TESIS - IS185401

KEPUTUSAN HARGA DAN STRATEGI UNTUK PRODUK PERANGKAT LUNAK DALAM KEHADIRAN PERUBAHAN KEBUTUHAN

PANDU DWI LUHUR PAMBUDI

NRP. 05211750012007

DOSEN PEMBIMBING:

CHENG-HAN WU., Ph.D.

MAHENDRAWATHI ER., ST., M.Sc., Ph.D.

NIP. 197610112006042001

PROGRAM MAGISTER

DEPARTEMEN SISTEM INFORMASI

FAKULTAS TEKNOLOGI ELEKTRO DAN INFORMATIKA CERDAS

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

2020

(This page is intentionally left blank)

TESIS - IS185401

PRICING DECISIONS AND STRATEGIES FOR SOFTWARE PRODUCTS IN THE PRESENCE OF REQUIREMENT CHANGES

PANDU DWI LUHUR PAMBUDI

NRP. 05211750012007

SUPERVISOR:

CHENG-HAN WU., Ph.D.

MAHENDRAWATHI ER., ST., M.Sc., Ph.D.

NIP. 197610112006042001

POSTGRADUATE PROGRAM

DEPARTEMENT OF INFORMATION SYSTEM

FACULTY OF INTELLIGENT ELECTRICAL AND INFORMATICS TECHNOLOGY

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

2020

(This page is intentionally left blank)

國立成功大學

碩士論文

具需求變更下軟體商品定價決策與策略之研究

Pricing Decisions and Strategies for Software Products in the Presence of Requirement Changes

研究生：潘陸宏

本論文業經審查及口試合格特此證明

論文考試委員：董橋

吳沛儒

吳政翰

指導教授：吳政翰

系(所)主管：王惠嘉

中 華 民 國 109 年 5 月 30 日

ABSTRACT

The presence of software requirement changes (RC) during project development is a critical challenge for the developer to offer software contract designs. Because under the presence of RC, the decisions toward the contract offer will impact to project's price spent by the developers. Managers of software companies must decide what contract designs to offer to clients in the development of software. Abstracting from an example drawn from the software outsourcing industry, we exhibit three designs of software contracts incorporating fixed price and time-and-materials policies. Specifically, a software company offers a fixed-price but declines the modification for RC (Contract N), offers a fixed-price and agree to RC with additional charge (Contract W), or initially provides a fixed price and then charges an additional fee based on the time-and-material in response to RC (Contract P).

We examine the strategic choices of three contract designs in a two-period game. We carry out a full analysis of monopoly and duopoly models; we use the monopoly model as the base model to construct the duopoly model. Meanwhile, in the duopoly model, we capture nine combination scenarios between two developers. We characterize the conditions under which the contracts can be the best decision for developers in different competitive models with price as our decision variable. Furthermore, we provide managerial insights into contract strategies and developers' performance under the presence of RC. Our finding states if the level of the second period valuation due to additional RC will influence the price and profit depending on the contract designs and combination scenarios.

***Keywords*:** software industry; price competition; software contract design; requirements change

抽象

在项目开发过程中，软件需求变更（RC）的存在是开发人员提供软件合同设计的关键挑战。因为在存在 RC 的情况下，对合同要约的决定将影响开发商所花费的项目价格。软件公司的经理必须决定在软件开发中向客户提供哪些合同设计。从一个来自软件外包行业的例子中抽象出来，我们展示了三种设计的软件合同设计，这些设计结合了固定价格和时间与材料政策。具体来说，一家软件公司提供固定价格，但拒绝对RC的修改（合同N），提供固定价格并同意RC并收取额外费用（合同 W），或者最初提供固定价格，然后收取额外费用基于响应 RC 的时间和材料（合同 P）。

我们在两个周期的游戏中研究了三个合同设计的战略选择。我们对垄断和双头垄断模型进行了全面分析；我们使用垄断模型作为基础模型来构建双头垄断模型。同时，在双头垄断模型中，我们捕获了两个开发人员之间的九种组合方案。我们以价格为决策变量，描述了在何种竞争条件下合同可以成为开发商最佳决策的条件。此外，在存在 RC 的情况下，我们提供了有关合同策略和开发人员绩效的管理洞察力。我们的发现表明，由于附加的 RC，第二期的水平是否会根据合同设计和组合方案而影响价格和利润。

关键字：软件业；价格竞争；软件合同设计；需求变更

ACKNOWLEDGEMENT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my research work to complete the research successfully.

I would like to express my deep and sincere gratitude to my research supervisor, Professor Cheng Han Wu., Ph.D. for giving me the opportunity to do research and providing invaluable guidance throughout this research. His dynamism, vision, sincerity, and motivation have deeply inspired me. He has taught me the methodology to carry out the research and to present the research works as clearly as possible. It was a great privilege and honor to work and study under his guidance. I am extremely grateful for what he has offered me. I would also like to thank him for his friendship, empathy, and dedication. I also would like to express my gratitude to my supervisor from ITS (Institut Teknologi Sepuluh Nopember) Dr. Mahendrawathi ER for guiding me to develop the research ideas and to do the scientific thinking and writing.

I am extremely grateful to my parents, my brother and sister for their love, prayers, caring and sacrifices for educating and preparing me for my future. My Special thanks goes to my Sustainable SCM Lab friends (Jeff, Harry, Eric, and Long) for the helps shown to complete this thesis successfully. Also, P’Song and all Indonesian friends here, I love you all and thank you.

Pandu Dwi Luhur Pambudi

CONTENTS

LIST OF TABLES

LIST OF FIGURES

(This page is intentionally left blank)

1. INTRODUCTION

1.1 Research Background

In every software development project, we use contracts to regulate project work and results. Throughout substantial evidence and proven theories over decades, some experts introduce and promote success variety of contract types. However, two contract types have dominated and still dominate most work of software projects: fixed price and time and materials types of contracts. The two contracts are frequently used in the software industry, as discussed in (Fink et al. 2013; Gopal et al. 2003; Jørgensen et al. 2017).

Jørgensen et al. (2017) state that the fixed price type of contract is the condition when the client agrees to pay the developer a specific price for a delivered software product. Meanwhile, a time and materials type of contract is when the client agrees to pay for the effort spent by the developer, usually based on the price agreed by both sides per hour of work for different types of competence and other necessary expenditures. In the fixed price contract, the price for completing the project is predetermined in advance. Conversely, Fink et al. (2013) explore that time and materials contract does not specify a price, but rather reimburse the vendor for its costs plus a predetermined profit.

Developers usually compete in offering low prices and excellent competence, and the client will choose one developer with satisfactory competence and the lowest price. Unfortunately, based on the practice, fixed price and time and materials contracts still unable to accommodate the presence of Requirement Change (RC). RC is a particular software development activity and can occur due to changes in user requirements (Ali & Lai, 2016). RC is considered a significant source of risk because it will increase the budget overruns (Nurmuliani et al. 2006). That is why, in most cases, a complete software specification and final contract are difficult to achieve due to the presence of RC. It is almost impossible to achieve optimal and complete software quality when there is a fixed price for variable content. In fixed price contract, price and requirements specifications are entirely predetermined in advance, and contracts do not include a mechanism to accommodate the RC (Fink et al. 2013).

The contracting consequences of disallowing change are essential because ex-post negotiation is costly and creates a contractual hazard (Bajari, 2001; Bolton & Dewatripont, 2005). Although price and specifications are entirely predetermined in advance, the fixed price contract includes a clause with an explicit change management procedure that allows changing both specifications and prices. Managers in the industry explain that it is standard practice to include a change provision in fixed price contracts, as is indeed reflected in the research (Chen & Bharadwaj, 2009; Sia et al. 2008). Furthermore, Corts (2012) examines if clauses explicitly open the possibility of an augmented fixed price contract that allows the developer to perform modifications on a time and materials basis.

Meanwhile, when the client chooses a time and materials contract, the presence of RC will be increasingly high because client can freely request the additional RC. In the time and materials contract, requirements specification is not sufficiently detailed and the price for project completion is not constrained. The primary benefit of this contract types relative to a fixed price contract. This contract can reduce negotiation cost (Bajari, 2001). The developer is more likely to accept changes requested by the client without the need for renegotiation (Kalnins & Mayer, 2004).

1.2 Research Objective

Despite the extensive study in software contract, the issues related to contract choice and performance for software projects remain unclear (Dey et al. 2010). The high complexity in software engineering processes makes software contracts pose many unique challenges that are usually not seen in other industries. We capture two main problems that challenge developer in offering a software contract under the presence of RC.

Incomplete requirement specification and difficulty in the quality assessment are only two of these challenges (Dey et al. 2010). This means that accommodating the incomplete requirement specification, such as the initial requirements and additional RC are included as the challenges. These problems could make software contract challenging to manage by often incurring significant cost overruns from RC.

In a software contract design, it does matter to know the consequences of which type of contracts to propose to the client. Importantly, we need to analyze how contracts are designed and the extent to which contract designs are better to deal with the presence of RC. Especially understanding the linkage between contract designs and profits under the presence of RC. Based on this motivation, we examine the adoption of the two contracts in the software industry: fixed-price contract and time-and-materials contract. We propose and develop the fixed price contract into two contract designs, which are Contract N under the pure fixed price contract mechanism, Contract W under the combination between fixed price and time and materials mechanism with single pricing and Contract P under the time and materials mechanism with two pricings. Our research contributions are threefold, as explained below:

1. **Investigate the presence of RC in different software contracts under the monopoly case.** This study is one of the papers to analytically study the determinant of contracts choice in the software industry as a monopolist, including (Dey et al. 2010; Dharma et al. 2010; Oh et al. 2016). Although there are some other discussions of the contract types using empirical methods in the (Fink et al. 2013; Gopal et al. 2003; Jørgensen et al. 2017; Kalnins & Mayer, 2004; Suprapto et al. 2016).
2. **Explore the impacts of competition of two developers under the duopoly case.** We discuss this study under a duopolistic scheme between Developer A and Developer B. We propose the hoteling model-game theory that previously

3. never been addressed in the software contract as a competitive option between two software developers. We build nine different combination scenarios of software contract designs under competition between two software developers.
4. **Analyze the performance of varying software contracts both in monopoly and duopoly cases.** We address the linkage between contract designs, project prices and profits under the presence of RC to understand the consumer behavior for developer's strategic decisions. We provide analytical and numerical analysis to investigate the best response of contract choice on the equilibrium price.

We compare our study with the prior studies in (Dey et al. 2010; Dharma et al. 2010; Oh et al. 2016). Dey et al. (2010) focus on comparing two software contracts fixed price and time and materials under uncertainties and focus on the analytical analysis of effort and time. Meanwhile, our study focuses on pricing decision under the presence of RC. We also model a competitive form of duopolistic market which competing under different scenarios and use the hoteling model to capture the competition.

Meanwhile, Dey et al. (2010) do not spesifically construct their model under a competition. Dharma et al. (2010) discuss about the optimal of three contracts comparison fixed price, time-based, and cost-based contracts with the decision variable of work rate. They do not specify the project or case limited for software industry only but also for general projects. They also do not discuss the model under external competition. Meanwhile, Oh et al. (2016) propose a study on the impact of cost uncertainty on pricing decisions under risk aversion in services, generally they do not specify the service into software industry only but also another service industry. They also don't specify the contract types into fixed price or time and materials in their model because they discuss the contract in general. They use price as their decision variable by ignoring a market competition.

1.3 Research Process

Our study only focuses on contract designs for software development projects or commonly called software outsourcing in the software industry. We build this model regarding the developer's perspective, so we do not consider the client's perspective. From a software developer perspective, we only consider the model regarding the role position of a developer who offers the contracts for the general scope of the project. In this thesis, we propose a monopoly and duopoly model cases concerning contract profits with price as the decision variable. We aim to study this problem at a project level by modeling the characteristics of a software contract based on established theories in software development.

RC is directly linked with the decision making in software development. The developer must make decisions with respect to the presence of RC. Including how to decide the software pricing under the presence of RC. Of course, the developer must be careful to define the software pricing since the software pricing will influence the profit. Importantly, the developer must design the software contracts that consider the presence of RC.

The rest of the paper is organized as follows. Chapter 2 surveys the related prior studies while comparing these studies with our work. Chapter 3 describes the analytical model and proposes a game-theoretical approach for our duopoly model. Chapter 4 presents an analytical analysis of our results. Chapter 5 presents a numerical analysis and discusses about the findings. Chapter 6 concludes the paper with directions for contributions and future research. The diagram of the research flow is presented below (see Figure 1.1).

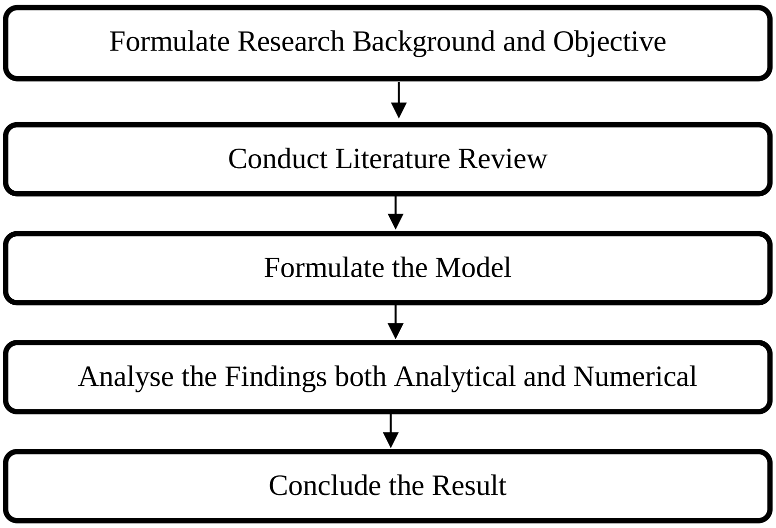


Figure 1.1 The Diagram of Research Flow

2. LITERATURE REVIEW

This study relates to three streams of literature: one examines the presence of RC in the software industry. Second provides a survey on software pricing, and the third investigates types of software contracts. We present the prior studies specifically the analytical and theoretical studies as follows.

2.1 RC in The Software Industry

In any software development activity, RCs are inevitable and can occur due to changes in user requirements, increase understanding of the stakeholders' needs, customer organizational re-structure, and availability of new technologies (Ali & Lai, 2016; Basri et al. 2016). With every change in requirements, a developer can be affected by the change of the overall cost, quality, and schedule of the software, which is why the RC as one of the significant causes of software failure Nurmuliani et al. (2006). However, dealing with RC not only poses a risk to the successful delivery of software but also provides an opportunity to improve usability, value, and enhance software development process (AlSanad & Chikh, 2015; McGee & Greer, 2012). Topics related to the first stream have been examined in the field of computer science and software development projects, e.g., (Gaebert, 2014; Mao et al. 2006; Tong et al. 2017). These topics mainly discussed how RC could affect software development by using the analytical methods.

Gaebert (2014) studies the customer and the supplier dilemma condition regarding the effort for closing the gaps of the RCs in the software industry. He suggests switching off the contract level, describing the interaction of the involved organizations in terms of game theory. It also carries out an empirical investigation that shows gaps in requirements and conflicts in the project. They provide a theoretical rationale for the failure of software development projects and analyze contractual situations for software projects concerning risks of failure.

Tong et al. (2017) focus on how the RC affected the software project, especially in the context of green software development in a game trade-off between client and software vendor using Nash Equilibrium. They find the RC might lead to unnecessary labor and time cost. Moreover, it might also result in the waste of hardware and computing resources once unreasonable requirements are realized. Thus, to perform green computing in software engineering, it is necessary to propose effective scenarios to manage the RC.

Mao et al. (2006) firstly construct an assessment framework for the factors of RC distribution. Apart from the rough prediction method based on the statistic process control of RC, an artificial neural network method for predicting RC' distribution is presented. In this case, the weight of each factor is calculated by a fuzzy logic method, called experts ranking. Furthermore, they propose a model to pre-evaluate the cost caused by RC.

In this study, we use the term of RC as a description in (Mao et al. 2006). RC is the number of changes (addition, deletion, and modification) in each period of the development life cycle. It means that every addition, deletion, and modification of software requirements are referred to as RC.

2.2 Pricing in The Software Industry

Software companies often struggle with communicating the value of their solution to their clients. As a result, developers will get less profit because they fail to estimate the work under the presence of RC. Many researchers tried to optimize and analyze software pricing with some scenarios. They develop a pricing scenario that is tailored to customer segments that optimizes a company's financial goals. Here we summarize the second stream surveys about pricing in the software industry. Studies related to this topic are broadly discussed in the field of information systems, operation management, and industrial management. e.g., (Bala & Carr, 2010; Cheng et al. 2015; Cheng & Tang, 2010; Choudhary et al. 2005; Liu at al. 2011; Mehra at al. 2012; Nan at al. 2016).

Nan et al. (2016) examine the duopoly model in which one firm adopt the limited-feature free trial scenario, and the other employ the seeding scenario in the software industry. They drive the equilibrium prices and profits of two firms and explore how the optimal prices and profits are affected by the quality and the service level of the free trial version, the network intensity, and the seeding ratio. Meanwhile, Cheng et al. (2015) develop an analytical model to examine software free trial scenarios limited version, time-locked, and hybrid. They find that the hybrid scenario weakly dominated the limited and time-locked versions, and the intensity of the network effects is a key factor determining which scenario is optimal in terms of the optimal price, quality level, and free trial time. Mehra et al. (2012) formulate a game-theoretic model for software product upgrade involving an incumbent and entrant where both firms can offer discounts to existing customers of the incumbent. Although several equilibrium possibilities exist, they establish that an equilibrium with competitive upgrade discount pricing is observed only for a unique market structure and a corresponding unique set of prices.

Cheng & Tang (2010) examine the trade-off between network effects and the cannibalization effect of the software product and aim to uncover the conditions under which firms should introduce the free trial product. They find that when network intensity is intense, it is more profitable for a software monopoly to offer free trial than to segment the market with two versions of different qualities. This paper also solves the joint decision problem of finding the optimal quality for the firm's free trial software and the optimal price of its commercial product. Choudhary et al. (2005) develop an analytical framework to investigate the competitive implications of personalized pricing (PP) in the software industry, whereby firms charge different prices to different consumers based on their willingness to pay. Liu et al. (2011) develop an analytical model that embedded empirical findings on software diffusion to examine optimal pricing scenarios for a spreadsheet software product under coalescing effects of piracy and word-of-mouth under multi-period multi-price software pricing. Bala & Carr (2010) explain the software as a service pricing scheme under fixed-price or usage price in duopoly competition. They only discuss the software as a readily used product and not discuss the contract pricing on a software development project. Rohitratana & Altmann (2012) present an agent-based simulation system that allow modeling the interactions between software buyers and vendors in a software market. The market offers Software-as-a-Service (SaaS), and perpetual software (PS) licensed under different pricing schemes. Four dynamic pricing schemes are analyzed: derivative-follower pricing, demand-driven pricing, skimming pricing, and penetration pricing.

2.3 Software Contract Policy

The third stream examines software contract policy. A software contract is a binding agreement between the software developer and the client in the outsourcing mechanism. The contract provides the development policy of the software to the client. A typical software development contract must deal with a variety of closely related issues such as the quality of the developed system, the timeliness of delivery, the effort and cost associated with the project, and the support (Dey et al. 2010). The economists have studied contract design for at least 40 years (see McCall, 1970) in (Lippman et al. 2013). Meanwhile, Kalnins & Mayer (2004) examine the use of fixed-fee and time-and-materials (or cost-plus) contracts and a hybrid contract that consisted of a time and materials contract with a cap. Previous related studies about contracts are considering a multitude of issues. They have produced an enormous literature with some emphasis on moral hazard, adverse selection, signaling, asymmetric information, contracting in a dynamic case, and contracting in competitive markets. We summarize three studies from (Dey et al. 2010; Dharma et al. 2010; Oh et al. 2016) that directly discussed the use of software contracts under different issues in analytical way. The studies can be found in the field of information systems, project management, and production economics.

Dharma et al. (2010) examine three types of project contracts commonly used in practice, fixed price, time-based (i.e., price depends on the realized project completion time), and cost-based (i.e., price depends on the actual cost). The study relates to the software projects with uncertainty and shows that fixed-price contracts and cost-plus contracts cannot coordinate a channel. Meanwhile, Oh et al. (2016) study about cost uncertainty in the services contract generally. They cite the software industry as one of the industries with cost uncertainty that impacted pricing decisions. They first identify the root causes of cost uncertainty in the services contracts and investigate how cost uncertainty affected a risk-averse service provider’s pricing decision in a make-to-order scenario. Using the expected utility theory framework, they show that cost uncertainty increases the optimal price, whereas demand uncertainty reduces it. Dey et al. (2010) present a contract-theoretic model that incorporate these factors to analyze how software outsourcing contracts can be designed. They find that despite their relative inefficiency, fixed-price contracts are often appropriate for simple software projects that require short development time. Meanwhile, time and materials contracts work well for more complex projects when the auditing process is efficient and effective. They also examine a type of performance-based contract called quality-level agreement and find that the first-best solution can be reached with such a contract. Finally, they consider profit-sharing contracts that are useful in situations where the developer has more bargaining power.

We summarize our survey of prior studies into the table as illustrated in Table 2.1. We survey prior studies based on the criteria set in the beginning through keywords searching from reputable journal databases. As we can see in Table 2.1, we highlight our study with a thick font that describes if this study has a novelty aspect compared to the previous studies. We consider the presence of RC and emphasis the duopoly market as the external competition which never been discussed before in the software development contract.

Table 2.1 Survey of The Previous Study

Author(s) (Year)	RC Presence	External Competition	Software Pricing	Software Contract Type		Decision Variable
				fixed price	time and materials	
This study	√	√	√	√	√	**Price**
Bala & Carr (2010)	–	√	√	–	–	Price
Tong et al. (2017)	√	–	–	–	–	RC
Oh et al. (2016)	√	–	√	–	–	Price
Liu et al. (2011)	–	–	√	√	–	Price
Choudhary et al. (2005)	–	√	√	√	–	Price
Cheng & Tang (2010)	–	√	√	–	–	Price, quality
Dharma Kwon et al. (2010)	√	–	–	√	√	Work rate
Mehra et al. (2012)	–	√	√	–	–	Price
Nan et al. (2016)	–	√	√	–	–	Price
Dey et al. (2010)	√	–	–	√	√	Effort, release time
Cheng et al. (2015)	–	√	√	–	–	Price, time

3. THE MODEL

3.1 Model Setup

We formulate our model into two different models of monopoly and duopoly. The first model is a monopoly model (see Figure 3.1). The developer will offer three different software contracts. As the model is built under a monopoly model, we do not consider the competition with other developers, because there are some cases when the client chooses directly one software developer to execute their project or there is only one biggest software developer in an area that monopoly the industry. We want to highlight under which contracts the developer as a monopolist can obtain the best profit.

The second model examines the software contracts competition in a duopoly market between two software developers. Developer A competes with Developer B in order to offer their software contracts. This duopoly competition usually happens when the client opens and invites some developers for their software outsourcing project. In this duopoly model, we want to know which scenarios role as the best contract decision for developers. We describe our duopoly model into the diagram as below (see Figure 3.2).

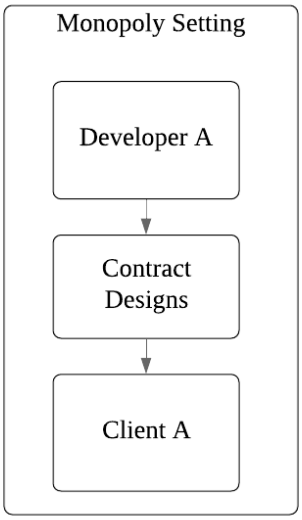


Figure 3.1 The Monopoly Model

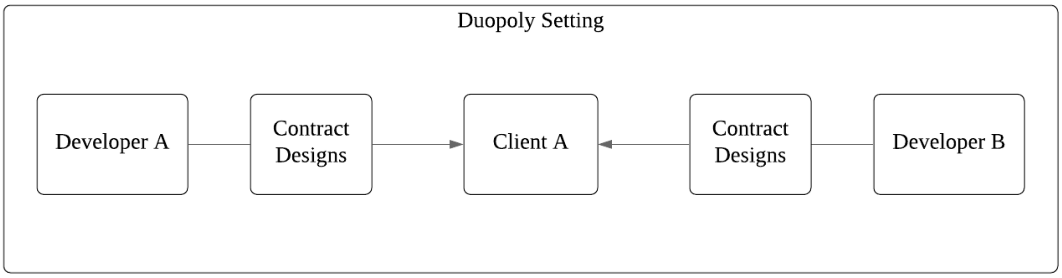


Figure 3.2 The Duopoly Model

We follow the prior studies in (Cheng &Tang, 2010; Dey et al. 2010; Nan et al. 2016) to construct our monopoly model and duopoly model.

3.1.1 Two-periods model

In this study, we set this model into two periods of contract offers. **In the first period** of the contract, the developer offers a work of initial requirement with the price to accommodate the initial requirement cost. Meanwhile, **during the second period** of the contract, the developer can provide additional RC work with charge to accommodate the additional RC after the first period is completed. We describe our period model in a decision tree in Figure 3.3.

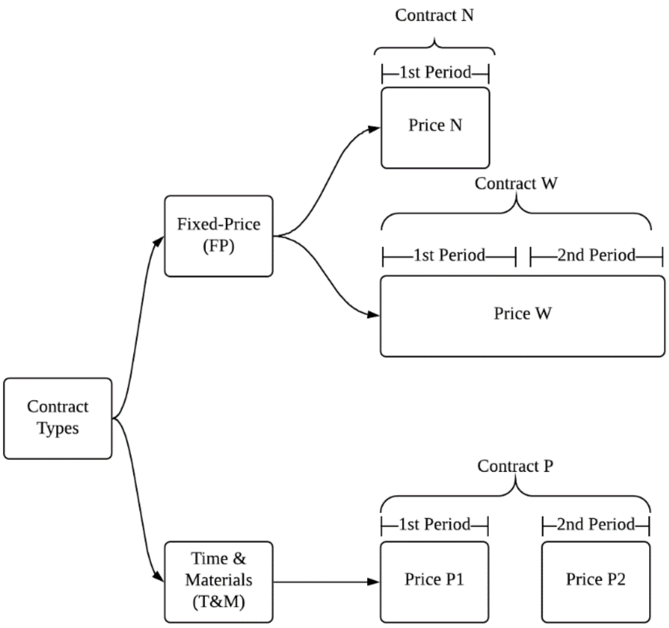


Figure 3.3 Contract Decision Tree

3.1.1 Contract Designs

In our monopoly and duopoly models, we introduce three contract designs, as we mentioned before. They are N, W, and P contracts. We describe the three contracts, one by one as follows:

(1) Contract N. This contract offers a pure fixed price contract with single pricing of a software product with the agreement between the developer and client in advance. This design includes only the initial requirement work in software development. As a result, the client cannot propose an additional RC. Thus, the developer only charges the initial requirement cost. The initial requirement work in software development only covers the basic features of the software. We characterize the software product based on the quality attribute that refers to the completeness of the software features. The completeness of the software features depends on the presence of RC. This statement is agreed upon by the previous studies in (AlSanad & Chikh, 2015; McGee & Greer, 2012). As we do not consider the second-period price for additional RC in Contract N, so the software quality is assumed to be lack features. However, in the real world of software outsourcing practice, almost impossible for the client not to propose the additional RC and only goes with the Contract N. Because the client will always tend to do additional RC in a software development process for the shake of software quality, this statement supported by (Dey et al. 2010; Ghosh et al. 2013). Based on that case, we formulate our second contract design that is Contract W.

(2) Contract W. This contract accommodates both works for the initial requirement work and additional RC work under the single pricing. This contract is a combination between fixed price and time and materials contracts. Because the developer also accommodates the additional RC, the developer will charge two costs, which are initial requirement cost and additional RC cost. Due to that condition, we expect the increasing level of software features as the impact of the work of additional RC in the second period. However, Contract W only proposes a single pricing to accommodate two costs. This condition stimulates us to know the impact of this scheme on client behavior. At last, to study and compare the performance between fixed price and time and materials contract, we design Contract P to represent time and materials contract.

(3) Contract P. This contract accommodates both works for the initial requirement work and additional RC work under the two pricing. This contract also charges a specific fix cost for additional RC allocation asides the additional RC cost because client can request additional RC in repeated times based on time and materials concept. Contract P refers to the concept of a time and materials contract, which is the more effort and time spent by the developer, the higher cost charged into it. In the whole of model, we assume if time and effort already accounted in costs. We model the price one to accommodate the initial requirement work and the price two to accommodate the additional RC work. The price two includes two costs: an additional RC cost and a specific fix cost for additional RC allocation.

The developer splits the contract into two periods. We assume that after completing the first-period work, the client will always go for the second-period work because the lack features of the software. Based on that condition, the client is in the provision of sunk cost. They do not have many choices in the first period to complete their software features due to the lack of requirements (client cannot propose for additional RC). So, the developer provides the second-period work to deal with the unexpected additional RC. According to that case, Contract P will be an appealing offer for the client who does not want to go for the N and W contracts.

In practice, a client tends to continue the contract from the same developer instead of changing it to another developer due to the competition cost to change to another developers. This condition happens due to the code structure of the previous software project. The code structure usually can be understood well by the same developer. When the client chooses a different developer to continue the project development, it will take another time and effort for the new developer to review and learn the code structure. This condition sometimes makes the new developer suggests a new software development from the beginning instead of continuing the previous project. That is why we assume the first period model as a private information of the developers and is not directly observable by the client. This is because in the first period the developers will only work for initial requirement which functionally lack of features (because developers do not accommodate the additional RC). We also assume if the project is always delivered on time.

3.1.2 List of Notation

We introduce the necessary notation in Table 3.1 for our monopoly and duopoly models as below:

Table 3.1 Monopoly and Duopoly Model Notation

U	Utility function
t	Misfit cost or traveling cost
p_j^i	Price for each software contract or scenario $i, i \in \{N, W, P, NP, WP, PP\}$, for developer $j, j \in \{A, B\}$.
S	Primary valuation
θ	Consumer type, i.e., consumer's preference or valuation
c_j	The cost charged for each period for developer $j, j \in \{A, B\}$.
δ	Second-period valuation due to the additional RC
D_j^i	Demand to accept the software contracts. $i, i \in \{N, W, P, NP, WP, PP\}$. For developer $j, j \in \{A, B\}$.
π_j^i	The expected profit for software contracts. $i, i \in \{N, W, P, NP, WP, PP\}$. For developer $j, j \in \{A, B\}$.
ρ	The discount factor for service B in Developer B
λ	Intertemporal value discount for the second period
γ	Sensitivity for the second-period price
F_j	A specific fix cost for additional RC allocation. $j, j \in \{A, B\}$.

3.1.3 List of Assumptions

We build models based on some assumptions that describe the theoretical approaches. Here is the list of assumptions for monopoly and duopoly models:

1. The information in the first period is private information of the developer and is not directly observable by the client.
2. We assume that the project is always delivered on time to the client, so there is no variable to decrease the utility of the software due to the project delay.
3. We assume that the cost parameters (c_A and c_B) in the model already account for the necessary discounting.
4. We normalized the demand into one
5. We assume that the quality attribute of software for the initial requirement already attached to the client's primary valuation S.
6. We assume that the quality attribute due to additional RC is embedded in δ.
7. We assume that the effort and time already attached to costs.

3.2 Monopoly Model

In this section, we study the behavior of a client when a monopolist is selling a software product by offering three contract designs. This monopoly model will only present as the base model for the duopoly model building. We derive the conditions under which the developer as a monopolist will offer the N, W, or P contracts to the client. Under Contract N and Contract W, we use only a single pricing model due to the explanation in Chapter 3.1. Otherwise, Contract P considers the two-pricing model. We follow prior works of (Dey et al. 2010) and (Cheng & Tang, 2010). We also use contract profit to measure the contract performance by maximizing the price.

Let θ denote the client's primary valuation to quality that closely related to the features of the software product. We assume that the quality of software for an initial requirement is attached to the client's primary valuation. We consider the benchmark cases where the developer offers the N, W, or P contracts. We build the model step by step by solving the utility function, formulating the demand, and solving the first-order condition of developer's problems. After that, we derive the optimal price for the developer's problems. We explain the steps as follows:

3.2.1 Utility and Demand Function

Let *N* be the number of clients with a positive valuation for the software contracts. We normalize *N* to 1 for the sake of simplicity and to capture the real event. The client's primary valuation for the software is denoted by (θ) and uniformly distributed over $[0,1]$. Therefore, it corresponds to the demand (D_A) for all the software contracts, a demand can be expressed as $D_A = 1 - \theta$. In our model, we assume clients always get positive valuation θ. Figure 3.4 shows the demand for the software contracts. We denote the demand $(D_A^N, D_A^W,$ and $D_A^P)$ as a demand for the client to choose the software contracts.

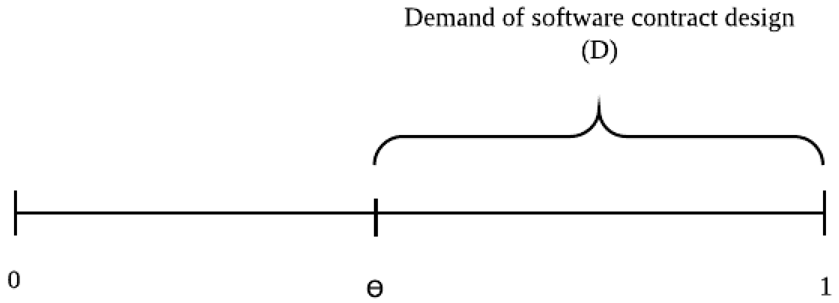


Figure 3.4 Demand for Software Contract in Monopoly

The monopoly offers one software product under Contract N with software utility $U = \theta - p_A^N$, Contract W with software utility $U = (1+\delta)\theta - p_A^W$ and Contract P with software utility $U = (1+\delta)\theta - p_A^{P1} - \gamma p_A^{P2}$. The software prices are stated as (p_A^N, p_A^W, and p_A^P). The primary valuation is influenced by the price, when the price is higher, the primary valuation is also higher, and it will increase the utility of the software contract.

Meanwhile, in the second period, when the additional RC is acted, the $(1 + \delta)$ added as second-period valuation due to RC. This means that the primary valuation (θ) in the first period will be upgraded with the second valuation of $(1 + \delta)$, so the primary valuation will be higher. Let (γ) as the sensitivity for the price two in the Contract P (p_A^{P2}). In this phase, we assume that the project is always delivered on time to the client, so there is no variable to decrease the utility of the software due to the project delay. Thus, the expected software contract utility for Contract N who has zero net utility can be expressed as $\theta - p_A^N = 0$. The software contract utility for Contract W who has zero net utility can be expressed as $(1+\delta)\theta - p_A^W = 0$, while software contract utility for Contract P who has zero net utility can be expressed as $(1+\delta)\theta - p_A^{P1} - p_A^{P2} = 0$. After that, we solve (θ) for each utility and get $\theta = p_A^N$ for Contract N, $\theta = \frac{p_A^W}{(1+\delta)}$ for Contract W and $\theta = \frac{p_A^{P1}+\gamma p_A^{P2}}{(1+\delta)}$ for Contract P. After solving the (θ), we get the demand for each contract as $D_A^N = 1 - p_A^N$, $D_A^W = 1 - \frac{p_A^W}{(1+\delta)}$, and $D_A^P = 1 - \frac{p_A^{P1}+\gamma p_A^{P2}}{(1+\delta)}$. We discuss the monopoly model of the N, W, and P contracts step by step and the calculation of the optimal solution for the price for a developer's problem under the N, W, and P contracts are provided in Appendix A. Throughout the paper, we present the equilibrium results with the asterisk symbol (*).

3.2.2 Contract N

The monopoly developers seek to find the optimal price to maximize their profits by solving the following problem. We model the developer's problem under Contract N, as follows:

$$\underset{p_A^N}{Max}\ \pi_A^N = [(p_A^N - c_A)]D_A^N. \tag{1}$$

Let (p_A^N) as the software price for Contract N and the initial requirement cost incurred by the developer is denoted with (c_A). We simply set the initial requirement cost (c_A) without discounting. This is because we assume that the cost parameter (c_A) in the model already account for the necessary discounting. We denote demand for Contract N as (D_A^N). In order to get the optimal price, we derive the first-order condition for the developer's problem with respect to (p_A^N) as follows:

$$\frac{\partial \pi_A^N}{\partial p_A^N} = (p_A^N - c_A)(1 - p_A^N) = 0, \tag{2}$$

and from Eq. (2), we get the optimal price $(p_A^{N^*})$ for the developer's problem under Contract N as follows:

$$p_A^{N^*} = \frac{1}{2}(1 + c_A). \tag{3}$$

Because we only consider price as our decision variable, we derive the second-order condition for developer's problem to see whether the objective function in $(p_A^{N^*})$ is concave or convex. We get the result in the second derivation of the objective function in $(p_A^{N^*})$ is less than 0. $\frac{\partial^2 \pi_A^N}{\partial p_A^{N^2}} = -2 < 0$. Therefore, the objective function is concave and is the optimal solution.

Contract N shows that the developer only facilitates the initial requirement work of software development under the initial requirement cost (c_A) in the first period. Because the development cost only covers initial requirement work, the software price is equal, with half of the initial requirement cost.

3.2.3 Contract W

The monopoly developers seek to find the optimal price to maximize their profits by solving the following problem. We model the developer's problem under Contract W, as follows:

$$\underset{p_A^W}{Max}\ \pi_A^W = [(p_A^W - 2c_A)]D_A^W. \tag{4}$$

Let (p_A^W) as the software price for Contract W and the initial requirement cost incurred by the developer and the additional RC cost are denoted with c_A. Because this contract covers two works, so the cost to accommodate it will be doubled. The cost (c_A) under this contract is also already account for the necessary discounting.

We denote demand for Contract W as (D_A^W). In order to get the optimal price, we derive the first-order condition for the developer's problem with respect to (p_A^W) is as follows:

$$\frac{\partial \pi_A^W}{\partial p_A^W} = (p_A^W - 2c_A)\left(1 - \frac{p_A^W}{(1+\delta)}\right) = 0, \tag{5}$$

and from Eq. (5), we get the optimal price $(p_A^{W^*})$ for the developer's problem under Contract W, as follows:

$$p_A^{W^*} = \frac{1+\delta}{2} + c_A. \tag{6}$$

We only consider price as our decision variable. We derive the second-order condition for developer's problem to see whether the objective function in $(p_A^{W^*})$ is concave or convex. We get the result in the second derivation of the objective function of $(p_A^{W^*})$ is less than 0**.**

$$\frac{\partial^2 \pi_A^W}{\partial {p_A^W}^2} = -\frac{2}{1+\delta} < 0.$$

Therefore, the objective function is concave because $\frac{\partial^2 \pi_A^W}{\partial {p_A^W}^2} < 0$ and is the optimal solution.

Contract W suggests that the developer will facilitate both the initial requirement work of software development and additional RC work under the same cost of c_A. Contract W works under the two-period model with the simultaneous concept, so there is only one price with double costs to charge.

3.2.4 Contract P

We model the developer's problem under Contract P by solving it using the dynamic concept of backward induction. We firstly solve the price two (p_A^{P2}) in the second period by solving the Eq. (7) and get the reference price as $p_A^{P2} = \frac{1+\delta+\gamma c_A - p_A^{P1}}{2\gamma}$. After that, we calculate both the profit one and profit two as $\pi_A^P = \pi_A^{P1} + \lambda\, \pi_A^{P2}$ and we derive the price one (p_A^{P1}) in the first period by solving Eq. (8). Then, we take back the (p_A^{P2}) to $\pi_A^P|_{p_A^{P2}}$ in Eq. (8). We express our model as follows:

$$\underset{p_A^{P2}}{Max}\, \pi_A^{P2} = [(p_A^{P2} - c_A)]D_A^P - F_A, \tag{7}$$

and

$$\underset{p_A^{P1}}{Max}\, \pi_A^P = [(p_A^{P1} - c_A)]D_A^P + \lambda\, \pi_A^{P2}|_{p_A^{P2} = p_A^{P2^*}}. \tag{8}$$

Let (p_A^{P1}) and (p_A^{P2}) as the software prices for Contract P. We model (p_A^{P1}) as a software price for the initial requirement work and (p_A^{P2}) as a software price for the additional RC work. Because the client could propose the additional RC in repeated times in the second period, so the developer needs a specific fix cost for additional RC allocation (F_A) asides the cost (c_A). Both costs for the initial requirement work and the additional RC work are denoted with (c_A) with necessary discounting included in this cost. We denote demand in the first and second period for Contract P as (D_A^P). The demand for both periods is the same due to some assumptions above. We also denote intertemporal value discount as (λ) and price sensitivity for the second period as (γ).

In order to get the optimal prices, we derive the first-order condition for the developer's problem in Contract P using standard backward induction. We examine the first-order condition for the developer's problem in Eq. (9) by firstly executing the (p_A^{P2}) as a reference for the (p_A^{P1}). The first-order condition for the developer's problem in Contract P with respect to (p_A^P) is described as follows:

$$\frac{\partial \pi_A^{P2}}{\partial p_A^{P2}} = (p_A^{P2} - c_A)\left(1 - \frac{p_A^{P1} + \gamma p_A^{P2}}{(1+\delta)}\right) - F_A = 0, \tag{9}$$

and from Eq. (9), we get the reference price for (p_A^{P2})

$$p_A^{P2} = \frac{1 + \delta + \gamma c_A - p_A^{P1}}{2\gamma}. \tag{10}$$

After that, we calculate the profit one and profit two in Eq. (11) as follows:

$$\frac{\partial \pi_A^{P}}{\partial p_A^{P1}} = (p_A^{P1} - c_A)\left(1 - \frac{p_A^{P1} + \gamma p_A^{P2}}{(1+\delta)}\right) + \lambda\, \pi_A^{P2}|_{p_A^{P2} = p_A^{P2^*}} = 0. \tag{11}$$

We get the result for the optimal price $(p_A^{P1^*})$ in Eq. (12) and take back the (p_A^{P2}) for the optimal price $(p_A^{P2^*})$ in Eq. (13) as follows:

$$p_A^{P1^*} = \frac{(1+\delta)(\gamma - \lambda) + \gamma(1 - \gamma + \lambda)c_A}{2\gamma - \lambda}, \tag{12}$$

and

$$p_A^{P2^*} = \frac{1 + \delta + (-1 + 3\gamma - 2\lambda)c_A}{4\gamma - 2\lambda}. \tag{13}$$

We only consider price as our decision variable under this contract. So, we derive the second-order condition for developer's problem to see whether the objective function in $(p_A^{P1^*})$ and $(p_A^{P2^*})$ are concave or convex. We get the result in the second derivation of the objective function in $(p_A^{P1^*})$ and $(p_A^{P2^*})$ are less than 0**.**

$$\frac{\partial^2 \pi_A^P}{\partial {p_A^{P1}}^2} = -\frac{1}{1+\delta} + \frac{\lambda}{2\gamma(1+\delta)} < 0,$$

and

$$\frac{\partial^2 \pi_A^P}{\partial {p_A^{P2}}^2} = -\frac{2\gamma}{1+\delta} < 0,$$

Therefore, the objective functions are concave because $\frac{\partial^2 \pi_A^P}{\partial {p_A^{P1}}^2} < 0$ and $\frac{\partial^2 \pi_A^P}{\partial {p_A^{P2}}^2}$.

Contract P suggests that the developer will facilitate both the initial requirement work and additional RC work under two pricing strategy. Unlike Contract W, this contract has two prices in the model with a dynamic approach. These two works will be executed under the two costs (c_A) that will be charged in each period. This contract also includes a specific fix cost for additional RC allocation (F_A).

3.3 Duopoly Model

In this section, we study the behavior of the client when duopolistic developer selling software products based on three contract designs. We derive the conditions under nine different scenarios. In this duopoly model, we use the monopoly model as the base model to construct the combination scenarios. These nine scenarios will reflect to a competitive market between Developer A and Developer B. We follow (Cheng & Tang, 2010; Nan et al. 2016b) to formulate the duopoly model.

Adopting different software contracts in competitive market are important in the software market. However, previous studies on software contract in (Dharma et al. 2010; Oh et al. 2016) took more attention to monopoly case rather than duopoly case with general case of industry (not specifically mention the software industry). Meanwhile, Dey et al. (2010) examine the effort and time on the optimum profit of software contracts. This study examines the duopoly case in asymmetric models that considers a software market with duopolistic firms. We suppose that Developer A chooses to provide three different contract designs (N, W, and P), and Developer B also provides three different contracts. We describe the duopoly scenario in Figure 3.5 below. The duopoly combination scenarios between Developer A and Developer B:

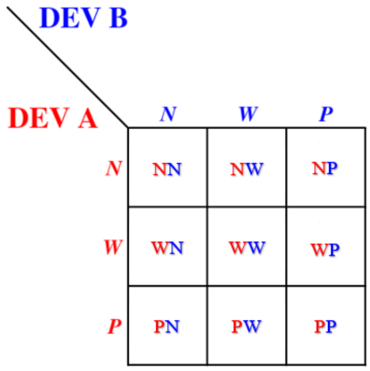


Figure 3.5 Combination Scenarios between Developer A and Developer B

We discuss the nine scenarios into detail explanation in Chapter 3.3.1 to 3.3.4. Because some scenarios have the same model from another scenarios, and the steps to obtain the equilibrium prices are the same. So, we only focus on showing the three scenarios as follows: **NP, WP,** and **PP.** However, we summarize all the indifferent points, the demands, and equilibrium prices for the nine scenarios into the tables.

3.3.1 Utility and Demand Function

Our model is derived from the well-known Hotelling model (Hotelling, 1929). Each client demands at most one unit of software contract design, and all clients are distributed along a unit line according to their preference of the software contracts. Developer A situated at the left end of the market and Developer B located at the right end of the market; that is, Developer A is located at 0 and Developer B at 1. The client's location represents their ideal contract preference. Because the client is only 1, so we assume that the client's location $\hat{x}$ is always in the middle between 0 and 1. If the contract quality does not perfectly match their needs, it incurs a traveling cost, or we call it as misfit cost t, which is increasing in the distance between the location of the client and the contract designs. It is noted that the maximum of this cost is t. As a result, for a client located at $\hat{x}$, the contract designs from Developer A incur a misfit cost $t\hat{x}$ and the contract design from Developer B incurs a misfit cost $t(1-\hat{x})$.

We describe this scenario in Figure 3.6. This duopoly model has two possibilities for each developer, that is a demand A (D_A) for Developer A and demand B (D_B) for Developer B. Both the first and second periods acquired with the same demand. Therefore, the (D_A) and (D_B) are respectively described as $D_A = \hat{x}$ and $D_B = 1 - \hat{x}$.

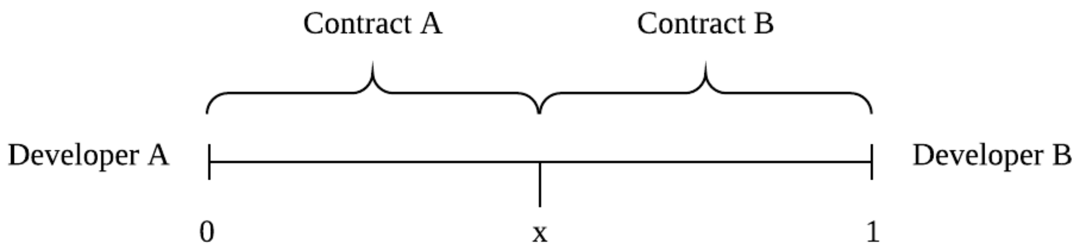


Figure 3.6 Demand of Software Contract in Duopoly

Let $\hat{x}$ represents the marginal consumer type that is indifferent to buy the software from Developer A or Developer B. A client located at the point $(\hat{x})$ obtains the utility of $U_j^i = S - t\hat{x} - p_j^i$, when dealing with the contract design from Developer A and a client located at the point $(1 - \hat{x})$ obtains the utility of $U_j^i = \rho\, S - (1 - \hat{x})\, t - p_j^i$ when dealing with the contract design from Developer B. In line with the monopoly model. We assume that the project is always delivered on time to the client, so there is no variable that will decrease the utility of the software due to the project delay.

We model the utility function with (S) as primary valuation, (t) as the misfit cost due to the contract negotiation. Under Contract N, the utility function modelled with $U_A^N = S - t\hat{x} - p_A^N$ for Developer A and $U_B^N = \rho S - t(1 - \hat{x}) - p_B^N$ for Developer B. The primary valuation (S) must be positive. If the misfit cost tx or $t(1 - \hat{x})$ and price (p_A^N) or (p_B^N) are increased, the primary valuation be decreased. So, the utility gained by the client will reduce too. Developer B needs to put the discount factor (ρ) for his service because to avoid the tendency of the client's intention to always choosing Developer A. This condition applies to all contract designs of service B by Developer B.

Meanwhile under Contract W, the utility function modelled with $U_A^W = (1 + \delta)S - t\hat{x} - p_A^W$ for Developer A and $U_B^W = \rho(1 + \delta)S - (1 - \hat{x})\, t - p_B^W$. Because Contract W accommodates the additional RC, so in its primary valuation (S), we include $(1 + \delta)$ as the second period valuation due to additional RC. Under Contract P, the utility function modelled with $U_A^P = (1 + \delta)S - t\hat{x} - p_A^{P1} - \gamma\, p_A^{P2}$ for Developer A and $U_B^P = \rho(1 + \delta)S - (1 - \hat{x})\, t - p_B^{P1} - \gamma\, p_B^{P2}$ for Developer B. We can see Contract P offers two different pricing in each period. We put price sensitivity (γ) on the second period price for Developer A (p_A^{P2}) and (p_B^{P2}) for

Developer B. Usually, the client already understood with the quality of the software in the first period. So, the lower price sensitivity in (p_A^{P2}) and (p_B^{P2}) will lead to the higher intention to purchase in the second-period contract. The client with better information related to the price and quality of the product will have a higher intention for a product purchase.

Solving the indifferent point ($\hat{x}$) from the utility above to obtain the demand function for all scenarios. Recall that ($\hat{x}$) denote the marginal consumer type. The indifferent market point is obtained by setting the utility function $U_A = U_B$. Because we only show three scenarios to represent the whole nine scenarios, so we will only focus on the NP, WP, and PP scenarios as the representative. These indifferent points are described by the following equations:

Indifferent point of scenario NP when Developer A offers Contract N and Developer B offers the P contact, $U_A^N = U_B^P$. It is described in the equation as follows:

$$S - t\hat{x} - p_A^N = \rho(1+\delta)S - (1-\hat{x})\, t - p_B^{P1} - \gamma\, p_B^{P2}, \tag{14}$$

We derive the indifferent point for scenario NP and get the result as below

$$\hat{x} = \frac{S + t - S\rho - S\delta\rho - p_A^N + p_B^{P1} + \gamma p_B^{P2}}{2t}. \tag{15}$$

We solve the indifferent point of Scenario WP when Developer A offers Contract W, and Developer B offers Contract P, $U_A^W = U_B^P$. It is described in the equation as follows:

$$(1+\delta)S - t\hat{x} - p_A^W = \rho(1+\delta)S - (1-\hat{x})\, t - p_B^{P1} - \gamma\, p_B^{P2}, \tag{16}$$

We derive the indifferent point for Scenario WP and get the result as below

$$\hat{x} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^W + p_B^{P1} + \gamma p_B^{P2}}{2t}. \tag{17}$$

Scenario PP's indifferent point is a condition when Developer A and Developer B offer the same Contract P, $U_A^P = U_B^P$. It is described in the equation as follows:

$$(1+\delta)S - t\hat{x} - p_A^{P1} - \gamma\, p_A^{P2} = \rho(1+\delta)S - (1-\hat{x})\, t - p_B^{P1} - \gamma \tag{18}$$

We derive the indifferent point for Scenario PP and get the result as below

$$\hat{x} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{P1} - \gamma p_A^{P2} + p_B^{P1} + \gamma p_B^{P2}}{2t}. \tag{19}$$

After that, we get the demand of Developer A and Developer B for each scenario, $D_A = \hat{x}$ and $D_B = 1 - \hat{x}$. We only show three scenarios: NP, WP, and PP to represent the whole study. So, the demands are described by the following equations:

The demand for Scenario NP is described in the equation as follows:

$$D_A^{NP} = \frac{S + t - S\rho - S\delta\rho - p_A^N + p_B^{P1} + \gamma p_B^{P2}}{2t}, \tag{20}$$

and demand of Developer B as follows:

$$D_B^{NP} = 1 - \frac{S + t - S\rho - S\delta\rho - p_A^N + p_B^{P1} + \gamma p_B^{P2}}{2t}. \tag{21}$$

The demand for Scenario WP is described in the equation as follows:

$$D_A^{WP} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^W + p_B^{P1} + \gamma p_B^{P2}}{2t}, \tag{22}$$

and demand of Developer B as follows:

$$D_B^{WP} = 1 - \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^W + p_B^{P1} + \gamma p_B^{P2}}{2t}. \tag{23}$$

The demand for Scenario PP is described in the equation as follows:

$$D_A^{PP} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{P1} - \gamma p_A^{P2} + p_B^{P1} + \gamma p_B^{P2}}{2t}, \tag{24}$$

and demand of Developer B as follows:

$$D_B^{PP} = 1 - \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{P1} - \gamma p_A^{P2} + p_B^{P1} + \gamma p_B^{P2}}{2t}. \tag{25}$$

We summarize the indifferent point $\hat{x}$ of all combination scenarios in the following table. All indifferent points $\hat{x}$ for all combination scenarios of Developers A and B are expressed in Table 3.2.

Table 3.2 Indifferent Points for All Combination Scenarios

Scenario NN	Scenario NW
$\hat{x} = \frac{S + t - S\rho - p_A^N + p_B^N}{2t}$	$\hat{x} = \frac{S + t - S\rho - S\delta\rho - p_A^N + p_B^W}{2t}$
Scenario WN	**Scenario WW**
$\hat{x} = \frac{S + t + S\delta - S\rho - p_A^W + p_B^N}{2t}$	$\hat{x} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^W + p_B^W}{2t}$
Scenario NP	**Scenario WP**
$\hat{x} = \frac{S + t - S\rho - S\delta\rho - p_A^N + p_B^{P1} + \gamma p_B^{P2}}{2t}$	$\hat{x} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^W + p_B^{P1} + \gamma p_B^{P2}}{2t}$
Scenario PN	**Scenario PW**
$\hat{x} = \frac{S + t + S\delta - S\rho - p_A^{P1} - \gamma p_A^{P2} + p_B^N}{2t}$	$\hat{x} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{P1} - \gamma p_A^{P2} + p_B^W}{2t}$
Scenario PP	
$\hat{x} = \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{P1} - \gamma p_A^{P2} + p_B^{P1} + \gamma p_B^{P2}}{2t}$	

The demand for Scenario NP, WP, and PP can be expressed as (D_A^{NP}, D_A^{WP}, D_A^{PP}) for Developer A. Meanwhile, the demand for Developer B can be expressed as (D_B^{NP}, D_B^{WP}, D_B^{PP}). All the demands for Developers A and B are displayed in Table 3.3.

Table 3.3 Developers' Software Contract Demand

Scenario NN	Scenario NW
$D_A^{NN} = \frac{S+t-S\rho-p_A^N+p_B^N}{2t}$	$D_A^{NW} = \frac{S+t-S\rho-S\delta\rho-p_A^N+p_B^W}{2t}$
$D_B^{NN} = 1-\frac{S+t-S\rho-p_A^N+p_B^N}{2t}$	$D_B^{NW} = 1-\frac{S+t-S\rho-S\delta\rho-p_A^N+p_B^W}{2t}$
Scenario WN	**Scenario WW**
$D_A^{WN} = \frac{S+t+S\delta-S\rho-p_A^W+p_B^N}{2t}$	$D_A^{WW} = \frac{S+t+S\delta-S\rho-S\delta\rho-p_A^W+p_B^W}{2t}$
$D_B^{WN} = 1-\frac{S+t+S\delta-S\rho-p_A^W+p_B^N}{2t}$	$D_B^{WW} = 1-\frac{S+t+S\delta-S\rho-S\delta\rho-p_A^W+p_B^W}{2t}$
Scenario NP	**Scenario WP**
$D_A^{NP} = \frac{S+t-S\rho-S\delta\rho-p_A^N+p_B^{P1}+\gamma p_B^{P2}}{2t}$	$D_A^{WP} = \frac{S+t+S\delta-S\rho-S\delta\rho-p_A^W+p_B^{P1}+\gamma p_B^{P2}}{2t}$
$D_B^{NP} = 1-\frac{S+t-S\rho-S\delta\rho-p_A^N+p_B^{P1}+\gamma p_B^{P2}}{2t}$	$D_B^{WP} = 1-\frac{S+t+S\delta-S\rho-S\delta\rho-p_A^W+p_B^{P1}+\gamma p_B^{P2}}{2t}$
Scenario PN	**Scenario PW**
$D_A^{PN} = \frac{S+t+S\delta-S\rho-p_A^{P1}-\gamma p_A^{P2}+p_B^N}{2t}$	$D_A^{PW} = \frac{S+t+S\delta-S\rho-S\delta\rho-p_A^{P1}-\gamma p_A^{P2}+p_B^W}{2t}$
$D_B^{PN} = 1-\frac{S+t+S\delta-S\rho-p_A^{P1}-\gamma p_A^{P2}+p_B^N}{2t}$	$D_B^{PW} = 1-\frac{S+t+S\delta-S\rho-S\delta\rho-p_A^{P1}-\gamma p_A^{P2}+p_B^W}{2t}$
Scenario PP	
$D_A^{PP} = \frac{S+t+S\delta-S\rho-S\delta\rho-p_A^{P1}-\gamma p_A^{P2}+p_B^{P1}+\gamma p_B^{P2}}{2t}$	
$D_B^{PP} = 1-\frac{S+t+S\delta-S\rho-S\delta\rho-p_A^{P1}-\gamma p_A^{P2}+p_B^{P1}+\gamma p_B^{P2}}{2t}$	

Solving the same steps as in the monopoly model by solving the developers' problems and obtain the price equilibrium for each developer. We refer to the developers' problems as in the equation (1-13) of the monopoly model, see subchapter (3.2.2-3.2.4). After solving all the developers' problems, we get the equilibrium prices of all combination scenarios. We examine Scenario NP, WP, and PP to represent all the combination scenarios as below.

3.3.2 Scenario NP

We model Developer's A problem in Eq. (26) and Developer's B problem in Eq. (26) and Eq. (27) under scenario NP as follows:

$$\underset{p_A^{NP}}{Max}\ \pi_A^{NP} = [(p_A^{NP} - c_A)]D_A^{NP}, \tag{26}$$

and

$$\underset{p_B^{NP2}}{Max}\ \pi_B^{NP2} = [(p_B^{NP2} - c_B)]D_B^{NP} - F_B, \tag{27}$$

and

$$\underset{p_B^{NP1}}{Max}\ \pi_B^{NP} = [(p_B^{NP1} - c_B)]D_B^{NP} + \lambda\, \pi_B^{NP2}|_{p_B^{NP2} = p_B^{NP2^*}}. \tag{28}$$

We do the same step as in Contracts N, W and P of monopoly model by solving the developers' problems and obtain the equilibrium price for each developer. In this scenario, Developer A offers Contract N and Developer B offers Contract P. In order to get the equilibrium prices, we derive the first-order condition for the developers' problems in Scenario NP with respect to $(p_A^{NP}, p_B^{NP1}, p_B^{NP2})$ as follows:

$$\frac{\pi_A^{NP}}{\partial p_A^{NP}} = \frac{(-c_A + p_A^{NP})(S + t - S\rho - S\delta\rho - p_A^{NP} + p_B^{NP1} + \gamma p_B^{NP2})}{2t} = 0, \tag{29}$$

from the Eq. (29) we get the equilibrium price for Developer A as in Eq. (30)

$$p_A^{NP^*} = \frac{S\gamma + 7t\gamma - 4t\lambda - S\gamma(1+\delta)\rho + (2\gamma - \lambda)c_A + \gamma(1+\gamma)c_B}{3\gamma - \lambda}. \tag{30}$$

After that we derive the first-order condition for the Developer's B problem as follows:

$$\frac{\pi_B^{NP2}}{\partial p_B^{NP2}} = -F_B + (-c_B + p_B^{NP2})\left(1 - \frac{S + t - S\rho - S\delta\rho - p_A^{NP} + p_B^{NP1} + \gamma p_B^{NP2}}{2t}\right) = 0, \tag{31}$$

and

$$\frac{\pi_B^{NP}}{\partial p_B^{NP1}} = (-c_B + p_B^{NP1})\left(1 - \frac{\begin{array}{c} S + t - S\rho - S\delta\rho - p_A^{NP} \\ +p_B^{NP1} + \gamma p_B^{NP2} \end{array}}{2t}\right) + \lambda\, \pi_B^{NP2}|_{p_B^{NP2} = p_B^{NP2^*}} = 0. \tag{32}$$

From Eq. (31) and Eq. (32) we can get the equilibrium prices for Developer B as follows:

$$p_B^{NP1^*} = \frac{(\gamma - \lambda)(5t + S(-1 + \rho + \delta\rho) + c_A) + \gamma(2 - \gamma + \lambda)c_B}{3\gamma - \lambda}, \tag{33}$$

and

$$p_B^{NP2^*} = \frac{5t + S(-1 + \rho + \delta\rho) + c_A + (-1 + 5\gamma - 2\lambda)c_B}{6\gamma - 2\lambda}. \tag{34}$$

Scenario NP suggests that Developer A will facilitate only initial requirement work with single price (p_A^{NP}) and single cost to charge the work (c_A). Developer B will work on two works under the two cost for initial requirement work and additional RC work (c_A).

3.3.3 Scenario WP

We model Developer's A problem in Eq. (35) and Developer's B problem in Eq. (36) and Eq. (37) under Scenario WP as follows:

$$\underset{p_A^{WP}}{Max}\, \pi_A^{WP} = [(p_A^{WP} - 2c_A)]D_A^{WP}, \tag{35}$$

and

$$\underset{p_B^{WP2}}{Max}\, \pi_B^{WP2} = [(p_B^{WP2} - c_B)]D_B^{WP} - F_B, \tag{36}$$

and

$$\underset{p_B^{WP1}}{Max}\, \pi_B^{WP} = [(p_B^{WP1} - c_B)]D_B^{WP} + \lambda\, \pi_B^{WP2}|_{p_B^{WP2} = p_B^{WP2^*}}. \tag{37}$$

We solve Scenario WP with the same steps as in Scenario NP as below:

$$\frac{\pi_A^{WP}}{\partial p_A^{WP}} = \frac{(-2c_A + p_A^{WP})(S + t + S\delta - S\rho - S\delta\rho - p_A^{WP} + p_B^{WP1} + \gamma p_B^{WP2})}{2t} = 0, \tag{38}$$

from the Eq. (38) we get the equilibrium price for Developer A as follows:

$$p_A^{WP^*} = \frac{t(7\gamma - 4\lambda) - S\gamma(1+\delta)(-1+\rho) + (4\gamma - 2\lambda)c_A + \gamma(1+\gamma)c_B}{3\gamma - \lambda}. \tag{39}$$

After that we derive the first-order condition for the Developer's B problem as follows:

$$\frac{\pi_B^{WP2}}{\partial p_B^{WP2}} = -F_B - c_B p_B^{WP2}\left(1 - \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{WP} + p_B^{WP1} + \gamma p_B^{WP2}}{2t}\right) = 0, \tag{40}$$

and

$$\frac{\pi_B^{WP}}{\partial p_B^{WP1}} = (-c_B + p_B^{WP1})\left(1 - \frac{S + t + S\delta - S\rho - S\delta\rho - p_A^{WP} + p_B^{WP1} + \gamma p_B^{WP2}}{2t}\right) + \lambda\, \pi_B^{WP2}\big|_{p_B^{WP2} = p_B^{WP2^*}} = 0. \tag{41}$$

From Eq. (40) and Eq. (41) we can get the equilibrium prices for Developer's B problem as follows:

$$p_B^{WP1^*} = \frac{(\gamma - \lambda)(5t + S(1+\delta)(-1+\rho) + 2c_A) + \gamma(2 - \gamma + \lambda)c_B}{3\gamma - \lambda}, \tag{42}$$

and

$$p_B^{WP2^*} = \frac{5t + S(1+\delta)(-1+\rho) + 2c_A + (-1 + 5\gamma - 2\lambda)c_B}{6\gamma - 2\lambda}. \tag{43}$$

Scenario WP suggests that Developer A will facilitate both initial requirements work and additional RC work with single pricing (p_A^{WP}) by charging a double cost (c_A). Meanwhile, Developer B will work on the two works under the two-pricing strategy (p_B^{WP1}) and (p_B^{WP2}).

3.3.4 Scenario PP

We model the Developer's A problem in Eq. (44) and Eq. (45), and Developer's B problem in Eq. (46) and Eq. (47) under Scenario PP as follows:

$$\underset{p_A^{PP2}}{Max}\, \pi_A^{PP2} = [(p_A^{PP2} - c_A)]D_A^{PP} - F_A, \tag{44}$$

and

$$\underset{p_A^{PP1}}{Max}\, \pi_A^{PP} = [(p_A^{PP1} - c_A)]D_A^{PP} + \lambda\, \pi_A^{PP2}|_{p_A^{PP2} = p_A^{PP2^*}}. \tag{45}$$

And Developer's B problem as follows:

$$\underset{p_B^{PP2}}{Max}\, \pi_B^{PP2} = [(p_B^{PP2} - c_B)]D_B^{PP} - F_B, \tag{46}$$

and

$$\underset{p_B^{PP1}}{Max}\, \pi_B^{PP} = [(p_B^{PP1} - c_B)]D_B^{PP} + \lambda\, \pi_B^{PP2}|_{p_B^{PP2} = p_B^{PP2^*}}. \tag{47}$$

In this scenario, Developer A and Developer B offer the same Contract P. We solve Scenario PP with the same steps as in Scenario NP and Scenario WP. We derive the first-order condition for the developers' problems in Scenario PP with respect to $(p_A^{PP1}, p_A^{PP2}, p_B^{PP1}, p_B^{PP2})$ as follows:

$$\frac{\pi_A^{PP2}}{\partial p_A^{PP2}} = -F_A + \frac{\begin{array}{c}(-c_A + p_A^{PP2})(S + t + S\delta - S\rho - S\delta\rho \\ -p_A^{PP1} - \gamma p_A^{PP2} + p_B^{PP1} + \gamma p_B^{PP2})\end{array}}{2t} = 0, \tag{48}$$

and

$$\begin{aligned}\frac{\pi_A^{PP}}{\partial p_A^{PP1}} &= \frac{(-c_{A1} + p_A^{PP1})\begin{pmatrix} S + t + S\delta - S\rho - S\delta\rho - p_A^{PP1} \\ -\gamma p_A^{PP2} + p_B^{PP1} + \gamma p_B^{PP2}\end{pmatrix}}{2t} \\ &\quad + \lambda\, \pi_A^{PP2}|_{p_A^{PP2} = p_A^{PP2^*}} = 0.\end{aligned} \tag{49}$$

From the Eq. (48) and Eq. (49) we get the equilibrium prices for Developer's A problem as follows:

$$p_A^{PP1^*} = \frac{\begin{array}{c}\gamma(-3(-2+\gamma)\gamma + 2(-1+\gamma)\lambda)c_A + (3\gamma - 2\lambda) \\ (t(9\gamma - 4\lambda) - S\gamma(1+\delta)(-1+\rho) + \gamma(1+\gamma)c_B)\end{array}}{\gamma(9\gamma - 4\lambda)}, \tag{50}$$

and

$$p_A^{PP2^*} = \frac{\begin{array}{c}t(9\gamma - 4\lambda) - S\gamma(1+\delta)(-1+\rho) \\ +\gamma(-1+8\gamma-4\lambda)c_A + \gamma(1+\gamma)c_B\end{array}}{\gamma(9\gamma - 4\lambda)}. \tag{51}$$

We solve the Developer's B problem in Eq. (52) and Eq. (53) as follows:

$$\frac{\pi_B^{PP2}}{\partial p_B^{PP2}} = -F_B + (-c_B + p_B^{PP2})(1 - \frac{\begin{array}{c}S + t + S\delta \\ -S\rho - S\delta\rho - \\ p_A^{PP1} - \gamma p_A^{PP2} \\ +p_B^{PP1} + \gamma p_B^{PP2}\end{array}}{2t}) = 0, \tag{52}$$

and

$$\begin{aligned}\frac{\pi_B^{PP}}{\partial p_B^{PP1}} &= (-c_B + p_B^{PP1})(1 - \frac{\begin{array}{c}S + t + S\delta - S\rho - S\delta\rho - \\ p_A^{PP1} - \gamma p_A^{PP2} + p_B^{PP1} + \gamma p_B^{PP2}\end{array}}{2t}) \\ &\quad +\lambda\, \pi_B^{PP2}|_{p_B^{PP2} = p_B^{PP2^*}} = 0.\end{aligned} \tag{53}$$

From Eq. (52) and Eq. (53) we can get the equilibrium prices for Developer's B problem as follows:

$$p_B^{PP1^*} = \frac{\begin{array}{c}(3\gamma - 2\lambda)(t(9\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) + \\ \gamma(1+\gamma)c_A) + \gamma(-3(-2+\gamma)\gamma + 2(-1+\gamma)\lambda)c_B\end{array}}{\gamma(9\gamma - 4\lambda)}, \tag{54}$$

and

$$p_B^{PP2^*} = \frac{\begin{array}{c}t(9\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) + \gamma(1+\gamma)c_A + \\ \gamma(-1+8\gamma-4\lambda)c_B\end{array}}{\gamma(9\gamma - 4\lambda)}. \tag{55}$$

Scenario PP suggests that Developer A and Developer B will facilitate both initial requirements work and additional RC work with two pricing strategy by charging a cost (c_A, c_B) in each period plus a specific fix cost for additional RC allocation (F_A, F_B).

We provide all the developers' equilibrium prices of developers' problems for the nine scenarios in Table 3.4. We examine the nine scenarios to obtain the equilibrium prices by adopting the exact steps from our monopoly model. We present the equilibrium prices of all combination scenarios as follows:

Table 3.4 Developers' Equilibrium Prices

Scenario NN	**Scenario NW**
$p_A^{NN^*} = \frac{1}{3}(S + 3t - S\rho + 2c_{A1} + c_{B1})$	$p_A^{NW^*} = \frac{1}{3}(S + 3t - S(1+\delta)\rho + 2c_{A1} + c_{B1} + c_{B2})$
$p_B^{NN^*} = \frac{1}{3}(3t + S(-1+\rho) + c_{A1} + 2c_{B1})$	$p_B^{NW^*} = \frac{1}{3}(3t + S(-1+\rho+\delta\rho) + c_{A1} + 2c_{B1} + 2c_{B2})$
Scenario WN	**Scenario WW**
$p_A^{WN^*} = \frac{1}{3}(S + 3t + S\delta - S\rho + 2c_{A1} + 2c_{A2} + c_{B1})$	$p_A^{WW^*} = \frac{1}{3}(3t - S(1+\delta)(-1+\rho) + 2c_{A1} + 2c_{A2} + c_{B1} + c_{B2})$
$p_B^{WN^*} = \frac{1}{3}(3t + S(-1-\delta+\rho) + c_{A1} + c_{A2} + 2c_{B1}$	$p_B^{WW^*} = \frac{1}{3}(3t + S(1+\delta)(-1+\rho) + c_{A1} + c_{A2} + 2c_{B1} + 2c_{B2})$

Scenario PN
$p_A^{PN1^*} = \frac{2\gamma c_{A1} + (\gamma-\lambda)(S + 5t + S\delta - S\rho - \gamma c_{A2} + c_{B1})}{3\gamma - \lambda}$
$p_A^{PN2^*} = \frac{S + 5t + S\delta - S\rho - c_{A1} + (5\gamma - 2\lambda)c_{A2} + c_{B1}}{6\gamma - 2\lambda}$
$p_B^{PN^*} = \frac{t(7\gamma - 4\lambda) + S\gamma(-1-\delta+\rho) + \gamma(c_{A1} + \gamma c_{A2}) + (2\gamma-\lambda)c_{B1}}{3\gamma - \lambda}$
Scenario PW
$p_A^{PW1^*} = \frac{2\gamma c_{A1} - (\gamma-\lambda)(-5t + S(1+\delta)(-1+\rho) + \gamma c_{A2} - c_{B1} - c_{B2})}{3\gamma - \lambda}$
$p_A^{PW2^*} = \frac{t(7\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) + \gamma c_{A1} + \gamma^2 c_{A2} + (2\gamma-\lambda)(c_{B1} + c_{B2})}{3\gamma - \lambda}$
$p_B^{PN^*} = \frac{(5t - S(1+\delta)(-1+\rho) - c_{A1} + (5\gamma - 2\lambda)c_{A2} + c_{B1} + c_{B2})}{6\gamma - 2\lambda}$
Scenario NP
$p_A^{NP^*} = \frac{S\gamma + 7t\gamma - 4t\lambda - S\gamma(1+\delta)\rho + (2\gamma-\lambda)c_{A1} + \gamma(c_{B1} + \gamma c_{B2})}{3\gamma - \lambda}$ $p_B^{NP1^*} = \frac{(\gamma-\lambda)c_{A1} + 2\gamma c_{B1} + (\gamma-\lambda)(5t + S(-1+\rho+\delta\rho) - \gamma c_{B2})}{3\gamma - \lambda}$ $p_B^{NP2^*} = \frac{5t + S(-1+\rho+\delta\rho) + c_{A1} - c_{B1} + (5\gamma - 2\lambda)c_{B2}}{6\gamma - 2\lambda}$
Scenario WP
$p_A^{WP^*} = \frac{1}{3\gamma - \lambda}(t(7\gamma - 4\lambda) - S\gamma(1+\delta)(-1+\rho) + (2\gamma-\lambda)c_{A1} + (2\gamma-\lambda)c_{A2} + \gamma(c_{B1} + \gamma c_{B2}))$ $p_B^{WP1^*} = \frac{1}{3\gamma - \lambda}((\gamma-\lambda)(5t + S(1+\delta)(-1+\rho)) + (\gamma-\lambda)c_{A1} + (\gamma-\lambda)c_{A2} + 2\gamma c_{B1} + \gamma(-\gamma + \lambda)c_{B2})$ $p_B^{WP2^*} = \frac{5t + S(1+\delta)(-1+\rho) + c_{A1} + c_{A2} - c_{B1} + (5\gamma - 2\lambda)c_{B2}}{6\gamma - 2\lambda}$

Scenario PP
$p_A^{PP1^*} = \frac{2\gamma(3\gamma-\lambda)c_{A1} - (3\gamma-2\lambda)(-9t\gamma+4t\lambda+S\gamma(1+\delta)(-1+\rho)+\gamma^2 c_{A2} - \gamma(c_{B1}+\gamma c_{B2}))}{\gamma(9\gamma-4\lambda)}$
$p_A^{PP2^*} = \frac{t(9\gamma-4\lambda) - S\gamma(1+\delta)(-1+\rho) + \gamma(-c_{A1} + (8\gamma-4\lambda)c_{A2} + c_{B1} + \gamma c_{B2})}{\gamma(9\gamma-4\lambda)}$
$p_B^{PP1^*} = \frac{(3\gamma-2\lambda)(t(9\gamma-4\lambda)+S\gamma(1+\delta)(-1+\rho)) + \gamma(3\gamma-2\lambda)c_{A1} + \gamma(\gamma(3\gamma-2\lambda)c_{A2} + (6\gamma-2\lambda)c_{B1} + \gamma(-3\gamma+2\lambda)c_{B2})}{\gamma(9\gamma-4\lambda)}$
$p_B^{PP2^*} = \frac{t(9\gamma-4\lambda) + S\gamma(1+\delta)(-1+\rho) + \gamma(c_{A1} + \gamma c_{A2} - c_{B1} + 8\gamma c_{B2} - 4\lambda c_{B2})}{\gamma(9\gamma-4\lambda)}$

4. ANALYTICAL ANALYSES

In this chapter, we will analyse the analytical result from our model based on the optimal equilibrium prices from the two software developers under the nine scenarios in Chapter 3. The trend analysis of the various influential parameters on the decision variable is carried out. The research findings are presented using the analytical approach. We discuss the analysis of monopoly and duopoly models in this chapter. We only take three scenarios for our duopoly model to represent the whole findings. However, we also summarize the full results in Table 4.1.

4.1 Analyses under Monopoly Model

We analyse the prices and profits under different contracts in our monopoly model. We have two different contracts to explore: Contract W and Contract P. The proofs of all calculations are described in the appendix.

Under Contract W, Developer A as monopolist will offer Contract W to the client. We aim at the influential parameter: the primary valuation (θ) and the second-period valuation due to additional RC (δ). We explain the influential parameters of Developer A under **Proposition 1** as follows:

Proposition 1. *Contract W price $p_A^{W^*}$ and profit $\pi_A^{W^*}$ analyses with respect to δ, θ*

$$\frac{\partial p_A^{W^*}}{\partial \theta} = 0, \frac{\partial \pi_A^{W^*}}{\partial \theta} = 0, \frac{\partial p_A^{W^*}}{\partial \delta} > 0, \frac{\partial \pi_A^{W^*}}{\partial \delta} > 0.$$

We analyze the changing trend of the parameters in Proposition 1. We see that there is no significant impact on the higher or lower degree of primary valuation (θ). Meanwhile, it can be found that under Contract W condition, the higher the second-period valuation due to additional RC (δ), will lead to the higher price and will increase the profit. This is indicating if the higher degree of additional RC to the developer, will give impact to the higher price because the client must pay the additional work. This condition will influence to increase the profit.

Under Contract P, Developer A as monopolist will offer Contract P to the client. We aim at the influential parameters: the primary valuation (θ), the second-period valuation due to additional RC (δ), price sensitivity (γ) for the second-period price, a specific fix cost for additional RC allocation (F_A) and intertemporal value discount (λ). We explain each influential parameter of Developer A under **Proposition 2** as follows:

Proposition 2. *Contract P price $p_A^{P1^*}, p_A^{P2^*}$ and profit $\pi_A^{P^*}$ analyses with respect to $\theta, \delta, \gamma, F_A, \lambda$*

$$\frac{\partial p_A^{P1^*}}{\partial \theta} = 0, \frac{\partial p_A^{P1^*}}{\partial \delta} > 0, \frac{\partial p_A^{P1^*}}{\partial \gamma} > 0, \frac{\partial p_A^{P1^*}}{\partial F_A} = 0, \frac{\partial p_A^{P1^*}}{\partial \lambda} < 0,$$

$$\frac{\partial p_A^{P2^*}}{\partial \theta} = 0, \frac{\partial p_A^{P2^*}}{\partial \delta} > 0, \frac{\partial p_A^{P2^*}}{\partial \gamma} < 0, \frac{\partial p_A^{P2^*}}{\partial F_A} = 0, \frac{\partial p_A^{P2^*}}{\partial \lambda} > 0,$$

$$\frac{\partial \pi_A^{P^*}}{\partial \theta} = 0, \frac{\partial \pi_A^{P^*}}{\partial \delta} < 0, \frac{\partial \pi_A^{P^*}}{\partial \gamma} < 0, \frac{\partial \pi_A^{P^*}}{\partial F_A} < 0, \frac{\partial \pi_A^{P^*}}{\partial \lambda} > 0.$$

We can see the changing trend of each parameter of Proposition 2. It can be found that in Contract P condition, a primary valuation (θ) and a specific fix cost for additional RC allocation (F_A) are equal to 0 towards the first and second prices. It means that the higher or the lower level of (θ) and (F_A) have no significant influence or impact on both prices. Meanwhile, the higher (F_A) will decrease the profit. Otherwise, the higher (δ) will increase both prices but will decrease the profit. Indicating that when the client requests the high number of additional RC to the developer, the developer will increase the price to charge the additional work.

The higher the price sensitivity (γ) in Developer A will increase the first period price. Otherwise, it will decrease the second-period price and profit. Indicating that the client's price sensitivity is higher in the first period because they do not know more about information related to the quality and price of the software. Meanwhile, the clients know better information in the second period because they already gained some information related to price and quality. So, they will decrease their price sensitivity. According to the analytical result, the higher the (λ) will decrease first-period price. Meanwhile, it will increase the second-period price. This condition will lead Developer A to increase the profit.

Overall, the conditions above are triggered by the consumer behavior of the client. Generally, when a software product has a higher valuation from the additional RC, the total valuation of the product will increase because client request related to the additional RC can be accommodated. At the end, it will reflect to the quality of the software because the features completeness.

4.2 Analyses under Duopoly Model

We analyse nine different combination scenarios and take three combination scenarios to represent the whole analysis, which are Scenarios NP, WP, and PP. The proofs of all calculations are described in the appendix. We highlight the column in Table 4.1-4.6 with grey color to emphasize only two parameters: the primary valuation (S), and the second-period valuation due to additional RC (δ). We use (+) to represent the positive effect, $(-)$ to represent negative effect and 0 to represent neutral effect which indicate no significant impact of parameters towards price and profit.

Under Scenario NP, Developer A chooses Contract N and Developer B chooses Contract P. We aim at the influential parameters: the primary valuation (S), the second-period valuation due to additional RC (δ), price sensitivity (γ) for the second-period price, discount factor for service B (ρ) and intertemporal value discount (λ). We will compare the competition between Developer A and Developer B under **Proposition 3** as follows:

Proposition 3. *Scenario NP prices* p_A^{NP}, p_B^{NP} *and profits* $\pi_A^{NP^*}, \pi_B^{NP^*}$ *analyses with respect to* $S, \delta, \gamma, \rho, \lambda$

Table 4.1 Influential Parameters Analyses under Scenario NP on Prices

$\boldsymbol{p^*}$	**Influential Parameters under Scenario NP on Prices**				
	S	δ	γ	ρ	λ
$\boldsymbol{p_A^{NP^*}}$	+	−	−	−	−
$\boldsymbol{p_B^{NP1^*}}$	−	+	+	+	−
$\boldsymbol{p_B^{NP2^*}}$	−	+	−	−	−

Table 4.2 Influential Parameters Analyses under Scenario NP on Profits

$\boldsymbol{\pi^*}$	**Influential Parameters under Scenario NP on Profits**				
	S	δ	γ	ρ	λ
$\boldsymbol{\pi_A^{NP^*}}$	+	−	+	−	−
$\boldsymbol{\pi_B^{NP^*}}$	−	−	−	+	−

We can see the trend of each parameter of Proposition 3 in Table 4.1 and 4.2. It can be found that in this scenario condition, the trend of each parameter will influence the prices and profits of Developer A and Developer B. Especially the influence of primary valuation (S) and second period valuation due to additional RC (δ). We carry out the analysis of Scenario NP as below:

1. Under Scenario NP, **the higher the degree of primary valuation ($\boldsymbol{S}$), will increase price. It leads to increase the profit for Developer A.**
2. **The higher the second valuation due to additional RC ($\boldsymbol{\delta}$), will decrease price and profit for Developer A.** This is because Contract N does not accommodate additional RC.
3. **Meanwhile, the higher the degree of ($\boldsymbol{S}$) will decrease price and profit. Otherwise, the higher the degree of ($\boldsymbol{\delta}$) will increase prices but will decrease profit for Developer B.** This is because Developer B chooses Contract P which accommodates the additional RC.

Under Scenario WP, Developer A chooses Contract W and Developer B chooses Contract P. We will compare the competition between Developer A and Developer B under **Proposition 4** as follows:

Proposition 4. *Scenario WP prices* p_A^{WP}, p_B^{WP} *and profits* $\pi_A^{WP^*}, \pi_A^{WP^*}$ *analyses with respect to* $S, \delta, \gamma, \rho, \lambda$

Table 4.3 Influential Parameters Analyses under Scenario WP on Prices

$\boldsymbol{p^*}$	**Influential Parameters under Scenario WP on Prices**				
	S	δ	γ	ρ	λ
$\boldsymbol{p_A^{WP^*}}$	+	−	+	−	−
$\boldsymbol{p_B^{WP1^*}}$	−	+	+	+	−
$\boldsymbol{p_B^{WP2^*}}$	−	+	−	+	+

Table 4.4 Influential Parameters Analyses under Scenario WP on Profits

π^*	Influential Parameters under Scenario WP on Profits				
	S	δ	γ	ρ	λ
$\pi_A^{WP^*}$	+	+	+	–	–
$\pi_B^{WP^*}$	–	+	–	+	–

We can see the trend of each parameter of Proposition 4 in Table 4.3 and 4.4. It can be found that in this scenario condition, the trend of each parameter will influence the prices and profits of Developer A and Developer B. Especially the influence of primary valuation (S) and second period valuation due to additional RC (δ). We carry out the analysis of Scenario WP as below:

1. Under Scenario WP condition, **the higher the degree of primary valuation (*S*) will increase the price and profit.**
2. **The higher the second valuation due to RC (*δ*) will decrease the price but will increase the profit for Developer A.** This is because Developer A only offers a single pricing for two works with double costs to charge.
3. Meanwhile, Developer B offers Contract P under two pricing strategy to accommodate the additional RC.
4. On the other hand, **the higher the degree of (*S*) will decrease prices and profit of Developer B.** Meanwhile, **the higher the degree of (*δ*) will increase prices and profit.** We believe this is because of the discount factor (ρ) offered by Developer B.

Under Scenario PP, both Developer A and Developer B choose to offer Contract P. We will compare the competition between Developer A and Developer B under **Proposition 5** as follows:

Proposition 5. *Scenario PP prices* p_A^{PP}, p_B^{PP} *and profits* $\pi_A^{PP^*}, \pi_A^{PP^*}$ *analyses with respect to* $S, \delta, \gamma, \rho, \lambda$

Table 4.5 Influential Parameters Analyses under Scenario PP on Prices

p^*	Influential Parameters under Scenario PP on Prices				
	S	δ	γ	ρ	λ
$p_A^{PP1^*}$	+	–	+	–	–
$p_A^{PP2^*}$	+	+	–	–	+
$p_B^{PP1^*}$	–	+	+	+	–
$p_B^{PP2^*}$	–	+	–	+	–

Table 4.6 Influential Parameters Analyses under Scenario PP on Profits

π^*	Influential Parameters under Scenario PP on Profits				
	S	δ	γ	ρ	λ
$\pi_A^{PP^*}$	+	+	+	–	–
$\pi_B^{PP^*}$	–	+	+	+	–

We can see the trend of each parameter of Proposition 5 in Table 4.5 and 4.6. It can be found that in this scenario condition, the trend of each parameter will influence the prices and profits of Developer A and Developer B. Especially the influence of primary valuation (S) and second period valuation due to additional RC (δ). We carry out the analysis of Scenario PP as below:

1. Under Scenario PP condition, **the higher the degree of primary valuation (S) will increase prices and profit for Developer A. Meanwhile, the higher the degree of second period valuation due to additional RC (δ) will decrease the first period price but will increase the second period price and profit.**
2. This is because Contract P offers two pricing strategy to deal with the specific work of additional RC in the second period.
3. Meanwhile, **the higher the degree of (S) will decrease prices and profit of Developer B.** Meanwhile, **the higher the degree of (δ) will decrease prices and profit.** We believe this is because Developer B offers discount factor to attract the client.

Generally, the total valuation of software product is influenced by the increase of second-period valuation level. Because **when the additional valuation is added, the primary valuation will absorb the addition from the second valuation, so the total valuation will be higher.** Meanwhile throughout this analysis, Developer B uses a discount for its service to convert the market. As the impact, **they will perceive lower price and higher profit from the increasing of those two parameters.**

4.3 Analyses Comparison between Monopoly and Duopoly Model

We compare the behavior of the market when it goes under monopoly or duopoly market. We examine second-period valuation due to additional RC (δ), primary valuation in monopoly θ and duopoly S as the influential parameters under different conditions of the market. In a monopoly market, Developer offers three different contracts: N, W, and P. Otherwise, in a duopoly market, there is a condition when Developer A offers three different contracts: N, W, and P, while Developer B always consistent by offering Contract P in all conditions. We describe the comparison between monopoly and duopoly model in Table 4.8 below.

Table 4.8 Comparison Analyses between Monopoly and Duopoly Model

Case	Parameters	Decision Variables	Developer A		
			N	W	P
Monopoly	δ	p_1		+	+
		π		+	−
Duopoly	S	p_1	+	+	+
		π	+	+	+
	δ	p_1	−	−	−
		π	−	+	+

We see the different behavior of the market under monopoly and duopoly model. From the three contracts, we analyze how the developer reacts under the presence of RC. We summarize the information in Table 4.8 into some points as follows:

1. Under the monopoly market, when Developer A chooses to offer Contract W, **the price and profit will increase due to the increase of ($\boldsymbol{\delta}$).** Meanwhile, under the duopoly market **when ($\boldsymbol{\delta}$) increases, the price will decrease but the profit will increase.**
2. Under Contract P, both developers' trend of primary valuation (S) and second period valuation due to RC (δ) have different pattern on price and profit. Under monopoly model, **Developer A will increase their first period price, as the impact the profit will decrease.** Meanwhile, under duopoly model, **Developer A will decrease their first period price, as the impact the profit will increase.**

4.4 Comparative Analysis among Scenarios

We explore a comparative analysis of different scenarios. We analyze the influence of second-period valuation due to additional RC (δ) towards the scenario change of developers. We investigate the performance of Scenario WP and PP for Developer A under the impact of (δ).

Solving the profit difference between Scenario WP and PP is an indicator to measure the profitability of both scenarios for Developer A. This study will also analyze the impact of (δ) on profit. The profit difference between the two scenarios of Developer A is shown as follows:

$$\Delta\pi_A^i = \pi_A^{WP} - \pi_A^{PP}$$

Since the profit equilibrium of Scenario WP and PP are more complicated, we eliminate the effect of some parameters. We degenerate cost $C_j = 0$, for Developer A or B *{j=A, B}* by setting them at lower bounds. We also simplify $\gamma = \lambda$, which means that the developers and clients will perceive the same discount values of the second period compared to the first period. After simplification, the degenerated parameters are disappeared. We will analyze the threshold of (δ), which will potentially affect the profitability of both scenarios. We solve the difference between Scenario WP and PP, and the threshold of (δ) can be obtained. This value represents the lower limit of the difference between Scenario WP and PP. Therefore, if the difference between the two scenarios does not reach their threshold, the profit of Scenario WP will be lower than that in the unprofitable condition. Indicating that Scenario WP cannot be increased, so the developer will not choose this scenario. The threshold value of the difference between the two scenarios is expressed, and the subscripts WP and PP represent Scenario WP and Scenario PP. The analysis result is stated in Proposition 6:

Proposition 6. *The threshold of the second-period valuation due to additional RC* (δ)

$$\delta > \delta_A^{PP} \equiv \frac{1}{9}(-9 + \frac{5t}{S - S\rho} - \frac{20\sqrt{S^2 t(-1+\rho)^2(4t - 9\lambda F_A)}}{S^2(-1+\rho)^2} \Leftrightarrow \pi_A^{WP} > \pi_A^{PP}.$$

In proposition 6, for Developer A, whether the second-period valuation (δ) is higher or lower than the level showed as result above, Developer A will always be profitable to choose Scenario PP instead of Scenario WP. Because we already know Developer A preference, we examine the Developer's B scenario. By using the same step with Proposition 6. We examine the profit difference between the three scenarios of Developer B, which are Scenarios PN, PW, and PP. We firstly check the profit difference between Scenario PN and Scenario PW (π_B^{PN}) and (π_B^{PW}) and obtain the threshold of the second-period valuation due to additional RC (δ) in Proposition 7. (1):

Proposition 7. (1) *The difference between π_B^{PN} and π_B^{PW}*

$$\Delta\pi_B^i = \pi_B^{PN} - \pi_B^{PW},$$

We use the result of $\pi_B^{PN} - \pi_B^{PW}$ to confirm. We get the result as:

$$\Delta\pi_B^i = -\frac{S\delta\rho(6t - 2S(1+\delta) + S(2+\delta)\rho)}{16t} \Leftrightarrow \pi_B^{PN} < \pi_B^{PW}.$$

We can see the result is always negative for Scenario PN. So, from the observation above, Scenario PN is less than scenario PW. After knowing the difference between Scenarios PN and PW. We then examine Scenarios PW and PP. We conduct the same steps as in Proposition 6. We will analyze the difference between the two scenarios with the subscripts PW and PP to represent Scenario PW and Scenario PP and take the second-period valuation due to additional RC (δ) between π_B^{PN} and π_B^{PW} as the threshold. The threshold between the two scenarios of Developer B is stated in Proposition 7. (2):

Proposition 7. (2) *The threshold of the second-period valuation due to additional RC (δ) between π_B^{PW} and π_B^{PP}*

$$\delta_B^{PP} \equiv -1 + \frac{5(St(-1+\rho) - 4\sqrt{S^2 t(-1+\rho)^2(4t - 9\lambda F_B)})}{9S^2(-1+\rho)^2} \Leftrightarrow \pi_B^{PW} < \pi_B^{PP}.$$

In proposition 7. (2), as we see, Scenario PW does not perform better compared to Scenario PP in term of (δ). So, the profit of Scenario PW is less than Scenario PP, $\pi_B^{PW} < \pi_B^{PP}$ for Developer B when the level of (δ) is low or high. We can take a decision whether the level of level of (δ) is low or high, Developer B always chooses Scenario PP among the other two Scenarios PW and PN.

5. NUMERICAL ANALYSIS

In this chapter, we will use numerical analysis to verify the formulas of Developer A and Developer B profit maximization and to analyze the impact of parameters on profits. In this numerical valuation, we only focus on exploring the duopoly market. The basic setting parameters must be satisfied with following conditions:

1. The price p_1 for all contracts must be greater or equal to 0
2. The price p_2 for all contracts must be greater or equal to 0
3. The demands for all contracts must be greater or equal to 0

We follow the prior literature from Dey et al. (2010) to illustrate the result. By using this condition, we choose the basic parameters as follows: First, we define the costs (c_A) and (c_B) under the same value of 0.1. We assign all the costs values are the same for the first and the second period of both developers. We also define (S) into 1, (γ) into 1, (t) into 0.5, (F_A) and (F_B) into 0.2. The parameters $(\mathrm{c_A}, \mathrm{c_B}, S, \gamma, t, F_A, F_B)$ will be set up with the same value and will never change due to some assumptions. Meanwhile, we explore $(\delta,\ \rho,$ and $\lambda)$ into different value settings (low, moderate, and high). Using three different cases, we generate the range of each parameter specifically for the influential parameters $(\delta,\ \rho,$ and $\lambda)$. The range for (δ) is between 0.1 and 0.6, the range for (ρ) is between 0.3 and 0.6, and the range for (λ) is between 0.155 and 0.55. We describe the case parameters in Table 5.1.

Table 5.1 The Basic Case Parameters

Case	**Parameters**		
	δ	ρ	λ
Range	[0.1,0.6]	[0.3,0.6]	[0.155,0.555]
Basic	0.3	0.425	0.355

5.1 Profit Comparison of Single Parameters

In this chapter, we discuss the impact of changing single parameters (δ, ρ, λ) on profits under nine different scenarios. The results of this analysis are strategic decisions in the form of payoff matrixes and graphs to represent the influence of single parameters on profits. We examine one by one parameter by decreasing or increasing its level. We examine each parameter change under different scenarios and draw the payoff matrixes of normal form game. We then analyse the results using Nash Equilibria to obtain the best response for each developer. There are nine scenarios with possibilities to come out as the best response for Developer A and Developer B.

We generate the equilibrium prices of each scenario and apply three cases on these parameters (δ, ρ, λ). Here are the three cases to propose to examine the scenarios under the numerical analysis. We discuss them one by one as below:

1. **Case 1:** Profit in terms of (δ) with the value range [0.1,0.6]

2. **Case 2:** Profit in terms of (ρ) with the value range [0.3,0.6]
3. **Case 3:** Profit in terms of (λ) with the value range [0.155,0.555]

(1) We examine Case 1 as follows: we set (δ) into the range of [0.1,0.6], (ρ) into 0.425, and (λ) into 0.355. We got the result as illustrated in Figure 5.2 (A) to 5.2 (C) for the payoff matrixes. We can see the comparison under three different payoff matrixes that are showing payoff matrix A with (δ) under the set-up value 0.1, pay off matrix B with (δ) under the set-up value 0.3, and payoff matrix C with (δ) under set-up value 0.6. We set payoff matrix A as the low value, payoff matrix B as the moderate value and payoff matrix C as the high value.

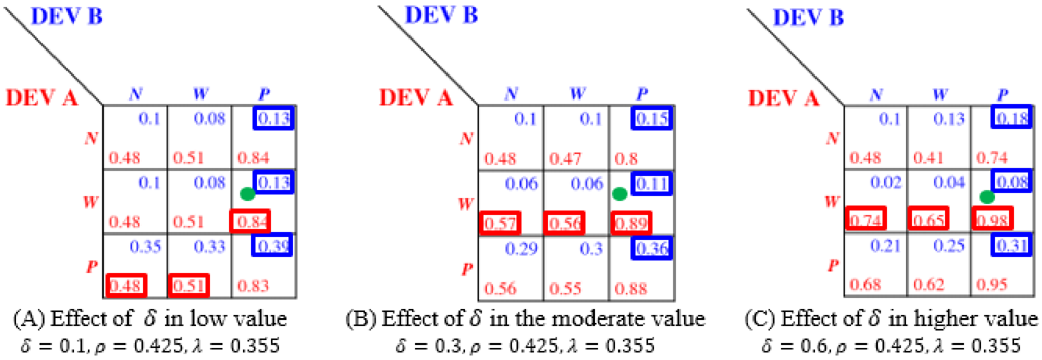
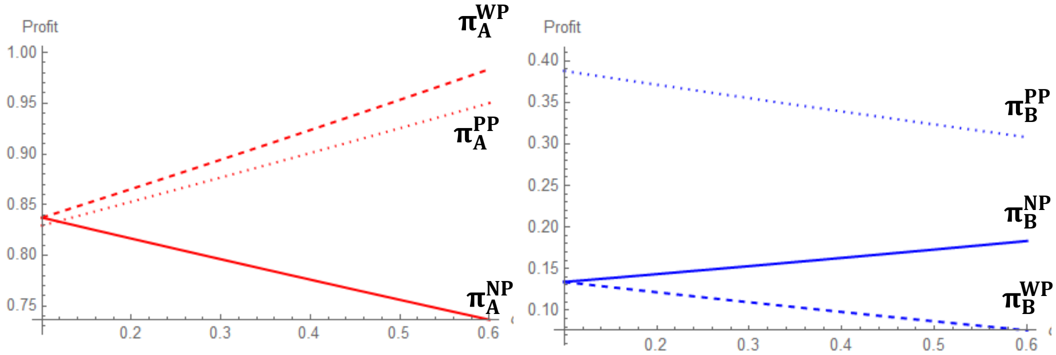


Figure 5.2 Payoff Matrixes of Nine Scenarios with (δ) under Value Range [0.1,0.6]

As we can see on the payoff matrixes A, B, and C, when the value of (δ) is set up higher, the trend of profit will increase for Developer A. However, the profit will be slightly decreased for Developer B. The interesting finding is when the (δ) is higher, Developer A will potentially change the strategy to either Contract W or P. In another hand, Developer B seems to confident by always offering Scenario P in all conditions. We can see in the matrixes when the (δ) is set up higher, the profit will increase, and Scenario WP comes out as the best response among all scenarios because the Nash Equilibrium exist in it. Contract N never be chosen as an alternative strategy for both Developer A and Developer B. This finding is supported by the graphs illustrated in Figure 5.3 (A) and 5.3 (B). Based on the payoff matrixes results, the potential change of strategy happened among Scenario NP, WP, and PP. So, we investigate the trend among them through the graphs below.



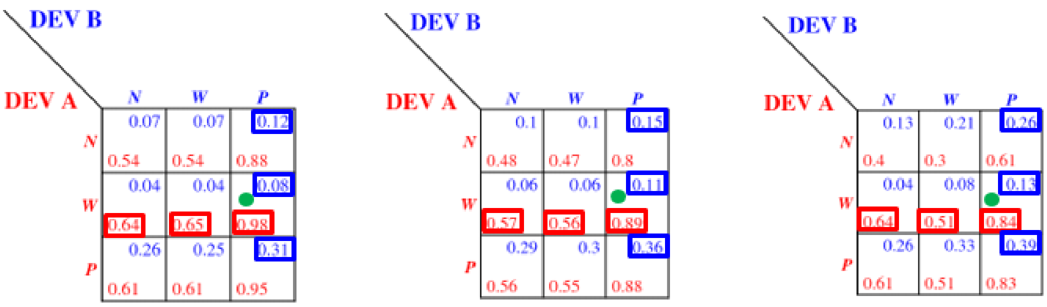
(A) Effect of δ in Developer A
$\delta = [0.1,0.6], \rho = 0.425, \lambda = 0.355$

(B) Effect of δ in Developer B
$\delta = [0.1,0.6], \rho = 0.425, \lambda = 0.355$

Figure 5.3 The Effect of (δ) in Developer A and B (Scenario NP, WP, and PP)

We can analyze from Figure 5.3 (A), when the (δ) is set up low, Developer A will get lower profit. On the other hand, when the (δ) is set up high, Developer A will get a better profit. We also can see, the higher (δ) will potentially change the strategy of Developer A. Here, Scenario WP comes out as the best response for Developer A to deal with second-period valuation due to the additional RC (δ). Meanwhile, in Developer B, see the Figure 5.3 (B), Contract P comes out as the best response among the other two contracts. The higher (δ) will tend to decrease the profit for Developer B. Meanwhile Contract N acts as the less profitable contract to deal with additional RC (δ) for both developers. In any scenarios of NN, NW, and NP play no significant role to change the strategy. This is because Contract N does not promote the additional RC. Developer will only work for the initial requirement and will not accommodate the additional work related to RC.

(2) We examine Case 2 as follows: we set (δ) into 0.3, (ρ) into the range [0.3,0.6], and (λ) into 0.355. We got the result as illustrated in Figure 5.4 (A) to 5.6 (C) for the payoff matrixes. We can see the comparison of three different payoff matrixes that are showing in payoff matrix A with (ρ) under the set-up value 0.3, pay off matrix B with (ρ) under the set-up value 0.425, and payoff matrix C with (ρ) under the set-up value 0.6. We set payoff matrix A as the low value, payoff matrix B as the moderate value and payoff matrix C as the high value.



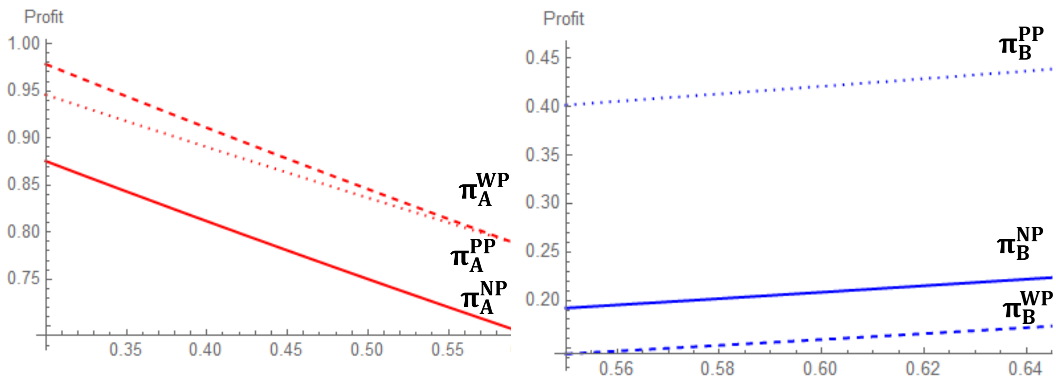
(A) Effect of ρ in low value
$\delta = 0.3, \rho = 0.3, \lambda = 0.355$

(B) Effect of ρ in the moderate value
$\delta = 0.3, \rho = 0.425, \lambda = 0.355$

(C) Effect of ρ in high value
$\delta = 0.3, \rho = 0.6, \lambda = 0.355$

Figure 5.4 Payoff Matrixes of Nine Scenarios with (ρ) under Value Range [0.3,0.6]

As we can see on the payoff matrixes A, B, and C, when discount factor for service B (ρ) is set up in low or high level, Developer A will always choose Contract W and Developer B will always choose Contract P. The trend of profit will decrease for Developer A but will increase for Developer B. This is because when Developer B implement the discount for their service, the potential market will likely choose the product from Developer B. The market will convert from Developer A to Developer B, so the profit of Developer B will increase. Both developers get the optimum equilibrium price under Scenario WP. This finding is supported by the graphs illustrated in Figure 5.5 (A) and 5.5 (B). Based on the payoff matrixes results, the potential change of strategy happened among Scenarios NP, WP, and PP, so we investigate the trend among them through the graphs below.



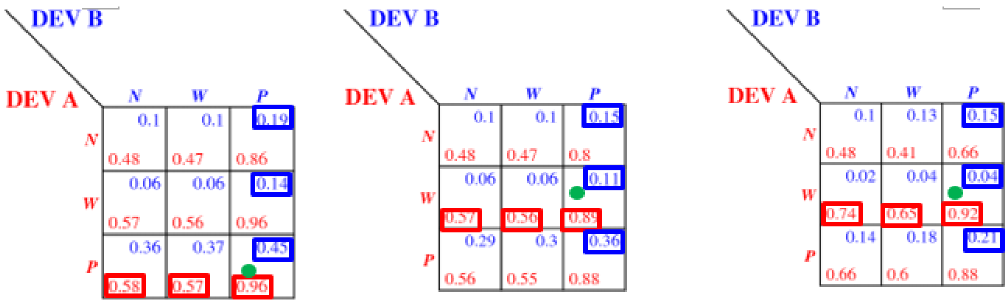
(A) Effect of ρ in Developer A
$\delta = 0.3, \rho = [0.3,0.6], \lambda = 0.355$

(A) Effect of ρ in Developer B
$\delta = 0.3, \rho = [0.3,0.6], \lambda = 0.355$

Figure 5.5 The Effect of (ρ) in Developer A and B (Scenario NP, WP, and PP)

We can analyze from the Figure 5.5 (A) and 5.5 (B), when the (ρ) is set up lower, Developer A will get a higher profit, but Developer B will get lower profit. In another hand, when the (ρ) is set up higher, Developer A will get lower profit, but Developer B will get higher profit. Here, Scenario WP comes out as the best response for Developer A and B to deal with the discount factor for service B (ρ).

(3) We examine Case 3 as follows: we set (δ) into 0.3, (ρ) into 0.425, and (λ) into the range of [0.155,0.555]. We got the result as illustrated in Figure 5.6 (A) to 5.6 (C) for the payoff matrixes. We can see the comparison in three different payoff matrixes that are showing payoff matrix A with (λ) under the set-up value 0.155, payoff matrix B with (λ) under the set-up value to 0.355, and payoff matrix C with (λ) under the set-up value 0.555. We set payoff matrix A as the low value, payoff matrix B as the moderate value and payoff matrix C as the high value.



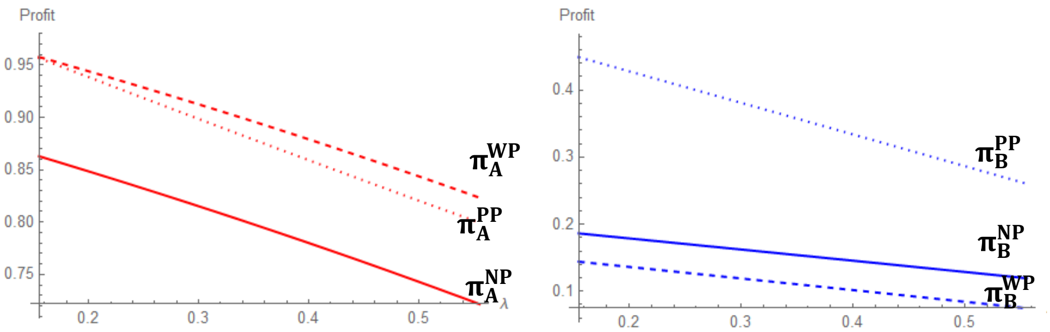
(A) Effect of λ in low value
$\delta = 0.3, \rho = 0.425, \lambda = 0.155$

(B) Effect of λ in the moderate value
$\delta = 0.3, \rho = 0.425, \lambda = 0.355$

(C) Effect of λ in high value
$\delta = 0.3, \rho = 0.425, \lambda = 0.555$

Figure 5.6 Payoff Matrix of Nine Scenarios with (λ) under Value Range [0.155,0.555]

As we can see on the payoff matrixes A, B, and C, when the intertemporal value discount (λ) for Contract P is set in low value, both developers will choose Contract P as their best response. This makes scenario PP as the equilibrium. Meanwhile, when the (λ) is set up in high value, Developer A will change its strategy to Contract W, meanwhile Developer B will stick to Contract P. So, the level of intertemporal value discount (λ) has the impact to change the strategy from Scenario PP to Scenario WP when the value of (λ) is high. This finding is supported by the graphs illustrated in Figure 5.7 (A) and 5.7 (B). Based on the payoff matrixes results, the potential change of strategy happened among Scenarios NP, WP, and PP, so we investigate the trend among them through the graphs below.



(A) Effect of λ in Developer A
$\delta = 0.6, \gamma = 0.8, \rho = 0.425,$
$\lambda = [0.115, 0.555]$

(B) Effect of λ in Developer B
$\delta = 0.6, \gamma = 0.8, \rho = 0.425,$
$\lambda = [0.115, 0.555]$

Figure 5.7 The Effect of (λ) in Developer A and B (Scenario NP, WP, and PP)

We can analyze from the Figure 5.7 (A) and 5.7 (B), when the (λ) is set in low value, Developer A and Developer B will get a likely higher profit, in another hand when the (λ) is set up higher, Developer A and Developer B will get lower profit. Here, Scenario PP comes out as the equilibrium when the (λ) is in low value. Meanwhile, Scenario WP comes out as the equilibrium when the (λ) is in high value. This means there is a strategy change for Developer A under the change level of (λ). After all, we summarize the findings into some points as follows:

1. Developer B will always stick to offer Contract P in all scenarios
    - This contract is suitable for a project with high level of RC such as project with high number of features and complexities that potentially propose a massive change to its requirements.
    - The client can expect a better quality of software because whenever the additional RC can be accommodated well, the valuation of the final software product will increase.
2. Contract W performs well when the level of (δ, ρ, λ) are high for Developer A
    - This contract is suitable for a project with additional RC intention with client's preference to single pricing contract.
3. Contract N never be chosen as the best response to deal with additional RC for both developers
    - Contract N only covers the initial requirement work, so the additional RC never be accommodated in this contract.
    - Due to that situation, in term of project size, this contract is suitable for a project with less complexity, small features with clear software requirement. This statement is stated in Jørgensen et al. (2017), the projects under this contract were on average smaller than those using Contract P.
    - Because the client could not propose for the additional RC, the project may work well under small software development package, such as company profile or personal page website that do not need high level of additional RC.
    - This finding aligns with Jørgensen et al. (2017) result in their empirical research. They found that clients using Contract N tended to have a stronger focus on low price/cost than clients using Contract P.
4. There are two exist equilibriums under Contract W and Contract P for both developers among the nine scenarios
    - Scenario PP and WP come out as the equilibrium among all scenarios. This is because when the level of δ, ρ, λ are high the equilibrium for both developers is under Scenario WP. Meanwhile when the level of λ is low the equilibrium for both developers is under Scenario PP.

5.2 Profit Comparison towards Two Parameters

We examine the influence between two parameters relationship. We execute each parameter towards the profits of some scenarios and draw them using the contour plot. We then analyze it in a comparative way with respect to only scenario NP, WP, and PP due to the previous discussion in subsection 5.1. Developer A is illustrated with the red line and Developer B with the blue line.

5.2.1 The Influences of δ and λ on The Profits

We explore the influence between the second-period valuation and intertemporal discount between two developers under Scenarios NP, WP, and PP. We draw the plot in Figure 5.8. In Developer A, when the level of second-period-valuation (δ) is high, and the intertemporal value discount (λ) is low, Developer A will choose Contract W. Meanwhile, when (λ) is high, Developer A will choose Contract P. Because Developer B always choose Contract P so there will be exist two equilibrium WP and PP when the (δ) and (λ) is set in low or high value.

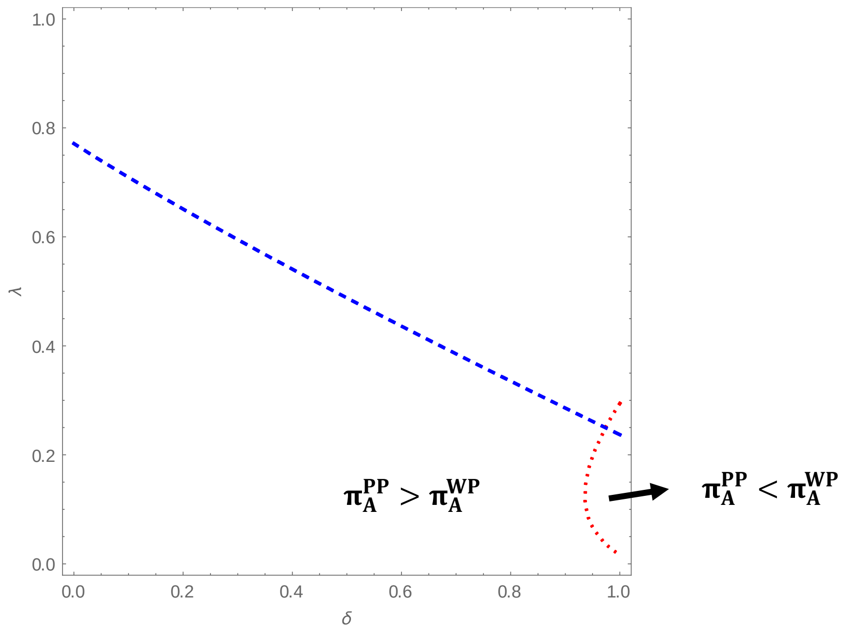


Figure 5.8 The Influences of δ and λ on The Profits

5.2.2 The Influences of ρ and λ on The Profits

We explore the influence between the discount factor for service B (ρ) and intertemporal value discount (λ) between two developers under Scenarios NP, WP, and PP. We draw the plot in Figure 5.9. We can observe when (λ) is in the high level, and (ρ) is low, Developer A will choose Contract P. Otherwise, when the discount for service B (ρ) is high, Developer A will choose Contract W. Because Developer B always choose Contract P so there will be exist two equilibriums WP and PP when the (λ) and (ρ) are set in low or high value.

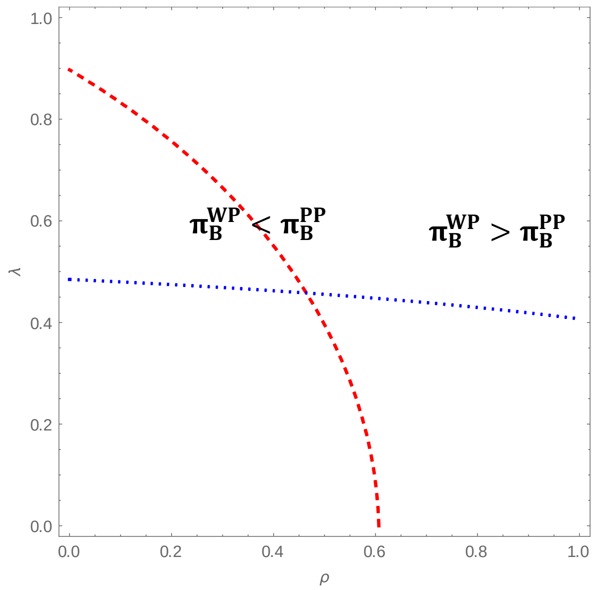


Figure 5.9 The Influences of ρ and λ on The Profits

6. CONCLUSION

6.1 Research Conclusions

This study considers a monopolistic and duopolistic market with pricing and contract designs for software outsourcing products. We propose two different contract types fixed price and time and materials, and develop three contract designs N, W, and P. In duopoly model, we construct nine combination scenarios in the duopoly market to examine which contract design performs as the best response using Nash Equilibrium. The presence of RC will make an impact on how the contract designs are constructed.

We construct a mathematical model for maximizing profit for the two developers, and sets price as a decision variable, and explores the impact of different parameters on the decision variables and profitability of the two developers with different degree of parameters. Based on the analysis results of Chapter 4 and 5, the major findings can be summarized in the following points:

1. Developer B will always stick to offer Contract P in all scenarios
2. Contract W performs well when the level of (δ, ρ, λ) are high for Developer A
3. Contract N never be chosen as the best response to deal with additional RC for both developers
4. There are two exist equilibriums under Contract W and Contract P for both developers among the nine scenarios

6.2 Future Directions

This study proposes insight on competition between two developers, which offering three contract designs under nine different scenarios. The three contracts are designed with respect to the presence of RC. We discuss the best response for the strategic decision that Developer A or Developer B can choose. Our future directions include, consider the effort responding to RC as decision variable, model the time to represent the project delivery time (on time or late), compare the findings under the scheme of software as a service (SaaS) subscription, and examine the findings with the empirical data from real case practices of software contracts

REFERENCES

Ali, N., & Lai, R. (2016). A method of requirements change management for Global software development. *Information and Software Technology*, *70*, 49–67. https://doi.org/10.1016yj/infsof.2015.09.005

AlSanad, A., & Chikh, A. (2015). The Impact of Software Requirement Change – A Review. In *Á. Rocha et al. (eds.), New Contributions in Information Systems and Technologies, Advances in Intelligent Systems and Computing 353* (Vol. 353). https://doi.org/10.1007/978-3-319-16528-8

Bajari, P. (2001). Incentives versus Transaction Costs : A Theory of Procurement Contracts Author (s): Patrick Bajari and Steven Tadelis Published by : Wiley on behalf of RAND Corporation Stable URL : http://www.jstor.org/stable/2696361 Your use of the JSTOR archive indic, *32*(3), 387–407.

Bala, R., & Carr, S. (2010). Usage-based pricing of software services under competition. *Journal of Revenue and Pricing Management*, *9*(3), 204–216. https://doi.org/10.1057/rpm.2010.12

Basri, S., Kama, N., Haneem, F., & Ismail, S. A. (2016). Predicting effort for requirement changes during software development. In *Proceedings of the Seventh Symposium on Information and Communication Technology - SoICT '16* (pp. 380–387). New York, New York, USA: ACM Press. https://doi.org/10.1145/3011077.3011096

Bolton, P., & Dewatripont, M. (2005). *Contract theory*. *Reimagining Contract Law Pedagogy: A New Agenda for Teaching*. The MIT Press Cambridge, Massachusetts London, England. https://doi.org/10.4324/9781315178189

Chen, Y., & Bharadwaj, A. (2009). An empirical analysis of contract structures in IT outsourcing. *Information Systems Research*, *20*(4), 484–506. https://doi.org/10.1287/isre.1070.0166

Cheng, H. K., Li, S., & Liu, Y. (2015). Optimal software free trial strategy: Limited version, time-locked, or hybrid? *Production and Operations Management*, *24*(3), 504–517. https://doi.org/10.1111/poms.12248

Cheng, H. K., & Tang, Q. C. (2010). Free trial or no free trial: Optimal software product design with network effects. *European Journal of Operational Research*, *205*(2), 437–447. https://doi.org/10.1016/j.ejor.2010.01.014

Choudhary, V., Ghose, A., Mukhopadhyay, T., & Rajan, U. (2005). Personalized pricing and quality differentiation. *Management Science*, *51*(7), 1120–1130. https://doi.org/10.1287/mnsc.1050.0383

Corts, K. S. (2012). The interaction of implicit and explicit contracts in construction and procurement contracting. *Journal of Law, Economics, and Organization*, *28*(3), 550–568. https://doi.org/10.1093/jleo/ewr023

Dey, D., Fan, M., & Zhang, C. (2010). Design and analysis of contracts for software outsourcing. *Information Systems Research*, *21*(1), 93–114. https://doi.org/10.1287/isre.1080.0223

Dharma Kwon, H., Lippman, S. A., & Tang, C. S. (2010). Optimal time-based and cost-based coordinated project contracts with unobservable work rates. *International Journal of Production Economics*, *126*(2), 247–254. https://doi.org/10.1016/j.ijpe.2010.03.013

Fink, L., Lichtenstein, Y., & Wyss, S. (2013). Ex post adaptations and hybrid contracts in software development services. *Applied Economics*, *45*(32), 4533–4544. https://doi.org/10.1080/00036846.2013.791021

Gaebert, C. (2014). Dilemma structures between contracting parties in software development projects. *ICSOFT-EA 2014 - Proceedings of the 9th International Conference on Software Engineering and Applications*, 539–548.

Ghosh, S., Ramaswamy, S., & Jetley, R. P. (2013). Towards requirements change decision support. *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, *1*, 148–155. https://doi.org/10.1109/APSEC.2013.30

Gopal, A., Sivaramakrishnan, K., Krishnan, M. S., & Mukhopadhyay, T. (2003). Contracts in Offshore Software Development: An Empirical Analysis. *Management Science*, *49*(12), 1671–1683. https://doi.org/10.1287/mnsc.49.12.1671.25120

Jørgensen, M., Mohagheghi, P., & Grimstad, S. (2017a). Direct and indirect connections between type of contract and software project outcome. *International Journal of Project Management*, *35*(8), 1573–1586. https://doi.org/10.1016/j.ijproman.2017.09.003

Jørgensen, M., Mohagheghi, P., & Grimstad, S. (2017b). Direct and indirect connections between type of contract and software project outcome. *International Journal of Project Management*, *35*(8), 1573–1586. https://doi.org/10.1016/j.ijproman.2017.09.003

Kalnins, A., & Mayer, K. J. (2004). Relationships and hybrid contracts: An analysis of contract choice in information technology. *Journal of Law, Economics, and Organization*, *20*(1), 207–229. https://doi.org/10.1093/jleo/ewh030

Lippman, S. A., McCardle, K. F., & Tang, C. S. (2013). Using Nash bargaining to design project management contracts under cost uncertainty. *International Journal of Production Economics*, *145*(1), 199–207. https://doi.org/10.1016/j.ijpe.2013.04.036

Liu, Y., Cheng, H. K., Tang, Q. C., & Eryarsoy, E. (2011). Optimal software pricing in the presence of piracy and word-of-mouth effect. *Decision Support Systems*, *51*(1), 99–107. https://doi.org/10.1016/j.dss.2010.11.032

Mao, C., Lu, Y., & Wang, X. (2006). A study on the distribution and cost prediction of requirements changes in the software life-cycle. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 3840 LNCS, pp. 136–150). https://doi.org/10.1007/11608035_14

McGee, S., & Greer, D. (2012). Towards an understanding of the causes and effects of software requirements change: Two case studies. *Requirements Engineering*, *17*(2), 133–155. https://doi.org/10.1007/s00766-012-0149-0

Mehra, A., Bala, R., & Sankaranarayanan, R. (2012). Competitive behavior-based price discrimination for software upgrades. *Information Systems Research*, *23*(1), 60–74.

https://doi.org/10.1287/isre.1100.0291

Nan, G., Shi, F., Dou, R., & Li, M. (2016). Duopoly pricing of software products under free strategy: Limited-feature vs. seeding. *Computers and Industrial Engineering*, *100*, 13–23. https://doi.org/10.1016/j.cie.2016.07.026

Nurmuliani, N., Zowghi, D., & Williams, S. P. (2006). Requirements Volatility and Its Impact on Change Effort: Evidence Based Research in Software Development Projects.". *Australian Workshop on Requirements Engineering*. Retrieved from http://www.researchgate.net/publication/228946043_Requirements_volatility_and_its_impact_on_change_effort_Evidence-based_research_in_software_development_projects/file/9c960520ecb3089ce7.pdf

Oh, S., Rhodes, J., & Strong, R. (2016). Impact of cost uncertainty on pricing decisions under risk aversion. *European Journal of Operational Research*, *253*(1), 144–153. https://doi.org/10.1016/j.ejor.2016.02.034

Rohitratana, J., & Altmann, J. (2012). Impact of pricing schemes on a market for Software-as-a-Service and perpetual software. *Future Generation Computer Systems*, *28*(8), 1328–1339. https://doi.org/10.1016/j.future.2012.03.019

Sia, S. K., Koh, C., & Tan, C. X. (2008). Strategic maneuvers for outsourcing flexibility: An empirical assessment. *Decision Sciences*, *39*(3), 407–443. https://doi.org/10.1111/j.1540-5915.2008.00198.x

Suprapto, M., Bakker, H. L. M., Mooi, H. G., & Hertogh, M. J. C. M. (2016). How do contract types and incentives matter to project performance? *International Journal of Project Management*, *34*(6), 1071–1087. https://doi.org/10.1016/j.ijproman.2015.08.003

Tong, Z., Su, X., Cen, L., & Wang, T. (2017). Greening Software Requirements Change Management Strategy Based on Nash Equilibrium. *Wireless Communications and Mobile Computing*, *2017*, 1–10. https://doi.org/10.1155/2017/4020162

Appendix A. Proofs

A.1 The optimal solution in p_A^{P1} and p_A^{P2} under the Contract P.

We look at the Hessian matrix, which is given by

$$det(\boldsymbol{H^P}) = \begin{bmatrix} \frac{\partial^2 \pi_A^P}{\partial p_A^{P2^2}} & \frac{\partial \pi_A^P}{\partial p_A^{P2} \partial p_A^{P1}} \\ \frac{\partial \pi_A^P}{\partial p_A^{P1} \partial p_A^{P2}} & \frac{\partial^2 \pi_A^P}{\partial p_A^{P1^2}} \end{bmatrix} = 0 \geq 0.$$

It is easy to see that $\boldsymbol{H^P}$ is negative semidefinite for any value of p_A^{P1} and p_A^{P2}. This is because det($\boldsymbol{H^P}$) ≥ 0. We derive the second profit from seeing whether the objective function in p_A^{P1} and p_A^{P2} are concave or convex, so we get the result of the second derivation, as follows:

$$\frac{\partial^2 \pi_A^P}{\partial p_A^{P1^2}} = -\frac{1}{1+\delta} + \frac{\lambda}{2\gamma(1+\delta)} < 0,$$

and

$$\frac{\partial^2 \pi_A^P}{\partial p_A^{P2^2}} = -\frac{2\gamma}{1+\delta} < 0.$$

Therefore, the objective functions in p_A^{P1} and p_A^{P2} are concave, and the equilibrium results are the optimal solutions. □

We take three examples of the optimal price solution for the duopoly model. We choose NN, WN, and PN to represent the nine scenarios combination.

A.2 The optimal solution in p_A^N and p_B^N under the NN scenario.

We look at the Hessian matrix, which is given by

$$det(\boldsymbol{H^{NN}}) = \begin{bmatrix} \frac{\partial^2 \pi_A^{NN}}{\partial p_A^{N^2}} & \frac{\partial \pi_A^{NN}}{\partial p_A^N \partial p_B^N} \\ \frac{\partial \pi_B^{NN}}{\partial p_B^N \partial p_A^{NN}} & \frac{\partial^2 \pi_B^{NN}}{\partial p_B^{N^2}} \end{bmatrix} = \frac{3}{4t^2} > 0.$$

It is easy to see that $\boldsymbol{H^{NN}}$ is negative semidefinite for any value of p_A^N and p_B^N. This is because det($\boldsymbol{H^{NN}}$) > 0. We derive the second profit from seeing whether the objective function in p_A^N and p_B^N are concave or convex, so we get the result of the second derivation, as follows:

$$\frac{\partial^2 \pi_A^{NN}}{\partial {p_A^N}^2} = -\frac{1}{t} < 0,$$

and

$$\frac{\partial^2 \pi_B^{NN}}{\partial {p_B^N}^2} = -\frac{1}{t} < 0.$$

Therefore, the objective functions in p_A^N and p_B^N are concave, and the equilibrium results are the optimal solutions. □

A.3 The optimal solution in p_A^W and p_B^N under scenario WN.

We look at the Hessian matrix, which is given by

$$det(\boldsymbol{H^{WN}}) = \begin{bmatrix} \frac{\partial^2 \pi_A^W}{\partial {p_A^W}^2} & \frac{\partial \pi_A^W}{\partial p_A^W \partial p_B^N} \\ \frac{\partial \pi_B^N}{\partial p_B^N \partial p_A^W} & \frac{\partial^2 \pi_B^N}{\partial {p_B^N}^2} \end{bmatrix} = \frac{3}{4t^2} > 0.$$

It is easy to see that $\boldsymbol{H^{WN}}$ is negative semidefinite for any value of p_A^W and p_B^N. This is because det($\boldsymbol{H^{WN}}$) > 0. We derive the second profit from seeing whether the objective function in p_A^W and p_B^N are concave or convex, so we get the result of the second derivation, as follows:

$$\frac{\partial^2 \pi_A^{WN}}{\partial {p_A^W}^2} = -\frac{1}{t} < 0,$$

and

$$\frac{\partial^2 \pi_B^{WN}}{\partial {p_B^N}^2} = -\frac{1}{t} < 0.$$

Therefore, the objective functions in p_A^W and p_B^N are concave, and the equilibrium results are the optimal solutions. □

A.4 The optimal solution in p_A^P and p_B^N under scenario PN.

We look at the Hessian matrix, which is given by

$$det(H^{PN}) = \begin{bmatrix} \frac{\partial^2 \pi_A^{PN}}{\partial {p_A^P}^2} & \frac{\partial \pi_A^{PN}}{\partial p_A^P \partial p_B^N} \\ \frac{\partial \pi_B^{PN}}{\partial p_B^N \partial p_A^P} & \frac{\partial^2 \pi_B^{PN}}{\partial {p_B^N}^2} \end{bmatrix} = \frac{3}{4t^2} > 0.$$

It is easy to see that $\boldsymbol{H^{PN}}$ is negative semidefinite for any value of p_A^P and p_B^N. This is because det($\boldsymbol{H^{PN}}$) > 0. We derive the second profit from seeing whether the objective function in ${p_A^{P1}}, {p_A^{P2}}$ and p_B^N are concave or convex, so we get the result of the second derivation, as follows:

$$\frac{\partial^2 \pi_A^{PN}}{\partial {p_A^{P1}}^2} = -\frac{1}{2t} + \frac{\lambda}{4t\gamma} < 0,$$

and

$$\frac{\partial^2 \pi_A^{PN}}{\partial {p_A^{P2}}^2} = -\frac{\gamma}{t} < 0.$$

Developer's B objective function in p_B^N as follows:

$$\frac{\partial^2 \pi_B^{PN2}}{\partial {p_B^N}^2} = -\frac{1}{t} < 0.$$

Therefore, the objective functions in p_A^{P1} , p_A^{P2} and p_B^N are concave, and the equilibrium results are the optimal solutions. □

Proof of Proposition 1. The first order conditions of the optimal price and profit in Contract W, price $p_A^{W^*}$and profit $\pi_A^{W^*}$decisions are:

$$\frac{\partial p_A^{W^*}}{\partial \delta} > 0, \frac{\partial \pi_A^{W^*}}{\partial \delta} > 0.$$

We analyze the impact of the degree of various influential parameters on Contract W price $p_A^{W^*}$and profit $\pi_A^{W^*}$ decision: the primary valuation θ, the second-period valuation due to additional RC δ, price sensitivity γ for second-period price p_A^{P2}, the investment cost for additional RC F_A and intertemporal value discount for second-period profit λ. Since we only focus on the second-period valuation due to additional RC δ, so we proof only the influence of δ on $p_A^{W^*}$ and $\pi_A^{W^*}$.

$$\frac{\partial p_A^{W^*}}{\partial \delta} = \frac{1}{2},$$

$$\frac{\partial \pi_A^{W^*}}{\partial \delta} = \frac{(1 + \delta - c_{A1} - c_{A2})(1 + \delta + c_{A1} + c_{A2})}{4(1 + \delta)^2}.$$

According to the assumption if the price is always greater than zero (due to $p_A^{W^*} > 0$) and the denominator are positive for both prices and profit. So, we know that $\frac{\partial p_A^{W^*}}{\partial \delta} > 0$, which implies that the optimal price increases with the second period valuation due to RC δ. The greater the δ is, the higher the developers' optimal price is. Similarly, $\frac{\partial \pi_A^{W^*}}{\partial \delta} > 0$ which indicates that the optimal profit of developers increases with the δ. Because for any c_A in [0, 0.1]. c_A must be greater than zero.

Proof of Proposition 2. The first order conditions of the optimal prices and profit in Contract P, prices $p_A^{P1^*}, p_A^{P2^*}$ and profit $\pi_A^{P^*}$ decisions are:

$$\frac{\partial p_A^{P1^*}}{\partial \delta} > 0, \frac{\partial p_A^{P2^*}}{\partial \delta} > 0, \frac{\partial \pi_A^{P^*}}{\partial \delta} < 0.$$

We analyze the impact of the degree of second period valuation due to RC δ on Contract P prices $p_A^{P1^*}, p_A^{P2^*}$ and profit $\pi_A^{P^*}$ decision. Since we only focus on the second-period valuation due to additional RC δ. So, we proof only the influence of δ on $p_A^{P1^*}, p_A^{P2^*}$ and profit $\pi_A^{P^*}$.

$$\frac{\partial p_A^{P1^*}}{\partial \delta} = 1 + \frac{\gamma}{-2\gamma + \lambda},$$

$$\frac{\partial p_A^{P2^*}}{\partial \delta} = \frac{1}{4\gamma - 2\lambda},$$

$$\frac{\partial \pi_A^{P^*}}{\partial \delta} = -\frac{\gamma(-1 - \delta + c_A + \gamma c_2)(1 + \delta + c_A + \gamma c_A)}{4(1+\delta)^2(2\gamma - \lambda)}.$$

According to the assumption if the price is always greater than zero (due to $p_A^{P1^*}, p_A^{P2^*} > 0$) and the numerator are positive for both prices. So, we know that $\frac{\partial p_A^{P1^*}}{\partial \delta} > 0$ and $\frac{\partial p_A^{P2^*}}{\partial \delta} > 0$ which implies that the optimal prices increase with the second period valuation due to RC δ. The greater the δ are, the higher the developers' optimal price are. Similarly, $\frac{\partial \pi_A^{P^*}}{\partial \delta} < 0$ which indicates that the optimal profit of developers decreases with the increase of δ. Because we see the denominator is always positive due to assumption above, but we can see if the numerator is always negative $-\gamma(-1 - \delta + c_A + \gamma c_A)(1 + \delta + c_A + \gamma c_A)$.

Proof of Proposition 3. The first order conditions of the optimal prices and profits in Scenario NP, prices $p_A^{NP^*}, p_B^{NP1^*}, p_B^{NP2^*}$ and profit $\pi_A^{NP^*}, \pi_B^{NP^*}$ decisions are:

$$\frac{\partial p_A^{NP^*}}{\partial \delta} < 0, \frac{\partial \pi_A^{NP^*}}{\partial \delta} < 0,$$

$$\frac{\partial p_B^{NP1^*}}{\partial \delta} > 0, \frac{\partial p_B^{NP2^*}}{\partial \delta} > 0, \frac{\partial \pi_B^{NP^*}}{\partial \delta} < 0,$$

We analyze the impact of the degree of second period valuation due to RC δ on Scenario NP prices $p_A^{NP^*}, p_B^{NP1^*}, p_B^{NP2^*}$ and profit $\pi_A^{NP^*}$, $\pi_B^{NP^*}$ decisions. Since we only focus on the second-period valuation due to additional RC δ so we proof only the influence of δ on prices $p_A^{NP^*}, p_B^{NP1^*}, p_B^{NP2^*}$ and profit $\pi_A^{NP^*}$, $\pi_B^{NP^*}$.

$$\frac{\partial p_A^{NP^*}}{\partial \delta} = -\frac{S\gamma\rho}{3\gamma - \lambda},$$

$$\frac{\partial \pi_A^{NP^*}}{\partial \delta} = \frac{S\gamma\rho(-7t\gamma + 4t\lambda + S\gamma(-1 + \rho + \delta\rho) + \gamma c_{\mathrm{A}} - \gamma(c_{\mathrm{B}} + \gamma c_{\mathrm{B}}))}{2t(-3\gamma + \lambda)^2},$$

$$\frac{\partial p_B^{NP1^*}}{\partial \delta} = \frac{S(\gamma - \lambda)\rho}{3\gamma - \lambda},$$

$$\frac{\partial p_B^{NP2^*}}{\partial \delta} = \frac{S\rho}{6\gamma - 2\lambda},$$

$$\frac{\partial \pi_B^{NP^*}}{\partial \delta} = -\frac{S\gamma(2\gamma - \lambda)\rho(S - 5t - S(1 + \delta)\rho - c_{\mathrm{A}} + c_{\mathrm{B}} + \gamma c_{\mathrm{B}})}{4t(-3\gamma + \lambda)^2}.$$

We can analyze if the numerator is negative $-S\gamma\rho$ and the denominator is positive due to the proof above because $p_A^{NP^*} > 0$. Overall, the denominator is always positive. So, the $\frac{\partial p_A^{NP^*}}{\partial \delta} < 0$, which implies that the optimal prices decrease with the second period valuation due to RC δ. The greater the δ are, the lower the developers' optimal price are. We then examine the profit of Developer A $\pi_A^{NP^*}$, we can see if the denominator is always positive based on the proof above. So, we examine the numerator $S\gamma\rho\big(-7t\gamma + 4t\lambda + S\gamma(-1 + \rho + \delta\rho) + \gamma c_{\mathrm{A}} - \gamma(c_{\mathrm{B}} + \gamma c_{\mathrm{B}})\big)$. We find that the developers' optimum profit increases with δ when $S\gamma\rho\big(-7t\gamma + 4t\lambda + S\gamma(-1 + \rho + \delta\rho) + \gamma c_{\mathrm{A}} - \gamma(c_{\mathrm{B}} + \gamma c_{\mathrm{B}})\big) > 0$, it decreases with δ when $S\gamma\rho\big(-7t\gamma + 4t\lambda + S\gamma(-1 + \rho + \delta\rho) + \gamma c_{\mathrm{A1}} - \gamma(c_{\mathrm{B}} + \gamma c_{\mathrm{B}})\big) < 0$. Note that $\pi_A^{NP^*} > 0$, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$, which means that the developers and clients will perceive the same discount values of the second period compared to the first period. After simplification we get $\frac{S\rho(-3t+S(-1+\rho+\delta\rho))}{8t}$, which implies that $\frac{\partial \pi_A^{NP^*}}{\partial \delta} < 0$. We proof that $\frac{\partial p_B^{NP1^*}}{\partial \delta} > 0, \frac{\partial p_B^{NP2^*}}{\partial \delta} > 0$ due to some assumptions and properties above. Meanwhile, $\frac{\partial \pi_B^{NP^*}}{\partial \delta} < 0$, we can see the denominator is always positive for

$4t(-3\gamma+\lambda)^2$ and the numerator, it suffices to check the sign of the following derivative, which is trivially negative $-\frac{S\gamma(2\gamma-\lambda)\rho(S-5t-S(1+\delta)\rho-c_{A1}+c_{B1}+\gamma c_{B2})}{4t(-3\gamma+\lambda)^2}$.

Proof of Proposition 4. The first order conditions of the optimal prices and profits in Scenario WP, prices $p_A^{WP^*}, p_B^{WP1^*}, p_B^{WP2^*}$ and profit $\pi_A^{WP^*}$, $\pi_B^{WP^*}$ decisions are:

$$\frac{\partial p_A^{WP^*}}{\partial \delta} < 0, \frac{\partial \pi_A^{WP^*}}{\partial \delta} > 0,$$

$$\frac{\partial p_B^{WP1^*}}{\partial \delta} > 0, \frac{\partial p_B^{WP2^*}}{\partial \delta} > 0, \frac{\partial p_B^{WP^*}}{\partial \delta} > 0,$$

We analyze the impact of the degree of second period valuation due to RC δ on Scenario WP prices $p_A^{WP^*}, p_B^{WP1^*}, p_B^{WP2^*}$ and profit $\pi_A^{WP^*}$, $\pi_B^{WP^*}$ decisions. Since we only focus on the second-period valuation due to additional RC δ so we proof only the influence of δ on prices $p_A^{WP^*}, p_B^{WP1^*}, p_B^{WP2^*}$ and profit $\pi_A^{WP^*}$, $\pi_B^{WP^*}$.

$$\frac{\partial p_A^{WP^*}}{\partial \delta} = -\frac{S\gamma(-1+\rho)}{3\gamma-\lambda},$$

$$\frac{\partial \pi_A^{WP^*}}{\partial \delta} = \frac{\begin{array}{c} S\gamma(-1+\rho)(t(-7\gamma+4\lambda)+S\gamma(1+\delta)(-1+\rho) \\ +\gamma(2c_A-c_B-\gamma c_B)) \end{array}}{2t(-3\gamma+\lambda)^2},$$

$$\frac{\partial p_B^{WP1^*}}{\partial \delta} = \frac{S(\gamma-\lambda)(-1+\rho)}{3\gamma-\lambda},$$

$$\frac{\partial p_B^{WP2^*}}{\partial \delta} = \frac{S(-1+\rho)}{6\gamma-2\lambda},$$

$$\frac{\partial p_B^{WP^*}}{\partial \delta} = \frac{\begin{array}{c} S\gamma(2\gamma-\lambda)(-1+\rho)(5t+S(1+\delta)(-1+\rho) \\ +2c_{A1}-c_B-\gamma c_B) \end{array}}{4t(-3\gamma+\lambda)^2}.$$

We can analyze if the numerator is negative $-S\gamma(-1+\rho)$ and the denominator is always positive due to the proof above $p_A^{WP^*} > 0$. Because it suffices to see the sign of the derivative above is trivially negative. So, the $\frac{\partial p_A^{WP^*}}{\partial \delta} < 0$, which implies that the optimal prices decrease with the second period valuation due to RC δ. We then examine the profit of Developer A $\pi_A^{WP^*}$, we can see if the denominator is always positive based on the proof above. So, we examine the numerator $S\gamma(-1+\rho)(t(-7\gamma+4\lambda)+S\gamma(1+\delta)(-1+\rho)+\gamma(2c_A-c_B-\gamma c_B))$. We find that the developers' optimum profit increases with δ when $S\gamma(-1+\rho)(t(-7\gamma+4\lambda)+S\gamma(1+\delta)(-1+\rho)+\gamma(2c_A-c_B-\gamma c_B)) > 0$, it decreases with δ when $S\gamma(-1+\rho)(t(-7\gamma+4\lambda)+S\gamma(1+\delta)(-1+\rho)+\gamma(2c_A-c_B-\gamma c_B)) < 0$.

Note that $\pi_A^{WP^*} > 0$, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$, which means that the developers and clients will perceive the same discount values of the second period compared to the first period. After simplification we get $\frac{S(-3t+S(1+\delta)(-1+\rho))(-1+\rho)}{8t}$, which implies that $\frac{\partial\pi_A^{WP^*}}{\partial\delta} > 0$.

We proof that $\frac{\partial p_B^{WP1^*}}{\partial\delta} > 0, \frac{\partial p_B^{WP2^*}}{\partial\delta} > 0$ due to some assumptions and properties above. Meanwhile, we then examine the profit of Developer B $\pi_B^{WP^*}$, we can see if the denominator is always positive based on the proof above. So, we examine the numerator $S\gamma(2\gamma-\lambda)(-1+\rho)(5t+S(1+\delta)(-1+\rho)+2c_{\mathrm{A}}-c_{\mathrm{B}}-\gamma c_{\mathrm{B}})$. We find that the developers' optimum profit increases with δ when $\gamma(2\gamma-\lambda)(-1+\rho)(5t+S(1+\delta)(-1+\rho)+2c_{\mathrm{A}}-c_{\mathrm{B}}-\gamma c_{\mathrm{B}}) > 0$, it decreases with δ when $S\gamma(2\gamma-\lambda)(-1+\rho)(5t+S(1+\delta)(-1+\rho)+2c_{\mathrm{A}}-c_{\mathrm{B}}-\gamma c_{\mathrm{B}}) < 0$. Note that $\pi_B^{WP^*} > 0$, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$, which means that the developers and clients will perceive the same discount values of the second period compared to the first period. After simplification we get $\frac{S(5t+S(1+\delta)(-1+\rho))(-1+\rho)}{16t}$, which implies that $\frac{\partial\pi_B^{WP^*}}{\partial\delta} > 0$.

Proof of Proposition 5. The first order conditions of the optimal prices and profits in Scenario PP, prices $p_A^{PP1^*}, p_A^{PP2^*}, p_B^{PP1^*}, p_B^{PP2^*}$ and profit $\pi_A^{WP^*}$, $\pi_B^{WP^*}$ decisions are:

$$\frac{\partial p_A^{PP1^*}}{\partial\delta} < 0, \frac{\partial p_A^{PP2^*}}{\partial\delta} > 0, \frac{\partial\pi_A^{PP^*}}{\partial\delta} > 0,$$

$$\frac{\partial p_B^{PP1^*}}{\partial\delta} > 0, \frac{\partial p_B^{PP2^*}}{\partial\delta} > 0, \frac{\partial\pi_B^{PP^*}}{\partial\delta} > 0,$$

We analyze the impact of the degree of second period valuation due to RC δ on Scenario PP prices $p_A^{PP1^*}, p_A^{PP2^*}, p_B^{PP1^*}, p_B^{PP2^*}$ and profit $\pi_A^{WP^*}$, $\pi_B^{WP^*}$ decisions. Since we only focus on the second-period valuation due to additional RC δ so we proof only the influence of δ on prices $p_A^{PP1^*}, p_A^{PP2^*}, p_B^{PP1^*}, p_B^{PP2^*}$ and profit $\pi_A^{WP^*}, \pi_B^{WP^*}$

$$\frac{\partial p_A^{PP1^*}}{\partial\delta} = -\frac{S(3\gamma-2\lambda)(-1+\rho)}{9\gamma-4\lambda},$$

$$\frac{\partial p_A^{PP2^*}}{\partial\delta} = \frac{S-S\rho}{9\gamma-4\lambda},$$

$$\frac{\partial\pi_A^{PP^*}}{\partial\delta} = \frac{S^2\gamma(3\gamma-\lambda)(-1+\rho)^2}{t(9\gamma-4\lambda)^2},$$

$$\frac{\partial p_B^{PP1^*}}{\partial \delta} = \frac{S(3\gamma - 2\lambda)(-1+\rho)}{9\gamma - 4\lambda},$$

$$\frac{\partial p_B^{PP2^*}}{\partial \delta} = \frac{S(-1+\rho)}{9\gamma - 4\lambda},$$

$$\frac{\partial \pi_B^{PP^*}}{\partial \delta} = \frac{\begin{array}{c} S(3\gamma - \lambda)(-1+\rho)(t(9\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) \\ +\gamma(c_{\mathrm{A}} + \gamma c_{\mathrm{A}} - c_{\mathrm{B}} - \gamma c_{\mathrm{B}})) \end{array}}{t(9\gamma - 4\lambda)^2}.$$

We can analyze if the numerator is negative $-S(3\gamma - 2\lambda)(-1+\rho)$ and the denominator is always positive due to the proof above $p_A^{PP^*} > 0$. Because it suffices to see the sign of the derivative above is trivially negative. So, the $\frac{\partial p_A^{PP1^*}}{\partial \delta} < 0$, which implies that the optimal price decreases with the second period valuation due to RC δ. Similarly, based on the proof above $\frac{\partial p_A^{PP2^*}}{\partial \delta} > 0$ which indicates the optimal price increases when the second period valuation due to RC δ is high. We examine $\frac{\partial \pi_A^{PP^*}}{\partial \delta} > 0$ is positive based on some assumption and proof above. This is also similar with $\frac{\partial p_B^{PP1^*}}{\partial \delta} > 0, \frac{\partial p_B^{PP2^*}}{\partial \delta} > 0$which implies that the optimal prices increase with the second period valuation due to RC δ. Meanwhile, we then examine the profit of Developer B $\pi_B^{PP^*}$, we can see if the denominator is always positive $t(9\gamma - 4\lambda)^2$. So, we examine the numerator $S(3\gamma - \lambda)(-1+\rho)(t(9\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) + \gamma(c_{\mathrm{A}} + \gamma c_{\mathrm{A}} - c_{\mathrm{B}} - \gamma c_{\mathrm{B}}))$ .We find that the developers' optimum profit increases with δ when $(3\gamma - \lambda)(-1+\rho)(t(9\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) + \gamma(c_{\mathrm{A}} + \gamma c_{\mathrm{A}} - c_{\mathrm{B}} - \gamma c_{\mathrm{B}})) > 0$, it decreases with δ when $(3\gamma - \lambda)(-1+\rho)(t(9\gamma - 4\lambda) + S\gamma(1+\delta)(-1+\rho) + \gamma(c_{\mathrm{A}} + \gamma c_{\mathrm{A}} - c_{\mathrm{B}} - \gamma c_{\mathrm{B}})) < 0$. Note that $\pi_B^{PP^*} > 0$, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$, which means that the developers and clients will perceive the same discount values of the second period compared to the first period. After simplification we get $\frac{2S(5t+S(1+\delta)(-1+\rho))(-1+\rho)}{25t}$, which implies that $\frac{\partial \pi_B^{PP^*}}{\partial \delta} > 0$.

Proof of Proposition 6. We analyze the threshold of the difference between the two scenarios WP and PP. For the shake of simplicity, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$. Firstly, we solve the difference in $\Delta\pi_A^i = \pi_A^{WP} - \pi_A^{PP}$, we get the result as follows:

$$\Delta\pi_A^i = \frac{(5t + S(1+\delta)(-1+\rho))(-35t + 9S(1+\delta)(-1+\rho)) + 400t\lambda}{400t}$$

We see that the denominator is always positive, so we need to consider the numerator of $\pi_A^{\boldsymbol{WP}} - \pi_A^{\boldsymbol{PP}}$. We derive the δ from the numerator, $\partial^2(5t + S(1+\delta)(-1+\rho))(-35t + 9S(1+\delta)(-1+\rho)) + 400t\lambda F_A/\partial\delta$. We take the limit values of δ into the numerator. We set $\delta = 0$ and δ =1, and we find that when δ is equal to 0, the numerator is positive $(5t + S(-1 + \rho))(-35t + 9S(-1 + \rho)) + 400t\lambda F_A$. And when δ is equal to 1, it is also positive $(5t + 2S(-1 + \rho))(-35t + 18S(-1 + \rho)) + 400t\lambda F_A$, as proven above. So, we can confirm that there is not exist a threshold of δ denoted by $0 > \delta^{\dagger} \geq 1$. Above which $\pi_A^{\boldsymbol{WP}} - \pi_A^{\boldsymbol{PP}} > 0 \Leftrightarrow \pi_A^{\boldsymbol{WP}} > \pi_A^{\boldsymbol{PP}}$. Based on the above properties, we solve the $\pi_A^{WP} - \pi_A^{PP} = 0$ concerning δ. We get two roots as below:

Giving two roots $(i.e., \frac{1}{9}(-9 + \frac{5t}{S - S\rho} + \frac{20\sqrt{S^2 t(-1+\rho)^2(4t - 9\lambda F_A)}}{S^2(-1+\rho)^2}))$.

We know that a root above exceeds one, thus

$$\delta^{\dagger} \equiv \frac{1}{9}(-9 + \frac{5t}{S - S\rho} - \frac{20\sqrt{S^2 t(-1+\rho)^2(4t - 9\lambda F_A)}}{S^2(-1+\rho)^2})$$

Proof of Proposition 7. (1) We analyze the threshold of the difference between the two scenarios PN and PW in Developer B. For the shake of simplicity, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$. Firstly, we solve the difference in $\Delta\pi_B^i = \pi_B^{PN} - \pi_B^{PW}$, we get the result as follows:

$$\Delta\pi_B^i = -\frac{S\delta\rho(6t - 2S(1 + \delta) + S(2 + \delta)\rho)}{16t}.$$

We see that the denominator is always positive, so we need to consider the numerator of $\pi_B^{\boldsymbol{PN}} - \pi_B^{\boldsymbol{PW}}$. Otherwise, it is obvious to see if the numerator is negative. So, we can confirm from the result above which $\pi_B^{\boldsymbol{PN}} - \pi_B^{\boldsymbol{PW}} < 0 \Leftrightarrow \pi_B^{\boldsymbol{PN}} < \pi_B^{\boldsymbol{PW}}$.

Proof of Proposition 7. (2)

We analyze the difference between the two scenarios PW and PP in Developer B. For the shake of simplicity, we degenerate $C_j = 0$, *{j=A, B}* by set them at lower bounds. We also simplify $\gamma = \lambda$. Firstly, we solve the difference in

$$\Delta\pi_B^i = \pi_B^{PW} - \pi_B^{PP},$$

we get the result as follows:

$$\Delta\pi_B^i = \frac{(-5t + S(1 + \delta)(-1 + \rho))(35t + 9S(1 + \delta)(-1 + \rho)) +}{400t}$$

We see that the denominator is always positive, but we need to examine the numerator whether it is positive or negative. So, we derive the δ from the numerator, $\partial^2(-5t + S(1 + \delta)(-1 + \rho))(35t + 9S(1 + \delta)(-1 + \rho)) + 400t\lambda F_B/\partial\delta$. We take the limit values of δ into the numerator. We set $\delta = 0$ and

δ =1, and we find that when δ is equal to 0, the numerator is positive $(-5t+S(-1+\rho))(35t+9S(-1+\rho))+400t\lambda F_B$. And when δ is equal to 1, it is also positive $(-5t+2S(-1+\rho))(35t+18S(-1+\rho))+400t\lambda F_B$, as proven above. So, we can confirm that there is not exist a threshold of δ denoted by $0>\delta^{\dagger}\geq 1$. Above which $\pi_B^{\boldsymbol{PW}}-\pi_B^{\boldsymbol{PP}}<0 \Leftrightarrow \pi_B^{\boldsymbol{PW}}<\pi_B^{\boldsymbol{PP}}$. Based on the above properties, we solve the $\pi_B^{\boldsymbol{PW}}-\pi_B^{\boldsymbol{PP}}=0$ concerning δ. We get two roots as below:

Giving two roots $(i.e., -1+\frac{5(St(-1+\rho)+4\sqrt{S^2t(-1+\rho)^2(4t-9\lambda F_B)})}{9S^2(-1+\rho)^2})$.

We know that a root above exceeds one, thus

$$\delta^{\dagger}\equiv -1+\frac{5(St(-1+\rho)-4\sqrt{S^2t(-1+\rho)^2(4t-9\lambda F_B)})}{9S^2(-1+\rho)^2}.$$

(This page is intentionally left blank)