

Kirk Thelen

Team 1: Programming Analogies (Dr. Briana Bettin)

App Idea: An application (“the App”) that stores structured analogies for explaining programming concepts. The analogies will be sortable, allowing an organized format from which to find a relevant analogy. The App must allow users to search for specific analogies from an existing database and see a “showcase” of the structured components of the analogy selected. Ideally, users would also be able to create new analogies, add notes to existing analogies, and compare existing analogies side by side, and the App’s design would be responsive (suitable for different resolutions/mobile devices/etc.).

Users:

01. **Instructors:** Instructors, specifically computer science instructors, are the primary user of the App. They possess computing expertise that they want to teach to students, and are using the App to find an appropriate analogy to do so. Other than their computing expertise, their demographics vary (gender, age, native language, etc.) and the App should be careful to be as accommodating to these factors as possible. It should be noted that although these users have the most technical knowledge and experience, not all of them will be technologically savvy and they will want the App to be simple and accessible, yet efficient in order for it to be a useful option for them.
02. **Teaching Assistants (TAs):** TAs, specifically TAs for computer science or other programming courses, are the secondary users of the App. Their reasons for using the App are similar to Instructors, and their demographics vary similarly as well. Most of these users are in their 20s, though not all may be. Although these users may have less computer science-specific experience, they are technologically savvy and will likely have the least trouble using an App.
03. **Students:** Students are the tertiary users of the App. Students will not be using the App to teach others, but may be using it to supplement their own learning. Their demographics will be similar to TAs, possibly even slightly younger, but some users may be older. For the most part, they are as technologically savvy as TAs, but a small portion of these users may not be. In general, the only difference between their goals and the goals of TAs are that they want to teach themselves and TAs want to teach others.

User Goal Table: The goals of our users are mostly aligned with some minor differences.

User	Goal
Instructor	<u>Essential:</u> <ul style="list-style-type: none">● Effective, Efficient: The App must accomplish its core goals of providing analogies for programming concepts. The App presents a lot of information, and the way that it does so must be clear and easy to follow.● Learnable, Memorable: These users are very busy and not all

	<p>will be quick to pick up new technology, so the App must be simple and easy to learn.</p> <ul style="list-style-type: none"> ● Accessible: The App must be accessible to a wide variety of backgrounds, as these users are all very different outside of their computing expertise. ● Ergonomics: The App must have a responsive design, allowing users to access it from the device of their choice (laptop, smartphone, tablet, etc.). <p><u>Ideal:</u></p> <ul style="list-style-type: none"> ● Utility: The App ideally will provide tools for these users to add their own analogies or comment on existing ones, allowing them more control over their experience.
<p>Teaching Assistant (TA), Students</p>	<p><u>Essential:</u></p> <ul style="list-style-type: none"> ● Effective, Efficient: The App must accomplish its core goals of providing analogies for programming concepts. The App presents a lot of information, and the way that it does so must be clear and easy to follow. ● Learnable, Memorable: These users are busy, but are quick to learn new technology. These technologically savvy users will not be held back by minor obstacles. ● Accessible: Most of these users are in the same age range, but not all will be. The App must be accessible to users from a variety of backgrounds, as these users may or may not have different life experiences, language mastery, and disabilities. ● Ergonomics: The App must have a responsive design, allowing users to access it from the device of their choice (laptop, smartphone, tablet, etc.). <p><u>Ideal:</u></p> <ul style="list-style-type: none"> ● Utility: The App ideally will provide tools for these users to add their own analogies or comment on existing ones, allowing them more control over their experience.