

METHODOLOGY OF MOBILE COMPUTING IN CONSTRUCTION MANAGEMENT

Kenji Kimoto, Mitsuhiro Fujiwara and Takahiro Kataoka
Research Institute of Technology, Konoike Construction Co., Ltd.
{kimoto_kj, fujiwara_mh, kataoka}@konoike.co.jp

Shuichi Matsumura
Department of Architecture, School of Engineering, The University of Tokyo
smatsu@buildcon.t.u-tokyo.ac.jp

Kazuyoshi Endo
Department of Architecture, Kogakuin University
endo@sin.cc.kogakuin.ac.jp

Abstract: This paper reports the development of mobile computing in construction management. The paper focuses on the methodology of the development based on EUC. First, the paper indicates the effectiveness of EUC in construction projects. The authors introduce design pattern of user interface and the use of UML for EUC. Secondly, the paper describes the process of development of PDA Inspection Support System for Finish Work with UML. User interface is an interface between users and computer-aided system, and can be one between users and system developers. In the developed system, user interfaces are classified to some design patterns with the use of control function such as list, option button and scroll bar. End users can customize and combine them freely. As a result, they can make their own flowcharts. System developers can build computer-aided systems with them easily and quickly. The paper also shows the effectiveness of the system. Finally, the paper summarizes this methodology and concludes the effectiveness.

Keywords: Construction Management, Mobile Computing, PDA (Personal Digital Assistants), EUC (End User Computing), Design Pattern, UML (Unified Modeling Language)

1. INTRODUCTION

Construction managers have recently handled various types of digital information such as drawings, specification, checklists and daily reports [2, 4]. However, they need to access the real construction site to manage the construction project. They usually use sheets of paper and/or field notes. As a result, they still have a lot of typical and routine jobs in construction site, such as the collection of construction data and the inspection. A gap in time and space between the outdoor construction site and the office occurs [4].

Now the current mobile computing can improve the field work in construction, and enhance the total productivity of construction management. The authors have developed the mobile computer-aided engineering systems, as follows: Inspection System, Position Check System, and Progress Monitoring System [4]. In the paper, the authors made clear the necessary functions for the system through the interview with construction managers, and adopted PDA as a mobile computer. The mobile system is composed of two programs: the data input program

in PDA and the output and analysis program in the spreadsheet application software of PC. User interfaces in PDA are mainly for data input, and answer various kinds of data input such as text and number.

On the other hand, construction projects have unique features and limitations. Computer-aided engineering systems for construction projects need the flexibility for each of projects. The authors recognize that the concept of EUC is useful and necessary to achieve it. However, there has been no useful methodology to realize EUC in the development of computer-aided engineering systems for construction projects. There has been no useful tool for it.

This paper aims to establish the methodology of the development of computer-aided engineering systems. The authors indicate two points for EUC. One is the mechanism to acquire the flexibility of system. They pay attention to design pattern of user interface. The other is a useful communication tool between end users and system developers. Common language between them is necessary. They pay attention to the Unified Modeling Language (UML).

2. EUC AND MODELING

As mentioned above, the authors recognize that EUC is useful as a concept corresponding to acquire the flexibility of system. However, it is still unrealistic that end users, construction managers, develop the whole of the computer-aided engineering system for the project. In general, their current literacy for the development of the system is not enough.

The authors have developed the systems by the collaboration of end users and system developers [4, 5]. The development is based on the EUC. EUD (End User Development) may be suitable for this process. That is, system developers prepare all-purpose components of the system in advance. End users customize and combine them freely to execute the construction rationally and effectively. System developers complete the computer-aided engineering system under the basic design. They can develop it quickly and achieve the flexibility for each.

2.1 Main Points in Modeling based on EUC

Through the above development of system, the authors understand that the modeling, which both end users and system developers can understand, is necessary. There are two points in the modeling.

One is the mechanism to acquire the flexibility of system. The authors think that it is design pattern of user interface. It is necessary that end users can recognize the functions, controls, and behaviors of model. It is also necessary that they can design the basic structure of the system with the model. On the other hand, it is necessary that system developers can build the system with the model. In other words, the important object in modeling for EUC will be “the object of control” from the view point of end users, and be “the base of design” from that of system developers. Design pattern of user interface can satisfy both requirements.

The other is how to communicate between end users and system developers for the development of the system. Common language between end users and system developers is necessary. A useful communication tool not only between end users and system developers but also among system developers is also necessary.

UML offers nine graphical diagrams that describe the computer system from various kinds of viewpoints. There is some possibility of using UML as the useful communication and design tool.

2.2 Application of UML

The UML is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system. UML offers a standard way to write a system’s blueprints, including conceptual things such as business processes and system functions as well as concrete

things such as programming language statements, database schemas, and reusable software components [7]. UML defines some diagrams to describe complex systems at different views.

The authors arranged UML diagrams and set up the modeling flow for this research. Figure 1 shows the flow and the role of each diagram. In this modeling, users are end users or system developers. They are classified in three broad categories. They are basically in ascending order, and are sometimes in parallel one.

The first category is the case study.

- 1) Description of Use Case: Users arrange actual use cases and define the relationship among them with use case diagram. As a result, the scope of model and system will become definite.
- 2) Description of Process Case: Users describe a scenario for each use case. Scenario is not a UML diagram. However, the research in JAHIS [3] uses it effectively. It will help the structure of model behavior. So, the authors adopt it.
- 3) Description of Object Case: Users describe the object diagram for each use case. This will help the construction of class diagram.

The second category is the description of model behavior. In other words, it is the description of dynamic structure of system.

- 4) Description of Model Process: Users describe the work flow of the model with activity diagram. This will help the definition of scope of computer-aided systematization.
- 5) Description of Message Flow: Users describe the message flow in the model with sequence diagram. This will help the definition of structure of system.
- 6) Description of GUI Flow: Users describe the GUI flow with sequence diagram and/or GUI screen. The authors understand that GUI is an important component in EUC, and put it in the modeling in this research, even though it is not a UML diagram.
- 7) Description of State Transition: Users describe the state transition of the model with statechart diagram. This is especially useful for the description of the repetitive procedure.

Case Study	1	Use Case Diagram	Description of Use Case
	2	Scenario	Description of Process Case
	3	Object Diagram	Description of Object Case
Model Behavior	4	Activity Diagram	Description of Model Process
	5	Interaction Diagram	Description of Message Flow
	6	GUI Flowchart	Description of GUI Flow
	7	Statechart Diagram	Description of State Transition
System Structure	8	Class Diagram	Description of Class Structure
	9	Component Diagram	Description of Software Structure
	10	Deployment Diagram	Description of Processor Structure

Figure 1 Modeling Flow with UML

The last category is the description for the implementation of system.

- 8) Description of Class Structure: Users describe class structure with class diagram. Class is an abstract concept and is refined through the description of case studies and model behavior.
- 9) Description of Software Structure: Users describe the dependencies among software with component diagram.
- 10) Description of Processor Structure: Users describe the configuration of run-time processing elements and the software components.

3. MODELING WITH UML

The paper explains the modeling for the case of the finish work inspection with PDA Inspection Support System for Finish Work, especially for dwelling apartment building projects. The first step of modeling in this research is case study.

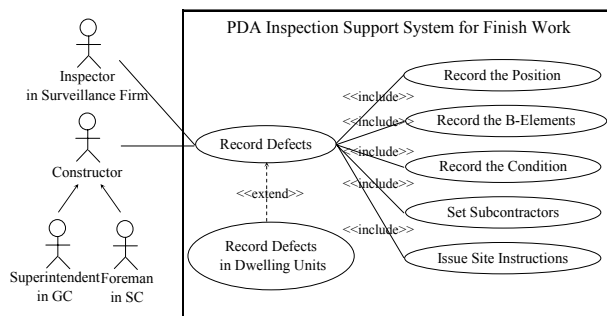


Figure 2 Use Case Diagram

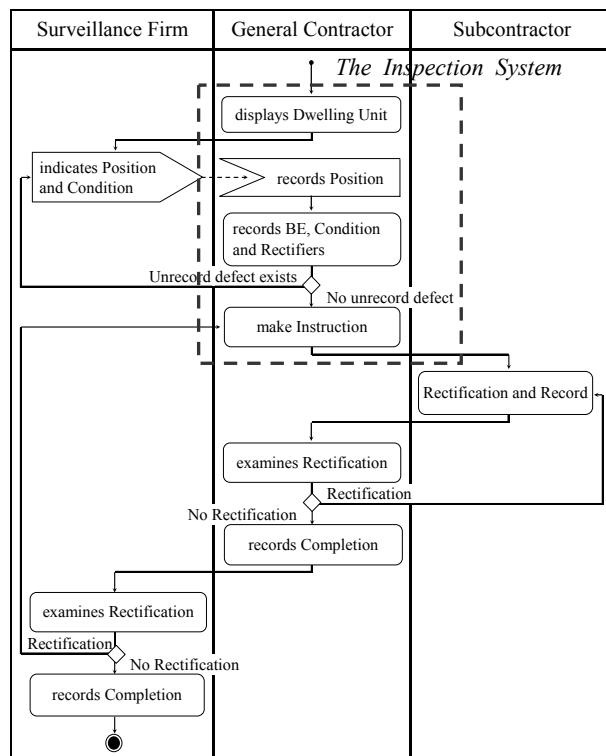


Figure 4 Activity Diagram

3.1 Use Case and Scenario

Figure 1 shows a use case diagram in the case. It shows that an inspector in surveillance firm, a superintendent in general contractor and a foreman in subcontractor can be the user of the PDA system. The inspection work with the system includes five subworks such as “record the position” and “issue site instruction”. It is extended from the work of “record defects in dwelling units.” The inspection for common corridor is possible as another use case.

Figure 2 shows a scenario in the case that an inspector in surveillance firm inspects dwelling units. In this scenario, a superintendent in general contractor uses the PDA system, and issues site instruction for the rectifying. A rectifier is a foreman in subcontractor.

3.2 Flow of Activity and Message

Figure 4 shows an activity diagram in the case.

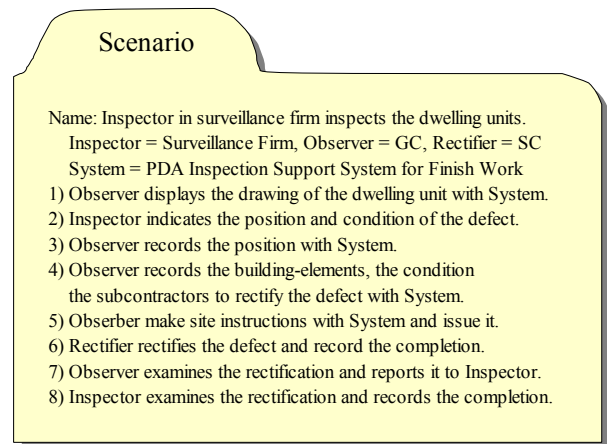


Figure 3 Scenario

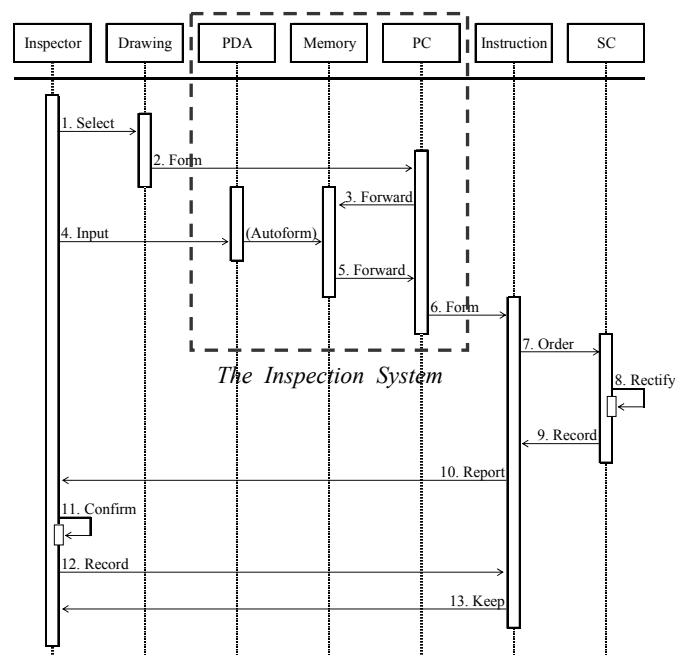


Figure 5 Sequence Diagram

It basically indicates a flowchart in the case. The flow and sequence of works make clear. It also sets the lane for affiliates. Each role makes clear in the inspection. Moreover, it can express the decision when guard conditions are used to indicate different possible transitions that depend on Boolean conditions [7]. Works can be classified by the type of content such as decision making and data input. This classification is related to the scope and role of computer-aided systematization.

Figure 5 shows a sequence diagram in the case. It focuses on the message flow. The related subjects are the sender and the receiver on messages. They include engineers such as inspector, and constituents of the system such as PDA. This diagram is also related to the scope and role of computer-aided systematization.

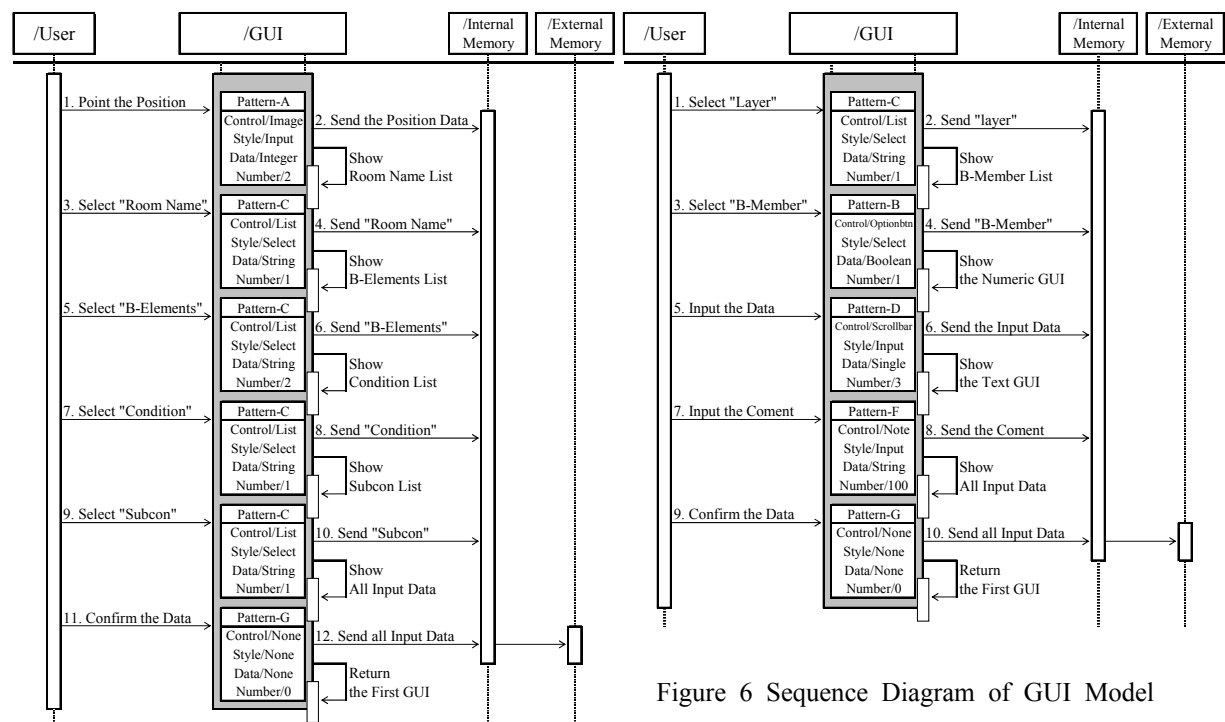


Figure 6 Sequence Diagram of GUI Model

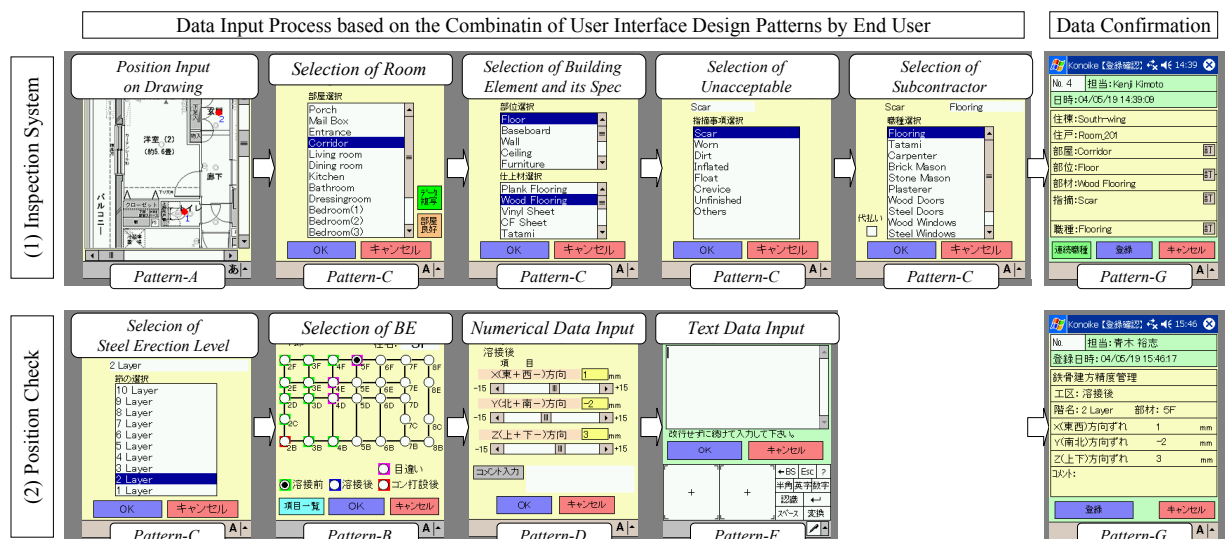


Figure 7 Development of GUI by GUI model

3.3 Design Pattern in GUI

The authors recognize GUI as the important object of design pattern for EUC. It is familiar to both end users and system developers. The left sequence diagram in Figure 6 shows message flow around a core of GUI model in the case. This GUI model, which the authors have arranged for EUC, has five properties. The first one is the name such as pattern-A and pattern-B, which is just for the identification. The second one is the type of control such as list and scrollbar. In this case, this means a tool in data input. The third one is the type of access to GUI such as the selection of item and the direct input of information. The forth one is the type of data such as Integer, String and Boolean. The fifth one is the number of data input.

The type of control and that of access are related to that of data. For instance, the selection from the list is suitable for the input of the item. The use of scrollbar is suitable for the input of numerical data, especially by users with gloves out of doors. This is possible in construction site. As sequence diagram shows message flow, the input and output in the system makes clear. The content of this GUI model is also familiar to both end users and system developers. Therefore, this flow can be designed by the collaboration of end users and system developers.

The right one in Figure 6 shows the case of Position Monitoring System with PDA. This case uses the right control such as list, option button and scrollbar, in the right place. Once the sequence diagram of GUI model is established, the development of actual GUI is easy and doesn't need much time. Figure 7 shows the development of actual GUI in the system. The upper flow of GUI shows the case of Inspection System. The lower one shows that of Position Monitoring System. These cases well express the feature of data input in PDA systems.

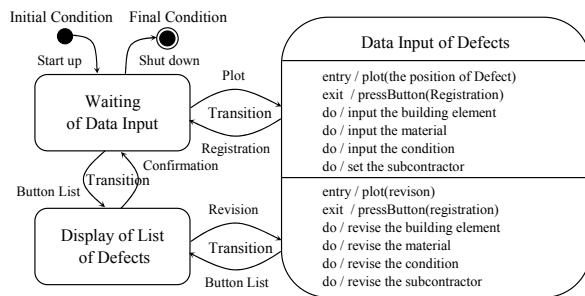


Figure 8 Statechart Diagram

3.4 State Transition in GUI

Actual use of the system is the process of not sequential, but trial and error. Users will go ahead and back in the process. The system must correspond to this situation. Figure 8 shows the statechart diagram of GUI in the case. Basic states are three: the waiting state of data input, the inputting state of data, and the reading state of list. This diagram can be a users' manual and specification for the development. Figure 6 and 8 are complementary each other.

3.5 Improvement of System

The UML is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system [7]. Therefore, it can express the difference among systems visually. There is some possibility of expressing the effect of the introduction of system, and that of the improvement of one. Figure 9 shows sequence diagrams of three models about the inspection work. Model 1 is the traditional way with paper sheets for both the data input and the rectifying order. Model 2 uses the PDA system for the data input, and paper sheets for the order. Model 3 uses the PDA system and the network for both the data input and the order. The work volume of GC decreases at the transition from Model 1 to Model 2. To make clear the effect, the authors divide a model to two domains: man area and machine one, and classify information to two domains: analog information and digital one. Through the improvement, machine area increases and is refined. The information is digitized.

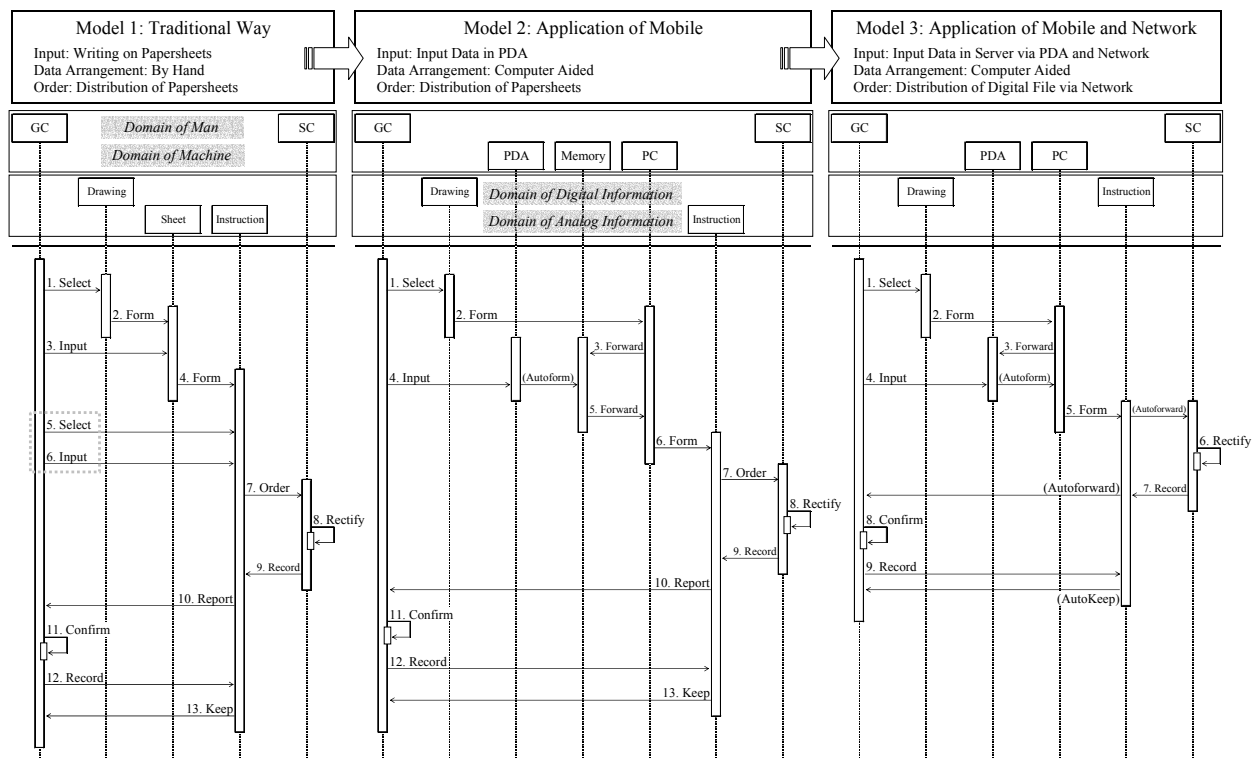


Figure 9 Comparison of Model by Sequence Diagram

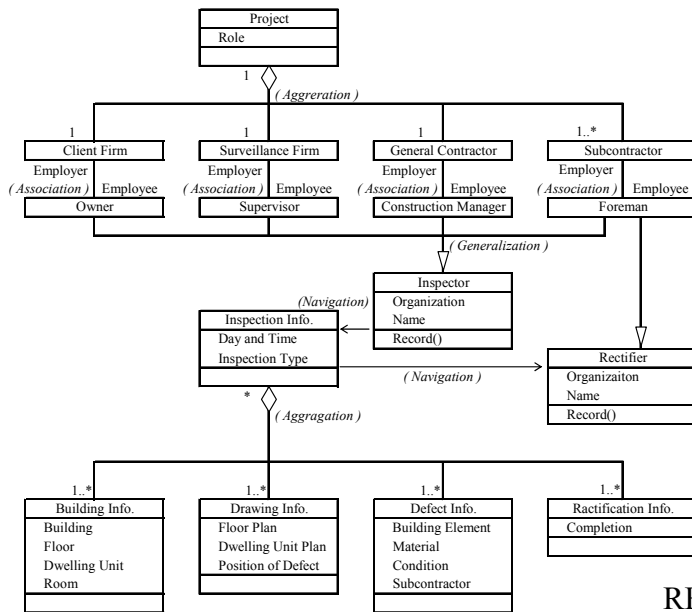


Figure 10 Class Diagram

3.6 Static System Structure

Figure 10 shows a class diagram in the case. A class diagram is a graph of classifier elements connected by their various static relationships [7]. The diagram shows the structure of elements including the relation among the elements. Defect information is composed of building information, drawing information, defect information, and rectification information.

Figure 11 shows a deployment diagram in the case. Processors are PDA and PC. It also expresses component diagram. Software in both processors depends on operating systems.

4. SUMMARY AND CONCLUSION

This paper describes the methodology of the development of mobile computing system based on EUC. The authors see the advantages of design pattern of user interface as the mechanism to acquire the flexibility of system. Moreover, they see that of UML as the communication tool not only between end users and system developers but also among system developers.

The paper introduces the description of the development process of PDA system with UML. Sequential diagram of GUI model well expresses the feature and design pattern of user interface. End users can take part in the design, and system developers can execute the coding of real system with it. That is, it works as modeling based on EUC. Moreover, the comparison between models with sequential diagram well expresses the effect of the introduction of computer-aided system, and that of the improvement of one. Other diagrams in UML also express the model from various kinds of viewpoints that are useful for the development of system.

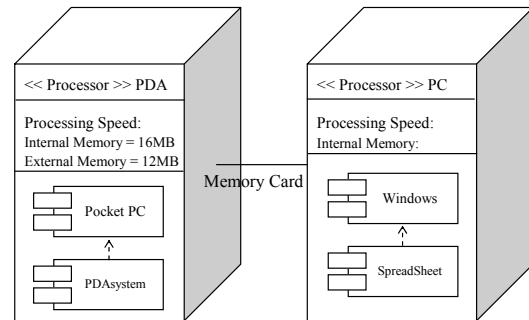


Figure 11 Deployment Diagram

REFERENCES

- [1] Fayad, M., and Schmidt, D. C., *Object-Oriented Application Frameworks*, The Communications for ACM, Special Issue on Object-Oriented Application Frameworks, Vol. 40, No. 10, 1997
- [2] Garrett, J. H. Jr., Burgy, C., Reinhardt, J., and Sunkpho, J. (2002). *An Overview of the Research in Mobile/Wearable Computer-Aided Engineering Systems in the Advanced Infrastructure Systems Laboratory at Carnegie Mellon University*, Bauen mit Computern 2002, Bonn, Germany. VDI Verlag GmbH, Duesseldorf, Germany
- [3] Japanese Association of Healthcare Information Systems Industry, *4th Modeling Workshop –Ward Information Model-*, JAHIS, 2001
- [4] Kimoto, K., Endo, K., Iwashita, S., and Fujiwara M., *The Application of Personal Digital Assistants as Mobile Computing Device on Construction Site*, Proceedings of the 20th International Symposium on Automation and Robotics in Construction, IAARC, Eindhoven, The Netherlands, 2003
- [5] Kimoto, K., Matsumura S., and Fujiwara, M., *The Application of Unified Modeling Language in the Development of Computer-Aided Engineering System*, Proceedings of Twentieth Symposium on Building Construction and Management of Projects, AIJ, Kyoto, Japan, 2004
- [6] Magdic, A., and Suman, N., *Efficient Support for Mobile Computing on Construction Site*, Proceedings of the 20th International Symposium on Automation and Robotics in Construction, IAARC, Eindhoven, The Netherlands, 2003
- [7] Object Management Group, Inc., *OMG Unified Modeling Language Specification*, OMG, 2003