It should be updated whenever a design change occurs.

*It is RARELY the case that the first design you consider is the best one that you can come up with that meets the requirements and that can be implemented, tested, and delivered on schedule. Your instructor will be looking for signs that you considered at least a few approaches, and that you had a coherent rationale for preferring the design your team eventually adopts.*

*This is the place to record such thoughts – what alternatives did you consider? What are the strengths (and deficiencies) of the final design compared to the other alternatives considered? Why did you select the approach you finally chose? This last question should be answered with an eye to the tradeoffs inevitably involved in creating an appropriate design.*

*In addition, if (not when) the design has to be adjusted to meet unexpected problems or new requirements, this is the place to record what changes were made, what effect these had on the work that had been completed to date, and the rationale for the making changes (as opposed to “just toughing it out”).*