EECS 2311

Software Development Project

Click to edit Master text styles
Second level
Third level
Fifth level

January 14, 2020

YORK UNIVERSITÉ UNIVERSITY

Our project

- Venn diagrams are a great way to present relationships between sets of objects, such as set intersection or set difference
- Many different ways to draw Venn diagrams exist
- We will develop a desktop app that can draw **customizable Venn diagrams**
- Let's look at an online example (link on course website)

Intentionally vague requirements

- In a real software development project, requirements are vague and ever-changing
- The exact requirements will be refined iteratively by interacting with the “customer” on an **ongoing basis**
- Some requirements may change after the project has started
- New requirements may be added after the project has started

Workload

- This course requires 8-10 hours per week per student
- Have to start working immediately
- In the second part of each lecture, there will be time to present your progress to the “customer” and ask questions

Evaluation (individual)

- 20% - Weekly lab tasks completed
- 20% - 4 peer assessments
  - For each peer assessment, you will receive either a document or a prototype from another group to provide feedback on
  - Your grade in the peer assessment will be based on the quality of the feedback you provide

Evaluation (group-based)

- 20% - Midterm submission (due Feb 23)
- 40% - Final submission (due April 6)
- Each group submission will receive a grade based on its merit. Individual grades may be less if full participation has not been demonstrated.

Groups

- Groups are assigned randomly by the “manager”
- As enrollment in the course changes in the first few weeks, the “manager” may rearrange the groups
- Same as a real software project!

Class “exercise”: Find your group

- Login to the EECS 2311 page on Moodle (use your Passport York credentials)
- Find your group number
- Go to the designated desk in the classroom to find your teammates
- Introduce yourselves and exchange contact information

To get started

- Make sure you understand what the project is about
  - Read up on Venn diagrams online
  - Research what features online Venn diagram tools offer
- Setup your group's github repository and make sure every group member can push / pull
- Start coding your GUI

All group members

- Go to github.com and sign up for an account
- If you already have a github account, you can use it for the course
- However, if your existing github account has a username that has nothing to do with your name, you might want to create another account for the course
- **Your participation in the project will be assessed based on your github activity!**

All group members

- Open Eclipse
- Go to Window → Preferences → Team → Git → Configuration
- Click Add Entry, add the pair
  [**`user.name`**, your name]
- Click Add Entry, add the pair
  [**`user.email`**, your email]
- These **must** be the same as the ones used at github.com
- Click Apply, then OK

All group members

- In this course, we will use Gradle to build and deploy our system
- We will discuss Gradle in more detail later, but for now, we need to install Gradle and create our project as a Gradle project
- Search for Buildship Gradle in the Eclipse Marketplace and install it, if it is not already installed

Once per group

- Sign in to one group member's github account
- Create a new repository
- Copy the URL to access your repository
- Click on Settings → Collaborators and add the remaining group members as collaborators
- Your group repository is ready to receive the first version of your software

Once per group

- In Eclipse, click on File → New → Project...
- Select Gradle Project, and click Next
- Name your project **`Venn`**, and click Next and Finish until the project is created
- Under **`src/main/java`**, create a package called **`venn`** containing a class called **`Main`**
- Add a **`main`** method to the **`Main`** class, that prints something to the console

Once per group

- Rightclick on the Venn project
- Select Team → Share Project…
- Select Git, and hit Next
- Click on Create…
- Provide a name for your local repository, and click Finish
- Your **local** repository is now set up

Once per group

- Rightclick on the Venn project, then select Team → Commit…
- Add a commit message
  - It is important that you add a meaningful message every time you commit, makes it much easier to find a version later
- Select all files, click on plus sign to stage them
- Click Commit
- The first version of your software has been added to the local repository

Once per group

- Rightclick on the Venn project, select Team → Remote → Push…
- Copy the URL from github.com in the URI field
- Enter your github.com username and password, click Next
- Click on Add All Branches Spec
- Click Finish, then OK
- You should be able to see the Venn project in github.com

Remaining group members

- Go to File → Import → Git → Projects from Git
- Click Next, select Clone URI, click Next
- Copy the URL of your repository from github.com on the URI field
- Keep clicking Next, and finally Finish
- You now have a copy of the project in your local repository

Push

- Make some changes to any of the classes in the project
- Rightclick on any element that has changes (could be the whole project), and select Team → Commit
- Add a commit message
- If you do not want to publish the changes yet, click Commit
- If they are ready to be published, click Commit and Push

Pull

- To get changes published by other group members, rightclick on the project, and select Team → Pull
- Before starting to make changes to the code, it is important to Pull, so that you are working on the latest version

Lab Task

- Get git working for every group member
- This should be for code / documents / notes etc.
- Demonstrate that everybody can pull / push code on Monday's lab
- Spend the rest of your time coding the GUI of your app
  - In a couple of weeks, you'll have to demo a first prototype