

Final Report - Grading Criteria

Team Absent Week 2 – Ken Carskadon & Steve Taniguchi

The following are the sections required for the Final Report. Attach a copy of this grading sheet to front of your report.

Points Earned	Points Possible	Category	Grading Notes
	5	Organization Follows same guidelines as the feasibility report	
	10	Spelling & Grammar Document is free of spelling and grammatical errors.	
	15	Management Summary	
	20	Current System	
		Proposed System	
	20	<ul style="list-style-type: none"> • Functional Requirements 	
	10	<ul style="list-style-type: none"> • Diagrams 	
	50	<ul style="list-style-type: none"> • Solution 	
	15	Recommendations	
	5	Appendix	
	15	Extra Credit	
	150	TOTAL (excluding extra credit) COMMENTS	

Research Project 2: System Requirements Document

Edmonds Community College – Computer Information Systems

Team Absent Week 2

Ken Carskadon
Steve Taniguchi

March 17th, 2016

Bank of Xanadu – Systems Requirements Document
March 17th, 2016

Team Absent Week 2

Ken Carskadon
Steve Taniguchi

Contents

Management Summary	5
Current System	6
Introduction	6
Project Stakeholders.....	6
Project constraints	7
Project Scope	7
Current Procedures	8
Proposed System.....	10
Functional requirements	10
Diagrams	12
Solution.....	16
Variants	17
Recommendation	18
Appendix	19

Management Summary

This report, the System Requirements Document, is the culmination of our analysis for Xanadu's new contract payables system. We have thoroughly reviewed the Accounting team's processes and procedures for tracking contract payables, and are ready to recommend a solution to replace them. In this document, you will find our completed analysis of the current system, our modeling of the requirements, and then finally, our new system proposal.

Our analysis started at the beginning of 2016. We began by examining the documents that were turned over to us, and then sitting down for an interview Patrick Jay. Based on this initial analysis, we proceeded to create our first report for Xanadu, the Feasibility study. In this report, we discussed our understanding of your existing processes for handling and manually processing contracts and invoices, as well as your expectations for the new system. We outlined the scope of the project, the strengths and weaknesses of the existing system, and the expected benefits of a new system. In the conclusion of this report, we gave our recommendation to move forward to the next step in our analysis, the creation of the System Requirements Document.

Once we determined that your project was indeed feasible, and that the Bank of Xanadu and Team Absent Week 2 were on the same page, we moved ahead with our analysis plans. This time, we refocused our investigation into examining how precisely your teams and systems work together. To complete this research, we created multiple diagrams to visualize and model your existing processes, how they all worked together in the existing system, and how the data flows from one entity to another. The insights gleaned from this analysis gave us a much better understanding of your exact needs and specifications, allowing us to present to you our recommendations for your new system.

The details of this research can be found all throughout this report. We start off with an introduction of sorts, and discuss how the current system is structured. Here we discuss the project's stakeholders, constraints, scope, and current procedures. We then move on to laying out our proposal for the new contract payables system. We begin by discussing the new system's requirements, as evidenced from our analysis. We then move on to a discussion of the diagrams we used to visualize and model your system and its needs. This leads to an overview of the solution we are proposing, including some variations, and their strengths and weaknesses. Finally, you will find the conclusion: Our recommendation for moving forward with this project.

Going forward, this System Requirements Document will serve as a blueprint of sorts for the next step in the process, the System Design phase. With this in mind, it is imperative that this document is not only accurate, but that it also fully encompasses all of your accounting system needs. That said, please don't hesitate to offer your input, or suggest any improvements.

Current System

Introduction

This Project Feasibility Study has been prepared for the Bank of Xanadu, in anticipation of your upcoming contract payables automation project. The proposed system will automate your current manual process for checking vendor invoices against contracts, as well as tracking and calculating contracted monies remaining.

The purpose of this document is to present our completed analysis of your existing systems, review the requirements we have compiled, and to present our recommendations for the new system. This study was produced by Team Absent Week 2, consisting of Ken Carskadon, and Steve Taniguchi. Our services were requested by Mr. Patrick Jay, Vice President and Manager of Accounting for Bank of Xanadu.

Project Stakeholders

The following is a list of stakeholders for the Bank of Xanadu automation project.

- **Sanjay Rupeedaal, CFO:** As the Chief Financial Officer, Mr. Rupeedaal is responsible for the financial well-being of the company. This means that any endeavor that benefits the company from a financial standpoint falls under the interest of Mr. Rupeedaal. Any success that the branch has in the project will result in streamlined process that will save manpower hours, and in the end improve the solvency of the company.
- **Anne Casey, Sr. Vice President:** Miss Casey resides over the Bellevue branch of the Bank of Xanadu, which will be the first branch to incorporate this new automated process. Success in this project will set her branch apart as a pioneer of company improvement, and will affect her career positively. This improvement of standing in the company gives her a personal vested interested in the success of this project.
- **Patrick Jay, VP of Accounting:** Mr. Jay has a direct connection to this project; being one of the individuals that we interviewed to understand the current process and request that has been made. His employees are currently responsible for the workload that is created with the current system. This means that improving that system directly affects the productivity of his employees, and on a larger scope, his department.

Project constraints

In addition to the general system requirements, there are additional limitations or restrictions for this project that we call constraints. These constraints can be thought of as conditions that must be met for the new system to be considered a success. For this project, the constraints are as follows:

- The system must meet the basic requirements set forth in the original request.
- The system must be able to provide reporting as needed.
- The total system implementation costs must fall within the estimate.
- The system must be in place by the specified due date.
- The system must be expandable from the initial pilot.

Identifying these constraints helps to clearly define the scope of the project. It is imperative that each of these constraints be satisfied in the proposed system.

Project Scope

Understanding and remaining within the scope of this project is a key component to its success. The purpose of this project is to automate the processing and verification of contracts and invoices for external programming resources, and provide the bank with a streamlined way to enter information for database purposes. With this in mind, we are considering only the data that currently flows in and out of the existing contract payables tracking system as in scope. This does not include any interactions that occur outside of the existing system, such as sending the information in to accounting, nor will it encompass any other portion of the process outside of what has been explicitly defined in this report.

Current Procedures

The Bank of Xanadu has an existing process which is currently implemented for verification and processing. This is currently done manually by an employee in the bank. Given the relatively complex nature of this process it is necessary to outline it in a flow chart which you can see below (see figure 1: Contract processing flowchart).

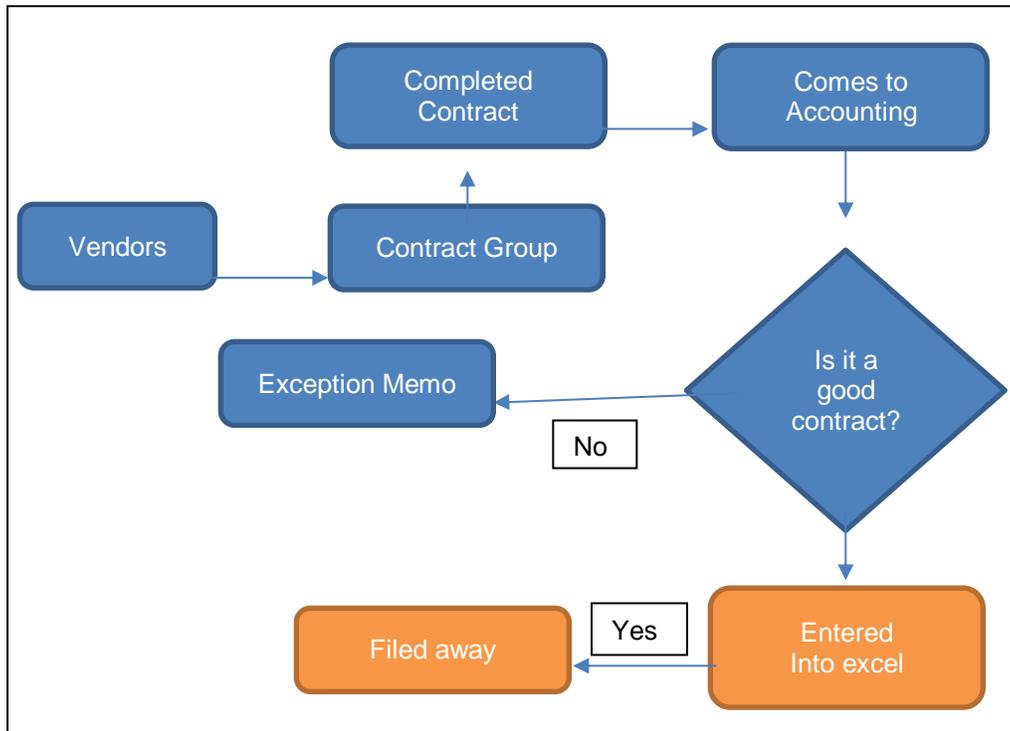


Figure 1

The figure above shows us the process currently implemented. The portion of the project that is highlighted in green is the process which the Bank of Xanadu has requested to be automated, and the focus of this project will be on those sections. This flowchart displays the work done manually by the employee within the company. It is important to note that there is also a process implemented by the company on a larger scale which is shown in figure 2 below.

In the invoice processing flowchart (figure 2), you can see the process that occurs within the company when an invoice request is received from a vendor. This encompasses the entire process, and much of this is out of scope, but it is important to display the process in its entirety for a complete understanding of this project.

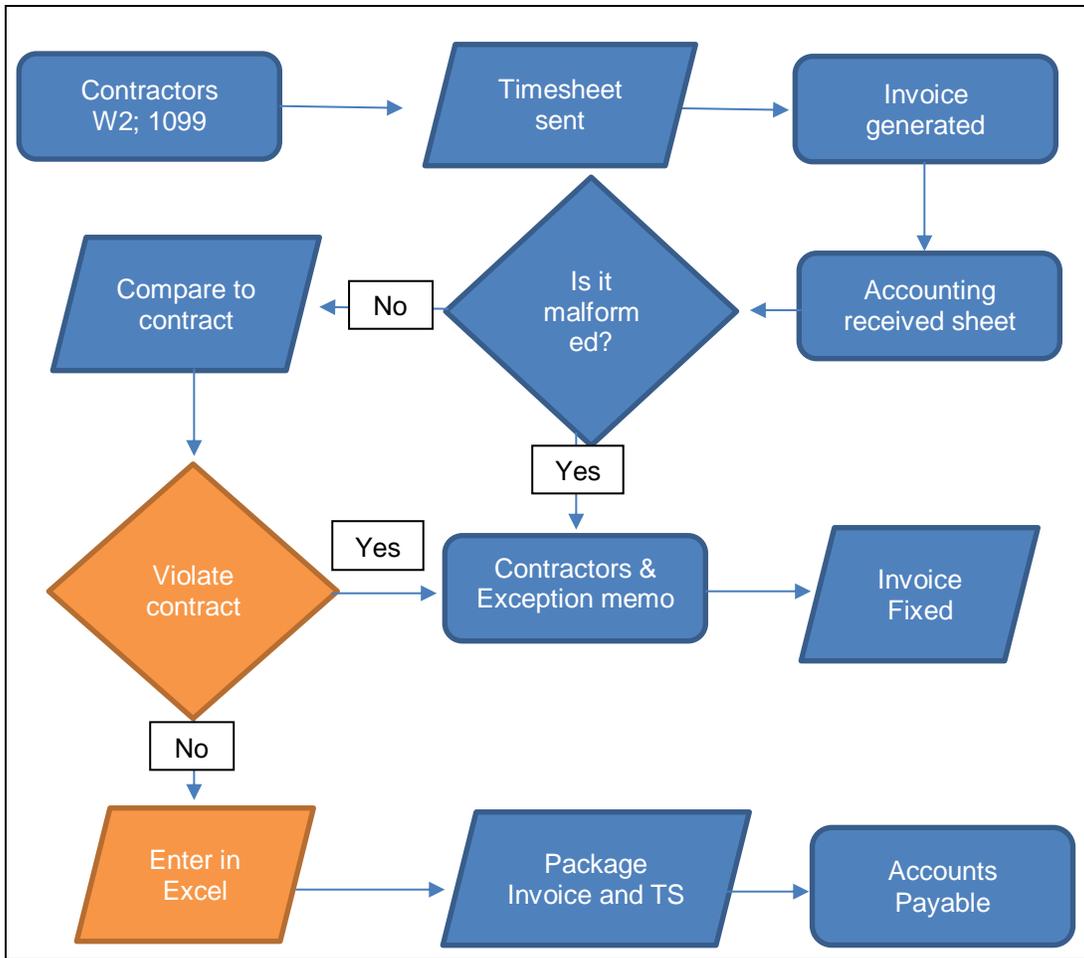


Figure 2

Proposed System

Functional requirements

In our analysis of your processes and requirements, we have identified nine distinct procedures that comprise the contract payables system: three for processing contracts, and six for processing invoices.

- **Contracts**
 - Receive contracts – ID UC001
 - Contract exceptions – ID UC002
 - Update contracts – ID UC003

- **Invoices**
 - Receive invoices – ID UC004
 - Invoice exceptions - ID UC005
 - Update invoices - ID UC006
 - Pay invoices - ID UC007
 - Vendor invoice inquiries - ID UC008
 - Invoice accrual - ID UC009

In order to fully understand these procedures, we compiled a small report for each called a Use Case Scenario. Each of these scenarios contains the following information:

- Use case name (procedure)
- Use case ID
- Primary actor (main player for procedure)
- Brief description of procedure
- Trigger (action that starts procedure)
- Main flow (normal steps for procedure)
- Alternative flow (any alternative steps for the main flow)
- Exception flow (any exceptions to the main flow)
- Preconditions
- Post-conditions
- Information requirements
- Assumptions
- Business Rules

Figure 3 is an example of the Use Case Scenario we have created for receiving invoices (additional use case scenarios can be found in the Appendix):

USE CASE NAME:	Receive Invoice	ID: UC004
Primary Actor:	Accountant	
Brief Description:	This use case describes the steps from the time the accountant receives the invoice to the point where the invoice has been filed for payment.	
Trigger:	A new invoice is received by the accountant.	
Main flow:	<p>This use case begins when the accountant received a new invoice from the buyer.</p> <ol style="list-style-type: none"> 1. The accountant visually verifies that the invoice has a timesheet attached to it. 2. The accountant visually inspects the invoice to confirm the invoice falls within the contract time limits. 3. The accountant confirms that the invoice’s hourly rates match the hourly rates stipulated in the contract. 4. The accountant verifies that there is money available to pay the invoice. 5. The accountant generates a data entry sheet and attaches it to the original invoice. 6. The accountant opens the excel spreadsheet used for tracking invoice information. 7. The accountant validates the invoice; signifying that the information is correct and the invoice have been entered into the system. 8. The accountant sends the invoice with the data entry sheet to accounts payable to be paid. 	
Alternate flows:	<ol style="list-style-type: none"> 1. If there is a missing timesheet an exception will be written for the invoice (UC005). 1. If the invoice sent is an update to a pre-existing invoice the accountant will complete the Invoice Update use case (UC006). 5. Once entered the invoice will be paid out to the buyer (UC007). 5. If the invoice is an accrual and not a payment the invoice will be accrued on the final contract (UC009). 	
Exception flows:	<ol style="list-style-type: none"> 1. If the invoice is missing the timesheet it will be returned the buyer to be resubmitted. 2. If the invoice does not fall within the contract time limitations the accountant will generate an exception memo (UC005). 3. If the invoice’s hourly rates do not match the rates stipulated on the contract the accountant will file an exception memo, and the buyer will need to be notified 	

	(out of scope).
Preconditions:	A new invoice has been received and is ready to be verified and filed for payment.
Post-conditions:	The invoice has been verified, entered into the system, validated, and filed for payment.
Information Requirements:	<ul style="list-style-type: none"> Name of the buyer/vendor Hours being billed in invoice Hourly rate of billed hours Contract start date Contract end date Contract hourly rate Current available funds status
Assumptions:	<ol style="list-style-type: none"> 1. The buyer will submit a complete and valid invoice. 2. The invoice is validated and entered into the system before any payment is issued.
Business Rules:	<ol style="list-style-type: none"> 1. The buyer must submit a complete and valid invoice to the accounting department. 2. Invoices are to be validated and filed before accrual or payment is made. 3. Each invoice must have an identification number assigned to it during verification. 4. Any change in an invoice requires a new invoice to be submitting to the accounting department.

Figure 3

Diagrams

During our analysis, we create different types of diagrams to help visualize how your systems work together. These diagrams allow us to model the way data flows in and out of the system, determine who the main users are, and make sure we have all of our bases covered. Here are a few examples of the diagrams we have created for your new system, and how they have helped us create your solution.

Use Case Diagram

The Use Case Diagram is an important tool to help us visualize the system. It shows all of the processes that compose the system, as well as the people who interact with it, known as actors. Our analysis shows only 1 actor for the system – the accounting team. In figure 4, we have modeled how the processes all relate to each other, and how and where the users will interact with them.

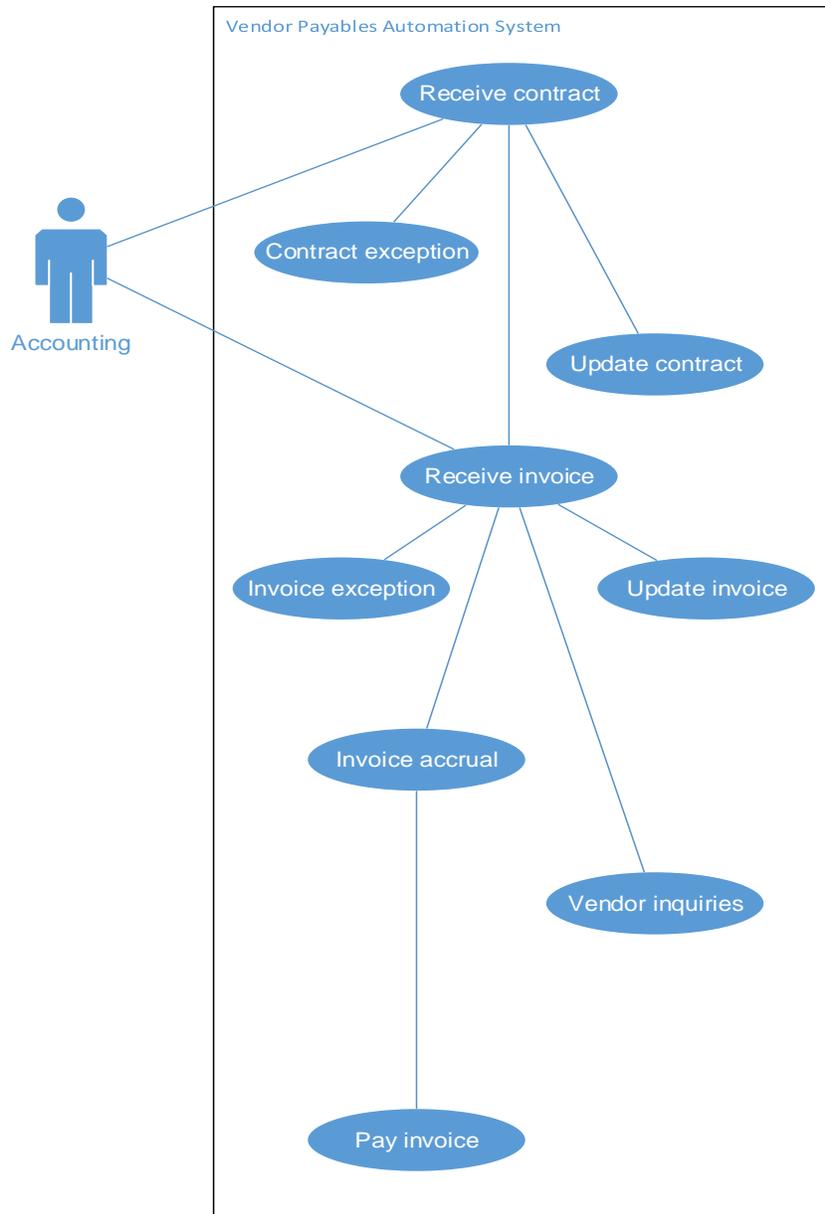


Figure 4

Functional Decomposition Diagram (FDD)

The FDD is another helpful analysis tool. This diagram shows all of the steps for each of the different procedures that combine to make your process. Each of these procedures has a Use Case Scenario (see above) that explains the start and end points, the normal steps required, and any exception or alternative steps. We have included an example of the FDD for receiving invoices below (figure 5). For the entire FDD, please see the Appendix.

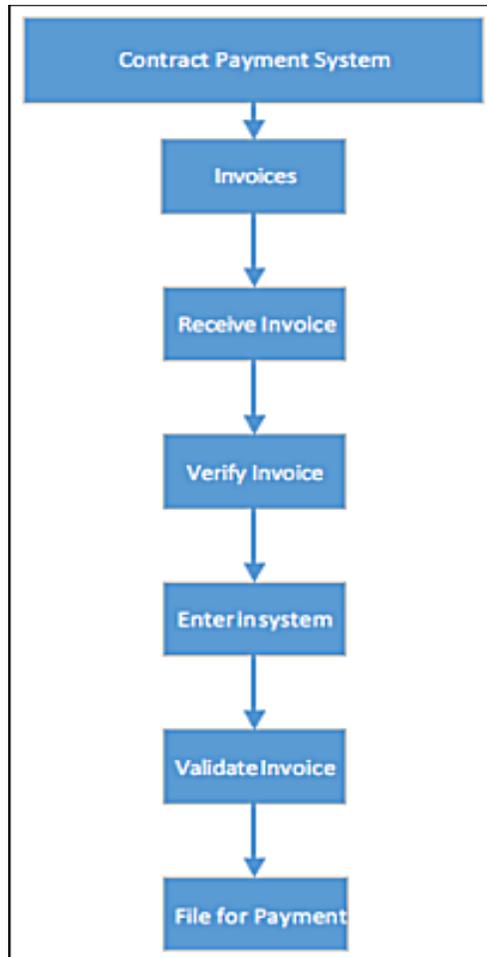


Figure 5

Context Data Flow Diagram (Context DFD)

To help visualize data moving in and out of the new system, we have created what is known as a Content Data Flow Diagram or DFD. This diagram takes a top-level look at how the data flows, and gives us a visual model to help with analysis. In the diagram below, Process 0 represents the accounting contract payables system. As you can see in figure 6 below, data flows to and from the accounting system and external entities (vendors, buyers in the Contract group, and Accounts Payable).

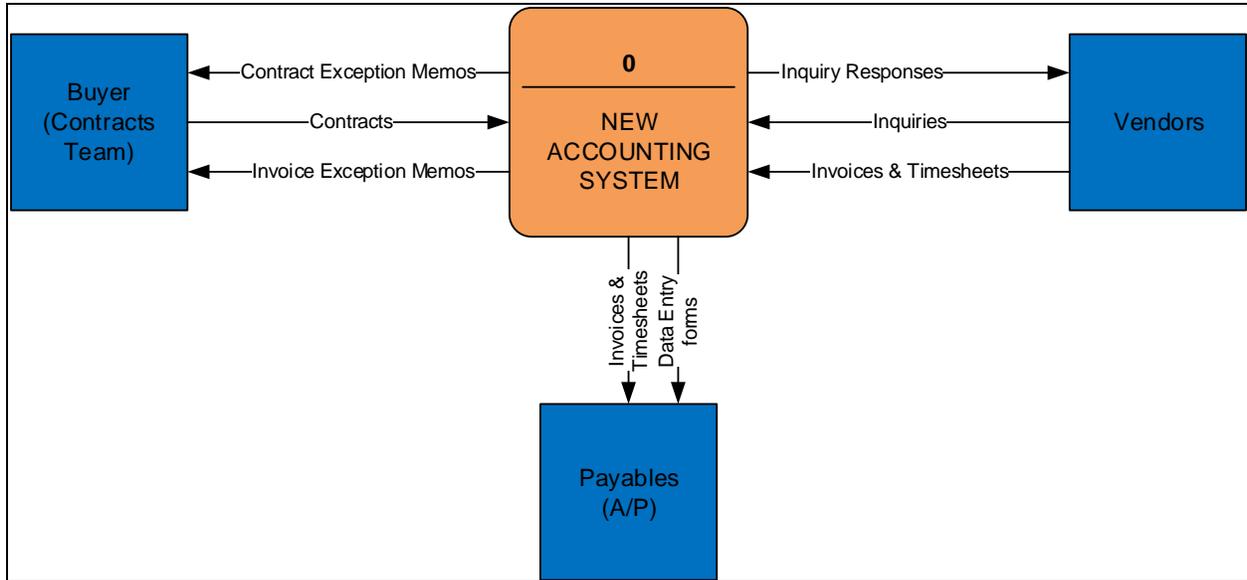


Figure 6

This diagram is also helpful in defining scope. Everything found on the DFD is in scope for this project. Likewise, anything missing from the diagram, like vendors working with a buyer to build a contract, would be out of scope.

Solution

Our proposal is to build a new modular payables tracking system that stores contract and invoice information in an industry-standard database server. We see this system as being relegated to the Accounting team, with access also available for certain managers (as needed). This system would open up the processing of contracts and invoices to the entire accounting team, create multiple backups on the team (in case anyone is sick or on vacation), and would provide a massive time savings over the existing all manual system. In addition, the system would reduce the opportunities for human error, and provide more secure data storage on a server with regular backups.

As we showed in figure 6, data needs to flow between the accounting system and various internal and external groups. We have no intention of changing this flow; the new automated system will handle the vast majority of this data. Accounting will visually check each document, enter all contract and invoice information into the system, and then file away the hard copies of these documents. The system will automatically pull contract data during invoice entry, perform contract comparisons for work dates and vendor/contractor data, calculate the total amount due (hours worked * hourly rate) and the total monies paid on the contract, and track accruals. It will also automatically generate exception memos for contracts and invoices, data entry forms for payables, and reporting. The same data will continue to flow between the same entities; it will just be processed quicker and more accurately thanks to automation.

Locking down system access to Accounting and managers ensures that external entities (vendors, buyers on the contract team, payables, etc.) can keep their existing processes and procedures unchanged, and remain BAU. By adopting this narrow focus, we are able to keep this solution simple. This means less complexity for the project, ensuring that Bank of Xanadu will see a cost savings versus an overbuilt one-size-fits-all solution. Removing unnecessary complication allows for a quicker setup and turn-around for this project, with fewer documents to create, and fewer employees to train.

The new system will also offer 24/7 access to reporting, with more robust options. Instead pouring through the spreadsheet to find the data you need, reporting can be captured from the database server with just a click or two - even by a manager. While we were admittedly unable to discern much reporting information from your documents, we can create any number of report requests, with as much or as little data as you need, that can be pulled at a moment's notice. Additional reporting requests can be created easily in the future by using industry-standard SQL queries.

Variants

This system proposal will address each of Accounting's pain points in the existing system, increase their productivity, and reduce opportunities for human error. By using an industry-standard database server, it will also expand access to data and reporting, improve the backup situation, and allow for expansion. It focuses on simplicity. We aimed to improve the current process, while keeping external entities business as usual (BAU). There are alternative options, however, that could provide additional time and money savings, at the cost of increased complexity.

One such possibility is to have accounting run all documents through a scanner, instead of entering contract and invoice data into the system manually. Using Optional Character Recognition (OCR), the data from the scanned invoices could be automatically loaded into the database, saving additional time for accounting, and further reducing the chance of human error. On the flip side, there would be an increase in costs associated with this solution. Not only would there be additional equipment and training required for the accounting team, the scanners could require document changes for contracts, timesheets, and invoices.

Another option for the system would be to move to electronic correspondence exclusively for contracts, invoices, and timesheets. This variant would produce additional time savings and a cost reduction in office supplies (pens, paper, folders, files, etc.). It would also, however, require changes for buyers, vendors, and payables, meaning increased costs – at least initially.

Recommendation

After months of analysis, and careful consideration, Team Absent Week 2 is recommending that we move forward to the system design phase of this project. In this report, we have proven that we possess an in-depth understanding of Xanadu's existing systems and requirements, and have included our modeling all throughout. We are also presenting a modular database-centric system that will not only meet your current requirements for automation and reporting, but also allow for expansion down the road. Although this system would mean big changes for the accounting team, it is a solution that focuses on simplicity, enabling other external and internal groups doing business with accounting can remain BAU.

This approach has key efficiencies that other solutions just can't offer. Basing the solution around an industry-standard database server means that we can have a pilot up and running quickly, with minimal costs for setup and integration. This system can also be customized however you see fit, and expanded upon as desired. Reporting requests can be added or changed using standard SQL queries, and scanners could be added if you wanted to try going digital with all of your documents.

We have previously proven this project to be feasible operationally, technically, and economically, and that it can be expected to produce a number of tangible and intangible benefits. We have also shown how the new system would pay for itself (and then some) before the first year of operation is over. At this time, we see no reason not to move forward.

Our next step will be to transition to what we refer to as the System Design phase, and begin working on the physical design to meet the specifications laid out in this report. Since we will be referencing this System Requirements Document as a blueprint of sorts, it is of the utmost importance that everything is accurate and accounted for. Therefore, if you have any suggestions or corrections for this report, please don't hesitate to reach out to us.

We at Team Absent Week 2 have enjoyed working with you all at Bank of Xanadu, and are excited to move forward with this project. Thank you for your business. We look forward to discussing this more in-depth at our next meeting in April.

Appendix

Use Case Scenario ID UC001 - Receive contracts

USE CASE NAME:	Receive Contract	ID: UC001
Primary Actor:	Accounting	
Brief Description:	This use case describes the process from the time the accountant receives the contract until the contract is accepted and entered into the system.	
Trigger:	Accounting receives a contract from the buyer	
Main flow	<p>This use case begins when the accountant receives a contract from the buyer:</p> <ol style="list-style-type: none"> 1. Accountant examines the contract for required information. 2. The accountant logs onto the system. 3. Accountant navigates to the spreadsheet where the contract information is entered. 4. Accountant assigns the contract number use by the accounting department. 5. Accountant selects the appropriate vendor for the contract. 6. Accountant selects the appropriate project manager for the contract. 7. Accountant selects the correct charge unit for the contract. 8. Accountant selects the appropriate bank division for the contract. 9. Accountant enters the programmer's name. 10. Accountant enters the project start and end dates. 11. Accountant enters hourly pay rate agreed upon in the contract. 12. Accountant enters the fee maximum amount from the contract. 13. Accountant enters the project description. 	
Alternate flows	<ol style="list-style-type: none"> 1. If the vendor on the contract is not already in the system the accountant will add the vendor into the system. 	
Exception flows	<ol style="list-style-type: none"> 1. Information is missing from the contract that cannot be looked up, and the accountant generates an exception memo (UC005). 	
Preconditions	A new contract has been received from the buyer.	
Post-conditions	The received contract has been verified and entered into the system for future reference.	

<p>Information Requirements</p>	<p>Contract Number Vendor Name Project Manager Name Charge Unit Bank Division Programmer’s Name Start Date of contract End Date of contract Hourly Pay Rate Fee Maximum Project Description</p>
<p>Assumptions</p>	<ol style="list-style-type: none"> 1. The buyer will submit a complete and valid contract. 2. The contract will be validated by accounting before it is entered into the system.
<p>Business Rules</p>	<ol style="list-style-type: none"> 1. The buyer must complete and submit a valid contract to the accounting department. 2. The contract is to be validated before it is entered by accounting. 3. Accounting will enter all the contract information into the system for reference. 4. Each contract must have an ID assigned to it by accounting when entered into the system.

Use Case Scenario ID UC002 - Contract exceptions

USE CASE NAME:	Contract Exception	ID: UC002
Primary Actor:	Accounting	
Brief Description:	This use case describes the process when a contract is received that cannot be processed, and must be returned to the buyer.	
Trigger:	Accounting determines a contract is mal-formed.	
Main flow	<p>This use case begins when the accountant determines that a contract is mal-formed:</p> <ul style="list-style-type: none"> 14. Accountant drafts and exception memo. 15. Accountant indicates any missing or incorrect information by selecting the options on the exception memo table. 16. Accountant returns contract to buyer with the exception memo attached. 17. Accountant receives revised contract from the buyer. 18. Accountant examines contract; verifying that the corrections have been made and the contract is now well-formed. 	
Alternate flows	2. If the contract is not corrected the account will go on hold until the buyer contacts accounting.	
Exception flows	2. If the corrected contract is received and still has errors on it the accountant will repeat the Contract Exception (UC002) process.	
Preconditions	A contract received from the buyer is determined to be mal-formed.	
Post-conditions	The contract is sent back to the buyer with an exception memo detailing the changes that need to be made.	
Information Requirements	<ul style="list-style-type: none"> Contractor Name Master Agreement Number Project Number Charge Unit Project Manager Total Fee Hourly Rate Contractor Signature Bank Manager Signature 	

Assumptions	<ol style="list-style-type: none">3. The buyer will submit a well-formed contract to accounting to be processed.4. All corrections needed on a mal-formed contract will be made in the first revision received from accounting.
Business Rules	<ol style="list-style-type: none">5. The buyer must complete and submit a valid contract to the accounting department.6. The contract is to be validated before it is entered by accounting.7. Accounting will enter all the contract information into the system for reference.8. Each contract must have an ID assigned to it by accounting when entered into the system.

Use Case Scenario ID UC003 - Update Contracts

USE CASE NAME:	Update Contract	ID: UC003
Primary Actor:	Accounting	
Brief Description:	This use case describes the process from the time the accountant receives a corrected contract until it is entered into the system.	
Trigger:	Accounting receives a contract from the buyer that has previously been returned for missing or invalid information.	
Main flow	<p>This use case begins when the accountant receives an updated contract from the buyer:</p> <ol style="list-style-type: none"> 19. Accountant reviews exception memo to determine what information was missing or incorrect. 20. Accountant examines the contract for updated information. 21. Accountant verifies that information submitted in contract meets requirements described in the exception memo. 22. Accountant examines contract for missing or incomplete information not described in exception memo. 23. Accountant logs into the system. 24. Accountant navigates to spreadsheet where contract information is entered. 25. Accountant finds original contract information in the system. 26. Accountant updates contract in system to include all missing or incorrect information in the updated contract. 	
Alternate flows	<ol style="list-style-type: none"> 3. If the contract was not entered into the system the first time it was received the accountant will need to complete all steps in the receive contract scenario (UC001). 4. If the vendor for the contract is not in the system the accountant will need to enter their information. 	
Exception flows	<ol style="list-style-type: none"> 3. Information is missing from the contract that cannot be looked up, and the accountant generates an exception memo (UC005). 	
Preconditions	An updated contract has been received from the buyer.	
Post-conditions	The received contract has been verified and entered into the system for future reference.	
Information Requirements	<p>Contract Number Vendor Name Project Manager Name</p>	

	Charge Unit Bank Division Programmer's Name Start Date of contract End Date of contract Hourly Pay Rate Fee Maximum Project Description
Assumptions	<ol style="list-style-type: none">5. The buyer will submit a contract with all of the requested updates and information from the exception memo.6. The contract will be validated by accounting before it is entered into the system.
Business Rules	<ol style="list-style-type: none">9. The buyer must complete and submit a valid contract to the accounting department.10. The contract is to be validated before it is entered by accounting.11. Accounting will enter all the contract information into the system for reference.12. Each contract must have an ID assigned to it by accounting when entered into the system.

Use Case Scenario ID UC004 - Receive invoices

USE CASE NAME:	Receive Invoice	ID: UC004
Primary Actor:	Accountant	
Brief Description:	This use case describes the steps from the time the accountant receives the invoice to the point where the invoice has been filed for payment.	
Trigger:	A new invoice is received by the accountant.	
Main flow:	<p>This use case begins when the accountant received a new invoice from the buyer.</p> <ol style="list-style-type: none"> 9. The accountant visually verifies that the invoice has a timesheet attached to it. 10. The accountant visually inspects the invoice to confirm the invoice falls within the contract time limits. 11. The accountant confirms that the invoice's hourly rates match the hourly rates stipulated in the contract. 12. The accountant verifies that there is money available to pay the invoice. 13. The accountant generates a data entry sheet and attaches it to the original invoice. 14. The accountant opens the excel spreadsheet used for tracking invoice information. 15. The accountant validates the invoice; signifying that the information is correct and the invoice have been entered into the system. 16. The accountant sends the invoice with the data entry sheet to accounts payable to be paid. 	
Alternate flows:	<ol style="list-style-type: none"> 2. If there is a missing timesheet an exception will be written for the invoice (UC005). 6. If the invoice sent is an update to a pre-existing invoice the accountant will complete the Invoice Update use case (UC006). 10. Once entered the invoice will be paid out to the buyer (UC007). 6. If the invoice is an accrual and not a payment the invoice will be accrued on the final contract (UC009). 	
Exception flows:	<ol style="list-style-type: none"> 4. If the invoice is missing the timesheet it will be returned the buyer to be resubmitted. 5. If the invoice does not fall within the contract time limitations the accountant will generate an exception memo (UC005). 6. If the invoice's hourly rates do not match the rates stipulated on the contract the accountant will file an 	

	exception memo, and the buyer will need to be notified (out of scope).
Preconditions:	A new invoice has been received and is ready to be verified and filed for payment.
Post-conditions:	The invoice has been verified, entered into the system, validated, and filed for payment.
Information Requirements:	<ul style="list-style-type: none"> Name of the buyer/vendor Hours being billed in invoice Hourly rate of billed hours Contract start date Contract end date Contract hourly rate Current available funds status
Assumptions:	<ul style="list-style-type: none"> 3. The buyer will submit a complete and valid invoice. 4. The invoice is validated and entered into the system before any payment is issued.
Business Rules:	<ul style="list-style-type: none"> 5. The buyer must submit a complete and valid invoice to the accounting department. 6. Invoices are to be validated and filed before accrual or payment is made. 7. Each invoice must have an identification number assigned to it during verification. 8. Any change in an invoice requires a new invoice to be submitting to the accounting department.

Use Case Scenario ID UC005 - Invoice exceptions

USE CASE NAME:	Invoice Exception	ID: UC005
Primary Actor:	Accounting	
Brief Description:	This use case describes the process when an invoice is received that cannot be processed, and must be returned to the vendor.	
Trigger:	Accounting determines an invoice is mal-formed.	
Main flow	<p>This use case begins when the accountant determines that an invoice is mal-formed:</p> <ol style="list-style-type: none"> 27. Accountant drafts an invoice exception memo. 28. Accountant indicates any missing or incorrect information by selecting the options on the exception memo table. 29. Accountant returns invoice to vendor with the exception memo attached. 30. Accountant receives revised invoice from the buyer. 31. Accountant examines invoice; verifying that the corrections have been made and the invoice is now well-formed. 	
Alternate flows	5. If the invoice is not corrected the account will go on hold until the vendor contacts accounting.	
Exception flows	4. If the corrected invoice is received and still has errors on it the accountant will repeat the Invoice Exception (UC005) process.	
Preconditions	An invoice received from the vendor is determined to be mal-formed.	
Post-conditions	The invoice is sent back to the vendor with an exception memo detailing the changes that need to be made.	
Information Requirements	<p>Name of the buyer/vendor Hours being billed in invoice Hourly rate of billed hours Contract start date Contract end date Contract hourly rate Current available funds status Bank Manager Signature</p>	
Assumptions	<ol style="list-style-type: none"> 7. The vendor will submit a well-formed invoice to accounting to be processed. 8. All corrections needed on a mal-formed invoice will be made in the first revision received from accounting. 	

Business Rules	<ol style="list-style-type: none">13. The vendor must complete and submit a valid invoice to the accounting department.14. The invoice is to be validated before it is entered by accounting.15. Accounting will enter all the invoice information into the system for reference.16. Each invoice must have a contract associated with it by accounting when entered into the system.
-----------------------	--

Use Case Scenario ID UC006 - Update invoices

USE CASE NAME:	Update Invoice	ID: UC006
Primary Actor:	Accounting	
Brief Description:	This use case describes the process from the time the accountant receives an updated invoice to the time that the invoice is entered into the system and filed for accounts payable.	
Trigger:	Accounting receives an updated invoice.	
Main flow	<p>This use case begins when the accountant receives an updated invoice from the vendor:</p> <ul style="list-style-type: none"> 32. Accountant verifies that the timesheet is attached to the invoice. 33. Accountant verifies that the hours on the timesheet match the information shown on the invoice. 34. Accountant logs onto the system. 35. Accountant navigates to where invoices are entered into the system. 36. Accountant selects the correct vendor for the invoice. 37. Accountant selects the correct contractor for the invoice. 38. Accountant compares the information on the exception memo to the information on the updated invoice. 39. Accountant updates any information required to satisfy the request from the exception memo. 40. Accountant verifies that all information is correct and entered into the system. 41. Accountant sends invoice through validation process. 42. Accountant marks the invoice as ready for payment. 43. Accountant files invoice and sends it to accounts payable. 	
Alternate flows	6. If the accountant is unable to locate the exception memo they will verify everything on the invoice with the contract.	
Exception flows	5. Information is missing after the update has been sent another exception memo will be generated (UC005), and then invoice will be sent back to the vendor for updates.	
Preconditions	An updated invoice from the vendor is sent to accounting.	
Post-conditions	The received invoice is verified updated in the system, and sent to accounts payable.	

<p>Information Requirements</p>	<p>Contract Number Vendor Name Hourly Pay Rate Billable Hours</p>
<p>Assumptions</p>	<p>9. The vendor will submit a complete, valid, updated invoice to accounting. 10. The invoice will be verified before it is entered into the system. 11. The invoice will be validated and entered into the system before it is sent to accounts payable.</p>
<p>Business Rules</p>	<p>17. The vendor must complete and submit a complete invoice. 18. The invoice is to be validated before it is entered by accounting. 19. Accounting will enter all the invoice information into the system to be sent to accounts payable. 20. Every invoice will go through the validation process before it is sent to accounts payable.</p>

Use Case Scenario ID UC007 - Pay invoices

USE CASE NAME:	Pay Invoice	ID: UC007
Primary Actor:	Accounting	
Brief Description:	This use case describes the process when an invoice has been processed and needs to be paid out to the vendor.	
Trigger:	The Contract and Invoice have been verified and entered into the system.	
Main flow	<p>This use case begins when the accountant starts the process of paying out an invoice.</p> <p>44. Accountant creates a data entry sheet to send to accounts payable.</p> <p>45. Accountant enters all information regarding the invoice into the data entry sheet.</p> <p>46. Accountant attaches data entry sheet to a copy of the invoice.</p> <p>47. Accountant sends the invoice and data entry sheet to accounts payable (AP) to be processed.</p>	
Alternate flows		
Exception flows	6. If the data entry sheet has missing or incorrect information it will be sent back to accounting to be fixed.	
Preconditions	An invoice and contract have been verified and entered into the system.	
Post-conditions	An invoice with a data entry sheet is sent to accounts payable to be processed.	
Information Requirements	<p>The name of the Vendor</p> <p>Vendor number</p> <p>Invoice number</p> <p>Description</p> <p>Invoice date</p> <p>Due date</p> <p>Invoice total</p> <p>General Ledger account number</p> <p>P.O. Number</p> <p>Charge unit</p>	
Assumptions	<p>12. The vendor will submit a well-formed invoice to accounting to be processed.</p> <p>13. The accountant will verify the contract and invoice before sending it to accounts payable.</p> <p>14. The information on the data entry sheet will be correct and complete when sent to AP.</p>	

Business Rules	<ol style="list-style-type: none">21. The vendor must complete and submit a valid invoice to the accounting department.22. The contract must be verified and entered into the system before the invoice can be processed.23. The invoice is to be validated before it is entered by accounting.24. Accounting will enter all the invoice information into the system for reference.25. Each invoice must have a contract associated with it by accounting when entered into the system.26. Each invoice must have an attached data entry sheet before it is sent to AP for processing.

Use Case Scenario ID UC008 - Vendor invoice inquiries

USE CASE NAME:	Vendor Inquiry	ID: UC008
Primary Actor:	Accounting	
Brief Description:	This use case describes the process when accounting received an inquiry from the vendor.	
Trigger:	An inquiry from the vendor is received by accounting	
Main flow	<p>This use case begins when the accountant receives an inquiry from the vendor.</p> <p>48. Accountant reviews information received from the vendor regarding their inquiry.</p> <p>49. Accountant verifies that information provided by vendor matches the contract information.</p> <p>50. Accountant investigates the invoice that the inquiry is related to.</p> <p>51. Accountant informs the vendor of information related to their inquiry.</p> <p>52. Accountant provides vendor with instructions on further action that is required or recommended.</p>	
Alternate flows	<ol style="list-style-type: none"> 1. If the inquiry is regarding an exception memo the accountant will inform them of what needs to be corrected to continue. 2. If the invoice is filed as an accrual the accountant informs the vendor of this information, and tells them that payment will occur in the future. 	
Exception flows	<ol style="list-style-type: none"> 7. If the accountant is unable to locate an invoice in the system related to the vendor's inquiry they will request to receive a new invoice from the vendor. 	
Preconditions	An inquiry is received through phone, mail, or email from a vendor regarding an invoice.	
Post-conditions	The vendor is informed of the status of the invoice, and given additional directions on how to proceed.	
Information Requirements	<p>Vendor Name</p> <p>Contact Information</p> <p>Relevant invoice ID</p> <p>Description of inquiry</p>	
Assumptions	<ol style="list-style-type: none"> 15. The vendor will submit the required information before discussing the inquiry with accounting. 16. The accountant will verify the information provided before discussing the inquiry with the vendor. 	

Business Rules	<ol style="list-style-type: none">27. The vendor must complete and submit a valid invoice to the accounting department.28. The vendor must provide correct and complete information for any inquiry made regarding an invoice.29. The invoice information must be verified before any correspondence occurs.30. Accounting will log the inquiry in their system after the issue is resolved.
-----------------------	---

Use Case Scenario ID UC009 - Invoice accrual

USE CASE NAME:	Invoice Accrual	ID: UC009
Primary Actor:	Accounting	
Brief Description:	This use case describes the process when accounting processed an accrual from a received invoice.	
Trigger:	Accounting received an invoice before the pay period date.	
Main flow	<p>This use case begins when the accountant receives an invoice that needs to be submitted as an accrual.</p> <p>53. Accountant verified information on the invoice. 54. Accountant compares invoice to contract information for accuracy. 55. Accountant navigates to their accruals excel workbook. 56. Accountant enters invoice information into the workbook. 57. Accountant reverses entries in the workbook when they are paid by accounts payable.</p>	
Alternate flows	3. If the invoice received is a corrected document (exception memo attached) the accountant will verify the information is correct before proceeding.	
Exception flows	8. If the invoice has missing or incorrect information an invoice exception memo will be filled out (UC005).	
Preconditions	An invoice is received from the vendor between pay periods.	
Post-conditions	The invoice is filed into the accruals log and sent to accounts payable for future payment.	
Information Requirements	Programmer Name Vendor Name Charge Unit Invoice Number Total Invoice Amount Accrual Date Reversed Date (When invoice is reversed).	
Assumptions	17. The vendor will submit a valid invoice to accounting. 18. The accountant will verify the information on the invoice before entering it into the accrual log.	
Business Rules	31. The vendor must complete and submit a valid invoice to the accounting department. 32. The invoice information must be verified before it is entered into the system. 33. The accountant must send the invoice to Accounts	

	<p>Payable after it has been processed and logged. 34. Accounting will reverse the accrual in the log after Accounts Payable has paid out the invoice.</p>
--	--

