

Project 0

For Project 0, you will be building a console-based application that will be presented on Wednesday, February 9th, 2022. For this application you may choose one of the following concepts:

- a banking application
- course registration application
- any other trainer-approved idea that is similar in functionality to the above project ideas (explicit trainer approval required for this)

Banking Application Details

User Stories

As a user, I can:

- register a new user account with the system (must be secured with a password)
- login with my existing credentials
- create at least one account
- deposit funds into an account (use doubles, not ints)
- withdraw funds from an account (no overdrafting!)
- view the balance of my account(s) (all balance displays must be in proper currency format)

Suggested Bonus User Stories

As a user I can:

- view the transaction history for an account
- create multiple accounts per user (checking, savings, etc.)
- share a joint account with another user
- transfer money between accounts

Course Registration Application Details

User Stories

As any kind of user, I can:

- login with my existing credentials

As a faculty member I can:

- add new classes to the registration catalog
- change the registration details for a class
- remove a class from the registration catalog (this should unregister all registered students)

As a student, I can:

- register a new account with the system (must be secured with a password)
- view classes available for registration
- register for an open and available class
- cancel my registration for a class (if within window)
- view the classes that I have registered for

Custom Project Concept Details

You are welcome to come up with your own project concept if you would like. The basic requirements of the application are as follows:

- Persist data into a relational database
- Allow users to interface (via the console) with an application that allows for them to create, read, update, and delete that persisted data
- Complies with the minimum features (specified below)
- Minimum number of user stories: 7

Send your project concept descriptions to your trainer via Slack and await concept approval. Additionally, please specify the following when writing up your pitch:

- Application Name
- High Level Description
- User Story List

Minimum Features

All project concepts must display the following domain-independent features/requirements:

- Use of custom data structures (do not use java.util Collection types!)
- Basic validation of user input (e.g. no registration for classes outside of registration window, no negative deposits/withdrawals, no overdrafting, etc.)
- Unit tests for all business-logic classes
- All exceptions are properly caught and handled
- Proper use of OOP principles
- Documentation (all classes and methods have basic documentation)
- Database is 3rd Normal Form Compliant
- Referential integrity (e.g. if a class is removed from the catalog, no students should be registered for it)
- Logging messages and exceptions to a file using a custom logger
- Generation of basic design documents (e.g. relational diagram, class diagram, flows, etc.)

Tech Stack

Regardless of the project concept you choose, you will be required to adhere to the following tech stack:

- Java 8
- Apache Maven
- PostgreSQL (hosted on Azure PostgreSQL Server)
- Git SCM (hosted on GitHub)

Repo Instructions

- Create a new repository within this organization team and (naming convention: `firstname_lastname_p0`)
- Keep this repository up to date with frequent commits and pushes (follow best practices!)
- It is recommended (though not required) that you implement some kind of branching strategy when implementing new features into your application

Presentation

- finalized version of application must be pushed to personal repository within this organization by the presentation date
- 5 minute live demonstration of the implemented features