

Project 02: Savnac
CST338-40_2252 - Spring 2025

Savnac

Savnac is a somewhat faithful recreation of the canvas.instructure.com (where teachers can create their own courses) gradebook system. There are two types of users: Students and Teachers. Anyone can create either a Teacher account or a Student account. Teachers have the ability to create courses and assignments in said courses, and Students can enroll themselves into courses that Teachers create. There is no limitation on how many courses a Teacher can create, and there is no limitation on how many courses a Student can join.

Github: <https://github.com/vmc0823/Savnac>

Table of contents

Completed Rubric Items	2
Initial Layout	3
Entity Relationship Diagram (ERD)	4
Team members	5
Brandon Evans	5
Wootark Kim	5
Michael Sorensen	5
Valentina Waltman	5
User Stories	6

Completed Rubric Items

Please **Highlight** the items you have completed (Points from highlighted sections: 134 + 70 = **204**)

Item	The following are Required	Score
Persistence	Minimum 3 tables implemented using the Room database wrapper If you do not use the Room Database wrapper the project will be marked as 0.	30
Github issues	There must be at least 5 GitHub issues PER TEAMMATE	10
Presentation	Each application must have a video showing its functionality. The video must be between 6 – 8 minutes long. Each person must present their portion for 1-2 minutes	35
Highlighted Rubric	This rubric must be included in the video AND the submission. All completed tasks must be highlighted	5
Activities	Each team member must create 2-3 activities, one of which must interact with the database.	12
Activity: Login Page	An activity where a username and password are entered. The username and password must be stored in the local room database.	4
Activity: Landing Page	After successful login an activity should start that displays the username and some other information retrieved from the local database.	4
Admin landing page	If the user is an administrator the landing page must indicate this with additional options. The same Activity may be used for both admin and non-admin users.	4
Github	Code must be pushed to Github in a public repository. Each teammate must have at least 3 branches, with 5 commits EACH; each branch must be merged into main using a pull request.	10
Unit tests	Each teammate must contribute at LEAST 2 unit tests. The tests must pass.	20

Commented [1]: Savnac uses a Room database with five unique tables: users, enrollments, courses, assignments, and grade_entries.

Commented [2]: <https://github.com/vmc0823/Savnac/issues>
Brandon: 7/5 issues
Michael: 5/5 issues
Valentina: 6/5 issues
Wootark: 6/5 issues

Commented [3]: Individual Videos
Brandon: https://drive.google.com/file/d/1om-fcLxnXaJpRZDgn0nqk44r564d9_Lw/view?usp=sharing
Michael: <https://drive.google.com/file/d/1rUlfzUHihCCZvqs5TBB7SgnOYKFXut-U/view?usp=sharing>
Valentina: <https://drive.google.com/file/d/1Qn-5xXw7YeOn1eL8j25YPJ7u3q2sB3th/view?usp=sharing>
Wootark: https://cdn.discordapp.com/attachments/1317567739061338223/1365912920105549854/demonstration_video.mp4?ex=680f0909&is=680db789&hm=b9337076ccf8b0fca20410d68faefa530a0f3e214cb409cabb662e87f4ab901b&

Finalized Video: [soon]

Commented [4]: Completed tasks are highlighted in green but video is still being made

Commented [5]: I am pretty sure that every single one of our activities interact with the database lol. Every team member has at least 2 activities.

Commented [6]: LoginActivity meets this requirement.

Commented [7]: CoursesActivity meets this requirement. Has an Account button that opens a dialog with account information and a logout feature.

Commented [8]: Landing page features are dynamic based on the type of account (Student or Teacher) Teacher accounts are de-facto admins in this application

Commented [9]: <https://github.com/vmc0823/Savnac/>

Commented [10]: Brandon: 11/2 Unit Tests (SavnacUserDaoTest and IntentFactoryTest)
Michael: 3/2 Unit Tests (SavnacGradeDaoTest)
Valentina: 6/2 Unit Tests (SavnacEnrollmentDaoTest and SavnacCourseDaoTest)
Wootark: 3/2 Unit Tests (SavnacAssignmentDaoTest)

Optional Item	The following are Optional	Score
Intent Factories	All Intents to start an activity must use the factory pattern Either each activity must have an Intent factory method OR use a Factory class.	10
Intent Tests	At least 1 unit test per activity transition. That is each intent must have a unit test.	20
Database Tests	Each database table must have at least <u>three tests</u> (insert, update, delete)	20
Recycler View	At least 1 activity has a <u>recycler view</u> for displaying information	10
Livedata	Incorporate <u>liveData</u> in your project. This dovetails nicely with adding a recycler view	10
API Integration	Use <u>Retrofit</u> to consume an external API	20
External Library	Use an external Library. <u>Here</u> is a list of examples. <u>Retrofit</u> is included and would count for this.	16
Fragments	Incorporate <u>fragments</u> into your application	8
OAuth	Implement login with OAuth 2. This can replace the login procedure described above	16

Commented [11]: <https://github.com/vmc0823/Savnac/issues/61>
All activities have and use Intent Factory methods.

Commented [12]: These tests has been implemented, see this pull request
<https://github.com/vmc0823/Savnac/pull/74>

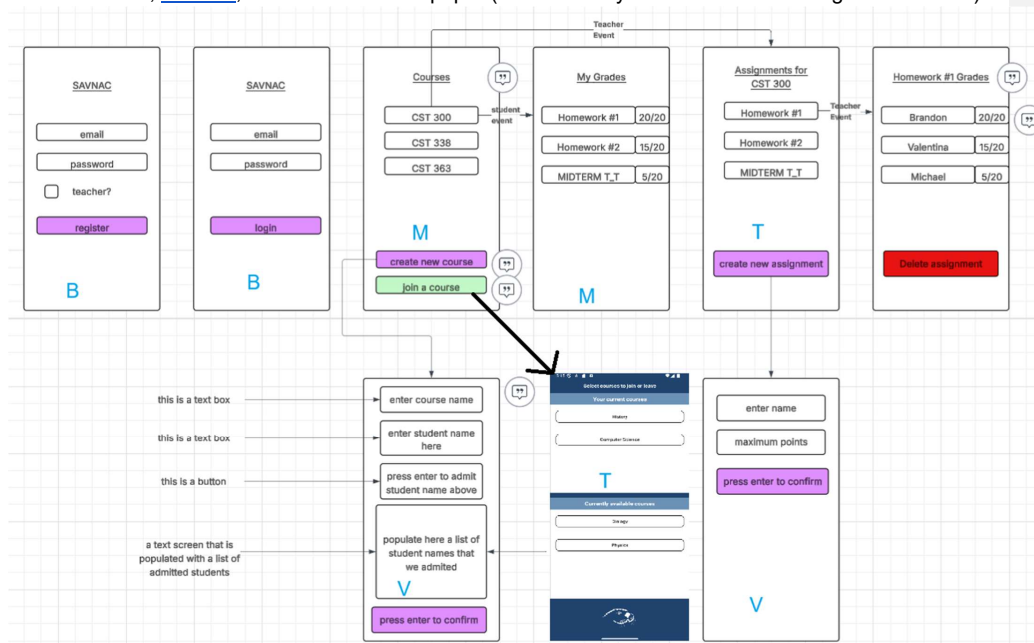
Commented [13]: The following classes satisfy this requirement:
SavnacUserDaoTest (made by Brandon)
SavnacGradeDaoTest (made by Michael)
SavnacEnrollmentDaoTest (made by Valentina)
SavnacCourseDaoTest (made by Valentina)
SavnacAssignmentDaoTest (made by Wootark)

Commented [14]: Most activities utilize a recycler view but only one is required for these points. CoursesActivity is one that makes use of it. (shown in video)

Commented [15]: LiveData is used in the project and tandems with Recycler View requirement

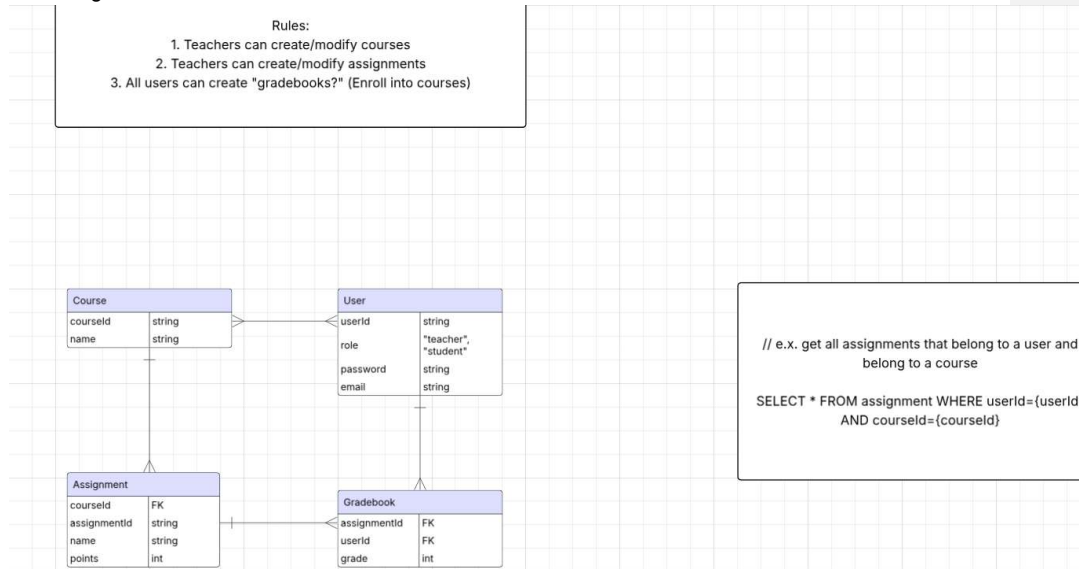
Initial Layout

Include a mock layout similar to the one shown below. This can be created using screenshots from Android Studio, [Draw.io](https://draw.io), or sketched out on paper (or a tablet if you are fortunate enough to have one).



Entity Relationship Diagram (ERD)

The assignment MUST use a database and must have at least three tables



Team members

Brandon Evans

- [Link to Brandon's issues on GitHub](#)
- [Link to Brandon's commit history to main branch on GitHub](#)
- Branches merged into main: AuthenticationActivities, LoginActivity, SignupActivity, LogoutBranch, currentUserFix, IntentFactoriesBE, UnitTestsBE. All of these branches have at least five commits

Wootark Kim

- [Link to Wootark's issues on GitHub](#)
- [Link to Wootark's commit history to main branch on GitHub](#)
- Branches merged into main: assign_grades_teacher_exclusive_page, show_list_of_active_assignments, ActivityJavaFileForJoinOrLeaveCoursesTeacherPerspective, Course_join_and_GradeDisplayPage, Enrollment-entity-and-dao

Michael Sorensen

- [Link to Michael's issues on GitHub](#)
- [Link to Michael's commit history to main branch on GitHub](#)
- Branches merged into main: connecting_activities, courses_activity, courses_and_grades_activity, grades_activity, misc_fixes, misx_fixes2, misc_fixes3, tests

Valentina Waltman

- [Link to Valentina's issues on GitHub](#)
- [Link to Valentina's commit history to main branch on GitHub](#)
- Branches merged into main: BackButtonCourses, Course&AssignmentExtensions, CourseExtAdapter, DB_2, Fixes3, fixesRepo, fixesRepo2, GitignoreTemplates, GradleDependencies, LoginActivity, SavnacDB, TestingVW

User Stories

Abilities of Students AND Teachers:

- Should be able to login to Savnac
- Should be able to create a Savnac account
- Should be able to select a course from the home page

Abilities of Students ONLY:

- Should be able to join a course
- Should be able to leave a course
- Should be able to view THEIR gradebook for a selected course

Abilities of Teachers ONLY:

- Should be able to create a course (which also makes the teacher join the course)
- Should be able to update a course
 - Should be able to create assignments
 - Should be able to delete assignments
 - Should be able to update assignments
 - Should be able to grade each student on a particular assignment

Issues

- [Click here to see the Issues list on our GitHub repository](#)