

**Course code**

**Course Name**

Project Name

**Final Project**

Full Name ID

Nov 2020

Contents

[Instructions [To be removed] 3](#_Toc402452668)

[Team 3](#_Toc402452669)

[Document Purpose and Audience 3](#_Toc402452670)

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope 4](#_Toc402452673)

[Definitions, acronyms, and abbreviations 4](#_Toc402452674)

[Requirements 4](#_Toc402452675)

[Functional Requirements 4](#_Toc402452676)

[Non Functional Requirements 4](#_Toc402452677)

[System Models 5](#_Toc402452678)

[Use Case Model 5](#_Toc402452679)

[Use Case Tables 5](#_Toc402452680)

[Ownership Report 6](#_Toc402452681)

[Policy Regarding Plagiarism: 6](#_Toc402452682)

# Introduction

## Software Purpose

* **Summarize the purpose of the software**

## Software Scope

* **Any software could have too many components / Major features .. but we should implement specific things...this is the scope**
* **In simple points, what is the software scope (focus on components / Major features, not tiny things)**

## Definitions, acronyms, and abbreviations

* **In a table, list all needed ones. Consider the audience**
* **Think as following: Document has abbreviation ATM..IFF audience doesn’t know it, let’s clarify it.**

# Requirements

## Functional Requirements

* **This is the most critical part...** **functional requirements describe what the system should do**
  + **E.g. an ATM allows you to enter Card, enter user name password and withraw a money**
* **List the system requirements, respecting the problem statement** 
  + **Make sure to go in the missing details for the mentioned features/components**
  + **Going beyond them (e.g. adding new complete major feature / component) is breaking the statement scope**
* **Each requirement should be clearly described, such that it can be understood without the presence of the one who wrote it.**
* **This part is the basis for writing the contract with client and estimating the size, time and cost of developing the software.**

## Non Functional Requirements

* **Non-functional requirements describe how the system works**
  + **E.g. Withdraw operation will be done within 20 second. Network is using secured protocols. System allows up to 30,000 withdrawals per minute.**
  + **Think about the operation / system quality**
* **There are too many non functional requirements. Read in** [**wiki**](http://en.wikipedia.org/wiki/Non-functional_requirement)**. Pick the suitable ones for your system. Non-functional requirements must be VERIFIABLE, i.e., MEASURABLE.**
  + **Some Types as just examples: Usability, Reliability, Performance, Security, Scalability, Portability, Maintainability**
  + **Select the sutiable ones, for each one write the details**
  + **Be realistic ☺**

|  |  |
| --- | --- |
|  | **Details** |
| **Performance** | * **Withdraw operation will be done within 20 second** |
| **Scalability** | * **System should be able to support up to 1000 simultaneous game players.** |

# System Models

## Use Case Model

* **Using UML, write the use case model expressing the system actors & operations**

## Use Case Tables

* **Using below table template, for each requirement write a use case table that shows user/system interaction**
  + **If one requirement is so big, you could divide it to more than table**
  + **If some requirements are not major, you could plugin them in other senario**
    - **E.g. you may not do LogIn Usecase table as it is simple functionality**
* **Flow of events should be very detailed**

|  |  |  |
| --- | --- | --- |
| Use Case ID: |  | |
| Use Case Name: |  | |
| Actors: |  | |
| Pre-conditions: |  | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter Card and Password. |  |
|  | 2- System Verify user data |
| 3- User Select Vodafone from the list |  |
|  | 4- System retrieves Vodafone bills |
| and so on |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter Card and Password. |  |
|  | 2- Card is invalid and unreadable.  3- System rejects cars. |
| Includes: |  | |
| Notes and Issues: |  | |