Bisual Bytes
CS 317 Laboratory no. 2

Members:
Ella Norienne C. Dacapio — Team Lead
Allen S. Roldan
Cherry Lee H. Jimenez
Eros P. Lucagbo
Zayq Rashid F. Maulod

RESEARCH FINDINGS

Study 1: Implementation of Waterfall Method in Model Development to Improve Learning Quality of Computer Network Courses

1. **Source:**
   Sallu, S., Harsono, Y., & Fajarianto, O. (2023). Implementation of Waterfall Method in Model Development to Improve Learning Quality of Computer Network Courses. JTP - Jurnal Teknologi Pendidikan, 25(3), 496-513. https://doi.org/10.21009/jtp.v25i3.44418
2. **Methodology Used:**
   The study employs the Waterfall methodology, a structured and linear approach, dividing the project into sequential stages: Needs and Requirements Analysis, Learning Design, Learning Implementation, Verification and Evaluation, and Continuous Maintenance and Development.
3. **Project Context and Goals:**
   The project aimed to enhance the quality of Computer Network courses at a university level by systematically developing a learning model that integrates educational, technical, and infrastructural needs. The goal was to create a curriculum that is both relevant and effective for students, ensuring that the learning experience is aligned with industry standards.
4. **Implementation Details:**
   The Waterfall methodology was implemented in a structured manner:
   - **Needs Analysis:** Surveys and interviews were conducted to identify the specific educational and technical needs of students and faculty.
   - **Design:** A curriculum was created that included both theoretical content and practical labs, ensuring that students could apply their knowledge in real-world scenarios.
   - **Implementation:** The curriculum was applied in classroom settings, with careful monitoring to ensure it met educational objectives.
   - **Verification and Evaluation:** Continuous assessments and feedback mechanisms were established to evaluate the effectiveness of the learning model and make necessary adjustments.
   - **Maintenance:** The curriculum and teaching methods were regularly updated based on feedback and technological advancements.
5. **Challenges Faced and Solutions Applied:**
   Challenges included accurately identifying learning needs, effectively integrating theory with practice, and keeping up with rapid technological changes. To address these, the project employed comprehensive data collection techniques, involved

stakeholders in validating needs, and established flexible review processes to adapt the curriculum as needed.

6. **Outcomes and Methodology Impact:**
   The implementation of the Waterfall methodology led to improved student understanding and satisfaction, resulting in a curriculum that was more relevant and effective. The systematic approach ensured that the course content was up-to-date and aligned with industry requirements, ultimately producing competent, work-ready graduates.

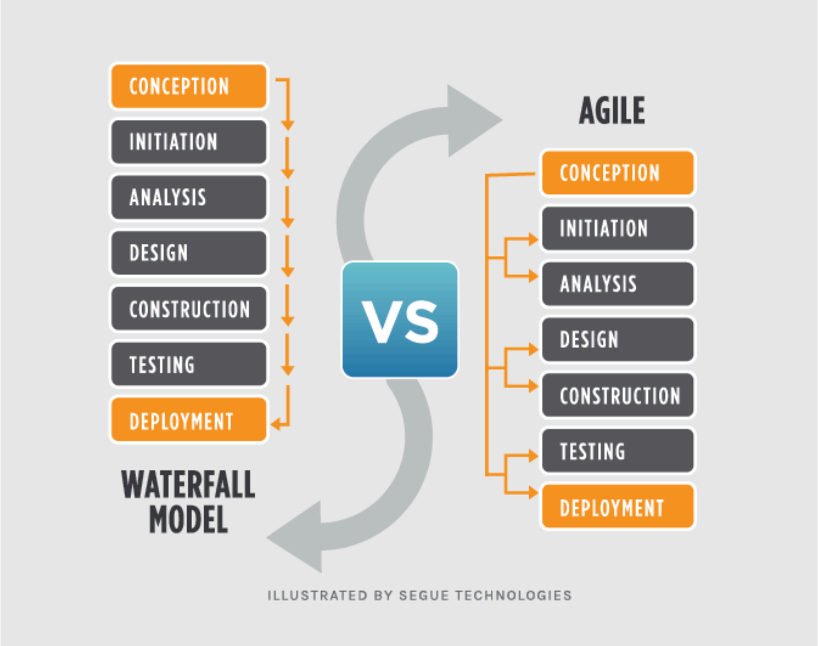
Study 2: Agile Processes and Methodologies: A Conceptual Study

1. **Source:**
   Sharma, Sheetal & Sarkar, Darothi & Gupta, Divya. (2012). Agile Processes and Methodologies: A Conceptual Study. International Journal on Computer Science and Engineering. 4.
2. **Methodology Used:**
   The study examines various Agile methodologies, particularly Extreme Programming (XP) and Scrum. These methodologies focus on iterative development, customer involvement, and flexibility in adapting to changes throughout the software development lifecycle.
3. **Project Context and Goals:**
   The Agile methodologies were explored in the context of software development projects where rapid delivery, high customer satisfaction, and adaptability to changing requirements were critical. The goals included improving productivity, enhancing software quality, and reducing time-to-market by employing Agile practices that allow for continuous feedback and iterative improvements.
4. **Implementation Details:**
   Agile methodologies were applied by breaking down the software development process into small, manageable increments or iterations:
   - **Iterations:** Each iteration involved planning, development, testing, and obtaining customer feedback, ensuring that the final product closely aligned with user requirements.
   - **Minimal Documentation:** Agile emphasized minimal documentation, focusing on the most essential features and user needs.
   - **Customer Interaction:** Frequent meetings with stakeholders were held to gather feedback and adjust the project direction as needed.
5. **Challenges Faced and Solutions Applied:**
   The main challenges included managing customer expectations and maintaining consistent communication throughout the development process. These were mitigated by holding regular meetings with stakeholders to gather feedback and adjust the project direction accordingly. Additionally, Agile teams adapted their methods to accommodate changes in requirements without causing significant disruptions.
6. **Outcomes and Methodology Impact:**
   The outcomes included higher customer satisfaction due to increased involvement and feedback, improved product quality, and enhanced team productivity. The study concluded that Agile processes significantly outperformed traditional software

development models in terms of flexibility, risk management, and overall project success.

ANALYSIS AND APPLICATION

Comparative Analysis



1. **Structure and Process Flow:**
    a. *Waterfall:* A linear and structured model, ideal for projects with well-defined requirements. It provides clear documentation and progress tracking but lacks flexibility if changes are needed later.
    b. *Agile:* An iterative model focused on adaptability and customer feedback. It's flexible and suited for projects with evolving requirements but may lack thorough documentation and predictability.

2. **Customer Involvement:**
    a. *Waterfall:* Involves customers mainly at the start and end of the project, which can lead to mismatches with changing needs.
    b. *Agile:* Encourages continuous customer feedback throughout, ensuring alignment with user requirements, but requires constant customer involvement.

3. **Flexibility and Adaptability:**
    a. *Waterfall:* Suited for projects with stable requirements but struggles with changes after a phase is completed.
    b. *Agile:* Highly flexible and can adapt to changes quickly but can lead to scope creep if not managed well.

4. **Documentation and Planning:**

a. *Waterfall:* Emphasizes detailed documentation, beneficial for regulated environments but can slow down the process.
b. *Agile:* Minimal documentation focuses on working software but can make it harder to onboard new team members or track progress.

5. **Suitability in Different Contexts:**
    a. *Waterfall:* Best for projects with fixed requirements, like government or education, where documentation and control are crucial.
    b. *Agile:* Ideal for fast-changing industries like software development, where rapid iterations and customer feedback are key.

Methodology for Proposed Project

1. **Chosen Methodology:**
   Agile Development Methodology

2. **Rationale for Selection:**
    a. Why is this methodology suitable for your proposed project?
    i. Agile is well-suited for game development, especially for a visual novel project like ours. The iterative nature of Agile allows for continuous testing and feedback, which is essential when developing a story-driven game where player choices significantly impact the narrative. It supports the creative process, enabling the team to refine story elements, artwork, and game mechanics as they develop.

    b. How does it align with the project goals and requirements?
    i. Phase One: Game Concept & Design
    Agile aligns with this phase by allowing the team to iteratively develop and refine the game's core concept, characters, and plot. As new ideas emerge during brainstorming sessions, Agile provides the flexibility to incorporate these changes without disrupting the overall timeline.
    ii. Phase Two: Development and Integration
    Agile supports the gradual development of the game's art, graphics, and programming. As each sprint is completed, elements like character design, UI, and core mechanics can be integrated into the game engine in stages. This approach minimizes risks and ensures that all components are aligned with the project's vision.
    iii. Phase Three: Internal Testing
    Regular playtesting at the end of each sprint aligns with Agile's emphasis on continuous testing and improvement. Feedback can be quickly acted upon, allowing the team to fine-tune gameplay, fix bugs, and ensure a polished final product.
    iv. Agile's flexibility, focus on iteration, and emphasis on collaboration make it the ideal methodology for creating a dynamic, story-driven visual novel game. It ensures that all aspects of the game—from design to final testing—are developed cohesively, aligning with the project goals and meeting the expectations of your target audience.

CONCLUSION

Summary of Key Insights

The research highlighted the strengths and weaknesses of both the Waterfall and Agile methodologies, leading to a well-informed choice for our project:

1. **Flexibility and Adaptability:**
   Agile's iterative nature was identified as crucial for our visual novel game project. The ability to adapt to changes in story elements, character design, and gameplay mechanics is vital, especially as new ideas and feedback emerge during development.

2. **Customer/Player Involvement:**
   Agile's focus on continuous feedback aligns with our goal of creating a player-driven experience. By regularly testing and refining based on internal playtests, we ensure that the final product meets the expectations of our target audience, young adolescents.

3. **Iterative Development:**
   The ability to break down the project into smaller, manageable sprints allows for steady progress in storyboarding, scripting, and integrating visual and programming elements. This iterative approach ensures that each aspect of the game is fully developed and polished before moving on to the next.

4. **Risk Management:**
   Agile's flexibility mitigates the risks associated with creative projects like game development. By regularly assessing progress and making adjustments, we can address potential challenges early, ensuring that the project stays on track.

These insights informed our decision to choose Agile as the methodology for our visual novel game project, as it provides the necessary flexibility, focus on player feedback, and iterative progress that align with our project goals.

Reflections

Researching and applying software development methodologies was a valuable experience that deepened our understanding of software engineering, particularly in the context of game development.

1. **Challenges Faced:**
   a. *Choosing the Right Methodology:* One of the main challenges was selecting the most suitable methodology for our project. Initially, the structured approach of Waterfall seemed appealing due to its clear stages, but we quickly realized it lacked the flexibility needed for a creative project like ours. This challenge highlighted the importance of aligning the methodology with the specific needs of the project.

    b. *Understanding the Trade-offs:* Balancing the need for flexibility with the requirement for documentation and planning was another challenge. Agile offers adaptability but can lead to less thorough documentation, which is crucial in complex projects. This forced us to consider how we might incorporate aspects of Waterfall, like documentation, into an Agile framework.
    c. *Applying Theoretical Concepts Practically:* Translating theoretical knowledge of Agile into practical steps for our visual novel game was challenging. Understanding how to break down tasks into sprints and maintain consistent team communication required careful planning and coordination.

2. **Contribution to Understanding Software Engineering:**
    a. *Importance of Flexibility:* The process reinforced the importance of flexibility in software development, especially in projects with evolving requirements. It showed us that methodologies are not one-size-fits-all and must be tailored to the project's unique context.
    b. *Iterative Development:* We gained a deeper appreciation for iterative development. By breaking down the project into smaller, manageable pieces, we were able to focus on refining each aspect of the game, ensuring higher quality and better alignment with our goals.
    c. *Collaboration and Communication:* The experience underscored the significance of collaboration and communication in software engineering. Agile's emphasis on regular feedback loops and teamwork is essential for keeping the project on track and ensuring that all team members are aligned with the project's vision.

REFERENCES

[1] Sallu, S., Harsono, Y., & Fajarianto, O. (2023). Implementation of Waterfall Method in Model Development to Improve Learning Quality of Computer Network Courses. JTP - Jurnal Teknologi Pendidikan, 25(3), 496-513. https://doi.org/10.21009/jtp.v25i3.44418
[2] Sharma, Sheetal & Sarkar, Darothi & Gupta, Divya. (2012). Agile Processes and Methodologies: A Conceptual Study. International Journal on Computer Science and Engineering. 4.