Happy Go Lucky Introduction

Dirk Riehle, FAU Erlangen

ADAP C01

Happy Go Lucky Vision

Happy Go Lucky (HGL) is

- A web app to support our project based teaching

Original HGL solutions are

- Happiness index
- Standup emails
- Code tracking

Setup and Test of Happy Go Lucky

Fork happy-go-lucky to your account, e.g. friedalex

```
git clone git@github.com:friedalex/happy-go-lucky.git
cd happy-go-lucky
npm install
npm run build
npm run test
npm run generate-mockdata
npm run test
npm run dev
```

Happy Go Lucky Base Design

HGL is a web app to support our project based teaching

An admin (professor) can create courses (by semester)

A course can have one or more projects associated with it

A course has an associated schedule (homework delivery dates)

A project can have one or more members (ADAP = 1, AMOS = 6..12)

A project is linked to exactly one GitHub repository

ADAP and AMOS

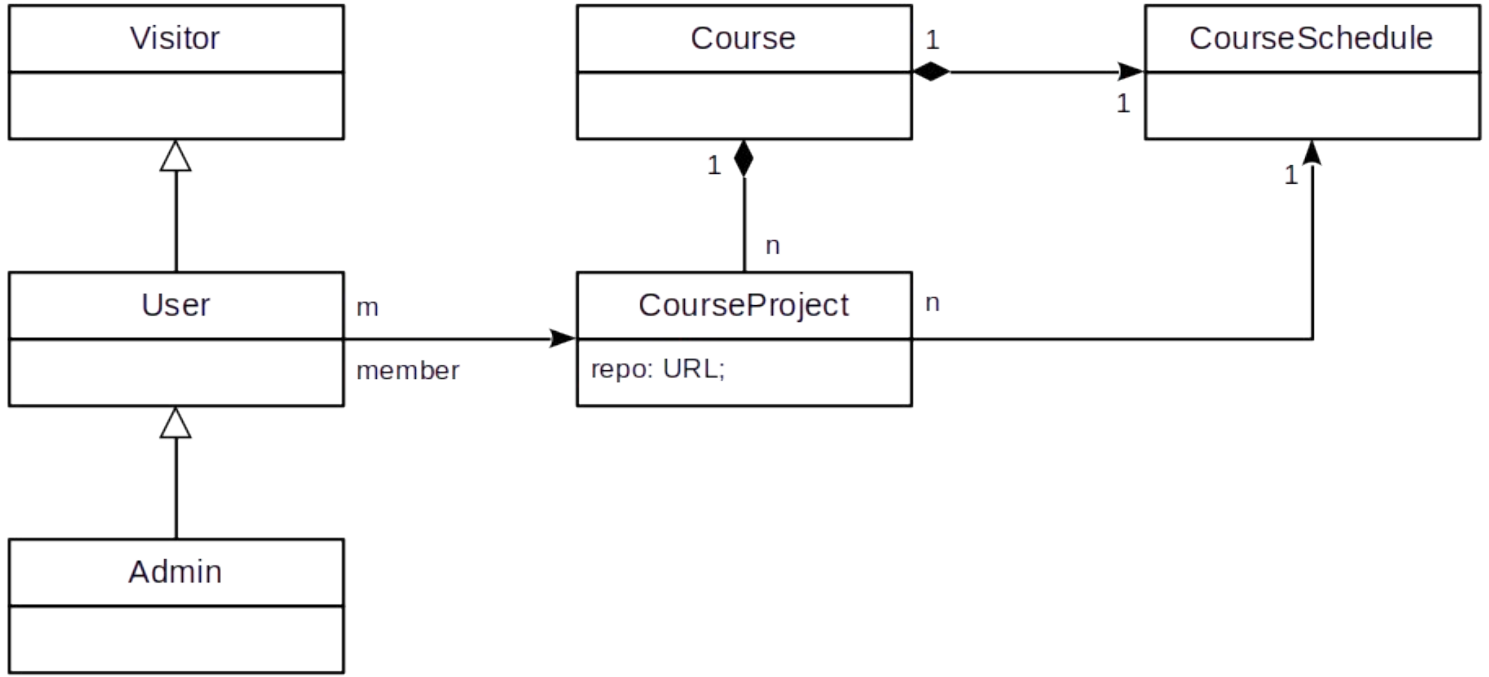
ADAP

- One course for a given term
- Each student their own project
- Students can create their project

AMOS

- One course for a given term
- Small number of set projects
- Students join existing project

Class Model (Logically)



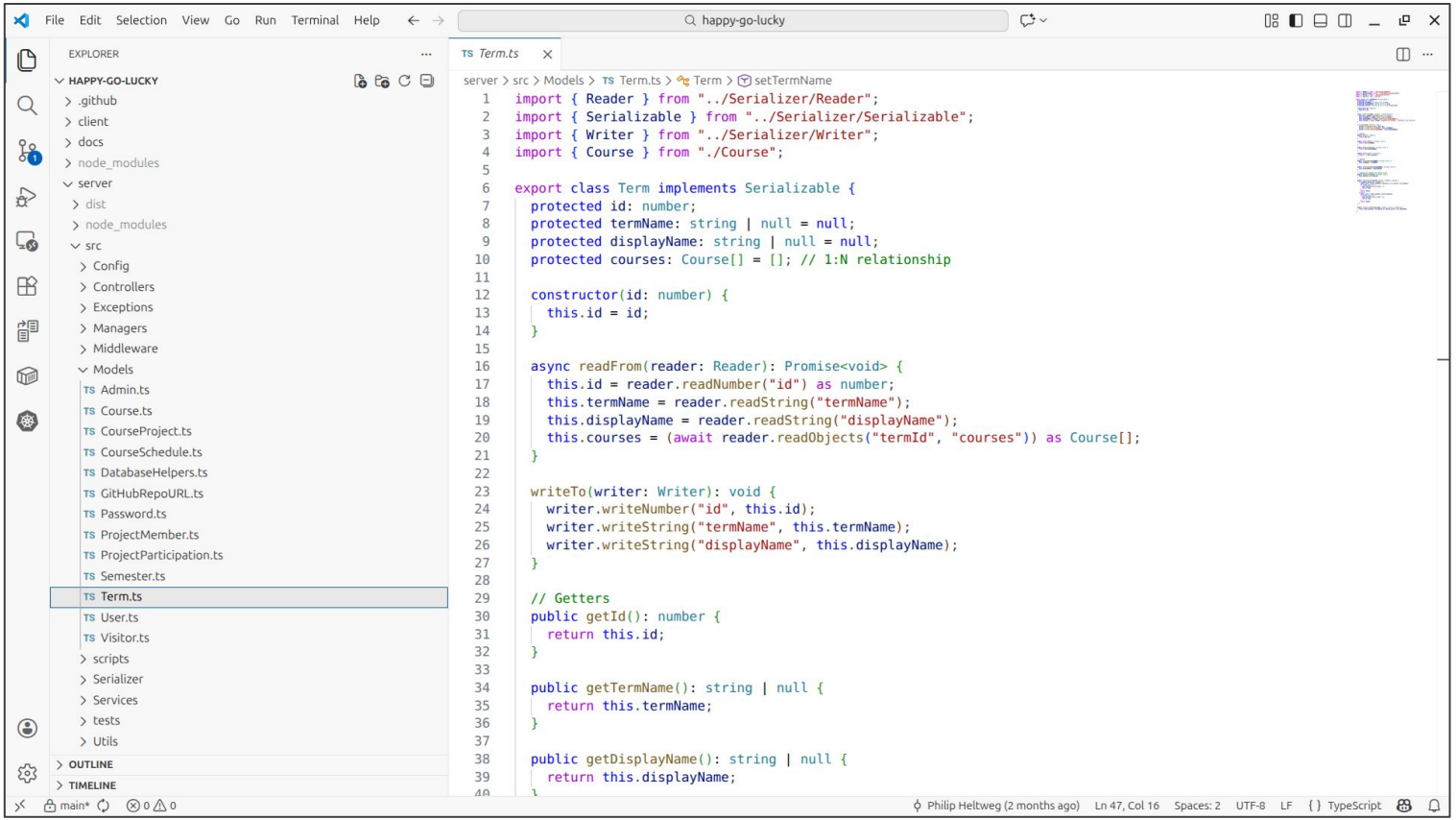
Original HGL Solutions

For a given project, a member

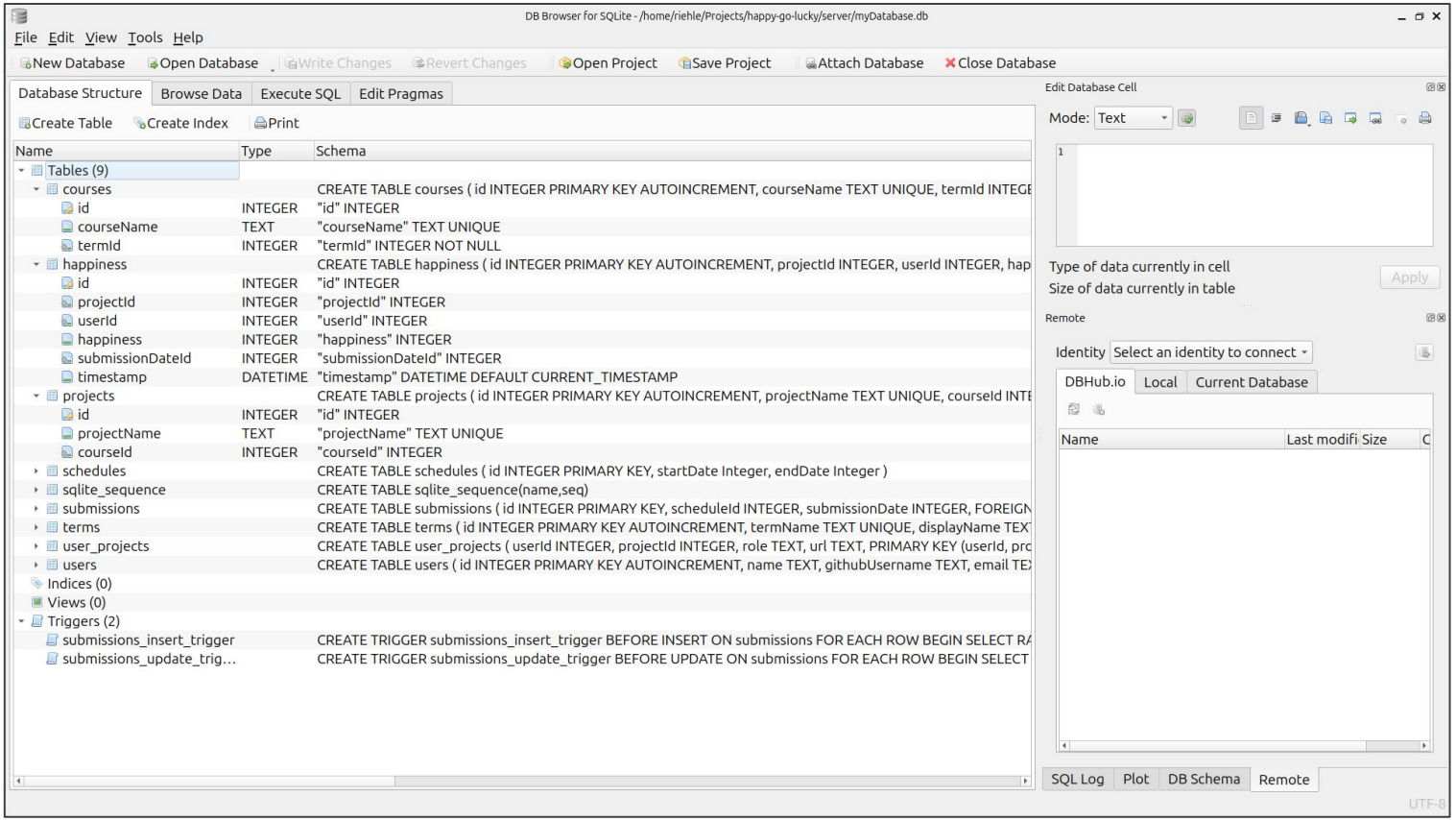
- Can enter their happiness
- Can send out stand-up emails
- Can review their coding activities

all scoped by the course schedule

Use Your Fav IDE But Keep Cruft Out



SQLite Database Browser (myDatabase.db)



Optional Homework

See this table (same as project, but use table)

https://docs.google.com/spreadsheets/d/1Td5wvsGZAJnNZjEvm8cfsthcylOhCpth3aT8vSCUZk0/edit?usp=sharing

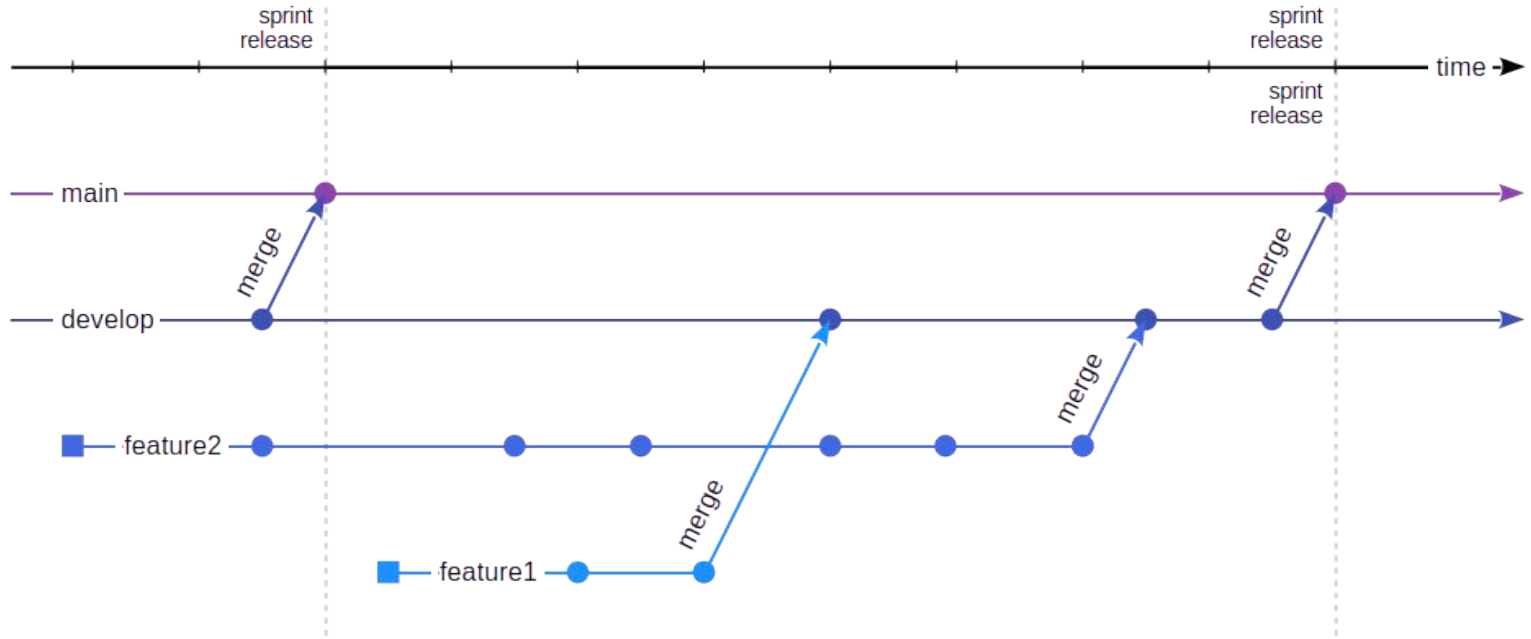
You will not be penalized if you don’t participate

You’ll get extra credit if you participate

Extra credit might lift your grade if you are at a grade boundary

View this experience as an idea of your first industry job

Working With Branches



Homework Work and Git Flow

1. Fork and work from your own repository
2. Pick and mark item to work on (spreadsheet)
3. Create a branch and a WIP pull request where you explain what you are going to do (to get feedback early on, discuss design alternatives, …)
   a. Source branch: feature branch on your repo
   b. Target branch: **develop-ADAP-12** on **riehlegroup/happy-go-lucky**
   c. Tag **@georg-schwarz** for feedback
4. Semantically chunk work into commits
5. Make sure the checks (GitHub Actions) run through successfully
6. When done, remove WIP status
7. Wait and incorporate code review feedback (amend pull request)
8. Your work may or may not be integrated

Thank you! Any questions?

dirk.riehle@fau.de – https://oss.cs.fau.de

dirk@riehle.org – https://dirkriehle.com – @dirkriehle

Legal Notices

License

- Licensed under the CC BY 4.0 International license

Copyright

- © 2012-2026 Dirk Riehle, some rights reserved