

Assignment 2

Example Business Blueprint Derivation Rationale

This document offers a sample submission for Part 1 (*Prioritize Stakeholder Needs and Associate them with System Qualities*) and Part 2 (*Derive a Business Blueprint*) from Assignment 2 using the Course Registration domain. In general, there is no single answer to these questions, and I do not expect you to present your answer in a particular form. As long as you satisfy the elements requested, you may organize your response however you wish.

The objective of this assignment is not to learn a particular technique for architecture derivation but to devise a rational plan for satisfying stakeholder concerns. Specifically, your derivation plan should focus on the following:

- *Attempting to address issues as early as possible:* The business blueprint is intentionally very abstract, allowing you to think conceptually and identify issues early when they are easier and less costly to correct.
- *Addressing all stakeholder qualities/constraints:* Be explicit about concerns that cannot be addressed in the business blueprint.
- *Acknowledging priorities:* Priorities for qualities/constraints may have been stated explicitly by stakeholders or may need to be inferred from stakeholder comments or roles.
- *Recognizing conflicts:* Stakeholder needs often conflict, forcing tradeoffs based on priorities and other factors.

In this example, assume the following stakeholder needs (qualities) were described in Section 1.5 in Assignment 1 (*Qualities and Constraints*).

Table 1: Stakeholder needs from the Course Registration domain

Quality	Description
Cost	Budget for course registration implementation and maintenance is being obtained by reducing budget allocated to other proposed projects.
Extensibility	While the essential registration process remains relatively constant, catalog rules and major programs of work evolve
Performance	10K students must register within a two-week period.
Consistent student master record	Students, staff, and faculty need to see consistent student information.

Given the qualities above, the following is a possible prioritization for Part 1 of Assignment 2 (*Prioritize Stakeholder Needs and Associate them with System Qualities*).

Table 2: Prioritized Stakeholder Needs

Priority	Need/Quality	Classification	Priority Justification
1	Budget for course registration implementation and maintenance is being obtained by reducing budget allocated to other proposed projects.	Cost	Dictated by university administration based on legislative appropriations for the fiscal year during which development will occur. There are no other sources of funding.
2	10K students must register within a two-week period.	Performance	Based on proposed registration period schedules and historical statistics. The current registration implementation has had difficulty supporting this demand.
3	Students, staff (e.g., advisors), and faculty need to see consistent student information.	Consistent Student Record	In the current registration implementation, a “real-time” student record is not available to all users. While staff and faculty have been able to work around lags resulting from batch updates, their jobs (and decisions) would be made considerably easier if all users saw a consistent picture of a student’s status.
4	While the essential registration process remains relatively constant, catalog rules and major programs of work evolve.	Extensibility	Whenever rules change, a portion of the system must be reprogrammed. The annual budget includes this effort, but administration would like to minimize the cost.

Using the priorities and qualities from Table 2, Table 3 is a possible business blueprint derivation plan for Part 2 (*Derivation Plan and Rationale*) using heuristics from the *Business Blueprint Allocation Heuristics* handout. Note that Assignment 2 requires that the plan include at least 5 heuristics from this handout and that at least one of those heuristics is related to coupling/cohesion. The heuristics that are shown with strikethrough were pruned from the plan due to potential conflicts (see Table 4 below).

Table 3: Derivation Plan

1	Goal: Cost	
1.1	<i>BB Heuristic:</i> Group based on Implementation Reality (allow existing off-the-shelf solutions to influence component functionality)	<ul style="list-style-type: none"> • <i>Why:</i> By aligning components and their respective functions to the capabilities of known off-the-shelf solutions, there will be more opportunities to purchase solutions rather than develop custom applications. • <i>Priority justification:</i> Limited in-house developer resources.
1.2	<i>BB Heuristic:</i> Group based on Similar Capabilities (course registration performer roles)	<ul style="list-style-type: none"> • <i>Why:</i> By mapping to existing performer roles, it may be possible to use automated solutions for some components and manual solutions for others. • <i>Priority justification:</i> While this is an option, the preference is for a fully automated solution.
2	Goal: Performance	
2.1	<i>BB Heuristic:</i> Group based on Data Usage Frequency (student master record colocated with the functions that update it)	<ul style="list-style-type: none"> • <i>Why:</i> The intent is to colocate the student master record with the functions that update it since it will be accessed often. • <i>Priority justification:</i> The student record is clearly the most significant point of contention within the system.
2.2	<i>BB Heuristic:</i> Reduce Data/Event Dependency (reduce component-to-component coupling from inputs/outputs)	<ul style="list-style-type: none"> • <i>Why:</i> Reducing inputs and outputs sent across component boundaries may reduce (i) the likelihood of communication bottlenecks and (ii) the need for inter-component communication channels, which may be slower than intra-component communication. • <i>Priority justification:</i> Much of the data exchanged are portions of the student record, so 2.1 should take precedence.
2.3	<i>BB Heuristic:</i> Group based on Architectural Style – Client/Server based on Data (student record is categorized as needing to be “highly available,” so related functions are colocated with it in a given	<ul style="list-style-type: none"> • <i>Why:</i> The client/server architecture has a good track record for high-performance applications. A lot of system software exists for the

	component)	<p>development and tuning of these applications.</p> <ul style="list-style-type: none"> • <i>Priority justification:</i> While a client/server architecture may ultimately be selected for the system design, the designers would prefer not to be constrained to this option.
3	Goal: Consistent Student Record	
3.1	BB Heuristic: Group based on Data Usage Frequency (student master record colocated with the functions that update it)	<ul style="list-style-type: none"> • <i>Why:</i> Colocating the student master record with the functions that update it will reduce the possibility that it will be replicated across components, which would increase the possibility of introducing inconsistencies. • <i>Heuristic Priority:</i> N/A
4	Goal: Extensibility	
4.1	BB Heuristic: Isolate risk – Functions (isolate advising service to facilitate enhancement)	<ul style="list-style-type: none"> • <i>Why:</i> Since the catalog rules and major programs of work evolve, the advising service is likely to need frequent enhancements. The intent is to isolate the advising service so that it can be updated with minimal impact on other functions. • <i>Priority justification:</i> This was explicitly mentioned by the stakeholders.
4.2	BB Heuristic: Reduce Class Complexity—Size (reduce number of functions in a component)	<ul style="list-style-type: none"> • <i>Why:</i> Software development best practices suggests that smaller, simpler components may be easier to modify. • <i>Priority justification:</i> While a worthwhile objective, this should not be emphasized at the expense of 4.1.
4.3	BB Heuristic: Reduce Data/Event Dependency (reduce component-to-component coupling from inputs/outputs)	<ul style="list-style-type: none"> • <i>Why:</i> When modifying system components, minimizing coupling reduces the number of interfaces required, simplifying testing as well as development. • <i>Priority justification:</i> As with 4.2, this should not be emphasized at the expense of 4.1.

The assignment asks you to discuss any stakeholder needs that cannot be addressed in some fashion in your derivation plan. In this example, all qualities are referenced in the above plan; however, cost will become a greater issue when designing/purchasing technologies in the solution blueprint and defining runtime environment in the deployment blueprint.

Assignment 2 also asks you to identify potential conflicts among heuristics and describe how the plan might be modified based on priorities and heuristic effectiveness. Table 4 identifies conflicts from the derivation plan in Table 3 and offers possible resolutions. If the resolution results in pruning a heuristic, the heuristic is shown with strikethrough in Table 3.

Table 4: Potential conflicts and impact on derivation plan

Potential Conflict	Possible Resolution
<i>Group based on Implementation Reality</i> (1.1) may conflict with a number of the other heuristics as it would force component boundaries based on existing solutions.	Given its usage in Goal #1, apply <i>Group based on Implementation Reality</i> when bootstrapping the blueprint, defining whatever components are possible based on known solutions. Then apply other heuristics for refinement iterations.
<i>Reduce Class Complexity – Size</i> (4.2) may conflict with <i>Reduce Data/Event Dependency</i> (2.2 and 4.3) because the latter heuristic may attempt to collect functions into larger components while the former would likely split components.	Emphasize <i>Reduce Data/Event Dependency</i> given its usage in Goal #2.
<i>Group based on Data Usage Frequency</i> (2.1 and 3.1) may conflict with <i>Group based on Similar Capabilities (performer roles)</i> (1.2) since it is likely that more than one role updates the student record.	Given that <i>Group based on Similar Capabilities (performer roles)</i> will probably conflict with <i>Group based on Implementation Reality</i> , which was chosen for the bootstrap, apply <i>Group based on Data Usage Frequency</i> during refinement