(*Italics* indicate glossary item)

AS AN administrator, I WANT to be able to access all data in the system, SO THAT I can remain informed about courses, instructors, and TAs

M: Create *course table*, create *user table*, write test cases, design and implement user interfaces to view the *course table*, *user table*, and individual courses or users.

GIVEN an administrator is logged in
WHEN an administrator selects the <View Courses> option
THEN open an interface displaying the *course table*.

GIVEN an administrator is viewing the *course table*
WHEN an administrator selects a course
THEN open an interface displaying the *course information* for that course.

GIVEN an administrator is logged in
WHEN an administrator selects the <View Users> option
THEN open an interface displaying the *user table*.

GIVEN an administrator is viewing the *user table*
WHEN an administrator selects a user
THEN open an interface displaying the *user information* for that user.

AS AN administrator, I WANT to be able to create courses, SO THAT I can add new courses to the course table.

S: Write test cases, design and implement user interface to enter *course information*

GIVEN an administrator is viewing the *course table*
WHEN the administrator selects the <Add Course> option
THEN open an interface for the user to enter *course information*
WHEN the administrator enters the *course information* for a course that does not already exist (unique course name and course number)
THEN create a new entry in the course table with the *course information* that was provided

GIVEN an administrator is entering *course information* for a new course
WHEN the administrator enters *course information* for a course that already exists (duplicate course name or number)
THEN display an error message that the course already exists and indicate duplicate information

---

AS AN administrator, I WANT to be able to archive courses, SO THAT obsolete courses can be removed from the *course table*.

M: Create a database table for archived courses, write test cases, design and implement user interface to archive a course

GIVEN an administrator is viewing the *course information* for a course
WHEN the administrator selects the <Archive Course> option

THEN display a prompt confirming course archival
WHEN the administrator selects the <Yes> option
THEN add the course to the *archived course table*
AND remove the course from the *course table*

GIVEN an administrator is viewing the prompt to confirm course archival
WHEN the administrator selects the <No> option
THEN close the confirmation prompt

AS AN administrator, I SHALL be able to send out notifications to all users via UWM email, SO THAT I can easily notify the entire department without looking through every single user's contact information.

M: Write test cases, design and implement an interface for an administrator to write their notification message and send it to all other users' UWM email addresses

---

GIVEN an administrator is logged in WHEN the administrator selects the <Publish Notification> option THEN display a window for them to select which roles should be notified AND to enter a notification title AND to enter notification message WHEN the administrator selects the <Send> option THEN compose an email from the administrator's UWM email address to themselves AND bcc the UWM email addresses of all users with the previously specified roles from the user table AND use the notification title as the email subject AND use the notification as the email body AND send the emaill

---

AS AN administrator, I WANT to be able to create new user accounts, SO THAT new instructors and TAs can be added to the *user table*

S: Write test cases, design and implement interface to enter new *user information*

GIVEN an administrator is viewing the *user table*
WHEN the administrator selects the <Create User> option
AND the administrator enters *user information* with a unique ID number
THEN add a new user to the *user table* with the provided *user information*

GIVEN an administrator is viewing the user table
WHEN the administrator selects the <Create User> option
AND the administrator enters an ID number that already exists
THEN display a message saying that a user already exists with that ID number

---

AS AN administrator,I WANT to be able to edit user accounts, SO THAT I can update user information as needed

---

S: Write test cases, design and implement an interface for changing *user information*

GIVEN an administrator is viewing a user from the *user table*

WHEN the administrator selects the <Edit Account> option
THEN display a window showing current *user information* and allowing the user to change that information
WHEN the administrator selects the <Save> option
THEN update the *user information* with the new information and close the editing window

---

AS AN administrator,I WANT to be able to delete user accounts, SO THAT when a user can be removed from the system when they quit/retire/are fired/die

M: Write test cases, design and implement an interface for account deletion

GIVEN an administrator is viewing a user from the *user table*
WHEN the administrator selects the <Delete Account> option
THEN display a prompt confirming account deletion
WHEN the administrator selects the <Yes> option
THEN display a second prompt confirming account deletion
WHEN the administrator selects the <Yes> option
THEN remove the user from all tables (*user table*, *assigned TAs table*, etc)

GIVEN an administrator is viewing either prompt confirming account deletion
WHEN the administrator selects the <No> option
THEN close the prompt

AS AN administrator, I WANT to be able to create course sections, SO THAT I can have multiple instructors teach the same course in a semester.

M: Write test cases, create *course sections table*, design and implement an interface to create a new section of a course.

GIVEN an administrator is viewing a course
WHEN the administrator selects the <Add Section> option
THEN display a window to enter *section information*
WHEN the administrator enters unique *section information* (i.e. section number)
THEN create the new section and add it to the *course sections table* of the *course information*

GIVEN an administrator is viewing a course
WHEN the administrator selects the <Add Section> option
THEN display a window to enter *section information*
WHEN the administrator enters a duplicate *section information* (i.e. section number)
THEN display an error message indicating the duplicate *section information*

AS AN administrator, I WANT to be able to delete course sections, SO THAT after a semester ends, I can remove any sections that will not be offered the next semester.

S: Write test cases, design and implement an interface for deleting a section from a course.

GIVEN an administrator is viewing a course from the *course table*

WHEN the administrator selects a section from the section table of the *course information*
AND the administrator selects the <Remove Section> option
THEN display a prompt confirming section removal
WHEN the administrator selects the <Yes> option
THEN remove the section from the *course sections table* of the *course information*

GIVEN an administrator is viewing the prompt to confirm section removal
WHEN the administrator selects the <No> option
THEN close the confirmation prompt

AS AN administrator, I WANT to be able to assign TAs to a course, SO THAT TAs can see which courses they are assigned to and Instructors can assign TAs to sections within their courses.

M: Write test cases, create *assigned TAs table*, create and implement an interface to add a TA to a course

GIVEN an administrator is viewing a course
WHEN the administrator selects the <Add TA> option
THEN display only the TAs from the *user table* minus the *assigned TAs table* of the *course information*
WHEN the administrator selects a TA
AND enters their information for the *assigned TAs table* (number of labs and grader status)
THEN add the selected TA to the *assigned TAs table* of the *course information*
AND add the course to the selected TA's *assigned courses table*

AS AN administrator, I WANT to be able to remove TAs from a course, SO THAT TAs that are no longer affiliated with a course can be removed for clarity.

M: Write test cases, create and implement an interface for removing TAs from a course

GIVEN an administrator is viewing a course
WHEN the administrator selects a TA from the *assigned TAs table* of the *course information*
THEN display the TA's *user information* and *assigned sections table*
WHEN the administrator selects the <Remove TA> option
THEN display a prompt confirming TA removal from the course
WHEN the administrator selects the <Yes> option
THEN the TA is removed from the course and all sections of the course

AS AN administrator or instructor, I WANT to be able to assign TAs to a lab or discussion section, SO THAT each lab has a TA teaching it.

M: Write test cases, create *assigned TAs table*, create and implement an interface to add a TA to a course

GIVEN an administrator or instructor is viewing a lab or discussion section from the *course sections table* of a course's *course information*

WHEN the administrator or instructor selects the <Assign a TA> option
THEN display the *assigned TAs table* of the *course information*
WHEN the administrator or instructor selects a TA
THEN add the selected section to the selected TA's *assigned courses table*

AS AN instructor, I SHALL be able to send out course notifications to all the TAs of a given course via UWM email, SO THAT I can easily notify all of my TAs at once without the risk of forgetting to include one.

M: Write test cases, design and implement an interface for an instructor to send out a course notification

GIVEN an instructor is viewing one of their assigned courses WHEN the instructor selects the <Course Notification> option THEN display a window for them to enter a notification title AND to enter a notification message WHEN the instructor selects the <Send> option THEN compose an email from the instructor's UWM email address to the UWM email addresses of all TAs from the course's assigned TAs table AND use the notification title as the email subject AND use the notification as the email body AND send the email

---

AS AN instructor or TA, I WANT to be able to view all TA assignments for a course, SO THAT I know who I am working alongside.

S: Write test cases, create an interface to display the *assigned TAs table*

GIVEN an instructor or TA is viewing a course from their *assigned courses table*
WHEN the instructor or TA selects the <Assigned TAs> option
THEN display the *assigned TAs table* of the *course information*

AS AN instructor or TA, I WANT to be able to edit my contact information, SO THAT I can update the information if it changes and still be contacted.

M: Write test cases, design and implement an interface for Instructors to view and edit their contact information

GIVEN an instructor or TA is logged in
WHEN the instructor or TA selects the <My Contact Information> option
THEN display a window with the user's contact information
WHEN the instructor or TA selects the <Edit Contact Information> option
THEN allow them to edit the contact information in each field
WHEN the instructor or TA enters valid contact information (i.e. a 10 digit number for a phone number or a string including @ for an email address, etc)
THEN update their contact information with the provided information

GIVEN an instructor or TA is entering new contact information
WHEN the instructor or TA enters invalid contact information (i.e. a non-numerical characters

for a phone number or a string without an @ for an email address, etc)   THEN display an error indicating the invalid contact information

AS AN instructor or TA, I WANT to be able to view my course assignments, SO THAT I know what my assignments are each semester.
S: Write test cases, create an *assigned courses table*, design and implement an interface to display the assigned courses table
GIVEN an instructor or TA is logged in   WHEN the instructor or TA selects the <My Courses> option   THEN display the *assigned courses table* of the *user information* for that instructor or TA

AS AN instructor or TA, I WANT to be able to view public contact information of all users, SO THAT I can contact anyone I need to.
M: Write test cases, design and implement an interface that allows the instructor to view the *user table* but prevents them from viewing private or administrator-only information
GIVEN an instructor or TA is logged in   WHEN the instructor or TA selects the <View Contacts> option   THEN display the *user table*   WHEN the instructor or TA selects a user   THEN display that user's public contact information

Glossary:
- *Course Table*: Displays course name and course number. Filterable by course name and number.
- *User Table*: Displays user's first name, user's last name, and user's role. Filterable by last name and role. (Administrator only: filter by ID number)
- *Course Information*: Course name, course number, *course sections table*, *assigned instructors table*, *assigned TAs table*, etc.
- *User Information*: First name, middle initial, last name, role, UWM email, *assigned courses table*, etc. (Administrator only: Private information, i.e. ID number, phone number, home address)
- *Archived Course Table*: Displays all deleted courses from the last semester.
- *Section Information*: Course name, section number, section type (lab, lecture, discussion, etc), instructor, TA, etc
- *Course Sections Table*: Displays each section of a course
- *Assigned TAs Table*: Displays each TA assigned to a course, their number of labs, and their grader status
- *Assigned Instructors Table*: Displays each Instructor assigned to a course
- *Assigned Courses Table*: Displays the courses and sections that a user is assigned to