

Applying Persona Knowledge to the Test-Runner UI/Prototype

Recognizing that students and professors have different needs and face various challenges in their studies is important when developing the Test-Runner extension so that it becomes more user-friendly.

One challenge that a student new to programming such as Jose may encounter is dealing with dependencies installation and tool configuration which are complicated.

In order to solve this problem, there should be setup wizard within the VSCode extension which automatically detects OS used by a user and installs all required dependencies thus eliminating manual configurations and command line interactions. The process should be accompanied by clear instructions as well as progress indicators so that learners can be able to follow them easily feeling less scared.

Students may also be confused by command-line navigation and test execution, so this process needs to become simpler. The VSCode interface should incorporate Test-Runner functionality directly into it. For easy understanding of progress and identification of weak areas, results must be shown in the editor with clear visual indicators such as icons or color coding to represent pass/fail status. Therefore, UI needs contextual help and guidance because learners might not know certain terms or commands. Thus, this part can explain important ideas through tooltips, explanations on hover-over events as well as tutorials which are found under a “Help” section in the extension; all these will ensure that students feel supported while learning.

But teachers like Carlos, they need assignment management and feedback tools for the whole class. Professors should be able to generate test cases easily, configure grading rubrics or track student progress with this extension among other features. This will ensure that time is saved while evaluating assignments and also ensures evaluation consistency.

More than just being functional, the UI's visual design should align with VSCode's aesthetics thereby creating an environment of familiarity and beauty for users. Equally important is accessibility – making sure that anyone can use it regardless their abilities. Additionally, feedback mechanisms should be integrated so that users can report bugs and contribute towards further development of the software.

By incorporating these elements of persona-driven design seamlessly into Test-Runner extension, we can meet needs of both students as well as instructors hence improving programming courses' learning-teaching interaction paradigms in context and wider educational environments too

References

Personas. (n.d.). The Interaction Design Foundation. <https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/personas>