# Extreme Programming for Developing Additional Employee Income System (Case Study: Karangasem Regency Government)

I Putu Noven Hartawan[1], Made Sudarma[2] and Widyantara, I.M.O.[3*]

[1]Department of Electrical and Computer Engineering, Post Graduate Program, Udayana University
[2,3]Department of Electrical and Computer Engineering, Udayana University
*oka.widyantara@unud.ac.id

**Abstract** The State Civil Apparatus (ASN) is an important part of the government. ASN performance measurement needs to be done, because besides ASN being the provider in terms of community service, performance measurement is also used as a reference in providing additional employee income. Karangasem regency government developed a system to measure the performance of the existing 6,449 ASNs. The system is applied to measure performance and as a system that provides additional employee income. However, the pandemic situation resulted in the Karangasem district government having to press the budget for developing this system. The researcher proposes the Extreme Programming (XP) approach in system development. XP, which has the value of Communication, Simplicity, Feedback and Courage, is the right choice in developing a system with a small budget and time. Feedback that is quickly handled also greatly influences the speed and cost of development.

***Keywords*—TPP, ASN, Agile. Extreme Programing.**

## I. INTRODUCTION

REGULATION of the Minister of State Apparatus Empowerment No. PER / 15 / M.PAN / 7/2008 concerning General Guidelines for Bureaucratic Reform emphasizes that the administration of government must rely on three aspects, namely institutions, human resources and governance. All these aspects give birth to the concept of performance management [1]. Performance management directs all organizational elements of government agencies, starting from the individual employee level, work units to the organizational level as a work functional unit. Therefore, performance management always looks at the contribution of individuals, teams and work units in the effort to realize organizational goals and organizational contributions in realizing local government goals.

Ministry of Home Affairs Decree No. 061-5449 year 2019. This decree states that 60% to 70% of additional employee income is calculated from work productivity. Work productivity is carried out based on the assessment of the appraisal official of the results of the implementation of the duties of the employees they lead [2]. Therefore, local governments need to implement ASN performance measurements. Because in addition to the role of ASN as an organizer in terms of community service [3], performance measurement is also a requirement in calculating additional employee income.

The number of ASN in the Karangasem Regency government is as many as 6,449 people. At present, the assessment and evaluation of ASN performance in the Karangasem Regency government is still carried out manually. With more than 5,000 ASNs, manually validating the correctness of performance reports will require more effort. This will be more sensitive because the ASN performance recap will be the basis for calculating additional revenue. The development of information technology is very rapid and has become part of running the government system [4]. Therefore, it is necessary to develop a system that can manage ASN performance.

The Karangasem Regency Government wants to develop a performance management system to supplement employee income. However, the pandemic situation resulted in the Karangasem district government having to suppress the budget for developing this system, because the budget had to be diverted to handling the pandemic. Researchers propose system development with the Extreme Programing (XP) approach. Because XP emphasizes customer satisfaction. The speed in receiving feedback and the frequent release of it is very influential in reducing system flaws. Low defect rate will shorten time and also reduce costs [5].

## II. EXTREME PROGRAMING

Extreme Programming (XP) is a lightweight software development method and is one of the agile methods pioneered by Kent Beck, Ron Jeffries, and Ward Cunningham. XP is the most widely used agile method and

is a very popular approach. The XP target is a team that is formed between small to medium sizes only, no need to use a large team. This is intended to deal with unclear requirements and very fast changes in requirements. XP is a model with full iteration, where iteration is the development of small parts of the system from the most basic features to releasing the entire system [6].

### A. XP Values

XP as a dynamic method is shown in its four values and the four are the basics needed in XP. Kent Beck states that individual short-term goals often clash with long-term social goals. Because of that, values are made that become rules, punishments, and rewards. The four values are: Communication, Simplicity, Feedback, Courage [7].

### B. XP Phase

As depicted in Fig. 1, the system development process with XP has 6 phases: Exploration Phase, Planning Phase, Iterations to Release Phase, Productionizing Phase, Maintenance Phase and Death Phase.

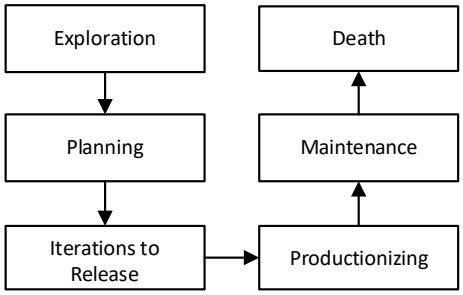


Figure 1 XP Life Cycle

#### 1) Exploration Phase

The exploration phase is the phase where the client writes down the most basic system requirements. Every need written by the client will be made in the form of a simple module or also known as User Stories. the result of the exploration phase is knowing the documentation of the vision and scope of work [8].

#### 2) Planning Phase

The planning phase is oriented towards the exploration phase. This phase will estimate business requirements, user requirements, and system requirements. This phase will also produce a schedule that describes the planning implementation time of the system development.

#### 3) Iterations to Release Phase

At this phase, it is a phase that is carried out repeatedly. In this phase consists of design, coding and testing activities. Each iteration is determined based on the part of the system requirements that have been analyzed by the developer and the client [9].

#### 4) Productionizing Phase

This production phase is the phase where a small part of the system will be released. The system that is released according to the needs is divided into several iterations. During this phase the changes of the system will be discovered. Ideas and suggestions will continue in the next maintenance phase.

#### 5) Maintenance Phase

At this phase the developer continues to develop new functions. But the developer must be careful because at the same time the system is also being used.

#### 6) Death Phase

This phase is the last phase of XP. There are two conditions by which development reaches this phase. The first condition is if the user doesn't have any new features that he wants to add and the user is satisfied with the results. The second condition is if the customer still wants to add new features, but from an economic point of view it is not possible. Better to end system development. At this phase, documentation of the system is also carried out for future development [10].

## III. METHODOLOGY

By using the XP approach as a basis, the process of activities carried out must still uphold XP Values. Fig. 2 illustrates a more detailed flow of what will be done in each phase of XP.

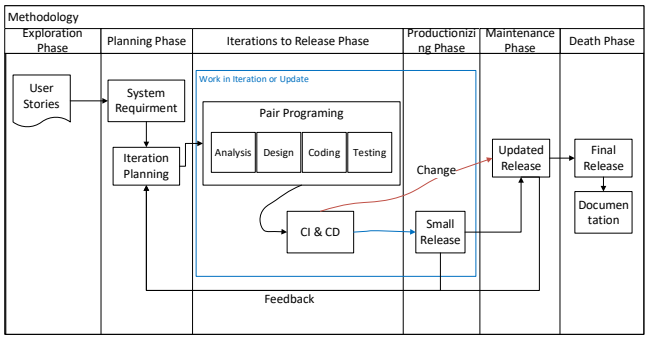


Figure 2 Model of System Development

### A. Exploration Phase

At this phase several activities will be carried out, namely determining Person In Charge (PIC). Listen to stories from clients. The developer and client will discuss the system requirements required. And determine the processing time.

### B. Planning Phase

At this phase there will be several phases carried out, namely Developing system requirements from client stories. Determine equipment and number of personnel in working on the system. Planning the iteration strategy and small releases that will be carried out.

### C. Iterations to Release Phase

At this phase, a system development is carried out which consists of a process of analysis, design, coding and testing. All of this is done together with the pair programming technique to reduce errors. This phase also uses the Continuous Integration and Continuous Deployment (CI & CD) system to proceed to the production phase.

### D. Productionizing Phase

At this phase, the system is released in the form of small pieces. These small pieces represent a set of needs that have

been broken down during the planning phase. This is done in order to get faster feedback from clients.

### *E. Maintenance Phase*

At this phase, system maintenance is carried out and is prepared to be prepared to fix system problems that will arise. Improvements will be made in conjunction with planned feature development.

### *F. Death Phase*

This is the last phase if all the needs have been met, documentation will be made as the final report. Besides being useful for future development, the final report is also an administrative requirement in government.

## IV. Result and Discussion

This section will explain the implementation results of the development model using the XP approach.

### *A. Exploration Phase*

The first thing that needs to be done in system development is knowing the PIC or who will be coordinated with. The PIC in this system project is the Personnel and Human Resources Development Agency (BKPSDM), the Organization Division and the Regional Financial and Asset Management Agency (BPKAD). Table I provides an explanation for each PIC.

TABLE I
Person In Charge

| PIC | About |
|---|---|
| BKPSDM | This department is the manager and person in charge of the system to be built. |
| Organization Division | This department handles the organization and positions in the Karangasem district government. |
| BPKAD | This department is responsible for how employee income is disbursed. |

After knowing the PIC, a joint meeting was held to find out the system requirements. As an initial phase, the developer will discuss a similar system flow that has been used in other district governments as an initial phase. During the discussion the developer collects stories from each PIC for later analysis.

### *B. Planning Phase*

From the results of the client's story that has been collected, it is analyzed to determine the system requirements. Figure 3 is an overview of the system to be made. Position and employees will be mapped by the administrator. Mapping will produce subordinates and superiors. Employees will make reports which will later be verified by their superiors. After verification, this activity will become a point that becomes the basis for providing additional income.

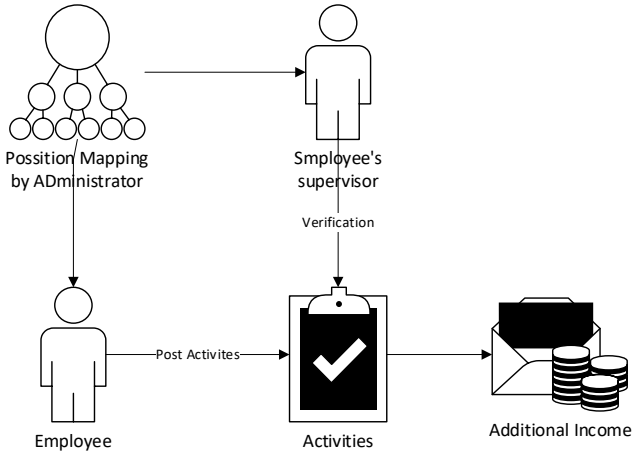


Figure 3 General Description of System

From the analysis results also produce application features which can be seen in Figure 4. There are three actors, namely the central admin, Department administrator and employees. The central administrator can access the settings features, Employee Master Data, Position Mapping, Employee Mapping, Additional Income Report. Department Administrators can access the Employee Mapping feature according to Central Administrator permission and the Additional Income Report. Employees can access the employee's own Post Activities, Position Select, Activities Verification, and Additional Income.

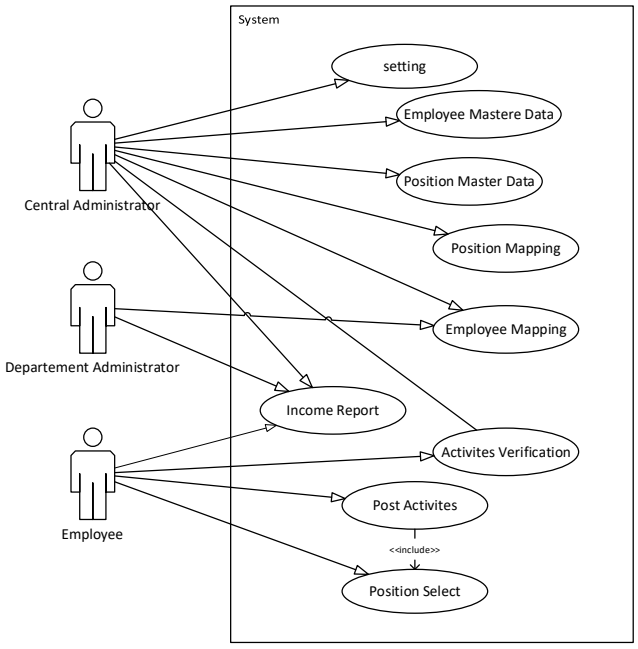


Figure 4 Use Case Diagram

### *C. Iterations to Release Phase*

This phase will be carried out in three iterations. The iteration has been previously analyzed to accelerate development. At this phase, analysis of each feature is carried out. Database design, coding and unit testing.

*1) First Iteration*

The first iteration begins with the development of the master data. There are six master data, namely OPD master data, employees, echelon, class positions, positions, activities.

The developer imports data to all master data except the activity master data. The data is obtained from the PIC. After the master data is complete, job and employee mapping features are developed.

*2) Second Iteration*

The second iteration represents the development of the main features of the system. Because the user has already made a small release in the first iteration, users can map their positions and employees while waiting for the development of the second iteration to take place. The second iteration focuses on the management features of reporting employee activities.

*3) Third Iteration*

After making a small release in the second iteration, employees began experimenting with activity reporting. Simultaneously with the trial, a third iteration was developed that focused on the Additional Employee Income report. This iteration is the most difficult iteration. Because in this iteration the programmer will receive additional or input from employees who started testing the system that was previously released in the second iteration.

## *D. Productionizing Phase*

At this phase it is also included in the iteration. There are 3 minor releases. Fig. 8 is the result of the release of the first iteration. The first iteration release focuses only on master data and job and employee mapping.

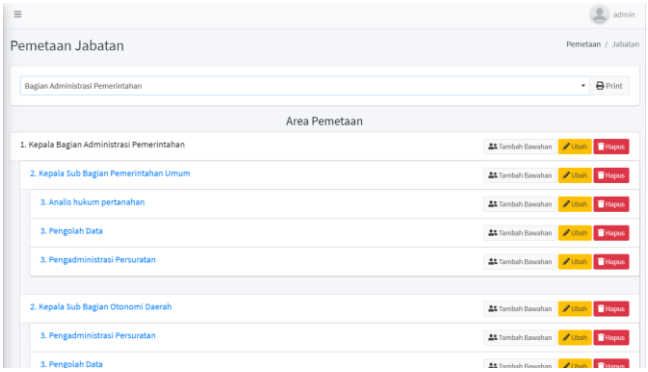


Figure 5 Position Mapping Page

After releasing the first iteration, a meeting was held again with the PIC. The Central Administrator must input the employee position mapping first so that the system can be used. At the same time the second iteration is also carried out. Fig. 6 is a small release in the second iteration that allows employees to do activity reporting.

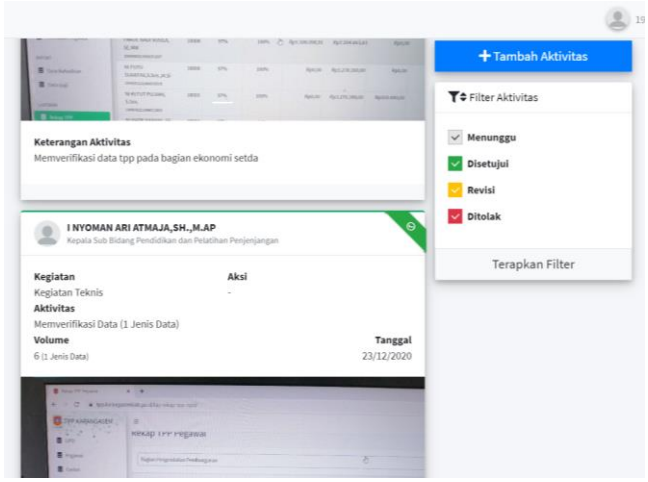


Figure 6 Activities Post Page

After the release of the second iteration, a socialization was conducted for the procedure for reporting activities and conducting trials. At the same time the third iteration was also developed. Fig. 7 is a small release that focuses on the additional employee income report.

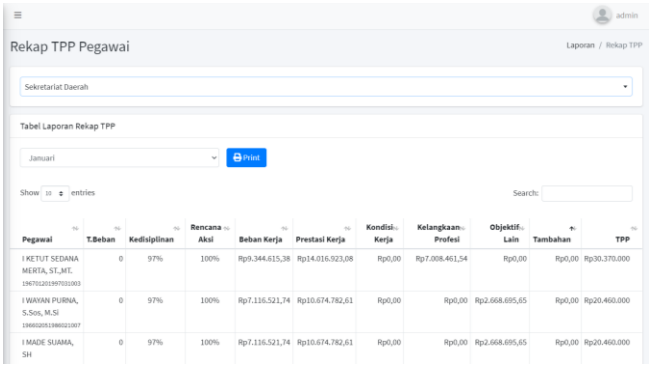


Figure 7 Additional Income Report

## *E. Maintenance Phase*

At this phase an update is carried out according to the feedback received. In a small release that runs through each iteration, there is some input from the user. Users can immediately provide input because a discussion group has been created to receive input and questions. Some of the implemented inputs such as: new assignment type to employee mapping, adding activity status filters, changing formulas to calculate additional employee income. These inputs are worked on as the development iteration progresses.

## *F. Death Phase*

The system is fully used. The distribution of additional employee income is completely done from the system. New ideas for the system other than the main features can no longer be added because of insufficient time and budget. So the final result of the system is determined. Documentation in the form of a final report for government administration purposes. As well as ideas that cannot be developed yet will be developed in the following year's budget.

## V. CONCLUSION

The results of the development of the additional employee income system using the XP approach show that with the XP perspective it is successful in building a system in government with not much time and money. Changing policies can also be handled quickly. The iteration is very helpful in terms of processing time. Because every small development and release iteration occurs at the same time. The closeness of the developer to the client is also very influential in the perfection of system development, because incoming input is handled quickly.

## ACKNOWLEDGMENT

Thank you to the BKPSDM of Karangasem Regency, for being willing to provide data for the needs of this research.

## REFERENCES

[1] Anonim, "Peraturan Menteri Negara Pendayagunaan Aparatur Negara Nomor 15 TAHUN 2008 tentang Pedoman Umum Reformasi Birokrasi." Jakarta, 2008.

[2] Anonim, "Keputusan Menteri Dalam Negeri Nomor 061-5449 Tahun 2019 Tentang Tata Cara Persetujuan Menteri Dalam Negeri Terhadap Tambahan Penghasilan Pegawai Aparatur Sipil Negara di Lingkungan Pemerintah Daerah." Jakarta, 2019.

[3] P. W. Kastawan, D. M. Wiharta, and M. Sudarma, "Implementasi Algoritma C5.0 pada Penilaian Kinerja Pegawai Negeri Sipil," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 3, p. 371, 2018, doi: 10.24843/mite.2018.v17i03.p11.

[4] A. Laksmidewi, L. Linawati, and W. I.M.O, "Evaluasi Sistem Informasi Manajemen Kepegawaian dengan DS5 dan DS9 COBIT 4.1 Studi Kasus : Pemprov Bali," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 1, p. 25, 2017, doi: 10.24843/mite.2018.v17i01.p04.

[5] F. Anwer, S. Aftab, S. Shah Muhammad, S. Shah Muhammad Shah, and U. Waheed, "Comparative Analysis of Two Popular Agile Process Models: Extreme Programming and Scrum," *Int. J. Comput. Sci. Telecommun.*, vol. 8, no. 2, pp. 1–7, 2017, [Online]. Available: www.ijcst.org.

[6] F. Anwer and S. Aftab, "SXP: Simplified Extreme Programing Process Model," *Int. J. Mod. Educ. Comput. Sci.*, vol. 9, no. 6, pp. 25–31, 2017, doi: 10.5815/ijmecs.2017.06.04.

[7] M. S. Widodo, "Requirements management pada extreme programming," *Semin. Nas. Apl. Tenologi Inf.*, vol. 2006, no. Snati, pp. 95–100, 2006.

[8] S. Janti, G. Taufik, and I. Komarudin, "XP Model for Information Systems Recording Academic Data," *SinkrOn*, vol. 3, no. 2, p. 242, 2019, doi: 10.33395/sinkron.v3i2.10086.

[9] N. Ramadan Darwish, "Improving the Quality of Applying eXtreme Programming (XP) Approach," *IJCSIS) Int. J. Comput. Sci. Inf. Secur.*, vol. 9, no. 11, 2011, [Online]. Available: http://sites.google.com/site/ijcsis/.

[10] K. Beck, *Extreme Programming Explained: embrace change, Second Edition*. 2004.