Egerton University

Group 3 project

Object Oriented Analysis and Design

Estates Department Inventory Management System

Brian John Mbugua SP13/02574/16

Shedrack kasyoki S13/15342/16

Doris Nyawira S13/15354/16

Morris Waweru S13/15313/16

Brian Ariga SP13/02810/16

Victor Musembi S13/15331/16

COMP362 Group Project

Table of Contents

[1. INTRODUCTION 1](#_Toc963842)

[1.1 Purpose 1](#_Toc963843)

[1.2 Intended Audience 1](#_Toc963844)

[1.4 Project Scope 1](#_Toc963845)

[1.5 Glossary 1](#_Toc963846)

[1.6 References 1](#_Toc963847)

[2. Overall Description 2](#_Toc963848)

[2.1 Product Perspective 2](#_Toc963849)

[2.2 Operating Environment 2](#_Toc963850)

[2.3 Assumptions and dependencies 2](#_Toc963851)

[3. System Requirements 3](#_Toc963852)

[3.1 User Interface Requirements 3](#_Toc963853)

[3.2. Functional Requirements 3](#_Toc963854)

[3.3 Non-functional Requirements 3](#_Toc963855)

[Use Case Diagram 4](#_Toc963856)

[Use case description 5](#_Toc963857)

[Log in to the system 5](#_Toc963858)

[Add new user 6](#_Toc963859)

[Update database 7](#_Toc963860)

[View Records 8](#_Toc963861)

[Use case description for materials required 9](#_Toc963862)

[Use case description for adding stock 10](#_Toc963863)

[Use case diagram for shopkeeper’s approval 11](#_Toc963864)

[Use case diagram for Manager’s approval 12](#_Toc963865)

[Activity Diagrams 13](#_Toc963866)

[Class Diagram 14](#_Toc963867)

[Sequence Diagrams 15](#_Toc963868)

[Manager Sequence Diagram 15](#_Toc963869)

[Worker sequence diagram 16](#_Toc963870)

[Storekeeper sequence diagram 17](#_Toc963871)

# 

# 1. INTRODUCTION

## 1.1 Purpose

To come up with a system that help track inventory in the Estates Department store.

## 1.2 Intended Audience

This store management system is intended for Estates department staff for use within Egerton university. There will be three classes of users. The worker who is the one in charge of finding out what materials are required for a specific repair and maintenance activity. The storekeeper who is focused on managing inventory. Lastly, the manager, who oversees all operations and accesses records to allow for budget planning.

## 1.4 Project Scope

This project will help the Estates department in tracking their inventory as they carry out their duties. Management staff will be able to manage and quantify materials available in the store. It will also help them develop an accurate budget plan at the end of the financial year.

The Estates department represents the University in capital projects and they are responsible for repair and maintenance of university infrastructure. There are various sections such as masonry, carpentry, painting, welding etc. Materials are bought at the beginning of the financial year after they receive money from the university.

Currently, they do not have an automated system for monitoring their inventory. They use bin cards to find out which materials are used up and which one are available. This system will aid them in tracking their inventory records.

## 1.5 Glossary

Software – A collection of instructions that are implemented in a computer to perform certain instruction.

Project – A temporary undertaking with a start and an end.

System – A set of things working together as part of mechanism or interconnective network.

# 1.6 References

Ojo, A., & Estevez, E. (2005). Object-oriented analysis and design with UML. *Book Object-Oriented Analysis and Design with UML*.

# 2. Overall Description

## 2.1 Product Perspective

The inventory management system stores the following information.

i. Materials Available

The database stores materials that are available for use. Once materials are used they are deducted from the record.

ii. Cost of materials

Materials bought will be recorded, together with their cost. This will help determine the expenditure of the department.

iii. Materials used

The database will show how frequently materials are needed throughout the year and provide information on how to make purchases for the next financial year.

## 2.2 Operating Environment

- Windows Operating System

- MySQL database

- Java Platform

## 2.3 Assumptions and dependencies

The Estates department needs to have a network in place allowing all computers to access each other.

# 3. System Requirements

## 3.1 User Interface Requirements

1. Easy to use

The system should be easy to learn for all users. It should not take long for users to get used to the system.

2. Readable

The font should be large enough for users to read. No unnecessary animations.

## 3.2. Functional Requirements

1. Add new user to system – The manager should be able to add new staff to the system. This functionality will take information provided by the manager and add it to the database provided they have been validated and found to be okay.
2. Add new materials to the database – The storekeeper will be able to add new materials to the database. This process will require the permission of the manager who will verify and confirm that money has been spent acquiring new materials.
3. Update material information to database – After a worker specifies the materials that are necessary to do some specified amount of repair work, this information is submitted to the system and the storekeeper will allow the materials to be used. This will trigger the database to update its records and deduct the provisioned materials.
4. Calculate cost of materials used per financial year – The system should record all the information about the buying of materials and how frequently they are used.
5. Display stock records – The system should generate a report showing how materials have been used for a given period of time. This should give the manager enough information to draft a budget plan for the next financial year.

## 3.3 Non-functional Requirements

1. Access database object to manipulate records – The information about stock and users will be stored in a database. This database will be secured by passwords and access privileges.
2. Access a network for information sharing – The database will be stored on a central computer which is available on the network for all the other computers to access it.

# Use Case Diagram

Store Keeper

Manager

Worker

Inventory Management System

<<extends>>

<<extends>>

<<includes>>

<<includes>>

<<includes>>

# Use case description

## Log in to the system

|  |  |
| --- | --- |
| Use case name | Log in to the system |
| Description | Allows user to enter a username and a password. These entries are validated against entries in a database and access is provided according to the user’s rank in the organization. |
| Primary Actor | Worker, Storekeeper, Manager |
| Secondary actor | None |
| Included use cases | None |
| Extended use cases | None |
| Preconditions | None |
| Postconditions | Access to system information |
| Main flow | 1. User provides a username  2. User provides a password  3. User submits the entries to the system  4. System validates the entries  5. User is granted access to system resources |
| Alternative flow | 3. a) Wrong username  1. User enters username  2. User enters password  3. User submits the entries  4. System validates entries  5. System displays error message  3. b) Wrong password  1. User enters username  2. User enters password  3. User submits the entries  4. System validates entries  5. System displays error message |

## Add new user

|  |  |
| --- | --- |
| Use case name | Add new user |
| Description | Allows new users to be added into the system. The manager has the permission to add new staff. Each new member will be added and given a rank to dictate privileges and access permissions. |
| Primary Actor | Manager |
| Secondary actor | None |
| Included use cases | Update database |
| Extended use cases | Display error message |
| Preconditions | Manager needs to have logged in |
| Postconditions | A new user will be added to the database |
| Main flow | 1. Manager provides user details  2. Manager submits the entries to the system  3. System validates the entries  4. A new user is added to the system |
| Alternative flow | 3. a) Invalid details  1. Manager enters username  2. Manager submits the entries  3. System validates entries  4. System displays error message |

## Update database

|  |  |
| --- | --- |
| Use case name | Update database |
| Description | Allows the user specifically the storekeeper to add or remove materials in the store. Materials needed at certain points are freed whereas the needed are easily added through updating of the stock. |
| Primary Actor | Storekeeper. |
| Secondary actor | None |
| Included use cases | Add |
| Extended use cases | Approve |
| Preconditions | None |
| Postconditions | Managers approval. |
| Main flow | 1. Storekeeper enters into the system  2. Storekeeper adds material.  3. The System updates the stock with the added materials. |
| Alternative flow | a) Material overflow  1. Shopkeeper adds materials.  2. The system displays an error message.  3. The materials added are rendered maximum.      b) Low or no materials  1. The Shopkeeper adds materials.  2. The system displays error message.  3. The materials added are unavailable. |

## View Records

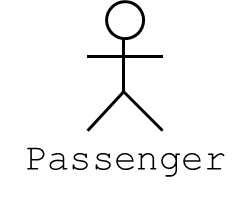
Manager

|  |  |
| --- | --- |
| Use case name | View Records |
| Description | Allows the manager be able to access the store database and view the current or existing stock records |
| Primary Actor | Manager |
| Secondary actor | None |
| Included use cases | None |
| Extended use cases | None |
| Preconditions | Manager needs to have logged in |
| Postconditions | Manager will have viewed the records |
| Main flow | 1. Manager provides the query to view records from the database  2. The system displays the stock records depending on criteria  3. Actor then views the records |
| Alternative flow | None |

|  |  |
| --- | --- |
| Use case Name | Materials Required |
| Description | Allows the worker to identify the fault in an operation and the materials required for the repairment. The worker then enquires from the store keeper if the materials are available. The store keeper approves if the materials are available and adds them if they are not available. |
| Primary Actor | Worker |
| Secondary Actor | None |
| Included Use case | None |
| Extended use cases | None |
| Pre-conditions | User needs to log in |
| Post-conditions | The system displays the available materials. |
| Main flow | 1. The worker enters the materials needed. 2. The worker submits the materials. 3. The system validates the materials. 4. The system verifies the materials. 5. The system displays all the materials available. |
| Alternative flow | 3. a) Materials are not available:   * The system prints an error message showing that the specified materials are not available. |

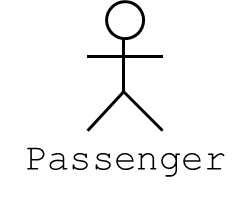
## Use case description for materials required

Worker Use case



## Use case description for adding stock

|  |  |
| --- | --- |
| Use case Name | Materials Required |
| Description | Allows the storekeeper to add stock into the store. Takes in material information, checks whether the materials exist, if they do it adds the new number to the existing records, if not it creates a new record. |
| Primary Actor | Storekeeper |
| Secondary Actor | None |
| Included Use case | Update Database |
| Extended use cases | None |
| Pre-conditions | User needs to log in |
| Post-conditions | The system displays the available materials. |
| Main flow | 1. The shopkeeper enters the information of every stock. 2. The shopkeeper submits the stock information. 3. 3.The system validates the stock information. 4. 4. The system validates the stock information. |
| Alternative flow | 3. a) Stock information incorrect:   * The system prints an error message showing that the information needs to be re-entered. |



Storekeeper

Use case

## Use case diagram for shopkeeper’s approval

Storekeeper

Storekeepers Approval

Use case description

|  |  |
| --- | --- |
| Use case name | Storekeeper’s approval |
| Description | Allows the storekeeper to approve for materials to be removed from the store |
| Primary actor | Storekeeper and manager |
| Secondary actor | None |
| Included use case | Manager Approval |
| Extended use case | Display Error message, Update database |
| Preconditions | Must be logged into the system and the materials requested must be available |
| Postconditions | Access system information and approve for materials |
| Main flow | 1. Give username 2. Give password 3. Validation of the credentials given to the system 4. Access stock records in the system 5. Check if the materials needed are available 6. If the materials are available approve |
| Alternative flow | 3.(a) Give wrong username validation fails and error message is displayed  (b) Give wrong password validation fails and error message is displayed  5.(a)If materials not available, cannot approve. |

## Use case diagram for Manager’s approval

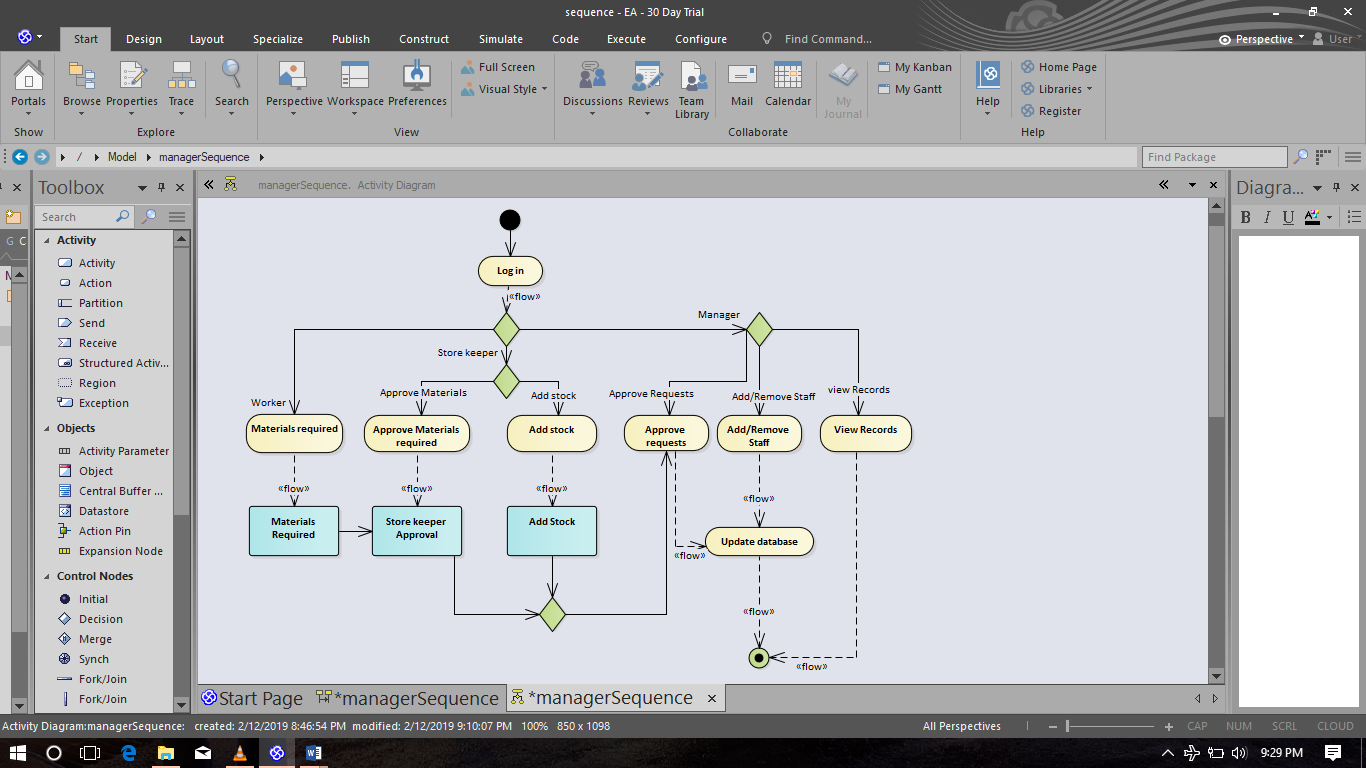
Manager

Manager Approval

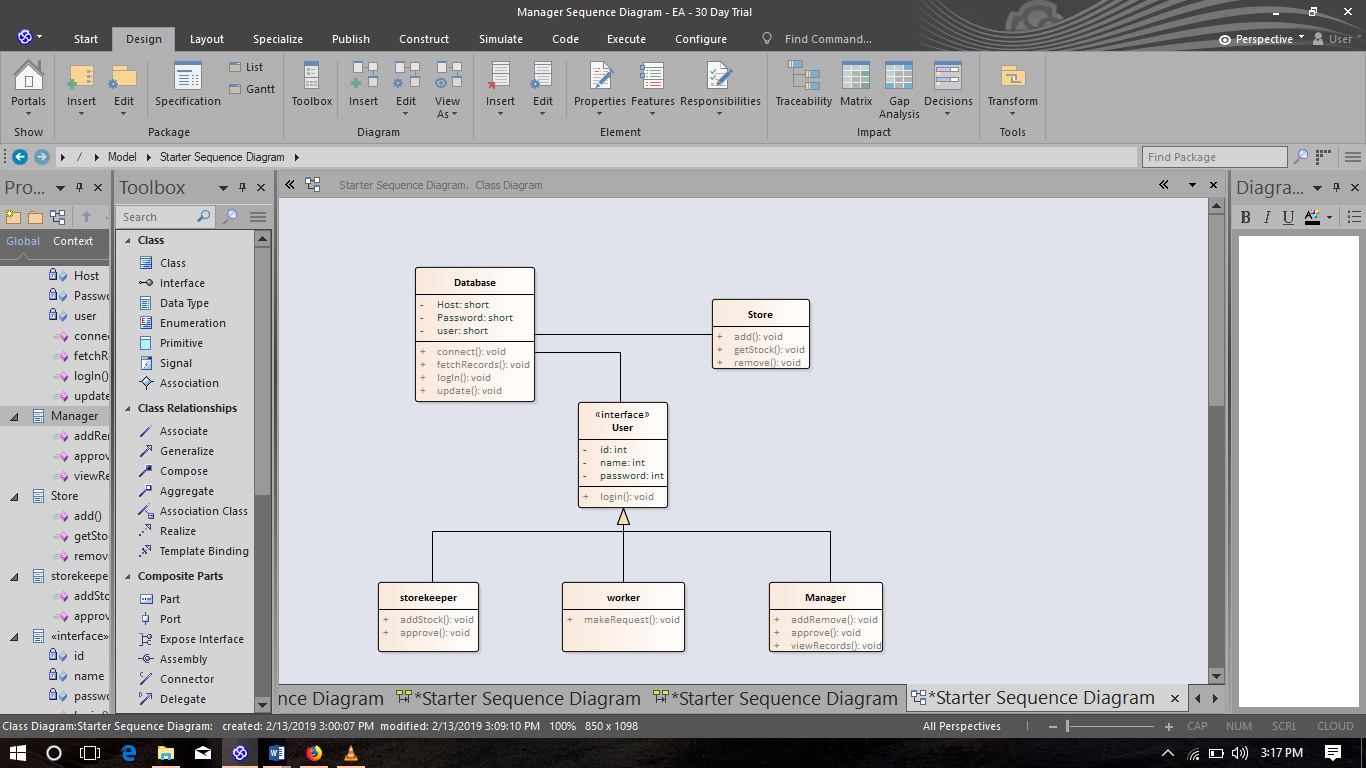
Use case description

|  |  |
| --- | --- |
| Use case name | Manager’s approval |
| Description | Allows the manager to approve for materials to be removed from the store |
| Primary actor | Storekeeper and manager |
| Secondary actor | None |
| Included use case | None |
| Extended use case | Storekeeper approval |
| Preconditions | Must be logged into the system and the materials requested must be available |
| Postconditions | Access system information and approve for materials |
| Main flow | 1. Give username 2. Give password 3. Validation of the credentials given to the system 4. Access stock records in the system 5. Check if the materials needed are available 6. If the materials are available approve |
| Alternative flow | 3.(a) Give wrong username validation fails and error message is displayed  (b) Give wrong password validation fails and error message is displayed  5.(a) If materials not available, cannot approve. |

# Activity Diagrams

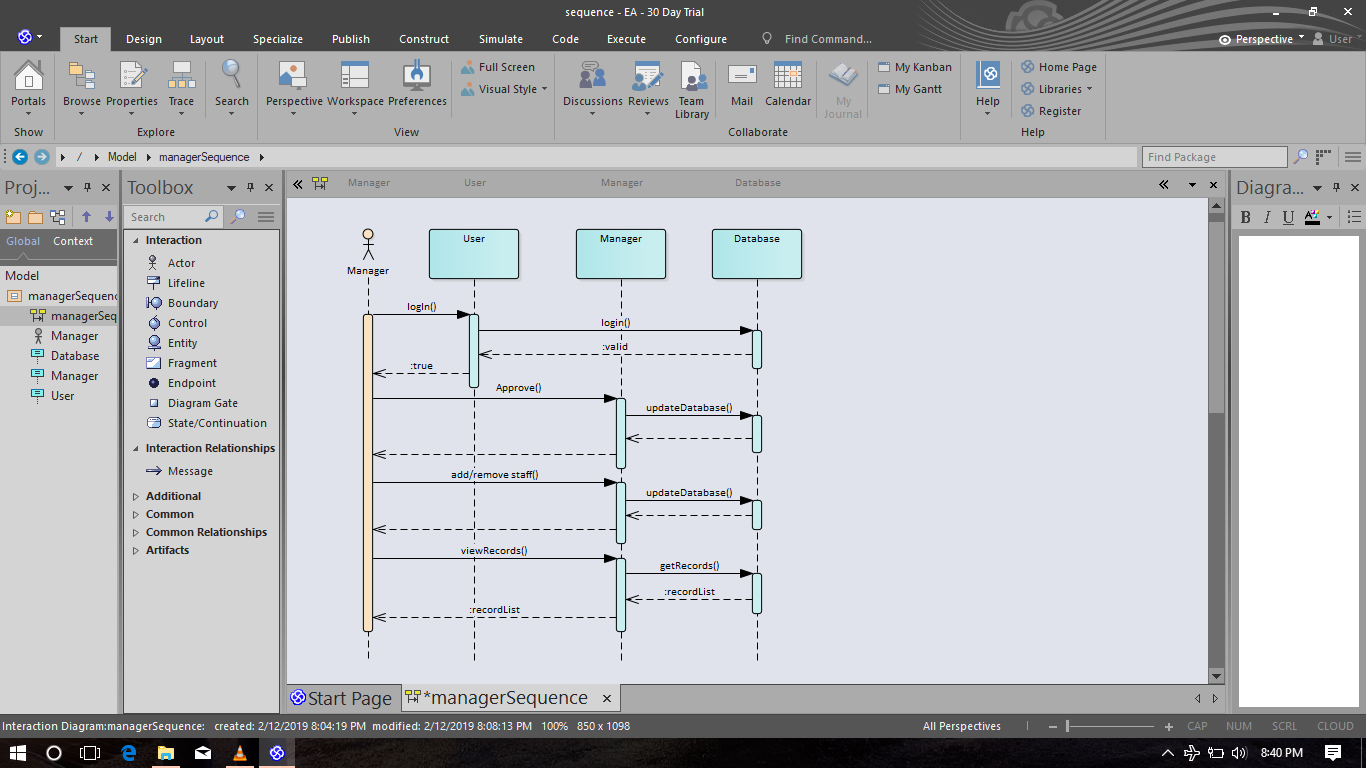


# Class Diagram

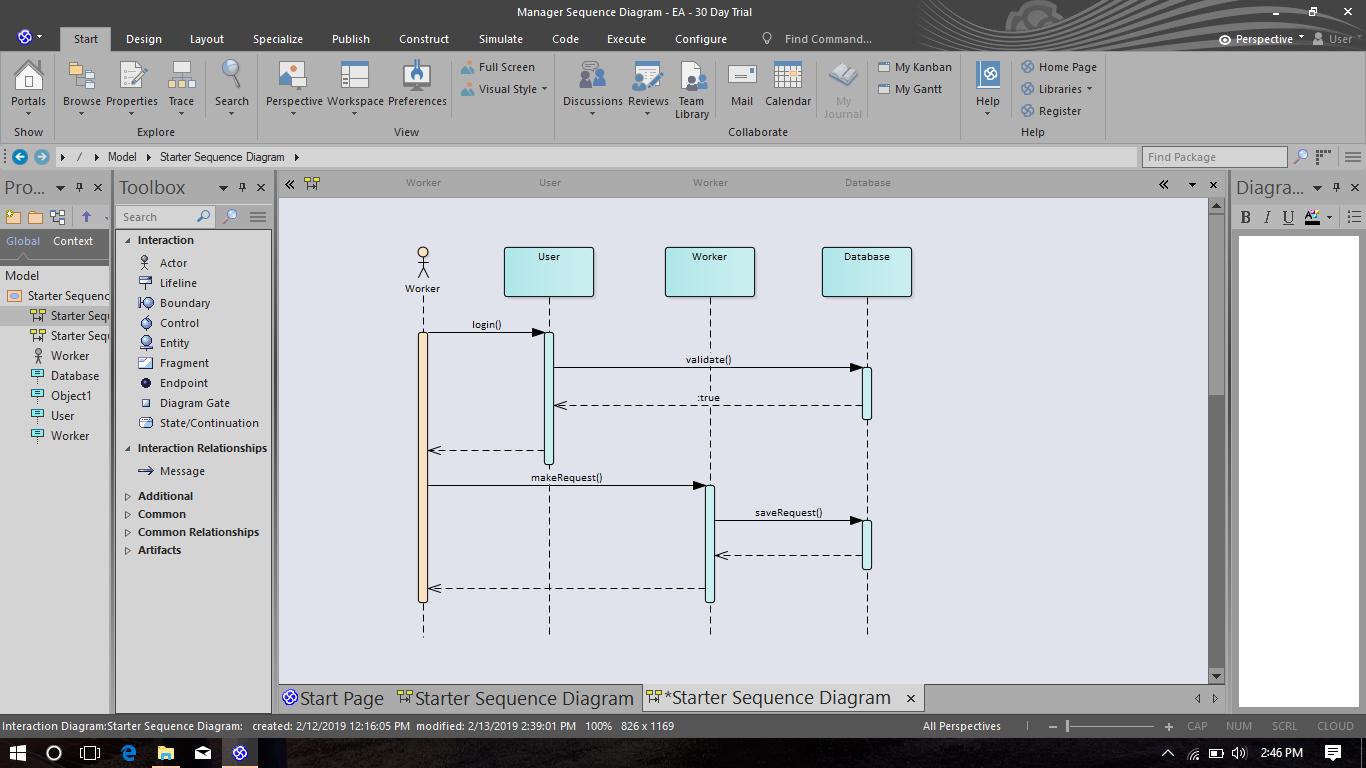


# Sequence Diagrams

## Manager Sequence Diagram



## Worker sequence diagram



## Storekeeper sequence diagram

