

## EduViz – Comparing Approaches

### Waterfall

---

#### **Strengths**

Having a rigid and linear schedule of workload can sometimes help to reduce uncertainty and provide workers with a better vision of their endeavors. The use of phases ensures that work periods don't overlap, and gives concrete deadlines for which deliverables can be produced. Waterfall is elegant in its simplicity. For projects that are simple and have little chance of unforeseen events, Waterfall may be an appropriate choice.

#### **Weaknesses**

Waterfall fails to account for the integrated and highly-involved nature of this project. "Handing off" code, or a dataset forgoes many opportunities for team members to get involved at every step of the process. The fundamental limitation of Waterfall in this project is the precarious disconnect between team members when there is an inherent need for everyone to be involved. In addition, after code has been produced and testing has begun, it is difficult to "go up" the Waterfall. This is a severe limitation for our project.

### Agile/Scrum

---

#### **Strengths**

This approach calls for a Scrum master—in this case it would be a project management expert with experience in both software development and interface design. As this project spans multiple disciplines and requires a highly varied and adept skillset, having a well-rounded leader is imperative.

The use of daily standups ensures strong collaboration between team members, and helps to keep all members informed of progress. Sprints are a sensible way to organize the project, and the regular sprint planning and reviews help to reduce uncertainty and give structure to future work.

### **Weaknesses**

Agile eschews rigid long-term planning in a way that allows for flexibility and response to unpredictable events. However, the lack of long-term planning almost certainly guarantees re-work during production. For both statisticians and developers, person-hours are expensive and resources are limited. As the number of iterations cannot be determined from the beginning, this can garner re-work costs and late delivery.

The Scrum approach depends on the appointment of a Scrum master. While the Scrum master is generally conveyed as a strength, it can be a risk also. If the Scrum master does not hold other accountable for deliverables, or uses sprints ineffectively, the entire project can fall into disarray. There is also a potential for impaired knowledge-continuity in the organization after the Scrum master departs.

## **Lean**

---

### **Strengths**

Lean methodology values cross-functional teams, which is absolutely mandatory for this project. Because this is a highly-technical project that demands rigorous programming and statistical analysis, collaboration is essential. Lean also focuses on *small batch size* and an MVP, or Minimum Viable Product.

The immensity of the dataset lends itself well to small batch sizes; pulling data into smaller subsets that are more easily understood helps to reduce the overwhelm of a daunting dataset. Parsing the entire dataset and producing dozens of visualizations

demands much time and effort, but producing independent visualizations from smaller parts of the dataset is quite feasible.

### **Weaknesses**

Many posit that Lean is more appropriate for high-volume work, whereas Agile is more appropriate for high-uncertainty work. This project has both, however, it is more important to manage uncertainty. It is known that the dataset will have some limitations in the information that it can provide, but it isn't clear how exactly. Intensive statistical analysis will yield findings, but these findings can be unpredictable and will determine the way that visualizations are produced. Lean is preferable to Waterfall, but lacks the ability to respond to uncertainty in the way that Agile/Scrum does.