Iteration 3 Plan

An overview of the final system requirements

# Users

|  |  |  |
| --- | --- | --- |
| Username | Access Point | Roles and Permissions |
| Scholar | Frontend | * View own records * Create own subjects list * Create own grades records * Edit unapproved own subjects list * Edit unapproved own records * Upload files * Send messages to SM Foundation Incorporated |
| SM Foundation Accounts   * Linda Atayde * Ling Lansang * Tess Samillano * Greg Baclea-an Jr. * Cath Llarena * Jenn Ocampo | Backend | * Admin * Full access to all backend features * Approve scholar-created records * CRUD rights to all records |

**General Note:**

For all functions that require approval, I leave it to the discretion of the programmers involved how to implement the approval notification system. By that I mean whether the one creating the record needs to press a “send for approval” button, OR whether the SM Foundation accounts are automatically notified upon creation of the record that needs approval. Alternatively, the SM Foundation accounts may simply have a screen/function in the backend that tells them whether or not there exist records that have not yet been approved (I think this is the easiest, but I could be wrong).

Regarding names of database columns: the ones listed here are not the verbatim names. The names used merely reflect what that column contains, and are not the same as the names of the actual columns in the database.

# Frontend

* Retain all current features
  + Implement minor changes stated during client meeting
    - Example: Contact SM Foundation should not have a name field – it should be automatically generated in the email based on who is logged in
  + Ensure that all currently implemented features work with the new database
    - All buttons lead to pages
    - All information is generated properly
      * Information generated is in-line with the information contained in the database, and with SM Foundation’s requirements
    - The email feature actually emails
    - The calendar works
      * No changes or additional features to calendar, just the default features that came with the add-on
* Add a new screen for creating a **Subject List**
  + The subject list is the list of all the subjects the scholar will take during his/her schooling
  + The page will have two display types:
    - ***No Existing Subject List***
      * This screen will display if the system detects that there are no records in the **Subject List Table** with the logged in scholar’s number
      * The screen will tell the scholar that they have not yet created a subject list, and that they will be required to create one before they can submit any grades
      * There will be a button on the screen allowing the scholar to create a subject list
      * **Subject List Creation Process:**
        + System will ask the scholar to input the number of terms (integer)
        + Scholar presses a button to confirm input

Store the input in a variable

Based on scholar input, the system will display screens that will allow the scholar to start creating subjects

* + - * + **Subject Creation Screen Contents:**

Header contains the course of the logged-in scholar (retrieved from database) and the term number of the subjects to be created

Example:

Course: Computer Science

Term: 1

Buttons:

Add a subject **(This action may be repeated)**

A new record appears **(same screen)**

Scholar must input values for:

Subject Name

Subject Units

Submit Term Subjects **(This takes scholar to another screen)**

Clicking this button **CREATES** records in the database for each subject inputted containing the following:

Subject Number (PK, auto-increment)

Scholar Number (FK)

Term

Subject Name

Subject Units

Taken Status (default: Not Taken)

Approval Status (default: Not Approved)

Approved By (default: null)

* + - * + After the scholar clicks the button to Submit their entered Term Subjects, the system must check the current term number against the total number of terms input by the scholar

Example a: A Scholar with 12 total terms presses “Submit Term Subjects” for Term 1

System checks if 1 >= 12

Since not true, **a new Subject Creation Screen** is displayed with the next Term number (Term 2 in this case)

Example b: A Scholar with 12 total terms presses “Submit Term Subjects” for Term 12

System checks if 12 >= 12

Since true, **scholar is taken to the *With Existing Subject List* screen**

* + - ***With Existing Subject List***
      * This screen will display if the system detects that there are existing records in the **Subject List Table** with the logged in scholar’s number
      * The screen will display the subjects list records found in the database that match the logged in scholar
        + All records are grouped according to Term Number
        + The scholar will be able to **EDIT** or **DELETE** any record that still has the “Not Approved” status

The scholar may edit the following:

Term

Subject Name

Subject Units

* + - * + The scholar may **ONLY VIEW** records that have already been Approved by the SM Foundation
* Add a new screen for creating and/or editing **Grade Records**
  + Scholars may only create grade records once they have already created a subject list
    - If they have not yet created a subject list, then the screen will tell them to do so
  + SUGGESTION: You may make it so that this screen is accessible from the view grades page via a tab or button
  + This screen will first ask the scholar to select a term
  + Once a term has been selected, all subjects from the **Subjects List** that have a matching term number are displayed
    - If a subject already has a grade record associated with it, then the value of the raw grade is displayed
      * If the grade record has not yet been approved, then the scholar may still edit the value of the grade (i.e. the raw grade they entered)
      * If the grade record has already been approved, then the scholar may only view that grade
    - If a subject does not have any grade record associated with it, then the scholar may enter the following values for that subject:
      * Raw Grade (string)
      * School Year Start (integer)
      * School Year End (integer)
    - Upon confirming the grade value they have entered (they should be required to press a confirmation button), a new grade record is **CREATED** in the database containing the following:
      * Grade Number (PK, auto-increment)
      * Scholar Number (FK)
      * Subject number (FK)
      * School Year Start
      * School Year End
      * Raw Grade
      * Approval Status (default: Not Approved)
      * Approved By (default: null)

# Backend

* Contains only the features listed here
  + Do not retain unlisted features
    - Exception: sub-features that I may have overlooked but are still important
      * Example: search functions on certain pages
  + Feel free to suggest additional features, or point out any that I might have missed
* ***School Records***
  + Here SM Foundation can create, view, edit, or delete any and all existing school records
    - How these records are displayed, I leave to the programmers
    - SUGGESTION: retain the color-coding per area
  + Each school record, regardless of which of the above operations is being performed, must contain the following:
    - School Number (PK, auto-increment)
    - School Name
    - School Area
    - School Address
    - School Contact Email(s)
    - School Contact Number(s)
    - School Vendor Code
  + SM Foundation **MUST** create these records, and provide this information, individually
* **Grade Equivalence Rules**
  + Here SM Foundation can create, view, edit, or delete any and all existing rule records
    - Rule records should be grouped by school
  + Each rule record, regardless of which of the above operations is being performed, must contain the following:
    - Rule Number (PK, auto-increment)
    - School Number (FK)
    - Numerical Grade (decimal(5,2))
    - Letter Grade
    - Percentile Equivalent Lower
    - Percentile Equivalent Upper
    - School Rating
    - Foundation Rating
      * PASS
      * FAIL
  + SM Foundation **MUST** create these records, and provide this information, individually
* ***Scholar Records***
  + Here SM Foundation can create, view, edit, or delete any and all existing scholar records
    - Scholar records **NOT** displayed individually, but by school
      * The schools would be the records SM Foundation would see, and the expanded view would reveal the actual scholar records
  + Scholar records, regardless of which of the above operations is being performed, must contain the following:
    - Scholar Number (PK, auto-increment)
    - School Number (FK)
    - Scholar First Name
    - Scholar Middle Name
    - Scholar Last Name
    - Scholar Gender
    - Scholar Address
    - Scholar Course
    - Scholar Graduate Status
      * Graduated
      * Not Graduated (default)
    - Scholar Year Level
    - Scholar email(s)
    - Scholar Contact Number(s)
    - Scholar Allowance Status (**Do not include this in any CRUD operation of this feature because it has its own section below, simply display it with its default)**
      * Granting (default)
      * Withheld
    - Scholar Cash Card Number
    - Scholar Sponsor
    - Scholar Type
      * With Allowance, Tuition Fee, and Incentive
        + SMFI
        + My Scholar A
        + Kabayan Scholar
      * With Tuition Fee and Incentive (No Allowance)
        + My Scholar B
        + ICA Grant Scholar
  + SM Foundation **MUST** create these records, and provide this information, individually
* ***Grades Records***
  + There is one screen for the grades records.
  + This one screen should have two tabs for viewing purposes:
    - Approved Grades Tab
      * Contains a list of all approved grades records sorted by scholar
        + The same as current implementation (has drop down expand)
        + SM Foundation should be able to **VIEW** the following (at least, feel free to include more information as necessary):

Scholar Number, Name, School

this as the grouping, rest on expanded view

Subject name

Term

School Year Start

School Year End

Raw Grade

Approved By

* + - * + Retain the ability for the SM Foundation to **EDIT** or **DELETE** everything they see on this tab
        + Retain the ability for the SM Foundation to **CREATE** new grades records
        + **Grades Creation Process**

This process is mostly the same as the scholar’s process for creating grade records

SM Foundation must first select a scholar

They should be able to search by scholar name or by scholar id

Then, based on the scholar, they must select a term number

Once a term has been selected, the system will display all available subjects for that term

SM Foundation may then choose to edit the information of any existing records for the subjects of that term, or add values to a subject that has none, namely:

Raw Grade (string)

School Year Start (integer)

School Year End (integer)

Both edit and create actions should require the pressing of a confirmation button. Upon confirmation, the grade record is either edited (if existing) or created (if new). The following data should exist for each grade record based on the selections of SM Foundation:

Grade Number (PK, auto-increment)

Scholar Number (FK)

Subject number (FK)

Term

School Year Start

School Year End

Raw Grade

Approval Status (default: Not Approved)

Approved By (default: null)

These records must still undergo the approval process

* + - Pending Grades Tab
      * Contains a list of all grades records that have not yet been approved sorted by scholar
      * SM Foundation should be able to search for specific records (goes without saying, usually, but is especially important here)
      * SM Foundation **MUST** approve each record individually
      * **Grades Approval Process**
        + SM Foundation selects a grade record

Grades records in this screen must display at least the following information:

Scholar Number

Scholar Name

Scholar School

Subject Name

Term

School Year Start

School Year End

Raw Grade

* + - * + Once a record is selected, SM Foundation is given the following options:

Set Subject Taken Status (From **Subject List Table**)

Default: Not Taken (Cannot be this)

Taken

Failed

Set Approval Status

Default: Not Approved (Cannot be this)

Approved

* + - * + After setting the above options, SM Foundation must confirm their selection (confirm button)

Alternatively, if SM Foundation wishes to reject a grade record (for whatever reason), there should be a button next to confirm to **DELETE** the record

* + - * + Upon confirmation, the system changes the appropriate fields, and the grade record is no longer viewable in the Pending Grades Tab (it can instead be viewed in the Approved Grades Tab)
* **Tuition Records**
  + Here SM Foundation can create, view, edit, or delete any and all existing tuition records
  + Tuition records are grouped by school, but still contain the scholar number, name, type, and sponsor
  + Tuition records must contain the following information:
    - Tuition Number (PK, auto-increment)
    - Scholar Number (FK)
      * It is important to retrieve the following columns from the scholar table:
        + Scholar Name
        + Scholar Type
        + Scholar Sponsor
    - Term
    - School Year Start
    - School Year End
    - Date of Enrollment
    - Tuition Amount
    - Tuition Paid Status
      * Paid
      * Not Paid
    - Date of Payment
  + ***Tuition Creation Process***
    - Tuition records are created by SM Foundation in batches
    - SM foundation first selects a school
      * They should be able to search by school name or by school number
    - After selecting a school, SM Foundation must narrow the selection down further by inputting the following information:
      * Term
      * School Year Start
      * School Year End
    - Term, School Year Start, and School Year End are all stored in variables
      * This information is to be added upon record creation
    - Once that information has been input, all scholars listed under that school are displayed
      * Scholar Number, Name, Type, and Sponsor are displayed
      * On the header of this page are the inputted values for Term, School Year Start, and School Year End
    - Beside each scholar’s information are fields where SM Foundation must input the following information:
      * Date of Enrollment
      * Tuition Amount
      * Tuition Paid Status
        + Paid
        + Not Paid
      * Date of Payment
    - The fields displayed are based off of the inputted Term, School Year Start, and School Year End
    - If any of the records already have information, then SM Foundation may edit them at this screen
    - Both edit and create actions should require the pressing of a confirmation button. Upon confirmation, the tuition record is either edited (if existing) or created (if new).
  + If possible, include the ability to create tuition records individually as well
* **Deduction Records**
  + Essentially the same as current (displayed by scholar, color coded, etc.)
  + Make sure that everything displays properly
  + Make sure that new database columns (if any) are included
* **Allowance Withholding**
  + Very similar to how deductions work in the current system
  + This screen shows a list of all scholars
    - Color coded by area
    - Has the drop down expanded view
    - Expanded view contains the columns from the Withholding Table
  + Only the following columns are shown:
    - Scholar Number
    - Scholar Name
    - Scholar Area
    - Scholar Type
    - Scholar Sponsor
    - Scholar Cash Card Number
    - Scholar Allowance Status
      * Granting (default)
      * Withheld
    - Date Withheld
    - Date Released
    - Remark
  + SM Foundation **MUST** create a new withholding record every time they wish to withhold an allowance (this is primarily for historical functionality)
  + To create (withhold) a withholding record, SM Foundation should click the “Withhold” button (which is just the create record button in disguise)
    - **Withholding Process**
      * SM Foundation must first select a scholar
      * After that, SM Foundation is prompted to select a new Scholar Allowance Status
        + Since this case is for withholding, cannot be “Granting”
      * SM Foundation must then input the Date Withheld
      * Since this is the case for withholding, the Date Released field should not be visible
      * SM Foundation may also input a Remark on why the Scholar Allowance is being withheld
      * SM Foundation hits the confirm button
      * Upon confirmation, a new record is **CREATED** in the Withholding Table containing the following:
        + Withholding Number (PK, auto-increment)
        + Scholar Number (FK)
        + Date Withheld
        + Remark
        + Date Released (default: null)
  + To edit a withholding record, SM Foundation should click the edit button from the drop down expanded view next to the record they wish to edit
    - Editing can simply be a case of correcting a wrong input, but is primarily the way that scholar allowances are released
    - **Releasing Process**
      * SM Foundation is prompted to select a new Scholar Allowance Status
        + Since this case is for releasing, cannot be “Withheld”
      * SM Foundation must input the Date Released (cannot be blank)
      * The system displays the Date Withheld and withholding Remark (if any) as well
        + SM Foundation may edit both, if desired
      * SM Foundation hits the confirm button
      * Upon confirmation, the record is updated
* **Refund Records**
  + Change Name to **Incentive Records**
  + Largely unchanged from current version
    - One minor edit: do not include the SM/Scholar share, only include the amount
* **View Uploads**
  + Same as current upload forms function
* **Calendar**
  + No changes or additional features to calendar, just the default features that came with the add-on
* **Reports**
  + \*This section to be added at a later date\*