**Task 1**

**Principles:**

* *“Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.”*

**Issues or challenges**

1. Changing requirements late in development takes to rework of tasks that could be already finished, unexpectedly increasing the workload of the development team.
2. This could lead to the postponement of the project’s due dates.

* *“Working software is the primary measure of progress.”*

**Issues or challenges:**

1. At the initial stages of the project, planning can be more valuable than delivering working software.
2. Focusing only on early development can lead to not having the best implementation possible for lack of research and planning.

* *“The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.”*

**Issues or challenges:**

1. It may not be feasible for every team, seeing that many developers work remotely, some even from different time zones.
2. Some projects include members of the client company as project managers or other managing positions, and the client may not be available to be constantly attending face-to-face meetings.

As CTS have a small team, they could have frequent face-to-face meetings. As they do not need to plan a new project for each service, they also could make use of the “early and continuous delivery of valuable software” principle.

**Task 2**

**Scrum boards and flowcharts comparison:**

**Similarities:**

1. Scrum Boards and flowcharts both provide a good visualization of the project’s tasks, state, and workflow, breaking the development into steps and allowing the team to focus on the development of each step individually.
2. With that, while the tasks are being completed both scrum boards and flowcharts help the development team to visualize the big picture of the development process, getting essential information like what still needs to be done, the status of each task and of the project in general, and what should be prioritized.

**Differences:**

1. Flowcharts usually show the steps to be done, while the scrum board represents the state of each task, showing which tasks are done, which are being done and which are still to be done. As the state of tasks are always changing, scrum boards need to be kept up to date by the team members, and because of that, are much more interactive than flowcharts.
2. As scrum divides the project development into sprints, each sprint has its own scrum board, showing only the tasks included in that sprint. As an advantage of not being changed constantly, flowcharts get to be more detailed, both in the big picture and in individual tasks, focusing more on workflow and user stories than scrum boards.

One important thing to mention is that these tools serve distinct purposes. With that, not only could they be useful to CTS, but they could be even more useful together. Flowcharts would help the CTS team execute tasks and project workflow in general, while the scrum board would help with project management, clarifying what should be done at each moment.

**Task 3**

**DevOps:**

**Benefits**

1. DevOps involves the automation of processes, what saves time and prevents human error.
2. It provides scalability, making the project easier to increase and flexible to changes.

**Challenges**

1. DevOps rely heavily on software, and the choice of the right tools is very important, and not necessarily easy. These tools might also bring a cost and a learning curve.
2. DevOps helps the development team in many ways, but it can also require a new routine in some steps of the development process, especially after finishing each task, and a period of adaptation is needed.

CTS could take advantage of DevOps practices to improve in many ways, like automating repetitive processes, using tools to monitor software and hardware, making sure everything is updated and working as expected, and improving security to avoid loss of data.