

Implementation Of Extreme Programming In The Development Of Class And Reservation Management Application At Fitness Centers

Denny Jean Cross Sihombing

Information System Study Program, Atma Jaya Catholic University of Indonesia

Article Info

Keywords:

Reservation Management,
Extreme Programming,
Fitness Centers.

ABSTRACT

This research investigates the implementation of Extreme Programming (XP) in developing a class management and reservation application in fitness centers. The issues under scrutiny include identifying precise system requirements, applying XP principles in the design and implementation phases of the application, and evaluating application performance through comprehensive testing. The research methodology encompasses system requirement analysis, architectural planning, prototype development, functionality testing, and performance evaluation. The findings indicate that implementing XP effectively enhances application quality, user interface responsiveness, and user satisfaction. The contribution of this research lies in providing practical guidance for software developers in applying XP for efficient and high-quality application development, as well as enriching the literature on software development methodologies.

This is an open access article under the [CC BY-NC](https://creativecommons.org/licenses/by-nc/4.0/) license



Corresponding Author:

Denny Jean Cross Sihombing
Atma Jaya Catholic University of Indonesia
Jakarta, Indonesia
denny.jean@atmajaya.ac.id

INTRODUCTION

The use of technology in the fitness industry is increasingly widespread, particularly in class management and reservation systems at fitness centers (Blocken et al., 2020; Eberth & Smith, 2010; Fu et al., 2012). This reflects a global trend where technology is crucial in enhancing operational efficiency and providing better customer experiences across various sectors, including the fitness industry. Fitness centers are becoming more complex, with various types of classes, instructors, and schedules that need to be efficiently managed. This challenge further underscores the need for management systems to integrate information from various sources, optimize scheduling, and facilitate smooth communication among relevant stakeholders.

A software development method worth considering is Extreme Programming (XP), known for its adaptive and collaborative practices, fitting well with the dynamic operational (Akhtar et al., n.d.; Fojtik, 2011a, 2011b; Sihombing, 2023; Wood et al., 2013) nature of fitness centers. XP offers an approach that is responsive to changing business needs and enables developers to focus on delivering value faster to users. In the context of class management and reservation applications in fitness centers, the main challenge is to provide a system that

can manage class schedules, registrations, payments, and communication among customers, instructors, and management. A comprehensive and integrated software solution is needed to enhance efficiency and improve user experience.

The importance of implementing Extreme Programming (XP) in this application development lies in its ability to accommodate changing business needs and be responsive to feedback from application users and other stakeholders. With rapid iterations and continuous testing, XP allows for the early identification of issues and more effective fixes. Previous studies have shown that appropriate development methodologies, such as Extreme Programming (XP), can improve the quality, speed, and stakeholder engagement in software development (Ahmed et al., 2023; Akhtar et al., n.d.; Al-Saqqa et al., 2020; Bansal et al., 2023; Bomström et al., 2023; Chen et al., 2020; Dingsoeyr et al., 2019; Dingsøyr et al., 2012; Fojtik, 2011b; Kaur et al., 2023; Santos et al., n.d.; Serrador & Pinto, 2015; Shrivastava & Rathod, 2014; Sihombing, 2023; Wood et al., 2013). This is crucial to ensure that the resulting applications meet quality standards and established requirements.

Regarding class management and reservation applications in fitness centers, factors such as integration with existing systems, scalability, data security, and intuitive user interfaces are crucial to consider in development. Data security and user-friendly interfaces will be critical to the success of this application. Implementing Extreme Programming (XP) also requires close collaboration between the development team, fitness center management, and end-users to ensure successful implementation and application fit with actual needs. This collaboration includes a deep understanding of business needs and user desires and effective communication throughout the development process.

The success of implementing Extreme Programming (XP) in this application development can be measured by aspects of operational efficiency, improved user experience, increased revenue through easier class registration, and the system's ability to meet data security regulations and standards. These are essential performance indicators to evaluate the positive impact of implemented software solutions. Therefore, this research aims to investigate how Extreme Programming (XP) can be effectively implemented in the development of class management and reservation applications in fitness centers and identify the benefits and challenges associated with this approach. Thus, the findings of this research are expected to provide valuable insights for the fitness industry in improving its class and reservation management processes and enhancing customer satisfaction.

METHODS

In this research, the process is divided into three main phases. First is System Requirements Analysis, where an in-depth study of application requirements is conducted through document collection, stakeholder interviews, and functional and non-functional requirements mapping. Second, application design and implementation involves application architecture planning, prototype development, and implementation of extreme programming (XP) principles in application development. Lastly, Testing and Evaluation, where the application is thoroughly tested with predefined test scenarios and performance and application success are

evaluated based on pre-established criteria. These stages aim to produce a responsive, efficient, user-friendly fitness center class and reservation management application.

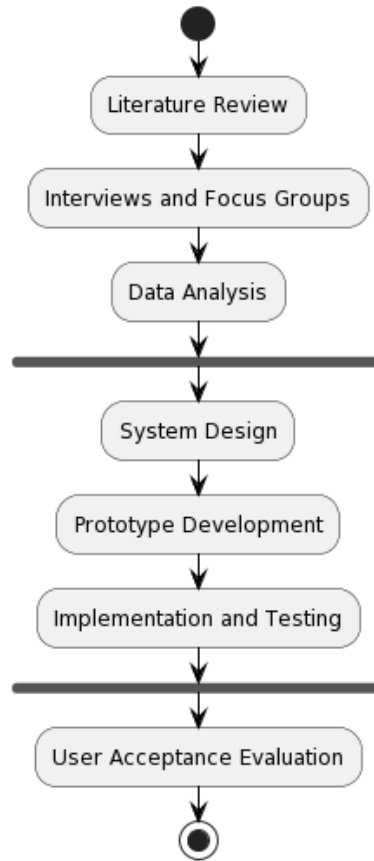


Figure 1. Research Stages

System Requirements Analysis

In the System Requirements Analysis phase, a thorough study is conducted on the class management and reservation application requirements in fitness centers. This begins with gathering relevant documents, such as class schedules, registration procedures, and reservation policies, and conducting interviews with fitness center management, instructors, and administrative staff. The aim is to understand existing operational processes, identify encountered problems, and establish the functional and non-functional requirements of the application to be developed. The outcome of this phase is a requirements mapping that includes features such as online reservations, integration with payment systems, class data analysis reporting, and customer data security.

Application Design and Implementation

Once the system requirements are identified, the next phase is Application Design and Implementation. In this phase, application architecture is planned based on the previously mapped requirements. This includes selecting appropriate technologies, such as program-

ming languages, frameworks, and databases. Subsequently, prototype development is conducted to test the application's user interface design and core functionality. These prototypes are then iteratively evaluated based on user and stakeholder feedback. Throughout the development process, XP principles such as pair programming, continuous testing, and code refactoring are applied to enhance the quality and flexibility of the application.

Testing and Evaluation

The final phase is Testing and Evaluation, where the developed application undergoes comprehensive testing. Test preparation involves drafting test scenarios covering various application usage cases and ensuring a test environment that aligns with production conditions. During testing, evaluations are made on functionality, integration with external systems, user interface responsiveness, and data security. Testing results are evaluated to identify issues, errors, and improvement areas. Comparisons are made between test results and pre-established success criteria, such as response time, system reliability, and user satisfaction levels.

RESULTS AND DISCUSSION

User Needs Analysis

The outcomes of the System Requirements Analysis phase, notably the Initial Study, encompass the collection of documents related to fitness center operations, such as class schedules, registration procedures, payment systems, and reservation policies. Additionally, interviews are conducted with fitness center management, instructors, and administrative staff to understand existing operational processes and problems that must be addressed. In the Requirements Mapping stage, a list of functional and non-functional requirements for the application is created, including features like online reservation, payment system integration, class data analysis reporting, and customer data security. Moreover, end-user requirements, including fitness center users, instructors, and customers, are identified and classified based on importance and urgency. In the Additional Requirements Analysis stage, additional needs that may arise from stakeholder interactions, such as integration with wearable devices for performance tracking or direct communication features between instructors and customers, are identified. Thus, this phase provides a strong foundation for developing class management and reservation applications in fitness centers that can effectively meet user needs and address daily operational challenges.

Table 1. Data Collection Results

Stakeholder	Key Needs	Challenges Addressed	Expectations and Desires
Fitness Center Management	Improve operational efficiency and class scheduling management	Complex scheduling issues, complicated registration system	Expecting a system that can automate class scheduling and registration efficiently

Instructors	Ease in managing classes and interacting with clients	Difficulty in accessing class data and client communication	Expecting tools that assist in class management and better communication with clients
Clients	Ease in class reservations and clear information about classes	Difficulty in class reservations and quick access to information	Desiring an intuitive reservation system and accurate information about class schedules
Administrative Staff	Efficient and accurate administrative management	Limitations in the payment system and data reporting	Hoping for a system that simplifies administration and provides comprehensive and timely reporting

The results of this stage are summarized in Table 1, indicating that fitness center management prioritizes operational efficiency and class scheduling management, with the expectation of a system that can automate registration and scheduling processes. Instructors seek tools to aid in class management and improve customer interaction, while customers expect an intuitive reservation system and accurate class schedule information. Administrative staff also seek a system to streamline tasks and provide accurate and timely reports. This analysis suggests that application development needs to consider the diverse needs of each stakeholder to create a solution that satisfies all parties involved.

Design and Implementation

The outcomes of the Application Design and Implementation phase encompass several crucial aspects. Firstly, in Architecture Planning, the application architecture design is conducted, covering security, scalability, and integration with existing systems. Subsequently, selecting appropriate technologies, such as programming languages, frameworks, and databases, becomes critical for efficient application development. In the Prototype Development stage, application prototypes are created to test user interface design, navigation, and key features, with rapid iterations based on user and stakeholder feedback. Lastly, Functional Development focuses on developing core application features such as class scheduling management, customer registration, payment systems, and class performance analysis dashboards. Moreover, the application of Extreme Programming (XP) principles such as pair programming, continuous testing, and code refactoring is also undertaken to enhance the quality and flexibility of the application. Thus, this phase is a strong foundation for developing a responsive, efficient application that meets user needs.

Table 2. Features

Feature	Function
Online Reservation	Allows customers to make class reservations online through the application, with easy access to available class schedules.
Payment Integration	Enables customers to pay online or integrates the payment system with the application for transaction convenience.

Schedule Management	Allows instructors to manage class schedules, add or modify schedules as needed, and update class information.
Notifications	Provides notifications to customers about class schedules, schedule changes, or other important information via text messages or email.
Performance Analysis Dashboard	Displays class performance analysis data such as participant numbers, attendance rates, and customer feedback for evaluation and improvement.
Customer Management	Allows administrative staff to manage customer information, register members, set membership packages, and contact information.
Instructor-Customer Communication	Provides a means of direct communication between instructors and customers, such as messages or notifications about schedule changes or class information.

Table 2 presents the essential features of the fitness center's class management and reservation application and their respective functions. With these features, the application can provide a better user experience and aid in the operational efficiency of fitness centers.

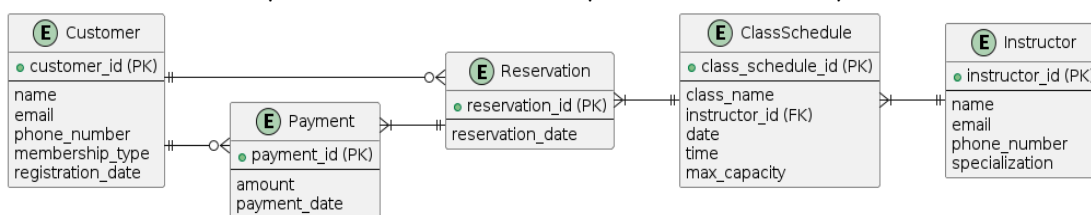


Figure 3. ERD

Figure 2 Entity-Relationship Diagram (ERD) depicts the class management and reservation application data structure in fitness centers. The main entities include Customer, Reservation, Payment, Class Schedule, and Instructor, each with corresponding attributes. Relationships between entities are established through clear connections, such as one-to-many relationships between Customer and Reservation as well as Payment and many-to-one relationships between Reservation and Class Schedule. Primary keys such as Customer ID, Reservation ID, Payment ID, and Class Schedule ID serve as unique identifiers for each entity, while foreign keys such as Customer ID and Class Schedule ID link data between related entities. Thus, this ERD provides a solid foundation for developing an efficient application for managing reservation, payment, and class scheduling processes in fitness centers and facilitating accurate and structured data analysis.

User Acceptance Evaluation

Firstly, preparation is conducted by drafting test scenarios covering various application usage cases, such as class reservation, payment, searching for classes based on time or instructor, and issue reporting. The test environment is also meticulously prepared to reflect production conditions, minimizing the risk of undetected errors. Subsequently, comprehensive functional testing of the application is carried out. This includes testing integration with external systems, user interface responsiveness across various devices, and data security.

Every encountered issue is meticulously recorded, tracked, and prioritized for follow-up during the testing process. Feedback from users and stakeholders is also gathered to identify further areas for improvement. Lastly, the test results are comprehensively evaluated. Analysis is conducted to identify failures, errors, and areas requiring improvement. Comparisons are made between test results and pre-established success criteria, such as response time, system reliability, and user satisfaction levels. Based on this evaluation, appropriate improvement measures are then taken to ensure optimal application performance when launched to users.

CONCLUSION

Based on the research on developing class management and reservation applications in fitness centers using the Extreme Programming (XP) methodology, XP provides significant benefits in system requirements, application design and implementation, and testing and evaluation aspects. Careful system requirements analysis ensures the identification of precise functional and non-functional requirements while applying XP principles such as pair programming and continuous testing enhances application quality. Comprehensive functional testing ensures application responsiveness and security, while performance evaluation helps identify areas for improvement. Thus, using XP in this application development can create practical solutions that meet user needs and positively contribute to the operational efficiency of fitness centers and overall user experience.

REFERENCE

- Ahmed, M., Khan, S. U. R., & Alam, K. A. (2023). An NLP-based quality attributes extraction and prioritization framework in Agile-driven software development. *Automated Software Engineering*, 30(1). <https://doi.org/10.1007/s10515-022-00371-9>
- Akhtar, A., Bakhtawar, B., & Akhtar, S. (n.d.). EXTREME PROGRAMMING VS SCRUM: A COMPARISON OF AGILE MODELS. *International Journal of Technology, Innovation and Management (IJTIM)*, 2, 2022. <https://doi.org/10.54489/ijtim.v2i1.77>
- Al-Saqqa, S., Sawalha, S., & Abdelnabi, H. (2020). Agile software development: Methodologies and trends. *International Journal of Interactive Mobile Technologies*, 14(11). <https://doi.org/10.3991/ijim.v14i11.13269>
- Bansal, N., Singh, D., & Kumar, M. (2023). Computation of energy across the type-C piano key weir using gene expression programming and extreme gradient boosting (XGBoost) algorithm. *Energy Reports*, 9, 310–321. <https://doi.org/10.1016/j.egy.2023.04.003>
- Blocken, B., van Druenen, T., van Hooff, T., Verstappen, P. A., Marchal, T., & Marr, L. C. (2020). Can indoor sports centers be allowed to re-open during the COVID-19 pandemic based on a certificate of equivalence? *Building and Environment*, 180, 107022. <https://doi.org/https://doi.org/10.1016/j.buildenv.2020.107022>
- Bomström, H., Kelanti, M., Annanperä, E., Liukkunen, K., Kilamo, T., Sievi-Korte, O., & Systä, K. (2023). Information needs and presentation in agile software development. *Information and Software Technology*, 162. <https://doi.org/10.1016/j.infsof.2023.107265>

- Chen, J., Yu, T., Yin, L., Tang, J., & Wang, H. (2020). A unified time scale intelligent control algorithm for microgrid based on extreme dynamic programming. *CSEE Journal of Power and Energy Systems*, 6(3), 583–590. <https://doi.org/10.17775/CSEEJPES.2019.00100>
- Dingsoeyr, T., Falessi, D., & Power, K. (2019). Agile Development at Scale: The Next Frontier. In *IEEE Software* (Vol. 36, Issue 2, pp. 30–38). IEEE Computer Society. <https://doi.org/10.1109/MS.2018.2884884>
- Dingsøy, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. In *Journal of Systems and Software* (Vol. 85, Issue 6). <https://doi.org/10.1016/j.jss.2012.02.033>
- Eberth, B., & Smith, M. D. (2010). Modelling the participation decision and duration of sporting activity in Scotland. *Economic Modelling*, 27(4), 822–834. <https://doi.org/https://doi.org/10.1016/j.econmod.2009.10.003>
- Fojtik, R. (2011a). Extreme programming in development of specific software. *Procedia Computer Science*, 3, 1464–1468. <https://doi.org/10.1016/j.procs.2011.01.032>
- Fojtik, R. (2011b). Extreme programming in development of specific software. *Procedia Computer Science*, 3, 1464–1468. <https://doi.org/10.1016/j.procs.2011.01.032>
- Fu, F. H., Guo, L., & Zang, Y. (2012). An overview of health fitness studies of Hong Kong residents from 2005 to 2011. *Journal of Exercise Science & Fitness*, 10(2), 45–63. <https://doi.org/https://doi.org/10.1016/j.jesf.2012.10.001>
- Kaur, J., Singh, O., Anand, A., & Agarwal, M. (2023). A goal programming approach for agile-based software development resource allocation. *Decision Analytics Journal*, 6. <https://doi.org/10.1016/j.dajour.2022.100146>
- Santos, R., Cunha, F., Rique, T., Perkusich, M., Almeida, H., Perkusich, A., & Icaro Costa, ´. (n.d.). *A Comparative Analysis of Agile Teamwork Quality Instruments in Agile Software Development: A Qualitative Approach*. <https://doi.org/10.18293/DMSVIVA2023-217>
- Serrador, P., & Pinto, J. K. (2015). Does Agile work? - A quantitative analysis of agile project success. *International Journal of Project Management*, 33(5). <https://doi.org/10.1016/j.ijproman.2015.01.006>
- Shrivastava, S. V., & Rathod, U. (2014). Risks in Distributed Agile Development: A Review. *Procedia - Social and Behavioral Sciences*, 133, 417–424. <https://doi.org/10.1016/j.sbspro.2014.04.208>
- Sihombing, D. J. C. (2023). Analysis and development of the ProTrack application: construction timeline management using Extreme Programming Methodology. In *Jurnal Mantik* (Vol. 7, Issue 2). Online.
- Wood, S., Michaelides, G., & Thomson, C. (2013). Successful extreme programming: Fidelity to the methodology or good teamworking? *Information and Software Technology*, 55(4), 660–672. <https://doi.org/10.1016/j.infsof.2012.10.002>