American International University-Bangladesh (AIUB)

Department of Computer Science

Faculty of Science &Technology (FST)

Summer_20-21

Section: A

Software Quality Assurance and Testing

PROJECT TITLE: Zoo Food Maintenance System

A Report submitted By: **ArrowCross** {Group Name}

Project Collaborators:

SN	Student Name	Student ID
1	EMON, RASEL IQBAL [***Group Leader***]	16-32332-2
2	ALAM SHAHNAZ	17-35531-3
3	JAHAN, SYEDA ISRAT	18-36488-1
4	PARVIN MAHNAJ	18-36801-1

Under the supervision of

ABHIJIT BHOWMIK

Designation: Associate Professor

Software Test Plan

for

< *Zoo Food Maintenance System* >

Version 1.0 approved

Prepared by: < **ArrowCross** >

American International University-Bangladesh (AIUB)

< 15-AUGUST-2021 >

Table of Contents

RevisioN HistorY

Revision	Date	Updated by	Updated Comments	Draft Sequences
0.1	2021.08.**01**	ALAM SHAHNAZ	Startup of Project Report	Draft - 1
0.2	2021.08.**02**	ALAM SHAHNAZ	Covered Page + Project Title	Draft - 2
0.3	2021.08.**02**	EMON, RASEL IQBAL	Project Title Changed + Finalized	Draft - 3
0.4	2021.08.**03**	EMON, RASEL IQBAL	Introduction [**3**] + System Features List [**4.1**]	Draft – 4
0.5	2021.08.**08**	JAHAN, SYEDA ISRAT	System Quality Attributes [**4.2**]	Draft – 5
0.6	2021.08.**08**	PARVIN MAHNAJ	System Interface [**4.3**]	Draft – 6
0.7	2021.08.**10**	EMON, RASEL IQBAL	Project Requirements [**4.4**]	Draft – 7
0.8	2021.08.**11**	ALAM SHAHNAZ	Features Not to be Tested [**5**]	Draft – 8
0.9	2021.08.**12**	ALAM SHAHNAZ	Testing Levels [**6.1**] + Testing Tools [**6.2**]	Draft – 9
0.10	2021.08.**14**	EMON, RASEL IQBAL	Meetings [**6.3**] + Test Case Updated [**7**]	Draft - 10
1.0	2021.08.**15**	EMON, RASEL IQBAL	Ultimate Check Done and Submitted	**FinaL Draft**

1. TEST PLAN IDENTIFIER: *ZFMS_V1.0*

2. REFERENCES

- Any reference documents……..
- System Interface: **draw.io** [https://app.diagrams.net/?libs=general;flowchart]
- Software Automation Testing Tool: **Selenium** [https://www.selenium.dev/]
- Test Case Online: **LambdaTest** [https://www.lambdatest.com/automation-testing]
- Testing Scheduling: **ClickUp** [https://sharing.clickup.com/l/h/6w311-64]

3. INTRODUCTION

3.1 Background to the Problem

As being the best of all creations by the Almighty, it is a general responsibility for each human being to protect all the essential elements or helpless lives for the sake of own survival as well as the world peace. Animals keep ecosystems functional. Healthy ecosystems allow us to survive, to get enough food to eat and make a satisfied living in the way of life.

Similarly, in the case of creating any object oriented system, it is a general responsibility for an ethical engineer to develop the entire system keeping the benefit of real life environment or surroundings alongside. In this entire project, the main motive includes the service through which any specified zoo can run their respective food processing or maintenance for not only the animals or wildlife into the cages rather also for themselves indeed. Hopefully an entire food management system for any sort of zoo can be maintained much precisely through some basic object oriented programming concepts as much as possible.

3.2 Solution to the Problem

As a human being it is our responsibility to help other. One human can help another human here and there but nowadays animals are not getting their deserved affections to the environment which made us bound to think about such type of project domain and title as well....

The primary and main reason of choosing such type of project includes the service through which any specified zoo can run their respective food processing or maintenance for not only the animals or wildlife into the cages rather also for themselves indeed. Hopefully an entire food management system can also be appropriately run into a zoo through some basic object oriented programming concepts as much as possible.

This project not only can bring any sort of revolution to the food for creatures rather by running our application one particular user can also feel the satisfaction of feeding any animals according to their respective demand, track the foods available for the creatures and also give a positive or negative review on the basis of the uses indeed. From the perspective of Business strategy, if any particular company receives such project and make a measurable investment to the developer and tester organizations, then it will definitely reach up to those general mass who probably could not bother to other sorts of applications or software of that certain company before. This is definitely an open source application where both the developers and clients or users can interact with each other anyways to enhance the existence of creatures worldwide. And that is why we named our application as "***ZOODLE SPAGHETTI***" and set our tagline as: "***FOOD FOR CREATURES CREATES OUR PLEASURE.***"

4. REQUIREMENT SPECIFICATIONS

4.1 System Features

In this Object Oriented (**C#**) project a GUI (Graphical User Interface) based desktop application was tried to create with the collaboration of four members and thus the entire system and functionalities of this application is surrounded within four users as featured here:

Common Features:

All the users can login to their respective homepage or dashboard
Every users can create new account (or registration) for the respective entries into the system
All the users can view their own profile after successful login through their dashboards
All the users can logout from their respective accounts anytime after login to the system.

➢ ADMIN

1. Change Password to the System
2. Add/Remove/Edit Recruiters Information.
3. Add/Remove/Edit Zoo Keepers Information.
4. Add/Remove/Edit Visitors Information.
5. Add/Remove/Edit Animal Information.
6. Search for all User List:
   - ⇨ Admin/ Recruiters/ Zoo Keepers/ Visitors/ Animal
7. View All Transactions:
   - ⇨ View All Sales
   - ⇨ View Revenue
   - ⇨ View total list of food consumption by Animals
8. Create/Remove Contacts and Schedules with Food Provider Organizations.
9. Deactivate/Block Any Users: [Recruiter, Zoo Keeper, Visitor]
10. View All Food Items
11. View All Animals

➢ RECRUiTER

1. Insert/Update/Remove ZooKeeper
2. View all existing ZooKeepers
3. Insert/Update/Remove Visitors
4. View all existing Visitors
5. Create Salaries for existing ZooKeepers
6. Update/Remove Salaries for all ZooKeepers
7. Search all users [Admin, ZooKeeper, Visitors]
8. Insert new transactions for food sales [foods cost]
9. Update/Remove food costs of outlets/food carts
10. Insert/Update/Remove Transaction Types/categories [bkash, cards, bankChequeBook]
11. View all transaction records/history
12. Insert schedules for new meetings
13. Update/Remove schedules of new meetings
14. Search and view all meetings

15. View all food carts and food lists
16. Contact with admin and other users

➢ **ZOOKEEPER**

1. Crud operations for Food outlets/Carts [insert/update/remove]
2. Insert/Update Food Lists/Menu [both for human and animals]
3. Insert/Update Food Schedule [duty hours]
4. Feeding Animals [Managing food for animals]
5. Insert/update All Sales Records
6. View all sales Records
7. View All Animals
8. View All transaction lists
9. Contact with Admin

➢ **ViSiTOR**

1. View Cost and Schedule of visits
2. Insert/Update new form to book Tickets
3. Remove own created form before submission
4. View all food outlets of the system
5. View all food list of the outlets
6. Request for purchase any food
7. View all Animals
8. Insert/Update Reviews about animals [giving feedback on visit]
9. Contact with admin and other users

4.2 System Quality Attributes

Quality can be defined in different manners. The quality definition may differ from person to person. But finally, there should be some standards. If the product meets customer requirements then quality of a product can be defined as to be in a user-friendly manner by providing the required functionalities.

Quality Assurance Activities or Attributes are oriented towards the prevention of the introduction of defects and *Quality Control Activities* are aimed at detecting defects in products and services. The following can be relatable to assure the top quality for every products of the specified company stated in the statement here.

- ***Reliability***

Measure if the product is reliable enough to sustain in any condition. Also should give consistently correct results. Product reliability is measured in terms of working of the project under different working environments and different conditions.

- ***Maintainability***

Different versions of the product should be easy to maintain. For development it should be easy to add code to the existing system, should be easy to upgrade for new features and new technologies from time to time.

Maintenance should be cost-effective and easy. The system is easy to maintain and correcting defects or making a change in the software.

- ***Usability***

This can be measured in terms of ease of use. The application should be user-friendly. Also should be easy to learn. Navigation should be simple.

The system must be:

- Easy to use for input preparation, operation, and interpretation of the output.
- Provide consistent user interface standards or conventions with our other frequently used systems.
- Easy for new or infrequent users to learn to use the system.

- ***Portability***

This can be measured in terms of Costing issues related to porting, Technical issues related to porting, Behavioral issues related to porting.

- ***Correctness***

The application should be correct in terms of its functionality, calculations used internally and the navigation should be correct. This means the application should adhere to functional requirements.

- ***Efficiency***

A Major system quality attribute that is measured in terms of time required to complete any task given to the system. **For example,** the system should utilize processor capacity, disk space, and memory efficiently.

If the system is using all the available resources then the user will get degraded performance failing the system for efficiency. If the system is not efficient then it cannot be used in real-time applications.

- ***Integrity or security***

Integrity comes with security. System integrity or security should be sufficient to prevent unauthorized access to system functions, preventing information loss, ensure that the software is protected from virus infection, and protecting the privacy of data entered into the system.

- ***Testability***

The system should be easy to test and find defects. If required should be easy to divide into different modules for testing.

- ***Flexibility***

Software should be flexible enough to modify. Must be adaptable to other products with which it needs interaction. And sidewise also should be easy to interface with other standard 3rd party components.

- ***Reusability***

Software reuse is a good cost-efficient and time-saving development way. Different code library classes should be generic enough to use easily in different application modules. We are dividing the application into different modules so that modules can be reused across the application.

4.3 System Interface

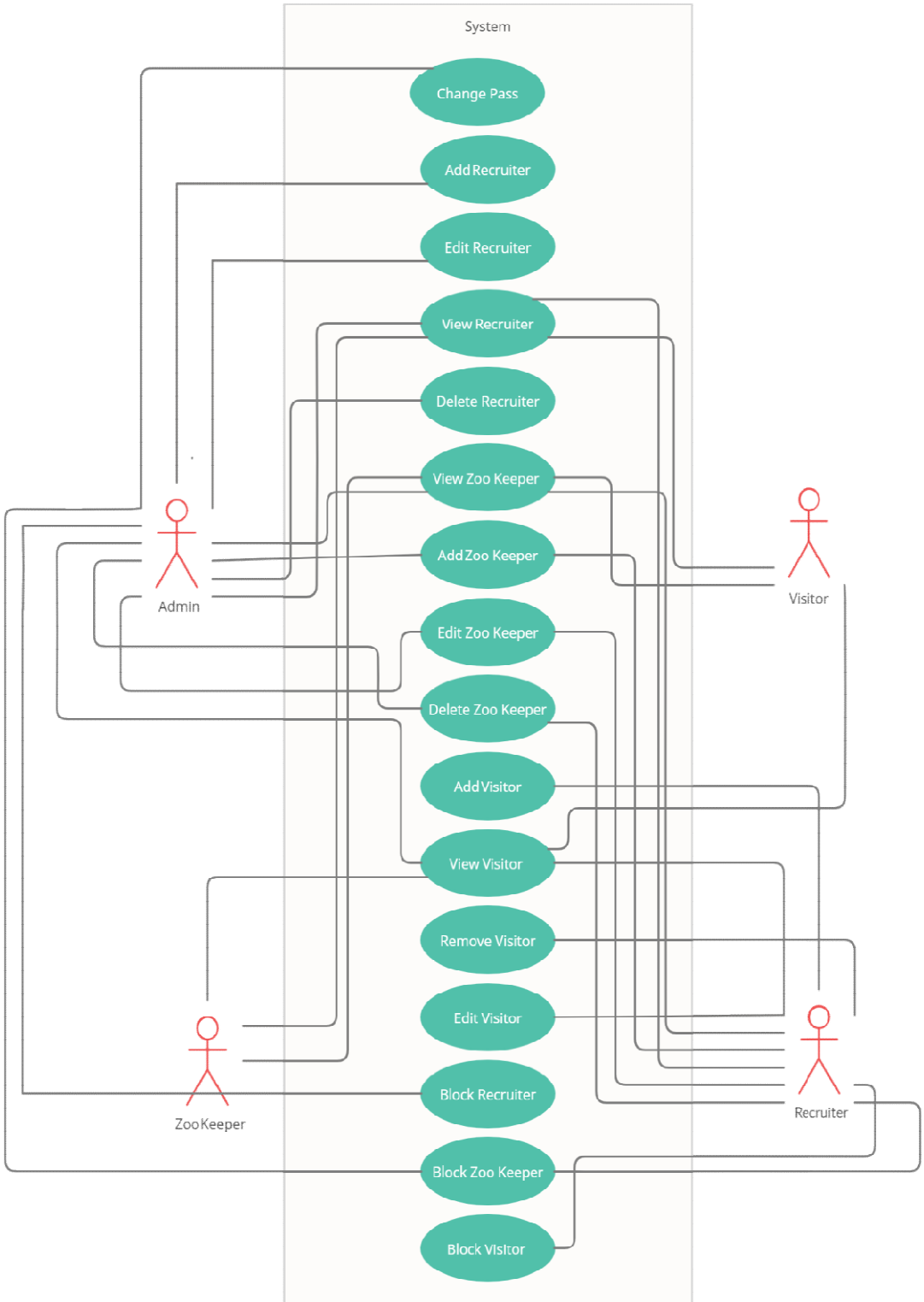

Fig 1: ***All the functions in the system Interface***

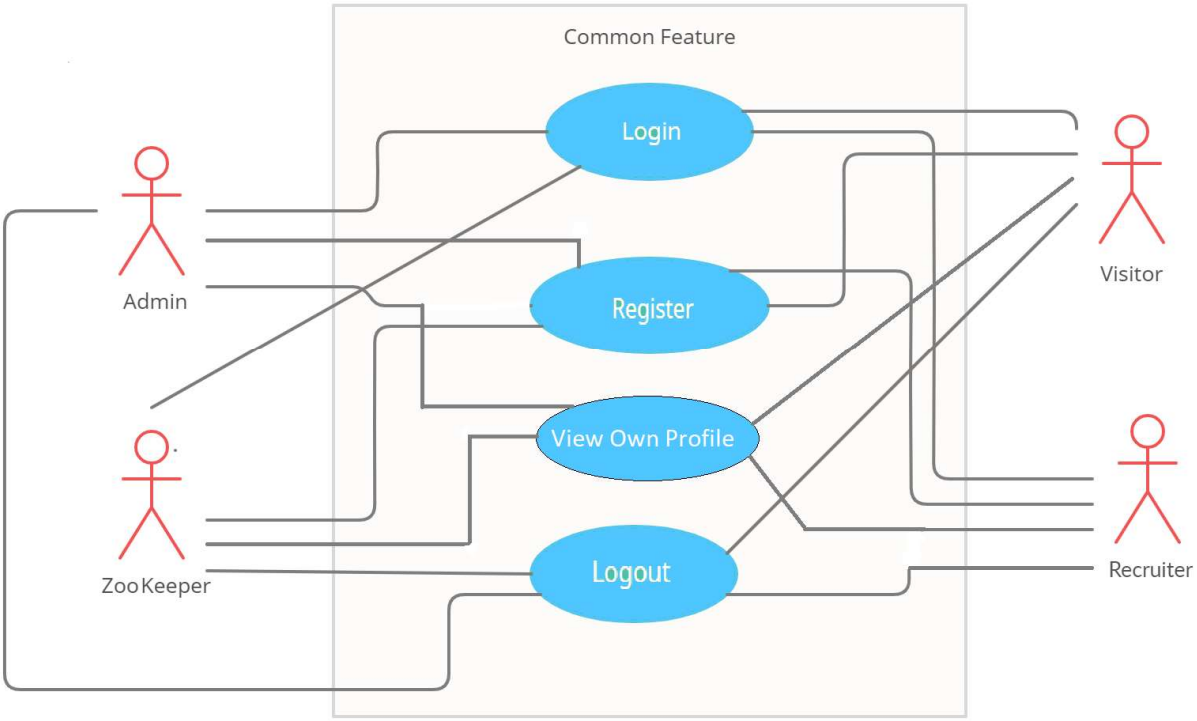

Fig 2: ***All the Common features in the system Interface***

4.4 Project Requirements

- Development Time [15-20 days]
- Sufficient Storage of Local Drive [at least 90 GB in Local Drive C for safety]
- Microsoft Visual Studio [an open source IDE software for **C#** programming]
- MSSQLSERVER [database connection done through this software]
- Appropriate Environment + Genuine Group Collaboration [since it is a group project]
- Optimum Knowledge on *Software Quality and Testing* + *Object oriented Programming* course

5. FEATURES NOT TO BE TESTED

The following is a list of the areas that will not be specifically addressed. All testing in these areas will be indirect as a result of other testing efforts. For example:

- USER ID and PASSWORD of each users should not be given to the Tester
- Contact with each and every user is an internal system function. So, it need not be tested by the tester.
- A certain Testing Team will not calculate any sort of quantity based transactional stuffs. Rather they will test how smooth and accurate the entire software quality stands for.
- The system information should be protected by ADMIN of this system for the ultimate security of entire PC.

6. TESTING APPROACH

6.1 Testing Levels

The testing levels for the particular project "ZOODLE SPAGHETTi" consists of Unit Testing, System/Integration (combined) Testing and Acceptance Test levels. It is believed that there would be at least one full time independent test person for system/integration testing. However, with the budget constraints and timeline established; most testing will be done by the test manager with the development team`s participation.

- ✓ **UNIT TESTiNG** is such a White Box testing technique that is usually performed by the developer. UNIT Testing will be done by the developer and will be approved by the development team leader.

  Proof of unit testing (test case list, sample output, data printouts, and defect information) must be provided by the programmer to the team leader before unit testing will be accepted and passed on to the test person. All unit test information will also be provided to the test person. Unit Tests isolate a section of code and verify its correctness. A unit may be an individual function, method, procedure, module, or object.
- ✓ **SYSTEM/INTEGRATION TESTiNG** will be performed by the test manager and development team leader with assistance from the individual developers as required. LambdaTest can be one of the integration testing tools that help to provide the percentage of accuracy of this entire project if any software developer can test the other sides of this project accordingly and appropriately. Programs will enter into System/Integration test after all critical defects have been corrected. A program may have up to two Major defects as long as they do not impede testing of the program (I.E. there is a work around for the error).
- ✓ **ACCEPTANCE TESTiNG** is a testing technique performed to determine whether or not the software system has met the requirement specifications. The main purpose of this test is to evaluate the system's compliance with the business requirements and verify if it is has met the required criteria for delivery to end users. Acceptance Testing will be performed by the actual end users or client side of this project with the assistance of the test manager and development team leader. The acceptance test will be done in parallel with the existing manual ZIP/FAX process for a period of one month after completion of the System/Integration test process.

6.2 Test Tools

The only test tools to be used are the standard AS/400 provided utilities and commands. The Program Development Manager (PDM) will be used as the source version configuration management tool in conjunction with the in-house check-in/check-out control utility. The check-in/out utility is part of each developer's standard AS/400 access menu.

The initial prototypes for the new screens will be developed using the AS/400 Screen Design Aid (SDA). The initial layout and general content of the screens will be shown to the sales administration staff prior to proceeding with testing and development of the screens.

Moreover some special sort of testing tools like **Selenium, LambdaTest** had been also used for this project not only to test the automated browser testing of the real time Zoo related desktop application (Selenium) but also to integrate the entire system configuration (LambdaTest) specially using *Mozilla Firefox* as the specific Browser in both cases.

6.3 Meetings

The test team should meet once every two weeks to evaluate progress to date and to identify error trends and problems as early as possible. The test team leader must meet with development and the project manager once every two weeks as well. These two meetings will be scheduled on different weeks. Additional meetings can be called as required for emergency situations.

And within last two weeks for some important task oriented purposes we did 5 countable meetings through the application **Microsoft Teams** starting since 01-AUGUST-2021 till 13-AUG-2021 indeed.

7. TEST CASES/TEST ITEMS

Project Name: Zoo Food Maintenance System	Test Designed by: EMON
Test Case ID: ZS_OOP2_1	Test Designed Date: 07.08.2021
Test Priority (Low, Medium, High): HIGH	Test Executed by: MAHNAJ
Module Name: Run The Software Successfully	Test Execution date: 09.04.2021
Test Title: Login to the system with valid username and password and logout properly	
Description: Test all the buttons and functions of the desktop application	

Test Steps	Test Data	Actual Results	Expected Results	Status
1. Go to the github Link Provided below: https://github.com/emon-iqbal/OOP2_PROJECT_Software 2. Download the code folder and extract it to the desktop 3. Run the software version in the desktop directory with the extension .exe of the filename 4. Start the login into the system 5. If username and password is unknown then press the **CREATE NEW ACCOUNT** button. 6. Fill the specified form 7. Now go back to login page 8. For the sake of this time, start the login as a **RECRUiTER**		**Start the desktop software application, Login and Successfully logout from the entire System**	As expected	**SUCCESSFUL**

9. Now follow all the buttons of the dashboard and try to click each of them one by one 10. Test the insert operation of the page where **ADD** button is visible with Blink function at the bottom of screen 11. Try to update some existing values 12. If possible Insert any particular attribute to the **Salaries of ZooKeeper** Button of Dashboard and then update any one value and finally remove or delete it. 13. After all doings, try to logout from the account with the **LOGOUT** button. 14. Finally try to exit the application not by the default x button rather the custom X button.				
Post-Condition: if everything goes correctly, then the system interface is operated successfully. For any sort of errors please do check the database connection first.				

8. ITEM PASS/FAIL CRITERIA

The test process will be completed once the initial set of distributors have successfully sent in reassigned sales data for a period of one month and the new EDI data balances with the old ZIP/FAX data received in parallel. When the sales administration staff is satisfied that the data is correct the initial set of distributors will be set to active and all parallel stopped for those accounts.

9. TEST DELIVERABLES

- Acceptance test plan
- System/Integration test plan
- Unit test plans/turnover documentation
- Screen prototypes
- Report mock-ups
- Defect/Incident reports and summaries
- Test logs and turnover reports

10. STAFFING AND TRAINING NEEDS

It is preferred that there will be at least one (1) full time tester assigned to the project for the system/integration and acceptance testing phases of the project. This will require assignment of a person part time at the beginning of the project to participate in reviews etc... and approximately four months into the project they would be assigned full time. If a separate test person is not available the project manager/test manager will assume this role. In order to provide complete and proper testing the following areas need to be addressed in terms of training.

- The developers and tester(s) will need to be trained on the basic operations of the EDI interface. Prior to final acceptance of the project the operations staff will also require complete training on the EDI communications process.
- The sales administration staff will require training on the new screens and reports.

11. RESPONSIBILITIES

	TM	PM	Dev Team	Test Team	Client
Acceptance test Documentation & Execution	X	X		X	X
System/Integration test Documentation & Exec.	X		X	X	
Unit test documentation & execution	X		X	X	
System Design Reviews	X	X	X	X	X
Detail Design Reviews	X	X	X	X	
Test procedures and rules	X	X	X	X	
Screen & Report prototype reviews			X	X	X
Change Control and regression testing	X	X	X	X	X

12. TESTING SCHEDULE

13. PLANNING RISKS AND CONTINGENCIES

- Limited Reassigned staff. The Reassigned Sales administration staff currently has two positions unfilled. As a result of this staff shortage there may be delays in getting staff to review appropriate documents and to participate in the Acceptance test process. Should client staff become a problem, the appropriate dates for reviews and acceptance testing will slip accordingly. No attempt will be made to bypass any part of the review and testing processes.

14. APROVALS

Project Sponsor - Steve Sponsor	
Development Management - Ron Manager	
EDI Project Manager - Peggy Project	
RS Test Manager - Dale Tester	
RS Development Team Manager - Dale Tester	
Reassigned Sales - Cathy Sales	
Order Entry EDI Team Manager - Julie Order	