Software Requirements Specification

“Version 1.0”
October 21,2018

Department Coordinator

Prepared by A Team of students

Fayoum University

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the department coordinator app.

- *It will explain the purpose and features of the system,*
- *the interfaces of the system, what the system will do,*
- *the constraints under which it must operate and how the system will react to external stimuli*

1.2 Document Conventions

- *Titles in Bold font with capital every word*
- *External technology or shortcut will be all in capital*
- *Formatting and section division referred from IEEE standard station template*

1.3 Intended Audience and Reading Suggestions

developers, testers, and documentation writers: may refer to section 2 which describes overall view of project and its classes

1.4 Product Scope

Program source code will be available at this report
https://github.com/abdo1819/tansiq

1.5 References

[none]

2. Overall Description

2.1 Product Perspective

The Department Coordinator should facilitate the way of organizing the process of disturbing the students of a certain university inside the departments of their faculties by an automotive way and replace the old way of doing that process that depends mostly on human work. Grades of students will be stored through database for each faculty and it will be compared automatically by software and then they will be distributed over departments according to their grades and choices. The choices of students will be entered via web page.

2.2 Product Functions

Functions that are related directly to user's interaction via the web app:

- *Login and storing data of users.*
- *Verify that data of users is correct.*
- *Student Functions:*
- *Filling their choices of the departments.*

Admin's Functions:

- *Specify time of the ending of the process.*
- *The required data for starting the process of disturbing the students.*
- *Specify other required people for the process.*

Functions that are related the software and there is no need for interaction from users:

- *Storing student's data and admins data in database for each faculty*
- *Comparing students grades and specifying the department for each one of them.*
- *Storing the result of each department in a table.*

2.3 User Classes and Characteristics

Student:

Student is the targeted user for that web app he will access the site through login page and his own grades are stored in a private database.

- ***Student's actions:***
  - *Will have access to enter his choices of the departments according to his preferred order*
  - *Will have access to modify his choices in a specified period specified by an admin*
  - *Students will be able to read an introduction about each department*

- ***Student's data:***
  - *Each student has his own grades that are the only factor that make him join the chosen departments.*
  - *The national id of the student, a password and an e-mail addresses are stored in database for verifying the student's data.*

- ***Server admin:***
  Administrator hired by the university responsible for adding admins for each faculty and managing the database and maintenance of the program.

- ***Faculty Admin:***
  - *Will be responsible for inserting student data every year.*
  - *Specify a period for students to enter their choices.*
  - *Specify the department s data:*
    - *Required courses to join that department.*
    - *Required capacity of each department.*

2.4 Operating Environment

- *U**ser side:***
  any browser that can view html site program is made and tested for google chrome

- ***Server side:***
  any operating system with support to JVM

2.5 Design and Implementation Constraints

Security Constraints:

The admins will not have the ability to change student's wished department only the student has the right to change his own order of choices in specific time determined by admins.

a password will be provided for student so he can access the site and change his wishes
this password may be provided by mail or from faculty [to be determined]

Tools:

- *Spring framework*
- *MVC pattern*
- *Oracle database*

Communication protocols:
HTTP and HTTPS

2.6 User Documentation

will be provide a document for student explaining a step by step how he can make his department wishes and edit them

for admin there will be a document explaining the how to manage department - add/edit/delete- how to add student from excel datasheet and required format of sheet with example of required tested sheet

2.7 Assumptions and Dependencies

we use MAVEN [1] for managing dependencies which assume there is an unrestricted internet connection to maven project at first run
JRE [2] v10 is installed in system for running java program
[to be added] packaging the whole program in single file with all dependencies

3. External Interface Requirements

3.1 User Interfaces

When A Registered User, as an unregistered user will see an error message if they haven't for wrong data, Opens The web page, They Should See a Login Page Asking for Their Id and Password. When The User Enters the Application correctly, he will see his Data. This Happens Only When the User Isn't a First-Time User. After That, The User Can Choose Departments, organized in A Numerical Order and This Application Will Choose One Due to The Terms of Every Department,

student interface:

- *login page*
  - *simple login page requires*
    - *student national id or a specific unique model [to be determined]*
    - *password provided by our system for every student*

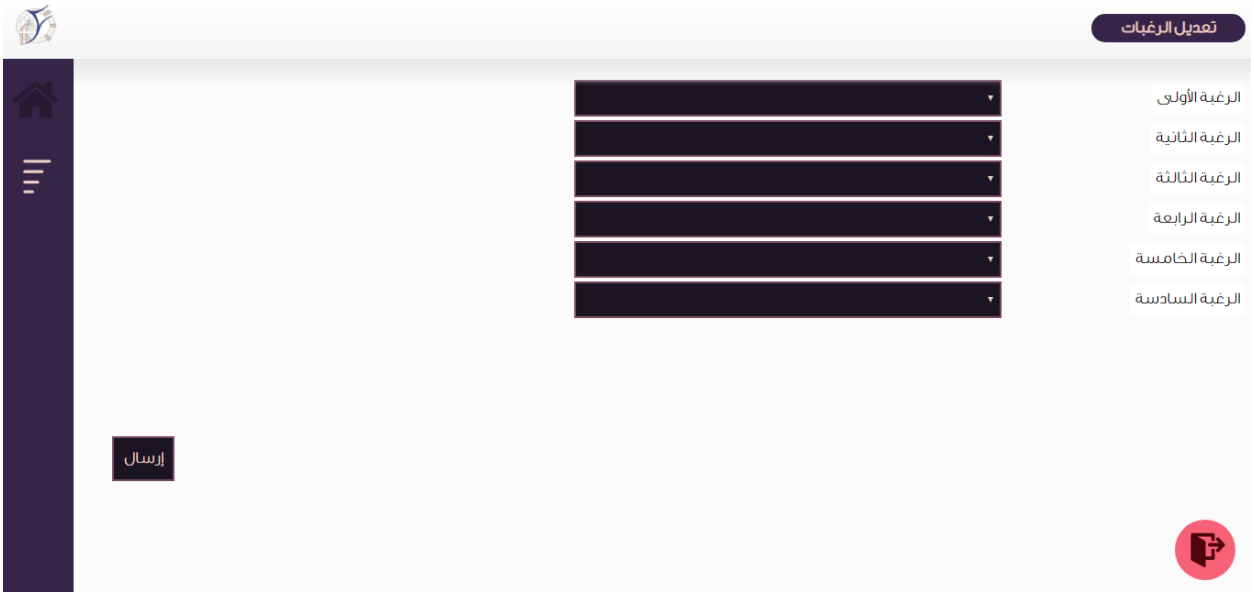
- *data reviewing*
  - *page containing student data saved in our database so he can review it and if there is any error he may refer to admin for edit with ability to send mail for admin from feedback page*

[photo of data reviewing page] [next version]

- *feedback page*
  - *text box for student to send a feedback or error in his data to admin*

[photo of feedback page] [next version]

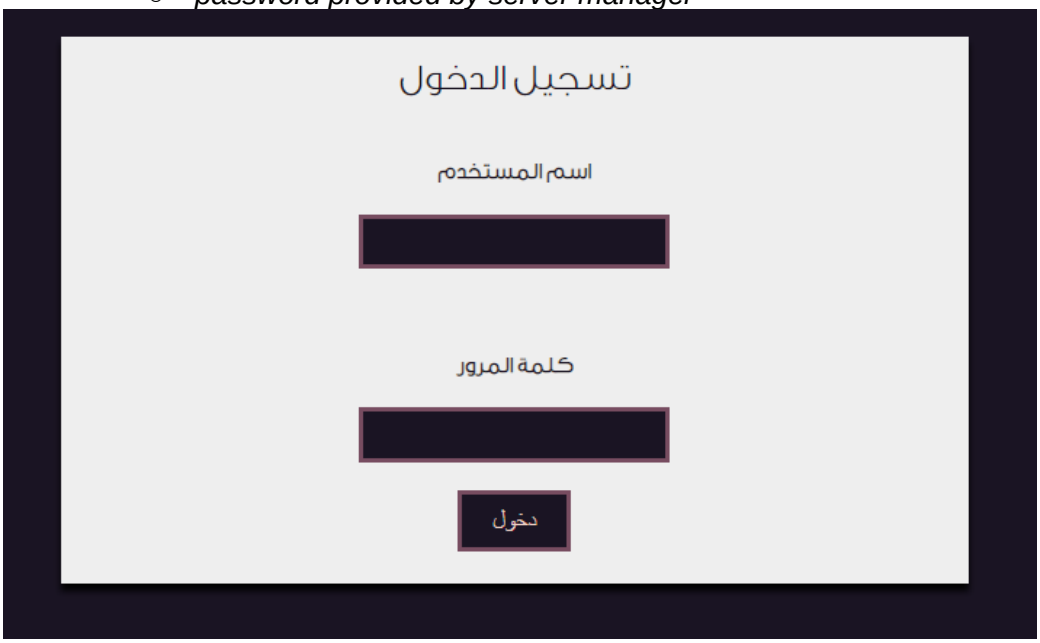
- *filling department wishes*
    - *here student can fill in his wishes page consist of*
        - *drop list containing available department to choose from with button to add*
        - *editable order list of his wishes with sorting buttons*



- *final page [to be determined]*

faculty admin interface:

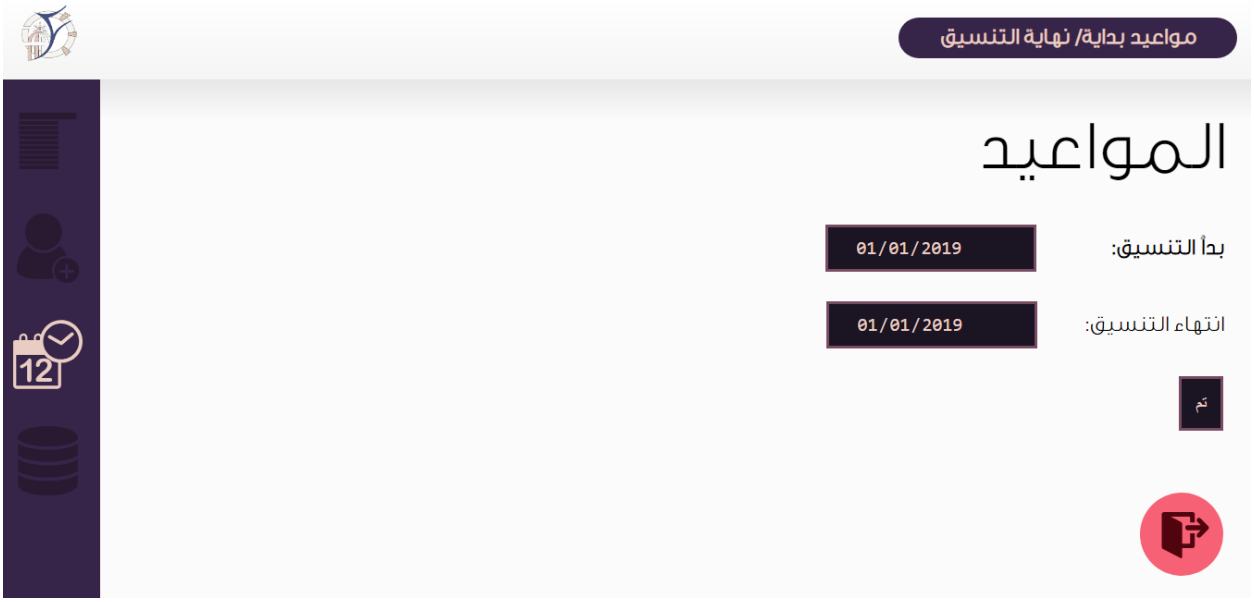
- *l login page requires*
    - *admin user name [to be determined]*
    - *password provided by server manager*



- *home page:*
    - *has side navigation linked to tools he can set*

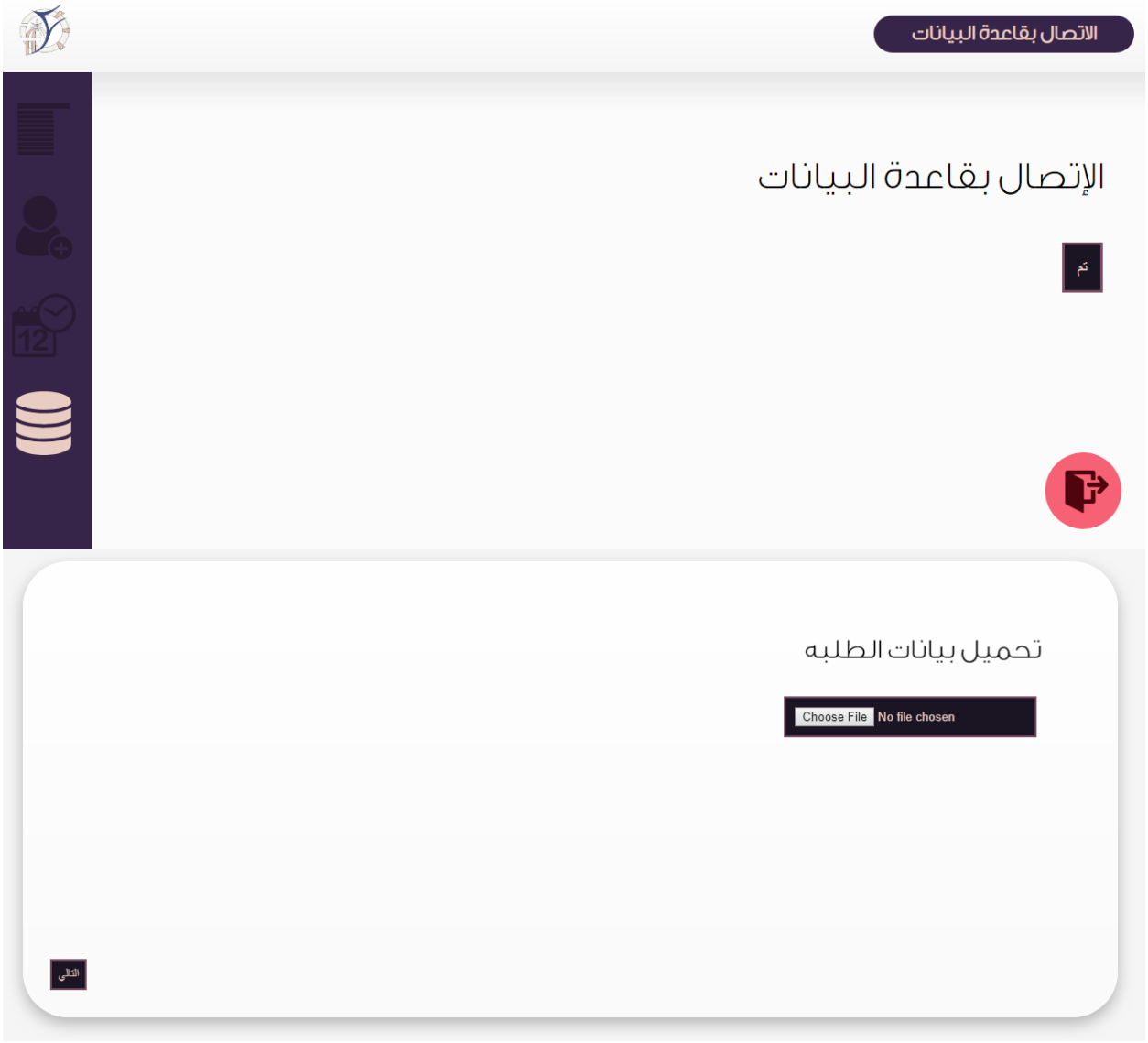
[photo of admin home] [to be provided next version]

- *date setting page*
  - *setting the start and end of date for student to choose departments*



- *department editor*
  - *add, edit or remove a department*
  - *ability to choose specific requirement for department*
  - *** UI for specific requirement will be updated*

- *student data*
  - *upload student data from excel sheet*
  - *or from database*



- *year managing*
    - *add new year with its departments*

[photo] [to be provided next version]

server admin:

- *login page*
- *admin editing page*
    - *adding new faculty admin*

[photo]

error page:

- *if any user entered unavailable link*

3.2 Hardware Interfaces

The application must run over the internet.
The hardware connection to the database server is managed by all operating system.

3.3 Software Interfaces

The database in order gets the information about student.
*The communication between the database and the web portal consist of operation concerning both a reading and modifying the data, while the communication between the database and the operating system consist of only reading operating***.**

3.4 Communications Interfaces

The communication between the different parts of the system is important. they depend each other.
The system shall use the HTTP and HTTPS protocol for communication over the internet.

4. Other Nonfunctional Requirements

4.1 Performance Requirements

server side:

- *we use java as working environment its minimum*

- *minimum of 128MB of memory*
- *disk space 181 MB suggest 1gb*
- *minimum a Pentium 2 266 MHz processor*

*** those are required only by java after finish program may provide more accurate requirement depend on program performance*

user -student/admin- side:

- *user will need only a browser for viewing the web page*
  - *preferred desktop web browser*
  - *stable internet connection*

4.2 Safety Requirements

[none]

4.3 Security Requirements

server admin / faculty admin:

there data stored in private database can only be accessed by server admin through web interface the password will be stored encrypted into database, the authentication model will only compare hash of the password was send from user
both admins have ability to change password

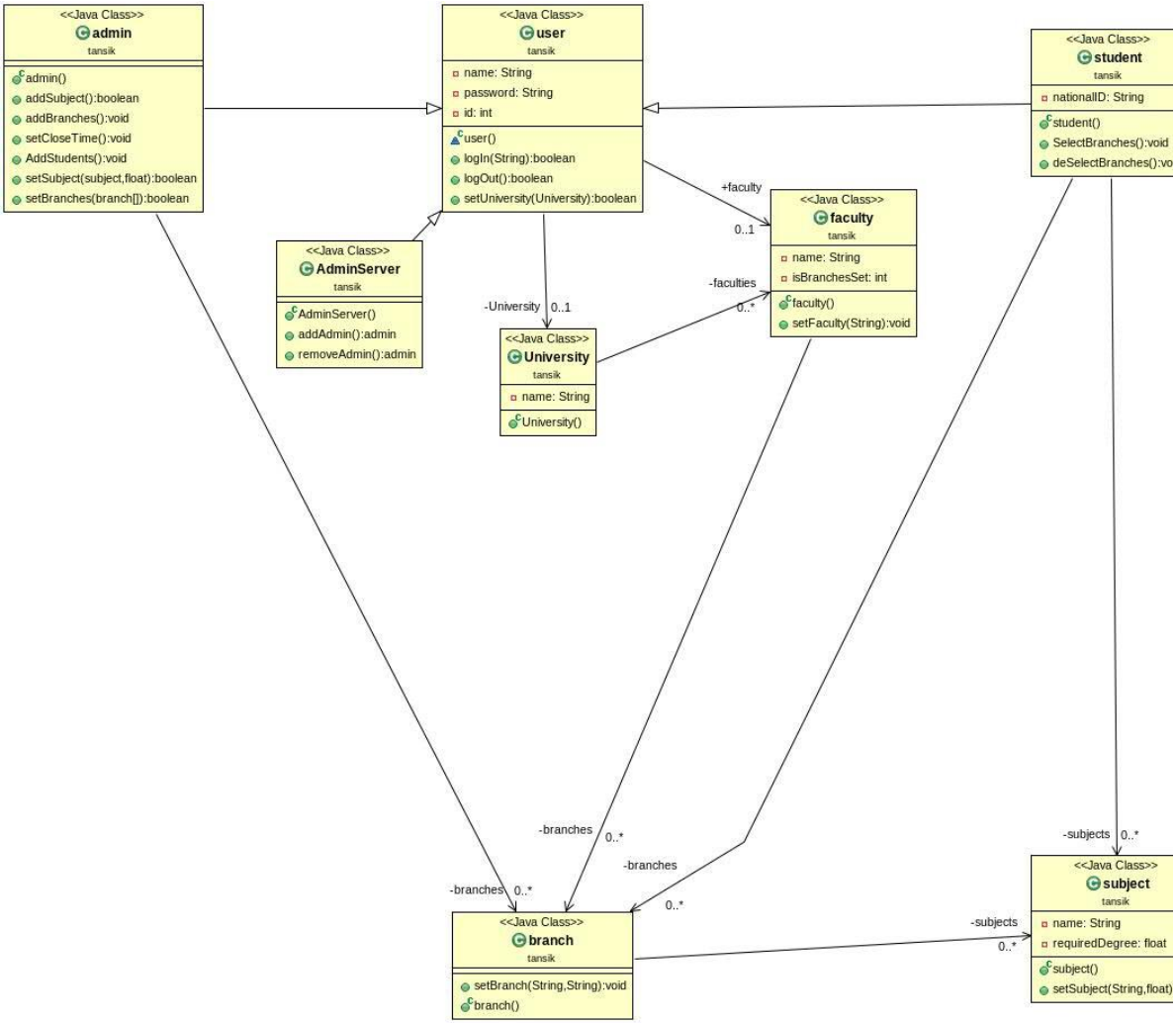
student:

passwords will be generated randomly and send by mail to students [to be determined] and student have same login in interface with ability to change password by faculty admin only

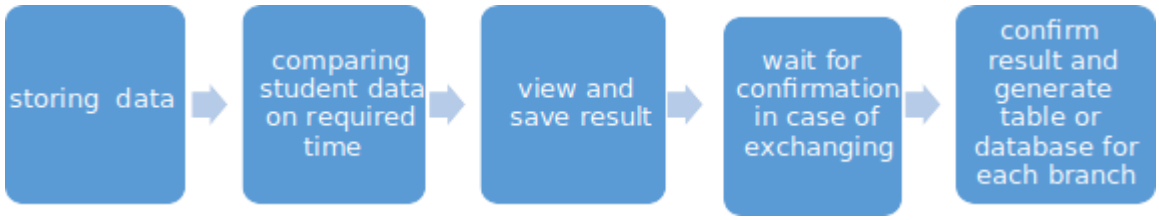
5. Other Requirements

- *UML for the whole program.*
- *Testing and Verification of the software.*
- *Implementing database.*

Appendix B: Analysis Models



UML Diagram for Users



Simple Program Flow

Appendix C: To Be Determined List

[to be added]
[items to be determined has this label]

Appendix D: developer team names

name	note
اسامه محمد عبد التواب رمضان	
اسراء جمال حسن احمد	
ايمان محمود رشوان راشد	
حسين مصطفى سعيد الخولى	
رنا علاء محمد على	
زغلوله عاطف عبدالمولى عبدالوهاب	
عبدالرحمن رجب هاشم اسماعيل	
عبدالله خالد كمال السيد على صيام	
عبدالله محمد ابوالمجد على محمد البسيونى	
محمد ايمن فتحى سيد	
محمد معوض محمود	
مصطفي جمال الدين قرني علي	