PRINCE2 (Projects IN Controlled Environments)

I define benefits and problems which Just Pick solve in Pitch and Brief document.(Kloda, 2025).   
Me as the Project Owner define the product vision, prioritize the backlog, and ensure the team delivers value to users.

Martin, as the Scrum Master, facilitates Agile processes, removes roadblocks, and helps the team stay productive.

Both of us also contribute as developers and testers, writing and reviewing code. We strive to ensure software quality in accordance with best practices. we balance leadership, development, and quality assurance to drive the project’s success.

We applied “Learn from Experience" principle which aligns well with CMMI's emphasis on continuous improvement. We organized retrospective and planning sprint meetings. It helped us focus on what went well, what didn't go so well, and what actions can be taken to improve in the next iteration.

We followed practice of creating Pull Requests it triggered a code review process for other team members to examine the changes for quality and bugs before they are merged.  This relates to ISO 10006 (Quality Management) and CMMI (Process and Product Quality Assurance) by emphasizing the importance of ensuring the quality of the code.

Phase 1: Planning

During the planning phase of our movie recommendation project, we followed principles outlined in ISO 12207 (Software Life Cycle Processes). I created wireframes and developing design ideas using Figma.In brief I defined the scope, project MVP, core features and timeline phases.   
We focus on common understanding of the project and prioritized list of the features you plan to implement. We identified user need creating user stories. (Mural) We determined our core requirements and eventually identified a dataset that contained keywords for each movie, which would be valuable for our recommendation algorithm. We also determined that we’ll need to fetch movie posters from TMDB based on movie titles.

Phase 2: Algorithm Development

We did research about different types of recommendation algorithms and wanted to focus on content-based filtering and TF-IDF. However, because of human resources and lack of time required for similarity calculation, we opted to explore the capabilities of a Large Language Model (LLM) for our movie recommendation project.

Phase 3: Integration  
Develop a user-friendly interface that allows users to interact with the movie recommendation system. Connect the front-end interface to backend.   
  
Phase 4: Testing

We used pytest and jest testing to test individual components.

Evaluate how the system performs under different loads (e.g., how quickly recommendations are generated). We also used postman to test APIs and pdb – python debugger to identify and resolve issues in recommender algorithm.

Phase 5: Deployment