CSE 460
Software Analysis and Design
(Fall 2022)

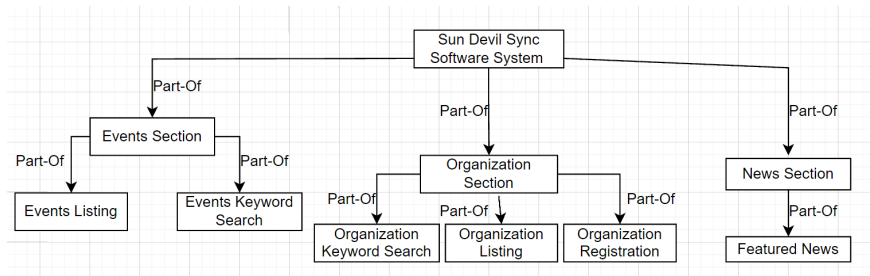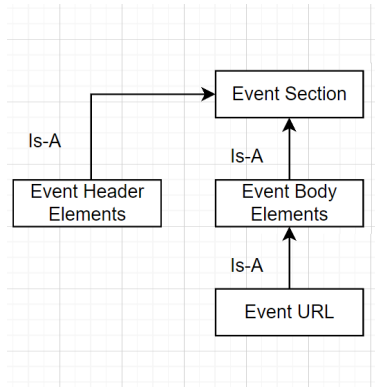**Homework #1**
Assigned: August 26, 11:59 pm
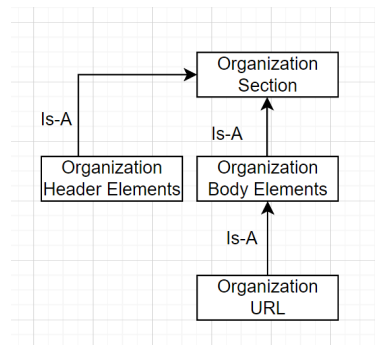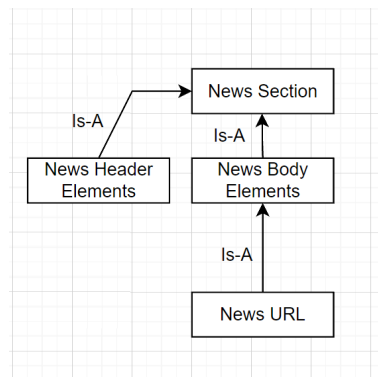**Due:** September 4, 11:59 pm

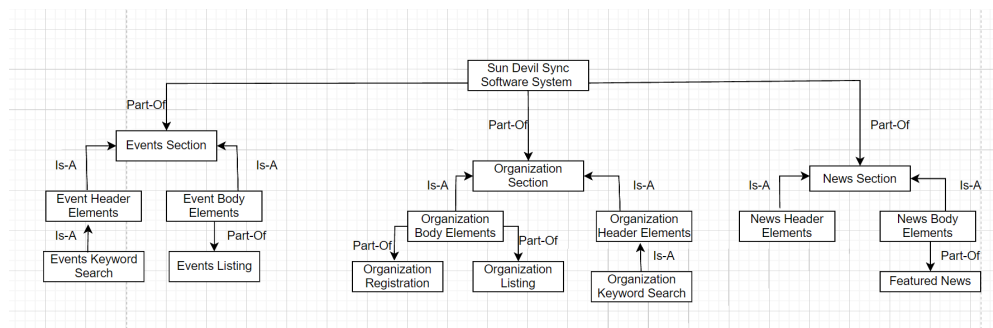Posting ID   9816-379

1) (i)

(ii)

Event Section

Organization Section

News Section

(iii)

2) (i) With reference to the Sun Devil Sync software system, it takes the form of a hierarchy. It can be split into a higher level of view of its webpage consisting of only header and body. And the body and header can be split further into smaller subsystems. For instance, header has a search field and body has listing and registration (with reference to organization section).

(ii) The Intra-component linkages are indeed stronger than the inter-component linkages. With reference to question 1(iii), at the second level, we can see that the Sun Devil Sync software system is split into organization, events, and news section. These sections are loosely coupled between each of them since each components can only be access through their own webpage.

3) Model building directs system development based on principles of decomposition, abstraction, and hierarchy. A model shows part of the system that is being analyzed and it can be used to test our system under controlled conditions. If the model were to fail, we make changes to improve the system.

4) Having better tools to develop software is important to help reduce the development time by automating repetitive work and complex works.

A deeper understanding of the software systems can lead to better understand the limitation and hence steer software analysis and development.

5) The 2 complexities attribute are accidental complexity and large number of diverse functional requirements.

Accidental complexity can be caused due to non-optimal design decisions and unaware business needs. When that happens, the complexity of the system will drastically increase. When the system is already complex, the accidental complexity can cause the overall complexity to scale exponentially. The complexity it causes would impact in terms of code complexity and variants complexity. Since the system is now defined in a more complex manner due to accidental complexity, the code complexity will be directly affected to be able to align with the system defined. Moreover, the variants complexity will also result since the system complexity is increased, the variants of each software system built will also have to be significantly different from each other variants depending on the variants requirement. Hence both the variants and code complexity is directly tied to the accidental complexity.

The large number of diverse functional requirements is also directly tied to the code and variants complexities. With more functional requirements, the code will also have to accommodate for all the stated functional requirements. The code base will likely be larger and more complex due to the greater amount of different functional requirements. This results in code complexity. The variants complexity will also be increase since the various variants may require different forms of the large diverse functional requirements. So each variants will have a greater degree of variance leading to further variants complexity.