TP1 Sprint Backlog

Sprint window: ~01/29/26 to 02/11/26   **Document date:** 02/01/26

New User Stories

ID	User Story / Requirement	Acceptance Criteria	Status	Owner(s)	Notes
NU-01	As the first user of the system to log in, I can establish my username, password, and account information using a special screen for the first user. As the first user, I am assigned the role of an admin. Once I have finished providing the required account information, I'm required to log in again.	Detect first run (no users). Show first-user setup screen. Validate inputs. Create account and force logout/re-login. Assign admin role to the first user.	Done	N/A	Essentially implemented by default – just connected more within the whole thing -Sutton
NU-02	As a user of the system, I can be assigned one or more user roles so the various functions of the system can be performed, some of which require special knowledge, skills, and authorizations.	Support roles (admin, student, staff). Persist role membership per account (one or more). Enforce authorization checks in UI + service layer.	Backlog	All	
NU-03	As a user of the system with only one role assigned, I can start playing my role immediately after logging in (i.e., I'm taken to that role's home screen).	If user has exactly 1 role, route directly to that role home screen after successful login (no role selection screen).	Backlog	N/A	Implemented by default
NU-04	As a user of the system with more than one role assigned, I can select which role I want to play upon logging in (i.e., I'm then taken to that role's home screen).	If user has 2+ roles, show role selection after login. Selecting role routes to corresponding home screen.	Backlog	Sutton	
NU-05	As a user of the system, I can log out when I'm finished using the system or when I wish to change roles.	Logout ends session and returns to login screen. If multi-role user, logout enables re-login to pick a different role.	Done	Ahad	

Admin Stories

ID	User Story / Requirement	Acceptance Criteria (summary)	Status	Owner(s)	Notes
ADM-01	As an admin, I can invite users to join the system and play one or more roles. This invitation uses a one-time code and includes a deadline for joining the system.	Admin generates invite with roles + expiration date/time. System validates code + deadline. Code is one-time use (invalid after successful join).	Done	Sutton	
ADM-02	As an admin, I can set a one-time password for a user who has forgotten their password... Once a user has logged in with the one-time password, it is cleared... After establishing a new password, the user is required to log in again.	Admin sets temp password. User can login with temp password once. On login, require new password set. Clear temp password after use. Force logout/re-login after password change.	Done	Rufat	
ADM-03	As an admin, I can "delete" a user account... An admin is not allowed to remove that admin's access. When deleting an account, an 'Are you sure?' message must be answered with 'Yes' to remove access.	Soft-delete/disable account (no longer can log in). Confirmation prompt requires explicit 'Yes'. Prevent admin from deleting self. Preserve data per project expectations.	Done	Ahad	
ADM-04	As an admin, I can list all the user accounts... list shows each account's username, the user's name, the user's email address, and the roles the user plays.	Admin-only list view/table. Columns: username, name, email, roles. Includes disabled accounts if required (note behavior).	Done	Ahad	
ADM-05	As an admin, I can add or remove roles for each user... An admin may not remove the admin role from that admin's account. There must always be at least one user able to perform the admin role.	Role management UI. Enforce: cannot remove admin from self; cannot remove last remaining admin role from system. Persist updates immediately.	Done	Rufat	

Notes / Open Questions:

We kinda forgot about this one. All statuses updated to done. (2/11)